

## THE CREATION OF A DRON REMOTE CONTROLLING CROSS-PLATFORM SOFTWARE APPLICATION

A. Fedorov, E. Shikula

*Institute of Post-Diploma Training of the National University of Food Technologies*

---

**Key words:**

*Dron  
Quadcopter  
Arduino  
ESP  
UWP  
Crossplatform  
Microcontroller boards  
Wi-Fi*

---

**Article history:**

Received 08.05.2019  
Received in revised form  
20.05.2019  
Accepted 11.06.2019

---

**Corresponding author:**

A. Fedorov  
**E-mail:**  
xlr8.19@gmail.com

---

**ABSTRACT**

An analysis of existing analogs of drones was carried out. Almost all of the drones have a flight controller, which is responsible for managing, stabilizing the movement and reporting the state of the drone. These flight controllers have a significant drawback — they cannot be physically modified, that is, it is not possible to connect a new module if the developer did not anticipate its use. The study revealed that there are solutions to create your own quadcopter based on already programmed boards, but the main functionality is very limited, and the price is too high. The developed software solution allows user to create a quadcopter, using cheaper boards with the possibility of further modifications. The software package includes 3 software products: a cross-platform application, board-software for Arduino and ESP.

A quadcopter remote controlling software the base of microcontrollers Arduino and ESP is created. The developed software product is a cross-platform application for quadcopter control, and the software developed for used microcontrollers can be easily modified and supplemented with new functions. In the development process, modern technologies were used, such as C#, XAML, Wi-Fi, UWP which proved that they are reliable enough to be used. The result of this work is the simplified creation of quadcopter, including the use of cheaper components and the expansion of the quadcopter functionality.

The developed software product is a cross-platform application for controlling the quadcopter, and the used board-software can be easily modified and supplemented with new functionality. Using the developed application, it's possible to control the drone by a regular smartphone or PC.

## РОЗРОБКА КРОСПЛАТФОРМЕННОГО ПРОГРАМНОГО КОМПЛЕКСУ ДЛЯ КЕРУВАННЯ ДРОНОМ

А. В. Федоров, О. М. Шикла

*Інститут післядипломної освіти Національного університету харчових технологій*

*У статті проведено аналіз вже існуючих аналогів дронів. Зазначено, що практично всі дрони мають на собі польотний контролер, який відповідає за керування, стабілізацію руху та повідомлення про стан дрона. Однак польотні контролери не піддаються фізичній модифікації, тобто неможливо під'єднати новий модуль, якщо розробник не передбачив його використання. У результаті дослідження з'ясовано, що існують рішення для створення власних дронів на основі вже запрограмованих плат, але їхній головний функціонал дуже обмежений, а ціна завищена. Розроблене програмне рішення дає змогу створити дрон, використовуючи дешевші плати з можливістю подальших модифікацій за допомогою різноманітного обладнання. До складу програмного комплексу входить три програмних продукти: кросплатформений додаток, програмне забезпечення для Arduino та ESP.*

*Програмне забезпечення використовує такі сучасні технології для функціонування, як: XAML, UWP, C#, Arduino IDE. Ці технології забезпечують просту та стабільну роботу пристрою. Розроблене програмне забезпечення для плат працює майже з будь-якої плати подібного сімейства, тобто можна використовувати будь-яку модель плати Arduino на базі AtMega328p та будь-яку плату сімейства ESP-8266.*

*Розроблений програмний продукт є кросплатформеним додатком для контролю дрона, а програмне забезпечення для використаних плат може бути легко модифіковане та доповнене новим функціоналом. Розроблений додаток дає змогу керувати дроном зі звичайного смартфона або ПК.*

**Ключові слова:** дрон, Arduino, ESP, UWP, кросплатформа, мікроконтролери плати, Wi-Fi.

**Постановка проблеми.** Протягом декількох останніх років можна спостерігати значний приріст у продажі та виробництві літальних апаратів на дистанційному керуванні. Вони класифікуються за кількістю моторів (трикоптери, квадрокоптери та октакоптери — з трьома, чотирма та шістьма дугами відповідно). Найбільш розповсюджені квадрокоптери складаються з чотирьох дуг, на яких закріплений мотор і пропелер. Якщо встановити чотири мотори з пропелерами на легкий корпус із карбону, скловолокна або легкої деревини та з'єднати їх з приймачем дистанційного керування через плату керування, оснащену стабілізацією з гіроскопом, можна отримати просту модель, компоненти якої мають майже нескінченну варіативність.

