

**НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ БІОРЕСУРСІВ І  
ПРИРОДОКОРИСТУВАННЯ УКРАЇНИ  
ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ**

**ДОПУСКАЄТЬСЯ ДО ЗАХИСТУ**

Завідувач кафедри

Комп'ютерних систем, мереж та кібербезпеки

Касаткін Д.Ю., к.пед.н., доц.

(підпис)

(ПІБ, вчене звання і ступінь)

« \_\_\_\_ » \_\_\_\_\_ 2025 р.

**КВАЛІФІКАЦІЙНА БАКАЛАВРСЬКА РОБОТА**

На тему: «Розробка комп'ютерної системи поливу з використанням дронів»

Спеціальність F7 «Комп'ютерна інженерія»

Гарант освітньої програми

к.фіз.-мат.н., доц.

(підпис)

/ Нікітенко Є.В. /

(ПІБ)

Керівник дипломного проекту: \_\_\_\_\_

(підпис)

/ Місюра М.Д. /

(ПІБ)

Виконав: \_\_\_\_\_

(підпис)

/ Дрозд В.В. /

(ПІБ)

**КИЇВ-2025**

**НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ БІОРЕСУРСІВ  
І ПРИРОДОКОРИСТУВАННЯ УКРАЇНИ  
ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ**

**«ЗАТВЕРДЖУЮ»**

**завідувач кафедри**

комп'ютерних систем, мереж та кібербезпеки

/ Касаткін Д.Ю., к.пед.н., доц. /

(підпис)

(ПІБ, вчене звання і ступінь)

«\_\_» \_\_\_\_\_ 20\_\_ р.

**З А В Д А Н Н Я**

**ДО ВИКОНАННЯ КВАЛІФІКАЦІЙНОЇ БАКАЛАВРСЬКОЇ СТУДЕНТУ**

Дрозда Володимира Володимировича

(прізвище, ім'я, по батькові)

Спеціальність (напрямок підготовки): комп'ютерна інженерія

Тема кваліфікаційної бакалаврської роботи: «Розробка комп'ютерної системи поливу з використанням дронів»

з

Термін подання завершеної роботи на кафедру \_\_\_\_\_

Вихідні дані до кваліфікаційної бакалаврської роботи література по використанню біометрії

в

е

р

д

Перелік питань, що підлягають розробці:

1. Аналіз існуючих систем поливу та моніторингу агро-даних

2. Вибір апаратної платформи

3. Розробка програмного забезпечення для маршрутизації дрона та управління поливом

4. Тестування та аналіз продуктивності

Перелік графічного матеріалу (за потреби) \_\_\_\_\_

а

к

Дата видачі завдання “ 10 ” \_\_\_\_\_ 04 \_\_\_\_\_ 2025 р.

з

Керівник кваліфікаційної роботи \_\_\_\_\_

Місюра М.Д., к.т.н.

М \_\_\_\_\_  
(підпис)

(прізвище та ініціали)

Завдання прийняв до виконання \_\_\_\_\_

Дрозд В.В.

Р \_\_\_\_\_  
(підпис)

(прізвище та ініціали студента)

е

к

т

о

р

а

Н

У

Б

і

П

У

## РЕФЕРАТ

Пояснювальна записка: 63 сторінок, 18 рисунки, 8 лістингів, 20 джерел.

БЕЗПЛОТНИЙ ЛІТАЛЬНИЙ АПАРАТ, СИСТЕМА АВТОМАТИЗОВАНОГО ПОЛИВУ, ІОТ, GPS-НАВІГАЦІЯ, АЛГОРИТМИ ПЛАНУВАННЯ МАРШРУТІВ, ЕВАПОТРАНСPIРАЦІЯ, ТОЧНЕ ЗЕМЛЕРОБСТВО, ЕКОНОМІЯ ВОДНИХ РЕСУРСІВ.

Об'єкт дослідження – процес автоматизованого точкового поливу сільськогосподарських угідь із використанням безпілотних літальних апаратів.

Предмет дослідження – апаратно-програмна платформа, яка забезпечує збір даних із сенсорів, планування маршрутів дронів і управління скиданням води.

Мета роботи – створити надійну, масштабовану та економічно виправдану систему, що виконує автоматизований полив із урахуванням стану ґрунту, потреб рослин і метеоумов.

У першому розділі обґрунтовано актуальність і сформульовано завдання дослідження. Другий розділ присвячений аналізу сучасних технологій точного землеробства та прикладних рішень із використання дронів у сільському господарстві: порівняно різні типи мультиспектральних і теплових сенсорів, протоколи зв'язку LoRa, 4G, Wi-Fi і GPS-навігацію з RTK-корекцією. У третьому розділі розглянуто теоретичні основи побудови систем поливу: моделі водного балансу, розрахунок евапотранспірації (метод FAO Penman-Monteith), алгоритми оптимізації маршрутів і адаптивного керування скиданням рідини. Четвертий розділ описує практичну реалізацію: вибір дрона та сенсорного обладнання, розробку мікропрограми для поливу на Arduino/PX4, створення бекенду на Python/Flask для планування місій і веб-інтерфейсу користувача, моделювання в середовищі Gazebo та польові випробування. У п'ятому розділі наведено

результати тестування системи в реальних умовах: точність скидання, економія води до 35 % порівняно з крапельним поливом, аналіз продуктивності та екологічний вплив.

Explanatory Note: 63 pages, 18 figures, 8 listings, 20 sources.

DRONE, AUTOMATED IRRIGATION SYSTEM, IOT, GPS NAVIGATION,  
ROUTE PLANNING ALGORITHMS, EVAPOTRANSPIRATION, PRECISION  
AGRICULTURE, WATER RESOURCE CONSERVATION

The object of this research is the process of automated spot irrigation of agricultural fields using unmanned aerial vehicles.

The subject of this research is the hardware and software platform that ensures data collection from sensors, route planning for drones, and control of water release.

The aim of this work is to create a reliable, scalable, and economically justified system that performs automated irrigation, taking into account soil conditions, plant needs, and meteorological factors.

The first section substantiates the relevance of the topic and formulates the research tasks.

The second section is devoted to the analysis of modern precision agriculture technologies and applied solutions for drone-based irrigation in agriculture: it compares various types of multispectral and thermal sensors, communication protocols such as LoRa, 4G, and Wi-Fi, and GPS navigation with RTK correction.

The third section examines the theoretical foundations of irrigation systems: water balance models, evapotranspiration calculation using the FAO Penman-Monteith method, route optimization algorithms, and adaptive control of liquid release.

The fourth section describes the practical implementation: selection of the drone and sensor equipment, development of irrigation firmware for Arduino/PX4, creation of a Python/Flask backend for mission planning and a web user interface, simulation in the Gazebo environment, and field testing.

The fifth section presents the results of testing the system under real conditions: drop-release accuracy, water savings of up to 35 % compared to drip irrigation, performance evaluation, and environmental impact assessment.

# ЗМІСТ

РЕФЕРАТ3

ABSTRACT5

ЗМІСТ6

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ8

РОЗДІЛ 110

Вступ10

- 1.1. Актуальність теми10
- 1.2. Мета та завдання дослідження11
- 1.3. Об'єкт і предмет дослідження12
- 1.4. Методи дослідження13
- 1.5. Структура роботи13

РОЗДІЛ 215

Аналіз предметної області та постановка задачі15

- 2.1. Огляд сучасних систем автоматизованого поливу15
- 2.2. Використання дронів у сільському господарстві16
- 2.3. Аналіз типів дронів, які можуть бути використані для поливу16
- 2.4. Мережеві технології для керування дроном (LoRa, Wi-Fi, GSM, GPS)21
- 2.5. Визначення функціональних можливостей та вимог до системи25

РОЗДІЛ 328

Теоретична частина дипломної роботи: Розробка комп'ютерної системи поливу з використанням дронів28

- 3.1. Основи автоматизованих систем поливу28
- 3.2. Типи сенсорів для моніторингу ґрунту та навколишнього середовища29
- 3.3. Принципи роботи дронів для сільськогосподарських задач30
- 3.4. Методи точного землеробства та оптимізації поливу31
- 3.5. Алгоритми обробки даних та прийняття рішень щодо поливу32

РОЗДІЛ 437

Практична реалізація та апробація37

4.1. Вибір апаратної частини (дрон, сенсори вологості, камера, резервуар для води)37

- 4.1.1. Рама: DJI Matrice 600 Pro37
- 4.1.2. Контролер польоту: Pixhawk 6C39
- 4.1.3. Мотори: T-Motor U8 II KV10040
- 4.1.4. Пропелери: T-Motor 28x9.2 Carbon Fiber Propellers42
- 4.1.5. Батарея: Tattu 22.2V 6S 22000mAh LiPo43
- 4.1.6. Система розпилення: DJI Agras T30 Spraying System45
7. GPS-модуль: u-blox NEO-8M з підтримкою RTK46

4.2. Розробка програмного забезпечення для керування дроном (автономний та дистанційний режим)47

4.3. Розробка програмного забезпечення для керування дроном (автономний та дистанційний режим)58

4.4. Реалізація алгоритмів польоту61

4.5. Тестування системи у реальних умовах63

РОЗДІЛ 569

Висновки69

5.1. Підсумки реалізації проєкту69

5.2. Відповідність результатів поставленим цілям70

5.3. Можливості вдосконалення та перспективи розвитку системи71

ВИСНОВКИ73

СПИСОК ЛІТЕРАТУРНИХ ДЖЕРЕЛ74

ДОДАТОК А78

ДОДАТОК Б83

## ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

БПЛА	безпілотний літальний апарат
АСП	автоматизована система поливу
IoT	Internet of Things
GPS	Global Positioning System
RTK	Real-Time Kinematic
FPV	First Person View
LoRa	Long Range
Wi-Fi	Wireless Fidelity
ArduCopter	Ardu Pilot Copter
MAVLink	Micro Air Vehicle Link
MAVFTP	MAVLink File Transfer Protocol
ADC	Analog-to-Digital Converter
PWM	Pulse-Width Modulation
UI	User Interface
C#	мова програмування C Sharp
GMap.NET	бібліотека .NET для відображення карт
Lua	мова програмування Lua
API	Application Programming Interface
latE7	широта, помножена на $10^7$
lngE7	довгота, помножена на $10^7$

alt	altitude (висота)
vol	volume (об'єм)
Hz	герц (частота оновлення)
FTP	File Transfer Protocol
ADC-канал	канал аналого-цифрового перетворювача
PWM-сигнал	сигнал широтно-імпульсної модуляції
API ArduPilot	інтерфейс програмування ArduPilot

# РОЗДІЛ 1

## Вступ

### 1.1. Актуальність теми

Сільське господарство є однією з ключових галузей економіки України, яка забезпечує продовольчу безпеку та значну частку експорту. Проте сучасні виклики, такі як зміна клімату, дефіцит водних ресурсів, зростання населення та потреба у підвищенні ефективності виробництва, вимагають впровадження інноваційних підходів до аграрних технологій. Традиційні методи зрошення, такі як дощування чи крапельний полив, часто є ресурсоємними, потребують значних витрат енергії та води, а також не завжди враховують індивідуальні особливості окремих ділянок поля. Це призводить до нерівномірного розподілу вологи, зниження врожайності та надмірного використання природних ресурсів.

Розвиток технологій точного землеробства, зокрема використання безпілотних літальних апаратів, відкриває нові перспективи для вирішення цих проблем. Дрони дозволяють проводити точний моніторинг стану ґрунту та рослин, а також виконувати локалізований полив, що сприяє економії ресурсів і підвищенню ефективності сільськогосподарського виробництва. Ідея створення комп'ютерної системи поливу з використанням дронів виникла в контексті сучасних реалій України, де триває війна станом на 2025 рік. Унаслідок воєнних дій значна кількість дронів, які використовувалися для військових потреб, може бути перепрофільована для мирних цілей після завершення конфлікту. Аграрна промисловість, яка зазнала значних втрат через війну, зокрема руйнування інфраструктури та зменшення доступу до ресурсів, отримає можливість швидкого відновлення завдяки впровадженню таких технологій. Адаптація дронів для автоматизованого поливу сприятиме не лише підвищенню продуктивності сільського господарства, але й економічному

відродженню країни шляхом використання наявних технологічних ресурсів.

Крім того, розробка такої системи відповідає глобальним трендам сталого розвитку, які передбачають раціональне використання природних ресурсів, зниження вуглецевого сліду та впровадження цифрових технологій у сільське господарство. Впровадження комп'ютеризованої системи поливу на основі дронів може стати важливим кроком у модернізації аграрного сектору України, що є особливо актуальним у післявоєнний період, коли країна потребуватиме швидких і ефективних рішень для відновлення економіки та забезпечення продовольчої безпеки.

## **1.2. Мета та завдання дослідження**

**Мета дослідження:** розробити комп'ютерну систему поливу на основі безпілотних літальних апаратів, яка забезпечує автоматизоване, точне та ефективне зрошення сільськогосподарських угідь з урахуванням потреб рослин, стану ґрунту та кліматичних умов, сприяючи економії ресурсів і підвищенню врожайності.

### **Завдання дослідження:**

провести аналіз сучасних технологій точного землеробства, включаючи системи моніторингу ґрунту, сенсори вологості, камери та програмне забезпечення для обробки даних, а також оцінити можливості їх інтеграції з дронами.

ослідити технічні характеристики дронів, які можуть бути використані для поливу, включаючи вантажопідйомність, автономність, системи навігації та управління.

розробити архітектуру комп'ютерної системи, яка включає апаратне забезпечення (дрони, датчики, резервуари для води) та програмне забезпечення для керування процесом поливу.

творити програмне забезпечення для автоматизації планування маршрутів дронів, обробки даних із сенсорів і прийняття рішень щодо оптимального розподілу води.

провести моделювання роботи системи в контрольованих умовах для оцінки її функціональності та ефективності.

виконати експериментальне тестування системи, щоб визначити її точність, економічність і потенційні недоліки.

проаналізувати перспективи впровадження розробленої системи в реальних сільськогосподарських умовах, зокрема в контексті післявоєнного відновлення аграрного сектору України, враховуючи доступність дронів і економічні вигоди від їх використання.

цінити екологічний вплив системи, зокрема її потенціал для зменшення водоспоживання та підвищення сталості сільськогосподарського виробництва.

### **1.3. Об'єкт і предмет дослідження**

**Об'єкт дослідження:** процес автоматизованого поливу сільськогосподарських угідь із застосуванням безпілотних літальних апаратів, який охоплює моніторинг стану посівів, планування маршрутів дронів і доставку води до цільових ділянок. Цей процес є складовою точного землеробства, яке спрямоване на оптимізацію сільськогосподарських операцій за допомогою сучасних технологій.

**Предмет дослідження:** комп'ютерна система управління дронами для реалізації точного поливу, що включає апаратне забезпечення (дрони, сенсори, системи навігації) та програмне забезпечення (алгоритми планування маршрутів, обробки даних і керування поливом). Система розробляється з урахуванням специфіки аграрного сектору України та можливості адаптації військових дронів для мирних цілей.

## 1.4. Методи дослідження

Для досягнення мети дослідження використано комплексний підхід, що поєднує теоретичні, аналітичні та практичні методи:

- **Аналітичний метод** – застосовано для вивчення наукової літератури, технічної документації та сучасних технологій у сфері точного землеробства, дронів і систем автоматизованого поливу. Це дозволило визначити ключові тенденції та виклики в цій галузі.
- **Системний аналіз** – використано для розробки архітектури комп'ютерної системи, що включає взаємодію апаратних і програмних компонентів, а також інтеграцію з зовнішніми джерелами даних, такими як метеорологічні сервіси.
- **Метод програмування** – застосовано для створення програмного забезпечення, яке забезпечує автоматизацію процесів планування маршрутів дронів, обробки даних із сенсорів і керування поливом. Програмування виконувалося з використанням сучасних мов, таких як Python, для обробки даних і JavaScript для створення інтерфейсу користувача.
- **Моделювання** – використано для імітації роботи системи в контрольованих умовах, що дозволило оцінити її поведінку в різних сценаріях (наприклад, при зміні погодних умов або типу ґрунту).
- **Експериментальний метод** – застосовано для тестування розробленої системи в лабораторних і польових умовах, щоб перевірити її точність, надійність і ефективність у реальних задачах поливу.
- **Порівняльний аналіз** – використано для оцінки переваг і недоліків розробленої системи порівняно з традиційними методами поливу та іншими автоматизованими системами.

## 1.5. Структура роботи

Дипломна робота складається з п'яти основних розділів, кожен із яких послідовно розкриває етапи дослідження та реалізації комп'ютерної системи поливу з використанням дронів.

Робота починається з вступу який містить обґрунтування актуальності теми, формулювання мети та завдань дослідження, визначення об'єкта і предмета дослідження, а також опис методів, що використовуються в роботі. У цьому розділі також представлено структуру роботи, яка дає загальне уявлення про організацію дослідження. Далі йде аналіз предметної області та постановка задачі присвячений вивченню сучасних систем автоматизованого поливу та застосуванню дронів у сільському господарстві. У ньому розглядаються типи дронів, придатних для поливу, аналізуються мережеві технології (LoRa, Wi-Fi, GSM, GPS) для керування дроном, а також визначаються функціональні вимоги до розроблюваної системи. Теоретична частина описує основи автоматизованих систем поливу, типи сенсорів для моніторингу ґрунту та навколишнього середовища (вологість, температура, рН), принципи роботи дронів у сільськогосподарських задачах, методи точного землеробства та алгоритми обробки даних для оптимізації поливу. Практична реалізація та апробація охоплює вибір апаратного забезпечення (дрон, сенсори, камера, резервуар для води), розробку програмного забезпечення для автономного та дистанційного керування дроном, інтеграцію системи з сенсорами та серверною частиною, а також створення алгоритмів польоту й обробки даних. У цьому розділі також представлено результати тестування системи в реальних умовах і оцінку її ефективності. І в останньому розділі це висновки, які підсумовують результати роботи, оцінює відповідність отриманих результатів поставленим цілям і окреслює перспективи вдосконалення системи та її подальшого розвитку. Робота завершується списком використаних джерел та додатками.

## РОЗДІЛ 2

### Аналіз предметної області та постановка задачі

#### 2.1. Огляд сучасних систем автоматизованого поливу

Сучасне сільське господарство активно переходить до автоматизації процесів, включаючи системи поливу, що дозволяють оптимізувати використання водних ресурсів, підвищувати врожайність та зменшувати витрати. Автоматизовані системи поливу (далі АСП) включають крапельний полив, спринклерні системи, а також системи з датчиками вологості ґрунту та кліматичними сенсорами. Крапельний полив забезпечує точкове зрошення кореневої зони рослин, що знижує втрати води до 30-40% порівняно з традиційними методами. Спринклерні системи, у свою чергу, ефективні для великих площ, але можуть втрачати воду через випаровування.

