

Національний технічний університет України "Київський політехнічний
інститут імені Ігоря Сікорського"

Міністерство освіти і науки України

Національний технічний університет України "Київський політехнічний
інститут імені Ігоря Сікорського"

Міністерство освіти і науки України

Кваліфікаційна наукова праця
на правах рукопису

Світловський Євгеній Володимирович

УДК 654.026

ДИСЕРТАЦІЯ
ОБРОБЛЕННЯ, ЗАПИС ТА ПЕРЕДАВАННЯ ЗАСОБАМИ ІоТ
МОВНОГО АУДІОСИГНАЛУ З ДЕФЕКТАМИ

17 – Електроніка та телекомунікації

171 – Електроніка

Подається на здобуття наукового ступеня доктора філософії.

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

_____/Світловський Є.В.

Науковий керівник Трапезон Кирило Олександрович, кандидат технічних наук,
доцент

Київ – 2025

АНОТАЦІЯ

Світловський Є.В. Оброблення, запис та передавання засобами IoT мовного аудіосигналу з дефектами. – Кваліфікаційна робота на правах рукопису.

Дисертація на здобуття наукового ступеня доктора філософії за спеціальністю 171 «Електроніка». – Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського», МОН України, Київ, 2025.

Дисертаційна робота присвячена дослідженню підходів обробки та передачі засобами IoT мовного сигналу з дефектами з можливістю додавання додаткової інформації на основі методу найменшого біта.

Зміст дисертаційного дослідження викладено в шістьох розділах, де представлено та обґрунтовано основні результати роботи.

Актуальність дисертаційної роботи обґрунтовано у вступі, де сформульовано мету та задачі дослідження, описано методи дослідження, надано інформацію про наукову новизну та практичне значення одержаних результатів.

Системи IoT, що працюють з мовними сигналами, так або інакше стикаються з проблемами зниження шумового забруднення, компенсації дефектів мовлення та необхідності адаптивної обробки самих сигналів з можливістю за потреби додаткової передачі інформації. Під час процесів оброблення необхідно зберегти якість звуку без додавання нових шумів та артефактів. Додатково, має бути можливість і ефективно знижувати вже присутній рівень шуму в оригінальному сигналі, враховуючи збереження розбірливості мовлення записаного в аудіосигналі. Сучасні методи обробки орієнтовані переважно на іноземні мови і не мають на жаль якісних адаптацій для української мови, що у підсумку при розробленні пристроїв IoT може бути причиною некоректної обробки, неправильної інтерпретації команд або контексту повідомлення.

Для вирішення цих завдань необхідно розробити нові алгоритми, які не лише покращують співвідношення сигнал/шум, а й підвищують розбірливість мовлення та мінімізують втрати інформації під час обробки чи передачі, враховуючи при

цьому специфіку української мови. Крім того, обмежені обчислювальні ресурси та пропускну здатність пристроїв вимагають створення рішень, здатних ефективно працювати в умовах низької якості обладнання та недостатньої шумоізоляції.

Актуальність даного дослідження полягає у необхідності розробки нових рішень для оброблення та передачі дефектних аудіо фрагментів українською мовою з можливістю дублювання та передачі мовного сигналу стеганографічним методом без втрати якості для подальшого декодування та зчитування інформації. Отримані результати можуть знайти практичне застосування в різних сферах, зокрема в системах "розумного будинку", при автоматизованому записі та аналізі онлайн-лекцій, забезпечуючи при цьому новий рівень ефективності та інтерактивності.

У першому розділі визначено основні недоліки існуючих рішень по обробці мовних аудіо сигналів в умовах зашумлення засобами IoT, досліджено основні чинники, які слід враховувати при проведенні якісного запису мовної інформації. Наведено вимоги до вибору приміщень щодо проведення запису мовних аудіосигналів. Зазначено ключові моменти, які сприяють покращенню розбірливості мовлення та зниженню впливу фізичних і електронних шумів.

У другому розділі наведені дані щодо підготовчих кроків до проведення частотного аналізу мовного сигналу. Зокрема, зазначено про такі базові процедури підготовки: розбиття сигналу на сегменти, ідентифікація максимумів та аналіз формант, аналіз фундаментальної частоти.

В третьому розділі дослідження наведено основні принципи кодування текстової інформації за форматами UTF-8 та ASCII та визначені основні етапи розпізнавання мовних сигналів.

У четвертому розділі проведено порівняння характеристик мікрофонів та можливих умов їх застосування. Визначено оптимальну направленість мікрофону для дослідження та розробки алгоритму обробки звукового фрагменту з дефектами. Розглянуто пристрої та методи передачі інформації для реалізації розробленого алгоритму в середовищі Інтернету речей в умовах обмежених обчислювальних ресурсів.

У п'ятому розділі роботи проведено практичний експеримент з підвищення якості та зниження рівня шумового забруднення записаного мовного сигналу українською мовою з наявними технічними дефектами. Зокрема, створено на основі мови програмування Python програмний алгоритм з елементами циклічності, де визначено окремі послідовні етапи обробки сигналу з урахуванням фундаментальної частоти, динамічних та частотних характеристик, рівня шумового забруднення. Досліджено основні підходи до зниження рівня шуму в сигналі, та контролю динамічної і частотної складової сигналу. Визначено міжнародні стандарти нормалізації для приведення звукового сигналу до необхідного рівня гучності. На основі отриманих експериментальних результатів визначено підходи оброблення аудіосигналів, які адаптовано для роботи з українською фонетичною групою.

У шостому розділі наведено алгоритм визначення та кодування тексту з метою додавання супутньої прихованої інформації в аудіофайл. Так, на основі відкритої бібліотеки розпізнавання, вилучено з записаного сигналу текстові дані, і після їх корегування та представлення у необхідній формі, додано за допомогою стеганографічного методу LSB до вмісту аудіосигналу. Показано, що модифікований аудіосигнал практично не змінив свої характеристики у порівнянні з початковим сигналом.

Представлені в дисертації нові практичні результати можуть бути рекомендовані до застосування в умовах дистанційного навчання для запису інформації, адаптивної обробки та передачі сигналів методами Інтернету речей із додаванням супутньої інформації. Наведені розробки можуть бути використані при розробленні складових в системах “розумного будинку” з підтримкою української локалізації. Технології обробки аудіо можуть бути адаптовані для допомоги людям з порушенням слуху методом декодування тексту в зручний формат.

В дисертаційній роботі отримано наступні **наукові результати**:

1. Вперше досліджено та запропоновано алгоритм обробки аудіофайлу українською мовою в умовах зашумлення, адаптований до вимог середовища IoT, який складається з окремих етапів та має риси циклічності.

2. Уточнено алгоритм обробки мовного сигналу, який записано українською мовою, на основі аналізу частотної характеристики з урахуванням особливостей визначення фундаментальної частоти та адаптивних обробок.

3. Вперше розроблено алгоритм подвійної обробки аудіо сигналу з вмістом вимовлених слів українською мовою, який дозволяє реалізувати один з способів приховування потрібної інформації в структурі аудіофайлу зі збереженням якості та без значної зміни енергетичного вмісту останнього.

Практичне значення отриманих результатів полягає у наступному.

1. Визначені підходи до вибору мікрофонного обладнання для запису аудіосигналів, що можуть бути використані при створенні звукових IoT-систем для забезпечення високої якості записаного мовного контенту.

2. Запропоновані ефективніші рішення щодо створення програм обробки аудіосигналів, які дозволяють ефективно очищувати аудіосигнали від шумів та підвищувати розбірливість мовлення, враховуючи специфіку середовища та АЧХ спікера, що сприяє підвищенню якості відтворення записаного контенту в IoT-системах.

3. Використання методу LSB для приховування та передачі супутньої текстової інформації в аудіосигналі забезпечує можливість передачі додаткової інформації без збільшення обсягу даних та помітного впливу на якість звуку.

Ключові слова: акустичне поле, графіки спрямованості акустичного поля, звук, модель, контент, моделювання, процес, Інтернет речей, IoT, стеганографія, комп'ютерна система, рівень сигналу, контент, спектр мови, якість мовлення, тестовий сигнал, розбірливість мовлення.

ANNOTATION

Svitlovskiy Y.V. Assessment of the Processing, Recording, and Transmitting Defective Speech Audio Signals Using IoT Technologies. – Qualifying thesis as a manuscript.

Dissertation for the Doctor of Philosophy degree in Electronics. – National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute", Ministry of Education and Science of Ukraine, Kyiv, 2025.

The dissertation is devoted to the study of approaches to processing and transmitting a speech signal with defects by IoT means with the possibility of adding additional information based on the least bit method.

The content of the dissertation research is presented in six chapters, where the main results of the work are presented and substantiated.

The relevance of the thesis is substantiated in the introduction, which formulates the purpose and objectives of the study, describes the research methods, and provides information on the scientific novelty and practical significance of the results.

IoT systems that work with speech signals face the challenges of reducing noise pollution, compensating for speech defects, and the need for adaptive processing of the signals themselves with the ability to transmit additional information if necessary. During processing, it is necessary to preserve the sound quality without adding new noise and artifacts. Additionally, it should be possible and effective to reduce the noise level already present in the original signal, taking into account the preservation of speech intelligibility recorded in the audio signal. Modern processing methods are focused mainly on foreign languages and, unfortunately, do not have high-quality adaptations for the Ukrainian language, which, as a result, when developing IoT devices, can cause incorrect processing, misinterpretation of commands or message context.

To solve these problems, it is necessary to develop new algorithms that not only improve the signal-to-noise ratio, but also increase speech intelligibility and minimize information loss during processing or transmission, while taking into account the specifics of the Ukrainian language. In addition, limited computing resources and bandwidth of

devices require the creation of solutions that can work effectively in conditions of low quality equipment and insufficient noise insulation.

The relevance of this study lies in the need to develop new solutions for processing and transmitting defective audio fragments in Ukrainian with the ability to duplicate and transmit the speech signal using the steganographic method without loss of quality for further decoding and reading information. The results obtained can find practical application in various fields, including smart home systems, automated recording and analysis of online lectures, while providing a new level of efficiency and interactivity.

The first section identifies the main disadvantages of existing solutions for processing speech audio signals in conditions of noise by IoT, and investigates the main factors that should be taken into account when conducting high-quality recording of speech information. The requirements for the selection of premises for recording speech audio signals are presented. The key points that contribute to improving speech intelligibility and reducing the impact of physical and electronic noise are noted.

The second section presents data on the preparatory steps for conducting frequency analysis of a speech signal. In particular, the following basic preparation procedures are described: splitting the signal into segments, identifying maxima and analyzing formants, and analyzing the fundamental frequency.

The third section of the study presents the basic principles of encoding text information in UTF-8 and ASCII formats and identifies the main stages of speech signal recognition.

Section 4 compares the characteristics of microphones and possible conditions of their use. The optimal microphone directionality for researching and developing an algorithm for processing a sound fragment with defects is determined. The devices and methods of information transmission for the implementation of the developed algorithm in the Internet of Things environment under conditions of limited computing resources are considered.

In the fifth section of the work, a practical experiment was conducted to improve the quality and reduce the level of noise pollution of the recorded speech signal in Ukrainian with existing technical defects. In particular, a software algorithm with elements of

cyclicity was created on the basis of the Python programming language, which defines separate sequential stages of signal processing, taking into account the fundamental frequency, dynamic and frequency characteristics, and the level of noise pollution. The main approaches to reducing the noise level in the signal and controlling the dynamic and frequency components of the signal are investigated. International normalization standards for bringing the sound signal to the required volume level are determined. Based on the experimental results obtained, the approaches to audio signal processing are determined, which are adapted to work with the Ukrainian phonetic group.

Section 6 presents an algorithm for detecting and encoding text in order to add related hidden information to an audio file. Thus, based on an open recognition library, text data is extracted from the recorded signal, and after its correction and presentation in the required form, it is added to the audio signal content using the LSB steganographic method. It is shown that the modified audio signal has practically not changed its characteristics compared to the original signal.

The new practical results presented in this dissertation can be recommended for use in distance learning for recording information, adaptive processing and transmission of signals using the Internet of Things methods with the addition of related information. These developments can be used in the design of components in smart home systems with support for Ukrainian localization. Audio processing technologies can be adapted to help people with hearing impairments by decoding text into a convenient format.

In the dissertation research, the following scientific results have been obtained:

1. For the first time, an algorithm for processing an audio file in Ukrainian under noise conditions, adapted to the requirements of the IoT environment, which consists of separate stages and has the features of cyclicity, is investigated and proposed.
2. The algorithm for processing a speech signal recorded in Ukrainian is refined based on the analysis of the frequency response, taking into account the peculiarities of determining the fundamental frequency and adaptive processing.
3. For the first time, an algorithm for double processing of an audio signal containing spoken words in Ukrainian was developed, which allows to implement one of

the ways to hide the necessary information in the structure of an audio file while maintaining quality and without significantly changing the energy content of the latter.

The practical significance of the obtained results lies in the following:

1. The approaches to the selection of microphone equipment for recording audio signals that can be used in the creation of audio IoT systems to ensure high quality of recorded speech content are determined.

2. More effective solutions for creating audio signal processing programs are proposed that allow to effectively clean audio signals from noise and increase speech intelligibility, taking into account the specifics of the environment and the speaker's frequency response, which contributes to the quality of playback of recorded content in IoT systems.

3. Using the LSB method to hide and transmit accompanying textual information in the audio signal allows the transmission of additional information without increasing the amount of data and significantly affecting the sound quality.

Keywords: acoustic field, acoustic field directivity graphs, sound, model, content, modeling, process, Internet of Things, IoT, steganography, computer system, signal strength, content, speech spectrum, speech quality, test signal, speech intelligibility.

Статті у наукових фахових виданнях України

1. Світловський Є.В., Трапезон К.О. Аналіз мовних акустичних сигналів в системах зв'язку з частковим зашумленням. // Вчені записки Таврійського національного університету імені В.І. Вернадського. 2022. Vol. 33. № 5 (72). Р. 380-385. <https://doi.org/10.32782/2663-5941/2022.5/59>
2. Світловський Є.В., Трапезон К.О. Стеганографічні підходи до оброблення аудіо сигналів. // Вісник Кременчуцького національного університету імені Михайла Остроградського. 2023. Vol. 3. Р. 185-192. <https://doi.org/10.32782/1995-0519.2023.3.22>
3. Світловський Є.В. Моделі і алгоритми створення цифрових знаків для аудіо файлів. // Перспективні технології та прилади. Луцький національний технічний університет. 2024. Vol. 1. № 24. Р. 99-106. <https://doi.org/10.36910/10.36910/6775-2313-5352-2024-24-15>

Доповіді на конференціях:

4. Світловський Є.В., Трапезон К.О. Аналіз мовних акустичних сигналів в системах зв'язку з частковим зашумленням. // «Радіотехнічні поля, сигнали, апарати та системи» XI Міжнародна науково-технічна конференція, м. Київ, 2022.
5. Світловський Є.В. Стеганографія в інформаційних системах. // «Science and technology: challenges, prospects and innovations» Міжнародна науково-практична конференція, Осака, Японія, 2024.

Статті що додатково відображають результати дисертації

6. Світловський Є.В., Трапезон К.О. Алгоритмічний підхід реалізації програмного скремблінгу аудіосигналів. // Вісник Кременчуцького національного університету імені Михайла Остроградського. 2024. Vol. 1 Р. 273-280. <https://doi.org/10.32782/1995-0519.2024.1.36>.

ЗМІСТ

	ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, ТЕРМІНІВ ТА СКОРОЧЕНЬ	15
	ВСТУП.....	16
1	ОГЛЯД ОСОБЛИВОСТЕЙ ЗАПИСУ ТА ПЕРЕДАЧІ АУДІОІНФОРМАЦІЇ	21
1.1	Проблеми та виклики в середовищі IoT	21
1.2	Особливості запису аудіосигналів.....	23
1.3	Акустика приміщення.....	23
1.4	Вимоги до приміщень звукозапису	24
1.5	Розбірливість мови.....	25
1.6	Звукоізоляція	25
1.7	Звукопоглинання	25
1.8	Частотний діапазон та чутливість мікрофонів	26
1.9	Вплив електронних та фізичних шумів на звук	27
1.10	Якість звукової карти та записуючого обладнання.....	28
	Висновки до розділу 1.....	28
2	ОСОБЛИВОСТІ ОБРОБКИ АУДІОФАЙЛІВ З ЗАШУМЛЕННЯМ	30
2.1	Розбиття сигналу на сегменти.....	30
2.2	Ідентифікація піків, аналіз формант та визначення шумів у частотному аналізі мовного сигналу	32
2.2.1	Ідентифікація максимумів	32
2.2.2	Аналіз формант	33
2.3	Класичні підходи до оцінки шуму.....	34
2.4	Аналіз фундаментальної частоти.....	35
2.4.1	Застосування фундаментальної частоти.....	37
	Висновки до розділу 2.....	38
3	ДЕКОДУВАННЯ ТЕКСТУ ТА ПЕРЕДАЧА ДОДАТКОВОЇ ІНФОРМАЦІЇ	
	МЕТОДОМ МЕНШ ЗНАЧУЩОГО БІТА.....	40
3.1	Етапи автоматичного розпізнавання мови та приклади систем	40
3.2	Кодування тексту за форматом ASCII	42

	13
3.3	Кодування тексту за форматом UTF-8..... 43
3.4	Метод LSB (Least Significant Bit) для стеганографії..... 45
	Висновки до розділу 3..... 46
4	ПРАКТИЧНА ЧАСТИНА ДОСЛІДЖЕННЯ..... 47
4.1	Вибір аудіоінтерфейсу для експерименту 47
4.2	Вибір мікрофону для експерименту 49
4.2.1	Мікрофон з кардіоїдною направленістю..... 50
4.2.2	Мікрофон типу "фігура-вісімка" 53
4.2.3	Всенаправлений мікрофон..... 55
4.3	Порівняння мікрофонів..... 57
4.4	Вибір обладнання..... 60
4.5	Передача мовних сигналів засобами IoT 61
	Висновки до розділу 4..... 64
5	ОБРОБКА МОВНОГО СИГНАЛУ ЗАПИСАНОГО УКРАЇНСЬКОЮ МОВОЮ З ДЕФЕКТАМИ 65
5.1	Попередня обробка мовного сигналу..... 66
5.1.1	Видалення постійної складової 68
5.1.2	Нормалізація сигналу 69
5.1.3	Визначення фундаментальної частоти 71
5.1.4	Застосування фільтру високих та низьких частот 72
5.2	Зниження шумового забруднення та поліпшення розбірливості мови..... 76
5.2.1	Зниження шумового забруднення..... 76
5.2.2	Шумовий поріг..... 78
5.2.3	Багатосмугова компресія 82
5.2.4	Компресія високих частот..... 85
5.2.5	Спектральне віднімання..... 88
5.2.6	Нормалізація сигналу 91
5.3	Порядок обробки та фінальні результати 93
5.4	Аналогова реалізація обробки сигналу 99
5.4.1	Переваги цифрової обробки 101

	14
Висновки до розділу 5.....	103
6 РОЗПІЗНАВАННЯ МОВИ ТА КОДУВАННЯ МОВНОГО СИГНАЛУ	105
6.1 Розпізнавання мовлення з аудіофайлу	105
6.2 Кодування розпізнаного тексту в бінарний формат	106
6.3 Вбудовування бінарного повідомлення в аудіосигнал методом LSB	108
6.4 Інтеграція в середовище IoT	115
Висновки до розділу 6.....	117
ВИСНОВКИ	119
СПИСОК ДЖЕРЕЛ ПОСИЛАНЬ	121
ДОДАТОК А КОД ДОСЛІДЖЕНЬ ТА ОБРОБКИ ЗВУКУ	134

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, ТЕРМІНІВ ТА СКОРОЧЕНЬ

АЦП	- Аналого-цифровий перетворювач;
АЧХ	- Амплітудно-частотна характеристика;
ДПФ	- Дискретне перетворення Фур'є
МП	- Мікшерний пульт;
ПК	- Персональний комп'ютер;
СКВ	- Середньо-квадратичне відхилення;
ЦАП	- Цифро-аналоговий перетворювач;
ASCII	- American Standard Code for Information Interchange;
ASR	- Automatic Speech Recognition;
DCT	- Discrete Cosine Transform;
DSP	- Digital Signal Processor;
DWT	- Discrete Wavelet Transform;
EQ	- Equalizer;
FFT	- Fast Fourier Transform;
IoT	- Internet of Things;
LF	- Line Feed;
LoRaWAN	- Long Range Wide Area Network;
LSB	- Least Significant Bit;
MFCC	- Mel-frequency cepstral coefficients;
RMS	- Root Mean Square;
SBC	- Subband Coding;
SNR	- Signal-to-Noise Ratio;
UTF-8	- Unicode Transformation Format – 8-bit;
VAD	- Voice Activity Detection.

ВСТУП

Актуальність роботи. У сучасному світі технологій Інтернету речей (IoT) наявний стрімкий розвиток електронних засобів збору, обробки та передачі аудіо інформації, які використовуються для створення різноманітних приладів, які, наприклад, підвищують комфортність життя людей. Зростає потреба в системах, здатних ефективно опрацювати і вилучати інформацію з мовного сигналу з дефектами та шумовим забрудненням, враховуючи можливість передачі при цьому додаткової інформації. Дослідження у цій галузі значною мірою зосереджені на обробці та передачі сигналів англійською та інших поширених мов, тоді як українська мова залишається недостатньо вивченою. Це обумовлено, як відсутністю належної бази досліджень, так і мовними особливостями, такими як складна фонетика, інтонаційна структура та сибілянти, які можуть створювати додаткові труднощі для оброблення та передачі.

Крім того, передові системи IoT, що працюють з мовними сигналами, так або інакше стикаються з проблемами зниження шумового забруднення, компенсації дефектів мовлення та необхідності адаптивної обробки самих сигналів з можливістю за потреби додаткової передачі інформації. Важливо, щоб при цьому враховувалось збереження якості звуку, не створювалось додаткових шумів та водночас, була можливість передавати додаткову текстову інформацію. Ця проблема набула розповсюдження в таких сферах як: управління розумними будинками, запис онлайн-лекцій, відеоконференції та інші інтерактивні застосунки.

Для цього необхідно розробити алгоритми, які дозволяють не лише покращувати відношення корисного сигналу до рівня шуму, а й підвищувати розбірливість мовлення та мінімізувати втрати інформації при обробці або передачі, враховуючи особливості української мови. Не менш важливим фактором виступають обмежені обчислювальні ресурси та пропускна спроможність приладів, що впливає на ефективність розроблених рішень, їх здатність працювати в умовах низької якості обладнання, та недостатньої шумоізоляції спікера.

Актуальність даного дослідження полягає у необхідності розробки нових рішень для оброблення та передачі дефектних аудіо фрагментів українською мовою з можливістю дублювання та передачі мовного сигналу стеганографічним методом без втрати якості для подальшого декодування та зчитування інформації. Отримані результати можуть знайти практичне застосування в різних сферах, зокрема в системах "розумного будинку", при автоматизованому записі та аналізі онлайн-лекцій, забезпечуючи при цьому новий рівень ефективності та інтерактивності.

Питаннями застосування різних алгоритмів оброблення акустичної інформації займаються такі вітчизняні та іноземні вчені як, Продеус А.М., Можєєв О.О., Розорінов Г.М., Ніколов М., Белінський В., Ольсон Г., Гарсія Л., та інші.

Мета та задачі дослідження. *Метою даної роботи є розроблення алгоритму зменшення шумового забруднення мовного сигналу з дефектами, який записано українською мовою та створення об'єднаного практичного підходу з процедури розпізнавання мовлення з аудіо сигналу і додавання до нього додаткової прихованої інформації методом менш значущого біта.*

Об'єктом дослідження є мовні аудіосигнали, які записано українською мовою з дефектами в IoT-середовищі.

Предметом дослідження є методи та засоби підвищення якості обробки українськомовних мовних сигналів з дефектами.

Для досягнення поставленої мети необхідно вирішити наступні завдання:

1. Визначити основні недоліки існуючих засобів обробки мовного аудіо файлу з дефектами. Розглянути особливості, щодо забезпечення якісного запису аудіо сигналів на основі вибору пристроїв.
2. Розробити, задля зменшення шумового забруднення мовного сигналу українською мовою з наявними дефектами, програмний алгоритм в середовищі мови об'єктно-орієнтованого програмування Python, який має визначатись окремими послідовними етапами з ознаками циклічності.
3. Визначити основні принципи кодування текстової інформації за форматами UTF-8 та ASCII та перспективи використання методу LSB для додавання додаткової прихованої інформації в аудіосигнал. Перевірити

запропонований алгоритм додавання прихованої інформації в аудіо сигнал на предмет його практичної доцільності.

Методи дослідження. Для виконання поставленої мети і вирішення поставлених завдань використано методику кодування UTF-8 для бінарного перетворення мови, використано стеганографічний метод для передачі додаткової інформації в оброблений аудіо файл. Для покращення якості та очищення файлу від шумів застосовано ряд програмних інструментів обробки сигналів, принцип функціонування котрих визначаються в літературі на основі математичних співвідношень.

Наукова новизна отриманих результатів.

1. Вперше досліджено та запропоновано алгоритм обробки аудіофайлу українською мовою в умовах зашумлення, який складається з окремих етапів та адаптований до роботи в середовищі IoT.

2. Уточнено алгоритм обробки мовного сигналу, який записано українською мовою, на основі аналізу частотної характеристики з урахуванням особливостей визначення фундаментальної частоти.

3. Вперше розроблено алгоритм подвійної обробки аудіо сигналу з вмістом вимовлених слів українською мовою, який дозволяє реалізувати один з способів приховування потрібної інформації в структурі аудіофайлу зі збереженням якості та без практично незмінності енергетичного вмісту останнього.

Особистий внесок здобувача. Усі результати, наведені у дисертаційній роботі і винесені на захист, отримано за активної участі здобувача та опубліковано у спеціалізованих фахових виданнях.

У роботі «Аналіз мовних акустичних сигналів в системах зв'язку з частковим зашумленням». Світловський Є.В., Трапезон К.О. «Вчені записки Таврійського національного університету імені В.І.Вернадського.», 2022, опублікованій в співавторстві, здобувач особисто дослідив вплив дії шуму на мовний сигнал.

У роботі «Стеганографічні підходи до оброблення аудіо сигналів». Світловський Є.В., Трапезон К.О. «Вісник Кременчуцького національного

університету імені Михайла Остроградського.», 2023, опублікований в співавторстві, здобувач особисто дослідив вплив методу LSB на якість аудіофайлу.

У роботі «Моделі і алгоритми створення цифрових знаків для аудіо файлів». Світловський Є.В. «Перспективні технології та прилади. Луцький національний технічний університет», 2024, здобувач особисто дослідив можливість приховування великих обсягів інформації.

У роботі «Алгоритмічний підхід реалізації програмного скремблінгу аудіосигналів». Світловський Є.В., Трапезон К.О. «Вісник Кременчуцького національного університету імені Михайла Остроградського», 2024, опублікований в співавторстві, здобувач особисто дослідив підхід і реалізацію програмного алгоритму для кодування текстових даних в аудіосигнал на частотах, які є нечутними для людини.

Практичне значення отриманих результатів. Практичне значення отриманих результатів полягає у розробці алгоритму обробки та очищення аудіосигналів від шумів, кодування тексту та передачі супутньої інформації методом менш значущого біта. Розроблені методи та алгоритми дозволяють ефективно видаляти шуми з мовних аудіо сигналів записаних українською мовою, підвищуючи їх якість та розбірливість, що є критичним для надійного спілкування та передавання інформації в IoT-системах. Одночасно, використання стеганографічного методу найменш значущого біта (LSB) для приховування та передачі додаткової текстової інформації в аудіосигналі забезпечує можливість передачі супутньої інформації без збільшення обсягу передавальних даних та без помітного впливу на якість звуку. Цей комплексний підхід дозволяє створювати багатофункціональні IoT-рішення, які поєднують високоякісну обробку мовлення з безпечною передачею додаткової інформації. Розроблені підходи до очищення аудіо від шумів та передачі супутньої інформації інтегруються в єдине рішення, що відповідає сучасним вимогам до якості та безпеки передачі даних в IoT-середовищі.

Апробація результатів дисертації. Основні положення та результати дисертаційного дослідження доповідались на 2 міжнародних науково-практичних конференціях:

1. Аналіз мовних акустичних сигналів в системах зв'язку з частковим зашумленням. Світловський Є.В., Трапезон К.О. «Радіотехнічні поля, сигнали, апарати та системи» XI Міжнародна науково-технічна конференція м. Київ, 2022

2. Стеганографія в інформаційних системах. Світловський Є.В. «Science and technology: challenges, prospects and innovations» Міжнародна науково-практична конференція Японія, Осака, 2024

Публікації. За результатами досліджень опубліковано 4 наукових публікацій (з них 3 статті у наукових фахових виданнях України за спеціальністю 171 Електроніка), 2 доповіді у збірниках матеріалів конференцій.

Структура та обсяг дисертаційної роботи. Робота складається зі вступу, шести розділів, списку використаних джерел із 117 найменувань та 1 додатку. Робота містить 28 рисунків та 2 таблиці. Загальний обсяг дисертаційної роботи складає 145 сторінок.

1 ОГЛЯД ОСОБЛИВОСТЕЙ ЗАПИСУ ТА ПЕРЕДАЧІ АУДІОІНФОРМАЦІЇ

1.1 Проблеми та виклики в середовищі IoT

Сучасні технології Інтернету речей (IoT) активно використовують обробку мовлення для голосового управління, моніторингу та безпечного передавання інформації. Однак, робота з мовними сигналами в умовах IoT стикається з рядом проблем, які обмежують ефективність розроблених існуючих рішень. Більшість сучасних алгоритмів обробки мовлення орієнтовані переважно на англійську мову. Вони не враховують особливості української фонетики, що призводить до втрати розбірливості, і як наслідок, система може неправильно інтерпретувати команди, або звучати спотворено. Проблема ускладнюється ще тим, що стандартні мовні моделі для автоматичного розпізнавання мови (ASR) значно менш ефективні для роботи з українською, так як адаптовані під інші фонетики [1, 2].

Зазвичай, в середовищі IoT для оброблення звуку застосовуються статичні параметри, які не змінюються залежно від рівня шуму, акустичних умов чи особливостей голосу. Як наслідок, вони або недостатньо ефективно видаляють шум, або занадто агресивно фільтрують сигнал, що призводить до втрати мовної інформації. У реальних умовах, де рівень шуму може змінюватися динамічно (наприклад, у міському середовищі, вдома чи на виробництві), такі методи не забезпечують якісного результату [3].

Широкого розповсюдження останнім часом набувають нейромережеві методи, такі як RNNoise, DeepFilterNet. Проте, високе споживання енергії, необхідність значних обчислювальних ресурсів та пам'яті, може бути причиною їх обмеженого використання для вбудованих систем у пристроях з низькою продуктивністю, де критичним є баланс між ефективністю та витратою електричної енергії. Так, для зміни параметрів нейронної мережі потребують довготривалого навчання, а напрям обробки може бути зосереджений лише на автоматичне видалення шумів, не

враховуючи при цьому розбірливість мовлення, спектральні, динамічні характеристики сигналу [4, 5].

Ще одним напрямком з адаптації системи IoT до зашумлення є мультиканальні системи. Їх принцип полягає у використанні кількох мікрофонів з різною спрямованістю (може бути як адаптивна, так і статична реалізація) для подальшого порівняння та аналізу записаного звуку з різних напрямків. Основними проблемами такого підходу є обмеження в кількості підключених мікрофонів, підвищення вартості такої системи, обчислювальна складність, чутливість до розташування мікрофонів, і як наслідок, погана адаптація до можливої зміни умов середовища [6].

Обмін даними в середовищі Інтернету речей передбачає використання обмежених ресурсів, що зумовлено поширеністю пристроїв з автономним живленням, мікроконтролерів із низькими обчислювальними можливостями, невеликим обсягом вбудованої пам'яті та специфічними конструктивними обмеженнями. В таких умовах пріоритетним завданням є забезпечення надійного зв'язку на значній відстані за мінімальних енерговитрат і з оптимальним використанням пропускної здатності мережі. Розглянемо на прикладі мікроконтролера Heltec WiFi LoRa 32 V, який поєднує в собі модуль LoRaWAN (Long Range Wide Area Network) та ESP32 [7], мікрофон SPH0645LM4H [8] для запису аудіосигналу, який може забезпечити реалізацію первинної обробки звукових даних безпосередньо на пристрої з можливістю підключення через LoRaWAN [9]. Відомо, що технологія LoRaWAN здатна забезпечувати передавання даних на великі відстані за низької швидкості, що дає змогу зменшити енергоспоживання передавального пристрою та одночасно підтримувати велику кількість кінцевих пристроїв. Такий підхід відкриває можливість створення енергоощадних, проте функціональних IoT-систем, що здатні працювати автономно і виконувати завдання збору та обробки даних.

Через обмежені ресурси мікроконтролера та специфіку LoRaWAN-зв'язку, до обробки мовного сигналу постає необхідність розробки оптимізованих алгоритмів, що здатні забезпечити високу якість і точність в обробці мовного сигналу,

виділенні ключових характеристик в умовах обмежених обчислювальних ресурсів, об'єму та швидкості передачі даних [10].

1.2 Особливості запису аудіосигналів

Запис аудіосигналів є важливим кроком у багатьох застосуваннях, від професійного звукозапису до систем розпізнавання мови. Якість запису визначається багатьма факторами, включаючи вибір мікрофону, акустичні властивості приміщення, налаштування запису, а також електронні та фізичні шуми [11].

Конструкція сучасних мікрофонів є доволі складною та різноманітною. Виходячи з задачі та умов можна визначити тип мікрофону, наприклад для звукозапису, найчастіше використовуються конденсаторні мікрофони, конструкція яких має високу чутливість та широкий частотний діапазон [12-14]. Для публічних виступів частіше застосовують динамічні або лавальнірні мікрофони, які менш чутливі до навколишніх шумів та середовища запису [15].

Цифрові мікрофони, на відміну від аналогових, використовують аналого-цифровий перетворювач і цифровий процесор для попередньої обробки звуку, замінюючи традиційний передпідсилювач [16]. Ця технологія дозволяє отримати більш точне відтворення звуку та полегшує подальшу його обробку.

Для забезпечення якісного звукозапису велике значення має правильне розташування мікрофонів, яке залежить від типу та розташування джерел звуку [17].

1.3 Акустика приміщення

Акустичні властивості приміщення, де здійснюється запис, мають суттєвий вплив на якість аудіосигналу. Важливо мінімізувати реверберацію та зовнішні шуми, що може бути досягнуто за допомогою акустичних обробок. Розмір приміщення відіграє важливу роль. Велике приміщення з високими стелями може

створювати значну реверберацію, яка ускладнює запис [18]. Натомість, маленькі приміщення можуть сприяти накопиченню стоячих хвиль, що також негативно впливає на якість звуку. Оптимальний вибір – середні за розміром приміщення з належною акустичною обробкою [19]. Форма приміщення також має значення. Прямокутні приміщення часто створюють проблеми з реверберацією через паралельні стіни, що відбивають звукові хвилі. Приміщення неправильної форми можуть сприяти більш рівномірному розподілу звуку. Матеріали поверхонь приміщення теж впливають на акустику. Відбивальні поверхні, такі як скло та бетон, можуть збільшувати реверберацію, тоді як звукопоглинаючі матеріали, такі як килими та тканини, можуть допомогти зменшити її [20].

Обробка приміщення включає кілька підходів. Використання звукопоглинаючих панелей на стінах та стелі допомагає зменшити реверберацію та поліпшити загальну акустику приміщення. Панелі поглинають звукові хвилі, зменшуючи їх відбиття. Бас-пастки призначені для поглинання низькочастотних звуків, які можуть викликати резонанс у приміщенні. Встановлення бас-пасток у кутах приміщення допомагає зменшити низькочастотні резонанси. Дифузори розсіюють звукові хвилі, що допомагає уникнути фокусування звуку в певних точках приміщення. Вони забезпечують більш рівномірне розподілення звуку по приміщенню [21].

1.4 Вимоги до приміщень звукозапису

Акустичні властивості студії звукозапису значно відрізняються від властивостей аудиторій. Студія повинна бути акустично «мертвою» з дуже коротким часом реверберації, що вимагає від приміщення високого рівня поглинання звуку та достатньої звукоізоляції [22].

1.5 Розбірливість мови

Розбірливість мови є важливим критерієм якості студії. Вона визначається відсотком артикуляції (4%), отриманим в результаті артикуляційних випробувань у приміщенні. Ідеальною розбірливістю вважається 96% артикуляції, а незадовільною – 65%. Оптимальний час реверберації для мовних програм визначається за формулою [23].

1.6 Звукоізоляція

Щоб запобігти проникненню низькочастотних звуків (наприклад, шуму транспорту та літаків), кімната запису зазвичай ізолюється від основної конструкції подвійними стінками. Ця стратегія, відома як «кімната у кімнаті», мінімізує структурний зв'язок між студією та фундаментом будівлі, зменшуючи передачу низькочастотних звуків через тверді конструкції. Важливо забезпечити ретельну герметизацію приміщення та правильне проектування системи опалення та кондиціонування. Додатково використовуються басові пастки для зменшення низькочастотного фону [19].

1.7 Звукпоглинання

Звукпоглинальні матеріали, такі як ізоляція з пінопласту в стінах, допомагають герметизувати приміщення та поглинати звуки середньої та високої частоти. Низькі частоти, як правило, ефективніше передаються через тверді конструкції, тому для ізоляції студії від низьких частот часто використовується конструкція з подвійними стінками. Герметичність конструкції є важливою, оскільки навіть невеликі отвори можуть ускладнити звукоізоляцію [24].

Для досягнення необхідного рівня звукоізоляції важливо враховувати логарифмічну реакцію людського вуха на інтенсивність звуку. Зменшення гучності агресивних звуків у десять разів знижує їх інтенсивність лише вдвічі. Це означає,

що сильні зовнішні звуки потрібно зменшити більш ніж у мільйон разів, щоб зробити їх нечутними. Це значення виражається у необхідності забезпечення значного ослаблення звуку для досягнення комфортних умов у студії.

Окрім того, необхідно враховувати акустичні матеріали, які забезпечують поглинання звуків середньої та високої частоти, а також конструкції з подвійними стінками, які ефективно зменшують передачу басів [25].

Вимоги до приміщень звукозапису охоплюють комплекс заходів з акустичної обробки та звукоізоляції для забезпечення високоякісного запису звуку. Це включає мінімізацію реверберації, ефективну ізоляцію від зовнішніх шумів та застосування звукопоглинальних матеріалів. Використання концепції «кімната у кімнаті» дозволяє досягти необхідного рівня звукоізоляції, що забезпечує комфортні умови для звукозапису [25].

