

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
«ПОЛТАВСЬКА ПОЛІТЕХНІКА ІМЕНІ ЮРІЯ КОНДРАТЮКА»

А. С. Янко
Б. Р. Боряк
О. І. Лактіонов

МОДУЛЬНІ РОБОТИЗОВАНІ КОМПЛЕКСИ: ТЕХНІЧНІ ТА ПРОГРАМНІ РІШЕННЯ

Монографія



Львів – Торунь
Liha-Pres
2025

УДК [681.5+004.896]:355.424.4
М74

Автори:

Янко Аліна Сергіївна – кандидат технічних наук, доцент, доцент кафедри комп'ютерних та інформаційних технологій і систем, Національний університет «Полтавська політехніка імені Юрія Кондратюка», Полтава, Україна.

e-mail: al9_yanko@ukr.net, тел.: +380954238943, ORCID: 0000-0003-2876-9316, Scopus Author ID: 57094953000;

Боряк Богдан Радиславович – кандидат технічних наук, доцент кафедри автоматики, електроніки та телекомунікацій, Національний університет «Полтавська політехніка імені Юрія Кондратюка», Полтава, Україна.

e-mail: boriakbr@nupp.edu.ua, тел.: +380961901882, ORCID: 0000-0002-8114-7930, Scopus Author ID: 58983614200;

Лактіонов Олександр Ігорович – кандидат технічних наук, доцент кафедри автоматики, електроніки та телекомунікацій, Національний університет «Полтавська політехніка імені Юрія Кондратюка», Полтава, Україна.

e-mail: itm.olaktionov@nupp.edu.ua, тел.: +380664169586, ORCID: 0000-0002-5230-524X, Scopus Author ID: 57210360300

Рецензенти:

Сільвестров Антон Миколайович – доктор технічних наук, професор, професор кафедри теоретичної електротехніки, Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського», Київ, Україна;

e-mail: silvestrovanton@gmail.com, тел.: +380955547571, ORCID: <https://orcid.org/0000-0002-2511-5029>, Scopus Author ID: 6603359555;

Конох Ігор Сергійович – доктор технічних наук, доцент, професор кафедри автоматизації та інформаційних систем, Кременчуцький національний університет імені Михайла Остроградського, Кременчук, Україна;

e-mail: konokh.is.univer@gmail.com, тел.: +30686507066, ORCID: <https://orcid.org/0000-0001-5930-1957>, Scopus Author ID: 57194700932

*Рекомендовано до опублікування Вченою радою
Національного університету «Полтавська політехніка імені Юрія Кондратюка»
(протокол № 13 від 30.09.2025 р.)*

Модульні роботизовані комплекси: технічні та програмні рішення : монографія /
М74 А. С. Янко, Б. Р. Боряк, О. І. Лактіонов ; Національний університет «Полтавська політехніка імені Юрія Кондратюка». – Львів – Торунь : Liha-Pres, 2025. – 178 с.

ISBN 978-966-397-495-8

DOI 10.36059/978-966-397-495-8

Монографія належить до галузі робототехніки та мехатроніки, зосереджуючись на розробці та підвищенні захищеності автоматизованих наземних і повітряних роботизованих комплексів, що є особливо важливим у сучасних умовах. Розглядаються питання проєктування модульних архітектур, використання адитивних технологій, ефективних систем керування рухом та обробки відеоданих. Особливу увагу приділено забезпеченню стійкості до впливу радіоелектронної боротьби, зокрема через математичне моделювання надійності. Монографія містить аналіз сучасних підходів і практичні рекомендації для фахівців, що працюють над створенням надійних та захищених роботизованих систем.

УДК [681.5+004.896]:355.424.4

© А. С. Янко, Б. Р. Боряк, О. І. Лактіонов, 2025

© Національний університет

ISBN 978-966-397-495-8

«Полтавська політехніка імені Юрія Кондратюка», 2025

ЗМІСТ

ПЕРЕДМОВА.....	5
ВСТУП.....	7
РОЗДІЛ 1	
СУЧАСНІ ПІДХОДИ ДО РОЗРОБКИ МОДУЛЬНИХ РОБОТОТЕХНІЧНИХ ПЛАТФОРМ.....	10
1.1. Методологія дослідження.....	10
1.2. Апаратно-програмне забезпечення системи.....	16
1.2.1. Загальний опис робототехнічної платформи.....	16
1.2.2. Керування кроковим двигуном на базі (Arduino/Raspberry Pi).....	19
1.2.3. Підключення ER8 ELRS до Raspberry Pi 5.....	26
1.2.4. Реалізація UART/CRSF-комунікації з пультом RM TX16S.....	28
1.2.5. Автоматичний запуск одноплатника.....	33
1.2.6. Налаштування бібліотек для дослідження комп'ютерного зору на Raspberry Pi.....	34
1.2.7. Практична реалізація інструментів для аналізу відеопотоку.....	40
Висновок до розділу 1.....	49
РОЗДІЛ 2	
СИСТЕМИ КЕРУВАННЯ ТА БУДОВА НАЗЕМНИХ РОБОТИЗОВАНИХ КОМПЛЕКСІВ.....	51
2.1. Архітектури систем керування наземними безпілотними апаратами.....	52
2.2. Система зв'язку оператора та НРК.....	55
2.3. Система керування рухом роботизованої платформи.....	84
2.3.1. Вибір двигунів для наземної роботизованої платформи.....	86
2.3.2. Вибір контролерів двигунів для наземної роботизованої платформи.....	88
2.3.3. Вибір основного контролеру для наземної роботизованої платформи.....	96
2.3.4. Вибір приймачів радіосигналу для наземної роботизованої платформи.....	101

2.4. Розробка прототипу системи керування рухом наземної роботизованої платформи.....	104
2.5. Модуль прийняття рішень.....	115
2.6. Обґрунтування вибору обладнання.....	120
2.7. Розроблення прототипу НРК.....	141
Висновок до розділу 2.....	147
РОЗДІЛ 3	
ПІДВИЩЕННЯ ЗАХИЩЕНОСТІ АВТОМАТИЗОВАНИХ	
НАЗЕМНИХ РОБОТОТЕХНІЧНИХ ПЛАТФОРМ	
В УМОВАХ РАДІОЕЛЕКТРОННОЇ БОРОТЬБИ.....	149
3.1. Аналіз невирішеності питання захищеності наземних робототехнічних платформ.....	150
3.2. Опис методики проведення дослідження.....	154
3.3. Математична модель надійності СОІУ у СЗК та кількісна оцінка показників ймовірності безвідмовної роботи.....	156
Висновок до розділу 3.....	162
ВИСНОВКИ.....	164
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	169

ПЕРЕДМОВА

Пропонована монографія присвячена актуальним питанням розробки та підвищення захищеності автоматизованих наземних і повітряних роботизованих комплексів, що є критично важливим у контексті сучасних технологічних викликів та вимог воєнного часу. Робота охоплює широкий спектр тем, починаючи від проєктування модульних архітектур та використання передових адитивних технологій для швидкого прототипування, до розробки ефективних систем керування рухом, обробки відеоданих та забезпечення стійкості до впливу радіоелектронної боротьби.

Перший розділ закладає основи дослідження, детально аналізуючи сучасний стан розробки модульних робототехнічних платформ, особливо в сфері дронів технологій в Україні. Розглядаються переваги використання адитивних технологій для прискорення розробки апаратного забезпечення, систематизуються рівні проєктування робототехнічних систем та описується методологія дослідження, що включає розробку багатофункціонального допоміжного модуля, стратегії автоматизованого розпізнавання об'єктів, методи керування електродвигунами та дослідження різних підходів до дистанційного керування. Особливу увагу приділено апаратно-програмному забезпеченню розробленої платформи та процедурам налаштування ключових компонентів.

