Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя

Факультет комп'ютерно-інформаційних систем і програмної інженерії
(повна назва факультету)
Кафедра комп'ютерних систем та мереж
(повна назва кафедри)

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття освітнього ступеня

магістра
(назва освітнього ступеня)

на тему: **Методи та засоби побудови комп'ютерної системи для потокового шифрування та передавання фотографічних зображень**

Виконав(ла): студент(ка) *6* курсу, групи *СІм-62*
спеціальності *123 «Комп'ютерна інженерія»*
(шифр і назва спеціальності)

	(підпис)	*Козарик Д.В.* (прізвище та ініціали)
Керівник	(підпис)	*Лещишин Ю.З.* (прізвище та ініціали)
Нормоконтроль	(підпис)	*Луцик Н.С.* (прізвище та ініціали)
Завідувач кафедри	(підпис)	*Осухівська Г.М.* (прізвище та ініціали)
Рецензент	(підпис)	(прізвище та ініціали)

Тернопіль
2023

Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя

Факультет *комп'ютерно-інформаційних систем і програмної інженерії*
(повна назва факультету)

Кафедра *комп'ютерних систем та мереж*
(повна назва кафедри)

ЗАТВЕРДЖУЮ
Завідувач кафедри
Осухівська Г.М.
(підпис) (прізвище та ініціали)
« ____ » _____ 2023 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

на здобуття освітнього ступеня *магістр*
(назва освітнього ступеня)

за спеціальністю *123 «Комп'ютерна інженерія»*
(шифр і назва спеціальності)

студенту *Козарику Дмитру Валерійовичу*
(прізвище, ім'я, по батькові)

1. Тема роботи *Методи та засоби побудови комп'ютерної системи для потокового шифрування та передавання фотографічних зображень*

Керівник роботи *Лещишин Юрій Зіновійович, к.т.н., доцент*
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

Затверджені наказом ректора від «*01*» *12* *2023* року № *4/7-1132*

2. Термін подання студентом завершеної роботи *27.12.2023 р.*

3. Вихідні дані до роботи *Інформація про особливості передавання та шифрування фотографічних зображень*

4. Зміст роботи (перелік питань, які потрібно розробити)
Вступ. 1. Аналіз існуючих методів шифрування та цифрових систем передавання зображень. 2. Модель цифрової системи передавання фотографічних зображень з шифруванням. 3. Моделювання методів симетричного шифрування в системах передавання фотографічних зображень. 4. Охорона праці та безпека в надзвичайних ситуаціях. Висновки

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень, слайдів)
1. Тема. Актуальність і мета дослідження. Задачі дослідження, об'єкт і предмет дослідж.
2. Математична модель каналу передавання фотографічного зображення.
3. Модель передавача та приймача з OFDM модуляцією.
4. Алгоритм шифрування AES-128.
5. Моделювання систем передавання фотографічних зображень.
6. Шифрування і передавання зображень.
7. Приймання і дешифрування зображень
9. Ефективність систем передавання фотографічних зображень. Наукова новизна.
8. Висновки.

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Охорона праці	*Осухівська Г.М.*		
Безпека в надзвичайних ситуаціях	*Стадник І.Я.*		

7. Дата видачі завдання *20.11.2023р.*

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1	*Аналіз існуючих методів шифрування*	*20.11-01.12*	
2	*Аналіз цифрових систем передавання зображень*	*20.11-02.12*	
3	*Пошук моделі цифрової системи передавання*	*22.11-04.12*	
4	*Розробка програмного забезпечення методів симетричного шифрування*	*24.11-10.12*	
5	*Моделювання методів симетричного шифрування*	*26.11-10.12*	
6	*Тестування і визначення ефективності в системах передавання фотографічних зображень*	*26.11-11.12*	
7	*Безпека життєдіяльності, основи охорони праці*	*02.12-06.12*	
8	*Оформлення кваліфікаційної роботи*	*06.12-12.12*	
9	*Попередній захист кваліфікаційної роботи*	*12.12-17.12*	
10	*Захист кваліфікаційної роботи*	*26.12-28.12*	

Студент _____ *Козарик Д.В.*
(підпис) (прізвище та ініціали)

Керівник роботи _____ *Лещишин Ю.З.*
(підпис) (прізвище та ініціали)

АНОТАЦІЯ

Методи та засоби побудови комп’ютерної системи для потокового шифрування та передавання фотографічних зображень // Кваліфікаційна робота магістра // Козарик Дмитро Валерійович // ТНТУ, Комп’ютерна інженерія, група СІм-62 // Тернопіль, 2023 // с. – 81, рис. – 19, табл. – 2, аркушів А1 – 8, бібліогр. – 20.

Ключові слова: симетричне шифрування, цифрові системи зв’язку, AES-128, модель каналу зв’язку.

У кваліфікаційній роботі досліджено методи та засоби побудови комп’ютерної системи для потокового шифрування та передавання фотографічних зображень.

На підставі аналізу існуючих методів потокового шифрування у цифрових системах зв'язку для передавання фотографічних зображень був обраний метод симетричного шифрування AES-128.

На базі існуючої моделі каналу зв'язку, було вибрано метод тестування ефективності передачі даних, використовуючи обраний метод шифрування. Результати моделювання процесу передавання фотографічних зображень дозволили отримати характеристики ефективності приймання сигналів у цифрових системах зв'язку при використанні симетричного шифрування методом AES-128.

Запропонована модель каналу зв'язку та методу шифрування AES-128 була реалізована з використанням інструментів Matlab.

ANNOTATION

Methods and tools for building a computer system for stream encryption and transmission of photographic images // Master thesis // Kozaryk Dmytro Valeriyovych // TNTU, Computer Engineering, group CIm-62 // Ternopil, 2023 // p. – 81, fig. - 19, tab. - 2, sheets A1 - 8, bibliography, - 20.

Keywords: symmetric encryption, digital communication systems, AES-128, communication channel model.

Methods and means of building a computer system for streaming encryption and transmission of photographic images were investigated in the qualification work.

Based on the analysis of the existing methods of stream encryption in digital communication systems, the AES-128 symmetric encryption method was chosen for the transmission of photographic images.

On the basis of the existing model of the communication channel, a method of testing the efficiency of data transmission was chosen, using the selected encryption method. The results of the simulation of the process of transmitting photographic images made it possible to obtain the characteristics of the efficiency of signal reception in digital communication systems when using symmetric encryption by the AES-128 method.

The proposed model of the communication channel and the AES-128 encryption method was implemented using Matlab tools.

ЗМІСТ

ВСТУП

Актуальність теми. Фото та відео інформація є важливою частиною сучасних комп’ютерних систем розумного будинку, систем охорони та обладнання для виконання різноманітних технологічних процесів. При передаванні цієї інформації на будь яку відстань виникає небезпека перехоплення або зловмисного втручання в її передачу. Крім того для передавання інформації дуже часто використовують системи з невеликої потужності із малим споживанням енергії, що побудовані на мікроконтролерах. Цих два факти призводять до необхідності захисту фотографічної інформації при її передаванні з мінімальними витратами енергії. Тобто виникає задача розробити методи та засоби побудови комп’ютерної системи для потокового шифрування та передавання фотографічних зображень.

Мета і задачі дослідження. *Метою дослідження* є розроблення методів та засобів побудови комп’ютерної системи для потокового шифрування та передавання фотографічних зображень.

Для досягнення поставленої мети необхідно розв’язати такі задачі:

— проаналізувати відомі методи шифрування та сучасні системи передавання фотографічних зображень, для вибору напряму дослідження;

— розробити математичну модель системи передавання фотографічних зображень з шифруванням, що уможливить моделювання таких систем;

— побудувати модель системи передавання фотографічних зображень, для дослідження її ефективності при застосуванні методів симетричного шифрування;

— отримати характеристики ефективності систем передавання фотографічних зображень, для оцінки впливу на них методів симетричного шифрування.

Об’єкт дослідження — процес прийому та передачі сигналів в системах передавання фотографічних зображень із застосуванням методів симетричного шифрування.

Предмет дослідження — методи та засоби побудови комп'ютерної системи для потокового шифрування та передавання фотографічних зображень.

Методи дослідження базуються на положеннях:

— статистичної радіотехніки для побудови моделей систем передавання фотографічних зображень;

— криптографії для застосування методів симетричного шифрування.

Наукова новизна одержаних результатів.

1. Вперше отримано характеристики ефективності системи передавання фотографічних зображень з OFDM модуляцією при використання у її складі алгоритму шифрування AES-128, що уможливлює порівняння впливу різних методів шифрування на ефективність систем передавання фотографічних зображень.

2. Набуло подальшого розвитку застосування моделі каналу зв'язку із затуханням до задачі моделювання системи передавання фотографічних зображень з OFDM модуляцією.

Практичне значення одержаних результатів полягає в наступному: отримані результати, уможливлюють застосування методів шифрування при в системах передавання фотографічних зображень портативних та IoT пристроїв, що підвищує захищеність їх використання.

Публікації. Результати дослідження були опубліковані у вигляді 2-х тез та апробовані на XI науково-технічної конференції «Інформаційнімоделі, системи та технології» (Тернопіль, 13-14 грудня 2023 р.) .

Структура роботи. Кваліфікаційна робота складається зі вступу, чотирьох розділів, висновків по роботі і переліку використаної літератури. Кваліфікаційна робота містить 81 сторінок, з них 66 сторінок основного тексту, 19 рисунків, 2 таблиць, 4 додатків і 20 найменувань переліку літератури.

РОЗДІЛ 1
АНАЛІЗ ІСНУЮЧИХ МЕТОДІВ ШИФРУВАННЯ ТА ЦИФРОВИХ СИСТЕМ ПЕРЕДАВАННЯ ЗОБРАЖЕНЬ

1.1. Порівняння існуючих методів апаратного шифрування

Передача зображень через системи може стати об'єктом стороннього втручання, що може призвести до втрати контролю і важливої інформації. Для захисту від неправомірних втручань використовують шифрування даних. При цьому важливим є вибір методу шифрування, який має надійний рівень захисту та може бути реалізований апаратними засобами, що використовуються паралельно з основними функціями процесора і мінімально завантажують його. Зокрема, для апаратної або програмної реалізації часто вибирають потокові або блочні методи шифрування, оскільки вони забезпечують високу швидкість шифрування [1, 2].

Більшість існуючих шифрів з секретним ключем можна однозначно віднести або до потокових, або до блочних шифрів. Важливо відзначити, що теоретична границя між ними може бути неоднозначною. Наприклад, алгоритми блочного шифрування можуть використовуватися в режимах потокового шифрування, як, наприклад, режими CFB і OFB для алгоритму DES.

Розглянемо основні відмінності між потоковими і блочними шифрами не лише з точки зору їх безпеки і зручності, але й з погляду їх вивчення у світі.

Швидкість шифрування. Потокові шифри відзначаються високою швидкістю шифрування, яка порівнюється зі швидкістю надходження вхідної інформації. Це дозволяє забезпечити шифрування практично в реальному масштабі часу незалежно від обсягу та розрядності потоку даних [3-7].

Швидкість блочних шифрів може бути меншою в порівнянні з потоковими через необхідність обробки блоків даних.

Розмноження помилок. У синхронних потокових шифрах відсутній ефект розмноження помилок. Кількість спотворених елементів у розшифрованій

послідовності дорівнює числу спотворених елементів зашифрованої послідовності, що була прийнята з каналу зв'язку.

Блочні шифри можуть виявити ефект розмноження помилок, де одна помилка в блоку може призвести до появи більше помилок у вихідному тексті.

Вразливість структури ключа. Структура потокового ключа може мати вразливі місця, які надають можливість криптоаналітику отримати додаткову інформацію про ключ, особливо при малому періоді ключа.

Блочні шифри, їх структура ключа менше схильна до подібних вразливостей.

Атаки лінійною алгеброю. Потокові шифри часто вразливі до атак, включаючи лінійну алгебру, особливо через кореляцію між виходами регістрів зсуву зі зворотнім лінійним зв'язком.

Блочні шифри менше схильні до атак лінійною алгеброю, що робить їх більш стійкими.

Враховуючи ці відмінності, вибір між потоковими і блочними шифрами залежить від конкретних вимог і контексту використання.

У зв'язку із становищем у світі варто відзначити кілька ключових аспектів, що стосуються вивчення та застосування потокових і блочних шифрів:

Напрямки вивчення шифрів. Більшість досліджень та аналізу спрямовані на блочні шифри, зокрема на алгоритми, що базуються на стандарті DES. Велика частина криптоаналізу спрямована на виявлення слабкостей у DES-алгоритмах.

Дослідження потокових шифрів менш систематизовані, і не існує чітко виокремленого напрямку вивчення. Методи злому потокових шифрів виявляються більш різноманітними та менш передбачуваними.

Критерії надійності шифрів. Немає чітких критеріїв надійності для блочних шифрів, і їх оцінка часто залежить від конкретного застосування. Стандартні критерії, що визначають стійкість, можуть варіюватися.

Для потокових шифрів існують встановлені вимоги до надійності, такі як більші періоди вихідних послідовностей, принципи Голомба та нелінійність, які слугують критеріями їх якості та ефективності.

Географія досліджень шифрів. Головним чином, дослідження та розробка блочних шифрів проводяться американськими криптографічними центрами.

Європейські криптографічні центри займаються більш динамічним дослідженням та розробкою потокових шифрів, що призводить до більшої різноманітності в цій області.

Останнім часом у сфері блочних шифрів помітних відкриттів не було, і дослідження у цьому напрямку може визначатися великою стабільністю стандартних DES-алгоритмів.

На відміну від блочних шифрів, область потокових шифрів є більш динамічною, і вона відзначається численними успіхами та невдачами в розробці нових схем та методів.

Ці аспекти свідчать про те, що вивчення і застосування потокових і блочних шифрів мають відмінності як у технічному, так і у географічному плані.

1.1.1. Опис і характеристики основних потокових методів шифрування.

Потоковий шифр представляє собою групу симетричних шифрів, які проводять шифрування кожного символу відкритого тексту незалежно від інших символів. Ці шифри базуються на операції XOR, де гама для шифру XOR формується за допомогою криптостійкого генератора псевдовипадкової послідовності символів, використовуючи ключ. Два найпоширеніших потокових шифри - A5 та RC4, а також інші алгоритми, такі як SEAL та Salsa20, варто розглянути детальніше [8-14].

A5 — це потоковий алгоритм шифрування, використовуваний для захисту даних, що передаються між телефоном і базовою станцією в європейській системі мобільного зв'язку GSM.

В основі A5 лежить шифр XOR, але з унікальним алгоритмом формування гами, що реалізується на основі трьох лінійних регістрів зсуву зі зворотнім зв'язком. Однак A5 зазнав змін, що дозволяють злом за прийнятний час.

RC4 — потоковий шифр, розроблений Роном Рівестом, широко використовується в протоколах безпеки, таких як TLS та WEP.

Незважаючи на швидкість та простоту реалізації, RC4 має вади, і його використання не рекомендується через виявлені методи успішної атаки.

SEAL — симетричний потоковий алгоритм шифрування, оптимізований для програмної реалізації.

Використовує велику таблицю, отриману з ключа, та чергування арифметичних операцій для забезпечення ефективності.

Шифр Salsa20 є системою потокового шифрування та став переможцем конкурсу "eStream" для програмного застосування з великою пропускною здатністю.

Використовується для шифрування даних, переданих поштовими системами, і відзначається високою швидкістю та стійкістю.

Кожен із зазначених потокових шифрів має свої переваги та недоліки, і вибір конкретного шифру залежить від конкретного застосування та вимог до безпеки системи.

Шифр Salsa20 використовує ряд операцій для забезпечення шифрування даних. Основні операції, які використовуються у цьому алгоритмі, включають: додавання 32-бітних чисел, побітове додавання по модулю 2 (XOR), зміщення бітів,

Salsa20 використовує операцію додавання для комбінування 32-бітних чисел, що додають стійкість та складність шифру.

Операція XOR використовується для побітового додавання по модулю 2, що є ключовим елементом утворення шифру.

Salsa20 використовує зміщення бітів для створення динамічної та непередбачуваної гами для шифрування.

Алгоритм включає хеш-функцію з 20 циклами, що забезпечує додатковий рівень захисту та безпеки. Основні перетворення хеш-функції нагадують алгоритм AES (Advanced Encryption Standard), що підсилює впевненість у його надійності та стійкості.

Загалом, Salsa20 володіє високою швидкодією та відмінною стійкістю, забезпечуючи ефективне шифрування для різноманітних застосувань, включаючи передачу даних через поштові системи.

1.1.2. Опис і характеристики основних блочних методів шифрування.

Найбільш використовувані алгоритми шифрування наведено в таблиці 1.1 [15-20].

Таблиця 1.1

Найбільш використовувані у світі алгоритми шифрування

Алгоритм	Розмір ключа, біт	Довжина блока, біт
DES	56	64
3DES	168	64
IDEA	128	64
Blowfish	32-448	64
ГОСТ 28147-89	256	64
AES Rijndael	128, 192, 256	128

DES (Data Encryption Standard) – це симетричний алгоритм шифрування, який був розроблений компанією IBM і отримав схвалення від уряду США у 1977 році як офіційний стандарт. Алгоритм використовує мережу Фейстеля з 16 раундами та ключем довжиною 56 біт. DES вважається застарілим, і його замінюють алгоритмами 3DES або AES.

Triple DES (3DES) – це симетричний блоковий шифр, створений на основі DES для усунення його головного недоліку – короткого ключа (56 біт). Хоча 3DES працює повільніше, його криптостійкість набагато вища. Зараз 3DES виходить з ужитку, і його заміняє алгоритм AES Rijndael, який працює швидше.

IDEA (International Data Encryption Algorithm) – це симетричний блоковий алгоритм шифрування, розроблений швейцарською фірмою Ascom. У порівнянні з DES, IDEA працює вдвічі швидше, має ключ довжиною 128 біт і має апаратну реалізацію. Проте IDEA повільніший за алгоритм ГОСТ 28147-89 та Blowfish у програмній реалізації.

Blowfish – це криптографічний алгоритм, який реалізує блокове симетричне шифрування зі змінною довжиною ключа. Він не має апаратної реалізації, проте вільно поширюється і відомий своєю незапатентованістю.

ГОСТ 28147-89 (Магма) – це радянський стандарт симетричного шифрування. Застосовує блочний шифр з 256-бітовим ключем та 32 циклами перетворення, працюючи з 64-бітними блоками. Має високу швидкодію, але різні реалізації можуть бути несумісними та мають вади.

Advanced Encryption Standard (AES) – це сучасний симетричний алгоритм блокового шифрування, який визнаний стандартом урядом США. Має апаратну реалізацію і широко використовується, зокрема, в процесорах Intel. AES відповідає сучасним вимогам і швидко заміняє застарілі алгоритми.

