

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ  
Факультет кібербезпеки, комп'ютерної та програмної інженерії  
Кафедра інженерії програмного забезпечення

ДОПУСТИТИ ДО ЗАХИСТУ

Завідувач кафедри

Горський О.М.

« \_\_\_ » \_\_\_\_\_ 202 р.

**ДИПЛОМНА РОБОТА  
(ПОЯСНЮВАЛЬНА ЗАПИСКА)  
ВИПУСКНИКА ОСВІТЬОГО СТУПЕНЯ  
МАГІСТРА**

**Тема:** «Методика та застосунок системи підвищення життєдіяльності безпілотних апаратів»

**Виконавець:** Крючок Дмитро Вікторович

**Керівник:** Горський О.М.

**Нормоконтролер:** Трофимчук Вікторія Миколаївна

Київ 2023

**НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ****Факультет кібербезпеки, комп'ютерної та програмної інженерії****Кафедра інженерії програмного забезпечення****Освітній ступінь магістр****Форма навчання: заочна****Спеціальність 121 Інженерія програмного забезпечення****Освітньо-професійна програма «Інженерія програмного забезпечення»****ЗАТВЕРДЖУЮ**

Завідувач кафедри

Горський О.М.

«\_\_\_\_\_» \_\_\_\_\_ 2023

р.

**ЗАВДАННЯ**

на виконання дипломної роботи студента

Крючок Дмитра Вікторовича

1. Тема дипломної роботи: «Методика та застосунок системи підвищення життєдіяльності безпілотних апаратів», затверджена наказом ректора від 04.10.2023 р. № 2034/ст.
2. Термін виконання проекту: з 30.10.2023 р. до 08.12.2023 р.
3. Вихідні дані до роботи: розробити методику дослідження предметної області та створити на її основі прототип модуля згідно поставленої задачі
4. Зміст пояснювальної записки

## КАЛЕНДАРНИЙ ПЛАН-ГРАФІК ПРАКТИКИ

№ пор.	Завдання	Термін виконання	Відмітка про виконання
1.	Ознайомлення з поставленою задачею. Затвердження графіку роботи	19.09.22- 23.09.22	
2.	Підготовка та написання 1 розділу	23.09.22- 30.09.22	
3.	Представлення 1 розділу керівнику. Підготовка та написання 2 розділу	30.09.22- 07.10.22	
4.	Представлення 2 розділу керівнику Підготовка та написання 3 розділу	08.10.22- 16.10.22	
5.	Представлення 3 розділу керівнику та загальне редагування роботи	16.10.22- 24.10.22	
6.	Форматування тексту пояснювальної записки та графічного матеріалу	24.10.22- 03.11.22	
7	Проходження контролю на плагіат. Отримання відгуку керівника	24.10.22- 03.11.22	
8.	Проходження нормо-контролю, перепліт пояснювальної записки. Підготовка презентації та тексту доповіді.	01.11.22- 03.11.22	
9.	Допуск до захисту (підпис зав.каф.).	04.11.22	
10.	Отримання рецензії	04.11.22- 11.11.22	

12.	Здати секретарю ДЕК: ПЗ, ГМ, CD-R з електронними версіями ПЗ, ГМ, презентацію, відгук керівника, рецензію, довідку про успішність, 2 папки, 2 конверта	напередодні захисту	
13.	Захист дипломної роботи перед ЕК	26.11.22- 28.11.22	

Керівник кваліфікаційної роботи:

Горський О.М.

(підпис)

Виконав: \_\_\_\_\_ Крючок Дмитро Вікторович  
(підпис)

## РЕФЕРАТ

Пояснювальна записка до дипломної роботи «**Методика та застосунок системи підвищення життєдіяльності безпілотних апаратів**»: 105 сторінок, 54 рисунків, 2 таблиці, 19 використаних джерел, х додаток.

**МЕТОДИКА ДОСЛІДЖЕНЬ, МІНІМАЛЬНО ЖИТТЄЗДАТНЕ ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ, МАКЕТ, АНАЛІТИКА.**

**Об'єкт дослідження** – безпілотні літальні апарати, їх характеристики

**Предмет дослідження** – методи та технології проведення досліджень та реалізації автоматизованих систем.

**Мета дипломної роботи** – розробка методики дослідження та розробка на її основі програмної та апаратної частини.

**Метод дослідження** – використання вербальних і графічних методів для моделювання та глибоке дослідження основ та процесів предметної області. В роботі проведено дослідження предметної області, створено моделі на різних етапах проектування, визначено актуальні ІТ-потреби у зазначеній темі.

**Методологічною основою** є теорія та практика розробки модуля для підвищення життєдіяльності безпілотних літальних апаратів

**Результати** роботи можуть використовуватися на практиці в різних сферах використання безпілотних літальних апаратів.

## ABSTRACT

Explanatory note for the thesis “**Methodology and shutdown of the life-sustaining system of unmanned vehicles**”: 105 pages, 54 figures, 2 tables, 19 victorious parts, x appendix.

METHODS OF FOLLOWING, MINIMUM LIFE OF DATA SECURITY PROGRAM, LAYOUT, ANALYTICS.

**Object of investigation** – unmanned aerial vehicles, their characteristics

**Subject of investigation** – methods and technologies of investigation implementation of automated systems.

**Meta of the thesis work** – development of research methods and development on the basis of software and hardware.

**The investigation method** is a combination of verbal and graphic methods for modeling and in-depth investigation of the fundamentals and processes of the subject area. In the robot, research of the subject area was carried out, a model was created on. At various stages of design, the current IT needs of the assigned topic were identified.

**The methodological** basis is the theory and practice of developing a module to improve the life of unmanned aircraft

**The results** of the work can be used in practice in various areas of the development of unmanned aerial vehicles.

## ЗМІСТ

<b>ВСТУП .....</b>	<b>9</b>
<b>РОЗДІЛ 1.....</b>	<b>11</b>
<b>АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ БЕЗПІЛОТНОГО ЛІТАЛЬНОГО АПАРАТУ .....</b>	<b>11</b>
<b>1.1. Аналіз предметної області програмного та апаратного забезпечення безпілотного літального апарату, методи їх пошуку та автоматичної системи автопілоту .....</b>	<b>11</b>
<b>2.1. Аналіз існуючих аналогів автопілоту та пошуку у відкритому просторі .....</b>	<b>14</b>
<b>2.1. Постановка задачі .....</b>	<b>15</b>
<b>Висновки .....</b>	<b>17</b>
<b>РОЗДІЛ 2 АНАЛІЗ ІСНУЮЧИХ СИСТЕМ РАДІО-ЕЛЕКТРОННОЇ БОРОТЬБИ. РОЗРОБКА БАЗОВИХ КОМПОНЕНТІВ МЕТОДИКИ ПОБУДОВИ МОДУЛЕЙ ДЛЯ ПРОТИСТОЯННЯ РЕБ .....</b>	<b>19</b>
<b>2.1. Принцип роботи зв'язку БЛА, види та методи їх управління.....</b>	<b>19</b>
<b>2.1.1. Зв'язок БЛА.....</b>	<b>19</b>
<b>2.1.2. Види БЛА.....</b>	<b>22</b>
<b>2.2. Аналіз існуючих систем РЕБ. Розгляд методики роботи цих систем та вплив на БЛА .....</b>	<b>25</b>
<b>2.3. Формування моделі роботи рятувального модуля .....</b>	<b>29</b>
<b>2.3.1. Модуль візуального виявлення .....</b>	<b>34</b>
<b>2.3.2. Модуль радіо-маячок.....</b>	<b>36</b>
<b>2.3.3. Модуль автопілот .....</b>	<b>41</b>
<b>Висновки .....</b>	<b>43</b>
<b>РОЗДІЛ 3 РОЗРОБКА БАЗОВИХ КОМПОНЕНТІВ МЕТОДИКИ ПОБУДОВИ МОДУЛЕЙ ДЛЯ ПІДВИЩЕННЯ ЖИТТЄДІЯЛЬНОСТІ БЛА ТА ПРОТИСТОЯННЯ ЗОВНІШНІМ ФАКТОРАМ.....</b>	<b>44</b>
<b>3.1. Розробка модуля візуального виявлення .....</b>	<b>44</b>
<b>3.2. Розробка модуля радіо-маячка .....</b>	<b>62</b>
<b>3.3. Розробка модуля автопілота .....</b>	<b>76</b>
<b>Висновки .....</b>	<b>102</b>

## **ПЕРЕЛІК ПРИЙНЯТИХ СКОРОЧЕН**

ПЗ – програмне забезпечення.

ІТ – інформаційно-технологічні.

ПО – предметна область.

БЛА – безпілотний літальний апарат.

РЕБ – радіо електронна боротьба.

ІВП – інерційно вимірювальні пристрої.

LoRa (англ. Long Range) – технологія передачі зв'язку на відстані.

GPS (англ. Global Position System) – супутникова система навігації.



## ВСТУП

У сучасному світі розробка безпілотних літальних апаратів (БПЛА) здобуває все більшу популярність, особливо в галузі військової, рятувальної та промислової сфер. З появою цих технологій зростає також і потреба у вдосконаленні їх життєдіяльності та стійкості до різних впливів, таких як радіоелектронна боротьба (РЕБ) та негативні атмосферні умови.

**Метою дослідження** є розробка методології підвищення життєдіяльності та ефективності безпілотних літальних апаратів в умовах РЕБ та екстремальних погодних умов. Дослідження охоплює створення трьох ключових модулів, спрямованих на покращення функціональності та безпеки БПЛА.

Для досягнення поставленої мети розробки методології підвищення життєдіяльності БПЛА в умовах РЕБ та екстремальних погодних умов, передбачено вирішення ряду ключових завдань:

1. **Багатосторонній аналіз умов та загроз:** Ретельний розгляд усіх можливих впливів, які можуть виникнути в умовах радіоелектронної боротьби та екстремальних погодних умов.
2. **Створення моделі візуального виявлення:** Розробка та імплементація системи візуального аналізу для реагування на навколишнє середовище, виявлення загроз та прийняття швидких та ефективних рішень.
3. **Розробка системи радіо маячків:** Створення надійної системи зв'язку, яка забезпечить стійкість та неперервність управління БПЛА, навіть в умовах інтерференції РЕБ.
4. **Автопілот для автоматичного повернення:** Розробка автономного автопілота, який забезпечить повернення БПЛА в точку зльоту, якщо втрачено зв'язок або зіткнення з непередбаченими обставинами.

5. **Технологічний вибір для реалізації:** Визначення оптимальних технологій та засобів для реалізації розроблених модулів, забезпечення їх ефективності та сумісності.
6. **Створення прототипу та тестування:** Розробка базового прототипу системи, який включає всі розроблені модулі, та подальше проведення випробувань в умовах, які найкраще відображають реальні сценарії використання.
7. **Формування рекомендацій:** Створення конкретних порад та рекомендацій для подальшої розробки та вдосконалення методології в контексті підвищення ефективності БПЛА в умовах РЕБ та екстремальних погодних умов.

Ці завдання спрямовані на комплексний підхід до розробки та вдосконалення безпілотних літальних апаратів, забезпечуючи їх надійність та функціональність в умовах викликів, що можуть виникнути в реальних сценаріях використання.

**Об'єктом дослідження** є безпілотні літальні апарати (БПЛА) та їхні характеристики, функціональність і ефективність в умовах радіоелектронної боротьби (РЕБ) та екстремальних погодних умов.

**Новизна** даної розробки полягає в тому, що вона пропонує модульний підхід до підвищення життєдіяльності безпілотних літальних апаратів (БПЛА) в умовах радіоелектронної боротьби та екстремальних погодних умов, де стандартні методи виявляються неефективними. Дослідження передбачає розробку та імплементацію трьох ключових модулів: візуального виявлення, системи радіо маячків та автопілота для автоматичного повернення в точку зльоту.

## РОЗДІЛ 1

### АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ БЕЗПІЛОТНОГО ЛІТАЛЬНОГО АПАРАТУ

#### 1.1. Аналіз предметної області програмного та апаратного забезпечення безпілотного літального апарату, методи їх пошуку та автоматичної системи автопілоту

Аналіз предметної області програмного та апаратного забезпечення безпілотних літальних апаратів (БЛА) є важливою частиною дослідження, пов'язаного з розробкою та вдосконаленням технологій. На меті я маю розробити додатковий модуль для візуального або радіо пошуку втраченого БЛА внаслідок роботи РЕБ або з інших технічних причин. Давайте подивимося на ці аспекти більш детально:

##### 1. Аналіз предметної області:

- Цей аналіз включає в себе докладне вивчення всіх аспектів БЛА, включаючи їхню конструкцію, функції та додаткове обладнання.
- Важливо визначити, які завдання вирішує БЛА, які галузі використання (наприклад, військові, геологічні, агрокультура) існують, і які є основні вимоги до програмного та апаратного забезпечення.

##### 2. Методи пошуку:

- В цьому контексті методи пошуку включають в себе алгоритми, які допомагають БЛА визначати своє місцезнаходження і навігувати в просторі.
- Геолокаційні системи, такі як GPS, GLONASS, або внутрішні сенсори, можуть використовуватися для цієї мети.
- Також важливо дослідити методи обробки та інтерпретації отриманих даних для точної навігації та виконання завдань.
- Потрібно зазначити що пошук може відбуватися не тільки по радіосигналу а й візуально

##### 3. Автоматична система автопілоту:

- Автопілот – це ключовий елемент БЛА, який забезпечує автоматизоване управління та стабільність польоту.
- Система автопілоту може включати в себе алгоритми регулювання та стабілізації, системи управління двигунами, а також системи уникнення зіткнень.
- Для реалізації автопілоту використовуються високопродуктивні мікроконтролери та програмне забезпечення для обробки даних і прийняття рішень.

Аналіз цих аспектів допоможе нам зрозуміти сутність технологій безпілотних літальних апаратів, а також допоможе вдосконалювати їхню продуктивність та надійність. Дослідження цих питань сприяє подальшому розвитку безпіотної авіації та розширенню її можливостей у різних галузях.

Потрібно розділити тему аналізу на 2 частини

- 1.1. Пошук БЛА якщо він втратив зв'язок пультом керування та впав на землю нерухомий
- 1.2. Включення автопілоту для повернення з точки зльоту при роботі РЕБ.

З розробки безпілотного літального апарата (БЛА) з метою полегшення його виявлення у траві чи на поверхні землі важливо враховувати декілька факторів і можливостей:

- Використання світлових сигналів: Світлові сигнали на БЛА, які можуть бути активовані для підвищення видимості, особливо в низьких світлових умовах або вночі.
- GPS-трекінг: GPS-трекінговий пристрій на БЛА. Це дозволить відстежувати його місцезнаходження в реальному часі за допомогою спеціального програмного забезпечення або мобільного додатку.
- Запис даних про місцезнаходження: Дані про місцезнаходження апарата на вбудованому дисплеї або індикаторі в режимі

реального часу, що дозволить операторові впевнитися, що БЛА не втрачено.

- Автоматичне приземлення в небезпечних ситуаціях: Система автоматичного приземлення або відключення, яка активується в разі втрати зв'язку або інших небезпечних ситуацій.

Кожен із цих способів має свої недоліки та переваги. Почну зі світлової індикації. Плюси цього методу що це візуальний пошук, він може відбуватись або через інший БЛА не ризикуючи життям в пошуках в небезпечних ділянках. Основний недолік що візуально його може знайти й інша людина яка опинилася поряд.

GPS – трекінг. Його можна використовувати в цивільних потребах. Переваги очевидні: дуже швидкий та точний пошук втраченого БЛА. Координати можна виводити на пристрій прийому або на карту. Недоліки: повна недієздатність та працездатність в умовах РЕБ. Більш того, використання GPS та мобільного зв'язку в таких умовах є неймовірно небезпечною для оператора.

Запис даних про місцезнаходження. Доволі непоганий варіант для пошукових заходів адже тут немає світлового сигналу та його ніхто не побачить, GPS не видає сигнал. Записувати маршрут можна по waypoints та передавати його через LoRa сигнал який буде з певною частотою відправляти на пульт сигнал. Недолік цього методу криється в тім що це лише приблизне місце де втрачений БЛА приземлиться.

Автопілот при роботі РЕБ. Якщо глушилка включена поблизу БЛА, це може призвести до таких наслідків:

- Втрата зв'язку: Глушилка може блокувати радіосигнали, які використовуються для керування БЛА. В результаті цього може виникнути втрата зв'язку між пультом керування і БЛА.
- Автоматичне повернення на базу: Багато сучасних БЛА мають функцію автоматичного повернення на базу (Return to Home), яка

активується при втраті зв'язку. Це означає, що БЛА може автоматично повертатися до точки старту або до позначеної безпечної області, коли відсутній зв'язок з пультом.

- Вплив на безпеку і навігацію: Блокування радіосигналів може вплинути на безпеку польоту БЛА та його навігацію. Безпілотний літальний апарат може втратити здатність до автономної навігації та стати неконтрольованим.

Функцію **return to home** роздивимось в другому розділі існуючих систем та поясню його найважливіший недолік. В цілому при роботі РЕБ зв'язок з БЛА втрачається та як показує практика він йде в контрольоване падіння, цьому сприяють датчики на борту, та приземляється в полях. БЛА можна і надалі використовувати але його потрібно знайти та забрати, і іноді зробити це безпечно для життя оператора, а це і є моя мета роботи – швидко та безпечно шукати втрачений БЛА.

## **2.1. Аналіз існуючих аналогів автопілоту та пошуку у відкритому просторі**

Для візуального пошуку втраченого безпілотного літального апарата (БЛА) існують різні модулі та інструменти, які можна використовувати для полегшення цього процесу:

1. Теплові камери: Теплові камери або інфрачервоні камери дозволяють виявити теплові випромінювання апарата, таке як двигуни та акумулятор, навіть у нічний час або в умовах обмеженої видимості. Такі камери можуть бути вбудовані в пульт керування, відеореєстратори або дрони пошуку. Очевидний їх недолік це те що немає змоги запускати БЛА при роботі РЕБ.
2. GPS-трекінг: Якщо втрачений БЛА обладнаний GPS-модулем і зберігає трекінгові дані, ви можете використовувати цю

інформацію для визначення останнього місця розташування апарата.

3. Мультикоптери зі світловими маяками: Деякі БЛА для пошуку мають вбудовані світлові маяки або світлодіодні системи, які можуть бути активовані під час пошуку. Це допомагає збільшити видимість БЛА у повітрі.
4. Використання додатків і програм: Існують додатки та програми для смартфонів і планшетів, які дозволяють вам відслідковувати і керувати БЛА через GPS-координати та інші дані.

Всі ці методи об'єднує одне – потрібен зв'язок з втраченим БЛА або з БЛА який шукає. Іноді це неможливо або небезпечно для життя.

Щодо функції **return to home** (автопілоту) в БЛА. Тут треба зрозуміти як це все працює. Сам БЛА керується пультом на зазначеній частоті при налаштуванні перед стартом, при втрачанні пульта та БЛА постійного зв'язку запускається режим return to home, така конфігурація буде актуальна при цивільному використанні, РЕБ в свою чергу виявляє на якій частоті відбувається мовлення між пультом та БЛА та просто посилає свої команди з більш сильною інтенсивністю що заставляє БЛА виконувати інструкції саме з РЕБ мовлення. Заводські налаштування ніяк не можуть протидіяти РЕБам різного калібру, мовлення на радіозв'язку не зникає тому і автопілот не спрацьовує. Додаткових модулів для цивільних БЛА ще немає.

## 2.1. Постановка задачі

Мета дипломної роботи – створити проект модулів які допомагають в пошуках або протистоянні РЕБам або уникнути життєвих ризиків при пошуках в небезпечних ділянках. Створення проекту модулів для пошуку та протистояння радіоелектронним блокувальникам (РЕБам) та забезпечення безпеки при пошуках у небезпечних ділянках може мати велике значення для різних галузей, таких як рятувальні операції, військові застосування, дронний контроль і багато інших. Варіанти систем:

- Системи відновлення зв'язку – Розробка методів і технологій для відновлення зв'язку з БЛА в умовах дії РЕБів;
- Захист від вторгнень – Розробка систем захисту від вторгнень і заборони несанкціонованого доступу до БЛА;
- Системи аварійного відновлення – Розробка автоматичних систем, які дозволяють БЛА повертатися на базу або в безпечну зону у випадку атаки РЕБа або інших аварій;
- Система аварійної подачі світлосигналу у разі неможливості продовжити політ.

Розробка модулю подачі світлосигналу не є складним алгоритмом, але потрібно враховувати декілька моментів. Світло не повинно з'являтися під час польоту, а лише в тому випадку коли БЛА лежить нерухомим деякий час. Також для економії енергії на доцільного використання світлодіодів необхідно забезпечити роботу цього модуля саме в темний час. Це повинен бути автономний модуль зі своїм акумулятором, датчиками та контролером.

Модуль пошуку з допомогою LoRa можна використовувати як радар в якому діапазоні та на якій відстані лежить БЛА, це буде стабільно працювати при певному налаштуванні радіочастот, але не дуже стабільно в «польових» умовах та за наявності РЕБів.

Модуль автопілоту можна доробити майже на всіх БЛА, з певними доопрацюванням та інтеграцією контролера та маніпуляторів для руху БЛА з'явиться можливість відключити основний контролер управління та передати маніпуляцію над БЛА додатковому контролеру який весь час працював в режимі запису координат. В певний момент контролер перейде в режим зчитування та вернеться по координатам на місце зльоту ігноруючи РЕБ вказівки тому що основний чіп вже не буде відповідати за маніпуляцію БЛА.



## Висновки

У ході дослідження та розробки модуля для безпілотних літальних апаратів (БЛА) з метою протидії радіоелектронним блокувальникам (РЕБам), були зроблені наступні висновки:

- Актуальність проблеми: Проблема впливу РЕБів на функціонування та безпеку БЛА є актуальною та важливою в сучасному світі. Зростаюча кількість РЕБів створює загрозу для застосування БЛА у різних сферах, включаючи рятувальні операції, військове використання, дронний контроль та інші.
- Спеціалізований модуль: Розробка спеціалізованого модуля для протидії РЕБам в БЛА може значно підвищити ефективність та надійність роботи апаратів в умовах дії радіоелектронних перешкод.
- Безпека і зворотний зв'язок: Важливо розглянути питання безпеки та забезпечити можливість відновлення зв'язку або відновлення управління апаратом у випадку дії РЕБів.
- Навігація та автопілот: Модуль повинен включати функціональність для автоматичної навігації та керування апаратом. Він повинен здати відповідність реакції на РЕБи та забезпечити можливість реагування на їх дію.
- Інтеграція з іншими системами: Модуль повинен бути інтегрованим з іншими системами та компонентами БЛА для забезпечення їх сумісності та спільної роботи.
- Потенційні застосування: Розроблений модуль може бути використаний у різних галузях, включаючи рятувальні операції, військові дії, дронний контроль та інші, де БЛА використовуються для важливих завдань.

Загальним висновком є те, що розробка модуля для протидії РЕБам у безпілотних літальних апаратах має великий потенціал для підвищення безпеки та ефективності їх використання в умовах радіоелектронних

перешкод. Даний проект відкриває нові можливості для подальших досліджень та розробок у цій області.

## РОЗДІЛ 2

### АНАЛІЗ ІСНУЮЧИХ СИСТЕМ РАДІО-ЕЛЕКТРОННОЇ БОРОТЬБИ. РОЗРОБКА БАЗОВИХ КОМПОНЕНТІВ МЕТОДИКИ ПОБУДОВИ МОДУЛЕЙ ДЛЯ ПРОТИСТОЯННЯ РЕБ

#### 2.1. Принцип роботи зв'язку БЛА, види та методи їх управління

##### 2.1.1. Зв'язок БЛА

Безпілотні літальні апарати знаходять все більше застосувань у сферах життєдіяльності людини. Наприклад, БПЛА, оснащені GPS-навігацією, використовуються в сільському господарстві, з їх допомогою зручно запилювати поля, заощаджуючи витрату сировини та більш ретельно обробляючи посіви.

БПЛА використовується і в інших областях:

- для оперативної доставки гуманітарної допомоги, медикаментів до важкодоступних ділянок країни;
- для перевірки трубопроводів, ліній електропередач та виявлення секторів, яким потрібний ремонт;
- для прогнозування НС та постійного моніторингу потенційно небезпечних районів;
- відстеження річкових заторів, дорожніх заторів.

Але технічний прогрес у галузі подібних розробок має й інший бік — застосування БПЛА з розвідувальною та терористичною метою. Це виводить завдання боротьби з безпілотниками на актуальний і насущний рівень.

Для управління безпілотним літальним апаратом (БПЛА) використовується канал зв'язку, який може бути провідним або бездротовим. Цей канал може передавати команди оператора до апарата та отримувати зворотні дані від БПЛА.

1. **Радіоканали:** Більшість безпілотних літальних апаратів використовують радіоканали для забезпечення бездротового зв'язку. Це може бути частотний діапазон від звичайного радіо до

високочастотних діапазонів, залежно від використовуваної технології та вимог.

2. **Супутниковий зв'язок:** Деякі БПЛА можуть використовувати супутниковий зв'язок для отримання команд та передачі даних. Це дозволяє управляти апаратом на великій відстані, навіть коли він перебуває в недосяжному місці для звичайних радіоканалів.
3. **Оптичні зв'язки:** Деякі системи БПЛА можуть використовувати оптичні зв'язки, такі як лазерне або інфрачервоне з'єднання. Це може бути корисно в ситуаціях, де потрібен безперервний лазерний зв'язок.

