

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Факультет інформаційних технологій

Кафедра телекомунікацій, медійних та інтелектуальних технологій

## ДИПЛОМНА РОБОТА

другого (магістерського) рівня

Освітній рівень

Галузь знань 17 Електроніка та телекомунікації

Шифр і назва спеціальності

Спеціальність 172 Телекомунікації та радіотехніка

Шифр і назва спеціальності

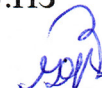
Освітня програма Телекомунікації та радіотехніка

Назва освітньої програми

на тему Метод підвищення пропускної здатності безпроводних локальних мереж стандарту IEEE 802.11

ДРТР.022189.01.03.ПЗ

Виконав: здобувач 2 курсу, група ТР<sub>М</sub>-22-1

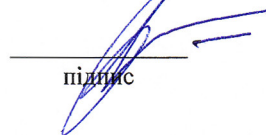


підпис

Д.В. ПАНАСЮК

Ініціали, прізвище

Керівник: д-р техн. наук, проф.



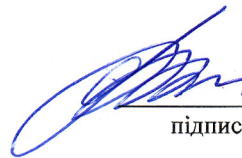
підпис

Ю.М. БОЙКО

Ініціали, прізвище

До захисту допускаю:

Зав. кафедри: д-р техн. наук, проф.



підпис

С.К. ПІДЧЕНКО

Ініціали, прізвище

5 12 2023 р.

Хмельницький, 2023

Хмельницький національний університет

Факультет Інформаційних технологій

Кафедра Телекомунікації, медійних та інтелектуальних технологій

Освітній рівень Другий (магістерський)

Галузь знань 17 – Електроніка та телекомунікації

Спеціальність 172 – Телекомунікації та радіотехніка

Освітня програма Телекомунікації та радіотехніка

ЗАТВЕРДЖУЮ

Зав. кафедрою

« 1 » 09 2023 р.

**ЗАВДАННЯ  
НА ДИПЛОМНУ РОБОТУ**

ПАНАСЮКУ Дмитру Володимировичу

1 Тема роботи: Метод підвищення пропускної здатності безпроводних локальних мереж стандарту IEEE 802.11

керівник роботи БОЙКО Юлій Миколайович, д.т.н, професор.

Затверджено наказом по університету від «15» серпня 2023р. № 30.

2 Термін подання здобувачем роботи на кафедру: 05.12.2023р.

3 Вихідні дані роботи:

Метою дипломної роботи є дослідження методів підвищення пропускної здатності безпроводних локальних мереж стандарту IEEE 802.11, моделювання системи обробки сигналів у процесорі основної смуги частот передавача і приймача.

Предмет дослідження – методи та засоби підвищення пропускної здатності безпроводного зв'язку стандарту IEEE 802.11.

Об'єкт дослідження – процес передачі сигналів у безпроводних локальних мереж стандарту IEEE 802.11.

4. Зміст пояснювальної записки (перелік питань, які потрібно розробити):

1 Основні відомості про стандарт бездротової мережі IEEE 802.11ах. 2

Побудова кодеру і декодеру LDPC. 3 Декодер QC-LDPC з малою затримкою.

4 Дослідження пропускної здатності безпроводних локальних мереж.

Висновки.


5. Перелік графічного матеріалу: презентація обсягом 11 слайдів

6. Дата видачі завдання: 15.08.2023

### КАЛЕНДАРНИЙ ПЛАН

№ п/п	Назва етапу (розділу) дипломної роботи	Термін виконання етапу дипломної роботи	Примітка
1	Вступ. Основні відомості про стандарт бездротової мережі IEEE 802.11ax	12.09.2023	Вик.
2	Побудова кодера і декодера LDPC	10.10.2023	Вик.
3	Декодер QC-LDPC з малою затримкою	31.10.2023	Вик.
4	Дослідження пропускну здатності безпроводних локальних мереж	17.11.2023	Вик.
5	Висновки. Презентаційні матеріали за результатами виконання дипломної роботи.	03.12.2021	Вик.

Здобувач



Підпис

Д.В. ПАНАСЮК  
Ініціали, прізвище

Керівник роботи



Підпис

Ю.М. БОЙКО  
Ініціали, прізвище

## АНОТАЦІЯ

Тема дипломної роботи: Метод підвищення пропускної здатності безпроводних локальних мереж стандарту IEEE 802.11

Автор роботи: ПАНАСЮК Дмитро Володимирович

Керівник роботи: БОЙКО Юлій Миколайович

Пояснювальна записка : 98 сторінок, 43 рисунків, 36 джерело, 2 додатки

Графічна частина: 11 презентаційних слайдів

КЛЮЧОВІ СЛОВА: Wi-Fi, LDPC, FPGA, BER, SNR, HDL, КОДУВАННЯ

Метою дипломної роботи є дослідження методів підвищення пропускної здатності безпроводних локальних мереж стандарту IEEE 802.11, моделювання системи обробки сигналів у процесорі основної смуги частот передавача і приймача.

Об'єкт дослідження : процес передачі сигналів у безпроводних локальних мереж стандарту IEEE 802.11.

Предмет дослідження : методи та засоби підвищення пропускної здатності безпроводного зв'язку стандарту IEEE 802.11.

В першому розділі дипломної роботи здійснено огляд принципів побудови безпроводних локальних мереж за стандартом IEEE 802.11. У другому розділі роботи представлені математичні моделі та способи побудови кодеру і декодеру LDPC. В третьому розділі дипломної роботи розглянуті квазіциклічні коди LDPC, їх апаратна реалізація у FPGA. Четвертий розділ кваліфікаційної роботи присвячений дослідженню пропускної здатності безпроводних локальних мереж, дослідженні Simulink моделі системи зв'язку, визначенні коефіцієнту бітових помилок при зміні позиційності цифрової модуляції, швидкості коду, розміру фрейму.

## ЗМІСТ

Вступ.....	6
1 Основні відомості про стандарт бездротової мережі IEEE 802.11ax.....	9
1.1 Загальна будова систем зв'язку.....	9
1.2 Використання стандарту Wi-Fi 6.....	12
1.3 Огляд особливостей WI-FI: IEEE 802.11ax.....	14
1.4 Очікувані проблеми, пов'язані з використанням Wi-Fi.....	20
1.5 Порівняння сигналів Wi-Fi 5 і Wi-Fi 6.....	23
Висновки до першого розділу.....	26
2 Побудова кодеру і декодеру LDPC.....	27
2.1 Загальні відомості про коди LDPC.....	27
2.2 Будова кодів LDPC.....	29
2.3 Декодер кодів LDPC.....	31
2.4 Пряме кодування QC-LDPC.....	38
2.5 Універсальний декодер LDPC.....	42
2.6 Аналіз пропускної спроможності.....	47
Висновки до другого розділу.....	49
3 Декодер QC-LDPC з малою затримкою.....	50
3.1 Квазіциклічні коди LDPC.....	50
3.2 Архітектура декодера квазіциклічних кодів.....	55
3.3 Залежність даних багаторівневого декодера.....	57
3.4 Алгоритм декодування з низькою затримкою.....	60
3.5 Архітектура декодера з мінімальною затримкою.....	63
3.6 LDPC декодер низької складності.....	67
Висновки до третього розділу.....	71
4 Дослідження пропускної здатності безпроводних локальних мереж.....	72
4.1 Приймач цифрової системи зв'язку.....	72
4.2 Коды LDPC і алгоритм мінімальної суми.....	73

4.3 Кодування і декодування кодів LDPC.....	74
4.4 Алгоритм нормалізованої мінімальної суми.....	75
4.5 Результати експериментальних досліджень.....	86
4.6 Апаратне забезпечення декодера LDPC.....	90
Висновки до четвертого розділу.....	92
Висновки.....	93
Перелік джерел посилання.....	94
Додаток А. Презентація.....	99
Додаток Б. Апробація роботи.....	110

## ВСТУП

### **Актуальність теми.**

Новий стандарт бездротових локальних мереж Wi-Fi 6 (IEEE 802.11ax) базується на сучасному стандарті 802.11ac (Wi-Fi 5ac) та використовує вже існуючі технології. Деякі нові технології будуть корисні при розгортанні Wi-Fi-мереж із високою щільністю. Окремі рішення покращать якість зв'язку в місцях з високим навантаженням на мережу та в умовах високого заповнення радіоефіру (наприклад, у громадському транспорті, торгових центрах, готелях, на стадіонах чи корпоративних мережах).

На даний час, оптимальним варіантом для підключення до інтернету та організації власної мережі Wi-Fi є використання дводіапазонного Wi-Fi-роутера 802.11ac та мобільних пристроїв з підтримкою стандарту Wi-Fi 5.

У Wi-Fi 6 теоретична швидкість передачі даних становитиме 600 Мбіт/с (80 МГц, 1 просторовий потік) і 9607 Мбіт/с (160 МГц, 8 просторових потоків), у порівнянні з 433 Мбіт/с (80 МГц, 1 просторовий потік) та 6933 Мбіт/с (160 МГц, 8 просторових потоків) для стандарту Wi-Fi 5.

У Wi-Fi 6 доданий режим OFDMA (Orthogonal Frequency Division Multiple Access, множинний доступ з ортогональним частотним поділом каналів) для поліпшення спектральної ефективності. Технологія OFDMA була запозичена з стільникової індустрії 4G LTE і схожа на розраховану на багато користувачів версію OFDM, яка використовується в Wi-Fi 5.

OFDMA забезпечує можливість встановлення з'єднань між точкою доступу і кількома клієнтами одночасно за рахунок розподілу сигналу на частоти підносійних (додаткові дрібніші підканали) і виділяти їх в групи для обробки окремих потоків даних, званих ресурсними одиницями (Resource Units, RU). Вона дозволить одночасно транслювати дані відразу кільком клієнтам Wi-Fi 6 із усередненою швидкістю і використовувати один і той же канал не використовуючи очікування.

Wi-Fi 6 забезпечує для технології MU-MIMO (Multi-User Multiple-Input, Multiple-Output; багатокористувацький багатоканальний вхід - вихід) підтримку висхідного і низхідного напрямку (UL і DL MU-MIMO).

Передбачається, що Wi-Fi 6 теоретично може підтримувати одночасну передачу до 8 просторових потоків в обох напрямках (8x8) з можливістю доставки чотирьох одночасних потоків одному клієнту.

У 1024-QAM збільшена довжина кожного символу кодування з 8 біт (256-QAM на стандарті Wi-Fi 5) до 10 біт, що підвищує швидкість передачі даних і ефективність використання спектра приблизно на 25%, оскільки у кожний пакет буде розміщено більше даних.

Коди перевірки на парність з низькою щільністю (LDPC) були прийняті в різних стандартах бездротового зв'язку, таких як WiMAX і WiFi, в основному завдяки їх чудовим характеристикам виправлення помилок і властивому паралелізму. Код LDPC є кодом з лінійною корекцією помилок, який визначається розрідженою двійковою матрицею перевірки на парність, яка у графічному вигляді представляється дводольним графом Таннера.

#### **Мета і задачі дослідження.**

Метою роботи є дослідження методів підвищення пропускної здатності безпроводних локальних мереж стандарту IEEE 802.11, моделювання системи обробки сигналів у процесорі основної смуги частот передавача і приймача. Для досягнення цієї мети поставлені наступні завдання:

- провести огляд шляхів підвищення пропускної здатності безпроводних локальних мереж;
- дослідити модель системи безпроводного зв'язку;
- дослідити коефіцієнт бітових помилок від відношення сигнал-шум для системи безпроводного зв'язку;

**Об'єкт дослідження** – процес передачі сигналів у безпроводних локальних мереж стандарту IEEE 802.11.

**Предмет дослідження** – методи та засоби підвищення пропускну здатності безпроводного зв'язку стандарту IEEE 802.11.

**Методи досліджень.** Для вирішення поставлених наукових завдань використовується математичний апарат теорії цифрової обробки сигналів, передачі та приймання сигналів.

**Наукова новизна одержаних результатів:**

1. Набули подальшого розвитку методи підвищення пропускну здатності безпроводних локальних мереж.
2. Набули подальшого розвитку методи декодування LDPC кодів з різною кодовою швидкістю.

**Практичне значення одержаних результатів:**

Проведене дослідження завадостійкості кодів LDPC різної кодової швидкості, сигнально-кодових конструкцій з різною швидкістю передавання інформації. Досліджені залежності коефіцієнту бітових помилок від відношення сигнал-шум для системи зв'язку з цифровою квадратурною модуляцією і кодуванням LDPC, побудований декодер LDPC за алгоритмом нормалізованої мінімальної суми з використанням HDL.

**Публікації.** Результати дипломної роботи магістра опубліковані в одній статті у науковому журналі «Вимірювальна та обчислювальна техніка в технологічних процесах».

**Структура та обсяг магістерської атестаційної роботи.** Дипломна робота магістра складається із вступу, чотирьох розділів, висновків, переліку джерел посилання та додатків. Дипломна робота магістра має загальний обсяг 98 сторінок, з яких основний зміст викладений на 89 сторінках друкованого тексту, містить 43 рисунків. Перелік джерел посилання складається з 36 джерел.

# 1 ОСНОВНІ ВІДОМОСТІ ПРО СТАНДАРТ БЕЗДРОТОВОЇ МЕРЕЖІ IEEE 802.11ax

## 1.1 Загальна будова систем зв'язку

Бездротові мережі стали свідками постійної та зростаючої популярності, що приваблює все більше користувачів. Це призвело до значного збільшення споживання даних у всіх мережах.

Багато трафіку генерується і споживається всередині приміщень, тому рішення для підключення всередині приміщень можуть відігравати важливу роль у задоволенні високих вимог до пропускної спроможності.

Мережі WLAN на базі IEEE 802.11, які є найбільш популярними та успішними бездротовими рішеннями всередині приміщень, перетворилися на ключову технологію, що дозволяє охопити середні та великі підприємства, точки доступу в громадських місцях, житлові комплекси, тощо. Такі середовища характеризуються тим, що включають декілька невеликих сот з багатьма точками доступу (ТД) і обслуговують велику кількість клієнтів при наявності збільшення покриття і високої швидкості передачі даних.

В останні роки відбувся значний сплеск розгортання WLAN у географічно обмежених середовищах (включаючи декілька базових наборів послуг, що перекриваються – OBSS). Стратегічна важливість технології Wi-Fi (з точки зору очікуваної кількості пристроїв з підтримкою Wi-Fi) для задоволення потреб у трафіку обговорюється операторами зв'язку. Наступні факти про розмір світового ринку Wi-Fi очевидні: глобальна вартість ринку Wi-Fi, за прогнозами, збільшиться до 33,6 млрд. доларів США.

Комітет стандартизації IEEE 802.11 активно продовжує випускати нові проекти поправок, що включають останні технологічні досягнення. У порівнянні зі стільниковими технологіями стандарти IEEE 802.11 створюються з метою забезпечення зворотної сумісності і, таким чином, доповнюють один одного, впроваджуючи ключові технічні аспекти.

Нещодавно робоча група IEEE 802.11 схвалила розробку нового стандарту WLAN – IEEE 802.11ax. Робоча група 802.11ax (TGax) зараз працює над розробкою розширення стандарту IEEE 802.11ac з метою підвищення пропускної спроможності системи (а не тільки підтримуваних швидкостей передачі даних на рівні каналу). Більш конкретно, він призначений для підвищення ефективності сценаріїв з обмеженими завадами (через високу щільність пристроїв IEEE 802.11).

Розглянемо нову поправку до високоефективного Wi-Fi (тобто IEEE 802.11ax), необхідність її внесення. Розглянемо варіанти використання та надамо огляд ключових технологічних особливостей, що пропонуються для поправки до IEEE 802.11ax. Наголосимо на двох основних проблемах, з якими зіткнеться наступне покоління мереж Wi-Fi: а) співіснування з неліцензійним LTE та б) прийняття парадигми Інтернету речей.

Майбутній стандарт IEEE 802.11ax спрямований на підвищення ефективності використання спектра та пропускної спроможності у реальних середовищах Wi-Fi з великою кількістю користувачів. Оскільки доступна кількість ортогональних каналів для IEEE 802.11 обмежена, базові набори послуг, що перекриваються (OBSS) у мережах на базі IEEE 802.11 є частою проблемою. Механізм запобігання колізій має тенденцію знижувати пропускну спроможність мережі та збільшувати затримки передачі, де, незважаючи на прийнятну ймовірність колізій, середовище ніколи не використовується повністю. Рис. 1.1 вказує на проблему OBSS.

Моделюється сценарій трьох комірок поєднаного каналу, що перекриваються, кожна з яких включає одну точку доступу і п'ять зв'язаних станцій. Зі зменшенням площі перекриття середня пропускну здатність усередині кожної комірки збільшується (наприклад, збільшення пропускної здатності більш ніж на 400% видно, коли площа перекриття між сусідніми комірками зменшується з 38% до 7%). Зменшення областей, що перекриваються, може бути досягнуто шляхом застосування різних методів, представлених IEEE 802.11ax.

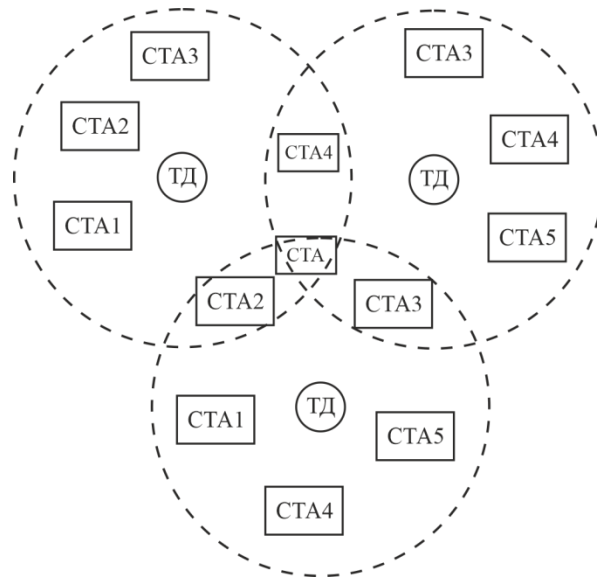


Рисунок 1.1 – Сценарій високої щільності, коли пристрої Wi-Fi співіснують із проблемою перекриття базового набору послуг (BSS)

Хоча поточні стандарти IEEE 802.11 (тобто IEEE 802.11n/ac) були розроблені з наміром покращити пікову сукупну пропускну спроможність мережі з кількома станціями, належне зниження завад, що виникають, ще не було вирішено. Більш того, метод доступу до каналу у вищезгаданих стандартах є надмірно захищеним, що призводить до зменшення повторного використання простору. Зокрема, майбутній стандарт IEEE 802.11ax призначений для використання методів, які дозволять збільшити фізичну швидкість передачі даних, а також знизити частоту помилок кадрів (FER) та покращити повторне використання спектра за рахунок забезпечення високоефективного розрахованого на багато користувачів доступу та зменшення завад, що, своєю чергою, збільшить пропускну спроможність зони покриття.

Стандарт IEEE 802.11ax в першу чергу розробляється для забезпечення високоефективної роботи WLAN як усередині, так і поза приміщеннями, де першочергове значення надається підвищенню надійності як при передачі поза приміщенням, так і висхідною лінією зв'язку. Цей стандарт спрямований на покращення показників продуктивності (середня пропускну

спроможність на станцію), що безпосередньо призводить до підвищення ефективності порівняно з декількома розташованими розгортаннями базового набору послуг (BSS). Таким чином, мета запропонованої поправки полягає у визначенні стандартизованих модифікацій фізичного рівня PHY та MAC успадкованого стандарту IEEE 802.11 для покращення якості обслуговування кінцевих користувачів у щільно розгорнутих середовищах WLAN. Сферу запропонованої поправки можна додатково уточнити так: Поліпшено пропускну здатність системи та користувачів у щільних розгортаннях.

Очікується, що ця поправка збільшить пропускну спроможність району як мінімум у чотири рази, одночасно прагнучи збільшити середню пропускну спроможність на станцію до десяти відсотків.

Крім збільшення пропускну спроможності кінцевих користувачів, вищезгаданий стандарт також спрямований на підтримку та підвищення енергоефективності за рахунок включення спрощених режимів енергозбереження. Очікується, що стандарт надасть методи, які забезпечать ефективне використання спектральних ресурсів.

IEEE 802.11ax в основному орієнтований на роботу WLAN на частотах 2,4 та 5 ГГц, але підтримує режим роботи від 1 до 6 ГГц. Увімкнення зворотної сумісності. Очікується також, що він матиме зворотну сумісність для підтримки зв'язку з будь-яким застарілим пристроєм IEEE 802.11.

## **1.2 Використання стандарту Wi-Fi 6**

Метою стандарту IEEE 802.11ax є надання можливостей самоконфігурації та самоадаптації для збільшення пропускну спроможності зони покриття. Тому TGax визначила пріоритетність таких варіантів використання для розробки та оцінки різних функцій.

У житловому середовищі OBSS високої щільності створюється, коли велику кількість точок доступу WLAN встановлено у безпосередній близькості, наприклад, багатоквартирному будинку. У такому разі підвищений рівень завад (через некероване та незаплановане розгортання) може суттєво вплинути на продуктивність пристроїв у мережі.

Подібно до житлових приміщень, підприємства/організації надають Wi-Fi як основне джерело доступу до Інтернету через керовану мережу. Питання управління завадами та політики «Принеси свій власний пристрій» (BYOD) мають першорядне значення у цих середовищах, та сценаріях, коли різні корпоративні мережі знаходяться в безпосередній близькості.

Невеликі внутрішні точки доступу BSS є сценарієм з високою щільністю точок доступу і станцій, де BSS від кожного оператора розгорнуто з регулярною симетрією (наприклад, торгові центри, аеропорти, залізниця тощо). Різні комірки різних операторів можуть перекриватися та викликати завади, які можуть погіршити продуктивність мережі на площі покриття.

Великі зовнішні точки доступу BSS мають метою – змодельовати зовнішнє розгортання (аналогічно мобільним мережам), що складається з високої щільності станцій, а також максимального поділу між різними точками доступу. У такому середовищі потенційні завади від різних станцій, які не є точками доступу, можуть серйозно вплинути на роботу кінцевого користувача та знизити загальну продуктивність.

IEEE 802.11ax покликаний зменшити вплив змінних завад сусідніх транспортних засобів, а також вивчити можливі методи зменшення обмежень для транспортного засобу на зв'язок з інфраструктурою (тобто міркування мобільності та спрямованості сигналу).

Кампуси, заводські середовища (де кілька сотень точок доступу можуть бути зосереджені на невеликій території), невеликі офіси (один BSS з обмеженою кількістю пристроїв, що зазнають некерованих завад) та варіанти використання Інтернету речей також розглядаються TGax як можливі варіанти використання середовища. IEEE 802.11ax пропонує

використовувати LDPC при використанні великої смуги пропускання (тобто при об'єднанні каналів) та використовувати BCC при більш вузькій смузі пропускання.

