

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
“КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМ. ІГОРЯ
СІКОРСЬКОГО”

Факультет електроніки

Кафедра електронної інженерії

“На правах рукопису”
УДК 621.3

“До захисту допущено”

Завідувач кафедри

В. І. Тимофєєв

“ ” _____ 20__ р.

Магістерська дисертація

за спеціальністю 176 Мікро- та наносистемна техніка

на тему «Аналіз радіочастотного спектру для розпізнавання типів
сигналів»

Виконав студент 2 курсу, групи ДМ-31мн

Гончаров Серафим Сергійович

(прізвище, ім'я, по батькові)

_____ (підпис)

Науковий керівник к.т.н., доц. Вунтесмері Юрій Володимирович

(посада, вчене звання, науковий ступінь, прізвище та ініціали)

_____ (підпис)

Рецензент проф.каф. ЕПС дтн., проф. Мельник Ігор Віталійович

(посада, вчене звання, науковий ступінь, прізвище та ініціали)

_____ (підпис)

Засвідчую, що у цій дипломній роботі
немає запозичень з праць інших авторів без
відповідних посилань.

Студент _____

(підпис)

Київ – 2025 року

Факультет електроніки
Кафедра електронної інженерії
Освітній рівень «магістр»
за спеціальністю 176 Мікро- та наносистемна техніка

ЗАТВЕРДЖУЮ

Завідувач кафедри

_____ В. І. Тимофєєв

“ ___ ” _____ 20__ р.

З А В Д А Н Н Я
НА ДИПЛОМНУ РОБОТУ СТУДЕНТУ

Гончаров Серафим Сергійович

1. Тема роботи “ Аналіз радіо спектру для розпізнавання типів передавачів”, керівник роботи: доц, к.т.н. Вунтесмері Юрій Володимирович,
затверджені наказом по університету від “25” березня 2025 року № 1233-с
2. Строк подання студентом роботи:
3. Вихідні дані до роботи: літературні джерела щодо програмно-визначуваних радіосистем, теорії сигналів
4. Зміст дипломної роботи (перелік питань, які потрібно розробити): дослідження алгоритмів для аналізу спектрограм у реальному масштабі часу
5. Перелік графічного (ілюстративного) матеріалу (із зазначенням обов’язкових креслень, плакатів, презентацій тощо): презентація MS Power Point.

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв

7. Дата видачі завдання _____

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів виконання дипломної роботи	Строк виконання етапів роботи	Примітка
1	Проведення літературного огляду щодо використання програмно-визначуваних радіосистем	17.04.2025 - 30.04.2025	
2	Проведення літературного огляду щодо побудови, та обробки даних спектрограм	30.04.2025 - 15.05.2025	
3	Побудова узагальненого алгоритма детектору	15.05.2025 - 29.05.2025	
4	Відлагодження програми для побудови спектрограми	29.05.2025 - 12.06.2025	
5	Оформлення дипломної роботи	01.06.2025-12.06.2025	
6	Підготовка до захисту	12.06.2025-20.06.2025	

Студент _____ Гончаров С. С. _____

Керівник роботи _____ Вунтесмері Ю.В. _____

РЕФЕРАТ

SDR, АНАЛІЗАТОР СПЕКТРУ

Магістерська робота містить 87 сторінок, 17 рисунків, 2 таблиці, 52 джерела та 1 додаток.

Метою цього дослідження є аналіз і вдосконалення алгоритмів для ефективного аналізу спектрограм у реальному часі, що дозволить вирішувати актуальні завдання у галузі радіочастотного моніторингу та аналізу сигналів. У першому розділі представлені теоретичні основи спектрального аналізу, його історія розвитку, принципи роботи спектроаналізаторів, а також основні труднощі, які виникають при аналізі спектрограм у реальному часі. Другий розділ зосереджений на методах побудови та обробки спектрограм. Описано алгоритми та техніки, що використовуються для отримання спектрограм з різних типів сигналів, з особливим акцентом на методи швидкого перетворення Фур'є (FFT), параметри розгортки та постобробку даних. Третій розділ містить огляд спектральних характеристик різних типів модуляцій і методи їх класифікації на основі спектрограм. У четвертому розділі розглядаються практичні аспекти застосування алгоритмів для аналізу спектрограм з використанням технології Software Defined Radio (SDR), зокрема акцент на бібліотеці SoapySDR, яка спрощує інтеграцію з різними SDR-платформами. Проведено порівняння популярних SDR платформ, докладно описано використання HackRF One в експериментальних дослідженнях. Окремо розглянуті методи розпізнавання типів модуляції за допомогою спектрограм, а також запропоновано метод адаптивного порогу на основі медіанного рівня шуму для виявлення частотно-

модульованих сигналів (FSK). Розроблено програмне забезпечення та проведено чисельні експерименти.

ABSTRACT

SDR, SPECTRUM ANALYZER

The master's thesis contains 87 pages, 17 figures, 2 tables, 52 sources, and 1 appendix.

The aim of this research is to analyze and improve algorithms for high-performance real-time spectrogram analysis, which will address current challenges in the field of radio frequency monitoring and signal analysis. The first chapter presents the theoretical foundations of spectral analysis, its historical development, the principles of operation of spectrum analyzers, as well as the main challenges faced when analyzing spectrograms in real time. The second chapter focuses on methods for constructing and processing spectrograms. It describes algorithms and techniques used to obtain spectrograms from various types of signals, with particular attention given to Fast Fourier Transform (FFT) methods, sweep parameters, and post-processing of data. The third chapter provides an overview of the spectral characteristics of different modulation types and methods for their classification based on spectrograms. The fourth chapter discusses the practical application of algorithms for spectrogram analysis using Software Defined Radio (SDR) technology, with a focus on the SoapySDR library, which simplifies integration with various SDR platforms. A comparison of popular SDR platforms is presented, with a detailed description of HackRF One, which was used in experimental research. Methods for recognizing modulation types based on spectrograms are analyzed, and an adaptive threshold detection method based on the median noise level is proposed for detecting frequency-modulated signals

(FSK). Software has been developed, and numerical experiments were conducted

ЗМІСТ

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ	
ВСТУП	
1 ЗАГАЛЬНІ ВІДОМОСТІ ПРО АНАЛІЗ СПЕКТРУ	
1.1 Поняття спектрограми як функції спектру у часі.....	
1.2 Історія розвитку спектрального аналізу	
1.3 Архітектура цифрового спектроаналізатора	
1.4 Автоматизація сканування частотного діапазону	
2 ПОБУДОВА ТА ОБРОБКА ДАНИХ СПЕКТРОГРАМИ	
2.1 Отримання даних із SDR-приймача (HackRF)	
2.2 Нормалізація та попередня обробка сигналу (AGC, обмеження по довжині).....	
2.3 Перетворення Фур'є та згладжування спектру (FFT, Savitzky-Golay)	
2.4 Побудова спектрограм на GPU (STFT, CuPy, cuSignal).....	
2.5 Використання віконних фільтрів (FIR, Nuttall, Bandpass)	
2.6 Детекція піків та оцінка ширини смуги (CuPy find_peaks).....	
2.7 Візуалізація результатів (Matplotlib).....	
3 ВИКОРИСТАННЯ СПЕКТРОГРАМ ДЛЯ ВИЯВЛЕННЯ ПЕРЕДАВАЧІВ.....	
3.1 Типи модуляцій і їхній спектральний вигляд.....	
3.2 Визначення основної частоти та гармонік.....	
3.3 Параметри спектрів FM, PSK, OFDM, ППРЧ.....	
3.4 Алгоритм автоматичного визначення типу передавача.....	
4 ПРОГРАМНА РЕАЛІЗАЦІЯ ТА ЕКСПЕРИМЕНТАЛЬНЕ	

ДОСЛІДЖЕННЯ	
4.1 Основи технології SDR	
4.2 Використання бібліотек osmosdr, gnuradio, cuSignal	
4.3 Особливості пристрою HackRF	
4.4 Результати спектрального сканування	
5 МЕТОДИ ВИЗНАЧЕННЯ ТИПІВ МОДУЛЯЦІЇ	
5.1 Класичні методи класифікації цифрових модуляцій на основі спектрограм	
5.2 Реалізований метод	
ВИСНОВКИ	
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	
ДОДАТОК А (Фрагменти коду)	

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

CWT - Continuous Wavelet Transform (Неперервне вейвлет-перетворення)

DSP - Digital Signal Processing (Цифрова обробка сигналів)

DFT - Discrete Fourier Transform (Дискретне перетворення Фур'є)

FFT - Fast Fourier Transform (Швидке перетворення Фур'є)

ПЧ - Проміжна Частота (Intermediate Frequency)

STFT - Short-Time Fourier Transform (Короткочасне перетворення Фур'є)

ВСТУП

Актуальною є проблема виявлення каналів передачі на невідомих частотах, що стає все більш важливим у контексті розвитку радіозв'язку та моніторингу радіоспектра. У сучасних умовах, коли спектр радіочастот стає все більш насиченим, ефективне виявлення активних каналів передачі є важливим для запобігання завад та оптимізації використання радіочастотного ресурсу. Системи, що здатні виконувати таку задачу, повинні працювати в реальному часі, забезпечуючи точність і швидкість виявлення сигналів при обмежених обчислювальних ресурсах.

Предметом дослідження є аналіз змінного у часі спектру радіовипромінювання.

Об'єктом дослідження є радіосигнали у широкому діапазоні частот.

Метою дослідження є виявлення та ідентифікація радіоканалів за допомогою аналізу змінного у часі спектру радіовипромінювання.

Завданнями дослідження є

- дослідження методів та засобів вимірювання та зберігання змінних у часі спектрів радіовипромінювання;
- розробка лабораторного макету для аналізу спектрів
- дослідження методів аналізу спектру з метою виявлення частот потенційних радіоканалів;
- дослідження методів аналізу спектрограм радіоканалів для ідентифікації типу модуляції та\або передавача;
- розробка алгоритму аналізу спектрограм для ідентифікації радіоканалів, придатного для застосування на обчислювальних платформах малої потужності.

1 ЗАГАЛЬНІ ВІДОМОСТІ ПРО АНАЛІЗ СПЕКТРУ

1.1 Поняття спектрограми як функції спектру у часі

Спектрограми є важливими інструментами для аналізу та обробки різноманітних сигналів, які забезпечують візуалізацію того, як частотний склад змінюється з часом. Вони дозволяють отримати уявлення про те, як змінюються частотні компоненти сигналу, що має велике значення для багатьох наукових і технічних галузей, включаючи обробку мовних сигналів, музичний аналіз, вивчення радіолокаційних сигналів і багато іншого. Розглянувши основні характеристики спектрограм, а також їхні складові та області застосування, можна зрозуміти, чому вони є незамінним інструментом для вивчення та аналізу сигналів, що змінюються з часом. Спектрограма може бути визначена як зображення, яке демонструє, як частотні складові сигналу змінюються впродовж його існування. У порівнянні з традиційними методами представлення сигналів у частотній області, де відображаються лише частоти в певний момент часу, спектрограми надають більш динамічне і гнучке представлення частотного змісту сигналу протягом певного проміжку часу. Цей динамічний аспект робить спектрограми надзвичайно корисними для аналізу сигналів, які змінюються з часом, де важливими є не тільки частоти, а й часові зміни, а також те, як ці зміни взаємодіють з іншими компонентами сигналу[14]. Це в особливості актуально для задач, таких як розпізнавання мовлення, акустичний аналіз, дослідження вібрацій та багатьох інших областей, де частотні компоненти і їхня зміна в часі є важливими. Для кращого розуміння концепції спектрограми, можна звернутися до прикладу музичного твору. Спектрограма цього твору дозволяє побачити, які ноти та звуки містяться в записі, а також те, як ці ноти змінюються в часі. Високі звуки, як правило, розташовуються вище на спектрограмі, в той час як низькі звуки зображуються нижче. Крім того,

гучність звуку також має важливе значення: чим гучніше звучить певна нота або частота, тим ширша її смуга на спектрограмі. Це дозволяє візуально відстежити динаміку звукових змін, що робить спектрограму важливим інструментом для музичних досліджень, наприклад, при створенні аналізу музичних творів або їх інтерпретації. Завдяки можливості наочно демонструвати, як частотні компоненти сигналу змінюються з часом, спектрограми стають ефективними інструментами в аналізі широкого спектра сигналів. Вони дозволяють не тільки з'ясувати, які частоти містяться в сигналі, але й як ці частоти змінюються, що в свою чергу дає глибше розуміння властивостей сигналу і дозволяє вирішувати складні завдання, пов'язані з його обробкою[16]. Спектрограми знайшли широке застосування в різних наукових і технічних галузях, оскільки вони є потужним інструментом для візуалізації та аналізу сигналів. Спектрограми дозволяють досліджувати різноманітні характеристики мовного сигналу, такі як інтонація, наголос і артикуляція. Наприклад, спектрограма мовного сигналу візуалізує основні частоти мовця, а також форманти — резонансні частоти ротової порожнини, що визначають звучання голосних звуків (Рис. 1.1).

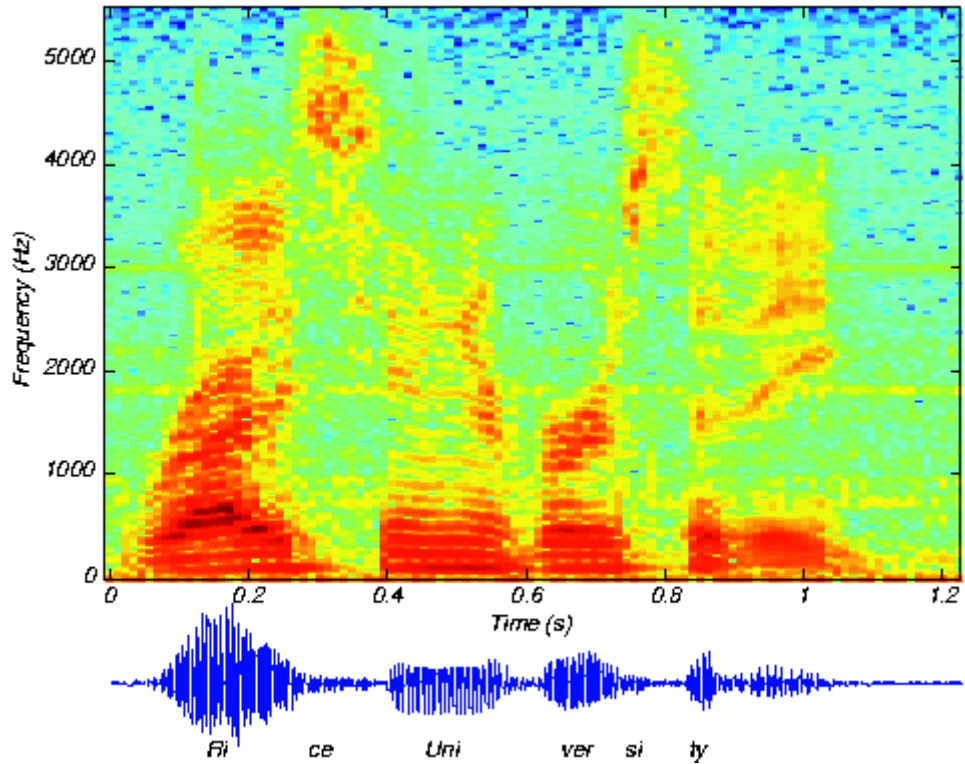


Рисунок 1.1 - Спектрограма мовного сигналу

Спектрограми допомагають комп'ютерам "розуміти" мову, оскільки вони дозволяють розпізнавати та класифікувати різні мовні звуки та їхні варіації в часі, що є важливим для автоматичних систем розпізнавання мовлення. Аналіз спектрограм музичних записів дозволяє вивчати структуру мелодії та гармонії. Спектрограма музичного твору надає детальну інформацію про ноти та тони, що складають композицію, а також про динаміку гучності, яка змінюється з часом (Рис. 1.2).

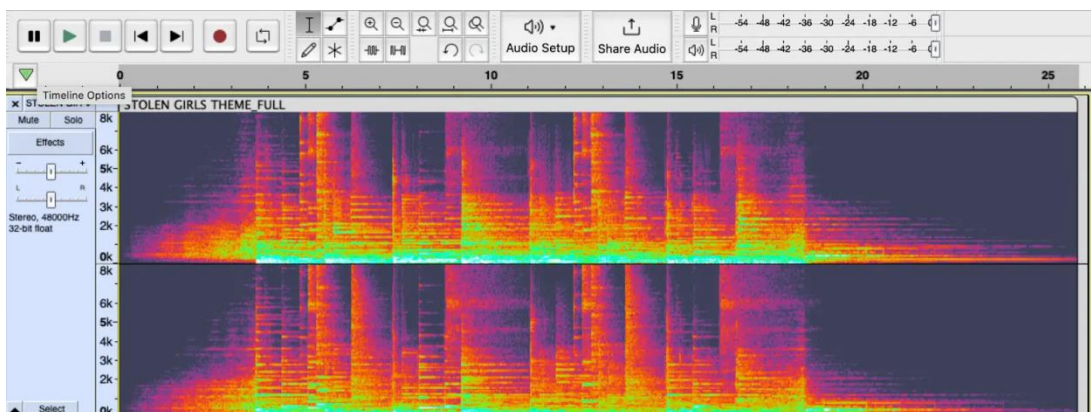


Рисунок 1.2 - Приклади спектрограм (музика)

Використовуються для виявлення об'єктів під водою, таких як човни чи підводні човни. Спектрограма сигналу сонара відображає відбиті звукові хвилі, які дозволяють визначити місцезнаходження та розмір об'єкта під водою (Рис. 1.3).

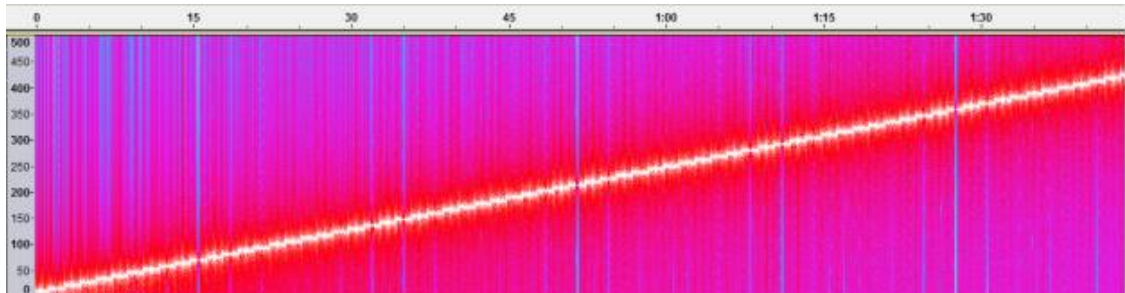


Рисунок 1.3 - Приклади спектрограм (сонор)

Спектрограми допомагають у вивченні сейсмічних хвиль, виявленні типу хвиль (Р-хвиля, S-хвиля, L-хвиля), визначенні глибини землетрусу та оцінці його магнітуди. У медичних дослідженнях спектрограми використовуються для діагностики різних захворювань, таких як порушення серцевої діяльності або голосові розлади. Наприклад, спектрограма серцевого шуму допомагає лікарям визначити тип та причину шуму, а також оцінити його тяжкість. Спектрограма складається з двох основних осей:

- Часова вісь (t): Показує, як сигнал змінюється з часом.
- Частотна вісь (f): Показує, які частоти присутні в сигналі в кожен момент часу.

Яскравість або колір спектрограми вказує на інтенсивність частоти в конкретний момент часу. Чим яскравіша область спектрограми, тим вища інтенсивність сигналу на цій частоті в даний момент часу.

Спектрограми допомагають візуалізувати різні характеристики сигналу:

1. Частотний вміст: які частотні компоненти містить сигнал.

2. Зміни частоти: як частота сигналу змінюється з часом (наприклад, для аналізу інтонації мови).
3. Нестаціонарність: як інтенсивність сигналу змінюється на різних частотах з часом.

Основні методи для отримання спектрограм включають:

- Перетворення Фур'є - розкладає сигнал на спектральні компоненти.
- Веївлет-перетворення використовує локалізовані в часі функції (веївлети) для аналізу сигналу.
- Короткочасне перетворення Фур'є розбиває сигнал на короткі інтервали часу та застосовує перетворення Фур'є до кожного інтервалу[21].

Спектрограми мають кілька корисних властивостей:

1. Візуалізація спектральної еволюції дозволяють побачити, як змінюється спектральний склад сигналу з часом.
2. Виявлення нестаціонарних сигналів допомагають аналізувати сигнали, спектр яких змінюється з часом.
3. Аналіз гармонійних компонентів використовуються для виявлення та аналізу гармонійних складових сигналу.
4. Спектрограми можуть бути використані для синтезу сигналів з заданим спектральним вмістом.