1.2. Опис алгоритму шифрування AES

В основному, алгоритм, запропонований Рейменом і Дейцменом (алгоритм Рейндол), і AES не є тотожними. Алгоритм Рейндол підтримує різні розміри блоку та ключа, дозволяючи вибирати їх у широкому діапазоні. З іншого боку, AES має фіксований розмір блоку у 128 біт, а довжина ключа може бути 128, 192 або 256 біт [3-8].

Алгоритм Рейндол дозволяє використовувати розміри блоку та ключа з кроком 32 біти у діапазоні від 128 до 256 біт. У порівнянні з цим, AES використовує фіксований розмір блоку 4×4 байти, що називається станом, незалежно від довжини ключа.

Для ключа у довжиною 128 біт, алгоритм роботи AES включає 10 послідовних раундів, в кожному з яких застосовуються відповідні процедури та

кроки шифрування, рис.1.1. Різноманітні версії AES із більшими розмірами блоку можуть мати додаткові колонки, оскільки структура блоку залишається фіксованою.

subBytes()

shiftRows()

mixcolumns() (у 10-му раунді пропускається)

xorRoundKey()

Процедура SubBytes()

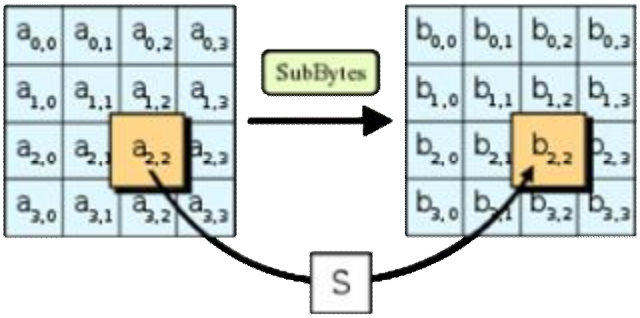


Рис. 1.1. Процедура SubBytes, передбачає що кожен байт в state замінюється елементом що поставлений у відповідь у фіксованій 8-бітній таблиці пошуку, S; b ij = S (a ij).

Процедура SubBytes() виконує обробку кожного байта стану ізольовано, застосовуючи нелінійну заміну байтів з використанням спеціальної таблиці замін, відомої як S-box. Ця операція важлива для забезпечення нелінійності та стійкості алгоритму шифрування. Побудова S-box включає два основні кроки.

На першому етапі отримується зворотне число для кожного байта в полі Галуа. Це важливий крок для створення нелінійних замін.

На другому етапі до кожного байта b, який утворює S-box, застосовується конкретна операція. Ця операція може включати в себе різні криптографічні перетворення, що змінюють значення байта згідно певних визначених правил.

Ці два етапи разом формують S-box, який використовується в SubBytes() для заміни байтів у стані алгоритму.

$$B'_i = b_i \oplus b_{(i+4) \bmod 8} \oplus b_{(i+5) \bmod 8} \oplus b_{(i+6) \bmod 8} \oplus b_{(i+7) \bmod 8} \oplus c_i$$

Де 0≤i<8}, і де bi є і-ий біт b, а ci - і-ий біт константи c. У такий спосіб забезпечується стійкість до атак, заснованих на простих алгебраїчних властивостях.

Процедуру S-box можна зобразити таблицею підстановки, таблиця 1.2:

Таблиця 1.2.

Таблиця підстановки для заданих параметрів масиву

S-box																
\	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
0	63	7c	77	7b	f2	6b	6f	c5	30	01	67	2b	fe	d7	ab	76
1	ca	82	c9	7d	fa	59	47	f0	ad	d4	a2	af	9c	a4	72	c0
2	b7	fd	93	26	36	3f	f7	cc	34	a5	e5	f1	71	d8	31	15
3	04	c7	23	c3	18	96	05	9a	07	12	80	e2	eb	27	b2	75
4	09	83	2c	1a	1b	6e	5a	a0	52	3b	d6	b3	29	e3	2f	84
5	53	d1	00	ed	20	fc	b1	5b	6a	cb	be	39	4a	4c	58	cf
6	d0	ef	aa	fb	43	4d	33	85	45	f9	02	7f	50	3c	9f	a8
7	51	a3	40	8f	92	9d	38	f5	bc	b6	da	21	10	ff	f3	d2
8	cd	0c	13	ec	5f	97	44	17	c4	a7	7e	3d	64	5d	19	73
9	60	81	4f	dc	22	2a	90	88	46	ee	b8	14	de	5e	0b	db
a	e0	32	3a	0a	49	06	24	5c	c2	d3	ac	62	91	95	e4	79
b	e7	c8	37	6d	8d	d5	4e	a9	6c	56	f4	ea	65	7a	ae	08
c	ba	78	25	2e	1c	a6	b4	c6	e8	dd	74	1f	4b	bd	8b	8a
d	70	3e	b5	66	48	03	f6	0e	61	35	57	b9	86	c1	1d	9e
e	e1	f8	98	11	69	d9	8e	94	9b	1e	87	e9	ce	55	28	df
f	8c	a1	89	0d	bf	e6	42	68	41	99	2d	0f	b0	54	bb	16

Наприклад, якщо на вході 19 то на виході будемо мати d4. Тобто це простий шифр звичайної підстановки.

Процедура ShiftRows(), рис.1.2.

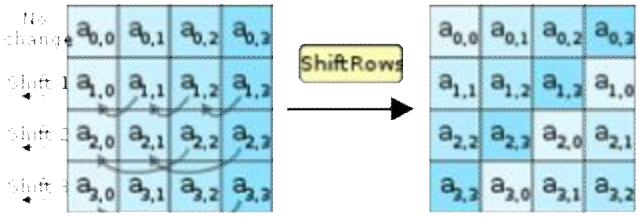


Рис. 1.2. Процедура ShiftRows, передбачає що байти у кожному рядку state періодично зміщаються вліво. Розмір зсуву байтів кожного рядка залежить від його номера

Функція ShiftRows взаємодіє з рядками таблиці State, виконуючи циклічний зсув рядків по горизонталі на r байтів, де r визначається номером рядка. Для нульового рядка r = 0, для першого - r = 1, і так далі. Після застосування цієї трансформації кожна колонка вихідного стану формується з байтів, що розташовані відповідно у кожній колонці початкового стану.

У випадку алгоритму Rijndael для 128-бітних і 192-бітних рядків патерн зсуву рядків однаковий. Проте, для блоку розміром 256 біт, він відрізняється тим, що 2-й, 3-й і 4-й рядки зміщуються на 1, 3 і 4 байти відповідно.

За сутністю, це є простою операцією перестановки байтів у таблиці 4x4 State, що надає додатковий рівень обертання та змішування даних, що сприяє стійкості шифру.

Процедура MixColumns(), рис.1.3

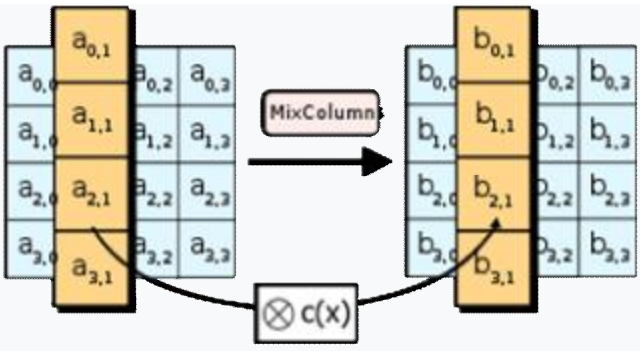


Рис. 1.3. Послідовність виконання процедури MixColumns, коли кожен стовпець стану перемножується з фіксованим многочленом c(x).

Послідовність виконання процедури MixColumns, коли чотири байти кожного стовпця State змішуються, застосовуючи для цього зворотну лінійну трансформацію. Процедура MixColumns обробляє стан по колонках, трактуючи кожен стовпець як поліном четвертого степеня. Після чого над цими поліномами виконується операція множення впо модулю на фіксований многочлен. Разом з процедурою ShiftRows, процедура MixColumns задає дифузію в шифр.

Під час операції MixColumns, кожен стовпчик стану множиться на матрицю, яка для 128-бітного ключа має вигляд

$$\begin{bmatrix} 2 & 3 & 1 & 1 \\ 1 & 2 & 3 & 1 \\ 1 & 1 & 2 & 3 \\ 3 & 1 & 1 & 2 \end{bmatrix}.$$

Процедура AddRoundKey(), рис.1.4

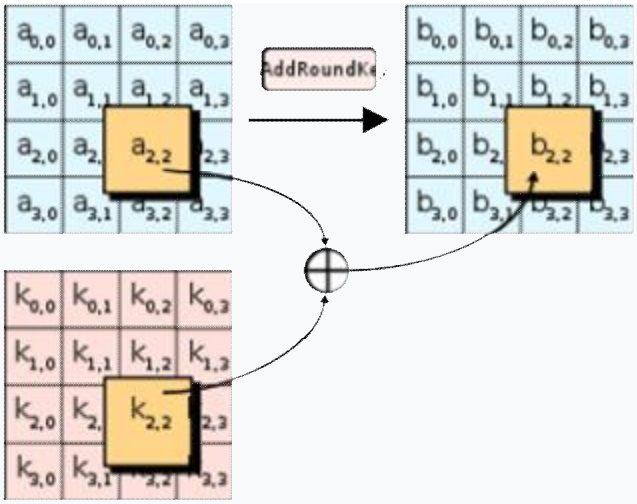


Рис. 1.4. Послідовність виконання процедури AddRoundKey, передбачає що кожен байт стану сполучається з RoundKey при виконанні операції XOR.

При виконанні процедури AddRoundKey, RoundKey отриманий для кожного раунду об'єднується із значенням State. Для кожного раунду Roundkey

отримується значення `CipherKey` застосовуючи процедуру `KeyExpansion`; розмір кожного RoundKey буде таким же, що і розмір State. Ця процедура робить побітовий XOR для кожного байта значення `State` із кожним байтом значення `RoundKey`.

Тобто це простий побайтовий алгоритм XOR байту ключа з байтами таблиці значень State.

Тобто алгоритм шифру AES подібний до алгоритмів потокових шифрів, адже в основі алгоритмів таких шифрів є функція XOR.

1.3. Сучасні цифрові методи передачі даних

Багато стандартів цифрового зв'язку використовують технологію OFDM (Orthogonal Frequency-Division Multiplexing — мультиплексування з ортогональним частотним поділом каналів). OFDM є цифровою схемою модуляції, яка базується на використанні великої кількості близько розташованих ортогональних піднесучих частот. Кожна з цих піднесучих частот модулюється за допомогою стандартних методів модуляції, таких як квадратурна амплітудна модуляція, на низькій символьній швидкості. Це забезпечує збереження загальної швидкості передачі даних, аналогічно звичайним схемам з однією несучою частотою в тій же смузі пропускання. Застосування зворотного перетворення Фур'є використовується для отримання сигналів OFDM на практиці.

OFDM модуляція широко використовується в різних областях, таких як:

бездротових системах зв'язку стандартів IEEE 802.11 (Wi-Fi) і HIPERLAN/2;

наземних системах цифрового телебачення DVB-T, DVB-T2 і ISDB-T;

наземних системах мобільного телебачення DVB-H, DVB-T2, T-DMB, ISDB-T і Mediaflo;

бездротових системах зв'язку стандартів IEEE 802.16 (WiMAX);

бездротових системах зв'язку стандартів IEEE 802.20, IEEE 802.16e (Mobile WiMAX) і WiBro;

бездротових системах зв'язку стандарту IEEE 802.15.3a.

Однією з основних переваг OFDM (рис.1.5) порівняно з схемою з однією несучою частотою є її здатність ефективно протистояти складним умовам у каналі. Наприклад, OFDM успішно вирішує проблеми, такі як загасання в області високих частот у довгих мідних провідниках, вплив вузькосмугових завад та частотно-вибіркове загасання, спричинене множинним розповсюдженням. Це досягається без застосування складних фільтрів-еквалайзерів. OFDM спрощує канальне фільтрування, розглядаючи сигнал як безліч повільно модульованих вузькосмугових сигналів, а не як один швидко модульований широкосмуговий сигнал. Використання низької символьної швидкості також дозволяє використовувати захисний інтервал між символами, що сприяє подоланню тимчасового розсіювання та усуває міжсимвольну інтерференцію.

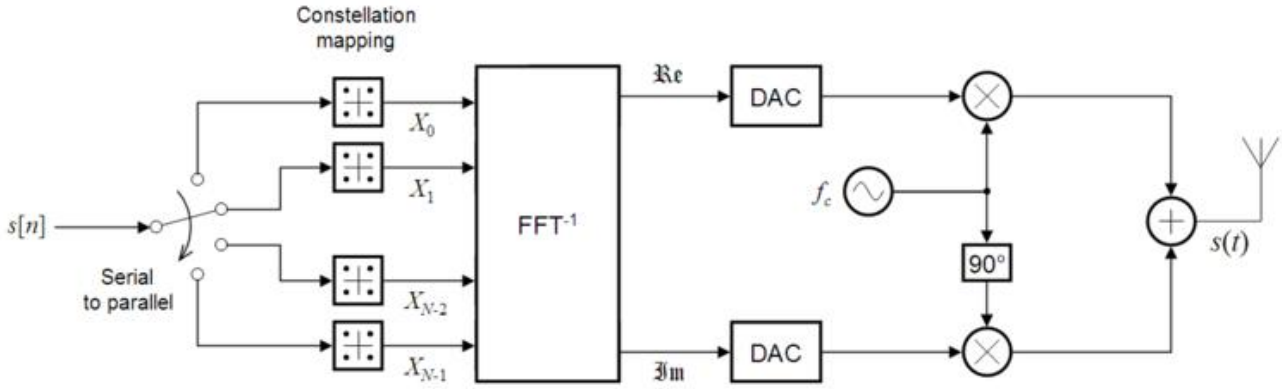


Рис.1.5. Структурна схема передавача сигналів з OFDM

Сигнал OFDM представляє собою комбінацію декількох ортогональних піднесучих частот, на кожній з яких передаються дані, модульовані незалежно за допомогою одного з типів модуляції (BPSK, QPSK, 8-PSK, QAM і ін.). Далі, цей сумарний сигнал модулюється на радіочастоті.

Позначимо s[n] як послідовність двійкових цифр. Перед застосуванням оберненого швидкісного перетворення Фур'є (FFT) ця послідовність спершу

перетворюється в N паралельних потоків, кожен з яких відображається в потік символів за допомогою процедур фазової (BPSK, QPSK, 8-PSK) або амплітудно-фазової квадратурної модуляції (QAM). При використанні BPSK, отримуємо потік двійкових чисел (1 і -1), у випадку QPSK, 8-PSK, QAM — потік комплексних чисел. Оскільки ці потоки є незалежними, метод модуляції та кількість біт на символ можуть відрізнятися в кожному з них. Таким чином, різні потоки можуть мати різну бітову швидкість. Наприклад, якщо пропускна здатність лінії складає 2400 бод (символів в секунду), перший потік може працювати з QPSK (2 біта на символ) та передавати 4800 біт/с, тоді як інший може використовувати QAM-16 (4 біта на символ) та передавати 9600 біт/с.

Застосування оберненого FFT визначає N одночасно отриманих символів, створюючи таку ж кількість комплексних відліків в часовій області, рис.1.6. Після цього цифро-аналогові перетворювачі (DAC) конвертують їх в аналогову форму окремо для дійсної та уявної компонент, які подальше модулюються косинусоїдою та синусоїдою радіочастоти відповідно. Ці сигнали об'єднуються, утворюючи переданий сигнал s(t).

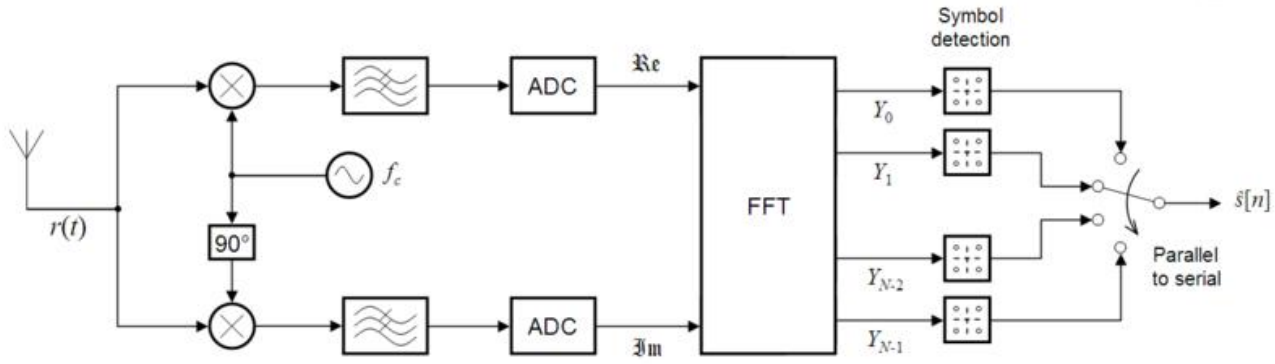


Рис.1.6. Структурна схема приймача сигналів з OFDM

Приймач отримує сигнал r(t), виділяє косинусну (cos) і синусну (sin) квадратурні складові, використовуючи множення r(t) на і , і фільтри нижніх частот, які відсікають коливання в смузі близько 2fc. Отримані сигнали піддаються цифровому перетворенню за допомогою аналого-цифрових

перетворювачів (ADC) та швидкому перетворенню Фур'є (FFT). В результаті отримуємо сигнал у частотній області.

Тепер маємо N паралельних потоків, кожен з яких перетворюється в двійкову послідовність за допомогою визначеного алгоритму фазової модуляції (застосованого BPSK, QPSK, 8-PSK в передавачі) або амплітудно-фазової квадратурної модуляції (застосованого QAM в передавачі). В ідеалі отримуємо потік бітів, ідентичний потоку, який передавався передавачем.

1.4 Висновки до розділу 1

Алгоритм шифрування AES є подібним до потокових шифрів і відзначається високою швидкістю шифрування, що робить його сприятливим для застосування в різних сферах, зокрема, в цифрових системах зв'язку. Зокрема, в алгоритмах OFDM (Orthogonal Frequency Division Multiplexing), які використовуються в стандартах Wi-Fi, WiMAX, DVB-T2 та інших, застосовується модуляція з використанням BPSK, QPSK, 8-PSK, QAM та інших типів.

З урахуванням швидкості та ефективності AES, зокрема з ключем 128 біт, цей алгоритм є відповідним для застосування в сучасних мікроконтролерах від різних виробників, таких як Atmel, Mikrochip, Texas Instruments та інші.

