Форма № 24

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Національний університет «Запорізька політехніка»

Факультет інформаційної безпеки та електронних комунікацій

(повне найменування факультету)

Кафедра радіотехніки та телекомунікацій

(повне найменування кафедри)

Пояснювальна записка

до дипломного проєкту (роботи)

бакалавра

(ступінь вищої освіти)

на тему **СТЕГАНОГРАФІЧНИЙ МЕТОД ПЕРЕДАЧІ ІНФОРМАЦІЇ З ВИКОРИСТАННЯМ ТЕХНОЛОГІЇ BLUETOOTH**

(назва теми)

Виконав(ла): студент(ка) 4 курсу, групи БК-911

Спеціальності 172

(код і найменування спеціальності)

«Телекомунікації та радіотехніка

Освітня програма (спеціалізація)

Інформаційні мережі зв'язку

ФІЛІПОВИЧ Є.В.

(ПРІЗВИЩЕ та ініціали)

Керівник КОСТЕНКО В.О.

(ПРІЗВИЩЕ та ініціали)

Рецензент КАСЬЯН М.М.

(ПРІЗВИЩЕ та ініціали)

2025 рік

Форма № 25

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»

Факультет *Інформаційної безпеки та електронних комунікацій*
Кафедра *Радіотехніки та телекомунікацій*
Ступінь вищої освіти *бакалавр*
Спеціальність *172 «Телекомунікації та радіотехніка»*
(код і найменування)
Освітня програма (спеціалізація) *«Інформаційні мережі зв'язку»*
(назва освітньої програми (спеціалізації))

ЗАТВЕРДЖУЮ
В.О. завідувача кафедри *РТТ*
Сергій САМОЙЛИК
«_____» червня 2025 року

З А В Д А Н Н Я
НА ДИПЛОМНИЙ ПРОЄКТ (РОБОТУ) СТУДЕНТА(КИ)

ФІЛІПОВИЧУ Євгенію Валерійовичу
(ПРІЗВИЩЕ, ім'я, по батькові)

1. Тема проєкту (роботи) *Стеганографічний метод передачі інформації з використанням технології Bluetooth*
керівник проєкту (роботи) *к.т.н., доцент, КОСТЕНКО Валер'ян Остапович*
(науковий ступінь, вчене звання, ПРІЗВИЩЕ, ім'я, по батькові)
затверджені наказом закладу вищої освіти від «*17*» *квітня* 2025 року №*189*
2. Строк подання студентом проєкту (роботи)_____
3. Вихідні дані до проєкту (роботи) *Принципи побудови бездротових мереж. Теоретичні та практичні аспекти стеганографії (метод найменш значущого біта – LSB). Можливості використання середовища Matlab для моделювання. Технічна специфікація Bluetooth-протоколу. Дані про типові спотворення в нестабільних бездротових каналах (для оцінки BER).*
4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) *Аналіз принципів побудови самоорганізованих бездротових мереж. Огляд стеганографічних методів шифрування інформації. Дослідження можливостей технології Bluetooth для прихованої передачі даних. Моделювання стеганографічної передачі інформації в Matlab. Опис проведеного моделювання, включно з аналізом результатів.*
5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень, кількість слайдів, плакатів) *презентація роботи в Microsoft PowerPoint*

6. Консультанти розділів проєкту (роботи)

Розділ	ПРІЗВИЩЕ, ініціали та посада консультанта	Підпис, дата	
		завдання видав	прийняв виконане завдання
1-4	*КОСТЕНКО В.О., доцент кафедри РТТ*		
нормо-	*МОРОЗ Г.В., ст. викладач кафедри РТТ*		
контроль			

7. Дата видачі завдання *«17» квітня 2025* року

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проєкту (роботи)	Строк виконання етапів проєкту (роботи)	Примітка
1	*Принципи побудови безпровідних мереж, що самоорганізуються*	*17.02.2025-17.03.2025*	*виконано*
2	*Стеганографічні методи шифрування інформації.*	*18.03.2025-31.03.2025*	*виконано*
3	*Можливості технології Bluetooth для прихованої передачі інформації.*	*01.04.2025-17.04.2025*	*виконано*
4	*Моделювання передачі фрагмента інформації стеганографічним способом у програмному середовищі Matlab*	*01.04.2025-17.04.2025*	*виконано*
5	*Опис процесу проведеного моделювання*	*01.05.2025-04.05.2025*	*виконано*
6	*Проходження нормоконтролю, рецензування, антиплагіату.*	*05.05.2025-29.06.2025*	*виконано*

Студент(ка) _____ *Євгеній ФІЛІПОВИЧ*
(підпис) (Ім'я ПРІЗВИЩЕ)

Керівник проєкту (роботи) _____ *Валер'ян КОСТЕНКО*
(підпис) (Ім'я ПРІЗВИЩЕ)

РЕФЕРАТ

Пояснювальна записка до бакалаврської роботи: 91 с., 3 табл., 18 рис., 4 дод., 29 джерел.

СТЕГАНОГРАФІЯ, НАЙМЕНШ ЗНАЧУЩИЙ БІТ, ПЕРЕДАЧА ІНФОРМАЦІЇ, BLUETOOTH, ПРИХОВАНІ ДАНІ, БІТОВІ ПОМИЛКИ, БЕЗДРОТОВІ МЕРЕЖІ, МОДЕЛЮВАННЯ.

Мета роботи – розробка та дослідження моделі прихованої передачі інформації за допомогою стеганографічного методу у середовищі Bluetooth, а також оцінка його стійкості до спотворень та визначення ефективності відновлення прихованих даних після проходження через нестабільний бездротовий канал зв’язку.

Об’єкт дослідження – процес прихованої передачі інформації в бездротових Bluetooth-мережах.

Предмет дослідження – стеганографічний метод приховання та передачі інформації в Bluetooth-пакетах із використанням найменш значущого бітаLSB-модифікацій.

Методи дослідження: математичного моделювання (для опису процесу приховання та передачі даних), симуляції у середовищі Matlab, статистичний аналіз результатів експериментів (для визначення рівня бітових помилок (BER) при різних умовах передачі), а також програмна реалізація алгоритмів кодування та декодування повідомлень.

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ

LSB – (Least Significant Bit) найменш значущий біт

BER – (Bit Error Rate) коефіцієнт бітових помилок

SON – (Self-Organizing Network) самоорганізована безпровідна мережа

UUID – (Universally Unique Identifier) універсальний унікальний ідентифікатор

BLE – (Bluetooth Low Energy) енергозберігаючий Bluetooth

SDP – (Bluetooth Low Energy) енергозберігаючий Bluetooth

L2CAP – (Logical Link Control and Adaptation Protocol) логічне керування каналами та адаптація

RF – (Radio Frequency) радіочастота

IoT – (Internet of Things) Інтернет речей

КПК – кишеньковий персональний комп’ютер

Hz – герц, одиниця частоти (1 Гц = 1 коливання в секунду)

GHz – гігагерц (1 ГГц = 10^9 Гц)

Mb/s – мегабіт в секунду (швидкість передачі даних)

dB – децибел (одиниця вимірювання рівня сигналу)

p – ймовірність спотворень в каналі передачі

$\oplus$ – оператор виключного "АБО" (XOR), використовується для визначення відмінностей між бітами

ЗМІСТ

ВСТУП

Сучасний світ стрімко розвивається, що спричинено активним впровадженням цифрових технологій у всі сфери людської діяльності. У зв’язку зі зростанням використання цифрових інструментів, гостро постає питання захисту інформації, що передається, особливо коли йдеться про бездротові з’єднання. Останніми роками широкого впровадження як у побуті, так і в промисловості набув Bluetooth – один із найпоширеніших способів передавання даних між пристроями. Однак передані дані можуть бути перехоплені, змінені або загублені через зовнішні впливи, зменшуючи таким чином стабільність і безпечність даного каналу зв’язку.

Одним із перспективних методів захисту даних є стеганографія, яка дозволяє приховувати дані усередині інших файлів так, щоб ніхто не запідозрив, що щось передається. На відміну від шифрування, яке лише робить дані незрозумілими, стеганографія маскує сам факт наявності інформації, що передається. Одним із найпопулярніших способів такого приховування є метод найменш значущого біта (LSB), коли окремі біти в даних змінюються так, що загальний вигляд файлу майже не змінюється. Якщо цей метод використати в Bluetooth-з’єднаннях, можна створити прихований канал для обміну інформацією навіть у нестабільних умовах.

Актуальність дослідження зумовлена зростаючою кількістю загроз у сфері кібербезпеки, і перехопленням бездротових даних, а також необхідністю не тільки захистити дані, а й зробити сам факт передачі непомітним. До того ж стеганографія дає додатковий рівень безпеки на відміну від традиційних способів шифрування, які можуть бути виявлені та зламані. Технологія Bluetooth стає цікавим каналом для прихованої передачі даних, а метод LSB

можна застосовувати до тексту, зображень та інших мультимедійних файлів, що робить його ще кориснішим на практиці.

Отже, вивчення можливостей стеганографії для передачі інформації через Bluetooth є важливим кроком у напрямку підвищення безпеки даних у сучасних цифрових мережах.

Метою роботи є розробка та дослідження моделі прихованої передачі інформації за допомогою стеганографічного методу у середовищі Bluetooth, а також оцінка його стійкості до спотворень та визначення ефективності відновлення прихованих даних після проходження через нестабільний бездротовий канал зв’язку.

Об’єкт дослідження – процес прихованої передачі інформації в бездротових Bluetooth-мережах.

Предмет дослідження – стеганографічний метод приховання та передачі інформації в Bluetooth-пакетах із використанням найменш значущого біта LSB-модифікацій.

Для досягнення поставленої мети необхідно вирішити такі завдання:

а) проаналізувати особливості стеганографічних методів та їх застосування у бездротових мережах;

б) розробити алгоритм приховання та вилучення інформації методом LSB у Bluetooth-середовищі;

в) провести моделювання процесу передачі даних із використанням Matlab;

г) дослідити вплив спотворень Bluetooth-каналу на точність відновлення прихованих даних;

д) провести тестування моделі з використанням різних типів носіїв (пакетів, зображень) та оцінити ефективність методу за допомогою коефіцієнта бітових помилок (BER).

Методи дослідження. Для досягнення мети використовувалися методи математичного моделювання (для опису процесу приховання та передачі даних), симуляції у середовищі Matlab, статистичний аналіз результатів експериментів (для визначення рівня бітових помилок (BER) при різних умовах передачі), а також програмна реалізація алгоритмів кодування та декодування повідомлень.

Теоретичне значення отриманих результатів полягає в поглибленні уявлень про можливості стеганографії у бездротових каналах зв’язку, зокрема в умовах завадостійкого середовища, а також у розробці математичної моделі процесу вбудовування та вилучення інформації.

Практичне значення роботи полягає у можливості впровадження запропонованої стеганографічної моделі у мобільні додатки, IoT-пристрої або системи передачі конфіденційної інформації, що потребують підвищеного рівня захисту без використання складних криптографічних механізмів.

Отримані результати можуть бути використані для підвищення рівня безпеки даних у бездротових мережах, а також у розробці програмного забезпечення для стеганографічного приховання інформації.

Структура та обсяг роботи. Робота складається зі вступу, чотирьох розділів, висновків, списку літератури у кількості 30 найменувань та 4 додатків, загальний обсяг роботи становить 90 сторінок, містить 18 рисунків, 3 таблиці.

1 ПРИНЦИПИ ПОБУДОВИ БЕЗПРОВІДНИХ МЕРЕЖ, ЩО САМООРГАНІЗУЮТЬСЯ

У сучасних умовах розвитку інформаційних технологій зростає потреба у швидкому розгортанні надійних і гнучких систем зв’язку, особливо у сферах, де традиційна інфраструктура або відсутня, або є складною для підтримки. Одним із перспективних рішень у цій галузі є безпровідні мережі, що самоорганізуються. Такі мережі здатні автоматично налаштовуватись, підтримувати з’єднання між вузлами без централізованого керування, а також адаптуватися до змін у топології чи навколишньому середовищі [1].

1.1 Поняття та загальна характеристика самоорганізованих мереж

Самоорганізована безпровідна мережа (англ. Self-Organizing Wireless Network (SON) – це тип мережі, де окремі пристрої (вузли) не потребують попередньої конфігурації або централізованого керування [2]. На відміну від традиційних мереж, у яких керування й конфігурація здійснюються централізовано, у SON усі функції – від налаштування до оптимізації та відновлення – виконуються автоматично на рівні окремих пристроїв. Основна ідея полягає в тому, що кожен вузол (пристрій, модуль, сенсор) виконує не лише роль кінцевого користувача, а й функції маршрутизатора. Він може приймати, обробляти та пересилати дані іншим вузлам. Таким чином формується розподілена топологія, де немає єдиного центру або точки відмови, що суттєво підвищує надійність та гнучкість системи.

Сучасні самоорганізовувані мережі беруть свій початок з 1970-х років, коли були створені PRNET (Packet Radio Networks) за фінансування

Міністерства оборони США. Основна мета таких мереж – забезпечити можливість підключення до Інтернету в будь-якому місці, навіть у русі, без залежності від стаціонарної інфраструктури.

З розвитком технологій та поширенням бездротового зв'язку виникла потреба у мережах нового типу, які не мають сталої структури та здатні адаптуватися до змін умов передачі даних. Такі мережі отримали назву самоорганізованих. Перші комерційні мобільні самоорганізовані мережі почали функціонувати у США та Японії в період 2009-2010 років.

Залежно від рівня автономності та ступеня участі користувачів, самоорганізовувані мережі поділяють на дві основні категорії: adhoc та mesh-мережі. У adhoc мережі (від лат. adhoc – «для цього») всі вузли рівноправні, кожен може виконувати функції передачі, маршрутизації та обробки інформації [3]. Більш складні структури у деяких випадках будуються за допомогою mesh-топології, де кожен вузол може мати кілька з'єднань з іншими, утворюючи сітчасту структуру та дозволяючи зменшити затримки, підвищити пропускну здатність та стійкість до відмов [4]. Головна відмінність між цими видами мереж полягає в тому, що adhoc здебільшого використовується для термінальних з'єднань, а mesh-мережі – для транзитних, хоча це розмежування є умовним і прийнятим на даний момент.

Крім двох зазначених можна ще виділити наступні типи безпровідних самоорганізованих мереж:

– MANET (Mobile Adhoc Networks) – мобільні adhoc-мережі, де вузли постійно змінюють своє положення;

– WSN (Wireless Sensor Networks) – мережі безпровідних сенсорів, що збирають та передають дані про навколишнє середовище;

– Vehicular Adhoc Networks (VANET) – мережі між транспортними засобами, що використовуються в інтелектуальних транспортних системах.

Однак всі перераховані мережі мають спільні характеристики: спонтанність і динамічність. Перша характеристика описує можливість мережі швидко розгортатися «з нуля» у будь-якому місці, де є пристрої, здатні до самоорганізації. Дана особливість актуальна в умовах, де недоступна або пошкоджена звичайна інфраструктура (під час природних катастроф, у військових умовах, на віддалених територіях тощо). Завдяки динамічності мережа здатна змінювати свою топологію в режимі реального часу, реагуючи на появу нових вузлів, втрату зв’язку, зміну умов передачі сигналу. Завдяки цьому система здатна самовідновлюватися – при зникненні одного вузла, маршрути автоматично перебудовуються так, щоб зберегти зв’язок між іншими вузлами.

На рисунку 1.1 представлено загальну структуру самоорганізованої безпровідної мережі з урахуванням її основних принципів, типів, характеристик та сфер застосування.

