

КВАЛІФІКАЦІЙНА РОБОТА

Програмно-технічний засіб керування дистанційним металошукачем на основі мікроконтролера ESP32

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.2301109.23.01.43 ПЗ

Виконав здобувач III курсу, група КІ2с-23-1


Підпис

Віталій КОЛЕСНИКОВ

Ініціали, прізвище

Керівник

Науковий ступінь, учене звання


Підпис

Сергій ЛИСЕНКО

Ініціали, прізвище

Нормоконтролер

Науковий ступінь, учене звання


Підпис

Сергій ЛИСЕНКО

Ініціали, прізвище

До захисту допускаю:
завідувач кафедри КІС
«01» червня 2026 р.


Підпис

Ольга ПАВЛОВА

Ініціали, прізвище

дата

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Факультет ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

Кафедра КОМП'ЮТЕРНОЇ ІНЖЕНЕРІЇ ТА ІНФОРМАЦІЙНИХ СИСТЕМ

Рівень вищої освіти ПЕРШИЙ (БАКАЛАВРСЬКИЙ)

Галузь знань 12 ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ

Спеціальність 123 КОМП'ЮТЕРНА ІНЖЕНЕРІЯ

Освітня програма «КОМП'ЮТЕРНА ІНЖЕНЕРІЯ ТА ПРОГРАМУВАННЯ»

ЗАТВЕРДЖУЮ

Завідувачка кафедри КІС

 Ольга ПАВЛОВА

“ 10 ” 01 2026 р.

ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

Колеснікову Віталію Руслановичу

Прізвище, ім'я, по батькові студента

1. Тема проекту (роботи) Програмно-технічний засіб керування дистанційним металошукачем на основі мікроконтролера ESP32

Керівник проекту (роботи) Лисенко Сергій Миколайович, д.т.н., проф.

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

Затверджена наказом ректора університету від 20.01.2026 р. № 7

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) _____

Програмно-технічний засіб керування дистанційним металошукачем на основі мікроконтролера ESP32 та постановка задачі щодо її удосконалення

Проектування програмно-технічного засобу керування дистанційним металошукачем на основі мікроконтролера ESP32

Програмно-апаратна реалізація програмно-технічного засобу керування дистанційним металошукачем на основі мікроконтролера ESP32

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

Схема електрична функціональна

Схема електрична принципова

Блок-схема алгоритму програмного забезпечення

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)	Примітки
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 програмно-технічного засобу керування дистанційним металошукачем на основі мікроконтролера ESP32	01.04.2026	виконано
5	Робота над розділом 3 – проектування програмно-технічного засобу керування дистанційним металошукачем на основі мікроконтролера ESP32	29.04.2026	виконано
6	Оформлення пояснювальної записки згідно вимог	25.05.2026	виконано
7	Попередній захист ВКР	26.05.2026	виконано
8	Захист ВКР на засіданні ЕК	Червень 2026 року	

Здобувач


Підпис

Віталій КОЛЕСНИКОВ

Імя, ПРІЗВИЩЕ

Керівник кваліфікаційної роботи


Підпис

Сергій ЛИСЕНКО

Імя, ПРІЗВИЩЕ

АНОТАЦІЯ

Тема кваліфікаційної роботи: «Програмно-технічний засіб керування дистанційним металошукачем на основі мікроконтролера ESP32».

Автор роботи: Віталій КОЛЕСНИКОВ.

Керівник роботи: Сергій ЛИСЕНКО.

Пояснювальна записка: 71 с., 20 рис., 6 дод., 59 джерел.

Графічна частина: 4 креслення.

АВТОМАТИЧНЕ КАЛІБРУВАННЯ, ВЕБ-ІНТЕРФЕЙС, ДИСТАНЦІЙНЕ КЕРУВАННЯ, МЕТАЛОШУКАЧ, МІКРОКОНТРОЛЕР, ПРОГРАМНО-ТЕХНІЧНИЙ ЗАСІБ, ESP32, WI-FI.

Кваліфікаційна робота бакалавра присвячена розробленню програмно-технічного засобу керування дистанційним металошукачем на основі мікроконтролера ESP32. Актуальність теми зумовлена потребою підвищення безпеки та зручності використання металошукачів у ситуаціях, де важливими є віддалений доступ до результатів вимірювання, гнучке налаштування режимів роботи та можливість оперативного контролю параметрів системи.

Метою роботи є проектування, програмна реалізація та перевірка працездатності системи дистанційного керування металошукачем на базі ESP32. У процесі виконання роботи проаналізовано призначення та сфери застосування металошукачів, розглянуто існуючі технічні рішення, обґрунтовано вибір апаратної платформи й допоміжних компонентів, розроблено структурну, функціональну та електричну принципову схеми пристрою, а також реалізовано програмну логіку цифрової обробки сигналу й автоматичної калібровки. У роботі описано експериментальний зразок системи, локальний інтерфейс на базі OLED-дисплея, режими мережевої взаємодії через Wi-Fi, збір супровідних даних із сенсорів та формування результатів вимірювання для подальшого аналізу.



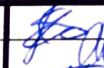
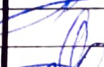


Підпис здобувача

30.05.2026

Дата

ЗМІСТ

Вступ	4
1 Аналіз предметної області та існуючих рішень у сфері металошукачів з дистанційним керуванням	6
1.1 Призначення та сфери застосування металошукачів	6
1.2 Класифікація та принципи роботи сучасних металошукачів	10
1.3 Огляд існуючих технічних рішень та підходів до побудови металошукачів	13
1.4 Аналіз способів дистанційного передавання даних та керування	17
1.5 Постановка задачі створення програмно-технічного засобу дистанційного керування металошукачем	21
2 Проектування програмно-технічного засобу дистанційного керування металошукачем	23
2.1 Принципи побудови та взаємодія функціональних вузлів системи	23
2.2 Обґрунтування вибору апаратної платформи та компонентів системи	29
2.3 Розроблення структурної та електричної принципової схеми системи	36
2.4 Принципи обробки сигналу та автоматичної калібровки металошукача	43
2.5 Організація взаємодії з користувачем та обробки даних	46
2.6 Реалізація програмної логіки системи	48
3 Реалізація програмно-технічного засобу дистанційного керування металошукачем	51
3.1 Загальний опис експериментального зразка системи	51
3.2 Демонстрація роботи системи та режимів функціонування	55
3.3 Аналіз обробки сигналу та результатів вимірювань	60
3.4 Перевірка ефективності автоматичної калібровки	65

КвРКІ.2301109.23.01.43 ПЗ								
Зм.	Арк.	№ док.ум.	Підпис	Дата	Програмно-технічний засіб керування дистанційним металошукачем на основі мікроконтролера ESP32	Літера	Аркуш	Аркушів
Виконав		Віталій КОЛЕСНИКОВ		01.06.20		у		2
Перевір.		Сергій ЛИСЕНКО		01.06.20	Пояснювальна записка	ХНУ КІ2с-23-1		
Н.контр.		Сергій ЛИСЕНКО		01.06.20				
Затвер.		Ольга ПАВЛОВА		01.06.20				

3.5 Аналіз взаємодії з користувачем та передачі даних	69
Висновки.....	74
Перелік джерел посилань	75
Додаток А Копія креслення «Схема електрична функціональна»	82
Додаток Б Копія креслення «Схема електрична принципова»	83
Додаток В Копія креслення «Блок-схема алгоритму програмного забезпечення».....	84
Додаток Г Перелік елементів до схеми електричної принципової	86
Додаток Д Графічний інтерфейс користувача веб-застосунку	89
Додаток Е Текст системного програмного забезпечення	92

ВСТУП

Сучасний розвиток вбудованих систем і бездротових технологій вже призвів до широкого впровадження компактних пристроїв, у яких поєднано апаратну частину та програмну логіку керування. Унаслідок цього технічні засоби дедалі частіше розглядаються не як окремі автономні прилади, а як елементи більш складних інформаційних і керувальних систем. У сфері пошуку металевих об'єктів така тенденція має особливе значення, оскільки безпека оператора, зручність експлуатації та можливість віддаленої взаємодії з пристроєм набувають практичної цінності. Перехід від ізольованих аналогових приладів до інтелектуальних вузлів, здатних інтегруватися в загальний інформаційний простір, стає не просто технічною інновацією, а об'єктивною вимогою сучасних інженерних та безпекових стандартів. Така еволюція дозволяє суттєво мінімізувати вплив людського фактора під час виконання рутинних або небезпечних пошукових робіт.

Металошукачі широко застосовуються у сфері безпеки, інженерних обстеженнях, пошукових роботах і побутових задачах. Водночас значна частина наявних рішень орієнтована на локальне використання без розвинених засобів дистанційного керування та передавання даних. Це ускладнює інтеграцію таких пристроїв у мобільні або роботизовані платформи та обмежує можливості контролю й налаштування під час роботи у важкодоступних або потенційно небезпечних умовах. Розвиток мікроконтролерних платформ із вбудованими засобами бездротового зв'язку створив передумови для побудови більш гнучких систем, у яких керування та моніторинг виконуються через мережеві інтерфейси без потреби у складній зовнішній інфраструктурі. Завдяки цьому оператор отримує змогу не знаходитися в безпосередній близькості до пошукової котушки, а дистанційно керувати процесом виявлення, зберігаючи при цьому повний контроль над чутливістю, калібруванням та режимами відображення інформації.

					КВРКІ.2301109.23.01.43 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		4

Актуальність теми зумовлена потребою підвищення рівня безпеки та зручності використання металошукачів, а також прагненням до створення універсальних програмно-апаратних рішень, здатних працювати у складі дистанційно керованих систем. Платформа ESP32 поєднує обчислювальні можливості та вбудовані інтерфейси бездротового зв'язку, що дозволяє реалізувати компактний і функціонально насичений пристрій із підтримкою мережевої взаємодії. Це відкриває можливість організації віддаленого доступу до параметрів роботи, передавання результатів вимірювань і побудови зручного інтерфейсу керування на базі стандартних мережевих технологій. Також цифровізація вимірів за допомогою ESP32 дає можливість застосовувати складні алгоритми програмної фільтрації та автоматичної калібровки, що значно покращує загальну надійність виявлення металевих об'єктів у середовищах із високим рівнем завад.

Об'єктом кваліфікаційної роботи визначено програмно-апаратні системи пошуку металевих об'єктів. Предметом кваліфікаційної роботи обрано методи та засоби реалізації дистанційного керування і передавання даних у складі металошукача на базі мікроконтролерної платформи. Метою кваліфікаційної роботи визначено створення програмно-технічного засобу керування дистанційним металошукачем, який забезпечує приймання сигналів, їх обробку та передавання інформації оператору з використанням бездротових технологій зв'язку. Для досягнення цієї мети у межах роботи опрацьовано аналіз існуючих підходів до побудови металошукачів і систем керування, сформовано вимоги до структури пристрою, спроектовано архітектуру програмно-апаратної частини, реалізовано програмне забезпечення та перевірено працездатність створеного засобу в умовах, наближених до практичної експлуатації.

У підсумку виконання кваліфікаційної роботи дозволяє отримати цілісне уявлення про принципи побудови сучасних дистанційно керованих вбудованих систем і продемонструвати можливості використання платформи ESP32 для створення функціонального металошукача з розширеними можливостями керування та моніторингу.

					КВРКІ.2301109.23.01.43 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		5

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ІСНУЮЧИХ РІШЕНЬ У СФЕРІ МЕТАЛОШУКАЧІВ З ДИСТАНЦІЙНИМ КЕРУВАННЯМ

1.1 Призначення та сфери застосування металошукачів

Металошукачі на сьогодні посідають важливе місце серед технічних засобів, що застосовуються для виявлення прихованих, малодоступних або візуально непомітних металевих об'єктів у різних умовах експлуатації. Розвиток елементної бази, поява доступних мікроконтролерних платформ, здешевлення сенсорних компонентів і вдосконалення методів обробки сигналів призвели до того, що такі пристрої перестали сприйматися виключно як вузькоспеціалізоване обладнання. У сучасних умовах металошукачі вже використовуються не лише у професійній сфері, а й у побутових, напівпрофесійних, навчальних та експериментальних задачах, де важливо швидко отримати відомості про наявність металевих предметів у ґрунті, конструкціях, під покриттям або всередині різних матеріалів.

За своїм призначенням металошукач є засобом первинного виявлення, який дозволяє фіксувати наявність металу без механічного руйнування середовища та без прямого контакту з об'єктом. Саме ця властивість і визначила його широку затребуваність. У багатьох ситуаціях візуальний контроль є недостатнім, оскільки об'єкт може знаходитися під шаром ґрунту, під облицюванням, під водою, всередині стіни або серед сторонніх включень. У таких випадках металошукач виконує роль проміжного технічного інструмента між середовищем і людиною, дозволяючи ще до початку активних дій оцінити, чи є в зоні контролю металеві елементи, наскільки ймовірною є їх присутність і чи доцільно продовжувати роботу в обраній точці. Це особливо важливо там, де помилка може призвести не лише до втрати часу, а й до пошкодження конструкцій, небезпечної ситуації або прямої загрози для оператора.

Однією з найбільш відповідальних сфер застосування металошукачів залишається безпека, зокрема роботи, пов'язані з пошуком вибухонебезпечних предметів, уламків боєприпасів, мін, металевих корпусів небезпечних залишків

					КВРКІ.2301109.23.01.43 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		6

озброєння та інших потенційно ризикованих об'єктів [1-6]. У таких умовах на перший план виходить не лише чутливість пристрою, а й стабільність його роботи, відтворюваність результатів та можливість зменшити прямий контакт людини з небезпечною зоною. Саме тут особливо чітко проявляється потреба в рішеннях, які дозволяють виконувати пошук на відстані, автоматизувати частину дій оператора та організувати передавання даних без постійного перебування людини поруч із пошуковою катушкою. У цьому контексті металошукач уже розглядається не просто як окремих вимірювальний прилад, а як частина більш складної технічної системи, що може функціонувати у складі дистанційно керованих платформ, роботизованих візків або мобільних наземних комплексів.

Не менш важливою є роль металошукачів у сфері археології та пошуково-історичної діяльності, де їх застосування пов'язане з виявленням монет, прикрас, фрагментів спорядження, залишків стародавніх конструкцій та інших об'єктів, що мають культурну, історичну або наукову цінність [7, 8]. У таких задачах особливо важливою стає не лише сама здатність до виявлення металу, а й стабільність роботи на тривалих ділянках пошуку, точність локалізації, можливість регулювання чутливості та загальна зручність експлуатації. Для археологічного чи пошукового використання важливо, щоб пристрій не лише реагував на металевий об'єкт, а й дозволяв працювати довго, без надлишкової втоми оператора, без великої кількості хибних спрацювань і з можливістю коректного налаштування під тип місцевості. Саме тому в цій сфері особливо цінуються рішення, що поєднують апаратну точність із продуманим інтерфейсом і гнучкістю керування.

У промисловій та інженерній практиці металошукачі застосовуються для контролю якості продукції, перевірки сировини та готових виробів, пошуку прихованих металевих елементів у будівельних конструкціях, технічних обстеженнях інженерних об'єктів, а також для аналізу ґрунтів і водного середовища на наявність важких металів та інших елементів [9-12]. У таких випадках пристрій уже виконує роль інструмента технічного контролю, де особливого значення набуває не просто факт виявлення, а можливість

					КВРКІ.2301109.23.01.43 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		7

інтегрувати результат у ширшу систему вимірювань, документування та подальшого аналізу. Для виробничих і технічних задач важливо, щоб результат можна було передати, зберегти, порівняти з попередніми вимірюваннями або включити до загального звіту. Саме це поступово зміщує акцент від простих автономних приладів до програмно-апаратних рішень, у яких металошукач доповнюється цифровою логікою, засобами збереження даних і мережевими інтерфейсами.

Окрему нішу становить будівельна та комунікаційна сфера, де металошукачі застосовуються під час виявлення армування, металевих труб, прихованих кабельних ліній, монтажних елементів і закладних деталей усередині стін, перекриттів або підземних комунікацій. У таких умовах особливо важливо уникати помилкових дій під час свердління, демонтажу або земляних робіт. Виявлення прихованого металу до початку втручання дозволяє зменшити ймовірність аварій, уникнути пошкодження конструктивних елементів і краще планувати технічні операції. На практиці це означає, що металошукач виконує не лише пошукову, а й профілактичну функцію, оскільки допомагає попередити небажані наслідки ще до моменту виконання основної роботи.

Поширеним залишається і побутове та аматорське використання металошукачів. Пошук загублених предметів, монет, металевих деталей у ґрунті, визначення місця проходження елементів інженерних комунікацій на приватних ділянках або використання в межах технічної творчості вже давно перестали бути поодинокими прикладами. У цій сфері попит формують доступність, компактність, невисока вартість і простота використання. Водночас навіть у побутовому сегменті поступово зростає інтерес до рішень, що мають розширені можливості: цифрове відображення параметрів, збереження результатів, підтримку декількох режимів роботи, підключення до смартфона чи бездротове керування. Це означає, що навіть користувачі непрофесійного рівня все частіше орієнтуються не на найпростіші прилади з однією світлодіодною індикацією, а на більш функціональні системи, здатні адаптуватися до різних сценаріїв використання.

					КВРКІ.2301109.23.01.43 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		8

Не менш показовою є й навчальна сфера, де металошукачі застосовуються як приклад поєднання аналогової електроніки, сенсорних вузлів, мікроконтролерного керування та програмної обробки сигналів. У межах технічної підготовки такі пристрої дозволяють наочно продемонструвати, яким чином фізичне явище електромагнітної взаємодії перетворюється на цифровий сигнал, що надалі аналізується програмними засобами. Саме тому металошукачі часто використовуються як зручна навчальна платформа для відпрацювання навичок схемотехніки, роботи з аналого-цифровим перетворенням, калібровки, побудови інтерфейсів та організації передачі даних. У цьому випадку вони мають цінність не лише як пристрої практичного пошуку, а й як об'єкти інженерного опрацювання.

Варто також відзначити, що в сучасних умовах сфери застосування металошукачів дедалі частіше перетинаються з напрямками робототехніки, Інтернету речей та вбудованих систем. Якщо раніше пристрій сприймався переважно як автономний ручний прилад, то тепер усе більше уваги приділяється рішенню, у якому результат пошуку передається на мобільний пристрій, зберігається у структурованому вигляді, доповнюється координатами, сенсорними параметрами та використовується як частина єдиного інформаційного контуру. Це особливо актуально для віддаленого контролю, експериментальних вимірювальних систем та ситуацій, де важливо не просто зафіксувати факт наявності металу, а й документувати результати, аналізувати їх у часі та пов'язувати з іншими даними.

Через це сучасний металошукач дедалі частіше набуває ознак багатофункціонального програмно-апаратного засобу. У ньому поєднуються не лише вимірювальні вузли, а й засоби цифрової обробки, локальний або мережевий інтерфейс, модулі геолокації, збереження та передавання інформації. Для окремих сценаріїв використання таке поєднання є вже не перевагою, а необхідністю. Наприклад, у дистанційно керованих системах важливим стає саме передавання результатів без безпосереднього доступу до основного блока, а в аналітичних і картографічних задачах – можливість фіксувати координати й

					КВРКІ.2301109.23.01.43 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		9

формувати структуровані набори даних. Унаслідок цього розвиток металошукачів закономірно рухається в бік інтеграції з мікроконтролерними платформами та бездротовими інтерфейсами.

У підсумку сфери застосування металошукачів охоплюють широкий спектр завдань – від критично важливих робіт у сфері безпеки до побутових, навчальних, технічних та експериментальних сценаріїв. Саме така широта використання дозволяє розглядати металошукач як універсальний програмно-апаратний засіб, у якому поєднуються вимірювальні, обчислювальні та комунікаційні можливості. Багатофункціональність подібних пристроїв формує підґрунтя для впровадження дистанційного керування, цифрової обробки та бездротового передавання даних, що повністю відповідає сучасним тенденціям розвитку вбудованих систем і визначає напрям подальшого вдосконалення таких рішень у межах кваліфікаційної роботи.

1.2 Класифікація та принципи роботи сучасних металошукачів

Сучасні металошукачі сформувалися як окремий клас вимірювальних пристроїв, у яких поєднано електромагнітні методи з цифровою обробкою сигналів. За час розвитку елементної бази та схемотехніки склалося кілька основних підходів до побудови таких систем, кожен з яких має власні особливості, переваги та обмеження. У межах цієї кваліфікаційної роботи розглянуто найпоширеніші принципи, що використовуються у практичних пристроях, оскільки саме вони визначають як можливості пошуку, так і вимоги до апаратної та програмної частин.

Найбільш відомим і поширеним підходом є використання низькочастотних індукційно-балансних схем, які часто позначаються як VLF-системи [13-15]. У таких металошукачах застосовується дві або більше котушок, між якими сформовано електромагнітний баланс [16]. Одна з котушок формує змінне магнітне поле, а інша приймає сигнал, що змінюється під впливом металевих об'єктів. За відсутності сторонніх предметів система перебуває у стані, близькому

					КВРКІ.2301109.23.01.43 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		10

до балансу, а поява металу в зоні дії поля порушує цей стан і призводить до появи різниці сигналів. Отримана різниця підсилюється, фільтрується та перетворюється у форму, зручну для подальшої обробки або індикації. Такий підхід дозволяє не лише виявляти наявність металу, а й у певних межах виконувати дискримінацію, тобто відрізнити різні типи металів за фазовими або амплітудними характеристиками сигналу. Разом із тим подібні системи є чутливими до мінералізації ґрунту та вимагають точного налаштування котушок і балансування схеми.

Іншим поширеним напрямом стали імпульсно-індукційні металошукачі, у яких принцип роботи базується на формуванні коротких імпульсів струму в пошуковій котушці та аналізі процесу затухання індукованих струмів у навколишньому середовищі [17, 18]. Після подачі імпульсу в котушці формується магнітне поле, яке збуджує вихрові струми в металевих об'єктах. Ці струми, у свою чергу, створюють власне поле, що впливає на сигнал, який реєструється приймальним трактом. Характер затухання цього сигналу залежить від електропровідності та геометричних параметрів об'єкта, що й використовується для виявлення металу [19]. Такий принцип забезпечує відносну простоту схемної реалізації та високу стійкість до впливу ґрунту, через що імпульсно-індукційні системи часто застосовуються у складних умовах, зокрема на сильно мінералізованих або вологих ґрунтах [20, 21]. Водночас можливості дискримінації у подібних системах зазвичай є обмеженими, а основний акцент робиться саме на факті виявлення об'єкта.

Окрему групу становлять високочастотні та надвисокочастотні системи, у яких використовуються радіочастотні коливання та аналіз змін параметрів резонансних контурів або відбитих сигналів. Подібні рішення зустрічаються рідше у побутових металошукачах, проте знаходять застосування у спеціалізованих приладах для технічного контролю або промислових задач. Принцип дії тут ґрунтується на тому, що присутність металевого об'єкта змінює електромагнітні характеристики простору поблизу антени або котушки, що призводить до зсуву частоти, фази чи амплітуди сигналу. Такі зміни фіксуються

					КВРКІ.2301109.23.01.43 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		11

приймальним трактом і перетворюються у керуючий або інформаційний сигнал. Подібні підходи вимагають більш складної апаратної частини та ретельної цифрової обробки, проте відкривають можливості для підвищення точності та стабільності вимірювань у спеціалізованих умовах.

Незалежно від конкретного принципу побудови, у сучасних металошукачах чітко простежується тенденція до поєднання аналогових вимірювальних трактів із цифровими засобами обробки сигналів. Первинний сигнал, отриманий з котушки або антени, проходить етапи підсилення та фільтрації, після чого перетворюється аналого-цифровим перетворювачем у цифрову форму. Далі обробка виконується програмними засобами мікроконтролера або вбудованого процесора, де реалізуються алгоритми згладжування, виявлення корисного сигналу на фоні шумів, оцінювання рівня сигналу та формування даних для індикації або передавання. Такий підхід дозволяє гнучко змінювати логіку роботи пристрою без втручання в апаратну частину та адаптувати систему під різні умови експлуатації.

Важливу роль у класифікації металошукачів відіграє і конструкція пошукової котушки, оскільки саме вона визначає форму електромагнітного поля та зону чутливості пристрою. Зміна геометричних розмірів, кількості витків і способу екранування безпосередньо впливає на глибину виявлення, роздільну здатність і стійкість до завад. У практичних системах часто застосовуються змінні котушки, що дозволяє адаптувати пристрій до конкретних задач, наприклад, до пошуку великих об'єктів на значній глибині або до більш точного виявлення дрібних предметів поблизу поверхні.

У підсумку сучасні металошукачі є багаторівневими програмно-апаратними системами, де електромагнітне виявлення поєднано з цифровою обробкою. Така еволюція призвела до того, що вибір конкретного класу пристрою визначається не лише фізичними властивостями об'єктів пошуку, а й вимогами до інтерфейсу, способів передавання даних і можливостей інтеграції з іншими технічними засобами. Це створює передумови для використання

					КВРКІ.2301109.23.01.43 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		12

мікроконтролерів і переходу до рішень, де металошукач є частиною дистанційно керованої системи.

1.3 Огляд існуючих технічних рішень та підходів до побудови металошукачів

Сучасні технічні рішення у сфері металошукачів охоплюють широкий спектр пристроїв - від простих переносних приладів для побутового використання до професійних систем контролю та модульних програмно-апаратних комплексів [13, 14]. Кожен із цих напрямів сформувався під впливом різних вимог до глибини виявлення, стабільності роботи, зручності експлуатації та можливостей інтеграції з іншими технічними засобами [9, 10].

У сегменті серійних переносних металошукачів поширення набули універсальні прилади, орієнтовані на пошукові роботи в польових умовах. Характерним прикладом такого підходу є металошукач Garrett ACE 250, у якому реалізовано низькочастотний принцип роботи з цифровою обробкою сигналу та базовою дискримінацією металів [15, 16, 19]. Подібні пристрої мають завершену конструкцію, оснащуються дисплеєм і набором режимів роботи, що дозволяє оператору змінювати чутливість без втручання в апаратну частину. Зовнішній вигляд металошукача Garrett ACE 250 зображено на рисунку 2.1.



Рисунок 1.1 – Металошукач Garrett ACE 250 [51]

					КВРКІ.2301109.23.01.43 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		13

У таких рішеннях основний акцент зроблено на зручність використання та стабільність роботи в типових умовах експлуатації, а функціональність обмежується заздалегідь визначеним набором можливостей.

Професійний сегмент представлений системами, що застосовуються у сфері безпеки та промислового контролю, де важливою є не лише чутливість, а й повторюваність результатів та стійкість до завад [7, 8]]. Як приклад такого підходу можна розглядати рамкові металодетектори CEIA для контролю доступу, які використовуються в аеропортах та на об'єктах із підвищеними вимогами до безпеки [11]. У подібних системах апаратна частина поєднується з програмними алгоритмами фільтрації та калібрування, що забезпечує стабільну роботу в умовах змінного електромагнітного фону. Приклад такого стаціонарного металодетектора зображено на рисунку 1.2.



Рисунок 1.2 – Металодетектор CEIA [52]

Основна роль програмної складової полягає у зменшенні кількості хибних спрацювань і підтриманні заданих параметрів контролю протягом тривалого часу експлуатації [4].

Окремий напрям становлять модульні та напівпрофесійні конструкції, що будуються на базі простих імпульсно-індукційних схем [18]. Типовим

представником такого підходу є плата MD PIRAT, яка широко використовується в аматорських і навчальних проєктах [21, 22]. Зовнішній вигляд цієї плати наведено на рисунку 1.3.

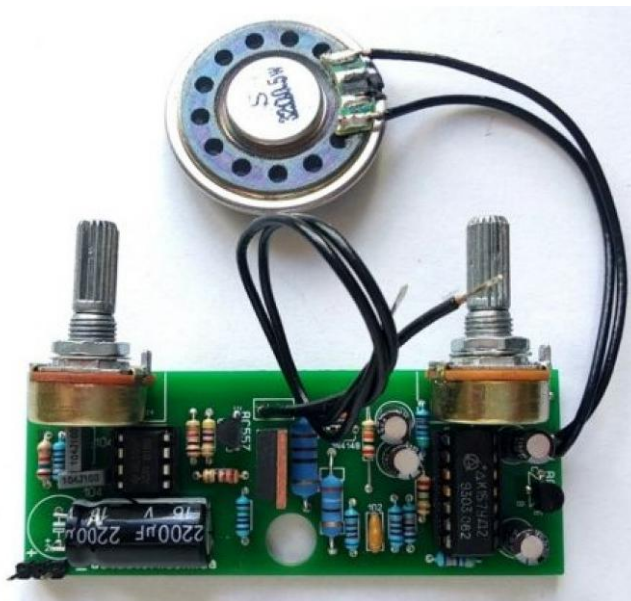


Рисунок 1.3 – Плата MD PIRAT [53]

У подібних рішеннях формування імпульсів і приймання сигналу реалізується аналоговими засобами, а індикація результатів пошуку зазвичай обмежується звуковим або світловим сигналом. Перевагою такого підходу є простота повторення та можливість швидкої зміни параметрів, наприклад, шляхом заміни котушки або налаштування порогів підсилювача. Водночас функціональні можливості подібних пристроїв залишаються обмеженими, оскільки відсутні розвинені засоби цифрової обробки та збереження даних.

Подальший розвиток металошукачів пов'язаний із використанням мікроконтролерних платформ для розширення функціональності [23, 24]. У цій групі як приклад можна навести рішення на базі Arduino Uno у поєднанні з радіомодулем NRF24L01, де аналоговий тракт виявлення металу доповнюється цифровим модулем керування та передавання даних [25-28].

У такій архітектурі сигнал з підсилювального тракту оцифровується, після чого обробляється програмними засобами мікроконтролера, а результати можуть

передаватися на віддалений приймач. Це дозволяє реалізувати базові елементи дистанційного керування та моніторингу, проте потребує додаткових модулів зв'язку і більш складного налаштування каналу передачі даних. Зовнішній вигляд мікроконтролерної плати Arduino Uno наведено на рисунку 1.4.

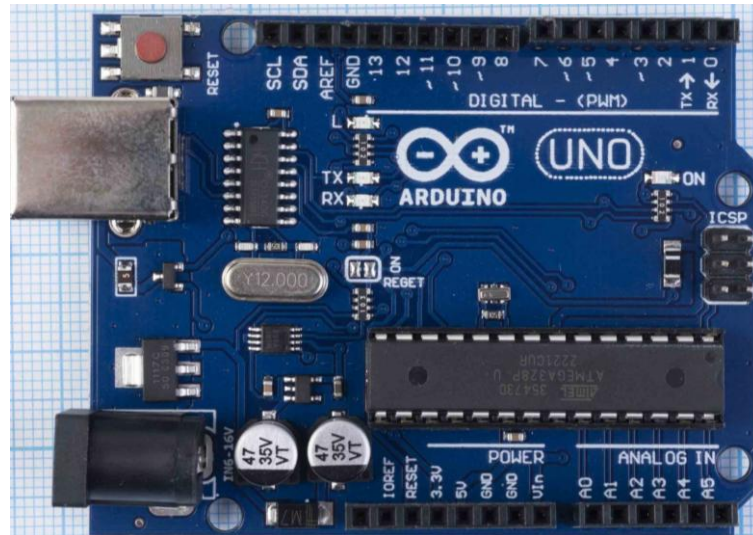


Рисунок 1.4 – Мікроконтролер Arduino Uno [54]

Окрему увагу в сучасних рішеннях привертають системи на базі мікроконтролерів із вбудованими мережевими інтерфейсами [29-31]. Характерним прикладом такого підходу є використання платформи ESP32 з організацією керування через веб-інтерфейс по Wi-Fi [32-36]. У подібних системах металошукач виступає джерелом вимірювальної інформації, а мікроконтролер забезпечує не лише обробку сигналів, а й формування мережевого інтерфейсу для відображення результатів і зміни параметрів роботи. Перевагою такого рішення є універсальність доступу, оскільки для взаємодії з пристроєм достатньо стандартного браузера на смартфоні або комп'ютері, без потреби у спеціалізованому програмному забезпеченні.

У підсумку огляд існуючих технічних рішень показує, що розвиток металошукачів відбувається у напрямі переходу від простих автономних пристроїв до програмно-апаратних систем із цифровою обробкою та бездротовими інтерфейсами [20]. Кожен із розглянутих сегментів, від серійних

переносних приладів і професійних систем контролю до модульних імпульсних схем і мікроконтролерних рішень на базі ESP32 - демонструє власний підхід до реалізації задачі виявлення металу. Це дозволяє розглядати сучасний металошукач не лише як окремий прилад, а як елемент більш складної дистанційно керованої системи, що відповідає актуальним тенденціям розвитку вбудованих та мережевих технологій [6, 12].

1.4 Аналіз способів дистанційного передавання даних та керування

Розвиток дистанційно керованих технічних систем зумовив активне впровадження бездротових каналів зв'язку у складі вимірювальних і керувальних пристроїв [37]. Для металошукачів це має особливе значення, оскільки можливість відокремити модуль пошуку від робочого місця оператора підвищує безпеку, зручність експлуатації та гнучкість використання пристрою. У сучасних умовах дистанційне передавання даних і керування може реалізовуватися різними способами, що відрізняються за дальністю дії, швидкістю обміну, складністю інтеграції та вимогами до апаратних ресурсів.

Одним із найпростіших і водночас поширених підходів є використання радіоканалів на базі спеціалізованих радіомодулів, наприклад сімейства NRF24L01 [38-42]. Зовнішній вигляд такого модуля наведено на рисунку 1.5.



Рисунок 1.5 – Радіомодуль NRF24L01 [55]

Подібні модулі дозволяють організувати двосторонній обмін даними між двома мікроконтролерними пристроями на відстані в десятки або навіть сотні метрів за умови використання відповідних антен. Перевагою такого підходу є відносно низьке енергоспоживання, простота апаратної інтеграції та достатня швидкість обміну для передавання керуючих команд і базових вимірювальних даних. Разом із тим застосування радіомодулів потребує окремої реалізації протоколу обміну, обробки помилок передавання та синхронізації, що ускладнює програмну частину системи і знижує універсальність рішення з точки зору взаємодії з користувацькими пристроями.