Другий розділ монографії фокусується на розробці прототипу системи керування наземною роботизованою платформою. Проводиться детальний аналіз вибору апаратних компонентів, розглядаються особливості розробки програмного забезпечення для керування рухом та прийняття рішень на базі одноплатного комп'ютера. Значну увагу приділено дослідженню алгоритмів відстеження об'єктів для забезпечення автономної роботи. У розділі також представлено розроблений прототип триколісної платформи та описано етапи його тестування.

Третій розділ присвячений критично важливій проблемі підвищення захищеності автоматизованих наземних робототехнічних платформ в умовах радіоелектронної боротьби. Проаналізовано існуючі підходи до створення стійких систем обробки інформації та управління, виявлено їхні обмеження. У розділі запропоновано математичну модель надійності COIU, що функціонує в системі залишкових класів,

з урахуванням впливу перешкод. Проведено порівняльний аналіз надійності різних архітектур та сформовано практичні рекомендації щодо впровадження розроблених рішень для підвищення відмовостійкості наземних роботизованих комплексів в умовах радіоелектронної боротьби.

Монографія буде корисною для науковців, інженерів, розробників та всіх, хто цікавиться сучасними тенденціями розвитку робототехнічних систем, особливо в контексті їхнього застосування в умовах підвищених вимог до надійності та захищеності.

ВСТУП

Актуальність питань, розглянутих у даній монографії, зумовлена кількома ключовими факторами.

По-перше, динамічний розвиток технологій безпілотних систем, зокрема дронів, в Україні та світі є беззаперечним трендом. Збройні конфлікти останніх років яскраво демонструють вирішальну роль роботизованих платформ у розвідці, спостереженні, логістиці, розмінюванні та безпосередньому вогневому ураженні. Ефективність та гнучкість застосування роботизованих комплексів безпосередньо впливають на хід бойових дій та безпеку особового складу. У цьому контексті розробка модульних, легко адаптованих та швидко виготовлюваних роботизованих платформ стає стратегічним завданням для підвищення обороноздатності країни.

По-друге, використання адитивних технологій (3D-друку) революціонує процес розробки та виробництва апаратного забезпечення для робототехнічних платформ різного призначення. Швидке прототипування корпусних елементів, виготовлення спеціалізованих кріплень та інших компонентів дозволяє значно скоротити часові та фінансові витрати на етапах дослідження та розробки. Оперативне створення прототипів та їхня швидка модифікація відповідно до отриманих результатів випробувань є критично важливими для прискорення впровадження інноваційних рішень у сфері робототехніки.

По-третє, ефективне керування роботизованими платформами є складним багатограним завданням, що включає обробку великих обсягів сенсорних даних, прийняття рішень у реальному часі та забезпечення надійного зв'язку з оператором або іншими елементами системи. Розробка багатифункціональних модулів керування, здатних обробляти відеопотоки з множини камер, здійснювати автоматизоване розпізнавання об'єктів та керувати рухом виконавчих механізмів, є ключовим аспектом підвищення автономності та ефективності робототехнічних платформ. Особливої уваги потребує дослідження методів дистанційного керування, що забезпечують надійний зв'язок на значних відстанях та в умовах складного електромагнітного середовища.

По-четверте, в умовах сучасного військового протистояння забезпечення захищеності автоматизованих наземних роботизованих платформ в умовах радіоелектронної боротьби (РЕБ) набуває критичного

значення. Здатність противника впливати на системи керування та зв'язку роботизованих комплексів шляхом створення електромагнітних перешкод може призвести до втрати контролю над платформою та зриву виконання бойових завдань. Тому розробка стійких до РЕБ систем керування, що використовують альтернативні методи обробки інформації та резервування, є нагальною потребою.

Перспективи розвитку тематики

Дослідження, представлені в даній монографії, відкривають широкі перспективи для подальшого розвитку тематики автоматизованих роботизованих платформ. Серед найбільш перспективних напрямів можна виділити:

Інтеграція елементів штучного інтелекту (ШІ) для підвищення рівня автономності та прийняття рішень роботизованих комплексів у складних та динамічних умовах. Це включає розробку просунутих алгоритмів обробки сенсорних даних, машинного навчання для розпізнавання образів та прийняття тактичних рішень.

Розробка розподілених та мережецентричних систем керування, що дозволяють координувати роботу групи роботизованих платформ для виконання складних завдань. Це вимагає створення надійних та захищених каналів зв'язку, а також розробки ефективних протоколів обміну даними та алгоритмів колективного прийняття рішень.

Подальше вдосконалення адитивних технологій для розширення спектру матеріалів, що використовуються у виробництві роботизованих комплексів, підвищення міцності та функціональності друкованих компонентів, а також оптимізації процесів масового виробництва.

Дослідження та впровадження нових методів підвищення стійкості до РЕБ, включаючи використання альтернативних каналів зв'язку (оптоволоконні, лазерні), розробку алгоритмів адаптивного керування частотою та потужністю передачі, а також застосування методів обробки сигналів, стійких до навмисних перешкод.

Розробка уніфікованих модульних архітектур та стандартів інтерфейсів для спрощення інтеграції різних функціональних модулів та забезпечення сумісності між платформами різних типів. Це дозволить створювати гнучкі та легко масштабовані роботизовані системи, що відповідають різноманітним вимогам.

Дослідження питань енергоефективності та автономного живлення роботизованих комплексів для збільшення часу їхньої безперервної

роботи та розширення радіуса дії. Це включає розробку нових типів акумуляторів, систем керування енергоспоживанням та інтеграцію альтернативних джерел енергії.

Представлені в монографії результати є важливим кроком на шляху створення ефективних, надійних та захищених автоматизованих роботизованих платформ, здатних виконувати широкий спектр завдань в умовах сучасних технологічних викликів. Подальші дослідження в цій галузі матимуть значний вплив на розвиток військових технологій, промислової автоматизації та багатьох інших сфер діяльності, сприяючи підвищенню безпеки, ефективності та якості життя.

РОЗДІЛ 1

СУЧАСНІ ПІДХОДИ

ДО РОЗРОБКИ МОДУЛЬНИХ

РОБОТОТЕХНІЧНИХ ПЛАТФОРМ

Розвиток дронівих технологій в Україні є одним із найшвидших у Європі. Це потребує постійного моніторингу за програмним та апаратним забезпеченням. Адитивні технології прискорюють процес розробки електроніки за рахунок швидкого виготовлення корпусів та допоміжного обладнання для робототехнічної платформи.

Існуючі проекти [1] стосовно створення робототехнічних платформ та спеціалізованих модулів, які розширюють їх функціонування, розробляються на різних рівнях проєктування, зокрема Component-level design, Physical-Level Design / PCB Design тощо.

Робототехнічна платформа враховує кілька складових частин. У дослідженні розглядається лише частина модулів відеоспостереження із системою керування та аспекти налаштування обладнання до роботи, розробка корпусів обладнання та допоміжних кріплень для підвищення ефективності дослідження. З точки зору проєктування дослідження відноситься до System-Level Design і Subsystem-Level Design рівнів проєктування, де розглядається вибір існуючого обладнання та інтеграція у єдину схему. Після цього відбувається налагодження обладнання й підготовка до роботи – Subsystem-Level Design. При проєктуванні корпусів та допоміжного обладнання використовується рівень проєктування Physical-Level Design, де корпус проєктується у 3d-просторі та роздруковується на принтері.

Метою дослідження є підвищення ефективності розробки додаткових модулів робототехнічної платформи.