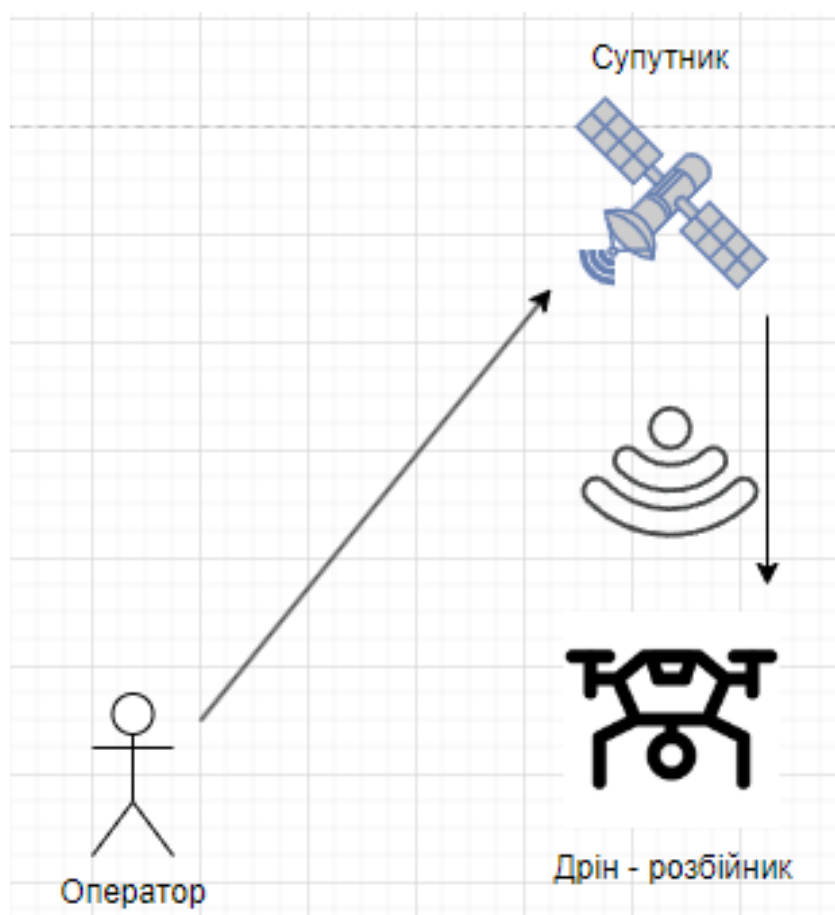


Рис 2.1.1 Зв'язок через супутник

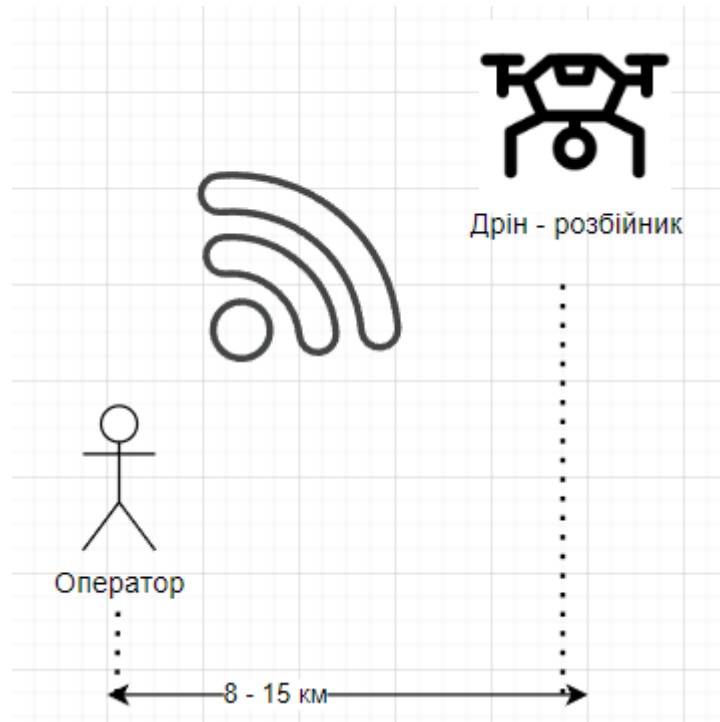


Рис 2.1.2 Зв'язок напряду через радіосигнал

Важливою частиною процесу управління БПЛА є забезпечення стійкості та безпеки каналу зв'язку, щоб уникнути його перехоплення чи втручання з боку сторонніх осіб. Роздивимось поближче радіозв'язок адже він є найбільш популярним методом керування БЛА цивільного та малого радіусу роботи.

#### Переваги радіозв'язку:

1. **Широкий діапазон частот:** Радіозв'язок може використовувати різні частоти, що дозволяє підбирати оптимальний діапазон для конкретних умов та вимог.
2. **Велика дальність передачі:** Радіосигнали можуть подолати великі відстані, що дозволяє управляти БПЛА на великій території.
3. **Висока проникність сигналу:** Радіохвилі можуть проникати через перешкоди, такі як будівлі чи ліс, що робить радіозв'язок ефективним в різних умовах.
4. **Низька затримка сигналу:** Радіосигнали передаються зі швидкістю світла, що забезпечує низьку затримку в управлінні.

#### Недоліки радіозв'язку:

1. **Піддатливість до перешкод:** Радіосигнали можуть бути заваджені і перехоплені, що створює потенційну загрозу безпеці та конфіденційності.
2. **Обмежена пропускна здатність:** У зоні великої концентрації безпілотних апаратів або інших радіоелектронних пристроїв може виникнути конфлікт за канали зв'язку, обмежуючи пропускну здатність.
3. **Вплив погодних умов:** Деякі погодні умови, такі як сильний дощ чи буря, можуть впливати на якість радіосигналу та зменшувати дальність передачі.
4. **Потреба в ліцензіях:** Використання деяких частот може вимагати ліцензії від відповідних регуляторів.

### 2.1.2. Види БЛА.

За методом управління:

- Некеровані. У завдання оператора входить запуск апарата та попереднє введення параметрів та завдань польоту. Такі моделі відносяться до бюджетних, що не потребують облаштування спеціальних злітно-посадкових майданчиків.
- Дистанційні. Апарати, де передбачено налаштування траєкторії польоту за допомогою наземної служби відстеження.
- Автоматичні. Виконують завдання в автономному режимі.

За масою та габаритами

Таблиця 2.1.1

Модель	Радіус дії	Злітна маса (в кг)	Дальність дії
Міні та мікро	Ближній	До 5	25 - 40
Легкі	Малий	5 - 50	10 – 70
	Середній	50 - 100	70 – 150
Середні	Середній	100 - 300	150 – 1000
Середньо-важкі		300 - 500	70 – 300
Важкі	Середній	Від 500	70 – 300

	Великий	Від 1500	1500
--	---------	----------	------

За призначенням:

- Комерційні
- Споживчі
- Військові

Дрони ще можна класифікувати за кількістю та розташуванням двигунів, і основною категорією за цим критерієм є кількість двигунів. Існують три основні групи, а саме трикоптери, квадрокоптери та гексакоптери. У трикоптерів три двигуни розташовані на фронтальній та бічних сторонах, що може призводити до обмеженої маневреності при поворотах та обмеженої вантажопідйомності.

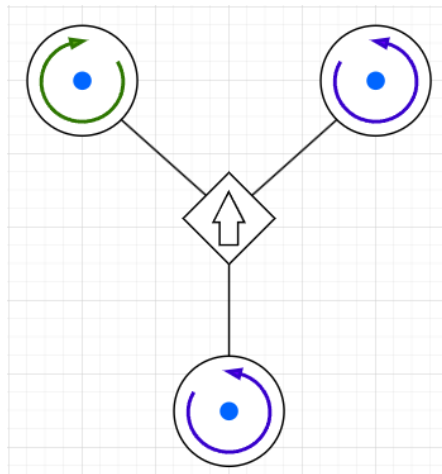


Рис. 2.3.4 Схема трикоптера

До другої категорії входять квадрокоптери - найбільш поширені які оснащені чотирма двигунами. Однак основним недоліком цієї групи літальних апаратів є їхня обмежена стійкість до відмов: у випадку втрати одного з двигунів дрон втрачає баланс та падає. Якщо використовувати потужні двигуни, ці конструкції можуть перевозити корисне навантаження до 5 кг. Є два різновиди конфігурація «X» та «+»

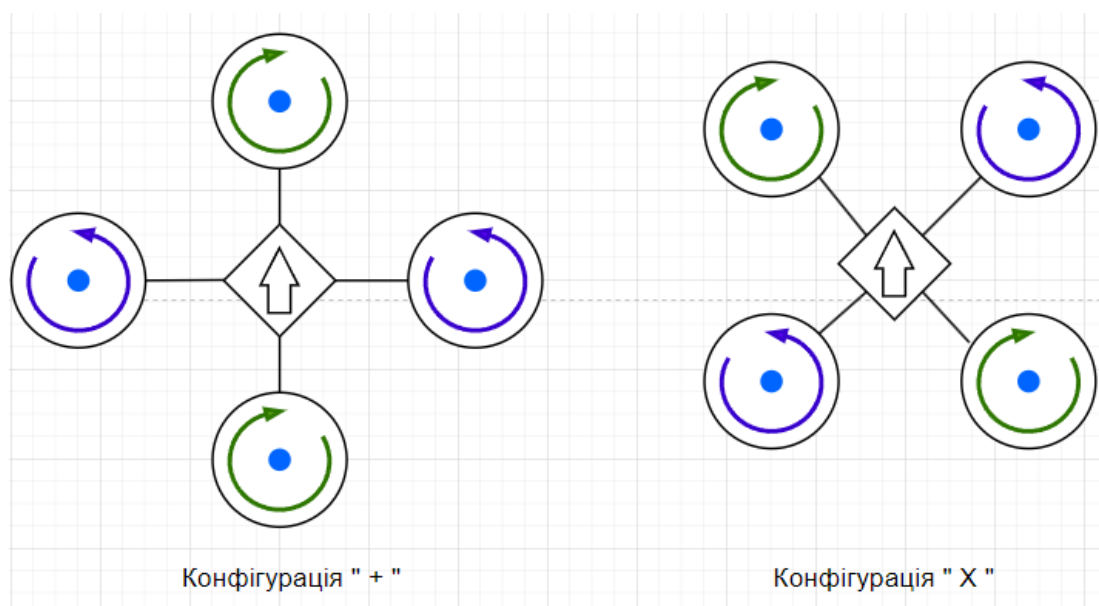


Рис. 2.3.5 Схема різновидів квадрокоптера

У третю категорію входять гексокоптери та октокоптери, що представляють собою важкі літаючі апарати, здатні перевозити вантажі вагою до 20 кг. Конструкція цих дронів є складною, вимагає спеціального обладнання та програмування для управління політним контролером. Важливо відзначити, що дрони цієї групи володіють високою стійкістю до відмов: у разі виходу з ладу одного з двигунів дрон може продовжити польот, рухаючись на залишкових двигунах.

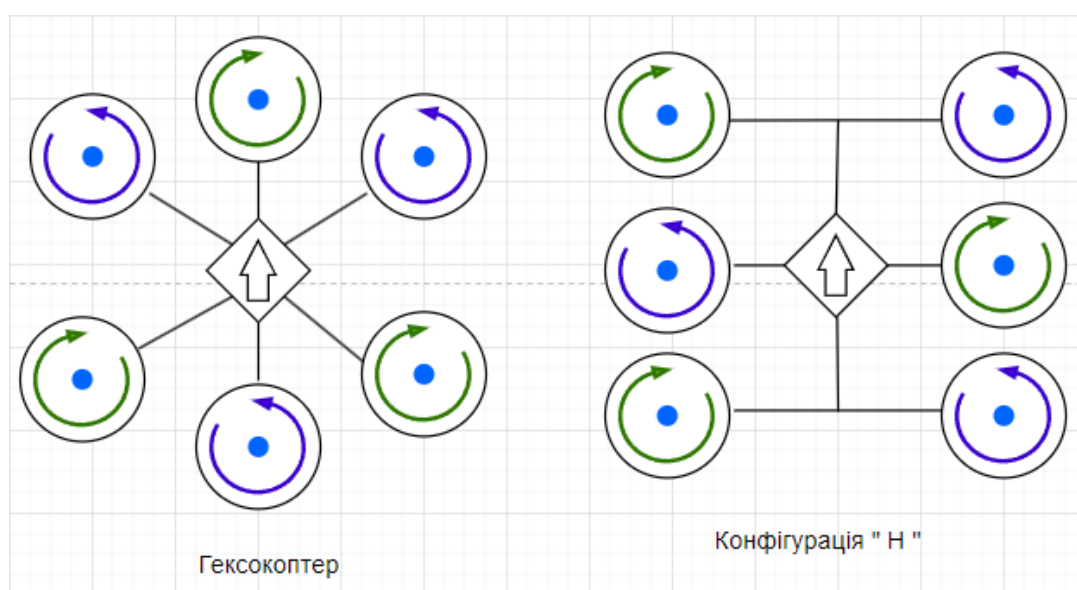


Рис. 2.3.6 Схема різновидів гексокоптера

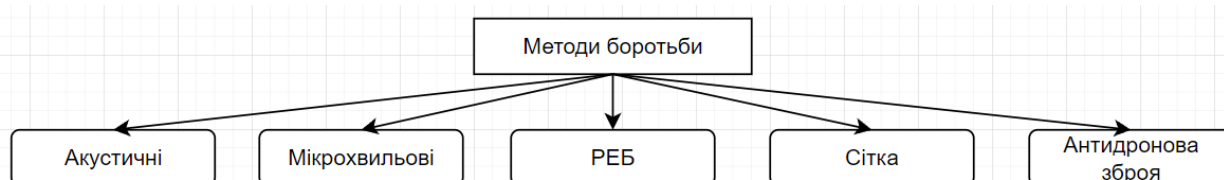


Коптер керується чотирма гвинтами, кожен з яких приводиться в рух власним двигуном. Половина гвинтів обертається за годинниковою стрілкою, а інша половина - проти, що виключає необхідність хвостового гвинта. Для маневрування апарат змінює швидкість обертання гвинтів.

Управління рухом дрона базується на конкретних алгоритмах. Наприклад, для підняття дрона прискорюється обертання всіх гвинтів за рахунок збільшення керуючої напруги на двигунах. Для руху в бік або інший напрямок, необхідно змінити швидкість обертання гвинтів на одному боці та уповільнити на іншому. Для повороту дрона прискорюється обертання гвинтів, які обертаються за годинниковою стрілкою, і уповільнюється ті, що обертаються проти (і навпаки). Управління двигунами виконується за допомогою мікропроцесора, який має програмовані алгоритми для різних режимів руху дрона.

## 2.2. Аналіз існуючих систем РЕБ. Розгляд методики роботи цих систем та вплив на БЛА

У світі стрімкого розвитку технологій безпілотні літальні апарати стають не лише популярними інструментами для цивільного використання, але й предметом ростиючої уваги в контексті безпеки та оборони. З інноваційними можливостями, які надають БПЛА, зростає і потенційна загроза їх використання в неправомірних або ворожих цілях. Таким чином, актуальність питань, пов'язаних із засобами боротьби з безпілотними системами, стає надзвичайно важливою. У даному дослідженні розглянемо різноманітні методи боротьби з БПЛА, їхні переваги та недоліки, а також визначимо тенденції у розвитку технологій для контролю та протидії цим автоматизованим літакам.



### Рис 2.2.1 Методи боротьби

Розглянемо кожний з них, це нам необхідно для розуміння як ми можемо протистояти або рятували БПЛА в різних випадках.

#### *Акустичні*

Акустична зброя виводить з ладу безпілотник через дестабілізацію його гіроскопа. У цій ситуації безпілотний апарат втрачає здатність орієнтуватися у повітряному просторі та падає. Але такий вид впливу несе потенційну небезпеку оточуючим, адже невідомо, куди впаде безпілотник.

**Мікрохвильові.** Радіочастотна мікрохвильова зброя працює на найпростішому принципі: в район перехоплення дрону йде пучок спрямованих радіохвиль з певною щільністю енергопотуку. Мікрохвильові промені руйнівню діють на електронну начинку апарату та виводять її з ладу.

**РЕБ.** Установки для радіоелектронної боротьби належать до одним з найпоширеніших методик боротьби з безпілотниками. Такі моделі відрізняються компактністю і за розмірами набагато меншими за лазер. РЕБ провокує перешкоди між БЛА, супутником та пультом управління, що призводить до акуратного приземлення дрону.

Існує кілька різновидів РЕБ за видами дії:

- автоматизовані для виявлення безпілотних апаратів у певному спектрі (радарні, оптичні, акустичні, з радіовипромінюванням, змішаного типу);
- для перехоплення керування дрону;
- для створення перешкод у системі керування безпілотника;
- створення перешкод у пристроях геопозиціонування на частотах ГЛОНАСС/GPS;
- для створення перешкод у системах бортової електроніки (мікрохвильові модулі, що працюють на ЕМІ).

Подібні установки функціонують у режимі імпульсів і мають такі різновиди:

- установка ближнього радіусу дії (дальність до 300 м), система відрізняється мобільністю та легко розміщується навіть на невеликому автомобілі;
- далекобійна (дальність кілька кілометрів), більш габаритні установки з потужними випромінюючими антенами та генераторами.

### *Сітка*

Противодрони, забезпечені мережами з кевлару (пара-арамідне волокно підвищеної міцності). Вони мають радіус захоплення до 300 м (залежить від модифікації) і можуть працювати як в автономному, так і керованому режимі. Корпус снаряда складається з розташованих уздовж центральної осі металевих блоків, у кожному з них закладена вибухівка та спеціальні грузила (вражаючі елементи), з'єднані еластичними ріжучими складовими. Вибухівка автоматично спрацьовує при досягненні кінцевої мети (дрону), вивільняючи вантажі, які розтягують сполучні елементи. Формується ріжуча сітка, що знищує безпілотник.

Ці системи розроблені для боротьби з безпілотними літальними апаратами (БПЛА) шляхом перешкоджання їхнім системам зв'язку та управління. Вони можуть використовувати різні методи для завадження роботі дронів. Давай розглянемо деякі характеристики та можливості антидронових РЕБ-рушниць:

1. **Перешкоджання радіоканалам:** Антидронові РЕБ-рушниць можуть генерувати електромагнітні сигнали для завадження радіоканалам, які використовуються для управління та зв'язку з дронами. Це може призводити до втрати зв'язку або зменшення контролю над БПЛА.
2. **Імітація сигналів:** Деякі РЕБ-рушниць можуть імітувати сигнали, які зазвичай використовуються дронами для навігації або отримання команд. Це може спричинити сплутання дрона та вивести його з ладу.
3. **Інтерференція з GPS:** Деякі системи можуть створювати завади в сигналах GPS, що призводить до непрецезної навігації для дрона.

4. **Детекція і блокування сигналів дронів:** РЕБ-рушниці можуть виявляти сигнали, що відповідають діапазонам, використовуваним безпілотниками, і блокувати або перешкоджати їм.
5. **Аналіз та виявлення загроз:** Деякі системи обладнані алгоритмами аналізу сигналів для розпізнавання типу дрона та визначення його потенційної загрози.
6. **Мобільність та портативність:** Багато антидронових РЕБ-рушниць розроблені з урахуванням мобільності, що дозволяє їм легко пересуватися та реагувати на загрози в різних умовах.

Важливою перевагою цих систем є їхні можливості здійснювати боротьбу з дронами, знищуючи або вилучаючи їх з експлуатації, і тим самим забезпечувати захист від можливих загроз.

Для формування вимог до модуля потрібно брати до уваги дані та методи які перешкоджають роботі БЛА. У військових цілях на фронті більш всього використовують мобільний та стаціонарний РЕБ малого та більшого радіусу, а також антидронові рушниці. В цих випадках БЛА «губиться» та сідає або РЕБ перехвачує управління з допомогою інших програм та інструментів але важливо зрозуміти що в багатьох випадках БЛА сідає або падає на землю цілим або майже цілим, є доцільно використати його ще раз. Однак знайти свій БЛА після того як його заглушив ворожий РЕБ не дуже легка задача. Якщо БЛА підданий впливу засобів радіоелектронної боротьби, його поведінка може залежати від конкретних характеристик використовуваного РЕБ і те, які системи БЛА виявилися порушеними.

Можливі сценарії включають втрату управління БЛА, зміну його маршруту, тимчасову неможливість передачі даних або команд оператору. У ряді випадків БЛА може втратити стабільність і автоматично сідати (екстрена посадка), щоб уникнути руйнування. Для цього використовують сильні станції для прикриття стратегічних об'єктів або розтягуються на 2 – 3 лініях оборони по фронту. Такі станції не завжди ефективно покривають кінцевий

радіус дії, а розставити їх по всій лінії фронту можливості немає. На перших лініях оборони застосовується застаріла техніка або стаціонарна, в обох випадках дія на БЛА в них є але вони не вміють виконувати перехоплення БЛА для його подальшого керування. Тому в більшості випадків БЛА глушаться та сідають на землю в фронтових зонах, звідки їх потрібно дістати. На зараз не існує чіткого, вбудованого інструменту для ЗСУ який би допомагав знаходити БЛА в таких випадках без додаткових ризиків для життя.

### 2.3. Формування моделі роботи рятувального модуля

Отже підрахувавши всі можливості втрати БЛА під час різноманітних задач, будь то військова розвідка або тренувальні польоти можемо прийти до висновку що об'єкт виживає в таких випадках і його можна використовувати ще раз, але його потрібно знайти та забрати на це витрачається час та не на всі БЛА обладнані функцією подальшого пошуку. Хотів би виділити два методи для пошуку та один для протидії РЕБ:

- Модуль для візуального пошуку
- Модуль радіо-маячок
- Модуль перехоплення управління (автопілот)



Рис. 2.3.1 Умовне позначення модулів

За основу платформи для побудови такої системи буде використовуватися Arduino. Ардуіно — це відмінна платформа для побудови таких проектів і ось чому:

1. **Простота використання:** Ардуіно має зручний і простий інтерфейс, що робить його доступним навіть для початківців. Код на Ардуіно пишеться мовою C/C++, але використовується спрощений фреймворк, що полегшує розробку.
2. **Велика спільнота та ресурси:** Існує активна спільнота користувачів Ардуіно по всьому світу, і ви зможете знайти безліч ресурсів, таких як форуми, блоги, та підручники. Це полегшує вирішення проблем та отримання допомоги.
3. **Широкий вибір модулів і датчиків:** На ринку існує величезна кількість різноманітних модулів та датчиків, які легко інтегруються з Ардуіно. Це робить платформу дуже гнучкою для різноманітних проектів.
4. **Відкритий вихідний код та модульність:** Ардуіно — це відкритий вихідний код, що дозволяє розробникам вносити власні зміни та розширювати можливості платформи. Модульність коду сприяє швидкому розробленню та тестуванню різних функцій.
5. **Екосистема:** Існує багато готових бібліотек та скетчів, які можна використовувати в своїх проектах. Це дозволяє значно економити час та зусилля під час розробки.

З'єднання датчиків із мікроконтролером Ардуіно, має ряд значущих переваг, що сприяє успішній реалізації різноманітних проектів:

1. **Збільшення функціональності:** Датчики дозволяють отримувати реальні дані з навколишнього середовища або об'єкта. Це може включати в себе інформацію про температуру, вологість, освітленість, рух, тиск і т.д. Додавання датчиків розширює можливості вашого

проекту та дозволяє здійснювати реакції на зміни в навколишньому середовищі.

2. **Природній ввід даних:** Датчики забезпечують природний спосіб введення даних у ваш проект. Наприклад, можна використовувати датчики температури для вимірювання та контролю температурного режиму в якомусь пристрої або середовищі.
3. **Зручність інтерфейсу:** Багато датчиків мають стандартні інтерфейси, такі як I2C, SPI або аналогові та цифрові входи/виходи. Це робить їх легкими у підключенні та використанні з мікроконтролерами. Більшість датчиків постачаються з документацією та прикладами коду, що полегшує їхнє використання.
4. **Можливість автоматизації:** Датчики можна використовувати для автоматизації реакцій на зміни в середовищі або стані об'єкта. Наприклад, вимикач світла, який реагує на рівень освітленості, або система автоматичного поливу, що враховує рівень вологості ґрунту.
5. **Модульність та легкість відладки:** Датчики дозволяють створювати модульні системи, де кожен датчик відповідає за конкретний аспект функціонування. Це полегшує відладку та підтримку, оскільки ви можете вносити зміни в окремі частини системи без впливу на інші.

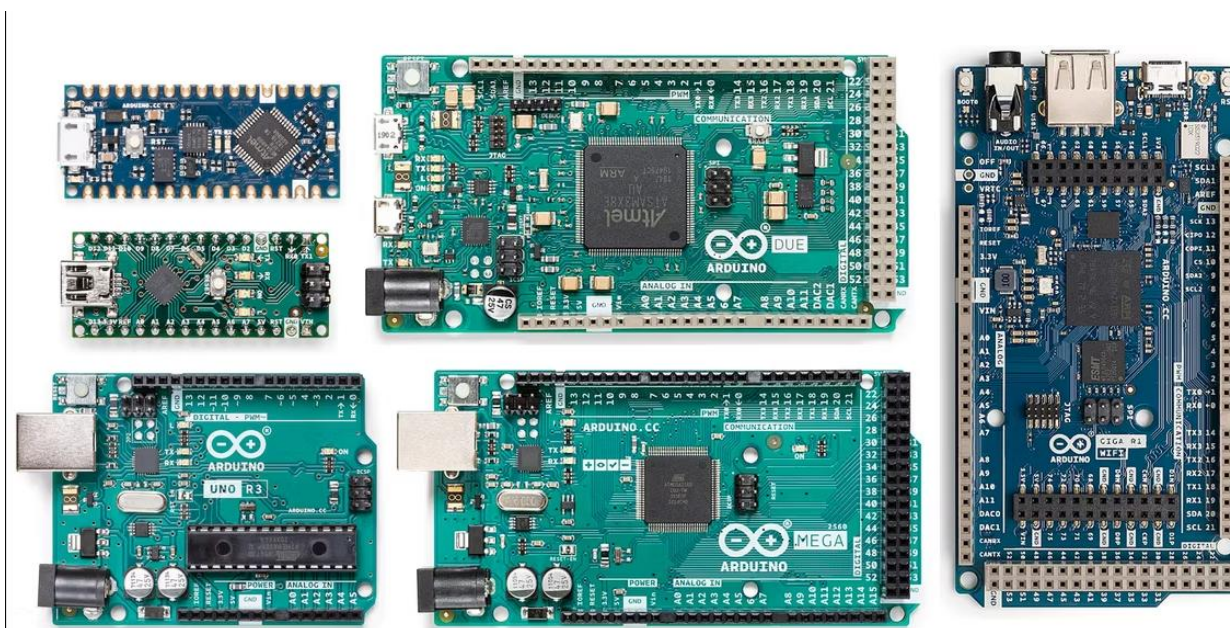


Рис 2.3.2 Моделі Arduino

З картинки ми можемо побачити різноманітність вибору моделі для будь-яких задач. Чим більший форм-фактор тим більш спроможний чіп та функціонал можна реалізувати. Для реалізації проекту варто брати до уваги такі характеристики кінцевого продукту:



Рис 2.3.3 Важливі характеристики модулю

З цієї схеми ми розуміємо що використовувати потрібно збалансовану схему Ардуіно дивлячись на співвідношення вага-габарити-функціонал.