Іншим поширеним варіантом є використання технології Bluetooth, яка орієнтована насамперед на безпосередню взаємодію з мобільними пристроями та персональними комп'ютерами [3, 43, 44]. Модулі типу HC-05 або HC-06 дозволяють відносно просто організувати канал зв'язку між металошукачем і смартфоном чи ноутбуком, використовуючи стандартні засоби операційної системи. Загальний вигляд модуля Bluetooth HC-05 наведено на рисунку 1.6.

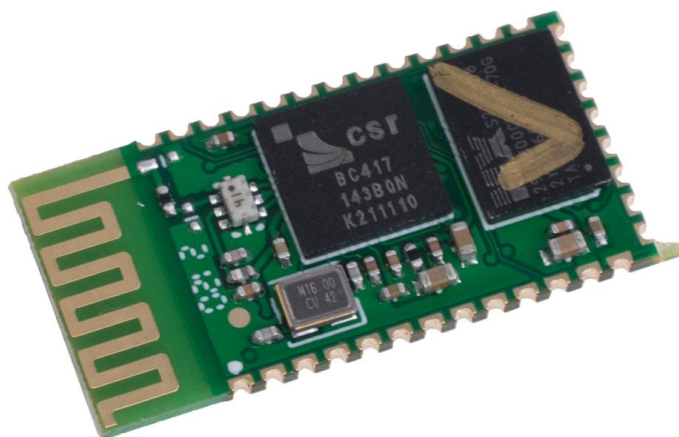


Рисунок 1.6 – Модуль Bluetooth HC-05 [56]

Перевагою такого підходу є простота підключення та мінімальні вимоги до налаштування мережевої інфраструктури. Слід окремо зауважити, що сучасні версії починаючи з Bluetooth 4.0 і вище, підтримують технологію Bluetooth Low Energy (BLE), яка вирішує проблему високого енергоспоживання, притаманну класичному Bluetooth. Це робить їх ефективними для портативних пристроїв з

акумуляторним живленням, де важлива тривала автономна робота. Однак, незважаючи на це, дальність дії Bluetooth зазвичай обмежується кількома десятками метрів, а стабільність з'єднання може залежати від умов середовища та завантаженості радіоефіру. Крім того, для повноцінної роботи часто потрібне спеціалізоване прикладне програмне забезпечення на стороні користувацького пристрою, що знижує універсальність системи.

Більш універсальним і функціонально гнучким підходом є використання бездротових мереж Wi-Fi, зокрема на базі мікроконтролерних платформ із вбудованими мережевими інтерфейсами, таких як ESP8266 або ESP32 [45-49].

Раніше одуль ESP8266 став революційним рішенням, оскільки забезпечив надзвичайно дешевий і простий спосіб додавання Wi-Fi функціоналу до мікроконтролерних проектів. Він має достатню обчислювальну потужність для виконання базових завдань та інтегрований стек протоколів TCP/IP. Зовнішній вигляд мікроконтролера ESP8266 наведено на рисунку 1.7.

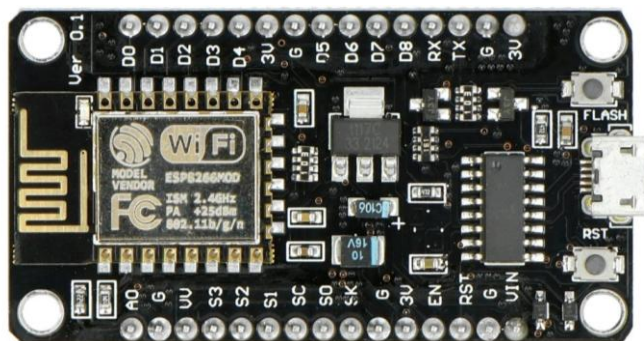


Рисунок 1.7 – Мікроконтролер ESP8266 [57]

Вдосконаленим продовженням цієї лінійки став мікроконтролер ESP32, який пропонує суттєво ширші можливості для розроблення комплексних систем. На відміну від свого попередника, ESP32 має значно вищу обчислювальну потужність завдяки двоядерному процесору з тактовою частотою до 240 МГц та значно більший обсяг оперативної пам'яті (SRAM). Важливою інновацією стала інтеграція двох бездротових стандартів на одному кристалі: мікроконтролер підтримує не лише Wi-Fi, але й класичний Bluetooth та його енергоефективну

версію Bluetooth Low Energy (BLE). Крім того, ESP32 володіє розширеною апаратною периферією, що включає більшу кількість портів загального призначення (GPIO), високоточні аналого-цифрові (АЦП) та цифро-аналогові перетворювачі (ЦАП), а також підтримку сучасних промислових інтерфейсів. Загальний вигляд платформи ESP32 подано на рисунку 1.8.

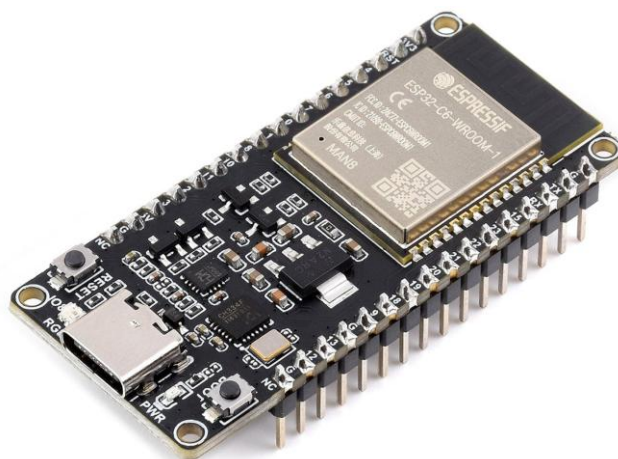


Рисунок 1.8 – Мікроконтролер ESP32 [58]

У цьому випадку металошукач може виступати як окремий мережевий вузол, що забезпечує передавання даних і приймання керуючих команд через стандартні мережеві протоколи. Однією з важливих переваг такого рішення є можливість реалізації керування через веб-інтерфейс, доступний із будь-якого пристрою з браузером без встановлення додаткового програмного забезпечення. Це спрощує взаємодію з системою, підвищує її масштабованість і дозволяє легко інтегрувати металошукач у більш складні інформаційні або керувальні комплекси.

Порівнюючи наведені підходи, можна відзначити, що радіомодулі типу NRF24L01 доцільно використовувати у випадках, коли необхідна проста й енергоефективна передача даних між двома пристроями. Bluetooth є зручним для швидкого підключення до смартфонів і комп'ютерів у межах невеликої відстані, однак його можливості обмежуються як за дальністю, так і за гнучкістю. Використання Wi-Fi, своєю чергою, відкриває можливість побудови більш

універсальних систем, де керування та моніторинг здійснюються через стандартні мережеві засоби, а сам пристрій може працювати як елемент локальної мережі або автономна точка доступу.

Окремо слід відзначити, що вибір способу дистанційного передавання даних безпосередньо впливає на архітектуру програмно-апаратної системи. Для радіоканалів і Bluetooth необхідно реалізовувати власні протоколи обміну та механізми синхронізації, тоді як мережеві рішення на базі Wi-Fi дозволяють використовувати готові стеки протоколів і стандартні підходи до організації клієнт-серверної взаємодії. Це спрощує розроблення, підвищує надійність обміну та розширює можливості подальшого розвитку системи [50].

У підсумку аналіз показує, що для задач дистанційного керування металошукачем найбільш перспективним є підхід із використанням мережевих технологій на базі Wi-Fi, оскільки він поєднує достатню дальність зв'язку, універсальність доступу та гнучкість у реалізації інтерфейсу керування. Разом із тим розглянуті альтернативні способи також залишаються актуальними для спеціалізованих або спрощених систем, де пріоритетами є мінімальне енергоспоживання чи простота апаратної реалізації.

1.5 Постановка задачі створення програмно-технічного засобу дистанційного керування металошукачем

Проведений у межах першого розділу аналіз призначення металошукачів, принципів їх роботи та існуючих технічних рішень показує, що сучасні пристрої цього класу поступово переходять від простих автономних приладів до більш складних програмно-апаратних систем [17].

Разом із тим значна частина наявних рішень або орієнтована на локальне використання без розвинених засобів взаємодії, або потребує застосування додаткових зовнішніх модулів і спеціалізованого програмного забезпечення для організації дистанційного керування та передавання даних, що обмежує гнучкість використання.

					КВРКІ.2301109.23.01.43 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		21

У зв'язку з цим постає задача створення програмно-технічного засобу, який поєднує функції виявлення металевих об'єктів із можливістю дистанційного керування та моніторингу параметрів роботи. Такий засіб має забезпечувати стабільне зчитування сигналів із вимірювального тракту, їх первинну обробку та подальше передавання у зручному для користувача вигляді.

Важливою вимогою є також можливість зміни режимів роботи та основних налаштувань без безпосереднього доступу до пристрою, що особливо актуально для застосування в потенційно небезпечних або важкодоступних умовах [1, 2, 5].

У межах кваліфікаційної роботи задача формулюється як розроблення програмно-технічного засобу дистанційного керування металошукачем, який забезпечує приймання сигналів від вимірювального тракту, їх цифрову обробку, відображення результатів і передавання керуючих команд між оператором і пристроєм.

Для досягнення поставленої мети у кваліфікаційній роботі необхідно вирішити комплекс взаємопов'язаних задач, що охоплюють аналіз вимог до системи, проектування її структурної схеми, вибір елементної бази, розроблення програмного забезпечення керування та обробки даних.

Окрім забезпечення базової функціональності, розроблювана система повинна відповідати сучасним вимогам щодо ергономіки та кросплатформності. Використання веб-інтерфейсу дозволить уникнути прив'язки до конкретної операційної системи та забезпечить повноцінне керування з будь-якого пристрою без встановлення додаткового програмного забезпечення. Також на етапі проектування слід врахувати необхідність оптимізації мережевого обміну даними. Це критично важливо для мінімізації затримок передавання сигналу в реальному часі між фізичним виявленням об'єкта котушкою та візуалізацією на екрані оператора.

У підсумку сформульована задача визначає напрям подальшого проектування та реалізації системи у наступних розділах кваліфікаційної роботи, де основна увага приділяється архітектурі, вибору апаратної платформи та програмній реалізації запропонованого рішення.

					КВРКІ.2301109.23.01.43 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		22

2 ПРОЄКТУВАННЯ ПРОГРАМНО-ТЕХНІЧНОГО ЗАСОБУ ДИСТАНЦІЙНОГО КЕРУВАННЯ МЕТАЛОШУКАЧЕМ

2.1 Принципи побудови та взаємодія функціональних вузлів системи

Основну увагу зосереджено на проєктуванні програмно-технічного засобу дистанційного керування металошукачем, у якому поєднано аналогові вузли формування та зчитування сигналу, цифрову обробку даних, допоміжні сенсори, локальний інтерфейс користувача та бездротову взаємодію з мобільним пристроєм. Такий підхід дозволяє розглядати розроблюваний пристрій не як окремий простий металошукач із базовою індикацією, а як багатокомпонентну систему, де кожен функціональний блок виконує власну задачу, а сукупна робота всіх блоків забезпечує виявлення металевих об'єктів, налаштування режимів, зчитування додаткових параметрів середовища та передавання результатів оператору у зручному вигляді.

Загальна архітектура системи сформована за модульним принципом. Саме такий принцип побудови в даному випадку є найбільш доцільним, оскільки дозволяє розділити систему на окремі логічні частини, кожна з яких має чітко визначене функціональне призначення. Це спрощує не лише етап проєктування, а й подальше налагодження, модернізацію та можливу заміну окремих вузлів без потреби повного перегляду всієї структури. У підсумку сформовано архітектуру, що об'єднує блок металошукача, вузол узгодження та фільтрації сигналу, мікроконтролерний блок керування, модулі відображення інформації та керування, бездротовий інтерфейс зв'язку, блоки додаткових сенсорів, а також підсистему живлення та перетворення напруги.

В основі всієї системи знаходиться аналогова частина металошукача, оскільки саме вона формує первинний корисний сигнал, на якому ґрунтується подальша робота всього пристрою. У цій частині здійснюється генерація імпульсів, взаємодія з пошуковою котушкою та формування електричного відгуку, що змінюється залежно від наявності або відсутності металевих об'єктів в зоні контролю. По суті, аналогова частина є тим вузлом, який безпосередньо

					КВРКІ.2301109.23.01.43 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		23

контактує з фізичним середовищем і перетворює факт наявності металу на зміну електричних параметрів сигналу. Саме тому до її побудови висуваються вимоги щодо достатньої чутливості, стабільності та узгодженості з іншими елементами системи.

Проте вихідний сигнал металошукача не може безпосередньо передаватися на мікроконтролер, оскільки його форма, амплітуда та характер зміни не відповідають умовам безпечного цифрового зчитування. У реальній схемі на виході аналогового тракту можуть виникати імпульси з амплітудою, яка перевищує допустимий рівень для входів мікроконтролера. Крім того, корисний сигнал супроводжується шумами, короткочасними сплесками, залишковими перехідними процесами та іншими складовими, що можуть негативно впливати на коректність вимірювання. Саме з цієї причини наступним обов'язковим елементом архітектури виступає вузол узгодження та фільтрації.

Цей вузол виконує одразу кілька важливих функцій. По-перше, здійснюється обмеження амплітуди сигналу до безпечного рівня, придатного для входу аналого-цифрового перетворювача мікроконтролера. По-друге, виконується часткова фільтрація завад і згладжування імпульсного характеру вихідного сигналу. По-третє, забезпечується захист цифрової частини від небажаних перенапруг та паразитних процесів, які в іншому випадку могли б призвести або до спотворення даних, або до нестабільної роботи мікроконтролера. Отже, блок узгодження фактично є перехідною ланкою між аналоговим світом металошукача та цифровим світом програмної обробки, і без нього побудова надійної системи є практично неможливою.

Центральним керувальним елементом системи обрано мікроконтролерну платформу ESP32. У загальній архітектурі вона виконує роль ядра всієї системи, оскільки саме на цьому рівні зосереджено обробку зчитаних даних, керування периферійними модулями, реалізацію локального інтерфейсу, організацію бездротового зв'язку та підтримку логіки роботи в різних режимах. Використання ESP32 у даній архітектурі є доцільним через поєднання кількох важливих властивостей. По-перше, платформа має достатню продуктивність для

					КВРКІ.2301109.23.01.43 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		24

виконання періодичного зчитування сигналу та його програмної обробки. По-друге, вона вже містить вбудовані засоби Wi-Fi, що усуває потребу у зовнішньому модулі бездротового зв'язку. По-третє, мікроконтролер дозволяє одночасно працювати з кількома периферійними пристроями, такими як дисплей, кнопки керування, GPS-модуль, датчики середовища та цифровий потенціометр автокалібровки.

З погляду загальної архітектури ESP32 виступає не просто обчислювальним блоком, а координувальним центром усієї системи. Саме він виконує періодичне зчитування обробленого сигналу з виходу металошукача, аналізує зміну його рівня, визначає стан системи, відображає інформацію на локальному дисплеї, реагує на натискання кнопок та формує мережеві дані для віддаленого доступу. Через це можна стверджувати, що без мікроконтролерного блока дана система залишалася б лише аналоговим пристроєм із мінімальними можливостями, тоді як додавання ESP32 переводить її в клас повноцінних програмно-технічних засобів.

Для забезпечення дистанційного керування та адаптації пристрою до змінних умов, до архітектури введено блок автоматичної калібровки на базі цифрового потенціометра. Замість традиційного ручного налаштування змінними резисторами, зміна еквівалентного опору здійснюється програмним способом. Якщо під час запуску або роботи фіксується надмірний рівень шуму, мікроконтролер автоматично змінює положення потенціометра до досягнення стабільної точки спокою. Це перетворює пристрій на систему зі зворотним зв'язком, де цифрова частина самостійно коригує параметри роботи аналогової частини металошукача.

Для забезпечення взаємодії з користувачем у складі системи передбачено два рівні інтерфейсу: локальний і дистанційний. Локальний інтерфейс реалізовано за допомогою компактного OLED-дисплея та набору кнопок. Наявність локального інтерфейсу підвищує автономність та універсальність пристрою, оскільки він залишається працездатним у різних умовах і не вимагає постійного мережевого з'єднання. Другий рівень інтерфейсу - дистанційний -

					КВРКІ.2301109.23.01.43 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		25

реалізовано за рахунок використання бездротового зв'язку Wi-Fi. Це забезпечує віддалений доступ до даних металошукача та можливість керування пристроєм зі смартфона як у режимі автономної точки доступу, так і в режимі клієнтського підключення до наявної мережі.

Кнопковий блок керування в архітектурі системи виконує не лише допоміжну функцію, а фактично є елементом повноцінного меню. Через нього можуть вибиратися режими точки доступу або клієнтського підключення, режим роботи GPS, демонстраційний режим, параметри відображення та інші службові налаштування. Наявність кількох кнопок дозволяє створити просту, але достатньо гнучку систему навігації по меню, що є цілком виправданим для автономного приладу невеликого розміру.

Другий рівень інтерфейсу - дистанційний - реалізовано за рахунок використання бездротового зв'язку Wi-Fi. У загальній архітектурі це дає системі суттєво нові можливості. По-перше, забезпечується віддалений доступ до даних металошукача без прямого дротового з'єднання. По-друге, з'являється можливість керування пристроєм зі смартфона або іншого клієнтського пристрою. По-третє, у режимі підключення до зовнішньої мережі система може використовувати інтернет-з'єднання для розширення функціональності, наприклад для роботи з координатними даними. Така архітектурна побудова є дуже вдалою, оскільки не вимагає окремих передавачів або спеціалізованих приймачів. Усе зводиться до використання стандартного мобільного пристрою, який і так є у більшості користувачів.

У системі передбачено два базові режими бездротової роботи. У першому випадку ESP32 виступає як точка доступу, тобто сама формує локальну мережу, до якої підключається смартфон. Такий режим є зручним там, де відсутня зовнішня мережа або потрібна повністю автономна робота. У другому випадку мікроконтролер працює як клієнт та підключається до вже наявної мережі, наприклад до точки доступу, створеної телефоном. Подібний режим дає змогу розширити функціональність, оскільки у користувача зберігається доступ до інтернету, а це важливо для роботи з картами чи геопросторовими сервісами.

					КВРКІ.2301109.23.01.43 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		26

Архітектурно це означає, що система є адаптивною: вона не прив'язана до одного фіксованого сценарію використання, а може працювати по-різному залежно від поставленої задачі.

Суттєвим розширенням системи є інтеграція допоміжних сенсорів, що перетворює металошукач на комплексний аналітичний інструмент. Для збору супровідних метаданих використовуються: GPS-модуль – для просторової фіксації та картографування точок виявлення; ультразвуковий датчик відстані – для оцінки умов зчитування сигналу відносно поверхні; датчики температури, вологості повітря та ґрунту – для моніторингу зовнішніх умов експлуатації, що можуть впливати на стабільність роботи пристрою. Живлення на датчик ґрунту подається короткочасно для запобігання окисненню.

Окремого розгляду потребує підсистема збереження та експорту даних. У загальній архітектурі вона не є фізично окремим модулем, але логічно займає важливе місце. Після зчитування та обробки сигналу система повинна не лише відобразити його поточний стан, а й за потреби сформувати структуровані дані для подальшого використання. Це особливо важливо для точок із координатною прив'язкою. Експорт даних у форматах, придатних для відкриття в QGIS, Google Earth Pro, uMap або інших інструментах, переводить систему на новий рівень функціональності. У такому випадку користувач отримує не просто значення сигналу “тут і зараз”, а базу для накопичення результатів, побудови карт, аналізу маршруту та просторового порівняння виявлених об'єктів.

Важливе місце в загальній архітектурі займає підсистема живлення. Для даного пристрою вона є особливо критичною, оскільки різні вузли потребують різних рівнів напруги, а стабільність живлення безпосередньо впливає на роботу як аналогового, так і цифрового блоків. Аналогова частина металошукача, цифровий модуль, дисплей, GPS і додаткові сенсори не повинні створювати один одному небажані перешкоди через погано організований розподіл живлення. Саме тому в архітектурі передбачено модулі пониження та стабілізації напруги, а також окремі гілки живлення для різних функціональних вузлів. Такий підхід

					КВРКІ.2301109.23.01.43 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		27

дозволяє підвищити загальну надійність і зменшити вплив шумів від цифрової частини на чутливий вимірювальний тракт металошукача.

З конструктивної точки зору архітектура системи реалізується у вигляді багатошарового компонування. Це рішення є не лише механічно зручним, а й логічно виправданим. Рознесення вузлів по шарах дозволяє чітко відділити частини з різною функціональною природою. Наприклад, нижній рівень може містити живлення та аналогову частину металошукача, середній - цифрову логіку та обчислювальний блок, а верхній - елементи взаємодії з користувачем. Така просторово-структурна організація покращує зручність монтажу, спрощує доступ до окремих модулів та полегшує подальше обслуговування й модернізацію.

Якщо розглядати принцип побудови системи в цілому, то її архітектуру можна представити у вигляді взаємодії спеціалізованих блоків. Давачі інформації, допоміжні сенсори середовища та просторової прив'язки збирають первинні показники та передають їх в обробник даних. Після структуризації обробник даних надсилає цей масив інформації в центральний керуючий блок. Керуючий блок здійснює безперервний двосторонній обмін даними з пошуковим модулем, аналоговим трактом металошукача, коригуючи його роботу та отримуючи сигнал виявлення. Одночасно керуючий блок обмінюється даними з блоком взаємодії з користувачем для виведення результатів та отримання команд від оператора, що наочно продемонстровано на рисунку 2.1.

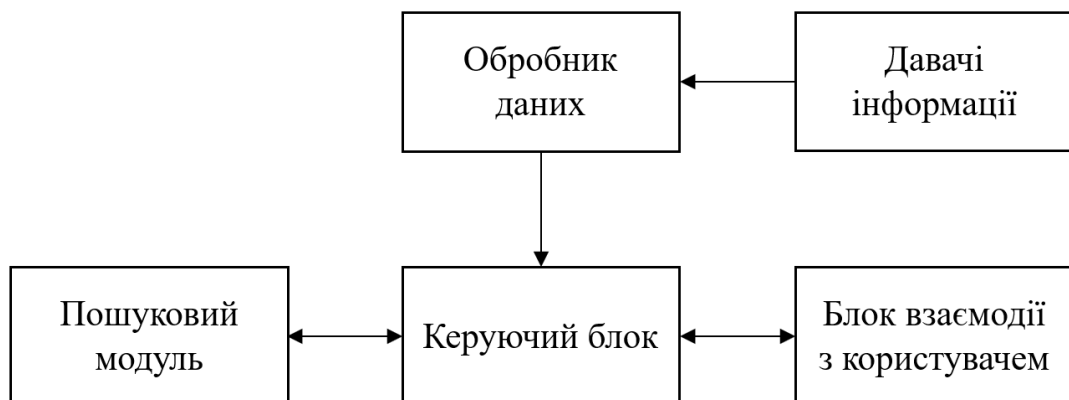


Рисунок 2.1 – Загальна архітектура системи

У підсумку загальна архітектура розроблюваного програмно-технічного засобу є достатньо гнучкою, функціонально насиченою та придатною до подальшого розвитку. Вона вже враховує потребу у безпечному зчитуванні аналогового сигналу, його програмній обробці, автоматичній калібровці, локальній та дистанційній взаємодії з користувачем, збиранні супровідних сенсорних даних, фіксації координат і подальшому експорті результатів. Саме завдяки такій архітектурній побудові створюваний пристрій може розглядатися як цілісний програмно-технічний засіб дистанційного керування металошукачем, а не як простий набір окремих електронних модулів.

2.2 Обґрунтування вибору апаратної платформи та компонентів системи

Побудова програмно-технічного засобу дистанційного керування металошукачем безпосередньо залежить від того, наскільки вдало підібрано його апаратну основу. У даній кваліфікаційній роботі вибір елементної бази розглядається не як другорядний етап, а як одна з ключових складових загального проектування, оскільки саме від характеристик окремих модулів залежить стабільність зчитування сигналу, коректність його подальшої обробки, зручність керування системою, автономність роботи та можливість реалізації дистанційного доступу. Через це обґрунтування вибору кожного компонента виконано з урахуванням функціонального призначення модуля, його технічних можливостей, сумісності з іншими елементами системи, простоти інтеграції та практичної доцільності використання у складі єдиного пристрою.

Центральною апаратною платформою обрано мікроконтролер ESP32. Його використання обумовлене високим рівнем інтеграції: наявність вбудованого модуля Wi-Fi з підтримкою режимів точки доступу AP та клієнта STA усуває потребу у зовнішніх мережевих компонентах і розширює гнучкість системи. Крім того, обчислювальна потужність платформи є достатньою для одночасного виконання вимогливих задач: частого зчитування й програмної фільтрації

					КВРКІ.2301109.23.01.43 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		29

аналогових сигналів, керування периферією (дисплеєм, сенсорами, цифровим потенціометром) та обслуговування веб-інтерфейсу без затримок.

З огляду на ключову роль мікроконтролера ESP32 у розроблюваній системі, доцільно детальніше розглянути його технічні характеристики, архітектуру периферійних інтерфейсів та специфіку використання контактів.

Для взаємодії з електронними компонентами та сенсорами платформа володіє широким набором вбудованої периферії, що значно перевищує можливості класичних платформ попередніх поколінь. Важливою особливістю мікроконтролера є високий рівень апаратної гнучкості: розробник має змогу програмно перепризначати більшість інтерфейсів (UART, I2C, SPI, ШІМ) на довільні контакти введення-виведення. Загалом апаратна база ESP32 включає:

- 18 каналів аналого-цифрового перетворювача (АЦП);
- 2 канали 8-бітного цифро-аналогового перетворювача (ЦАП);
- 16 незалежних каналів широтно-імпульсної модуляції (ШІМ);
- 3 інтерфейси UART, 2 інтерфейси I2C та 3 інтерфейси SPI;
- 10 ємнісних сенсорних входів.

Схему призначення контактів платформи ESP32 наведено на рисунку 2.2.

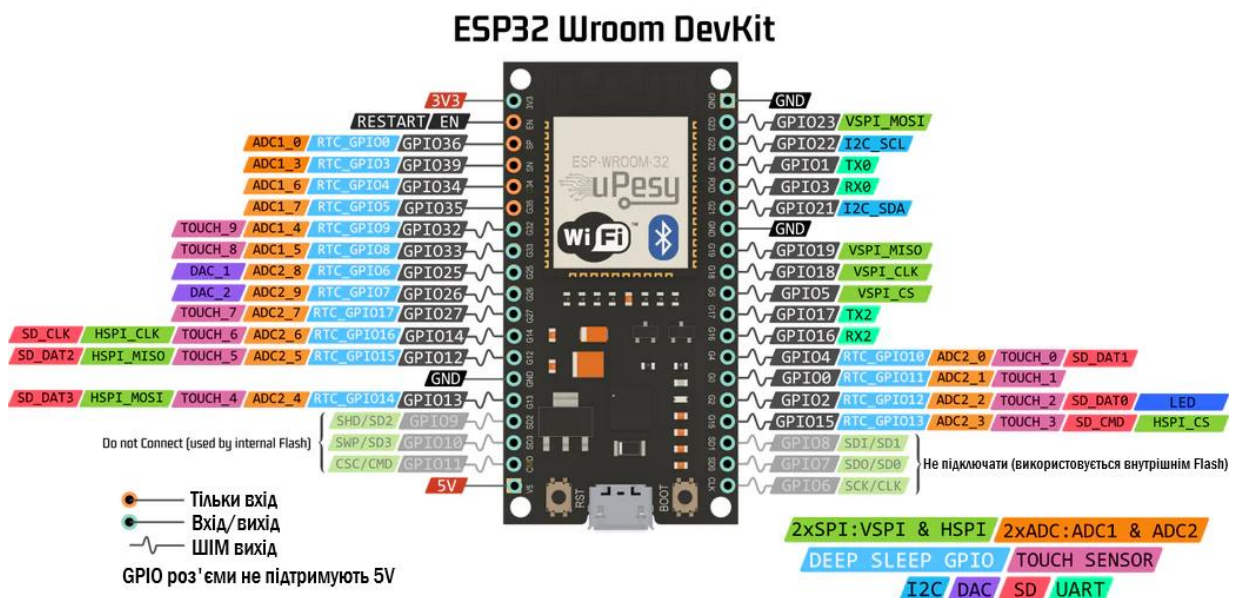


Рисунок 2.2 – Схема призначення контактів платформи ESP32 [59]

Незважаючи на наявність 33 робочих контактів GPIO, частина з них має апаратні обмеження або специфічну поведінку, що було враховано під час розроблення принципової електричної схеми пристрою. Зокрема, виводи GPIO 6, 7, 8, 9, 10 та 11 апаратно підключені до інтегрованої флеш-пам'яті мікроконтролера, тому їх використання для підключення зовнішніх пристроїв або датчиків металошукача є неприпустимим, оскільки це може призвести до порушення роботи всієї системи. Виводи GPIO 34, 35, 36 (VP) та 39 (VN), навпаки, придатні для зчитування аналогових сигналів із вимірювального тракту, оскільки працюють лише як входи, хоча й не мають внутрішніх підтягувальних резисторів. Особливої обережності потребують так звані керуючі контакти GPIO 0, 2, 12 та 15, від логічного стану яких під час подачі живлення залежить режим запуску ESP32, тобто звичайне виконання програми або перехід у режим прошивки. Через це підключення до них зовнішньої периферії має виконуватися з урахуванням ризику блокування старту мікроконтролера. Крім того, на виводах GPIO 1, 3, 5, 14 та 15 під час увімкнення можуть короткочасно з'являтися ШІМ-сигнали або певні логічні рівні, тому їх недоцільно використовувати для підключення критичних виконавчих елементів, щоб уникнути хибних спрацювань.

Особливої уваги потребує й архітектура аналого-цифрового перетворювача. Мікроконтролер ESP32 оснащено двома окремими модулями АЦП - ADC1 із вісьмома каналами та ADC2 із десятима каналами. Водночас блок ADC2 архітектурно використовується внутрішнім драйвером Wi-Fi, а тому при одночасній роботі бездротової мережі можливі апаратні конфлікти під час зчитування аналогових сигналів. З огляду на те, що розроблюваний металошукач активно використовує Wi-Fi для дистанційного керування, зчитування аналогових даних із котушки доцільно виконувати виключно через канали ADC1, що дозволяє забезпечити стабільність вимірювання та уникнути небажаних збоїв у роботі системи.

Передача даних між мікроконтролером та периферійними пристроями реалізована за допомогою стандартних інтерфейсів. Наприклад, для зв'язку з

					КВРКІ.2301109.23.01.43 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		31

комп'ютером і програмування зарезервовано інтерфейс UART0 (виводи 1 і 3), тоді як для підключення зовнішніх цифрових модулів може застосовуватися UART2 (виводи 16 і 17). Для обміну даними з локальними цифровими вузлами штатно використовуються дві шини I2C.

Для локального відображення інформації в системі обрано компактний OLED-дисплей на базі контролера SSD1306. Використання саме такого дисплея обумовлене відразу кількома факторами. По-перше, модулі цього типу мають невеликі габаритні розміри, що дозволяє інтегрувати їх у конструкцію без суттєвого збільшення корпусу. По-друге, OLED-дисплеї характеризуються низьким енергоспоживанням, що є важливим для автономних систем, де будь-яке перевантаження по живленню небажане. По-третє, SSD1306 легко підключається до мікроконтролера за допомогою інтерфейсу I2C, а це зменшує кількість задіяних контактів і спрощує загальну схему підключення.

Функціонально такий дисплей добре підходить для виведення базових службових даних. На ньому можуть відображатися поточний режим роботи системи, показники сигналу металошукача, стан бездротового з'єднання, службова інформація з меню, статус роботи GPS-модуля, а також інші допоміжні параметри. У контексті цієї роботи SSD1306 виконує важливу роль локального інтерфейсу, який не залежить від підключення до смартфона чи веб-сторінки. Це особливо важливо в тих випадках, коли пристрій використовується автономно або коли швидкий перегляд стану потрібен без запуску додаткових інструментів. Отже, вибір OLED-дисплея є цілком логічним як з конструктивного, так і з функціонального погляду.

Для контролю параметрів середовища та просторової прив'язки в системі використано набір стандартних периферійних модулів. Датчик DHT11 забезпечує вимірювання температури та вологості повітря. Ультразвуковий датчик HC-SR04 застосовується для вимірювання відстані, при цьому в схемі враховано необхідність узгодження рівнів його сигналу для входів мікроконтролера ESP32. Для фіксації координат обрано GPS-модуль GY-NEO6M, що працює через послідовний інтерфейс і дозволяє отримувати дані без

					КВРКІ.2301109.23.01.43 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		32

складного налагодження апаратної частини. Також у системі використано датчик вологості ґрунту на базі компаратора LM393 , живлення на який подається лише в момент зчитування показників для суттєвого зменшення деградації сенсора.

Окремо слід відзначити простоту інтеграції DHT11. Датчик не вимагає складного обв'язування, підтримується великою кількістю бібліотек і не створює надмірного навантаження на мікроконтролер. Це дозволяє включити його в систему без істотного ускладнення програмного забезпечення. З урахуванням того, що цей модуль виконує допоміжну функцію, саме така комбінація простоти, доступності та достатності характеристик виявляється цілком доречною.

Для вимірювання відстані до поверхні використано ультразвуковий датчик HC-SR04, дані з якого виступають додатковим аналітичним параметром для точнішої інтерпретації сигналу металошукача. Оскільки логічний рівень виходу датчика може становити 5 В, що перевищує допустимі межі для входів ESP32, у схемі реалізовано апаратне узгодження рівнів сигналу за допомогою дільника напруги.

Для фіксації координат обрано GPS-модуль GY-NEO6M, який працює через стандартний послідовний інтерфейс і не потребує складного апаратного налагодження. Його використання забезпечує просторову прив'язку результатів виявлення в польових умовах, дозволяючи формувати структуровані дані для подальшого експорту в геоінформаційні та картографічні сервіси. Точності та функціональності цього модуля цілком достатньо для поставленої задачі.