Враховуючи високу завадостійкість та швидкість передачі даних, яку надає OFDM модуляція в системах передавання зображень, доцільно провести моделювання впливу шифрування зображень за допомогою алгоритму AES-128 на характеристики ефективності каналу зв'язку, що використовує OFDM модуляцію.

РОЗДІЛ 2
МОДЕЛЬ ЦИФРОВОЇ СИСТЕМИ ПЕРЕДАВАННЯ ФОТОГРАФІЧНИХ ЗОБРАЖЕНЬ З ШИФРУВАННЯМ

2.1 Математична модель каналу зв’язку

Основна вимога до системи радіозв'язку полягає в ефективній та надійній передачі значної кількості інформації на великі відстані при обмеженій потужності передавача [6-10]. Однак, достовірна передача інформації на реальних каналах зв'язку стикається з рядом викликів, серед яких можна виділити три основні причини.

Зовнішні перешкоди включають різноманітні електромагнітні процеси в атмосфері, іоносфері та космічному просторі, такі як атмосферні перешкоди і космічні шуми. Внутрішні перешкоди можуть виникати від електроустановок, сусідніх радіостанцій та віддзеркалення сигналу від місцевих об'єктів.

Спотворення сигналу та виникнення помилок в каналі можуть бути спричинені технічною недосконалістю обладнання. Поліпшення апаратури може зменшити ці види спотворень.

Технічні обмеження пристроїв також можуть впливати на достовірність передачі інформації.

Хоча спотворення сигналу та технічна недосконалість пристроїв можуть бути поліпшені через вдосконалення обладнання, зовнішні перешкоди залишаються складним викликом, оскільки їх вплив може бути непередбачуваним і впливати на систему зв’язку, рис.2.1.

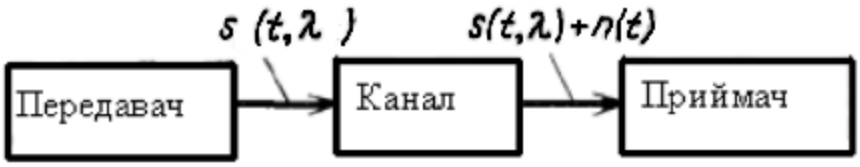


Рис. 2.1 Система радіозв'язку.

Окрім вже зазначених зовнішніх джерел перешкод, існують внутрішні перешкоди, що локалізовані в різних елементах передавача та приймача. До цих перешкод можна віднести флуктуаційний шум ламп, напівпровідникових приладів, опір втрат, нестабільності живлячої напруги, мікрофонний ефект і інші.

В різних конкретних ситуаціях можливі різні види перешкод. Проте у всіх випадках загальним і характерним є наявність нормального флуктуаційного шуму, який обумовлений природними причинами і не може бути повністю усунутий. Це включає в себе теплові і інші шуми навколишнього середовища та власні шуми радіоприймального пристрою. Теплові шуми оточуючого простору, які вхоплюються антеною, додаються до корисного сигналу і складаються із власного шуму радіоприймача.

Власний шум радіоприймача і теплові шуми навколишнього середовища лінійно додаються до вхідного корисного сигналу приймача. Перешкоди, що лінійно додаються до сигналу, відомі як адитивні завади.

Отже, в практичних випадках коливання *y(t)* можна представити у вигляді суми (2.1).

$$y(t) = s(t, l) + n(t). \tag{2.1}$$

Тут $s(t, l)$— пердставляє приймальний сигнал, що залежить від кількох параметрів, які позначені абстрактним вектором l ; $n(t)$— узагальнений шум радіоприймача і навколишнього середовища.

Загалом для спрощення трактуватимемо $n(t)$ як гаусів білий шум з нульовим математичним сподіванням і дельта-функцією кореляції (2.2):

$$\langle n(t) \rangle = 0, \ \langle n(t_1) n(t_2) \rangle = \frac{N_0}{2} d(t_2 - t_1). \tag{2.2}$$

Тут N_0 представляє односторонню спектральну щільність адитивного шуму. Це можна пояснити тим, що тепловий шум навколишнього середовища визначається відомою формулою Найквіста, і його спектральна густина майже постійна в діапазоні радіочастот. Власний шум радіоприймача, головним чином, визначається його високочастотною частиною, яка має дуже широку смугу пропускання.

Ще одним з часто використовуваних показників рівня адитивного шуму є ефективна температура шуму T_e:

$$N_0 = kT_e,$$

де $k = 1{,}38 * 10^{-23}\ Дж / К$ — стала Больцмана; T_e — ефективна шумова температура в градусах Кельвіна (К).

Зрозуміло, виразом (2.1) не враховує всі можливі практичні сценарії. Є різноманітні інші види перешкод, які можуть впливати на приймальний сигнал, такі як вузькосмугові перешкоди, імпульсні перешкоди і інші. Проте, в незалежності від інших перешкод, завжди присутній адитивний білий шум $n(t)$.

У зв'язку зі спотвореннями в каналі та наявністю різних видів перешкод, приймальний сигнал завжди є випадковим і виявляється точно не передбаченим. З теорії оптимального радіоприйому відомо, що чим більше апріорних відомостей маємо про сигнал і перешкоди, тим вища ймовірність правильного прийому переданого повідомлення.

Врахуємо особливості можливих спотворень переданого сигналу в каналі, тобто вид трансформації переданого сигналу $s_0(t, l_0)$ в прийнятий $s(t, l)$. Вектор l, параметрів отриманого сигналу, крім параметрів вектор l_0 переданого сигналу може містити специфічні нові складові частини пов’язані із часом запізнювання та ін. Звісно, врахування всіх фізичних процесів на шляху розповсюдження радіохвиль може бути складним завданням через широкий спектр можливих умов та впливів. Зазвичай використовують апроксимовані математичні моделі для опису цих процесів.

Навіть при значному розмаїтті радіоканалів залежно від їх призначення, можна виділити різні типи. Серед них: 1) системи зв'язку в оптичному діапазоні видимості; 2) системи зв'язку, які використовують дифракцію за межами видимості; 3) системи, що використовують відбиття від іоносфери; 4) системи іоносферного або тропосферного розсіяння; 5) зв'язок за допомогою дипольних відбивачів, метеорних слідів або іонізованої газової плазми; 6) системи, які використовують штучні супутники Землі.

Незважаючи на відмінності в характеристиках окремих каналів, всі вони мають кілька загальних рис: 1) присутність одного чи кількох входів та виходів; 2) в основному, вони лінійні; 3) радіосигнал пройшовши канал, має тимчасову затримку та може втрачати сигнал, що може змінюватися в часі; 4) канали піддаються впливу різних перешкод. Враховуючи різні радіотехнічні системи, можна також зазначити тип передаваних сигналів, а саме сигнали, випромінювані передавальною антеною, як вузькосмугові.

$$s_0(t, l_0) = f(t)\cos[w_0 t + j(t)] = R_e\left\{F(t)e^{jw_0 t}\right\}, \qquad (2.3)$$

де $F(t) = f(t)\exp[jj(t)]$—комплексна складова, огинаюча радіочастотного сигналу. Її складові означають, що функції $f(t)$ і $j(t)$, що відображають закономірності амплітудної і фазової (частотної) модуляції, поступово змінюються в порівнянні з коливанням високочастотної несучої частоти $\cos w_0 t$. Тому ширина смуги Δw спектру сигналу $s_0(t)$ буде набагато меншою за ширину смуги несучої частоти w_0.

$$\Delta w \langle\langle w_0 . \qquad (2.4)$$

Основні принципи класифікації каналів визначаються за різними ознаками, проте дві з них виявляються основними:

1. Характер адитивних перешкод, що втручаються в канал, і форма взаємодії цих перешкод з сигналом.
2. Метод перетворення переданого сигналу в прийняття його отримувачем.

Як зазначено вище, у простих випадках присутність перешкод у каналі можна моделювати нормальним адитивним білим шумом на стороні приймача, як відображено в рівнянні (2.1). В подальших розглядається переважно такий тип каналів, які відповідають першому принципу і визначаються як канали з адитивним гаусовим (нормальним) білим шумом.

Крім того, перешкоди можуть взаємодіяти з сигналом шляхом їхнього перемноження (наприклад, амплітудні затухання). У таких випадках вживається термін "мультиплікативні перешкоди".

Також канали класифікують за наступним принципом, тобто залежно від різного характеру сигналів $s(t, l)$, отримуваних на виході каналу зв'язку, при наперед заданому сигналі $s_0(t, l_0)$ на вході каналу. За цим принципом виділяють кілька типів каналів.

Однопроменеві канали

Припустимо, що електромагнітні коливання, випромінені передавальною антеною, поширюються вздовж одного шляху. Для коротких проміжків часу можна розглядати такі комунікаційні лінії, як лінії наземного короткого хвильового та ультракороткого хвильового діапазонів радіозв'язку, лінії прямої видимості між Землею та повітрям, повітрям та повітрям, а також між Землею та космічним апаратом тощо [6].

Для опису однопроменевого каналу позначимо коефіцієнт ослаблення амплітуди отриманого сигналу (в порівнянні з радіо випромінюванням) у момент часу t через функцію $a(t)$, а час запізнювання отриманого сигналу через $\Delta(t)$. Тоді отриманий сигнал матиме вигляд

$$s(t, l) = R_e\{a(t)F(t - \Delta(t))\exp[jw_0(t - \Delta(t))]\}. \qquad (2.5).$$

Зазвичай, параметри $a(t)$ і $\Delta(t)$ є випадковими, і при різних частинних випадках характер цих параметрів отриманого сигналу буде різним.

Для випадку багатьох прикладних завдань у записі отриманого сигналу (2.5) допустимі такі логічні спрощення.

1. Час запізнювання отриманого сигналу зазвичай змінюється досить повільно, а тому його можна описати:

$$\Delta(t) = \Delta + \frac{d\Delta}{dt}t + \frac{1}{2}\frac{d^2\Delta}{dt^2}t^2 + \ldots \simeq \Delta + \dot{\Delta}t \tag{2.6}$$

де Δ — деякий стартовий, початковий час запізнювання, а $\dot{\Delta}$(t)— швидкість зміни часу запізнювання в часі.

2. Зміна часу запізнювання, задається величиною Δt, на інтервалі спостереження є відносно малою і не впливає істотно на огинаючу сигналу. Тому величиною Δt для функції F можна знехтувати:

$$F(t - \Delta - \Delta t) \cong F(t - \Delta). \tag{2.7}$$

Наведене спрощення справедливе для таких випадків, коли зміна величини Δt є значно меншою за $1/\Delta f$, де $\Delta f = \Delta w / 2p$.

3. Нехай позначимо t як деякий середній час запізнювання. Зміни запізнювання Δ, відповідно будуть $w_0|\Delta - t\ | \approx \pm p$, тобто істотно змінюється лише аргумент функції $w_0(t - \Delta(t))$, але практично не вплмває на $F(t - \Delta)$. Тому Δ доцільно представити у вигляді

$$\Delta = t + J / w_0, \tag{2.8}$$

де J — фаза отриманого сигналу, що приймає будь-які значення в інтервалі $(-p, p)$.

З урахуванням виразів (2.6) – (2.8) отриманий сигнал, можна записати:

$$s(t, l) = \mathrm{Re}\{a(t)F(t - t)\exp j[(w_0 - \Omega)t - w_0 t - J]\} \quad (2.9)$$

де Ω — доплерівський зсув частоти отриманого сигналу:

$$\Omega = \Delta w_0 \quad (2.10)$$

Порівнюючи (2.3) і (2.9) помітно, що цьому у випадку під параметрами переданого і отриманого сигналів слід розуміти

$l_0 = \{w_0, f(t), j(t)\}$ і $l = \{w_0, f(t), j(t), a(t), t, \Omega\}$.

Зазвичай вектор параметрів l трактують. як багатокомпонентний марківський процес і сигнал $s(t, l)$ спостерігається на фоні марківського шуму, тобто отримуємо марківську модель каналу.

Для випадку однопроменевого каналу виділяють декілька таких моделей каналу.

1. Якщо параметри Ω, t і $a(t) = a$ незмінні, причому задається що a і J — заздалегідь невідомі. То у такому випадку отриманий сигнал має вигляд

$$s(t, l) = af(t - t)\cos[(w_0 - \Omega)t + j(t - t) - w_0 t - J]. \quad (2.11)$$

Тобто отримують модель каналу зв’язку коли розглядають випадок некогерентних ліній радіозв'язку при умовах прямої видимості.

2. Якщо вважати коефіцієнт $a(t)$ дійсною стохастичною функцією часу, то загалом отримуємо сигнал

$$s(t, l) = a(t)f(t - t)\cos[(w_0 - \Omega)t + j(t - t) - w_0 t - J]. \quad (2.12)$$

Оскільки стохастична функція $a(t)$ перемножується з корисним сигналом, то вважають, що в каналі зв’язку зявилась мультиплікативна завада. Таку

мультиплікативну заваду $a(t)$ описують за допомогою одновимірної густини імовірності $P(a)$ і автокореляційній функції, для якої густина імовірності є релеевскою

$$P(a) = \frac{a}{s^2} \exp\left(-\frac{a^2}{2s^2} \right), a \rangle 0, \qquad (2.13)$$

У такому випадку спостергаємо релеєвске завмирання отриманого сигналу.

Залежно від часу тривалості кореляції мультиплікативні перешкоди $a(t)$ поділяють на два типи: повільні флуктуації — $a_0(t)$, та швидкі — $a_1(t)$. В цьому випадку $a(t) = a_0(t) a_1(t)$.

Фізичні причини повільного та швидкого завмирання приймального сигналу різняться, а також різні підходи використовуються для зменшення їхнього впливу. Звичайно, повільне завмирання зазвичай пов'язане з природними явищами, що впливають на умови поширення електромагнітних хвиль. Ці явища включають зміни в тропосфері відповідно до метеорологічних умов, часу доби, року та клімату, а також варіації стану іоносфери, які обумовлені геомагнітною та сонячною активністю, ядерними вибухами і іншими факторами. Всі ці зміни відбуваються плавно та поступово.

Натомість швидке завмирання, яке характеризується кореляційним часом в долі секунд, переважно спричинене явищем багатопроменевості. Це означає, що в каналі існує велика кількість променів (шляхів), якими електромагнітні хвилі можуть поширюватися від точки передачі до точки прийому.

Для зменшення негативного впливу швидких завмирань можна використовувати оптимальний вибір сигналів і їх оптимальний прийом. Однак у випадку повільних завмирань ці підходи можуть бути менш ефективними. Для забезпечення надійної роботи системи під час тривалого погіршення стану каналу важливо забезпечити достатню потужність випромінювання, правильно

вибрати розміри антен і так далі (тобто використовувати системний запас) або застосовувати адаптивні системи, такі як тимчасове зниження швидкості передачі чи зміни робочої частоти.

Запропонована математична модель каналу зв’язку враховує вплив завад від середовища поширення сигналів та багато променеве поширення радіо хвиль та пов’язані з цим завмирання в каналі зв’язку.

2.2 Модель системи зв’язку на основі OFDM сигналів

Системи передавання фотографічних зображень для збільшення пропускної здатності використовують алгоритм з мультиплексуванням та з ортогональним частотним поділом каналів використовують велике число ортогональних піднесучих частот, для кожної з яких можливо незалежне застосування різних схем модуляції й кодування інформаційної послідовності.

При розробці моделі системи з ортогональним частотним поділом каналів використовувався низькочастотний еквівалент OFDM сигналу [7-12]:

$$s(t)=\sum_{k=0}^{N-1} s_k(t)=\sum_{k=0}^{N-1} A_k e^{j2\pi kf/T}, \quad 0\le t\le T, \qquad (2.14)$$

де k – індекс піднесучих частот, sk(t) – сигнал на k- піднесучих частот, Ak – амплітудна складова послідовності інформаційних символів, N – кількість піднесучих частот, T – тривалість інформаційного символу.

При розробці OFDM систем передачі даних важливо враховувати, що сигнали на к-піднесучих частотах є ортогональними на тактовому інтервалі T. Проте ортогональність кожної з піднесучих частот напряму пов'язана зі швидкістю передачі даних, і у деяких ситуаціях спектри сигналів на кожній з піднесучих частот можуть перекриватися частково. Багато OFDM сигналів традиційно допускають часткове перекриття спектрів, таке як QAM-4, QAM-16 та інші.

Ефективне використання спектра частот значно залежить від типу реалізації детекції даних, такого як когерентне чи некогерентне детектування. У представленій моделі використовується некогерентне детектування [6].

На етапі передавача процес формування OFDM сигналів розкладається на кілька стадій, які ефективно виконуються незалежно одна від одної як у апаратній частині, так і за допомогою відповідних алгоритмів [7]. Можна умовно виділити п'ять основних етапів: перетворення потоку ("Serial to parallel unit"), цифрова модуляція ("Digital modulator"), обчислення зворотного швидкого перетворення Фур'є ("IFFT"), цифро-аналогове перетворення ("DAC") та квадратурна модуляція ("Quadrature modulator").

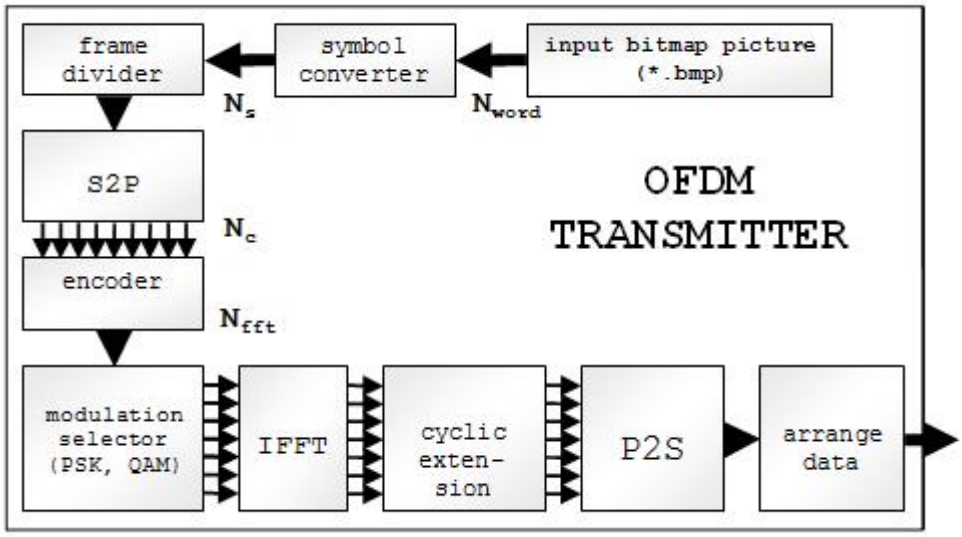


Рис. 2.2. Структурна схема OFDM передавача.