### 2.3.1. Модуль візуального виявлення

*Модуль візуального пошуку* базується на самому простому принципі роботи це подавати світлові сигнали в разі падіння на землю. Є деякі жорсткі умови роботи такого модуля – не можна щоб він мигав вночі коли БЛА рухається адже при використанні в військових цілях він стає дуже легким таргетом для його збиття з будь якої зброї. Це здається дуже очевидним фактом але з «заводу» на даний момент БЛА цивільного використання обладнані саме таким принципом роботи, також постійна робота світлового маячка зменшує ресурс роботи самого БЛА, частина енергії витрачається в нікуди тому що він працює і в день коли від світлового маячка ефективність нульова. Потрібен принцип роботи який буде максимально ефективний:

1. Подавати сигнал тільки вночі або в темний час доби
2. Під час польоту виключити можливість спрацювання світлового маячка
3. Додати можливість заряджання модулю для його повторного використання
4. Реалізувати кнопку включення-виключення для використання

Важливо зазначити – чим менш технологічний пристрій тим він надійний. Саме це потрібно для візуального пошуку втраченого БЛА.

На платформі ардуіно можна реалізувати такий пристрій з використанням компонентів:

1. Мікросхема Ардуіно
2. Модуль зарядки
3. Акселерометр
4. Фоторезистор
5. Кнопка вкл-викл
6. Акумулятор
7. Світлодіоди

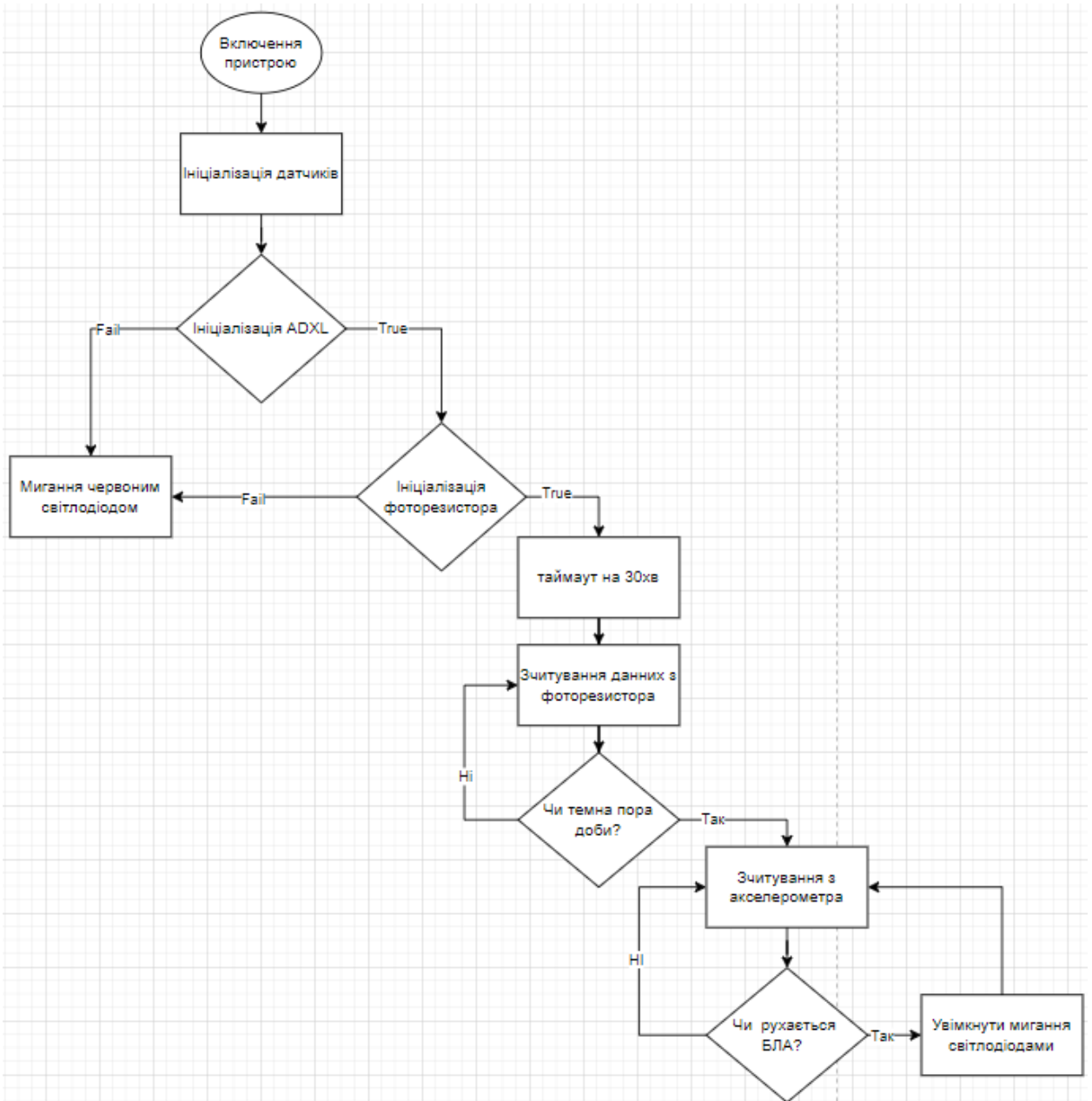


Рис. 2.3.7. Блок-схема роботи світлового маячка

### 2.3.2. Модуль радіо-маячок

*Модуль радіо-маячок* потребує більш сприятливих умов його використання, адже в основі принципу роботи використовується радіосигнал. Пам'ятаємо що в деяких ситуаціях радіо-сигнал може глушитись РЕБом і ефективність даного модуля буде зведена до 0. За основу взято все теж саме Ардуіно та LoRa, Ардуіно вмiє з ним працювати. Технологія LoRa (Long Range) є бездротовим зв'язком, розробленим для передачі даних на великі відстані при низькому споживанні енергії. Вона працює на низькочастотному радіочастотному діапазоні і забезпечує значну дальність передачі даних, що робить її ідеальною для використання в системах відстеження, включаючи пошук загублених дронів.

Якщо дрон обладнаний LoRa-модулем, його можна відслідковувати за допомогою LoRa-мережі. Основні особливості LoRa для пошуку загублених дронів:

1. **Дальність передачі даних:** LoRa здатна передавати дані на великі відстані, із можливістю пройти через перешкоди, такі як будівлі чи ліс, що дозволяє визначати місцезнаходження дрона на значних відстанях.
2. **Низьке споживання енергії:** Технологія спроектована для роботи при низькому рівні споживання енергії, що особливо важливо для дронів, де обмежені ресурси акумуляторів.
3. **Велика витривалість батареї:** Завдяки оптимізації споживання енергії, дрон, обладнаний LoRa, може використовувати свою батарею протягом тривалого часу, що важливо при пошуку.
4. **Мережева архітектура:** LoRa може використовувати мережеві топології, такі як мережі "зірка", "сітка" або "мішаний тип", що дозволяє побудувати ефективну систему відстеження для знаходження загублених дронів.

Загалом, технологія LoRa стала популярною для використання в сценаріях, де необхідно відстежувати рухомі об'єкти на великій відстані з ефективним використанням енергії, такі як пошук та відновлення загублених БЛА.

Взаємодія між технологією LoRa і платформою Arduino відбувається за допомогою LoRa-модулів, які забезпечують можливість бездротової комунікації на великій відстані. Ось кілька ключових аспектів цієї взаємодії:

1. **LoRa-модулі:** Для забезпечення бездротового зв'язку за допомогою технології LoRa, використовуються спеціальні LoRa-модулі. Ці модулі мають вбудований чіп LoRa, а також інші елементи, що дозволяють їм працювати з мікроконтролерами, такими як Arduino.
2. **Вибір модулів:** Є кілька різних LoRa-модулів на ринку, і багато з них можна легко інтегрувати з Arduino. Наприклад, модулі від Semtech, Dragino, або Heltec зазвичай є популярними серед користувачів Arduino.
3. **Бібліотеки для Arduino:** Існують спеціальні бібліотеки для Arduino, які дозволяють легко взаємодіяти з LoRa-модулями. Наприклад, бібліотека "RadioHead" часто використовується для реалізації комунікації LoRa на платформі Arduino.
4. **Підключення та конфігурація:** Щоб почати взаємодію між Arduino та LoRa-модулем, спочатку підключіть модуль до плати Arduino з використанням відповідних пінів. Потім використовуйте спеціальні функції та команди в бібліотеці, щоб конфігурувати та відправляти/приймати дані через мережу LoRa.
5. **Робота в режимі передавача та приймача:** Arduino може бути налаштовано як передавач або приймач за допомогою програмного коду. Відповідно до конфігурації, Arduino LoRa-модуль може надсилати дані на великі відстані або приймати дані з інших вузлів в LoRa-мережі.

Ця взаємодія відкриває широкі можливості для створення проектів з використанням технології LoRa та Arduino, таких як системи відстеження, та інші застосування, де важлива бездротова комунікація на великій відстані.

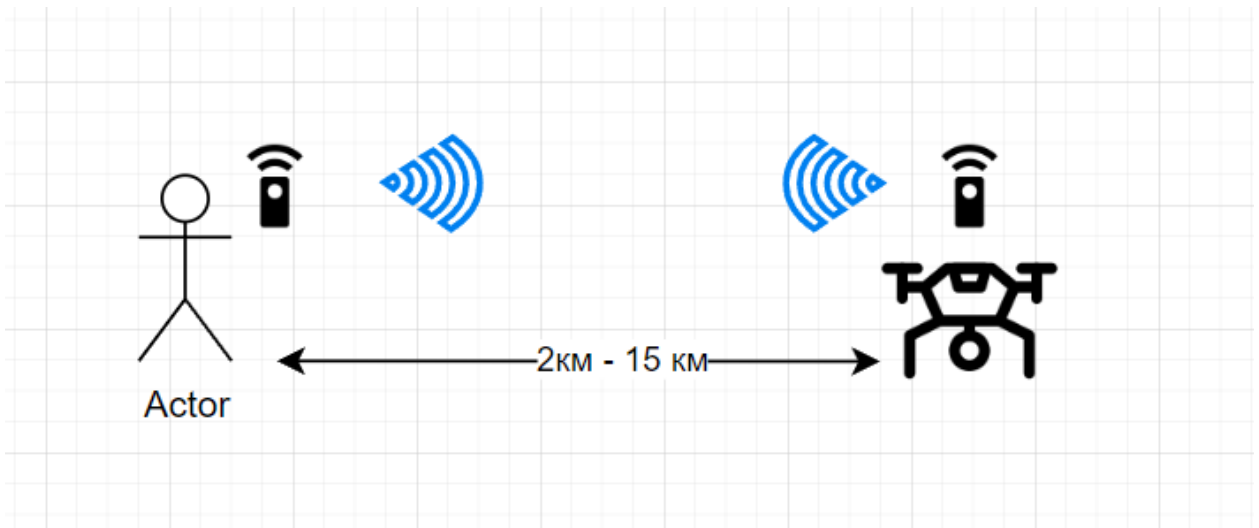


Рис. 2.3.8 Схема дальності роботи LoRa сигналу

Технологія LoRa (Long Range) працює на різних частотах залежно від регіону світу. Існує кілька основних діапазонів частот, які використовуються для реалізації LoRa-мереж:

- 868 МГц: Використовується в Європі.
- 915 МГц: Використовується в Північній та Південній Америці.
- 433 МГц: Також є діапазоном частот для деяких регіонів.

Важливо враховувати, що вибір частотного діапазону залежить від регіона та законодавства, яке регулює використання радіочастот в цьому регіоні. У кожному регіоні можуть бути встановлені обмеження і правила для використання бездротових частот. Таким чином, при виборі LoRa-модулів або розробці проекту важливо враховувати відповідність обраного частотного діапазону з місцевими нормативними вимогами.

Для виявлення дронів у рамках радіолокаційних засобів розглядаються також опції їх деактивації. Спеціальні радіочастотні генератори перешкод використовуються для "осліплення" дронів, впливаючи на сигнали керування та частоти супутників навігації.

Однак у рамках використання радіолокаційних засобів для виявлення дронів існують обмеження. Вони ефективно функціонують на відкритих просторах, де відсутні високі дерева, великі будівлі, акваторія, аеропортські території та інші об'єкти. При цьому вони майже не виявляють компактні безпілотники, що запускаються близько або летять на низьких висотах. Такі "малюки" залишаються непоміченими і для камер, які працюють з радіолокаційними системами, через обмежені кути огляду. Радіоелектронні бортові системи (РЕБ) можуть бути розроблені для глушення різних частотних діапазонів в залежності від їхньої конфігурації та призначення. Загалом, РЕБ може глушити частоти в широкому діапазоні від низьких до високих. Тут деякі загальні частотні діапазони, які можуть включати в себе можливості глушення РЕБ:

1. **Низькі частоти (LF, MF, HF):** 30 кГц - 30 МГц
2. **Середні частоти (VHF, UHF):** 30 МГц - 3 ГГц
3. **Високі частоти (SHF, EHF):** 3 ГГц - 300 ГГц

Ці діапазони включають в себе широкий спектр частот, від коротких хвиль до мікрохвильового діапазону, що дозволяє РЕБ глушити різноманітні радіочастоти, що використовуються різними засобами зв'язку та радіорозвідки. Важливо враховувати, що конкретні характеристики РЕБ можуть різнитися в залежності від моделі та призначення.

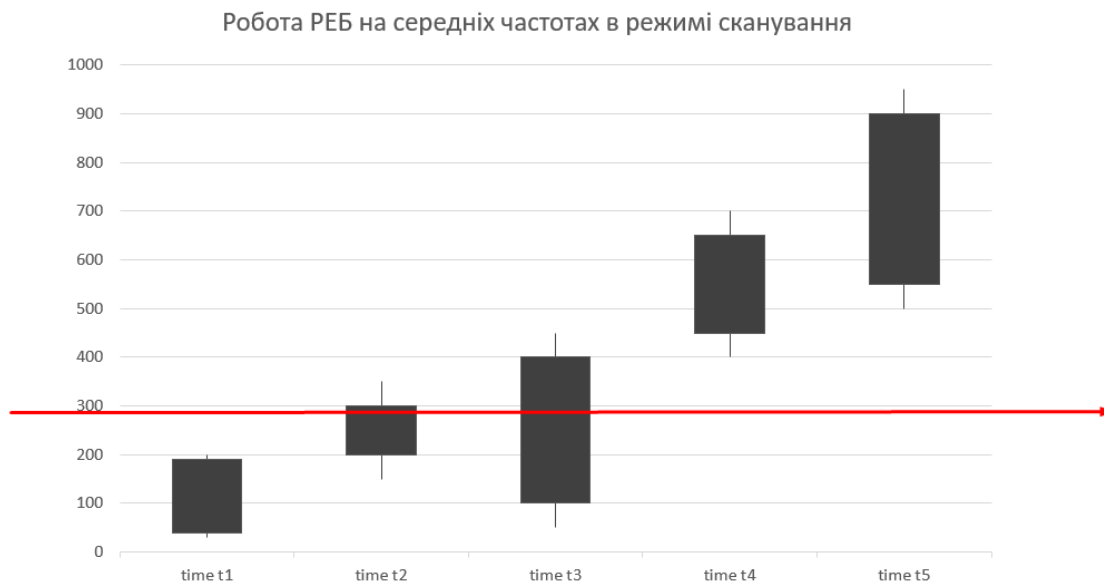


Рис. 2.3.9 Схема роботи РЕБ в режимі сканування

На схемі ми можемо побачити приблизну схему роботи РЕБ в режимі сканування чи блокування радіосигналів на вибраних частотах. Якщо глянемо на список можливостей глушення РЕБ і враховуючи широкополосність можна вважати що радіосигнал від LoRa не пройде між передатчиками, але варто зазначити в такому режимі РЕБ не завжди має змогу працювати адже при таких умовах він стає цілю номер один для авіації з ракетами протирадіолокаційного застосування. Тому беремо до уваги що робота РЕБ буде змішаною або беремо до уваги людський фактор та зміщення частот блокування на яких буде налаштована LoRa. На рисунку 2.3.6 червоною лінією позначено сигнал від передатчика LoRa що стоїть на БЛА, ми можемо побачити що сигнал між пристроями пройде у випадках time1, time4 та time5 в інших випадках він заблокується, але цього уже буде достатньо для передачі сигналу на пуль керування для передачі координат, позиції та відстані до нього.



### 2.3.3. Модуль автопілот

Данна концепція є найбільш технологічною та найскладнішою у реалізації оскільки в вона має бути інтегрована в плату управління БЛА. Сам алгоритм має працювати наступним чином: Керування БЛА відбувається з стандартного ПУ з паралельним використанням модульного ПУ, модульний ПУ постійно обмінюється з приймачем на БЛА зашифрованими ключами які неможливо перехопити, кожну секунду генерується ключ та відправляється на БЛА, ці два ПУ налаштовуються на одну частоту та запускаються. У випадку коли РЕБ глушить сигнал він пропадає з обох пристроїв, як тільки приймач перестає бачити нові ключи він перехватує управління БЛА на себе. Весь час, з моменту включення модулю, він записує шлях по системі «waypoints» якщо інвертувати ці значення то по цій системі можна запустити БЛА назад. Оскільки головний керуючий модуль буде «осліплений» роботою РЕБ його необхідно виключити та передати керування двигунами на запасний модуль.

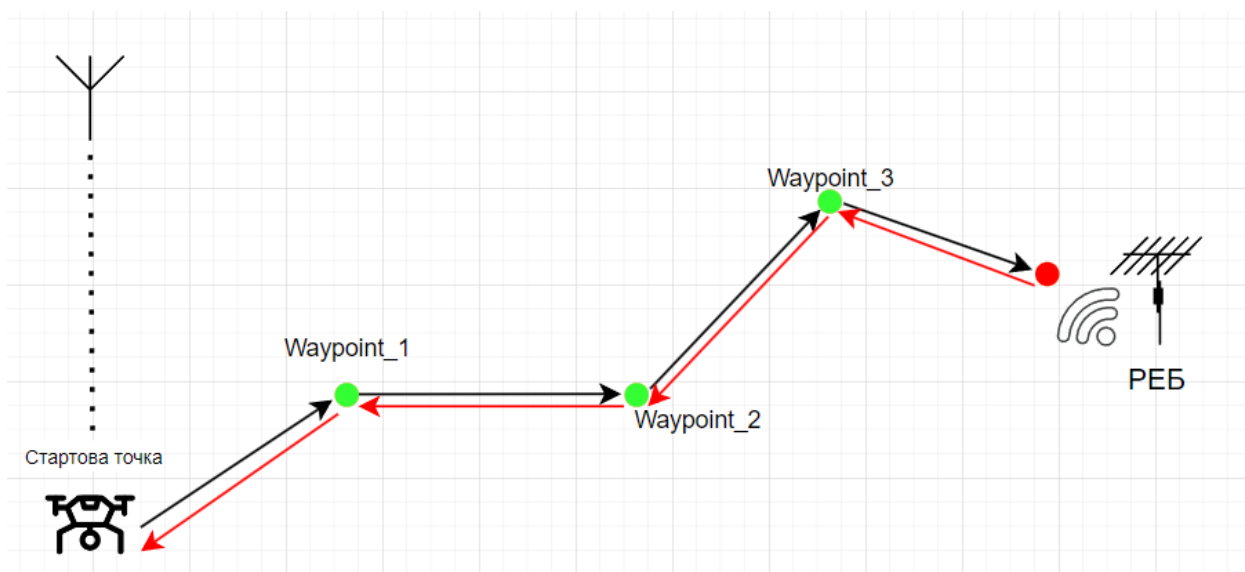


Рис 2.3.10 Абстрактна модель роботи waypoints

Для створення цих waypoints буде реалізована програма на основі датчику акселерометра ADXL саме він може передавати в масив дані щодо вектору руху БЛА. Данні з датчику йдуть в реальному часі що дозволяє записати точні waypoints або навіть створити точний вектор руху в зворотному напрямку. Недоліки та перевага цієї системи це повна втрата керування над

БЛА поки не виконається алгоритм «повернення додому» під час маневрування по waypoint БЛА стає повністю захищеним від любых систем РЕБ адже керує контролер по заданому маршруту без будь якої системи навігації, але стає легкою здобиччю для стрілецької зброї або ППО систем.

## Висновки

У ході розгляду проблеми виявлення та відновлення безпілотних літальних апаратів (БЛА) було виявлено, що існуючі системи навігації та зв'язку можуть стикатися з різними труднощами, такими як глушення сигналу, обмежена ефективність у важкодоступних місцях та недостатня надійність у разі втрати зв'язку. Для вирішення цих проблем та підвищення ефективності використання БЛА, запропоновано створення комплексу з трьох ключових модулів: візуального виявлення, радіо-маячка та автопілота.

Модуль візуального виявлення базується на принципі визначення положення БЛА за допомогою світлових сигналів, що активуються лише в темний період доби. Це не лише ефективно для уникнення видимості ворогам, але й зберігає енергію, оскільки пристрій не працює вдень.

Модуль радіо-маячка використовує технологію LoRa для бездротового обміну даними на великі відстані. Завдяки його характеристикам, таким як велика дальність передачі, низьке споживання енергії та можливість роботи в різних мережевих топологіях, він є ідеальним для відстеження та відновлення БЛА.

Модуль автопілота вирішує завдання автоматизованого управління та повернення в точку зльоту. Принцип його роботи полягає в обміні зашифрованими ключами між головним та резервним пультами управління, що дозволяє підтримувати надійність та безпеку керування БЛА.

Такий комплекс модулів не лише дозволяє уникнути труднощів, пов'язаних із заглушенням сигналу, але і робить систему стійкою до втрати зв'язку та ефективною у різних умовах роботи. Разом вони створюють інтегровану систему, що сприяє покращенню безпеки та надійності БЛА.

## РОЗДІЛ 3

### РОЗРОБКА БАЗОВИХ КОМПОНЕНТІВ МЕТОДИКИ ПОБУДОВИ МОДУЛЕЙ ДЛЯ ПІДВИЩЕННЯ ЖИТТЄДІЯЛЬНОСТІ БЛА ТА ПРОТИСТОЯННЯ ЗОВНІШНІМ ФАКТОРАМ

#### 3.1. Розробка модуля візуального виявлення

За основу розробки береться технологія Arduino як вже писалось вище. Саме на ній буде реалізована програмна та «залізна» частина проекту. Основні вимоги до модуля візуального знаходження:

- Реалізація функції вкл – викл
- Реалізація миготіння в темну пору доби
- Можливість зарядки
- Маскування, оптимізація та налаштування модулю

Виходячи з потреб можемо створити список необхідних частин для роботи:

- Arduino pro mini / Arduino NANO
- ADXL 345
- Фоторезистор GL5516
- Зарядний модуль TP4056
- Акумулятор Li-Po 1000мАч 3.7В формату 523450
- Світлодіоди яскравого світла
- Кнопка вкл – викл
- PNP транзистори
- Резистори 10 – 100 Ком
- Корпус

Зауваження до алгоритму роботи які необхідно дотримуватись або взяти до уваги під час створення програмної частини:

Середній час польоту БЛА малого-середнього радіусу 40 – 50 хвилин під час польоту немає сенсу вмикати миготіння, це безкорисно та іноді демаскує пристрій, враховуючи його задачі. Хибне спарювання тягне електроенергію та демаскує БЛА під час польоту.

Необхідно реалізувати затримку в коді «відкладений старт» та з допомогою ADXL зчитувати чи переміщається БЛА, це дасть нам змогу не вмикати

миготіння під час руху. Фоторезистор необхідний для зчитування кількості світла який припадає на об'єкт, це дозволить реалізувати миготіння лише в темну пору доби. Отже враховуючи всі нюанси ми маємо алгоритм при якому БЛА повинен миготіти тільки при виконанні двох умов:

- Пристрій не повинен рухатися (це означає падіння або втрату керування)
- Повинна бути темна пора доби

При таких умовах це збереже заряд, оскільки пристрій буде миготіти тільки в тих випадках коли це ефективно.

Наступним кроком необхідно зібрати всі деталі попередньо створивши схему

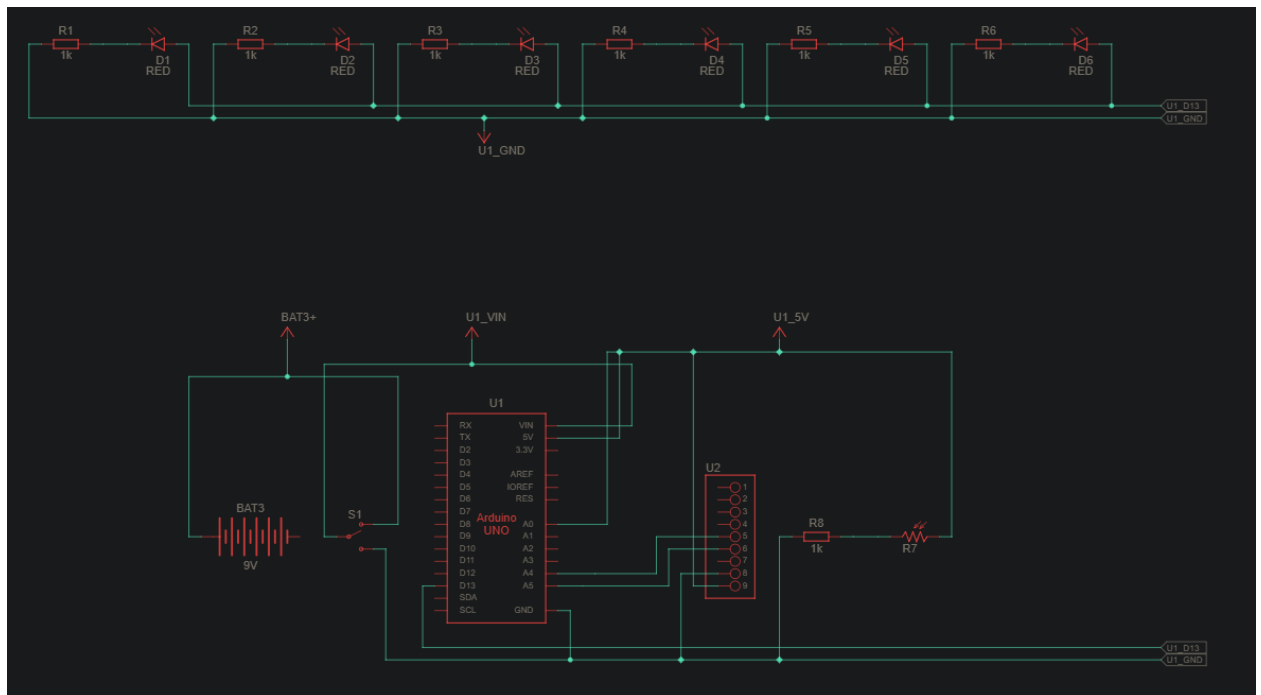


Рис. 3.1.1 Електрична схема модулю візуального виявлення (без зарядного модулю TP4056)

На схемі в ролі головного контролера виступає Atmega 328p на базі Arduino UNO R3 але в фінальному проєкті UNO буде замінено на NANO, база NANO також використовує чіп Atmega 328 та має тіж самі піни для підключення, по розмірам вона менше ніж UNO що позитивно буде впливати на технічні характеристики. Світлодіоди будуть підключені послідовно що

збільшить навантаження на базу Ардуіно, при такому підключенні світлодіоди будуть споживати приблизно 10-12 вольт що є майже максимальним допуском для даної бази Ардуіно. Також з акумулятора регулюється сила струму з допомогою зарядного модуля TP4056 та вбудованого чіпа на самій платі Ардуіно. Сам зарядний модуль впаюється між кнопкою та акумулятором.

