

Шикула О.М., д.ф.-м.н.; Федоров А.В., магістр

КРОСПЛАТФОРМЕННИЙ ПРОГРАМНИЙ ДОДАТОК ДЛЯ КЕРУВАННЯ КВАДРОКОПТЕРОМ

Shikula E.N., Fedorov A.V. A quadcopter remote controlling cross-platform software application.

A quadcopter - is an unmanned flying machine on a remote control. They consist of 4 arcs on which the motor and propeller are mounted. If you install four propeller motors on a light carbon fiber case, fiberglass or light wood and connect them to the remote control via a control board equipped with a gyro stabilization, you can get a simple drone with components that have almost endless designs.

A quadcopter remote controlling software on 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 who proved they are reliable enough to use. The result of this work is the simplified creation of quadcopter, including the use of cheaper components and the expansion of the quadcopter functionality.

Key words: quadcopter, Arduino, ESP, C#, UWP, crossplatform, microcontroller boards, Wi-Fi.

Шикула О.М., Федоров А.В. Кроссплатформенный программный добавок для управления квадрокоптером.

Квадрокоптеры – це безпілотні літаючі апарати на дистанційному керуванні. Вони складаються з 4х дуг, на яких закріплені мотор та 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 нескінчену варіативність.

Розроблено програмний продукт для управління квадрокоптером. Розроблений програмний продукт є кроссплатформеним додатком для контролю квадрокоптера, а програмне забезпечення для використаних плат може бути легко модифіковане та доповнено новим функціоналом. Для розробки було використано сучасні технології, такі як: C#, XAML, Wi-Fi, UWP, що забезпечують стабільну та просту роботу з розробленим рішенням. Результатом впровадження цього рішення є поліпшення створення квадрокоптерів, а саме використання більш дешевих компонентів та розширення функціоналу квадрокоптера.

Ключові слова: квадрокоптер, Arduino, ESP, C#, UWP, кроссплатформа, мікроконтролерні плати, Wi-Fi.

Шикула Е.Н., Федоров А.В. Кроссплатформенное программное приложение для управления квадрокоптером.

Квадрокоптеры – это беспилотные летающие аппараты на дистанционном управлении. Они состоят из 4х дуг, на которых закреплены мотор и пропеллер. Если установить четыре мотора с пропеллерами на легкий корпус из карбона, стекловолокна или легкого дерева и соединить их с приемником дистанционного управления через плату управления, оснащенную стабилизацией с гироскопом, можно получить простую модель, компоненты которой имеют почти бесконечную вариативность.

Создан программный продукт для управления квадрокоптером на основе плат Arduino и ESP. Разработанный программный продукт представляет собой кроссплатформенное приложение для контроля квадрокоптера, а программное обеспечение для использованных плат может быть легко модифицировано и дополнено новым функционалом. Для разработки были использованы современные технологии, такие как: C#, XAML, Wi-Fi, UWP, обеспечивающие стабильную и простую работу с разработанным решением. Результатом внедрения этого решения является упрощение создания квадрокоптеров, а именно использование более дешевых компонентов и расширение функционала квадрокоптера.

Ключевые слова: квадрокоптер, Arduino, ESP, C#, UWP, кроссплатформа, микроконтролерные платы, Wi-Fi.

Вступ

Протягом декілька останніх років можна спостерігати значний приріст у продажі та виробництві літаючих апаратів на дистанційному керуванні, а саме квадрокоптерів. Ці безпілотні літаючі апарати складаються з 4х дуг, на яких закріплений мотор та 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 нескінчену варіативність.

Квадрокоптери – це вільна платформа для винахідників, вони можуть бути використанні у багатьох сферах, варто лиш встановити на них необхідний модуль.

Наприклад, використання камери у квадрокоптера в сфері спорту дозволяє отримувати неймовірні кадри з перспектив, які неможливо отримати від оператора із відеокамерою, наприклад у гірських велосипедних змаганнях. У військовій галузі використання безпілотних літальних апаратів 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, проведення географічних досліджень, та досліджень для військових цілей. 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. Ентузіасти в змаганнях з'ясовують, чия модель є найшвидшою, найакробатичнішою, тощо.