Додатково в системі реалізовано програмний режим використання умовних тестових координат, що дозволяє повноцінно демонструвати роботу пристрою навіть за відсутності супутникового сигналу, наприклад у приміщенні.

У системі використано також датчик вологості ґрунту на базі компаратора LM393. Подібні модулі є доволі простими за конструкцією і дозволяють отримувати як аналогову оцінку рівня вологості, так і цифровий сигнал при досягненні певного порогового значення. У контексті даної роботи цей сенсор виступає як допоміжне джерело даних про стан середовища. Його введення до

					КВРКІ.2301109.23.01.43 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		33

складу системи дозволяє збирати додаткові параметри, які можуть мати значення при аналізі умов роботи металошукача. Хоча сам по собі датчик вологості ґрунту не впливає на базову функцію виявлення металу, його використання розширює можливості пристрою як комплексного програмно-технічного засобу.

Разом із тим під час вибору такого модуля враховано його характерну особливість - схильність до поступового окиснення чутливого елемента при тривалому перебуванні під напругою. Саме тому в системі доцільно реалізовувати короткочасну подачу живлення лише в момент зчитування показників. Такий підхід дозволяє суттєво зменшити деградацію сенсора та зробити його використання більш виправданим у тривалій перспективі. Отже, навіть цей допоміжний компонент вибрано з урахуванням не лише функціональності, а й практичних особливостей експлуатації.

Однією з найважливіших спеціалізованих елементів системи виступає цифровий потенціометр TPL0501. Його вибір зумовлений необхідністю реалізації автоматичної калібровки металошукача. TPL0501 дозволяє змінювати еквівалентний опір програмним способом, тобто мікроконтролер може самостійно підбирати значення, орієнтуючись на поточний стан сигналу металошукача.

TPL0501 дозволяє змінювати еквівалентний опір програмним способом, тобто мікроконтролер може самостійно підбирати значення, орієнтуючись на поточний стан сигналу металошукача. Це відкриває можливість реалізації логіки, за якої система автоматично змінює параметри до тих пір, поки не буде досягнуто стану, близького до стабільної точки спокою. Такий принцип калібровки є дуже важливим для даної роботи, оскільки особливості котушки, відхилення параметрів елементів, зміни зовнішнього середовища та навіть відмінності між екземплярами схеми можуть призводити до зміщення робочої точки. Використання TPL0501 дозволяє компенсувати ці фактори без ручного обертання регуляторів і без порушення ідеї дистанційного керування системою.

Окремої уваги заслуговує вибір модуля живлення XL4015. Для багатокомпонентної системи з аналоговою частиною металошукача, цифровим

					КВРКІ.2301109.23.01.43 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		34

мікроконтролером, дисплеєм, GPS і сенсорами стабільне живлення є однією з базових умов коректної роботи. XL4015 використовується як понижуючий DC-DC перетворювач і дозволяє отримати стабільну напругу для живлення вузлів із різними вимогами. Його застосування є виправданим через достатній запас по струму, простоту налаштування та широку доступність. Особливо важливим це є в системі, де одночасно присутні цифрові вузли, чутливі аналогові елементи та бездротовий модуль, для якого небажані просідання живлення або нестабільність напруги.

Використання XL4015 дозволяє сформувати більш контрольовану та передбачувану систему живлення. Це важливо не лише для безперебійної роботи пристрою, а й для зменшення впливу шумів на вимірювальний тракт металошукача. Якщо живлення організоване нестабільно, то це може проявлятися у зміні рівня шуму, помилкових спрацюваннях або нестабільному запуску периферійних модулів. Через це вибір якісного понижуючого модуля є не просто бажаним, а необхідним.

Крім понижуючого перетворювача, у системі використано модуль для зарядки акумуляторів. Його наявність обумовлена тим, що пристрій орієнтований на автономне використання, а отже живлення має бути не лише стабільним, а й зручним з точки зору експлуатації. Застосування акумуляторного живлення робить систему мобільною, а це є дуже важливою властивістю для металошукача як польового приладу. Модуль зарядки дозволяє організувати обслуговування енергосистеми без складних зовнішніх рішень і зробити сам пристрій більш самодостатнім.

Окремо слід відзначити, що вибір допоміжних компонентів виконувався не ізольовано, а з урахуванням їхньої спільної роботи в межах єдиної архітектури. Наприклад, дисплей має бути сумісним із логічними рівнями та інтерфейсами ESP32, датчики не повинні створювати надмірного навантаження на обчислювальну частину, а вузли живлення мають забезпечувати коректну роботу всіх підсистем одночасно. Це означає, що обґрунтування вибору не зводиться лише до опису окремих модулів. Насправді йдеться про формування

					КВРКІ.2301109.23.01.43 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		35

узгодженого комплексу компонентів, який у сукупності забезпечує реалізацію поставленої задачі.

Загалом обраний набір апаратних компонентів демонструє прагнення до балансу між функціональністю, конструктивною простотою та можливістю подальшого розвитку. ESP32 забезпечує центральне керування і мережеву взаємодію, OLED-дисплей і кнопки - локальний інтерфейс, TPL0501 - автокалібровку, GPS - просторову фіксацію подій, сенсори середовища - допоміжні метадані, а система живлення - стабільну та автономну роботу. Усе це в сукупності дозволяє сформувати апаратну основу, яка не лише підтримує базову функцію виявлення металу, а й значно розширює загальну функціональність створюваного програмно-технічного засобу.

У підсумку вибір апаратної платформи та компонентів системи виконано на основі технічної доцільності, функціональної достатності та логіки побудови єдиної інтегрованої системи. Кожен модуль у складі пристрою має власну чітко визначену роль, а їх поєднання дозволяє реалізувати дистанційно керований металодетектор із розширеними можливостями збирання, обробки, відображення та передавання даних. Саме це створює необхідну основу для подальшого розроблення структурної схеми, принципової електричної реалізації та програмної логіки системи.

2.3 Розроблення структурної та електричної принципової схеми системи

Наступним етапом проєктування є розроблення структурної та електричної принципової схем системи. Структурна схема конкретизує склад функціональних блоків пристрою, послідовність проходження сигналів та їхню взаємодію, створюючи базу для подальшої детальної схемотехнічної реалізації.

У розроблюваній системі структурна схема формується навколо кількох основних функціональних центрів. Першим із них є блок металодетектора, який виконує задачу первинного формування корисного сигналу. Саме в цьому блоці здійснюється генерування імпульсів, робота з пошуковою котушкою та

					КВРКІ.2301109.23.01.43 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		36

утворення відгуку, що змінюється при наближенні металевого об'єкта. Цей вузол є джерелом первинної інформації, тому його положення в структурі є базовим. Усі інші блоки вже або обробляють сформований сигнал, або забезпечують керування, відображення та передавання отриманих результатів.

Наступною важливою частиною структурної схеми є блок узгодження та фільтрації сигналу. Необхідність його введення обумовлена тим, що вихід металошукача не може бути безпосередньо поданий на вхід мікроконтролера. Аналоговий сигнал має імпульсний характер, містить завади та може досягати рівнів напруги, небезпечних для цифрової платформи. Саме тому між виходом металошукача і входом ESP32 введено окремий проміжний вузол, який виконує функцію пониження амплітуди, захисту, згладжування сигналу та часткового очищення від шумів. У структурній схемі цей блок займає ключове місце, оскільки без нього перехід від аналогового тракту до цифрового рівня був би технічно некоректним.

Центральне місце в структурній схемі займає мікроконтролер ESP32. Він виступає вузлом, до якого сходяться сигнали від основного вимірювального тракту та допоміжних датчиків, а також через який здійснюється керування іншими елементами системи. У структурному представленні ESP32 виконує відразу кілька функцій. По-перше, він зчитує сигнал з виходу блоку фільтрації. По-друге, керує цифровим потенціометром, який використовується для автоматичної калібровки металошукача. По-третє, опитує GPS-модуль, датчики температури, вологості та відстані. По-четверте, передає інформацію на OLED-дисплей і обслуговує кнопочий інтерфейс. По-п'яте, організовує роботу мережевої частини системи в режимах точки доступу або клієнта. Саме така концентрація функцій у межах одного вузла дозволяє спростити побудову системи та забезпечити її узгоджену роботу.

У структурній схемі також обов'язково виділяється блок автоматичної калібровки., оскільки з функціонального погляду роль цифрового потенціометра є настільки важливою, що він повинен бути окремо показаний у структурі. Цей вузол забезпечує програмне регулювання параметрів аналогової частини

					КВРКІ.2301109.23.01.43 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		37

металошукача. Через нього мікроконтролер впливає на робочу точку системи, підлаштовуючи її до умов експлуатації, типу котушки та фактичного рівня шуму. У результаті структурна схема відображає не лише прямий потік сигналу від металошукача до мікроконтролера, а й наявність зворотного зв'язку, за рахунок якого цифрова частина коригує параметри аналогової.

Окремий сегмент структурної схеми формують допоміжні сенсорні модулі. До них належать GPS-приймач, ультразвуковий датчик HC-SR04, датчик температури та вологості повітря DHT11, а також модуль вимірювання вологості ґрунту. Усі вони підключаються до ESP32 і забезпечують надходження додаткових даних, що супроводжують основний сигнал металошукача. У структурному представленні це дозволяє показати, що система є не тільки засобом пошуку металу, а й комплексним інструментом збору супровідної інформації. GPS-модуль додає координатну прив'язку, DHT11 - характеристики середовища, HC-SR04 - відстань до поверхні або об'єкта, а датчик вологості ґрунту - додаткові параметри зовнішніх умов.

Не менш важливою частиною структурної схеми є підсистема взаємодії з користувачем. До неї належать OLED-дисплей та кнопки керування. Ці елементи утворюють локальний інтерфейс системи. У структурному сенсі вони підключаються до ESP32 як пристрої введення та виведення, але з погляду функціонального наповнення вони є засобами локального контролю над пристроєм. Саме через них користувач отримує доступ до меню, перегляду базових параметрів, перемикання режимів та виконання службових налаштувань. Наявність окремо виділеного локального інтерфейсу у структурній схемі показує, що система не залежить виключно від зовнішнього мобільного пристрою та може працювати автономно.

Поряд із локальним інтерфейсом у структурній схемі необхідно показати підсистему бездротового зв'язку. У випадку ESP32 вона не представлена окремим зовнішнім модулем, однак функціонально є настільки важливою, що її доцільно виділити як окремий логічний блок у структурі. Через цей блок відбувається дистанційне передавання даних, підключення смартфона або

					КВРКІ.2301109.23.01.43 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		38

іншого клієнтського пристрою, робота веб-інтерфейсу та організація режимів AP або STA. Це дозволяє на структурному рівні зафіксувати, що система має два незалежні канали взаємодії з користувачем: локальний через кнопки та дисплей і дистанційний через Wi-Fi.

Останнім, але не менш важливим елементом структурної схеми є блок живлення. Його не можна розглядати як допоміжний чи другорядний, оскільки стабільність роботи всієї системи безпосередньо залежить від коректного формування напруги для різних вузлів. У структурній схемі цей блок повинен охоплювати джерело живлення, вузол зарядки акумуляторів та понижуючий перетворювач, який забезпечує формування потрібних рівнів напруги. Саме від нього живляться металошукач, мікроконтролер, дисплей і сенсори. Структурне виділення цього вузла дозволяє підкреслити, що в системі передбачено автономну роботу та розподіл живлення між елементами з різними електричними вимогами. Розроблену структурну схему, яка об'єднує всі описані функціональні блоки, наведено на рисунку 2.3.

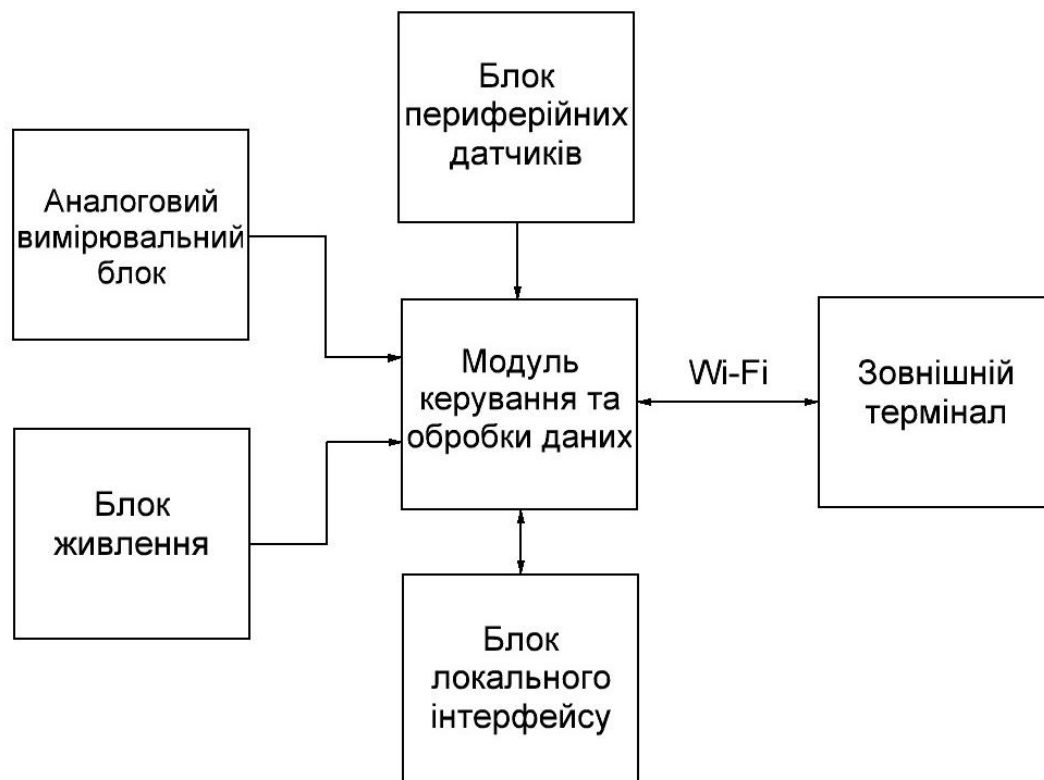


Рисунок 2.3 – Структурна схема системи

У результаті сформована структурна схема дає цілісне уявлення про побудову всієї системи. У ній чітко простежується головний шлях проходження сигналу: пошукова котушка і металошукач - блок узгодження та фільтрації - ESP32 - блоки відображення, збереження та передавання даних. Паралельно із цим у структуру інтегровано сенсорні модулі, локальний інтерфейс, блок автокалібровки та живлення. Такий підхід дозволяє вважати структурну схему не лише графічним супроводом, а й логічним відображенням всієї архітектури системи.

Після розроблення структурної схеми наступним етапом стало формування електричної принципової схеми. Якщо структурне представлення відображає логіку побудови пристрою, то принципова схема вже деталізує конкретні електричні зв'язки між елементами, номінали основних компонентів, спосіб підключення вузлів і характер їхньої взаємодії на рівні сигналів та живлення. Саме принципова схема дозволяє перейти від загальної функціональної моделі до конкретної електротехнічної реалізації системи.

Основу принципової схеми становить аналогова частина металошукача. У роботі використано схему імпульсного типу, яка модифікується для інтеграції з цифровим керуванням. У цій частині реалізовано генерацію імпульсів, роботу з котушкою, формування сигналу та його підсилення. Саме тому в принциповій схемі один із важливих вузлів утворює цифровий потенціометр TPL0501.

У схемі цифровий потенціометр підключається таким чином, щоб забезпечити зміну робочої точки металошукача в діапазоні, достатньому для пошуку стану спокою. Послідовно з ним включено додатковий резистор, який виконує функцію обмеження та підбору правильного зміщення. Необхідність цього елемента обумовлена тим, що пряме використання цифрового потенціометра без додаткового резистивного ланцюга може призводити до некоректного розподілу напруги та зміщення порогу калібрування. Підбір такого резистора залежить від конкретної котушки, параметрів елементів металошукача та потрібного робочого діапазону. Отже, у принциповій схемі цей вузол є не просто формальним включенням модуля, а спеціально адаптованим рішенням

					КВРКІ.2301109.23.01.43 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		40

для забезпечення автокалібровки.

Наступним критично важливим вузлом принципової схеми виступає блок фільтрації та узгодження сигналу. Як уже зазначалося, на виході операційного підсилювача металошукача формується сигнал, який може досягати значень до 12 В. Такий рівень неприйнятний для входу ESP32, що працює з напругою 3,3 В. Саме тому у принциповій схемі між металошукачем і мікроконтролером введено ділянку напруги, який приводить амплітуду сигналу до безпечного рівня. Цей ділянку є одним із базових захисних елементів схеми, адже він унеможливорює перевищення допустимого рівня на вході аналого-цифрового перетворювача.

Крім ділянки, у схемі використано діод Шотткі. Його введення обумовлене потребою зменшення падіння напруги та одночасного захисту від небажаного зворотного струму. Порівняно зі звичайним діодом, діод Шотткі має менше пряме падіння напруги, що в даному вузлі є перевагою. Він дозволяє коректніше формувати захисну функцію без зайвого спотворення корисного сигналу. У контексті принципової схеми цей елемент виконує роль додаткового бар'єра між чутливою аналоговою частиною та мікроконтролером.

Фільтрація сигналу в принциповій схемі реалізується за допомогою двох конденсаторів різного типу. Один із них є електrolітичним і має більшу ємність, інший - керамічним із меншою ємністю. Таке поєднання використовується не випадково. Електrolітичний конденсатор ефективніше згладжує повільніші складові та накопичує заряд, тоді як керамічний краще працює з високочастотними перешкодами та короткими імпульсними шумами. У результаті формується комбінований фільтр, який забезпечує прийнятне згладжування сигналу без надмірного спотворення його корисної частини. Після проходження цього вузла на вході мікроконтролера формується вже не різкий імпульсний сигнал великої амплітуди, а більш згладжена пилкоподібна форма, придатна для подальшого програмного аналізу.

У принциповій схемі окремо показано підключення ESP32 до периферійних модулів. OLED-дисплей SSD1306 підключається за інтерфейсом I2C, що дозволяє зменшити кількість ліній зв'язку та спростити монтаж. Такий

					КВРКІ.2301109.23.01.43 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		41

спосіб підключення є цілком доцільним для системи, де частина контактів уже використовується для сенсорів, кнопок і вимірювальних входів. Через той самий або окремо виділений цифровий інтерфейс можуть підключатися й інші модулі, що робить побудову принципової схеми більш впорядкованою.

GPS-модуль у принциповій схемі підключається через послідовний інтерфейс, оскільки саме такий спосіб є для нього стандартним і найбільш зручним. Це дозволяє організувати безперервне надходження координатних даних до мікроконтролера без складної логіки синхронізації. Датчик DHT11, своєю чергою, використовує окрему цифрову лінію, а ультразвуковий датчик HC-SR04 потребує двох ліній - для запуску імпульсу та приймання сигналу ЕЧНО. При цьому в схемі має бути враховано узгодження рівнів, щоб виключити подавання завищеної напруги на вхід ESP32.

Кнопковий блок у принциповій схемі реалізується через кілька цифрових входів мікроконтролера. Кожна кнопка відповідає за окрему функцію, наприклад перехід у меню, підтвердження вибору або навігацію вгору та вниз. У такій реалізації принципова схема повинна враховувати підтягувальні резистори або використання внутрішніх підтягувань мікроконтролера. Це потрібно для формування стабільного логічного рівня при ненависнутому стані кнопок і виключення помилкових спрацювань.

У системі живлення принципова схема відображає кілька важливих гілок. Одна з них пов'язана з джерелом акумуляторного живлення, друга - із зарядним модулем, третя - з понижуючим перетворювачем XL4015. Далі напруга розподіляється між вузлами системи відповідно до їхніх потреб. Мікроконтролер і частина цифрових модулів працюють від нижчої стабілізованої напруги, тоді як металошукач може потребувати окремого рівня живлення. Такий розподіл у принциповій схемі повинен бути показаний максимально чітко, оскільки саме від нього залежить надійність роботи всієї системи. Неправильне живлення навіть одного вузла може призвести або до нестабільного функціонування, або до перекручення сигналів, або до надмірного нагріву компонентів.

З конструктивного погляду принципова схема повинна також враховувати

					КВРКІ.2301109.23.01.43 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		42

багатошарове розміщення вузлів. Хоча безпосередньо шари належать уже до компонувального рішення, вони логічно впливають із самої побудови електричних з'єднань. Аналогова частина з металошукачем і живленням доцільно розміщується окремо від цифрової логіки, щоб зменшити взаємний вплив завад. Модулі інтерфейсу користувача, навпаки, доцільно виносити на верхній рівень для зручності доступу. Саме тому розроблення принципової схеми в даній роботі не можна розглядати ізольовано від подальшої конструктивної реалізації. У додатку В зображене повне креслення схеми електричної принципової.

У підсумку розроблення структурної та електричної принципової схеми системи дозволило перейти від загального уявлення про архітектуру пристрою до конкретної технічної моделі, у якій уже чітко визначено склад основних вузлів, характер їхньої взаємодії та спосіб електричної реалізації. Структурна схема відобразила логіку побудови системи, а електрична принципова схема конкретизувала шляхи проходження сигналів, методи захисту, фільтрації, живлення та підключення периферії. Це створює необхідну основу для подальшого розгляду принципів обробки сигналу металошукача, автоматичної калібровки та організації програмної логіки системи. Для більш детального розуміння, у додатку Б зображено схему, де структурна схема розділена на окремі функціональні блоки. Завдяки цій схемі є можливість побудувати чітке розуміння за що відповідають ті чи інші блоки та модулі, які присутні у схемі. Згідно цієї схеми можна чітко зрозуміти вектори взаємодії усіх функціональних блоків, які належать системі.

2.4 Принципи обробки сигналу та автоматичної калібровки металошукача

Для усунення залишкових коливань та внутрішніх шумів аналогового тракту, оцифрований сигнал проходить програмну фільтрацію. Процес базується на періодичному накопиченні вибірки даних із частотою дискретизації, що оптимізує навантаження на ESP32 та подальшому її згладжуванні за допомогою

					КВРКІ.2301109.23.01.43 ПЗ	Арк.
Зм.	Арк.	№ док.ум.	Підпис	Дата		43

алгоритму рухомого середнього. Такий підхід ефективно відфільтровує високочастотні стрибки та поодинокі шумові відхилення, забезпечуючи стабільну оцінку поточного рівня сигналу без втрати швидкості реакції системи.

Наступним рівнем обробки є оцінка розкиду значень у вибірці. Для цього визначається різниця між максимальним і мінімальним значенням або обчислюється відхилення від середнього. Така характеристика дозволяє оцінити рівень шуму в системі. Якщо розкид значень є значним, це означає, що сигнал нестабільний і потребує додаткової обробки або корекції параметрів. У випадку, коли розкид незначний, система може вважати сигнал стабільним і придатним для подальшої інтерпретації.

Після визначення стабільності сигналу виконується порівняння з базовим рівнем. Базовий рівень формується як еталонне значення, яке відповідає стану спокою. У робочому режимі система постійно порівнює поточний сигнал із цим значенням. Якщо різниця перевищує встановлений поріг, фіксується відхилення, яке може бути пов'язане з появою металевого об'єкта. Такий підхід дозволяє перейти від аналізу абсолютних значень до аналізу змін, що є більш ефективним у нестабільних умовах.

Особливістю реалізації є адаптивний характер порогового значення. Замість жорстко заданого порогу використовується підхід, при якому він може змінюватися залежно від рівня шуму.

Наприклад, якщо система фіксує збільшення флуктуацій, поріг автоматично підвищується, щоб уникнути хибних спрацювань. У більш стабільних умовах, навпаки, поріг може бути зменшений, що підвищує чутливість до слабких сигналів. Така адаптація дозволяє системі працювати більш універсально та не потребує постійного ручного втручання.

Цифрова обробка сигналу реалізується у вигляді циклічного алгоритму, який постійно повторюється протягом роботи системи. У кожному циклі виконується зчитування значення, оновлення вибірки, обчислення середнього, оцінка відхилення та прийняття рішення.

Завдяки цьому система реагує на зміну сигналу практично в реальному часі, але при цьому зберігає стійкість до випадкових коливань.

Окремою складовою цього процесу є автоматична калібровка. Її основною метою є визначення такого режиму роботи, при якому сигнал у стані спокою є максимально стабільним і не містить значних коливань. У даній системі калібровка реалізована програмно та виконується автоматично під час запуску або за потреби.

Алгоритм калібровки організовано як послідовний пошук оптимального значення параметра. Мікроконтролер змінює значення керуючого параметра в певному діапазоні та після кожного кроку виконує аналіз сигналу. Для кожного варіанта налаштування оцінюється рівень шуму, тобто ступінь коливання сигналу у вибірці. Результати порівнюються між собою, і система визначає варіант, який забезпечує найменший рівень нестабільності.

У процесі пошуку використовується критерій мінімізації відхилення. Це означає, що оптимальним вважається той стан, при якому сигнал змінюється найменше. Такий підхід є ефективним, оскільки дозволяє знайти точку спокою без необхідності точно знати її значення наперед.

Важливо, що алгоритм не зупиняється на першому прийнятному варіанті, а аналізує певний діапазон значень, що підвищує точність налаштування.

Після завершення калібровки визначене значення параметра фіксується, і система переходить у робочий режим. Однак навіть у цьому режимі можливе періодичне повторення калібровки або її часткове коригування. Це особливо актуально у випадках, коли змінюються зовнішні умови або параметри системи. Такий підхід дозволяє підтримувати стабільність роботи без втрати чутливості.

Важливою особливістю такого підходу є компроміс між швидкістю реакції системи та стійкістю до випадкових збурень. Якщо кількість відліків у вибірці є надто малою, система швидше реагує на зміну сигналу, проте водночас зростає вплив одиничних шумових коливань. Якщо ж вибірка надто велика, результат стає більш згладженим, але зменшується оперативність виявлення корисного відхилення. Саме тому цифрова обробка в розробленій системі побудована так,

					КВРКІ.2301109.23.01.43 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		45

щоб зберігати баланс між цими двома вимогами: забезпечувати достатню чутливість до зміни стану вимірювального тракту і водночас не допускати того, щоб короточасні нестабільності інтерпретувалися як реальний сигнал. Унаслідок цього цикл зчитування, усереднення та оцінювання відхилення виступає не просто послідовністю програмних операцій, а основою формування надійного цифрового представлення сигналу металошукача.

У підсумку реалізовані принципи цифрової обробки сигналу та автоматичної калібровки забезпечують перехід від нестабільного аналогового сигналу до стійкого цифрового результату. Завдяки використанню накопичення вибірок, усереднення, аналізу відхилень і адаптивного порогового визначення досягається висока стійкість до завад.

Одночасно алгоритм автоматичної калібровки дозволяє підтримувати оптимальний режим роботи без ручного втручання, що суттєво підвищує зручність використання системи та її ефективність у реальних умовах експлуатації.

2.5 Організація взаємодії з користувачем та обробки даних

Взаємодія з користувачем у системі реалізована на двох рівнях: локальному та дистанційному. Локальне керування забезпечується OLED-дисплеєм, на який виводяться ключові параметри, такі як: поточний рівень сигналу, активний режим, стан мережі, дані сенсорів, та кнопковим блоком для навігації по меню. Це дозволяє повноцінно та автономно працювати з пристроєм у польових умовах без підключення зовнішніх гаджетів.

Через веб-інтерфейс користувач отримує доступ до розширеної інформації та можливостей керування. Тут можуть відображатися детальніші дані про сигнал металошукача, стан сенсорів, координати GPS, а також надається можливість змінювати режими роботи або запускати окремі функції системи. Порівняно з локальним дисплеєм, веб-інтерфейс не обмежений розміром екрана, що дозволяє подати інформацію більш детально і структуровано. Це робить

					КВРКІ.2301109.23.01.43 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		46

дистанційний доступ зручним інструментом для аналізу та налаштування системи.

Важливою складовою взаємодії є організація обробки даних. Після зчитування сигналу металошукача та його попередньої обробки система формує цифрове представлення значення, яке далі використовується для прийняття рішень. Цей процес включає усереднення, оцінювання відхилення від базового рівня та визначення наявності значущого сигналу. Отримані дані не лише відображаються, а й можуть зберігатися або передаватися для подальшого аналізу.

Окрім основного сигналу, система обробляє дані з допоміжних сенсорів. Зокрема, координати GPS дозволяють прив'язати результат до конкретної точки на місцевості. Дані з датчиків температури та вологості доповнюють інформацію про умови експлуатації, а показники з ультразвукового датчика можуть використовуватися як додатковий параметр для аналізу. У результаті формується багатовимірний набір даних, який значно перевищує за інформативністю простий сигнал металошукача.

Особливу увагу в системі приділено організації збереження та експорту даних. Під час роботи пристрою можуть фіксуватися точки, у яких зафіксовано підвищений рівень сигналу. Кожна така точка може містити координати, значення сигналу та супровідні параметри. Дані формуються у структурованому вигляді, зокрема у форматі JSON або GeoJSON, що дозволяє використовувати їх у сторонніх програмних засобах. Це відкриває можливість подальшого аналізу, побудови карт, візуалізації результатів у геоінформаційних системах та порівняння отриманих даних у часі.

Завдяки такій організації обробки даних система переходить від простого режиму індикації до режиму накопичення та аналізу інформації. Це суттєво розширює функціональність пристрою та дозволяє використовувати його не лише як інструмент виявлення, а і як засіб документування та дослідження. У практичному застосуванні це означає, що результати роботи не втрачаються після завершення сеансу, а можуть бути використані для подальшої обробки та

					КВРКІ.2301109.23.01.43 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		47

прийняття рішень.

У підсумку організація взаємодії з користувачем та обробки даних у розроблюваній системі побудована як багаторівнева структура, що поєднує локальний інтерфейс, дистанційний доступ і внутрішню логіку обробки інформації. Такий підхід забезпечує зручність використання, гнучкість керування та можливість подальшого аналізу отриманих результатів. Це дозволяє розглядати систему не лише як пристрій для виявлення металу, а як повноцінний програмно-технічний засіб, здатний працювати в умовах сучасних вимог до збору, обробки та представлення даних.

2.6 Реалізація програмної логіки системи

Розглянуто реалізацію програмного забезпечення системи дистанційного керування металошукачем, побудованого на базі мікроконтролера ESP32. Програмна частина сформована як цілісна система, що поєднує обробку сигналу, автоматичну калібровку, взаємодію з користувачем, роботу з сенсорами та організацію бездротового зв'язку. Незважаючи на те, що реалізація виконана у вигляді одного програмного файлу, логічно вона поділена на окремі функціональні модулі, кожен з яких відповідає за конкретний аспект роботи системи.

Основою програмної реалізації є використання стандартного середовища Arduino для ESP32, що дозволяє працювати з апаратними ресурсами мікроконтролера через високорівневі інтерфейси. У програмі використано набір бібліотек, які забезпечують взаємодію з периферійними модулями. Для роботи з OLED-дисплеєм застосовано бібліотеки Adafruit_GFX та Adafruit_SSD1306, які дозволяють формувати текстові та графічні елементи інтерфейсу. Завдяки цьому реалізовано кілька екранів відображення, між якими здійснюється перемикання залежно від стану системи та дій користувача.

Для зчитування температури та вологості використано бібліотеку DHT, яка реалізує протокол обміну з датчиком і повертає значення у зручному форматі.

					КВРКІ.2301109.23.01.43 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		48

Робота з GPS-модулем реалізована через бібліотеку TinyGPS++, яка забезпечує обробку потоку даних і виділення координат. Для організації бездротового зв'язку використано бібліотеки WiFi, WebServer та DNSServer, що дозволяють реалізувати режим точки доступу та клієнтський режим, а також обслуговувати HTTP-запити.

Збереження налаштувань виконано за допомогою бібліотеки Preferences, яка дозволяє записувати параметри у енергонезалежну пам'ять. Це використовується, зокрема, для збереження результатів калібровки та налаштувань мережі. Для роботи з цифровим потенціометром застосовано інтерфейс SPI через стандартну бібліотеку SPI, а взаємодія з дисплеєм здійснюється через інтерфейс I2C за допомогою бібліотеки Wire.

Програмна логіка системи організована навколо функцій `setup()` та `loop()`. У функції `setup()` виконується ініціалізація всіх апаратних компонентів: налаштовуються режими роботи GPIO, запускається дисплей, ініціалізуються сенсори, встановлюється початковий режим роботи Wi-Fi, а також відновлюються збережені параметри. Після завершення ініціалізації система переходить у безперервний цикл `loop()`, у якому послідовно виконуються всі функціональні задачі.

Зчитування сигналу металошукача реалізовано через окрему функцію, що виконує багаторазове читання значення з аналогового входу. Замість одиничного виклику `analogRead()` використовується накопичення декількох відліків із подальшим обчисленням середнього значення. Такий підхід дозволяє зменшити вплив випадкових коливань і сформувати більш стабільний результат.

Автоматична калібровка реалізована як окремий алгоритм із чітко визначеними станами. У програмі використано перелік станів, серед яких початковий режим, грубе налаштування, точне налаштування та завершення. На етапі грубого налаштування виконується перебір значень керуючого параметра з великим кроком. Метою цього етапу є швидке визначення діапазону, в якому сигнал має мінімальний рівень шуму. Після цього система переходить до точного налаштування, де перебір виконується з меншим кроком, що дозволяє знайти

більш точне значення. У процесі калібровки для кожного значення виконується зчитування сигналу та оцінка його стабільності. У підсумку вибирається варіант, що забезпечує найменший розкид значень.

Калібровка запускається під час ініціалізації системи або за запитом користувача. Після її завершення знайдене значення зберігається і використовується в подальшій роботі. У разі виникнення нестабільності система може повторно виконати процедуру калібровки, що забезпечує адаптацію до змін умов експлуатації.

У програмі також реалізовано збір і збереження даних у вигляді структур, що містять координати, значення сигналу та параметри середовища. Кожна така структура відповідає окремій точці вимірювання. Це дозволяє формувати набір даних, який може бути використаний для подальшого аналізу або візуалізації. Окрім реальних даних, у системі передбачено режим генерації тестових значень, що дозволяє перевіряти роботу інтерфейсу без наявності GPS-сигналу.

