НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ»

ім. Ігоря Сікорського

ННК "Інститут прикладного системного аналізу"
(повна назва інституту/факультету)

Кафедра Системного проектування
(повна назва кафедри)

«До захисту допущено»

Завідувач кафедри
_____ А.І.Петренко
(підпис) (ініціали, прізвище)
"___"_____2017 р.

Дипломна робота

на здобуття ступеня бакалавра

з напряму підготовки 6.050101 Комп'ютерні науки
(код і назва)

на тему: Розробка програмно-апаратної реалізації захищеного бездротового інтерфейсу до безпілотного літального апарату

Виконав: студент _4__ курсу, групи __ДА-31_
(шифр групи)

Бєлоносов Тимофій Михайлович _____
(прізвище, ім'я, по батькові) (підпис)

Керівник доцент, к.т.н., Кірюша Б.А. _____
(посада, науковий ступінь, вчене звання, прізвище та ініціали) (підпис)

Консультант економічний __доцент к.е.н Рощина Н. В._____ _____
(назва розділу) (посада, вчене звання, науковий ступінь, прізвище, ініціали) (підпис)

Рецензент _____ _____
(посада, науковий ступінь, вчене звання, науковий ступінь, прізвище та ініціали) (підпис)

Нормоконтроль _____ст. викладач Британов О.А._____

Засвідчую, що у цій дипломній роботі немає запозичень з праць інших авторів без відповідних посилань.
Студент _____
(підпис)

Київ – 2017 року

Національний технічний університет України
«Київський політехнічний інститут»
ім. Ігоря Сікорського

Факультет (інститут) ННК "Інститут прикладного системного аналізу"
(повна назва)

Кафедра Системного проектування
(повна назва)

Рівень вищої освіти – перший (бакалаврський)

Напрям підготовки 6.050101 Комп'ютерні науки

(код і назва)

ЗАТВЕРДЖУЮ
Завідувач кафедри
_____ А.І.Петренко
(підпис) (ініціали, прізвище)
«___»_____2017 р.

ЗАВДАННЯ

на дипломний проект (роботу) студенту

Бєлоносову Тимофію Михайловичу

(прізвище, ім'я, по батькові)

1. Тема проекту (роботи) Розробка програмно-апаратної реалізації захищеного бездротового інтерфейсу до безпілотного літального апарату
керівник проекту (роботи) Кірюша Богдан Анатолійович, к.т.н., доцент

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом по університету від «___»_____ 20__ р. №_____

2. Строк подання студентом проекту (роботи) 06.06.2017

3. Вихідні дані до проекту (роботи) _____

- Raspberry Pi 3 Model B
- 2x Raspberry Pi 2 Model B
- Польотний контролер
- Платформа коптера DJI
- FPV камера
- Video capture device
- 2x DVB-T модем

4. Зміст розрахунково-пояснювальної записки (перелік завдань, які потрібно розробити)

1. Проаналізувати предметну область виділивши особливості та вразливості систем зв’язку з БПЛА
2. Проаналізувати доступні методи реалізації компонент системи зв’язку з БПЛА
3. Розробити прототип лінії зв’язку на основі проведених досліджень
4. Провести функціонально-вартісний аналіз програмного продукту

5. Перелік графічного матеріалу (з точним зазначенням обов’язкових креслеників, плакатів тощо)

1. Діаграма потоків даних розробленої системи зв’язку з БПЛА – плакат
2. Загальна ієрархічна схема БПЛА – плакат
3. Таблиця порівняння мікрокомп’ютерів – плакат

6. Консультанти розділів проекту (роботи)*

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Економічний	Рощина Н. В., доцент.		

7. Дата видачі завдання 01.02.2017

Календарний план

№ з/п	Назва етапів виконання дипломного проекту (роботи)	Строк виконання етапів проекту (роботи)	Примітка
1	Отримання завдання	01.02.2017	
2	Збір інформації	15.02.2017	
3	Вивчення варіантів реалізації та вибір варіанту для розробки	28.02.2017	
4	Вибір способу реалізації	10.03.2017	
5	Розробка плану тестування	15.03.2017	
6	Розробка програмної моделі	25.03.2017	
7	Розробка опису системи	25.04.2017	
8	Тестування прототипу	30.04.2017	
9	Оформлення дипломної роботи	31.05.2017	
10	Отримання допуску до захисту та подача роботи в ДЕК	06.06.2017	

Студент _____ Т.М. Бєлоносов

(підпис) (ініціали, прізвище)

Керівник проекту (роботи) _____ Б.А. Кірюша

(підпис) (ініціали, прізвище)

* Консультантом не може бути зазначено керівника дипломного проекту (роботи).

АНОТАЦІЯ

до бакалаврської дипломної роботи Бєлоносова Тимофія Михайловича

на тему: «Розробка програмно-апаратної реалізації захищеного бездротового інтерфейсу до безпілотного літального апарату»

Дипломна робота присвячена аналізу вразливостей у системах зв’язку з БПЛА та побудові прототипу безпілотника із захищеним зв’яком.

Метою дипломної роботи є розробка апаратно програмної архітектури двостороннього зв’язку між БПЛА та мобільною наземною станцією. Вибір та реалізація алгоритмів шифрування та захисту відео потоку та потоків керування БПЛА. Виготовлення та тестування прототипу системи зв’язку з БПЛА.

В роботі проведено аналіз вразливостей систем зв’язку, можливих реалізацій основних систем БПЛА у захищеному режимі, порівняння апаратних і програмних складових що дозволяють реалізувати поставлені задачі.

В ході виконання дипломної роботи був побудований і протестований робочий прототип БПЛА з захищеним зв’язком.

Загальний обсяг роботи – 74 сторінки, 11 рисунка, 8 таблиць, 16 бібліографічних посилань.

Ключові слова: БПЛА, відео, захищений зв’язок, автоматизована система, радіоелектронна боротьба.

АННОТАЦИЯ

К бакалаврской дипломной работе Белоносова Тимофея Михайловича

На тему: «Разработка программно-аппаратной реализации защищенного беспроводного интерфейса для беспилотного летательного аппарата»

Дипломная работа посвящена анализу уязвимостей в системах связи с БПЛА и построении прототипа беспилотника с защищенной связью.

Целью работы является разработка аппаратно-программной архитектуры двусторонней связи между БПЛА и мобильной наземной станцией. Выбор и реализация алгоритмов шифрования и защиты видеопотока и потоков управления БПЛА. Изготовление и тестирование прототипа системы связи с БПЛА.

В работе проведен анализ уязвимостей систем связи, возможных реализаций основных систем БПЛА в защищенном режиме, сравнение аппаратных и программных составляющих позволяющие реализовать поставленные задачи.

В ходе выполнения дипломной работы был построен и протестирован рабочий прототип БПЛА с защищенным связью.

Общий объем работы - 74 страницы, 11 рисунка, 8 таблиц, 16 библиографических наименований.

Ключевые слова: БПЛА, видео, защищенная связь, автоматизированная система, радиоэлектронная борьба.

AN ABSTRACT OF THE THESIS OF

Bielonosov Tymofii for the degree bachelor of the computer science in NTUU "KPI"

Title: "Development of software and hardware implementation of a secure wireless interface for an unmanned aerial vehicle"

Thesis is devoted to analysis of vulnerabilities in UAV communication systems and the construction of a prototype of the UAV with protected connection.

The aim of the work is the development of the hardware and software architecture of full duplex communication layer between UAV and mobile ground station. Selection and implementation of algorithms for encryption and protection of video stream and control streams of UAV. Manufacturing and testing of the prototype of the communication system with UAV.

The analysis of vulnerabilities of communication systems, possible realizations of the main UAV systems in a protected mode, comparison of hardware and software components, allowing to realize the tasks, was carried out.

In the course of the graduation work, a working prototype of a UAV with a protected connection has been built and tested.

The total volume of work - page 74, 11 drawing tables 8, 16 bibliographic references.

Tags: UAV, video, secure communication layer, automated system, electronic warfare.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ,СКОРОЧЕНЬ І ТЕРМІНІВ

БПЛА - Безпілотний літальний апарат

UAV - Unmanned aerial vehicle

ДПЛА - Безпілотний дистанційно-пілотований літальний апарат

НСК - Наземна станція керування

САП - Системіа автоматичного порятунку

GPS - Global positioning system

РЕБ - радіоелектронна боротьба

ГЛОНАСС - Глобальна навігаційна супутникова система

FPV - First person view

FOV - Field of view

RTC - Real time clock

GPIO - General-purpose input/output

GPU - Graphics processing unit

STANAG - Standardization agreement

JTIDS - Joint tactical information distribution system

GMSK - Gaussian minimum shift keying

WNW - Wideband networking waveform

NTP - Network time protocol

AES - Advanced encryption standard

RPi - Raspberry Pi

PAL - Phase alternating line

OMX - OpenMax

ФВА - Функціонально-вартісний аналіз

ПП - Програмний продукт

ВСТУП

Як правило, основний обов'язок, який покладено на комплекси БПЛА, - проведення розвідки важкодоступних районів, в яких отримання інформації звичайними засобами, включаючи авіарозвідку, ускладнене або ж є небезпечним для здоров'я та навіть життя людей. Крім військового використання застосування комплексів БПЛА відкриває можливість оперативного і недорогого способу обстеження важкодоступних ділянок місцевості, періодичного спостереження заданих районів, цифрового фотографування для використання в геодезичних роботах і у випадках надзвичайних ситуацій. Отримана бортовими засобами моніторингу інформація повинна в режимі реального часу передаватися на пункт управління для обробки і прийняття адекватних рішень.

В наш час найбільшого поширення набули тактичні комплекси мікро і міні-БПЛА. У зв'язку з більшою злітною масою міні-БПЛА за своїм функціональним складом найбільш повно представляє склад бортового обладнання, що відповідає сучасним вимогам до багатофункціонального розвідувального БПЛА.

Спостерігається різке збільшення застосування різних безпілотних авіаційних комплексів у всіх сферах життєдіяльності людини - від торгівлі до військової справи. Безпілотні авіаційні комплекси, як правило, включають в себе оператора (пілот-оператор, пункт управління), безпілотний літальний апарат та канали зв'язку, проте їх захисту від зовнішніх програмно-апаратних впливів, не дивлячись на зростання кількості інцидентів, не приділяється достатньої уваги.

Атаки можуть бути спрямовані на перехоплення управління, виведення з ладу БПЛА, отримання розвідувальної інформації або для подальшої атаки на пілота-оператора і взаємодіючі з ним системи.

Частковий захист інформації, що циркулює в БПЛА, здійснюється тільки в БПЛА військового призначення - інформація, передана по каналу управління, інформаційного каналу, а також інформація, що містить відомості про завдання, і інформація, що зберігається в пристрої зберігання видової інформації.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Загальний огляд

Безпілотний літальний апарат – літальний апарат, який літає та сідає без фізичної присутності пілота на його борту. За сучасним визначенням, «безпілотником» є тільки той апарат, який знаходиться під постійним дистанційним контролем пілота або пілотів і призначений для повернення на аеродром і для подальшого повторного використання.

Раніше радіокеровані та повністю автоматизовані апарати об'єднували поняттям безпілотна авіація – літаки, керування (пілотування) якими здійснюється без пілота, за допомогою приладів різних систем, що засобами радіо (радіолокації, телебачення) подають команди на автопілот. Елементи системи керування містяться поза літаком і можуть бути на землі, на воді і в повітрі, на місці старту, на маршруті польоту і в районі цілі.

Залежно від принципів керування є наступні різновиди безпілотних літальних систем:

- безпілотні некеровані;
- безпілотні автоматичні;
- безпілотні дистанційно-пілотовані літальні апарати (ДПЛА).

У авіації після 2000 року йде стрімке розширення саме останнього типу апаратів, і про них йдеться, коли вживають термін «безпілотник», «дрон», або абревіатуру UAV. Тобто, під терміном «безпілотник», «БПЛА», «UAV» мається на увазі саме повітряне судно, яким через канали зв'язку керує один або декілька пілотів.

Спочатку БПЛА розроблялись для військових і оборонних цілей, але сьогодні вони все частіше використовуються для різних цивільних цілей, в тому числі фотографій, рятувальних операцій, моніторингу інфраструктури, сільського господарства і авіа зйомки.

1.2 Системи зв’язку

1.2.1 Особливості систем зв’язку БПЛА

На сьогоднішній день стало можливим здійснення управління літаками за допомогою автопілоту при повній відсутності зв'язку між бортом літального апарату (ЛА) і НСК. При цьому польотне завдання виконується в автономному режимі. Тим не менш, це не дозволяє говорити про те, що командно-телеметрична радіолінія зв'язку може бути виключена зі складу БПЛА. В силу підвищеної складності і вартості комплексу при його експлуатації потрібен постійний контроль за станом ЛА в повітрі. Крім того, іноді виникає необхідність коригування параметрів польоту БПЛА.

Актуальним завданням також є передача даних корисного навантаження ЛА на НСК. В цьому випадку потрібно забезпечити передачу великого обсягу даних при заданих вимогах по смузі пропускання, ймовірності бітової помилки та ін.

При створенні малих і надмалих БПЛА висуваються вимоги щодо мінімізації розмірів приймально-передавального і антенно-фідерного обладнання.

Підвищені вимоги по відмовостійкості пред'являються до обладнання БПЛА, який здійснює навігацію і літаководіння, що забезпечує режими ручної посадки (якщо це необхідно), до сервоприводу і системі автоматичного порятунку (САП). Перераховане обладнання входить в першу групу класифікації і забезпечує надійність комплексу БПЛА в цілому. Поломка будь-якого елементу обладнання першої групи призводить до негайного припинення виконання льотного завдання та поверненню ЛА на базу. Якщо ж це неможливо, спрацьовує САП і відбувається викид парашута.

Решту обладнання ЛА відносять до другої групи класифікації. При виході з ладу обладнання цієї групи рішення про подальші дії приймається

керуючим персоналом комплексу. Взаємодія обладнання першої і другої груп здійснюється за допомогою керуючих інтерфейсів.

В процесі роботи системи зв'язку оцінюються ймовірності бітової помилки для кожного каналу зв'язку і приймається рішення про розподіл командно-телеметричного потоку даних між каналами. Використання декількох каналів зв'язку підвищує надійність системи передачі даних і в той же час є надлишковим з точки зору ефективного використання радіочастотного спектру. Одним із способів підвищення ефективності системи зв'язку є адаптивна робота системи, яка має на увазі передачу по командно-телеметричних каналах зв'язку частини даних корисного навантаження, обсяг яких варіюється в залежності від поточних умов передачі радіосигналу.

