МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Сумський державний університет

Факультет електроніки та інформаційних технологій

Кафедра електроніки і комп'ютерної техніки

«До захисту допущено»

Завідувач кафедри ЕКТ

_____ Анатолій ОПАНАСЮК

(підпис) (Ім’я та ПРІЗВИЩЕ)

____ _____ 2024 р.

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття освітнього ступеня «бакалавр»

зі спеціальності 172 «Телекомунікації та радіотехніка»

освітньо-професійної програми «Мережеві та інтернет-технології»

на тему:

ТЕЛЕКОМУНІКАЦІЙНИЙ ПРИСТРІЙ ЗАХИСТУ ДАНИХ ДЛЯ БЕЗПІЛОТНИХ ЛІТАЛЬНИХ АПАРАТІВ

Здобувача групи ТК-01 _____ Забуги Артема Костянтиновича

Кваліфікаційна робота містить результати власних досліджень. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело.

_____ _____

(підпис) (Ім’я та ПРІЗВИЩЕ)

Керівник, доцент, к.т.н., доцент Ігор КУЛИК _____

(підпис)

Суми – 2024

СУМСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ

Факультет електроніки та інформаційних технологій

Кафедра електроніки і комп'ютерної техніки

Напрям підготовки 172 Телекомунікації та радіотехніка

Освітня програма Мережеві та інтернет-технології

ЗАТВЕРДЖУЮ

Зав. кафедрою Опанасюк А. С.

"___"_____ 2024 р.

З А В Д А Н Н Я

на кваліфікаційну роботу бакалавра

1. Тема роботи _____

затверджена наказом по університету "13" березня 2024 р. № 0255-VI.

2. Термін здачі студентом завершеної роботи _____

3. Вихідні дані до роботи _____

4. Зміст розрахунково-пояснювальної записки (перелік питань, що належить розробити): _____

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень):

_____

_____

_____

6. Дата видачі завдання _____

8. Керівник роботи _____

9. Завдання прийняв до виконання _____

КАЛЕНДАРНИЙ ПЛАН

№ п/п	Назва етапів кваліфікаційної роботи	Термін виконання етапів роботи	Примітки
1	Постановка завдання	06.05.24	
2	Огляд літератури	10.05.24	
3	Розробка алгоритму роботи пристрою	14.05.24	
4	Порівняння алгоритмів шифрування	17.05.24	
5	Структурна схема пристрою	21.05.24	
6	Написання вступу і висновків до роботи	25.05.24	
7	Подача електронного варіанту роботи для перевірки на плагіат в деканат	04.06.24	
8	Представлення роботи керівнику і отримання відгуку	09.06.24	
9	Представлення роботи кафедрі для отримання рецензії	09.06.24	

Студент _____

Керівник роботи _____

«___» _____ 2024 р

РЕФЕРАТ

Тема кваліфікаційної роботи бакалавра «Телекомунікаційний пристрій захисту даних для безпілотних літальних апаратів».

Пояснювальна записка складається зі вступу, 4 розділів, висновків, переліку посилань, додатків. Загальний обсяг роботи становить 52 сторінки, у тому числі 47 сторінок основного тексту, 2 сторінки переліку посилань, 3 сторінки додатків. Текст включає 5 таблиць, 13 ілюстрацій, 17 джерел посилання, 3 додатки.

В першому розділі досліджується актуальність проблеми та технології які застосовуються в обладнанні безпілотних літальних апаратів. Досліджуються методи, які можна використати при виконанні роботи та встановлюється задача проектування.

В другому розділі наводиться пропонований алгорим роботи пристрою захисту даних, з покроковим поясненням його роботи.

Третій розділ містить порівняння за критеріями алгоритмів, що застосовуються для криптографічного шифрування даних, визначення кращого алгоритму для виконання завдання.

Четвертий розділ містить структурну схему пристрою і опис його елементів. Також наведено детальний опис функціонування пристроїв шифрування та завадостійкого кодування інформації.

В результаті проведеної роботи було спроектовано пристрій захисту даних: проведений аналіз і порівняння алгоритмів шифрування дозволили обрати оптимальний варіант – шифрування за стандартом AES. Метод завадостійкого кодування Боуза-Чоудхурі-Хоквінгема, якому надано перевагу в роботі, забезпечує високий рівень захисту від помилок. У висновку – пристрій захисту є ефективним, здатним підтримувати конфіденційність та завадостійкість передаваних даних.

Ключові слова: БПЛА, ЗАХИСТ ДАНИХ, ШИФРУВАННЯ, ЗАВАДОСТІЙКЕ КОДУВАННЯ, РАДІОКАНАЛ.

ЗМІСТ

					ЕлІТ 6.172.00.02.122 ПЗ		
Змн.	*Арк.*	*№ докум.*	*Підпис*	*Дата*			
Розроб.		Забуга А. К.			Телекомунікаційний пристрій захисту даних для безпілотних літальних апаратів. ПЗ	*Літ.*	*Арк.* 3 / *Акрушів*
Перевір.		Кулик І. А.					
Н. Контр.						СумДУ, гр. ТК-01	
Затверд.		Опанасюк А. С.					

ВСТУП

Безпілотні літальні апарати набули значного поширення – стрімко зростають об’єми їх виробництва та використання, а технологічне насичення безпілотників постійно оновлюється. Враховуючи їх швидкий розвиток, завжди існує потреба в більш досконалому обладнанні, що використовується для встановлення надійного зв’язку між бортом і землею.

Питання забезпечення завадостійкого, конфіденційного зв’язку з БпЛА стоїть досить гостро також тому, що у випадку успіху зловмисник може втручатись в маршрут слідування борту, припиняти політ, отримувати дані про місцезнаходження пілота, тощо.

Рішенням проблем з безпекою комунікацій безпілотника є застосування пристрою захисту даних. Він передбачає поєднання завадостійкого кодування і криптографічних алгоритмів, багато з яких добре досліджені та відповідають мінімальним вимогам роботи з БпЛА.

Проте, тут постає необхідність у виборі оптимального алгоритму, з огляду на обмеження, які висувають телекомунікації між наземною станцією управління та бортом. Потреба в надійному зв’язку зумовлює актуальність даного пристрою і для цивільної, і для військової сфер застосування дронів.

Метою роботи є проектування такого телекомунікаційного пристрою захисту даних БпЛА.

1 ОГЛЯД ЛІТЕРАТУРИ

1.1 Радіоканали, що застосовуються

Експлуатацію безпілотних літальних апаратів не можливо уявити без застосування радіоканалів – завдяки надійному зв'язку і ефективному розподіленню радіочастотного спектру БпЛА стало дуже зручно використовувати в різних сферах людської діяльності. Кожна система безпілотника має окремий виділений канал зв'язку, типовий їх перелік наведено в таблиці 1.

Таблиця 1 – Склад та призначення радіоканалів у БпЛА

Назва радіоканалу	Призначення радіоканалу
Канал супутникової навігації	Прийом сигналів супутникових систем для обчислення місцезнаходження борта і корекції курсу.
Канал ручного дистанційного керування	Передача сигналу ручного пульту управління зі станції наземного керування до БпЛА.
Зворотній відеоканал	Передача відеозображення курсової камери, яка дозволяє пілоту управляти дроном, з БПЛА до станції керування;Передача з борта зображення кутомірної відеокамери для виділеного спостереження в реальному часі.
Канал телеметрії	Дуплексний радіоканал зв'язку: з борта надходять дані про польотні параметри та стан в якому перебувають бортові системи; На борт надходять разові команди, наприклад на перемикання режиму керування.
Командна радіолінія	Симплексний радіоканал зв'язку: на борт надходять разові команди (не телеметричні), наприклад щодо управління корисним навантаженням або пов'язаного обладнання.

Командна радіолінія і радіоканал телеметрії. Командна лінія зв'язку використовується для зв'язку з БпЛА, а також передачі даних до наземного пункту керування на високій швидкості. Зазвичай для передачі команд застосовують хвилі ультракороткого діапазону до 400 МГц, L-діапазону (від 1 до 2 ГГц), S-діапазону (від 2 до 4 ГГц), C-діапазону (від 3,5 до 8 ГГц) та X-смуги (від 7 до 10,7 МГц) у випадку візуального спостереження. Організуючи зв'язок на більших застосовують дрони-ретранслятори сигналів та інфраструктуру супутникового зв'язку. У випадку бортів малого та надмалого розміру також мають місце мережі стільникового зв'язку (другого покоління, застосовуючи частоту від 780 до 960 МГц або діапазон від 1,7 до 2,7 ГГц), WiFi мережі (діапазони 2,4 ГГц, від 5,15 до 6,4 ГГц), LTE (діапазони 0,8 ГГц, 1,8 ГГц, 2,6 ГГц), а також рішення з LoRaWAN [1].

Радіоканал для супутникової навігації. Переважна частина апаратів використовує супутникові системи навігації. Найпоширенішими з них є GPS (США), ГЛОНАСС (Росія), Beidou/GNSS (Китай), Galileo (ЄС) та IRNSS (Індія). В супутникових системах сигнали генеруються на частотах від 1,1-1,6 ГГц.

Зворотній відеоканал. Серед особливостей систем, що підтримують керування за відеоканалом важливою є можливість не зважати на сигнал супутникової навігації. Завдяки безперервній відеотрансляції з БпЛА і невеликих затримках, пілот бере на себе ручне керування бортом (рисунок 1). Подібні схеми керування мають збільшений радіус дії завдяки потужним відеопередавачам (до 10 Вт в діапазоні 1,2 ГГц), а також застосуванню підсилювачів/ретрансляторів сигналів ручного управління в діапазоні 433 МГц (до прикладу, RMILEC T4346NB18) потужністю 2-5 Вт.

Пункт керування також забезпечується направленими антенами, що мають посилення 12 дБ. Підтримувана дальність ручного керування в результаті застосування вищевказаних засобів складає близько 60 км при чутливості

приймача в пункті управління мінус 92 дБм, а приймача команд на борта мінус 110 дБм [2].

Типовим недоліком керування за зворотнім відеоканалом стає потужний сигнал від бортової антени (всенаправленої), що до того ж посилюється під час аналогової відеопередачі (PAL). Це є вагомим демаскуючим чинником. Застосувавши будь-яку перешкоду, яка перериває відеопередачу на десять секунд, спостерігаємо втрату керування. Перешкоди за каналом управління створюють ті ж самі проблеми, з тією відмінністю, що значно зростає час, щоб встановити аналізатор/постановник перешкод і здійснити перехоплення керування бортом [1].

В контексті захисту каналів зв'язку, відповідно, керування бортом за зворотним відеоканалом є найменш захищеним варіантом: потужні передавачі БпЛА та на землі легко будуть виявлені та блоковані за допомогою відносно простих засобів невеликої потужності.

Режими управління БпЛА різняться відповідно до технічного рівня виконання апарату і завдань, виконання яких від нього очікують. При ручному керуванні БпЛА в зоні візуального спостереження пілотом використовується один канал – пропорційне керування вручну (рисунок 1). Практична дальність роботи в такому режимі становить до 0,7 км, в залежності від оснащення апарата.

У випадку необхідності управління за багатьма (8-16 каналів) радіоканалами застосовують широкосмуговий цифровий сигнал, промодульований FSK (частотна маніпуляція) та стрибками за частотою (FHSS/DSM2). Ширина спектру становить 200-500 кГц, діапазон – 2,4 ГГц. Ідентифікацію каналу забезпечує адресне кодування пакетів, з ціллю не допускати затримку при управлінні.

Керування бортом відбувається вручну при візуальному спостереженні, за зворотним радіоканалом, або у автоматичному режимі (рисунок 1).

У випадку ручного візуального керування користуються наступними засобами: пульт дистанційного ручного управління (1), симплексний канал ручного управління (2), приймач сигналів керування (3);

При керуванні за зворотним відеоканалом використовується: передавач команд пульту управління з підсиленням потужності (4), направлені антени (5), канал відеозв’язку з борта (6), окремий передавач відеоданих (якщо задіяний, 7), курсова камера (8); направлена антена відеозв’язку з борта (9), приймач відеоданих (10), монітор відеозображення з борта (11);

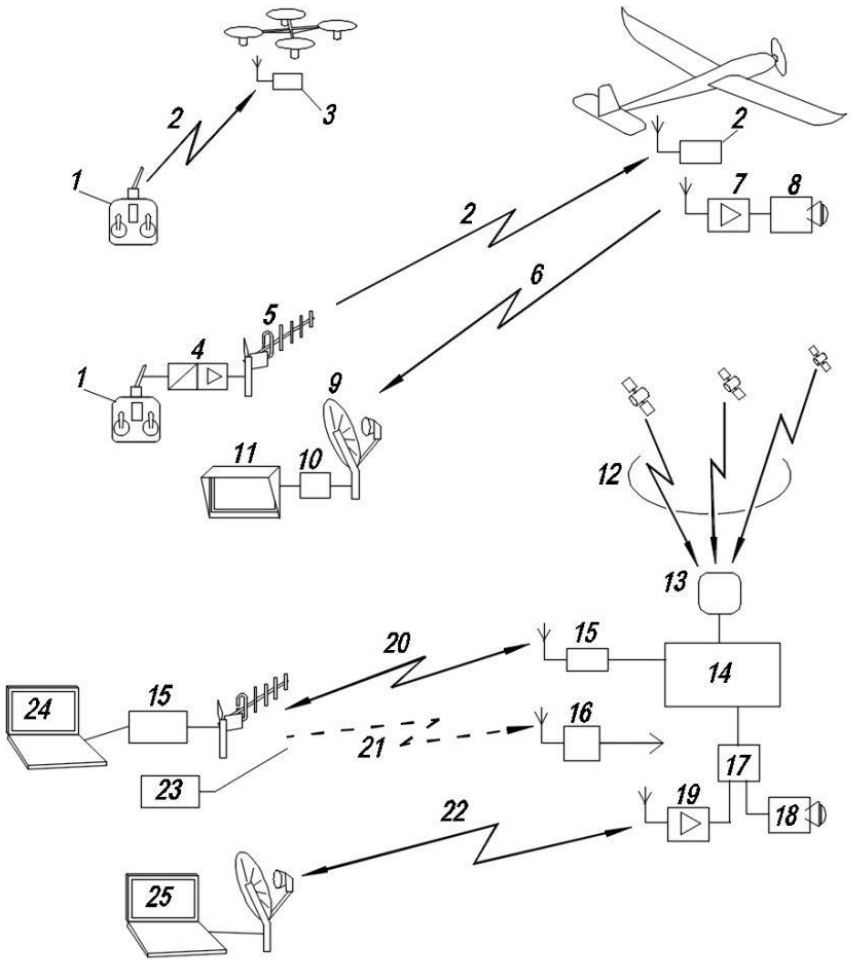


Рисунок 1 – Радіоканали безпілотних авіаційних комплексів

У випадку автоматичного управління: навігаційний супутниковий канал (12), приймач сигналів навігації (13), бортовий комп’ютер (14), передавач телеметричних даних (15), приймач командних сигналів (16), пристрій синхронізації телеметрії і відеоданих (17), корисне навантаження (наприклад, відеокамера) (18), додатковий передавач корисного навантаження (19), дуплексний канал телеметрії (20), симплексний канал команд (щодо обладнання борта) (21),

цифровий дуплексний канал корисного навантаження (22), передавач командних сигналів (23), пристрій управління польоту наземного пункту керування (24), пристрій. що керує корисним навантаженням (25).

