

Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет Автоматики і комп'ютеризованих технологій
(повна назва)

Кафедра Комп'ютерно-інтегрованих технологій, автоматизації та
роботехніки
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА
Пояснювальна записка

Рівень вищої освіти перший (бакалаврський)
Розробка системи автоматизації для відстеження та відображення
місцезнаходження мобільного робота на карті місцевості
(тема)

Виконав:
студент IV курсу, групи АКТАКІТ-20-1
Шевченко О. О.
(прізвище, ініціали)

Спеціальність 151 – автоматизація та
комп'ютерно-інтегровані технології
(код і повна назва спеціальності)

Тип програми освітньо-професійна
Освітня програма автоматизація та
комп'ютерно-інтегровані технології
(повна назва освітньої програми)

Керівник доц. Сичова О. В.
(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри _____
(підпис)

Невлюдов І. Ш.
(прізвище, ініціали)

2024р.

Я, як студент ХНУРЕ, розумію і підтримую політику закладу із академічної доброчесності. Я не надавав і не одержував недозволену допомогу під час підготовки кваліфікаційної роботи. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело.

13.06.2024 р.



(підпис)

Шевченко О. О.

(ПІБ)

Харківський національний університет радіоелектроніки

Факультет Автоматики і комп'ютеризованих технологій
 Кафедра КІТАР
 Рівень вищої освіти перший (бакалаврський)
 Спеціальність 151 Автоматизація та комп'ютерно-інтегровані технології
 (код і повна назва)
 Тип програми освітньо-професійна
 Освітня програма Автоматизація та комп'ютерно-інтегровані технології
 (повна назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри _____

(підпис)

« _____ » _____ 20__ р.

ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

студентові Шевченку Олександрю Олеговичу
 (прізвище, ім'я, по батькові)

1. Тема роботи Розробка системи автоматизації для відстеження та відображення місцезнаходження мобільного робота на карті місцевості

затверджена наказом університету від 03 06 2024 р. № 544 Ст

2. Термін подання студентом роботи до екзаменаційної комісії 17 06 2024 р.

3. Вихідні дані до роботи Середовище програмування Visual Studio

Тип інтерфейсу – графічний

Мова програмування: C#, C++, C

Сервіс ThingSpeak

4. Перелік питань, що потрібно опрацювати в роботі _____

Аналіз технічного завдання

Проектування архітектури модуля з мобільним роботом та алгоритмів

його роботи

Розробка апаратного забезпечення модуля відстеження мобільного робота

Розробка програмного забезпечення модуля відображення мобільного

робота

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (п.5 включається до завдання за рішенням випускної кафедри) Демонстрацію та матеріал представлений у форматі презентації PowerPoint (*.ppt) = 17с. формату А4


6. Консультанти розділів роботи (п.6 включається до завдання за наявності консультантів згідно з наказом, зазначеним у п.1)

Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Терміни виконання етапів роботи	Примітка
1	Видача теми проекту, узгодження і затвердження	22.12.2023	виконано
2	Аналіз вимог технічного завдання, аналіз способів відстеження місцезнаходження, аналіз аналогічних модулів	16.04.2024	виконано
3	Вибір електронних компонентів для апаратного модуля	10.05.2024	виконано
4	Розробка макету апаратного модуля	12.05.2024	виконано
5	Розробка програми для апаратного модуля	20.05.2024	виконано
6	Вибір програмних компонентів для програмного модуля	21.05.2024	виконано
7	Розробка програмного модуля	25.06.2024	виконано

Дата видачі завдання 22 грудня 2023 р.

Студент 
(підпис)

Керівник роботи доц. Сичова О. В.
(підпис) (посада, прізвище, ініціали)

РЕФЕРАТ

Пояснювальна записка: 77с., 59 рис., 3 дод., 19 джерел.

МОБІЛЬНИЙ РОБОТ, КАРТА МІСЦЕВОСТІ, СИСТЕМА, АПАРАТНИЙ МОДУЛЬ, ПРОГРАМНИЙ МОДУЛЬ, ІНТЕРФЕЙС КОРИСТУВАЧА, C#, VISUAL STUDIO, SERVIC THINGSPEAK.

Метою кваліфікаційної роботи є підвищення точності визначення та відображення місцезнаходження мобільного робота на карті місцевості за допомогою автоматизованої системи, яка використовує дані з сенсорів. Ця система повинна інтегруватися з картографічними сервісами для отримання актуальних карт та надавати інтуїтивно зрозумілий інтерфейс для взаємодії з користувачем.

Об'єктом розробки є процес відстеження та відображення місцезнаходження мобільного робота, що включає як апаратну, так і програмну складові.

Предметом розробки є конкретні технічні рішення, алгоритми, програмні засоби та апаратні компоненти, що забезпечують точне визначення та відображення місцезнаходження мобільного робота на карті місцевості.

Виконано аналіз технічного завдання де визначені основні вимоги до системи: точне визначення місцезнаходження, підключення до картографічних сервісів та створення користувацького інтерфейсу.

Розроблено та проведено комплексне тестування апаратного та програмного модулів, що підтвердило високу точність і надійність відстеження та відображення місцезнаходження мобільного робота в реальному часі.

Отримані результати відповідають переліку цілей сталого розвитку, зокрема цілі 9 промисловість, інновації та інфраструктура.

ABSTRACT

Explanatory note: 77 p., 59 fig., 3 adj., 19 sources.

MOBILE ROBOT, LOCATION MAP, SYSTEM, HARDWARE MODULE, SOFTWARE MODULE, USER INTERFACE, C#, VISUAL STUDIO, THINGSPEAK SERVICE.

The purpose of the qualification work is to increase the accuracy of determining and displaying the location of a mobile robot on a terrain map using an automated system that uses data from sensors. This system should integrate with cartographic services to obtain up-to-date maps and provide an intuitive user interface.

The object of development is the process of tracking and displaying the location of a mobile robot, which includes both hardware and software components.

The subject of the development is specific technical solutions, algorithms, software tools and hardware components that ensure accurate determination and display of the location of the mobile robot on the terrain map.

An analysis of the technical task was carried out, where the main requirements for the system were determined: accurate location determination, connection to cartographic services and creation of a user interface.

Comprehensive testing of hardware and software modules was developed and conducted, which confirmed the high accuracy and reliability of tracking and displaying the location of the mobile robot in real time.

The obtained results correspond to the list of sustainable development goals, in particular, goal 9 industry, innovation and infrastructure.

ЗМІСТ

Перелік умовних скорочень.....	9
Вступ.....	10
1 Аналіз технічного завдання	12
1.1 Аналіз вимог технічного завдання	12
1.2 Аналіз способів відстеження місцезнаходження	13
1.3 Інтеграція модуля з мобільним роботом.....	18
1.4 Аналіз аналогічних модулів	21
1.5 Постановка задач	25
2 Проектування архітектури взаємодії системи з мобільним роботом та алгоритмів його роботи	26
2.1 Розробка архітектури взаємодії системи з мобільним роботом	26
2.2 Розробка структурної схеми модуля	28
2.3 Опис алгоритмів роботи модуля	28
2.4 Опис протоколу взаємодії між модулем та мобільним роботом.....	30
2.5 Висновки до другого розділу.....	32
3 Розробка апаратного забезпечення модуля відстеження мобільного робота...	33
3.1 Вибір електронних компонентів модуля.....	33
3.2 Віртуальне підключення компонентів	37
3.3 Практичне підключення компонентів	40
3.4 Висновки до третього розділу	54
4 Розробка програмного забезпечення модуля відображення мобільного робота	55
4.1 Вибір програмних компонентів.....	55
4.2 Опис програмних функцій та модулів	57
4.3 Тестування програми та опис результату роботи модуля	63
4.4 Висновки до четвертого розділу	70
5 Охорона праці	71
5.1 Технічна безпека	71

5.2 Організаційна безпека.....	72
Висновки	73
Перелік посилань	75
Додаток А Програмний код для інтерфейсу користувача	78
Додаток Б Програмний код для апаратної частини.....	88
Додаток В Демонстраційний матеріал.....	94

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

- ККД – коефіцієнт корисної дії;
- НІРР – нормалізований індекс різниці рослинності;
- СКД – середнє квадратичне відхилення;
- API – application programming interface;
- FDM – fused deposition modeling;
- GNSS – global navigation satellite system;
- GPS – global positioning system;
- GUI – graphical user interface;
- GSM – groupe special mobile;
- HTTPS – hyper text transfer protocol secure;
- IDE – sntegrated development environment;
- IoT – internet of things;
- MPP – map pilot pro;
- SSL – secure sockets layer;
- TLS – transport layer security;
- URI – uniform resource identifier.

ВСТУП

На сьогоднішній день, розробка автоматизованої системи відстеження та відображення місцезнаходження мобільного робота на карті місцевості має значення в сучасному технологічному просторі. Ця ініціатива не лише сприятиме ефективному управлінню ресурсами та оптимізації логістики, а й підвищить безпеку праці, моніторинг екологічних показників та підтримку автономних систем. Враховуючи розвиток індустрії та потребу у точних даних для прийняття рішень, така система може стати необхідним інструментом у різних сферах, від виробництва до логістики.

Технологічний прогрес завжди спричиняє зміни в промисловості. Для динамічних, оптимізованих у реальному часі та самоорганізованих промислових процесів ключовими є отримання інформації та її обробка [1 - 2].

У виробництві особливого значення набуває використання мобільних роботів для об'єднання промислового обладнання в один інтегрований виробничий процес. Завдання використання роботів у середовищі людини дуже складне. По-перше, необхідно прокласти шлях для пересування мобільних засобів серед об'єктів, положення яких динамічно змінюються. По-друге, необхідно забезпечити безпеку працівників, які обслуговують обладнання, за рахунок використання інтелектуальних датчиків [3 - 5].

Один з головних викликів, що стоять перед розробниками автоматизованих модулів відстеження, полягає у забезпеченні безперебійної роботи систем у різних умовах експлуатації. Це включає необхідність адаптації до змінних погодних умов, перешкод на місцевості та потенційних загроз з боку кібербезпеки. Однак, попри ці виклики, перспективи використання таких модулів є надзвичайно обнадійливими. У поєднанні з іншими технологіями, такими як 5G, ці модулі можуть забезпечити ще більш швидку і надійну передачу даних, що відкриває нові горизонти для їх застосування у різних сферах.

Метою кваліфікаційної роботи є підвищення точності визначення і відображення положення мобільного робота на карті місцевості шляхом створення автоматизованої системи, яка використовує інформацію від сенсорів. Ця система повинен мати можливість інтеграції з картографічними сервісами для отримання актуальних карт та забезпечувати зручний інтерфейс для користувача.

Об'єктом розробки є система, яка дозволяє відстежувати та відображати місцезнаходження мобільного робота, включаючи як апаратні, так і програмні компоненти.

Предметом розробки є конкретні технічні рішення, алгоритми, програмне забезпечення і апаратні складові, які забезпечують точне визначення та відображення положення мобільного робота на карті місцевості.

Для досягнення поставленої мети необхідно вирішити такі завдання:

- розробити алгоритми, які забезпечать точне визначення місцезнаходження мобільного робота на основі отриманих даних з сенсорів або зовнішніх пристроїв;
- підключитися до картографічних сервісів для отримання актуальних карт та можливості відображення на них місцезнаходження мобільного робота;
- створити інтерфейс, що дозволить користувачу взаємодіяти з модулем, налаштовувати його параметри та переглядати результати відстеження на карті;
- оформити кваліфікаційну роботу згідно ДСТУ 3008:2015 [6], а також з методичними вказівками з підготовки й оформлення кваліфікаційної роботи здобувачами першого (бакалаврського) рівня вищої освіти спеціальності 151 Автоматизація та комп'ютерно-інтегровані технології освітньої програми «Автоматизація та комп'ютерно-інтегровані технології» [7] й навчальним посібником з підготовки кваліфікаційної роботи бакалавра для здобувачів вищої освіти денної і заочної форм навчання спеціальності 151 [8].

1 АНАЛІЗ ТЕХНІЧНОГО ЗАВДАННЯ

1.1 Аналіз вимог технічного завдання

Розроблювальна автоматизована система повинен бути призначений для відслідковування руху мобільного робота та його відображення на карті місцевості в реальному часі.

Дане програмне забезпечення надасть можливість контролювати напрям руху мобільного робота, що дозволить точно визначати та прогнозувати маршрути, оптимізувати час переміщень та забезпечувати ефективне виконання завдань у визначеній місцевості.

Метою проєкту є підвищення точності визначення і відображення положення мобільного робота на карті місцевості шляхом створення автоматизованої системи, яка використовує інформацію від сенсорів.

Для досягнення мети необхідно вирішити такі задачі:

- створення алгоритмів, які забезпечать точне визначення місцезнаходження мобільного робота на основі отриманих даних з сенсорів або зовнішніх пристроїв;
- підключення до картографічних сервісів для отримання актуальних карт та можливості відображення на них місцезнаходження мобільного робота;
- розробка інтерфейсу, що дозволить користувачу взаємодіяти з модулем, налаштовувати його параметри та переглядати результати відстеження на карті;
- проведення випробувань модуля для перевірки його працездатності та відповідності вимогам;
- валідація результатів відстеження місцезнаходження для підтвердження їх точності та надійності.

Вимоги до розроблюваного автоматизованого система:

- система повинен забезпечувати точне визначення місцезнаходження мобільного робота з мінімальною похибкою;
- система повинна бути здатна оперативно реагувати на зміни місцезнаходження та відображати їх на карті у реальному часі;
- забезпечити можливість інтеграції з різними типами мобільних роботів та використання різних видів датчиків для отримання даних про місцезнаходження;
- система повинен мати зручний та інтуїтивно зрозумілий інтерфейс користувача для налаштування параметрів та відображення результатів;
- забезпечення стабільної роботи модуля без збоїв або помилок протягом тривалого часу роботи.

1.2 Аналіз способів відстеження місцезнаходження

Основні методи відстеження місцезнаходження мобільного робота:

- GNSS (Global navigation satellite systems);
- Мобільна мережа;
- Wi-Fi та Bluetooth.

Глобальна навігаційна супутникова система (GNSS) – це мережа супутників, що передають інформацію про час і орбітальну інформацію, яка використовується для навігації та вимірювань позиціонування. Робота полягає в тому, що супутники передають сигнали, які повідомляють, де вони перебувають у який час, і ця інформація використовується для визначення того, де ви знаходитесь у світі. За допомогою складної серії трилатераційних обчислень, технологія обчислює місцезнаходження на основі положення мобільного робота відносно щонайменше чотирьох супутників (рис. 1.1).

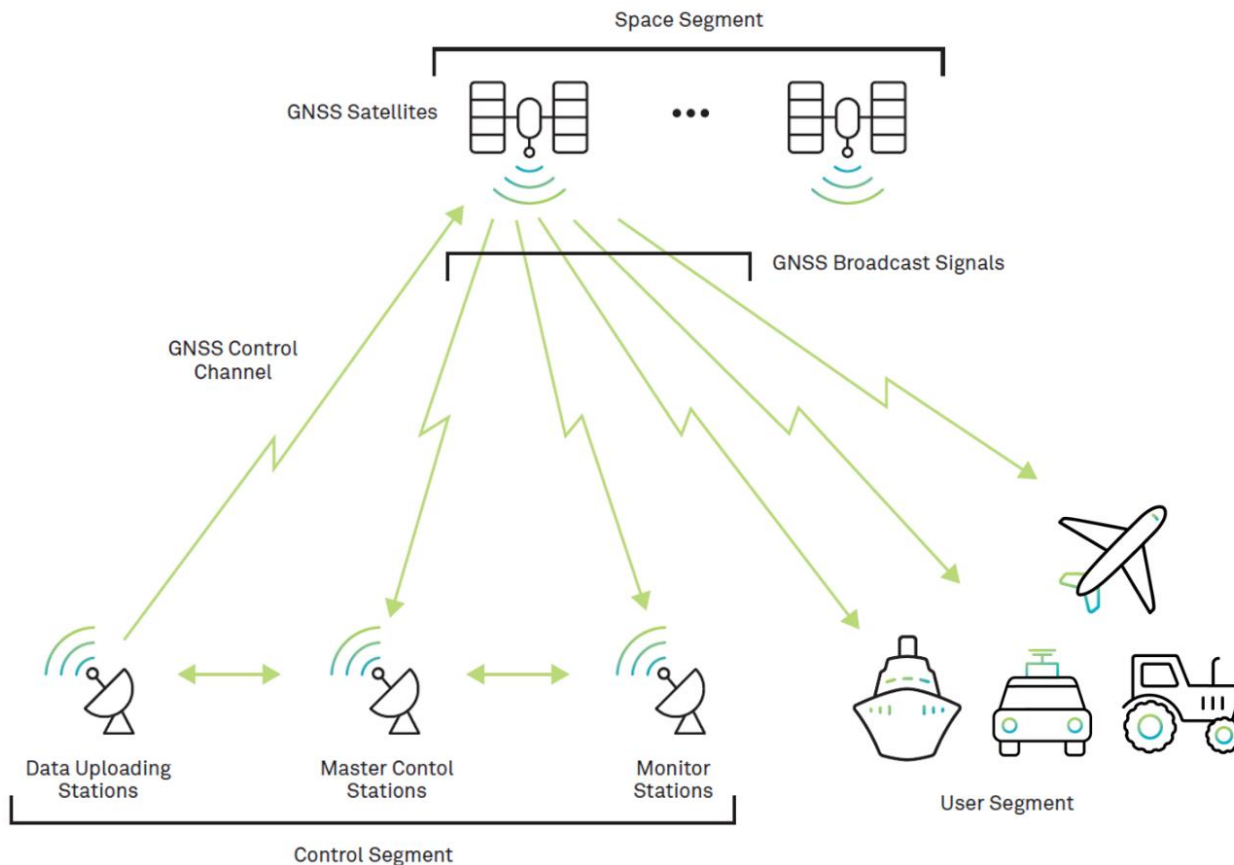


Рисунок 1.1 – Візуалізація роботи GNSS для місцезнаходження мобільного робота [9]

GNSS – це більше, ніж супутники, що обертаються навколо Землі. Кілька груп супутників, відомих як сузір'я, транслюють сигнали головним станціям керування та користувачам GNSS по всій планеті. Ці три сегменти – простір, контроль і користувач – усі вважаються частиною GNSS. Але найчастіше GNSS використовують для опису супутників у космосі.

Космічний сегмент описує сузір'я GNSS, що обертаються на висоті від 20 000 км до 37 000 км над Землею. Ці супутники транслюють сигнали, які визначають, який супутник передає, а також його час, орбіту та статус або стан. На орбіті знаходяться чотири основні угруповання – GPS, ГЛОНАСС, Галілей і БейДоу, а також дві регіональні системи QZSS і IRNSS, і кожною з них керує окрема країна [9].

Сегмент керування – це мережа головних станцій контролю, завантаження даних і моніторингу, розташованих по всьому світу.

Станції приймають сигнал супутника та порівнюють його розташування з моделями орбіти, які показують, де він має бути. Оператори цих станцій можуть контролювати положення супутників, щоб виправляти або змінювати їхні орбітальні шляхи, наприклад, якщо супутник дрейфував або його потрібно перемістити, щоб уникнути зіткнення з уламками. Цей процес, а також моніторинг працездатності супутника, забезпечує базову точність позиціонування GNSS.

Сегмент користувача включає обладнання, яке приймає супутникові сигнали та видає позицію на основі часу та орбітального розташування принаймні чотирьох супутників. Цей сегмент включає в себе антени користувача для ідентифікації та прийому якісних сигналів, а також високоточні приймачі та механізми позиціонування, які обробляють сигнали та усувають можливі помилки синхронізації.

У всіх сучасних мережах мобільного зв'язку оператор може обчислити, де знаходиться робот, якщо на роботі встановлена SIM-карта і він увімкнений, і знаходиться в мережі. Можливість зробити це залежить від того, як побудовано мобільну мережу, а сама технологія називається тріангуляцією.

Один із способів, яким оператор може скористатися, це спостереження за рівнями сигналу, що отримується різними вежами від конуретного робота. На основі цих даних оператор може обчислити, де повинен бути робот, щоб відповідати цим спостереженням (рис. 1.2). Це робиться за допомогою вимірювання кута парафії (Angle of Arrival, або AoA).

Точність, з якою оператор може визначити місцезнаходження мобільного робота, залежить від багатьох факторів, включаючи технологію, яку використовує оператор, а також кількість вишок стільникового зв'язку на даній ділянці. Зазвичай, маючи мінімум 3 вежі зв'язку, оператор може визначити місцезнаходження з точністю до кілометра. Для сучасних мереж та GSM (Groupe

Spécial Mobile) модулів, які встановлені на роботах, також використовується трилатерація. Зокрема, цей спосіб використовується там, де підтримується функція "locationInfo-r10". Ця функція надає звіт, що містить точні GPS координати мобільного робота.

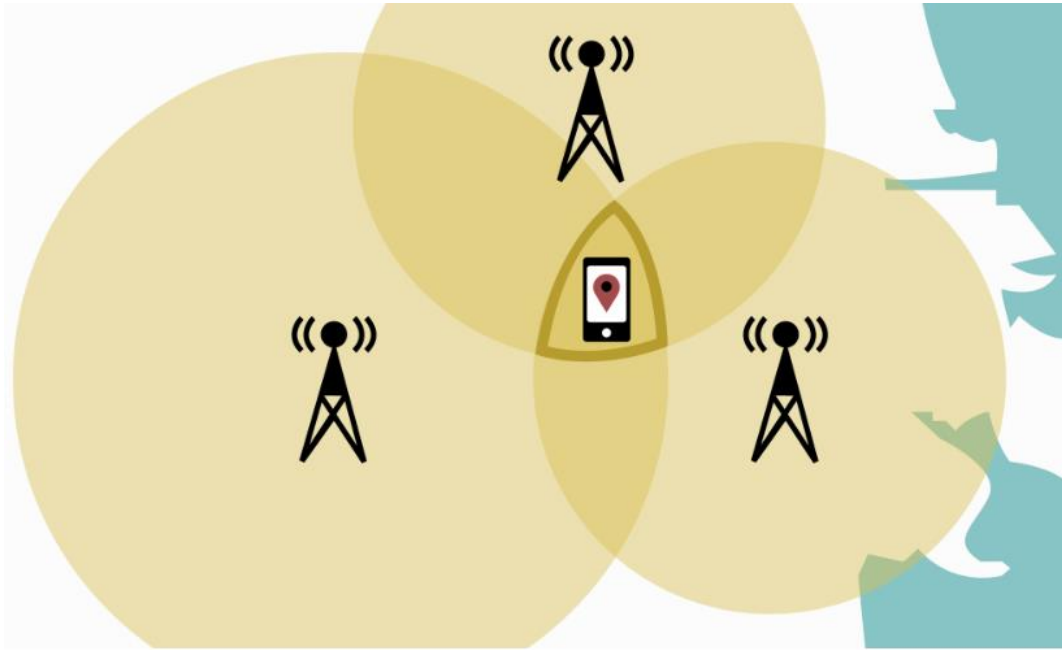


Рисунок 1.2 – Зображення методу місцезнаходження робота через мобільну мережу [10]

Не існує способів, які захищають від подібного виду відстеження, якщо GSM модуль увімкнено і він має SIM-картку то буде все одно передаватися сигнали в мережу мобільного оператора. Оператори також обмінюються один з одним інформацією про місцезнаходження робота, з якого на даний момент здійснюється підключення. Найчастіше ці дані менш точні, ніж інформація з кількох вишок зв'язку, але вони все одно можуть бути використані [10].