Таблиця 1.1 - Характеристики сигналу, які можна візуалізувати за допомогою спектрограм

Характеристика	Опис	Шкала вимірювання	Приклад
Частотний вміст	Які частотні компоненти містить сигнал.	Герц (Гц)	Спектрограма мови показує, що мова містить широкий спектр частот, з акцентом на нижчі частоти (100-250 Гц) для голосних і вищі частоти (2000-4000 Гц) для приголосних.
Зміни частоти	Як частота сигналу змінюється з часом.	Герц на секунду (Гц/с)	Спектрограма співу показує, як співак змінює висоту звуку протягом пісні (наприклад, від 200 Гц до 1000 Гц).
Нестационарність	Як інтенсивність сигналу на різних частотах змінюється з часом.	Децибели (дБ)	Спектрограма удару по барабану показує, що удар генерує короткий імпульс енергії на широкому спектрі частот (-20 дБ до 0 дБ), який потім швидко згасає (-30 дБ до -40 дБ).

Спектрограми є потужним інструментом для аналізу та візуалізації різноманітних сигналів. Вони широко використовуються в різних наукових і технічних областях, таких як музика, мовлення, сейсмологія, гідролокація та медицина. З їхньою допомогою можна ефективно

аналізувати сигнали, виявляти їхні характеристики та зміни, що є важливим для глибокого розуміння складних явищ.

1.2 Історія розвитку спектрального аналізу

Історія аналізу спектру в часі є значним етапом у розвитку наукових досліджень та технологій, які використовуються для вивчення та обробки сигналів. Цей процес розпочався з ранніх наукових відкриттів і продовжувався через століття, з кожним новим етапом відкриваючи нові можливості для точного аналізу сигналів. Перші фундаментальні експерименти, які мали великий вплив на розуміння хвильових явищ, були проведені Томасом Юнгом у 1801 році. У своїх дослідженнях дифракції та інтерференції світла він продемонстрував хвильову природу цього явища, що стало основою для подальших відкриттів у спектральному аналізі. Його роботи вивели на новий рівень розуміння, що стало важливим для аналізу не тільки світла, а й інших хвильових сигналів, таких як звукові хвилі. Значний розвиток в аналізі сигналів відбувся у 1822 році, коли Жозеф Фур'є представив свою працю «Аналітична теорія тепла». У цьому дослідженні він розвинув ідею перетворення Фур'є, яке дозволяє розкласти складні періодичні сигнали на набір простих синусоїдальних компонентів з різними частотами та амплітудами. Завдяки цьому підходу стало можливим представлення будь-якої складної функції у вигляді суми простих тригонометричних функцій. Цей принцип став основою для спектрального аналізу, що дозволило досліджувати частотний склад сигналів і стало важливим кроком на шляху до розвитку спектрограм. З поширенням радіотехніки та розвитком телекомунікаційних технологій у 1920-х роках виникла необхідність у більш точному аналізі електричних сигналів. У цей період спектральний аналіз почав активно застосовуватися для вивчення частотного складу радіосигналів. Інженери використовували електронні осцилографи для візуалізації сигналів, що дозволяло їм

ефективно налаштувати радіопередавачі та приймачі, зосереджуючись на точному налаштуванні частотних характеристик сигналу.

Прорив у спектральному аналізі стався в 1946 році завдяки Денісу Габору, угорсько-британському фізику, лауреату Нобелівської премії, який розробив теорію короткочасного перетворення Фур'є (STFT). Це дозволило здійснювати одночасний аналіз як в часі, так і в частоті, розкладаючи сигнали на короткі сегменти та застосовуючи перетворення Фур'є до кожного з них. Такий підхід став важливим для аналізу сигналів, що змінюються з часом, таких як звукові сигнали, музика та мовлення. Ще однією важливою віхою у розвитку спектрального аналізу стало створення алгоритму швидкого перетворення Фур'є (FFT) у 1965 році. Джеймс Кулі та Джон Тьюкі запропонували метод, що дозволив значно знизити обчислювальні витрати, зменшивши складність перетворення з $O(N^2)$ до $O(N \log N)$. Це досягнення відкрило можливість виконання спектрального аналізу в реальному часі на цифрових комп'ютерах, що стало революційним у таких сферах, як телекомунікації, медична діагностика та аудіоінженерія. У 1970-х роках, з розвитком цифрових обчислювальних технологій, широко впроваджувалась цифрова обробка сигналів (DSP). Це дозволило реалізувати складні алгоритми для обробки, фільтрації та модифікації сигналів у цифровому вигляді. Спектрограми стали основним інструментом для аналізу звукових сигналів, зокрема у галузях розпізнавання мовлення та музики. Спеціалізовані мікропроцесори для DSP забезпечили високу швидкість обробки, що дозволило виконувати спектральний аналіз у реальному часі, навіть на вбудованих системах.

У 1980-х роках з'явилося вейвлет-перетворення, яке стало альтернативою традиційному перетворенню Фур'є для аналізу сигналів з нестабільними частотними компонентами. Вейвлети використовують функції з обмеженою підтримкою, які можуть бути масштабовані, що дає змогу досягти високої роздільної здатності як у часі, так і в частоті.[25] Це

відкриття стало особливо корисним для аналізу складних сигналів, таких як електроенцефалограми (ЕЕГ) або звукові сигнали, що мають швидко змінювані частотні компоненти. У наступні десятиліття, з розвитком машинного навчання та штучного інтелекту, спектрограми набули нового застосування в області розпізнавання образів. Нейронні мережі та інші методи машинного навчання використовують спектрограми для аналізу звукових даних, що дозволяє класифікувати звукові сигнали і застосовувати ці технології в таких галузях, як автоматичне розпізнавання мовлення або біомедичні дослідження.

З 2010-х років, завдяки розвитку високопродуктивних обчислень та розподілених систем, стало можливим обробляти великі обсяги даних та виконувати складний спектральний аналіз у реальному часі. Хмарні технології та паралельні обчислювальні алгоритми значно прискорили обробку сигналів, що стало важливим для таких застосувань, як моніторинг здоров'я пацієнтів, а також для аналізу даних у наукових дослідженнях, зокрема астрономічних. Таким чином, історія спектрального аналізу і спектрограм є не тільки історією наукових відкриттів, а й історією розвитку технологій, що дозволили вивчати та обробляти сигнали, роблячи можливим виявлення нових закономірностей і відкриттів у найрізноманітніших наукових і технічних сферах.

1.3 Архітектура цифрового спектроаналізатора

Спектроаналізатор є складним і високотехнологічним приладом, який поєднує в собі різні методи та технології для точного вимірювання частотного спектру сигналів. Від першого етапу обробки сигналу до кінцевого виведення результату, кожен компонент спектроаналізатора відіграє важливу роль у забезпеченні точності та надійності аналізу. Сучасні спектроаналізатори широко застосовуються в багатьох галузях, таких як телекомунікації, радіоелектроніка, біомедичні дослідження, а

також аудіоінженерія, даючи змогу глибше розуміти частотні характеристики сигналів.

Узагальнена конструкція спектроаналізатора є важливою для розуміння принципу його роботи, зокрема того, як цей прилад розкладає складний сигнал на його частотні компоненти. Детальніше розглянемо основні блоки, які складають структуру спектроаналізатора. На самому початку сигнал проходить через вхідний блок, що має кілька важливих функцій. Перша з них — це атенюатор, який регулює рівень вхідного сигналу, щоб уникнути перевантаження подальших етапів обробки. Потім сигнал надходить до вхідного підсилювача, який збільшує амплітуду слабких сигналів до потрібного рівня, що дозволяє ефективно працювати наступним етапам обробки. Далі йде вхідний фільтр, який проводить попередній відбір частот, виділяючи лише потрібний діапазон і усуваючи непотрібні, позасмугові сигнали, що можуть спотворити результати аналізу. Наступним важливим етапом є блок змішувача. Цей блок займається перетворенням частоти вхідного сигналу. У ньому є гетеродин, який генерує стабільний сигнал з відомою частотою, а також змішувач, що комбінує сигнал гетеродина з вхідним сигналом. Після цього утворюється сигнал проміжної частоти (ПЧ), який має фіксовану частоту та є зручнішим для подальшої обробки. Обробка сигналу продовжується в блоці проміжної частоти (ПЧ). Цей блок включає в себе підсилювач ПЧ, що підвищує амплітуду сигналу до необхідного рівня, а також фільтр ПЧ, який виділяє вузьку смугу частот навколо ПЧ і придушує залишкові позасмугові сигнали. Останнім компонентом цього блоку є детектор, який здійснює виділення інформаційного сигналу з модуляції сигналу ПЧ. Тип детектора залежить від виду модуляції, наприклад, для амплітудної модуляції використовується амплітудний детектор, а для частотної модуляції — частотний детектор.

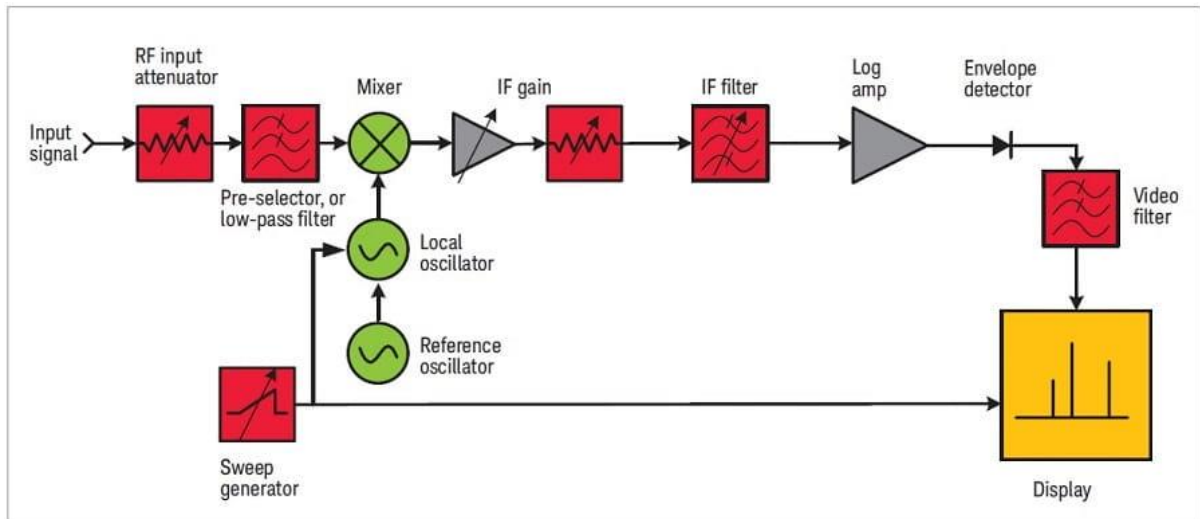


Рисунок 1.4 – Типова схема аналізатора спектру з розгорткою[1]

Останнім етапом обробки є блок відображення, який відповідає за переведення обробленого сигналу у форму, зручну для сприйняття користувачем. В цьому блоці розташований аналогово-цифровий перетворювач (АЦП), що перетворює аналоговий сигнал ПЧ на цифровий формат. Це дозволяє проводити подальшу обробку сигналу за допомогою цифрового процесора сигналів (ЦПС)[1]. ЦПС виконує спектральний аналіз цифрового сигналу, визначаючи амплітуди частотних компонентів. Результати цього аналізу можуть відображатися на дисплеї у вигляді графіків спектра або таблиць значень амплітуд. Сучасні спектроаналізатори здебільшого є цифровими приладами, де більшість функцій виконується цифровими процесорами сигналів. Вони забезпечують широкий діапазон частот, високу роздільну здатність, великий динамічний діапазон і різноманітні функції для аналізу сигналів, що робить їх незамінними інструментами в наукових і технічних сферах. Деякі моделі спектроаналізаторів також оснащені додатковими функціями, такими як генератори сигналів, векторні та логічні аналізатори, що значно розширює їх функціональність і робить їх незамінними в роботі інженерів і науковців[1]. Завдяки своїм численним можливостям, сучасні спектроаналізатори займають важливе місце в дослідженнях, де точний

аналіз частотних характеристик сигналів є критичним для успішного виконання завдань.

TinySA — це мініатюрний спектроаналізатор, який забезпечує точний спектральний аналіз радіочастотних сигналів у широкому діапазоні частот, від 100 кГц до 960 МГц. Завдяки своїм невеликим розмірам і можливості живлення від акумулятора, TinySA є відмінним варіантом для використання як в лабораторних умовах, так і в польових. Прилад можна зручно носити з собою, що робить його ідеальним інструментом для радіоелектронних вимірювань в будь-яких умовах[2].

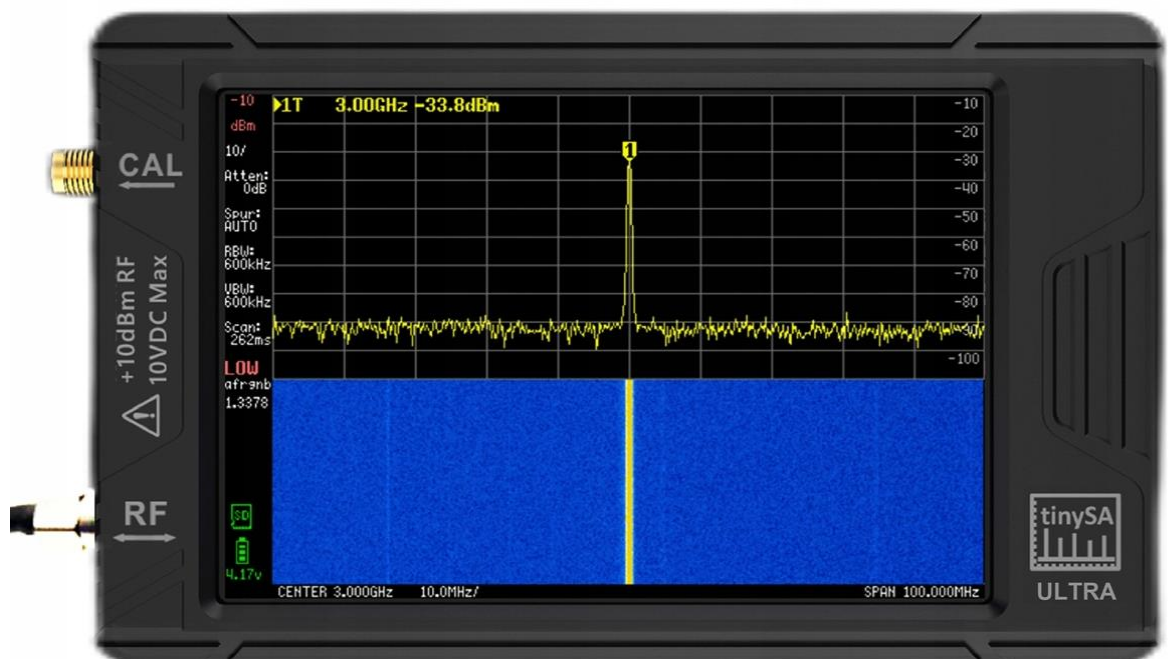


Рисунок 1.5 – TinySA

Основні характеристики:

- Діапазон частот: 100 кГц — 960 МГц.
- Динамічний діапазон: до 120 дБ.
- Роздільна здатність: до 1 Гц.
- Тип екрану: кольоровий TFT-дисплей 320x240 пікселів для відображення результатів.
- Живлення: акумулятор 7,4 В, можливість заряджання через USB.

- Розміри: компактний розмір, що дозволяє носити прилад в кишені.

TinySA має простий і інтуїтивно зрозумілий інтерфейс, що дозволяє користувачам швидко налаштовувати та отримувати спектральний аналіз сигналів. Це дуже корисний інструмент для тестування та вимірювання радіочастотних сигналів в радіоелектроніці, комунікаціях, а також для навчальних цілей. На вході спектроаналізатора TinySA є роз'єм SMA для підключення антени або інших джерел сигналу. Для захисту приймача та запобігання перевантаженням використовується настроюваний аттенюатор з діапазоном 0-31 дБ, що дозволяє регулювати рівень вхідного сигналу[2].

TinySA оснащений фільтром низьких частот з відсіченням на 350 МГц для усунення небажаних компонент сигналу і псевдонімів, що можуть спотворювати результати вимірювання. Для перетворення сигналу на проміжну частоту (ПЧ), TinySA використовує гетеродин і змішувач, що дозволяє змінювати частоту сигналу для зручності подальшої обробки. У блоці RX сигнал проходить через другий гетеродин і мікшер, які перетворюють високу ПЧ в низьку ПЧ на 870 кГц. Це дозволяє проводити точний аналіз частот у потрібному діапазоні[2]. TinySA оснащений детектором потужності, що має динамічний діапазон 120 дБ, що дозволяє здійснювати точні вимірювання рівнів сигналу в широкому діапазоні. TinySA дозволяє перевіряти частотні характеристики радіообладнання, проводити вимірювання потужності сигналу і оцінювати ефективність роботи передавачів і приймачів. Прилад можна використовувати для вимірювання характеристик сигналів, таких як амплітуда, частота та модуляція. Завдяки своїм компактным розмірам і живленню від акумулятора, TinySA є відмінним інструментом для тестування обладнання в польових умовах, де важливо мати легкий і зручний спектроаналізатор.

1.4 Автоматизація сканування частотного діапазону

Управління вимірюванням спектрограми є важливою складовою спектрального аналізу, що вимагає детального розуміння теоретичних аспектів та практичного застосування, а також умінь налаштовувати параметри пристрою для досягнення оптимальних результатів. Це включає не тільки вибір правильних параметрів для вимірювання, але й вміння інтерпретувати отримані дані з огляду на особливості сигналу та цілі аналізу. Центральна частота визначає точку, навколо якої формується спектрограма. Її вибір залежить від знань про досліджуваний сигнал і його спектральні характеристики[3]. Діапазон частот (Span) визначає ширину частотного діапазону, що відображається на спектрограмі. Вибір відповідного діапазону дозволяє зосередити увагу на певних частинах спектра, що дозволяє точніше виявити деталі, які можуть бути не помітні при використанні занадто широкого діапазону.

Роздільна здатність по частоті (Resolution Bandwidth, RBW) є важливим параметром, що визначає здатність спектроаналізатора розрізняти близькі по частоті компоненти сигналу. Мала RBW дає високу роздільну здатність, але це може збільшити час вимірювання і погіршити відношення сигнал/шум. Роздільна здатність за часом (Video Bandwidth, VBW) визначає швидкість оновлення спектрограми і її здатність відображати швидкі зміни сигналу. Вибір VBW залежить від того, як швидко змінюється сигнал, і від вимог до часової роздільної здатності.

Вікно (Window Function) — це математична функція, що застосовується до кожного сегмента сигналу перед перетворенням Фур'є[3]. Вибір вікна впливає на форму спектральних піків та може допомогти зменшити ефект розтікання спектра. Тип детектора (Detector Type) визначає метод вимірювання амплітуди частотних компонентів сигналу. Вибір детектора залежить від характеру сигналу і завдань аналізу. Наприклад, піковий

детектор використовується для вимірювання максимальної потужності сигналу, а середньоквадратичний — для визначення середньої потужності. Шкала амплітуди (Amplitude Scale) визначає, як амплітуда відображається на спектрограмі. Лінійна шкала показує амплітуду в лінійних одиницях, тоді як логарифмічна шкала (децибелі або дБм) дозволяє краще візуалізувати сигнали з великим динамічним діапазоном[3]. Інтерпретація спектрограми вимагає глибокого розуміння теорії спектрального аналізу та фізичних процесів, які визначають спектральні характеристики сигналу. З аналізу спектрограми можна визначити такі параметри сигналу, як центральна частота, ширина спектра, тип модуляції, наявність гармонік та інтермодуляційних спотворень, а також оцінити рівень шуму та перешкод. Правильне керування вимірюванням спектрограми є важливою умовою для отримання точних і корисних результатів. Це потребує від дослідника не тільки знань про налаштування технічних параметрів пристрою, але й глибокого розуміння природи сигналу і процесів, що відбуваються при його формуванні.

2 ПОБУДОВА ТА ОБРОБКА ДАНИХ СПЕКТРОГРАМИ

2.1 Отримання даних із SDR-приймача (HackRF)

Використання програмних радіоприймачів (SDR) для отримання та аналізу радіочастотних сигналів стає все більш популярним завдяки їхній гнучкості, низьким витратам та високій точності. У пункті 2.1 описано процес отримання та обробки радіочастотних сигналів за допомогою SDR-приймача HackRF[4]. Даний приймач дозволяє здійснювати збір даних для подальшого спектрального аналізу, а також забезпечує високу точність вимірювань на широкому діапазоні частот. Система реалізована за допомогою бібліотек GNU Radio та osmosdr, що дозволяє інтегрувати SDR-приймачі в середовище програмного радіо для обробки сигналів. Для отримання сигналу з HackRF використовується компонент osmosdr.source, який надає доступ до приймача через стандартні інтерфейси[4]. Подальша обробка сигналів виконується за допомогою блоку автоматичного регулювання підсилення (AGC) та різних фільтрів для оптимізації сигналів перед подальшим аналізом.

