

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»**

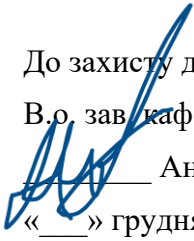
Радіотехнічний факультет

Кафедра прикладної радіоелектроніки

«На правах рукопису»
УДК 004.77:004.424

До захисту допущено:

В.о. зав. кафедри

 Андрій МОВЧАНЮК

«__» грудня 2024 р.

Магістерська дисертація

на здобуття ступеня магістра

**за освітньо-професійною програмою «Інтелектуальні технології
радіоелектронної техніки»**

зі спеціальності 172 Електронні комунікації та радіотехніка

на тему: «Управління потоками даних у персональних мережах Bluetooth»

Виконав:

студент 2 курсу, групи РЕ-331мп

Мельник Володимир Вадимович

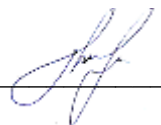
Керівник

ст.викл. Нікітчук Артем Валерійович

Рецензент

к.т.н., доц. Літвінцев Сергій Миколайович

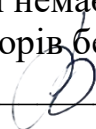






Засвідчую, що у цій магістерській
дисертації немає запозичень з праць
інших авторів без відповідних посилань.

Студент



Київ – 2024 року

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»
Радіотехнічний факультет

Кафедра прикладної радіоелектроніки

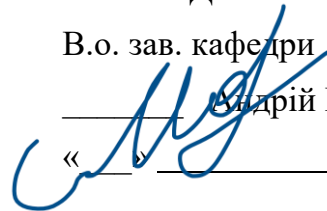
Рівень вищої освіти – другий (магістерський)

Спеціальність – 172 Електронні комунікації та радіотехніка

Освітньо-професійна програма «Інтелектуальні технології радіоелектронної техніки»

ЗАТВЕРДЖУЮ

В.о. зав. кафедри


_____ Андрій МОВЧАНЮК

«_____» 2024 р.

ЗАВДАННЯ
на магістерську дисертацію студента

Мельнику Володимирі Вадимовичу

1. **Тема дисертації** «Управління потоками даних у персональних мережах Bluetooth»

науковий керівник дисертації Нікітчук Артем Валерійович,
затверджені наказом по університету від «08» листопада 2024 р. № 5024-с

2. **Строк подання студентом дисертації** 9 грудня 2024 року

3. **Об'єкт дослідження:** передача даних у персональних Bluetooth-мережах.

4. **Предмет дослідження (Вихідні дані – для магістерської дисертації за ОПП/ОНП):** методи та алгоритми агрегації аудіопотоків в Bluetooth-мережі.

5. **Перелік завдань, які потрібно розробити:** провести огляд сучасних підходів до управління потоками даних у персональних Bluetooth-мережах. Розробити модель агрегації аудіопотоків, що враховує специфіку Bluetooth-з'єднання. Створити алгоритм обробки та об'єднання аудіопотоків з декількох джерел. Реалізувати програмний або апаратний прототип. Провести експериментальне тестування. Проаналізувати результати та зробити висновки. Розробити стартап-проект.

6. **Орієнтовний перелік графічного (ілюстративного) матеріалу:** презентація на 10-15 слайдів.

7. Орієнтовний перелік публікацій: доповідь на конференції з публікацією тез.

8. Дата видачі завдання 25 жовтня 2024 року.

Календарний план

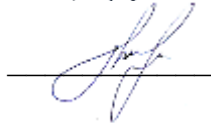
№ з/п	Назва етапів виконання магістерської дисертації	Термін виконання етапів магістерської дисертації	Примітка
1	Аналіз сучасних методів передачі даних та керування їх потоками у Bluetooth-мережах	28.10.24 – 4.11.24	Виконано з запізненням
2	Розробка моделі та алгоритму агрегації аудіо сигналів в Bluetooth-мережі	4.11.24 – 11.11.24	Виконано з запізненням
3	Реалізація програмного або апаратного прототипу	11.11.24 – 18.11.24	Виконано з запізненням
4	Експериментальне тестування	18.11.24 – 25.11.24	Виконано з запізненням
5	Розробка стартап-проекту	25.11.24 – 02.12.24	Виконано з запізненням
6	Оформлення документації	02.12.24 – 09.12.24	Виконано з запізненням

Студент



Володимир МЕЛЬНИК

Науковий керівник



Артем НІКІТЧУК

РЕФЕРАТ

Користувачі сучасних бездротових аудіо-пристроїв та мультимедійних систем зіткаються з неможливістю одночасної роботи з декількома джерелами аудіо. Традиційні підходи до використання технології Bluetooth здебільшого обмежені взаємодією «один на один» (одне джерело – один приймач), що ускладнює реалізацію сценаріїв спільного використання кількох пристроїв. Існуючі Bluetooth-трансивери зазвичай потребують використання дротових з'єднань та не реалізують сценарії спільного використання в реальному часі лише бездротових пристроїв, що зумовлює необхідність розробки та дослідження нових підходів до управління потоками даних в таких мережах.

Метою дисертації є розробка методу та відповідних алгоритмів для інноваційного вирішення завдання присвяченого агрегації потоків даних (аудіо) в персональних Bluetooth-мережах, а також підтвердження працездатності запропонованого рішення.

Об'єктом дослідження є процес передачі даних у персональних Bluetooth-мережах. Предметом дослідження є методи та алгоритми агрегації аудіо-потоків в персональних Bluetooth-мережах.

Для досягнення поставленої мети виконувався аналіз літературних джерел з застосуванням методів контент-аналізу, порівняльного та критичного аналізу для вивчення сучасних підходів до управління потоками даних у Bluetooth-мережах. Застосовувалось об'єктно-орієнтоване програмування та комп'ютерне моделювання для створення запрограмованої моделі у середовищі онлайн-симулятора. Перевірка працездатності запропонованої концепції відбувалась за допомогою емпіричних методів.

Розроблений підхід до керування потоками даних в Bluetooth-мережах має практичне значення. Його застосування дозволяє створити новий клас комерційних продуктів, які будуть забезпечувати одночасний прийом сигналів від кількох джерел та їх передачу на один приймач. Це значно розширить можливості використання Bluetooth у мультимедійних системах,

таких як персональні бездротові пристрої, домашні кінотеатри, конференц-зали та системи озвучення.

В процесі виконання дисертації прийнято участь у XIII Міжнародній науково-технічній конференції «Радіотехнічні проблеми, сигнали, апарати та системи» (27-28.11.2024) з публікацією тез на тему «Data Streams Management In Personal Bluetooth Networks».

Ключові слова: BLUETOOTH NETWORKS, AUDIO, STREAM, AGGREGATION, SCATTERNET, SOFTWARE.

SUMMARY

Users of modern wireless audio devices and multimedia systems face the inability to simultaneously work with multiple audio sources. Traditional approaches to Bluetooth technology are mostly limited to "one-to-one" interactions (one source – one receiver), which complicates the implementation of scenarios involving the shared use of multiple devices. Existing Bluetooth transceivers typically require wired connections and do not support real-time scenarios exclusively involving wireless devices. This necessitates the development and research of new approaches to data stream management in such networks.

The aim of this dissertation is to develop a method and corresponding algorithms for an innovative solution to the task of audio data stream aggregation in personal Bluetooth networks, as well as to validate the feasibility of the proposed solution.

The object of the research is the data transmission process in personal Bluetooth networks. The subject of the research is the methods and algorithms for audio stream aggregation in personal Bluetooth networks.

To achieve the stated aim, a literature review was conducted using content analysis, comparative analysis, and critical analysis methods to study modern approaches to data stream management in Bluetooth networks. Object-oriented programming and computer modeling were applied to create a programmed model in an online simulation environment. The proposed concept was validated using empirical methods.

The developed approach to data stream management in Bluetooth networks has practical significance. Its implementation enables the creation of a new class of commercial products that allow the simultaneous reception of signals from multiple sources and their transmission to a single receiver. This significantly expands the capabilities of Bluetooth in multimedia systems, such as personal wireless devices, home theaters, conference rooms, and public address systems.

During the dissertation, participation was taken in the XIII International Scientific and Technical Conference «Radio Engineering Problems, Signals,

Devices, and Systems» (November 27-28, 2024), with the publication of abstracts on the topic «Data Streams Management in Personal Bluetooth Networks».

Keywords: BLUETOOTH NETWORKS, AUDIO, STREAM, AGGREGATION, SCATTERNET, SOFTWARE.

ЗМІСТ

ПЕРЕЛІК СКОРОЧЕНЬ.....	10
ВСТУП	11
1 АНАЛІЗ ОСОБЛИВОСТЕЙ ТА МЕТОДІВ УПРАВЛІННЯ ПОТОКАМИ ДАНИХ У ПЕРСОНАЛЬНИХ BLUETOOTH-МЕРЕЖАХ	12
1.1 Технологічні аспекти обміну даними у Bluetooth-мережах	12
1.2 Методи управління потоками даних	17
1.3 Пристрої управління потоками даних для мультимедійних застосувань.....	20
1.4 Висновки до розділу 1	24
2 РОЗРОБКА АЛГОРИТМУ АГРЕГАЦІЇ АУДІОПОТОКІВ У BLUETOOTH- МЕРЕЖАХ.....	25
2.1 Визначення технічних вимог до системи агрегації	25
2.2 Розробка концептуальної схеми скаттер-мережі.....	26
2.3 Розробка алгоритму обробки і об'єднання аудіо-потоків	31
2.4 Висновки до розділу 2	34
3 ПІДТВЕРДЖЕННЯ ЗАПРОПОНОВАНОЇ КОНЦЕПЦІЇ ТА РОЗРОБКА ПРОТОТИПУ ДЛЯ АГРЕГАЦІЇ АУДІО-ПОТОКІВ.....	35
3.1 Вибір апаратної платформи та інструментів розробки програмного забезпечення	35
3.2 Підготовка коду для реалізації розробленого алгоритму	38
3.3 Створення моделі для доказу концепції з використанням онлайн- симулятора	42
3.4 Висновки до розділу 3	45

4 СТАРТАП-ПРОЕКТ	47
4.1 Опис ідеї проекту	47
4.2 Технологічний аудит ідеї проекту	48
4.3 Аналіз ринкових можливостей запуску стартап-проекту	49
4.4 Розроблення ринкової стратегії проекту	52
4.5 Розроблення маркетингової програми стартап-проекту	54
4.6 Висновки до розділу 4	55
ВИСНОВКИ	57
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	58
ДОДАТКИ	63

ПЕРЕЛІК СКОРОЧЕНЬ

A2DP	—	Advanced Audio Distribution Profile
ATT	—	Enhanced Data Rate
AVDTP	—	Audio/Video Distribution Transport Protocol
AVRCP	—	Audio/Video Remote Control Profile
BLE	—	Bluetooth Low Energy
BNEP	—	Bluetooth Network Encapsulation Protocol
CDM	—	Code Division Multiplexing
CVSD	—	Continuous Variable Slope Delta Modulation
DRR	—	Deficit Round Robin
EDR	—	Enhanced Data Rate
FDM	—	Frequency Division Multiplexing
GATT	—	Generic Attribute Profile
HFP	—	Hands-Free Profile
HSP	—	Headset Profile
IoT	—	Інтернет речей
L2CAP	—	Logical Link Control and Adaptation Protocol
PAN	—	Personal Area Network
RFCOMM	—	Radio Frequency Communication
SBC	—	Subband Codec
SMP	—	Security Manager Protocol
TDM	—	Time Division Multiplexing
WFQ	—	Weighted Fair Queuing

ВСТУП

Розвиток персональних мереж Bluetooth надає можливість інтеграції широкого спектру пристроїв, які забезпечують передачу даних та мультимедіа в реальному часі. Зокрема, технологія Bluetooth знаходить застосування у таких сферах, як портативна електроніка, системи «розумного будинку», автомобільні аудіосистеми, гарнітури та носимі пристрої. В умовах швидкого зростання кількості Bluetooth-пристроїв виникає потреба в ефективному управлінні потоками даних, особливо в мультимедійних додатках.

Традиційна архітектура Bluetooth піко-мереж має обмеження, що створює виклик для реалізації сценаріїв, де необхідно об'єднати кілька джерел аудіо в єдиний потік для відтворення на одному приймачі. В новітніх пристроях впроваджується технологія Multipoint, але вона лише автоматизує перемикання відтворюючого пристрою між декількома джерелами аудіо.

Вирішення цієї проблеми вимагає розробки інноваційних методів управління потоками даних, які дозволять агрегувати дані з кількох бездротових джерел для їх подальшої передачі між піко-мережами в режимі реального часу. Результати дисертації сприятимуть розширенню меж застосування персональних мереж Bluetooth в мультимедійних та інших системах.

1 АНАЛІЗ ОСОБЛИВОСТЕЙ ТА МЕТОДІВ УПРАВЛІННЯ ПОТОКАМИ ДАНИХ У ПЕРСОНАЛЬНИХ BLUETOOTH-МЕРЕЖАХ

1.1 Технологічні аспекти обміну даними у Bluetooth-мережах

Bluetooth є провідною технологією бездротового зв'язку, що активно використовується у сучасних мережах для передачі даних між пристроями. Вона забезпечує стабільний зв'язок між пристроями з низьким енергоспоживанням, що робить її ідеальною для використання у персональних мережах, пристроях IoT, мультимедійних системах та мобільних гаджетах [1].

Bluetooth-мережі мають гнучку архітектуру. Вони побудовані на основі *піко-мереж* – невеликих мереж, де один пристрій є головним (*master*), а до семи інших можуть бути підлеглими (*slaves*). Для розширення можливостей з'єднання декілька *піко-мереж* можуть об'єднуватися в *scatternet* за допомогою спільних пристроїв. Це дозволяє підключати більше пристроїв і створювати складніші мережі [2].

Піко-мережа – це базова топологія Bluetooth-мережі, що об'єднує один головний пристрій (*master*) і до семи активних підлеглих пристроїв (*slaves*). Головний пристрій виконує функцію координатора мережі, забезпечуючи синхронізацію та розподіл часових слотів для передачі даних. Підлеглі пристрої отримують і передають дані відповідно до команд головного пристрою.

Така структура забезпечує стабільність з'єднання та низькі затримки під час передачі даних у межах невеликої мережі. Крім того, кожен пристрій у піко-мереж може перебувати в пасивному режимі, що дозволяє економити енергію, коли активна передача даних не потрібна.