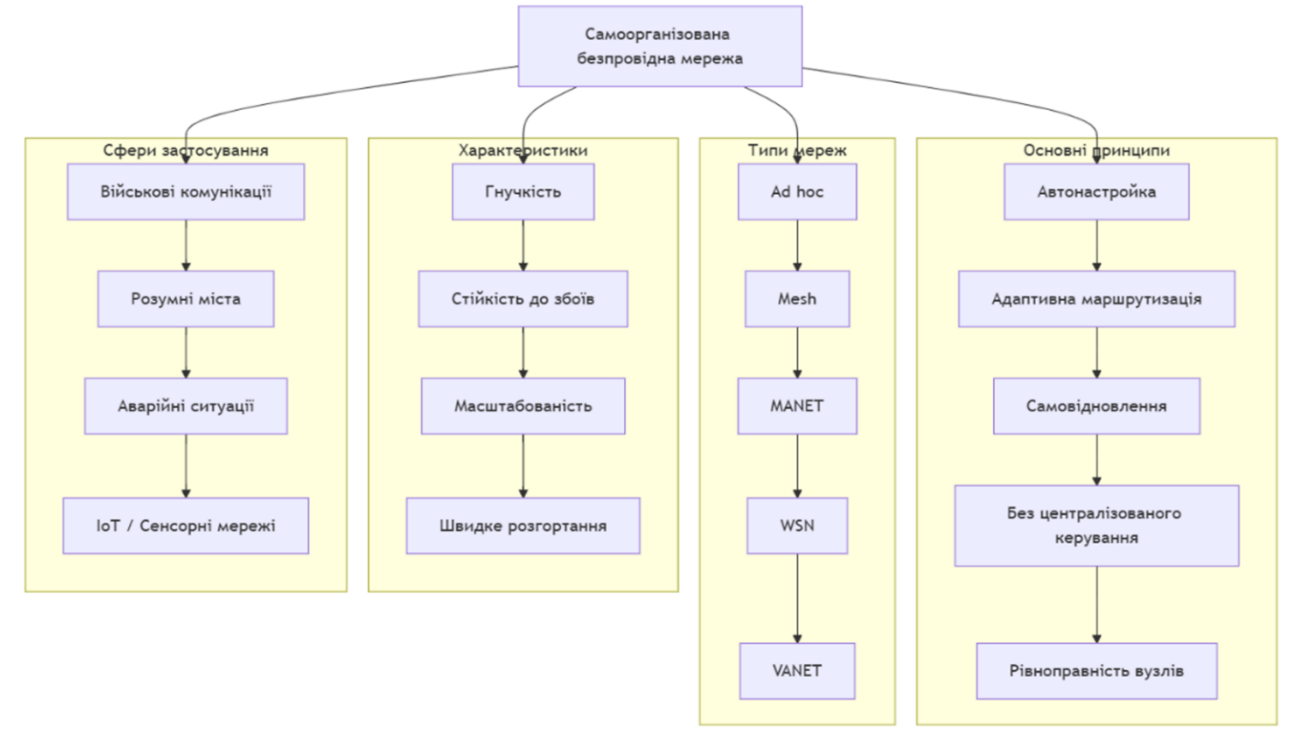

Рисунок 1.1 – Структурна схема ключових характеристик та компонентів самоорганізованої безпровідної мережі

Серед основних характеристик самоорганізованих мереж можна виділити:

- автоматичне налаштування – мережа не потребує ручної конфігурації при додаванні нових вузлів;
- оптимізація ресурсів – адаптація параметрів зв'язку для забезпечення найкращої якості обслуговування;
- виявлення та усунення несправностей – здатність мережі самостійно виявляти і компенсувати збої;
- розподілений характер керування – відсутність єдиного контролюючого вузла, що знижує ймовірність критичних відмов.

Таким чином, безпровідні самоорганізовані мережі є гнучкими, надійними й адаптивними комунікаційними системами. Широке їх застосування (у мобільних, військових, аварійних, сенсорних та IoT-мережах) обумовлене здатністю працювати в складних умовах без централізованої підтримки.

1.2 Основні принципи побудови самоорганізованих безпровідних мереж

Побудова безпровідної мережі, що самоорганізується, ґрунтується на таких ключових принципах [5, 6, 7] (рис. 1.2):

а) автоматична конфігурація (Self-Configuration) – кожен вузол самостійно приймає рішення про з'єднання з іншими вузлами, базуючись на локальній інформації. Така функціональність особливо важлива в умовах динамічної топології або при масовому розгортанні пристроїв Інтернету речей (IoT) [8];

б) самоорганізація (Self-Organization) – мережа змінює свою структуру відповідно до зміни кількості вузлів, їх розміщення або умов навколишнього

середовища (наприклад, зникнення зв’язку, перешкоди), що забезпечує адаптивність до зміни кількості вузлів або їх переміщення в просторі [9];

в) самовідновлення (Self-Healing) – у разі виходу з ладу одного чи кількох вузлів система самостійно перебудовує маршрути передачі, зберігаючи функціональність, таким чином забезпечуючи високу живучість мережі в умовах відмов або ворожого втручання [10];

г) самооптимізація (Self-Optimization) – SON постійно аналізує параметри якості обслуговування: потужність сигналу, навантаження на вузли, затримки та інші метрики. Відповідно до результатів аналізу, мережа автоматично оптимізує роботу – наприклад, перемикається на найменш завантажений канал або змінює потужність передавача [11];

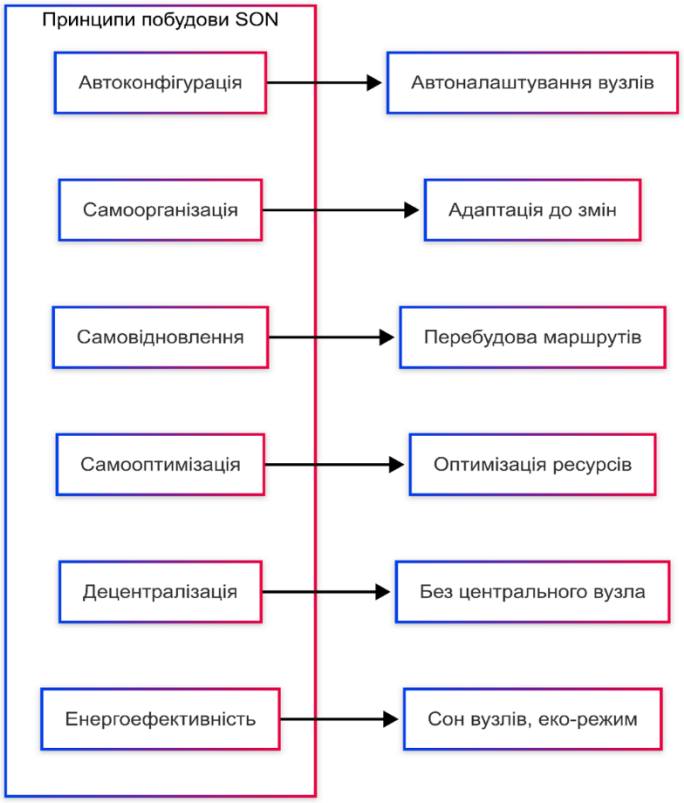

Рисунок 1.2 – Основні принципи побудови самоорганізованих безпровідних мереж

а) децентралізоване управління – у SON логіка розподілена між вузлами. Така архітектура забезпечує стійкість до точок відмови та знижує затрати на підтримку мережі [8];

б) енергоефективність – вузли працюють в енергозберігаючому режимі, переходячи в стан сну при простої, а також оптимізують маршрути для мінімізації енергоспоживання. Це критично важливо для сенсорних мереж, що живляться від батарей [9].

На рисунку 1.3 проілюстровано основні принципи побудови самоорганізованих безпровідних мереж (SON) шляхом порівняння їх із традиційними мережами: традиційні мережі характеризуються централізованим керуванням, ручною конфігурацією, статичною топологією та високою залежністю від центрального вузла.

1.3 Безпека та функціональність самоорганізованих безпровідних мереж

Самоорганізовані мережі забезпечують автоматичне налаштування, адаптацію до змін у середовищі та ефективне управління ресурсами без необхідності централізованого контролю, однак попри численні переваги, SON мають специфічні загрози, які необхідно враховувати при їх розгортанні та експлуатації, а саме:

а) SON-мережі можуть бути вразливими до атак (перехоплення даних, підміна вузлів, маніпуляція маршрутами) через відсутність централізованого контролю;

б) для запобігання несанкціонованому доступу важливим є автентифікація вузлів та використання шифрування даних;

в) SON-мережі можуть використовувати інтелектуальні алгоритми аналізу трафіку для виявлення загроз (аномалій та потенційних атак);

г) для запобігання внутрішніх загроз необхідно впроваджувати механізми довіри та контролю, оскільки вузли SON-мереж можуть належати різним адміністративним доменам.

Рисунок 1.3 – Порівняння традиційних та самоорганізованих безпровідних мереж

SON-мережі, дозволяючи створювати гнучкі, надійні та ефективні системи зв’язку, проникають у різні сфери людської діяльності:

– мобільні мережі та телекомунікації (оператори мобільного зв’язку використовують SON для автоматичного налаштування базових станцій, оптимізації покриття та управління навантаженням, що дозволяє зменшити витрати на обслуговування та підвищити якість зв’язку) [12];

– сенсорні мережі для моніторингу (безпровідні сенсорні мережі використовуються для контролю рівня забруднення повітря, вологості ґрунту та стану обладнання, тощо) [13];

– інтернет речей (IoT) та розумні міста (наприклад, інтелектуальні системи управління дорожнім рухом можуть адаптуватися до змін у реальному часі, покращуючи ефективність міської інфраструктури) [14];

– військові та рятувальні операції (використовуються у військових системах зв’язку та рятувальних операціях, де необхідно швидко розгортати автономні мережі без централізованого управління) [15];

– автономні транспортні системи (застосовуються у безпілотних автомобілях та дронах, забезпечуючи координацію між пристроями без необхідності централізованого контролю).

Висновки до першого розділу

У першому розділі висвітлено основні моменти організації безпровідних мереж, що функціонують за принципами самоорганізації, а також подано їхню типологію. Самоорганізовані мережі (SON) характеризуються децентралізованим підходом до передачі інформації між окремими вузлами, що дозволяє ефективно використовувати їх у середовищах із нестабільною або відсутньою традиційною інфраструктурою. Завдяки здатності до автономного

налаштування, підтримки зв’язку та оптимізації роботи, ці мережі демонструють високий рівень надійності, гнучкості й енергоефективності.

Залежно від специфіки застосування, SON поділяються на кілька типів: adhoc, mesh, MANET, WSN, VANET. Вони охоплюють широке коло галузей — від мобільних і сенсорних систем до військових комунікацій і Інтернету речей. Основними перевагами SON є стійкість до відмов, адаптивність до змін середовища, а також раціональне використання доступних ресурсів. Проте, через відсутність централізованого керування зростає ризик кіберзагроз, тому особливого значення набувають засоби автентифікації користувачів та аналізу мережевого трафіку для забезпечення захисту даних.

Таким чином, самоорганізовані безпровідні мережі становлять важливий напрям розвитку сучасних технологій зв’язку, відкриваючи нові можливості для створення автономної, масштабованої та ефективної цифрової інфраструктури.

2 СТЕГАНОГРАФІЧНІ МЕТОДИ ШИФРУВАННЯ ІНФОРМАЦІЇ

2.1 Визначення та основні поняття стеганографії

Обсяг інформації, яку людина щоденно взаємодіє, постійно зростає, а значна частина цієї інформації надходить саме з електронних джерел. Така тенденція спричиняє виникнення нових загроз, зокрема ризиків порушення цілісності даних, втрати конфіденційності, а також можливості блокування чи знищення інформаційних ресурсів. У відповідь на ці та подібні виклики постала потреба у створенні систем захисту інформації, головним завданням яких є забезпечення її конфіденційності, цілісності та доступності. Проблема захисту інформації від несанкціонованого втручання залишається актуальною й сьогодні, набуваючи особливого значення у зв’язку зі зростанням кіберзагроз.

Інформаційна безпека, згідно зі статтею 17 Конституції України, разом із забезпеченням суверенітету, економічної стабільності та територіальної цілісності, є ключовим завданням держави [15]. Досягнення цього можливе завдяки впровадженню сучасних законодавчих актів, розвиток безпечних інформаційних технологій, створення комплексної національної інформаційної інфраструктури, а також формування зрілих інформаційних відносин.

У Законі України «Про захист інформації в інформаційно-телекомунікаційних системах» подано визначення поняття «захист інформації» як «діяльність, спрямована на запобігання несанкціонованим діям щодо інформації в системі» [16]. Одним із напрямів цієї діяльності є технічний захист інформації, а також захисту режиму доступу до неї. Під доступом до інформації розуміють можливість особи (фізичної або юридичної) обробляти дані в межах

інформаційної системи, а будь-які дії, що порушують встановлені правила доступу, вважаються несанкціонованими.

Серед основних методів захисту інформації традиційно виокремлюють її шифрування та приховування. Ще в давнину сформувалися два ключові підходи, які залишаються актуальними і в сучасному світі – це криптографія та стеганографія.

Криптографія спрямована на забезпечення конфіденційності переданої інформації шляхом її шифрування, тобто перетворення змісту повідомлення у форму, недоступну для сторонніх осіб. Як окрема галузь знань, вона розробляє засоби й алгоритми, що гарантують не лише конфіденційність, а й цілісність та автентичність даних. Основною метою криптографії є унеможливлення ознайомлення сторонніх з інформацією під час її передавання. Водночас слід зазначити, що навіть шифроване повідомлення залишається вразливим до фізичного знищення або блокування, тож саме по собі шифрування не забезпечує абсолютного захисту.

Підвищити рівень інформаційної безпеки дозволяє поєднання криптографічних методів із технологіями прихованого передавання даних. Зокрема, стеганографія не лише оберігає зміст, а й маскує сам факт існування повідомлення. На відміну від шифрування, при якому наявність секретної інформації легко встановити, стеганографічні методи дозволяють вбудовувати повідомлення у звичайний цифровий контент – текст, зображення чи аудіо – не викликаючи підозри в потенційного спостерігача.

На рисунку 2.1 представлено порівняння криптографії та стеганографії у передачі інформації, яке демонструє основні відмінності між методами приховання даних і захисту їхнього змісту.

Витоки стеганографії сягають глибокої давнини. Її перші прояви були зафіксовані ще до нашої ери. У ті часи застосовувалися переважно фізичні

способи приховування, наприклад, на шкірі раба писали повідомлення, після чого чекали, поки відросте волосся, і лише тоді його відправляли до отримувача. Також широко використовувались різноманітні способи маскування тексту в межах звичайного листування – від написів, зроблених молоком, до використання складних хімічних речовин, які виявлялись лише після спеціальної обробки. Її суть полягає в тому, щоб інтегрувати секретну інформацію в звичний інформаційний потік або носій (наприклад, зображення, аудіо чи текст), тим самим зберігаючи в таємниці сам факт передавання повідомлення.

Рисунок 2.1 – Порівняльна схема криптографії та стеганографії у прихованому передаванні даних

Стеганографія (від грец. steganos – прихований, таємний, та grapho – писати, записувати) – це сукупність методів і засобів, що використовуються для маскування самого факту передачі повідомлення [17].

Одним із важливих понять стеганографії є стеганосистема – комплекс методів і засобів, що призначені для створення прихованого каналу передачі даних.

Стеганосистема включає наступні компоненти (рис. 2.2):

а) контейнер – файл (текст, зображення, аудіо тощо), який використовується для маскування прихованого повідомлення;

б) порожній контейнер – початковий файл без вбудованої інформації;

в) вбудоване повідомлення – секретні дані, які вставляються в контейнер для подальшої передачі;

г) стеганографічний канал – засіб передачі, через який передається контейнер із прихованими даними;

д) стегоключ – секретна інформація, необхідна для процесу вбудовування та/або витягання прихованого повідомлення.