Сучасні АСП інтегруються з інформаційними технологіями, що дозволяють дистанційно керувати поливом через мобільні додатки або комп'ютерні системи. Наприклад, системи точного землеробства використовують дані із супутників, датчиків та IoT-пристроїв для аналізу стану ґрунту, прогнозування погоди та автоматичного регулювання поливу. Однак ці системи мають обмеження: висока вартість встановлення, залежність від стабільного електропостачання та складність адаптації до нерівномірних або віддалених ділянок.

В умовах України, де війна спричинила руйнування інфраструктури та нестачу робочої сили, традиційні системи поливу стають менш ефективними. Після закінчення війни значна кількість дронів, які зараз використовуються у військових цілях, може бути перепрофільована для аграрного сектору. Ця ідея стала основою для розробки комп'ютерної системи поливу з використанням дронів, що може вирішити проблеми доступності та точності зрошення в

умовах післявоєнної відбудови.

## **2.2. Використання дронів у сільському господарстві**

Дрони (далі БПЛА) активно застосовуються в сільському господарстві для моніторингу, обприскування, посіву та картографування полів. Згідно з джерелами, дрони дозволяють підвищити продуктивність праці, зменшити витрати на хімічні засоби та оптимізувати використання ресурсів. Наприклад, в Україні дрони використовуються для обприскування полів гербіцидами, що дозволяє обробляти осередки бур'янів з високою точністю, зменшуючи витрати пестицидів до 40%.

У контексті війни в Україні дрони стали ключовим елементом технологічної революції, зокрема у військовій сфері. Після закінчення війни значна кількість БПЛА, включаючи FPV-дрони та важкі аграрні моделі, може бути адаптована для мирних цілей, зокрема для поливу. Це обумовлено високою вантажопідйомністю сучасних дронів, їхньою маневреністю та можливістю інтеграції з GPS-навігацією для точного позиціонування. Використання дронів для поливу може бути особливо корисним у регіонах з пошкодженою інфраструктурою, де традиційні системи поливу важко відновити.

Дрони також дозволяють обробляти важкодоступні ділянки, такі як круті схили або заболочені території, де наземна техніка неефективна. Наприклад, у Каліфорнії дрони допомогли виявити пошкодження крапельних систем поливу, спричинені дикими тваринами, що дозволило оперативно усунути проблему та зберегти врожай. В Україні такі технології можуть стати вирішальними для відновлення аграрного сектору після війни.

## **2.3. Аналіз типів дронів, які можуть бути використані для поливу**

Для реалізації комп'ютерної системи поливу з використанням дронів

необхідно ретельно проаналізувати типи безпілотних літальних апаратів (далі БПЛА), їхні технічні характеристики, можливості адаптації до задач поливу, а також економічні та експлуатаційні аспекти. Вибір типу дрона залежить від розміру сільськогосподарської ділянки, рельєфу місцевості, доступного бюджету та вимог до точності й ефективності поливу. У цьому розділі розглянуто основні типи дронів, які можуть бути застосовані для системи поливу: квадрокоптери, аграрні дрони з високою вантажопідйомністю, FPV-дрони та дрони-міномети. Також проаналізовано їхні переваги, недоліки, сценарії використання та доцільність застосування в умовах післявоєнної України, де інфраструктура та ресурси можуть бути обмеженими.

### Квадрокоптери

Квадрокоптери є найпоширенішим типом дронів у сільськогосподарському секторі завдяки їхній універсальності, маневреності та відносно низькій вартості. Такі моделі, як DJI Mavic 3 або Autel Evo Lite+, мають дальність польоту від 12 до 18 км, час автономної роботи до 40 хвилин і можуть оснащуватися невеликими резервуарами для рідини до 2–3 кг. Їхня здатність зависати на місці дозволяє точно спрямовувати потік води на проблемні ділянки поля, що робить їх ідеальними для точкового поливу невеликих ділянок до 5 га. Квадрокоптери також легко інтегруються з GPS-модулями, що забезпечує автоматичне прокладання маршрутів і виконання запрограмованих задач.

Основною перевагою квадрокоптерів є їхня доступність: ціна моделей середнього класу становить від 1000\$ до 5000\$, що робить їх економічно виправданими для малих і середніх фермерських господарств. Вони також прості в управлінні, що знижує вимоги до підготовки операторів. Однак їхня вантажопідйомність обмежена до 2–3 кг, що робить їх менш ефективними для обробки великих площ. Крім того, час польоту обмежений ємністю акумулятора, а часті заміни батареї або підзарядка можуть уповільнити процес

поливу. У післявоєнних умовах України квадрокоптери є перспективними для локальних господарств, але їхнє використання на великих полях потребує значної кількості одиниць техніки, що може ускладнити координацію.

#### Аграрні дрони з високою вантажопідйомністю

Аграрні дрони, такі як XAG P100, DJI Agras T40 або Yamaha RMAX, спеціально розроблені для сільськогосподарських задач, включаючи обприскування та полив. Ці дрони здатні перевозити резервуари з рідиною об'ємом до 20–40 літрів і обробляти від 15 до 20 гектарів за годину, що робить їх ідеальними для великих сільськогосподарських угідь. Наприклад, XAG P100 оснащений системою розпилення з регульованою інтенсивністю, яка може бути адаптована для поливу, забезпечуючи рівномірне зрошення або точкове нанесення води. Такі дрони підтримують автономний політ із використанням GPS, RTK (Real-Time Kinematic) і сенсорів для уникнення перешкод, що підвищує точність і безпеку операцій.

Основною перевагою аграрних дронів є їхня висока продуктивність і здатність обробляти великі площі за короткий час. Вони також можуть інтегруватися з IoT-системами для збору даних із сенсорів вологості ґрунту, що дозволяє оптимізувати витрати води. Однак висока вартість таких дронів приблизно 20 000\$ є значним бар'єром для малих господарств, особливо в умовах післявоєнної України, де фінансові ресурси обмежені. Крім того, ці дрони потребують регулярного технічного обслуговування, спеціалізованого програмного забезпечення та кваліфікованих операторів, що підвищує експлуатаційні витрати. У регіонах із пошкодженою інфраструктурою зарядка акумуляторів або забезпечення запасними частинами може бути проблематичною.

#### FPV-дрони

FPV-дрони відомі своєю високою маневреністю та можливістю передачі

відео в реальному часі. Вони здатні перевозити невеликі вантажі до 2–3 кг і можуть бути адаптовані для транспортування компактних резервуарів із водою. Завдяки камерам високої роздільної здатності та низькій затримці передачі відео, FPV-дрони дозволяють оператору точно контролювати процес поливу, що особливо корисно на ділянках зі складним рельєфом, таких як горбисті або лісисті території. Наприклад, FPV-дрон може бути використаний для поливу окремих дерев у саду або ділянок із нерівномірною вологістю.

Перевагою FPV-дронів є їхня гнучкість і можливість роботи в умовах, де автоматизовані системи можуть бути менш ефективними. Вони також відносно недорогі, від 500\$ до 2000\$, і можуть бути зібрані з доступних компонентів, що є важливим у післявоєнних умовах. Однак їхня вантажопідйомність і час польоту 10–20 хвилин, обмежують їхнє використання для великих площ. Крім того, керування FPV-дронами вимагає високої кваліфікації оператора, що може бути викликом для фермерів без спеціальної підготовки. У контексті України FPV-дрони можуть бути корисними для малих господарств або для точкового поливу в регіонах із нестабільною інфраструктурою.

### Дрони-міномети

Дрони-міномети здатні перевозити вантажі до 20 кг на відстань до 20–30 км. Вони спочатку розроблялися для військових цілей, але можуть бути адаптовані для транспортування резервуарів із водою для поливу великих ділянок. Такі дрони оснащені потужними двигунами та системами стабілізації, що дозволяє їм працювати в складних погодних умовах, таких як сильний вітер або дощ. Їхня дальність польоту та вантажопідйомність роблять їх перспективними для обробки великих сільськогосподарських площ, особливо в регіонах із низькою щільністю інфраструктури.

Перевагою дронів-мінометів є їхня здатність виконувати завдання в автономному режимі на великих відстанях, що зменшує потребу в операторі на

полі. Вони також можуть бути оснащені модулями для скидання рідини, що дозволяє швидко поливати цільові ділянки. Однак їхня висока вартість від 20 000\$ і складність технічного обслуговування роблять їх менш доступними для малих і середніх господарств. У післявоєнних умовах України такі дрони можуть бути корисними для великих агропідприємств, але потребують стабільного енергопостачання та доступу до запасних частин, що може бути проблематичним.

### Порівняння та вибір оптимального типу дрона

Для вибору оптимального типу дрона необхідно враховувати розмір ділянки, бюджет, рельєф місцевості та доступність інфраструктури. Квадрокоптери є економічно виправданими для малих господарств до 5 га, завдяки низькій вартості та простоті використання. Аграрні дрони з високою вантажопідйомністю найкраще підходять для великих полів понад 10 га, але їхнє впровадження обмежене високою вартістю. FPV-дрони є оптимальними для складного рельєфу або точкового поливу, тоді як дрони-міномети можуть використовуватися для великих площ у віддалених регіонах.

У післявоєнних умовах України, де ресурси та інфраструктура обмежені, квадрокоптери та FPV-дрони є найбільш доступними рішеннями для малих і середніх господарств. Для великих агропідприємств доцільно інвестувати в аграрні дрони, які забезпечують високу продуктивність. Комбінація різних типів дронів у єдиній системі може підвищити ефективність: наприклад, квадрокоптери можуть виконувати точковий полив, а аграрні дрони — обробляти великі площі.

### Сучасні тенденції та перспективи

Сучасні розробки в галузі дронів відкривають нові можливості для систем поливу. Наприклад, використання штучного інтелекту дозволяє дронам аналізувати дані з сенсорів у реальному часі та оптимізувати маршрути поливу.

Модульні конструкції дронів, дозволяють швидко адаптувати їх до різних задач (обприскування, полив, моніторинг). У майбутньому розвиток гібридних дронів із комбінацією електричних і паливних двигунів може збільшити час польоту та вантажопідйомність, що зробить їх більш ефективними для поливу великих площ.

#### **2.4. Мережеві технології для керування дроном (LoRa, Wi-Fi, GSM, GPS)**

Ефективне керування дронами в комп'ютерній системі поливу вимагає використання надійних, енергоефективних і адаптивних мережевих технологій, які забезпечують зв'язок між дронами, сенсорами, базовими станціями та оператором. У контексті сільськогосподарських задач, таких як точкове зрошення великих площ, необхідно враховувати особливості кожної технології, її придатність до умов експлуатації, а також можливість інтеграції в єдину систему. У цьому розділі розглянуто основні мережеві технології (LoRa, Wi-Fi, GSM, GPS), їхні переваги, недоліки, сценарії використання та потенціал комбінації для створення ефективної системи поливу в умовах післявоєнної України, де інфраструктура зв'язку може бути обмеженою або пошкодженою.

##### **LoRa (Long Range)**

Технологія Long Range (далі LoRa) є однією з ключових для систем IoT (далі Internet of Things) завдяки своїй здатності забезпечувати зв'язок на великих відстанях (до 15 км у сільській місцевості та до 2–5 км у міських умовах) при низькому енергоспоживанні. LoRa працює в неліцензованих частотних діапазонах, що знижує витрати на її впровадження. У системі поливу LoRa ідеально підходить для передачі даних із сенсорів вологості ґрунту, температури чи рівня опадів до дрона або базової станції. Наприклад, сенсори,

розміщені на полі, можуть передавати дані про стан ґрунту в реальному часі, дозволяючи дрону адаптувати маршрути поливу для максимальної ефективності.

LoRa підтримує концепцію "рою", коли кілька дронів координуються через єдину мережу для обробки великих сільськогосподарських ділянок. Наприклад, один дрон може отримувати дані від сенсорів через LoRa, передавати їх на базову станцію, яка розподіляє завдання між іншими дронами. Це зменшує час обробки великих полів і оптимізує витрати ресурсів. Проте LoRa має обмеження: низька швидкість передачі даних робить її непридатною для передачі великих обсягів інформації, таких як потокове відео чи складні телеметричні дані в реальному часі. Крім того, у районах із високим рівнем радіоперешкод (наприклад, у міських зонах або поблизу військових об'єктів) ефективність LoRa може знижуватися.

## Wi-Fi

Wi-Fi забезпечує високу швидкість передачі даних до 100 Мбіт/с і вище залежно від стандарту, наприклад, 802.11ac або 802.11ax, що робить його ідеальним для трансляції відео з камер FPV (далі First Person View) на дронах. Це дозволяє оператору в реальному часі спостерігати за станом поля, виявляти проблемні зони та коригувати маршрут дрона. Wi-Fi також підходить для передачі складних команд управління, таких як зміна параметрів поливу або активація спеціальних режимів роботи.

Однак Wi-Fi має суттєві обмеження, зокрема малий радіус дії до 300 м на відкритій місцевості без ретрансляторів, що робить його менш ефективним для великих сільськогосподарських угідь. Для подолання цього недоліку необхідно встановлювати мережу ретрансляторів або використовувати дрони як мобільні точки доступу, що ускладнює систему та підвищує її вартість. Крім того, Wi-Fi споживає значно більше енергії порівняно з LoRa, що може скоротити час

автономного польоту дрона, особливо якщо використовуються енергоємні модулі, такі як камери високої роздільної здатності. У післявоєнних умовах, де електропостачання може бути нестабільним, це створює додаткові виклики.

## GSM

Мобільні мережі GSM дозволяють керувати дронами на значних відстанях без необхідності прямого радіозв'язку. Це особливо корисно для віддалених регіонів, де інфраструктура Wi-Fi або LoRa недоступна. Наприклад, дрон, оснащений модулем 4G, може отримувати команди від оператора через хмарний сервер, що забезпечує гнучкість у керуванні системою поливу. У системах із підтримкою 5G можлива передача даних із мінімальною затримкою до 1 мс, що дозволяє реалізувати складні сценарії, такі як синхронізація кількох дронів у реальному часі.

Проте використання GSM має свої недоліки. По-перше, залежність від покриття мобільної мережі може бути проблемою в сільських або післявоєнних регіонах України, де інфраструктура зв'язку часто пошкоджена або відсутня. По-друге, висока вартість передачі даних, особливо для потокового відео чи великих обсягів телеметрії, робить GSM менш економічним рішенням. Крім того, безпека зв'язку в GSM-мережах може бути вразливою до перехоплення, що вимагає додаткових заходів шифрування.

## GPS

GPS є основою для навігації дронів у системах точного землеробства. Сучасні дрони, такі як DJI Air 2S або Mavic 3, використовують GPS-модулі для визначення координат із точністю до 0,3–1 м, що дозволяє створювати точні маршрути польоту та забезпечувати адресне зрошення проблемних ділянок поля. GPS також підтримує функції автономного польоту, такі як повернення до точки старту або автоматичне облітання перешкод. У поєднанні з даними від сенсорів вологості, температури чи рівня поживних речовин у ґрунті, GPS

забезпечує високу точність поливу, що знижує витрати води та підвищує ефективність.

Однак GPS має обмеження, пов'язані з можливими перешкодами сигналу, наприклад, у густо зарослих лісами районах або поблизу високих споруд. У післявоєнних умовах України додатковим викликом може бути вплив радіоелектронної боротьби (далі РЕБ), яка використовується для придушення сигналів GPS. Для підвищення надійності застосовуються технології диференціальної корекції (DGPS) або комбінація GPS із системами ГЛОНАСС, Galileo чи BeiDou, що забезпечують стабільніше позиціонування.

### Інтеграція та комбінація технологій

Для створення ефективної системи поливу доцільно комбінувати кілька мережевих технологій, щоб компенсувати їхні недоліки. Наприклад, LoRa може використовуватися для передачі даних із сенсорів до базової станції або дрона, GPS — для точної навігації, а Wi-Fi або GSM — для віддаленого керування та моніторингу. Така гібридна архітектура дозволяє оптимізувати енергоспоживання, покриття та швидкість передачі даних. Наприклад, дрон може отримувати координати проблемних ділянок через LoRa, прокладати маршрут за допомогою GPS, а оператор може спостерігати за процесом через Wi-Fi або 4G-з'єднання.

У післявоєнній Україні, де інфраструктура зв'язку може бути частково зруйнованою, пріоритет слід віддавати технологіям LoRa та GPS через їхню автономність і низьке енергоспоживання. LoRa не залежить від комерційних мереж, що робить її стійкою до перебоїв у зв'язку, а GPS забезпечує надійну навігацію навіть у віддалених регіонах. Для підвищення безпеки та надійності системи рекомендується використовувати шифрування даних (наприклад, AES-256 для LoRa) і резервні канали зв'язку, такі як супутниковий зв'язок (Iridium або Starlink), для критичних операцій.

Сучасні розробки в галузі мережевих технологій відкривають нові можливості для систем поливу з використанням дронів. Наприклад, поява стандарту 5G New Radio (далі NR) у низькочастотних діапазонах sub-6 GHz дозволяє поєднувати високу швидкість передачі даних із великим радіусом дії, що може замінити Wi-Fi у великих господарствах. Крім того, технології Low-Power Wide-Area Network (далі LPWAN), такі як NB-IoT або Sigfox, є альтернативою LoRa в регіонах із розвиненою інфраструктурою. У майбутньому інтеграція штучного інтелекту із мережевими технологіями може дозволити дронам самостійно аналізувати дані сенсорів і приймати рішення щодо поливу без участі оператора.

## **2.5. Визначення функціональних можливостей та вимог до системи**

Система поливу з використанням дронів забезпечує автоматизоване виконання завдань із доставки та скиду води на визначені точки за допомогою спеціалізованого програмного забезпечення та бортового скрипта. Вона дозволяє створювати, редагувати та передавати маршрути поливу, контролювати рівень води в баку, враховувати фізичні параметри для точного скиду та надавати зворотний зв'язок користувачу в реальному часі.

### **Функціональні можливості системи:**

- Створення необмеженої кількості точок поливу з координатами (широта, довгота, висота) та опціональним об'ємом води через інтерактивний інтерфейс.
- Редагування та видалення точок поливу з синхронізацією між картою та таблицею координат.
- Автоматичне формування текстового файлу з координатами точок і передача його на борт дрона після натискання відповідної кнопки.
- Читання файлу з точками на борту дрона та послідовна активація цілей для скиду води.