1.1. Методологія дослідження

Досліджується допоміжний багатofункціональний модуль робототехнічної платформи на базі одноплатного комп'ютера, котрий використовується як один із елементів аналізу відеопотоку за допомогою

кількох відеокамер. Для гнучкості огляду камери керовані електродвигунами. Крім того, додатково вивчалася аспекти автоматизації розпізнавання об'єктів на відеопотоці за допомогою різних регуляторів. Керування досліджуваного обладнання під час налаштування й відлагодження здійснювалося із клавіатури комп'ютера та за допомогою апаратури, що забезпечувала дистанційне керування. Для зручності експлуатації у польових умовах одноплатний комп'ютер під'єднувався до зовнішнього джерела живлення (акумуляторної батареї).

У процесі дослідження вивчалися аспекти керування кроковими електродвигунами різними способами. Умовно способи керування диференційовано на автоматичні та із залученням засобів віддаленого керування (провідний і радіо варіант). Перевага надавалася обладнанню, котре потребує мінімум зусиль щодо налаштування, зокрема як монтаж, так і програмування та відлагодження.

У обох випадках використовувалися контролер Arduino Uno та одноплатний комп'ютер Raspberry Pi 5, де невід'ємним елементом був також драйвер крокових двигунів TB6600 із способом підключення до контролера STEP/DIR/ENABLE. Загальна схема підключення подана на рисунку 1.1.

Перемикачі драйвера мали наступні стани SW3 OFF, всі інші ON. Для зручності керування до Arduino Uno підключався Analog Joystick [4]. При програмуванні контролера для керування кроковим електродвигуном важливим аспектом була реалізація функцій

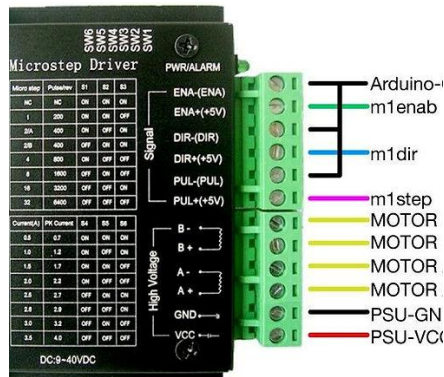


Рис. 1.1. Загальна схема підключення драйвера до контролера з диференційованими виходами [2, 3]

джойстика мовою програмування C++. Він мав у нейтральному положенні не надсилати вхідний сигнал керування на кроковий електродвигун, а при натисненні джойстика вперед/назад, вісь Y, вмикати електродвигун. Для цього було обрано величину сигналу в 50 абсолютних одиниць, що відповідали за нейтральне положення джойстика. Якщо рівень сигналу був більшим за $512+50$, то джойстик нахилено вперед, інакше, $-512-50$, джойстик нахилено назад.

Керування кроковим двигуном на базі Raspberry, за допомогою аналогового джойстика, потребує аналого-цифрового перетворювача. Тому, одноплатний комп'ютер отримував сигнали керування за допомогою пульта Radiomaster TX16S [5] із зовнішнього або внутрішнього передавача. Для цього до Raspberry Pi 5 було підключено приймач. Розглядалося кілька варіантів приймачів, зокрема ER8 [6] на базі протоколу CRSF.

Передумовою дослідження була попередня інсталяція операційної системи на одноплатний комп'ютер. У якості мови програмування використано python із додатковим пакетом бібліотек, котрий записувався у віртуальне середовище одноплатного комп'ютера, – `muenv`. Для керування кроковими двигунами використано бібліотеку `gpiozero`.

Передача сигналу від передавача до приймача вивчалася за допомогою протоколів Universal Asynchronous Receiver-Transmitter [7], далі UART та Crossfire Serial Protocol [8, 9], далі CRSF на швидкості 115200 bps. Особливістю протоколу CRSF є існування пакетів, де, наприклад, інформація про стан кнопок пульта, передається Control Packets (RC_CHANNELS_PACKED). При програмуванні обробки подій кнопок пульта враховувалося кодування, зокрема 16-бітні значення, де значення сигналу містилося у діапазоні $[0, 65535]$ та 11-бітні, – значення сигналу у діапазоні $[0, 2047]$.

На перших етапах дослідження обома протоколами зчитувалися і оброблялися дані із серійного порту вручну без використання пакетів протоколу. Важливо відмітити, кожного разу при вмиканні одноплатного комп'ютера необхідно активувати дозвіл для роботи з портами за допомогою команд `ls -l /dev/ttyS0` та `sudo chmod 666 /dev/ttyS0`.

Розглянуті принципи дослідження використовувалися для вивчення ком'ютерного зору, котрий реалізовувався мовою програмування python з використанням бібліотеки `cv2` [10]. Керування двигунами потребувало використання бібліотеки `gpiozero` [11]. Для графічного

представлення досліджуваних результатів використано бібліотеку matplotlib [12]. Серед використовуваних трекерів вивчалися Discriminative Correlation Filter with Channel and Spatial Reliability, Kernelized Correlation Filters, Minimum Output Sum of Squared Error, Multiple Instance Learning, Tracking, Learning, and Detection.

У якості середовищ розробки використовувалися Visual Studio Code, Google Colab, де відлагоджувався текст програми. З метою поглибленого тестування розроблених програмних продуктів використовувався інструментарій теорії автоматів. Остаточна верифікація розроблених рішень тестувалася на одноплатному комп'ютері.

Додатково для зручності проведення досліджень здійснювалося виготовлення корпусів та кріпильних елементів обладнання. Існуючі моделі корпусів [13], що пропонуються у вільному доступі, не адаптовані для використання на робототехнічній платформі (див. рис. 1.2).

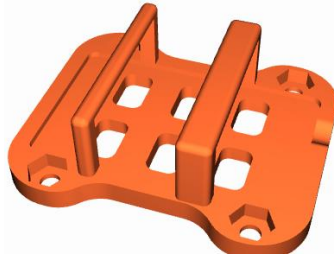


Рис. 1.2. Приклад корпусу відеопередавача Rush Cherry Max Solo [13]

Цільовим завданням розробки корпусу є досягнення універсальності та забезпечення багатофункціональності конструкції. Це передбачає необхідність інтеграції в дизайн спеціальних елементів кріплення, призначених для встановлення різноманітних допоміжних компонентів, зокрема кріплень для відеокамери та інших сенсорів або аксесуарів, що можуть знадобитися для розширення функціональних можливостей пристрою.

Проектування тривимірних моделей здійснювалося у сучасному середовищі автоматизованого проектування Solid Edge 2024, що дозволило точно візуалізувати та маніпулювати майбутньою конструкцією. Після завершення етапу моделювання створені цифрові моделі

зберігалися у файлах із розширеннями .stl або .stp, які є стандартними форматами для передачі даних на 3D-принтери.

Важливо відмітити, що для досягнення оптимальної точності відображення криволінійних елементів у системі проєктування Solid Edge 2024 знадобилося виконати додаткове налаштування параметру відображення. Зокрема, значення параметру View/Arc smoothness було встановлено на 10. Це коригування дозволило отримати більш гладкі та точні лінії при візуалізації тривимірних моделей корпусів, оскільки стандартне значення параметру Arc smoothness у програмі становить 3, що могло б призвести до менш деталізованого відображення дуг та кривих.

Для фізичного виготовлення спроектованих деталей використовувався сучасний 3D-принтер Bambu Lab A1, що характеризується високою якістю друку та продуктивністю. В якості друкувальної поверхні використовувалася текстурована PEI пластина (Textured PEI Plate), яка забезпечує надійну адгезію першого шару. Керування процесом друку здійснювалося за допомогою спеціалізованого програмного забезпечення Bambu Studio (див. рис. 1.3), яке надає широкі можливості для налаштування параметрів друку. У більшості випадків для надрукованих деталей встановлювалася щільність заповнення на рівні 15 %, що забезпечувало оптимальний баланс між міцністю та витратою пластику. При виборі типу підтримки для складних елементів конструкції враховувався аспект заощадження витратних матеріалів (пластику) та мінімізації часу, необхідного для виготовлення деталі. Інші параметри налаштування друку залишалися без змін, використовуючи значення за замовчуванням.