Для розширення можливостей декілька піко-мереж об'єднуються через спільні пристрої у **скаттер-мережі**. Наприклад, один пристрій може бути підлеглим у першій мережі та головним у другій. Scatternet дозволяє створювати масштабовані системи з багатьма пристроями, розширюючи функціональність стандартної піко-мережі (рис. 1.1).

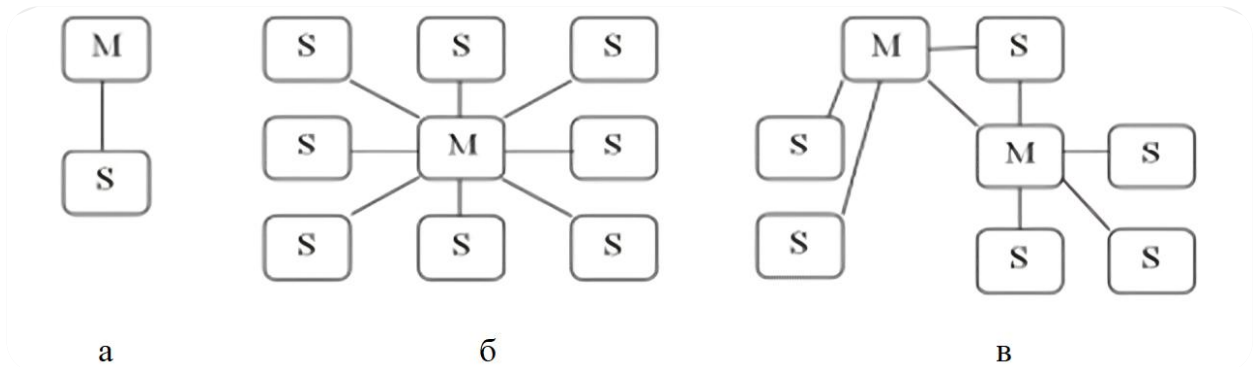


Рисунок 1.1 — Різновиди піко-мереж: а – з одним залежним пристроєм;

б – з кількома залежними пристроями; в – розгалужена мережа [3]

Архітектура Bluetooth також підтримує кілька режимів роботи, що впливають на пропускну здатність і енергоспоживання. У режимі Active пристрій постійно підтримує зв'язок із мережею, забезпечуючи високу швидкість передачі. У режимах Sniff або Park пристрій знижує енергоспоживання, зменшуючи частоту активного обміну даними, що критично для носимих гаджетів і сенсорів IoT [4].

Процес встановлення з'єднання в Bluetooth-мережах включає кілька основних етапів. Спочатку відбувається сканування, під час якого пристрої шукають один одного. Після виявлення можливого партнера здійснюється зв'язування (*pairing*), яке включає обмін ключами для забезпечення безпеки даних. Наступним етапом є встановлення з'єднання, після чого пристрої можуть передавати інформацію один одному [5].

Bluetooth-мережі забезпечують ефективну передачу даних завдяки багаторівневій архітектурі протоколів, яка охоплює фізичний, транспортний і прикладний рівні. Ключовими протоколами є L2CAP, що відповідає за сегментацію й реасемблювання великих пакетів, і RFCOMM, який емулює серійний порт для забезпечення сумісності з пристроями старих стандартів. L2CAP також підтримує агрегацію потоків даних, дозволяючи декільком сервісам одночасно використовувати один Bluetooth-канал, що є критично важливим для мультимедійних додатків [6].

Для зниження енергоспоживання використовується АТТ-протокол, який працює за клієнт-серверною моделлю, та GATT, що організовує дані в BLE. Bluetooth також інтегрує протоколи для передачі IP-даних (BNEP) та забезпечує безпеку через SMP, який гарантує аутентифікацію й шифрування даних. Завдяки адаптивному управлінню потоками мережі динамічно змінюють пріоритети передачі, забезпечуючи якість обслуговування для аудіо й відео додатків [7-9].

Таким чином, протоколи Bluetooth формують універсальну й надійну систему бездротової передачі даних, забезпечуючи високу сумісність пристроїв, багатозадачність і адаптацію до сучасних сценаріїв використання, зокрема в мультимедіа та IoT [10, 11].

Bluetooth-технологія широко використовується для передачі мультимедійних даних завдяки профілям A2DP (Advanced Audio Distribution Profile) та AVRCP (Audio/Video Remote Control Profile). A2DP забезпечує передачу високоякісного стереозвуку через протокол AVDTP (Audio/Video Distribution Transport Protocol), що мінімізує затримки й забезпечує стабільність зв'язку. Основною перевагою A2DP є підтримка різних аудіокодеків, таких як SBC, AAC, aptX і LDAC, які дозволяють балансувати між якістю звуку та пропускнуою здатністю каналу. AVRCP, у свою чергу, додає функції дистанційного керування, дозволяючи користувачам змінювати треки, регулювати гучність та взаємодіяти з іншими мультимедійними функціями [12].

Для голосових викликів застосовуються профілі HFP (Hands-Free Profile) і HSP (Headset Profile), які забезпечують якісну передачу голосу через сучасні кодеки, як-от mSBC. З огляду на зростаючі вимоги до якості звуку та енергоефективності, було розроблено профіль LE Audio, який працює на основі Bluetooth Low Energy (BLE) та використовує кодек LC3 (Low Complexity Communication Codec). LE Audio підтримує функцію Audio Sharing, що дозволяє передавати аудіо на кілька пристроїв одночасно,

зменшуючи енергоспоживання й оптимізуючи якість звуку навіть за низької швидкості передачі [12].

Сучасні стандарти Bluetooth також враховують потреби в аудіо високої чіткості та багатоканальному звуці. Наприклад, інтеграція підтримки просторового звуку (Dolby Atmos, DTS:X) робить Bluetooth оптимальним рішенням для домашніх кінотеатрів і ігрових систем. Кодеки з низькою затримкою, такі як aptX Low Latency та LC3, забезпечують синхронізацію між аудіо й відео, що критично для відеоігор і потокових платформ. Таким чином, мультимедійні профілі Bluetooth забезпечують універсальність, високу якість та зручність у використанні для широкого спектра пристроїв [12].

Bluetooth-технологія забезпечує передачу даних різної швидкості завдяки постійним оновленням стандартів. У версії Bluetooth 5.0 швидкість передачі даних досягає 2 Мбіт/с, а функція Long Range дозволяє збільшити радіус дії до 200 метрів із компромісом у швидкості. Особливістю Bluetooth Low Energy (BLE) є його енергоефективність, яка досягається завдяки коротшим пакетам даних і менш частому їх передаванню, що ідеально підходить для пристроїв IoT [13]. На рисунку 1.2 наведено порівняння версій Bluetooth, яке демонструє еволюцію від класичного Bluetooth, орієнтованого на передачу файлів і потокового аудіо, до BLE, що фокусується на тривалому часі роботи пристроїв [13].

Технічні характеристики	Bluetooth Classic BR/EDR	BLE	
		Bluetooth 4.x	Bluetooth 5
Радіочаст. (МГц)	2400 до 2483,5	2400 до 2483,5	2400 до 2483,5
Канали	79 (1 МГц)	40 (2 МГц)	40 (2 МГц)
Відстань (м)	До 100	До 100	До 200
Затримка (мс)	100	<6	<6
Швидкість передачі даних (Мбіт/с)	1, 2, 3	1	0,5, 0,125, 1, 2
Макс. активні вузли	8	Необмежений	
Розмір повідомлення (байти)	До 358	31	255
Максимальне корисне навантаження (байти)	1021	37,255	255
Піковий струм (мА)	<30	<15	<15

Рисунок 1.2 — Порівняння версій Bluetooth [13]

Подальші вдосконалення, реалізовані в Bluetooth 5.2, включають підтримку ізохронних каналів (Isochronous Channels) для синхронізації аудіопотоків, що є основою для профілю LE Audio. Також було впроваджено кодек LC3 (Low Complexity Communication Codec), який забезпечує високу якість звуку навіть за низької швидкості передачі даних. Ця версія підтримує функцію Audio Sharing для одночасної трансляції аудіо на кілька пристроїв, що розширює можливості використання Bluetooth у мультимедійних і IoT-застосунках [3, 14]. Bluetooth також використовує радіочастотний діапазон 2,4 ГГц із технологією FHSS, яка мінімізує вплив перешкод і забезпечує стабільність з'єднання.

Отже, архітектурні особливості Bluetooth-мереж забезпечують їхню гнучкість, масштабованість і енергоефективність. Підтримка різних режимів роботи та високої пропускної здатності робить Bluetooth універсальною технологією для широкого спектра застосувань – від простого обміну файлами до складних IoT-систем і мультимедійних платформ [15].

Множинна передача даних у Bluetooth-мережах є критично важливою для забезпечення якості обслуговування (QoS) у мультимедійних додатках і системах IoT. Вона базується на алгоритмах пріоритизації, які визначають порядок передачі даних, забезпечуючи мінімальну затримку для часокритичних потоків, таких як аудіо чи відео. Наприклад, протоколи AVDTP і HFP отримують вищий пріоритет порівняно з менш терміновими задачами. Для покращення ефективності використовуються механізми чергування пакетів, зокрема алгоритми Weighted Fair Queuing (WFQ) та Dynamic Priority Scheduling, які враховують пріоритетність і адаптуються до поточних умов у мережі [16].

Стабільність і синхронізація потоків досягаються завдяки часовим слотам, що синхронізують роботу пристроїв, а механізми буферизації запобігають втратам даних у разі перевантаження каналу. Bluetooth також підтримує канали з гарантованою якістю обслуговування, резервуючи пропускну здатність для критичних потоків. Завдяки цим підходам Bluetooth-мережі забезпечують стабільну та якісну передачу мультимедійних даних навіть у складних умовах [16].

1.2 Методи управління потоками даних

Управління потоками даних у мережах Bluetooth – це комплекс методів, спрямованих на організацію, передачу та обробку інформації між пристроями. Ефективне управління потоками дозволяє оптимізувати використання ресурсів мережі, зменшити затримки, забезпечити узгодженість даних і підтримувати якість обслуговування.

Для управління потоками даних може застосовуватись їх обробка, агрегація, мультиплексування/демультиплексування, протоколи синхронізації та буферизація.

Обробка аудіо-даних у Bluetooth-мережах включає декодування, фільтрацію, стиснення, корекцію помилок і адаптацію до умов передачі. Мета

обробки полягає в забезпеченні високої якості звуку навіть за несприятливих умов роботи мережі, таких як перешкоди або низький рівень сигналу.

У Bluetooth використовується кілька кодеків для стиснення аудіо-даних, таких як SBC (Subband Codec) та aptX. Стиснення дозволяє зменшити розмір даних для передачі, зберігаючи їхню якість.

Фільтрація шуму – це процес видалення сторонніх звуків для покращення якості аудіо. Сучасні алгоритми шумопригнічення використовують адаптивні фільтри, які підлаштовуються до типу шуму [12].

Корекція помилок – Bluetooth використовує механізми корекції помилок, такі як FEC (Forward Error Correction), для мінімізації втрат даних і зниження впливу перешкод на якість передачі.

Адаптація до умов передачі – це зміна параметрів передачі, таких як бітрейт чи частота семплювання, залежно від якості зв'язку [8].

Ефективна обробка аудіо-даних дозволяє забезпечити високу якість звуку при мінімальному використанні ресурсів. Це є критично важливим для пристроїв, які працюють на батареях.

Агрегація даних передбачає об'єднання дрібних пакетів даних у більші для зменшення накладних витрат на передачу. Кожен переданий пакет має службову інформацію, яка додає до загального обсягу трафіку. Якщо передавати малі пакети, то велика частина доступної пропускної здатності витрачається на службові потреби, що знижує ефективність мережі [14].

Агрегація дозволяє зменшити затримки на рівні обробки даних, оскільки зменшується кількість окремих операцій, пов'язаних із передачею пакетів. Також знижується енергоспоживання, оскільки зменшується час, необхідний для активної передачі даних. Водночас неправильна агрегація може призводити до значних затримок через необхідність очікування, поки будуть зібрані дані для формування пакета. Для ефективної агрегації необхідно враховувати баланс між розміром пакета, затримками і умовами роботи мережі [4].

Мультиплексування – це метод, за допомогою якого кілька потоків даних передаються через один фізичний канал. Для Bluetooth-мереж мультиплексування є важливим компонентом, оскільки дозволяє оптимально використовувати обмежену пропускну здатність і забезпечувати стабільну передачу даних для кількох пристроїв [15].

Одним із ключових аспектів мультиплексування є управління часом. Кожен потік отримує певну кількість слотів, у межах яких він може передавати дані. Це дозволяє уникнути конфліктів і забезпечити безперервність передачі.

Основним викликом при реалізації мультиплексування є забезпечення справедливого розподілу ресурсів між потоками, які мають різні вимоги до якості та затримки. Наприклад, потоки голосу вимагають мінімальних затримок, тоді як потоки даних можуть бути менш чутливими до часу.

Перевагами мультиплексування є підвищення ефективності використання каналу, зменшення накладних витрат і забезпечення стабільності зв'язку для кількох потоків [7].

Можна виділити декілька методів мультиплексування:

- Часове мультиплексування (TDM – Time Division Multiplexing) – кожному потоку виділяється окремий часовий слот у межах доступного каналу.
- Частотне мультиплексування (FDM – Frequency Division Multiplexing) – різні потоки даних передаються на різних частотних каналах [17].
- Кодове мультиплексування (CDM – Code Division Multiplexing) також може використовуватися в Bluetooth-мережах, хоча цей метод рідше застосовується у порівнянні з іншими [17].
- Гібридне мультиплексування – поєднує кілька методів одночасно.

Таким чином, застосування різноманітних методів дозволяє адаптувати мережу до конкретних задач і умов роботи, забезпечуючи високу якість обслуговування навіть у складних сценаріях.

1.3 Пристрої управління потоками даних для мультимедійних застосувань

У сучасних Bluetooth-мережах існують технології та пристрої для управління потоками даних, які забезпечують ефективне функціонування різних сценаріїв використання.

Multipoint (багатоточковий Bluetooth) є ключовою технологією для сучасних Bluetooth-мереж, яка дозволяє одному пристрою підтримувати з'єднання з кількома іншими пристроями одночасно. На відміну від одноадресного з'єднання, Multipoint надає можливість передавати кілька потоків даних через один інтерфейс, але у послідовному порядку. Це робиться шляхом динамічного чергування каналів передачі між підключеними пристроями [18, 19].

