

Розробка прототипу польотного контролера на базі мікроконтролера ESP32

<https://doi.org/10.31713/MCIT.2025.123>

Семенко Богдан-Роман

Національний університет водного господарства
та природокористування
м. Рівне, Україна
semenko_vg24@nuwm.edu.ua

Андрій Христюк

Національний університет водного господарства
та природокористування
м. Рівне, Україна
a.o.khrystyuk@nuwm.edu.ua

Ігор Мельник

Національний університет водного господарства
та природокористування
м. Рівне, Україна
i.i.melnyk@nuwm.edu.ua

Анотація — В роботі проведено аналіз ринку сучасних польотних контролерів, а також досліджено можливості використання мікроконтролера ESP32 для побудови власної системи управління безпілотним літальним апаратом. Розроблено принципову електронну схему, яка включає основні сенсори (GPS, барометр, магнітометр, акселерометр), виконавчі механізми (серводвигуни, електродвигун) та модулі зв'язку (Wi-Fi, радіоканал). Для програмного забезпечення використано FreeRTOS, що дозволило організувати багатозадачність: одночасну обробку даних датчиків, стабілізацію польоту, роботу веб-сервера та радіозв'язку з пультом

Ключові слова — БПЛА; мікроконтролер; керування; FreeRTOS; STM32; польотний контролер

I. АКТУАЛЬНІСТЬ І ПОСТАНОВКА ПРОБЛЕМИ

Сучасний світ переживає технологічну революцію, одним із ключових каталізаторів якої стали безпілотні літальні апарати (БПЛА). За останні десятиліття дрони еволюціонували від нішевих пристроїв для ентузіастів до незамінних інструментів у цивільних та військових галузях. Глобальний ринок БПЛА демонструє стрімке зростання, яке, за прогнозами аналітиків, продовжиться й у 2026 році, перетворюючи цілі сектори економіки та докорінно змінюючи підходи до ведення бізнесу, моніторингу та безпеки. Особливої актуальності розвиток та виробництво безпілотних систем набули в Україні, де вони стали критично важливим елементом як для оборони країни, так і для майбутньої відбудови та модернізації економіки.

II. МЕТА ТА ЗАВДАННЯ ДОСЛІДЖЕННЯ

Метою є розробка та тестування прототипу польотного контролера на базі мікроконтролера ESP32. Для досягнення мети поставлено такі

завдання: (1) провести аналіз ринку сучасних польотних контролерів для БПЛА; (2) визначити ключові технічні вимоги до польотного контролера; (3) спроектувати апаратну частину контролера на базі ESP32 та необхідних сенсорів; (4) розробити або адаптувати програмне забезпечення для стабілізації та управління польотом.

III. ВИРІШЕННЯ ПОСТАВЛЕНОЇ ЗАДАЧІ

Для успішної розробки конкурентоспроможного вітчизняного польотного контролера (ПК) було проведено аналіз поточного ринку. Дослідження охоплювало вивчення домінуючих технологічних платформ, ключових виробників, функціональних можливостей популярних моделей та їх цінової політики станом на 2025 рік. Цей аналіз дозволив чітко визначити незайняті ринкові ніші та сформулювати технічні вимоги до прототипу. Ринок польотних контролерів умовно поділяється на кілька ключових сегментів, що відрізняються апаратною базою, програмним забезпеченням та сферою застосування (контролери на базі мікроконтролерів STM32; контролери екосистеми Pixhawk; DJI та інші закриті (пропріетарні) системи). В умовах повномасштабної війни в Україні з'явилася низка вітчизняних виробників, які розробляють власні контролери, наприклад, VYRIY або контролери, представлені на платформі Brave1. Це свідчить про високий попит та намагання зменшити залежність від іноземних постачальників, хоча поки що вони не займають домінуючої частки ринку. Проведений аналіз ринку польотних контролерів для БПЛА дозволяє зробити кілька ключових висновків. Сучасний ринок пропонує широкий вибір технологічно довершених та потужних рішень, переважно на базі архітектури STM32 та стандарту Pixhawk. Ці контролери забезпечують високу надійність та величезний функціонал, що задовольняє потреби професійних та вимогливих хобійних застосувань.

Однак, зворотним боком цієї технологічної складності є висока вартість кінцевих пристроїв, що стає суттєвим обмежувальним фактором для масового виробництва, особливо в умовах обмеженого бюджету. Для значної кількості практичних завдань — таких як прості FPV-дрони, апарати для навчання пілотів, базового моніторингу чи виконання спеціалізованих функцій — функціонал топових контролерів є надлишковим, а їхня ціна — економічно невиправданою. Таким чином, стає очевидним існування чітко окресленої ринкової ніші для бюджетних, але достатньо функціональних польотних контролерів. Існує гострий попит на рішення, які б пропонували необхідний мінімум для стабільного та керованого польоту, але при цьому мали б значно нижчу собівартість та були б розроблені на доступній елементній базі. Саме цю нішу і покликана зайняти розробка вітчизняного польотного контролера на базі мікроконтролера ESP32, який може стати економічно обґрунтованою та конкурентоспроможною альтернативою існуючим на ринку дорогим рішенням.

