

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»

Інститут інформатики та радіоелектроніки
Факультет комп'ютерних наук та технологій
(повне найменування інституту, факультету)

Кафедра комп'ютерних систем та мереж
(повне найменування кафедри)

Пояснювальна записка

до дипломного проекту (роботи)

магістра

(ступінь вищої освіти (освітній ступінь))

на тему СИСТЕМА ПОЛИВУ РОСЛИН НА БАЗІ АВТОНОМНОГО ДРОНУ

Виконав: студент 2 курсу, групи КНТ-511м
спеціальності _____

123 «Комп'ютерна інженерія»

(код і найменування спеціальності)

Освітня програма (спеціалізація) _____

«Комп'ютерні системи та мережі»

Тітюєв Олександр Саїдович

(прізвище та ініціали)

Керівник Тягунова М.Ю.

(прізвище та ініціали)

Рецензент Самойлик С.С.

(прізвище та ініціали)

м. Запоріжжя
2022 рік

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»
(повне найменування вищого навчального закладу)

Інститут, факультет інформатики та радіоелектроніки, комп'ютерних наук і технологій

Кафедра «Комп'ютерні системи та мережі»

Ступінь вищої освіти (освітній ступінь) магістерський

Спеціальність 123 Комп'ютерна інженерія

(код і найменування)

Освітня програма (спеціалізація) Комп'ютерні системи та мережі

(назва освітньої програми (спеціалізації))

ЗАТВЕРДЖУЮ

Завідувач кафедри Кудерметов Р.К.

“ ” 2022 року

ЗАВДАННЯ
НА ДИПЛОМНИЙ ПРОЕКТ (РОБОТУ) СТУДЕНТА

Тітюєва Олександра Саїдовича

(прізвище, ім'я, по батькові)

1. Тема проекту (роботи) Система поливу рослин на базі автономного дрону

керівник проекту (роботи) Тягунова Марія Юріївна, к. т. н., доцент

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом вищого навчального закладу від “07” листопада 2022 року №367

2. Строк подання студентом проекту (роботи) 7 жовтня 2022 року

3. Вихідні дані до проекту (роботи) мобільна роботизована система та дрон. Необхідно розробити систему, здатну автономно виконувати моніторинг стану ґрунту рослин та здійснювати керування вірно спланованого процесу їхнього поливу.

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

1) Аналіз предметної області;

2) Проектування системи;

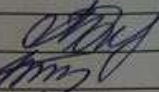
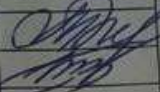
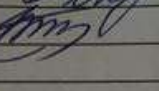
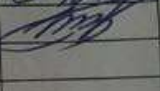
3) Апаратне забезпечення системи;

4) Налаштування роботи системи.

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

Загальна структура схеми роботизованої системи; Загальний алгоритм функціонування дрону; Загальний алгоритм функціонування док-станції; Структурна схема дрону; Структурна схема док-станції; Електрична схема дрону; Електрична схема докстанції;

6. Консультанти розділів проекту (роботи)

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	прийняв виконане завдання
1-4	Тягунова М.Ю., к.т.н., доцент		
Нормоконтроль	Щербак Н.В., старший викладач		

7. Дата видачі завдання 30.09.2022 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проекту (роботи)	Строк виконання етапів проекту (роботи)	Примітка
1	Аналіз схожих роботизованих систем та підходів щодо їх розробки	05.10.2022 р.	
2	Постановка завдання	20.10.2022 р.	
3	Розробка вимог до системи	25.10.2022 р.	
4	Розробка загальної структурної схеми системи	29.10.2022 р.	
5	Розробка структурної схеми складових системи	05.11.2022 р.	
6	Розробка алгоритму функціонування дрону	12.11.2022 р.	
7	Розробка алгоритму функціонування док- станції	16.11.2022 р.	
8	Вибір апаратного забезпечення системи	24.11.2022 р.	
9	Налаштування роботи системи	28.11.2022 р.	
10	Аналіз результатів роботи системи	30.11.2022 р.	
11	Оформлення пояснювальної записки	06.12.2022 р.	
12	Проходження нормоконтролю	07.12.2022 р.	

Студент

(підпис)

О.С. Тітюєв

(ініціали та прізвище)

Керівник проекту (роботи)

(підпис)

М.Ю. Тягунова

(ініціали та прізвище)

РЕФЕРАТ

ПЗ: 81 с. 41 рис., 25 джерел.

АВТОМАТИЗАЦІЯ, РОБОТИЗОВАНА СИСТЕМА. ДРОН, АВТОМАТИЗАЦІЯ НА ОСНОВІ ШТУЧНОГО ІНТЕЛЕКТУ, НЕЙРОННА МЕРЕЖА, МІКРОКОНТРОЛЕР, AVR, RASPBERRYPI, OPENCV, RASPBERRY PICO W.

Об'єкт розробки – система поливу рослин на базі автономного дрону.

Мета роботи – автоматизація процесу управління роботою дрону на базі розробленого апаратно-програмного комплексу роботизованої системи.

У першій частини магістерської роботи проведено аналіз роботизованих систем та польових дронів, а також описано основні підходи щодо створення схожих на аналоги роботизованих систем.

У другій частини магістерської роботи було окреслено вимоги до створюваної системи, спроектовано загальну структурну схему роботизованої системи та її складових, розроблено алгоритми функціонування.

У третій частини розроблено апаратну складову роботизованої системи, дрону та док-станції з доступних на ринку модулів та пристроїв.

У четвертій частині описано процес налаштування системи та встановлення необхідного програмного забезпечення, що дозволяє системі працювати в автоматичному режимі й виконувати полив рослин, які заздалегідь вказані користувачем.

Автономна робота системи можлива у тому числі і завдяки розробленому модулю розпізнавання рослин на базі нейронної мережі, вірність роботи якої перевірено у програмному продукті Unreal Engine5.

Розроблена система поливу рослин на базі автономного дрону є повністю працездатною та відповідає усім висунутим до неї вимогам.

ЗМІСТ

РЕФЕРАТ	3
ВСТУП	6
1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ	8
1.1 Загальні відомості про роботизовані системи.....	8
1.2 Аналіз роботизованих систем та підходів щодо їх розробки	11
1.3 Постановка завдання.....	23
2 ПРОЄКТУВАННЯ СИСТЕМИ.....	25
2.1 Вимоги до системи.....	25
2.2 Розробка загальної структурної схеми системи.....	26
2.3 Розробка структурної схеми дрону	28
2.4 Розробка структурної схеми док-станції	30
2.5 Функціонування системи	31
2.5.1 Алгоритм функціонування системи	33
3 АПАРАТНЕ ЗАБЕЗПЕЧЕННЯ СИСТЕМИ	36
3.1 Вибір стеку технологій	36
3.2 Блок керування дрону	37
3.3 Сенсорна система	39
3.3.1 Пристрій відстеження руху	40
3.3.2 Датчик температури та вологості	43
3.3.3 Модуль виявлення та визначення дальності світла.....	44
3.3.4 Камера	45
3.4 Актуаторна система	45
3.4.1 Двигуни системи	45
3.4.2 Контролери	46
3.4.3 Комунікаційні пристрої	47
3.5 Системи живлення	48
3.6 Допоміжні пристрої апаратної складової дрону	50
3.7 Блок керування док-станції.....	51

3.8 Розробка електричної схеми компонентів системи.....	52
3.8.1 Розробка електричної схеми дрону	52
3.8.2 Розробка електричної схеми док-станції	54
4 НАЛАШТУВАННЯ РОБОТИ СИТЕМИ	57
4.1 Вибір стеку технологій	57
4.2 Встановлення Raspberry Pi OS Lite	57
4.3 Встановлення бібліотеки OpenCV.....	61
4.4 Встановлення бібліотеки WiringPi	65
4.5 Налаштування системи розпізнавання зображень.....	66
4.6 Встановлення FTP-серверу	67
4.7 Налаштування віддаленої середи розробки SSH	68
4.8 Особливості роботи системи.....	70
4.9 Результат роботи системи	74
ВИСНОВКИ.....	78
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ	79

Перелік графічного матеріалу:

Плакат 1 – Загальна структурна схема роботизованої системи

Плакат 2 – Зальний алгоритм функціонування дрону

Плакат 3 – Зальний алгоритм функціонування док-станції

Плакат 4 – Структурна схема дрону

Плакат 5 – Структурна схема до-станції

Плакат 6 – Електрична схема дрону

Плакат 7 – Електрична схема док-станції

ВСТУП

Спостерігаючи стрімкий темп розвитку алгоритмів штучного інтелекту в останні десятиріччя та впровадження їх у повсякденну діяльність людини значною мірою змінило взаємовідносини між людьми, між людьми та інтелектуальними комп'ютерними системами й комп'ютерних систем між собою. Після того, як у перші роки 21 століття в період значного прогресу в галузі штучного інтелекту стала розробка реалізація самонавчаючої нейронної мережі. Алгоритм, реалізація якого, сприяє навчанню на основі даних за участю вчителя або без його втручання, передбачала великі перспективи і прикувала інтерес багатьох вчених та інженерів, завдяки чому відбулись інші найважливіші досягнення. Це стала розробка алгоритмів навчання з підкріпленням на основі генеративних моделей, які можуть генерувати нові приклади із заданого класу, що допомагає вивчити складну поведінку з дуже невеликої кількості даних, наприклад, за їх допомогою, нейронної мережі мають змогу навчитися керувати автомобілем всього за 20 хвилин досвіду водіння.

На додаток до цього важливим досягненням останніх років є реалізація так званих глибоких нейронних мереж, згорток і фільтрів, для завдань комп'ютерного зору, що дозволяють розпізнавати, і прогнозувати об'єкти в зображенні з камери. Крім обробки даних з матричної сітки камери, використання інструментів згортки глибинного машинного навчання підходить для завдань обробки мови носія, а саме, вилучення інформації та відповіді на питання. Стає можливим рішення задач розпізнавання мовлення, таких як автоматичне розпізнавання мовлення (ASR) та ідентифікація мовця (SID). На сьогоднішній день існує безліч документацій з реалізації таких алгоритмів, а також реалізації архітектури готових систем на базі нейронних мереж з відкритим кодом, що дозволяє реалізувати свій власний продукт, призначений для певного різновиду

завдань автоматизації на основі штучного інтелекту, як основного підходу. Таким чином, обрана тема магістерської роботи є актуальною.

Метою роботи є автоматизація процесу управління роботою дрону на базі розробленого апаратно-програмного комплексу роботизованої системи.

Для досягнення поставленої мети необхідно виконати наступні завдання:

- провести аналіз архітектурних та функціональних рішень різних роботизованих систем;
- спроектувати роботизовану систему та розробити алгоритм її функціонування;
- обрати необхідне апаратне забезпечення системи;
- зібрати та налаштувати роботу системи;
- проаналізувати отримані результати.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

У цьому розділі проводиться увага загальному поняттю роботизованих систем. Наведена основна класифікація типів робототехнічних систем з поточним аналізом й розбором підходу розробки схожих систем та концепцій, що надасть змогу визначити вибір оптимального підходу щодо розробки роботизованої системи призначеної для вирішення поставлених в роботі завдань. Детальний аналіз таких систем дасть можливість на початковому етапі сформулювати чіткі вимоги та розуміння щодо підходу розроблюваної платформи робота та апаратної й інформаційної системи, що у свою чергу дозволить мінімізувати витрачений час розробки й кошти.

1.1 Загальні відомості про роботизовані системи

Під концепцією роботизованої системи прийнято розуміти об'єктну композицію ряду технічних дисциплінарних рішень: прикладну математику, лінійну алгебру, машинобудування, електротехніку, розробка програмного забезпечення. Результати цих рішень надають змогу створити перепрограмовану механічну систему, керовану комп'ютером, яка може сприймати й реагувати на атрибути навколишнього середовища, виконуючи призначені завдання з деяким ступенем автономності.

Проаналізувавши доступні джерела інформації щодо існуючого різноманіття популяції роботів [13,15], які були розроблені вздовж минулих років, можна виокремити, насамперед, роботи-маніпулятори та автономні мобільні роботи, які сконструйовані подорожувати повітрям, землею чи морем. Ці роботи мали змогу імітувати людей чи тварин або мати нові топології для виконання бажаних завдань.

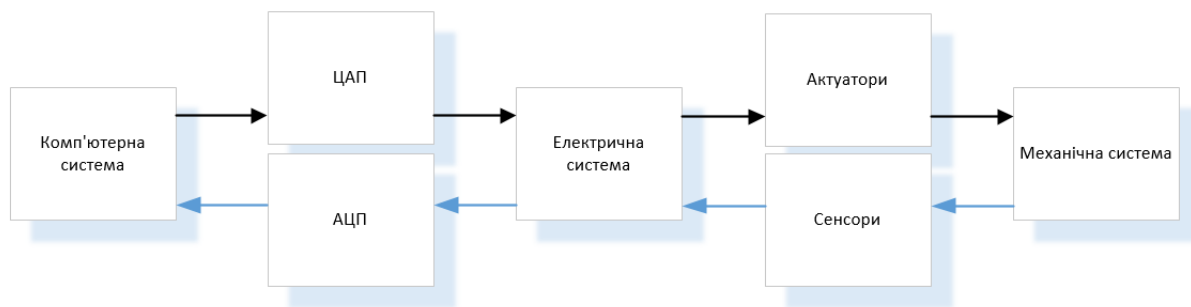


Рисунок 1.1 - Схема потоку сигналу для типової мехатронної системи

Однак, незважаючи на ці відмінності на сьогоднішній час майже всі роботизовані системи оснащені приводами, які за своєю технологічною реалізацією, мають в собі синергетичне об'єднання вузлів точної механіки з електротехнічними компонентами, що носить назву як мехатронна система. Асортимент задач, розроблених для мехатронних систем у робототехніці величезний, але є особливості притаманні для більшості, якщо не всім таким системам. Рисунок 1.1 ілюструє схематичне зображення потоку сигналу для типової мехатронної системи. Сигнали від цифрових комп'ютерних систем до аналогових, електричні системи проходять через цифро-аналоговий перетворювач і ці команди реалізуються на приводах, що з'єднують електричні та механічні системи, а механічні системи складаються з фізичних систем, які взаємодіють зі своїм оточенням.

Маючи на увазі типову структуру й поняття концепції роботизованих систем зробити визначення роботам вже постає не настільки очевидним у простоті заданням. Існує безліч визначень, які намагаються відокремити робота від автоматизації. Класифікувати їх також непросто. Є багато способів класифікації роботів. Їх можна класифікувати на те, як вони виконують завдання, на основі їх застосування, на основі того, як вони рухаються і таке інше. Було вирішено дотримуватися класифікації роботів на основі найбільш популярних способів, а саме, щодо їх застосування або їх пересування. У роботизованих системах виокремимо три основні типи. В залежності від сфери застосування розрізняють:

- маніпуляційні роботизовані системи;
- мобільні роботизовані системи;
- роботизовані системи збору та управління даними.

Маніпуляторні роботизовані системи найчастіше використовується в обробній промисловості. Ці системи складаються з безлічі маніпуляторів, які мають в собі від 4 до 6 вісей з різним ступенем свободи. Вони можуть виконувати кілька різних функцій, включаючи зварювання, вантажно-розвантажувальні роботи та видалення матеріалу. Як приклад, маніпуляторний цех збору автомобілів у компанії Tesla.

Мобільна роботизована система зазвичай є автоматизованою платформою, яка перевозить товари з одного місця в інше. Рух системи також може контролюватись автономно й може мати заздалегідь запрограмоване місце призначення, звідки система може завантажуватись або завантажувати автоматично. Такі робототехнічні комплекси також використовуються в основному в промислових цілях для перенесення інструментів та запасних частин на склад. Ще одне застосування аналогічних робототехнічних систем - це ферми, де їх можна використовувати для тягової техніки для оранки полів або транспортування сільськогосподарської продукції. Мобільні роботи також використовуються нафто- та газовидобувними компаніями для розвідки нафти та газу на шельфі та рятувальними компаніями для пошуку затонулих суден. Мобільність цих роботів може бути виражена у польоті, плаванні або пересуванні по землі. Як приклад, марсохід, польові роботи, підводні розвідуючі дрони.

Роботизована система збору та управління даними використовується для збору, обробки та передачі важливих даних, що використовуються для генерації різних сигналів. Як правило, призначена для дій, що вимагають меншої участі людини, керуюча система генерує сигнали, які можна використовувати для управління іншими роботами. Роботизована система збору та управління даними також використовується для систем САПР, що використовуються в інженерних та бізнес-процесах. Багато мобільних

роботизованих систем, особливо безпілотні судна, що використовуються для дослідження морського дну, оснащені роботизованою системою збору та управління даними для отримання важливої інформації та відправки її назад на берег у вигляді сигналів. Як приклад, космічна обсерваторія Gaia.

1.2 Аналіз роботизованих систем та підходів щодо їх розробки

Існує багато технік, які дозволяють прискорити процес розробки пристрою або продукту, однією з них є аналіз існуючі готових систем, які в собі мають вже готове рішення і виконують такі самі задачі або віддалено схожі, але які все-таки заслуговують уваги. Ще одним способом прискорення є використання існуючих апаратних платформ та програмних засобів, цей метод також може послугувати причиною модульності платформи на виході продукту, що може значно спростити обслуговування пристрою.

Розглянемо сільськогосподарські роботи. Основна мета таких робіт - автоматизація сільськогосподарської монотонної або шкідливої роботи, де це тільки можливо реалізувати технологіями з алгоритмами. Сьогодні основна сфера застосування робіт у сільському господарстві перебуває на етапі збирання та поливу врожаю. Нові, більш дорого вартісні застосування робіт або дронів у сільському господарстві включають боротьбу з бур'янами, посів насіння, збирання врожаю, моніторинг навколишнього середовища та аналіз ґрунту.

Однією з фундаментальних задач для мобільної роботизованої системи є отримання даних до чисельної системи точного знання положення транспортного засобу, роботу, дрону. У пошуках рішення задачі дослідники та інженери розробили різноманітні системи, датчики та методи позиціонування. Яскравий прикладом може послугувати серія польових чотириколісних робіт BoniRob, результат спільної праці компаній Amazone та Bosh,

призначених для роботи з гуртом та рослинами. Платформа BoniRob – це багатоцільовий робот від Bosch робототехніка DeepField. BoniRob розроблений для додатків у точному землеробстві, тобто для механічного знищення бур'янів, селективного обприскування гербіцидами, а також для моніторингу рослин і ґрунту. Робот забезпечує кріплення для встановлення різних інструментів для конкретних завдань. BoniRob оснащений чотирма колесами, якими можна керувати незалежно одне від одного, що забезпечує гнучкість пересування та навігації по пересіченій місцевості. На рисунку 1.2 показана сенсорна система позиціювання, розташування всіх датчиків, встановлених на BoniRob. Вони надають візуальні дані, глибину, 3D-лазер, GPS та дані одометрії.

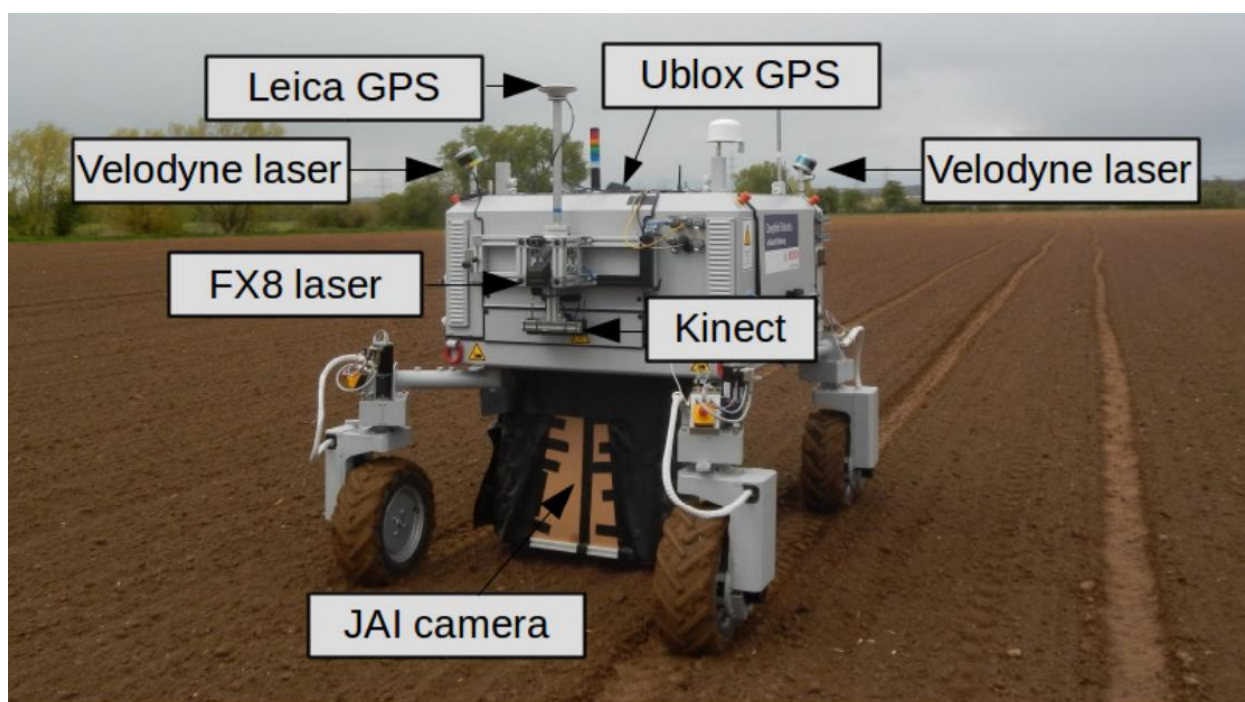


Рисунок 1.2 - Польовий робот BoniRob з усіма датчиками

Камера JAI AD130GE встановлена всередині кожуха під шасі робота і дивиться прямо вниз є багато спектральним датчиком бачення з двома ПЗЗ(прилад з зарядовим зв'язком, англійською CCD matrix) на основі призми, який надає дані зображення трьох смуг у візуальному спектрі (RGB) і спостерігає одну смугу ближнього інфрачервоного (NIR) спектру. Оскільки

листя рослин демонструє високу відбивну здатність у NIR-спектрі через вміст хлорофілу, NIR-канал корисний для відділення рослинності від ґрунту та інших фонових даних. Мозаїчна кольорова ПЗЗ-матриця Bayer і монохромна ПЗЗ-матриця камери JAI, обидва розміром 1/3 дюйма, забезпечують роздільну здатність зображення 1296×966 пікселів відповідно.

Velodyne VLP16 Puck 3D-сенсор лідар забезпечує вимірювання відстані та коефіцієнта відбиття за допомогою обертової колони з 16 лазерних діодів. Таким чином, датчик має 16 площин сканування, кожна з яких забезпечує 360° горизонтальне поле зору та 30° вертикальне поле зору з горизонтальною роздільною здатністю $0,4^\circ$ та вертикальною роздільною здатністю приблизно 2° . Датчик забезпечує вимірювання до діапазон 100м при частоті 20Гц для повного сканування 360° . На рисунку 1.2 показано фрагмент одного сканування, типові дані виглядають у виді тривимірної хмари точок, на рисунку сканування показує двох людей, які йдуть поруч із роботом. BoniRob оснащено двома такими датчиками, один у передньому правому верхньому куті шасі, а інший у задньому лівому верхньому куті. Вони трохи нахилені до землі, щоб краще виявляти об'єкти поблизу робота. Основне призначення датчиків Velodyne – надавати дані для створення 3D-карти навколишнього середовища, та виявлення перешкод.