Основний клас RfScanner відповідає за ініціалізацію SDR-приймача та його налаштування для збору даних з певного частотного діапазону. Приймач налаштовується на бажану центральну частоту, з вибором відповідної швидкості дискретизації та кількості зразків, які необхідно зібрати для аналізу.

- Центральна частота (`center_freq`) визначає точку, навколо якої здійснюється сканування спектра.

- Швидкість дискретизації (`sample_rate`) вказує на кількість зразків, які приймач здатний отримати за одиницю часу, що визначає точність вимірювань.
- Кількість зразків (`num_samples`) встановлює кількість отриманих значень за кожен період вимірювання.

Отримані дані обробляються для отримання спектрограми за допомогою різних методів, зокрема перетворення Фур'є (FFT). Це дозволяє аналізувати спектр сигналу, а також виділяти ключові характеристики, такі як амплітуда та частота[4].

1. Частотний аналіз: Після отримання зразків здійснюється перетворення сигналу в частотну область за допомогою FFT. Це дозволяє побудувати спектр сигналу та виявити всі частотні компоненти.
2. Перетворення у логарифмічну шкалу: Для полегшення візуалізації та аналізу результатів значення потужності спектра перетворюються в логарифмічну шкалу (децибели).
3. Обробка даних: За допомогою алгоритмів, таких як `savgol_filter`, виконуються додаткові операції фільтрації для згладжування спектра та зменшення шуму.

Сканація частотного діапазону

Важливою частиною системи є функція `scan_frequency_range`, яка здійснює сканування заданого частотного діапазону, на якому виконуються вимірювання. Для кожної частоти в діапазоні система отримує спектр сигналу і зберігає його для подальшого аналізу.

1. Пошук пікових частот: зібрані спектри аналізуються на наявність пікових значень потужності, що можуть свідчити про присутність сигналів або передавачів в даному діапазоні.
2. Ідентифікація кандидатів: сигнали, що мають потужність вище за певний поріг, фільтруються та зберігаються як потенційно цікавими для подальшого аналізу.

Після того, як були виявлені цікаві частоти, функція `build_spectrogram` створює спектрограми для кожної з них, використовуючи методи STFT (перетворення Фур'є). Це дозволяє отримати візуалізацію змін частотного спектра сигналу з часом[5].

- Розмір вікна: Для кожного сигналу вибирається оптимальний розмір вікна та параметри фільтрації для побудови спектрограми.
- Частотна та часова роздільна здатність: Використовуються фільтри для корекції спектра та зменшення ефекту перекриття частот.

Візуалізація результатів

Отримані спектрограми виводяться на графіки, де по осі X відображається час, а по осі Y — частота. Амплітуда спектра представлена в децибелах, що дозволяє зручно порівнювати різні частотні компоненти сигналу. Залежно від потреб користувача, візуалізація може включати такі елементи, як[6]:

- Спектрограма потужності: Відображення амплітуди сигналу на різних частотах у часі.
- Фазова спектрограма: Визначення зміни фази сигналу в часі.
- Миттєва частота: Оцінка змін миттєвої частоти на основі фази сигналу.

Процес отримання даних з SDR-приймача HackRF і подальшого аналізу спектра є важливою частиною багатьох досліджень, що включають радіочастотні вимірювання. Зібрані дані дозволяють виявляти важливі характеристики сигналів, визначати присутність передавачів та оцінювати рівень шуму.

2.2 Нормалізація та попередня обробка сигналу (AGC, обмеження по довжині)

Нормалізація та попередня обробка сигналів є важливими етапами в обробці радіочастотних сигналів, оскільки ці процеси дозволяють забезпечити стабільність та точність вимірювань. У цьому розділі розглядаються два ключові етапи попередньої обробки сигналу, що здійснюються в процесі збору та аналізу даних з програмного радіоприймача (SDR): автоматичне регулювання підсилення (AGC) та обмеження за довжиною сигналу[7]. Автоматичне регулювання підсилення (AGC, Automatic Gain Control) є важливою технічною функцією для стабілізації рівня сигналу під час його отримання. Зазвичай, в реальних умовах прийом сигналу може бути нестабільним, із значними варіаціями потужності (наприклад, через зміни в атмосферних умовах або відстані до джерела сигналу). Відсутність регулювання рівня сигналу може призвести до спотворень, що ускладнюють подальший спектральний аналіз. AGC використовується для автоматичного коригування підсилення сигналу таким чином, щоб забезпечити його стабільний рівень протягом всього процесу вимірювань. Цей процес допомагає уникнути як перевантаження сигналу (коли сигнал надто сильний, що призводить до спотворень), так і недостатнього сигналу (коли сигнал занадто слабкий для точних вимірювань)[8].

Принцип роботи AGC

В процесі обробки сигналу застосовується спеціальний фільтр, що оцінює рівень вхідного сигналу. Якщо сигнал занадто слабкий, підсилення автоматично збільшується. Якщо сигнал надто сильний, підсилення зменшується, щоб уникнути перевантаження. Цей процес забезпечує постійну стабільність рівня сигналу на вході системи, що дозволяє отримати точні результати вимірювань, незалежно від зовнішніх умов. У реалізації RfScanner класу для SDR-приймача HackRF, AGC представлено за допомогою блоку `analog.agc_cc`, що налаштовується на певний коефіцієнт затримки та максимальний рівень підсилення[9].

Параметри AGC

1. Коефіцієнт регулювання підсилення визначає швидкість, з якою система адаптується до змін рівня сигналу.
2. Максимальне підсилення обмежує рівень підсилення, щоб уникнути перевантаження приймача та спотворень.
3. Порогове значення визначає мінімальний рівень сигналу, нижче якого сигнал вважається шумом.

У результаті роботи AGC вхідний сигнал автоматично нормалізується, що дозволяє уникнути непотрібних спотворень і підвищує точність подальшого спектрального аналізу. Обмеження по довжині сигналу є ще одним важливим етапом попередньої обробки, який застосовується для обмеження обсягу даних, що передаються до системи обробки. Це є особливо корисним при роботі з великими обсягами даних, оскільки дає змогу зменшити час обробки, знизити витрати на пам'ять і підвищити ефективність системи[10].

У радіочастотних вимірюваннях довжина сигналу визначається як кількість зразків, що використовуються для побудови спектрограми. У

реальних умовах тривалість сигналу може бути дуже великою, і для ефективної обробки необхідно обмежити її певною кількістю зразків[10].

Обмеження по довжині сигналу

1. Налаштування кількості зразків: у класі RfScanner для SDR-приймача HackRF, параметр num_samples визначає кількість зразків, які будуть використовуватися для подальшого аналізу. Зазвичай це значення вибирається таким чином, щоб охопити весь період сигналу, при цьому зберігаючи максимальну кількість корисних даних для аналізу.
2. Вибір кількості зразків на кожен інтервал часу дозволяє зберегти час обробки та пам'ять системи, при цьому не втрачаючи важливої інформації для спектрального аналізу.
3. Чим вища швидкість дискретизації, тим більше зразків потрібно для точного відображення сигналу. Це також впливає на час обробки та точність вимірювань.

Обмеження за довжиною сигналу дозволяє оптимізувати систему для роботи з великими обсягами даних, гарантуючи при цьому високу точність результатів. Обидва етапи попередньої обробки сигналу — AGC та обмеження за довжиною — працюють у тісній взаємодії для досягнення найкращих результатів у спектральному аналізі. AGC стабілізує рівень сигналу, що дозволяє уникнути його перевантаження або слабкості, в той час як обмеження за довжиною сигналу дозволяє зменшити обсяг оброблюваних даних без втрати важливої інформації[10]. Ці два етапи допомагають зробити процес збору та аналізу сигналів більш ефективним, зменшуючи навантаження на систему при збереженні високої точності вимірювань. Нормалізація та попередня обробка сигналу є важливими етапами в процесі збору і аналізу радіочастотних сигналів. Використання

автоматичного регулювання підсилення (AGC) дозволяє стабілізувати рівень сигналу, забезпечуючи точність вимірювань, а обмеження за довжиною сигналу дає змогу оптимізувати обсяг даних для ефективної обробки.

2.3 Перетворення Фур'є та згладжування спектру (FFT, Savitzky-Golay)

Перетворення Фур'є є основним інструментом для аналізу частотних компонентів сигналу. Воно дозволяє розкласти складний сигнал, що змінюється в часі, на простіші складові — синусоїдальні хвилі з певними частотами, амплітудами та фазами. Це перетворення є фундаментальним методом для аналізу сигналів у багатьох галузях науки і техніки, таких як фізика, електроніка, обробка сигналів, астрономія та багато інших. Перетворення Фур'є засноване на теоремі, яка стверджує, що будь-яка періодична функція може бути представлена як нескінченну суму синусоїдальних функцій з різними частотами, амплітудами і фазами[11]. Для неперіодичних сигналів використовується неперервне перетворення Фур'є, а для дискретних сигналів застосовується дискретне перетворення Фур'є (DFT). У випадку аналізу цифрових сигналів зазвичай використовують швидке перетворення Фур'є (FFT) — ефективний алгоритм для обчислення DFT. FFT дозволяє значно зменшити обчислювальні витрати, що робить можливим використання цього методу для реального часу і великих обсягів даних.

Дискретне перетворення Фур'є для сигналу, що складається з N дискретних зразків, можна записати як[11]:

$$X_k = \sum_{n=0}^{N-1} x_n \cdot e^{-\frac{2\pi i k n}{N}}, k = 0, 1, 2, \dots, N-1$$

де:

- x_n — значення сигналу в точці n ,
- X_k — комплексне значення перетворення на частоті k ,
- N — кількість зразків сигналу,
- i — уявна одиниця.

Швидке перетворення Фур'є (FFT) — це алгоритм для обчислення DFT, який використовує рекурсивний розподіл задачі на підзадачі, що дозволяє значно зменшити обчислювальну складність з $O(N^2)$ до $O(N \log_2 N)$ [12]. Після виконання перетворення Фур'є отриманий спектр може містити шум, що заважає коректному аналізу. Згладжування спектру є важливою технікою для покращення якості спектрограми, видалення високочастотного шуму і підвищення чіткості визначення частотних компонентів. Одним із найбільш популярних методів згладжування є фільтр Савіцького-Голлея (Savitzky-Golay filter), який здійснює локальне поліноміальне наближення для згладжування сигналу або спектру. Цей метод застосовується до сигналів, де необхідно зберегти як низькі, так і високі частоти, але при цьому усунути високочастотний шум.

Принцип роботи фільтра Савіцького-Голлея

Фільтр Савіцького-Голлея є відновлювальним фільтром, який для кожного зразка сигналу апроксимує його за допомогою полінома низького ступеня (зазвичай 2 або 3 ступеня), адаптуючи форму полінома до значень сусідніх точок сигналу[12]. Кількість точок, що використовуються для фільтрації, визначається вікном фільтра, яке ковзає по сигналу.

Процедура включає в себе наступні етапи:

1. Вибір розміру вікна (довжина сегмента сигналу, який буде згладжуватись).

2. Для кожного сегмента сигналу застосовується поліноміальне наближення, що мінімізує середньоквадратичну помилку між реальними значеннями і значеннями, що передбачаються поліномом.
3. Після фільтрації кожен новий зразок сигналу визначається як результат наближення, отриманого для кожного сегмента.

Математично, для сигналу y_n , значення в кожній точці n можна описати рівнянням[12]:

$$y_n = \sum_{k=-M}^M c_k \cdot x_{n+k}$$

де:

- y_n — значення фільтрованого сигналу в точці n ,
- x_{n+k} — значення сигналу в сусідніх точках,
- c_k — коефіцієнти поліноміальної апроксимації для кожного вікна,
- M — кількість точок, що беруть участь у згладжуванні.

У спектральному аналізі перетворення Фур'є та фільтр Савіцького-Голлея часто використовуються разом для досягнення точних і стабільних результатів. Першим кроком є перетворення сигналу у частотну область за допомогою FFT, що дозволяє отримати спектр сигналу. Потім застосовується фільтр Савіцького-Голлея, щоб зменшити шум у спектрі і зробити його більш виразним[13]. Це особливо корисно в контексті радіочастотного спектрального аналізу, де часто зібрані сигнали можуть містити шуми або інші спотворення, які заважають точному визначенню характеристик частотних компонентів. Згладжування спектру дозволяє точно виявляти ключові частотні компоненти, що мають важливе значення для подальшої інтерпретації та аналізу сигналів.

2.4 Побудова спектрограм на GPU (STFT, CuPy, cuSignal)

Спектрограма є потужним інструментом для аналізу частотного складу сигналу в часі. Вона дозволяє візуалізувати, як змінюються частотні компоненти сигналу протягом його тривалості[13]. Одним з основних методів для побудови спектрограми є віконне перетворення Фур'є (STFT). Віконне перетворення Фур'є (STFT) фрагментує сигнал, поділяючи на вікна. Для кожної частини сигналу(вікна) окремо обчислюється перетворення Фур'є. Таким чином ми отримуємо тенденцію потужностей частот в часі. STFT дає змогу побудувати 2D-матрицю, де одна ось представляє час, а інша — частоту. Висота кожної точки на спектрограмі відображає потужність сигналу на відповідній частоті в певний момент часу. Однак, при роботі з великими обсягами даних або вимірюваннями в реальному часі, стандартні методи обчислення STFT можуть бути дуже повільними і вимагати значних обчислювальних ресурсів. Для прискорення цього процесу можна використовувати обчислення на графічному процесорі (GPU)[13]. Використання GPU для обробки сигналів дозволяє значно зменшити час обчислення, що особливо корисно при обробці великих масивів даних або виконанні реальних вимірювань в режимі реального часу.

CuPy — це бібліотека для обчислень на GPU, яка є подібною до NumPy, але має оптимізацію для роботи на графічних процесорах за допомогою бібліотеки CUDA. CuPy дозволяє здійснювати числові операції та маніпуляції з масивами даних без необхідності вручну писати код для GPU. Одним із основних застосувань CuPy є обчислення STFT на GPU, що значно прискорює обробку сигналів, порівняно з використанням CPU. Важливими перевагами CuPy є[14]:

1. Простота у використанні: Бібліотека має інтерфейс, схожий на NumPy, що дозволяє легко переходити з однієї бібліотеки на іншу.
2. Прискорення обчислень: Всі операції виконуються на графічному процесорі, що значно зменшує час обробки великих обсягів даних.
3. Масштабованість: CuPy підтримує паралельні обчислення, що дозволяє ефективно працювати з великими масивами даних.

cuSignal — це бібліотека, розроблена на основі CuPy, яка спеціалізується на наукових обчисленнях та обробці сигналів на GPU. Вона надає готові інструменти для спектрального аналізу, включаючи функції для обчислення STFT. STFT за допомогою cuSignal дозволяє ефективно розраховувати спектрограми сигналів, зберігаючи високу точність і швидкість обробки. cuSignal реалізує STFT, використовуючи технології CUDA для обчислень на GPU, що дозволяє значно пришвидшити обробку великих сигналів і забезпечити можливість роботи з даними в реальному часі[15].

2.5 Використання віконних фільтрів (FIR, Nuttall, Bandpass)

Використання віконних фільтрів є важливим методом для обробки сигналів, що дозволяє покращити характеристики сигналу та зменшити ефекти спотворень. Віконні фільтри зазвичай застосовуються в дискретних сигналах для зміни або згладжування спектральних компонентів. Ключовою ідеєю є те, що сигнал розбивається на менші сегменти (вікна), кожне з яких обробляється окремо для досягнення бажаного ефекту фільтрації. Віконні фільтри можуть бути реалізовані як FIR (Finite Impulse Response) фільтри, які мають скінчену кількість коефіцієнтів і забезпечують надійне зменшення шуму та спотворень у спектрі сигналу[16].

FIR-фільтри використовують певну кількість коефіцієнтів для обчислення відгуку системи. Вони є основою для багатьох методів фільтрації, зокрема для згладжування сигналів, що містять високочастотні шуми. FIR фільтри забезпечують стабільну фазову характеристику, тому вони не спотворюють форму сигналу, що робить їх ефективними для обробки різних типів даних[17]. Вони можуть бути використані для виділення певних частотних компонентів сигналу або для застосування низькочастотних чи високочастотних фільтрів.

Вікно Нуттала — це спеціальний тип віконної функції, що використовується для фільтрації сигналу та зменшення розсіяння спектра. Воно має одну з найкращих характеристик у порівнянні з іншими типами вікон, таких як Хеммінга або Гауссівське вікно, особливо при обробці спектрів з високою роздільною здатністю[17]. Вікно Нуттала має дуже гладку форму та добре підходить для спектральних вимірювань, де важливо зберегти форму частотних компонентів та уникнути великих помилок в амплітудному спектрі. Воно також зменшує ефекти «розтікання» частот у спектрі, що робить його популярним у спектральному аналізі сигналів.

Bandpass фільтри використовуються для того, щоб пропускати тільки певний діапазон частот і блокувати всі інші. Це особливо корисно при аналізі сигналів, де ми хочемо виділити тільки певні частотні компоненти. Такі фільтри застосовуються для відфільтровування шуму поза заданим діапазоном частот і дозволяють зберегти важливу інформацію, яка знаходиться всередині діапазону[18]. Використовуючи Bandpass фільтри разом із віконними функціями, можна досягти високої точності у виділенні потрібних частотних компонентів сигналу, що робить їх важливими для задач спектрального аналізу. При обробці сигналу в реальному часі, застосування віконних фільтрів дає змогу знизити рівень шуму, покращити якість сигналу та зменшити спотворення. Зокрема,

використання FIR фільтрів разом з вікном Нуттала дозволяє виконати фільтрацію без значного спотворення частотних компонентів сигналу. Bandpass фільтри допомагають ізольовати потрібний діапазон частот, що є важливим при скануванні радіочастотних діапазонів, зокрема для розпізнавання передавачів у радіосигналах.

2.6 Детекція піків та оцінка ширини смуги (CuPy find_peaks)

Детекція піків є важливим етапом в аналізі спектра сигналів, оскільки дозволяє виявити максимуми амплітуди в спектрі, які відповідають за важливі частотні компоненти сигналу. Піки на спектрограмі можуть вказувати на наявність активних передавачів, специфічних модуляцій або інших значущих подій у сигналу[19]. Для виявлення піків у спектрі використовуються різні математичні методи, серед яких один з найпоширеніших — це алгоритм `find_peaks`. Цей метод шукає максимуми в масиві даних, використовуючи певні критерії, такі як мінімальна висота піка, мінімальна відстань між піками та інші параметри.

CuPy є бібліотекою для виконання числових обчислень на графічному процесорі (GPU), і вона має вбудовану функцію `find_peaks`, яка дозволяє здійснювати детекцію піків на масивах даних, зберігаючи високі обчислювальні швидкості завдяки паралельному виконанню операцій на GPU. CuPy `find_peaks` використовує ефективні алгоритми для пошуку піків у великих масивах даних. Завдяки паралельним обчисленням на GPU, ця функція значно прискорює процес детекції піків, що особливо корисно при обробці великих сигналів або в режимі реального часу[20].

Оцінка ширини смуги є важливим етапом в спектральному аналізі, оскільки дозволяє визначити, яка частина спектра є активною. Для цього можна використовувати кілька методів, зокрема оцінку ширини пікових структур у спектрі. Ширина смуги визначається як відстань між двома точками на спектрі, де амплітуда сигналу падає до певного порогу

відносно максимуму[20]. Для оцінки ширини смуги використовується аналіз відстані між піками в спектрі, а також визначення їх амплітудних рівнів на заданому порозі. Алгоритм `find_peaks` у `CuPy` дозволяє не тільки знаходити піки, але й визначати їх ширину, що є корисним для аналізу спектральних характеристик сигналів, таких як радіочастотні передавачі. У спектральному аналізі, особливо при обробці радіочастотних сигналів, детекція піків та оцінка ширини смуги є критичними для виявлення активних каналів, оцінки частотного діапазону передавачів та виявлення перешкод. Застосування `CuPy` для цих завдань дозволяє значно прискорити обчислення, що особливо важливо при аналізі сигналів у реальному часі. Використання віконних фільтрів, таких як `FIR`, `Nuttall` та `Bandpass`, є потужним інструментом для обробки сигналів у спектральному аналізі. Ці фільтри дозволяють ефективно знижувати рівень шуму та спотворень, а також виділяти важливі частотні компоненти сигналу. Детекція піків та оцінка ширини смуги, зокрема за допомогою `CuPy find_peaks`, є важливими етапами для виявлення активних частот і оцінки характеристик сигналу. Застосування цих методів у поєднанні з обчисленнями на `GPU` забезпечує високу швидкість обробки і дозволяє ефективно аналізувати великі обсяги даних в реальному часі[21].

