МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ВСП «ОДЕСЬКИЙ ТЕХНІЧНИЙ ФАХОВИЙ КОЛЕДЖ ОНТУ»

Спеціальність: 123 «Комп'ютерна інженерія»

Освітня програма: «Комп'ютерна інженерія»

Група: 2БКС-28

КВАЛІФІКАЦІЙНА РОБОТА БАКАЛАВРА

здобувача освіти денної форми навчання

БКС.28.17.000.КРБ

НЕПОМИЛУЄВОЇ ОЛЬГИ ПЕТРІВНИ

м. Одеса
2024 р.

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

ВСП «ОДЕСЬКИЙ ТЕХНІЧНИЙ ФАХОВИЙ КОЛЕДЖ ОНТУ»

Спеціальність: **123 «Комп'ютерна інженерія»**

Освітньо-професійна програма: **«Комп'ютерна інженерія»**

Група: **2БКС-28**

ПОЯСНЮВАЛЬНА ЗАПИСКА

До кваліфікаційної роботи бакалавра на тему: **«Дослідження методів стеганофонії для аудіо-файлів різних музичних жанрів»**

Проектний матеріал складається з пояснювальної записки на 74 сторінках та графічного (презентаційного) матеріалу на 25 аркушах (слайдах)

Виконавець _____ (Непомилуєва О.П.)

Керівник проекту _____ (Кривченко Ю.В.)

Консультанти:

з розділу охорони праці та техніки безпеки _____ (Чорновол Н.І.)

з нормоконтролю _____ (Петрашова В.І.)

старший консультант _____ (Кривченко Ю.В.)

До захисту допущений

Завідувач кафедри _____ (Іванова Л.В.)

Завідувач відділення _____ (Скорнякова О.В.)

Захист «27» 06 2024 р. Протокол ЕК № 3

Оцінка ЕК 4 (добре) 85

Секретар ЕК _____

АНОТАЦІЯ

У випускній кваліфікаційній роботі розглянуті методи стеганофонії для аудіо-файлів та проаналізовано ефективність їх застосування.

При передачі конфіденційного повідомлення, вбудованого в аудіо-контейнер необхідно мінімізувати спотворення контейнеру задля безпеки цієї передачі. Таким чином, розробка ефективних методів захисту цифрової інформації, зокрема методів комп'ютерної стеганографії та стеганоаналізу, вбудованої в різноманітні контейнери, актуальна та має велике значення.

Розглянуто модель стеганографічної системи та основні її характеристики, способи приховування даних. Проаналізовано структуру стеганографічних систем, зокрема стеганофонічної. Виконано короткий огляд методів стеганофонії, зокрема LSB, кодування парності, кодування фази, розповсюдження спектру, приховування відлуння. Реалізовано порівняння методів стеганофонії та обрано оптимальний. Обґрунтовано необхідність модифікації алгоритму LSB. Складено математичну модель та побудовано блок-схему алгоритму виконання модифікації LSB. Проаналізовано методи визначення спотворення вхідних аудіо-файлів під час атаки.

Описано засоби розробки для реалізації програмного застосунку стеганофонії. Побудовано блок-схему алгоритму, що використовує розглянуті методи, зокрема модифікований LSB. Для перевірки ефективності модифікації алгоритму LSB розроблено програмний застосунок, роботу якого перевірено на музичних аудіо-файлах різних жанрів та різного розміру. Проаналізовано ефективність методів стеганофонії та результати процесу приховування даних, AWGN-атаки, розрахунку показників цілісності.

Описано заходи з охорони праці та техніки безпеки.

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

ВСП «ОДЕСЬКИЙ ТЕХНІЧНИЙ ФАХОВИЙ КОЛЕДЖ ОНТУ»

Відділення *Комп'ютерних систем* Кафедра *Комп'ютерної інженерії*

Спеціальність *123 «Комп'ютерна інженерія»*

Освітньо-професійна програма *«Комп'ютерна інженерія»*

ЗАТВЕРДЖУЮ:

Заст. дир. з НВР *Беркань І.В.*

"15" 04 2024 р.

ЗАВДАННЯ

на кваліфікаційну роботу бакалавра

здобувачеві освіти *Непомилуєвої Ользі Петрівні*

(прізвище, ім'я, по батькові)

1. Тема кваліфікаційної роботи *Дослідження методів стеганофонії для аудіо-файлів різних музичних жанрів*

затверджена наказом по коледжу від "02" листопада 2023 р. № 244-АГ-05

2. Термін здачі студентом кваліфікаційної роботи 13.06.24р.

3. Вихідні дані до роботи *1. Характеристики аудіо-даних різних типів (мови, музики, складного сигналу); 2. Специфікації методу стеганофонії LSB; 3. Специфікації методів стиснення аудіо-даних (OGG, AC3, AAC, WMA, FLAC, APE, ALAC); 4. Забезпечити у алгоритмі стійкість до атак та зменшення спотворення вхідного контейнеру; 5. Провести дослідження ефективності стеганофонічних алгоритмів на аудіо-записах творів різних музичних жанрів*

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

Модель, структура та характеристики стеганографічної системи; Огляд методів стеганофонії та вибір оптимального; Модифікація алгоритму LSB та побудова алгоритмів для реалізації стеганофонічного застосунку; Реалізація програмного застосунку та аналіз ефективності методів стеганофонії; Питання охорони праці та техніки безпеки

5. Перелік графічного матеріалу (слайдів мультимедійної презентації) *Стенографічний алгоритм приховування повідомлення; Структурна схема стегосистеми; Метод LSB для стеганофонії; Алгоритм стиснення інформації; Блок-схема алгоритму приховування даних у аудіо-файлі; Набір вхідних даних; Оцінка спотворення вхідного файлу; Інтерфейс програмного продукту для стеганофонії; Результати процесу приховування даних; Результати додавання АБГШ-атаки; Результати розрахунку показників цілісності; Результати розрахунку показників цілісності для 1-LSB методу*

6. Консультанти по кваліфікаційній роботі, із зазначенням розділів, що їх стосуються

Розділ	Консультант	ПІДПИС	
		Завдання видав	Завдання прийняв
Основний розділ	*Кривченко Ю.В.*		
Розділ охорони праці	*Чорновол Н.І.*		
Нормоконтроль	*Петрашова В.І.*		
Старший консультант	*Кривченко Ю.В.*		

7. Дата видачі завдання 15.01.24

Керівник роботи *Кривченко Ю.В.* _____
(підпис)

Завдання прийняв до виконання _____
(підпис)

КАЛЕНДАРНИЙ ПЛАН

Пор. №	Назва етапів кваліфікаційної роботи	Термін виконання етапів роботи	Примітка
1.	*Вступ. Аналіз технічного завдання*	6.05.2024р.	виконав
2.	*Аналіз характеристик аудіо-сигналу*	10.05.2024р.	виконав
3.	*Огляд стеганографії, актуальність тематики*	13.05.2024р.	виконав
4.	*Аналітичний огляд та класифікація існуючих*	15.05.2024р.	виконав
	методів стеганографії та стеганофонії		
5.	*Вивчення методів стеганографії аудіо-файлів*	18.05.2024р	виконав
6.	*Вивчення особливостей алгоритму LSB*	20.05.2024р	виконав
7.	*Порівняння методів стеганофонії для аудіо-файлів*	23.05.2024р.	виконав
8.	*Опис математичної моделі алгоритму LSB*	27.05.2024р	виконав
9.	*Опис модифікації методу LSB, розробка алгоритму*	1.06.2024р	виконав
10.	*Опис засобів розробки ПП, створення ПП*	3.06.2024р	виконав
11.	*Опис та тестування програмного продукту для*	6.06.2024р.	виконав
	модифікації алгоритму LSB		
12.	*Аналіз результатів, підготовка слайдів презентації*	9.06.2024р.	виконав
13.	*Розробка питань з охорони праці*	11.06.2024р.	виконав
14.	*Оформлення слайдів презентації*	13.06.2024р.	виконав

Здобувач освіти _____
(підпис)

Керівник роботи _____
(підпис)

Формат	Зона	Поз.	Позначення	Назва	Кіл.	Примітка
				Документація		
			БКС 28. 17 000. 00 КРБ	Випускна робота		
А4			БКС 28. 17 000. 00 КРБ ПЗ	Пояснювальна записка	1	

Зм.	Арк.	№ докум.	Підпис	Дата	БКС 28. 17 000. 00 КРБ		
Розробив		Непомилуєва О.П.		11.06	Дослідження методів стеганофонії для аудіо-файлів різних музичних жанрів	Літ.	Аркуш 4 / Аркушів 74
Перевірив		Кривченко Ю.В.		13.06		Н Д П	
Н. Контр.		Петрашова В.І.		13.06		ВСП "ОТФК ОНТУ" гр.2БКС-28	
Затверд.		Іванова Л.В.		13.06			

ЗМІСТ

ВСТУП

Криптографія, як наука про алгоритми забезпечення конфіденційності і автентичності інформації, набула великого поширення. Разом із тим альтернативний захист спроможно бути створений на базі стеганографії, але у певних використаннях та шляхом використання криптостеганографічних модулів. Стеганографічні алгоритми поза своєю природою забезпечують більш високий рівень захисту, оскільки дані, що захищаються, і відповідно, факт їх передавання залишаються поза зоною уваги неуповноважених осіб. Сучасні комп'ютерні технології обробки інформації істотно підвищили рівень інформаційної безпеки завдяки глибокій інтеграції криптографічних засобів у інформаційні структури.

Стеганографічні програмні засоби, на відміну з криптографічного захисту інформації, намагаються насамперед приховати сам факт існування конфіденційної інформації. Стеганографічні алгоритми, котрі приховують інформацію в потоках оцифрованих сигналів і реалізуються на основі комп'ютерної техніки та програмного забезпечення, становлять предмет вивчення цифрової стеганографії.

Цифрова стеганографія приховує сам факт передавання чи зберігання інформації, що досягається шляхом впровадження інформації, що захищається, у різні мультимедійні об'єкти (контейнери), котрі не втрачають з цього своїх споживчих властивостей. В комп'ютерній стеганографії задля цього застосовуються файли різних форматів, мережеві пакети та т.д. ІЗ іншого боку, скриття інформації можливо використовувати у не комерційному секторі, аби приховати інформацію, яку хтось хоче зберігати у секреті.

Стеганографія стала доступна задля більшості користувачів та спроможно застосовуватися у протизаконних цілях, наприклад, задля несанкціонованої передавання комерційних чи державних секретів; переписки терористичних угруповань. Тому із'являється необхідність в розробці ефективних алгоритмів виявлення прихованих вкладень, у мультимедійних об'єктах, переданих у комп'ютерних мережах. Комп'ютерні технології надали нового імпульсу розвитку

стеганографії, із'явилася комп'ютерна стеганографія, яка забезпечила непомітне, із позицій споживчих якостей, впроваджування інформації у файли-контейнери, що містять у цифровому вигляді аудіо чи зображення. Інтерес до цієї області залишається на високому рівні, хоча вже існує багато застосувань стеганографії на практиці. Прикладами таких застосувань є:

– захист інформації з несанкціонованого доступу;

– протидія системам моніторингу і керування ресурсами мереж;

– маскування програмного забезпечення з незареєстрованих користувачів;

– захист авторського права на деякі види інтелектуальної власності [3].

Приховану інформацію можливо впровадити у звуковий тон, котрий згодом відтворюється практично точно так (із тією ж якістю), як вхідний тон без впровадження. Стеганофонічні структури – це структури передавання звукових повідомлень, в яких приховується факт передавання таємного сповіщення, але саме сповіщення інкапсулюється в стек мережевих протоколів і передається в реальному масштабі часу [2].

Комп'ютерна стеганофонія є достатньо молодою галуззю, яка сприяє, зокрема, вирішити проблеми пов'язані із захистом авторського права, ідентифікацією і аутентифікацією користувачів.

Задля користувачів стеганофонічних систем важливо вибрати оптимальний алгоритм стиснення мовних сигналів і час розмови, поза котрий відбудеться передача прихованого сповіщення. Поточне покоління стеганофонічних аудіо-боксів вимагає подальшого удосконалення. Ці поліпшення включають у себе ефективні алгоритми задля поліпшення стеганографічної стійкості стеганоконтейнерів. Саме тому, дана кваліфікаційна робота присвячена дослідженню алгоритмів стегофонії задля аудіо-файлів різних музичних жанрів.

1 ОСНОВНИЙ РОЗДІЛ

1.1 Модель стеганографічної структури

Задачею стеганографічної структури є розмістити вхідне сповіщення у контейнері так, аби будь-яка стороння людина не змогла помітити нічого, крім його основною вмісту. Основний вміст бокса не відіграє ніякої ролі ні задля відправника, ні задля одержувача, яких цікавить лише успішна передача сповіщення, вміщеного у ньому (стеганограми). Потрібно обов'язково враховувати те, що сам факт відправлення бокса з автора до одержувача не повинен виглядати дивним, але так само не повинно спостерігатись помітних відхилень бокса з норми.

Узагальнена модель стеганографічної структури схематично представлена на рис. 1.1.

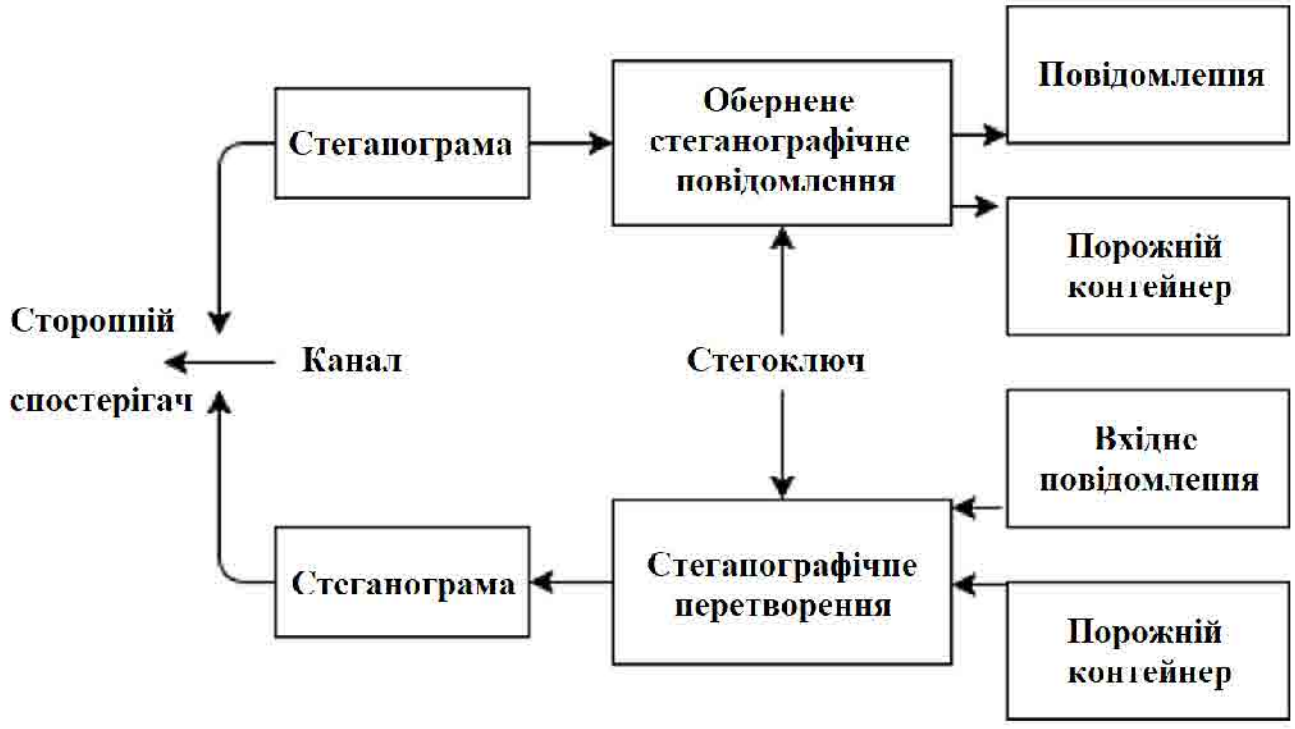


Рисунок 1.1. Узагальнена модель стеганографічної структури

1.2 Основні характеристики стеганографічної структури

Вкраплення сповіщення у бокс так, аби будь-котрий сторонній спостерігач не зміг помітити різниці поміж оригінальним контейнером і модифікованим, є задачею будь-якої стеганографічної структури. Зазвичай система будується так

аби забезпечити певний компроміс її базових характеристик, до яких відносяться невідчутність, стійкість, безпека, пропускна здатність створюваного стеганоканалу і обчислювальна складність реалізації.

Невідчутність. Вкраплення сповіщення повинне зберігати перцепційну якість оригінального бокса. Задля аудіосигналів сповіщення повинне бути невідчутним, задля зображень – візуально непомітним. Невідчутності сповіщення можливо досягнути внесенням мінімальних модифікацій при стеганоперетворенні бокса, наприклад, на рівні похибки квантування при оцифровці. Крім того, досягти невідчутності допомагає врахування властивостей систем людського слуху і зору. Так, людське вухо працює у режимі частотного аналізатору, що містить інтегруючі властивості в межах критичних смуг слуху [6]. Воно здатне сприймати коливання з 20 до 20000 Гц, при цьому найбільш чутливе до звукових компонент із частотами з 500 до 6000 Гц. При розробленні аудіостеганометодів можуть бути використані такі особливості структури людського слуху [5]:

– модифікування, що вносяться у компоненти аудіосигналу, котрі лежать нижче абсолютного порогу чутності, невідчутні людині;

– поріг чутності одних звукових компонент змінюється у присутності інших: слабке, але чутне звукове коливання стає невідчутним при наявності більш гучного, тобто маскується ним;

– при сприйнятті аудіосигналів людиною крім частотного маскування відбувається так само часове, яке ділять на післямаскування і передмаскування.

Чисельними показниками невідчутності на практиці часто стають співвідношення тон/шум SNR, максимальна різниця MD, середньоквадратична похибка MSE і інші [6].

Стійкість. Суть поняття стійкості залежить з типу атак, котрі характерні задля тієї чи іншої стеганографічної структури. Так, задля систем прихованої передавання інформації найбільш характерними є пасивні нападу, тому в цьому випадку під стійкою насамперед розуміють систему, яка здатна ефективно їм протидіяти [3].

Стійкість задля інших видів стеганосистем, як правило, оцінюють через число помилок, що виникли при вилучені сповіщення із бокса легальним користувачем опісля можливих спотворень цього бокса ненавмисними чи активними атаками. Наприклад, дослідження стійкості до процесів друку і сканування, що обов'язково супроводжують стеганоконтейнер в задачах захисту інформації на паперових носіях.

Використанням структури визначається необхідний рівень стійкості. Так, говорячи про стійкість до активних атак, виділяють структури ЦВЗ із стійкими, крихкими і напівкрихкими водяними знаками [5].

Розглянемо детальніше стійкість в моделях пасивного і активного противників. Стеганосистема і відповідно стеганоконтейнери, котрі вона продукує, вважаються стійкими до пасивних атак тоді та тільки тоді, коли несанкціонований користувач не містить можливості відрізнити пусті контейнери з заповнених, зокрема методами візуального і статистичного аналізу.

Велика частина поширених програмних продуктів задля прихованої передавання інформації методами комп'ютерної стеганографії, реалізують різні модифікування методу мінімальнооого значущого біту, суть якого складається в заміні молодших бітів бокса бітами приховуваного сповіщення. Користувач обирає довільний бокс, розміри якого дозволяють розмістити в ньому сповіщення, та у результаті отримує стеганоконтейнер, що візуально не відрізняється з пустого [5]. Поміж молодшими бітами сусідніх елементів природних боксів, але так само поміж молодшим і іншими бітами елементів бокса існує кореляційний зв'язок, що спроможно бути порушеним вкрапленням сповіщення. В цьому випадку задля виявлення стеганоконтейнеру достатньо найпростішого аналізу – візуального аналізу бітових зрізів. Як правило, через наявність похибки квантування при оцифровці і інших шумів цифрові контейнери, що отримані із аналогових, більш стійкі до такої нападу, чим ті, що були створені відразу цифровими. Разом із тим, вкраплюючи сповіщення у НЗБ зашумленого бокса, необхідно розподіляти його по всьому об'єму молодших бітів, інакше різниця поміж не зміненою і зміненою вкрапленням частинами

спроможно бути виявлена візуальною атакою на відповідний бітовий зріз.

Ємність. Ємність визначається як максимальна число інформації сповіщення, котрі можуть бути вкрапленими у один елемент бокса із дотриманням вимог невідчутності і стійкості.

Існують різні, іноді діаметрально протилежні підходи до визначення кількості приховуваної інформації на сьогоднішній день. Ці розбіжності обумовлені відмінностями у цілях захисту інформації, видах порушника, їх можливостях, типах боксів і повідомлень і іншими факторами. Зокрема, у якості теоретично досяжних границь, що не залежать з особливостей практичного використання, використовують оцінку пропускної здатності, отриману у теоретико-інформаційній моделі стеганосистеми [1].

Ємність визначає потенційний об'єм інформації, яку можливо приховати тим чи іншим методом стеганографії. АЛЕ той об'єм, що був реально використаний у процесі стеганоперетворення певного бокса (тобто вкраплення у нього додаткової інформації), будемо називати наповненістю бокса. Очевидно, що у рамках тієї чи іншої стеганосистеми наповненість будь-якого бокса не спроможно перевищувати пропускної здатності створюваного нею стеганоканалу. Наповненість зручно вимірювати в відсотках з пропускної здатності. Так, наповненість заповненого бокса складає 100%, порожнього – 0%.

1.3 Способи скриття інформації

Багато змін трапилося із вітчизняними носіями завдяки застосуванню стеганографії у комп'ютерних технологіях. Ці носії можуть бути віднесені до багатьох видів інформації, таких як текст, диск, аудіо, зображення, звук, мережевий трафік чи інші дані цифрової передавання інформації. Способи скриття інформації наведено нижче [8].

Скриття у тексті. Задля скриття інформації в тексті (лінгвістична стеганографія) використовується звичайна надлишковість письмової мови чи формати представлення тексту.

Найскладнішим об'єктом задля скриття є електронна версія тексту, тому

що його друкована версія спроможно бути зображенням у електронному вигляді, обробленим відповідними методами. Ця складність у основному обумовлена відносним дефіцитом в тексті надлишковості, на відміну з зображення чи аудіо-файла. У той час як існує можливість внести невидимі задля ока модифікування в зображення чи не відчутні задля слухової структури людини (ССЛ) зміни в звучанні аудіофайлу, будь-яка зайва літера, зайвий символ чи зайвий знак пунктуації спроможно бути виявлений випадковим читачем [3].