Аналіз останніх досліджень і постановка проблеми. Зазвичай усі квадрокоптери мають на собі польотний контролер, який відповідає за стабілізацію, керування та повідомлення про стан дрону. Ці 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 свого дрона 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. Та навіть в керування є значний недолік, майже всі контролери адаптовані для роботи с пультом дистанційного керування. Такі пульти зазвичай коштують дуже дорого та мають обмежені можливості.



Рис. 1. Пульт дистанційного керування Flysky FS-i6

Для прикладу візьмемо програмне забезпечення Multiwii. Інтерфейс програмного забезпечення виконано в простому стилі на англійській мові. На екрані відображено дані с гіроскопу та візуальний графік, можемо бачити потужність кожного електродвигуна. До основних можливостей можна віднести: роботу з компасом, підтримку відеозйомки, дистанційне налаштування через Bluetooth, підтримку GPS за рядом умов.

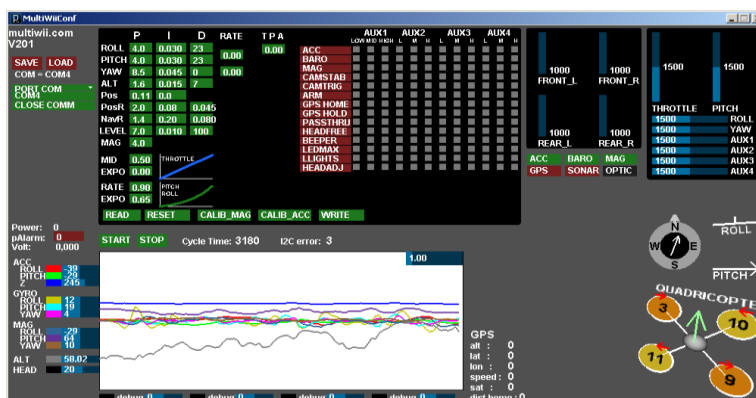


Рис. 2. Інтерфейс MultiWii

Інший приклад - програмне забезпечення CleanFlight, яке мало найбільшу популярність у середині 2016 року, оболонка додатка працює як додаток для Google Chrome. Додаток підтримує роботу з багатьма кількістю моторів, працює лише на польотних контролерах Naze32, Flip32, CC3D; підтримує роботу із світлодіодними панелями, записує усі дані польоту по принципу «чорна скринька», підтримує Bluetooth.

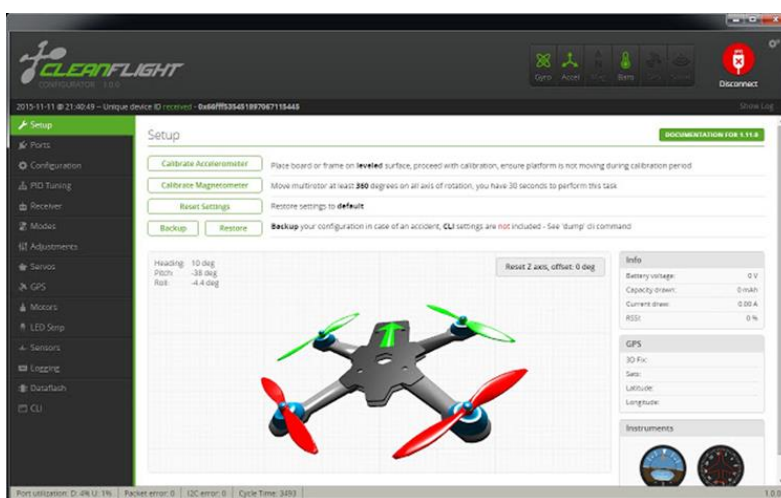


Рис. 3. Інтерфейс додатку CleanFlight Configurator

Мета і завдання дослідження. Метою роботи є розробка програмного забезпечення для поліпшення та здешевлення зборки квадрокоптерів, а також надання можливості модифікації квадрокоптера за допомогою різноманітного обладнання.