### **1.3 Огляд особливостей WI-FI: IEEE 802.11ax**

Щоб покращити взаємодію з користувачем за рахунок зменшення завад, а також забезпечити покращену сукупну пропускну спроможність кількох станцій, розробляється поправка до IEEE 802.11ax, метою якої є значне підвищення ефективності WLAN поряд з продуктивністю на рівні системи щільного розгортання. TGax має намір впровадити радіотехнологію, засновану на множинному вході та множині виходів (MIMO) і OFDMA, щоб можна було передавати більше бітів за одну можливість передачі (TXOP).

Розглянемо важливі функції, що пропонуються для виправлення IEEE 802.11ax. Розглянемо різні пропозиції до наступних чотирьох категорій; PHY, MAC, розрахований на багато користувачів режим та інші примітні функції. Виділимо очікувані поліпшення (з точки зору пропускну спроможності системи) у чотирьох вищезгаданих категоріях (де розраховані на багато користувачів методи показують найбільший вигравш). Доречно відзначити, що очікуване відсоткове поліпшення кожної пропозиції визначається дослідженнями, представленими та обговорюваними на TGax.

Хоча IEEE 802.11ax є розвитком стандарту IEEE 802.11ac, він спрямований на впровадження нових технологій, зберігаючи при цьому зворотну сумісність. Наприклад, IEEE 802.11ax PPDU має намір включити застарілу преамбулу, дубльовану на кожному підканалі 20 МГц, щоб вирішити проблему зворотної сумісності та співіснування. Крім того, TGax також розглядає можливість розробки нових типів преамбул, необхідних для підтримки нових функцій. Примітні поправки, запропоновані фізично для IEEE 802.11ax, пояснюються наступним чином.

1) Рішення про фізичне кодування (LDPC та BCC):

Схема прямого виправлення помилок (FEC) за умовчанням, запропонована для IEEE 802.11n та IEEE 802.11ac, заснована на двійковому згортковому кодуванні (BCC) з частотним перемещуванням на кожен символ мультиплексування з ортогональним частотним поділом каналів (OFDM). Використання перевірки парності низької щільності (LDPC) не є обов'язковим і поки не користується великою популярністю у WLAN через високі обчислювальні витрати. Однак було показано, що коди LDPC забезпечують значний вигравш (пропускної спроможності) порівняно з BCC.

2) 1024 QAM. Одним із рішень, запропонованих TGах для досягнення чотириразового збільшення середньої пропускної здатності, є включення дуже високої додаткової схеми модуляції (тобто 1024-QAM), де кожен символ кодує більшу кількість бітів даних при використанні сузір'я.

3) Поліпшення для зовнішнього зв'язку: Щоб покращити спектральну ефективність станцій у передбачуваних випадках використання, TGах має намір використовувати розмір FFT у чотири рази більший, ніж той, що використовується для IEEE 802.11ac. Передбачається, що більший розмір ШПФ підвищить надійність роботи на відкритому повітрі, а також покращить середню пропускну здатність усередині приміщення.

Щоб зменшити великі втрати на тракті та затримку каналу, що виникають у великих точках доступу на відкритому повітрі, TGах визначає новий формат блоку даних протоколу (HE-PPDU) конвергенції фізичного рівня (PLCP), званий PPDU з розширеним діапазоном для одного користувача (SU), в якому поля, що містять інформацію, необхідну для інтерпретації пакетів, повторюються.

4) Частотно-вибіркове планування: OFDMA-системи виграють від частотної вибірковості з погляду частотного рознесення та частотно-вибіркового планування (FSS). У TGах активно використовується FSS для забезпечення збільшення пропускної спроможності віддалених станцій (по

відношенню до ТД) шляхом виділення блоків фізичних ресурсів з найменшою кількістю завмирань для їх передачі.

Крім того, IEEE 802.11ax має намір адаптувати модуляцію з двома підносійними (схему, яка модулює ту саму інформацію на парі віддалених один від одного підносійних) для поліпшення продуктивності FER і стійкості до вузькосмугових завад при щільних розгортанні.

TGax працює над наступними покращеннями MAC:

1) Поліпшення просторового повторного використання (модифікації RYSSA). Застарілий стандарт IEEE 802.11 використовує модулі оцінки фізичної чистоти каналу (RYSSA) визначення стану каналу (тобто зайнятий чи вільний) шляхом вимірювання отриманої енергії. Запропонована поправка IEEE 802.11ax спрямована на формальне ухвалення динамічних модифікацій RYSSA. Ці методи дозволяють співіснувати кільком одночасним передачам і, таким чином, збільшують повторне використання спектра. Інтуїція включення цих модифікацій полягає в тому, що при щільному розгортанні станції можуть в кінцевому підсумку завжди припускати, що канал зайнятий (через фіксований діапазон вимірювання несучої), навіть незважаючи на те, що множинні одночасні передачі все ще можуть бути можливі. TGax брав активну участь у розробці схем модифікації RYSSA, де алгоритм динамічного контролю чутливості (DSC) був запропонований як одна з ключових інноваційних технологій, які можуть збільшити загальну пропускну здатність. Основна ідея схеми DSC полягає в оптимізації існуючих розгортань шляхом відповідного настроювання порога контролю несучої (CST) для кожного вузла розподіленим чином. DSC намагається обмежити збільшення та зменшення CST для станції обмеженою областю, щоб уникнути як надзвичайно агресивної, так і консервативної поведінки. Приріст пропускну спроможності, що досягається за допомогою DSC, становить у середньому більше 20% [4] у поєднанні з оптимальним вибором каналу (приріст збільшується на 40%, коли станції використовують повільні швидкості передачі даних та відправляють довгі кадри).

2) Для управління потужністю передачі є можливість стандартизації механізмів управління потужністю передачі (TPC) кожного каналу з метою зменшення завад, збільшення просторового повторного використання. Метою механізму управління потужністю є динамічне регулювання мінімально можливої потужності для станцій з найбільшими втратами на трасі з метою досягнення цільового відношення сигнал/завади плюс шум (SINR) (для правильного декодування отриманих кадрів).

3) Поліпшення просторового повторного використання (BSS кольори). Це інноваційна схема збільшення пропускної спроможності щільних мереж WLAN, де кожному BSS призначається певний колір (у термінах бітів, що позначають поля L-SIG фізичного заголовка). Станція, отримавши кадри від сусідньої BSS, може відмовитися від процесу прийому, припускаючи, що канал вільний під час цієї передачі, тим самим збільшуючи можливості передачі. Ця схема була запропонована для стандарту IEEE 802.11ah, і продемонструвала поліпшення при використанні у мережах IEEE 802.11ah.

4) Поліпшення просторового повторного використання (кілька NAV для просторового повторного використання). У застарілому стандарті IEEE 802.11 віртуальне визначення несучої використовується для вирішення проблеми колізій, пов'язаної із прихованими вузлами. Цей метод працює шляхом резервування бездротового каналу за допомогою підтвердження RTS/CTS (що передують кадрам даних). Сусідні станції прослуховування після отримання кадрів RTS/CTS встановлюють таймер (вектор мережевого розподілу – NAV), який блокує їх передачу протягом певного часу.

Виправлення до IEEE 802.11ah пропонує використовувати два таймери NAV на кожній станції (один ідентифікується як Intra-BSS NAV, а другий називається NAV), де Intra-BSS NAV скидається або збільшується тільки на кадри з цього BSS. Таким чином, просторове повторне використання можна збільшити, якщо станції ігноруватимуть RTS/CTS, що передаються OBSS.

5) Управління завадами. Оскільки традиційні методи управління завадами при застосуванні до щільних розгортань також полегшують

загальні умови мережі, IEEE 802.11ax прагне розумно використовувати метод RTS/CTS на основі умов каналу для кожного вузла (тобто точка доступу може використовувати нові механізми віддаленого включення RTS/CTS для будь-якої зі зв'язаних з ним станцій). Якщо передача утруднена через підозру існування прихованих вузлів (наприклад, через використання механізмів адаптації контролю несучої, таких як DSC), станції можуть вибрати використання вищезазначеного методу. Нехай відбувається передача по висхідній лінії зв'язку в умовах насичення, коли кожна станція безперервно передає фрейми максимальної тривалості (сценарій найгіршого випадку). Є значний виграш при поєднанні DSC (тобто приріст пропускної спроможності близько 40%), а також DSC з інтелектуальним механізмом чотиристороннього квітування (тобто приріст пропускної спроможності близько 60%) у густонаселеному житловому середовищі.

Огляд розрахованих на багато користувачів методів, запропонованих TGax, виглядає наступним чином.

1) OFDMA низхідної та висхідної лінії зв'язку: OFDMA працює поверх OFDM, при цьому базова станція виділяє підмножину несучих кожному користувачу, щоб забезпечити можливість кількох одночасних передач. OFDMA використовує синхронний доступ до середовища, що призводить до зменшення конфліктів (тобто зменшення колізій). Таким чином, робоча група IEEE 802.11ax визначила OFDMA висхідної та низхідної лінії зв'язку (де мінімальний розмір ресурсної одиниці (RU) складається з 26 підносійних) як ключову багато-користувацьку функцію для підвищення ефективності фізичного рівня. Різним станціям у щільному середовищі, які неефективно виборюють загальні ресурси, виділяються виділені подканали, які збільшують середню пропускну спроможність кінцевого користувача.

Щоб дозволити роботу OFDMA, IEEE 802.11ax пропонує використовувати спеціальний формат HE-PPDU, який називається PPDU на основі тригера HE, який дозволяє оголошувати рішення щодо планування. Ця функція допомагає знизити складність синхронізації. Механізм виділення

каналів (що складається з методів виділення доступних RU в низхідній та висхідній лінії зв'язку) управляється ТД.

У висхідній лінії зв'язку стандарт IEEE 802.11ax визначає механізм розподіленого довільного доступу на основі OFDMA, який випадково вибирає блоки ресурсів, призначені точкою доступу для передачі PPDU висхідної лінії зв'язку. Фрейм тригера включає параметр для ініціювання довільного доступу в висхідній лінії зв'язку.

2) Розрахований на багато користувачів (MU) MIMO низхідній і висхідній лінії зв'язку. У MU-MIMO передачі на кілька станцій перекриваються в одних і тих же частотно-часових ресурсах (тобто кілька станцій одночасно зв'язуються з базовою станцією, оснащеною декількома антенами) при використанні просторового рознесення каналу зв'язку.

TGax має намір додати MU-MIMO висхідної лінії зв'язку для роботи разом з MU-MIMO низхідної лінії зв'язку. У висхідній лінії зв'язку MU-MIMO декільком станціям дозволено одночасно передавати дані на приймач по одним і тим самим частотним ресурсам. Подібно до OFDMA, PPDU на основі тригера використовується для вказівки передавальним станціям, коли передавати PPDU MU-MIMO по висхідній лінії зв'язку.

3) Розрахована на багато користувачів агрегація. Агрегація фреймів була введена в IEEE 802.11n для зменшення накладних витрат за рахунок можливості передачі декількох фреймів даних в одному каналі доступу (за умови, що вони мають один і той же пункт призначення). IEEE TGax ще більше розширює процедуру агрегації, визначаючи схему розрахованої на багато користувачів агрегації, яка дозволить здійснювати відправку фреймів багатьох одержувачів. Це зменшує накладні витрати передачі.

1) Щоб зменшити та зберегти споживану енергію, TGax удосконалює поточний стан сну та впроваджує методи енергозбереження, які дозволяють або продовжити час сну, або скоротити час неспання. Ці механізми корисні в мережах з високою густиною та при роботі з низьким енергоспоживанням.

Також, TGAx також вивчає можливість повторного використання різних методів енергоефективності, запропонованих для майбутнього стандарту IEEE 802.11ah (наприклад, Target Wake Time (TWT), де процедура та розклад сну дозволені точкою доступу для зв'язаних станцій.).

#### **1.4 Очікувані проблеми, пов'язані з використанням Wi-Fi**

Оскільки IEEE 802.11ah буде використовуватися разом з передовими бездротовими стільниковими технологіями, такими як Long Term Evolution (LTE) або його розширена версія (LTE-A), ми підкреслюємо очікувану проблему співіснування. Виправлення до IEEE 802.11ah також розглядається як життєздатна комунікаційна мережа для підтримки парадигми Інтернету речей (IoT).

Крім мереж Wi-Fi інші технології широкосмугового доступу розглядають можливість почати конкурувати на арені неліцензійного спектру. LTE у неліцензійному діапазоні був оцінений форумом LTE-U та 3GPP для боротьби з вибуховим зростанням обсягу трафіку.

Застарілий стандарт IEEE 802.11 використовує процес прослуховування перед розмовою (LBT) на основі PHYSSA перед передачею фрейму даних. PHYSSA складається з методів вимірювання фізичної несучої (PHYCS) та виявлення фізичної енергії (PHYED). Метод PHYCS використовується для виявлення та декодування преамбули фреймів інших станцій Wi-Fi: якщо рівень енергії виявленої преамбули перевищує CST, канал вважається зайнятим. PHYED (вперше представлений у IEEE 802.11a для протидії шуму, що генерується передавачами OFDM, а потім розвинений для виявлення будь-якого сигналу в загальному каналі) працює для визначення наявності в каналі будь-якої енергії.

На відміну від Wi-Fi, де пристрої використовують розподілений механізм для боротьби за доступ до бездротового середовища, LTE

покладається на базові станції як центральні планувальники для доступу до середовища.

Щоб скоротити час виведення ринку першої хвилі пристроїв LTE, сумісних із частотою 5 ГГц, початкова структура LTE-U прагне мінімального впливу поточної специфікації і покладається на LBT.

Натомість LTE-U включає динамічну схему включення/вимкнення, звану адаптивною передачею з визначенням несучої (CSAT). CSAT дозволяє планувати передачі LTE-U відповідно до робочого циклу (де період відключення вибирається на основі виявленої активності каналу). Ранні дослідження співіснування Wi-Fi та неліцензійного LTE вказують на невідповідність результатів моделювання та демонстрації. Деякі результати показують, що відсутність LBT у LTE-U викликає проблему співіснування [8], тоді як інші результати вказують на незначний вплив або відсутність [9]. Однак правила використання спектра, визначені у стандарті Європейського інституту телекомунікаційних стандартів (ETSI) EN 301 893, вимагають використання LBT у діапазоні ISM 5 ГГц по всій Європі.

Варіант неліцензійного LTE 3GPP називається доступом із підтримкою ліцензії (LAA-LTE). Метою LAA-LTE є розробка специфікацій LTE для глобальної гармонізації, що забезпечує справедливе співіснування зі стандартом IEEE 802.11. LAA-LTE використовує схему доступу до середовища, аналогічну до схеми розширеного доступу до розподілених каналів (EDCA) стандарту IEEE 802.11. EDCA використовує визначення несучої та механізм відстрочення на основі пріоритетів.

Можна сказати, що в той час як LAA-LTE та IEEE 802.11 використовують схожі механізми доступу до середовища і, таким чином, конкурують у порівнянних умовах, LTE-U використовує різний підхід, що підходить не для всіх областей регулювання. Однак з урахуванням майбутньої специфікації LTE-U, яка вводить LBT CSAT, необхідно буде переглянути дослідження співіснування IEEE 802.11ax і LTE-U.

Таким чином, LTE, будучи централізованою схемою планування, змінить екосистему в спектрі, що не ліцензується. Різниця у технологіях призведе до відсутності загального каналу управління між LTE та Wi-Fi. Нові технології, запропоновані у IEEE 802.11ax, допоможуть боротися з завадами.

Незважаючи на вищезгадану проблему співіснування, розробка безшовних методів, що дозволяють вищезгаданим технологіям працювати шляхом поєднання їх можливостей, може надати користувачам переконливий досвід. LTE та агрегування каналів Wi-Fi (LWA) — ще одна пропозиція, висунута 3GPP. На відміну від LTE-U та LAA-LTE, LWA представляє не новий механізм співіснування, а структуру взаємодії. Найбільш важливим аспектом LWA є те, що його можна включити за допомогою простих оновлень програмного забезпечення та забезпечити одночасну потокову передачу даних через інтерфейси Wi-Fi і LTE, використовуючи спеціальні протоколи, такі як багатоколіїний TCP (MPTCP).

Найбільш важливою частиною інфраструктури Інтернету речей є система бездротового зв'язку, яка діє як міст для доставки даних та керуючих повідомлень. Однак існуючим стільниковим технологіям не вистачає здатності підтримувати обмін величезними обсягами даних від багатьох пристроїв з батарейним живленням, розташованих на великій території. WLAN на базі IEEE (через простоту розгортання та економічної ефективності) можуть використовуватися як життєздатну альтернативну технологію для Інтернету речей тільки в тому випадку, якщо будуть подолані обмеження, пов'язані з високим енергоспоживанням, радіусом дії, кількістю зв'язаних станцій та проблемою ефективності.

Що стосується поточних удосконалень стандарту IEEE 802.11, запропонована поправка до IEEE 802.11ah [11] (сфокусована на роботі в діапазоні частот нижче 1 ГГц) розроблена спеціально для додатків Інтернету речей. Ключові аспекти IEEE 802.11ah – це покращене енергозбереження (наприклад, за рахунок TWT і більш тривалих періодів очікування), найкраще покриття (з використанням нижчої смуги частот і надійнішої

модуляції та кодування), а також здатність одночасно обробляти 8000 вузлів. Затримки в процесі розробки можуть призвести до ситуації, коли IEEE 802.11ah зіткнеться з жорсткою конкуренцією після своєї появи з боку інших вже представлених та перспективних технологій, які працюють на ринку IoT.

Вищезгадана проблема призвела до того, що TGах вивчає нові пропозиції для реалізації сценаріїв використання Інтернету речей. Нещодавно робоча група IEEE 802.11 створила нову групу за інтересами під назвою Long Range Low Power (LRLP) для задоволення потреб міжмашинної взаємодії (M2M), Інтернету речей, управління енергоспоживанням і сенсорних додатків. Ця група має намір розробити методи, які забезпечують роботу Wi-Fi більшої дальності в діапазоні 2,4 ГГц. Ця нова технологія ставить нове завдання співіснування для наступного покоління WLAN. В рамках LRLP і TGах обговорювалися два підходи: і) впровадження вузькосмугової передачі множинного доступу з ортогональним частотним поділом каналів (OFDMA) з меншим рознесенням піднесуть і ii) впровадження передач LRLP у формі модуляції однієї несучої в новій схемі.

## 1.5 Порівняння сигналів Wi-Fi 5 і Wi-Fi 6

Символ OFDM для Wi-Fi 5 має такі часові характеристики (рис. 1.2):



Рисунок 1.2 – Символ OFDM для Wi-Fi 5

Відбиті сигнали надходять у МС з затримками (рис. 1.3).

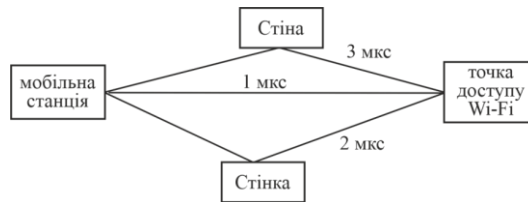


Рисунок 1.3 – Внутрішні багатоканальні відбиття

Відстань між підносійними OFDM у стандарті Wi-Fi 5 (рис. 1.4).

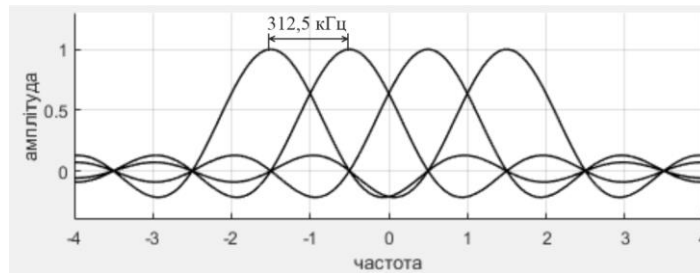


Рисунок 1.4 – Підносійні OFDM сигналу для Wi-Fi 5

Для стандарту IEEE 802.11n/ac, відстань між підносійними складає 312,5 кГц, а для стандарту IEEE 802.11ax - складає 78,125 кГц (рис. 1.6).

Тривалість символу для Wi-Fi 6 до 16 мкс, як показано на рисунку 1.5.



Рисунок 1.5 – Символ OFDM для Wi-Fi 6

15 ортогональних підносійних мають вигляд:

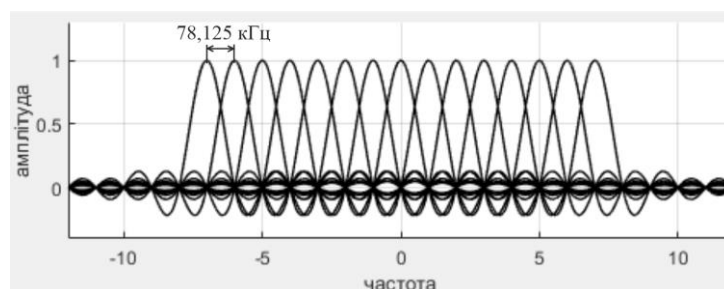


Рисунок 1.6 – Підносійні OFDM сигналу для Wi-Fi 6

Вони займають наступний спектр:

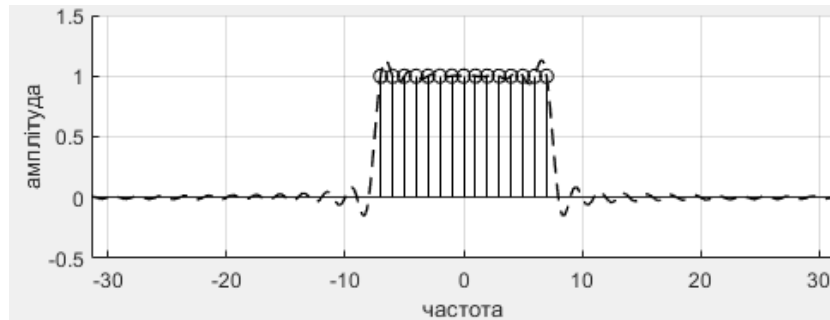


Рисунок 1.7 – Спектр OFDM сигналу для Wi-Fi 6

В OFDMA канал розділений на підканали, які називаються ресурсними одиницями (RU). Кожен RU складається з відомої кількості підносійних.