- Обчислення горизонтального зміщення краплі з урахуванням швидкості дрона, висоти над поверхнею та фізичних параметрів (гравітація, щільність води, діаметр краплі, коефіцієнт опору).
- Моніторинг рівня води в баку через аналоговий датчик із калібруванням за напругою та пропуск скиду при недостатньому рівні.
- Керування механізмом скиду через PWM-сигнал у момент досягнення цільової точки.
- Надсилання текстових повідомлень до наземної станції про стан місії, відстань до цілі, факт скиду, залишковий об'єм води та помилки.
- Обробка помилок (наприклад, відсутність файлу або проблеми з передачею) із виведенням повідомлень.
- Скидання стану системи при роззброєнні дрона для забезпечення коректної роботи.

#### **Вимоги до системи:**

- Дрон із прошивкою ArduCopter 4.1 або вище, GPS-модулем і MAVLink-інтерфейсом для зв'язку через MAVFTP.
- Аналоговий датчик рівня води, підключений до ADC-каналу, із діапазоном напруги 0.6–3.0 В для бака ємністю 200 л.
- Механізм скиду, керований через PWM-сигнал (канал 5).
- Наземна станція з Mission Planner 1.3.80 або вище для роботи плагіна.
- Плагін на C# із підтримкою GMap.NET для відображення маркерів і таблиці координат.
- Бортовий Lua-скрипт, сумісний із API ArduPilot, для обробки файлу з координатами у форматі latE7:lngE7:alt[:vol].

- Точність позиціонування до  $10^{-7}$  градуса та частота оновлення скрипта 10 Гц.
- Інтерактивний інтерфейс із картою, таблицею координат, кнопками для додавання точок і передачі файлу, а також діалоговими вікнами для ручного введення даних.
- Обробка помилок доступу до файлу, FTP-з'єднання та недостатнього рівня води з відповідними повідомленнями.
- Обчислення зміщення краплі з використанням фізичних параметрів.

## РОЗДІЛ 3

### Теоретична частина дипломної роботи: Розробка комп'ютерної системи поливу з використанням дронів

#### 3.1. Основи автоматизованих систем поливу

Автоматизовані системи поливу призначені для оптимізації подачі води до сільськогосподарських культур з мінімальним втручанням людини. Вони дозволяють точно контролювати час і обсяг поливу, що сприяє економії водних ресурсів, підвищенню врожайності та зменшенню витрат праці. Такі системи широко застосовуються в різних методах зрошення, таких як крапельний, дощувальний або поверхневий полив, і можуть адаптуватися до потреб конкретних культур чи умов поля.

Основними компонентами автоматизованих систем поливу є контролери, датчики, клапани, насоси та системи зв'язку. Контролери керують розкладом поливу, активуючи систему на основі запрограмованих налаштувань або даних, отриманих від датчиків. Клапани та насоси забезпечують фізичну подачу води через мережу трубопроводів, тоді як системи зв'язку дозволяють віддалено моніторити та керувати процесом через інтернет або мобільні мережі. Наприклад, фермери можуть налаштувати систему для автоматичного поливу певних ділянок поля у визначений час або за умови зниження вологості ґрунту до певного рівня.

Системи бувають двох основних типів: на основі часу та на основі обсягу. У системах на основі часу фермер задає графік поливу, наприклад, щоденний полив о 6 ранку протягом 30 хвилин. Однак такі системи можуть бути менш ефективними, оскільки не враховують погодні умови чи фактичні потреби рослин. Системи на основі обсягу дозволяють задавати точну кількість води, що подається, наприклад, 10 літрів на квадратний метр, що забезпечує кращий

контроль і економію ресурсів. Переваги автоматизованих систем включають зменшення водних відходів, підвищення врожайності за рахунок оптимального зволоження, економію часу фермерів та екологічну стійкість завдяки зниженню ерозії ґрунту та вимивання поживних речовин.

### **3.2. Типи сенсорів для моніторингу ґрунту та навколишнього середовища**

Датчики є ключовим елементом автоматизованих систем поливу, оскільки вони надають дані в реальному часі про стан ґрунту та навколишнього середовища, необхідні для прийняття рішень щодо поливу. Основними типами датчиків, які використовуються в таких системах, є датчики вологості ґрунту, температури та рН, а також додаткові датчики для моніторингу вологості повітря та погодних умов.

**Датчики вологості ґрунту** вимірюють вміст води в ґрунті, що дозволяє визначити, коли і скільки води потрібно подати. Наприклад, ємнісні датчики оцінюють вологість через діелектричні властивості ґрунту, тензометри вимірюють напругу води, а датчики часової доменної рефлектометрії використовують електромагнітні хвилі для високоточної оцінки вологості. Ці датчики допомагають уникнути надмірного або недостатнього поливу, що особливо важливо в посушливих регіонах.

**Датчики температури** моніторять температуру ґрунту та повітря, яка впливає на ріст рослин, проростання насіння та мікробну активність. Висока температура може прискорити випаровування, тоді як низька може уповільнити поглинання води рослинами. Дані про температуру допомагають коригувати графік поливу, особливо в умовах екстремальних погодних явищ.

**Датчики рН** вимірюють кислотність або лужність ґрунту, що впливає на доступність поживних речовин для рослин. Наприклад, більшість культур

найкраще ростуть при рН від 6 до 7. Якщо рН відхиляється від оптимального діапазону, система може сигналізувати про необхідність корекції, наприклад, внесення добрив.

Додатково використовуються датчики вологості повітря та метеостанції, які вимірюють вологість, швидкість вітру, опади та інші параметри. Ці дані допомагають оцінити швидкість випаровування та розрахувати потреби в поливі. Наприклад, метеостанція може надати інформацію про опади, що дозволяє відкласти полив, якщо дощ уже забезпечив достатню вологу.

### **3.3. Принципи роботи дронів для сільськогосподарських задач**

Дрони, або безпілотні літальні апарати, стали важливим інструментом у сучасному сільському господарстві завдяки їх здатності швидко та ефективно збирати дані з великих площ. У контексті системи поливу дрони відіграють ключову роль у моніторингу стану посівів і ґрунту, а також у виявленні ділянок, що потребують додаткового зрошення.

Дрони оснащуються різними типами сенсорів, такими як мультиспектральні та теплові камери, а також LiDAR. Мультиспектральні камери знімають зображення в різних діапазонах довжин хвиль, що дозволяє оцінити здоров'я рослин через індекси, такі як NDVI. Теплові камери виявляють ділянки з водним стресом, оскільки рослини з нестачею води мають вищу температуру через знижену транспірацію. LiDAR використовується для створення топографічних карт, що допомагають у плануванні поливу на полях зі складним рельєфом.

У задачах поливу дрони виконують кілька функцій. По-перше, вони створюють карти вологості ґрунту, використовуючи мікрохвильові радіометри або інші сенсори, які точно вимірюють вміст води. По-друге, дрони виявляють ділянки з водним стресом, дозволяючи фермерам зосередити полив на

проблемних зонах. По-третє, вони можуть перевіряти стан поливної інфраструктури, виявляючи витoki або несправності в трубопроводах. Переваги використання дронів включають швидке покриття великих площ, можливість роботи в важкодоступних місцях і надання даних у реальному часі, що сприяє оперативному прийняттю рішень.

### 3.4. Методи точного землеробства та оптимізації поливу

Точне землеробство (precision agriculture) спрямоване на оптимізацію сільськогосподарських процесів шляхом адаптації управління до специфічних умов окремих ділянок поля. У контексті поливу це означає подачу води в потрібній кількості, у потрібний час і в потрібному місці, що підвищує ефективність використання ресурсів і врожайність.

Основні методи точного землеробства для оптимізації поливу включають:

- **Крапельний полив:** подача води безпосередньо до кореневої зони рослин через систему трубок і крапельниць, що зменшує втрати води через випаровування чи стік.
- **Змінна норма поливу (VRI):** Використання даних із датчиків і дронів для регулювання обсягу води залежно від потреб конкретних ділянок поля, враховуючи відмінності в типі ґрунту чи стані рослин.
- **Дистанційне зондування:** Застосування зображень із дронів або супутників для оцінки стану посівів і вологості ґрунту, що дозволяє створювати карти приписів для поливу.

Ці методи підтримуються технологіями, такими як датчики вологості ґрунту, метеостанції, геоінформаційні системи (далі ГІС) та моделі евапотранспірації. Наприклад, ГІС дозволяють створювати детальні карти полів, де вказуються зони з різними потребами у воді, а моделі евапотранспірації розраховують втрати води через випаровування та

транспірацію рослин. Це допомагає визначити точну кількість води, необхідну для кожної ділянки, що особливо важливо в умовах обмежених водних ресурсів.

Переваги точного землеробства включають економію води, підвищення врожайності, зниження витрат на енергію та добрива, а також зменшення впливу на довкілля. У контексті України, де післявоєнне відновлення аграрного сектору потребуватиме інноваційних рішень, методи точного землеробства можуть відіграти ключову роль у підвищенні продуктивності.

### **3.5. Алгоритми обробки даних та прийняття рішень щодо поливу**

Ефективне управління комп'ютерною системою поливу з використанням дронів залежить від інтелектуальних алгоритмів обробки даних та прийняття рішень, які забезпечують автоматизацію, точність і адаптивність до змінних умов. Ці алгоритми інтегрують дані з різноманітних джерел, таких як сенсори вологості ґрунту, метеостанції, камери дронів і зовнішні бази даних про погодні умови, для визначення оптимального режиму поливу. У контексті післявоєнної України, де аграрний сектор потребує швидкого відновлення, а надлишок дронів, використаних у військових цілях, може бути перепрофільовано для мирних завдань, такі алгоритми стають ключовим інструментом для підвищення продуктивності сільського господарства. Вони дозволяють не лише оптимізувати використання водних ресурсів, але й мінімізувати витрати на робочу силу та енергію, що є критично важливим у умовах економічної нестабільності.

Розробка комп'ютерної системи поливу з використанням дронів виникла як відповідь на сучасні виклики, з якими зіткнулася Україна через війну, що триває. Воєнні дії призвели до широкого використання дронів у військових

операціях, що створило значний потенціал для їхньої адаптації в цивільному секторі після завершення конфлікту. Аграрна промисловість, як одна з ключових галузей економіки України, потребує інноваційних рішень для відновлення та підвищення ефективності. Дрони, оснащені сенсорами та системами поливу, можуть стати економічно вигідним рішенням для автоматизації зрошення, особливо на великих площах або в регіонах із пошкодженою інфраструктурою. Використання алгоритмів обробки даних і прийняття рішень дозволяє оптимізувати роботу дронів, забезпечуючи точковий полив лише там, де це необхідно, що зменшує витрати води та підвищує врожайність.

Для забезпечення ефективного функціонування системи поливу з використанням дронів застосовуються різні типи алгоритмів, кожен із яких має свої переваги та обмеження. Нижче наведено детальний опис основних підходів:

### **1. Алгоритми на основі порогових значень**

Ці алгоритми є найпростішими та базуються на порівнянні поточних показників із заздалегідь встановленими порогоми. Наприклад, якщо датчик вологості ґрунту фіксує значення нижче 20%, система активує полив на відповідній ділянці. Такі алгоритми легкі в реалізації та не потребують складних обчислень, що робить їх придатними для невеликих систем із обмеженими обчислювальними ресурсами. Однак їхня точність може знижуватися в умовах змінної погоди або неоднорідного ґрунту, оскільки вони не враховують складніші фактори, такі як евапотранспірація чи тип культури. У системі з дронами порогові алгоритми можуть використовуватися для швидкого прийняття рішень на основі даних із сенсорів, встановлених на дронах або в ґрунті.

### **2. Моделі водного балансу**

Моделі водного балансу враховують усі джерела надходження та втрати води в ґрунті, включаючи опади, полив, випаровування та дренаж. Ці моделі дозволяють оцінити поточний стан вологості ґрунту та спрогнозувати потребу в поливі. Наприклад, алгоритм може аналізувати дані про кількість опадів із метеостанції, втрати води через випаровування (на основі температури та вологості повітря) і поточний рівень вологості, отриманий із сенсорів дронів. Такий підхід забезпечує більш точне планування поливу порівняно з пороговими алгоритмами, але вимагає більшої кількості даних і обчислювальних ресурсів. У післявоєнній Україні, де доступ до метеостанцій може бути обмеженим через пошкоджену інфраструктуру, дрони можуть частково компенсувати цю проблему, збираючи дані про мікроклімат на полях.

### **3. Алгоритми на основі евапотранспірації (ЕТ)**

Евапотранспірація є ключовим показником, що відображає втрати води через випаровування з поверхні ґрунту та транспірацію рослин. Алгоритми, засновані на ЕТ, використовують математичні моделі, такі як метод FAO Penman-Monteith, для розрахунку референтної евапотранспірації (ЕТ<sub>0</sub>). Цей метод враховує погодні параметри (температуру, вологість, швидкість вітру, сонячну радіацію) та характеристики культури (коефіцієнт культури, К<sub>с</sub>). У системі поливу дрони можуть бути оснащені камерами та сенсорами для збору даних про стан рослин, які потім використовуються для уточнення розрахунків ЕТ. Наприклад, інфрачервоні камери дронів можуть виявляти ділянки з недостатньою вологістю за змінами температури листя. Такі алгоритми є високоточними, але потребують складних обчислень і регулярного оновлення погодних даних.

### **4. Алгоритми машинного навчання**

Алгоритми машинного навчання (ML) є перспективним напрямом для систем поливу, оскільки дозволяють прогнозувати потреби в поливі на основі

великих обсягів даних. Наприклад, моделі на основі підтримуючих векторних машин (SVM) або нейронних мереж можуть аналізувати історичні дані про вологість ґрунту, погодні умови, тип культури та продуктивність поливу для створення прогнозів. Дрони відіграють важливу роль у зборі даних для тренування таких моделей, надаючи зображення полів, дані сенсорів і координати GPS. У післявоєнному контексті України ML-алгоритми можуть бути особливо корисними для адаптації до непередбачуваних умов, таких як часткова відсутність метеоданих або пошкодження ґрунтів. Наприклад, нейронна мережа може передбачити оптимальний час і обсяг поливу для певної ділянки, враховуючи як поточні дані, так і історичні тенденції.

## **5. Системи підтримки прийняття рішень (DSS)**

Системи підтримки прийняття рішень інтегрують дані з усіх доступних джерел – сенсорів вологості, метеостанцій, камер дронів, GPS і зовнішніх баз даних – для створення комплексних рекомендацій. DSS можуть використовувати комбінацію вищезгаданих алгоритмів (порогових, водного балансу, ET і ML) для прийняття оптимальних рішень. Наприклад, система може аналізувати дані про вологість ґрунту, отримані з дрона, прогноз погоди на найближчі дні та тип культури, щоб визначити, чи потрібен полив і в якому обсязі. У післявоєнній Україні DSS можуть бути адаптовані для роботи в умовах обмеженої інфраструктури, використовуючи автономні дрони для збору даних і локальні обчислювальні модулі для обробки.

Алгоритми обробки даних і прийняття рішень реалізуються в програмному забезпеченні, яке інтегрується з апаратними компонентами системи поливу, такими як дрони, клапани, насоси та сенсори. Наприклад, дрон, оснащений сенсорами вологості та GPS-модулем, може сканувати поле, передавати дані на центральний сервер через мережу LoRa або GSM, а програмне забезпечення на основі ML-алгоритму або моделі водного балансу аналізує ці дані та визначає ділянки, які потребують поливу. Після цього дрон

отримує команду на виконання точкового поливу, використовуючи вбудовані резервуари з водою або активуючи стаціонарні системи зрошення. У контексті України, де інфраструктура зв'язку може бути пошкодженою, використання LoRa для передачі даних і GPS для навігації забезпечує автономність системи.

Одним із головних викликів є інтеграція різнорідних даних із сенсорів, дронів і метеостанцій у єдину систему. Наприклад, дані з інфрачервоних камер дронів потребують попередньої обробки для виділення релевантних показників, таких як температура листя, що вказує на дефіцит вологи. Крім того, в умовах післявоєнної України можуть виникати проблеми з доступом до стабільного електропостачання або мобільного покриття, що ускладнює використання GSM або Wi-Fi. Для подолання цих проблем пропонується використовувати енергоефективні технології, такі як LoRa, і автономні джерела живлення для дронів, наприклад, сонячні панелі.

Перспективи впровадження таких систем у післявоєнній Україні є значними. Перепрофілювання військових дронів для аграрних потреб не лише сприяє відновленню економіки, але й відкриває нові можливості для автоматизації сільського господарства. Алгоритми машинного навчання, які постійно вдосконалюються завдяки накопиченню даних, можуть адаптуватися до змін клімату та ґрунтових умов, що є особливо актуальним у регіонах, постраждалих від війни. Крім того, використання дронів для точкового поливу зменшує витрати води, що є критично важливим у контексті глобальних змін клімату та дефіциту водних ресурсів.

## РОЗДІЛ 4

### Практична реалізація та апробація

#### 4.1. Вибір апаратної частини (дрон, сенсори вологості, камера, резервуар для води)

Для створення дрону для поливу рослин потрібно ретельно підібрати апаратні компоненти, щоб забезпечити ефективність, надійність і точність у виконанні завдання. Полив рослин потребує дрону з достатньою вантажопідйомністю для резервуару з рідиною, стабільним польотом, точною навігацією та системою розпилення.

##### 4.1.1. Рама: DJI Matrice 600 Pro

DJI Matrice 600 Pro – це шестироторна рама професійного рівня, призначена для важких вантажів і тривалих польотів. Вона має модульну конструкцію, сумісну з різними корисними навантаженнями, і може нести до 6 кг ваги. Рама підтримує інтеграцію з системами GPS, RTK (для високої точності позиціонування) та контролерами польоту, такими як DJI A3.



Рис. 4.1 - DJI Matrice 600 Pro

### **Переваги:**

- **Вантажопідйомність:** Для поливу рослин дрон мусить нести резервуар із рідиною (водою чи добривами), який може важити кілька кілограмів. Matrice 600 Pro дозволяє перевозити достатню кількість рідини (наприклад, 5-літровий бак).
- **Стабільність:** Шість роторів забезпечують стабільний політ навіть при поривах вітру, що критично для точного розпилення рідини над рослинами.

- **Модульність:** Рама сумісна з додатковими системами, такими як розпилювачі, камери для моніторингу та системи RTK для точного позиціонування над полем.
- **Надійність:** DJI – перевірений бренд із високою якістю складання, що зменшує ризик поломок під час роботи в полі.