Як правило, максимальна відстань для прямого радіозв'язку між БПЛА цивільного призначення та НСК на сьогоднішній день складає не більше 100 км. Для командно-телеметричного зв'язку на великих відстанях можливе використання супутникового зв'язку. У цьому випадку потік даних обмежується мінімально необхідною інформацією про стан БПЛА, інтервал передачі якої може складати, наприклад, від 30 до 300 секунд.

Перспективним напрямком у розвитку систем зв'язку з БПЛА є використання частотних діапазонів вище 5 ГГц. При цьому стає можливою передача великого обсягу даних корисного навантаження в режимі реального часу (наприклад, це можуть бути зображення з датчиків випромінювання різного діапазону довжин хвиль). Факторами, різко обмежуючими радіус дії радіосистеми зв'язку при використанні даних діапазонів, є сильна залежність умов поширення електромагнітних хвиль від погодних умов, необхідність прямої видимості і вплив багатопроменевості.

1.2.2 Способи злому ліній зв’язку

«Ахіллесовою п'ятою» БПЛА є вразливість каналів зв'язку - сигнали GPS навігаторів, як і будь-які сигнали, що приймаються і відсилаються літальним

апаратом, можна глушити, перехоплювати і підміняти. Для управління БПЛА потрібні канали зв'язку високої пропускної здатності, які складно організувати.

У 2012 році вченими з Техаського університету в Остіні була доведена практична можливість злому і перехоплення управління БПЛА шляхом так званого «GPS-спуфинга», але тільки для тих апаратів, які використовують незашифрований цивільний сигнал GPS.

У Криму був посаджений шляхом перехоплення управління дрон армії США (MQ-5B Hunter зі складу 66-ї американської бригади військової розвідки, офіційно дислокованої в Баварії), широко відомий випадок захоплення повністю справного важкого дрона США в Ірані (в 2011 посадив на своїй території американський секретний безпілотник RQ-170 Sentinel).

Для боротьби із військовими безпілотниками Росії та Китаю у США розроблено, і у 2015 році випробовано спеціальну гармату «chain gun».

Портал Yahoo повідомив, що фірма Bettelle (США) розробила портативний пристрій направленої дії для радіоелектронної боротьби із невеликими дронами-шпигунами (такими, на зразок, як популярний коптер DJI Phantom). Пристрій важить приблизно 4 кг, має приклад та приціл. За даними фірми ця «електронна рушниця» готова до дії через 0.1 сек після включення, і може паралізувати дрон-шпигун на відстані до 300 м. У 2016 році електронна рушниця Bettelle Drone Defender постачається до державних установ США.

На Донбасі проросійські найманці, як повідомлялося, успішно нейтралізують безпілотники місії ОБСЄ за допомогою радіотехнічних засобів радіоелектронної боротьби (РЕБ'ів) російського виробництва.

Станом на початок російської збройної агресії проти України, Збройні сили України фактично не мали власних сучасних безпілотних літальних апаратів. Наявні на озброєнні Ту-141 «Стриж» були морально застарілі. Гостру потребу в безпілотних літаках-розвідниках спершу взялися задовольняти волонтери, адаптуючи цивільні апарати до вимог військових. Були створені, зокрема, БПЛА «Фурія», «Кажан-1», PD-1.

Зараз на озброєнні багатьох армій є велика кількість різноманітних систем радіоелектронної боротьби (РЕБ). Як вже говорилося раніше, для успішного виведення з ладу ворожого дрона потрібно встановити частоти, на яких здійснюється управління апаратом, а потім «забити» їх перешкодами. Далеко не всі сучасні безпілотні літальні апарати мають на борту автоматику, здатну перехопити управління в випадку втрати або порушення сигналу від оператора. Також слід зазначити і інший момент: при втраті зв'язку з оператором стає неможливою і передача розвідувальної інформації з відеокамер БПЛА. Надалі залишився без управління дрон може бути знищений боку, існує перехоплення, що насправді не є складним завданням. Або трофейний БПЛА може бути використаний для якихось інших потреб - його доля повністю в руках перехоплювача.

У деяких дронів передбачений варіант обриву зв'язку з оператором. В цьому випадку, якщо канал зв'язку втрачений, дрон переходить у відповідний режим роботи - автоматика перестає реагувати на всі сигнали ззовні і відповідно до заданої програми веде БПЛА до заздалегідь визначеного місця посадки, використовуючи систему GPS або ГЛОНАСС. Апарат використовує супутникову навігацію і визначає своє місце розташування, напрямок руху, відстань до оператора або точки посадки, щоб мати можливість повернутися на базу.

Щоб не допустити «евакуацію» дрона, засоби радіоелектронної боротьби повинні пригнічувати не тільки канал управління, але і сигнали навігаційної системи. В результаті успішного «глушіння» всіх цих сигналів противник, з високою ймовірністю, позбудеться техніки, що потрапила в зону дії РЕБ.

Варто виділити зростаючий спектр засобів мобільних РЕБ, які часом називають "кібер-гвинтівками". І не дивлячись на простоту і відносну дешевизну в порівнянні зі станціями РЕБ у них є досить істотний недолік - вони використовують можливість передачі сигналів на частоті каналу керування безпілотника. Так можна вивести з ладу лише деякі моделі дронів, а не будь-

який існуючий апарат. Автономним безпілотникам, які не отримують будь-які сигнали ззовні, така система не загрожує.

Системи перехоплення управління безпілотних літальних апаратів зазвичай доповнюють системи РЕБ або є самостійними комплексами, розгорнутими на певних областях.

Серед основних способів злому БПЛА можна перерахувати наступні:

1. Злом шифрованого каналу або підміна даних авторизації і отримання за рахунок цього доступу до управління дроном.
2. Використання вразливостей програмного забезпечення, в тому числі переповнення буфера.
3. Використання інтерфейсів і каналів даних оригінального програмного забезпечення для "протягування" стороннього коду.

Дорогі БПЛА, які використовуються поліцією або іншими державними структурами, службами МНС і окремими компаніями в приватному секторі, досить просто зламати і викрасти.

Для зв'язку по Wi-Fi між модулем контролю безпілотного апарату і пристроєм управління як правило використовується дуже слабке шифрування, так як відомо, що WEP (Wired Equivalent Privacy) можна зламати за кілька секунд. Причому атакуючий може досить просто потрапити в з'єднання між дроном і оператором, перебуваючи на відстані близько ста метрів, і послати БПЛА неправдиву команду або відключити його від вихідної мережі. [1]

Чіп Xbee, який використовується багатьма моделями дронів, небезпечний. Незважаючи на те, що Xbee підтримує шифрування, через проблеми з продуктивністю і для виключення затримок між командами оператора і реакцією БПЛА, воно просто відключено. Внаслідок чого зловмисник має можливість здійснити атаку man-in-the-middle, перебуваючи на відстані двох кілометрів від дрона. Атакуючий може перенаправити пакети,

заблокувати справжнього оператора, або пропускати всі пакети через себе, але більшість атакуючих воліють викрасти дрон.

1.3 Висновки

У цьому розділі було розглянуто загальні відомості про сучасний стан сфери БПЛА, більш конкретно розглянуто принципи систем зв'язку з БПЛА та способи їх злому. Проведений аналіз показав, що існують випадки злому зв'язку на поліцейських і воєнних БПЛА, а цивільні дрони є не захищеними взагалі.

На основі проведеного аналізу, було прийнято рішення розробляти окремий модуль зв'язку, що може бути встановлено в будь-який корпус БПЛА. Модуль зв'язку повинен забезпечити захист від РЕБ, підміни пакетів, GPS спуфінгу та перехоплення даних.

2 АНАЛІЗ ДОСТУПНИХ МЕТОДІВ РЕАЛІЗАЦІЇ КОМПОНЕНТ ЗВ’ЯЗКУ

2.1 Декомпозиція досліджуваної системи

Для зручності проведення аналізу і порівнянь необхідно розділити досліджувану систему, а саме систему захищеного зв’язку з БПЛА, на підсистеми. Основними компонентами системи зв’язку, що мають найбільший інтерес у процесі дослідження і реалізації є:

1. Система передачі відео потоку з БПЛА – включає в себе програмні та апаратні складові для створення, кодування, обробки і передачі відео потоку та інших мультимедійних складових.
2. Основна лінія зв’язку – включає в себе програмні та апаратні складові для реалізації каналів управління і телеметрії, а також методи і стандарти радіоліній зв’язку.
3. Захист лінії зв’язку – включає в себе алгоритми шифрування, алгоритми протидії РЕБ, GPS-спуфінгу та підміні даних, що передаються на БПЛА.

Всі три компоненти тісно пов’язані одна з одною, але мають свої основні особливості, тому, у цьому і подальших розділах, будемо розглядати їх як окремі частини системи додатково вказуючи їх зв’язки з іншими компонентами якщо це важливо.

2.2 Апаратні складові

2.2.1 Система передачі відео

FPV камера - це одна з найбільш важливих частин коптера. Не важливо на скільки хороший відеопередавач, якість картинки в окулярах або на моніторі обмежена саме FPV камерою.

CCD і CMOS - тип сенсора

CCD і CMOS - це два різних види матриць, які використовуються в камерах, кожен з них має свої унікальні характеристики і переваги. Безліч HD камер використовують CMOS матриці. Для цілей FPV кілька років тому найкраще підходили камери на CCD матрицях, зараз це не зовсім так. Коротке порівняння CCD і CMOS:

CCD

- Менше проявляється ефект «желе»
- Менше шуму при поганому освітленні
- Краще управління експозицією

CMOS

- Вища роздільна здатність
- кращі кольори
- Вища частота кадрів
- Споживають менше енергії

У будь якому випадку, камеру по цим критеріям краще обирати методом тестувань - подивитися як вона відпрацьовує при яскравому денному освітленні, при слабкому світлі, чи сліпне якщо її спрямувати на сонце, оцінити ширину динамічного діапазону, а також проаналізувати основні характеристики - затримка відео і якість картинки.

NTSC і PAL - стандарти кодування відео

Насправді немає великої різниці який стандарт використовувати - NTSC або PAL, тому що вони обидва підтримуються більшістю обладнання для FPV. NTSC використовується в північній Америці, Японії і Південної Кореї. PAL використовується в більшій частині Європи, Австралії, і значній частині Африки і Азії. Непогано вибрати стандарт який відповідає вашій країні, проте це не є обов’язковим.

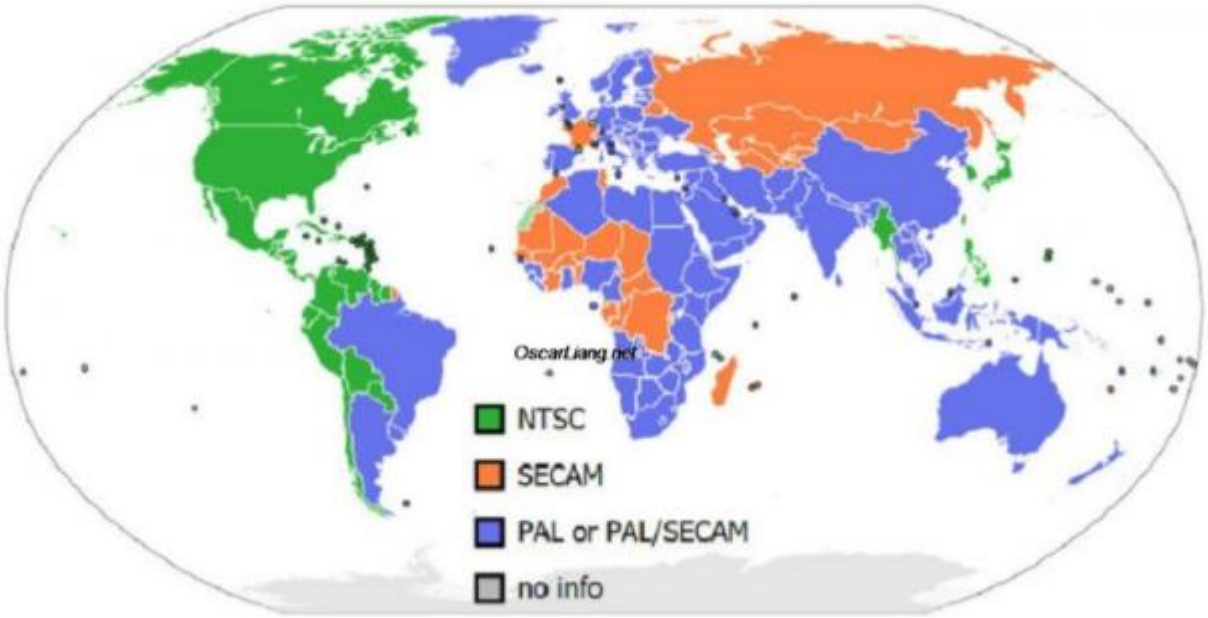

Рисунок 2.1 - PAL-NTSC map country [12]

Основна різниця в тому, що PAL забезпечує велику роздільну здатність, в той час як NTSC – більшу частоту кадрів. Отже, якщо потрібно відмінне зображення, то краще використовувати PAL, а для більш плавного відео краще вибрати NTSC.

PAL: 720 x 576 @ 25fps

NTSC: 720 x 480 @ 30fps

Поле зору - FOV - фокусна відстань об'єктива

Зазвичай є кілька варіантів з різними об'єктивами, з різною фокусною відстанню і відповідно різним полем зору (FOV). Найбільш часто

використовуються фокусні відстані 3.6мм і 2.8мм. Наприклад, лінза з фокусом 3.6мм дасть поле зору в 90 градусів, що добре підходить для FPV. Лінза на 2.8мм дасть ще більший кут - 112 градусів.

Важливо вибрати відповідну фокусна відстань під конкретні задачі, що будуть виконуватися БПЛА. Менша фокусна відстань означає більш широкий кут. Це не завжди добре - чим ширше кут тим більше помітний ефект риб'ячого ока. Об'єкти по середині кадру здаються менше, а чим ближче до краю, тим більше картинка викривлена. Якщо фокусна відстань велика, то картинка є наближеною. Найбільш розповсюдженими є 2.8мм і 3.6мм (90-100 градусів) лінзи.