На рисунку 2.2 наведено структурну схему передавача OFDM, реалізовану в середовищі "Matlab". Давайте розглянемо призначення ключових компонентів системи на прикладі передачі графічного зображення.

Початкові дані представлені файлами графічних зображень у форматі *.bmp з глибиною передачі кольору 8 біт у відтінках сірого. В такому форматі доступні 256 відтінків сірого кольору. Декодування графічних зображень в відтінках сірого у моделі виконується за допомогою вбудованої функції "imread.m" з пакету "Matlab". Кожен піксель перетворюється в однобайтове

слово, утворюючи двовимірний масив. Висота вихідної картинки h визначає кількість рядків, а ширина w – кількість стовпців у згенерованій матриці. Матриця зображення, яка завантажується, конвертується в послідовний потік даних, при цьому кожен елемент масиву проходить послідовне перетворення з 8-бітного формату в інформаційні символи (слова) заданого розміру. Розрядність слова закодованого символу залежить від вибору порядку модуляції і зростає зі збільшенням порядку модуляції. Для виконання цього перетворення в моделі використовується функція "imgconv.m".

У блоці “symbol converter” матриця зображення перерозподіляється на двійкові стовпці відповідно до заданого порядку модуляції Ns і розміром вхідного зображення Nw і перетвориться у вектор вихідних символів N_d для необхідного типу модуляції. У цьому випадку розмір вектора вихідних символів визначатися по формулі (2.15):

$$N_d = N_w \frac{N_{word}}{N_s}, \tag{2.15}$$

де Ns – порядок модуляції, Nw – розмір зображення на вході, N_{word} – глибина графічного зображення.

Далі проводиться розбиття послідовності на необхідну кількість символьних блоків, які потім перекладаються у код, відповідний глибині передачі кольору пікселя.

При виконанні цього перетворення може виникнути ситуація, коли який-небудь символ "випадає" з послідовності через великий рівень шуму в каналі. У такому випадку використовується алгоритм прогнозування, який порівнює поточний і наступний символи, заповнюючи пропущені місця оціненими значеннями.

Далі символи в інформаційній послідовності групуються в складені кадри OFDM передавача залежно від змінної Spf (від англійської "Symbol per Frame").

Змінна Spf визначає кількість символів, які припадають на одну піднесучу частоту в кожному переданому кадрі, і обчислюється за формулою [8]:

$$SpF = ceil\left(\frac{2^{14}}{N_c}\right), \tag{2.16}$$

де ceil – функція “Matlab” округлення в більшу сторону, N_c – кількість піднесучих частот.

Операцію перенесення символів у кадри піднесучих частот в OFDM модуляторі виконує блок “frame divider”. Модулятор опрацьовує дані послідовно кадр за кадром. У випадку, якщо довжина інформаційної послідовності N_d не кратна заданій кількості N_c піднесучих частот, то послідовність наприкінці доповнюється нулями.

В OFDM передавачі блок “S2P” (від англ. “serial-to-parallel”) ділить дані з послідовного потоку в паралельний потік з N_c кількістю піднесучих частот.

Для поліпшення параметрів системи в схемах модуляторів передавачів використовують різні способи кодування даних [1,5]. У представленій моделі реалізоване диференціальне кодування, яке здійснюється в блоці “encoder”. Як приклад на рис. 2.3. представлений фазовий спектр сигналу QAM-16 з урахуванням диференціального кодування.

При диференціальнім кодуванні, в отриману раніше матрицю несучих частот, додатково додається перевірочний вектор довжиною N_c, який може бути як набором випадкових чисел, так і заданим набором значень.

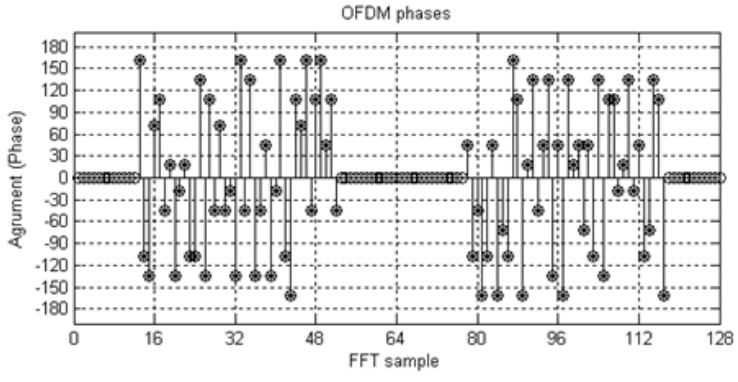


Рис. 2.3. Фазовий спектр QAM-16 з диференціальним кодуванням.

Далі закодовані інформаційні слова перетворюються у відповідні значення фаз і амплітуд. На цьому етапі також відбувається поділ потоку на Nc піднесучих частот і задається тип модуляції – фазова (PSK) або квадратурна амплітудна (QAM).

Далі виконується обчислення зворотного швидкого перетворення Фур'є у блоці “IFFT” відповідно до заданого розміру перетворення й кількістю Nc піднесучих частот, які задаються користувачем. Результат обчислення зворотного перетворення Фур’є на виході блоку дає один символьний період у часовій області. Кількість символьних періодів відповідає кількості рядків Nb у масиві переданих даних.

Для зменшення рівня міжсимвольних спотворень до кожного отриманого часового блоку додається деякий локальний захисний інтервал, що представляє циклічний префікс, формований у блоці “cyclic extension” шляхом додавання певного кількості символів у початок послідовності інформаційного блоку.

У блоці ”P2S” відбувається перетворення Nb+1 рядків двовимірного масиву в послідовність інформаційних блоків і глобальних захисних інтервалів, які формуються шляхом додавання відліків нулів між кадрами.

У блоці “arrange data” відбувається впорядкування отриманої послідовності. При цьому формується низькочастотний еквівалент, що відповідає OFDM сигналу.

Приймач. Усі функції, виконувані цифровими блоками приймача, є взаємно зворотними операціями тим операціям, які проводилися в передавачі й були описані вище. Розглянемо призначення основних елементів OFDM приймача, структурна схема якого представлена на рис. 2.4.

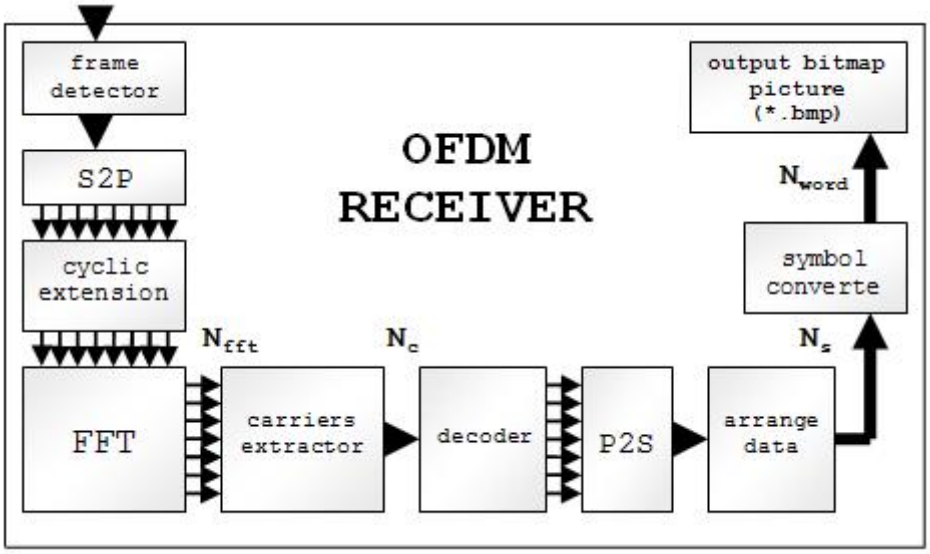


Рис. 2.4. Структурна схема OFDM приймача.

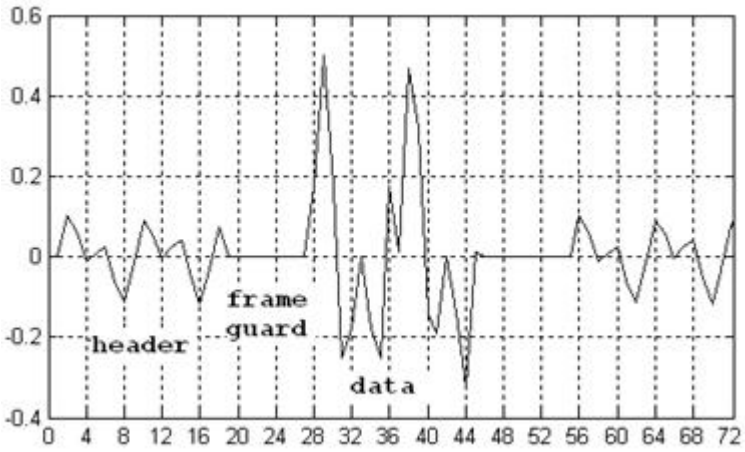


Рис. 2.5. Послідовність із одного кадра.

В структурі приймача ключовим елементом є детектор кадрів "frame detector," який виконує важливі завдання для ефективної обробки прийнятої послідовності. Основна мета цього детектора полягає в забезпеченні синхронізації прийому інформаційної послідовності шляхом точного визначення початку інформаційних символів в прийнятій послідовності. Крім цього, функції детектора кадрів включають в себе розпізнавання захисного інтервалу, визначення заголовка та закінчення кадра, а також видалення циклічних префіксів в усьому блоку інформації.Приклад одного інформаційного кадра представлений на рис. 2.5.

Для декодування сигналу в приймачі визначається обсяг переданих даних по числу кадрів Nf прийнятої послідовності [6]:

$$N_f = ceil\left[\frac{h \cdot w\left(\frac{N_{word}}{N_s}\right)}{SpF \cdot N_c}\right], \tag{2.17}$$

де h – висота зображення, w – ширина зображення, Ns – порядок цифрової модуляції.

Кількість кадрів прямо пропорційна обсягу переданої інформації, що пов'язана з розмірами та глибиною кольору зображення, і зворотно пропорційна порядку модуляції. Рівень інформаційної частини може в кілька разів перевищувати рівень заголовка сигналу та закінчення кадра, як зображено на рис. 4. При цьому глобальні захисні інтервали дорівнюють нулю, що дозволяє розрізнювати та відкидати компоненти, що не несуть інформації.

Для цього виконується пошук початку кадра, який можна розділити на кілька етапів. На першому етапі обчислюється модуль кожного відліку послідовності і проводиться дискретизація кадра із кроком Rstep, прямо пропорційним довжині ПФ (потужність Фур'є). Далі дані проходять через цифровий фільтр, який визначає положення нульових значень до та після інформаційної частини. Потім сигнал зсувається на величину, обумовлену відношенням Ts/Rstep (де Ts – символьний період), і знову фільтрується.

У результаті визначення положення початку кадра (фрейму) frame_loc можна розрахувати за формулою:

$$frame\_loc = \min\left(idx + T_s + nerr\right) - 1, \tag{2.18}$$

де idx – значення початку захисного інтервалу, $nerr$ – помилка оцінки виміру захисного інтервалу зверху.

Оскільки фреймів у реальних системах може бути небагато, узагальнимо формулу на багатокадрову систему:

$$frames\_loc = \min\left(idx + T_s + nerr + \left|R_{x\_st} - 1\right|\right) - 1, \tag{2.19}$$

де $Rx_{\_st}$ – номер фрейму.

Кадри (фрейми) послідовно обробляються з урахуванням того, що до першого був доданий заголовок, а до останнього – закінчення кадра. OFDM демодулятор обробляє дані кадр за кадром. Після визначення положень початку і кінця фреймів послідовний потік розділяється на паралельні, які подальше надходять на блок "cyclic extension", де віддаляються всі захисні інтервали від інформаційної частини сигналу.

Далі кожен з паралельних потоків даних надходить на блок прямого швидкого перетворення Фур'є (БПФ), який перетворює кожний інформаційний кадр у спектральні відліки. Залежно від типу обраної модуляції корисна інформація може містити як амплітуду і фазу сигналу (для сигналів QAM), так і лише фазу сигналу (для сигналів PSK).

У блоку "carriers extractor" відбувається добування інформації з кожної піднесучої частоти для блоку розміром Nb. Алгоритми отримання корисної інформації відрізняються відповідно до типу модуляції. Під час добування інформації з кожної уявної компоненти піднесучої частоти та спектральних відліків, їх зіставляють із відповідними еталонними значеннями, отриманими з коду для конкретного типу та порядку модуляції.

Слід врахувати, що в інформаційному потоці зберігається набір нулів, необхідний для модулятору для рівномірного складання паралельного потоку. Цей набір нулів віддаляється в блоку "arrange data". У блоку демодулятора "P2S" дані перетворюються в послідовний вигляд, утворюючи підсумковий

інформаційний вектор, який подається в блок "symbol converter", де інформаційні символи перетворюються в "слова", які потім конвертуються в графічне зображення.

За підсумками обробки даних приймачем у моделі передбачений вивід сигнальних сузір'їв за значеннями дійсної й уявної компонентів вектора кадра сигналу.

2.3 Висновки до розділу 2

Наведена математична модель каналу зв'язку є загальною для значної кількості випадків передачі даних, та враховує вплив завад від середовища у якому відбувається поширення сигналів та багатопроменевого поширення радіохвиль разом із пов'язаними з цим, явищами завмираннь в каналі зв'язку.

Наведена математична модель каналу зв'язку реалізується за допомогою інструментів Matlab.

Вибрана модель системи зв'язку передавання фотографічних зображень на основі OFDM з використанням квадратурної амплітудної модуляції дозволяє проводити моделювання системи зв'язку з використанням Matlab і додавати до неї різноманітні алгоритми шифрування.

РОЗДІЛ 3
МОДЕЛЮВАННЯ МЕТОДІВ СИМЕТРИЧНОГО ШИФРУВАННЯ В СИСТЕМАХ ПЕРЕДАВАННЯ ФОТОГРАФІЧНИХ ЗОБРАЖЕНЬ

3.1 Особливості моделювання методів шифрування та систем зв’язку в середовищі Matlab.

У MATLAB існують два підходи до моделювання систем зв'язку. Перший полягає в використанні розрахунків, що описують передачу інформації, а другий — в динамічному моделюванні з використанням Simulink®. Ці можливості втілюються через спеціалізовані компоненти: Communications Toolbox містить необхідні функції MATLAB, а Communications Blockset надає користувачеві блоки для моделювання систем зв'язку в Simulink. Ці блоки і функції спрямовані переважно на фізичний та канальний рівні інформаційних мереж згідно з семирівневою моделлю OSI. Використання цих спеціалізованих пакетів базується на загальних функціях цифрової обробки сигналів, які реалізовані в Signal Processing Toolbox та DSP Blockset.

На рис. 3.1 показана загально відома структурна схема системи зв'язку.

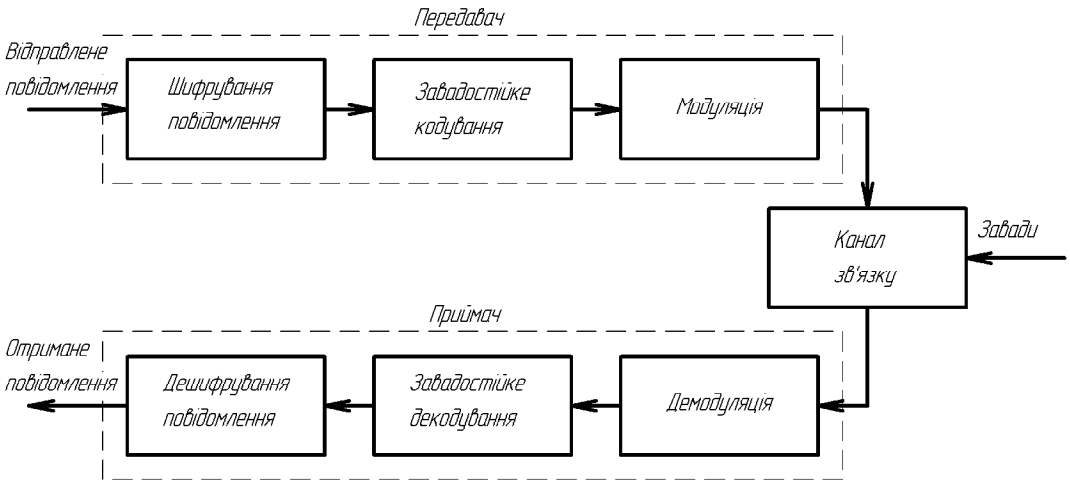


Рис 3.1. Структурна схема системи зв'язку.

Система зв'язку призначена для передачі повідомлень від одного пункту до іншого через канал зв'язку з певними характеристиками. Для вирішення цієї задачі необхідно виконати ряд трансформацій.

Початкове повідомлення спершу піддається первинному кодуванню (кодуванню джерела) з метою перетворення аналогового повідомлення в цифрове або стискання інформації. Наступним кроком є завадостійке кодування, де в повідомлення вводиться додаткова інформація для можливості виправлення помилок на приймальній стороні.

Після застосування завадостійкого коду повідомлення передається у модулятор, який перетворює цифрове повідомлення в аналоговий модульований сигнал, займаючи певний діапазон частот. Під час передачі сигналу через канал зв'язку відбувається вплив шумів і перешкод, і спотворений сигнал надходить на вхід приймача.

Приймальна частина має структуру, яка є відображенням структури передавача. Сигнал проходить через блоки в зворотньому порядку, здійснюючи зворотні трансформації порівняно з передавачем. Спочатку сигнал піддається демодуляції, в процесі якої аналоговий модульований сигнал перетворюється в цифрове повідомлення.

Далі відбувається декодування завадостійкого коду, що дозволяє виправити частину або всі помилки, що виникли в процесі передачі. Після виправлення помилок проводиться декодування джерела для відновлення початкового повідомлення.

Згідно з поданою структурою, можна виділити основні категорії функцій, які реалізовані як у Communications Toolbox, так і в Communications Blockset. Ці категорії включають:

— Функції для моделювання та аналізу сигналів.
— Функції для кодування та декодування джерела.
— Функції для кодування та декодування каналу зв'язку, включаючи завадостійке кодування та декодування.
— Функції для модуляції та демодуляції.

— Функції для моделювання каналів зв'язку.

Крім того, в пакеті розширення Communications Toolbox і наборі блоків Communications Blockset доступні додаткові функції та бібліотеки, які реалізують більше спеціалізовані можливості.

3.1.1. Моделювання та шифрування сигналів.

Пакет розширення Communications розширює можливості MATLAB для моделювання телекомунікаційних систем. Він пропонує широкий спектр функцій, що дозволяють користувачам:

Моделювати та аналізувати сигнали, генерувати, обробляти та візуалізувати сигнали, як аналогові, так і цифрові.