Bluetooth-модулі з підтримкою аудіо мають ключову характеристику – низьку затримку передачі даних, яка становить 30-50 мс при використанні кодеків *aptX Low Latency*. Вони підтримують частоту вибірки 44,1-48 кГц і частоту передачі даних до 328 кбіт/с для високоякісного стереозвуку. Ці модулі можуть працювати в ролі джерела або приймача аудіосигналів, забезпечуючи якісну передачу звуку навіть у багатозадачних умовах.

Bluetooth-адаптери спрямовані на оптимізацію потоків даних між сенсорами, контролерами та іншими пристроями. Вони підтримують режим низької енергії (Bluetooth Low Energy, BLE), що дозволяє значно знижувати енергоспоживання. Основою їхньої роботи є протокол *ATT* (Attribute Protocol), який забезпечує ефективний обмін даними між пристроями. Такі адаптери використовуються в системах моніторингу обладнання, автоматизації промислових процесів та управління розумними системами освітлення. Вони відзначаються низьким енергоспоживанням – у режимі очікування вони споживають лише 0,01 мА, а в активному режимі – до 15 мА. Вони підтримують швидкість передачі даних до 1 Мбіт/с (BLE), а також працюють на частоті 2,4 ГГц із технологією адаптивної частотної модуляції (AFH), що

знижує ризик перешкод від інших бездротових пристроїв. Радіус дії зазвичай становить 10-50 метрів, залежно від умов експлуатації [20].

Ще одним цікавим типом пристроїв є *багатофункціональні Bluetooth-гейти*, які поєднують можливості Bluetooth-з'єднання з іншими мережами, такими як Wi-Fi або Ethernet. Вони відіграють ключову роль у передачі даних між локальними Bluetooth-мережами та хмарними сервісами, забезпечуючи маршрутизацію та фільтрацію даних для оптимізації потоків. Такі гейти часто використовуються в інтелектуальних будинках, де потрібно інтегрувати різні пристрої в єдину мережу.

Багатофункціональні *Bluetooth-гейти* мають вбудовані модулі Wi-Fi 802.11 b/g/n та *Ethernet-порт* для провідного підключення. Їхня потужність передавача дозволяє забезпечити стабільний сигнал на відстані до 200 метрів. Вони підтримують одночасну роботу з десятками пристроїв завдяки інтеграції стеків протоколів Bluetooth і IP, дозволяючи гнучке керування потоками даних у гібридних мережах [21].

Серед Bluetooth-пристроїв великий інтерес привертають бездротові подвійні аудіо-передавачі і приймачі, які можна використовувати з різними з'єднаннями. Наприклад, AUGUST MR280 бездротовий подвійний аудіопередавач і приймач Bluetooth (рис. 1.3) [22].



Рисунок 1.3 — AUGUST MR280 та схема підключення [22]

MR280 підтримує версію BT 5.0 та дозволяє легко відтворювати музику зі смартфона, планшета чи ноутбука на динаміки, Wi-Fi системи, підсилювачі або інші аудіопристрої. Адаптер працює за принципом Plug & Play, не потребує драйверів і підтримує різні типи підключень [22].

Іншим подібним пристроєм є Wireless Triple Bluetooth Audio Transmitter (рис. 1.4). Даний передавач підключившись до джерела живлення може передавати сигнал від 1 до 3 аудіо-приймачів [23].



Рисунок 1.4 — Потрійний аудіопередавач PMD Way [23]

Bluetooth аудіоресивер/трансмідер **Vikefon BT-B6** (рис. 1.5) – це універсальний пристрій, що поєднує функції прийому та передачі аудіосигналу через Bluetooth.



Рисунок 1.5 — Трансмідер Vikefon BT-B6 [24]

Він підтримує протоколи A2DP та AVRCP, забезпечуючи надійне з'єднання з радіусом дії до 10 метрів. Вбудований акумулятор ємністю 600 мА·год дозволяє працювати до 8 годин автономно, а повна зарядка триває приблизно 2 години. Пристрій оснащений світлодіодною індикацією стану та живиться через вбудований Micro-USB роз'єм [24].

Bluetooth аудіо трансмітер innoTronics (рис. 1.6), також є одним з найпоширеніших – це сучасний пристрій, що поєднує функції передавача та приймача аудіосигналу через Bluetooth. Він підтримує стандарт Bluetooth 5.0, забезпечуючи стабільне з'єднання з радіусом дії до 15 метрів. Завдяки технології aptX Low Latency, затримка звуку зведена до мінімуму, що гарантує синхронізацію аудіо та відео. Пристрій оснащений чипом Qualcomm CSR 8670, що забезпечує високу якість звуку та надійність роботи. Вбудований акумулятор ємністю 350 мА·год дозволяє працювати до 12 годин у режимі передавача та до 15 годин у режимі приймача. Адаптер підтримує одночасне підключення двох Bluetooth-пристроїв, що розширює можливості його використання [25].



Рисунок 1.6 — Bluetooth аудіо-трансмiтер innoTronics [25]

1.4 Висновки до розділу 1

1. Піко-мережі є базовою структурою для невеликих груп пристроїв, тоді як скаттер-мережі дозволяють масштабувати систему, інтегруючи більшу кількість пристроїв.
2. Еволюція пропускної здатності Bluetooth підтверджує, що сучасні версії (Bluetooth 5.0, 5.2) забезпечують оптимальне співвідношення швидкості, енергоефективності та дальності дії, що створює передумови для впровадження більш складних мультимедійних додатків.
3. Вивчення протоколів обміну даними, таких як L2CAP, ATT і AVDTP, свідчить про необхідність інтеграції механізмів адаптації та стабілізації з'єднання у запропонованих рішеннях. Це особливо важливо для забезпечення якісної передачі в умовах змінного навантаження мережі.
4. Нові рішення, такі як LE Audio та LC3-кодек, відкривають можливості для зниження енергоспоживання та підвищення якості звуку.

2 РОЗРОБКА АЛГОРИТМУ АГРЕГАЦІЇ АУДІОПОТОКІВ У BLUETOOTH-МЕРЕЖАХ

2.1 Визначення технічних вимог до системи агрегації

Розробка алгоритму агрегації аудіопотоків у Bluetooth-мережах вимагає визначення ключових технічних вимог, які забезпечать ефективну та стабільну роботу системи в умовах багатопотокової передачі даних. Ці вимоги базуються на специфіці Bluetooth-технології, обмеженнях її пропускної здатності, потребах у синхронізації потоків, а також необхідності забезпечення високої якості передачі аудіо в реальному часі.

Першою і найважливішою вимогою є *мінімізація затримок у передачі даних*. У Bluetooth-мережах, які передають аудіопотоки для голосових викликів чи музики, затримки можуть значно погіршити користувацький досвід. Алгоритм має забезпечувати пріоритетну обробку часокритичних потоків, таких як голосові виклики, та мінімізацію їх затримок навіть в умовах високого навантаження мережі.

Другою вимогою є *синхронізація між пристроями*. У випадках, коли аудіопотоки розподіляються між кількома приймачами (наприклад, у бездротових навушниках із підтримкою стерео), навіть незначна розсинхронізація може спричинити перекручування звуку або затримки в одному з каналів. Алгоритм повинен враховувати часові мітки та забезпечувати узгодженість даних на приймальній стороні [26].

Третьою ключовою вимогою є *ефективне управління пропускною здатністю Bluetooth-каналу*. Відомо, що Bluetooth-мережі мають обмежену пропускну здатність, яка залежить від версії протоколу. Наприклад, у Bluetooth 5.0 максимальна швидкість передачі даних становить 2 Мбіт/с. Для досягнення оптимального використання каналу алгоритм має розподіляти ресурси між потоками на основі їхніх вимог до пропускної здатності, забезпечуючи водночас стабільність мережі.

Наступною важливою вимогою є *гнучкість у адаптації до мережесевих умов*. У середовищах із високим рівнем радіоперешкод або змінним навантаженням (наприклад, підключення нових пристроїв до мережі) система повинна динамічно змінювати алгоритм розподілу слотів і каналів передачі, щоб уникнути втрат даних і збоїв у роботі. Для цього алгоритм має використовувати механізми моніторингу стану мережі та адаптивного управління потоками [26].

Енергоефективність також є важливою технічною вимогою, особливо для носимих пристроїв та інших Bluetooth-пристроїв із живленням від батареї. Алгоритм повинен мінімізувати споживання енергії шляхом оптимізації активного часу пристроїв, зменшення кількості зайвих передач даних та використання енергоефективних кодеків, таких як LC3, які дозволяють передавати звук високої якості з мінімальними витратами енергії.

Останньою вимогою є *забезпечення якості обслуговування (QoS)*. Для цього необхідно реалізувати пріоритизацію аудіопотоків залежно від їхньої критичності. Наприклад, голосові потоки в реальному часі повинні оброблятися з вищим пріоритетом, ніж музика чи фонові дані. Крім того, слід гарантувати мінімальні втрати пакетів для потоків із високим пріоритетом, навіть за умов перевантаження мережі [27].

Ці вимоги є основою для створення ефективного пристрою, який забезпечить надійну передачу аудіопотоків у Bluetooth-мережах.

2.2 Розробка концептуальної схеми скаттер-мережі

Скаттер-мережі в Bluetooth є складною структурою, яка поєднує декілька піко-мереж, де один пристрій виконує функцію ведучого (master), а інші – підлеглих (slave). Концепція скаттер-мережі дозволяє розширити можливості Bluetooth-з'єднання, збільшуючи кількість пристроїв і покращуючи ефективність передачі даних, що забезпечує підвищену гнучкість і

масштабованість завдяки об'єднанню кількох піко-мереж через вузли-містки, які виступають ключовими елементами для передачі даних.

Ця топологія є критично важливою для реалізації багатопотокових застосувань у мультимедійному середовищі та IoT-системах, оскільки дозволяє інтегрувати більше пристроїв у мережу, підтримуючи стабільну передачу даних.

Концептуальна схема скаттер-мережі передбачає функціонування трьох ключових елементів. Головний пристрій піко-мережі (master) виконує функцію координатора мережі, синхронізує передачу даних і розподіляє часові слоти [28]. Підлеглі пристрої (slaves) отримують і передають дані в межах своєї піко-мережі відповідно до команд головного пристрою. Мостові пристрої (bridge nodes) одночасно виконують роль головного пристрою в одній піко-мережі та підлеглого в іншій, забезпечуючи передачу даних між мережами. Це дозволяє уникнути значних затримок і підтримувати цілісність даних при переході між мережами.

Для аудіопотоків можна поділити всі пристрої на дві піко-мережі (рис. 2.1).



Рисунок 2.1 — Мережа Bluetooth аудіопотоків

У першій піко-мережі знаходяться всі джерела аудіосигналів та агрегатор, завдання якого – збір аудіоданих від усіх джерел. У другій піко-мережі агрегатор виступає джерелом аудіосигналу (Bluetooth-передавач), а приймач (наприклад, бездротові навушники або динамік) служить пристроєм для відтворення аудіо. Bluetooth не підтримує прямого зв'язку між пристроями з різних піко-мереж (вони не «бачать» один одного), тому мультиплексор працює як місток. Його завдання – обробка та передача даних між мережами, тобто передача єдиного аудіопотоку, сформованого з даних першої піконети, до приймача у другій мережі [29].

Запропонована схема скаттер-мережі дозволяє підключати більшу кількість пристроїв, розподіляти трафік між піко-мережами та забезпечувати стабільність роботи навіть у складних топологіях.

Однією з ключових особливостей скаттер-мереж є багатoproфільність, коли один пристрій може одночасно виконувати функції ведучого в одній піко-мережі й підлеглого в іншій. Це забезпечує гнучкість у розподілі ресурсів та дозволяє ефективно адаптувати мережу до змін. Завдяки цій структурі скаттер-мережі можуть охоплювати десятки пристроїв, тоді як піко-мережі обмежені сімома активними підлеглими пристроями [29]. Передача даних між піко-мережами: виконується за допомогою вузлів-містків, які виконують роль шлюзів для пересилання інформації. Вони є ключовими елементами для збереження цілісності та ефективності роботи мережі. Кожен вузол має визначену роль у мережі (ведучий, підлеглий або вузол-місток). На рис. 2.2 представлено концептуальну схему скаттер-мережі, яка складається з декількох піко-мереж (A, B, C), з'єднаних між собою за допомогою вузлів-містків (Bridge A-B та Bridge B-C). У кожній піко-мережі є один ведучий пристрій (Master) і кілька підлеглих пристроїв (Slave).

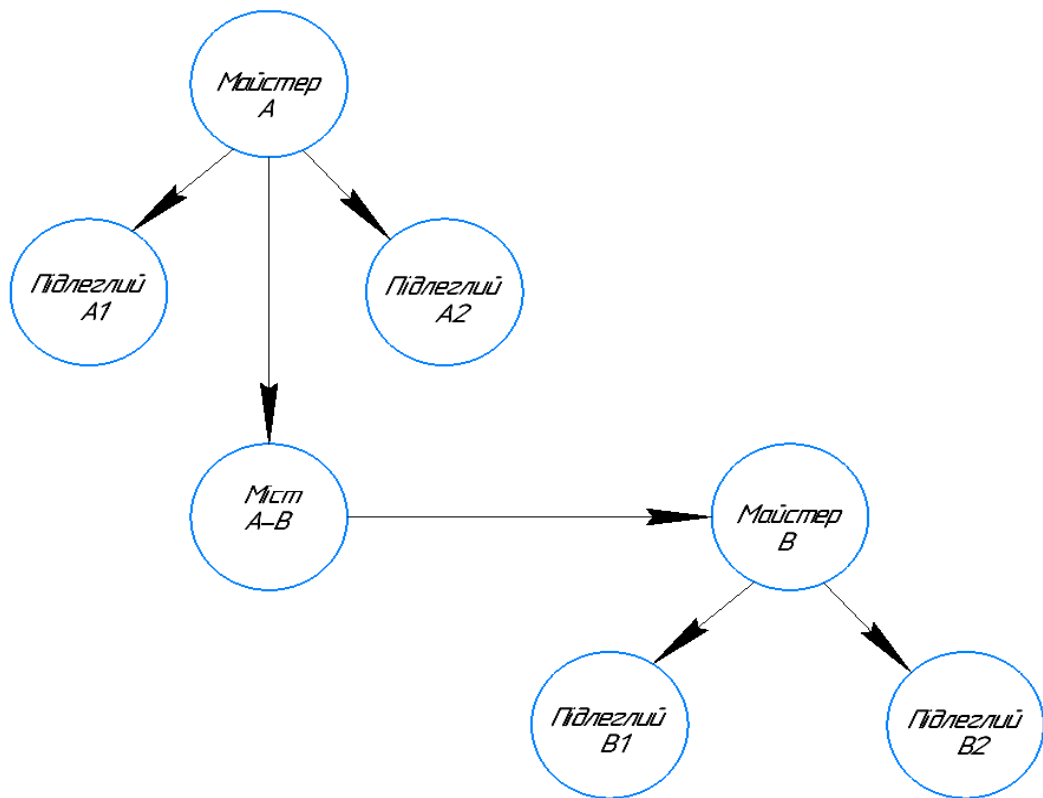


Рисунок 2.2 — Концептуальна схема скаттер-мережі з одним мостом

Особливості функціонування скаттер-мереж включають розподіл часових слотів, синхронізацію та маршрутизацію даних. Для ефективної роботи кожен мостовий пристрій використовує адаптивний алгоритм управління часовими слотами, який враховує обсяг переданих даних і пріоритетність потоків. Синхронізація забезпечується узгодженням часових параметрів між піко-мережами, що дозволяє уникати конфліктів під час передачі даних. Маршрутизація здійснюється через мостові пристрої, які передають дані з однієї піко-мережі до іншої, використовуючи алгоритми, що мінімізують затримки [30].