Роздільна здатність камери - TVL - телевізійні лінії

TVL – це кількість чорних і білих ліній, які можуть бути відображені на зображенні горизонтально. Камера 600TVL може відобразити 300 чорних і 300 білих ліній по черзі, в одному кадрі. Найбільш часто зустрічаються камери що мають 380, 480, 540, 600, 700, 800, 1200 TVL.

Звичайно, більша роздільна здатність йде разом із збільшеною ціною і затримкою відео (через обробку відео). 600TVL найбільш популярний варіант в останні роки. Крім того, не завжди «більше означає краще», через обмеження що накладаються аналоговим відео передавачем: 5.8ГГц - ліміт на обсяг переданих даних, так що якість може бути обмеженою.

Розміри камери, вага і робоча напруга

Наступна характеристика про яку варто подумати - це розмір і робоча напруга. В наш час камери FPV стандартизовані і більшість зроблено у вигляді квадрата зі стороною 32 або 38мм або 26 на 26 всередині корпусу. Розміри визначають складність кріплення до рами.

Камери можуть важити 20-50 грамм (без корпусу), але існують і міні варіанти камер вагою всього 5 грамм.

Підключення камер, як правило, дуже просте, камери мають 3 або 4 дроти. Червоний - це «плюс» живлення, чорний - «земля», жовтий - відео сигнал. Іноді є додатковий дріт для звуку (якщо є вбудований мікрофон).

Більшість камер що є у продажу мають широкий діапазон вхідної напруги, типу 5-17В. Це дозволяє живити камеру як через стабілізатор, так і безпосередньо від LiPo акумулятора (2S-4S). Ще я вважаю за краще ставити LC-фільтр, для очищення живлення камери і відео передавача від перешкод і отримання максимально якісної картинки.

ІК чутливі камери і ІК блокуючі

Деякі камери пропонуються в двох варіантах, одні з IR block (ІК блокуванням) та інші IR Sensitive (ІК чутливі). Перші дають кращі кольори, другі краще працюють при поганому освітленні. Вибір базується на аналізі задач, які поставлені перед конкретним БПЛА.

Затримка

Затримка камери може бути вирішальним фактором, при русі БПЛА з перешкодами. Незважаючи на те, що це менш важливо, якщо ви летите у вільній зоні, затримка все одно буде впливати на ваш польот. Для прикладу, якщо дрон летить зі швидкістю 100 км/г, затримка 50 мс означає, що дрон пролетить 1.4 м, перш ніж пілот це побачить і зможе зреагувати.

Затримка камери часто пов’язана з TVL, і обробкою зображення. Більшість камер CCD 600TVL, таких як Runcam Swift і Foxeer HS1177 мають затримку менше 20мс. Камери з більш високим TVL мають більшу затримку через додаткову кількість даних, що треба обробити.

Затримка не зазначена у специфікації камери, і перевіряється онлайн рецензентами у більшості випадків, так що необхідно проводити дослідження і з'ясувати цей критерій.

2.2.2 Основна лінія зв’язку

У разі малих БПЛА (злітна маса до 5 кг) внаслідок обмежень за габаритами і масою приймально-передавального обладнання раціональним є використання єдиного радіоканалу зв'язку для передачі командно-телеметричних даних і даних корисного навантаження. Посадка таких ЛА здійснюється, як правило, за допомогою парашута, що не вимагає додаткового радіоканалу зв'язку для передачі зображення з відеокамер ЛА, необхідного при ручній посадці. Додатковим радіоканалом зв'язку є тільки лінія передачі даних САП. Для задоволення вимог по пропускній здатності каналу зв'язку при передачі як даних телеметрії, так і даних корисного навантаження, необхідно розширювати смугу частот приймально-передавального обладнання і використовувати спектрально-ефективні методи модуляції, що призводить до підвищених вимог по відношенню сигнал / шум на вході приймача, зниження дальності дії радіосистеми, підвищення ймовірності бітової помилки і т. д.

Таким чином, додатковий зв'язок обладнання першої і другої груп призводить до погіршення робочих характеристик пристроїв першої групи. Високий ступінь інтеграції пристроїв двох груп призведе до зменшення значення ймовірності безвідмовної роботи життєво важливих елементів комплексу. Виходячи з цього, на комплексах БПЛА із злітною масою більше 5 кг доцільним є використання окремих радіоліній зв'язку для передачі командно-телеметричних даних і даних корисного навантаження. При цьому на перший план виходять питання електромагнітної сумісності приймально-передавального обладнання, частотного поділу каналів зв'язку і розміщення антенно-фідерного обладнання на борту БПЛА.

Вибір робочого частотного діапазону радіоканалу зв'язку обумовлюється декількома факторами:

- вимогами до маси, габаритів і споживання приймально-передавального пристрою БПЛА;
- необхідної дальністю роботи при заданій ймовірності бітової помилки;
- можливістю отримання ліцензії на роботу в необхідному діапазоні або можливістю безліцензійної роботи.

Для систем зв'язку малих БПЛА вирішальними факторами при виборі частотного діапазону є маса і габарити бортового приймача і антенно-фідерного пристрою. Доцільним є вибір діапазону надвисоких частот, при цьому можна використовувати антену малих розмірів, здатну розміститися в профілі крила. Щільна компоновка обладнання всередині малого БПЛА не дозволяє ефективно використовувати приймачі великої потужності з укороченими антенами ультракороткохвильової діапазону внаслідок проблем з електромагнітною сумісністю і великим впливом навколишніх об'єктів на характеристики антени. Одним з відповідних частотних діапазонів є діапазон 2,4 ГГц.

До систем зв'язку БПЛА середнього і великого класу пред'являються більш жорсткі вимоги по дальності роботи, стійкості до заглушення і ймовірності бітової помилки. В цьому випадку є можливим і оптимальним комплексування декількох каналів зв'язку, що працюють в різних частотних діапазонах.

2.2.3 Захист лінії зв’язку

Основним апаратним компонентом для реалізації шифрування є процесор. Очевидно, що в умовах обмежень на вагу та форм-фактор які накладає на комп’ютер необхідність його вбудови у БПЛА, доцільним є використання одно платних комп'ютерів.

Таблиця 2.1 - Порівняння мікрокомп'ютерів

Найменування	CPU	RAM	Порти	Інші корисні особливості	Ціна (грн)
Raspberry Pi 3	Quad Core 1.2GHz Broadcom BCM2837 64bit	1GB	4 USB; 40 GPIO; Full HDMI порт; Ethernet порт; CSI; DSI	VideoCore IV 3D; 802.11n Wireless LAN; Bluetooth 4.1	1370
Banana Pi M2 Ultra	Allwinner R40 ARM Cortex A7 Quad core	2GB DDR3	40 GPIO; два USB 2.0 host і один USB 2.0 OTG; HDMI порт; Ethernet порт; CSI; DSI	GPU: Mali 400 MP2 Dual Core; 802.11n Wireless LAN; Bluetooth 4.1	1636
ODROID-XU4	Samsung Exynos5422 ARM Cortex 2GHz/1.4GHz	2GB LPDDR3	42 GPIO; два USB 3.0 і один USB 2.0 OTG; HDMI порт; Ethernet порт;	GPU: Mali-T628 MP6; Роз'єм для підключення батареї резервного живлення для RTC	2272
LATTEPANDA 4GB/64GB	Intel Cherry Trail Z8300 Quad Core 1.8GHz	4GB DDR3L	USB 2.0 x 2, USB 3.0 x 1; HDMI порт; Ethernet порт; GPIO для Intel X-Z8300 і ATmega32u4	GPU: Intel HD Graphics; 802.11n Wireless LAN; Bluetooth 4.1	4640

Також, треба зазначити, що компресія відео потоку також сильно навантажує процесор (у більшості випадків не менше ніж алгоритми шифрування), тому при виборі комп’ютера це необхідно зауважити.

Окрім потужного процесора від мікрокомп'ютера також потребується велика кількість портів для підключення модулів, таких як камера, радіо модем, порт для телеметрії, порт для RTC, порт для передачі сигналів керування та інші.

У таблиці наведено основні характеристики доступних у продажу мікрокомп'ютерів відомих брендів.

2.3 Технологічні та алгоритмічні рішення

2.3.1 Лінія зв’язку

У першому поколінні радіоліній зв'язку з БПЛА збройних сил країн НАТО використовувалася існуючі інфраструктури комунікацій (наприклад, JTIDS / Link 16). Зокрема, такої концепції дотримуються і розробники автоматизованої системи протидії терористичним загрозам в гаванях LEXXWAR, демонструвалися на виставці TechDemo'08. Однак недостатня пропускна здатність Link 16 - до 50 Кбіт/с - не дозволяє повною мірою реалізувати потенціал БПЛА. Тому сьогодні ведуться численні розробки радіозасобів для зв'язку з БПЛА, причому вони характеризуються різноманіттям підходів.

Певну частку міжнародного ринку займають системи з традиційними, перевіреними протягом багатьох років методами модуляції сигналів. Характерним прикладом є аналоговий канал передачі відеоданих з борта німецького БПЛА "Місяць" з полосою пропускання 5 МГц, за яким також транслюються зображення місцевості отримані з бортовою РЛС. Інший приклад використання застарілих, з точки зору STANAG 4609, аналогових методів

зв'язку з традиційною частотною модуляцією сигналів GMSK - розроблений компанією Enerdyne програмований модем для тактичних систем БПЛА EnerLinksIII. У режимі прямої видимості він передає відеодані NTSC, PAL і RS170 в частотних діапазонах 1700-1850 МГц (L-діапазон), 2200-2500 МГц (S-діапазон), 4400-4950 МГц (нижній С-діапазон) і 5250-5850 МГц (верхній С-діапазон). Кожен з них може використовуватися для висхідного і низхідного каналів. При цьому досягається швидкість передачі даних 11 Мбіт/с на відстані 75 морських миль і 5 Мбіт/с - на відстані до 100 морських миль. У типовому складі наземного обладнання передбачена дзеркальна параболічна антена діаметром 24 дюйми з автоматичним супроводом БПЛА в межах зони прямої видимості.

Зазначені публікації [3] вимоги до пропускної здатності змушують розробників радіоліній БПЛА шукати нові підходи до підвищення швидкості передачі даних від мультисенсорних бортових платформ. Один з найбільш ефективних підходів - застосування модуляції OFDM, і C-OFDM. Серед перших проектів, в яких досліджувалася можливість застосування OFDM-модуляції на лінії зв'язку з БПЛА, - проект Minuteman, який фінансувався відділом перспективних досліджень ВМФ США (Управління військово-морських досліджень - ОНР). Він реалізовувався в 2000-2005 роки в лабораторіях відділів електротехніки та комп'ютерних наук Каліфорнійського університету в Лос-Анджелесі (UCLA) [4]. Метою проекту була розробка системи радіозв'язку та обміну даними сил флоту з безпілотними повітряними, надводними і наземними апаратами.

Застосування OFDM-сигналів в радіолініях "БПЛА - наземні абоненти" передбачається в проекті Інституту електроніки і зв'язку Української академії наук зі створення системи передачі даних на базі висотного БПЛА (СПД «Фаетон"). [5] При передачі даних в висхідних каналах (з борта на землю) в цьому проекті пропонується використовувати стандарт DVB-S з модуляцією OFDM-256, а в низхідних - методи множинного доступу з частотним і

тимчасовим поділом частот. Діапазон OFDM-сигналів стандарту DVB-S складає 11,7-12,5 ГГц, смуга одного радіоканалу за рівнем -30 дБ досягає 40 МГц. Граничний радіус зони обслуговування однієї центральної станції в умовах прямої видимості при потужності передавача БПЛА 50 мВт і інтенсивності опадів до 40 мм/год заявлений в межах 50-60 км. За рахунок збільшення потужності бортового передавача радіус зони покриття може бути збільшений до 250 км.

Постійно розширюється і військове напрямок застосування OFDM-модуляції. У сухопутних військах НАТО з'явилися система зв'язку, що використовує військова версія стандарт IEEE 802.11g, їх виробництво освоїли нідерландську фірма MobiComm. Компанія Nova Engineering вже кілька років пропонують серійні комплекти зв'язку для ВМС США (HDR LOS радиомодем), які реалізують принцип OFDM.

Широке поширення OFDM сприяло вибору даної технології модуляція сигналів в якості фізичної основи створення тактичних широкосмугових мереж (Wideband Networking Waveform) в рамках програми Joint Tactical Radio System (JTRS). Як зазначено в перспективному плані розвитку безпілотних авіаційних систем США [6], WNW планується використовувати в якості радіоліній зв'язку з БПЛА, наприклад, в частотному діапазоні 225-400 МГц. При цьому очікується досягнення швидкості передачі даних 10 Мбіт/с. При міграції на інші частоти в залежності від смуги пропускання каналу зв'язку швидкість передачі може бути збільшена. Наприклад, WNW-модем SDR-4000 компанії L-3 Communications Нова Інжиніринг при ширині смуги 10 МГц забезпечує швидкість передачі до 23 Мбіт/с.

Для одночасного зв'язку з декількома БПЛА в найпростішому випадку використовуються кодовані OFDM-сигнали. Наприклад, фірма Cobham Surveillance просуває систему зв'язку на основі сигналів DVB-T з модуляцією C-OFDM, і шестигранною антеною решітки. Система функціонує в діапазоні

1,7-1,85 і 1,99-2,5 ГГц. Її приймач дозволяє забезпечити зв'язок з мобільними джерелами сигналів.

2.3.2 Аналіз алгоритмів синхронізації часу

Синхронізація часу між наземним і повітряним модулями БПЛА є необхідною складовою для функціонування інших модулів і коректного логування. Існує декілька стандартних підходів синхронізації годинників що мають свої переваги, недоліки та особливості. Ключем до побудови якісного алгоритму синхронізації є аналіз основних можливостей лінії зв'язку між відокремленими системами. Також треба зауважити які вимоги на точність і швидкість синхронізації накладають компоненти систем що будуть використовувати синхронний годинник.

Будь-який комп'ютер має механізм підрахунку часу. Хоча його зазвичай називають «годинник», це не зовсім вірно - це, скоріше, таймер. Таймер реалізований наступним чином: є кристал кварцу; перебуваючи під напругою, він коливається з певною частотою, яка залежить від властивостей конкретного кристала і напруги, що подається. Кожне коливання викликає зміну лічильника, відповідального за формування системного часу. Таким чином, через різну частоту коливання кристала має місце рассинхронізация часу в розподілених системах.