Завжди існує ризик перехоплення керування зловмисником шляхом перебору адрес, для зупинки польоту борта. Це можливо тому, що кожен серійний пульт управління формує пакети команд за стандартною моделлю. Натомість, шумова загороджувальна завада далеко не завжди критично впливає на політ, бо наявні приймачі радіокерування на борта автоматично діагностують розрив зв'язку, перемикаючи виконавчі механізми, та виводять БпЛА з зони постановки перешкод [3].

Серед ефективних способів захистити радіолінію ручного керування є вибір направленої антени для пульту управління. Тому що дальність між оператором і бортом несуттєва, кут місцезнаходження БпЛА становить більше 30 градусів та спостерігається значне зниження рівня сигналу в області постановки перешкод. Таким чином, достовірний аналіз пакетів команд ускладняється або сильно втрачає в оперативності.

Системи автоматичного керування виключають канал ручного управління, тому не вимагають мінімальних затримок при трансляції відео. В цьому випадку контроль здійснюється за двонаправленим радіоканалом телеметрії, який відображає оператору наземного керування місцезнаходження та параметри борта. Режим польоту змінюється разовими командами. Захищений вузькосмуговий канал телеметричних даних (швидкістю менше 10 кБ/с) не вибагливий до затримок при шифруванні та дешифруванні даних. Передавач телеметрії або цифрового каналу зв'язку активується або деактивується за односторонньою командою в реальномучасі (рисунок 1), без перешкод для автоматично виконуваного польоту або вже заданої траєкторії польоту.

На сьогодні передавачі телеметричних даних дронів є поєднанням трансивера і процесора, що ним керує. За умови достатньої швидкодії процесора

шифрування даних можна здійснювати різними алгоритмами шляхом програмної реалізації або апаратно, гарантуючи захист від перехоплення керування під час польоту [4].

В режимі автоматичного керування БпЛА радіоканал супутникової навігації є вразливим. Якщо відсутні дані супутникового навігаційного приймача, в залежності від різновиду автопілота і його конфігурації, відбувається аварійне завершення польоту, інакше помилки позиціонування будуть зростати в системі управління. Окремої уваги варті так звані «розумні перешкоди», шо надходять по каналу супутникового зв'язку, вносячи в систему керування спотворені координати борта. Якщо навігаційний канал не подає ознак відмови це тягне за собою відхилення апарату від початкової траєкторії, яке неможливо спрогнозувати.

1.2 Методи кодування

Одним із засобів підвищити вірність повідомлень під час прийому є кодування завадостійкими кодами, прикладом якого може слугувати мажоритарний спосіб.

Мажоритарний спосіб закладається в тому, що при передачі до каналу зв'язку одне й те ж повідомлення повторюється кілька разів, а під час прийому порівнюються однойменні кодові комбінації (однойменні двійкові розряди). Вірною обирається та комбінація (біт), що приймалася більше разів. В мажоритарному кодуванні вірогідність помилки при прийомі символу повідомлення визначається за формулою:

$$p_{\text{біт маж}} = \sum_{i=\frac{c+1}{2}}^{c} C_c^i \cdot p_{\text{біт0}}^i \cdot (1 - p_{\text{біт0}})^{c-i}$$

де с – число повторень при передачі біта повідомлення;

$P_{\text{біт0}}$ – ймовірність бітової помилки.

Розрахунки ймовірності помилки виходячи з надлишковості кодування наведено на рисунку 2.

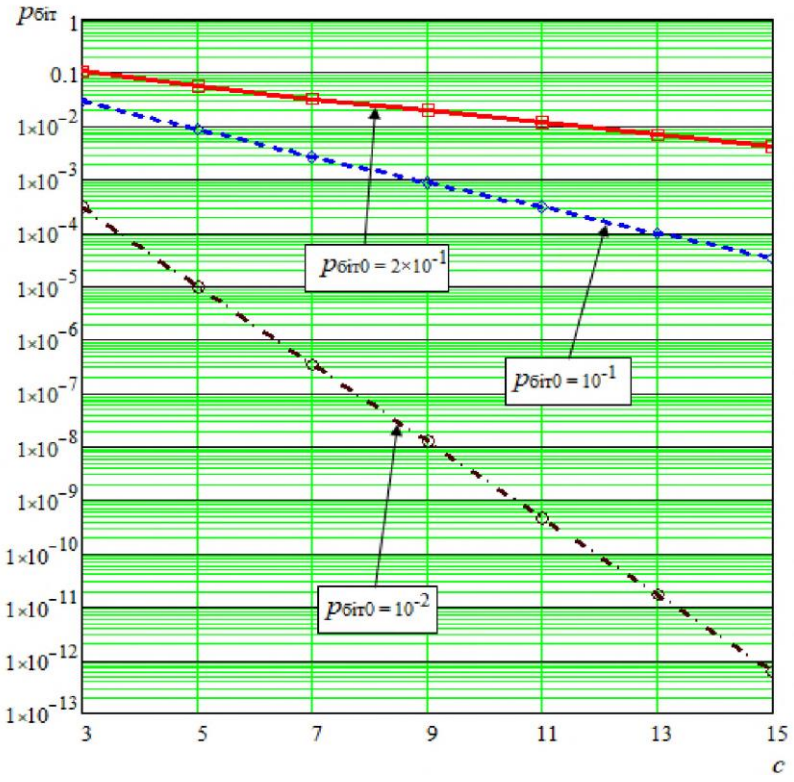


Рисунок 2 – Ймовірність $P_{біт}$ при зміні кратності кодування

Отже, застосування мажоритарного методу кодування забезпечує зростання вірності прийому повідомлення навіть в умовах критичних завад – при ймовірності бітового спотворення в каналі $P_{бітo} = [2x10^{-1}; 10^{-1};10^{-2}]$.

Однак, при використанні мажоритарного способу пропорційно до числа повторів зростає і надлишковість повідомлень. Саме тому його застосування обмежується часовими вимогами передачі [5].

В залежності від області застосування БпЛА, для його систем – телеметрії, навігації, управління борта, окрім завдання підтримки високої вірності прийому, може стояти задача прихованої передачі даних. Серед принципів, які дозволяють виконувати ці вимоги, є застосування комбінованого випадкового кодування (КВК). Воно поєднує завадостійке кодування з стохастичним – псевдовипадковою

зміною ансамблю комбінацій. Таким чином, визначення словника кодових комбінацій стає нетривіальною задачею.

Схематично перетворення повідомлення методом КВК зображено на рисунку 3. Завадостійке кодування застосоване ззовні, а стохастичне – всередині. Кодування слова a_i, його подальше відновлення та корекція помилок під час приймання проходить за два етапи.

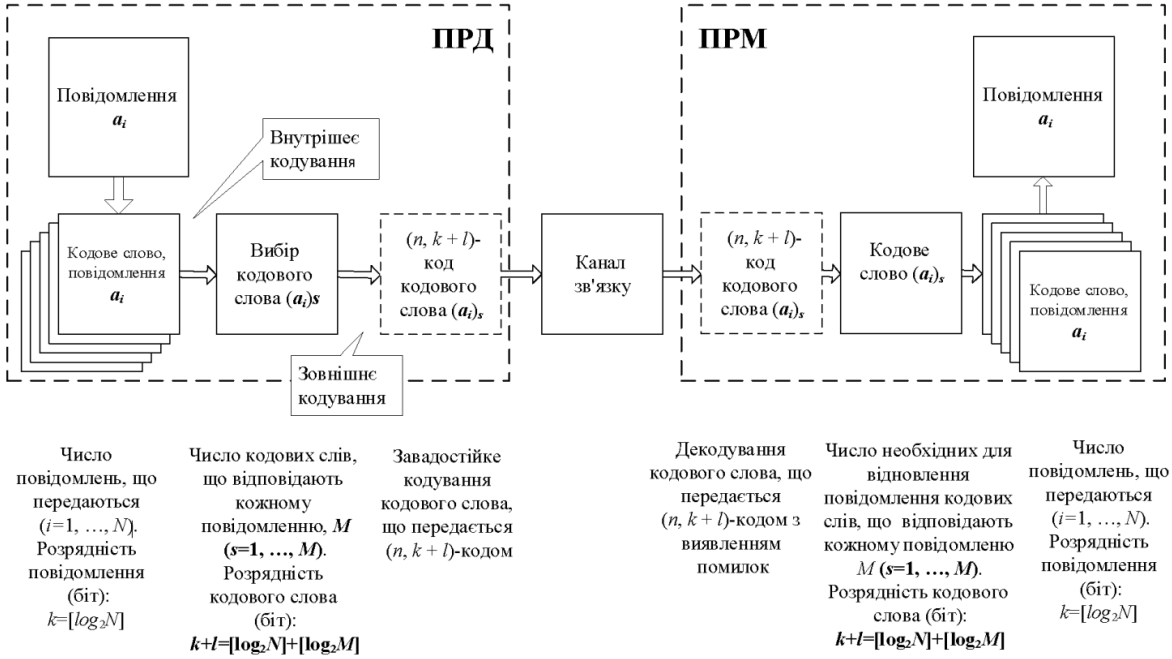


Рисунок 3 – Передача даних методом КВК

Вихідне повідомлення a_i має розрядність k = [1og$_2$N], а числом розрядів комбінації, що передається $(a_i)_s$ є k + l, де l = [log$_2$M]. Першим здійснюється стохастичне кодування. Це відбувається шляхом формування M кодових слів $(a_i)_s$ за кодовою книгою. Скориставшись генератором псевдовипадкових послідовностей (ПВП) обирається s-те слово, яке відповідає повідомленню a_i. Далі відбувається кодування завадостійким кодом [6]. З допомогою блочного коду (n, k + l), що виправляє помилки, підготовлене повідомлення передається в канал зв’язку.

Під час прийому даних декодується блочний код з виправленням помилок. Потім виділяється кодове слово $(a_i)_s$, за цим словом з книги кодування береться

відповідне повідомлення. Тому передавач і приймач мають бути забезпечені однаковими кодовими книгами, а для зловмисник вона невідома.

Стохастичне кодування із застосуванням кодової книги зображено на рисунку 4. Ансамбль повідомлень сформовано за дискретними повідомленнями від джерела, кожне довжиною k. Їх число $N = 2^k$ становить об'єм ансамблю. Надалі повідомлення випадково отримують відповідні їм $M = 2^l$ слів, які зазначені у окремому рядку кодової книги. Загальне число слів кодової книги $K = MN = 2^{k+l}$.

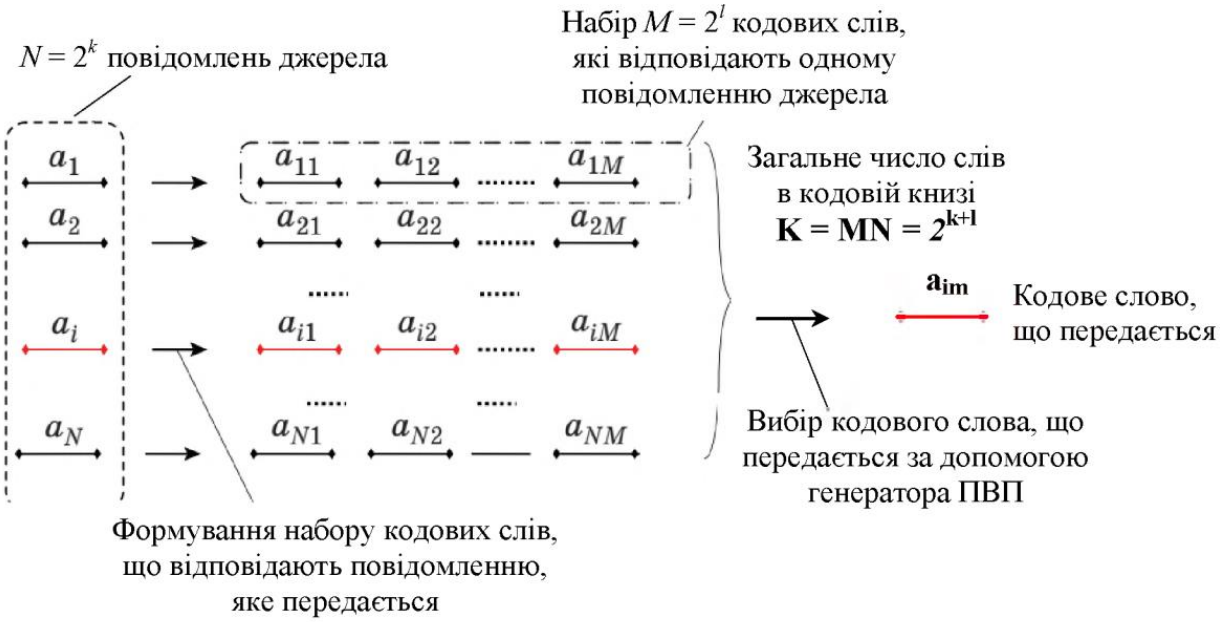


Рисунок 4 – Стохастичне кодування з кодовою книгою

Відповідну стохастичну кодову комбінацію V можна подати у вигляді лінійного коду, який утворено множиною двійкових комбінацій V_i. Для цих послідовностей справедливим є співвідношення $V = \cup_{i=1}^{N} V_i,\ V \cap V_j = \emptyset,\ i \neq j$. Кожна з підмножин V_i, що вміщає M кодових слів $(a_i)_s$ розрядністю M(k + l) співвідноситься зі своїм повідомленням a_i. Обране випадково, з однаковою ймовірністю слово передається для кодування коригуючим кодом, а надалі – в радіоканал.

Поєднавши стохастичне і завадостійке кодування, вірогідність бітової помилки несуттєво зростає, що також зменшує швидкість передачі в (n - r)/(n - r -

l) раз, де n - розрядність повідомлення, r - кількість перевірочних розрядів, l - число символів стохастичного коду. Проте, інформаційна скритність передачі даних зростає.

Коди Боуза-Чоудхурі-Хоквінгема (БЧХ) це об'ємний різновид кодів, які спроможні виявляти і виправляти декілька помилок. Коди БЧХ є цікавими, оскільки, маючи невеликі за довжиною коди ($n < 511$), забезпечують гарну завадостійкість. Як в теорії, так і при практичних реалізаціях їм належить помітне місце, тому їх вигідно застосовувати, якщо постає питання підвищення достовірності прийому [6].

Популярність використання зумовлює різні методи їхнього кодування та декодування, що дозволяє обирати потрібний підхід для кожного випадку. Коригувальні здатності циклічного коду можна описати наступним твердженням: при необмежених значеннях m та $t_в$ існує код, довжина якого $n = 2m - 1$, що виправляє помилки, кратні $t_в$ і менше ($t_в < m$) та має не більше $n - k \le m \cdot t_в$ перевірочних розрядів. Породжуючий поліном $G(x)$, що бере участь у створенні коду с параметрами (n, m) і $d_{БЧХ} = 2t_в + 1$:

$$G(x) = НСК[\mu_1(x),\mu_3(x),\mu_5(x),\ldots,\mu_{БЧХ}-2\ (x)],$$

де *НСК* – найменше спільне кратне, $\mu_i\ (x)$ – мінімальні поліноми елементів β_i, тобто коренів, що мають мінімальний ступінь, $d_{БЧХ}$ – мінімальна кодова відстань (у випадку БЧХ $d_{БЧХ} \le d_0$).