Після проведення аналізу існуючих програмних продуктів щодо керування та налаштування літаючих 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, тим самим відкривши можливості до вільної модифікації та значного здешевлення розробки.

Матеріали та методи дослідження. Для вирішення поставленої задачі пропонується виключити із звичайної схеми для конструювання квадрокоптера дві деталі – польотний контролер та радіоприймач; замінити польотний контролер на плату Arduino, а радіоприймач на плату ESP-12 (ESP8266). Коли літальний апарат буде ввімкнено, плата ESP-12 створить мобільну точку доступу Wi-Fi, для наступних дій користувачу необхідно під'єднатися до неї. Цей зв'язок буде аналогом зв'язка за допомогою радіоприймача. В свою чергу всю інформацію, що отримує плата ESP буде переадресовано до Arduino та навпаки, інформація від Arduino відправляється до додатку оператора. Таким чином, користувач зможе побачити реальний стан квадрокоптера. [1][5][12]

Опис алгоритму польоту. 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, тим самим вирівнюючи коптер у потрібний момент. Для регулювання квадрокоптера використовується гіроскоп з акселометром[2][13].

Для стабілізації дрона використовуємо дві осі: вісь никання (pitch) та крену (roll). Отримуємо дані с гіроскопу, 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 для випадку: при куті стабілізації (0 pitch, 0 roll) квадрокоптер нахилився до кута (15 pitch, 0 roll). Щоб вирівняти його, потрібно додати потужності до передніх електродвигунів M1, M2 та зменшити потужності в M3, M4 (нумерація двигунів див. рис. 12). При цьому, 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).

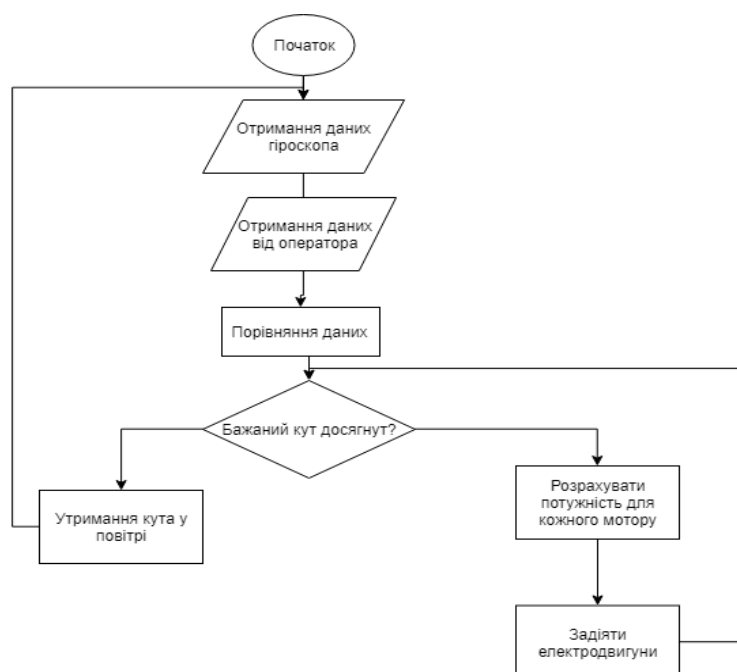


Рис. 4. Опис алгоритму польоту

Для розрахунку нахилу по осі roll буде застосований той самий метод. Якщо ж нахил буде одразу по двом осям, будуть проведені обидва обчислення, а потужність мотора буде середнім математичним між двома результатами.

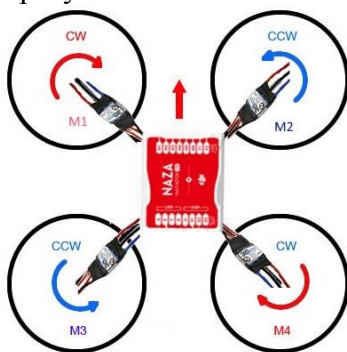


Рис. 5. Нумерація моторів

Щоб дрон почав рух у певний бік, йому потрібно нахилитись. Тому команди оператора змінюють кут стабілізації, тим самим квадрокоптер вирішує, що він нахилений і намагається вирівнятися.[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 доки він не набере достатньої швидкості. Після того, як поступить команда від оператора рухатися вперед, кут стабілізації змінюється із (0 pitch, 0 roll) на (-15 pitch, 0 roll), тим самим використовуючи описаний раніше алгоритм, дрон вирівнюється і починає рух. Як тільки оператор перестає посилати команду рухатися, кут стабілізації знову перейде до (0 pitch, 0 roll).