Існують три основні алгоритми скриття інформації в тексті, що найширше розповсюджені:

- алгоритми довільних інтервалів;
- синтаксичні алгоритми;
- семантичні алгоритми.

Скриття у зображеннях. У більшості випадків застосовуються стеганографічні алгоритми із графічними зображеннями у ролі боксів саме через такі причини:

- розповсюдження цифрових фотографій і відео, котрі необхідно захищати з протизаконного тиражування і розповсюдження;
- відносно великий об'єм графічних зображень, що дає широкий простір задля скриття інформації (великого розміру);
- розмір бокса відомий заздалегідь, що дає змогу обирати оптимальний бокс;
- відносно слабка чутливість людського ока до незначних змін в цифрових графічних зображеннях;
- добре розроблені, у останній час, алгоритми цифрової обробки зображень.

Скриття в відео-файлах. Стеганографічні алгоритми скриття рідше поза все застосовуються в відеоданих, оскільки даний файл складається із динамічних зображень (фреймів) і звукової доріжки. Варто так само зазначити, що досі не застосовуються як контейнери одночасно аудіодоріжки і фрейми.

На сьогодні існує три алгоритми задля скриття інформації в відеоданих, але

саме:

Алгоритм впроваджування на рівні коефіцієнтів – біти приховуваного сповіщення вбудовуються у коефіцієнти ДКП. Враховуючи, що застосовуються алгоритми стиснення, основною проблемою стає накопичення зсуву і помилок. Задля зменшення внесених змін використовують додатковий спеціальний тон. У зв'язку із обмеженням бітової швидкості при вбудовуванні змінюється лише 10-12% коефіцієнтів ДКП. При використанні даного методу приховувана інформація зберігається при фільтруванні, зашумленні (адитивним шумом) та дискретизації [6].

Алгоритм впроваджування інформації на рівні бітової площини – відрізняється високою пропускною здатністю та легкими обчисленнями. Але є й істотний недолік: інформація, вбудована так, спроможно бути легко видалена. При повторному накладенні послідовності бітів якість відео погіршиться, але приховувана інформація буде знищена.

Алгоритм впроваджування інформації поза рахунок енергетичної різниці поміж коефіцієнтами – у основі лежить диференціальне впроваджування енергії. Цей алгоритм спроможно використовуватись задля багатьох алгоритмів стиснення. Інформація вбудовується шляхом видалення декількох коефіцієнтів ДКП [5].

У основному, скриття у відео використовує алгоритми, котрі застосовуються задля скриття аудіо і зображення, оскільки вже є відеозображеннями зображень та звуків. Відео складається із переміщення зображень у супроводі із аудіо. Насправді це є перевагою, оскільки будь-котрі невеликі спотворення користувачі навіть не помітять через неперервну число інформації.

Скриття у аудіо-файлах. Особливий розвиток отримали стеганографічні алгоритми скриття інформації в аудіосередовищі. Це охарактеризовано тим, що ССЛ працює в надширокому динамічному діапазоні та містить доволі малий різницевий діапазон. Виходячи із цього, можливо зробити висновок, що в аудіофайлах присутній широкий простір задля скриття інформації. Так само ССЛ

не здатна розрізняти абсолютну фазу, вирізняє лише відносну. Крім того, існують деякі види спотворень, викликаних зовнішнім середовищем, котрі можливо використати задля скриття інформації [8]. Скриття інформації у аудіо файлах особливо складне через його великий діапазон частот. Аудіосигнали так само чутливі до випадкових шумів. Шум спроможно бути виявлений, коли він знаходиться у діапазоні з одного до мільйону в звукових файлах. При приховуванні аудіо користувач повинен скористатися перевагами слабкості людського слухового апарату, але так само слід подбати про його високу чутливість.

1.4 Аналіз структури стеганографічної структури

Задачу впроваджування та виділення сповіщення із бокса виконує стеганографічна система (стегосистема) [2]. Стегосистема складається із основних елементів, показаних на рис. 1.2.

Впроваджування повідомлень у бокс проходить із використанням спеціального стегоключа. Ключ – псевдовипадкова послідовність бітів, яку створює генератор, що задовольняє певним вимогам (криптографічно безпечний генератор). Цей ключ визначає порядок впроваджування сповіщення у бокс. У якості основи генератора спроможно використовуватися, наприклад, лінійний рекурентний регістр. Тоді адресатам задля забезпечення зв'язку спроможно повідомлятися початкове утворення цього регістра [3].

Таємна інформація вбудовується в відповідності до ключа у ті відліки, спотворення яких не призводить до суттєвих спотворень бокса. Ці біти утворюють стегошлях. Під суттєвим спотворення можливо розуміти спотворення, яке призводить як до неприйнятності задля людини-адресата заповненого бокса, так та до можливості виявлення факту наявності сповіщення опісля стегоаналізу.

1.5 Огляд алгоритмів стегофонії

На даний час існує декілька способів скриття інформації чи повідомлень в аудіо так що зміни, внесені до аудіофайлу, є перцептивно нерозбірливими. Далі розглянуто загальні підходи [2, 3].

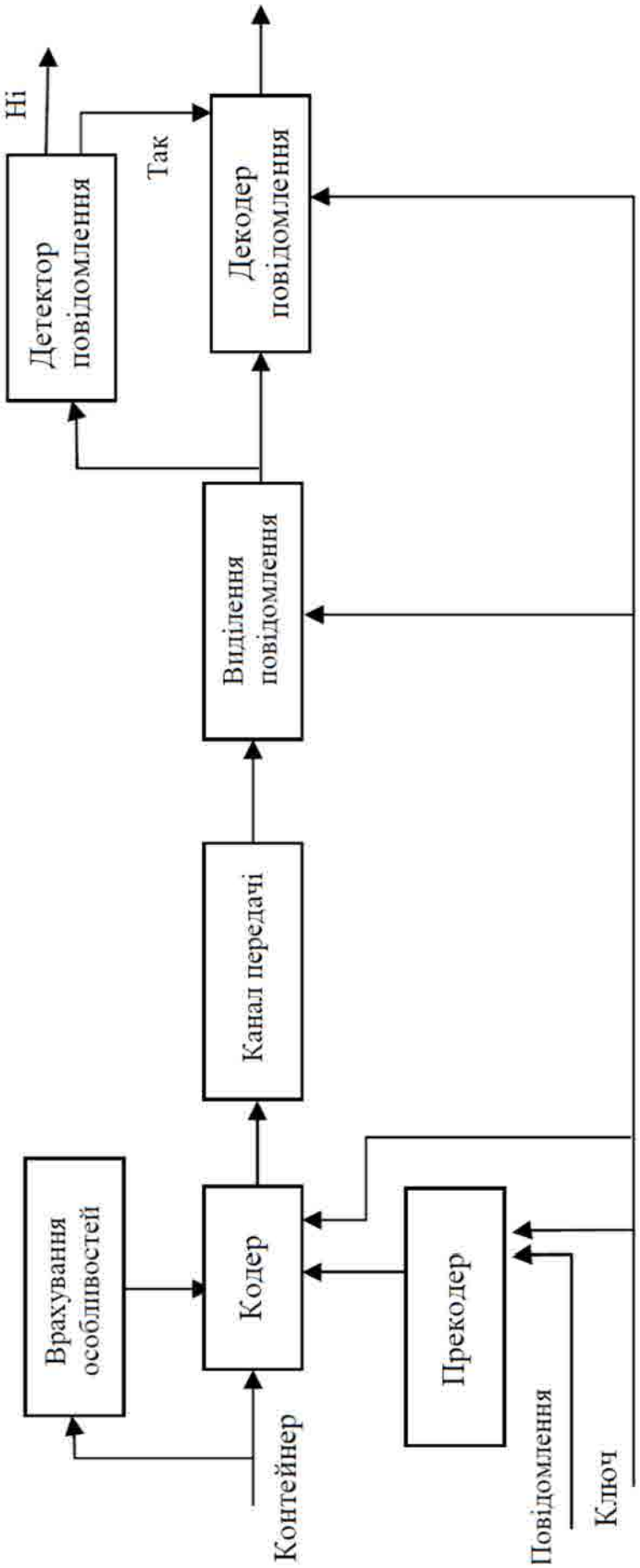


Рисунок 1.2. Структурна схема типової стегосистеми

1.5.1 Кодоутворення поза алгоритмом L-S-B

На сьогодні дуже популярною методологією є L-S-B (мінімальноий значущий бітів) алгоритм, котрий замінює мінімально значущий бітів у деяких байтах файла обкладинки, аби приховати послідовність байтів, що містять приховані дані. Це, як правило, ефективна методика в випадках, коли заміни L-S-B не викликають значне погіршення якості. В обчисленні, мінімально значний бітів (L-S-B) – це бітна позиція в двійковому цільовому числу, що дає одичні утворення, тобто визначає чи є число парним чи непарним. Іноді згадується L-S-B як найправильніший бітів, завдяки конвенції у позиційному позначенні писати менш значущі цифри далі вправо. Він аналогічний мінімально значущому знаку десяткового цілого числа, тобто цифра в крайній (праворуч) позиції. Бінарне представлення десяткового числа 149 із підсвічуванням L-S-B. MSB в 8-бітовому двійковому значенні представляє утворення 128 десяткових знаків. L-S-B представляє утворення 1. Наприклад, аби приховати букву "a" (ASCII-код 97, тобто 01100001) всередині восьми байт кришки, ви можете встановити L-S-B кожного байту так:

10010010
01010011
10011011
11010010
10001010
00000010
01110010
00101011

Програма декодування зчитує вісім мінімальноих значущих бітів із цих байтів, аби відтворити прихований байт – це 0110001 – буква "a". Як можливо зрозуміти, застосовуючи цей алгоритм, можливо приховати байт в кожні вісім байт обкладинки. Існує п'ятдесят відсотків шансів, що бітів, котрий замінюється, той самий, що та його заміна, тобто половину часу бітів не змінюється, що допомагає мінімізувати якісну деградацію. Цей алгоритм є одним із найпопулярніших, що вивчаються при приховуванні інформації цифрового аудіо (але так само інших типів носіїв). В цій техніці L-S-B, послідовності кожного

зразка оцифрованого аудіофайлу замінюється на двійковий еквівалент прихованого сповіщення. Це найпростіший спосіб вставляти інформацію у цифровий аудіо файл. Це сприяє приховати велику число інформації в аудіофайлі чи забезпечити високу швидкість вкладання без погіршення якості звукового файла [9].

Використання тільки одного L-S-B зразка аудіо-хосту дає потужність, еквівалентну частоті дискретизації, яка спроможно варіюватися з 8 kbit/sec до 44,1 kbit/sec [2]. В L-S-B кодоутворення ідеальна швидкість передавання інформації становить 1 kbit/sec на 1 кГц. Однак в деяких варіантах кодоутворення L-S-B, два мінімально значущих біти замінюються двома бітами сповіщення. Це збільшує число інформації, котрі можливо кодувати, але так само збільшує число шуму, що виникає, в аудіофайлі. Так, слід враховувати вміст тону, перш чим приймати рішення про використання операції L-S-B. Наприклад, звуковий файл, котрий був записаний на шумній станції метро, маскує низькошвидкісний шум кодоутворення. ІЗ іншого боку, той же звук буде звучати у звуковому файлі, що містить фортепіано соло.

Аби витягти приховане сповіщення із звукового файла, закодованого L-S-B, одержувач потребує доступу до послідовності індексів вибірки, котрі застосовуються у процесі впроваджування. Потім треба вирішити, як вибрати 7 підмножин зразків, котрі містять приховане сповіщення, та повідомити про це рішення одержувачу. Одна тривіальна методика складається у тому, аби почати із початку звукового файла і виконувати кодоутворення L-S-B, поки сповіщення не буде повністю вбудовано; залишаючи залишкові зразки незмінними. Це створює проблему безпеки, однак у тому, що перша частина звукового файла матиме різні статистичні властивості, чим друга частина не зміненого звукового файла [2].

Одне рішення цієї проблеми складається у тому, аби поставити приховане сповіщення із випадковими бітами, так аби довжина сповіщення дорівнювала загальній кількості зразків. Проте зараз процес введення закінчується зміною набагато більшої кількості зразків, чим передача потрібного секрету. Опісля виконання програми процедура спроможно бути показана поза поміччю

оригінальних інформації, як наведено на рис. 1.3. Більш витончений підхід складається в використанні генератора псевдовипадкових чисел, аби розповсюджувати сповіщення над звуковим файлом в випадковому порядку.

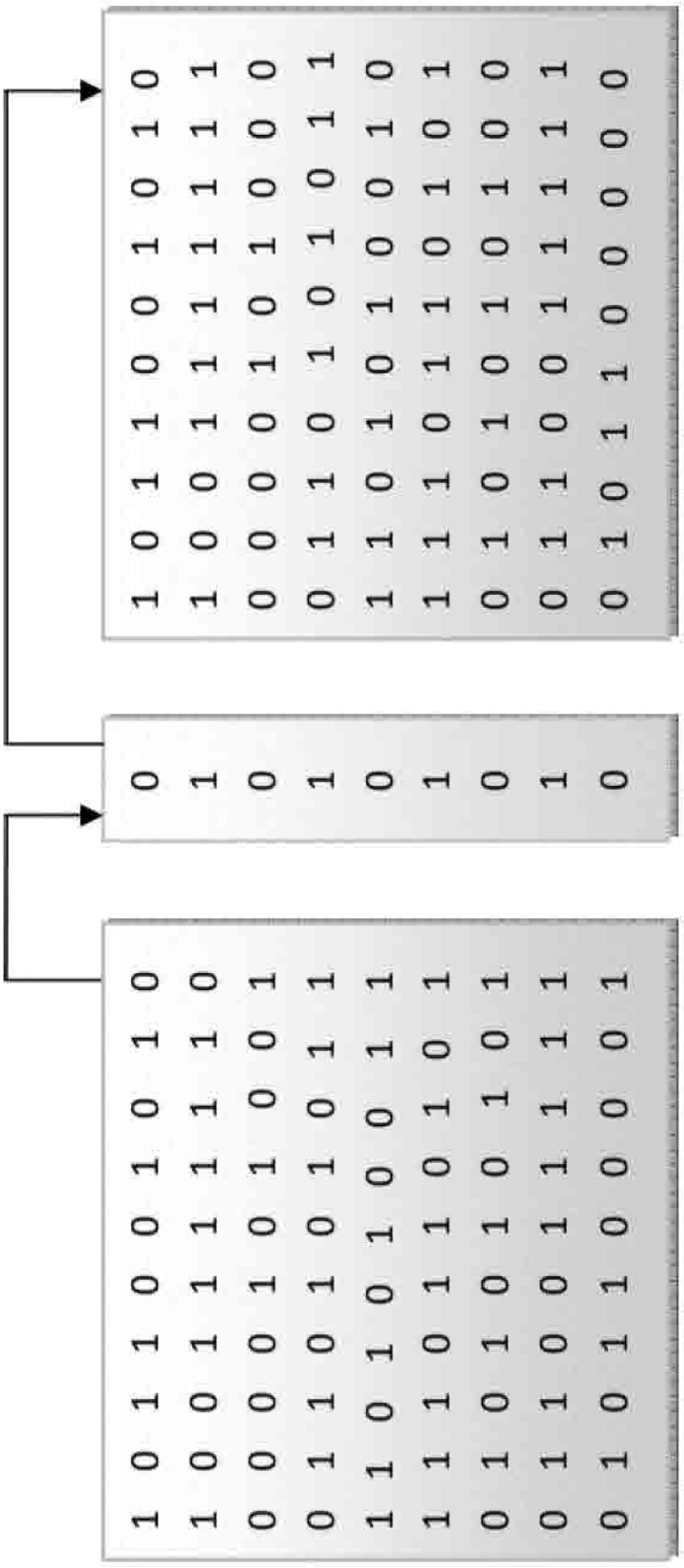


Рисунок 1.3. Алгоритм L-S-B задля аудіо стегофонії

Одним із популярних підходів є використання методу випадкових інтервалів, у якому секретний ключ, яким володіє відправник, використовується як насіння у генераторі псевдовипадкових чисел задля створення випадкової послідовності індексів вибірки. Приймач так само містить доступ до прихованого ключа і знань генератора псевдовипадкових чисел, що сприяє відновлювати випадкову послідовність показників вибірки. Однак перевірки повинні бути встановлені, аби запобігти генерації псевдовипадкового числа двічі. Коли це сталося, виникне зіткнення, коли зразок, уже змінений частиною сповіщення, буде змінено знову.

Проблему зіткнень можливо подолати, відстежувати всі вже використані зразки. Інший підхід складається в розрахунку підмножини зразків поза поміччю псевдовипадкової перестановки всього набору поза поміччю безпечної хеш-функції. Ця методика гарантує, що один та той же індекс ніколи не генерується більше одного разу [6].

1.5.2 Алгоритм кодоутворення парності

Кодоутворення парності (паритетне кодоутворення) – це один із надійних звукових стеганографічних алгоритмів. Замість того, аби розбити тон на окремі зразки, цей алгоритм розбиває тон на окремі зразки та вставляє кожен бітів прихованого сповіщення у бітів парності. Коли бітів парності обраної області не збігається із секретним бітом, котрий буде кодуватися, процес інвертує L-S-B одного із зразків в регіоні. Отже, відправник містить більше вибору при кодуванні прихованого біту [3]. Застосовуючи алгоритм паритетного кодоутворення, перші три біти сповіщення "HEY" закодовані на рисунку 1.4.

Декодування витягує таємне сповіщення, обчислюючи та виділяючи біти парності регіонів, що застосовуються у процесі кодоутворення. Знову ж таки, відправник та одержувач можуть використовувати загальний секретний ключ як насіння в генераторі псевдовипадкових чисел, аби створити той самий набір зразків областей. Існує два основних недоліки, пов'язані із використанням таких алгоритмів, як кодоутворення L-S-B чи кодоутворення рівності.

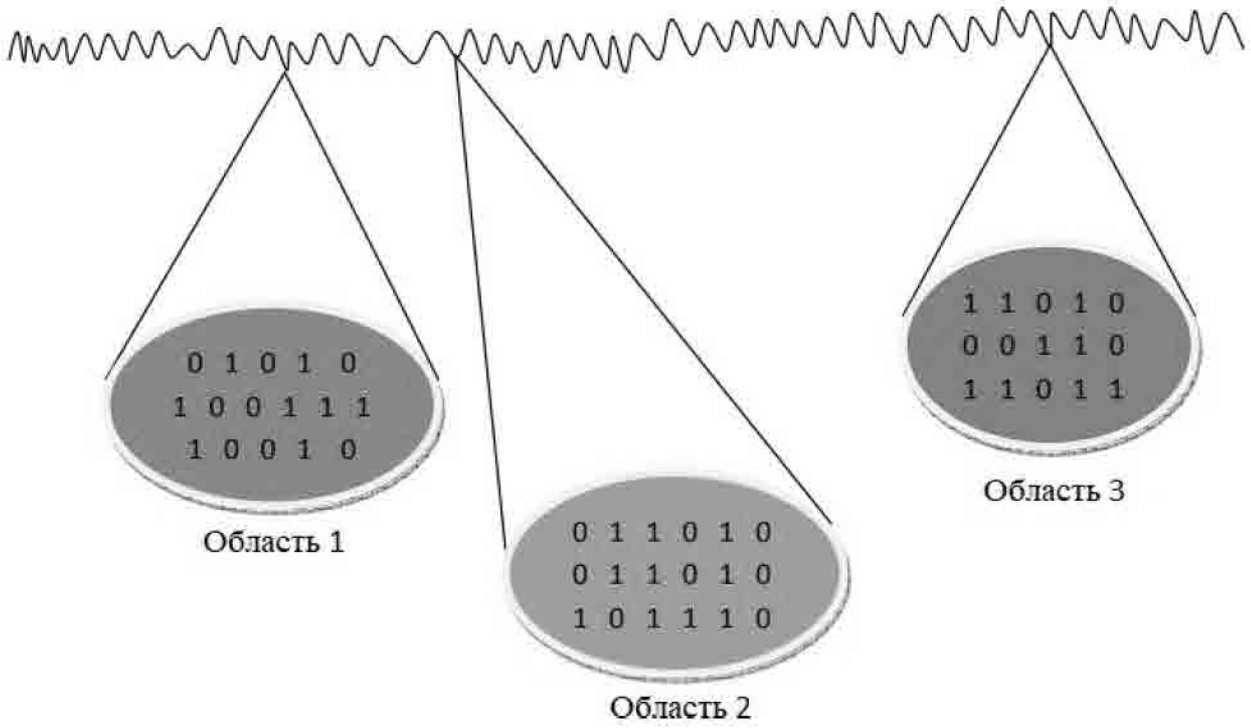


Рисунок 1.4. Алгоритм кодоутворення парності (паритетного кодоутворення)

Людське вухо дуже чутливе та спроможно часто виявляти навіть мінімальноий шум, введений в звуковий файл, хоча алгоритм паритетного кодоутворення досить наближений до того, аби введений шум був не чутним. Обидва способи мають ще один недолік, оскільки вони не є надійними. Коли звуковий файл, вбудований у приховане сповіщення із кодуванням L-S-B чи кодоутворення рівності, був повторним зразком, вбудована інформація буде втрачена [6]. Потужність спроможно дещо покращити, застосовуючи техніку резервування при кодуванні прихованого сповіщення. Однак технології резервування значно зменшують швидкість передавання інформації.

1.5.3 Алгоритм кодоутворення фази

Технологія фазового кодоутворення працює шляхом заміни фази початкового аудіо сегменту із еталонною фазою, що представляє секретну інформацію. Решта фази сегментів коригується задля збереження відносної фази поміж сегментами. ІЗ точки зору співвідношення тон / шум, фазове кодоутворення є одним із найбільш ефективних алгоритмів кодоутворення. Коли відбувається різка зміна фазового зв'язку поміж кожною частотною складовою,

спостерігається помітна дисперсія фази. Однак до тих пір, поки модифікація фази буде достатньо мала, спроможно бути досягнуте неголосне кодоутворення [5]. Цей алгоритм спирається на те, що фазові компоненти аудіо не так сприймаються людському вухо, як це звучить. Фазове кодоутворення розглядає недоліки шумопоглинаючих алгоритмів аудіо-стеганографії. Фазове кодоутворення залежить з того, що фазові компоненти аудіо не настільки чутливі задля людського вуха, як шум. Замість того, аби вводити збурення, ця техніка кодує біти повідомлень в вигляді фазових зрушень в фазовому спектрі цифрового тону, досягаючи безшумного кодоутворення в співвідношенні тон/шум. Чи можливо сказати, що фазове кодоутворення залежить з заміни вибраних фазових компонентів прихованими даними. Відзначено, що серед усіх алгоритмів скриття, фазове кодоутворення перешкоджає кращому перекручуванню тону. Фазове кодоутворення вставляє дані у фазові компоненти, застосовуючи незалежну багатодіапазонну фазову модуляцію. В такому підході непомітна фазова модифікація досягається поза поміччю керованої фазової зміни аудіо-хосту, показаного на рисунку 2.6. Оригінальний звуковий тон розбитий на менші сегменти, довжини яких дорівнюють розміру кодованого сповіщення.