2.7 Візуалізація результатів (Matplotlib)

Візуалізація даних є важливою складовою частиною процесу аналізу сигналів, оскільки вона дозволяє дослідникам і інженерам ефективно інтерпретувати результати обробки та зробити висновки на основі графічного відображення. Застосування візуалізації сигналів особливо актуальне при роботі з великими обсягами даних, де труднощі інтерпретації можуть виникнути через велику кількість вимірювань або складні взаємозв'язки між різними параметрами сигналу. `Matplotlib` — це одна з найбільш популярних бібліотек для візуалізації даних у `Python`.

Вона дозволяє створювати широкий спектр графічних відображень, таких як графіки, гистограми, діаграми, спектрограми тощо. Завдяки своїй гнучкості, Matplotlib є незамінним інструментом для візуалізації результатів у багатьох галузях науки та техніки, зокрема при аналізі сигналів. Побудова спектрограм є важливим етапом у візуалізації результатів спектрального аналізу сигналу. Спектрограма відображає частотний склад сигналу в часі, що дозволяє дослідникам побачити, як змінюються частотні компоненти сигналу протягом часу. Matplotlib дозволяє зручно візуалізувати спектрограми, що є особливо корисним при аналізі складних сигналів або радіочастотних спектрів[52].

Основні етапи побудови спектрограми за допомогою Matplotlib:

1. Для побудови спектрограми сигнал спочатку піддається короткочасному перетворенню Фур'є (STFT). Це дозволяє отримати двовимірну матрицю, де одна ось відображає час, а інша — частоту.
2. Оскільки спектрограма зазвичай показує амплітуду сигналу, логарифмічне перетворення амплітуди в децибелі дозволяє отримати більш наочне відображення сигналу з великим динамічним діапазоном.
3. Використовуючи `imshow` або `pcolormesh` з Matplotlib, можна побудувати кольорову картину спектрограми, де колір кожної точки на графіку відображає потужність сигналу на відповідній частоті і в певний момент часу.

3 ВИКОРИСТАННЯ СПЕКТРОГРАМ ДЛЯ ВИЯВЛЕННЯ ПЕРЕДАВАЧІВ

3.1 Типи модуляцій і їхній спектральний вигляд

У сфері обробки сигналів, зокрема в радіозв'язку та телекомунікаціях, важливим аспектом є розуміння різних типів модуляцій, оскільки вони впливають на спектральні характеристики сигналів. Модуляція — це процес, при якому властивості несучого сигналу (наприклад, амплітуда, частота або фаза) змінюються відповідно до інформаційного сигналу[22]. У радіо- і телекомунікаціях модуляція дозволяє передавати інформацію по радіочастотних каналах. Існує кілька основних типів модуляцій, кожен з яких має свої спектральні характеристики. Основні види модуляції можна поділити на три основні категорії: амплітудну модуляцію (АМ), частотну модуляцію (FM) і фазову модуляцію (PM). Кожен тип модуляції має свій характерний спектр, який визначає, як енергія сигналу розподіляється по частотному діапазону[23].

Амплітудна модуляція (АМ) є одним із найстаріших типів модуляції, що широко використовуються для радіомовлення. У модуляції амплітуди несучого сигналу змінюється відповідно до інформаційного сигналу, наприклад, голосу чи музики. Спектр сигналу АМ складається з несучої частоти та двох бокових смуг, що утворюються в результаті зміни амплітуди. Спектральний вигляд АМ-сигналу[24]:

1. Несуча частота (carrier frequency): Це центральна частота, яка залишається незмінною.

2. Бокові частоти (sidebands): Це дві симетричні частоти, які з'являються навколо несуча частоти, і відстань між ними відповідає частоті модуляційного сигналу.

Спектр АМ сигналу можна описати наступним чином:

- Сигнал АМ має основну несучу частоту, навколо якої розташовані дві симетричні бокові смуги на відстані від носія, що дорівнює частоті модуляційного сигналу.
- Зазвичай, потужність сигналу більш зосереджена в бокових смугах, і саме на них передається інформація.

Спектр АМ сигналу має вигляд двох піків на бокових частотах, з піком на несучій частоті, якщо немає надмірного фільтрування. Збільшення амплітуди сигналу або модуляційного сигналу призводить до ширшого спектра, що може займати значний діапазон частот.

Частотна модуляція (FM) є методом модуляції, при якому частота несучого сигналу змінюється відповідно до модуляційного сигналу. FM широко використовується для радіомовлення, особливо для високоякісних звукових передач, оскільки ця модуляція має високу стійкість до шумів і спотворень[25]. Спектр FM сигналу має вигляд серії гармонік, що розташовані навколо несучої частоти. Ключові особливості FM спектру:

1. Несуча частота: Як і в АМ, це основна частота сигналу, яка є центральною у спектрі.
2. Гармоніки: У спектрі FM сигналу є кілька гармонік, розташованих на цілих кратних відстанях від несучої частоти. Відстань між гармоніками залежить від ступеня модуляції, тобто від максимального відхилення частоти.

3. Бокові смуги: Замість двох бокових смуг, як у АМ, FM має набагато більше бокових компонентів, що зумовлене природою частотної модуляції.

Чим більше частотне відхилення (коли модулюючий сигнал має велику амплітуду), тим більше гармонік буде присутньо в спектрі. Це означає, що FM має значно ширший спектр, ніж АМ, і вимоги до ширини каналу для передачі FM сигналу також вищі. Фазова модуляція (PM) є типом модуляції, при якому фаза несучого сигналу змінюється відповідно до модуляційного сигналу. Вона є математично еквівалентною частотній модуляції, оскільки зміна фази носія з часом аналогічна зміні частоти. Однак, на відміну від FM, де частота змінюється в залежності від амплітуди модуляційного сигналу, у PM фаза змінюється безпосередньо, і це визначає спектр сигналу[26].

Спектральний вигляд PM-сигналу:

1. Несуча частота: Як і в АМ та FM, це центральна частота сигналу.
2. Гармоніки: Спектр PM сигналу містить гармоніки, розташовані на відстані, що залежить від величини зміщення фази.
3. Бокові смуги: Хоча спектр PM сигналу дуже схожий на спектр FM сигналу, його гармоніки можуть мати інший розподіл, залежно від типу фази і амплітуди сигналу.

Спектр PM-сигналу також може бути схожим на спектр FM, але в ньому менш виражена симетрія, і він більш залежить від фази модулюючого сигналу. У сучасних телекомунікаціях також використовуються складніші види модуляцій, такі як модуляція з амплітудною та фазовою зміною (QAM, Quadrature Amplitude Modulation), модуляція частоти з фазовою зміною (FSK, Frequency Shift Keying) та інші[27]. Ці методи використовують комбінації АМ, FM та PM для

передачі більшої кількості інформації на обмеженому частотному діапазоні. У QAM спектр складається з двох взаємно перпендикулярних сигнальних компонентів, один з яких модуляційно змінюється за амплітудою, а інший — за фазою. Спектр QAM сигналу має більш компактну структуру, ніж у AM чи FM, і характеризується меншою шириною смуги.

3.2 Визначення основної частоти та гармонік

Основна частота (fundamental frequency) є найнижчою частотою в спектрі періодичного сигналу і визначає основну гармоніку цього сигналу. У випадку чистого синусоїдального сигналу основна частота — це єдина частота, яка присутня в спектрі. Однак у більшості реальних сигналів, таких як модуляційні сигнали, основна частота може бути супроводжувана гармоніками, що утворюються в результаті змін частоти або амплітуди[28].

Для визначення основної частоти сигналу можна використовувати різні методи, зокрема[29]:

1. Аналіз спектра: Оскільки основна частота відповідає піковому значенню в спектрі сигналу, її можна знайти шляхом ідентифікації найбільш вираженого піка.
2. Автокореляція: Для періодичних сигналів автокореляція дає пік на інтервалі, що дорівнює періоду сигналу, від якого можна визначити основну частоту.

У частотному спектрі періодичного сигналу основна частота завжди є найнижчою частотою, що присутня в його розкладі. Вона відображає періодичність сигналу і є ключовою характеристикою для подальшого аналізу.

Гармоніки — це частоти, які є цілими кратними основної частоти сигналу[30]. Для ідеального синусоїдального сигналу гармоніки відсутні. Однак у реальних сигналах наявність гармонік є звичайним явищем, оскільки сигнал часто має спотворення, викликані нелінійними ефектами при його формуванні або передачі. Наприклад, якщо основна частота f_0 , то гармоніки будуть мати частоти:

- 2-й гармонік: $2f_0$
- 3-й гармонік: $3f_0$
- 4-й гармонік: $4f_0$
- і так далі.

Гармоніки зазвичай мають зменшену амплітуду порівняно з основною частотою, і їх можна визначити через спектрограму, де кожна гармоніка з'являється як пік на кратних відстанях від основної частоти[31]. Для виявлення гармонік у спектрі часто використовуються методи спектрального аналізу, такі як FFT (Fast Fourier Transform), що дозволяють швидко і точно виявляти не тільки основну частоту, а й всі присутні гармоніки.

3.3 Параметри спектрів FM, PSK, OFDM, ППРЧ

Частотна модуляція (FM) — це тип модуляції, при якому частота несучого сигналу змінюється відповідно до модуляційного сигналу[32]. Спектр FM сигналу має наступні характеристики:

1. Несуча частота - частота, на якій розташований основний пік спектру.
2. Гармоніки: спектр FM сигналу складається з множини гармонік, які з'являються на відстані від несучої частоти. Це означає, що FM має набагато більший спектр порівняно з AM.

3. Ширина смуги: Ширина смуги FM сигналу залежить від величини частотного відхилення та частоти модуляційного сигналу. Чим більше відхилення частоти, тим ширший спектр.
4. Модулюючий сигнал: Частотна модуляція використовує амплітуду та частоту модуляційного сигналу для створення спектральних компонентів. Якщо модуляційний сигнал має велику амплітуду, спектр сигналу розширюється, і навпаки.

Фазова модуляція (PSK, Phase Shift Keying) є типом модуляції, при якому фаза несучого сигналу змінюється відповідно до цифрового модуляційного сигналу. Спектр PSK сигналу має наступні характеристики[33]:

1. Несуча частота: Як і в інших модуляціях, несуча частота є основною частотою сигналу.
2. Дискретність: У спектрі PSK сигналу присутні дискретні спектральні компоненти, які відповідають змінам фази.
3. Бокові смуги: Спектр PSK сигналу містить бокові смуги, які залежать від кількості змін фаз, що використовуються для модуляції. Кількість бокових смуг зростає при збільшенні кількості фазових рівнів (наприклад, в QPSK використовується чотири фази).
4. Ширина смуги: Спектр PSK сигналу має обмежену ширину смуги, але може мати високі пік-фактори в залежності від кількості фазових рівнів.

Ортогональний частотний поділ (OFDM) — це метод модуляції, який використовує множину субнесучих частот, розташованих так, щоб вони не перекривалися, зберігаючи ортогональність між собою. Спектр OFDM сигналу має наступні характеристики[34]:

1. Основна частота: Це центральна частота, яка знаходиться в середині спектру, і на ній немає значної потужності.
2. Субнесучі частоти: Спектр OFDM складається з ряду вузькочастотних компонентів (субнесучих), розташованих на певних інтервалах. Ці компоненти ортогональні один до одного, що дозволяє ефективно використовувати ширину смуги.
3. Ширина смуги: Ширина смуги OFDM сигналу визначається кількістю субнесучих частот, які використовуються для передачі даних. Чим більше субнесучих, тим ширший спектр.
4. Міжсмугові інтерференції: Оскільки субнесучі частоти ортогональні, вони не заважають одна одній, що дозволяє використовувати весь доступний частотний ресурс ефективно.

Псевдопорушені радіочастотні сигнали (ППРЧ) — це сигнали, які використовуються для захищених передач у радіозв'язку[35]. Вони генеруються шляхом порушення частоти або фази несучого сигналу за допомогою спеціальних алгоритмів, що забезпечують стійкість до перешкод.

1. Ширина смуги: ППРЧ сигнали зазвичай мають широкі спектри, оскільки для покращення стійкості до перешкод використовуються складні методи модуляції.
2. Шумоподібність: Спектри ППРЧ сигналів можуть виглядати як шум, що ускладнює їх виявлення і підвищує стійкість до спотворень.
3. Висока стійкість до інтерференцій: Завдяки своїй структурі, ППРЧ сигнали здатні зберігати стійкість навіть у разі сильних інтерференцій, що робить їх ідеальними для систем з високими вимогами до надійності.

Типи модуляцій мають свої унікальні спектральні характеристики, які визначають ефективність передачі сигналів у різних умовах. Амплітудна модуляція (AM) має просту структуру спектра з основною частотою і боковими смугами, частотна модуляція (FM) характеризується широким спектром і гармоніками, а фазова модуляція (PSK) має спектр з дискретними компонентами, залежно від кількості фаз[36]. Ортогональний частотний поділ (OFDM) забезпечує ефективне використання частотного діапазону завдяки ортогональним піднесучим, а ППРЧ сигнали використовуються для захищених комунікацій з широкими спектрами і високою стійкістю до перешкод.

3.4 Алгоритм автоматичного визначення типу передавача

Алгоритм автоматичного визначення типу передавача включає в себе кілька ключових етапів, які дозволяють ідентифікувати тип модуляції та властивості передавальних сигналів. Основною метою є автоматичне виявлення активних передавачів, оцінка їхніх спектральних характеристик та визначення їхнього типу модуляції[37]. Алгоритм використовує комбіновані методи обробки сигналів, що включають визначення порогу потужності на основі медіани, зменшення роздільної здатності (downsampling), пошук піків і побудову спектрограм для аналізу частотних коливань і модуляцій.

1. Медіана потужності для визначення порогового рівня

Медіана потужності є ефективним методом для визначення порогу, коли сигнал вважається кандидатом. Цей підхід базується на визначенні порогу потужності як певного відсотка від медіанної потужності в спектрі. Такий підхід дозволяє надійно виділяти значущі сигнали, що значно перевищують загальний рівень шуму, навіть якщо сигнал містить викиди. Медіана є стійким методом, оскільки вона не чутлива до крайнощів у даних, що робить її особливо корисною у випадках, коли сигнал містить

шум або інші аномалії[38]. Поріг потужності визначається як множення медіани потужності на певний коефіцієнт (наприклад, 1.05 для 5% перевищення). Якщо потужність сигналу перевищує цей поріг, сигнал вважається кандидатом.

2. Downsampling через середнє значення

Для зменшення роздільної здатності сигналу (downsampling) використовується метод середнього значення. В даному випадку сигнал групується в вікна, і для кожного вікна обчислюється середнє значення. Це дозволяє значно зменшити обсяг даних, зберігаючи при цьому основні характеристики сигналу[39]. Метод середнього значення є ефективним для зменшення шуму і спрощення даних, оскільки він дозволяє зберігати основні тенденції сигналу, одночасно знижуючи вплив високочастотних компонентів, які можуть бути шуми.

3. Детекція піків з використанням `find_peaks`

Виявлення піків на спектрі сигналу здійснюється за допомогою функції `find_peaks` з бібліотеки SciPy. Цей метод дозволяє ідентифікувати найбільш виражені компоненти спектру, які можуть вказувати на активні передавачі. Основні параметри для пошуку піків включають:

Висота піків: Піки, що перевищують порогове значення (визначене через медіану потужності).

Ширина піків: Оцінка ширини спектрального компонента для визначення типу модуляції.

Відстань між піками: Для зменшення хибних спостережень необхідно враховувати мінімальну відстань між піками, щоб уникнути їхнього накладання. Цей метод є швидким і ефективним, що робить його особливо корисним при обробці великих обсягів даних або при вимірюваннях у реальному часі.

4. Запис нового сигналу на вузькому діапазоні спектру і побудова спектрограми

Після виявлення піків на широкому спектрі сигналів, фокусується увага на вузькому частотному діапазоні для детальнішого аналізу. Для цього використовуються методи спектрального аналізу, такі як Fast Fourier Transform (FFT) або Wavelet Transform, що дозволяють отримати часо-частотні характеристики сигналу. Спектрограма дає змогу візуалізувати коливання частоти сигналу навколо несучої частоти. Це важливо для оцінки модуляційних характеристик, таких як частотна модуляція (FM), або для вивчення інших видів модуляцій, таких як фазова модуляція (PSK), де частота змінюється в залежності від часу.

Медіана потужності забезпечує стійкість до викидів у сигналі, що робить її оптимальним методом для виявлення найбільш виразних характеристик передавача. Оскільки цей метод не чутливий до викидів і шумів, він дозволяє точно виявляти активні сигнали навіть за умов непередбачуваних перешкод. Середнє для `downsampling` дає можливість значно зменшити обсяг даних, зберігаючи при цьому основні характеристики сигналу. Це простий і ефективний метод для підготовки сигналу до подальшого аналізу, зменшуючи шум і спрощуючи дані без значних втрат важливої інформації.

Функція `find_peaks` є швидким і ефективним методом для виявлення піків у спектрі сигналу. Вона підходить для аналізу великих обсягів даних і реального часу, де важливе швидке реагування і точність виявлення. Спектрограма дозволяє наочно оцінити частотні коливання сигналу і досліджувати модуляцію сигналу в часі. Цей метод є корисним для розпізнавання типів модуляцій, таких як FM або PSK, де важливим є точне визначення частотних змін у часі.

Алгоритм автоматичного визначення типу передавача, базуючись на поєднанні методів для фільтрації, виявлення піків і спектрального аналізу, забезпечує високу точність у визначенні характеристик сигналів. Використання медіани для порогу потужності, середнього для

downsampling, функції `find_peaks` для детекції піків, а також побудова спектрограм дозволяють ефективно ідентифікувати типи модуляцій і передавачів, що важливо для розпізнавання радіочастотних сигналів у реальному часі.

4 ПРОГРАМНА РЕАЛІЗАЦІЯ ТА ЕКСПЕРИМЕНТАЛЬНЕ ДОСЛІДЖЕННЯ

4.1 Основи технології SDR

Software Defined Radio (SDR) — це технологія радіозв'язку, де основні функції радіоприймача і радіопередавача виконуються програмно за допомогою комп'ютерних алгоритмів замість традиційних апаратних компонентів, таких як фільтри, підсилювачі та модулятори[40]. Ця концепція дозволяє створювати гнучкі та універсальні радіосистеми, які можуть працювати в різних частотних діапазонах і підтримувати різноманітні стандарти зв'язку, змінюючи програмне забезпечення замість апаратного оновлення.

1. Основи SDR

Технологія SDR виникла для подолання обмежень традиційних апаратних радіосистем, які часто були обмежені одним частотним діапазоном або конкретним стандартом модуляції[41]. Замість того, щоб будувати спеціалізовані пристрої для кожного застосування, SDR дозволяє використовувати один апаратний пристрій, який можна перепрограмувати для виконання різних функцій. Основні компоненти SDR[41]:

- Апаратне забезпечення: включає в себе приймачі і передавачі радіосигналів, які можуть працювати на різних частотах та з різними типами модуляцій. До таких компонентів відносяться аналого-цифрові перетворювачі (ADC), цифро-аналогові перетворювачі (DAC), фільтри та підсилювачі.

- Програмне забезпечення: управляє радіосистемою, виконуючи необхідні алгоритми для обробки сигналів, таких як демодуляція, фільтрація, аналіз спектра, ідентифікація сигналу та інші.

2. Структура SDR-системи

Типова SDR-система складається з кількох основних компонентів[42,43]:

- Антена: Спеціально налаштована для прийому та передачі радіосигналів на певних частотах.
- Приймач (Rx): Приймає радіосигнал, перетворює його в електричний сигнал і передає його на аналого-цифровий перетворювач (ADC).
- Цифро-аналоговий перетворювач (DAC): Вихідний сигнал після обробки програмним забезпеченням передається через DAC для перетворення в аналоговий сигнал для передачі.
- Процесор обробки сигналу (DSP): Це програмна частина SDR, яка відповідає за обробку отриманого сигналу, включаючи фільтрацію, демодуляцію, аналіз спектра, розпізнавання модуляцій тощо.

Основна відмінність SDR від традиційних радіо-систем полягає в тому, що більшість функцій, які зазвичай виконуються апаратними компонентами, виконуються програмно. Це дозволяє змінювати характеристики радіосистеми без необхідності заміни апаратного забезпечення.