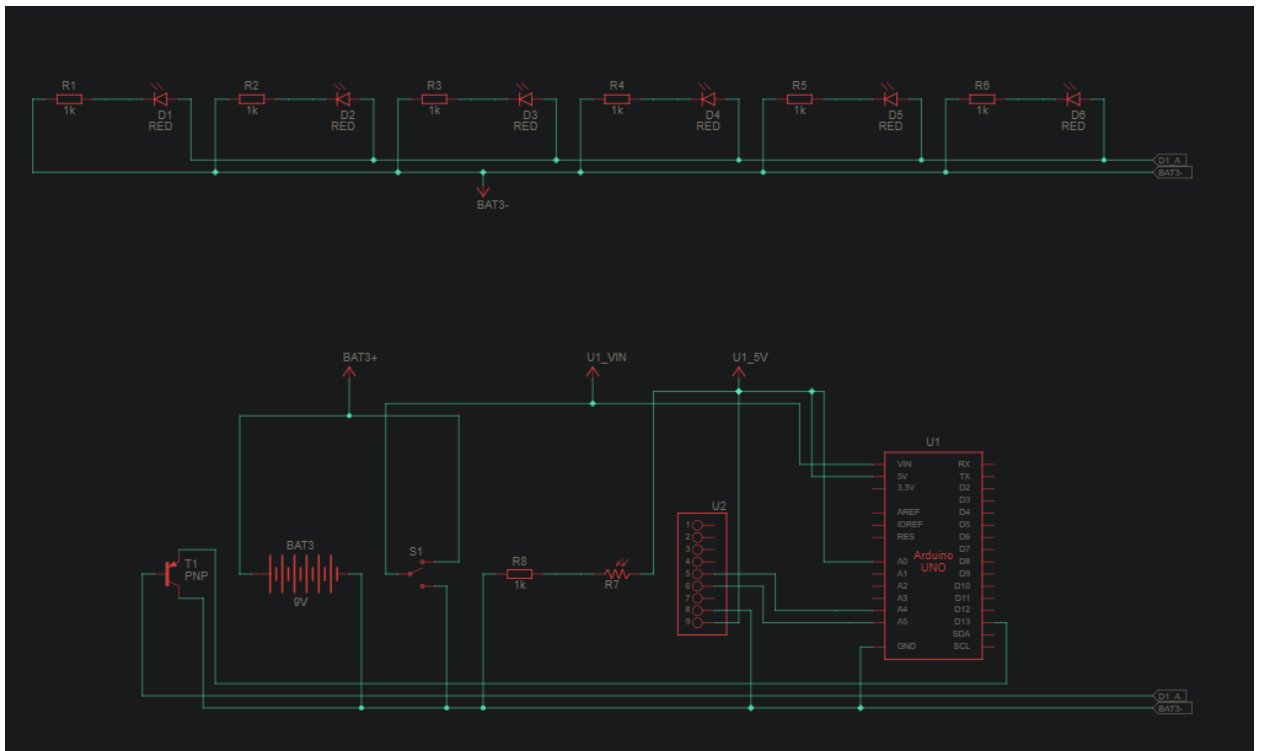


Рис 3.1.2. Електрична схема модулю візуального виявлення оновлена

Під час тестувань було виявлено декілька проблем з підключенням та хибним спрацюванням світлодіодів, оскільки Arduino «пробиває» струм було вирішено поставити транзистор перед світлодіодами для ліквідації цієї помилки.

Для більш зручної інсталяції та роботи з елементами потрібно це все зробити на макетній платі. Вона знадобиться для тестування та швидкої заміни елементів в разі помилки при проектуванні, також дозволить швидко перевірити різні конфігурації та з'єднання.

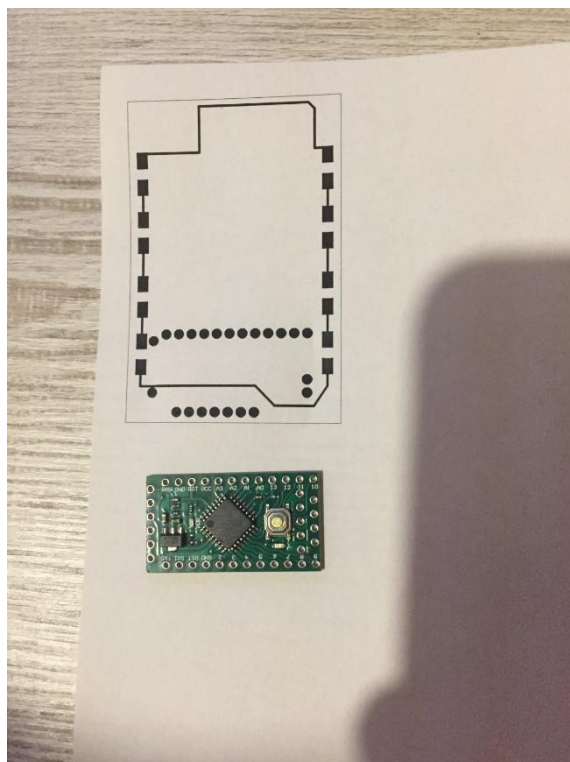


Рис. 3.1.3 Створення приблизного макету

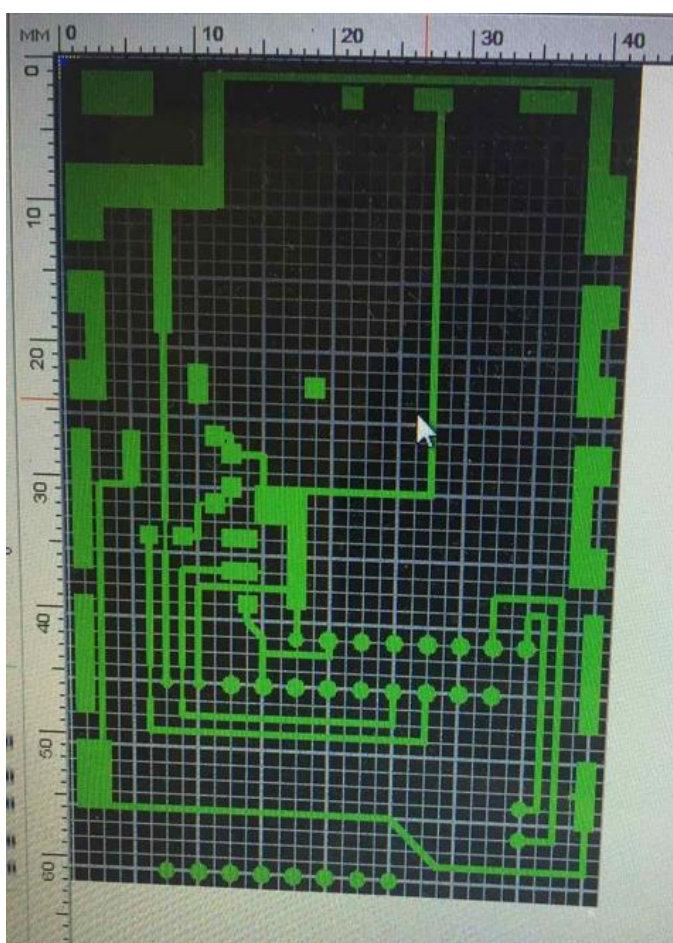


Рис 3.1.4. Проектування макетної плати

Це одна із перших версій, вона буде як основа конкретно для модуля візуального виявлення. На ній розташовуватимуться всі елементи та будуть паятись між собою. Це облегшить процес створення більшої партії.

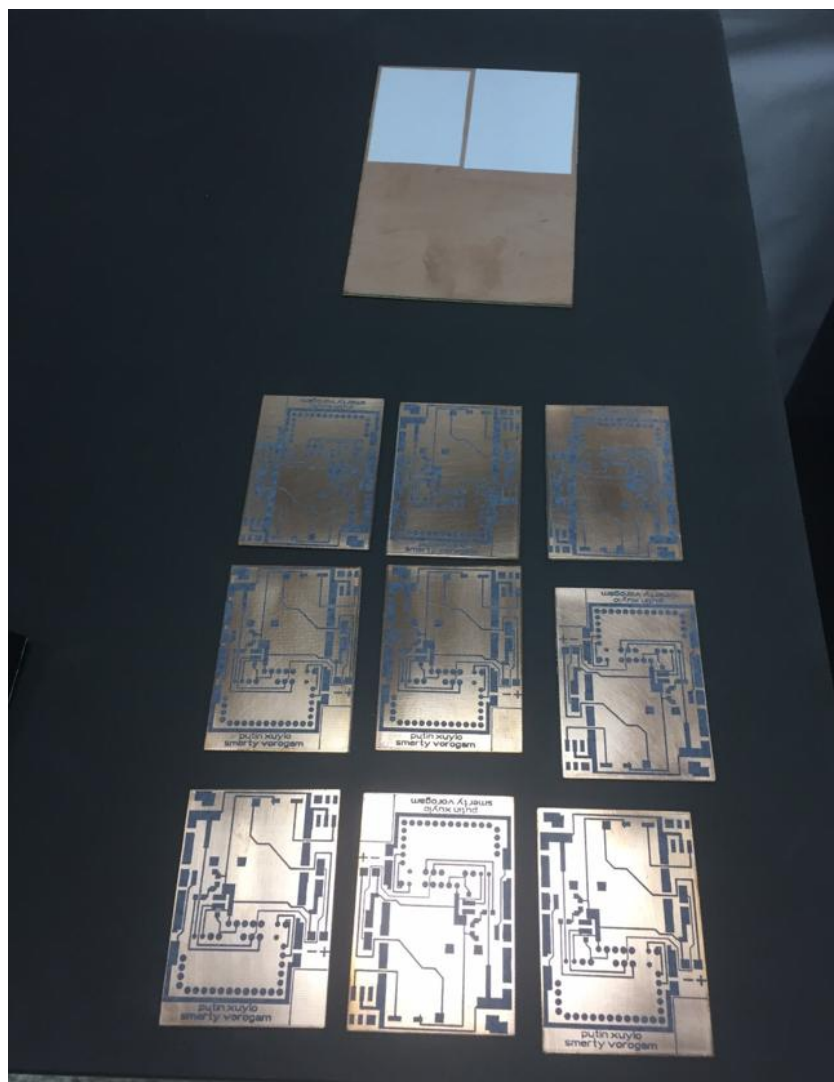


Рис. 3.1.5. Розроблені плати

Саме на них будуть припаюватись елементи, а приблизне розташування можна побачити на наступному малюнку:



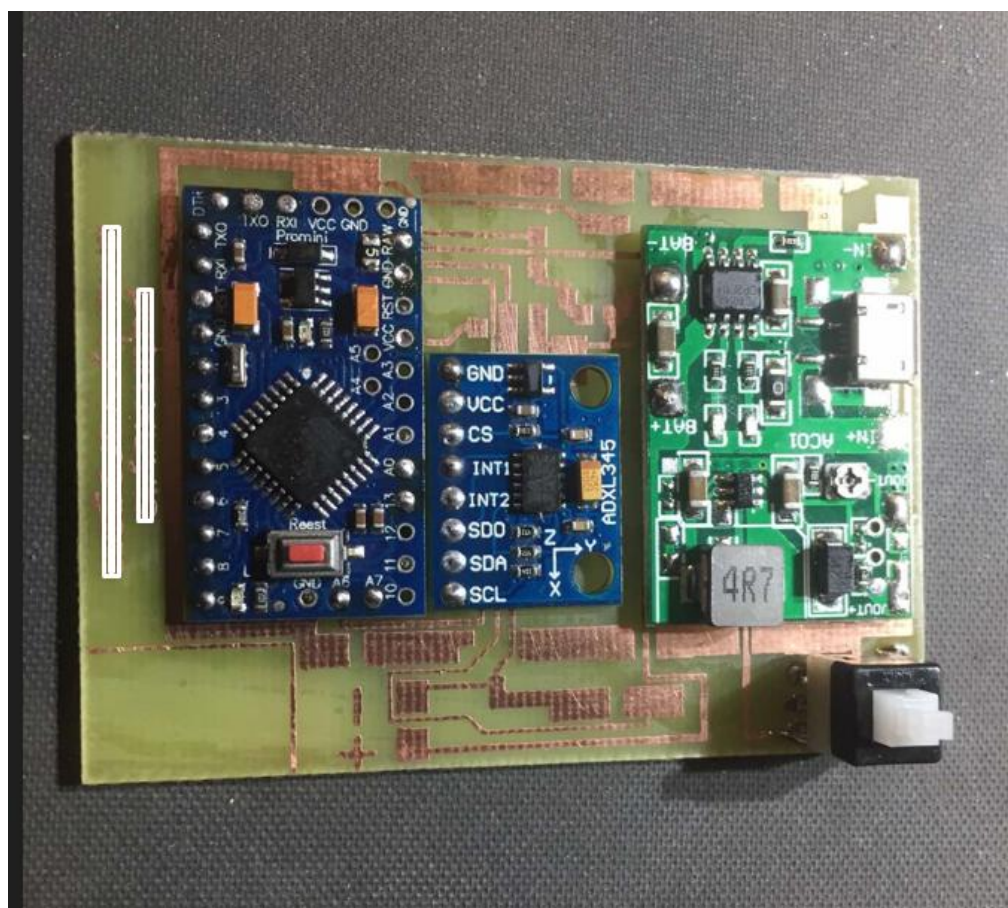


Рис 3.1.6. Розташування елементів на макетній платі(не розпаяні)

На рисунку 3.1.6 вже можна побачити модуль заряду, як і писалось вище він впаюється між кнопкою та акумулятором, до речі кнопка вже впаяна, вона буде відповідати саме за передачу електроенергії з акумулятора до контролера та елементів на платі.



Рис 3.1.7. Розпаяна плата

Було прийнято конструктивне рішення повернути одну групу світлодіодів на 90 градусів для того щоб у разі падіння БЛА на бік світловий пучок формувався в різні сторони а не тільки вверх. Така обумовленість підкріплюється різносторонніми випадками того як саме падають дрони, від РЕБ можуть плавно опускатися, а можуть від вітру або удару об що небудь впасти в різному положенні. Таким чином модуль буде створювати освітленість навколо себе. Також зверху можна побачити кріплення для стяжок, в перших версіях воно буде залите в епоксидну смолу, а технологічні отвори залишаться.

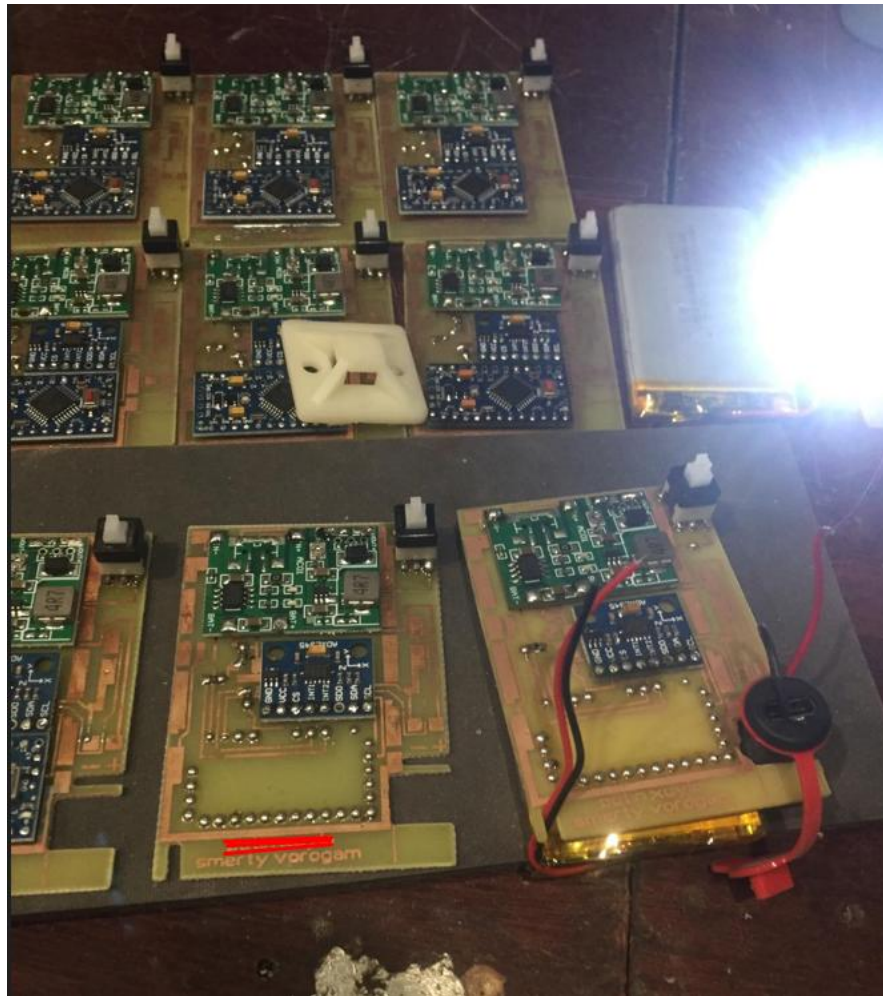


Рис 3.1.8 Інсталяція деталей

Після припаювання усіх деталей необхідно замислитись о корпусі де буде знаходитись плата, важним критерієм є це супротив погодним. Дуже важливо щоб на схему не потрапляв дощ, враховуючи що БЛА може літати в погані погодні умови, або навіть якщо втратить керування, звалиться і піде дощ необхідно щоб модуль відпрацював. На корпус є два варіанта:

- Епоксидна смола
- Корпус з АБС пластику

Габарити плати: 69.2 x 56.8 x 17.9 мм. Знайти потрібний корпус під задані габарити модуля дуже важко. Є необхідність в проектуванні корпусу для печаті на 3д принтері, але це займе час тому першу партію можна залити епоксидною смолою. Мінус такого варіанта що модуль стане важчим майже вдвічі, але з іншої сторони плата набуває класу захисту IP65 що прибавляє їй стійкості в сурових умовах де експлуатуються БЛА. Таким чином модуль в

епоксидній смолі може пролежати під дощем і працювати, також епоксидна смола є прозорою і частенько використовується в військових напрацюваннях. Це потрібно врахувати при проектуванні пластикового корпусу, та забезпечити належний захист від погодних умов.



Рис 3.1.9 Модуль залитий в епоксидну смолу

На рисунку 3.1.9 можна побачити вже готовий модуль до експлуатації. Для створення потрібного корпусу під задані габарити допоможе технологія 3д друку, він використовується для виготовлення фізичних об'єктів за допомогою шарування матеріалу. Це дозволяє створювати складні деталі та пристосування, швидко та з високою точністю, що зручно для прототипування, розробки пристроїв та виробництва унікальних деталей. З допомогою програми Blender створимо прототип корпусу. Базові вимоги до нього це: щільно замикався, мав отвори для кнопки та зарядного порта, був

розбірним, прозорим. Виходячи з потреб можемо створити першу версію корпусу

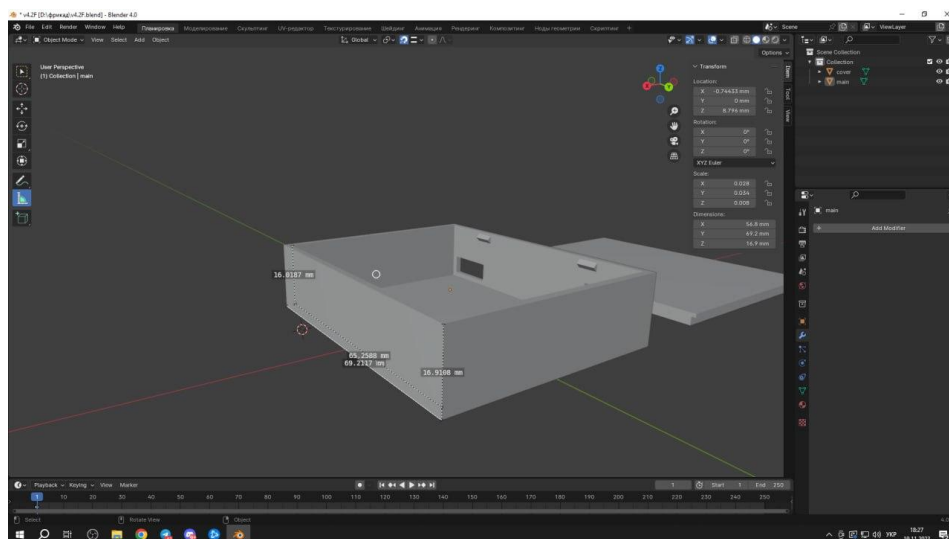


Рис. 3.1.10 Проектування в Blender

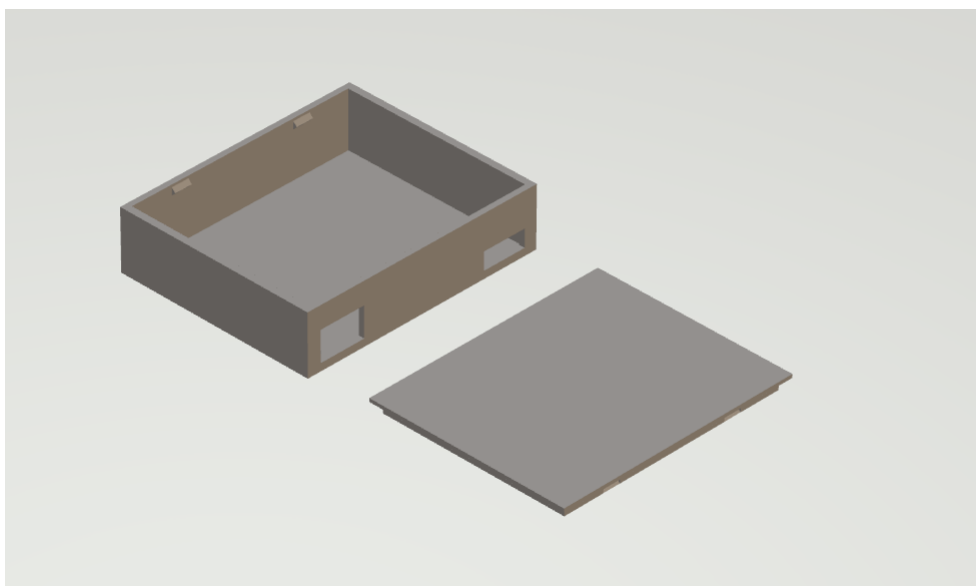


Рис. 3.1.11 Корпус та кришка

На рис 3.1.10 можна побачити зубчики на корпусі вони використовуються як засувки, на кришці є спеціальні отвори для цього. Також можна побачити технологічні отвори в корпусі під кнопку та модуль зарядки.



Рис. 3.1.12 Корпус після печаті на 3д принтері

Програмна частина була залита в мікроконтролер ще до розпайки на макетну плату з допомогою програматора СН341.

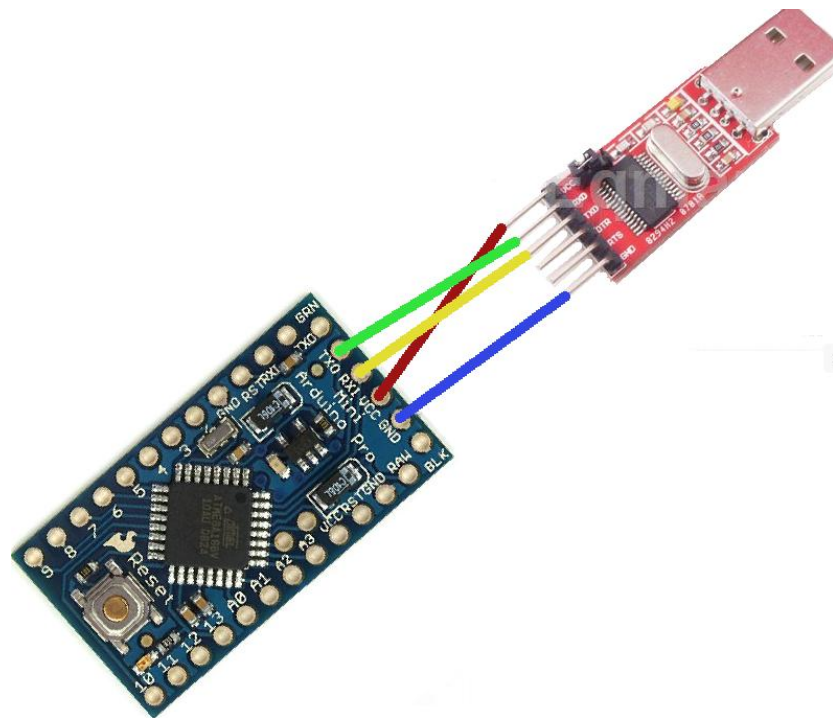


Рис. 3.1.13 Схема підключення програматора СН341а

Оскільки основа на Ардуіно то і використовувати будемо Arduino IDE, схема алгоритму роботи коду є в другій частині, візьмемо її для написання коду:

Почнемо з головної функції `setup` – в ній нам необхідно задати всі базові налаштування для датчиків та пінів самої бази ардуіно

```

void setup() {
  Wire.begin();
  pinMode(13, OUTPUT);
  pinMode(A1, INPUT);
  analogWrite(A1, LOW);
  Serial.begin(9600);
  for(int i=0; i<3; ++i) {
    accelerometer_data[i] = 0;
  }
  init_adxl345();
  check_batt();
}

```

Рис 3.1.14 Функція setup()

В цій функції підключаємо піни в правильному режимі також запускаємо ініціалізатор акселерометра та скидаємо його дані що були записані в контролері під час минулого запуску, для корекції датчика. Також є функція перевірки стану батареї – заряджена чи розряджена. Під час вмикання якщо блимають світлодіоди:

- 3 рази – заряд більше 90%
- 2 рази – заряд в межах 50% – 90%
- 1 раз – заряд менше 50%

Це дасть змогу виявляти перед запуском заряджений модуль чи ні.

```

void init_adxl345() {
  byte data = 0;

  i2c_write(ADXL345_ADDRESS, 0x31, 0x0B); // 13-bit mode +- 16g
  i2c_write(ADXL345_ADDRESS, 0x2D, 0x08); // Power register

  i2c_write(ADXL345_ADDRESS, 0x1E, 0x00); // x
  i2c_write(ADXL345_ADDRESS, 0x1F, 0x00); // Y
  i2c_write(ADXL345_ADDRESS, 0x20, 0x05); // Z

  // Check to see if it worked!
  i2c_read(ADXL345_ADDRESS, 0x00, 1, &data);
  if(data==0xE5)
    Serial.println("it work Success");
  else
    Serial.println("it work Fail");
}

```

Рис 3.1.15 Ініціалізація акселерометра ADXL 345

Основним запитанням в коді це – як зрозуміти що об'єкт рухається? В нас є акселерометр який повинен це питання вирішувати, в нього навіть є функція для трьох сторін:

*abs(event.acceleration.x)*

*abs(event.acceleration.y)*

*abs(event.acceleration.z)*

У контексті коду Arduino з акселерометром, `event.acceleration.x` являє собою вимірне прискорення по осі X в тривимірному просторі. Прискорення може бути позитивним чи негативним залежно від напрямку руху об'єкта.