Бездротовий інтерфейс реалізовано у вигляді двох режимів роботи. У режимі точки доступу мікроконтролер створює власну мережу, до якої може підключитися користувач. У клієнтському режимі система підключається до зовнішньої мережі. Обслуговування запитів здійснюється через вбудований веб-сервер, що дозволяє отримувати дані та взаємодіяти з системою через браузер.

Загалом програмна реалізація побудована таким чином, щоб забезпечити одночасну роботу всіх підсистем у межах одного циклу виконання. У кожній ітерації циклу виконується зчитування сигналів, оновлення інтерфейсу, обробка кнопок і обслуговування мережевих функцій.

У підсумку програмне забезпечення системи реалізовано як інтегрований набір взаємопов'язаних модулів, що забезпечують обробку сигналу, автоматичну калібровку, взаємодію з користувачем і передачу даних. Використання готових бібліотек дозволило спростити реалізацію низькорівневих операцій і зосередитися на логіці роботи системи, що забезпечило досягнення поставленої мети кваліфікаційної роботи.

					КВРКІ.2301109.23.01.43 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		50

3 РЕАЛІЗАЦІЯ ПРОГРАМНО-ТЕХНІЧНОГО ЗАСОБУ ДИСТАНЦІЙНОГО КЕРУВАННЯ МЕТАЛОШУКАЧЕМ

3.1 Загальний опис експериментального зразка системи

Реалізовано експериментальний зразок програмно-технічного засобу дистанційного керування металошукачем, у якому поєднано аналоговий пошуковий вузол, мікроконтролерну платформу, допоміжні сенсори, локальний інтерфейс користувача та засоби бездротової взаємодії. На відміну від спрощених стендових макетів, де окремі модулі розміщуються без конструктивного впорядкування, у даному випадку сформовано цілісний прототип. Це дозволило перейти від етапу базової перевірки схемотехнічних рішень до повноцінного тестування пристрою в умовах, наближених до реальної експлуатації, оцінюючи узгоджену роботу всієї системи.

Зовнішній вигляд експериментального зразка наведено на рисунку 3.1. На цьому зображенні добре простежується загальна конструктивна побудова пристрою: пошукова котушка винесена окремо від основного корпусного блока та з'єднана з ним через кабель. Така компоновка є цілком виправданою, оскільки дає змогу організувати зручне переміщення чутливого елемента в зоні пошуку, водночас зберігаючи електронні модулі в межах захищеного блока. Сам корпус виконано у вигляді компактного закритого контейнера, що надає пристрою завершеного вигляду, захищає плати від зовнішніх впливів та спрощує експлуатацію експериментального зразка.

Завдяки такому чіткому функціональному поділу на зону безпосереднього пошуку та зону обробки й керування, система набуває ознак повноцінного приладу. Подібне просторове розмежування не лише покращує ергономіку, знімаючи зайве вагове навантаження з пошукової частини, а й мінімізує вплив цифрових шумів мікроконтролера на високочутливий аналоговий тракт металошукача. За рисунком 3.1 можна зробити висновок, що під час реалізації прототипу акцент було зміщено не лише на працездатність електричної частини, а й на загальну конструктивну цілісність.

					КВРКІ.2301109.23.01.43 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		51



Рисунок 3.1 – Загальний вигляд програмно-технічного засобу металошукача

Внутрішня апаратна реалізація системи наведена на рисунку 3.2. Саме це зображення дає змогу більш детально оцінити реальне компонування модулів усередині корпусу та показує, як сформовано взаємодію між окремими електронними вузлами. У центральній частині конструкції розміщено основну плату з мікроконтролером ESP32, до якої підключено дисплей, сенсорні модулі та інші допоміжні елементи.

Поряд із центральною платою видно навігаційний модуль із керамічною антеною, датчик температури та вологості, локальний OLED-дисплей, а також з'єднувальні провідники, які формують загальну логіку міжмодульної взаємодії.

Особливо важливим є те, що внутрішнє розміщення компонентів має не випадковий характер. Із рисунка 3.2 видно, що центральна плата з ESP32 фактично виконує роль ядра всієї системи, навколо якого згруповано інші модулі.

той самий час така побудова вже не виглядає хаотичною: основні елементи згруповано в межах одного корпусу, а проводів зв'язки формують завершену внутрішню структуру.

На внутрішніх фотографіях також добре простежується, що в системі реалізовано не лише базовий пошуковий вузол, а й розширений набір допоміжних елементів. Це свідчить про комплексний характер експериментального зразка. Пристрій не обмежується лише функцією фіксації зміни сигналу металошукача, а одночасно підтримує локальне відображення інформації, збір навігаційних даних та роботу з додатковими сенсорними параметрами. Унаслідок цього система переходить від рівня звичайного металошукача до рівня багатфункціонального програмно-технічного засобу, в якому вимірювальна частина тісно інтегрована з цифровою логікою.

Ще однією важливою особливістю експериментального зразка є поєднання закритого корпусного виконання з доступністю до внутрішніх модулів. Це означає, що прототип можна використовувати як для демонстрації завершеної конструкції, так і для інженерного аналізу окремих вузлів. На рівні кваліфікаційної роботи це є суттєвою перевагою, оскільки дозволяє одночасно показати і зовнішній вигляд системи, і її внутрішню технічну реалізацію. Саме тому використання двох окремих ілюстрацій для цього підрозділу є цілком виправданим: перший рисунок демонструє загальний вигляд пристрою, а другий - його реальну внутрішню побудову.

З огляду на представлені рисунки можна стверджувати, що експериментальний зразок уже реалізовано як працездатний комплекс, у якому всі основні елементи об'єднано в єдину структуру. Пошукова котушка виконує функцію первинного взаємодійного елемента з об'єктом контролю, корпусний блок містить обчислювальну та інтерфейсну частини, а внутрішні модулі забезпечують збір, обробку, відображення і передавання даних. Це дозволяє розглядати створений зразок не як абстрактну експериментальну схему, а як завершений прототип, на основі якого вже можна виконувати практичну

					КВРКІ.2301109.23.01.43 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		54

демонстрацію режимів роботи, аналіз обробки сигналу, перевірку калібровки та оцінювання взаємодії з користувачем у наступних підрозділах.

3.2 Демонстрація роботи системи та режимів функціонування

Після складання експериментального зразка та завершення базового налагодження було виконано практичну демонстрацію роботи системи в основних режимах функціонування. На цьому етапі перевірялося не лише те, чи запускається пристрій після подачі живлення, а й те, наскільки узгоджено між собою працюють мікроконтролерний блок, локальний інтерфейс, модулі збору даних, підсистема калібровки та мережевий доступ. Саме така перевірка дозволяє перейти від суто проєктного опису до підтвердження того, що система дійсно функціонує як завершений програмно-технічний засіб.

У процесі запуску система переходить у стан ініціалізації, під час якого активуються основні програмні модулі, готуються до роботи інтерфейси введення-виведення, виконується запуск дисплея, зчитуються збережені параметри та формується поточний робочий режим. Після завершення цього етапу на локальному OLED-дисплеї відображається головний екран, який виконує роль основного інформаційного вікна для користувача. Як видно з рисунка 3.3, на цьому екрані подається зведена інформація про стан системи: активний мережевий режим, службові параметри, поточне значення сигналу, а також дані від допоміжних сенсорів.

Головний екран є важливим з точки зору практичного використання, оскільки саме він дозволяє без переходу до інших режимів оцінити, чи перебуває пристрій у робочому стані. На ньому відображаються не лише параметри самого металошукача, а й пов'язана з ним супровідна інформація. Це означає, що локальний інтерфейс не зводиться до простого виведення одного числового значення, а вже на базовому рівні надає узагальнену картину роботи системи. За таким підходом користувач одразу бачить не лише факт функціонування

пристрою, а й поточний режим мережевої роботи, значення вимірювального сигналу, а також частину метаданих, що супроводжують процес вимірювання.

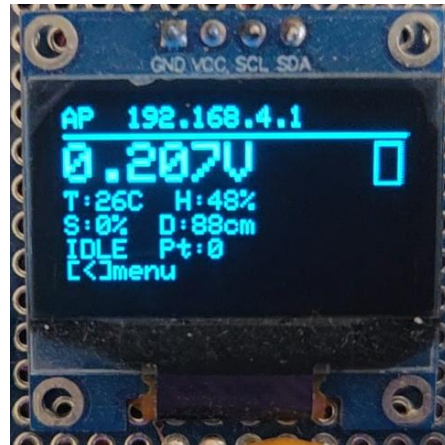


Рисунок 3.3 – Головний екран системи на OLED-дисплеї

Після запуску система забезпечує перехід до меню керування, яке реалізовано через кнопочий інтерфейс. З рисунка 3.4 видно, що меню побудовано у вигляді списку функціональних пунктів, між якими користувач може послідовно переміщатися.

У представленому варіанті чітко виділено кілька основних напрямів взаємодії: налаштування мережевого режиму, робота з GPS або тестовими даними, перегляд параметрів датчика вологості ґрунту та запуск калібровки.



Рисунок 3.4 – Меню керування системою

Це свідчить про те, що структура меню сформована логічно й орієнтована не на демонстрацію окремих фрагментів коду, а на реальні сценарії роботи з пристроєм.

Побудова меню саме в такому вигляді є виправданою, оскільки кожен із пунктів відповідає окремому функціональному блоку системи. У межах практичної демонстрації це дозволило перевірити, наскільки зручно здійснюється локальна навігація, чи не виникає затримок під час переходів між пунктами, а також чи зберігається зрозуміла послідовність взаємодії. Було встановлено, що навіть за використання компактного дисплея і кнопочового керування система залишається достатньо зручною в експлуатації, оскільки користувач не втрачає логіку переходів між режимами і може швидко перейти до потрібної функції.

Окрему увагу під час демонстрації приділено режиму автоматичної калібровки. На рисунку 3.5 показано, що калібровка в системі реалізована не як прихований фоновий процес, а як окремий керований режим із вибором сценарію виконання. На екрані відображаються доступні варіанти, серед яких виділено кілька профілів, орієнтованих на різні умови використання. Це важливий момент, оскільки він підтверджує, що система не обмежується одним жорстко фіксованим алгоритмом, а дозволяє адаптувати логіку калібровки до конкретного режиму роботи.

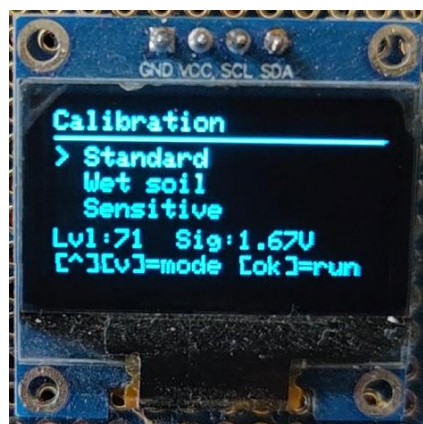


Рисунок 3.5 – Режим автоматичної калібровки

У практичному плані такий підхід виявився зручним, оскільки дає змогу ще на рівні локального інтерфейсу обрати потрібний сценарій налаштування без втручання у програмний код. Це особливо важливо для експериментального зразка, де система використовується в різних умовах та потребує швидкого переходу між режимами без перепрошивки або ручного переналаштування окремих елементів. Сам факт того, що процес калібровки виведено на рівень повноцінного екранного режиму, уже свідчить про завершеність реалізації та орієнтацію не лише на налагодження, а й на практичну експлуатацію.

Ще одним суттєвим режимом функціонування є перегляд даних від допоміжних сенсорів. На рисунку 3.6 показано екран, на якому відображаються значення, отримані від датчиків температури, вологості та інших підключених модулів. Цей рисунок добре демонструє, що система не обмежується обробкою лише основного вимірювального сигналу, а працює як багатофункціональний комплекс збору даних. У результаті користувач отримує доступ не тільки до інформації про стан металошукача, а й до супровідних параметрів, які можуть бути корисними для аналізу умов експлуатації.

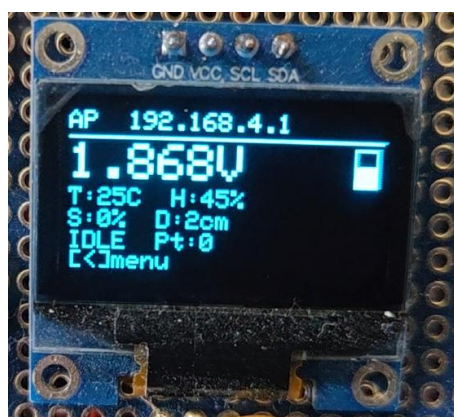


Рисунок 3.6 – Відображення параметрів сенсорів під час роботи

Використання такого екрану є особливо доцільним у контексті польового або експериментального застосування, коли результат роботи системи бажано розглядати не ізольовано, а разом із параметрами середовища. Наприклад, температура, вологість повітря, вологість ґрунту чи відстань можуть прямо не

визначати факт виявлення металу, але вони допомагають повніше інтерпретувати умови, у яких отримано конкретний результат. Саме тому включення сенсорних показників до загальної структури інтерфейсу є виправданим як з функціонального, так і з аналітичного погляду.

Під час демонстрації також було перевірено, як система поводить себе під час переходів між різними режимами. У цьому аспекті важливим є те, що зміна екрану або вибір нового пункту меню не призводять до порушення загальної роботи пристрою. Іншими словами, система продовжує підтримувати вимірювальний цикл, зберігає актуальність даних та не втрачає зв'язок між окремими підсистемами. Це означає, що інтерфейсна частина не конфліктує з вимірювальною і не блокує виконання основних функцій.

Серед режимів функціонування окремо було перевірено й мережеві сценарії роботи. На головному екрані та в меню вже видно, що система підтримує щонайменше два базові варіанти організації бездротового доступу: автономний режим із власною точкою доступу та режим підключення до зовнішньої мережі. У практичному використанні це дозволяє змінювати спосіб взаємодії залежно від умов. Якщо зовнішня мережа відсутня, система працює самостійно й забезпечує підключення через власний мережевий інтерфейс. Якщо ж необхідно поєднати роботу пристрою з уже наявною мережею, користувач може перейти до відповідного режиму через меню. Наявність таких варіантів підтверджує, що система орієнтована не лише на лабораторне тестування, а й на більш гнучке практичне використання.

Окремо варто звернути увагу на режим GPS/SEED, що також відображено в структурі меню. Наявність такого пункту демонструє, що в системі передбачено не лише роботу з реальними координатними даними, а й можливість використання тестового сценарію. Це є важливим практичним рішенням, оскільки під час роботи в приміщенні або в умовах нестабільного супутникового сигналу демонстрація функціональності системи не зривається, а може виконуватися з використанням змодельованих даних. Такий підхід суттєво підвищує гнучкість експлуатації та зручність перевірки програмної частини.

Ще однією важливою ознакою завершеності системи є те, що всі перелічені режими функціонують у межах єдиного програмного середовища й не вимагають перемикання між окремими прошивками чи різними апаратними конфігураціями. Це означає, що користувач працює з цілісною системою, у якій вимірювання, калібровка, локальний інтерфейс, сенсорні режими та мережеві сценарії інтегровано в одному програмно-технічному засобі. Саме така інтеграція і є одним із головних результатів виконаної кваліфікаційної роботи.

У підсумку практична демонстрація підтвердила, що експериментальний зразок здатний стабільно працювати в різних режимах функціонування, а локальний інтерфейс на OLED-дисплеї забезпечує достатній рівень інформативності та керованості. Головний екран надає зведену інформацію про стан системи, меню забезпечує доступ до основних функціональних блоків, режим калібровки дозволяє адаптувати роботу пристрою до поточних умов, а екран сенсорних параметрів розширює інформативність результатів. Усе це дозволяє вважати, що система пройшла не лише теоретичне проектування, а й практичну перевірку в умовах реального функціонування.

3.3 Аналіз обробки сигналу та результатів вимірювань

Одним із найважливіших етапів перевірки працездатності розробленого програмно-технічного засобу став аналіз того, як система обробляє сигнал металошукача та яким чином змінюються результати вимірювань у різних режимах роботи. Саме на цьому рівні можна об'єктивно оцінити, наскільки вдало поєднано аналогову частину з цифровою обробкою, чи достатньо стабільно поводить ся сигнал у стані спокою, як система реагує на появу металевих об'єктів та наскільки помітною є різниця між фоновим режимом і режимом виявлення.

У процесі роботи системи первинний сигнал, що надходить із металошукача, не використовується безпосередньо як готовий результат. Після проходження апаратного узгодження та подальшого зчитування

					КВРКІ.2301109.23.01.43 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		60

мікроконтролером він перетворюється на послідовність цифрових значень, які вже можуть аналізуватися програмно. На цьому етапі сигнал ще не є повністю стабільним, оскільки навіть у стані спокою спостерігаються незначні відхилення, зумовлені залишковими шумами, особливостями вимірювального тракту, впливом живлення та загальною динамікою аналогової частини. Саме тому під час аналізу особливу увагу було приділено не окремим миттєвим значенням, а характеру зміни сигналу в часі та його поведінці після цифрової обробки.

Початково сигнал було досліджено у стані спокою, тобто за відсутності металевго об'єкта поблизу пошукової котушки. У цьому режимі система працює як базова вимірювальна структура, що дозволяє оцінити фоновий рівень і природну нестабільність вимірювань. У результаті спостережень встановлено, що у стані спокою сигнал не є абсолютно незмінним, однак його коливання залишаються в обмеженому діапазоні. Після застосування усереднення ці зміни стають менш різкими, а загальна форма сигналу набуває більш стабільного характеру. Саме така поведінка і є бажаною, оскільки вона створює умови для коректного формування базового рівня, від якого надалі відраховуються всі значущі відхилення. Такі незначні фонові коливання є цілком нормальним явищем для високочутливих вимірювальних систем. Вони відображають сумарний вплив теплових шумів електронних компонентів, мікрофлуктуацій напруги живлення та природного електромагнітного фону середовища. Графічне представлення цього процесу наочно доводить, що робоча точка системи надійно утримується в межах вузького коридору значень, який алгоритми мікроконтролера ESP32 успішно компенсують. Типовий вигляд сигналу металошукача у стані спокою наведено на рисунку 3.7.

Аналіз даних у фоновому режимі показав, що навіть за відсутності зовнішнього впливу повністю ідеальна лінія не формується. Це означає, що система повинна працювати не за принципом реагування на будь-яку зміну, а за принципом оцінювання відхилення від допустимого фонового діапазону. Саме тому у програмній логіці використовується не одиничне вимірювання, а серія зчитувань із наступним усередненням.

					КВРКІ.2301109.23.01.43 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		61

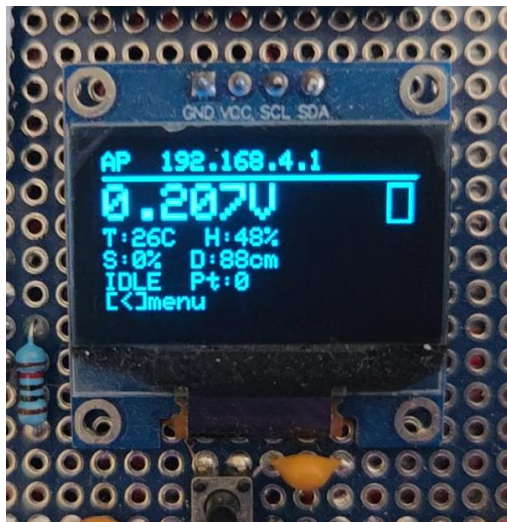


Рисунок 3.7 – Сигнал металошукача у стані спокою

Унаслідок цього короточасні випадкові стрибки не призводять до хибного спрацювання, а сигнал у стані спокою набуває форми, придатної для подальшого порівняння з робочими значеннями.

Після цього було досліджено поведінку системи при появі металевого об'єкта поблизу котушки. У такому режимі сигнал починає змінюватися вже значно помітніше. На відміну від стану спокою, де спостерігаються лише незначні фонові коливання, у режимі виявлення з'являється стійке відхилення від базового рівня. При цьому характер зміни залежить від кількох факторів: відстані до об'єкта, його розмірів, положення відносно котушки та загальної чутливості системи після калібровки. У практичному вимірюванні це проявляється як зростання або зміщення рівня сигналу, яке вже не можна пояснити випадковим шумом. Приклад такого сигналу наведено на рисунку 3.8.

Порівняння рисунків 3.7 і 3.8 дозволяє зробити кілька важливих висновків. По-перше, у стані спокою сигнал перебуває в обмеженому діапазоні та має відносно передбачуваний характер зміни. По-друге, при появі металевого об'єкта характер сигналу вже змінюється не випадково, а закономірно, тобто система фіксує стійке відхилення від фону. По-третє, саме використання цифрової обробки дає змогу виділити ці зміни не як одиничні стрибки, а як інформативні події, які можуть бути використані для формування результату вимірювання.

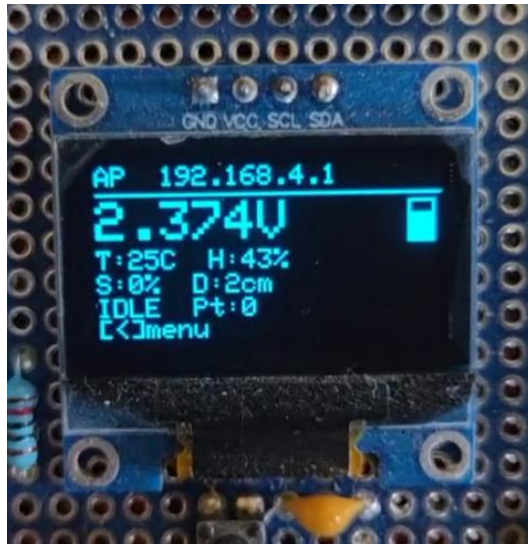


Рисунок 3.8 – Зміна сигналу при виявленні металевого об’єкта

У межах аналізу обробки сигналу окремо оцінювався вплив програмного усереднення. Без такого етапу система була б значно чутливішою до дрібних випадкових змін, а кількість хибних реакцій на нестабільність сигналу зросла б. Усереднення дозволяє не лише згладити окремі короткі сплески, а й сформувати більш надійну оцінку поточного стану сигналу. У практичному сенсі це означає, що система реагує не на окремий випадковий пік, а на реальну зміну загальної тенденції у вибірці вимірювань. Завдяки цьому логіка прийняття рішення стає більш стійкою та передбачуваною.

Також у процесі перевірки було встановлено, що важливе значення має не лише саме усереднене значення, а й різниця між ним і базовим рівнем, який відповідає стану спокою. Саме ця різниця фактично виступає основним критерієм виявлення. Якщо поточний сигнал відхиляється від базового в межах допустимого шумового діапазону, система залишається у звичайному режимі. Якщо ж це відхилення стає стійким і перевищує порогове значення, формується реакція на виявлення. Такий підхід є більш надійним, ніж орієнтація на абсолютне значення сигналу, оскільки дозволяє враховувати реальний фоновий стан системи.

Окрему увагу під час аналізу було приділено тому, як система поводить себе в перехідних режимах, коли металевий об’єкт лише наближається до котушки

					КВРКІ.2301109.23.01.43 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		63

або поступово віддаляється. У таких випадках зміна сигналу має не стрибкоподібний, а плавний характер. Це особливо важливо, оскільки дозволяє оцінити не лише сам факт спрацювання, а й загальну динаміку реакції системи. У процесі експериментальної перевірки було помічено, що при поступовому наближенні об'єкта система формує плавне збільшення інтенсивності сигналу, а при віддаленні - поступове повернення до базового рівня. Така поведінка свідчить про достатню чутливість і коректну роботу алгоритму цифрової обробки.

Ще одним важливим результатом стало підтвердження того, що обробка сигналу не відбувається ізольовано від інших параметрів системи. Під час вимірювань паралельно з основним сигналом фіксувалися допоміжні дані з інших сенсорів, зокрема температура, вологість, відстань та координати. Хоча ці параметри не впливають безпосередньо на факт виявлення металу, вони дозволяють доповнити основний результат супровідною інформацією та створити ширший контекст вимірювання. Унаслідок цього одиничний сигнал металошукача вже не розглядається як ізольоване значення, а входить до загального масиву даних, який може використовуватися для подальшого аналізу.

У процесі дослідження також було встановлено, що результати вимірювань є більш переконливими тоді, коли оцінюється не один окремих момент, а серія послідовних значень. Саме тому система орієнтується не лише на одиничний пік сигналу, а на його стійке повторення протягом певного проміжку часу. Такий підхід дозволяє виключити випадкові короточасні аномалії та зробити реакцію системи більш обґрунтованою. У практичній роботі це проявляється у зменшенні кількості помилкових спрацювань та більш передбачуваний поведінці пристрою.

З точки зору оцінювання результатів вимірювань найбільш важливим є те, що система демонструє чітко помітну різницю між фоновим режимом і режимом реального виявлення. Саме ця різниця й підтверджує працездатність програмно-технічного засобу як засобу вимірювання та інтерпретації сигналу. Якщо б сигнал у стані спокою і при наявності металевго об'єкта майже не відрізнявся,

					КВРКІ.2301109.23.01.43 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		64

це означало б низьку практичну цінність усієї системи. У розробленому зразку така різниця простежується, а отже можна стверджувати, що запропоновані підходи до обробки сигналу є виправданими.

У підсумку аналіз обробки сигналу та результатів вимірювань показав, що система забезпечує коректне розмежування між станом спокою та станом виявлення металевих об'єктів. Використання цифрового усереднення, порівняння з базовим рівнем і оцінювання стійкості відхилення дозволяє отримати достовірні результати та зменшити вплив шумів. Це підтверджує доцільність обраного підходу до програмної обробки сигналу та показує, що експериментальний зразок може розглядатися як працездатна основа для подальшого розвитку системи дистанційного керування металодетектором.

3.4 Перевірка ефективності автоматичної калібровки

Одним із найбільш показових етапів практичної перевірки розробленої системи стала оцінка роботи автоматичної калібровки, оскільки саме вона визначає, наскільки стабільно поводитиметься металодетектор після запуску та чи буде цифрова обробка сигналу виконуватися в коректному робочому діапазоні. Для даного програмно-технічного засобу калібровка не є допоміжною процедурою, яку можна розглядати окремо від основного процесу вимірювання. Навпаки, вона виступає стартовим режимом, від якого безпосередньо залежить подальша якість виявлення, рівень фонових коливань і стійкість системи до помилкових спрацювань. На практиці це означає, що без правильно визначеної точки спокою навіть коректно зібрана апаратна частина не забезпечує достатньо передбачуваної поведінки в робочому режимі.

Під час демонстрації було встановлено, що запуск калібровки реалізовано не як прихований фоновий процес, а як окремий керований режим, доступний через меню пристрою. На екрані системи відображаються кілька варіантів калібрування, серед яких передбачено стандартний режим, режим для вологого ґрунту та більш чутливий режим. Така побудова вже сама по собі є важливою

					КВРКІ.2301109.23.01.43 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		65

ознакою завершеності програмної реалізації, оскільки дозволяє не лише автоматизувати налаштування, а й адаптувати його до різних умов експлуатації. Унаслідок цього система не обмежується однією жорстко фіксованою процедурою, а надає можливість запускати алгоритм у різних сценаріях залежно від характеру середовища та бажаного рівня чутливості. Це добре видно на екрані режиму калібровки, де відображено перелік доступних профілів і поточний вимірний рівень сигналу перед початком процедури.

Практична перевірка ефективності калібровки виконувалася як послідовне спостереження за всіма її етапами. На початковій стадії система повідомляє про старт процедури та починає пошук такого рівня керуючого параметра, за якого сигнал наближається до стану спокою. На екрані відображаються як службові повідомлення про поточну фазу, так і числові значення, що характеризують обраний рівень і відповідну реакцію сигналу. Уже на цьому етапі видно, що алгоритм не працює за принципом випадкового підбору, а реалізує керований пошук, під час якого мікроконтролер послідовно змінює параметр цифрового потенціометра й аналізує результат кожного кроку. Це підтверджує, що в системі реалізовано не просто автоматичне ввімкнення калібровки, а повноцінний алгоритм із чітко визначеною логікою проходження між станами.

Подальше спостереження за процесом показало, що калібровка поділяється на два основні етапи - грубе сканування та точне сканування. На стадії грубого пошуку система проходить по більшому діапазону значень і швидко визначає область, у якій сигнал набуває більш стабільного характеру. Відповідне відображення на дисплеї показує не лише назву етапу, а й поточний прогрес його виконання, що дає змогу оцінити динаміку процесу. Після цього система переходить до етапу тоншого налаштування, де зміна керуючого параметра виконується вже з меншою дискретністю. Саме ця друга стадія забезпечує точніше визначення робочого значення та дозволяє наблизитися до такого стану, у якому фоновий рівень сигналу є найбільш придатним для подальшого вимірювання. Такий поділ на грубий і точний пошук є технічно

виправданим, оскільки дає можливість поєднати швидкість первинного сканування з точністю фінального налаштування.

На основі зображень екранів можна простежити, що під час калібровки система не просто проходить по рівнях, а реально оцінює відповідні значення сигналу. На стартовому етапі зафіксовано одні значення, далі в процесі грубого сканування вони змінюються, а на етапі точного пошуку зменшуються до меншого рівня. Завершення процедури супроводжується окремим повідомленням про успішне виконання та фіксацією знайденого значення рівня разом із поточним значенням сигналу. Особливо важливо, що на завершальному екрані додатково вказано факт збереження результату в енергонезалежній пам'яті. Це означає, що калібровка не є тимчасовою операцією лише для поточного сеансу, а формує робочий параметр, який надалі може бути використаний після повторного ввімкнення пристрою. З практичного погляду це суттєво підвищує зручність експлуатації, оскільки зменшує потребу в повторному повному налаштуванні після кожного перезапуску системи.

Ефективність калібровки доцільно оцінювати не тільки за фактом успішного завершення, а й за тим, як змінюється характер сигналу до і після її виконання. Під час демонстрації було видно, що до завершення підбору робочого рівня система може працювати з помітно вищими або менш стабільними значеннями, тоді як після успішного завершення алгоритму формується значно спокійніший базовий режим. У цьому і полягає головна практична цінність процедури: вона не просто знаходить чергове число, а переводить вимірювальний тракт у такий стан, за якого цифрова частина отримує більш рівномірний фоновий сигнал. Саме цей стан надалі використовується як опорний для виявлення реальних відхилень, що вже можуть бути пов'язані з наявністю металевих об'єктів. Унаслідок цього робота системи після калібровки стає більш передбачуваною, а порогова логіка отримує стабільнішу основу для прийняття рішення.

					КВРКІ.2301109.23.01.43 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		67

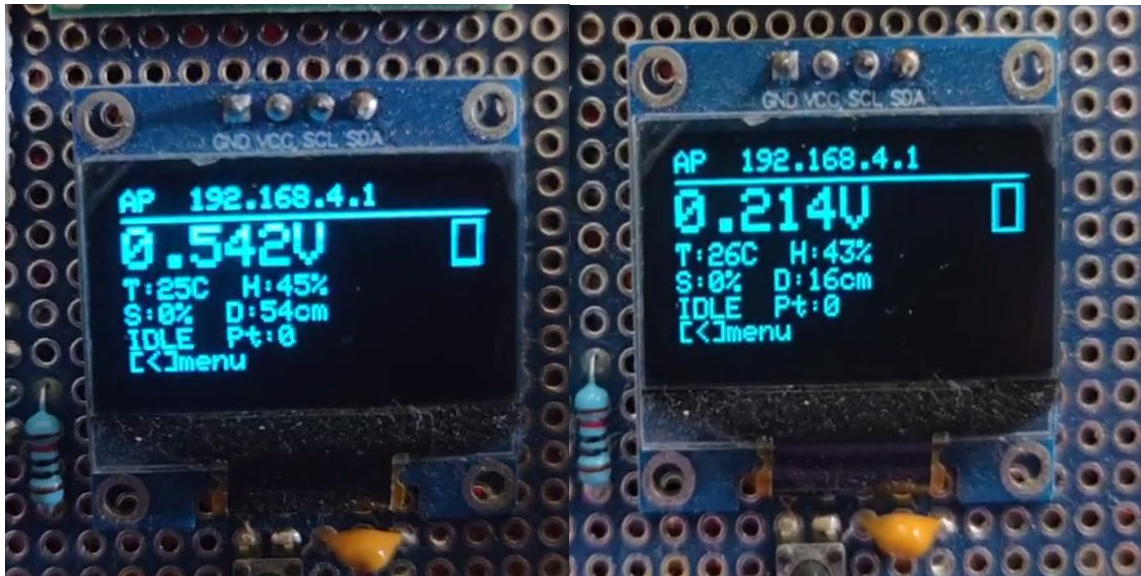


Рисунок 3.9 – Порівняння сигналу до та після автоматичної калібровки

Важливо й те, що під час перевірки було зафіксовано не лише успішний сценарій роботи, а й можливість аварійного завершення процедури. На одному з екранів система формує повідомлення про помилку, пов'язану з неможливістю знайти “тихий” режим. Наявність такого сценарію є не недоліком, а навпаки, свідченням коректної реалізації алгоритму. Це означає, що програмна логіка передбачає не лише сприятливий варіант, а й ситуацію, коли через зовнішні умови або поточний стан вимірювального тракту неможливо досягти прийняттого фону. У такому випадку користувач не залишається без пояснення, а отримує чітке повідомлення про проблему та можливість повторити процедуру. Для експериментального зразка це є важливою перевагою, оскільки підтверджує наявність елементів контролю помилок і підвищує надійність системи в реальних умовах використання.

Ще одним важливим результатом перевірки стало підтвердження того, що автоматична калібровка не зводиться до простого “заглушення” сигналу. Ефективність процедури полягає не в тому, щоб зробити систему максимально нечутливою, а в тому, щоб зменшити фонові коливання і при цьому зберегти здатність реагувати на корисні зміни. Це добре узгоджується з практикою роботи металошукача: надмірне придушення сигналу зменшило б кількість шумів, але одночасно знизило б і чутливість до слабких об’єктів. Результати спостереження

за калібровкою показали, що після завершення процедури система переходить у більш спокійний режим, але не втрачає працездатності. Саме такий баланс між стабільністю і чутливістю і можна вважати основною ознакою якісної автоматичної калібровки.