Квадрокоптери — це вільна платформа для винахідників, вони можуть бути використанні у багатьох сферах, варто лиш встановити на них необхідний модуль [6]. У військовій галузі використання безпілотних літальних апа-

ратів надає можливість проводити розвідку на території ворога, патрулювання важливих областей, проведення бойових атак. Використовують дрони також для замірів атмосферних даних на висотах, проведення рятувальних операцій, географічних досліджень тощо. Компанія Amazon, що займається продажем речей з власного онлайн-магазину, має можливість робити доставку за допомогою літальних апаратів. Із зростанням популярності використання дронів з'явився і розвивається новий вид змагань — гонки на дронах. Тому розробка програмного забезпечення для поліпшення та здешевлення зборки квадрокоптерів, а також надання можливості модифікації квадрокоптера за допомогою різноманітного обладнання є актуальною проблемою.

**Аналіз останніх досліджень і публікацій.** Зазвичай, усі квадрокоптери мають на собі польотний контролер [16], який відповідає за стабілізацію, керування та повідомлення про стан дрона. Однак польотні контролери мають значний недолік — вони не піддаються фізичній модифікації, тобто неможливо під'єднати новий модуль, якщо розробник не передбачив його використання. Якщо власник захоче використовувати свій дрон для стабільної відеозйомки або для стеження за об'єктом, програмне забезпечення та польотний контролер повинні мати нативну підтримку цієї функції. Також не буде під'єднуватися додаткове обладнання: сенсор тиску, руху, кисню тощо.

Інакше кажучи, більшість польотних контролерів призначені лише для керування та стабілізації. Майже всі контролери адаптовані для роботи з пультом дистанційного керування. Такі пульти коштують дуже дорого та мають обмежені можливості [13].

Для прикладу візьмемо програмне забезпечення Multiwii [2]. До основних можливостей можна віднести: роботу з компасом, підтримку відеозйомки, дистанційне налаштування через Bluetooth, підтримку GPS за рядом умов. Інший приклад — програмне забезпечення CleanFlight, яке мало найбільшу популярність у середині 2016 року. Оболонка додатка працює як додаток для Google Chrome [6]. Додаток контролює роботу моторів, працює лише на польотних контролерах Naze32, Flip32, CC3D, підтримує Bluetooth, роботу із світлодіодними панелями, записує всі дані польоту за принципом «чорна скринька».

**Мета статті:** розробка програмного забезпечення для поліпшення та здешевлення збирання квадрокоптерів, а також його модифікації за допомогою різноманітного обладнання.

**Викладення основних результатів дослідження.** Після проведення аналізу існуючих програмних продуктів для керування та налаштування літальних апаратів було встановлено, що всі вони вимагають покупки дорогих плат польотного контролера та радіоприймача. Тому було прийнято рішення розробити додаток для використання сучасного смартфона замість радіопередавача через протокол Wi-Fi, а замість використання польотних контролерів розробити програмне забезпечення для плати Arduino, що забезпечить вільну модифікацію та значно здешевить розробки.

Для вирішення цього завдання пропонується виключити із звичайної схеми для конструювання квадрокоптера дві деталі — польотний контролер і радіоприймач; замінити польотний контролер на плату Arduino, а радіоприй-

мач — на плату ESP-12 (ESP8266). Коли літальний апарат буде ввімкнено, плата ESP-12 створить мобільну точку доступу Wi-Fi, користувачу необхідно лише під'єднатися до неї. Цей зв'язок буде аналогом зв'язку за допомогою радіоприймача. Вся інформація, що отримує плата ESP, переадресовується до Arduino. Інформація від Arduino відправляється до додатку оператора. Так користувач зможе оцінити реальний стан квадрокоптера [1; 5; 12].