Розглянемо основні методи синхронізації часу в розподілених системах, відразу сформулювавши дві основні вимоги:

- в певні моменти системний час вузлів системи повинен максимально збігатися;
- при синхронізації неприпустимо переведення годинників в зворотну сторону (зменшення системного часу), оскільки це може привести до виходу за початок інтервалу синхронізації або до порушення роботи механізму визначення черговості подій;

Таким чином, ми можемо оперувати лише переведенням часу вперед або уповільненням його ходу. Якщо в розподіленої системі присутній вузол, який має зовнішні фізичні годинники або приймач сигналів точного часу, то завдання синхронізації зводиться до необхідності синхронізації інших вузлів системи з даним еталонним.

Алгоритм Крістіана

Періодично кожен вузол посилає еталонному запит його поточного часу. Еталонний вузол максимально швидко відповідає на цей запит. Але слід пам'ятати, що між відправкою повідомлення еталоном і його отриманням вузлом проходить деякий час, який не є постійним і залежить від поточного завантаження мережі передачі даних. Для оцінки цього часу Крістіан запропонував приймати час передачі відповіді як половину часу між відправленням запиту і одержанням відповіді. Для підвищення точності слід виконати серію таких замірів.

Переваги: простота реалізації, висока ефективність в невеликих мережах і мережах з малим завантаженням: оскільки в системі присутнє джерело точного часу і синхронізація проводиться по ньому, то час в системі в цілому відповідає реальному.

Недоліки: вимагає зовнішнього джерела точного часу, не має вбудованої захисту від переведення годинників в зворотну сторону, некоректно працює в мережах з різкими стрибками завантаження мережі, не дозволяє коректно встановлювати час в разі, якщо запит і відповідь передаються по різних маршрутах (потрібен різний час для передачі даних повідомлень).

Алгоритм Берклі

Даний алгоритм призначений для випадку, коли джерело точного часу відсутнє. Сервер часу опитує всі вузли для з'ясування їх поточного часу, усереднює це значення і розсилає команди на установку нового значення часу або уповільнення годинників.

Переваги: простота реалізації, висока ефективність для систем, некритичних до відхилень за часом.

Недоліки: потрібно виділити вузол - сервер часу, оскільки джерело точного часу відсутнє, а за загальносистемний час приймається усереднений час вузлів, він може сильно відрізнятися від реального часу, вузли з уповільненим для корекції часом (все ще некоректним) впливають на загальносистемне час.

Усереднюючі алгоритми

Сімейство алгоритмів, що мають загальний принцип роботи. З заданою періодичністю усі вузли системи генерують широкомовну розсилку поточного часу. Потім протягом певного часу приймають повідомлення про поточний час від інших вузлів. Коли всі повідомлення прийняті, запускається алгоритм обчислення поточного часу. Найпростіший варіант - усереднення отриманих значень. Алгоритм може бути удосконалений шляхом відкидання m найбільших і m найменших значень для захисту від завідомо неправдивих значень. Інший шлях удосконалення алгоритму - спроба оцінки часу проходження повідомлення від джерела (можливо з урахуванням інформіції про топологію мережі) для отримання більш точних значень.

Переваги: не потрібен виділений вузол - сервер часу, висока ефективність підстроювання під конкретні завдання.

Недоліки: сильна залежність ефективності від завантаження мережі, низький ступінь синхронізації, не забезпечують установки на всіх вузлах однакового часу (частина повідомлень на конкретному вузлі може не бути отримана, чи не отримана вчасно і відкинута як помилкова).

Відмітки часу Лампорта

При обміні даними між взаємодіючими вузлами відбувається також обмін інформацією про їх локальний час. Якщо позначка часу, зазначена в повідомленні, виявляється менше, ніж локальний час вузла, така ситуація вважається штатною, так як повідомлення було відправлено раніше, ніж

отримано. Якщо ж в повідомленні зазначено більш ранній час, ніж локальний час вузла, тобто повідомлення було відправлено пізніше, ніж отримано, робиться висновок про необхідність корекції системного часу вузла. Час виставляється в значення рівне позначці в повідомленні плюс одиниця.

Переваги: не потрібен виділений вузол - сервер часу, висока ефективність при малій кількості вузлів, високий ступінь синхронізації (час переводиться тільки вперед, і немає необхідності чекати ефекту від уповільнення на поспішаючих вузлах).

Недоліки: не приймається в розрахунок час передачі повідомлень між вузлами, оскільки має місце постійне переведення годинників вперед - загальносистемний час сильно відрізняється від реального.

Протокол NTP

Network Time Protocol (NTP) - мережевий протокол для синхронізації внутрішнього годинника комп'ютера з використанням мереж зі змінною латентністю.

NTP використовує для своєї роботи протокол UDP і порт 123.

Поточна версія протоколу - NTP 4. NTP використовує ієрархічну систему «часових рівнів» (їх так само називають Stratum (рис. 4)). Рівень 0 (або Stratum 0) - це, як правило, пристрої, що представляють собою атомний годинник (молекулярні, квантові), GPS годинник або радіо-годинник. Дані пристрої зазвичай не публікуються у всесвітню мережу, а підключаються безпосередньо до серверів часу рівня 1 за допомогою протоколу RS-232. Рівень 1 синхронізований з високоточними годинниками рівня 0, зазвичай працюють в якості джерел для серверів рівня 2. Рівень 2 синхронізується з однією з машин рівня 1, також можлива синхронізація з серверами свого рівня. Рівень 3 працює аналогічно другому. Зазвичай в мережу публікуються сервера рівнів від другого і нижче. Протокол NTP підтримує до 256 рівнів. Також, треба зазначити, що сервера рівнів 1 і 2, а іноді і 3 не завжди відкриті для загального

доступу. Іноді, щоб синхронізуватися з ними, необхідно надіслати запит поштою - адміністраторам домену.

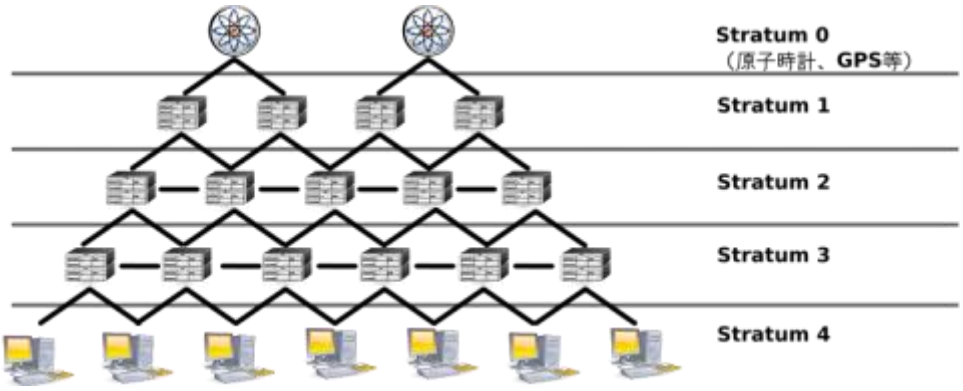

Рисунок 2.2 - Схема рівнів NTP [17]

Для чого робиться обмеження на доступ до серверів? З переходом на кожен рівень трохи зростає похибка щодо первинного сервера, але збільшується загальна кількість серверів і, отже, зменшується їх завантаження.

Призначення сервера NTP в локальній мережі

Існують служби в операційних системах, які можуть залежати від синхронізованого часу. Найбільш яскравим прикладом таких служб є протокол аутентифікації Kerberos. Для його роботи необхідно, щоб на комп'ютерах, доступ до яких здійснюється з використанням цього протоколу, системний час відрізнялося не більше ніж на 5 хвилин. Крім того, точний час на всіх комп'ютерах значно полегшує аналіз журналів безпеки при розслідуванні інцидентів в локальній мережі.

Режими роботи NTP сервера / клієнта

Клієнт / сервер

Цей режим на сьогоднішній день найбільш часто використовується в мережі Інтернет. Схема роботи - класична. Клієнт посилає запит, на який протягом деякого часу сервер надсилає відповідь. Налаштування клієнта проводиться за допомогою директиви server в файлі конфігурації, де вказується DNS ім'я сервера часу.

Симетричний активний / пасивний режим

Цей режим використовується в тому випадку, якщо проводиться синхронізація часу між великою кількістю рівноправних машин. Крім того, що кожна машина синхронізується із зовнішнім джерелом, вона також здійснює синхронізацію зі своїми сусідами (peer), виступаючи для них в якості клієнта і сервера часу. Тому навіть якщо машина «втратить» зовнішнє джерело, вона все ще зможе отримати точний час від своїх сусідів. Сусіди можуть працювати в двох режимах - активному і пасивному. Працюючи в активному режимі, машина сама передає свій час всім машинам-сусідам, перерахованим в секції peers конфігураційного файлу ntp.conf. Якщо ж в цій секції сусіди не вказані, то вважається, що машина працює в пасивному режимі. Для того щоб зловмисник не зміг скомпрометувати інші машини, прикидаючись активним джерелом, необхідно використовувати аутентифікацію.

Режим Broadcast

Цей режим рекомендується використовувати в тих випадках, коли мала кількість серверів обслуговує велику кількість клієнтів. Працюючи в цьому режимі, сервер періодично розсилає пакети, використовуючи широкомовну адресу підмережі. Клієнт, налаштований на синхронізацію таким способом, отримує широкомовний пакет сервера і виробляє синхронізацію з сервером. Особливістю цього режиму є те, що час доставляється в рамках однієї підмережі (обмеження broadcast-пакетів). Крім того, для захисту від зловмисників необхідно використовувати аутентифікацію.

Режим Multicast

Цей режим багато в чому схожий на broadcast. Відмінність полягає в тому, що для доставки пакетів використовуються multicast-адреси мереж класу D адресного простору IP-адрес. Для клієнтів і серверів задається адреса multicast-групи, яку вони використовують для синхронізації часу. Це робить можливим синхронізацію груп машин, розташованих в різних підмережах.

Режим Manycast

Цей режим є нововведенням четвертої версії протоколу NTP. Він бере за основу пошук клієнтом серед своїх мережевих сусідів manycast-серверів, отримання від кожного з них зразків часу (з використанням криптографії) і вибір на підставі цих даних трьох «кращих» manycast-серверів, з якими клієнт буде робити синхронізацію. У разі виходу з ладу одного з серверів клієнт автоматично оновлює свій список.

2.3.3 Аналіз алгоритмів шифрування

На сьогоднішній день в сфері ЕБ широко представлені системи як з симетричним шифруванням, так і з асиметричним. Кожен із підходів має свої переваги і недоліки.

Основний недолік симетричного шифрування полягає в необхідності публічної передачі ключів - "з рук в руки". На цей недолік не можна не звернути увагу, так як стає практично неможливим використання симетричного шифрування з необмеженою кількістю учасників. В іншому ж алгоритм симетричного шифрування можна вважати досить пророблениим і ефективним, з мінімальною кількістю недоліків, особливо на тлі асиметричного шифрування. Недоліки останнього не настільки значні, щоб говорити про те, що алгоритм чимось поганий.

Перший недолік асиметричного шифрування полягає в низькій швидкості виконання операцій шифрування і розшифрування, що обумовлено необхідністю обробки ресурсномістких операцій. Як наслідок, вимоги до апаратної складової такої системи часто бувають неприйнятні.

Додаткові проблеми виникають і при захисті відкритих ключів від підміни, адже досить просто підмінити відкритий ключ легального користувача, щоб згодом легко розшифрувати його своїм секретним ключем.

Очевидно, що недоліки симетричних алгоритмів легко нівелюються шляхом передачі ключів на модулі керування за допомогою локальної мережі,

зйомного накопичувача чи в будь який інший подібний спосіб, що є абсолютно безпечним.

Advanced Encryption Standard (AES), також відомий під назвою Rijndael — симетричний алгоритм блочного шифрування (розмір блока 128 біт, ключ 128/192/256 біт), фіналіст конкурсу AES і прийнятий як американський стандарт шифрування урядом США. Вибір припав на AES з розрахуванням на широке використання і активний аналіз алгоритму, як це було із його попередником, DES. Державний інститут стандартів і технологій США опублікував попередню специфікацію AES 26 жовтня 2001 року, після п'ятилітньої підготовки. 26 травня 2002 року AES оголошено стандартом шифрування. Станом на 2009 рік AES є одним із найпоширеніших алгоритмів симетричного шифрування.

Для трьох варіантів ключів AES повний перебір вимагає 2^{127}, 2^{191} або 2^{255} операцій відповідно. Навіть найменше з цих чисел свідчить, що атака з використанням перебору ключів сьогодні не має практичного значення.

Відповідно до оцінок розробників шифр стійкий проти таких видів крипто-аналітичних атак:

- диференціального крипто-аналізу;
- лінійного крипто-аналізу;
- крипто-аналіз на основі зв'язаних ключів (слабких ключів в алгоритмі немає).

У червні 2003 року АНБ США оголосило, що шифр AES є досить надійним для захисту відомостей, що становлять державну таємницю. Аж до рівня SECRET було дозволено використовувати ключі довжиною 128 біт, для рівня TOP SECRET - ключі довжиною 192 і 256 біт.

Єдиний працюючий спосіб злому шифру AES - це атаки по побічним каналам. Такі атаки не пов'язані з математичними особливостями AES, а використовують певні особливості реалізації систем, що використовують шифр, з метою розкрити частково або повністю секретні дані, в тому числі

ключ. Так, в квітні 2005 року Daniel J. Bernstein опублікував роботу з описом атаки на основі інформації про час виконання кожної операції шифрування. Дана атака потребувала понад 200 мільйонів обраних шифротекстів для знаходження ключа. Інша атака Даг Арне Освіка, Аді Шаміра і Еран Трумера, також заснована на тимчасовому аналізі виконання операцій, в жовтні 2005 року вже розкривала ключ всього лише за 800 операцій шифрування. Але атакуючий в цьому випадку повинен запускати програми на тій же системі, де виконувалося шифрування. У грудні 2009 року опублікована робота, в якій використання диференціального аналізу помилок дозволило відновити ключ за 2^{32} операцій (це різновид крипто-анализу на основі створення випадкових апаратних помилок).