1.8 Частотний діапазон та чутливість мікрофонів

Номінальний діапазон частот (frequency range) – це частотний діапазон, який мікрофон може записати і для конденсаторних мікрофонів він визначається в межах 20-20000 Гц, тоді як для вимірювальних мікрофонів діапазон може становити 20-50000 Гц [12]. З розвитком цифрової звукотехніки з'явилися мікрофони з діапазоном частот до 50 кГц, призначені для студійного звукозапису, наприклад, модель МКН800 від Sennheiser [26].

Чутливість (sensitivity) визначає здатність мікрофона перетворювати акустичний тиск в електричну напругу. Це відношення сигналу на виході мікрофона (U) до звукового тиску на вході мікрофона (p). Чутливість зазвичай виражається в мВ/Па ($S = U / p$). У міжнародних стандартах, таких як ІЕС 60268-4, чутливість визначається як середньоквадратичне значення напруги на виході мікрофона за опору навантаження 1 кОм на частоті 1 кГц при звуковому тиску 1 Па (94 дБ) при куті прийому 0 градусів [27].

У сучасних каталогах часто вказується рівень чутливості, який визначається як 20 логарифмів відносної чутливості мікрофона до значення 1 В/Па ($L = 20\lg(S / S_0)$), де $S_0 = 1$ В/Па) [23]. Значення рівня чутливості часто буває від'ємним.

Чутливість конденсаторних мікрофонів зазвичай знаходиться в межах 8 мВ/Па до 40 мВ/Па. Наприклад, мікрофон DPA 3530 має чутливість 10 мВ/Па, а рівень чутливості -40 дБВ [28], тоді як мікрофон AKG C3000B має чутливість 25 мВ/Па (-32 дБВ) [29]. Виробники часто вказують допустимі межі розкиду чутливості при виробництві (1-2 дБ).

1.9 Вплив електронних та фізичних шумів на звук

Електронні шуми виникають внаслідок електричних перешкод та неідеальностей компонентів аудіосистеми [24]. Основні джерела електронних шумів включають тепловий шум, який виникає через тепловий рух електронів у провідниках та компонентах. Він є незмінним і збільшується з підвищенням температури. Шум перетворення виникає під час оцифрування сигналу в аудіоінтерфейсі. Важливо вибирати аудіоінтерфейси з високоякісними АЦП (аналого-цифровими перетворювачами), які мають низький рівень шуму. Інтермодуляційний шум виникає через взаємодію різних частотних компонентів сигналу та може призвести до спотворення звуку [30].

Фізичні шуми виникають від зовнішніх джерел та навколишнього середовища. Основні джерела фізичних шумів включають вібрацію, яка може передаватися через конструктивні елементи приміщення та впливати на запис. Використання віброзахисту для мікрофонів та аудіоінтерфейсів допомагає зменшити цей вплив. Фонові шуми, такі як шум від кондиціонерів, вентиляторів та інших побутових приладів, можуть погіршити якість запису. Важливо мінімізувати ці шуми шляхом використання звукопоглинаючих матеріалів та ізоляційних методів. Акустичні відбиття від поверхонь приміщення можуть спричинити реверберацію та інші небажані ефекти. Правильне розташування мікрофонів та використання акустичних панелей допомагає зменшити цей вплив [31].

1.10 Якість звукової карти та записуючого обладнання

Звукові карти забезпечують перетворення аналогового сигналу в цифровий та навпаки. Вибір звукової карти залежить від кількох параметрів. Висока роздільна здатність забезпечує точне оцифрування сигналу з мінімальними втратами якості [32]. Чим вища роздільна здатність, тим більше деталей можна зберегти у записі. Чим нижчий рівень шуму, тим чистіший буде запис, що дозволить забезпечити високу розбірливість мовлення та можливість коректної обробки аудіосигналу. Навіть незначні шумові артефакти можуть вплинути на кінцеву якість продукту [33]. Вибір звукової карти з достатньою кількістю каналів забезпечує гнучкість та можливість запису кількох джерел звуку одночасно. Сучасні звукові карти використовують різні типи підключень, включаючи USB, Thunderbolt, FireWire та інші. Вибір залежить від сумісності з комп'ютером та вимог щодо швидкості передачі даних [34].

Таким чином, запис аудіосигналів є складним процесом, який включає вибір мікрофону, обробку приміщення, врахування електронних та фізичних шумів, а також використання якісного записуючого обладнання. Кожен з цих аспектів має критичне значення для досягнення високоякісного запису, який відповідає вимогам сучасних технологій та стандартів.

Висновки до розділу 1.

У першому розділі проаналізовано наукові праці та дослідження, присвячені дослідженням запису та обробки мовної аудіоінформації. Вивчено існуючі технології та засоби обробки мовного сигналу в середовищі інтернету речей та їх неоліки. Відмічено, що якість запису залежить від вибору типу мікрофона його розташування, акустичних властивостей приміщення та впливу фізичних і електронних шумів. Визначено рекомендації по застосуванню звукопоглинальних матеріалів, басових пасток та концепції "кімната у кімнаті" для зменшення впливу реверберації та шумів. Зазначено, що якість запису значно залежить від параметрів

звукових карт та аудіоінтерфейсів, які повинні забезпечувати високу роздільну здатність, низький рівень шуму та сучасні інтерфейси підключення.

2 ОСОБЛИВОСТІ ОБРОБКИ АУДІОФАЙЛІВ З ЗАШУМЛЕННЯМ

2.1 Розбиття сигналу на сегменти

Розбиття сигналу на сегменти є важливим етапом у процесі частотного аналізу аудіосигналів, що дозволяє розглянути локальні властивості сигналу, забезпечуючи стаціонарність в межах кожного фрейму. Це необхідно для того, щоб частотний аналіз, зокрема перетворення Фур'є, був точним і ефективним [35].

Аудіосигнали зазвичай є нестационарними, тобто їхні статистичні властивості змінюються з часом. Для того щоб застосувати методи частотного аналізу, такі як дискретне перетворення Фур'є (DFT) або швидке перетворення Фур'є (FFT), необхідно зробити припущення про стаціонарність сигналу. Розбиття на фрейми дозволяє локалізувати аналіз у часі, що дозволяє обчислювати частотні компоненти для коротких інтервалів, де сигнал можна вважати стаціонарним. Це забезпечує точний і детальний аналіз частотних компонентів сигналу [36, 37].

Розмір сегменту визначає тривалість часу, протягом якого здійснюється аналіз. Вибір оптимального розміру фрейму є критичним, оскільки він впливає на частотну та часову роздільну здатність аналізу.

Малий розмір сегменту (10-20 мс) забезпечує кращу часову роздільну здатність, що дозволяє детально аналізувати швидкі зміни сигналу. Однак це може призвести до погіршення частотної роздільної здатності. Великий розмір сегменту (20-40 мс) забезпечує кращу частотну роздільну здатність, що дозволяє детально аналізувати частотні компоненти сигналу. Однак це може призвести до зниження часової роздільної здатності та втрати детальних змін у сигналі. Вибір розміру фрейму зазвичай є компромісом між часовою та частотною роздільною здатністю і залежить від конкретної задачі аналізу [38].

Для забезпечення плавного переходу між сегментами та зменшення втрат інформації використовується перекриття (overlapping). Типове перекриття становить від 50% до 75%. Перекриття 50% означає, що кожен новий фрейм починається з середини попереднього фрейму, що забезпечує безперервність

аналізу і зменшує артефакти на межах фреймів. Збільшення перекриття до 75% дозволяє ще більше згладити перехід між фреймами, але потребує більше обчислювальних ресурсів.

Для кожного сегменту застосовується віконна функція для зменшення ефекту витікання спектра, що виникає через обмеженість фрейму в часі. Віконні функції згладжують краї фрейму, що зменшує спотворення частотних компонент.

Однією з найпоширеніших віконних функцій є вікно Ганна, яке має гарні властивості для зменшення витікання. Формула для обчислення вікна Ханна виглядає так [39]:

$$w(n) = 0.5 \left(1 - \cos \left(\frac{2\pi n}{N-1} \right) \right)$$

де N — довжина фрейму, n — індекс зразка.

Іншою популярною віконною функцією є вікно Хеммінга, яке має подібні властивості, але трохи більший рівень основної лоби в спектрі. Формула для обчислення вікна Хеммінга виглядає так [39]:

$$w(n) = 0.54 - 0.46 \cos \left(\frac{2\pi n}{N-1} \right).$$

Алгоритм розбиття сигналу на сегменти включає кілька ключових кроків. Спочатку сигнал розбивається на фрейми заданого розміру з визначеним перекриттям, що забезпечує поділ сигналу на короткі інтервали, кожен з яких може бути окремо проаналізований. Після цього до кожного фрейму застосовується обрана віконна функція для згладжування країв, що зменшує ефект витікання спектра і покращує точність частотного аналізу. Нарешті, для кожного фрейму обчислюється швидке перетворення Фур'є (FFT) для отримання частотного спектра, що дозволяє визначити частотні компоненти кожного фрейму і побудувати загальну картину частотного вмісту сигналу.

Розбиття сигналу на сегменти забезпечує точний і детальний аналіз частотних компонентів аудіосигналу. Вибір оптимального розміру фрейму, перекриття фреймів і застосування віконних функцій є ключовими аспектами, що впливають на якість частотного аналізу. Використання цих підходів дозволяє ефективно

обробляти аудіосигнали для різних додатків, включаючи розпізнавання мовлення, музичне продюсування та звуковий дизайн [39].

2.2 Ідентифікація піків, аналіз формант та визначення шумів у частотному аналізі мовного сигналу

Частотний аналіз мовного сигналу є важливим аспектом досліджень пов'язаних з обробкою аудіосигналів. Він дозволяє розкласти сигнал на його частотні компоненти, визначити основні характеристики голосу, а також виявити і усунути небажані шуми [40]. У цьому контексті ідентифікація піків, аналіз формант і визначення шумів є ключовими етапами, що дозволяють детально вивчити і обробити мовний сигнал [41].

2.2.1 Ідентифікація максимумів

Ідентифікація піків у спектрі сигналу дозволяє визначити частоти, на яких зосереджені основні енергетичні компоненти сигналу. Піки у спектрі відповідають гармонікам та основним тонам, які визначають акустичні властивості голосу.

Процес ідентифікації піків починається з обчислення спектра сигналу за допомогою швидкого перетворення Фур'є (FFT). Дискретне перетворення Фур'є (DFT) розкладає сигнал на суму синусоїдальних компонент, кожна з яких має певну частоту та амплітуду. Формула для DFT виглядає так [42]:

$$X_k = \sum_{n=0}^{N-1} x_n e^{-j \frac{2\pi kn}{N}}$$

де x_n — вхідний сигнал, X_k — перетворений сигнал, N — кількість зразків, j — уявна одиниця.

Після обчислення спектра наступним кроком є пошук локальних максимумів у спектрі, які відповідають пікам частотних компонент [43]. Локальний максимум у точці k визначається як точка, де амплітуда X_k більше, ніж амплітуда сусідніх точок X_{k-1} і X_{k+1} . Це можна формально записати так [42]:

$$X_k > X_{k-1} \quad \text{і} \quad X_k > X_{k+1}.$$

Після визначення локальних максимумів важливо проаналізувати їх амплітуду. Піки з вищою амплітудою зазвичай відповідають основним тонам і гармонікам, тоді як піки з меншою амплітудою можуть бути результатом шумів або інтерференцій. Частота кожного піку обчислюється за формулою [42]:

$$f_k = \frac{k f_s}{N},$$

де f_s — частота дискретизації, N — кількість зразків.

2.2.2 Аналіз формант

Форманти є резонансними частотами мовного тракту, які визначають тембр і акустичні характеристики голосу. Вони відіграють критично важливу роль у розпізнаванні мовлення, оскільки визначають вокальні якості мовного сигналу.

Аналіз формант включає кілька етапів. Спочатку визначаються частотні діапазони, у яких зазвичай знаходяться форманти. Перша форманта (F1) зазвичай розташовується між 200 і 800 Гц, друга форманта (F2) — між 800 і 2500 Гц, а третя форманта (F3) може досягати 3500 Гц і вище. Виявлення формант у цих діапазонах здійснюється за допомогою спектрального аналізу [44].

Форманти визначаються як піки у відповідних частотних діапазонах. Частота форманти f_F визначається як частота піка у спектрі, а ширина форманти Δf_F — як ширина піка на рівні половини його амплітуди (Half-Power Bandwidth). Формально ширина форманти визначається як [44]:

$$\Delta f_F = f_{\text{high}} - f_{\text{low}}$$

де f_{high} та f_{low} — частоти, на яких амплітуда піка зменшується до половини його максимальної амплітуди.

Динамічний аналіз формант дозволяє виявити зміни формант у часі, що відображають перехідні процеси та артикуляційні особливості мовлення. Це досягається шляхом обчислення формант для кожного фрейму сигналу і побудови графіка їх змін у часі [45].

2.3 Класичні підходи до оцінки шуму

Аналіз шумів може включати кілька підходів. Один із підходів полягає у виявленні низькочастотних і високочастотних компонент у спектрі, де часто знаходяться фонові шуми і інтерференції. Низькочастотний шум може бути результатом фонових гулів або структурних вібрацій, тоді як високочастотні шуми можуть виникати через електронні перешкоди або інтерференції. Одним з підходів це застосування частотних фільтрів, наприклад для притлумлення високочастотних шумів [46-48].

Статистичний аналіз шуму є іншим важливим методом, що дозволяє оцінити рівень шуму у спектрі. Один з методів включає обчислення середньоквадратичного відхилення (СКВ) амплітуд спектральних компонент [49]:

$$\sigma = \sqrt{\frac{1}{N} \sum_{k=0}^{N-1} (X_k - \mu)^2}$$

де μ — середнє значення амплітуд спектральних компонент, X_k — амплітуда спектрального компонента.

Спектральна ентропія є мірою нерегулярності або випадковості спектра сигналу і дозволяє оцінити рівень шуму. Вона обчислюється за формулою [50]:

$$H = - \sum_{i=1}^N P(f_i) \log_2 P(f_i)$$

де $P(f_i)$ — нормалізована потужність спектрального компоненту f_i . Високі значення спектральної ентропії свідчать про наявність шумів у сигналі.

Ідентифікація піків, аналіз формант та визначення шумів є ключовими етапами у частотному аналізі мовного сигналу. Використання методів швидкого перетворення Фур'є, статистичного аналізу та спектральної ентропії дозволяє детально аналізувати частотні компоненти сигналу, виявляти основні характеристики голосу та усувати небажані шуми. Це забезпечує високу якість обробки аудіосигналів для різних додатків, включаючи розпізнавання мовлення, музичне продюсування та звуковий дизайн [50].

2.4 Аналіз фундаментальної частоти

Фундаментальна частота визначає основний тон сигналу і є важливим параметром для розпізнавання мови, музичного аналізу та ідентифікації мовця. У контексті визначення гендеру мовця аналіз фундаментальної частоти дозволяє класифікувати голос як чоловічий або жіночий на основі фізіологічних особливостей голосових зв'язок, які впливають на висоту тону [51].

Для визначення фундаментальної частоти використовується метод спектрального аналізу з піковим трекінгом, який дозволяє досліджувати динамічні зміни частотного вмісту сигналу в часі. Цей підхід особливо корисний для аналізу аудіосигналів, музичних творів, мовлення та інших сигналів, де частотні компоненти змінюються з часом. Метод поєднує використання короткочасного перетворення Фур'є (Short-Time Fourier Transform, STFT) та виявлення піків у спектрі для кожного часового вікна, що дозволяє відстежувати еволюцію частотних компонентів сигналу. [51].

Для аналізу нестационарних сигналів використовується STFT, який розбиває сигнал на короткі часові вікна і виконує перетворення Фур'є для кожного з них. Це дозволяє отримати часово-частотне представлення сигналу, де можна спостерігати, як його частотний вміст змінюється з часом. Вибір довжини вікна та ступеня перекриття між вікнами визначається компромісом між часовою та частотною роздільною здатністю: короткі вікна забезпечують кращу часову роздільну здатність, але гіршу частотну, і навпаки [52].

Після отримання спектру для кожного вікна виконується виявлення піків у амплітудному спектрі. Піковий трекінг полягає у визначенні локальних максимумів у спектрі, які відповідають найбільш енергоємним частотним компонентам сигналу. Ці піки можуть відповідати фундаментальній частоті, гармонікам або резонансним частотам. Виявлені піки є основою для подальшого аналізу та відстеження.

Відстеження піків у часі здійснюється шляхом їх зіставлення та порівняння у послідовних часових вікнах. Це дозволяє побудувати траєкторії частотних

компонентів і спостерігати, як їх частоти та амплітуди змінюються з часом. Такий підхід застосовується для аналізу динамічних процесів у сигналі, наприклад, для відстеження мелодійної лінії в музичному творі або зміни фундаментальної частоти в мовленні [53].

Після отримання часово-частотного представлення сигналу за допомогою короткочасного перетворення Фур'є (STFT), спектр сигналу містить велику кількість частотних компонентів з різними амплітудами. Не всі з них є корисними для аналізу; багато з цих компонентів можуть бути результатом шуму, артефактів або гармонік. Тому необхідно виділити ті частотні компоненти, які мають найбільшу значущість з точки зору енергетичного вмісту та інформаційної цінності [53].

Одним із підходів до вибору значущих частотних компонентів є аналіз амплітудних спектрів для кожного часового вікна. У цьому контексті важливо ідентифікувати локальні максимуми або піки в спектрі, які відповідають частотам з найбільшою енергією. Ці піки можуть потенційно відповідати фундаментальній частоті сигналу або її гармонікам. Оскільки фундаментальна частота часто не має найбільшої амплітуди через присутність більш потужних гармонік, необхідно застосовувати додаткові критерії для її ідентифікації [54].

Фільтрація за амплітудою є одним із методів відбору значущих компонентів. Встановлюється певний поріг амплітуди, вище якого частотні компоненти вважаються значущими. Цей поріг може бути визначений на основі статистичних характеристик сигналу або як відсоток від максимальної амплітуди в спектрі. Такий підхід дозволяє відсіяти низькоамплітудні компоненти, які можуть бути результатом шуму або неістотних гармонік [54].

Обмеження частотного діапазону дозволяє звужити аналіз до певного діапазону частот, які є релевантними. Наприклад, при аналізі людського голосу фундаментальна частота зазвичай знаходиться в діапазоні від 50 до 500 Гц. Тому частоти поза цим діапазоном можуть бути ігноровані, що зменшує ймовірність помилкової ідентифікації [55].

Також необхідно врахувати гармонійну структуру сигналу, адже фундаментальна частота є найнижчою частотою в гармонічному ряді, і її гармоніки розташовуються на частотах, кратних їй. Аналіз співвідношень між частотами піків може допомогти визначити, які з них відповідають фундаментальній частоті, а які є її гармоніками. Наприклад, якщо в спектрі присутні піки на частотах 100 Гц, 200 Гц і 300 Гц, можна припустити, що фундаментальна частота становить 100 Гц. При роботі з сигналами з сильним зашумленням застосування статистичних методів може покращити вибір значущих компонентів. Методи кластерного аналізу або головних компонент можуть допомогти ідентифікувати частоти, які найбільше впливають на варіацію сигналу. Фундаментальна частота зазвичай змінюється плавно з часом, тоді як шумові компоненти можуть з'являтися випадково. Аналізуючи, наскільки стабільними є частотні компоненти впродовж часу, можна відсіяти нестабільні піки [56].

Після обчислення середньої фундаментальної частоти шляхом агрегації значень по всьому сигналу або по його значущим фрагментам, отримуючи середнє значення, яке представляє характерну фундаментальну частоту мовця проводиться класифікація гендеру: якщо середня частота нижча за певне значення, наприклад 165 Гц, голос класифікується як чоловічий; якщо вища за інше значення, наприклад 180 Гц, як жіночий; перекриваючий діапазон між цими значеннями може класифікуватися як невизначений або вимагати додаткового аналізу [57, 58].

2.4.1 Застосування фундаментальної частоти

Приклади застосування визначення фундаментальної частоти включають автоматичне визначення нот у музичних інструментах, що використовується при створенні нотного запису або в музичних додатках для навчання гри. У системах розпізнавання мовлення фундаментальна частота використовується для ідентифікації особистості мовця та розпізнавання емоційних станів. У медіаіндустрії аналіз фундаментальної частоти застосовується для редагування звукових доріжок та забезпечення гармонійного звучання композицій. У медицині

аналіз голосу за допомогою фундаментальної частоти дозволяє діагностувати захворювання гортані або інші медичні стани, пов'язані з голосовими характеристиками.

Таким чином, фундаментальна частота є основним параметром, що визначає висоту звуку та відіграє важливу роль у різних сферах, пов'язаних із звуковими сигналами. Точне визначення фундаментальної частоти дозволяє здійснювати ефективний аналіз, обробку та відтворення звуку, що є необхідним для досягнення високої якості в мовній обробці, акустичних системах та інших галузях [54].

Висновки до розділу 2.

У другому розділі визначено ключові особливості обробки аудіофайлів із зашумленням. Розбиття сигналу на фрейми дозволяє локалізувати стаціонарні інтервали в аудіосигналі, забезпечуючи оптимальний баланс між часовою та частотною роздільною здатністю завдяки вибору відповідного розміру фрейму та перекриття між ними. Використання віконних функцій, таких як вікно Ганна або Хеммінга, сприяє зменшенню ефекту витікання спектра, що підвищує точність подальшого аналізу.

Ідентифікація піків у спектрі дозволяє виділити основні енергетичні компоненти сигналу, що є критично важливим для аналізу гармонік та основних тонів. Аналіз формант спрямований на визначення резонансних частот мовного тракту, які формують акустичні властивості голосу, що особливо актуально для систем розпізнавання мовлення. Крім того, класичні підходи до оцінки шуму, включаючи статистичний аналіз та використання спектральної ентропії, дозволяють ефективно виявляти та усувати небажані компоненти, що спричинені шумом.

Аналіз фундаментальної частоти, який здійснюється за допомогою короткочасного перетворення Фур'є та пікового трекінгу, є ключовим для класифікації голосу, зокрема при визначенні гендеру мовця. Відстеження змін

фундаментальної частоти в часі забезпечує точне визначення основного тону сигналу.

Таким чином, визначено підходи до обробки аудіосигналів із зашумленням, що включає розбиття сигналу на фрейми, застосування віконних функцій, ідентифікацію піків, аналіз формант, оцінку шуму та визначення фундаментальної частоти, дозволяє досягти високої якості аналізу та обробки аудіоінформації.

3 ДЕКОДУВАННЯ ТЕКСТУ ТА ПЕРЕДАЧА ДОДАТКОВОЇ ІНФОРМАЦІЇ МЕТОДОМ МЕНШ ЗНАЧУЩОГО БІТА

3.1 Етапи автоматичного розпізнавання мови та приклади систем

Основний принцип роботи програмних систем автоматичного розпізнавання мови (Automatic Speech Recognition, ASR) полягає у декодуванні аудіосигналу, який містить мовні дані, та перетворенні його у текстову форму. Процес розпізнавання мови можна поділити на кілька етапів [59]:

- збір та попередня обробка сигналу: на цьому етапі аудіо записується та очищується від фонових шумів, покращується якість звуку, а також розбивається на короткі сегменти по 15-20 мс, які легше аналізувати.

- виділення ознак (Feature Extraction): аудіосигнал аналізується для виділення характерних ознак, таких як мел-частотні кепстральні коефіцієнти (MFCC), які ефективно представляють основні характеристики мовного сигналу.

- акустичне моделювання: використовуючи акустичні моделі, система визначає ймовірність того, що певні ознаки відповідають конкретним фонемам або звукам мови. Сучасні ASR системи часто використовують глибокі нейронні мережі (Deep Neural Networks) або рекурентні нейронні мережі (Recurrent Neural Networks) для підвищення точності розпізнавання.

- мовне моделювання: мовні моделі допомагають системі передбачити ймовірні послідовності слів на основі контексту. Вони використовують статистичні методи або моделі на основі глибокого навчання, такі як трансформери, для забезпечення граматично коректного та логічного тексту.

- постобробка та виведення результату: на завершальному етапі система комбінує інформацію з акустичного та мовного моделювання, щоб сформулювати остаточний текстовий результат, який відповідає розпізнаній мові [59].

Сучасні технології ASR значно покращилися завдяки розвитку машинного навчання. Наприклад, використання моделей типу Transformer, таких як BERT або GPT, дозволяє досягати високої точності розпізнавання навіть у складних умовах,

де присутні фонові шуми або різні акценти. Крім того, впровадження алгоритмів обробки природної мови (Natural Language Processing, NLP) дозволяє системам ASR краще розуміти контекст та значення слів, що підвищує загальну ефективність розпізнавання.

Однією з ключових переваг ASR є його універсальність та адаптивність. Наприклад, голосові асистенти, такі як Siri, Google Assistant або Amazon Alexa, використовують ASR для забезпечення інтуїтивної взаємодії з користувачами, дозволяючи їм виконувати різноманітні завдання за допомогою голосових команд.

Однак, незважаючи на значний прогрес, ASR стикається з кількома викликами. Одним із головних є забезпечення високої точності розпізнавання у різних умовах, таких як шумне середовище, різні акценти або швидка мова [60].

Vosk є однією з передових бібліотек для розпізнавання мови з відкритим вихідним кодом, яка забезпечує високу точність та ефективність у різних застосунках. Розроблена для роботи на різних платформах, включаючи Windows, Linux, macOS, а також мобільні операційні системи Android та iOS, Vosk пропонує широкий спектр можливостей для інтеграції розпізнавання мови в програмні продукти та сервіси [61].

Відкритий код забезпечує високий рівень гнучкості та адаптивності у використанні. Це дозволяє розробникам вільно модифікувати та розширювати функціональні можливості Vosk відповідно до специфічних вимог своїх проєктів, що сприяє швидкому впровадженню нових функцій, інтеграції з різними системами та пристроями, а також полегшує процес налагодження та оптимізації алгоритмів розпізнавання мови. Можливість роботи в офлайн-режимі значно підвищує адаптивність Vosk у різноманітних умовах застосування, де доступ до Інтернету може бути обмежений або відсутній. Крім того, офлайн-режим гарантує більшу конфіденційність та безпеку оброблюваних даних, оскільки аудіо записи не передаються на зовнішні сервери, що є важливим аспектом для додатків, які обробляють чутливу інформацію [62].

3.2 Кодування тексту за форматом ASCII

ASCII (American Standard Code for Information Interchange) є одним з найпоширеніших класичних методів кодування тексту в електронних системах. Цей стандарт був розроблений ще у 1960-х роках і використовує 7-бітове кодування для представлення 128 символів, що включають літери латинського алфавіту, цифри, знаки пунктуації та спеціальні керуючі символи. ASCII забезпечує кожному символу унікальний числовий код, який легко перетворюється у двійковий вигляд. Наприклад, буква 'A' в ASCII кодується як 01000001, а буква 'a' як 01100001 [63].

Основні принципи ASCII базуються на бінарному представленні даних, де кожен символ тексту представлений у вигляді унікального 7-бітного коду. Це дозволяє кодувати 128 різних символів, що включають літери, цифри, знаки пунктуації та спеціальні керуючі символи. ASCII можна поділити на кілька основних категорій: керуючі символи (0-31), печатні символи (32-126) та символ видалення (127). Керуючі символи використовуються для управління текстовими потоками та пристроями, наприклад, код 10 відповідає символу "перенесення рядка" (LF - Line Feed). Печатні символи включають літери латинського алфавіту, цифри та знаки пунктуації, наприклад, код 32 відповідає символу пробілу, а код 48 — цифрі '0'. Символ видалення (127) використовується для позначення символу видалення [63].

ASCII лишається основою для багатьох сучасних систем кодування, таких як UTF-8, який є сумісним з ASCII. Це означає, що перші 128 символів у UTF-8 збігаються з ASCII, що забезпечує зворотну сумісність і полегшує перехід від старих систем на нові. ASCII також широко використовується у протоколах передачі даних, таких як HTTP та FTP, де важливо мати стандартизовану форму представлення текстових даних [63].

Наукові дослідження продовжують вивчати різні аспекти ASCII та його застосування. Наприклад, робота "ASCII Embedding: An Efficient Deep Learning Method for Web Attacks Detection" описує використання ASCII для створення ефективних моделей глибокого навчання для виявлення веб-атак [64]. Інші

дослідження, такі як "Character-level Chinese-English Translation through ASCII Encoding", досліджують використання ASCII для перекладу з китайської на англійську на рівні символів, що підкреслює гнучкість і універсальність цього методу кодування [65].

Незважаючи на обмеження щодо кількості підтримуваних символів, ASCII залишається незамінним у сучасній комп'ютерній науці та цифровій обробці даних. Його простота, універсальність і сумісність з іншими системами кодування забезпечують йому ключове місце в історії та сучасності цифрових технологій [66].

3.3 Кодування тексту за форматом UTF-8

Кодування тексту за форматом UTF-8 є одним із найбільш поширених і універсальних методів представлення символів у цифрових системах. UTF-8 (Unicode Transformation Format – 8-bit) є частиною стандарту Unicode, який призначений для забезпечення унікального кодування кожного символу незалежно від мови чи платформи, на якій він використовується. Однією з ключових переваг UTF-8 є його сумісність з існуючими системами, які підтримують лише ASCII, оскільки перші 128 символів UTF-8 збігаються з ASCII. Це робить UTF-8 ідеальним вибором для веб-технологій, електронної пошти та інших інтернет-застосунків, де необхідна підтримка багатомовного тексту без зміни існуючих протоколів. Крім того, UTF-8 є ефективним у використанні простору пам'яті для текстів, що складаються переважно з символів, які кодуються одним байтом, що робить його оптимальним для зберігання та передачі великих обсягів текстових даних [67].

UTF-8 кодує кожен символ Unicode у послідовність байтів, кількість яких залежить від кодуової точки символу. Для символів, що знаходяться в діапазоні U+0000 до U+007F, UTF-8 використовує один байт, де перший біт завжди встановлений у 0, а наступні 7 бітів представляють сам символ. Це забезпечує повну сумісність з ASCII, дозволяючи текстам, закодованим у UTF-8, бути читабельними у системах, які підтримують лише ASCII. Для символів з кодуовими точками від U+0080 до U+07FF використовуються два байти. Перший байт

починається з двох бітів 1 і одного бітом 0 (110xxxxx), а другий байт завжди починається з бітів 10 (10xxxxxx). Таким чином, два байти можуть представити до 11 біт інформації, що достатньо для кодування символів кирилиці, грецької, арабської та інших мовних груп [67].

Однією з особливостей UTF-8 є відсутність залежності від порядку байтів (endianess), оскільки порядок байтів у символах є визначеним стандартом. Це робить UTF-8 самосинхронізуючимся, тобто навіть якщо частина тексту пошкоджена або відсутня, залишок тексту може бути правильно прочитаний, що підвищує надійність передачі даних. Крім того, UTF-8 не використовує байт-заповнення (padding), що робить його більш економічним у порівнянні з іншими форматами кодування Unicode, такими як UTF-16 або UTF-32 [67].

Процес конвертації існуючих текстових файлів у формат UTF-8 може бути здійснений за допомогою різних інструментів та програмних засобів. Наприклад, сучасні текстові редактори, такі як Visual Studio Code, Notepad++ або Sublime Text, підтримують збереження файлів у кодуванні UTF-8. Також існують командні утиліти, такі як `iconv` у UNIX-подібних системах, які дозволяють конвертувати файли з одного кодування в інше за допомогою простих команд [67].

Незважаючи на свої численні переваги, UTF-8 може зіткнутися з деякими проблемами під час використання. Однією з таких проблем є необхідність правильної обробки символів із високими кодовими точками, що можуть вимагати більше ресурсів для зберігання та обробки. Крім того, деякі старі системи або програмне забезпечення можуть не підтримувати UTF-8 належним чином, що може призвести до некоректного відображення тексту. Також варто враховувати наявність або відсутність BOM (Byte Order Mark), який у UTF-8 не є обов'язковим, але деякі програми можуть вимагати його для правильного розпізнавання кодування [68].

3.4 Метод LSB (Least Significant Bit) для стеганографії

Метод LSB (Least Significant Bit), або метод найменш значущого біта, є популярним методом стеганографії, який використовується для приховування інформації в цифрових носіях, таких як зображення, аудіо або відео. Цей метод полягає в заміні найменш значущих бітів елементів носія на біти секретного повідомлення, що дозволяє приховати інформацію з мінімальним впливом на якість носія [69].

Для реалізації методу LSB вибирається носій, наприклад, зображення, в якому кожен піксель представлений у вигляді трьох компонентів (червоний, зелений, синій) у бінарній формі. Найменш значущі біти цих компонентів замінюються на біти секретного повідомлення. Наприклад, якщо піксель має червоний компонент 11001010, і потрібно приховати біт '1', то останній біт змінюється на 1, і новий піксель стає 11001011. Цей процес повторюється для всіх бітів повідомлення, що потрібно приховати [70].

Маска нижніх бітів використовується для вибору конкретних бітів у числових кодах даних. У контексті LSB-стеганографії ця маска визначає, які саме біти будуть змінюватися для вбудовування прихованої інформації. Для кращого розуміння, розглянемо приклад: припустимо, що ми маємо 8-бітний код, який представляє значення пікселя зображення, семпла аудіо або іншого числового значення. Кожен біт у цьому коді може мати значення 0 або 1 [71].

Маска нижніх бітів зазвичай подається у вигляді бітової послідовності, де кожен біт вказує, чи буде відповідний біт оригінальних даних змінюватися під час вбудовування інформації. Зазвичай така маска налаштована так, що всі біти, крім найменш значущого, встановлені у значення 1, а найменш значущий біт — у значення 0. Наприклад, для 8-бітного коду маска нижніх бітів може виглядати як «11111110». Це означає, що всі біти, крім найменш значущого, можуть бути змінені для вбудовування даних, тоді як найменш значущий біт залишається незмінним.

Застосовуючи маску, біти текстового повідомлення замінюються на відповідні менш значущі біти пікселів зображення або семплів аудіо. Таким чином,

змінюються тільки ті біти, які найменше впливають на загальне сприйняття файлу, забезпечуючи мінімальний вплив на якість фрагменту [71].

Наприклад, щоб приховати біт інформації у семплі аудіо з 8-бітним представленням, використовуючи маску «11111110», ми замінюємо тільки останній біт цього семпла на біт текстового повідомлення. Інші біти залишаються без змін, що дозволяє зберегти якість аудіофайлу на високому рівні, не викликаючи помітних спотворень для слухача.

Метод LSB має кілька важливих переваг і недоліків. Основними перевагами є простота реалізації та мінімальний вплив на якість носія. Зміна найменш значущих бітів має мінімальний вплив на візуальне або аудіальне сприйняття носія, що робить зміни майже непомітними. Однак, цей метод має і свої недоліки. Носій, модифікований методом LSB, може бути легко пошкоджений під час обробки або стиснення, що призведе до втрати прихованого повідомлення. Крім того, метод LSB дозволяє приховувати лише невеликий обсяг даних порівняно з розміром носія [72].

Висновки до розділу 3.

Наведено основні етапи розпізнавання мови, які реалізовано в сучасних системах автоматичного розпізнавання мовних сигналів. Зазначено особливості використання таких систем, та їх ключові переваги. Виділено основні принципи кодування тексту за форматами UTF-8 та ASCII. Використання формату UTF-8 для кодування тексту забезпечує його сумісність із сучасними системами та можливість передачі текстового фрагменту українською мовою. Дослідженню можливості передачі текстової інформації разом із мовними сигналами за допомогою стеганографічного методу найменш значущого біта (LSB).

4 ПРАКТИЧНА ЧАСТИНА ДОСЛІДЖЕННЯ

4.1 Вибір аудіоінтерфейсу для експерименту

У дослідженні передбачається запис мовного аудіосигналу українською мовою з метою подальшого аналізу та розроблення підходів щодо зменшення шумового забруднення.

Розглянемо для проведення запису мовного аудіо сигналу з дефектами ключові характеристики аудіоінтерфейсу Scarlett Solo 2nd Gen, який забезпечує високу якість запису та точність відтворення звукових сигналів. Однією з основних характеристик є частота дискретизації та розрядність (глибина бітності). Scarlett Solo 2nd Gen підтримує частоту дискретизації до 192 кГц та глибину бітності 24 біт. Частота дискретизації 192 кГц забезпечує можливість точної фіксації навіть найвищих частотних компонентів звуку, що відповідає теоремі Найквіста-Шеннона, яка стверджує, що частота дискретизації повинна бути щонайменше вдвічі вищою за найвищу частоту звукового сигналу для уникнення ефекту аліасінгу. Таким чином, для людського слуху, який охоплює частоти до 20 кГц, частота 192 кГц дозволяє забезпечити безперервне та точне відтворення звуку без втрати важливих деталей [73].

Розрядність 24 біти забезпечує розширений динамічний діапазон до 144 дБ, що значно перевищує стандартну 16-бітну розрядність, яка забезпечує динамічний діапазон близько 96 дБ. Це дозволяє уникнути спотворень при записі тихих ділянок сигналу та забезпечує високу чутливість запису, що є особливо важливим при роботі з конденсаторними мікрофонами, які вимагають високої точності для відтворення тонких нюансів звуку [73].

Передпідсилювачі Scarlett Solo 2nd Gen характеризуються високим коефіцієнтом посилення до 56 дБ, що дозволяє ефективно працювати з мікрофонами з низькою чутливістю або тими, що розташовані на значній відстані від джерела звуку. Це забезпечує необхідний рівень сигналу для подальшої обробки без введення значних шумів чи спотворень. Крім того, система Phantom Power (+48

В) надає можливість живлення конденсаторних мікрофонів, що потребують стабільного постійного струму для роботи внутрішніх електронних кіл. Це забезпечує оптимальну чутливість, точність та збалансованість частотного діапазону, що є необхідним для високоякісного запису [74].

Передпідсилювачі мають рівень гармонічних спотворень нижче 0.001%, що гарантує природне та точне відтворення звуку без внесення небажаних артефактів. Частотна характеристика передпідсилювачів охоплює діапазон від 20 Гц до 20 кГц дозволяє точно передавати як низькочастотні, так і високочастотні компоненти звукових сигналів [74].