Для того, щоб квадрокоптер піднявся у повітря, електродвигуни із встановленими пропелерами повинні набрати достатньо обертів, створивши вертикальну тягу. Регулюючи тягу кожного двигуна, можна змінювати нахил квадрокоптера, вирівнюючи таким чином коптер у потрібний момент [17]. Для регулювання квадрокоптера використовується гіроскоп з акселометром [2; 13].

Для стабілізації дрона використовуємо дві осі: вісь нукання (pitch) та крену (roll). Отримуємо дані з гіроскопа, якщо нахил у будь-яку із сторін не відповідає необхідному, починаємо процедуру стабілізації. Під цією процедурою мається на увазі задання певної сили для кожного електродвигуна [18].

Розглянемо приклад: при куті стабілізації (0 pitch, 0 roll) нахил квадрокоптера (15 pitch, 0 roll). Щоб вирівняти його, потрібно додати потужності до передніх електродвигунів M1, M2 та зменшити потужності в M3, M4. Чим ближче дрон знаходиться до бажаної відмітки кута, тим слабша різниця в потужностях між моторами, тобто їх показники повертаються на базове значення — до значення утримання в повітрі (0, 0).

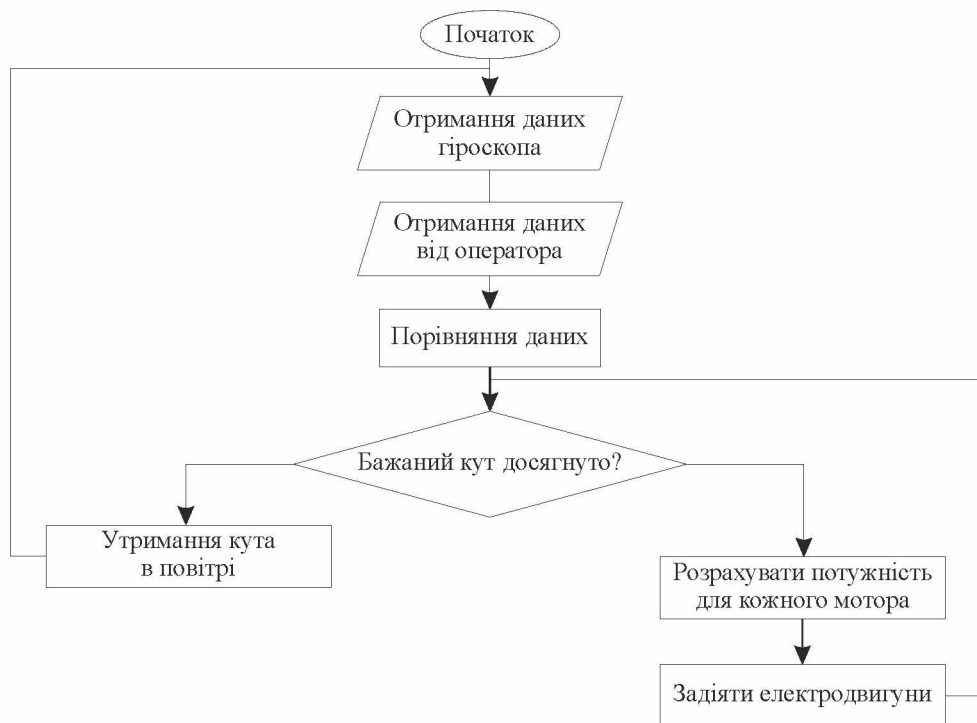
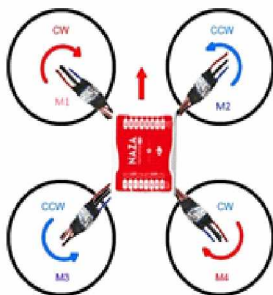


Рис. 1. Опис алгоритму польоту



Для розрахунку нахилу по осі roll застосовується той самий метод. Якщо ж нахил буде одразу по двох осях, треба провести обчислення для кожної. Потужність мотора — середнє математичне між двома результатами.



**Рис. 2. Нумерація моторів**

Щоб дрон почав рух у певний бік, йому потрібно нахилитись, тому команди оператора змінюють кут стабілізації [13]. Розглянемо приклад: квадрокоптер висить у повітрі з кутом (0 pitch, 0 roll). Для того, щоб він почав рухатися вперед, йому потрібно нахилитися також вперед і знаходитися в такому положенні до набрання достатньої швидкості. Після команди оператора рухатись вперед, кут стабілізації змінюється із (0 pitch, 0 roll) на (–15 pitch, 0 roll).