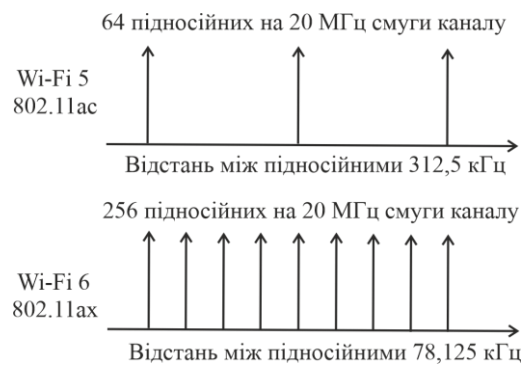


Рисунок 1.8 – Кількість підносійних на 20 МГц смуги каналу

В області часу послідовність OFDM символів показана на рис. 1.10

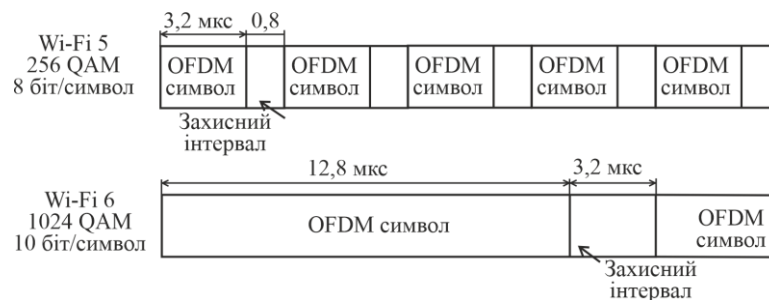


Рисунок 1.9 – Послідовність OFDM символів в області часу

За стандартом для модуляції 16QAM значення величини вектора помилки  $EVM = -19$  дБ; для 64QAM:  $EVM = -27$  дБ; для 256QAM:  $EVM = -32$  дБ; для 1024QAM:  $EVM = -35$  дБ. Різниця між OFDM та OFDMA (рис. 1.10).

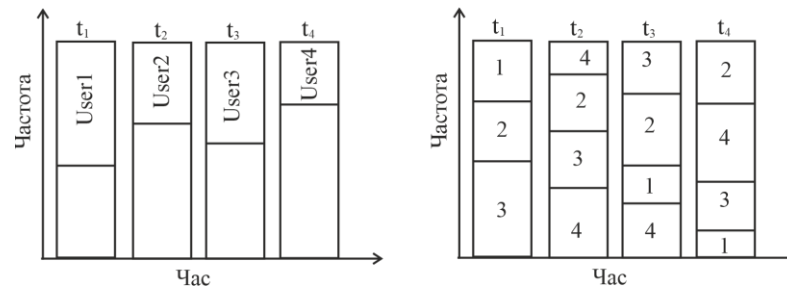


Рисунок 1.10 –Частотно-часове планування сигналу W-Fi

Wi-Fi 6 також підтримує технологію UL MU-MIMO, при якій дані передаються в декількох просторових потоках (рис. 1.10).

### Висновки до першого розділу

Проведений детальний огляд IEEE 802.11ax (майбутній високоефективний стандарт Wi-Fi, розроблений для збільшення пропускної спроможності в середовищах з високою щільністю розміщення та поза приміщеннями). Після того, як ми вкажемо на необхідність та масштаб пропонованої поправки, ми представимо найважливіші технологічні вдосконалення, які ляжуть в основу наступного покоління WLAN. Нарешті, ми наголошуємо на очікуваній проблемі співіснування IEEE 802.11ax з LTE-U. Крім того, ми розкриваємо очікувані можливості та проблеми для TGax у сценаріях Інтернету речей.

## 2 ПОБУДОВА КОДЕРУ І ДЕКОДЕРУ LDPC

### 2.1 Загальні відомості про коди LDPC

У системах зв'язку методи прямого виправлення помилок (FEC) використовуються для виявлення та виправлення помилок в отриманих бітових потоках. Коди з перевіркою парності низької щільності (LDPC) є прикладом кодів ECC, які були вперше запропоновані Робертом Галлагером в 1960 [1] і заново відкриті Маккеєм в середині 1990-х років [2]. Зараз вони використовуються як коди виправлення помилок у багатьох стандартах зв'язку, таких як IEEE 802.11n – стандарт бездротової локальної мережі.

Після введення кодів QC-LDPC [1] і прямого кодування [2], розрахованого з використанням розрідженої матриці перевірки на парність  $H$  без необхідності матриці генератора коду  $G$ , коди QC-LDPC стали невід'ємною частиною стандартів сучасного зв'язку протягом деякого часу.

Це варіюється від IEEE 802.16, через IEEE 802.11ax, DVB-S2 до останньої версії 5G 3GPP TS 38.212 і стандарту мережі Ethernet IEEE 802.

Через швидке декодування надвисокі швидкості 200 Гбіт/с та 400 Гбіт/с IEEE 802.3-2018 наказують використання кодів Ріда-Соломона (RS) для відповідності суворим обмеженням затримки. Незважаючи на те, що коди RS не досягають виняткових характеристик виправлення помилок кодів LDPC, близьких до межі пропускнуої спроможності, дослідження в цій винятковій області продовжуються. Ймовірно, найскладніші конструкції кодів LDPC представлені в специфікаціях консультативного комітету із систем космічних даних (CCSDS), при цьому ведеться інтенсивна дослідницька діяльність: пропонуються нові апаратні кодери для стандартних кодів CCSDS QC-LDPC, прискорення яких досягається головним чином за рахунок реструктуризації структури кодера та використання паралелізму. Проведено порівняння декількох алгоритмів декодування з жорстким і м'яким рішенням для коду

QC-LDPC(128512, 64256), розглядається потенційне енергозбереження супутникової лінії зв'язку. Описується, як можна включити декодування ансамблю автоморфізмів (AED) для кодів QC-LDPC, щоб поліпшити характеристики помилок для кодів з більш коротким  $N$  (від 128 до 256), що використовуються в специфікаціях Wi-Fi, 5G і CCSDS, і порівнюють продуктивність декодування поширення довіри, мінімальної суми та нейронної нормалізованої мінімальної суми (N-NMS) у контексті кодів лінійного добутку (LPC), також визначених у CCSDS. У літературі розглядається модифікація існуючих кодів LDPC для дослідження бажаних властивостей довжини коду, а також представлена математична основа для побудови кодів QC-LDPC з бажаним обхватом, наведені приклади, засновані на протографах, визначених CCSDS.

У будь-якому випадку коди LDPC стали повсюдними майже у всіх технологіях широкопasmового бездротового зв'язку. Реалізації кодерів і декодерів також численні. Однак це апаратні реалізації: або конструкції ASIC, призначені для інтеграції в модеми, або засновані на FPGA. Розглядаються суто програмні декодери: універсальні реалізації кодера/декодера C++, але не включає практичні, стандартні конструкції  $H$ -матриць. Також наведені коди IEEE802.11 та окремі реалізації кодера/декодера MATLAB та C, коди CCSDS LDPC та надається реалізації у MATLAB. Реалізація C++ визначає SIMD-прискорення з використанням векторизованої арифметики AVX-512, орієнтованої на стандарти DVB-S2 і DVB-T2, представлена реалізація MATLAB для кодера та декодера 5G New Radio TS38.212 Використовується паралельне виконання операцій (SIMD- за допомогою однієї інструкції, виконується обробка декількох даних).

Спільним знаменником всіх цих реалізацій є відсутність інтеграції швидкого коду C/C++ з MATLAB. Ця інтеграція є важливою, оскільки непрактично писати цілі моделі на C або підтримувати дві паралельні (і потенційно функціонально різні) версії для MATLAB та C/C++. Очевидним рішенням було використання швидкого коду C/C++ в MATLAB шляхом

реалізації оболонок MEX-файлів, що дозволяють викликати такий метод безпосередньо з MATLAB. (Макроси Application Express називають MEX файлами і використовують як Macro Express, файли відносять до категорії файлів даних, розроблених Insight Software Solutions.) Така реалізація існує як комерційний продукт із закритим початковим кодом і є частиною Communications Toolbox. Фактично, коди LDPC у MATLAB мають власну історію: перше впровадження LDPC у MATLAB відбулося у 2007 році. Вбудовані функції кодування і декодування LDPC дозволяють розширювати практичні матриці-прототипи QC-LDPC до розріджених матриць H, але завдання отримання матриць-прототипів залишається на розсуд користувача. Реалізації доступні лише у вигляді двійкових модулів MEX. Більше того, для дослідника, який хотів би протестувати алгоритми кодування/декодування, одночасно оцінюючи практичні коди LDPC, що використовуються в сучасних стандартах, такі реалізації малоприматні.

Саме тут знаходить застосування наша реалізація: вона забезпечує відносно швидку C99-реалізацію кодера і декодера QC-LDPC, яку можна інтегрувати в будь-який проект C, що безпосередньо містить практичні коди LDPC, що використовуються в IEEE802.11ax-2021 (фактично визначено в більш ранній специфікації IEEE802.11-2020) та стандартах IEEE 802.16-2017, а також надає оболонки MEX та допоміжні функції для безшовної інтеграції MATLAB для статистичної оцінки водоспадних кривих. Щоб полегшити переносимість і сумісність з платформою ARM, не використовувалися зовнішні бібліотеки, такі як оптимізований Intel MKL.

## 2.2 Будова кодів LDPC

Матриця перевірки парності коду LDPC визначає BER, пропускну здатність та складність декодера LDPC. Матриці перевірки парності, що використовуються в стандарті IEEE 802.11n, мають багаторівневу структуру і

складаються з зсунутих версій одиничних матриць, об'єднаних в 12 різних матриць для довжин блоків 648, 1296 і 1944 та кодових швидкостей 1/2, 2/3, 3/4, 5/6 [3]. Матриця перевірки парності 324x648, яка використовується в стандарті IEEE 802.11n для довжини блоку 648 і швидкості коду 1/2, показана на рисунку 2.1. Рівень складається з декількох рядків, і конкатенація цих рівнів утворює матрицю перевірки на парність. Наприклад, матриця перевірки парності швидкості коду 1/2 складається з 12 рівнів, кожен рівень складається з 24 підматриць розміром 27x27, які є або нульовими матрицями, або зсунутими одиничними матрицями.

$$P=[0 \ -1 \ -1 \ -1 \ 0 \ 0 \ -1 \ -1 \ 0 \ -1 \ -1 \ 0 \ 1 \ 0 \ -1 \ -1 \ -1 \ -1 \ -1 \ -1 \ -1 \ -1 \ -1 \ -1 \ 22 \ 0 \ -1 \ -1 \ 17 \ -1 \ 0 \ 0 \ 12 \ -1 \ -1 \ -1 \ -1 \ 0 \ 0 \ -1 \ -1 \ -1 \ -1 \ -1 \ -1 \ -1 \ -1 \ -1 \ 6 \ -1 \ 0 \ -1 \ 10 \ -1 \ -1 \ -1 \ 24 \ -1 \ 0 \ -1 \ -1 \ -1 \ 0 \ 0 \ -1 \ -1 \ -1 \ -1 \ -1 \ -1 \ -1 \ -1 \ -1 \ 2 \ -1 \ -1 \ 0 \ 20 \ -1 \ -1 \ -1 \ 25 \ 0 \ -1 \ -1 \ -1 \ -1 \ -1 \ 0 \ 0 \ -1 \ -1 \ -1 \ -1 \ -1 \ -1 \ -1 \ -1 \ 23 \ -1 \ -1 \ -1 \ 3 \ -1 \ -1 \ -1 \ 0 \ -1 \ 9 \ 11 \ -1 \ -1 \ -1 \ -1 \ 0 \ 0 \ -1 \ -1 \ -1 \ -1 \ -1 \ -1 \ 24 \ -1 \ 23 \ 1 \ 17 \ -1 \ 3 \ -1 \ 10 \ -1 \ -1 \ -1 \ -1 \ -1 \ -1 \ -1 \ 0 \ 0 \ -1 \ -1 \ -1 \ -1 \ -1 \ 25 \ -1 \ -1 \ -1 \ 8 \ -1 \ -1 \ -1 \ 7 \ 18 \ -1 \ -1 \ 0 \ -1 \ -1 \ -1 \ -1 \ -1 \ 0 \ 0 \ -1 \ -1 \ -1 \ -1 \ 13 \ 24 \ -1 \ -1 \ 0 \ -1 \ 8 \ -1 \ 6 \ -1 \ -1 \ -1 \ -1 \ -1 \ -1 \ -1 \ -1 \ -1 \ 0 \ 0 \ -1 \ -1 \ -1 \ 7 \ 20 \ -1 \ 16 \ 22 \ 10 \ -1 \ -1 \ 23 \ -1 \ -1 \ -1 \ -1 \ -1 \ -1 \ -1 \ -1 \ -1 \ -1 \ 0 \ 0 \ -1 \ 11 \ -1 \ -1 \ -1 \ 19 \ -1 \ -1 \ -1 \ 13 \ -1 \ 3 \ 17 \ -1 \ -1 \ -1 \ -1 \ -1 \ -1 \ -1 \ -1 \ 0 \ 0 \ -1 \ 25 \ -1 \ 8 \ -1 \ 23 \ 18 \ -1 \ 14 \ 9 \ -1 \ -1 \ -1 \ -1 \ -1 \ -1 \ -1 \ -1 \ -1 \ -1 \ -1 \ 0 \ 0 \ 3 \ -1 \ -1 \ -1 \ 16 \ -1 \ -1 \ 2 \ 25 \ 5 \ -1 \ -1 \ 1 \ -1 \ -1 \ -1 \ -1 \ -1 \ -1 \ -1 \ -1 \ 0]$$

Рисунок 2.1 – Матриця перевірки парності для довжини блоку 648 та швидкості коду 1/2

У літературі запропоновано кілька алгоритмів декодування кодів LDPC [4]. Ми використовуємо алгоритм декодування мінімальної суми з багаторівневим поширенням довіри в області логарифмічного відношення правдоподібності (LLR), оскільки він задовольняє вимогам до пропускну здатності та BER стандарту IEEE 802.11n та має низьку складність.

Оскільки апаратне забезпечення паралельного декодера LDPC не масштабується для великих матриць перевірки на парність [5], ми представляємо малопотужне гібридне обладнання декодера LDPC для стандарту бездротової локальної мережі IEEE 802.11n. Апаратне забезпечення декодера LDPC має 27 каналів даних контрольних вузлів та

пам'ять змінних вузлів 24x162. Апаратне забезпечення реалізовано на Verilog HDL та перевірено на коректну роботу у FPGA Xilinx Virtex II.

Потужність реалізації на FPGA Xilinx Virtex II оцінюється в 2052 мВт для довжини блоку 648 і швидкості коду 1/2 і 1989 мВт для довжини блоку 648 і швидкості коду 5/6 з використанням інструменту Xilinx XPower. Ми пропонуємо два нові методи – переупорядкування підматриці та диференціальний зсув – для зниження енергоспоживання апаратного забезпечення декодера LDPC.

Ми застосували методи зменшення завад, переупорядкування підматриць та диференціального зсуву для обладнання декодера LDPC. Ці методи не впливають на BER декодера LDPC. Для довжини блоку 648 і швидкості коду 1/2 ці три методи разом знизили енергоспоживання апаратного забезпечення декодера LDPC на 23,7% до 1565,84 мВт. Для довжини блоку 648 і швидкості коду 5/6 разом знизили енергоспоживання апаратного забезпечення декодера LDPC загалом на 38,98% до 1214,22 мВт. У літературі запропоновано кілька апаратних архітектур гібридного LDPC-декодера [6]. Деякі з цих декодерів LDPC пропонуються для IEEE 802.11n.

### 2.3 Декодер кодів LDPC

Декодування LDPC виконується з урахуванням матриці перевірки парності, що складається з «0» і «1», які визначаються з рівняння перевірки парності. Приклад матриці перевірки парності 4x8 показаний рис. 2. Матриця перевірки парності MxN має M рівнянь перевірки парності і N змінних.

$$H = \begin{vmatrix} 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 \end{vmatrix}$$

Рисунок 2.2 – 4X8 матриця перевірки парності

Для матриці перевірки парності  $M \times N$  має  $M$  перевірочних вузлів та  $N$  змінних вузлів, які обмінюються інформацією між собою ітеративно і відповідно до алгоритму декодування LDPC. «1» у матриці перевірки парності визначають зв'язки між вузлами змінних та вузлами перевірки. Обмін інформацією здійснюється лише між вузлами, підключеними один до одного. Процес декодування LDPC для матриці перевірки парності  $4 \times 8$  показано на рисунку 2.3.

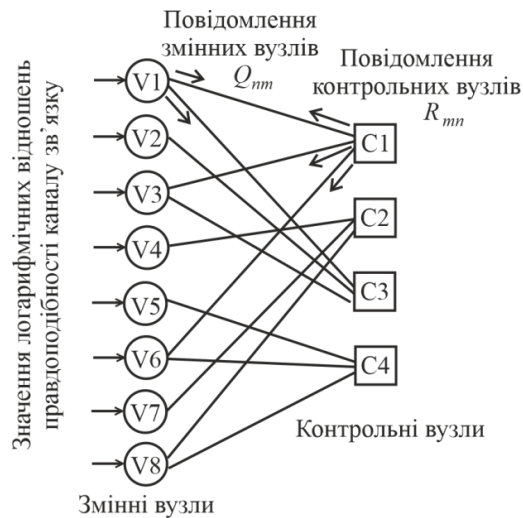


Рисунок 2.3 – LDPC код для  $4 \times 8$  матриці перевірки парності

Змінні вузли отримують м'яку інформацію, ймовірність відношення правдоподібності цього біта дорівнює 1 або 0 з каналу, і ця інформація ітеративно передається між перевірочними вузлами та змінними вузлами, щоб задовольнити рівняння перевірки парності, заданим матрицею парності [1, 2]. Цю операцію можна виконати в логарифмічній області, щоб спростити операції множення до операцій складання, і в цьому випадку декодер отримує коефіцієнти логарифмічної правдоподібності ( $\ln r$ ) з каналу [4]. Цей алгоритм можна додатково спростити до алгоритму декодування із мінімальною сумою з невеликим погіршенням BER. Кроки алгоритму декодування мінімальної суми наступні:

1. Візьміть значення  $\ln r$  з каналу для кожного вузла змінної як початкові повідомлення вузла змінної.

$$Q_n = LLR(n) \quad (1)$$

2. Оновіть кожен контрольний вузол повідомленнями змінних вузлів, до яких вони підключені, відповідно до алгоритму мінімальної суми.

$$R_{nm} = \prod_{n' \in C \setminus n} \text{sign}(Q_{n'm}) \times \min_{n' \in C \setminus n} (Q_{n'm}),$$

де  $C$  - набір змінних вузлів, підключених до перевірного вузла.

3. Оновіть кожен вузол змінної, вказавши повідомлення вузла перевірки, до яких вони підключені.

$$Q_{nm} = LLR(n) + \sum_{m' \in V \setminus m} R_{m'n},$$

де  $V$  - множина перевірочних вузлів, підключених до змінного вузла.

4. Після кожного рівня обчисліть вихідні дані декодера, підсумовуючи всі повідомлення контрольного вузла кожного вузла змінної.

$$Q_n = LLR(n) + \sum_{m' \in V} R_{m'n}$$

5. Наприкінці, жорстке рішення приймається відповідно до вихідних даних м'якого декодера.

Коли алгоритм декодування з мінімальною сумою реалізується з використанням апаратного забезпечення гібридного декодера LDPC, його продуктивність BER може бути покращена за рахунок використання методу багаторівневого декодування, при якому оновлення повідомлень виконуються не тільки після завершення всієї матриці перевірки парності, але також після кожного рівня перевірки матриця [6, 7]. Багаторівневе декодування може бути використане для матриць контролю парності з

багаторівневою структурою, таких як матриці контролю парності, що використовуються в стандарті IEEE 802.11n. Наприклад, для матриці перевірки парності, що використовується для довжини блоку 648 і швидкості коду 1/2 у стандарті IEEE 802.11n, після того, як 27 перевірочних вузлів завершують алгоритм мінімальної суми для вузлів змінних, до яких вони підключені на одному рівні, ці змінні вузли оновлюються, та 27 перевірочних вузлів використовують ці оновлені повідомлення для наступного рівня. Оскільки оновлення повідомлення також виконується після завершення кожного рівня в ітерації, час, що витрачається на обробку рівня, називається підітерацією. Отже, для матриці перевірки парності, що використовує швидкість коду 1/2, за одну ітерацію виконується 12 подітерацій.

Розглянемо коди LDPC у сучасних стандартах. Відкриті Галлагером коди LDPC є лінійними блоковими кодами (LBC) з базовими параметрами  $[N, K]$ , що визначаються їх розрідженою матрицею контролю парності  $H$ . Можна вивести генераторну матрицю  $G^{K \times N}$  яка може бути отримана від  $H^{M \times N}$  (визначено:  $M = N - K$  і швидкість коду  $R = K / N$ ), але він навряд чи буде розрідженим і, отже, непридатним для практичного кодування з високою пропускну здатністю. Однак кодування з використанням матриці  $G$  дуже просто описується множенням слова даних – вектора  $i$  довжиною  $K$  біт від  $G$  для отримання кодового слова  $c$  довжини  $N$ . Для систематичного коду систематична частина  $s$  кодового слова є підвектором  $c$ , рівним  $i$ . Ми можемо позначити:

$$c^{1 \times N} = \begin{bmatrix} s^{1 \times K} & p^{1 \times M} \end{bmatrix} = i^{1 \times K} \cdot G^{K \times N}, \quad s = i \quad (2.1)$$

Розрідженість великої  $H$ -матриці не тільки забезпечує чудові властивості корекції помилок кодів LDPC, але також дозволяє ефективно зберігати  $H$  у пам'яті: замість використання масиву розміром  $M \times N$  необхідно використовувати для зберігання тільки індекси одиниць. Будемо

використовувати  $n$  як індекс стовпця (також відомий як змінний індекс вузла в контексті м'якого декодування) та  $m$  для індексу рядка  $H$  (також відомого як індекс контрольного вузла). Тоді для кожного  $n \in \{1, 2, \dots, N\}$  набір індексів рядків одиниць у даному стовпці  $H$  можна назвати околицею  $n$  і позначити  $M(n)$ , у той час як для кожного  $m \in \{1, 2, \dots, M\}$ ,  $N(m)$  є набір індексів стовпців одиниць у цьому ряду  $m$ . Будь-яка розріджена двійкова матриця  $H$  може зберігатися ефективним способом використання пам'яті як масиву масивів  $M(n)$  чи масиву масивів  $N(m)$ . Число елементів  $g(m) = |N(m)|$  можна назвати степенем рядка  $m$ , а символ  $f(n) = |M(n)|$  позначає степінь стовпця  $n$ . Для практичних нерегулярних кодів LDPC  $g(m)$  і  $f(n)$  не є постійними по  $m$  і  $n$ , і становлять інтерес їхнє середнє і максимальне значення.

$$H = \begin{pmatrix} P_{0,0} & P_{0,1} & \dots & P_{0,Nb-1} \\ P_{1,0} & P_{1,1} & \dots & P_{1,Nb-1} \\ \dots & \dots & \dots & \dots \\ P_{Mb-1,0} & P_{Mb-1,1} & \dots & P_{Mb-1,Nb-1} \end{pmatrix} \quad (2.2)$$

Підклас кодів QC-LDPC, що використовується в стандартах зв'язку [3-5], був навмисно побудований з такою структурою  $H$ -матриці, яка дозволяє повністю та ефективно реалізувати процес кодування. Як показано у (2.2), двійкова розріджена матриця  $H$  спроектована як блочна матриця, що складається із сітки підматриць  $P_{i,j}$  розміру  $Z \times Z$ .