Сучасні мобільні роботи мають інші радіопередавачі, зазвичай вони також підтримують Wi-Fi і Bluetooth. Ці сигнали передаються з меншою потужністю, ніж сигнал мобільного зв'язку, і зазвичай можуть використовуватись лише на невеликій відстані (наприклад, в межах однієї кімнати або однієї будівлі, гаражі)

(рис. 1.3). Хоча хтось, використовуючи потужну антену, може виявити ці сигнали з несподівано великих відстаней.

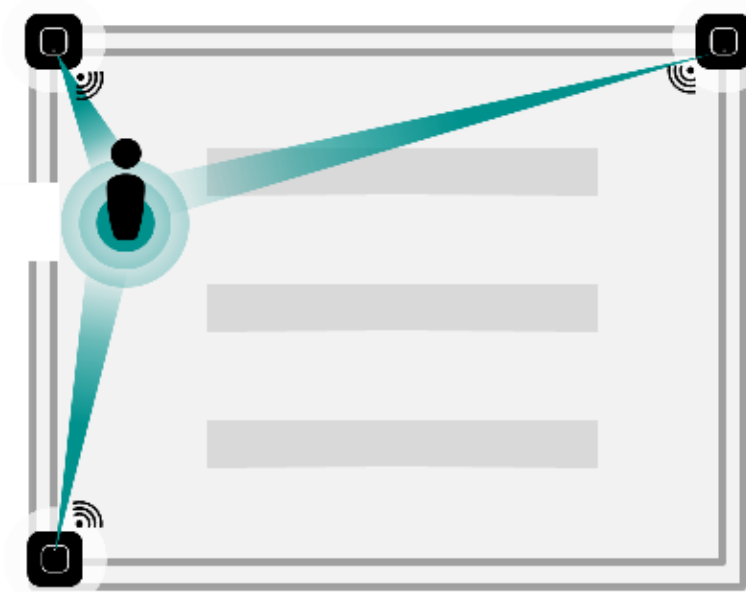


Рисунок 1.3 – Зображення підключення мобільного робота до Wi-Fi та Bluetooth модулів

Обидва ці види бездротових технологій використовують унікальний серійний номер пристрою, званий MAC-адресою, який може бачити будь-хто, хто приймає сигнал. У всіх випадках, коли Wi-Fi включений, звичайний Wi-Fi модуль, який встановлений на роботі, періодично передає «пробний запит», що включає MAC-адресу. Тим самим він повідомляє про свою присутність всім пристроям поблизу. Bluetooth модулі роблять щось подібне. Ці ідентифікатори традиційно використовуються як цінний інструмент для збору даних про переміщення мобільного робота та людей, що відстежують у місцях де будуть такі сигнали. Однак в останніх оновленнях, MAC-адреса з пробного запиту є випадковою, що значно ускладнює відстеження. Хоча сучасні Wi-Fi модулі робота при пробному запиті зазвичай змінюють адресу на випадкову, більшість із них все ще використовують постійну MAC-адресу для підключення до мереж.

Як наслідок, власники мережі можуть збирати дані про підключення та розуміти, чи підключалися ви раніше.

1.3 Інтеграція модуля з мобільним роботом

Для інтеграції автоматизованого модуля з мобільним роботом використовуються сервіси та платформи Internet of Thing (IoT).

Розглянемо найпопулярніші сервіси IoT, а саме:

- Adafruit IO;
- InfluxDB;
- Losant Enterprise IoT Platform;
- ThingSpeak.

Adafruit IO – це хмарний сервіс Інтернету речей (IoT), який дозволяє надсилати та зберігати дані з мобільного робота. У ньому можливо створювати власні "канали" для зберігання та візуалізації даних у реальному часі (рис. 1.4).

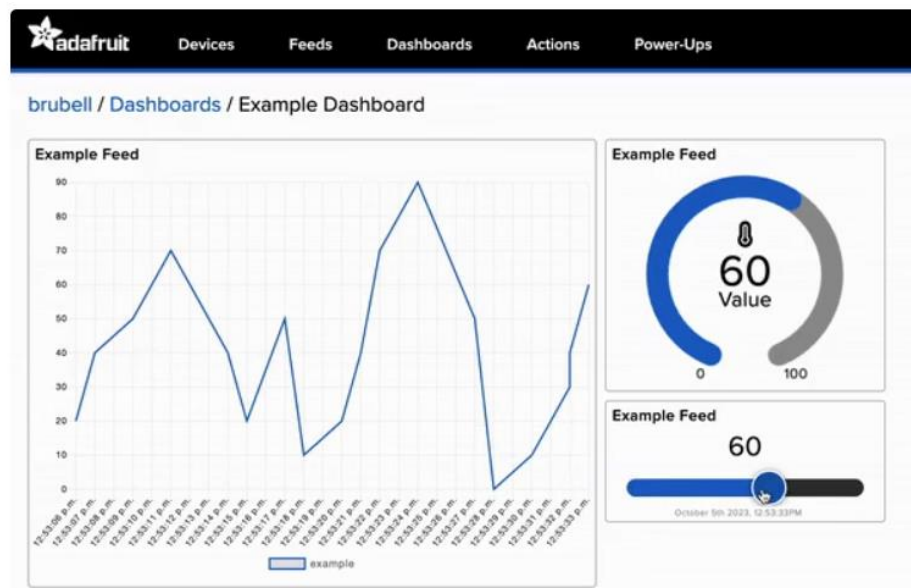


Рисунок 1.4 – Вигляд інтерфейсу сервісу Adafruit IO на якому відображаються отримані дані з мобільного робота

InfluxDB – це продуктивна, еластична платформа даних часових рядів, яка надає все необхідне для даних часових рядів, включаючи показники, події, журнали та трасування, дозволяє отримувати дані у режимі реального часу з необмеженою потужністю, а також зберігати, аналізувати та обробляти ці дані – усе в одній базі даних, має власну підтримку SQL, яка прискорює процес початку роботи (рис. 1.5).

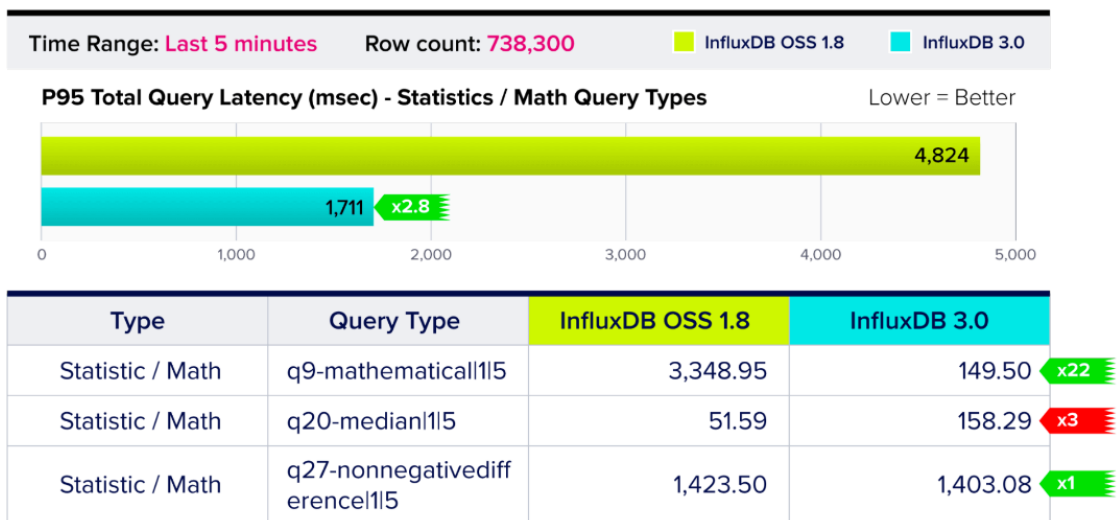


Рисунок 1.5 – Вигляд інтерфейсу сервісу InfluxDB отриманих даних

Losant Enterprise IoT Platform – це платформа для додатків, яка дозволяє підприємствам ефективно створювати додатки та рішення, які безпечно масштабуються для мільйонів пристроїв та роботів. Компоненти бездоганно працюють разом, щоб перетворювати дані в режимі реального часу з підключених і не підключених пристроїв у користувальницькі умови роботи та бізнес-інсайти. Платформа Losant надає інструменти для збору, зберігання, аналізу та візуалізації даних, а також для створення автоматизованих робочих процесів і додатків (рис. 1.6).

Losant забезпечує гнучкість та масштабованість, необхідні для успішного впровадження IoT-рішень у різних галузях, від виробництва та логістики до роздрібною торгівлі та охорони здоров'я. Платформа дозволяє підприємствам

підвищити операційну ефективність, зменшити витрати та створювати нові можливості для бізнесу.

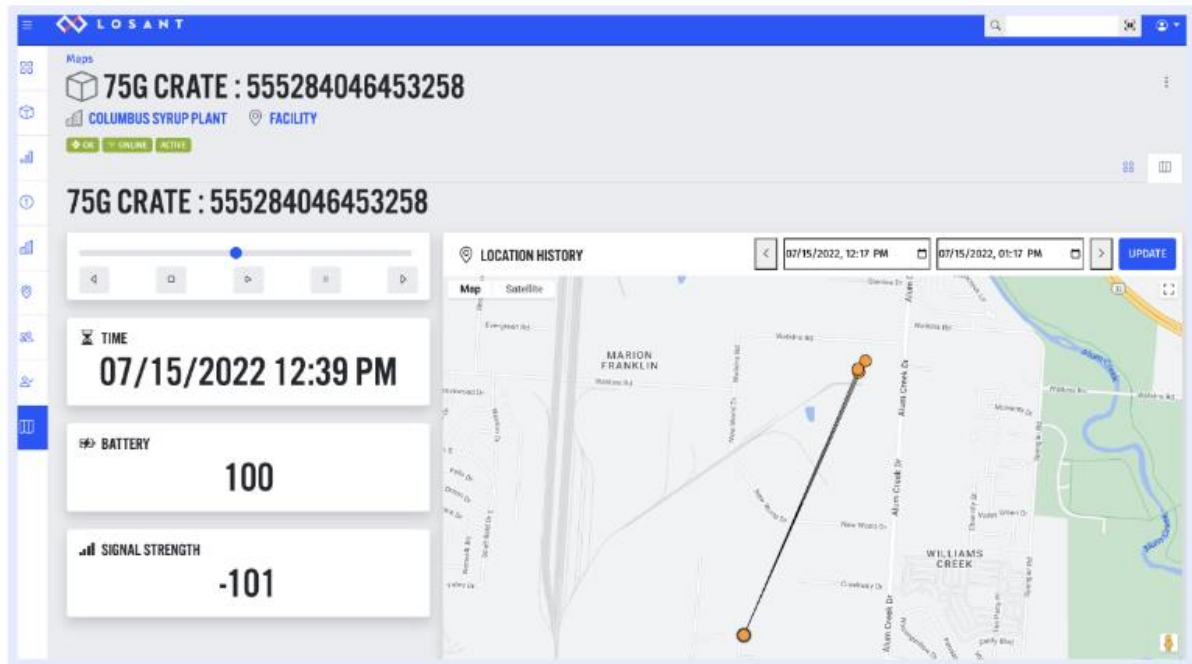


Рисунок 1.6 – Вигляд інтерфейсу сервісу Losant на якому відображаються отримані дані з мобільного робота

ThingSpeak – це служба аналітичної платформи IoT, яка дозволяє агрегувати, візуалізувати й аналізувати живі потоки даних у хмарі. це служба аналітичної платформи IoT, яка дозволяє агрегувати, візуалізувати й аналізувати живі потоки даних у хмарі. Надсилаючи дані до ThingSpeak з мобільного робота, є можливість створювати миттєву візуалізацію даних у реальному часі та надсилати сповіщення (рис. 1.7).

Платформа ThingSpeak пропонує різноманітні інструменти для обробки та аналізу даних, що дозволяє користувачам отримувати корисну інформацію з підключених пристроїв та робити обґрунтовані рішення на основі цих даних.

ThingSpeak є потужним інструментом для розробників, дослідників та інженерів, які працюють з IoT-проектами. Вона допомагає легко збирати,

аналізувати та використовувати дані, що сприяє підвищенню ефективності та продуктивності різних систем і процесів.

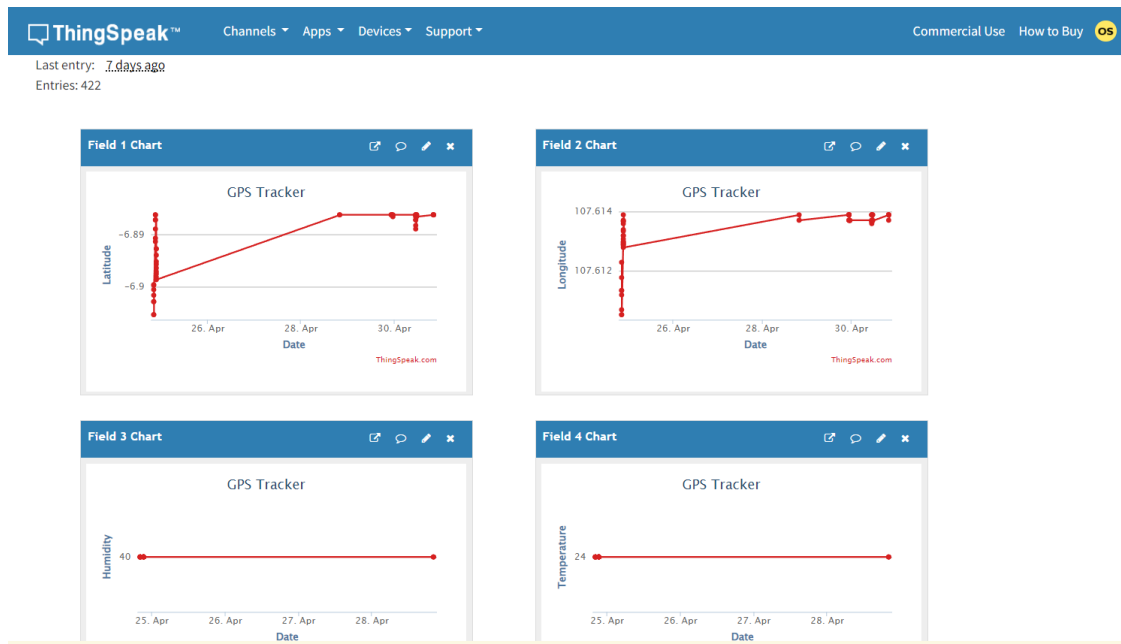


Рисунок 1.7 – Вигляд інтерфейсу сервісу ThingSpeak на якому відображаються отримані дані з мобільного робота

1.4 Аналіз аналогічних модулів

Для аналізу аналогічного модуля було обрано програмний продукт Map Pilot Pro (MPP) (рис. 1.8).

Map Pilot Pro – це додаток для мобільних та десктопних пристроїв, розроблений компанією Maps Made Easy. Це інструмент, який дозволяє користувачам створювати точні картографічні дані за допомогою мобільних роботів, а саме – безпілотних літальних апаратів.

Додаток дозволяє користувачам легко планувати маршрути польоту для дронів, забезпечуючи оптимальне покриття території. Користувачі можуть визначати параметри польоту, такі як висота, швидкість, перекриття кадрів, та маршрут, що дозволяє отримати максимально точні та детальні дані.

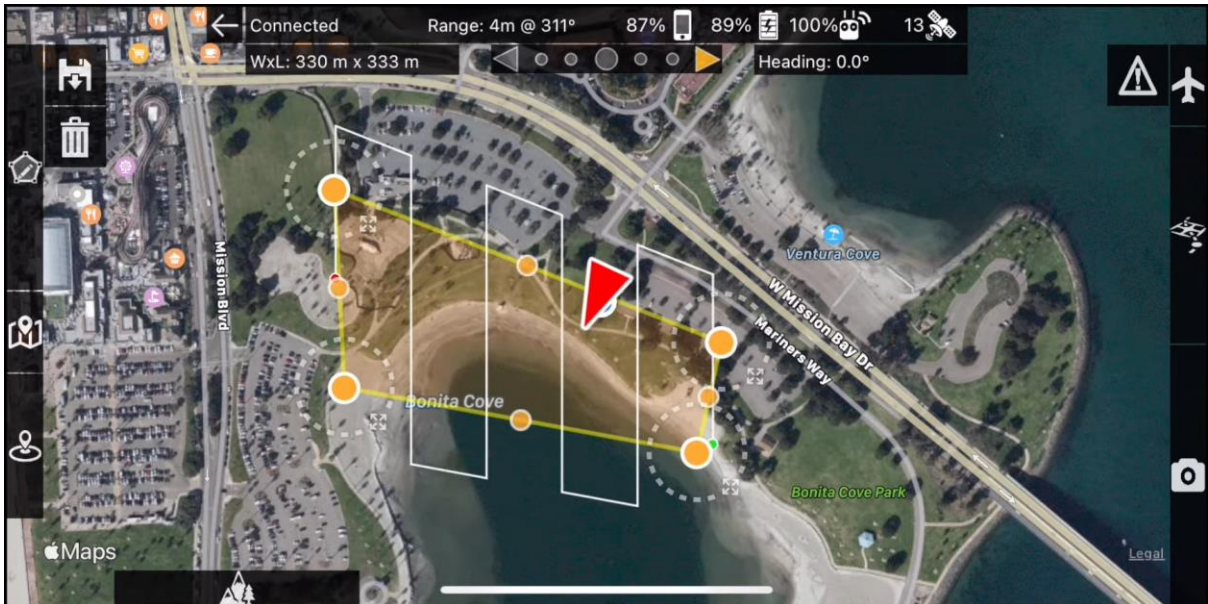


Рисунок 1.8 – Зображення місцезнаходження мобільного робота у програмі MPP [11]

MPP представляє новий погляд на карту, а саме: Google Maps чудова карта, але зображенням, як правило, їм бракує роздільної здатності для реальної роботи. Можна створити власні карти та отримати новий погляд на речі за допомогою точних зображень із високою роздільною здатністю (рис. 1.9).



Рисунок 1.9 – Порівняння Google Maps та карти MPP [11]

Додаток використовує найсучасніші методи створення своїх карт. Об'єднання кількох поглядів на територію під час прольоту мобільного робота – дрону дозволяє створити 3D-модель, на основі якої можна отримати неймовірно

точні ортофотокарти. Текстуровану 3D-модель можна завантажити та попередньо переглянути онлайн (рис. 1.10).

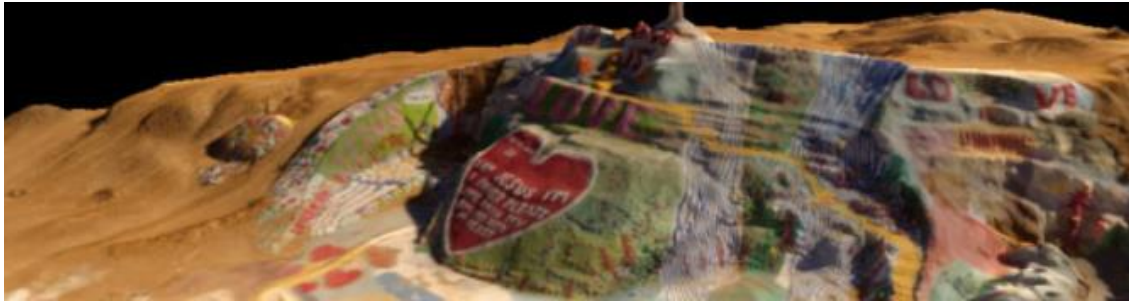


Рисунок 1.10 – Створена 3D-модель з фотографії заснятої роботом [11]

Веб-об'ємні вимірювання дозволяють швидко та легко вимірювати та відстежувати обсяги запасів (рис. 1.11). Не потрібно купувати дороге програмне забезпечення, і вимірювання можна проводити безпосередньо з веб-браузера. Об'ємні вимірювання легко надавати іншим і зберігати для відстеження прогресу з часом.

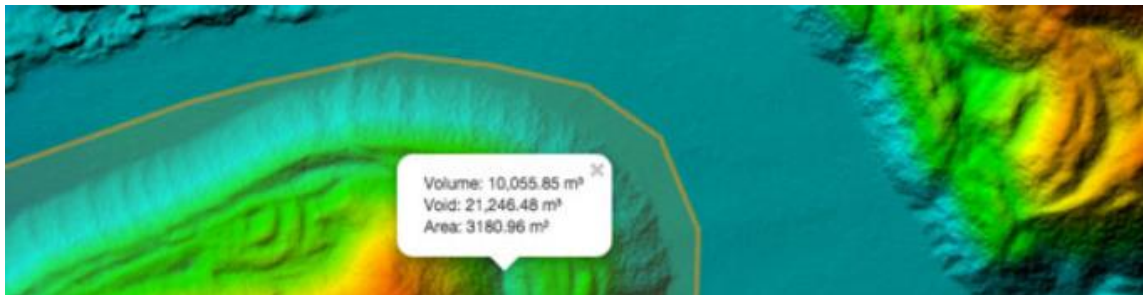


Рисунок 1.11 – Об'ємне вимірювання [11]

Компанія розробника пропонує найкращі у своєму класі додатки для iOS та Android, які допоможуть забезпечити правильне планування польоту мобільного робота. Можна визначити зону огляду, вибрати рівень деталізації та летіти за заданим курсом (рис. 1.12). Про все інше подбає додаток. За допомогою додатку легко планується дослідження великих територій, які потребують кількох батарей для дрону.



Рисунок 1.12 – Зображення побудованого шляху для мобільного робота [11]

Програма може виконувати обробку нормалізованого індексу різниці рослинності (NIPR). NIPR – це простий графічний індикатор, який можна використовувати для аналізу вимірювань дистанційного зондування та оцінки того, чи містить об’єкт спостереження здорову рослинність чи ні (рис. 1.13). Mars Made Easy — це найпростіший в Інтернеті спосіб зшивати, каталогізувати та анотувати карти NIPR.

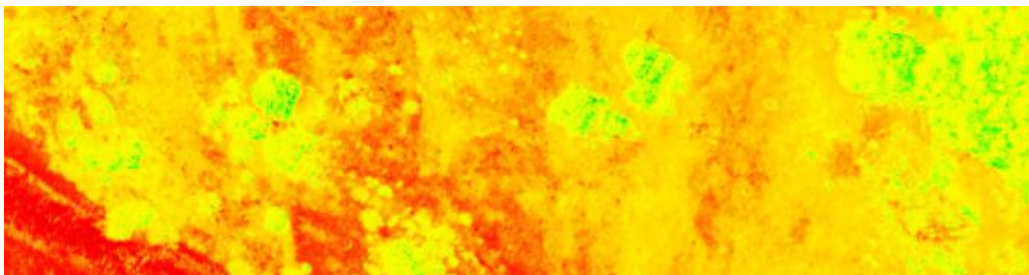


Рисунок 1.13 – Вигляд NIPR з зображення дрона [11]

Обробка карти складна. Це вимагає багато обчислювальних сил і процесорного часу. Якби спробувати виконати таку саму обробку, яка робиться на нашому спеціально створеному сервері, швидше за все, користувач не мав би системних ресурсів мобільного робота або не міг використовувати його годинами поспіль. Тому програма потребує підключення до інтернету – повна онлайн обробка.

Застосунок має багаторазовий менеджмент, тому кілька візитів із геоприв’язкою до одного сайту картографування можна комбінувати та

організувати за допомогою інтерфейсу перетягування, можливо створити та швидко переглянути історію для кожної точки на карті (рис. 1.14). Ця функція особливо корисна для моніторингу будівництва, сільського господарства, документації трубопроводів та перевірки сонячних батарей.