Крім того, для контролю температурного режиму роботи розробленого обладнання та виявлення можливих зон перегріву використовувався професійний тепловізор Testo, що дозволило отримати візуальне представлення розподілу тепла.

Механічна обробка окремих елементів обладнання, зокрема розсвердлювання отворів та інші операції, реалізовувалася на свердильному верстаті з урахуванням заданих допусків і посадок, відповідно до конструкторської документації [14].

Представлене дослідження має ітераційний характер, де кожен складовий елемент обладнання спочатку детально досліджувався як окрема віртуальна модель у програмному середовищі.

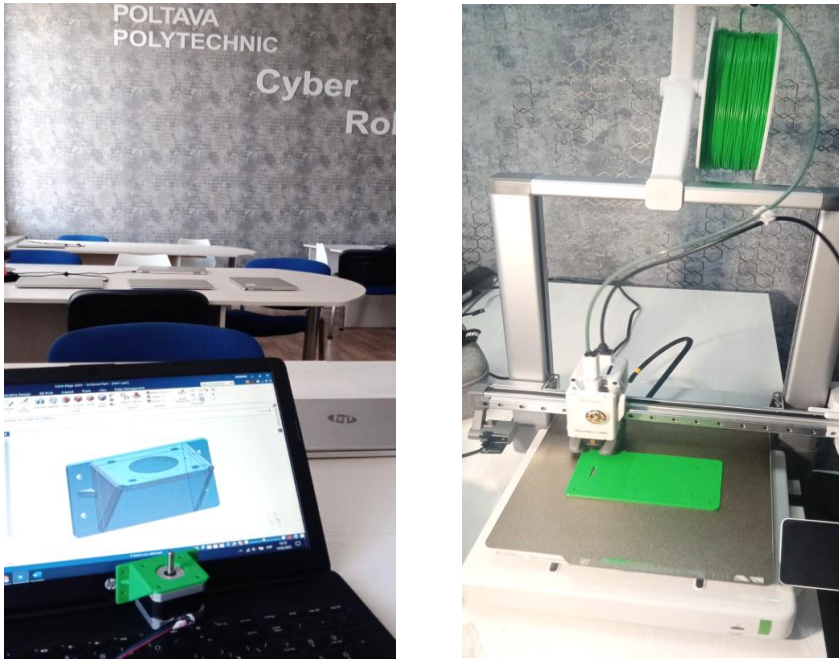


Рис. 1.3. Приклад створеного кронштейна для крокового електродвигуна у системі Solid Edge 2024 (ліворуч) та процес друку корпусу для одноплатного комп'ютера на принтері Bambu Lab A1 (праворуч)

На наступному етапі проводився монтаж електричної схеми, після чого створювався фізичний корпус для подальшого вивчення та тестування в лабораторних умовах, що дозволяло виявляти та усувати можливі недоліки на кожному етапі розробки.

У процесі дослідження враховувалися вимоги міжнародних стандартів, зокрема BS EN ISO 12100:2010 «Безпечність машин. Загальні принципи проєктування. Оцінювання ризиків та зменшення ризиків» [15].

1.2. Апаратно-програмне забезпечення системи

1.2.1. Загальний опис робототехнічної платформи

Робототехнічна платформа включає у свою структуру кілька модулів:

- основний силовий модуль робототехнічної платформи;
- єдиний багатофункціональний модуль на міні ПК Raspberry Pi Starter Kit 8GB;
- модуль дистанційного керування виконавчими органами робототехнічної платформи;
- модуль керування допоміжними електродвигунами робототехнічної платформи.

Модулі робототехнічної платформи виконують функції: керування дистанційним чином силовою частиною та допоміжними механізмами; керування відеопотоком для аналізу даних.

Основний силовий модуль робототехнічної платформи. Виконання комплектації комплектуючих для створення основного силового модуля робототехнічної платформи передбачає використання основної конструкції, до котрої кріпляться: плата керування, драйвер безколекторних двигунів у кількості 2 штуки, мотор-колесо у кількості 2 штуки, приймач радіосигналу TBS Crossfire Diversity Nano RX (Long Range), акумулятор для електросамоката 36 В у кількості 1 штука.

Єдиний багатофункціональний модуль на міні ПК Raspberry Pi Starter Kit 8GB. Комплектація комплектуючих для створення єдиного багатофункціонального модуля на міні ПК Raspberry Pi Starter Kit 8 GB. Модуль включає плату розширення Raspberry Pi для підключення електродвигунів, модуль зв'язку для Raspberry Pi та камера для Raspberry Pi (високої якості), камера Raspberry Pi Camera Module 3 й Камера з широким кутом огляду Waveshare IMX477-160 12.3 MP Camera. Підключення камери Raspberry Pi Camera Module 3 реалізовується кабелем зв'язку Raspberry Pi для камери.

Дистанційне керування міні ПК Raspberry Pi Starter Kit 8GB забезпечується модулем радіопередачі TBS Crossfire Diversity Nano RX (Long Range) та пультом управління для дрона із вбудованим передавачем радіосигналу керування. Живлення пульта управління для дрона реалізовано від акумуляторів літій-іонних 18650.

Підключення окремих модулів здійснюється залежно від пріоритетів проекту та конкретних завдань.

Сценарії використання:

Сценарій 1. Використання міні ПК Raspberry Pi Starter Kit 8GB алюмінієвий корпус, подвійний кулер самостійно без підключення додаткових модулів.

Сценарій 2. Використання міні ПК Raspberry Pi Starter Kit 8GB з платою розширення Raspberry Pi для керування електродвигунами.

Сценарій 3. Використання міні ПК Raspberry Pi Starter Kit 8GB з платою розширення Raspberry Pi для керування електродвигунами та обробкою відеопотоку з відеокамер. Всього доступно до використання камера Raspberry Pi Camera Module 3, камера для Raspberry Pi (високої якості) та камера з широким кутом огляду Waveshare IMX477-160 12.3 MP Camera.

Сценарій 4. Використання міні ПК Raspberry Pi Starter Kit 8GB з модулем зв'язку.

Сценарій 5. Використання міні ПК Raspberry Pi Starter Kit 8GB з модулем зв'язку та камерами для Raspberry Pi.

Сценарій 6. Використання TBS Crossfire Diversity Nano RX (Long Range) у взаємодії з пультом управління для дрона із передавачем радіосигналу керування забезпечує дистанційне керування обладнанням зафіксованим у сценарії 1-сценарій 5.

Ретельний вибір одного з представлених сценаріїв розгортання системи керування надає розробникам гнучку можливість адаптувати кінцевий продукт до специфічних операційних умов та індивідуальних вимог конкретного застосування. Більше того, такий модульний підхід не лише спрощує процес інтеграції окремих компонентів, але й забезпечує критично важливу можливість проведення поетапного тестування та точного налаштування кожного функціонального модуля. Ця детальна оптимізація на рівні окремих елементів дозволяє досягти вивіреного балансу між загальною продуктивністю всієї системи та її широким спектром функціональних можливостей, що є абсолютно необхідною умовою для забезпечення стабільної, надійної та передбачуваної роботи платформи в різноманітних експлуатаційних режимах.

Багатофункціональний модуль, що базується на компактному та енергоефективному міні-комп'ютері Raspberry Pi Starter Kit, оснащеному значним обсягом оперативної пам'яті у 8 ГБ, демонструє

виняткову універсальність у своєму застосуванні. Завдяки своїй обчислювальній потужності та гнучкості програмного забезпечення, цей модуль може бути розгорнутий як повністю автономний та самостійний елемент системи, відповідаючи за виконання чітко визначених завдань. Окрім того, він здатен безперешкодно інтегруватися в складніші архітектури для виконання ключових функцій керування електричними двигунами, забезпечуючи високоточне регулювання їхньої швидкості та напрямку обертання, або ж ефективно обробляти вхідні потоки відеоданих, здійснюючи аналіз зображень у реальному часі для виявлення об'єктів, розпізнавання образів чи побудови карт місцевості. Не менш вагомою є його здатність виступати в ролі невід'ємної складової більш комплексної робототехнічної платформи, значно розширюючи її загальний функціонал та інтелектуальні можливості.