Кожна з підматриць може бути нульовою матрицею або одиничною перестановочною матрицею, причому єдиною дозволеною перестановкою є циклічний зсув, що призводить до циркулянту  $P_{i,j}$  іноді званою матрицею циклічного зсуву.

Обмеження на  $P_{i,j}$  дозволяють стисле представлення  $H$  як модельної матриці  $H_{bm}$  розміру  $M_b \times N_b$  (позначаючи  $M_b = N_b - K_b$ ), де кожен елемент

$H_{bm}(i, j)$  представляє циркулянтну матрицю перестановок  $P_{i,j}$  і  $N = N_b \times Z$ ,  $K = K_b \times Z$ . За згодою значення  $H_{bm}(i, j)$ , що дорівнює -1, представляє нульову підматрицю  $P_{i,j}$ , значення 0 - одиничну підматрицю, а будь-яке позитивне число представляє одиничну матрицю, зсунуту по колу на  $H_{bm}(i, j)$ . Стиснуті матриці  $H_{bm}$  потім безпосередньо визначаються стандартами з різними матрицями моделей, визначеними для кожної довжини кодового слова  $N$ , як [5], або загальною структурою для найбільшого значення  $N$ , заданої разом з відповідним перетворенням для всіх інших підтримуваних значень  $N$ . У всіх випадках велика розріджена двійкова матриця  $H$  виходить з  $H_{bm}$  шляхом розширення кожного елемента  $H_{bm}$  до квадратної матриці розміру  $Z$ . Для обох стандартів матриці  $H$ , що представляють інтерес, мають додаткову подібність:  $H_{bm}$  можна додатково розділити на три підматриці як представлено виразом:

$$H_{bm}^{M_b \times N_b} = \begin{bmatrix} H_{b1}^{M_b \times K_b} & H_{b2}^{M_b \times M_b} \end{bmatrix} = \begin{bmatrix} H_{b1}^{M_b \times K_b} & h_b^{M_b \times 1} & H_{b2}^{M_b \times (K_b - 1)} \end{bmatrix} \quad (2.3)$$

З точки зору індексації підматриць підматриці  $H_{b1}$ ,  $h_b$  і  $H_{b2}$  можна визначити як:

$$H_{b1} = H_{bm} [0, 1, \dots, M_b - 1; 0, 1, \dots, K_b - 1] \quad (2.4)$$

$$h_b = H_{bm} [0, 1, \dots, M_b - 1; K_b] \quad (2.5)$$

$$H_{b2} = H_{bm} [0, 1, \dots, M_b - 1; K_b + 1, \dots, N_b - 1], \quad (2.6)$$

де  $h_b$  - перший стовпець матриці  $H_{b2}$ . Ми використовуємо математично неточну індексацію, починаючи з нуля замість одиниці. Це потрібно для сумісності з стандартом [3].

Алгоритм прямого кодування додатково вимагає, щоб вектор-стовпець  $h_b$  мав спеціальну структуру, що містить два парних (рівних) значення в першій та останній позиціях та одне позитивне значення серед інших позицій. Всім іншим елементам надається значення -1, що вказує на пізніше розширення до нульових матриць. Також потрібна структура підматриці  $H_{b2'}$ , де дві нульові діагоналі в матриці з усіма -1  $H_{b2}$  перетворюються на дводіагональну розширену підматрицю  $H$ . Рівняння (2.7) показує приклад структури  $h_b$  і  $H_{b2'}$  для  $N_b = 24$ ,  $K_b = 18$  для коду QC-LDPC  $R = 3/4$ :

$$H_{b2}^{6 \times 6} = \left[ \begin{array}{c|c} h_b^{6 \times 1} & H_{b2'}^{6 \times 5} \end{array} \right] = \left[ \begin{array}{c|ccccc} 0 & 0 & -1 & -1 & -1 & -1 \\ -1 & 0 & 0 & -1 & -1 & -1 \\ 80 & -1 & 0 & 0 & -1 & -1 \\ -1 & -1 & -1 & 0 & 0 & -1 \\ -1 & -1 & -1 & -1 & 0 & 0 \\ 0 & -1 & -1 & -1 & -1 & 0 \end{array} \right] \quad (2.7)$$

Обидва стандарти визначають набір швидкостей кодування та розмірів кодових слів, які впливають на розробку кодера та декодера.

Важливо, що різні розділи стандарту Wi-Fi 6 визначають кілька різних кодів LDPC і різні версії кожного стандарту можуть використовувати деякі коди LDPC. У цьому розділі ми будемо мати справу з кодами, визначеними в IEEE 802.11-2020 [5].

Всі довжини кодових слів  $N$  кратні 8 біт. Це не вірно для всіх розмірів слова даних  $K$ . Коефіцієнт розширення (або розмір підблоку)  $Z$  навмисно обраний таким чином, щоб набувати значення 27, 54 або 81, очевидно, з реалізацією апаратного кодера у вигляді спеціально розробленої схеми ASIC.

## 2.4 Пряме кодування QC-LDPC

Спеціальна структура матриці  $H$  була розроблена для забезпечення алгоритму прямого кодування, при якому при кодуванні систематичного коду використовується матриця  $H$  замість матриці генератора коду  $G$ . Для практичної реалізації кодера  $C$  фактично використовується модельна матриця  $H_{mb}$ , тому велика розріджена двійкова матриця  $H$  або навіть підматриці перестановок  $P_{i,j}$  не зберігаються у пам'яті. Оскільки множення деяких субвекторів  $i$  довжини  $Z$  на матрицю  $P_{i,j}$  за результатом ідентично циклічному зсуву, фактичною операцією, яку слід реалізувати як будівельний блок кодера, є циклічний зсув субвекторів розміру  $Z$  біт на  $H_{bm}(i, j)$  позиції. Алгоритм прямого кодування, можна резюмувати наступним чином:

Інформаційний блок  $i$  (також відомий як систематичний вектор  $s$ ) розділений на  $K_b$  блоки по  $Z$  біт. Згрупований  $i$  можна позначити як матрицю  $u$ , де кожен елемент  $u_i$  являє собою вектор-стовпець:

$$u^{Z \times K_b} = \begin{bmatrix} u_0 & u_1 & \dots & u_{K_b-1} \end{bmatrix} \quad (2.8)$$

$$u_i = \begin{bmatrix} s_{iZ} & s_{iZ+1} & \dots & s_{(i+1)Z-1} \end{bmatrix}^T \quad (2.9)$$

Послідовність парності  $p$  обчислюється в блоках  $Z$  біт безпосередньо з використанням матриці  $H_{bm}$ . Знову ж таки, послідовність парності  $p$  може бути виражена у вигляді матриці у вигляді блоків, де кожен елемент  $v_i$  являє собою вектор-стовпець довжиною  $Z$  біт:

$$p^{Z \times M_b} = \begin{bmatrix} v_0 & v_1 & \dots & v_{M_b-1} \end{bmatrix} \quad (2.10)$$

$$v_i = \begin{bmatrix} p_{iZ} & p_{iZ+1} & \cdots & p_{(i+1)Z-1} \end{bmatrix}^T \quad (2.11)$$

Тут використовуються кілька нестандартних позначень векторів  $u_i$  і  $v_i$ . Кодування поділяється на ініціалізацію та рекурсію. Спочатку ініціалізація визначається так:

$$v_0 = P_{H_{bm}(y, K_b)}^{-1} \cdot \left( \sum_{i=0}^{M_b-1} \sum_{j=0}^{K_b-1} P_{H_{bm}(i, j)} u_j \right), \quad (2.12)$$

де  $H_{bm}(i, j)$  є елементом (потенційно масштабованої) матриці моделі  $H_{bm}$ ,  $P_{H_{bm}(i, j)}$  є циркулянтною матрицею перестановок, а множення на  $P_{H_{bm}(i, j)}$  реалізує операцію циклічного зсуву на підвекторі  $u_j$ . Елемент  $H_{bm}(y, K_b)$  є єдиним невід'ємним елементом підвектора  $h_b$  поза парними значеннями, як показано в прикладі (2.7) з  $H_{bm}(y, K_b) = h_b(2) = 80$ . Зверніть увагу, що значення  $y$  різне для кожного  $H_{bm}$  і має бути знайдено кодером на льоту. Зворотна перестановка  $P^{-1}$  це просто циклічний зсув в протилежному напрямку. Усі операції складання/підсумовування виконуються над GF(2).

Після ініціалізації та успішного обчислення  $v_0$  кодер приступає до ітерації/рекурсії, знаходячи підблоки парності  $v_i$ , що залишилися :

$$v_i = P_{H_{bm}(0, K_b)} v_0 + \sum_{j=0}^{K_b-1} P_{H_{bm}(0, j)} u_j, \quad i = 0 \quad (2.13)$$

$$v_{i+1} = v_i + P_{H_{bm}(0, K_b)} v_0 + \sum_{j=0}^{K_b-1} P_{H_{bm}(0, j)} u_j, \quad i > 0 \quad (2.14)$$

Оскільки суми сум (2.12):

$$S_i = \sum_{j=0}^{K_b-1} P_{H_{bm}(i,j)} u_j, \quad i = 0, \dots, M_b - 1 \quad (2.15)$$

обчислюються під час ініціалізації та повторно використовуються під час ітерації, їх значення вигідно зберігати для повторного використання. Оскільки елементи  $H_{bm}$ , рівні -1, розширюються до нульової підматриці  $P$ , множення на  $P_{H_{bm}}$  призводить до нульового члена і може бути пропущено.

Математичне визначення кодера, визначає конструкцію остаточного алгоритму  $C$ , тоді як фактичні значення параметрів коду  $N$ ,  $K$ ,  $M$  та  $Z$  відіграють важливу роль. Розглянемо дві реалізації кодера. Перша - це реалізація універсального масиву, яка підтримує всі комбінації параметрів коду, але вимагає більше пам'яті, оскільки кожен біт фактично зберігається як елемент масиву  $C$  (зазвичай, але не обов'язково 8-бітний беззнаковий символ). Перевага - простий код, в якому доступ до бітів безпосередньо підтримується оператором доступу до масиву. Недоліком реальних систем передачі/модемів, що обробляють блоки двійкових даних, є необхідність розширення даних усередині кодера в 8 разів.

Друга реалізація - це кодер растрових зображень (упакованих бітів) - справжній двійковий кодер, який обробляє блоки двійкових даних як растрові зображення та отримує доступ до бітів усередині байтів за допомогою побітових операторів. Хоча можна реалізувати циклічні зсуви навіть для підблоків розміру  $Z$ , визначеного в стандарті Wi-Fi {27, 54, 81}, вони вимагають доступу до пам'яті без вирівнювання байт, що знижує продуктивність. Цей підхід був випробуваний і протестований, і виміряна низька пропускна здатність стала причиною відмови від реалізації кодера растрових зображень для цих значень  $Z$ . З іншого боку, стандарт Wi-MAX надає кілька комбінацій параметрів коду, де всі  $N$ ,  $K$ ,  $M$ ,  $Z$  поділяються на 8 (16, 32, або 64), що забезпечує чисту та швидку реалізацію  $C$  з мінімальними вимогами до пам'яті.

Усі рівняння кодера (2.12)-(2.15) були реалізовані безпосередньо, замінивши множення  $H_{bm}(i, j) \cdot u_j$  з циклічним зсувом вектора  $u_j$  на позиції бітів  $H_{bm}(i, j)$  і кожним додаванням з виключаючим побітовим АБО. Мова C не підтримує безпосередньо конструкцію, яка зверталася б до базових інструкцій обертання ISA, тому викликаються два лінійних зсуви змінної і об'єднуються за допомогою побітового оператора АБО. Це один із відомих недоліків мови C, і завдання компіляції трьох пов'язаних операцій C в одну команду обертання залишається на розсуд компілятора.

Оскільки алгоритм кодування визначається у вигляді блоків бітів розміру  $Z$ , які можуть охоплювати кілька байтів або багатобайтових слів, індексування масиву та побітові оператори повинні бути об'єднані для реалізації кругового зсуву, який є основою рівнянь (2.12) - (2.15). Розглянемо процедуру циклічного зсуву кодера растрового зображення з ефективним використанням пам'яті, де визначено три індекси: бітовий індекс  $I_{bBL}$  всередині блоку розміру  $Z$ , бітовий індекс всередині слова  $I_{bw}$  і індекс  $I_w$  слово в масиві. Оскільки масив зберігається у пам'яті з адресацією в байтах, індекс байта  $I_B$  означає зміщення байта від початку масиву. Може бути зручно, але необов'язково, щоб розмір слова містив лише один байт, причому  $I_w = I_B$ . Цей вибір зроблено для простоти прикладу. Позначимо через  $wb$  розмір слова у бітах.

Зсув  $s$  можна концептуально розділити на дві частини: зсув  $sw$ , виражений цілим числом слів, і зсув, що залишився до  $wb - 1$  біт. Показані для вибраних значень  $s = 3, 11, 19$ , всі співпадаючі за модулем  $wb = 8$ , результуючі елементи масиву, хоча і розташовані в різних позиціях масиву, ідентичні порозрядно:  $Out_s = 11[1] = Out_s = 3[0] = Out_s = 19[2]$ .

Довільний циклічний зсув на  $s$  позицій можна представити у вигляді:  $s = sw \cdot wb + sb$ , де  $sw = div(s, wb)$  - часткове обертання масиву в цілих словах (елементах масиву), а  $sb = mod(s, wb)$  - залишкове обертання в бітах (з

використанням стандартного цілісного поділу та операцій за модулем) . Очевидно,  $sb < wb$  . Цей поділ служить мети ефективної реалізації за рахунок комбінації двох операцій: спочатку весь масив зсувається на  $sb$  біт таким чином, що два послідовні елементи масиву об'єднуються разом у результуючий елемент масиву за допомогою порозрядних операцій і де останній елемент масиву також використовується як елемент, що передує першому. Далі решта зсуву масиву за словами  $sw$  реалізується шляхом перетасовування всього масиву з використанням індексації по модулю. Ці два кроки можна поєднати в одне сканування вхідного масиву. Алгоритм A1 надає результуючий алгоритм, що реалізує довільний зсув бітів у масиві, де  $zw$  – кількість слів у масиві, а  $Z = zw \cdot wb$  .

Цей алгоритм ефективний і для будь-якого розміру зсуву від  $s=1$  до  $Z-1$ . З важливим обмеженням розміру блоку  $Z$ , кратного розміру слова  $wb$ . Таким чином, цей алгоритм буде безпосередньо використовуватися для кодування половини підтримуваних розмірів кодових слів, визначених стандартом WIMAX, але при цьому він абсолютно не підходить для Wi-Fi 6.

## 2.5 Універсальний декодер LDPC

У нашій реалізації декодера LDPC вибір алгоритму був мотивований нашими зусиллями створення ідеально точного або, принаймні, практично близького наближення до результатів декодування, отриманих при використанні широко популярного, але власного декодера MATLAB, що надається функцією `ldpcDecode()` пакету MATLAB. Набір інструментів для зв'язку R2021b. Це визначило вибір алгоритму як перевірений часом алгоритм мінімальної суми, визначеного в [40,41], який забезпечує високу пропускну здатність і є хорошим наближенням до набагато більш вимогливого в обчислювальному відношенні алгоритму поширення довіри на основі апостеріорного логарифмічного відношення правдоподібності (LLR).

[42]. Для економії пам'яті була реалізована версія алгоритму мінімальної суми з одним скануванням, як описано в [43], точніше багаторівнева версія з більш швидкою збіжністю, як описано в [44]. Були реалізовані дві версії декодера: одна використовувала арифметику з плаваючою комою - 32-бітний тип C з плаваючою комою, а інша, трохи швидша і ефективніша за пам'яттю, арифметичний декодер з фіксованою комою, де апроксимації LLR зберігалися як 16-бітові короткі цілочисельні значення. .

У статті Хуанга наведено короткий огляд алгоритму мінімальної суми, а також докладний опис його поліпшення за один прохід, оптимізований для пам'яті. Обидва варіанти алгоритму коротко підсумовуються тут: для гаусівського каналу і двійкової модуляції (BPSK) нехай  $x_n$  буде  $n$ -им бітом, що передається,  $n_n$  - вибіркою адитивного білого гаусовського шуму (AWGN), а  $y_n$  - вибіркою в приймачі. Тоді модуляція та передача каналу можуть бути описані за допомогою (2.16), а алгоритм мінімальної суми м'якого декодування працює з метрикою LLR  $Z_n$ , що визначається формулою (2.17):

$$y_n = (-2x_n + 1) + n_n$$

$$Z_n^{(0)} = \ln \frac{p(x_n = 0 / y_n)}{p(x_n = 1 / y_n)} = 2y_n / \sigma^2 \quad (2.17)$$

де  $\sigma^2$  - це дисперсія каналу (може бути опущена у реалізації з мінімальною сумою), а верхній індекс ( $k$ ) вказує номер ітерації. Кожна ітерація реалізує такі кроки:

1. Горизонтальне сканування – оновлення контрольного вузла:

Для кожного  $t$  і кожного  $N(t)$  обчислюють повідомлення  $L_{mn}$  перевірконого вузла на основі повідомлень  $Z_{mn}$  вузла змінної, що надходять від усіх інцидентних змінних вузлів, крім вихідного (з індексом  $n$ ).

$$L_{mn}^{(k)} = \min_{n' \in N(m) \setminus n} |Z_{mn'}^{(k)}| \cdot \prod_{n' \in N(m) \setminus n} \text{sign}(Z_{mn'}^{(k)}) \quad (2.18)$$

2. Вертикальне сканування – оновлення вузла змінної:

Для кожного  $n$  та кожного  $M(n)$  обчислюємо змінне вузлове повідомлення  $Z_{mn}$ , що надходить від вузлів, крім вихідного (з індексом  $m$ ).

$$Z_{mn}^{(k)} = Z_n^{(0)} + \sum_{m' \in M(n) \setminus m} L_{m'n}^{(k)} \quad (2.19)$$

3. Для кожного  $n$  та  $M(n)$  обчисліть апостеріорну оцінку LLR  $Z_n^{(k)}$ :

$$Z_n^{(k)} = Z_n^{(0)} + \sum_{m' \in M(n)} L_{mn'}^{(k)} \quad (2.20)$$

Розглянемо хорстке рішення та припинення дії. Для кожної позиції кодового слова  $n$  обчислюють бітову оцінку (2.21) і якщо ортогональність матриці  $H$  задоволена:  $H\hat{x} = 0$ , припиняємо декодування.

$$\hat{x}_n^{(k)} = \begin{cases} 0, & Z_n^{(k)} > 0 \\ 1, & \text{інакше} \end{cases} \quad (2.21)$$

Значення  $Z_{mn}$  ініціалізуються отриманими вибірками каналу:  $Z_{mn}^{(0)} = Z_n^{(0)}$  та (19) можна переписати за допомогою (2.20):

$$Z_{mn}^{(0)} = Z_n^{(k)} - L_{mn}^{(k)} \quad (2.22)$$

Модифікація алгоритму мінімальної суми з одним скануванням використовує таке спостереження: коли рівняння (18) застосовується до вектора значень, результуючий вектор містить лише елементи з двома

можливими величинами: мінімум вхідних елементів та другий мінімум. Якщо знаки зберігаються окремо від величин, стиснутих у вигляді растрового зображення в одне слово, ця властивість, використана в алгоритмі [3] може бути використана для значного зниження вимог до пам'яті декодера. Наприклад, максимальні степінь контрольного вузла для Wi-Fi 6  $R = 5/6$  дорівнює  $\max(g(m)) = 20$ , тому така оптимізація забезпечує економію 18 величин для кожного контрольного вузла. Крім того, необхідно зберігати лише повідомлення контрольного вузла  $L_{mn}$ , що відразу знижує вимоги до пам'яті майже на 50%. Рівняння (2.18) можна переписати так:

$$L_{mn}^{(k)} = \min_{n' \in N(m) \setminus n} |Z_n^{(k-1)} - L_{mn'}^{(k-1)}| \cdot \prod_{n' \in N(m) \setminus n} \text{sign}(Z_n^{(k-1)} - L_{mn'}^{(k-1)}) \quad (2.23)$$

Щоб зменшити кількість ітерацій, необхідних для отримання бажаного BER, була реалізована подальша модифікація обчислень – багаторівневе декодування, описане в [4]. Без докладного опису ми можемо резюмувати це наступним чином: Горизонтальне сканування (2.23) не виконується по кожному з  $M$  перевірочних вузлів в одному циклі, а організовано в яруси - групи перевірочних вузлів розміру  $Z$  блокових рядків матриці  $H$ . Потім обчислення (2.20), (2.22) та (2.23) чергуються під час кожної ітерації декодера ( $k$ ): (2.23) оцінюється для блоку з  $Z$  рядків/перевірок, після чого оновлене значення  $Z_n$  використовуються при оцінці (2.20) та (2.22), що, у свою чергу оновлює вхідні дані для розрахунку контрольних вузлів (2.23) наступного рівня. Тоді алгоритм складається з двох вбудованих ітерацій: внутрішньої, що чергує горизонтальний і вертикальний крок, і зовнішньої ітерації груп  $M_b$  перевірочних вузлів/рядків.

Розглянемо реалізацію декодера з фіксованою точкою. Для потенційного використання декодера LDPC в обмежених системах, особливо з урахуванням економії пам'яті, ми також реалізували арифметичний декодер

з фіксованою точкою, що за замовчуванням використовує цілий тип даних `int16_t` точної ширини, представлений в C99 для метрик алгоритму мінімальної суми. Зазвичай, але не обов'язково, це компілюється у `short int`. Оскільки алгоритм декодування використовує лише операції мінімуму, плюсу та мінусу, арифметику з фіксованою комою можна додатково спростити, просто відображення значень з плаваючою комою в цілий діапазон. Наш вибір 16-бітного уявлення набагато ширший, ніж різні 8-бітові і навіть менш бітні, що використовуються в декодерах, описаних у літературі, але реалізація повністю параметризована, тому теоретично легко переключитися на вужче 8-бітве уявлення.

Оскільки перевіряти переповнення щоразу при додаванні було б витратно, основний спосіб, який використовується в нашій реалізації, — використовувати ширше сховище, ніж потрібно. У разі 16-бітного типу C `short int16_t` значення LLR з плаваючою комою фактично вибираються як цілі числа бітів  $Q_B$  зі значенням за замовчуванням  $Q_B = 10$ . З додатковим бітом знака це ефективно визначає 11-бітний знаковий діапазон від -1023 до 1023. Ці значення зберігаються у 16-бітних цілих числах, залишаючи 5 старших бітів вільними, тому коли значення накопичуються під час роботи алгоритму мінімальної суми, випадки переповнення будуть досить рідкісними і не катастрофічно впливатимуть на підсумковий BER. Логічно очікується, що цей негативний ефект буде сильнішим зі збільшенням числа елементів підсумовування, тому коди з меншою швидкістю повинні піддаватися більшому впливу. Це підтверджується у розділі результатів.