Алгоритм має не тільки дуже високою захищеність, а й дуже високою швидкість шифрування. Програмна реалізація на машині з частотою 2 ГГц дозволяє шифрувати дані зі швидкістю 700 Мбіт/с. Такої швидкості достатньо для шифрування відео в форматі MPEG-2 в реальному часі. Апаратні реалізації працюють ще швидше. Останнім часом з'явилася нова версія AEC-NI (New Instructions), яка дозволяє оптимізувати роботу алгоритму (знизити завантаження процесора на 50%). Ця версія може використовуватися і спільно з SSL (SSL - криптографічний протокол, який забезпечує встановлення безпечного з'єднання між клієнтом і сервером). Компанія Intel розробила мікросхему, що реалізує цей алгоритм (серія X5600). Кількість клієнтів при роботі з версією AES-NI збільшується на 13%.

2.4 Висновки

У цьому розділі було розглянуто варіанти реалізації головних компонент системи зв'язку. Для зручності систему було розділено на складові частини для проведення більш детального аналізу. Було проведено детальний огляд доступних варіантів реалізації поставлених задач: алгоритмів шифрування,

стандартів передачі даних, опис головних характеристик камери БПЛА та алгоритмів синхронізації часу у розподілених системах. На основі цієї інформації в наступному розділі були обрані конкретні рішення для впровадження їх у прототипі.

3 РОЗРОБКА РОБОЧОГО ПРОТОТИПУ БПЛА З ЗАХИЩЕНИМ ІНТЕРФЕЙСОМ ЗВ’ЯЗКУ

3.1 Головний мікрокомп’ютер

У якості головного мікрокомп’ютера було обрано Raspberry Pi 3 Model B. Як показали попередні дослідження і практична реалізація прототипу, RPi 3 є достатньо потужною платформою для реалізації можливостей прототипу БПЛА.

Основними критеріями при виборі мікрокомп’ютера були наявність великої кількості портів, достатня потужність для виконання шифрування потоку і інших, менш затратних процесів одночасно, а корисними особливостями вважалися вбудований WiFi модуль і наявність GPU. RPi 3 повністю відповідає критеріям вибору: кількості портів достатньо для підключення всіх необхідних апаратних компонент (DVB-T модем, video capture device, порт телеметрії та порт керування для пілотного контролера, RTC, GPIO індикатор та інші ситуативні), процесор справляється з навантаженням від всіх процесів у режимі повного функціонування, наявні вбудований WiFi модуль і VideoCore IV 3D.

Також, треба зазначити, що популярність, яку здобув RPi, значно прискорює процес розробки програмного забезпечення - доступні зручні збірки ОС, бібліотек та інших програмних засобів. Так, підготовка повністю робочої платформи на базі ОС Raspbian Jessie (Debian Jessie сконфігурований під RPi) з нуля, включаючи всі необхідні бібліотеки і програмні компоненти займає не більше двох годин.

3.2 Відео система

Для прототипу було обрано звичайну бюджетну FPV камеру 1/3-inch Sony super HAD color CCD з максимальною розподільною здатністю 752 x 582 (PAL) що є цілком достатньо зважаючи на те, що обраний video capture device має обмеження розподільної здатності 720 x 576.

Відео потік генерується за допомогою фреймворка GStreamer. GStreamer – дуже потужний фреймворк, що базується на пайплайнах, які в свою чергу, складаються із плагінів. Досить велика кількість плагінів забезпечує варіативність варіантів побудови потоку та дозволяє підлаштовувати кінцевий результат використовуючи різні додаткові можливості кастомізації. Крім того, вихідний код плагінів є відкритим, тому можна модифікувати плагіни під конкретні рішення, якщо базова версія не має необхідного функціоналу.

Ще однією вагомою перевагою фреймворка GStreamer є наявний для неї плагін OpenMax (OMX), що має конфігурацію під RPi 3 та бездоганно використовує зазначений вище VideoCore IV 3D. Порівнюючи у htop завантаженість процесора під час використання стандартного плагіну для кодування відео у формат H.264 [рис. 3.1] і плагіну OMX [рис. 3.2], легко зробити висновок, що в першому варіанті використання відео у форматі 576i є неможливим, в той час як при другому підході звільнюється значна частина головного процесору за рахунок використання VideoCore IV 3D. Маємо середню завантаженість процесора при використані x264 енкодера близько 300%, а при використанні плагіну OMX – 80%.

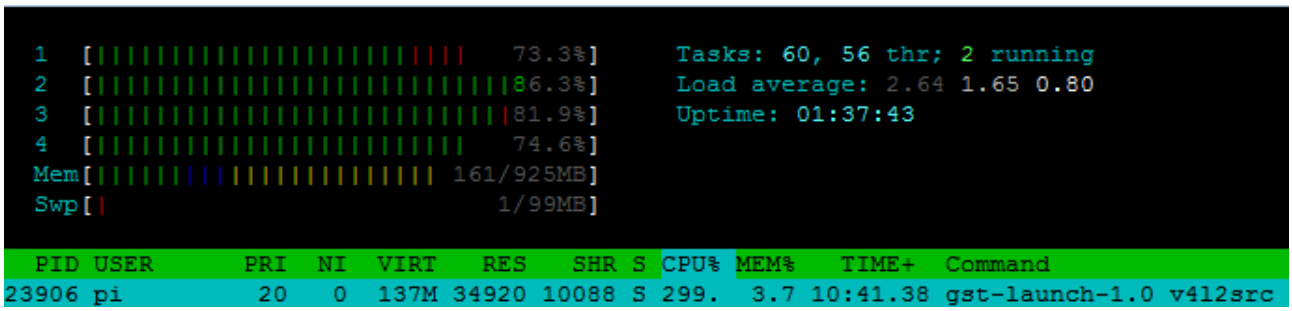

Рисунок 3.1 - Завантаженість процесора при використанні плагіну x264

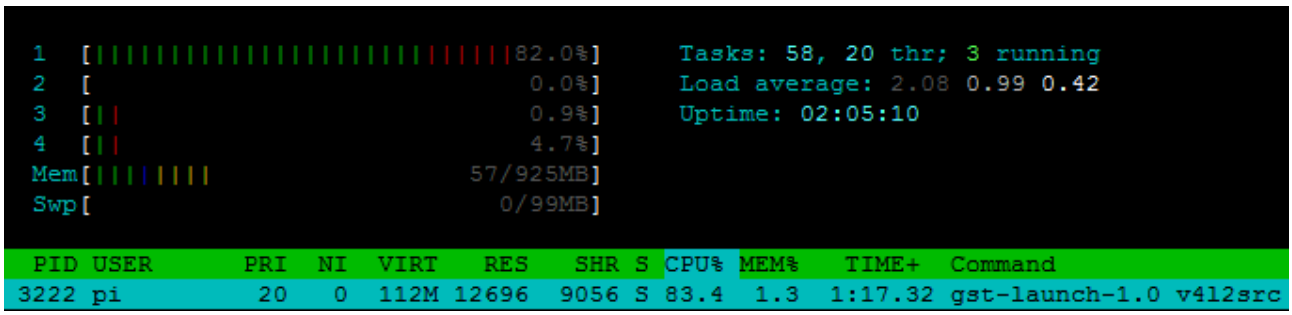

Рисунок 3.2 - Завантаженість процесора при використанні плагіну OMX

3.3 Лінія зв’язку

Для передачі даних між наземним і повітряним модулями було обрано європейський стандарт наземного цифрового мовлення - DVB-T. DVB-T призначений для передачі єдиного транспортного потоку MPEG-TS з цифровими сервісами (мультиплексу), використовуючи модуляцію COFDM, зі швидкістю до 31 Мбіт/с.

Було обрано компактний повно дуплексний DVB-T передавач, параметри якого задовольняють умовам використання лінії зв’язку (необхідно мати можливість рознести частоти приймачів на наземному і повітряному модулі БПЛА, а також мати значну кількість частот, на які можна налаштувати приймачі, для реалізації алгоритму протидії РЕБ).

Таблиця 3.1 - Характеристики DVB-T модему

Параметр	**Значення**	
Полоса пропускання	Передавач	2/3/4/5/6/7/8 MHz
	Приймач	5/6/7/8 MHz
Frequency range	Передавач	50~950 MHz та 1200~1350 MHz з кроком 1KHz
	Приймач	50~950 MHz з кроком 1KHz
Вихідний рівень RF	0 dBm (108 dBuV)	
Цифрове підсилення	Діапазон: +6/-25 dB , з кроком 1 dB	

Програмна частина лінії зв’язку є основною і найбільш складною складовою проекту: вона включає в себе мультиплексування/ демультиплексування потоку, фільтрацію пакетів, інтерфейси для взаємодії із драйвером радіо передавача/приймача, збору статистики та інших компонент.

За допомогою бібліотеки MavLink виконується обмін повідомленнями і сигналами керування між польотним контролером та наземною станцією керування. Всі дані включаючи сигнали керування, телеметрії, відео потік та інші інформаційні об’єднуються у єдиний MPEG-TS потік, також до пакетів прикріплюються додаткові дані для контролю над пакетами, що забезпечує більший захист інформації.

Локальні повідомлення в межах кожного з модулів передаються за допомогою бібліотеки LCM, що забезпечує дуже швидкий обмін даними і, як наслідок малі затримки.

3.4 Захист лінії зв’язку

Оскільки алгоритм AES є стандартом шифрування у наш час, для реалізації шифрування даних у проекті використовується вже реалізований алгоритм AES-128 із однієї з багатьох бібліотек шифрування. Реалізація алгоритмів протидії РЕБ і захисту від підміни пакетів є однією з головних ідей проекту, тому є строго конфіденційними і не можуть бути описані в цій роботі.

Проте, значну частину досліджень становив аналіз і розробка алгоритму синхронізації часу у розподіленій системі, а саме між наземним та повітряним модулем зв’язку з БПЛА. Синхронний годинник на даний момент необхідно використовувати у двох основних напрямах:

1. Синхронне логування
2. Синхронна зміна радіочастот у алгоритмі протидії РЕБ

Для логування необхідно встановити час що відповідає реальному з точністю до однієї секунди. Для ефективного функціонування алгоритму зміни частот необхідно значно більша точність - до 20 мсек або краще. Також, треба зазначити, що система зв'язку базується на використанні радіо модемів і має затримку на прийом даних до 50 мсек (затримка є нерівномірною і може відрізнятися від затримку у протилежну сторону).

На основі аналізу алгоритмів синхронізації, вимог до алгоритму та апаратних обмежень побудуємо власний алгоритм, що максимально задовольняє умовам його використання.

По-перше, будемо синхронізувати час на наземному модулі використовуючи протокол NTP. Наземний модуль, на відміну від повітряного,

має можливість підключення до мережі у перед польотному режимі. Альтернативним джерелом синхронізації часу, у разі відсутності мережі, до наземного модуля підключено RTC, але RTC має меншу точність синхронізації і повинен перевірятися і налаштовуватися час від часу.

Таким чином, маємо еталонне джерело часу на наземному модулі. Тепер необхідно синхронізувати повітряний модуль з наземним.

За основу алгоритму синхронізації було обрано алгоритм Крістіана оскільки він простий у реалізації і має високу ефективність в невеликих мережах і мережах з малим завантаженням. Недоліки алгоритму нівелюються наступним чином:

1. Вимагає зовнішнього джерела точного часу – як зазначено вище, наземний модуль буде використовуватись як джерело еталонного часу.
2. Не має вбудованої захисту від переведення годинників у зворотну сторону – будемо проводити синхронізацію годинників один раз перед початком польоту. У цей час ключові системи що залежать від синхронності часу не будуть запущені і переведення годинників у зворотну сторону не спровокує неоднозначності у роботі алгоритмів.
3. Не дозволяє коректно встановлювати час в разі, якщо запит і відповідь передаються по різних маршрутах (потрібен різний час для передачі даних повідомлень) – будемо встановлювати початковий час як половина часу повного кола передачі усередненого за декілька ітерацій. Для уточнення часу при різному часі при передачі у різні сторони будемо корегувати час шляхом відправки поточного еталонного часу на повітряний модуль і перевірки розрахованого часу затримки з затримкою при передачі еталонного часу.

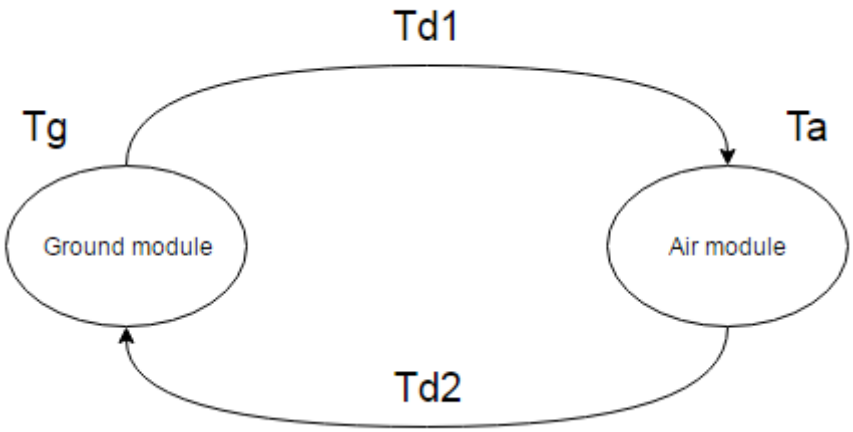

Рисунок 3.3 – Зміщення часу

Затримка при передачі даних з наземного на повітряний модуль (Td1) та з повітряного на наземний (Td2) є невідомими і можуть відрізнятися [рис. 3.3]. Проте, можливо визначити затримку повного кола (Td1 + Td2) і спираючись на алгоритм Крістіана апроксимувати Td1 = Td2 = (Td1 + Td2) / 2. Для більшої точності виконується декілька (на даний момент 10) обчислень затримки повного кола. З 10 обчислених затримок обираємо найменшу, як найбільш точну. Після обчислення затримки змінюємо час на повітряному модулі Ta = Tg + (Td1 + Td2) / 2.

Наступний крок – коректування обраної на першому кроці затримки. Коректування відбувається шляхом передачі таймстампів з наземного модуля на повітряний. На повітряному модулі, як і на першому кроці знаходимо найменшу затримку при передачі таймстампу. Ця затримка порівнюється із поточним часовим зміщенням (на першому кроці (Td1 + Td2) / 2) на наземній станції, якщо затримка має похибку що потрапляє у заду дельту точності синхронізації, то годинники вважаються синхронізованими, в іншому випадку часове зміщення задається корегується на половину різниці поточних затримок на повітряному і наземному модулі.