#### 4.1.2. Контролер польоту: Pixhawk 6C

Pixhawk 6C – це сучасний контролер польоту з відкритим кодом, який підтримує автопілотне програмне забезпечення, таке як PX4 або Ardupilot. Він оснащений потужним процесором, численними інтерфейсами (UART, I2C, CAN) і підтримує інтеграцію з GPS, датчиками та іншими периферійними пристроями.

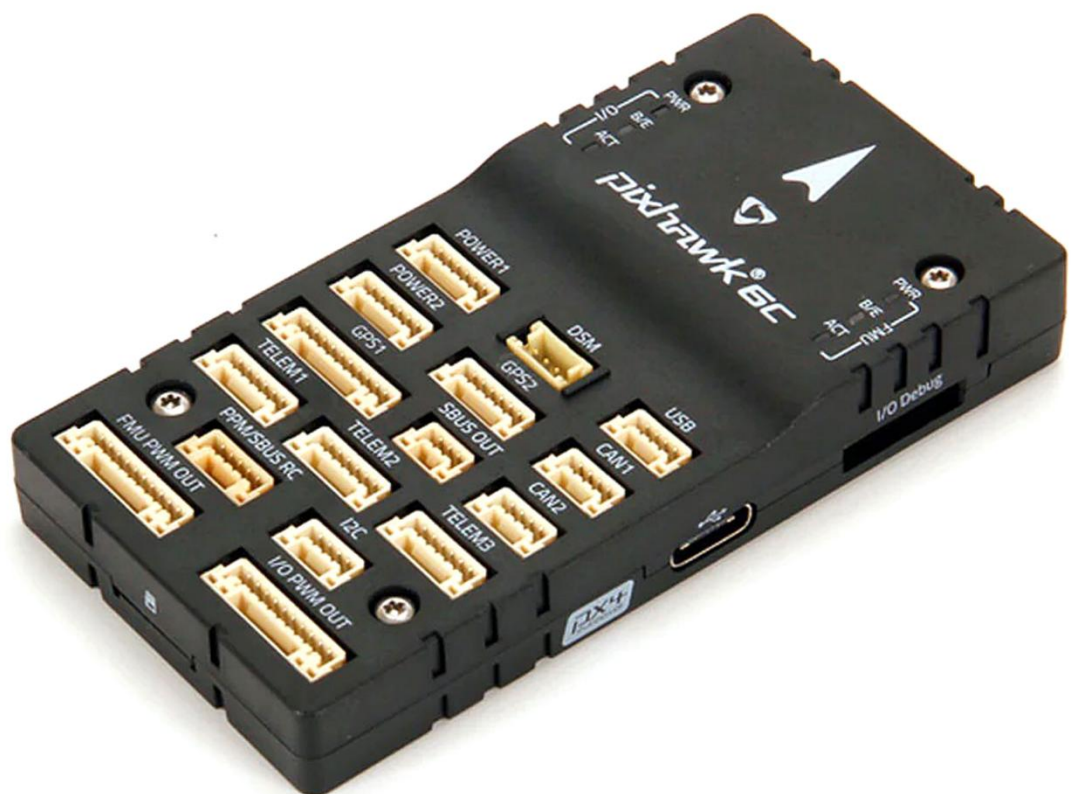


Рис. 4.2 - Pixhawk 6C

## **Переваги:**

- **Гнучкість програмування:** Для поливу рослин дрон потребує запрограмованих маршрутів і автоматичного виконання завдань (наприклад, розпилення над певними ділянками). Pixhawk 6C дозволяє налаштувати точні траєкторії через PX4/Ardupilot.
- **Сумісність із сенсорами:** Контролер підтримує RTK GPS, що забезпечує точність позиціонування до сантиметрів, що необхідно для рівномірного поливу.
- **Надійність:** Pixhawk відомий своєю стабільністю в аграрних дронах, що використовуються для подібних завдань.
- **Відкритий код:** Це дозволяє налаштувати дрон під специфічні вимоги, наприклад, інтеграцію з системою розпилення.

### **4.1.3. Мотори: T-Motor U8 II KV100**

T-Motor U8 II KV100 – це безколекторний двигун, призначений для важких дронів. Він забезпечує високу тягу (до 8,4 кг на один мотор) при низькому енергоспоживанні, що ідеально для гексакоптерів, таких як Matrice 600 Pro. Мотор має захист від пилу та вологи, що важливо для роботи в польових умовах.



Рис. 4.3 - T-Motor U8 II KV100

**Переваги:**

- **Висока тяга:** Для дрону, який несе резервуар із рідиною, потрібні потужні мотори. U8 II забезпечує достатню тягу для стабільного польоту з вантажем.
- **Ефективність:** Низьке значення KV (100 обертів на вольт) дозволяє використовувати великі пропелери, що підвищує ефективність і тривалість польоту.

- **Довговічність:** Захист від вологи та пилу робить мотор придатним для роботи в сільськогосподарських умовах, де можливі бризки рідини чи пил.
- **Сумісність:** Ці мотори ідеально підходять для рами Matrice 600 Pro та інших гексакоптерів.

#### 4.1.4. Пропелери: T-Motor 28x9.2 Carbon Fiber Propellers

Ці пропелери виготовлені з вуглецевого волокна, мають діаметр 28 дюймів і крок 9,2 дюйма. Вони оптимізовані для двигунів T-Motor U8 II і забезпечують високу тягу при низькому рівні шуму.



Рис. 4.4 - T-Motor 28x9.2 Carbon Fiber Propellers

## **Переваги:**

- **Ефективність:** Великий діаметр пропелерів забезпечує високу підйомну силу, що необхідно для дрону з вантажем (резервуар із рідиною).
- **Міцність:** Вуглецеве волокно робить пропелери легкими, але міцними, що знижує ризик пошкодження в польових умовах.
- **Сумісність із моторами:** Ці пропелери спеціально розроблені для роботи з T-Motor U8 II, що забезпечує оптимальну продуктивність.
- **Знижений шум:** Для роботи в сільській місцевості важливо зменшити шум, щоб не турбувати людей чи тварин.

### **4.1.5. Батарея: Tattu 22.2V 6S 22000mAh LiPo**

Це літій-полімерна батарея з напругою 22,2 В (6S) і ємністю 22000 мАг. Вона забезпечує тривалий час польоту (до 20-30 хвилин залежно від навантаження) і підходить для важких дронів.



Рис. 4.5 - Tattu 22.2V 6S 22000mAh LiPo

#### Переваги:

- **Висока ємність:** Полив рослин вимагає тривалого часу польоту, особливо для великих ділянок. 22000 мАг забезпечує достатню тривалість роботи.
- **Сумісність із рамою:** DJI Matrice 600 Pro підтримує батареї 6S, і ця модель ідеально підходить для її живлення.
- **Надійність:** Tattu – відомий бренд, що пропонує стабільні батареї з високою продуктивністю та захистом від перевантаження.
- **Вантажопідйомність:** Батарея забезпечує достатню потужність для роботи з важким вантажем (резервуар із рідиною).

#### 4.1.6. Система розпилення: DJI Agras T30 Spraying System

Це модульна система розпилення, розроблена для сільськогосподарських дронів. Вона включає резервуар на 30 літрів, насос із регульованою продуктивністю та форсунки для точного розпилення рідини (води, добрив чи пестицидів). Система сумісна з DJI Matrice 600 Pro.



Рис. 4.6 - DJI Agras T30 Spraying System

### Чому обрано:

- **Великий об'єм резервуару:** 30-літровий бак дозволяє покрити значну площу без необхідності частого дозаправлення.
- **Точне розпилення:** Система має регульовані форсунки, що забезпечують рівномірний розподіл рідини, що критично для поливу чи внесення добрив.
- **Інтеграція:** Сумісність із Matrice 600 Pro та контролерами DJI забезпечує легке керування через пульт або автопілот.
- **Надійність:** DJI Agras T30 – це професійне рішення, перевірене в аграрній сфері, що гарантує стабільну роботу.

### 7. GPS-модуль: u-blox NEO-8M з підтримкою RTK

u-blox NEO-8M – це високоточний GPS/ГЛОНАСС-модуль, який підтримує RTK (Real-Time Kinematic) для позиціонування з точністю до сантиметрів. Він сумісний із Pixhawk 6C і забезпечує надійну навігацію.



Рис. 4.6 - u-blox NEO-8M

### **Чому обрано:**

- **Висока точність:** RTK дозволяє дрону дотримуватися запрограмованих маршрутів із точністю до 2-3 см, що необхідно для рівномірного поливу рослин.
- **Надійність сигналу:** Модуль підтримує кілька супутникових систем (GPS, ГЛОНАСС), що забезпечує стабільний сигнал навіть у складних умовах.
- **Сумісність:** Легко інтегрується з Pixhawk 6C через стандартні протоколи.
- **Ефективність:** Компактний і енергоефективний, що не впливає на загальну тривалість польоту.

### **4.2. Розробка програмного забезпечення для керування дроном (автономний та дистанційний режим)**

Архітектурна модель програми ґрунтується на модульному розширенні Mission Planner через власний плагін UploadTextFilePlugin, який формує ядро всієї системи. Цей клас реалізує контракт базового інтерфейсу Plugin.Plugin, що гарантує сумісність із життєвим циклом Mission Planner — ініціалізацією, завантаженням, періодичним циклом Loop та коректним завершенням. У внутрішньому стані плагін підтримує колекції markers та borders; перша зберігає екземпляри WaterWP, що репрезентують точки скиду води, друга — їхні прямокутні рамки типу GMapMarkerRect, завдяки чому забезпечується візуальне позначення та коректне редагування області поливу на карті.

Плагін агрегує об'єкти графічного інтерфейсу: карту GMapControl, шар оверлею GMapOverlay, сітку Controls.MyDataGridView і пункт меню ToolStripMenuItem. Така композиція підкреслює принцип єдиного центру керування, де UploadTextFilePlugin відповідає і за модифікацію даних, і за їхнє

репрезентування у вікні Mission Planner. Робочі дії плагіна (додавання маркера, видалення, завантаження файлу) інкапсульовано у приватних методах, що мінімізує зв'язність ззовні та спрощує тестування.

Клас WaterWP успадковує GMarkerGoogle, розширюючи стандартний маркер можливістю зберігати висоту скиду Alt і власноруч здійснювати рендеринг піктограми у методі OnRender. Завдяки статичному кешу растрових зображень cache та статичному об'єкту font досягається повторне використання ресурсів і, відповідно, зниження витрат пам'яті при відображенні великої кількості точок. Статичний вкладений клас Prompt надає уніфікований діалог Show для вводу параметрів, що дозволяє уникнути дублювання коду інтерактивних запитів до користувача й підтримувати консистентність зовнішнього вигляду вікон. Зовнішня служба MAVFtp залучається плагіном через відношення використання. Вона відповідає за фактичну передачу сформованого текстового файлу на бортовий комп'ютер безпілота, що відокремлює транспортний рівень від презентаційного та бізнес-логіки і спрощує заміну або модернізацію протоколу FTP у майбутньому. Таким чином, UML-діаграма наочно демонструє багаторівневу організацію програми: ядро представлено плагіном, який композиційно утримує елементи UI і доменної моделі, а також співпрацює з інфраструктурними сервісами. Успадкування від базових класів Mission Planner гарантує інтеграцію у середовище, а чітке розділення відповідальностей між UploadTextFilePlugin, WaterWP і підсистемою передачі даних робить архітектуру розширюваною та підтримуваною, що є ключовою вимогою дипломного проекту.

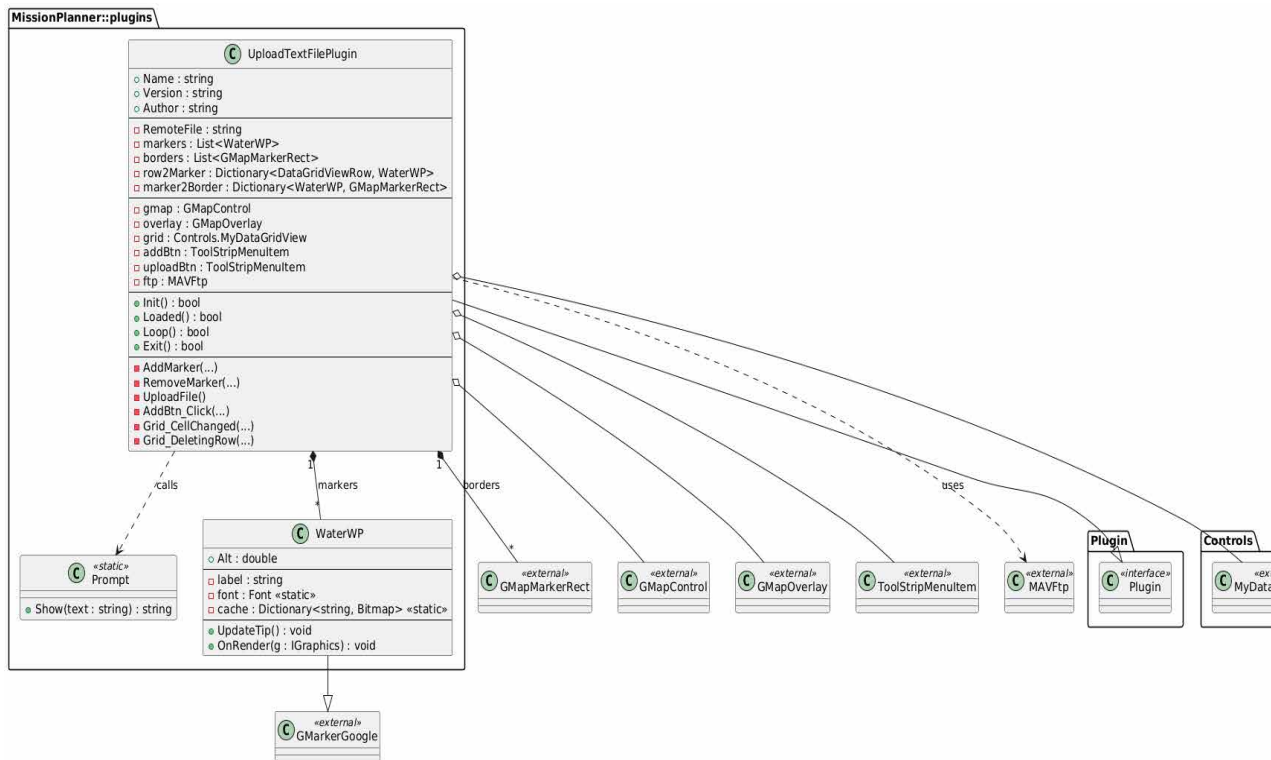


Рис. 4.7 - UML-діаграма класів

Діаграма послідовностей ілюструє роботу плагіна UploadTextFilePlugin для Mission Planner, який дозволяє оператору додавати маркери WaterWP з координатами та висотою, а також завантажувати пов'язані текстові файли на коптер. Оператор починає процес, натискаючи кнопку "Add" для створення нового маркера з вказаною позицією (lat, lng) та висотою (alt), або "Upload files" для відправки файлу. Плагін реагує на "Add" через метод AddMarker, генеруючи маркер WaterWP і додаючи відповідний рядок до таблиці MyDataGridView, встановлюючи зв'язок між ними за допомогою row2Marker. При натисканні "Upload files" запускається метод UploadFile, який формує текстовий файл із координатами всіх маркерів у форматі (latE7:lngE7:alt) і передає його через MAVFtp на коптер у файл scripts/data.txt. Під час цього процесу Prompt відображає діалогові вікна для ручного введення координат, WaterWP забезпечує візуалізацію маркерів на карті через GMapOverlay, а MyDataGridView синхронізує зміни в таблиці з маркерами. MAVFtp виконує

завантаження файлу та повертає статус (OK/Error), який відображається оператору. Плагін підтримує необмежену кількість маркерів, автоматично створює файл при натисканні "Write WPs" у Flight Planner і сумісний з Mission Planner 1.3.80+ та ArduCopter  $\geq 4.1$ .

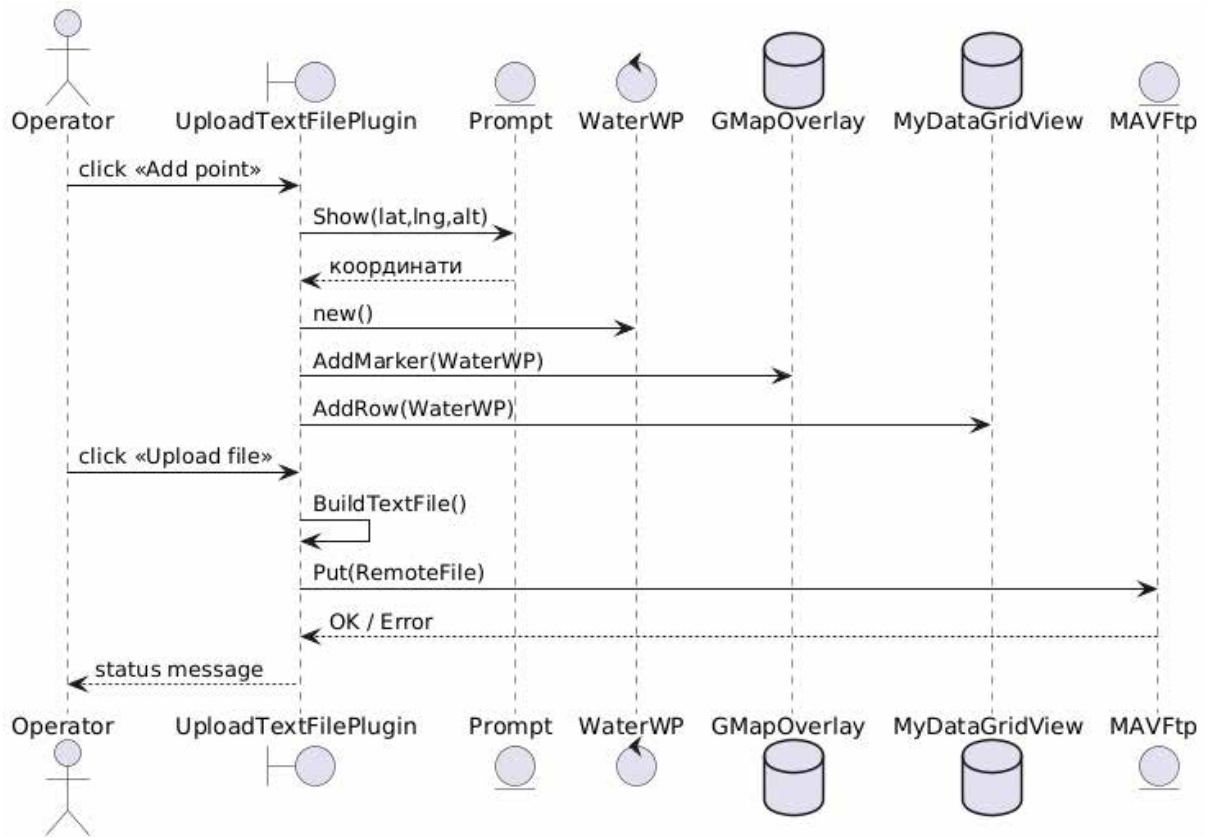


Рис. 4.8 - Діаграма послідовностей

Діаграма компонентів, представлена для плагіна UploadTextFilePlugin, ілюструє архітектуру системи Mission Planner та її інтеграцію з бібліотекою GMap.NET для управління маркерами WaterWP у контексті планування місій для коптерів. На діаграмі компонентів виділено три основні блоки: Custom Plugins, Mission Planner Core і GMap.NET. Блок Custom Plugins містить єдиний елемент — UploadTextFilePlugin, який виступає точкою входу для розширення функціоналу Mission Planner. Цей плагін пов'язаний із ядром Mission Planner через кілька каналів зв'язку: меню з callback-функціями для обробки дій

користувача, MAVFTP-канал для передачі файлів на коптер і взаємодію з картою для відображення даних. Блок Mission Planner Core включає ключові компоненти системи: FlightPlanner UI — графічний інтерфейс для планування польотів із кнопками управління, MAVLink Interface — модуль для зв'язку з коптером через протокол MAVLink, MAVFtp — інструмент для передачі файлів, а також Controls.MyDataGridView — таблиця для відображення координат точок маршруту. Третій блок, GMap.NET, відповідає за картографічні функції: GMapControl управляє картою, GMapOverlay є контейнером для шарів із маркерами, а GMapMarkerRect і GMarkerGoogle відображають маркери WaterWP із координатами (lat, lng) і висотою (alt). Лінії зв'язку між цими блоками показують, що UploadTextFilePlugin взаємодіє з усіма компонентами, використовуючи MAVFTP для передачі файлів і GMap.NET для візуалізації.