На рисунку 2.3 зображено запропоновану схему роботи системи для агрегації аудіо-потоків в приватній Bluetooth-мережі.

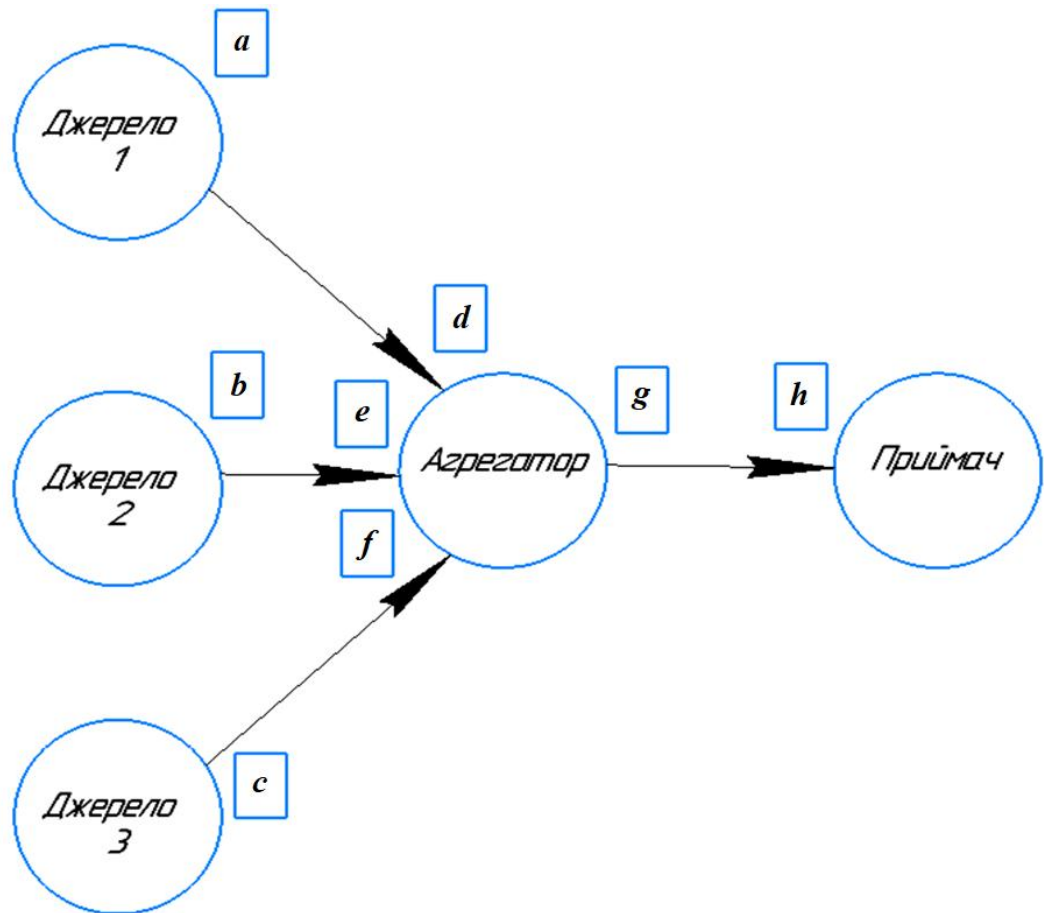


Рисунок 2.3 — Схема розробленої скаттер-мережі

Кожне з підключених Джерел передає свої дані до Агрегатора. Агрегатор – це центральний компонент системи, який приймає аудіо-сигнали від усіх джерел та об’єднує їх в єдиний сигнал, який в реальному часі передає до Bluetooth-приймача.

Позначення a , b , c , d , e , f , g , h відповідають ідентифікаторам кожного з Bluetooth-інтерфейсів, які використовуються для взаємодії між пристроями:

- a , b , c – ідентифікатори джерел аудіо;
- d , e , f – ідентифікатори інтерфейсів на стороні агрегатора, які відповідають за прийом аудіо-потоків;
- g – ідентифікатор інтерфейсу агрегатора, який використовується для передачі об’єднаного аудіо-потoku;
- h – ідентифікатор інтерфейсу приймача.

Розроблена схема створює основу для подальшої розробки алгоритмів управління потоками даних у Bluetooth-мережах.

2.3 Розробка алгоритму обробки і об'єднання аудіо-потоків

Алгоритм роботи «агрегатора» для об'єднання декількох аудіо-потоків складається з кількох основних етапів: ініціалізація, очікування підключення, приймання аудіо, міксування семплів та передача оброблених даних (рис. 2.4).



Рисунок 2.4 — Послідовний алгоритм агрегації аудіо-потоків

На етапі *ініціалізації* виконується налаштування апаратних і програмних компонентів. Для забезпечення коректної роботи агрегатора здійснюється активація профілю A2DP Sink, що відповідає за приймання аудіо, та профілю A2DP Source, що забезпечує передачу аудіо. Ініціалізація передбачає налаштування кодеків, наприклад SBC або AAC, для роботи з аудіопотоками. Далі створюються буфери для кожного аудіо-джерела та приймача.

Після ініціалізації система переходить у режим *очікування підключення* джерел аудіо (наприклад, смартфонів чи комп'ютерів) і приймачів (навушників чи колонок). У цьому режимі забезпечується автоматичний процес створення пари через Bluetooth-стек із підтримкою автентифікації. Для інформування користувача про стан підключення можуть використовуватися індикатори, наприклад, світлодіоди.

Етап *приймання аудіо* починається з отримання аудіо-пакетів від джерел. Дані приймаються блоками, наприклад, по 512 байт, і записуються до відповідних буферів. Під час цього процесу виконується контроль переповнення буфера. Також реалізується моніторинг розривів з'єднання, який виконується через таймери або сигнали Bluetooth-стека.

На етапі *міксування* відбувається обробка аудіосемплів, що надходять від різних джерел. Враховуючи специфіку формату A2DP, основною задачею є коректна робота з PCM-даними, які представляють амплітуду сигналу у вигляді 16-бітних значень. Процес міксування передбачає нормалізацію амплітуд, суму значень семплів із різних потоків та врахування можливих перевищень допустимого діапазону (від -32768 до 32767). Для уникнення артефактів насичення застосовується обмеження значень у межах цього діапазону.

Заключний етап включає *підготовку даних для передачі та їх відправку* приймачу. Після міксування PCM-дані стискаються у формат SBC або інший, підтримуваний профілем A2DP. Готові дані передаються у канал передачі Bluetooth із використанням механізму FIFO для управління буферами.

Запропонований алгоритм може бути вдосконалений шляхом оптимізації обчислень (наприклад, використанням SIMD-інструкцій) або впровадженням механізмів компенсації затримок, що підвищить точність синхронізації між джерелами та приймачем.

При розробці програмного забезпечення для систем обробки аудіопотоків використання операційної системи з підтримкою багатозадачності є

обґрунтованим і доцільним підходом. Застосування багатозадачності дозволяє оптимізувати використання апаратних ресурсів, забезпечуючи їх раціональний розподіл між різними процесами. Крім того, такий підхід значно підвищує стійкість системи до виняткових ситуацій, наприклад, обриву з'єднання або перевантаження буферів, за рахунок швидкого реагування на події через окремі задачі, що функціонують незалежно одна від одної.

Ключовими перевагами використання багатозадачності є:

- Ефективне використання ресурсів таких як, процесорного часу, пам'яті та засобів вводу-виводу.
- Гнучкість і модульність, що забезпечується розподілом функціоналу на окремі задачі, що полегшує їх розробку, тестування та обслуговування.
- Обробка подій у реальному часі, що гарантує мінімальні затримки при прийманні та передачі даних.
- Здатність до розширення, що реалізується через можливість додавати нові функції та задачі, що легко інтегруються до системи без необхідності суттєвих змін у базовій архітектурі.

Реалістичний алгоритм, який враховує всі зазначені переваги, зображено на рисунку 2.5. Цей алгоритм розділений на чотири основні задачі, кожна з яких виконується одночасно в окремому нескінченному циклі. Такий підхід дозволяє ефективно організувати взаємодію між задачами через спільні ресурси, наприклад, буфери або черги повідомлень. У задачах забезпечується синхронізація, уникнення конфліктів доступу до спільних даних, а також обробка подій у реальному часі.



Рисунок 2.5 — Паралельний алгоритм агрегації аудіо-потоків

Структура алгоритму робить систему масштабованою, що дозволяє додавати нові функції чи задачі з мінімальними змінами у програмному коді. Це забезпечує високу адаптивність програмного забезпечення до нових вимог та умов експлуатації.

2.4 Висновки до розділу 2

Використання скаттер-мережі для об'єднання декількох піко-мереж створює гнучку та масштабовану структуру, що дозволяє підключати більше пристроїв і забезпечує стійкість до локальних збоїв.

Розроблений алгоритм демонструє інноваційний підхід до вирішення поставленого завдання.

3 ПІДТВЕРДЖЕННЯ ЗАПРОПОНОВАНОЇ КОНЦЕПЦІЇ ТА РОЗРОБКА ПРОТОТИПУ ДЛЯ АГРЕГАЦІЇ АУДІО-ПОТОКІВ

3.1 Вибір апаратної платформи та інструментів розробки програмного забезпечення

Для підтвердження запропонованої концепції доцільно розробити прототип. Апаратна платформа для нього повинна забезпечувати підтримку Bluetooth-з'єднань та аудіо-профілів (наприклад, A2DP), а також мати достатню потужність для обробки кількох потоків у реальному часі.

Для задач бездротового зв'язку та IoT існують різноманітні мікроконтролери та модулі. Одним із найпоширеніших є модуль Nordic Semiconductor nRF52840, який підтримує стандарт Bluetooth 5.3 і забезпечує дальність дії до 100 метрів за умов прямої видимості. Цей модуль відзначається низьким енергоспоживанням, що є критично важливим для IoT-пристроїв. Він також підтримує функції Thread, Zigbee, NFC і власні протоколи 2,4 ГГц, що робить його універсальним для розумних будинків, сенсорних мереж і носимих пристроїв [31]. Silicon Labs EFR32BG22 підтримує стандарт Bluetooth 5.2 і відзначається винятково низьким енергоспоживанням, яке становить лише 0,6 мкА в режимі очікування. Радіус дії цього трансивера може сягати 200 метрів на відкритому просторі, що робить його ідеальним для розумних пристроїв і медичних додатків. Крім того, модуль підтримує функції безпеки та Bluetooth Mesh, що є важливим для масштабованих мереж [32].

Однією з найпопулярніших апаратних платформ з підтримкою Bluetooth є ESP32. Це мікроконтролер із двоядерним процесором Tensilica Xtensa (LX6), що підтримує Wi-Fi і Bluetooth. Його архітектура забезпечує високу продуктивність при обробці аудіо-потоків, включаючи декодування, мікшування та синхронізацію. ESP32 має вбудовані режими енергозбереження (Light Sleep, Deep Sleep), що дозволяють зменшити споживання енергії під час простою.

Плати розробника ESP32 доступні в різних версіях, кожна з яких має свої особливості та призначення. Bluetooth Classic присутній на більшості моделей, окрім ESP32-S3, ESP32-C3, ESP32-C6. Він забезпечує роботу з профілями A2DP – для передачі аудіо з високою якістю (SBC, AAC) та HFP – для голосових дзвінків. Деякі моделі також підтримують Bluetooth Low Energy та профіль LE Audio. При цьому існують моделі, такі як ESP32-C6, які сертифіковані для Bluetooth LE, але не підтримують LE Audio. Також важливою характеристикою є об'єм оперативної пам'яті для обробки аудіо.

Доцільно обрати плату розробника ESP32-LyraT (рис. 3.1), яка орієнтована на роботу з аудіо-застосунками.