Рисунок 2.2 – Структура стеганосистеми: основні компоненти та принципи взаємодії

Стеганографія має на меті забезпечення непомітного перенесення інформації через звичайну комунікаційну мережу. Основні вимоги до стеганографічних систем включають непомітність (imperceptibility), коли зміни, внесені у носій, повинні бути настільки незначними, щоб навіть при детальному аналізі спостерігач не зміг виявити наявність прихованої інформації, високу місткість (capacity), яка визначає обсяг даних, що можуть бути в ньому сховані, при цьому зберігаючи їхню непомітність, стійкість до атак (robustness), що забезпечує відновлюваність зашифрованого повідомлення при зміні носія, наприклад, унаслідок стискання, масштабування чи фільтрації, а також безпеку, яка гарантує, що навіть при виявленні схованих даних розшифрування інформації буде неможливим без наявності додаткового ключа чи паролю.

2.2 Класифікація стеганографічних методів

Залежно від каналу, носія та принципу впровадження прихованих даних, усі методи стеганографії умовно поділяють на кілька категорій.

Класифікація за типом носія наступна (рис. 2.3):

– текстова стеганографія приховує таємне послання всередині шматка тексту, реалізується за допомогою синонімів, маніпуляцій зі шрифтами, пробілами, регістром літер або невидимих символів (наприклад, найпростіший варіант стеганографії тексту може використовувати першу літеру в кожному реченні для формування прихованого повідомлення);

– стеганографія зображення найбільш поширений носій, що дозволяє маніпулювати найменш значущими бітами пікселів зображення без візуального спотворення (наприклад, одне зображення може бути приховане всередині іншого, використовуючи нижні значущі біти кожного пікселя зображення для представлення прихованого зображення);

– аудіо- та відеостеганографія – застосовується вставлення даних у фоновий шум файлів (зображень/відео), спектральні компоненти або часові відрізки сигналу. Однією з простих форм аудіостеганографії є «зворотне маскування», при якому секретні повідомлення відтворюються задом наперед на доріжці (вимагаючи від слухача відтворити весь трек задом наперед). Більш складні методи можуть включати найменш значущі біти кожного байта в аудіофайлі, подібно до стеганографії зображення;

– мережева стеганографія – це складна техніка цифрової стеганографії, яка приховує інформацію в мережевому трафіку. Наприклад, приховані повідомлення вбудовуються у службові поля протоколів (наприклад, TCP, IP, DNS, Bluetooth). Відправник може навіть передавати інформацію в залежності від часу між відправкою різних пакетів;

– інші цифрові канали включають стеганографію в QR-кодах, структурах PDF-документів, HTML-коді, метаданих тощо.

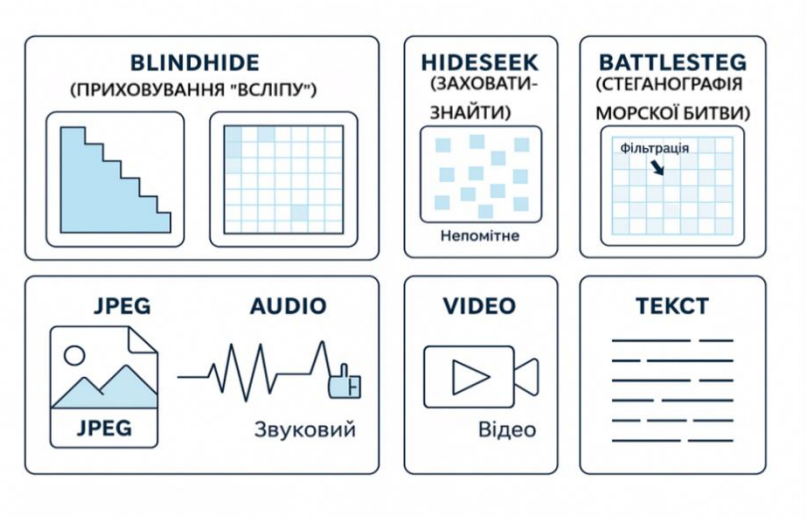

Рисунок 2.3 – Класифікація стеганографічних методів за типом носія інформації

За способом вставлення прихованої інформації методи стеганографії поділяються на такі групи:

– методи прямої заміни (Substitution-based) – найбільш прості й поширені, реалізуються шляхом заміни окремих бітів чи символів у носії;

– методи перетворень (Transform domain methods) – передбачають попередню обробку носія (наприклад, дискретне косинусне перетворення – DCT), після чого дані вбудовуються в коефіцієнти перетворень;

– структурні методи (Structure-based) – використовують зміну внутрішньої структури носія (наприклад, порядку сегментів файлу) без зміни відображуваного змісту;

– статистичні методи забезпечують передачу даних за рахунок зміни статистичних характеристик сигналу або файлу, наприклад, гістонограми яскравості.

Класифікація за технікою вбудовування:

а) вбудовування в просторовій області (Spatial Domain Embedding) – приховання даних безпосередньо у значеннях пікселів (наприклад, зміна молодших бітів LSB (Least Significant Bit));

б) вбудовування в частотній області (Transform Domain Embedding) – конвертування зображення у частотне представлення, після чого відбувається вбудовування у його частотні компоненти (наприклад, вбудовування у середні частоти зображення JPEG (DiscreteCosine Transform)).

Також методи класифікуються за рівнем стійкості до змін:

– стійкі (robust) методи – зберігають приховану інформацію навіть після компресії, повторного кодування або цифрового оброблення сигналу (наприклад, використовуються для цифрових водяних знаків);

– вразливі (fragile) методи – навпаки, реагують на будь-яке спотворення носія, що робить їх ефективними для виявлення фальсифікації або несанкціонованого доступу до даних;

– напівстійкі (умовно стійкі) методи – часткова стійкість до певного класу атак (наприклад, лише до JPEG-стиснення, але не до обрізання зображення).

Класифікація за стратегією приховування:

а) послідовне (сліпе) приховування – дані вставляються послідовно – від першого пікселя до останнього (наприклад, LSB у просторі зображення);

б) псевдовипадкове приховування – місця для вбудовування вибираються на основі псевдовипадкових чисел (PRNG), які генеруються з пароля або ключа (наприклад, вбудовування в пікселі, фрейми, частоти);

в) адаптивне приховування – аналізуються властивості контейнера (яскравість, контрастність, текстури) для вибору найменш помітних ділянок (наприклад, фільтрація пікселів зображень або ділянок аудіо);

г) приховування із заповненням – вбудовування даних до певного рівня контейнера (наприклад, верхня частина зображення) до заповнення;

д) приховування після маркерів – дані додаються після «кінця» основного файлу (наприклад, після EOF JPEG);

е) метод зі зміною структури контейнера – використання нестандартних структур, slack-space, метаданих, нестандартних заголовків.

2.3 Метод найменш значущого біта у зображеннях

Один із найпоширеніших носіїв завдяки великій різноманітності піксельних значень є метод LSB, метод заміни найменш значущих бітів цифрового зображення, коли невеликі зміни в останніх бітах пікселя не є візуально або акустично помітними для людини.

Основні етапи методу представлено на рисунку 2.4.

Однією з ключових переваг методу LSB є його простота реалізації та висока здатність до вбудовування інформації. Проте цей метод має і певні

обмеження, наприклад, під час обробки зображень, таких як стиснення або фільтрація, можлива втрата або спотворення прихованої інформації, що ставить під сумнів ефективність його використання в таких випадках.

Рисунок 2.4 – Основні етапи методу найменш значущого біта у зображеннях

Суттєвим недоліком методу LSB є його чутливість до роздільної здатності зображення, тобто при використанні зображень «малого» розміру виникає ймовірність значної візуальної різниці між сусідніми пікселями, що може видати факт вбудовування даних. Тому доцільно застосовувати зображення з високою роздільною здатністю, де подібні зміни майже непомітні.

Як контейнери для приховування інформації доцільно використовувати зображення формату `*.bmp` з високою роздільною здатністю і глибиною кольору 24 або 32 біти. Таємні дані також можуть бути представлені у форматах `*.bmp`, `*.gif`, `*.png`, `*.jpeg`.

Підвиди LSB-алгоритмів для растрових зображень без палітри (рис. 2.3):

а) приховування "всліпу" (BlindHide) – дані вписуються в наймолодші біти кольорів пікселів, починаючи з верхнього лівого кута зображення і послідовно до правого нижнього, розміщення даних у зображенні при цьому є нерівномірним, тобто якщо весь контейнер не буде заповнений, зміни торкнуться лише верхньої частини зображення;

б) заховати-знайти (HideSeek) – псевдовипадкове розміщення прихованого повідомлення, використовуючи пароль як основу для генерації послідовності, однак при цьому не враховується специфіку самого зображення-контейнера;

в) фільтрація перед вставкою (FilterFirst) – спочатку виконується аналіз зображення для виявлення тих пікселів, у яких зміни будуть найменш помітними для людського ока, потім прихована інформація вписується саме у ці зони;

г) стеганографія морської битви (BattleSteg) – на першому етапі виконується фільтрація зображення, далі дані розміщуються у «найбезпечніші» області з використанням псевдовипадкових позицій, аналогічно до HideSeek. Даний метод є найбільш розвиненим серед перерахованих.

За результатами проведеного аналізу можна сказати, що незважаючи на простоту і легкість реалізації методу LSB, його модифікації з використанням частотних характеристик зображення можуть мати дуже значний рівень секретності і стабільності. Більш того, останнім часом розроблені методи запису на основі фракталів і хаотичних систем, які вимагають подальшого вивчення.

2.4. Сфери використання стеганографії

Стеганографія є важливим засобом захисту інформації, що знаходить застосування у різних сферах людської діяльності (рис. 2.5). Одним із ключових напрямків її використання є захист авторських прав та обмеження копіювання. Використовується для запобігання незаконному дублюванню в електронній торгівлі, захисту цифрового контенту (наприклад, DVD), а також при розповсюдженні мультимедіа, зокрема відео на запит.

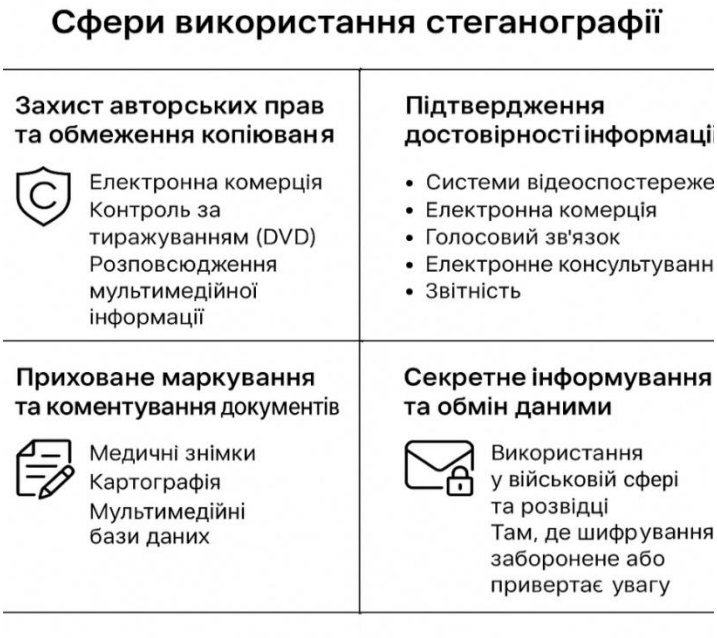

Рисунок 2.5 – Сфери використання стеганографії

Підтвердження достовірності інформації (аутентифікація) є актуальним в системах відеоспостереження, електронному бізнесі, голосових сервісах, а також у захищеному електронному документообігу. Приховане маркування та коментування документів застосовується у таких галузях, як медична діагностика (цифрові знімки), картографічні дані, а також в мультимедійних базах даних для вбудовування невидимих міток чи описів. Секретне інформування та обмін даними використовується для непомітної передачі інформації у військовій сфері, розвідці, а також у ситуаціях, де використання криптографії є недоцільним або забороненим.

Висновки до другого розділу

Стеганографія є важливою складовою інформаційної безпеки, оскільки вона не лише захищає зміст повідомлення, а й приховує сам факт його передавання. Висвітлено історичні витоки цієї технології, ключові поняття та структуру стеганосистеми, що включає контейнер, стеганографічний канал та стегоключ. Проведений аналіз показав, що одним із найефективніших способів прихованої передачі інформації є метод найменш значущого біта. Такий підхід дозволяє впроваджувати дані практично непомітно, зберігаючи візуальну цілісність носія.

У ході дослідження також було здійснено докладну класифікацію стеганографічних методів за критеріями типу носія, механізму вбудовування та ступеня стійкості до зовнішніх втручань. Визначено ключові вимоги до сучасних стеганографічних систем, серед яких – малопомітність, висока місткість для вміщення даних, стійкість до змін носія та загальний рівень безпеки.

Значущість стеганографії як інструменту забезпечення інформаційної безпеки підтверджується її активним використанням у військових технологіях, сфері розвідки, цифрових мультимедійних системах, а також у системах електронного документообігу. Комплексне застосування стеганографічних методів разом із криптографічними засобами суттєво підвищує ефективність захисту інформації, ускладнюючи несанкціонований доступ і зменшуючи ризик виявлення самої передачі даних.

3 МОЖЛИВОСТІ ТЕХНОЛОГІЇ BLUETOOTH ДЛЯ ПРИХОВАНОЇ ТА БЕЗПЕЧНОЇ ПЕРЕДАЧІ ІНФОРМАЦІЇ

3.1 Загальна характеристика технології Bluetooth

Bluetooth – це технологія бездротового зв’язку, яка підтримує зв’язок між пристроями на коротких відстанях і може виконувати бездротовий зв’язок між мобільними телефонами, КПК, бездротовими гарнітурами, ноутбуками та іншими відповідними зовнішніми пристроями [18]. Технологія базується на розподіленій мережевій структурі, застосовує швидке стрибкоподібне перемикання частот та короткі пакети даних. Bluetooth підтримує зв’язок типу «точка-точка» та «точка-багатоточка», працює у діапазоні ISM 2,4 ГГц зі швидкістю передачі до 1 Мбіт/с [19]. Bluetooth має декілька версій, кожна з яких розширює функціонал і покращує характеристики (табл. 3.1).

Таблиця 3.1 – Версії Bluetooth і їхні особливості [20, 21, 22]

Версія	Рік	Основні особливості
1.0-1.2	1999-2003	основи протоколу, швидкість до 721 кбіт/с
2.0 + EDR	2004	підвищена швидкість (до 3 Мбіт/с)
3.0 + HS	2009	передача великих обсягів через Wi-Fi
4.0 (BLE)	2010	низьке енергоспоживання, нові профілі для IoT
5.0-5.4	2016-2023	збільшена дальність і швидкість, підтримка Mesh
6.0	2024	Bluetooth Channel Sounding, покращена ефективність
6.1	2025	підвищена конфіденційність і енергоефективність

Bluetooth-з’єднання можуть будуватися за різними топологіями, серед яких найпростіша – структура «point-to-point», проте підтримуються й складніші – piconet та scatternet [23]. Piconet – це мережа з одним головним (master) та до 7 підлеглих (slave) пристроїв (рис. 3.1). Scatternet – це об’єднання декількох piconet, де деякі пристрої можуть бути одночасно у кількох мережах, що дозволяє масштабувати зв’язок. Розуміння топології з’єднань дозволяє ефективніше використовувати Bluetooth у практичних застосуваннях.