Рисунок 1.14 – Вигляд карти з проміжком часу [11]

1.5 Постановка задач

Таким чином, для досягнення поставленої мети роботи, необхідно виконати наступні завдання:

- розробити алгоритми, які забезпечать точне визначення місцезнаходження мобільного робота на основі отриманих даних з сенсорів або зовнішніх пристроїв;
- підключитися до картографічних сервісів (наприклад, Google Maps або OpenStreetMap) для отримання актуальних карт та можливості відображення на них місцезнаходження мобільного робота;
- створити інтерфейс, що дозволить користувачу взаємодіяти з модулем, налаштовувати його параметри та переглядати результати відстеження на карті;
- провести випробування модуля для перевірки його працездатності й відповідності вимогам та підтвердити точність та надійність результатів відстеження.

2 ПРОЄКТУВАННЯ АРХІТЕКТУРИ ВЗАЄМОДІЇ СИСТЕМИ З МОБІЛЬНИМ РОБОТОМ ТА АЛГОРИТМІВ ЙОГО РОБОТИ

2.1 Розробка архітектури взаємодії системи з мобільним роботом

Архітектура взаємодії автоматизованого модуля з мобільним роботом визначає те що як всі компоненти системи будуть пов'язані між собою. В неї входять: мобільний робот, сервіс та програмний модуль.

Мобільний робот складатиметься з таких компонентів:

- сенсор, який вказуватиме місцезнаходження;
- мікроконтролер, який оброблятиме дані;
- зв'язковий модуль з сервісом.

Отримані дані з мобільного робота передаватимуться на сервіс, а він вже буде:

- приймати дані;
- зберігати дані;
- оброблювати дані.

Програмний модуль буде вже робити такі дії:

- отримувати дані про місцезнаходження мобільного робота з сервісу;
- відображати мобільний робот на карті.

Ця узагальнена архітектура описує, як різні компоненти системи взаємодіють один з одним, забезпечуючи ефективне збирання, обробку та відображення даних про місцезнаходження мобільного робота (рис. 2.1).

Завдяки цій архітектурі, система може забезпечити високу точність та надійність у моніторингу та управлінні мобільними роботами, що є критично важливим для багатьох застосувань у промисловості, логістиці, сільському господарстві та інших галузях.

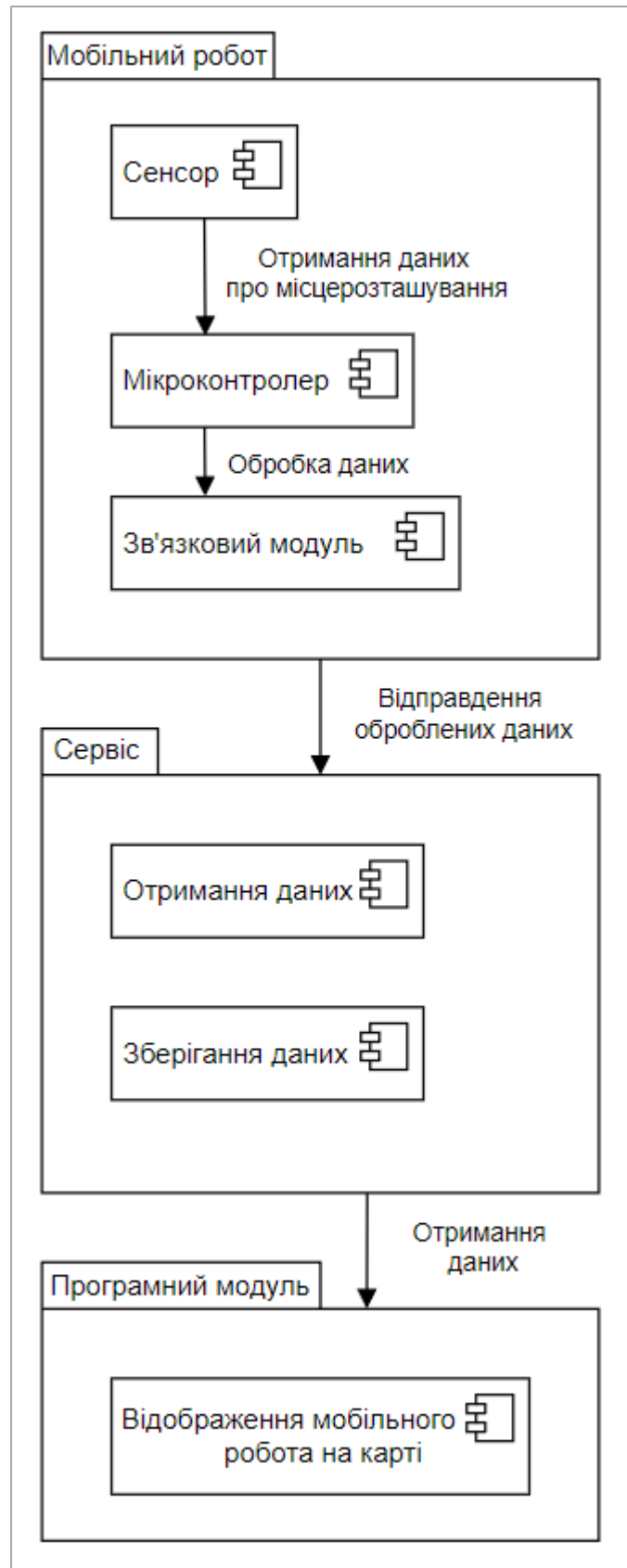


Рисунок 2.1 – Архітектура взаємодії системи з мобільним роботом

2.2 Розробка структурної схеми модуля

Структурна схема модуля деталізує фізичні та логічні зв'язки між компонентами системи. Основні компоненти які будуть включені:

- сенсор для отримання координат мобільного робота;
- мікроконтролер для обробки даних з сенсора;
- зв'язковий модуль для передачі даних на сервіс;
- сервіс на якому зберігатимуться дані;
- інтерфейс користувача для відображення місцезнаходження мобільного робота.

Структурна схема деталізує фізичні та логічні зв'язки між компонентами системи, забезпечуючи ефективну взаємодію між сенсорами, мікроконтролером, зв'язковим модулем, сервісом та інтерфейсом користувача для точного відображення місцезнаходження мобільного робота (рис. 2.2).

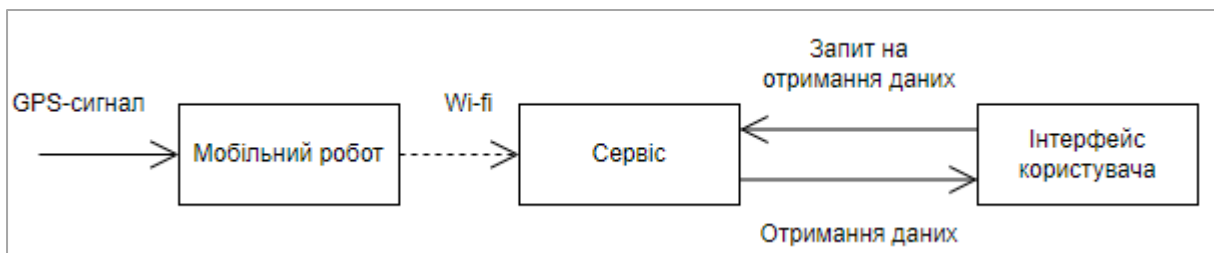


Рисунок 2.2 – Структурна схема модуля

2.3 Опис алгоритмів роботи модуля

Алгоритм роботи модуля відстеження та відображення місцезнаходження мобільного робота починається зі збору даних сенсору мобільного робота, який передає інформацію про поточне місцезнаходження та оточення робота до мікроконтролера.

Мікроконтролер обробляє ці дані для визначення ключових параметрів, таких як координати, орієнтація та швидкість робота. Далі, оброблені дані передаються через зв'язковий модуль до сервісу.

Сервіс приймає отримані дані від мобільного робота через зв'язковий модуль. Після отримання даних він переходить до їх подальшої обробки.

Цей процес включає в себе ряд операцій, таких як перевірка на коректність та цілісність даних, виконання різних видів аналізу.

Після обробки дані зберігаються у відповідному сховищі даних. Забезпечуючи надійне збереження отриманих даних, сервіс гарантує доступність цієї інформації для подальшого використання, аналізу або візуалізації.

Це дозволяє зберігати історію даних та забезпечує можливість використання їх у майбутніх дослідженнях, прийнятті рішень або для інших цілей, необхідних для діяльності системи.

Потім оброблені дані стають доступними для інтерфейсу користувача, який через відповідний запит отримує необхідну інформацію з сервісу.

Інтерфейс користувача виконує візуалізацію місцезнаходження мобільного робота на інтерактивній карті, відображаючи його поточний стан та траєкторію руху.

Крім того, інтерфейс користувача дозволяє побачити відеопотік з мобільного робота у реальному часі.

Таким чином, алгоритм забезпечує безперервний цикл збору, обробки та візуалізації даних, що дозволяє точно відстежувати місцезнаходження мобільного робота та ефективно керувати ним.

Цей алгоритм забезпечує безперервний цикл збору, обробки та візуалізації даних, що дозволяє точно відстежувати місцезнаходження мобільного робота та ефективно керувати ним (рис. 2.3).

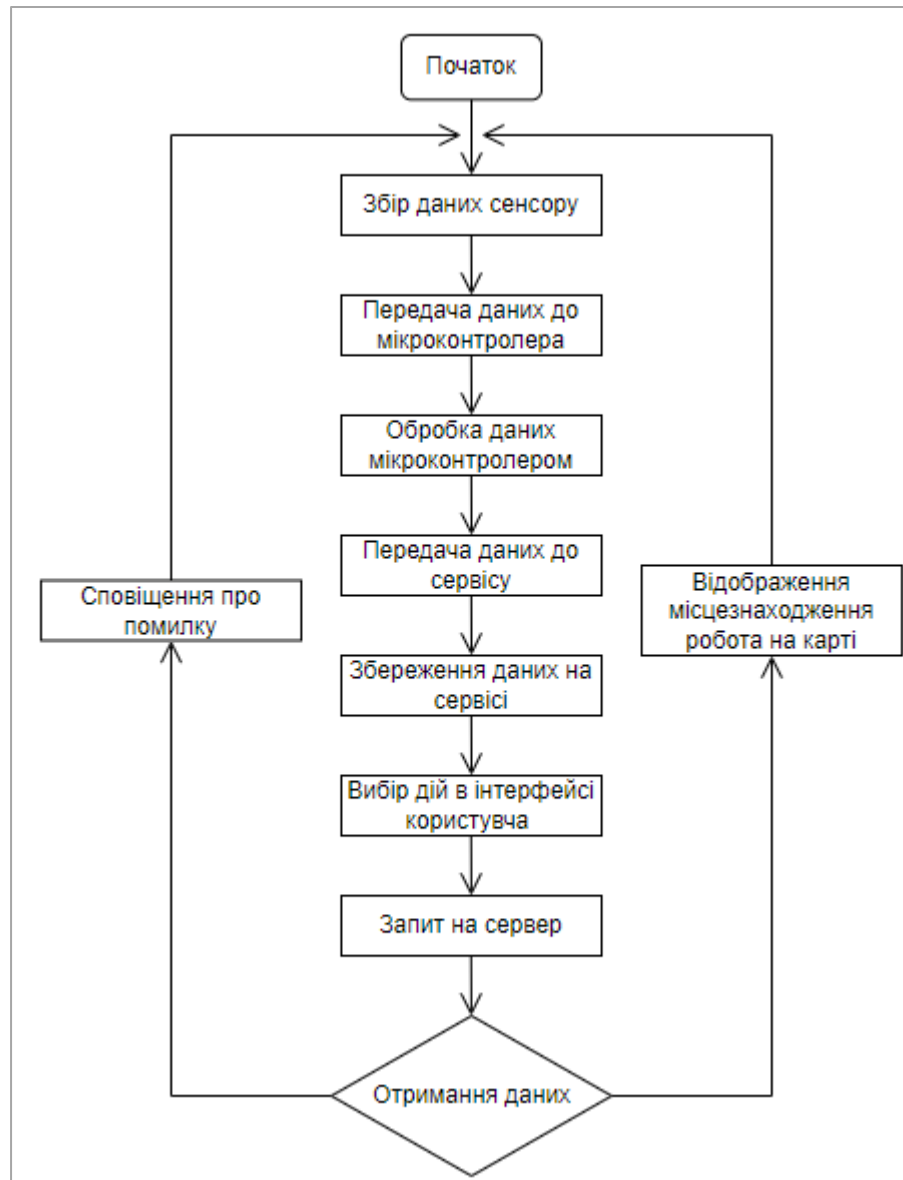


Рисунок 2.3 – Алгоритм роботи системи

2.4 Опис протоколу взаємодії між модулем та мобільним роботом

Протокол взаємодії між модулем і мобільним роботом заснований на HTTPS, REST API та методі GET для обміну даними. Протокол забезпечує безпечну передачу даних між компонентами системи, зберігаючи конфіденційність і цілісність інформації.

HTTPS – це розширення протоколу HTTP, яке використовується для безпечного обміну інформацією через Інтернет. Він забезпечує шифрування

даних, що передаються між клієнтом і сервером, за допомогою протоколів SSL (Secure Sockets Layer) або TLS (Transport Layer Security).

HTTPS використовує шифрування для захисту даних від перехоплення та несанкціонованого доступу під час передачі. Це гарантує, що конфіденційна інформація, така як паролі або платіжні дані, залишається приватною. Протокол має цілісність даних, яке є гарантією що дані не були змінені або пошкоджені під час передачі.

Сертифікати SSL/TLS підтверджують справжність веб-сайту, що дозволяє клієнтам бути впевненими, що вони спілкуються саме з тим сервером, з яким вони мають намір спілкуватися.

REST API – це архітектурний стиль для розробки веб-сервісів, який базується на протоколі HTTP. У REST API кожен ресурс ідентифікується унікальним URI (Uniform Resource Identifier).

Статусні коди HTTP використовуються для інформування клієнта про результати запиту (200 OK, 404 Not Found, 500 Internal Server Error). Кожен запит до сервера повинен містити всю необхідну інформацію для його обробки. Сервер не зберігає стан клієнта між запитами. REST API зазвичай використовує JSON або XML для передачі даних.

GET – це метод HTTP, який використовується для отримання даних з сервера (рис. 2.4). Він є одним з основних методів у REST API і має такі характеристики:

- ідемпотентність (виконання одного і того ж GET запиту кілька разів матиме однаковий результат і не змінює стан сервера);
- безпека (GET запити зазвичай використовуються для отримання даних і не повинні змінювати ресурс на сервері);
- параметри запиту (GET запити можуть містити параметри у вигляді рядка запиту для фільтрації або уточнення даних, які потрібно отримати);

– кешування (веб-браузери та інші клієнти можуть кешувати відповіді на GET запити, щоб зменшити навантаження на сервер і прискорити доступ до даних).

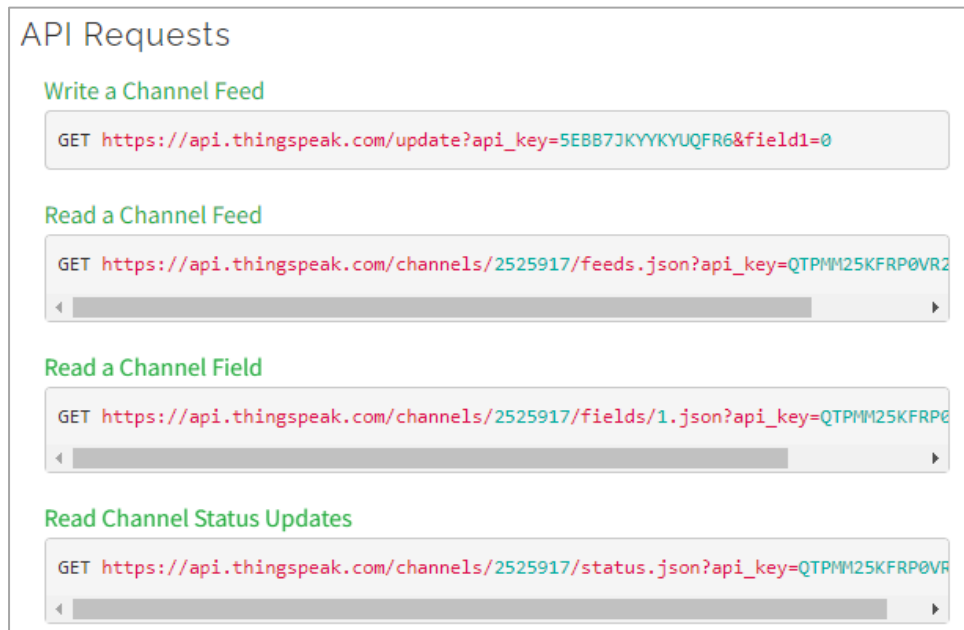


Рисунок 2.4 – Приклад GET запитів на запис та читання даних

2.5 Висновки до другого розділу

У цьому розділі було розроблено основні компоненти архітектури автоматизованого модуля відстеження та відображення місцезнаходження мобільного робота. Було детально описано структурну схему модуля, алгоритми його роботи та протоколи взаємодії з мобільним роботом. У підсумку, ці технічні рішення створюють основу для побудови ефективної системи відстеження, яка може бути легко інтегрована в різні автоматизовані платформи, забезпечуючи високу точність та надійність роботи мобільного робота.

3 РОЗРОБКА АПАРАТНОГО ЗАБЕЗПЕЧЕННЯ МОДУЛЯ ВІДСТЕЖЕННЯ МОБІЛЬНОГО РОБОТА

3.1 Вибір електронних компонентів модуля

Для отримання даних про місцезнаходження мобільного робота на карті місцевості потрібен датчик відслідковування – GPS-модуль.

На даний момент одним з найкращих виборів для цієї задачі є модуль NEO-6M (рис. 3.1).



Рисунок 3.1 – GPS-модуль NEO-6M [12]

Модуль NEO-6M має високу чутливість (-161 дБм), що дозволяє йому працювати навіть при слабкому сигналі. Точність позиціонування складає приблизно 2,5 метра, що достатньо для більшості застосувань, де потрібні точні координати.

Час холодного старту (перше ввімкнення) становить приблизно 27 секунд, а гарячого старту (повторне ввімкнення) – всього 1 секунду. Це дозволяє швидко починати роботу після ввімкнення пристрою.

Модуль може працювати від 3,3 В або 5 В, що робить його сумісним з різними мікроконтролерами та іншими пристроями. Потреба в енергії становить близько 45 мА в активному режимі, що підходить для використання в пристроях з обмеженим джерелом живлення.

Модуль підтримує стандартні протоколи передачі даних (NMEA та UBX), а інтерфейс UART дозволяє легко підключити його до більшості мікроконтролерів.

Невеликі розміри модуля (приблизно 25 мм × 35 мм) дозволяють використовувати його в пристроях з обмеженим простором. Є можливість підключення активної або пасивної антени дозволяє покращити якість прийому сигналу та розширити можливості застосування модуля [13].

Обравши GPS-модуль NEO-6М, можливо забезпечити мобільному роботу надійне і точне визначення місцезнаходження, що є критично важливим для ефективної навігації та виконання завдань.

Для відображення інформації щодо місцезнаходження мобільного робота на карті місцевості потрібен дисплей.

Одним із найзручніших варіантів є дисплей LCD 1602 (рис. 3.2) з модулем I2C (рис. 3.3), який забезпечує простоту підключення та використання, а також дозволяє відображати текстову інформацію на двох рядках по 16 символів у кожному.

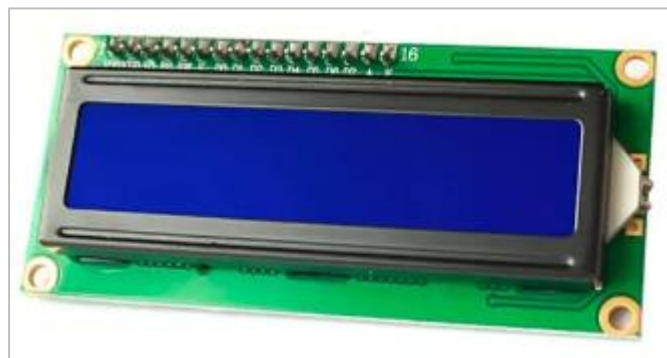


Рисунок 3.2 – LCD дисплей 1602 [14]

Використання I2C значно скорочує кількість проводів, необхідних для підключення дисплея до мікроконтролера, з 6-10 до всього 4 (VCC, GND, SDA, SCL).

Подібна конфігурація дозволяє ефективно використовувати обмежений ресурс з'єднань мікроконтролера, забезпечуючи при цьому стабільну роботу та зручність у використанні. Більш того, інтерфейс I2C є стандартним для багатьох електронних пристроїв, що дозволяє легко розширювати функціональність системи шляхом додавання нових пристроїв та модулів без значного збільшення кількості необхідних з'єднань.

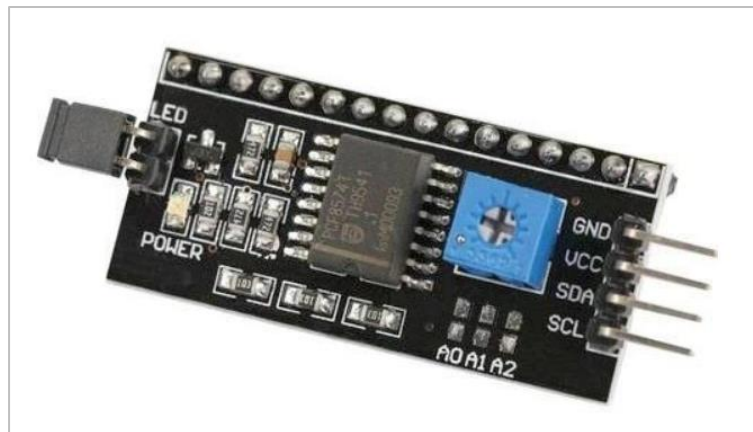


Рисунок 3.3 – Шина I2C [15]

Для обробки інформації про місцезнаходження мобільного робота на карті місцевості вибрано модуль – ESP-WROOM-32 (рис. 3.4).

Модуль ESP-WROOM-32 є популярним вибором для багатьох IoT (Internet of Things) проектів завдяки своїй потужності, багатофункціональності та підтримці бездротових протоколів.

Переваги використання ESP-WROOM-32:

- двоядерний процесор та достатня кількість оперативної пам'яті дозволяють виконувати складні завдання та обробляти дані в реальному часі;
- підтримує Wi-Fi та Bluetooth, що робить модуль ідеальним для бездротових комунікацій та підключення до мережі Інтернет;

- доволі велика кількість пінів введення/виведення;
- режим зниження енергоспоживання дозволяють значно продовжити час роботи від батареї, що важливо для автономних пристроїв.

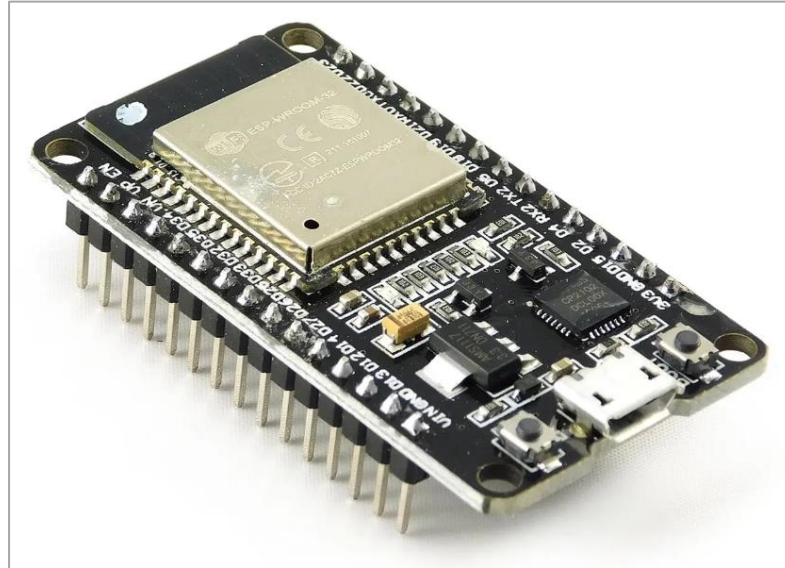


Рисунок 3.4 – Модуль ESP-WROOM-32 [16]

Для відслідковування візуального відображення мобільного робота на карті місцевості у реальному часі було обрано модуль ESP32-CAM.