Таким чином, маємо годинники синхронізовані з точністю до половини затримки повного кола, що теоретично становить приблизно 30 мсек, проте

фактично, як показали практичні результати використання, завдяки відносній рівномірності розподілу значень затримок у дві сторони, маємо похибку приблизно у 5 мсек, що є цілком прийнятним результатом для подальшого використання даного алгоритму.

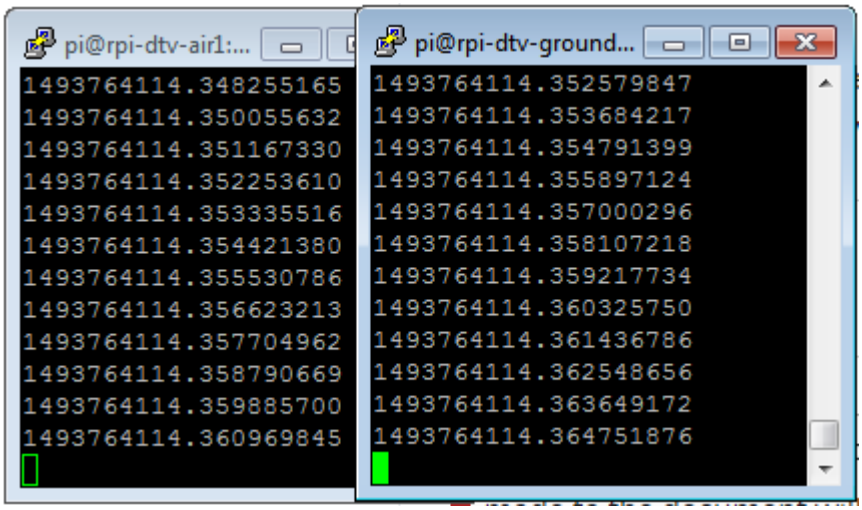

Рисунок 3.4 – Порівняння часу після синхронізації

3.5 Висновки

У цьому розділі, на основі попередніх досліджень, було обрано конкретні апаратні та програмні рішення для побудови прототипу. Також у розділі наведено результати практичного тестування можливих рішень задач на готовому прототипі і наведено результати функціонування розроблених компонент зв'язку.

4 ФУНКЦІОНАЛЬНО-ВАРТІСНИЙ АНАЛІЗ ПРОГРАМНОГО ПРОДУКТУ

У даному розділі проводиться оцінка основних характеристик програмного продукту для захищеного бездротового зв’язку з БПЛА.

Програмний продукт є тісно пов’язаним з апаратною складовою і спирається на особливості використання конкретних апаратних складових. Так, при зміні однієї з апаратних складових зазвичай необхідно переписувати програмний інтерфейс для її використання.

Нижче наведено аналіз різних варіантів реалізації ПЗ з метою вибору оптимальної, з огляду при цьому як на економічні фактори, так і на характеристики продукту, що впливають на продуктивність роботи і на його сумісність з апаратним забезпеченням. Для цього було використано апарат функціонально-вартісного аналізу.

Функціонально-вартісний аналіз (ФВА) – це технологія, яка дозволяє оцінити реальну вартість продукту або послуги незалежно від організаційної структури компанії. Як прямі, так і побічні витрати розподіляються по продуктам та послугам у залежності від потрібних на кожному етапі виробництва обсягів ресурсів. Виконані на цих етапах дії у контексті метода ФВА називаються функціями.

Мета ФВА полягає у забезпеченні правильного розподілу ресурсів, виділених на виробництво продукції або надання послуг, на прямі та непрямі витрати. У даному випадку – аналізу функцій програмного продукту й виявлення усіх витрат на реалізацію цих функцій.

Фактично цей метод працює за таким алгоритмом:

– визначається послідовність функцій, необхідних для виробництва продукту. Спочатку – всі можливі, потім вони розподіляються по двом групам: ті, що впливають на вартість продукту і ті, що не впливають. На цьому ж етапі оптимізується сама послідовність скороченням кроків, що не впливають на цінність і відповідно витрат.

– для кожної функції визначаються повні річні витрати й кількість робочих часів.

– для кожної функції на основі оцінок попереднього пункту визначається кількісна характеристика джерел витрат.

– після того, як для кожної функції будуть визначені їх джерела витрат, проводиться кінцевий розрахунок витрат на виробництво продукту.

4.1 Постановка задачі техніко-економічного аналізу

У роботі застосовується метод ФВА для проведення техніко-економічний аналізу розробки системи аналізу нелінійних нестаціонарних процесів. Оскільки основні проектні рішення стосуються всієї системи, кожна окрема підсистема має їм задовольняти.

Відповідно цьому варто обирати і систему показників якості програмного продукту.

Технічні вимоги до продукту наступні:

– програмний продукт повинен функціонувати на конкретній апаратній платформі, що проектується і поставляється разом з програмним продуктом;

– необхідно забезпечити мінімальну затримку при передачі даних між наземною станцією і БПЛА;

– необхідно забезпечити шифрування даних для запобігання перехоплення інформації з БПЛА і неможливості підміни інформаційних сигналів;

– необхідно забезпечити можливість протидії РЕБ, та підміні пакетів;

– необхідно швидко реалізувати ПЗ для прототипу для отримання фінансування;

– необхідно передбачати мінімальні витрати на впровадження програмного продукту у рамках проектування програмно-апаратного прототипу.

4.1.1 Обґрунтування функцій програмного продукту

Головна функція F_0– розробка програмного продукту, який буде встановлений у наземний і повітряний модулі системи захищеного зв’язку з БПЛА. Виходячи з конкретної мети, можна виділити наступні основні функції ПП:

F_1 – вибір мови програмування;

F_2 – вибір алгоритму шифрування даних;

F_3 – вибір фреймворку для відео стримінгу.

Кожна з основних функцій може мати декілька варіантів реалізації.

Функція F_1:

а) мова програмування С;

б) мова програмування С++;

Функція F_2:

а) стандарт AES із 128 бітним ключем;

б) стандарт AES із 256 бітним ключем;

Функція F_3:

а) фреймворк VLC;

б) фреймворк GStreamer.

4.1.2 Варіанти реалізації основних функцій

Варіанти реалізації основних функцій наведені у морфологічній карті системи (рис. 4.1). На основі цієї карти побудовано позитивно-негативну матрицю варіантів основних функцій (таблиця 4.1).

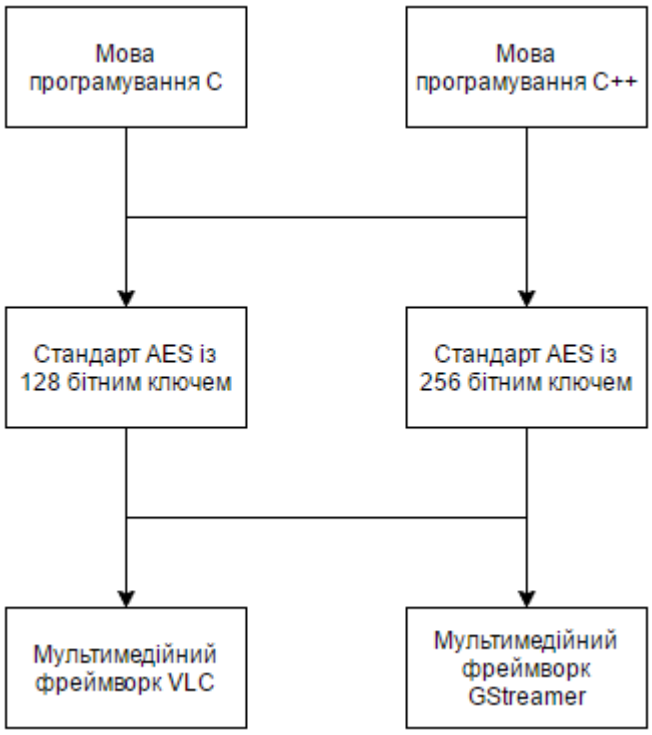

Рисунок 4.1 – Морфологічна карта

Морфологічна карта відображує всі можливі комбінації варіантів реалізації функцій, які складають повну множину варіантів ПП.

Таблиця 4.1 – Позитивно-негативна матриця

Основні функції	**Варіанти реалізації**	**Переваги**	**Недоліки**
F1	*А*	Займає менше часу при написанні коду	Гірша структурованість коду
	Б	Можливість використовувати об’єктно орієнтований підхід і, як наслідок, краща структурованість коду	Займає більше часу при написанні коду і вимагає більшої кількості спеціалістів
F2	А	Більша швидкість шифрування	Злом ключа займає менше часу
	Б	Менша швидкість шифрування	Злом ключа займає більше часу
F3	*А*	Має байндинги під більшіть мов програмування	Має невелику кількість плагінів
	Б	Дуже велика кількість плагінів	Має стабільні байндинги тільки під C++ та Python

На основі аналізу позитивно-негативної матриці робимо висновок, що при розробці програмного продукту деякі варіанти реалізації функцій варто відкинути, тому, що вони не відповідають поставленим перед програмним продуктом задачам. Ці варіанти відзначені у морфологічній карті.

Функція *F1:*

Оскільки написання програмної складової на C++ вимагає більше часу на структурування коду і проектування ієрархій, в той час, як C зручно використовувати при програмуванні апаратних драйверів та суміжних з ними інтерфейсів, для швидшого написання програмної частини пріорітетним є вибір мови програмування C.

Функція *F2:*

Надійне шифрування є найбільш пріорітетною складовою захищеного зв'язку. Проте, зважаючи на те, що шифрування AES із 128 бітним ключем на сьогоднішній день є, фактично, незламним, а шифрування 264 бітним ключем займає в середньому на 40% більше часу, оптимальним варіантом шифрування для зменшення затримки при передачі даних є 128 бітний ключ. Якщо з плином часу з'являться можливості теоретичного злому 128 бітного ключа, шифрування буде змінено на 264 бітний ключ.

Функція *F3:*

Обидва фреймворка мають мають досить широкий спектр кодеків та плагінів для реалізації поставленої мети, а також зручні інтерфейси для програмування нових поагінів на C, тому розглянемо варіанти реалізації програмного продукту із двома можливими фреймворками.

Таким чином, будемо розглядати такі варіанти реалізації ПП:

1. F1а – F2а – F3а
2. F1а – F2а – F3б

Для оцінювання якості розглянутих функцій обрана система параметрів, описана нижче.

4.2 Обґрунтування системи параметрів ПП

4.2.1 Опис параметрів

На підставі даних про основні функції, що повинен реалізувати програмний продукт, вимог до нього, визначаються основні параметри виробу, що будуть використані для розрахунку коефіцієнта технічного рівня.

Для того, щоб охарактеризувати програмний продукт, будемо використовувати наступні параметри:

– *X1* – час проектування та написання коду;

– *X2* – затримка при передачі даних;

– *X3* – завантаженість процессора мікрокомп’ютера;

– *X4* – час повного відновлення лінії зв’язку після атаки.

X1: Відображає час, за який буде спроектовано і реалізовано ПП що відповідає поставленим вимогам у розрахунку на одного розробника.

X2: Відображає затримку при передачі даних між наземним і повітряним модулем зв’язку з БПЛА.

X3: Відображає завантаженість процессора у режимі функціонування всіх вузлів зв’язку. Еталонним процессором для порівнянь є процессор Raspberry Pi 3 model B.

X4: Відображає час відновлення всіх компонент зв’язку після РЕБ.

4.2.2 Кількісна оцінка параметрів

Гірші, середні і кращі значення параметрів вибираються на основі вимог замовника й умов, що характеризують експлуатацію ПП як показано у табл. 4.2.

Таблиця 4.2 – Основні параметри ПП

Назва Параметра	**Умовні позначення**	**Одиниці виміру**	**Значення параметра**		
			гірші	**середні**	**кращі**
Час проектування та написання коду	X1	міс	12	9	5
Затримка при передачі даних	X2	мс	1000	500	150
Завантаженість процесора мікрокомп’ютера	X3	%	95	70	50
Час повного відновлення лінії зв’язку після атаки	X4	с	20	10	5

За даними таблиці 4.2 будуються графічні характеристики параметрів – рис. 4.2 – рис. 4.5.