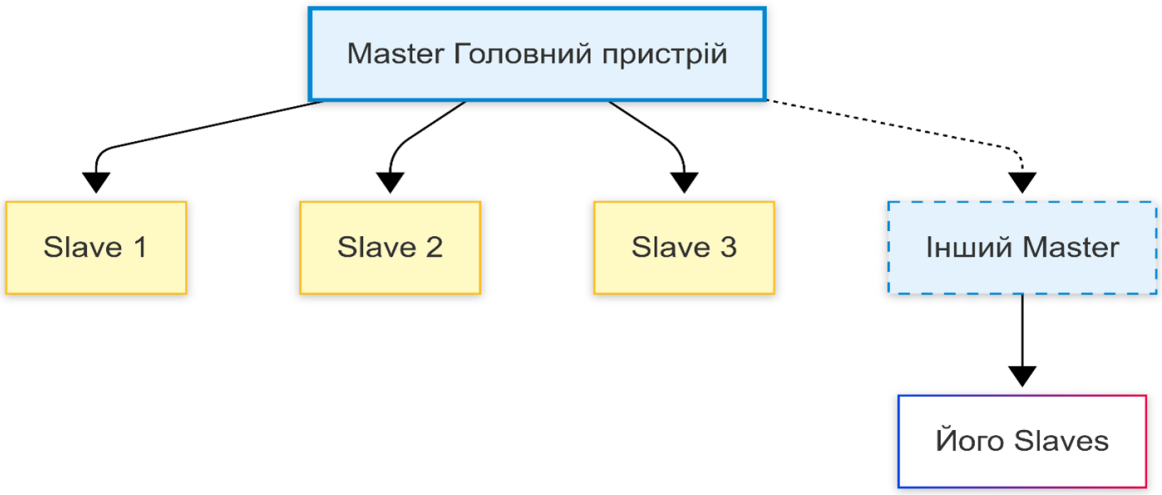

Рисунок 3.1 – Схема взаємодії головного пристрою та підлеглих вузлів у моделі Master-Slave

Архітектура Bluetooth складається з кількох рівнів протоколів:

– Baseband (BB) – управління радіочастотами;

– Link Manager Protocol (LMP) – керування з’єднаннями;

– Logical Link Control and Adaptation Protocol (L2CAP) – адаптація та мультиплексування каналів;

– Service Discovery Protocol (SDP) – пошук доступних сервісів.

Також існують додаткові протоколи (RFCOMM, TCS, PPP, IP, TCP/UDP, OBEX тощо), які розширюють можливості.

Безпека Bluetooth базується на аутентифікації, шифруванні та паруванні пристроїв, що забезпечує захист переданих даних. Проте, технологія має відомі вразливості, наприклад, атаки Bluebugging, Bluesnarfing та BlueBorne можуть призвести до несанкціонованого доступу або контролю над пристроями. Bluebugging, дозволяє отримати контроль над пристроєм через Bluetooth-з'єднання, здійснюючи приховане надсилання повідомлень або дзвінки. Bluesnarfing здійснюється через несанкціонований доступ до файлів на пристрої, що може призвести до викрадення контактів, повідомлень та інших конфіденційних даних. BlueBorne використовує вразливості в Bluetooth-протоколах для дистанційного захоплення контролю над пристроєм без потреби в активному сполученні чи взаємодії користувача. Найбільш небезпечною є остання атака.

Для мінімізації ризиків рекомендовано регулярно оновлювати ПЗ, використовувати останні версії Bluetooth, вимикати з'єднання, коли вони не потрібні, та уникати підключення до невідомих пристроїв.

3.2 Методи прихованої передачі інформації через Bluetooth

Bluetooth, крім легального обміну даними, може використовуватися для прихованої передачі інформації. Ці методи зазвичай реалізуються шляхом стеганографії, обфускації трафіку або зловживання службовими каналами. Приховані канали створюються всередині існуючих протоколів, що ускладнює їх виявлення.

Одним із перспективних напрямів дослідження є вивчення методів прихованої передачі інформації через протоколи Bluetooth. На рівні службових

параметрів, зокрема протоколу Service Discovery Protocol (SDP) [24], можливе використання стеганографії. Зловмисник може імітувати легітимний сервіс, вбудовуючи приховані повідомлення в службову інформацію, зокрема в 128-бітні поля UUID. Ці поля, що рідко перевіряються користувачем або операційною системою, можуть містити заздалегідь закодовані символьні повідомлення. Такий спосіб дає змогу передавати невеликі обсяги даних, які розпізнаються лише за допомогою спеціалізованого програмного забезпечення, що підвищує скритність обміну.

Іншим методом є використання назв пристроїв або пов’язаних атрибутів, таких як Bluetooth Device Name [25]. Наприклад, стандартна назва пристрою на кшталт "Galaxy S24" може містити закодований текст. Під час сесії передавання можна реалізовувати зміну імені за визначеним алгоритмом, формуючи своєрідну послідовність повідомлень. Перевага цього методу полягає в тому, що для зчитування даних не потрібно встановлювати повноцінне з’єднання, що робить його ефективним у peer-to-peer середовищі з мінімальним ризиком виявлення.

Також існує можливість використання протоколу Logical Link Control and Adaptation Protocol (L2CAP) для створення прихованих каналів [26]. Цей протокол відповідає за передачу даних між протоколами верхнього рівня в Bluetooth-стосі, і зловмисники можуть модифікувати L2CAP-пакети для вбудовування додаткової інформації у вигляді службових повідомлень або нестандартних фрагментів. Такі пакети часто не аналізуються системами безпеки, що дозволяє обходити фаєрволи або засоби контролю трафіку на мобільних пристроях.

У пристроях, що підтримують Bluetooth Low Energy (BLE), можливим вектором передачі є рекламні пакети (advertising packets), які транслюються періодично без попереднього встановлення з’єднання [27]. Ці пакети можуть

містити довільну інформацію у полях Manufacturer Specific Data, зокрема зашифровану або закодовану. Через доступність такого трафіку для будь-яких спостерігачів у зоні дії сигналу, метод підходить для односторонньої передачі з високим ступенем автономності.

Ще одним цікавим способом є використання скатернет-структури – ситуації, коли Bluetooth-пристрій одночасно належить до кількох піко-мереж [28]. Такий пристрій може виступати в ролі ретранслятора, передаючи приховану інформацію між логічно роз'єднаними вузлами. Це дозволяє формувати багатоступеневі ланцюжки обміну, що важко відстежити з точки зору традиційного аналізу мережевого трафіку.

У таблиці 3.2 представлено класифікацію методів прихованої передачі даних через Bluetooth з урахуванням технічних особливостей кожного підходу.

Таблиця 3.2 – Порівняльна характеристика методів прихованої передачі через Bluetooth

№	Метод	Ризик виявлення	Рівень протоколу	Коментар
1	стеганографія в UUID	дуже низький	SDP	слабкий контроль вмісту UUID
2	зміна імені пристрою	низький	GAP (Generic Access)	ефективно для коротких фраз
3	передача через L2CAP	середній	L2CAP	створення нестандартних пакетів
4	advertising пакети BLE	високий	BLE Advertising	одностороння передача, обмежена довжина
5	прихована ретрансляція в scatternet	низький	топологічний рівень	передача в обхід основної мережі

Поява цих методів вимагає розвитку ефективних засобів виявлення і захисту [29]. Для цього використовуються як аналіз мережевого трафіку, так і програмні рішення для моніторингу активності пристроїв.

3.3 Метод найменш значущого біта у стеганографії для Bluetooth

Метод LSB, який є одним із класичних методів стеганографії, у контексті Bluetooth може бути застосований для прихованого кодування інформації в таких полях, як адреси, ідентифікатори (UUID), рекламні пакети (advertising packets), а також в інших службових або даних протоколів.

Для Bluetooth, зокрема BLE, LSB-методи можуть реалізовуватись шляхом заміни найменш значущих бітів у полі Manufacturer Specific Data або інших параметрах рекламних пакетів, що транслюються без встановлення з’єднання. Оскільки багато з цих полів не проходять жорсткий контроль або фільтрацію, прихована інформація може залишатися непоміченою звичайними інструментами моніторингу.

Застосування методів LSB у Bluetooth-стеганографії відкриває низку технічних і практичних переваг, що роблять цей підхід перспективним у сфері прихованої передачі інформації в бездротових мережах, серед яких зокрема:

а) мінімальна спотворюваність сигналу – зміни в цифрових даних на рівні найменш значущих бітів практично не впливають на сприйняття або функціональність переданого повідомлення, тобто дані можуть бути вбудовані в службову або мультимедійну інформацію без помітного впливу на якість передачі чи швидкодію каналу;

б) висока швидкість вбудовування та витягнення інформації завдяки низькій обчислювальній складності LSB-алгоритмам;

в) сумісність із різними типами даних, що розширює можливості використання LSB-стеганографії у багатьох сценаріях – від побутових до спеціалізованих (військових, розвідувальних, медичних);

г) маскування під звичайний трафік, які складно виявити без спеціалізованих інструментів, що зменшує ризик перехоплення або виявлення прихованої інформації;

д) можливість інтеграції з іншими методами захисту, наприклад, з криптографічними засобами, що забезпечить подвійний рівень безпеки: навіть у разі виявлення прихованого повідомлення його зміст залишатиметься зашифрованим і недоступним без відповідного ключа;

е) мала енергозатратність (майже не впливає на споживання енергії мобільних або портативних пристроїв), що особливо важливо для Bluetooth-систем з автономним живленням.

Попри низку переваг, використання LSB-методів у Bluetooth-стеганографії має і певні обмеження, які необхідно враховувати при проєктуванні систем прихованої передачі даних. Насамперед, до недоліків можна віднести обмежений обсяг даних, який можна закодувати: через невелику кількість доступних бітів для вставлення прихованої інформації обсяг повідомлень залишається незначним, що унеможливлює передавання великих обсягів даних. Крім того, висока вразливість до модифікацій є суттєвим ризиком: пакети Bluetooth можуть бути змінені випадково або навмисно в процесі передавання, що призводить до спотворення або повної втрати прихованого повідомлення. Ще одним важливим обмеженням є обмежена придатність для двонаправленої безз’єднальної комунікації, оскільки приймач повинен мати можливість оперативного зчитування змінених пакетів, інакше цілісність та доступність інформації опиняється під загрозою.

Застосування LSB-методів у Bluetooth стеганографії є ефективним способом прихованої одно- або двонапрямної передачі малих обсягів секретних даних. Поєднання LSB з іншими методами, наприклад, із зміною імені пристрою чи використанням UUID, дозволяє підвищити обсяг і складність передачі та ускладнити її виявлення.

У контексті Bluetooth LSB можна використати для прихованої передачі інформації у полях службових або інформаційних пакетів, зокрема в:

– полях Manufacturer Specific Data у BLE advertising пакетах;

– байтах L2CAP-пакетів, які не перевіряються або не використовуються за призначенням.

Наприклад, вбудувати повідомлення "OK" (у вигляді двійкових бітів) в LSB 16 байтів даних Bluetooth-пакету можна замінивши останній біт кожного байта на біт із повідомлення "OK" (табл. 3.3).

Таблиця 3.3 – Схема LSB-стеганографії для передачі повідомлення "OK"

№	Байт (до)	Вставлений біт	Байт (після)
1	11001010	0	11001010 (не змінюється)
2	10011000	1	10011001
3	01100111	0	01100110
4	10101101	0	10101100
5	11110000	1	11110001
6	00011110	1	00011111
7	00100011	1	00100011 (не змінюється)
8	01101001	1	01101001 (не змінюється)
9	01010101	0	01010100
10	00110011	1	00110011 (не змінюється)
11	11001100	0	11001100 (не змінюється)

Кінець таблиці 3.3.

12	10000001	0	10000000
13	11111110	1	11111111
14	01111100	0	01111100 (не змінюється)
15	10101010	1	10101011
16	01000010	1	01000011

Повідомлення з 16 бітів (2 байти="OK") зашифроване в найменш значущих бітах 16 байтів Bluetooth-даних.

При отриманні можна пройтись по 16 байтам, витягнути LSB з кожного і зібрати назад у байти.

Таким чином, змінюється лише останній біт у кожному байті. Людське око чи типові моніторингові системи не виявляють змін, але приймач, який знає розташування бітів, може легко реконструювати оригінальне повідомлення.

Висновки до третього розділу

Таким чином, технологія Bluetooth є поширеним і зручним засобом бездротового зв’язку, але її можливості можуть використовуватись не лише для легального обміну даними, а й для прихованої передачі інформації. Аналіз існуючих методів показує, що зловмисники можуть застосовувати стеганографічні підходи, модифікувати службові параметри та використовувати топологічні особливості (наприклад, scatternet) для організації несанкціонованих каналів.

Заходи кібербезпеки мають враховувати специфіку Bluetooth-протоколів, особливо в умовах зростання використання BLE і нових версій стандарту.

Важливими є методи аналізу трафіку, системи захисту на рівні операційних систем і відповідальні практики використання пристроїв.

Дослідження можливостей прихованої передачі інформації через Bluetooth є актуальним як для забезпечення безпеки бездротових мереж, так і для розробки нових методів захисту даних від несанкціонованого доступу.

4 МОДЕЛЮВАННЯ ПЕРЕДАЧІ ФРАГМЕНТА ІНФОРМАЦІЇ СТЕГАНОГРАФІЧНИМ СПОСОБОМ У ПРОГРАМНОМУ СЕРЕДОВИЩІ MATLAB

4.1 Постановка задачі моделювання

У даному розділі розглядається моделювання процесу прихованої передачі інформації за допомогою стеганографічного методу найменш значущого біта через канал бездротового зв’язку Bluetooth.

Основною метою моделювання був аналіз стійкості стеганографічного способу передавання інформації до спотворень, що виникають у процесі передачі по нестабільному Bluetooth-каналу, та визначення рівня збереження прихованих даних після передачі.

Процес моделювання включав кілька етапів: розробку математичної моделі вбудовування прихованої інформації у відкриті дані (контейнер) методом LSB, моделювання впливу середовища передачі Bluetooth у вигляді спотворень даних із певною ймовірністю інверсії бітів, розробку алгоритму вилучення прихованої інформації після проходження через канал, а також оцінку точності передачі прихованої інформації шляхом розрахунку коефіцієнта бітових помилок (BitErrorRate, BER).

При моделюванні враховувалися певні обмеження та припущення. Зокрема, контейнером виступав довільний масив бітів, що імітував службові дані або медіа інформацію. Вплив середовища передачі моделювався випадковим інвертуванням бітів із ймовірністю p, яка змінюється в діапазоні [0, 0.3]. Механізм вбудовування інформації не змінював суттєво основну структуру контейнера, що забезпечувало допустиму якість передачі основних даних. При

відновленні прихованої інформації передбачалося, що обидві сторони (відправник та отримувач) використовують однакову схему вбудовування та вилучення.

Математична постановка задачі передбачала використання таких позначень: M – бітовий масив контейнера розміром N біт, S – бітовий масив прихованого повідомлення розміром n біт, де $n \leq N$, $E(M,S)$ – функція вбудовування повідомлення S у контейнер $M, C = E(M,S)$ – модифікований контейнер із прихованим повідомленням, $\hat{C}$ – контейнер після проходження через канал Bluetooth зі спотвореннями, $D(\hat{C})$ – функція вилучення прихованої інформації з прийнятого контейнера.

Таким чином, процес моделювання включав такі етапи: побудову контейнера із вбудованою інформацією ($C = E(M,S)$), передачу контейнера через канал із ймовірністю спотворення p ($\hat{C} = Channel(C,p)$), вилучення прихованої інформації з прийнятого контейнера ($\hat{S} = D(\hat{C})$), а також оцінку точності передачі за допомогою коефіцієнта бітових помилок (BER), який визначається як:

$$BER = \frac{1}{n}\sum_{i=1}^{n}(S_i \oplus \hat{S}_i), \tag{4.1}$$

де BER – бітова ймовірність помилки (BitErrorRate);

n – загальна кількість переданих бітів;

S_i – біт оригінального (правильного) повідомлення,;

$\hat{S}_i$ – біт прийнятого (отриманого) повідомлення;

$\oplus$ – оператор виключного "АБО" (XOR), який дорівнює 1, якщо біти різні, і 0, якщо однакові.