Перш за все, при виборі коду потрібно визначити породжуючий поліном $G(x)$, що від нього залежить здатність коду виправляти помилки. Для побудови коду необхідно мати довжину кодової комбінації n та кратність помилок, які треба виправляти. Також, число інформаційних розрядів k буде залишатись невідомим до вибору породжуючого багаточлена. Визначивши добуток мінімальних поліномів, що мають непарні індекси, можна отримати утворюючий поліном:

$$G_{БЧХ}\ (x) = \mu_1(x),\mu_3(x),\mu_5(x),\mu_7(x),\mu_{11}(x),\mu_{15}(x)\ .$$

Код, при конструюванні якого були обрані занадто великі величини $t_в$ або $d_{БЧХ}$ буде мати один перевірний розряд $r > n$ або будуть відсутні інформаційні розряди. БЧХ поділяються на короткі коди ($n < 127$) і довгі коди ($n > 127$). Таблиці 2 і 3 містять характеристики і породжуючі багаточлени деяких кодів.

Таблиця 2 – Параметри та породжуючі багаточлени деяких кодів БЧХ

m	*n*	*k*	*r*	$t_в$	*G(X) – mod 8*
3	7	4	3	1	13
4	15	11	4	1	23
		7	8	2	37
5	31	26	5	1	45
		21	10	2	75
		16	15	3	67
		11	20	5	57
6	63	57	6	1	103
		51	12	2	127
		45	18	3	147
		39	24	4	111
7	127	120	7	1	211
		113	14	2	217
		106	21	3	235
		99	28	4	367
8	255	247	8	1	435
		239	16	2	567
		231	24	3	763
		223	32	4	551

Таблиця 3 – Характеристика кодів БЧХ довжини 31

n	*k*	*r*	*G(x)*	*G(x) mod 8*	$d_{БЧХ}$	d_0	$t_{вп}$
31	26	5	$G_1(x)= \mu_1(x)$	45	3	3	1
	21	10	$G_2(x)= \mu_3(x) \times G_1(x)$	75	5	5	2
	16	15	$G_3(x)= \mu_5(x) \times G_2(x)$	67	7	7	3
	11	20	$G_4(x)= \mu_7(x) \times G_3(x)$	57	11	11	4-5
	6	25	$G_5(x) = \mu_{11}(x) \times G_4(x)$	73	15	15	6-7

Представником коротких БЧХ є циклічний код (n = 63, k = 39) і $d_0 = 9$, що застосовується у системі супутникового зв'язку INMARSAT. Багаточлени наведені у вісімковій системі числення, старший ступінь знаходиться ліворуч.

У випадку довжини коду $2m - 1$, він є примітивним кодом. До примітивних зараховують коди Хемінга, що корегують одиночні помилки. Коди, з довжиною, що являється дільником $2m - 1$ теж є кодами БЧХ.

Кодова відстань примітивних кодів становить $d_{БЧХ} = 2t_в + 1$. Кодова відстань представників БЧХ, наведених у таблицях 2 і 3, збільшується шляхом введення мінімального полінома $\mu_0(x) = x + 1$ ($G1(x) = \mu_0(x) \cdot G_{БЧХ}(x)$) під час обрахунку $G(x)$. Джерелом даних для конструювання кодів слугує таблиця 3. Так отримують породжуючі поліноми для будування примітивного коду, що має довжину $n = 2m - 1$, більшим степенем $G(x)$ і корегувальною здатністю [7].

Скорочені циклічні коди. Через те, що циклічні коди породжені дільниками бінома $x^n + 1$, для більшої частини значень k та n існує досить незначна кількість кодів, що матимуть всі ознаки циклічних. Скорочений код отримують додаючи l нулів зліва до кодової послідовності повного циклічного коду. Виключивши з породжуючої матриці перші l стовпців і рядків, будується скорочений код ($n - l$, $k - l$). Такий код не можна твердо назвати циклічним, бо в результаті циклічного зсуву не кожного разу будемо мати чергову дозволену комбінацію. Відповідно, скорочені коди ще отримали назву псевдоциклічних або квазіциклічних. В них збережені суттєві ознаки повних циклічних кодів:

1. Скорочений код утворюється дільниками бінома $x^n + 1$ – породжуючими багаточленами $G(x)$, які застосовані у класичних кодах;

2. Скорочений код зараховують до групових кодів, тобто підсумувавши його дозволені комбінації можна отримати дозволену кодову комбінацію;

3. З вихідного коду скорочений код отримує його кодову відстань і кількість перевірочних розрядів;

4. Скорочений код має такі самі коригуючі властивості як і повний код;

5. У проектуванні кодеків скорочених кодів застосовуються ті самі схеми, як для класичних кодів, з умовою що для скороченого коду зліва дописуються *l* нулів.

Алгоритм кодування циклічних кодів БЧХ. У більшості цих кодів використовується єдина послідовність дій для генерації кодової послідовності, але спосіб обирання породжуючого полінома відрізняється. Конструкція породжуючого полінома сильно покладається на 2 умови: величина комбінації n та кількості помилок, які корегуються кодом $t_в$. Зокрема, для коду БЧХ корекція $t_в \geq 2$ потребує не тільки мінімальної кодової відстані між комбінаціями $d_{\min} = 2t_в + 1$, але і з довжиною коду n має виконуватись умова $n = 2^h - 1$, де h – довільне ціле число.

В даному виразі n завжди приймає значення непарного числа: 1, 3, 7, 15, 31, 63, 127 та далі, відповідно не кожне m можна використовувати будуючи код. Кількість контрольних розрядів r виходить з довжини n:

$$r \leq h \cdot t_в \leq t_в \cdot \log_2 (n+1).$$

Допустима кількість інформаційних розрядів обирається за відомою умовою коригуючої здатності коду. Для цього в нагоді стає таблиця з наведеними параметрами коду БЧХ – перевірними та інформаційними розрядами. Породжуючий поліном є найменшим спільним кратним непарних мінімальних поліномів. Найбільший допустимий порядок мінімальних поліномів:

$$p = 2t_в - 1.$$

Порядок багаточленів використовується для визначення кількості співмножників. Для того, що побудувати $G(x)$ застосовують виключно непарні поліноми. При утворенні породжуючого полінома задіяні $V = t_в$ мінімальних поліномів, при цьому максимальний ступінь:

$$v = h.$$

посилається на рядок таблиці, котра використовувалась для вибору мінімального полінома для формування $G(x)$. Беруть участь тільки непарні

поліноми ступенем v, рідше є допустимими поліноми й менших ступенів. Відповідний ступінь $G(x)$ знаходиться за формулою:

$$b = r = v \cdot t_в = h \cdot t_в .$$

Загальний вигляд – $G(x) = НСК[\mu_1(x),\mu_3(x),\mu_5(x),\ldots,\mu_{БЧХ}-2(x)]$.

1.3 Постановка задачі проектування

Безпілотні літальні комплекси знаходять все більше і більше розповсюдження, особливо в військовій сфері. Відповідно виникає загроза перехоплення керування зловмисниками. На сьогодні передавачі телеметричних даних безпілотних літальних апаратів є поєднанням трансивера і процесора, що ними керує.

За умови достатньої швидкодії процесора завадостійке кодування і шифрування даних можна здійснювати різними алгоритмами шляхом програмної реалізації або апаратно.

З метою забезпечення високонадійного зв'язку безпілотними літальними комплексами та зменшення ризику перехоплення керування в даній кваліфікаційній роботі пропонується до розв’язку наступні задачі:

– провести порівняльний аналіз алгоритмів шифрування даних для їх ефективного застосування в телекомунікаційних пристроях захисту телеметричної інформації бортової апаратури дронів та пристроїв їх керування;

– провести вибір методів завадостійкого кодування для боротьби з шумоподібними завадами в каналах зв'язку безпілотних літальних апаратів;

– побудувати структуру та загальний алгоритм проектованого телекомунікаційного пристрою захисту даних для безпілотних літальних комплексів;

– розробити функціональні схеми високонадійних шифратора та кодера даних телеметрії та керуючих сигналів для проектованого телекомунікаційного пристрою захисту даних.

Основними вимогами до технічних характеристик блоків шифратора та кодера телекомунікаційного пристрою захисту даних є наступні:

– довжина блоків і ключів застосованого блокового шифру: 128 біт, 192 біт або 256 біт;
– виконання ітерації алгоритму шифрування шляхом використання таблиць пошуку та операцій XOR;
– кількість блоків модульної пам'яті не менше 3, обсяг пам'яті кожного з яких не повинно перевищувати 2 Кбіт;
– пропускна здатність шифрування не менше 250 Мбіт/с;
– виявлення пакетних помилок та їх корегування за допомогою БЧХ-кодування.

2 АЛГОРИТМ РОБОТИ ПРИСТРОЮ

Пристрій працює наступним чином. На початку відбувається встановлення блоків у вихідний стан. Після надходження даних, які потрібно зашифрувати, вони поділяються на масиви з 16 байт $in_0, in_1, \ldots in_{15}$, потім байтами масиву послідовно заповнюють стовбці матриці *InputBlock* розмірами 4x4 (зверху вниз). Алгоритм, за яким працює пристрій наведено на рисунку 5.

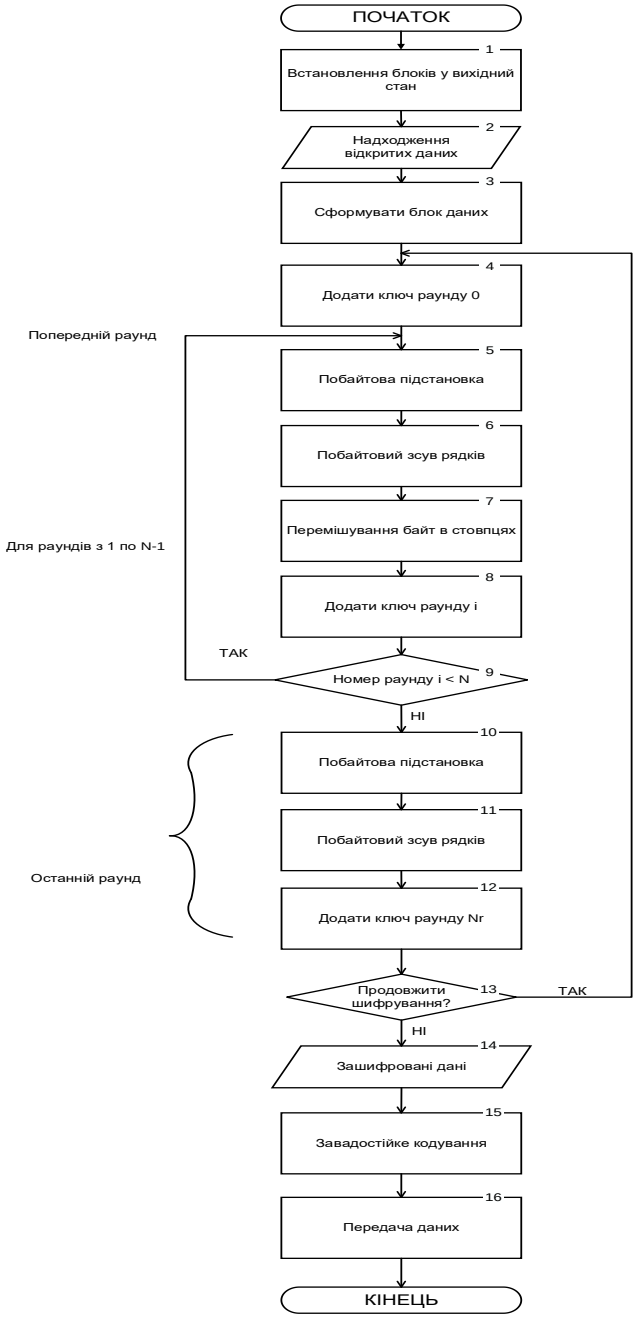


Рисунок 5 – Алгоритм роботи пристрою

Також у процесі застосовується матриця станів (*State*) яка має такі ж розміри, як і матриця вхідних даних. В кінці шифрування, перетворену матрицю станів називають матрицею *OutputBlock* яка в свою чергу стає послідовністю байт вихідних даних.

Дані ключа шифрування також поміщаються в матрицю 4x4 *InputKey*. Перетворення матриць схематично зображено на рисунку 6.

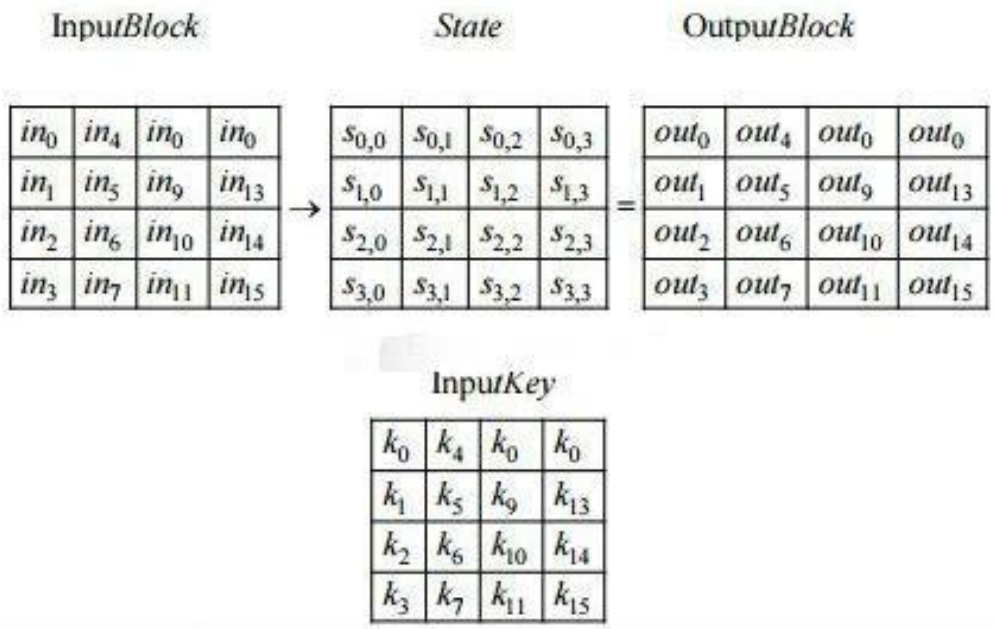


Рисунок 6 – Відношення між матрицями

Ключ в алгоритмі *AES*-128 складається з 128 бітів, що поділяються на 16 байт k_0, $k_1 ... k_{15}$, а потім записуються в стовбці матриці *InputKey*. В результаті ключем є чотири слова $w_0 w_1 w_2 w_3$ де $w_0 = k_0 k_1 k_2 k_3$.

За алгоритмом, з цих слів формується послідовність вже з 44 слів ($w_0 ... w_{43}$), надалі кожного раунду шифрування подається чотири слова описаної послідовності [8].