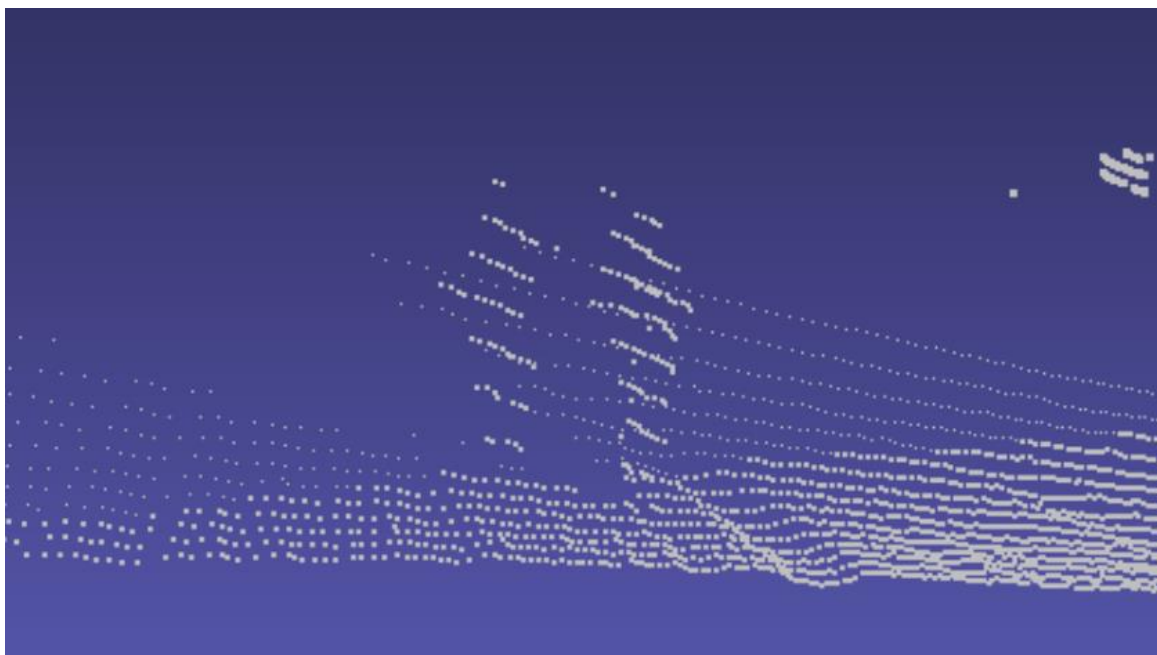


Рисунок 1.2 - Результат ділянки сканування одного оберту 16 лазерних діодів датчика Velodyne VLP-16

FX8 — це 3D лазерний датчик дальності від Nirron Signal, який забезпечує вимірювання відстані до максимального діапазону 15м. Він має горизонтальне поле зору 60° і вертикальне поле зору 50° . Дані надаються з частотою 4Гц з роздільною здатністю 97×61 пікселів. Датчик встановлено на передній частині робота та трохи нахиленою до землі. Його можна використовувати для уникнення перешкод і виявлення рядів рослин під час навігації по полю. Зображення виглядають так, як на рисунках 1.3 та 1.4. Датчик встановлено на передній частині робота та трохи нахилено до землі. Його можна використовувати для уникнення перешкод і виявлення рядів рослин під час навігації по полю.



Рисунок 1.3 - Зображення діапазону, отримане за допомогою лазерного сканера FX8

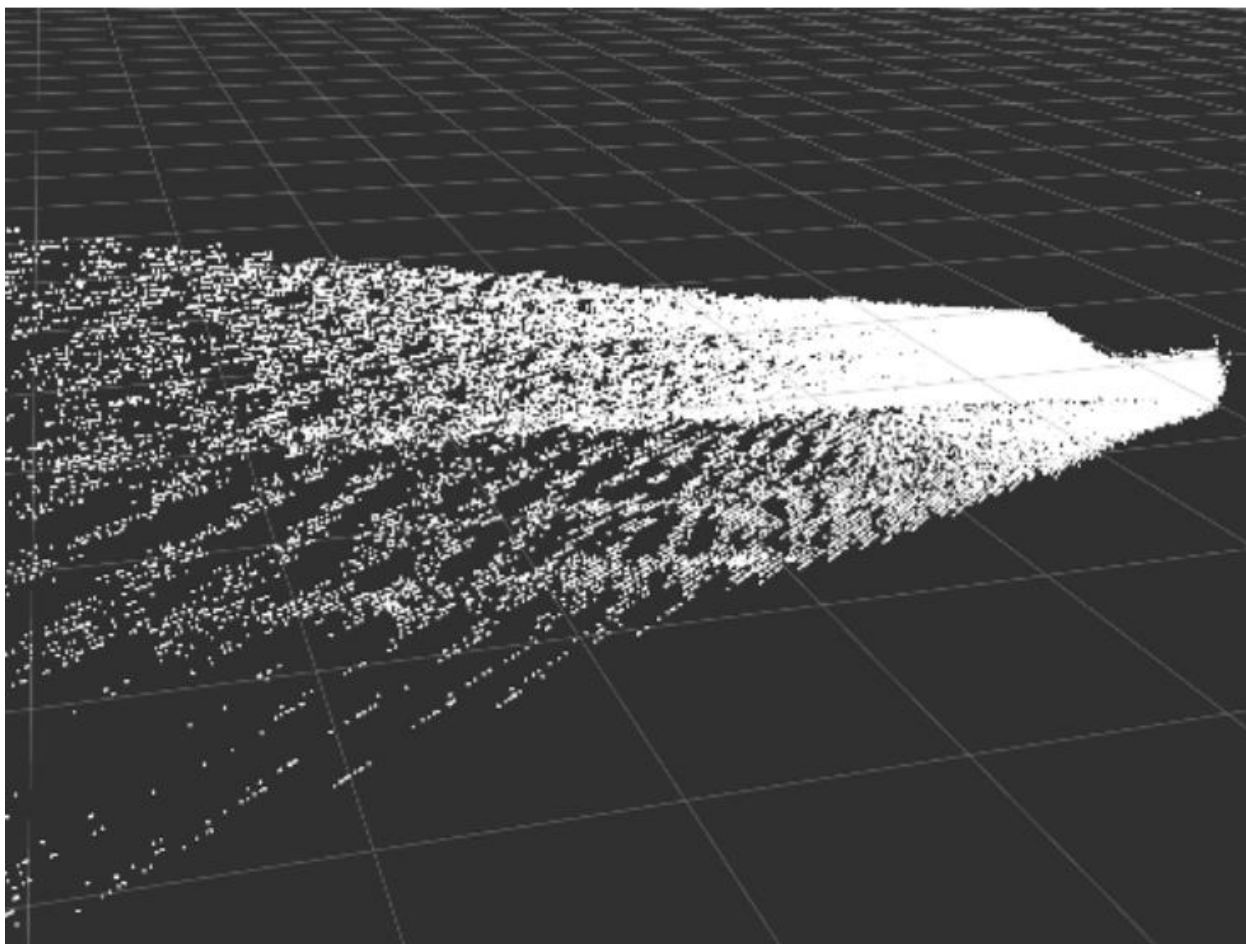


Рисунок 1.4 - Відповідної хмари точок збоку, надані FX8

Щоб відстежувати місцезнаходження робота, розробники використали систему GPS у реальному часі (RTK) від Leica яка забезпечує точні оцінки положення. GPS-приймач RTK відстежує сигнали супутників і додатково отримує спостереження від сусідньої базової станції з відомим розташуванням. За допомогою цієї інформації приймач обчислює поправки стандартного сигналу GPS і покращує оцінку положення з точністю лише до кількох сантиметрів. Запис положення GPS-антени, встановленої на роботі, відносно всесвітньої геодезичної системи 1984 (WGS84) на частоті 10 Гц. На додаток до досить дорогого рішення RTK GPS від Leica, на борту використовувався недорогий GPS-приймач Ublox EVK7-P для відстеження положення робота. Основним принципом визначення положення датчика є точне позиціонування точки. Відстежування позиції за допомогою цього датчика проводилось з частотою 4 Гц відносно WGS84.

Kinect безконтактний ігровий сенсорний контролер камера, спочатку представлений для консолі Xbox 360 та значно пізніше для Xbox One та персональних комп'ютерів, продукт компанії Microsoft. камера надає інформацію про RGB і глибину сцени. Датчик розміщений у передній частині робота, поза кожухом, і нахилили його до землі. Основною причиною розміщення такого була перешкода камері JAI, зокрема в спектрі NIR. У наборі даних випрямлені зображення RGB, NIR і зображення глибини. Оскільки їхні пікселі відповідають один одному, їх можна використовувати для створення тривимірних хмар точок. На малюнках 1.5 та 1.6 показано приклади даних датчика Kinect.

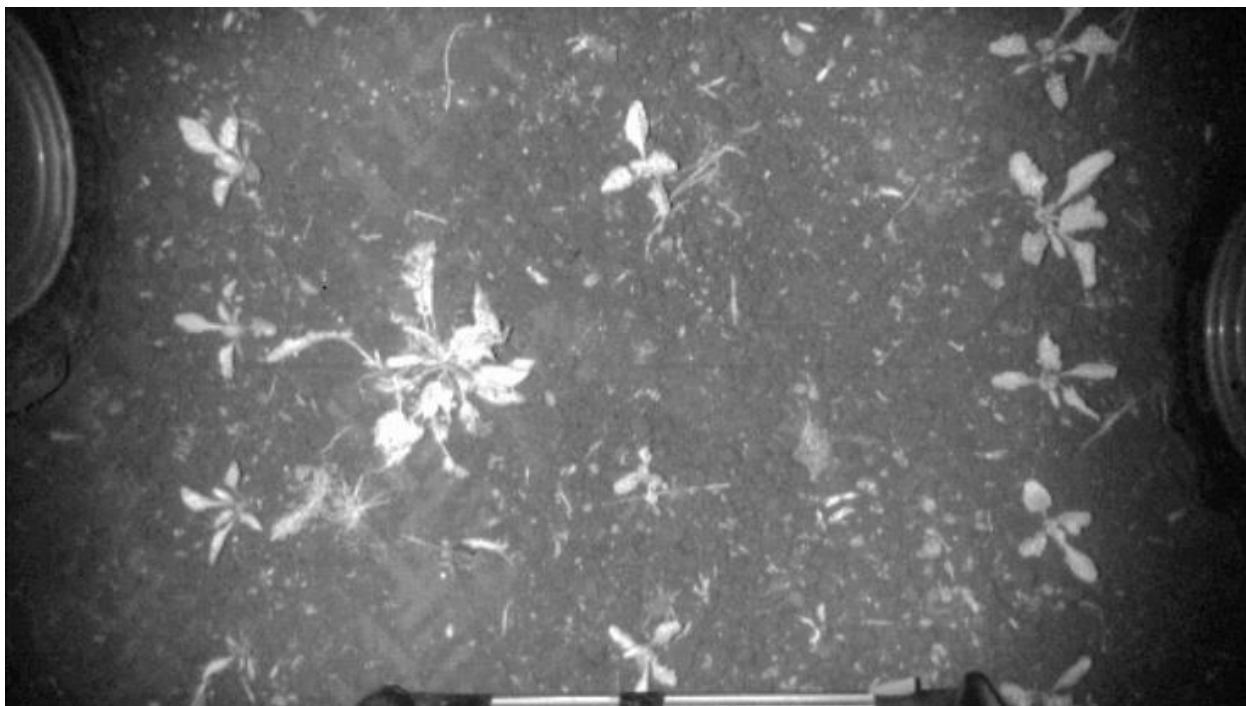


Рисунок 1.5 - ІЧ-зображення камери Kinect

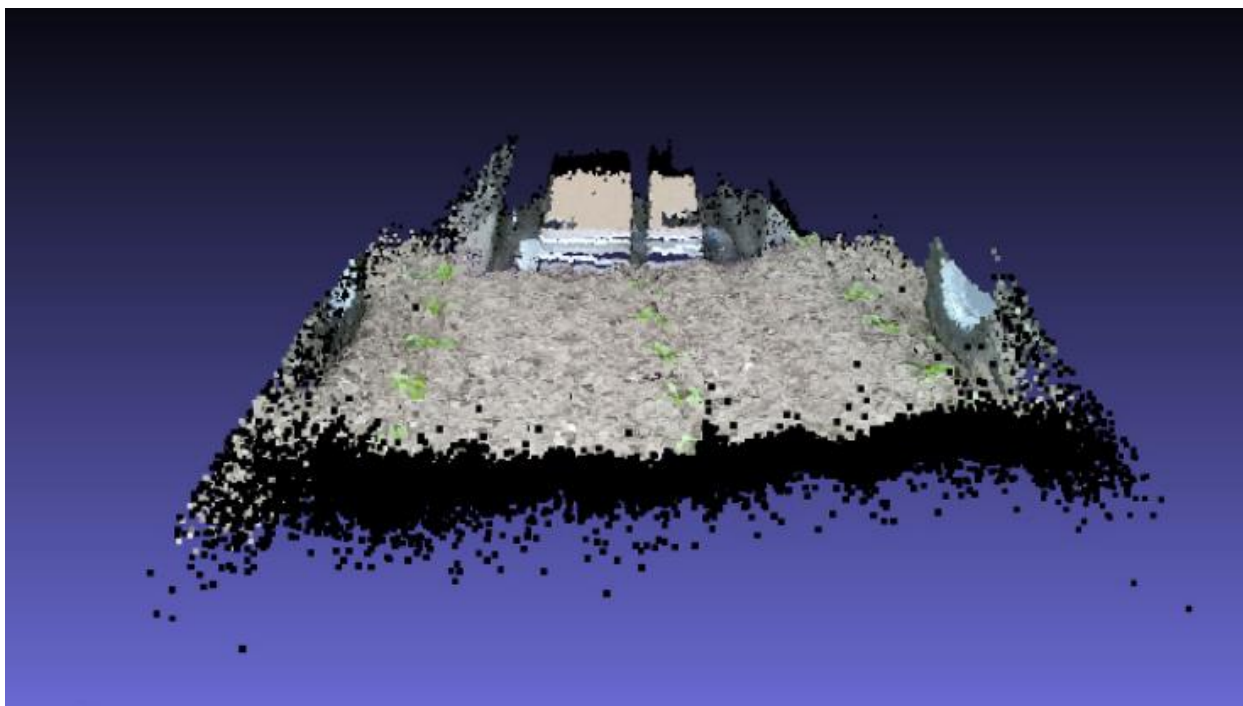


Рисунок 1.6 - Оброблена хмара точок з використанням додаткової інформації про глибину з камери Kinect

За отримання й обробку інформації з приведених вище датчиків відповідає бортовий ПК BoniRob з процесором Intel Core i7 з 6-тю ГБ пам'яті DDR3. Встановлена операційна система Ubuntu 14.04. Окрім Kinect, усі драйвери датчиків запускаються на цій конфігурації, у пристрої використовується додатково, популярне для роботизованих систем, проміжне програмне забезпечення Robot Operating System (ROS). Два сканери Velodyne, камера JAI і сканер FX8 підключені до бортового комп'ютера через концентратор Ethernet. Через вимоги високої пропускної здатності даних, яку вимагає Kinect, розробники під'єднали цей датчик до окремого комп'ютера, який перед записом програмно синхронізували через локальну мережу з головним ПК. Перед експлуатацію система потребує разового налаштування що представляє собою колекцію даних які містять повний набір параметрів калібрування. Це включає в себе внутрішні, тобто специфічні для датчика параметри калібрування для відповідної інтерпретації даних датчика, і набір статичних зовнішніх параметрів калібрування, які кодують позиції датчиків відносно базового зв'язку системи координат робота. Всі внутрішні для

датчика параметри калібрування для відповідності даних камери також враховуючи внутрішні особливості, тому на додаток проходить калібрування параметрів за допомогою бібліотеки калібрування, яка надана розробником, або через OpenCV.

Центром базового зв'язку виступає середина нижньої площини шасі робота, як показано на рисунку 1.7.

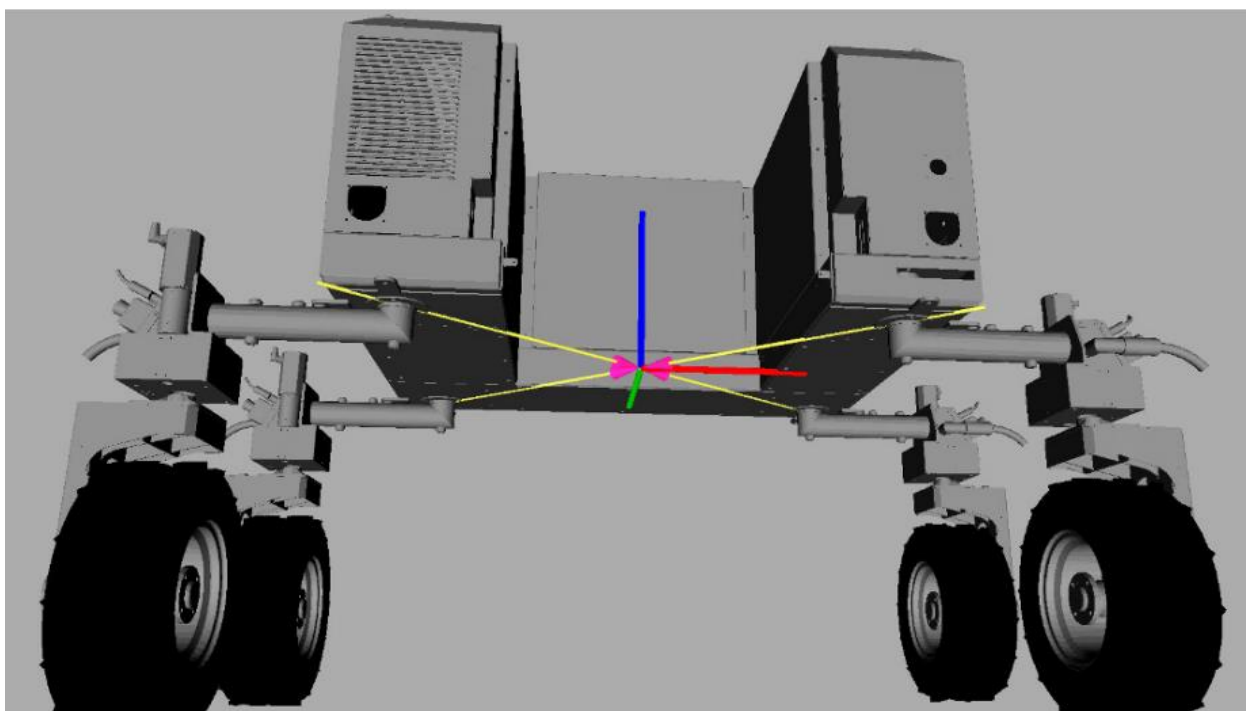


Рисунок 1.7 - Середина відліку координат системи BoniRob

Результатом калібрування на виході отримується файл, який містить інформацію на рисунку 1.8. Вона потрібна процесам при старті системи, щоб виконуючі алгоритми розраховували поставлені задачі, враховуючи габарити платформи й особливості знаходження датчиків. Положення задається у тривимірному представленні x , y та z ; обертання задається кватерніоном q .

Sensor	x [m]	y [m]	z [m]	qx	qy	qz	qw
JAI AD-130GE	0.081	0.138	-0.073	-0.698	0.716	0.008	0.008
Microsoft Kinect v2	0.876	0.040	0.152	0.712	-0.702	0.019	-0.022
Velodyne VLP-16 (front)	0.690	-0.612	0.977	0.168	-0.183	0.968	0.015
Velodyne VLP-16 (rear)	-0.705	0.528	0.972	-0.070	-0.253	-0.638	0.724
Nippon Signal FX8	1.073	-0.086	0.473	0.845	-0.005	0.535	0.010
Leica GPS	-0.861	-0.081	-1.146	—	—	—	—
Ublox GPS	-0.685	-0.053	-0.764	—	—	—	—

Рисунок 1.8 - Координати кожного датчика робота

Щодо підходу обробки й організації даних, то розробники вирішили розділити увесь набір даних на декілька папок, в кожній з яких, записана інформація, поділена по дням збору даних. Усі зображення з камер зберігаються у файлах PNG, стиснутих без втрат. Файли називаються відповідно до наступної угоди:

camera/<sensor>/<type>/<type>_<index> .png

Лазерні дані які реєструються за допомогою двох лазерних сканерів Velodyne (спереду та ззаду) та сканера Nippon Signal FX8. Результируючі хмари точок мають двійковий формат, що містить поля [x, y, z, інтенсивність, кільце]. Перші три поля дають положення виявленої точки в метрах. інтенсивність - це значення в [0, 255]; більш високі значення позначають вищу відбивну здатність. Для Velodyne VLP-16 кожен номер д кільця відповідає певному лазерному діоду. У сумі кожен із 16 лазерних діодів вимірює профіль у певній площині сканування. Бінарні файли записуються за правилом:

laser/<sensor>/scan_<index> .bin

Дані GPS двох пристроїв системи Leica RTK та недорогого Ublox EVK7-PPP зберігаються у вигляді текстових файлів у gps<sensor>gps.txt.

Було проаналізовано та вивчено ще декілька роботизованих систем від різних компаній призначених працювати у полі всі вони мають дуже багато спільного з приводу реалізації навігації та розташування датчиків але відрізняються дизайном та ходовою рисунок 1.9.



Рисунок 1.9 - Приклади реалізації ходової та дизайну польових роботів

Особливу увагу привернув компактний і простий, у технологічному плані, безпілотний наземний польовий дрон компанії XAG – XAG R150 (рисунок 1.10) створений для багатоцільової роботизації сільського господарства. Завдяки можливості заміни різних модулів, не потребуючи додаткового обладнання, виникає можливість змінювати режими роботи.

Робот оснащений повним приводом з двома безколекторними електродвигунами. Передня та задня ходова реалізована на колісних важелях які можна регулювати, змінити висоту шасі, конструкція дозволяє відрегулювати передні колісні важелі окремо або відрегулювавши передні та задні важелі разом. Згідно задумці інженерів зміни, внесені до колісної бази, можуть покращити характеристики платформи в деяких ситуаціях на підйомі.



Рисунок 1.10 – Зовнішній вигляд дрону XAG – XAG R150

Така реалізація ходової дозволяє отримати гарну прохідність та стійкість платформи при великій вазі наприклад у конфігурації поливу. У баку який можна побачити у конфігурації на рисунку 1.11 відсутні так звані хвилерізи, які призначені погасити енергію хвилі я виникає під час руху по нерівному ґрунту. Незважаючи на форму баку, де там хвилі є де потенційно розгулятись, платформа досить стійкою та впевнено рухається по нерівному ґрунту.



Рисунок 1.11 – Дрон XAG R150 у конфігурації поливу

1.3 Постановка завдання

З наведеного вище аналізу розглянутих робототехнічних систем, було виявлено, що у сфері створення мобільних роботизованих систем, компанії-початківці, які часто не мають контрактів із заводами з виготовлення спеціальних високотехнологічних робототехнічних пристроїв, датчиків чи модулів, успішно обмежуються й використовують існуючі загальнодоступні стандартизовані інтерфейси та розроблені до них пристрої. На сьогоднішній день на ринку можна знайти безліч периферійних пристроїв, що дозволяють реалізувати те чи інше завдання, положення, позиціонування, вимірювання, руху для розробки власної роботизованої системи.

За технічним завданням, найбільш підходящим прикладом є автономні польові роботи, аналіз яких показав, що існують деякі готові рішення й підходи. Це надало змогу визначитися з вимогами до розроблювальної системи. Таким чином, на початковому етапі будівництва робота за основу візьмемо спосіб пересування робота та його дизайн, корпус. Необхідно також враховувати, що конструкція шасі роботизованої платформи повинна вміти подолати нерівності ландшафту. Ще необхідно брати до уваги матеріали та існуючі технології для побудови корпусу робота для вирішення конкретного завдання. Наступним кроком - збудуємо інформаційну систему.

2 ПРОЄКТУВАННЯ СИСТЕМИ

У наведеному розділі описується підхід щодо створення структурної системи дрону та його док-станцій. Основне завдання, яке було поставлене - це отримати на виході систему, функціонал якої б забезпечив розроблюваному пристрою автономність його роботи при виконанні поставлених завдань.

2.1 Вимоги до системи

Реалізація власної роботизованої системи потребує проектування та складання апаратної складової системи з доступних на ринку високотехнологічних модулів. Система на виході повинна забезпечити необхідну продуктивність. Електроніка, що буде розміщуватися на конструкції, має забезпечити підтримку виконання програмних технологій штучного інтелекту, алгоритмів комп'ютерного зору та підтримки підключення кількох датчиків.

Датчики надають необхідні дані на основі яких відбувається досягнення виконання поставлених завдань роботи. Таким чином, від апаратної складової потрібно достатня обчислювальна потужність, яка дозволить обробляти на рівні, достатньому для реалізації, вхідні дані з декількох датчиків, а також паралельно виконувати алгоритми, що забезпечують належну поведінку робота (комп'ютерний зір, навчання, реакція, позиціонування). При цьому роботизована платформа повинна розміщувати незалежну енергосистему для вільної подорожі або дослідження вказаної території.