З точки зору користувача реалізований алгоритм також демонструє високу практичну доцільність. Якщо в класичному підході налаштування робочої точки виконувалося б вручну, це вимагало б постійного контролю, суб'єктивної оцінки рівня шуму та багаторазового підлаштування. У даній системі ця задача вже перенесена на рівень програмної логіки, а користувач отримує окремий керований режим, у якому результат формується автоматично. Це значно спрощує підготовку пристрою до роботи, зменшує вплив людського фактора і робить систему придатнішою до дистанційного використання.

У підсумку перевірка ефективності автоматичної калібровки показала, що реалізований у системі алгоритм виконує свою функцію коректно. Він проходить через послідовні стадії пошуку, відображає хід процедури на локальному інтерфейсі, дозволяє адаптувати налаштування до різних режимів, зберігає результат після завершення та передбачає окремий сценарій для аварійної ситуації. Найважливішим результатом є те, що після виконання калібровки базовий режим роботи системи стає стабільнішим, а це безпосередньо підвищує якість подальшої цифрової обробки сигналу. Саме тому автоматична калібровка в розробленому програмно-технічному засобі виступає не допоміжним сервісом, а одним із ключових механізмів забезпечення надійної роботи всієї системи.

3.5 Аналіз взаємодії з користувачем та передачі даних

Однією з важливих переваг розробленого програмно-технічного засобу є те, що взаємодія з ним організована одразу на двох рівнях - локальному та дистанційному. Такий підхід дозволяє використовувати систему як автономний пристрій із власним інтерфейсом керування і водночас як мережевий вузол, до якого можна підключатися зі смартфона або іншого клієнтського пристрою.

					КВРКІ.2301109.23.01.43 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		69

Саме поєднання цих двох способів роботи суттєво підвищує зручність експлуатації та робить систему більш універсальною в реальних умовах використання.

Локальна взаємодія реалізована за допомогою OLED-дисплея та кнопочового блоку керування. У процесі перевірки було підтверджено, що такий інтерфейс є достатнім для виконання основних операцій без потреби у зовнішньому пристрої. Головний екран відображає поточний стан системи, рівень сигналу та службові параметри, що дозволяє швидко оцінити, чи знаходиться пристрій у робочому режимі, чи завершено калібровку, а також чи надходять дані з допоміжних сенсорів. Приклад такого екрана наведено додатку Д.1.

Попри те, що основна демонстрація дистанційної взаємодії подається через веб-інтерфейс, локальний дисплей залишається важливим елементом системи. Саме через нього здійснюється базова навігація по меню, запуск окремих режимів та перегляд службових параметрів. Під час практичної перевірки встановлено, що використання чотирьох кнопок - для входу в меню, підтвердження вибору та переміщення вгору і вниз - забезпечує достатньо просту й зрозумілу логіку керування. Така організація не перевантажує користувача і водночас дозволяє змінювати режими роботи без перепідключення пристрою або використання комп'ютера.

Структура меню побудована так, щоб розділити основні сценарії використання системи. У межах локального інтерфейсу користувач може перейти до налаштувань бездротового режиму, перегляду GPS-даних, запуску калібровки, перегляду сенсорної інформації та інших службових функцій. Це важливо не лише з погляду зручності, а й з погляду незалежності системи від зовнішнього керування. Навіть без доступу до смартфона пристрій зберігає працездатність і дає змогу виконувати основні дії на рівні меню.

Разом із локальним інтерфейсом у системі реалізовано дистанційну взаємодію через Wi-Fi. Саме цей механізм найбільшою мірою відповідає загальній темі кваліфікаційної роботи, оскільки забезпечує віддалений доступ до

					КВРКІ.2301109.23.01.43 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		70

даних металошукача та параметрів його функціонування. У процесі випробування підтверджено працездатність двох режимів бездротової роботи. У першому режимі мікроконтролер створює власну точку доступу, що дозволяє підключитися до системи без використання зовнішньої мережевої інфраструктури. Такий варіант є зручним у польових умовах, де відсутній доступ до стаціонарної мережі. У другому режимі система підключається до вже наявної мережі, наприклад до точки доступу мобільного телефона, що розширює можливості використання додаткових мережевих сервісів.

Використання веб-інтерфейсу є одним із найбільш вдалих рішень у розробленій системі, оскільки взаємодія з пристроєм виконується через звичайний браузер. Це означає, що для доступу не потрібне окреме прикладне програмне забезпечення, а сам користувач може працювати з будь-якого сумісного пристрою. Під час перевірки веб-інтерфейсу було встановлено, що він дозволяє отримувати поточні значення сигналу, переглядати стан системи, працювати з додатковими даними та загалом забезпечує більш зручний формат представлення інформації, ніж компактний OLED-дисплей. Саме завдяки веб-інтерфейсу система переходить від рівня локального пристрою до рівня повноцінного мережевого програмно-технічного засобу.

Окрему увагу в межах цього підрозділу доцільно приділити тому, як система формує та передає дані. У програмній логіці реалізовано збір не лише основного вимірювального сигналу, а й супровідних параметрів, що суттєво розширює інформаційну цінність результату. Кожна зафіксована точка може містити координати, інтенсивність сигналу, значення температури, вологості, відстані та інші параметри середовища. У такому вигляді дані вже не є простим числовим показником, а перетворюються на структурований набір інформації, який може бути використаний для подальшого аналізу.

Практичне значення такого підходу полягає в тому, що система забезпечує не лише поточне відображення результату, а й підготовку даних до збереження та експорту. Під час роботи пристрій накопичує інформацію про точки

					КВРКІ.2301109.23.01.43 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		71

вимірювання, а далі вона може бути представлена у структурованому форматі. Приклад такого представлення наведено у додатку Д.2.

Використання структурованого формату даних є важливим не лише з технічного, а й з практичного погляду. Якщо результат роботи системи існує лише у вигляді значення на дисплеї, то після завершення вимірювання він фактично втрачається. Якщо ж дані формуються у форматі, придатному для подальшої обробки, система набуває значно ширших можливостей. У цьому випадку результати можуть накопичуватися, переноситися до інших програмних засобів, порівнюватися між собою та аналізуватися у просторовому розрізі.

Саме тому важливою частиною перевірки стала оцінка координатної прив'язки та подальшого відображення результатів на карті. Якщо під час вимірювання доступні GPS-координати, система може формувати точки, придатні для просторової візуалізації. Це особливо корисно для сценаріїв, де металошукач використовується не просто для епізодичного спрацювання, а для послідовного обстеження території. У такому випадку результат уже можна розглядати як карту або набір координатно прив'язаних подій. Приклад такого відображення подано у додатку Д.3.

Ще однією сильною стороною системи є те, що взаємодія з користувачем не обмежується лише “читанням інформації”. Меню та веб-інтерфейс дозволяють не тільки переглядати параметри, а й впливати на режими роботи. Це означає, що користувач отримує не пасивний індикатор, а керований засіб, у якому можна змінювати сценарій використання залежно від поточної задачі. У практичному сенсі це особливо важливо для експериментального зразка, оскільки дозволяє швидко перевіряти різні режими без внесення змін у код або апаратну схему.

Під час перевірки також було встановлено, що система зберігає цілісність логіки при переході між локальним і дистанційним рівнями керування. Інакше кажучи, використання веб-інтерфейсу не скасовує роботу локального меню, а локальний дисплей не заважає мережевому обміну. Обидва механізми доповнюють один одного. Така побудова є вдалою, оскільки робить систему

					КВРКІ.2301109.23.01.43 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		72

гнучкою: в одному випадку її зручно використовувати як автономний пристрій із кнопками та дисплеєм, в іншому - як мережевий вузол з віддаленим доступом до даних.

Додатково варто відзначити високу надійність роботи бездротового каналу зв'язку в умовах польової експлуатації. Використання мікроконтролера ESP32 у режимі автономної точки доступу гарантує стабільне з'єднання між металошукачем та мобільним пристроєм оператора навіть на місцевості, де повністю відсутнє покриття стільникових мереж. Це критично важливо для безперервного моніторингу, оскільки втрата зв'язку могла б призвести до пропуску значущих сигналів або втрати просторових метаданих. Веб-сервер, розгорнутий безпосередньо на мікроконтролері, оптимізовано для швидкої віддачі сторінок та мінімізації мережевих затримок, що забезпечує оновлення показників вимірювального тракту в режимі, максимально наближеному до реального часу.

Досягнення такого результату стало можливим завдяки використанню асинхронного обміну даними між клієнтським пристроєм та мікроконтролером. Це виключає необхідність повного перезавантаження веб-сторінки при кожній зміні показників, суттєво економить енергетичні ресурси пристрою та робить відображення динамічних змін сигналу плавним і комфортним для візуального сприйняття оператором.

У підсумку аналіз взаємодії з користувачем та передачі даних показав, що розроблений програмно-технічний засіб забезпечує достатньо високий рівень функціональної зручності. Локальний інтерфейс дозволяє працювати із системою без зовнішніх пристроїв, а бездротовий доступ значно розширює можливості керування та моніторингу. Формування структурованих даних і можливість координатної прив'язки результатів підвищують практичну цінність системи та дозволяють розглядати її не лише як металошукач, а як інструмент збору, подання та подальшого аналізу вимірювальної інформації.

ВИСНОВКИ

У результаті виконання кваліфікаційної роботи спроектовано, програмно реалізовано та практично перевірено програмно-технічний засіб керування дистанційним металошукачем на базі мікроконтролера ESP32 . Проведений на початковому етапі аналіз предметної області підтвердив, що сучасні вимоги до систем виявлення металів потребують переходу від ізольованих аналогових приладів до інтелектуальних рішень, здатних дистанційно передавати дані.

Створений пристрій повністю задовольняє ці потреби, об'єднуючи в єдиному комплексі виміри, цифрову логіку обробки, допоміжні сенсори та бездротові мережеві технології. Апаратна реалізація розробленого пристрою базується на гнучкій модульній архітектурі, центральним ядром якої виступає мікроконтролер ESP32. Завдяки його високій обчислювальній потужності та інтегрованим інтерфейсам вдалося мінімізувати кількість додаткових компонентів і забезпечити високу стабільність роботи.

Важливим етапом стала розробка ефективного блоку узгодження та фільтрації аналогового сигналу. Це дозволило безпечно адаптувати імпульсний вихід металошукача для зчитування аналого-цифровим перетворювачем мікроконтролера, усунувши ризики перенапруги та суттєво знизивши вплив високочастотних завад. Крім того, спроектоване живлення забезпечило стабільну роботу чутливої аналогової частини і цифрових складових.

Одним із головних результатів роботи є успішна програмна реалізація методів цифрової обробки сигналу та системи автоматичної калібровки. Замість традиційного покладання на нестабільні миттєві значення, мікроконтролерний алгоритм використовує періодичне накопичення вибірок та адаптивне порогове порівняння. Це дозволило чітко розмежувати природні шуми стану спокою та реальне виявлення металевих об'єктів. Функція програмної автокалібровки повністю усунула потребу в ручному налаштуванні приладу, а реалізована дистанційна взаємодія через Wi-Fi підтвердила працездатність створеного експериментального зразка та повне досягнення мети кваліфікаційної роботи.

					КВРКІ.2301109.23.01.43 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		74

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. Hemalatha R. et al. Sentinel Rover: Cutting-Edge Wireless Mine Detection and Alert System for High-Risk Terrains. *2023 International Conference on Self Sustainable Artificial Intelligence Systems (ICSSAS)*. IEEE, 2023. P. 1632–1636. DOI: 10.1109/ICSSAS57918.2023.10331842.
2. Capineri L. et al. Enhancing Minefield Detection: A Robotic Approach for Multi-Sensor Detection and Discrimination of Small Low-Metal-Content Landmines. *IEEE Transactions on Geoscience and Remote Sensing*. 2025. Vol. 63. P. 1–11.
3. Hasan M., Muda N. R. S. Rancang Bangun Robot Ranjau Anti Tank Dengan Detektor VLF (Very Low Frequency) dan Pulse Induction (PI). *International Journal of IJNRSM*. 2024. Vol. 4, No. 2. P. 1–8.
4. Jat S. et al. Wi-Fi Based Remotely Controlled Robot with Android Application for Bomb and Landmine Detection. *PRATIBODH*. 2023. URL: <https://pratibodh.org/index.php/pratibodh/article/view/47> (дата звернення: 29.04.2026).
5. Hutsul T., Tkach V., Khobzei M. Humanitarian Demining: How Can UAVs and Internet of Things Help? *Security of Infocommunication Systems and Internet of Things*. 2023. Vol. 1, No. 2. P. 02004. DOI: 10.31861/sisiot2023.2.02004.
6. Чернявський М. А. Development of the Wireless Metal Detector X-Finder: Innovative Design, Advanced Signal Processing Methods, and Scientific Novelty for Archaeology and Demining. *International Electronic Scientific Journal Science Online*. 2024. URL: <http://nauka-online.com/> (дата звернення: 29.04.2026).
7. Wester H. *The Use of Metal Detectors in Schools* : master's thesis / Sam Houston State University. Huntsville, 2025. URL: <https://shsu-ir.tdl.org/items/60c438e3-34d5-44d0-819c-d3f90283a781> (дата звернення: 29.04.2026).
8. Wessman A., Oksanen E. Metal-Detecting Data as Citizen Science Archaeology. *Oodeja Mikalle – Odes to Mika: Festschrift for Professor Mika Lavento*

					КВРКІ.2301109.23.01.43 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		75

on the Occasion of His 60th Birthday. Archaeological Society of Finland, 2022. P. 293–302.

9. Xu X. et al. Progress in Remote Sensing of Heavy Metals in Water. *Remote Sensing*. 2024. Vol. 16, No. 20. P. 3888. DOI: 10.3390/rs16203888.

10. Lovynska V. et al. Monitoring Heavy Metals and Metalloids in Soils and Vegetation by Remote Sensing: A Review. *Remote Sensing*. 2024. Vol. 16, No. 17. P. 3221. DOI: 10.3390/rs16173221.

11. Shen D. et al. Application of Metal Detection Technology in Agricultural Machinery Equipment. *AgriEngineering*. 2026. Vol. 8, No. 1. P. 15. DOI: 10.3390/agriengineering8010015.

12. Zhang K., Shi Y., Karnouskos S., Sauter T., Fang H., Colombo A. W. Advancements in Industrial Cyber-Physical Systems: An Overview and Perspectives. *IEEE Transactions on Industrial Informatics*. 2023. Vol. 19, No. 1. P. 716–729. DOI: 10.1109/TII.2022.3199481.

13. Ibrahim S. et al. Design and Construction of an Effective Metal Detector. *Majlesi Journal of Mechatronic Systems*. 2023. Vol. 12, No. 1. P. 11–14. URL: <https://ms.majlesi.info/index.php/ms/article/view/552> (дата звернення: 29.04.2026).

14. Salah W. A., Shabaneh A. A. Development and Implementation of a Low-Cost Metal Detector Device. *Bulletin of Electrical Engineering and Informatics*. 2024. Vol. 13, No. 5. P. 3100–3111. DOI: 10.11591/eei.v13i5.7613.

15. Najafi E., Hasanzadeh S., Kheradmandan K. Induction Balance Metal Detector Using Multi-Level Chirp Signal. *2022 13th Power Electronics, Drive Systems, and Technologies Conference (PEDSTC)*. IEEE, 2022. P. 479–483. DOI: 10.1109/PEDSTC53976.2022.9767474.

16. Zhang H. et al. Model of a Novel PCB Coil for High-Sensitivity Metal Detector. *IEEE Access*. 2025. Vol. 13. P. 31603–31612. DOI: 10.1109/ACCESS.2025.3542528.

17. Skoropada T. *Exploring Metal Detector Signals: Inductive Power Transfer and the Dimensionality of Responses* : doctoral dissertation / University of Otago. Dunedin, 2024.

					КВРКІ.2301109.23.01.43 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		76

18. Marsh L. A., Fletcher A. D., Peyton A. J. Balancing of Resonant Differential Coils for Broadband Inductive Sensor Systems. *Sensors*. 2024. Vol. 24, No. 18. P. 6009. DOI: 10.3390/s24186009.

19. Wang Z. et al. Research on Metal Target Classification Method Based on Multi-Frequency Electromagnetic Induction. *Journal of Physics: Conference Series*. 2023. Vol. 2447, No. 1. P. 012002. DOI: 10.1088/1742-6596/2447/1/012002.

20. Minhas S. F. et al. Minimizing Soil Mineralization Effects in Pulse Induced Metal Detector Through Dual Adaptive Filter-Based Compensation to Detect Low Metal Contents. *2024 21st International Bhurban Conference on Applied Sciences and Technology (IBCAST)*. IEEE, 2024. P. 1–7.

21. Minhas S. F., Shah M. H. Adaptive Filtering Technique for Mitigating Soil Mineralization Effects in Pulse-Induced Metallic Mine Detectors. *Measurement*. 2026. Vol. 266. P. 120473. DOI: 10.1016/j.measurement.2026.120473.

22. Biloshchytskyi A. et al. Development of the Architecture of a Software and Hardware Complex for Data Collection and Management of Smart Technologies. *2025 IEEE 13th International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS)*. IEEE, 2025. P. 777–783.

23. Idris M. R., Irdyanti M. N., Ramlee M. A Review of Microcontroller Design in Future Industrial Revolution. *AIP Conference Proceedings*. 2024. Vol. 2750, No. 1. P. 050031. DOI: 10.1063/5.0149680.φφ

24. Moharkar K. A. et al. Review on Different Microcontroller Boards Used in IoT. *International Journal for Research in Applied Science and Engineering Technology*. 2022. Vol. 10, No. 1. P. 234–242. DOI: 10.22214/IJRASET.2022.39778.

25. Pandey H. K. et al. Development of a Versatile Wireless Control System Using nRF24L01 Transceiver and Arduino Unit. *2023 International Conference on Computational Intelligence for Information, Security and Communication Applications (CIISCA)*. IEEE, 2023. P. 103–108. DOI: 10.1109/CIISCA59740.2023.00030.

					КВРКІ.2301109.23.01.43 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		77

26. Didi Z., El Azami I. IoT, Comparative Study Between the Use of Arduino Uno, ESP32, and Raspberry Pi in Greenhouses. *International Conference on Digital Technologies and Applications*. Cham : Springer International Publishing, 2022. P. 718–726. DOI: 10.1007/978-3-031-02447-4_74.

27. Muntian O. *Comparative Analysis of Arduino, STM32 and ESP32 Platforms for Autonomous Sensor Systems* : кваліфікаційна робота / Харківський національний університет радіоелектроніки. Харків, 2026. URL: <https://openarchive.nure.ua/entities/publication/ae61d74f-7783-40b2-9176-64ea1010472a> (дата звернення: 29.04.2026).

28. Stanciu M. E. et al. Transmission Techniques in Data Acquisition Systems Using Arduino and ESP32. *2023 15th International Conference on Electronics, Computers and Artificial Intelligence (ECAI)*. IEEE, 2023. P. 1–6. DOI: 10.1109/ECAI58194.2023.10193921.

29. Belghachem N., Benbouzid M. A. Cost-Efficient Microcontroller-Based Laser Wireless Communication Prototypes for Internet of Things Applications. *Engineering Research Express*. 2025. Vol. 7, No. 2. P. 025346. DOI: 10.1088/2631-8695/add647.

30. Malele R. H., Mendu B., Monchusi B. B. Microcontroller-Driven Battery Management in Hybrid Energy Systems: A Systematic Review of Applications, Control Strategies, and Emerging Trends. *IEEE Access*. 2025. Vol. 13. P. 1–30.

31. Plaуска I., Liutkevičius A., Janavičiūtė A. Performance Evaluation of C/C++, MicroPython, Rust and TinyGo Programming Languages on ESP32 Microcontroller. *Electronics*. 2023. Vol. 12, No. 1. P. 143. DOI: 10.3390/electronics12010143.

32. Gupta E., Gupta V. A Comparative Analysis of ESP32 and ESP8266 for AI-Powered Applications. *2025 International Conference on Next Generation Communication & Information Processing (INCIP)*. IEEE, 2025. P. 992–996. DOI: 10.1109/INCIP64058.2025.11019649.

					КВРКІ.2301109.23.01.43 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		78

33. Espinosa-Gavira M. J. et al. Characterization and Performance Evaluation of ESP32 for Real-Time Synchronized Sensor Networks. *Procedia Computer Science*. 2024. Vol. 237. P. 261–268. DOI: 10.1016/j.procs.2024.05.104.

34. Hercog D., Lerher T., Truntič M., Težak O. Design and Implementation of ESP32-Based IoT Devices. *Sensors*. 2023. Vol. 23, No. 15. P. 6739. DOI: 10.3390/s23156739.

35. Chikurtev D., Karkov B., Blagoeva E. An Overview of ESP32 Microcontroller Application in Embedded Systems, Robotics and Internet of Things. *International Conference on Mathematical Modeling in Physical Sciences*. Cham : Springer Nature Switzerland, 2024. P. 753–766.

36. Bhaganagare S. et al. The Vocal Cortex of the Smart Home: A Review of ESP32-Based Control and Automation Architectures. *International Journal of Advanced Research in Science, Communication and Technology*. 2025. Vol. 5, No. 4. P. 1071–1094.

37. Gavrilov A. et al. Using IoT Protocols in Real-Time Systems: Protocol Analysis and Evaluation of Data Transmission Characteristics. *Journal of Computer Networks and Communications*. 2022. Vol. 2022. P. 7368691. DOI: 10.1155/2022/7368691.

38. Lech P., Marciniak B., Okarma K. A Low-Cost Energy-Efficient IoT Camera Trap Network for Remote Forest Surveillance. *Electronics*. 2025. Vol. 14, No. 21. P. 4266. DOI: 10.3390/electronics14214266.

39. Vaz F. A. J. et al. Performance Analysis of nRF24L01 Wireless Module. *2024 5th International Conference on Data Intelligence and Cognitive Informatics (ICDICI)*. IEEE, 2024. P. 216–222. DOI: 10.1109/ICDICI62993.2024.10810938.

40. Segbaya M. et al. Design and Construction of Remote-Controlled Vehicle. *International IOT, Electronics and Mechatronics Conference*. Singapore : Springer Nature Singapore, 2025. P. 403–412.

41. Debadarshini J. et al. TimeCast: Real-Time Many-to-Many Data-Sharing in Low-Power Wireless Distributed Systems. *IEEE Systems Journal*. 2023. Vol. 17, No. 4. P. 5726–5737. DOI: 10.1109/JSYST.2023.3292892.

					КВРКІ.2301109.23.01.43 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		79

42. Baker B. et al. A Survey of Short-Range Wireless Communication for Ultra-Low-Power Embedded Systems. *Journal of Low Power Electronics and Applications*. 2024. Vol. 14, No. 2. P. 27. DOI: 10.3390/jlpea14020027.

43. Cameron N. ESP32 Formats and Communication. *Apress eBooks*, 2023. DOI: 10.1007/978-1-4842-9376-8 (дата звернення: 29.04.2026).

44. García-Rivada P. et al. Bluetooth Protocol for Opportunistic Sensor Data Collection on IoT Telemetry Applications. *Electronics*. 2025. Vol. 14, No. 16. P. 3281. DOI: 10.3390/electronics14163281.

45. Oliveira C. E. A. et al. Evaluation of Data Transmission Performance of Low-Cost Embedded System for Real-Time Monitoring of the Thermal Environment in Animal Facilities. *3rd US Precision Livestock Farming Conference (USPLF2025)*. USPLF2025 Proceedings Committee, 2025. P. 104–105.

46. Chen C. Y. et al. Web-Based Internet of Things on Environmental and Lighting Control and Monitoring System Using Node-RED, MQTT and Modbus Communications within Embedded Linux Platform. *Internet of Things*. 2024. Vol. 27. P. 101305. DOI: 10.1016/j.iot.2024.101305.

47. Lagartinho-Oliveira C. et al. Integrated Development of Embedded Systems with Remote Graphical User Interfaces. *IEEE Access*. 2023. Vol. 11. P. 63795–63807.

48. Morchid A. et al. IoT-Based Smart Irrigation Management System to Enhance Agricultural Water Security Using Embedded Systems, Telemetry Data, and Cloud Computing. *Results in Engineering*. 2024. Vol. 23. P. 102829. DOI: 10.1016/j.rineng.2024.102829.

49. Lin M. H. et al. Node-RED Web-Based Monitor and Control of Power System Using Modbus and Message Queuing Telemetry Transport Communication in Raspberry Pi Embedded Platform. *Sensors & Materials*. 2024. Vol. 36.

50. Ranganeni V. et al. AccessTeleopKit: A Toolkit for Creating Accessible Web-Based Interfaces for Tele-Operating an Assistive Robot. *Proceedings of the 37th Annual ACM Symposium on User Interface Software and Technology*. New York : Association for Computing Machinery, 2024. P. 1–12. DOI: 10.1145/3654777.3676355.

					КВРКІ.2301109.23.01.43 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		80

51. Металошукач Garrett ACE 250. *Rozetka*. [б. р.]. URL: https://rozetka.com.ua/ua/garrett_ace_250/p101793 (дата звернення: 29.04.2026).

52. Металодетектор CEIA CMD. *Kamrad*. [б. р.]. URL: <https://kamrad.ua/ua/ceia-cmd> (дата звернення: 29.04.2026).

53. Металошукач Пірат (плата зібрана з 2 резисторами). *Webtronic*. [б. р.]. URL: <https://webtronic.com.ua/metalloiskatel-pirat-plata-sobrannaya-s-2-rezistorami> (дата звернення: 29.04.2026).

54. Плата мікроконтролера Arduino Uno R3. *Mehanika*. [б. р.]. URL: <https://mehanika.net.ua/product/plata-mikrokontrolera-arduino-uno-r3-atmega328p-ch340g-usb-type-b/> (дата звернення: 29.04.2026).

55. Бездротовий модуль трансивер NRF24L01+ 2.4 ГГц з антеною. *Ledplus*. [б. р.]. URL: <https://ledplus.com.ua/ua/p1121273442-besprovodnoj-modul-transiver.html> (дата звернення: 29.04.2026).

56. HC-05 Wireless Bluetooth RF Transceiver Module Serial RS232 TTL. *RCS Components*. [б. р.]. URL: https://www.rcscomponents.kiev.ua/product/hc-05-wireless-bluetooth-rf-transceiver-module-serial-rs232-ttl_84538.html (дата звернення: 29.04.2026).

57. ESP8266 WiFi Module NodeMCU v3. *Botland*. [б. р.]. URL: <https://botland.store/esp8266-wifi-modules/8241-esp8266-wifi-module-nodemcu-v3-5904422300630.html> (дата звернення: 29.04.2026).

58. IoT ESP-32S 2.4GHz Dual-Mode WiFi + Bluetooth. *LaskaKit*. [б. р.]. URL: <https://www.laskakit.cz/en/iot-esp-32s-2-4ghz-dual-mode-wifi-bluetooth-rev-1--sr2102/> (дата звернення: 29.04.2026).

59. Розпіновка ESP32 (ESP32 Pinout Reference). *uPesy*. [б. р.]. URL: <https://www.upesy.com/blogs/tutorials/esp32-pinout-reference-gpio-pins-ultimate-guide> (дата звернення: 29.04.2026).

					КВРКІ.2301109.23.01.43 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		81

ДОДАТОК Г

(обов'язковий)

Перелік елементів до схеми електричної принципової

Поз. позн.	Найменування	Кількість	Примітка					
Bat1	Акумулятор 18650	3						
Датчики								
B1	Датчик ультразвуковий HC-SR04	1						
B2	Датчик температури та вологості DHT11	1						
B3	Датчик вологості ґрунту (на базі LM393)	1						
B4	Модуль GPS GY-NEO5MV2	1						
Конденсатори								
C1	Конденсатор K50-35-2200 мкФ-16В	1						
C2, C3, C9–C12, C14	Конденсатор K10-176-0,1 мкФ ± 20%	7						
C4, C5	Конденсатор K50-35-10 мкФ-25В	2						
C6	Конденсатор K10-176-0,001 мкФ ± 20%	1						
C7, C8, C13	Конденсатор K50-35-1 мкФ-25В	3						
D1–D3	Діод 1N4148	3						
HG1	Дисплей OLED SSD1306 (128x64)	1						
Резистори								
R1	Резистор МЛТ-0,125-120 кОм ± 5%	1						
R2	Резистор МЛТ-0,125-47 кОм ± 5%	1						
R3, R4	Резистор МЛТ-0,125-62 кОм ± 5%	2						
КВРКІ.2301109.23.01.43 ПЕЗ								
Зм.	Арк.	№докум.	Підпис	Дата	Програмно-технічний засіб керування дистанційним металошукачем на основі мікроконтролера ESP32 Перелік елементів	Літера	Аркуш	Аркуші
Виконав	Колесніков Віталій					v	1	3
Перевір.	Сергій ЛИСЕНКО							
Н.контр.	Сергій ЛИСЕНКО							
Затвер.	Ольга ПАВЛОВА							
						ХНУ КІ2с-23-1		

Поз. позн.	Найменування	Кількість	Примітка
R5	Резистор МЛТ-0,125-100 кОм ± 5%	1	
R6	Резистор МЛТ-0,125-1 МОм ± 5%	1	
R7	Резистор МЛТ-0,125-1,5 кОм ± 5%	1	
R10	Резистор МЛТ-0,125-1 кОм ± 5%	1	
R11	Резистор МЛТ-0,125-470 Ом ± 5%	1	
R12	Резистор МЛТ-0,125-47 Ом ± 5%	1	
R13	Резистор МЛТ-0,125-150 Ом ± 5%	1	
R14	Резистор МЛТ-0,125-470 кОм ± 5%	1	
R15	Резистор МЛТ-0,125-10 Ом ± 5%	1	
R16, R17	Резистор МЛТ-2,0-390 Ом ± 5%	2	
R18–R24	Резистор МЛТ-0,125-10 кОм ± 5%	7	
R25	Резистор МЛТ-0,125-20 кОм ± 5%	1	
R26	Резистор МЛТ-0,125-12 кОм ± 5%	1	
R27	Резистор МЛТ-0,125-4,7 кОм ± 5%	1	
R28	Резистор МЛТ-0,125-33 кОм ± 5%	1	
Перемикачі			
S1-S4	Кнопка тактова TS-A62R	4	
S5	Перемикач двох		
Мікросхеми та модулі			
U1	Мікросхема NE555P	1	
U2	Мікросхема TL082CP	1	
U3	Мікросхема TPL0501-100RDCKR	1	
U4	Контролер ESP32-WROOM-32	1	
U5	Модуль BMS 3S 2A Booster	1	
U6	Модуль понижуючий XL4015	1	
КВРКІ.2301109.23.01.43 ПЕЗ			Арк.
Зм.	Арк.	№ докум.	Підпис
			Дата
			2

ДОДАТОК Д (обов'язковий)

Графічний інтерфейс користувача веб-застосунку



Рисунок Д.1 – Веб-інтерфейс системи дистанційного керування у режимі точки доступу

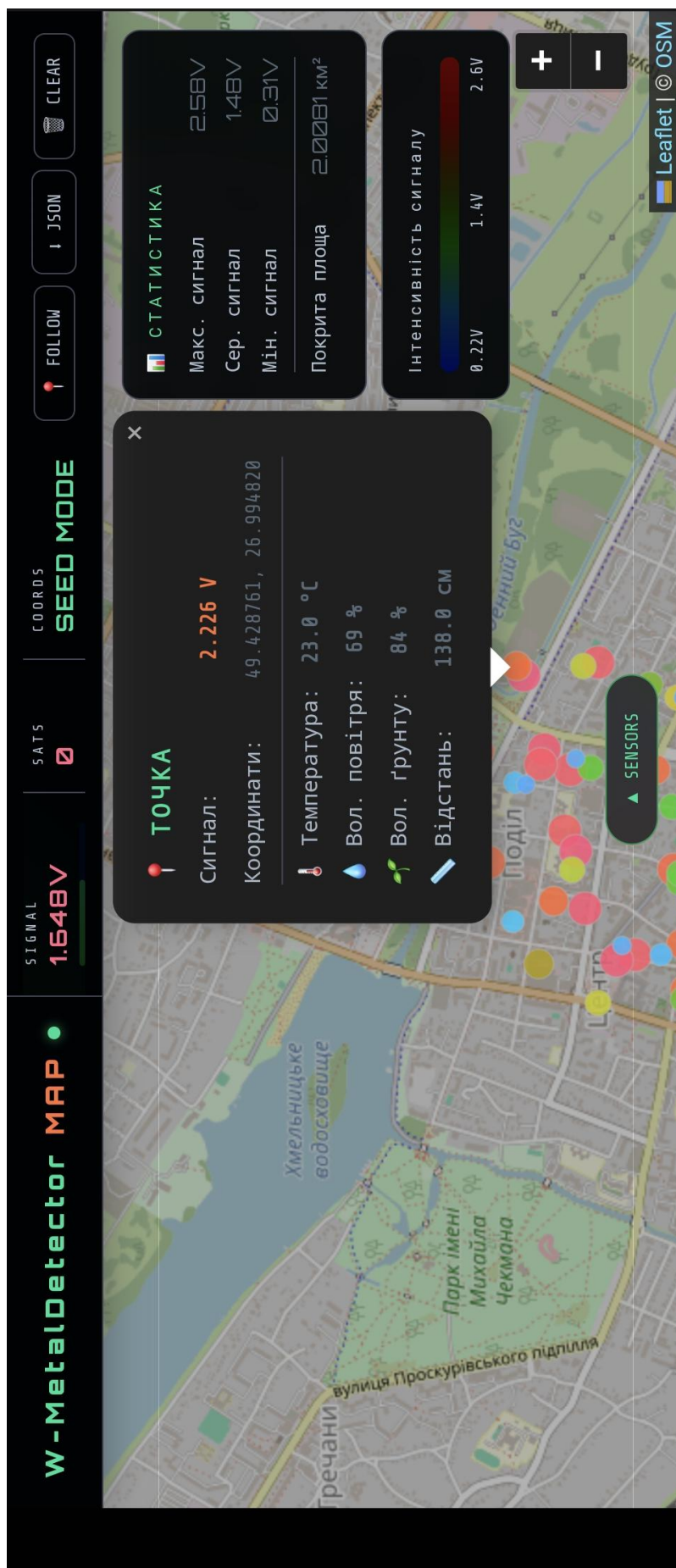


Рисунок Д.2 – Відображення координат вимірювань на карті

powered by mapbox [Sign up for Mapbox](#) [Table](#) [? Help](#)