Окремий модуль відповідає за забезпечення надійного та інтуїтивно зрозумілого дистанційного керування виконавчими органами робототехнічної платформи. Цей модуль включає в себе використання спеціалізованих пристроїв для передачі та прийому відеосигналу, таких як високочастотні передавачі та приймачі, що працюють, наприклад, у діапазоні 1,2 ГГц, для забезпечення стабільної та якісної передачі відеоінформації на значні відстані в режимі реального часу. Слід підкреслити, що даний модуль може функціонувати як абсолютно відокремлений та незалежний блок, надаючи оператору можливість автономно контролювати певні критично важливі функції платформи. У той же час, він може бути інтегрований як ключова складова частина загальної системи керування робототехнічної платформи, злагоджено взаємодіючи з іншими функціональними модулями для досягнення всеосяжного та скоординованого контролю над усіма аспектами її роботи. До складу цього модуля також входить спеціалізоване допоміжне обладнання, розроблене для забезпечення зручного та безпечного віддаленого розміщення оператора під час керування платформою, що значно підвищує рівень безпеки операцій та ергономічність робочого процесу.

Ще один важливий функціональний модуль відповідає за точне та ефективне керування допоміжними електричними двигунами робототехнічної платформи, забезпечуючи їхню синхронізовану роботу для виконання різноманітних другорядних, але часто необхідних завдань, таких як керування сенсорами, маніпуляторами або іншими допоміжними механізмами.

Окремий модуль, спеціально розроблений для керування рухомими механізмами виконавчих органів робототехнічної платформи, забезпечує їхню високоточну та енергоефективну роботу. Цей модуль може бути розгорнутий як повністю самостійний та незалежний блок, відповідаючи за керування конкретними виконавчими механізмами. Крім того, він може бути інтегрований в якості ключової складової частини більш складної та комплексної системи керування робототехнічної платформи, забезпечуючи злагоджену та скоординовану роботу всіх рухомих елементів. До складу цього модуля входять наступні основні компоненти: чотири високопродуктивні драйвери крокових двигунів моделі TB 6600, чотири надійні крокові двигуни стандарту NEMA 23, чотири прецизійні редуктори, призначені для крокових двигунів, та одна центральна плата керування, яка здійснює координацію та синхронізацію роботи всіх перелічених компонентів.

1.2.2. Керування кроковим двигуном на базі (Arduino/Raspberry Pi)

Базова електрична схема керування кроковим двигуном складається з наступних елементів: драйвер крокових двигунів TB6600, кроковий двигун NEMA 17, Arduino Uno, Analog Joystick.

Драйвер крокових двигунів TB6600 працює на напрузі 9–42 Вольт постійного струму та забезпечує вихідний струм 4 А та мікрокроки 1/1, 1/2, 1/4, 1/8, 1/16, 1/32 [2]. Відповідно до методології дослідження драйвер підключається до контролера з використанням ідей STEP/DIR/ENABLE. Сигнал напрямку DIR+ і сигнал кроку PUL+ підключаються до цифрових виводів контролера з котрих подається вхідний сигнал керування кроковими двигунами, (див. рис. 1.4).

Реалізована електрична схема здатна керувати кроковим електродвигуном за допомогою аналогового джойстика, котрий підключений до контролера дротом. Для поліпшення надійності керування, елементи схеми можуть бути дубльовані, наприклад драйвер і аналоговий джойстик. Таким чином реалізована сема може бути використана для керування виконавчими органами, зокрема поворотом щогли.

Варто відмітити, що механічна частина досліджуваного обладнання була модифікованою за рахунок монтування редуктора. Редуктор має кріпильні отвори для гвинтів M5, а кроковий двигун – кріпильні

отвори для гвинтів М4. Тому, процедура зажадала використання свердильного верстата для розсвердлювання кріпильних отворів двигунів. Після чого двигун був з'єднаний з редуктором гвинтами М5х12 із шайбами-гроверами.

Для зручності дослідження обладнання було розміщене у корпус (див. рис. 1.5).

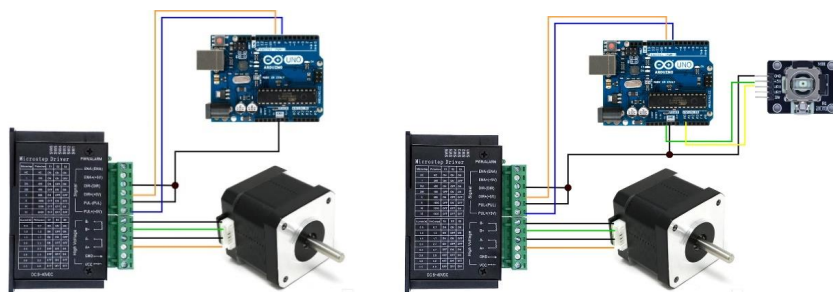


Рис. 1.4. Базова електрична схема підключення крокового електродвигуна на основі Arduino Uno

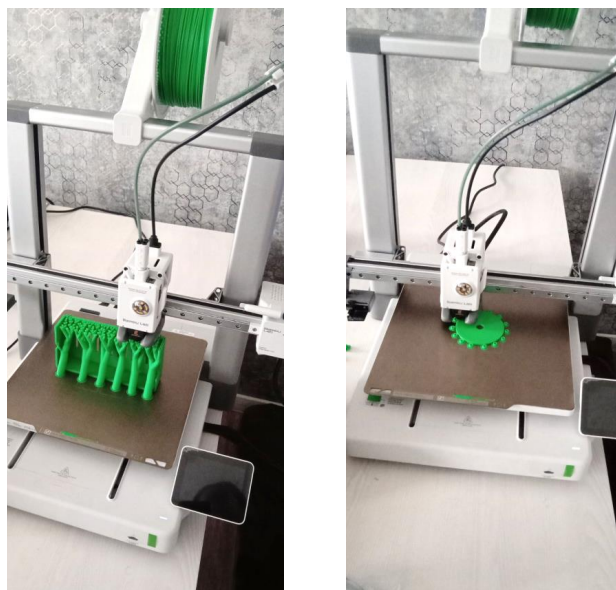


Рис. 1.5. Процес виготовлення корпусу та додаткових елементів кріплення досліджуваного обладнання

У процесі використання принтера з'ясувалося, що використання файлу з розширенням .stp є гнучкішим для створення корпусів. Це пояснюється існуванням функцій лінійного відхилення та лінійного кута, що дозволяє точніше налаштувати модель для друку.

Реалізована електрична схема може бути використана на робототехнічній платформі для керування механізмами або для керування зовнішніми тримачами антен дистанційним чином. Розглянемо використання одноплатного комп'ютера для керування електродвигунами робототехнічної платформи.

Виходимо з того, що однаплатний комп'ютер має інсталювану операційну систему за інструкцією [16, 17]. Використання одноплатного комп'ютера передбачає налаштування програмного забезпечення. Raspberry Pi використовувався у поєднанні із пристроями введення/виведення як самостійна одиниця. Для налаштувань використовувалася комунікація мережевого протоколу Secure Shell [18], котра потребувала визначення IP-адреси одноплатного комп'ютера.

Пошук IP-адреси здійснювався відомим способом, за допомогою браузера, де вводився пошуковий запит, – відома адреса маршрутизаторів, 192.168.1.1 або 192.168.0.1. Після цього у форму, яка відкрилася необхідно ввести логін та пароль мережі (див. рис. 1.6).