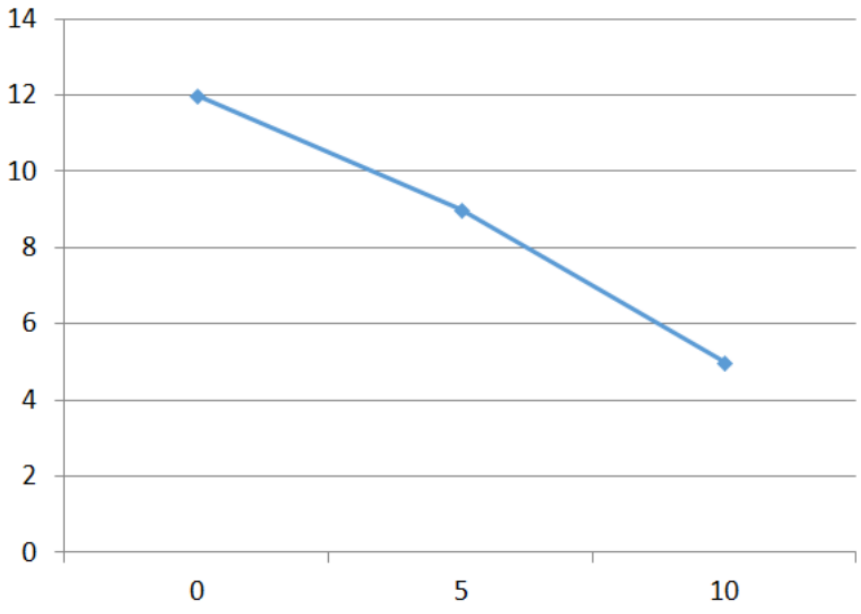

Рисунок 4.2 – X1: Час проектування та написання коду

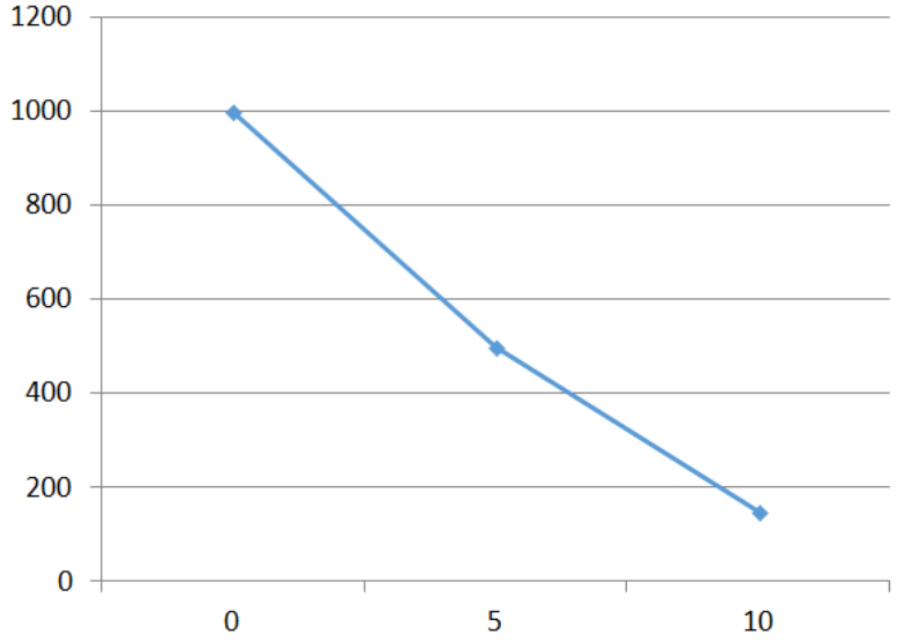

Рисунок 4.3 – X2: Затримка при передачі даних

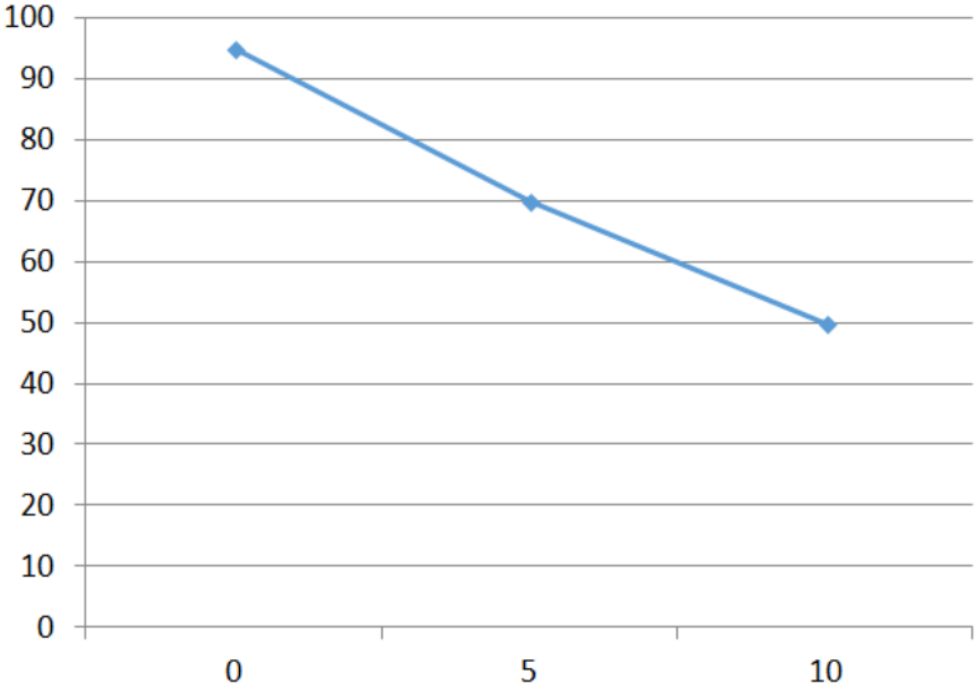

Рисунок 4.4 – X3: Завантаженість процесора мікрокомп’ютера

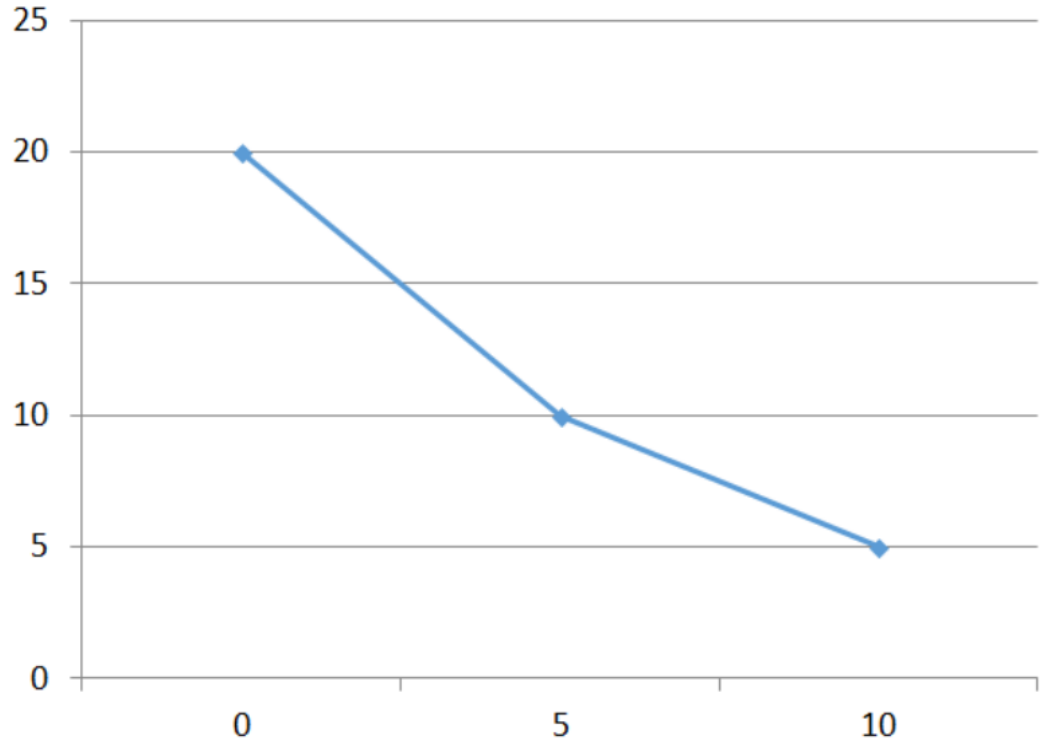

Рисунок 4.5 – X4: Час повного відновлення лінії зв’язку після атаки

4.2.3 Аналіз експертного оцінювання параметрів

Після детального обговорення й аналізу кожний експерт оцінює ступінь важливості кожного параметру для конкретно поставленої цілі – розробка програмного продукту, який дає найбільш точні результати при знаходженні параметрів моделей адаптивного прогнозування і обчислення прогнозних значень.

Значимість кожного параметра визначається методом попарного порівняння. Оцінку проводить експертна комісія із 7 людей. Визначення коефіцієнтів значимості передбачає:

– визначення рівня значимості параметра шляхом присвоєння різних рангів;

– перевірку придатності експертних оцінок для подальшого використання;

– визначення оцінки попарного пріоритету параметрів;

– обробку результатів та визначення коефіцієнту значимості.

Результати експертного ранжування наведені у таблиці 4.3.

Для перевірки степені достовірності експертних оцінок, визначимо наступні параметри:

а) сума рангів кожного з параметрів і загальна сума рангів:

$$R_i = \sum_{j=1}^{N} r_{ij} R_{ij} = \frac{Nn(n+1)}{2} = 70,$$

де N – число експертів, n – кількість параметрів;

б) середня сума рангів:

$$T = \frac{1}{n} R_{ij} = 17{,}5.$$

в) відхилення суми рангів кожного параметра від середньої суми рангів:

$$\Delta_i = R_i - T$$

Сума відхилень по всім параметрам повинна дорівнювати 0;

г)загальна сума квадратів відхилення:

$$S = \sum_{i=1}^{N} \Delta_i^2 = 201.$$

Таблиця 4.3 – Результати ранжування параметрів

Позначення параметра	Назва параметра	Одиниці виміру	Ранг параметра за оцінкою експерта							Сума рангів R_i	Відхилення Δ_i	Δ_i^2
			1	2	3	4	5	6	7			
Час проектування та написання коду	X1	міс	1	2	2	2	1	2	2	12	-5,5	30,25
Затримка при передачі даних	X2	мс	2	1	1	1	2	1	1	9	-8,5	72,25
Завантаженість процесора мікрокомп’ютера	X3	%	3	3	4	3	4	4	3	24	6,5	42,25
Час повного відновлення лінії зв’язку після атаки	X4	с	4	4	3	4	3	3	4	25	7,5	56,25
	Разом		10	10	10	10	10	10	10	70	0	201

Порахуємо коефіцієнт узгодженості:

$$W = \frac{12S}{N^2(n^3 - n)} = \frac{12 \cdot 201}{7^2(4^3 - 4)} = 0{,}82 > W_k = 0{,}67$$

Ранжування можна вважати достовірним, тому що знайдений коефіцієнт узгодженості перевищує нормативний, котрий дорівнює 0,67.

Скориставшись результатами ранжирування, проведемо попарне порівняння всіх параметрів і результати занесемо у таблицю4.4.

Таблиця 4.4 – Попарне порівняння параметрів

Параметри	Експерти							Кінцева оцінка	Числове значення
	1	2	3	4	5	6	7		
X1 і X2	>	<	<	<	>	<	<	<	0,5
X1 і X3	>	>	>	>	>	>	>	>	1,5
X1 і X4	>	>	>	>	>	>	>	>	1,5
X2 і X3	>	>	>	>	>	>	>	>	1,5
X2 і X4	>	>	>	>	>	>	>	>	1,5
X3 і X4	>	>	<	>	<	<	>	>	1,5

Числове значення, що визначає ступінь переваги *i*–го параметра над *j*–тим, a_{ij} визначається по формулі:

$$aij = \begin{cases} 1.5 \text{ при } Xi > Xj \\ 1.0 \text{ при Xi} = \text{Xj} \\ 0.5 \text{ при Xi} < Xj \end{cases}$$

З отриманих числових оцінок переваги складемо матрицю $A=\| a_{ij} \|$.

Для кожного параметра зробимо розрахунок вагомості $K_{вi}$ за наступними формулами:

$K_{\text{ві}} = \frac{b_i}{\sum_{i=1}^{n} b_i}$,де $b_i = \sum_{i=1}^{N} a_{ij}$.

Відносні оцінки розраховуються декілька разів доти, поки наступні значення не будуть незначно відрізнятися від попередніх (менше 2%).На другому і наступних кроках відносні оцінки розраховуються за наступними формулами:

$K_{\text{ві}} = \frac{b_i'}{\sum_{i=1}^{n} b_i'}$,де $b_i' = \sum_{i=1}^{N} a_{ij} b_j$.

Як видно з таблиці 4.5, різниця значень коефіцієнтів вагомості не перевищує 2%, тому більшої кількості ітерацій не потрібно.

Таблиця 4.5 – Розрахунок вагомості параметрів

Параметриx_i	Параметриx_j				Перша ітер.		Друга ітер.		Третя ітер	
	X1	X2	X3	X4	b_i	$K_{\text{ві}}$	b_i^1	$K_{\text{ві}}^1$	b_i^2	$K_{\text{ві}}^2$
X1	1,0	0,5	1,5	1,5	4,5	0,281	16,25	0,275	59,125	0,274
X2	1,5	1,0	1,5	1,5	5,5	0,344	21,25	0,360	77,875	0,361
X3	0,5	0,5	1,0	1,5	3,5	0,219	12,25	0,208	44,875	0,207
X4	0,5	0,5	0,5	1,0	2,5	0,156	9,25	0,157	34,125	0,158
Всього:					16	1	59	1	216	1

4.3 Аналіз рівня якості варіантів реалізації функцій

Визначаємо рівень якості кожного варіанту виконання основних функцій окремо.

Абсолютні значення параметрів X1(Час на вивчення API та написання коду) та X4 (Потенційний об’єм програмного коду) відповідають технічним вимогам умов функціонування даного ПП.

Абсолютне значення параметра X2(Об’єм пам’яті для збереження даних) буде найкращим у випадку обрання у F3 варіанта Б і становитиме 10, для варіанту А це значення буде 28.

Абсолютне значення параметра X3(Час отримання данних з хмари) буде найкраще при обрані варіанту А 180, а при обрані варіанту Б воно буде середнім 900.

Коефіцієнт технічного рівня для кожного варіанта реалізації ПП розраховується так (таблиця 4.6):

$$K_K(j) = \sum_{i=1}^{n} K_{вi,j} B_{i,j} ,$$

де n – кількість параметрів;$K_{вi}$– коефіцієнт вагомості i–го параметра; B_i – оцінка i–го параметра в балах.

За даними з таблиці 4.6 за формулою

$$K_K = K_{\text{ТУ}}[F_{1k}] + K_{\text{ТУ}}[F_{2k}] + \ldots + K_{\text{ТУ}}[F_{zk}],$$

визначаємо рівень якості кожного з варіантів:

K_{Kl} = 3,249 + 1,035 + 1,422 + 2,466 = 8,172

K_{K2} = 3,249 + 1,035 + 0,316+ 0,548= 5,148

Як видно з розрахунків, кращим є перший варіант, для якого коефіцієнт технічного рівня має найбільше значення.

Таблиця 4.6 – Розрахунок показників рівня якості варіантів реалізації основних функцій ПП

Основні функції	Варіант реалізації функції	Параметри	Абсолютне значення параметра	Бальна оцінка параметра	Коефіцієнт вагомості параметра	Коефіцієнт рівня якості
F1	А	X1	6	9	0,361	3,249
F2	А	X4	10	5	0,207	1,035
F3	А	X2	88	2	0,158	0,316
		X3	900	2	0,274	0,548
	Б	X2	55	9	0,158	1,422
		X3	210	9	0,274	2,466

4.4 Економічний аналіз варіантів розробки ПП

Програмний продукт, що буде встановлюватись на сервер кафедри вже розроблено, проте необхідно увесь рік тримати під контролем сервер, на якому буде встановлено програмне забезпечення. Отже

$T_1 = 90 \cdot 1.7 \cdot 0.8 = 122.4$ людино-днів.