Інтерфейс Scarlett Solo 2nd Gen забезпечує співвідношення сигнал/шум (SNR) понад 120 дБ та динамічний діапазон інтерфейсу 106 дБ, що відповідає міжнародним стандартам якості звуку. Високий рівень SNR свідчить про низький рівень шумів у записаному сигналі, що дозволяє отримувати чистий та деталізований звук без паразитних шумів. Динамічний діапазон інтерфейсу забезпечує точне відтворення як тихих, так і гучних звуків без виникнення кліпінгу чи компресії, що дозволяє якісно провести аналіз та подальшу обробку мовного аудіосигналу [74, 75].

У порівнянні з іншими аудіоінтерфейсами, такими як Audient iD4 або PreSonus AudioBox, Scarlett Solo 2nd Gen демонструє вищі показники коефіцієнта посилення (56 дБ проти 50 дБ у Audient iD4) та нижчий рівень гармонічних спотворень (менше 0.001% проти до 0.1% у Audient iD4) [76]. Крім того, інтерфейси Audient iD4 мають нижчі показники частотної характеристики та динамічного діапазону, що робить їх менш придатними для дослідження характеристик сигналу у порівнянні зі Scarlett Solo 2nd Gen. Ці відмінності підкреслюють переваги Scarlett Solo 2nd Gen у забезпеченні високоякісного запису та точності відтворення звукових сигналів, що відповідає міжнародним стандартам.

Таким чином, аналіз характеристик Scarlett Solo 2nd Gen демонструє його здатність забезпечувати високу якість аудіозапису та точність відтворення звукових сигналів завдяки високій частоті дискретизації, розрядності, ефективним

передпідсилювачам з низьким рівнем спотворень та широкому динамічному діапазону.

4.2 Вибір мікрофону для експерименту

Вимоги до мікрофону передбачають наявність комплексу характеристик, що забезпечують точний, чистий та детальний запис звукового сигналу. Одним із ключових параметрів є чутливість, оскільки мікрофон повинен вловлювати навіть дуже слабкі звуки з рівнем не нижче -36 дБ, дозволяючи реєструвати найделікатніші нюанси мовлення. Не менш важливою вимогою є широкий частотний діапазон: від 20 Гц до 20 кГц, що охоплює весь чутний спектр та дає змогу передати як глибокі низькі частоти, так і яскраві високі обертони [77].

Наступним аспектом є налаштування направленості мікрофона, яке дає можливість обрати оптимальну полярну діаграму під конкретні умови запису. Це може бути спрямованість на джерело звуку (кардіоїдна, суперкардіоїдна), всеспрямована або спрямована на відсікання небажаних сигналів з інших кутів падіння звуку. Низький рівень власного шуму мікрофона мінімізує вплив сторонніх шумових складових та спотворень, що особливо критично для запису мовного сигналу зберігаючи його розбірливість та низький рівень шумового забруднення [78].

Крім того, мікрофон має забезпечувати високу деталізацію для аналізу специфічних звуків, зокрема свистячих і шиплячих приголосних у мовленні та характерних вищих гармонік. Така здатність відтворювати дрібні нюанси й тонкі частотні складові дозволяє точно оцінити якість та природу джерела звуку, а також сприяє ефективному опрацюванню аудіоматеріалу у подальшому звуковому ланцюгу.

Таким чином, під час вибору мікрофону необхідно врахувати такі параметри: чутливість, частотний діапазон, спрямованість, тип з'єднання та шумові обмеження. Різні типи мікрофонів мають свої особливості і призначені для конкретних умов експлуатації [79].

Для проведення дослідження було обрано конденсаторний мікрофон Rode NT2-A, який забезпечує високу чутливість (-36 дБ), широкий частотний діапазон і дозволяє проводити аналіз різних спрямованостей мікрофона (кардіоїдна, суперкардіоїдна, омнінаправлена). У поєднанні з професійним аудіоінтерфейсом Scarlett Solo 2nd Gen створюється можливість запису звукових сигналів із високою деталізацією та мінімальними спотвореннями.

Чутливість мікрофона дає змогу фіксувати навіть найтонші звукові нюанси, що важливо для дослідження фонетичних особливостей української мови, таких як свистячі та шиплячі приголосні, що вимагають точної передачі високих частот. Широкий частотний діапазон Rode NT2-A дозволяє записувати як низькі, так і високі частоти з максимальною деталізацією, забезпечуючи повноту інформації для аналізу.

Дослідження передбачає запис звукового фрагмента: “дзига, життя, паляниця, джміль, гава, Харків”. Ці слова обрано для врахування фонетичних особливостей української мови та аналізу потенційно проблемних частот, які потребують обробки.

На звуковій карті Scarlett Solo 2nd Gen налаштовується рівень гучності таким чином, щоб найгучніші звуки не викликали перевантаження запису. Мікрофон встановлюється в необхідний режим спрямованості, забезпечується стабільне положення в просторі, відстань мовця до мікрофону, гучність відтворення.

Цей підхід дозволить проаналізувати вплив спрямованості мікрофона, частотних характеристик і шумового забруднення на якість запису, а також допоможе в дослідженні методів обробки мовних сигналів [80].

4.2.1 Мікрофон з кардіоїдною направленістю

Кардіоїдний мікрофон характеризується спрямованою діаграмою, що дозволяє ефективно ізолювати звук з передньої площини, знижуючи при цьому рівень захоплення фонових шумів з боків і ззаду. Однак, його чутливість до акустичних відбиттів приміщення може призвести до підвищення рівня шуму. Кардіоїдна

направленість мікрофонів застосовується в умовах із мінімальними відбиттями та низьким рівнем фонового шуму, що дозволяє максимально використовувати їх здатність до ізоляції цільового сигналу [81].

Спектрограма (рис. 4.1) має чітко виражені імпульсні ділянки сигналу, де основна енергія зосереджена в області середніх частот із поступовим спадом амплітуди у високочастотному діапазоні. На спектрограмі спостерігаються періодичні яскраві вертикальні смуги, які свідчать про наявність чітких звукових подій або імпульсів. Амплітудно-частотна характеристика (рис. 4.2) запису підтверджує тенденцію до зниження рівня сигналу з ростом частоти, що може бути зумовлено як власною частотною характеристикою мікрофона, так і відсутністю в джерелі звуку потужних високочастотних складових.

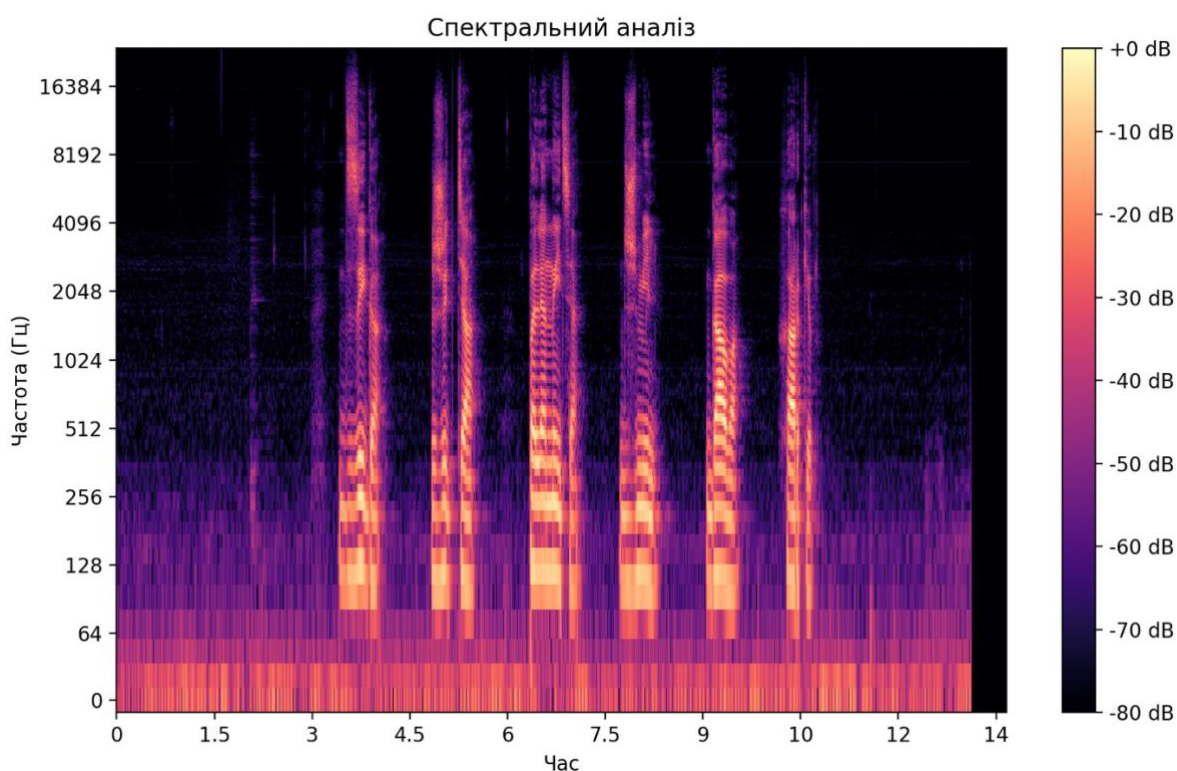


Рисунок 4.1 – Спектральна характеристика кардіоїдної направленості мікрофону

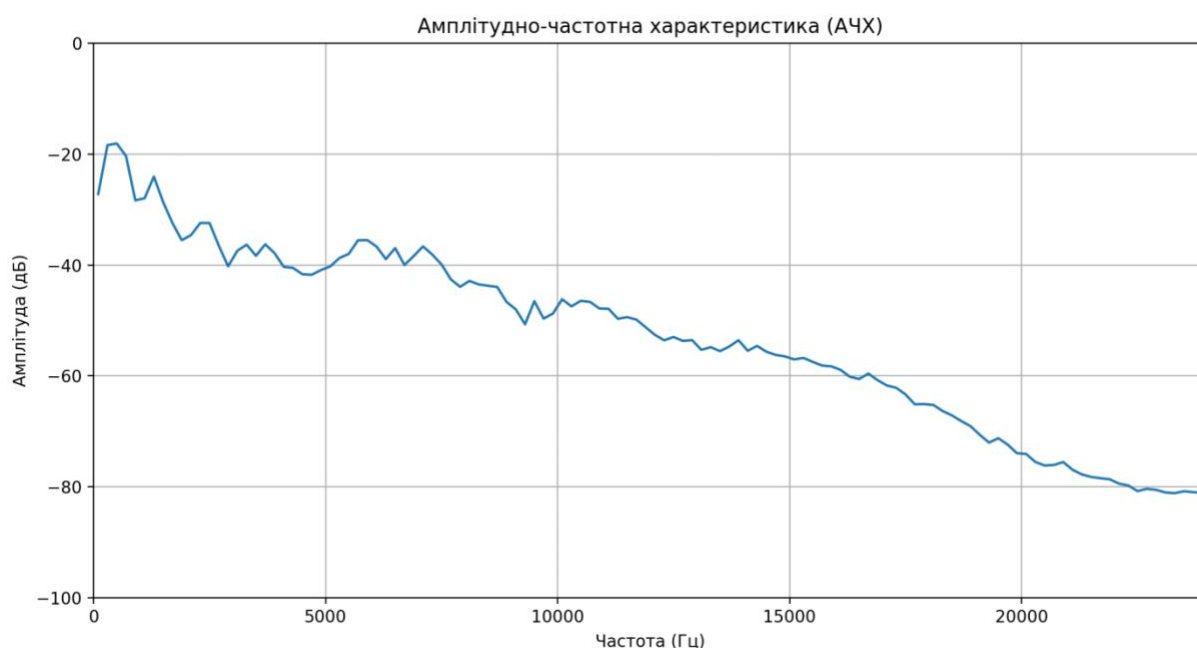


Рисунок 4.2 – АЧХ кардіоїдної направленості мікрофону

Рівень гучності RMS: 0.071339.

Відношення сигнал/шум (SNR): 81.42 дБ.

Середньоквадратичний рівень (RMS) становить 0,071339, що вказує на помірний рівень сигналу без надмірного підсилення, а відношення сигнал/шум на рівні 81,42 дБ свідчить про дуже високий показник чистоти запису і відсутність суттєвого фоново-шумового забруднення. Завдяки кардіоїдній спрямованості мікрофон ефективно придушує звуки, що надходять із задньої півсфери, забезпечуючи фокусування на основному джерелі сигналу та зменшуючи вплив відбитого звуку чи сторонніх завад. Як результат, отримано досить якісний аудіозапис із чітко виділеними імпульсними компонентами, стабільною частотною характеристикою в середньому діапазоні та мінімальним фоновим шумом, що характеризується горизонтальними лініями на спектрограмі (рис. 4.1).

Кардіоїдний мікрофон краще підходить для направлених джерел звуку, таких як голос під час мовлення або подкастів, особливо якщо потрібно захиститися від зовнішніх шумів. Він дозволяє сфокусуватися на основному джерелі звуку, зменшуючи вплив навколишніх перешкод.

1.2.2 Мікрофон типу "фігура-вісімка"

Мікрофон типу "фігура-вісімка" має двосторонню діаграму спрямованості, що дозволяє захоплювати звук з передньої та задньої площин, при цьому ігноруючи бокові напрямки. Це робить його корисним для запису діалогів або ситуацій, де потрібно одночасно захопити кілька джерел звуку, розташованих по обидва боки від мікрофона. Вісімка є оптимальним вибором у студіях з гарною акустикою, де потрібно захопити звуки з двох протилежних напрямків без значного впливу бокових шумів [82].

На спектрограмі (рис. 4.3) можна побачити характерні періодичні імпульсні події, схожі на попередні результати, але слід враховувати особливості цієї діаграми спрямованості. Мікрофони типу "вісімка" мають приблизно однакову чутливість до сигналів, що надходять спереду та ззаду, та мінімальну чутливість до звуків з боків. Така спрямованість сприяє ефективному підхопленню джерел, розташованих по осі мікрофона, але водночас може підкреслити присутність звукових відбитків або інших джерел, розташованих позаду та відбиттів основного сигналу.

На спектрограмі (рис. 4.3) видно чіткі вертикальні смуги підвищеної енергії, розташовані з певним часовим інтервалом. Ці імпульсні події характеризуються значною енергією у нижньому та середньому діапазоні частот з подальшим спадом інтенсивності у високочастотній області. Амплітудно-частотна характеристика (АЧХ) (рис. 4.4) свідчить про поступове зниження рівня сигналу при зростанні частоти, що є типовим для багатьох джерел звуку та умов запису, а також може відображати тональну природу самого сигналу чи особливості приміщення.

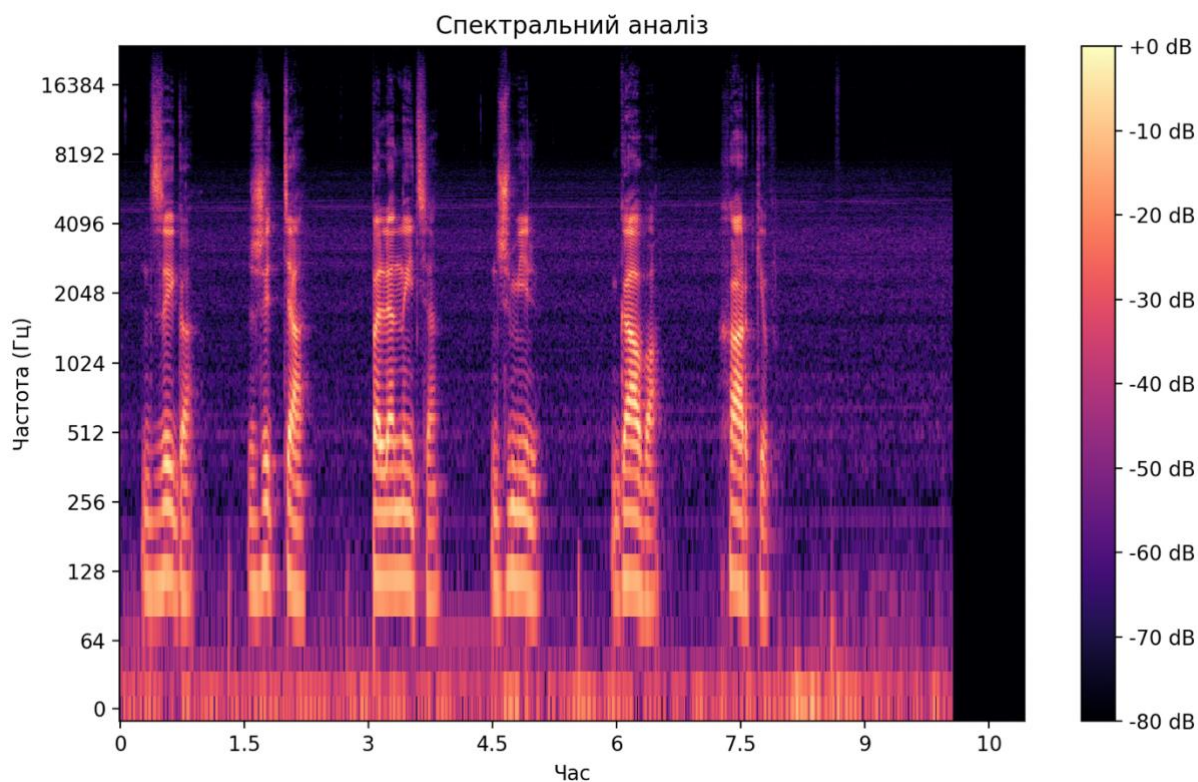


Рисунок 4.3 - Спектральна характеристика направленості мікрофону
«Вісімка»

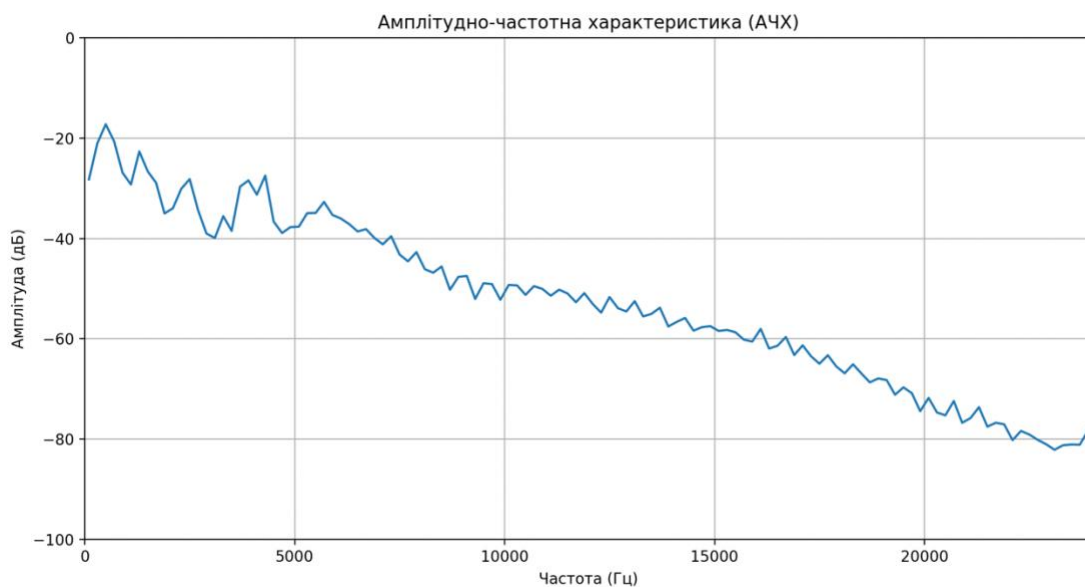


Рисунок 4.4 - АЧХ направленості мікрофону «Вісімка»

Рівень гучності (RMS): 0.068726

Відношення сигнал/шум (SNR): 80.96 дБ

Середньоквадратичний рівень (RMS) у 0,068726 є подібним до попередніх значень, що свідчить про стабільний рівень гучності запису. Відношення

сигнал/шум (SNR) на рівні 80,96 дБ залишається досить високим, однак на спектрограмі можна відслідкувати підвищений рівень фонового шуму майже на всьому спектрі сигналу. Шумові складові та фонова акустика не домінують над корисним сигналом, навіть при тому, що мікрофон “вісімка” може “чути” джерела позаду.

Застосування мікрофона зі спрямованістю “вісімка” в даній ситуації призвело до отримання стабільного, чистого сигналу з періодичними імпульсними подіями та поступовим зниженням енергії на високих частотах. Високе відношення сигнал/шум вказує на те, що навіть при чутливості з тилу вдалося ефективно мінімізувати вплив фонових шумів, зберігаючи природну характеристику джерела та оточення.

4.2.3 Всенаправлений мікрофон

Всенаправлений мікрофон, з третьою найнижчою величиною шуму (0.067139), має діаграму спрямованості, яка дозволяє захоплювати звук з усіх напрямків рівномірно. Це забезпечує максимальне охоплення звукового простору, проте робить його більш вразливим до фонових шумів та відбиттів звуку. Проте, в умовах з контрольованою акустикою, де рівномірне захоплення звуку є перевагою, всенаправлений мікрофон може забезпечити найвищу якість запису. Омнінаправлені мікрофони підходять для запису ансамблів, оркестрів, або запису фонових звуків, де важливо захопити повну картину звукового простору без упереджень у напрямках [83].

Запис ілюструє (рис. 4.5) аналогічні до попередніх експериментів імпульсні звукові події із чітко вираженими вертикальними “стовпами” на спектрограмі. Основна енергія зосереджена у нижньому та середньому частотному діапазоні, при цьому з ростом частоти спостерігається планомірний спад рівня амплітуди, що відображено на амплітудно-частотній характеристиці (рис. 4.6). Всеспрямований мікрофон рівномірно сприймає звук з усіх напрямків, на відміну від кардіоїди чи

“вісімки”, тому він менш схильний до фокусування на одному джерелі та краще фіксує акустичне оточення, включаючи фонові шуми та відбиті сигнали.

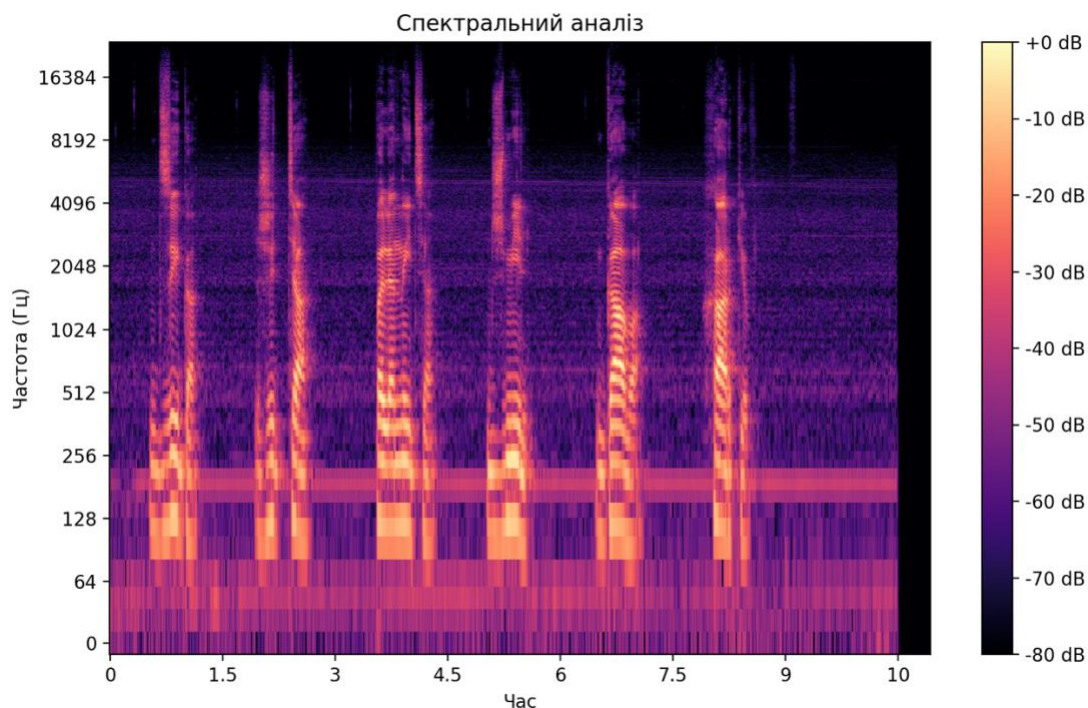


Рисунок 4.5 – Спектрограма всенаправленого мікрофону

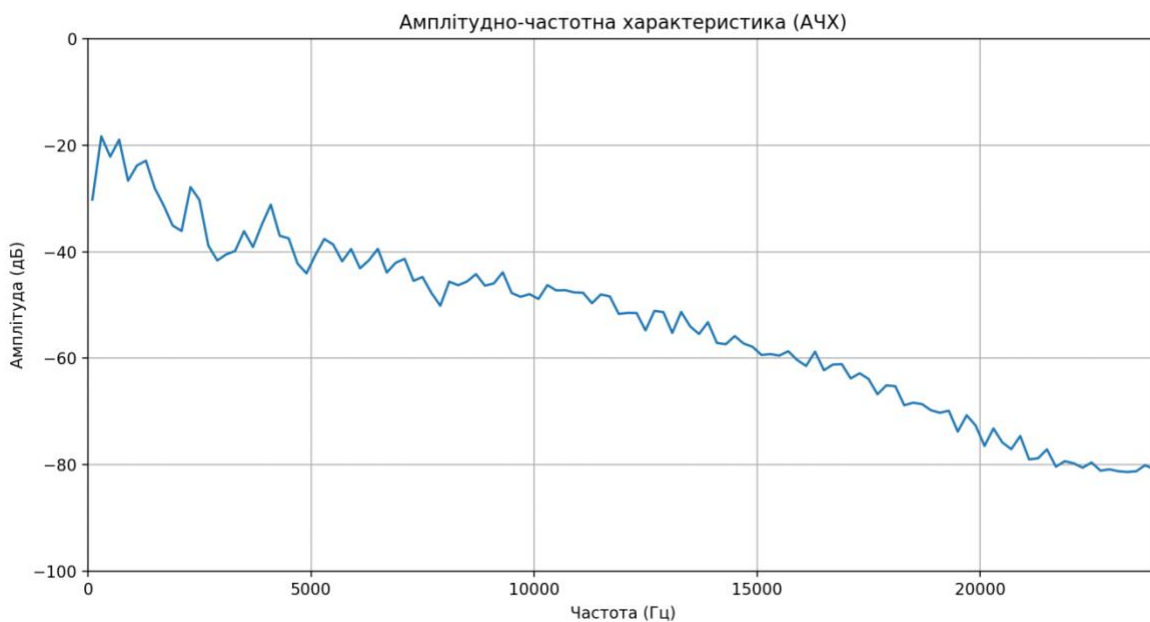


Рисунок 4.6 – АЧХ всенаправленого мікрофону

Рівень шуму (RMS): 0.067139

Відношення сигнал/шум (SNR): 81.80 дБ

Втім, попри чутливість до повного звукового поля, рівень середньоквадратичного значення (RMS) сигналу становить близько 0,067139, що є приблизно на тому ж рівні, що й у попередніх експериментах, і свідчить про схожу середню інтенсивність записаного сигналу. Відношення сигнал/шум (SNR) у 81,80 дБ є навіть дещо вищим за попередні показники, що вказує на дуже низький рівень фонового шуму. Однак, як і в попередньому випадку, помітно незначне шумове забруднення протягом всієї довжини сигналу, що характерно для даної направленості. Такий високий SNR означає, що небажані шуми або інтерференції не істотно вплинули на якість результату, навіть попри збалансовану чутливість мікрофона до всіх напрямків.

Використання омнінаправленого мікрофона у даному випадку забезпечило отримання стабільного, чіткого звукового сигналу з періодичними імпульсними компонентами, відсутністю суттєвих артефактів та мінімальним рівнем шуму, дозволяючи достовірно зафіксувати не лише основне джерело, а й акустичну атмосферу приміщення без втрати загальної якості запису.

4.3 Порівняння мікрофонів

Порівнюючи результати запису (таблиця 4.1) з трьох мікрофонів – із кардіоїдною, вісімковою (бідіредекційною) та всеспрямованою (омнінаправленою) направленостями – можна визначити ключові особливості кожного варіанту та зробити висновок щодо їх оптимального застосування. В основі аналізу лежать дані, отримані зі спектрального аналізу, амплітудно-частотної характеристики (АЧХ) та рівнів шуму кожного з мікрофонів. Кожен із цих мікрофонів має свої особливості, що впливають на якість запису мовлення, особливо враховуючи фонетичні особливості української мови.

Таблиця 4.1 – Порівняння направленостей мікрофону

Тип мікрофона	Рівень гучності RMS	Відношення сигнал/шум SNR	Опис	Особливості
Кардіоїдний	0.071339	81.42 дБ	Мінімальний рівень фонових шумів, виражений горизонтальними лініями	Ефективно приглушує звуки, що надходять з задньої півсфери. Зменшує вплив відбитків та сторонніх завад, забезпечуючи фокусування на основному джерелі сигналу
Вісімка	0.068726	80.96 дБ	Спад інтенсивності у високих частотах, значне шумови забруднення	Може підкреслювати присутність звукових відбитків позаду та інших джерел
Всенаправлений	0.067139	81.80 дБ	Планомірний спад рівня амплітуди при зростанні частоти, значне шумови забруднення	Рівномірно сприймає звук з усіх напрямків. Фіксує акустичне оточення, включаючи фонові шуми та відбиті сигнали

При записі за допомогою кардіоїдного мікрофона спостерігається найнижчий рівень фонових шумів та сторонніх звукових відбитків, що можна пояснити фокусованістю такої діаграми спрямованості. Кардіоїда найбільш чутлива до джерела звуку, розташованого перед нею, і менш чутлива до звуків, які надходять з інших напрямків. Така характеристика дозволяє мінімізувати небажані перешкоди та відбиття від стін, знижуючи загальний рівень шуму демонструючи найкращі характеристики для запису мовлення. Це підтверджується високим відношенням сигнал/шум (SNR) та чітко виділеними імпульсними компонентами

на спектрограмі. Широкий частотний діапазон та стабільна чутливість у всьому спектрі забезпечують точну передачу вихідного сигналу, даючи можливість подальшої коректної обробки (еквалізація, фільтрація, аналіз гармонік тощо).

У випадку з мікрофоном “вісімкою” відсутність концентрації лише на передньому напрямку призводить до уловлювання сигналів як спереду, так і ззаду. Це може бути корисним у стереотехнічних або особливих сценічних умовах, однак не забезпечує такого ефективного пригнічення стороннього шуму, як кардіоїдна направленість, а також забезпечує гіршу розбірливість мови. Мікрофон демонструє слабше захоплення високочастотних компонентів у порівнянні з кардіоїдним. Це може призвести до втрати чіткості шиплячих і свистячих приголосних звуків української мови.

Всенаправлений мікрофон, у свою чергу, сприймає звуки рівномірно з усіх напрямків. Він добре передає природню акустичну картину оточення, але за рахунок цього його показник SNR зазвичай нижчий від кардіоїдного аналогу. Проте, у більш звичайних умовах він вразливіший до фонових шумів, відбиттів і сторонніх джерел, що не завжди є бажаним для подальшої інженерної обробки сигналу. На спектрограмі видно, що він добре захоплює низькочастотні компоненти, але високочастотні деталі, які є важливими для чіткого запису українських шиплячих звуків, є значно слабшими.

З огляду на отримані результати, кардіоїдний мікрофон буде оптимальним варіантом для аналізу та оцінки основного сигналу. Він найкраще пригнічує фонові шуми, надає стабільну та точну передачу звуку в широкому частотному діапазоні, що дозволяє максимально чітко виявити характеристики звукового джерела, виконати якісний спектральний аналіз та підготувати сигнал для подальшої професійної обробки.

Таким чином, для проведення практичного експерименту обрано для запису мовних сигналів мікрофон Rode NT2-A. Цей мікрофон забезпечує високу чутливість (-36 дБ), широкий частотний діапазон (20 Гц – 20 кГц) та низький рівень власних шумів, що дозволяє фіксувати навіть найменші акустичні нюанси мовлення. Завдяки підтримці трьох варіантів направленості (кардіоїдна, "вісімка",

омнінаправлена), мікрофон є універсальним для різних умов запису, забезпечує широкий частотний діапазон, високу чутливість та здатність детально відтворювати звукові характеристики.

4.4 Вибір обладнання

Для перевірки розробленого, у практичній частині дослідження, алгоритму обробки аудіо файлу обрано пристрій Heltec WiFi LoRa 32 V, що поєднує можливості бездротового зв'язку (Wi-Fi, Bluetooth, LoRa) з енергоефективною архітектурою. Його основою є мікропроцесор ESP32, який працює на частоті 240 МГц і має двоядерний процесор Tensilica LX6, що дозволяє виконувати складні обчислення в реальному часі. Пристрій оснащений 520 КБ вбудованої SRAM-пам'яті та має 8 МБ Flash-пам'яті, що забезпечує достатній обсяг для зберігання прошивки та даних [7].

Для бездротової комунікації мікроконтролер підтримує технологію Wi-Fi 802.11 b/g/n, що дозволяє передавати дані через локальні мережі, а також технологію Bluetooth 4.2, яка може бути корисною для підключення до мобільних пристроїв. Крім того, завдяки LoRa-модулю (SX1276) пристрій здатний забезпечити зв'язок на 2-5 км у міських умовах з низьким енергоспоживанням, що робить його ідеальним для застосувань як в домашніх, так і промислових масштабах [7].

Пристрій має 22 GPIO-вивід, а також підтримує 3 інтерфейси UART, 2 I²S, 2 SPI, що дозволяє підключати різноманітні сенсори та периферійні пристрої. Мікроконтролер підтримує роботу з живленням 3.7 В на основі літієвих акумуляторів, а також має вбудований контролер заряду, що дозволяє жити його автономно. У режимі сну споживання енергії складає менше 800 мкА, що робить його придатним для застосувань, які вимагають тривалого автономного режиму роботи. Завдяки двоядерному процесору на 240 МГц, пристрій може виконувати розрахунки в реальному часі [7].

Для запису сигналів обрано мікрофон SPH0645LM4H – це цифровий MEMS-модуль, який дозволяє реєструвати акустичні коливання та передавати їх безпосередньо у цифровому форматі через інтерфейс I²S. Він працює від напруги 1.8-3.3 В. Зручність підключення забезпечується наявністю вбудованого аналого-цифрового перетворювача, тож немає потреби у зовнішніх АЦП чи підсилювачах. Така конструкція дає змогу мінімізувати шуми, оскільки сигнал відразу оцифровується, а передача до мікроконтролера відбувається в цифровому форматі. Частотний діапазон (близько 100 Гц – 10 кГц) і співвідношення сигнал/шум (SNR понад 60 дБ) роблять мікрофон придатним для запису голосу. Завдяки стандартному інтерфейсу I²S, SPH0645LM4H легко під'єднати до Heltec WiFi LoRa 32 V, де потрібно лише налаштувати відповідні виводи (GPIO) для забезпечення синхронізації та передачі даних. Таким чином, цей мікрофон ідеально підходить для вбудованих та IoT-проектів, у яких пріоритетом є чистий аудіосигнал, простота інтеграції та мінімальне енергоспоживання [8].

4.5 Передача мовних сигналів засобами IoT

Технологія LoRaWAN (Long Range Wide Area Network) може забезпечити розроблений алгоритм ефективним каналом зв'язку для передавання обробленого мовного аудіосигналу в середовищі IoT, включаючи ситуації, коли пристрої з мікрофонами чи додатковими сенсорами розташовані на значній відстані від шлюзу, який використовується в рамках технології. Основна перевага LoRaWAN полягає в її здатності забезпечувати стабільний радіозв'язок на великих відстанях (до кількох кілометрів) за умов низького рівня енергоспоживання [9].

Додатковим чинником на користь технології LoRaWAN, є її гнучкість щодо впровадження в інфраструктуру. Один шлюз може обслуговувати значну кількість пристроїв, що робить розгортання такої мережі простішим та економічнішим порівняно з альтернативами на кшталт мереж стільникового зв'язку. Крім того, наявні специфікації LoRaWAN (наприклад, версія 1.1) передбачають декілька класів пристроїв (A, B, C), що відрізняються режимами енергоспоживання та можливостями приймання й передачі. Це допомагає адаптувати алгоритми обробки

аудіосигналу під реальні умови використання. Для пристроїв, що працюють від батареї, можна мінімізувати енергозатрати, виконавши більшу частину розпізнавання мовлення лише за наявності тригера (наприклад, виявлення голосу) [10].

Окрім цього, технологія LoRaWAN може забезпечити базові механізми безпеки, оскільки передбачає двостороннє шифрування даних (як на рівні мережі, так і додатка). При відправленні критично важливих результатів розпізнавання мовних команд (наприклад, для керування обладнанням) це сприяє захисту від несанкціонованого втручання. Водночас, якщо потрібне вбудоване чи стеганографічне шифрування аудіоданих, його можна реалізувати до передачі через LoRaWAN, покладаючись на цей протокол лише як на транспортний шар.

На рисунку 4.7 наведена схема підключення приладів для роботи в середовищі IoT.

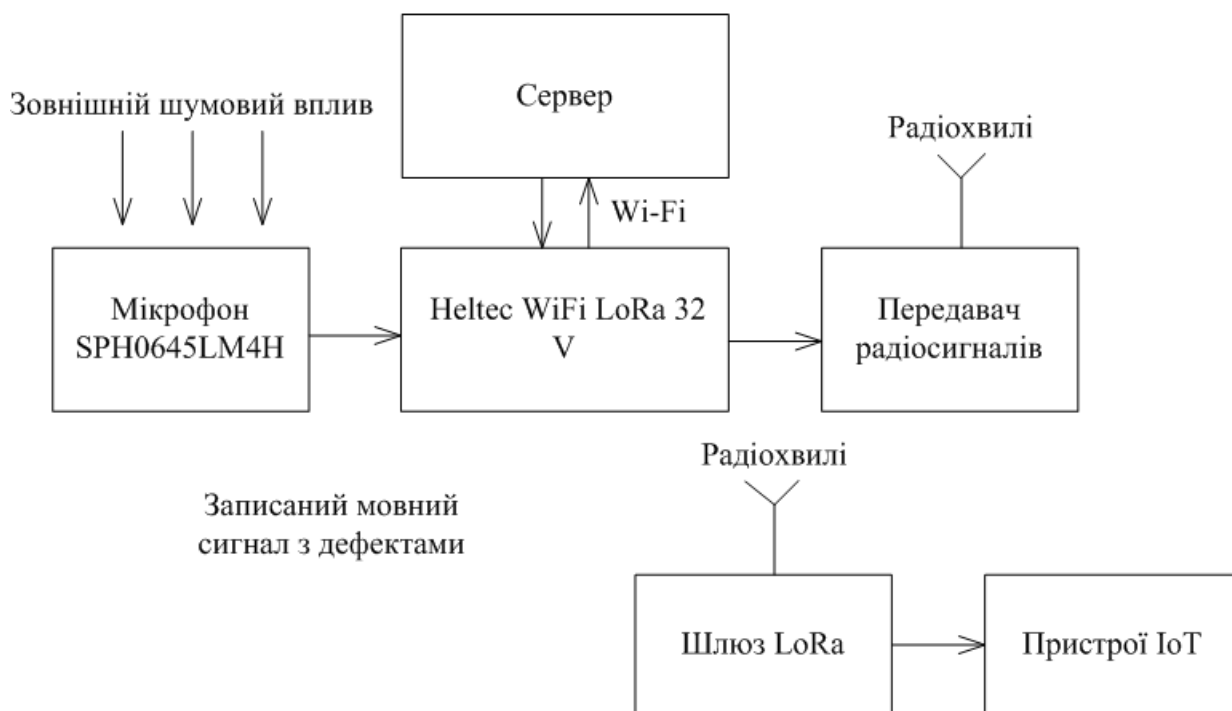


Рисунок 4.7 – Схема каналу радіозв'язку на основі технології LoRaWAN

Пристрій Heltec WiFi LoRa 32 V поєднує можливості двох основних каналів зв'язку: Wi-Fi і LoRaWAN. До нього підключається цифровий мікрофон

SPH0645LM4H, який реалізує функцію захоплення аудіосигналу та передає дані через інтерфейс I²S.