Кодувати та декодувати джерела, стискати дані для економії пропускної здатності каналу зв'язку.

Використовувати завадостійке кодування/декодування, захищати дані від помилок, що виникають при передачі по каналу зв'язку.

Модулювати та демодулювати сигнали, перетворювати цифрові дані в аналогові сигнали для передачі по каналу зв'язку та декодувати їх на приймаючому боці.

Моделювати канали зв'язку, досліджувати вплив різних типів каналів зв'язку на передачу даних.

Для моделювання телекомунікаційної системи важливо мати можливість генерувати сигнали та шуми. Шуми та повідомлення, які передаються, є випадковими процесами. Їх відмінність полягає в статистичних характеристиках.

Функції для генерації випадкових даних:

— Функція randint — створює матрицю випадкових чисел, рівномірно розподілених у заданому інтервалі.

— Функція randsrc — більш гнучка функція, яка дозволяє задавати алфавіт (множину цілих чисел) та ймовірності появи символів в повідомленні.

— Функція randerr — генерує помилки в цифровому сигналі. Створює матрицю, де кожен рядок містить задану кількість випадково розташованих ненульових елементів.

— Функція wgn — моделює дискретний білий гаусівський шум із заданою потужністю.

Оцінка завадостійкості:

Для оцінки завадостійкості системи зв'язку необхідно порівняти вихідне (передане) повідомлення з отриманим на прийомі. Це робиться за допомогою функцій: функція symerr — підраховує кількість незбіжних символів у двох повідомленнях, функція biterr — підраховує кількість незбіжних бітів у двійкових поданнях цих символів.

Ці функції також можуть повертати частку помилок та індикатори місць їх виникнення.

Функції кодування/декодування джерела виконують різноманітні операції для обробки інформації. Кодування джерела, або source coding, має на меті перетворення вихідного повідомлення в формат, придатний для передачі.

У пакеті Communications реалізовано функції, які виконують такі операції кодування/декодування джерела:

— Нерівномірне квантування та оптимізація його параметрів.

— Логарифмічне та експонентне перетворення.

— Диференціальна імпульсно-кодова модуляція (ДІКМ) та оптимізація параметрів цієї модуляції.

Функція quantiz використовується для реалізації нерівномірного квантування. Функція dpcmenco використовується для диференціального кодування повідомлення, з заданими коефіцієнтами передбачуваного фільтра та параметрами квантування помилки прогнозування. Функція dpcmdeco відновлює вихідне повідомлення.

Пакет Communications також підтримує роботу з різними лінійними блоковими кодами, включаючи циклічні коди, Боуза-Чоудхурі-Хоквінгема (БЧХ), коди Гемінга та коди Ріда-Соломона.

Функції високого рівня, такі як encode і decode, виконують кодування та декодування повідомлення з використанням блокового коду. Тип використаного коду визначається серед параметрів цих функцій.

Функція gfweight дозволяє визначити кодову відстань для лінійного блокового коду за його породжувальним поліномом або перевірочною матрицею.

Для циклічних кодів у пакеті Communications є функції cyclpoly для отримання породжувального поліному циклічного коду та cyclgen для генерації перевірочної матриці для даного коду.

Коди БЧХ та Гемінга, як підкласи циклічних блокових кодів, обробляються спеціалізованими функціями bchenco, bchdeco та hammgen.

Для кодів Ріда-Соломона використовуються функції rsenco, rsdeco, rsencode, rsdecode, rsencof, rsdecof та rspoly для виконання операцій кодування, декодування та генерації поліномів.

3.1.2. Модуляція й демодуляція.

Пакет Communications включає функції для виконання аналогової та цифрової модуляції та демодуляції. У залежності від типу сигналу (аналоговий чи цифровий), доступно вісім функцій:

— Функція amod — Аналогова модуляція звичайним аналоговим вихідним сигналом.
— Функція amodce — Аналогова модуляція з вихідним сигналом у вигляді комплексної огинаючої.
— Функція ademod — Аналогова демодуляція звичайним аналоговим вхідним сигналом.
— Функція ademodce — Аналогова демодуляція з вхідним сигналом у вигляді комплексної огинаючої.
— Функція dmod — Цифрова модуляція звичайним цифровим вихідним сигналом.

— Функція dmodce — Цифрова модуляція з вихідним сигналом у вигляді комплексної огинаючої.
— Функція ddemod — Цифрова демодуляція звичайним цифровим вхідним сигналом.
— Функція ddemodce — Цифрова демодуляція з вхідним сигналом у вигляді комплексної огинаючої.

Процес цифрової модуляції та демодуляції включає дві стадії. При модуляції, цифрове повідомлення конвертується в аналоговий сигнал за допомогою функції modmap, і потім відбувається аналогова модуляція. При демодуляції, спочатку отримуємо аналоговий демодульований сигнал, який потім перетворюється в цифрове повідомлення за допомогою функції demodmap.

Три останні функції спеціалізовані на роботу з конкретними типами сигнальної квадратурної маніпуляції. Функції qaskenco і qaskdeco виконують кодування та декодування повідомлення з використанням "квадратного" типу, а функція apkconst виводить на екран зображення "концентричного" типу.

3.1.3. Моделювання каналів зв'язку .

Пакет Communications в Matlab надає широкий спектр функцій для моделювання каналів зв'язку. Ці функції можна використовувати для:

Створення моделей різних типів каналів зв'язку, таких як канали з затуханням, канали з шумом і канали з множинним доступом.

Розробка алгоритмів для боротьби з впливом каналів зв'язку.

Канал з затуханням — це канал, який вносить затухання в сигнал, що передається. Це може бути спричинено різними факторами, такими як відстань, перешкоди та затухання.

Функція rayleighchan в Matlab можна використовувати для створення моделі каналу з затуханням Релея. Цей тип каналу часто використовується для моделювання мобільних каналів зв'язку.

Канал з шумом – це канал, який додає шум до сигналу, що передається. Це може бути спричинено різними факторами, такими як тепловий шум і перешкоди від інших систем зв'язку.

Функція awgnchan в Matlab можна використовувати для створення моделі каналу з гаусовим шумом. Цей тип каналу часто використовується для моделювання каналів зв'язку з фіксованим зв'язком.

Канал з множинним доступом – це канал, де кілька користувачів спільно використовують один і той самий канал. Це може призвести до колізій, коли сигнали від різних користувачів перешкоджають один одному.

Функція fdmachan в Matlab можна використовувати для створення моделі каналу з множинним доступом з частотним поділом (FDMA). Цей тип каналу часто використовується для моделювання мобільних систем зв'язку.

3.1.4. Спеціальні фільтри.

В безлічі систем зв'язку використовуються різноманітні фільтри з різними завданнями. В основному для обчислень фільтрів у середовищі MATLAB використовуються функції, що належать пакетам Signal Processing і Filter Design. Таким чином, пакет Communications містить лише функції, спрямовані на розрахунок двох спеціалізованих типів фільтрів.

Функція hilbiir виконує розрахунки рекурсивного фільтра, який наближено відтворює перетворення Гільберта. Фільтр Гільберта вносить постійне фазове зрушення на всіх частотах, рівне 90°, при цьому зберігаючи амплітуди всіх спектральних складових. Цей фільтр застосовується у функції аналогової модуляції amod для формування сигналу з односмуговою модуляцією. Крім того, перетворення Гільберта використовується для обчислення уявної частини комплексного аналітичного сигналу з однобічним спектром.

Функція rcosine виконує розрахунки фільтра з косинусоїдальним згладжуванням амплітудно-частотної характеристики (raised cosine filter), який широко використовується для формування спектра сигналу у квадратурній

маніпуляції. Для розрахунків використовуються дві функції нижчого рівня: rcosfir (безрекурсивний варіант) і rcosiir (рекурсивний варіант).

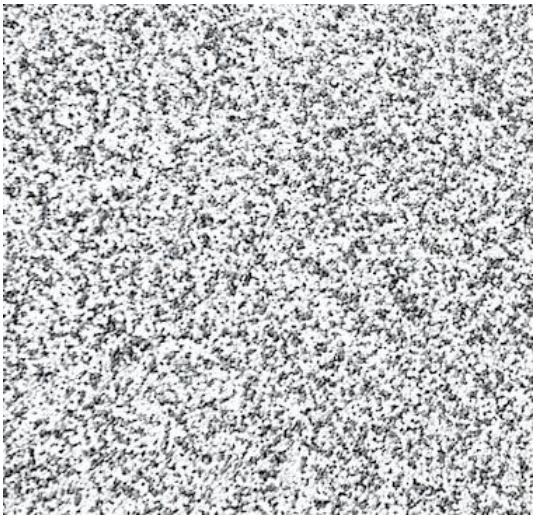
Функція rcosflt збільшує частоту дискретизації сигналів до цілого числа разів, використовуючи інтерполяцію із застосуванням фільтра з косинусоїдальним згладжуванням АЧХ.

3.2. Моделювання систем системи передавання фотографічних зображень із застосуванням симетричного шифрування AES-128

Для моделювання систем передавання зображень із застосуванням симетричного шифрування AES-128, використано пакет Matlab. У цьому пакеті розроблено програмне забезпечення див. додаток Б.

Для моделювання систем передавання зображень з OFDM модуляцією застосуємо як вхідний сигнал вибране тестове зображення, див рис.3.2.

а)



б)

Рис. 3.2. Зображення для тестування систем передавання зображень: а) не зашифроване; б) зашифроване за допомогою алгоритму AES-128.

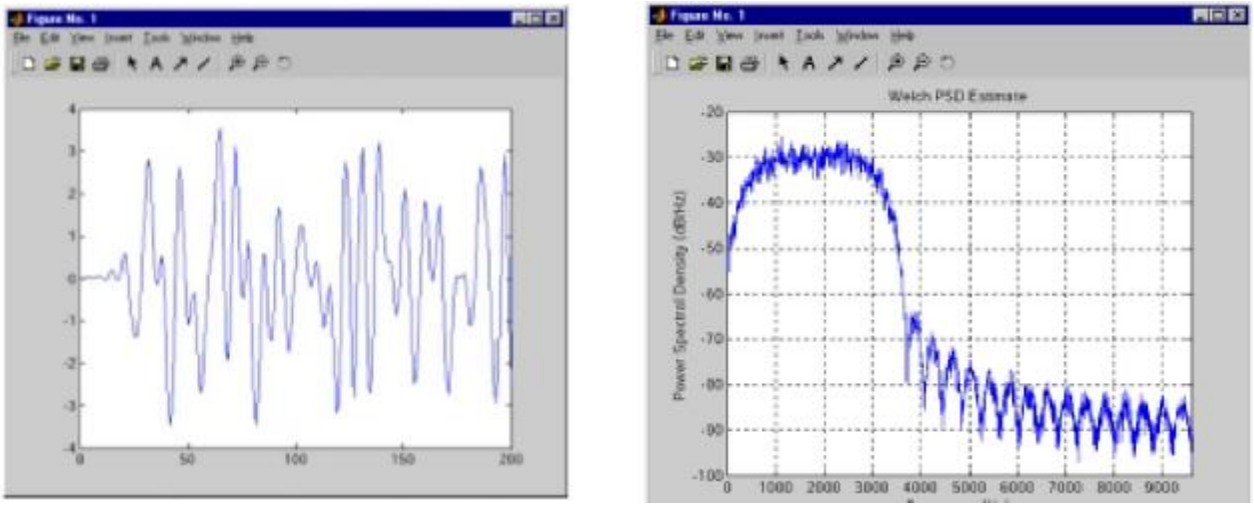
Загалом у вбудованих системах функція шифрування та дешифрування алгоритму AES-128 знаходиться за межами цифрових приймачів і передавачів,

тому для випробування тестове зображення в одному випадку не шифруємо (див. рис.3.2, а), а в другому випадку за допомогою алгоритму AES-128 шифруємо (див. рис.3.2, б) використовуючи розроблену програму див. додаток Г.

Відповідно на виході приймача будемо порівнювати параметри BER для зображення без шифрування і з шифруванням.

3.2.1 Моделювання впливу каналу зв'язку

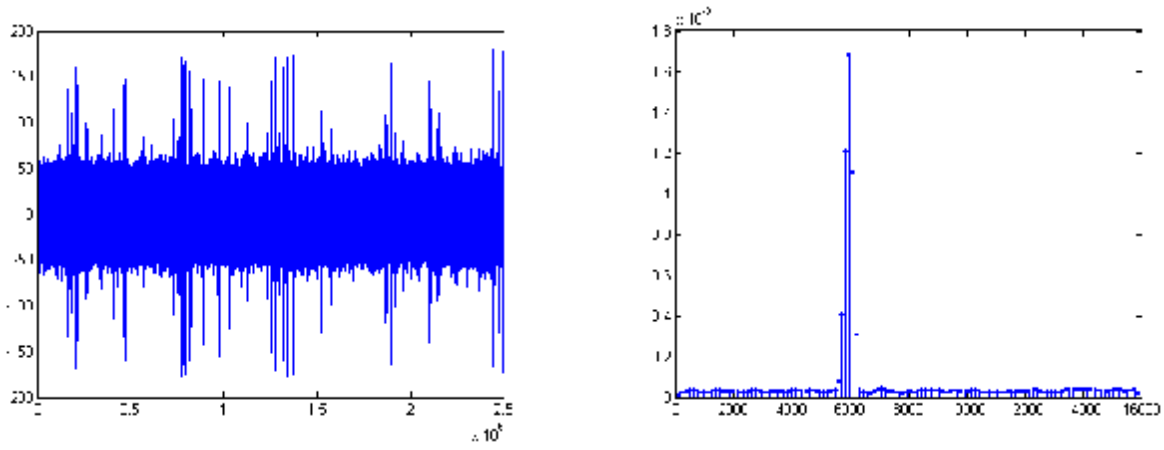
Для моделювання впливу каналу зв'язку використаємо запропоновані моделі у 2-у розділі. Зокрема використаємо передавач з OFDM модуляцією сигнал на виході якого є багатоканальним, причому сигнал для кожного модулюється QAM квадратурною модуляцією, див рис.3.3.



а) б)

Рис. 3.3. Згенерований сигнал передавача з OFDM модуляцією та використанням QAM (а) та його спектральна густина потужності (б).

При передаванні сигналу через канал зв’язку до нього додається шум оточуючого середовища, згідно моделі це білий шум, див рис.3.4.



а) б)

Рис. 3.4 Суміш сигналу і шуму в каналі зв’язку: а) зашумлений сигнал (вісь абсцис – час с., вісь ординат – напруга мВ.); б) спектр потужності зашумленого сигналу (вісь абсцис – частота Гц., вісь ординат – потужність мВт.)

Моделювання впливу каналу зв'язку є невід'ємною частиною проектування та аналізу систем приймання-передачі даних. Цей процес дозволяє дослідити, як характеристики каналу впливають на якість переданої інформації, та розробити алгоритми, що роблять систему стійкою до помилок і завад.

Канал з адитивним гаусовим білим шумом (AWGN): модель, що використовується для опису каналів з лінійними характеристиками та випадковим шумом, що має нормальний розподіл. AWGN є типовою моделлю для каналів зв'язку з фіксованим зв'язком, таких як телефонні лінії.

Релеевський канал: модель, що використовується для опису каналів з множинним шляхом поширення сигналу, де амплітуда сигналу зазнає випадкових змін, що підкоряються релеевському розподілу. Релеевський канал є типовою моделлю для мобільних каналів зв'язку, де сигнал може відбиватися від будівель, дерев та інших об'єктів.

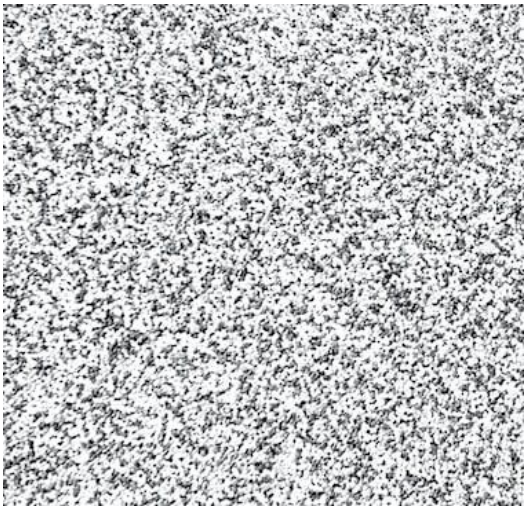
Розроблена модель враховує вплив каналу зв'язку на параметри сигналу, такі як потужність, амплітуда та фаза. Модель також включає параметр усікання

амплітуди сигналу, який може бути спричинений нелінійними характеристиками каналу або обмеженнями приймача.

Результати моделювання дозволяють зробити висновки про вплив типу каналу зв'язку та параметрів сигналу на якість передачі даних. Наприклад, модель може показати, як збільшення потужності сигналу або використання більш стійкого до помилок кодування може покращити якість приймання в каналі AWGN.

Отримане на стороні приймача та дешифроване зображення наведено на рис.3.5.

а)



б)

в)

Рис. 3.5. Отримані зображення на виході приймача: а) не зашифроване; б) зашифроване (AES-128); в) дешифроване (AES-128).

Моделювання впливу каналу зв'язку є важливим інструментом для проектування та аналізу систем приймання-передачі даних. Розроблена модель може використовуватися для оцінки параметрів каналу, розробки алгоритмів, аналізу впливу різних факторів на якість передачі даних, а також для оптимізації характеристик системи зв'язку.

Шум на зображенні який міг виникнути через помилки при передачі зображення відсутні, оскільки моделювання передавання зображення виконувалось при високому відношенні сигнал/шум (10дБ).

3.3 Оцінювання ефективності системи передавання фотографічних зображень із шифруванням AES-128.

Оцінювання ефективності системи передавання зображення із шифруванням AES-128 проводиться засобами Matlab. Вихідними параметрами для налаштування в Matlab є: тестове зображення у відтінках сірого із градацією Nword = 8 біт, з розмір зображення по висоті h = 512, ширині w = 512.

За отриманими результатами побудовано сигнальні сузір'я для QAM-16 при різних значеннях сигнал-шум (SNR), див. рис. 3.6.