Використовуючи `abs(event.acceleration.x)`, ми отримуємо абсолютне значення прискорення осі X. Це корисно, коли нам важлива величина прискорення, незалежно від його напрямку. Наприклад, якщо об'єкт рухається праворуч, то `event.acceleration.x` буде негативним, і `abs()` перетворює їх у позитивне число, зберігаючи його величину. Таким чином, функція `abs(event.acceleration.x)` у цьому контексті допомагає врахувати лише величину прискорення, ігноруючи його напрямок. Але є нюанс, він полягає в тому що ця функція зчитує дані в певному проміжку часу, наприклад отримавши за одну секунду три цифри: 0.2 -> 0.4 -> 0.9 можна зрозуміти що БЛА почав рух в певній координаті і дуже важливо якщо він продовжить рух в цій же координаті з тим же кутом нахилу акселерометр прийде в звичайне положення та буде сприймати цей кут як дефолтний (тобто останнє значення 0.9 стане 0.0), із-за цього почне збоїти датчик та видавати неправильні значення.



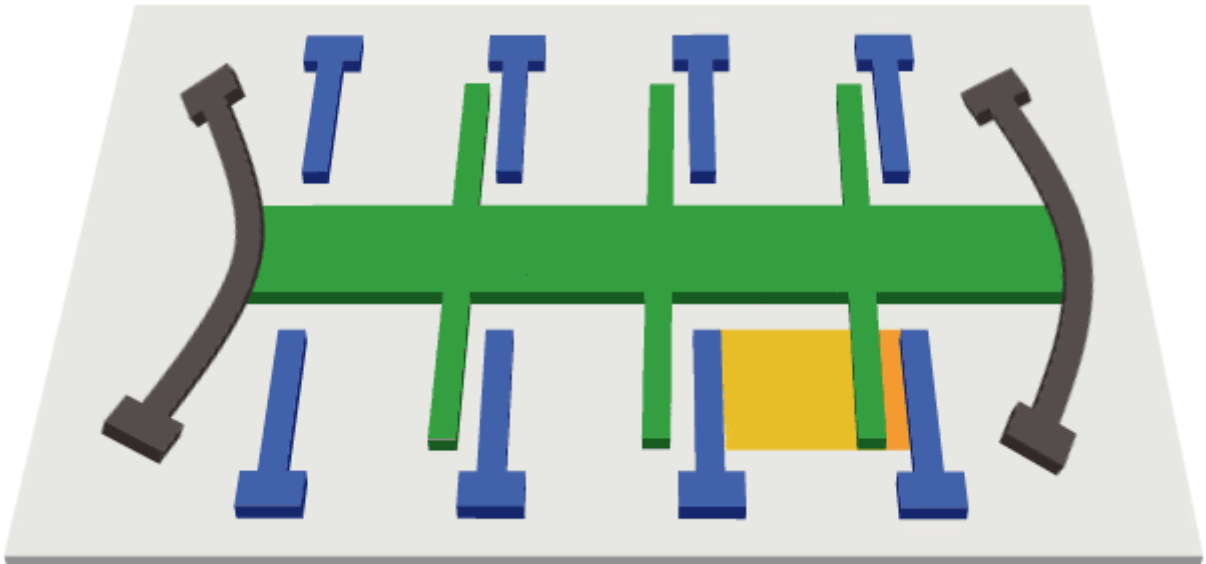


Рис. 3.1.16 Робота акселерометра

В такому випадку не має можливості зрозуміти чи летить БЛА чи впав, адже і під час польоту та приземлення БЛА може нахилитися та завмерти в певному напрямку. Я розробив алгоритм який дає змогу виявити рух об'єкта:

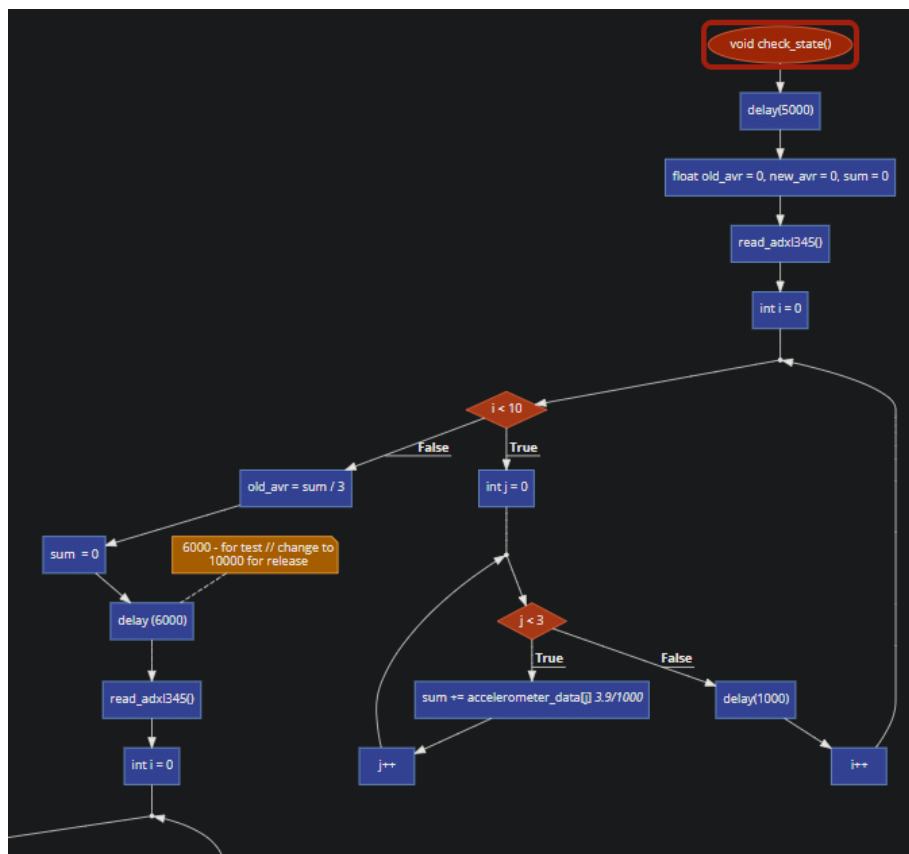


Рис. 3.1.17 Блок-схема алгоритму частина 1

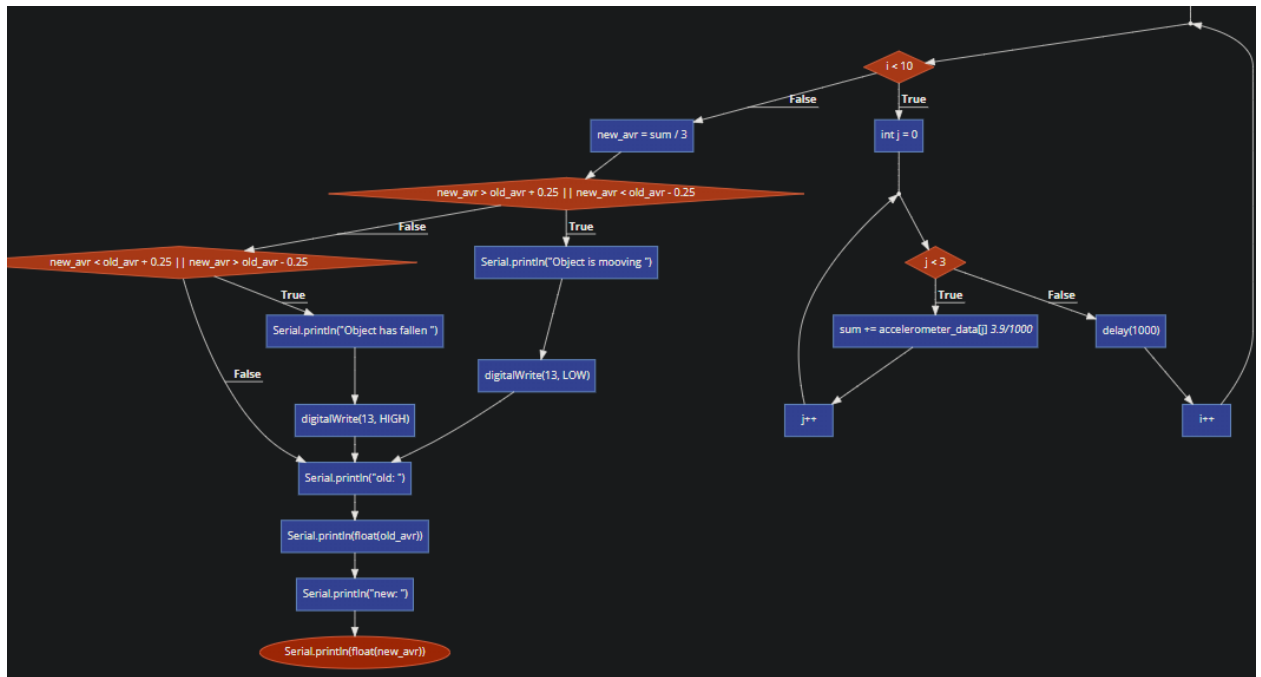


Рис. 3.1.18 Блок-схема алгоритму частина 2

Опишемо алгоритм більш загальним чином:

- Початкова затримка: програма починається з п'яти секундної затримки для стабілізації системи.
- Ініціалізація змінних: Створюються змінні зберігання середніх значень прискорення у двох періодах - `old_avr` і `new_avr`, і навіть змінна `sum` для підсумовування вимірів.
- Перший вимірювальний цикл: Вимірює прискорення протягом 10 секунд кожену секунду. Значення прискорення перетворюються та підсумовуються для розрахунку середнього значення `old_avr`.
- Очікування перед другим виміром: Програма чекає 6 секунд для тестування (може бути змінено на 10 секунд для релізу).
- Другий вимірювальний цикл: Повторюється процес вимірювання прискорення та розрахунку середнього значення `new_avr`.
- Порівняння середніх значень: Перевіряється різниця між `old_avr` та `new_avr`. Якщо різниця більша або менша за  $0.25 \text{ м/с}^2$ , виводиться повідомлення про рух або падіння об'єкта, і керується світлодіод на піні 13.

- Виведення результатів: Виводяться старі та нові середні значення прискорення для налагодження.

Таким чином, програма регулярно вимірює прискорення об'єкта, обчислює середні значення у двох часових періодах та визначає, чи рухається об'єкт чи відбулося його падіння. Результати виводяться в Serial Monitor для відстеження та налагодження. Перевірка руху об'єкта відбувається лише у випадку темної пори доби, це дозволяє не витратити заряд акумулятору на прорахунок коли в цьому немає сенсу.

Тестування модулю відбувається після заливки програми та повної зборки:

1. *Перевірити чи коректно відображає стан зарядку акумулятора.* Для цього використовуємо вольтметр та заміряємо в акумуляторі вихідну напругу та звіряємося з даними що описані вище. **Тест пройдено**
2. *Відкалібрувати фоторезистор на темну пору доби.* Для цього з допомогою програматора я міняю значення та підставляю потрібне поки не задовольнить умовам. **Тест пройдено**
3. *Перевірити спокійний стан модулю.* В спокійному стані світлодіоди повинні почати миготіти, адже об'єкт не рухається. **Тест пройдено**
4. *Перевірити стан польоту.* Під час польоту значення на акселерометрі будуть змінюватися а це означає що БЛА в керованому польоті, світлодіоди не повинні бликати. **Тест пройдено**
5. *Базова затримка перед виконанням коду.* Після включення модулю повинно пройти 30 хвилин перед тим як він почне виконувати алгоритм зчитування значення з датчиків. **Тест пройдено**

Формування потреб до інструкції з використання:

- Умовний опис пристрою
- Заряджання пристрою
- Підготовка на використання
- Важливі додаткові моменти та нюанси

- Технічні характеристики

Оскільки пристрій та документ не є офіційним є можливість відійти від оформлення по ГОСТу оскільки на даний момент це перевантажить інструкцію і буде багато «води» на такий маленький пристрій тим самим створить незручність у використанні.

Висновок:

Розробка модулю візуального виявлення для платформи Arduino виявляється цікавим та перспективним завданням. Зазначені вимоги до модулю включають функції включення-виключення, миготіння в темну пору доби, можливість зарядки, маскування та оптимізацію. Щоб відповісти на ці вимоги, використовуються такі компоненти, як Arduino pro mini, ADXL 345, фоторезистор GL5516, зарядний модуль TP4056, акумулятор Li-Po 1000мАч, світлодіоди яскравого світла, кнопка включення-виключення, PNP транзистори та інші.

Алгоритм роботи модулю розроблено таким чином, щоб ефективно виявляти рух або падіння об'єкта тільки в темну пору доби, оптимізуючи використання енергії. Акселерометр ADXL 345 грає ключову роль у виявленні руху, а алгоритм зчитування та порівняння середніх значень прискорення дозволяє визначити, чи об'єкт знаходиться в стані руху чи падіння.

Процес тестування включає в себе перевірку різних аспектів модулю, таких як коректність відображення заряду акумулятора, калібрування фоторезистора, стан спокою та польоту. Завершальний висновок тестування підтверджує правильну роботу модулю та його відповідність визначеним вимогам.

У процесі розробки виникають ситуації, що вимагають корекцій та оптимізацій, такі як зміни в підключенні світлодіодів для формування освітленості навколо об'єкта, або кріплення для стяжок для забезпечення додаткової стійкості.



Рис. 3.1.19 Зібраний модуль першої версії на зарядці

### 3.2. Розробка модуля радіо-маячка

Концепція модуля радіо-маячка полягає в передачі координат де знаходиться БЛА з модулем. Основна перевага LoRa сигналу в тім що вона працює там де відсутній зв'язок та інтернет, або там де його використовувати небезпечно. Але іноді це є і мінусом адже LoRa відноситься до радіосигналу і вона віддається впливу РЕБ. В другому розділі я роздивися доцільність цієї технології та знайшов певні переваги та можливості тому прийнято рішення на цій технології побудувати модуль пошуку з допомогою радіосигналу.

Проект також буде базуватися на Arduino та мати деякі спільні деталі з минулого проекту, для модуля – передатчика:

1. Arduino PRO MINI / NANO
2. Акумулятор 1500 mAh
3. Кнопка
4. Модуль LoRa
5. Антена передачі
6. Модуль зарядки

Для модуля – приймача

1. Arduino PRO MINI / NANO
2. Акумулятор 3000 mAh
3. Кнопка включення
4. Кнопки навігаційні
5. Дисплей 1602
6. Антена передачі
7. Модуль зарядки

Алгоритм роботи буде базуватися на передачі на вимірювання RSSI сигналу та методу ТОА, вони допоможуть без використання gps виміряти відстань до модуля. Було обрано одразу дві технології оскільки кожна з них має свої недоліки чи переваги. У випадку некоректної роботи одного з них буде відпрацьовувати інший алгоритм.

**Технологія виміру з допомогою RSSI.** Вимірювання відстані з використанням рівня сигналу (RSSI - Received Signal Strength Indicator) в Лора-модулях може бути складним завданням, оскільки точність вимірювань RSSI залежить від різних факторів, таких як навколишнє середовище, перешкоди і перешкоди. Тим не менш, спробуємо використати просту модель для оцінки відстані на основі RSSI рівня. Зверніть увагу, що це буде приблизною оцінкою і не забезпечить високу точність.

Формула для оцінки відстані на основі рівня RSSI може виглядати так:

$$\text{Відстань} = 10^{((TXPower - RSSI) / (10 \times PathLossExp))} \quad (3.1)$$

Де:

*TXPower* – потужність передачі (в дБм), виміряна на передавальному пристрої.

*RSSI* – рівень сигналу (дБм), виміряний на приймальному пристрої.

*PathLossExp\** - коефіцієнт втрат по дорозі, залежить від середовища проживання і умов поширення сигналу. Це параметр, який потрібно калібрувати для певного оточення.

\* Коефіцієнт *PathLossExp* зазвичай визначається експериментально, калібруючи його у конкретних умовах роботи.

**Технологія виміру TOA.** TOA – це метод, у якому визначається час, який потрібно передачі сигналу від передавача до приймача. Шляхом виміру цього часу можна оцінити відстань між пристроями.

Формула для розрахунку відстані з використанням TOA:

$$\text{Відстань} = \frac{\text{Швидкість розповсюдження сигналу} * TOF}{2} \quad (3.2)$$

Де:

*Швидкість поширення сигналу* - швидкість, з якою сигнал переміщається серед. Зазвичай використовується швидкість світла, що дорівнює приблизно  $3 \times 10^8$  метрів за секунду.

*TOF* - час у дорозі, виміряний як різниця між часом відправки сигналу і часом його прийому.

Важливим моментом є синхронізація між пристроями. У прикладі використовувалася мітка часу, додана до пакета під час відправлення. Потім, при отриманні пакета, вимірюється поточний час прийому і TOF розраховується як різниця між часом відправлення і часом прийому.

TOA-вимірювання схильні до різних затримок, включаючи затримки в електроніці, часу обробки та передачі. Ці затримки можуть бути враховані для покращення точності вимірів. Калібрування також може знадобитися для корекції систематичних помилок. Точність вимірювань TOA може бути високою за точної синхронізації часу та мінімальних затримок. Однак, у реальних умовах, таких як наявність перешкод, багатопроменеве поширення сигналу та інші фактори, точність TOA може знижуватися. Проведення калібрувальних експериментів та облік різних факторів допоможуть підвищити точність вимірів.

#### ***Різниця між TOA та RSSI методом.***

Різниця між методами TOA (Time of Arrival) та RSSI (Received Signal Strength Indicator) полягає в тому, як вони використовують інформацію про сигнал для визначення відстані між передавачем і приймачем.



Таблиця 3.1 Зрівняння технологій RSSI та TOA

	RSSI	TOA
Основний принцип	Оцінює рівень потужності сигналу, що надходить від передавача до приймача. Використовує рівень RSSI для оцінки відстані, припускаючи зворотну залежність між потужністю сигналу та відстанню.	Вимірює час, який потрібний сигналу для проходження від передавача до приймача. Вимірювання часу між відправкою та прийомом сигналу використовуються для оцінки відстані.
Точність	Зазвичай менш точний через вплив різних факторів, таких як перешкоди, перешкоди та зміни у навколишньому середовищі.	Висока точність при точній синхронізації часу та мінімізації затримок.
Залежність	Залежить від сили сигналу та сприйнятливий до впливу різних факторів.	Залежить від точної синхронізації між пристроями.
Переваги	Простота реалізації не вимагає точної синхронізації часу, але часто менш точний.	Потенційно більш точні вимірювання за умов, де TOA може бути правильно реалізований.

Як висновок, комбінування TOA та RSSI може бути ефективним підходом для покращення точності визначення відстані в різних сценаріях, але потребує уважного налаштування та адаптації до конкретних умов використання.

### **Макетування проекту.**

Оскільки в нас має бути модуль передавач та модуль приймач, схеми буде дві. Модуль передавач кріпиться на БЛА перед вильотом та він після певного часу починає передавати сигнал на модуль приймач який залишається у оператора. Оснащення модулів трішки відрізняються, в передавачі не повинно бути нічого зайвого окрім частин та деталей що забезпечують передачу LoRa сигналу, а на приймачі повинен бути дисплей куди виводиться вся потрібна інформація яка описувалась раніше – дальність до приймача в метра за методом розрахунку TOA та RSSI.

Також варто відмітити це спосіб розповсюдження сигналу та пошук саме напрямлення де знаходиться модуль передавач. Розглянемо схему:

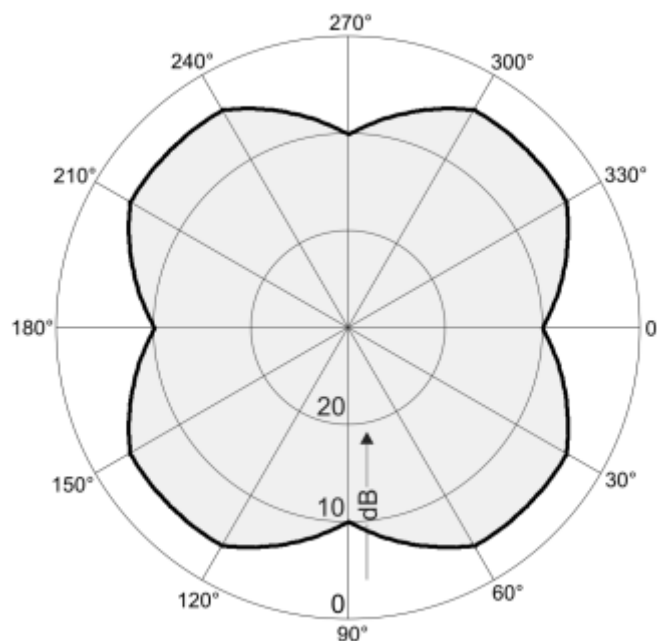


Рис. 3.2.1 Загальна схема розповсюдження сигналу

Ще хочу зазначити та ввести нові змінні для розуміння сил сигналу та розрахунку відстані між ними. Оскільки сигнал LoRa при використанні антени всенаправленої дії зменшує силу сигналу але дозволяє покривати велику площу. За технічними характеристиками БЛА можуть відлітати від оператора на десятки кілометрів, в схемі зазначена сила сигналу до 20 дБ, що при використанні LoRa та сприятливої місцевості означає 5 кілометрів на

частоті 433 ГГц. Більш детально розглянемо один із модулів з солідними показниками:

- максимальний бюджет каналу 168 дБ;
- регульована вихідна потужність +20 дБм (100 мВт);
- чутливість приймача -148 дБм (в режимі LoRa);
- стійкість до навантажень (ІІРЗ = -12,5 дБм);
- встановлення частоти з кроком 61 Гц;

Тепер розрахуємо на яку відстань вистачить такого модулю при використанні всенпарвленої антени:

Для розрахунку максимальної дистанції зв'язку Лора-модуля можна використовувати формулу фріспейсу, яка враховує бюджет каналу (Channel Link Budget)

$$\text{Link Budget} = \text{Потужність передавача (dBm)} - \text{Згасання сигналу (dB)} + \text{Чутливість приймача (dBm)} \quad (3.3)$$

Де: Потужність передавача = +20 дБм (100 мВт)

Згасання сигналу залежить від відстані та інших факторів (у даному випадку ми виходимо з показника 168 дБ бюджету каналу)

Чутливість приймача = -148 дБм

$$\text{Link Budget} = 20 - \text{Згасання сигналу (dB)} - 148 \quad (3.4)$$

Розв'яжемо рівняння щодо Загасання сигналу:

$$\text{Згасання сигналу (dB)} = 20 - \text{Link Budget} - 148 \quad (3.5)$$

Тепер, якщо ми встановлюємо Загасання сигналу, що дорівнює бюджету каналу (168 дБ), ми можемо визначити максимальну відстань:

$$\text{Максимальна відстань} = 10^{\frac{\text{Link Budget} - 148}{20}} \quad (3.6)$$

Підставимо значення та розрахуємо:

$$\text{Максимальна відстань} = 10^{\frac{168 - 148}{20}} \quad (3.7)$$

$$\text{Максимальна відстань} = 10 \text{ км} \quad (3.8)$$

В результаті ми маємо покриття в радіусі 10 кілометрів що непогано, але можна використати направлену антену та значно збільшити цю цифру.

Забігаючи наперед скажу: оператор буде схожий на радар який буде крутитися на місцевості в пошуках сигналу але це вирішує ряд питань пов'язаних з приймачем.

Використання спрямованої антени може значно збільшити дальність зв'язку Лора-модулів. Спрямовані антени мають вузький напрямок випромінювання, що дозволяє концентрувати сигнал у певному напрямку та зменшити згасання на великих відстанях.

Переваги спрямованих антен для розширення дальності включають:

- Збільшення посилення: Направлені антени мають більш високе посилення порівняно з омні-антенами, що дозволяє збільшити силу сигналу в потрібному напрямку.
- Зниження зворотних випромінювань: Зменшення зворотних випромінювань дозволяє зменшити вплив перешкод та підвищити ефективність зв'язку.
- Фокусування сигналу: Спрямовані антени фокусують сигнал у вузькому напрямку, що зменшує втрати на великих відстанях.

Однак, варто враховувати, що спрямовані антени вимагають більш точного налаштування та вирівнювання, оскільки вони чутливі до напрямку. Також, для максимальної ефективності, необхідно враховувати перешкоди на шляху сигналу та вибирати оптимальний напрямок антени.