3. Ключові компоненти SDR

ADC (Analog-to-Digital Converter) — перетворює аналоговий сигнал, який надходить через антени, в цифровий сигнал, який може бути оброблений комп'ютером[44]. Важливим параметром є швидкість

дискретизації ADC, оскільки вона визначає, як точно цифровий сигнал відобразатиме аналоговий.

DAC (Digital-to-Analog Converter) — використовується для перетворення цифрового сигналу в аналоговий перед його передачею через антени[45]. Якість DAC також важлива для точності переданих сигналів. DSP у SDR складається з численних програмних блоків, таких як фільтрація, модуляція, демодуляція, аналіз спектра та інші функції. Наприклад, для фільтрації сигналу можуть використовуватись FIR (Finite Impulse Response) або IIR (Infinite Impulse Response) фільтри. Обробка сигналу може включати різноманітні алгоритми для демодуляції, що дозволяють зняти інформацію з сигналу. Також застосовуються алгоритми для пошуку і виявлення частотних компонентів. В деяких SDR системах для прискорення обробки сигналів використовуються FPGA (Field-Programmable Gate Arrays) та GPU (Graphics Processing Units). FPGA дозволяє реалізувати частину обробки сигналу на апаратному рівні, забезпечуючи високу швидкість і ефективність. GPU також використовується для паралельної обробки великих обсягів даних, таких як аналіз спектрів або фільтрація сигналів в реальному часі[45].

4. Переваги та недоліки SDR

Переваги SDR:

- Одне апаратне забезпечення можна перепрограмувати для роботи з різними частотами і стандартами зв'язку, що робить SDR системи універсальними.
- Оновлення чи зміна параметрів радіосистеми може бути здійснено шляхом оновлення програмного забезпечення, що значно знижує витрати на нове апаратне забезпечення.

- SDR дозволяє реалізувати різні типи модуляцій і частотних діапазонів без необхідності фізичного зміщення чи перепрограмування апаратних компонентів.
- Одне SDR-устрій може підтримувати декілька стандартів зв'язку (наприклад, AM, FM, PSK, QAM) і різні частотні діапазони, що робить його ідеальним для комплексних систем, таких як мобільні та супутникові комунікації.

Недоліки SDR:

- SDR вимагає великих обчислювальних потужностей для обробки сигналу, що може призводити до потреби у високопродуктивних комп'ютерах, особливо для роботи в реальному часі.
- Хоча програмне забезпечення може бути гнучким, високоякісні ADC та DAC, а також FPGA або GPU можуть бути дорогими.
- Завдяки складним алгоритмам обробки сигналів може виникати затримка в обробці сигналу, що важливо враховувати при застосуванні в реальному часі.

5. Застосування SDR

SDR дозволяє створювати гнучкі системи для мобільного зв'язку, які підтримують різні стандарти без потреби заміни апаратного забезпечення. В супутникових системах SDR дозволяє адаптувати передавачі та приймачі для роботи в різних частотних діапазонах та з різними протоколами. SDR використовується для побудови систем для аналізу радіочастотного спектра, виявлення перешкод або непроханих передавачів. Використання SDR дозволяє адаптувати навігаційні приймачі для роботи з різними навігаційними сигналами[45]. SDR дозволяє підтримувати протоколи та стандарти для бездротових мереж, що постійно змінюються, такі як Wi-Fi, Bluetooth або 5G. Технологія Software Defined

Radio (SDR) є потужним інструментом, що дозволяє значно спростити проектування радіосистем та забезпечити високу гнучкість і масштабованість. Завдяки використанню програмного забезпечення для виконання більшості функцій обробки сигналу, SDR дозволяє створювати універсальні системи зв'язку, які можуть підтримувати різні стандарти і працювати в широкому частотному діапазоні. Однак для досягнення високої ефективності і мінімізації затримок необхідно враховувати вимоги до обчислювальних потужностей та апаратного забезпечення.

4.2 Використання бібліотек `osmosdr`, `gnuradio`, `cuSignal`

Використання бібліотек `osmosdr`, `gnuradio` та `cuSignal` є важливою частиною програмування для систем Software Defined Radio (SDR). Кожна з цих бібліотек має свої унікальні можливості і застосування, що дозволяє розробникам створювати високопродуктивні системи для обробки сигналів і комунікаційних протоколів. Нижче детально розглянемо кожну з цих бібліотек і їх застосування в SDR-системах.

`osmosdr` — це бібліотека, що забезпечує інтерфейс для роботи з різними SDR-приймачами, такими як HackRF, RTL-SDR, USRP, і багатьма іншими. Вона дозволяє взаємодіяти з апаратним забезпеченням SDR і здійснювати прийом/передачу сигналів, отриманих через SDR-приймач[46].

Основні функції бібліотеки `osmosdr`:

- Інтерфейс з різними SDR-пристроями: через `osmosdr` можна працювати з різними пристроями SDR, такими як HackRF, RTL-SDR, USRP тощо. Це дозволяє розробляти універсальні програми, які можуть працювати з кількома апаратними платформами.

- Взаємодія з SDR через потокову передачу даних: бібліотека дозволяє передавати і приймати потоки радіосигналів у реальному часі. Вона підтримує як безперервний потік даних, так і обмежений набір даних.
- Налаштування параметрів пристрою: через `osmosdr` можна налаштувати різні параметри SDR, такі як частота, швидкість дискретизації, рівень посилення тощо.

GNU Radio — це потужна відкрито програмна платформа для розробки SDR-систем[47]. Вона дозволяє створювати складні радіо-комунікаційні системи, використовуючи графічний інтерфейс або програмний код. GNU Radio підтримує різноманітні функції для обробки сигналів, такі як фільтрація, модуляція, демодуляція, спектральний аналіз та інші алгоритми цифрової обробки сигналу.

Основні особливості GNU Radio:

- Графічний інтерфейс: GNU Radio має вбудований графічний інтерфейс під назвою GNU Radio Companion (GRC), який дозволяє створювати SDR-системи за допомогою блочних діаграм. Це дозволяє користувачам зручно розробляти і тестувати системи без написання коду.
- Модульність: GNU Radio має модульну архітектуру, що дозволяє додавати нові блоки обробки сигналів або змінювати наявні для специфічних завдань.
- Підтримка апаратних засобів: GNU Radio підтримує широкий спектр апаратних платформ SDR, включаючи USRP (Universal Software Radio Peripheral), HackRF, RTL-SDR та інші.

`cuSignal` — це бібліотека, яка дозволяє виконувати обробку сигналів на графічному процесорі (GPU) за допомогою `CuPy`[48]. Вона

оптимізована для швидкої обробки великих масивів даних і дозволяє використовувати всі переваги паралельних обчислень на GPU, що забезпечує високу швидкість при обробці великих обсягів сигналів у реальному часі.

Особливості cuSignal:

- Обчислення на GPU: cuSignal використовує потужність графічних процесорів для прискорення операцій, таких як перетворення Фур'є (FFT), фільтрація, аналіз спектра та інші.
- Сумісність з NumPy: Бібліотека cuSignal побудована на основі CuPy, що дозволяє використовувати ті ж самі функції та інтерфейси, як у NumPy, але з виконанням на GPU.
- Паралельна обробка: cuSignal дозволяє ефективно виконувати паралельні обчислення для великих обсягів даних, що є критичним для SDR-систем, де обробка сигналів у реальному часі є важливою.

osmosdr забезпечує інтерфейс для роботи з різними SDR-приймачами, дозволяючи працювати з апаратним забезпеченням, таким як HackRF або RTL-SDR. gnuradio є потужною платформою для створення складних SDR-систем, що підтримує різні модуляції, демодуляції, фільтрацію та аналіз спектра. cuSignal дозволяє значно прискорити обробку сигналів на GPU, що є важливим при роботі з великими обсягами даних або в реальному часі. Завдяки цим бібліотекам, розробники можуть створювати ефективні та гнучкі SDR-системи для різноманітних завдань у радіозв'язку, моніторингу спектра, дослідження модуляцій та багатьох інших областях.

4.3 Особливості пристрою HackRF

HackRF — це універсальний апаратний засіб для програмованого радіозв'язку (SDR), який може працювати як передавач, так і приймач сигналів[49]. HackRF став популярним серед радіоаматорів, дослідників і інженерів завдяки своїй доступності, гнучкості та підтримці широкого спектра частот. В основному він використовує програмне забезпечення для визначення, як обробляти сигнали, але має апаратні можливості для передачі і прийому радіосигналів в широкому діапазоні частот. HackRF підтримує частотний діапазон від 1 МГц до 6 ГГц, що охоплює величезний спектр використаних частот для бездротових технологій. Це дозволяє працювати як з низькочастотними радіо-сигналами (наприклад, АМ і FM радіо), так і з високочастотними системами (Wi-Fi, Bluetooth, LTE, супутникові комунікації). HackRF має швидкість дискретизації до 20 МС/с (мільйонів вибірок на секунду) для передачі і прийому сигналів. Ця швидкість є достатньою для обробки широких частотних діапазонів, що дозволяє проводити спектральний аналіз і обробку сигналів з високою точністю. HackRF підтримує смугу пропускання до 10 МГц, що дозволяє передавати і отримувати сигнали в значному частотному діапазоні. Це робить HackRF зручним для роботи з різними стандартами бездротового зв'язку. HackRF підключається до комп'ютера через порт USB 2.0. Це дозволяє здійснювати обробку сигналів на ПК за допомогою різних бібліотек програмного забезпечення, таких як GNU Radio або SDRSharp. HackRF має вбудований генератор частоти, який дозволяє налаштовувати пристрій на точні частоти з високою точністю. Проте варто зазначити, що для досягнення більш високої точності може знадобитися використання зовнішнього генератора сигналу або так званого GPS-DO (GPS Disciplined Oscillator)[49].

HackRF є одним з найбільш доступних SDR-пристроїв на ринку, що робить його популярним серед радіоаматорів, дослідників і тих, хто починає працювати з SDR. HackRF підтримує широкий спектр частот і може бути налаштований на різні види сигналів, включаючи AM, FM, цифрові модуляції, Wi-Fi, Bluetooth, LTE, а також супутникові сигнали. Це дозволяє використовувати HackRF для багатьох застосувань, від радіоаматорських експериментів до професійних досліджень в області бездротових технологій[50]. Оскільки HackRF працює з програмним забезпеченням, користувачі можуть легко змінювати конфігурацію пристрою для роботи з різними стандартами зв'язку і здійснювати налаштування параметрів передачі або прийому. Це забезпечує гнучкість у використанні і можливість модернізації системи за допомогою програмного оновлення. HackRF сумісний з багатьма популярними бібліотеками SDR, такими як GNU Radio, SDRSharp, Pothosware і osmoSDR. Це дозволяє легко інтегрувати HackRF у більші системи для бездротового зв'язку або аналізу спектра. HackRF має відкритий апаратний і програмний код, що дозволяє користувачам модифікувати або розширювати його функціональність для специфічних завдань. Це також сприяє активній спільноті розробників, які можуть поділитися своїми вдосконаленнями[50]. HackRF може працювати з невеликою потужністю сигналу, що обмежує його використання в деяких сферах, де потрібна висока потужність для передачі на великі відстані (наприклад, для супутникових зв'язків або мобільних мереж). HackRF має досить хорошу точність частоти, але для більш вимогливих додатків, де потрібна надзвичайно висока точність, можуть знадобитися більш дорогі пристрої з точними генераторами частоти. Хоча HackRF досить хороший для багатьох застосувань, він може не бути оптимальним для задач, де необхідна дуже висока лінійність сигналу або де важливе зниження шуму

на виході. Для таких завдань можуть знадобитися професійні пристрої, як, наприклад, USRP (Universal Software Radio Peripheral)[50,51].

HackRF є потужним і доступним пристроєм для програмованого радіозв'язку, що підтримує широкий частотний діапазон і надає можливість створювати універсальні SDR-системи. Завдяки своїй гнучкості, HackRF є ідеальним інструментом для багатьох застосувань, від радіоаматорських експериментів до професійних досліджень у галузі бездротових технологій. Хоча він має обмеження на точність і потужність сигналу, HackRF є оптимальним вибором для багатьох задач завдяки своїй доступності і можливості керування через програмне забезпечення.

4.4 Результати спектрального сканування

Спектральне сканування є важливою технікою в області радіочастотного моніторингу та аналізу, що дозволяє виявляти активні сигнали в заданому частотному діапазоні. Воно застосовується для дослідження частотного спектра з метою виявлення, ідентифікації та оцінки характеристик різних типів передавачів. Результати спектрального сканування надають інформацію про розподіл потужності сигналу по частотах, дозволяючи визначити не тільки наявність активних сигналів, але й їхні характеристики, такі як тип модуляції, ширина смуги та рівень потужності. У цьому підпункті ми розглянемо процес сканування частотного діапазону, визначення активних сигналів та аналіз їх спектральних характеристик. Завдяки використанню SDR-пристроїв, таких як HackRF, і програмних бібліотек для обробки сигналів, ми можемо отримати точні та детальні результати, що допомагають виявляти навіть слабкі сигнали на фоні шуму. Під час сканування особливу увагу приділено виявленню піків потужності сигналів, що є важливим індикатором для подальшого аналізу типів передавачів та їхніх характеристик. Результати спектрального сканування дозволяють

отримати візуальні та кількісні дані, що використовуються для подальших досліджень. У результаті такого аналізу можна не тільки ідентифікувати активні джерела сигналів, але й оцінити їхні впливи на інші системи радіозв'язку. На Рисунку 4.1 зображена комбінована спектрограма, що відображає спектр сигналу в частотному діапазоні від 95.00 МГц до 110.00 МГц. Графік показує розподіл потужності (в дБ) по частоті (в Гц). Синя лінія на графіку представляє загальний спектр з усіма частотними компонентами сигналу, в той час як червоні кола позначають піки потужності — кандидати на виявлення активних сигналів. Ці піки відповідають максимальним значенням потужності на певних частотах, що можуть вказувати на присутність радіопередавачів або інших активних джерел сигналу в цьому діапазоні. Комбінований спектр дає наочне уявлення про рівень потужності в різних частотних смугах і дозволяє виявляти області з підвищеною активністю, що можуть бути цікавими для подальшого аналізу або моніторингу спектра.

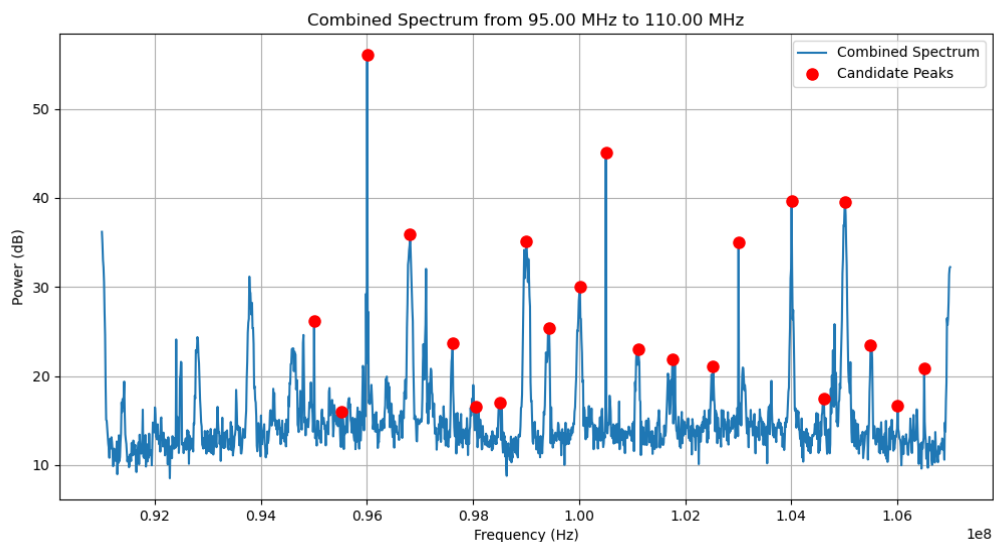


Рисунок 4.1 - Комбінована спектрограма

На Рисунку 4.2 зображена гістограма для частоти 960000 кГц. Гістограма відображає розподіл кількості виявлених сигналів (Hits) по частотах у певному діапазоні. На графіку видно, що найбільше значення

спостерігається на частоті приблизно 9.56 МГц, де сигнал досягає свого піку, що свідчить про наявність активного джерела на цій частоті, ми спостерігаємо чіткі чотири різні значення фази. Це є характерною ознакою QPSK (Quadrature Phase Shift Keying) — модуляції, яка використовує чотири різні фази для передачі двох бітів інформації за один символ. Фази сигналу змінюються чітко, що дозволяє відрізнити різні стани символів. У цьому випадку, кожен стрибок між фазами чітко вказує на те, що сигнал передає один з чотирьох можливих символів, що і є основною характеристикою QPSK.

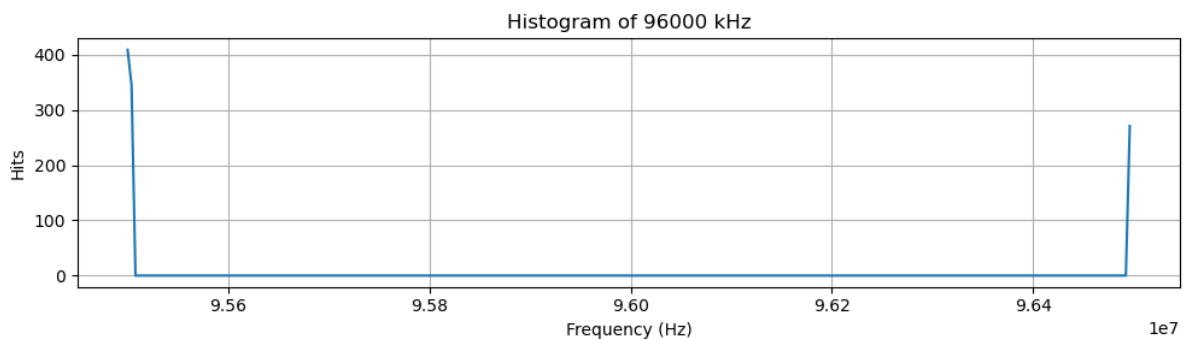


Рисунок 4.2 - Гістограма частоти 960000 kHz

На рисунку 4.3, зображена спектрограма величини сигналу на частоті 95004 кГц. Графік показує зміну потужності сигналу по частотах і часу. Чим яскравіший колір (від жовтого до зеленого), тим вища потужність сигналу. Це дозволяє побачити, як сигнал змінюється з часом і визначити його стабільність та інтенсивність на різних частотах. Спектрограма допомагає візуалізувати частотні коливання і модуляцію сигналу в часі, що є важливим для аналізу його характеристик.

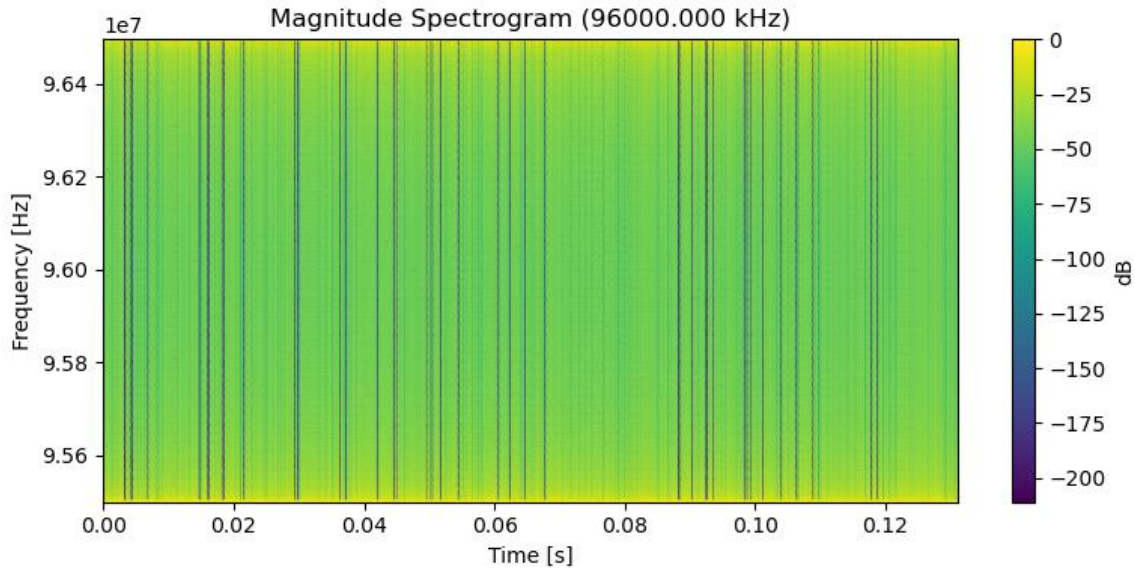


Рисунок 4.3 - Спектрограма величини 960000 kHz

На Рисунку 4.4 зображена гістограма фазових змін для носія 960000 кГц. Гістограма показує розподіл змін фази (в радіанах) для цієї частоти, де видно варіації фази сигналу. Більшість змін фази мають невеликі значення, що свідчить про стабільність сигналу, проте є й більші відхилення, що можуть вказувати на наявність складних модифікацій або перешкод у сигналі.