Після захоплення звуку, Heltec застосовує до сигналу розроблений алгоритм обробки для досягнення шумозаглушення та підвищення розбірливості мовлення. Завдяки цьому видаляються лише фонові шуми та мінімізується втрата корисної інформації під дією різноманітних шумів середовища. На основі обробленого сигналу створюється Mel-спектрограма, що забезпечує менший об'єм даних для передачі.

Сформовану Mel-спектрограму, Heltec відправляє на сервер за допомогою каналу передачі Wi-Fi, який забезпечує достатню пропускну здатність для швидкого пересилання бінарних спектрограм або стиснених аудіофрагментів. На серверній частині підключена бібліотека VOSK, і відбувається прийом Mel-спектрограми та відтворення її у вигляді тексту, використовуючи внутрішню акустичну модель, після чого застосовується алгоритм приховування текстової інформації в аудіо сигнал методом менш значущого біта (LSB).

Коли результат розпізнавання мовлення закінчено, сервер визначає, яка команда була озвучена. Це може бути голосове керування певним пристроєм, вимога отримати якусь інформацію або будь-який інший сценарій, передбачений системою. Згідно з визначеною командою, сервер відправляє відповідь на Heltec через Wi-Fi, після чого пристрій, отримавши розпізнану команду, ухвалює рішення щодо подальших дій, а підключенн'я LoRaWAN забезпечує передачу команд на інші IoT-пристрої або мережеві шлюзи, розташовані на відстані кількох кілометрів.

Сервер надає можливість зберігання сигналу для проведення аналітики, навчання чи моніторингу роботи пристрою. Завдяки такій архітектурі створюється гнучке і масштабоване рішення, де початкова обробка здійснюється локально на Heltec, а завдання з розпізнавання мовлення, стеганографії виконуються на сервері, зберігаючи інші пристрої мережі IoT у централізованій інфраструктурі.

Висновки до розділу 4.

1. Виконано комплексний аналіз обладнання, необхідного для запису мовних аудіосигналів з дефектами, та обґрунтовано критерії його вибору для досягнення високої якості обробки та передачі сигналів у системах Інтернету речей (IoT).

1. Визначено та обґрунтовано вибір аудіоінтерфейсу Scarlett Solo 2nd Gen, який підтримує частоту дискретизації 192 кГц і глибину бітності 24 біт, і відповідає міжнародному стандарту якості (ITU-R BS.1534-1). Частотний діапазон передпідсилювачів, коефіцієнт посилення (до 56 дБ) та рівень гармонічних спотворень ($<0.001\%$) забезпечили точність захоплення звукових сигналів без втрати їхніх гармонічних складових. Це дозволило досягти високого співвідношення сигнал/шум ($SNR > 120$ дБ) необхідного для запису мовлення.

2. Обрано мікрофон для роботи з мовним аудіосигналом - конденсаторний мікрофон Rode NT2-A. Цей мікрофон забезпечує високу чутливість (-36 дБ), широкий частотний діапазон (20 Гц – 20 кГц) та низький рівень власних шумів, що дозволяє фіксувати навіть найменші акустичні нюанси мовлення. Завдяки підтримці трьох варіантів направленості (кардіоїдна, "вісімка", омнінаправлена), мікрофон є універсальним для різних умов запису, забезпечує широкий частотний діапазон, високу чутливість та здатність детально відтворювати звукові характеристики.

3. Експерименти з трьома варіантами спрямованості мікрофона продемонстрували, що кардіоїдна спрямованість є найефективнішою для аналізу мовлення, оскільки забезпечує мінімальний рівень фонових шумів і відбиттів, що дозволяє провести якісний аналіз записаного фрагменту та дослідити методи обробки.

4. Визначено обладнання для роботи зі звуком в середовищі IoT. Наведені основні характеристики та канали зв'язку, схема підключення та інтеграції алгоритму обробки мовного сигналу в систему.

5 ОБРОБКА МОВНОГО СИГНАЛУ ЗАПИСАНОГО УКРАЇНСЬКОЮ МОВОЮ З ДЕФЕКТАМИ

Українська мова має багату фонетичну структуру, зокрема в ній є велика кількість шиплячих і свистячих приголосних ([с], [ш], [з], [ж]), які відіграють важливу роль у чіткості мовлення. Ці звуки проявляються у високочастотному діапазоні, і тому важливо, щоб мікрофон був здатен добре передавати ці частоти. Важливим також є ізоляція від фонових шумів, оскільки чистота запису мовлення суттєво впливає на якість відтворення [84].

Також, українська мова характеризується багатством голосних та приголосних звуків, зокрема, значною кількістю щільних приголосних та чіткими голосними, що особливо виразні в діапазоні 1-4 кГц. Це важливо враховувати при обробці звукового сигналу для забезпечення чіткості дикції та природного звучання мовлення.

Перш за все, необхідно застосувати фільтрацію нижніх частот (High-Pass Filter) для усунення шумів нижче 100-150 Гц, які можуть включати фоновий шум від вібрацій або низькочастотні відблиски. Наступним кроком є зменшення зайвих резонансів у середньому частотному діапазоні (200-500 Гц), де часто виникає "гудіння" через акустичні відбиття та нерівномірну акустику приміщення.

Для підкреслення чіткості голосу важливо акцентувати увагу на діапазоні 1-4 кГц, де знаходяться більшість фонем української мови. Адаптивна компресія динамічного діапазону, опис якої було розглянуто в розділ 2 роботи дозволяє підсилити ключові частоти та знизити пікові, забезпечуючи рівномірну природну структуру сигналу, розбірливість та виразність мовлення.

Використання м'якого співвідношення компресії (2:1 або 3:1) дозволяє вирівняти динамічний діапазон голосу, забезпечуючи стабільний рівень гучності та запобігаючи виникненню різких піків. Це особливо важливо для мовлення українською, де варіативність гучності може бути значною через інтонаційні та акцентні особливості.

Додатково, застосування шумозаглушення (Noise Reduction) може допомогти знизити рівень фонового шуму, зберігаючи при цьому натуральність голосу. Важливо здійснювати цю обробку з обережністю, щоб уникнути спотворення або втрати високочастотних деталей, що є важливими для передачі чіткості та виразності мовлення [85].

Крім цього, мовний сигнал українською мовою до обробки був записаний так, щоб в його структурі були наявні технічні дефекти:

- акустичні шуми;
- клацання на клавіатурі;

При записі сигналу був відсутній поп фільтр для обраного мікрофону в рамках експерименту.

Акустичний шум був створений таким чином, що в приміщенні при запису був наявний шум кондиціонера та вентилятора, створювався скрип підлоги в приміщенні.

Саме приміщення, де проводився запис, не було акустично оформлено та підготовлено щодо вибору звукопоглинальних матеріалів та забезпечення потрібного часу реверберації. Тобто при записі в кімнаті наявні відбиття які створюють додаткове відлуння.

5.1 Попередня обробка мовного сигналу

На рисунках 5.1 та 5.2 наведено сигналограму та спектрограму оригінального сигналу. Для врахування фонетичних особливостей української мови на звуковому фрагменті записані слова: “дзига, життя, паляниця, джміль, гава, Харків”.

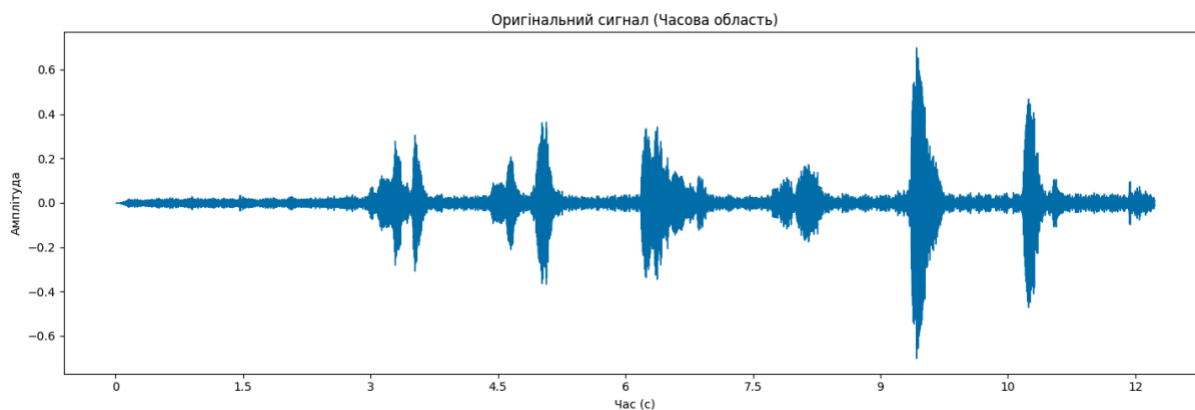


Рисунок 5.1 – Сигналограма оригінального сигналу

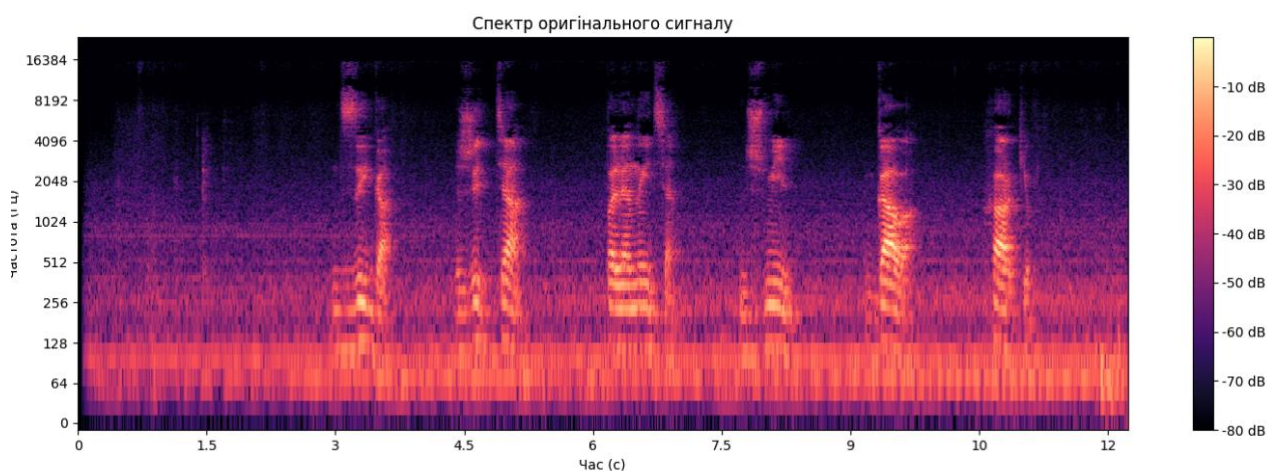


Рисунок 5.2 – Спектрограма оригінального сигналу

На всій часовій області сигналу спостерігається нерівномірність амплітуди, що свідчить про різкі перепади гучності, які ускладнюють сприйняття корисного сигналу, наявна велика кількість шумів, кліків, різких сплесків. Помітно декілька виразних піків із великою амплітудою (приблизно 0,4–0,6), що трапляються в межах від 2 до 12 секунд. Ці різкі стрибки, відповідають виділеним приголосним звукам або акцентованим складам у мовленні, зокрема тим, які пов'язані з виразним видихом (плозивні та фрикативні звуки на зразок [п], [т], [к], [с], [ш] тощо). На часовому проміжку до 1 секунди рівень сигналу незначний, що свідчить про початкову тишу без корисного сигналу.

На спектрограмі (рис. 5.2) можна побачити низку повторюваних вертикальних ділянок із насиченим енергетичним спектром у діапазоні до 3–4 кГц, які чергуються з відносно тихішими зонами. Такі вертикальні смуги відповідають

голосним звукам або сонорним приголосним, так як в голосних зазвичай спостерігається потужна енергія у зоні перших формант (F_1 та F_2) в межах до 1–1,5 кГц, та вище. У верхній частині спектрограми (понад 5 кГц) енергія зосереджена переважно у вибухових та фрикативних приголосних, де утворюються високочастотні компоненти шумового характеру. Також чітко простежується наявність фундаментальної частоти (F_0) у низькочастотному діапазоні (100–200 Гц), звідки вгору поширюються гармоніки, утворюючи характерні для мовлення “сходишки” частотних складових.

5.1.1 Видалення постійної складової

Видалення постійної складової сигналу полягає у вирівнюванні середнього значення амплітуди сигналу до нуля. Це важливо зробити перед обробкою, які спираються на статистичні характеристики сигналу (наприклад, точний розрахунок RMS або аналіз високих гармонік). У випадку з кардіоїдним мікрофоном, усунення постійної складової дозволить уникнути викривлень у подальшій обробці, зокрема при застосуванні динамічних процесорів та фільтрів. Після видалення постійної складової частотний аналіз та робота з компресією або фільтрами будуть більш точними, оскільки сигнал не міститиме штучно зсунутого рівня “нуля”, який може хибно впливати на математичні перетворення [86].

Фільтр постійної складової можна реалізувати за допомогою простих фізичних компонентів. Для усунення DC-компоненти з аудіосигналу використовується конденсатор, який виконує функцію блокування постійного струму, пропускаючи лише змінну складову сигналу. Цей конденсатор встановлюється послідовно в ланцюг передачі сигналу, розташовуючись між джерелом аудіосигналу (наприклад, мікрофоном) і наступним каскадом, таким як звукова карта. Ключовою характеристикою конденсатора є його ємність, оскільки саме вона визначає нижню граничну частоту сигналу, що буде пропускатися через фільтр.

Найнижча частота, яку може створювати голосовий апарат людини, зазвичай становить приблизно 80 Гц. Цей діапазон характерний для чоловічих голосів з

низьким тембром, таких як баси, або в особливих вокальних техніках. Однак, навіть при такій низькій частоті, можуть бути присутні додаткові гармоніки, які впливають на сприйняття тембру та природного звучання голосу. Зважаючи на це, обмежено частоту пропускання до 50 Гц [44].

Отже, необхідно розрахувати ємність конденсатора, який буде фільтрувати обрану частоту, для цього застосовується формула:

$$C = \frac{1}{2\pi f R}$$

де C — ємність конденсатора, f — бажана нижня гранична частота, R — вхідний опір наступного каскаду.

Опір звукової карти становить 3 кОм, бажаний зріз – 50Гц, отримаємо наступне значення ємності конденсатора:

$$C = \frac{1}{2\pi \cdot 50 \cdot 3000} \approx 1.06 \text{ мкФ}.$$

Таким чином, для ефективного фільтрування DC-компоненти з аудіосигналу слід використовувати конденсатор ємністю приблизно 1 мкФ. Найкраще підходять поліпропіленовий або поліестерний конденсатор, оскільки ці типи мають низькі втрати і високу стабільність параметрів в умовах роботи з аудіосигналами. Робоча напруга конденсатора повинна перевищувати максимальну амплітуду сигналу, щоб запобігти його пошкодженню.

Застосування такого фільтра забезпечує очищення сигналу від постійної складової, яка могла б впливати на роботу електронних схем, таких як передпідсилювачі, аналого-цифрові перетворювачі або підсилювачі потужності.

Це дозволяє підвищити якість запису та подальшої обробки звуку, забезпечуючи точне відтворення корисної аудіоінформації без спотворень [87].

5.1.2 Нормалізація сигналу

Після попереднього аналізу, де було встановлено, що запис, отриманий за допомогою кардіоїдного мікрофона, має високе відношення сигнал/шум та досить рівномірний частотний баланс, нормалізація дозволить підкреслити деталі

основного сигналу. Підсилення сигналу при нормалізації не викличе значного збільшення небажаних шумових компонент. Таким чином, нормалізація після використання кардіоїдного мікрофона не тільки збільшить загальний рівень сигналу, але й збереже природну частотну структуру, даючи змогу точніше оцінити динаміку і тембральні особливості запису [88].

Нормалізація амплітуди звукового сигналу полягає в процедурному коригуванні його гучності, орієнтуючись на максимальну амплітуду, щоб найвищий пік сигналу досягав заздалегідь визначеного цільового рівня. Для цього спочатку визначається максимальне абсолютне значення амплітуди, а потім обчислюється масштабуючий коефіцієнт, що дорівнює відношенню цільового рівня до максимальної амплітуди [89]. Кожна вибірка сигналу множиться на цей коефіцієнт, унаслідок чого динамічний діапазон не спотворюється, але сигнал стає "підтягнутим" до заданого рівня. Такий підхід забезпечує уникнення перенасичення (кліпінгу), уніфікує динамічний діапазон різних записів і спрощує їхнє порівняння та аналіз, оскільки різні рівні гучності більше не впливають на результати подальшої обробки [90, 91].

В даному випадку, застосовано коефіцієнт нормалізації 0.7. Це означає, що після нормалізації найвищий пік сигналу буде становити 70% від максимально допустимого значення, створюючи запас гучності, що лишає місце для подальшої обробки (наприклад, застосування ефектів, еквалізації чи компресії), та запобігання перевантаженню на наступних етапах обробки аудіо.

Для мовних сигналів, зокрема україномовної розмовної чи дикторської мови, нормалізація має кілька важливих наслідків. По-перше, якщо початковий запис був занадто тихим, нормалізація допомагає зробити мовний матеріал більш чутним без необхідності збільшувати гучність на відтворювальному пристрої. Тихі приголосні, шепіт, паузи й дихання стають більш помітними, а голосові форманти і гармонічні складові набувають чіткішої звукової форми. Нормалізований рівень гучності може покращити роботу алгоритмів, підвищуючи точність розпізнавання слів і зменшуючи вплив шуму [92].

По-друге, нормалізація створює сприятливі умови для подальшої обробки. Наприклад, після нормалізації можна точніше застосовувати компресори чи еквалайзери, адже весь сигнал знаходиться у передбачуваному динамічному діапазоні. Це дозволяє коректно підібрати параметри ефектів, не зважаючи на надто низький або надто високий початковий рівень [93, 94].

5.1.3 Визначення фундаментальної частоти

Визначення фундаментальної частоти для аналізу мовного сигналу на основі акустичних характеристик його голосу дає можливість подальшої обробки сигналу з врахуванням особливостей голосу спікера. Така необхідність обумовлена значними відмінностями голосового апарату людини та фонетичних особливостей української мови. Фундаментальна частота представляє собою основну частоту, на якій коливається голос, і є ключовим параметром, що відрізняє чоловічі голоси від жіночих [53]. У представленому коді в роботі (додаток А) використовується бібліотека `librosa` для аналізу фундаментальної частоти та визначення гендеру на основі цього параметра. Процес починається з використання функції `librosa.piptrack`, яка здійснює спектральний аналіз сигналу, повертаючи два масиви: `pitches` (фундаментальні частоти) та `magnitudes` (сила сигналу на відповідних частотах) [95].

Далі, для кожного кадру сигналу визначається частота з максимальною амплітудою, що потенційно відповідає фундаментальній частоті. Обираються лише ті частоти, які знаходяться в діапазоні від 50 Гц до 500 Гц, що охоплює типовий спектр фундаментальних частот для більшості мовців чоловічого та жіночого гендеру, уникаючи при цьому надто низьких або високих частот, які можуть бути шумовими або нехарактерними для голосу.

Після збору валідних значень частоти обчислюється їх середнє арифметичне, яке використовується для класифікації гендеру: якщо середнє значення менше 300 Гц, гендер визначається як "Чоловічий", якщо більше 350 Гц – "Жіночий", а якщо знаходиться між цими межами – "Невідомо". При цьому навіть якщо гендер не було

визначено, фундаментальна частота зберігається для застосування в подальшій обробці звукового фрагменту. Обрані параметри діапазону частот та порогів для класифікації базуються на типовій акустичній характеристиці голосів різних гендерів, що дозволяє отримувати коректні результати [57].

Видалення постійної складової перед цим процесом забезпечує більш точний аналіз фундаментальної частоти, оскільки сигнал коливається навколо нульової лінії, що знижує ризик спотворення розрахунків, оскільки додаткова енергія на низьких частотах не буде хибно інтерпретована як низька фундаментальна частота, характерна для чоловічих голосів. Таким чином, визначення фундаментальної частоти не тільки покращує точність аналізу мовного сигналу, але й дозволяє ефективніше використовувати інформацію про гендер для подальшої обробки аудіо, включаючи оптимізацію параметрів фільтру, компресії та інших ефектів [96].

5.1.4 Застосування фільтру високих та низьких частот

Процес фільтрації починається з вибору відповідного типу фільтра та налаштування його параметрів. В даному випадку використовується Баттервортовий фільтр, що забезпечує рівномірну амплітудно-частотну характеристику та відсутність фазових спотворень при двосторонньому застосуванні. Основними параметрами, що визначаються при налаштуванні фільтра, є частота зрізу та порядок фільтра [47].

Частота зрізу визначає межу, за якою фільтр починає пригнічувати або пропускати частоти. Для високочастотного фільтра встановлюється нижня межа пропускання, тобто частоти нижче цієї межі будуть значно зменшені, тоді як для низькочастотного фільтра встановлюється верхня межа пропускання, тобто частоти вище цієї межі будуть приглушені [48].

У наведеному прикладі налаштування частот зрізу залежать від визначеного гендеру мовця. Якщо гендер визначено як "Чоловічий", нижня частота зрізу для високочастотного фільтра обирається як максимальне значення між 50 Гц та 60% від середньої фундаментальної частоти голосу. Це дозволяє ефективно видаляти

низькочастотні шуми, характерні для чоловічих голосів, зберігаючи при цьому основні гармонічні компоненти. Верхня частота зрізу для низькочастотного фільтра встановлюється на 5000 Гц, що забезпечує усунення високочастотних шумів, таких як шипіння або перешкоди, які можуть виникати в записі.

Для "жіночих" голосів нижча частота зрізу для високочастотного фільтра підвищується до 80 Гц або до 60% від середньої фундаментальної частоти, залежно від того, яке значення більше. Це враховує вищу фундаментальну частоту жіночих голосів, дозволяючи краще видаляти відповідні низькі частоти. Верхня частота зрізу для низькочастотного фільтра підвищується до 6000 Гц, що відповідає більш високим гармонічним компонентам жіночих голосів.

У випадку невизначеного гендеру встановлюються стандартні значення: нижня частота зрізу для високочастотного фільтра – 70 Гц, а верхня частота зрізу для низькочастотного фільтра – 6000 Гц. Це забезпечує загальну фільтрацію, яка підходить для широкого спектру голосів, мінімізуючи вплив небажаних шумів без специфічної адаптації до певного гендеру.

Порядок фільтра визначає крутість переходу між пропусканням та придушенням частот. Встановимо порядок фільтра на 1, що забезпечить м'який перехід і менш агресивну фільтрацію. Низький порядок фільтра зменшує ризик спотворення основного звукового сигналу, зберігаючи природну динаміку та гармонічну структуру голосу.

Застосування високочастотного фільтра дозволяє ефективно видалити низькочастотні шуми та небажані низькі частоти, що можуть бути результатом фонових шумів або технічних перешкод. Це покращує чистоту голосу, роблячи його більш чітким і зрозумілим для подальшого аналізу. Низькочастотний фільтр, навпаки, видаляє високочастотні шумові складові, такі як шипіння чи інші перешкоди, що можуть спотворювати сприйняття голосу та ускладнювати точність аналізу [97].

Таким чином, застосування високочастотного та низькочастотного фільтрів з належним налаштуванням параметрів забезпечує ефективне очищення аудіосигналу від небажаних частотних компонентів (рис. 5.3). Завдяки

адаптивному налаштуванню частот зрізу відповідно до гендеру мовця та використанню фільтрів низького порядку, забезпечується баланс між видаленням шумів та збереженням важливих акустичних характеристик голосу, зберігається природнє звучання.

Під час аналізу було визначено: середнє значення pitch: 265.96 Гц.

Визначений гендер на основі pitch: Чоловічий.

Частота зрізу high-pass фільтра: 50 Гц.

Частота зрізу low-pass фільтра: 5000.00 Гц.

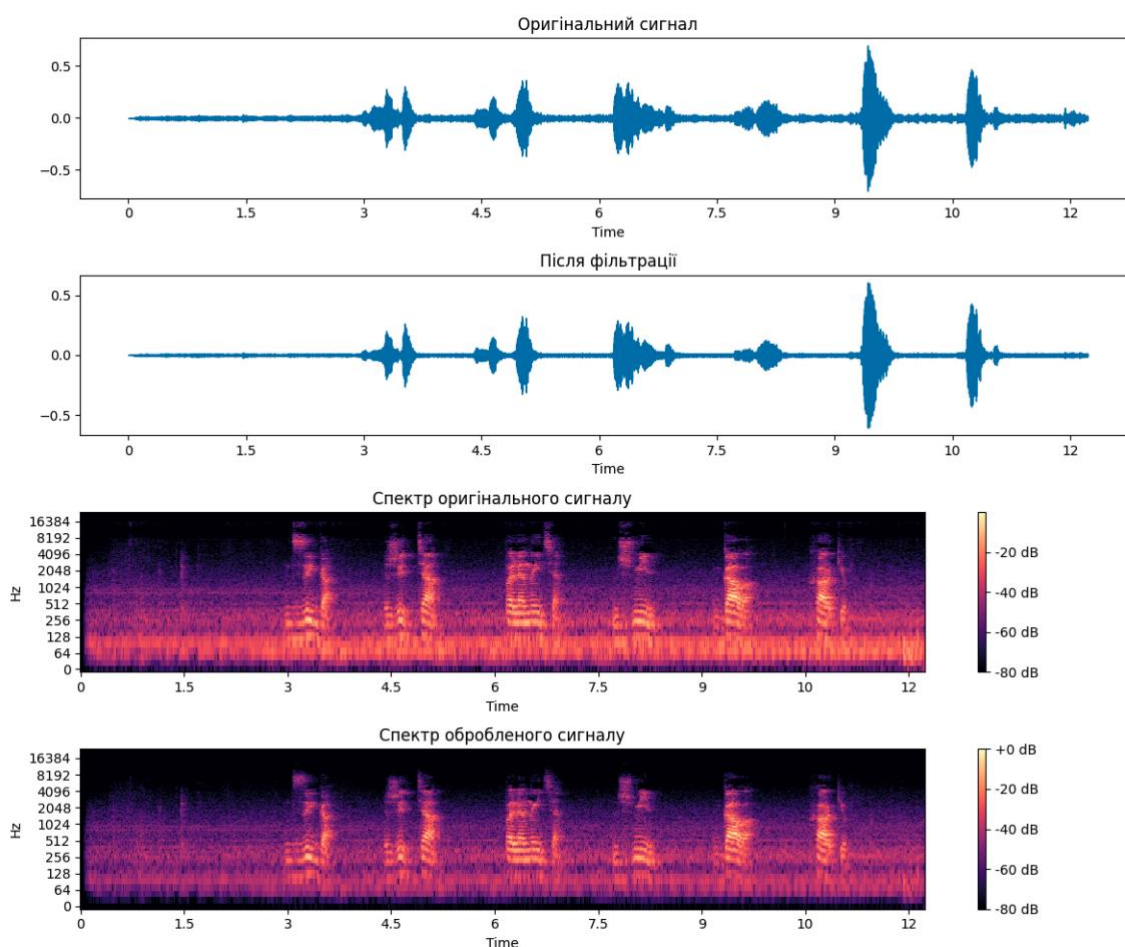


Рисунок 5.3 – АЧХ та сигналограма до та після застосування обробки

В ході обробки мовного сигналу українською мовою було здійснено кілька етапів, що мали на меті покращити його якість для подальшого аналізу. Зокрема, проведено видалення постійної складової, нормалізацію амплітуди, аналіз фундаментальної частоти для визначення гендеру мовця, а також застосування

фільтрації за допомогою низькочастотного та високочастотного фільтрів. Нижче наведено детальний аналіз впливу кожного з цих етапів на оброблений сигнал.

Видалення постійної складової було спрямоване на усунення DC-компоненти, яка призводить до зсуву сигналу відносно нульової лінії. Присутність такої компоненти створює низькочастотний шум, який може негативно впливати на подальші етапи обробки та аналізу сигналу. На спектрограмі після фільтрації видно, що сигнал вирівняний відносно нульової осі, що вказує на відсутність зсуву та кращу симетрію амплітуди.

Нормалізація амплітуди сприяє стандартизації рівня гучності всіх сегментів сигналу, що дозволяє підвищити розбірливість тихіших частин, зберігаючи при цьому цілісність голосових характеристик. На осцилограмі обробленого сигналу амплітуда є більш рівномірною, що сприяє покращенню співвідношення сигнал/шум і забезпечує послідовність в інтенсивності звучання.

Аналіз фундаментальної частоти (pitch) та визначення гендеру дозволив класифікувати мовця як чоловіка на основі середньої частоти pitch, яка становить 265.96 Гц. Хоча це значення є доволі високим для типового чоловічого голосу, класифікація була здійснена з огляду на специфіку фундаментальної частоти та її варіації. Гендерна ідентифікація є важливим етапом, оскільки вона впливає на параметри фільтрації, що, в свою чергу, дозволяє підвищити якість сприйняття мовного сигналу за рахунок адаптації частотних зрізів.

Застосування фільтрації із використанням високочастотного (high-pass) та низькочастотного (low-pass) фільтрів дозволило оптимізувати частотний спектр сигналу, зосередившись на діапазоні, що є значущим для передачі мовної інформації. Зокрема, фільтр низьких частот із частотою зрізу 50 Гц виключив низькочастотний шум, характерний для більшості мовних записів, що дозволило покращити чистоту звуку в низькочастотній зоні. Фільтр високих частот із частотою зрізу 5000 Гц обмежив верхню межу частотного діапазону, що зменшило кількість сибілянтів та інших високочастотних артефактів, які можуть негативно впливати на сприйняття мовлення.

На спектрограмі обробленого сигналу чітко видно зменшення енергетичного спектра в низькочастотній та високочастотній зонах порівняно з вихідним сигналом, що свідчить про успішне видалення компонентів, які не сприяють покращенню розбірливості. Оброблений сигнал сконцентрований на діапазоні частот, важливому для мовної інформативності, що підвищує його якість та знижує рівень перешкод.

Загальний ефект обробки забезпечує очищення від зайвих шумів та більш структурований мовний сигнал, який є придатним для подальшого аналізу або використання, забезпечуючи високу чіткість та інформативність мовлення. Виконані етапи обробки дозволили знизити вплив низько- та високочастотних шумів, а також забезпечити стабільність амплітуди, що, у свою чергу, підвищує ефективність подальших процедур аналізу або передачі мовного сигналу.

5.2 Зниження шумового забруднення та поліпшення розбірливості мови

5.2.1 Зниження шумового забруднення

У процесі покращення якості аудіосигналу застосовано послідовність методів обробки, кожен з яких виконує певну функцію та впливає на кінцевий результат. Порядок застосування цих методів був обраний з урахуванням їх взаємодії та впливу на сигнал, щоб досягти максимальної ефективності та зберегти якість звуку.

Першим етапом необхідно застосувати шумовий поріг (noise gate). Цей метод використовується для видалення небажаних фонових шумів у паузах між словами або фразами, що підвищує чіткість та чистоту звуку. Шумовий поріг працює шляхом зниження гучності сигналу, коли його рівень падає нижче певного порогу. Використання цього методу на початку обробки дозволяє підготувати сигнал для подальших етапів, усунувши фонові шуми, які могли б бути підсилені або вплинути на роботу наступних процесів, таких як компресія. Правильний вибір параметрів

порогу та ступеня послаблення забезпечує ефективне пригнічення шуму без впливу на корисний сигнал, зберігаючи природність звучання [98].

Наступний крок - багатосмугова компресія. Цей метод дозволяє контролювати динамічний діапазон сигналу, зменшуючи інтенсивність піків та зберігаючи загальну динаміку звучання. Компресія застосовується після шумового порогу, оскільки видалення шуму на попередньому етапі запобігає підсиленню небажаних шумових компонентів під час компресії. Багатополосна компресія обробляє різні частотні діапазони окремо, що дозволяє більш точно контролювати динаміку сигналу та уникати спотворень. Це сприяє отриманню більш рівномірного та збалансованого звучання, зменшуючи різкі перепади гучності та покращуючи сприйняття мовлення.

Після компресії необхідно застосувати компресор високих частот (De-Esser) для зменшення інтенсивності сибілянтів, шиплячих та свистячих звуків у високочастотному діапазоні (4000–8000 Гц). Ці звуки можуть бути особливо нав'язливими після компресії, оскільки компресор може підсилювати високочастотні піки. Тому компресор має приглушити небажані частоти та покращити якість мови. Використання компресора високих частот після багатосмугової компресії забезпечує більш ефективне зниження інтенсивності сибілянтів, оскільки вони стають більш вираженими після попередніх обробок. Це дозволяє зберегти чіткість та розбірливість мовлення без втрати якості.

Метод спектрального віднімання дозволяє вибірково зменшити інтенсивність шумових компонентів, віднімаючи спектр шуму від спектру сигналу. Застосування спектрального віднімання після попередніх обробок допомагає усунути залишкові шуми, які могли залишитися або стати більш помітними після багатосмугової компресії та компресії високих частот. Це сприяє подальшому покращенню чистоти звуку без суттєвого впливу на корисний сигнал. Правильний вибір зразка шуму та налаштування параметрів дозволяє ефективно знизити рівень шуму без виникнення артефактів.

Завершальним етапом є нормалізація гучності до міжнародного стандарту ITU-R BS.1770. Це дозволить привести сигнал до цільового рівня гучності -23

LUFS, що забезпечує стабільний та комфортний рівень звучання для слухача. Нормалізація гучності важлива для того, щоб уникнути різких перепадів гучності при відтворенні різних аудіоматеріалів та забезпечити відповідність професійним стандартам. Це також допомагає захистити сигнал від перевантаження та спотворень, що можуть виникати при надмірній амплітуді.

Порядок застосування цих методів був вибраний з урахуванням їх взаємодії та впливу на сигнал. Початкове блокування шуму забезпечує чисту основу для подальшої обробки, запобігаючи підсиленню шуму на етапі компресії. Компресія, застосована після шумового порогу, контролює динамічний діапазон, роблячи звук більш рівномірним. Компресія високих частот використовується після багатосмугової компресії, оскільки сибілянти можуть стати більш вираженими після стиснення динаміки. Спектральне віднімання та додаткове гейтування шуму допомагають усунути залишкові шуми та артефакти, забезпечуючи максимальну чистоту сигналу. Нормалізація гучності завершує процес, приводячи рівень сигналу до стандарту та забезпечуючи комфортне прослуховування.

5.2.2 Шумовий поріг

На першому етапі обробки аудіосигналу було застосовано шумовий поріг. Цей етап дозволяє ефективно знизити низькорівневі шуми, які присутні в паузах між словами або фразами, що підвищує чистоту звучання та розбірливість мови, знижує вплив середовища в якому був зроблений запис [99]. Попереджує підсилення небажаних шумів у наступних процесах, таких як компресія. Це забезпечує більш точну та ефективну роботу обробок без ризику виникнення додаткових артефактів [100].

Визначаються основні параметри функції: час спрацювання (`attack_time`) встановлюється на 0.005 секунд (5 мілісекунд), що визначає, як швидко поріг переходить від приглушеного стану до повної гучності, коли сигнал піднімається вище порогу. Час відновлення (`release_time`) встановлюється на 0.05 секунд (50 мілісекунд), що визначає, як повільно поріг повертається до приглушеного стану,

коли сигнал опускається нижче порогу. Ступінь приглушення (`attenuation_db`) встановлено на -15 дБ, що означає, що сигнал, який падає нижче порогу, буде зменшено на 15 дБ. Тривалість плавного переходу (`ramp_duration`) встановлюється на 0.05 секунд (50 мілісекунд), що забезпечує природний і безперебійний перехід між різними рівнями гучності без різких стрибків.

Для початку необхідно визначити оголовний контур сигналу (рис. 5.4), який визначає, як змінюється амплітуда з часом. Оголовний контур реагує на зміни амплітуди сигналу з часами атаки та релізу, що дозволяє швидко піднімати гучність сигналу до повного рівня протягом 5 мс, коли сигнал перевищує поріг, і повільно знижувати гучність до заданого рівня приглушення протягом 50 мс, коли сигнал падає нижче порогу.



Рисунок 5.4 – Оголовний контур та шумовий поріг

Після визначення оголового контуру функція автоматично визначає поріг відсікання шуму (`threshold_db`) згідно з виміряним фрагментом сигналу довжиною 0.5с. та підвищує його на 12 дБ, що дозволяє перекрити коливання рівня шуму в майбутньому. Наприклад, якщо середньоквадратичне значення шуму становить -50 дБ, поріг гейтування буде встановлено на -38 дБ. Це забезпечує, що тільки значні аудіосигнали, які перевищують цей поріг, проходять без приглушення, тоді як тихі частини сигналу, що містять фоновий шум, приглушуються. Далі функція застосовує маску (рис. 5.5) до сигналу.