При проектуванні власної системи спочатку було поставлено мету розробити власну загальну структурну схему системи та її алгоритми

функціонування, що стане відправної точкою, яка надалі дозволить структуровано розвивати ідею для досягнення поставлених у роботі завдань.

2.2 Розробка загальної структурної схеми системи

Розроблювана роботизована система поливу на основі дрону складається з двох взаємодіючих по бездротовому зв'язку систем, дрону та док-станції. Загальна структурна схема системи поливу представлена на рисунку 2.1. Дрон є мобільною роботизованою системою, який реалізує більшу частину автономності загальної системи, і бере на себе такі завдання як дослідження виділеної області, подорож до об'єкта поливу, полив і стеження за станом вологості ґрунту. На загальній структурі системи, дрон складається з таких компонентів:

- блоку керування;
- актуаторної системи;
- сенсорної системи;
- автономної системи живлення.

Блок управління дрону призначений для прийому і генерації керуючих імпульсів на актуаторну і сенсорну систему.

Актуаторна система приймає дані з блоку управління і надає можливість дрону проявляти відповідну фізичну активність в навколишнє фізичне середовище.

Сенсорна система забезпечує блок управління дрону даними з навколишнього середовища.

Автономна система живлення забезпечує деякий час вище перераховані блоки необхідним енергоспоживанням.

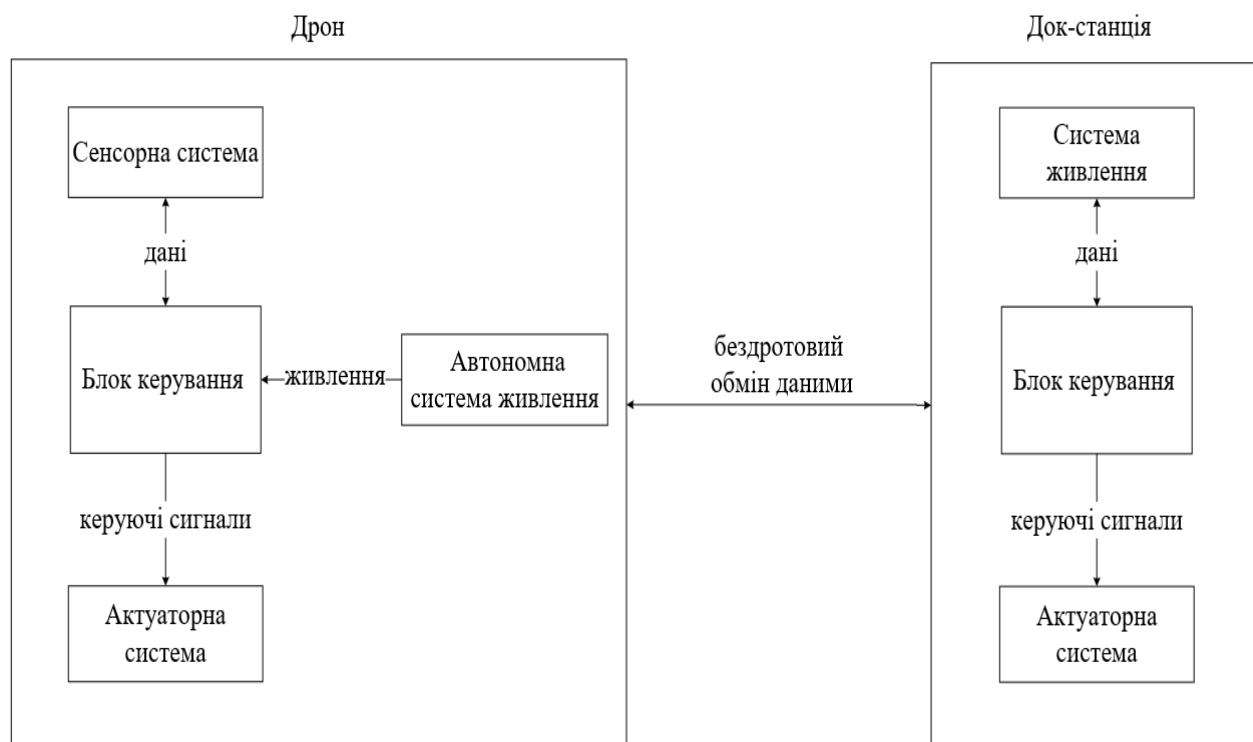


Рисунок 2.1 – Загальна структура роботизованої системи

Док-станція представляє із себе платформу, до якої дрон вимушений подорожувати для того, щоб поповнити запаси ресурсів заряду чи води. Док-станція складеться з наступних блоків:

- блоку керування;
- актуаторної системи;
- системи живлення.

Блок управління приймає дані через канал бездротового обміну даними від дрону й посиляє відповідні сигнали до свого блоку актуаторної системи.

Блок актуаторної системи забезпечує фізичний підвід конструкцій, за допомогою яких здійснюється поповнення ресурсів дрону.

Блок системи живлення надає необхідний рівень напруги для нормальної роботи вузлів док-станції.

2.3 Розробка структурної схеми дрону

Приведена на рисунку 2.2 структурна схема дрону складається з наступних компонентів:

- плати CPU;
- блоку елементу розв'язки;
- двох контролерів;
- двох електродвигунів;
- драйверу крокового двигуна;
- акумулятору;
- DC-DC конвертеру;
- електричної помпи;
- датчику температури й вологості;
- лідару;
- двох гіроскопів;
- GPS модулю;
- відеокамери.

Плата CPU виступає у ролі основного компонента управління у завдання якого входить обробка вхідних сигналів з датчиків й обчислення алгоритмів позиціонування, комунікація, реагування, бачення що реалізують поведінку дрону. Обробивши сигнал або виконавши певний алгоритм плата CPU генерує керуючі сигнали.

Блок елементу розв'язку призначений для повторення й посилення керуючих сигналів, а також забезпечує захист плати CPU від непередбачених стрибків напруги.

Контролери електродвигунів це готові високотехнологічні пристрої які приймають сигнали від блоку розв'язки, призначені забезпечити керування й коректну роботу електродвигунів. Контролер керує лише одним двигуном,

тому для роботизованої системи у якій використовуються два двигуни їх використано також два.

Електродвигуни - це потужні елементи мехатронної системи апаратної складової роботизованої системи, які забезпечують можливість прояву фізичної активності дрону у навколишнє середовище.

Драйвер крокового двигуна є також готовим високотехнологічний пристроєм який приймає сигнали від блоку розв'язки, на основі яких керує подачею струму та напруги до обмоток крокового двигуна.

Кроковий двигун ще один пристрій для досягнення прояву фізичної активності, відмінність лише в тому, що він менш потужний й дозволяє здійснювати дію, з високим ступенем точності.

Акумулятор виступає основним елементом енергосистеми роботи, що здатний забезпечити роботу енерго-компонентів дрону тривалий час незалежно від мережі.

Конвертер DC-DC надає потрібний рівень напруги й струму для коректної роботи плати CPU і елементу розв'язки.

Електрична помпа це елемент, що забезпечує контрольовану подачу води, коли це потрібно.

Датчики: лідар, GPS модуль, температури вологості, відеокамера та гіроскопи, складають сенсорну систему, яка забезпечує роботизовану систему інтерпретованими даними з модулів із навколишнього середовища, які пустують в плату CPU. На основі отриманих даних CPU модуль будує карту оточення, розташування об'єктів та висот, а також розпізнавання об'єктів, і стан навколишнього середовища, тиск, температура, вологість.

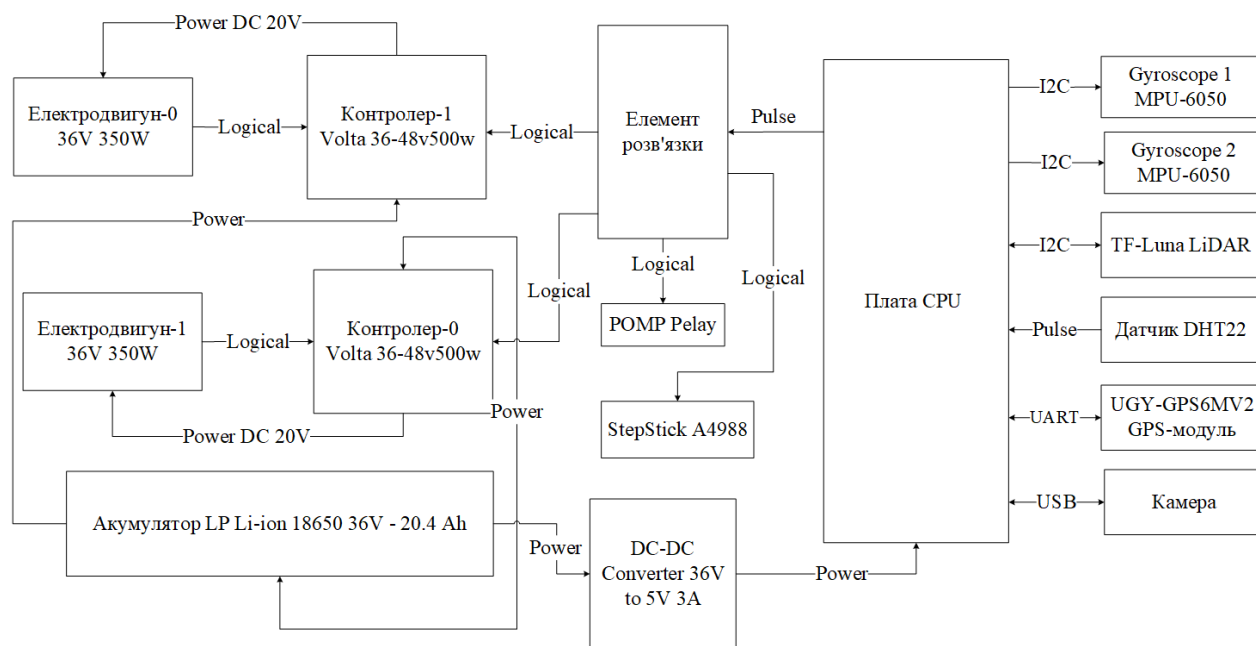


Рисунок 2.2 – Структурна схема дрону

2.4 Розробка структурної схеми док-станції

На рисунку 2.3 наведена структурна схема док-станції, що складається з наступних компонентів:

- плати CPU;
- блоку живлення;
- DC-DC перетворювачу.
- реле помпи;
- драйверу StepStick A4988.

Плата CPU є елементом блоку керування док-станції, яка працює як сервер, що приймає дані від дрону через безпроводний зв'язок, і виконує відповідні дії.

Блок живлення і DC-DC перетворювач реалізують блок системи живлення та надають потрібний рівень постійної напруги. Для роботи помпи

й драйверу, з блоку живлення подається постійна напруга 12V. DC-DC перетворювач знижує напругу до 5V для нормальної роботи блоку плати CPU.

Реле помпи є елементом блоку актуаторної системи й реалізує контрольовану подачу води до дрону.

Драйверу StepStick A4988 є елементом блоку актуаторної керує кроковим двигуном який підводить конструкцію (контакти з фазою) для підзарядки дрону.

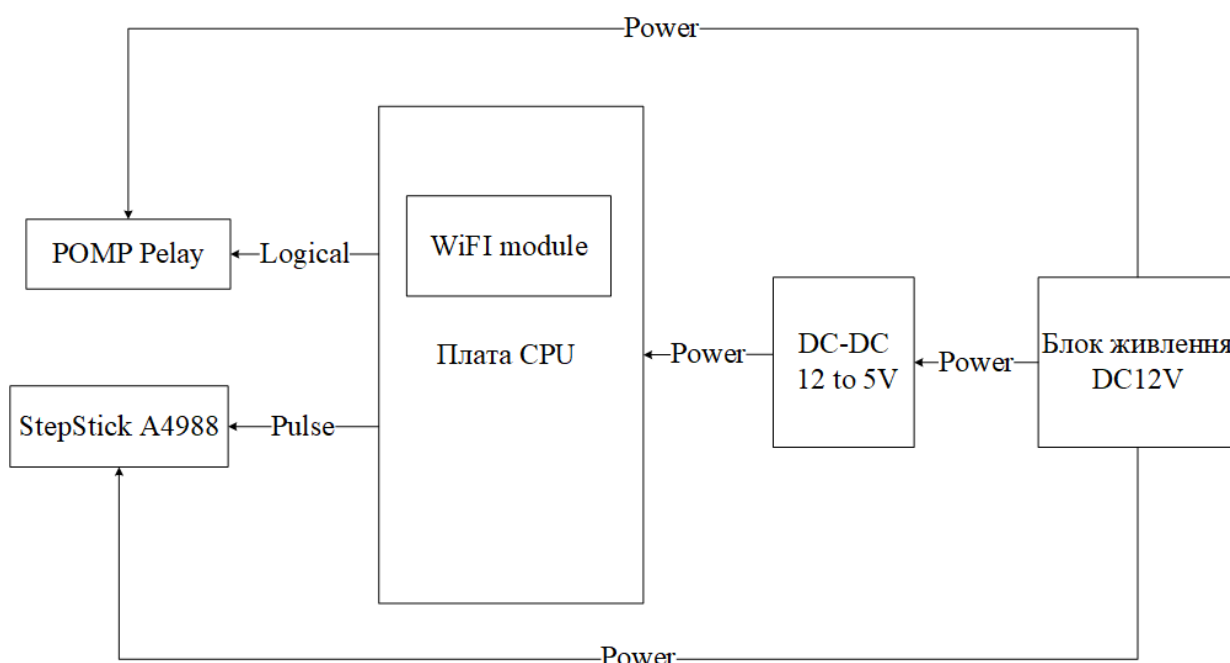


Рисунок 2.3 – Структурна схема док-станції

2.5 Функціонування системи

Взаємодія блоку управління дрону та сенсорної систем здійснюється шляхом безперервного двостороннього обміну даними.

Сенсорна система – це набір датчиків, які приймають дані з навколишнього середовища і передають отриману інформацію в зручному

вигляді для блоку управління. Блок управління приймає та обробляє інформацію з кількох датчиків і робить відповідні висновки. Кожен датчик у блоці сенсорної системи, постачає дані, під які він розроблений. Виходячи з отриманих даних блоку управління дрон отримує можливість орієнтуватися в навколишньому фізичному середовищі, у якому він може себе проявляти, посилаючи керуючі сигнали на блок актуаторної системи.

Блок актуаторної системи являє собою набір силових електромеханічних пристроїв різної потужності, які дозволяють дрону вільно переміщуватися по поверхні ґрунту саду й подорожувати до об'єктів поливу та док-станції. Коли дрон наближається до об'єкта поливу, він звертається до тих даних у блоці сенсорної системи, які надають необхідну інформацію про об'єкт поливу, що стоїть перед ним. Обробивши їх, дрон робить відповідні висновки - виконує полив рослини, подавши управляючий сигнал на актуаторну систему. Коли блок управління дрону прогнозує, що для наступного об'єкта поливу води в резервуарі буде недостатньо або запас блоку автономної системи живлення закінчується, дрон направляється до док-станції для заповнення ресурсів необхідних для поставлених йому завдань.

Коли дрон успішно припаркувався до док-станції, через бездротовий зв'язок він передає сигнал-запит. Приймавши відповідний сигнал від дрону док-станція за допомогою своєї актуаторної системи здійснює підведення механічних пристроїв, які забезпечують відновлення ресурсів дрону води, або ресурсу необхідного для роботи автономної системи живлення дрону. По завершенню процесу док-станція відправляє керуючі сигнали на свою актуаторну систему, яка відводить механічні елементи, що забезпечують заповнення ресурсів дрону. Док-станція надсилає дані дрону про успішне поповнення ресурсів. Дрон прийнявши дані від док-станції продовжує виконувати поставлені йому завдання у блоці управління.

2.5.1 Алгоритм функціонування системи

Загальне функціонування системи можна описати двома взаємодіючими між собою алгоритмами функціонування дрону та док-станції. Алгоритмом загального функціонування дрону наочно представлений блок-схемою на рисунку 2.4. З початку дрон намагається виявити док-станцію, що стоїть поруч, при успішному обміні даними він переходить до наступного етапу - перевірки роботи сенсорної та актуаторної системи, у випадку невдачі на цьому етапі система дрону записує звіт про помилку, потім він завершує свою роботу. При успішній перевірці всіх систем, алгоритм роботи системи дрона переходить до нескінченного циклу. У циклі в першу чергу обробляються дані, що надходять з сенсорної системи, на основі отриманих даних процес алгоритму поведінки, залежно від необхідності видає керуючий сигнал на актуаторну систему. Алгоритм поведінки дрона є складним багато-потоким процесом, в завдання якого входить структурувати інформацію, що надходить з сенсорної системи та обробляти її для подачі в зручному вигляді на виконання алгоритмів нечіткої логіки й різних методів прийняття рішень, які дозволяють системі належним чином інтерпретувати інформацію і реагувати – постилати сигнали на актуаторну систему.

Алгоритм загального функціонування док-станції наочно наведено на рис. 2.5. З моменту початку роботи док-станція чекає на прийом даних. Коли док-станція отримує дані, виконання алгоритму переходить до нескінченного циклу, в якому відпрацьовується процес поведінки. Він оброблює прийняті дані з дрону й подає відповідні керуючі сигнали на свою актуаторну систему. При відсутності керуючого сигналу з блоку процесу алгоритму поведінки відбувається вихід з нескінченного циклу. Вихід з циклу супроводжується відправкою даних дрону про успішне завершення процесу обслуговування та переходить у стан прослуховування даних.

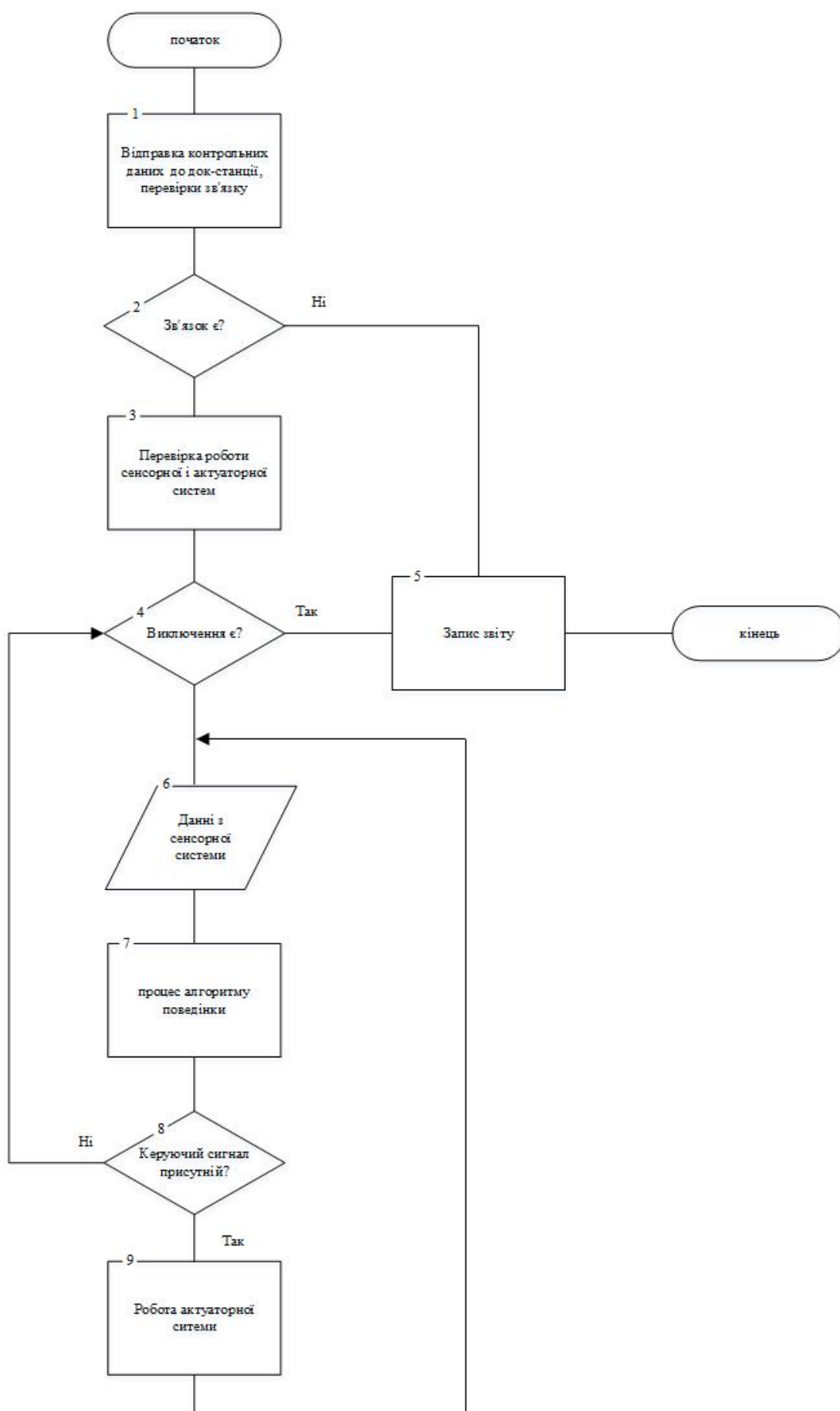


Рисунок 2.4 – Узагальнений алгоритм функціонування системи дрону

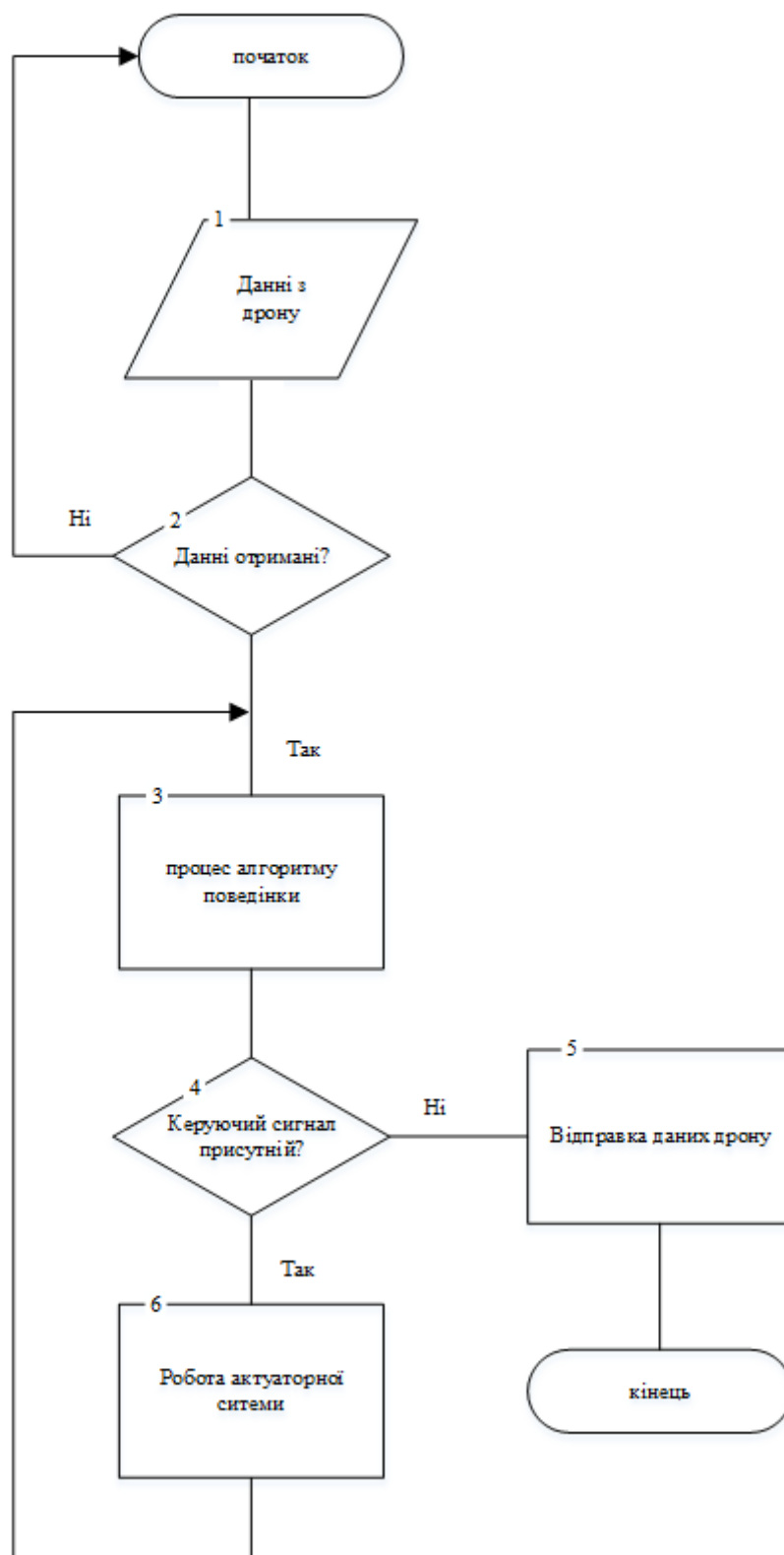


Рисунок 2.5 – Загальний алгоритм функціонування системи док-станції

3 АПАРАТНЕ ЗАБЕЗПЕЧЕННЯ СИСТЕМИ

У наведеному розділі описується підхід щодо процесу розробки побудови електричної схеми, яка включає себе обчислювальну, сенсорну та актуаторну системи мобільної роботизованої платформи. Наведено висновок, на основі яких переваг, був обраний той чи інший модуль в електричній схемі пристрою. У процесі розробки так само до уваги приймалися і розглядалися включення деяких готових рішень та ідей досліджених пристроїв у першому розділі, для прискорення процесу розробки. Основне завдання розробки – отримати на виході систему здатну приймати паралельно сигнали з кількох датчиків та обчислювати алгоритми прийняття рішень, за допомогою позиціонування та комп'ютерного зору.