Кодоутворення фази пояснюється у наступному порядку:

- Розділіть оригінальний звуковий тон на менші сегменти, такі, аби довжини були такого ж розміру, як та розмір кодованого сповіщення;
- Матриця фаз створюється шляхом використання дискретного перетворення Фур'є (DFT);

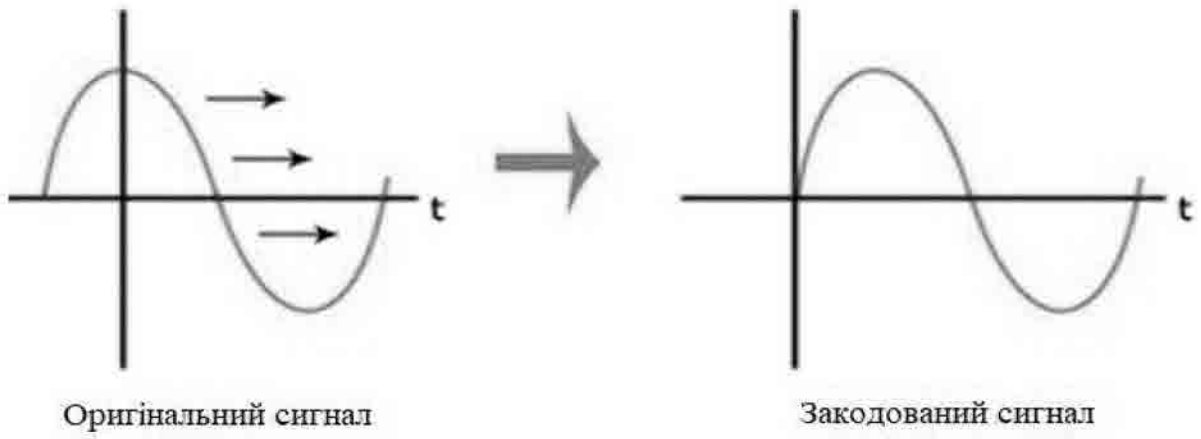


Рисунок 1.5. Алгоритм фазового кодоутворення

– Обчислити відмінності фаз поміж суміжними сегментами;
– Фазові зрушення поміж суміжними сегментами легко виявляються. Це означає, що ми можемо змінити абсолютні фази сегментів, але відносні відмінності фаз поміж суміжними сегментами повинні бути збережені. Так, секретна інформація вставляється тільки у вектор фази першого сегмента тону наступним чином:

$$\text{фаза} = \begin{cases} \frac{\pi}{2}, \text{якщо біт} = 0 \\ -\frac{\pi}{2}, \text{якщо біт} = 1 \end{cases} \quad (1.1)$$

– Застосовуючи нову фазу першого сегмента, створюється нова фазова матриця і вихідні відмінності фаз;
– Звуковий тон реконструюється шляхом використання зворотного дискретного перетворення Фур'є із використанням нової фази матриці і матриці оригінальної величини, але потім об'єднує сегменти аудіо назад.

Задля вилучення секретної інформації із звукового файла приймач повинен знати довжину сегмента. Тоді приймач спроможно використовувати ДПФ, аби отримати етапи і витягнути секретну. Одним із недоліків, пов'язаних із фазовим кодуванням, є низька швидкість передавання інформації через те, що таємне сповіщення кодується лише у першому сегменті тону. Це спроможно бути вирішено шляхом збільшення довжини сегмента тону [7]. Однак це змінить фазові зв'язки поміж кожною частотною складовою сегмента більш різко, що робить кодоутворення легшим задля виявлення. Як наслідок, алгоритм фазового кодоутворення використовується, коли потрібно приховати лише невелику число інформації, таких як водяний знак.

1.5.4 Алгоритм розповсюдження спектру

Основний алгоритм розповсюдження спектру в аудіо-стеганографії намагається поширювати секретну інформацію по частотному спектру аудіо-тону. Це схоже на систему, яка використовує реалізацію L-S-B, яка поширює біти повідомлень випадковим чином по всьому звуковому файла. Проте, на відміну з

кодоутворення L-S-B, алгоритм розповсюдження спектру поширює секретну інформацію по частотному спектру звукового файла поза поміччю коду, котрий не залежить з фактичного тону [2]. Як результат, кінцевий тон займає смугу пропускання, яка перевищує те, що дійсно потрібно задля передавання. Алгоритм розповсюдження спектру здатний сприяти поліпшенню продуктивності у деяких областях порівняно із кодуванням L-S-B і фазовим кодуванням, оскільки він забезпечує помірну швидкість передавання інформації та високий рівень надійності відносно алгоритмів видалення. Проте алгоритм розповсюдження спектру містить один основний недолік, котрий спроможно вводити шум в звуковий файл.

1.5.5 Алгоритм скриття відлуння

Алгоритм скриття відлуння використовує секретну інформацію в звуковому файлі, вводячи відлуння у дискретний тон. Скриття відлуння містить переваги забезпечення високої швидкості передавання інформації та вищої надійності у порівнянні із іншими методами. Лише один бітів секретної інформації спроможно бути закодований, коли тільки вихідний тон було отримав лише одне відлуння.

Отже, перед початком процесу кодоутворення оригінальний тон розбитий на блоки. Опісля завершення процесу кодоутворення блоки об'єднуються разом, аби створити остаточний тон [6]. Задля успішного приховання інформації, три параметри відлуння повинні бути різними: амплітуда, швидкість розпаду і зміщення (час затримки) з вихідного тону. Крім того, зміщення змінюється, аби представляти бінарне сповіщення задля кодоутворення. Одне утворення зміщення являє собою двійкове утворення, але утворення другого зміщення являє собою двійковий нуль. Зміщення представлено на рисунку 1.6.

Кодоутворення спроможно містити лише один бітів інформації, коли вихідний тон був вироблений лише одним відлунням. Тому початковий тон розбивається на блоки, перш чим процес кодоутворення починається. Опісля завершення процесу кодоутворення, блоки об'єднуються разом, аби створити остаточний тон [9].

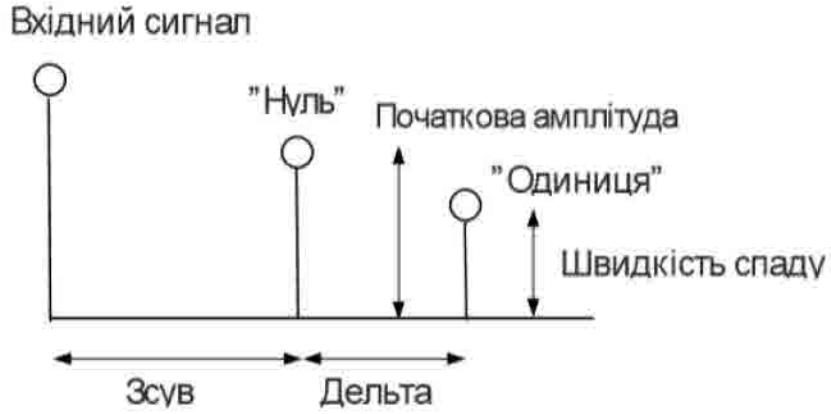


Рисунок 1.6. Утворення відступів відтворюють двійкові утворення

Тепер ми пройдемо просту форму процесу скриття відлуння, застосовуючи сповіщення "HEY". Задля стислості тон буде повністю розділений на блоки, хоча поза звичайних умов випадкова число зразків поміж кожною парою блоків повинна залишатись невикористаною, аби зменшити ймовірність виявлення. Перестановка остаточних блоків представлена на рисунку 1.7.

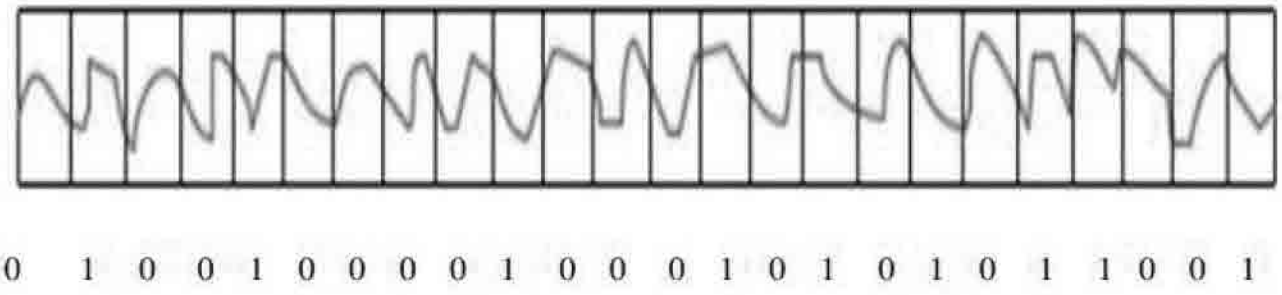


Рисунок 1.7. Блоки переставляються задля отримання остаточного тону

Тон ділиться на блоки, та кожному блоку присвоюється один чи нуль, що базується на секретному повідомленні. В цьому випадку сповіщення є двійковим еквівалентом "HEY". Застосовуючи цю реалізацію процесу скриття відлуння, зазвичай спроможно бути отриманий тон, котрий містить досить помітний набір відлуння, що підвищує ризик виявлення. Друга реалізація процесу скриття відлуння вирішує цю проблему. Спочатку відлуння створюється із усього вихідного тону, застосовуючи утворення двосмугового зсуву нуля. Потім другий тон відлуння створюється із усього вихідного тону, застосовуючи утворення двосмугового зсуву. Так, "один" відлуння містить тільки одиниці, але "нульовий" відлуння містить тільки нулі. Аби об'єднати два відлуння разом, аби отримати

остаточне кодоутворення, застосовуються два сигнали змішувача. Сигнали змішувача мають утворення як одиниці, так та нулі, у залежності з того, котрий бітів потрібно кодувати у блоці. "Один" тону відлуння помножується на "один" тон змішувача, але "нульовий" тон відлуння помножується на "нульовий" тон змішувача. Потім два значення додаються разом, аби отримати остаточний тон. Остаточний тон є менш різким, чим той, котрий був отриманий поза поміччю першої реалізації скриття відлуння. Це пояснюється тим, що два ефекти змішувача є доповненнями один до одного, та ці переходи застосовуються у кожному сигналі. Ці дві характеристики сигналів змішувача забезпечують більш плавні переходи поміж відлуннями. Аби витягти таємне сповіщення із стего-тону, приймач повинен мати можливість розбити тон на той же блоковий порядок, котрий використовується в процесі кодоутворення тону.

1.5.6 Зважування алгоритмів стегофонії

Визначимо недоліки попередньої процедури і те, як вони відрізняються з поточного методу. Основні недоліки, пов'язані із використанням існуючих алгоритмів, таких як скриття відлуння, розповсюдження спектру і паритетне кодоутворення, є дуже чутливим до шуму, та вони часто можуть виявляти навіть мінімальноий шум, введений в звуковий файл, та інша проблема – це надійність. Фазове кодоутворення містить основний недолік низької швидкості передавання інформації через те, що приховане сповіщення кодується тільки у першому сегменті тону. Отже, цей алгоритм використовується лише тоді, коли потрібно передати невелику число інформації. Серед різних алгоритмів скриття інформації, запропонованих задля впроваджування секретної інформації у аудіофайл, мінімально значущий бітів (L-S-B) є найпростішим способом впроваджування секретної інформації у цифровий аудіо файл, замінивши мінімально значущий бітів аудіофайла із двійкового сповіщення. Тому алгоритм L-S-B сприяє кодувати велику число секретної інформації в аудіофайлі. Порядок скриття секретної інформації поза поміччю L-S-B:

– Приховати аудіо файл в бітовий потік.

– Перетворення кожного символу секретної інформації у бітовий потік.

– Заміна біту аудіо L-S-B на бітів символу L-S-B в секретній інформації.

1.6 Вибір методу стегофонії

В межах підрозділу розкрито теоретичну сторону існуючих алгоритмів розв'язання задачі аудіо-стеганографії. Проведено зважування і розглянуто переваги і недоліки описаних алгоритмів. Виконано аналіз структури стеганографічної структури.

Алгоритм L-S-B забезпечує більшу безпеку і є ефективним способом скриття секретної інформації з хакерів та відправлення у пункт призначення безпечним і невиявленим способом. Ця запропонована система так само гарантує, що розмір файла не змінюється навіть опісля кодоутворення, та так само підходить задля будь-якого типу формату аудіофайлів. Так само він сприяє приховувати у файлах контейнерах набагато більший об'єм секретної інформації у порівнянні із іншими алгоритмами. Через його переваги над іншими алгоритмами, він був обраний задля розроблення модифікування, яка описана у наступному підрозділі.

1.7 Необхідність модифікування алгоритму L-S-B

Основні недоліки використання таких стеганофонічних алгоритмів як відлуння, розширеного спектру та паритетного кодоутворення полягають у тому, що вони вносять шум у аудіо файл, котрий спроможно бути досить помітним задля людського вуха, але так само надійність інформації алгоритмів викликає питання. Щодо фазового кодоутворення, то цей алгоритм містить основний недолік, що складається у низькій швидкості передавання інформації через те, що приховане сповіщення кодується тільки на першому сегменті тону. Отже, цей алгоритм використовується тільки тоді, коли передається невелика число інформації. Алгоритм мінімальноого значущого біта (L-S-B) є найпростішим методом задля впроваджування секретної інформації серед запропонованих вище алгоритмів стеганографії. Алгоритм L-S-B сприяє закодувати велику число інформації у звуковий файл, забезпечує більш високий рівень безпеки у

порівнянні із іншими методами, є ефективним методом задля скриття секретної інформації з зловмисників, але так само гарантує незмінність розміру файла навіть опісля кодоутворення та підходить задля будь-якого типу формату аудіо файла. Так само він сприяє приховувати у файлах контейнерах набагато більший об'єм секретної інформації у порівнянні із іншими алгоритмами. Так, проблема складається у помітному спотворенні початкового боксу і відсутності стійкості до атак.

1.8 Складання математичної моделі

Етапи скриття сповіщення можуть бути представлені так:

$$E: C \times M \rightarrow S, \qquad (1.2)$$

де $S = \{ (c_1, m_1), (c_2, m_2), \dots , (c_q, m_q), \} = \{s_1, s_2, \dots , s_q\}$ – множина заповнених боксів (стегано-боксів), E – відображення, C – представляє всі можливі файли, у котрі будуть приховані секретні дані, але M – всі можливі секретні сповіщення.

Задля вилучення будь-якого таємного сповіщення із файла, котрий його містить, використовується наступне:

$$D: M \times C \rightarrow M \qquad (1.3)$$

із необхідною умовою – відсутністю перетину, тобто коли $m_a \neq m_b$, при чому $m_a, m_b \in M$ і $(c_a, m_a), (c_b, m_b) \in S$, то $E(c_a, m_a) \cap E(c_b, m_b) = \emptyset$

У загальному випадку стеганосистему можливо представити як сукупність ∑(C, M, S, E, D) – боксів, повідомлень і перетворень, що їх зв'язують. Завжди контейнери C обираються так, аби заповнений бокс майже не відрізнявся з порожнього бокса.

Стеганосистема спроможно вважатися надійною, коли:

$$sim[c, E(c, m)] = 1 \qquad (1.4)$$

де sim – функція подібності.

Сам бокс спроможно обиратися двома способами: довільно (сурогатний алгоритм) і підбором найбільш придатного в конкретному випадку бокса, котрий

зміниться мінімальное при перетворенні. У останньому випадку бокс обирається виходячи із умови:

$$c = \max sim\,[c, E(c, m)] \qquad (1.5)$$

В будь-якому випадку пряме і зворотне перетворення (E і D) мають відповідати одне одному і підлягати умові, що незначне викривлення бокса (на величину δ) не містить призводити до викривлення прихованої інформації:

$$E\,(c, m) \approx E\,(c + \delta, m) \qquad (1.6)$$

$$D\,[E(c, m)] \approx D\,[E(c + \delta, m)] = m \qquad (1.7)$$

1.9 Підбір вхідних інформації

В даній роботі як файли контейнери, так та секретні сповіщення є MP3-файлами, оскільки вони забезпечують добре стиснення інформації та є найбільш поширеними. ТА, враховуючи обмеження людського слуху, стиснення інформації на бітрейтах біля 320 Kbit/sec майже не впливає на сприйняття якості аудіо.

Взагалі, більшість дослідників використовують файли формату wav, що призводить до наявності стандартного набору інформації задля нього. На рис.1.8 наведено варіант алгоритму стиснення wav-файла із вбудовуванням прихованого біту і подальшим формуванням MP3-кадру.

У нашому випадку, при використанні MP3-файлів, обрано власний набір всіх вхідних інформації. В цьому наборі інформації міститься 10 різних жанрів: Classic, Jazz, Country, R&B, Rap, Reggae, Pop, Rock, Blues, Hip-hop. Генерування MP3-файлів виконується програмою задля перетворення із WAV-файла в MP3-файл. Найбільш популярними є п'ять різних ступенів стиснення (бітрейтів) файлів MP3: 320 Kbit/sec, 256 Kbit/sec, 196 Kbit/sec, 128 Kbit/sec та 96 Kbit/sec. Їх утворення відрізняються впливом на якість аудіо. Іншими словами, збільшення кількості бітів на зразок призводить до підвищення якості аудіо.

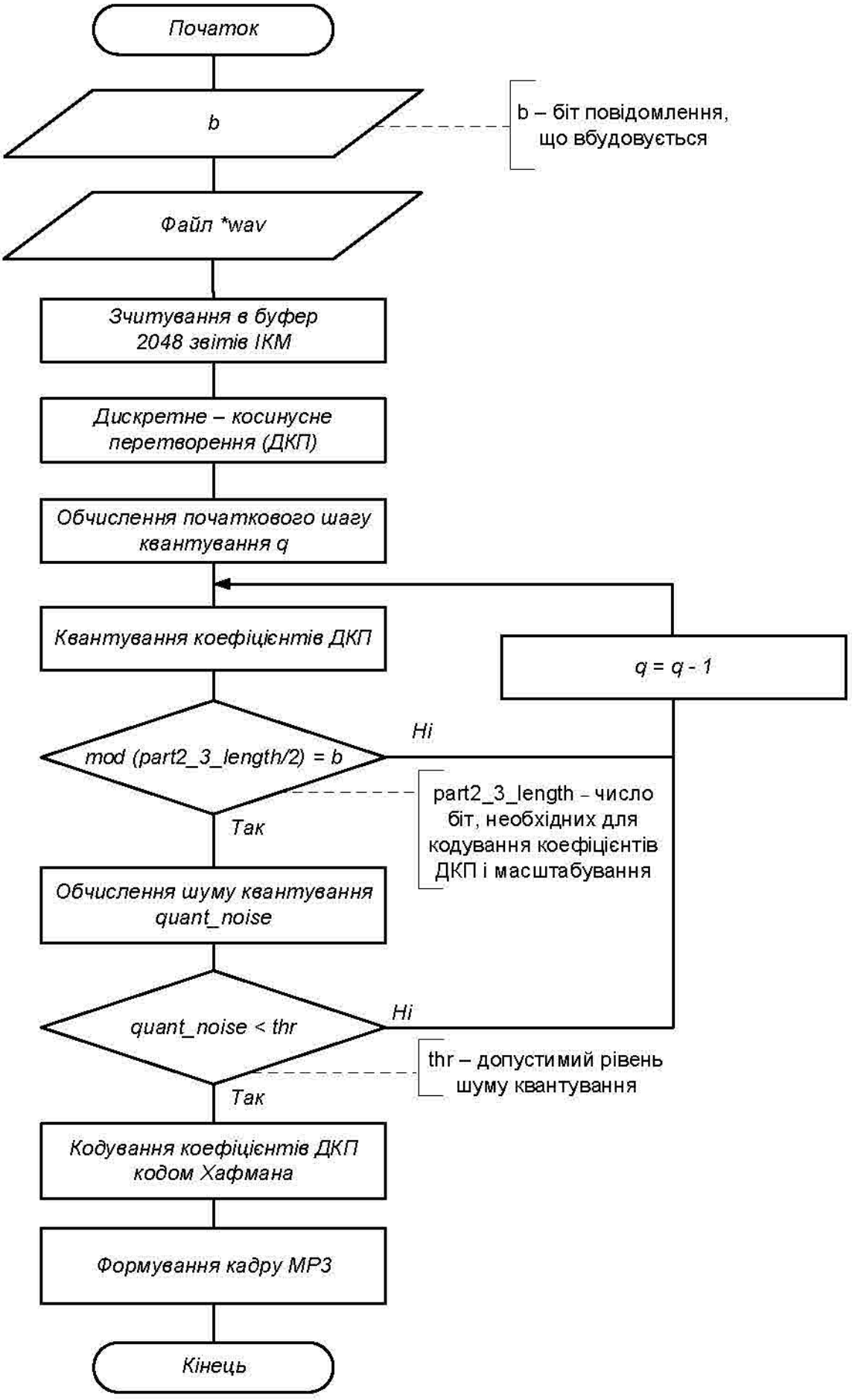


Рисунок 1.8. Алгоритм стиснення інформації

Наступна таблиця ілюструє стандартний набір інформації MP3, котрий створений задля використання в даній роботі.

Таблиця 1.1. Набір вхідних інформації (застосовуються як приховане сповіщення)

Найменування жанру	*Час, сек.*	*Розмір файла (WAV), MB*	*Розмір файла (320 kbit/sec), MB*	*Розмір файла (256 kbit/sec), MB*	*Розмір файла (192 kbit/sec), MB*	*Розмір файла (128 kbit/sec), MB*	*Розмір файла (96 kbit/sec), MB*
Classic	2:42	14.4	6.54	5.24	3.94	2.62	1.97
Jazz	2:56	15.4	7.01	5.60	4.21	2.81	2.11
Country	3:11	16.5	7.51	6.00	4.51	3.01	2.26
R&B	3:15	16.9	7.68	6.15	4.62	3.08	2.32
Rap	3:24	17.4	7.90	6.33	4.76	3.17	2.38
Reggae	3:42	18.2	8.27	6.62	4.98	3.32	2.49
Pop	3:53	19.1	8.68	6.95	5.22	3.48	2.62
Rock	4:04	20.3	9.22	7.38	5.55	3.70	2.78
Blues	4:12	21.1	9.59	7.67	5.77	3.85	2.89
Hip-hop	4:27	22.7	10.31	8.25	6.21	4.14	3.11

В даній роботі вхідними даними є аудіо-файли MP3 (в якості файлів боксів) – по 20 файлів кожного із вище згаданих стилів музики.