Формула для розрахунку максимальної дальності з використанням спрямованої антени може бути модифікована з огляду на посилення спрямованої антени ( $G_{\text{антени}}$ ):

$$\text{Максимальна відстань} = \left( \frac{\text{Link budget} - 148 - +G(\text{антени})}{20} \right) \quad (3.9)$$

Де  $G_{\text{антени}}$  – підсилення направленої антени в децибелах

Наприклад якщо підсилення дорівнює 10 дБ то:

$$\text{Максимальна відстань з направленою антеною} = 10^{\frac{108-148+10}{20}} \quad (3.10)$$

$$\text{Максимальна відстань з направленою антеною} \approx 31.58 \text{ км} \quad (3.11)$$

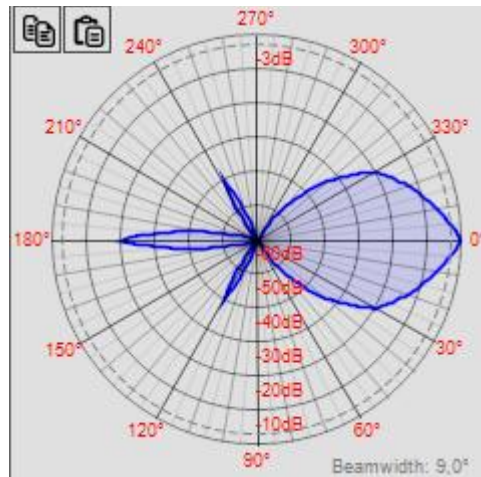


Рис. 3.2.2 Схема з направленою антеною

Таким чином, використання спрямованої антени з посиленням 10 дБ може збільшити максимальну відстань зв'язку приблизно з 10 км до 31.62 км. Однак, як я згадував раніше, це все приблизні розрахунки, і реальні умови, такі як довкілля та перешкоди, можуть суттєво впливати на результати.

### ***Розробка макету та програмної частини приймача.***

Роль приймача у бездротовому зв'язку з використанням технології Лора дуже важлива. Основні функції та ролі приймача:

- **Прийом даних:** Основна функція приймача - приймати дані, передані від Лора-передавача. Ці дані можуть являти собою інформацію з датчиків, що управляють команди або будь-яку іншу форму інформації, яку необхідно передати.
- **Декодування та обробка даних:** Отримані дані, як правило, потрібно декодувати та обробити. Це може включати перетворення бінарних даних в читаний формат або застосування якихось алгоритмів обробки в залежності від типу інформації, що передається.
- **Передача даних мікроконтролеру або іншому пристрої:** Приймач зазвичай підключається до мікроконтролера або іншого пристрою, який займається подальшою обробкою та використанням прийнятих даних. Мікроконтролер може виконати додаткові завдання з урахуванням отриманої інформації.

- Відповідь на запити (якщо необхідно): У деяких сценаріях приймач може надсилати відповіді запити, отримані від передавача. Це корисно в ситуаціях, коли потрібен зворотний зв'язок між передавачем та приймачем.
- Керування процесом зв'язку: Приймач також може керувати параметрами зв'язку, такими як частота, режим передачі/приймання та інші настройки, щоб забезпечити стабільну взаємодію між двома модулями.

В цілому, роль приймача полягає в отриманні та обробці даних, що робить його важливою частиною системи бездротового зв'язку на базі Лора.

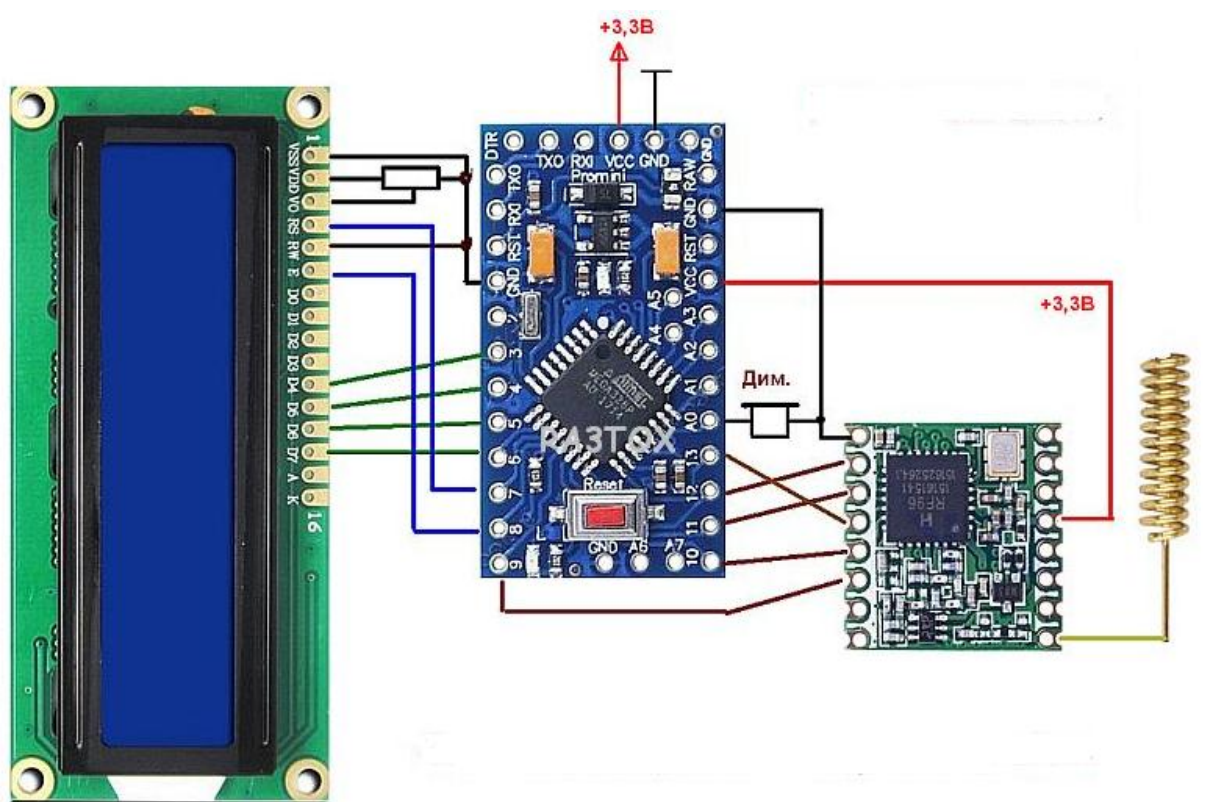


Рис. 3.2.3 Схема підключення LoRa та LCD 1602

Почнемо розглядати схему з антени адже саме вона є першим елементом що активізує роботу алгоритму. На антену приходять сигнал з іншого модуля він передається на LoRa модуль RF96. Далі за зазначеною схемою підключення до потрібних пінів Арудіно про міні дає змогу зчитати з контролера дані та перетворити їх потрібні данні для виводу на екран. Саме на екрані виводиться сила сигналу RSSI, приблизна відстань по методу розрахунку

RSSI та відстань за методом розрахунку TOA. Акумулятор, зарядка та кнопка включення-виключення підключається так само як в проекті 3.1 «Модуль візуального виявлення» та впаюється в плату Ардуіно.

Програмна частина базується на алгоритмі:

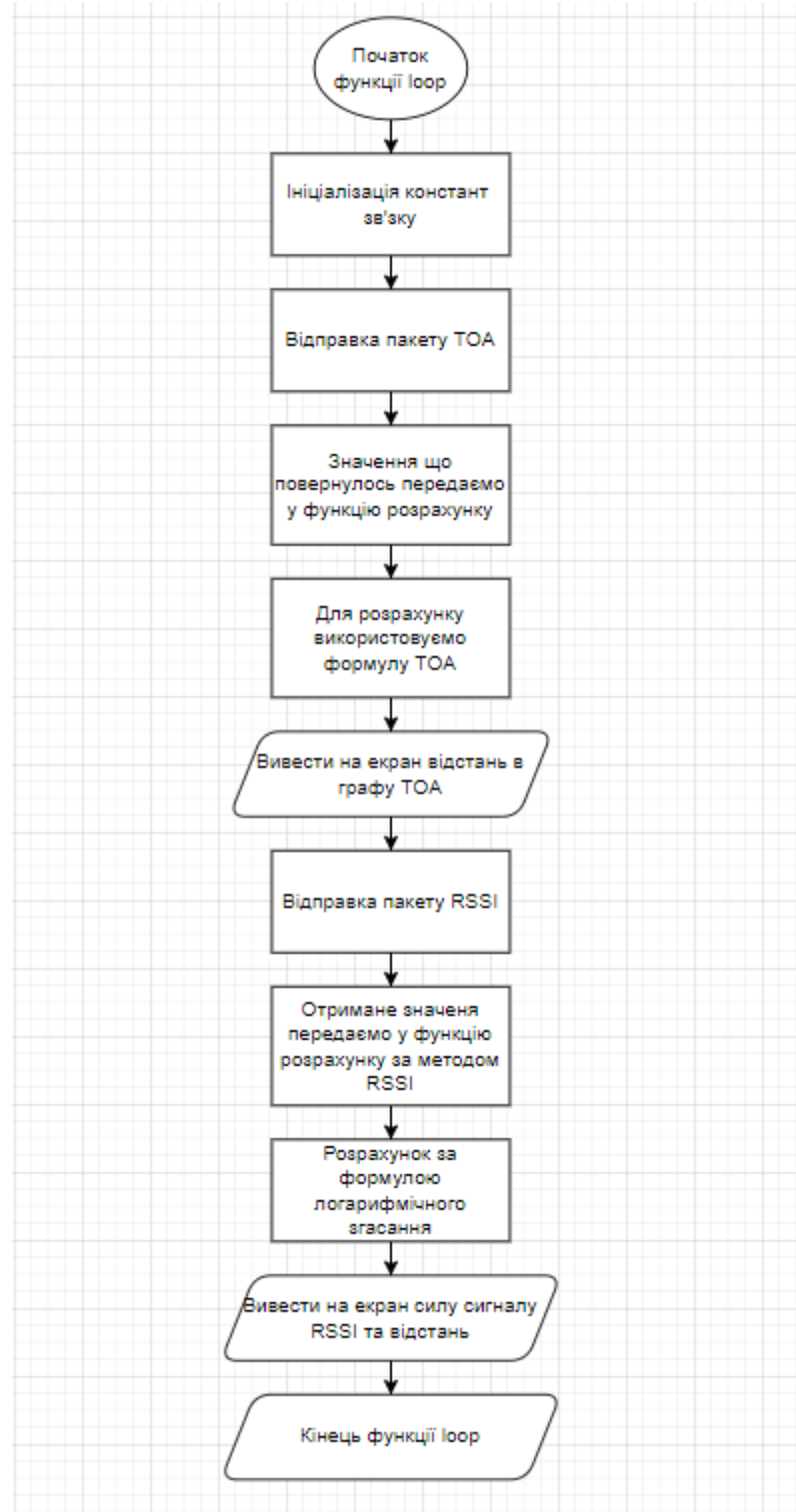


Рис. 3.2.4 Алгоритм функції `loop()`

Функція `loop()` в Arduino є циклічною, тому доречно писати весь код саме в ній, також треба додати затримку в коді, приблизно 1 секунду, щоб не спамити відправкою пакетів занадто багато, адже це впливає на розряд акумулятора.

Роздивимося функції відправки пакетів:

```
unsigned long sendWithTOA() {
    // Відправка пакету
    LoRa.beginPacket();
    LoRa.print("Hello, TOA!");
    LoRa.endPacket();

    // Розрахунок TOA
    unsigned long toa = LoRa.packetTime();
    return toa;
}

int sendWithRSSI() {
    // Відправка пакету
    LoRa.beginPacket();
    LoRa.print("Hello, RSSI!");
    LoRa.endPacket();

    // Розрахунок RSSI
    int rssi = LoRa.packetRssi();
    return rssi;
}
```

Рис. 3.2.5 Функції TOA та RSSI для відправки пакетів

Тепер розглянемо функції розрахунку дистанції на основі TOA для прикладу:



```
float calculateDistanceTOA(unsigned long toa) {  
    float speedOfLight = 299792458.0; // Швидкість світла в м/с  
    float frequency = 433E6; // Частота в Гц  
  
    // Для розрахунку на основі TOA  
    float distanceTOA = (float)toa * speedOfLight / (2 * frequency);  
    return distanceTOA;  
}
```

Рис. 3.2.6 Функція розрахунку відстані за методом TOA

Таким чином ми завершили проектування модулю приймача. Він має змогу приймати значення від передавача, декодувати та оброблювати їх для формування та виведення на екран потрібної інформації.

#### ***Розробка макету та програмної частини передавача.***

Передавач має кріпитися на БЛА, для зменшення впливу на технічно-льотні характеристики дрону модуль повинен бути максимально легким та малогабаритним. Список потрібних деталей та їх функціональність зазначені в самому початку розділу. Макетування передавача буде аналогічним до приймача за виключенням екрану, там він не потрібен.

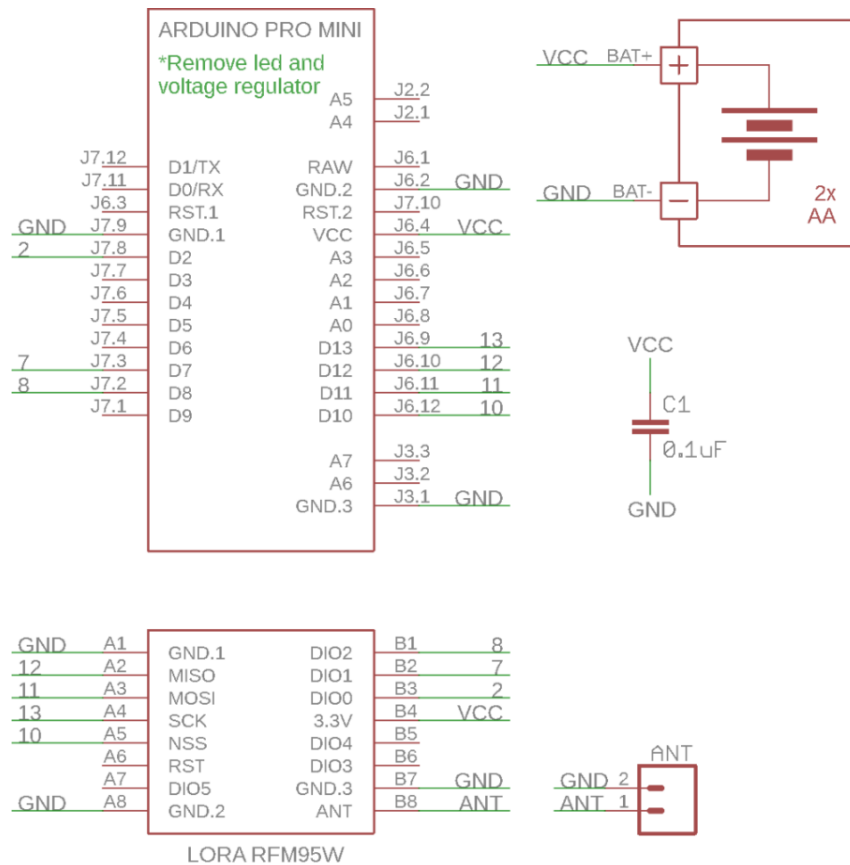


Рис. 3.2.6 Загальна електрична схема підключення передавача

Алгоритм та код для передавача дуже простий та легкий, адже все що повинен робити це приймати сигнал та повертати на приймач. Розглянемо дві основні функції прийому сигналу та відправки назад по приймача:

```

unsigned long receiveTOA() {
    if (LoRa.parsePacket()) {
        while (LoRa.available()) {
            String data = LoRa.readStringUntil('\n');
            if (data.startsWith("TOA:")) {
                return data.substring(4).toULong();
            }
        }
    }
    return 0;
}

int receiveRSSI() {
    if (LoRa.parsePacket()) {
        while (LoRa.available()) {
            String data = LoRa.readStringUntil('\n');
            if (data.startsWith("RSSI:")) {
                return data.substring(5).toInt();
            }
        }
    }
    return 0;
}

```

Рис. 3.2.7 Код прийому пакету та відправки на передавач

За результатами аналізу проекту можна сформулювати висновок, що розроблена система на базі технології Лора виявляється досить ефективною для створення бездротового зв'язку між модулями на повітряних об'єктах, зокрема БЛА. Ретельне розглядання та врахування різних аспектів, таких як методи виміру відстані, використання антен, алгоритми передачі та обробки даних, дозволяє системі працювати оптимально в умовах реального світу. Проект добре структурований та збалансований. Чітко визначені функції приймача та передавача, а також їх ролі в системі. Застосування спрямованих антен для підвищення дальності зв'язку виявляється важливим аспектом проекту, що враховує реальні умови та можливості вдосконалення системи. Результатом є деталізований проект, який обіцяє бути ефективним у реальних умовах застосування.

### 3.3. Розробка модуля автопілота

Автопілот у безпілотних літальних апаратах (БПЛА) є комплексом електронних систем, призначених для автоматичного управління польотом. Його завдання можуть включати управління висотою, курсом, швидкістю, а також виконання певних завдань і маневрів без постійної участі оператора. Основні функції автопілота включають:

1. Стабілізація польоту: Автопілот забезпечує стабільність БПЛА у повітрі, компенсуючи зовнішні фактори, такі як вітри чи турбулентність.
2. Навігація: Автопілот використовує різні сенсори та системи навігації (GPS, гіроскопи, акселерометри) для визначення розташування та напрямку.
3. Виконання програмних завдань: Оператор може заздалегідь задати маршрут або серію завдань, які БПЛА має виконати. Автопілот забезпечує автоматичне виконання цих завдань.
4. Автоматична посадка та зліт: Деякі автопілоти здатні автоматично виконувати процеси зльоту та посадки.
5. Довгострокові місії: Для довгострокових місій, таких як спостереження за територією або збирання даних, автопілот може бути ефективним засобом управління БПЛА без необхідності постійної присутності оператора.

Автопілоти суттєво покращують ефективність та точність виконання завдань, звільняючи операторів від рутинних завдань управління, що особливо важливо для довгострокових місій або в умовах, де потрібна висока стабільність польоту. Але в більшості своєї нас цікавить автопілот у випадку втрати керування з пульта, багато сучасних безпілотних літальних апаратів (БПЛА) обладнано функцією автоматичного повернення на базу (Return to Home, RTH), яка призначена для повернення БПЛА в заздалегідь задану

точку (зазвичай точку зльоту) у разі втрати зв'язку з пультом управління або іншими проблемами.

Коли зв'язок з пультом керування втрачається або виникають інші проблеми зв'язку, активується функція RTH і автопілот автоматично повертає БПЛА на базову точку. Це дозволяє запобігти втраті БПЛА та забезпечує додатковий рівень безпеки. Важливо, що з використання функції RTH необхідно заздалегідь переконатися, що точка зльоту задана коректно, і врахувати довкілля, щоб уникнути перешкод шляху повернення. Суттєвий недолік що функція автоматичного повернення на базу (Return to Home, RTH) зазвичай використовує систему глобального позиціонування (GPS) для визначення поточного розташування безпілотного літального апарата (БПЛА) та точки зльоту. Ефективність RTH залежить від точності системи GPS, надійності зв'язку та правильного налаштування параметрів функції. Якщо розглядати використання модуля в умовах роботи РЕБ то GPS втрачає свою ефективність.

Існують варіанти автоматичного повернення на базу (RTH) без використання GPS, хоча вони можуть бути менш точними та залежать від інших технологій та сенсорів. Наприклад:

- Інерційні вимірювальні пристрої (ІВП): БПЛА може використовувати гіроскопи та акселерометри для відстеження свого руху та напрямку. Однак, з часом, помилки накопичуються, і ця система може бути менш точною, ніж GPS.
- Комп'ютерний зір: Деякі сучасні БПЛА використовують камери та системи комп'ютерного зору для моніторингу навколишнього середовища. Вони можуть використовувати візуальні орієнтири для навігації та повернення на базу.

- Радіозв'язок: У деяких випадках, коли зв'язок з пультом керування відновлюється, оператор може вручну керувати поверненням на базу, навіть якщо GPS недоступний.

Хоча ці методи можуть надати альтернативу GPS, вони мають обмеження і можуть бути менш ефективними в деяких умовах.

В описі ми можемо побачити деякі варіанти на заміну GPS, вони додають самостійності БЛА у випадках втрати зв'язку адже використовуючи додаткові модулі є можливість повернути його назад, в точку відправки.

Хотів би відмітити інерційно вимірювальні пристрої (далі ІВП). це комплекс сенсорів, який вимірює прискорення та кутову швидкість БПЛА. Їхня мета полягає в тому, щоб надати інформацію про рух та орієнтацію у просторі.

ІВП зазвичай включають гіроскопи, що вимірюють кутову швидкість обертання, і акселерометри, що вимірюють прискорення вздовж різних осей.

На БЛА ІВП використовуються для стабілізації та керування польотом без GPS або у випадку обмеженого зв'язку з супутниковою системою. Ось як вони зазвичай застосовуються:

Стабілізація польоту: Гіроскопи та акселерометри в ІВП вимірюють зміни кута та прискорення, що дозволяє автопілоту дрона підтримувати стабільне положення у повітрі, компенсуючи зовнішні впливи, такі як вітер чи турбулентність.

- Оцінка положення: ІВП допомагає оцінювати поточне положення та орієнтацію дрона у просторі. Це особливо важливо в ситуаціях, коли GPS сигнал недоступний або ненадійний.
- Контроль напрямку: Гіроскопи дозволяють автопілоту точно контролювати напрям дрону, забезпечуючи точний курс польоту.
- Автономне управління: На основі даних від ІВП, дрон може приймати рішення про коригування траєкторії польоту без необхідності постійного втручання оператора.

Хоча ІВП є важливим елементом для стабілізації польоту, вони можуть бути схильні до накопичення помилок з часом. Тому, без GPS, їх застосування може бути обмежено за часом польоту. Комбінація ІВП з іншими технологіями, такими як комп'ютерний зір або радіозв'язок, може підвищити ефективність автономного керування дроном але може критично збільшити вагу або погіршити технічно-льотні характеристики.

Розглянемо декілька датчиків які вміють працювати з Ардуіно:

1. MPU6050 (гіроскоп + акселерометр): Цей модуль включає 3-осьовий гіроскоп і 3-осьовий акселерометр. Він є популярним вибором для створення ІВП на Arduino через свою доступність та хорошу продуктивність.
2. MPU9250 (гіроскоп + акселерометр + магнітометр): Цей модуль включає не тільки гіроскоп і акселерометр, але також магнітометр для вимірювання магнітного поля. Це може бути корисним для визначення орієнтації в просторі.
3. BNO055 (інтегрований датчик орієнтації): Цей датчик поєднує гіроскоп, акселерометр та магнітометр в одному корпусі, що спрощує його використання та інтеграцію з Arduino. Він також надає дані про калібрування та орієнтацію.

Хочу відмітити саме датчики сімейства MPU вони є дуже легкими, малогабаритними, простими в роботі. Є деяка різниця між MPU6050 та MPU 9250, в останньому є магнітометр, він більш точніший та має 9 напрямів осі. Хоча в проекті ми і не будемо використовувати магнітометр, задумки такої нема, інші його переваги прослужать нам тільки на користь, а точність в такому проекті дуже важлива.

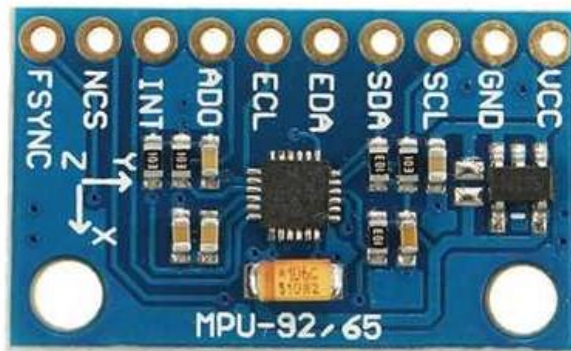


Рис. 3.3.1 Модуль MPU9250

Отже виходячи зі змісту є ідея розробити власний ІВП базований на Ардуіно, в якості головного елемента є модуль MPU9250. Він буде передавати данні кожну 0.1 секунду на контролер, але є деякі обмеження в виді кількості RAM пам'яті. На Arduino Nano/Uno/Pro Mini використовується чіп ATMEGA328p який має обмеження в 2 кБ RAM пам'яті. Розрахуємо кількість записів, за водні дані візьмемо 60 хвилин польоту:

$$\text{Кількість записів} = \frac{1 \text{ година}}{0.1 \text{ сек}} = 36000 \quad (3.12)$$

Тепер ми знаємо що на один час контролер зробить 36000 записів, порахуємо скільки оперативної пам'яті нам знадобиться, за водні дані візьмемо масив типу даних float що займає 4 байта:

$$36000 * 4 = 144000 = 144.625 \text{ кБ} \quad (3.13)$$

Ми бачимо що ОЗП не достатньо для такого обчислення навіть на одну годину польоту, не кажучи вже про більший обсяг часу. В Ардуіно є чудова можливість працювати з txt файлами, SD картами та файловою системою FAT32. Виходячи з цих даних з'являється можливість записувати на SD карту в txt файл кожне переміщення БЛА навіть з меншим розривом часу не 0.1 а 0.05 або ще менше, це дозволяє великий обсяг SD карт який має на борту гігабайти вільного простору, це зменшить похибки в польоті і буде більш точно вести БЛА назад до точки відльоту.