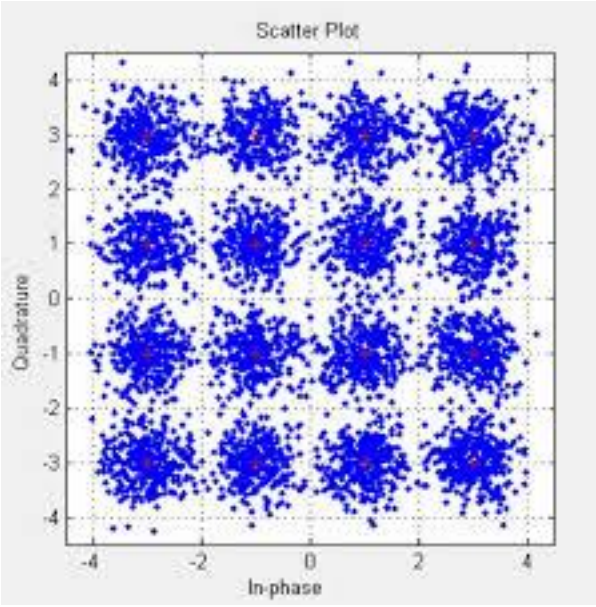


Рис. 3.6. Побудоване сигнальне сузір'я отриманого зображення для QAM-16 (Snrdb = 10 дБ).

На рисунку 3.6 можна спостерігати, що при зменшенні відношення сигнал-шум точки на сигнальному сузір'ї зсуваються всторону від центру. Це призводить до формування менш чіткої області навколо відповідних значень сигналу, що викликає помилки демодуляції даних та відповідно і шуму на зображенні.

Для оцінки ефективності передачі зображення використовуються характеристики символьної помилки (SER) від енергії при передачі даних, нормованої до спектральної густини шуму (Es/N0) [9].

Такі характеристики символьної помилки від відношення енергії передачі даних до спектральної густини шуму (Es/N0) наведено на рисунку 3.7, їх наведено для різних типів модуляції. Неперевні лінії на характеристиках відображають теоретичні результати для відповідних видів модуляції QAM [1].

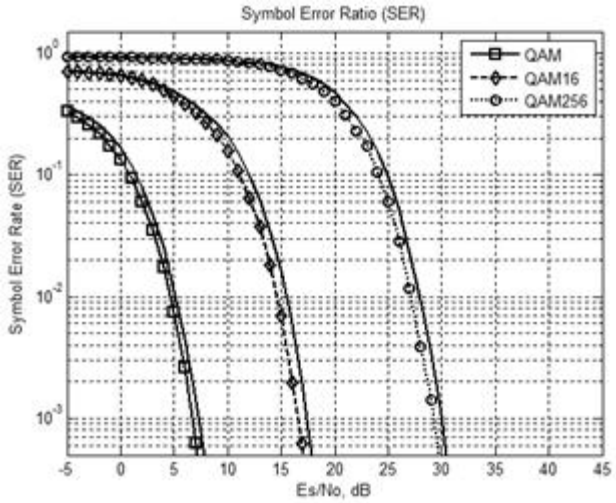


Рис. 3.7. Характеристики символьної помилки для різних типів модуляції QAM (Ns = 1, 4, 8).

Із наведених на рис.3.7 характеристик робимо висновок, що для більш складних видів QAM модуляції помилка демодулювання символу при заданому відношенні Es/N_0 буде меншою. Тому для збільшення швидкості передачі зображень необхідно підвищувати порядок модуляції,та збільшувати відношення сигнал шум.

Також за результатами моделювання отримано характеристики бітової помилки BER від Es/N_0 для OFDM, представлена на рис. 3.8.

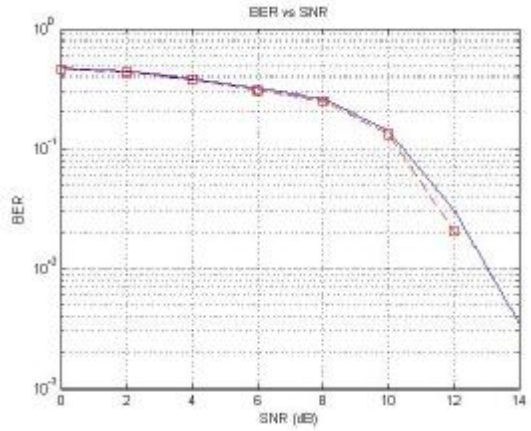


Рис. 3.8. Характеристики бітової помилка BER для QAM-16 при різних відношеннях сигнал/шум, для випадку зашифрованого (штрихова лінія) і для нешифрованого зображення (суцільна лінія).

Отримані характеристики свідчать що для передавання зображення процес шифрування не впливає на величину BER, а відхилення на графіках вкладається в межі помилки при зростанні відношення сигнал/шум.

3.4 Висновки до розділу 3

Для моделювання методів симетричного шифрування та систем передавання зображень зв’язку використано програмні засоби Signal Processing Toolbox та Comunication Toolbox в MATLAB, що спростило процедуру моделювання.

Отримані характеристики свідчать що для передавання зображення процес шифрування не впливає на величину BER, а відхилення на графіках вкладається в межі помилки при зростанні відношення сигнал/шум.

РОЗДІЛ 4
ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ

4.1 Охорона праці

Кваліфікаційна робота магістра пов’язана із дослідженням методів та засобів побудови комп’ютерної системи для потокового шифрування та передавання фотографічних зображень виконується на ПК, тому важливо дотримуватись вимог охорони праці при експлуатації ПК.

При розробці інструкції з охорони праці необхідно виконувати санітарні правила і норми ДСанПІН 3.3.2.007-98 “Державні санітарні правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин”.

Заходи з безпеки працівників мають відповідати вимогам НПАОП 0.00-7.15-18 “Вимоги щодо безпеки та захисту здоров’я працівників під час роботи з екранними пристроями”.

Директива Ради Європейських Співтовариств 89/391/ЕЕС «Про впровадження заходів, що сприяють поліпшенню безпеки й гігієни праці працівників»

Для приладів, які працюють у складі складних вимірювальних комплексів, функціональних кабінетів, обчислювальних центрів, обладнаних різноманітною електронною технікою.

Розроблювана комп’ютерна система повинна бути безпечною при всіх передбачених функціональними можливостями і вказаних у правилах користування умовах її експлуатації. Захист досягається дотриманням таких основних вимог:

1) правильною конструкцією апарата, яка гарантує безумовну безпеку;

2) використанням спеціальних засобів зовнішнього захисту, які забезпечують умовну безпеку;

3) вказівкою умов, за яких робота з обладнанням є безпечною (описова безпека).

За способом захисту персоналу і пацієнта від електроудару і електротравми все устаткування, яке використовує зовнішнє живлення, ділиться на п’ять класів (окремо виділяється устаткування з внутрішніми джерелами живлення, наприклад, батареями).

Роботодавець повинен створити для кожного працівника безпечні і нешкідливі умови праці шляхом належного облаштування робочих місць відповідно до Загальних вимог стосовно забезпечення роботодавцями охорони праці працівників, затверджених наказом Міністерства надзвичайних ситуацій України від 25 січня 2012 року № 67, зареєстрованих у Міністерстві юстиції України 14 лютого 2012 року за № 226/20539 (НПАОП 0.00-7.11-12).

Параметри мікроклімату в межах робочої зони повинні відповідати вимогам Санітарних норм мікроклімату виробничих приміщень ДСН 3.3.6.042-99. Рівень шуму на робочих місцях повинен відповідати нормам, встановленим Санітарними нормами виробничого шуму, ультразвуку та інфразвуку ДСН 3.3.6.037-99.

Загальні вимоги безпеки до захисту від шуму на робочих місцях, шумові характеристики машин та механізмів повинні відповідати вимогам. А роботодавець повинен здійснювати контроль рівня шуму відповідно до вимог ДСТУ 2867-94 «Шум. Методи оцінювання виробничого шумового навантаження. Загальні вимоги».

Рівень вібрації на робочих місцях не повинен перевищувати норм, встановлених Державними санітарними нормами виробничої загальної та локальної вібрації ДСН 3.3.6.039-99.

Параметри електромагнітних полів на робочих місцях повинні відповідати вимогам Державних санітарних норм і правил при роботі з джерелами електромагнітних полів, затверджених наказом Міністерства охорони здоров’я України від 18 грудня 2002 року № 476, зареєстрованих у Міністерстві юстиції України 13 березня 2003 року за № 203/7524 (ДСН 3.3.6.096-2002).

У робочій зоні виробничих приміщень вміст шкідливих речовин не повинен перевищувати граничнодопустимих концентрацій, встановлених ГОСТ 12.1.005-88 Загальні санітарно-гігієнічні вимоги до повітря робочої зони.

Забороняється захаращувати робочі місця готовою продукцією, матеріалами, деталями і предметами, які не використовуються у процесі виробництва.

Площа робочої поверхні столу повинна забезпечувати зручне розміщення технологічного устаткування, приладів та інструментів з урахуванням зони досяжності працівника в горизонтальній і вертикальній площинах.

Контрольно-вимірювальні прилади повинні відповідати вимогам ДСТУ EN 55011:2017 «Обладнання промислове, наукове та медичне. Характеристики радіочастотних завад. Норми та методи вимірювання».

Температура нагрітих поверхонь устаткування та огороджень не повинна перевищувати +43 °С згідно з вимогами ДСТУ EN 563-2001 «Безпечність машин. Температури поверхонь, доступних для дотику. Ергономічні дані для встановлення граничних значень температури гарячих поверхонь».

Не дозволяється виконання робіт з використанням легкозаймистих і горючих рідин у приміщеннях, які не обладнані припливно-витяжною вентиляцією.

Вимоги безпеки оточуючого середовища при експлуатації комп'ютера грають ключову роль у забезпеченні ефективності та безпеки робочого простору. Ось деякі основні вимоги які необхідно врахувати: вентиляція, температурний режим, контроль вологості, захист від статичної електрики, захист від води, стабільність живлення, безпека кабелів, електробезпека, освітлення, акустичний комфорт, система пожежогасіння.

Забезпечте ефективну систему вентиляції для уникнення перегріву комп'ютерної техніки. Займайтеся регулярним обслуговуванням та очищенням систем охолодження.

Утримуйте стабільний температурний режим в приміщенні, де знаходиться комп'ютерна техніка, для попередження перегріву чи надмірного охолодження.

Уникайте екстремально високого або низького рівня вологості, оскільки це може спричинити корозію та інші пошкодження електронних компонентів.

Встановіть антистатичні килими або мати для захисту комп'ютерної техніки від пошкоджень, спричинених статичним електричним розрядом.

Уникайте розливання напоїв чи інших рідин на комп'ютер та його аксесуари. Застосовуйте захисні кришки на клавіатурі та миші.

Забезпечте стабільність напруги та струму в електромережі, використовуйте стабілізатори напруги, якщо потрібно.

Організуйте кабелі таким чином, щоб уникнути їхнього плутання та розтягування. Зафіксуйте їх так, щоб вони не становили небезпеку для персоналу.

Використовуйте електротехнічне обладнання, яке відповідає стандартам електробезпеки. Заземлення та ізоляція є ключовими аспектами.

Встановіть вогнегасники або систему пожежогасіння в приміщенні, де розташований комп'ютерний обладнання.

Забезпечте оптимальний акустичний комфорт у робочому просторі, особливо якщо використовуються обладнання з великим рівнем шуму.

Забезпечте належне освітлення, щоб уникнути напруження очей під час роботи на комп'ютері.

Загальна мета вимог безпеки оточуючого середовища полягає в забезпеченні довговічності та ефективності роботи комп’ютера.

4.2 Функціонування державної системи спостереження, збирання, оброблення та аналізу інформації про стан довкілля під час надзвичайних ситуацій мирного та воєнного часу.

Забезпечення ефективного функціонування державної системи спостереження за станом довкілля є важливим аспектом забезпечення національної безпеки та виявлення надзвичайних ситуацій як у мирний, так і воєнний час. Ця система включає в себе збір, обробку та аналіз інформації про стан довкілля з метою вчасного виявлення потенційних загроз та прийняття ефективних заходів для захисту населення та навколишнього середовища.

Завданнями державної системи спостереження є:

Моніторинг та прогнозування надзвичайних ситуацій, шляхом збирання та аналізу даних про стан довкілля для вчасного виявлення природних катастроф, техногенних аварій чи інших небезпек.

Спостереження за забрудненням та захистом навколишнього середовища, а також виявлення джерел забруднення, моніторинг стану водних, повітряних та ґрунтових ресурсів для запобігання та ліквідації забруднень.

Контроль за рухом забруднюючих речовин, шляхом виявлення та відстеження руху небезпечних речовин у випадку техногенних аварій чи інших подій, що можуть призвести до надзвичайних ситуацій.

Реагування на надзвичайні ситуації та розробка і впровадження стратегій та планів реагування на різні типи надзвичайних ситуацій для максимальної ефективності та мінімізації наслідків.

Структура та організація системи передбачає об’єднання багатьох технічних та організаційних систем. Державна система спостереження за станом довкілля включає в себе мережу датчиків, обчислювальні центри для обробки даних, а також високотехнологічні засоби зв'язку. Організаційною структурою може бути централізована або розподілена система, залежно від вимог та можливостей кожної конкретної держави.

Сучасні технології грають визначальну роль у розвитку та функціонуванні системи спостереження за станом довкілля. Основні технологічні інструменти включають:

Дистанційне зондування Землі, шляхом використання супутників та аерокосмічних апаратів для отримання зображень та даних про стан довкілля,

що надає можливість виявлення змін в рельєфі, водних ресурсах та інших аспектах.

Системи автоматизованого моніторингу передбачає розгортання сучасних датчиків та датчикових мереж для неперервного моніторингу різних параметрів навколишнього середовища, таких як рівень забруднення, температура, вологість тощо.

Системи інформаційної безпеки передбачають захист зібраної інформації від несанкціонованого доступу та забезпечення надійності передачі даних є критичними аспектами для забезпечення цілісності системи.

Штучний інтелект та аналітика даних враховує використання алгоритмів штучного інтелекту для аналізу великих обсягів даних та виявлення паттернів, що може значно полегшити виявлення потенційних загроз.

Застосування інтегрованих систем, які поєднують усі технологічні аспекти, дозволяє забезпечити комплексний підхід до спостереження за станом довкілля. Узагальнені системи дозволяють в реальному часі виявляти та реагувати на загрози, забезпечуючи ефективне управління надзвичайними ситуаціями.

Незважаючи на досягнення, перед системою стоять виклики та перспективи.

Кіберзагрози та безпека інформації, тобто захист від кібер атак та забезпечення безпеки інформації вимагає постійного вдосконалення технічних та організаційних заходів.

Глобальне співробітництво потребує розвитку міжнародної співпраці у галузі спостереження за довкіллям для спільного виявлення та вирішення глобальних проблем.

Інновації та дослідження передбачає здійснення постійних досліджень та впровадження новітніх технологій для вдосконалення ефективності системи та відповіді на нові загрози.

Розробка та удосконалення державної системи спостереження за станом довкілля є невід’ємною частиною стратегії забезпечення національної безпеки

та стійкості у надзвичайних ситуаціях. Використання сучасних технологій та ретельне планування можуть значно підвищити ефективність системи та забезпечити швидке та адекватне реагування на потенційні загрози довкілля.

Освітлення під час роботи на комп'ютері важливо для забезпечення комфорту, збереження зору та підтримки продуктивної робочої атмосфери. Ось кілька порад щодо належного освітлення у робочому середовищі з комп'ютером:

Натуральне світло: Розташовуйте робоче місце так, щоб ви могли користуватися натуральним світлом. Вікно повинно бути розташоване біля робочого столу, але при цьому уникається блиск сонця на екрані.

Якщо вікно розташоване прямо перед робочим місцем, використовуйте штори чи жалюзі для регулювання яскравості світла.

Доповнюйте натуральне світло штучним, особливо вночі чи в темний час доби. Використовуйте стелеве освітлення та настільні лампи.

Розмістіть антивідблисковий екран на моніторі для зменшення відблисків і блисків від інших джерел світла.

Забезпечте рівномірне освітлення приміщення, уникайте раптових змін яскравості. Це може бути досягнуто за допомогою різних джерел світла.

Вибирайте лампи з природним білим світлом, близьким до денного світла. Вони допомагають зберегти реалістичність кольорів на екрані.

Використовуйте регульовані світильники або лампи, які дозволяють вам контролювати яскравість світла в залежності від потреби.

Встановлюйте лампи так, щоб вони не відбивали світло безпосередньо на екран комп'ютера.

Забезпечте, щоб світильники не створювали тіні на робочому столі, і вони були розташовані на відстані від екрану.

Робіть короткі паузи, особливо якщо працюєте в темному приміщенні. Вони дозволяють очам відпочити і запобігають втомленості.

Дотримання цих порад допоможе забезпечити комфортне та безпечне освітлення під час роботи на комп'ютері, сприяючи збереженню зору та підвищенню продуктивності.

4.3 Підвищення стійкості роботи комп’ютеризованих систем в умовах дії ЕМІ ядерних вибухів.

З розвитком сучасних технологій та комп’ютеризації у всіх сферах людської діяльності виникає необхідність в підвищенні стійкості комп’ютеризованих систем в умовах електромагнітних перешкод, що можуть бути викликані ядерними вибухами. Електромагнітні перешкоди, або ЕМІ (електромагнітна інтерференція), можуть значно вплинути на нормальне функціонування комп’ютерних систем, призводячи до втрати даних, збоїв у роботі апаратного забезпечення, або навіть до повного відмови системи.

Однією з критичних ситуацій, яка може викликати ЕМІ, є ядерний вибух. Інтенсивний потік електромагнітної енергії, яка виникає під час вибуху, може вразити електронні системи, перериваючи їхню роботу та призводячи до серйозних наслідків.

ЕМІ, породжена ядерним вибухом, може викликати електричні струми та напруги в електронних системах, перебої в роботі радіоелектронних пристроїв та інтерференцію з сигналами передачі даних. Це створює великий виклик для забезпечення надійності та стійкості комп’ютерних систем у таких умовах.

Стратегії підвищення стійкості комп’ютеризованих систем в умовах дії ЕМІ повинні передбачати комплексний підхід до проєетування і реалізації таких систем.

Для підвищення стійкості комп’ютеризованих систем в умовах дії ЕМІ ядерних вибухів, розробники повинні приділяти увагу кільком ключовим аспектам.

Електромагнітно сумісне проектування (ЕМС) передбачає ретельне планування та дизайн систем з урахуванням ЕМС може допомогти зменшити

електромагнітну чутливість пристроїв і забезпечити їх стабільну роботу під час ядерних вибухів.

Використання захисних екранів та фільтрів шляхом встановлення екранів та фільтрів може зменшити вплив EMI на комп'ютерні системи, створюючи бар'єр між електромагнітними перешкодами та обладнанням.

Застосування резервного живлення та використання джерел живлення з резервними модулями та джерелами або акумуляторами може забезпечити неперервну роботу системи під час збоїв в основному електропостачанні.

Сучасні технологічні рішення в галузі підвищення стійкості комп'ютерних систем в умовах дії EMI ядерних вибухів націлені на поєднання попередньо зазначених стратегій та впровадження новаторських технологій.