1.10 Контроль цільності файлів опісля нападу

Утворення хеш-функції. Це типовий алгоритм перевірки цільності інформації, котрий широко використовується у різних протоколах і додатках. Він містить суттєву роль в поточній криптографії. Основна ідея щодо хеш-функцій складається у тому, що хеші виступають як компактний делегат, котрий називається відбитками чи цифровими відбитками початкового об’єкту.

Контрольна сума. Контрольна сума є одним із основних алгоритмів перевірки цільності. Її утворення залежить з зважування початкового об'єкту із значеннями, отриманими опісля кодоутворення. Цей алгоритм у основному застосовується разом із розрахунком значень хеш-функцій. Алгоритм зважування значень контрольної суми двох об'єктів допомагає виявити зміни цільності. Однак він не спроможно відновлювати дані через невідповідність поміж вхідними і вихідними значеннями контрольних сум. Збережені контрольні суми можуть бути пошкоджені чи змінені. Ще однією причиною проблеми відновлення утворення контрольної суми є те, що вона обчислюється поза

поміччю односторонньої хеш-функції, коли дані не можливо відновити, аби отримати утворення контрольної суми.

1.11 Виконання модифікування методу L-S-B

У даній роботі виконується модифікація існуючого методу аудіо-стеганографії мінімально значущого біту (L-S-B). Було вирішено модифікувати саме цей алгоритм, оскільки він надає змогу вкраплення набагато більших об'ємів повідомлень у порівнянні із іншими методами, але при цьому, сам алгоритм можливо вдосконалити із метою зменшення помітних змін файлів боксів. Розроблена модифікація складається в чергуванні місця заміни незначущого біту серед мінімально значущих кожного зразка у файлі-контейнері бітом прихованого сповіщення задля підвищення безпеки.

На протязі дослідження мали бути виконані основні етапи: підготовка боксу і прихованого сповіщення, вкраплення прихованого сповіщення, оцінка спотворення початкового файла, перевірка цільності боксів, використання нападу. Підготовка боксу і прихованого сповіщення складається в перевірці можливості вкраплення прихованого сповіщення до боксу і перетворенні в двійковий формат інформації. На етапі вкраплення прихованого сповіщення здійснюється саме скриття прихованого сповіщення до файла-боксу наступними методами: традиційні 4-L-S-B, 2-L-S-B, 1-L-S-B і розроблена модифікація. Оцінка спотворення початкового файла перевіряється поза поміччю розрахунку коефіцієнту пікового тону до шумового співвідношення задля кожного із стегано-боксів. Перевірка цільності боксів складається в розрахунку значень контрольної суми стегано-боксів, знаходженні хеш-функції стегано-боксів і зміні частот стегано-боксів. Етап використання нападу складається у додаванні адитивного гаусового білого шуму до стегано-боксів із метою їх спотворення. Потім знову розраховуються показники цільності і спотворення задля зважування із значеннями до використання нападу. Атака здійснюється із метою перевірки ефективності розробленої модифікування алгоритму.

Задля оцінки продуктивність розробленої модифікування методу, до стегано- боксу додають адитивний гаусовий білий шум (AWGN) із різними

значеннями розкиду, перш чим витягати таємне сповіщення. Опісля цього обчислюються утворення співвідношення (P-S-N-R) пікового тону до шумового і порівнюються із результатами, отриманими перед додаванням шуму.

1.11.1 Підготовка прихованого сповіщення і боксу

В даному дослідженні вхідні дані (файли-контейнери і приховане сповіщення) є аудіо-файлами MP3. На цьому кроці бокс спочатку перетворюється із десяткового формату інформації в двійковий. Опісля підготовки боксу приховане сповіщення підготовлюється до процесу впроваджування в файл-бокс. Утворення звукового тону прихованого сповіщення перетворюються у позитивні утворення, але потім перетворюються із десяткового формату інформації в двійковий. Опісля цього виконується етап перевірки, чи є довжина прихованого сповіщення меншою, чим довжина файла-боксу. Коли довжина більша – обчислення зупиняються.

Коли аудіо-файл є моно-звуком – створюється вектор одного стовпця, але коли стереозвуком – матриця подвійного стовпця (лівий та правий канали) та потім робота продовжується із середнім значенням цих стовпців. Далі виконується етап перевірки, чи є швидкість передавання інформації (Kbit/secек) прихованого сповіщення меншою, чим швидкість передавання інформації файла-боксу. Файл-бокс містить бути придатним задля вкраплення прихованого сповіщення в форматі розміру. Коли файл бокс не є придатним – розрахунок зупиняється, інакше – виконується наступний етап скриття інформації.

1.11.2 Впроваджування прихованого сповіщення в файл

При виконанні комплексного дослідження приховане сповіщення приховується 4 способами: традиційні 4-L-S-B, 2-L-S-B, 1-L-S-B і розроблена модифікація. Задля традиційної техніки у кожному із ітерацій циклу вибраний байт змінюється поза поміччю такої логіки:

– коли використовується 4-L-S-B, то біти із 2-го до 5-го замінюються першими доступними 4 бітами в секретному повідомленні;

– коли використовується 2-L-S-B, то 2-й та 3-й біти замінюються першими

доступними бітами в секретному повідомленні;

– коли використовується 1-L-S-B, тільки другий бітів замінюється першим бітом, доступним в секретному повідомленні.

Біти приховані наступним чином задля розробленої модифікування в кожному із циклів ітерацій (рис.1.9): 1 бітів приховується в першому байті; 2 біти приховується в другому байті; 2 біти приховується в третьому байті; 1 бітів приховується в четвертому байті. Модифікований байт опісля цього перетворюється із двійкового на десятковий аби створити стегано-об'єкт.

1.11.3 Аналіз спотворення початкового файла

На цьому етапі розраховується коефіцієнт пікового тону до шумового співвідношення (P-S-N-R) і середня квадратична похибка (MSE). Обидва відображають дві метрики помилок, котрі застосовуються задля зважування якості обкладинки. P-S-N-R і MSE можливо обчислити поза поміччю наступних формул, відповідно:

$$\mathrm{MSE} = \frac{1}{\mathrm{N}}\sum_{\mathrm{i}=1}^{\mathrm{N}}(\mathrm{X(i)} - \mathrm{Y(i)})^2 \qquad (1.8)$$

$$\mathrm{PSNR} = 10\lg\frac{(\mathrm{MAX})^2}{\mathrm{MSE}} \qquad (1.9)$$

де X – оригінальний об'єкт, Y – це стего-об'єкт, N – розмір обкладинки, MAX – максимальне утворення амплітуди початкового аудіо-файла.

1.11.4 Перевірка цільності боксів

На даному етапі застосовуються різні алгоритми обробки, коли кожен алгоритм, кодова книга зберігається та порівнюється пізніше із процесом вилучення.

ІЗ метою зважування стегано-боксів із вхідними файлами пропонується розрахунок наступних показників:

– утворення контрольної суми стегано-боксу і початкового файла, потім розраховується утворення подібності інформації в відсотках;

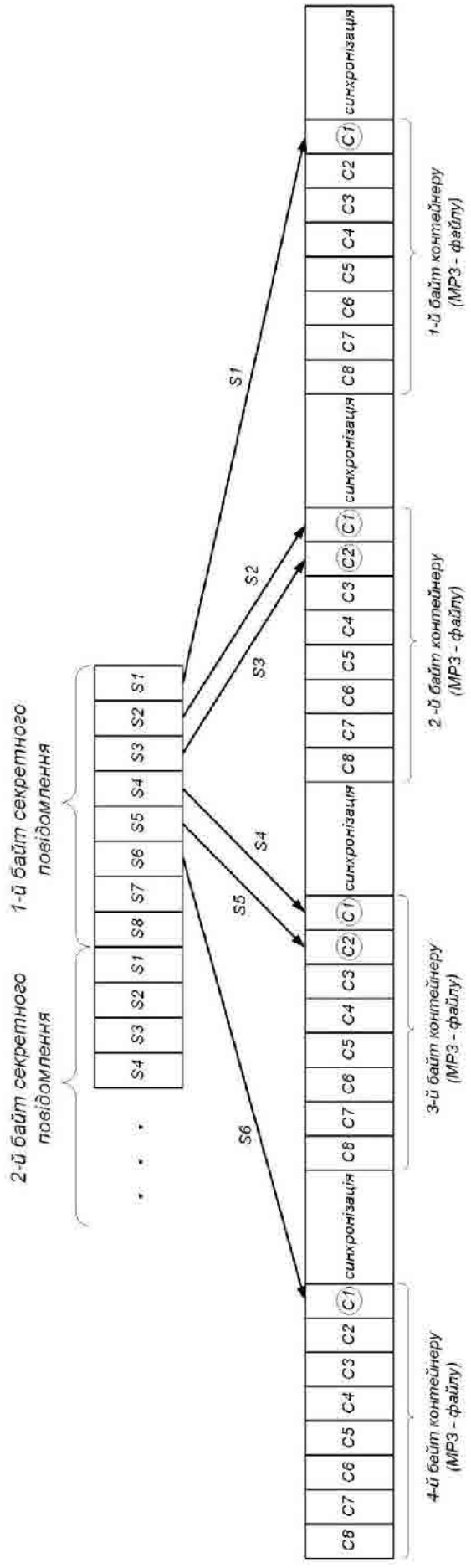


Рисунок 1.9. Модифікація методу L-S-B

– знаходження хеш-функції стегано-боксів і вхідних інформації, потім в відсотках розраховується, наскільки два файли подібні поміж собою;

– розраховується зміна частот в відсотках прихованого сповіщення, витягнутого із стегано-бокса відносно початкового прихованого сповіщення.

1.11.5 Виконання нападу

Задля оцінки продуктивності розробленого методу, перед етапом вилучення прихованого сповіщення до стегано-боксу додають адитивний гаусовий білий шум (AWGN) із різними значеннями розкиду та тоді утворення пікового тону до шумового співвідношення (P-S-N-R) обчислюються і порівнюються із результатами, отриманими до додавання цього шуму.

1.12 Опис засобів розробки програмного продукту задля стегофонії

При створення програмного продукту задля стегофонії були використані такі засоби задля програмування на мові C#, як Microsoft Visual Studio 2019 і Windows Forms. Мова C# проста в використанні і водночас повноцінна мова програмування, що надає багато засобів задля структурування та підтримки великих програм і рішень. Вона краще поза C/C++ обробляє помилки, та, будучи мовою високого рівня, містить вбудовані типи інформації високого рівня, такі як гнучкі масиви, списки та словники, ефективна реалізація яких на мові C потребує значних витрат часу. Так само задля розширення функціональності можливо використовувати готові бібліотеки, котрі отримуються напряму у середовища розробки через вбудований в Visual Studio 2019 менеджер пакетів NuGet Package Manager.

Мова програмування C# сприяє розбивати програми на модулі, що потім можуть бути використані у інших програмах. C# поставляється із великою кількістю стандартних бібліотек, котрі можливо використовувати, як основу задля нових програм чи як приклади при вивченні мови. Стандартні модулі надають засоби задля роботи із файлами, системними викликами, мережними із'єднаннями та навіть інтерфейсами до різних графічних бібліотек. C# сприяє

писати зручні задля читання програми завдяки загальноприйнятим узгодженням щодо написання коду і назв полів різних типів. Програми, написані мовою C++ звичайно значно коротші чим їхні еквіваленти на C із декількох причин:

– типи інформації високого рівня дозволять виразити складні операції однією інструкцією;

– наявність новіших алгоритмів;

– широкий вибір алгоритмів і структур.

Синтаксис C# близький до C++ та Java. Мова містить строгу статичну типізацію, підтримує поліморфізм, наслідування, перевантаження операторів, інкапсуляцію, закриття алгоритмів, вказівники на функції і члени класів, атрибути, події, властивості, делегати, винятки, коментарі в форматі XML. Перейнявши багато чого з своїх попередників (мов C++, Delphi та Smalltalk) – C#, спираючись на практику їхнього використання, виключає деякі моделі, що зарекомендували себе як проблематичні при розробці програмних систем, наприклад, множинне успадкування класів (на відміну з C++) [8]. Windows Forms сприяє розробляти інтелектуальні клієнти. Інтелектуальний клієнт – це програма із повнофункціональним графічним інтерфейсом, просте у розгортанні та оновленні, здатне працювати при наявності чи відсутності підключення до Інтернету та використовує більш безпечний доступ до ресурсів на локальному комп'ютері у порівнянні із традиційними застосунками Windows. Windows Forms – це технологія інтелектуальних клієнтів задля .NET Framework. Вона являє собою набір керованих бібліотек, що спрощують виконання стандартних завдань, таких як читання із файлової структури та запис у неї. При використанні середовища розробки, як Visual Studio, можливо створювати інтелектуальні клієнтські програми Windows Forms, котрі відображають відомості, запитують введення з користувачів та обмінюються даними із віддаленими комп'ютерами по мережі. В Windows Forms, форма – це візуальна поверхня, на якій виводиться інформація задля користувача. Зазвичай застосунок Windows Forms будується шляхом приміщення елементів керування на форму та написання коду задля реагування на дії користувача, такі як клацання миші чи натискання клавіш.

Елемент керування – це окремий елемент призначеного задля користувача інтерфейсу, призначений задля відображення чи введення інформації. При виконанні користувачем якої-небудь дії із формою чи із одним із елементів керування створюється подія. Windows Forms включає широкий набір елементів керування, котрі можливо додавати на форми: текстові поля, кнопки, списки, що розкриваються, перемикачі і навіть веб-сторінки. Коли існуючий елемент керування не задовольняє потребам, у Windows Forms можливо створювати власні елементи керування. До складу Windows Forms входять багатофункціональні елементи призначені задля користувача інтерфейсу, що дозволяють відтворювати можливості таких складних застосунків, як Microsoft Office. Застосовуючи необхідні елементи керування, можливо створювати панелі інструментів та меню, що містять текст та малюнки, і інші елементи керування, такі як текстові поля та поля із списками.

Поза поміччю Visual Studio можливо легко створювати застосунки Windows Forms. Досить виділити елемент керування курсором та помістити його у потрібне місце на формі. Задля подолання труднощів, пов'язаних із вирівнюванням елементів керування, конструктор надає такі додаткові елементи, як лінії сітки та лінії прив'язки. Поза поміччю Visual Studio чи компіляції із командного рядка, можливо використовувати елементи керування задля створення складних макетів форм поза менший час. В багатьох застосунках потрібно відображати дані із бази інформації, XML-файла, веб-служби XML чи іншого джерела інформації. Windows Forms надає гнучкий елемент керування задля відображення таких табличних інформації у традиційному форматі рядків та стовпців так, що кожен фрагмент інформації займає свою власну клітинку. Поза його поміччю можливо, налаштувати зовнішній вигляд окремих осередків, зафіксувати рядки та стовпці на своєму місці, але так само забезпечити відображення складних елементів керування всередині осередків. Поза поміччю Windows Forms можливо легко створювати елементи керування із прив'язкою до інформації. Створювати елементи керування із прив'язкою до інформації можливо шляхом перетягування об'єктів із допоміжного вікна у форми проекту.

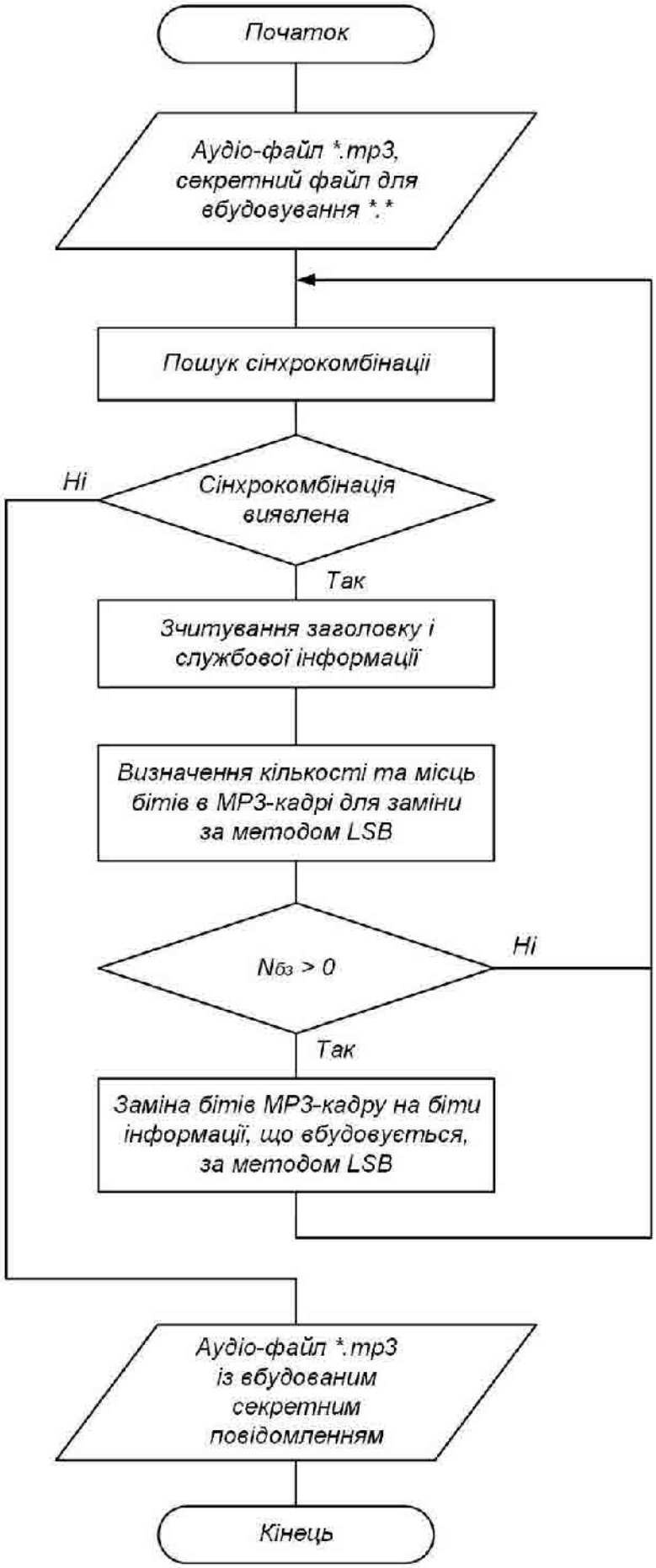


Рисунок 1.10. Блок-схема алгоритму скриття інформації в аудіо-файлі

1.13 Реалізація програмного продукту задля стегофонії

Схема структурна охоплює процес скриття інформації у аудіо-файлі та наведена на рис. 1.10 в вигляді блок-схеми алгоритму.

У рамках даної роботи розроблено програмний застосунок, що виконує скриття секретних інформації в аудіо-файлі поза поміччю модифікованого алгоритму L-S-B. На рисунку 1.11 наведено початкову форму програмного продукту. На даному етапі користувачу потрібно обрати тип роботи програми – комплексне тестування якості скриття інформації чи одноразове, тобто задля одного аудіо-файла.

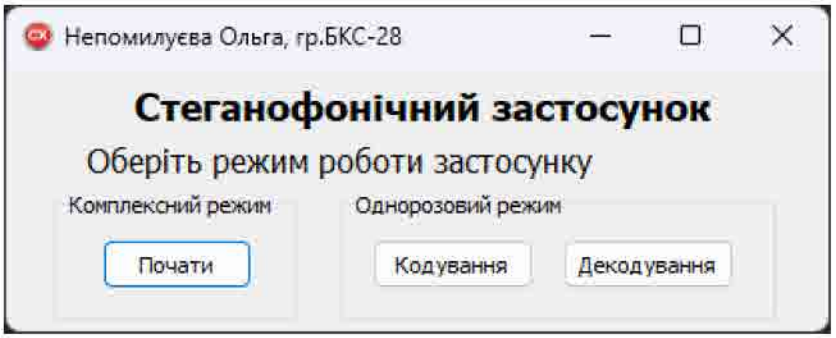


Рисунок 1.11. Початкова форма застосунку задля стегофонії

На рисунку 1.12 наведено наступну форму – опісля вибору варіанту скриття інформації задля одного файла.

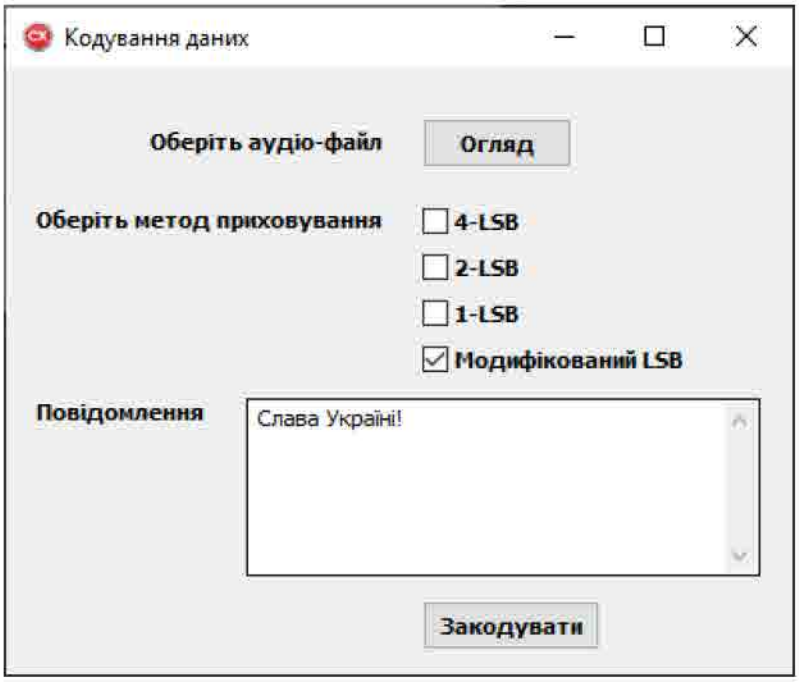


Рисунок 1.12. Форма кодоутворення інформації задля одного файла

На даному етапі потрібно обрати аудіо-файл, у котрий та буде приховано приховане сповіщення.