Інструментальні засоби розробки програмного продукту. Для досягнення поставленої мети найкращим вибором було використовувати візуальну середу Microsoft Visual Studio 2017 для розробки клієнтського додатку та Arduino IDE з додатком vMicro для розробки програмного забезпечення для плат Arduino та ESP.

Arduino IDE – кросс-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 дозволяють працювати з багатьма пристроями, що можна під'єднати до Arduino, а менеджер пакетів дозволяє встановлювати бібліотеки інших розробників для ще більш розширеного функціоналу. Саме за таким принципом плати ESP можна програмувати, використовуючи Arduino IDE [3].

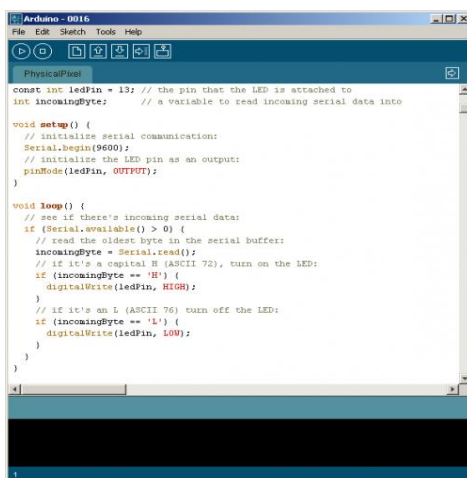


Рис. 6. Інтерфейс середовища Arduino IDE

Visual Micro – це плагін для Microsoft Visual Studio, що дозволяє проводити розробку для плат Arduino, використовуючи інструментальне середовище Visual Studio. Тобто програміст користується усіма перевагами функціональної Visual Studio та водночас проводить розробку для Arduino. Visual Micro дозволяє проводити налаштування, підтримує сторонні бібліотеки, моніторинг входів/виходів.