Список необхідних елементів:



1. Arduino NANO
2. Акумулятор
3. Кнопка включення – виключення
4. SD карта обсягом від 1 Гб
5. SD модуль
6. Модуль для ІВП MPU9250

Розглянемо схему підключення елементів до Arduino:

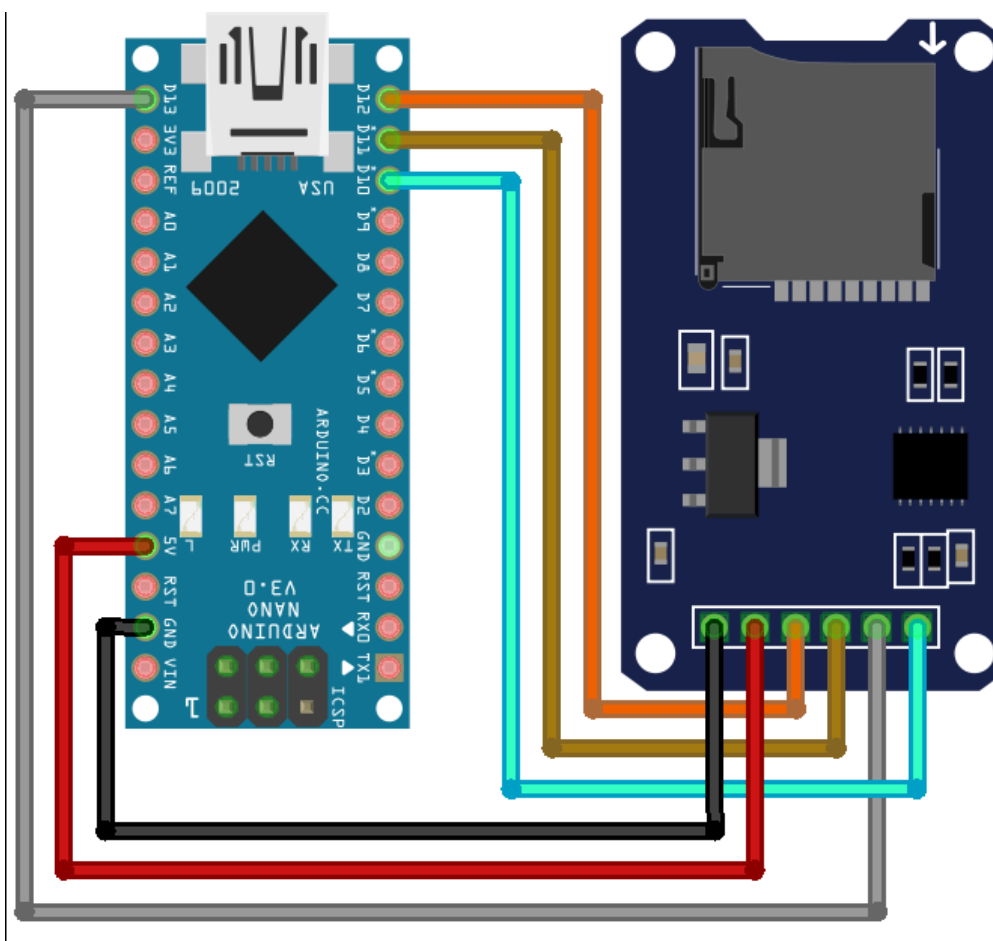


Рис. 3.3.2 Схема підключення SD карти

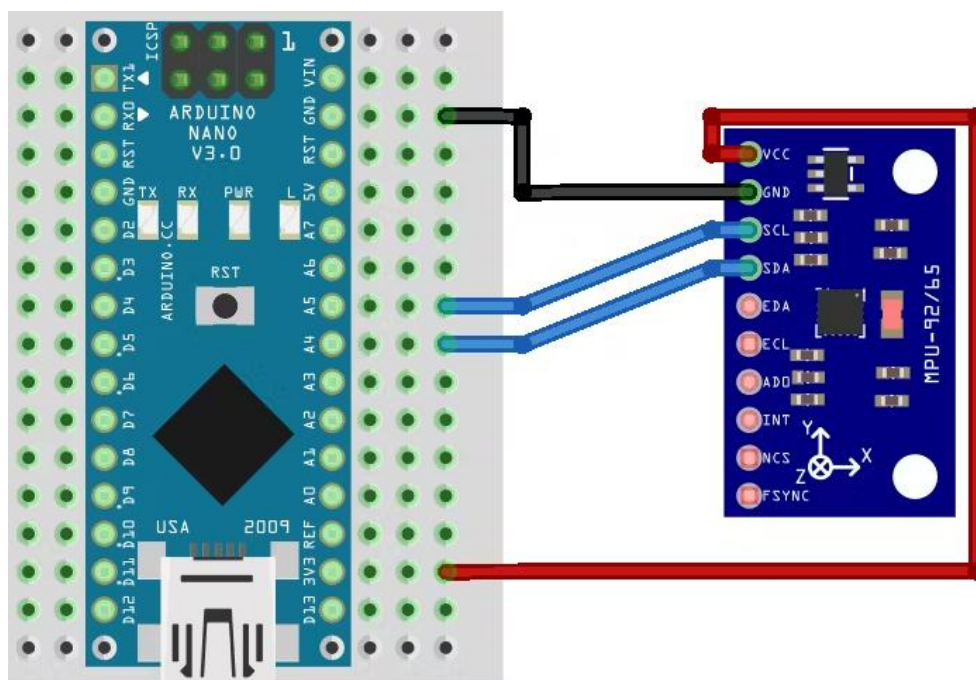


Рис. 3.3.3 Схема підключення модулю MPU9250

В схемах підключення нічого складного немає кожен модуль кріпиться на свій пін та передає відповідні данні, спосіб підключення та пайки кнопки вмикання та акумулятора беремо з першого варіанту модуля візуального виявлення, оскільки ніяких змін в цьому ланцюгу не буде. Звернемося до програмної частини код повинен працювати по такому алгоритму:

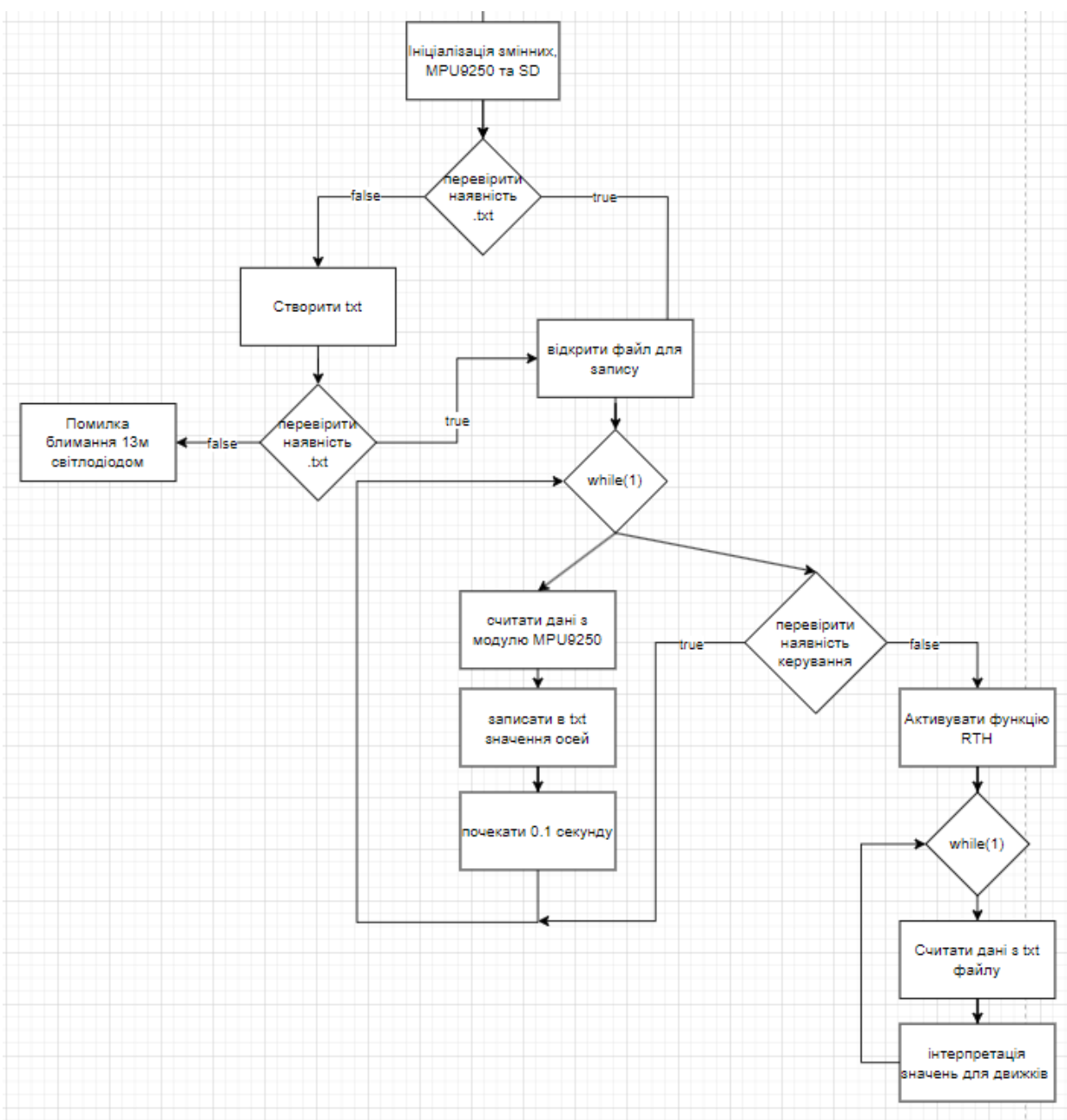


Рис. 3.3.4 Узагальнена схема роботи модуля автопілот

На схемі можна побачити цікаві новинки та нюанси які були пропущені раніше. Почнемо з початку – на SD картці повинен бути txt файл, в нього записуються значення які приходять з гіроскопу, робиться це кожну 0.1 секунду в разі відсутності доступу до файлу є можливість бликати світлодіодом що на корпусі Ардуіно, за це відповідає тринадцятий пін, це дасть змогу сигналізувати що запис модуль не працює належним чином, з низки різних причин: з SD карткою проблеми, з модулем SD поганий контакт, або програмна помилка всередині тощо. В разі доступу до файлу ми

його відкриваємо і записуємо туди значення в безкінечному циклі але з перевіркою, на блок схемі намальовано розгалуження коду, це – багатопоточність. В один момент Ардуіно може виконувати декілька команд пропускаючи їх майже одночасно і це дуже важливий момент в цьому проекті нижче в рисунку буде показано як формуються команди в процесорі Atmega при використанні цього методу. Він нам дозволить виконати перевірку втрати керування БЛА поки виконується зчитування та формування на SD картку даних без втрати часу, це неймовірно важливо, адже будь-яка непланова затримка в коді хоча б на 0.1 секунду буде впливати на похибки які і без того сформуються під час польоту. Нажаль похибки для ІВП модулів це звичайна справа, в гарних умовах їх можна пофіксувати з допомогою GPS навігації, але в нас такої можливості немає тому варто мінімізувати всі похибки. В разі перевірки наявності керування, о цій функції пізніше, ми маємо позитивну відповідь то продовжуємо виконувати тіло циклу але у разі розриву зв'язку з пультом керування активується автопілот який повертає БЛА до точки зльоту зворотнім шляхом, інвертуючи значення та повторюючи всі точки маршруту. В цьому випадку йде вже зчитування з файлу txt на SD картці та передача сили на двигуни для зворотного руху.

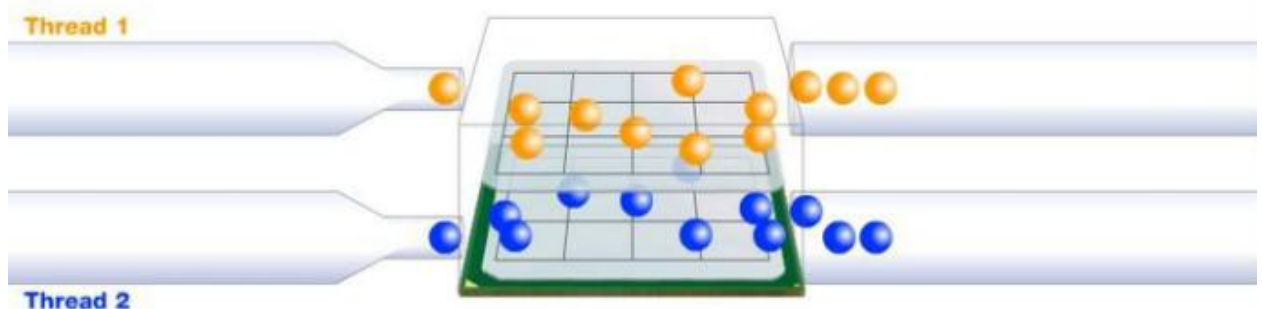


Рис. 3.3.5 Схематичне зображення роботи багатопоточності в Ардуіно  
Варто взяти до уваги що такого як легасі багатопоточності немає, зате є механіка та користувацькі бібліотеки які дозволяють виконувати команди з

іншої функції або наступні команди за кодом під час вільного часу та відсутності команд в процесорі.

```
void writeToSD(float angleX, float angleY) {  
    // Відкриття файлу для додавання даних  
    flightDataFile = SD.open("flight_data.txt", FILE_WRITE);  
    if (flightDataFile) {  
        // Запис даних у форматі "Час\tКутX\tКутY"  
        flightDataFile.print(millis());  
        flightDataFile.print("\t");  
        flightDataFile.print(angleX);  
        flightDataFile.print("\t");  
        flightDataFile.println(angleY);  
  
        flightDataFile.close();  
    } else {  
        Serial.println("Error opening flight_data.txt");  
    }  
}
```

Рис. 3.3.6 Код запису даних польоту у файл

Функція запису починає працювати одразу після включення модулю, вона записує весь шлях та викликається кожен раз після відпрацювання, такий підхід дозволить мінімізувати похибку та більш точно та коректно записувати весь шлях що пройшов БЛА.

```

void returnToHome() {
  flightDataFile = SD.open("flight_data.txt");
  if (flightDataFile) {

    flightDataFile.readStringUntil('\n');

    // Читання даних з файлу та керування двигунами
    while (flightDataFile.available()) {
      String line = flightDataFile.readStringUntil('\n');
      float time = line.substring(0, line.indexOf('\t')).toFloat();
      float angleX = line.substring(line.indexOf('\t') + 1, line.lastIndexOf('\t')).toFloat();
      float angleY = line.substring(line.lastIndexOf('\t') + 1).toFloat();

      // Тут код управління двигунами на основі кутів angleX та angleY
      // встановлення швидкості двигунів залежно від кутів

    }

    flightDataFile.close();
  } else {
    Serial.println("Error opening flight_data.txt");
  }
}

```

Рис. 3.3.7 Функція RTH

Як вже описувалось раніше функція RTH побудована на запису даних під час польоту. В момент активації цієї функції починається зчитування з файлу кутів нахилу. Але є нюанс в виді чисел, адже функція WriteToSD запише данні польоту в одну сторону, а потрібно інвертувати ці числа щоб БЛА летів в зворотному напрямку. Для інвертування використаєм множення на -1. Оскільки кількість ОЗП обмежено а дані зберігаються в масиві для їх подальшої передачі в процесор то потрібно розрахувати кількість масивів та їх величину. Є потреба в двох масивів: поточні данні для передачі в процесор та масив в якому будуть підготовлюватись наступні числа. Такій підхід зменшить час на підготовку наступних чисел. Масив `op_angles` відповідає за поточні данні які опрацьовує процесор, `prepare_angles` відповідальний за наступні числа. В момент проходження по останньому елементу в масиві `op_angles` відбувається заміщення чисел в масиві `op_angles = prepare_angles`. Виходить що під час останнього проходу по масиву наступна ітерація вже буде відбуватися з новими даними польоту. А масив `prepare_angles` почне заново з файлу зчитувати нові данні. За розрахунками кількості споживання

ОЗП можна виділити 2 масива по 100 чисел за умовами використання типу даних float. Це дозволить опрацьовувати дані польоту для передачі на двигуни та підготувати наступні 100 чисел для економії часу. В сумі 2х масивів виходить споживання 1.6 кБ що доволі затратно, адже на роботу залишається 0.4 кБ. Але оскільки в проекті використовується Ардуіно що славиться своєю модульністю є можливість використати чіп **23LC512**, підключений інтерфейсом SPI він розширює ОЗП на 512 кБ SRAM пам'яттю що розв'язує руки в програмному коді. За габаритами модуль малий, він не вплине на ТЛХ ще й навпаки, дасть змогу записувати та оброблювати більші масиви даних що позитивно вплине в цілому на результат.

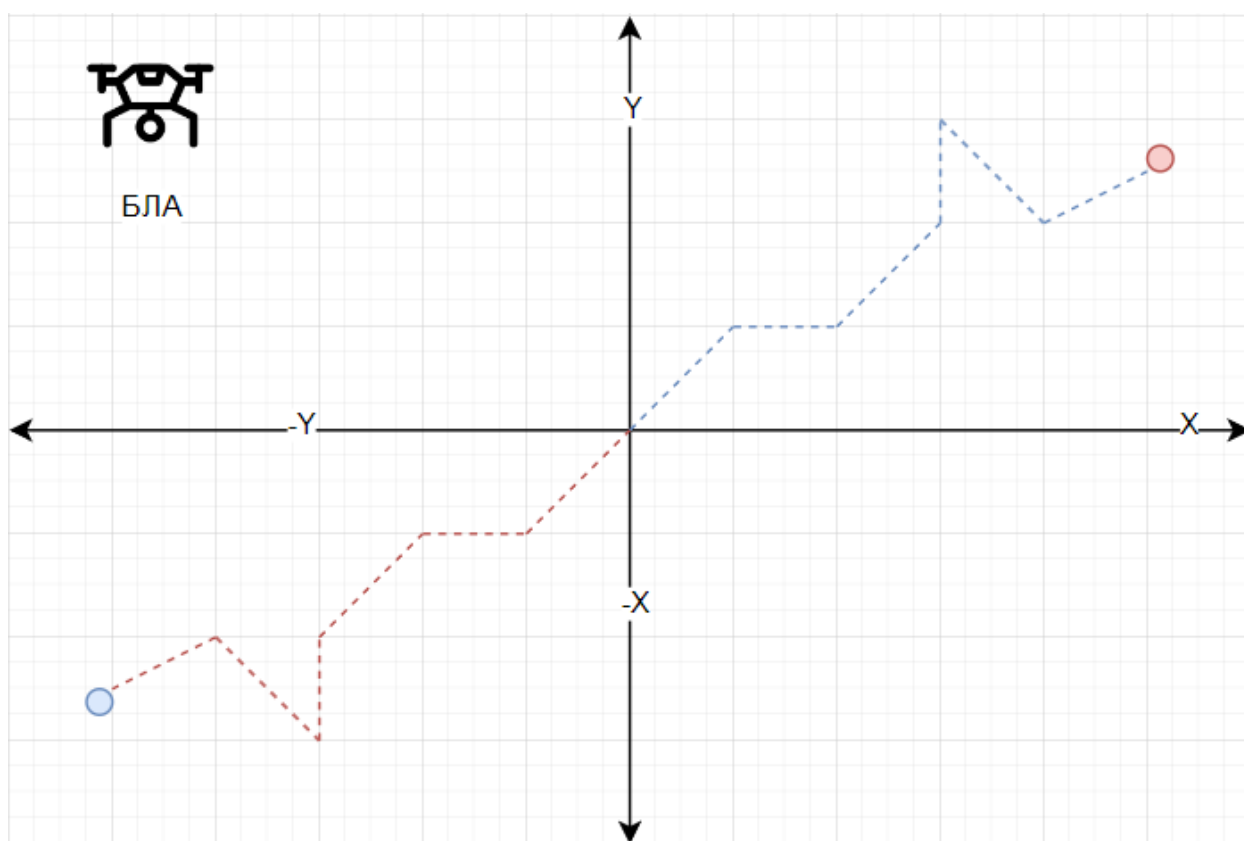


Рис 3.3.7 Схематичне зображення руху БЛА

Синім кольором позначений процес запису даних руху БЛА, по всім осям можуть бути як негативні так і позитивні всі вони записуються. Під час активації функції RTN ці дані зчитуються та інвертуються що дає змогу летіти в зворотному напрямку. В результаті використовуючи цей алгоритм

БЛА має повернутися на точку зльоту повністю повторив свій маршрут до активації RTH.

Роздивимось можливості керування двигунами. Для керування нахилом дрону на основі даних гіроскопа та регулювання потужності двигунів існують різні алгоритми та підходи. Один з найпоширеніших способів - це пропорційно-інтегрально-диференціальний (PID) контролер. У загальному вигляді формула PID-регулятора для кута нахилу може виглядати так:

$$\text{Керуючий сигнал} = K_p * \text{Похибка} + K_i * \int \text{Похибка} dt + K_d * \frac{d\text{Похибка}}{dt} \quad (3.14)$$

Де:

- $K_p, K_i, K_d$  – коефіцієнти PID, які потрібно налаштувати експериментально
- Похибка – різниця між бажаним нахилом та поточним кутом дрону (виміряним гіроскопом)
- $\int \text{Похибка} dt$  – інтегральна частина, що враховує накопичені помилки у часі
- $\frac{d\text{Похибка}}{dt}$  – диференціальна частина, яка враховує швидкість зміни помилки.

Цей сигнал керування потім використовується для регулювання потужності двигунів так, щоб задовольнити умови бажаного нахилу. Процес налаштування PID-контролера може вимагати часу та експериментів для досягнення потрібних нам результатів.

Коефіцієнти  $K_p, K_i, K_d$  в PID-контролер – це числа, які потрібно налаштувати відповідно до вимог вашої системи. Вони визначають вагу кожної з трьох компонентів PID: пропорційної, інтегральної та диференціальної.

- $K_p$  – (Коефіцієнт пропорційності) визначає, наскільки сильно керуючий сигнал повинен реагувати на поточну помилку (різницю між бажаним і поточним значенням).



- $K_i$  – (Коефіцієнт інтеграції) визначає, як інтегральна частина реагує на накопичені помилки в часі. Він виправляє системні усунення та довготривалі помилки.
- $K_d$  – (Коефіцієнт диференціації) визначає, наскільки сильно керуючий сигнал повинен реагувати на швидкість зміни помилки. Це допомагає запобігти перереагуванню системи на умови, що швидко змінюються.

При тестуванні модулю почнемо з малих коефіцієнтів і поступово будемо збільшувати для досягнення бажаного рівня роботи. Експериментуйте з різними значеннями, спостерігайте за реакцією системи та коригуйте коефіцієнти доти, доки не отримаєте бажану поведінку. Налаштування PID-контролера може бути ітеративним процесом і важливо розуміти, як кожен коефіцієнт впливає на поведінку системи.

Змоделюємо ситуацію, поточний нахил (кут) по осях X і Y дорівнює 0 градусів. Нам потрібно нахилити дрон по осі X на -15 градусів та по осі Y на 30 градусів.

Для прикладу давайте встановимо коефіцієнти PID наступним чином:

- $K_p = 1$
- $K_i = 0.1$
- $K_d = 0.01$

Візьмемо час оновлення даних – кожна секунда.

1. Початкова похибка:

$$\text{Похибка}_x = -15 \quad (3.15)$$

$$\text{Похибка}_y = 30 \quad (3.16)$$

2. Обчислення інтегральної та диференціальної компонента:

$$\int \text{Похибка}_x dt = -15 * 1 = -15 \quad (3.17)$$

$$\frac{d\text{Похибка}_x}{dt} = 0 - (-15) = 15 \quad (3.18)$$

$$\int \text{Похибка}_y dt = 30 * 1 = 30 \quad (3.19)$$

$$\frac{d\text{Похибка}_y}{dt} = 0 - 30 = -30 \quad (3.20)$$

3. Обчислення керуючого сигналу для осі X:

$$\text{Керуючий сигнал}_x = K_p * \text{Похибка}_x + K_i * \int \text{Похибка}_x dt + K_d * \frac{d\text{Похибка}_x}{dt} \quad (3.21)$$

$$\text{Керуючий сигнал}_x = 1 \cdot (-15) + 0.1 \cdot (-15) + 0.01 \cdot 15 = -15 - 1.5 + 0.15 = -16.35 \quad (3.22)$$

4. Обчислення сигналу керуючого для осі Y:

$$\text{Керуючий сигнал}_y = 1 \cdot 30 + 0.1 \cdot 30 + 0.01 \cdot (-30) = 30 + 3 - 0.3 = 32.7 \quad (3.23)$$

В арудіно існує спеціальна бібліотека яка дозволяє проводити такі обчислення PID `pidX(&InputX, &OutputX, &SetpointX, Kp, Ki, Kd, DIRECT)` достатньо ввести значення і вона поверне величину керуючого сигналу.

Для аналізу руху безпілотного літального апарата (БПЛА), розглянемо базову модель, що складається з легкої хрестоподібної несучої конструкції з чотирма гвинтами (конструкція X), встановленими на її кінцях. Передній (дв. 1) і задній (дв. 3) гвинти обертаються проти годинникової стрілки, тоді як лівий (дв. 2) і правий (дв. 4) обертаються за годинниковою стрілкою. Ця парна конфігурація з протилежними напрямками обертання усуває необхідність використання хвостового гвинта, як у звичайних гелікоптерах. Кутові швидкості для кожного пропелера позначаються індексами, що відповідають порядковим номерам роторів. Крім змінної швидкості для кожного пропелера, стрілка вгору представляє вектор швидкості і завжди вказує в даному напрямку.

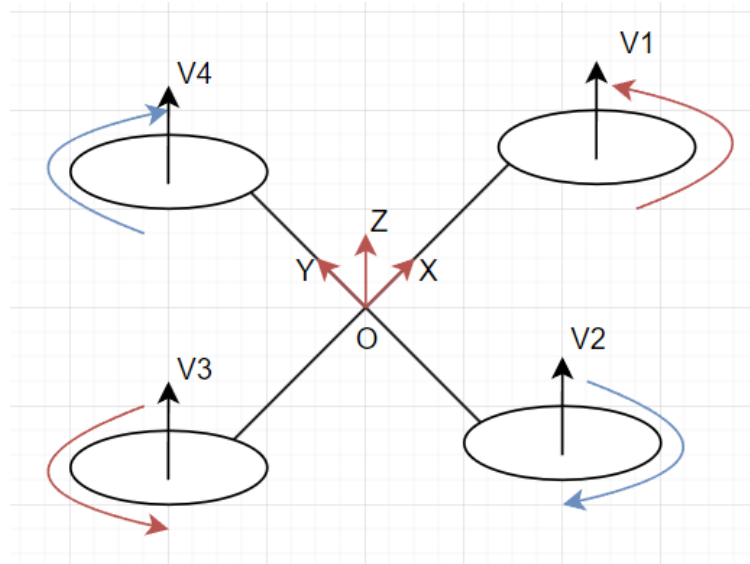


Рис. 3.3.8 Загальна модель квадрокоптера

Відповідно до моделі, показаної на рисунку 3.3.8, усі гвинти обертаються з однаковою швидкістю  $\Omega$  [рад  $\cdot$  с $^{-1}$ ], таким чином скасовуючи прискорення вільного падіння, коли дрон знаходиться в режимі висіння. Таким чином, дрон нерухомий, оскільки немає сил або моментів, які рухають його з поточного положення.