Для підтримки архітектур з обмеженою пам'яттю можна визначити три різні конфігурації з різними вимогами до пам'яті: Конфігурація А - повністю універсальна реалізація без обмежень пам'яті, буде корисною для моделювання MATLAB. Він підтримує всі параметри коду LDPC, визначені стандартом, за допомогою кодера масиву та декодера з плаваючою комою. Конфігурація В представляє реалізацію з обмеженнями пам'яті: кодер растрових зображень та декодер з фіксованою точкою. Він підтримує лише

підмножину параметрів сумісного коду (N, K), а точніше половину стандартних параметрів коду WiMAX. Конфігурація C буде реалізацією з мінімальним обсягом пам'яті і підтримуватиме лише один із параметрів коду LDPC:  $N = 576$  і  $R = 1/2$ .

Хоча написання всієї реалізації моделювання мовою C можливе і забезпечує дуже високу продуктивність, гнучкість MATLAB як інструмента для швидкого прототипування та тестування потенційних модифікацій існуючих алгоритмів робить його незамінною платформою. З усіх реалізацій, жодна не забезпечує як реалізацію мову C з її продуктивністю, і інтеграцію з MATLAB. Однак іноді користувач може вибрати один із двох. Наша реалізація заповнює цю нішу, оскільки вона реалізує кодер і декодер C, які можна використовувати в реальній системі зв'язку, а також дозволяє запускати їх без змін з середовища MATLAB. Щоб досягти цього, окремі вихідні файли C реалізують необхідні оболонки MEX, які дозволяють викликати функції кодера та декодера з MATLAB.

Як для кодера, так і для декодера необхідні буфери визначаються як статично виділені масиви, а важливі параметри коду записуються як константи (макриси препроцесора) в автоматично створюваному заголовному файлі `ldpc.h`.

## 2.6 Аналіз пропускної спроможності

Ми оцінили можливості виправлення помилок нашої реалізації для кодів QC-LDPC, які використовуються у стандартах зв'язку IEEE 802.11ax та IEEE 802.16-2017, використовуючи класичний підхід моделювання Монте-Карло, створюючи криву водоспаду для каналу AWGN. Весь комунікаційний ланцюжок: кодер LDPC, канал і декодер LDPC була оцінена разом з комерційною реалізацією LDPC у наборі засобів зв'язку, представленої в останній версії MATLAB R2021b.

Як і очікувалося, декодер із фіксованою комою з обмеженням пам'яті (конфігурація В) вносить збільшення приблизно 0,1 дБ. Цей ефект найбільш помітний для великих значень  $N$  та менших швидкостей коду, де рівняння (2.22) підсумовує за великою кількістю рядків перевірконої матриці. Для менших  $N = 576$  і більш високих  $R = 5/6$  крива декодера з фіксованою комою стає ще ближчою до реалізації з плаваючою комою. У ході всіх симуляцій для водопадної кривої було зібрано щонайменше 10 000 помилок.

Оскільки декодер LDPC є найбільш ресурсомістким завданням у нашому моделюванні, оцінка пропускної спроможності була зосереджена на функції декодера. Було оцінено кілька різних одно- і багатопотокових конфігурацій декодера та проведено порівняння пропускної здатності з оптимізованою реалізацією за допомогою функції `ldpcDecode()`. Порівнювалися дві платформи x86-64: MATLAB R2021b на Ubuntu 18.04LTS, що працює на 8-ядерному/16-потоківому процесорі Intel Core i7-9800X з тактовою частотою 3,80 ГГц з архітектурою Skylake-X, що підтримує розширення та MATLABR2022a на Ubuntu 20.04. LTS працює на 16-ядерному/32-потоківому процесорі AMD Ryzen 9 5950X.

Розглянемо пропускну здатність декодера для коду WIMAX QC-LDPC зі швидкістю  $R = 5/6$  та розміром кодового слова  $N = 2304$ . Дані оброблялися блоками по 10 кодових слів на потік, а декодер був завжди налаштований на виконання фіксованої кількості ітерацій 10, щоб запобігти порівнянню часу виконання з різними числами ітерацій. 32-бітна реалізація з плаваючою комою і 16-бітна реалізація з фіксованою комою (ціле число) були оцінені разом з двома різними підходами багатопоточності: перший, більш простий, реалізація породжує робочі потоки щоразу, коли функція MEX запускається MATLAB, і знищує їх відразу після декодування блоку. Більш складний метод, позначений як MTX, запускає робочі потоки один раз, а потім синхронізує їх щоразу, коли викликається функція MEX декодера.

Одно потокова реалізація охоплює лише близько 60% еквівалентної в іншому функції Communications Toolbox. Це не дивно, враховуючи той факт, що декодер LDPC у MATLAB тепер є зрілою оптимізованою реалізацією, яка доступна лише у вигляді MEX-файлу із закритим початковим кодом.

Оскільки сучасні процесори давно використовують багато ядер, саме многопоточная реалізація насправді має значення. Тут перевага нашого підходу полягає в можливості точного настроювання кількості потоків, які користувач може вказати явно, порівняно з налаштуванням «Увімк./Вимк.» у функції панелі інструментів. Низька пропускна здатність багатопоточного декодера Toolbox дещо дивовижна і може вказувати на помилку в реалізації, яка потенційно буде виправлена в деяких пізніших випусках Toolbox. Фактичне використання кількох потоків перевірялося вбудованою в ОС утилітою системного монітора.

Що трохи розчаровує, так це майже незначне покращення пропускної спроможності складнішої конструкції MTX, де робочі потоки працюють (або чекають) паралельно з основним потоком MATLAB і синхронізуються з основним потоком за допомогою умовних змінних.

## **Висновки до другого розділу**

У розділі докладно описано реалізація кодера QC-LDPC та універсального декодера LDPC C99, які орієнтовані на сучасні коди LDPC, визначені у стандартах IEEE 802.11-2020 та IEEE 802.16-20. Ми надали результати оцінки продуктивності помилок, порівняні з сучасною реалізацією із закритим початковим кодом, що надається MATLAB R2021b Communications Toolbox, а також порівняння пропускної спроможності різних конфігурацій на платформах Intel та AMD.

## 3 ДЕКОДЕР QC-LDPC З МАЛОЮ ЗАТРИМКОЮ

### 3.1 Квазіциклічні коди LDPC

Коди QC-LDPC набули широкого поширення для досягнення чудових характеристик виправлення помилок у багатьох системах бездротового зв'язку, таких як Wi-Fi в IEEE 802.11n/ac/ax, Wi-Max (IEEE 802.16e), 5G NR(3GPP TS 38) та DVB-S2. Для квазіциклічних кодів QC-LDPC використовуються переважно два типи декодерів. Стандартна передача повідомлень складається із двох етапів і відома як графік лавинного розсилання (FS) або алгоритм суми добутоків. На першому етапі всі повідомлення від вузлів змінних (VN) до вузлів перевірки (CN) оновлюються, а потім у другому етапі обчислюються всі вузли перевірки (CN). Галлагер представив цей метод декодування в 1962 році, а Маккей використовував його при повторному відкритті кодів LDPC. У багаторівневому розкладі (LS) матриця перевірки парності вважається такою, що складається з послідовності горизонтальних або вертикальних шарів. Цей принцип дозволяє позбутися FS підвищення швидкості збіжності з допомогою негайного обміну інформацією між рівнями. І цей коефіцієнт становить приблизно два при високому відношенні сигнал/шум (SNR) [9]. Це вигідно для тих стандартних пропозицій, де швидкість каналу декодування каналу низька (наприклад, 4 мкс або менше для IEEE 802.11ax). Горизонтально-рівневі декодери (LD) стали домінуючою формою LD через меншу кількість ресурсів пам'яті та меншу кількість тактових циклів на ітерацію [10] [11]. Однак конвеєрна обробка LD передбачає передачу повідомлень взаємозалежності між сусідніми рівнями через час очікування, необхідного для отримання оновлених повідомлень від попереднього рівня перед початком обробки наступного рівня, особливо високого паралелізму. Взаємозалежність призводить до виникнення циклів зупинки, збільшення затримки та зниження пропускної спроможності [3].

Ступінь паралелізму декодерів QC-LDPC можна розділити на три категорії: послідовний, частково паралельний та повністю паралельний. Послідовний означає, що  $VN$ , і  $CN$  мають ступінь паралелізму, що дорівнює одиниці. Еквівалентність повністю паралельних  $VN$  дорівнює довжині коду. Паралелізм  $CN$  - це довжина бітів парності. Через низьку пропускну здатність послідовних декодерів, дослідження зосереджені на паралелізмі.

Часткова рівність має блокову серіалізацію, що означає, що парність  $VN$  і  $CN$  дорівнює розміру коефіцієнта підйому  $Z$ . Паралелізм рядків передбачає, що однаковість  $VN$  аналогічна довжині коду, еквівалентність  $CN$  - це розмір  $Z$ . Повністю паралельні  $VN$  мають паралелізм, рівний довжині коду, а  $CN$  мають рівність, що дорівнює довжині бітів парності.

Оскільки повністю паралельна архітектура (FPA) не застосовується до LS, її зазвичай поєднують з FS для виконання ітерації за один такт. Рядно-паралельна архітектура (RPA) та блочно-послідовна архітектура (BSA) використовують LS, де RPA стикається з більшою залежністю від даних.

Для RPA у багатьох дослідженнях використовується поперемінне багатоканальне декодування у поєднанні з технологією міжрівневого злиття для вирішення проблеми залежності даних. Однак багатоканальне чергування призведе до збільшення ресурсів зберігання. Технологія міжрівневого злиття використовується для зменшення глибини конвеєрів для низької швидкості кодування, але не підходить для протоколів WLAN. Через код QC-LDPC в протоколах вага перших декількох стовпців дорівнює кількості шарів. Висока складність апаратного забезпечення та перевантаження маршрутизації FPA та RPA займатимуть надто велику площу чіпа [8]. BSA може переставляти рядки та стовпці матриць перевірки парності (PCM) в автономному режимі, щоб зменшити або усунути період затримки [1]. Петрович використовує як LS, і FS, коли виникають небезпеки, але уповільнює швидкість збіжності [2].

У цій статті пропонується двоколіїний декодер послідовної обробки з малою затримкою, який прагне мінімізувати затримку при обмеженій складності обладнання та є найкращим вибором для невеликих площ

кристалів. Декодер забезпечує двоблоковий паралелізм за допомогою алгоритмів 1 і 2, що знижує затримку майже вдвічі при високих швидкостях кодування порівняно з BSA. По-перше, алгоритм 1 перетворює всі PCM у два шляхи. Обидва шляхи можуть одночасно отримати доступ до невід’ємних елементів у прототипах PCM. Алгоритм 2 є двоколіїним застосуванням алгоритму нормалізованої мінімальної суми (NMS), що використовує певний порядок доступу для подолання залежності LS від даних.

Решта статті організована так: У розділі II наводяться загальні відомості про графіку та архітектуру декодування. У розділі III представлена стратегія двоколіїного декодування з малою затримкою. Відповідна апаратна архітектура обговорюється у розділі IV. У розділі V наведено результати синтезу та показано порівняння базових робіт. Нарешті розділ VI завершує статтю.

Двійковий код LDPC, що задовольняє виразу:

$$H \cdot x^T = 0, \forall x \in C. \quad (3.1)$$

поле Галуа у (1)  $H$  являє собою  $M \times N$  розріджену двійкову матрицю, яка називається матрицею перевірки парності. Вектор  $x$  розміру  $1 \times N$  у наборі допустимих кодових слів  $C$ . Коди LDPC також можуть бути наочно представлені графом Таннера [2], як показано на рис. 1.

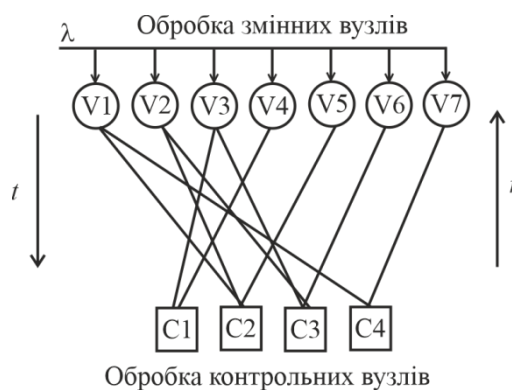


Рисунок 3.1 – Граф Танера LDPC коду

Граф Таннера містить вузли змінних (VN) та вузли перевірки (CN), які пов'язані. Коди QC-LDPC – це різновид кодів LDPC, які можуть бути створені за допомогою прототипів  $M_p \times N_p$  матриці  $H_p$ . Кожен елемент  $H_p$  являє собою квадрат підблоків  $P^l$ . Матриця циклічних перестановок  $P^l$  виходить із одиничної матриці  $Z \times Z$  шляхом циклічного зсуву стовпців праворуч на  $l$  елементів, причому  $P^l = 0_{Z \times Z}$  [1]. Величину  $Z$  називають коефіцієнтом підйому (LF), або коефіцієнтом розширення.

Коротко графік лавинного розсилання (FS) можна охарактеризувати як підвищення надійності повідомлень за допомогою передачі повідомлень між змінними (VN) і перевірочними (CN) вузлами. Початкові повідомлення називаються повідомленнями логарифмічного відношення правдоподібності (LLR).  $\lambda$  спочатку є початковими відомостями про VN, переданих CN (V2C). Тобто передача VN  $j$  передається до CN  $i$ ,  $t_{j2i} = \lambda_j$ . Крім того, CN завершують оновлення повідомлення CN для VN (C2V), таким чином:

$$|r_{i2j}| = \psi^{-1} \sum_{j' \in J_i \setminus j} \psi(|t_{j'2i}|) \quad (3.2)$$

$$\text{sign}(r_{i2j}) = \prod \text{sign}(r_{j'2i}) \quad (3.3)$$

де  $J_i$  - набір VN, підключених до CN  $i$ , та

$$\psi(x) = -\lg \left( \text{tgh} \left( \frac{x}{2} \right) \right) \quad (3.4)$$

На  $(k + 1)$ -й ітерації VN  $j$  отримує повідомлення  $r^k$  від підключених CN і апостеріорну ймовірність  $(k - 1)$ -ї ітерації  $q_{post}^{k-1}$  потім оновлює повідомлення  $t_{j2i}^{k+1}$  на основі (3.5). Оцінку бітів  $k$ -ої ітерації можна згідно виразів (3.6).

$$t_{j2i}^{k+1} = q_{j,post}^{k-1} + \sum_{i' \in I_j \setminus i} r_{i'2j}^k \quad (3.5)$$

$$q_{j,post}^k = q_{j,post}^{k-1} + \sum_{i' \in I_j} r_{i'2j}^k \quad (3.6)$$

де  $I_j$  - множина CN, підключених до VN  $j$ , і (3.7) (3.8) можна отримати, об'єднавши (3.5), (3.6)

$$t_{j2i}^{k+1} = q_{j,post}^{k-1} - r_{i2j}^k \quad (3.7)$$

$$q_{j,post}^k = t_{j2i}^{k+1} + r_{i2j}^k \quad (3.8)$$

Тобто V2C може бути отриманий за допомогою апостеріорної ймовірності кодового слова та повідомлення CN де  $q_{j,post}^0 = \lambda_j$ ,  $r_{i2j}^k = 0$ .

Багаторівневий графік (LS)  $q_{j,post}^k$  завершує оновлення на кожному рівні. Через більшу кількість оновлень в одній ітерації  $q_{j,post}^k$  швидкість збіжності приблизно вдвічі вище, ніж у графіка лавинного розсилання (FS). Для квазіциклічних структур на кожному рівні VN з'єднаний лише з одним CN, і якщо поточний шар дорівнює  $m$ , то можна записати:

$$t_{j2i}^{k+1}[m] = q_{j,post}^k[m-1] - r_{i2j}^k \quad (3.9)$$

$$\left| r_{j2i}^{k+1} \right| = \psi^{-1} \sum_{j' \in J_i \setminus j} \psi \left( \left| q_{j,post}^k[m-1] - r_{i2j'}^k \right| \right) \quad (3.10)$$

$$q_{j,post}^k[m] = q_{j,post}^k[m-1] + r_{i2j}^{k+1} - r_{i2j}^k \quad (3.11)$$

У реальному прототипі матриці перевірки на парність (PCM), якщо на рівні є негативні числа, тобто результат являє собою повну нульову матрицю, а  $q_{j,post}^k[m]$  відповідного стовпця рівня не буде оновлюватися.  $q_{j,post}^k[M]$  - кінцевий результат  $(k + 1)$ -ої ітерації після завершення оновлення рівня.

$$q_{j,post}^{k+1} = q_{j,post}^{k+1}[0] - q_{j,post}^k[M] \quad (3.12)$$

Оскільки функція  $\psi(x)$  надто складна для апаратної реалізації, зазвичай застосовується покращена апроксимація алгоритму мінімальної суми, алгоритму мінімальної суми зі зміщенням (OMS) та алгоритму нормалізованої мінімальної суми (NMS), як показано у (3.13) та (3.14):

$$|r_{i2j}^{k+1}| = \max\left(\min_{j' \in J_i \setminus j} (q_{j',post}^k[m-1] - r_{i2j'}^k) - \beta, 0\right) \quad (3.13)$$

$$|r_{i2j}^{k+1}| = \alpha \cdot \min_{j' \in J_i \setminus j} (q_{j',post}^k[m-1] - r_{i2j'}^k), \quad (3.14)$$

де  $\beta > 0$  та  $0 < \alpha < 1$ . 0,75 є відповідним значенням для  $\alpha$ .

### 3.2 Архітектура декодера квазіциклічних кодів

Розглянемо повністю паралельну архітектуру. Як показано на рисунку 3.2 для FPA паралелізм процесора змінних вузлів (VNP) дорівнює довжині кодового слова, а паралелізм процесора контрольних вузлів (CNP) аналогічний числу CN. Отже, кількість обробних елементів (PE) пропорційно довжині коду і обернено пропорційно швидкості коду.

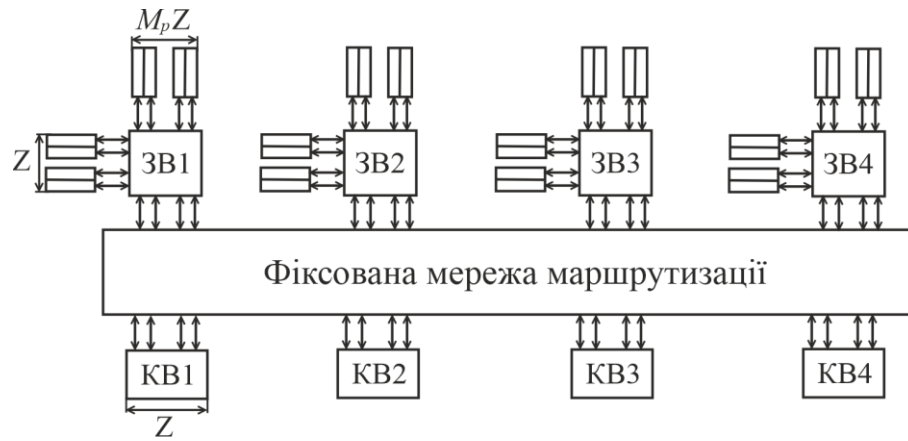


Рисунок 3.2 - Класична повністю паралельна архітектура  
(ЗВ – змінний вузол; КВ – контрольний вузол)

Для серії протоколів IEEE 802.11n/ac/ax, коли довжина коду дорівнює 1944, а швидкість коду дорівнює  $1/2$ , кількість блоків обробки змінних вузлів (VNPU) дорівнює 1944. При цьому кількість блоків обробки контрольних вузлів (CNPU) становить 972. -за надмірної кількості з'єднувальних проводів повністю паралельна структура не підходить для сумісності по довжині коду або швидкості коду. В іншому випадку це призведе до навантаження маршрутизації. FPA використовує паралелізм, виконує одну ітерацію за такт і зазвичай досягає максимальної пропускної спроможності.

Розглянемо рядково - паралельну архітектуру декодера квазіциклічних кодів. Як показано на рисунку 3.3, РСМ розділений на кілька горизонтальних шарів залежно від розміру  $LF$ , а паралельно рядкова архітектура завершує обробку горизонтального рівня за один такт.

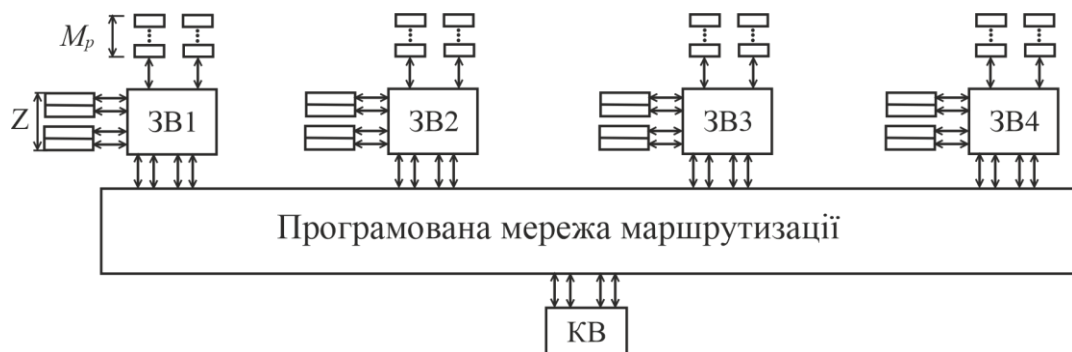


Рисунок 3.3 - Класична архітектура з паралельними рядками

Паралелізм VNP дорівнює довжині кодового слова, а паралелізм CNP – розміру LF. Кожен CNPU з'єднує 24 віртуальні мережі через мережу маршрутизації, що налаштовується.

Розглянемо блочно-послідовну паралельну архітектуру. Як показано на рисунку 3.4, паралелізм VNP та CNP BSA відповідає розміру LF. VNP і CNP обходять невід'ємні елементи РСМ прототипу, або ненульові підматриці.

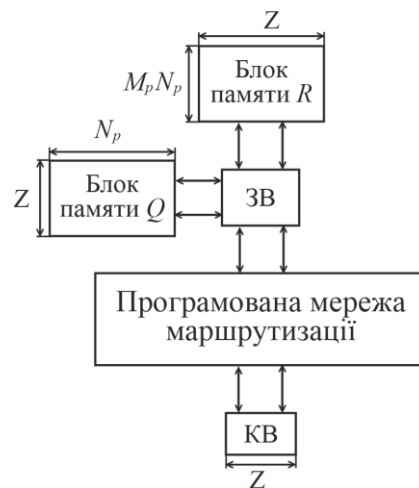


Рисунок 3.4 – Класична блочна послідовно-паралельна архітектура

При обробці одного блоку кожному CNPU необхідно підключити лише один вузол змінної через мережу маршрутизації. BSA зазвичай мінімізує сферу реалізації коду QC-LDPC.