Оцінка результатів моделювання здійснювалася за такими критеріями: рівень бітових помилок при різних ймовірностях спотворень p (за формулою (4.1), залежність точності передачі від ступеня спотворення, а також якість відновленого повідомлення за різних умов моделювання.

Запропонована модель дозволяє не лише протестувати працездатність стеганографічного способу передачі даних через нестабільний канал Bluetooth, але й оцінити реальні обмеження та межі застосування такого підходу у бездротових мережах.

4.2 Розробка алгоритму

У запропонованому коді використовувався метод стеганографії LSB для приховування повідомлень у каналі передачі даних через Bluetooth. Алгоритм стеганографії LSB для Bluetooth складається з двох основних етапів: приховування повідомлення та його вилучення (рис. 4.1).

На етапі приховування повідомлення здійснювалася підготовка даних, яка включала перетворення текстового повідомлення у послідовність бітів, де кожен символ представлений 8 бітами, а також додавання службової інформації у вигляді 16 бітів для збереження довжини повідомлення. Далі відбувалося вбудовування даних, що передбачало генерацію «легального» корисного навантаження у вигляді масиву випадкових байтів, заміну найменш значущих бітів у кожному байті корисного навантаження на біти прихованого повідомлення, причому за замовчуванням використовувалося 1-2 найменш значущих біти кожного байта. Після цього формувався пакет, який містив спеціальний заголовок (0xAA, 0x55, 0xAA, 0x55) для ідентифікації, розмір даних (2 байти) та модифіковані дані з прихованим повідомленням. Заключним

етапом була передача через Bluetooth, що включала встановлення Bluetooth-з’єднання, відправлення сформованого пакета та закриття з’єднання.

Особливості алгоритму включали:

– його непомітність, оскільки зміна найменш значущих бітів майже не впливала на загальні характеристики даних;

– гнучкість, що дозволяло налаштовувати кількість використовуваних LSB для компромісу між ємністю та непомітністю;

– надійність завдяки використанню заголовків і полів довжини, що забезпечують правильне вилучення даних;

– самодостатність, оскільки у прихованому повідомленні міститься інформація про його довжину, що дозволяє правильно вилучити дані.

Метод LSB є одним з найпростіших і найпоширеніших методів стеганографії, а застосування його до Bluetooth-комунікації дозволяє створити прихований канал зв’язку, що практично не виявляється при поверхневому аналізі даних.

4.3 Моделювання передачі та отримання прихованої інформації

Моделювання передачі та отримання прихованої інформації виконувалось у три етапи.

На початковому етапі ми написали базовий код для приховування повідомлення «Hello» у Bluetooth-полі за допомогою методу LSB (додаток А). Спочатку код працював, однак при моделюванні шуму (випадкові зміни значень у полі) ми отримали спотворене повідомлення, що не відповідало оригіналу (рис. 4.2).

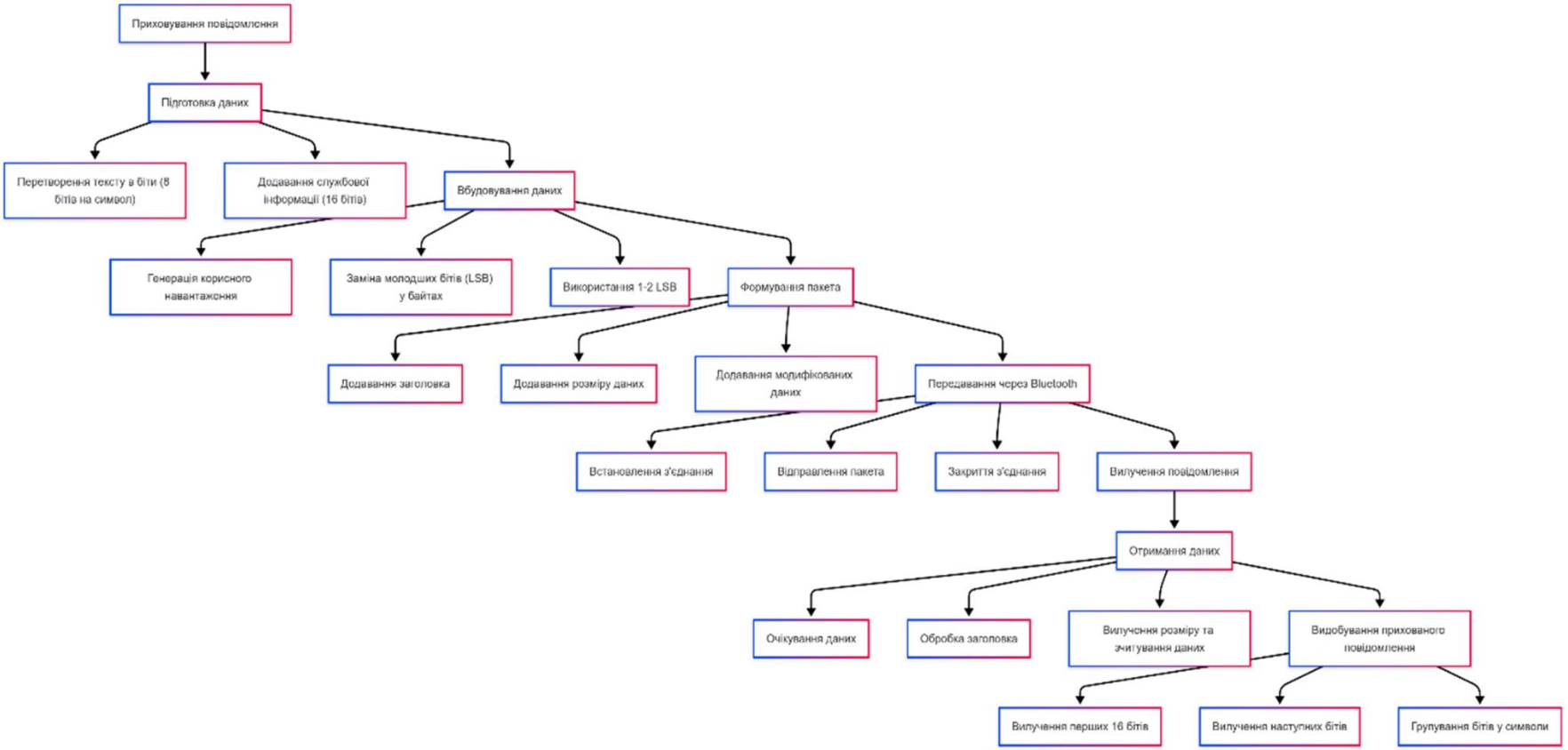

Рисунок 4.1 – Алгоритм стеганографії LSB для Bluetooth

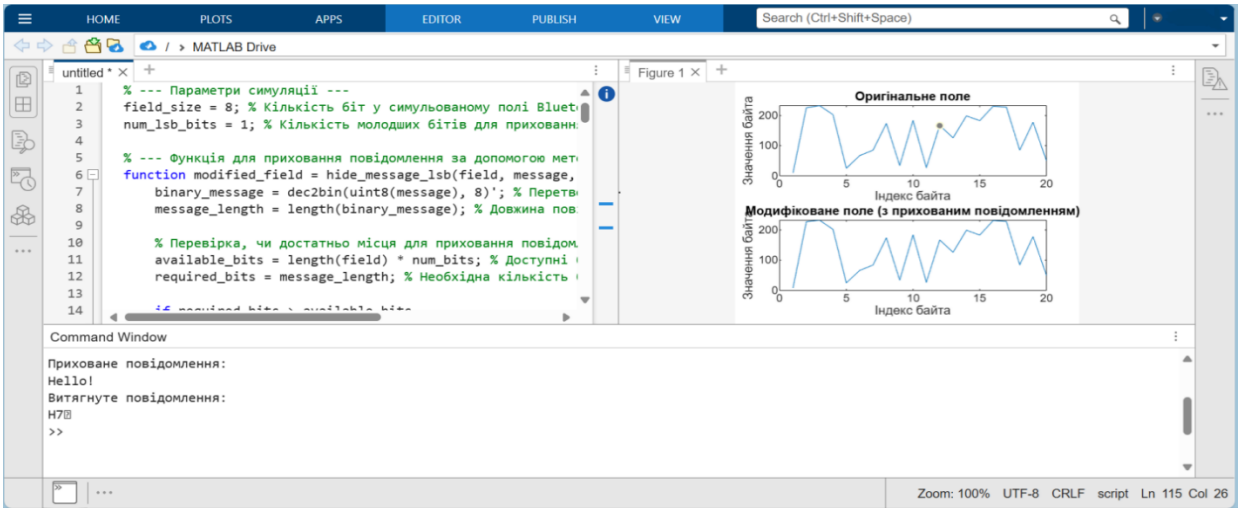

Рисунок 4.2 – Лістинг коду (додаток А)

Спотворення повідомлення (наприклад, витягнутий текст виглядає як H7), може бути викликане кількома причинами:

– помилки при прихованні біта (при кодуванні LSB метод некоректно вставляє біти (наприклад, через порушення послідовності);

– шуми в даних (при моделюванні або реальній передачі даних за допомогою Bluetooth виникає шум, і кілька бітів у модифікованому полі стають спотвореними, оскільки найменш значущі біти дуже чутливі до завад);

– контрольна сума відсутня або некоректно перевіряється (якщо контрольна сума не додається або не перевіряється при вилученні, навіть невелике спотворення призводить до некоректного декодування);

– проблеми з кодуванням символів (кодування тексту (UTF-8 vs ASCII) може спричинити проблему, особливо якщо витягуються некоректні байти або відсутня підтримка спеціальних символів).

Із графіка отриманого коду (рис. 4.3) бачимо, зміни настільки малі, що вони практично непомітні для людського ока, але їх можна відновити через алгоритм вилучення. Лінія верхнього графіка («Оригінальне поле») стабільна, що демонструє відсутність змінених даних за допомогою Bluetooth

до приховання. Нижній графік («Модифіковане поле (з прихованим повідомленням)») демонструє невеликі зміни в лінії порівняно з оригінальним полем, що підтверджує модифікацію даних для приховання повідомлення.

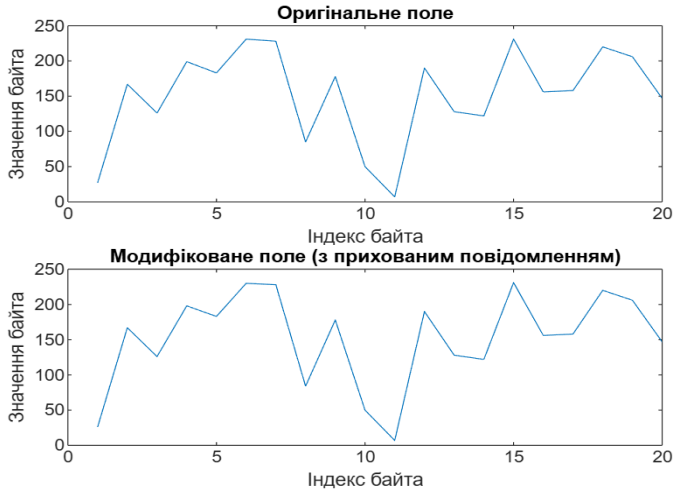

Рисунок 4.3 – Графік отриманого коду

На другому етапі були внесені правки (додаток Б), зокрема додано контрольну суму для перевірки цілісності повідомлення після вилучення, виправлено код для коректного оброблення двійкових масивів, а також забезпечено перевірку наявності достатнього місця у полі перед приховання повідомлення. Після цих змін повідомлення «Hello» успішно відновилося навіть за умови наявності шуму в полі (рис. 4.4). Контрольна сума гарантувала правильність даних.

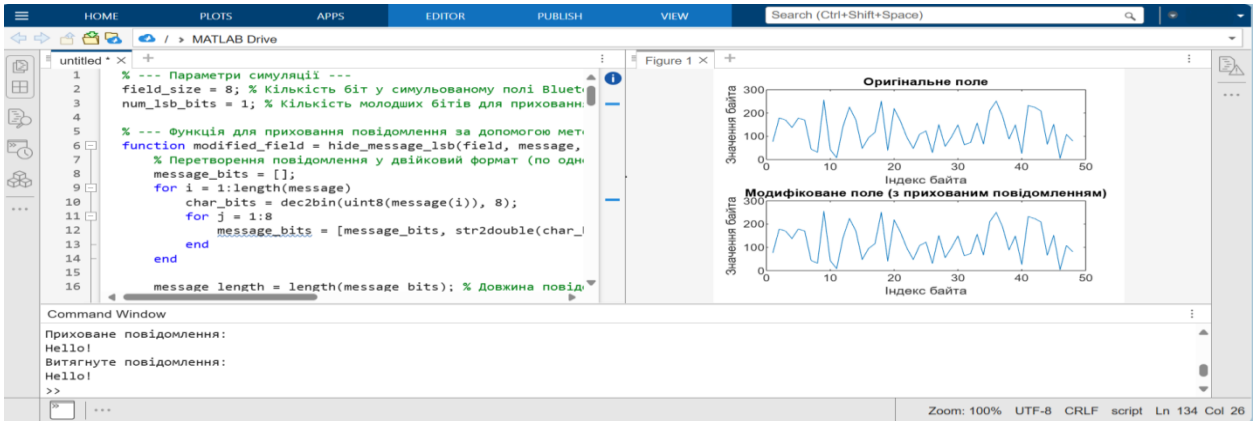

Рисунок 4.4 – Лістинг коду (додаток Б)

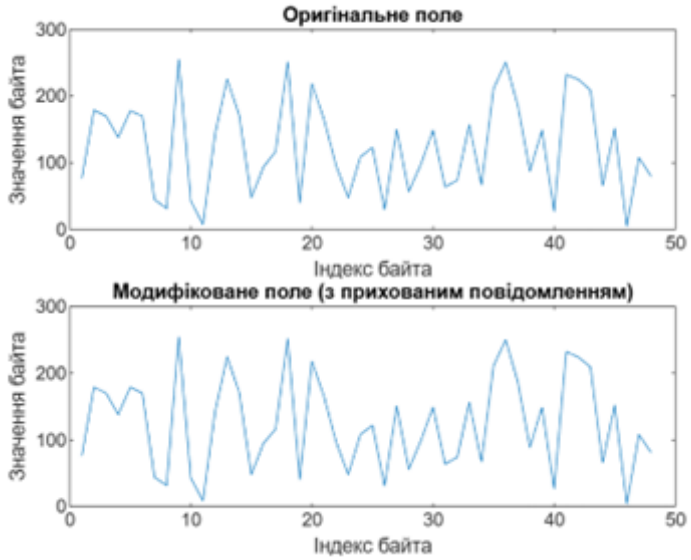

Рисунок 4.5 – Графік отриманого коду

На верхньому графіку рисунку 4.5 («Оригінальне поле») видно вихідні значення байтів, які не містять прихованої інформації – початкова версія даних без будь-яких змін. На нижньому графіку («Модифіковане поле з прихованим повідомленням») спостерігаються незначні відхилення у значеннях байтів, що свідчить про модифікацію даних для приховання тексту за допомогою методу LSB. Різниця між графіками мінімальні – практично невидимими для ока або інших базових способів аналізу. Це демонструє, що

метод LSB є ефективним для приховання даних, оскільки навіть при модифікації молодших бітів загальна структура значень залишається практично незмінною.