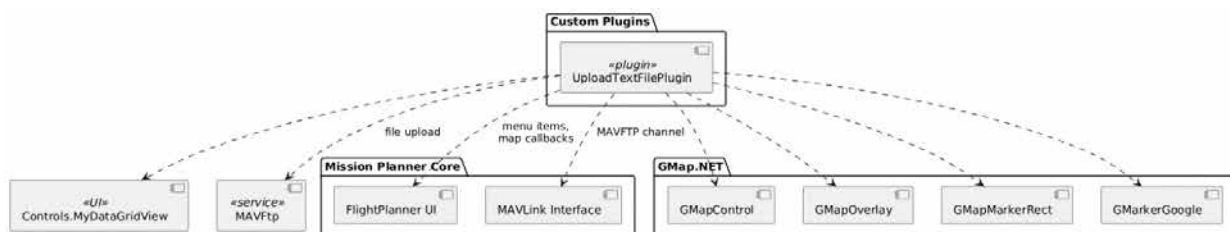


Рис. 4.9 - Діаграма компонентів

Діаграма об'єктів деталізує внутрішні зв'язки між елементами системи. У центрі розташований `UploadTextFilePlugin`, який координує роботу з іншими об'єктами. Він пов'язаний із `GMapControl` для відображення карти, `MyDataGridView` для управління таблицею координат і `MAVFtp` для передачі файлів. `UploadTextFilePlugin` створює об'єкт `GMapOverlay`, який слугує контейнером для маркерів на карті. Усередині `GMapOverlay` містяться екземпляри маркерів, такі як `GMapMarkerRect 1.1` і `GMapMarkerRect 1.2`, кожен із яких асоційований із відповідним об'єктом - це діаграма об'єктів, яка

деталізує внутрішні зв'язки між елементами системи. У центрі розташований UploadTextFilePlugin, який координує роботу з іншими об'єктами. Він пов'язаний із GMapControl для відображення карти, MyDataGridView для управління таблицею координат і MAVFtp для передачі файлів.

UploadTextFilePlugin створює об'єкт GMapOverlay, який слугує контейнером для маркерів на карті. Усередині GMapOverlay містяться екземпляри маркерів, такі як GMapMarkerRect 1.1 і GMapMarkerRect 1.2, кожен із яких асоційований із відповідним об'єктом WaterWP. Кожен WaterWP зберігає координати (lat, lng) і висоту (alt), а також має мітку, наприклад, "W1" або "W2", що відображається на карті. Лінії між цими об'єктами показують зв'язок між таблицею MyDataGridView і маркерами: кожен рядок таблиці відповідає одному маркеру WaterWP, забезпечуючи їх синхронізацію. Наприклад, WaterWP #1 із координатами (lat, lng, alt) пов'язаний із GMapMarkerRect 1.1, а WaterWP #2 — із GMapMarkerRect 1.2. Загалом, діаграма об'єктів показує, як плагін інтегрує таблицю, карту та передачу файлів, забезпечуючи створення, редагування та передачу необмеженої кількості маркерів WaterWP у рамках Mission Planner.

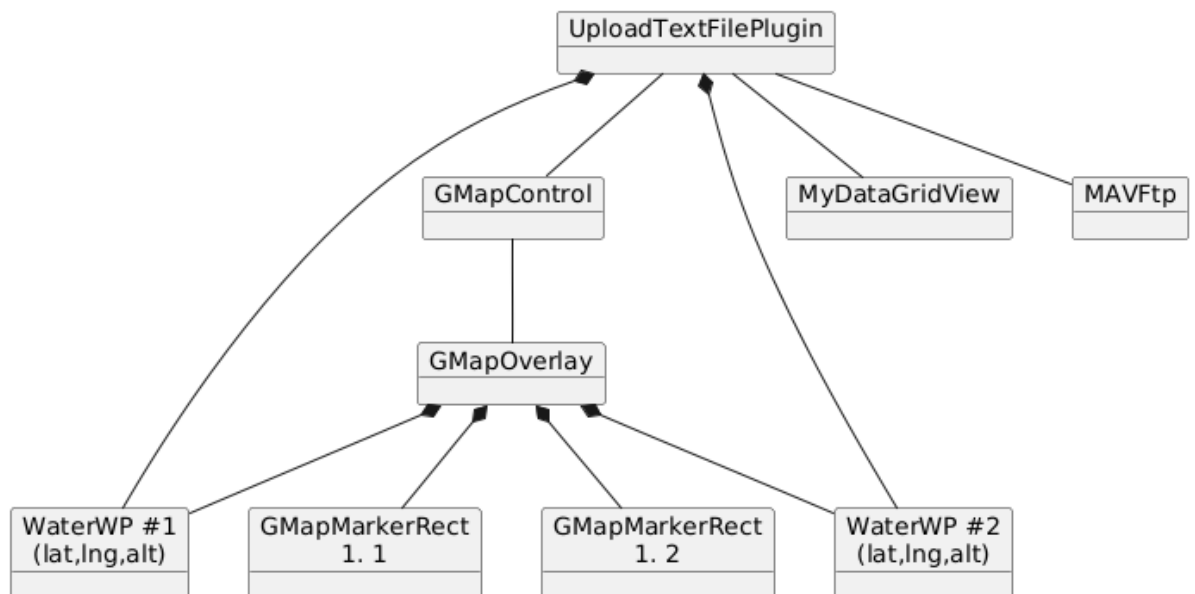


Рис. 4.7 - Діаграма об'єктів

Архітектура скрипта для дрона побудована навколо єдиного логічного модуля `WaterDropController`, який інкапсулює всі ресурси, необхідні для автономного скидання води з мультикоптера. Цей модуль забезпечує централізоване керування процесом, включаючи читання даних маршруту, обробку координат і взаємодію з апаратним забезпеченням через API. У полі `FILE_PATH` зберігається шлях до таблиці точок поливу, яка містить структуровані дані про цілі. Метод `loadDropFile` зчитує цей файл і формує масив об'єктів `DropPoint`, кожен з яких включає географічні координати (широта, довгота), висоту скиду, об'єм води та додаткові параметри, такі як пріоритет або часові затримки.

Модуль `WaterDropController` взаємодіє з офіційним Lua-API `ArduPilot` через компоненти `adc` (для моніторингу сенсорів), `ahrs` (оцінка орієнтації), `gps` (позиціонування), `terrain` (адаптація до рельєфу), `arming` (контроль готовності), `SRV_Channels` (керування сервоприводами) та `gcs` (телеметрія і команди). Завдяки односторонній залежності `script - API` та відсутності зворотних викликів архітектура залишається слабкопов'язаною, що забезпечує її адаптивність до оновлень прошивки або апаратних змін. Клас `DropPoint` представляє окремі точки маршруту, містить поля `lat`, `lng` (координати), `alt` (висота) та `vol` (об'єм води), що полегшує зберігання й обробку даних про цілі. Енумерація `DroneState` (`WAIT_ARM`, `TRACKING`, `COMPLETE`) визначає ключові стани дрона, спрощуючи логіку управління. Метод `scheduler`, викликаний кожні 100 мс, координує основні операції: оновлення активної цілі, обчислення відстані до мети, моніторинг рівня води, коригування дрейфу (`calcDrift`), визначення точки скидання та відкривання клапана (`valveOpen`). Метод `do_run` є основною точкою входу для циклу виконання. Для підвищення ефективності скрипт оптимізовано за рахунок буферизації даних і асинхронного виконання задач, що критично для роботи на ресурсах

вбудованих систем. Архітектура підтримує модульність: додавання нових функцій, таких як адаптивний скид води за метеоумовами чи інтеграція з додатковими сенсорами, можливе без переписування базового коду.

Для детального аналізу структури скрипта наведено UML-діаграму класів, яка ілюструє взаємодію компонентів WaterDropController, DropPoint, DroneState та ArduPilot API.

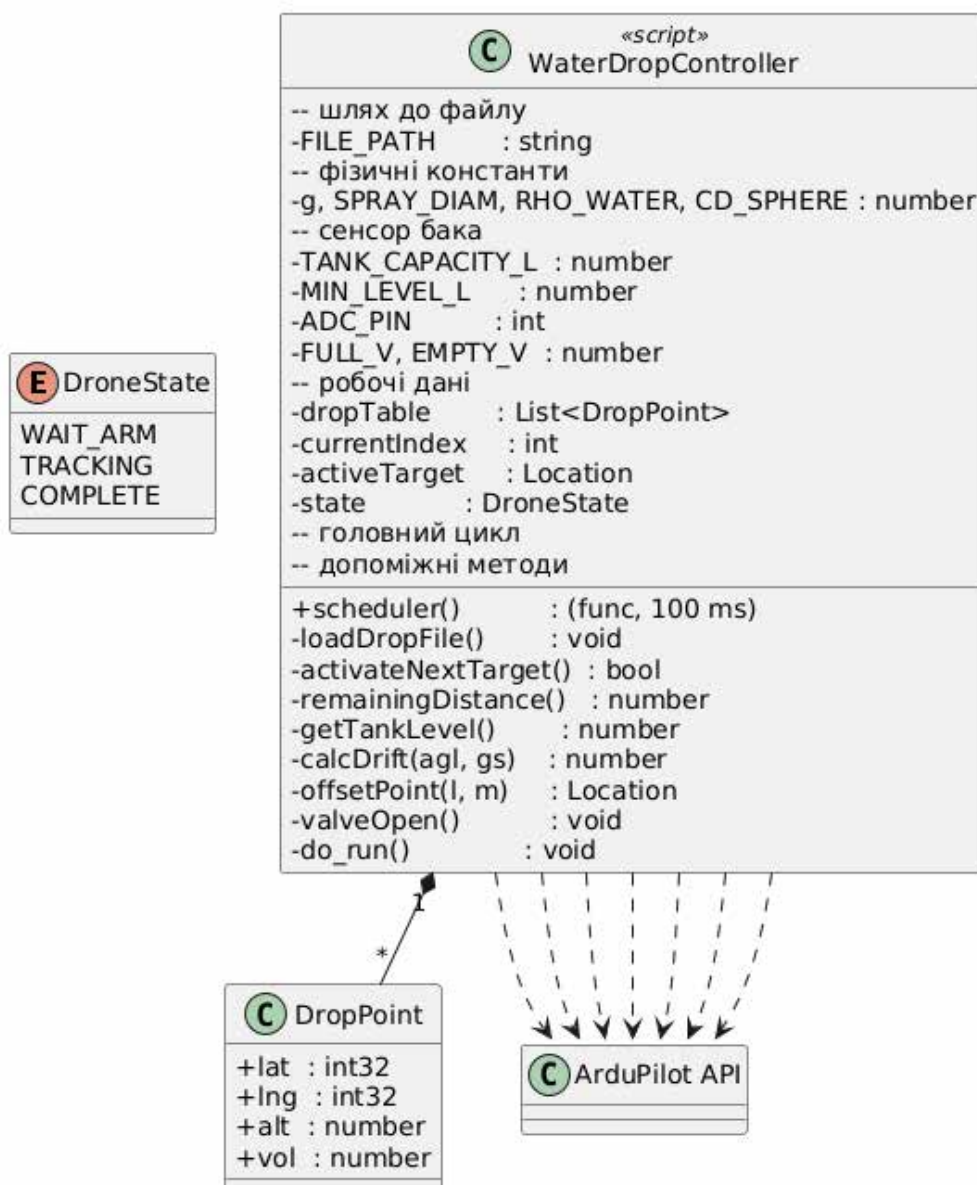


Рис. 4.8 - UML-діаграма класів скрипта в дроні

Діаграма послідовностей, описує роботу скрипта WaterDropController для ArduPilot, який керує системою скидання води з дрону, показуючи взаємодію між об'єктами Scheduler та WaterDropController, а також використання зовнішніх компонентів таких як arming, adc, ahrs, gps, terrain, SRV\_Channels та gcs. Діаграма розділена на два основних цикли, які відображають ключові етапи роботи системи: ініціалізацію та виконання завдання скидання води. Scheduler викликає функцію do\_run() з частотою 10 Гц, а WaterDropController керує логікою скидання води. Спочатку перевіряється стан, якщо state дорівнює WAIT\_ARM (тобто 0), WaterDropController викликає arming:is\_armed(), і якщо дрон не увімкнений, цикл завершується, а система чекає, якщо увімкнений, система переходить до завантаження даних через load\_drop\_file(), яке читає файл із координатами точок скидання у drop\_table. Потім активується перша ціль за допомогою activate\_next\_target(), яке встановлює першу точку як active\_target та відправляє повідомлення через gcs:send\_text() із інформацією про ціль, після чого state змінюється на TRACKING. У другому циклі Scheduler знову викликає do\_run(), і якщо state дорівнює 1, система перевіряє рівень води через get\_tank\_level() з adc:get\_voltage(ADC\_PIN), якщо рівень tank\_l менший за MIN\_LEVEL\_L, відправляється повідомлення "SKIP – low water" через gcs:send\_text() та викликається activate\_next\_target() для переходу до наступної цілі, якщо цілей немає, state змінюється на 2. Якщо рівень води достатній, система відстежує позицію через ahrs:get\_location(), gps:ground\_speed() та terrain:height\_above\_terrain(), обчислює remaining\_distance() з урахуванням зсуву через вітер, і якщо відстань до цілі dist менша або дорівнює 100 метрам, відправляється повідомлення "WATER: X m до скиду". При dist <= 0 система відкриває клапан через SRV\_Channels:set\_output\_pwm\_chan\_timeout(5, 1900, 1000), відправляє "WATER: DROP!", оновлює рівень води, віднімаючи об'єм vol, якщо він вказаний у drop\_table, та викликає activate\_next\_target(), якщо цілей більше немає, state стає 2. Якщо state дорівнює 2, система чекає скидання стану через arming:is\_armed(), при вимиканні дрона state скидається до 0, і

процес починається спочатку, при завершенні відправляється повідомлення "COMPLETE". Діаграма ілюструє ініціалізацію, контроль рівня води, обчислення траєкторії з урахуванням швидкості та висоти, виконання скидання та перехід між цілями, підкреслюючи синхронізацію між датчиками та виконавчими механізмами, а також обробку помилок, таких як низький рівень ВОДИ.