### 3.3 Залежність даних багаторівневого декодера

Для простоти візьмемо за приклад останні п'ять стовпців прототипу матриці довжиною 648 і швидкістю 5/6, які показані на рисунку 3.5, а порядок доступу до її невід'ємних примітив підматриці показаний на рисунку 3.6.

	1	2	3	4	5
1	13	1	0	-1	-1
2	2	-1	0	0	-1
3	15	0	-1	0	0
4	14	1	-1	-1	0

Рисунок 3.5 – Прототип матриці для блокової послідовної архітектури

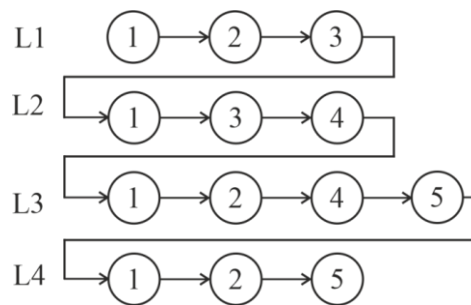


Рисунок 3.6 – Спрощений порядок доступу для блокової послідовної архітектури

Зазвичай використовують двофазне перекриття передачі повідомлень (TRMP) для реалізації конвеєра.

Тобто обробка оновлення повідомлення VN на рівні та читання м'яких повідомлень кодового слова також називаються фазою читання (RP), а оновлення CN та зворотний запис повідомлення апостеріорної ймовірності також відомі як фаза запису (WP). ). Оскільки наступний рівень повинен чекати на оновлене попереднього рівня, генерується період очікування.

Отримані з нього цикли простою називаються довжиною очікування (WL), та його значення суворо пов'язані з апаратної реалізацією. Більше того, кількість допустимих елементів для послідовних шарів може бути неоднаковою, що призводить до утворення бульбашок конвеєра, які ми можемо назвати довжиною заповнення (PL).

Бульбашки у трубопроводі, викликані циклом зупинки, ефективно зменшуються після зміни порядку колон. Якщо RP працює на (i+1)-му рівні, WP відповідно працює на i-му рівні. Спеціальний метод сортування дозволяє

RP розставляти пріоритети для повідомлень апостеріорної ймовірності, які були оновлені, залишаючи WP більше часових інтервалів обробки. Тобто блоки повідомлень, які були оновлені на  $(i-2)$ -му або верхньому рівні, тобто на рівні  $(i-1)$  і я не оновлюю ці блоки, мають вищий пріоритет і записуються як R1. . Якщо рівень  $(i-1)$  оновлює ці блоки (рівень  $i$  не оновлює ці блоки), то блоки мають вищий пріоритет, записані як R2, а блоки, які очікують на оновлення рівня  $i$ , записуються як R3. Колір заливки квадрата відрізняє пріоритети RP, а остаточний порядок обробки - [R1 R2 R3]. Такий самий пріоритет  $i$  в порядку стовпців від меншого до більшого. Аналогічно, WP записує терміново необхідне м'яке повідомлення від RP рівня  $(i+1)$  як W1, для  $(i+2)$ -го рівня як W2 та інших як W3. Кольорові кружки позначають пріоритети WP, а остаточний порядок обробки [W1 W2 W3].

Стратегії апроксимації, які ставлять під загрозу продуктивність декодування та зайнятість області, є розумними рішеннями для багатьох досліджень, спрямованих на подолання проблеми надмірних витрат ресурсів. Наприклад, Ченг набуває більш значущого значення останнього компаратора дерева як субхвилинного значення [2]. Тобто максимальна ймовірність субхвилини використовується замість фактичного значення субхвилини. При цьому декодер працює на частоті 1996 МГц та забезпечує пропускну здатність 45420 Мбіт/с. RPA може використовувати або FS або LS. Якщо вибрано FS, для завершення конвеєра проекту потрібні два кадри почергового декодування. В іншому випадку, якщо архітектура поєднується з LS, їй необхідно передавати кадри глибини, щоб уникнути залежності даних. Лі використовує піврядкову структуру, зменшуючи залежність даних та кількість кадрів, зберігаючи при цьому переваги LS та уникаючи подальшого розширення сховища [1]. BSA зазвичай супроводжує LS, оскільки перекриття TRMP може використовуватися для конвеєризації. Більше того, автономне перегрупування PCM може ще більше подолати залежність від даних [2]. Використання механізму обходу дозволяє скоротити час декодування, а й заощадити енергоспоживання. З іншого боку, існує паралельна структура

кодового слова [3]. Використовуючи сегментований циклічний зсувник, що реконфігурується, [4] може одночасно декодувати кілька кодових слів (або вхідних даних) різної довжини. Ця структура необхідна для запобігання простою апаратних ресурсів та досягнення вищої середньої пропускну здатності для користувачів з низьким трафіком.

### 3.4 Алгоритм декодування з низькою затримкою

Пропонований двоколіїний алгоритм декодування забезпечує компроміс між складністю обладнання та затримкою обробки з використанням двоблочного паралелізму. Три шляхи прийнятні для високих швидкостей коду, таких як 5/6. Стратегію подвійного шляху можна поділити на дві частини. Перша частина полягає в отриманні кращих прототипів підматриці  $H_{p1}$ ,  $H_{p2}$  і порядку доступу, а саме алгоритму 1. Друга частина полягає в декодуванні відповідно до зазначеного порядку, а саме за алгоритмом 2.

Коди IEEE 802.11n/ac/ax LDPC призначені для трьох довжин кодового слова:  $L_0$  648,  $L_1$  1296 і  $L_2$  1944, відповідають  $Z_0$  1, 2  $Z_1$  2 2. Кожен розмір має чотири кодові швидкості:  $r_0$  1/2,  $r_1$  2/3,  $r_2$  3/4 і  $r_3$  5/6, всього 12 кодів.

Взявши за приклад довжину коду 648 і швидкість коду 5/6, компанія Aiming розділила прототип РСМ на дві групи для виконання паралельної обробки. Спостерігаючи за структурою прототипу, бачимо, що невід'ємні елементи заповнюють перші п'ять стовпців, а ненульові елементи останніх п'яти стовпців рідкісні. Тому необхідно змінити порядок стовпців, щоб зробити ненульові об'єкти двох груп якомога більш рівними, щоб забезпечити максимально узгоджену затримку обробки. Проходячи компонування, останній замовлений стовпець утворює новий прототип РСМ, перші 12 стовпців передаються РАТН 1, записуються як  $H_{p1\_c}$ , а останні 12 стовпців передаються в РАТН 2, позначаються як  $H_{p2\_c}$ . Детальна процедура

відповідає КРОКУ 1 алгоритму 1, де  $M$  являє собою кількість рядків прототипу РСМ,  $i$  являє собою набір з перших п'яти та останніх п'яти стовпців, які необхідно переставити.

Результати перестановки становлять  $c\_order$ , а  $c\_size$  — кількість рядків  $c\_order$ . Ітерація по  $c\_order$  повертає мінімальну різницю невід'ємних елементів двоколієних підпрототипів  $num\_min$ , а відповідне сортування стовпця позначається як  $Cbest$ . Перші 12 стовпців матриці прототипу РСМ у цьому порядку складаються з підматриці  $Hr1\_c$ , а останні 12 стовпців утворюють ще одну підматрицю  $Hr2\_c$ .

Як тільки  $Hr1\_c$  і  $Hr2\_c$  зафіксовані, можна вивчити кращий порядок доступу до стовпця, а також мінімальну кількість тактів на ітерацію TL. Детальна процедура відповідає КРОКУ 2 Алгоритму 1.

Перерахуйте  $Hr1\_c$  та  $Hr2\_c$  з фіксованим порядком стовпців  $Cbest$  та випадковим ієрархічним порядком як  $Hr1\_t$  та  $Hr2\_t$ . Подібно до методу, згаданого в П.Д, ми ділимо період на дві основні стадії: RP і WP. Для PATH1 певний порядок доступу до стовпця RP –  $read\_seq1$ , WP –  $writ\_seq1$ , а відповідні порядки –  $read\_seq2$  та  $writ\_seq2$  для PATH2. Візьміть PATH1 як приклад; ми можемо визначити  $R1\_1$  як рівень пріоритету 1 для читання для RP,  $R1\_2$  як рівень 2 і  $R1\_3$  як рівень 3.  $W1\_1$ ,  $W1\_2$  і  $W1\_3$  використовуються для WP.

Відповідними елементами PATH2 є відповідно  $R2\_1$ ,  $R2\_2$ ,  $R2\_3$ ,  $W2\_1$ ,  $W2\_2$  та  $W2\_3$ . Пройдіться по всіх шарах двох підпрототипів, щоб отримати оновлені  $read\_seq1$ ,  $read\_seq2$ ,  $writ\_seq1$  та  $writ\_seq2$ . Потім обчисліть відповідні розміри WL для  $Hr1\_t$  і  $Hr2\_t$ ,  $WL1$  і  $WL2$ , щоб додатково обчислити тактовий цикл  $len\_layer1$  і  $len\_layer2$ , необхідний кожної ітерації рівня. Оскільки затримка обробки кожного рівня двох підпрототипів повинна бути узгодженою, ми приймаємо більш значуще значення як фактичну затримку для кожного рівня, що позначається як  $len\_layer$ . Підсумковий TL – це накопичення значень кожного шару  $len\_layer$ .

Враховуючи залежність LS від даних, ми досліджували оптимальне розташування шарів вивчення схеми мінімальної затримки, а результати  $H_{p1\_c}$  і  $H_{p2\_c}$  переоцінені як  $H_{p1}$  і  $H_{p2}$ . Детальна процедура відповідає КРОКУ 3 алгоритму 1. Перестановка результатів шарів складає  $r\_order$ , а  $r\_size$  - це кількість рядків  $r\_order$ . Ми повторюємо  $r\_order$  для оновлення підпрототипів і повторюємо КРОК 2 щоб отримати мінімальний TL і позначити його як BL. Нарешті, отримайте оптимальний порядок рядків  $R_{best}$ , порядок доступу  $read\_seq1$ ,  $read\_seq2$ ,  $writ\_seq1$ ,  $writ\_seq2$  та прототипи підматриць  $H_{p1}$  і  $H_{p2}$ .

Після отримання розташованих  $H_{p1}$  і  $H_{p2}$  алгоритм декодування може бути реалізований як алгоритм 2. Алгоритм 2 є застосуванням багаторівневого алгоритму мінімальної суми з двома шляхами.

По-перше, потрібна ініціалізація, оновлення повідомлення CN  $r1$  і  $r2$  встановлюються в нулі, а м'яка ітеративна інформація  $q1$  і  $q2$  повинна бути вихідними LLR  $\lambda$ . Оскільки ми змінили порядок стовпців на  $C_{best}$ , відповідну інформацію LLR також налаштують на  $C_{best}$ , та призначити  $q1$  та  $q2$ .

Після початку ітерації  $k$  представляє кількість ітерацій,  $m$  представляє поточний рівень, а  $RATH1$  та  $RATH2$  обробляються паралельно RP. Мінімальне значення  $m1$  і позначення  $m2$  повідомлення  $t$ , що передається в CN, ініціалізуються нескінченністю, а накопичувальний знак  $s$  ініціалізується 1. Взявши в якості прикладу  $RATH1$ , спочатку дотримуйтесь порядку доступу до стовпця  $read\_req1$ , отриманого за допомогою алгоритму 1, витягніть по черзі  $H_{p1}$  і наведіть значення  $s1$ . Після завершення циклічного зсуву обчисліть  $V2C\ t1$  згідно з (3.9). Пройшовши всі шари  $H_{p1}$ , ми можемо отримати мінімальні значення абсолютного розміру  $t1$ , званого  $m11$ , позначення  $m12$  і накопичувального символу  $s1$ . Те саме стосується і  $RATH2$ , щоб отримати  $m21$ ,  $m22$  і знак накопичення  $s2$ .

WR  $RATH1$  і  $RATH2$  також обробляються паралельно для завершення оновлення повідомлення CN і оновлення апостеріорної ймовірності цього кодового слова.  $m$  — фактичне мінімальне абсолютне значення, яке можна

вибрати відповідно до індексу поточного елемента  $n1$  та  $idx$ . Якщо значення узгоджені,  $m$  дорівнює  $m2$  та навпаки  $m1$ . Після того, як коефіцієнт коригує  $m$ , це останнє повідомлення  $r1$ , що передається  $VN$ , відповідне (3.14).  $q1$  - апостеріорна ймовірність зворотного запису, і варто відзначити, що порядок зворотного запису необхідно змінити на зворотний.

Після досягнення максимальної кількості ітерацій  $K$  приймається складне рішення. Біти знака апостеріорної ймовірності кодового слова  $q$  становлять результат, і вихідні дані декодування перебудовуються відповідно до  $C_{inver}$ , який є  $C_{best}$  зворотного порядку.

### 3.5 Архітектура декодера з мінімальною затримкою

Апаратна архітектура з мінімальною затримкою і подвійним шляхом для декодера QC-LDPC розроблена на основі алгоритму 2. Декодер містить обчислювач параметрів, препроцесор, ядро та постпроцесор (рис. 3.7).

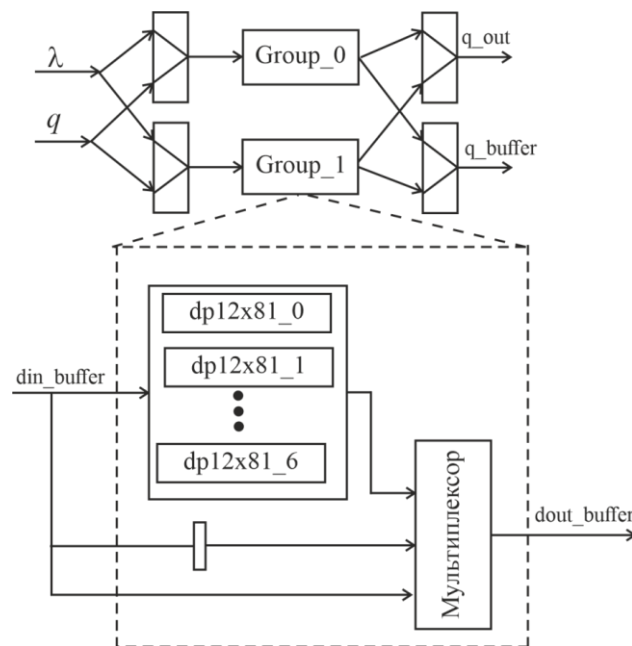


Рисунок 3.7 – Блок-схема буфера Q1

Препроцесор перетворює отримані повідомлення LLR на реальні кодові слова. Сто дванадцять біт відносяться до програмної вхідної інформації за цикл із розміром квантування 7. Базовий модуль перетворює повні кодові слова на інформаційні біти. Постпроцесор видаляє укорочені біти та забезпечує правильну ширину системи зв'язку.

Ми можемо зосередитися на основному модулі, який складається з буферів, циклічних зрушень (CS), N-LUT (таблиця пошуку  $N_{p1}$  та  $N_{p2}$ ), Т-калькулятора (TC), процесора (PU), жорсткого рішення (HD), та Глобальний контролер (GC). Вхідна інформація LLR спочатку зберігається в буфері Q, а буфер Q також використовується як пам'ять оновлення повідомлень LLR, у той час як вихідний сигнал буфера R встановлюється в 0 для першої ітерації. Враховуючи конвеєрну обробку безперервних кодових слів, буфер Q встановлюється на ті самі дві групи розмірів. Глибина Q-буфера дорівнює 12, а ширина – 567 ( $z_2$  - Квантування).

Щоб скоротити етапи конвеєра, тобто зменшити WL, буфер Q використовує стратегію обходу, конкретну структуру показано на рис. 16. Вихідний  $q\_buffer$  може бути оновлений у тому ж чи наступному тактовому циклі з вхідним  $q$ . Глибина буфера R дорівнює 45, а розрядність - 405.

При цьому враховується, що максимальна кількість невід'ємних елементів  $N_{p1}$  і  $N_{p2}$  дорівнює 45. Буфер R зберігає тільки повідомлення оновлення CN цих елементів для економії області.

Оскільки розрядність квантування повідомлення контрольного вузла дорівнює 5 розрядність дорівнює 405.

Процес ітеративного декодування починається, коли повне кодове слово зберігається у Q-буфері. Перший етап – розрахунок оновленої інформації ВН. На основі запропонованого алгоритму декодування витягніть інформацію LLR, що зберігається в буфері Q, і відніміть повідомлення оновлення CN останньої ітерації в буфері R, щоб оновити значення повідомлень. TC завершує цей крок. Значення  $t$  будуть відправлені одночасно в буфер T і модуль PU. Довжина T дорівнює 12, що відповідає кількості

блоків, а розрядність відповідає буферу Q. Варто зазначити, що LLR в буфері Q необхідно циклічно зрушувати, щоб побудувати реальний зв'язок між VN і CN, як описано в Нр1 і Нр2. Значення циклічного зсуву праворуч зберігаються в модулі H-LUT і виконуються CS.

На другому етапі обчислюється інформація оновлення CN та інформація оновлення LLR, які переважно обробляються модулем PE. Щоб відповідати порядку, що зберігається в буфері Q, повідомленням LLR блоку потрібно циклічне зсув, і цю функцію виконує модуль CS2.

Третій етап – вивести результат жорсткого рішення на основі даних LLR, що зберігаються в буфері Q. Коли кількість ітерацій перевищує максимальну, модуль жорсткого рішення видає розробку та передає її `post_processor`. Якщо поточний кадр не закінчився, інша група буферів Q може продовжити декодування нового кодового слова цього кадру.

RTL-модель декодера розроблена на Verilog HDL та синтезована за технологією SMIC 55 нм CMOS. Детальні результати реалізації показані в Таблиці I. Реалізація забезпечує тактову частоту 500 МГц і вимагає загальної площі 0,62 мм<sup>2</sup>, з яких більшу частину займають основний модуль та модуль препроцесора - 0,27 та 0,71 відповідно.

Для модуля ядра буфери і ПУ займають 54% і 27% відповідно, їх площа тісно пов'язана зі ступенем паралелізму. Що стосується кількості елементів зберігання (SE), якщо в конструкції RPA використовується чергування декількох кадрів, щоб уникнути конфліктів доступу, таких як [15], кількість необхідно подвоїти або більше. Ресурс зберігання BSA відповідає цьому дослідженню, оскільки суть подвійного шляху полягає у поділі вихідного сховища на дві частини. Що стосується PE, RPA приблизно в 12 разів більше, ніж у нинішній конструкції. Кількість НП у BSA складає половину цієї роботи. Припустимо, що кількість PE та SE фіксовано. І тут споживання конкретного логічного ресурсу лінійно пропорційно квантування, а повідомлення V2C і C2V у цій роботі мають квантовану розрядність (7, 5). Зменшення ширини квантованого біта може забезпечити

значну оптимізацію площі, тоді як, з іншого боку, продуктивність виправлення помилок також значно погіршиться. Продуктивність квантування розрядності (6,4) порівняно з (7,5) погіршується на 0,2 дБ.

Розглянемо затримку та пропускну здатність для всіх дванадцяти режимів запропонованої схеми на частоті 500 МГц, а також поліпшення ірв затримки обробки в порівнянні з традиційним BSA з переупорядкуванням матриці. Пропускна здатність інформаційного біта декодера  $\Theta$  розраховується в (15), де Pre являє собою затримку pre\_preprocessor. Оскільки Q буфер має структуру пінг-понгу, затримка модуля попередньої обробки дорівнює нулю без урахування затримки першого заповнення. Як згадувалося раніше, BL – мінімальна кількість тактів для однієї ітерації, K – максимальна кількість ітерацій, ll – довжина останнього шару останньої ітерації, Post – затримка Post\_preprocessor із фіксованим значенням. один. OL — це кількість циклів, необхідне однієї ітерації в [7], видно, що затримка значно знижується проти блочним послідовним підходом.

$$\Theta = \frac{L \cdot rate}{Pre + BL \cdot K + ll + Post} \cdot f_{clk} \quad (3.15)$$

Реалізація цієї статті порівнюється з іншими представленими роботами, як показано в Таблиці III. Оскільки інші декодери не включають попередню обробку, калькулятор параметрів та постобробку, враховується лише основний модуль. Для довідки ми вибираємо декодери з різним ступенем паралелізму, де [1] використовують BSA, [2] використовують RPA.

Пропускна здатність FPA є найпомітнішою, особливо [3] з досягненням 458 Гбіт/с, що також зумовлено його ітеративною структурою розгортання. Тим часом зайнятість його ресурсів також досягла максимуму. Нормалізовану пропускну здатність декодера (NDTE) введено для справедливого порівняння. При цьому враховуються складність апаратного забезпечення, ітерації та технологія процесу. PE FPA займають більшу

частину площі чіпа. Він підтримує лише певний режим кодування, фіксовану швидкість коду та довжину коду, що також має бути помітним у NDTE.

Однак NDTE є RPA, представлений в [9], для досягнення найкращих результатів через перевантаження маршрутизації обслуговування FPA.

Конструкція [7] жертвує своєю продуктивністю при NDTE, а середня пропускна здатність є видатною у випадку багатьох сценаріїв короткого коду. Двоколійна структура забезпечує найменше значення логічних елементів, а результат NDTE на 89% вище, ніж у [2], на 500% вище, ніж у [6].

### 3.6 LDPC декодер низької складності

Продуктивність LDPC кодів краща лише для великих довжин блоків. Великі довжини блоків призводять до великої матриці контролю парності і породжуючої матриці. Розріджену матрицю контролю парності можна подати у вигляді  $[P \ T \ I]$  через виключення Гауса. З цього породжуюча матриця може бути представлена у вигляді:  $G = [I \ P]$ . Матриця  $P$  не розріджена і, отже, складність кодування буде збільшуватись з збільшенням довжини блоку. Для зменшення складності можуть бути використані ітераційні алгоритми декодування.

Декодування кодів LDPC. Найбільш поширеним способом декодування LDPC-кодів є алгоритм поширення довіри. Він також називається алгоритмом передачі повідомлень чи алгоритмом суми продукту (SPA).

Враховуючи передане слово  $c = \{c_0, c_1, \dots, c_{n-1}\}$  і слово  $v = \{v_0, v_1, \dots, v_{n-1}\}$ , що приймається, необхідно обчислити логарифмічне відношення правдоподібності (LLR) прийнятого символу. LLR визначається виразом:

$$L(v_i) = \log \left( \frac{\Pr(v_i = 0 | y_i)}{\Pr(v_i = 1 | y_i)} \right) \quad (3.16)$$

Якщо сигнал вважається антиподним і канал з адитивним білим гаусовим шумом, LLR прийнятого символу каналу задається виразом:

$$L(v_i) = 2y_i / \sigma^2, \quad (3.17)$$

де  $y_i$  – прийнятий символ та  $\sigma^2$  – напруга шуму. Ці повідомлення, які передаються перевірочними вузлами, стають  $L(q_{ij})$  від змінного вузла  $i$  до перевірочного вузла  $j$ . знак і величина  $L(q_{ij}) \in \alpha_{ij}$  та  $\beta_{ij}$  відповідно.

