

Грабець Н. Б.,
здобувач вищої освіти,
Волинський національний університет
імені Лесі Українки, м. Луцьк

УДК 629.05:004.896
DOI: <https://doi.org/10.64076/ihrc250728.04>



ДОСЛІДЖЕННЯ СУЧАСНИХ ЗАСОБІВ, МЕТОДІВ ТА ТЕХНОЛОГІЙ РОЗРОБКИ НАЗЕМНИХ РОБОТИЗОВАНИХ ЗАСОБІВ

У контексті швидкого розвитку мобільної робототехніки наземні автономні платформи (UGV — unmanned ground vehicles) відіграють важливу роль у різних галузях: від сільського господарства до військової сфери. Особливої актуальності набуває створення малогабаритних дронів-платформ для досліджень, логістичних задач, моніторингу та автономної навігації. Метою цієї роботи є розробка та виготовлення дослідного зразка наземної роботизованої платформи із використанням сучасних інструментів та технологій на всіх етапах розробки.

Конструктивні особливості наземного дрона

Роботизований засіб виконано у вигляді малогабаритної колісної платформи з двостороннім незалежним приводом. Кожна пара коліс з одного боку керується окремим каналом драйвера, що дозволяє реалізувати диференційне керування.

Основу шасі становить рама, виготовлена з алюмінієвих профілів, що забезпечують жорсткість конструкції та зручність монтажу. Допоміжні кріплення, редукторні кожухи та інші елементи спроектовано у середовищі SolidWorks та надруковано за допомогою FDM 3D-принтера з використанням PLA-пластику (рис. 1).



Рис. 1. Вигляд наземної платформи

Джерело: авторська розробка.

Апаратна частина системи керування

Для приводу використовуються постійного струму двигуни з редукторами, які забезпечують достатній крутний момент і плавність ходу. Контроль за моторами здійснюється через двоканальний драйвер двигунів (рис. 2), розроблений автором в системі для проектування друкованих плат EasyEDA. Один канал керує парою двигунів на лівому борту, інший — на правому.

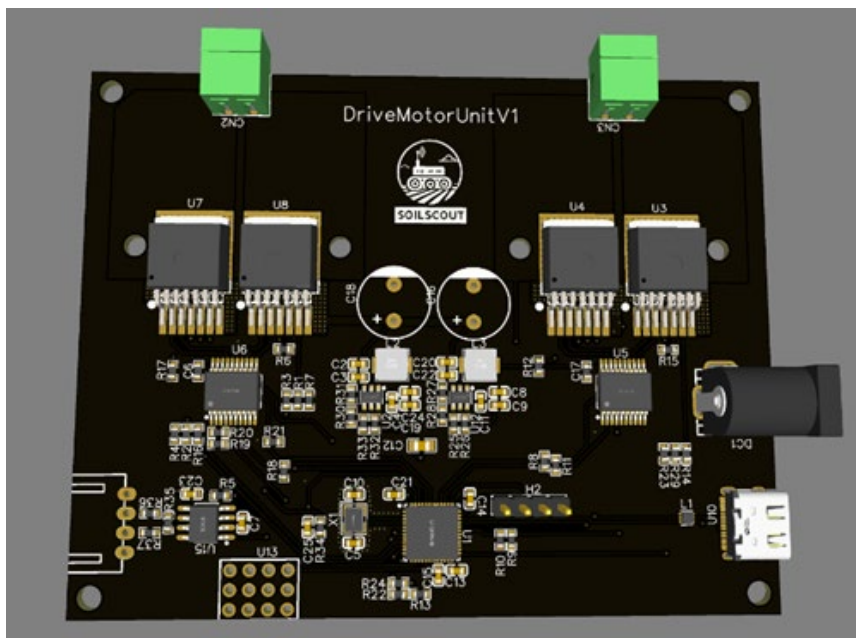


Рис. 2. Вигляд розробленої плати драйвера шасі

Джерело: авторська розробка.

Особливості драйвера:

- підтримка двох режимів керування: через PWM-сигнал (сумісний з радіоапаратурою або контролерами типу ArduPilot/Pixhawk);
- можливість керування через USB-інтерфейс (режим роботи як CDC-пристрій);
- реалізовано фільтрацію сигналів, захист по струму та індикацію стану.

Друкована плата драйвера розроблена в середовищі EasyEDA, виготовлена у Китаї через один з популярних онлайн-сервісів (JLCPCB).

Мікроконтролерна частина реалізована на основі STM32, а програмна логіка написана у STM32CubeIDE мовою C з використанням бібліотеки HAL.

Інтерфейси керування:

- Залежно від конфігурації, платформа може керуватись:
- через радіоканал, підключивши PWM-виходи від приймача до входів драйвера;
- через польотний контролер, який видає PWM-сигнали;
- через персональний комп'ютер, що передає команди по USB (через віртуальний COM-порт).

Команди містять інформацію про напрям і швидкість обертання кожного борту, формат повідомлень розроблений автором та має структуру, зручну для майбутньої інтеграції у ROS або інші фреймворки автономної навігації.