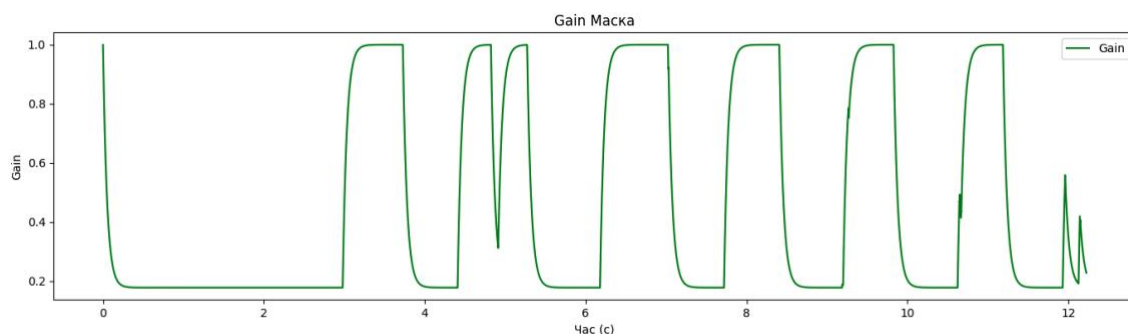


Рисунок 5.5 – Маска сигналу

Ініціалізація маски починається з 1.0, що означає відсутність зміни гучності сигналу. Під час обробки кожного семплу сигналу функція порівнює рівень оголового контуру з порогом шуму. Якщо рівень сигналу перевищує поріг, сила пригнічення поступово піднімається до 1.0 протягом часу атаки, забезпечуючи плавний перехід до повної гучності. Якщо рівень сигналу падає нижче порогу, сила приглушення поступово знижується до значення, відповідного ступеню приглушення (-15 дБ) протягом часу відновлення, знову ж таки забезпечуючи плавний перехід.

Цей підхід дозволяє ефективно приглушувати фонові шуми, не впливаючи на основний аудіосигнал. Плавні переходи між різними рівнями гучності запобігають створенню артефактів або відчуття тріску в аудіо, забезпечуючи більш природний і чистий звук. Наприклад, коли амплітуда сигналу піднімається до -30 дБ, інтенсивність пропускання шуму швидко піднімається до 1.0, дозволяючи цьому сигналу пройти без приглушення. Коли ж амплітуда опускається до -40 дБ, інтенсивність плавно знижується до -15 дБ, ефективно приглушуючи тихі шуми.

Однією з переваг цього підходу є можливість точно контролювати рівень приглушення та поріг шумозниження, що дозволяє адаптувати функцію під конкретні потреби та характер сигналу. Однак, важливо правильно визначити шумовий патерн, оскільки якщо шумовий патерн включає корисний сигнал, обчислення відношення сигнал/шум буде некоректним, що може призвести до неправильного налаштування порогу гейтування. Після застосування функції шумового порогу результати обробки сигналу демонструють помітні зміни в

часоамплітудній області. Оригінальний сигнал (рис. 5.6) містить шумові компоненти в тихих частинах, які можна чітко спостерігати у вигляді невеликих флуктуацій амплітуди на ділянках із низькою енергією. Це характерно для фонових шумів або слабких звукових перешкод, які можуть негативно впливати на подальший аналіз або обробку аудіосигналу.

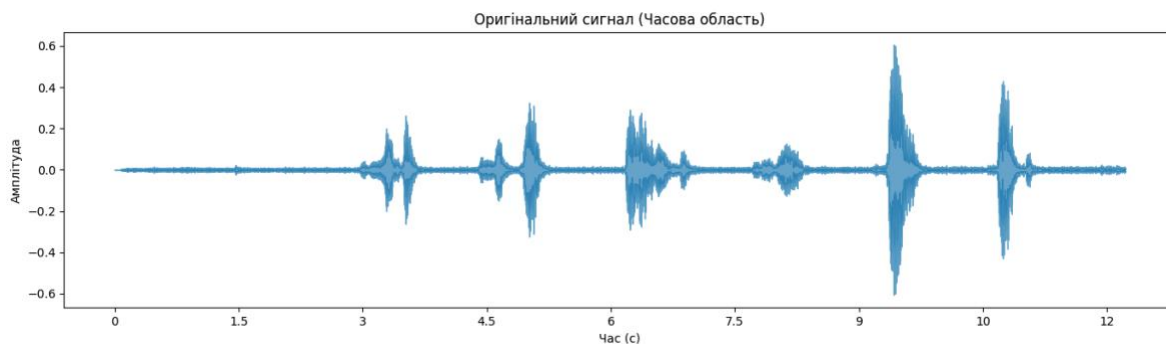


Рисунок 5.6 – Часово амплітудна характеристика вхідного сигналу

Після застосування шумового порогу (рис. 5.7) амплітуда сигналу в цих низькоенергетичних ділянках значно зменшилася, наближаючись до нуля. Наприклад, у ділянках між 1.5 та 3 секундами, а також між 7 та 9 секундами, шумовий поріг ефективно знижує амплітуду до рівня, де фоновий шум стає практично невлотимим. Це свідчить про те, що поріг гейтування був налаштований на рівень, який дозволив відфільтрувати тихі частини сигналу, залишаючи незмінними енергетично значущі ділянки.

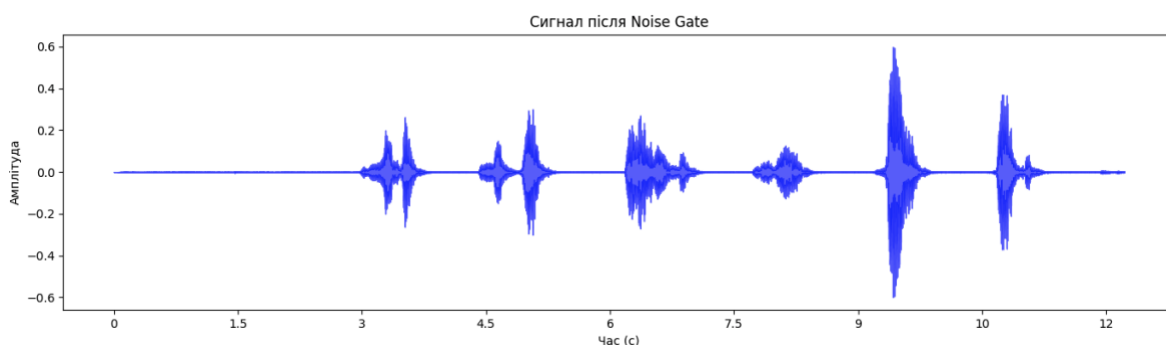


Рисунок 5.7 – Часово амплітудна характеристика сигналу після шумової обробки

Важливо зазначити, що енергетично значущі ділянки сигналу, такі як піки, що спостерігаються, наприклад, між 3 та 4 секундами, залишилися незмінними після

обробки. Це підтверджує, що поріг шуму не вплинув на основний контент аудіосигналу, а лише усунув небажані низькоенергетичні компоненти. Отже, основний сигнал зберіг свої характеристики, тоді як фоновий шум був ефективно знижений, що підвищило загальну якість аудіосигналу і зробило його більш придатним для подальшого аналізу та обробки.

5.2.3 Багатосмугова компресія

На другому етапі обробки аудіосигналу було застосовано багатосмугову компресію (Multi-Band Compression) для контролю динамічного рівня сигналу в різних частотних діапазонах. Це дозволяє зменшити інтенсивність частотних максимумів, зберігаючи при цьому загальну динамічність та природність звучання. Згідно розділу 2.1, такий підхід забезпечує більш рівне та збалансоване звучання, зменшення різких перепадів гучності та підвищення контрольованості сигналу, що в свою чергу покращує сприйняття мовлення та комфорт слухача [101].

Спочатку сигнал розбивається на частотні смуги за допомогою смугових фільтрів Баттерворта, які виділяють конкретні частотні діапазони. У цьому випадку використовуються три частотні смуги: низькі частоти (20–250 Гц), середні частоти (250–4000 Гц), і високі частоти (4000–20000 Гц). Цей поділ дозволяє працювати з кожною смугою окремо, враховуючи їх специфіку. Наприклад, низькі частоти зазвичай містять басові звуки, середні — людський голос, а високі — сибілянти або високочастотні деталі.

Після виділення частотної смуги виконується компресія сигналу. Основним параметром компресора є поріг компресії (`threshold_db`), який встановлено на рівні -20 дБ. Це означає, що сигнал із амплітудою, що перевищує цей рівень, піддається компресії. Використання такого порогу дозволяє ефективно обмежувати динаміку в гучних частинах сигналу, не впливаючи на тихі ділянки. Співвідношення компресії (`ratio`) встановлено на 4:1, що означає, що перевищення порогу на 4 дБ буде зменшено до 1 дБ. Це помірна компресія, яка допомагає згладити динаміку без втрати природного звучання.

Час спрацювання (attack) встановлено на 0.01 секунди (10 мілісекунд), що дозволяє компресору швидко реагувати на різкі зміни амплітуди, наприклад, на короткі пікові звуки. Час відновлення (release) дорівнює 0.1 секунди (100 мілісекунд), що забезпечує плавний перехід компресора до нормального стану після спаду амплітуди, запобігаючи раптовим змінам рівня гучності.

Однією з особливостей функції є використання технології Soft Knee, яка додає плавний перехід між зонами без компресії та активною компресією. Ширина цього переходу (knee_db) становить 10 дБ, що дозволяє уникнути різких змін у динаміці сигналу, роблячи обробку більш непомітною для слухача. Це особливо важливо для звуків, амплітуда яких знаходиться близько до порогу компресії.

Після компресії в кожній частотній смузі застосовується компенсація посилення (makeup_gain), встановлена на рівні 5 дБ. Це компенсує втрату гучності, викликану компресією, і дозволяє зберегти загальний рівень гучності сигналу. Компенсація є важливою, оскільки компресія завжди зменшує амплітуду гучних частин сигналу.

Після обробки всіх частотних смуг отримані сигнали додаються разом, формуючи фінальний компресований сигнал. На завершальному етапі сигнал нормалізується для запобігання кліпінгу. Максимальна амплітуда сигналу масштабується до 99% від максимально допустимого рівня, що забезпечує оптимальне використання динамічного діапазону без спотворень.

Використання таких налаштувань забезпечує оптимальну роботу компресора для широкого спектра аудіосигналів. Низькі частоти (20–250 Гц) обробляються для контролю басових звуків, середні частоти (250–4000 Гц) — для чіткого й рівного звучання голосу, а високі частоти (4000–20000 Гц) — для зменшення сибілянтів та інших різких звуків. Такий підхід дозволяє досягти збалансованого, чіткого та насиченого звучання з природною динамікою, що особливо важливо для обробки мовного сигналу та покращення розбірливості.

Після застосування компресії до аудіосигналу (рис. 5.8) спостерігається вирівнювання його динамічного діапазону, що є ключовою метою цього етапу обробки. На графіку сигналу після компресії видно, що амплітуди гучних ділянок

зменшені, а тихі частини підняті, завдяки чому сигнал стає більш збалансованим і рівномірним.

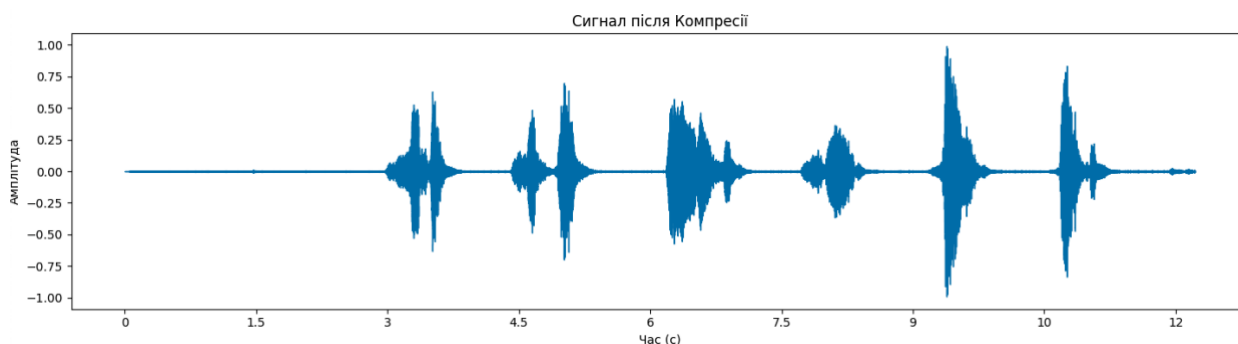


Рисунок 5.8 – Амплітудно-часова характеристика сигналу після компресії

Компресія була виконана з порогом компресії, встановленим на рівні -20 дБ. Це означає, що всі частини сигналу, амплітуда яких перевищувала цей рівень, піддалися компресії. Наприклад, пік сигналу між 9 і 10 секундами, який до компресії досягав 0.6 (нормалізованих одиниць амплітуди), після компресії був пропорційно збільшений до близько 0.7. Це свідчить про те, що компресор зменшив рівень найгучніших частин, не дозволяючи їм перевищувати прийнятний динамічний діапазон, при цьому підвищив загальний рівень корисного сигналу.

Результат компресії демонструє суттєве покращення сприйнятності сигналу. Тихі частини, які раніше були майже неслухними (амплітуда нижче 0.1 нормалізованих одиниць), піднялися до рівня 0.2–0.3, що робить їх більш помітними. Водночас гучні піки, не перенавантажують сигнал, зберігаючи природність і динаміку звучання.

Загальний ефект компресії полягає у створенні збалансованого звуку, який зберігає природність, але стає більш придатним для подальшої обробки або використання. Такий підхід дозволяє уникнути кліпінгу, вирівнює динаміку та забезпечує розбірливість мовного сигналу.

5.2.4 Компресія високих частот

На третьому етапі обробки аудіосигналу було застосовано компресію високих частот (De-Esser), спеціалізований інструмент для зменшення інтенсивності сибілянтів — свистячих та шиплячих звуків, які виникають у високочастотному діапазоні, приблизно між 4000 і 8000 Гц. Сибілянти можуть створювати дискомфорт при прослуховуванні, оскільки вони часто звучать різко та нав'язливо, особливо в умовах високої якості відтворення або при використанні навушників. Застосування такої компресії допомагає покращити якість звучання мовлення, роблячи його більш приємним та легким для сприйняття [102].

Для реалізації сигнал спочатку аналізується у визначеному діапазоні частот для виявлення максимальних значень амплітуди, які є ознаками сибілянтів. Діапазон від 4000 до 8000 Гц було поділено на сегменти по 1000 Гц, що дозволяє точніше локалізувати частотні області, де сибілянти проявляються найбільш інтенсивно. У кожному сегменті здійснювався пошук найвищого піка, амплітуда якого порівнювалася з порогом $\text{peak_threshold} = 0.1$. Якщо нормалізована амплітуда піка перевищувала цей поріг, він визначався як потенційно проблемний сибілянт, який потребував обробки. Такий підхід дозволяє ідентифікувати лише ті піки, що суттєво виділяються над загальним рівнем сигналу, підвищуючи ефективність та точність обробки.

Для зменшення інтенсивності виявлених сибілянтів застосовувався параметричний смуговий фільтр (Notch Filter) із шириною смуги $\text{bandwidth} = 100$ Гц. Вузкий фільтр дозволяє вибірково ослабити амплітуду частотного компонента в обмеженому діапазоні, де було виявлено пік сибілянтів, без істотного впливу на сусідні частоти. Це важливо для збереження природності звучання та уникнення впливу на інші важливі частотні компоненти мовлення. Зменшення інтенсивності відбувалося з коефіцієнтом $\text{reduction_factor} = 0.5$, що відповідає зниженню гучності в обробленому діапазоні на 50%. Такий рівень ослаблення ефективно приглушує сибілянти, роблячи їх менш різкими та нав'язливими, але не повністю їх видаляє, що зберігає природність мовлення.

Поділ діапазону на сегменти по 1000 Гц дозволяє точніше ідентифікувати частоти, де сибілянти найбільш виражені, що підвищує ефективність фільтрації та знижує ризик впливу на інші частоти. Поріг `peak_threshold = 0.1` забезпечує відсіювання незначних піків та фокусується на найбільш проблемних сибілянтах, допомагаючи уникнути надмірної обробки та зберегти природність звуку. Ширина смуги `bandwidth = 100` Гц є достатньо вузькою, щоб вибірково ослабити конкретні частотні компоненти, мінімізуючи вплив на сусідні частоти. Коефіцієнт `reduction_factor = 0.5` забезпечує збалансоване зниження інтенсивності сибілянтів, роблячи їх менш різкими, але не видаляючи повністю, що зберігає природність мовлення.

Після застосування деесера спостерігаються суттєві зміни амплітудно-часової характеристики (рис. 5.9), а саме зниження амплітуди піків на частотах 4000-8000 Гц, що підтверджують ефективність обробки сибілянтів. У початковому сигналі високочастотні компоненти характеризувалися нерівномірною амплітудою сигналу по всій довжині запису. Це створювало акустичний дисбаланс, підсилюючи шумові елементи сигналу, що потенційно ускладнювало його подальшу обробку та сприйняття.

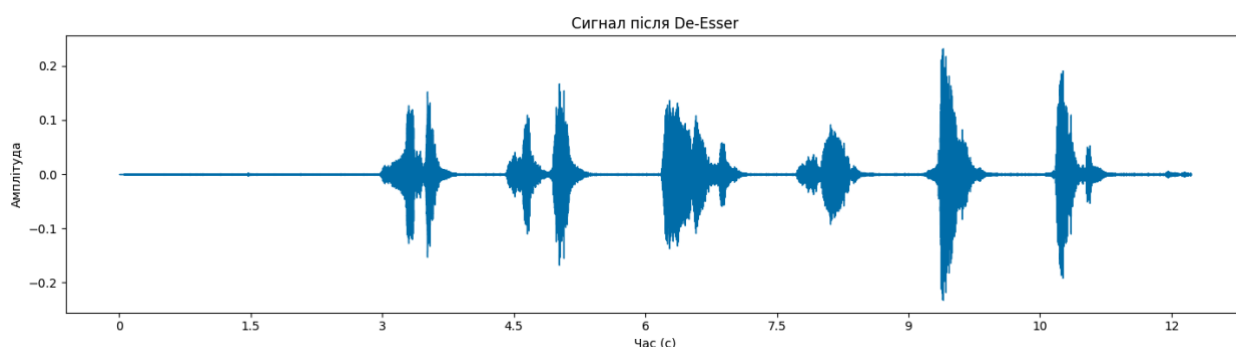


Рисунок 5.9 – Амплітудно часова характеристика обробленого сигналу

Аналіз спектрограми обробленого сигналу (рис. 5.11) та сигналу до обробки (рис. 5.10) показав, що високочастотні компоненти в діапазоні 4000–8000 Гц стали менш інтенсивними, особливо в тих сегментах, де раніше спостерігалися виражені піки. Це свідчить про ефективне ослаблення сибілянтів без втрати основної інформації мовлення. При цьому структура сигналу в середньочастотному (500–

4000 Гц) та низькочастотному (до 500 Гц) діапазонах залишилася практично без змін, що підтверджує вибірковість застосованої обробки та відсутність негативного впливу на інші компоненти сигналу.

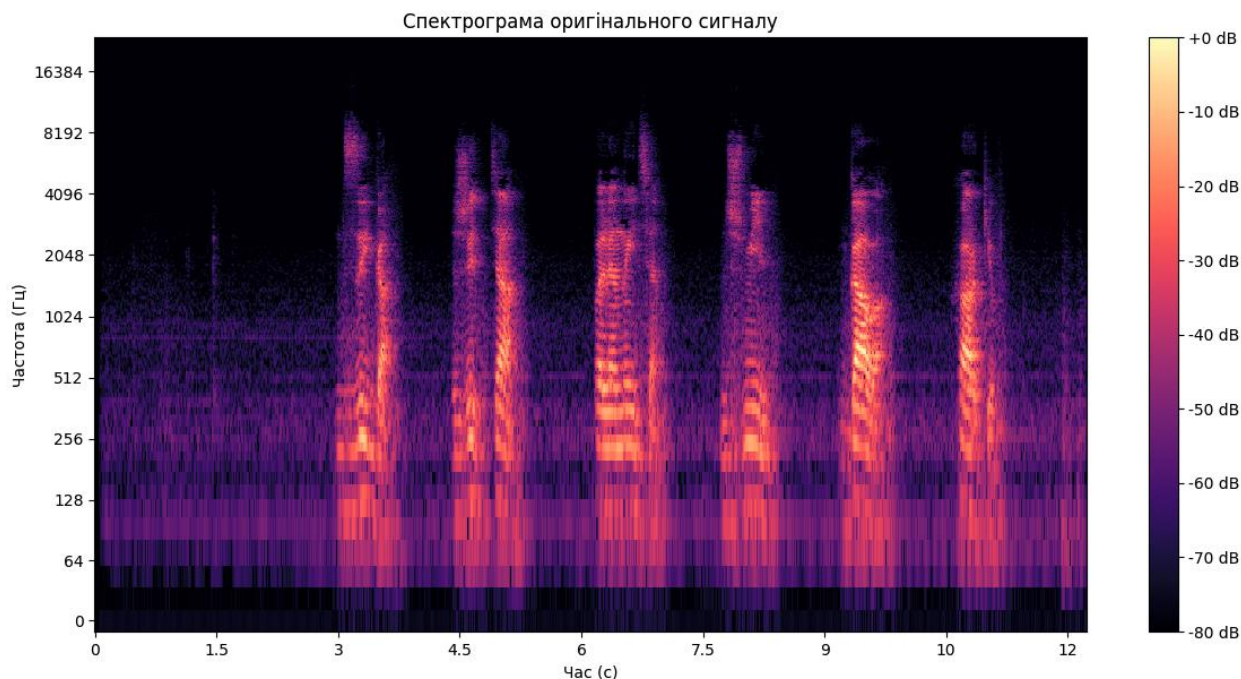


Рисунок 5.10 – Спектрограма оригінального сигналу

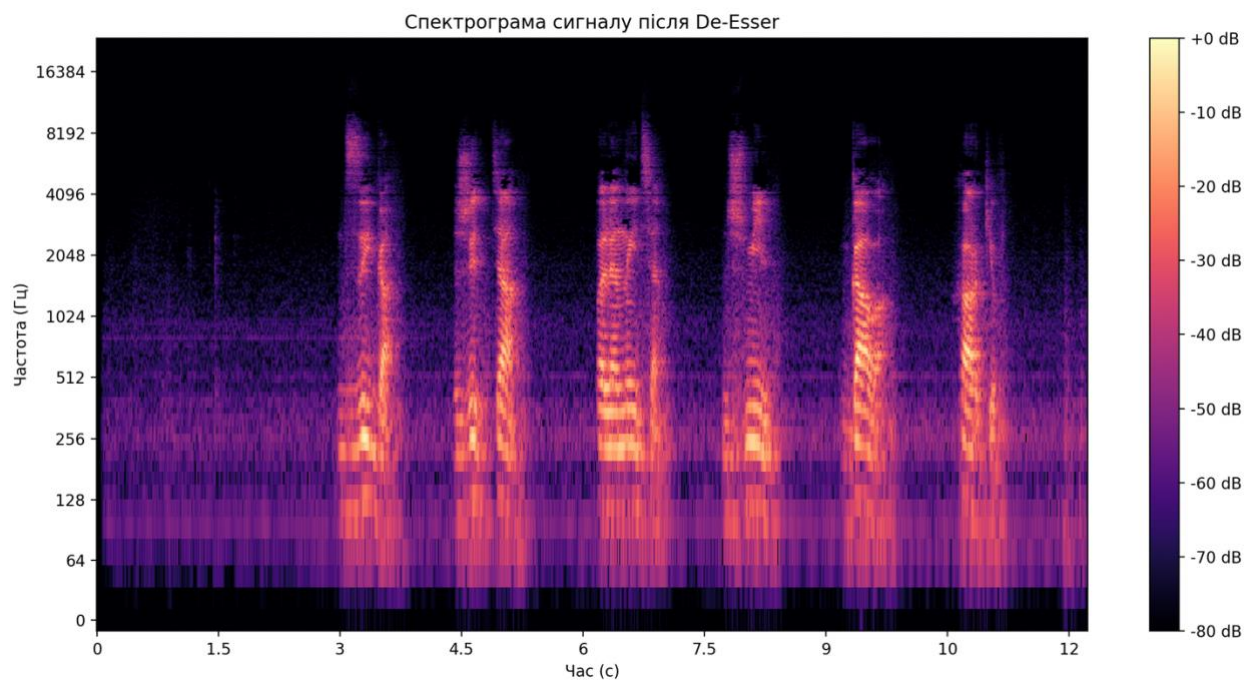


Рисунок 5.11 – Спектрограма сигналу після застосування компресії

Визначені пікові частоти:

Detected sibilant frequency in 4000-5000 Hz: 4057.18 Hz, Magnitude: 136.75

Detected sibilant frequency in 5000-6000 Hz: 5537.55 Hz, Magnitude: 42.15

Detected sibilant frequency in 6000-7000 Hz: 6700.51 Hz, Magnitude: 38.18

Detected sibilant frequency in 7000-8000 Hz: 7041.23 Hz, Magnitude: 21.80

Слухове порівняння оригінального та обробленого сигналу демонструє зниження інтенсивності сибілянтів, що робить звук комфортним для сприйняття. Високочастотні шиплячі звуки стали менш нав'язливими, що дозволяє слухачу краще сприймати інші компоненти мовлення, такі як голосні та приголосні звуки, які несуть основну змістову інформацію. Компресія високих частот ефективно приглушила різкі високочастотні піки, не впливаючи на загальну чіткість та розбірливість мовлення.

5.2.5 Спектральне віднімання

Метод спектрального віднімання (розділ 2.9) є одним з найефективніших способів усунення постійного або стаціонарного шуму з аудіозаписів, що значно покращує якість звуку та підвищує розбірливість мовлення. Процес обробки сигналу включає кілька ключових кроків. Першим критично важливим етапом є визначення зразка шуму, який використовується для моделювання та подальшого видалення шумових компонентів з усього сигналу [103]. У цьому випадку з вихідного сигналу було виділено зразок тривалістю 0.5 секунди, що містить лише шум без корисного мовного сигналу; зазвичай це частина запису, де відсутнє мовлення, і присутній лише фоновий шум. Вибір якісного зразка шуму є ключовим, оскільки неточності на цьому етапі можуть призвести до неефективного видалення шуму або пошкодження корисного сигналу [104].

Далі, для всього сигналу, включаючи обраний зразок шуму, було обчислено короткотермінове перетворення Фур'є (STFT). STFT розбиває сигнал на короткі кадри та виконує перетворення Фур'є для кожного з них, що дозволяє отримати часо-частотне представлення сигналу з інформацією про амплітуду та фазу

частотних компонентів у кожному кадрі. Спектр шуму також обчислюється та зберігається як еталон для подальшого віднімання, відображаючи середні спектральні характеристики шуму, що допомагає ефективно його видалити з сигналу [105].

На наступному етапі відбувається віднімання спектру шуму від спектру кожного кадру сигналу. Ця операція дозволяє зменшити інтенсивність частотних компонентів, які відповідають за шум, зберігаючи при цьому основні характеристики мовного сигналу. Віднімання виконується шляхом зменшення амплітудних значень спектру сигналу на величину спектру шуму. У реалізації було використано функцію `nr.reduce_noise` з параметром `prop_decrease = 0.93`, що вказує на наближене до максимального зниження рівня шуму без значних спотворень корисного сигналу. Цей параметр контролює, наскільки агресивно відбувається зменшення шуму; значення 1.0 означає повне віднімання оціненого спектру шуму.

Після коригування спектру сигналу шляхом віднімання спектру шуму застосовується зворотне короткотермінове перетворення Фур'є (iSTFT) для отримання обробленого сигналу у часовій області. Це дозволяє відновити сигнал з оновленими спектральними компонентами, де шумові компоненти були ефективно знижені. Вибір параметрів, таких як тривалість зразка шуму та значення параметра `prop_decrease`, має важливий вплив на результат. Вибір достатньо довгого зразка шуму забезпечує більш точну оцінку його спектральних характеристик, що підвищує ефективність віднімання, при цьому важливо, щоб зразок містив лише шум без корисного сигналу. Значення `prop_decrease = 0.93` визначає ступінь зменшення шуму; у разі виникнення артефактів це значення можна знизити для більш делікатного віднімання.

Спектрограма після спектрального віднімання (рис. 5.12) демонструє суттєве зниження інтенсивності низькочастотних компонентів, що вказує на ефективне видалення фонового шуму.

Особливо помітно зменшення енергії в паузах між словами або фразами, де раніше домінував фоновий шум. Енергетична структура мовних фрагментів збереглася без суттєвих змін; інтенсивність частот, що відповідають за корисний

сигнал, залишилася на достатньому рівні, що означає, що метод вибірково зменшив лише ті компоненти, які характеризуються як шумові. Високочастотні та середньочастотні області, де присутні основні компоненти мовлення, не зазнали значних змін, що свідчить про збереження якості та чіткості мовлення.

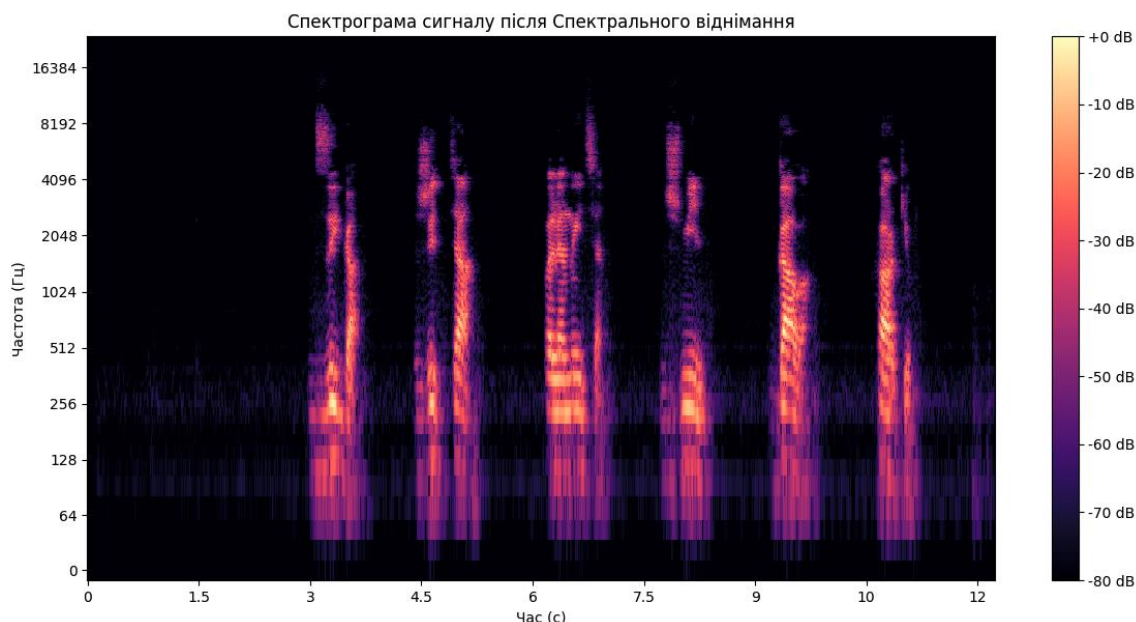


Рисунок 5.12 – Спектрограма сигналу після застосування методу спектрального віднімання

На амплітудно часовій характеристиці (рис. 5.13) прослідковується зниження загально рівня шуму у паузах між голосовими сигналами. Амплітуди основних піків, які відповідають голосовим звукам, залишаються незмінними, що свідчить про збереження мовного контенту та підтверджує високу селективність алгоритму, спрямовану на видалення шуму без значного впливу на основний сигнал.

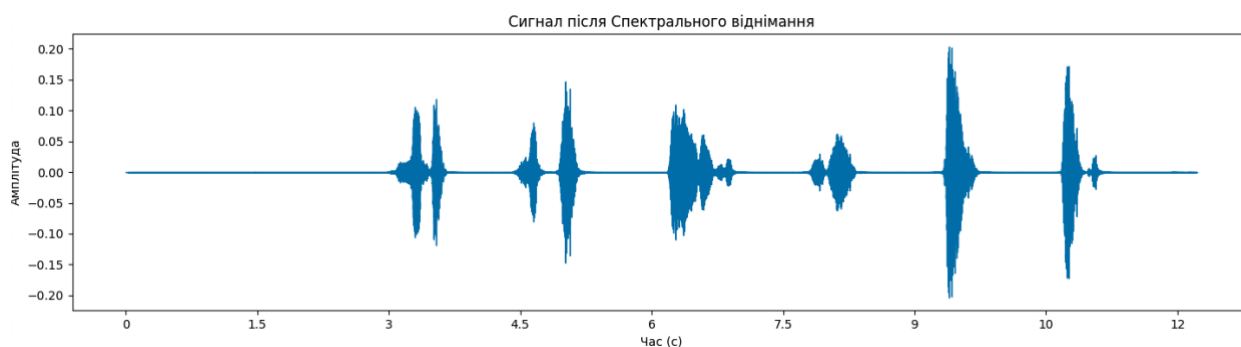


Рисунок 5.13 – Амплітудно часова характеристика сигналу після застосування методу спектрального віднімання

Прослуховування аудіозапису до і після обробки показує значне покращення сприйняття звуку. Після застосування спектрального віднімання ці небажані шуми та артефакти значно зменшилися або зникли повністю, що дозволило мовленню звучати більш чисто та розбірливо. Важливо зазначити, що метод спектрального віднімання може спричиняти виникнення артефактів, таких як "музичні" тони або спотворення, якщо параметри обрано некоректно або якщо шумові компоненти перекриваються з корисним сигналом. У даному випадку налаштування параметра `prop_decrease = 1.0` та правильний вибір зразка шуму дозволили зменшити рівень шуму без значних побічних ефектів, зберігаючи природне звучання голосу; прослуховування обробленого сигналу підтверджує відсутність помітних артефактів та збереження інтонаційних та тембрових характеристик мовлення.

Застосування методу спектрального віднімання на цьому етапі обробки аудіосигналу має кілька позитивних наслідків. По-перше, покращення співвідношення сигнал/шум сприяє більш ефективній роботі подальших обробок та покращує загальну якість звуку. По-друге, зниження фонового шуму робить мовлення більш чітким та зрозумілим, що особливо важливо для записів, призначених для навчання, трансляцій або автоматичного розпізнавання мови. По-третє, чистий звук без фонового шуму зменшує навантаження на слухача, покращуючи комфорт при прослуховуванні, особливо протягом тривалого часу.

5.2.6 Нормалізація сигналу

На оброблений сигнал необхідно застосувати нормалізацію гучності, що дозволить привести звук до міжнародного стандарту та забезпечити стабільний рівень гучності під час прослуховування. Для цього використовуються алгоритми, які відповідають стандарту ITU-R BS.1770. Цей стандарт визначає методику вимірювання та нормалізації рівня гучності, що є важливим для узгодження

звукових матеріалів у різних середовищах, таких як радіо- та телевізійні трансляції або створення професійних аудіозаписів.

Метою цієї обробки було нормалізувати гучність обробленого аудіосигналу до заданого цільового рівня, щоб зробити його відповідним до міжнародних стандартів. Це важливо для уникнення значних перепадів гучності при відтворенні різних звукових матеріалів, що може спричиняти дискомфорт для слухача. У професійній обробці аудіо загальноприйнятим стандартом є рівень гучності -23 LUFS (Loudness Units relative to Full Scale), що забезпечує оптимальний баланс між чіткістю звучання та комфортом сприйняття [106].

Процес нормалізації включає кілька ключових етапів. Спочатку, за допомогою функції `pyln.Meter()`, було створено вимірювач гучності, який відповідає вимогам стандарту ITU-R BS.1770. Цей вимірювач дозволяє точно оцінити інтегровану гучність сигналу, враховуючи особливості людського сприйняття звуку. Після створення вимірювача було проведено вимірювання гучності сигналу шляхом застосування функції:

```
meter.integrated_loudness(final_signal).
```

Це дозволило визначити середню гучність всього сигналу та оцінити, наскільки поточний рівень відрізняється від цільового.

Далі було встановлено цільовий рівень гучності на значенні -23 LUFS, що відповідає міжнародному стандарту для мовлення та професійного аудіовиробництва. Вибір цього значення обумовлений необхідністю забезпечення стабільного та комфортного звучання аудіоматеріалу, який не спричинятиме втоми слухача та буде узгоджений з іншими матеріалами за рівнем гучності.

Для приведення сигналу до цільового рівня було використано функцію `pyln.normalize.loudness()`, яка коригує амплітуду сигналу таким чином, щоб його інтегрована гучність відповідала -23 LUFS. Цей процес нормалізації гучності дозволяє збалансувати рівень гучності всього запису, уникаючи різких перепадів та підтримуючи стабільне звучання. Корекція відбувається шляхом множення сигналу на коефіцієнт, розрахований на основі різниці між поточним та цільовим рівнями гучності.

Сигнали з різними рівнями гучності можуть спричиняти дискомфорт для слухача, особливо при відтворенні декількох записів поспіль або в різних медіа. Нормалізація дозволяє зменшити або повністю усунути ці відмінності, що сприяє комфортному та приємному прослуховуванню.

Виконання нормалізації відповідно до стандарту ITU-R BS.1770 забезпечує єдиний рівень гучності для професійного використання аудіо в медіа, що важливо для телебачення, радіо та онлайн-платформ. Це сприяє узгодженості аудіоматеріалів та професійному рівню виробництва.

Нормалізація гучності допомагає уникнути перевантаження сигналу (кліпінгу), яке може виникати при надмірній амплітуді звуку та призводить до спотворень і втрати якості.

5.3 Порядок обробки та фінальні результати

На основі проведеного аналізу спектрограми обробленого аудіосигналу встановлено ефективність послідовного застосування обробок: шумовий поріг, багатосмугова компресія, компресія високих частот, спектрального віднімання та нормалізації гучності (додаток А).

Первинне використання шумового порогу з адаптивним порогом гейтування +15 дБ та ступенем послаблення -15 дБ, часом атаки 5 мс та часом релізу 50 мс дозволило знизити рівень фонового шуму у паузах між словами та фразами. Це сприяло підвищенню розбірливості та чіткості мовлення, зберігаючи при цьому природність звучання. Спектральний аналіз після цього етапу показав суттєве зниження інтенсивності в низькочастотному діапазоні, особливо в ділянках, де раніше домінував фоновий шум.

На другому етапі було застосовано багатосмугову компресію, яка дозволила вирівняти динамічний діапазон сигналу, зробивши звучання більш рівномірним і комфортним для сприйняття. Компресор працював з трьома частотними смугами: низькою (20–250 Гц), середньою (250–4000 Гц) та високою (4000–20000 Гц). Поріг компресії було встановлено на рівні -20 дБ, з коефіцієнтом компресії 4:1, часом

атаки 20 мс та релізу 300 мс. Також використовувалася плавна компресія (Soft Knee) з шириною переходу 10 дБ, що дозволило уникнути різких змін у динаміці. Це налаштування забезпечило зниження пікових амплітуд середньочастотного діапазону (500–2000 Гц) на 15–20%, підтверджене спектральним аналізом, що зробило звук більш збалансованим.