Кожний раунд *AES*-128 складається з чотирьох різних перетворень:

- *SubBytes* – побайтова підстановка в матрицю станів (S-бокс) з фіксованої таблиці замін;

- *ShiftRows* – побайтовий зсув рядків у матриці станів на різну кількість байт;
- *MixColumms* – перемішування байт в стовбцях;
- *AddRoundKey* - додавання з раундовим ключем (*XOR*).

В останньому раунді функція *MixColumms* не застосовується.

Зашифровані дані передаються до кодуючого пристрою. Кодування відбувається за схемою кодера БЧХ з попереднім множенням на величину x^k – без елемента затримки.Функції збудження елементів пам'яті:

$$D_i^t = D_{i-1}^{t-1} \oplus g_i \cdot (D_k^{t-1} \oplus u^t), i = \overline{1..k-1}$$

$$D_0^t = u^t \oplus g_0 \cdot D_{k-1}^{t-1} = u^t \oplus D_{k-1}^{t-1}$$

На перших m тактах зворотний зв'язок активний – ключ $к_1$ відкритий, і закривається під час наступних k тактів, щоб перервати зворотний зв'язок. Ключ $к_2$ у закритому положенні протягом m початкових тактів, щоб розімкнути регістр і вихід, на наступних тактах $к_2$ відкривається для видачі перевірної частини до каналу зв’язку.

Перемикання відбувається за сигналом керування аналогічно до попередньої схеми. Надлишкові розряди помноженого на x^k інформаційного полінома обраховуються впродовж перших m тактів роботи схеми

Після кодування повідомлення передається системі радіозв’язку.

3 ПОРІВНЯННЯ АЛГОРИТМІВ ШИФРУВАННЯ

Широке застосування безпілотних літальних апаратів (БпЛА) на сьогодні дозволяє виконувати складні і важливі завдання не задіюючи більшу кількість людей, ресурсів і засобів. Доставка вантажів, пошук і порятунок людей, фотографування і моніторинг місцевості, застосування в бойових умовах. З поширенням БпЛА постало питання забезпечення надійного і конфіденційного зв’язку.

Безпілотні літальні апарати різняться за розмірами, вагою і обладнанням, яке включає корисного навантаження на борта. Тому існують кілька способів захисту даних, що передаються в системі зв'язку борта, один з них – використання алгоритмів шифрування.

Шифруванням називають процес перетворення оригінального повідомлення (яке називають відкритим текстом) або даних з метою отримати зашифровані дані, що будуть недоступні для зчитування зловмиснику, тобто зашифрований текст не буде становити для нього жодної користі.

На сьогодні існує два типи криптографічних алгоритмів – симетричні та асиметричні. Симетричне шифрування застосовує однакові ключі для шифрування і дешифрування, або перетворює один в інший.

При асиметричному шифруванні ці ключі є різними і ніяк не пов’язаними [9]. Через те, що цей метод є більш складним і повільнішим в порівнянні з симетричними алгоритмами, його не доцільно використовувати в контексті БпЛА.

В свою чергу симетричні алгоритми потребують додаткового убезпеченого каналу зв’язку для обміну ключами шифрування, а асиметричне шифрування – ні. Обраний метод безпосередньо впливатиме на захист даних, тому дуже важливо обирати надійні алгоритми, оглядаючись на доступні технічні засоби.

3.1 Симетричний блоковий криптоалгоритм DES

Стандарт шифрування даних DES – це блоковий криптоалгоритм, що використовує симетричні ключі і був розроблений Національним Інститутом Стандартів та Технології США в 1977 році. Він опрацьовує відкритий текст довжиною 64 біти й перетворює його в зашифрований текст такої ж довжини. З метою шифрування та дешифрування алгоритм користується однаковим ключем довжиною 56 біт. Один раунд шифрування зображено на рисунку 7.

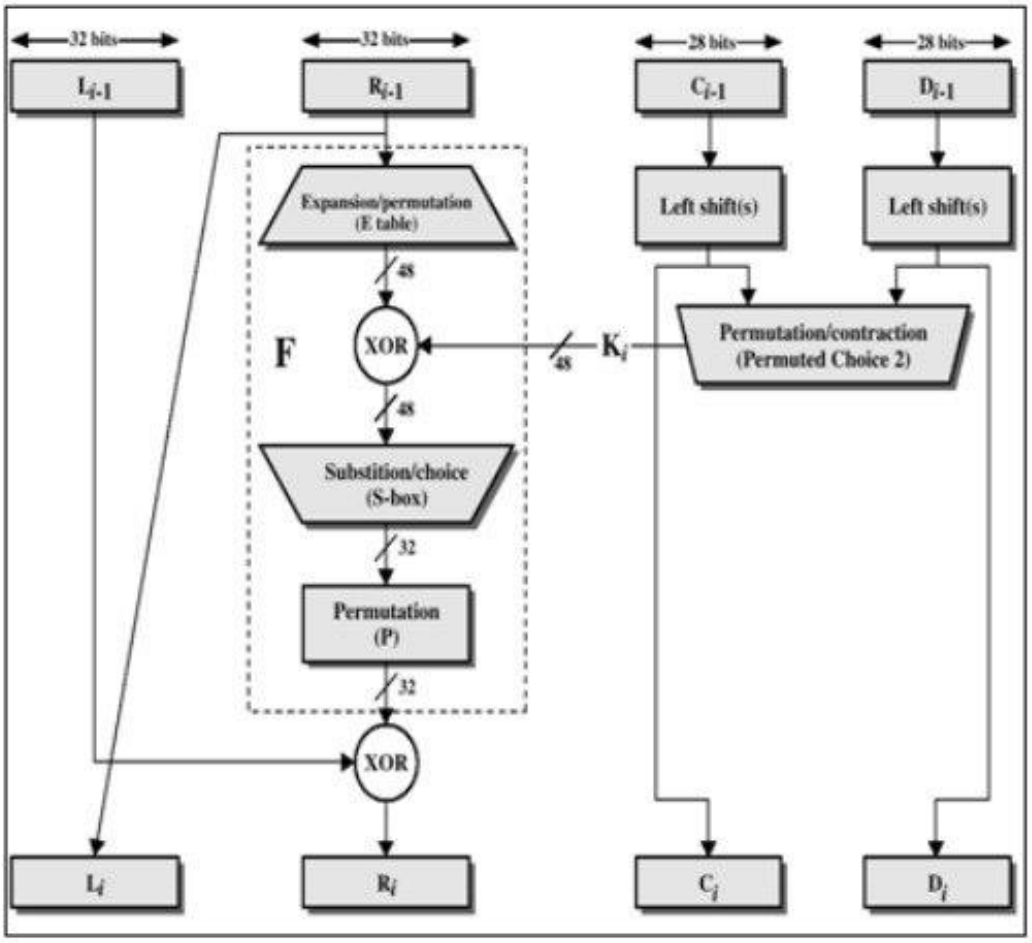


Рисунок 7 – Раунд шифрування DES

Сам процес шифрування включає дві взаємно обернені перестановки – початкову та фінальну, а також 16 раундів Фейстеля. На початку перестановки 64 біти відповідно до спеціальних таблиць (box) зміщуються. При цьому кожного

раунду Фейстеля генеруються різні 48 бітові ключі. Таким чином, 58-й біт під час початкової перестановки стане на першу позицію, а під час фінальної – перший вхідний біт потрапить на 58 позицію. DES використовує 48-бітовий ключ, щоб зашифрувати 32 правих біти, на виході отримавши результат довжиною 32 біти [9].

Для цього процесу алгоритм користується чотирма операціями: XOR, група S-box, P-таблиця розширення та пряма таблиця. DES утворює 16 раундових ключів k^i довжиною 48 бітів кожен із загального ключа k на 56 біт.

Проте, задання ключа вимагає додатково вписати 8 біт на позиції 8, 16, ..., 64 серед його 56 бітів, з метою перевірити парність і щоб кожен байт мав непарну кількість одиниць. Дана операція потрібна щоб знайти помилки після обміну або зберігання ключа.

Найсуттєвіший недолік алгоритму це довжина ключа (56 бітів). Для здійснення атаки повного перебору необхідно перевірити 2^{56} ключів. У випадку створення обчислювальної машини з мільйоном крипточипів, навіть таке число ключів буде перебрано за 20 годин. DES зламали 1998 року, протягом 56 годин проводилася атака компанією Electronic Frontier Foundation, вони застосували спеціально створений комп'ютер DES Cracker.

3.2 Симетричний блоковий криптоалгоритм CAST 128

CAST-128 (також CAST5) є алгоритмом, що реалізує блокове шифрування, має популярні програмні реалізації, наприклад PGP та GNU Privacy Guard. CAST-128 був ухвалений Відомством безпеки зв’язку для застосування в комунікаціях всередині уряду Канади.

Алгоритм здійснює шифрування даних блоками довжиною 64 біт, застосовуючи для цього ключі розмірами 40-128 біт (кожен ключ через 8 біт), а

також 40, 64, 80 та 128 біт [10]. Блок-схему CAST-128 і окремого раунду наведено на рисунку 8.

CAST являється мережею Фейстеля, що послідовно проводить 12 або 16 раундів перетворень залежно від розмірів ключа:

- довжина ключа до 80 біт – 12 раундів;
- довжина від 80 біт – 16 раундів.

В алгоритмі передбачено розширення розміру ключа. На початку формуються 32 підключі довжиною 32 розряди, половина з них стають підключами-масками K_{mi}, а інші 16 підключами для зсуву K_{rt} (лише 5 молодших біт задіяні). Обов'язковим етапом є доповнення ключа нулями, якщо він коротший за 128 біт.

Процедура є складнішою у порівнянні з CAST-256: у CAST-128 присутні 8 таблиць заміни (а не 4), половина яких застосовуються тільки при розширенні ключа. Ця обставина значно збільшує вимоги алгоритму щодо енергонезалежної пам'яті.

Кістяк алгоритму становлять операції додавання, віднімання за модулем 2 ** 32, виключної диз'юнкції і циклічний зсув вліво. Дешифрування аналогічне процесу шифрування, проте ключі раундів йдуть у зворотному порядку.

CAST-128 належить Entrust Technologies, але є безкоштовним як для комерційного, так і для некомерційного використання. CAST-256 це безкоштовне розширення CAST-128 доступне кожному. CAST-256 приймає 256-бітний розмір ключа і має 128-бітний розмір блоку. CAST-256 був одним з перших кандидатів на конкурсі AES.

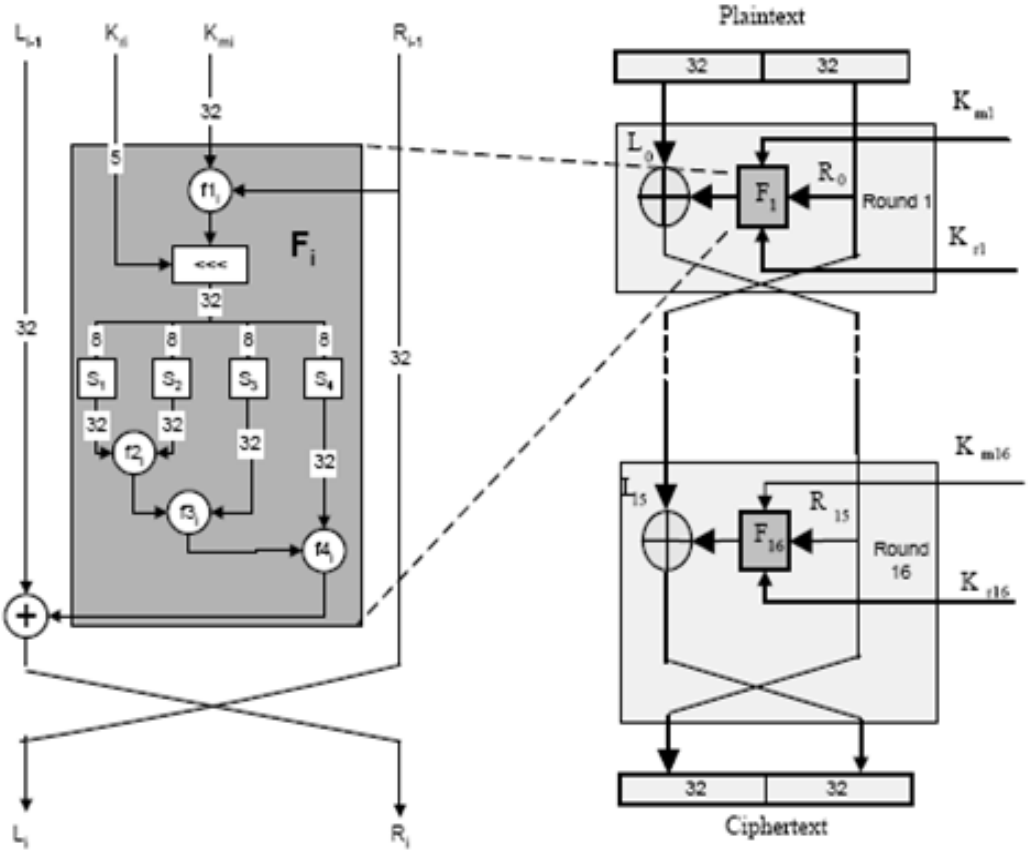


Рисунок 8 – Алгоритм CAST-128

Єдиний на сьогодні відомий метод злому шифрування цього алгоритму це атака перебору, що вимагає значних витрат часу. Проте, модифіковану версію CAST було успішно атаковано, застосовуючи диференціальний криптоаналіз.

3.3 **Симетричний блоковий криптоалгоритм Blowfish**

Blowfish є криптографічним алгоритмом, що реалізує блокове симетричне шифрування з перемінним розміром ключа. По суті алгоритм застосовує кілька нескладних та швидких операцій – це підстановка, XOR та додавання (рисунок 9). За кроками алгоритм подібний до попереднього – спочатку розширення ключа, потім шифрування даних. Першим здійснюється збільшення загального ключа (довжина < 448 біт), він поділяється і трансформується в 18 підключів обсягом 32

біт та 4 таблиці S-box обсягом 32 біт, що мають 256 елементів. В результаті маємо загальний розмір ключів 33344 біт/4168 байт.