Інструментальні засоби розробки програмного продукту: Microsoft Visual Studio 2017 [7; 8].

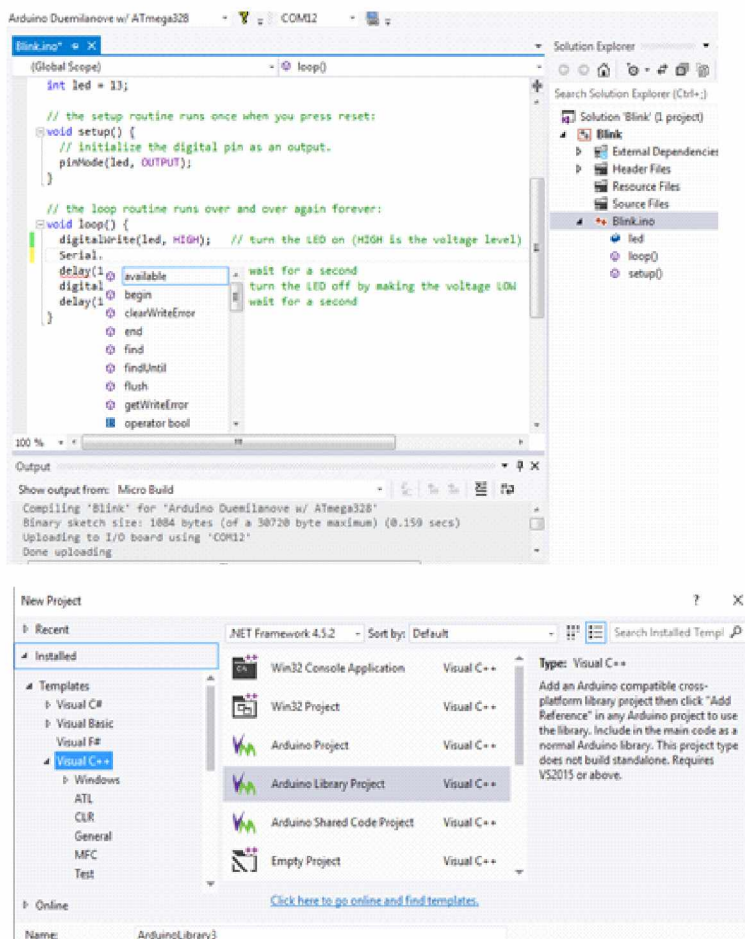
Microsoft Visual Studio — серія продуктів фірми Microsoft, які включають інтегроване середовище розробки програмного забезпечення та ряд інших інструментальних засобів. Ці продукти дають змогу розробляти як консольні програми, так і програми з графічним інтерфейсом, в тому числі з підтримкою технології Windows Forms, а також веб-сайти, веб-додатки, веб-служби як у рідному, так і в керованому кодах для всіх платформ, що підтримуються Microsoft Windows, Windows Mobile, Windows Phone, Windows CE, .NET Framework, .NET Compact Framework та Microsoft Silverlight [7; 10].

Додатки Windows, створені за допомогою класів .NET Framework, мають і інші переваги. Так, можна отримати доступ до служб операційної системи і скористатися іншими перевагами, наданими користувачу обчислювальним середовищем. Доступ до даних можна отримати за допомогою ADO.NET. GDI+ надає додаткові можливості для малювання та розмальовки в додатках Windows Forms. Додатки Windows можуть використовувати методи, які звичайно пропонуються службами XML, надаючи можливість скористатися відомостями і комп'ютерними ресурсами з багатьох джерел і партнерів.

Як і у випадку з іншими додатками, .NET Framework, можна створювати додатки Windows в текстовому редакторі, викликати методи і класи .NET Framework, компілювати додаток в командному рядку і поширювати отриману в результаті виконувати програму. Ці функції аналогічні функціям, доступним у більш ранніх версіях Visual Basic і Visual C++. У Visual Studio 2005 ці можливості розширені й утворюють широкий набір засобів для розробки додатків Windows.