Search

```

1852 26.98875,
1853 49.423271
1854 ]
1855 },
1856 "properties": {
1857   "intensity": 0.139,
1858   "voltage": 0.551,
1859   "temp": 25.1,
1860   "humidity": 75.4,
1861   "soil": 43,
1862   "dist": 114,
1863   "index": 97
1864 }
1865 },
1866 {
1867   "type": "Feature",
1868   "geometry": {
1869     "type": "Point",
1870     "coordinates": [
1871       26.99296,
1872       49.416901
1873     ]
1874   },
1875   "properties": {
1876     "intensity": 0.05,
1877     "voltage": 0.339,
1878     "temp": 29.8,
1879     "humidity": 68.2,
1880     "soil": 76,
1881     "dist": 69,
1882     "index": 98
1883   }
1884 }
1885 ]
1886 }

```

© Mapbox © OpenStreetMap Improve this map

Рисунок Д.3 – Приклад сформованих даних у форматі JSON

ДОДАТОК Е

(обов'язковий)

Текст системного програмного забезпечення

```
#include <WiFi.h>
#include <WebServer.h>
#include <SPI.h>
#include <Preferences.h>
#include <Wire.h>
#include <Adafruit_GFX.h>
#include <Adafruit_SSD1306.h>
#include <HardwareSerial.h>
#include <TinyGPSPlus.h>
#include <DHT.h>

#define PIN_CS          5
#define PIN_SCLK        18
#define PIN_MOSI        23
#define PIN_SIGNAL      34
#define PIN_SDA          21
#define PIN_SCL          22
#define PIN_GPS_RX      16
#define PIN_GPS_TX      17

#define PIN_BTN_UP      15
#define PIN_BTN_DOWN    13
#define PIN_BTN_OK      19
#define PIN_BTN_MENU    0 // WARNING: boot
pin - do not hold during power-on!

#define PIN_DHT          25
#define PIN_SOIL_ANA    35
#define PIN_SOIL_PWR    27
#define PIN_SOIL_BTN    26
#define PIN_TRIG        32
#define PIN_ECHO        33 // WARNING: voltage
divider 5V→3.3V required!

enum OledScreen {
  SCR_MAIN,
  SCR_MENU,
  SCR_MENU_WIFI,
  SCR_MENU_GPS,
  SCR_MENU_SOIL,
  SCR_MENU_CAL,
};
OledScreen currentScreen = SCR_MAIN;
int menuCursor = 0;

#define MENU_ITEMS 4

bool useSeedData = false;

#define DHT_TYPE DHT11

const char* AP_SSID = "W-MetalDetector";
const char* AP_PASSWORD = "password123";
const char* STA_SSID = "S25";
const char* STA_PASSWORD = "1234567890987654321";

#define SCREEN_WIDTH 128
#define SCREEN_HEIGHT 64
Adafruit_SSD1306 display(SCREEN_WIDTH,
SCREEN_HEIGHT, &Wire, -1);

HardwareSerial gpsSerial(2);
TinyGPSPlus gps;

DHT dht(PIN_DHT, DHT_TYPE);

struct SensorData {
  float temperature;
  float humidity;
  int soilRaw;
  int soilPct;
};
```

```

float distanceCm;
unsigned long lastDhtMs;
unsigned long lastSoilMs;
unsigned long lastSonicMs;
};
SensorData sens = { NAN, NAN, 0, 0, -1.0f, 0,
0, 0 };

#define MAX_POINTS 300
struct HeatPoint {
float lat, lon;
float intensity;
float voltage;
float temp;
float hum;
int soil;
float dist;
};
HeatPoint gpsPoints[MAX_POINTS];
int pointCount = 0;

const float TARGET_VOLTAGE = 0.22f;
const int POT_MAX = 255;
const int POT_RESET_DEFAULT = 128;
const float SIGNAL_MAX_V = 2.6f;
const float SIGNAL_MIN_V = 0.22f;

const int AVG_SAMPLES = 32;
const int AVG_DELAY_US = 100;
const int PEAK_WINDOW_MS = 40;
const int DATA_THROTTLE_MS = 200;
const int SETTLE_MS = 280;
const int STEP_COARSE = 20;
const int STEP_FINE = 1;

WebServer server(80);
Preferences prefs;

uint8_t currentLevel = 128;
float lastVoltage = 0.0f;
unsigned long lastVoltageMs = 0;

unsigned long lastNvsWriteMs = 0;
bool needNvsWrite = false;
bool isSTAMode = false;
bool staConnected = false;
bool oledOk = false;

bool pendingWiFiSwitch = false;
bool pendingWiFiApMode = true;

struct Button {
uint8_t pin;
bool lastState;
unsigned long lastDebounceMs;
};
bool btnPressed(Button &b);
void startWiFi(bool apMode);
void generateSeedPoints();
void updateOLED();

enum CalState { CAL_IDLE, CAL_COARSE,
CAL_FINE, CAL_FINISHING, CAL_ERROR };
enum CalMode { MODE_STANDARD, MODE_WET,
MODE_SENSITIVE };

CalState calState = CAL_IDLE;
CalMode calMode = MODE_STANDARD;
int calProgress = 0;
String calStatus = "";
unsigned long settleUntil = 0;
int scanLevel = 0;
int quietLevel = 0;

bool isCalibrating() { return calState !=
CAL_IDLE && calState != CAL_ERROR; }

void writePot(uint8_t value) {
digitalWrite(PIN_CS, LOW);
SPI.transfer(value);
digitalWrite(PIN_CS, HIGH);
currentLevel = value;
}

```

```

float readAvg() {
    long sum = 0;
    for (int i = 0; i < AVG_SAMPLES; i++) {
        sum += analogRead(PIN_SIGNAL);
        delayMicroseconds(AVG_DELAY_US);
        if (i % 16 == 0) server.handleClient();
    }
    return (sum / (float)AVG_SAMPLES) * (3.3f /
4095.0f);
}

float readVoltageCached() {
    unsigned long now = millis();
    if (now - lastVoltageMs >= DATA_THROTTLE_MS)
{
        long sum = 0;
        for (int i = 0; i < 40; i++) { sum +=
analogRead(PIN_SIGNAL);
delayMicroseconds(100); }
        lastVoltage = (sum / 40.0f) * (3.3f /
4095.0f);
        lastVoltageMs = now;
    }
    return lastVoltage;
}

void scheduleNvsWrite() {
    needNvsWrite = true;
    lastNvsWriteMs = millis();
}

void flushNvsIfNeeded() {
    if (needNvsWrite && millis() -
lastNvsWriteMs > 2000) {
        prefs.putUChar("sens", currentLevel);
        needNvsWrite = false;
        Serial.println("[NVS] Saved level=" +
String(currentLevel));
    }
}

void readDHT() {
    if (millis() - sens.lastDhtMs < 1500)
return;
    sens.lastDhtMs = millis();

    float h = dht.readHumidity();
    float t = dht.readTemperature();

    if (isnan(h) || isnan(t)) {
        Serial.println("[DHT] Read failed!");
    } else {
        sens.humidity = h;
        sens.temperature = t;
        Serial.printf("[DHT]           T=%.1f°C
H=%.1f%%\n", t, h);
    }
}

void readSoilOnButton() {
    static bool lastBtnState = HIGH;
    static unsigned long lastDebounceMs = 0;

    bool btnState = digitalRead(PIN_SOIL_BTN);

    if (btnState == LOW && lastBtnState == HIGH)
{
        unsigned long now = millis();
        if (now - lastDebounceMs < 300) {
            lastBtnState = btnState;
            return;
        }
        lastDebounceMs = now;

        digitalWrite(PIN_SOIL_PWR, HIGH);
        delay(100);

        long sum = 0;
        for (int i = 0; i < 8; i++) {
            sum += analogRead(PIN_SOIL_ANA);
            delayMicroseconds(500);

```

```

    }
    digitalWrite(PIN_SOIL_PWR, LOW);

    sens.soilRaw = sum / 8;
    const int SOIL_DRY_VAL = 2800;
    const int SOIL_WET_VAL = 800;
    sens.soilPct = constrain(
        map(sens.soilRaw,          SOIL_DRY_VAL,
SOIL_WET_VAL, 0, 100),
        0, 100
    );
    sens.lastSoilMs = millis();

    Serial.printf("[SOIL] btn: raw=%d →
%d%\n", sens.soilRaw, sens.soilPct);
}

lastBtnState = btnState;
}

float singlePing() {
    digitalWrite(PIN_TRIG, LOW);
    delayMicroseconds(4);

    digitalWrite(PIN_TRIG, HIGH);
    delayMicroseconds(10);
    digitalWrite(PIN_TRIG, LOW);

    long duration = pulseIn(PIN_ECHO, HIGH,
24000);

    if (duration == 0) return -1.0f;

    float cm = duration * 0.01715f;

    if (cm < 2.0f || cm > 300.0f) return -1.0f;
    return cm;
}

float readUltrasonic() {
    float a = singlePing(); delay(10);
    float b = singlePing(); delay(10);
    float c = singlePing();

    int valid = (a > 0) + (b > 0) + (c > 0);
    if (valid == 0) return -1.0f;

    if (valid == 1) return (a > 0) ? a : (b > 0)
? b : c;

    if (a < 0) a = 9999; if (b < 0) b = 9999; if
(c < 0) c = 9999;
    float mn = min(a, min(b, c));
    float mx = max(a, max(b, c));
    float med = a + b + c - mn - mx;
    return (med >= 9999) ? -1.0f : med;
}

void pollUltrasonic() {
    if (millis() - sens.lastSonicMs < 350)
return;
    sens.lastSonicMs = millis();
    sens.distanceCm = readUltrasonic();
}

float normalizeSignal(float v) {
    return constrain((v - SIGNAL_MIN_V) /
(SIGNAL_MAX_V - SIGNAL_MIN_V), 0.0f, 1.0f);
}

void addGpsPoint(float lat, float lon, float
rawV) {
    if (pointCount >= MAX_POINTS) {
        memmove(gpsPoints, gpsPoints + 1,
(MAX_POINTS - 1) * sizeof(HeatPoint));
        pointCount = MAX_POINTS - 1;
    }
    gpsPoints[pointCount++] = {
        lat, lon,
        normalizeSignal(rawV),
        rawV,
        sens.temperature,

```

```

    sens.humidity,
    sens.soilPct,
    sens.distanceCm
};
}

void generateSeedPoints() {
    float baseLat = 49.4220f;
    float baseLon = 26.9870f;
    randomSeed(42);
    for (int i = 0; i < 99; i++) {
        float lat = baseLat + ((float)random(-
800, 800) / 100000.0f);
        float lon = baseLon + ((float)random(-
800, 800) / 100000.0f);
        float rawV;
        if (i < 33) rawV = 2.2f +
((float)random(0,1000)/1000.0f) * 0.4f;
        else if (i < 66) rawV = 1.3f +
((float)random(0,1000)/1000.0f) * 0.5f;
        else rawV = 0.3f +
((float)random(0,1000)/1000.0f) * 0.4f;

        if (pointCount >= MAX_POINTS) break;
        gpsPoints[pointCount++] = {
            lat, lon,
            normalizeSignal(rawV),
            rawV,
            20.0f + ((float)random(0,100)/10.0f),
            50.0f + ((float)random(0,300)/10.0f),
            random(10, 90),
            (float)random(10, 150)
        };
    }
    Serial.println("[SEED] Generated " +
String(pointCount) + " test points");
}

void calFinish(int finalLevel) {
    finalLevel = constrain(finalLevel, 0,
POT_MAX);
    writePot((uint8_t)finalLevel);
    calState = CAL_FINISHING;
    settleUntil = millis() + 150;
}

void calTick() {
    unsigned long now = millis();
    if (now < settleUntil) return;

    float sig = readAvg();
    now = millis();
    lastVoltage = sig; lastVoltageMs = now;
    bool isQuiet = (sig < TARGET_VOLTAGE);

    switch (calState) {
        case CAL_COARSE: {
            calProgress = map(scanLevel, 0, POT_MAX,
2, 55);
            calStatus = "Coarse scan: lvl=" +
String(scanLevel) + " sig=" + String(sig, 3)
+ "V";
            if (!isQuiet) {
                int next = scanLevel + STEP_COARSE;
                if (next > POT_MAX) {
                    calState = CAL_ERROR;
                    calStatus = "Error: no silence. Lift
coil!";
                    calProgress = 100;
                } else {
                    scanLevel = next;
                    writePot((uint8_t)scanLevel);
                    settleUntil = now + SETTLE_MS;
                }
            } else {
                int fineStart = max(0, scanLevel -
STEP_COARSE);
                scanLevel = fineStart;
                writePot((uint8_t)scanLevel);
                settleUntil = now + SETTLE_MS + 150;
                calState = CAL_FINE;
                calProgress = 55;
            }
        }
    }
}

```

```

        calStatus      = "Fine scan from " +
String(fineStart);
    }
    break;
}
case CAL_FINE: {
    calProgress      = constrain(55 +
map(scanLevel, quietLevel, quietLevel +
STEP_COARSE, 0, 40), 55, 95);
    calStatus      = "Fine scan: lvl=" +
String(scanLevel) + " sig=" + String(sig, 3)
+ "V";
    if (!isQuiet) {
        int next = scanLevel + STEP_FINE;
        if (next > POT_MAX) {
            calState = CAL_ERROR;
            calStatus = "Error: boundary not
found";
            calProgress = 100;
        } else {
            scanLevel = next;
            writePot((uint8_t)scanLevel);
            settleUntil = now + SETTLE_MS;
        }
    } else {
        int offset = (calMode == MODE_WET) ?
-5 : (calMode == MODE_SENSITIVE) ? 1 : -4;
        int finalLevel = constrain(scanLevel
- offset, 0, POT_MAX);
        calFinish(finalLevel);
    }
    break;
}
case CAL_FINISHING: {
    float fv = readAvg();
    lastVoltage = fv;
    lastVoltageMs = millis();
    calProgress = 100;
    calStatus      = "Done! Level: " +
String(currentLevel) + " sig=" + String(fv, 3)
+ "V";
    calState      = CAL_IDLE;
    scheduleNvsWrite();
    break;
}
case CAL_ERROR:
    calState = CAL_IDLE;
    break;
default: break;
}
}

void startCalibration(CalMode mode) {
    if (calState != CAL_IDLE) return;
    calMode      = mode;
    calState      = CAL_COARSE;
    calProgress = 0;
    quietLevel = 0;
    scanLevel = 0;
    writePot(0);
    settleUntil = millis() + 600;
    calStatus      = "Start: lvl 0, seeking
silence...";
    Serial.println("[CAL] Start mode=" +
String((int)mode));
}

void drawCheckbox(int x, int y, bool checked)
{
    if (checked) {
        display.fillRect(x, y, 7, 7,
SSD1306_WHITE);
    } else {
        display.drawRect(x, y, 7, 7,
SSD1306_WHITE);
    }
}

void drawScreenMain() {
    float v = readVoltageCached();
    display.setTextSize(1);

```

```

display.setCursor(0, 0);
if (isSTAMode) {
    display.print("STA ");
    display.print(staConnected
WiFi.localIP().toString() : "connecting..");
} else {
    display.print("AP 192.168.4.1");
}

display.drawLine(0, 10, 127, 10,
SSD1306_WHITE);

display.setTextSize(2);
display.setCursor(0, 13);
display.print(v, 3); display.print("V");

int bh = constrain((int)(v / 3.3f * 16), 0,
16);
display.drawRect(116, 13, 10, 16,
SSD1306_WHITE);
display.fillRect(116, 13 + (16 - bh), 10,
bh, SSD1306_WHITE);

display.setTextSize(1);

display.setCursor(0, 31);
display.print("T:");
display.print(isnan(sens.temperature) ? "--
" : String(sens.temperature, 0));
display.print("C H:");
display.print(isnan(sens.humidity) ? "--" :
String(sens.humidity, 0));
display.print("%");

display.setCursor(0, 40);
display.print("S:");
display.print(sens.soilPct);
display.print("%");
display.print(" D:");
if (sens.distanceCm < 0) display.print("---
");

else { display.print((int)sens.distanceCm);
display.print("cm"); }

display.setCursor(0, 49);
if (isSTAMode) {
    display.print(gps.location.isValid()
"GPS:FIX " : "GPS:--- ");
    display.print(gps.satellites.value());
display.print("sat ");
    display.print("Pt:");
display.print(pointCount);
} else {
    display.print(isCalibrating()
calStatus.substring(0, 21) : "IDLE Pt:");
    if (!isCalibrating())
display.print(pointCount);
}

display.setCursor(0, 57);
display.print("<]menu");

static const char* menuLabels[] = {
    "WiFi mode",
    "GPS / SEED",
    "Soil moisture",
    "Calibration"
};

void drawScreenMenu() {
    display.setTextSize(1);

    display.setCursor(0, 0);
    display.print("=== MENU ===");
    display.drawLine(0, 9, 127, 9,
SSD1306_WHITE);

    for (int i = 0; i < MENU_ITEMS; i++) {
        int y = 11 + i * 12;
        if (i == menuCursor) {

```

```

        display.fillRect(0, y - 1, 128, 10,
SSD1306_WHITE);
        display.setTextColor(SSD1306_BLACK);
    } else {
        display.setTextColor(SSD1306_WHITE);
    }
    display.setCursor(4, y);
    display.print(menuLabels[i]);
}
display.setTextColor(SSD1306_WHITE);
display.setCursor(0, 56);
display.print("[<]back [^][v][ok]");
}

void drawScreenWifi() {
    display.setTextSize(1);
    display.setCursor(0, 0);
    display.println("WiFi mode");
    display.drawLine(0, 10, 127, 10,
SSD1306_WHITE);

    display.setCursor(0, 16);
    display.print(menuCursor == 0 ? ">" : " ");
    drawCheckbox(10, 16, !isSTAMode);
    display.setCursor(20, 16);
    display.print("AP (detector)");

    display.setCursor(0, 28);
    display.print(menuCursor == 1 ? ">" : " ");
    drawCheckbox(10, 28, isSTAMode);
    display.setCursor(20, 28);
    display.print("STA (GPS map)");

    display.setCursor(0, 42);
    display.print("Now: ");
    display.print(isSTAMode ? "STA" : "AP");
    display.print(" ->: ");
    display.print(menuCursor == 0 ? "AP" :
"STA");

    display.setCursor(0, 57);

        display.print("[^][v]=move [ok]=set");
    }

void drawScreenGps() {
    display.setTextSize(1);
    display.setCursor(0, 0);
    display.println("GPS / SEED mode");
    display.drawLine(0, 10, 127, 10,
SSD1306_WHITE);

    display.setCursor(0, 16);
    display.print(menuCursor == 0 ? ">" : " ");
    drawCheckbox(10, 16, !useSeedData);
    display.setCursor(20, 16);
    display.print("GPS real");

    display.setCursor(0, 28);
    display.print(menuCursor == 1 ? ">" : " ");
    drawCheckbox(10, 28, useSeedData);
    display.setCursor(20, 28);
    display.print("SEED test");

    display.setCursor(0, 42);
    display.print("Active: ");
    display.print(useSeedData ? "SEED" : "GPS");

    display.setCursor(0, 57);
    display.print("[^][v]=move [ok]=set");
}

void drawScreenSoil() {
    display.setTextSize(1);
    display.setCursor(0, 0);
    display.println("Soil moisture");
    display.drawLine(0, 10, 127, 10,
SSD1306_WHITE);

    display.setCursor(0, 14);
    display.print("Last: ");
    display.print(sens.soilPct);
    display.println("%");
}

```

```

    display.print("Raw:                ");
display.println(sens.soilRaw);

    int barW = map(sens.soilPct, 0, 100, 0,
120);
    display.drawRect(0,    36,    120,    8,
SSD1306_WHITE);
    display.fillRect(0,    36,    barW,    8,
SSD1306_WHITE);

    display.setCursor(0, 48);
    display.print(sens.soilPct < 30 ? "Dry" :
sens.soilPct < 60 ? "Normal" : "Wet");

    display.setCursor(0, 57);
    display.print("[ok]=measure [<]back");
}

void drawScreenCal() {
    display.setTextSize(1);
    display.setCursor(0, 0);
    display.println("Calibration");
    display.drawLine(0,    10,    127,    10,
SSD1306_WHITE);

    if (calState == CAL_IDLE && calProgress ==
100) {
        display.setCursor(0, 16);
        display.println("*** DONE! ***");
        display.setCursor(0, 28);
        display.print("Level:                ");
display.println(currentLevel);
        display.setCursor(0, 38);
        display.print("Sig:                ");
display.print(readVoltageCached(),    3);
display.println("V");
        display.setCursor(0, 50);
        display.print("Saved to NVS");
        display.setCursor(0, 56);
        display.print("[<]back [ok]=again");
    } else if (calState == CAL_IDLE) {
        display.setCursor(0, 14);
        display.print(menuCursor == 0 ? ">" : "
"); display.println(" Standard");
        display.setCursor(0, 24);
        display.print(menuCursor == 1 ? ">" : "
"); display.println(" Wet soil");
        display.setCursor(0, 34);
        display.print(menuCursor == 2 ? ">" : "
"); display.println(" Sensitive");
        display.setCursor(0, 46);
        display.print("Lvl:");
display.print(currentLevel);
        display.print("                Sig:");
display.print(readVoltageCached(),    2);
display.print("V");
        display.setCursor(0, 56);
        display.print("[^][v]=mode [ok]=run");
    } else if (calState == CAL_ERROR) {
        display.setCursor(0, 16);
        display.println("*** ERROR ***");
        display.setCursor(0, 28);
        display.println(calStatus.substring(0,
21));
        display.setCursor(0, 56);
        display.print("[ok]=retry [<]back");
    } else {
        display.setCursor(0, 14);
        display.println(calStatus.substring(0,
21));
        int barW = map(calProgress, 0, 100, 0,
120);
        display.drawRect(0,    28,    120,    8,
SSD1306_WHITE);
        display.fillRect(0,    28,    barW,    8,
SSD1306_WHITE);
        display.setCursor(0, 40);

```

```

    display.print(calProgress);
display.print("%");
    display.setCursor(0, 52);
    display.print("Lvl:");
display.print(scanLevel);
    display.print("          Sig:");
display.print(readVoltageCached(), 2);
display.print("V");
}
}

void updateOLED() {
    if (!oledOk) return;
    display.clearDisplay();
    display.setTextColor(SSD1306_WHITE);

    switch (currentScreen) {
        case SCR_MAIN:      drawScreenMain();
break;
        case SCR_MENU:      drawScreenMenu();
break;
        case SCR_MENU_WIFI: drawScreenWifi();
break;
        case SCR_MENU_GPS:  drawScreenGps();
break;
        case SCR_MENU_SOIL: drawScreenSoil();
break;
        case SCR_MENU_CAL:  drawScreenCal();
break;
    }

    display.display();
}

Button btnUp   = { PIN_BTN_UP,   HIGH, 0 };
Button btnDown = { PIN_BTN_DOWN, HIGH, 0 };
Button btnOk   = { PIN_BTN_OK,   HIGH, 0 };
Button btnMenu = { PIN_BTN_MENU, HIGH, 0 };

#define DEBOUNCE_MS 50

bool btnPressed(Button &b) {
    bool state = digitalRead(b.pin);
    bool pressed = false;

    if (state == LOW && b.lastState == HIGH) {
        if (millis() - b.lastDebounceMs >
DEBOUNCE_MS) {
            b.lastDebounceMs = millis();
            pressed = true;
            Serial.printf("[BTN] pin=%d pressed\n",
b.pin);
        }
    }

    b.lastState = state;
    return pressed;
}

void handleButtons() {
    bool up   = btnPressed(btnUp);
    bool down = btnPressed(btnDown);
    bool ok   = btnPressed(btnOk);
    bool menu = btnPressed(btnMenu);

    switch (currentScreen) {

        case SCR_MAIN:
            if (menu) { currentScreen = SCR_MENU;
menuCursor = 0; updateOLED(); }
            break;

        case SCR_MENU:
            if (menu) { currentScreen = SCR_MAIN;
updateOLED(); }
            if (up)   { menuCursor = (menuCursor -
1 + MENU_ITEMS) % MENU_ITEMS; updateOLED(); }
            if (down) { menuCursor = (menuCursor +
1) % MENU_ITEMS; updateOLED(); }
            if (ok) {
                int selected = menuCursor;
                if (selected == 0) {

```

```

    currentScreen = SCR_MENU_WIFI;
    menuCursor = isSTAMode ? 1 : 0;
} else if (selected == 1) {
    currentScreen = SCR_MENU_GPS;
    menuCursor = useSeedData ? 1 : 0;
} else if (selected == 2) {
    currentScreen = SCR_MENU_SOIL;
    menuCursor = 0;
} else {
    currentScreen = SCR_MENU_CAL;
    menuCursor = 0;
}
updateOLED();
}
break;

case SCR_MENU_WIFI:
    if (menu) { currentScreen = SCR_MENU;
menuCursor = 0; updateOLED(); }
    if (up || down) { menuCursor = 1 -
menuCursor; updateOLED(); }
    if (ok) {
        bool wantsAP = (menuCursor == 0);
        Serial.printf("[BTN] WiFi menu OK:
cursor=%d wantsAP=%d isSTAMode=%d\n",
            menuCursor, wantsAP,
isSTAMode);
        currentScreen = SCR_MAIN;
        if (wantsAP == !isSTAMode) {
            Serial.println("[BTN] WiFi mode
unchanged");
            updateOLED();
        } else {
            pendingWiFiSwitch = true;
            pendingWiFiApMode = wantsAP;
            Serial.println("[BTN] WiFi switch
pending -> " + String(wantsAP ? "AP" : "STA"));
            display.clearDisplay();
            display.setTextSize(1);
            display.setCursor(0, 0);
            display.println("Switching
WiFi...");
            display.println(wantsAP ? "-> AP
mode" : "-> STA mode");
            display.display();
        }
        break;

case SCR_MENU_GPS:
    if (menu) { currentScreen = SCR_MENU;
menuCursor = 1; updateOLED(); }
    if (up || down) { menuCursor = 1 -
menuCursor; updateOLED(); }
    if (ok) {
        bool newSeed = (menuCursor == 1);
        if (newSeed != useSeedData) {
            Serial.println("[BTN] GPS/SEED
changed -> " + String(newSeed ? "SEED" :
"GPS"));
            pointCount = 0;
            useSeedData = newSeed;
            if (useSeedData) {
                randomSeed(analogRead(34));
                generateSeedPoints();
            }
            updateOLED();
        }
        break;

case SCR_MENU_SOIL:
    if (menu) { currentScreen = SCR_MENU;
menuCursor = 2; updateOLED(); }
    if (ok) {
        Serial.println("[BTN] Soil
measure...");
        digitalWrite(PIN_SOIL_PWR, HIGH);
        delay(100);
        long sum = 0;

```

```

        for (int i = 0; i < 8; i++) { sum +=
analogRead(PIN_SOIL_ANA);
delayMicroseconds(500); }
        digitalWrite(PIN_SOIL_PWR, LOW);
        sens.soilRaw = sum / 8;
        const int DRY = 2800, WET = 800;
        sens.soilPct
constrain(map(sens.soilRaw, DRY, WET, 0, 100),
0, 100);
        sens.lastSoilMs = millis();
        Serial.printf("[SOIL] BTN raw=%d →
%d%\n", sens.soilRaw, sens.soilPct);
        updateOLED();
    }
    break;

    case SCR_MENU_CAL:
        if (menu) { currentScreen = SCR_MENU;
menuCursor = 3; updateOLED(); break; }
        if (calState == CAL_IDLE && calProgress
== 100) {
            if (ok) { calProgress = 0; menuCursor
= 0; updateOLED(); }
            } else if (calState == CAL_IDLE ||
calState == CAL_ERROR) {
                if (up) { menuCursor = (menuCursor
- 1 + 3) % 3; updateOLED(); }
                if (down) { menuCursor = (menuCursor
+ 1) % 3; updateOLED(); }
                if (ok) {
                    CalMode mode = (menuCursor == 0) ?
MODE_STANDARD :
                        (menuCursor == 1) ?
MODE_WET : MODE_SENSITIVE;
                    Serial.println("[BTN] Start cal
mode=" + String(menuCursor));
                    startCalibration(mode);
                    updateOLED();
                }
            }
        }
        break;
    }

    void handleData() {
        float v = readVoltageCached();
        String j = "{";
        j += "\"s\":" + String(v, 4);
        j += ",\"c\":" + String(isCalibrating()
? "true" : "false");
        j += ",\"step\":" + String(currentLevel);
        j += ",\"prog\":" + String(calProgress);
        j += ",\"msg\":" + calStatus + "\"";
        j += ",\"temp\":"
(isnan(sens.temperature) ? String("null") :
String(sens.temperature, 1));
        j += ",\"hum\":" + (isnan(sens.humidity)
? String("null") : String(sens.humidity, 1));
        j += ",\"soil\":" + String(sens.soilPct);
        j += ",\"dist\":" + String(sens.distanceCm,
1);
        j += ",\"gpsFix\":"
String(gps.location.isValid() ? "true" :
"false");
        j += ",\"gpsSats\":"
String(gps.satellites.value());
        j += ",\"gpsLat\":"
String(gps.location.lat(), 6);
        j += ",\"gpsLon\":"
String(gps.location.lng(), 6);
        j += ",\"ptCount\":" + String(pointCount);
        j += "}";
        server.send(200, "application/json", j);
    }

    void handleStep() {
        if (!isCalibrating() && server.hasArg("v"))
        {
            int target =
constrain(server.arg("v").toInt(), 0,
POT_MAX);
            writePot((uint8_t)target);
        }
    }
}

```

```

        scheduleNvsWrite();
    }
    server.send(200, "application/json",
"{\"step\": \" + String(currentLevel) + \"}\"");
}

void handleCal() {
    int mode = 0;
    if (server.hasArg("mode")) mode =
server.arg("mode").toInt();
    CalMode m = (mode == 1) ? MODE_WET : (mode
== 2) ? MODE_SENSITIVE : MODE_STANDARD;
    server.send(200, "text/plain", "ok");
    startCalibration(m);
}

void handleReset() {
    if (!isCalibrating()) {
        writePot(POT_RESET_DEFAULT);
        calStatus = "";
        calProgress = 0;
        scheduleNvsWrite();
    }
    server.send(200, "text/plain", "ok");
}

void handleDiag() {
    float sig = readAvg();
    String o = "level=" + String(currentLevel)
+ "\n"
        + "avg=" + String(sig, 4) + "V\n"
        + "target="
String(TARGET_VOLTAGE, 3) + "V\n"
        + "cal=" + String((int)calState)
+ "\n"
        + "mode=" + String(isSTAMode ?
"STA" : "AP") + "\n"
        + "seed=" + String(useSeedData ?
"YES" : "NO") + "\n"
        + "gps="
String(gps.location.isValid() ? "FIX" : "NO
FIX") + "\n"
        + "points=" + String(pointCount) +
"\n"
        + "temp="
String(sens.temperature, 1) + "C\n"
        + "hum=" + String(sens.humidity,
1) + "%\n"
        + "soil=" + String(sens.soilPct)
+ "% (raw=" + String(sens.soilRaw) + ")\n"
        + "dist="
String(sens.distanceCm, 1) + "cm\n";
    server.send(200, "text/plain", o);
}

void handleMapData() {
    float currentVoltage = readVoltageCached();
    bool hasValidFix
= gps.location.isValid() && (gps.location.age()
< 3000);
    server.setContentLength(CONTENT_LENGTH_UNKNOW
N);
    server.send(200, "application/json", "");
    String chunk;
    chunk.reserve(512);
    chunk = "{";
    chunk += "\"valid\":"
String(hasValidFix ? "true" : "false");
    chunk += ", \"lat\":"
String(gps.location.lat(), 6);
    chunk += ", \"lon\":"
String(gps.location.lng(), 6);
    chunk += ", \"sats\":"
String(gps.satellites.value());
    chunk += ", \"signal\":"
String(currentVoltage, 3);
}

```

```

    chunk += "\",intensity\":" + "\",s\":" +
String(normalizeSignal(currentVoltage), 3); String(gpsPoints[i].soil)
    chunk += "\",level\":" + "\",d\":" +
String(currentLevel); String(gpsPoints[i].dist, 1)
    chunk += "\",point_count\":" + " + "}}";
String(pointCount);
    chunk += "\",temp\":" +
(isnan(sens.temperature) ? String("null") :
String(sens.temperature, 1));
    chunk += "\",hum\":" +
(isnan(sens.humidity) ? String("null") :
String(sens.humidity, 1));
    chunk += "\",soil\":" +
String(sens.soilPct);
    chunk += "\",dist\":" +
String(sens.distanceCm, 1);
    chunk += "\",seedMode\":" +
String(useSeedData ? "true" : "false");
    chunk += "\",points\":[ ";
server.sendContent(chunk);
    chunk = "";

const int BATCH_SIZE = 10;
for (int i = 0; i < pointCount; i++) {
    if (i > 0) chunk += ",";
    chunk += "{\lat\":" +
String(gpsPoints[i].lat, 6)
        + "\",lon\":" +
String(gpsPoints[i].lon, 6)
        + "\",i\":" +
String(gpsPoints[i].intensity, 3)
        + "\",v\":" +
String(gpsPoints[i].voltage, 3)
        + "\",t\":" +
(isnan(gpsPoints[i].temp) ? String("null") :
String(gpsPoints[i].temp, 1))
        + "\",h\":" +
(isnan(gpsPoints[i].hum) ? String("null") :
String(gpsPoints[i].hum, 1))