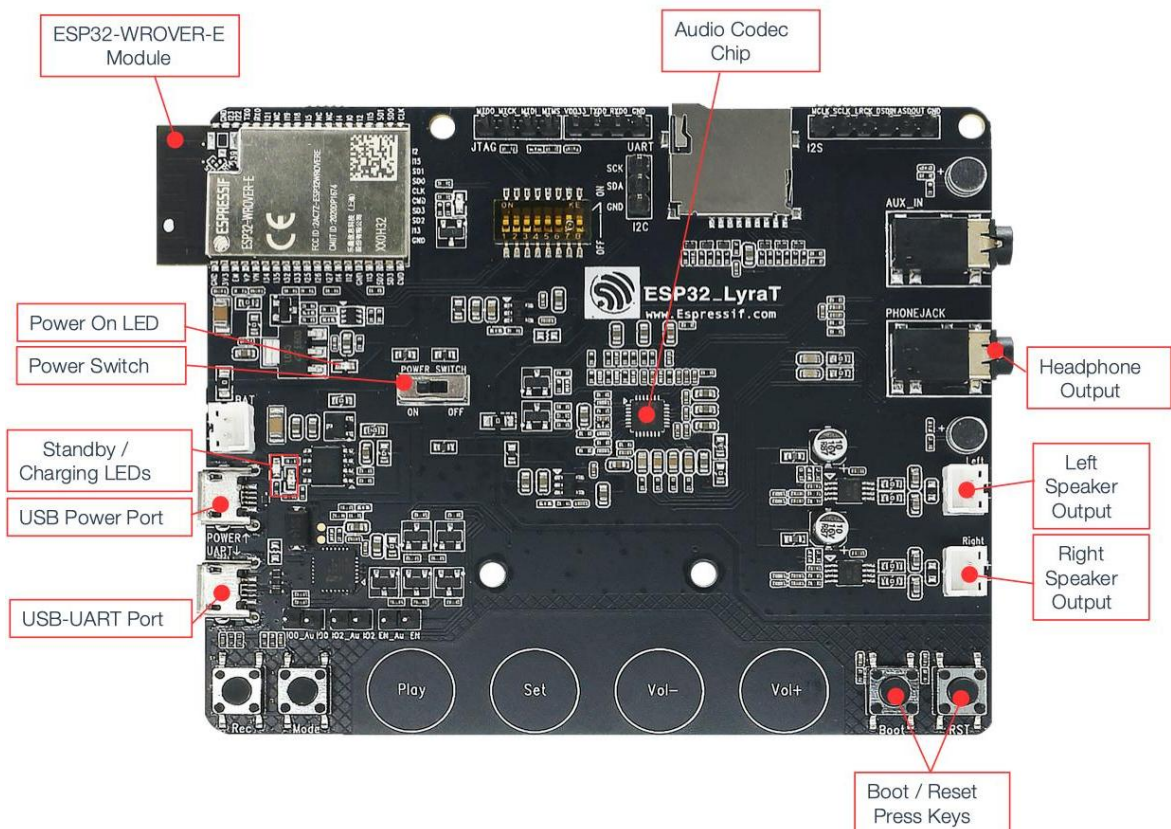


Рисунок 3.1 — Плата розробника ESP32-LyraT [33]

В основі цієї плати знаходиться модуль ESP32-WROWER, який згідно специфікації [34]:

- оснащений ESP32-D0WD-V3 або ESP32-D0WDR2-V3, Xtensa dual-core 32-bit LX6 до 240 МГц;

- має вбудовану Flash-пам'ять до 16 MB;
- відповідає специфікаціям Bluetooth V4.2 BR/EDR та Bluetooth LE.

На момент написання дисертації Espressif Systems надає Bluetooth API для роботи з Bluetooth Classic (GAP, A2DP, AVRC, SPP, HFP, HID, L2CAP, SDP) та Bluetooth LE (GAP, GATT, BluFi), а також фреймворк для роботи з аудіо ESP-ADF, який дає можливість: легко додавати в розроблювані застосунки підтримку різноманітних аудіо-форматів (MP3, AAC, WAV, OGG та ін.); забезпечувати відтворення музики через HTTP, HLS, SDCARD, Bluetooth A2DP/HFP; інтегрувати різноманітні медіа сервіси (напр. WeChat, Internet Radio, помічники Alexa, DuerOS та інше).

Основним інструментом для розробки програмного забезпечення для платформи ESP32 є Arduino IDE. Це середовище, яке має простий інтерфейс і підтримує велику кількість бібліотек, включаючи бібліотеку BluetoothSerial, що є необхідною для реалізації двостороннього зв'язку Bluetooth. Однією з переваг Arduino IDE є його простота у використанні та широка спільнота розробників, що значно полегшує пошук рішень для різних проблем. Крім того, зручність у роботі з бібліотеками дозволяє швидко розпочати розробку і тестування, що особливо важливо для експериментальних і прототипних проектів.

Проте для більш складних і продуктивних рішень можна використовувати ESP-IDF (Espressif IoT Development Framework). ESP-IDF дозволяє розробникам повністю контролювати функціональність пристрою, що дає більше можливостей для налаштування продуктивності і енергоспоживання. Цей інструмент включає в себе всі необхідні бібліотеки для роботи з Bluetooth, Wi-Fi, а також багатозадачність через FreeRTOS, що дозволяє ефективно керувати потоками даних у реальному часі.

Для роботи з Bluetooth аудіо можливо застосувати бібліотеку ESP32-A2DP [35-37], яка надає можливість простого налаштування мікроконтролера як в якості отримувача, так і в якості джерела аудіо.

Для тестування програми без необхідності навантаження реального обладнання можливо використати онлайн-симулятор Wokwi [38]. Він не підтримує апаратні аудіо-кодеки та не дозволяє протестувати Bluetooth-комунікацію, але дозволяє тестувати загальні алгоритми та функції.

3.2 Підготовка коду для реалізації розробленого алгоритму

Бібліотека ESP32-A2DP [35] підтримує профіль для передачі аудіоданих A2DP та відповідає за виконання більшості низькорівневих операцій, пов'язаних із підключенням, встановленням аудіопрофілів та обміном даними між пристроями. Передусім необхідно підключити відповідні файли для роботи з Bluetooth A2DP на ESP32 (рис. 3.2).

```
#include "AudioTools.h"  
#include "BluetoothA2DPSink.h"  
#include "BluetoothA2DPSource.h"
```

Рисунок 3.2 — Підключення бібліотек та створення

AudioTools.h – бібліотека, що містить інструменти для роботи з аудіо, такі як кодування/декодування, обробка та інше. BluetoothA2DPSink.h та BluetoothA2DPSource.h – бібліотеки для роботи з приймачами (sink) та передавачами (source) через Bluetooth A2DP.

Також, на початку необхідно визначити кількість приймачів та передавачів, створивши відповідні об'єкти для кожного з них (рис. 3.3).

```
BluetoothA2DPSink a2dp_sink1;  
BluetoothA2DPSink a2dp_sink2;  
BluetoothA2DPSource a2dp_source;
```

Рисунок 3.3 — Створення приймачів та передавача

Робота пристрою після ввімкнення розпочинається з ініціалізації, яка включає налаштування Bluetooth та характеристик аудіо-оброки, а також створення задач FreeRTOS для їх паралельного виконання під час подальшої роботи системи. В функції setup() необхідно ініціалізувати кожен з аудіо-приймачів задавши їм імена, за якими до них можливо підключити різні

джерела аудіо (наприклад комп'ютер та смартфон). На рисунку 3.4 приймачам задано імена Sink1 та Sink2.

```
a2dp_sink1.start("Sink1");
a2dp_sink2.start("Sink2");
```

Рисунок 3.4 — Ініціалізація A2DP приймачів Sink1 та Sink2

При ініціалізації джерела аудіо необхідно вказати ім'я або вектор імен, що містить перелік всіх імен до яких це джерело буде намагатись підключитись для передачі їм аудіо-даних. На рисунку 3.5 зображено створення вектора з іменами: BTSpeaker, EarBuds, AirPods. Підключення відбувається до першого доступного пристрою із них.

```
static std::vector<char*> bt_names = {" BTSpeaker", "EarBuds", "AirPods"};
a2dp_source.start(bt_names, get_sound_data);
```

Рисунок 3.5 — Ініціалізація A2DP джерела

Bluetooth Classic підтримує SDP (Service Discovery Protocol), який дозволяє дізнатися, які сервіси підтримує підключений пристрій. Після підключення ESP32 запитує список доступних профілів на пристрої та ідентифікує його як джерело аудіо або як приймач. Підтримуваним аудіо-кодеком в ESP32 A2DP є SBC. В API платформи використовуються PCM дані (сирі аудіо-дані без компресії), які зазвичай мають частоту дискретизації (Sampling Rate) 44,1 кГц (сигнал оцифровується 44100 разів в секунду). Аудіо-дані, які слід передавати по Bluetooth визначаються за допомогою «бібліотечної» callback функції `get_sound_data` [35], яка є аргументом `a2dp_source.start` та відповідає за генерацію звуку.

Для кожного аудіо-потoku необхідно створити буфери. Це можливо реалізувати за допомогою механізму черг в FreeRTOS (рис. 3.6), оптимальний розмір яких необхідно підібрати експериментальним шляхом.

```
QueueHandle_t audioQueue1 = xQueueCreate(x, sizeof(int));
QueueHandle_t audioQueue2 = xQueueCreate(x, sizeof(int));
```

Рисунок 3.6 — Створення черг для кожного аудіо-потoku

Для синхронізації доступу завдань FreeRTOS до спільних ресурсів доцільно використовувати семафор (рис. 3.7).

```
SemaphoreHandle_t aggregationSemaphore = xSemaphoreCreateBinary();
xSemaphoreGive(aggregationSemaphore);
```

Рисунок 3.7 — Створення та звільнення семафору

Для розподілення ресурсів ESP32 між усіма наступними частинами алгоритму необхідно визначити основні завдання FreeRTOS та callback функції для:

- управління Bluetooth-з'єднаннями;
- прийому аудіо від кожного джерела;
- агрегування;
- передачі даних до приймача.

В подальшому можливо доопрацювати програму для узгодження різних аудіо-кодеків (під час встановлення A2DP-з'єднання пристрої обмінюються інформацією про підтримувані кодеки та обирається найбільш сумісний та якісний), а також впровадження функцій логування стану, відстежування проблем та реагування на різноманітні події, такі як підключення пристроїв, розрив з'єднання та ін.

Для створення визначених завдань можливо використовувати функцію FreeRTOS `xTaskCreatePinnedToCore` [39] (рис. 3.8).

```
xTaskCreatePinnedToCore(blueetoothManagerTask, "BluetoothManager", 10000, NULL, 2, NULL, 0);
xTaskCreatePinnedToCore(audioReceiver1Task, "AudioReceiver1", 10000, NULL, 1, NULL, 0);
xTaskCreatePinnedToCore(audioReceiver2Task, "AudioReceiver2", 10000, NULL, 1, NULL, 0);
xTaskCreatePinnedToCore(audioAggregatorTask, "AudioAggregator", 10000, NULL, 2, NULL, 1);
xTaskCreatePinnedToCore(audioTransmitterTask, "AudioTransmitter", 10000, NULL, 3, NULL, 1);
```

Рисунок 3.8 — Створення завдань FreeRTOS

Підібрати оптимальні розміри стеку, пріоритети кожного з завдань, розподілити навантаження між ядрами CPU необхідно експериментальним шляхом.

Для керування Bluetooth-з'єднаннями окреме завдання повинно постійно перевіряти підключення та намагатись їх відновити. В наведеному коді (рис.3.9) використовується функція `Is_connected` для послідовної перевірки та відновлення з'єднання кожну секунду.

```
void bluetoothManagementTask(void *param) {
    while (true) {
        Is_connected(a2dp_source);
        Is_connected(a2dp_sink1);
        Is_connected(a2dp_sink2);
        vTaskDelay(pdMS_TO_TICKS(1000));
    }
}
```

Рисунок 3.9 — Код для перевірки Bluetooth-з'єднання

Кількість завдань для обробки вхідних даних повинна відповідати кількості джерел аудіо. Для обробки отриманих даних від кожного з джерел можливо призначити відповідну callback-функцію (рис. 3.10). Тобто, при отриманні даних від джерела `a2dp_sink1` буде викликатись функція `read_audio_stream1`, параметрами якої є отримані дані та їх довжина.

```
a2dp_sink1.set_stream_reader(read_audio_stream1);
```

Рисунок 3.10 — Призначення callback-функції

За агрегацію аудіо даних з двох джерел, їх обробку та запис у чергу для передачі повинно відповідати завдання `audioAggregatorTask`, а за передачу готових даних із черги до приймача – завдання `audioTransmitterTask`.

Дані з кожного джерела A2DP отримуються у вигляді масивів `int16_t` (16-бітні значення). Перед міксуванням необхідно переконатися, що частота дискретизації (*sampling rate*) та формат даних для всіх джерел однаковий. Виконувати міксування можливо різними методами, одним з найпростіших є

обчислення середнього арифметичного, коли семпл з джерела 1 додається до відповідного семплу з джерела 2, а результат ділиться на 2 (рис. 3.11).

```
int16_t mix_sample(int16_t sample1, int16_t sample2) {
    return (sample1 + sample2) / 2;
}
```

Рисунок 3.11 — Функція простого міксування окремих семплів

В подальшому результат міксування записується в відповідний буфер або одразу направляється для передачі по Bluetooth.

3.3 Створення моделі для доказу концепції з використанням онлайн-симулятора

Використання бібліотеки для роботи з Bluetooth-аудіо дозволяє зосередитися на створенні логіки обробки даних та їх агрегування. Для відлагодження та тестування частини пов'язаної з Bluetooth-комунікацією необхідно використання реального апаратного забезпечення. Роботу основної логіки багатозадачної програми можливо перевірити без використання фізичного обладнання, за допомогою онлайн симулятора Wokwi. Для цього створено спрощену модель (рис. 3.12) пристрою, яка включає:

- ESP32 – в якості центрального вузла скаттер-мережі;
- Кнопки – для генерації даних. При натисканні кожної з них генерується окремий набір числових послідовностей, які імітують аудіосигнал.
- Buzzer (п'єзодинамік) – для відтворення агрегованих даних, імітуючи роботу вихідного аудіопристрою.

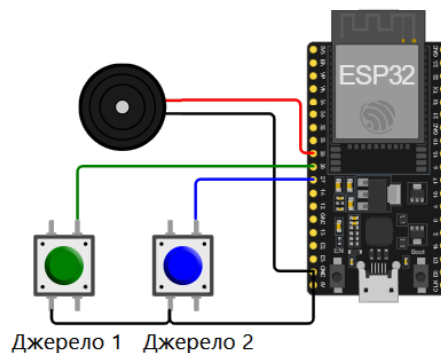


Рисунок 3.11 — Спрощена модель агрегатора аудіо-потоків

Діаграму в формат JSON, для відтворення зображеної на рисунку моделі, наведено в додатку А. Програмна частина розробленої моделі на мові C++ міститься в додатку Б. Оскільки симулятор Wokwi не дозволяє працювати з Bluetooth, отримання даних від кожного з джерел моделюється за допомогою окремих задач `audio1Task` та `audio2Task`. Кожна з них відповідає за генерацію числових послідовностей, що імітують вхідні аудіопотоки. Кожна з задач починає генерацію вхідних даних після натискання відповідної кнопки. Перша створює послідовні частоти 100 та 200 Гц, друга реалізує мелодію укладену в масиві. Дані з обох джерел надходять у черги.