Використання високочастотної компресії з порогом виявлення піків 0,1, коефіцієнтом зменшення інтенсивності 0,7 та шириною смуги фільтра 100 Гц ефективно знизило інтенсивність сибілянтів у високочастотному діапазоні 4000–8000 Гц. Це покращило сприйняття звуку, зменшивши навантаження на слухача, та зберегло чіткість мовлення. Спектральний аналіз після цього етапу продемонстрував суттєве зниження інтенсивності компонентів, відповідальних за сибілянти.

Метод спектрального віднімання, застосований з використанням зразка шуму тривалістю 0,5 секунди та параметром `prop_decrease` = 0,93, додатково зменшив фоновий шум, особливо в низькочастотній області. Це сприяло подальшому очищенню сигналу без спотворення корисних частотних компонентів. Спектрограма після цього етапу показала подальше зниження шуму в паузах між словами, що підтверджує ефективність методу.

Додаткове гейтування шуму з порогом -10 дБ та ступенем послаблення -35 дБ усунуло залишкові шумові компоненти, забезпечивши максимальну чистоту сигналу. Завершальна нормалізація гучності до цільового рівня -23 LUFS відповідно до стандарту ITU-R BS.1770 гарантувала стабільний та комфортний рівень звучання, сумісний з професійними вимогами та забезпечила узгодженість з іншими аудіоматеріалами.

Фінальна обробка аудіосигналу продемонструвала суттєві покращення у якості звуку, які підтверджуються як спектральним аналізом (рис. 5.14), так і кількісними показниками, зокрема співвідношенням сигнал/шум (SNR). Початкове значення SNR для оригінального сигналу становило 15.66 дБ, що свідчило про наявність значного шумового забруднення, особливо у низькоамплітудних

інтервалах між мовними фрагментами. Після комплексної обробки значення SNR збільшилось до 23.17 дБ, що відображає ефективність усіх застосованих методів.

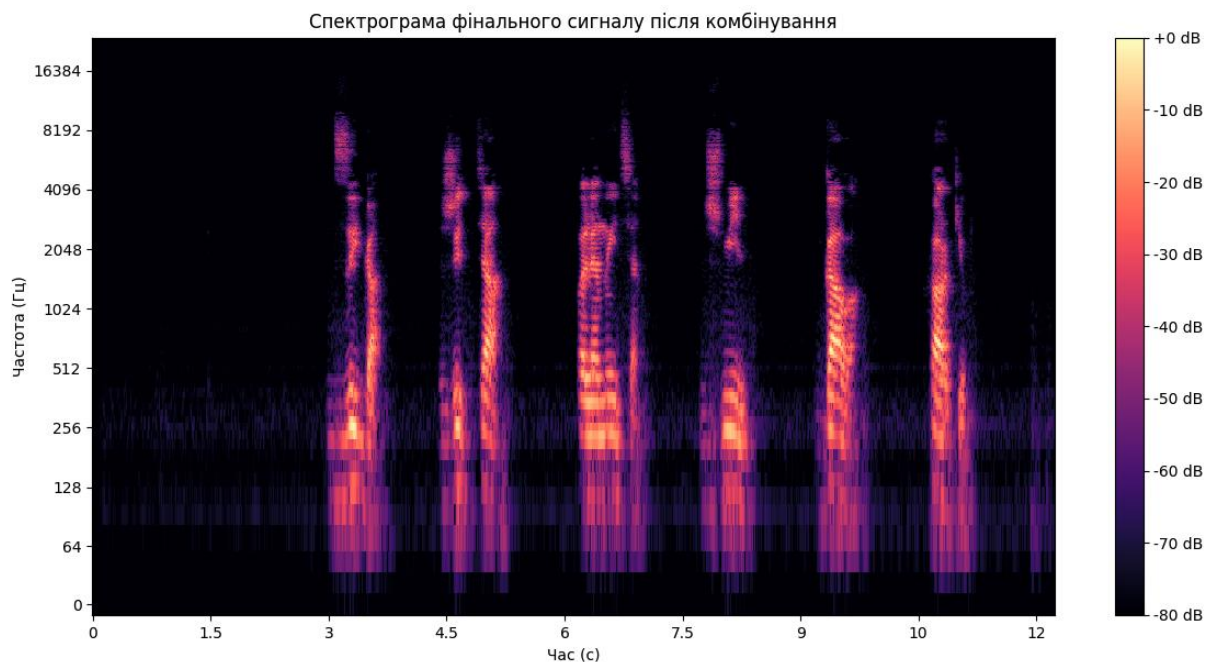


Рисунок 5.14 – Спектрограма обробленого сигналу

На спектрограмі фінального сигналу після комбінування чітко видно суттєве зменшення рівня шуму у всіх частотних діапазонах, особливо в паузах між активними мовними сигналами. Найбільш помітно це у низькочастотному діапазоні (до 500 Гц) та високочастотному діапазоні (4000–8000 Гц), де попередні артефакти та сибілянти були практично усунуті. Зменшення інтенсивності шуму в паузах між словами свідчить про високу ефективність методів спектрального віднімання та шумового порогу.

Часово-амплітудна характеристика (рис. 5.15) фінального сигналу демонструє рівномірність динамічного діапазону після багатосмугової компресії. Амплітудні піки були вирівняні, що зменшило різницю між гучними та тихими сегментами, зберігаючи при цьому природність звучання. Це створило комфортне для сприйняття звучання з оптимальним балансом між інтенсивністю мовлення та шумовими складовими.

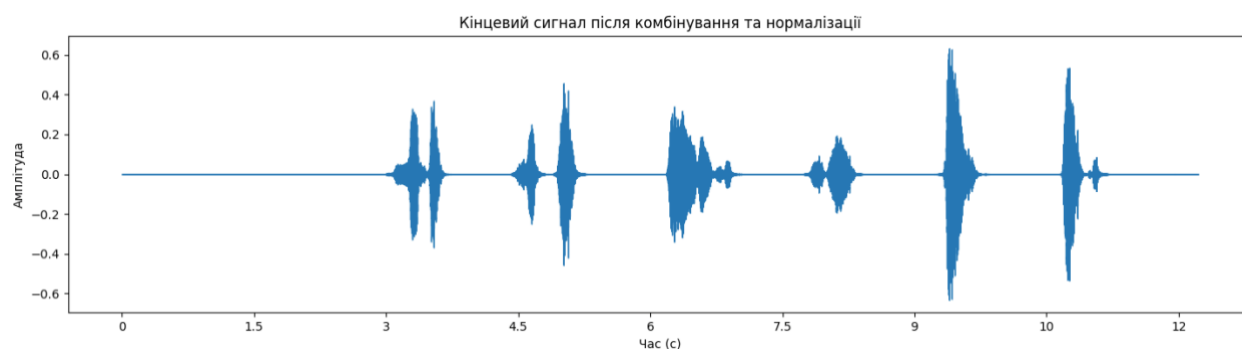


Рисунок 5.15 – Часово амплітудна характеристика обробленого сигналу

Додаткове гейтування шуму зі ступенем послаблення -15 дБ ефективно усунуло залишкові фонові артефакти, зберігаючи цілісність мовного сигналу. Нормалізація гучності до рівня -23 LUFS забезпечила узгодженість з професійними стандартами аудіообробки (ITU-R BS.1770), що важливо для використання сигналу в широкому спектрі медіазастосунків.

Таким чином, фінальний сигнал характеризується високою чіткістю мовлення, значним зменшенням фонових шумів, а також збалансованим частотним і динамічним спектром. Поліпшення якості звуку підтверджується аналізом спектральних характеристик, а також значним зростанням показника SNR на 7.51 дБ у порівнянні з початковим сигналом. Це підкреслює ефективність застосованої послідовності методів обробки для усунення шумового забруднення.

В таблиці 5.1 можна побачити зміну основних характеристик сигналу після застосування кожної обробки окремо. Визначені параметри для аналізу: середня амплітуда, відношення сигнал/шум, максимальне значення амплітуди, максимальне значення амплітуди до RMS, RMS сигналу.

Таблиця 5.1 – Кількісні параметри практичної частини дослідження

Обробка	Середня амплітуда	Відношення сигнал/шум (дБ)	Максимальне значення амплітуди (лін. / дБ)	Максимальне значення амплітуди до RMS (лін. / дБ)	RMS (лін. / дБ)
Оригінальне аудіо	0.009462	15.66	0.032035 / -29.89	2.645999 / 8.45	0.011669 / -38.66

Попередня обробка	0.017860	17.68	0.061571 / -24.21	2.740802 / 8.76	0.022140 / -33.10
Поріг шуму	0.007646	18.17	0.026490 / -31.54	2.731380 / 8.73	0.009482 / -40.46
Багатосмугова компресія	0.036496	18.28	0.123046 / -18.20	2.689670 / 8.59	0.045002 / -26.94
Компресія сибілянтів	0.008751	18.24	0.029441 / -30.62	2.685328 / 8.58	0.010788 / -39.34
Спектральне віднімання	0.004294	23.16	0.014925 / -36.52	2.789580 / 8.91	0.005318 / -45.48
Нормалізація	0.013373	23.16	0.046483 / -26.65	2.789580 / 8.91	0.016563 / -35.62
Сигнал після кодування методом менш значущого біта	0.013344	23.17	0.046455 / -26.66	2.821777 / 9.01	0.016539 / -35.63

Оригінальний аудіо сигнал до попередньої обробки має невелику середню амплітуду (0.009462) і відношення сигнал/шум близько 15.66 дБ, що підтверджує наявність шуму в сигналі. Усереднений піковий рівень складає -29.89 дБ, а відношення максимальної амплітуди до RMS сигналу рівне 8.45 дБ, отже максимальний рівень у фреймах удвічі-втричі перевищує RMS (2.645999 у лінійному масштабі). Значення RMS складає -38.66 дБ, що вказує на низьку гучність фрагменту.

Попередня обробка (нормалізація, видалення постійної складової, фільтрація високих та низьких частот) підвищує середню амплітуду майже в два рази (0.017860) завдяки нормалізації сигналу. Відношення сигнал/шум зростає до 17.68 дБ, адже фільтри відсікають частину небажаних частот і шуму, а нормалізація піднімає рівень корисного сигналу. Усереднений піковий рівень стає -24.21 дБ (більший у лінійному масштабі), RMS теж зростає до -33.10 дБ. Максимальна амплітуда до рівня RMS становить 8.76 дБ, що відображає незначну зміну в порівнянні з оригінальним сигналом, тобто співвідношення піку до RMS лишається приблизно тим самим, просто вищим за рівнем.

Після застосування шумового порогу середня амплітуда зменшується до 0.007646, оскільки ділянки нижче визначеного порогу видаляються. Водночас середнє значення сигнал/шум зростає до 18.17 дБ, так як присутність шуму в сигналі стає ще нижче та частково прибирає шум в паузах між словами. Усереднений піковий рівень становить -31.54 дБ, тобто сигнал став тихіши, за рахунок видалення шуму. Відношення максимальної амплітуди до рівня RMS лишилось майже без змін (8.73 дБ), а RMS упав до -40.46 дБ — через приглушення тихих фрагментів.

Застосування багатосмугової компресії забезпечило підвищення рівня середньої амплітуди сигналу до 0.036496, завдяки обробці динамічного діапазону. Відношення сигнал/шум зростає до 18.28 дБ, адже компресор водночас оброблює як корисний сигнал, так і залишковий шум. Усереднений піковий рівень становить -18.20 дБ (0.123046 лін.) і RMS -26.94 дБ демонструють підвищення рівня гучності у сегментах. Рівень максимальної амплітуди до RMS (8.59 дБ) дещо знизився порівняно з попередніми кроками, оскільки компресор зменшує різницю між максимальною і середньою амплітудою.

Після застосування компресії сибілянтів середня амплітуда знову знижується до 0.008751, оскільки фокус сформовано на подавлення високочастотного діапазону сигналу. Відношення сигнал/шум майже не змінюється - 18.24 дБ. Усереднений піковий рівень -30.62 дБ, відношення максимальної амплітуди до рівня RMS 8.58 дБ та RMS -39.34 дБ демонструють, що загальний енергетичний рівень знизився.

Спектральне віднімання демонструє зниження середньої амплітуди до рівня 0.004294, але відношення сигнал/шум різко зростає до 23.16 дБ. Це означає, що рівень шуму значно знизився, а спектральне віднімання забезпечило якісне видалення шумової складової сигналу. Усереднений піковий рівень складає -36.52 дБ, відношення максимальної амплітуди до рівня RMS трохи збільшується до 8.91 дБ, а RMS падає до -45.48 дБ. Це наслідок того, що шум і тихі складові сигналу сильно знижено, тому середній рівень падає, а піки лишаються на відносно високому рівні.

Нормалізація гучності забезпечує рівень середньої амплітуди 0.013373, при тому не змінюючи відношення сигнал/шум (23.16 дБ). Усереднений піковий рівень тепер -26.65 дБ, відношення максимальної амплітуди до рівня RMS не змінюється (8.91 дБ), а RMS піднімається до -35.62 дБ. Таким чином, відбувається масштабування сигналу до заданого рівня гучності без зміни пропорцій.

Після застосування стеганографії методом менш значущого біта, середня амплітуда майже не зазнала змін (0.013344), так само відношення сигнал/шум (23.17 дБ), усереднений піковий рівень (-26.66 дБ), відношення максимальної амплітуди до рівня RMS (9.01 дБ) та рівень RMS (-35.63 дБ). Це підтверджує, що застосування методу практично не вплинуло на характеристики сигналу.

5.4 Аналогова реалізація обробки сигналу

На рисунку 5.16 позначено основні етапи обробки звукового сигналу. Аналогово-цифрове перетворення (АЦП) реалізується за допомогою спеціалізованої мікросхеми, яка містить схему вибірки і зберігання, квантувальний блок та цифровий інтерфейс. Схема вибірки і зберігання базується на використанні конденсатора та ключового елемента, що утримує значення аналогового сигналу протягом певного часу. Квантування здійснюється через порівняння вхідного сигналу із референсною шкалою напруг, яка генерується за допомогою резисторів та джерела напруги, а отримані результати кодуються у двійковий формат для подальшої цифрової обробки.

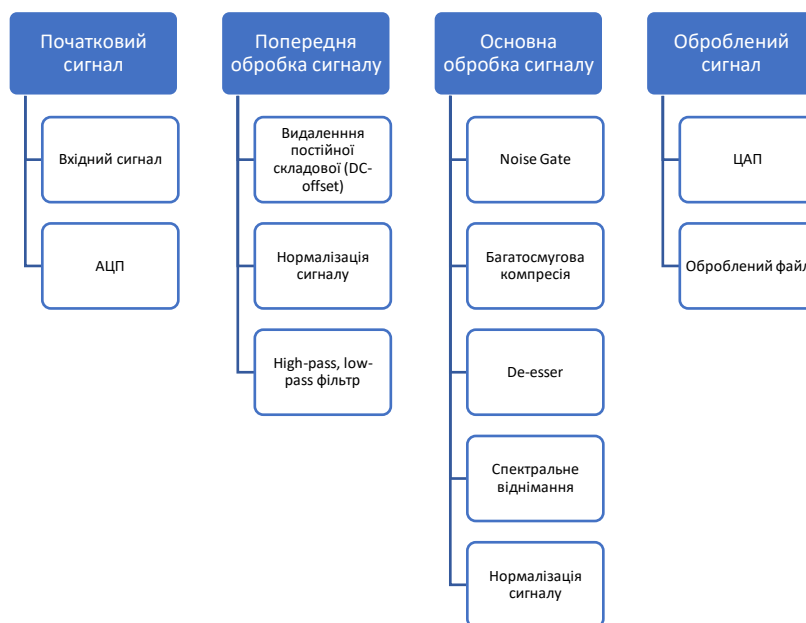


Рисунок 5.16 – Схема реалізації обробки сигналу

Видалення постійної складової (DC-offset) виконується за допомогою високочастотного фільтра, що складається з конденсатора та резистора. Конденсатор блокує постійну складову сигналу, дозволяючи проходити тільки змінному сигналу, тоді як резистор задає частоту зрізу фільтра. Нормалізація сигналу здійснюється за допомогою підсилювального каскаду, побудованого на операційному підсилювачі з підстроювальними резисторами. Це дозволяє змінювати коефіцієнт посилення, вирівнюючи амплітуду сигналу до заданого рівня. У цифрових системах для цього можуть використовуватись цифрові потенціометри, керовані мікроконтролером.

Високочастотні та низькочастотні фільтри реалізуються як активні або пасивні електронні ланцюги. Активні фільтри базуються на операційних підсилювачах у комбінації з резисторами та конденсаторами, що дозволяє забезпечувати високу точність і стабільність роботи. Пасивні фільтри, які складаються з резисторів, котушок індуктивності та конденсаторів, використовуються для простіших задач фільтрації.

“Шумові ворота” (Noise Gate) працюють на основі порівняльної схеми, де компаратор визначає рівень сигналу і порівнює його із заданим порогом. Якщо амплітуда сигналу менша за поріг, транзисторний ключ або реле блокує сигнал,

таким чином видаляючи небажані шуми на фоні. У цифрових системах цей процес виконується програмно через логіку керування сигналом. Багатосмугова компресія реалізується через поділ сигналу на кілька частотних діапазонів за допомогою смугових фільтрів. Кожна смуга обробляється компресором, який динамічно знижує амплітуду сигналу, що перевищує заданий поріг. Така схема базується на операційних підсилювачах, діодах та логічних елементах.

Компресор високих частот виконує усунення шиплячих звуків через використання високочастотного фільтра, який виділяє частотний діапазон 4-10 кГц, та компресора, що знижує амплітуду звуків у цьому діапазоні. Для реалізації застосовуються операційні підсилювачі, транзистори або діодні каскади. Спектральне віднімання виконується цифровими методами, де модуль швидкого перетворення Фур'є (FFT) аналізує спектр сигналу, а шумовий профіль, отриманий із тихих ділянок, віднімається зі спектру основного сигналу. У разі аналогової реалізації використовуються смугові фільтри та інверсійні підсилювачі.

Цифро-аналогове перетворення (ЦАП) здійснюється за допомогою R-2R резисторної матриці, яка генерує аналоговий сигнал із цифрового коду. Для усунення високочастотних гармонік, що виникають під час дискретизації, застосовується низькочастотний фільтр, який згладжує вихідний сигнал. Завершальним етапом є формування обробленого файлу, яке здійснюється через цифрові процесори або мікроконтролери, що зберігають оброблений сигнал у пам'яті у вигляді аудіофайлів (наприклад, WAV чи MP3). Уся схема інтегрується в спеціалізовані модулі або пристрої (наприклад, аудіоінтерфейси), де кожен етап обробки налаштовується для досягнення найкращих результатів.

5.4.1 Переваги цифрової обробки

Використання віртуальних пристроїв у системах цифрової обробки аудіосигналів має кілька суттєвих переваг у порівнянні з традиційними фізичними мікшерами або апаратними DSP-процесорами. По-перше, віртуальні пристрої надають високу гнучкість та адаптивність системи, що дозволяє користувачам

легко конфігурувати та змінювати параметри обробки сигналів без необхідності фізичного переналаштування обладнання. Що дає можливість за швидко змінювати налаштування та параметри до аудіообробки, такі як: адаптація до зміни патерну шумового забруднення, голосу спікера, тембру, акустичних особливостей приміщення, зміна коефіцієнту фільтрації що пришвидшить обробку сигналу та забезпечить оптимальне використання наявних ресурсів.

Віртуальні пристрої є економічно ефективніші, оскільки суттєво знижують витрати на придбання та обслуговування фізичних компонентів мікшера або DSP-процесорів, потребують менше електроенергії та не займають додатковий простір. Віртуальні рішення реалізуються на мікроконтролерах, комп'ютерах або серверах, що робить їх доступними для застосування та реалізації проектів з низьким бюджетом та простором. Замість того, щоб інвестувати значні кошти в дорогі апаратні пристрої, користувачі можуть використовувати програмне забезпечення, яке часто має менші витрати на ліцензії та оновлення

Інтеграція та автоматизація забезпечує можливість інтеграції з іншими цифровими системами та програмним забезпеченням, що дозволяє автоматизувати процеси обробки аудіо та підвищити загальну ефективність роботи. Наприклад, програмні інтеграції через API дозволяють створювати складні ланцюжки обробки сигналів, що можуть бути керовані автоматично або за допомогою скриптів та адаптуватись до зміни аудіо фрагменту. Сучасні алгоритми цифрової обробки сигналів, реалізовані у віртуальних пристроях, забезпечують високу точність та якість обробки, часто перевищуючи можливості аналогових пристроїв.

Віртуальні системи також спрощують процес оновлення та розширення функціональності. Нові функції або покращення можуть бути впроваджені шляхом оновлення програмного забезпечення, без необхідності заміни апаратної складової. Це забезпечує гнучкість та адаптивність системи до нових технологічних вимог та стандартів.

Висновки до розділу 5.

1. У процесі попередньої обробки сигналу було здійснено низку важливих етапів, спрямованих на покращення якості звукового сигналу та підготовку його до подальшої обробки та аналізу. Перш за все, проведена нормалізація амплітуди дозволила стандартизувати рівень гучності сигналу, що забезпечило збереження природної частотної структури та підвищення розбірливості тихих частин мовного матеріалу. Вибір коефіцієнта нормалізації 0.7 забезпечив необхідний запас гучності для подальшої обробки, запобігаючи перенасиченню та зберігаючи динамічний діапазон запису.

2. Видалено постійну складову сигналу для запобігання спотворень у подальших етапах обробки, таких як динамічні процесори та фільтри. Завдяки вирівнюванню середнього значення амплітуди до нуля було забезпечено коректність математичних перетворень та точність спектрального аналізу, що є необхідним для адекватної оцінки енергетичного вмісту сигналу на різних частотах.

3. Визначення фундаментальної частоти мовного сигналу дозволило класифікувати гендер мовця, що в свою чергу сприяло оптимізації параметрів фільтрації. Адаптивне налаштування частот зрізу високочастотного та низькочастотного фільтрів відповідно до гендеру мовця забезпечило ефективне видалення небажаних шумів без суттєвого впливу на важливі акустичні характеристики голосу. Використання Баттервортових фільтрів першого порядку з плавними схилами гарантувало збереження природного звучання мовлення та уникнення артефактів.

4. У процесі зниження шумового забруднення та поліпшення розбірливості мови було успішно застосовано комплексну послідовність п'яти методів обробки аудіосигналу, що забезпечило значне покращення його якості. Початкове застосування шумового порогу дозволило ефективно усунути низькорівневі фонові шуми в паузах між словами та фразами, підвищуючи чистоту та чіткість звуку без втрати важливих мовних компонентів. Наступний етап

багатосмугової компресії вирівняв динамічний діапазон сигналу, забезпечивши рівномірність гучності та збереження природної динаміки звучання. Використання компресії високих частот значно зменшило інтенсивність сибілянтів у високочастотному діапазоні, що покращило сприйняття мовлення та знизило навантаження на слухача. Метод спектрального віднімання дозволив вибірково зменшити рівень шумових компонентів, зберігаючи при цьому основні мовні характеристики сигналу, що підтверджується збільшенням співвідношення сигнал/шум (SNR) з 15.66 дБ до 23.17 дБ. Завершальна нормалізація гучності відповідно до стандарту ITU-R BS.1770 забезпечила стабільний та комфортний рівень звучання, що відповідає професійним вимогам та забезпечує узгодженість аудіоматеріалів у різних середовищах. Загалом, застосована послідовність методів обробки ефективно знизила рівень шумового забруднення, підвищила розбірливість мовлення та забезпечила високу якість звукового сигналу, що є критично важливим для подальшого аналізу та використання аудіоданих у різноманітних застосунках.

6 РОЗПІЗНАВАННЯ МОВИ ТА КОДУВАННЯ МОВНОГО СИГНАЛУ

6.1 Розпізнавання мовлення з аудіофайлу

Першим кроком у кодуванні та передачі супутньої інформації в аудіофайл буде розпізнавання мовлення з аудіофайлу. Для цього використовується бібліотека Vosk, яка є інструментом для офлайн-розпізнавання мовлення з підтримкою української мови. Vosk дозволяє здійснювати розпізнавання мовлення без необхідності підключення до інтернету, що забезпечує швидкість і незалежність від мережових з'єднань. Також програма містить файл-словник, що дає змогу оновлювати та виправляти слова в разі необхідності. Також це надає змогу опрацювати інформацію офлайн, що забезпечує конфіденційність даних та швидкість обробки. Крім того, Vosk підтримує адаптацію моделей, що дозволяє підвищити точність розпізнавання для специфічних голосів або умов запису.

Принцип роботи Vosk базується на використанні акустичних та мовних моделей, які були попередньо навчені на великому обсязі даних. Акустична модель відповідає за перетворення звукових сигналів у фонетичні ознаки, тоді як мовна модель прогнозує ймовірні послідовності слів на основі цих ознак. Разом ці моделі дозволяють точно розпізнавати мовлення і трансформувати його у текст.

Бібліотека використовує сучасні нейронні мережі для аналізу звуку. Вона спочатку виконує перетворення звукового сигналу в спектрограму за допомогою перетворення Фур'є, що дозволяє отримати частотні характеристики сигналу. Далі ці дані передаються в акустичну модель, яка була навчена розпізнавати фонemi — найменші звукові одиниці мови. На основі послідовності фонем мовна модель прогнозує найбільш ймовірні слова і фрази, використовуючи ймовірнісні методи та статистичні дані про мову [107].

Всі маніпуляції відбуваються у форматі WAV, оскільки він забезпечує високу якість звуку і є сумісним з багатьма аудіоінструментами. Перевіряється, щоб файл був монофонічним та мав 16-бітне РСМ-кодування. Ці параметри є критичними для

коректної роботи розпізнавача, оскільки інші формати або параметри можуть призвести до помилок або неточностей у розпізнаванні.

Після цього створюється об'єкт моделі Vosk, який завантажує акустичну та мовну модель з заданого шляху. Створюється також розпізнавач, який використовує цю модель і налаштований на частоту дискретизації аудіофайлу.

Аудіодані зчитуються з файлу частинами, що дозволяє обробляти навіть великі файли без перевантаження пам'яті. Кожна порція аудіоданих передається розпізнавачу, який аналізує її та намагається розпізнати мовлення. Розпізнавання відбувається в реальному часі, і розпізнавач може повертати проміжні результати, що особливо корисно для довгих записів.

Отримані результати розпізнавання накопичуються у текстову змінну. Після того, як всі аудіодані були оброблені, отримується фінальний результат розпізнавання, який містить повний текстовий транскрипт аудіофайлу. Цей текст є точним відображенням мовлення, яке містилося в аудіофайлі, і може містити як слова, так і фрази, залежно від вмісту.

6.2 Кодування розпізнаного тексту в бінарний формат

Вилучивши текст з аудіофайлу за допомогою бібліотеки Vosk, наступним важливим етапом є кодування цього тексту в бінарний формат. Це необхідно для того, щоб підготувати повідомлення до стеганографічного вбудовування в аудіосигнал за допомогою методу найменш значущого біта (LSB). Процес кодування тексту в бінарний формат полягає в послідовному перетворенні кожного символу тексту в його двійкове представлення згідно з кодуванням UTF-8.

Кодування UTF-8 є стандартом для представлення текстових даних у комп'ютерних системах і підтримує широкий спектр символів, включаючи український алфавіт. Як зазначено в розділі 3.3, це кодування використовує змінну кількість байтів (від одного до чотирьох) для представлення кожного символу. Символи латинського алфавіту та деякі спеціальні символи зазвичай кодуються

одним байтом, тоді як символи кирилиці, зокрема українські літери, часто займають два байти [108].

Процес кодування розпочинається з ітерації по кожному символу розпізнаного тексту. Для кожного символу виконується наступне:

1. Перетворення символу в кодову точку Unicode: Кожен символ має унікальний числовий код у стандарті Unicode. Наприклад, літера "А" має кодову точку U+0410.

2. Кодування в UTF-8: Кодова точка символу перетворюється в одну або кілька байтів згідно з правилами кодування UTF-8. Для українських літер це зазвичай два байти. Наприклад, літера "П" перетворюється в байти з шістнадцятковими значеннями 0xD0 та 0x9F.

3. Перетворення байтів у бінарну форму: Кожен байт переводиться у двійкове представлення. Байт складається з 8 бітів, тому кожне шістнадцяткове значення перетворюється в послідовність з восьми бітів. Для байта 0xD0 двійкове представлення буде "11010000", а для 0x9F — "10011111".

4. Об'єднання бітових послідовностей: Бітові послідовності байтів, отримані з одного символу, об'єднуються в одну послідовність, що представляє цей символ у бінарній формі. Продовжуючи попередній приклад, для літери "П" отримаємо бінарну послідовність "1101000010011111".

5. Накопичення бінарних даних: Цей процес повторюється для всіх символів у тексті. Бінарні послідовності кожного символу послідовно додаються одна до одної, формуючи довгу бінарну стрічку, яка представляє весь текст [109].

Особливу увагу слід приділити символам, які не є літерами, наприклад, пробілам, розділовим знакам або спеціальним символам. Вони також кодуються згідно з UTF-8 і їх необхідно включити в бінарну послідовність, щоб зберегти цілісність та правильність відтворення повідомлення при декодуванні.

Наприклад, пробіл кодується в UTF-8 як байт 0x20, що в бінарній формі буде "00100000". Розділовий знак, такий як кома ",", має шістнадцяткове значення 0x2C і бінарне представлення "00101100".

Важливо забезпечити точність і послідовність при перетворенні, оскільки будь-яка помилка в бітовому представленні може призвести до неправильного декодування символів під час зворотного процесу. Також слід враховувати порядок бітів і байтів, оскільки зміна порядку може змінити значення символів [110].

Після того, як весь текст перетворено в бінарну послідовність, отримуємо суцільний ланцюжок бітів, готовий до вбудовування в аудіосигнал. Кожен біт цього ланцюжка буде відповідати найменш значущому біту одного семплу аудіоданих.

Необхідно також оцінити загальну кількість бітів у отриманій бінарній послідовності, щоб переконатися, що аудіофайл має достатню кількість семплів для вбудовування всього повідомлення. Якщо кількість бітів у повідомленні перевищує кількість доступних семплів, процес вбудовування буде неможливим без втрати частини повідомлення.

6.3 Вбудовування бінарного повідомлення в аудіосигнал методом LSB

Наступний етап буде спрямований на вбудовування бінарного повідомлення в оброблений аудіосигнал за допомогою методу найменш значущого біта (LSB). Цей метод дозволяє приховати інформацію в аудіофайлі, змінюючи лише найменш значущі біти семплів, що мінімально впливає на якість звуку та є практично непомітним для людського вуха (розділ 3.4).

Процес вбудовування бінарного повідомлення в аудіосигнал методом LSB розпочинається з підготовки аудіоданих. Оригінальний аудіосигнал, зчитаний з файлу, представлений у вигляді масиву чисел з плаваючою точкою в діапазоні від -1.0 до 1.0, що відповідає амплітудам семплів. Для можливості маніпулювання окремими бітами цих значень необхідно конвертувати їх у цілочисельний формат з фіксованою розрядністю.

Зазвичай використовується 16-бітний цілочисельний формат, де кожен семпл аудіосигналу представлений цілим числом у діапазоні від -32768 до 32767. Це досягається шляхом множення значень семплів на 32767 і перетворення їх у цілі

числа. Такий підхід дозволяє безпосередньо змінювати окремі біти кожного семплу [111].

Після конвертації аудіоданих у цілочисельний формат необхідно переконатися, що довжина бінарного повідомлення не перевищує кількості доступних семплів у аудіофайлі. Кожен біт повідомлення буде вбудований в один семпл, тому кількість семплів повинна бути не меншою за кількість бітів у бінарному повідомленні. Якщо повідомлення занадто велике, слід обрати довший аудіофайл або скоротити повідомлення.

Вбудовування повідомлення здійснюється шляхом послідовного проходження по кожному біту бінарного повідомлення та заміни найменш значущого біта відповідного семплу аудіоданих на цей біт. Це реалізується через побітові операції над семплами. Для кожного семплу виконується наступне:

1. Очищення найменш значущого біта семплу: За допомогою побітової операції "AND" з маскою, де всі біти рівні 1, окрім найменш значущого, який рівний 0, встановлюється найменш значущий біт семплу в 0. Наприклад, маска може мати вигляд 11111110 у двійковій системі для 8-бітного числа або 1111111111111110 для 16-бітного.

2. Встановлення найменш значущого біта згідно з бітом повідомлення: За допомогою побітової операції "OR" встановлюється найменш значущий біт семплу в значення біта повідомлення (0 або 1). Якщо біт повідомлення рівний 1, то найменш значущий біт семплу встановлюється в 1; якщо 0 — залишається 0.

Таким чином, найменш значущий біт семплу замінюється на біт повідомлення (рис. 6.1), а інші біти семплу залишаються незмінними.

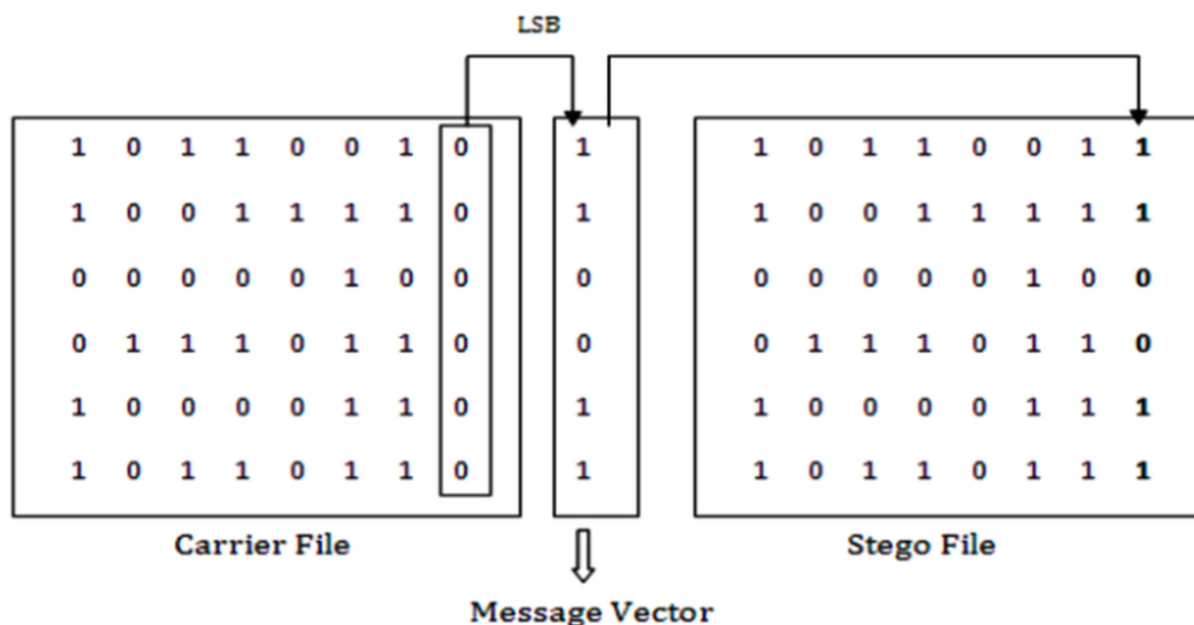


Рисунок 6.1 – Схема кодування методом LSB [112]

Це мінімізує вплив на загальну амплітуду семплу та, відповідно, на якість звуку [113].

Процес повторюється для всіх бітів бінарного повідомлення. У результаті отримаємо модифікований масив семплів, який містить приховане повідомлення в найменш значущих бітах.

Після завершення вбудовування модифікований масив семплів конвертується назад у формат з плаваючою точкою, нормалізуючи значення до діапазону від -1.0 до 1.0. Це необхідно для збереження аудіосигналу у форматі, придатному для відтворення та збереження у файл стандартними засобами.

Завершальним кроком є збереження модифікованого аудіосигналу у новий файл. При цьому важливо зберегти початкові параметри аудіофайлу, такі як частота дискретизації та кількість каналів, щоб забезпечити сумісність і правильне відтворення аудіо [114].

Метод LSB є ефективним, оскільки зміни в найменш значущих бітах семплів практично не впливають на сприйняття звуку людиною. Це пояснюється тим, що слух людини не є чутливим до таких незначних змін амплітуди сигналу. Водночас, дані, вбудовані в найменш значущі біти, можуть бути надійно витягнуті за умови знання алгоритму вбудовування та порядку бітів.

Під час вбудовування слід враховувати можливі перешкоди, які можуть виникнути при подальшій обробці або компресії аудіофайлу. Наприклад, стиснення аудіо з втратами (як у форматі MP3) може змінити або знищити найменш значущі біти, що призведе до втрати прихованого повідомлення. Тому рекомендується використовувати безшкваті або слабо стиснені формати аудіо для зберігання та передачі файлів з вбудованим повідомленням.

У підсумку, вбудовування бінарного повідомлення в аудіосигнал методом LSB є простим і ефективним способом приховування інформації в аудіофайлах. Цей метод забезпечує мінімальний вплив на якість звуку та дозволяє надійно передавати приховані дані за умови правильного використання та дотримання рекомендацій щодо форматів аудіо.

Аналіз спектрограм та статистичних параметрів початкового та модифікованого сигналів після кодування текстового повідомлення методом найменш значущого біта (LSB) демонструє, що цей метод є ефективним для стеганографічного приховування даних в аудіосигналі без суттєвого впливу на його якість. Детальний розгляд змін у параметрах сигналу підтверджує, що вбудовування інформації відбувається практично непомітно для слухача.

Рівень гучності початкового сигналу становив 0.04072302579879761, тоді як модифікований сигнал має рівень гучності 0.04071835055947304. Зміна рівня гучності складає приблизно 6.07×10^{-6} , що є надзвичайно малим відхиленням. RMS-рівень гучності є мірою середньоквадратичної амплітуди сигналу і відображає його енергетичний зміст. Така незначна зміна свідчить про те, що енергетичний вміст сигналу залишився практично незмінним. З точки зору фізіології слуху, людське вухо не здатне виявити такі мінімальні зміни гучності. Поріг сприйняття змін гучності людиною становить приблизно 1 дБ, тоді як розрахована зміна є набагато меншою. Це означає, що кодування повідомлення методом LSB не впливає на суб'єктивне сприйняття гучності аудіосигналу.