3.1 Вибір стеку технологій

Резюмуючи проведений аналіз у першій частині основних порівняно стійких напрямків розвитку у сфері створення мобільних роботизованих систем, можна зробити висновок, що у починаючих компаніях, і не тільки, розробники дотримуються використання модульного принципу побудови систем керування, які сумісні між собою за рахунок уніфікованих інтерфейсів інформаційного обміну на основі шин, а також використання поширеного спеціалізованого додаткового програмного забезпечення роботизованих систем (ROS, або його альтернатив) для популярних операційних систем. Тому апаратне ядро системи керування повинно забезпечити підтримку популярних операційних систем, під яку можна встановити або скомпілювати потрібний програмний продукт, для роботи компонентів власного

програмного забезпечення для прийому та обробки інформації з датчиків через популярні шини передачі даних.

3.2 Блок керування дрону

Для блоку управління дрону розглянемо два пристрої: материнська плата MiniITX з процесором на x86 архітектурі з інтегрованою графікою та популярний одноплатний комп'ютер Raspberry PI.

Вибір материнської плати MiniITX з процесором й інтегрованою графікою Intel Core i5 забезпечив би роботизовану систему високою обчислювальною потужністю. Розглянута плата ELSA H510 Mini ITX Motherboard LGA 1200, без процесору (рис. 3.1), має два SODIM слоти під DDR4 й також є два M2 слоти. Один розрахований на підключення твердотілого SSD накопичувача, другий розрахований під модуль WiFi.



Рисунок 3.1 – Плата ELSA H510 Mini ITX Motherboard LGA 1200

Плата має виводи живлення 19-12V DC та стандартний набір інтерфейсів введення виводу, таких як Ethernet, USB, COM. Для інтеграції цієї плати в апаратну систему управління робота постає проблема з нестачі таких популярних інтерфейсів, як SPI, TWI або просто можливість подати логічний сигнал, плата формату Mini ITX, які задовольняють потреби, не мають LTP порт.

Аналіз ринку показав, що необхідні модулі та датчики, які популярні серед розробників зазвичай постачаються з інтерфейсами, SPI, TWI, UART або своєю логікою(протоколом). Були виявлені також пристрої розраховані на роботу з шиною USB, але документація знаходилася непросто, адже пристрої мали свій власний драйвер, реалізований у режимі device. Це сіяло сумніви, що при придбанні пристрою і підключення його до системи, модуль або датчик, що працює в режимі device не вимагатиме установки драйверів в систему, виникають проблеми доступу у своїй програмі до даних пристрою, маючи сумнівну документацію й жодного прикладу роботи.

Одним з простих виходів в даній ситуації - це докупити модулі USB to SPI або TWI, UART, а модулі, які керуються логічними сигналами керувати через такі готові рішення, як плат Arduino на базі мікроконтролерів AVR або плат з ARM мікроконтролерами.

Розглянута система материнської плати плюс процесор розглядався як основний обчислювальний модуль роботизованої системи з причини планування використання такого модуля, як Kinect, розглянутий в розділі 1.2, у своєму проекті.

Камера Kinect популярний пристрій серед розробників. Є багато прикладів використання його в проектах з відкритим кодом у використанні з бібліотекою OpenCV. Оскільки пристрій надає дані про глибину і колір він вимагає чималу обчислювальну потужність.

При виборі однопалатного комп'ютера Raspberry PI, як центральний обчислювальний пристрій, витрати часу на розробку ПЗ значно зменшуються, на відміну від використання плати ELSA H510 Mini ITX. Комп'ютер Raspberry

PI 4 B представляє із себе плату, на якій розпаяні центральний і графічний процесор, пам'ять й 40 контактів GPIO для підключення цифрових датчиків, модулів розширення та іншої периферії. Комп'ютер Raspberry PI дуже популярний пристрій, до якого існує безліч прикладів використання різної периферії, а також великий обсяг документації та літератури, що в значній мірі дозволяє прискорити процес розробки. З цих причин на роль апаратного ядра було обрано мікрокомп'ютер Raspberry PI 4B, вигляд якого можна побачити на рисунку 3.2.



Рисунок 3.2 – Мікрокомп'ютер Raspberry Pi 4 B

3.3 Сенсорна система

Для реалізації блоку сенсорної системи дрону, яка була описана у підрозділі 2.3, були обрані такі готові апаратні пристрої, які спроможні наділити роботизовану систему дрону можливістю взаємодії з навколишнім середовищем. Сенсорна система складеться заступних пристроїв:

- двох гіроскопів MPU-6050;
- GPS модулю GY GPS6MV2;

- датчику вологості та температури DHT22;
- лідару TF-Luna;
- камери Trust HD 1080P.

3.3.1 Пристрій відстеження руху

Сенсорний пристрій MPU6050 є готовим 6-осьовим пристроєм відстеження руху. Він поєднує в собі 3-осьовий гіроскоп, 3-осьовий акселерометр та цифровий процесор руху з англійської Digital Processor Motion (DPM) в невеликому корпусі (рис. 3.3).

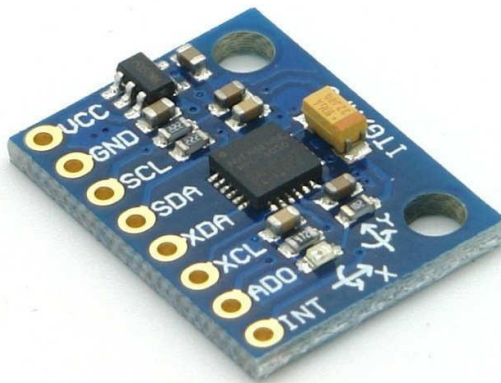
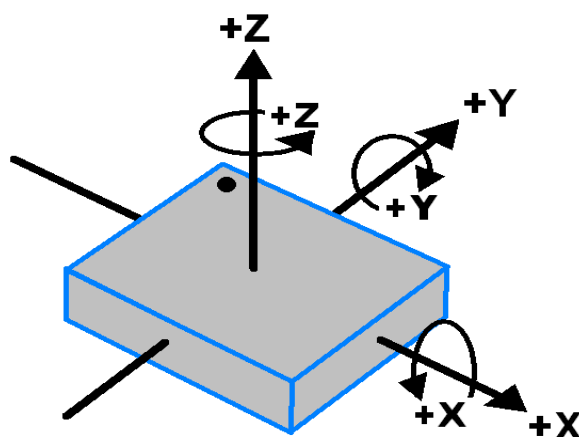


Рисунок 3.3 – Пристрій MPU6050

Трьох осевий гіроскоп виконаний за технологією мікроелектромеханічної системи (MEMS). Він спроможний визначати швидкості обертання по осях X, Y, Z, як показано на рисунку 3.4.



MPU-6050
Orientation & Polarity of Rotation

Рисунок 3.4 – Три осі обертання гіроскопу

Для детектування вісєвого обертання модуль оснащений трьох вісєвим акселерометром з мікроелектромеханічною технологією (MEM). Він використовується для визначення кута нахилу або нахилу по осях X, Y та Z (рис. 3.5). Прискорення вздовж осей відхиляє рухому масу, яка являє собою зсув пластини, що рухається, яка розбалансує диференціальний конденсатор. Це призводить до вихідного сигналу датчика. Вихідна амплітуда пропорційна прискоренню.

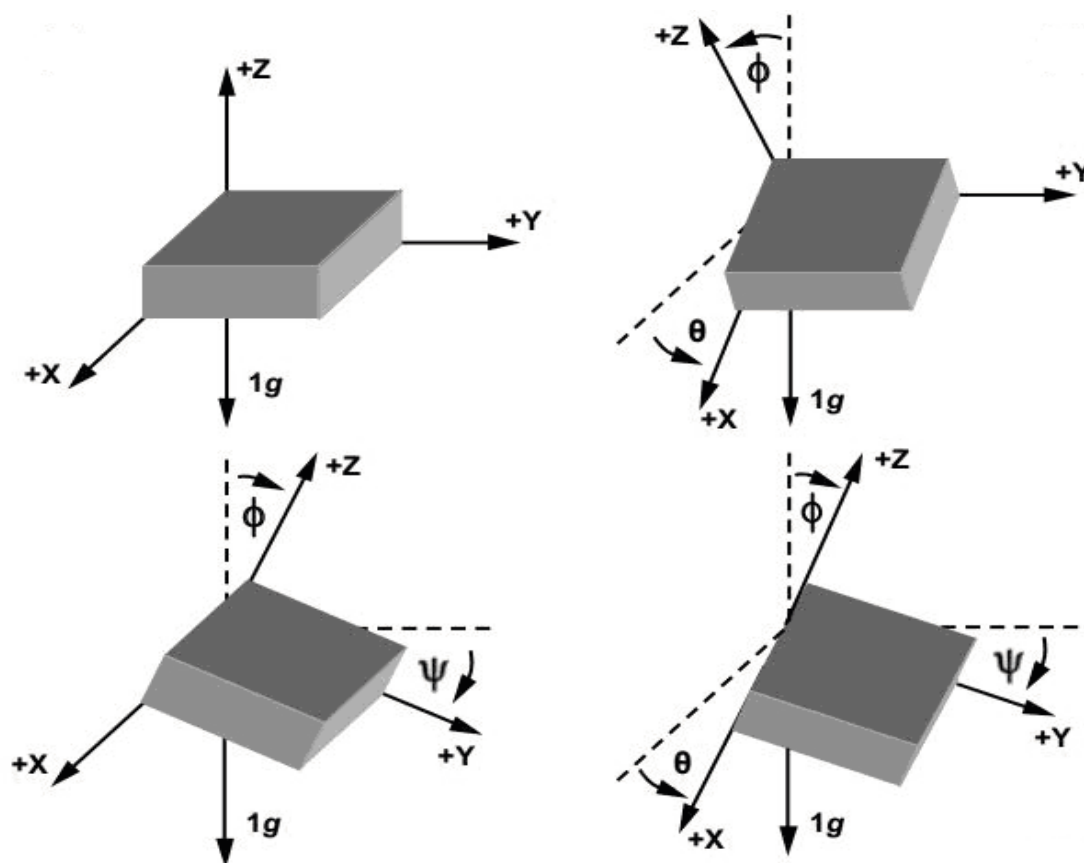


Рисунок 3.5 – Приклад випадків прискорення вздовж осей

Крім того, пристрій має додаткову функцію вбудованого датчика температури. Пристрій використовує інтерфейс шини I2C для зв'язку з іншими пристроями, мікроконтролерами й ПЛІС або комп'ютерами на кшталт таких як LibreComputer Board AML-S905X-CC (Le Potato), Asus Tinker Board S R2.0, Raspberry Pi 4B. Датчик має ще допоміжну шину I2C для зв'язку з іншими сенсорними пристроями, такими як датчик тиску та 3-осьовий магнітометр. Вбудований цифровий процесор руху (DMP) використовується для обчислення алгоритмів обробки руху. Він бере дані з гіроскопа, акселерометра та додаткового стороннього датчика, такого як наприклад магнітометр, та обробляє дані. Він надає дані про рух, такі як крен, тангаж, кути нишпорення, відчуття ландшафту. Він мінімізує процеси хоста при обчисленні даних про рух. Результати можуть бути зчитані з регістрів DMP.

GPS-приймач GY-GPS6MV2 модуль є оновленою версією GY-GPS6MV1. Приймач GY-GPS6MV2 побудований на базі GPS-чіпа NEO-6M,

вважається точним та надійним. Плата має вбудовану пам'ять та резервну батарею для зберігання налаштувань. Передача даних здійснюється за допомогою інтерфейсу UART тому може бути використаний в пристроях Arduino, AVR, PIC, ARM, FPGA і комп'ютерів, які аналогічні Raspberri PI. Пристрій поєднує в собі плату приймач та керамічну антену (рис. 3.6).



Рисунок 3.6 – Зовнішній вигляд GPS приймача та антени

GPS-приймач GY-GPS6MV2 модуль є оновленою версією GY-GPS6MV1. Приймач GY-GPS6MV2 побудований на базі GPS-чіпа NEO-6M, вважається точним та надійним. Плата має вбудовану пам'ять та резервну батарею для зберігання налаштувань.

3.3.2 Датчик температури та вологості

DHT22 – це простий недорогий цифровий датчик температури та вологості. Він використовує ємнісний датчик вологості та термістор для вимірювання стану навколишнього середовища та видає цифровий сигнал на контакт даних. Він є досить простим у використанні, після запиту приблизно через 2 секунди надає вимірювальні данні. Надійні пристрій для базової реєстрації даних.

3.3.3 Модуль виявлення та визначення дальності світла

TF-Luna (рис. 3.7) модуль виявлення та визначення дальності світла (LiDAR), розроблений компанією Benewake. Лідар використовує принцип часу прольоту (ToF) для виявлення об'єктів у поле зору датчика. Пристрій може вимірювати об'єкти на відстані від 20 см до 8 м, залежно від умов навколишнього освітлення та відбивної здатності поверхні об'єктів, на які спрямований промінь. Промінь це результат роботи лазера із поверхневим випромінюванням з вертикальним резонатором (VCSEL) знаходиться в центрі TF-Luna, який класифікується як лазер класу 1, що робить його дуже безпечним практично для всіх застосувань.



Рисунок 3.7 – Зовнішній вигляд TF-Luna

TF-Luna має частоту дискретизації, що вибирається, від 1 Гц до 250 Гц, це робить його задовільним для більш швидких сценаріїв виявлення на відстані. Датчик підключається до комп'ютера Raspberry Pi або його аналогів, можна і до мікроконтролеру, через послідовний порт mini UART та живиться від контакту 5 В. Приведений модуль LiDAR є популярним й широко використовуються в багатьох проектах системах навігації для дронів та

квадрокоптерів у 2D-3D-скануванні об'єктів та інших додатках для виявлення відстані на великій швидкості.

3.3.4 Камера

Камера Trust GXT 1160 Vero Streaming (TR22397) веб-камера Full HD 1080p для створення плавних відео трансляцій та запису стереозвуку. Веб-камера роздільної здатності до 8 мегапікселів (3840 x 2160). Вміє в автоматичний баланс білого на виході чітке зображення за будь-якого освітлення за частотою кадрів 30 за секунду.

3.4 Актуаторна система

Для того, щоб роботизована система була спроможна відповідати на подразники та мати змогу пересуватись, було обрано такі пристрої, які реалізують актуаторну апаратну систему дрону:

- два електродвигуна 36v 350w BLDC;
- двигун NEMA17;
- два контролера Volta 36-48v500w;
- драйвер StepStick A4988;
- електромагнітний клапан 2w-250-25DC12V.

3.4.1 Двигуни системи

Електродвигун BLDC від компанії "Volta bikes" представляє із себе 3-х фазний без щітковий електродвигун (рис. 3.8) з постійним неодимовими магнітами з датчиками Холла. Електродвигун при придбанні в своїй конструкції йде з встановленим планетарним редуктором який забезпечує високий момент обертоту простоту старту. Редуктор немає муфти вільного ходу. Це дозволяє змінювати напрямок обертання валу електродвигуна (рух вперед – назад) та забезпечувати ефективне гальмування двигуном.



Рисунок 3.8 – Електродвигун BLDC 36v 350w з редуктором

NEMA 17 є гібридним кроковим двигуном (рис. 3.9) з кутом кроку $1,8^\circ$ (200 кроків/оборот). Кожна фаза споживає 1,2 А при 4, що забезпечує утримуючий момент 3,2 кг-см. Кроковий двигун NEMA 17 користуються популярністю й зазвичай його можна побачити як він зазвичай використовується в проектах принтерів, верстатах з ЧПУ та лазерних різаків.

3.4.2 Контролери

Контролер Volta 36-48v500w перетворює елементом розв'язки між керуючими сигналами процесорної та силової (рис. 3.9). Основна функція це регулювання швидкості обертання валу колекторних або безколекторних електродвигунів постійного струму, що мають номінальну напругу 48 вольт й в короткочасних режимах реалізує потужність електродвигуна до 1200 ват. Рекомендується для використання з електродвигунами постійного струму як з редуктором, так і без нього, з номінальною напругою 48v та номінальною потужністю 350w – 500w.



Рисунок 3.9 – Вигляд універсального контролеру Volta 36-48v500w

3.4.3 Комунікаційні пристрої

StepStick A4988 продукт є перехідною платою для мікро крокового драйвера Allegro A4988 DMOS з транслятором і захистом від перевантаження по струму. Надає змогу керувати біполярними кроковими двигунами в режимах повної, половини, четверті, восьми та шістнадцяти кроків з вихідною потужністю приводу до 35 В та 2 А. Просте введення одного імпульсу на вхід STEP переводить двигун на один мікрокрок. Пристрій немає таблиць чергування фаз, високочастотних ліній управління чи складних інтерфейсів програмування. Інтерфейс A4988 ідеально підходить для програм, в яких складний мікропроцесор недоступний або перевантажений.

Мембранний електромагнітний клапан 2w-250-25DC12V, який працює від постійного струму, на виході має два проводи чорний та червоний, при підключенні 12V живлення, переходить зі стану закритого на відкритий. Зовнішній вигляд пристрою можна побачити на рисунку 3.10.



Рисунок 3.10 – Зовнішній вигляд мембранного електромагнітного клапану

3.5 Системи живлення

Система енергозабезпечення дрону складається з таких компонентів:

- акумулятор LP Li-ion 18650 36V - 20.4 Ah (BMS 30A/20A);
- зарядний пристрій 36В 2А для літій-іонних акумуляторів;
- DC 5V Converter 36V To 5V 3A;

Літієва акумуляторна батарея тип батареї, що перезаряджається, заснованої на літій-іонній технології, в якій використовується полімерний електроліт. Ці батареї набули величезної популярності, особливо в індустрії радіоуправління. В даний час вони є найпопулярнішим варіантом для тих, хто шукає високу потужність та тривалий час роботи. Зазвичай для того щоб отримати бажану потужність на виході, акумулятор складають з банок 18650 (рис. 3.11).

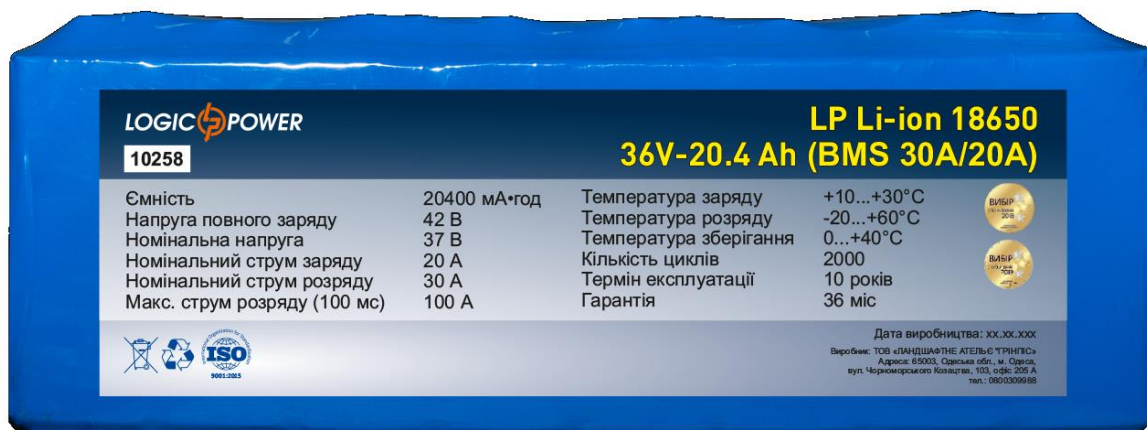


Рисунок 3.11 – Зовнішній вигляд зібраної батареї з банок 18650

Автоматичний зарядний пристрій 36В 2А, для літій-іонних акумуляторних батарей електровелосипеда ємністю 10-14Ач. Зовнішній вигляд на рисунку 3.12. Зарядний пристрій забезпечений кількома ступенями захисту, такими як:

- захист від короткого замикання;
- захист від перевантаження струмом;
- захист від перенапруги;
- захист від помилок полярності батареї.

Захист від перезарядження. Коли процес заряду йде процес заряду батареї горить - червоний світлодіод, коли батарея повністю заряджена, процес зарядки автоматично зупиняється і спалахує - зелений світлодіод. Це можна використати для інформування системи керування робота, розібравши корпус пристрою й брати з світлодіодів напругу.



Рисунок 3.12 – Зовнішній вигляд корпусу зарядного пристрою

DC 5V Converter 36V To 5V 3A - це перетворювач постійного струму на постійний являє собою електронну схему яка перетворює джерело постійного струму (DC) з одного рівня напруги на інший. У проекті модуль забезпечує живлення плати Raspberry Pi 4 В.

3.6 Допоміжні пристрої апаратної складової дрону

Додаткові компоненти\модулі, які реалізують системи розв'язки розроблюваної апаратної системи дрону, націлені забезпечити максимальну ізоляцію ланцюгів змінного та постійного струму, забезпечуючи при цьому сумісність із ТТЛ. Додаткових компонентів два це мікросхема HCPL-0600 і мікроконтролер AVR-ATtiny24.

Мікросхема HCPL-0600 – це оптопара, що є логічним елементом з оптичним зв'язком, який поєднує в собі GaAsP-світлодіод і вбудований

фотодетектор з високим коефіцієнтом посилення. HCPL-0600 підходить для високошвидкісного логічного інтерфейсу, буферизації вводу/виводу, стробування.

Мікроконтролер AVR-ATtiny24. Малопотужний восьми-розрядний мікроконтролер від компанії Microchip на основі AVR® RISC. На кристалі 2 КБ флеш-пам'яті ISP, 128 байт EEPROM, 128 байт SRAM, має 12 ліній введення/виводу загального призначення, 32 робочих регістра загального призначення та два таймери/рахунок. (8-біт/16-біт) з двома каналами ШІМ у кожному, внутрішніми та зовнішніми перериваннями, 8-канальний 10-бітний аналого-цифровий перетворювач, програмований каскад посилення (1x, 20x) для 12 диференціальних пар каналів АЦП, програмований сторожовий таймер з внутрішнім генератором, внутрішнім каліброваним генератором і чотири програмно вибираючи режими енергозбереження.

Виконуючи потужні інструкції за один такт, пристрій досягає пропускної здатності, що наближається до одного MIPS на МГц.

Розглянуті пристрої у даному розділі представляють із себе основні блоки конструкції з яких спроектована апаратної система яка включає в себе керуючу і актуаторну системи.