На перевірочних вузлах ці повідомлення обробляються і передаються повідомлення від вузла перевірки до змінного вузла, що дається виразом:

$$L(r_{ij}) = \left[ \prod_{i' \in V_j \setminus i} a_{i'j} \right] \cdot \varphi \left( \sum_{i' \in V_j \setminus i} \varphi(\beta_{i'j}) \right) \quad (3.18)$$

$$\varphi(x) = \log \left( \frac{e^x + 1}{e^x - 1} \right) \quad (3.19)$$

Повідомлення  $L(r_{ij})$  не залежить від повідомлення, яке надійшло з  $i$ -ого перевірочного вузла. Змінний вузол обробляє повідомлення, отримане відповідно до заданого рівняння

$$L(q_{ij}) = L(V_i) + \sum_{j' \in C_i \setminus j} L(r_{ji'}) \quad (3.20)$$

Вище наведене рівняння показує, що повідомлення зі змінного вузла до перевірного вузла є простим підсумовуванням повідомлень, що надходять з

усіх вузлів, крім вузла перевірки  $j$ . Обробка повідомлень обома перевірочним вузлом і змінним вузлом процесором еквівалентно однієї ітерації SPA.

Символи декодуються шляхом порівняння  $L(Q_i)$  з порогом.

$$L(Q_i) = L(V_i) + \sum_{j \in C_i} L(r_{ji}) \quad (3.21)$$

Декодований символ задається виразом:

$$C_i = \begin{cases} 1, & L(Q_i) < 0 \\ 0, & \text{інакше} \end{cases} \quad (3.22)$$

Алгоритм min-sum. Алгоритм Min-Sum (MSA) є модифікованою версією алгоритму Sum-продукту (SPA). Тут операція перевірного вузла спрощується, щоб зменшити складність алгоритму. У MSA, квантоване внутрішнє повідомлення, також відоме як логарифмічні відношення правдоподібності (LLR) і зовнішнє повідомлення рівні по довжині. Рівняння оновлення перевірного вузла задається виразом:

$$L(r_{ji}) = \left[ \prod_{i' \in V_j \setminus i} \alpha_{i'j} \right] \cdot \min_{i' \in V_j \setminus i} \beta_{i'j} \quad (3.23)$$

Квантування каналу даних. На ефективність алгоритму впливає квантування вхідних повідомлень. Квантування даних каналу при декодуванні LDPC представлено на блок-схемі, яка показана на рис. 3.8.



Рисунок 3.8 – Блок-діаграма системи

Модифікований алгоритм min-sum (MMS). Алгоритм MMS використовує більш високу точність LLR повідомлень і нижчу точність (2-розрядна версія) зовнішніх повідомлень. Операції перевірного вузла та змінного вузла алгоритму MMS наступні:

1) Робота змінного вузла. Цей алгоритм практично аналогічний алгоритму min-sum. Єдина відмінність полягає в тому, що змінний вузол виконує більш високу точність операцій квантується LLR і відображає результат для 2-бітового повідомлення. 2-бітове повідомлення передається до контрольного вузла для подальшої обробки. Операція визначається виразом:

$$V_i = g \left( LLR_n + \sum_{j \neq i} f(C_j) \right) \quad (3.24)$$

де  $n=1,2,\dots,N$  (змінні вузли) та  $i=j=1,2,\dots,d_v$  (ступінь змінного вузла 'n')

$$g(y) = \begin{cases} 01, & y > T_m \\ 00, & 0 \leq y \leq T_m \\ 10, & 0 > x \geq -T_m \\ 11, & x < -T_m \end{cases} \quad (3.25)$$

$$f(x) = \begin{cases} +W, & x = 01 \\ +w, & x = 00 \\ -w, & x = 10 \\ -W, & x = 11 \end{cases} \quad (3.26)$$

де  $T_m$  – це поріг для відображення,  $W$  є верхньою цілісною константою, а  $w$  нижньою цілісною константою.

2) Робота перевірного вузла. У цьому кроці XOR операція ( $M_k$ ) використовується, щоб знайти добуток знаку вхідних повідомлень. Операція

AND ( $S_k$ ) використовується для визначення мінімумів. Вихідне повідомлення ( $C_k$ ) визначається об'єднанням бітів знака та бітів величини.

$$S_k = V_1^{(s)} \oplus V_2^{(s)} \oplus \dots \oplus V_l^{(s)} \quad \forall l \neq k$$

$$M_k = V_1^{(M)} \oplus V_2^{(M)} \oplus \dots \oplus V_l^{(M)} \quad \forall l \neq k$$

$$C_k = \{S_k M_k\} \quad \forall l \neq k$$

Порівняння продуктивності алгоритмів.

Створення програмної моделі для алгоритмів SPA, min-sum (MSA) та модифікованого min-sum алгоритму (MMS) і аналіз продуктивність BER кожного алгоритму було виконано з використанням MATLAB. Алгоритм прогресивного зростання ребер (PEG) використовується для створення LDPC кодів. Коди модульовані з використанням двійкової фазової маніпуляції.

### Висновки до третього розділу

У третьому розділі представлена конструкція декодера QC-LDPC з мінімальною затримкою і подвійним шляхом, низькою складністю, високою пропускною здатністю та безперервною обробкою. За допомогою запропонованого алгоритму декодер може обробляти подвійний блок паралельно та протистояти залежності даних від алгоритму багаторівневого декодування. Стратегія подвійного шляху також зменшує затримку обробки та підвищує пропускну здатність декодера при приблизно рівній продуктивності BER. Більше того, також спроектовано та реалізовано відповідну апаратну архітектуру цієї схеми.

## 4 ДОСЛІДЖЕННЯ ПРОПУСКНОЇ ЗДАТНОСТІ БЕЗПРОВІДНИХ ЛОКАЛЬНИХ МЕРЕЖ

### 4.1 Приймач цифрової системи зв'язку

Розглянемо модель приймача цифрової системи зв'язку с кодами LDPC, що приведена на рисунку 4.1.

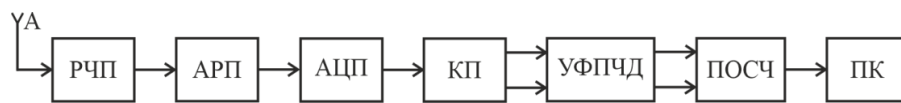


Рисунок 4.1. SDR архітектура приймача (РЧП – радіочастотний приймач; (АРП – автоматичне регулювання підсилення; АЦП – аналого-цифровий перетворювач; КП – квадратурний перетворювач; УФПЧД – узгоджений фільтр з пониженням частоти дискретизації; ПОСЧ – процесор основної смуги частот; ПК – персональний комп'ютер)

Радіочастотні сигнали перетворюються зі зниженням частоти в сигнали основної смуги частот в радіочастотному приймачі (РЧП).

Процесор основної смуги частот складається з блоків демодуляції QAM та декодування LDPC, які реалізовані за допомогою інструментарію HDL.

Simulink модель дослідження обробки сигналів приведена на рис. 4.2.

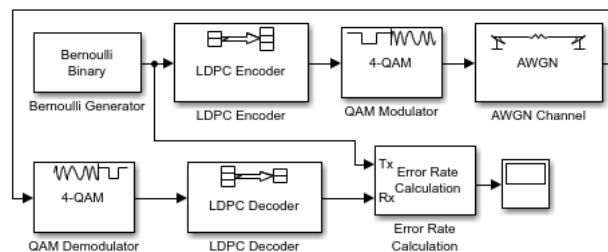


Рисунок 4.2 – Simulink модель дослідження ефективності обробки сигналів у процесорах основної смуги частот передавача і приймача

## 4.2 Коды LDPC і алгоритм мінімальної суми

Двійковий код LDPC являє собою лінійний блоковий код, який визначається розрідженою  $M \times N$  матрицею  $H$  перевірки на парність, де  $M$  представляє кількість перевірок на парність,  $N$  представляє кількість бітів у кодовому блоці. Його можна представити дводольним графом Таннера з  $M$  контрольними вузлами і  $N$  змінними вузлами. Ребро з'єднує контрольний вузол  $m$  зі змінним вузлом  $n$  тоді, коли  $H(m,n)$  не дорівнює нулю. Кількість одиниць у рядку називається вагою рядка, а кількість одиниць у стовпці – вагою стовпця. Код LDPC називається регулярним, якщо ваги рядків та стовпців не змінюються вздовж матриці.

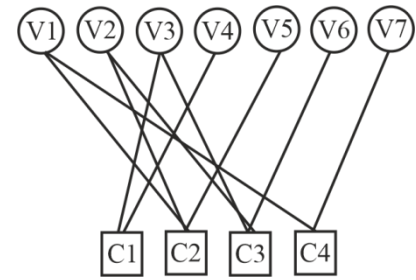
Структуровані коди LDPC є найпопулярнішими, оскільки мають дуже хорошу продуктивність і допускають нескладні апаратні реалізації. Ці коди генеруються шляхом розбиття матриці  $H$  на  $M_b \times N_b$  квадратних підматриць розміром  $z \times z$  з врахуванням, що  $M = M_b \cdot z$  і  $N = N_b \cdot z$ .

Нехай ми позначили множину (сусідніх) бітів, що приймають участь у перевірці  $m$ , через  $N_m = \{n : H(m,n) = 1\}$ , а множину (сусідніх) перевірок у бітах  $n$  - через  $M_n = \{m : H(m,n) = 1\}$ . Нехай  $N_{m \setminus n}$  позначає множину  $N_m$  з виключеним бітом  $n$  і  $M_{n \setminus m}$  множину  $M_n$  з виключеною перевіркою парності  $m$ . Припустимо, що кодове слово  $w = (w_1, w_2, \dots, w_N)$  передається каналом з адитивним білим гаусовим шумом (AWGN) з нульовим середнім значенням та дисперсією  $\sigma^2$  з використанням квадратурної фазової маніпуляції (QPSK), і нехай  $r = (r_1, r_2, \dots, r_N)$  буде відповідною прийнятою кодовою послідовністю.

На рис. 4.3 наведений приклад матриці  $H$  перевірки на парність і відповідний граф Танера. Рядки матриці представлені квадратами і називаються контрольними вузлами, стовпці матриці представлені кружками і називаються вузлами даних, а одиниці представлені ребрами у графі.

$$H = \begin{vmatrix} 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 \end{vmatrix}$$

а)



б)

Рисунок 4.3 – Приклад матриці перевірки на парність (а)  
та відповідний граф Таннера (б)

### 4.3 Кодування і декодування кодів LDPC

Операція кодування полягає спочатку у знаходженні генераторної матриці  $G$  такої, що  $G \cdot H^T = 0$ . Перевірочна матриця проходить попередню обробку, мета якої - представлення матрицю у нижчій псевдо-трикутній формі (рис. 4.4), використовуючи перестановки рядків або стовпців. Матриця складається з 6 розріджених підматриць, позначених  $A$ ,  $B$ ,  $C$ ,  $D$ ,  $E$ , та нижньої трикутної підматриці  $T$ .  $O$  – це нульова матриця. Розмір підматриці  $T$  дорівнює  $(m - g) \times (m - g)$ ,  $g$  має мати якнайменше значення. Після попередньої обробки  $H$ , відбувається кодування у відповідності з рівнянням:

$$G \cdot H^T = 0^T.$$

Кодове слово  $C$  визначається з інформації  $u$ , що передається, виразом:

$$C = u \cdot G$$

Розглянемо декодування кодів LDPC. Декодування кодів LDPC виконується за допомогою ітеративного алгоритму поширення довіри (belief propagation algorithm – BPA).

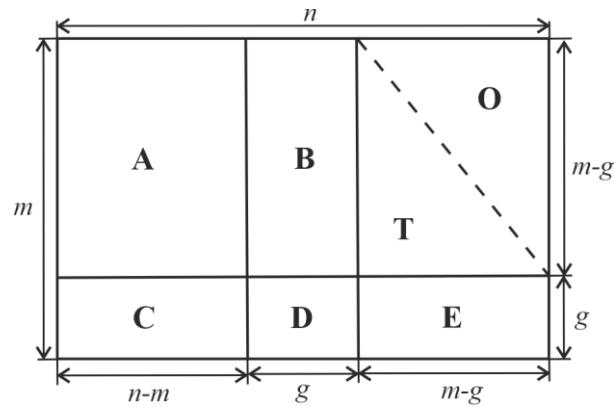


Рисунок 4.4 – Матриця перевірки парності

У роботі проведено дослідження алгоритму нормалізованої мінімальної суми, який адаптований до апаратної реалізації. Алгоритм полягає в тому, щоб спочатку оновлювати вузли даних, потім перевіряти вузли на кожній ітерації і в кінці приймати жорстке рішення декодування, яке є найбільш ймовірним кодовим словом [3].

#### 4.4 Алгоритм нормалізованої мінімальної суми

Розглянемо реалізацію LDPC-декодера за допомогою ПЛІС. Для того, щоб вважати процедуру декодування успішною, на всіх перевірочних вузлах мають встановитись нульові значення. Алгоритм мінімальної суми (МС) є ітеративним двоетапним алгоритмом передачі повідомлень: на  $i$ -ій ітерації спочатку обчислюються повідомлення від змінного до перевірочного вузла, які представляють оцінку апостеріорного логарифмічного відношення правдоподібності (LLR), і відправляються відповідному сусіду перевірочного вузла. По-друге, повідомлення від перевірочного до змінного вузла обчислюються та надсилаються назад сусіднім змінним вузлам. Алгоритм МС виконується під час  $i = 1, \dots, I_{\max}$  ітерацій у наступній послідовності.

Ініціалізація  $\mu_{m,n}^{(i=0)} = 0, \forall m \in \{1, \dots, M\}, \forall n \in N_m$ :

Крок 1 (оновлення вузла змінних): для  $n \in \{1, \dots, N\}$  і  $m \in M_n$

$$\lambda_{n,m}^{(i)} = l_n + \sum_{m' \in M_{n \setminus m}} \mu_{m',n}^{(i-1)}$$

Крок 2 (оновлення вузла перевірки): для  $m \in \{1, \dots, M\}$  і  $n \in N_m$

$$\mu_{m,n}^{(i)} = \Gamma_{m,n}^{(i)} \cdot \min_{n' \in N_{m \setminus n}} (|\lambda_{n',m}^{(i)}|)$$

$$\Gamma_{m,n}^{(i)} = \prod_{n' \in N_{m \setminus n}} \text{sign}(|\lambda_{n',m}^{(i)}|)$$

Жорстке рішення: Ітераційний процес зупиняється, коли досягається максимальне число ітерацій  $I_{\max}$ , чи коли всі перевірки на парність задовольняються за допомогою жорстких рішень, розрахованих виразом:

$$\lambda_n = l_n + \sum_{m' \in M_n} \mu_{m',n}^{(i)}, \quad z_n = \begin{cases} 1, & \lambda_n > 0 \\ 0, & \text{інакше} \end{cases}$$

На практиці операція отримання мінімуму для кожного перевірного вузла  $m$  зводиться до обчислення значень першого мінімуму та другого мінімуму, та вибору відповідного для кожного виходу  $n$ . В апаратній реалізації два мінімальні значення для  $k$  входів розраховуються з використанням дерева двох мінімальних компараторів [4], складність яких складається з  $2 \cdot k - 3$  додавань/віднімань та  $3 \cdot k - 4$  мультиплексорів.

В апаратній реалізації знаходять мінімальне та друге мінімальне значення. Для знаходження двої значень – реалізувати трирівневе дерево компараторів, (рис. 4.5). Є сім однакових блоків, і кожен із них використовується для знаходження першого мінімального та другого мінімального значень. Коли мінімум та другий мінімум знайдені,

відбувається множення на масштабний коефіцієнт 0,75 для нормалізації. На рис. 4.6 приведена схема алгоритму нормалізованої мінімальної суми.

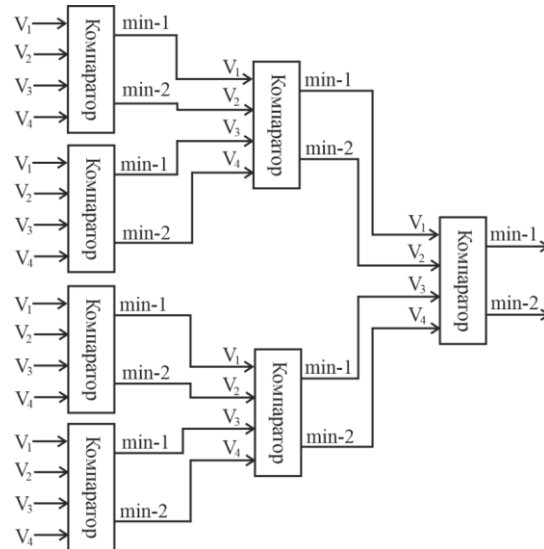


Рисунок 4.5 – Трирівневе дерево компараторів

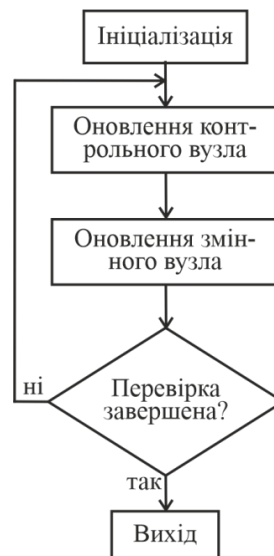


Рисунок 4.6 – Схема алгоритму нормалізованої мінімальної суми

Схеми, що використовуються для оновлення вузлів змінних та вузлів перевірки, показано на рисунку 4.7 та рисунку 4.8 відповідно.

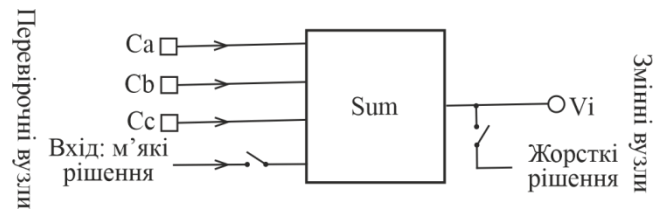


Рисунок 4.7 – Схема оновлення змінних вузлів

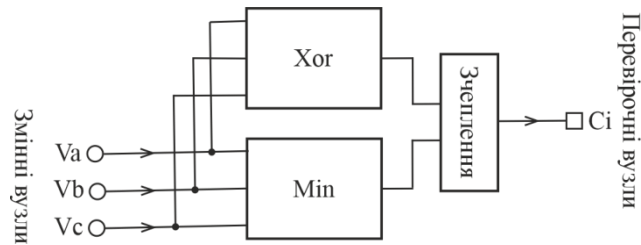
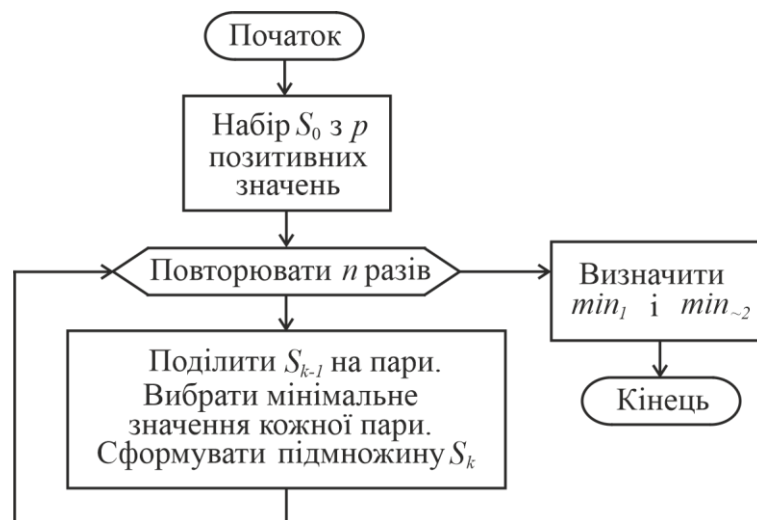


Рисунок 4.8 – Схема оновлення перевірочних вузлів

Алгоритм exMin- $n$  (рис. 4.9) вводить  $n$ -рівневу апроксимацію exMin, в якій процес розбиття та мінімальної ідентифікації, визначений на кроці 1, послідовно застосовується  $n$  разів.

Рисунок 4.9 – Алгоритм exMin- $n$ 

Другий мінімум  $S_n$  є апроксимацією другого мінімуму  $S_0$ . Позначимо через  $S_{\min}$  множину  $\{\min_1, \min_2\}$ , що складається з першого та наближеного другого мінімумів  $S_0$ , отриманих за алгоритмом exMin- $n$ .

Як наочний приклад припустимо, що набір  $S_0$  складається з  $p=8$  вхідних повідомлень, де  $S_0 = \{1,8,2,6,3,6,8,5\}$  і  $n=1$ . На кроці 1 набір  $S_0$  розбивається на пари як  $P = \{(1,8), (2,6), (3,6), (8,5)\}$ . Визначається мінімальне значення кожної пари і формується підмножина  $S_0$ , яке позначається як  $S_1$  і формується наступним чином ( $S_1 = \{1,2,3,5\}$ ). На кроці 2 ідентифікуються два мінімуми  $S_1$ , що формується наступним чином ( $S_{\min} = \{1,2\}$ ). Замість того, щоб знаходити два мінімуми  $S_1$ , крок 1 можна застосовувати ітеративно. В цьому випадку, посилаючись на вищезгаданий приклад,  $S_1$  можна додатково розбити на  $\{(1,2), (3,5)\}$ , після чого алгоритм повертає  $S_{\min} = \{1,3\}$ . Це двоетапне розбиття вводить наближення exMin-2.

Алгоритм exMin- $n$  призводить до значення  $\min_1$ , яке завжди є мінімальним серед значень  $S_0$ . Значення  $\min_2$  не обов'язково є другим найнижчим значенням  $S_0$ ; натомість це другий мінімум підмножини ( $S_n$ ) у  $S_0$ . Підмножина  $S_n$  не обов'язково містить друге мінімальне значення  $S_0$ . Це пов'язано з тим, що реальний другий мінімум  $\min_2$  може бути виключений з порівнянь через спосіб отримання підмножин  $S_k (1 \leq k \leq n)$ . Цей виняток відбувається, коли на якомусь етапі кроку 1  $\min_1$  та  $\min_2$  поділяються разом. У міру збільшення значення параметра  $n$  алгоритму exMin- $n$  збільшується ймовірність помилки, тобто ймовірність того, що розраховане значення  $\min_2$  не є дійсним другим нижчим значенням початкової множини  $S_0$ . exMin-1 повертає  $\min_2 = 2$ , що є правильним, на відміну від алгоритму exMin-2, який повертає наближення другого мінімального значення ( $\min_2 = 3$ ).