Arduino IDE — крос-платформений додаток, написаний на мові програмування Java, використовується для розробки та завантаження в плати Arduino. Середовище підтримує мови програмування C та C++ лише з невеликими змінами. Вбудовані бібліотеки дають змогу працювати з багатьма пристроями, що можна під'єднати до Arduino, а менеджер пакетів — встановлювати бібліотеки інших розробників для ще більш розширеного функціоналу. Саме за таким принципом плати ESP можна програмувати, використовуючи Arduino IDE [3].

Visual Micro — це плагін для Microsoft Visual Studio, що дає змогу проводити розробку для плат Arduino, використовуючи інструментальне середовище Visual Studio. Тобто програміст користується всіма перевагами функціональної Visual Studio та водночас проводить розробку для Arduino [10].



**Рис. 3. Робота с додатком vMicro у VisualStudio**

*Мови програмування.* Для розробки додатку використовувалася мова програмування C# [4], а для розробки програмного забезпечення для плат — Arduino та ESP — C++.

*Використані технічні засоби.* Плата Arduino складається з мікроконтролера Atmel AVR, а також елементів об'язки для програмування та інтеграції з іншими пристроями. Інтегроване середовище розробки Arduino — це багатоплатформовий додаток на Java, що включає в себе редактор коду, компілятор і модуль передачі прошивки в плату. Середовище розробки засноване на мові програмування Processing та спроектоване для програмування новачками, не знайомими близько з розробкою програмного забезпечення. Мова програмування аналогічна мові Wiring [14; 15]. Програми обробляються за допомогою препроцесора, а потім компілюються за допомогою AVR-GCC. Програми Arduino пишуться на мові програмування C або C++. Середовище розробки Arduino поставляється разом із бібліотекою програм «Wiring». Користувачам необхідно визначити лише дві функції для того, щоб створити програму, яка буде працювати за принципом циклічного виконання:

- `setup()`: функція виконується лише раз при старті програми і дає змогу задати початкові параметри;

- `loop()`: функція виконується періодично, доки плата не буде вимкнена.

ESP8266 — мікроконтролер китайського виробника Espressif з інтерфейсом Wi-Fi. Окрім Wi-Fi, мікроконтролер відрізняється можливістю виконувати програми із зовнішньої флеш-пам'яті з інтерфейсом SPI. Виробник не надає документації на внутрішню периферію контролеру. Замість цього він надає набір бібліотек, через API яких програміст отримує доступ до периферії. Arduino IDE для ESP8266 надає можливість створювати і зашифровувати їх у ESP8266 точно так само, як з Arduino. При цьому плати Arduino не потрібні, це не той випадок, коли ESP8266 використовується як Wi-Fi shield для Arduino. Крім того, є можливість використовувати практично всі архітектурні бібліотеки з ESP8266 після невеликої доробки [5]. Досить багато бібліотек адаптовано для використання з ESP8266 [19].

*Опис прототипної моделі.* Для реалізації розробленої системи керування квадрокоптером потрібно створити модель з такими параметрами:

- плата Arduino на основі чіпа Atmel Mega 328P;
- плата ESP сімейства ESP8266;
- чотири електронних регулятори швидкості (ESC) з підтримкою роботи 2S живлення;
- акселометр та гіроскоп моделі MPU—6050;
- літій-полімерний акумулятор 2S 1P, здатний видати не менше 20 ампер;
- лінійний стабілізатор для живлення плат.

Загальний алгоритм роботи:

1. Літій-полімерний акумулятор живить Arduino, MPU-6050 та ESP через понижуючий конвертер або лінійний стабілізатор. Електронний регулятор швидкості живиться напряму.

2. Arduino отримує дані від гіроскопа/акселометра (MPU-6050) через порти SCA/SCL.

3. ESP створює Wi-Fi точку доступу, до якої з'єднується клієнт. Тепер ESP виступає в ролі ретранслятора, тобто всі дані від додатка йдуть до Arduino та навпаки.



4. Arduino проводить необхідні обчислення, основані на даних з гіроскопа та акселерометра, щоб визначити необхідну потужність для кожного електродвигуна.