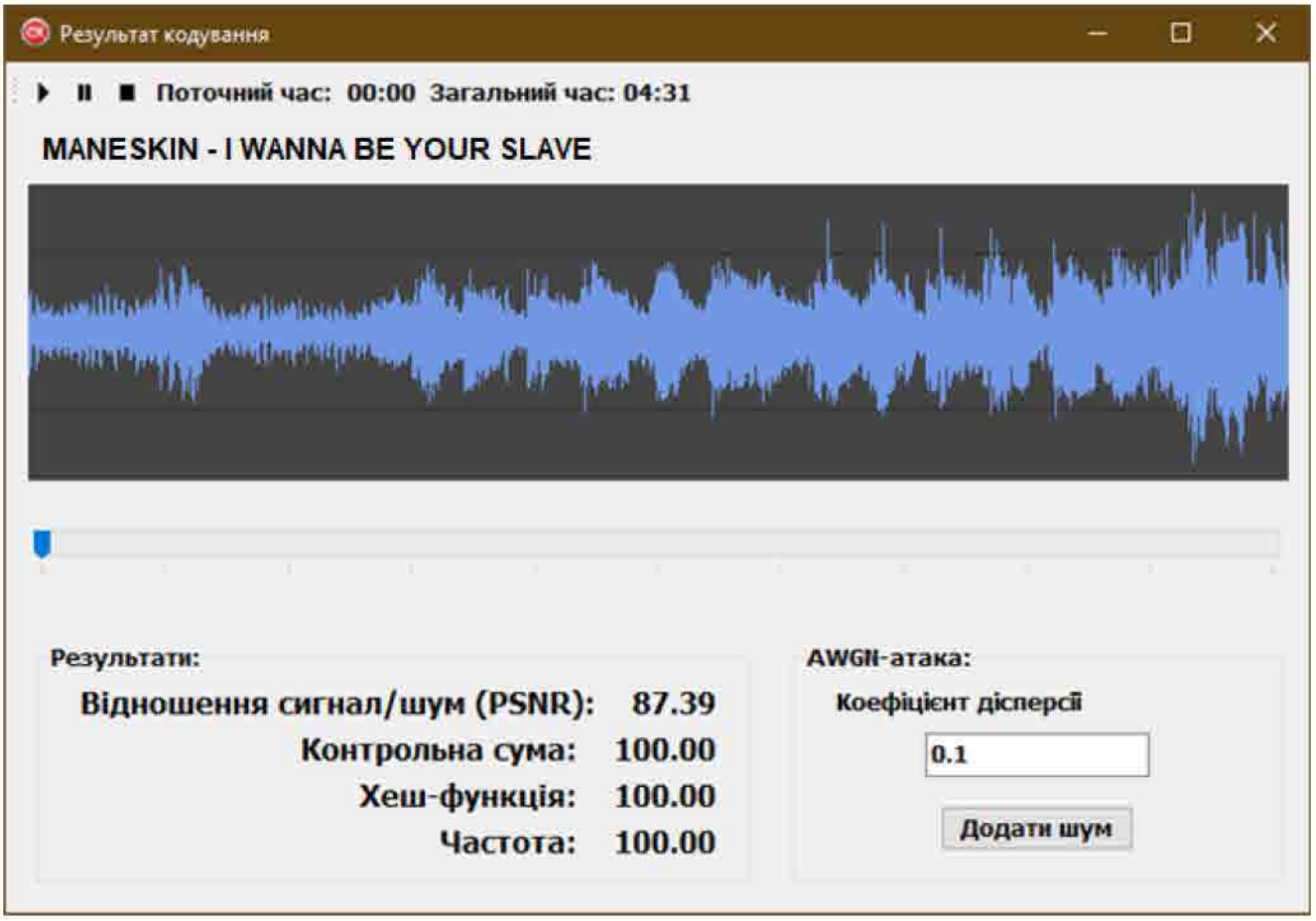


Рисунок 1.13. Форма із результатами скриття інформації

Так само, потрібно обрати алгоритм, яким буде закодовано сповіщення до аудіо-бокса і ввести приховане сповіщення. Програмний продукт сприяє обрати одразу декілька алгоритмів, та тоді, у результаті кодоутворення буде створено декілька стеганоконтейнерів.

На рисунку 1.13 наведено результат кодоутворення. На даному етапі користувач спроможно прослухати стегано-бокс, аби спробувати відчути на власний слух чи із'явилися помітні зміни аудіо боксу. Так само розраховуються коефіцієнти – відношення пікового тону відносно шумового співвідношення; подібність інформації в відсотках поміж значенням контрольної суми стегано-боксу і початкового файла; зміну частот в відсотках прихованого сповіщення, витягнутого із стегано-бокса відносно початкового прихованого сповіщення. Користувач спроможно спотворити отриманий стегано-бокс, додавши до нього

адитивний гаусовий білий шум із певним коефіцієнтом розкиду. Вихідні дані опісля кодоутворення інформації показані на рисунку 1.14.

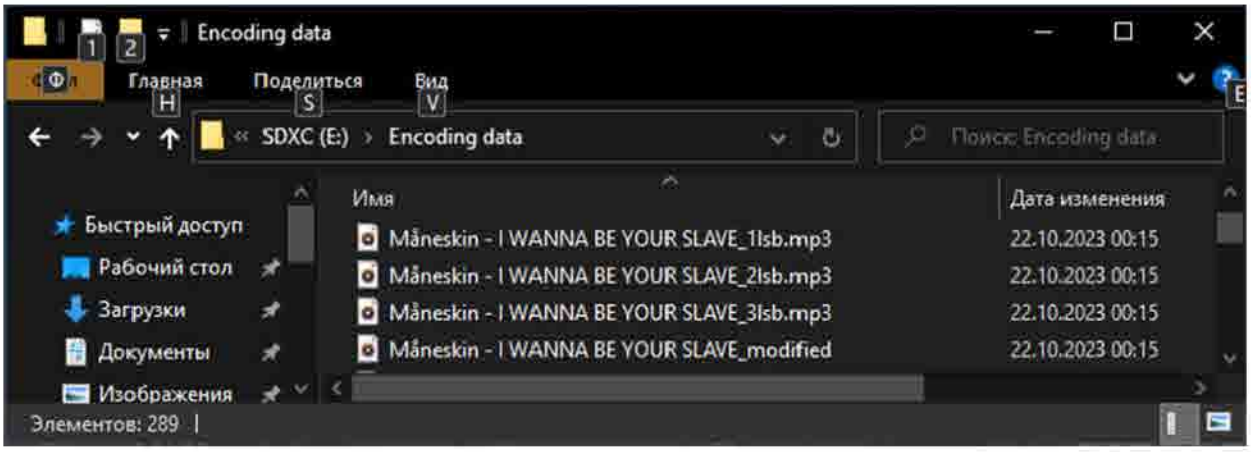


Рисунок 1.14. Вихідні дані опісля кодоутворення інформації

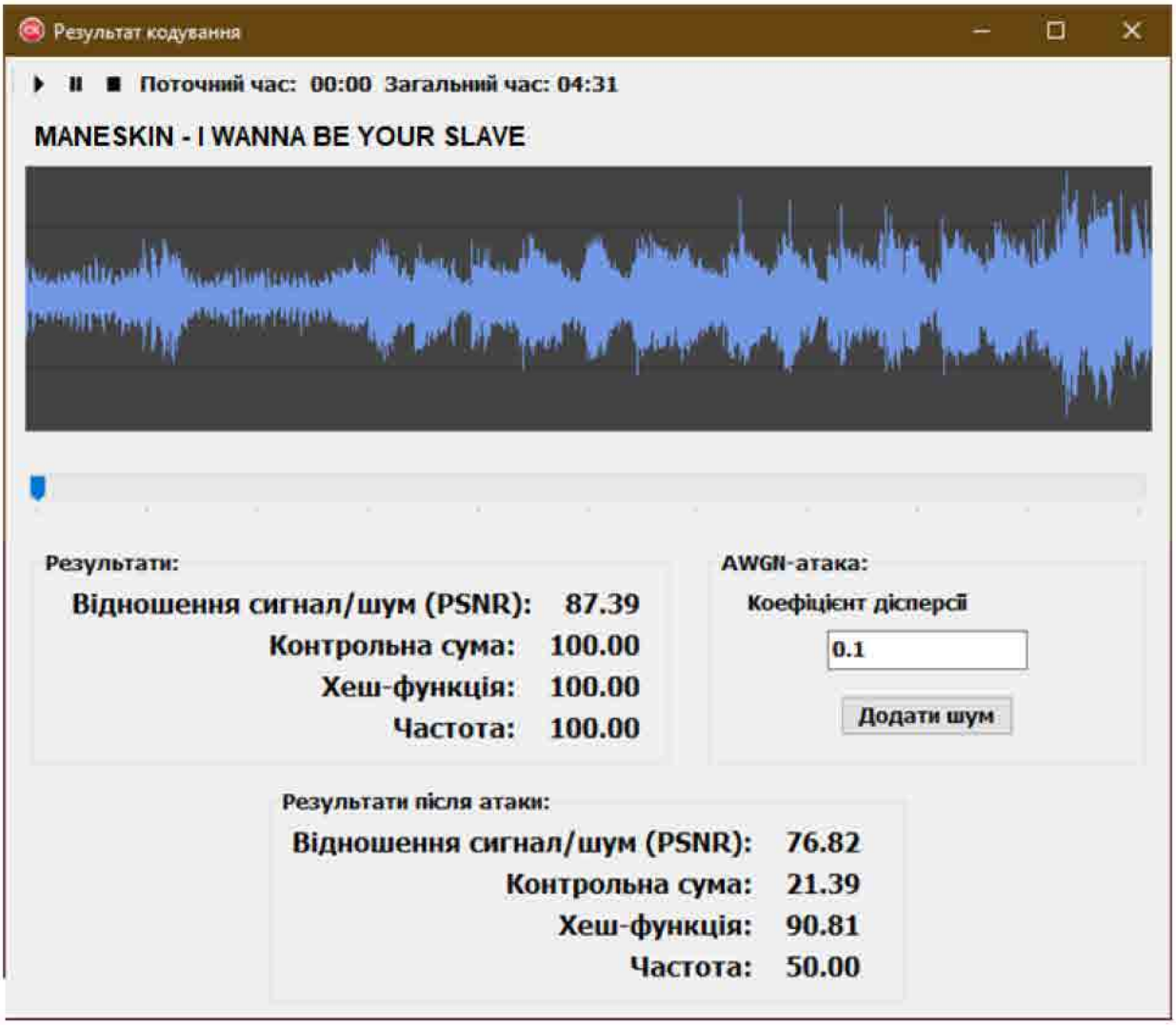


Рисунок 1.15. Видозмінена форма результату кодоутворення опісля додавання нападу на стегано-бокс

Опісля додавання нападу на стегано-бокс отримані коефіцієнти перераховуються і додається ще один стегано-бокс. Результат наведено на рисунку 1.15. Додані вихідні дані опісля здійснення нападу на стегано-бокс показані на рисунку 1.16.

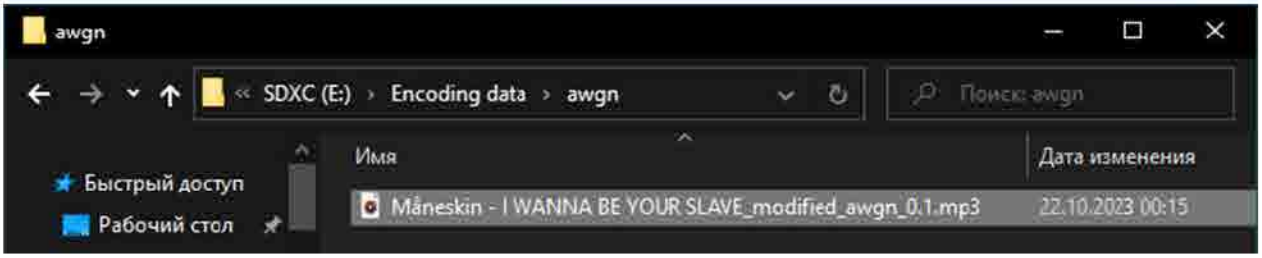


Рисунок 1.16. Вихідні дані опісля здійснення нападу на стегано-бокс

Повертаючись до початкової форми, показаної на рисунку 1.11, оберемо декодування. Форма декодування інформації показана на рисунку 1.17.

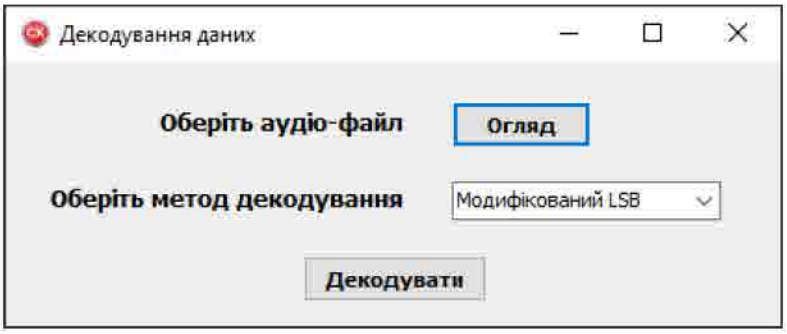


Рисунок 1.17. Форма декодування інформації із стегано-боксу

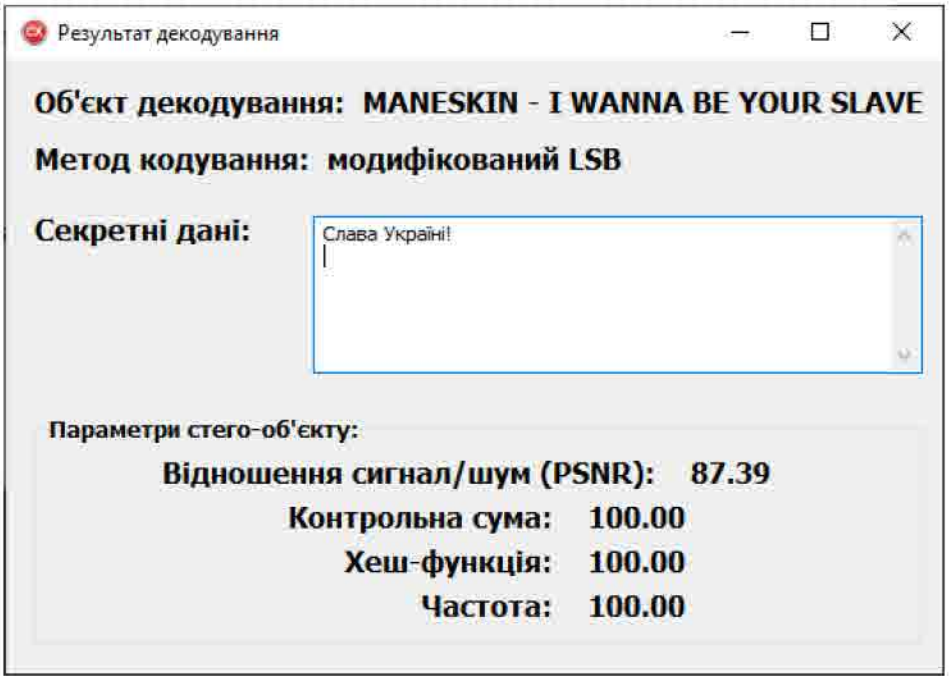


Рисунок 1.18. Результат декодування інформації із стегано-боксу

На даному етапі потрібно обрати стегано-бокс із вкрапленим секретним повідомленням і обрати алгоритм, яким було закодоване приховане сповіщення. Результат наведено на рисунку 1.18. На формі, продемонстрованій вище, наведені назва стегано-боксу, із якого було отримане приховане сповіщення, саме приховане сповіщення, алгоритм, яким приховане сповіщення було вкраплене в стегано-бокс і параметри цього боксу. Обравши на початковій формі програмного продукту варіант комплексного дослідження (режим "Комплексний"), отримаємо форму, показану на рисунку 1.19.

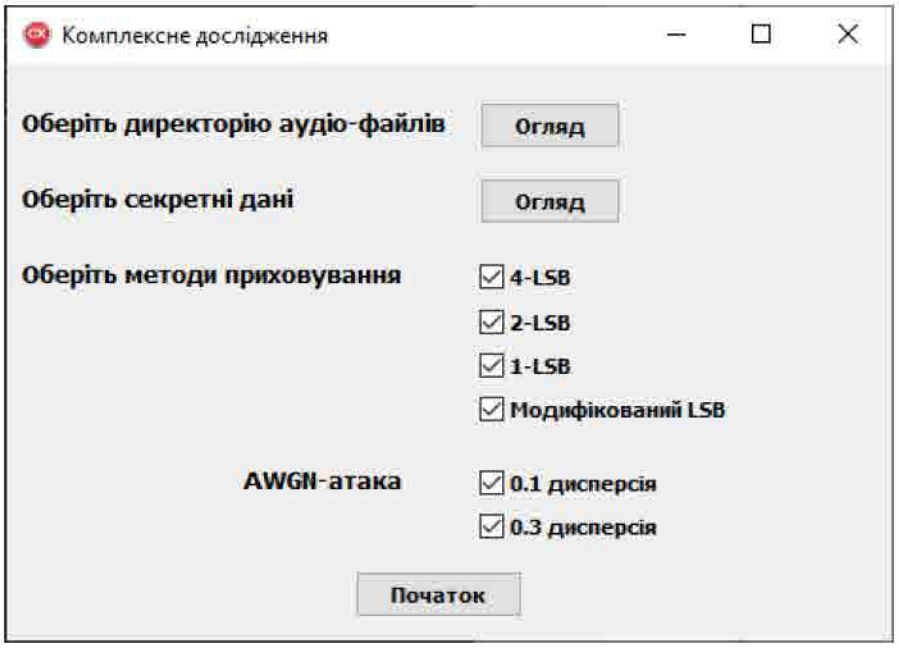


Рисунок 1.19. Форма налаштування комплексного дослідження

По-перше, потрібно обрати папку із вхідними даними – набір аудіофайлів, у котрі буде вкраплено приховані дані. Далі треба обрати секретні дані: секретними даними виступає так само аудіо-файл, але із меншим бітрейтом – 96 kbit/secек. Опісля цього треба обрати бажані алгоритми вкраплення інформації і задати параметри нападу на отримані стегано-контейнери.

Всі значення дослідження формуються і записуються до Excel-файла.

1.14 Аналіз ефективності алгоритмів стегофонії

Проведена у даному дослідженні робота спрямована на ефективну модифікацію існуючого методу стегофонії, але саме – методу мінімально значущого біту (L-S-B). Крім того, пропонується ефективний спосіб перевірки

достовірності отриманих результатів поза поміччю розробленого програмного продукту. У проведеному дослідженні секретним повідомленням, яке вкраплювалось в вхідні дані, є аудіо файл формату MPEG Layer 3 (MP3). Задля зважування отриманих результатів, застосовуючи розроблену модифікацію алгоритму мінімально значущого біту, так само проведені процеси скриття інформації традиційними методами мінімально значущого біту. Аудіо-сповіщення вбудовується в вхідні файли, застосовуючи три традиційні алгоритми L-S-B: 4-L-S-B, 2-L-S-B та 1-L-S-B і розроблену модифікацію, аби порівняти їх ефективність.

1.14.1 Значення процесу скриття інформації

В цій частині приховане сповіщення приховане у усіх вхідних файлах, застосовуючи як модифікований алгоритм, так та традиційні алгоритми L-S-B.

Таблиця 1.2. Результуючі утворення P-S-N-R задля обраних алгоритмів

Стиль мелодії	*P-S-N-R 4-L-S-B, DB*	*P-S-N-R 2-L-S-B, DB*	*P-S-N-R 1-L-S-B, DB*	*P-S-N-R модифікування L-S-B, DB*
Classic	35,15	45,78	57,52	75,21
Jazz	47,26	70,12	83,48	102,12
Country	42,47	60,15	74,31	92,84
R&B	46,94	68,85	82,54	101,98
Rap	38,41	58,54	64,87	79,54
Reggae	40,51	62,48	71,12	90,26
Pop	29,05	51,96	57,96	73,47
Rock	41,98	66,21	80,25	99,09
Blues	44,14	63,88	76,12	95,87
Hip-hop	33,48	54,63	62,15	77,32

У цьому дослідженні вхідними даними виступають по 20 MP3-файлів кожного із наступних стилів музики: Classic, Jazz, Country, R&B, Rap, Reggae, Pop, Rock, Blues, Hip-hop. Результуючі утворення P-S-N-R оцінені опісля вставки прихованого сповіщення, розміром близько 1 Мбайт задля 10 різних стилів музики. Приховане сповіщення приховане поза поміччю розробленої модифікування і трьох традиційних алгоритмів L-S-B – 4-L-S-B, 2- L-S-B та 1-L-S-B. Значення показані у таблиці 1.2 і на рисунку 1.20.

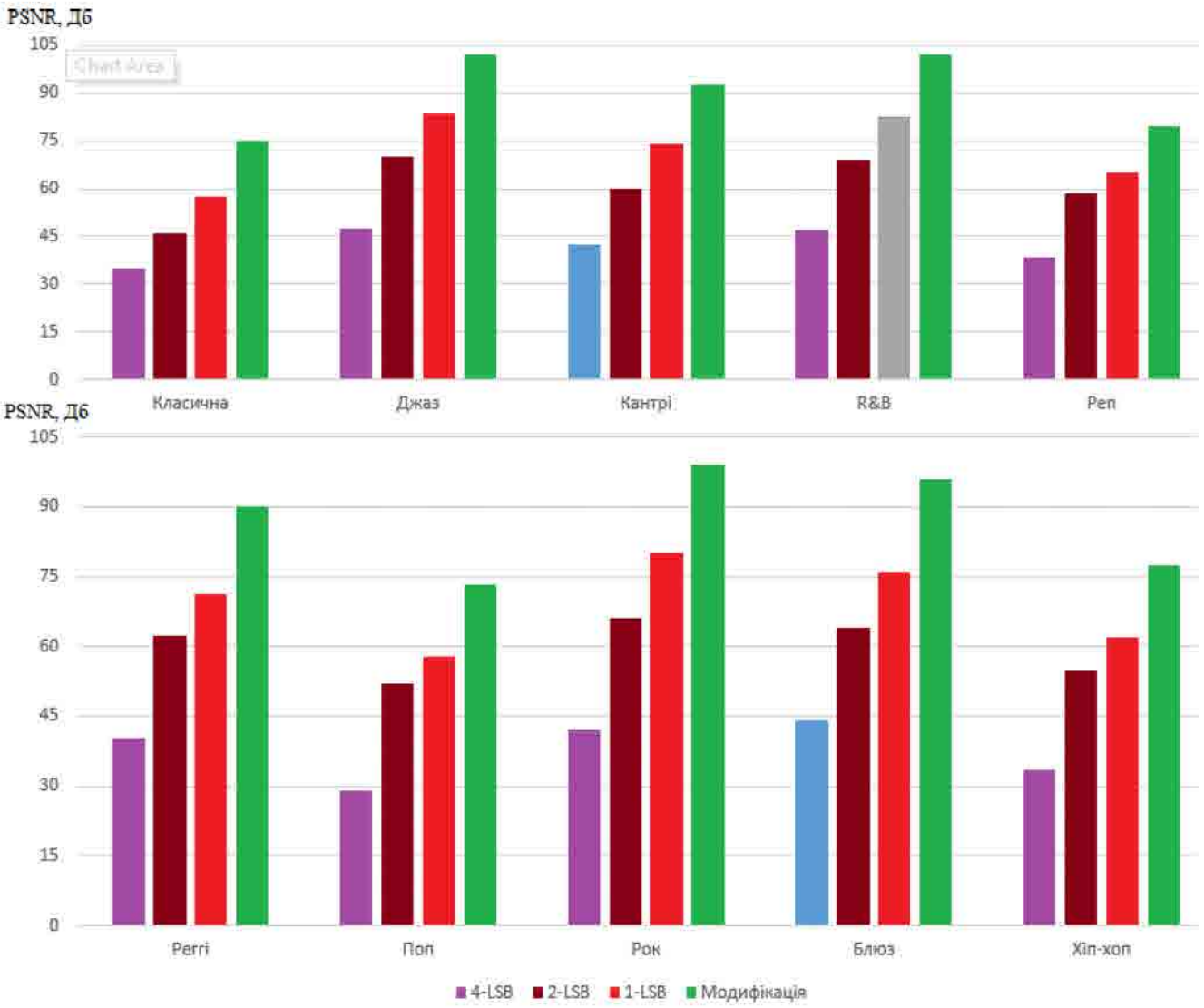


Рисунок 1.20. Результуючі утворення P-S-N-R задля обраних алгоритмів

Можливо чітко бачити, що значення P-S-N-R пропонованого методу краще, чим традиційні L-S-B задля всіх жанрів. Як наведено вище, Jazz показує найвищий P-S-N-R, оскільки він є одним із MP3-файлів, котрий містить більш високий рівень шуму, чим інші звукові файли.