ESP32-CAM – це компактний модуль, який поєднує в собі мікроконтролер ESP32 та камеру (рис. 3.5). Цей модуль став дуже популярним завдяки своїй здатності до зйомки та передачі зображень та відео через мережу Wi-Fi.

Модуль може здійснювати зйомку фотографій у високій роздільній здатності, яка зазвичай становить 2 мегапікселі. Також може записувати відео в різних форматах та роздільній здатності. Це відкриває широкі можливості для створення проектів, пов'язаних з відео-наглядом та моніторингом за мобільним роботом.

ESP32-CAM має потужний процесор, який дозволяє проводити обробку зображень навіть на самому модулі. Це дозволяє створювати проекти, пов'язані з розпізнаванням облич, предметів, руху, виконанням фільтрів та іншими видами обробки зображень.

Крім камери, ESP32-CAM має вбудовані інтерфейси введення/виведення, такі як GPIO, UART та SPI, що розширює його можливості для підключення додаткових пристроїв та сенсорів.



Рисунок 3.5 – Модуль ESP32-CAM [17]

3.2 Віртуальне підключення компонентів

Перед тим як приступати до збирання реального макету, потрібно перевірити працездатність у віртуальному середовищі. Для цього було обрано середовище Wokwi.

Wokwi – це платформа для симуляції електронних схем і проектів на основі мікроконтролерів. Вона дозволяє користувачам створювати, налагоджувати та тестувати електронні проекти у віртуальному середовищі без необхідності фізичних компонентів.

Wokwi підтримує популярні мікроконтролери, такі як Arduino (UNO, Nano), ESP32, і Raspberry Pi Pico. Інтерфейс Wokwi включає візуальний редактор, де користувачі можуть розміщувати компоненти на віртуальній платі, з'єднувати їх дротами та бачити, як виглядатиме їхній проект у реальному житті. Платформа має велику бібліотеку компонентів, таких як світлодіоди, резистори,

конденсатори, сенсори, дисплеї та інші. Також є можливість створення кастомних електронних елементів, що набагато розширює функціонал.

Wokwi дозволяє писати і завантажувати код для мікроконтролерів безпосередньо в браузері. Можна тестувати і налагоджувати код у реальному часі, використовуючи вбудовані інструменти для дебагінгу.

Для початку створюємо проєкт на базі модуля ESP-WROOM-32, який у даній програмі має назву – ESP32 DevKit v1 (рис. 3.6).

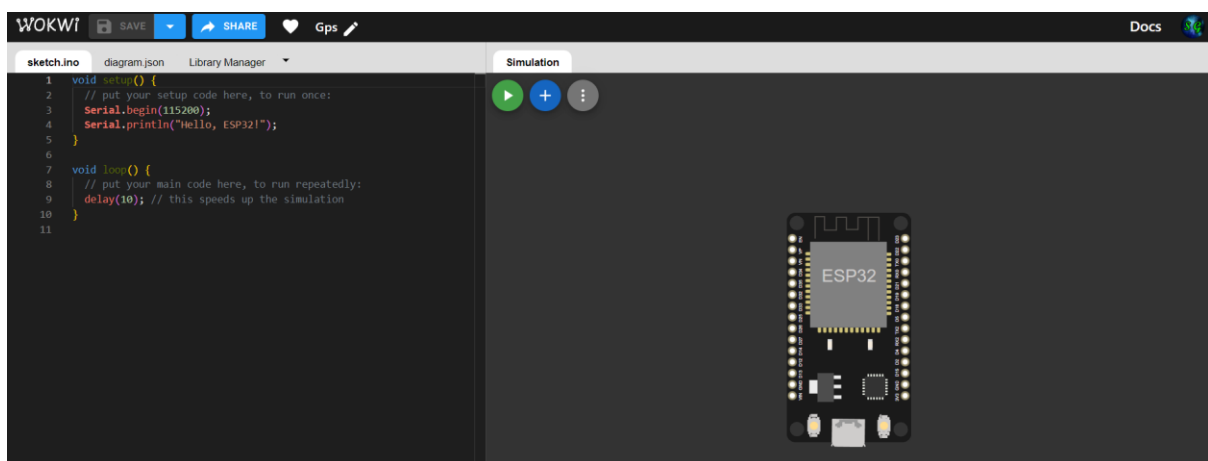


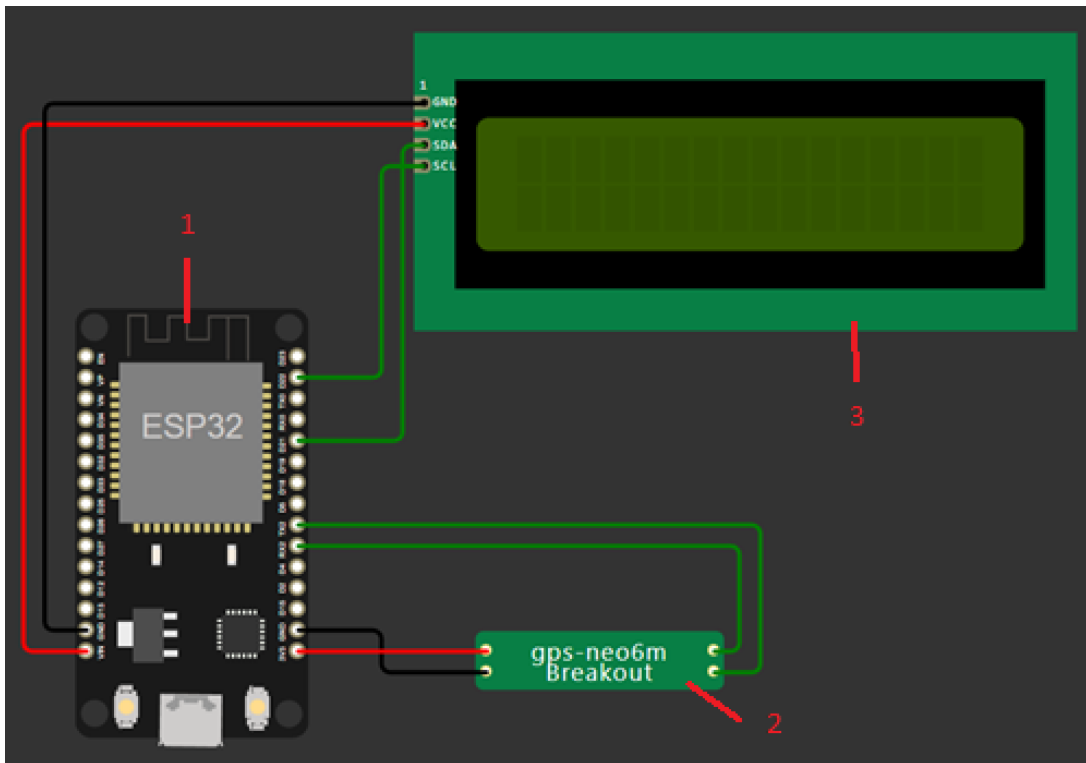
Рисунок 3.6 – Початкове вікно створеного проєкту на базі модуля ESP-WROOM-32

Далі потрібно створити головний елемент для відслідковування мобільного робота – GPS-модуль. Для цього обрано створення кастомного чіпу, який буде запрограмований на мові C. Даний модуль створюється з стандартними пінами, тому потрібно замінити вхідний контакт на RX, а вихідний на TX.

Для відображення координат буде використано LCD дисплей 1602 з шиною I2C, тому додаємо цей елемент до макету.

Тепер потрібно під'єднати всі елементи до модуля ESP32. GPS-модуль підключаємо так, щоб пін VCC був до пину 3.3V, GND до GNG, TX до RX2, RX до TX2, а LCD дисплей підключаємо так: GND до GNG, VCC до VIN, SDA до

D21 та SCL до D22 (рис. 3.7). Виконуючи зазначені підключення, необхідно дотримуватися правильної полярності та обережності, щоб уникнути можливих пошкоджень електроніки. Також слід звернути увагу на правильність підключення ліній TX та RX між GPS-модулем та ESP32, оскільки це важливо для передачі даних між ними.



1 – модуль ESP-WROOM-32; 2 – GPS-модуль;
3 – LCD дисплей 1602 з шиною I2C

Рисунок 3.7 – Зібраний віртуальний макет

Після програмування модулів ESP-WROOM-32 та GPS-модуля, запускаємо симуляцію та отримуємо передачу даних про місцезнаходження мобільного робота на карті місцевості на вивід на дисплей координат (рис. 3.8).

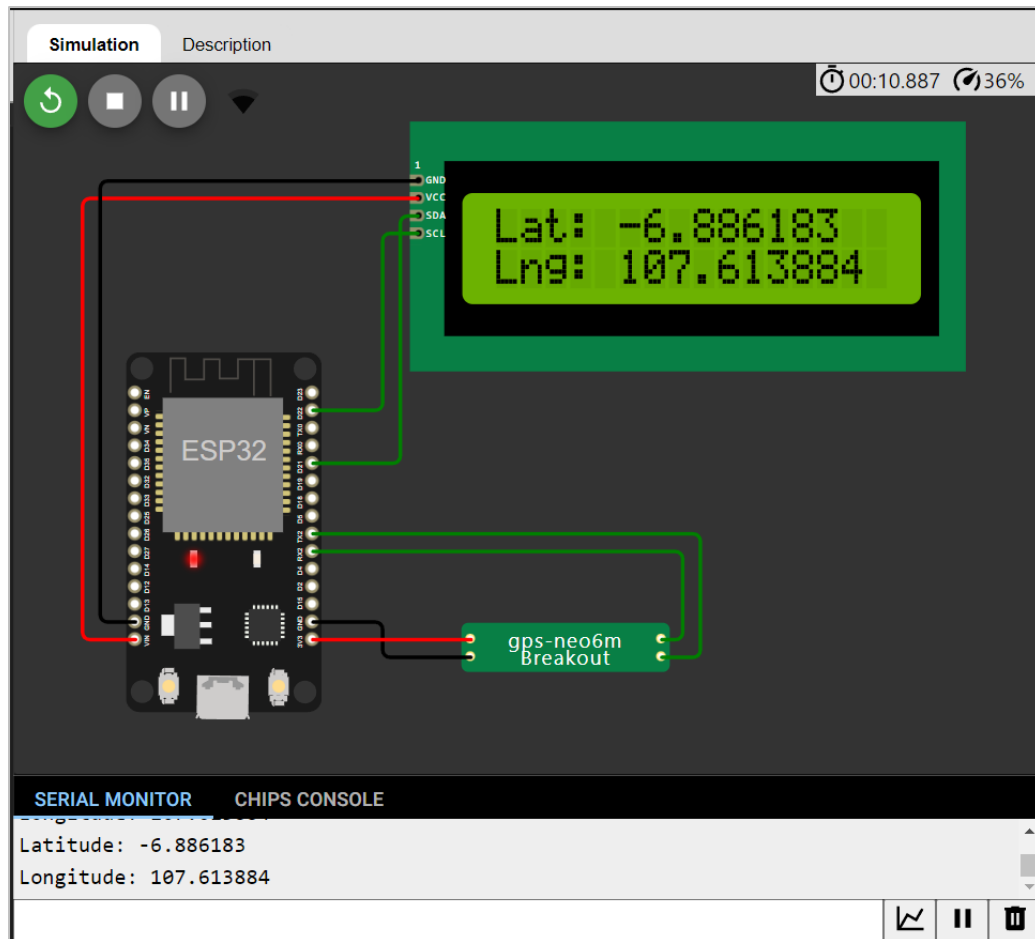


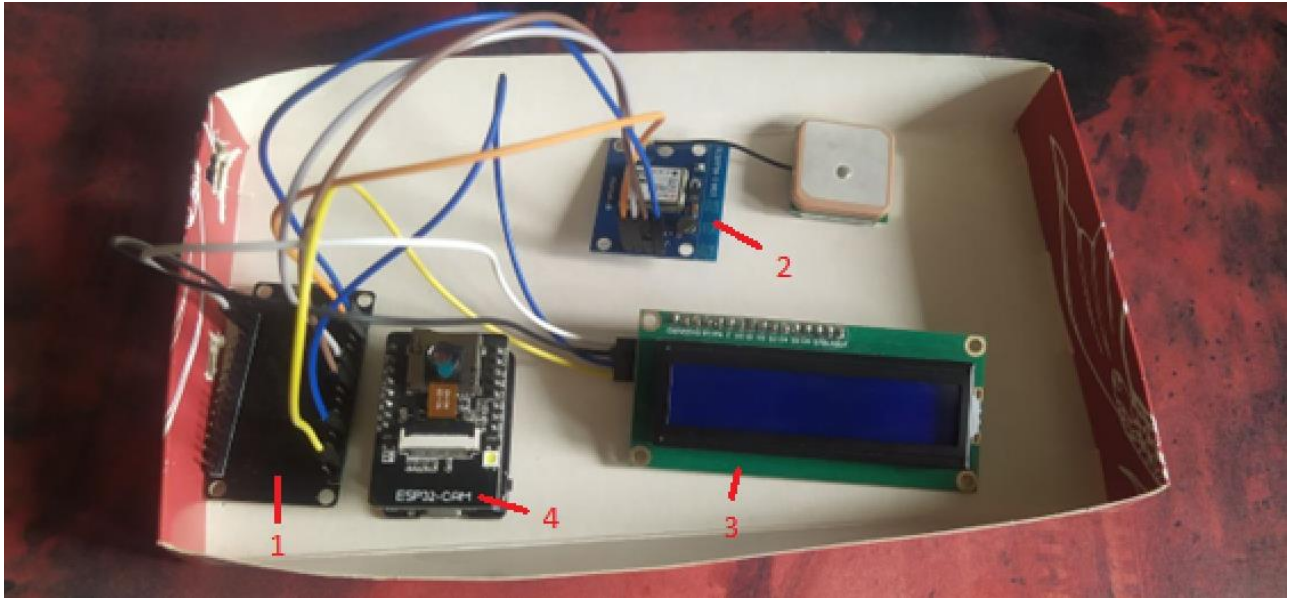
Рисунок 3.8 – Робота віртуального макету

3.3 Практичне підключення компонентів

Після того як було проведено симуляцію у віртуальному середовищі та отримані результати коректного функціонування модулів, то тепер можна переходити до покупки реальних пристроїв та їх з'єднання між собою.

Підключення всіх компонентів виконується так само як й на віртуальному макеті, крім двох контактів на GPS-модулі, а саме: TX до D16, RX до D17 (рис. 3.9).

Цей процес дозволяє перевірити, чи працюють всі з'єднання на реальних пристроях так само ефективно, як і на віртуальному макеті, та виявити можливі проблеми з підключенням чи зв'язком між компонентами.



1 – модуль ESP-WROOM-32; 2 – GPS-модуль;
3 – LCD дисплей 1602 з шиною I2C; 4 – модуль ESP32-CAM

Рисунок 3.9 – Зібраний фізичний макет

Для подачі напруги буде використано павербанк фірми Redmi на 20000 мА, підключення до модулів відбувається за допомогою кабелів USB TYPE C та USB TYPE B. Але це може бути не тільки павербанк, а й ноутбук, смартфон та інші компоненти які підтримують подачу напруги.

У середовищі проектування Fusion 360 для фізичного макету було виконано побудову 3D-модель коробка для того щоб компактно розмістити електронні компоненти, він має розміри: 61,2 мм × 154 мм × 41 мм (рис. 3.10).

Ця 3D-модель дозволяє ефективно спроектувати корпус таким чином, щоб забезпечити оптимальну організацію простору та зручний доступ до всіх компонентів при потребі обслуговування чи ремонту. Крім того, вона допомагає попередньо оцінити зовнішній вигляд та функціональність макету перед його фізичним створенням.

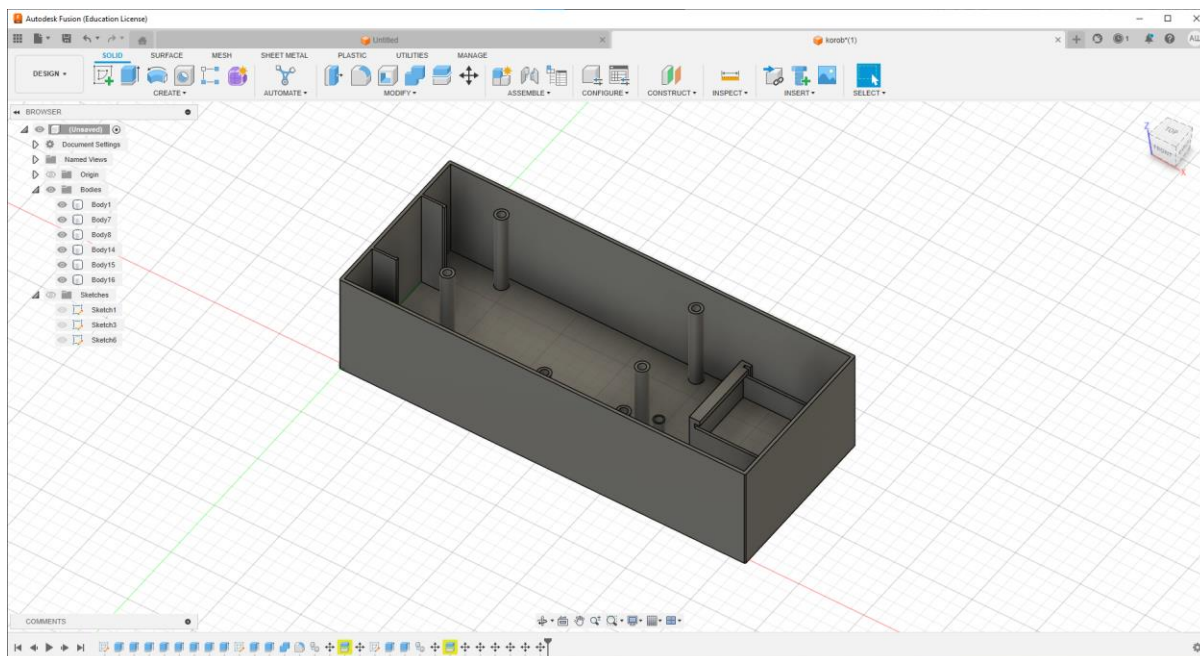


Рисунок 3.10 – 3D-модель коробка

3D-модель коробка має розміри 61,2 мм × 154 мм × 1,2 мм, а виступ для кріплення кришки висотою 4 мм (рис. 3.11).

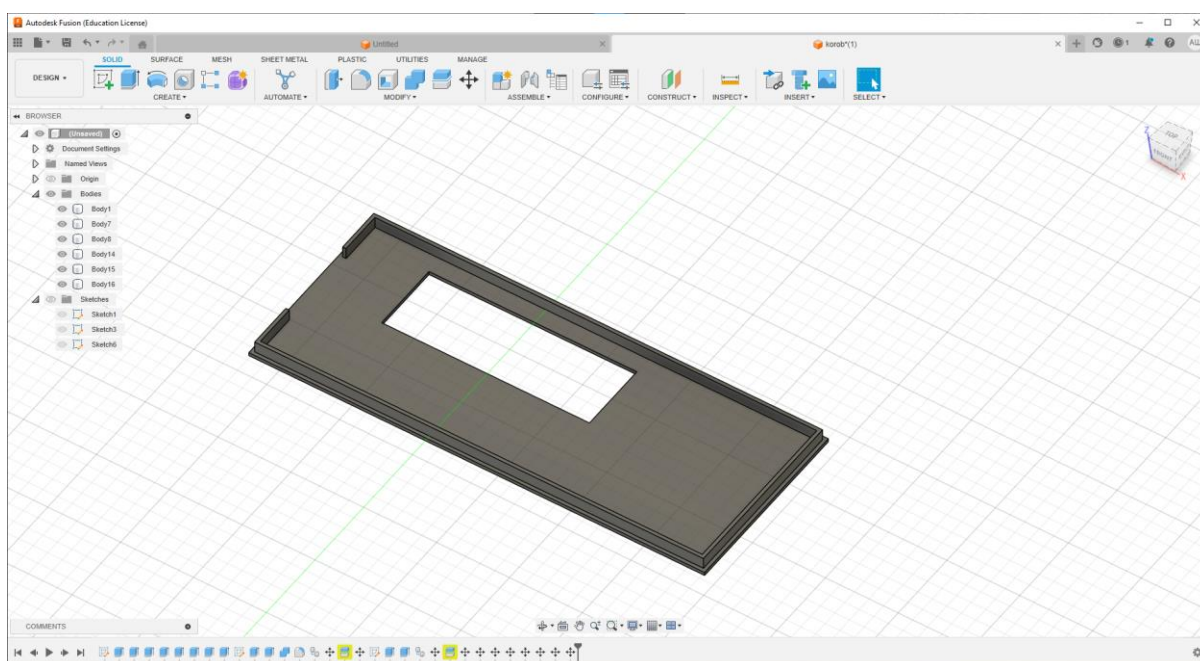


Рисунок 3.11 – 3D-модель кришки коробка

Для друку використовується 3D-принтер Easy Thread X1. Технологію друку яку він використовує це FDM (Fused Deposition Modeling) – метод шаруватого нанесення розплавленого пластика. Область друку становить 100 мм × 100 мм × 100 мм (рис. 3.12).

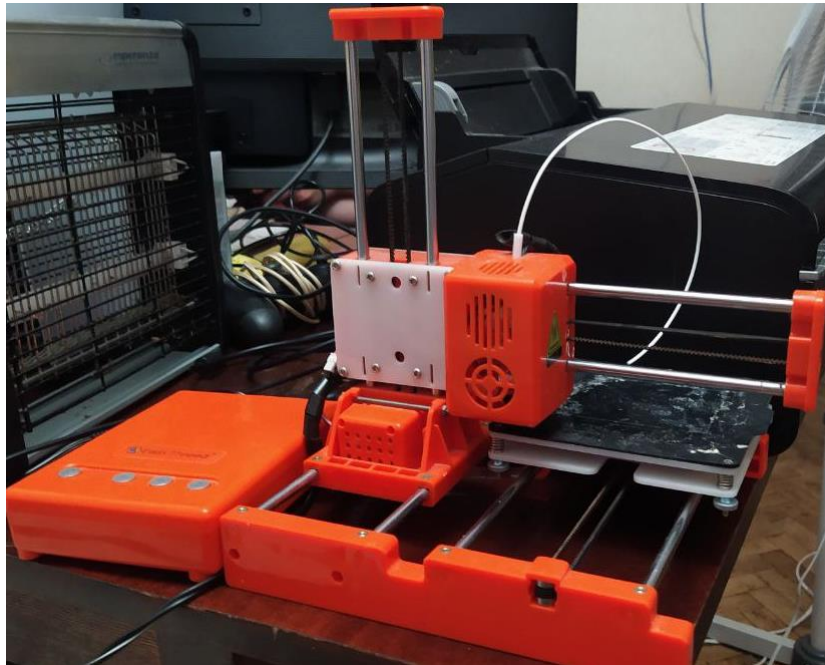


Рисунок 3.12 – 3D-принтер Easy Thread X1

Цей 3D принтер Підтримує два види пластику: PLA та ABS , що забезпечує гнучкість у виборі матеріалів залежно від вимог проекту та бажаних властивостей кінцевих виробів. Використання цих матеріалів дозволяє отримувати міцні та стабільні компоненти для корпусу та інших елементів проекту.