5. Кожен ESC з'єднаний з Arduino окремим сигнальним кабелем, використовуючи ШІМ-сигнал, керує потужністю електродвигуна.

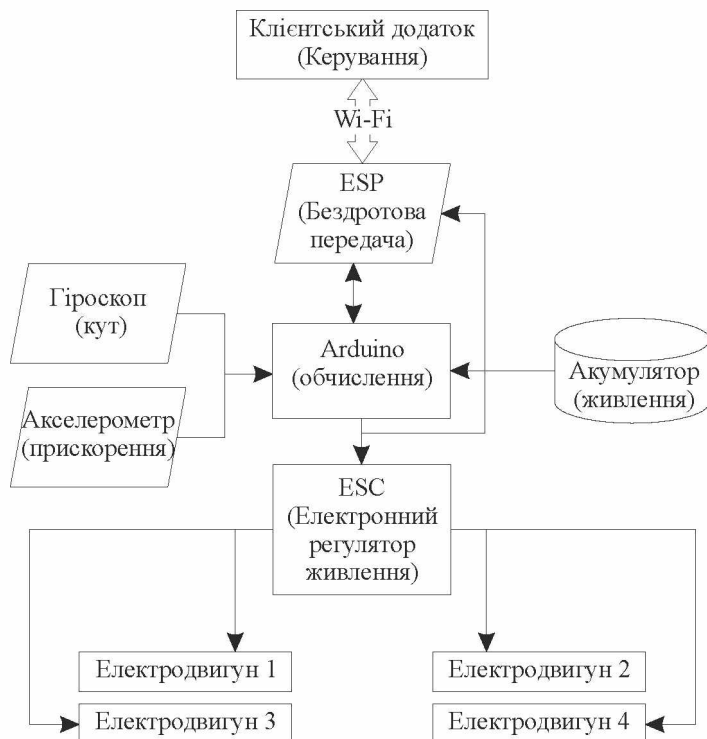


Рис. 4. Загальний алгоритм роботи квадрокоптера-прототипа

Для зарядки акумулятора існує спеціальний кабель-балансир та спеціальне обладнання для підзарядки, наприклад, ітаx b3.