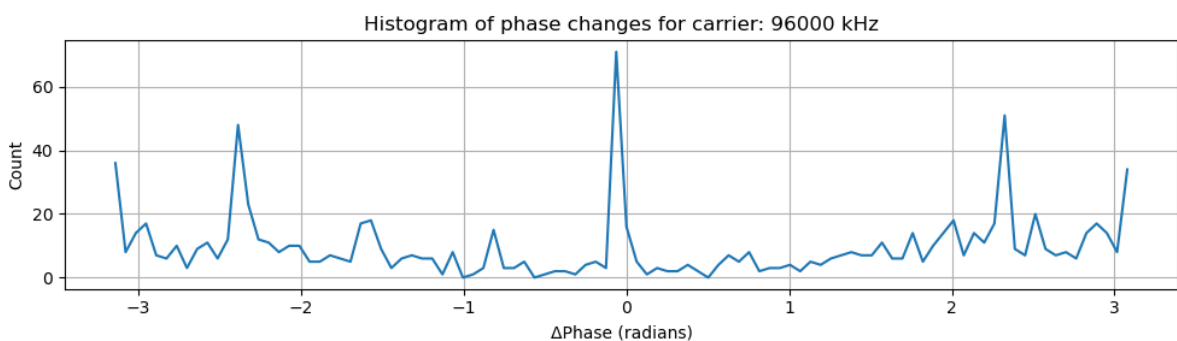


Рисунок 4.4 - Гістограма фазових змін для носія 960000 kHz

На Рисунку 4.5 показана гістограма частоти 100000 кГц. Це графік, який відображає кількість виявлених сигналів у певному частотному діапазоні, і можна побачити, як розподіляються сигнали по частотах в цьому діапазоні. Великий пік на початку вказує на наявність сильного сигналу в конкретному частотному діапазоні, а інші частоти мають значно

менші значення. На графіку видно, що сигнал стрибає між двома різними частотами. Це є типова характеристика FSK (Frequency Shift Keying), де два різні рівні частот використовуються для передачі двійкового сигналу, що означає, що сигнал приймає одне з двох значень: 0 або 1. Чітко виражене "стрибання" між двома частотами свідчить про те, що система використовує зміну частоти для кодованою передачі інформації.

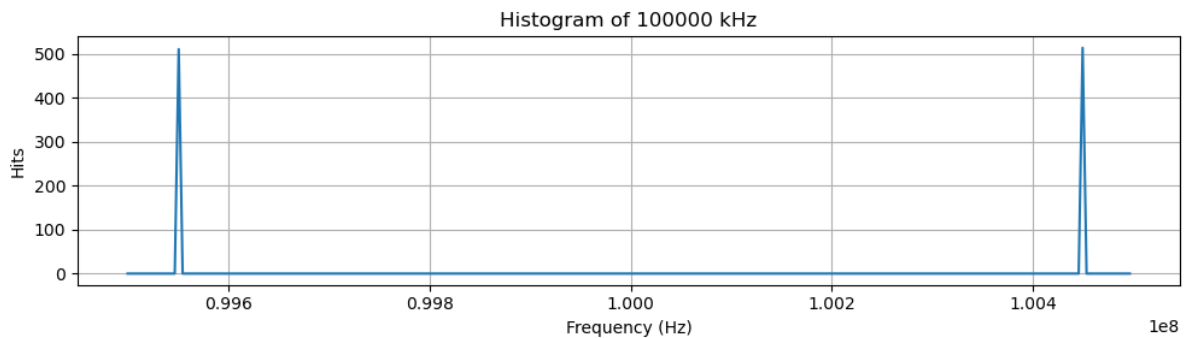


Рисунок 4.5 - Гістограма частоти 100000 kHz

На Рисунку 4.6 зображена гістограма фазових змін для носія 100000 кГц. Як і на попередньому графіку, тут показано, як фаза сигналу змінюється в часі. Цей графік дозволяє оцінити стабільність фази та визначити характер модуляції сигналу, що допомагає виявити перешкоди або нестабільні частоти в системі.

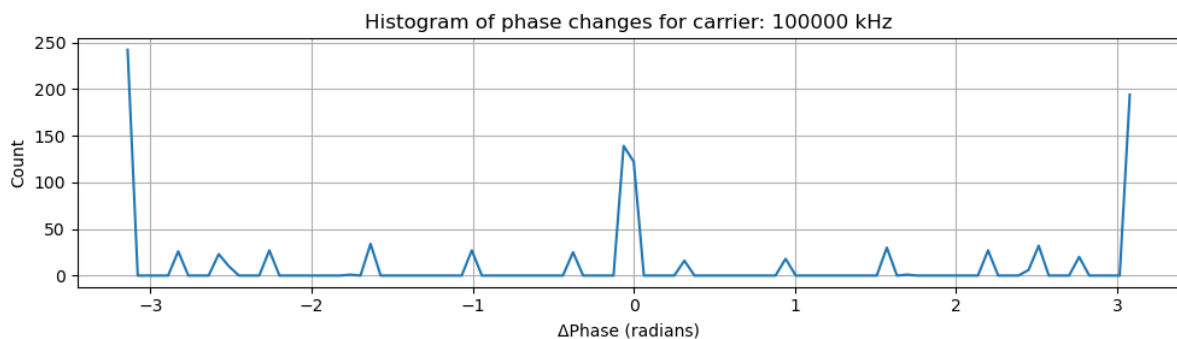


Рисунок 4.6 - Гістограма фазових змін для носія 100000 kHz

На Рисунку 4.7 зображена спектрограма величини для частоти 100000 кГц. Спектрограма показує потужність сигналу (в децибелах) у залежності від частоти та часу. Жовті та зелені кольори вказують на високі рівні потужності сигналу, в той час як блакитний колір відображає низькі

рівні. Це дозволяє оцінити частотну модуляцію та зміну потужності сигналу в часі, що важливо для аналізу стабільності сигналу.

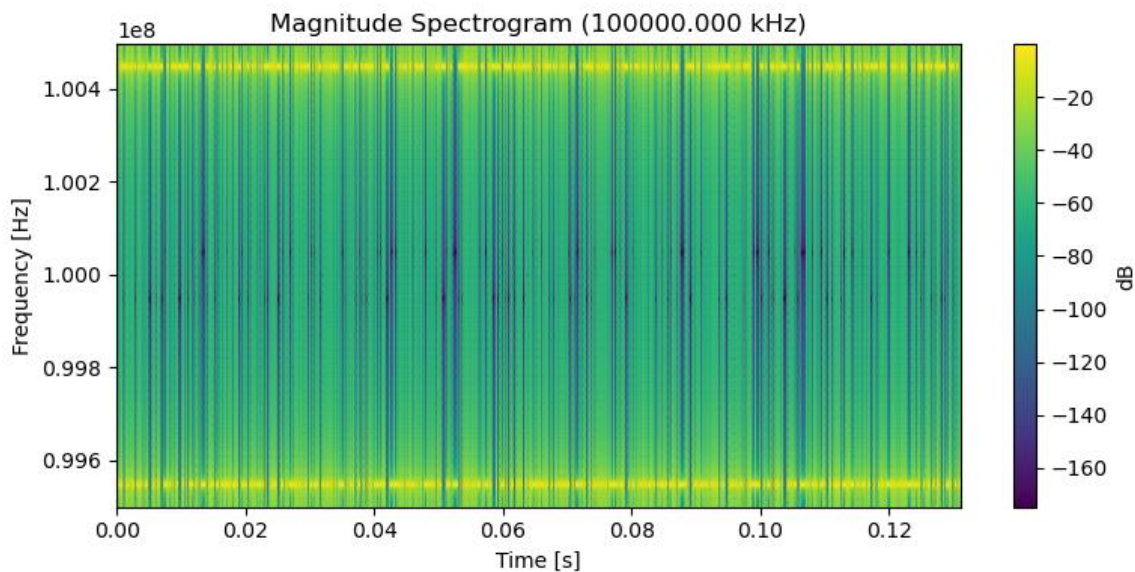


Рисунок 4.7 - Спектрограма величини 100000 kHz

На Рисунку 4.8 зображена гістограма частоти 105000 кГц. Гістограма показує розподіл сигналів по частотах у певному діапазоні. Виявлений пік на частоті близько 0.999 МГц вказує на наявність сильного сигналу в цьому діапазоні, а інші частоти мають значно менший рівень, що свідчить про відсутність активності в цих областях спектра. Графік відображає ASK (Amplitude Shift Keying), де модуляція здійснюється за допомогою зміни амплітуди сигналу. У даному випадку, амплітуда сигналу змінюється в залежності від переданої інформації, що дозволяє відрізняти різні стани. У графіку видно, що фаза сигналу майже не змінюється, а зміни зосереджені саме на амплітуді. Це свідчить про використання ASK, де на різних етапах сигнал або присутній, або відсутній, відповідно до того, чи є передача даних.

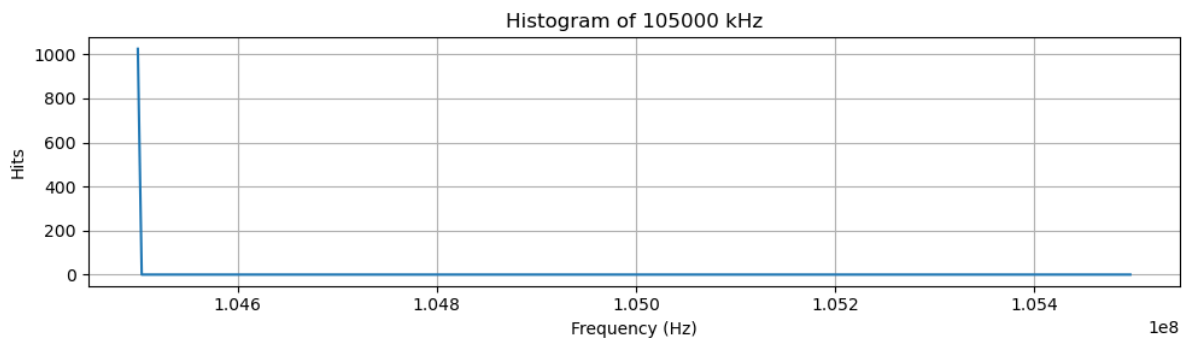


Рисунок 4.8 - Гістограма частоти 105000 кГц

На Рисунку 4.9 зображена гістограма фазових змін для носія 105000 кГц. Графік показує розподіл змін фази (в радіанах) для сигналу на частоті 100014 кГц. Більшість змін фази мають значення, близькі до нуля, що свідчить про стабільність фази сигналу, але також присутні й більш значні зміни фази, що можуть свідчити про модуляцію або перешкоди в системі.

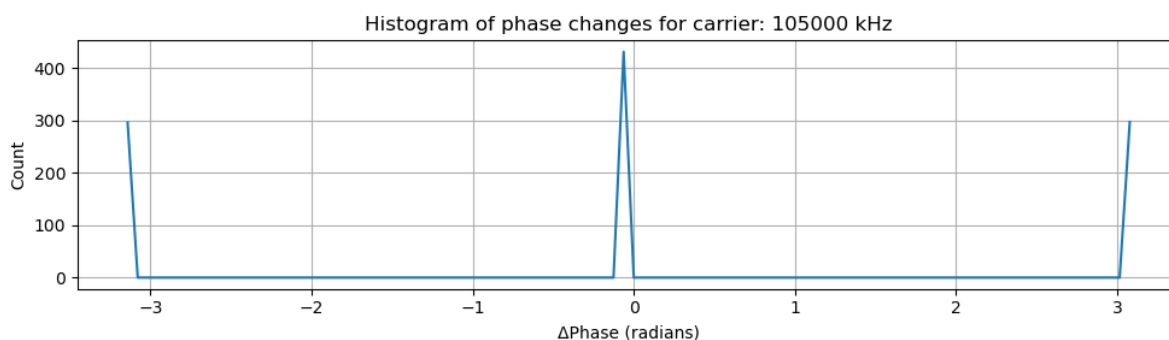


Рисунок 4.9 - Гістограма фазових змін для носія 105000 кГц

На Рисунку 4.10 представлена спектрограма величини для частоти 105000 кГц. Спектрограма показує розподіл потужності сигналу по частотах і часу, де кольори від жовтого до зеленого вказують на високі рівні потужності сигналу, а сині — на низькі. Цей графік дає змогу візуалізувати частотні коливання і модуляцію сигналу в часі, що важливо для оцінки його стабільності та інтенсивності.

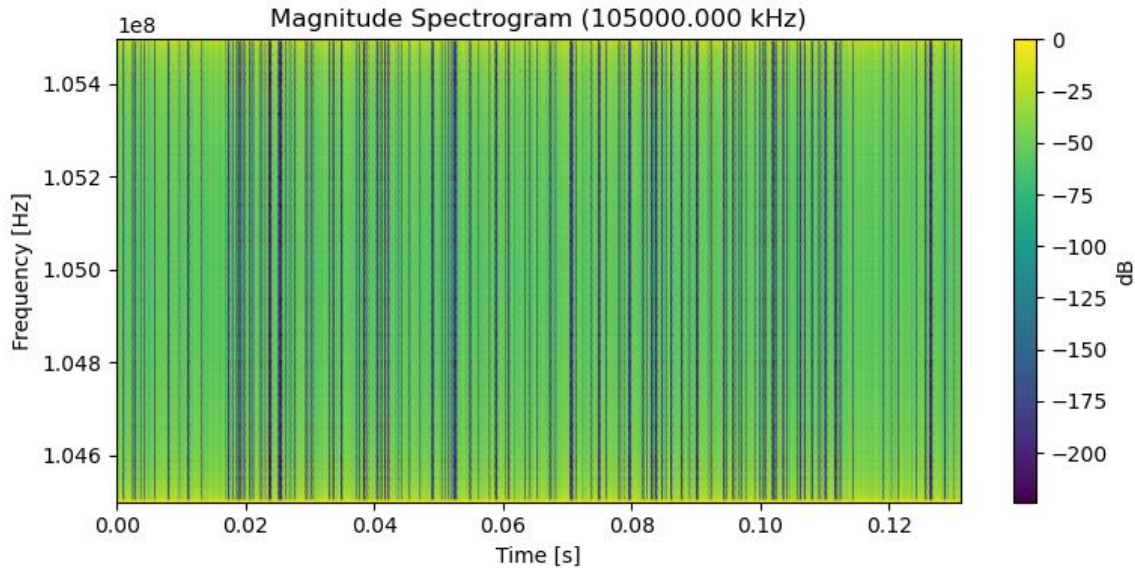


Рисунок 4.10 - Спектрограма величини 105000 kHz

96 МГц демонструє чітку зміну фаз у чотирьох значеннях, що характерно для QPSK, де кожен з чотирьох фазових станів передає два біти інформації. 100 МГц відображає стрибки між двома частотами, що є характерним для FSK, де зміна частоти символізує передачу бітів 0 або 1. 105 МГц показує зміни в амплітуді без значних змін фази, що відповідає ASK, де інформація передається через варіації амплітуди сигналу.

Для розв'язання задач аналізу радіоспектру в реальних умовах важливим є баланс між продуктивністю та розміром пристрою. Потужні сервери забезпечують значну точність і масштабованість, але малогабаритні пристрої на основі NVIDIA Jetson можуть бути більш практичними у польових умовах, забезпечуючи потрібну ефективність при мінімальних розмірах.

1. NVIDIA Jetson Orin Nano

NVIDIA Jetson Orin Nano — це потужний малогабаритний пристрій, що забезпечує до 67 TOPS продуктивності для завдань штучного інтелекту. Він оснащений 1024-ядерним графічним процесором архітектури Ampere з 16 тензорними ядрами, що дозволяє ефективно

виконувати обчислення з використанням CUDA. Цей пристрій ідеально підходить для застосувань, де потрібна висока продуктивність при обмеженому енергоспоживанні.

2. NVIDIA Jetson Orin NX

Jetson Orin NX пропонує ще вищу продуктивність — до 275 TOPS, що робить його підходящим для більш складних завдань штучного інтелекту та обробки даних в реальному часі. Цей пристрій також підтримує використання CUDA та TensorRT, що дозволяє ефективно виконувати глибоке навчання та інші обчислювальні завдання.

3. AMD Radeon Embedded Graphics

AMD пропонує лінійку вбудованих графічних процесорів, які забезпечують оптимальний баланс між продуктивністю та енергоспоживанням. Ці графічні процесори підходять для застосувань, де потрібна висока продуктивність при обмежених ресурсах.

Таблиця 4.1 - Порівняння варіантів

Пристрій	Продуктивність (TOPS)	Підтримка CUDA	Підходить для застосувань
NVIDIA Jetson Orin Nano	до 67	Так	Високопродуктивні завдання з обмеженим енергоспоживанням
NVIDIA Jetson Orin NX	до 275	Так	Складні завдання штучного інтелекту та обробки даних
AMD Radeon Embedded Graphics	Залежить від моделі	Ні	Завдання з високими вимогами до графічної продуктивності

Для розробки малогабаритних детекторів радіоканалів, де важлива висока продуктивність при обмежених розмірах та енергоспоживанні, оптимальним вибором є пристрої серії NVIDIA Jetson. Вони забезпечують необхідну обчислювальну потужність та підтримку технологій CUDA, що дозволяє ефективно реалізувати алгоритми обробки сигналів. У рамках дослідження було запропоновано метод сканування спектру, який дозволяє ефективно визначати частоти можливих каналів передачі сигналу. Сканування спектра на різних частотах дозволяє виявляти активні передавачі та оцінювати їхні характеристики, такі як потужність і ширина смуги. За допомогою цього методу можна здійснювати моніторинг радіочастотного спектра для виявлення перешкод і оцінки використання частотних ресурсів. Також було проаналізовано кілька методів визначення модуляції за спектрограмою. Спектрограма дозволяє візуалізувати частотні коливання і модуляцію сигналу в часі, що є важливим для розпізнавання типів модуляції, таких як AM, FM або PSK. Використання різних методів обробки спектрограм дозволяє здійснювати точний аналіз сигналу і визначати його характеристики. Реалізовано метод виявлення піків спектра і оцінки потужності на різних частотах, що дає змогу автоматично визначати активні частоти і типи передавачів. Цей підхід дозволяє ефективно аналізувати радіосигнали та здійснювати точний моніторинг спектра. Отриманий алгоритм може бути застосований для побудови малогабаритних детекторів радіоканалів. Такі детектори можуть використовуватися для аналізу частотного спектра в реальному часі, виявлення активних каналів зв'язку та оцінки їхніх характеристик у широкому частотному діапазоні.

5 МЕТОДИ ВИЗНАЧЕННЯ ТИПІВ МОДУЛЯЦІЇ

5.1 Класичні методи класифікації цифрових модуляцій на основі спектрограм

Розвиток цифрових комунікацій та потреба в швидких і надійних системах зв'язку призводить до постійної еволюції цифрових обробок і методів цифрового зв'язку. Разом з розвитком мобільних технологій та різноманітних стандартів, що використовуються в системах персональних служб зв'язку (PCS) та безпроводових локальних мережах (WLAN), виникає необхідність у створенні багатосмугових і багатофункціональних радіосистем, таких як програмне радіо (software radio). Важливу роль у таких системах відіграє автоматична класифікація модуляцій, що дозволяє вибрати відповідний демодулятор. Це знайшло своє застосування не лише в цивільних, але й у військових технологіях, а також у моніторингу та аналізі загроз. Метою автоматичного розпізнавання типу цифрової модуляції є точне визначення використовуваного методу передачі даних без необхідності попереднього знання параметрів сигналу. Для цієї мети було обрано кілька класичних типів модуляцій, таких як ASK, BPSK, QPSK, FSK, QAM та MSK, які активно використовуються в сучасних радіотелекомунікаційних системах, таких як GSM, WiFi та WiMAX. Метод класифікації, описаний у статті, ґрунтується на спектрограмному аналізі сигналів. Для класифікації типів модуляцій автори використовували спектрограми сигналів, отримані за допомогою Matlab.