Проведене дослідження підтвердило гіпотезу про те, що розробка польотного контролера на альтернативній, дешевшій платформі, такий як ESP32, є виправданою та має значний ринковий потенціал. Головною вимогою до апаратної реалізації польотного контролера ставилась наявність на платі ESP32 достатньої кількості вільних GPIO-пінів, які можуть бути сконфігуровані як 6 PWM-виходів, один UART (Tx/Rx) та один SPI-інтерфейс, на додачу до стандартних пінів I2C. В результаті розроблено схему реалізації поставлених задач (рис.1.).

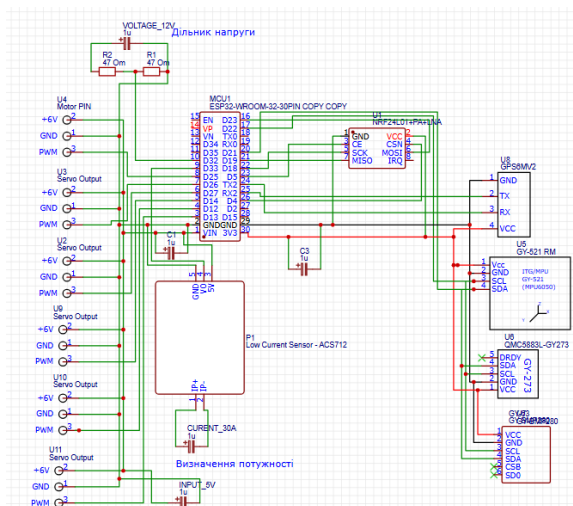


Рис. 1 Принципова електронна схема розробленого прототипу

Розроблений прототип містить: мікроконтролер MCU1 - ESP32-WROOM-32-30PIN, систему живлення та моніторингу; комплекс сенсорів для автономної навігації; канал зв'язку та керування; виконавчі механізми.

Також розроблено та реалізовано друковану плату пристрою (Рис.2.).

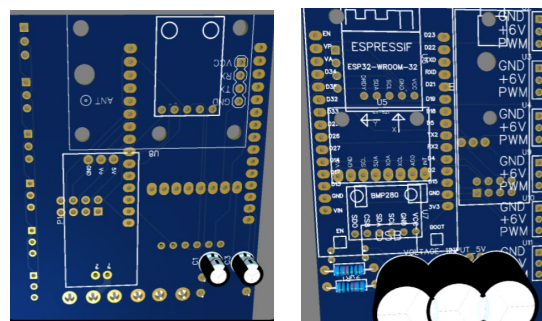


Рис. 2. Вигляд двосторонньої друкованої плати прототипу польотного контролера

Розроблена плата чітко розділена на функціональні зони: логічну/обчислювальну в центрі, силову/вихідну справа та навігаційну зверху. Це класичний підхід у проектуванні електроніки, який допомагає мінімізувати перешкоди. Вона демонструє розуміння принципів проектування електроніки для БПЛА, зокрема в частині компактного розміщення компонентів, фільтрації живлення та чіткого функціонального зонування.

Для програмного забезпечення використано FreeRTOS, що дозволило організувати багатозадачність: одночасну обробку даних датчиків, стабілізацію польоту, роботу веб-сервера та радіозв'язку з пультом. Це програмне забезпечення - складна, багатозадачна система, розроблена для керування дроном. Воно поєднує в собі кілька ключових функцій, що працюють паралельно завдяки операційній системі FreeRTOS:

- (1) збір даних з датчиків (система постійно зчитує дані з GPS, барометра, IMU та компаса, щоб знати своє місцезнаходження, орієнтацію та висоту);
- (2) зв'язок (Wi-Fi: створює власну точку доступу, що дозволяє будь-якому пристрою (смартфону, ноутбуку) підключитися, отримати доступ до веб-інтерфейсу і змінити параметри місії; радіозв'язок (nRF24): приймає команди з пульта (ручний режим) і надсилає назад телеметрію (поточний статус, висоту, координати) для відображення на пульті);
- (3) автономне керування (автопілот): програмне забезпечення використовує алгоритми ПІД-регулювання та геодезичні розрахунки, щоб автоматично підтримувати задану висоту, орієнтувати до цільових координат, виконувати складні послідовності дій, як-от зниження, скидання вантажу, набір висоти і повернення);
- (4) керування виконавчими механізмами: на основі поточного режиму та даних з датчиків, ПЗ керує сервоприводами, що контролюють кермо висоти, кермо напрямку, а також двигун і механізм скидання вантажу.