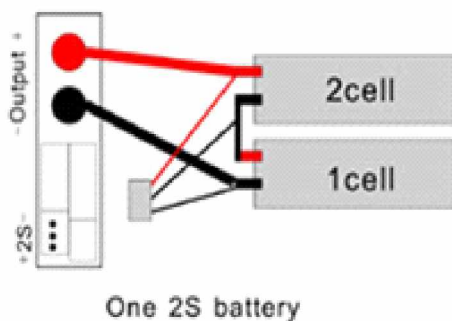


Рис. 5. Схема підключення 2S акумулятора



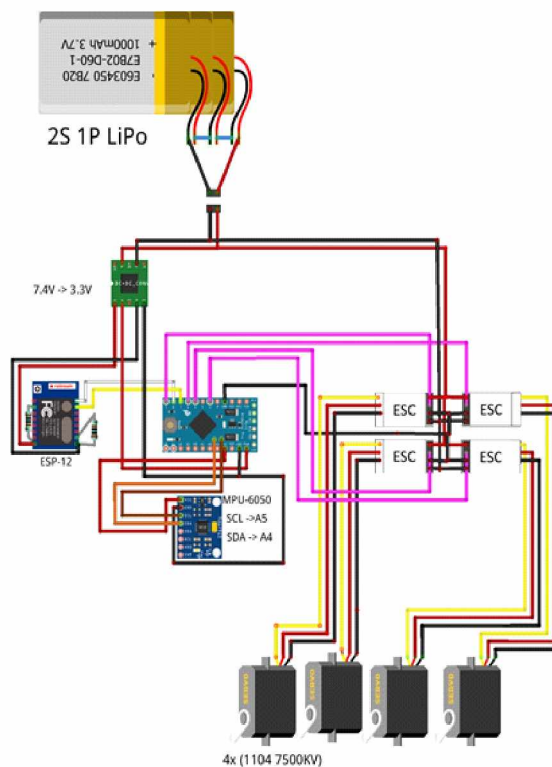


Рис. 6. Рекомендована схема підключення кватеркоптера-прототипа

*Опис основних функцій системи.* Зв'язок через W-Fi. Ця критично необхідна функція дає змогу створювати бездротовий зв'язок між літальним пристроєм і додатком. Для цього використовується технологія бездротового зв'язку Wireless-Fidelity (Wi-Fi) на частоті 2.4 Ghz за стандартом 802.11b/g/n. Плата ESP8266 із встановленим додатком створює бездротову точку доступу із заданим ім'ям і паролем, використовуючи шифрування WPA2. Ця точка доступу стає доступною до підключення для одного пристрою на ефективній відстані до 50 м. ESP буде переадресовувати всі вхідні команди через серійну комунікацію до контролюючої плати та навпаки, використовуючи серійну комунікацію.

*Система команд.* Для передачі даних між літальним апаратом та платою було розроблено систему команд. Основне призначення цієї системи команд — надійна передача інформації і команд між кватеркоптером та додатком і навпаки. Синтаксис команд є простим для людського розуміння, відкриває можливість робити налаштування та пересилати команди власноруч, використовуючи серійну комунікацію, наприклад, додаток PUTTY. Кожна команда — це один текстовий рядок, який складається з двох частин: тип команди (header), зміст команди (body)

*Стабілізація та керування кватеркоптером у повітрі.* Для стабілізації кватеркоптера у повітрі використовується такий алгоритм: кожен момент

часу квадрокоптер звіряє дані з гіроскопом, якщо дані змінилися або надійшов запит на зміну кута (керування) від додатку, тоді Arduino робить нові розрахунки для утримання в повітрі.

Квадрокоптер має чотири електродвигуни та може нахилитися по двох осях — pitch та roll. Щоб вирівняти квадрокоптер по одній із цих осей, треба додати потужності до тих електродвигунів, що нахилились ближче до землі, а для протилежних зменшити потужність. Але треба враховувати, що квадрокоптер може нахилитися по двох осях одночасно, для цього потужність мотора буде визначатися з урахуванням нахилу по другій осі.

*Опис інтерфейсу.* Інтерфейс програмного продукту можна представити у трьох виглядах: у вигляді додатку для персонального комп'ютера, застосунка для мобільного пристрою на базі операційних систем Android та iOS та у режимі консольних команд.

Перша вкладка — панель з'єднання (Connection). Мета цієї панелі — з'єднати додаток із квадрокоптером та відстежити стан цього зв'язку.

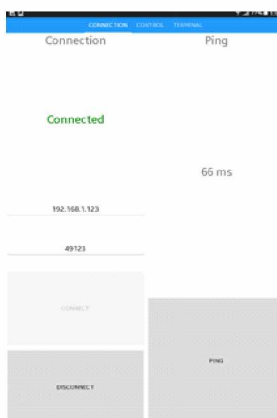


Рис. 7. Панель з'єднання (Connection) інтерфейсу

Друга вкладка — панель керування (Control). Ця вкладка призначена для керування квадрокоптером та відстеження його загального стану.

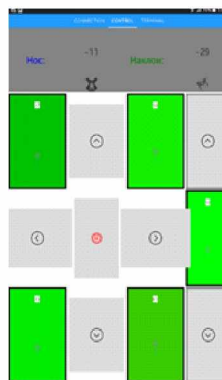
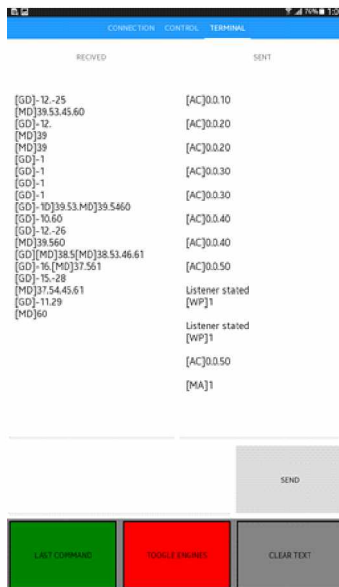


Рис. 8. Панель керування (Control) інтерфейсу

Третя вкладка Terminal призначена для ручного введення команд та отримання даних з квадрокоптера.