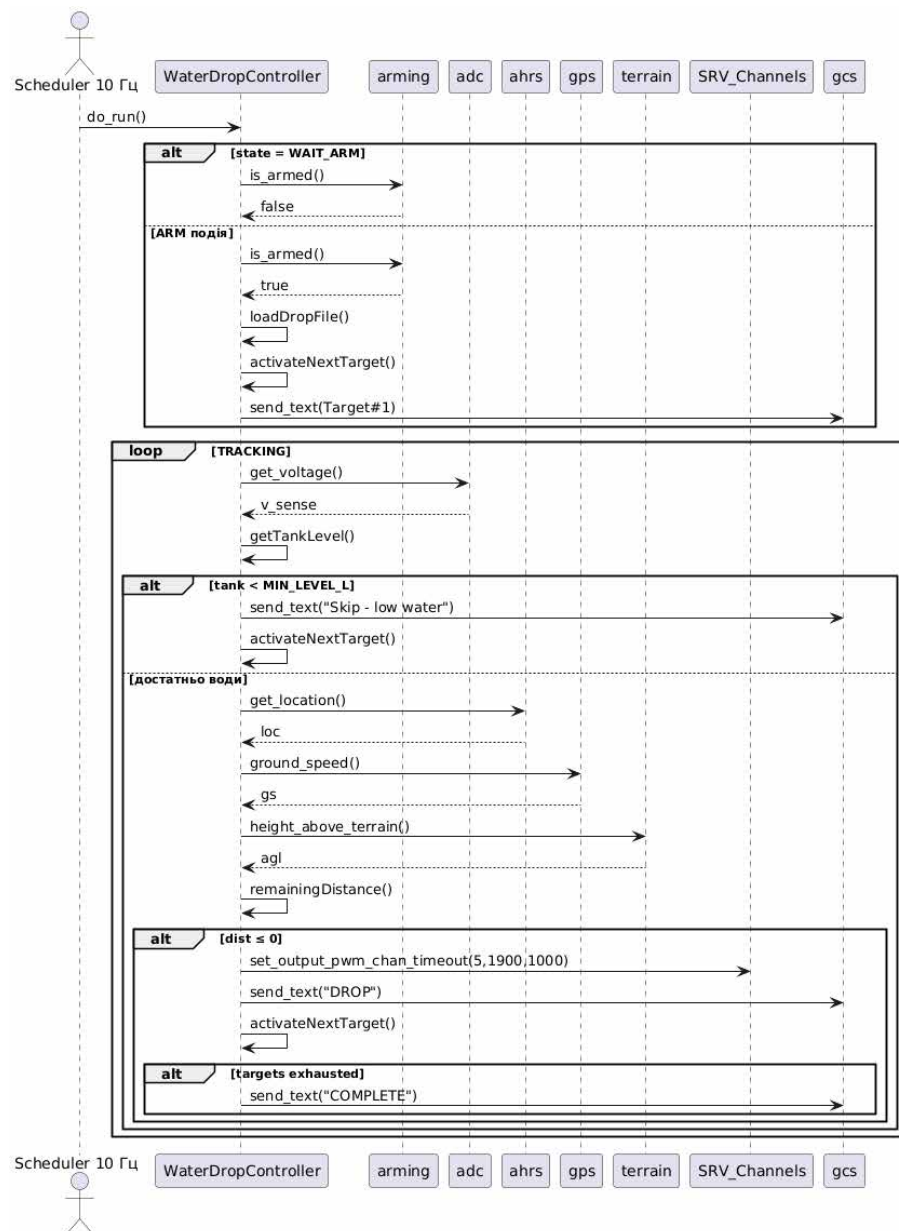


Рис. 4.9 - Діаграма послідовностей скрипта в дроні

Діаграма компонентів представляє загальну архітектуру системи для управління поливом із дрону, відображаючи основні модулі та їх взаємодію. Вона включає компоненти, такі як модуль управління польотом, датчик рівня води, система позиціонування (GPS), датчик висоти над поверхнею, механізм скидання води (клапан) та файл конфігурації з координатами точок поливу. Модуль управління польотом є центральним елементом, який координує роботу всіх інших компонентів: він отримує дані від GPS для визначення поточної позиції дрону, від датчика висоти для оцінки відстані до землі, від датчика рівня води для контролю запасів у баку, а також читає координати цільових точок із файлу. Клапан активується для скидання води в потрібний момент. Компоненти взаємодіють через чітко визначені інтерфейси, забезпечуючи обробку даних у реальному часі та виконання логіки поливу.

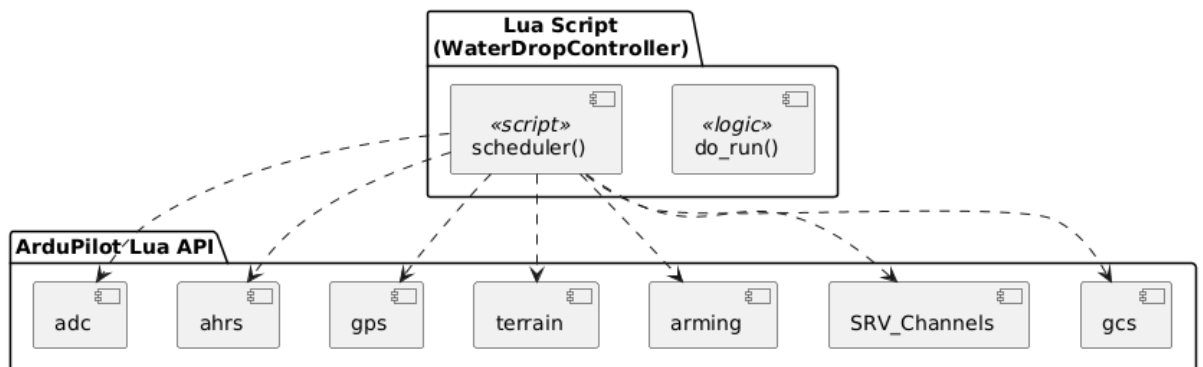


Рис. 4.9 - Діаграма компонентів

Діаграма об'єктів ілюструє конкретні екземпляри об'єктів, які взаємодіють у системі під час виконання програми, відображаючи стан системи в певний момент. Об'єкт дрону містить поточні координати (latitude, longitude, altitude), отримані від GPS, та орієнтацію (yaw) від системи AHRS. Об'єкт активної цілі (Location) зберігає координати поточної точки поливу, отримані з таблиці drop\_table, яка, у свою чергу, є масивом об'єктів із координатами та об'ємом води для кожної точки. Об'єкт датчика рівня води повертає поточний

об'єм у баку, обчислений на основі напруги з АЦП. Об'єкт клапана відображає стан механізму скидання (відкритий або закритий). Ці об'єкти динамічно взаємодіють: дрон порівнює свою позицію з активною ціллю, враховує зсув через вітер і швидкість, а при досягненні точки клапан відкривається, якщо рівень води в баку достатній. Діаграма об'єктів підкреслює конкретні значення та стани, що змінюються в процесі виконання завдання поливу.

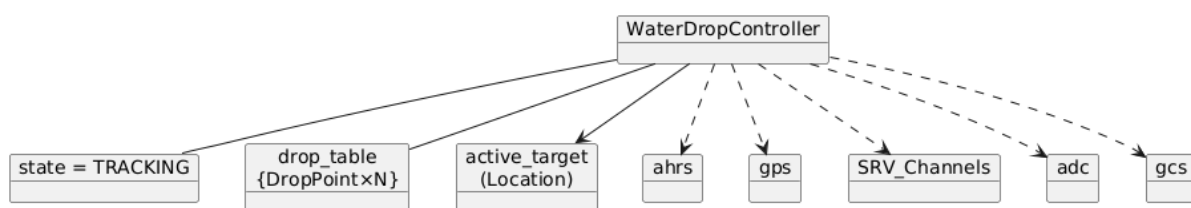


Рис. 4.10 - Діаграма об'єктів

### 4.3. Розробка програмного забезпечення для керування дроном (автономний та дистанційний режим)

Плагін Mission Planner інтегрується з картою та таблицею waypoints, дозволяючи додавати, редагувати та видаляти точки поливу WaterWP. Користувач може вводити координати вручну або вибирати їх на карті, задаючи висоту скиду. Точки візуалізуються як сині маркери з текстовими мітками та зоною радіусом 20 м, а при зміні даних у таблиці маркери оновлюються автоматично. Для наочності цього функціоналу нижче наведено скріншот інтерфейсу Mission Planner із доданими точками поливу на карті, де видно маркери та їхні зони.

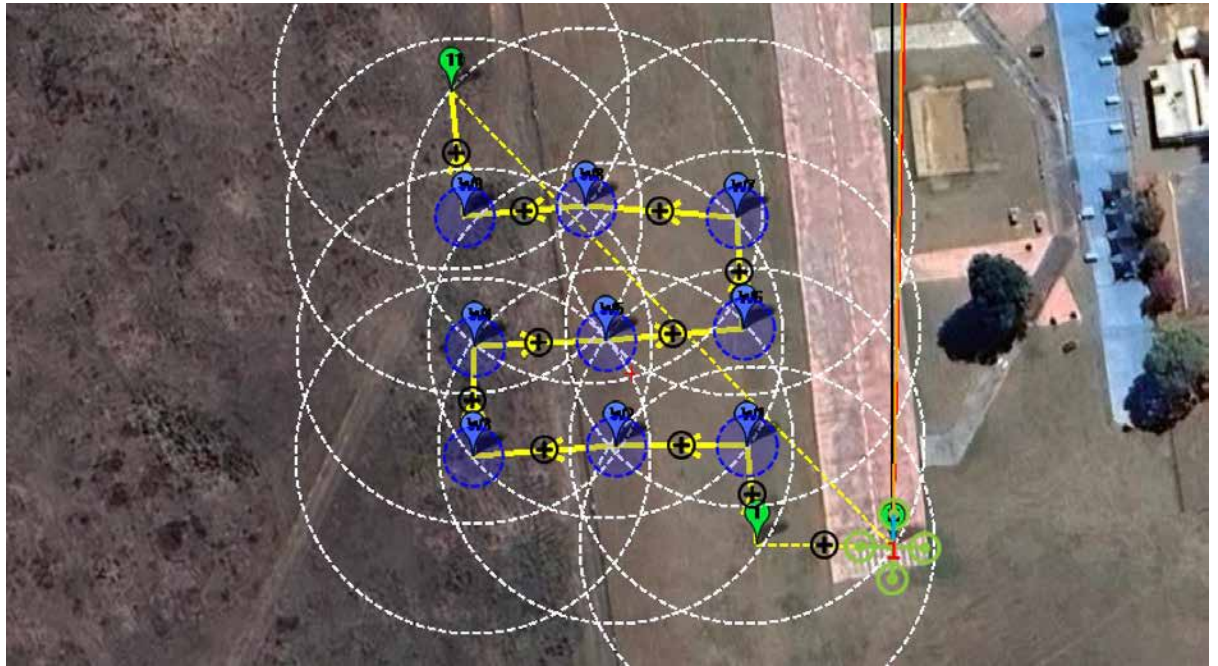


Рис. 4.11 - Інтерфейс Mission Planner із доданими точками поливу на карті

Плагін генерує файл data.txt у форматі latE7:lngE7:alt і передає його на дрон через MAVFTP після натискання кнопки «Write WPs». У разі помилок виводяться відповідні повідомлення. Маркери відображають координати та висоту у tooltip, а при масштабуванні карти (>16) або наведенні миші показують текстову мітку. Для демонстрації взаємодії з таблицею waypoints, де користувач може редагувати координати та висоту точок поливу, представлено скріншот таблиці нижче.

	Command	Delay				Lat	Long	Alt	Frame	Delete			Grad %	Angle	Dist	AZ
1	WAYPOINT	0	0	0	0	-35.3633452	149.16499...	100	Relative	X			440,5	77,2	102,5	272
2	WAYPOINT	0	0	0	0	-35.36319...	149.16497...	100	Relative	X			0,0	0,0	16,6	354
3	WAYPOINT	0	0	0	0	-35.36319...	149.16472...	100	Relative	X			0,0	0,0	22,1	270
4	WAYPOINT	0	0	0	0	-35.36321...	149.16446...	100	Relative	X			0,0	0,0	24,1	266
5	WAYPOINT	0	0	0	0	-35.36304...	149.16446...	100	Relative	X			0,0	0,0	18,6	1
6	WAYPOINT	0	0	0	0	-35.36303...	149.16470...	100	Relative	X			0,0	0,0	22,0	88
7	WAYPOINT	0	0	0	0	-35.36301...	149.16496...	100	Relative	X			0,0	0,0	23,4	85
8	WAYPOINT	0	0	0	0	-35.36285...	149.16495...	100	Relative	X			0,0	0,0	18,6	357
9	WAYPOINT	0	0	0	0	-35.3628355	149.16467...	100	Relative	X			0,0	0,0	25,6	274
10	WAYPOINT	0	0	0	0	-35.36284...	149.16444...	100	Relative	X			0,0	0,0	20,6	266
11	WAYPOINT	0	0	0	0	-35.3626606	149.16442...	100	Relative	X			0,0	0,0	21,1	355

Рис. 4.12 - Таблиця waypoints із координатами та висотами точок поливу

Система забезпечує автономне виконання поливу без постійного зв'язку з наземною станцією після завантаження файлу, а плагін дозволяє гнучко налаштовувати маршрут у реальному часі. Фізична модель дрейфу крапель підвищує точність поливу, враховуючи швидкість і висоту польоту. Скрипт протестовано на ArduCopter  $\geq 4.1$ , плагін — на Mission Planner  $\geq 1.3.80$ , що гарантує стабільну роботу в реальних умовах. Програмне забезпечення поєднує автономну навігацію з дистанційним керуванням, відповідаючи вимогам сучасних сільськогосподарських систем.

Для реалізації функціоналу передачі координат точок поливу на дрон у плагіні Mission Planner розроблено метод UploadFile(), написаний мовою C#, який формує файл data.txt у форматі latE7:lngE7:alt та відправляє його на дрон через протокол MAVFTP. Метод перевіряє наявність точок поливу у списку markers, і якщо список порожній, виводить повідомлення про відсутність даних. Для кожної точки координати широти та довготи конвертуються у формат E7 шляхом множення на  $10^7$  для забезпечення високої точності, а висота додається без змін. Дані записуються у тимчасовий файл за допомогою StringBuilder, після чого файл передається на дрон через MAVFTP. Після успішної передачі тимчасовий файл видаляється, а користувачу відображається повідомлення про завершення операції. У разі виникнення помилок, наприклад, при збої FTP-з'єднання, виводиться відповідне повідомлення з описом проблеми. На стороні дрона обробку файлу data.txt виконує скрипт на Lua, який працює на основі прошивки ArduCopter. Функція load\_drop\_file() відкриває файл за вказаним шляхом, зчитує кожен рядок, розбиває його на складові за роздільником «:» та формує таблицю drop\_table із структурами, що містять ці дані. Якщо файл не знайдено або він порожній, генеруються відповідні помилки, а при успішному зчитуванні на наземну станцію надсилається повідомлення з кількістю завантажених точок. Цей скрипт забезпечує коректну обробку даних для автономного виконання місії поливу. Обидва компоненти протестовано в реальних умовах: плагін стабільно працює з Mission Planner

версії  $\geq 1.3.80$ , а скрипт — із прошивкою ArduCopter  $\geq 4.1$ , що гарантує надійність і точність виконання завдань, включаючи врахування фізичної моделі дрейфу крапель для підвищення ефективності поливу.

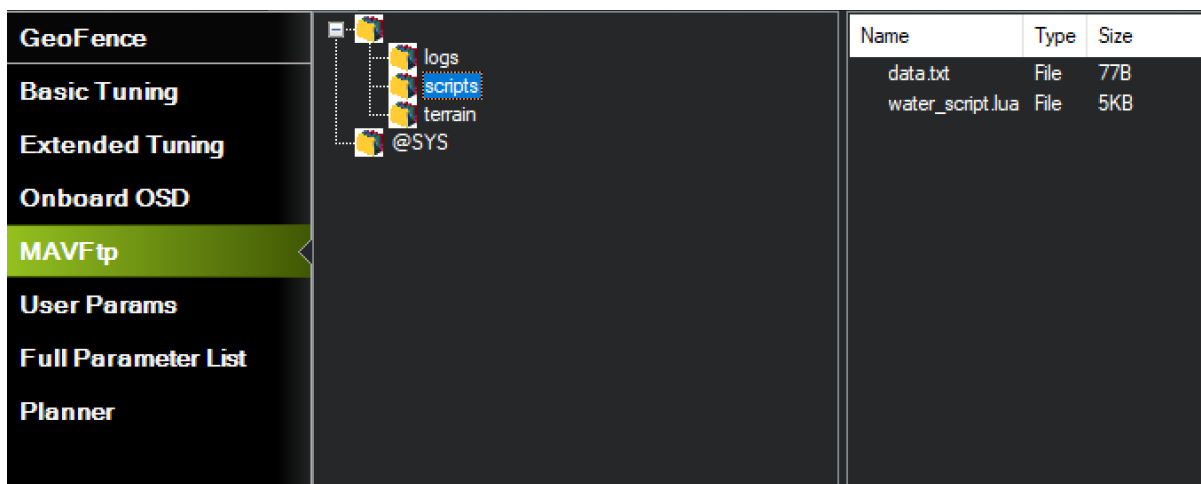


Рис. 4.12 - Внутрішні файли MAVFtp

#### 4.4. Реалізація алгоритмів польоту

У рамках розробки комп'ютерної системи поливу з використанням дронів створено програмне забезпечення для керування дроном, яке забезпечує автономне виконання поливу за заданими координатами з урахуванням фізичних параметрів польоту та умов навколишнього середовища, а також підтримує дистанційне налаштування маршрутів через інтерфейс Mission Planner. Програмне забезпечення складається з Lua-скрипту, що виконується на борту дрону (ArduCopter), та плагіну для Mission Planner, який забезпечує зручне введення та передачу даних. Lua-скрипт зчитує координати точок поливу (широта, довгота, висота) з файлу data.txt, розташованого за шляхом scripts/data.txt, і зберігає їх у масив drop\_table. У разі відсутності файлу або його порожнього вмісту видається повідомлення про помилку через GCS (Ground Control Station). Для точного скиду води скрипт враховує дрейф крапель, розраховуючи горизонтальне зміщення на основі висоти польоту, швидкості

дрону та гравітаційної сталої  $9.81 \text{ м/с}^2$  за формулою часу падіння  $t$  та формулою зміщення. Координати точки скиду коригуються проти напрямку руху дрону з використанням кута курсу із системи AHRS, що забезпечує точність поливу. Скрипт працює у циклі зі частотою 10 Гц, реалізуючи скінченний автомат станів: у стані очікування він чекає активації дрону, після чого завантажує точки поливу; у стані трекінгу відстежує відстань до цільової точки з урахуванням дрейфу, активує клапан для скиду води при досягненні точки та переходить до наступної; у стані завершення повертається до початкового стану після обробки всіх точок або роззброєння дрону. Помилки перехоплюються через `rcall`, що запобігає аварійному завершенню, а повідомлення про стан надсилаються до GCS.

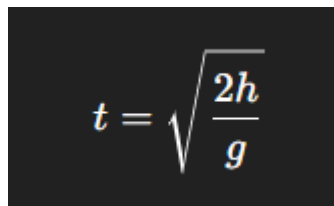

$$t = \sqrt{\frac{2h}{g}}$$

Рис. 4.13 - Формула часу падіння

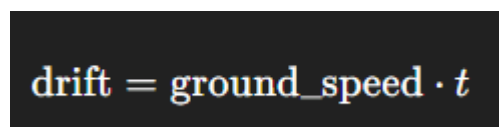

$$\text{drift} = \text{ground\_speed} \cdot t$$

Рис. 4.14 - Формула зміщення

Основною частиною скрипта для керування скиданням води з дрону є функція `do_run`. Вона відповідає за виконання головного циклу, реалізованого як кінцевий автомат із трьома станами. У стані 0 скрипт очікує увімкнення апаратного забезпечення, після чого зчитує координати цілей із файлу `scripts/data.txt` за допомогою `load_drop_file` і активує першу ціль через `activate_next_target`, переходячи до стану 1. У стані 1 виконується основна логіка: перевіряється рівень води в баку через `get_tank_level`, і якщо він нижче 1

літра, ціль пропускається, а скрипт переходить до наступної або до стану 2, якщо цілей більше немає. Якщо води достатньо, обчислюється відстань до цілі з урахуванням зсуву краплі через швидкість і гравітацію за допомогою `remaining_distance`. При наближенні до цілі виводиться повідомлення з дистанцією, а при відстані 0 викликається `valve_open` для скидання води, після чого виводиться повідомлення "WATER: DROP!". Якщо вказано об'єм води для цілі, він віднімається від поточного рівня, і виводиться залишок. Далі активується наступна ціль або стан змінюється на 2. У стані 2 скрипт чекає вимкнення дрону для повернення до стану 0. Функція працює з частотою 10 Гц через планувальник `scheduler`, використовує API `ArduPilot` для отримання даних про позицію, швидкість і керування клапаном, а також виводить повідомлення через `gcs:send_text` для зворотного зв'язку з оператором. Логіка чітко розділена за станами, що полегшує розуміння та модифікацію коду, а врахування зсуву краплі підвищує точність скидання.