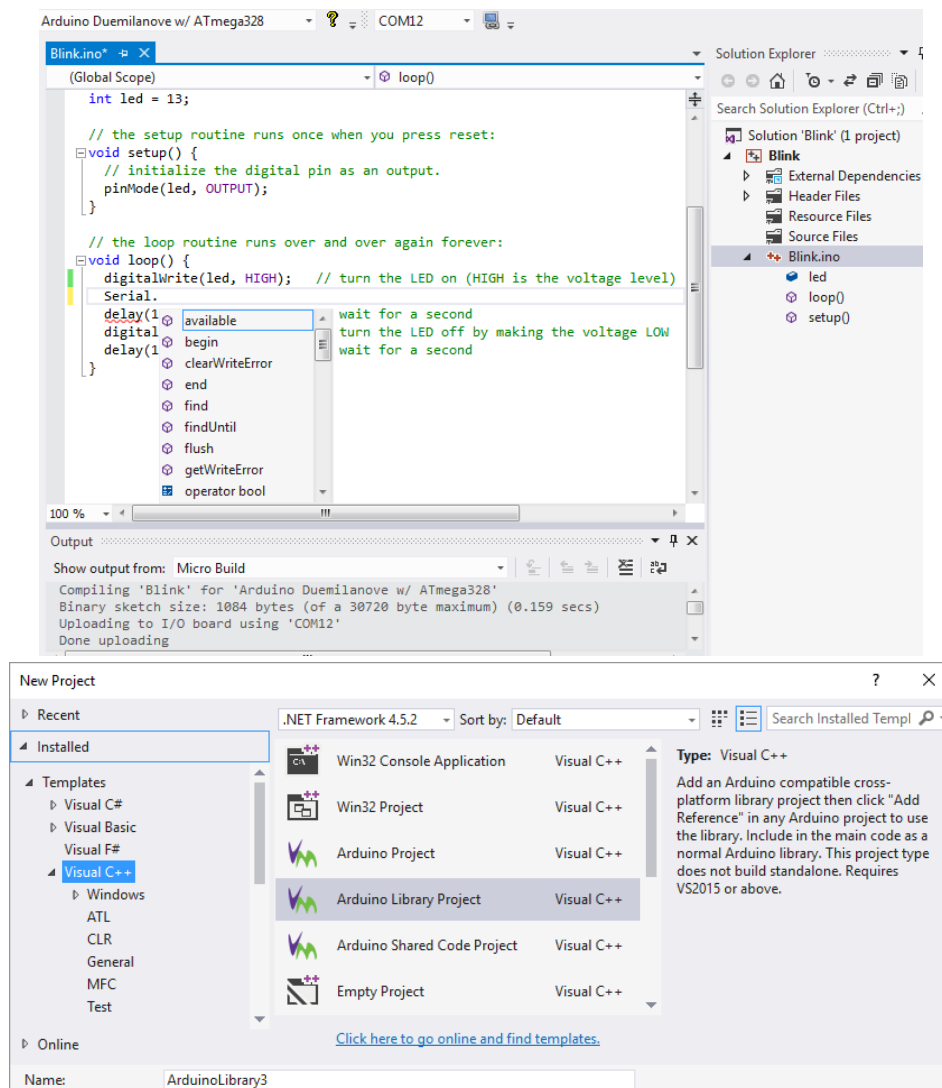


Рис. 7. Робота с додатком vMicro у VisualStudio

Мови програмування. Для розробки додатку використовувалася мова програмування C#, а для розробки програмного забезпечення для плат Arduino та ESP – C++.

Використані технічні засоби. Плата Arduino складається з мікроконтролера Atmel AVR, а також елементів обв'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. Загалом, це C++, доповнений деякими бібліотеками. Програми обробляються за допомогою препроцесора, а потім компілюються за допомогою AVR-GCC.

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 дозволяє створювати і зашифровувати їх у ESP8266 точно так само, як з Arduino. При цьому ніяких плат Arduino не потрібно, 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. В теперешній час досить багато бібліотек адаптовано для використання з ESP8266.

Опис основних функцій системи. Зв'язок через Wi-Fi. Ця критично необхідна функція дозволяє створювати бездротовий зв'язок між літальним пристроєм та додатком. Для цього використовується технологія бездротового зв'язку Wireless-Fidelity (Wi-Fi) на частоті 2.4 Ghz, по стандарту 802.11b/g/n. Плата ESP8266 із встановленим додатком створює бездротову точку доступу із заданим ім'ям та 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 та команд між квадрокоптером та додатком і навпаки. Синтаксис команд є простим для людського розуміння, тим самим відкриває можливість робити налаштування та пересилати команди власноруч, використовуючи серійну комунікацію, наприклад додаток PUTTY. Кожна команда є один текстовий рядок, який складається з двох частин:

Тип команди (header)

Зміст команди (body)

Стабілізація та керування квадрокоптером у повітрі. Для стабілізації квадрокоптера у повітрі використовується наступний алгоритм. Кожен момент часу квадрокоптер звіряє дані с гіроскопом, якщо дані змінилися, або поступив запит на зміну кута (керування) від додатку, тоді Arduino робить нові розрахунки для утримання у повітрі.

Квадрокоптер має чотири електродвигуни та може нахилитися по двом осям – pitch та roll. Щоб вирівняти квадрокоптер по одній із цих осей, треба додати потужності до тих електродвигнів, що нахилились ближче до землі, а для протилежних зменшити потужність. Але треба враховувати, що квадрокоптер може нахилитися по двом осям одночасно, для цього потужність мотора буде визначатися з врахуванням 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.