Середнє значення абсолютної амплітуди початкового сигналу дорівнювало 0.0125122657045722, а модифікованого сигналу — 0.01249700877815485. Різниця між цими значеннями складає приблизно 1.52×10^{-5} , що також є дуже малим

значенням. Це свідчить про те, що амплітудна структура сигналу зберігається після вбудовування повідомлення. Незначна зміна середнього значення амплітуди вказує на те, що метод LSB не вносить суттєвих викривлень, які могли б вплинути на якість звуку або призвести до появи артефактів при відтворенні.

Аналіз спектрограм також підтверджує ефективність методу. Спектрограма початкового сигналу відображає часово-частотну структуру аудіосигналу з яскраво вираженими мовними компонентами в середньочастотному діапазоні (500–4000 Гц), що відповідають основним формантам мовлення та забезпечують розбірливість мови. Високочастотний діапазон (4000–8000 Гц) містить слабші компоненти, включаючи сибілянти та високочастотні шуми, важливі для передачі тембру та чіткості звуків. Низькочастотний діапазон (до 500 Гц) містить базові тональні частоти голосу та гармоніки, що визначають інтонацію та загальний тембр.

При аналізі спектрограми модифікованого сигналу (рис. 6.2) спостерігаємо, що вона практично ідентична спектрограмі початкового сигналу (рис. 6.3).

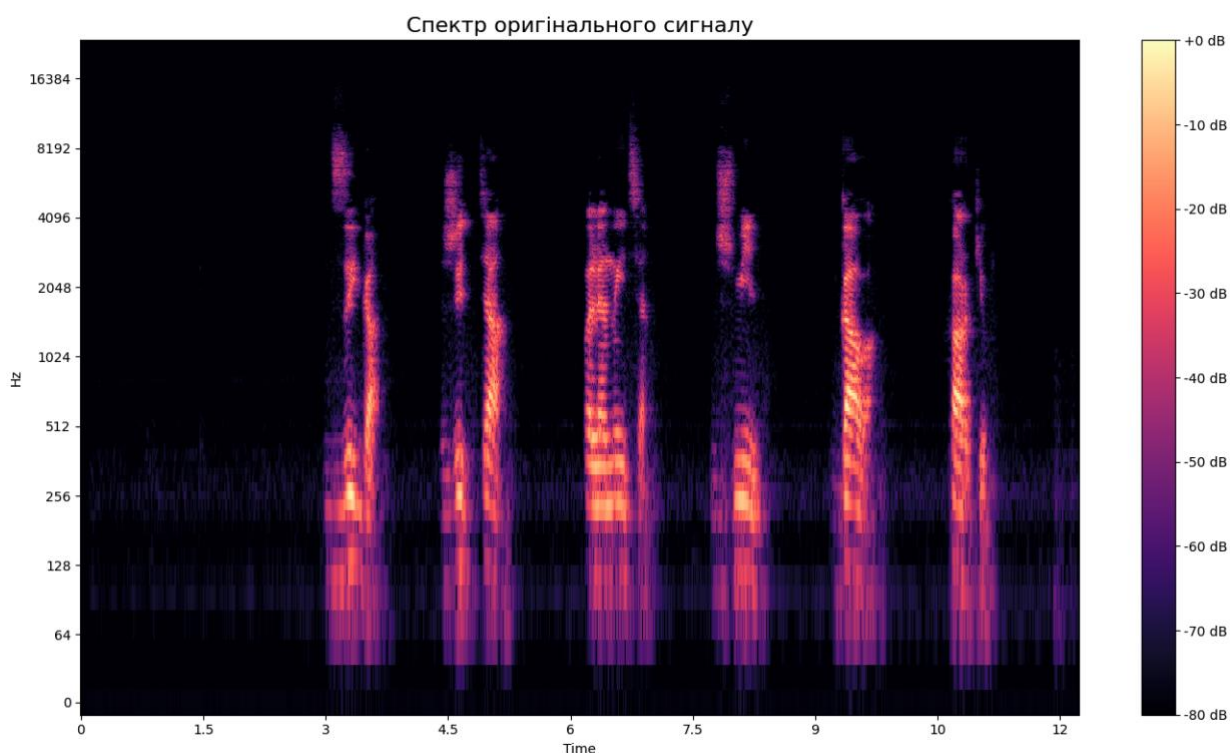


Рисунок 6.2 – Спектрограма оригінального сигналу

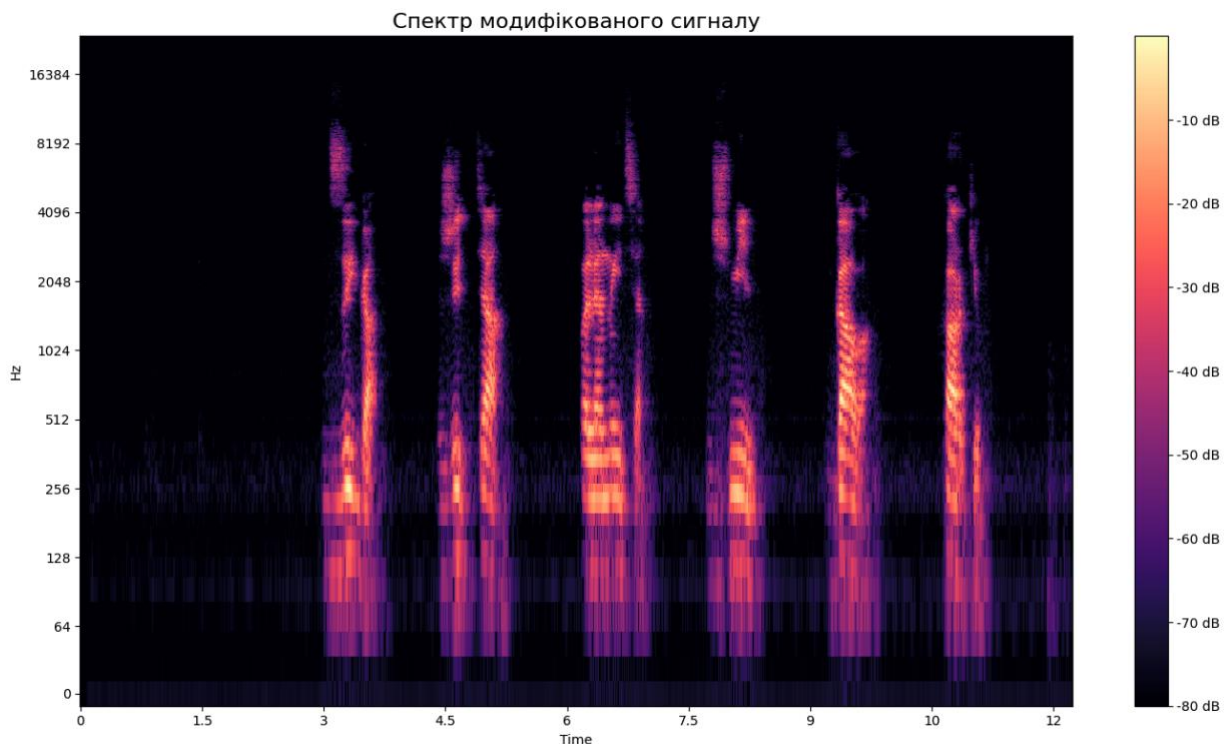


Рисунок 6.3 – Спектрограма сигналу з закодованим повідомленням

Частотний розподіл залишається незмінним у всіх діапазонах, що означає, що енергетичний розподіл по частотах не зазнав суттєвих змін. Часові характеристики також збережені без змін, всі мовні звуки та їх тривалість залишилися такими ж, що свідчить про відсутність викривлень, які могли б вплинути на розбірливість мовлення.

Відсутність видимих артефактів або шумів підтверджує, що метод LSB не вплинув на часово-частотну структуру аудіосигналу (рис. 6.4-6.5). Це важливо, оскільки навіть незначні зміни в частотному домені можуть вплинути на сприйняття звуку, особливо в мовних сигналах, де розбірливість та природність є критичними.

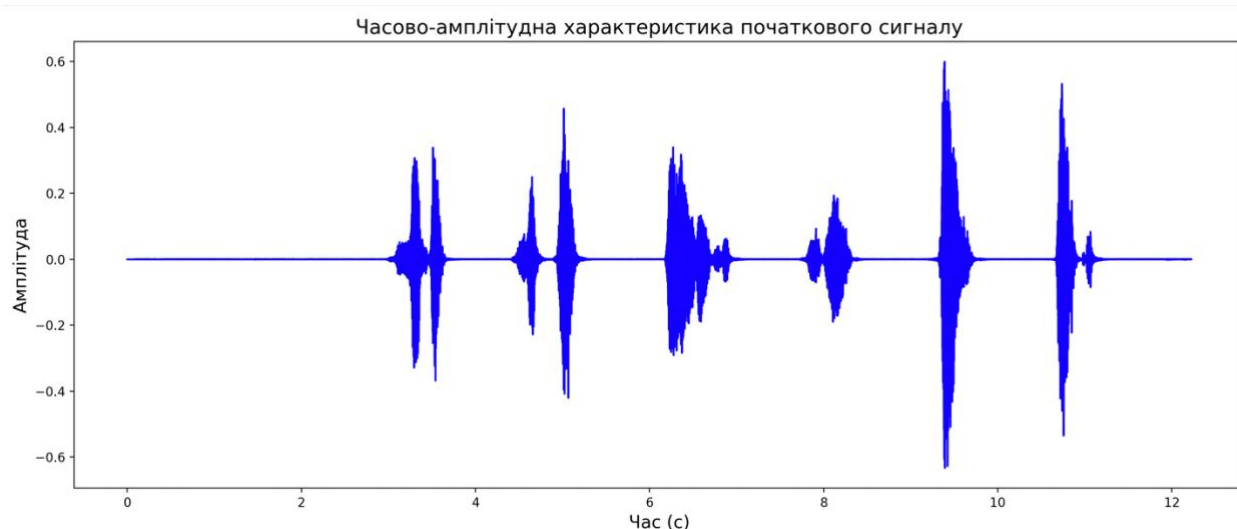


Рисунок 6.4 – Часово амплітудна характеристика оригінального сигналу

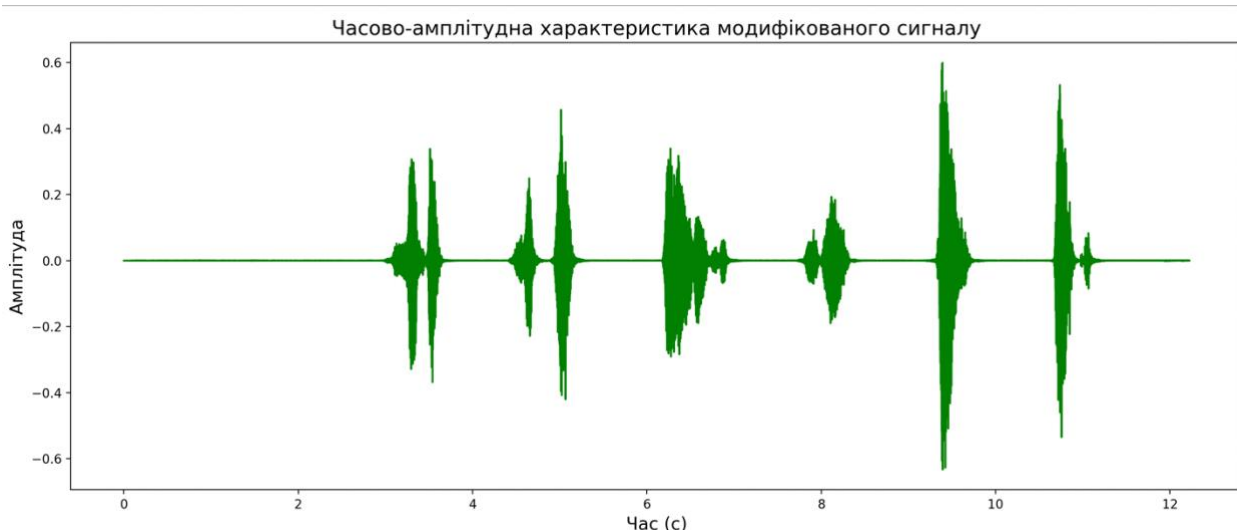


Рисунок 6.5 – Часово амплітудна характеристика сигналу з закодованим повідомленням

Додатковий аналіз відношення сигнал-шум (SNR) показує, що внесений шум є значно меншим за корисний сигнал. Високі значення SNR свідчать про те, що модифікації, внесені в сигнал, мають дуже низький енергетичний рівень порівняно з самим сигналом, і такий шум є непомітним для людини, оскільки його енергія знаходиться нижче порогу сприйняття.

Метод LSB використовує особливості людського слуху, зокрема нечутливість до незначних змін у найменш значущих бітах аудіосигналу. Це дозволяє приховувати інформацію без впливу на якість звучання. Завдяки ефекту

маскування, сильні звуки маскують слабкі, тому внесені зміни залишаються непомітними. Людське вухо менш чутливе до змін фази та амплітуди на високих частотах, що сприяє ефективності методу.

Підсумовуючи, кодування текстового повідомлення методом LSB є ефективним засобом стеганографії в аудіосигналах. Мінімальні зміни параметрів сигналу підтверджують, що інформація може бути прихована без помітного впливу на якість звуку. Аудіосигнал після вбудовування повідомлення зберігає всі важливі акустичні характеристики, включаючи розбірливість мовлення, тембр та інтонацію. Як статистичний, так і спектральний аналіз показують, що зміни в сигналі є настільки малими, що їх неможливо виявити без спеціалізованих інструментів. Це робить метод LSB підходящим для ситуацій, де необхідно приховати інформацію в аудіосигналі без втрати його якості.

6.4 Інтеграція в середовище IoT

Розроблений алгоритм обробки мовного аудіо файлу українською мовою дозволяє удосконалити існуючі підходи до обміну інформацією засобами IoT та має широкий спектр застосування, зокрема в освітніх та побутових призначеннях.

Розроблений алгоритм може бути використаний в контексті “розумного будинку” та оптимізований для української локалізації, зокрема в реалізації у пристроях з обмеженими обчислювальними ресурсами, наприклад, у невеликі мікроконтролери чи мінікомп’ютери. Кінцевий обсяг програмного коду становить приблизно 20КБ, що забезпечує можливість інтеграції алгоритму в сучасні пристрої (розділ 1.2). Завдяки технологіям фільтрації, обробки, детектуванню голосу (VAD) та адаптивному шумозниженню, система коригує сигнали відповідно до акустики приміщення, типових фонів, тембру голосу та можливостей записуючого обладнання [115]. Це дозволяє точно й ефективно розпізнавати команди українською мовою, передавати їх центральному вузлу керування та виконувати потрібні дії. Крім того, програмні модулі, які забезпечують кодування за методом LSB, можуть використовуватися для обміну прихованими повідомленнями або

зберігання конфіденційних даних у вигляді аудіозаписів, що суттєво підвищує рівень безпеки й конфіденційності мережі [116].

Використовуючи приклад проєкту “IoT Based Home Automation System With Speech Recognition” із платформи Instructables [117], де реалізовано голосове керування домашніми пристроями, зазвичай за рахунок розпізнавання команд (переважно англomовних) та подальшої передачі сигналу про вмикання або вимикання певного обладнання. Інтеграція алгоритму з адаптивним пригніченням шуму, оптимізацією під українську мову та стеганографічним кодуванням тексту методом LSB, дозволить розширити та оптимізувати функціонал. По-перше, адаптивна обробка мовних сигналів підвищить точність розпізнавання, оскільки методи шумозниження, допоможуть відфільтрувати зайві звуки (фонові розмови, телевізор, вуличний шум). По-друге, оптимізація алгоритму під українську дасть змогу користувачам спілкуватися рідною мовою без залучення зовнішніх сервісів (перекладачів, тощо), зберігаючи конфіденційність і стабільність роботи навіть без активного інтернет-з’єднання. По-третє, реалізація методу LSB для стеганографічного вбудовування або зберігання текстових команд безпосередньо в аудіосигналі уможливить приховане передавання та архівування даних: наприклад, ключів доступу чи логів команд. Це підвищить рівень безпеки без необхідності в додатковому просторі для зберігання супровідних текстових файлів. Крім того, така система зручна для локального розгортання на мікроконтролерах і SBC (розробки Raspberry Pi, тощо), які мають обмежені обчислювальні ресурси, адже алгоритм не потребує надмірної потужності й водночас забезпечує точне розпізнавання та надійний захист даних. Об’єднання методів з уже наявною базовою функціональністю дозволить збудувати ефективну та безпечну систему розумного дому, що підтримує голосові команди українською мовою з покращеним розпізнаванням команд у зашумленому середовищі та можливістю прихованого зберігання або передачі текстової інформації в аудіопотоці.

Також, варто зазначити, що розроблена технологія пропонує нові рішення для обробки мовного сигналу українською мовою в середовищі IoT, та може знайти застосування, наприклад для запису лекцій, як в аудиторії так і онлайн, з

подальшою обробкою від шумів розпізнаванням мови та вбудовуванням розшифрованого тексту безпосередньо у звуковий файл. Передача закодованого повідомлення методом LSB дає можливість застосування інформації без потреби в додатковому просторі для зберігання та негативного впливу на якість сигналу. Це дає змогу користувачу відтворювати запис як звичайне аудіо або зчитувати з нього текст для створення субтитрів чи подальшого збереження в зручний формат. Такий підхід спрощує пошук потрібних фрагментів, робить матеріал доступним людям із порушеннями слуху та не потребує додаткового простору для зберігання даних.

Завдяки поєднанню розпізнавання української мови, стійкості до різноманітних шумів та умов запису, декодування та вбудовування додаткової інформації в аудіофрагмент алгоритм дозволяє масштабувати рішення та легко інтегрувати його в сучасні IoT-архітектури — від невеликих побутових приладів і ноутбуків до промислових серверів та хмарних сховищ, що робить описану технологію універсальним інструментом для збереження, передачі та аналізу мовного контенту.

Висновки до розділу 6.

Детально розглянуто процес розпізнавання мовлення, кодування тексту у бінарний формат та його вбудовування в аудіосигнал за допомогою методу найменш значущого біта (LSB). Спектральний аналіз підтвердив, що частотний розподіл аудіосигналу після вбудовування повідомлення не зазнає суттєвих змін, зокрема у середньочастотному та високочастотному діапазонах. Це гарантує, що якість звуку залишається високою, а мовлення — розбірливим і природним. Досліджено можливість передачі даних за протоколом LoRaWAN та подальшу інтеграцію в системи IoT.

Таким чином, комплексний підхід до кодування мовного сигналу та застосування методу LSB дозволив успішно приховати текстову інформацію в аудіофайлі без шкоди для його акустичних властивостей. Це відкриває можливості

для швидкої та ефективної передачі даних у різноманітних застосунках, де необхідно зберегти високу якість.

ВИСНОВКИ

В процесі виконання дисертаційного дослідження отримано такі результати:

1. Проведено детальний аналіз та вибір основного обладнання для запису мовних аудіосигналів, яке буде відповідати необхідним вимогам щодо рівня внутрішніх шумів, розбірливості записаної інформації та ширини частотного діапазону для подальшого аналізу та обробки, забезпечивши при цьому виборі мінімальний вплив обладнання на самий записаний аудіо фрагмент. Додатково обрано обладнання для забезпечення роботи в середовищі Інтернету речей, а саме: процесор Heltec WiFi LoRa 32 V2 для обробки та передачі мовного сигналу, мікрофон SPH0645LM4H для запису мовних аудіосигналів.

2. Вперше проведено ряд експериментів з використанням мови програмування Python для реалізації та поєднання різних методів обробки звуку, таких як: зниження рівня шуму, видалення постійної складової, багатосмугова компресія, нормалізація сигналу, метод спектрального віднімання, стеганографічний підхід передачі аудіо інформації. Встановлено, що за рахунок впровадження адаптивного спектрального аналізу (розбиття на часові фрейми, ідентифікація піків, аналіз формант, оцінка низько- та високочастотних компонент та визначення фундаментальної частоти) вдалось забезпечити збереження ключових характеристик сигналу, при цьому підвищивши розбірливість мовлення та видалення шуму. Визначено основні підходи роботи з українською фонетичною групою в умовах зашумлення. Підготовлено для проведення практичного експерименту ряд мовних сигналів, записаних українською мовою з технічними дефектами і для виправлення цих дефектів, зокрема, для зменшення шумового забруднення, запропоновано новий програмний алгоритм обробки, який складається з послідовних етапів і має риси циклічності. Отримано підтвердження, що поєднання запропонованих підходів дозволяє ефективно зменшити рівень шумів та покращити якість звуку в умовах реального використання забезпечуючи підвищення показника відношення сигнал/шум на 7.51 дБ. Використання шумового порогу дозволило ефективно усунути низькорівневі фонові шуми в паузах між

словами та фразами, підвищуючи чистоту та чіткість звуку без втрати важливих мовних компонентів. Завдяки попередній обробці сигналу, а саме застосуванню нормалізації рівня гучності, видалення постійної складової, аналізу фундаментальної частоти та застосуванню фільтру високих та низьких частот, досягнуто підвищення середнього показника амплітуди на 0.017860, відношення сигнал/шум на 2.02 дБ та RMS сигналу на 5.56 дБ.

3. Розроблено алгоритм додавання супутньої прихованої інформації в аудіо файл, який отримано за допомогою підключення бібліотеки VOSK для офлайн-розпізнавання мовлення та містить записану розмовну компоненту українською мовою. Для реалізації алгоритму використано послідовно процес кодування розпізнаного тексту у бінарний формат з використанням стандарту кодування UTF-8 та метод найменш значущого біта (LSB). Знайдено, що на основі запропонованого алгоритму, модифікований результуючий сигнал майже не відрізняється від оригінального. Так, рівень гучності оригінального сигналу складає 0,040723, а модифікованого 0,040718 (абсолютна різниця складає $6,07 \times 10^{-6}$). Середнє значення абсолютної амплітуди для модифікованого сигналу виявилось на рівні 0,012494, яке відрізняється від аналогічного значення для оригінального сигналу на $1,52 \times 10^{-5}$. Таким чином, незначне зниження середньої амплітуди на 0.000029, підвищення рівня гучності та значення максимальної амплітуди на 0.01 дБ для сигналу з закодованим повідомленням свідчать про збереження природної динаміки та інтонаційних характеристик мовлення, роблячи приховану інформацію непомітною для слухача.

СПИСОК ДЖЕРЕЛ ПОСИЛАНЬ

1. K. K. Sethy, L. M. Varalakshmi, Rajkumar E., R. R. Pandey, Karthika R. N. B., and P. Vijayakumar, IoT based speech recognition system, *RECENT TRENDS IN SCIENCE AND ENGINEERING*, vol. 2393, p. 020096, 2022. doi: 10.1063/5.0074140.
2. R. Vijayakumari, A. A. Bannatti, R. Maranan, J. Kaur and S. Kamatchi, English Speech Recognition using Convolution Neural Networks and IoT, *2024 IEEE International Conference on Computing, Power and Communication Technologies (IC2PCT)*, Greater Noida, India, 2024. P. 711-716, doi: 10.1109/IC2PCT60090.2024.10486292.
3. L. Farhan, S. T. Shukur, A. E. Alissa, M. Alrweg, U. Raza, and R. Kharel, A survey on the challenges and opportunities of the Internet of Things (IoT), *2017 Eleventh International Conference on Sensing Technology (ICST)*, Dec. 2017. P. 1–5. doi: 10.1109/icsenst.2017.8304465.
4. J.-M. Valin, A Hybrid DSP/Deep Learning Approach to Real-Time Full-Band Speech Enhancement, *arXiv preprint arXiv:1709.08243*, 2018. doi: 10.48550/arXiv.1709.08243.
5. H. Schröter, A. N. Escalante-B., T. Rosenkranz, and A. Maier, DeepFilterNet: A Low Complexity Speech Enhancement Framework for Full-Band Audio based on Deep Filtering, *arXiv preprint arXiv:2110.05588*, 2021. doi: 10.48550/arXiv.2110.05588.
6. C. Zonios and V. Tenentes, Energy Efficient Speech Command Recognition for Private Smart Home IoT Applications, *2021 10th International Conference on Modern Circuits and Systems Technologies (MOCAST)*, Thessaloniki, Greece, 2021. P. 1-4. doi: 10.1109/MOCAST52088.2021.9493392.
7. Heltec WiFi LoRa 32 V, URL: <https://heltec.org/project/wifi-lora-32v2/>.
8. SPH0645LM4H, URL: https://www.rcscomponents.kiev.ua/product/sph0645lm4h-b_158651.html.
9. Що таке технологія LoRaWAN і як вона працює, *ДЕПС*, URL: <https://deps.ua/ua/knowegable-base/reference-information/66634.html>.

10. LoRaWAN Specification v1.1, *LoRa Alliance*, URL: <https://resources.lora-alliance.org/technical-specifications/lorawan-specification-v1-1>.
11. Дідковський В.С., Луньова С.А., Богданов О.В. Архітектурна акустика. – Київ: НТУУ «КПІ», 2012. С.385.
12. R. A. Rayburn, Microphone Measurements, Standards, and Specifications, Eargle's The Microphone Book, 2012. P. 129–143. doi: 10.1016/b978-0-240-82075-0.00007-9.
13. G. Ballou, J. Ciaudelli, and V. Schmitt, *Electroacoustic Devices: Microphones and Loudspeakers*, 1st ed., Elsevier, 2009. P. 3–193. doi: 10.1016/B978-0-240-81267-0.00003-X.
14. S. L. Ehrlich, AIP Handbook of Condenser Microphones (Theory, Calibration, and Measurements), *The Journal of the Acoustical Society of America*, vol. 98, no. 1, Jul. 1995. P. 20–20. doi: 10.1121/1.413754.
15. Rumsey, F. *Digital Audio Recording Formats and Editing Principles*. In: Havelock, D., Kuwano, S., Vorländer, M. (eds) Handbook of Signal Processing in Acoustics. Springer, New York, NY. 2008. DOI: 10.1007/978-0-387-30441-0_36.
16. Smith, S. W. *Modern Recording Techniques*. 8th Edition, Focal Press, 2017.
17. Microphone Techniques, Practical Recording Techniques, P. 127–162, May 2013, doi: 10.4324/9780240824635-13.
18. Everest, F. A., & Pohlmann, K. C. *Master Handbook of Acoustics*. McGraw-Hill, 2015.
19. Rasmussen, B. Sound insulation between dwellings – Requirements in building regulations in Europe. *Applied Acoustics*, 2010. 71(4). P. 373-385. doi: 10.1016/j.apacoust.2009.08.011.
20. E.C Sewell, Transmission of reverberant sound through a single leaf partition surrounded by an infinite rigid baffle, *Journal of Sound and Vibration*, 1970. 12. P. 21-32.
21. О. О. Дворник, *Методи та системи оцінки та корекції акустичних характеристик приміщень для публічних виступів*, дис. доктора філософії,

Національний технічний університет України "Київський політехнічний інститут імені Ігоря Сікорського", Київ, Україна, 2024. URL: <https://ela.kpi.ua/items/b20dd0c7-6f69-4d62-8f9c-f261f86b5c41>.

22. Henríquez, V.C., & Rasmussen, K. Final report on the key comparison, CCAUV.A-K3. *Metrologia*, 43(1A), 09001, 2006, C1–84.

23. R. A. Rayburn, *Microphone Measurements, Standards, and Specifications*, Eargle's The Microphone Book, 2012. P. 129–143. doi: 10.1016/b978-0-240-82075-0.00007-9.

24. IEC 60268-1:2018 Sound system equipment – Part 1: General requirements. International Electrotechnical Commission, 2018.

25. М. М. Злобін, *Акустичні особливості студії звукозапису*, кваліфікаційна робота, Національна академія керівних кадрів культури і мистецтв, Київ, Україна, 2023. URL: <https://shorturl.at/Mv83wi>.

26. Sennheiser AG. *MKH 800 Studio Microphone*. URL: <https://en-us.sennheiser.com/mkh-800>.

27. IEC 60268-4:2020 Sound system equipment – Part 4: Microphones. International Electrotechnical Commission, 2020.

28. DPA Microphones. *DPA 3530 Studio Microphone*. URL: <https://dpa-microphones.com/products/dpa-3530>.

29. AKG. *C3000B Condenser Microphone*. URL: <https://www.ake.com/Microphones/C3000B.html>.

30. R. Tavčar, J. Bojkovski, and S. Beguš, Sound-card-based Johnson noise thermometer, *Measurement*, vol. 225, p. 114077, Feb. 2024, doi: 10.1016/j.measurement.2023.114077.

31. J. G. Švec and S. Granqvist, Guidelines for Selecting Microphones for Human Voice Production Research, *ERIC*, 2010. URL: <https://eric.ed.gov/?id=EJ909025>.

32. M. Stavrovskyi, *Obrobka Audio (Rozrobka ekvalayzera dlya tsyfrovyykh zvukovykh robochykh stantsiy)*, Bachelor's thesis, National University of Kyiv-Mohyla Academy, 2021. URL: <https://ekmair.ukma.edu.ua/handle/123456789/21984>.

33. О. Гребінь, *Прикладна акустика – 1. Електроакустика*, навчальний посібник, Київ: Національний технічний університет України «КПІ», 2018. URL: https://ela.kpi.ua/bitstream/123456789/23604/1/Electroacoustic_navchalnyi-posibnyk.pdf.
34. J. Šaliga and L. Michaeli, Software for metrological characterisation of PC sound cards, *Computer Standards & Interfaces*, vol. 24, no. 3, 2002. P. 209–216. doi: 10.1016/S0920-5489(02)00077-6.
35. H. M. Kasem and M. El-Sabrouty, A Comparative Study of Audio Compression Based on Compressed Sensing and Sparse Fast Fourier Transform (SFFT): Performance and Challenges, *arXiv preprint*, arXiv:1403.3061, Mar. 2014. doi: 10.48550/arXiv.1403.3061.
36. C. D. Giurcăneanu, I. Tăbuș and J. Astola, "Adaptive context based sequential prediction for lossless audio compression," *9th European Signal Processing Conference (EUSIPCO 1998)*, Rhodes, Greece, 1998. P. 1-4.
37. S. Srinivas and P. V. Kumar, Spectral Fluctuation Analysis for Audio Compression Using Adaptive Wavelet Decomposition, in *Lecture Notes in Computer Science*, vol. 642, Springer, Berlin, Heidelberg, 2010. P. 575–584. doi: 10.1007/978-3-642-12214-9_71.
38. C. E. Shannon, A mathematical theory of communication, *The Bell System Technical Journal*, vol. 27, no. 3, July 1948. P. 379-423. doi: 10.1002/j.1538-7305.1948.tb01338.x.
39. F. J. Harris, "On the use of windows for harmonic analysis with the discrete Fourier transform," in *Proceedings of the IEEE*, vol. 66, no. 1, Jan. 1978, P. 51-83, doi: 10.1109/PROC.1978.10837
40. I. Daubechies, *Ten Lectures on Wavelets*, 1st ed. Philadelphia, PA, USA: SIAM, 1992, P. 1–50. DOI: 10.1137/1.9781611970104.
41. Метод визначення формантних частот із використанням спектрального розкладання мовного сигналу, *BICT*, vol. 1, no. 6, Mar. 2023. P. 51–60. doi: 10.17721/ISTS.2023.1.51-60.

42. MathWorks, Discrete Fourier Transform, *MathWorks Documentation*. URL: <https://www.mathworks.com/help/signal/ug/discrete-fourier-transform.html>.
43. ITU-R, *Recommendation ITU-R BS.1770-4: Algorithms to measure audio programme loudness and true-peak audio level*, International Telecommunication Union, Geneva, Switzerland, 2015.
44. Jyh-Cherng Gu and Sun-Li Yu, Removal of DC offset in current and voltage signals using a novel Fourier filter algorithm, *IEEE Transactions on Power Delivery*, vol. 15, no. 1, 2000. P. 73–79. doi: 10.1109/61.847231.
45. National Instruments, *The Fundamentals of FFT-Based Signal Analysis and Measurement*, National Instruments Application Note 041, 2009. URL: https://www.sjsu.edu/people/burford.furman/docs/me120/FFT_tutorial_NI.pdf.
46. S. Wang, C. Pham, and B. A. Plummer, A Unified Framework for Connecting Noise Modeling to Boost Noise Detection, *arXiv preprint*, arXiv:2312.00827, Dec. 2023. doi: 10.48550/arXiv.2312.00827.
47. Y. Wang, X. Li, and Z. Chen, Design Methods of High-order Low Pass Filters, in *Proceedings of the 2023 International Conference on Software, Electronics and Electrical Engineering (ICSECE)*, Guangzhou, China, 2023. P. 45–52. doi: 10.1109/ICSECE58870.2023.10263550.
48. N. J. Dahl, P. L. Muntal, and M. A. E. Andersen, Time-Based High-Pass, Low-Pass, Shelf, and Notch Filters, *Elektronika ir Elektrotechnika*, vol. 29, no. 3, 2023. P. 45–52. doi: 10.5755/j02.eie.35277.
49. S. W. Smith, *The Scientist and Engineer's Guide to Digital Signal Processing*, 2nd ed. California Technical Publishing, 1999.
50. J. Proakis and D. G. Manolakis, *Digital Signal Processing: Principles, Algorithms, and Applications*, 4th ed., Upper Saddle River, NJ, USA: Prentice Hall, 2007.
51. S. D. Gapochenko, T. M. Shelest, and S. S. Kryvonos, *Osnovy spektralnoho analizu* (Foundations of Spectral Analysis). Kharkiv, Ukraine: NTU "KhPI", 2020. URL: <https://repository.kpi.kharkov.ua/bitstreams/eb72151f-485f-4331-9992-12553a9915d4/download>.

52. В. Білінський, Квантування сигналів, *Електронні системи*, Вінницький національний технічний університет. URL: https://web.posibnyky.vntu.edu.ua/firen/6bilynskyj_elektronni_systemy/43.htm.

53. W. Yifan, C. Kai, X. Bo, Y. Yunpeng, and L. Yun, A high precision fundamental frequency measurement method based on FPGA, *2019 14th IEEE International Conference on Electronic Measurement & Instruments (ICEMI)*, Nov. 2019. P. 254–260. doi: 10.1109/icemi46757.2019.9101426.

54. Laura Verde, Giuseppe De Pietro, Giovanna Sannino, A methodology for voice classification based on the personalized fundamental frequency estimation, *Biomedical Signal Processing and Control*, Volume 42, 2018. P. 134-144. doi: <https://doi.org/10.1016/j.bspc.2018.01.007>.

55. C. Jo and J. Wang, Measuring Variations of Voice Source and Vocal Tract Characteristics from Korean Emotional Voice, *Sixth International Conference on Intelligent Systems Design and Applications*, vol. 2, Oct. 2006. P. 800–805. doi: 10.1109/isda.2006.253715.

56. Yen-Liang Shue and M. Iseli, The role of voice source measures on automatic gender classification, *2008 IEEE International Conference on Acoustics, Speech and Signal Processing*, Mar. 2008. P. 4493–4496. doi: 10.1109/icassp.2008.4518654.

57. M. Kumari and I. Ali, An efficient algorithm for Gender Detection using voice samples, *2015 Communication, Control and Intelligent Systems (CCIS)*, Nov. 2015. P. 221–226. doi: 10.1109/ccintels.2015.7437912.

58. F. Lastow, E. Ekberg, and P. Nugues, Language-agnostic Age and Gender Classification of Voice using Self-supervised Pre-training, *2022 Swedish Artificial Intelligence Society Workshop (SAIS)*, Jun. 2022. P. 1–9. doi: 10.1109/sais55783.2022.9833071.

59. L. Pfeifenberger, T. Schrank, M. Zohrer, M. Hagmuller, and F. Pernkopf, Multi-channel speech processing architectures for noise robust speech recognition: 3rd CHiME challenge results, *2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, Dec. 2015. P. 452–459. doi: 10.1109/asru.2015.7404830.

60. O. Hazrati, S. Ghaffarzadegan, and J. H. L. Hansen, Leveraging automatic speech recognition in cochlear implants for improved speech intelligibility under reverberation, *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Apr. 2015. P. 5093–5097. doi: 10.1109/icassp.2015.7178941.
61. Vosk, "Offline speech recognition API for Android, iOS, Raspberry Pi and servers," Alphacephei. URL: <https://alphacephei.com/vosk/>.
62. Vosk Speech Recognition Toolkit. URL: <https://alphacephei.com/vosk/>.
63. J. Little, Impact of the ASCII code and printing devices on conventions for alphanumeric display terminals: Part 1, *Communications Society*, vol. 10, no. 1, Mar. 1973. P. 7–10. doi: 10.1109/mcomd.1973.1145826.
64. I. Jemal, M. A. Haddar, O. Cheikhrouhou, and A. Mahfoudhi, ASCII Embedding: An Efficient Deep Learning Method for Web Attacks Detection, *Pattern Recognition and Artificial Intelligence*, 2021. P. 286–297. doi: 10.1007/978-3-030-71804-6_21.
65. N. Nikolov, Y. Hu, M. X. Tan, and R. H. R. Hahnloser, Character-level Chinese-English Translation through ASCII Encoding, *Proceedings of the Third Conference on Machine Translation: Research Papers*, 2018 P. 10–16. doi: 10.18653/v1/w18-6302.
66. ASCII Codes, *Electronics Simplified*, 2011. P. 335–336, doi: 10.1016/b978-0-08-097063-9.10020-2.
67. F. Yergeau, RFC 3629: UTF-8, a Transformation Format of ISO 10646, *Internet Engineering Task Force (IETF)*, Nov. 2003. URL: <https://www.rfc-editor.org/rfc/rfc3629>.
68. M. Crispin, UTF-9 and UTF-18 Efficient Transformation Formats of Unicode, *RFC Editor*, Apr. 2005. doi: 10.17487/rfc4042.
69. ASLANTAŞ, F., HANİLÇİ, C. Comparative Analysis Of Audio Steganography Methods. *Journal of Innovative Science and Engineering (JISE)*, 2022. P.1–6. doi:10.38088/jise.932549.
70. Є. В. Світловський, К. О. Трапезон, Стеганографічні підходи до оброблення аудіо сигналів, *Вісник Кременчуцького національного університету*

імені Михайла Остроградського, том 3, 2023. С. 185–192, doi:10.32782/1995-0519.2023.3.22.

71. C. Cachin, An information-theoretic model for steganography, *Information and Computation*, vol. 192, 2004. P. 41–56. doi: 10.1016/j.ic.2004.02.003.