Завдяки графікам ми бачимо, як модифікація конкретних байтів може змінювати їх значення в обмеженому діапазоні. Ці зміни залишаються контрольованими, тому метод не порушує цілісності поля (наприклад, Bluetooth-пакету).

Далі, для перевірки правильності шифрування, ми провели шифрування складнішого повідомлення. Було використано вдосконалений код для приховування довшого речення: «Steganographic method of transmitting in formation using Bluetooth technology» (додаток В). Повідомлення було успішно зашифроване у Bluetooth-поле, а потім відновлене без втрат, навіть за умови моделювання шумів (рис. 4.6, 4.7).

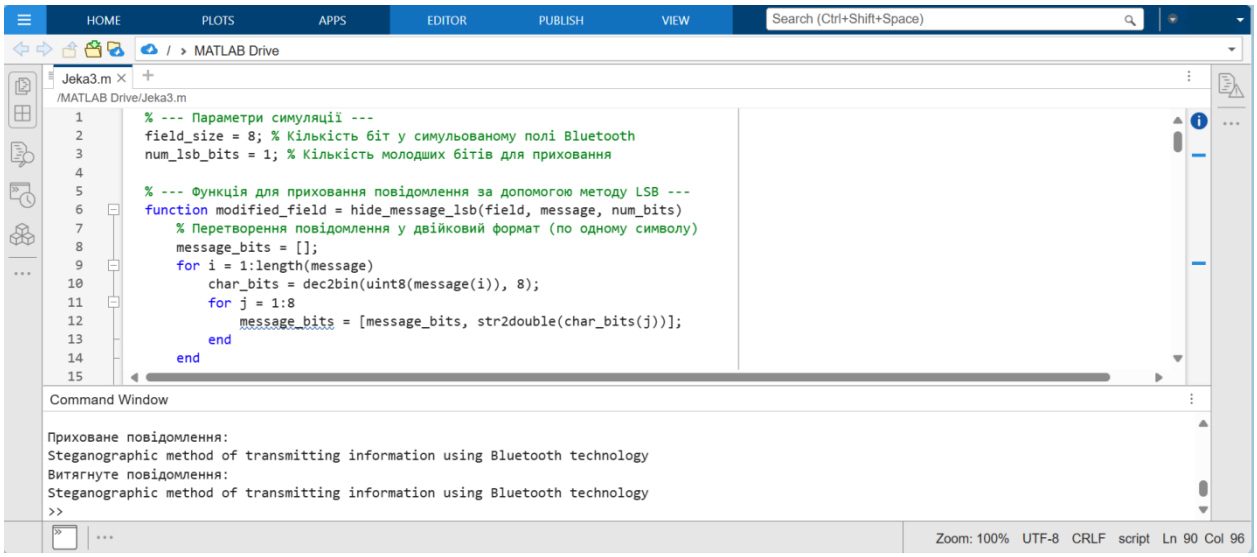

Рисунок 4.6 – Лістинг коду (додаток В)

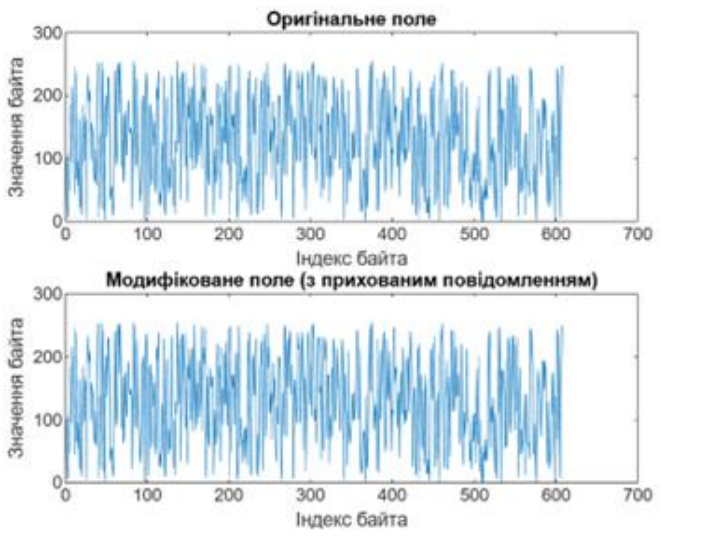

Рисунок 4.7 – Графік отриманого коду

Задля перевірки надійності й ефективності методу LSB частиною експерименту було приховування тексту в картинці. Ми змогли впевнитися, що дані правильно кодуються і декодуються навіть після внесення шуму.

Експеримент полягав у використанні методу LSB для приховування тексту в зображенні. Ми інтегрували текст «Steganographic method of transmitting information using Bluetooth technology», у червоний канал зображення. Візуальні зміни були непомітними, а текст успішно витягнуто після декодування (рис. 4.8, 4.9).

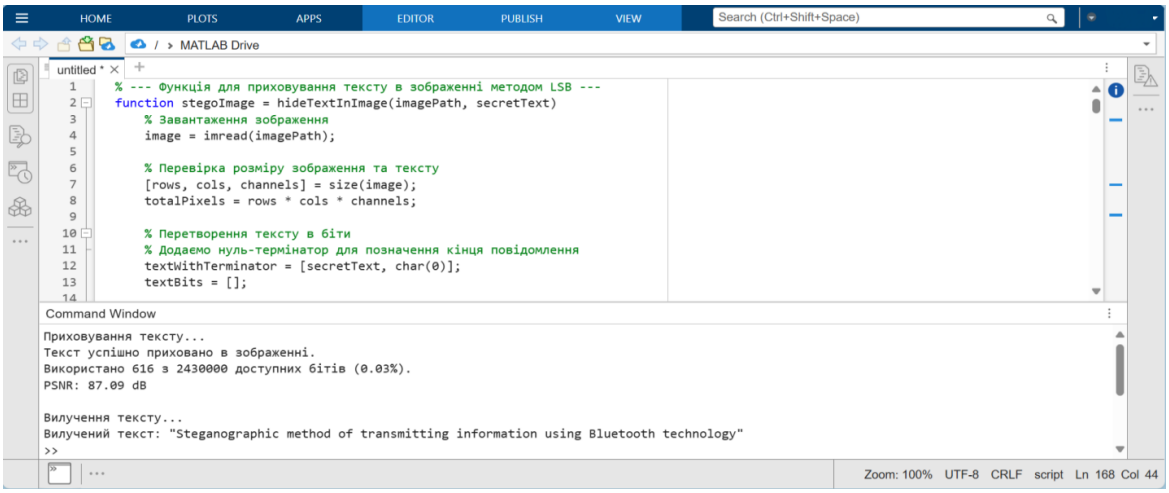

Рисунок 4.8 – Лістинг коду (додаток Г)

Рисунок 4.9 – Вихідне та декодоване зображення

4.4 Аналіз результатів моделювання

За результатами проведених експериментів було здійснено детальний аналіз отриманих даних.

Під час тестування методу LSB для приховання інформації підтверджено його ефективність. Метод дозволяє приховувати текст у числових полях, зокрема у Bluetooth-пакетах, а також у зображеннях. Важливим аспектом стало те, що приховане повідомлення було успішно відновлено без втрат після внесення контрольної суми, навіть за умов наявності шуму.

Аналіз впливу шуму показав, що зміни в байтах після шифрування повідомлення є незначними, що підтверджується графіком «Модифіковане поле», де видно, що значення байтів було трохи змінено для приховання тексту. Водночас шум може викликати спотворення повідомлення, якщо не враховувати помилки в каналі передачі, як це спостерігається у Bluetooth-з’єднаннях. Впровадження контрольної суми стало вирішальним фактором для успішного вилучення прихованої інформації.

Окремо було досліджено можливість роботи з довшими повідомленнями. Успішно зашифровано довге повідомлення, зокрема фразу «Steganographic method of transmitting information using Bluetooth technology», що підтвердило масштабованість методу. Водночас приховування довгого

тексту потребує більшого розміру поля, такого як зображення або пакет, для коректного кодування даних.

На заключному етапі було протестовано приховання тексту у зображеннях. Текст успішно приховано у червоному каналі зображення, при цьому зміни залишилися візуально непомітними, що підтвердило ефективність стеганографічного підходу. Успішне вилучення тексту свідчило, що зображення можуть використовуватися як ще один захищений спосіб передачі інформації.

Висновки до четвертого розділу

У ході дослідження встановлено, що використання стеганографічного підходу на основі найменш значущого біта є дієвим засобом для прихованого передавання інформації. Ключовим чинником підвищення надійності передачі є впровадження механізму контрольної суми, що забезпечує можливість точного відновлення закодованих даних навіть у разі наявності перешкод у каналі зв’язку.

Отримані результати демонструють доцільність застосування LSB-методу як у цифрових пакетах бездротових мереж (зокрема, Bluetooth), так і в мультимедійному контенті, включаючи зображення. Надійність запропонованого алгоритму підтверджено експериментально: завдяки використанню контрольної суми та оптимізованих процедур обробки вдалося досягти високої точності відновлення інформації в умовах зашумленого середовища, що свідчить про перспективність запропонованого підходу для задач захисту даних шляхом прихованої передачі без суттєвого втручання в структуру носія.

ВИСНОВКИ

У ході виконання кваліфікаційної бакалаврської роботи було здійснено всебічне дослідження стеганографічного методу приховання та передачі інформації з використанням Bluetooth-технології. У результаті:

а) проаналізовано принципи побудови бездротових самоорганізованих мереж і визначено їхню придатність до реалізації прихованих каналів зв’язку;

б) досліджено теоретичні основи стеганографії, її класифікацію, принципи та основні вимоги до стеганосистем;

в) обґрунтовано вибір методу найменш значущого біта для вбудовування даних у цифрові пакети та мультимедійні контейнери;

г) розроблено алгоритм прихованої передачі інформації через Bluetooth із використанням LSB-модифікації та контрольної суми;

д) здійснено моделювання процесу передачі в середовищі Matlab, оцінено вплив шуму в каналі зв’язку та визначено ефективність відновлення даних;

е) результати експериментів підтвердили надійність і точність алгоритму навіть в умовах нестабільності каналу, що дозволяє рекомендувати запропонований підхід для практичного застосування у сфері інформаційної безпеки.

Таким чином, поставлену мету досягнуто, а всі задачі – реалізовано. Результати роботи мають як наукове, так і прикладне значення, зокрема у контексті створення захищених бездротових систем обміну інформацією.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. Шумаков В.Ю. Комп’ютерна система виявлення загроз в мережі VANET з використанням машинного навчання кваліфікаційна робота магістра спеціальності 121 «Інженерія програмного забезпечення» / наук. керівник Н.П. Полякова. – Запоріжжя: ЗНУ, 2023. – 99 с.

2. Пиріг Ю.В. Моделі та алгоритми маршрутизації інформаційних потоків в самоорганізованих мережах. На правах рукопису. Дисертація на здобуття наукового ступеня кандидата технічних наук (доктора філософії) за спеціальністю 05.12.02 «Телекомунікаційні системи та мережі» (172 – Телекомунікації та радіотехніка). Національний університет «Львівська політехніка» Міністерства освіти і науки України. – Львів, 2018.

3. Breed G. Wireless adhoc networks: basic concepts. High frequency electronics. – 2007. – V. 1. – Pp. 44-47.

4. Benyamina D. Wireless mesh networks design – A survey. IEEE Communications surveys & tutorials / D. Benyamina, A. Hafid, M. Gendreau. – 2011. – V. 14(2). – Pp. 299-310.

5. Aliu O.G. Survey of Self Organisation in Future Cellular Networks. IEEE Communications Surveys & Tutorials / O.G. Aliu, A. Imran, M.A. Imran, B.A. Evans. – 2013. – V. 15(1). – Pp. 336-361.

6. Prehofer C. Self-organization in communication networks: principles and design paradigms. IEEE Communications magazine / C. Prehofer, C. Bettstetter. – 2005. – V. 43(7). – Pp. 78-85.

7. Marchetti N. (2010, June). Self-organizing networks: State-of-the-art, challenges and perspectives. In 2010 8th International Conference on Communications / N. Marchetti, N.R. Prasad, J. Johansson, T. Cai. – IEEE. – pp. 503-508.

8. Jung J. On self-configuring IoT with dual radios: A cross-layer approach. IEEE Transactions on Mobile Computing / J. Jung, J. Hong, Y. Yi. – 2021. – V. 21(11). – Pp. 4064-4077.

9. Bayazeed A. A survey of self-coordination in self-organizing network. Computer networks / A. Bayazeed, K. Khorzom, M. Aljnidi. – 2021. – V. 196. – P. 108-222.

10. Thangaraj S.J.J. (2023, April). Data-driven ML Approaches for the concept of Self-healing in CWN, Including its Challenges and Possible Solutions. In 2023 Eighth International Conference on Science Technology Engineering and Mathematics (ICONSTEM) (pp. 1-7). IEEE.

11. Saad W.K. Handover and load balancing self-optimization models in 5G mobile networks. // Engineering Science and Technology, an International Journal / W.K. Saad, I. Shayea, A. Alhammadi, M.M. Sheikh, A.A. El-Saleh. – 2023. – V. 42. – P. 101-418.

12. Каптур В.А. Механізм керування розподіленими ресурсами в ситуативних мережах. Наукові праці ОНАЗ ім. О.С. Попова / В.А. Каптур, О.В. Степаненко. – 2010. – № 1. – С. 31-37.

13. Карпов М.С. Аналіз бездротових сенсорних мереж. Автоматизація та Приладобудування («Automation and Development of Electronic Devices» ADED-2023) [Електронний ресурс] / М.С. Карпов // Збірник студентських наукових статей Харківський національний університет радіоелектроніки; [редкол.: І.Ш. Невлюдов та ін.]. – Харків: ХНУРЕ, 2023. – Вип. 1. – С. 270-276.

14. Довженко Н. Інтеграція ІОТ та штучного інтелекту в інтелектуальні транспортні системи. Електронне фахове наукове видання «Кібербезпека: освіта, наука, техніка» / Н. Довженко, Н. Мазур, Ю. Костюк, С. Рзаєва. – 2024. – Вип. 2(26). – С. 430-444.

15. Ткачов Б.В. Аналіз принципів побудови і функціонування без проводових інфокомунікаційних мереж із самоорганізацією : дипломна робота на здобуття ступеня бакалавра / Ткачов Б.В.; наук. керівник

Валуйський С. В.; Нац. техн. ун-т України «Київ. політехн. ін-т» ім. І.Сікорського. Київ, 2024. 60 с.Конституція України: Закон України від 28 червня 1996 р. № 254к/96-ВР [Електронний ресурс]. – Режим доступу: https://urst.com.ua/konstytucija/st-17 (дата звернення: 26.05.2025) – Назва з екрану.

16. Закон України «Про захист інформації в інформаційно-телекомунікаційних системах» від 5 липня 1994 р. № 80/94-ВР [Електронний ресурс]. – Режим доступу: https://urst.com.ua/pro_zakhyst_informatsii_v_informatsiino-telekomunikatsiinykh_sys/st-1 (дата звернення: 26.05.2025) – Назва з екрану.

17. Джулій В.М. Аналіз методів та засобів прихованої передачі інформації. Вимірювальна та обчислювальна техніка в технологічних процесах / В.М. Джулій, Є.О. Коврига. – 2014. – Вип. 3. – С. 179-183.

18. Музика Д.С. Огляд технології безпровідної передачі інформації Bluetooth. XIII Науково-практична конференція студентів, аспірантів та молодих вчених «Погляд у майбутнє приладобудування», 13-14 травня 2020 р., м. Київ, Україна: збірник праць конференції / Д.С. Музика, Д.О. Півторак. – Київ: КПІ ім. Ігоря Сікорського, 2020. – С. 52-54.