Оскільки моделька більша за розміри стола, було прийнято рішення розділити її на дві частини (рис. 3.13).

Після друку двох частин потрібно виконати скріплення та виконати пост-обробку цих моделей.

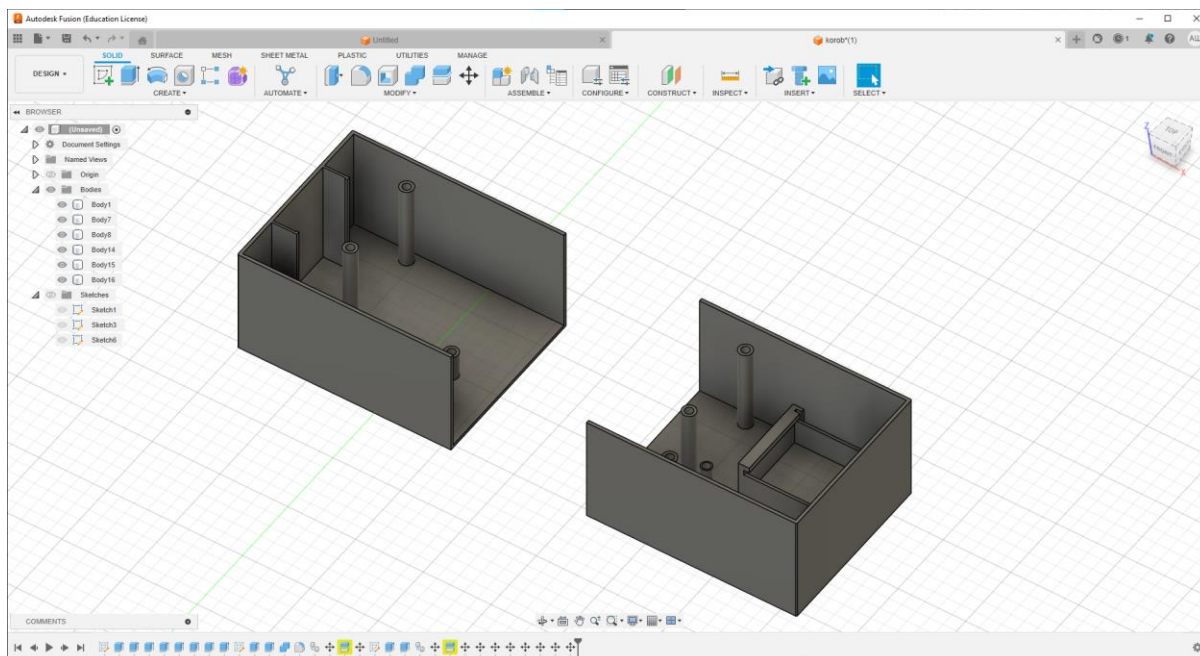


Рисунок 3.13 – 3D-модель розділеного коробка

Кришку коробка також потрібно розділити на дві частини, задля нормального друку 3D-принтера (рис. 3.14).

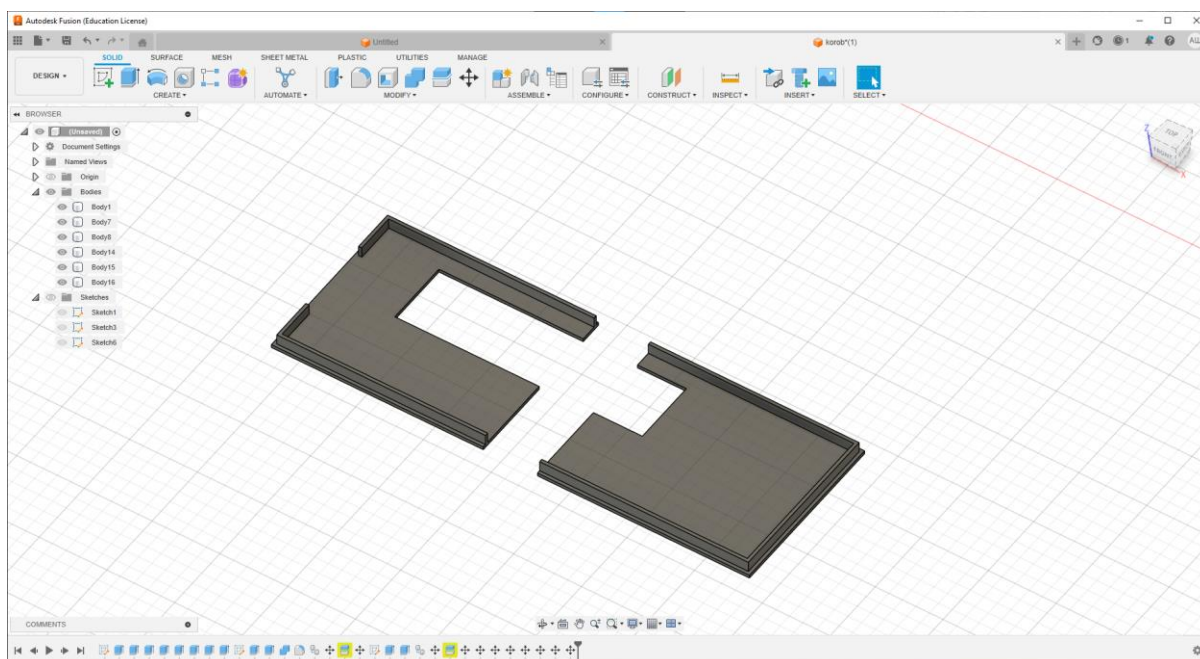
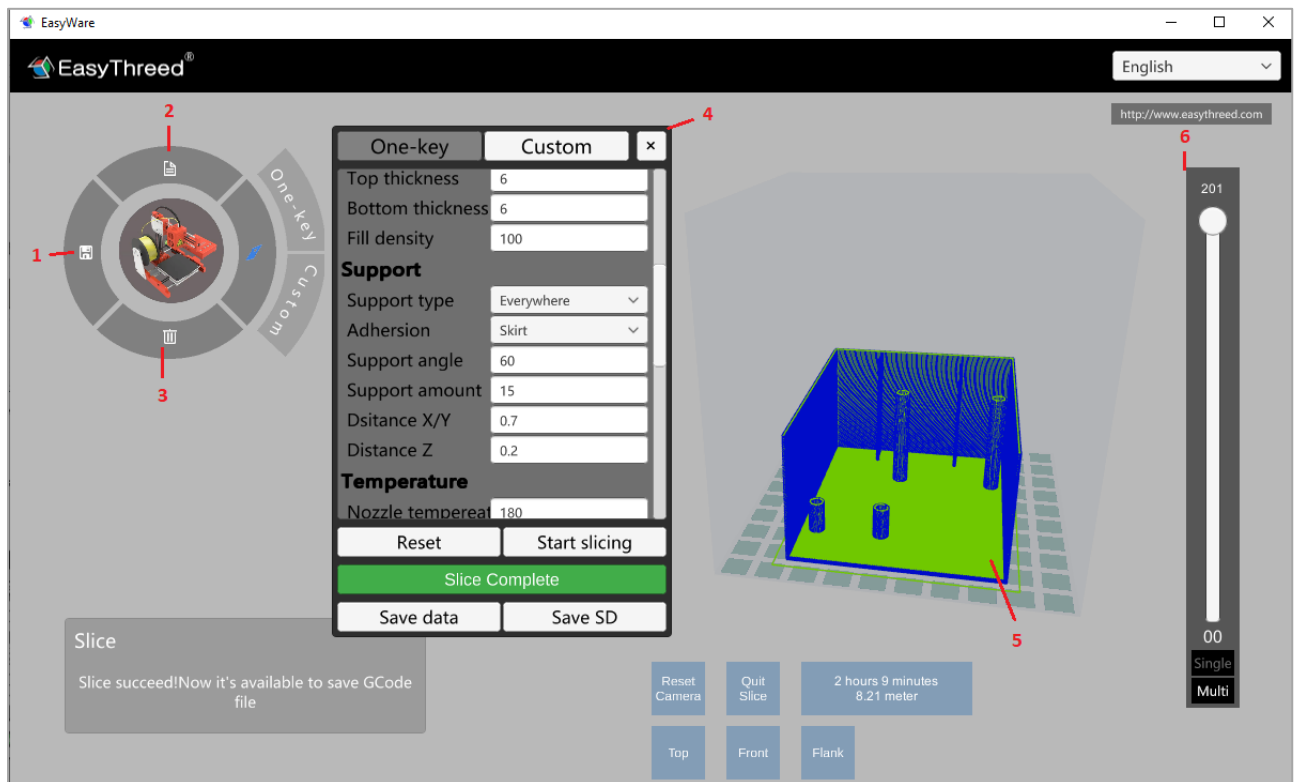


Рисунок 3.14 – 3D-модель розділеної кришки коробка

Для принтера марки Easy Thread є слайсер EasyWare, який використовується для нарізання моделей на шари.

EasyWare реалізований на основі слайсеру Cura, що забезпечує йому широкий функціонал та можливості для оптимального налаштування друку.

Використання EasyWare спрощує процес підготовки моделей до друку та дозволяє отримувати якісні результати за короткий час (рис. 3.15).



- 1 – зберегти модель; 2 – завантажити модель; 3 – видалити модель;
 4 – налаштування слайсування моделі; 5 – візуальне зображення моделі;
 6 – вигляд по шарового слайсування моделі

Рисунок 3.15 – Вигляд програми EasyWare

Для слайсування моделей коробка й кришки використанні стандартні налаштування середовища Easy Thread, але були змінені параметри щільності

заповнення моделі на 100 відсотків та адгезія на спідниця, задля отримання більш міцного корпусу й зменшення кількості використання пластику (рис. 3.16).

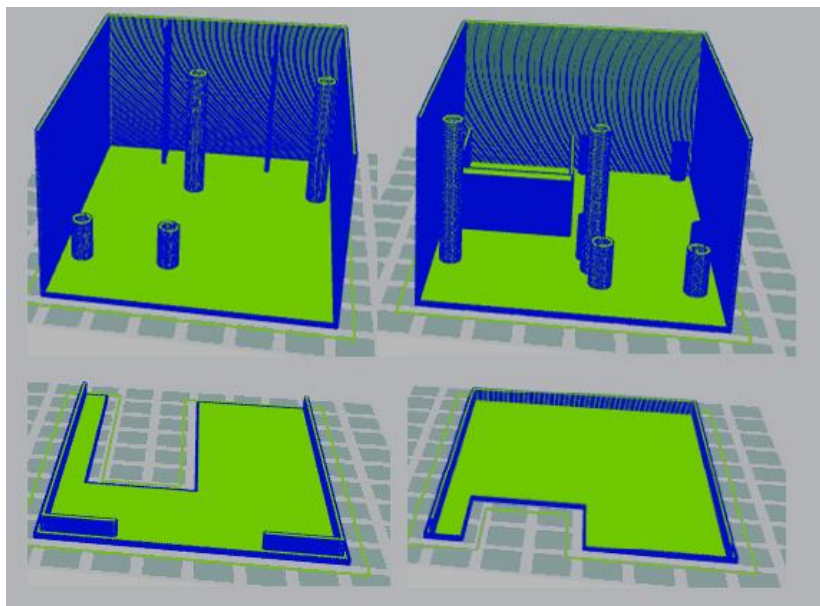


Рисунок 3.16 – Наслайсовані моделі

Після слайсування в EasyWare, було завантажено отриманий файл на карту пам'яті, вставлено її в 3D принтер та розпочато друк. При завершенні друку було отримано відповідні елементи коробочки (рис. 3.17).

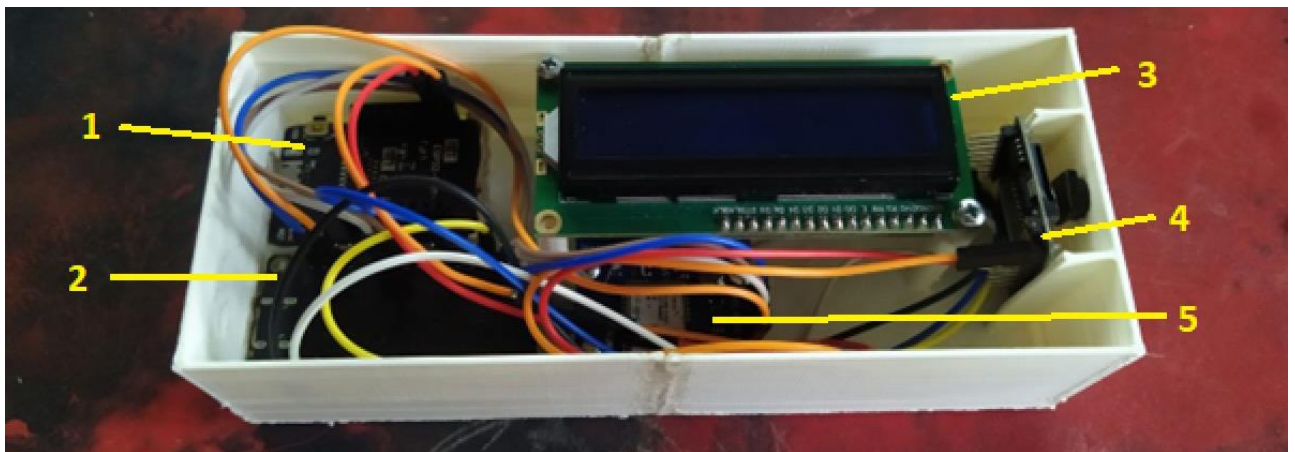


Рисунок 3.17 – Надруковані моделі

Пост-обробку надрукованих елементів було виконано за допомогою скальпеля, наждачного паперу та напилка по металу. Цей процес включав видалення зайвого матеріалу, згладжування поверхонь та корекцію дрібних дефектів, що виникли під час друку. Використання скальпеля дозволило акуратно обрізати залишки пластику та задирки, тоді як наждачний папір і напилки допомогли досягти гладкості поверхонь та точності деталей.

Для скріплення моделей під високою температурою використовувалася технологія термічного зварювання пластику. Цей метод забезпечує міцне з'єднання елементів, яке витримує механічні навантаження та забезпечує довговічність конструкції.

Після завершення пост-обробки, електронні компоненти було розташовано на відповідні місця всередині коробки. Для їх надійного закріплення використовувалися шурупи, що забезпечують стабільне розташування та запобігають руху компонентів під час експлуатації (рис. 3.18).



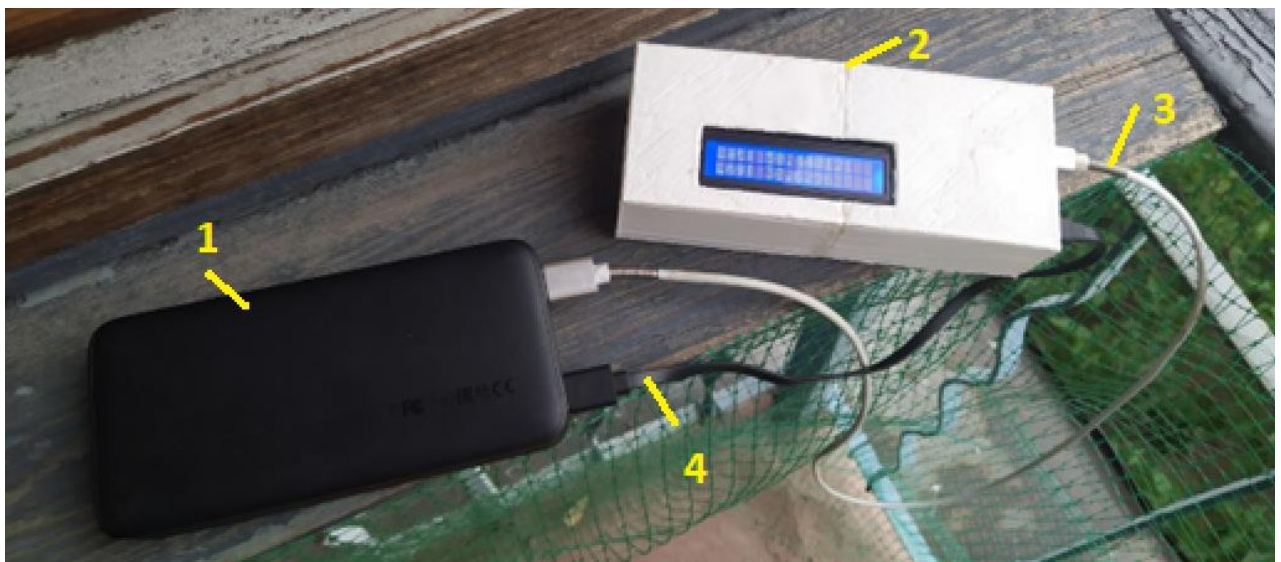
- 1 – модуль живлення ESP32-CAM; 2 – модуль ESP-WROOM-32;
 3 – LCD дисплей 1602 з шиною I2C; 4 – модуль ESP32-CAM;
 5 – GPS-модуль

Рисунок 3.18 – Розташування електронних компонентів

Закінчивши монтаж електронних компонентів, коробку було накрито кришкою, яка також була ретельно надрукована та оброблена.

Після цього система була підключена до живлення для проведення фінальних тестів (рис. 3.19).

Під час тестування перевірялася коректність роботи всіх компонентів, зокрема GPS-модуля та дисплея, їх взаємодія та відповідність програмному забезпеченню.



1 – пристрій живлення; 2 – апаратний модуль;
3 – кабель USB TYPE C; 4 – кабель USB TYPE B

Рисунок 3.19 – Створений фізичний макет

Для розрахунку часу роботи апаратного модуля потрібно визначити споживану потужність кожного пристрою та ємність павербанку.

Застосовані модулі споживають таку потужність: ESP WROOM 32 приблизно 160 мА при активній роботі, GPS-модуль NEO6M близько 30 мА, дисплей LCD 1602 – 20 мА, ESP32-CAM також 160 мА при активній роботі.

Теорія автоматичного управління включає в себе різні методи та критерії для оцінки та покращення систем управління.

Точність системи управління визначається відхиленням від бажаного стану або траєкторії. Середнє квадратичне відхилення (СКВ) є одним з основних критеріїв точності, який використовується для оцінки якості системи управління. Менше СКВ означає більш точне слідування заданій траєкторії, що є важливим аспектом в автоматичному управлінні.

Розрахунок середнього квадратичного відхилення (СКВ) виконується за формулою (3.1):

$$\text{СКВ} = \sqrt{\frac{1}{N} \sum_{i=1}^N [(x_i - x_{i\text{зад}})^2 + (y_i - y_{i\text{зад}})^2]}, \quad (3.1)$$

де N – кількість вимірів (точок маршруту);

x_i – фактична координата x точки i (фактична);

$x_{i\text{зад}}$ – задана координата x точки i ;

y_i – фактична координата y точки i (фактична);

$y_{i\text{зад}}$ – задана координата y точки i .

Розрахунок для першої точки (50,4501 та 30,5234) відхилення від заданої точки (50,4500 та 30,5230) за формулою (3.1):

$$\text{СКВ} = \sqrt{(50,4501 - 50,45)^2 + (30,5234 - 30,523)^2} = 17 \cdot 10^{-8}.$$

Розрахунок для другої точки (50,4505 та 30,5238) відхилення від заданої точки (50,4504 та 30,5237) за формулою (3.1):

$$\text{СКВ} = \sqrt{(50,4505 - 50,4504)^2 + (30,5238 - 30,5237)^2} = 2 \cdot 10^{-8}.$$

Розрахунок для третьої точки (50,4509 та 30,5242) відхилення від заданої точки (50,4508 та 30,5241) за формулою (3.1):

$$\text{СКВ} = \sqrt{(50,4509 - 50,4508)^2 + (30,5242 - 30,5241)^2} = 2 \cdot 10^{-8}.$$

СКВ по трьом точкам становитиме:

$$\text{СКВ} = \sqrt{\frac{21 \times 10^{-8}}{3}} \approx 264 \cdot 10^{-6}.$$

Швидкодія системи визначає, як швидко система може реагувати на зміни у вхідних даних та переходити до нового стану.

Час перехідного процесу є важливим показником для динамічних характеристик системи управління.

Він дозволяє оцінити, наскільки ефективно система може стабілізуватися після зміни умов.

Розрахунок швидкодії системи (3.2):

$$T_{\text{шв}} = t_{\text{кін}} - t_{\text{поч}} = 5 - 0 = 5 \text{ секунд}, \quad (3.2)$$

де $t_{\text{кін}}$ – час початку зміни маршруту або події;

$t_{\text{поч}}$ – час досягнення нового стабільного стану після зміни

Передаточна функція замкненої системи для зворотного зв'язку одиничного від'ємного зв'язку має вигляд:

$$W(s) = \frac{1}{s^4 + 1,1s^3 + 2,75s^2 + 1,25s}$$

Використовуючи передаточну функцію було побудовано віртуальний макет для моделювання САУ (рис. 3.20).

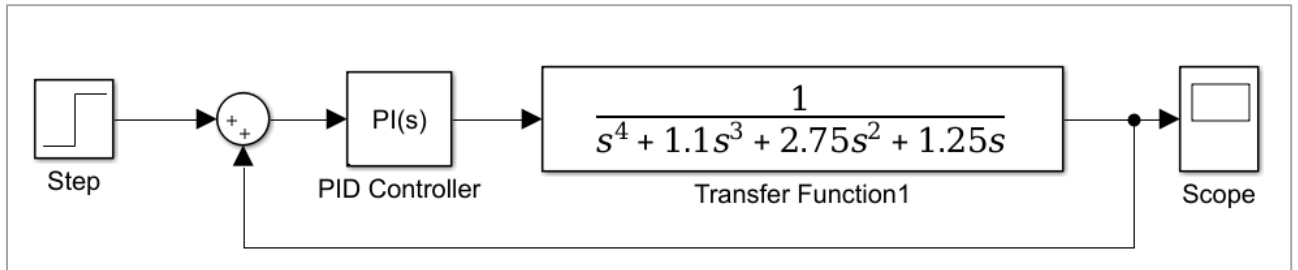


Рисунок 3.20 – Віртуальний макет для моделювання перехідної характеристики САУ

Після запуску віртуального макету перехідної характеристик САУ було отримано графік перехідної характеристики (рис. 3.21).