В наведеній нижче таблиці 1.3 показаний відсоток покращення поміж поточним методом і традиційними методами задля попередніх вхідних інформації на основі отриманих інформації на попередньому кроці.

1.14.2 Значення додавання A-W-G-N-нападу

В цій частині дослідження застосовується один із видів нападу – адитивний гаусовий білий шум (A-W-G-N) додається до стегано-контенерів, тобто задля всіх вихідних інформації, котрі були отримані модифікованим алгоритмом, перш чим

витягати із них приховане сповіщення. Потім сповіщення витягується і здійснюється зважування його оцінки P-S-N-R із значенням цієї оцінки стегано-боксів до використання нападу.

Таблиця 1.3. Перевага модифікованого алгоритму відносно традиційних

Стиль мелодії	*Відношення P-S-N-R 4-L-S-B до модифікування, %*	*Відношення P-S-N-R 2-L-S-B до модифікування, %*	*Відношення P-S-N-R 1-L-S-B до модифікування, %*
Classic	53,26	39,13	23,52
Jazz	53,72	31,34	18,25
Country	54,25	35,21	19,96
R&B	53,97	32,49	19,06
Rap	51,71	26,40	18,44
Reggae	55,12	30,78	21,21
Pop	60,46	29,28	21,11
Rock	57,63	33,18	19,01
Blues	53,96	33,37	20,60
Hip-hop	56,70	29,35	19,62

Атака AWGN додана до всіх стегано-боксів, котрі отримали секретні дані розробленим модифікованим алгоритмом із різними значеннями шумової розкиду. Таблиця 1.4 показує отримані утворення P-S-N-R задля стегано-боксів опісля нападу із значенням розкиду 0,1 бітів/сек/Гц задля всієї смуги кожного стегано-боксу.

Таблиця 1.4. Значення опісля нападу до стегано-боксів із значенням розкиду 0.1

Стиль мелодії	*P-S-N-R до нападу, DB*	*Дисперсія, бітів/сек/Гц*	*P-S-N-R опісля нападу, DB*	*Погіршення, %*
Classic	75,21	0,1	69,76	7,25
Jazz	102,12	0,1	95,33	6,65
Country	92,84	0,1	86,83	6,47
R&B	101,98	0,1	97,92	3,98
Rap	79,54	0,1	75,32	5,31
Reggae	90,26	0,1	85,56	5,21
Pop	73,47	0,1	70,52	4,02
Rock	99,09	0,1	93,82	5,32
Blues	95,87	0,1	92,56	3,45
Hip-hop	77,32	0,1	73,57	4,85

В таблиці 1.5 показані досягнуті утворення P-S-N-R задля стегано-боксів опісля нападу із значенням розкиду 0,3 бітів/сек/Гц задля всієї смуги кожного стегано- боксу.

Таблиця 1.5. Значення опісля нападу до стегано-боксів із значенням розкиду 0.3

Стиль мелодії	*P-S-N-R до нападу, DB*	*Дисперсія, біт/сек/Гц*	*P-S-N-R після нападу, DB*	*Погіршення, %*
Classic	75,21	0,3	65,92	12,35
Jazz	102,12	0,3	92,58	9,34
Country	92,84	0,3	83,33	10,24
R&B	101,98	0,3	94,70	7,14
Rap	79,54	0,3	72,41	8,96
Reggae	90,26	0,3	81,61	9,58
Pop	73,47	0,3	67,88	7,61
Rock	99,09	0,3	89,60	9,58
Blues	95,87	0,3	89,19	6,97
Hip-hop	77,32	0,3	71,05	8,11

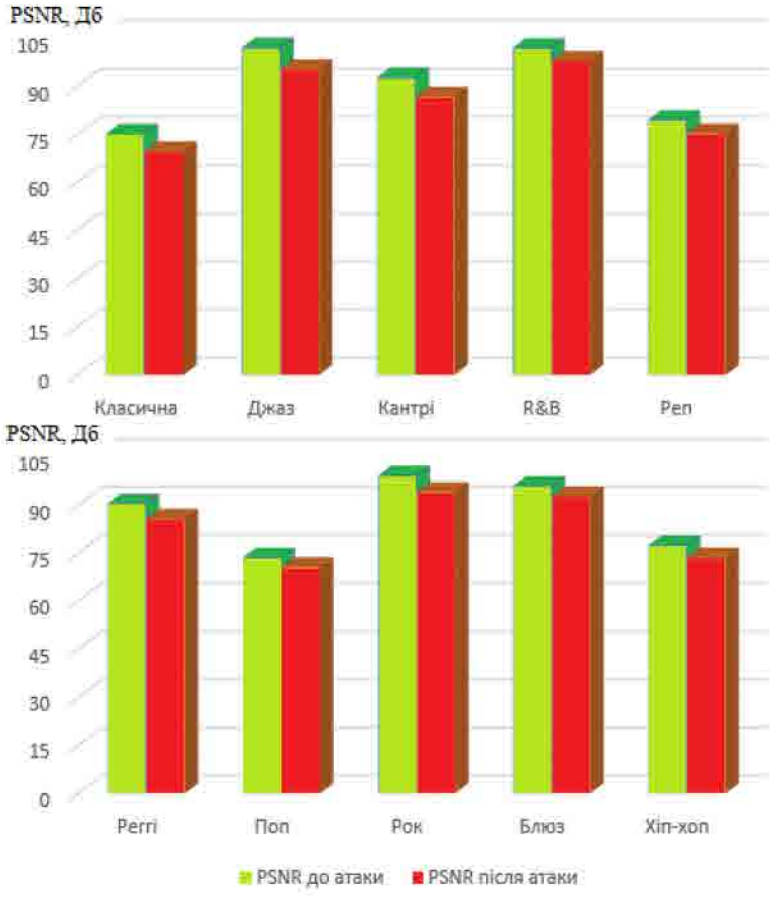


Рисунок 1.21. Зважування значень P-S-N-R до та опісля додавання нападу із значенням шумової розкиду 0,1 бітів/сек/Гц

Можливо зробити висновок про наявність очевидної деградації утворення P-S-N-R опісля додавання нападу AWGN. Як наведено у таблицях вище, деградація в значеннях P-S-N-R збільшується із збільшенням утворення розкиду шумів. Отримані значення значень P-S-N-R опісля додавання нападу AWGN із шумовою дисперсією 0,1 показані на рисунку 1.21.

Отримані значення значень P-S-N-R опісля додавання нападу AWGN із шумовою дисперсією 0,3 показані на рисунку 1.22.

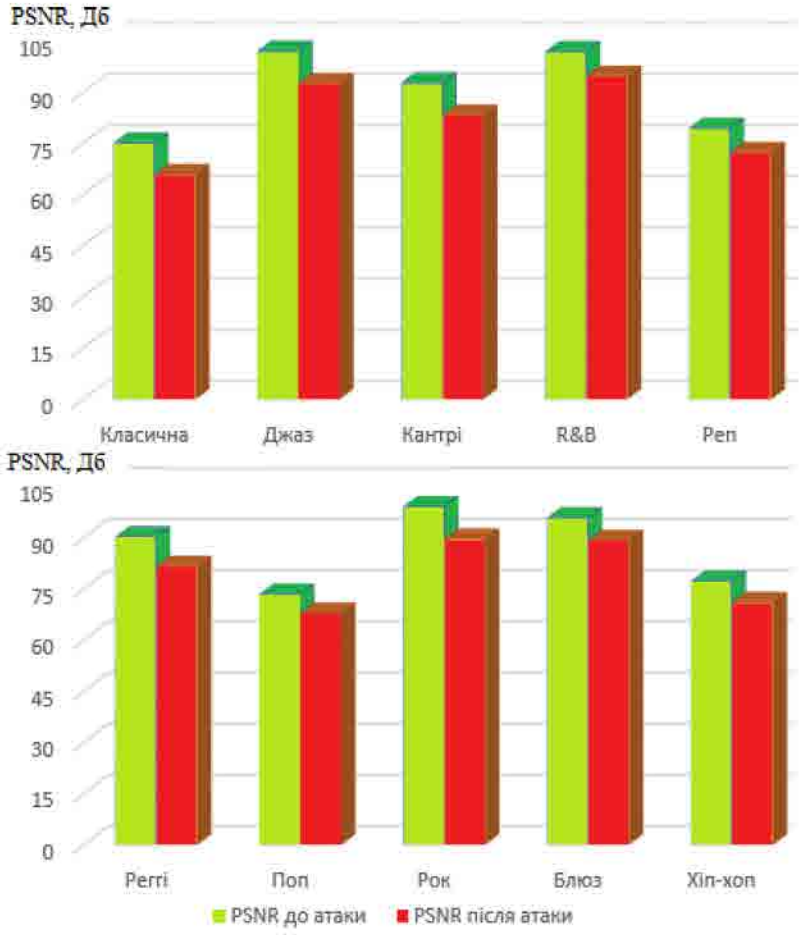


Рисунок 1.22. Зважування значень P-S-N-R до та опісля додавання нападу із значенням шумової розкиду 0,3 бітів/сек/Гц

1.14.3 Значення розрахунку показників цільності

Задля зважування стегано-боксів із вхідними файлами розроблений застосунок розраховує наступні показники: утворення контрольної суми стегано-боксу і початкового файла із видачою утворення подібності інформації в відсотках; розраховує хеш-функції стегано-боксів і вхідних інформації; зміна частот показує в відсотках зміну прихованого сповіщення, витягнутого із

стегано-бокса відносно початкового прихованого сповіщення. В наведеній нижче таблиці 1.6 представлені досягнуті відсотки задля вихідних інформації, отриманих модифікованим алгоритмом відносно вхідних інформації, до використання нападу на них.

Таблиця 1.6. Значення показників цільності задля модифікування алгоритму

Стиль мелодії	*Контрольна сума, %*	*Хеш-функція, %*	*Зміна частот, %*
Classic	100	100	100
Jazz	100	100	100
Country	100	100	100
R&B	100	100	100
Rap	100	100	100
Reggae	100	100	100
Pop	100	100	100
Rock	100	100	100
Blues	100	100	100
Hip-hop	100	100	100

Як наведено в таблиці вище, розроблена модифікування алгоритму не спотворює вихідні дані, оскільки всі показники оцінки якості отриманих інформації мають утворення 100% задля всіх стилів музики. Далі виконується процес нападу із додаванням гаусового білого шуму (A-W-G-N) до стегано-боксів у результаті розробленої модифікування алгоритму. Значення, отримані в процесі зважування стегано-боксів до нападу і опісля поза цими критеріями, показані в таблиці 1.7.

Таблиця 1.7. Значення показників цільності опісля використання нападу

Стиль мелодії	*Контрольна сума, %*	*Хеш-функція, %*	*Зміна частот, %*
Classic	21,42	92,14	50
Jazz	21,25	91,88	50
Country	21,71	92,54	50
R&B	21,14	92,17	50
Rap	20,84	91,25	50
Reggae	20,54	90,28	50
Pop	20,65	90,77	50
Rock	21,21	91,25	50
Blues	20,25	90,17	50
Hip-hop	21,17	90,14	50

Як наведено у таблиці вище, найкращий відсоток подібності стегано- боксів до нападу і опісля, демонструє перевірка хеш-функції, тоді як мінімальні досягнуті значення є задля утворення контрольної суми. Графічне відображення отриманих результатів, із таблиці 1.7 продемонстровано на рисунку 1.23.

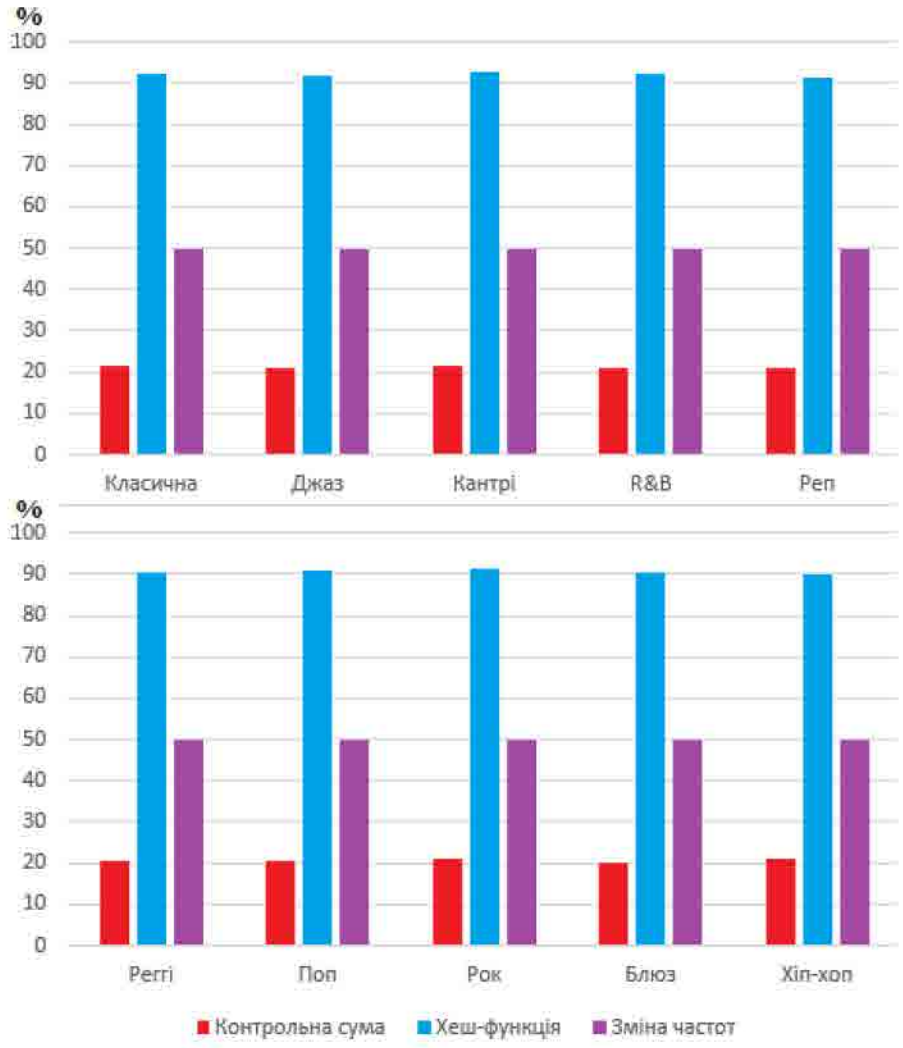


Рисунок 1.23. Показники цільності стегано-боксів опісля використання нападу

Окрім наведених вище результатів розрахунку показників цільності проведено оцінювання показників цільності задля традиційного 1-L-S-B методу. В цьому випадку атака застосовується до стегано-боксів отриманих методом 1-L-S-B. В таблиці 1.8 показані досягнуті утворення P-S-N-R задля стегано-боксів опісля нападу із значенням розкиду 0,1 бітів/сек/Гц задля всієї смуги кожного стегано- боксу. Таблиця показує, що опісля нападу, тобто додавання гаусового білого шуму (AWGN) до стегано-боксів, отриманих методом 1-L-S-B, спостерігається деградація значень P-S-N-R. Крім того, очевидно, що деградація P-S-N-R набагато більша порівняно із розробленою модифікацією алгоритму.

Таблиця 1.8. Значення опісля нападу до стегано-боксів із значенням розкиду 0.1

Стиль мелодії	*P-S-N-R до нападу, DB*	*Дисперсія, бітів/сек/Гц*	*P-S-N-R опісля нападу, DB*	*Погіршення, %*
Classic	57,52	0,1	48,34	15,96
Jazz	83,48	0,1	71,15	14,77
Country	74,31	0,1	66,14	10,99
R&B	82,54	0,1	72,58	12,07
Rap	64,87	0,1	53,74	17,16
Reggae	71,12	0,1	60,59	14,81
Pop	57,96	0,1	49,15	15,20
Rock	80,25	0,1	69,54	13,35
Blues	76,12	0,1	64,97	14,65
Hip-hop	62,15	0,1	50,85	18,18

В таблиці 1.9 показані значення, отримані в процесі зважування стегано-боксів до нападу і опісля задля техніки 1-L-S-B. Як наведено у таблиці, технологія 1-L-S-B пропонує гірші утворення показників цільності стегано-боксів опісля використання нападу на них, у порівнянні із розробленою модифікацією алгоритму.

Таблиця 1.9. Значення показників цільності опісля використання нападу задля методу 1-L-S-B

Стиль мелодії	*Контрольна сума, %*	*Хеш-функція, %*	*Зміна частот, %*
Classic	17,58	82,14	44,47
Jazz	19,58	86,88	44,47
Country	18,25	80,54	44,47
R&B	20,21	79,17	44,47
Rap	19,28	85,25	44,47
Reggae	18,17	88,28	44,47
Pop	18,25	84,77	44,47
Rock	17,85	89,25	44,47
Blues	20,85	83,17	44,47
Hip-hop	19,57	84,14	44,47

Під час тестування роботи програмного забезпечення збоїв і недоліків не виявлено, що говорить про можливість впровадження програмного продукту поза потреби.

2 РОЗДІЛ ОХОРОНИ ПРАЦІ І ТЕХНІКИ БЕЗПЕКИ

Безпека праці, спрямована на створення небезпечних та нешкідливих умов праці. На сучасному етапі розвитку виробництва вона набуває все більше важливого утворення.

Вирішення завдань охорони праці базується на досягненнях ергономіки, наукової організації праці, технічної естетики, гігієни і фізіології праці, психофізіології. Крім того, успіх охорони праці визначається темпами впровадження передової техніки, підвищення рівня механізації та автоматизації виробничих процесів, удосконаленням технології і організації виробництв

Безпека праці на підприємстві спроможно бути на належному рівні тільки тоді, коли всебічно відповідає вимогам трудового законодавства, державним стандартам України, норм та правил, розроблених задля збереження здоров'я працюючих. Важливе місце при цьому належить виконанню організаційних вимог із охорони праці, але так само трудовій і виробничій дисципліні працюючих.

В даній роботі передбачена розробка алгоритмічного і програмного забезпечення задля стегофонії. Виконання даної роботи проводилося поза поміччю персонального комп'ютера. В зв'язку із цим необхідно проаналізувати фактори ризику при роботі із сучасним персональним комп'ютером.

2.1 Аналіз небезпечних і шкідливих чинників, що впливають на працівника

Основними факторами шкідливого впливу ПК на організм людини є такі:

1. Електромагнітні поля;
2. Електромагнітні випромінювання;
3. Розгортка зображення на моніторі;
4. Мелькання зображення на екрані;
5. Тривала нерухомість пози оператора.

Сукупний вплив на людину всіх шкідливих факторів знижує загальний біоенергетичний потенціал та опірність організму, знижує імунітет, викликає

м'язову атрофію та застої у органах. Наслідки порушення норм безпеки при роботі поза ПК можуть викликати професійні захворювання чи призвести до нещасного випадку і травмування працівника.

2.2 Розробка заходів із охорони праці

Зменшити вплив перерахованих факторів ризику та зберегти здоров'я людині, яка постійно використовує у роботі ПК, сприяє дотримання всіх заходів та засобів, передбачених охороною праці.

2.2.1 Мікроклімат робочої зони працівників, вентиляція

В виробничих приміщеннях на робочих місцях із ВДТ мають забезпечуватись оптимальні утворення параметрів мікроклімату: температури, відносної вологості та рухливості повітря (ГОСТ 12.1.005-88, СН 4088-86).

Рівні позитивних та негативних іонів у повітрі приміщень із ВДТ повинні задовольняти санітарно-гігієнічним нормам № 2152-80.

2.2.2 Освітлення робочого місця, шум, вібрація

Штучне освітлення у приміщеннях із робочими місцями, обладнаними ЕОМ та ПЕОМ, містить здійснюватись системою загального рівномірного освітлення. Утворення освітленості на поверхні робочого столу у зоні розміщення документів містить становити 300 - 500 лк.

Як джерело світла при штучному освітленні застосовуються переважно люмінесцентні лампи.

Утворення освітленості на поверхні робочого столу у зоні розміщення документів містить становити 300 - 500 лк.

Система загального освітлення містить становити суцільні чи переривчасті лінії світильників, розташовані збоку з робочих місць (переважно зліва), паралельно лінії зору працюючих Використання світильників без розсіювачів і екрануючих грат заборонено.

Рівні звукового тиску у октавних смугах частот мають відповідати вимогам СН 3223-85, ГОСТ 12.1.003-83, ГР 2411-81.

Задля забезпечення допустимих рівнів шуму на робочих місцях слід застосовувати засоби звукопоглинання, При виконанні робіт із ЕОМ в виробничих приміщеннях утворення характеристик вібрації на робочих місцях не повинні перевищувати допустимі згідно СН 3044-84, ГОСТ 12.1.012-90. При розумовій праці, яка вимагає зосередженості припустимий рівень шуму становить 50дБ

2.2.3 Організація робочого місця користувача ПК

- Важливо, аби офісний працівник сидячи поза комп'ютером знаходився поза добре освітленим робочим столом. Найчастіше саме погане освітлення робочого місця надає більш згубний задля зору вплив, чим сам факт перебування поза комп'ютером.
- Робочі столи слід розміщувати так, аби монітори були орієнтовані бічною стороною до світлових прорізів, аби природне світло падало переважно ліворуч.
- При розміщенні робочих місць відстань поміж робочими столами повинна бути не менше 2,0 м, але відстань поміж бічними поверхнями відеомоніторів - не менше 1,2 м.
- Конструкція робочого столу повинна забезпечувати оптимальне розміщення на робочій поверхні використовуваного обладнання.
- Конструкція робочого стільця чи крісла повинна забезпечувати підтримку раціональної робочої пози працівника.
- Клавіатуру слід розташовувати на поверхні столу на відстані 100..300 мм з краю, зверненого до користувача, чи на спеціальній поверхні, відокремленій з основної стільниці.
- Екран відеомонітора повинен знаходитися з очей користувача на відстані 600-700 мм, але не ближче 500мм.