В програмному забезпеченні для проведення досліджень передбачено два розділи: статус системи та польотне завдання (рис.3.).

The screenshot shows a web interface for drone configuration. It is divided into two main sections: 'Статус системи' (System Status) and 'Польотне завдання' (Flight Task). The status section displays real-time data: Coordinates (0,0), Start altitude (0 m), Current altitude (0.94 m), Barometer (99406), MPU X/Y/Z (1.22, -1.38, 7.7), Compass (235°), Speed (0 km/h), and GPS OK (false). The flight task section contains input fields for Mode (0..3) set to 0, Target Latitude (0.000000), Target Longitude (0.000000), Target Altitude (m) set to 50, Distance to start descent (m) set to 50, and Drop altitude (m) set to 20. A blue 'Apply' button is at the bottom.

Рис. 3 Вигляд вікна налаштувань

Статус системи призначений для відображення телеметрії в реальному часі з датчиків літального апарата. Він надає пілоту або оператору критично важливі дані про поточний стан дрона:

- Координати: показує широту та довготу, отримані з GPS.
- Стартова висота: вказує висоту, на якій був запущений дрон. Це може бути використано як опорна точка для розрахунку відносної висоти.
- Поточна висота: показує поточну висоту літака над стартовою точкою, виміряну барометром.
- Барометр (Pa): відображає абсолютний атмосферний тиск. Зміни тиску використовуються для розрахунку висоти.
- MPU X/Y/Z: дані з інерційного вимірювального блоку (IMU), такого як MPU-6050. Ці значення (прискорення або кутова швидкість по трьох осях) використовуються для стабілізації та орієнтації літака.
- Компас: показує курс літака відносно магнітної півночі.
- Швидкість: визначається за даними GPS.
- GPS OK: логічний індикатор, який повідомляє, чи є дійсний сигнал GPS.

Польотне завдання є панеллю керування, що дозволяє оператору віддалено налаштовувати параметри автономної місії:

- Mode (0..3): Вибір режиму польоту.
 - o 0: idle (очікування)
 - o 1: nav (навігація до цільових координат)

- o 2: auto drop+return (автоматичне скидання вантажу і повернення)

- o 3: manual (ручне керування)

- Target Latitude/Longitude: Цільові GPS-координати, до яких повинен летіти дрон в автономному режимі.

- Target Altitude (m): Цільова висота для автономного польоту.

- Dist to start descent (m): Відстань від цілі, на якій дрон повинен розпочати зниження для скидання вантажу.

- Drop altitude (m): Висота, на якій має відбутися скидання вантажу.

- Apply: кнопка для застосування всіх встановлених параметрів і відправки їх на борт дрона.

Таким чином, ця система є повноцінною платформою для керування дроном, яка поєднує в собі ручний контроль, автономні можливості та зручний веб-інтерфейс для налаштування місій.

IV. ВИСНОВКИ

В роботі було проведено аналіз ринку сучасних польотних контролерів, а також досліджено можливості використання мікроконтролера ESP32 для побудови власної системи управління безпілотним літальним апаратом.

У ході роботи розроблено принципову електронну схему, яка включає основні сенсори, виконавчі механізми та модулі зв'язку. Для програмного забезпечення використано FreeRTOS, що дозволило організувати багатозадачність: одночасну обробку даних датчиків, стабілізацію польоту, роботу веб-сервера та радіозв'язку з пультом.

Також реалізовано веб-інтерфейс для моніторингу стану літака та введення параметрів польоту. Передбачено кілька режимів роботи: ручне керування, автоматичний політ за координатами та режим доставки вантажу.

Отримані результати підтверджують, що навіть на базі відносно доступної апаратної платформи можливо створити функціональний контролер з гнучким програмним забезпеченням.

ЛАТЕРАТУРА

- [1] Austin, R. Unmanned Aircraft Systems: UAVS Design, Development and Deployment. Wiley, 2010.
- [2] Mellinger, D., Michael, N., & Kumar, V. Trajectory Generation and Control for Precise Aggressive Maneuvers with Quadrotors. The International Journal of Robotics Research, 31(5), 2012.
- [3] Kilby T., Kilby B. Getting Started with Drones. USA : Make Community, LLC, 2015. 204 p.
- [4] Understanding ESCs for FPV Drones [Електронний ресурс]. – Режим доступу : <https://oscarliang.com/esc>.
- [5] Tang, L., Wang, H., Li, P., & Wang, Y. Real-time Trajectory Generation for Quadrotors using B-spline based Non-uniform Kinodynamic Search. arXiv:1904.12348, 2019.