Використання квантового захисту тобто розробка квантових комп'ютерів та квантових методів шифрування може зробити комп'ютеризовані системи менш чутливими до електромагнітних перешкод.

Впровадження інтелектуальних систем управління шляхом використання штучного інтелекту та автоматизованих систем управління може допомогти виявляти та компенсувати ефекти EMI на ходу, забезпечуючи неперервну роботу системи.

Розробка більш стійкого апаратного забезпечення, тобто використання нових матеріалів та технологій у виробництві апаратного забезпечення може зробити пристрої менш вразливими до електромагнітних впливів.

Підвищення стійкості комп'ютеризованих систем в умовах дії EMI ядерних вибухів вимагає комплексного підходу, що охоплює як традиційні стратегії, так і новітні технології. Інноваційні рішення, такі як квантовий захист та інтелектуальне управління, спільно з традиційними методами захисту, можуть забезпечити високий рівень стійкості комп'ютерних систем в умовах надзвичайних ситуацій, забезпечуючи надійну роботу систем навіть під впливом небезпеки ядерних вибухів.

ВИСНОВКИ

В кваліфікаційній роботі магістра розроблено методи та засоби побудови комп’ютерної системи для потокового шифрування та передавання фотографічних зображень

1. Алгоритм шифрування AES є подібним до потокових шифрів і відзначається високою швидкістю шифрування, що робить його сприятливим для застосування в різних сферах, зокрема, в цифрових системах зв'язку. Зокрема, в алгоритмах OFDM (Orthogonal Frequency Division Multiplexing), які використовуються в стандартах Wi-Fi, WiMAX, DVB-T2 та інших, застосовується модуляція з використанням BPSK, QPSK, 8-PSK, QAM та інших типів.

2. Наведена математична модель каналу зв'язку є загальною для значної кількості випадків передачі даних, та враховує вплив завад від середовища у якому відбувається поширення сигналів та багатопроменевого поширення радіохвиль разом із пов'язаними з цим, явищами завмиранн в каналі зв'язку.

3. Вибрана модель системи зв'язку передавання фотографічних зображень на основі OFDM з використанням квадратурної амплітудної модуляції дозволяє проводити моделювання системи зв'язку з використанням Matlab і додавати до неї різноманітні алгоритми шифрування.

4. Для моделювання методів симетричного шифрування та систем передавання зображень зв’язку використано програмні засоби Signal Processing Toolbox та Comunication Toolbox в MATLAB, що спростило процедуру моделювання.

5. Отримані характеристики свідчать що для передавання зображення процес шифрування не впливає на величину BER, а відхилення на графіках вкладається в межі помилки при зростанні відношення сигнал/шум.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Козарик Д., Лещишин Ю. Моделювання методів потокового шифрування та передавання фотографічних зображень / Матеріали XI науково-технічної конфції «Інформаційні моделі, системи татехнології» Тернопіль: ТНТУ ім. І. Пулюя, 2023. 157 с
2. Козарик Д., Лещишин Ю. Методи та засоби побудови комп’ютерної системи для потокового шифрування та передавання фотографічних зображень / Матеріали XI науково-технічної конфції «Інформаційні моделі, системи татехнології» Тернопіль: ТНТУ ім. І. Пулюя, 2023. 158 с
3. Ultra-wideband Wireless Communications and Networks Edited by Xuemin Shen et.al. USA: John Wiley & Sons, 2006. - 310 p.
4. Bernardo L., Lopes P.B. Quadrature chaotic symbolic OFDM communication over radio channels / "Communications (LATINCOM) 2012. IEEE Latin-America Conference", pp.1-6, 2012.
5. Al-Mahmoud M., Zoltowski M.D. Performance evaluation of Code-Spread OFDM with error control coding / "Military Communications Conference 2008 (IEEE MILCOM 2008) ". pp.1-6, 2008.
6. Proakis, John G. Digital communications / John G. Proakis, Masoud Salehi. 5th ed. 2008.1170p.
7. Kamilo Feher. Wireless Digital Communications: Modulation and Spread Spectrum Applications / Prentice Hall; Har/Dskt edition (May 17, 1995). - 544 p
8. Dogan H., Yildiz H., Cooklev T., Acar Y. Coded OFDM wireless systems with generalized prefix /"Application of Information and Communication Technologies (AICT) ", 2012 6th International Conference., pp.1-4. 2012.
9. Бабак В.П. та ін.. Обробка сигналів у радіоканалах цифрових систем передавання інформації. К.: книжкове видання НАУ, 2005, 476с.
10. Chenggao Han, Hashimoto T., Suehiro N. Constellation-rotated vector OFDM and its performance analysis over Rayleigh fading channels / Communications, IEEE Transactions, vol.58, no.3, pp.828-838, 2010.

11. Волощук Ю.І. Сигнали та процеси у радіотехніці: Підручник для студентів вищих навчальних закладів. Харків: «Компанія СМІТ», 2003. 444 с.
12. Філіпський Ю. К. Випадкові процеси у радіотехнічних колах. Навчальний посібник. Наука і техніка. URL: https://books.google.com.ua/books?id =KPXA GjvRmPcC&hl= ru&source=gbs_navlinks_s (дата звернення: 21.11.2023).
13. Радіотехніка: Енциклопедичний навчальний довідник: Навч. посібник / За ред. Ю. Л. Мазора, Є. А. Мачуського, В. І. Правди. К.: Вищ. шк., 1999. 838 с
14. Лещишин Ю. З., Романишин Н.Р., Наконечний В. В., Паламарчук А.О. Розробка системи зв'язку як інтегрованого елементу роботизованих систем // Зб. тез доповідей XXI Всеукр. наук.-пр. конф. Житомир, 2016. С. 102.
15. Марків В.А., Осухівська Г.М., Лещишин Ю.З., Луцків А.М. Комп'ютерна система аутентифікації осіб // Матеріали XX наукової конференції ТНТУ ім. І. Пулюя. 2017. С. 90–91.
16. Leschyshyn Y., Scherbak L., Nazarevych O., Gotovych V., Tymkiv P., Shymchuk G. Multicomponent Model of the Heart Rate Variability Change-point // IEEE XVth International Conference on the Perspective Technologies and Methods in MEMS Design (MEMSTECH). 2019. P. 110–113.
17. Tymkiv P., Leshchyshyn Y. Algorithm Reliability of Kalman Filter Coefficients Determination for Low-Intensity Electroretinosignal // IEEE 15th International Conference on the Experience of Designing and Application of CAD Systems (CADSM). 2019. P.1-5.
18. Геврик Є.О. Охорона праці. К.: Ельга, Ніка-Центр, 2003. 280 с.
19. Стадник І.Я., Зварич Н.М."Оцінка хімічної обстановки при аваріях на хімічно небезпечних об'єктах з викидом (виливом) небезпечних хімічних речовин та застосуванні хімічної зброї " ТНТУ, 2020. 36С.
20. Leschyshyn Y., Semchyshyn O. Periodically correlated heart rate variability detection by Neyman - Pearson criterion // 9th International Conference - The Experience of Designing and Applications of CAD Systems in Microelectronics. 2007. P. 139–140.

21. Лупенко С.А., д.т.н., проф.; Луцик Н.С., докт. філософ., доц.; Луцків А.М. к.т.н., доц.; Осухівська Г.М., к.т.н., доц.; Тиш Є.В., к.т.н. Методичні рекомендації до виконання кваліфікаційної роботи магістра // Затверджено на засіданні кафедри комп’ютерних систем та мереж, протокол №1 від 30 серпня 2021 р. 34.с.
22. НПАОП 0.00-1.28-10 «Правила охорони праці під час експлуатації ЕОМ». Наказ Держгірпромнагляду від 26.03.2010 No 6
23. Атаманчук П.С. Безпека життєдіяльності та охорона праці (Практичний курс): Навчальний посібник. Кам'янець-Подільський: "Думка". 2010. 152 с.

Додаток А
Опубліковані тези конференції за темою кваліфікаційної роботи магістра

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

ТЕРНОПІЛЬСЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ ІМЕНІ ІВАНА ПУЛЮЯ

М А Т Е Р І А Л И

XI НАУКОВО-ТЕХНІЧНОЇ КОНФЕРЕНЦІЇ

«ІНФОРМАЦІЙНІ МОДЕЛІ, СИСТЕМИ ТА ТЕХНОЛОГІЇ»

13-14 грудня 2023 року

ТЕРНОПІЛЬ
2023

УДК 004.056.55

Д. Козарик; Ю. Лещишин, к.т.н.
(Тернопільський національний технічний університет імені Івана Пулюя)

МЕТОДИ ТА ЗАСОБИ ПОБУДОВИ КОМП'ЮТЕРНОЇ СИСТЕМИ ДЛЯ ПОТОКОВОГО ШИФРУВАННЯ ТА ПЕРЕДАВАННЯ ФОТОГРАФІЧНИХ ЗОБРАЖЕНЬ

D. Kozaryk; Yu. Leshchyshyn, Ph.D.

METHODS AND MEANS FOR CONSTRUCTING A COMPUTER SYSTEM FOR STREAM ENCRYPTION AND TRANSMISSION OF PHOTOGRAPHIC IMAGES

Сучасне обладнання цифрового зв'язку для систем фото та відео нагляду активно використовуються у різноманітних сферах діяльності людини, від контролю якості виробництва до відео нагляду у охоронних системах та безпілотних літальних апаратів. Від цієї фото і відео інформації залежить якість виготовленої продукції, збереження майна та деколи і життя людини. Тобто така фото і відео інформація є цінною і має бути надійно захищеною від перехоплення та підміни зловмисниками.

Для захисту інформації від перехоплення використовують різноманітні методи та алгоритми шифрування, однак потоковий характер фото та і відео інформації зумовлює використання потокових алгоритмів шифрування з високим рівнем стійкості до зламу.

Специфіка використання фото і відео обладнання для таких задач потребує побудови портативних систем цифрового зв'язку, що використовують мікроконтролери з малим споживанням енергії та, як наслідок, невисокою обчислюваною потужністю. Тому для вирішення таких завдань використовують симетричні алгоритми шифрування, що ґрунтуються на єдиному ключі, який використовується для як шифрування, так і дешифрування [1]. У таких системах мікроконтролер, що відповідає за шифрування, діє як додатковий пристрій до цифрового модему, що уявує необхідність змінювати його конструкцію. Модулі шифрування повинні втілювати найбільш поширені методи апаратного шифрування, зокрема потоковий ARC4 та блочний AES-128. Ефективність цифрової системи зв'язку може бути оцінена за її стійкістю до завад, тобто за характеристиками прийому помилкового біта відповідно до виду модуляції при різних відношеннях сигнал-шум, як показано на рис. 1.

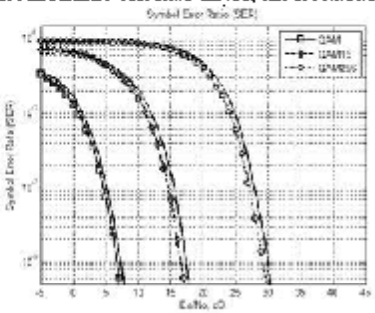


Рис. 1. Залежність приймання помилкового біта від виду модуляції, при застосуванні QAM (X_2 = 1, 4, 8).

Об'єднання зазначених технологій в одну цілісну комп'ютерну систему дозволяє надійно та ефективно передавати фото та відео інформацію засобами зв'язку. Подальше дослідження спрямоване на моделювання таких систем цифрового зв'язку, що уможливлює порівняння їх ефективності при застосуванні різних методів шифрування та передавання інформації, а також розробити портативні пристрої з високим рівнем захисту інформації.

Література

1. Whitfield Diffie and Martin Hellman, «Multi-user cryptographic techniques» [Diffie and Hellman, AFIPS Proceedings 45,1976].

УДК 004.056.55
Д. Козарик, Ю. Лещишин, к.т.н.
Тернопільський національний технічний університет імені Івана Пулюя, Україна

МОДЕЛЮВАННЯ МЕТОДІВ ПОТОКОВОГО ШИФРУВАННЯ ТА ПЕРЕДАВАННЯ ФОТОГРАФІЧНИХ ЗОБРАЖЕНЬ

D. Kozaryk; Yu. Leshchyshyn, Ph.D.

SIMULATION OF STREAM ENCRYPTION METHODS AND TRANSMISSION OF PHOTOGRAPHIC IMAGES

Сучасні засоби цифрового зв'язку для передачі фотографічних зображень, при сучасному розвитку технологій, ґрунтуються на цифрових модемах невеликої потужності та використанні різноманітних методів модуляції. Також їх все частіше використовують у різноманітних портативних та вбудованих комп'ютерних системах для передачі інформації та сигналів керування. З урахуванням необхідності захисту цієї інформації від стороннього втручання, використання криптографічних методів є одним із засобів забезпечення безпеки. Вибір конкретного методу шифрування, будь то симетричний чи асиметричний, залежить від переваг та недоліків, адаптованих до конкретної задачі. Після цього, важливо визначити, як змінюються характеристики системи зв'язку та оцінити її стійкість до завад.

Для побудови систем цифрового зв'язку для передачі фотографічних зображень застосовують мікроконтролери з низьким споживанням енергії та обмеженою обчислювальною потужністю. Таким чином, для цих завдань використовують симетричні алгоритми шифрування, які базуються на єдиному ключі для обох процесів – шифрування та дешифрування (або ключ дешифрування обчислюється за ключем шифрування) [1]. Симетричні алгоритми мають численні переваги, такі як: низькі вимоги до обчислювальної потужності, висока пропускна здатність, короткі ключі та гнучкість використання.

Незважаючи на ці переваги, симетричні алгоритми мають свої недоліки, такі як: складність збереження конфіденційності ключа, велика кількість ключів у розгалуженій мережі та необхідність частої або дистанційної зміни ключів. Проте, у випадку систем цифрового зв'язку, які використовуються в комп'ютерних системах, ці недоліки не є критичними, оскільки час злому таких алгоритмів залишається значною величиною (наприклад, для AES-128 цей час становить 40 років).

Моделюючи системи цифрового зв'язку для передачі фотографічних зображень, можна провести порівняння їх ефективності передачі радіоканалом при використанні різних методів шифрування та без них. Це дозволяє проектувати портативні комп'ютерні системи з високим рівнем захисту інформації використовуючи мінімальні обчислювальні потужності.

Література

1. Whitfield Diffie and Martin Hellman, «Multi-user cryptographic techniques» [Diffie and Hellman, AFIPS Proceedings 45,1976].

Додаток Б

Програма для моделювання каналу зв’язку з OFDM та 16-QAM

```
% OFDM Code
% No.of Carriers: 64
% coding used: Convolutional coding
% Single frame size: 96 bits
% Total no. of Frames: 100
% Modulation: 16-QAM
% No. of Pilots: 4
% Cylic Extension: 25%(16)

close all
clear all
clc

%%
% Generating and coding data
t_data=randint(9600,1)';
x=1;
si=1; %for BER rows
%%
for d=1:100;
data=t_data(x:x+95);
x=x+96;
k=3;
n=6;
s1=size(data,2);  % Size of input matrix
j=s1/k;

%%
% Convolutionally encoding data
constlen=7;
codegen = [171 133];    % Polynomial
trellis = poly2trellis(constlen, codegen);
codedata = convenc(data, trellis);

%%
%Interleaving coded data

s2=size(codedata,2);
j=s2/4;
matrix=reshape(codedata,j,4);

intlvddata = matintrlv(matrix',2,2)'; % Interleave.
intlvddata=intlvddata';

% Binary to decimal conversion

dec=bi2de(intlvddata','left-msb');

%%
%16-QAM Modulation
```

```
M=16;
y = qammod(dec,M);
% scatterplot(y);
% Pilot insertion

lendata=length(y);
pilt=3+3j;
nofpits=4;

k=1;

for i=(1:13:52)

    pilt_data1(i)=pilt;

    for j=(i+1:i+12);
        pilt_data1(j)=y(k);
        k=k+1;
    end
end

pilt_data1=pilt_data1';   % size of pilt_data =52
pilt_data(1:52)=pilt_data1(1:52);    % upsizing to 64
pilt_data(13:64)=pilt_data1(1:52);   % upsizing to 64

for i=1:52

    pilt_data(i+6)=pilt_data1(i);

end

% IFFT

ifft_sig=ifft(pilt_data',64);

% Adding Cyclic Extension

cext_data=zeros(80,1);
cext_data(1:16)=ifft_sig(49:64);
for i=1:64

    cext_data(i+16)=ifft_sig(i);

end

% Channel
 % SNR

 o=1;
for snr=0:2:50

ofdm_sig=awgn(cext_data,snr,'measured'); % Adding white Gaussian Noise
% figure;
% index=1:80;
% plot(index,cext_data,'b',index,ofdm_sig,'r'); %plot both signals
% legend('Original Signal to be Transmitted','Signal with AWGN');
```

```
%                    RECEIVER
%Removing Cyclic Extension

for i=1:64

    rxed_sig(i)=ofdm_sig(i+16);

end

% FFT

ff_sig=fft(rxed_sig,64);

% Pilot Synch%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

for i=1:52

    synched_sig1(i)=ff_sig(i+6);

end

k=1;

for i=(1:13:52)

    for j=(i+1:i+12);
        synched_sig(k)=synched_sig1(j);
        k=k+1;
    end
end

% scatterplot(synched_sig)
% Demodulation
dem_data= qamdemod(synched_sig,16);

% Decimal to binary conversion

bin=de2bi(dem_data','left-msb');
bin=bin';

% De-Interleaving

deintlvddata = matdeintrlv(bin,2,2); % De-Interleave
deintlvddata=deintlvddata';
deintlvddata=deintlvddata(:)';

%Decoding data
n=6;
k=3;
decodedata =vitdec(deintlvddata,trellis,5,'trunc','hard');  % decoding
datausing veterbi decoder
rxed_data=decodedata;

% Calculating BER
rxed_data=rxed_data(:)';
errors=0;
```

```
c=xor(data,rxed_data);
errors=nnz(c);

% for i=1:length(data)
%
%     if rxed_data(i)~=data(i);
%         errors=errors+1;
%
%     end
% end

BER(si,o)=errors/length(data);
o=o+1;

 end % SNR loop ends here
 si=si+1;
end % main data loop

% Time averaging for optimum results

for col=1:25;        %%%change if SNR loop Changed
    ber(1,col)=0;
for row=1:100;
        ber(1,col)=ber(1,col)+BER(row,col);
    end
end
ber=ber./100;

figure
i=0:2:48;
semilogy(i,ber,i,ber-0.01,'--rs');
title('BER vs SNR');
ylabel('BER');
xlabel('SNR (dB)');
grid on
```