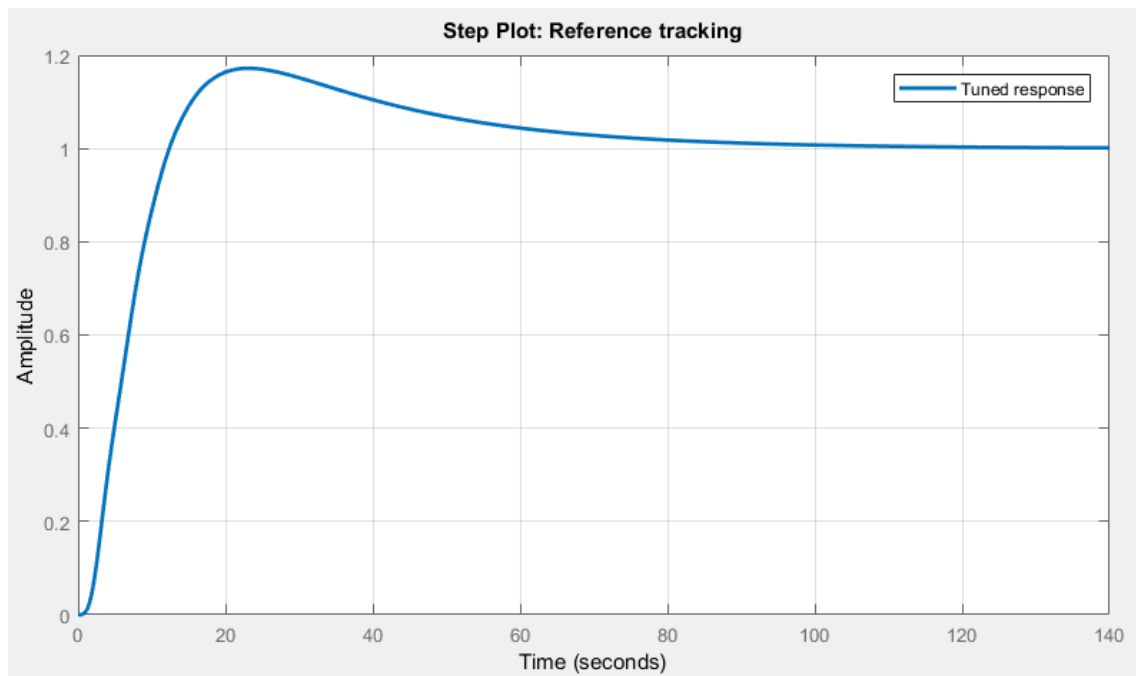


Рисунок 3.21 – Графік перехідної характеристики

З графіку перехідної характеристики було отримано його налаштування (рис. 3.22).

Controller Parameters	
	Tuned
P	-0.17471
I	-0.0055286
D	n/a
N	n/a

Performance and Robustness	
	Tuned
Rise time	8.1 seconds
Settling time	77.7 seconds
Overshoot	17.2 %
Peak	1.17
Gain margin	20.3 dB @ 1.04 rad/s
Phase margin	60 deg @ 0.139 rad/s
Closed-loop stability	Stable

Рисунок 3.22 – Налаштування отриманої перехідної функції

Скористаємося критерієм Гурвіца щоб визначити стійкість системи. Для цього нам потрібно скласти характеристичний поліном на основі передаточної функції замкнутої системи.

Характеристичний поліном визначається як знаменник передаточної функції $W(s)$:

$$p(s) = s^4 + 1,1s^3 + 2,75s^2 + 1,25s.$$

Розрахунок визначників Гурвіца:

$$\Delta^1 = 1,1,$$

$$\Delta^2 = \begin{vmatrix} 1,1 & 1,25 \\ 1 & 2,75 \end{vmatrix} = (1,1 \cdot 2,75) - (1 \cdot 1,25) = 1,725,$$

$$\Delta^3 = \begin{vmatrix} 1,1 & 1,25 & 0 \\ 1 & 2,75 & 0 \\ 0 & 1,1 & 1,25 \end{vmatrix} = 1,1 \cdot ((1,25 \cdot 2,75) - (1,1 \cdot 0)) = 3,4375,$$

$$\Delta^4 = \begin{vmatrix} 1,1 & 1,25 & 0 & 0 \\ 1 & 2,75 & 0 & 0 \\ 0 & 1,1 & 1,25 & 0 \\ 0 & 1 & 2,75 & 0 \end{vmatrix} = 1,1 \cdot ((2,75 \cdot 2,75) - (1,1 \cdot 1)) = 6,8125.$$

Оскільки всі визначники Гурвіца є додатними числами, то за критерієм Гурвіца система з одиничним від'ємним зворотним зв'язком є стійкою у замкнутому стані.

Таким чином, ми встановили, що система буде стійкою, коли працює у замкнутому режимі з одиничним від'ємним зворотним зв'язком.

Розрахунок загальної потужності компонентів (3.3):

$$P_3 = P_w + P_{gps} + P_{lcd} + P_{cam} = 160 + 30 + 20 + 160 = 370 \text{ мА}, \quad (3.3)$$

де P_w – потужність модуля ESP WROOM 32;

P_{gps} – потужність модуля GPS NEO6M;

P_{lcd} – потужність дисплея LCD 1602;

P_{cam} – потужність модуля ESP32-CAM.

Павербанк має ємність 20000 мАг та потужність 18 Вт. Перш ніж використовувати ємність, потрібно врахувати втрати при перетворенні напруги (ефективність перетворювача). Зазвичай ККД павербанків близько 90%, тому ефективна ємність буде (3.4):

$$C_e = C_{\Pi} \times A_{\Pi} = 20000 \times 0,9 = 18000 \text{ мАг}, \quad (3.4)$$

де C_{Π} – ємність павербанку;

A_{Π} – коефіцієнт корисної дії павербанку.

Для розрахунку часу роботи потрібно розділити ємність акумулятора на загальну споживану потужність (3.5):

$$t = \frac{C_e}{P_3} = \frac{18000}{370} \approx 48,65 \text{ годин}, \quad (3.5)$$

де C_e – ефективна ємність акумулятора;

P_3 – сподівана потужність.

Отже, апаратний модуль працюватиме приблизно 48,65 годин від повністю зарядженого павербанку ємністю 20000 мАг, враховуючи ККД перетворювача.

3.4 Висновки до третього розділу

У цьому розділі було виконано розробку апаратного забезпечення модуля відстеження мобільного робота на карті місцевості, обрано такі електронні компоненти для модуля: GPS-модуль NEO-6M, LCD дисплей 1602 з шиною I2C, модуль ESP-WROOM-32 та модуль ESP32-CAM. Проведено симуляцію підключення віртуальних компонентів у середовищі Wokwi та після отриманих результатів, було придбано й під'єднано фізичні електронні елементи згідно віртуального макету. Для фізичного макету було побудовано 3D-модель коробочки у середовищі проектування Fusion 360 для компактного розташування електронних елементів, нарізано на слої у застосунку EasyWare, надруковано на 3D-принтері та зібрано всі компоненти у єдиний корпус. Також було розраховано час роботи апаратного модуля, середнього квадратичного відхилення та швидкодійність системи, також побудовано графік перехідної характеристики.

4 РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ МОДУЛЯ ВІДОБРАЖЕННЯ МОБІЛЬНОГО РОБОТА

4.1 Вибір програмних компонентів

Для розробки автоматизованого модуля відстеження та відображення місцезнаходження мобільного робота на карті місцевості була обрана мова програмування C#.

C# є сучасною мовою програмування, яка постійно оновлюється та підтримується корпорацією Microsoft. Вона забезпечує широкий набір інструментів та бібліотек для різних завдань.

Ця мова програмування підтримує об'єктно-орієнтоване програмування (ООП), що сприяє структурованості та реюзабельності коду.

Мова C# тісно інтегрована з .NET Framework, що надає доступ до великої кількості готових компонентів та бібліотек.

Для обміну інформації щодо координат мобільного робота з автоматизованим модулем було обрано сервіс ThingSpeak.

ThingSpeak – це хмарний сервіс для збирання, зберігання, аналізу та візуалізації даних з пристроїв Інтернету речей (IoT). Сервіс дозволяє передавати координати робота на сервер, де вони зберігаються, обробляються та візуалізуються у реальному часі.

ThingSpeak дозволяє збирати GPS-дані з мобільного робота через HTTP-запити. Дані передаються у вигляді JSON, XML або CSV форматів. Дані про координати робота зберігаються в каналах, які містять поля для широти, довготи, часової мітки та інших параметрів, необхідних для точного відстеження. Сервіс інтегрується з MATLAB, що дозволяє виконувати складний аналіз GPS-даних, наприклад, для виявлення патернів руху або розрахунку відстаней.

ThingSpeak є потужним інструментом для роботи з GPS-даними мобільних роботів, забезпечуючи зручний інтерфейс для збирання, зберігання, аналізу та візуалізації даних. Використання цього сервісу дозволяє ефективно відстежувати та аналізувати переміщення мобільного робота в реальному часі.

Для відображення місцезнаходження мобільного робота на карті місцевості обрана бібліотека GMap.NET.

Gmap.NET підтримує роботу з різними картографічними сервісами, такими як Google Maps, Bing Maps, OpenStreetMap тощо. Це дає можливість вибирати найбільш підходящий картографічний сервіс для конкретних завдань.

Бібліотека дозволяє легко масштабувати та переміщувати карту, а також додавати на неї інтерактивні елементи, такі як маркери та полігони. Gmap.NET має простий та інтуїтивно зрозумілий API, що полегшує його інтеграцію в проекти на C#.

Для створення графічного інтерфейсу користувача (GUI) обрана платформа Windows Forms (WinForms).

WinForms дозволяє швидко створювати GUI завдяки візуальному дизайнеру у Visual Studio та надає великий набір готових контролів, які можна легко налаштувати та використовувати в проекті. Це скорочує час розробки та спрощує процес створення інтерфейсу.

WinForms є стабільною та перевіреною платформою, яка вже багато років використовується для розробки різноманітних додатків під Windows.

Для програмованого модуля вибрано інтегроване середовище розробки Visual Studio, бо воно надає потужні інструменти для редагування коду, налагодження, тестування та керування проектами та забезпечує повну підтримку мови C# та платформи WinForms, включаючи візуальний дизайнер форм. В Visual Studio легко інтегрувати та налаштувати бібліотеку Gmap.NET для роботи з картами.

Програмування апаратного модуля буде виконуватися у середовищі розробки Arduino IDE.

Arduino IDE (Integrated Development Environment) – це популярне середовище розробки для програмування мікроконтролерів сімейства Arduino і не тільки. Воно забезпечує простий та інтуїтивно зрозумілий інтерфейс для написання, компіляції та завантаження коду на мікроконтролери.

Arduino IDE підтримує широкий спектр мікроконтролерів та апаратних платформ, включаючи різні модулі GPS, датчики та інші компоненти, що можуть бути використані в проекті.

Інтерфейс Arduino IDE розроблений з урахуванням початківців та забезпечує легкий старт для розробників. Це дозволяє швидко налаштувати та розпочати програмування апаратного модуля.

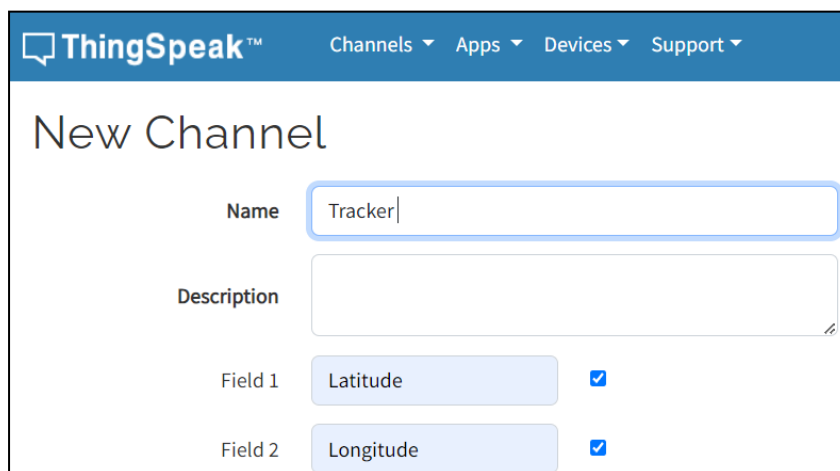
Існує велика кількість готових бібліотек для роботи з різними датчиками та модулями. Це спрощує процес інтеграції різних апаратних компонентів, таких як GPS-модулі.

Обрані програмні компоненти – мова програмування C#, бібліотека Gmap.NET, платформа WinForms, інтегроване середовище розробки Visual Studio та Arduino IDE, й сервіс ThingSpeak – забезпечують ефективну та зручну розробку автоматизованого модуля для відстеження та відображення місцезнаходження мобільного робота на карті місцевості. Ці компоненти забезпечують необхідну функціональність, простоту використання та надійність, що є важливими критеріями для успішної реалізації проекту.

4.2 Опис програмних функцій та модулів

Розробка автоматизованого модуля відстеження та відображення місцезнаходження мобільного робота включає кілька основних функцій та модулів, які забезпечують його ефективну роботу, а саме: програмування апаратного модуля для відправлення координат на сервіс у середовищі Arduino IDE, створення полів на сервісі ThingSpeak для координат, програмування модуля для роботи з користувачем.

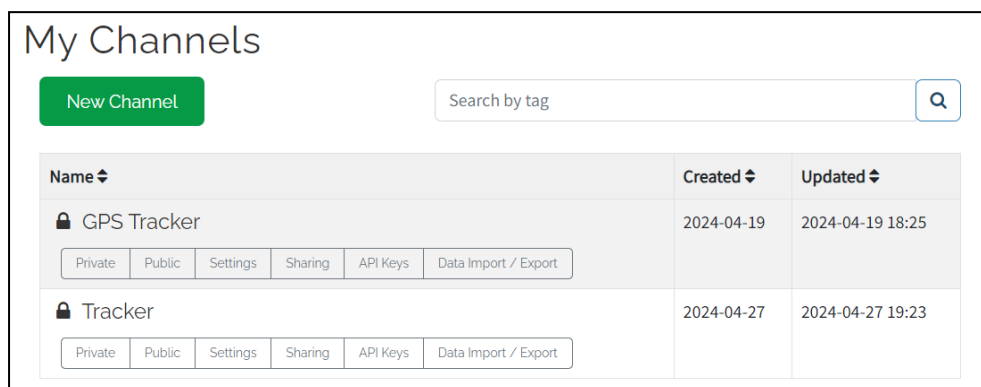
Для початку потрібно створити поля на сервісі ThingSpeak для записування координат мобільного робота. Зареєструвавшись, створюємо канал з назвою Tracker та полями longitude та latitude(рис. 4.1).



The screenshot shows the 'New Channel' page on ThingSpeak. The 'Name' field is filled with 'Tracker'. The 'Description' field is empty. Under 'Field 1', 'Latitude' is selected with a checked checkbox. Under 'Field 2', 'Longitude' is selected with a checked checkbox.

Рисунок 4.1 – Створення каналу Tracker

Після створення каналу було отримано його візуальне представлення у відповідному списку на сервісі ThingSpeak (рис. 4.2).



The screenshot shows the 'My Channels' page. There is a 'New Channel' button and a search bar. Below is a table of channels:

Name	Created	Updated
GPS Tracker Private Public Settings Sharing API Keys Data Import / Export	2024-04-19	2024-04-19 18:25
Tracker Private Public Settings Sharing API Keys Data Import / Export	2024-04-27	2024-04-27 19:23

Рисунок 4.2 – Створений канал Tracker

Тепер переходимо до написання коду для модуля ESP-WROOM-32. У середовищі розробки Arduino IDE потрібно додати бібліотеки для коректного функціонування коду, а саме: LiquidCrystal I2C, TinyGPSPlus, ThingSpeak (рис. 4.3).

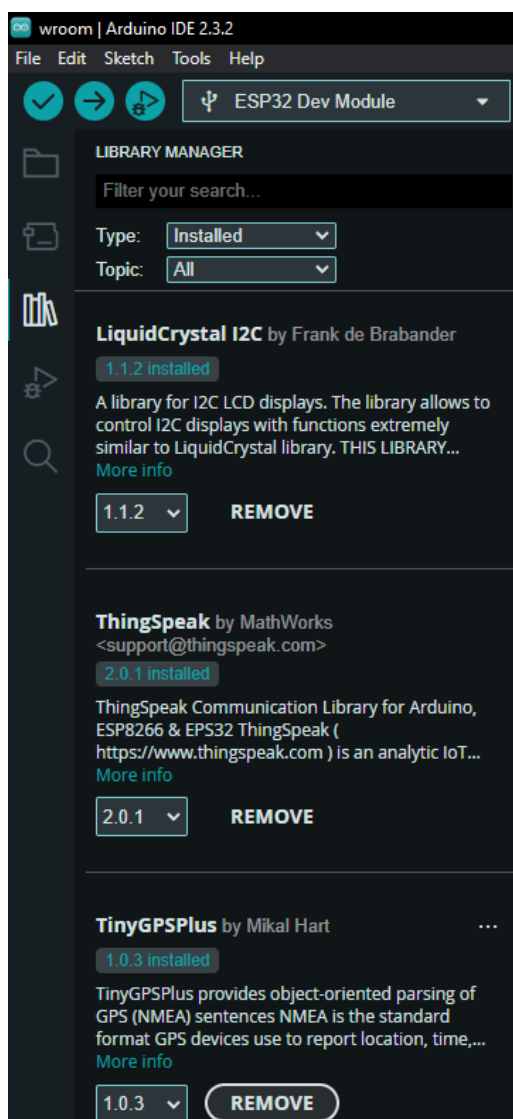


Рисунок 4.3 – Додавання бібліотеки

Для того щоб записувати координати у відповідні поля на сервісі, використовуються API ключі (рис. 4.4).

Write API Key

Key

[Generate New Write API Key](#)

Рисунок 4.4 – API ключ на запис даних

У результаті виконання програмного коду, модуль ESP-WROOM-32 підключається до точки WiFi та чекає координати з GPS-модуля, коли буде виконана фіксація з супутниками кожну секунду модуль буде передавати координати мобільного робота на сервіс та виводити дані на дисплей.

У тому ж середовищі розробки для показу відео руху мобільного робота у реальному часі запрограмовано модуль ESP32-CAM, який також підключається до WiFi та транслює відео потік.

Створення програмного інтерфейсу користувача виконується у програмному середовищі Visual Studio на платформі WinForms.

Для того щоб застосовувати карту потрібно завантажити до проекту бібліотеку Gmap.NET із маркету NuGet (рис. 4.5).

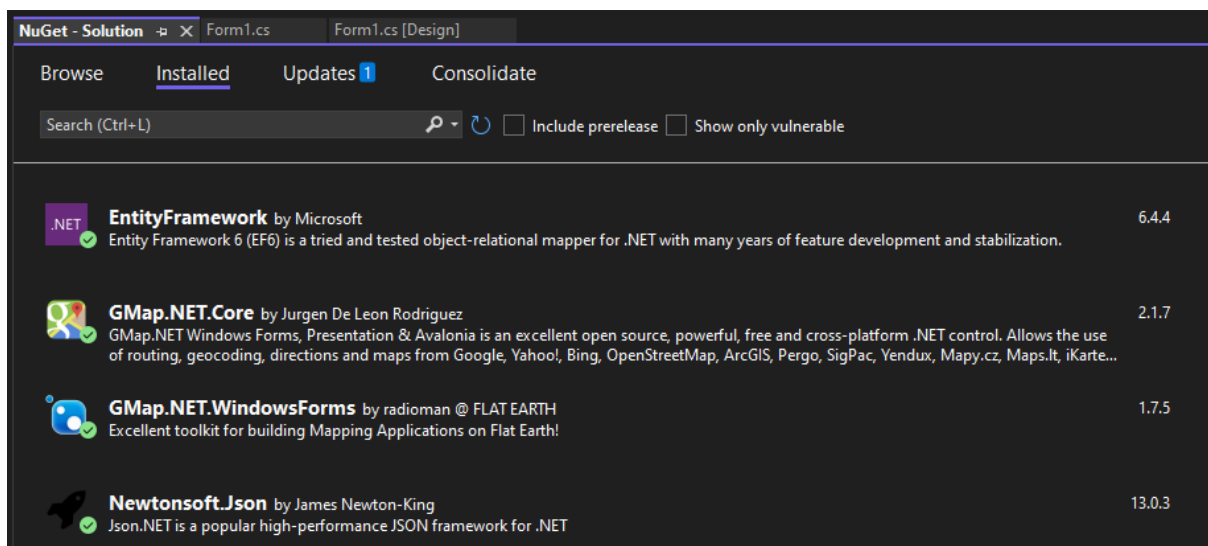


Рисунок 4.5 – Встановлені бібліотеки

Користувач матиме можливість слідкувати за місцезнаходженням мобільного робота та його рухом.

У інтерфейсі буде п'ять кнопок. Перша кнопка матиме назву Location, при натисканні на неї буде ставитися маркер, який відповідатиме розташуванню мобільного робота. Друга кнопка називатиметься Start, на карті буде ставитися маркер синього кольору, що означитиме початок руху, та малюватиметься шлях.

Третя кнопка Stop – завершення слідування за мобільним роботом, ставитиметься зелений маркер. Четверта кнопка Clear відповідатиме за очищення намальованих елементів. П'ята кнопка Video – виконуватиметься відкриття сторінки, на якій буде відображатися відео потік руху мобільного роботу у реальному часі (рис. 4.6).

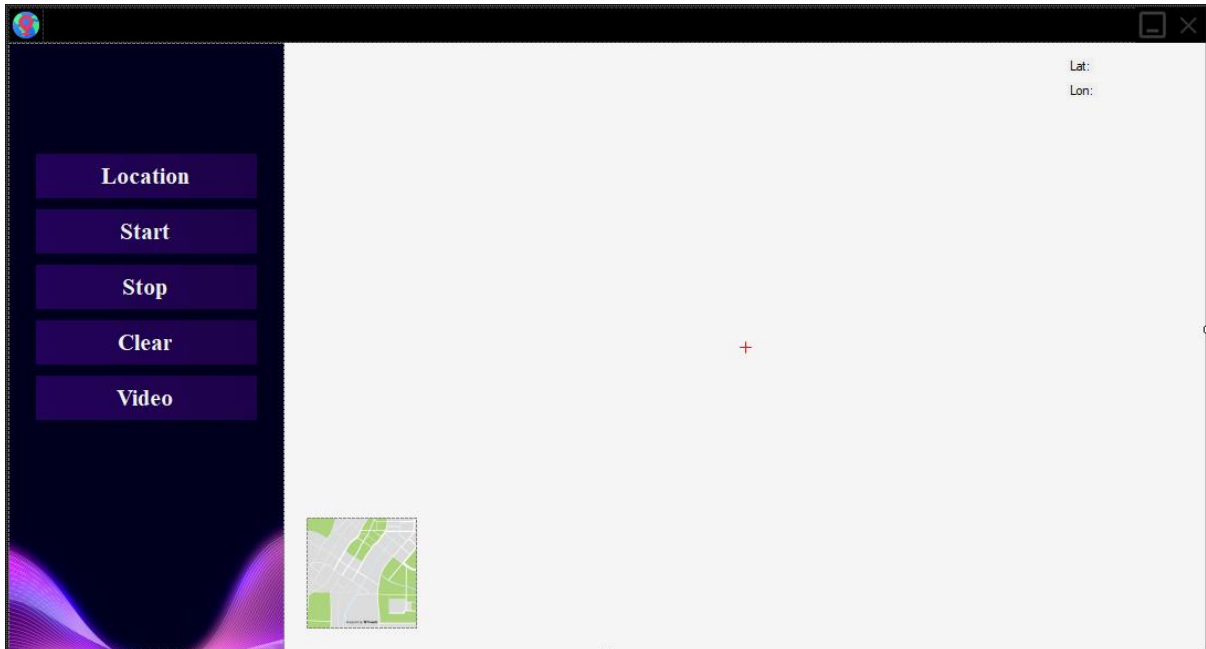


Рисунок 4.6 – Інтерфейс користувача

Також повинна бути можливість зміни звичайної карти на зображення зі супутника. Це дозволить користувачам перемикатися між різними режимами відображення в залежності від їхніх потреб. Наприклад, для деяких завдань може бути зручніше використовувати звичайну карту з чіткими лініями вулиць та назвами місць, тоді як для інших завдань більш корисним буде супутникове зображення, яке надає детальне уявлення про реальний стан місцевості.