Виготовлення та тестування

Після завершення складання механічної частини та монтажу електроніки було реалізовано та протестовано три режими керування наземною платформою, що дозволяє гнучко адаптувати пристрій до різних сценаріїв використання — як у ручному, так і в автоматизованому або автономному режимах.

1. Режим ручного керування через радіоапаратуру FlySky

Для базового тестування платформи використовувалась популярна бюджетна радіоапаратура FlySky FS-i6, яка забезпечує передачу команд по PWM-каналах. Приймач FS-iA6B підключався безпосередньо до драйвера двигунів, де кожен PWM-вхід відповідає за керування відповідною парою коліс (лівий/правий борт). Цей режим дозволив:

- швидко перевірити працездатність всієї електромеханіки;
- протестувати динамічну поведінку шасі в польових умовах;
- оцінити чутливість та інерційність платформи під час маневрування.

2. Інтеграція з польотним контролером на прошивці ArduRover

На наступному етапі тестування платформа була інтегрована з польотним контролером Matek F405-Wing, що працював під керуванням прошивки ArduRover (один із компонентів open-source пакету ArduPilot).

Контролер надсилав керуючі сигнали у вигляді PWM-команд на входи драйвера двигунів, що забезпечувало:

- сумісність із навігаційними модулями (GPS, компас)
- можливість програмування автоматичних місій;
- підключення до наземної станції через Mission Planner;
- базову стабілізацію платформи.

Цей варіант підтвердив, що розроблений драйвер є сумісним з типовими автопілотами без необхідності вносити зміни у прошивку або електросхему.

3. Керування з одноплатного комп'ютера Raspberry Pi

Для потреб автономного або дистанційного керування також був реалізований варіант взаємодії з платформою через Raspberry Pi 4 Model B. У цьому випадку плата керування підключалась до Raspberry через USB-інтерфейс (CDC ACM) і сприймалась як віртуальний COM-порт. На стороні Raspberry Pi запускалась програма на C++, а в подальшому розроблений автором пакет на ROS2, яка:

- надсилала сформовані пакети керування у власному форматі (швидкість/напрямок для кожного борту);

- отримувала зворотній зв'язок від драйвера (наприклад, стан або аварії);
- дозволяла підключити телеметрію, Wi-Fi керування або SSH-доступ.

Цей режим особливо перспективний для майбутньої інтеграції з системами штучного інтелекту, комп'ютерного бачення та ROS2, а також для експериментів із хмарними системами управління.

Система показала стабільність роботи, адаптованість до низьковольтного живлення (2S/3S Li-Ion) та високу зручність обслуговування завдяки модульному підходу (рис. 3).

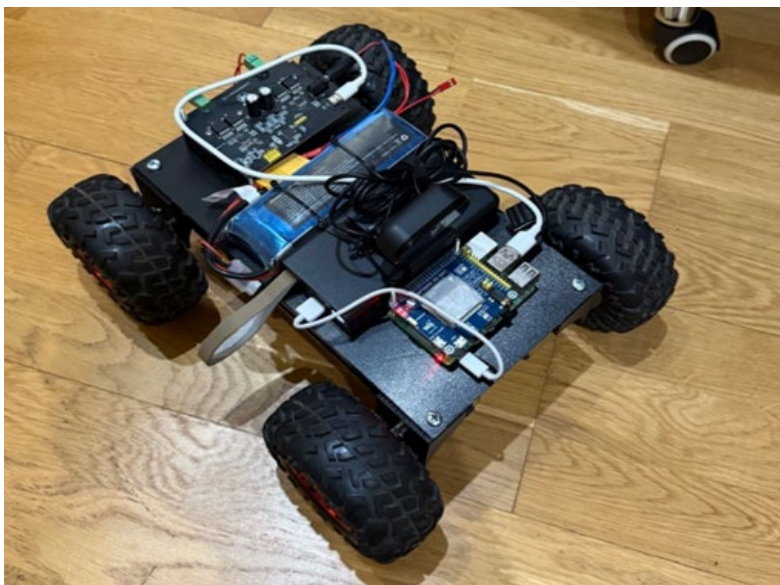


Рис. 3. Вигляд готової розробленої платформи в процесі тестування
Джерело: авторська розробка.

Висновки.

Представлений підхід демонструє ефективність використання сучасних цифрових інструментів на всіх етапах створення роботизованого засобу:

- механічна частина — проєктування в CAD та виготовлення за допомогою адитивних технологій;

- електроніка — схемотехніка та трасування в EasyEDA;

- програмування — прошивка в STM32CubeIDE з використанням HAL.

Цей прототип є базовою платформою для подальших досліджень у сфері автономного руху, комп'ютерного бачення, навігації та автоматизованої логістики.

Список використаних джерел

1. ДСТУ 8302:2015. Бібліографічне посилання. Загальні положення та правила складання. — К.: УкрНДНЦ, 2016. — 16 с.
2. Kalra, N., & Paddock, S. M. (2016). Driving to safety: How many miles of driving would it take to demonstrate autonomous vehicle reliability? RAND Corporation.
3. Siciliano, B., & Khatib, O. (Eds.). (2016). Springer Handbook of Robotics. Springer.