72. О. М. Іванова, О. В. Дрозд, К. В. Защолкін, і М. О. Кузнєцов, Підхід до нееквівалентного стеганографічного вбудовування додаткових даних у програмний код блоків lut FPGA, *Вісник Кременчуцького національного університету імені Михайла Остроградського*, вип. 6/2021(131), 2021. С. 60–61, doi: 10.30929/1995-0519.2021.6.60-65.

73. S. Pinjerla, S. Rao S, and C. Reddy P, Sampling Rate Conversion Techniques- A Review, *2021 4th International Conference on Recent Trends in Computer Science and Technology (ICRTCST)*, Feb. 2022. P. 278–282. doi: 10.1109/icrtcst54752.2022.9781914.

74. Focusrite, *Scarlett Solo 2nd Gen User Manual*, 2016. URL: <https://focusrite.com/products/audio-interfaces/scarlett-solo>.

75. B. T. Bosworth, W. R. Bernecky, J. D. Nickila, B. Adal, and G. C. Carter, Estimating Signal-to-Noise Ratio (SNR), *IEEE Journal of Oceanic Engineering*, vol. 33, no. 4, Oct. 2008. P. 414–418. doi: 10.1109/joe.2008.2001780.

76. Audient, *Audient iD4 User Manual*, 2016. URL: <https://audient.com/products/audio-interfaces/id4/overview/>.

77. H. Zhang, Q. Fu, and Y. Yan, A Compact-Microphone-Array-Based Speech Enhancement Algorithm Using Auditory Subbands and Probability Constrained Postfilter, *2008 Hands-Free Speech Communication and Microphone Arrays*, May 2008. P. 192–195. doi: 10.1109/hscma.2008.4538719.

78. Kai-Tai Song and Jian-Liang Chen, Sound direction recognition using a condenser microphone array, *Proceedings 2003 IEEE International Symposium on Computational Intelligence in Robotics and Automation. Computational Intelligence in Robotics and Automation for the New Millennium (Cat. No.03EX694)*, vol. 3, P. 1445–1450. doi: 10.1109/cira.2003.1222210.

79. Y. Peled and B. Rafaely, Study of Speech Intelligibility in Noisy Enclosures Using Spherical Microphones Arrays, *2008 Hands-Free Speech Communication and Microphone Arrays*, May 2008. P. 160–163. doi: 10.1109/hscma.2008.4538711.
80. Freedman Electronics, RØDE NT2-A Multi-Pattern Dual 1" Condenser Microphone: User Manual, *Freedman Electronics Pty Ltd*, Sydney, Australia, 2019. URL: <https://rode.com/en/microphones/condenser/nt2-a>.
81. E. De Sena, H. Hacıhabiboglu, and Z. Cvetkovic, A generalized design method for directivity patterns of spherical microphone arrays, *2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, May 2011. P. 125–128. doi: 10.1109/icassp.2011.5946344.
82. H. F. Olson, *Elements of Acoustical Engineering*, N. Y., New York:D. Van Nostrand Co., 1940.
83. H. F. Olson, Polydirectional Microphone, *Proceedings of the IRE*, vol. 32, no. 2, Feb. 1944. P. 77–82. doi: 10.1109/jrproc.1944.229734.
84. A. Magueresse, V. Carles and E. Heetderks, Low-resource Languages: A Review of Past Work and Future Challenges, 2020.
85. A. Kramov and S. Pogorilyy, Cross-Lingual Named Entity Recognition for the Ukrainian Language Based on Word Alignment, *2022 IEEE 4th International Conference on Advanced Trends in Information Theory (ATIT)*, Kyiv, Ukraine, 2022. P. 213-218. doi: 10.1109/ATIT58178.2022.10024219.
86. G. Benmouyal, Removal of DC-offset in current waveforms using digital mimic filtering, *IEEE Transactions on Power Delivery*, vol. 10, no. 2, Apr. 1995. P. 621–630. doi: 10.1109/61.400869.
87. Y. Mao, Z. Yiqiang, Z. Gongyuan and X. Ruishan, A dB-linear switched-resistor CMOS programmable gain amplifier with DC offset cancellation, *2017 International Conference on Electron Devices and Solid-State Circuits (EDSSC)*, Hsinchu, Taiwan, 2017. P. 1-2. doi: 10.1109/EDSSC.2017.8126420.
88. EBU, *EBU R128: Loudness Normalisation and Permitted Maximum Level of Audio Signals*, European Broadcasting Union, 2020.

89. S. Schlecht, L. Fierro, V. Välimäki, and J. Backman, Audio Peak Reduction Using a Synced Allpass Filter, in *Proceedings of the 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Singapore, 2022. P. 246–250. doi: 10.1109/ICASSP43922.2022.9747877.
90. AES/EBU, *AES Recommended Practice for Digital Audio Engineering — Serial Transmission Format for Two-Channel Linearly Represented Digital Audio Data (AES3)*, Audio Engineering Society, 2003.
91. L. Garcia, C. Benitez, J. C. Segura and S. Umesh, Combining speaker and noise feature normalization techniques for Automatic Speech Recognition, *2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Prague, Czech Republic, 2011. P. 5496-5499. doi: 10.1109/ICASSP.2011.5947603.
92. M. Thornton, Is Normalisation Still Important? *Production Expert*, Apr. 14, 2022. URL: <https://www.production-expert.com/production-expert-1/is-normalisation-still-important>.
93. M. Korycki and P. Szczepaniak, Normalization of audio signals for the needs of machine learning, in *Proceedings of the 2023 19th Telecommunications Forum (TELFOR)*, Belgrade, Serbia, 2023. P. 1–4. doi: 10.1109/TELFOR59449.2023.10372705.
94. M. Bolić, M. Đorđević, and M. Popović, Loudness normalization, in *Proceedings of the 2011 19th Telecommunications Forum (TELFOR)*, Belgrade, Serbia, 2011. P. 978–981. doi: 10.1109/TELFOR.2011.6143744.
95. X. Yan, S. Tan, J. Wang and Y. Wang, A High Accuracy Harmonic Analysis Method Based on All-Phase and Interpolated FFT in Power System, *2011 Asia-Pacific Power and Energy Engineering Conference*, 2011. P. 1-4.
96. O. Bozkurt and Z. C. Taygi, Audio-based gender and age identification, *2014 22nd Signal Processing and Communications Applications Conference (SIU)*, Apr. 2014. P. 1371–1374. doi: 10.1109/siu.2014.6830493.
97. K. Debnath, S. Dhabal and P. Venkateswaran, Design of High-Pass FIR Filter using Arithmetic Optimization Algorithm and its FPGA Implementation, *2022 IEEE Region 10 Symposium (TENSYP)*, Mumbai, India, 2022. P. 1-6, doi: 10.1109/TENSYP54529.2022.9864333.

98. X. Sang and X. Chen, Design, simulation and measurement of split-band digital audio expander and noise-gate, *2009 7th International Conference on Information, Communications and Signal Processing (ICICS)*, Dec. 2009. P. 1–4. doi: 10.1109/icics.2009.5397498.
99. D. M. Kiapuchinski, C. R. E. Lima, and C. A. A. Kaestner, Spectral Noise Gate Technique Applied to Birdsong Preprocessing on Embedded Unit, *2012 IEEE International Symposium on Multimedia*, Dec. 2012. P. 24–27. doi: 10.1109/ism.2012.12.
100. F. Torri, O. Leman, P. Malcovati and A. Baschiroto, IGS–IGD Gate Current Noise Model at Low Frequency, *2024 19th Conference on Ph.D Research in Microelectronics and Electronics (PRIME)*, Larnaca, Cyprus, 2024. P. 1-4. doi: 10.1109/PRIME61930.2024.10559738.
101. E. Lindemann, The continuous frequency dynamic range compressor, *Proceedings of 1997 Workshop on Applications of Signal Processing to Audio and Acoustics*, New Paltz, NY, USA, 1997. doi: 10.1109/ASPAA.1997.625580.
102. Jean-Michel Reveillac, Processing Hardware and Software, in *Recording and Voice Processing, Volume 2: Working in the Studio*, Wiley, 2021. P.1-28. doi: 10.1002/9781119887980.ch1.
103. S. Boll, A spectral subtraction algorithm for suppression of acoustic noise in speech, *ICASSP '79. IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 4, P. 200–203, doi: 10.1109/icassp.1979.1170696.
104. S. Boll, Suppression of Acoustic Noise in Speech Using Spectral Subtraction, *IEEE Transactions on Acoustic speech, Signal processing*, vol-27, 1979. P.113-120.
105. M. Yektaeian and R. Amirfattahi, Comparison of spectral subtraction methods used in noise suppression algorithms, *2007 6th International Conference on Information, Communications & Signal Processing*, 2007. P. 1–4. doi: 10.1109/icics.2007.4449542.
106. Loudness Normalisation and Permitted Maximum Level Of Audio Signals. Geneva: European Broadcasting Union, European Commission, 2014. URL: <https://tech.ebu.ch/publications/r128>.

107. A. Andriella, R. Ros, Y. Ellinson, S. Gannot and S. Lemaignan, Dataset and Evaluation of Automatic Speech Recognition for Multi-lingual Intent Recognition on Social Robots, *2024 19th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, Boulder, CO, USA, 2024. P. 865-869. <https://tech.ebu.ch/docs/r/r128.pdf>.
108. A. Gleave and C. Steinruecken, Making Compression Algorithms for Unicode Text, *2017 Data Compression Conference (DCC)*, Snowbird, UT, USA, 2017. P. 441-441. doi: 10.1109/DCC.2017.58.
109. Adam Gleave and Christian Steinruecken, Making compression algorithms for unicode text, *2017 Data Compression Conference DCC 2017 Snowbird UT USA April 4–7 2017*, 2017. P. 441.
110. Rincy Thayyalakkal Anto, Rajesh Ramachandran, A Compression System for Unicode Files Using an Enhanced Lzw Method, *Pertanika Journal of Science and Technology*, vol.28, no.4, 2020.
111. Hung-Min Sun, King-Hang Wang, Chih-Cheng Liang and Yih-Sien Kao, A LSB substitution compatible steganography, *TENCON 2007 - 2007 IEEE Region 10 Conference*, Taipei, Taiwan, 2007. P. 1-3. doi: 10.1109/TENCON.2007.4429023.
112. Gamal Abdellatif Ibrhaim, Ahmed & Tayel, Mazhar & Zied, Hamed. A Proposed Implementation Method of an Audio Steganography Technique. 2016.
113. A. Westfeld and A. Pfitzmann, Attacks on Steganographic Systems, *Berlin:Springer-Verlag*, vol. 1768, 2000. P. 61-75.
114. J. Mielikainen, LSB matching revisited, *IEEE Signal Processing Letters*, vol. 13, no. 5, May 2006. P. 285-287. doi: 10.1109/LSP.2006.870357.
115. S. Boll, Suppression of acoustic noise in speech using spectral subtraction, in *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 27, no. 2, April 1979. P. 113-120. doi: 10.1109/TASSP.1979.1163209.
116. M. A. Al Sibahee et al., Hiding scrambled text messages in speech signals using a lightweight hyperchaotic map and conditional LSB mechanism, *PLOS ONE*, vol. 19, no. 1, p. e0296469, Jan. 2024. doi: 10.1371/journal.pone.0296469.

117. IoT Based Home Automation System With Speech Recognition, URL: <https://www.instructables.com/Iot-Based-Home-Automation-System-With-Speech-Recog/>.

ДОДАТОК А

КОД ДОСЛІДЖЕНЬ ТА ОБРОБКИ ЗВУКУ

Початкова обробка сигналу

```
import numpy as np
import librosa
from scipy.signal import butter, filtfilt
import soundfile as sf

# Визначення шляхів до файлів
input_file_path = '/origin.wav'
output_file_path = '/processed.wav'

# Завантаження аудіо файлу
audio, fs = librosa.load(input_file_path, sr=None)

# 1. Нормалізація амплітуди
def normalize_amplitude(signal, target_level):
    max_amp = np.max(np.abs(signal))
    if max_amp == 0:
        return signal
    normalized_signal = signal * (target_level / max_amp)
    return normalized_signal

audio_normalized = normalize_amplitude(audio, target_level=0.7)

# 2. Видалення постійної складової
def remove_dc_offset(signal):
    mean_value = np.mean(signal)
    signal_without_dc = signal - mean_value
    return signal_without_dc

audio_no_dc = remove_dc_offset(audio_normalized)

# 3. Аналіз фундаментальної частоти та визначення гендера
def analyze_pitch_and_determine_gender(signal, fs):
    pitches, magnitudes = librosa.piptrack(y=signal, sr=fs)

    pitch_values = []
    for i in range(pitches.shape[1]):
        pitch = pitches[:, i]
        mag = magnitudes[:, i]
        if mag.any():
            index = mag.argmax()
            pitch_freq = pitch[index]
            if 50 < pitch_freq < 500:
                pitch_values.append(pitch_freq)

    if pitch_values:
        average_pitch = np.mean(pitch_values)
```

```

else:
    average_pitch = 0

if average_pitch > 0:
    if average_pitch < 300:
        gender = 'Чоловічий'
    elif average_pitch > 350:
        gender = 'Жіночий'
    else:
        gender = 'Невідомо'
else:
    gender = 'Невідомо'

print(f"Середнє значення pitch: {average_pitch:.2f} Гц")
print(f"Визначений гендер на основі pitch: {gender}")

return average_pitch, gender

average_pitch, gender = analyze_pitch_and_determine_gender(audio_no_dc, fs)

# 4. Застосування low-pass та high-pass фільтрів
def butter_filter(signal, cutoff, fs, order, btype):
    nyq = 0.5 * fs
    normal_cutoff = cutoff / nyq
    b, a = butter(order, normal_cutoff, btype=btype, analog=False)
    filtered_signal = filtfilt(b, a, signal)
    return filtered_signal

# Налаштування частот зрізу
if gender == 'Чоловічий':
    hp_cutoff = max(50, average_pitch * 0.6)
    lp_cutoff = 5000.0
elif gender == 'Жіночий':
    hp_cutoff = max(80, average_pitch * 0.6)
    lp_cutoff = 6000.0
else:
    hp_cutoff = 70.0
    lp_cutoff = 6000.0

print(f"Частота зрізу high-pass фільтра: {hp_cutoff:.2f} Гц")
print(f"Частота зрізу low-pass фільтра: {lp_cutoff:.2f} Гц")

# Порядок фільтрів
hp_order = 1
lp_order = 1

# Застосування high-pass фільтра
audio_highpassed = butter_filter(audio_no_dc, hp_cutoff, fs, hp_order, 'highpass')

# Застосування low-pass фільтра
audio_filtered = butter_filter(audio_highpassed, lp_cutoff, fs, lp_order, 'lowpass')

if write(output_file_path, audio_filtered, fs)

```

Noise gate

```

import numpy as np
import soundfile as sf
import matplotlib.pyplot as plt
import librosa
import librosa.display

def calculate_snr(signal, noise):
    signal_power = np.mean(signal ** 2)
    noise_power = np.mean(noise ** 2)

    if noise_power == 0:
        print("Потужність шуму дорівнює нулю. Неможливо обчислити SNR.")
        return None

    snr = 10 * np.log10(signal_power / noise_power)
    return snr

def compute_envelope(signal, fs, attack_time=0.005, release_time=0.05):
    envelope = np.zeros_like(signal)
    attack_coeff = np.exp(-1.0 / (fs * attack_time))
    release_coeff = np.exp(-1.0 / (fs * release_time))

    envelope[0] = abs(signal[0])
    for n in range(1, len(signal)):
        if abs(signal[n]) > envelope[n-1]:
            envelope[n] = attack_coeff * envelope[n-1] + (1 - attack_coeff) * abs(signal[n])
        else:
            envelope[n] = release_coeff * envelope[n-1] + (1 - release_coeff) * abs(signal[n])
    return envelope

def apply_noise_gate(signal, envelope, threshold_db, attenuation_db=-10, fs=44100, ramp_duration=0.5):
    # Перетворення ступеня приглушення з dB до лінійної шкали
    attenuation = 10 ** (attenuation_db / 20)

    # Ініціалізація маски Gain
    gain = np.ones_like(signal)

    # Встановлення параметрів атаки та релізу відповідно до ramp_duration
    attack_time = ramp_duration # Час підйому Gain до 1.0
    release_time = ramp_duration # Час спаду Gain до attenuation

    attack_coeff = np.exp(-1.0 / (fs * attack_time))
    release_coeff = np.exp(-1.0 / (fs * release_time))

    # Ініціалізація поточного Gain
    current_gain = 1.0

    # Обхід кожного семплу для плавної зміни Gain
    for n in range(len(signal)):
        # Перетворення оголошеного контуру в dB

```



```

env_db = 20 * np.log10(envelope[s] + 1e-10)

if env_db > threshold_db:
    # Якщо сигнал вище порогу, піднімаємо Gain до 1.0 плавно
    current_gain = attack_coeff * current_gain + (1 - attack_coeff) * 1.0
else:
    # Якщо сигнал нижче порогу, знижуємо Gain до attenuation плавно
    current_gain = release_coeff * current_gain + (1 - release_coeff) * attenuation

# Призначення поточного Gain до маски
gain[n] = current_gain

# Застосування Gain маски до сигналу
gated_signal = signal * gain

return gated_signal, gain

def main():
    input_file_path = 'processed.wav'
    output_file_path = 'noiseGate.wav'

    # Налаштування параметрів Noise Gate
    attack_time = 0.005 # Час атаки у секундах (5 мс)
    release_time = 0.05 # Час релізу у секундах (50 мс)
    attenuation_db = -15 # Знижуємо рівень шуму до -15 dB
    noise_duration = 0.5 # Використовуємо перші 0.5 секунд для визначення рівня шуму
    ramp_duration = 0.05 # Тривалість плавного переходу у секундах

    # Читання аудіофайлу
    signal, fs = sf.read(input_file_path)

    # Якщо аудіо стерео, перетворюємо на моно
    if len(signal.shape) > 1:
        signal = np.mean(signal, axis=1)

    # Перевірка на нечислові значення у вхідному сигналі
    if not np.isfinite(signal).all():
        print("Вхідний сигнал містить нечислові значення. Замінюємо їх на 0.")
        signal = np.nan_to_num(signal, nan=0.0, posinf=0.0, neginf=0.0)

    # Визначення шумового рівня з перших noise_duration секунд
    noise_samples = int(noise_duration * fs)
    noise_signal = signal[:noise_samples]
    noise_rms = np.sqrt(np.mean(noise_signal ** 2))
    noise_rms_db = 20 * np.log10(noise_rms + 1e-10) # Додаємо 1e-10 щоб уникнути log(0)

    # Визначення порогу гейтування: шумовий рівень + 10 dB
    threshold_db = noise_rms_db + 12
    print(f"Noise RMS: {noise_rms_db:.2f} dB")
    print(f"Threshold: {threshold_db:.2f} dB")

    # Обчислення огибающей контуру
    envelope = compute_envelope(signal, fs, attack_time, release_time)

    # Застосування Noise Gate з плавним приглушенням та поверненням
    gated_signal, gain_mask = apply_noise_gate(signal, envelope, threshold_db, attenuation_db, fs, ramp_duration)

    # Обчислення SNR до та після Noise Gate
    snr_original = calculate_snr(signal, noise_signal)
    snr_gated = calculate_snr(gated_signal, noise_signal)
    print(f"SNR оригінального сигналу: {snr_original:.2f} dB")
    print(f"SNR після Noise Gate: {snr_gated:.2f} dB")

    # Збереження обробленого сигналу
    sf.write(output_file_path, gated_signal, fs)
    print(f"Оброблений сигнал збережено у файл: {output_file_path}")

if __name__ == "__main__":
    main()

```

Основна обробка сигналу

```
import numpy as np
import scipy.signal as signal
import soundfile as sf
import matplotlib.pyplot as plt
import librosa
import librosa.display
import noisereduce as nr
from scipy.signal import find_peaks, butter, sosfilt
import pyloudnorm as pyln

# 1. Обчислення відношення сигнал/шум (SNR)
def calculate_snr(signal, noise):
    """
    Обчислює відношення сигнал/шум (SNR) у дБ.

    Parameters:
    signal (np.ndarray): Оброблений сигнал.
    noise (np.ndarray): Зразок шуму.

    Returns:
    float: Відношення сигнал/шум у дБ.
    """
    signal_power = np.mean(signal ** 2)
    noise_power = np.mean(noise ** 2)

    if noise_power == 0:
        print("Потужність шуму дорівнює нулю. Неможливо обчислити SNR.")
        return None

    snr = 10 * np.log10(signal_power / noise_power)
    return snr

def dynamic_multiband_compressor(signal_in, fs, bands=[(20, 250), (250, 4000), (4000, 20000)],
                                  threshold_db=-20, ratio=4.0, attack=0.01, release=0.1, makeup_gain=5,
                                  knee_db=10.0):
    """
    Застосовує багатосмугову компресію до сигналу з використанням Soft Knee.

    Parameters:
    signal_in (np.ndarray): Вхідний аудіосигнал.
    fs (int): Частота дискретизації.
    bands (list of tuples): Частотні смуги у Гц [(20, 250), (250, 4000), (4000, 20000)].
    threshold_db (float): Попіг компресії в дБ.
    ratio (float): Співвідношення компресії.
    attack (float): Час атаки в секундах.
    release (float): Час релізу в секундах.
    makeup_gain (float): Компенсація посилення після компресії в дБ.
    knee_db (float): Ширина переходу Soft Knee в дБ.

    Returns:
    np.ndarray: Компресований аудіосигнал.
    """

    def butter_bandpass(low, high, fs, order=4):
        sos = butter(order, [low, high], btype='band', fs=fs, output='sos')
        return sos

    def compressor(signal, threshold, ratio, attack, release, fs, knee):
        threshold_linear = 10**(threshold / 20)
        knee_linear = 10**(knee / 20)
        envelope = np.zeros_like(signal)
        gain = np.ones_like(signal)
        alpha_attack = np.exp(-1.0 / (attack * fs))
        alpha_release = np.exp(-1.0 / (release * fs))

        for n in range(1, len(signal)):
            rectified = np.abs(signal[n])
            if rectified > envelope[n-1]:
                envelope[n] = alpha_attack * envelope[n-1] + (1 - alpha_attack) * rectified
            else:
                envelope[n] = alpha_release * envelope[n-1] + (1 - alpha_release) * rectified

        # Перетворення оточення в дБ
        env_db = 20 * np.log10(envelope[n] + 1e-6)

        if env_db < (threshold_db - knee_db / 2):
            gain_db = 0.0
        elif (threshold_db - knee_db / 2) <= env_db < (threshold_db + knee_db / 2):
            # Soft Knee
            x = env_db - threshold_db
            gain_db = ((1 / ratio - 1) * (x + knee_db / 2)**2) / (2 * knee_db)
        else:
            gain_db = (threshold_db + (env_db - threshold_db) / ratio) - env_db

        gain_linear = 10**(gain_db / 20)
        gain[n] = gain_linear
```

```
def dynamic_multiband_compressor(signal_in, fs, bands=[(20, 250), (250, 4000), (4000, 20000)],
                                threshold_db=-20, ratio=4.0, attack=0.01, release=0.1, makeup_gain=5,
                                knee_db=10.0):
    """
```

Застосовує багатосмугову компресію до сигналу з використанням Soft Knee.

Parameters:

signal_in (np.ndarray): Вхідний аудіосигнал.

fs (int): Частота дискретизації.

bands (list of tuples): Частотні смуги у Гц [(20, 250), (250, 4000), (4000, 20000)].

threshold_db (float): Попір компресії в дБ.

ratio (float): Співвідношення компресії.

attack (float): Час атаки в секундах.

release (float): Час релізу в секундах.

makeup_gain (float): Компенсація посилення після компресії в дБ.

knee_db (float): Ширина переходу Soft Knee в дБ.

Returns:

np.ndarray: Компресований аудіосигнал.

"""

```
def butter_bandpass(low, high, fs, order=4):
```

```
    sos = butter(order, [low, high], btype='band', fs=fs, output='sos')
```

```
    return sos
```

```
def compressor(signal, threshold, ratio, attack, release, fs, knee):
```

```
    threshold_linear = 10**(threshold / 20)
```

```
    knee_linear = 10**(knee / 20)
```

```
    envelope = np.zeros_like(signal)
```

```
    gain = np.ones_like(signal)
```

```
    alpha_attack = np.exp(-1.0 / (attack * fs))
```

```
    alpha_release = np.exp(-1.0 / (release * fs))
```

```
    for n in range(1, len(signal)):
```

```
        rectified = np.abs(signal[n])
```

```
        if rectified > envelope[n-1]:
```

```
            envelope[n] = alpha_attack * envelope[n-1] + (1 - alpha_attack) * rectified
```

```
        else:
```

```
            envelope[n] = alpha_release * envelope[n-1] + (1 - alpha_release) * rectified
```

```
    # Перетворення оточення в дБ
```

```
    env_db = 20 * np.log10(envelope[n] + 1e-6)
```

```
    if env_db < (threshold_db - knee_db / 2):
```

```
        gain_db = 0.0
```

```
    elif (threshold_db - knee_db / 2) <= env_db < (threshold_db + knee_db / 2):
```

```
        # Soft Knee
```

```
        x = env_db - threshold_db
```

```
        gain_db = ((1 / ratio - 1) * (x + knee_db / 2)**2) / (2 * knee_db)
```

```
    else:
```

```
        gain_db = (threshold_db + (env_db - threshold_db) / ratio) - env_db
```

```
    gain_linear = 10**(gain_db / 20)
```

```
    gain[n] = gain_linear
```

```

for segment_start in range(start_freq, end_freq, segment_size):
    segment_end = min(segment_start + segment_size, end_freq)

    # Вибір частот у поточному проміжку
    mask = (fft_freqs >= segment_start) & (fft_freqs < segment_end)
    segment_freqs = fft_freqs[mask]
    segment_magnitudes = fft_magnitudes[mask]

    if len(segment_magnitudes) == 0:
        continue

    # Нормалізація для порівняння
    normalized_magnitudes = segment_magnitudes / np.max(segment_magnitudes)

    # Пошук піків у проміжку
    peaks, properties = find_peaks(normalized_magnitudes, height=peak_threshold)
    if len(peaks) == 0:
        continue

    # Знаходження найвищого піка
    peak_heights = properties['peak_heights']
    highest_peak_idx = np.argmax(peak_heights)
    highest_peak_freq = segment_freqs[peaks[highest_peak_idx]]
    highest_peak_mag = segment_magnitudes[peaks[highest_peak_idx]]

    # Вивід частот сибілянтів
    print(f"Detected sibilant frequency in {segment_start}-{segment_end} Hz: {highest_peak_freq/2f} Hz, Magn{
(highest_peak_mag/2f}")

    # Застосування фільтру до найвищого піка
    attenuated_signal = notch_filter_sos(attenuated_signal, fs, highest_peak_freq, bandwidth=bandwidth,
reduction_factor=reduction_factor)

    return attenuated_signal

# 4. Метод спектрального віднімання для шумозаглушення
def spectral_subtraction(signal_in, fs):
    """
    Використовує метод спектрального віднімання для зменшення шуму.
    """
    noise_sample = signal_in[int(0.5 * fs)]
    reduced_noise_signal = nr.reduce_noise(y=signal_in, sr=fs, y_noise=noise_sample, prop_decrease=0.93)
    return reduced_noise_signal

# 5. Застосування всіх функцій у комплексній обробці
def main():
    # Вкажіть шлях до вхідного та вихідного файлів
    input_file_path = '/noiseGate.wav' # Замініть на шлях до вашого вхідного файлу
    output_file_path = 'processedAudio.wav' # Замініть на бажаний шлях для вихідного файлу

    # Коефіцієнти налаштування
    filter_coeff = 1 # Коефіцієнт фільтрації (від 0 до 1)

```

```

for segment_start in range(start_freq, end_freq, segment_size):
    segment_end = min(segment_start + segment_size, end_freq)

    # Вибір частот у поточному проміжку
    mask = (fft_freqs >= segment_start) & (fft_freqs < segment_end)
    segment_freqs = fft_freqs[mask]
    segment_magnitudes = fft_magnitudes[mask]

    if len(segment_magnitudes) == 0:
        continue

    # Нормалізація для порівняння
    normalized_magnitudes = segment_magnitudes / np.max(segment_magnitudes)

    # Пошук піків у проміжку
    peaks, properties = find_peaks(normalized_magnitudes, height=peak_threshold)
    if len(peaks) == 0:
        continue

    # Знаходження найвищого піка
    peak_heights = properties['peak_heights']
    highest_peak_idx = np.argmax(peak_heights)
    highest_peak_freq = segment_freqs[peaks[highest_peak_idx]]
    highest_peak_mag = segment_magnitudes[peaks[highest_peak_idx]]

    # Вивід частот сибілянтів
    print(f"Detected sibilant frequency in {segment_start}-{segment_end} Hz: {highest_peak_freq/2f} Hz, Magn{
(highest_peak_mag/2f}")

    # Застосування фільтру до найвищого піка
    attenuated_signal = notch_filter_sos(attenuated_signal, fs, highest_peak_freq, bandwidth=bandwidth,
reduction_factor=reduction_factor)

    return attenuated_signal

# 4. Метод спектрального віднімання для шумозаглушення
def spectral_subtraction(signal_in, fs):
    """
    Використовує метод спектрального віднімання для зменшення шуму.
    """
    noise_sample = signal_in[int(0.5 * fs)]
    reduced_noise_signal = nr.reduce_noise(y=signal_in, sr=fs, y_noise=noise_sample, prop_decrease=0.93)
    return reduced_noise_signal

# 5. Застосування всіх функцій у комплексній обробці
def main():
    # Вкажіть шлях до вхідного та вихідного файлів
    input_file_path = '/noiseGate.wav' # Замініть на шлях до вашого вхідного файлу
    output_file_path = 'processedAudio.wav' # Замініть на бажаний шлях для вихідного файлу

    # Коефіцієнти налаштування
    filter_coeff = 1 # Коефіцієнт фільтрації (від 0 до 1)

```

```

# Читання аудіофайлу
data, samplerate = sf.read(input_file_path)

# Якщо аудіо стерео, перетворюємо на моно
if len(data.shape) > 1:
    data = np.mean(data, axis=1)

# Перевірка на нечислові значення у вхідному сигналі
if not np.isfinite(data).all():
    print("Вхідний сигнал містить нечислові значення. Замінюємо їх на 0.")
    data = np.nan_to_num(data, nan=0.0, posinf=0.0, neginf=0.0)

# 1. Обчислення SNR оригінального сигналу
noise_duration = 0.5 # Використовуємо перші 0.5 секунд для визначення рівня шуму
noise_samples = int(noise_duration * samplerate)
noise_signal = data[:noise_samples]
snr_original = calculate_snr(data, noise_signal)
print(f"SNR оригінального сигналу: {snr_original:.2f} дБ")

# 2. Мульти-Бенд Компресія (оновлена функція з Soft Knee)
signal_compressed = dynamic_multiband_compressor(
    signal_in=data,
    fs=samplerate,
    bands=[(20, 250), (250, 4000), (4000, 20000)],
    threshold_db=-20, # Попіг компресії в дБ
    ratio=4.0, # Співвідношення компресії
    attack=0.02, # Час атаки в секундах
    release=0.3, # Час релізу в секундах
    makeup_gain=5, # Компенсація посилення в дБ
    knee_db=10.0 # Ширина переходу Soft Knee в дБ
)

# Обчислення SNR після Компресії
snr_compressed = calculate_snr(signal_compressed, noise_signal)
print(f"SNR після Компресії: {snr_compressed:.2f} дБ")

# 3. De-Esser на компресованому сигналі
signal_de_essed = detect_and_compress_sibilants(
    data=signal_compressed,
    fs=samplerate,
    sibilant_range=(4000, 8000), # Діапазон сибілянтів
    peak_threshold=0.1, # Попіг для виявлення піків
    reduction_factor=0.7, # Коефіцієнт зменшення інтенсивності
    bandwidth=100 # Ширина смуги фільтра
)

# Обчислення SNR після De-Esser
snr_de_essed = calculate_snr(signal_de_essed, noise_signal)
print(f"SNR після De-Esser: {snr_de_essed:.2f} дБ")

# 4. Спектральне віднімання для шумозаглушення на де-ессованому сигналі
signal_denoised = spectral_subtraction(

```

```

    signal_in=signal_de_essed,
    fs=samplerate
)

# Обчислення SNR після Спектрального віднімання
snr_denoised = calculate_snr(signal_denoised, noise_signal)
print(f"SNR після Спектрального віднімання: {snr_denoised:.2f} дБ")

# 5. Кінцевий сигнал є комбінованим сигналом між обробленим та оригінальним сигналом
final_signal = filter_coeff * signal_denoised + (1 - filter_coeff) * data

# Обчислення SNR після комбінування
snr_final = calculate_snr(final_signal, noise_signal)
print(f"SNR після комбінування: {snr_final:.2f} дБ")

# Додаткова обробка: Нормалізація гучності до міжнародного стандарту
# Створюємо вимірювач гучності
meter = pyln.Meter(samplerate) # Створюємо вимірювач BS.1770

# Вимірюємо гучність фінального сигналу
loudness = meter.integrated_loudness(final_signal)

# Встановлюємо цільовий рівень гучності (наприклад, -23 LUFS)
target_loudness = -23.0

# Нормалізуємо гучність фінального сигналу
final_signal = pyln.normalize_loudness(final_signal, loudness, target_loudness)

# Обчислення SNR після нормалізації
snr_normalized = calculate_snr(final_signal, noise_signal)
print(f"SNR після нормалізації гучності: {snr_normalized:.2f} дБ")

# Запис результату з використанням кінцевого сигналу
sf.write(output_file_path, final_signal, samplerate)
print(f"Оброблений сигнал збережено у файл: {output_file_path}")

if __name__ == "__main__":
    main()

```

Кодування сигналу

```
import os
from vosk import Model, KaldiRecognizer
import json
import numpy as np
import matplotlib.pyplot as plt
import librosa
import librosa.display
import soundfile as sf

# Шляхи до файлів та моделей
AUDIO_FILE = '/processedAudio.wav'
MODEL_PATH = '/vosk-model-uk-v3-lgraph'
OUTPUT_FILE = '/voice_stego.wav'

# Перевірка наявності моделі Vosk
if not os.path.exists(MODEL_PATH):
    print("Завантажте модель 'vosk-model-uk-v3-lgraph' і розпакуйте її в поточну директорію.")
    exit(1)

# Функція для розрахунку SNR
def calculate_snr(signal, noise):
    """
    Розраховує відношення сигнал-шум (SNR).
    """
    # Обчислення потужності сигналу та шуму
    signal_power = np.mean(signal ** 2)
    noise_power = np.mean(noise ** 2)

    # Перевірка, щоб уникнути ділення на нуль
    if noise_power == 0:
        return np.inf # Безмежне SNR, оскільки шуму немає

    snr = 10 * np.log10(signal_power / noise_power)
    return snr

# Функція для розрахунку SDR
def calculate_sdr(original_signal, modified_signal):
    """
    Обчислює відношення сигнал-спотворення (SDR).
    """
    distortion = original_signal - modified_signal
    signal_power = np.mean(original_signal ** 2)
    distortion_power = np.mean(distortion ** 2)

    # Перевірка, щоб уникнути ділення на нуль
    if distortion_power == 0:
        return np.inf # Безмежне SDR, оскільки спотворень немає

    sdr = 10 * np.log10(signal_power / distortion_power)
    return sdr

# --- Розпізнавання мовлення та отримання бінарного повідомлення ---
```



```

audio_data, fs, librosa = librosa.load(AUDIO_FILE, sr=None)

# Перетворення аудіо у байти для Voak
audio_bytes = (audio_data * 32767).astype(np.int16).tobytes()

# Завантаження моделі та розпізнавання
model = Model(MODEL_PATH)
rec = KaldiRecognizer(model, fs, librosa)

rec.AcceptWaveform(audio_bytes)
result = rec.Result()
result_json = json.loads(result)
result_text = result_json.get('text', "")

# Вивід розпізнаного тексту
print("Розпізнаний текст:")
print(result_text.strip())

# Кодування тексту в бінарний формат UTF-8
binary_text = "".join(format(byte, '08b') for byte in result_text.strip().encode('utf-8'))

print("\nТекст у бінарному форматі UTF-8:")
print(binary_text)

# 3. Рівень гучності та середнє значення амплітуди початкового сигналу
volume_original = np.sqrt(np.mean(audio_data ** 2))
mean_amplitude_original = np.mean(np.abs(audio_data))

# **Розрахунок SNR початкового сигналу**

# Припустимо, що перші 0.5 секунд сигналу містять лише шум
noise_duration = 0.5 # тривалість шуму в секундах
noise_samples = int(noise_duration * fs, librosa)

# Перевіримо, чи достатньо довжини сигналу для виділення шуму
if noise_samples > len(audio_data):
    raise ValueError("Сигнал занадто короткий для виділення шуму.")

# Виділяємо ділянку шуму
noise_sample = audio_data[:noise_samples]

# Обчислимо SNR для початкового сигналу
snr_original = calculate_snr(audio_data, noise_sample)

# **Вбудовування повідомлення методом LSB**

# Конвертуємо аудіосигнал у 16-бітний формат для вбудовування
samples = (audio_data * 32767).astype(np.int16)

# Перевірка, чи достатньо семплів для вбудовування повідомлення
num_bits = len(binary_text)

if num_bits > len(samples):
    raise ValueError("Повідомлення занадто велике для вбудовування в даний аудіофайл.")

# Вбудовування повідомлення методом LSB
modified_samples = np.copy(samples)
for i in range(num_bits):
    # Заміна найменш значущого біта семплу на біт повідомлення
    modified_samples[i] = (modified_samples[i] & ~1) | int(binary_text[i])

# Нормалізація модифікованих семплів для запису
audio_modified = modified_samples.astype(np.float32) / 32767.0

# Збереження модифікованого аудіофайлу
sf.write(OUTPUT_FILE, audio_modified, fs, librosa)

print("\nМодифікований аудіофайл з вбудованим повідомленням збережено як {OUTPUT_FILE}")

# 3. Рівень гучності та середнє значення амплітуди модифікованого сигналу
volume_modified = np.sqrt(np.mean(audio_modified_loaded ** 2))
mean_amplitude_modified = np.mean(np.abs(audio_modified_loaded))

# Обчислимо SNR для модифікованого сигналу, використовуючи той самий зразок шуму
snr_modified = calculate_snr(audio_modified_loaded, noise_sample)

# **Розрахунок SDR після вбудовування**

sdr_after_embedding = calculate_sdr(audio_data, audio_modified_loaded)

```