Інтерфейс системи також включає функцію малювання шляху руху для мобільного робота при двійному натисканні на ліву кнопку миші (рис. 4.7).

Ця функція дозволяє користувачу легко визначати та візуалізувати маршрут, який повинен пройти робот. Користувач може клікати на різні точки на

карті, і система автоматично буде з'єднувати ці точки, формуючи шлях. Це значно спрощує процес планування маршрутів і дозволяє оперативно вносити зміни при необхідності.

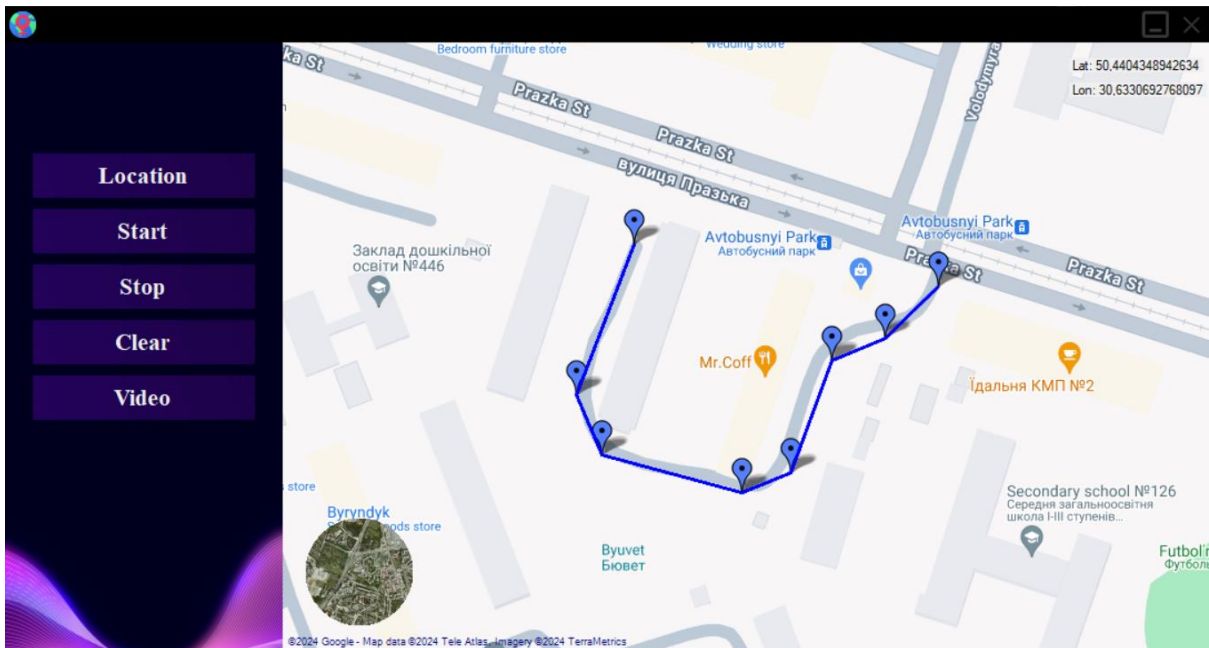


Рисунок 4.7 – Шлях руху мобільного робота

При двійному натисканні на праву кнопку миші система буде малювати поле, в якому буде виконуватися слідкування за роботом (рис. 4.8).

Ця функція дозволяє встановити межі, в межах яких система буде відстежувати робота. Це може бути корисно для обмеження робочої зони робота, наприклад, у випадках, коли робот повинен працювати лише в певній області чи уникати небезпечних зон.

Реалізація цих функцій потребує інтеграції з картографічними сервісами, таким як OpenStreetMap, які підтримують зміну типів відображення карти та забезпечують можливості для роботи з інтерактивними елементами. Додавання цих функцій до інтерфейсу значно підвищить зручність використання системи та надасть користувачам більше можливостей для управління та контролю мобільним роботом.

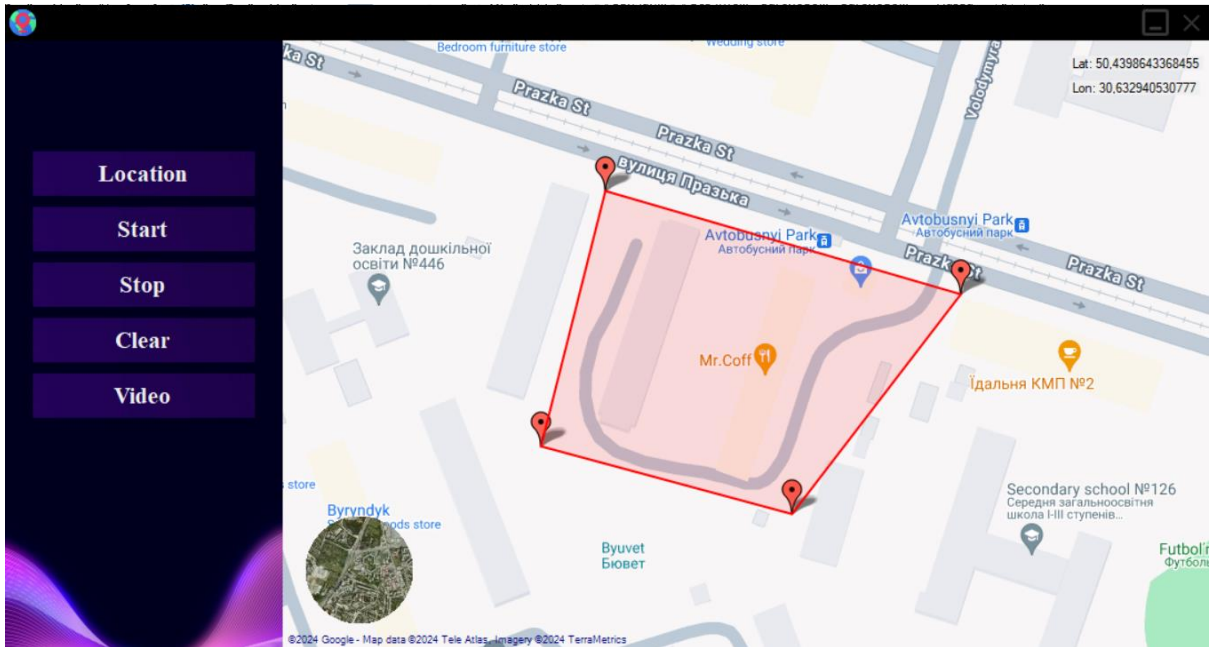


Рисунок 4.8 – Поле у якому рухається мобільний робот

При наведенні курсором на ділянку карти, справа зверху відображаються відповідні довгота та широта на місцевості (рис. 4.9).

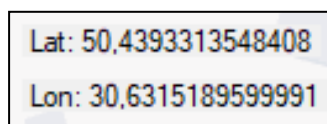


Рисунок 4.9 – Координати відповідно до курсора на карті

4.3 Тестування програми та опис результату роботи модуля

Для початку роботи автоматизованого модуля відстеження та відображення місцезнаходження мобільного робота на карті місцевості потрібно подати напругу на апаратний модуль, включити точку доступу Wifi та дочекатися червоного індикатора на GPS-модулі, що означитиме виправлення супутників й отримання координат.

Далі потрібно запустити програмний модуль та побачити її інтерфейс (рис. 4.10).

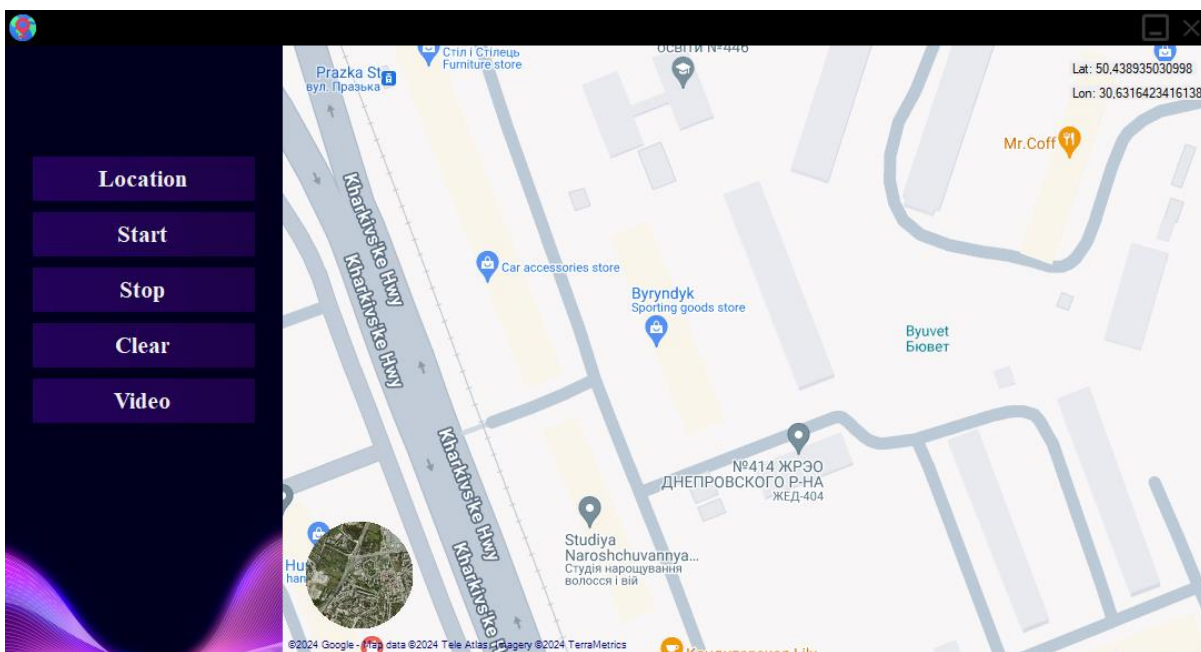


Рисунок 4.10 – Початкове вікно програмного модуля

Перейдемо до вигляду карти з супутника, бо таке зображення дозволяє побачити реальні об'єкти на земній поверхні та стає зрозумілим контекст для орієнтації на місцевості (рис. 4.11).

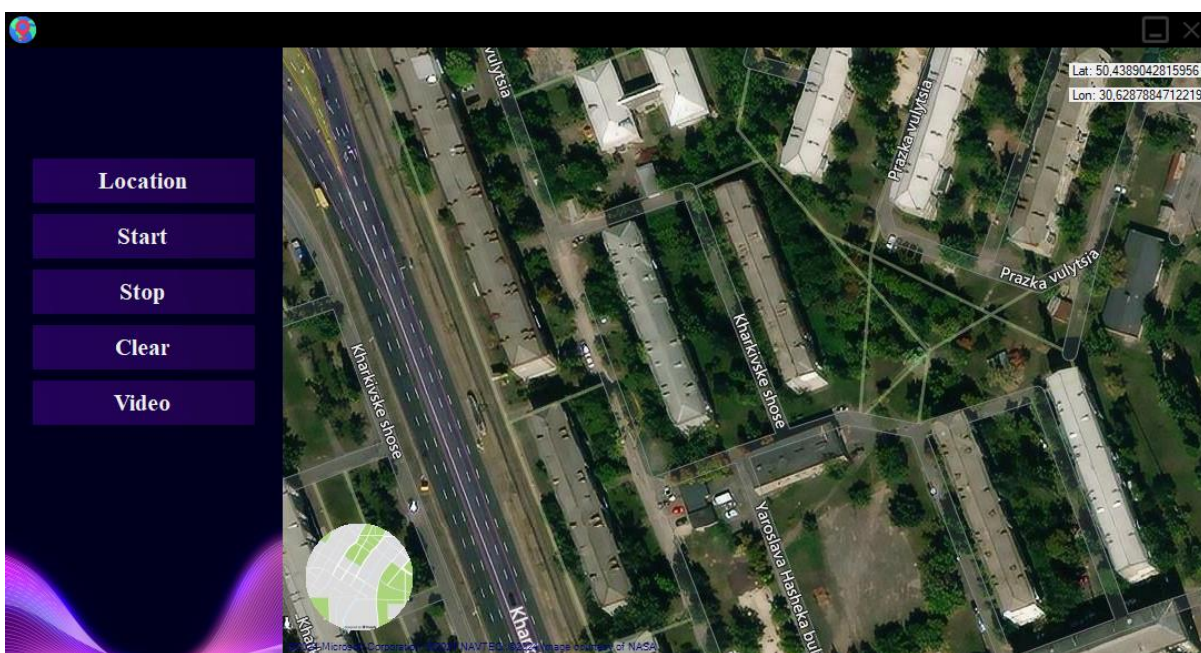


Рисунок 4.11 – Вигляд карти з супутника

Натискаючи на кнопку "Location" у користувацькому інтерфейсі, ми отримуємо актуальне місцезнаходження мобільного робота.

Це місцезнаходження відображається на карті у реальному часі, що дозволяє користувачу бачити точну позицію робота в даний момент (рис. 4.12).

Інформація про місцезнаходження отримується за допомогою GPS-модуля, який забезпечує точність визначення координат.

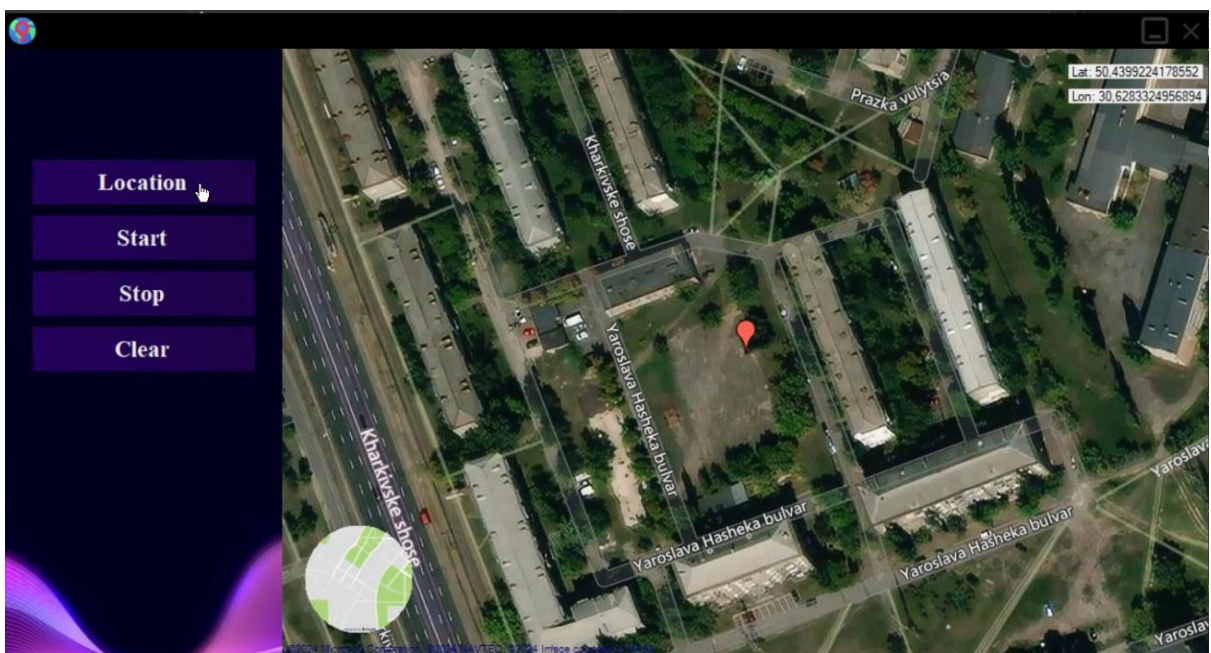


Рисунок 4.12 – Місцезнаходження мобільного робота

Після отримання місцезнаходження робота, ми переходимо до визначення робочого поля, у якому відбуватиметься слідкування за мобільним роботом.

Для цього користувач може використати інструменти малювання на карті, що дозволяє окреслити межі зони, в межах якої робот повинен здійснювати свої маневри (рис. 4.13).

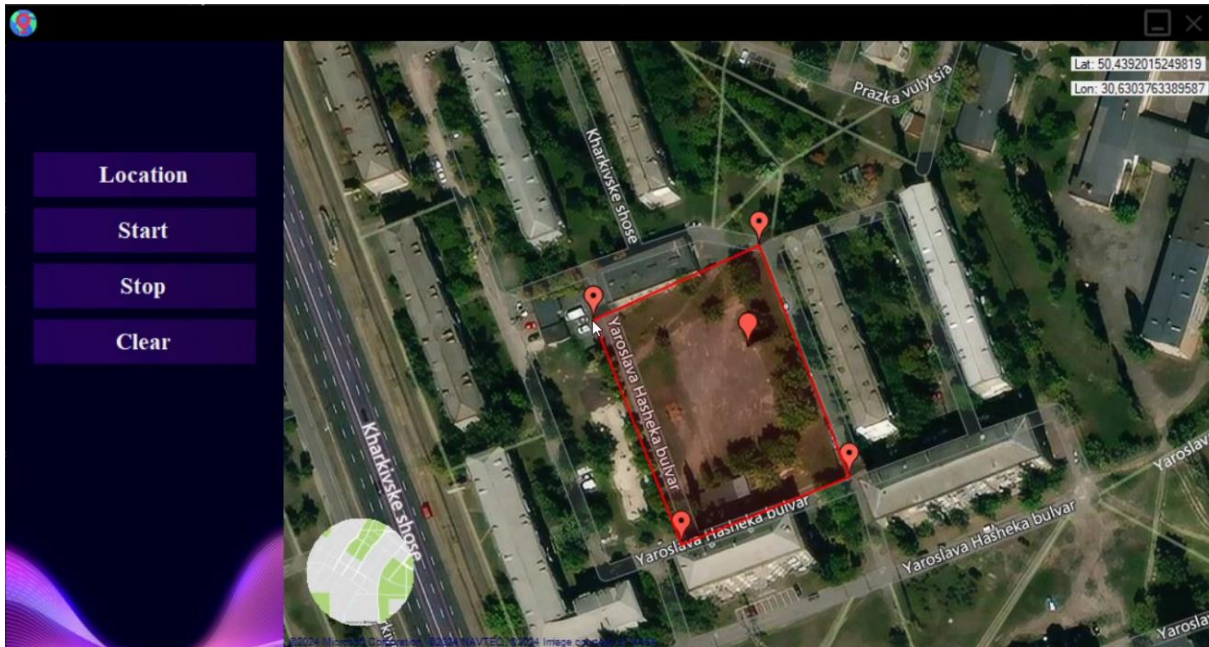


Рисунок 4.13 – Назначене поле мобільного робота

Після налаштування робочого поля та визначення початкового місцерозташування, користувач натискає на кнопку "Start", щоб розпочати відображення шляху руху мобільного робота.

З цього моменту на карті починають з'являтися елементи, що вказують на траєкторію руху робота. Синій маркер позначає початкову точку руху робота, з якої він розпочав свій шлях. Це дозволяє легко ідентифікувати стартову позицію на карті.

По мірі руху робота, на карті з'являється жовта лінія, яка відображає шлях його руху. Ця лінія оновлюється в реальному часі, надаючи користувачу можливість слідкувати за траєкторією робота.

Жовта лінія є візуальним представленням пройденого маршруту, що дозволяє оцінювати точність виконання завдань та аналізувати рух робота (рис. 4.14).

Такий інтерактивний підхід до відображення руху робота на карті значно полегшує контроль за його роботою, забезпечуючи візуальну інформацію про всі ключові аспекти його переміщень.

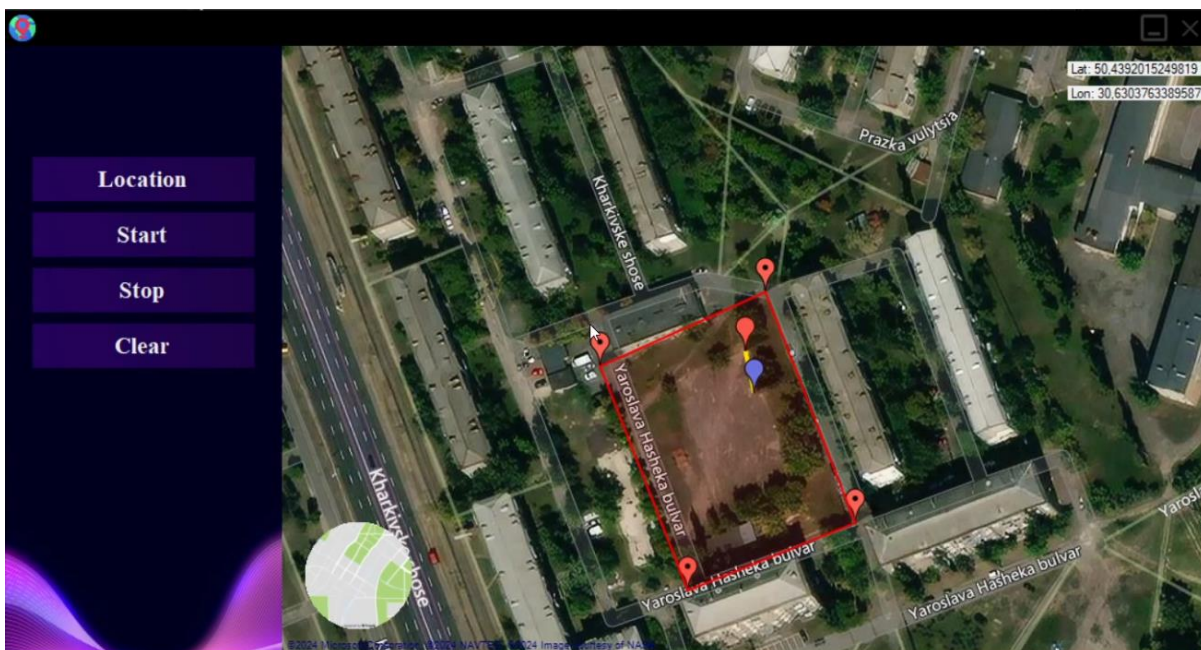


Рисунок 4.14 – Початок відображення руху мобільного робота

Натискаючи на кнопку "Stop", користувач завершує процес слідування за шляхом руху мобільного робота. Це завершення відображається на карті спеціальним маркером. Зелений маркер з'являється на карті, позначаючи кінцеву точку руху робота, що вказує на успішне завершення маршруту (рис. 4.15).

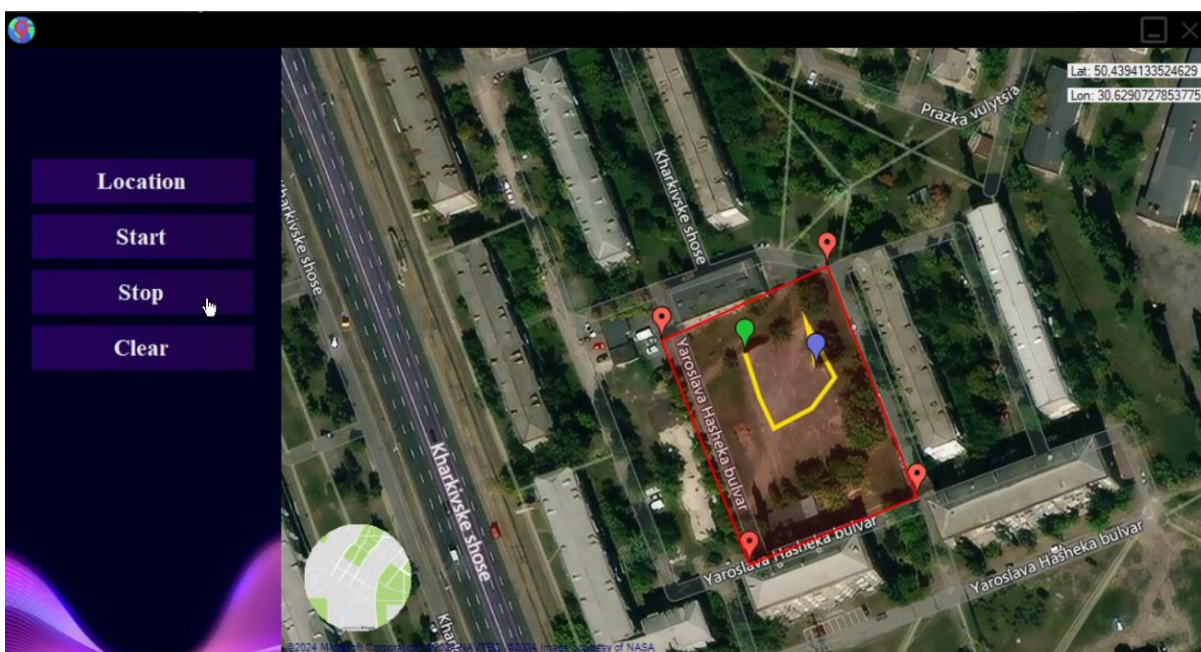


Рисунок 4.15 – Завершення відображення руху мобільного робота

Далі якщо мобільний робот продовжує свій рух буде відображатися його поточне місцезнаходження (рис. 4.16).