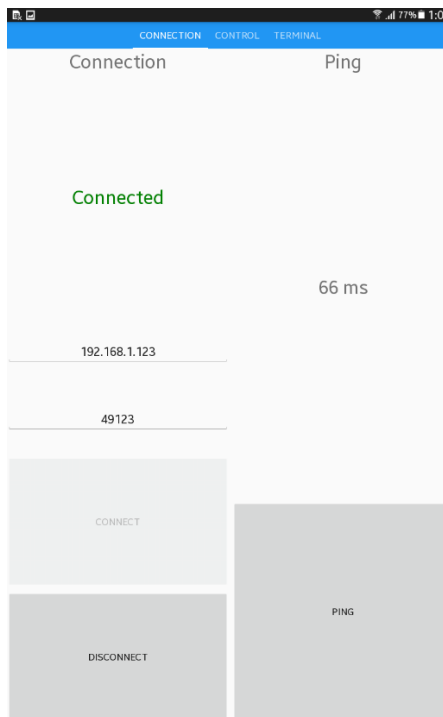


Рис. 8. Панель з'єднання (Connection) інтерфейсу

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

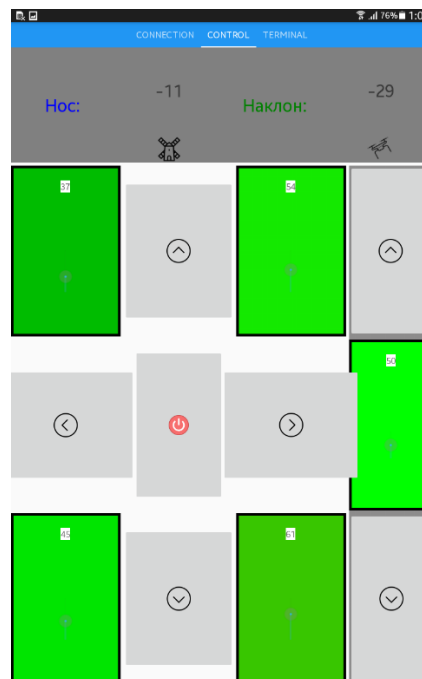


Рис. 9. Панель керування (Control) інтерфейсу

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

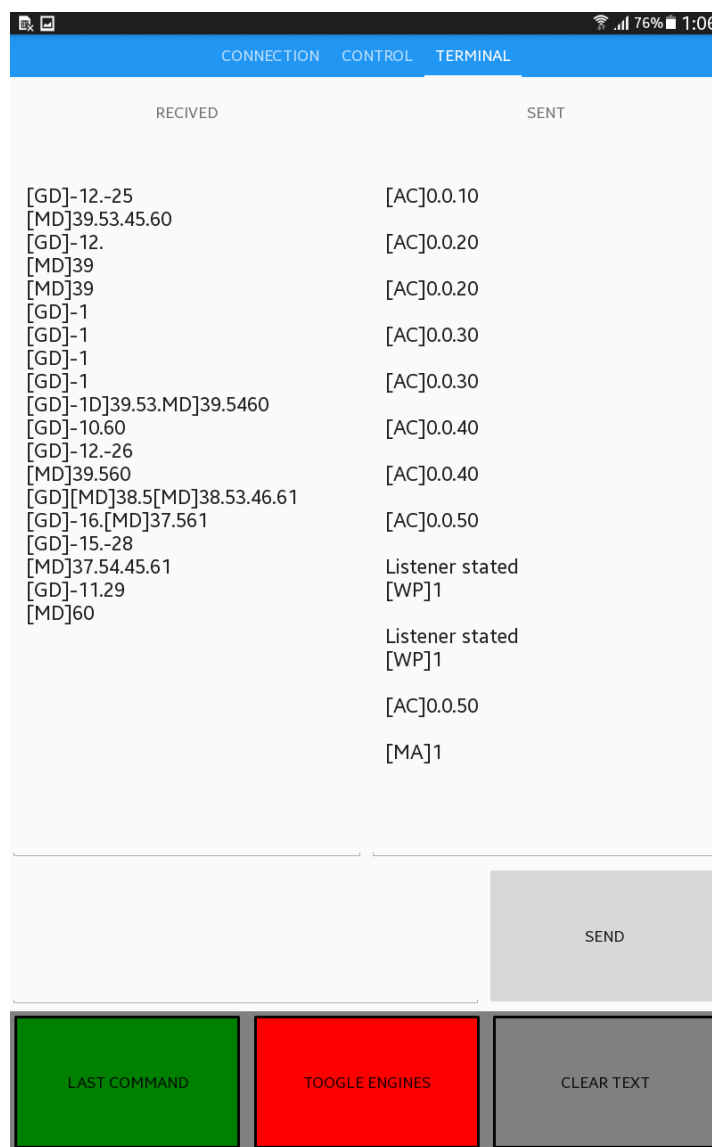


Рис. 10. Вкладка Terminal інтерфейсу

Тестування програмного продукту. Для виконання тестових завдань була створена модель, в якій використовувались наступні компоненти:

- Arduino Pro Mini AtMega328P 3.3v 16Mhz
- ESP8266 ESP-12E
- 4 регулятора швидкості EMAX Bullet 6A 2S BLHeliI_S Dshot
- DC-DC конвертер 3A
- Акселометр/гіроскоп/компас MPU-6050
- Акумулятор Infinity 550mAh 2S 7.4V LiPo Violence Lithium Battey
- Корпус ELF 88mm Micro Quadcopter
- 4 електродвигуни 1104 7500KV 2-3S Motor
- Лопаті 2030 Prop

Тестова модель квадрокоптера виглядає наступним чином:

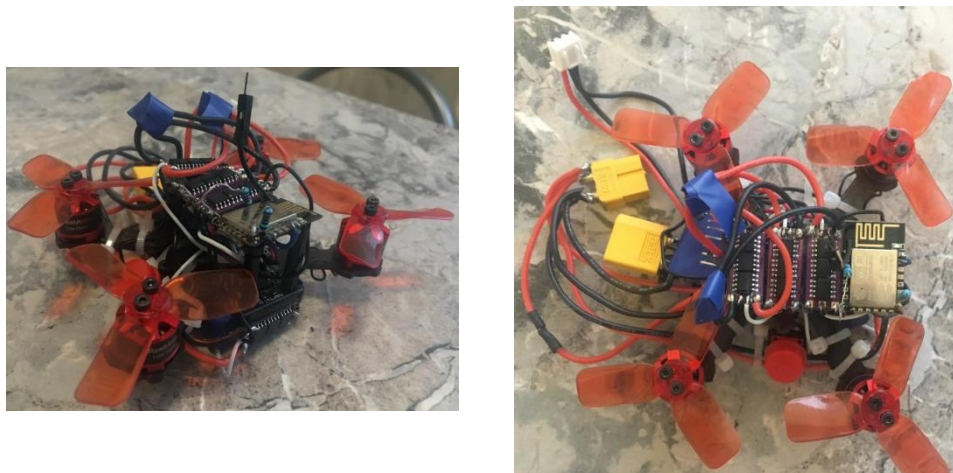


Рис. 11. Тестова модель квадрокоптера

Висновки

Було створено систему керування квадрокоптером на основі плат Arduino та ESP. Основною метою цього рішення є поліпшення та здешевлення зборки квадрокоптерів, а також надання можливості модифікації квадрокоптера за допомогою різноманітного обладнання. До складу програмного комплексу входить 3 програмних продуктів: кросплатформний додаток, програмне забезпечення для Arduino та ESP.

Було проведено аналіз вже існуючих аналогів. В результаті дослідження з'ясувалося, що існують рішення для створення власних квадрокоптерів на основі вже запрограмованих плат, але їх головний функціонал дуже обмежений, а ціна завищена. Розроблене програмне рішення дозволяє створити квадрокоптер, використовуючи дешевші плати із можливістю подальших модифікацій. Використовуючи розроблений додаток, керувати дроном стає можливим зі звичайного смартфона або ПК.

Програмне забезпечення використовує сучасні технології для функціонування, такі 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.

Інтерфейс, що було створено з використанням технології XAML, дуже гнучкий. Розміри елементів змінюються в залежності від розміру екрану. Керування коптером дуже інтуїтивне, мається відображення статусу пристрою в живому часі.