```

```

server.sendContent(chunk);

for (int i = 0; i < pointCount; i++) {
    HeatPoint &p = gpsPoints[i];
    chunk = (i > 0) ? "," : "";
    chunk
    +=
    "{\"type\": \"Feature\", \"geometry\": {\"type\":
    : \"Point\", \"coordinates\": [";
    chunk += String(p.lon, 6) + "," +
    String(p.lat, 6);
    chunk += "]}, \"properties\": {";
    chunk += "\"intensity\": " +
    String(p.intensity, 3);
    chunk += ", \"voltage\": " +
    String(p.voltage, 3);
    chunk += ", \"temp\": " + (isnan(p.temp)
    ? String("null") : String(p.temp, 1));
    chunk += ", \"humidity\": " + (isnan(p.hum)
    ? String("null") : String(p.hum, 1));
    chunk += ", \"soil\": " +
    String(p.soil);
    chunk += ", \"dist\": " + String(p.dist,
    1);
    chunk += ", \"index\": " + String(i);
    chunk += "}}";
    server.sendContent(chunk);

    if (i % 20 == 0) delay(1);
}

server.sendContent("}]);");
server.sendContent("");
Serial.println("[HTTP] Exported " +
String(pointCount) + " points as GeoJSON");
}

const char AP_HTML[] PROGMEM = R"rawliteral(
<!DOCTYPE html>
<html lang="uk">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-
width, initial-scale=1">
<title>W-MetalDetector</title>
<style>
    @import
    url('https://fonts.googleapis.com/css2?family
    =Share+Tech+Mono&family=Orbitron:wght@400;700
    ;900&display=swap');
    :root {
        --bg:#0a0c0f;        --panel:#111418;        --
border:#1e2530;
        --accent:#00e5a0;    --accent2:#ff6b35;    --
dim:#3a4a5a;
        --text:#c8d8e8;    --textdim:#5a6a7a;    --
danger:#ff3b5c;
        --mono:'Share Tech Mono',monospace;    --
display:'Orbitron',sans-serif;
    }
    *,*::before,*::after{box-sizing:border-
box;margin:0;padding:0}
    body{background:var(--bg);color:var(--
text);font-family:var(--mono);min-
height:100vh;padding-bottom:40px}
    .topbar{display:flex;align-
items:center;justify-content:space-
between;padding:14px 20px;background:var(--
panel);border-bottom:2px solid var(--
accent);position:sticky;top:0;z-index:50}
    .topbar-logo{font-family:var(--
display);font-size:1.3rem;font-
weight:900;letter-spacing:4px;color:var(--
accent);text-shadow:0 0 18px
rgba(0,229,160,.45)}
    .topbar-logo span{color:var(--accent2)}
    .status-pill{display:flex;align-
items:center;gap:7px;font-
size:.62rem;color:var(--textdim);letter-
spacing:2px}
    .dot{width:9px;height:9px;border-
radius:50%;background:var(--accent);box-

```

```

shadow:0 0 8px var(--accent);animation:blink
2s infinite}
@keyframes
blink{0%,100%{opacity:1}50%{opacity:.2}}
.grid{display:grid;grid-template-
columns:1fr 1fr;gap:14px;padding:16px;max-
width:900px;margin:0 auto}
@media(max-width:580px){.grid{grid-
template-columns:1fr}}
.span2{grid-column:span 2}
@media(max-width:580px){.span2{grid-
column:span 1}}
.panel{background:var(--panel);border:1px
solid var(--border);border-
radius:10px;overflow:hidden}
.ph{display:flex;align-
items:center;gap:9px;padding:9px
16px;background:rgba(0,229,160,.04);border-
bottom:1px solid var(--border);font-
family:var(--display);font-size:.6rem;letter-
spacing:3px;color:var(--accent);text-
transform:uppercase}
.pb{padding:18px 16px}
.big-v{font-family:var(--display);font-
size:3.6rem;font-weight:900;color:var(--
accent);text-align:center;line-height:1;text-
shadow:0 0 28px rgba(0,229,160,.45);transition:color
.25s;text-shadow .25s;letter-spacing:1px}
.big-v.hot{color:var(--danger);text-
shadow:0 0 28px rgba(255,59,92,.55)}
.volt-lbl{text-align:center;color:var(--
textdim);font-size:.65rem;letter-
spacing:3px;text-transform:uppercase;margin-
top:5px}
.mrow{display:flex;align-
items:center;gap:10px;margin-top:13px}
.mlbl{font-size:.63rem;color:var(--
textdim);width:88px;flex-shrink:0}
.mbar{flex:1;height:8px;background:var(--
border);border-radius:4px;overflow:hidden}
.mfill{height:100%;background:linear-
gradient(90deg,var(--accent),#00ffcc);border-
radius:4px;transition:width .3s}
.mfill.hot{background:linear-
gradient(90deg,var(--accent2),var(--danger))}
.irow{display:flex;justify-content:space-
between;align-items:center;padding:8px
0;border-bottom:1px solid var(--border);font-
size:.72rem}
.irow:last-child{border-bottom:none}
.ikey{color:var(--textdim)}
.ival{color:var(--text);font-family:var(--
display);font-size:.75rem}
.ival.g{color:var(--accent)}
.ival.o{color:var(--accent2)}
.ival.r{color:var(--danger)}
.chart-
wrap{position:relative;height:160px;backgroun
d:var(--bg);border-
radius:6px;overflow:hidden}
canvas#sigChart{position:absolute;inset:0;wid
th:100%;height:100%}
.sens-val{font-family:var(--display);font-
size:2.8rem;font-weight:900;color:var(--
accent);text-align:center;line-height:1;text-
shadow:0 0 20px rgba(0,229,160,.35);margin-
bottom:4px}
.sens-sub{text-align:center;font-
size:.62rem;color:var(--textdim);letter-
spacing:3px;margin-bottom:18px}
.nud-wrap{display:flex;flex-
direction:column;gap:8px}
.nud-row{display:flex;gap:6px}
.nud-row label{font-size:.6rem;color:var(--
textdim);letter-
spacing:2px;width:28px;display:flex;align-
items:center;flex-shrink:0}
.nud-btn{flex:1;padding:10px
4px;background:transparent;border:1px solid
var(--border);border-radius:7px;color:var(--

```

```

text);font-family:var(--mono);font-size:.78rem;font-weight:700;cursor:pointer;transition:all .15s;letter-spacing:1px}
.nud-btn:hover:not(:disabled){border-color:var(--accent);color:var(--accent);background:rgba(0,229,160,.06)}
.nud-btn:active:not(:disabled){transform:scale(.96)}
.nud-btn.disabled{opacity:.3;cursor:not-allowed}
.nud-btn.minus:hover:not(:disabled){border-color:var(--danger);color:var(--danger);background:rgba(255,59,92,.06)}
.nud-spinner{display:none;width:14px;height:14px;border:2px solid var(--border);border-top-color:var(--accent);border-radius:50%;animation:spin .6s linear infinite;margin:0 auto}
.nud-btn.loading .nud-lbl{display:none}
.nud-btn.loading .nud-spinner{display:block}
@keyframes spin{to{transform:rotate(360deg)}}

.btn{display:block;width:100%;padding:13px;border:none;border-radius:8px;font-family:var(--display);font-size:.72rem;font-weight:700;letter-spacing:2px;cursor:pointer;transition:all .18s;text-transform:uppercase}
.btn-p{background:var(--accent);color:#000;box-shadow:0 4px 20px rgba(0,229,160,.28)}
.btn-p:hover{box-shadow:0 4px 30px rgba(0,229,160,.55);transform:translateY(-1px)}

.btn-s{background:transparent;color:var(--accent);border:1px solid var(--accent);margin-top:8px}
.btn-s:hover{background:rgba(0,229,160,.07)}
.btn-o{background:transparent;color:var(--accent2);border:1px solid var(--accent2);margin-top:8px}
.btn-o:hover{background:rgba(255,107,53,.07)}
.btn-d{background:transparent;color:var(--danger);border:1px solid var(--danger);margin-top:8px}
.btn-d:hover{background:rgba(255,59,92,.07)}

.cal-grid{display:grid;grid-template-columns:1fr 1fr 1fr;gap:10px}
@media(max-width:600px){.cal-grid{grid-template-columns:1fr}}
.cal-desc{font-size:.68rem;color:var(--textdim);line-height:1.6;margin-bottom:10px}
.cal-mode-badge{display:inline-block;font-size:.58rem;letter-spacing:2px;padding:2px 7px;border-radius:4px;margin-bottom:8px;font-family:var(--display)}
.badge-std{background:rgba(0,229,160,.12);color:var(--accent);border:1px solid rgba(0,229,160,.3)}
.badge-wet{background:rgba(255,107,53,.12);color:var(--accent2);border:1px solid rgba(255,107,53,.3)}
.badge-sens{background:rgba(255,59,92,.12);color:var(--danger);border:1px solid rgba(255,59,92,.3)}
.sensor-grid{display:grid;grid-template-columns:repeat(4,1fr);gap:10px;margin-bottom:0}

```

```

@media(max-width:600px){.sensor-grid{grid-
template-columns:repeat(2,1fr)}}
.scard{background:var(--bg);border:1px
solid var(--border);border-
radius:8px;padding:12px 10px;text-
align:center}
.scard-icon{font-size:1.4rem;margin-
bottom:5px}
.scard-val{font-family:var(--display);font-
size:1.05rem;font-weight:700;color:var(--
accent);margin-bottom:3px}
.scard-lbl{font-size:.55rem;color:var(--
textdim);letter-spacing:2px;text-
transform:uppercase}
.scard-val.orange{color:var(--accent2)}
.scard-val.red{color:var(--danger)}
.scard-val.blue{color:#5599ff}
.soil-bar-wrap{height:5px;background:var(--
border);border-radius:3px;margin-
top:5px;overflow:hidden}
.soil-bar-fill{height:100%;border-
radius:3px;transition:width .5s;background
.5s}
.modal-
bg{display:none;position:fixed;inset:0;backgr-
ound:rgba(0,0,0,.88);z-index:100;align-
items:center;justify-content:center;backdrop-
filter:blur(5px)}
.modal-bg.open{display:flex}
.modal-box{background:var(--
panel);border:1px solid var(--accent);border-
radius:16px;padding:28px
22px;width:min(430px,92vw);box-shadow:0 0
60px rgba(0,229,160,.12);animation:su .28s
ease}
@keyframes
su{from{opacity:0;transform:translateY(20px)}
to{opacity:1;transform:none}}
.modal-title{font-family:var(--
display);font-size:1rem;font-
weight:700;color:var(--accent);letter-
spacing:3px;margin-bottom:8px}
.modal-sub{color:var(--textdim);font-
size:.72rem;line-height:1.6;margin-
bottom:18px}
.warn{background:rgba(255,107,53,.07);border:
1px solid rgba(255,107,53,.28);border-
radius:8px;padding:12px 14px;font-
size:.7rem;color:var(--accent2);line-
height:1.65;margin-bottom:16px}
.warn strong{display:block;margin-
bottom:4px;font-size:.78rem}
.cal-live-grid{display:grid;grid-template-
columns:1fr 1fr;gap:10px;margin:14px 0}
.cal-live-box{text-align:center;border:1px
solid var(--border);border-
radius:8px;padding:14px 8px}
.cal-live-
box.v{background:rgba(0,229,160,.04)}
.cal-
live-box.s{background:rgba(255,107,53,.04)}
.cal-num{font-family:var(--display);font-
size:2rem;font-weight:700;line-
height:1;margin-bottom:4px}
.cal-num.v{color:var(--accent2)}
.cal-
num.s{color:var(--accent)}
.cal-lbl{font-size:.6rem;color:var(--
textdim);letter-spacing:3px}
.prog-bar{height:10px;background:var(--
border);border-
radius:5px;overflow:hidden;margin:12px 0 4px}
.prog-fill{height:100%;background:linear-
gradient(90deg,var(--accent),#00ffcc);border-
radius:5px;transition:width .35s
ease;width:0%}
.prog-row{display:flex;justify-
content:space-between;font-
size:.6rem;color:var(--textdim);margin-
bottom:10px}
.cal-phases{display:flex;gap:4px;margin-
bottom:10px}

```

```

    .cal-phase{flex:1;height:4px;border-
radius:2px;background:var(--
border);transition:background .3s}
    .cal-phase.done{background:var(--accent)}
    .cal-phase.active{background:var(--
accent2);animation:pulse 1s infinite}
    @keyframes
pulse{0%,100%{opacity:1}50%{opacity:.4}}
    .cal-status-txt{font-
size:.68rem;color:var(--textdim);text-
align:center;min-height:17px;letter-
spacing:1px;margin-bottom:10px;font-
style:italic}
    ::-webkit-scrollbar{width:5px}
    ::-webkit-scrollbar-track{background:var(--
bg)}
    ::-webkit-scrollbar-thumb{background:var(--
dim);border-radius:3px}
</style>
</head>
<body>
<div class="topbar">
    <div class="topbar-logo">W-MetalDetector
<span>PRO</span></div>
    <div class="status-pill">AP MODE <div
class="dot"></div></div>
</div>
<div class="grid">
    <div class="panel span2">
        <div class="ph"> 🚧 Датчики
середовища</div>
        <div class="pb">
            <div class="sensor-grid">
                <div class="scard">
                    <div class="scard-icon"> 🚧 </div>
                    <div class="scard-val" id="sTemp">-
- </div>
                    <div class="scard-
lbl">Температура</div>
                </div>

```

```

<div class="scard">
    <div class="scard-icon"> 💧 </div>
    <div class="scard-val" id="sHum">- - </div>
    <div class="scard-lbl">Вологість
повітря</div>
</div>
<div class="scard" id="soilCard">
    <div class="scard-icon"> 🌱 </div>
    <div class="scard-val" id="sSoil">- - </div>
    <div class="scard-lbl">Вологість
ґрунту</div>
    <div class="soil-bar-wrap"><div
class="soil-bar-fill" id="sSoilBar"
style="width:0%;background:#ff6b35"></div></d
iv>
    <div class="soilAge" style="font-
size:.52rem;color:var(--textdim);margin-
top:4px;letter-spacing:1px"
id="soilAge">натисніть кнопку</div>
</div>
<div class="scard">
    <div class="scard-icon"> 📏 </div>
    <div class="scard-val" id="sDist">-
- </div>
    <div class="scard-lbl">Відстань
(live)</div>
    <div class="scard-lbl" style="font-
size:.52rem;color:var(--textdim);margin-
top:4px;letter-spacing:1px">HC-SR04 · оновл.
200мс</div>
</div>
</div>
<div class="panel span2">
    <div class="ph"> 📶 GPS · Запис
точок</div>
    <div class="pb">

```

```

    <div class="sensor-grid" style="grid-
template-columns:repeat(4,1fr)">
    <div class="scard">
        <div class="scard-icon"><img alt="GPS icon" /></div>
        <div class="scard-val"
id="gpsStatus">NO FIX</div>
        <div class="scard-lbl">GPS
статус</div>
    </div>
    <div class="scard">
        <div class="scard-icon"><img alt="GPS icon" /></div>
        <div class="scard-val blue"
id="gpsSats">0</div>
        <div class="scard-
lbl">Супутники</div>
    </div>
    <div class="scard">
        <div class="scard-icon"><img alt="GPS icon" /></div>
        <div class="scard-val orange"
id="gpsPoints">0</div>
        <div class="scard-lbl">Записано
точок</div>
    </div>
    <div class="scard">
        <div class="scard-icon"><img alt="GPS icon" /></div>
        <div class="scard-val"
id="gpsCoords" style="font-
size:.65rem;letter-spacing:0">--</div>
        <div class="scard-
lbl">Координати</div>
    </div>
    <div style="margin-top:10px;padding:8px
10px;background:var(--bg);border-
radius:6px;border:1px solid var(--
border);font-size:.65rem;color:var(--
textdim);line-height:1.7">
        Точки записуються автоматично в обох
режимах (AP i STA).<br>

```

```

        Умова: сигнал GPIO34 > 0.22V <b
style="color:var(--accent)">+</b> GPS фікс.
Перемкни в STA щоб побачити карту.
    </div>
</div>
</div>
<div class="panel">
    <div class="ph"><img alt="Signal icon" /> Сигнал</div>
    <div class="pb">
        <div class="big-v"
id="bigV">0.000V</div>
        <div class="volt-lbl">ADC . GPIO
34</div>
        <div class="mrow">
            <div class="mlbl">Рівень</div>
            <div class="mbar"><div class="mfill"
id="sigBar" style="width:0%"></div></div>
            <div class="mval"
id="sigPct">0%</div>
        </div>
        <div class="mrow">
            <div class="mlbl">Попіг тиші</div>
            <div class="mbar"><div class="mfill"
style="width:6.7%;opacity:.35"></div></div>
            <div class="mval" style="color:var(--
textdim)">0.22V</div>
        </div>
    </div>
</div>
<div class="panel">
    <div class="ph"><img alt="Info icon" /> Інформація</div>
    <div class="pb">
        <div class="irow"><span
class="ikey">Рівень TPL0501 (0-
255)</span><span class="ival g" id="iStep">--
</span></div>
        <div class="irow"><span
class="ikey">Попіг</span><span class="ival
o">0.22 V</span></div>

```

```

        <div class="irow"><span class="ikey">Offset реж.</span><span class="ival">+4 / +5 / -1</span></div>
        <div class="irow"><span class="ikey">Мікросхема</span><span class="ival o">TPL0501 (SPI)</span></div>
        <div class="irow"><span class="ikey">Статус</span><span class="ival g" id="iStatus">IDLE</span></div>
    </div>
</div>
<div class="panel span2">
    <div class="ph">📊 Графік сигналу (останні ~60 с)</div>
    <div class="pb" style="padding:12px 16px 14px">
        <div class="chart-wrap"><canvas id="sigChart"></canvas></div>
    </div>
</div>
<div class="panel span2">
    <div class="ph">🔧 Ручне керування</div>
    <div class="pb">
        <div class="sens-val" id="sensVal">-</div>
        <div class="sens-sub">PIBЕНЬ ПОТЕНЦІОМЕТРА (0 - 255)</div>
        <div class="nud-wrap">
            <div class="nud-row">
                <label>+</label>
                <button class="nud-btn" onclick="nudge(event,+1)"><span class="nud-lbl">+1</span><div class="nud-spinner"></div></button>
                <button class="nud-btn" onclick="nudge(event,+5)"><span class="nud-lbl">+5</span><div class="nud-spinner"></div></button>
                <button class="nud-btn" onclick="nudge(event,+10)"><span class="nud-
lbl">+10</span><div class="nud-spinner"></div></button>
                <button class="nud-btn" onclick="nudge(event,+25)"><span class="nud-lbl">+25</span><div class="nud-spinner"></div></button>
            </div>
            <div class="nud-row">
                <label>-</label>
                <button class="nud-btn minus" onclick="nudge(event,-1)"><span class="nud-lbl">-1</span><div class="nud-spinner"></div></button>
                <button class="nud-btn minus" onclick="nudge(event,-5)"><span class="nud-lbl">-5</span><div class="nud-spinner"></div></button>
                <button class="nud-btn minus" onclick="nudge(event,-10)"><span class="nud-lbl">-10</span><div class="nud-spinner"></div></button>
                <button class="nud-btn minus" onclick="nudge(event,-25)"><span class="nud-lbl">-25</span><div class="nud-spinner"></div></button>
            </div>
        </div>
        <div class="mrow" style="margin-top:14px">
            <div class="mlbl">Onip TPL0501</div>
            <div class="mbar"><div class="mfill" id="sensBar" style="width:50%"></div></div>
            <div class="mval" id="sensStep">128/255</div>
        </div>
    </div>
</div>
<div class="panel span2">
    <div class="ph">⚙️ Автокалібрування</div>
    <div class="pb">
        <div class="cal-grid">

```

```

    <div>
      <div class="cal-mode-badge badge-std">СТАНДАРТ</div>
      <div class="cal-desc">Каскадний пошук +20→+1. Знаходить точну межу тиші, відступ -4 вглиб зони тиші.</div>
      <button class="btn btn-p" onclick="openCalModal(0)">▶ СТАНДАРТ</button>
    </div>
    <div>
      <div class="cal-mode-badge badge-wet">ВОЛОГИЙ ГРУНТ</div>
      <div class="cal-desc">Той самий алгоритм, але відступ -5. Більший запас для вологого ґрунту.</div>
      <button class="btn btn-o" onclick="openCalModal(1)">▶ ВОЛОГИЙ ГРУНТ</button>
    </div>
    <div>
      <div class="cal-mode-badge badge-sens">НАДЧУТЛИВИЙ</div>
      <div class="cal-desc">Відступ +1 від межі тиші. Максимальна чутливість - можливі хибні спрацювання.</div>
      <button class="btn btn-d" onclick="openCalModal(2)">▶ НАДЧУТЛИВИЙ</button>
    </div>
    <div style="margin-top:14px;padding-top:14px;border-top:1px solid var(--border)">
      <div class="cal-desc" style="margin-bottom:8px">Reset: level 128 (midpoint of TPL0501 range).</div>
      <button class="btn btn-s" style="margin:0" onclick="doReset()">↻ RESET (128)</button>
      <button class="btn btn-o" style="margin-top:8px"
        onclick="exportPoints()">↓ EXPORT POINTS (GeoJSON)</button>
    </div>
  </div>
</div>
</div>
<div class="modal-bg" id="calModal">
  <div class="modal-box">
    <div id="mStep1">
      <div class="modal-title">ПІДГОТОВКА</div>
      <div class="modal-sub">Перед запуском автокалібрування:</div>
      <div class="warn">
        <strong>⚠ Обов'язково!</strong>
        <ul>
          <li>Підніміть котушку від землі (≥ 30 см)<br>
          <li>Приберіть метал у радіусі 1 м<br>
          <li>Зачекайте 5-10 с після підняття<br>
          <li>Металощукач має бути увімкнений та прогрітий
        </ul>
      </div>
      <div class="irow" style="margin-bottom:8px">
        <span class="ikey">Режим</span><span class="ival o" id="preMode">-</span>
      </div>
      <div class="irow" style="margin-bottom:14px">
        <span class="ikey">Поточний сигнал</span><span class="ival g" id="preV">-</span>
      </div>
      <button class="btn btn-p" onclick="startCal()">✓ ГОТОВО, ПОЧАТИ</button>
      <button class="btn btn-s" onclick="closeModal()">× СКАСУВАТИ</button>
    </div>
    <div id="mStep2" style="display:none">

```

```

    <div class="modal-
title">КАЛІБРУВАННЯ...</div>
    <div class="cal-phases">
        <div class="cal-phase active"
id="ph1"></div>
        <div class="cal-phase"
id="ph2"></div>
        <div class="cal-phase"
id="ph3"></div>
    </div>
    <div class="cal-live-grid">
        <div class="cal-live-box v"><div
class="cal-num v" id="calV">0.000V</div><div
class="cal-lbl">СИГНАЛ</div></div>
        <div class="cal-live-box s"><div
class="cal-num s" id="calS">0</div><div
class="cal-lbl">ПІВЕНЬ (0-255)</div></div>
    </div>
    <div class="cal-status-txt"
id="calMsg">Ініціалізація...</div>
    <div class="prog-bar"><div class="prog-
fill" id="progFill"></div></div>
    <div class="prog-
row"><span>0%</span><span
id="progPct">0%</span><span>100%</span></div>
    </div>
    <div id="mStep3" style="display:none">
        <div class="modal-title"
id="doneTitle">✔ ГОТОВО!</div>
        <div class="modal-sub"
id="doneSub">Чутливість налаштована.</div>
        <div class="irow"><span
class="ikey">Фінальний рівень</span><span
class="ival g" id="doneStep">-</span></div>
        <div class="irow" style="margin-
bottom:14px"><span
class="ikey">Сигнал</span><span class="ival
g" id="doneV">-</span></div>
        <button class="btn btn-p"
onclick="closeModal()">✓ ЗАКРИТИ</button>
    </div>

```

```

</div>
</div>
<script>
    const CHART_LEN = 150, V_MAX = 2.7, V_MIN =
0.0, THRESHOLD = 0.22;
    const chartData = new
Array(CHART_LEN).fill(0);
    const canvas =
document.getElementById('sigChart');
    const ctx = canvas.getContext('2d');
    function resizeCanvas(){const
r=canvas.parentElement.getBoundingClientRect(
);canvas.width=r.width*devicePixelRatio;canva
s.height=r.height*devicePixelRatio;}
    function vToY(v,H){return H-
((Math.min(Math.max(v,V_MIN),V_MAX)-
V_MIN)/(V_MAX-V_MIN))*H;}
    function drawChart(){
        const
W=canvas.width,H=canvas.height,dpr=devicePixe
lRatio,stepX=W/(CHART_LEN-1);
        ctx.clearRect(0,0,W,H);
        ctx.strokeStyle=' rgba(30,37,48,1)';ctx.linewi
dth=dpr;ctx.fillStyle=' rgba(90,106,122,0.75)';
        ctx.font=` ${10*dpr}px "Share Tech
Mono",monospace`;ctx.textAlign='left';
        for(const v of
[0,0.5,1.0,1.5,2.0,2.5]){const
y=vToY(v,H);ctx.beginPath();ctx.moveTo(0,y);c
tx.lineTo(W,y);ctx.stroke();ctx.fillText(v.to
Fixed(1)+'V',4*dpr,y-3*dpr);}
        const thY=vToY(THRESHOLD,H);
        ctx.strokeStyle=' rgba(255,59,92,0.7)';ctx.lin
ewidth=1.5*dpr;ctx.setLineDash([6*dpr,4*dpr])
;
        ctx.beginPath();ctx.moveTo(0,thY);ctx.lineTo(
W,thY);ctx.stroke();ctx.setLineDash([]);

```

```

ctx.fillStyle='rgba(255,59,92,0.85)';ctx.font
=`${9*dpr}px          "Share          Tech
Mono",monospace`;ctx.textAlign='right';
  ctx.fillText('0.22V',W-4*dpr,thY-3*dpr);
  const
grad=ctx.createLinearGradient(0,0,0,H);

grad.addColorStop(0,'rgba(0,229,160,0.35)');g
rad.addColorStop(0.6,'rgba(0,229,160,0.08)');
grad.addColorStop(1,'rgba(0,229,160,0)');
  ctx.beginPath();
  for(let      i=0;i<CHART_LEN;i++){const
x=i*stepX,y=vToY(chartData[i],H);i===0?ctx.mo
veTo(x,y):ctx.lineTo(x,y);}
  ctx.lineTo((CHART_LEN-
1)*stepX,H);ctx.closePath();ctx.fillStyle=gra
d;ctx.fill();
  ctx.lineWidth=2*dpr;ctx.lineJoin='round';
  for(let i=1;i<CHART_LEN;i++){
    const      x0=(i-
1)*stepX,y0=vToY(chartData[i-
1],H),x1=i*stepX,y1=vToY(chartData[i],H);

ctx.strokeStyle=chartData[i]>=THRESHOLD?'#ff3
b5c':'#00e5a0';

ctx.beginPath();ctx.moveTo(x0,y0);ctx.lineTo(
x1,y1);ctx.stroke();
  }
}
resizeCanvas();drawChart();

window.addEventListener('resize',()=>{resizeC
anvas();drawChart();});

let
wasCalibrating=false,nudgePending=false,pendi
ngMode=0;
  const modeNames=['СТАНДАРТ (-4)','ВОЛОГИЙ
ГРУНТ (-5)','НАДЧУТЛИВИЙ (+1)'];

```

```

var soilLastMs = 0;

function updateSensors(d) {
  var      tEl      =
document.getElementById('sTemp');
  if (d.temp === null || d.temp ===
undefined) { tEl.textContent = '--';
tEl.className='scard-val'; }
  else {
    tEl.textContent = d.temp.toFixed(1) +
'°C';
    tEl.className = 'scard-val ' + (d.temp
> 35 ? 'red' : d.temp < 5 ? 'blue' : '');
  }

  var      hEl      =
document.getElementById('sHum');
  if (d.hum === null || d.hum === undefined)
{ hEl.textContent = '--'; }
  else { hEl.textContent = d.hum.toFixed(0)
+ '%'; }

  var      soilEl   =
document.getElementById('sSoil');
  var      soilBar  =
document.getElementById('sSoilBar');
  var      soilAge  =
document.getElementById('soilAge');
  var soil = (d.soil !== undefined && d.soil
> 0) ? d.soil : null;
  if (soil !== null) {
    var prev = parseInt(soilEl.textContent)
|| 0;
    if (prev !== soil) soilLastMs =
Date.now();
    soilEl.textContent = soil + '%';
    soilBar.style.width = soil + '%';
    soilBar.style.background = soil > 60 ?
'#5599ff' : soil > 30 ? '#ffd166' : '#ff3b5c';

```

```

        soilEl.className = 'scard-val ' + (soil
> 60 ? 'blue' : soil > 30 ? 'orange' : 'red');
        if (soilLastMs > 0) {
            var ageSec = Math.floor((Date.now() -
soilLastMs) / 1000);
            soilAge.textContent = ageSec < 60 ?
'виміряно ' + ageSec + 'с тому'
            : 'виміряно ' +
Math.floor(ageSec/60) + 'хв тому';
        }
    } else {
        soilEl.textContent = '--';
        soilBar.style.width = '0%';
        soilAge.textContent = 'натисніть кнопку
GPIO26';
    }

    var dEl = document.getElementById('sDist');
    if (d.dist === undefined || d.dist < 0) {
        dEl.textContent = '--';
        dEl.className = 'scard-val';
    } else {
        dEl.textContent = d.dist.toFixed(1) + '
см';
        dEl.className = 'scard-val ' + (d.dist
< 10 ? 'red' : d.dist < 30 ? 'orange' : 'g');
    }
}

function nudge(evt,delta){
    if(nudgePending)return;
    const
cur=parseInt(document.getElementById('sensVal
').textContent)||0;
    const
nxt=Math.max(0,Math.min(255,cur+delta));
    if(nxt===cur)return;
    nudgePending=true;
    document.querySelectorAll('.nud-
btn').forEach(b=>b.disabled=true);
    evt.currentTarget.classList.add('loading');
    fetch('/step?v='+nxt).then(r=>r.json()).then(
d=>{
        applyStep(d.step);nudgePending=false;
        document.querySelectorAll('.nud-
btn').forEach(b=>{b.disabled=false;b.classLis
t.remove('loading');});
    }).catch(()=>{nudgePending=false;document.que
rySelectorAll('.nud-
btn').forEach(b=>{b.disabled=false;b.classLis
t.remove('loading');});});
}
function applyStep(step){
    document.getElementById('sensVal').textConten
t=step;
    document.getElementById('sensBar').style.widt
h=(step/255*100)+'%';
    document.getElementById('sensStep').textConte
nt=step+' /255';
    document.getElementById('iStep').textContent=
step;
}
function updatePhases(msg){
    const
fine=msg.includes('+1')||msg.includes('Точна'
);
    document.getElementById('ph1').className='cal
-phase '++(fine?'done':'active');
    document.getElementById('ph2').className='cal
-phase '++(fine?'active':'');
}

```

```

document.getElementById('ph3').className='cal
-phase';
}
setInterval(poll,400);
function poll(){
fetch('/data').then(r=>r.json()).then(d=>{
    const
v=d.s,pct=Math.min(100,v/3.3*100),hot=v>=0.22
;
chartData.push(v);if(chartData.length>CHART_L
EN)chartData.shift();drawChart();
document.getElementById('bigV').textContent=v
.toFixed(3)+'V';
document.getElementById('bigV').className='bi
g-v'+(hot?' hot':'');
    const
sb=document.getElementById('sigBar');sb.style
.width=pct.toFixed(1)+'%';sb.className='mfill
'+(hot?' hot':'');
document.getElementById('sigPct').textContent
=pct.toFixed(0)+'%';
document.getElementById('iStatus').textConten
t=d.c?'CALIBRATING':'IDLE';
if(!nudgePending&&!d.c)applyStep(d.step);
document.getElementById('preV').textContent=v
.toFixed(3)+' V';
    updateSensors(d);
    var
gpsFixEl
=
document.getElementById('gpsStatus');
    var
gpsSatsEl
=
document.getElementById('gpsSats');
    var
gpsPtsEl
=
document.getElementById('gpsPoints');
    var
gpsCoordsEl=
document.getElementById('gpsCoords');
    if (d.gpsFix) {
        gpsFixEl.textContent = 'FIX ✓';
        gpsFixEl.className = 'scard-val g';
        gpsCoordsEl.textContent
=
d.gpsLat.toFixed(5) + '\n' +
d.gpsLon.toFixed(5);
    } else {
        gpsFixEl.textContent = 'NO FIX';
        gpsFixEl.className = 'scard-val
red';
        gpsCoordsEl.textContent = '--';
    }
    var sats = d.gpsSats || 0;
    gpsSatsEl.textContent = sats;
    gpsSatsEl.className = 'scard-val ' +
(sats >= 6 ? 'g' : sats >= 3 ? 'orange' :
'red');
    gpsPtsEl.textContent = d.ptCount || 0;
    gpsPtsEl.className = 'scard-val ' +
((d.ptCount||0) > 0 ? 'orange' : '');
    if(d.c){
        wasCalibrating=true;showStep(2);
document.getElementById('calV').textContent=v
.toFixed(3)+'V';
document.getElementById('calS').textContent=d
.step;
document.getElementById('calMsg').textContent
=d.msg;
document.getElementById('progFill').style.wid
th=d.prog+'%';
document.getElementById('progPct').textConten
t=d.prog+'%';

```

```

        updatePhases(d.msg);
    } else if(wasCalibrating){
        wasCalibrating=false;
        const
isErr=d.msg.includes('Помилка');

document.getElementById('doneTitle').textCont
ent=isErr?' ❌ ПОМИЛКА':' ✅ ГОТОВО!';

document.getElementById('doneTitle').style.co
lor=isErr?'var(--danger)':'';

document.getElementById('doneSub').textConten
t=d.msg;

document.getElementById('doneStep').textConte
nt=d.step;

document.getElementById('doneV').textContent=
v.toFixed(3)+' V';
        showStep(3);
    }
}).catch(()=>{});
}
function
showStep(n){[1,2,3].forEach(i=>document.getEl
ementById('mStep'+i).style.display=i===n?'blo
ck':'none');document.getElementById('calModal
').classList.add('open');}
function
openCalModal(mode){pendingMode=mode;wasCalibr
ating=false;document.getElementById('preMode'
).textContent=modeNames[mode];showStep(1);}
function
closeModal(){document.getElementById('calModa
l').classList.remove('open');wasCalibrating=f
alse;}
function startCal(){
showStep(2);document.getElementById('calMsg')
.textContent='Старт...';