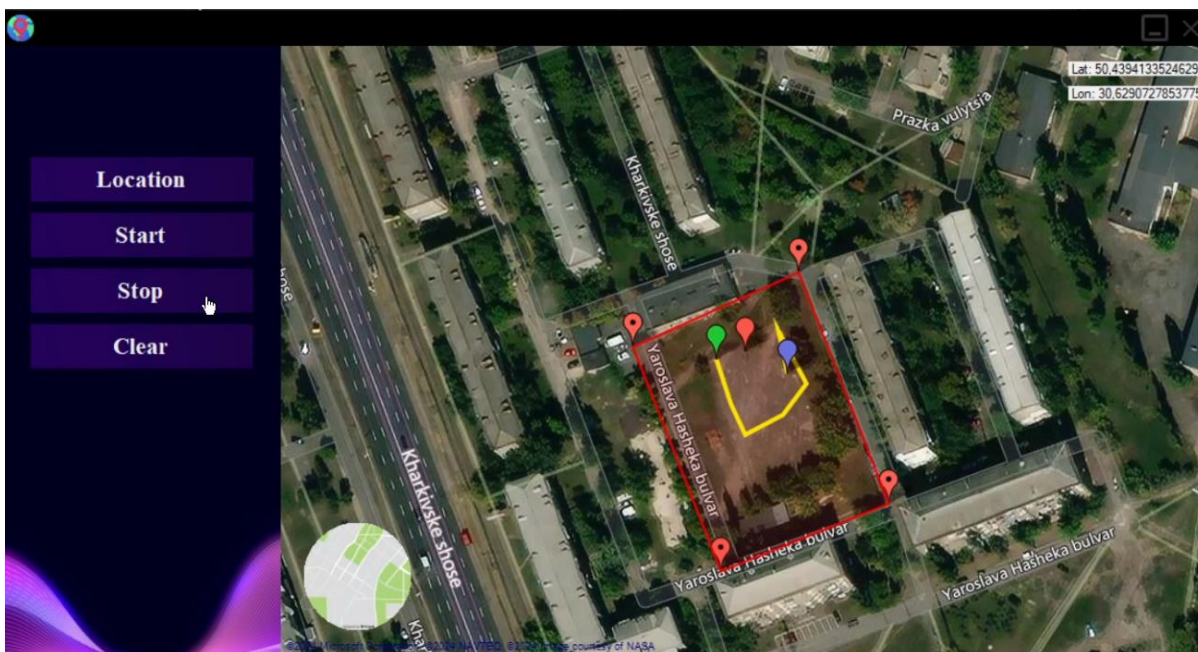


Рисунок 4.16 – Продовження руху мобільного робота

При натисканні на кнопку Clear виконується очищення всіх назначених полів та шляхів руху (рис. 4.17).

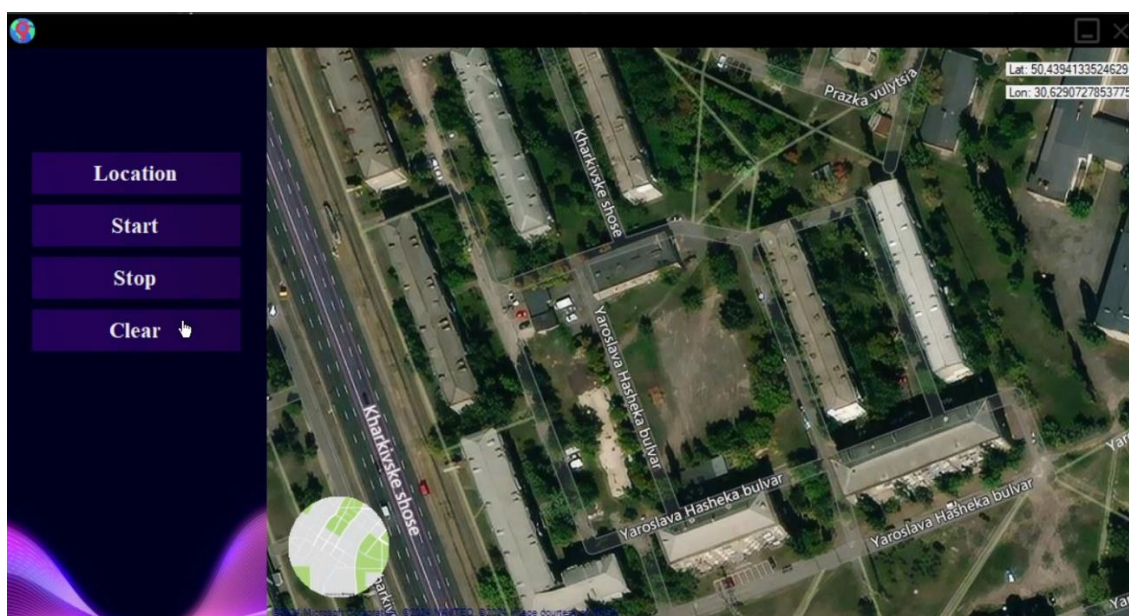


Рисунок 4.17 – Очищена карта

Натискання на кнопку Video призведе до відкриття сторінки в браузері, на якій буде відображатися відеопотік з мобільного робота та його налаштування (рис. 4.18).

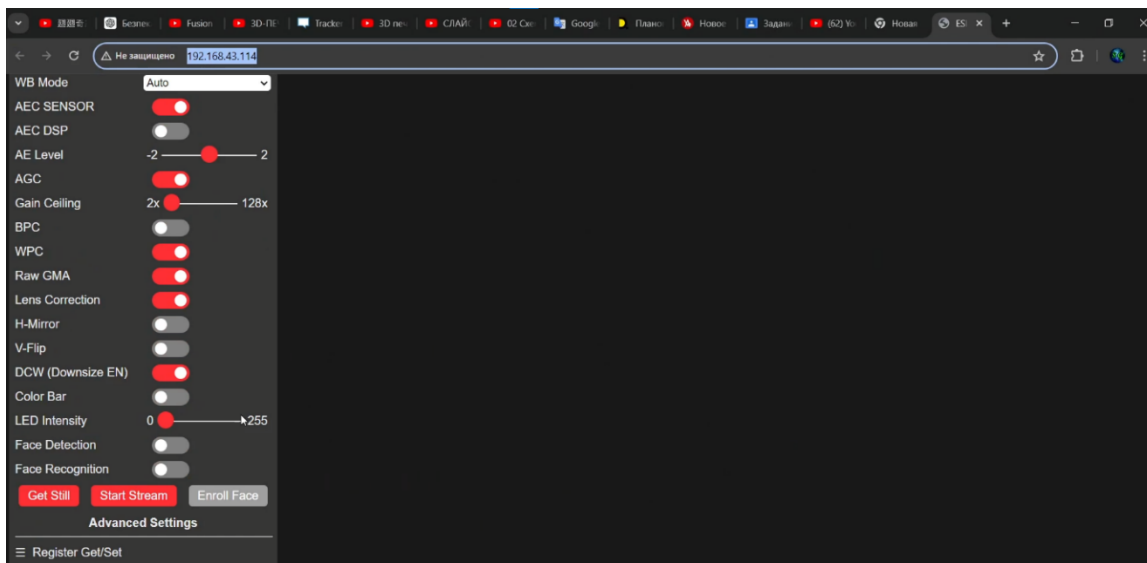


Рисунок 4.18 – Початковий вигляд інтерфейсу для камери

Для початку відеотрасляції потрібно натиснути на кнопку Start Stream, а для завершення Stop Sream (рис. 4.19).

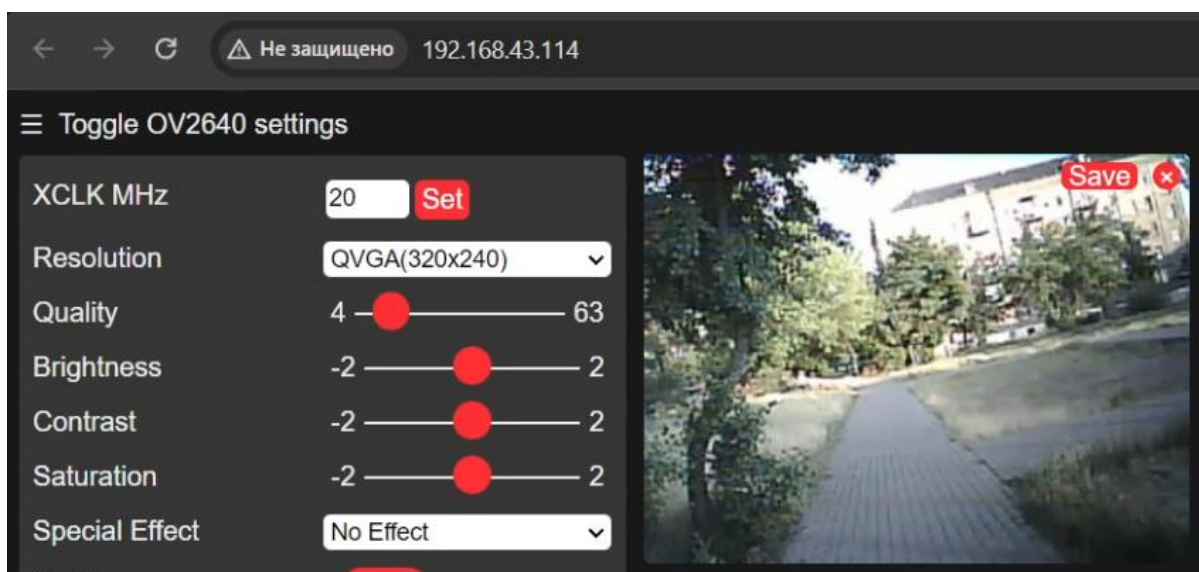


Рисунок 4.19 – Відеотрансляція з мобільного робота

4.4 Висновки до четвертого розділу

У цьому розділі було виконано вибір програмних компонентів, а саме: мова програмування C#, бібліотека Gmap.NET, платформа Windows Forms, середовище розробки Visual Studio, хмарний сервіс ThingSpeak. Описано програмні функції та модулі, які використані у проекті:

- відстеження на карті місцевості мобільного робота;
- відображення на карті місцевості мобільного робота;
- відстеження шляху мобільного робота;
- побудова шляху та поля дії мобільного робота;
- відображення відеопотоку з мобільного робота;
- виконання всіх дій у реальному часі.

Також було виконано тестування апаратного та програмного модуля, який показав високу точність і надійність у відстеженні та відображенні місцезнаходження мобільного робота на карті місцевості в реальному часі.

5 ОХОРОНА ПРАЦІ

5.1 Технічна безпека

При розробці та впровадженні автоматизованої системи відстеження та відображення місцезнаходження мобільного робота на карті місцевості особливу увагу було приділено питанням охорони праці. Важливість забезпечення безпечних умов роботи як для розробників, так і для користувачів системи не можна переоцінити [18 - 19].

На етапі проектування та складання апаратної частини були дотримані всі норми електробезпеки. Всі електронні компоненти, такі як GPS-модуль NEO-6M, LCD дисплей 1602 з шиною I2C, модуль ESP-WROOM-32 та модуль ESP32-CAM, обрані з урахуванням їх відповідності міжнародним стандартам безпеки. Для захисту від статичної електрики під час роботи з електронними компонентами використовувалися антистатичні браслети та підкладки.

Корпус модуля, розроблений в середовищі проектування Fusion 360, був спроектований для забезпечення максимального захисту від механічних пошкоджень. Він виготовлений з міцного матеріалу, що витримує значні фізичні навантаження, а також захищає внутрішні компоненти від пилу та вологи. Корпус надруковано на 3D-принтері, що забезпечує точність виготовлення і відповідність проектним вимогам.

На етапі програмування та тестування використовувалися методи, що зменшують ризики виникнення програмних збоїв та помилок. Використання стабільних і перевірених бібліотек, таких як GMap.NET, забезпечило високу надійність програмного забезпечення. Тестування програмних модулів проводилося ретельно, щоб мінімізувати можливість виникнення помилок під час експлуатації системи.

5.2 Організаційна безпека

Для зменшення ризиків, пов'язаних з людським фактором, був створений інтуїтивно зрозумілий інтерфейс користувача. Це допомагає запобігти помилковим діям користувачів, що можуть призвести до збоїв у роботі системи або неправильного відображення місцезнаходження мобільного робота.

При роботі з хмарним сервісом ThingSpeak було забезпечено високий рівень інформаційної безпеки. Всі дані, що передаються між модулем і хмарним сервісом, шифруються, що унеможлиблює їх перехоплення та несанкціонований доступ. Це важливо для захисту конфіденційної інформації та забезпечення надійності системи.

Під час монтажу та експлуатації модуля були дотримані всі необхідні заходи безпеки. Всі роботи з підключення та налаштування електронних компонентів проводилися з дотриманням правил охорони праці, що включає використання відповідних інструментів та засобів індивідуального захисту. Користувачі системи були інструктовані щодо безпечного використання модуля, що додатково знижує ризики травм та пошкоджень.

Таким чином, в процесі розробки та впровадження автоматизованого модуля відстеження та відображення місцезнаходження мобільного робота були враховані всі необхідні заходи охорони праці. Це забезпечило створення безпечної, надійної та ефективною системи, яка відповідає всім стандартам безпеки і може бути використана без ризику для здоров'я і життя користувачів.

ВИСНОВКИ

У цій атестаційній роботі було успішно реалізовано автоматизовану систему відстеження та відображення місцезнаходження мобільного робота на карті місцевості. Після аналізу технічного завдання були визначені вимоги до системи, які включали точне визначення місцезнаходження робота, підключення до картографічних сервісів та створення користувацького інтерфейсу. Ці вимоги стали основою для подальшої роботи над проектом.

На етапі розробки архітектури модуля було описано структурну схему та алгоритми його роботи, а також протоколи взаємодії з мобільним роботом. Ці рішення створили основу для побудови ефективної системи відстеження, яка може бути легко інтегрована в різні автоматизовані платформи.

Для реалізації апаратної частини було обрано такі компоненти, як GPS-модуль NEO-6M, LCD дисплей 1602 з шиною I2C, модуль ESP-WROOM-32 та модуль ESP32-CAM. Проведено симуляцію підключення у середовищі Wokwi, після чого придбано і підключено фізичні елементи відповідно до віртуального макету. Було розроблено та надруковано 3D-модель корпусу для компактного розташування електронних компонентів, що забезпечило зручність у зборці і експлуатації модуля.

Вибір програмних компонентів включав мову програмування C#, бібліотеку GMap.NET, платформу Windows Forms, середовище розробки Visual Studio та хмарний сервіс ThingSpeak. Це дозволило реалізувати необхідні функції для відстеження та відображення місцезнаходження мобільного робота, відстеження шляху, побудови поля дії та відображення відеопотоку з робота.

Після завершення всіх етапів було проведено комплексне тестування апаратного та програмного модулів, що підтвердило високу точність і надійність відстеження та відображення місцезнаходження мобільного робота в реальному часі.

Таким чином, розроблена система відстеження та відображення місцезнаходження мобільного робота повністю відповідає всім поставленим вимогам і може бути ефективно використаний у різних автоматизованих платформах. Реалізовані рішення забезпечують високу точність, надійність та інтеграцію з сучасними технологіями картографування і користувацькими інтерфейсами.

ПЕРЕЛІК ПОСИЛАНЬ

1. Невлюдов І. Ш. Технологія програмування промислових контролерів в інтегрованому середовищі CODESYS : навч. посіб. / І. Ш. Невлюдов, С. П. Новоселов, О. В. Сичова ; М-во освіти і науки України, Харків. нац. ун-т радіоелектроніки. – Харків : ХНУРЕ, 2019. – 264 с. : іл. – ISBN 978-966-659-265-4. DOI: 10.30837/978-966-659-265-4.

2. Невлюдов І. Ш. Застосування цифрових двійників технічних засобів автоматизації для розроблення програмно-технічних комплексів АСУ ТП : Навчальний посібник / І. Ш. Невлюдов, С. П. Новоселов, О. В. Сичова. – Харків: Видавництво Іванченка І. С., 2023. – 267 с. ISBN 978-617-8059-95-8, DOI: 10.30837/978-617-8059-95-8.

3. Novoselov S., Sychova O., Tesliuk S. Technology of multi-thread software motion control of a two-link manipulator with four freedom degree. Innovative integrated computer systems in strategic project management : Collective monograph edited by I. Linde, I. Chumachenko. Riga : ISMA. pp.151-163, 2022. DOI: <https://doi.org/10.30837/MMP.2022.151>. ISBN -13 (24) 978-9984-891-24-8.

4. Sergiy Novoselov, Oksana Sychova, Serhii Tesliuk. Algorithm for Finding the Optimal Way to Move a Mobile Platform Among Indefinite Obstacles. // IV International Scientific and Practical Conference «Theoretical and Applied Aspects of Device Development on Microcontrollers and FPGAs» MC&FPGA-2022, Kharkiv, Ukraine, June 23-24, 2022. – Kharkiv: NURE, MC&FPGA, 2022. – P. 21-23. DOI: DOI: 10.35598/mcfpga.2022.007.

5. I. Nevludov, S. Novoselov, O. Sychova, S. Tesliuk. Production workspace obstacle avoidance mobile robot trajectory modeling. Fifth International Scientific and Technical Conference «Computer and information systems and technologies», Kharkiv, April 22 – 23, 2021. - pp. 61-62. DOI: <https://doi.org/10.30837/csitic52021232778>.

6. ДСТУ 3008-15. Документація. Звіти у сфері науки та техніки. структура та правила оформлення. Введ. 2015-06-22. К. Держстандарт України, 2017. – 29 с.

7. Методичні вказівки з підготовки кваліфікаційної роботи для здобувачів першого (бакалаврського) рівня вищої освіти денної і заочної форми навчання спеціальності 151 «Автоматизація та комп'ютерно-інтегровані технології» освітньої програми «Автоматизація та комп'ютерно-інтегровані технології» / Упоряд.: І.Ш. Невлюдов, О.І. Филипенко, О.В. Токарева, С.П. Новоселов, О.В. Сичова. – Харків: ХНУРЕ, 2023. – 64 с.

8. Навчальний посібник з підготовки кваліфікаційної роботи бакалавра для здобувачів вищої освіти денної і заочної форм навчання спеціальності 151 «Автоматизація та комп'ютерно-інтегровані технології» освітньої програми «Автоматизація та комп'ютерно-інтегровані технології» : Навчальний посібник / І. Ш. Невлюдов, О. І. Филипенко, О. В. Токарева, С. П. Новоселов, О. В. Сичова. – Харків : Видавництво Іванченка І. С., 2023. – 151 с. ISBN 978-617-8059-94-1, DOI: 10.30837/978-617-8059-94-1.

9. What are Global Navigation Satellite Systems?. GPS & GNSS Equipment, Products & Solutions | NovAtel. [Електронний ресурс] / Режим доступу: <https://novatel.com/tech-talk/an-introduction-to-gnss/what-are-global-navigation-satellite-systems-gnss> (дата звернення: 16.04.2024).

10. Mobile Phones: Location Tracking. Surveillance Self-Defense. [Електронний ресурс] / Режим доступу: <https://ssd.eff.org/module/mobile-phones-location-tracking> (дата звернення: 17.04.2024).

11. Maps Made Easy - Features. Maps Made Easy - Home. [Електронний ресурс] / Режим доступу: <https://www.mapsmadeeasy.com/features> (дата звернення: 17.04.2024).

12. NEO-6M GPS-модуль GY-NEO6MV2 купити в Києві та Україні. *Arduino* в Україні. [Електронний ресурс] / Режим доступу: <https://arduino.ua/prod5017-neo-6m-gps-modyl-gy-gps-6mv2-neo6mv2> (дата звернення: 10.05.2024).

13. NEO-6 u-blox 6 GPS Modules Data Sheet. [Електронний ресурс] / Режим доступу: [https://content.u-blox.com/sites/default/files/products/documents/NEO-6_DataSheet_\(GPS.G6-HW-09005\).pdf](https://content.u-blox.com/sites/default/files/products/documents/NEO-6_DataSheet_(GPS.G6-HW-09005).pdf) (дата звернення: 12.05.2024).

14. Дисплей LCD 1602. [Електронний ресурс] / Режим доступу: https://rozetka.com.ua/ua/308767588/p308767588/?gad_source=1&gclid=CjwKCAjwyJqzBhBaEiwAWDRJVAtmDx7_L1UBLPvu9KpJ_3l2TTBBmK0nnKSNJqlrlOHa bo5X15PS9VoCEhwQAvD_BwE (дата звернення: 13.05.2024).

15. Модуль Arduino LCD1602 I2C AVR PIC. [Електронний ресурс] / Режим доступу: https://rozetka.com.ua/ua/346514130/p346514130/?gad_source=1&gclid=CjwKCAjwyJqzBhBaEiwAWDRJVJxmXYePzfrYgBZRnyt11vnN6kKGPCX5oYCALKvOuqW8YScig0k_sxoCz-QQAvD_BwE (дата звернення: 13.05.2024).

16. Модуль Wi-Fi Bluetooth ESP32 ESP-32S ESP32-WROOM-32 D0WDQ6 LWIP freertos 30pin. [Електронний ресурс] / Режим доступу: <https://prom.ua/ua/p1950938203-modul-bluetooth-esp32.html> (дата звернення: 13.05.2024).

17. Модуль розробки ESP32-CAM-MB з камерою OV2640 2Мп WiFi + Bluetooth, MicroSD. [Електронний ресурс] / Режим доступу: https://rozetka.com.ua/ua/423759117/p423759117/?gad_source=1&gclid=CjwKCAjwyJqzBhBaEiwAWDRJVACTrXT-ср_tEiv29TKZ0TgF_2EUy65xcjQS2_xEq8NBlo frAX21FRoCmv8QAvD_BwE (дата звернення: 13.05.2024).

18. Охорона праці: обов'язки роботодавців і працівників | Бериславська міська рада | Офіційний сайт. Бериславська міська рада | Офіційний сайт. [Електронний ресурс] / Режим доступу: <https://berislav-mr.gov.ua/news/ohorona-pratsi-obov-yazky-robotodavtsiv-i-pratsivnykiv/> (дата звернення: 14.06.2024).

19. Основи охорони праці: підручник / М. С. Одарченко, А. М. Одарченко, В. І. Степанов, Я. М. Черненко. – Х. : ХДУХТ, 2017. – 334 с.