При наявності даних в чергах, завдання присвячене мікшуванню обчислює середню частоту двох сигналів. Якщо дані присутні лише в одній з черг, то саме вони будуть відтворюватись. При роботі з реальними джерелами аудіо на цьому етапі відбуватиметься мікшування з семплів (вибірок). Семпл не містить інформації про частоти, а представляє амплітуду звукової хвилі в конкретний момент часу. Мікшування забезпечує гармонічне поєднання звуків із різних джерел, що сприймається людським вухом як єдиний сигнал.

Для дослідження цифрових сигналів в симуляторі Wokwi доступний віртуальний логічний аналізатор. Сигнал від першого джерела, згенерований у завданні `audio1Task` та відтворений за допомогою функції `tone` зображено на рисунку 3.13. Він свідчить, що для відтворення визначеного тону динаміком використовується широтно-імпульсна модуляція.

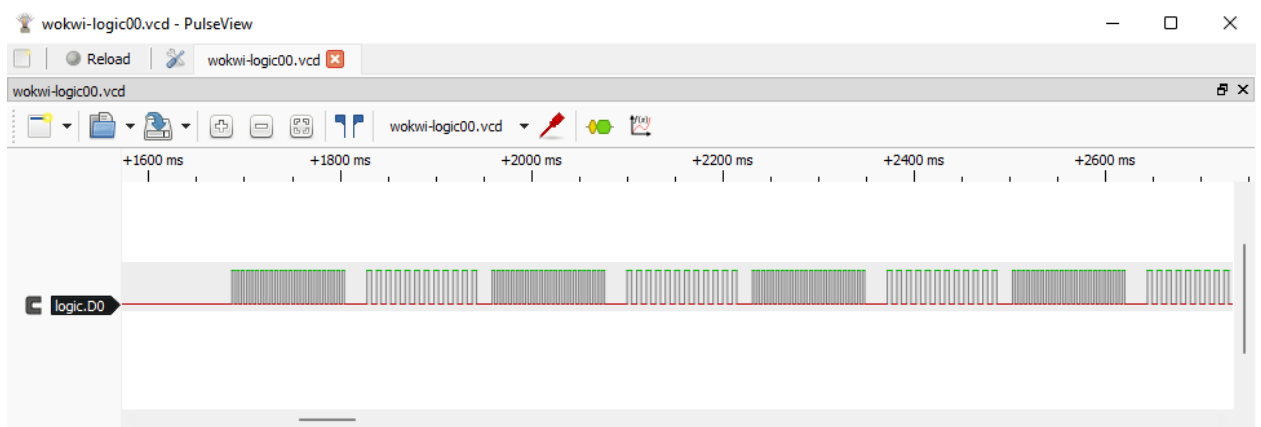


Рисунок 3.13 — Сигнал від джерела 1

На рисунку 3.14 зображено сигнал для відтворення мелодії з відповідного масиву.



Рисунок 3.14 — Сигнал від джерела 2

На рисунку 3.15 зображено результати міксування.

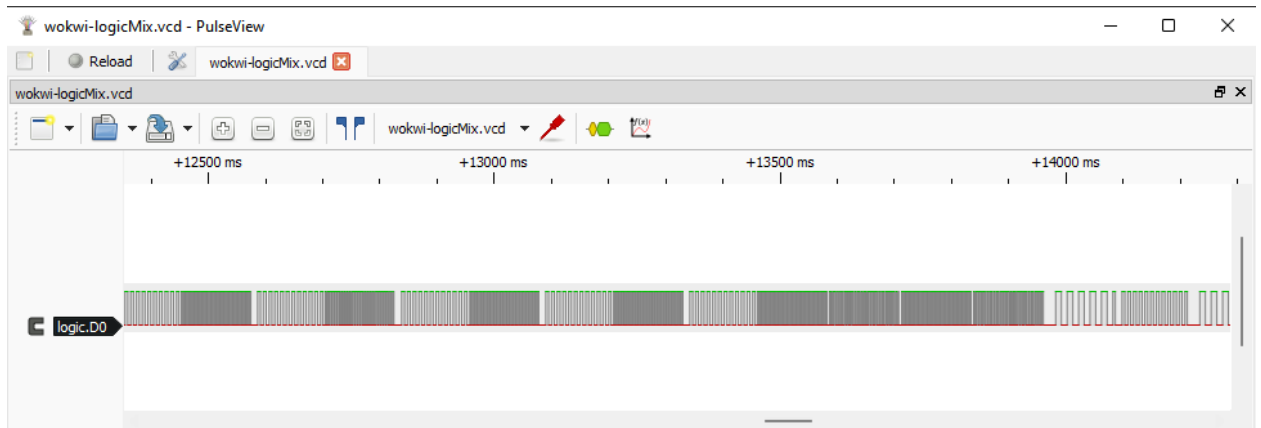


Рисунок 3.15 — Сигнал після міксування

Зміст серійного монітору під час послідовного запуску генерації джерелом 1 та джерелом 2 і виконання `playbackTask` зображено на рисунку 3.16.

```

Hello, ESP32!
Частота 1:200
Частота 1:100
Частота 1:200
Частота 1:100
Частота 1:200
Частота 1:100
Частота 1:200
Частота 1:100 Частота 2:659 Середня:379
Частота 1:200 Частота 2:0 Середня:100
Частота 1:100 Частота 2:659 Середня:379
Частота 1:200 Частота 2:0 Середня:100
Частота 1:100 Частота 2:659 Середня:379
Частота 1:200 Частота 2:659 Середня:429
Частота 1:100 Частота 2:0 Середня:50
Частота 1:200 Частота 2:0 Середня:100
Частота 1:100 Частота 2:0 Середня:50
Частота 1:200 Частота 2:659 Середня:429
Частота 1:100 Частота 2:0 Середня:50
Частота 1:200 Частота 2:659 Середня:429
Частота 1:100 Частота 2:0 Середня:50
Частота 1:200 Частота 2:659 Середня:379
Частота 1:200 Частота 2:0 Середня:100
Частота 1:100 Частота 2:0 Середня:50
Частота 1:200 Частота 2:0 Середня:100
Частота 1:100 Частота 2:659 Середня:379
Частота 1:200 Частота 2:0 Середня:100
Частота 2:784
Частота 2:0

```

Рисунок 3.16 — Зміст серійного монітору під час виконання програми

Розроблена модель продемонструвала можливість керування кількома аудіо-потоків з використанням можливостей операційної системи FreeRTOS та багатозадачного підходу. Подальші кроки можуть включати тестування цієї системи в реальних умовах та адаптацію до використання в поєднанні з комунікацією по Bluetooth.

3.4 Висновки до розділу 3

Платформа ESP32 виявилась доволі ефективною завдяки високій продуктивності, наявності апаратних інтерфейсів та простоті інтеграції з бібліотеками для роботи з аудіо-профілями та Bluetooth-з'єднаннями.

Підтримка FreeRTOS дозволила ефективно управляти паралельним виконанням декількох завдань та працювати з буферами (чергами).

Створена в онлайн-симуляторі модель продемонструвала життєздатність розробленої концепції, що дозволяє в подальшому перейти до тестування в реальних умовах та адаптації програмного забезпечення для впровадження рішення з використанням Bluetooth на реальному обладнанні

4 СТАРТАП-ПРОЕКТ

4.1 Опис ідеї проекту

Система агрегації аудіопотоків у Bluetooth-мережах, що забезпечує одночасну передачу кількох потоків із різних джерел до одного приймача, створена для мультимедійних систем, IoT-пристроїв та професійного звукового обладнання. Система використовує інноваційний гібридний алгоритм агрегації, що по закінченню часу і частотного розділення потоків, інтегрованих у скаттер-мережу.

Таблиця 4.1 – Опис ідеї стартап-проекту

Зміст ідеї	Напрямки застосування	Вигоди для користувача
Розробка системи агрегації аудіопотоків у Bluetooth-мережах	Домашні мультимедійні системи, IoT, професійне звукове	Висока якість передачі аудіо, одночасна робота з кількома джерелами, зменшення затрат

З таблиці видно, що ідея стартапу є актуальною, оскільки забезпечує якісну передачу аудіопотоків із мінімальними затримками, підвищує зручність використання мультимедійних систем і розширює можливості інтеграції Bluetooth-мереж у різні сфери.

Таблиця 4.2 – Визначення сильних, слабких та нейтральних характеристик ідеї проекту

№	Техніко-економічні Характеристики ідеї	(потенційні) товари/концепції конкурентів	W (слабка сторона)	N (нейтральна сторона)	S (сильна сторона)
1	Ціна, грн/од.	1200	1500	2000	+
2	Пропускна здатність, Мбіт/с	2	1,5	1,8	+
3	Якість звуку (суб'єктивна шкала)	9	7	8	+
4	Масштабованість	Висока	Середня	Середня	+
5	Енергоефективність	Висока	Середня	Середня	+

Після аналізу характеристик ідеї проєкту було визначено, що сильними сторонами проєкту є висока пропускна здатність, якість звуку та конкурентна ціна. Нейтральними є енергоефективність і масштабованість. Слабких сторін характеристик не виявлено.

4.2 Технологічний аудит ідеї проєкту

Таблиця 4.3 – Технологічна здійсненність ідеї проєкту

№	Ідея проєкту	Технології її реалізації	Наявність технології	Доступність технологій
1	Розробка системи агрегації аудіопотоків у Bluetooth-мережах	Використання гібридного алгоритму агрегації	+	+
2		Bluetooth 5.2 з підтримкою LC3-кодека для енергоефективності аудіопередачі	+	+
3		Використання скатер-мереж для масштабованості системи	+	+
4		Реалізація	+	+

Висновок: Обрана технологія реалізації ідеї проєкту включає використання сучасних апаратних та програмних рішень, які відповідають поставленим завданням. Основними перевагами технології є:

1. Енергоефективність: Використання LC3-кодека забезпечує якісну передачу аудіопотоків із мінімальними витратами енергії.
2. Масштабованість: Інтеграція скатер-мереж дозволяє легко розширювати систему, додаючи нові пристрої без втрати продуктивності.
3. Швидкість реалізації: Доступність мікроконтролерів ESP32, відкритих бібліотек і підтримка Bluetooth 5.2 значно скорочує час на розробку прототипу.
4. Стабільність: Використання стандартів Bluetooth 5.2 забезпечує високу якість зв'язку навіть у середовищах із великою кількістю підключень.

Технологія дозволяє реалізувати інтегровану систему агрегації аудіопотоків, яка відповідає вимогам до сучасних мультимедійних та IoT-пристроїв. Вона базується на доступних та перевірених технологіях, що гарантує її стабільну роботу і швидке впровадження на ринку.

4.3 Аналіз ринкових можливостей запуску стартап-проекту

Ринок системи агрегації аудіопотоків у Bluetooth-мережах є перспективним завдяки активному розвитку IoT-пристроїв, мультимедійних платформ і професійного звукового обладнання. Актуальність продукту зумовлена зростаючими потребами в інтеграції аудіопотоків із різними джерелами, що забезпечують високу якість звуку та енергії.

Таблиця 4.4 – Попередня характеристика потенційного ринку стартап-проекту

№	Показники стану ринку (найменування)	Характеристика
1	Кількість головних гравців, од	4
2	Загальний обсяг продаж, од/рік	10000
3	Динаміка ринку (якісна оцінка)	Зростає
4	Наявність обмежень для входу (вказати характер обмежень)	Висока конкуренція
5	Специфічні вимоги до стандартизації та сертифікації	Відповідність стандартам Bluetooth SIG
6	Середня норма рентабельності в галузі (або по ринку), %	15 %

Висновок: Ринок систем для агрегації аудіопотоків має високий потенціал. Незважаючи на певну конкуренцію, інноваційний характер стартапу та зростаюча потреба в енергоефективних рішеннях забезпечують перспективність виходу на цей ринок.

Таблиця 4.5 – Характеристика потенційних клієнтів стартап-проекту

№	Потреба, що формує ринок	Цільова аудиторія (цільові сегменти ринку)	Відмінності у поведінці різних потенційних цільових груп клієнтів	Вимоги споживачів до товару
1	Інтеграція аудіопотоків у мультіплексор	Виробники мультимедійних платформ	Висока якість звуку, готовність платити більше за інновації	Висока якість передачі, енергоефективність
	IoT-рішення	Виробники IoT-пристроїв	Орієнтація на енергоефективність	Низьке енергоспоживання, надійність
	Професій	Аудіокомпанії та студії звукозапису	Вимоги до мінімальних затримок та високої продуктивності	Стабільність передачі, синхронізація

Таблиця 4.6 – Фактори загроз та можливостей

№	Фактор	Зміст загрози	Зміст можливості
1	Висока конкуренція	Можливість втрати частки ринку	Позиціонування як інноваційного продукту
2	Зміна стандартів Bluetooth	Може ускладнити адаптацію продукту	Можливість випередити конкурентів у впровадженні нових стандартів
3	Розвиток ринку IoT	Поява нових конкурентів	Розширення ринку
4	Економічна криза	Скорочення платоспроможності клієнтів	Попит на енергоефе

Таблиця 4.7 – Ступеневий аналіз конкуренції на ринку

№	Особливості конкурентного середовища	В чому проявляється дана характеристика	Вплив на діяльність підприємства (можливі дії компанії, щоб бути конкурентоспроможною)
1	Тип конкуренції: монополія	Відсутність аналогічних продуктів у даній ніші	Можливість зайнятості унікальну по
2	Рівень боротьби: назький	Український ринок поки що мало зайнятий	Формування конкурентних переваг через інновації
3	Галузева ознака	Інтеграція з мультимедійними та IoT-рішеннями	Розширення партнерських зв'язків

Таблиця 4.8 – Обґрунтування факторів конкурентоспроможності

№	Фактор конкурентоспроможності	Обґрунтування (наведення чинників, що роблять фактор для порівняння конкурентних проектів значущим)
1	Ціна	Продукт має нижчу ціну порівняно з існуючими рішеннями, оскільки використовуються енергоефективні технології.
2	Термічна стійкість, °C	Завдяки LC3-кодеку, споживання енергії зменшується на 30% порівняно з конкурентами.
3	Міцність на злам, МПа	Використання скаттер-мереж дозволяє легко розширювати систему без суттєвих витрат.
4	Густина матеріалу, г/см ³	Інтеграція сучасного LC3-кодека забезпечує високу якість звуку навіть при низькій швидкості передачі даних.