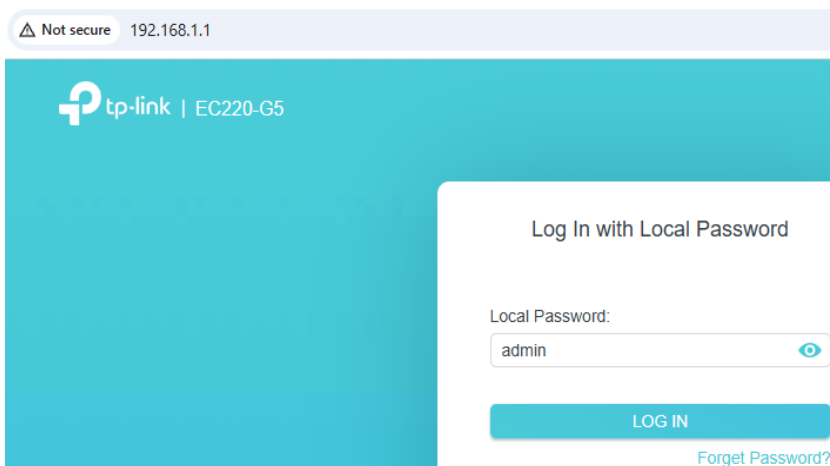


Рис. 1.6. Приклад активації форми адміністратора локальної мережі

Отримавши доступ до мережі проводили виявлення IP-адреси одноплатного комп'ютера (див. рис. 1.7).

Аналогічно здійснювався пошук у кількох мережах (див. рис. 1.8).

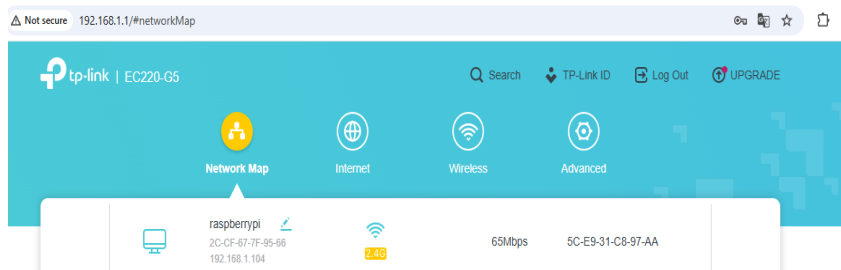


Рис. 1.7. Приклад пошуку IP-адреси одноплатного комп'ютера мережі № 1

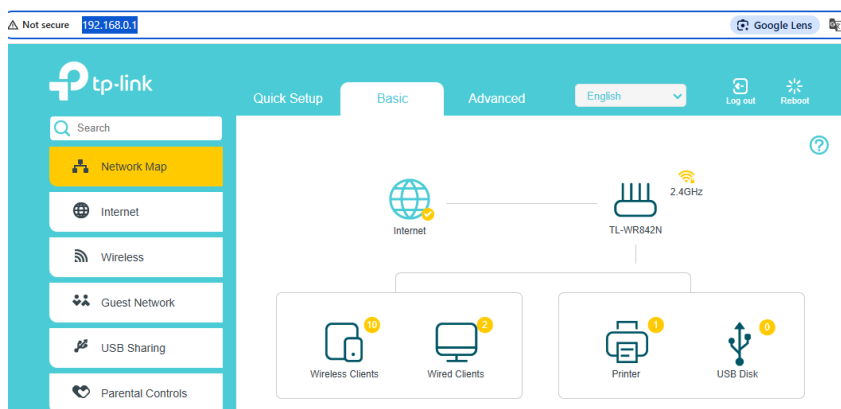


Рис. 1.8. Приклад пошуку IP-адреси одноплатного комп'ютера мережі № 2

Виявивши IP-адресу одноплатного комп'ютера маємо можливість отримати доступ до нього засобами мережевого протоколу Secure Shell. Для цього у Power Shell здійснювалася перевірка існування порту відомим чином (див. рис. 1.9).

Після цього отримували доступ до роботи з одноплатним комп'ютером (див. рис. 1.10).

```
PS C:\Users\lakti> Test-NetConnection 192.168.1.104 -Port 22

ComputerName      : 192.168.1.104
RemoteAddress     : 192.168.1.104
RemotePort        : 22
InterfaceAlias    : Wi-Fi
SourceAddress     : 192.168.1.102
TcpTestSucceeded  : True
```

**Рис. 1.9. Перевірка існування порту підключення
одноплатного комп'ютера**

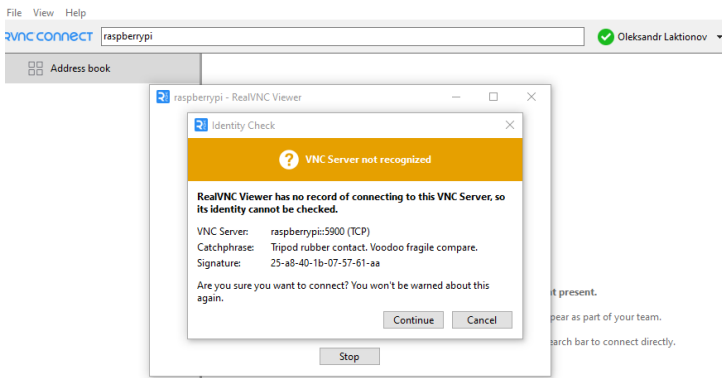
```
PS C:\Users\lakti> C:\Windows\System32\OpenSSH\ssh.exe pi@192.168.1.104
The authenticity of host '192.168.1.104 (192.168.1.104)' can't be established.
ECDSA key fingerprint is SHA256:rp2+0pkN6mFaMWqoC8lTo0f5qwDlJriXsrx2dlyfr4.
Are you sure you want to continue connecting (yes/no/[fingerprint])? y
Please type 'yes', 'no' or the fingerprint: yes
Warning: Permanently added '192.168.1.104' (ECDSA) to the list of known hosts.
pi@192.168.1.104's password:
Linux raspberrypi 6.6.51+rpt-rpi-2712 #1 SMP PREEMPT Debian 1:6.6.51-1+rpt3 (2024-10-08) 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 Mar 25 21:05:46 2025
pi@raspberrypi:~$
```

**Рис. 1.10. Приклад підключення до одноплатного комп'ютера
засобами Secure Shell**

Оскільки виконувалися дослідження із графічним інтерфейсом, то використовувався Virtual Network Computing, – віддалене керування. Приклад активації Realvnc (див. рис. 1.11 та рис. 1.12).



**Рис. 1.11. Форма введення назви апаратного забезпечення
або її IP-адреси**

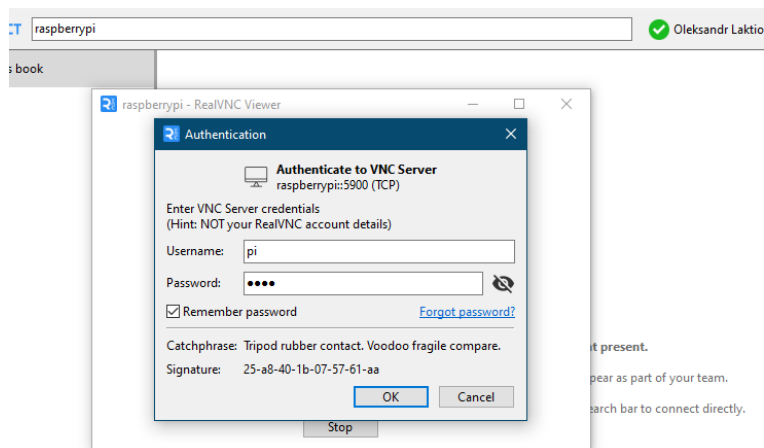


Рис. 1.12. Форма введення логіну та пароля від одноплатного комп'ютера

Після успішної активації служби RealVNC на одноплатному комп'ютері Raspberry Pi, наступним кроком є встановлення з'єднання з його робочим столом. Для цього на локальному комп'ютері запускається клієнтська програма RealVNC Viewer. У вікні програми необхідно ввести IP-адресу або ім'я хоста Raspberry Pi, після чого ініціюється процес підключення. У разі успішного встановлення зв'язку, на екрані локального комп'ютера відображається графічний інтерфейс робочого столу Raspberry Pi. Тепер користувач отримує повний візуальний доступ до файлової системи, встановлених програм та налаштувань операційної системи, ніби він безпосередньо працює за Raspberry Pi з підключеним монітором, клавіатурою та мишею. Таким чином, після встановлення віддаленого графічного з'єднання через RealVNC, одноплатний комп'ютер Raspberry Pi стає повністю готовим до використання для виконання різноманітних завдань, включаючи розробку програмного забезпечення, тестування, налагодження, запуск графічних додатків, конфігурацію системи та багато іншого, як це наочно демонструє рисунок 1.13.