Рисунок 2.1. Правильне положення оператора ПК на робочому місці

Безпека праці при роботі поза комп'ютером передбачає, що тривалість безперервної роботи поза комп'ютером без регламентованої перерви не повинна перевищувати 2 години.

Не рекомендується працювати поза комп'ютером більше 6 годин поза зміну. Рекомендується робити перерви у роботі поза ПК тривалістю 10 хвилин через кожні 50 хвилин роботи. Під час регламентованих перерв доцільно виконувати комплекси вправ.

При нерегламентованій роботі підвищеної інтенсивності можливі головні болі, нервові зриви і інше.

2.3 Пожежна безпека

Протипожежна безпека на підприємстві – невіддільна частина організації робочого простору та процесів згідно із нормами чинного законодавства. Зокрема, цю сферу регламентують Правила пожежної безпеки у Україні, затверджені наказом Міністерства внутрішніх справ України, із змінами, котрі періодично вносяться відповідними наказами.

Попри обладнання будівель будь-якими типами установок пожежогасіння, пожежної сигналізації чи внутрішніми пожежними кранами, офісні приміщення так само мають бути забезпечені первинними засобами пожежогасіння.

До первинних засобів пожежогасіння належать: вогнегасники, кошма (покривало із негорючого теплоізоляційного полотна), ящики із піском, бочки із водою, пожежні відра, багри, ломи, сокири тощо. Найбільш зручними задля використання є вогнегасники.

Відповідальними поза своєчасне і повне оснащення об'єктів засобами пожежогасіння, забезпечення їх технічного обслуговування, навчання працівників правил користування ними є роботодавець і керівники структурних підрозділів.

Відповідальні особи зобов'язуються розробити протипожежний режим та інструкції відповідно до вимог, викладених у нормативних актах на зазначених їм об'єктах.

Встановлений режим включає порядки із описом місць спеціального призначення і правила їх користування і утримання, наприклад:

- евакуаційних шляхів,
- так званих «курилок»,
- місць складування продукції і сировини,
- стоянки транспорту.

Важливою складовою протипожежного режиму на будь-якому об'єкті є розробка та впровадження порядку дій при виникненні пожежі. Неодмінно містить бути план евакуації, описано, як повинні відключатися електроустановки, що та у якій послідовності необхідно робити співробітникам.

ВИСНОВКИ

В випускній кваліфікаційній роботі проведено дослідження алгоритмів стегофонії задля аудіо-файлів різних музичних жанрів, зокрема Jazz, R&B, Rap, Pop, Rock. При цьому виконано модифікацію алгоритму мінімально значущого біту (L-S-B). Дана модифікація показала кращі значення чим традиційні алгоритми L-S-B, але так само продемонструвала вищий рівень стійкості до нападу із додаванням адитивного гаусового білого шуму.

Під час виконання роботи було проаналізовано предметну галузь і розглянуто існуючі алгоритми стегофонії, проаналізовано їх переваги і недоліки, проведене їх зважування. Було проведено аналіз зменшення внесення помітних змін до файлів-боксів, і щодо збільшення стійкості боксів до атак.

На основі проведеного аналізу було вирішено розробляти модифікацію саме методу мінімально значущого біту через те, що цей алгоритм забезпечує більшу безпеку і є ефективним способом скриття секретної інформації з хакерів та відправлення в пункт призначення безпечним і невиявленим способом. Так само алгоритм гарантує, що розмір файла не змінюється навіть опісля кодоутворення та так само підходить задля будь-якого типу формату аудіо-файлів. Він сприяє приховувати в файлах-контейнерах набагато більший об'єм секретної інформації в порівнянні із іншими алгоритмами.

При виконанні аналізу ефективності алгоритмів стегофонії використовувались традиційні алгоритми 4-L-S-B, 2-L-S-B та 1-L-S-B і розроблену модифікацію. Задля перевірки ефективності розробленого алгоритму було проведено дослідження на аудіо-файлах різних жанрів, різного розміру. Задля цього був розроблений відповідний стеганофонічний застосунок мовою програмування C#. Так само проведено імітацію нападу на аудіо-контейнери задля перевірки їх стійкості в порівнянні із традиційними методами мінімально значущого біту. При порівнянні із традиційним алгоритмом L-S-B наведено, що запропонована модифікація алгоритму L-S-B містить кращі значення ефективності.

ПЕРЕЛІК ВИКОРИСТАНИХ ІНФОРМАЦІЙНИХ ДЖЕРЕЛ

1. Конахович Г.Ф., Пузиренко А.Ю. Комп'ютерна стеганографія. Теорія та практика. // Київ: МК-Пресс., 2006 р. – 288 с.
2. Поліновський В.В. Інформаційна технологія для досліджень методів стеганографії і стеганоаналізу // Комп'ютерно-інтегровані технології: освіта, наука, виробництво. – 2011. – №5. – С. 236–242.
3. Кошкіна Н.В. Новий метод цифрових водяних знаків для аудіосигналів // Матеріали 1 Міжнар. наук.-техн. конф. «Захист інформації і безпека інформаційних систем». – Львів. – 2012. – 120–121.
4. Кузнецов О.О. Стеганографія: навчальний посібник – Х. : Вид. ХНЕУ, 2011. – 232 с.
5. Хорошко В.О. Основи комп'ютерної стеганографії: навч. посібн. для студентів і аспірантів – Вінниця : ВДТУ, 2003. – 143 с.
6. Задірака В.К. Ефективні алгоритми побудови стегоконтейнерів з використанням швидкого перетворення Фур'є – Праці міжнар. конф. "Питання оптимізації обчислень-XXXII". – Київ: Інститут кібернетики ім. В.М. Глушкова НАН України, 2005. –76-78 с.
7. Коноваленко І.В. Програмування мовою C# 6.0: навч. посіб. – Тернопіль, ТНТУ-2016 – 229с.
8. Кузьміних В. О. Управління версіями програмних засобів проекту: Навчальний посібник – КПІ ім. Ігоря Сікорського, 2023.
9. Цибульник С. О., Барандич К. С. Технології розроблення програмного забезпечення: Навчальний посібник – КПІ ім. Ігоря Сікорського, 2022.
10. К.Т. Кузьма, В.О. Поздєєв. Основи об'єктно-орієнтованого програмування мовою C#: Навчальний посібник – МНУ, 2022.
11. Огляд методів рішень аудіо-стеганографії [Електронний ресурс]: https://sibac.info/studconf/tech/xlii/54331.
12. C# Documentation: [Електронний ресурс]: https://learn.microsoft.com/uk-ua/dotnet/csharp/ (англійською мовою).

ДОДАТОК А. Лістинг основних класів стеганофонічного застосунку мовою C#

```
using System;
using System.Collections.Generic;
using System.IO;
using System.Runtime.Serialization;
using System.Text;
using Steganography.Media;

namespace Steganography
{
   [Serializable]

   //у цьому класі приховуємо та витягаємо повідомлення, що шифрується
   class HideAndExtract
   {
      private WavAudio _file;
      private List<byte> _bits;

      //ініціалізуємо об'єкт класу за допомогою класу WavAudio
      public HideAndExtract(WavAudio file)
      {
         _file = file;
      }

      //приховуємо повідомлення у лівому та правому потоках аудіо-файлу
      public void HideMessage(string message)
      {
         //отримуємо канали з файла WaveAudio
         List<short> leftStream = _file.GetLeftStream();
         List<short> rightStream = _file.GetRightStream();

         //приховуємо повідомлення у потоках

         //перетворюємо повідомлення у масив байтів
         byte[] bufferMessage = Encoding.UTF8.GetBytes(message);
         short tempBit;
         //місце - індекс, який буде йти за повідомленням
         int bufferIndex = 0;
         //довжина повідомлення
         int bufferLength = bufferMessage.Length;
                  //довжина аудіо (довжина лівого каналу дорівнює довжині правого каналу)
         int channelLength = leftStream.Count;
                  //блок збереження повідомлення. Це значення дорівнює 1. Якщо воно
                  //не дорівнює 1, то довжина повідомлення, що шифрується, більше,
                  //ніж довжина початкового аудіофайлу
                   int storageBlock = (int)Math.Ceiling((double)bufferLength / (channelLength * 2));
```

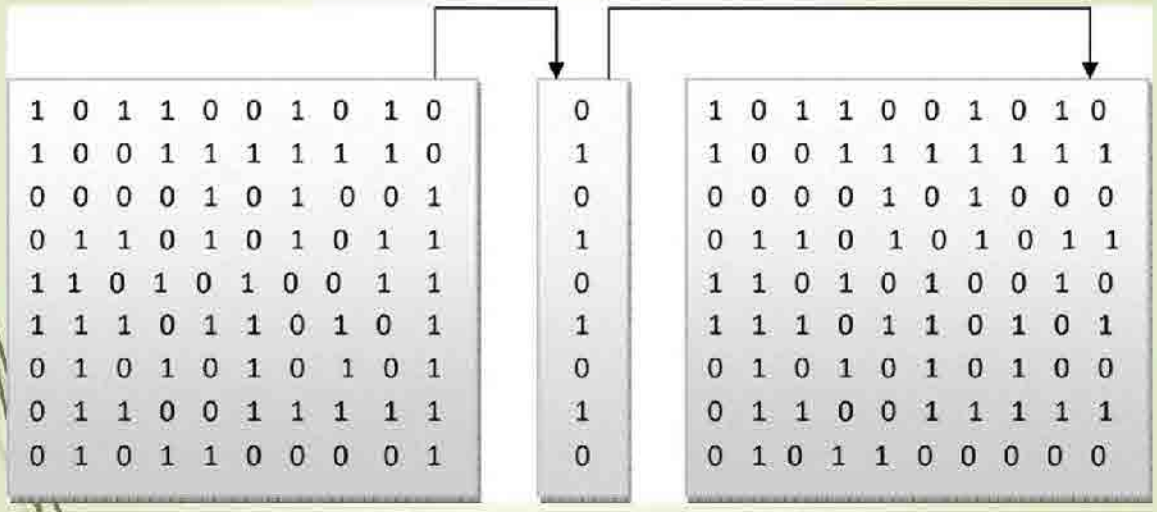
```
    //якщо довжина повідомлення більше, ніж довжина аудіоканалу
    if (bufferLength > channelLength)
        throw new Exception();

            //зберігаємо інформацію про довжину повідомлення, що шифрується,
            //у перших елементах лівого і правого потоків аудіо-файлу

            //беремо цілу частину розміру повідомлення і записуємо першим
            //елементом у лівому каналі
    leftStream[0] = (short)(bufferLength / 32767);
            //беремо залишок розміру повідомлення і записуємо першим
            //елементом у правому каналі
    rightStream[0] = (short)(bufferLength % 32767);
    var countBufferMessage = 0;
            //йдемо по довжині потоку, починаючи з 1, тому що у [0]
            //зберігається довжина повідомлення; зберігаємо біт повідомлення
            //у лівий і правий потоки
    for (int i = 1; i < leftStream.Count && countBufferMessage < bufferMessage.Length; i++)
    {
            //беремо залишок від ділення на 8, тому що працюємо з бітами;
            //також працюємо з цифрою 7, тому що діапазон складає [0..7]
        if (bufferIndex < bufferLength && i % 8 > 7 - storageBlock && i % 8 <= 7)
        {
            //отримуємо біт повідомлення
            tempBit = bufferMessage[bufferIndex++];
            //замінюємо біт аудіоданих бітом повідомлення
            leftStream.Insert(i, tempBit);
            leftStream[i] = tempBit;
            countBufferMessage++;
        }

        if (bufferIndex < bufferLength &&  i % 8 > 7 - storageBlock && i % 8 <= 7)
        {
            tempBit = bufferMessage[bufferIndex++];
            rightStream.Insert(i, tempBit);
            rightStream[i] = tempBit;
            countBufferMessage++;
        }
    }
            //у потоках тепер є введене повідомлення. Оновлюємо потоки
            //початкового WAV-файлу
    _file.UpdateStreams(leftStream, rightStream);
}

public string ExtractMessage()
{
    if (bufferLength > channelLength)
        throw new Exception();

    //отримуємо канали з файлу WaveAudio
```

```
            List<short> leftStream = _file.GetLeftStream();
            List<short> rightStream = _file.GetRightStream();

                //вилучаємо повідомлення з потоків і відображуємо у
                //відповідному полі

            //індекс у повідомленні, що шифрується
            int bufferIndex = 0;
            //ціла частина довжини повідомлення, що шифрується
            int messageLengthQuotient = leftStream[0];
            //дробова частина довжини повідомлення, що шифрується
            int messageLengthRemainder = rightStream[0];
            //довжина аудіоканалів
            int channelLength = leftStream.Count;

                //обчислюємо початкову довжину повідомлення, що шифрується
                int bufferLength = 32767 * messageLengthQuotient + messageLengthRemainder;
                 //блок збереження повідомлення. Це значення дорівнює 1. Якщо
                 //воно не дорівнює 1, то довжина повідомлення, що шифрується,
                 //більше, ніж довжина початкового аудіо-файлу
                 int storageBlock = (int)Math.Ceiling((double)bufferLength / (channelLength * 2));

            //створюємо повідомлення - масив байт
            byte[] bufferMessage = new byte[bufferLength];
            //йдемо по довжині потоку, починаючи з 1, тому що у [0]
                //зберігається довжина повідомлення; зберігаємо біт повідомлення
                //у лівий і правий потоки
            for (int i = 1; i < leftStream.Count; i++)
            {
                //беремо залишок від ділення на 8, тому що працюємо з
                //бітами; також працюємо з цифрою 7, тому що діапазон [0..7]
            if (bufferIndex < bufferLength && i % 8 > 7 - storageBlock && i % 8 <= 7)
               {
                 //отримуємо біти повідомлення з лівого та правого каналів
                  bufferMessage[bufferIndex++] = (byte)leftStream[i];
                  if (bufferIndex < bufferLength)
                     bufferMessage[bufferIndex++] = (byte)rightStream[i];
               }
            }
            //перетворюємо масив байтів у рядкове повідомлення
                //і повертаємо його
            return Encoding.UTF8.GetString(bufferMessage);
        }
    }
}
```

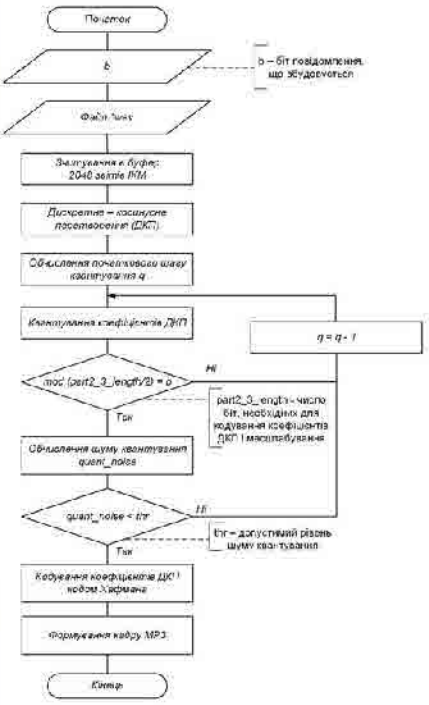
ДОДАТОК Б

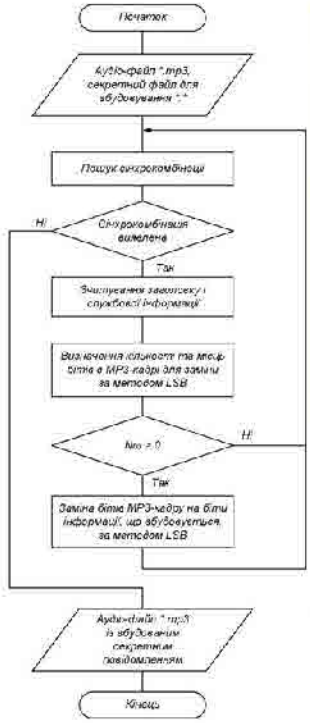
Слайди мультимедійної презентації

Метод LSB для стеганофонії



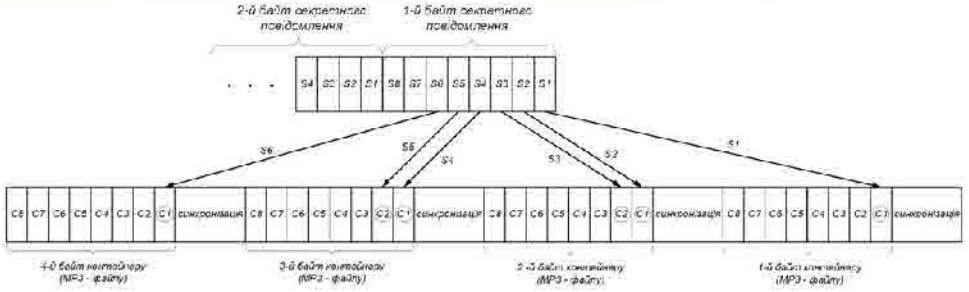
Алгорим стиснення звукової інформації





Блок-схема алгоритму приховування даних у аудіо-файлі

Модифікація методу LSB



Набір вхідних даних

Найменування жанру	*Час, сек.*	*Розмір файлу (WAV), МВ*	*Розмір файлу (320 кбіт/сек), МВ*	*Розмір файлу (256 кбіт/сек), МВ*	*Розмір файлу (192 кбіт/сек), МВ*	*Розмір файлу (128 кбіт/сек), МВ*	*Розмір файлу (96 кбіт/сек), МВ*
Класична	2:42	14.4	6.54	5.24	3.94	2.62	1.97
Джаз	2:56	15.4	7.01	5.60	4.21	2.81	2.11
Кантрі	3:11	16.5	7.51	6.00	4.51	3.01	2.26
R&B	3:15	16.9	7.68	6.15	4.62	3.08	2.32
Реп	3:24	17.4	7.90	6.33	4.76	3.17	2.38
Реггі	3:42	18.2	8.27	6.62	4.98	3.32	2.49
Поп	3:53	19.1	8.68	6.95	5.22	3.48	2.62
Рок	4:04	20.3	9.22	7.38	5.55	3.70	2.78
Блюз	4:12	21.1	9.59	7.67	5.77	3.85	2.89
Хіп-хоп	4:27	22.7	10.31	8.25	6.21	4.14	3.11

Оцінка спотворення вхідного файлу

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(X(i) - Y(i))^2$$

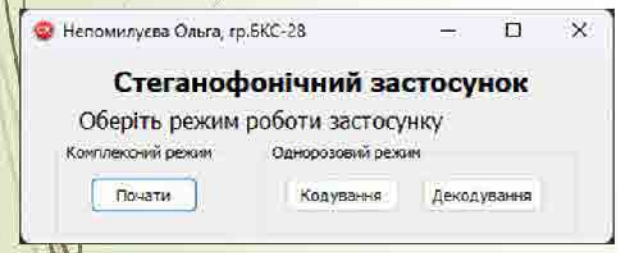
$$PSNR = 10\lg\frac{(MAX)^2}{MSE}$$

де X – оригінальний об'єкт, Y – це стего-об'єкт,

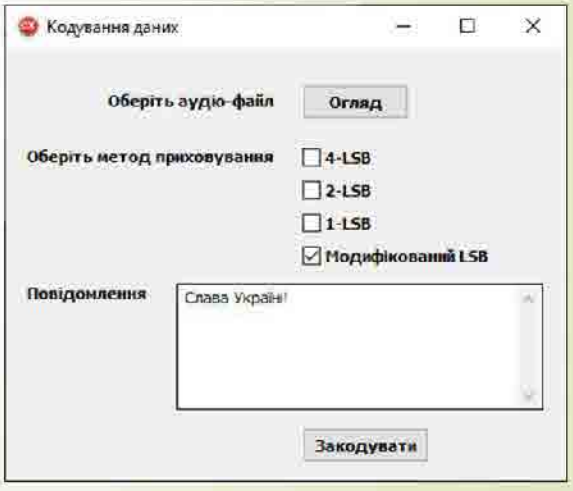
N – розмір обкладинки,

MAX – максимальне значення амплітуди вхідного аудіо-файлу.