$T_1 = 122.4 \cdot 8 = 979.2$ людино-годин;

В розробці беруть участь C/C++ програміст з окладом 12000 грн та системний аналітик з окладом 8000 грн. Визначимо зарплату за годину за формулою:

$$С_ч = \frac{М}{T_m \cdot t} грн.,$$

де М – місячний оклад працівників; T_m – кількість робочих днів тиждень; t – кількість робочих годин в день.

$$С_ч = \frac{11000 + 9000}{2 \cdot 21 \cdot 8} = 59{,}52 \text{ грн.}$$

Тоді, розрахуємо заробітну плату за формулою

$$С_{ЗП} = С_ч \cdot Т_i \cdot К_Д,$$

де $С_ч$– величина погодинної оплати праці програміста;$Т_i$ – трудомісткість відповідного завдання; $К_Д$ – норматив, який враховує додаткову заробітну плату.

Зарплата розробників за варіантами становить:

$С_{ЗП} = 59{,}52 * 979.2 \cdot 1.2 = 69900.6$ грн на рік.

Відрахування на єдиний соціальний внесок незалежно від групи професійного ризику становить 22%:

$С_{ВІД} = С_{ЗП} \cdot 0.22 = 69900.6 * 0.22 = 15378.13$ грн.

Тепер визначимо витрати на оплату однієї машино-години. ($С_М$)

Працюватиме одна електрона обчислювальна машина цілодобово:

$С_Г = 12 \cdot М \cdot К_З = 12 \cdot 12000 \cdot 0,2 = 28800$ грн.

З урахуванням додаткової заробітної плати:

$С_{ЗП} = С_Г \cdot (1 + К_З) = 28800 \cdot (1 + 0.2) = 34560$ грн.

Відрахування на єдиний соціальний внесок:

$С_{ВІД} = С_{ЗП} \cdot 0.2 = 34560 * 0,2 = 12669$ грн.

Амортизаційні відрахування розраховуємо при амортизації 25% та вартості ЕОМ – 25000 грн.

$С_А = К_{ТМ} \cdot К_А \cdot Ц_{ПР} = 1.15 \cdot 0.25 \cdot 25000 = 7187,5$ грн.,

де $К_{ТМ}$– коефіцієнт, який враховує витрати на транспортування та монтаж приладу у користувача; $К_А$ – річна норма амортизації; $Ц_{ПР}$ – договірна ціна приладу.

Витрати на ремонт та профілактику розраховуємо як:

$С_Р = К_{ТМ} \cdot Ц_{ПР} \cdot К_Р = 1.15 \cdot 25000 \cdot 0.05 = 1437,5$ грн.,

де $К_Р$– відсоток витрат на поточні ремонти.

Ефективний годинний фонд часу ПК за рік розраховуємо за формулою:

$Т_{ЕФ}$ =($Д_К$ – $Д_В$ – $Д_С$ – $Д_Р$) · $t_З$· $К_В$ = (365 – 104 – 8 – 16) · 8 · 0.9 = 1706.4 годин,

де $Д_К$ – календарна кількість днів у році; $Д_В$, $Д_С$ – відповідно кількість вихідних та святкових днів; $Д_Р$ – кількість днів планових ремонтів устаткування; t –кількість робочих годин в день; $К_В$– коефіцієнт використання приладу у часі протягом зміни. Витрати на оплату електроенергії розраховуємо за формулою:

$С_{ЕЛ}$ = $Т_{ЕФ}$· $N_С$· $Ц_{ЕН}$ =1706.4 · 0.6 · 1.93819 = 1984.39 грн.,

де $N_С$ – середньо-споживча потужність приладу; $К_З$– коефіцієнтом зайнятості приладу; $Ц_{ЕН}$ – тариф за 1 КВт-годин електроенергії.

Накладні витрати розраховуємо за формулою:

$С_Н$ = $Ц_{ПР}$·0.67 = 25000 · 0,67 = 16750 грн.

Тоді, річні експлуатаційні витрати будуть:

$$С_{ЕКС} = С_{ЗП} + С_{ВІД} + С_А + С_Р + С_{ЕЛ} + С_Н$$

$С_{ЕКС}$ = 69900.6 + 15378.13 + 7187.5 + 1437.5 + 1984.39 + 16750 = 112638.12 грн.

Собівартість однієї машино-години ЕОМ дорівнюватиме:

$С_{М-Г}$ = $С_{ЕКС}$/ $Т_{ЕФ}$ = 112638.12 / 1706,4 = 66 грн/час.

Оскільки в даному випадку всі роботи, які пов‘язані з розробкою програмного продукту ведуться на ЕОМ, витрати на оплату машинного часу, в залежності від обраного варіанта реалізації, складає:

$$С_М = С_{М-Г} \cdot Т$$

$C_М = 66 * 979.2 = 64627.2$

Накладні витрати складають 67% від заробітної плати:

$$C_Н = C_{ЗП} \cdot 0,67$$

$C_Н = 69900.6 * 0.67 = 46833.40$ грн.;

Отже, вартість розробки ПП за варіантами становить:

$$C_{ПП} = C_{ЗП} + C_{ВІД} + C_М + C_Н$$

$C_{ПП} = 69900.6 + 15378.13 + 63775.29 + 46833.40 = 196739.33$ грн.;

4.5 Вибір кращого варіанта ПП техніко-економічного рівня

Розрахуємо коефіцієнт техніко-економічного рівня за формулою:

$$К_{ТЕРj} = К_{Кj} / С_{Фj},$$

$К_{ТЕР1} = 8,172 / 196739.33 = 4,15 \cdot 10^{-5}$;

$К_{ТЕР2} = 5,148 / 196739.33 = 2,61 \cdot 10^{-5}$;

Як бачимо, найбільш ефективним є перший варіант реалізації програми з коефіцієнтом техніко-економічного рівня $К_{ТЕР1} = 4,15 \cdot 10^{-5}$.

4.6 Висновки

В даному розділі проведено повний функціонально-вартісний аналіз програмного забезпечення, який було розроблено в рамках дипломного проекту. Процес аналізу можна умовно розділити на дві частини.

У першій проведено дослідження програмного продукт з технічної точки зору: були поставлені основні параметри, що повинні бути головними при обранні кращої реалізації. На основі отриманих значень параметрів, оцінок експертів було обчислено коефіцієнт технічного рівня, який надалі у другій частині допоміг обрати найкращий варіант з техніко-економічної точки зору.

У другій частині виконувалося економічне обґрунтування альтернативних варіантів реалізації. Порівняння робились з урахуванням витрат на заробітні плати, електроенергії, накладні витрати.

Після виконання функціонально-вартісного аналізу програмного комплексу що розроблюється. Після проведення першої частини аналізу було виявлено, що перший варіант є найбільш оптимальним для реалізації. Його показник техніко-економічного рівня якості $К_{ТЕР1} = 4{,}15 \cdot 10^{-5}$;

Цей варіант реалізації програмного продукту має такі параметри:

- мова програмування – C;
- алгоритм шифрування - AES-128;
- мультимедійний фреймворк – GStreamer.

Даний варіант виконання програмного комплексу відповідає усім поставленим вимогам та може бути розроблений відносно швидко.

ВИСНОВКИ

У першому розділі було розглянуто загальні відомості про сучасний стан сфери БПЛА, більш конкретно розглянуто принципи систем зв'язку з БПЛА та способи їх злому. Проведений аналіз показав, що існують випадки злому зв'язку на поліцейських і воєнних БПЛА, а цивільні дрони є не захищеними взагалі. Спираючись на проведені дослідженння, було прийнято рішення розробляти окремий модуль зв'язку, що може бути встановлено в будь-який корпус БПЛА. Модуль зв'язку повинен забезпечити захист від таких розповсюджених атак, як заглушення, підміна пакетів, GPS спуфінг та перехоплення даних.

У другому розділі було розглянуто варіанти реалізації головних компонент системи зв'язку. Для зручності систему було розділено на складові частини для проведення більш детального аналізу. Було проведено детальний огляд доступних варіантів реалізації поставлених задач: алгоритмів шифрування, стандартів передачі даних, опис головних характеристик камери БПЛА та алгоритмів синхронізації часу у розподілених системах.

Функціонально-вартісний аналіз програмного продукту дозволив ретельно порівняти спірні технології та обрати найкращі рішення спираючись на широкий спектр параметрів. Зокрема, функціонально-вартісний аналіз допоміг зробити вибір мультимедійного фреймворку на користь GStreamer, хоча VLC, як альтернатива, виглядав ні чим не гірше до проведення аналізу.

У третьому розділі описується безпосередньо процес побудови прототипу на основі попередніх досліджень. Було остаточно порівняно та обрано конкретні апаратні та програмні рішення для побудови прототипу. Також у розділі наведено результати практичного тестування можливих рішень задач на готовому прототипі і результати функціонування розроблених компонент зв'язку.

Розроблений прототип представляє собою “black box” який можна підключити до будь якого фрейму безпілотника, що підтримується пілотним контроллером Pixhawk. Прототип має досить невеликі розміри та вагу, що робить його зручним у використанні при розробці нових БПЛА. Цей факт було підтверджено шляхом проектування власного міні-квадрокоптера, який мав задовільні габарити та задовольняв злітній вазі стандартних фреймів квадракоптерів від DJI.

Прототип забезпечує надійний захист лінії зв’язку з БПЛА. Шифрування потоку даних надає змогу протидіяти перехопленню даних з борту БПЛА для їх подальшого аналізу та використання. Програмним ядром прототипу є розроблені алгоритми протидії заглушенню лінії зв’язку та підміні пакетів. Ці алгоритми мають гнучкі можливості до відстеження відповідних атак та забезпечуть адаптацію системи польоту в умовах РЕБ. Для тестування в офісних умовах, було розроблено портативний заглушуючий пристрій. При тестуванні прототип успішно пройшов всі заплановані тести, що підтверджує ефективність розроблених систем.

В умовах настачі часу, на даний момент не реалізовано захист від GPS спуфінгу. Це, безумовно, є значною вразливістю і повністю унеможливлює безпечний політ по координатам. Проте, викрасти БПЛА шляхом GPS спуфінгу є фактично неможливим. Завдяки достатньо захищеній основній лінії зв’язку, можна перейти у режим ручного пілотування і посадити безпілотник у необхідному місці.

Актуальними шляхами розвитку розробленого прототипу є вдосконалення параметрі вже реалізованих компонент – підвищення максимально можливого бітрейту відео потоку, збільшення дальності радіо зв’язку, тощо. Модулі апаратної та програмної стабілізації відео є також досить актуальним функціоналом в умовах відео передачі с борта БПЛА.

ПЕРЕЛІК ПОСИЛАНЬ

1. Исследователь показал, как взломать полицейских дронов, имея аппаратуру за $40. Електронний ресурс. Режим доступу: https://xakep.ru/2016/04/05/uav-flaws/ – Дата доступу: 20.05.2017.
2. Теодорович Н. Н. Способы обнаружения и борьбы с малогабаритными беспилотными летательными аппаратами / Н. Н. Теодорович, С. М. Строганова, П. С. Абрамов // Интернет-журнал «НАУКОВЕДЕНИЕ». – 2017. – № 9. – С. 2–7.
3. Слюсар В. І. ЭЛЕКТРОНИКА: Наука, Технология, Бизнес / Слюсар В. І. – Москва : РИЦ Техносфера – 2010. – 80 с. – (Передача данных с борта БПЛА: стандарты НАТО).
4. Overview of UCLA MinuteMan-Project. Електронний ресурс. Режим доступу: www.icsl.ucla.edu – Дата доступу: 20.05.2017.
5. Илюшко В. М. Система передачи данных на базе высотного беспилотного летательного аппарата (СПД «Фаэтон») / В. М. Илюшко, Т. М. Нарытник // Зв’язок. – 2004. – № 7 – С. 38–39.
6. Unmanned Aircraft Systems (UAS) Roadmap / USA : Office of the Secretary of Defense. – 2005. – с. 33 – (Joint Land Attack Elevated Netted Sensor).
7. Слюсар В. І. ЭЛЕКТРОНИКА: Наука, Технология, Бизнес / Слюсар В. І. – Москва : РИЦ Техносфера – 2010. – 56 с. – (Радиолинии связи с БПЛА: Примеры реализации).
8. Advanced Encryption Standard. Електронний ресурс. Режим доступу: https://uk.wikipedia.org/wiki/Advanced_Encryption_Standard – Дата доступу: 20.05.2017.
9. Сушко С. А. Практическая криптология: лекция 9 / С. А. Сушко – 2016. – с. 1. – (Общее описание криптоалгоритма AES).

10. Достоинства и недостатки симметричного и асимметричного методов шифрования. Електронний ресурс. Режим доступу: http://wm-help.net/lib/b/book/1571601295/21 – Дата доступу: 20.05.2017.
11. Бєловол С. Світовий досвід правового регулювання використання безпілотників / С. Бєловол – 2016. – с. 2. – (Визначення понять, історія розвитку предмету досліджень та постановка проблеми) (Передумови розробки норм використання безпілотних літальних апаратів) (Класифікація безпілотних літальних апаратів)
12. How to choose FPV camera for drones. Електронний ресурс. Режим доступу: https://oscarliang.com/best-fpv-camera-quadcopter/ – Дата доступу: 20.05.2017.
13. Carr E. Unmanned Aerial Vehicles: Examining the Safety, Security, Privacy and Regulatory Issues of Integration into U.S. Airspace / E. Carr – 2012. – c. 15. – (Safety Issues) (Security Issues) (Privacy Issues)
14. Атакующие БПЛА и системы противодействия им, обзор. Електронний ресурс. Режим доступу: http://savepearlharbor.com/?m=201506&paged=339 – Дата доступу: 20.05.2017.
15. Результаты конференции «Индустрия беспилотных авиационных систем». Електронний ресурс. Режим доступу: http://www.helirussia.ru/ru/dlya_smi/press_relizyi/2016/05/31/uas_conference_results /339 – Дата доступу: 20.05.2017.
16. Tokyo's solution to rogue drones? Drones with nets. Електронний ресурс. Режим доступу: https://www.engadget.com/2015/12/11/tokyo-drone-net/ – Дата доступу: 20.05.2017.
17. Network Time Protocol. Електронний ресурс. Режим доступу: https://fr.wikipedia.org/wiki/Network_Time_Protocol – Дата доступу: 20.05.2017.