3.7 Блок керування док-станції

Апаратна складова док-станції роботизованої системи складається з наступних компонентів:

- двигун NEMA17;
- драйвер StepStick A4988;
- електромагнітний клапан 2w-250-25DC12V;
- мікроконтролер Raspberry Pi Pico W;

Перші три пристрої були розглянуті у розділі 3.4.1. Пристрій Raspberry Pi Pico W - це 32-біта високопродуктивна плата мікроконтролера від Raspberry Pi на основі чіпа RP2040. Мікроконтроллер має 2 МБ вбудованої флеш (QSPI) пам'яті, симетричний дводенний процесор Arm Cortex-M0+ 133 МГц й детерміновану структуру шини зі стандартний набором периферійних пристроїв, доповнених унікальною підсистемою програмованого вводу-виводу (PIO). Raspberry Pi Pico W має на платі модуль Wi-Fi, що забезпечує підтримку бездротової локальної мережі 2,4 ГГц 802.11 b/g/n із вбудованою антеною та модульною сертифікацією відповідності. Пристрій може працювати як у станції режимі, так і точкою доступу. Повний доступ до мережних функцій доступний у офіційному пакеті SDK.

3.8 Розробка електричної схеми компонентів системи

3.8.1 Розробка електричної схеми дрону

На рисунку 3.12 переставлена схема електрична принципова дрону. Блоком керування виступає мікрокомп'ютер Raspberry Pi 4B, який безпосередньо пов'язаний I2C шиною з трьома пристроями, що складають частину сенсорної системи, лідаром TF-luna і двома гіроскопами MPU-6050. Стандарт передачі даних I2C підтримує конфігурацію I2C Slave, Master і Multi-Master і може бути легко сумісний з іншими відомими та провідними пристроями сторонніх виробників. Шина I2C – це дводотовий апаратний інтерфейс промислового стандарту, що підтримує стандартні тактові частоти до 1000 Мбіт/с був розроблений компанією Philips. Master ініціює весь зв'язок по шині I2C і встановлює час для всіх введених пристроїв, на що і було розраховано при побудові електричної схеми, мікрокомп'ютер raspberry Pi 4B виступає як Master і в залежності від ситуації, в будь-якій момент, може

переініціалізувати шину тобто час опитування датчиків, для отримання якнайбільше потрібних даних з лідару або гіроскопів.

Приймач GPS, є частиною сенсорної системи дрону, на схемі він підключений до Raspberry Pi через шину UART, апаратний протокол зв'язку, що використовує асинхронний послідовний зв'язок з швидкістю, яка налаштовується. Асинхронність означає відсутність тактового сигналу для синхронізації вихідних бітів від передавального пристрою GY-GPS6MV2 до сторони, що приймає, Raspberry Pi 4B.

Наступним елементом сенсорної системи виступає датчик вологості і температури DHT22 він використовує один дріт/шину для зв'язку на схемі підключений дріт DATA, що використовується для зв'язку між мікрокомп'ютером і пристроєм DHT22, підключається до високого рівня за допомогою резистора на 10 кОм, для реалізації так званої підтяжки. Це робиться для того, щоб перевести шину в стан IDLE за відсутності обміну даними.

Останнім розглянутим елементом сенсорної системи виступає HD камера від компанії Trust, яка підключена до мікрокомп'ютера через USB шину, яка забезпечує пристрій живленням та високошвидкісною передачею даних.

Блок розв'язки реалізований за допомогою двох інтегральних схем мікросхеми HCPL-0600 та мікроконтролера ATtiny24. Мікросхема HCPL-0600 є логічним елементом з оптичною зв'язкою був використаний для реалізації власного швидкісного логічного інтерфейсу, при цьому надаючи повну ізоляцію мікрокомп'ютеру Raspberry Pi 4B. Комунікаційний процес відбувається за допомогою шини яка складеться з одного дроту між мікроконтролером ATtiny24 та Raspberry Pi через розв'язувальний елемент HCPL-060. Розраховуються що мікроконтролер буде запрограмований працювати тільки у режимі Slave. Для швидкого реагування дріт шини підключений к ніжці PA0 порту A, на якій реалізована апаратна можливість налаштувати її на роботу найвищого рівня зовнішнього переривання.

Мікроконтролер ATtiny24 обробляючи пакет прийнятих даних приймає рішення продовжувати подавати або подавати чи ні відповідні керуючі сигнали 5V на модулі що керують силовою частиною електричного кола актуаторної системи.

Керування двома безколекторними двигунами, які реалізують частину актуаторної системи, відбуваються завдяки контролерам Volta 36-48v500w, один пристрій розрахований керувати лише одним двигуном, управління здійснюється через спеціально відведений інтерфейс що керується рівнями напруги від ATtiny24.

Керуючий сигнал PA3 переключає стан транзисторного ключа TIP120, для включення, елемента актуаторної системи, соленоїда помпи у потрібний момент.

Завдяки реле SSR-25 DA мікроконтролер керує включенням зарядного модулю CM на схемі. Прилеглі контакти між мікроконтролером і зарядним модулем PA4, PA5 дають можливість зрозуміти чи відбуватися заряд в даний момент або про завершення заряду.

3.8.2 Розробка електричної схеми док-станції

Електрична схема принципова док-станції приведена на рисунку 3.13. Основним компонентом блоку апаратної системи док-станції є мікроконтролер raspberry pi pico w на платі з модулем Wi-Fi. Вся система запитується від блоку живлення, що складається з понижуючого трансформатора з випрямлячем, що дає на виході 12V постійної напруги, яка служить джерелом живлення для соленоїду помпи. Модуль перетворювача L7805cv знижує напругу до 5V для забезпечення нормальної роботи драйвера StepStick A4988 та модулю живлення на платі Raspberry PI Pico W. Інтегральна мікросхема HCPL-0600 забезпечу оптичну розв'язку й підсилення керуючого сигналу.

4 НАЛАШТУВАННЯ РОБОТИ СИТЕМИ

4.1 Вибір стеку технологій

Пристрій, який буде відігравати роль центрального обчислювального ядра розроблюваної роботизованої системи буде виступати одноплатний комп'ютер Raspberry Pi Model B. Комп'ютер підтримує потрібні технології передачі даних і задовольняє потреби чисельної потужності й потужності енергопотреби, має невеликі розміри, і є простим у налаштування та конфігуруванні й встановлення програмних компонентів. На Raspberry Pi буде розроблятися і компілюватися бібліотеки розробки та програмне забезпечення поведінки робота. Для забезпечення зручності розробки й налаштування ОС, було прийнято рішення насамперед налаштувати мережевий протокол SSH, для забезпечення віддаленого доступу до Raspberry Pi, що дозволить надалі використовувати Visual Studio Code як середовище віддаленої розробки.

4.2 Встановлення Raspberry Pi OS Lite

На сьогодні існує безліч операційних систем різного призначення для Raspberry Pi, при встановленні яких розробники того чи іншого образу ОС намагались охопити широкий спектр завдань, під який вона була створена, при цьому користувач отримає всі можливі потрібні і не потрібні програмні інструменти, компоненти, інтерпретатори і додатки, потоки чи сервіси яких працюють у фоновому режимі що іноді не мають якого-небудь інтерфейсу (демони), які можуть займати від 15 до 30 відсотків процесорного часу, тому було прийнято рішення вибрати Raspberry OS Lite 64-bit. По завершенні встановлення вона містить в собі встановленні інструменти для налаштування

апаратних інтерфейсів плати й готовий компілятор GNU GCC з інтерпретатором python.

Процес встановлення операційної системи здійснюється за допомогою кросплатформенного програмного інструменту Raspberry Pi Imager, який запускається на іншому комп'ютері. Програма дозволяє простим кроковим способом записати образ системи на microSD накопичувач, який з рекомендацій, повинен мати клас SD карти не нижче 10.



Рисунок 4.1 – Інтерфейс програми Raspberry Pi Imager

Після завантаження та встановлення програми Raspberry Pi Imager запустивши програму, побачимо інтерфейс (рис. 4.1) натискаємо CHOOSE OS знаходимо й вибираємо операційну систему Raspberry Pi OS Lite (64-bit) (рис. 4.2). Обравши потрібну ОС підключаємо SD карту до пристрою, через адаптер або спеціально відведений порт. Натиснувши кнопку CHOOSE STORAGE обираємо потрібний носій, та натискаємо WRITE. Після закінчення процесу

встановлення, вставляємо флеш накопичувач у слот для SD карт Raspberry Pi 4 B.

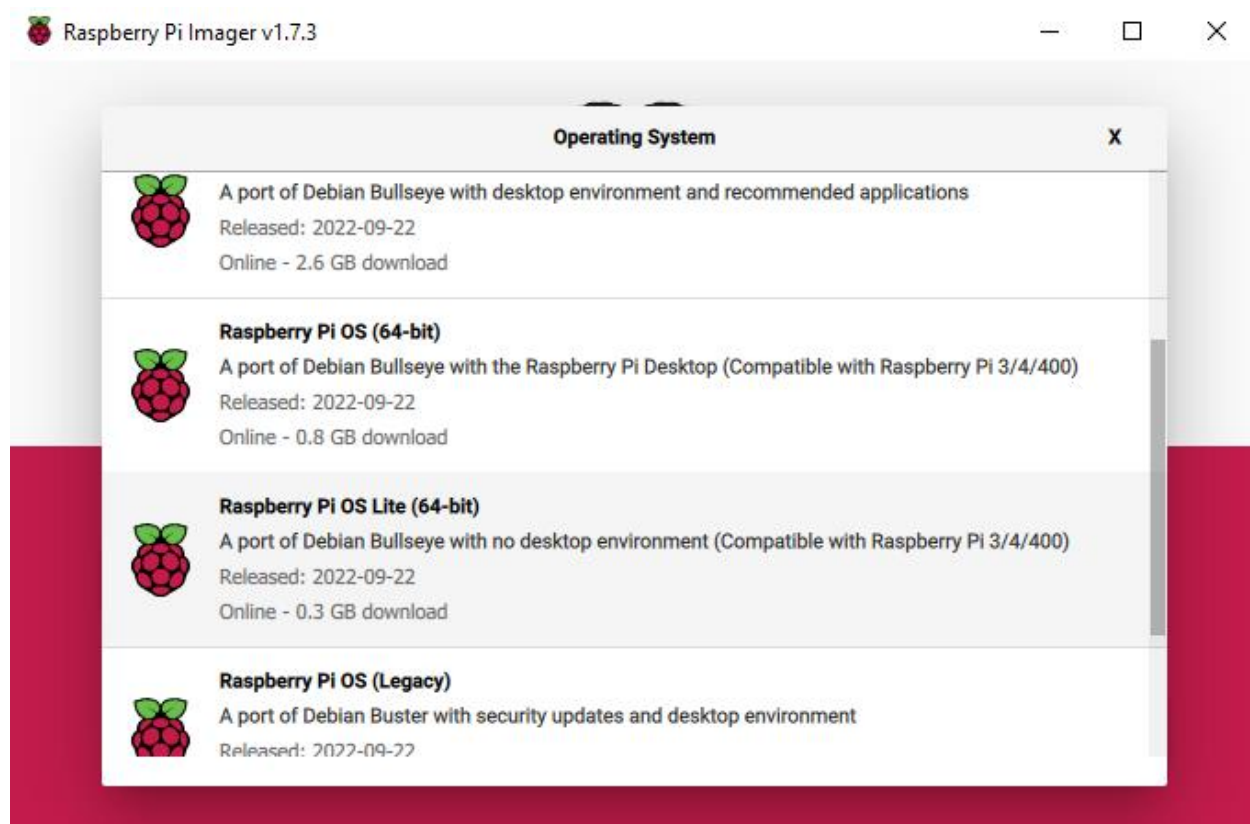


Рисунок 4.2 – Вибір Raspberry Pi OS Lite (64-bit)

При першому запуску необхідно підключити до плати миш з клавіатурою та монітор. Завершуймо процес інсталяції, вводимо логін та пароль щоб створити обліковий запис користувача, при першій же можливості налаштовуємо підключення до бездротової мережі Wi-Fi й запуск SSH серверу. Все це можна швидко зробити скориставшись командою:

```
$> sudo raspi-config
```

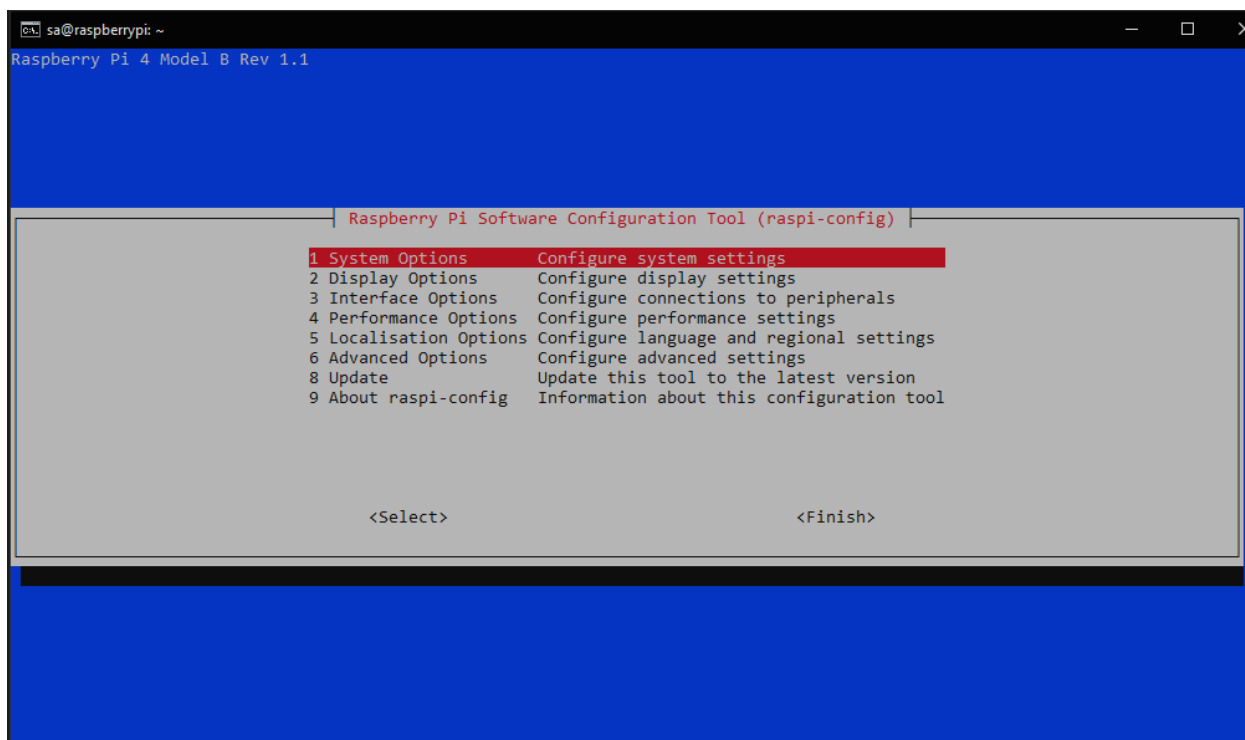


Рисунок 4.3 – Вигляд головного меню Configuration tool

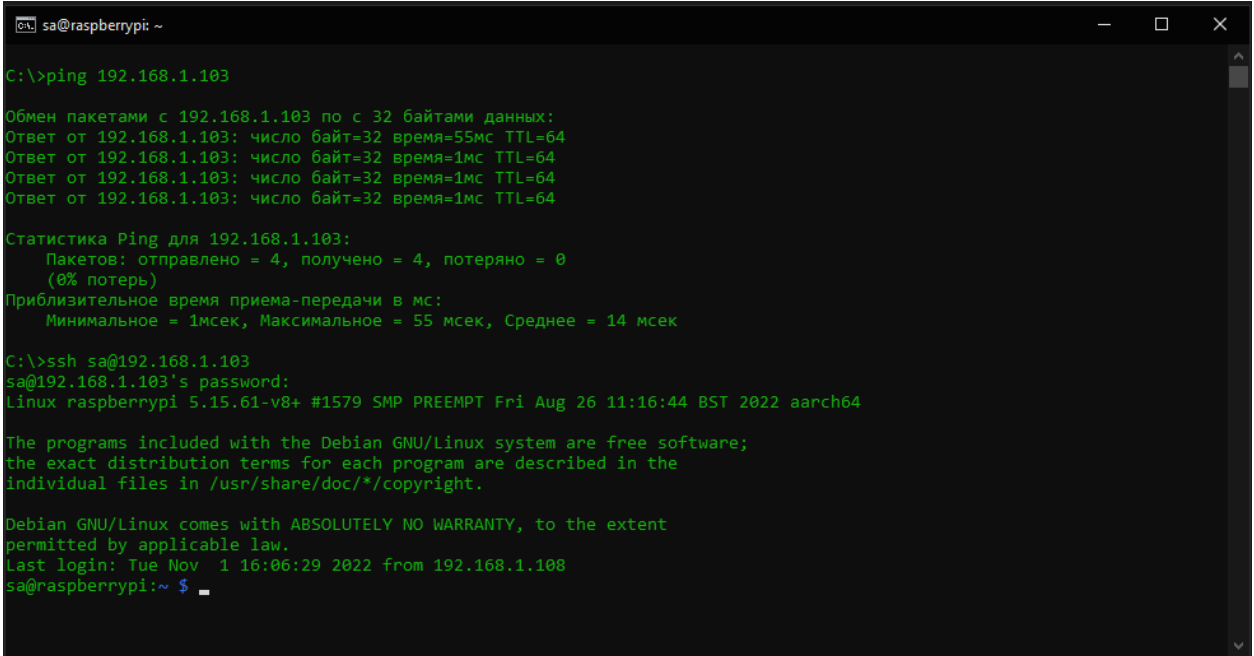
Після вводу, в оболонці малюється інтерфейс програмами конфігурації який можна побачити на рисунку 4.3. Налаштовуємо підключення до Wi-Fi переходимо до розділу System Options > Wireless LAN, вибираємо свій регіон SSID Мережі до якої хочемо підключитись та її пароль. Наступним кроком запускаємо SSH-сервер переходимо до головного меню Interface options > SSH обираємо YES у зауваженні виходимо з програмами. Дізнаємось IP адресу яку надав нам DHCP за допомогою команди у терміналі:

```
$> hostname -I
```

Дізнавшись адресу переконаємось що робочий комп'ютер має доступ до плати Raspberry Pi, у моєму випадку через cmd.exe:

```
C:\> ping 192.168.1.103
C:\> ssh sa@192.168.1.103
```

Результатом виконання команди `ping` повинні бути 4 успішні пакети, а відповідь на запит `ssh` - це введення паролю та віддалене підключення до терміналу Raspberry Pi OS Lite (рисунок 4.4).



```

sa@raspberrypi: ~
C:\>ping 192.168.1.103

Обмен пакетами с 192.168.1.103 по 32 байтами данных:
Ответ от 192.168.1.103: число байт=32 время=55мс TTL=64
Ответ от 192.168.1.103: число байт=32 время=1мс TTL=64
Ответ от 192.168.1.103: число байт=32 время=1мс TTL=64
Ответ от 192.168.1.103: число байт=32 время=1мс TTL=64

Статистика Ping для 192.168.1.103:
    Пакетов: отправлено = 4, получено = 4, потеряно = 0
    (0% потерь)
Приблизительное время приема-передачи в мс:
    Минимальное = 1мсек, Максимальное = 55 мсек, Среднее = 14 мсек

C:\>ssh sa@192.168.1.103
sa@192.168.1.103's password:
Linux raspberrypi 5.15.61-v8+ #1579 SMP PREEMPT Fri Aug 26 11:16:44 BST 2022 aarch64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Tue Nov  1 16:06:29 2022 from 192.168.1.108
sa@raspberrypi:~ $
  
```

Рисунок 4.4 – Результат виконання команд `ping` та `ssh`

4.3 Встановлення бібліотеки OpenCV

Головним модулем структури розробленого програмного забезпечення для роботи є бібліотека OpenCV. Це програмний продукт, який поширюється за двома різними ліцензіями, залежно від версії бібліотеки. На сьогодні OpenCV 4.5.0 і пізніші версії поширюються за ліцензією Apache 2 з відкритим початковим кодом. Бібліотека має у собі велику кількість оптимізованих алгоритмів для роботи із зображеннями та відео на низькому та високому рівнях, по повз алгоритмів комп'ютерного зору є рішення для машинного навчання.

Існує три способи встановлення бібліотеки на Raspberry Pi Lite OS, якщо це так можна назвати, один із найлегших способів це скористаються

скриптом інсталяції, їх можна знайти на репозиторіях системи контролю версій GitHub, завантажити й виконати, або редагувати під себе й потім виконати та дочекатися успішного завершення виконання. Другий спосіб знайти вже готову скомпільовану версію бібліотеки завантажити й розпакувавши її в зручне місце й додати в змінні середовища операційної системи та встановити бібліотеки від яких залежить OpenCV. Перший спосіб повністю розрахований покладатися на інструмент збору Cmake в об'єднанні з Geany що є легким текстовим редактором який включає базові функції IDE. Другий спосіб не завжди надає змогу отримати останню версію збірки для Raspberry OS. Третій спосіб надає більше контролю він полягає у тому щоб зібрати бібліотеку, з усіма залежними бібліотеками, з початкового коду потрібної версії у виконуваний модуль, що було зроблено при розробці програмного забезпечення для розроблюваної роботизованої системи.

Підключившись через налаштоване у попередньому розділі віддалене з'єднання SSH першим чином встановимо інструменти контролю версій, збору, завантаження, й розархівування наступною командою:

```
$> sudo apt install -y git cmake wget unzip
```

У Raspberry Pi Мікросхема фізичної оперативної пам'яті використовується як центральним процесором так і графічним процесором. На Raspberry Pi 4 за промовчанням для графічного процесора виділено об'єм пам'яті 76 Мбайт. Це може бути трохи замало для проектів зору тому треба змінити доступний обсяг на 128 Мбайт. Щоб збільшити обсяг пам'яті для графічного процесора, знову скористаємось утилітою Software Configuration Tool ввівши команду:

```
$> sudo raspi-config
```

У меню обираємо четвертий пункт Performance Option та переходимо до GPU Memory у полі зітремо старе значення й вписуємо туди потрібне. Обираймо Ok й виходимо через кнопку Finish та перезапускаємо плату наступною командою:

```
$> sudo shutdown -r now
```

Відновивши з'єднання встановлюємо потрібні бібліотеки, оскільки Raspberry Pi OS Lite не має при собі графічного інтерфейсу пропускаємо бібліотеки GTK tools, їх можна встановити якщо програмне забезпечення на етапі тестування, це дає змогу бачити результат виконання алгоритмів зору.