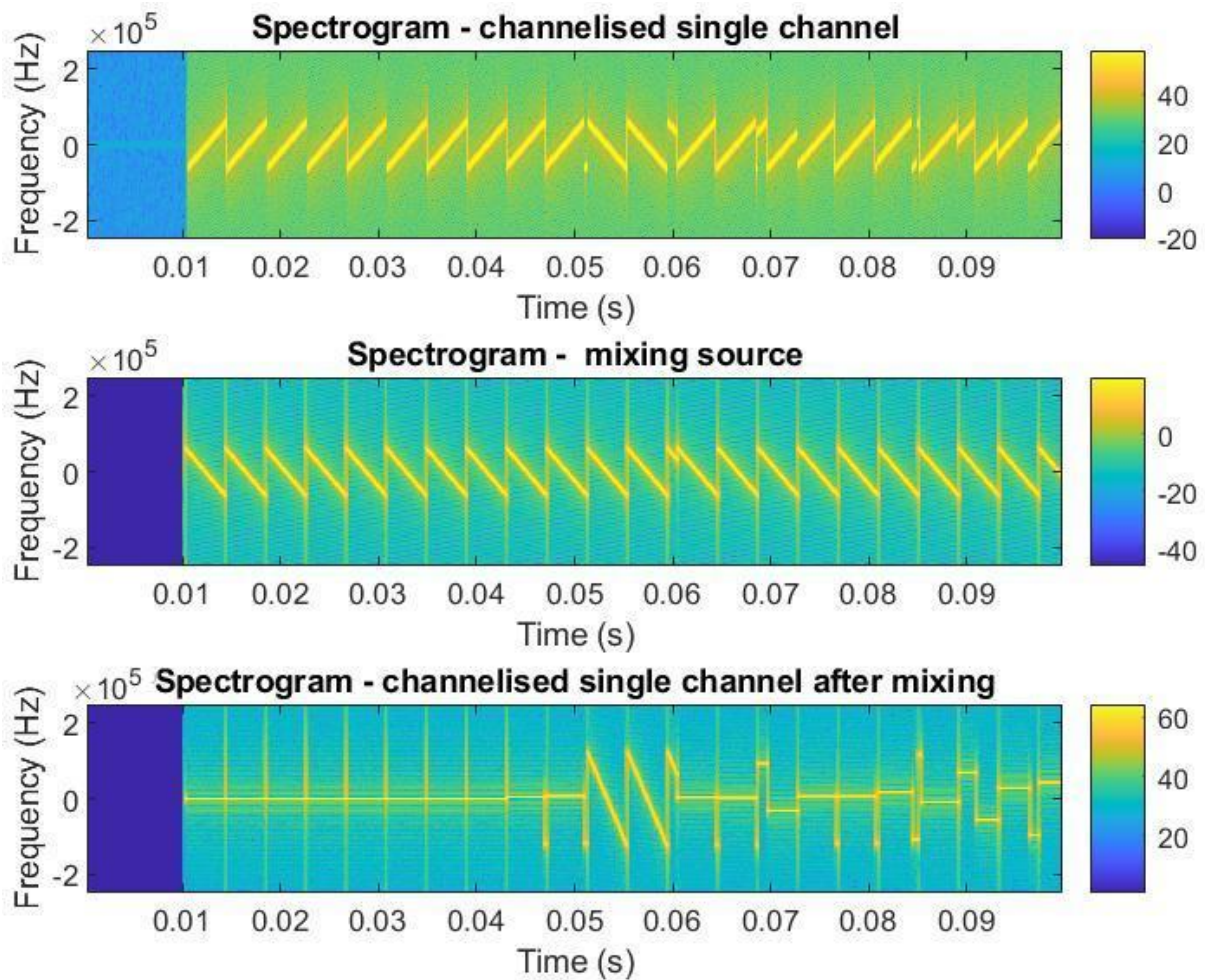


Рисунок 5.1 - Спектрограма сигналу Lora

Спектрограма сигналу LoRa (Long Range) відображає амплітуду сигналу в часі та частоті, дозволяючи побачити, як енергія сигналу розподіляється по різних частотах протягом часу. Вигляд спектрограми сигналу LoRa можна описати як широку смугу, що змінюється по частоті, з періодичними змінами частоти (характерними для CSS). На спектрограмі можна побачити кілька пік (відображення перехідних моментів між різними частотами), які «розтягуються» на широкому діапазоні частот. Спектрограма сигналу LoRa є корисним інструментом для аналізу ефективності та можливостей цієї технології у реальних умовах передачі даних, особливо для виявлення перешкод, інтерференцій та оптимізації використання частотного спектра.

Спектрограми для кожного з типів модуляцій (FSK, MSK, BPSK, QPSK, QAM16) були детально проаналізовані з огляду на їх модуляційні та фазові характеристики. Окремі сегменти сигналів використовувалися для спектрального аналізу, причому розмір сегмента був підібраний так, щоб він містив лише елементи з однаковими значеннями. Для MSK та FSK модуляцій важливою характеристикою є наявність двох окремих несучих частот, що легко виявляється за допомогою методу пошуку максимальних значень у спектрограмі модуля. Для аналізу фазових спектрограм було застосовано підрахування різниць фаз між сусідніми символами на частоті несучої. Спектрограми для кожного з типів модуляцій були поділені на два основних типи: модуляційні спектрограми, які оцінюють амплітудні характеристики, і фазові спектрограми, що використовуються для розпізнавання фазових станів модуляцій. Для виявлення FSK та MSK модуляцій застосовувалась техніка пошуку максимумів у відповідних колонках спектрограми. FSK була найбільш легкою для класифікації завдяки чітко вираженим двом несучим частотам. Для BPSK, QPSK та QAM-16 використовувалась більш складна техніка, оскільки ці модуляції мають кілька фазових значень. Для кожної з цих модуляцій на фазових спектрограмах визначалися максимальні кількості фазових переходів, що відповідали кількості можливих фазових значень: два для BPSK, чотири для QPSK та одинадцять для QAM-16.

Додатковий білий гаусів шум, який додається до сигналів з різними модуляціями, призводить до згладжування різких переходів у спектрограмах. Однак основні властивості сигналів залишалися помітними навіть при рівні сигнал-шуму (SNR) близько 0 дБ. Спектрограми модуляційних і фазових характеристик продовжували чітко вказувати на тип модуляції навіть після корупції сигналів шумом. Моделювання реального каналу, що включав фазовий шум, багатошляхове згасання Рейліївського типу та білий гаусів шум, показало, що цей метод ефективно

працює в умовах реальних передач. Симуляції, проведені для кожного з типів модуляцій при різних рівнях шуму та згасання, продемонстрували, що метод класифікації здатен точно визначати тип модуляції навіть після проходження сигналу через складні канали. Запропонований метод класифікації цифрових модуляцій на основі аналізу спектрограм є ефективним інструментом для автоматичного розпізнавання типів модуляцій у реальних умовах передачі. Він здатний точно класифікувати сигнали з FSK, BPSK, QPSK, MSK та QAM-16, навіть за умов наявності білого шуму та каналів з багатошляховим згасанням. Аналіз гістограм спектрограм дозволяє точно виявляти характерні параметри кожної модуляції, що робить цей метод перспективним для використання в сучасних системах бездротового зв'язку.

У статті "Adaptive modulation recognition based on evolutionary algorithms" представлений новий підхід до автоматичного розпізнавання модуляцій (AMR), який базується на кластеризації сигналів. Метою роботи є розробка методу, який ефективно відрізняє багатоканальні модуляції, зокрема OFDM (Orthogonal Frequency-Division Multiplexing), від одноносінкових модуляцій (Single-Carrier Modulations). Для класифікації використовуються статистичні характеристики амплітуди прийнятих сигналів на виході квадратичного мікзера як ключові ознаки. Ці ознаки з навчальних даних подаються в алгоритм кластеризації, що дозволяє оцінити центроїди для багатоканальних і одноносінкових модуляцій. Аналіз типу модуляції сигналу є важливою задачею для будь-якої комунікаційної системи. Автоматичне розпізнавання модуляцій має велике значення в багатьох сучасних технологіях, таких як програмовані радіосистеми, когнітивні радіо та військові комунікаційні системи. Це важливо для моніторингу спектра, адаптивних комунікацій, а також для забезпечення безпеки та ефективності передачі даних.

Існуючі алгоритми AMR можна поділити на два основні підходи: (1) методи, засновані на ознаках (Feature-Based, FB), та (2) методи, засновані на ймовірнісних моделях (Likelihood-Based, LB). Перший підхід передбачає класифікацію сигналів на основі виділених характеристик, таких як миттєві параметри, вищі статистики, спектральні ознаки тощо. Методи FB знижують обчислювальну складність і часто використовуються через свою простоту реалізації, однак можуть призводити до субоптимальних рішень. В останні роки спостерігається значне зростання використання багатоканальних модуляцій, зокрема OFDM, у таких застосунках, як DVB, WLAN, WMAN тощо.

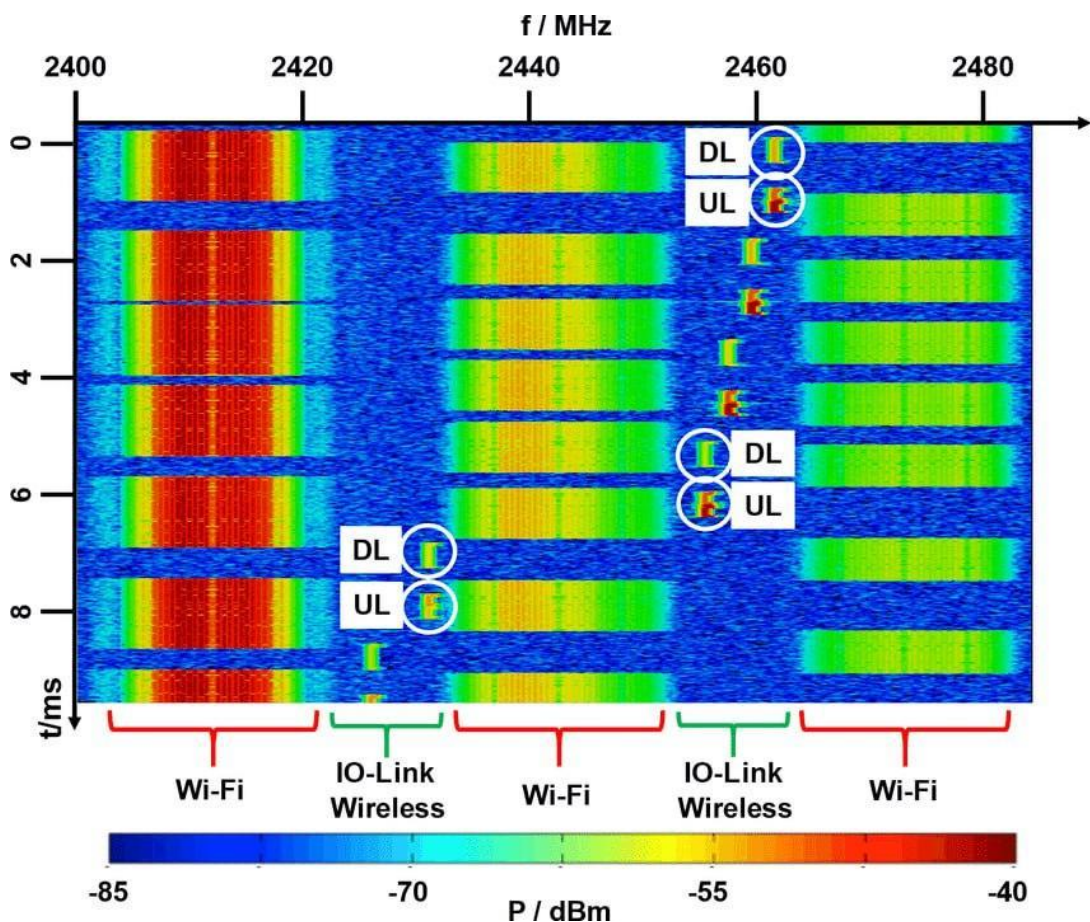


Рисунок 1.5 - Спектрограма wifі сигналу

Однак, завдяки складності цих модуляцій, розпізнавання OFDM серед одноносінкових модуляцій залишається викликом. Відомі підходи до класифікації OFDM включають використання тестів на гауссовість,

методи на основі кумулянтів вищих порядків, трансформати Вейвлетів та енергетичні характеристики спектра. Однак ці методи часто стикаються з проблемами обчислювальної складності, низькою надійністю при низьких SNR, а також обмеженнями, пов'язаними з моделями каналів. Запропонований підхід до класифікації, який використовує модифікований алгоритм К-середніх на основі еволюційних алгоритмів, дозволяє ефективно класифікувати багатоканальні модуляції, такі як OFDM, навіть за умов низького SNR.

5.2 Реалізований метод

Метод сканування радіочастотного спектра для виявлення потенційних каналів з обробкою отриманих даних складається з кількох важливих етапів. Цей метод включає в себе збір сигналів з радіоприймача, їх обробку та аналіз, а також візуалізацію отриманих результатів для виявлення частотних пік, що відповідають активним каналам.

Покроковий опис методу отримання сигналу:

На початковому етапі сканування радіочастотного спектра використовується програмно кероване радіо (SDR, Software Defined Radio). Одним із таких пристроїв є `hackrf`, який здатний приймати сигнали на широкому діапазоні частот. У нашому випадку пристрій налаштовується на бажану центральну частоту сканування. Це важливий крок, адже він визначає початок діапазону частот, який ми будемо аналізувати. Параметри, які задаються для пристрою:

- Частота центру (`center_freq`) — це центральна частота, навколо якої ми будемо здійснювати сканування спектра. Вона визначає точку, навколо якої здійснюється прийом сигналу.

- Частота дискретизації (`sample_rate`) — це кількість точок в секунду, яку ми отримуємо з пристрою. Висока частота дискретизації дозволяє точніше аналізувати сигнал, але збільшує вимоги до обробки даних.
- Кількість вибірок (`num_samples`) — це загальна кількість точок, які будуть використані для побудови спектра. Це важливий параметр для визначення роздільної здатності аналізованого сигналу.

Дані налаштування дозволяють нам отримати потрібну кількість даних для аналізу за заданий період часу.

Після налаштування апаратного забезпечення за допомогою `osmosdr.source` створюється об'єкт, який дозволяє зчитувати сигнал з радіоприймача. Пристрій налаштовується на бажану частоту центру, а також визначається частота дискретизації і кількість вибірок, які будуть зібрані. Сигнал, що приходить з пристрою, є складним, тобто він складається з реальної та уявної частин. Для зручності обробки цей сигнал зберігається в комплексній формі (реальна частина та уявна частина) і передається для подальшої обробки.

Для того, щоб проаналізувати сигнали на різних частотах, необхідно виконати перетворення Фур'є (FFT). Перетворення Фур'є дозволяє перевести сигнал з часової області в частотну, де можна оцінити потужність сигналу на різних частотах. Це важливо для визначення наявності активних частот у спектрі. Перетворення Фур'є дозволяє:

- Побудувати амплітудний спектр, де кожен пік в спектрі буде вказувати на наявність потужного сигналу на конкретній частоті.
- Оцінити потужність сигналу на кожній частоті в децибелах (dB), що дозволяє чітко побачити, на яких частотах сигнал найбільш виразний.

Далі, на основі отриманого спектра можна визначити, де є значні пік (потужні сигнали), які вказують на потенційно активні канали.

Після отримання спектра застосовуються фільтраційні методи для покращення якості аналізу та зменшення впливу шуму. Зокрема, для згладжування потужності використовується сгладжування за допомогою фільтра Савицького-Голея (Savitzky-Golay filter). Це дозволяє зменшити вплив короткочасових коливань і шуму, підвищуючи точність виявлення реальних пік. Фільтрація потужності допомагає чітко відділити реальні пік, які можуть бути каналами передачі, від фонових шумів. Також обчислюється медіана потужності, щоб оцінити рівень шуму в спектрі. Це дозволяє встановити поріг для виявлення реальних пік на основі потужності сигналу. Для виявлення активних каналів у спектрі використовуються методи пошуку пік. Пошук пік дозволяє знайти частоти, на яких спостерігаються значні збурення (піки потужності), що вказує на активні сигнали. Під час пошуку пік застосовуються додаткові параметри:

- Мінімальна та максимальна ширина каналу. Ці параметри допомагають виявити канали, які мають певну ширину (наприклад, вузькосмугові канали чи широкосмугові).
- Поріг шуму. Для того, щоб піки були реальними, їх амплітуда повинна перевищувати певний рівень, що визначається через медіану потужності та поріг шуму.

У результаті цього етапу ми отримуємо список частот, на яких виявлено активні сигнали. Це дозволяє ідентифікувати потенційні канали в радіочастотному спектрі. Для кращого розуміння результатів сканування використовуються графічні методи візуалізації, такі як спектрограми та гістограми. Спектрограми показують, як змінюється потужність сигналу по частоті та часу. Це дозволяє побачити активність сигналів у різних ділянках спектра, а також оцінити їх інтенсивність. На основі гістограм можна оцінити частоти, на яких спостерігається найбільша кількість пік.

Це допомагає виявити найбільш популярні частоти для каналів передачі даних.

Після того як пік виявлені і проаналізовані, можна отримати кандидатів на канали, тобто частоти, які можуть бути потенційно активними. Ці канали можуть бути використані для подальшого аналізу модуляцій, ширини каналу або якості зв'язку. Канали, які потрапили в діапазон, можуть бути використані для подальших операцій, таких як оцінка типів модуляції, що застосовуються для передачі сигналів. Отже, метод сканування радіочастотного спектра включає кілька етапів:

1. Налаштування апаратного забезпечення (програмно-керовані радіо, такі як `hackrf`).
2. Збір сигналів з приймача та перетворення їх у цифрові дані.
3. Перетворення сигналу в частотну область за допомогою перетворення Фур'є.
4. Фільтрація та обробка потужності для видалення шуму.
5. Виявлення пік на основі потужності спектра для ідентифікації каналів.
6. Візуалізація результатів (спектрограми, гістограми).
7. Вибір потенційних каналів для подальшого аналізу типу модуляції та інших властивостей.

ВИСНОВКИ

Розвиток мобільних комунікаційних систем, а також підвищення потреби в моніторингу та аналізі радіоспектра сприяють створенню більш ефективних систем для автоматизованого виявлення та ідентифікації активних радіоканалів. Це особливо важливо для забезпечення безпеки та оптимізації спектра в умовах зашумленого середовища. Використання технологій програмного радіо та апаратних платформ малого розміру, таких як NVIDIA Jetson, дозволяє розробляти системи для реального часу, що дає змогу ефективно працювати з великими обсягами даних при обмежених обчислювальних потужностях. У ході виконання роботи було розроблено метод автоматизованого спектрального сканування радіосигналів у широкому діапазоні частот з метою виявлення та ідентифікації активних радіоканалів. Метод ґрунтується на застосуванні адаптивного порогу виявлення, визначеного на основі медіанного рівня шуму, що дозволяє забезпечити надійне функціонування в умовах зашумленого середовища. Для підвищення точності аналізу спектральних характеристик було використано фільтрацію спектру за допомогою фільтра Савицького–Голея, яка забезпечує ефективне згладжування спектрограм та зменшення впливу шумових складових. Основою апаратної реалізації системи є SDR-пристрій HackRF, що забезпечує захоплення сигналів та їх подальшу обробку. Було розроблено програмну архітектуру, яка включає:

1. Модулі керування SDR-приймачем.
2. Блоки обробки сигналів у частотній області.
3. Засоби GPU-прискореного формування спектрограм та фазограм.

4. Алгоритми автоматичного виявлення активних частот та побудови спектрограм.
5. Засоби аналізу спектральних і фазових характеристик для ідентифікації типу модуляції сигналів.

Запропоновано підхід до сканування радіоспектру, що включає фіксацію сигналів, їх спектральний аналіз, визначення активних частот шляхом виявлення піків у спектрі та їх класифікацію за спектральними ознаками. Обробка здійснюється із використанням CUDA-прискорення, що дозволяє реалізувати систему на малопотужних обчислювальних платформах, таких як NVIDIA Jetson. Результати дослідження свідчать про можливість ефективного виявлення та ідентифікації радіоканалів у реальному часі при обмежених ресурсах. Такий підхід може бути основою для побудови малогабаритних детекторів радіоканалів, що працюють автономно, зокрема у складі мобільних або польових систем.

Основні результати роботи:

1. Розроблено алгоритм сканування частотного діапазону, який дозволяє виділяти діапазони з потенційною наявністю радіоканалів, що забезпечує більш точне і швидке виявлення активних сигналів.
2. Удосконалено метод адаптивної порогової функції для виявлення сигналів у зашумленому середовищі.
3. Для поліпшення точності спектрального аналізу було застосовано фільтрацію спектру за допомогою фільтра Савицького–Голея, що забезпечує ефективне згладжування спектрограм та зменшення впливу шумових складових на результат.
4. Проаналізовані методи ідентифікації модуляцій сигналів на основі спектрограм. Особлива увага приділена використанню спектральних

і фазових характеристик для точного визначення типів модуляцій, таких як QPSK, FSK, ASK.