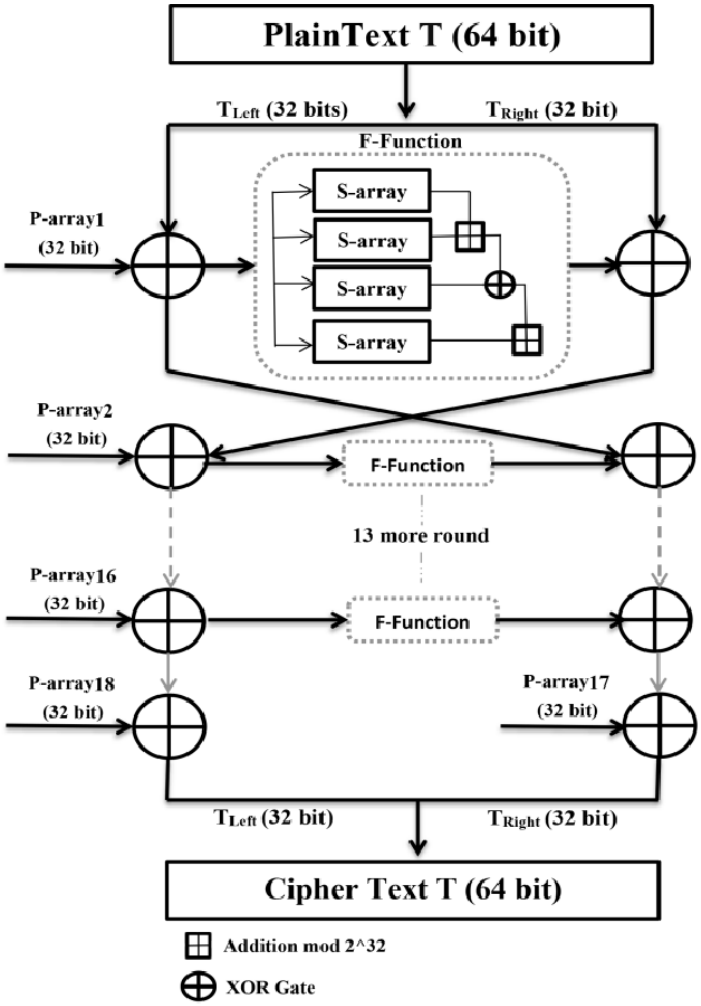


Рисунок 9 – Алгоритм Blowfish

Перший етап здійснює підготовку (формуються ключі зашифровки за секретним ключем) і перебігає наступним чином. Масиви *P* і *S* ініціалізуються із застосуванням секретного ключа *K* (якщо ключ закороткий, він накладається циклічно:

- P1-P18 ініціалізуються фіксованим шістнадцятковим рядком;
- Перші 32 біти ключа K у P1 піддаються операції XOR, потім наступні 32 біти і т.д.

Надалі відбувається шифрування використанням ключів та таблиць замін:

- Скориставшись ініціалізованими ключами P1 - P18 і таблицями замін S1 - S4, шифрується нульовий рядок (0x0000000000000000) розміром 64. Підсумок поміщається в P1, P2.
- Змінені значення, що належать таблицям замін, і змінені ключі зашифровують P1 і P2. Значення поміщається в P3, P4.
- Процес триватиме до перетворення всіх ключів від P1 до P18 та таблиць до S4.

Другий крок являє собою шифрування повідомлення (ключами, які отримані на попередньому етапі та F(x)), розбитого на 64 бітні блоки. Передбачаються різні режими шифрування для створення повідомлення, число блоків котрого буде цілим на випадок, що початкові дані неможливо розділити на блоки по 64 біт. Загальна пам'ять складає 4168 байт.

3.4 **Симетричний блоковий криптоалгоритм AES**

Розширений стандарт шифрування це алгоритм симетричного блокового шифрування (довжина блока 128 біт, ключі 128/192/256 біт). Перша назва алгоритму – Rijndael, після обрання його в якості стандартного Державним інститутом стандартів і технологій, він став відомий як AES. Виходячи з фіксованого розміру блоку, алгоритм оперує із масивом даних 4×4 байт, який називають *станом* (у версіях AES з більшим розміром ключа блоки мають додаткові колонки). У випадку довжини ключа 128 біт алгоритм здійснить 10 раундів, у яких послідовно виконає операції: *SubBytes*, *ShiftRows*, *MixColumns* (окрім десятого раунду) а також *AddRoundKey* [11].

Оцінюючи безпечність шифру можна зазначити, що він демонструє високу стійкість проти атак диференціального та лінійного криптоаналізу; аналізу за пов’язаними ключами (слабкі ключі відсутні). Дієвим методом злому являються атаки, які застосовують побічні канали, що не є зв’язаними з математичними

характеристиками алгоритму. Вони користуються технічними вразливостями реалізації систем, які застосовують шифр, і переслідують мету отримати повністю або частково ключ шифрування.

Реалізація алгоритму програмно на частоті 2 ГГц дозволить шифрувати дані зі швидкістю до 400 Мбіт/секунду, апаратні реалізації демонструють більшу швидкість, що більш ніж достатньо для зашифровки в реальному масштабі часу відеоданих формату MPEG-2[12]. Відносно новою є версія AES-NI (додавання нових інструкцій до процесорів). Вона дозволяє оптимізувати роботу алгоритму, знижуючи навантаження на процесор і прискорюючи обрахунок раундів, а також зменшити ризик атак за побічними каналами. Перші реалізації належать компанії Intel – чіп серії X5600.

3.5 **Порівняння криптоалгоритмів за визначеними критеріями**

Враховуючи, що число алгоритмів шифрування, що можуть бути застосовані в пристрої захисту даних БпЛА є значним, є сенс проведення порівняльного аналізу. Подібно до конкурсу відбору нового стандарту шифрування на заміну DES, скористаємось критеріями, щоб облегшити процес вибору (критерій перший – К1 і т.д.). Пропоновані критерії є:

1) блок даних, що шифруються, розміром 128 біт*;*

2) алгоритм є стійким проти криптоаналітичних атак, що застосовується на сьогодні*;*

3) шифрування за алгоритмом підтримує ключі мінімум трьох розмірів: 128/192/256 біт*;*

4) швидкість шифрування має бути високою при апаратній реалізації на різноманітних платформах, починаючи з 8 бітних, закінчуючи 64 бітними реалізаціями*;*

5) відсутність слабких та еквівалентних ключів (які відрізняються, проте ведуть до однакового результату зашифровки);

6) алгоритм повинен мати неускладнену, зрозумілу та обґрунтовану структуру, щоб забезпечити відсутність вкладених розробниками особливостей архітектури, які можна застосувати у зловмисних цілях);

7) алгоритм має мінімальні потреби в оперативній та енергонезалежній пам'яті;

8) алгоритм має застосовуватись при різних стандартизованих режимах роботи, виступати основою в генераторах псевдовипадкових послідовностей, побудові хеш-функцій. Аналіз алгоритмів наведено у таблиці 4.

Таблиця 4 – Критеріальний аналіз криптоалгоритмів

Критерії / Алгоритми	**К1**	**К2**	**К3**	**К4**	**К5**	**К6**	**К7**	**К8**
AES	+	+	+	+	+	+	+	+
DES	-	-	-	-	-	+/-	+	+
CAST 128	-	+	-	+	+	+	+	+
Blowfish	-	-	+	-	-	+	+	+

Порівняння вказує, що DES в обох режимах роботи забезпечує потужний лавинний ефект, алгоритми AES і CAST мають значну зміну терміну перевірки цілісності у порівнянні з іншими алгоритмами. Виходячи із дослідження, можна зробити висновок, що алгоритм AES є найбільш оптимальним. Використання алгоритму AES в якості стандарту шифрування зумовлює велику кількість публікацій на тему його реалізації: на базі програмованої логічної інтегрально схеми; варіації ПЛІС, поєднаної з ядром мікроконтролера, або програмна реалізація на ARM-процесорах. Достатня швидкість шифрування і можливість гнучко обирати потрібну реалізацію під окреме завдання є суттєвою перевагою AES.

4 СТРУКТУРНА СХЕМА ПРИСТРОЮ

4.1 Структурна схема та опис елементів БпЛА

Основними блоками і системами, що входять до складу безпілотного літального апарату являються (рисунок 10):

Система силової установка (Двигун). Надмалі та малі БпЛА зазвичай оснащені електродвигунами. Це забезпечує кращу надійність і безпеку. З огляду на температуру середовища та погоду вони можуть ефективно працювати протягом 3-4 годин. Електричний двигун зумовлює меншу вагу, яка не тільки покращує їх керованість і маневреність, але і функцію амортизації при посадці, що, в свою чергу, подовжує їх термін експлуатації.

Подібні дрони типово застосовують літій-полімерні батареї. Акумульована енергія, звичайно, менша ніж у двигунів внутрішнього згорання, натомість електродвигуни дешевше виробляти, також вони мають меншу вагу і рівень шуму під час експлуатації.

Рисунок 10 – Структурна схема безпілотного літального апарату

Електронна (комп'ютерна) система управління. Система управління функціонує завдяки бортовому комп'ютеру. Метою системи управління є керування роботою всіх систем БпЛА: отримання даних датчиків, що встановлені на дрон; в залежності від режиму роботи – обробка цієї інформації, визначення маршрутів, контроль за здійсненням польоту, передачею інформації на пункт управління.

Програмним забезпеченням, яке застосовується є комерційні-приватні протоколи комунікації та операційні системи: відкриті типу Linux або сучасні спеціалізовані системи реального часу: VxWorks, QNX, VME, XOberon. Популярними є схемотехнічні рішення, коли бортовий комп'ютер та основні системи мають єдину плату, що розміщена у єдиному корпусі [13]. Процесори, що застосовуються при конструюванні бортового комп'ютера, мають зменшений набір команд типу RISC ARM-архітектури, подібно до мобільних телефонів, смартфонів, планшетів.

Система зв'язку «телеметрії». Телеметрія безпілотників відноситься до процесу збору і передачі даних у реальному часі від безпілотника до віддаленого оператора або наземної станції. Ці дані включають інформацію про висоту, швидкість, місцезнаходження, термін батареї та інші важливі параметри польоту безпілотника.

Дані телеметрії зазвичай передаються через радіочастотні або стільникові мережі і можуть відображатися на панелі панелі або панелі управління для моніторингу. Ця інформація має вирішальне значення для забезпечення безпечних і ефективних операцій безпілотників, оскільки вона дає змогу операторам ухвалювати обґрунтовані рішення і швидко реагувати на будь-які питання, що виникають під час польоту. Крім того, дані телеметрії можуть використовуватися для аналізу після польоту та оптимізації продуктивності дронів.

В ситуації втрати зв’язку з пультом управління, безпілотник вмикає автоматичне пілотування. Складність режиму може різнитись в залежності від програмних установок борта – починаючи командами «повернення», «політ по прямій», «баражування», закінчуючи автоматичним польотом за маршрутом, що оперує електронними картами, модифікованими даними навігаційної системи.

Навігаційна система. Як правило, проста навігаційна система, що встановлюється на надмалі і малі БпЛА, використовує обробляє сигнали в інтегрованому режимі від різних супутникових радіонавігаційних систем, точність позиціонування в горизонтальній площині і на висоті складає до 2,5 м.

Складні безпілотники оснащені елементами автоматизованих навігаційних систем, такими як акселерометри, гіроскопи, барометри та лазерні висотоміри.

Загальноприйнята точність авіаційної інерціальної навігаційної системи це точність, похибка котрої 1 км за 1,4 години польоту. Ця точність досягається за допомогою авіаційної навігаційної системи інтегрованої з лазерним або волоконно-оптичним гіроскопом. Але тут виникає проблема – їх маса є завеликою (від 8 кілограмів), що робить неможливим використання такої системи на малих бортах та ускладнює реалізацію на середніх.

Це призводить до того, що надмалі та малі апарати користуються простішими системами навігації, що включають механічні датчики руху (наприклад, гіроскоп та акселерометр)[14]. Така навігаційна система не зможе обчислити пройдений шлях без корекції супутникових систем, причиною цього є дрейф гіроскопа.

Отже, механічна навігаційна система в разі, якщо не надійшов супутниковий сигнал, що корегує, буде множити помилки зі швидкість до 3 метрів по горизонталі за 1 хвилину. Очевидно, що позиціонування системи буде на прийнятному рівні точності (до 130 м) протягом періоду до 10 хвилин [2]. Сам політ має виключати різкі прискорення та маневрування. Навігаційною системою, яка має такі характеристики є Geo-iNAV.

Подальшого покращення точності при гарному рівні незалежності розрахунків навігації можна досягти додаванням лазерного висотоміра в комбінації з барометром. Вони дозволять збільшити точність позиціонування шляхом додаткових засобів корекції даних, окрім того, вони дозволяють точніше працювати автопілоту борта за рахунок даних барометра або профілів висоти місцевості.

Корисне навантаження – відеокамера і відеопередавач. З огляду на завдання і функції, які має виконувати безпілотний апарат, його борт може нести корисне навантаження (пристрої чи системи), серед них:

- апаратура, що допомагає в процесі автопілотування і посадки;
- засоби, що дозволяють вести радіоелектронну боротьбу або встановлювати радіоелектронні завади;
- засоби іригації;
- кріплення, відсіки для транспортування грузу;
- системи аеророзвідки (оптичної, з застосуванням тепловізора, радіолокаційної, радіаційної, хімічної, інших різновидів);

Типовим навантаженням для дрона є система відеозв'язку: курсова FPV-відеокамера та відеопередавач, що забезпечує трансляцію з борта.

Система радіозв'язку. Система забезпечує зв'язок між бортом та пунктом управління на землі, здійснює перетворення і передачу інформації. Її завданням також є передача інформації корисного навантаження літального апарату, для цього вона повинна забезпечити передавання великих обсягів даних при заданих вимогах по смузі пропускання, ймовірності бітової помилки та інших. Розміри приймально-передавальної апаратури та антенно-фідерного обладнання мають бути мінімальними для вміщення їх в конструкцію борта.

Під час передачі даних систематично оцінюється ймовірність бітової помилки та зашумленість кожного каналу зв'язку і приймається рішення про розподіл потоку даних між каналами.

Пристрій захисту даних. Функції системи радіозв'язку щодо забезпечення вірності і конфіденційності передаваних даних покладено на пристрій захисту даних. Він включає пристрій завадостійкого кодування і шифратор інформації, реалізований на програмованій логічній інтегральній схемі.

4.2 Будова шифратора

Даний шифратор використовує архітектуру криптографічного алгоритму AES Rijndael, що заснована на трьох методах для покращення реалізації:

— Пошукова таблиця для полегшення впровадження.

— Цифровий тактовий генератор (Digital Clock Manage) для оптимізації часу виконання.

— Модулі пам'яті (двопортова оперативна пам'ять), щоб зменшити площу конструкції.

Запропонована архітектура займає відносно невелику площу та має менший час виконання у порівнянні з програмною реалізацією і може бути використана для широкого спектру застосувань [15].

Опис алгоритму AES Rijndael. Rijndael це блоковий шифр, що може оперувати з різними ключами і довжинами блоків: 128 біт, 192 біт або 256 біт. Кожного раунду надходять вхідні дані, які складаються з блоку повідомлень, так званого стану (State), і ключа раунду. Ключ раунду змінюється кожного раунду. Матрицю стану можна подати у виді прямокутного масиву байт. Масив має чотири рядки, а кількість стовпців позначається N_b і дорівнює довжині блоку, розділеній на 32. Те саме можна сказати і про ключ шифру, для нього число стовпців позначається N_k.