Встановлення потрібних бібліотек виконується за допомогою команд:

```
$> sudo apt install -y libavcodec-dev:armhf  
$> sudo apt install -y libavformat-dev:armhf  
$> sudo apt install -y libswscale-dev:armhf  
$> sudo apt install -y libv4l-dev:armhf  
$> sudo apt install -y libxvidcore-dev:armhf  
$> sudo apt install -y libx264-dev:armhf
```

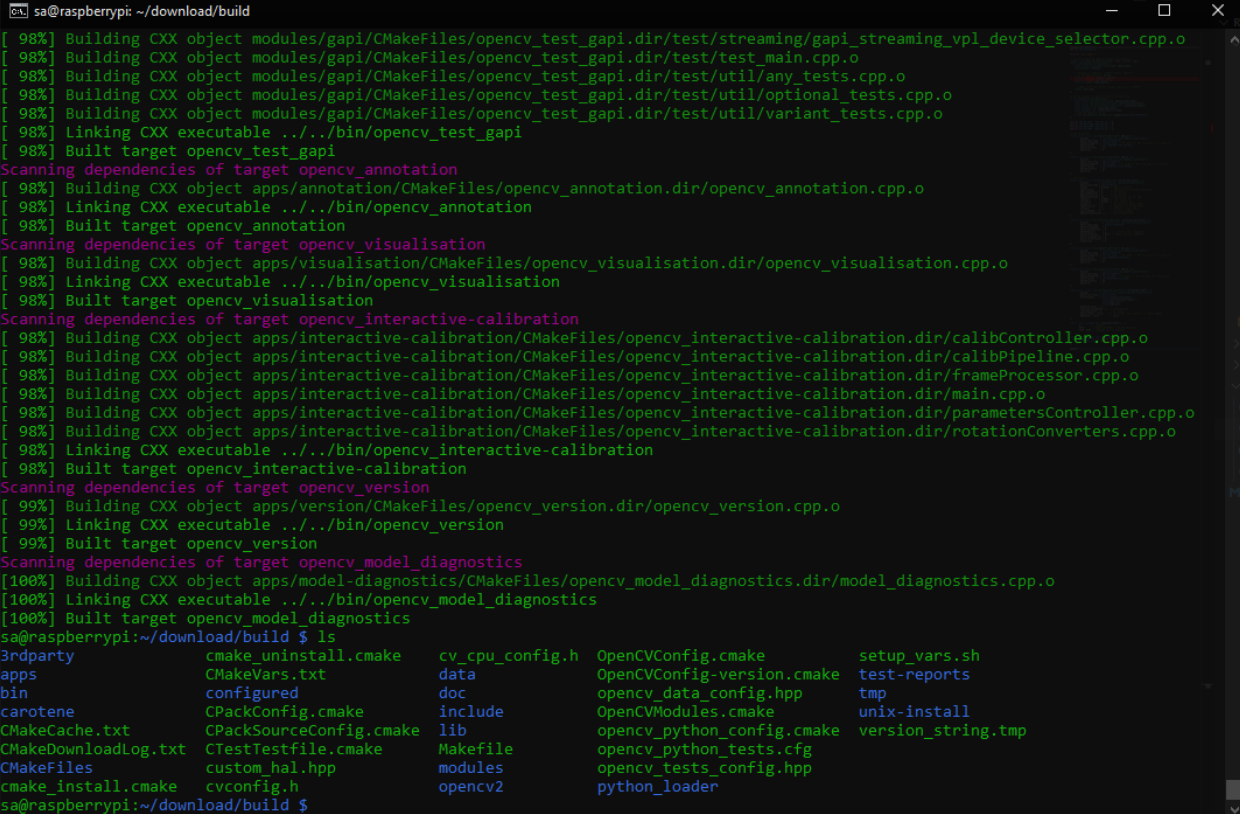
Наступним кроком завантажуюмо код бібліотеки OpenCV:

```
$> wget -O opencv.zip \  
https://github.com/opencv/opencv/archive/4.5.5.zip
```

Після отримання source code бібліотеки потрібної версії розпаковуємо архів за допомогою встановленої програми unzip та створюємо каталог – build, де буде зібрана бібліотека, у місці у якому розпакували завантажений файл. Далі переходимо до папки build й викликаємо конфігурацію, потрібний генератор збору утилі Smake за замовчанням, після завершення роботи генератору для отримання бібліотеки OpenCV, визиваємо команду make. Всі описані дії робляться за допомогою команд:

```
$> unzip opencv.zip
$> mkdir -p build && cd build
$> cmake ../opencv-4.5.5
$> cmake --build .
```

Час збору бібліотеки займає приблизно 4 години, після завершення процесу отримаємо файли (рис. 4.5).



```
sa@raspberrypi: ~/download/build
[ 98%] Building CXX object modules/gapi/CMakeFiles/opencv_test_gapi.dir/test/streaming/gapi_streaming_vpl_device_selector.cpp.o
[ 98%] Building CXX object modules/gapi/CMakeFiles/opencv_test_gapi.dir/test/test_main.cpp.o
[ 98%] Building CXX object modules/gapi/CMakeFiles/opencv_test_gapi.dir/test/util/any_tests.cpp.o
[ 98%] Building CXX object modules/gapi/CMakeFiles/opencv_test_gapi.dir/test/util/optional_tests.cpp.o
[ 98%] Building CXX object modules/gapi/CMakeFiles/opencv_test_gapi.dir/test/util/variant_tests.cpp.o
[ 98%] Linking CXX executable ../../bin/opencv_test_gapi
[ 98%] Built target opencv_test_gapi
Scanning dependencies of target opencv_annotation
[ 98%] Building CXX object apps/annotation/CMakeFiles/opencv_annotation.dir/opencv_annotation.cpp.o
[ 98%] Linking CXX executable ../../bin/opencv_annotation
[ 98%] Built target opencv_annotation
Scanning dependencies of target opencv_visualisation
[ 98%] Building CXX object apps/visualisation/CMakeFiles/opencv_visualisation.dir/opencv_visualisation.cpp.o
[ 98%] Linking CXX executable ../../bin/opencv_visualisation
[ 98%] Built target opencv_visualisation
Scanning dependencies of target opencv_interactive-calibration
[ 98%] Building CXX object apps/interactive-calibration/CMakeFiles/opencv_interactive-calibration.dir/calibController.cpp.o
[ 98%] Building CXX object apps/interactive-calibration/CMakeFiles/opencv_interactive-calibration.dir/calibPipeline.cpp.o
[ 98%] Building CXX object apps/interactive-calibration/CMakeFiles/opencv_interactive-calibration.dir/frameProcessor.cpp.o
[ 98%] Building CXX object apps/interactive-calibration/CMakeFiles/opencv_interactive-calibration.dir/main.cpp.o
[ 98%] Building CXX object apps/interactive-calibration/CMakeFiles/opencv_interactive-calibration.dir/parametersController.cpp.o
[ 98%] Building CXX object apps/interactive-calibration/CMakeFiles/opencv_interactive-calibration.dir/rotationConverters.cpp.o
[ 98%] Linking CXX executable ../../bin/opencv_interactive-calibration
[ 98%] Built target opencv_interactive-calibration
Scanning dependencies of target opencv_version
[ 99%] Building CXX object apps/version/CMakeFiles/opencv_version.dir/opencv_version.cpp.o
[ 99%] Linking CXX executable ../../bin/opencv_version
[ 99%] Built target opencv_version
Scanning dependencies of target opencv_model_diagnostics
[100%] Building CXX object apps/model-diagnostics/CMakeFiles/opencv_model_diagnostics.dir/model_diagnostics.cpp.o
[100%] Linking CXX executable ../../bin/opencv_model_diagnostics
[100%] Built target opencv_model_diagnostics
sa@raspberrypi:~/download/build $ ls
3rdparty          cmake_uninstall.cmake  cv_cpu_config.h      OpenCVConfig.cmake      setup_vars.sh
apps              CMakeVars.txt          data                 OpenCVConfig-version.cmake  test-reports
bin               configured              doc                  opencv_data_config.hpp   tmp
carotene          CPackConfig.cmake      include              OpenCVModules.cmake      unix-install
CMakeCache.txt    CPackSourceConfig.cmake lib                  opencv_python_config.cmake version_string.tmp
CMakeDownloadLog.txt CTestTestFile.cmake    Makefile             opencv_python_tests.cfg
CMakeFiles        custom_hal.hpp          modules              opencv_tests_config.hpp
cmake_install.cmake cvconfig.h              opencv2              python_loader
sa@raspberrypi:~/download/build $
```

Рисунок 4.5 – Список отриманих каталогів після збору бібліотеки

Папку з назвою lib треба додати у змінні середовища операційної системи:

```
$> nano /etc/ld.so.conf.d/opencv.conf
```

У текстовому редакторі вписуємо шлях до папки lib у моєму випадку /usr/local/lib/ на тискаємо Ctrl+X згодом Y та тиснемо Enter, та вводимо команду:

```
$> sudo ldconfig -v
```

Після чого знову перезавантажимо систему.

4.4 Встановлення бібліотеки WiringPi

При встановленні операційної системи з офіційного репозиторію до Raspberri Pi не поставляється інструмент для роботи з контактами вводу виводу на платі. виправити це можна завдяки бібліотеці WiringPi яка здійснює доступ до GPIO на основі PIN - коду яка написана на C для пристроїв SoC BCM2835, BCM2836 і BCM2837, що використовуються у всіх версіях Raspberry Pi. Програмний код не є загальнодоступним, але може бути наданий тим, хто бажає отримати комерційну підтримку.

На додаток бібліотека WiringPi включає утиліту командного рядка gpio, яку можна використовувати для програмування та налаштування контактів GPIO, це надає змогу керувати станом контактів зі сценаріїв оболонки.

WiringPi розроблюється безпосередньо на Raspberry Pi з 32 або 64 бітним Raspbian. Це одна з основних причин чому вибір впав на Raspberry Pi Lite OS x64, а не на неофіційну Raspberry Pi Debian OS, це мало ймовірна крос-компіляція на інші операційні системи. Мало інформації про портування на інші платформи та інші операційні системи.

Для того, щоб отримати бібліотеку можна завантажити програмний пакет за посиланням <https://github.com/WiringPi/WiringPi/releases/download/2.61-1/wiringpi-2.61-1-arm64.deb>, ввівши в терміналі віддаленого доступу:

```
$>wget  
https://github.com/WiringPi/WiringPi/releases/download/2.61-  
1/wiringpi-2.61-1-arm64.deb
```

Щоб встановити програмне забезпечення, що міститься у файлі DEB, використовуємо команду:

```
$> sudo dpkg -i wiringpi-2.61-1-arm64.deb
```

4.5 Налаштування системи розпізнавання зображень

Для виконання завдання розпізнавання об'єктів на зображеннях отриманих з кадру камери, розташованої на дроні, використовуються алгоритми нейронної мережі. Алгоритми нейронних мереж розроблені для комп'ютерного зору підходять для завдань ідентифікації зображень та його класифікації на одному з кількох зумовлених окремих класів. Таким чином, програмне забезпечення з нейронною мережею для розпізнавання зображень дозволять керуючій системі дрону визначити, що зображено на кадрі, і відрізнити один об'єкт від іншого.

Як відомо нейронна мережа являє собою потужну, що задовольняє на сьогодні, математичну реалізацію біологічних фізико-хімічних процесів, що відбуваються в різноманітності зав'язків нейронних клітин. Для того, щоб мережа приймала задану кількість вхідних даних, а потім обчислювала задану кількість вихідних, спрямованих на досягнення очікуваного фактичного результату, застосовується ряд математичних методів, що поєднують лінійну алгебру та статистику і не тільки, для вирішення задачі нейронної мережі. Тому при реалізації алгоритмів нейронної мережі була використана бібліотека Eigen. Вона має із себе набір рішень для лінійної алгебри, що дозволяє обчислювати матриці, вектори, численні вирішування та відповідні алгоритми, і все це у вигляді заголовних фалів шаблонів, які не знаходяться ні в яких залежностях, окрім стандартної бібліотеки C++, це чиста бібліотека шаблонів, визначена в заголовках. Головною цікавою можливістю є підтримка

всіх розмірів матриць, від невеликих фіксованих розмірів до доволіно великих щільних і навіть розрізнених матриць. Завдяки цьому Eigen виступає потужним інструментом при розробці нейронної мережі й алгоритмів кластеризації класифікації й бачення.

Розглянута бібліотека є вільним програмним забезпеченням, починаючи з версії 3.1.1, вона поширюється за ліцензією MPL2. Отримати останню стабільну копію бібліотеки можна здійснивши клонування з репозиторію GitHub:

```
$> git clone https://github.com/libigl/eigen.git
```

Після успішного клонування єдина важлива папка, на яку буде звернено увагу, це каталог Eigen, вміщуючий всі потрібні шаблони, який буде включено до робочої папки проекту, що буде створено у наступному підрозділі.

4.6 Встановлення FTP-серверу

Це відомий мережевий протокол, що використовується для передачі файлів між двома хостами. Для реалізації роботи протоколу на Raspberry Pi була використана утиліта vsftpd. Це легка та проста у використанні програма, яка дозволяє запустити FTP сервер. Для встановлення програми виконаємо команду терміналі.

```
$> sudo apt -y install vsftpd sudo nano /etc/vsftpd.conf
```

По завершенню встановлення утиліти її роботу треба налаштувати через конфігураційний файл що знаходиться у каталозі etc. Відкриваємо його за

допомогою вбудованого текстового редактору nano у операційній системі Raspberry Pi Lite OS наступною командою.

```
$> sudo nano /etc/vsftpd.conf
```

Видаливши знак решітки розкоментуємо наступні інструкції: `write_enable=YES`; `local_umask=022`; `chroot_local_user=YES`. Інструкцію `anonymous_enable=YES` змінюємо на `anonymous_enable=NO` та наприкінці конфігураційного файлу додаємо наш робочий каталог вписавши `user_sub_token=$USER` й `local_root=/home/$USER/FTP`.

Оскільки кореневий каталог `home/$USER/FTP` не може мати дозволу на запис, створюємо підкаталог наступною командою у терміналі:

```
$> mkdir -p /home/sa/FTP/droneTasks
```

Після всіх потрібних маніпуляцій треба перезапустити сервіс `vsftpd`, для того щоб зміни вступили в силу, виконавши команду:

```
$> sudo service vsftpd restart
```

4.7 Налаштування віддаленої середовища розробки SSH

Для реалізації віддаленого середовища розробки було використано текстовий редактор програмного коду від компанії Microsoft - Visual Studio Code. Програмний продукт має можливість підтримки інсталювання різного призначення розширень, безпосередньо з редактора, включаючи ті які забезпечуватимуть віддалену розробку за допомогою SSH з'єднання. Програма є вільно розповсюдженим кросплатформеним продуктом з відкритим кодом,

зібрану останню версію якого можна завантажити за посиланням Microsoft за посиланням: <https://code.visualstudio.com/download>.

По завершенню процесу інсталяції для нормальної роботи через SSH з'єднання потребується завантажити розширення, а саме Remote Development та Remote – SSH(Nightly). Завершивши встановлення розширень необхідно перезавантажити вікно редактору. Після цього натискаємо на зелену іконку з ліва у низу (рис. 4.6) Open a Remote Windows, далі у полі, обираємо пункт Connect to host й приписуємо SSH sa@192.168.1.103. Після введення паролю обираємо Linux й чекаємо, декілька секунд поки на віддалений пристрій завантажаться необхідні бібліотеки для роботи у режимі SSH.

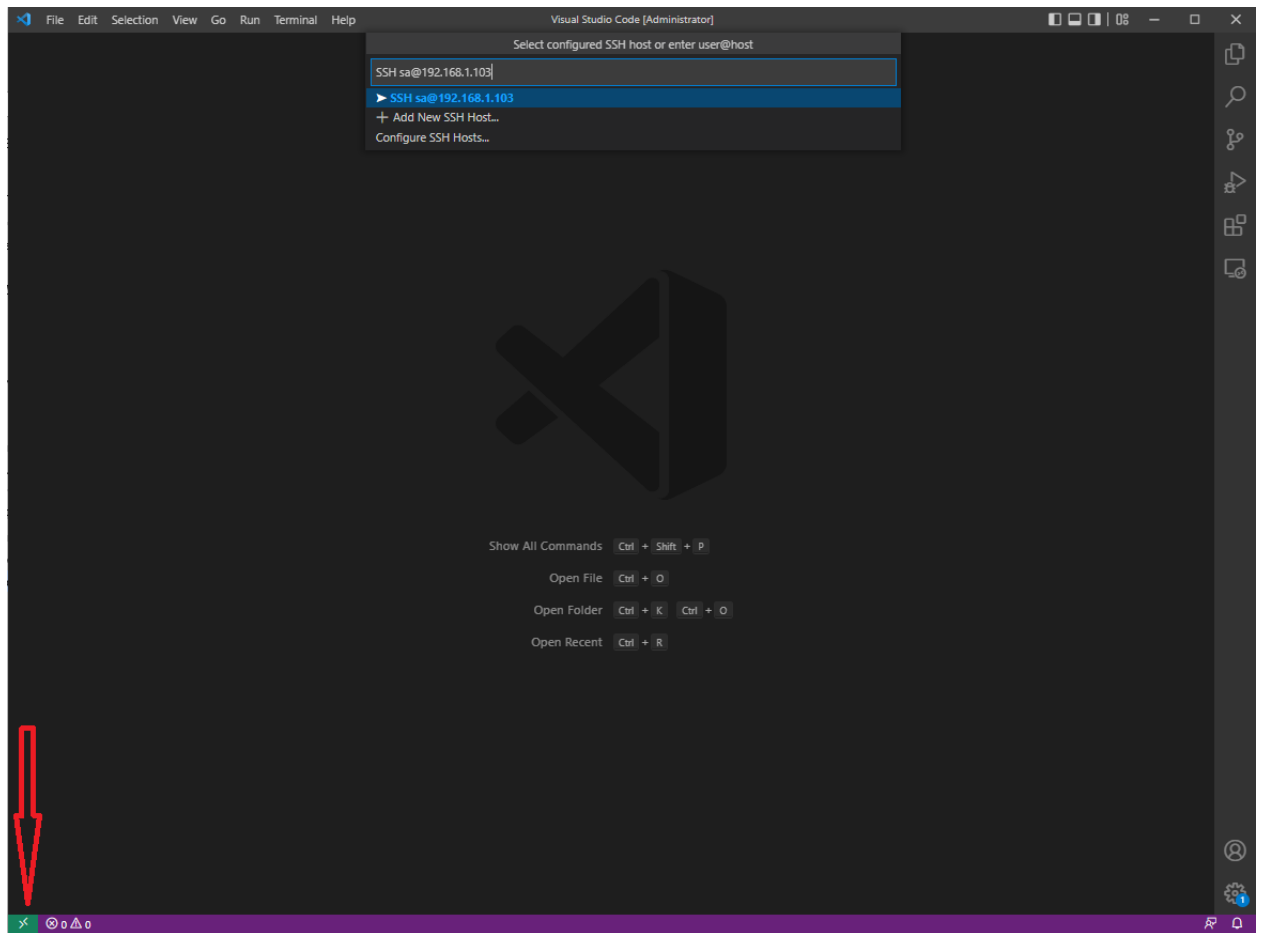


Рисунок 4.6 – Приклад процесу реалізації SSH з'єднання

Поки йде процес підключення, через раніше відкрите віддалене з'єднання було створено робочу папку проекту PjAWC - project autonom water currier у якій було створено ще три папки src, bin та Dependencies.

Після вдалого процесу підключення та встановлення необхідних бібліотек на місці зеленої іконки додається ще й адреса віддаленого пристрою. Натиснувши на кнопку Open Folder надається вибір обрати будь-який каталог (рис. 4.7), звідки буде відлік доступу у вбудованому файловому менеджері редактору.

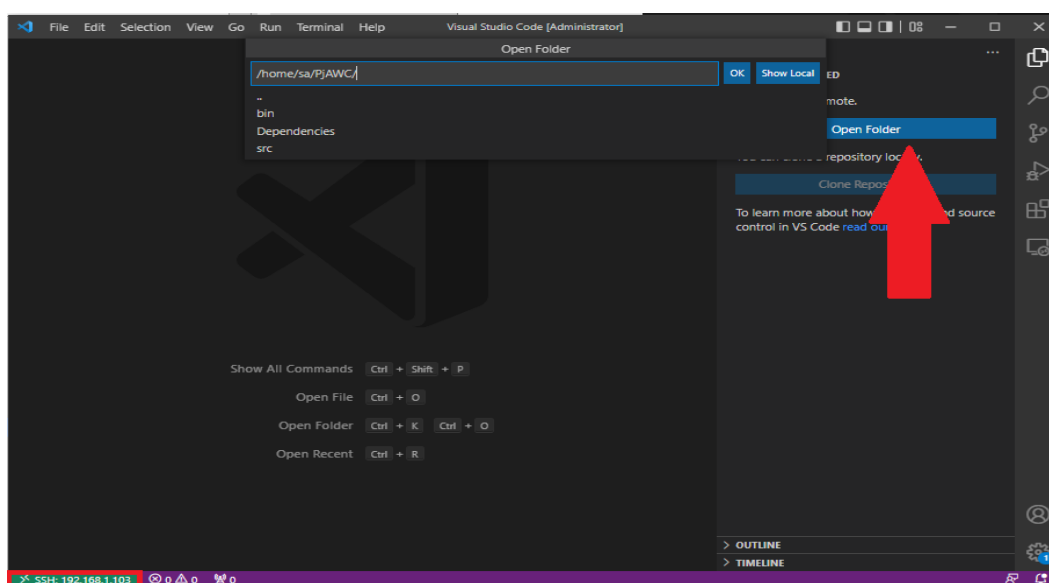


Рисунок 4.7 – Вибір робочої до папки проекту на віддаленому хості

4.8 Особливості роботи системи

При включенні дрону робот повинен перебувати на док-станції або поряд з нею, а док-станція повинна знаходитись у зоні досяжності Wi-Fi, бо комунікація дрону і док-станції здійснюється через точку доступу. Док-станція слугує локальною точкою системи відліку координат для робота, завантаживши дані розміри дослідження території та об'єктах поливу дрон

починає перший свій виїзд який є дослідженням території в процесі якого він отримує трьох вимірну карту місцевості та положення об'єктів поливу, завдяки даним з гіроскопів лідара і камери.

З даних температури і вологості дрон прогнозує час рівня вологості ґрунту й має можливість фактично підтвердити наявність опадів або їх присутності, таким чином система прогнозує полив в залежності від типу рослини і обстановки навколишньої середовища. Здійснивши аналіз на виході дрон має унікальну 3д карту місцевості з необхідними мітками кутами нахилу рельєфу й розташуванням об'єктів поливу. Полив здійснюється за допомогою виділеного для рослини літражу, обчислюється за часом відкриття клапана подачі води, таким чином обчислюється й об'єм, що залишився в баку, по спустошенню якого дрон прямує до док-станції, пакується і поповняється рівень води, док-станція наповняє рівень води по тому ж принципу, час відкриття клапана подачі води з передбачуваним постійним потоком через переріз труби.

Raspberry PI, що реалізує блок керування дроном розрахована працювати під керуванням операційної системи на якій працює розроблене програмне забезпечення, що визначає поведінку роботизованої системи отримуючи необхідні дані з датчиків розташованих на дроні. Присутні модулі які мають стандартизовані протоколи передачі даних здійснюють обмін інформацією за допомогою спеціально розроблених бібліотек. Пристрої, які реалізують власну шину передачі даних, наприклад датчик DHT22, реалізує свій протокол передачі даних. Резистор на лінії DATA без поданого нижнього рівня сигналу означає стан IDLE мікрокомп'ютер raspberry Pi діє, як майстер шини й отримує можливість ініціювати зв'язок, зчитувати дані.

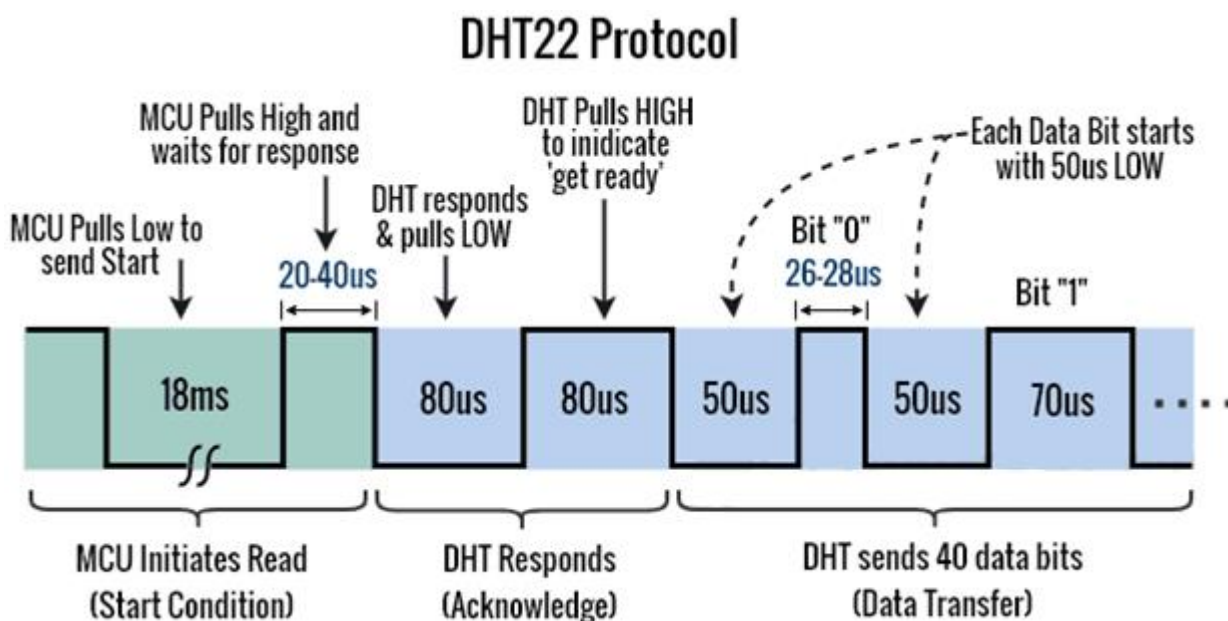


Рисунок 4.8 – Протокол передачі даних датчику DHT22

Приведений датчик вологості й температури DHT22 завжди залишається підлеглим та відповідає даними, коли CPU 0 запитує їх. Протокол, комунікаційний процес який можна побачити на рисунку 4.8, використовується для зв'язку. Протокол передачі даних може бути узагальнений таким чином:

- коли лінія перебуває у стані IDLE, мікрокомп'ютер переводить її у стан LOW на 18 мс;
- після цього CPU 0 піднімає його HIGH приблизно на 20-40 мкс;
- DHT22 виявить це як START від CPU 0 та відповідь, потягнувши лінію LOW на 80 мкс;
- потім DHT22 встановить високе значення на 80 мкс, що вказує на те, що він готовий до відправлення даних або "готується";
- далі пристрій надішле 40 біт даних, кожен біт з яких починається з 50 мкс LOW, за яким слідує 26-28 мкс для "0" або 70 мкс для "1";
- після завершення зв'язку підтягуючий резистор R4 на лінії встановлює HIGH рівень, і датчик переводиться в стан IDLE.