Список використаної літератури

1. Блум Д. Изучаем Arduino. Инструменты и методы технического волшебства / Джереми Блум. – Санкт-Петербург: БХВ-Петербург, 2018. – 336 с.
2. Килби Т. Дроны с нуля / Т. Килби, Б. Килби, В. Яценков. – Санкт-Петербург: БХВ-Петербург, 2016. – 192 с.
3. Монк С. Програмуємо Arduino. Професійна робота со скетчами / Саймон Монк. – Санкт-Петербург: Питер, 2017. – 272 с.
4. Скит Д. C# для професіоналов. Тонкості програмування / Джон Скит. – Москва: Вільямс, 2017. – 608 с.
5. Шварц М. Інтернет вещей с ESP8266 / Марко Шварц. – Санкт-Петербург: БХВ-Петербург, 2018. – 192 с.
6. Яценков В. С. Твой первый квадрокоптер: теория и практика / Валерий Станиславович Яценков. – Санкт-Петербург: БХВ-Петербург, 2016. – 256 с. – (1).

7. Johnson B. Professional Visual Studio 2017 / Bruce Johnson., 2017. – 864 с. – (1).
8. Marcin J. Windows Application Development Cookbook / Jamro Marcin. – Packt, 2016. – 512 с.
9. Peppers J. Xamarin Cross-platform Application Development / Jonathan Peppers., 2014. – 262 с.
10. Perkins B. Beginning C# 7 Programming with Visual Studio 2017 / B. Perkins, J. Hammer, J. Reid., 2018. – 806 с. – (1).
11. Reynolds M. Xamarin Essentials / Mark Reynolds., 2014. – 234 с.
12. Towaha O. Building Smart Drones with ESP8266 and Arduino / Omar Towaha., 2018. – 212 с.
13. Квадрокоптеры — как все начиналось? / Хабр [Электронный ресурс]. – 2015. – Режим доступа до ресурсу: <https://habr.com/post/384163/>.
14. Ефимов И. Програмуємо квадрокоптер на Arduino (часть 1) / Хабр [Электронный ресурс] / Иван Ефимов. – 2014. – Режим доступа до ресурсу: <https://habr.com/post/227425/>.
15. Программирование Ардуино | Аппаратная платформа Arduino [Электронный ресурс] – Режим доступа до ресурсу: <http://arduino.ru/Reference>
16. Что такое полетный контроллер? [Электронный ресурс] – Режим доступа до ресурсу: <https://dronomania.ru/faq/chto-takoe-poletnyj-kontroller.html>.
17. Васильков В. Регуляторы хода [Электронный ресурс] / Владимир Васильков – Режим доступа до ресурсу: http://www.rcdesign.ru/articles/radio/esc_intro.
18. Электродвигатели - устройство, принцип работы, типы, параметры, производители [Электронный ресурс] – Режим доступа до ресурсу: <http://engineering-solutions.ru/motorcontrol/motor/>.
19. ESP8266:Модули/Азы программирования ESP8266 — Онлайн справочник [Электронный ресурс] – Режим доступа до ресурсу: https://wikihandbk.com/wiki/ESP8266:Модули/Азы_программирования_ESP8266
20. eXtensible Application Markup Language (XAML) - Xamarin | Microsoft Docs [Электронный ресурс] – Режим доступа до ресурсу: <https://docs.microsoft.com/en-us/xamarin/xamarin-forms/xaml/index>.

Автори статті

Шикла Олена Миколаївна – доктор фізико-математичних наук, професор, завідувач кафедри Інформаційних технологій, Державний університет інфраструктури і технологій, Київ, Україна.

Федоров Андрій В'ячеславович – магістр, Державний університет телекомунікацій, Київ, Україна.

Authors of the article

Shykula Olena Mykolaivna – candidate of Science (technic), assistant professor, assistant professor of Department of Telecommunication systems and networks, State University of Telecommunications, Kyiv, Ukraine.

Fedorov Andrii Viacheslavovych - magistr, State University of Telecommunications, Kyiv, Ukraine.

Дата надходження в редакцію 14.11.2020 р. Рецензент: д.т.н., професор В.В. Вишнівський