19. Молоковський І.О. Аналіз технологій бездротового зв’язку у технологічних мережах промислових підприємств. Збірник наукових праць ДонІЗТ / І.О. Молоковський, В.В. Турупалов, Л.О. Шебанова. – 2011. – №28. – С. 88-93.

20. Comparing Bluetooth Versions. Minew Technologies. [Електронний ресурс]. – Режим доступу: https://www.minew.com/ur/comparing-bluetooth-versions/ (дата звернення: 25 квітня 2025 р.) – Назва з екрану.

21. Bluetooth® Core Specification Version 6.0. Bluetooth Technology Website. [Електронний ресурс]. – Режим доступу: https://www.bluetooth.com/specifications/specs/core-specification-6-0 (дата звернення: 25 квітня 2025 р.) – Назва з екрану.

22. Delivering on the bi-annual release schedule: Bluetooth® Core 6.1 is here. Bluetooth Technology Website. [Електронний ресурс]. – Режим доступу: https://www.bluetooth.com/blog/delivering-on-the-bi-annual-release-schedule-bluetooth-core-6-1-is-here/ (дата звернення: 25 квітня 2025 р.) – Назва з екрану.

23. Грудзинський Ю.Є. Побудова сенсорної mesh-мережі промислового інтернету речейна основі технології Bluetooth 4.2. Вісник Національного технічного університету ХПІ. Серія: Нові рішення в сучасних технологіях / Ю.Є. Грудзинський, Я.Ю. Лукомський. – 2018. – Вип. 16. – С. 107-111.

24. Bluetooth SIG. Bluetooth Core Specification. Part B: Service Discovery Protocol (SDP) Specification. [Електронний ресурс] Режим доступу: https://www.bluetooth.com/wp-content/uploads/Files/Specification/HTML/Core-54/out/en/host/service-discovery-protocol--sdp--specification.html (дата звернення: 20 квітня 2025 р.) – Назва з екрану.

25. Bluetooth SDP inthehand/32feet Wiki - GitHub. [Електронний ресурс]. Режим доступу: https://github.com/inthehand/32feet/wiki/Bluetooth-SDP (дата звернення: 24 квітня 2025 р.).

26. Bluetooth SIG. Bluetooth Core Specification. Part A: Logical Link Control and Adaptation Protocol Specification. [Електронний ресурс]. Режим доступу: https://www.bluetooth.com/wp-content/uploads/Files/Specification/HTML/Core-54/out/en/host/logical-link-control-and-adaptation-protocol-specification.html (дата звернення: 25 квітня 2025 р.) – Назва з екрану.

27. Novel Bits. How Bluetooth Low Energy Works: Advertisements (Part 1). [Електронний ресурс]. Режим доступу: https://novelbits.io/bluetooth-low-energy-advertisements-part-1/ (дата звернення: 23 квітня 2025 р.) – Назва з екрану.

28. Asaf K. A Review of Bluetooth based Scatternet for Mobile Ad hoc Networks. International Journal of Advanced Computer Science and Applications / K. Asaf, M.U. Sarwar, M.K. Hanif, R. Talib, I. Khan. – 2017. – V. 8(6).

29. Mazurczyk W. Steganography in modern smartphones and mitigation techniques. IEEE Communications Surveys & Tutorials / W. Mazurczyk, L. Caviglione. – 2014. – V. 17(1). – Pp. 334-357.

ДОДАТОК А

```
% --- Параметри симуляції ---
field_size = 8; % Кількість біт у симульованому полі Bluetooth
num_lsb_bits = 1; % Кількість молодших бітів для приховання

% --- Функція для приховання повідомлення за допомогою методу LSB ---
function modified_field = hide_message_lsb(field, message, num_bits)
    binary_message = dec2bin(uint8(message), 8)'; % Перетворення повідомлення у двійковий формат
    message_length = length(binary_message); % Довжина повідомлення у бітах

    % Перевірка, чи достатньо місця для приховання повідомлення
    available_bits = length(field) * num_bits; % Доступні біти у полі
    required_bits = message_length; % Необхідна кількість бітів для повідомлення

    if required_bits > available_bits
        error('Недостатньо місця для приховання повідомлення.');
    end

    modified_field = field; % Копія поля для модифікації
    message_index = 1; % Індекс для обходу повідомлення

    for i = 1:length(field)
        for j = 1:num_bits
            if message_index <= message_length
                % Отримання біта з повідомлення
                message_bit = str2double(binary_message(message_index));
```

```
            % Отримання бінарного представлення байта поля
            binary_field = dec2bin(field(i), 8);

            % Заміна молодшого біта
            bit_index = 8 - j + 1; % Індекс молодшого біта
            new_binary_field = binary_field;
            new_binary_field(bit_index) = num2str(message_bit);

            % Перетворення назад у десяткове число
            modified_field(i) = bin2dec(new_binary_field);

            message_index = message_index + 1; % Перехід до наступного біта
        else
            % Завершення, якщо повідомлення повністю приховано
            break;
        end
    end
    if message_index > message_length
        break;
    end
  end
end

% --- Функція для вилучення повідомлення за допомогою методу LSB ---
function extracted_message = extract_message_lsb(modified_field,
original_message_length, num_bits)
  extracted_bits = ''; % Змінна для збереження витягнутих бітів
  num_characters = original_message_length; % Кількість символів у
повідомленні
  total_bits = num_characters * 8; % Загальна кількість бітів для витягнення
```

```
    bit_index = 1; % Індекс біта

    for i = 1:length(modified_field)
        binary_field = dec2bin(modified_field(i), 8); % Отримання бінарного
представлення байта
        for j = 1:num_bits
            if bit_index <= total_bits
                lsb_index = 8 - j + 1; % Індекс молодшого біта
                extracted_bits = [extracted_bits binary_field(lsb_index)];
                bit_index = bit_index + 1;
            else
                break;
            end
        end
        if bit_index > total_bits
            break;
        end
    end

    extracted_message = ''; % Розшифроване повідомлення
    for i = 1:8:length(extracted_bits)
        byte = extracted_bits(i:min(i+7, length(extracted_bits)));
        extracted_message = [extracted_message char(bin2dec(byte))];
    end
end

% --- Симуляція Bluetooth-поля ---
original_field = randi([0, 255], 1, 20); % Генерація випадкового поля (20 байт)

% --- Повідомлення для приховання ---
```

```
message_to_hide = 'Hello!'; % Текстове повідомлення
original_message_length = length(message_to_hide); % Довжина повідомлення

% --- Приховання повідомлення ---
modified_field = hide_message_lsb(original_field, message_to_hide,
num_lsb_bits);

% --- Витягнення повідомлення ---
extracted_message = extract_message_lsb(modified_field,
original_message_length, num_lsb_bits);

% --- Виведення результатів ---
disp('Оригінальне поле:');
disp(original_field);
disp('Модифіковане поле (з прихованим повідомленням):');
disp(modified_field);
disp('Приховане повідомлення:');
disp(message_to_hide);
disp('Витягнуте повідомлення:');
disp(extracted_message);

% --- Додаткова візуалізація (порівняння оригінального та модифікованого
полів) ---
figure;
subplot(2,1,1);
plot(original_field);
title('Оригінальне поле');
xlabel('Індекс байта');
ylabel('Значення байта');
```

```
subplot(2,1,2);
plot(modified_field);
title('Модифіковане поле (з прихованим повідомленням)');
xlabel('Індекс байта');
ylabel('Значення байта');
```

ДОДАТОК Б

```
% --- Параметри симуляції ---
field_size = 8; % Кількість біт у симульованому полі Bluetooth
num_lsb_bits = 1; % Кількість молодших бітів для приховання

% --- Функція для приховання повідомлення за допомогою методу LSB ---
function modified_field = hide_message_lsb(field, message, num_bits)
    % Перетворення повідомлення у двійковий формат (по одному символу)
    message_bits = [];
    for i = 1:length(message)
        char_bits = dec2bin(uint8(message(i)), 8);
        for j = 1:8
            message_bits = [message_bits, str2double(char_bits(j))];
        end
    end

    message_length = length(message_bits); % Довжина повідомлення у бітах

    % Перевірка, чи достатньо місця для приховання повідомлення
    available_bits = length(field) * num_bits; % Доступні біти у полі

    if message_length > available_bits
        error('Недостатньо місця для приховання повідомлення. Потрібно %d
бітів, доступно %d бітів.', message_length, available_bits);
    end

    modified_field = field; % Копія поля для модифікації
    message_index = 1; % Індекс для обходу повідомлення
```

```
    for i = 1:length(field)
        binary_field = dec2bin(field(i), 8);
        new_binary_field = binary_field;

        for j = 1:num_bits
            if message_index <= message_length
                % Заміна молодшого біта
                bit_index = 9 - j; % Правильний індекс для LSB (8 для найменш значущого біта)
                new_binary_field(bit_index) = num2str(message_bits(message_index));
                message_index = message_index + 1;
            else
                break;
            end
        end

        % Перетворення назад у десяткове число
        modified_field(i) = bin2dec(new_binary_field);

        if message_index > message_length
            break;
        end
    end
end

% --- Функція для вилучення повідомлення за допомогою методу LSB ---
function extracted_message = extract_message_lsb(modified_field, original_message_length, num_bits)
    % Визначення загальної кількості бітів для вилучення
    total_bits = original_message_length * 8;
```

```
    extracted_bits = zeros(1, total_bits);
    bit_count = 0;

    for i = 1:length(modified_field)
        if bit_count >= total_bits
            break;
        end

        binary_field = dec2bin(modified_field(i), 8);

        for j = 1:num_bits
            if bit_count < total_bits
                bit_index = 9 - j; % Правильний індекс для LSB
                extracted_bits(bit_count + 1) = str2double(binary_field(bit_index));
                bit_count = bit_count + 1;
            else
                break;
            end
        end
    end

    % Перетворення бітів назад у символи
    extracted_message = '';
    for i = 1:8:total_bits
        byte_bits = extracted_bits(i:min(i+7, total_bits));
        byte_str = char('0' + byte_bits); % Перетворення чисел у символи '0' та '1'
        extracted_message = [extracted_message, char(bin2dec(byte_str))];
    end
end
```

```
% --- Симуляція Bluetooth-поля ---
original_field = randi([0, 255], 1, 20); % Генерація випадкового поля (20 байт)

% --- Повідомлення для приховання ---
message_to_hide = 'Hello!'; % Текстове повідомлення
original_message_length = length(message_to_hide); % Довжина повідомлення

% --- Перевірка чи достатньо місця перед викликом функції ---
message_bits_length = original_message_length * 8;
available_bits = length(original_field) * num_lsb_bits;
if message_bits_length > available_bits
    fprintf('Попередження: Недостатньо місця для приховання
повідомлення.\n');
    fprintf('Потрібно %d бітів, доступно %d бітів.\n', message_bits_length,
available_bits);
    fprintf('Збільшіть розмір поля або кількість LSB бітів.\n');
    % Збільшуємо поле до необхідного розміру
    additional_bytes_needed = ceil((message_bits_length - available_bits) /
num_lsb_bits);
    original_field = [original_field, randi([0, 255], 1, additional_bytes_needed)];
    fprintf('Поле автоматично збільшено до %d байтів.\n', length(original_field));
end

% --- Приховання повідомлення ---
modified_field = hide_message_lsb(original_field, message_to_hide,
num_lsb_bits);

% --- Витягнення повідомлення ---
extracted_message = extract_message_lsb(modified_field,
original_message_length, num_lsb_bits);
```

```
% --- Виведення результатів ---
disp('Оригінальне поле:');
disp(original_field);
disp('Модифіковане поле (з прихованим повідомленням):');
disp(modified_field);
disp('Приховане повідомлення:');
disp(message_to_hide);
disp('Витягнуте повідомлення:');
disp(extracted_message);

% --- Додаткова візуалізація (порівняння оригінального та модифікованого полів) ---
figure;
subplot(2,1,1);
plot(original_field);
title('Оригінальне поле');
xlabel('Індекс байта');
ylabel('Значення байта');

subplot(2,1,2);
plot(modified_field);
title('Модифіковане поле (з прихованим повідомленням)');
xlabel('Індекс байта');
ylabel('Значення байта');
```

ДОДАТОК В

```
% --- Параметри симуляції ---
field_size = 8; % Кількість біт у симульованому полі Bluetooth
num_lsb_bits = 1; % Кількість молодших бітів для приховання

% --- Функція для приховання повідомлення за допомогою методу LSB ---
function modified_field = hide_message_lsb(field, message, num_bits)
    % Перетворення повідомлення у двійковий формат (по одному символу)
    message_bits = [];
    for i = 1:length(message)
        char_bits = dec2bin(uint8(message(i)), 8);
        for j = 1:8
            message_bits = [message_bits, str2double(char_bits(j))];
        end
    end

    message_length = length(message_bits); % Довжина повідомлення у бітах

    % Перевірка, чи достатньо місця для приховання повідомлення
    available_bits = length(field) * num_bits; % Доступні біти у полі

    if message_length > available_bits
        error('Недостатньо місця для приховання повідомлення. Потрібно %d
бітів, доступно %d бітів.', message_length, available_bits);
    end

    modified_field = field; % Копія поля для модифікації
    message_index = 1; % Індекс для обходу повідомлення
```

```
    for i = 1:length(field)
        binary_field = dec2bin(field(i), 8);
        new_binary_field = binary_field;

        for j = 1:num_bits
            if message_index <= message_length
                % Заміна молодшого біта
                bit_index = 9 - j; % Правильний індекс для LSB (8 для найменш значущого біта)
                new_binary_field(bit_index) = num2str(message_bits(message_index));
                message_index = message_index + 1;
            else
                break;
            end
        end

        % Перетворення назад у десяткове число
        modified_field(i) = bin2dec(new_binary_field);

        if message_index > message_length
            break;
        end
    end
end

% --- Функція для вилучення повідомлення за допомогою методу LSB ---
function extracted_message = extract_message_lsb(modified_field, original_message_length, num_bits)
    % Визначення загальної кількості бітів для вилучення
    total_bits = original_message_length * 8;
```

```
    extracted_bits = zeros(1, total_bits);
    bit_count = 0;

    for i = 1:length(modified_field)
        if bit_count >= total_bits
            break;
        end

        binary_field = dec2bin(modified_field(i), 8);

        for j = 1:num_bits
            if bit_count < total_bits
                bit_index = 9 - j; % Правильний індекс для LSB
                extracted_bits(bit_count + 1) = str2double(binary_field(bit_index));
                bit_count = bit_count + 1;
            else
                break;
            end
        end
    end

    % Перетворення бітів назад у символи
    extracted_message = '';
    for i = 1:8:total_bits
        byte_bits = extracted_bits(i:min(i+7, total_bits));
        byte_str = char('0' + byte_bits); % Перетворення чисел у символи '0' та '1'
        extracted_message = [extracted_message, char(bin2dec(byte_str))];
    end
end
```

```
% --- Симуляція Bluetooth-поля ---
original_field = randi([0, 255], 1, 20); % Генерація випадкового поля (20 байт)

% --- Повідомлення для приховання ---
message_to_hide = 'Steganographic method of transmitting information using
Bluetooth technology'; % Текстове повідомлення
original_message_length = length(message_to_hide); % Довжина повідомлення

% --- Перевірка чи достатньо місця перед викликом функції ---
message_bits_length = original_message_length * 8;
available_bits = length(original_field) * num_lsb_bits;
if message_bits_length > available_bits
    fprintf('Попередження: Недостатньо місця для приховання
повідомлення.\n');
    fprintf('Потрібно %d бітів, доступно %d бітів.\n', message_bits_length,
available_bits);
    fprintf('Збільшіть розмір поля або кількість LSB бітів.\n');
    % Збільшуємо поле до необхідного розміру
    additional_bytes_needed = ceil((message_bits_length - available_bits) /
num_lsb_bits);
    original_field = [original_field, randi([0, 255], 1, additional_bytes_needed)];
    fprintf('Поле автоматично збільшено до %d байтів.\n', length(original_field));
end

% --- Приховання повідомлення ---
modified_field = hide_message_lsb(original_field, message_to_hide,
num_lsb_bits);

% --- Витягнення повідомлення ---
```

```
extracted_message = extract_message_lsb(modified_field,
original_message_length, num_lsb_bits);

% --- Виведення результатів ---
disp('Оригінальне поле:');
disp(original_field);
disp('Модифіковане поле (з прихованим повідомленням):');
disp(modified_field);
disp('Приховане повідомлення:');
disp(message_to_hide);
disp('Витягнуте повідомлення:');
disp(extracted_message);

% --- Додаткова візуалізація (порівняння оригінального та модифікованого
полів) ---
figure;
subplot(2,1,1);
plot(original_field);
title('Оригінальне поле');
xlabel('Індекс байта');
ylabel('Значення байта');

subplot(2,1,2);
plot(modified_field);
title('Модифіковане поле (з прихованим повідомленням)');
xlabel('Індекс байта');
ylabel('Значення байта');
```