Можливості одноплатного комп'ютера значно ширші за контролер, розглядалося керування автономним способом та дистанційним. Для цього були задіяні General-Purpose Input/Output, далі GPIO порти одноплатного комп'ютера (див. рис. 1.14).

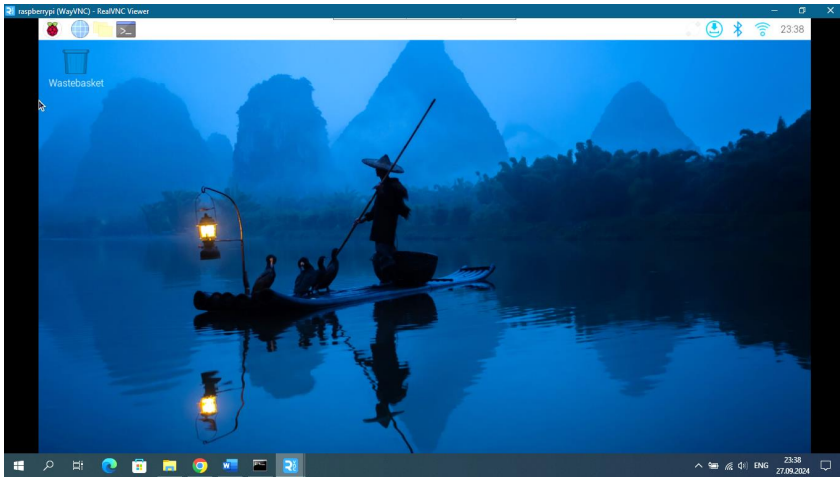


Рис. 1.13. Інтерфейс досліджуваного Raspberry Pi 5

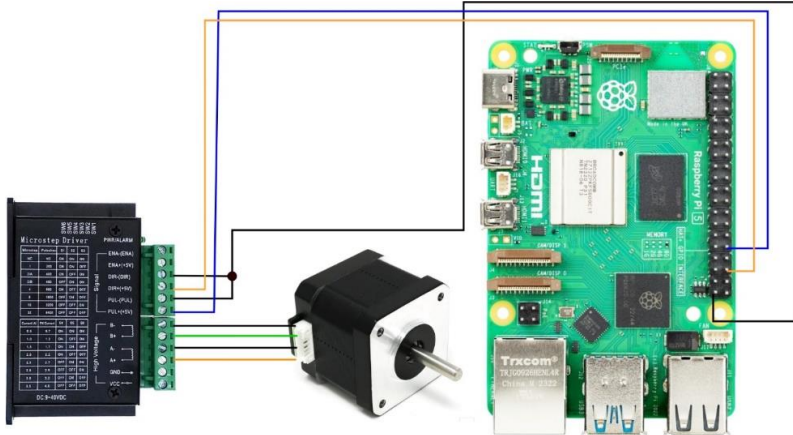


Рис. 1.14. Базова електрична схема підключення крокового електродвигуна на основі Raspberry Pi

При керуванні кроковим двигуном автоматично, де одноплатний комп'ютер надсилає сигнали керування на відповідний GPIO, труднощів не виникало (див. рис. 1.15).

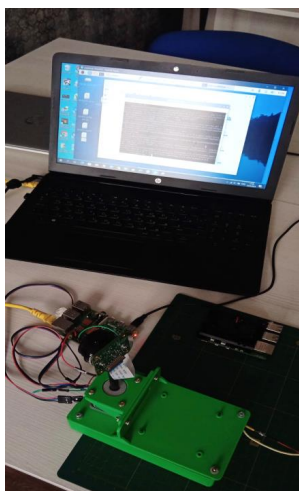


Рис. 1.15. Налаштування одноплатного комп'ютера для роботи із кроковим електродвигуном

Але відлагодження одноплатного комп'ютера Raspberry Pi 5 потребувало порівняльного аналізу із іншими одноплатними комп'ютерами, зокрема Raspberry Pi 4 model B.

1.2.3. Підключення ER8 ELRS до Raspberry Pi 5

Приймач er8, що використовує протокол ELRS (ExpressLRS), має ряд контактних виводів для забезпечення його функціональності та взаємодії з іншими компонентами системи керування. Вісім канальних виводів (ch1–ch8), кожен з трьома пінами, призначені для керування сервоприводами або іншими виконавчими механізмами за допомогою ШІМ-сигналів. Виводи “bat” (battery) з трьома пінами слугують для підключення живлення приймача, а також можуть використовуватися для моніторингу напруги акумулятора. Особливу увагу привертає порт з позначеннями –, +, rx, tx, до якого підведено три дроти: чорний (зазвичай «земля»), червоний (позитивне живлення) та білий (використовується для передачі даних), при цьому провідник “rx” (приймод даних) у даній конфігурації залишається непідключеним. Додатково присутні виводи “ext-v” (external voltage) з двома пінами (–, +), які можуть використовуватися для підключення зовнішнього

джерела живлення або для отримання інформації про зовнішню напругу. Процес інтеграції та налагодження цього приймача в систему керування вимагав виконання послідовних кроків, спрямованих на забезпечення коректного електричного з'єднання та програмної конфігурації для обміну даними з головним контролером. Підключення та налаштування роботи приймача здійснювалося у кілька кроків.

Крок 1. Монтаж електричної схеми. Підключення електричної схеми реалізовано таким чином. TX (білий) підключаємо RX (GPIO 15, пін 10) Raspberry Pi (див. рис. 1.16).

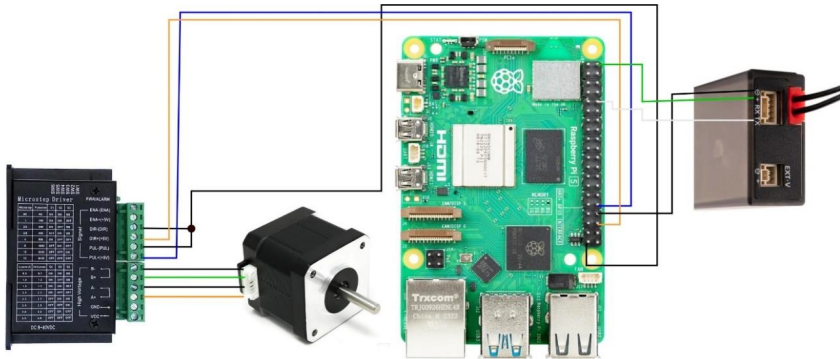
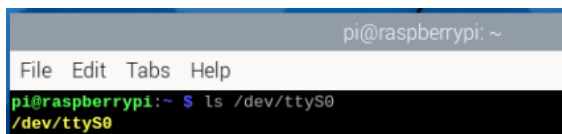


Рис. 1.16. Базова електрична схема підключення крокового електродвигуна на основі Raspberry Pi із приймачем

Крок 2. Наступним кроком є виявлення порту Raspberry Pi до якого підключено ER8. Для цього відкриваємо термінал одноплатного комп'ютера та вводимо команди `dmesg | grep tty` і `ls /dev/ttyS0`. Отриманий результат `/dev/ttyS0` є портом, до якого підключено приймач (див. рис. 1.17).