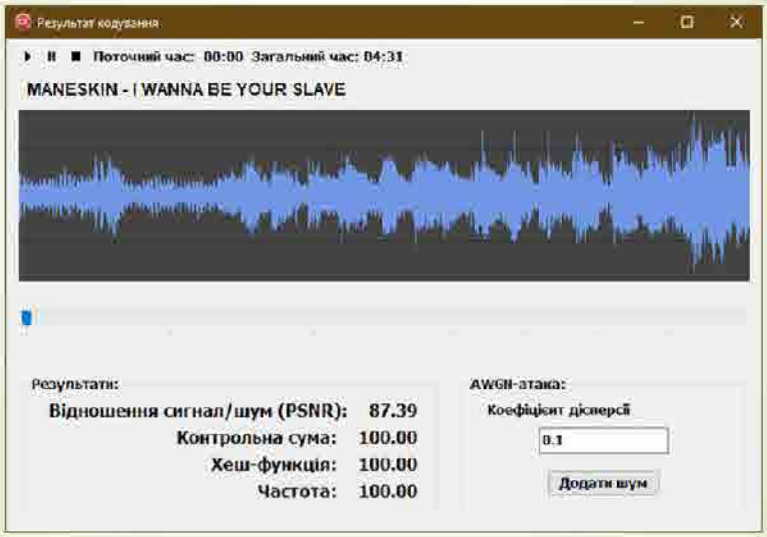
Головне вікно стеганофонічного застосунку



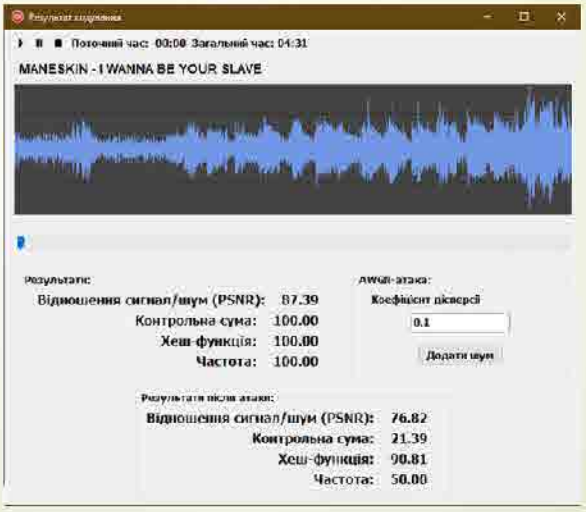
Вікно кодування даних для одного файлу



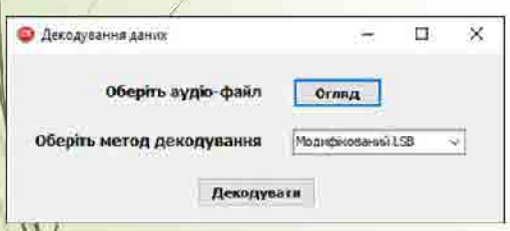
Вікно з результатами приховування даних



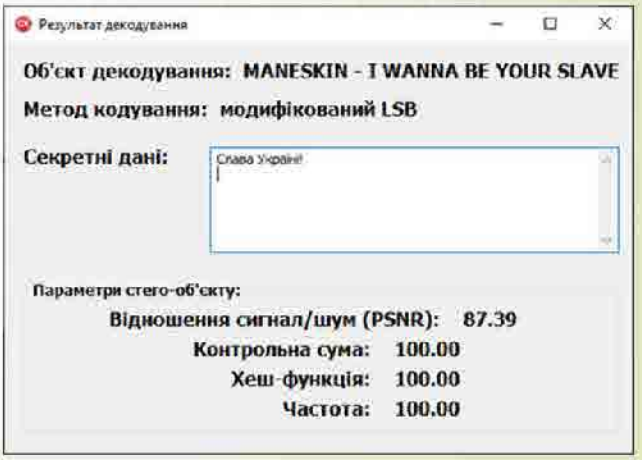
Видозмінене вікно результатів кодування після додавання атаки на стеганоконтейнер



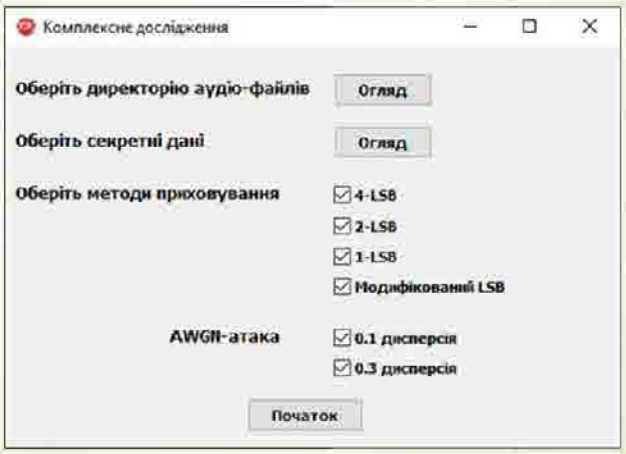
Вікно декодування даних зі стеганоконтейнеру



Результат декодування даних зі стегано-контейнеру



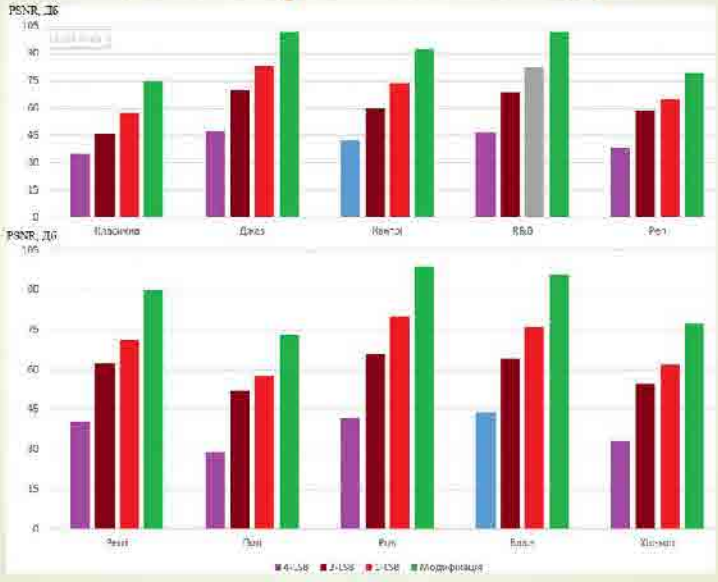
Вікно налаштування комплексного дослідження стегофонічних методів



Результуючі значення PSNR для обраних методів

Стиль мелодії	*PSNR 4-LSB, Дб*	*PSNR 2-LSB, Дб*	*PSNR 1-LSB, Дб*	*PSNR модифікації LSB, Дб*
Класична	35,15	45,78	57,52	75,21
Джаз	47,26	70,12	83,48	102,12
Кантрі	42,47	60,15	74,31	92,84
R&B	46,94	68,85	82,54	101,98
Реп	38,41	58,54	64,87	79,54
Регті	40,51	62,48	71,12	90,26
Поп	29,05	51,96	57,96	73,47
Рок	41,98	66,21	80,25	99,09
Блюз	44,14	63,88	76,12	95,87
Хіп-хоп	33,48	54,63	62,15	77,32

Результуючі значення PSNR для обраних методів



Перевага модифікованого алгоритму відносно традиційних

Стиль мелодії	*Відношення PSNR 4-LSB до модифікації, %*	*Відношення PSNR 2-LSB до модифікації, %*	*Відношення PSNR 1-LSB до модифікації, %*
Класична	53,26	39,13	23,52
Джаз	53,72	31,34	18,25
Кантрі	54,25	35,21	19,96
R&B	53,97	32,49	19,06
Реп	51,71	26,40	18,44
Регті	55,12	30,78	21,21
Поп	60,46	29,28	21,11
Рок	57,63	33,18	19,01
Блюз	53,96	33,37	20,60
Хіп-хоп	56,70	29,35	19,62

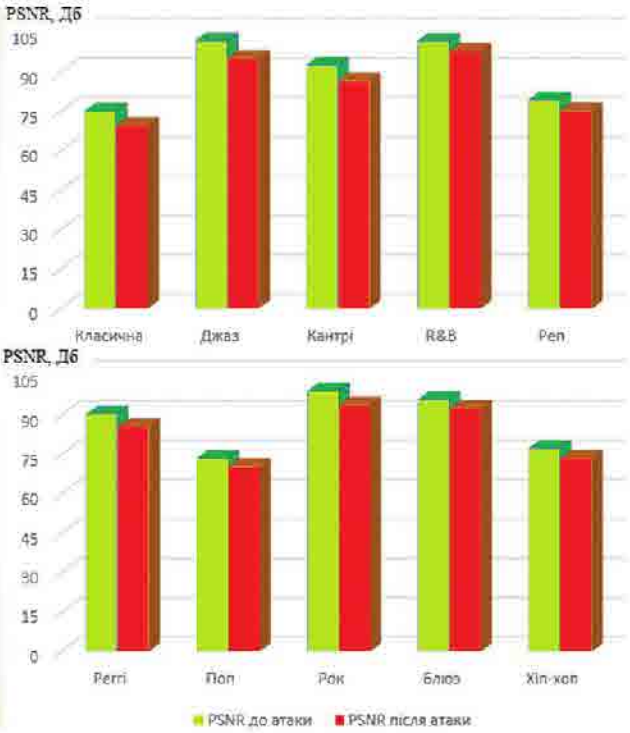
Результати після атаки до стегано-контейнерів зі значенням дисперсії 0.1

Стиль мелодії	*PSNR до атаки, Дб*	*Дисперсія, біт/сек/Гц*	*PSNR після атаки, Дб*	*Погіршення, %*
Класична	75,21	0,1	69,76	7,25
Джаз	102,12	0,1	95,33	6,65
Кантрі	92,84	0,1	86,83	6,47
R&B	101,98	0,1	97,92	3,98
Реп	79,54	0,1	75,32	5,31
Регті	90,26	0,1	85,56	5,21
Поп	73,47	0,1	70,52	4,02
Рок	99,09	0,1	93,82	5,32
Блюз	95,87	0,1	92,56	3,45
Хіп-хоп	77,32	0,1	73,57	4,85

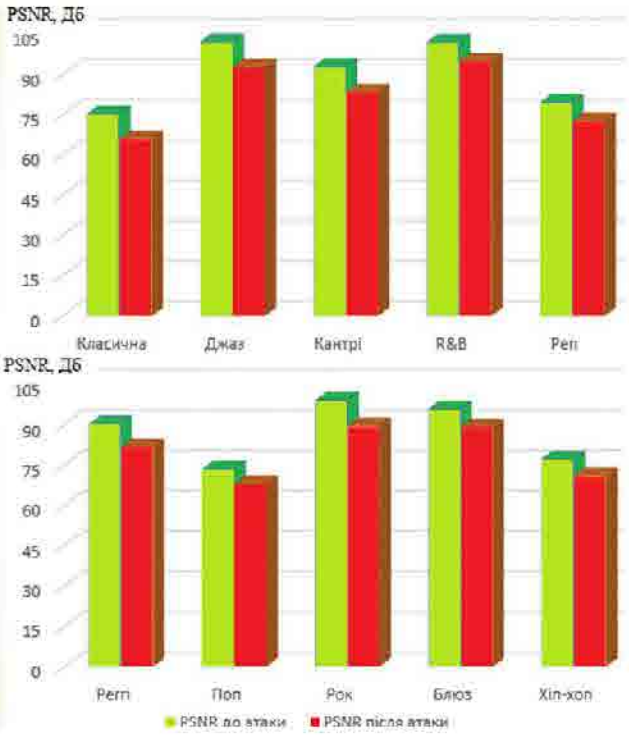
Результати після атаки до стегано-контейнерів зі значенням дисперсії 0.3

Стиль мелодії	*PSNR до атаки, Дб*	*Дисперсія, біт/сек/Гц*	*PSNR після атаки, Дб*	*Погіршення, %*
Класична	75,21	0,3	65,92	12,35
Джаз	102,12	0,3	92,58	9,34
Кантрі	92,84	0,3	83,33	10,24
R&B	101,98	0,3	94,70	7,14
Реп	79,54	0,3	72,41	8,96
Реггі	90,26	0,3	81,61	9,58
Поп	73,47	0,3	67,88	7,61
Рок	99,09	0,3	89,60	9,58
Блюз	95,87	0,3	89,19	6,97
Хіп-хоп	77,32	0,3	71,05	8,11

Порівняння значень PSNR до і після додавання атаки зі значенням шумової дисперсії 0.1 біт/сек/Гц



Порівняння значень PSNR до і після додавання атаки зі значенням шумової дисперсії 0.3 біт/сек/Гц

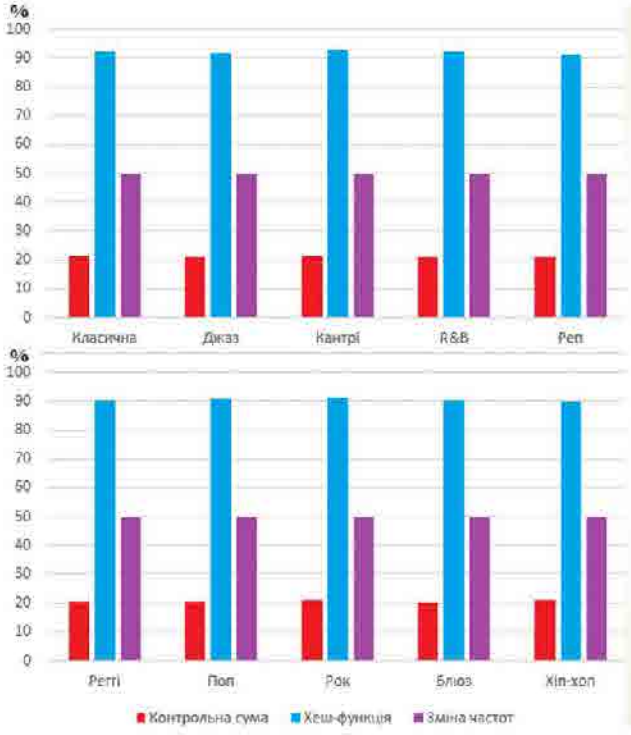


Результати показників цілісності для модифікованого алгоритму LSB

Стиль мелодії	*Контрольна сума, %*	*Хеш-функція, %*	*Зміна частот, %*
Класична	100	100	100
Джаз	100	100	100
Кантрі	100	100	100
R&B	100	100	100
Реп	100	100	100
Реггі	100	100	100
Поп	100	100	100
Рок	100	100	100
Блюз	100	100	100
Хіп-хоп	100	100	100

Результати показників цілісності після застосування атаки

Стиль мелодії	*Контрольна сума, %*	*Хеш-функція, %*	*Зміна частот, %*
Класична	21,42	92,14	50
Джаз	21,25	91,88	50
Кантрі	21,71	92,54	50
R&B	21,14	92,17	50
Реп	20,84	91,25	50
Регті	20,54	90,28	50
Поп	20,65	90,77	50
Рок	21,21	91,25	50
Блюз	20,25	90,17	50
Хіп-хоп	21,17	90,14	50



Показники цілісності стеганоконтейнерів після застосування атаки

Результати після атаки до стеганоконтейнерів зі значенням дисперсії 0.1

Стиль мелодії	*PSNR до атаки, Дб*	*Дисперсія, біт/сек/Гц*	*PSNR після атаки, Дб*	*Погіршення, %*
Класична	57,52	0,1	48,34	15,96
Джаз	83,48	0,1	71,15	14,77
Кантрі	74,31	0,1	66,14	10,99
R&B	82,54	0,1	72,58	12,07
Реп	64,87	0,1	53,74	17,16
Регті	71,12	0,1	60,59	14,81
Поп	57,96	0,1	49,15	15,20
Рок	80,25	0,1	69,54	13,35
Блюз	76,12	0,1	64,97	14,65
Хіп-хоп	62,15	0,1	50,85	18,18

Результати показників цілісності після застосування атаки для методу 1-LSB

Стиль мелодії	*Контрольна сума, %*	*Хеш-функція, %*	*Зміна частот, %*
Класична	17,58	82,14	44,47
Джаз	19,58	86,88	44,47
Кантрі	18,25	80,54	44,47
R&B	20,21	79,17	44,47
Реп	19,28	85,25	44,47
Регті	18,17	88,28	44,47
Поп	18,25	84,77	44,47
Рок	17,85	89,25	44,47
Блюз	20,85	83,17	44,47
Хіп-хоп	19,57	84,14	44,47

ВСП «Одеський технічний фаховий коледж ОНТУ»

ВІДГУК

керівника про кваліфікаційну роботу бакалавра

Непомилуєвої Ольги Петрівни

(прізвище, ім`я та по батькові)

Спеціальність *123 "Комп'ютерна інженерія"*

Тема кваліфікаційної роботи *Дослідження методів стеганофонії для аудіо-файлів різних музичних жанрів*

ХАРАКТЕРИСТИКА КВАЛІФІКАЦІЙНОЇ РОБОТИ

а) Обсяг і якість виконання роботи (графічного матеріалу і розрахунково-пояснювальної записки) *Випускна робота виконана відповідно технічному завданню. Пояснювальна записка до випускної роботи містить 74 сторінки. У пояснювальній записці описані методи стеганофонії та їх застосування для приховування інформації у аудіо-файлах різних музичних жанрів, визначена ефективність окремих алгоритмів та їх стійкість до атак. Виконано модифікацію методу LSB для покращення його властивостей. Графічна частина складається з окремих слайдів, оформлених у вигляді презентації, передбачених технічним завданням. Якість виконання пояснювальної записки та слайдів добра, розробку виконано у повному обсязі.*

б) Самостійність роботи

Протягом виконання випускної бакалаврської роботи Непомилуєва Ольга поступово та послідовно виконувала всі етапи, проявляла ініціативу у створенні загальної концепції та реалізації технічного завдання. Всі роботи вона виконувала самостійно, з оглядом на рекомендації керівника.

в) Теоретична підготовка здобувача освіти

Непомилуєва Ольга під час роботи над випускною бакалаврською роботою вивчила і опрацювала достатню кількість літературних джерел за даною тематикою.

Вважаю, що теоретична підготовка здобувачки освіти достатня і вона готова до захисту роботи.

г) Вміння розв`язувати виробничі і конструкторські питання на базі останніх досліджень науки і техніки, передових методів виробництва

Під час виконання роботи Непомилуєва Ольга мала змогу самостійно приймати окремі рішення з виконання програмної частини роботи та показала вміння організовано працювати над поставленою задачею, складати та оформлювати презентацію проекту, користуючись сучасними комп'ютерними програмними засобами, такими як Microsoft Visual Studio, Microsoft PowerPoint, Microsoft Visio.

Оцінка розрахункової частини *Добре*

Оцінка графічної частини *Відмінно*

Загальна оцінка *Добре*

Прізвище, ім`я, по батькові *Кривченко Юрій Вікторович*

Місце роботи і посада керівника роботи *ВСП "Одеський технічний фаховий коледж ОНТУ", викладач спецдисциплін комісії комп'ютерних технологій та програмної інженерії, голова циклової комісії КТ та ПІ*

Підпис

«13» червня 2024 р.

ВСП «ОДЕСЬКИЙ ТЕХНІЧНИЙ ФАХОВИЙ КОЛЕДЖ ОНТУ»

РЕЦЕНЗІЯ

на кваліфікаційну роботу бакалавра здобувача освіти

відділення комп'ютерних систем

Непомилуєвої Ольги Петрівни

(прізвище, ім`я та по батькові)

Спеціальність 123 "Комп'ютерна інженерія"

Освітня програма «Комп'ютерна інженерія»

Керівник дипломного проекту (роботи) Кривченко Юрій Вікторович

(прізвище, ім`я та по батькові)

Тема дипломного проекту (роботи) Дослідження методів стеганофонії для аудіо-файлів різних музичних жанрів

Обсяг розрахунково-пояснювальної записки 74 сторінок

Обсяг графічної (презентаційної) частини 25 аркушів (слайдів)

ХАРАКТЕРИСТИКА ДИПЛОМНОГО ПРОЕКТУ (РОБОТИ)

а) заключення про ступінь відповідності виконаного дипломного проекту (роботи) завданню

Представлена на рецензію кваліфікаційна робота бакалавра повністю відповідає меті проектування та технічному завданню. Тематика кваліфікаційної роботи є актуальною для своєї галузі та присвячена дослідженню методів стеганофонії для аудіо-файлів різних музичних жанрів.

б) характеристика виконання кожного розділу дипломного проекту (роботи)

Кваліфікаційна робота складається зі вступу, двох розділів, висновків, переліку використаних джерел. У основному розділі розглянуто модель стеганографічної системи, способи приховування даних, огляд методів стеганофонії, кодування за алгоритмом LSB, складено математичну модель, підібрані вхідні дані, виконано модифікацію методу LSB, реалізовано програмний продукт для стеганофонії, проаналізовано ефективність методів стеганофонії.

в) оцінка якості виконання пояснювальної записки та графічної частини дипломного проекту (роботи) Графічна частина виконана на достатньо високому рівні у вигляді презентації із використанням офісного пакету Microsoft PowerPoint та Visio. Пояснювальна записка виконана охайно та у відповідності до норм оформлення документів із використанням офісного пакету Microsoft Word. Загальна якість виконання документації – добра, академічного плагіату у роботі не виявлено.

г) перелік позитивних якостей дипломного проекту (роботи)

1. Детально описано мету та цілі аналізу;
2. Проведено серію експериментів для дослідження методів стеганофонії для аудіо-файлів різних музичних жанрів;
3. Виконано побудову графіків, що візуально відображують результати роботи.

д) основні недоліки дипломного проекту (роботи)

1. Бажано було передбачити у програмі перевірку коректності вхідних даних;
2. У роботі варто було більш детально описати отримані результати дослідження.

Оцінка розрахункової частини Відмінно

Оцінка графічної частини Відмінно

Загальна оцінка Відмінно

Прізвище, ім'я, по батькові рецензента к.т.н. Селіванова Алла Віталіївна

Місце роботи і посада рецензента Одеський національний технологічний університет, декан факультету комп'ютерної інженерії, програмування та кіберзахисту

Підпис:

«17» 06 2024 р.

UNICHECK
by Turnitin

Ім'я користувача:
Катерина Григорівна Краснокутська

ID перевірки:
1016226283

Дата перевірки:
03.05.2024 13:13:47 EEST

Тип перевірки:
Doc vs Internet + Library

Дата звіту:
03.05.2024 13:17:00 EEST

ID користувача:
100011688

Назва документа: **2БКС-28_Непомилуєва_Ольга**

Кількість сторінок: **52** Кількість слів: **9964** Кількість символів: **73671** Розмір файлу: **2.02 MB** ID файлу: **1016004226**

Виявлено модифікації тексту (можуть впливати на відсоток схожості)

15.3%
Схожість

Найбільша схожість: **9.01%** з Інтернет-джерелом (https://ela.kpi.ua/bitstream/123456789/23834/4/Polishchuk_magistr.pd

15.3% Джерела з Інтернету | **452** ..... Сторінка 54

Не знайдено джерел з Бібліотеки

0% Цитат

Вилучення цитат вимкнене

Вилучення списку бібліографічних посилань вимкнене

0%
Вилучень

Немає вилучених джерел

Модифікації

Виявлено модифікації тексту. Детальна інформація доступна в онлайн-звіті.

Замінені символи | **4**

Підозріле форматування | **8 сторінок**

ДОЗВІЛ
НА РОЗМІЩЕННЯ
ВИПУСКНОЇ КВАЛІФІКАЦІЙНОЇ РОБОТИ
В ЕЛЕКТРОННОМУ РЕПОЗИТАРІЇ ВСП «ОТФК ОНТУ»

Ми, що нижче підписалися,

Непомилуєва Ольга Петрівна,
здобувачка освіти гр. 2БКС-28, та

Кривченко Юрій Вікторович,
керівник випускної кваліфікаційної роботи,

не заперечуємо щодо розміщення електронного варіанту пояснювальної записки до випускної кваліфікаційної роботи бакалавра на тему:

«Дослідження методів стеганофонії для аудіо-файлів різних музичних жанрів» (автор роботи – Непомилуєва О.П., керівник роботи – Кривченко Ю.В.)

виконаного у ВСП «Одеський технічний фаховий коледж Одеського національного технологічного університету» в 2024 році, у повному обсязі в електронному репозитарії ВСП «ОТФК ОНТУ» для вільного доступу через мережу Інтернет.

Несемо відповідальність за ідентичність електронного та друкованого варіантів випускної кваліфікаційної роботи і даємо згоду на обробку персональних даних.

Виконавець _____ / Непомилуєва О.П. /

Керівник _____ / Кривченко Ю.В. /

«13» червня 2024 р.