Кількість раундів N_r алгоритму залежить від довжини блоку та ключа. Раунди функції шифрування Rijndael становлять чотирьох різних послідовні

перетворення: SubByte, ShiftRow, MixColumn і додавання ключа AddRoundKey. Результатом кожного перетворення і є матриця «Стан»:

$$State = \begin{bmatrix} d_{15} & d_{11} & d_7 & d_3 \\ d_{14} & d_{10} & d_6 & d_2 \\ d_{13} & d_9 & d_5 & d_1 \\ d_{12} & d_8 & d_4 & d_0 \end{bmatrix}$$

Перетворення суббайтів. Перетворення SubByte являється нелінійною заміною байт, що замінює кожен байт стану незалежно. Операція виконується із застосуванням заздалегідь розрахованої таблиці підстановок, яка називається S-box [12]. Ця таблиця S-box містить 256 чисел (від 0 до 255) і відповідні їм результуючі значення. Перетворення застосоване до таблиці стану, можна представити наступним чином:

$$SB(State) = \begin{bmatrix} SB(d_{15}) & SB(d_{11}) & SB(d_7) & SB(d_3) \\ SB(d_{14}) & SB(d_{10}) & SB(d_6) & SB(d_2) \\ SB(d_{13}) & SB(d_9) & SB(d_5) & SB(d_1) \\ SB(d_{12}) & SB(d_8) & SB(d_4) & SB(d_0) \end{bmatrix}$$

Зворотне перетворення InvSubByte виконується аналогічно, за допомогою попередньо розрахованої таблиці InvS-box. Вона містить 256 чисел (від 0 до 255) і відповідні їм значення.

Трансформація ShiftRow. Операція ShiftRow циклічно зсуває рядки стану вліво на різні відстані. Рядок 0 не зсувається, рядок 1 зсувається на один байт, рядок 2 – на два байти і рядок 3 –на три байти. Таким чином, перетворення ShiftRow відбувається наступним чином:

$$SR(SB(State)) = \begin{bmatrix} SB(d_{15}) & SB(d_{11}) & SB(d_7) & SB(d_3) \\ SB(d_{10}) & SB(d_6) & SB(d_2) & SB(d_{14}) \\ SB(d_5) & SB(d_1) & SB(d_{13}) & SB(d_9) \\ SB(d_0) & SB(d_{12}) & SB(d_8) & SB(d_4) \end{bmatrix}$$

Під час зворотного перетворення InvShiftRow відповідні рядки стану циклічно зсуваються вправо на різні зсуви. Рядок 0 не зсувається, рядок 1 зсувається на один байт і так далі.

Перетворення MixColumns. MixColumns розглядає стовпці матриці стану як поліноми, помножені на фіксований поліном c(x), що задається:

$$c(x) = '03' x3 + '01' x2 + '01' x + '02' .$$

Перетворення можна записати у вигляді матричного множення наступним чином:

$$R = MC\left(SR\left(SB(State)\right)\right) =$$

$$= \begin{bmatrix} '02' & '03' & '01' & '01' \\ '01' & '02' & '03' & '01' \\ '01' & '01' & '02' & '03' \\ '03' & '01' & '01' & '02' \end{bmatrix} \otimes \begin{bmatrix} SB(d_{15}) & SB(d_{11}) & SB(d_7) & SB(d_3) \\ SB(d_{10}) & SB(d_6) & SB(d_2) & SB(d_{14}) \\ SB(d_5) & SB(d_1) & SB(d_{13}) & SB(d_9) \\ SB(d_0) & SB(d_{12}) & SB(d_8) & SB(d_4) \end{bmatrix}$$

Під час зворотної операції InvMixColumn стовпці стану розглядаються як поліноми, помножені на фіксований поліном d(x), який визначається за допомогою:

$c(x) \otimes d(x) = '01'$.

$d(x) = '0B' x3 + '0D' x2 + '09' x + '0E'$

AddRoundKey. Функція AddRoundKey виконує додавання (побітове XOR) матриці стану з ключем раунду:

$$AK(R) = \begin{bmatrix} R_{15} & R_{11} & R_7 & R_3 \\ R_{14} & R_{10} & R_6 & R_2 \\ R_{13} & R_9 & R_5 & R_1 \\ R_{12} & R_8 & R_4 & R_0 \end{bmatrix} \otimes \begin{bmatrix} rk_{15} & rk_{11} & rk_7 & rk_3 \\ rk_{14} & rk_{10} & rk_6 & rk_2 \\ rk_{13} & rk_9 & rk_5 & rk_1 \\ rk_{12} & rk_8 & rk_4 & rk_0 \end{bmatrix}$$

Ключі раунду обраховуються шляхом розкладання ключа шифрування і виводу з нього підключів для кожного перетворення AddRoundKey. В алгоритмі оригінальний ключ шифру є першим (rk^0), який застосований у додатковому AddRoundKey на початку першого раунду [8].

Раундові ключі rk^i , де $0 < i \leq 10$, обчислюється з попереднього ключа rk^{i-1}. Нехай q(j) ($0 \leq j \leq 3$) – стовпчик j таблиці rk^{i-1} і нехай w(j) - стовпець j rk^i . Тоді новий rk^i буде дорівнювати:

$$w(0) = q(0) \oplus (Rot(SB(q(3))) \oplus rcon^i$$

$$w(1) = q(1) \oplus w(0)$$
$$w(2) = q(2) \oplus w(1)$$
$$w(3) = q(3) \oplus w(2)$$

Раундова константа $rcon^i$ має значення [$'02'^{i-1}$; $'00'$; $'00'$; $'00'$]. Rot є функцією, яка отримує на вхід чотири байти і зсуває їх на один байт.

Опис архітектури. Пропонований підхід до будови шифратора дозволяє обчислювати всю ітерацію алгоритму використовуючи таблиці пошуку та операції XOR. Таблиці пошуку являються завчасно обчисленими комбінаціями операцій Subbytes та Mixcolumn.

По суті, це масив, який замінює собою обчислення під час виконання алгоритму на простішу операцію індексування масиву. У порівнянні з таблицею пошуку T-box 8x32 біт, пропоновані таблиці становлять 8x8 біт:

Послідовні операції SubByte та MixColumn на першій чверті раунду виглядають наступним чином:

$$R = MC\left(SB\left(SR(State)\right)\right) = A(x) \otimes SB(SR(State))$$

$$A(x) = \begin{bmatrix} '02' & '03' & '01' & '01' \\ '01' & '02' & '03' & '01' \\ '01' & '01' & '02' & '03' \\ '03' & '01' & '01' & '02' \end{bmatrix}$$

$State$ - це перетворені дані, а $A(x)$ - матриця векторів множення. Наведену операцію множення можна виконати за допомогою таблиці логарифмів та антилогарифмів (рисунки 9 і 10).

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	0	0	19	1	32	2	1A	C6	4B	C7	1B	68	33	EE	DF	3
1	64	4	E0	E	34	8D	81	EF	4C	71	8	C8	F8	69	1C	C1
2	7D	C2	1D	B5	F9	B9	27	6A	4D	E4	A6	72	9A	C9	9	78
3	65	2F	8A	5	21	F	E1	24	12	F0	82	45	35	93	DA	8E
4	96	8F	DB	BD	36	D0	CE	94	13	5C	D2	F1	40	46	83	38
5	66	DD	FD	30	BF	6	8B	62	B3	25	E2	98	22	88	91	10
6	7E	6E	48	C3	A3	B6	1E	42	3A	6B	28	54	FA	85	3D	BA
7	2B	79	A	15	9B	9F	5E	CA	4E	D4	AC	E5	F3	73	A7	57
8	AF	58	A8	50	F4	EA	D6	74	4F	AE	E9	D5	E7	E6	AD	E8
9	2C	D7	75	7A	EB	16	B	F5	59	CB	5F	B0	9C	A9	51	A0
A	7F	C	F6	6F	17	C4	49	EC	D8	43	1F	2D	A4	76	7B	B7
B	CC	BB	3E	5A	FB	60	B1	86	3B	52	A1	6C	AA	55	29	9D
C	97	B2	87	90	61	BE	DC	FC	BC	95	CF	CD	37	3F	5B	D1
D	53	39	84	3C	41	A2	6D	47	14	2A	9E	5D	56	F2	D3	AB
E	44	11	92	D9	23	20	2E	89	B4	7C	B3	26	77	99	E3	A5
F	67	4A	ED	DE	C5	31	FE	18	D	63	8C	80	C0	F7	70	7

Рисунок 9 – Таблиця логарифмів

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	1	3	5	0F	11	33	55	FF	1A	2E	72	96	A1	F8	13	35
1	5F	E1	38	48	D8	73	95	A4	F7	2	6	0A	1E	22	66	AA
2	E5	34	5C	E4	37	59	EB	26	6A	BE	D9	70	90	AB	E6	31
3	53	F5	4	0C	14	3C	44	CC	AF	D1	68	B8	D3	6E	B2	CD
4	4C	D4	67	A9	E0	3B	4D	D7	62	A6	F1	8	18	28	78	88
5	83	9E	B9	D0	6B	BD	DC	7F	81	98	B3	CE	49	DB	76	9A
6	B5	C4	57	F9	10	30	50	F0	0B	1D	27	69	BB	D6	61	A3
7	FE	19	2B	7D	87	92	AD	EC	2F	71	93	AE	E9	20	60	A0
8	FB	16	3A	4E	D2	6D	B7	C2	5D	E7	32	56	FA	15	3F	41
9	C3	5E	E2	3D	47	C9	40	C0	5B	ED	2C	74	9C	BF	DA	75
A	9F	BA	D5	64	AC	EF	2A	7E	82	9D	BC	DF	7A	8E	89	80
B	9B	B6	C1	58	E8	23	65	AF	EA	25	6F	B1	C8	43	C5	54
C	FC	1F	21	63	A5	F4	7	9	1B	2D	77	99	B0	CB	46	CA
D	45	CF	4A	DE	79	8B	86	91	A8	E3	3E	42	C6	51	F3	0E
E	12	36	5A	EE	29	7B	8D	8C	8F	8A	85	94	A7	F2	0D	17
F	39	4B	DD	7C	84	97	A2	FD	1C	24	6C	B4	C7	52	F6	1

Рисунок 10 – Таблиця антилогарифмів

До прикладу: $C = a \times b$

C можна обчислити за допомогою таблиць логарифмів наступним чином:

$$C = Log'((Log\ a) + (Log\ b))$$

Перетворення змішування стовпців опрацьовує кожен рядок окремо. Щоб обійти матричне множення виразів, всі байти замінюються за таблицями

логарифмів (додаванням, а не множенням). Якщо ми знайдемо чотири таблиці (T0 - T3), що містять 256 чисел (від 0 до 255), як дані:

Таблиці для зашифровки:

$To(a) = Log'\,((Log\ (01)) + (Log\ (SB(a))))$

$T1(a) = Log'\,((Log\ (01)) + (Log\ (SB(a))))$

$T2(a) = Log'\,((Log\ (02)) + (Log\ (SB(a))))$

$T3(a) = Log'\,((Log\ (03)) + (Log\ (SB(a))))$

Таблиці для розшифровки:

$ITo(a) = Log'\,((Log\ (0E)) + (Log\ (SB(a))))$

$IT1(a) = Log'\,((Log\ (09)) + (Log\ (SB(a))))$

$IT2(a) = Log'\,((Log\ (0D)) + (Log\ (SB(a))))$

$IT3(a) = Log'\,((Log\ (0B)) + (Log\ (SB(a))))$

Остаточний результат можна буде отримати шляхом виняткової диз'юнкції вихідних даних чотирьох таблиць (T0 - T3), як показано у наступному виразі:

$R_{15} = T_2(d_{15})\ xor\ T_3(d_{10})\ xor\ T_1(d_5)\ xor\ T_0(d_0)\ xor\ \mathrm{rk}_{15}$;

$R_{14} = T_0(d_{15})\ xor\ T_2(d_{10})\ xor\ T_3(d_5)\ xor\ T_1(d_0)\ xor\ \mathrm{rk}_{14}$;

$R_{13} = T_1(d_{15})\ xor\ T_0(d_{10})\ xor\ T_2(d_5)\ xor\ T_3(d_0)\ xor\ \mathrm{rk}_{13}$;

$R_{12} = T_3(d_{15})\ xor\ T_1(d_{10})\ xor\ T_0(d_5)\ xor\ T_2(d_0)\ xor\ \mathrm{rk}_{12}$;

$R_{11} = T_2(d_{11})\ xor\ T_3(d_6)\ xor\ T_1(d_1)\ xor\ T_0(d_{12})\ xor\ \mathrm{rk}_{11}$;

$R_{10} = T_0(d_{11})\ xor\ T_2(d_6)\ xor\ T_3(d_1)\ xor\ T_1(d_{12})\ xor\ \mathrm{rk}_{10}$;

$R_9 = T_1(d_{11})\ xor\ T_0(d_6)\ xor\ T_2(d_1)\ xor\ T_3(d_{12})\ xor\ \mathrm{rk}_9$;

$R_8 = T_3(d_{11})\ xor\ T_1(d_6)\ xor\ T_0(d_1)\ xor\ T_2(d_{12})\ xor\ \mathrm{rk}_8$;

$R_7 = T_2(d_7)\ xor\ T_3(d_2)\ xor\ T_1(d_{13})\ xor\ T_0(d_8)\ xor\ \mathrm{rk}_7$;

$R_6 = T_0(d_7)\ xor\ T_2(d_2)\ xor\ T_3(d_{13})\ xor\ T_1(d_8)\ xor\ \mathrm{rk}_6$;

$R_5 = T_1(d_7)\ xor\ T_0(d_2)\ xor\ T_2(d_{13})\ xor\ T_3(d_8)\ xor\ \mathrm{rk}_5$;

$R_4 = T_3(d_7)\ xor\ T_1(d_2)\ xor\ T_0(d_{13})\ xor\ T_2(d_8)\ xor\ \mathrm{rk}_4$;

$R_3 = T_2(d_3)\ xor\ T_3(d_{14})\ xor\ T_1(d_9)\ xor\ T_0(d_4)\ xor\ \mathrm{rk}_3$;

$R_2 = T_0(d_3)\ xor\ T_2(d_{14})\ xor\ T_3(d_9)\ xor\ T_1(d_4)\ xor\ \mathrm{rk}_2$;

$R_1 = T_1(d_3)\ xor\ T_0(d_{14})\ xor\ T_2(d_9)\ xor\ T_3(d_4)\ xor\ \mathrm{rk}_1$;

$R_0 = T_3(d_3)\ xor\ T_1(d_{14})\ xor\ T_0(d_9)\ xor\ T_2(d_4)\ xor\ \mathrm{rk}_0$;

Розмір однієї таблиці Ті для шифрування дорівнює 2 Кбіт. Для реалізації обох таблиць можна використати блокову пам'ять з довільним доступом (BRAM), їй вистачить місця. Реалізація вимагатиме лише 2 блоків оперативної пам'яті для комбінації SubByte з наступною операцією MixColumn, за допомогою існуючої двопортової оперативної пам'яті шляхом додавання подвоєного тактового генератора та деякої додаткової логіки.

BRAM конфігурується як двопортовий ПЗП (режим тільки для читання) задля надання доступу до 8-бітних значень пошуку, які відповідають 8-бітним вхідним адресам. При читанні використовується один тактовий фронт, і дані з комірки пам'яті, обраної за адресою, відображаються на вихідних портах після закінчення часу доступу до BRAM.

Цифровий тактовий генератор генерує два такти: CLK0 (з такою ж частотою, як і такт вхідного джерела) та CLK2X (подвоєна частота вхідного джерела) від джерела вхідного тактового сигналу CLKIN. CLK2X дозволить BRAM виводити дані двічі протягом одного повного циклу CLK0. Зі вхідного боку три мультиплексори 2×1 M1, M2 і M3 використовуються для вибору відповідних вхідних даних. Керуючим сигналом для мультиплексора M3 є CLK0.