Коли датчик вологості та температури надсилає дані, він спочатку надсилає нумерацію бітів MSB. Дані на рисунку 4.9 структура даних

складається з 40 біт та поділена на 5 байтів. Значення першого і другого байту представляють інтегральні дані відносної вологості у процентному відношенні, перший біт ціла частина другий дробова частина. Наступні два байти третій і четвертий представляють інтеграл температури градусів Цельсія. П'ятий байт контрольна сума.

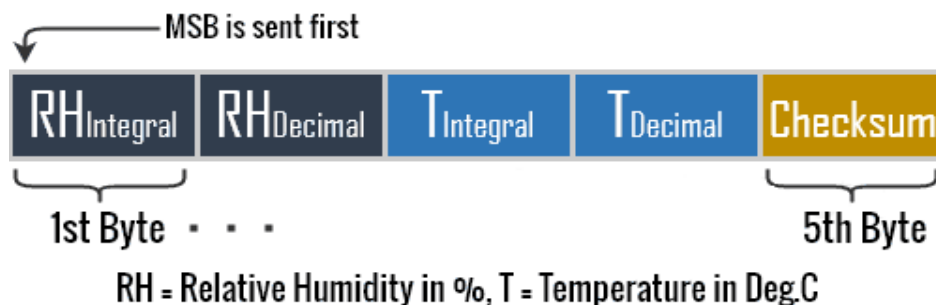


Рисунок 4.9 – Структура даних DHT22

Управління двигунами за допомогою контролерів Volta 36-48v500w відбувається за рахунок подачі логічних рівнів на інтерфейс, що представляє собою роз'єми на пристрої.

Роз'єм Throttle відповідає режим потужності, на схемі чорний і коричневий виводи з'єднані, що відповідає 50% потужності, з документації пристрою, чого буде цілком достатньо для дрону. Для того щоб регулювати швидкість у 10-20% використовується роз'єм Cruise, що складається з рожевого і чорного виводів, рожевий висить і завжди подає низький логічний рівень. Якщо на чорний висновок подати низький рівень, замкнути рожевий і чорний, то активується круїз-контроль, зберігається швидкість обертання або потужність, що подається на двигуни.

Роз'єм Brake має два дроти і працює аналогічним чином, чорний вихід приймає логічний нуль, здійснюючи гальмо і припиняючи газ. На схемі логічний нуль подає мікроконтролер AVR ATtiny24. Він веде час від початку припинення подачі логічного нуля на Brake, активується газ, і в залежності від прийнятих даних з мікрокомп'ютера, вичікує час і активує круїз-режим. Таким

чином регулюється швидкість пересування дрона. Мікроконтролер AVR ATtiny24 може подавати сигнали на кожен контролер незалежно з певною затримкою або тільки на один пристрій, що провокує відхилення в момент руху роботи або його розворот.

4.9 Результат роботи системи

Навігація дрону ґрунтується на даних камери лідера та GPS модуля. Дані з камери приймаються та обробляються за допомогою інструментів бібліотеки OpenCV, отримуючи на виході RGB зображення 480 на 640 пікселів, що передається нейронної мережі RCNN для виявлення об'єктів. Початок роботи мережі (рис. 4.10) знаходить у роботі шарах згорткової нейронної мережі, в цих шарах фільтрам витягуються відповідні ознаки зображення, основні признаки виду дерев, кущів, каменів, клумб, цей блок фільтрів змінюється, на інші свідомо навчені, залежно від пори року, часу доби, або високої хмарності, фільтри навчені за допомогою навчання форм і кольорів, це дозволяє на виході нейронної мережі підвищити результат асиміляції згортки бажаних об'єктів, що існують у саду.

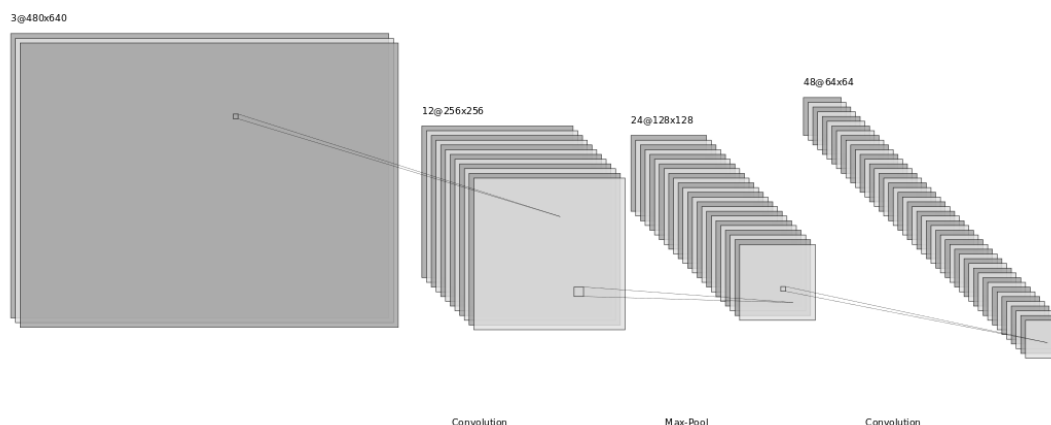


Рисунок 4.10 – Архітектура першого загорткового шару нейронної мережі

Результат роботи загорткової нейронної мережі переходить до регіональної мережі пропозицій RPN (з англійської Region Proposal Network), яка проходить за результатами об'єктів загорткових шарів, її робота передбачати, чи є об'єкт чи ні, а також прогнозувати рамку об'єктів, що обмежує, результатом цієї мережі є вихідний вектор 1×256 після функції SoftMax. Ймовірні значення, регіональної мережі пропозицій, надходять як вхідні дані на повно зв'язану нейронну мережу яка класифікує клас об'єкту й пророкує обмежувальні рамки – регресію.

Для навчання цієї архітектури використовувався метод стохастичного градієнтного спуску для оптимізації фільтрів шарів згортки, ваг RPN та ваги останніх повно зв'язкових шарів нейронної мережі. Для переконання у правильності роботи алгоритму нейронної мережі а також для перевірки ефективності зміни фільтрів та їх використання та налаштування був використаний програмний продукт Unreal Engine 5 з використанням відсканованих фотореалістичних дерев у різній стадії віку у різних варіантів ландшафту саду (рис. 4.11).

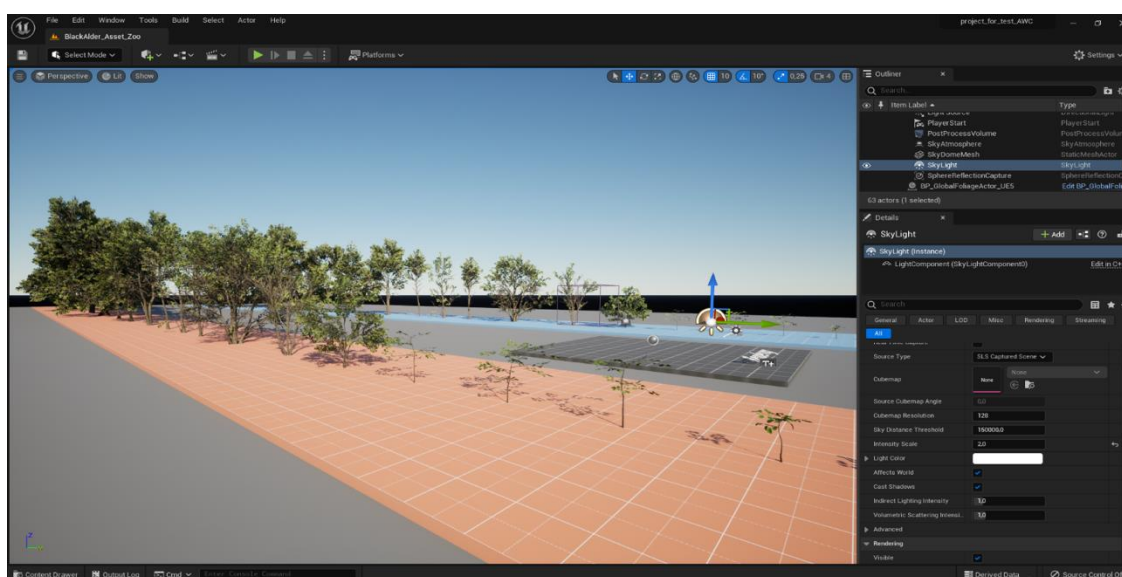


Рисунок 4.11 – Магаскан дерев у різній стадії віку у редакторі Unreal Engine

Досліджуючи дрон в момент руху завдяки роботі нейронній мережі (рисунок 4.12) ідентифікує клас пікселів наприклад газон, дані на виході нейронної мережі зіставляються з даними лідара про відстань об'єкта і його положення - гіроскопів, на основі цих даних будується тривимірна карта простору завдяки чому система робить висновок що, об'єкт пікселів газону лежить у площині нижче коліс й продовжує рух. Коли у напрямку руху дрону виникає об'єкт що надходить до високої вірогідності до класу рослини він наближаться, на відстані двох с половиною метрів та зупиняється далі система запускає у роботу іншу нейронну мережу заточену на розпізнання рослини. По завершенню ідентифікації що це є рослина, дрон звіряє її з набором фотографій які загрузив користувач на сервер дрону. Якщо відповідна рослина була знайдена дрон відмічає її на трьохмірній мапі.

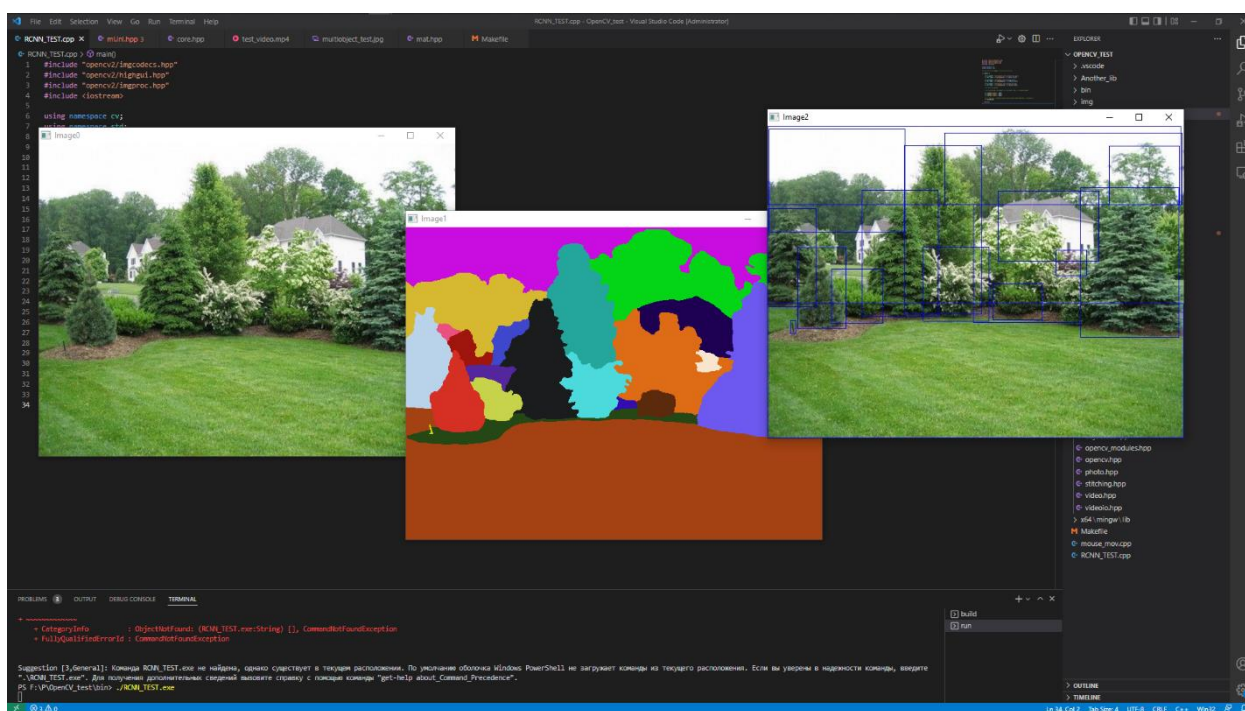


Рисунок 4.12 – Результат роботи алгоритму нейронної мережі RCNN

Вказати роботизованій системі область обслуговування, а також рослини, що вимагають поливу, користувач може підключившись до дрону за допомогою будь-якого файлового менеджера підтримуючого FTP передачу. Підключившись через FTP клієнт до каталогу /home/sa/FTP/droneTasks у папці

droneTasks знаходиться файл sector.txt, у який користувач повинен внести не менше двох точок, вказаних через символ перенесення рядка, на карті використовуючи google maps, це обмежує область дослідження та подальшу роботу дрону. Для вказівки об'єкта поливу користувачеві необхідно створити папку у droneTasks й помістити туди фотографії однієї й тієї ж рослини для того, щоб у алгоритмів було більше прикладів для навчання розпізнавання та класифікації, що збільшує ймовірність розпізнавання об'єкта серед інших. У папці з фотографіями об'єкта також потрібно створити текстовий файл з двома дійсними числами розділеними символом перенесення рядка (\n). Перше число відповідає за період поливу, відлік ведеться в тижнях, друге число це витрата води за один полив рослиною обчислюється в літрах.

Для прикладу на рисунку 4.13 використовуються FTP клієнт FileZilla завдяки якому був створений каталог з назвою Arborvita до якого було завантажено дев'ять зображень туї, та створено файл ArborvitaeTask.txt у який записано через \n два числа 1 і 12 система дрону інтерпретує це, як вказівку, що рослину треба поливати один раз на тиждень, витрачаючи 12 літрів на полив.

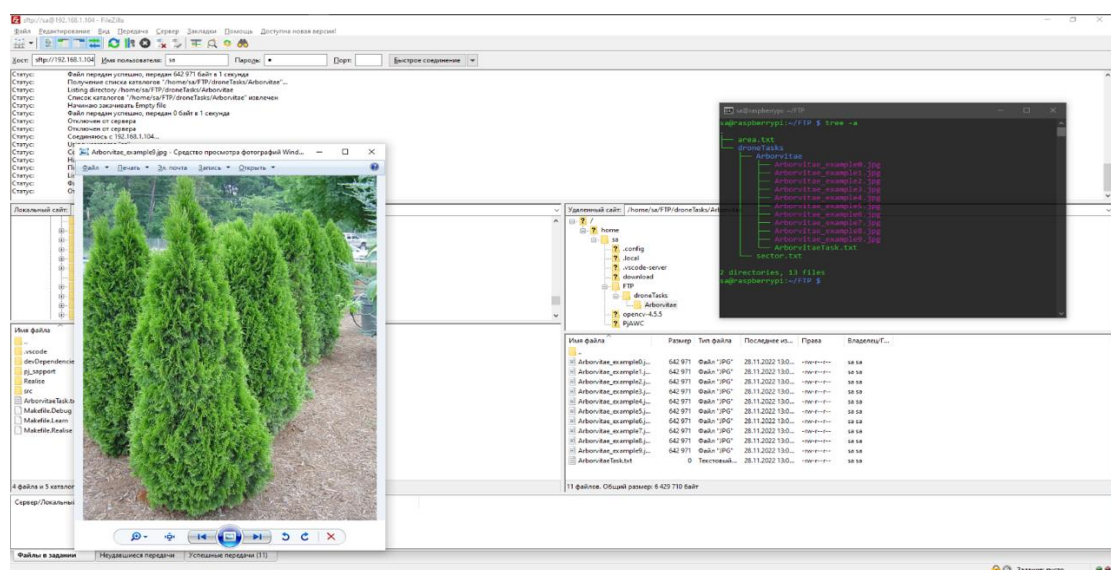


Рисунок 4.13 – приклад завантаження файлів за допомогою FTP клієнту

FileZilla

ВИСНОВКИ

Таким чином, у магістерській роботі автоматизовано процес управління роботою дрону на базі розробленого апаратно-програмного комплексу роботизованої системи. При цьому було розглянуто і проаналізовано підходи щодо створення схожих роботизованих систем та їх апаратної частини з методами навігації у просторі й засоби взаємодії апаратної системи з оточенням.

Розроблено апаратну й програмну складову дрону, що дозволяє йому в автоматичному режимі виконувати полив рослин, які заздалегідь вказані. Система адаптується під навколишнє середовище й відповідно реагує на погодні умови.

Проведено тестування на правильність роботи нейронної мережі у програмному продукті Unreal Engine 5.

При виконанні дипломної роботи була розроблена система, що об'єднує загальнодоступні модулі в єдиний комплекс, що забезпечує автономну роботу поливу рослин. Завдяки цьому обслуговування та заміна комплектуючих дрону проводиться з мінімальними зусиллями.

Таким чином, отримано продукт, що поєднує переваги інших подібних роботизованих систем та має низьку вартість й високий потенціал для майбутнього розвитку.

Концепція роботизованої системи на базі автономного дрону передбачає новий підхід у побудові життєвого процесу, на основі об'єднання різного комплексу інноваційних технічних модульних пристроїв і рішень, отримана автономна система поливу рослин призначена розширити комфорт користувачів за рахунок надання їм управління повністю автоматизованої системи поливу.

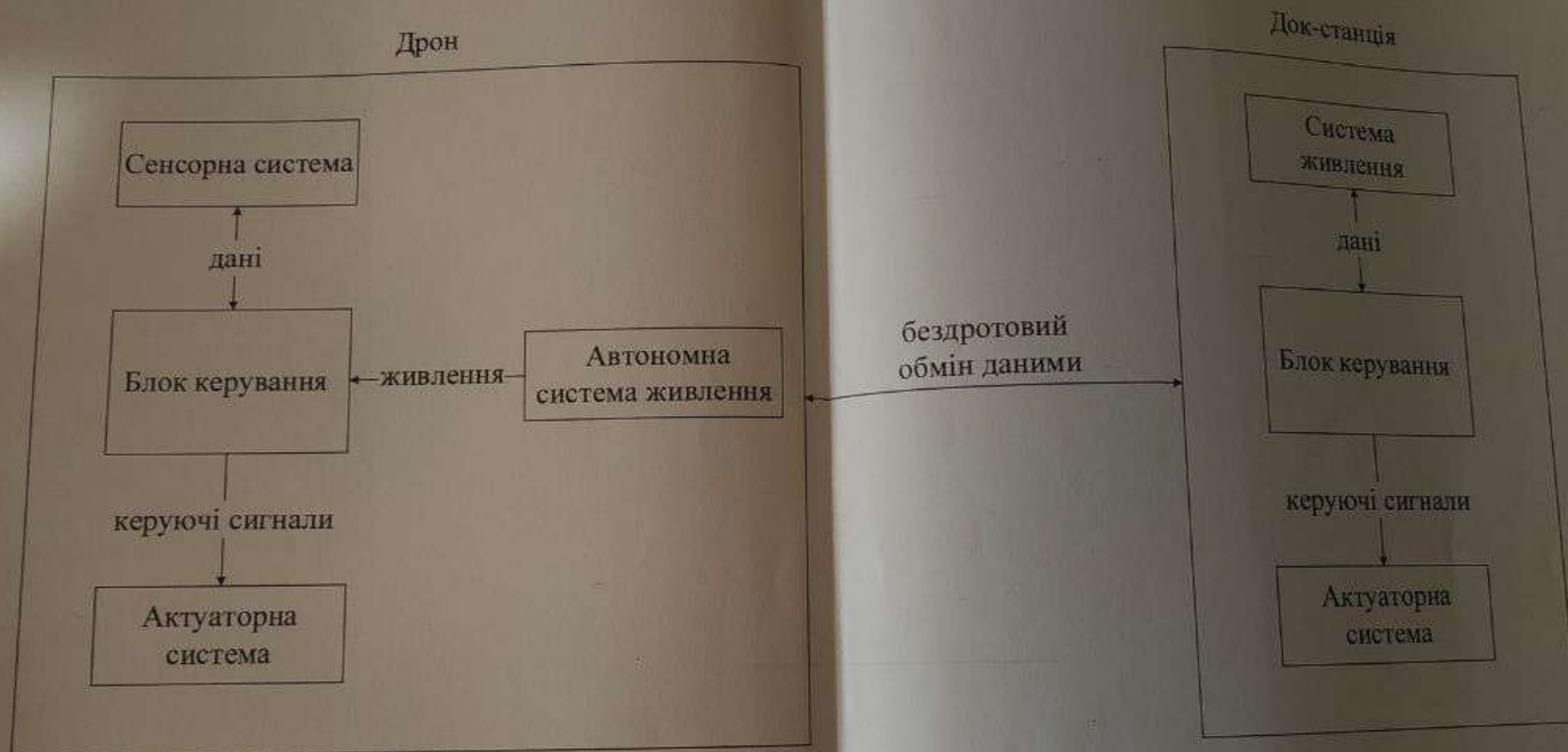
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. C++ Neural Networks and Fuzzy Logic Paperback – 1 Oct. 1995
2. Learning OpenCV: Computer Vision with the OpenCV Library
3. Hands-On Machine Learning with C++: Build, train, and deploy end-to-end machine learning and deep learning pipelines Paperback – 15 May 2020
The CNC Cookbook: An Introduction to the Creation and Operation of Computer Controlled Mills, Router Tables, Lathes, and More - Scited Publications 2009.
4. AVR Microcontroller and Embedded Systems: Pearson New International Edition: Using Assembly and C Muhammad Ali Mazidi, - DeVry University 2014
5. Learning Deep Learning: Theory and Practice of Neural Networks, Computer Vision, Natural Language Processing, and Transformers Using TensorFlow 1st Edition
6. Programming the Raspberry Pi: Getting Started with Python by Simon Monk- McGraw-Hill Education Tab 2012
7. Raspberry Pi Pico documentation. Режим доступу: <https://www.raspberrypi.com/documentation/microcontrollers/raspberry-pi-pico.html>
8. SPI with Raspberry Pi: вебсайт. Режим доступу: <https://core-electronics.com.au/tutorials/i-spi-with-raspberry-pi.html>
9. Building Computer Vision Projects with OpenCV 4 and C++: Implement complex computer vision algorithms and explore deep learning and face detection
10. ATtiny24/L Datasheet: вебсайт. Режим дуступу: https://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-7701_Automotive-Microcontrollers-ATtiny24-44-84_Datasheet.pdf

11. Глубокое обучение ДМК Москва 2018
12. Basics of Linear Algebra for Machine Learning: Discover the Mathematical Learning Discover the Mathematical Language of Data in Python Originally published: January 24, 2018
13. Agricultural robot dataset for plant classification, localization and mapping on sugar beet fields Article in The International Journal of Robotics Research · July 2017
14. Digital Image Semantic Segmentation Algorithms: A Survey Received June 2018; revised October 2018
15. Dynamics and Control of Robotic Systems 1st Edition by Andrew J. Kurdila (Author), Pinhas Ben-Tzvi (Author) December 16, 2019
16. Richards M. Software architecture patterns: O'Reilly Media, 2015. – 302 p.
17. Цифровая схемотехника и архитектура компьютера второе издание Дэвид М. Харрис и Сара Л. Харрис Издательство Morgan Kaufman © English Edition 2013
18. SD cards - Raspberry Pi Documentation: вебсайт. Режим доступа: <https://www.raspberrypi.org/documentation/installation/sd-cards.md> (дата звернения: 12.10.2022)
19. Installing operating system images - Raspberry Pi Documentation: вебсайт. Режим доступа: <https://www.raspberrypi.org/documentation/installation/installing-images> (дата звернения: 12.10.2022)
20. OpenCV API Documentation: вебсайт. Режим доступа: <https://docs.opencv.org/4.5.5/index.html> (дата звернения: 12.10.2022)
21. Raspberry Pi Pico API Documentation: вебсайт. Режим доступа: <https://raspberrypi.github.io/pico-sdk-doxygen/modules.html> (дата звернения: 12.10.2022)