**Рис. 9. Вкладка Terminal інтерфейсу**

## **Висновки**

Було створено систему керування квадрокоптером на основі плат Arduino та ESP. Основною метою цього рішення є поліпшення та здешевлення збирання квадрокоптерів, а також надання можливості модифікації квадрокоптера за допомогою різноманітного обладнання. До складу програмного комплексу входить три програмних продукти: кросплатформений додаток, програмне забезпечення для Arduino та ESP.

Розроблене програмне забезпечення для плат працює майже з будь-якої плати подібного сімейства, тобто можна використовувати будь-яку модель плати Arduino на базі AtMega328p та будь-яку плату сімейства ESP-8266.

Інтерфейс, що було створено з використанням технології XAML, дуже гнучкий. Розміри елементів змінюються залежно від розміру екрана. Керування коптером інтуїтивне, наявне відображення статусу пристрою в реальному часі.

У подальшому пропонується розширити можливий функціонал квадрокоптера, додавши до нього такі модулі, як відеокамера, барометр, компас, GPS-трекер, гучномовець, GSM-модем. Також планується реалізація фізичного пульта керування, за допомогою якого можна буде керувати квадрокоптером без використання смартфона.

## **Література**

1. Блум Д. Изучаем Arduino. Инструменты и методы технического волшебства. СПб: БХВ-Петербург, 2018. 336 с.
2. Килби Т., Килби Б. Дроны с нуля. СПб: БХВ-Петербург, 2016. 192 с.

3. Монк С. Программируем Arduino. Профессиональная работа со скетчами. СПб: Питер, 2017. 272 с.
4. Смит Д. C# для профессионалов. Тонкости программирования. М.: Вильямс, 2017. 608 с.
5. Шварц М. Интернет вещей с ESP8266. СПб: БХВ-Петербург, 2018. 192 с.
6. Яценков В. С. Твой первый квадрокоптер: теория и практика. СПб: БХВ-Петербург, 2016. 256 с.
7. Johnson B. Professional Visual Studio 2017. Wrox, 2018. 864 с.
8. Marcin J. Windows Application Development Cookbook. Packt Publishing, 2016. 512 с.
9. Peppers J. Xamarin Cross-platform Application Development. Packt Publishing, 2015. 462 p.
10. Perkins B., Hammer J., Reid J. Beginning C# 7. Programming with Visual Studio 2017. Wrox, 2018. 912 с.
11. Reynolds M. Xamarin Essentials. Packt Publishing, 2014. 234 с.
12. Towaha S. Building Smart Drones with ESP8266 and Arduino: Build exciting drones by leveraging the capabilities of Arduino and ESP8266. Packt Publishing, 2018. 212 с.
13. Квадрокоптеры — как все начиналось? Habr. 2015. URL: <https://habr.com/post/384163/> (дата звернення 20.11.2018).
14. Ефимов И. Программируем квадрокоптер на Arduino (часть 1). Habr. 2014. URL: <https://habr.com/post/227425/> (дата звернення 20.11.2018).
15. Программирование Ардуино | Аппаратная платформа Arduino. URL: <http://arduino.cc> (дата звернення 20.11.2018).
16. Что такое полетный контроллер? Дрономания. URL: <https://dronomania.ru/faq/chto-takoe-poletnyj-kontroller.html> (дата звернення 20.11.2018).
17. Васильков В. Регуляторы хода. RCdesign. URL: [http://www.rcdesign.ru/articles/radio/esc\\_intro](http://www.rcdesign.ru/articles/radio/esc_intro) (дата звернення 20.11.2018).
18. Электродвигатели — устройство, принцип работы, типы, параметры, производители. Инженерные решения. URL: <http://engineering-solutions.ru/motorcontrol/motor/> (дата звернення 20.11.2018).
19. ESP8266: Модули/Азы программирования ESP8266. Онлайн справочник. URL: [https://wikihandbk.com/wiki/ESP8266:Модули/Азы\\_программирования\\_ESP8266](https://wikihandbk.com/wiki/ESP8266:Модули/Азы_программирования_ESP8266) (дата звернення 20.11.2018).
20. eXtensible Application Markup Language (XAML). Xamarin | Microsoft Docs. URL: <https://docs.microsoft.com/en-us/xamarin/xamarin-forms/xaml/index> (дата звернення 20.11.2018).