Таким чином приймач фізично підключено до одноплатного комп'ютера і назва порта підключення відома.

Крок 3. Здійснимо перевірку приймача на предмет отримання даних. Для перевірки підключення передавача до одноплатного комп'ютера створимо текст програми у файлі `serial1.py`, відправимо тестові дані `ser.write(b'Hello, ER8!\n')`. Якщо тестові дані отримано, то вивести наступне повідомлення `print("Received:", data.decode('utf-8').strip())` (див. рис. 1.18).



```
pi@raspberrypi: ~
File Edit Tabs Help
pi@raspberrypi:~ $ ls /dev/ttyS0
/dev/ttyS0
```

Рис. 1.17. Результати використання команди `ls /dev/ttyS0` для виявлення порту Raspberry Pi, до якого підключено приймач



```
(myenv) pi@raspberrypi:~ $ python /home/pi/myenv/serial1.py
Message sent: Hello, ER8!
```

Рис. 1.18. Результати перевірки приймача на предмет отримання заданих даних

Таким чином, скрипт відправив повідомлення “Hello, ER8!” через серійний порт. Тому з’єднання між Raspberry Pi і приймачем ER8 ELRS встановлено правильно, обладнання готове до наступного етапу дослідження. Розглянемо кроки реалізації uart-комунікації між пультом *rm tx16s* і Raspberry Pi 5 з підключеним приймачем *er8*.

1.2.4. Реалізація UART/CRSF-комунікації з пультом *RM TX16S*

На пульті *RM TX16S* меню *SYS/Hardware*, вкладка *Serial port* має кілька варіантів вибору, зокрема *AUX1* обрано *Telem In*. Для реалізації uart-комунікації між пультом *rm tx16s* і Raspberry Pi 5 з підключеним приймачем *er8* сформуємо формальну постановку завдання обробки даних із послідовного порту.

Дано множина байтів $D = \{d_1, d_2, \dots, d_n\}$, де канали передавача закодовані відомим форматом *little-endian* і визначаються виразом (1.1):

$$C_i = d_{2i} + d_{2i+1} \cdot 256, \quad (1.1)$$

де i належить множині $\{1, 2, \dots, 16\}$;

d_i – i -й байт у послідовному потоці даних.

Для реалізації обробки даних із послідовного порту виконано наступні дії.

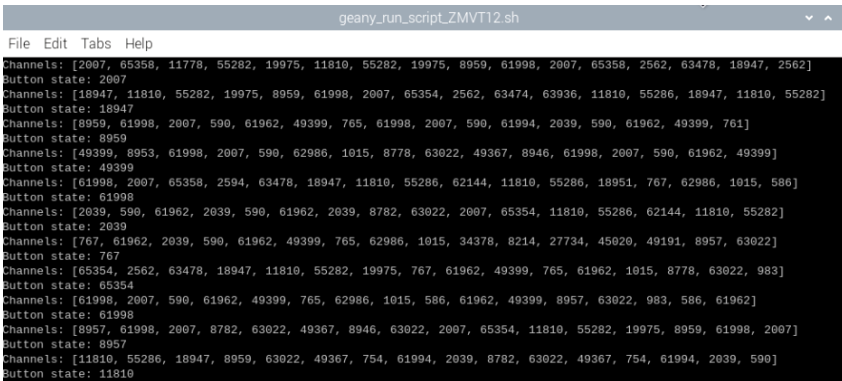
Крок 1. Налаштувати дозвіл на роботу з портами за допомогою команд: `source myenv/bin/activate; ls -l /dev/ttyS0; sudo chmod 666 /dev/ttyS0.`

Крок 2. Виконати підключення ER8 до Raspberry Pi за допомогою `ser = serial.Serial('/dev/ttyS0', 115200, timeout=1)`.

Крок 3. Здійснити зчитування даних з серійного порту.

Крок 4. Здійснити обробку даних, які мають діапазон від 0 до 65535, де пакет даних із 34 байтів.

Отриманий результат подано на рисунку 1.19.



```

File Edit Tabs Help
geany_run_script_ZMVT12.sh
Channels: [2007, 65358, 11770, 55282, 19975, 11810, 55282, 19975, 8959, 61998, 2007, 65358, 2562, 63478, 18947, 2562]
Button state: 2007
Channels: [18947, 11810, 55282, 19975, 8959, 61998, 2007, 65354, 2562, 63474, 63936, 11810, 55286, 18947, 11810, 55282]
Button state: 18947
Channels: [8959, 61998, 2007, 590, 61962, 49399, 765, 61998, 2007, 590, 61994, 2039, 590, 61962, 49399, 761]
Button state: 8959
Channels: [49399, 8953, 61998, 2007, 590, 62986, 1015, 8778, 63022, 49367, 8946, 61998, 2007, 590, 61962, 49399]
Button state: 49399
Channels: [61998, 2007, 65358, 2594, 63478, 18947, 11810, 55286, 62144, 11810, 55286, 18951, 767, 62986, 1015, 586]
Button state: 61998
Channels: [2039, 590, 61962, 2039, 590, 61962, 2039, 8782, 63022, 2007, 65354, 11810, 55286, 62144, 11810, 55282]
Button state: 2039
Channels: [767, 61962, 2039, 590, 61962, 49399, 765, 62986, 1015, 34378, 8214, 27734, 45020, 49191, 8957, 63022]
Button state: 767
Channels: [65354, 2562, 63478, 18947, 11810, 55282, 19975, 767, 61962, 49399, 765, 61962, 1015, 8778, 63022, 983]
Button state: 65354
Channels: [61998, 2007, 590, 61962, 49399, 765, 62986, 1015, 586, 61962, 49399, 8957, 63022, 983, 586, 61962]
Button state: 61998
Channels: [8957, 61998, 2007, 8782, 63022, 49367, 8946, 63022, 2007, 65354, 11810, 55282, 19975, 8959, 61998, 2007]
Button state: 8957
Channels: [11810, 55286, 18947, 8959, 63022, 49367, 754, 61994, 2039, 8782, 63022, 49367, 754, 61994, 2039, 590]
Button state: 11810
  
```

Рис. 1.19. Результати зчитування сирих даних за допомогою протоколу UART та їх обробка ручним способом

Як бачимо з вікна терміналу, через кожен період часу отримуємо значення кожного каналу, котрі змінюються динамічно. Таким чином маємо загальну основу програмної реалізації роботи з протоколом, але без використання пакетів, тому проведемо дослідження реалізації crsf-комунікації між пультом `tm tx16s` і Raspberry Pi 5. Розглянемо кроки реалізації crsf-комунікації між пультом `tm tx16s` і Raspberry Pi 5 з підключеним приймачем `er8`.

Дані, зокрема про стан кнопок (2-позиційний тумблер мінімальне значення 0, максимальне значення 65535; 3-позиційний тумблер: мінімальне значення 0, середнє значення 32767, максимальне значення 65535), передаються у вигляді пакетів. Пакет містить типи даних, для керування використовується Control Packets (RC_CHANNELS_PACKED), де стани органів керування пульта кодують як 16-бітні значення для кожного каналу.

Аналогічно як і для роботи з протоколом UART для реалізації обробки даних виконаємо наступні дії. Крок 1. Здійснити налаштування серійного порту `serial.Serial('/dev/ttyS0', 115200, timeout=1)`.