22. Basics of Interfacing DHT11/DHT22 Protocol: вебсайт. Режим доступу: <http://www.ocfreaks.com/basics-interfacing-dht11-dht22-humidity-temperature-sensor-mcu/> (дата звернення: 20.11.2022)
23. Raspberry Pi FTP Server Setup: вебсайт. Режим доступу: <https://phoenixnap.com/kb/raspberry-pi-ftp-server> (дата звернення: 29.11.2022)
24. Product Manual of TF-Luna
25. Datasheet HCPL-260L/060L/263L/063L

Загальна структурна схема роботизованої системи



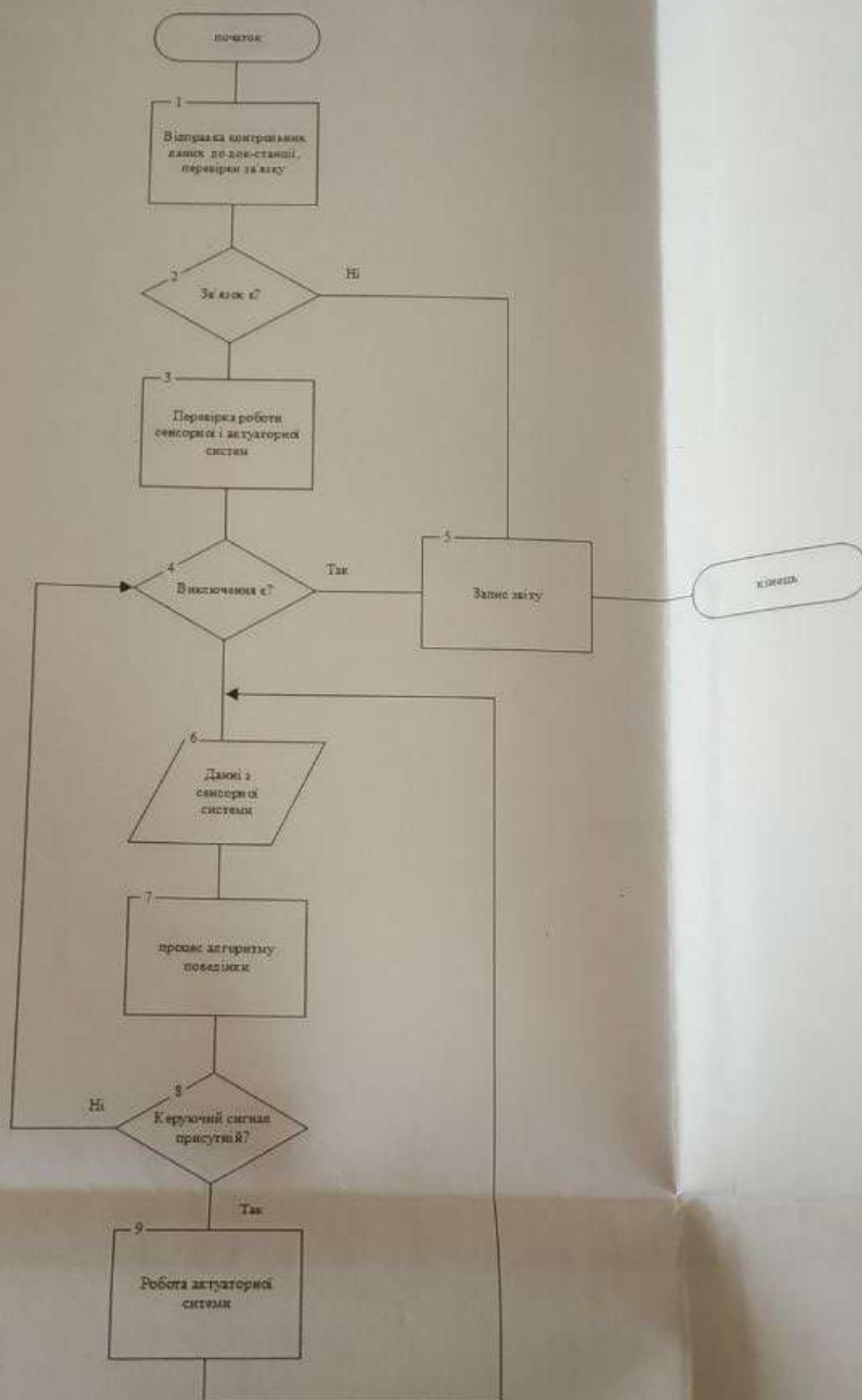
Зам.	Лист	№ док-м.	Підп.	Дата
Розроб.		Титюков О.С.		
Перев.		Тягунова М.Ю.		
Т.контр.				
Н.контр.		Щербак Н.В.		
Затв.		Кудрявцев Р.К.		

13.02070849.00014 ПЛ1

Загальна структурна схема
роботизованої системи

Лист	Маса	Масштаб
Лист 1		Листів 1

НУ «Запорізька політехніка»
КНТ-511м



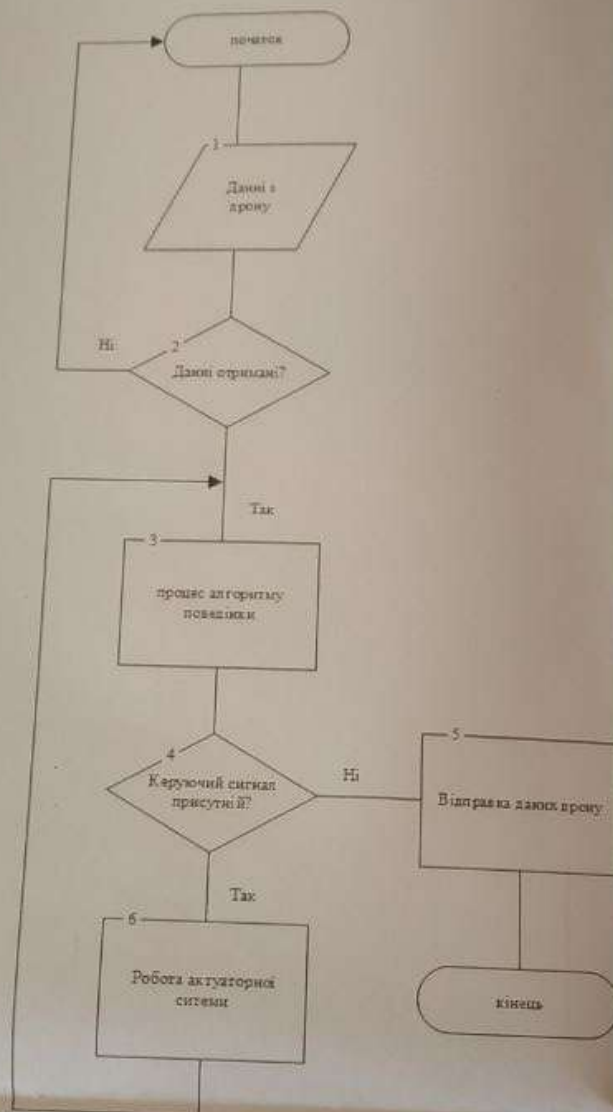
Зам.	Лист	№ докум.	Підп.	Дата
Розроб.		Тимошук О.С.		
Перев.		Ткаченко М.Ю.		
Т.контр.				
Н.контр.		Щербак Н.В.		
Затв.		Кудрявцев Р.К.		

13.02070849.00014 A1

Загальний алгоритм функціонування дрону

Лист	Маса	Масштаб
Лист 2		Листів 1

НУ «Запорізька політехніка»
КНТ-511М



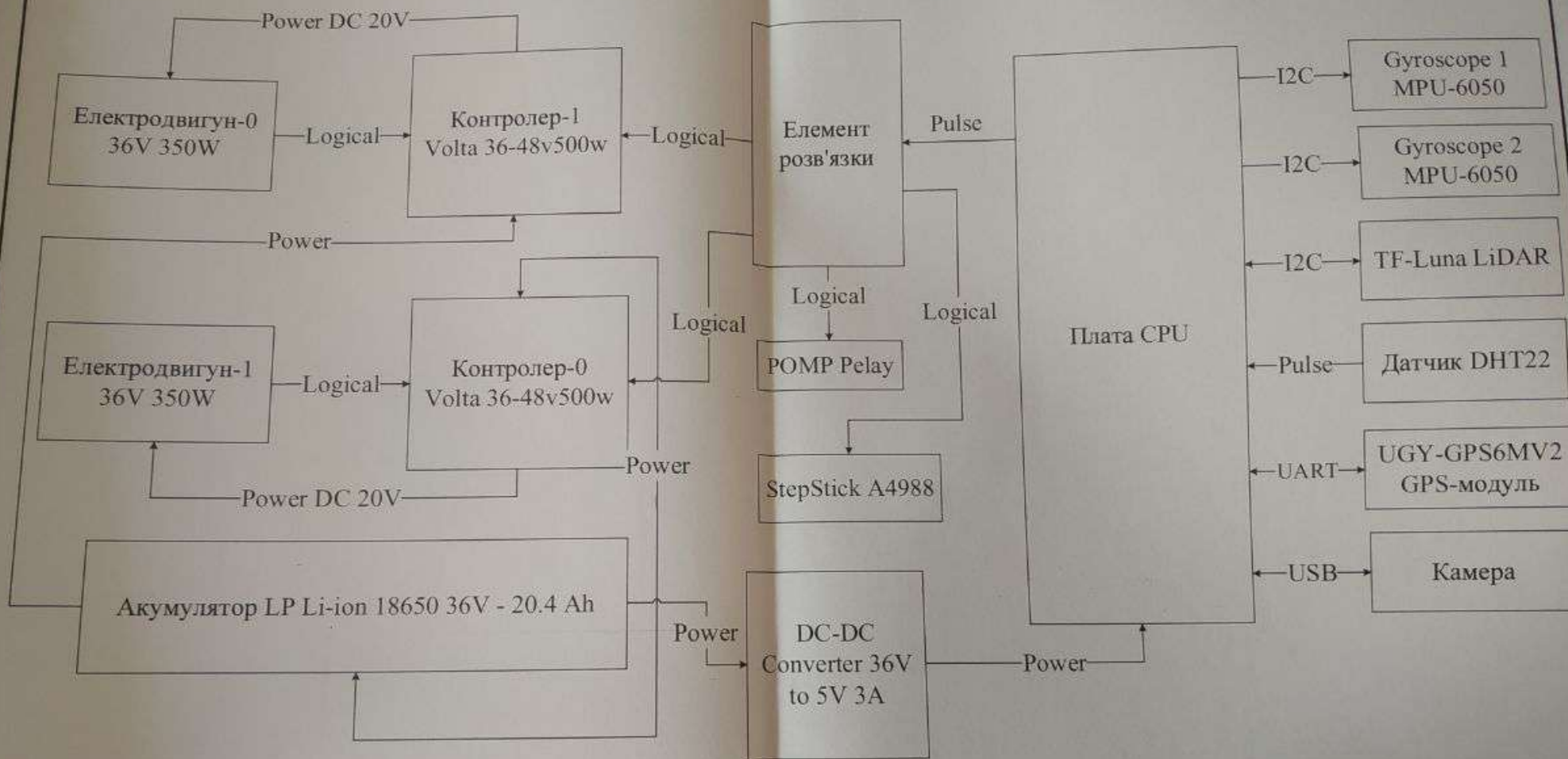
Зам.	Лист	№ докум.	Підп.	Дата	
Розроб.		Титов О.С.			
Перев.		Ткаченко М.Ю.			
Т.контр.					
Н.контр.		Цербах Н.В.			
Затв.		Куберикова Р.К.			

13.02070849.00014 A1

Загальний алгоритм
функціонування док-станції

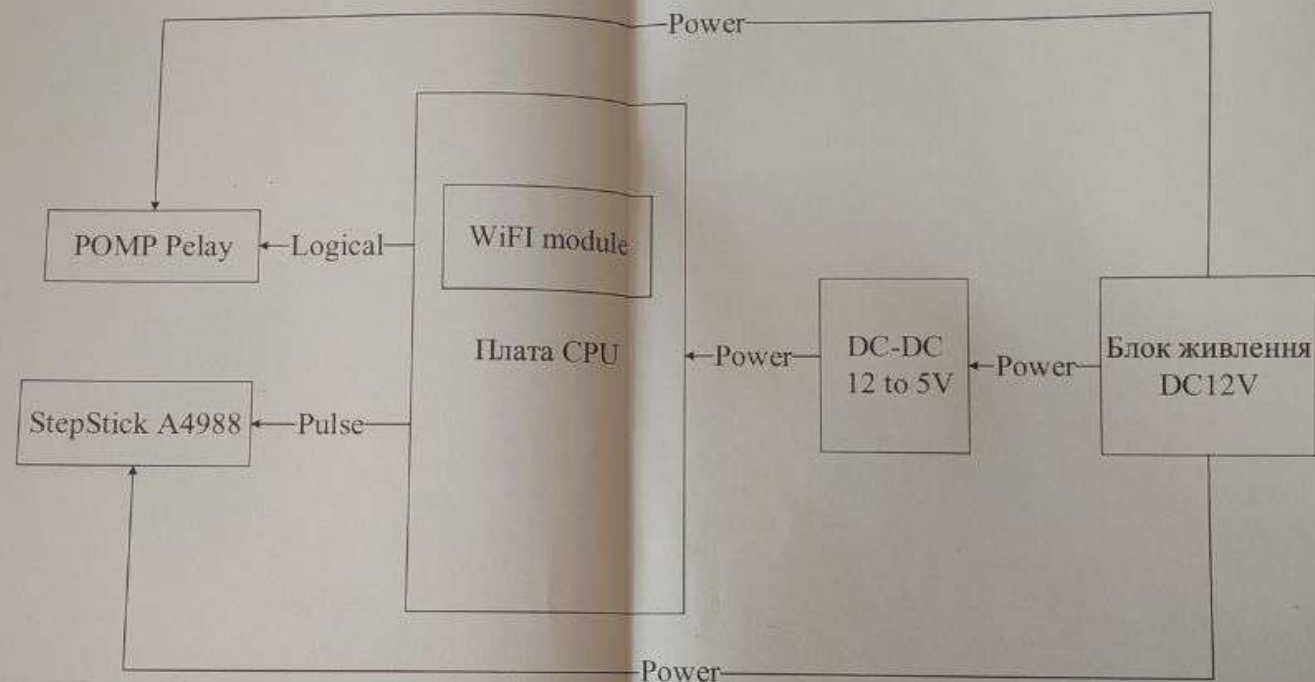
Лист	Маса	Масшт.
Лист 3	Листів 1	
НУ «Запорізька політехніка» КНТ-511М		

Структурна схема дрону



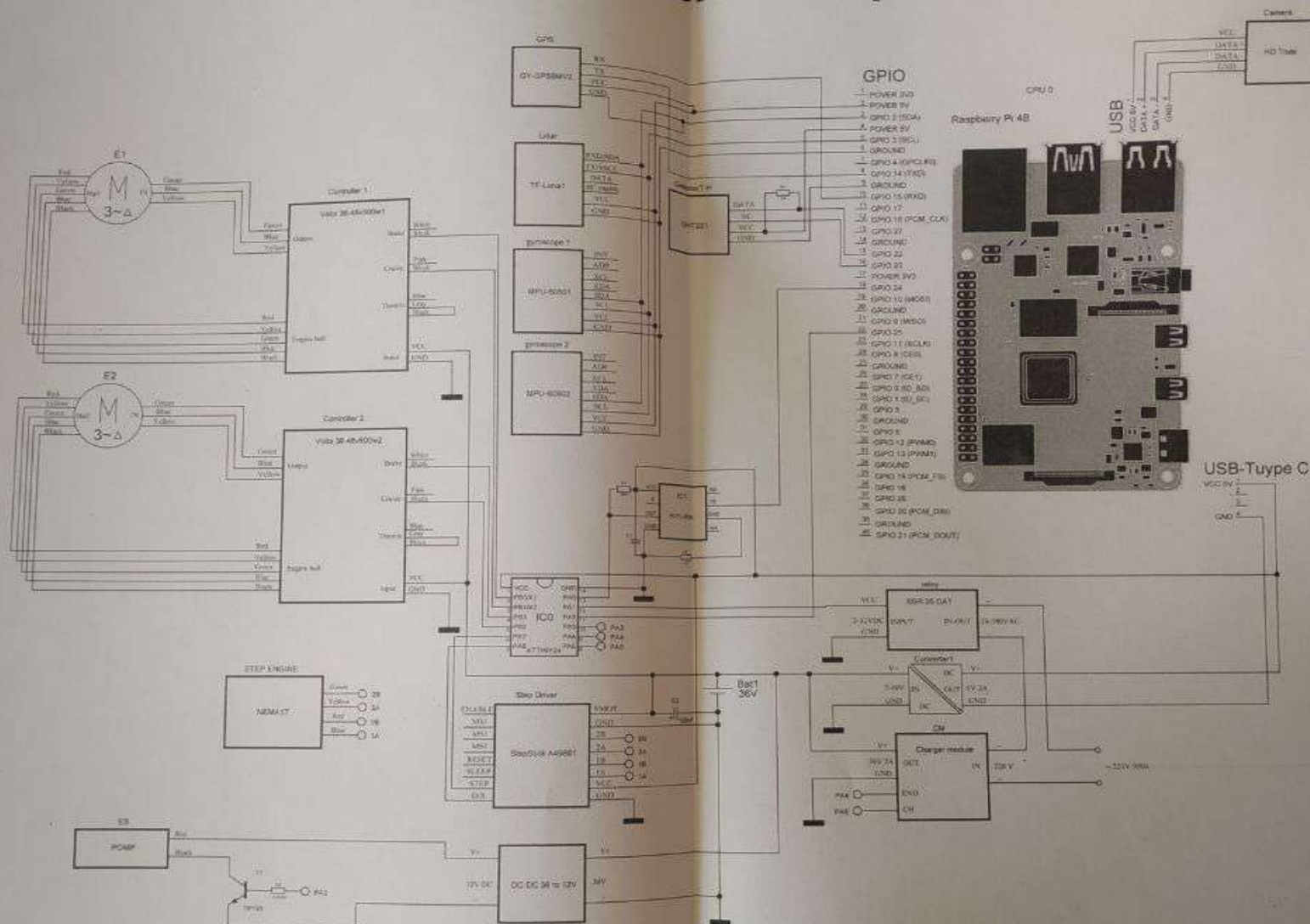
						13.02070849.00014 ПЛ2			
							Лист	Маса	Масштаб
Зам.	Лист	№ докум.	Підп.	Дат.	Структурна схема дрону				
Розроб.		Титова О.С.							
Перев.		Ткаченко М.Ю.							
Т.контр.									
Н.контр.		Щербак Н.В.							
Затв.		Кудряшов Р.К.							
						НУ «Запорізька політехніка» КНТ-511М			

Структурна схема док-станції



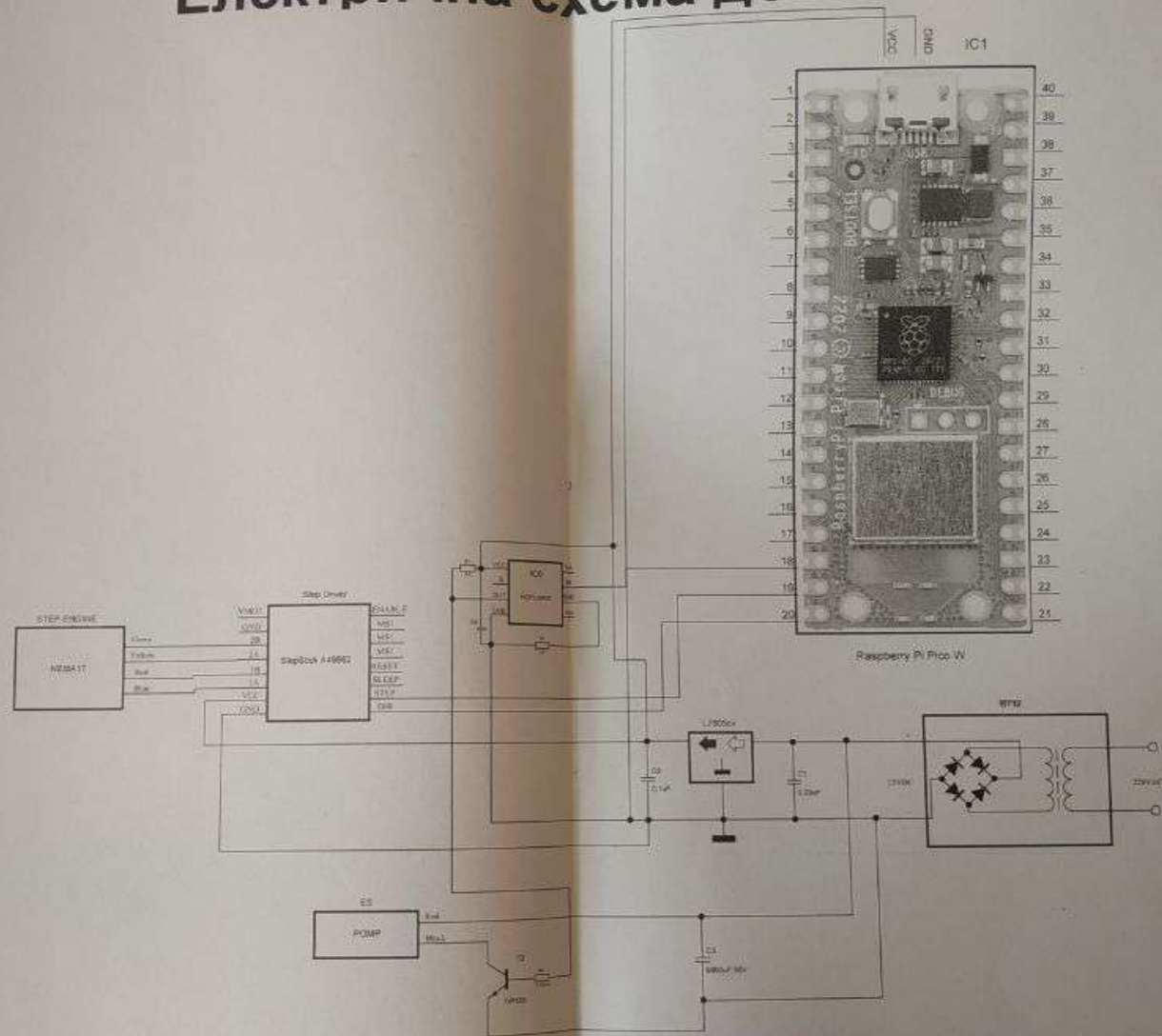
13.02070849.00014 ПЛЗ					
Зам.	Лист	№ дохрм.	Підп.	Дата	
Розроб.		Тимошук О.С.			
Перев.		Ткаченко М.Ю.			
Т.контр.					
Н.контр.		Щербак Н.В.			
Затв.		Куберник Р.К.			
Структурна схема док-станції					Лист
					Маса
					Лист 5
					Лист 1
					НУ «Запорізька політехніка» КНТ-511М

Електрична схема пристрою



						13.02070849.00014 ПЛ4					
Зам.	Лист.	№ докум.	Пом.	Дата					Лит.	Маса	Масшт.
Розроб.		Тимош О.С.	Шуф	2	Електрична схема дрону						
Перев.		Гаврилова М.Ю.	Шуф								
Т.контр.									Лист 6	Листа 1	
Н.контр.		Щербак Н.В.	Шуф						НУ «Запорізька політехніка» КНТ-511М		
Затв.		Куца Дмитро Р.К.	Шуф								

Електрична схема док-станції



					13.02070849.00014 ПЛ5		
					Електрична схема док-станції		
Зам.	Лист	№ докум.	Підп.	Дати	Лист	Маса	Масштаб
Розроб.		Титов О.С.					
Перев.		Ткаченко М.Ю.			Лист 1		Листів 1
Т. контр.					НУ «Запорізька політехніка» КНТ-511М		
Н. контр.		Щербак Н.В.					
Зам.		Кубишин О.К.					