Додаток В

Програма для шифрування методом AES-128

```
function [out] = aesencrypt(s, in)
% AESENCRYPT  Encrypt 16-bytes vector.
% Usage:            out = aesencrypt(s, in)
% s:                AES structure
% in:               input 16-bytes vector (plaintext)
% out:              output 16-bytes vector (ciphertext)

% Stepan Matejka, 2011, matejka[at]feld.cvut.cz
% $Revision: 1.1.0 $  $Date: 2011/10/12 $

if (nargin ~= 2)
    error('Bad number of input arguments.');
end

validateattributes(s, {'struct'}, {});
validateattributes(in, {'numeric'}, {'real', 'vector', '>=', 0, '<',
256});

% copy input to local
% 16 -> 4 x 4
state = reshape(in, 4, 4);

% Initial round
% AddRoundKey keyexp(1:4)
state = bitxor(state, (s.keyexp(1:4, :))');

% Loop over (s.rounds - 1) rounds
for i = 1:(s.rounds - 1)
    % SubBytes - lookup table
    state = s.s_box(state + 1);
    % ShiftRows
    state = shift_rows(state, 0);
    % MixColumns
    state = mix_columns(state, s);
    % AddRoundKey keyexp(i*4 + (1:4))
    state = bitxor(state, (s.keyexp((1:4) + 4*i, :))');
end

% Final round
% SubBytes - lookup table
state = s.s_box(state + 1);
% ShiftRows
state = shift_rows(state, 0);
% AddRoundKey keyexp(4*s.rounds + (1:4))
state = bitxor(state, (s.keyexp(4*s.rounds + (1:4), :))');

% copy local to output
% 4 x 4 -> 16
out = reshape(state, 1, 16);

% ------------------------------------------------------------------
-
function out = mix_columns(in, s)
```

```
% Each column of the state is multiplied with a fixed polynomial mod_pol

% Slow version
% out = zeros(size(in));
% for col = 1:4
%     for row = 1:4
%         % for each element
%         temp = 0;
%         for i = 1:4
%             % Multiplication in a finite field of
%             % row vector of poly_mat and
%             % column vector of the in
%             % finally xor
%             temp = bitxor(temp,...
%                 poly_mult(s.poly_mat(row, i),...
%                 in(i, col),...
%                 s.mod_pol, s.aes_logt,s.aes_ilogt));
%         end
%         % place to out
%         out(row, col) = temp;
%     end
% end

% Faster implementation
% out = zeros(size(in));
% for col = 1:4
%     temp = bitxor(in(3,col),in(4,col));
%     temp = bitxor(temp,
poly_mult(2,in(1,col),s.mod_pol,s.aes_logt,s.aes_ilogt));
%     out(1,col) = bitxor(temp,
poly_mult(3,in(2,col),s.mod_pol,s.aes_logt,s.aes_ilogt));
%     temp = bitxor(in(1,col),in(4,col));
%     temp = bitxor(temp,
poly_mult(2,in(2,col),s.mod_pol,s.aes_logt,s.aes_ilogt));
%     out(2,col) = bitxor(temp,
poly_mult(3,in(3,col),s.mod_pol,s.aes_logt,s.aes_ilogt));
%     temp = bitxor(in(1,col),in(2,col));
%     temp = bitxor(temp,
poly_mult(2,in(3,col),s.mod_pol,s.aes_logt,s.aes_ilogt));
%     out(3,col) = bitxor(temp,
poly_mult(3,in(4,col),s.mod_pol,s.aes_logt,s.aes_ilogt));
%     temp = bitxor(in(2,col),in(3,col));
%     temp = bitxor(temp,
poly_mult(3,in(1,col),s.mod_pol,s.aes_logt,s.aes_ilogt));
%     out(4,col) = bitxor(temp,
poly_mult(2,in(4,col),s.mod_pol,s.aes_logt,s.aes_ilogt));
% end

% Faster faster implementation
% out = zeros(size(in));
% for col = 1:4
%     temp = bitxor(in(3,col),in(4,col));
%     temp = bitxor(temp, s.mix_col2(in(1,col) + 1));
%     out(1,col) = bitxor(temp, s.mix_col3(in(2,col) + 1));
%     temp = bitxor(in(1,col),in(4,col));
%     temp = bitxor(temp, s.mix_col2(in(2,col) + 1));
%     out(2,col) = bitxor(temp, s.mix_col3(in(3,col) + 1));
```

```
%     temp = bitxor(in(1,col),in(2,col));
%     temp = bitxor(temp, s.mix_col2(in(3,col) + 1));
%     out(3,col) = bitxor(temp, s.mix_col3(in(4,col) + 1));
%     temp = bitxor(in(2,col),in(3,col));
%     temp = bitxor(temp, s.mix_col3(in(1,col) + 1));
%     out(4,col) = bitxor(temp, s.mix_col2(in(4,col) + 1));
% end

% Faster faster faster implementation
% slice1 = zeros(4,4);
% slice2 = slice1;
% slice3 = slice1;
% slice4 = slice1;
% for col = 1:4
%     slice1(1,col) = in(3,col);
%     slice2(1,col) = in(4,col);
%     slice3(1,col) = s.mix_col2(in(1,col) + 1);
%     slice4(1,col) = s.mix_col3(in(2,col) + 1);
%     slice1(2,col) = in(1,col);
%     slice2(2,col) = in(4,col);
%     slice3(2,col) = s.mix_col2(in(2,col) + 1);
%     slice4(2,col) = s.mix_col3(in(3,col) + 1);
%     slice1(3,col) = in(1,col);
%     slice2(3,col) = in(2,col);
%     slice3(3,col) = s.mix_col2(in(3,col) + 1);
%     slice4(3,col) = s.mix_col3(in(4,col) + 1);
%     slice1(4,col) = in(2,col);
%     slice2(4,col) = in(3,col);
%     slice3(4,col) = s.mix_col3(in(1,col) + 1);
%     slice4(4,col) = s.mix_col2(in(4,col) + 1);
% end
% out = bitxor(bitxor(bitxor(slice1, slice2), slice3), slice4);

% Faster faster faster faster implementation
out = bitxor(bitxor(bitxor([in(3,1:4); in(1,1:4); in(1,1:4);
in(2,1:4)],...
    [in(4,1:4); in(4,1:4); in(2,1:4); in(3,1:4)]),...
    [s.mix_col2(in(1,1:4) + 1); s.mix_col2(in(2,1:4) + 1);
s.mix_col2(in(3,1:4) + 1); s.mix_col3(in(1,1:4) + 1)]),...
    [s.mix_col3(in(2,1:4) + 1); s.mix_col3(in(3,1:4) + 1);
s.mix_col3(in(4,1:4) + 1); s.mix_col2(in(4,1:4) + 1)]);

% ------------------------------------------------------------------------
-
function p = poly_mult(a, b, mod_pol, aes_logt, aes_ilogt)
% Multiplication in a finite field

% Old slow implementation
% p = 0;
% for counter = 1:8
%     if (rem(b,2))
%         p = bitxor(p,a);
%         b = (b - 1)/2;
%     else
%         b = b/2;
%     end
%     a = 2*a;
```

```
%     if (a>255)
%         a = bitxor(a,mod_pol);
%     end
% end

% Faster implementaion
if (a && b)
    p = aes_ilogt(mod((aes_logt(a + 1) + aes_logt(b + 1)), 255) + 1);
else
    p = 0;
end

% ------------------------------------------------------------------------
-
function out = shift_rows(in, dir)
% ShiftRows cyclically shift the rows of the 4 x 4 matrix.
%
%   dir = 0 (to left)
%  | 1 2 3 4 |
%  | 2 3 4 1 |
%  | 3 4 1 2 |
%  | 4 1 2 3 |
%
%   dir ~= 0 (to right)
%  | 1 2 3 4 |
%  | 4 1 2 3 |
%  | 3 4 1 2 |
%  | 2 3 4 1 |
%

if (dir == 0)
    % left
    % use linear indexing in 2d array
    out = reshape(in([1 6 11 16 5 10 15 4 9 14 3 8 13 2 7 12]),4,4);
    % old safe method
%     temp = reshape(in,16,1);
%     temp = temp([1 6 11 16 5 10 15 4 9 14 3 8 13 2 7 12]);
%     out = reshape(temp,4,4);
else
    % right
    % use linear indexing in 2d array
    out = reshape(in([1 14 11 8 5 2 15 12 9 6 3 16 13 10 7 4]),4,4);
    % old safe method
%     temp = reshape(in,16,1);
%     temp = temp([1 14 11 8 5 2 15 12 9 6 3 16 13 10 7 4]);
%     out = reshape(temp,4,4);
end

% ------------------------------------------------------------------------
-
% end of file
```

Додаток Г

Програма для дешифрування методом AES-128

```
function [out] = aesdecrypt(s, in)
% AESDECRYPT Decrypt 16-bytes vector.
% Usage:            out = aesdecrypt(s, in)
% s:                AES structure
% in:               input 16-bytes vector (ciphertext)
% out:              output 16-bytes vector (plaintext)

% Stepan Matejka, 2011, matejka[at]feld.cvut.cz
% $Revision: 1.1.0 $  $Date: 2011/10/12 $

if (nargin ~= 2)
    error('Bad number of input arguments.');
end

validateattributes(s, {'struct'}, {});
validateattributes(in, {'numeric'}, {'real', 'vector', '>=', 0, '<',
256});

% copy input to local
% 16 -> 4 x 4
state = reshape(in, 4, 4);

% Initial round
% AddRoundKey keyexp(s.rounds*4 + (1:4))
state = bitxor(state, (s.keyexp(s.rounds*4 + (1:4), :))');

% Loop over (s.rounds - 1) rounds
for i = (s.rounds - 1):-1:1
    % ShiftRows
    state = shift_rows(state, 1);
    % SubBytes - lookup table
    state = s.inv_s_box(state + 1);
    % AddRoundKey keyexp(i*4 + (1:4))
    state = bitxor(state, (s.keyexp((1:4) + 4*i, :))');
    % MixColumns
    state = mix_columns(state, s);
end

% Final round
% ShiftRows
state = shift_rows(state, 1);
% SubBytes - lookup table
state = s.inv_s_box(state + 1);
% AddRoundKey keyexp(1:4)
state = bitxor(state, (s.keyexp(1:4, :))');

% copy local to output
% 4 x 4 -> 16
out = reshape(state, 1, 16);
% ------------------------------------------------------------------
-
function out = mix_columns(in, s)
% Each column of the state is multiplied with a fixed polynomial mod_pol
```

```
% Slow version
% out = zeros(size(in));
% for col = 1:4
%     for row = 1:4
%         % for each element
%         temp = 0;
%         for i = 1:4
%             % Multiplication in a finite field of
%             % row vector of poly_mat and
%             % column vector of the in
%             % finally xor
%             temp = bitxor(temp,...
%                 poly_mult(s.inv_poly_mat(row, i),...
%                 in(i, col),...
%                 s.mod_pol, s.aes_logt,s.aes_ilogt));
%         end
%         % place to out
%         out(row, col) = temp;
%     end
% end

% Faster implementation
% out = zeros(size(in));
% for col = 1:4
%     temp = poly_mult(14,in(1,col),s.mod_pol,s.aes_logt,s.aes_ilogt);
%     temp = bitxor(temp,
poly_mult(11,in(2,col),s.mod_pol,s.aes_logt,s.aes_ilogt));
%     temp = bitxor(temp,
poly_mult(13,in(3,col),s.mod_pol,s.aes_logt,s.aes_ilogt));
%     out(1,col) = bitxor(temp,
poly_mult(9,in(4,col),s.mod_pol,s.aes_logt,s.aes_ilogt));
%     temp = poly_mult(9,in(1,col),s.mod_pol,s.aes_logt,s.aes_ilogt);
%     temp = bitxor(temp,
poly_mult(14,in(2,col),s.mod_pol,s.aes_logt,s.aes_ilogt));
%     temp = bitxor(temp,
poly_mult(11,in(3,col),s.mod_pol,s.aes_logt,s.aes_ilogt));
%     out(2,col) = bitxor(temp,
poly_mult(13,in(4,col),s.mod_pol,s.aes_logt,s.aes_ilogt));
%     temp = poly_mult(13,in(1,col),s.mod_pol,s.aes_logt,s.aes_ilogt);
%     temp = bitxor(temp,
poly_mult(9,in(2,col),s.mod_pol,s.aes_logt,s.aes_ilogt));
%     temp = bitxor(temp,
poly_mult(14,in(3,col),s.mod_pol,s.aes_logt,s.aes_ilogt));
%     out(3,col) = bitxor(temp,
poly_mult(11,in(4,col),s.mod_pol,s.aes_logt,s.aes_ilogt));
%     temp = poly_mult(11,in(1,col),s.mod_pol,s.aes_logt,s.aes_ilogt);
%     temp = bitxor(temp,
poly_mult(13,in(2,col),s.mod_pol,s.aes_logt,s.aes_ilogt));
%     temp = bitxor(temp,
poly_mult(9,in(3,col),s.mod_pol,s.aes_logt,s.aes_ilogt));
%     out(4,col) = bitxor(temp,
poly_mult(14,in(4,col),s.mod_pol,s.aes_logt,s.aes_ilogt));
% end

% Faster faster implementation
% out = zeros(size(in));
% for col = 1:4
```

```
%     temp = s.mix_col14(in(1,col) + 1);
%     temp = bitxor(temp, s.mix_col11(in(2,col) + 1));
%     temp = bitxor(temp, s.mix_col13(in(3,col) + 1));
%     out(1,col) = bitxor(temp, s.mix_col9(in(4,col) + 1));
%     temp = s.mix_col9(in(1,col) + 1);
%     temp = bitxor(temp, s.mix_col14(in(2,col) + 1));
%     temp = bitxor(temp, s.mix_col11(in(3,col) + 1));
%     out(2,col) = bitxor(temp, s.mix_col13(in(4,col) + 1));
%     temp = s.mix_col13(in(1,col) + 1);
%     temp = bitxor(temp, s.mix_col9(in(2,col) + 1));
%     temp = bitxor(temp, s.mix_col14(in(3,col) + 1));
%     out(3,col) = bitxor(temp, s.mix_col11(in(4,col) + 1));
%     temp = s.mix_col11(in(1,col) + 1);
%     temp = bitxor(temp, s.mix_col13(in(2,col) + 1));
%     temp = bitxor(temp, s.mix_col9(in(3,col) + 1));
%     out(4,col) = bitxor(temp, s.mix_col14(in(4,col) + 1));
% end

% Faster faster faster implementation
% slice1 = zeros(4,4);
% slice2 = slice1;
% slice3 = slice1;
% slice4 = slice1;
% for col = 1:4
%     slice1(1,col) = s.mix_col14(in(1,col) + 1);
%     slice2(1,col) = s.mix_col11(in(2,col) + 1);
%     slice3(1,col) = s.mix_col13(in(3,col) + 1);
%     slice4(1,col) = s.mix_col9(in(4,col) + 1);
%     slice1(2,col) = s.mix_col9(in(1,col) + 1);
%     slice2(2,col) = s.mix_col14(in(2,col) + 1);
%     slice3(2,col) = s.mix_col11(in(3,col) + 1);
%     slice4(2,col) = s.mix_col13(in(4,col) + 1);
%     slice1(3,col) = s.mix_col13(in(1,col) + 1);
%     slice2(3,col) = s.mix_col9(in(2,col) + 1);
%     slice3(3,col) = s.mix_col14(in(3,col) + 1);
%     slice4(3,col) = s.mix_col11(in(4,col) + 1);
%     slice1(4,col) = s.mix_col11(in(1,col) + 1);
%     slice2(4,col) = s.mix_col13(in(2,col) + 1);
%     slice3(4,col) = s.mix_col9(in(3,col) + 1);
%     slice4(4,col) = s.mix_col14(in(4,col) + 1);
% end
% out = bitxor(bitxor(bitxor(slice1, slice2), slice3), slice4);

% Faster faster faster faster implementation
out = bitxor(bitxor(bitxor(...
    [s.mix_col14(in(1,1:4) + 1); s.mix_col9(in(1,1:4) + 1);
s.mix_col13(in(1,1:4) + 1); s.mix_col11(in(1,1:4) + 1)],...
    [s.mix_col11(in(2,1:4) + 1); s.mix_col14(in(2,1:4) + 1);
s.mix_col9(in(2,1:4) + 1);  s.mix_col13(in(2,1:4) + 1)]),...
    [s.mix_col13(in(3,1:4) + 1); s.mix_col11(in(3,1:4) + 1);
s.mix_col14(in(3,1:4) + 1); s.mix_col9(in(3,1:4) + 1)]),...
    [s.mix_col9(in(4,1:4) + 1);  s.mix_col13(in(4,1:4) + 1);
s.mix_col11(in(4,1:4) + 1); s.mix_col14(in(4,1:4) + 1)]);
% ------------------------------------------------------------------------
-
function p = poly_mult(a, b, mod_pol, aes_logt, aes_ilogt)
% Multiplication in a finite field
```

```
% Old slow implementation
% p = 0;
% for counter = 1:8
%     if (rem(b,2))
%         p = bitxor(p,a);
%         b = (b - 1)/2;
%     else
%         b = b/2;
%     end
%     a = 2*a;
%     if (a>255)
%         a = bitxor(a,mod_pol);
%     end
% end

% Faster implementaion
if (a && b)
    p = aes_ilogt(mod((aes_logt(a + 1) + aes_logt(b + 1)), 255) + 1);
else
    p = 0;
end
% ------------------------------------------------------------------------
-
function out = shift_rows(in, dir)
% ShiftRows cyclically shift the rows of the 4 x 4 matrix.
%   dir = 0 (to left)
%  | 1 2 3 4 |
%  | 2 3 4 1 |
%  | 3 4 1 2 |
%  | 4 1 2 3 |
%   dir ~= 0 (to right)
%  | 1 2 3 4 |
%  | 4 1 2 3 |
%  | 3 4 1 2 |
%  | 2 3 4 1 |

if (dir == 0)
    % left
    % use linear indexing in 2d array
    out = reshape(in([1 6 11 16 5 10 15 4 9 14 3 8 13 2 7 12]),4,4);
    % old safe method
%     temp = reshape(in,16,1);
%     temp = temp([1 6 11 16 5 10 15 4 9 14 3 8 13 2 7 12]);
%     out = reshape(temp,4,4);
else
    % right
    % use linear indexing in 2d array
    out = reshape(in([1 14 11 8 5 2 15 12 9 6 3 16 13 10 7 4]),4,4);
    % old safe method
%     temp = reshape(in,16,1);
%     temp = temp([1 14 11 8 5 2 15 12 9 6 3 16 13 10 7 4]);
%     out = reshape(temp,4,4);
end
% ------------------------------------------------------------------------
% end of file
```