ДОДАТОК Г

```
% --- Функція для приховування тексту в зображенні методом LSB ---
function stegoImage = hideTextInImage(imagePath, secretText)
    % Завантаження зображення
    image = imread(imagePath);

    % Перевірка розміру зображення та тексту
    [rows, cols, channels] = size(image);
    totalPixels = rows * cols * channels;

    % Перетворення тексту в біти
    % Додаємо нуль-термінатор для позначення кінця повідомлення
    textWithTerminator = [secretText, char(0)];
    textBits = [];

    for i = 1:length(textWithTerminator)
        charBits = dec2bin(uint8(textWithTerminator(i)), 8);
        for j = 1:8
            textBits = [textBits, str2double(charBits(j))];
        end
    end

    % Перевірка, чи вистачає місця для приховання тексту
    if length(textBits) > totalPixels
        error('Текст занадто великий для цього зображення. Максимальна
кількість бітів: %d', totalPixels);
    end

    % Приховування бітів тексту в найменш значущому біті зображення
```

```
stegoImage = image;
bitIndex = 1;

for i = 1:rows
    for j = 1:cols
        for k = 1:channels
            if bitIndex <= length(textBits)
                % Встановлення LSB пікселя згідно з бітом тексту
                if mod(stegoImage(i, j, k), 2) ~= textBits(bitIndex)
                    % Якщо LSB не співпадає з потрібним бітом, змінюємо його
                    if mod(stegoImage(i, j, k), 2) == 0
                        stegoImage(i, j, k) = stegoImage(i, j, k) + 1;
                    else
                        stegoImage(i, j, k) = stegoImage(i, j, k) - 1;
                    end
                end
                bitIndex = bitIndex + 1;
            else
                % Весь текст вже приховано
                break;
            end
        end
        if bitIndex > length(textBits)
            break;
        end
    end
    if bitIndex > length(textBits)
        break;
    end
end
```

```
    % Інформування про успіх
    fprintf('Текст успішно приховано в зображенні.\n');
    fprintf('Використано %d з %d доступних бітів (%0.2f%%).\n', ...
        length(textBits), totalPixels, (length(textBits)/totalPixels)*100);
end

% --- Функція для вилучення тексту з зображення з LSB приховуванням ---
function extractedText = extractTextFromImage(stegoImagePath)
    % Завантаження зображення
    stegoImage = imread(stegoImagePath);

    [rows, cols, channels] = size(stegoImage);

    % Вилучення бітів
    extractedBits = [];
    textFound = false;

    for i = 1:rows
        for j = 1:cols
            for k = 1:channels
                % Отримання LSB пікселя
                bit = mod(stegoImage(i, j, k), 2);
                extractedBits = [extractedBits, bit];

                % Перевірка кожні 8 бітів, чи знайдено нуль-термінатор
                if length(extractedBits) >= 8 && mod(length(extractedBits), 8) == 0
                    % Витягуємо останній символ
                    lastCharBits = extractedBits(end-7:end);
                    lastCharStr = char('0' + lastCharBits);
```

```
                lastChar = char(bin2dec(lastCharStr));

                if lastChar == char(0)
                    textFound = true;
                    break;
                end
            end
        end
        if textFound
            break;
        end
    end
    if textFound
        break;
    end
  end

  % Перетворення бітів назад у текст
  extractedText = '';
  for i = 1:8:length(extractedBits)-8  % -8 щоб не включати нуль-термінатор
    charBits = extractedBits(i:i+7);
    charStr = char('0' + charBits);
    extractedText = [extractedText, char(bin2dec(charStr))];
  end
end

% --- Приклад використання ---

% --- Приховування тексту ---
function runHideText(imagePath, outputPath, secretText)
```

```
    % Завантаження зображення
    originalImage = imread(imagePath);

    % Приховування тексту
    stegoImage = hideTextInImage(imagePath, secretText);

    % Збереження зображення з прихованим текстом
    imwrite(stegoImage, outputPath);

    % Відображення зображень (оригінального та зі стеганографією)
    figure;
    subplot(1, 2, 1);
    imshow(originalImage);
    title('Оригінальне зображення');

    subplot(1, 2, 2);
    imshow(stegoImage);
    title('Стеганографічне зображення');

    % Обчислення PSNR (пікове співвідношення сигнал-шум)
    mse = immse(double(originalImage), double(stegoImage));
    if mse > 0
        psnr_val = 10 * log10(255^2 / mse);
        fprintf('PSNR: %.2f dB\n', psnr_val);
    else
        fprintf('PSNR: Infinite (зображення ідентичні)\n');
    end
end

% --- Вилучення тексту ---
```

```
function extractedText = runExtractText(stegoImagePath)
    % Вилучення тексту
    extractedText = extractTextFromImage(stegoImagePath);

    % Відображення результату
    fprintf('Вилучений текст: "%s"\n', extractedText);
end

% --- Головний скрипт ---
% Замініть шляхи та текст вашими даними
imagePath =
'https://miro.medium.com/v2/resize:fit:1200/1*IEkm7DiGA7mcla326dH94Q.jpeg'
;          % Шлях до оригінального зображення
outputPath = 'stego_image.png';   % Шлях для збереження стеганографічного
зображення
secretText = 'Steganographic method of transmitting information using Bluetooth
technology';  % Текст для приховування

% Приховування тексту
fprintf('Приховування тексту...\n');
runHideText(imagePath, outputPath, secretText);

% Вилучення тексту
fprintf('\nВилучення тексту...\n');
extractedText = runExtractText(outputPath);
```

```
% --- Bluetooth Steganography System with Real Bluetooth Communication ---

% --- Функції для роботи з приховананням даних ---
function modified_field = hideMessageInData(data_field, secret_message,
num_lsb_bits)
    % Конвертація повідомлення в масив бітів
    message_bits = [];
    for i = 1:length(secret_message)
        char_bits = dec2bin(uint8(secret_message(i)), 8);
        for j = 1:8
            message_bits = [message_bits, str2double(char_bits(j))];
        end
    end

    % Додавання довжини повідомлення (перші 16 біт)
    msg_length_bits = dec2bin(length(secret_message), 16);
    length_bits = [];
    for i = 1:16
        length_bits = [length_bits, str2double(msg_length_bits(i))];
    end

    all_bits = [length_bits, message_bits];
    total_bits = length(all_bits);

    % Перевірка чи достатньо місця
    available_bits = length(data_field) * num_lsb_bits;
    if total_bits > available_bits
        error('Повідомлення занадто велике для приховання. Потрібно %d бітів,
доступно %d бітів.', ...
              total_bits, available_bits);
```

```
    end

    % Копіювання вхідних даних
    modified_field = data_field;

    % Приховання бітів
    bit_index = 1;
    for i = 1:length(data_field)
        binary_value = dec2bin(data_field(i), 8);
        modified_binary = binary_value;

        for j = 1:num_lsb_bits
            if bit_index <= total_bits
                % Заміна LSB біту
                lsb_position = 9 - j; % 8 для найменш значущого біта
                modified_binary(lsb_position) = num2str(all_bits(bit_index));
                bit_index = bit_index + 1;
            else
                break;
            end
        end

        modified_field(i) = bin2dec(modified_binary);

        if bit_index > total_bits
            break;
        end
    end
end
```

```
function secret_message = extractMessageFromData(received_data, num_lsb_bits)
    % Вилучення бітів довжини повідомлення (перші 16 біт)
    length_bits = zeros(1, 16);
    bit_index = 1;

    for i = 1:length(received_data)
        binary_value = dec2bin(received_data(i), 8);

        for j = 1:num_lsb_bits
            if bit_index <= 16
                lsb_position = 9 - j;
                length_bits(bit_index) = str2double(binary_value(lsb_position));
                bit_index = bit_index + 1;
            else
                break;
            end
        end

        if bit_index > 16
            break;
        end
    end

    % Конвертація бітів довжини в число
    length_str = char('0' + length_bits);
    message_length = bin2dec(length_str);

    % Вилучення бітів повідомлення
    total_message_bits = message_length * 8;
    message_bits = zeros(1, total_message_bits);
```

```
bit_index = 1;

% Продовжуємо з того місця, де зупинились при читанні довжини
data_index = ceil(16 / num_lsb_bits) + 1;
bit_offset = 0;

for i = data_index:length(received_data)
    binary_value = dec2bin(received_data(i), 8);

    for j = 1:num_lsb_bits
        if bit_index <= total_message_bits
            lsb_position = 9 - j;
            message_bits(bit_index) = str2double(binary_value(lsb_position));
            bit_index = bit_index + 1;
        else
            break;
        end
    end

    if bit_index > total_message_bits
        break;
    end
end

% Конвертація бітів повідомлення назад у текст
secret_message = '';
for i = 1:8:total_message_bits
    if i + 7 <= total_message_bits
        char_bits = message_bits(i:i+7);
        char_str = char('0' + char_bits);
```

```
            secret_message = [secret_message, char(bin2dec(char_str))];
        end
    end
end

% --- Bluetooth комунікаційні функції ---
function bt = initializeBluetooth()
    % Ініціалізація Bluetooth
    bt = Bluetooth('YourDeviceName', 1);  % Змініть 'YourDeviceName' на ім'я
вашого пристрою
    % Альтернативно можна з'єднатись за адресою
    % bt = Bluetooth('XX:XX:XX:XX:XX:XX', 1);  % Замініть
XX:XX:XX:XX:XX:XX на MAC-адресу

    % Встановлення параметрів з'єднання
    set(bt, 'Timeout', 30);  % Таймаут у секундах
    fprintf('З'єднання з Bluetooth пристроєм...\n');
end

function sendDataOverBluetooth(bt, data)
    % Відправлення даних через Bluetooth
    try
        % Додавання заголовка для розпізнавання стеганографічного пакета
        header = [0xAA, 0x55, 0xAA, 0x55];  % Легко розпізнаваний заголовок

        % Обчислення розміру пакету і додавання його до заголовка
        packet_size = length(data);
        size_bytes = [bitshift(bitand(packet_size, 65280), -8), bitand(packet_size,
255)];
```

```
        % Об'єднання заголовка, розміру та даних
        packet = [uint8(header), uint8(size_bytes), uint8(data)];

        % Відправка даних
        fwrite(bt, packet);
        fprintf('Дані успішно відправлені (%d байт).\n', length(packet));
    catch e
        fprintf('Помилка відправлення даних: %s\n', e.message);
    end
end

function received_data = receiveDataOverBluetooth(bt)
    % Отримання даних через Bluetooth
    try
        fprintf('Очікування даних...\n');

        % Читання заголовка
        header = [];
        while length(header) < 4
            if bt.BytesAvailable > 0
                new_bytes = fread(bt, bt.BytesAvailable);
                header = [header; new_bytes];
            end
            pause(0.1);
        end

        % Перевірка заголовка
        if ~isequal(header(1:4), [170; 85; 170; 85])  % [0xAA; 0x55; 0xAA; 0x55]
            error('Отримано некоректний заголовок пакета');
        end
```

```
        % Читання розміру пакета
        size_bytes = header(5:6);
        packet_size = bitshift(size_bytes(1), 8) + size_bytes(2);

        % Читання даних
        data_received = header(7:end);
        while length(data_received) < packet_size
            if bt.BytesAvailable > 0
                new_data = fread(bt, bt.BytesAvailable);
                data_received = [data_received; new_data];
            end
            pause(0.1);
        end

        % Вилучення тільки необхідних даних
        received_data = data_received(1:packet_size);
        fprintf('Отримано %d байт даних.\n', length(received_data));
    catch e
        fprintf('Помилка отримання даних: %s\n', e.message);
        received_data = [];
    end
end

function closeBluetooth(bt)
    % Закриття Bluetooth з'єднання
    try
        fclose(bt);
        delete(bt);
        clear bt;
```

```
        fprintf('Bluetooth зєднання закрито.\n');
    catch e
        fprintf('Помилка закриття зєднання: %s\n', e.message);
    end
end

% --- Основні функції для стеганографії через Bluetooth ---
function sendSecretMessage(device_name, secret_message, num_lsb_bits)
    try
        % Підготовка звичайних даних для передачі
        % Для прикладу, створюємо "легальні" дані як корисне навантаження
        payload_size = 512;  % Розмір корисного навантаження
        normal_data = randi([0, 255], 1, payload_size);

        % Приховання секретного повідомлення
        fprintf('Приховування секретного повідомлення...\n');
        modified_data = hideMessageInData(normal_data, secret_message,
num_lsb_bits);

        % Ініціалізація Bluetooth
        bt = Bluetooth(device_name, 1);
        fopen(bt);

        % Відправка даних
        fprintf('Відправка даних з прихованим повідомленням...\n');
        sendDataOverBluetooth(bt, modified_data);

        % Закриття з'єднання
        closeBluetooth(bt);
```

```
        fprintf('Повідомлення "%s" успішно приховано і відправлено.\n',
secret_message);
    catch e
        fprintf('Помилка при відправці повідомлення: %s\n', e.message);
    end
end

function receiveSecretMessage(device_name, num_lsb_bits)
    try
        % Ініціалізація Bluetooth
        bt = Bluetooth(device_name, 1);
        fopen(bt);

        % Отримання даних
        fprintf('Очікування даних...\n');
        received_data = receiveDataOverBluetooth(bt);

        if ~isempty(received_data)
            % Вилучення прихованого повідомлення
            secret_message = extractMessageFromData(received_data, num_lsb_bits);
            fprintf('Вилучене повідомлення: "%s"\n', secret_message);
        end

        % Закриття з'єднання
        closeBluetooth(bt);
    catch e
        fprintf('Помилка при отриманні повідомлення: %s\n', e.message);
    end
end
```

```
% --- Демонстраційний код ---
function demo()
    % Параметри
    bluetooth_device = 'YourDeviceName';  % Змініть на ім'я вашого пристрою
    num_lsb_bits = 2;  % Кількість найменш значущих бітів для приховання

    % Вибір режиму
    fprintf('Bluetooth стеганографія\n');
    fprintf('1 - Відправити приховане повідомлення\n');
    fprintf('2 - Отримати приховане повідомлення\n');
    choice = input('Виберіть опцію: ');

    if choice == 1
        % Режим відправки
        secret_message = input('Введіть повідомлення для приховання: ', 's');
        sendSecretMessage(bluetooth_device, secret_message, num_lsb_bits);
    elseif choice == 2
        % Режим отримання
        receiveSecretMessage(bluetooth_device, num_lsb_bits);
    else
        fprintf('Невірний вибір.\n');
    end
end

% Запуск демонстрації
demo();
```