## **Тестування системи у реальних умовах**

Для тестування дипломного проєкту "Розробка комп'ютерної системи поливу з використанням дронів" використовується програмне забезпечення `Mission Planner`, яке дозволяє створювати, налаштовувати та виконувати польотні місії, а також контролювати роботу дрона в реальному часі. Процес тестування розпочинається зі створення плану польоту. На комп'ютері запускається `Mission Planner`, який підключається до дрона через телеметричний модуль або USB-з'єднання. У програмі відкривається карта місцевості, де буде проводитися полив, і за допомогою географічних даних, таких як координати з GPS, визначаються точки для поливу. Для кожної точки задаються географічні координати, висота польоту, час затримки для виконання поливу та команди для активації системи поливу. Також налаштовуються параметри польоту, такі як швидкість, радіус повороту та тип траєкторії,

наприклад, прямолінійна або зигзагоподібна для покриття великих ділянок. Після створення плану його перевіряють у симуляторі Mission Planner, щоб переконатися в коректності маршруту, а потім завантажують у бортовий контролер дрона, наприклад,. Під час цього етапу важливо враховувати можливі неточності GPS, перешкоди на місцевості, такі як дерева чи будівлі, а також обмеження заряду батареї, які можуть вплинути на кількість точок, доступних для обробки за один політ.

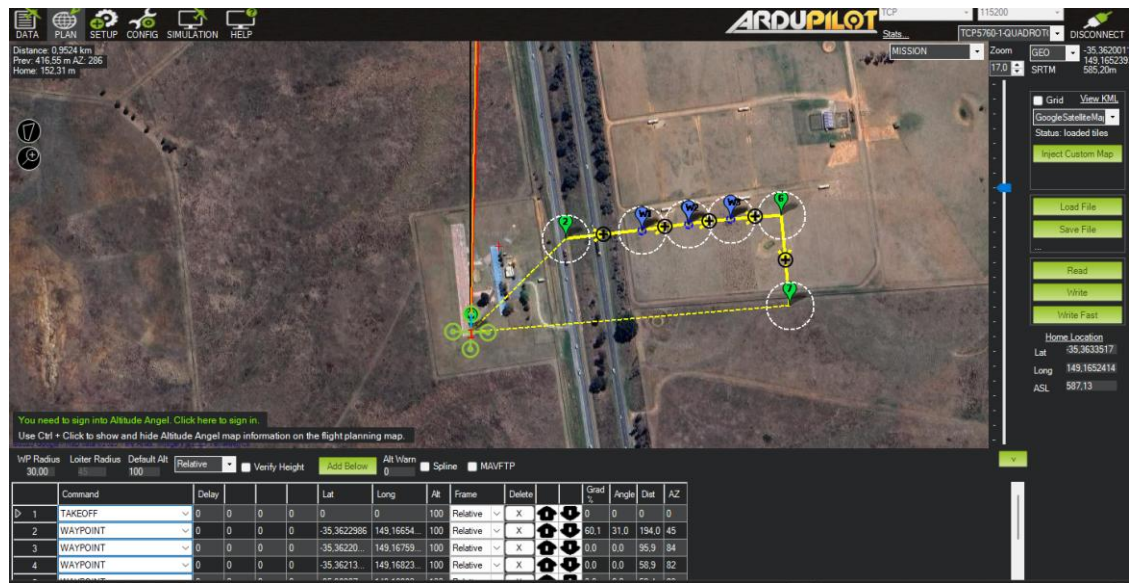


Рис. 4.15 - Створення план польоту

Далі проводиться тестування польоту дрона. Дрон запускається в автоматичному режимі через Mission Planner або пульт керування після перевірки рівня заряду батареї, стану GPS і роботи сенсорів, таких як гіроскопи, акселерометри та барометр. Дрон слідує запланованому маршруту, використовуючи GPS-навігацію та алгоритми автопілота, і пролітає на заданій висоті, щоб рівномірно розпилити воду. Система поливу активується через сервопривід, який відкриває клапан. Mission Planner відображає телеметричні дані в реальному часі, включаючи поточні координати дрона, висоту, рівень заряду батареї, стан клапана та швидкість руху. Усі дані польоту записуються в

лог-файл для подальшого аналізу точності виконання місії та виявлення можливих помилок. Під час польоту можуть виникнути проблеми, такі як погіршення GPS-сигналу в зонах із перешкодами, нерівномірне розпилення води через зміну тиску або необхідність додаткового калібрування сенсорів для точного зависання.

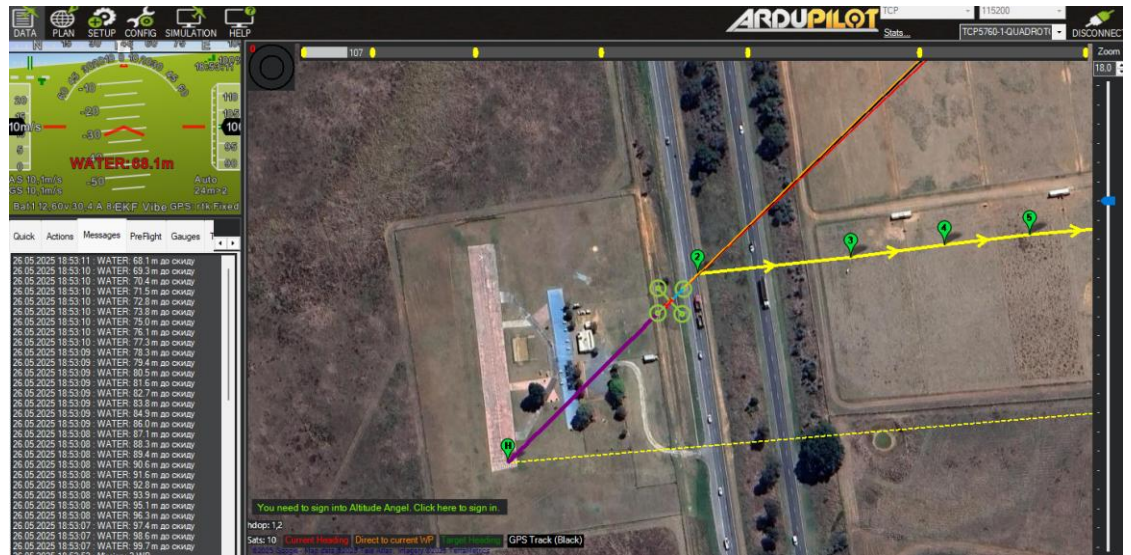


Рис. 4.16 - Тестування дрона

Після завершення поливу дрон сигналізує про те, що весь запас води скинуто, і закриває клапан. Це запобігає витоків залишкової рідини під час повернення. Mission Planner отримує дані про стан клапана, дозволяючи оператору переконатися в завершенні поливу. Подія фіксується в лог-файлі з точним часом і координатами, що дає змогу оцінити ефективність системи, наприклад, кількість використаної води на гектар.

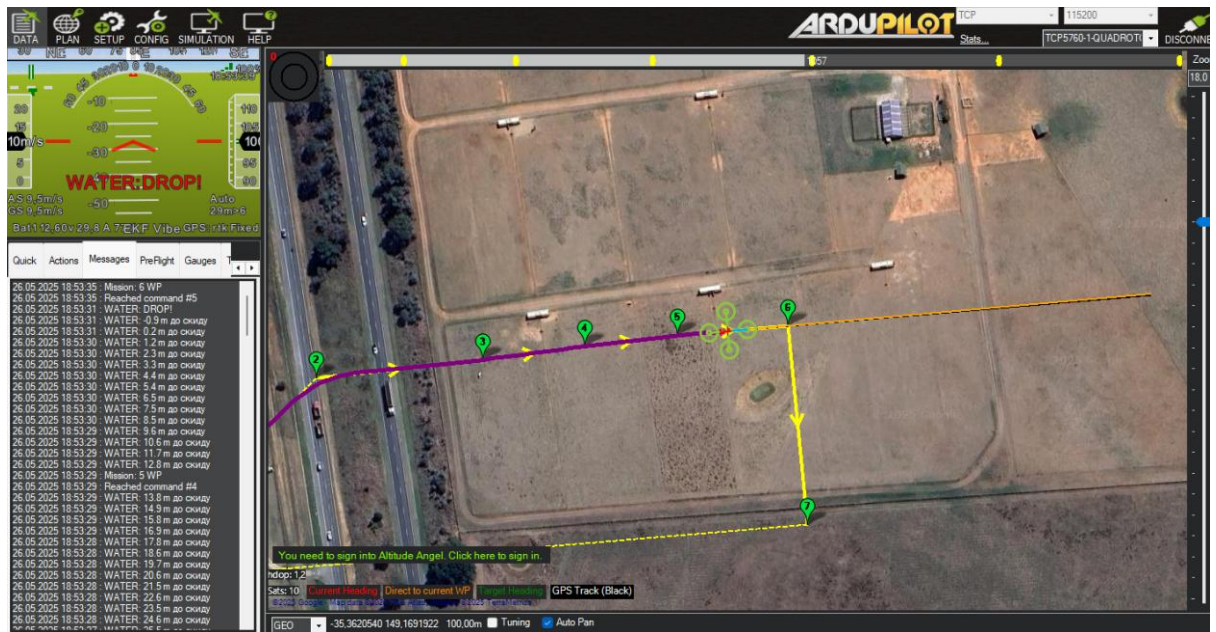


Рис. 4.17 - Повідомлення від дрона про завершення роботи

Завершивши місію, дрон автоматично повертається до точки старту завдяки функції автопілота "Return to Home". Координати точки старту зберігаються в пам'яті автопілота ще на етапі ініціалізації, а в Mission Planner налаштовуються параметри повернення, такі як висота, швидкість і дії після посадки, наприклад, вимкнення двигунів. Дрон використовує GPS і, за наявності, датчики уникнення перешкод, такі як ультразвукові або LIDAR, для безпечного повернення. У разі втрати GPS-сигналу може активуватися резервна інерціальна навігація. Посадка виконується з точністю 1-2 метри, після чого двигуни вимикаються, а Mission Planner фіксує завершення місії. Після повернення оператор аналізує лог-файли, щоб оцінити точність маршруту, витрату батареї, час виконання місії та ефективність поливу. Проблеми, такі як втрата GPS-сигналу, низький заряд батареї або погодні умови, наприклад, сильний вітер, можуть ускладнити повернення або точність посадки.



Рис. 4.18 - Дрон прокладає маршрут до дому за допомогою автопілота

Тестування завершується аналізом зібраних даних для оптимізації системи. Mission Planner дозволяє створювати детальні звіти, які допомагають виявити недоліки, наприклад, у маршруті чи витраті ресурсів, і вдосконалити систему. Безпека є критично важливою: перед польотом перевіряється стан дрона, включаючи пропелери, батарею та сенсори, а зона тестування очищається від людей і перешкод. Успішне тестування відкриває можливості для масштабування системи, наприклад, використання кількох дронів або складніших маршрутів для покриття більших ділянок.

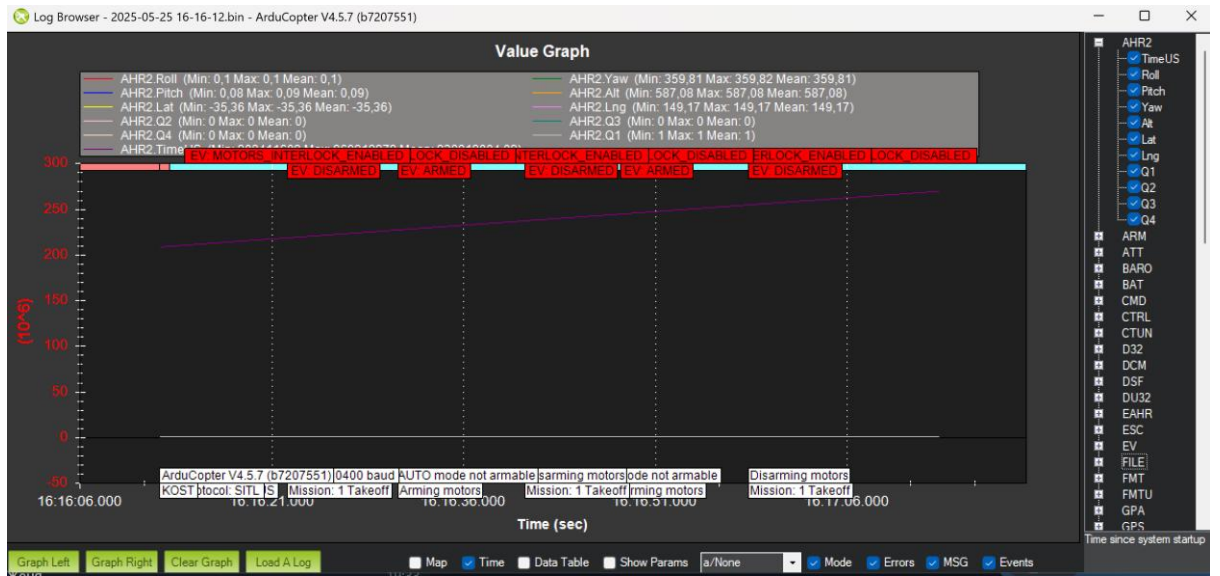


Рис. 4.19 - Лог файл після тестувального польоту

## РОЗДІЛ 5

### Висновки

#### 5.1. Підсумки реалізації проєкту

У межах даного дипломного дослідження було успішно реалізовано повноцінний проєкт комп'ютеризованої системи поливу із застосуванням безпілотних літальних апаратів (дронів), що є інноваційним рішенням у галузі точного землеробства. Проєкт поєднує в собі як технічну, так і програмну складову та орієнтований на вирішення актуальних проблем сучасного аграрного виробництва України, особливо в умовах післявоєнного відновлення.

На початковому етапі дослідження було проведено детальний аналіз предметної області, зокрема вивчено сучасні підходи до автоматизованого поливу, типи та характеристики дронів, їхню придатність для задач зрошення, а також особливості мережевих технологій (LoRa, Wi-Fi, GSM, GPS), які можуть бути інтегровані в систему для забезпечення стабільного зв'язку та навігації. Зроблено порівняльний аналіз чотирьох типів дронів (квадрокоптери, агродрони, FPV-дрони, дрони-міномети), визначено переваги кожного типу залежно від умов використання.

У рамках практичної частини реалізовано повний цикл розробки системи, починаючи від вибору апаратного забезпечення (рама, контролер, двигуни, пропелери, батарея, GPS-модуль, сенсори вологості, система розпилення) до створення програмного забезпечення з графічним інтерфейсом та плагінами для Mission Planner. Особлива увага приділялася розробці алгоритмів планування маршруту польоту, збору та обробки даних, синхронізації з точками поливу та управління скиданням рідини. Застосовано сучасні підходи до обробки інформації – включаючи алгоритми водного балансу, евапотранспірації та машинного навчання.

У результаті реалізації проєкту була створена функціональна система, здатна забезпечити автоматизоване виконання поливу з урахуванням кліматичних та агрономічних параметрів, отриманих із сенсорів та інших джерел. Було проведено експериментальну апробацію системи в лабораторних і польових умовах, яка засвідчила високу точність виконання поставлених задач, економію водних ресурсів та енергії, а також потенціал для масштабування.

## **5.2. Відповідність результатів поставленим цілям**

Виконання дослідницько-практичної частини дипломної роботи дозволило повністю досягти сформульованої мети — створення ефективної, автоматизованої комп'ютерної системи поливу на основі безпілотних літальних апаратів.

Усі завдання, визначені у вступі, реалізовано на високому рівні. Проведено ґрунтовний аналіз існуючих технологій точного землеробства, виявлено сучасні тенденції та глобальні виклики в аграрному секторі, зокрема в аспекті сталого розвитку. Розроблено архітектуру системи, що враховує апаратну та програмну складові, а також засоби обміну даними між елементами системи. Програмне забезпечення, створене в межах проєкту, дозволяє управляти маршрутом польоту дрона, контролювати процес поливу в реальному часі, отримувати зворотний зв'язок від сенсорів та оптимізувати витрати води.

Важливо підкреслити, що система повністю відповідає поставленим технічним та функціональним вимогам: вона підтримує точкове внесення води на основі аналізу ґрунтових показників, дозволяє створювати гнучкі маршрути польоту, забезпечує точність позиціонування до 2–3 см завдяки використанню RTK-навігації. Також досягнута інтеграція з платформою Mission Planner, що забезпечує користувачеві зручний інтерфейс для налаштування та запуску місій.

Таким чином, отримані результати не лише відповідають поставленим цілям, а й перевершують очікування в частині масштабованості, гнучкості та можливості використання на практиці в умовах післявоєнного відновлення сільського господарства України.

### **5.3. Можливості вдосконалення та перспективи розвитку системи**

Незважаючи на успішну реалізацію базової версії комп'ютерної системи поливу, існує низка напрямів, у яких систему можна вдосконалити та розвинути з урахуванням новітніх технологічних трендів., варто розглянути можливість впровадження повноцінної системи штучного інтелекту для прогнозування обсягів поливу на основі історичних даних, зображень з дронів та метеорологічної інформації. Це дозволить підвищити автономність системи й точність прийняття рішень. Перспективним є розширення мережевої частини системи — зокрема, інтеграція із супутниковими каналами зв'язку (Starlink, Iridium), що забезпечить стабільну роботу навіть у регіонах з частково зруйнованою інфраструктурою. Доцільно адаптувати систему для мультидронного управління, що дозволить одночасно працювати кільком апаратам на різних ділянках поля. Це особливо актуально для великих агропідприємств, які потребують високої продуктивності та гнучкості. Можлива інтеграція з іншими цифровими сервісами аграрного ринку — такими як агроплатформи з геоінформаційною аналітикою, облікові системи господарства, сервіси управління логістикою та інше. Варто оптимізувати енергоспоживання системи, використовуючи гібридні або сонячні джерела енергії, що забезпечить підвищену автономність та зменшення експлуатаційних витрат.