Шістнадцять 8-розрядних регістрів R0 – R15 застосовані, щоб зберігати вихідні дані BRAM як на передньому, так і на задньому фронті CLK0; R0 – R3 і R8 – R11 спрацьовують за позитивним фронтом сигналу, а R4 – R7 і R12 – R15 - за негативним фронтом сигналу. Всі шістнадцять 8-розрядних виходів регістра складаються за схемою XOR для отримання кінцевого результату. Функціональна схема шифратора наведена в Додатку Б.

Генератор ключів відповідає за створення ключів для кожного раунду (рисунок 11).

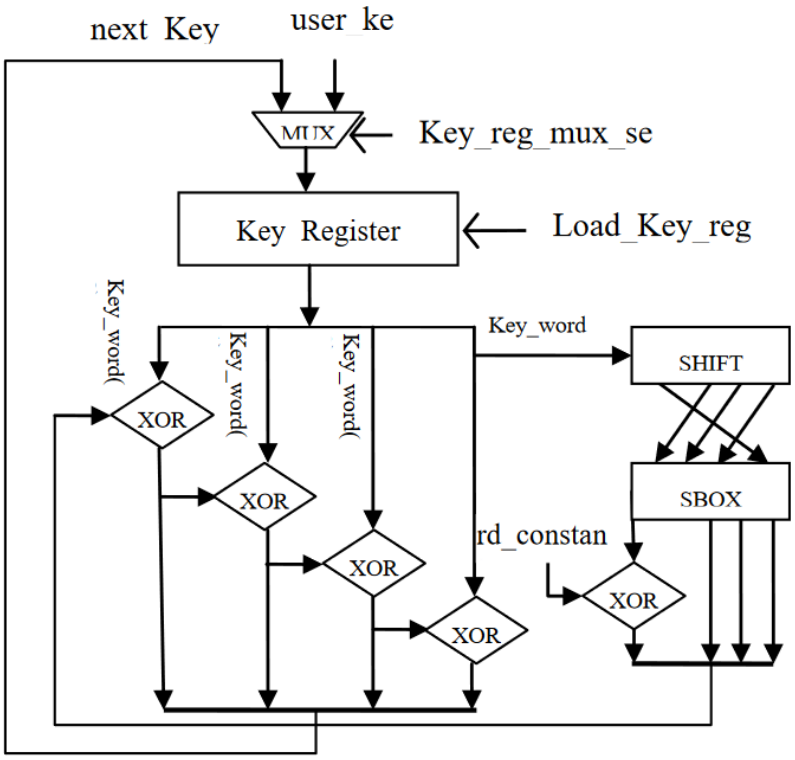


Рисунок 11 – Блок-схема модуля Key_schedule

Блок керування на основі скінченного автомату (Finite State Machine) генерує сигнали управління всіма мультиплексорами для коректного виконання перетворення ShiftRows.

Узагальнення та результати. Архітектура AES-128 з використанням таблиць пошуку, двопортової пам'яті та цифрового тактового генератора розроблена на мові VHDL та реалізується на ПЛІС Xilinx Spartan3E-4 XC3S500E.

У таблиці 5 наведено характеристики реалізації AES для шифрування 128-бітних даних з відповідним ключем 128 біт. Дизайн працює на тактовій частоті 168,765 МГц, що є достатньо високою для криптографічних додатків у реальному часі, і забезпечує пропускну здатність 270 Мбіт/с.

Таблиця 5 – Результати впровадження

Пропонований пристрій ПЛІС	Spartan3E XC3S500E-4 FT256
Максимальна тактова частота	168.765MHz
Кількість секцій	326
Модулі блокової пам’яті	3
Цикли	80
Пропускна здатність шифрування	270 Mbps

4.3 Будова кодера

Коди БЧХ завдячують своєю назвою творцям-розробникам – Боузу. Чоудхурі, Хоквінгему. Пристрій кодування має виконувати наступні функції:

1. Приймати і зберігати певний час інформаційну частину $u(x)$.

2. Обчислювати надлишкові (перевірні) розряди.

3. Сформувати поліном $V(x)$ та передати його до каналу зв'язку.

Головною складовою кодуючого пристрою є *регістр зсуву з лінійним логічним зворотним зв'язком* (РСЛЛЗЗ). В залежності від виду полінома, різняться способи побудови РСЛЛЗЗ.

В цьому випадку структурна схема регістру зсуву виходить з породжуючого полінома $g(x)$ і виглядає наступним чином (рисунок 12).

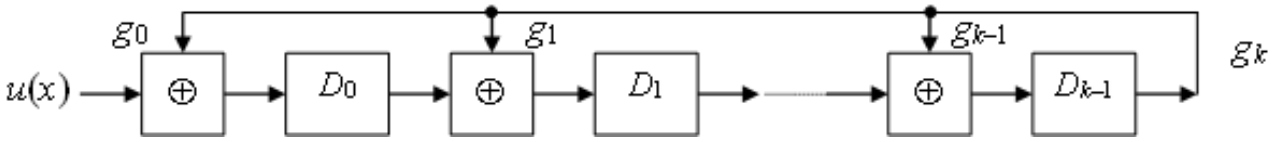


Рисунок 12 – Структурна схема пристрою поділу на поліном $g(x)$

На рисунку 12 g_i – це коефіцієнти утворюючого полінома $g(x)$ для конкретного обраного коду. За умови, що коефіцієнт $g_i = 0$ (x^i не входить у поліном),

i-й елемент зворотного зв'язку та суматор не додаються до схеми. Ступінь породжуючого полінома *k* дорівнює числу задіяних елементів пам'яті.

Кожен елемент пам'яті D_i відповідає x^i. Елементи працюють одномоментно від тактових імпульсів. Стан елемента пам'яті в будь-який момент часу залежить від пов’язаних з ним виходів інших елементів в попередній момент (фронт попереднього синхроімпульсу) і імпульсу на вході схеми [16]. Підсистема, що синхронізує регістр за циклом і тактом вважається за ідеальну та не зображена в структурах, що будуть подані далі. Функції збудження елементів пам'яті:

$$D_i^t = D_{i-1}^{t-1} \oplus g_i \cdot D_{k-1}^{t-1}, i = \overline{1..k-1}$$

$$D_0^t = u^t \oplus g_0 \cdot D_{k-1}^{t-1} = u^t \oplus D_{k-1}^{t-1}$$

Дана схема розподіляє інформаційний поліном $u(x)$ на породжуючий багаточлен $g(x)$. Поліном даних $u(x)$ множиться на x^k і, почавши зі старшого інформаційного розряду, послідовно подається на вхід. До моменту, коли розподіл закінчено (*n*-ий такт працюючої схеми), регістр містить залишок $D_0 \dots D_{k-1}$, що і буде являтись надмірними розрядами для заданого інформаційного вектора.

Пристрій кодування має розраховувати надлишкову частину комбінації, а також формувати поліном $V(x)$ та здійснити його передачу в канал зв'язку [16]. У зв'язку з тим, що перевірна частина буде формуватись *n* тактів, необхідною є затримка передачі до каналу зв'язку інформації на *k* тактів. Таким чином до $k + m = n$-ого такту інформаційний вектор буде виданий каналу зв'язку, а перевірна частина – обрахована та поміщена в регістр.

Наступним кроком є видача надлишкової частини до каналу зв'язку і припинення процедури поділу. Структурна схема кодуючого пристрою наведена на рисунку 13 (ЕЗ - елемент затримки, $к_1$, $к_2$ - ключі, К - керування).

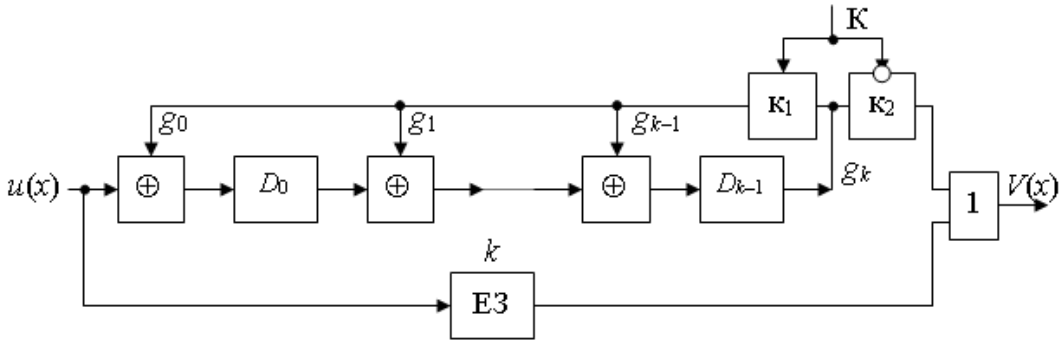


Рисунок 13 – Структурна схема кодуючого пристрою БЧХ-коду

Сенсом елемента затримки є організація затримки видачі інформаційного вектора на k тактів. Ключ $к_1$ відкритий перші n тактів для того, щоб зворотній зв'язок функціонував. Протягом наступних k тактів він закритий задля розриву зворотного зв'язку. Ключ $к_2$ навпаки – закритий протягом перших n тактів з метою відключення регістру від виходу і відкривається на наступних k тактах, щоб видати надлишкову частину до каналу зв'язку. Для перемикання ключів подається сигнал керування К: 1 - $к_1$ відкритий, $к_2$ закритий, 0 - $к_1$ закритий, $к_2$ відкритий.

Розглянута схема має недолік – наявна затримка на k тактів і додатковий елемент схеми для її реалізації. Перші k тактів проводиться множення інформаційного полінома $u(x)$ на x^k.

Елемент затримки можна прибрати, якщо подавати в схему багаточлен $u(x)$, помножений на величину x^k. Це здійснюється шляхом подання полінома $u(x)$ на вихід останнього елемента пам’яті, а не на вхід першого, який відповідає k-му ступеню. Функціональна схема кодера наведена в Додатку В.

Функції збудження елементів пам'яті:

$$D_i^t = D_{i-1}^{t-1} \oplus g_i \cdot (D_k^{t-1} \oplus u^t), i = \overline{1..k-1}$$

$$D_0^t = u^t \oplus g_0 \cdot D_{k-1}^{t-1} = u^t \oplus D_{k-1}^{t-1}$$

На перших m тактах зворотний зв'язок активний – ключ $к_1$ відкритий, і закривається під час наступних k тактів, щоб перервати зворотний зв'язок. Ключ $к_2$ у закритому положенні протягом m початкових тактів, щоб розімкнути регістр і вихід, на наступних тактах $к_2$ відкривається для видачі перевірної частини до

каналу зв'язку. Перемикання відбувається за сигналом керування аналогічно до попередньої схеми. Надлишкові розряди помноженого на x^k інформаційного полінома обраховуються впродовж перших m тактів роботи схеми.

У випадку схем, що формують коди БЧХ з парною мінімальною кодовою відстанню відсутні принципові відмінності окрім того, що для побудови РСЛЛЗЗ використовується утворюючий поліном $g(x) \cdot (1 \oplus x)$. Для кодування скорочених кодів $g(x)$ для скороченого і табличного коду збігаються [5].

Декодування циклічних кодів БЧХ. При декодуванні даних кодів відбувається виявлення та корекція помилок. Про спотворення розрядів кодової послідовності сигналізують залишки при діленні прийнятої послідовності $F(x)$ на породжуючий багаточлен $G(x)$. Повідомлення отримано без помилок за умови, що при діленні кодової комбінації на породжуючий поліном залишок дорівнює нулю. Ненульовий залишок вказує на наявність помилок. Для знаходження помилкових розрядів проводять такі дії:

1) отримана комбінація ділиться на породжуючий поліном;

2) рахується вага залишку (число одиниць).

3) за умови $W \le t_в$, де $t_в$ – кількість помилок, що коригуються кодом, отримане повідомлення сумується за модулем 2 разом з залишком кроку 2). Сума становить виправлену комбінацію.

У випадку $W > t_в$, відбувається зсув вліво прийнятого повідомлення і ділення результуючої комбінації на породжуючий поліном $G(x)$. Якщо $W \le t_в$ в залишку, ділене складається з залишком. Далі відбувається циклічний зсув послідовності вправо. В результуючій комбінації помилки відсутні. У випадку, що після проведених зсуву і поділу залишок має вагу $W > t_в$:

4) повторюється пункт 3 доти, поки не виконається умова $W \le t_в$.

5) якщо умова виконана – відбувається зсув вправо на ту кількість бітів, що на неї зсувалася підсумована з останнім залишком послідовність щодо прийнятої. Після цих дій комбінація виправлена.

Коди Боуза — Чоудхурі — Хоквінгема можна застосовувати для виправлення будь-якого числа спотворених розрядів. Проте ріст кратності помилки призводить до подовження коду і падіння швидкості передачі, а також вимагає більш складної приймально-передавальної апаратури.

ВИСНОВКИ

В результаті виконання роботи було спроектовано телекомунікаційний пристрій захисту даних для безпілотних літальних апаратів. Для цього проведено порівняльний аналіз криптографічних алгоритмів шифрування даних, відібрано найбільш оптимальний серед них – стандарт шифрування AES-128.

Також оглянуто методи завадостійкого кодування повідомлень, що дозволяють підтримувати високу надійність зв’язку, надано перевагу кодам Боуза — Чоудхурі — Хоквінгема. Опрацьовані рішення включені в алгоритм роботи пристрою, та, відповідно, в його структуру. Наведено функціональні схеми його складових — шифратора і кодера.

У висновку, розроблений пристрій повністю відповідає вимогам завдання: підтримує необхідну пропускну здатність шифрування в 270 Мбіт/сек, ефективно використовуючи блоки пам’яті, також виявляє та виправляє пакетні помилки.

Даний проєкт телекомунікаційного пристрою захисту даних для БпЛА забезпечує високу надійність передачі інформації та швидкодію, а також енергоефективність і тому посяде достойне місце серед вітчизняних чи іноземних аналогів.