```

```

document.getElementById('progFill').style.wid
th='0%';

document.getElementById('ph1').className='cal
-phase active';

document.getElementById('ph2').className='cal
-phase';

document.getElementById('ph3').className='cal
-phase';

wasCalibrating=true;fetch('/cal?mode='+pendin
gMode);
    }
function doReset(){if(confirm('Reset
potentiometer to level
128?')){fetch('/reset').then(()=>applyStep(12
8));}}
function exportPoints(){
    var a=document.createElement('a');
    a.href='/export.json';
    a.download='wmd_points.geojson';

document.body.appendChild(a);a.click();docume
nt.body.removeChild(a);
    }
</script>
</body>
</html>
)rawliteral";

const char STA_HTML[] PROGMEM = R"rawliteral(
<!DOCTYPE html>
<html lang="uk">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-
width, initial-scale=1">
<title>W-MetalDetector MAP</title>

```

```

<link rel="stylesheet" href="https://unpkg.com/leaflet@1.9.4/dist/leaflet.css"/>
<script src="https://unpkg.com/leaflet@1.9.4/dist/leaflet.js"></script>
<style>
  @import url('https://fonts.googleapis.com/css2?family=Share+Tech+Mono&family=Orbitron:wght@400;700;900&display=swap');
  :root {
    --bg:#080a0d; --panel:#0e1117; --border:#1a2230;
    --accent:#00e5a0; --accent2:#ff6b35; --danger:#ff3b5c;
    --text:#c8d8e8; --textdim:#4a5a6a; --warn:#ffd166;
    --mono:'Share Tech Mono',monospace; display:'Orbitron',sans-serif;
  }
  *,*::before,*::after{box-sizing:border-box;margin:0;padding:0}

html,body{height:100%;overflow:hidden;background:var(--bg);color:var(--text);font-family:var(--mono)}
#topbar{
  position:fixed;top:0;left:0;right:0;z-index:1000;
  background:rgba(8,10,13,0.95);border-bottom:1px solid var(--border);
  backdrop-filter:blur(12px);display:flex;align-items:center;gap:0;
  height:52px;box-shadow:0 2px 20px rgba(0,0,0,.5);
}
.tb-logo{font-family:var(--display);font-size:.85rem;font-weight:900;letter-spacing:3px;color:var(--accent);padding:0
16px;border-right:1px solid var(--border);height:100%;display:flex;align-items:center;gap:8px;text-shadow:0 0 12px
rgba(0,229,160,.4);white-space:nowrap;}
.tb-logo span{color:var(--accent2)}
.tb-logo .dot{width:7px;height:7px;border-radius:50%;background:var(--accent);box-shadow:0 0 6px var(--accent);animation:blink
2s infinite;flex-shrink:0}
@keyframes blink{0%,100%{opacity:1}50%{opacity:.15}}
.tb-stats{display:flex;align-items:stretch;flex:1;overflow-x:auto;scrollbar-width:none}
.tb-stats::-webkit-scrollbar{display:none}
.tb-stat{display:flex;flex-direction:column;justify-content:center;padding:0 14px;border-right:1px solid var(--border);min-width:72px;flex-shrink:0;cursor:default;}
.tb-stat-lbl{font-size:.48rem;color:var(--textdim);letter-spacing:2px;text-transform:uppercase;margin-bottom:2px}
.tb-stat-val{font-family:var(--display);font-size:.78rem;font-weight:700;color:var(--text);letter-spacing:1px;white-space:nowrap}
.tb-stat-val.green{color:var(--accent)}
.tb-stat-val.orange{color:var(--accent2)}
.tb-stat-val.red{color:var(--danger)}
.tb-stat-val.warn{color:var(--warn)}
.tb-stat-val.blue{color:#5599ff}
.tb-signal{display:flex;flex-direction:column;justify-content:center;padding:0 16px;border-right:1px solid var(--border);min-width:110px;background:linear-gradient(135deg,rgba(0,229,160,.06),transparent);}

```

```

.tb-sig-bar{height:3px;background:var(--border);border-radius:2px;margin-top:4px;overflow:hidden}
.tb-sig-fill{height:100%;background:linear-gradient(90deg,var(--accent),#00ffcc);border-radius:2px;transition:width .4s}
.tb-actions{display:flex;align-items:center;gap:6px;padding:0 12px;flex-shrink:0}
.tb-btn{padding:5px 12px;background:transparent;border:1px solid var(--border);border-radius:6px;color:var(--textdim);font-family:var(--mono);font-size:.6rem;cursor:pointer;transition:all .15s;white-space:nowrap;}
.tb-btn:hover{border-color:var(--accent);color:var(--accent);background:rgba(0,229,160,.06)}
.tb-btn.danger:hover{border-color:var(--danger);color:var(--danger);background:rgba(255,59,92,.06)}
.tb-btn.active{border-color:var(--accent2);color:var(--accent2);background:rgba(255,107,53,.08)}

#map{position:fixed;top:52px;left:0;right:0;bottom:0}
#sensorDrawer{
  position:fixed;bottom:0;left:0;right:0;z-index:950;
  background:rgba(8,10,13,.97);border-top:1px solid var(--border);
  backdrop-filter:blur(14px);
  transform:translateY(100%);transition:transform .3s ease;
  padding:0 0 env(safe-area-inset-bottom,0);
}
#sensorDrawer.open{transform:translateY(0)}

.drawer-handle{display:flex;align-items:center;justify-content:space-between;padding:10px 16px;cursor:pointer;border-bottom:1px solid var(--border)}
.drawer-handle-title{font-family:var(--display);font-size:.6rem;letter-spacing:3px;color:var(--accent)}
.drawer-handle-arrow{font-size:.7rem;color:var(--textdim);transition:transform .3s}
#sensorDrawer.open .drawer-handle-arrow{transform:rotate(180deg)}
.drawer-grid{display:grid;grid-template-columns:repeat(4,1fr);gap:8px;padding:12px 14px}
@media(max-width:480px){.drawer-grid{grid-template-columns:repeat(2,1fr)}}
.drawer-stat{text-align:center;background:rgba(255,255,255,.03);border:1px solid var(--border);border-radius:8px;padding:10px 6px}
.drawer-lbl{font-size:.48rem;color:var(--textdim);letter-spacing:2px;text-transform:uppercase;margin-bottom:4px}
.drawer-val{font-family:var(--display);font-size:.9rem;font-weight:700;color:var(--text)}
.drawer-val.green{color:var(--accent)}
.drawer-val.orange{color:var(--accent2)}
.drawer-val.blue{color:#5599ff}
#drawerToggleBtn{
  position:fixed;bottom:12px;left:50%;transform:translateX(-50%);
  z-index:940;background:rgba(8,10,13,.9);border:1px solid var(--accent);
  border-radius:20px;padding:7px 20px;font-family:var(--mono);font-size:.6rem;

```

```

    color:var(--
accent);cursor:pointer;backdrop-
filter:blur(10px);
    transition:all .2s;white-space:nowrap;
}

#drawerToggleBtn:hover{background:rgba(0,229,
160,.1)}

#sensorDrawer.open ~ #toast{position:fixed;bottom:20px;left:50%;tr-
ansform:translateX(-
#drawerToggleBtn{bottom:calc(env(safe-area-
inset-bottom,0px) + 160px)}

#sidePanel{position:fixed;top:62px;right:10px
;z-index:900;width:200px;display:flex;flex-
direction:column;gap:8px;pointer-
events:none;}

    .side-
card{background:rgba(8,10,13,0.92);border:1px
solid var(--border);border-
radius:10px;padding:12px 14px;backdrop-
filter:blur(10px);pointer-events:all;}

    .side-title{font-family:var(--
display);font-size:.55rem;letter-
spacing:3px;color:var(--accent);margin-
bottom:10px;text-transform:uppercase}

    .side-row{display:flex;justify-
content:space-between;align-
items:center;margin-bottom:6px;font-
size:.65rem}

    .side-key{color:var(--textdim)} .side-
val{color:var(--text);font-family:var(--
display);font-size:.68rem}

    .legend-
card{background:rgba(8,10,13,0.92);border:1px
solid var(--border);border-
radius:10px;padding:12px 14px;backdrop-
filter:blur(10px)}

    .legend-title{font-size:.55rem;letter-
spacing:2px;color:var(--textdim);margin-
bottom:8px;font-family:var(--display)}

    .legend-grad{height:10px;border-
radius:5px;background:linear-gradient(to
right,#2266ff,#00ccff,#00ff88,#ffff00,#ff6600
,#ff0033);margin-bottom:5px}

    .legend-labels{display:flex;justify-
content:space-between;font-
size:.55rem;color:var(--textdim)}

#toast{position:fixed;bottom:20px;left:50%;tr-
ansform:translateX(-
50%);background:rgba(14,17,23,0.95);border:1px
solid var(--accent);border-
radius:8px;padding:8px 18px;font-
size:.7rem;color:var(--accent);z-
index:2000;pointer-
events:none;opacity:0;transition:opacity
.3s;backdrop-filter:blur(10px);white-
space:nowrap;}

#toast.show{opacity:1}

#gpsOverlay{position:fixed;top:52px;left:0;ri-
ght:0;bottom:0;background:rgba(8,10,13,0.85);
z-index:800;display:flex;flex-
direction:column;align-items:center;justify-
content:center;backdrop-
filter:blur(4px);pointer-
events:none;transition:opacity .5s;}

#gpsOverlay.hidden{opacity:0;pointer-
events:none}

.gps-
spinner{width:48px;height:48px;border:3px
solid var(--border);border-top-color:var(--
accent);border-radius:50%;animation:spin 1s
linear infinite;margin-bottom:16px}

@keyframes
spin{to{transform:rotate(360deg)}}

.gps-text{font-family:var(--display);font-
size:.8rem;color:var(--accent);letter-
spacing:3px;margin-bottom:8px}

.gps-sub{font-size:.65rem;color:var(--
textdim);letter-spacing:1px}

```

```

    .gps-sats{font-size:.7rem;color:var(--
accent2);margin-top:6px;letter-spacing:1px}
</style>
</head>
<body>

<div id="topbar">
  <div class="tb-logo">W-MetalDetector
<span>MAP</span><div class="dot"></div></div>
  <div class="tb-stats">
    <div class="tb-signal">
      <div class="tb-stat-lbl">Signal</div>
      <div class="tb-stat-val orange"
id="tbSig">0.000V</div>
      <div class="tb-sig-bar"><div class="tb-
sig-fill" id="tbSigBar"
style="width:0%"></div></div>
    </div>
    <div class="tb-stat">
      <div class="tb-stat-lbl">Sats</div>
      <div class="tb-stat-val" id="tbSats">-
</div>
    </div>
    <div class="tb-stat">
      <div class="tb-stat-lbl">Coords</div>
      <div class="tb-stat-val green"
id="tbCoords">Searching...</div>
    </div>
    <div class="tb-stat">
      <div class="tb-stat-lbl">Points</div>
      <div class="tb-stat-val green"
id="tbPts">0</div>
    </div>
    <div class="tb-stat">
      <div class="tb-stat-lbl">Mode</div>
      <div class="tb-stat-val green"
id="tbSeedMode">GPS</div>
    </div>
    <div class="tb-stat">
      <div class="tb-stat-lbl">GPS</div>
      <div class="tb-stat-val red"
id="tbGps">NO FIX</div>
    </div>
  </div>
  <div class="tb-actions">
    <button class="tb-btn" id="btnFollow"
onclick="toggleFollow()"> FOLLOW</button>
    <button class="tb-btn"
onclick="exportPoints()">↓ JSON</button>
    <button class="tb-btn danger"
onclick="clearPoints()">🗑️ CLEAR</button>
  </div>
  <div id="map"></div>
  <div id="sensorDrawer">
    <div class="drawer-handle"
onclick="toggleDrawer()">
      <div class="drawer-handle-title">📶
SENSORS & GPS</div>
      <div class="drawer-handle-arrow">▲</div>
    </div>
    <div class="drawer-grid">
      <div class="drawer-stat">
        <div class="drawer-lbl">Signal</div>
        <div class="drawer-val orange"
id="dSig">0.000V</div>
      </div>
      <div class="drawer-stat">
        <div class="drawer-lbl">GPS</div>
        <div class="drawer-val red"
id="dGps">NO FIX</div>
      </div>
      <div class="drawer-stat">
        <div class="drawer-lbl">Satellites</div>
        <div class="drawer-val blue"
id="dSats">-</div>
      </div>
    </div>
  </div>

```

```

    <div class="drawer-lbl">Points</div>
    <div class="drawer-val green"
id="dPts">0</div>
  </div>
  <div class="drawer-stat">
    <div class="drawer-lbl">Temp</div>
    <div class="drawer-val" id="dTemp">--
°C</div>
  </div>
  <div class="drawer-stat">
    <div class="drawer-lbl">Humidity</div>
    <div class="drawer-val blue"
id="dHum">--%</div>
  </div>
  <div class="drawer-stat">
    <div class="drawer-lbl">Soil</div>
    <div class="drawer-val orange"
id="dSoil">--%</div>
  </div>
  <div class="drawer-stat">
    <div class="drawer-lbl">Distance</div>
    <div class="drawer-val" id="dDist">---
</div>
  </div>
</div>
<button id="drawerToggleBtn"
onclick="toggleDrawer()">▲ SENSORS</button>

<div id="sidePanel">
  <div class="side-card">
    <div class="side-title">📊
Статистика</div>
    <div class="side-row"><span class="side-
key">Макс. сигнал</span><span class="side-val
orange" id="statMax">--</span></div>
    <div class="side-row"><span class="side-
key">Сер. сигнал</span><span class="side-val"
id="statAvg">--</span></div>
    <div class="side-row"><span class="side-
key">Мін. сигнал</span><span class="side-val
green" id="statMin">--</span></div>
    <div class="side-row" style="margin-
top:6px;padding-top:6px;border-top:1px solid
var(--border)">
      <span class="side-key">Покрита
площа</span><span class="side-val"
id="statArea">--</span>
    </div>
  </div>
  <div class="legend-card">
    <div class="legend-title">Інтенсивність
сигналу</div>
    <div class="legend-grad"></div>
    <div class="legend-
labels"><span>0.22V</span><span>1.4V</span><s
pan>2.6V</span></div>
  </div>
</div>
<div id="gpsOverlay">
  <div class="gps-spinner"></div>
  <div class="gps-text">ПОШУК GPS</div>
  <div class="gps-sub">Очікуємо фіксацію
супутників</div>
  <div class="gps-sats"
id="overlayStats">Satellites: 0</div>
</div>
<div id="toast"></div>
<script>
  var map = L.map('map', { zoomControl: false
}).setView([49.4220, 26.9870], 15);
  L.tileLayer('https://{s}.tile.openstreetmap.o
rg/{z}/{x}/{y}.png', {
    maxZoom: 21, attribution: '@ <a
href="https://openstreetmap.org">OSM</a>'
  }).addTo(map);
</script>

```

```

L.control.zoom({ position: 'bottomright'
}).addTo(map);

var posMarker = null, dotLayer =
L.LayerGroup().addTo(map);
var lastCount = -1, firstFix = true,
following = true, allPoints = [];

map.on('dragstart', function() {
following=false;
document.getElementById('btnFollow').classList.remove('active'); });

function toggleFollow(){
following=!following;

document.getElementById('btnFollow').classList.toggle('active', following);

if(following&&posMarker)map.setView(posMarker
.getLatLng(),map.getZoom());
showToast(following?'📍 Слідкування
увімкнено':'🔒 Слідкування вимкнено');
}
var toastTimer;
function showToast(msg){var
t=document.getElementById('toast');t.textContent=msg;t.classList.add('show');clearTimeout(toastTimer);toastTimer=setTimeout(=>t.classList.remove('show'),2500);}

function clearPoints(){
if(!confirm('Очистити всі точки на
карті?'))return;

fetch('/clear').then(=>{dotLayer.clearLayers();allPoints=[];lastCount=0;updateStats([]);
showToast('🗑️ Точки очищено');});
}

function intensityToColor(t){
var
stops=[[0,[34,102,255]],[0.2,[0,200,255]],[0.4,[0,255,136]],[0.6,[255,255,0]],[0.8,[255,100,0]],[1.0,[255,0,51]]];
t=Math.max(0,Math.min(1,t));
for(var i=0;i<stops.length-1;i++){
if(t>=stops[i][0]&&t<=stops[i+1][0]){
var f=(t-stops[i][0])/(stops[i+1][0]-stops[i][0]);
var c0=stops[i][1],c1=stops[i+1][1];
return
'rgb('+Math.round(c0[0]+f*(c1[0]-c0[0]))+', '+Math.round(c0[1]+f*(c1[1]-c0[1]))+', '+Math.round(c0[2]+f*(c1[2]-c0[2]))+')';
}
}
return 'rgb(255,0,51)';
}

function updateStats(pts){
if(!pts.length){['statMax','statAvg','statMin','statArea'].forEach(id=>document.getElementById(id).textContent='-');return;}
var vs=pts.map(p=>p.i*(2.6-0.22)+0.22);
var
mx=Math.max(...vs),mn=Math.min(...vs),av=vs.reduce((a,b)=>a+b,0)/vs.length;

document.getElementById('statMax').textContent=mx.toFixed(2)+'V';
document.getElementById('statAvg').textContent=av.toFixed(2)+'V';
document.getElementById('statMin').textContent=mn.toFixed(2)+'V';
if(pts.length>1){

```



```

fetch('/mapdata').then(r=>r.json()).then(d=>{
    var sigV=parseFloat(d.signal);
    var
sigPct=Math.max(0,Math.min(100,(sigV-
0.22)/(2.6-0.22)*100));
    var
sigEl=document.getElementById('tbSig');
    sigEl.textContent=sigV.toFixed(3)+'V';
    sigEl.className='tb-stat-val
'+(sigV>1.0?'red':sigV>0.5?'orange':'green');
document.getElementById('tbSigBar').style.wid
th=sigPct+'%';

    var sats=d.sats;
    var
satsEl=document.getElementById('tbSats');
    satsEl.textContent=sats;
    satsEl.className='tb-stat-val
'+(sats>=6?'green':sats>=3?'warn':'red');

document.getElementById('tbPts').textContent=
d.point_count;

    var          seedModeEl
document.getElementById('tbSeedMode');
    if (d.seedMode) {
        seedModeEl.textContent = 'SEED 🌱';
        seedModeEl.className = 'tb-stat-val
orange';
    } else {
        seedModeEl.textContent = 'GPS';
        seedModeEl.className = 'tb-stat-val
green';
    }

    var
gpsEl=document.getElementById('tbGps');

    var
overlay=document.getElementById('gpsOverlay')
;
document.getElementById('overlayStats').textC
ontent='Satellites: '+sats;

    if(d.valid || d.seedMode){
        if (d.seedMode) {
            gpsEl.textContent='SEED';gpsEl.className='tb-
stat-val orange';
        } else {
            gpsEl.textContent='FIX
√';gpsEl.className='tb-stat-val green';
        }
        overlay.classList.add('hidden');
        var
lat=parseFloat(d.lat),lon=parseFloat(d.lon);
        if (!d.seedMode) {

document.getElementById('tbCoords').textConte
nt=lat.toFixed(5)+' , '+lon.toFixed(5);

            if(!posMarker){posMarker=L.marker([lat,lon],{
            icon:posIcon,zIndexOffset:1000}).addTo(map).b
            indPopup('<b style="color:#00e5a0">Current
            position</b>');}
            else{posMarker.setLatLng([lat,lon]);}

            if(firstFix){map.setView([lat,lon],17);firstF
            ix=false;}

            if(following&&posMarker)map.panTo([lat,lon],{
            animate:true,duration:0.5});
        } else {

document.getElementById('tbCoords').textConte
nt='SEED MODE';
        }
}

```

```

    } else {
        gpsEl.textContent='NO
FIX';gpsEl.className='tb-stat-val red';
        overlay.classList.remove('hidden');

document.getElementById('tbCoords').textConte
nt='Searching...';
    }

document.getElementById('dSig').textContent=s
igV.toFixed(3)+'V';

document.getElementById('dPts').textContent=d
.point_count;

document.getElementById('dSats').textContent=
sats;
    if(d.temp!==null&&d.temp!==undefined){

document.getElementById('dTemp').textContent=
d.temp.toFixed(1)+'°C';

document.getElementById('dHum').textContent=(
d.hum!==null?d.hum.toFixed(0):'--')+ '%';
    }
    if(d.soil!==undefined)
document.getElementById('dSoil').textContent=
d.soil+'%';
    if(d.dist!==undefined){

document.getElementById('dDist').textContent=
d.dist<0?'---':d.dist.toFixed(0)+' cm';
    }
    var
dGpsEl=document.getElementById('dGps');
    if(d.valid||d.seedMode){

dGpsEl.textContent=d.seedMode?'SEED':'FIX ✓';
    dGpsEl.className='drawer-val
'+(d.seedMode?'orange':'green');

    } else {
        dGpsEl.textContent='NO
FIX';dGpsEl.className='drawer-val red';
    }

if(d.point_count!==lastCount&&d.points){lastC
ount=d.point_count;allPoints=d.points;redrawP
oints(allPoints);}
    }).catch(=>{});
}

var drawerOpen=false;
function toggleDrawer(){
    drawerOpen=!drawerOpen;

document.getElementById('sensorDrawer').class
List.toggle('open',drawerOpen);

document.getElementById('drawerToggleBtn').te
xtContent=drawerOpen?'▼ SENSORS':'▲ SENSORS';
    var
h=document.getElementById('sensorDrawer').off
setHeight;

document.getElementById('map').style.bottom=d
rawerOpen?(h-32)+'px':'0';
}

function exportPoints(){
    var a=document.createElement('a');
    a.href='/export.json';
    a.download='wmd_points.geojson';

document.body.appendChild(a);a.click();docume
nt.body.removeChild(a);
}

poll();
setInterval(poll,3000);
</script>

```

```

</body>
</html>
)rawliteral";

bool serverRunning = false;

void registerRoutes() {
  server.on("/", []() {
    server.send(200, "text/html", isSTAMode ?
STA_HTML : AP_HTML);
  });
  server.on("/data",      handleData);
  server.on("/step",     handleStep);
  server.on("/cal",      handleCal);
  server.on("/reset",    handleReset);
  server.on("/diag",     handleDiag);
  server.on("/mapdata",  handleMapData);
  server.on("/clear",    handleClearPoints);
  server.on("/export.json",
handleExportJson);
}

void startWiFi(bool apMode) {
  isSTAMode = !apMode;

  WiFi.disconnect(true);
  WiFi.softAPdisconnect(true);
  delay(100);
  WiFi.mode(WIFI_OFF);
  delay(300);

  if (!isSTAMode) {
    WiFi.mode(WIFI_AP);
    delay(100);

    bool apOk = false;
    for (int i = 0; i < 3 && !apOk; i++) {
      apOk      =      WiFi.softAP(AP_SSID,
AP_PASSWORD);
      if (!apOk) {
        Serial.println("[WIFI] softAP attempt
" + String(i+1) + " failed, retrying...");
        delay(200);
      }
    }

    if (apOk) {
      Serial.println("[WIFI] AP Mode OK. IP:
" + WiFi.softAPIP().toString());
    } else {
      Serial.println("[WIFI] AP Mode FAILED
after 3 attempts!");
    }
  } else {
    WiFi.mode(WIFI_STA);
    delay(100);
    WiFi.begin(STA_SSID, STA_PASSWORD);
    Serial.println("[WIFI] STA Mode.
Connecting to: " + String(STA_SSID));
    if (useSeedData) {
      randomSeed(analogRead(34));
      generateSeedPoints();
      Serial.println("[SEED] Test mode
ACTIVE");
    } else {
      Serial.println("[GPS] Real GPS mode");
    }
  }

  if (!serverRunning) {
    registerRoutes();
    server.begin();
    serverRunning = true;
    Serial.println("[HTTP] Server started");
  }

  if (oledOk) {
    display.clearDisplay();
    display.setCursor(0, 0);
  }
}

```

```

    display.println(!isSTAMode ? "Mode: AP" :
"Mode: STA");
    display.println(!isSTAMode ? AP_SSID :
STA_SSID);
    if (isSTAMode) {
        display.println(useSeedData ? "SEED
MODE" : "Connecting...");
    } else {
        display.println("192.168.4.1");
    }
    display.display();
}

void initOLEDD() {
    Wire.begin(PIN_SDA, PIN_SCL);
    if (display.begin(SSD1306_SWITCHCAPVCC,
0x3C)) {
        oledOk = true;
        display.clearDisplay();
        display.setTextColor(SSD1306_WHITE);
        display.setTextSize(1);
        display.setCursor(0, 0);
        display.println("W-MetalDetector v14.0");
        display.println("Booting...");
        display.display();
        Serial.println("[OLED] OK");
    } else {
        Serial.println("[OLED] Not found!");
    }
}

void setup() {
    Serial.begin(115200);
    delay(200);
    Serial.println("\n[W-MetalDetector v14.0]
Boot");

    WiFi.mode(WIFI_OFF);
    WiFi.disconnect(true);
    delay(100);

    pinMode(PIN_CS, OUTPUT);
    digitalWrite(PIN_CS, HIGH);
    SPI.begin(PIN_SCLK, -1, PIN_MOSI, PIN_CS);
    SPI.setFrequency(1000000);
    SPI.setDataMode(SPI_MODE0);

    prefs.begin("wmd", false);
    currentLevel = prefs.getUChar("sens", 128);
    writePot(currentLevel);

    pinMode(PIN_BTN_UP, INPUT_PULLUP);
    pinMode(PIN_BTN_DOWN, INPUT_PULLUP);
    pinMode(PIN_BTN_OK, INPUT_PULLUP);
    pinMode(PIN_BTN_MENU, INPUT_PULLUP);
    Serial.println("[BTN] UP=15 DOWN=2 OK=19
MENU=0");

    initOLEDD();

    gpsSerial.begin(9600, SERIAL_8N1,
PIN_GPS_RX, PIN_GPS_TX);

    analogReadResolution(12);
    analogSetAttenuation(ADC_11db);

    dht.begin();
    Serial.println("[DHT] GPIO " +
String(PIN_DHT));

    pinMode(PIN_SOIL_PWR, OUTPUT);
    digitalWrite(PIN_SOIL_PWR, LOW);
    pinMode(PIN_SOIL_BTN, INPUT_PULLUP);
    Serial.println("[SOIL] BTN=GPIO" +
String(PIN_SOIL_BTN) + " PWR=GPIO" +
String(PIN_SOIL_PWR));

    pinMode(PIN_TRIG, OUTPUT);
    pinMode(PIN_ECHO, INPUT);
    digitalWrite(PIN_TRIG, LOW);

```

```

Serial.println("[SONIC] TRIG=" + float sig = readVoltageCached();
String(PIN_TRIG) + " ECHO=" + bool signalPresent = (sig >
String(PIN_ECHO)); TARGET_VOLTAGE);
bool hasGpsFix =
delay(300); gps.location.isValid() && (gps.location.age()
< 5000);
startWiFi(true);
static unsigned long lastPtMs = 0;
Serial.println("[HTTP] Ready. Mode=" + static bool wasAbove = false;
String(!isSTAMode ? "AP" : "STA"));
}
if (signalPresent && hasGpsFix) {
void loop() { unsigned long now = millis();
if (!pendingWiFiSwitch) { if (!wasAbove || (now - lastPtMs >=
pendingWiFiSwitch = false; 1000)) {
Serial.println("[WIFI] Switching → " + addGpsPoint(gps.location.lat(),
String(pendingWiFiApMode ? "AP" : "STA")); gps.location.lng(), sig);
startWiFi(pendingWiFiApMode); lastPtMs = now;
return; Serial.printf("[PT] +%.3fV soil=%d%%
T=%.1f D=%.1f pts=%d\n",
sig, sens.soilPct,
sens.temperature, sens.distanceCm,
pointCount);
}
} else {
wasAbove = true;
} else {
wasAbove = false;
}
}
server.handleClient();
if (isSTAMode) {
static unsigned long lastWifiCheck = 0;
if (millis() - lastWifiCheck >= 3000) {
lastWifiCheck = millis();
staConnected = (WiFi.status() ==
WL_CONNECTED);
static bool ipPrinted = false;
if (staConnected && !ipPrinted) {
Serial.println("[WIFI] Connected. IP:
" + WiFi.localIP().toString());
ipPrinted = true;
}
}
if (calState != CAL_IDLE && calState !=
CAL_ERROR) {
calTick();
}
flushNvsIfNeeded();
readDHT();
pollUltrasonic();
if (!useSeedData) {

```

```

    }
}

handleButtons();

static unsigned long lastOled = 0;
static bool wasCalibratingOled = false;
if (oledOk && millis() - lastOled >= 500) {
    lastOled = millis();
    if (currentScreen == SCR_MAIN) {
        static unsigned long lastMain = 0;
        if (millis() - lastMain >= 1000) {
lastMain = millis(); updateOLED(); }
        }
        else if (currentScreen == SCR_MENU_CAL) {
            bool calibrating = isCalibrating();
            if (calibrating || wasCalibratingOled)
{
                updateOLED();
            }
            wasCalibratingOled = calibrating;
        }
    }
}
}

```

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: Віталій КОЛЕСНИКОВ

Співавтор:

Назва: Програмно-технічний засіб керування дистанційним металошукачем на основі мікроконтролера ESP32

Експерт: Сергій ЛИСЕНКО

Підрозділ: Кафедра комп'ютерної інженерії та інформаційних систем

Коефіцієнт подібності 1: 3.25%

Коефіцієнт подібності 2: 0.94%

Мікропробіли: 3

Заміна букв: 1

Інтервали: 0

Білі знаки: 0

Дата створення звіту: 2026-05-19 17:53:35.0

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:

2026-05-19

Дата



Доцент Андрій Нічепорук

експерт

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

Максимальне співпадіння з одним документом 0.0%

Словники перевірки: en_US, ru_RU, ua_UA. Помилки в документах: 9%

ID: 271735 Назва: БКР Програмно-технічний засіб керування дистанційним металошукачем на основі мікроконтролера ESP32 Додано в БД: 2026-05-19 Автора: Віталій КОЛЕСНИКОВ Керівники: Сергій ЛИСЕНКО Консультанти: Опоненти:	Документ		Сумарний збіг по Базі Даних	
	Символи	Лексеми	Символи	Лексеми
	137314	967	2307 (2%)	33 (3%)

Джерело плагіату

ID	Опис	Наявність плагіату в документі	
		Символи	Лексеми

РЕЦЕНЗІЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

Дипломник: Колесніков Віталій Русланович

Тема: Програмно-технічний засіб керування дистанційним металошукачем на основі мікроконтролера ESP32

Спеціальність: 123 «Комп'ютерна інженерія»

Обсяг кваліфікаційної роботи:

Кількість листів креслень 4 Кількість сторінок записки 71

1. Короткий зміст роботи та прийнятих рішень: Метою роботи є розробка програмно-технічного засобу дистанційного керування металошукачем на базі мікроконтролера ESP32. Запропоновано комплексне рішення, яке переводить металошукач у розряд інтелектуальних систем. Прийняті рішення включають використання мікроконтролера ESP32, цифрового потенціометра для програмної автоматичної калібровки, GPS-модуля для картографування та розробку веб-інтерфейсу для дистанційного керування через Wi-Fi

2. Висновок про відповідність роботи дипломному завданню: Робота повністю відповідає поставленому завданню. Усі вимоги вказані у завданні на кваліфікаційну роботу, а саме проєктування та програмно-апаратна реалізація засобу керування дистанційним металошукачем на основі мікроконтролера ESP32, виконані в повному обсязі.

3. Характеристика виконання кожного розділу, ступінь використання останніх досягнень науки і техніки і передових методів роботи: В першому розділі проведено глибокий аналіз предметної області, розглянуто принципи роботи існуючих приладів, проаналізовано підходи до дистанційного керування та обґрунтовано вибір мережевих технологій як найбільш перспективного напрямку, а також виконано чітку постановку задачі. В другому розділі спроєктовано модульну архітектуру системи, грамотно обґрунтовано вибір мікроконтролера ESP32 та допоміжних компонентів. Розроблено структурну та принципову електричну схеми, а також сучасні алгоритми обробки сигналу з використанням рухомого середнього та адаптивних порогів. В третьому

розділі успішно виконано апаратну та програмну реалізацію експериментального зразка. Проведено вичерпне тестування режимів роботи, підтверджено ефективність розробленої системи автоматичної калібровки та протестовано веб-інтерфейс з функцією експорту даних для картографії.

4. Позитивні сторони роботи: висока практична цінність роботи

5. Негативні сторони роботи: -

6. Оцінка графічного оформлення та пояснювальної записки роботи: Пояснювальна записка оформлена коректно згідно з чинними стандартами оформлення документації.

7. Відгук про роботу в цілому: робота виконана на високому науково-технічному рівні.

8. Інші зауваження: _____

9. Оцінка дипломної роботи: відмінно (А / 93)

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

Єфремова Л.О., к.т.н., доцент, зав. кафедрою АКТІР

“01” червня 2026 р.

 (підпис)

Зав. кафедри КПС
д-р. філософії Ользі ПАВЛОВІЙ

Віталій КОЛЄСНІКОВ

ПІБ здобувача вищої освіти

ФІТ, 3 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



РІШЕННЯ ЕКСПЕРТНОЇ КОМІСІЇ

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

Назва кваліфікаційної роботи Програмно-технічний засіб керування дистанційним металошукачем на основі мікроконтролера ESP32
Автор Віталій КОЛЕСНИКОВ
Освітня програма Комп'ютерна інженерія та програмування
Рівень вищої освіти перший (бакалаврський)
Спеціальність 123 Комп'ютерна інженерія
Науковий керівник: д.т.н., проф Сергій ЛИСЕНКО

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

Сумарний обсяг всіх запозичень, визначений системою виявлення збігів/ ідентичності/схожості StrikePlagiarism, складає 3,25%; та системою Anti-Plagiarism складає 0%, 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

Сергій ЛИСЕНКО
Ім'я, ПРІЗВИЩЕ