5. Реалізовано метод автоматизованого спектрального сканування, що включає адаптивне визначення порогів і класифікацію активних частот за спектральними ознаками.
6. Побудовано вимірювальний стенд, на якому були протестовані розроблені алгоритми та методи. Стенд забезпечує реальний процес виявлення та класифікації активних радіоканалів в умовах реального часу.
7. Запропоновані рішення можуть бути використані для створення спектроаналізаторів на обчислювальних платформах малої потужності, таких як NVIDIA Jetson, що відкриває можливості для застосування цієї технології в мобільних та польових системах для моніторингу радіосигналів.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Design and Analysis of New Digital Modulation classification method .
URL:<https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=bed31707bd54704c8f5f13b60d1a9418f51bc93b>
2. Adaptive modulation recognition based on the evolutionary algorithms.
URL:<https://www.sciencedirect.com/science/article/abs/pii/S1568494616300795>
3. Modulation recognition with pre-denoising convolutional neural network.
URL:<https://ietresearch.onlinelibrary.wiley.com/doi/epdf/10.1049/el.2019.3586>
4. Fully Dense Neural Network for the Automatic Modulation Recognition.
URL:<https://scispace.com/papers/fully-dense-neural-network-for-the-automatic-modulation-32g2frn8qq>
5. Automatic modulation recognition of radiation source signals based on two-dimensional data matrix and improved residual neural network.
URL:<https://www.sciencedirect.com/science/article/pii/S2214914723001885>
6. Ghunsar, M., & Marriwala, N. K. (2019). A review on automatic signal classification techniques for software defined radios.
7. Oncu, S., Karakaya, M., Dalveren, Y., & Kara, A. (2024).
8. J. Jagannath, H. M. Saarinen and A. L. Drozd, "Framework for automatic signal classification techniques (FACT) for software defined radios,".
9. Soltani, S., Sagduyu, Y. E., Hasan, R., & Davaslioglu, K. (2019).
10. Real-time and embedded deep learning on FPGA for RF signal classification. Xu, T., & Darwazeh, I. (2019).

11. O'Shea, T. J., Hoydis, J. (2017). An Introduction to Deep Learning for the Physical Layer.
12. IEEE Transactions on Cognitive Communications and Networking. Krizhevsky, A., Sutskever, I., & Hinton, G. (2012).
13. O'Shea, T. J., & Corgan, J. (2016). Deep Learning Radio Signal Classification with Convolutional Neural Networks. LSTMs and Machine Learning Techniques.
14. IEEE Journal on Selected Areas in Communications. Reuther, A., Kepner, J., Byun, C., Samsi, S., Arcand, W., Bestor, D., & Bergeron, B. (2018). Survey of Machine Learning Algorithms for FFT-Based Signal Classification.
15. Gravelle, C., & Zhou, R. (2019). SDR Demonstration of Signal Classification in Real-Time Using Deep Learning.
16. 2019 IEEE Globecom Workshops (GC Wkshps). DOI:10.1109/GCWkshps45667.2019.9024661.
17. Bhatnagar, V., Ouedraogo, G. S., Gautier, M., & Carer, A. (2013). An FPGA Software Defined Radio Platform with a High-Level Synthesis Design Flow.
18. VTCAt: Dresden, Germany. DOI:10.1109/VTCSpring.2013.6691879.
19. Huang, Y., Tang, C., Duan, H.-L., & Zhou, Y. (2013). Efficient Time Synchronization Approach for Wireless Communication Systems on GPP-Based Software-Defined Radio Platform. Journal of Computer Science and Technology, 28(3), 387-396. DOI:10.1007/s11390-013-1344-2
20. Levy, J., Naitzat, A., & Zeevi, Y. Y. (2022). Classification of audio signals using spectrogram surfaces and extrinsic distortion measures.
21. EURASIP Journal on Advances in Signal Processing, 2022(1), 1–12. Sassi, M., Pecchia, L., & Haleem, M. S. (2024).

22. Spectrogram-based approach with convolutional neural network for human activity classification. *IFMBE Proceedings*, 1(40), 1–7. https://doi.org/10.1007/978-3-031-49068-2_40
23. Liu, W., Pareit, D., De Poorter, E., & Moerman, I. (2013). Advanced spectrum sensing with parallel processing based on software-defined radio.
24. *IEEE Transactions on Wireless Communications*, 12(12), 1-10. <https://doi.org/10.1109/TWC.2013.082013>.
25. Tan, L. (2020). *Digital Signal Processing: Fundamentals and Applications*. Academic Press.
- Lyons, R. G. (2011). *Understanding Digital Signal Processing*. Pearson Education.
26. Dick, C. (2015). *Real-time Digital Signal Processing: Implementations and Applications*. Springer.
27. Bedrosian, P., & Smith, J. (2018). Design and analysis of new digital modulation classification method. CiteSeerX. <https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=bed31707bd54704c8f5f13b60d1a9418f51bc93b>
28. Zhang, Z., & Zhou, Y. (2016). Adaptive modulation recognition based on the evolutionary algorithms. ScienceDirect. <https://www.sciencedirect.com/science/article/abs/pii/S1568494616300795>
29. Zhang, L., & Zhang, C. (2019). Spectrogram-based automatic modulation recognition using convolutional neural network. *IET Research Journals*. <https://ietresearch.onlinelibrary.wiley.com/doi/epdf/10.1049/el.2019.3586>
30. Liu, Z., & Wang, J. (2020). Fully dense neural network for the automatic modulation recognition. SciSpace. <https://scispace.com/papers/fully-dense-neural-network-for-the-automatic-modulation-32g2frn8qq>
31. Li, Y., Li, J., & Wu, D. (2023). Automatic modulation recognition of radiation source signals based on two-dimensional data matrix and

- improved residual neural network. ScienceDirect.
<https://www.sciencedirect.com/science/article/pii/S2214914723001885>
32. Zhang, F., Luo, C., Xu, J., & Luo, Y. (2022). Deep Learning Based Automatic Modulation Recognition: Models, Datasets, and Challenges
 33. Jafarigol, E., Alaghband, B., Gilanpour, A., Hosseinipoor, S., & Mirmozafari, M. (2025). AI/ML-Based Automatic Modulation Recognition: Recent Trends and Future Possibilities.
 34. Zhang, L., & Zhang, C. (2019). Spectrogram-based automatic modulation recognition using convolutional neural network. IET Research Journals.
 35. Modulation Classification With Deep Learning, Mathworks, Natick, MA, USA, 2019.
 36. Li, Y., Li, J., & Wu, D. (2023). Automatic modulation recognition of radiation source signals based on two-dimensional data matrix and improved residual neural network. ScienceDirect.
 37. O'Shea, T. J., & Hoydis, J. (2017). An introduction to deep learning for the physical layer. IEEE Transactions on Cognitive Communications and Networking.
 38. S. Ramjee, S. Ju, D. Yang, X. Liu, A. El Gamal, and Y. C. Eldar, "Fast deep learning for automatic modulation classification," 2019, arXiv:1901.05850. [Online]. Available: <http://arxiv.org/abs/1901.05850>
 39. Teng, C.-F., Liao, C.-C., Chen, C.-H., & Wu, A.-Y. (2020). Polar feature based deep architectures for automatic modulation classification considering channel fading. IEEE Transactions on Vehicular Technology.
 40. Y. Wang, M. Liu, J. Yang, and G. Gui, "Data-driven deep learning for automatic modulation recognition in cognitive radios," IEEE Trans. Veh. Technol., vol. 68, no. 4, pp. 4074–4077, Apr. 2019.
 41. C. Wang, J. Wang, and X. Zhang, "Automatic radar waveform recognition based on time-frequency analysis and convolutional neural

- network,” in Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP), Mar. 2017, pp. 2437–2441.
42. Li, X., & Liu, J. (2023). An Efficient CNN-based Modulation Classification for Wireless Communication Systems. *Sensors*.
 43. L. Ma, Y. Yang, and H. Wang, “DBN based automatic modulation recognition for ultra-low SNR RFID signals,” in Proc. 35th Chin. Control Conf. (CCC), Jul. 2016, pp. 7054–7057
 44. G. E. Hinton, S. Osindero, and Y.-W. Teh, “A fast learning algorithm for deep belief nets,” *Neural Comput.*, vol. 18, no. 7, pp. 1527–1554, Jul. 2006.
 45. F. Wang, S. Huang, H. Wang, and C. Yang, “Automatic modulation classification exploiting hybrid machine learning network,” *Math. Problems Eng.*, vol. 2018, pp. 1–14, Dec. 2018.
 46. S. Kharbech, I. Dayoub, E. P. Simon, and M. Zwingelstein-Colin, “On classifiers for blind feature-based automatic modulation classification over multiple-input–multiple-output channels,” *IET Commun.*, vol. 10, no. 7, pp. 790–795, May 2016.
 47. Y. Lin, X.-C. Xu, and Z.-C. Wang, “New individual identification method of radiation source signal based on entropy feature and SVM,” *J. Harbin Inst. Technol.*, vol. 21, no. 1, pp. 98–101, 2014.
 48. M. W. Aslam, Z. Zhu, and A. K. Nandi, “Automatic modulation classification using combination of genetic programming and KNN,” *IEEE Trans. Wireless Commun.*, vol. 11, no. 8, pp. 2742–2750, 2012.
 49. Y. Faquan, L. Zan, L. Hongyan, and H. Haiyan, “A specific combination scheme for modulation identification of mixed modulation signal,” *Sensors Verb Transducers*, vol. 170, no. 5, pp. 153–159, 2014.
 50. Yang, T., Huang, H., & Li, J. (2021). Automatic modulation recognition using convolutional recurrent neural networks. *IEEE Transactions on Wireless Communications*.

51. D. S. Chirov, A. N. Vynogradov, and E. O. Vorobyova, “Application of the decision trees to recognize the types of digital modulation of radio signals in cognitive systems of HF communication,” in Proc. Syst. Signal Synchronization, Generating Process. Telecommun. (SYNCHROINFO), Jul. 2018, pp. 1–6.
52. Чопик, В. В. Аналізатор спектрограм радіочастотних випромінювань у реальному масштабі часу : дипломна робота ... бакалавра : 153 Мікро- та наносистемна техніка / Чопик Владислав Вікторович. – Київ, 2024. – 103 с.

ДОДАТОК А (Фрагменти коду)

```
import osmosdr
from math import floor
import structlog
import numpy as np
from gnuradio import gr, blocks, analog
import matplotlib.pyplot as plt
from scipy.signal import find_peaks as sci_find_peaks,
savgol_filter, resample_poly
import cupyx.scipy.signal as cu_signal
import cupy

structlog.configure(processors=[structlog.processors.JSONRende
rer()])
logger = structlog.get_logger()

def median_truncate(data: np.ndarray, div_factor: int) ->
np.ndarray:
    out = np.zeros(len(data) // div_factor, dtype=data.dtype)
    sg_size = int(len(data) / div_factor) # segment size
    for i in range(len(out) - 1):
        out[i:i + sg_size] = np.median(data[i:i + sg_size])

    return out

class RfScanner(gr.top_block):
    source: osmosdr.source
    agc: analog.agc_cc
    head: blocks.head
    sink: blocks.vector_sink_c

    center_freq: float
    sample_rate: float
    num_samples: int

    def set_center_freq(self, center_freq):
        self.source.set_center_freq(center_freq)
        self.center_freq = center_freq
```

```

    def __init__(self, center_freq: float, sample_rate: float,
num_samples: int):
        gr.top_block.__init__(self)
        self.center_freq = center_freq
        self.sample_rate = sample_rate
        self.num_samples = num_samples

        self.source = osmosdr.source(args='hackrf')
        self.source.set_center_freq(self.center_freq)
        self.source.set_sample_rate(self.sample_rate)
        # self.source.set_gain(20)
        self.source.set_if_gain(16) # default from official
docs. No more than 47
        self.source.set_bb_gain(16)
        self.source.set_gain(0)

self.lpf = filter.fir_filter_ccf(1, taps)

        self.head = blocks.head(gr.sizeof_gr_complex,
self.num_samples)
        self.sink = blocks.vector_sink_c()

        self.connect(self.source, self.head, self.sink)

    def get_samples(self):
        return np.array(self.sink.data(), dtype=np.complex64)

def scan_frequency_range(start_freq: float,
                        stop_freq: float,
                        step: float,
                        sample_rate: float,
                        num_samples: int,
                        min_bandwidth: float,
                        max_bandwidth: float,
                        noise_above: float = 0.05,
                        visualize: bool = True) ->
list[float]:
    # detected_transmissions = []
    all_freqs_list = []

```

```

all_power_list = []
i = 0
for freq in np.arange(start_freq, stop_freq, sample_rate):
    scanner = RfScanner(float(start_freq), sample_rate,
num_samples)
    scanner.start()
    scanner.set_center_freq(freq)
    scanner.wait()

    samples = scanner.get_samples()
    #
    # np.save(f'samples/samples_{i}.npy', samples,
allow_pickle=True)
    # samples = np.load(f'samples/samples_{i}.npy')
    # i += 1

    scanner.stop()

    fft_data = np.fft.fftshift(np.fft.fft(samples))
    power_db = 20 * np.log10(np.abs(fft_data) + 1e-12)
    fft_bins =
np.fft.fftshift(np.fft.fftfreq(len(samples), 1.0/sample_rate))

    abs_freqs = freq + fft_bins

    all_freqs_list.append(abs_freqs)
    all_power_list.append(power_db)

all_freqs = np.concatenate(all_freqs_list)
all_power = np.concatenate(all_power_list)
sort_idx = np.argsort(all_freqs)
all_freqs_sorted = all_freqs[sort_idx]
all_power_sorted = all_power[sort_idx]

noise_median = float(np.median(all_power))

del all_freqs
del all_power

# windowed_freqs =
shrink_array_by_median(all_freqs_sorted, 100)
# windowed_power =
shrink_array_by_median(all_power_sorted, 100)

```

```

    windowed_freqs = all_freqs_sorted.reshape(-1,
1000).mean(axis=1)
    # windowed_power = all_power_sorted.reshape(-1,
1000).mean(axis=1)
    windowed_power = savgol_filter(all_power_sorted,
window_length=101, polyorder=4)
    windowed_power = windowed_power[:, :1000]
    # windowed_freqs = savgol_filter(all_freqs_sorted, 101, 2)
    # windowed_power = savgol_filter(all_power_sorted, 101, 2)

    delta_f = sample_rate / len(windowed_freqs)
    min_width_bins = min_bandwidth / delta_f
    max_width_bins = max_bandwidth / delta_f

    peaks, properties = sci_find_peaks(windowed_power,
noise_median * (1.0 + noise_above), distance=max_width_bins,
width=(min_width_bins, max_width_bins))

    detected_transmissions = [(windowed_freqs[idx],
windowed_power[idx]) for idx in peaks if windowed_freqs[idx] >
start_freq]
    center_frequencies = [point[0] for point in
detected_transmissions]

    logger.info(f"Threshold: {noise_median * (1 +
noise_above)}")

    if visualize:
        plt.figure(figsize=(12, 6))
        plt.plot(windowed_freqs, windowed_power,
label="Combined Spectrum")
        if detected_transmissions:
            trans_np = np.array(detected_transmissions)
            plt.plot(trans_np[:, 0], trans_np[:, 1], 'ro',
markersize=8, label="Candidate Peaks")
            plt.xlabel("Frequency (Hz)")
            plt.ylabel("Power (dB)")
            plt.title(f"Combined Spectrum from {start_freq /
1e6:.2f} MHz to {stop_freq / 1e6:.2f} MHz")
            plt.legend()
            plt.grid(True)
            plt.savefig('samples/images/all_spectrum.png')

    return center_frequencies

```

```

def downsample(samples, factor):
    return samples[::factor]

def get_noise_threshold(s_mag: np.ndarray, noise_additive =
3):
    flat_db = s_mag.flatten()
    hist_vals, bin_edges = np.histogram(flat_db, bins=200)
    noise_floor = bin_edges[np.argmax(hist_vals)]
    return noise_floor + noise_additive # db

def spectrogram_contrast(s_mag_db: cupy.ndarray, window_size:
int = 10, threshold_percentile: float = 50,
                        threshold_strength: float =
0.05) -> cupy.ndarray:

    tbins, nbins = s_mag_db.shape

    S_cpu = cupy.asnumpy(s_mag_db).astype(np.float32)
    AZ = np.zeros_like(S_cpu)

    for i in range(tbins):
        start = max(0, i - window_size + 1)
        AZ[i, :] = np.mean(S_cpu[start:i + 1, :], axis=0)

    AZ_norm = (AZ - AZ.min()) / (AZ.max() - AZ.min() + 1e-8)

    azt = np.percentile(AZ_norm, threshold_percentile)
    azt += (AZ_norm.max() - azt) * threshold_strength

    AZ_norm[AZ_norm < azt] = 0
    AZ_norm = 1 - AZ_norm

    return cupy.asarray(AZ_norm, dtype=cupy.float32)

def histogram_of_dominant_freqs(s_mag_db: np.ndarray, freqs:
np.ndarray, threshold_db: float = -60) -> tuple[np.ndarray,
np.ndarray]:
    time_bins = s_mag_db.shape[1]

```

```

dominant_freq_indices = []

for t in range(time_bins):
    slice_db = s_mag_db[:, t]
    idx = np.argmax(slice_db) # or use: slice_db >
threshold_db
    if slice_db[idx] > threshold_db:
        dominant_freq_indices.append(idx)

hist, bins = np.histogram(dominant_freq_indices,
bins=len(freqs), range=(0, len(freqs)))
freqs_hist = freqs # 1:1 mapping

return hist, freqs_hist

def build_spectrogram(candidates: list[float], sample_rate:
float, num_samples: int, desired_sample_rate: int):
    downsample_factor = int(sample_rate/desired_sample_rate)
    reduced_sample_rate = sample_rate / downsample_factor
    for freq in candidates:
        scanner = RfScanner(float(freq), sample_rate,
num_samples)
        scanner.start()
        scanner.set_center_freq(freq)
        scanner.wait()
        samples = scanner.get_samples()
        scanner.stop()

        # samples = lowpass_fir_filter(samples, sample_rate,
cutoff_hz=sample_rate / (2 * downsample_factor))

        # zoomed_samples = downsample(samples,
downsample_factor)
        resampled = resample_poly(x=samples, up=1,
down=downsample_factor)

        s_complex = cu_signal.stft(cupy.asarray(resampled),
fs=reduced_sample_rate)[2]

        # complex spectrogram
        # s_complex = SFT.stft(samples)
        del samples

    # axes

```

```

    n_freq, n_time = s_complex.shape
    duration = num_samples / sample_rate /
downsample_factor
    f = np.linspace(freq - reduced_sample_rate/2, freq +
reduced_sample_rate/2, n_freq)
    t = np.linspace(0, duration, n_time)

    # 1) magnitude (dB)

    s_mag_db_gpu = 20 * cupy.log10(cupy.abs(s_complex) +
1e-12)

    s_mag_db_gpu = spectrogram_contrast(s_mag_db_gpu)

    s_mag_db = cupy.asnumpy(s_mag_db_gpu)

    plt.figure(figsize=(8,4))
    plt.pcolormesh(t, f, s_mag_db, shading='gouraud')
    plt.title(f'Magnitude Spectrogram ({freq/1e6:.3f}
MHz) ')
    plt.xlabel('Time [s]'); plt.ylabel('Frequency [Hz]')
    plt.colorbar(label='dB')
    plt.tight_layout()

plt.savefig(f'samples/images/spec_{floor(freq/1e3)}_mag.png')
    plt.close()

    hist, freqs_hist =
histogram_of_dominant_freqs(s_mag_db, f)

    plt.figure(figsize=(10, 3))
    plt.plot(freqs_hist, hist)
    plt.title(f"Histogram of {floor(freq/1e3)} MHz")
    plt.xlabel("Frequency (Hz)")
    plt.ylabel("Hits")
    plt.grid(True)
    plt.tight_layout()

plt.savefig(f'samples/images/hist_{floor(freq/1e3)}.png')

def main():
    start_freq = 80e6
    stop_freq = 110e6

```

```
step = 1e6
sample_rate = 8e6
num_samples = int(sample_rate // 10)

freq_candidates = scan_frequency_range(
    start_freq,
    stop_freq,
    step,
    sample_rate,
    num_samples,
    5e3,
    200e3,
    visualize=True,
    noise_above=0.03
)

logger.debug('Detected frequencies',
detected=freq_candidates, num_detected=len(freq_candidates))
    build_spectrogram(freq_candidates, 2**23, int(2**21),
500_000)

if __name__ == '__main__':
    main()
```