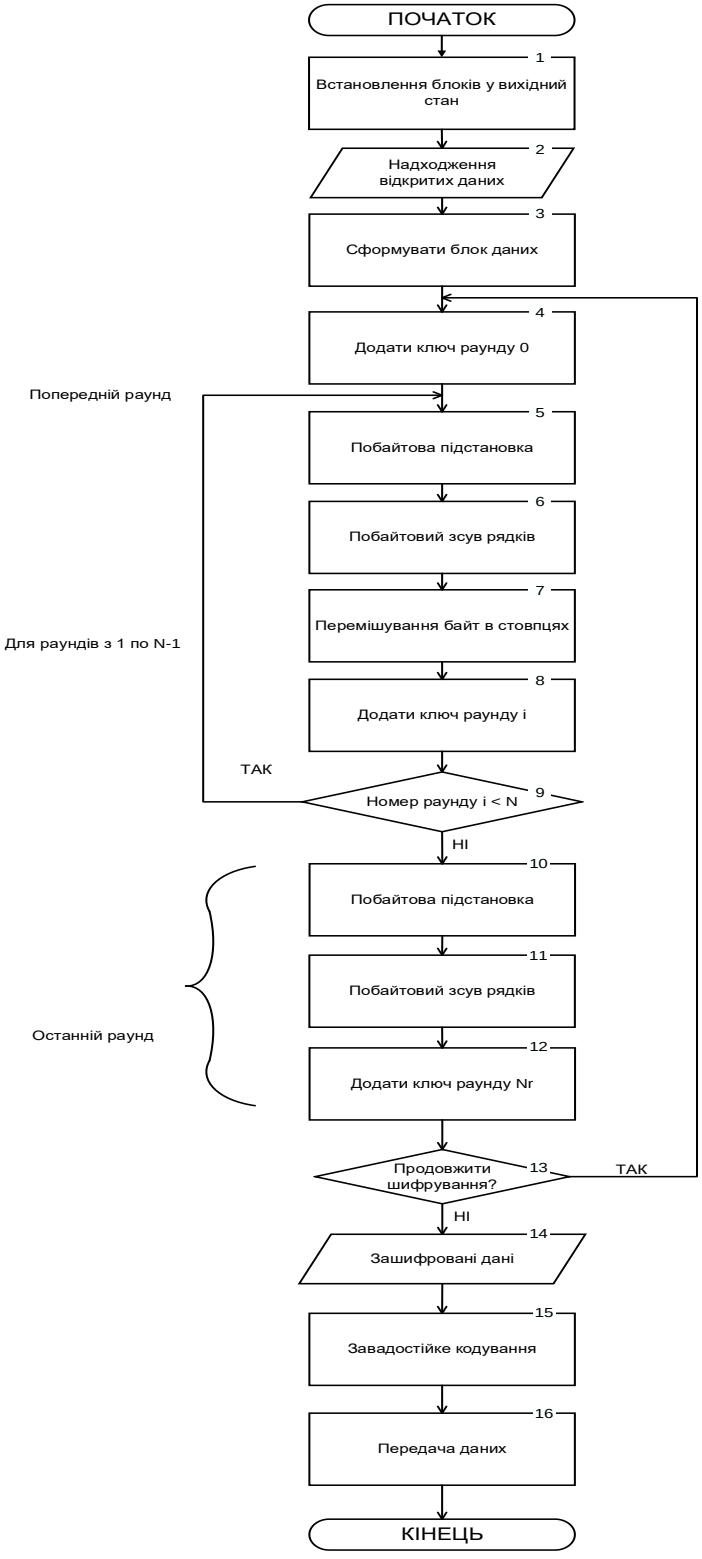
СПИСОК ЛІТЕРАТУРИ

1. Mykhatskyi O. Informative safety of unmanned aviation systems radio communication channels. Cybersecurity: Education, Science, Technique. 2018. №1. С. 56–62. URL: https://csecurity.kubg.edu.ua/index.php/journal/article/view/20/61 (дата звернення: 26.05.2024)
2. Особливості застосування безпілотних літальних апаратів органами та підрозділами поліції: метод. рек. / А. А. Саковський, С. М. Науменко та ін. Київ: Нац. акад. внутр. справ. 2022. 72 с.
3. Макаренко С. И. Противодействие беспилотным летательным аппаратам. Монография. – СПб.: Наукоемкие технологии, 2020. – 204 с.
4. Давиденко А. Н. Анализ вопросов закрытия информационного канала связи с беспилотным летательным аппаратом / А. Н. Давиденко, С. Я. Гильгурт // Зб. наук. пр. ІПМЕ НАН України. — К., 2014. — Вип. 71. — С. 61–64. URL: https://irbis-nbuv.gov.ua/cgibin/irbis_nbuv/cgiirbis_64.exe?C21COM=2&I21DBN=UJRN&P21DBN=UJRN&IMAGE_FILE_DOWNLOAD=1&Image_file_name=PDF/znpipm_2014_71_12.pdf (дата звернення: 26.05.2024)
5. Теория электрической связи. Помехоустойчивая передача данных в информационно-управляющих и телекоммуникационных системах: модели, алгоритмы, структуры: учеб, пособие / Е.Л. Кон, В.И. Фрейман. - Пермь: Изд-во Перм. гос. техн. ун-та, 2007. - 312 с.
6. Питерсон У., Уэлдон Э. Коды, исправляющие ошибки: перевод с англ./ под ред. Добрушина В.Л. – М.: «Мир», 1976 г. – 600 с.
7. Мельников В.П. Информационная безопасность и защита информации: Учебное пособие для вузов по спец. «Информационные системы и технологии» / В.П. Мельников, С.А. Клейменов, А.М. Петраков; Под ред. С.А. Клейменова. — 5-е изд., стер. — М. : Academia, 2011. — 331 с. : ил. — (Высшее профессиональное образование). — Библиогр.: с. 327-328
8. Шаповал, І. В.; Лебедев, Д. Ю. Алгоритм роботи пристрою AES шифратора. Problems of Informatization and Management, 2016, 1.53: 87-91. URL: https://scholar.archive.org/work/w5b6bvbajve5hc4req4mmp2lxe/access/wayback/https://jrnl.nau.edu.ua/index.php/PIU/article/download/10371/13652 (дата звернення: 26.05.2024)
9. Навроцький Д. О. Cryptographic system of protection UAVs communication channels against illegal intrusion. Ukrainian Scientific Journal of Information

Security. 2014. Т. 20, № 3. URL: https://core.ac.uk/download/pdf/325943177.pdf (дата звернення: 26.05.2024).

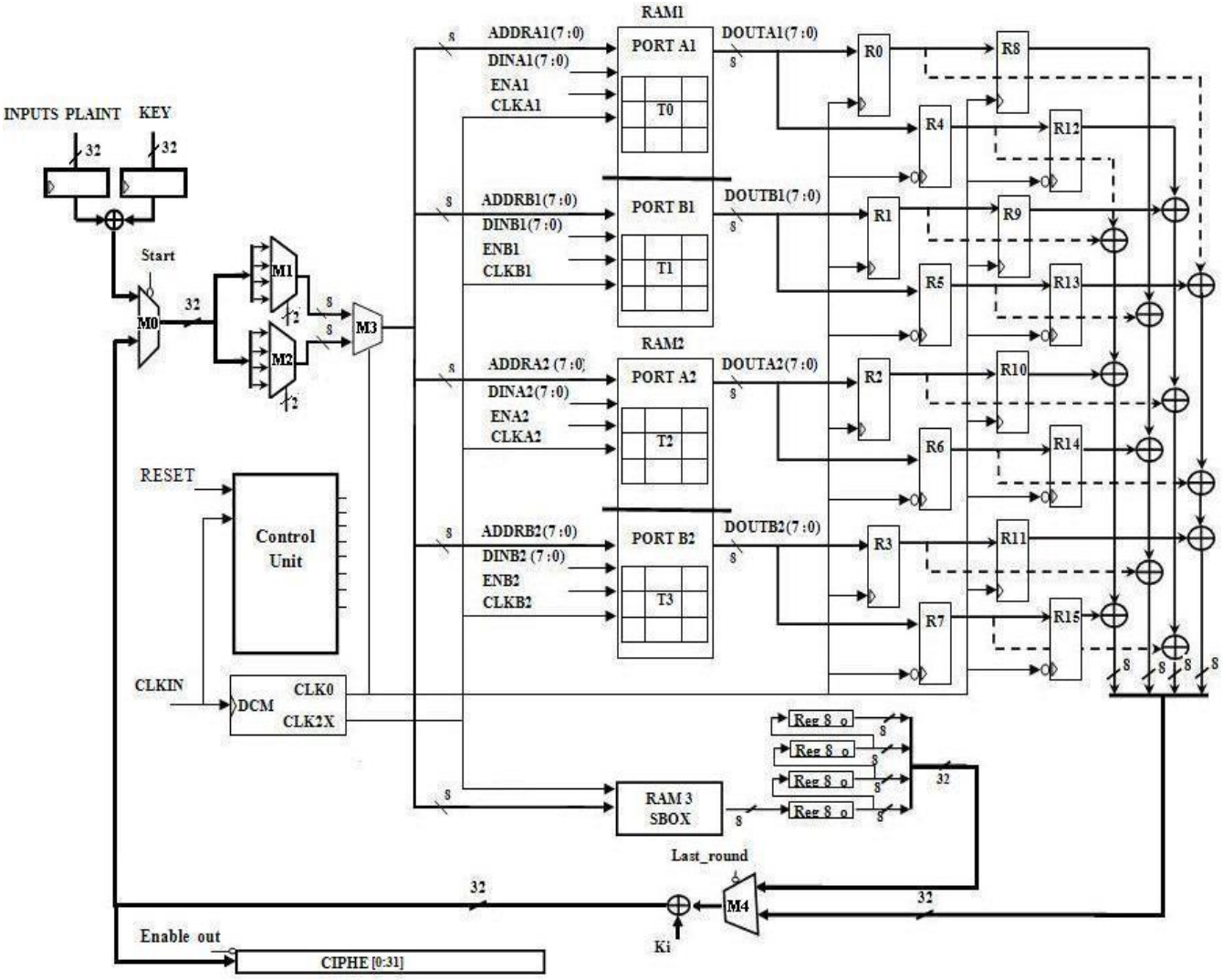
10. Easttom W. Modern Cryptography: Applied Mathematics for Encryption and Information Security. Springer International Publishing AG, 2022.
11. Кульчинська Н.З., Олійник Н.П., Дослідження алгоритму шифрування AES. Збірник матеріалів проблемно-наукової міжгалузевої конференції «Автоматизація та комп'ютерно – інтегровані технології» (АКІТ - 2021), Тернопіль, 2021. 145-149 с.
12. Syafaat, F., & Finandhita, A. (2019). Implementation of AES-128 Cryptography on Unm[anned Aerial Vechile and Ground Control System. Teknik Informatika–Universitas Komputer Indonesia, 10–19. URL: https://elibrary.unikom.ac.id/id/eprint/1149/13/22.10115361_FARHAN%20SYAFAAT_JURNAL%20DALAM%20BAHASA%20INGGRIS.pdf (дата звернення: 26.05.2024)
13. S. Gnatyuk, V. Kinzeryavyy, Y. Polishchuk, O. . Nechyporuk, і B. . Horbakha, «Аналіз методів забезпечення конфіденційності даних, які передаються з БпЛА», *Кібербезпека: освіта, наука, техніка*, вип. 1, вип. 17, с. 167–186, Вер 2022.
14. Скляров, О. В., Тітаренко, А. В., & Радченко, М. М. Огляд безпровідних каналів зв'язку для дистанційного управління БпЛА. In The III International Scientific and Practical Conference", 2024, Paris, France. 279 p.
15. Сучасні підходи до організації надійного завадостійкого та захищеного каналу супутникового зв'язку з безпілотними літальними апаратами / V. Soboliev та ін. Наукові праці Державного науково-дослідного інституту випробувань і сертифікації озброєння та військової техніки. 2021. № 7. С. 78–87. URL: https://dndivsovt.com/index.php/journal/article/view/90/87 (дата звернення: 26.05.2024).
16. El Adib S., Raissouni N. AES Encryption Algorithm Hardware Implementation Architecture: Resource and Execution Time Optimization. International Journal of Information and Network Security (IJINS). 2012. Т.1, № 2. URL: https://www.academia.edu/download/46180425/530-1961-1-PB.pdf (дата звернення: 26.05.2024).
17. Материалы к лекциям по дисциплинам ОПДС и ПДС/ В.М. Охорзин - Федеральное агентство связи Санкт-Петербургский государственный университет телекоммуникаций им.проф. М.А. Бонч-Бруевича.

ДОДАТОК А. Схема алгоритму



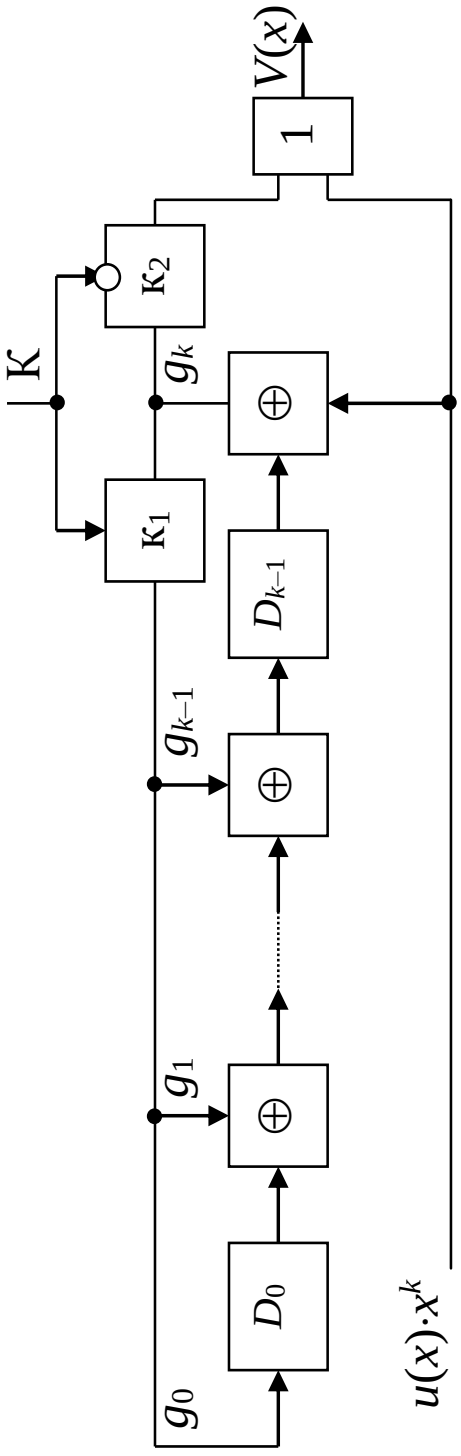
					ЕлІТ 6.172.00.02.122 СА		
					Телекомунікаційний пристрій захисту даних для безпілотних літальних апаратів. Схема алгоритму	Лит.	Масса / Масштаб
Изм.	Лист	№ документа	Подпись	Дата			
Разраб.		Забуга А. К.					
Пров.		Кулик І.А.				Лист 1	Листов 1
Т. контр.							
Н. контр.						СумДУ, гр. ТК-01	
Утв.		Опанасюк А. С.					

ДОДАТОК Б. Схема шифратора



					ЕлІТ 6.172.00.02.122 Е2		
					Телекомунікаційний пристрій захисту даних для безпілотних літальних апаратів. Схема електрична функціональна	Лит.	Масса / Масштаб
Изм.	Лист	№ документа	Подпись	Дата			
Разраб.		Забуга А. К.					
Пров.		Кулик І.А.				Лист 1	Листов 1
Т. контр.							
Н. контр.						СумДУ, гр. ТК-01	
Утв.		Опанасюк А. С.					

ДОДАТОК В. Схема кодера



					ЕлІТ 6.172.00.02.122 Е2		
					Телекомунікаційний пристрій захисту даних для безпілотних літальних апаратів. Схема електрична функціональна	Лит.	Масса	Масштаб
Изм.	Лист	№ документа	Подпись	Дата			
Разраб.		Забуга А. К.					
Пров.		Кулик І.А.				Лист 1	Листов 1
Т. контр.							
						СумДУ, гр. ТК-01	
Н. контр.							
Утв.		Опанасюк А. С.					