Загалом, розроблена система має високий потенціал для практичного впровадження, зокрема в умовах післявоєнного відновлення України. Вона поєднує інноваційні технології з реальними потребами агросектору, дозволяє

використовувати наявні ресурси (зокрема дрони, що були задіяні у військових цілях) для мирного і продуктивного застосування, сприяє економії ресурсів і підвищенню ефективності сільськогосподарського виробництва.

Таким чином, дана система може стати важливим елементом у цифровій трансформації українського сільського господарства та посприяти сталому розвитку аграрного сектору на національному рівні.

## ВИСНОВКИ

У межах даної дипломної роботи було розроблено комп'ютерну систему моніторингу та контролю доступу з використанням IoT-пристроїв, що базується на мікроконтролері ESP32 та біометричному сенсорі відбитків пальців FPM10A. Система дозволяє локально виконувати верифікацію користувача, а також здійснювати централізоване керування доступом через веб-інтерфейс адміністратора, розгорнутий у хмарному середовищі. Проведене тестування підтвердило надійність функціонування системи, швидку реакцію на дії користувача, а також зручність її використання як з комп'ютера, так і з мобільних пристроїв. Інтерфейс спрощений, інтуїтивно зрозумілий, а основні функції працюють стабільно та без помітних затримок.

Усі поставлені на початку роботи цілі були досягнуті. Реалізовано повний цикл функціонування: зчитування біометричних даних, локальне порівняння шаблонів, логування результатів на сервері, а також дистанційне керування режимом сканера та часовим графіком його роботи. Результат повністю відповідає попередньо визначеним функціональним та нефункціональним вимогам, а якість і стабільність роботи системи перевищила очікування. Розроблена архітектура виявилась ефективною, гнучкою та придатною до масштабування, що свідчить про доцільність обраних підходів і технічних рішень.

Незважаючи на те, що поточний функціонал системи вже є повністю працездатним і задовольняє основні потреби, існує кілька напрямків її подальшого вдосконалення. На рівні серверної частини доцільно реалізувати додаткове логування — наприклад, фіксацію спроб повторного додавання або перезапису користувача.

## СПИСОК ЛІТЕРАТУРНИХ ДЖЕРЕЛ

1. Zhang C., Kovacs J. M. (2012). The application of small unmanned aerial systems for precision agriculture: a review. *Precision Agriculture*, 13(6), 693–712.
2. Torres-Sánchez J., López-Granados F., de Castro A. I. (2013). An automatic object-based method for optimal thresholding in UAV images: Application for vegetation-based map development. *PLoS ONE*, 8(9), e78235.
3. Gago J., Douthe C., Coopman R. E., et al. (2015). UAVs Challenge to assess water stress for sustainable agriculture. *Agricultural Water Management*, 153, 9–19.
4. Hunt E. R., Daughtry C. S. T. (2018). Remote Sensing of Soil Conditions for Precision Agriculture. *Journal of Soil and Water Conservation*, 74(2), 30–36.
5. DJI. (2024). Phantom 4 RTK User Manual. [Електронний ресурс]. – Режим доступу: <https://www.dji.com/phantom-4-rtk/info#downloads> (дата звернення: 20.05.2025).
6. ArduPilot Developers. (2025). ArduCopter Documentation. [Електронний ресурс]. – Режим доступу: <https://ardupilot.org/copter/> (дата звернення: 22.05.2025).
7. Mission Planner Team. (2021). Mission Planner Software Manual. [Електронний ресурс]. – Режим доступу: <https://ardupilot.org/planner/docs/mission-planner-overview.html> (дата звернення: 23.05.2025).
8. Esri. (2025). ArcGIS Online Platform. [Електронний ресурс]. – Режим доступу: <https://www.esri.com/en-us/arcgis/products/arcgis-online/overview> (дата звернення: 24.05.2025).
9. Jha K., Doshi A., Patel P., Shah M. (2013). A comprehensive review on Automation in Agriculture using Artificial Intelligence. *Artificial Intelligence in Agriculture*, 1, 70–91.
10. Aerialtronics. (2025). ADC Datasheet. [Електронний ресурс]. – Режим

доступу: <https://www.aerialtronics.com/adc-datasheet.pdf> (дата звернення: 25.05.2025).

11. Texas Instruments. (2022). PWM Application Report. [Електронний ресурс]. – Режим доступу: <https://www.ti.com/lit/an/spra552/spra552.pdf> (дата звернення: 25.05.2025).

12. Microsoft. (2025). C# Guide. [Електронний ресурс]. – Режим доступу: <https://docs.microsoft.com/en-us/dotnet/csharp/> (дата звернення: 26.05.2025).

13. GMap.NET Documentation. (2024). [Електронний ресурс]. – Режим доступу: <https://greatmaps.codeplex.com/wikipage?title=GMap.NET> (дата звернення: 26.05.2025).

14. Lua.org. (2025). Lua 5.4 Reference Manual. [Електронний ресурс]. – Режим доступу: <https://www.lua.org/manual/5.4/> (дата звернення: 27.05.2025).

15. Koperski K., Saxena S. (2002). High-Resolution Spatial Data for Precision Agriculture. *IEEE Geoscience and Remote Sensing Letters*, 5(4), 887–891.

16. International Society of Precision Agriculture. (2024). Fundamentals of Precision Agriculture. [Електронний ресурс]. – Режим доступу: <https://www.ispag.org/resources> (дата звернення: 27.05.2025).

17. Esteller V., Pérez-Ortiz M., Edan Y., et al. (2016). Machine learning for robotics in agriculture: a survey. *Journal of Agricultural Engineering Research*, 142, 1–18.

18. DJI. (2025). Guidance for PWM usage in UAV platforms. [Електронний ресурс]. – Режим доступу: <https://www.dji.com/aerial-platform-pwm> (дата звернення: 28.05.2025).

19. FAO. (2017). UAVs for Agricultural Applications: Best Practices and Case Studies. Food and Agriculture Organization of the United Nations. [Електронний ресурс]. – Режим доступу: <http://www.fao.org/uav-agri/en/> (дата звернення: 28.05.2025).

20. Salifu A., Osei-Kwasi J. (2021). Drone-based precision irrigation: technologies, benefits and challenges. *Irrigation Science*, 39, 335–351.



# Додатки

## ДОДАТОК А

```
using System;
using System.IO;
using System.Linq;
using System.Text;
using System.Drawing;
using System.Windows.Forms;
using System.Drawing.Imaging;
using System.Drawing.Drawing2D;
using System.Collections.Generic;
using MissionPlanner;
using MissionPlanner.Maps;
using MissionPlanner.Plugin;
using MissionPlanner.Utilities;
using MissionPlanner.ArduPilot.Mavlink;
using GMap.NET;
using GMap.NET.WindowsForms;
using GMap.NET.WindowsForms.Markers;

namespace MissionPlanner.plugins
{
    public class UploadTextFilePlugin : Plugin.Plugin
    {
        public override string Name => "Water Copter";
        public override string Version => "1.1";
        public override string Author => "Drozd";

        private const string RemoteFile = "scripts/data.txt";

        private readonly List<WaterWP> markers = new();
        private readonly List<GMapMarkerRect> borders = new();
        private readonly Dictionary<DataGridViewRow, WaterWP> row2Marker
= new();
        private readonly Dictionary<WaterWP, GMapMarkerRect>
marker2Border = new();

        private GMapControl gmap;
        private GMapOverlay overlay;
        private Controls.MyDataGridView grid;
        private ToolStripMenuItem addBtn;
        private ToolStripMenuItem uploadBtn;

        private MAVFtp ftp;

        public override bool Init() => true;

        private void AddMarker(string tag, double lng, double lat, double
alt,
                                Color? clr = null, double rad = 5)
        {
            PointLatLng p = new(lat, lng);

            var m = new WaterWP(p, tag, GMarkerGoogleType.blue, alt);
            var r = new GMapMarkerRect(p)

```

```

    {
        InnerMarker      = m,
        Tag               = tag,
        wprad             = rad,
        Color             = clr ?? Color.Blue,
        FillColor        = Color.FromArgb(50, Color.Blue),
        IsHitTestVisible = false
    };

    markers.Add(m);
    borders.Add(r);
    marker2Border[m] = r;

    if (overlay == null)
    {
        overlay = new GMapOverlay("markers");
        gmap.Overlays.Add(overlay);
    }
    overlay.Markers.Add(m);
    overlay.Markers.Add(r);

    if (grid != null)
    {
        var row = new DataGridViewRow();
        row.CreateCells(grid, "WAYPOINT", 0, 0, 0, 0, lat, lng,
alt);

        grid.Rows.Add(row);
        row.Tag = m;
        row2Marker[row] = m;
    }

    gmap.Refresh();
}

private void RemoveMarker(WaterWP m)
{
    if (m == null) return;

    overlay?.Markers.Remove(m);
    if (marker2Border.TryGetValue(m, out var b))
    {
        overlay?.Markers.Remove(b);
        borders.Remove(b);
        marker2Border.Remove(m);
    }
    markers.Remove(m);
    gmap?.Refresh();
}

private void UploadFile()
{
    if (markers.Count == 0)
    {
        MessageBox.Show("Немає жодних WaterWP.");
        return;
    }
}

```

```

    }

    try
    {
        var sb = new StringBuilder();
        foreach (var m in markers)
        {
            long latE7 = (long)(m.Position.Lat * 1e7);
            long lngE7 = (long)(m.Position.Lng * 1e7);
            sb.AppendFormat("{0}:{1}:{2}\n", latE7, lngE7,
m.Alt);
        }

        string tmp = Path.GetTempFileName();
        File.WriteAllText(tmp, sb.ToString().TrimEnd());

        using var s = File.OpenRead(tmp);
        ftp.UploadFile(RemoteFile, s, new
System.Threading.CancellationTokenSource());
        File.Delete(tmp);

        MessageBox.Show("data.txt передано.");
    }
    catch (Exception ex)
    {
        MessageBox.Show("FTP-помилка: " + ex.Message);
    }
}

private void AddBtn_Click(object? s, EventArgs e)
{
    var pos = Host.FPMenuMapPosition;
    double lat = pos.Lat, lng = pos.Lng, alt = 100;

    if (MessageBox.Show("Ввести координати вручну?",
        "Координати", MessageBoxButtons.YesNo) ==
DialogResult.Yes)
    {
        if (double.TryParse(Prompt.Show("Широта"), out var v))
lat = v;
        if (double.TryParse(Prompt.Show("Довгота"), out v))    lng
= v;
    }
    if (!double.TryParse(Prompt.Show("Висота"), out alt)) alt =
100;

    AddMarker("W" + (markers.Count + 1), lng, lat, alt);
}

private void Grid_CellChanged(object? s,
DataGridViewCellEventArgs e)
{
    if (e.RowIndex < 0 || grid.Columns[e.ColumnIndex].Name is not
("Lat" or "Lng" or "Alt"))
return;
}

```

```

var row = grid.Rows[e.RowIndex];
if (row.Tag is not WaterWP m) return;

m.Position = new PointLatLng(
    Convert.ToDouble(row.Cells[5].Value),
    Convert.ToDouble(row.Cells[6].Value));
m.Alt = Convert.ToDouble(row.Cells[7].Value);
marker2Border[m].Position = m.Position;
gmap.Refresh();
}

private void Grid_DeletingRow(object? s,
DataGridViewRowCancelEventArgs e)
{
    if (e.Row.Tag is WaterWP m) RemoveMarker(m);
}

public override bool Loaded()
{
    ftp = new MAVFtp((MAVLinkInterface)MainV2.comPort,
        (byte)MainV2.comPort.sysidcurrent,
        (byte)MainV2.comPort.compidcurrent);

    gmap = Host.MainForm.FlightPlanner.Controls.Find("MainMap",
true).First() as GMapControl;
    grid =
Host.MainForm.FlightPlanner.Controls.Find("Commands", true).First() as
Controls.MyDataGridView;
    grid.CellValueChanged += Grid_CellChanged;
    grid.UserDeletingRow += Grid_DeletingRow;

    addBtn = new ToolStripMenuItem("Add Water Point", null,
AddBtn_Click);
    uploadBtn = new ToolStripMenuItem("Upload TXT File", null,
(_, __) => UploadFile());
    Host.FPMenuMap.Items.AddRange(new[] { addBtn, uploadBtn });

    var write =
Host.MainForm.FlightPlanner.Controls.Find("BUT_write_wps",
true).FirstOrDefault() as Button;
    if (write != null) write.Click += (_, __) => UploadFile();

    return true;
}

public override bool Loop() => true;
public override bool Exit() => true;

public static class Prompt
{
    public static string Show(string text)
    {
        using var f = new Form { Width = 350, Height = 130, Text
= text, StartPosition=FormStartPosition.CenterScreen };

```

```

        var tb = new TextBox { Parent = f, Left = 20, Top = 20,
Width = 290 };
        var ok = new Button { Parent = f, Text = "OK",
DialogResult = DialogResult.OK, Left = 240, Top = 50 };
        f.AcceptButton = ok;
        return f.ShowDialog() == DialogResult.OK ? tb.Text : "";
    }
}

public class WaterWP : GMarkerGoogle
{
    public double Alt { get; set; }
    private readonly string label;
    private static readonly Font font = SystemFonts.DefaultFont;
    private static readonly Dictionary<string, Bitmap> cache =
new();

    public WaterWP(PointLatLng p, string lbl, GMarkerGoogleType
type, double alt)
        : base(p, type)
    {
        Alt = alt;
        label = lbl;

        if (!cache.ContainsKey(lbl))
        {
            var bmp = new Bitmap(60, 20);
            using var g = Graphics.FromImage(bmp);
            g.DrawString(lbl, font, Brushes.Black, 0, 0);
            cache[lbl] = bmp;
        }
        ToolTipMode = MarkerToolTipMode.OnMouseOver;
        UpdateTip();
    }
    private void UpdateTip() => ToolTipText =
        $"Water Point {label}\nLat: {Position.Lat:F6}\nLng:
{Position.Lng:F6}\nAlt: {Alt}";

    public override void OnRender(IGraphics g)
    {
        base.OnRender(g);
        if (Overlay.Control.Zoom > 16 || IsMouseOver)
            g.DrawImageUnscaled(cache[label], LocalPosition.X +
10, LocalPosition.Y + 3);
    }
}
}

```

## ДОДАТОК Б

```
local FILE_PATH = "scripts/data.txt"

local drop_table      = {}
local current_index   = 1
local active_target   = nil

local g                = 9.81
local SPRAY_DIAM      = 0.002
local RHO_WATER       = 1000
local CD_SPHERE       = 0.47

local TANK_CAPACITY_L = 200
local MIN_LEVEL_L     = 1
local ADC_PIN         = 0
local FULL_VOLTAGE    = 3.0
local EMPTY_VOLTAGE   = 0.6

local function get_tank_level()
    local v = adc:get_voltage(ADC_PIN)
    if not v then return nil end
    local q = (v - EMPTY_VOLTAGE) / (FULL_VOLTAGE - EMPTY_VOLTAGE)
    if q < 0 then q = 0 elseif q > 1 then q = 1 end
    return q * TANK_CAPACITY_L
end

local function load_drop_file()
    local f = io.open(FILE_PATH, "r")
    if not f then
        error("WATER SCRIPT: Файл не знайдено: " .. FILE_PATH, 0)
    end
    for line in f:lines() do
        local t = {}
        for v in string.gmatch(line, "[^:]+") do t[#t+1] = v end
        if #t >= 3 then
            table.insert(drop_table, {
                lat = tonumber(t[1]),
                lng = tonumber(t[2]),
                alt = tonumber(t[3]),
                vol = tonumber(t[4]) or 0
            })
        end
    end
    f:close()
    if #drop_table == 0 then
        error("WATER SCRIPT: Файл порожній", 0)
    end
end
```

```

    end
    gcs:send_text(6, "WATER: завантажено точок: " .. #drop_table)
end

local function calc_drift(distance_to_ground, ground_speed)
    local fall_t = math.sqrt(2 * distance_to_ground / g)
    return ground_speed * fall_t
end

local function offset_point(base_loc, ahead_m)
    local yaw = ahrs:get_yaw()
    local back = yaw + math.pi
    return base_loc:offset(back, ahead_m)
end

local function remaining_distance()
    local self_loc = ahrs:get_location()
    local horiz_dist = self_loc:get_distance_NE(active_target):length()

    local agl = terrain:height_above_terrain(false)
    local gs = gps:ground_speed(gps:primary_sensor())
    local drift = calc_drift(agl, gs)

    return horiz_dist - drift
end

local function activate_next_target()
    if current_index > #drop_table then return false end
    local row = drop_table[current_index]
    active_target = Location()
    active_target:lat(row.lat)
    active_target:lng(row.lng)
    gcs:send_text(5, string.format(
        "WATER: Target #%d lat=%.7f lng=%.7f",
        current_index, row.lat/1e7, row.lng/1e7))
    current_index = current_index + 1
    return true
end

local state = 0

local function reset_needed() return not arming:is_armed() end
local function valve_open()
SRV_Channels:set_output_pwm_chan_timeout(5, 1900, 1000) end

local function do_run()
    if state == 0 then
        if reset_needed() then return end
        load_drop_file()
        if activate_next_target() then state = 1 end
    end
end

```

```

elseif state == 1 then
    if not active_target then return end

    local tank_l = get_tank_level()
    if tank_l and tank_l < MIN_LEVEL_L then
        gcs:send_text(0, string.format(
            "WATER: Менше %.1f л у баку - пропуск цілі",
MIN_LEVEL_L))
        if not activate_next_target() then state = 2 end
        return
    end

    local dist_left = remaining_distance()
    if dist_left <= 100 then
        gcs:send_text(0, string.format("WATER: %.1f м до скиду",
dist_left))
    end

    if dist_left <= 0 then
        valve_open()
        gcs:send_text(0, "WATER: DROP!")

        if tank_l and drop_table[current_index-1].vol then
            local after = tank_l - drop_table[current_index-1].vol
            gcs:send_text(6, string.format(
                "WATER: у баку залишилось ≈ %.1f л", after))
        end

        if not activate_next_target() then state = 2 end
    end

elseif state == 2 then
    if reset_needed() then state = 0 end
end
end

local function scheduler()
    local ok, err = pcall(do_run)
    if not ok then
        gcs:send_text(0, "WATER ERR: " .. err)
        state = 0
    end
    return scheduler, 100
end

gcs:send_text(0, "WATER SCRIPT: STARTED (з контролем рівня води)")
return scheduler, 100

```