Хоча дрон має шість ступенів свободи, він оснащений лише чотирма пропелерами, що ускладнює досягнення ідеального стану всіх ступенів свободи. Усі стани можна математично розглянути та змодельовати, але насправді система керування керує чотирма станами, пов'язаними з чотирма основними рухами, що дозволяє дрону досягти певної висоти та положення. Він підвищується або знижується залежно від значення швидкості. Скористаємося такою формулою для опису математичної моделі режиму висіння:

$$U_1 = \sum_{i=1}^4 (\Omega_i + \delta_a) \cdot \text{sign}(\Omega_i) \quad (3.24)$$

де  $U_1$  – позначення режиму зависання,  $\Omega_i$  – швидкість обертання  $i$ -го ротору, с $^{-1}$ ;  $\delta_a$  – додатна додана величина, с $^{-1}$

Крен відбувається за допомогою збільшення або зменшення швидкості лівого або правого двигуна, це створює обертальний момент по відношенню до осі OX, це і дозволить БЛА обертатися

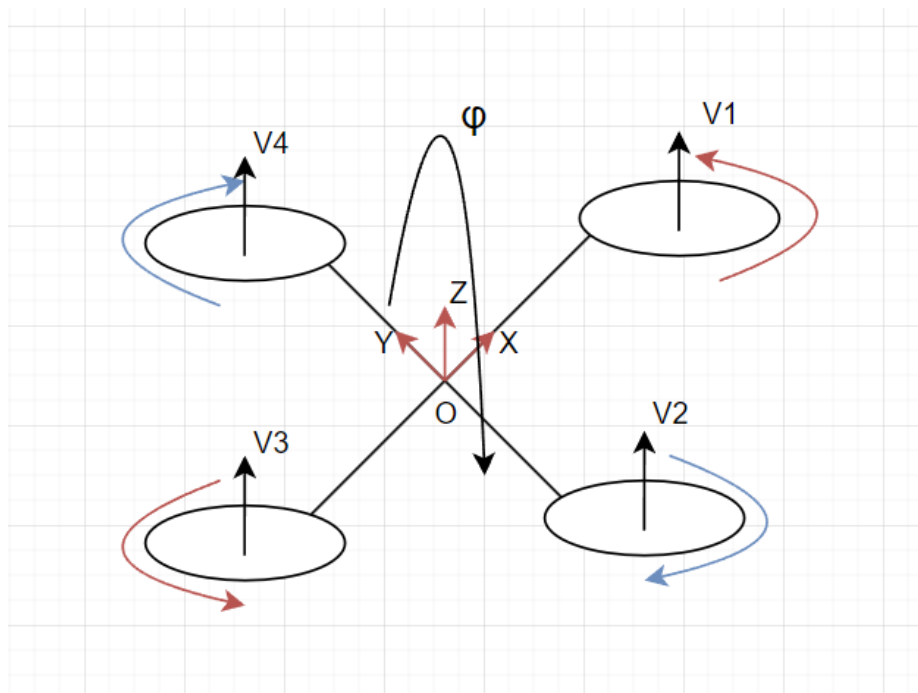


Рис. 3.3.9 Крен квадрокоптера

Математична модель режиму крену представлена за допомогою виразу:

$$U_2 = \Omega_1 - (\Omega_2 + \delta_B) + \Omega_3 - (\Omega_4 + \delta_B) \quad (3.25)$$

де  $U_2$  – позначення режиму крену;  $\delta_B$  – додатня додана величина,  $c^{-1}$

Тангаж забезпечується збільшенням або зменшенням швидкості заднього гвинта або зменшенням або збільшенням кутової швидкості переднього гвинта, що створює крутний момент навколо осі OY, що дозволяє дрону обертатися відносно осі OY. Цей режим подібний до режиму крену та призводить лише до зміни кута нахилу.

Математичну модель висоти тону можна записати у вигляді рівняння:

$$U_3 = (\Omega_1 - \delta_B) - \Omega_2 + (\Omega_3 + \delta_A) - \Omega_4 \quad (3.26)$$

де  $U_3$  – позначення режиму тангажу.

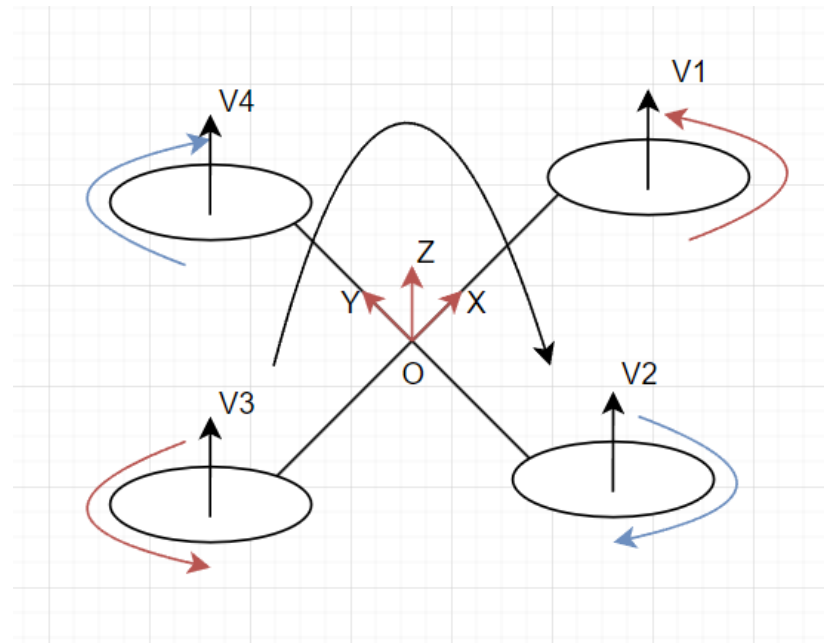


Рис. 3.3.10 Тангаж квадрокоптера

Режим ристання досягається за рахунок збільшення (або зменшення) швидкостей переднього і заднього гвинтів або зменшення (або збільшення) швидкостей лівого і правого гвинтів. Це призводить до утворення обертального моменту відносно вертикальної осі  $OZ$ . Рух в режимі ристання виникає тому, що ліві і праві гвинти обертаються за годинниковою стрілкою, тоді як передні і задні гвинти обертаються проти годинникової стрілки. Отже, якщо загальний обертальний момент незбалансований, квадрокоптер обертається навколо вертикальної осі  $OZ$ .

Математична модель режиму ристання записується у вигляді рівняння:

$$U_4 = (\Omega_1 - \delta_B) - (\Omega_2 + \delta_A) + (\Omega_3 - \delta_B) - (\Omega_4 + \delta_A) \quad (3.27)$$

де  $U_4$  – позначення режиму ристання.

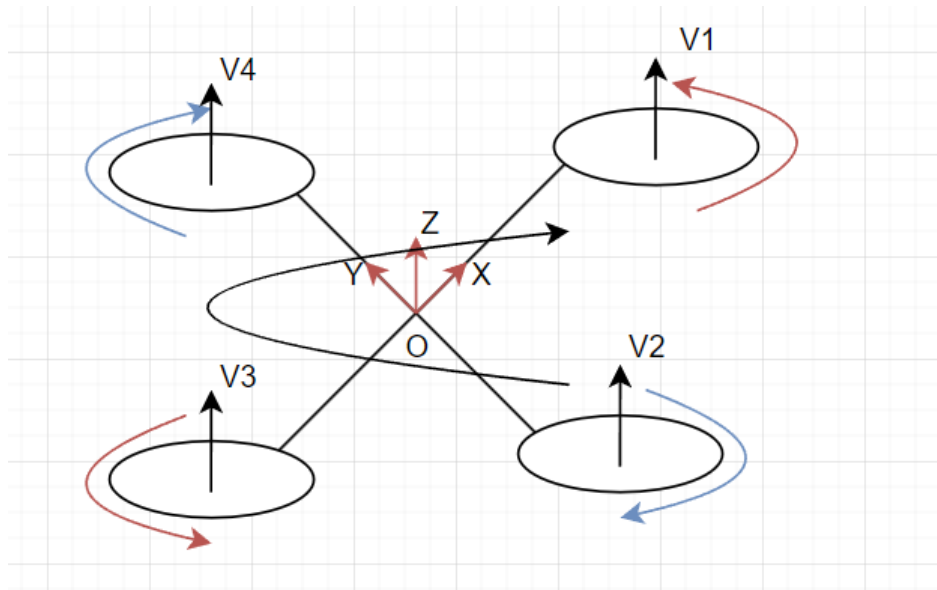


Рис. 3.3.11 Рискання квадрокоптера

Розглянемо рівняння які допоможуть сформуванати рух в правильному векторі. Динамічна система представляється за допомогою 4 векторів, взаємодія між якими визначає закони руху квадрокоптера. Положення дрону у просторі в залежності від фіксованої системи координат виражається через вектори положення і швидкості. Ці вектори включають в себе: вектор положення дрону  $\vec{P}$ , вектор поступальної швидкості дрону  $\vec{V}$ , вектор кутової швидкості дрону  $\vec{W}$  та вектор орієнтації дрону  $\vec{E}$ .

Позначимо вектор положення дрону відносно системи координат Землі як  $\vec{P}$ . Відповідне рівняння можна виразити наступним чином:

$$\vec{P} = \begin{pmatrix} P_x & X \\ P_y & Y \\ P_z & Z \end{pmatrix} \quad (3.28)$$

Індекси  $x, y, z$  – визначення проєкції вектора положення в рухомій системі координат.

Вектор поступальної швидкості швидкості  $\vec{V}$ :

$$\vec{V} = \begin{pmatrix} u & \dot{x} \\ v & \dot{y} \\ w & \dot{z} \end{pmatrix} \quad (3.29)$$

Де  $u, v, z$  – проєкції поступальної швидкості в рухомій системі координат

Вектор кутової швидкості БЛА в залежності від фіксованої системи координат  $\bar{W}$ :

$$\bar{W} = \begin{pmatrix} p \\ q \\ r \end{pmatrix} \quad (3.30)$$

Де  $p, q, r$  є проекціями обертальної швидкості в рухомій системі координат

Вектор орієнтації  $\bar{E}$  має вигляд:

$$\bar{E} = \begin{pmatrix} \varphi \\ \theta \\ \psi \end{pmatrix} \quad (3.31)$$

Де  $\psi, \theta, \varphi$  – кути Ейлера для тангажу, рискання та крену БЛА

Ці вектори представляють собою стани дрону в будь-який момент часу, і достатньо керувати ними для управління дроном. Узагальнений вектор стану, позначений як  $\bar{X}$ , об'єднує всі ці вектори і описується рівнянням:

$$\bar{X} = \begin{pmatrix} \bar{V} \\ \bar{W} \\ \bar{E} \\ \bar{P} \end{pmatrix} \quad (3.32)$$

Підбиваючи підсумок - кінематичні рівняння допоможуть змоделювати поведінку БЛА на основі його геометрії та маси, є зовнішні фактори котрі впливають на польотну модель БЛА. PID контролер використовується для регулювання величин, таких як кутова швидкість чи кутова орієнтація, для досягнення певних цілей. Обидві ці системи можуть використовуватися разом для досягнення точного та стабільного керування дроном. PID використовується для управління конкретними параметрами, тоді як кінематика вказує на взаємозв'язки між різними частинами дрона, що може враховуватися при розробці стратегій управління. Керуючий сигнал зможе передавати силу на ротори БЛА що змусить його виконувати команди польоту послідовними даними що надходять з SD картки. Для співвідношення кутів заданих та поточних використовується гіроскоп за формулами можемо прорахувати похибку та скоректувати дрон направивши

його за правильними значеннями. Реалізація в програмі буферу майбутніх обчислень допоможе мінімізувати похибку при розрахунках.

Для запуску протоколу повного керування БЛА додатковим модулем, а саме двигунами та маршрутом буде використовуватись алгоритм який передбачає вплив РЕБ на частоту та керування БЛА з допомогою пульта. У разі втрати керуючого сигналу потрібно запустити автопілот. Розглянемо можливість такої реалізації. Опираючись на дослідження в першому розділі ми маємо знання що до роботи РЕБ. Всі керуючі сигнали відправляються на БЛА з допомогою пульта який передає команди на радіочастоті. У разі глушіння певних частот відбувається втрата керування і БЛА плавно ірне сідає на землю. В інших випадках, РЕБ може перехопити керування БЛА та змусити його виконувати сигнали які надходять з джерела більш потужних передавачів. В цих обох випадках немає змоги перевірити наявність керування з точки зору оператора, адже зв'язок вже втрачено. Існують вбудовані системи автопілота які повертають БЛА назад до оператора, але майже всі такі системи працюють на GPS навігації що зводить їх ефективність в 0 при роботі в складних умовах, а саме при роботі РЕБ. Необхідно роздивитися систему яка допоможе приймачу на БЛА зрозуміти що керуючий сигнал оператора втрачений. Така система називається – контрольний пакет, в неї входять:

#### 1. Шифрування:

- Використання систем шифрування в рамках контрольного пакету може ускладнити перехоплення та несанкціоноване читання переданих даних.

#### 2. Частотні фільтри:

- Введення частотних фільтрів та спеціальних заходів для мінімізації ефективності перехоплення сигналів або завад.

#### 3. Аутентифікація:



- Вбудовання систем аутентифікації для перевірки легітимності джерела сигналу та уникнення приймання невірних чи підроблених пакетів.
4. Динамічні зміни параметрів:
- Зміна параметрів передачі, таких як частоти, потужності або модуляційні схеми, для ускладнення виявлення та блокування сигналу.
5. Використання антискрапіння:
- Вбудовання методів антискрапіння для захисту від перешкод та спроб завадження сигналу.

Для польоту БЛА на великі відстані, наприклад як 10 кілометрів, часто використовуються радіозв'язок з довгим діапазоном, такий як системи на основі технології LoRa або використання спеціальних систем радіозв'язку для дронів на великій відстані. Ось тут нам знадобляться напрацювання з частини 3.2.2 модуль радіо-маячок. З допомогою LoRa можна передавати контрольний пакет, чексуму або масив даних на приймач. У разі роботи РЕБ керуючий сигнал, та LoRa сигнал зникне. При налаштуванні програмного забезпечення в разі відсутності вхідного сигналу з LoRa передавача з контрольним пакетом необхідно запустити автопілот який описувався раніше.

```

// приклад очікуваних даних
int expectedData[] = {1, 2, 3, 4, 5};

int loop() {
    // Отримання даних через LoRa
    int receivedData[5]; // Припустимо, що дані мають розмір 5

    // Код для отримання даних через LoRa

    // Порівняння отриманих і очікуваних даних
    if (compareArrays(receivedData, expectedData, 5)) {
        return 1;
    } else {
        // Дії, якщо дані розходяться
        RTH_AP(); // Диклик функції повернення на точку взльоту
    }
}

// Функція для порівняння двох масивів
bool compareArrays(int arr1[], int arr2[], int size) {
    for (int i = 0; i < size; i++) {
        if (arr1[i] != arr2[i]) {
            return false;
        }
    }
    return true;
}

```

Рис 3.3.12 Приклад передачі та обробки контрольного пакету

Це приклад коду який дозволить показати концепцію передачі, зчитування та порівняння контрольного пакету. У ролі цього пакету виступає масив даних, в даному випадку числа, їх можна замінити на складну конструкцію певних чисел, літер та символів це дасть змогу перевіряти чи надходить не тільки радіозв'язок а й певні данні, при перехопленні РЕБ. Також можна реалізувати певне шифрування даних що передається, є можливість використати криптографічні бібліотеки, але вони будуть займати певну частину ресурсів Ардуіно що погано вплине на автономність пристрою. Як альтернативу розглянемо просте шифрування методом XOR, його має вистачити як мінімальна вимога то шифрування передачі сигналу. Саме шифрування XOR це бінарна операція, яка повертає істину (1), якщо кількість істинних бітів серед своїх операндів парна, і повертає хибу (0), якщо кількість істинних бітів непарна. У більш простому вигляді, результат XOR буде істинним, якщо кількість відмінних бітів у порівнюваних числах є непарною.

У контексті шифрування, XOR може використовуватися для зашифрування та розшифрування даних за допомогою ключа. Основна ідея полягає в тому, що вхідні біти (біти даних або біти ключа) комбінуються за допомогою XOR, щоб отримати вихідний результат. Наприклад:

- Якщо ми маємо два біти  $A = 1$  і  $B = 0$ , то  $A \oplus B = 1$  (бо кількість відмінних бітів — 1).
- Якщо  $C = 1$  і  $D = 1$  то  $C \oplus D = 0$  (бо кількість відмінних бітів — 0).

Шифрування та розшифрування XOR виглядає наступним чином:

- **Шифрування:** Зашифруйте кожен біт вхідних даних за допомогою відповідного біта ключа за допомогою XOR.
- **Розшифрування:** Також використовуйте XOR для розшифрування, оскільки  $A \oplus B \oplus B = A$

```
// Функція для шифрування даних (XOR)
void encryptData(int data[], int size) {
    // Ключ для XOR
    int key = 42;

    for (int i = 0; i < size; i++) {
        data[i] = data[i] ^ key;
    }
}

// Функція для розшифрування даних (XOR)
void decryptData(int data[], int size) {
    // Ключ для XOR (той самий)
    int key = 42;

    for (int i = 0; i < size; i++) {
        data[i] = data[i] ^ key;
    }
}
```

Рис 3.3.13 Функції шифрування та розшифрування XOR

В підсумку до цього розділу хотів відмітити процес концептуальної розробки модуля автопілот. Ми розглянули апаратну частину, з чого він складається, програмну частину реалізації певних потреб та протидії перешкод для БЛА. Сформувавши математичні рівняння що допоможуть при керуванні БЛА в рамках автопілоту та моделюванні польоту. Використання технології LoRa для передачі даних на великій відстані та впровадження контрольного пакету для визначення легітимності сигналу, шифрування XOR додає додатковий рівень безпеки до передачі даних. Такий підхід дозволяє нам вирішувати питання втрати керування, наприклад, в умовах роботи РЕБ. Автопілот та додатковий модуль для перехоплення сигналу від РЕБ і подальшого відновлення керування забезпечують надійність та стабільність. Праця підкреслює важливість інтеграції різних аспектів та заходів безпеки для забезпечення ефективності та надійності системи керування дроном. Цей підхід має великий потенціал у використанні в сучасних багатофункціональних БЛА.

**Висновок до розділу:** визначивши в першому та другому розділі проблематику використання БЛА в складних умовах, то роздивившись певні можливості, в розділі три було створено методологію розробки підвищення життєдіяльності БЛА під різні ситуації беручи до уваги умови їх використання та певні потреби. Було розроблено три різних модулів які мають переваги та недоліки котрі корелюються між собою, це змушує використовувати певний модуль під конкретну ситуацію під час планування польоту. Також це позитивно впливає на технічні характеристики БЛА не перевантажуючи його. В кожному розділі було детально розглянута схема, частини з яких створюється модуль та програмний код. Розглянуті програмні блоки та детально розібрані. Створені загальні схеми алгоритмів для демонстрації та планування модулів. Модуль візуального виявлення був розроблений виходячи з методології що наведені в розділі, та пройшовши усі тесті наразі використовується в БЛА під заплановані задачі. Висновки даного

дослідження вказують на ефективність та гнучкість розробленої методології для підвищення життєдіяльності безпілотних літальних апаратів (БЛА) в різних умовах експлуатації. Застосування трьох різних модулів, які враховують умови використання та специфічні потреби, дозволяє ефективно планувати вильоти та максимізувати використання ресурсів БЛА. Аналіз переваг і недоліків кожного модуля підкреслює важливість вибору оптимального підходу для конкретної ситуації, що в свою чергу сприяє підвищенню загальної ефективності системи.

Модуль візуального виявлення, пройшовши тестування та впроваджений у практику, підтверджує практичну застосовність розробленої методології.

Загальною метою дослідження було не лише створення ефективної системи, але й виявлення та розв'язання проблем використання БЛА в складних умовах. У результаті роботи вдалося досягти поставлених цілей, що робить цей проект значущим та перспективним у сфері розвитку безпілотних технологій.

## Висновки

Результатом магістерської роботи є розроблена методологія підвищення життєдіяльності БЛА яка за певних умов забезпечує легкий пошук втраченого БЛА чи його повернення на точку зльоту.

Актуальність теми набуває суттєвого значення при використанні модулів в складних умовах: несприятливі погодні умови, блокування зв'язку між оператором та БЛА методами РЕБ, загальна втрата керування, тощо. Головними причинами впровадження модулів в сферу безпеки БЛА є значний попит на використання та неспроможність базових, вбудованих систем, впоратись з вище написаними перешкодами та проблемами. Данні системи забезпечать більш точний та менш затратний за часом процес пошуку БЛА в складних місцевостях, в наших реаліях – під час та в зоні бойових дій.

Для вирішення задачі були виконані наступні кроки:

- Проведено огляд предметної області та аналіз БЛА, їх характеристик.
- Визначені загальні проблеми та перешкоди для керування БЛА. Виконаний огляд наявних систем РЕБ для боротьби з БЛА.
- На основі проблематики розроблені концепції протидій або усунення\спрощення результату втрати керування. Було поділено на три частини методи протидії: модуль візуального виявлення, радіо-маячок та автопілот. Кожна концепція використовується за певних умов які допомагають операторам БЛА.
- Розроблено схему та програмний код для модуля візуального виявлення. Виконано огляд його функціональності та актуальності на практиці.
- Модуль візуального виявлення пройшовши тести запроваджений у практику та поставлений у виробництво на основі розробленої методології.

- Було розглянуто принципи роботи РЕБ та їх класифікації для створення алгоритму радіо-маячка з максимальною ефективністю.
- Розроблено передачу даних між LoRa модулями та шифрування сигналу між ними. Для детекції позиціювання БЛА у просторі.
- Проведено аналіз можливостей стандартної функції RTH та визначені недоліки. Розроблено концепцію вбудованого автопілота для усунення недоліків використання в певних умовах.
- Визначено використання ІВП та розглянуто можливості керування двигунами для польоту за зазначеними координатами
- Реалізовано методику зчитування напрямку, куту нахилу, швидкості та запису на носій для подальшого використання та зчитування.

Огляд предметної області та аналіз характеристик БЛА дозволив з'ясувати загальні проблеми та перешкоди для керування. Для їх усунення та спрощення в результаті втрати керування були розроблені та впроваджені три основні модулі: модуль візуального виявлення, радіо-маячок та автопілот.

Модуль візуального виявлення, реалізований за допомогою програмної схеми та коду, успішно пройшов тести та вже впроваджений у виробництво.

Модуль радіо-маячка, побудований на технології LoRa, забезпечує довгий зв'язок та конфіденційність передачі даних між оператором та БЛА. Використання шифрування та ефективний алгоритм робить цей модуль надійним засобом відстеження та локації БЛА в режимі реального часу.

Концепція вбудованого автопілота вирішує проблеми стандартної функції Return to Home (RTH), дозволяючи ефективно повертати БЛА та точно виконувати програмовані маршрути. Використання ІВП та можливостей керування двигунами надає стабільний політ у заданому напрямку.

Загалом, ця інтегрована система модулів стає відповіддю на проблеми, які можуть виникнути в умовах експлуатації БЛА, забезпечуючи надійність та покращену функціональність. Розроблені рішення можуть знайти своє застосування в різних областях, зокрема в аерофотозйомці, військовому використанні та дослідженнях.



## СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Класифікація безпілотних літальних апаратів / О. І.Тимочко, Д. Ю. Голубничий, В. Ф. Третяк, І. В. Рубан. // Системи озброєння і військова техніка. –2007. – №1. – С. 30–69.
2. Боротьба з безпілотними літальними апаратами (за досвідом проведення ООС(раніше АТО) [Електронний ресурс] // Режим доступу до ресурсу: <https://jurkniga.ua/contents/borotba-z-bezpilotnymy-litalnymy-apatamy-za-dosvidom-provedennia-oos-ranishe-ato.pdf>
3. ArduPilot [Електронний ресурс] // ArduPilot Dev Team. – 2016. – Режим доступу до ресурсу: <http://ardupilot.org/ardupilot/>
4. The Free Dictionary [Електронний ресурс] – Режим доступу до ресурсу: <http://www.thefreedictionary.com/Unmanned+Aerial+Vehicle>.
5. Класифікація безпілотних літальних апаратів [Електронний ресурс] Режим доступу до ресурсу: <https://openarchive.nure.ua/server/api/core/bitstreams/878899d8-b7a7-4481-af22-9835c0748ba0/content>
6. Classification of the Unmanned Aerial Systems [Електронний ресурс] // Режим доступу до ресурсу: <https://www.e-education.psu.edu/geog892/node/5>
7. MPU-9250 Product Specification Revision 1.1 [Електронний ресурс] //
8. InvenSense Inc.. – 2016. – Режим доступу до ресурсу: <https://www.invensense.com/wp-content/uploads/2015/02/PS-MPU-9250A-01-v1.1.pdf>
9. iNEMO inertial module: 3D accelerometer, 3D gyroscope, 3D magnetometer [Електронний ресурс] // life.augmented. – 2015. – Режим доступу до ресурсу: <https://www.st.com/resource/en/datasheet/DM00103319.pdf>.
10. Real Time Kinematic [Електронний ресурс] // Wikipedia. – 2018. – Режим доступу до ресурсу: [https://ru.wikipedia.org/wiki/Real\\_Time\\_Kinematic](https://ru.wikipedia.org/wiki/Real_Time_Kinematic)

11. Fay G. Derivation of the aerodynamic forces for the mesicopter simulation / G.Fay. – 2001.
12. LoRa Configuration [Електронний ресурс] / Режим доступу до ресурсу: <https://meshtastic.org/docs/configuration/radio/lora>
13. Interfacing SX1278 (Ra-02) LORA Module with Arduino [Електронний ресурс] / how2electronics – 2022 – Режим доступу до ресурсу: <https://how2electronics.com/interfacing-sx1278-lora-module-with-arduino/>
14. Global positioning system / Pratap Misra // 2010
15. World's Smartest Precision Landing Solution [Електронний ресурс] // flytbase. – 2018. – Режим доступу до ресурсу: <https://flytbase.com/precision-landing/>
16. Cortex-M3 Technical Reference Manual [Електронний ресурс] // ARM Limited. – 2005. – Режим доступу до ресурсу: [http://infocenter.arm.com/help/topic/com.arm.doc.ddi0337e/DDI0337E\\_cortex\\_m3\\_r1p1\\_trm.pdf](http://infocenter.arm.com/help/topic/com.arm.doc.ddi0337e/DDI0337E_cortex_m3_r1p1_trm.pdf).
17. Радіо LoRa [Електронний ресурс]. — Режим доступу: <https://en.wikipedia.org/wiki/LoRa>
18. Радіо LoRa на чіпі SX1278 [Електронний ресурс]. — Режим доступу: <https://arduino.ua/prod3299-modem-lora-na-chipe-sx1278>
19. Олійник В.Ф. Приймання та оброблення сигналів. 2004.

**ДОДАТОК А**