Таблиця 4.9 – Порівняльний аналіз сильних та слабких сторін продукту

№	Фактор конкурентоспроможності	Бали 1-20	Рейтинг товарів-конкурентів у порівнянні з продуктом						
			-3	-2	-1	0	+1	+2	+3
1	Ціна	18		+					
2	Енергоефективність	15		+					
3	Енергоефективність	12		+					
4	Якість звуку	20	+						

Таблиця 4.10 – SWOT- аналіз стартап-проекту

Сильні сторони: Відносно низька ціна та висока якість звуку, Використання сучасних технологій Bluetooth 5.2 з LC3-кодеком. Масштабованість системи за рахунок скаттер-мереж.	Слабкі сторони: Висока вартість впровадження технології на початкових етапах. Невеликий досвід у просуванні подібних продуктів на ринку. Відсутність власних сертифікаційних стандартів для міжнародного ринку.
Можливості: 1. Розширення ринку IoT і мультимедійних систем. 2. Підвищення попиту на енергоефективні рішення	Загрози: 1. Поява міжнародних конкурентів у ніші. 2. Потенційні зміни стандартів Bluetooth

Таблиця 4.11 – Альтернативи ринкового впровадження стартап-проекту

№	Альтернатива (орієнтовний комплекс заходів) ринкової поведінки	Ймовірність отримання ресурсів	Строки реалізації, міс
1	Впровадження стандартів якості продукту	Використання LC3-кодека та сучасних мікроконтролерів ESP32	1
2	Розширення партнерської бази.	Співпраця з IoT-розробниками та виробниками аудіообладнання	1-3
3	Розширення функціоналу продукту	Додавання можливостей для відеопередачі та інтеграції в Wi-Fi	1-3

Висновок: Враховуючи проведений аналіз, найоптимальнішим варіантом є впровадження стандартів якості продукту через використання сучасних технологій, таких як LC3-кодек та ESP32. Це дозволить забезпечити високу якість передачі аудіо та конкурентоспроможність на ринку з мінімальними додатковими витратами.

4.4 Розроблення ринкової стратегії проекту

Таблиця 4.12 – Вибір цільових груп потенційних споживачів

№	Опис профілю цільової групи потенційних клієнтів	Готовність споживачів сприйняти продукт	Орієнтовний попит в межах цільової групи (сегменту)	Інтенсивність конкуренції в сегменті	Простота входу у сегмент
1	Виробники мультимедійних платформ	Готові	500 шт/міс	Помірна	Відносно проста
2	Виробники IoT-пристроїв	Готові	300 шт/міс	Висока	Складна
3	Компанії з виробництва професійного звукового обладнання	Готові	200 шт/міс	Низька	Проста

Висновок: Аналізуючи потенційні групи споживачів, обрано сегмент виробників мультимедійних платформ, як основний ринок для початкового впровадження. Відповідно, стратегія охоплення ринку – стратегія концентрованого маркетингу.

Таблиця 4.13 – Визначення базової стратегії розвитку

№	Обрана альтернатива розвитку проекту	Стратегія охоплення ринку	Ключові конкурентоспроможні позиції відповідно до обраної альтернативи	Базова стратегія розвитку*
1	Стратегія спеціалізації	концентрованого маркетингу	- Продукт поєднує високу якість звуку, енергоефективність та масштабованість системи. - Продукт здатен конкурувати завдяки ціні, яка нижча за аналогічні пропозиції	Стратегія спеціалізації

Таблиця 4.14 – Визначення базової стратегії конкурентної поведінки

№	Чи є проект «першопрохідцем» на ринку?	Чи буде компанія шукати нових споживачів, або забирати існуючих у конкурентів?	Чи буде компанія копіювати основні характеристики Товару конкурента, і які?	Стратегія конкурентної поведінки*
1	Так	Буде шукати нових і спробує забрати існуючих у конкурентів	Продукт є унікальним за рахунок агрегації потоків та високої енергоефективності	Стратегія лідера

Таблиця 4.15 – Визначення стратегії позиціонування

№	Вимоги до товару цільової аудиторії	Базова стратегія розвитку	Ключові конкурентоспроможні позиції власного стартап-проекту	Вибір асоціацій, які мають сформувати комплексну позицію власного проекту (три ключових)
1	Надійність, енергоефективність, масштабованість	Стратегія спеціалізації	Ціна, Якість звуку, Масштабованість	Надійний, Інноваційний, Енергоефективний

Висновок: У результаті аналізу обрано стратегію конкурентної поведінки – стратегію лідера. Базова стратегія розвитку – стратегія спеціалізації, а стратегія охоплення ринку – стратегія концентрованого маркетингу. Продукт має асоціюватися з такими характеристиками: Надійний, Інноваційний, Енергоефективний.

4.5 Розроблення маркетингової програми стартап-проекту

Таблиця 4.16 – Визначення ключових переваг концепції потенційного товару

№	Потреба	Вигода, яку пропонує товар	Ключові переваги перед конкурентами (існуючі або такі, що потрібно створити)
1	Інтеграція кількох аудіопотоків у Bluetooth-мережах	Ефективне мультиплексування з мінімальними затримками	- Низьке енергоспоживання завдяки LC3-кодеку. - Висока якість звуку навіть при низькій швидкості передачі. - Масштабованість завдяки скаттер-мережам

Таблиця 4.17 – Опис трьох рівнів моделі товару

Рівні товару	Сутність та складові
I. Товар за задумом	Забезпечення інтеграції кількох аудіопотоків із високою якістю звуку та мінімальними затримками Задовольняє потребу в об'єднанні аудіопотоків із різних джерел для якісної передачі через Bluetooth
II. Товар із функціональними характеристиками	Властивості/характеристики: - Енергоефективність LC3-кодека (економія енергії на 30 %); - Висока якість звуку; - Масштабованість через скаттер-мережі для збільшення кількості пристроїв у системі. - Швидкість передачі: 2 Мбіт/с.
III. Товар із підкріпленням	Якість: відповідність стандартам Bluetooth 5.2. Пакування: Брендowana упаковка, інструкції для користувачів.. Марка: "SmartStream", назва товару — «BlueAudio Agregator». Післяпродажне обслуговування: Гарантія на 12 місяців, технічна підтримка після продажу. Захист від копіювання: оформлення патенту на унікальний алгоритм агрегації.

Таблиця 4.18 – Визначення меж встановлення ціни

№	Рівень цін на товари-замінники	Рівень цін на товари-аналоги, грн	Рівень доходів цільової групи споживачів, грн	Верхня та нижня межі встановлення ціни на товар/послугу, грн
1	Таких немає	10000	15000-50000	8000-12000

Таблиця 4.19 – Формування системи збуту

№	Специфіка закупівельної поведінки цільових клієнтів	Функції збуту, які має виконувати постачальник товару	Глибина каналу збуту	Оптимальна система збуту
1	Зацікавлені у тривалому партнерстві	Пряма доставка, технічна підтримка	Внутрішня	Власна система збуту

Таблиця 4.20 – Концепція маркетингових комунікацій

№	Специфіка поведінки цільових клієнтів	Канали комунікацій, якими користуються цільові клієнти	Ключові позиції, обрані для позиціонування	Завдання рекламного повідомлення	Концепція рекламного звернення
1	Зацікавлені у співпраці Зацікавлені у нових технологіях та інноваціях	Сайти, тематичні форуми, телеграм-канали, Профільні конференції,	Надійний, енергоефективний, масштабований	Пошук нових клієнтів та партнерів	Донести основні ключові характеристики продукту

4.6 Висновки до розділу 4

Запропонована система агрегації аудіопотоків у Bluetooth-мережах має значний потенціал для впровадження на ринок завдяки своїй інноваційності та відповідності сучасним вимогам до якості передачі аудіоданих і енергоефективності.

1. Розроблена маркетингова стратегія дозволяє сконцентрувати ресурси на найбільш перспективних сегментах ринку, таких як виробники мультимедійних платформ, що сприятиме швидкому впровадженню продукту.

2. Конкурентні переваги товару, зокрема низька ціна, масштабованість і підтримка скаттер-мереж, створюють умови для формування стійкої позиції на ринку.

3. Успішна реалізація маркетингової програми залежить від ефективної комунікації з потенційними клієнтами та розробки унікального позиціонування, яке виділить продукт серед конкурентів.

Встановлені межі ціноутворення та обрані канали збуту забезпечують баланс між рентабельністю проєкту та доступністю продукту для споживачів.

ВИСНОВКИ

Аналіз сучасних технологій управління потоками даних у Bluetooth-мережах засвідчив, що існуючі підходи в основному обмежуються передачею аудіо від одного джерела до одного приймача, що унеможлиблює одночасне використання кількох пристроїв без додаткового обладнання.

Інноваційність рішення заключається у розробленому варіанті мережевої архітектури, що дозволяє створити окремі піко-мережі для бездротових джерел аудіо-сигналів і приймача та об'єднати їх у єдину скаттер-мережу за допомогою моста-агрегатора. Це дозволяє одночасно використовувати кілька джерел аудіо, вирішуючи проблему, яка традиційно не вирішується в межах стандартного використання технології Bluetooth.

Поставлені завдання дисертації виконано, а саме: проаналізовано існуючі обмеження та можливості технології Bluetooth; розроблено схему мережі та алгоритми роботи її центрального вузла; реалізовано програмно-апаратну комп'ютерну модель в середовищі онлайн-симуляції та здійснено її перевірку; розроблено стартап-проект.

Використання комп'ютерного моделювання дозволило підтвердити життєздатність запропонованої концепції за допомогою забезпеченої операційною системою реального часу багатозадачності. Це дозволяє перейти до адаптації програмного забезпечення для впровадження на реальному обладнанні та тестування в реальних умовах.

Запропоноване рішення має високий потенціал практичного застосування в комерційній сфері для побудови новітньої аудіо-техніки, домашніх мультимедійних систем та пристроїв IoT.

Отримані результати можуть стати основою для подальших досліджень спрямованих на оптимізацію апаратного забезпечення для мінімізації енергоспоживання і затримок, підвищення пропускну здатності та розширення функціональних можливостей.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Zeadally S. 25 years of bluetooth technology [Electronic resource]/ Zeadally S., Siddiqui F., Baig Z. //Future Internet. – 2019. – Vol. 11. – №. 9. – С. 194.
2. Woolley M. Bluetooth Low Energy Primer. Bluetooth Blog. 2022, Доступно в Інтернеті: <https://www.bluetooth.com/blog/introducing-the-bluetooth-low-energy-primer/>.
3. 7. Макаренко А.Ю. Бездротові технології передачі даних Wi-Fi, Bluetooth та ZigBee. / А.Ю. Макаренко, А.О. Парфенова, С.Б. Могильний // Вісник НТУУ «КПІ». Серія Радіотехніка. Радіоапаратобудування. – 2010. – № 41. – С. 171-181.
4. Pecunia V. Emerging Indoor Photovoltaic Technologies for Sustainable Internet of Things. / Pecunia V., Occhipinti L. G., Hoyer R. L. Z. // Advanced Energy Materials. – 2021. – Vol. 11, – №. – P. 29. 2100698. URL: <https://doi.org/10.1002/aenm.202100698>.
5. Д'яченко С. Ф. Бездротовий контроль температури за допомогою bluetooth технології з використанням мікропроцесорної техніки. 25-й Міжнародний молодіжний форум «Радіoeлектроніка та молодь у XXI столітті». Зб. матеріалів форуму. Т. 2. – Харків: ХНУРЕ. – 2021. – С. 29-30.
6. Fayyoubi E. L2CAP-based prototype media streaming via Bluetooth technology / E. Fayyoubi et al.// International Journal of Networking and Virtual Organisations. – 2014. – Vol. 14, – №. 3. – P. 221. URL: <https://doi.org/10.1504/ijnvo.2014.065785>.
7. Locatelli P. Device discovery and tracing in the Bluetooth Low Energy domain / P. Locatelli et al. // Computer Communications. – 2023. URL: <https://doi.org/10.1016/j.comcom.2023.02.008>.
8. Bensky A. Wireless local area networks / A. Bensky // Short-range Wireless Communication.– Elsevier Inc. – 2019. – P. 273–315. URL: <https://doi.org/10.1016/c2017-0-02356-x>.

9. Sahoo P. K. Novel route maintenance protocols for the Bluetooth ad hoc network with mobility/ P. K. Sahoo, C.-Y. Chang, S.-W. Chang // Journal of Network and Computer Applications. – 2008. – Vol. 31, – № 4. – P. 535–558. URL: <https://doi.org/10.1016/j.jnca.2007.12.001>.
10. Tahir S., Bakhsh S. T., Altalhi A. H. An Efficient Route Maintenance Protocol for Dynamic Bluetooth Networks / S. Tahir, S. T. Bakhsh, A. H. Altalhi // Journal of King Saud University - Computer and Information Sciences. – 2017. – Vol. 29, – №. 4. P. 449–461. URL: <https://doi.org/10.1016/j.jksuci.2016.07.003>.
11. Hirsch C. DynGATT: A dynamic GATT-based data synchronization protocol for BLE networks / C. Hirsch et al. // Computer Networks. – 2023. – Vol. 222. – P. 109560. URL: <https://doi.org/10.1016/j.comnet.2023.109560>.
12. Zeadally S. Design, implementation, and evaluation of the audio/video distribution transport protocol (AVDTP) for high quality audio support over Bluetooth / Zeadally S., Kumar A. // Computer Communications. 2005. Vol. 28, no. 2. P. 215–223. URL: <https://doi.org/10.1016/j.comcom.2004.09.007>.
13. Mansour M. Internet of Things: A Comprehensive Overview on Protocols, Architectures, Technologies, Simulation Tools, and Future Directions / M. Mansour et al. // Energies. – 2023. – Vol. 16, – № 8. – P. 3465. URL: <https://doi.org/10.3390/en16083465>.
14. Zhang J. The application of bluetooth technology in the internet of things / J. Zhang // Applied and Computational Engineering. – 2023. – Vol. 12, – № 1. – P. 177–183. URL: <https://doi.org/10.54254/2755-2721/12/20230334>.
15. Grid-Connected DFIG-Based Wind Energy Conversion System with ANFIS Neuro-Fuzzy Controller / M. Vyas et al. // Smart Energy and Advancement in Power Technologies. Lecture Notes in Electrical Engineering. Springer, Singapore. – 2023. – Vol. 927. URL: https://doi.org/10.1007/978-981-19-4975-3_48.
16. A traffic aggregation and differentiation scheme for enhanced QoS in IEEE 802.11-based Wireless Mesh Networks / R. Riggio et al. // Computer

Communications. – 2008. – Vol. 31, – № 7. – P. 1290–1300. URL: <https://doi.org/10.1016/j.comcom.2008.01.037>.

17. Analyse des pratiques en réunion de concertation pluridisciplinaire thorax (RCP) : impact du TEP-TDM au 18FDG sur le stade TNM / J. Hureaux et al. // Revue des Maladies Respiratoires. – 2006. – Vol. 23. – P. 20. URL: [https://doi.org/10.1016/s0761-8425\(06\)72122-2](https://doi.org/10.1016/s0761-8425(06)72122-2).

18. Що таке Multipoint Bluetooth? [Електронний ресурс]. – Режим доступу: <https://itech.co.ua/novyny/shcho-take-multipoint-bluetooth/> – Назва з екрана.

19. What Is Multipoint Bluetooth. [Електронний ресурс] – Режим доступу: <https://robots.net/tech/what-is-multipoint-bluetooth/> – Назва з екрана.

20. Все про Bluetooth: Технологія, Використання та Адаптери. [Електронний ресурс]. – Режим доступу: https://creativnost.com.ua/vse-pro-bluetooth-texnologiya-vikoristannya-ta-adapteri/#google_vignette. – Назва з екрана.

21. Типи шлюзів Bluetooth на ринку в 2024 році. [Електронний ресурс]. – Режим доступу: <https://www.dusuniot.com/uk/blog/types-of-bluetooth-gateways/>. – Назва з екрана.

22. August wireless dual bluetooth audio transmitter & receiver. [Електронний ресурс]. – Режим доступу: <https://www.westec.com.au/shop/p-august-wireless-dual-bluetooth-audio-transmitterreceiver.html>. – Назва з екрана.

23. Бездротовий потрійний аудіопередавач Bluetooth. [Електронний ресурс]. – Режим доступу: <https://pmdway.com/products/wireless-triple-bluetooth-audio-transmitter>. – Назва з екрана.

24. Bluetooth аудіоресивер/трансмідер Vikefon BT-B6 2 в 1. [Електронний ресурс]. – Режим доступу: <https://haamster.com.ua/ua/p2250459932-bluetooth-audio-resivertransmitter.html>. – Назва з екрана.

25. Bluetooth аудіо трансмітер innoTronics блютуз передавач. [Електронний ресурс]. – Режим доступу: <https://global-torg.com.ua/ua/p2027001618-bluetooth-audio-transmitter.html>. – Назва з екрана.
26. Bluetooth Special Interest Group. Bluetooth Core Specification Version 5.3. [Електронний ресурс]. – Режим доступу: <https://www.bluetooth.com/specifications/bluetooth-core-specification/>. – Назва з екрана.
27. Townsend K. Getting Started with Bluetooth Low Energy: Tools and Techniques for Low-Power Networking / K. Townsend, C. Cufi, R. Davidson // – Sebastopol: O'Reilly Media. – 2014. – 180 с. URL: <https://www.oreilly.com/library/view/getting-started-with/9781491900550/>.
28. Braeckel Paul. Feeling Bluetooth: From a Security Perspective / Paul Braeckel // Advances in Computers. Elsevier. – 2011. – Vol. 81, – P. 161-236, ISSN 0065-2458, ISBN 9780123855145. URL: <https://doi.org/10.1016/B978-0-12-385514-5.00004-5>.
29. Impact of Topology on Multi-hop Bluetooth Personal Area Network. / Huang Leping et al. – EUC. – 2004 – LNCS 3207, – P. 495-506.
30. Слюсарь І.І. Технологічний базис перспективних телекомунікаційних систем / І.І. Слюсарь, В.І. Слюсар, Р.О. Козуб, Р.М. Довбиш // Новітні інформаційні системи та технології. – 2017. – Т. 8. URL: <https://reposit.nupp.edu.ua/handle/PoltNTU/3097/>.
31. nRF52840. [Електронний ресурс]. – Режим доступу: <https://www.nordicsemi.com/-/media/Software-and-other-downloads/Product-Briefs/nRF52840-SoC-PB-23.pdf>. – Назва з екрана.
32. UG464: Thunderboard EFR32BG22 User's Guide. [Електронний ресурс]. – Режим доступу: <https://www.silabs.com/documents/public/user-guides/ug464-brd4184b-user-guide.pdf>. – Назва з екрана.
33. ESP32-LyraT V4.3 Посібник із початку роботи. [Електронний ресурс]. – Режим доступу: <https://docs.espressif.com/projects/esp-adf/en/latest/design-guide/dev-boards/get-started-esp32-lyrat.html>. – Назва з екрана.

34. ESP32-WROVER-E & ESP32-WROVER-IE. [Электронный ресурс]. – Режим доступа: https://www.espressif.com/sites/default/files/documentation/esp32-wrover-e_esp32-wrover-ie_datasheet_en.pdf. – Назва з екрана.
35. pschatzmann / ESP32-A2DP. [Электронный ресурс]. – Режим доступа: <https://github.com/pschatzmann/ESP32-A2DP>. – Назва з екрана.
36. Zakaria S. An Investigation of Energy Consumption in Fused Deposition Modelling using ESP32 IoT Monitoring System / S. Zakaria, P. Mativenga, E. A. R. E. Ariff // Procedia CIRP. – 2023. – Vol. 116. – P. 263–268. URL: <https://doi.org/10.1016/j.procir.2023.02.045>.
37. Tie Y. BLE Bluetooth remote-controlled car based on ESP32 / Y. Tie, P. Chen, Y. Ma // Applied and Computational Engineering. – 2024. – Vol. 81, – № 1. – P. 25–32. URL: <https://doi.org/10.54254/2755-2721/81/20240965>.
38. Simulate with Wokwi Online. [Электронный ресурс]. – Режим доступа: <https://wokwi.com/> – Назва з екрана.
39. FreeRTOS. Overview. [Электронный ресурс]. – Режим доступа: <https://docs.espressif.com/projects/esp-idf/en/v4.3/esp32/api-reference/system/freertos.html>. – Назва з екрана.

ДОДАТКИ

Додаток А

Діаграма моделі в форматі JSON

```

{
  "version": 1,
  "author": "",
  "editor": "wokwi",
  "parts": [
    { "type": "board-esp32-devkit-c-v4", "id": "esp", "top": 9.6, "left": -
187.16, "attrs": {} },
    {
      "type": "wokwi-pushbutton",
      "id": "btn2",
      "top": 172.7,
      "left": -399.7,
      "rotate": 90,
      "attrs": { "color": "green" }
    },
    {
      "type": "wokwi-buzzer",
      "id": "bz1",
      "top": 33.9,
      "left": -357.9,
      "rotate": 270,
      "attrs": { "volume": "0.1" }
    },
    {
      "type": "wokwi-pushbutton",
      "id": "btn3",
      "top": 172.7,
      "left": -313.3,
      "rotate": 90,
      "attrs": { "color": "blue" }
    },
    {
      "type": "wokwi-text",
      "id": "led1",
      "top": 240,
      "left": -393.6,
      "attrs": { "text": "Аудіо 1" }
    },
    {
      "type": "wokwi-text",
      "id": "led2",
      "top": 240,
      "left": -307.2,

```

```

    "attrs": { "text": "Аудіо 2" }
  }
],
"connections": [
  [ "esp:TX", "$serialMonitor:RX", "", [ ] ],
  [ "esp:RX", "$serialMonitor:TX", "", [ ] ],
  [ "bz1:1", "esp:CMD", "black", [ "h67.2", "v115.2" ] ],
  [ "btn3:2.r", "btn2:2.r", "black", [ "v9.8", "h-48.2" ] ],
  [ "esp:25", "bz1:2", "red", [ "h-9.45", "v-29.2" ] ],
  [ "esp:27", "btn3:1.1", "blue", [ "h0" ] ],
  [ "esp:26", "btn2:1.1", "green", [ "h0" ] ],
  [ "btn3:2.r", "esp:CMD", "black", [ "v9.8", "h105.25" ] ]
],
"dependencies": {}
}

```

Програмна частина

```

#include "pitches.h"

// Піни для п'єзoeлемента та кнопок
#define BUZZER_PIN 25
#define audio1 26
#define audio2 27

bool audio1Running = false;
bool audio2Running = false;

int melody[] = {
    NOTE_E5, 0, NOTE_E5, 0, NOTE_E5, NOTE_E5, 0, 0, 0,
    NOTE_E5, 0, NOTE_E5, 0, NOTE_E5, NOTE_E5, 0, 0, 0,
    NOTE_E5, 0, NOTE_G5, 0, NOTE_C5, 0, NOTE_D5, 0,
    NOTE_E5, NOTE_E5, NOTE_E5, NOTE_E5, 0, 0, 0, 0, 0,
    NOTE_F5, 0, NOTE_F5, 0, NOTE_F5, 0, NOTE_F5, 0,
    NOTE_F5, 0, NOTE_E5, 0, NOTE_E5, 0, NOTE_E5, 0, NOTE_E5, 0,
    NOTE_E5, 0, NOTE_D5, 0, NOTE_D5, 0, NOTE_E5, 0,
    NOTE_D5, NOTE_D5, 0, 0, 0, NOTE_G5
};

// Черги
QueueHandle_t audioQueue1, audioQueue2;

//Ідентифікатори завдань
TaskHandle_t Audio1Handle = NULL;
TaskHandle_t Audio2Handle = NULL;

void setup()
{
    // Ініціалізація зв'язку через серійний порт
    Serial.begin(115200);
    Serial.println("Hello, ESP32!");

    // Налаштування пінів для кнопок
    pinMode(audio1, INPUT_PULLUP);
    pinMode(audio2, INPUT_PULLUP);

    // Ініціалізація черг
    audioQueue1 = xQueueCreate(1000, sizeof(int));
    audioQueue2 = xQueueCreate(1000, sizeof(int));

    // Створення завдань
    xTaskCreatePinnedToCore(buttonsTask, "buttonsTask", 10000, NULL, 1, NULL,
0);

```

```

        xTaskCreatePinnedToCore(audio1Task,    "Audio1",    10000,    NULL,    1,
&Audio1Handle, 0);
        vTaskSuspend(Audio1Handle);
        xTaskCreatePinnedToCore(audio2Task,    "Audio2",    10000,    NULL,    1,
&Audio2Handle, 0);
        vTaskSuspend(Audio2Handle);
        xTaskCreatePinnedToCore(playbackTask, "Playback", 10000, NULL, 1, NULL, 1);
    }

    // Генерація частот 1
    void audio1Task(void *parameter)
    {
        int frequency = 100;
        while (true) {
            if (frequency == 100)
                frequency = 200;
            else frequency = 100;

            // Відправка частоти в чергу
            xQueueSend(audioQueue1, &frequency, portMAX_DELAY);
        }
    }

    // Генерація частот 2
    void audio2Task(void *parameter)
    {
        int noteIndex = 0;
        int frequency;

        while (true) {
            frequency = melody[noteIndex];
            if (noteIndex > 64)
                noteIndex = 0;
            // Відправка частоти в чергу
            xQueueSend(audioQueue2, &frequency, portMAX_DELAY);
            noteIndex = (noteIndex + 1); // Перехід до наступної ноти
        }
    }

    // Мікшування та відтворення
    void playbackTask(void *parameter) {
        int freq1 = 0, freq2 = 0;

        while (true) {
            bool received1 = xQueueReceive(audioQueue1, &freq1, pdMS_TO_TICKS(10));
            bool received2 = xQueueReceive(audioQueue2, &freq2, pdMS_TO_TICKS(10));

            if (received1 && received2)
            {

```

```

    // Якщо є дані від двох джерел
    Serial.print("Частота 1:");
    Serial.print(freq1);
    Serial.print(" Частота 2:");
    Serial.print(freq2);

    int mixedFreq = (freq1 + freq2) / 2;
    Serial.print(" Середня:");
    Serial.println(mixedFreq);
    tone(BUZZER_PIN, mixedFreq, 125);
    vTaskDelay(pdMS_TO_TICKS(125));
}
else if (received1)
{
    // Якщо є дані тільки від першого джерела
    Serial.print("Частота 1:");
    Serial.println(freq1);
    tone(BUZZER_PIN, freq1, 125);
    vTaskDelay(pdMS_TO_TICKS(125));
}
else if (received2)
{
    // Якщо є дані тільки від другого джерела
    Serial.print("Частота 2:");
    Serial.println(freq2);
    tone(BUZZER_PIN, freq2, 125);
    vTaskDelay(pdMS_TO_TICKS(125));
}
}
}

// Функція для обробки натискання кнопок
void buttonsTask(void *pvParameters)
{
    while (true)
    {
        //Обробка натискання кнопки 1
        static bool lastButton1State = HIGH;
        bool currentButton1State = digitalRead(audio1);
        if (lastButton1State == HIGH && currentButton1State == LOW)
        {
            audio1Running = !audio1Running;
            if (audio1Running)
            {
                vTaskResume(Audio1Handle); // Відновлення задачі
            }
            else
            {
                vTaskSuspend(Audio1Handle); // Призупинення задачі
            }
        }
    }
}

```

```

        xQueueReset(audioQueue1);
    }
    vTaskDelay(pdMS_TO_TICKS(300));
}
lastButton1State = currentButton1State;
//Обробка натискання кнопки 2
static bool lastButton2State = HIGH;
bool currentButton2State = digitalRead(audio2);
if (lastButton2State == HIGH && currentButton2State == LOW)
{
    audio2Running = !audio2Running;
    if (audio2Running)
    {
        vTaskResume(Audio2Handle); // Відновлення задачі
    }
    else
    {
        vTaskSuspend(Audio2Handle); // Призупинення задачі
        xQueueReset(audioQueue2);
    }
    vTaskDelay(pdMS_TO_TICKS(300));
}
lastButton2State = currentButton2State;

// Затримка для ефективної роботи
vTaskDelay(pdMS_TO_TICKS(10));
}
}

// Основна функція loop не використовується
void loop() {}

```