Позначимо як  $C_{p \sim 2}^n$  схему, яка ідентифікує два мінімуми з множини  $p$  вхідів для алгоритму exMin- $n$ . На рисунку 4.10, а показана схема  $C_{8 \sim 2}^1$ .

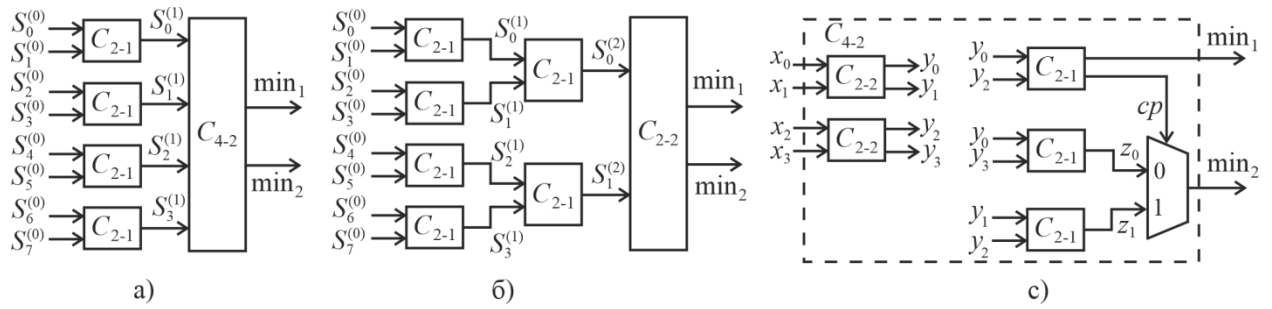


Рисунок 4.10 – Архітектура схеми  $C_{8-2}^1$  (а), схеми  $C_{8-2}^2$  (б) та схеми (с) для ідентифікації двох мінімумів серед набору з чотирьох значень.

До входів від  $s_0^{(0)}$  до  $s_7^{(0)}$  застосовується вісім повідомлень. Схема включає етап порівняння блоків  $c_{2-1}$  (рис. 4.10,а). Подаючи значення  $S_0$  на входи блоків порівняння, відбувається розбиття значень на пари. Кожен із блоків порівняння ідентифікує мінімальне значення пари значень на своїх входах. Це відповідає кроку 1 алгоритму exMin-1. Чотири результуючі значення  $s_0^{(1)}$  до  $s_3^{(1)}$ , які відповідають елементам вектора  $S_1$ , застосовуються до блоку обчислення мінімумів  $C_{4-2}$  (рис. 4.10, с). Ми позначаємо як  $C_{p-2}$  одиницю обчислення мінімумів, яка ідентифікує два справжні мінімальні значення серед  $p$  вхідних даних. Блок обчислення мінімумів  $C_{4-2}$  ідентифікує два мінімуми  $S_1$  і виводить ці значення  $\min_1$  і  $\min_2$  відповідно. Це відповідає кроку 2 алгоритму exMin-1. Схема на рис. 4.10, б може бути використана для виконання наближення exMin-2. Одиниці порівняння  $c_{2-1}$  у двоетапної деревоподібної організації отримують пари значень, отримані на попередньому етапі, або первинні вхідні дані, і кожен із них ідентифікує мінімальне значення пари. Отримана єдина пара значень, тобто  $s_0^{(2)}$  і  $s_1^{(2)}$  застосовується до одиниці  $C_{2-2}$  (рис. 4.11, б). Можна додати будь-яку кількість таких стадій, щоб реалізувати  $n$ -рівневу апроксимацію exMin.

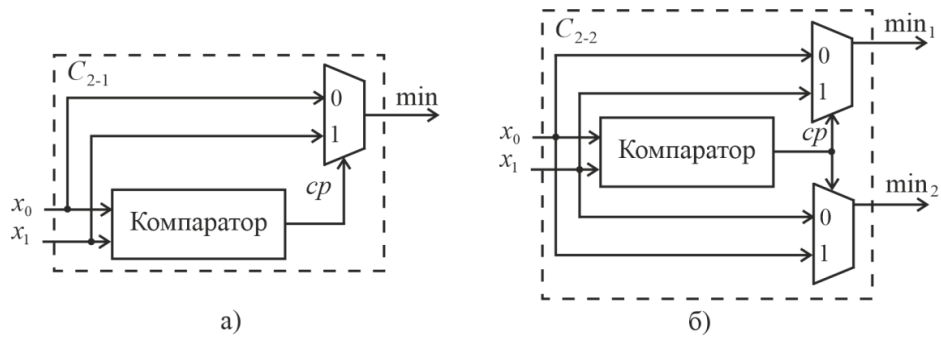


Рисунок 4.11 – Схеми для визначення (а) мінімуму з двох значень та (б) мінімуму та другого мінімуму з двох значень.

Розглянемо алгоритм  $r\text{ExMin-}n$ . Алгоритм  $\text{exMin-}n$  ідентифікує значення  $\text{min}_{\sim 2}$ , яке наближається до другого мінімуму вхідних повідомлень, і воно завжди більше або дорівнює справжньому другому мінімальному значенню. На додаток до пропозиції  $\text{exMin-}n$ , ми також пропонуємо його варіант, який ми називаємо  $r\text{ExMin-}n$ , де результуюче значення  $\text{min}_{\sim 2}$  правильно зменшується шляхом додавання негативного коефіцієнта  $r$ . Моделювання показало, що застосування цього понижуючого коефіцієнта значення  $\text{min}_{\sim 2}$  знижує деградацію BER, викликану апроксимацією  $\text{exMin-}n$ .

Сума  $r$  і  $\text{min}_{\sim 2}$  може бути меншою або дорівнює істинному мінімальному значенню  $\text{min}_1$ . Коли  $r + \text{min}_{\sim 2} \leq \text{min}_1$ , алгоритм  $r\text{ExMin-}n$  повертає  $\text{min}_{\sim 2}$  як другу мінімальну оцінку; інакше алгоритм повертає  $r + \text{min}_{\sim 2}$ . Алгоритм  $r\text{ExMin-}n$  складається з наступних кроків.

На рис. 4.12 представлена схема алгоритму  $r\text{ExMin-}n$ . Другий мінімум  $r \text{min}_{\sim 2}$  більше першого мінімуму дорівнює йому при  $\text{min}_1 = \text{min}_{\sim 2}$ .

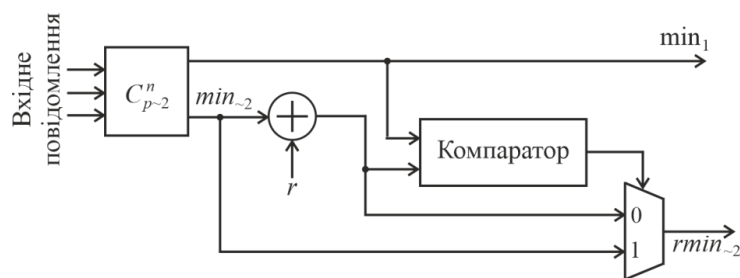


Рисунок 4.12 – Апаратна реалізація алгоритму  $r\text{ExMin-}n$

Ми оцінили продуктивність BER приблизних декодерів MS LDPC, які використовують алгоритми exMin для визначення двох мінімумів. Перевірка синдрому виконується після кожної ітерації. Якщо алгоритм сходиться до допустимого кодового слова, процес декодування завершується (дострокове завершення); в іншому випадку декодування продовжується доти, поки не буде досягнуто задане число ітерацій  $I_{\max}$ . Раннє завершення значно знижує середню кількість ітерацій (NOI), особливо у робочих точках з низьким BER, тим самим збільшуючи швидкість декодування.

Моделювання точності з плаваючою комою повністю паралельної двофазної ітеративної схеми планування було виконано в припущенні, що символи бінарної фазової маніпуляції передаються по каналу з адитивним білим гаусовим шумом. Ми порівнюємо продуктивність запропонованих схем із продуктивністю звичайних алгоритмів MS, NMS та алгоритмів mfMS та svwMS з одним мінімумом для  $I_{\max} = 30$ . Алгоритм exMin-3 забезпечує підсилення кодування на 0,15-0,2 дБ у порівнянні зі звичайним алгоритмом MS та кращими знайденими алгоритмами mfMS та svwMS. Продуктивність exMin-3 можна порівняти з продуктивністю NMS для значень BER до  $10^{-6}$  з погіршенням на 0,08 дБ при BER  $10^{-7}$ . Алгоритм rExMin-3 покращує продуктивність exMin-3 у діапазоні BER нижче  $10^{-6}$ . Ми також оцінили продуктивність алгоритму exMin-3 для коду (1944, 1458) LDPC. Результати на рис. 5 показують втрату 0,12-0,15 дБ порівняно з алгоритмом NMS. Алгоритм rExMin-3 забезпечує підсилення на 0,06 дБ проти exMin-3. Скорочуючи рівні апроксимації до двох, схема exMin-2 наближається до продуктивності алгоритму BER NMS. Усі реалізації exMin перевершують алгоритми MS, mfMS та svwMS. Схеми exMin забезпечують задовільну ефективність кодування, порівняну з ефективністю алгоритму NMS, перевищуючи продуктивність алгоритми MS і алгоритми з одним мінімумом.

Реалізації алгоритму exMin також добре працюють, коли  $I_{\max}$  встановлено на 10. Алгоритм exMin-1 наближається до продуктивності BER алгоритму NMS для випадку десяти ітерацій. Гнучкість у виборі кількості рівнів апроксимації дозволяє алгоритму exMin-n досягати задовільної ефективності кодування незалежно від максимального NOI. Це значна перевага в порівнянні з іншими спрощеними алгоритмами MS, які показують значне зниження продуктивності при низькому максимальному NOI.

У більшості випадків алгоритми exMin сходяться швидше, ніж алгоритм swMS, і приблизно так само швидко, як алгоритм NMS. Навіть при максимальній кількості ітерацій в алгоритмах exMin потрібно менше 14 ітерацій для збіжності.

Коефіцієнт нормалізації  $\alpha = 0,75$  був застосований до двох мінімумів, отриманих усіма алгоритмами, що оцінюються (за винятком звичайного алгоритму MS). Коефіцієнт зменшення  $r$  схем rExMin-n було встановлено рівним  $-0,5$ . Щоб досягти задовільної ефективності виправлення помилок, алгоритми з одним мінімумом оцінюють другий мінімум, застосовуючи оптимальні коефіцієнти поправки до мінімального значення. Ці фактори визначаються після тривалих симуляцій та залежать від конкретного коду. Відсутність такої операції корекції є додатковою перевагою алгоритму exMin-n. Наведемо найкращі вагові коефіцієнти алгоритмів mfMS і svwMS. Зокрема, для алгоритму svwMS вагові коефіцієнти дорівнюють  $w_1$  для ітерацій від 1 до  $u_1$ ;  $w_{i+1}$  для ітерацій від  $u_i + 1$  до  $u_{i+1}$ , де  $1 \leq i \leq f$  і  $f = \text{size}(u)$ ; та  $w_{f+1}$  для ітерацій від  $u_f + 1$  до  $I_{\max}$ .

Звичайний контрольний вузол MS (MS і NMS) включає схему, яка обчислює два мінімальних значення з набору  $p$  вхідних повідомлень. Найбільш ефективна за площею схема для знаходження фактичних двох мінімумів – це схема, апаратна складність якої дорівнює  $C_{Lee}^p = p + \lceil \log_2 p \rceil - 2$  компараторів та  $M_{Lee}^p = 3 \cdot p - 4$  мультиплексорів. Деревоподібна архітектура (TS), перевищує за критичною затримкою за рахунок збільшення

компараторів ( $C_{TS}^p = 2 \cdot p - 3$ ). Наближений алгоритм exMin-n суттєво зменшує необхідне обладнання для обчислення двох мінімумів.  $k$ -ий додаток кроку 1, де  $1 \leq k \leq n$ , використовує  $t_k = \lceil p_{k-1} / 2 \rceil$  одиниць порівняння [ $C_{2-1}$  на рис. 2, а), де  $p_k = \lceil p / 2^k \rceil$ . Результуючий набір  $S_k$  складається із значень  $p_k$ . Ми використовуємо схеми для ідентифікації двох мінімумів  $S_n$ , як визначено на кроці 2 алгоритму exMinn. У схемі exMin-n

використовуються  $C_n = C_{conv}^{p_n} + \sum_{k=1}^n t_k$  компараторів і

$M_n = M_{conv}^{p_n} + \sum_{k=1}^n t_k = (3 \cdot p_n - 4) + \sum_{k=1}^n t_k$  мультиплексорів, де  $C_{conv}^{p_n}$  дорівнює  $C_{Lee}^{p_n}$ ,

коли ми використовуємо площу, або  $C_{TS}^{p_n}$  у разі швидшого підходу TS. Зменшення площі залежить від параметра  $n$ .

Алгоритми smMS, mfMS та vwMS заміняють обчислення двох мінімумів обчисленням мінімуму ( $\min_1$ ) та оцінкою другого мінімального значення. Складність мінімальної ідентифікації для набору  $S_0$  із  $p$  значень становить  $p-1$  компаратор і  $p-1$  мультиплексор. Ці алгоритми вимагають вичерпного пошуку, щоб визначити, чи існує більше одного абсолютного мінімального значення. Вартість апаратного забезпечення є доступною для архітектур контрольних вузлів з послідовною передачею даних. Проте це неприйнятно для паралельних операцій check-node, де цей пошук повинен виконуватися за один такт. Для зниження складності алгоритм svwMS ділить вхідні повідомлення на дві групи і порівнюють мінімальні значення цих груп. Якщо ці значення дорівнюють, другим мінімум вважається рівним  $\min_1$ . В іншому випадку  $\min_1$  додається змінний ваговий коефіцієнт  $w$  для оцінки другого мінімуму. Для цієї операції потрібно один компаратор, один мультиплексор та один суматор. Реалізація svwMS з  $m$  різними ваговими коефіцієнтами  $w_i$  вимагає  $m-1$  мультиплексорів 2-до-1 для вибору  $w$ .

Код LDPC 802.11n/ac (1944, 1620) має степені рядків 19 і 20. Це означає, що до 20 повідомлень зі змінними для перевірки повинні оброблятися паралельно, щоб ідентифікувати два мінімуми серед них за один системний такт. Максимальні степені рядків кодів (1944, 1458) та (648,432) дорівнюють 15 та 11 відповідно. Зниження складності та зменшення затримки через апроксимацію exMin становлять від 6% до 15% і від 9% до 23% відповідно. Існує принаймні одна схема exMin, яка перевершує алгоритм svwMS з точки зору продуктивності виправлення помилок, використання області та затримки. Simulink модель HDL-реалізації LDPC декодера представлена на рис. 4.13.

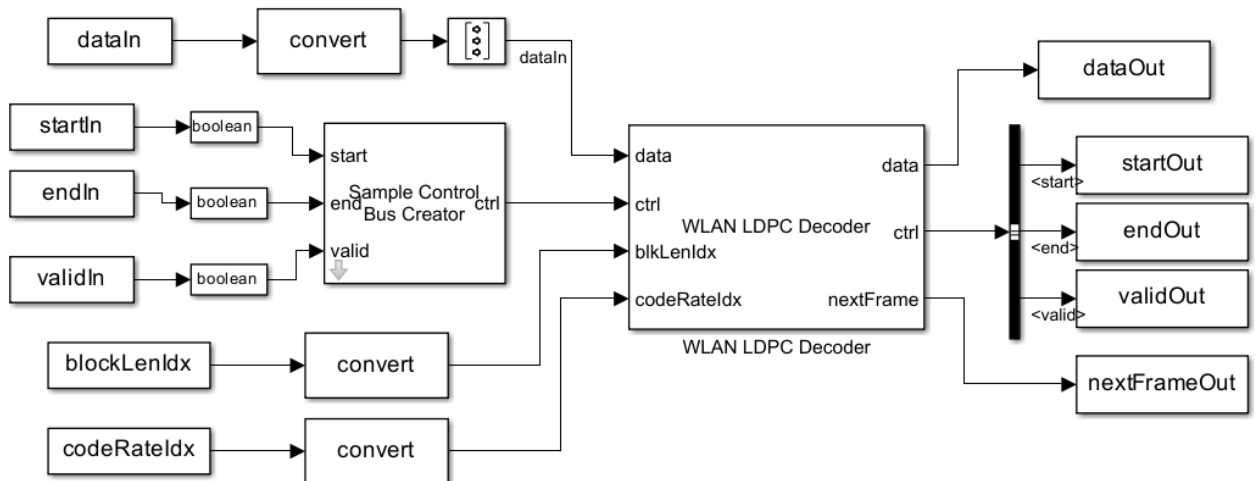


Рисунок 4.13 – Simulink модель LDPC декодера

Для декодера на базі FPGA перетворюються фрейми на вибірки.

На вході (сигнал `dataIn`) діють значення логарифмічного відношення правдоподібності (LLR), які отримані на виході демодулятора символів QAM; формуються управляючі сигнали, що вказують початок і кінець фрейму (сигнали `startIn`, `endIn`), сигнал вірності даних (сигнал `validIn`).

Для врахування затримки блоку декодера LDPC для зазначеної довжини блоку, швидкості коду та кількості ітерацій використовується змінна `decFrameGap`. Вихідний сигнал `nextFrame` використовується для визначення готовності блоку до початку обробки наступного фрейма.

Затримка блоку дорівнює  $r \cdot (t + 9m) + d$ , де  $r$  – кількість ітерацій,  $t$  – подвоєна загальна кількість елементів у матриці контролю парності, відмінних від 1,  $m$  – кількість рядків у матриці, а  $d$  –конверсні затримки.

#### 4.5 Результати експериментальних досліджень

Проведем дослідження залежності коефіцієнту бітових помилок від відношення сигнал-шум для системи зв'язку з кодуванням LDPC і різною кодовою швидкістю (рис. 4.14 – 4.17).

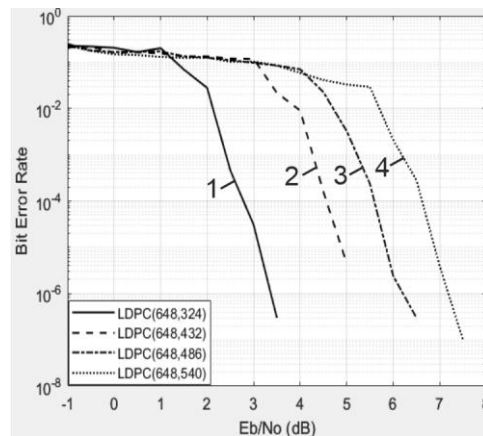


Рисунок 4.14 – Залежність BER від SNR для системи зв'язку з довжиною фрейму 648 бітів і кодовою швидкістю: 1 - 1/2; 2 - 2/3; 3 - 3/4; 4 - 5/6.

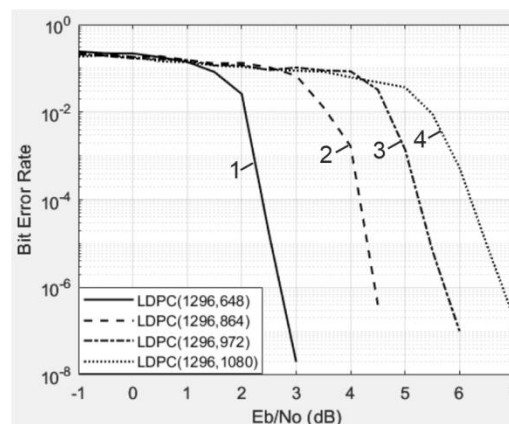


Рисунок 4.15 – Залежність BER від SNR для системи зв'язку з довжиною фрейму 1296 бітів і кодовою швидкістю: 1 - 1/2; 2 - 2/3; 3 - 3/4; 4 - 5/6.

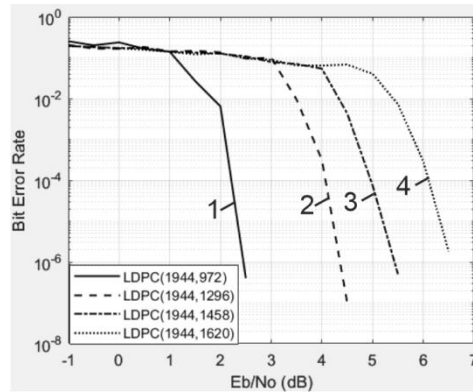


Рисунок 4.16. Залежність BER від SNR для системи зв'язку з довжиною фрейму 1944 бітів і кодовою швидкістю: 1 - 1/2; 2 - 2/3; 3 - 3/4; 4 - 5/6.

Під час передачі даних фізичним каналом, необхідно додати деякі надлишкові коди для виправлення помилок. Резервування використовується для досягнення високої надійності. Збільшення кодової швидкості з 1/2 до 5/6 потребує підвищення відношення сигнал-шум на 4 дБ

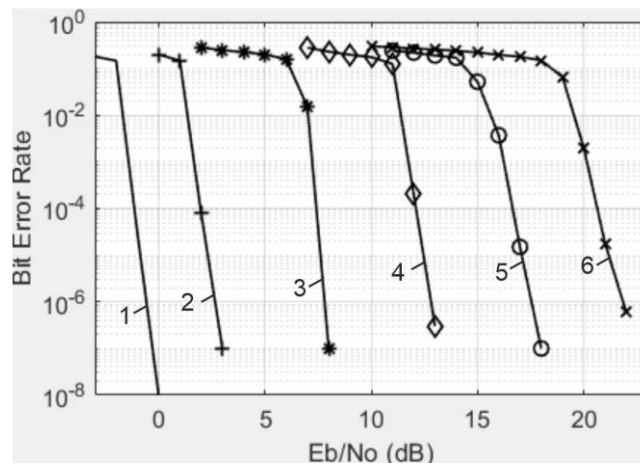


Рисунок 4.17 – Залежність BER від SNR системи зв'язку з кодуванням LDPC. довжина фрейму 1944 бітів, кодова швидкість 1/2. Модуляція: 1 – BPSK; 2 – QPSK; 3 – 16-QAM; 4 – 64-QAM; 5 – 256-QAM; 6 – 1024-QAM.

Використовується різна технологія модуляції, тобто кількість біт, яка може передаватись у символі. Від Wi-Fi 1 до Wi-Fi 6 кожна нова технологія

модуляції збільшувала швидкість просторового потоку на 20%. При цьому відношення сигнал-шум повинно збільшитись на 5 дБ.

Стандарт 802.11ax використовує множинний доступ з ортогональним частотним поділом каналів, тривалість символу 12,8 мкс, інтервал між підносійними складає 78,125 кГц. Менший інтервал між підносійними забезпечує більшу надійність каналу зв'язку. Стандарт 802.11ax може використовувати модуляцію 1024-QAM, що підвищує швидкість передачі інформації.

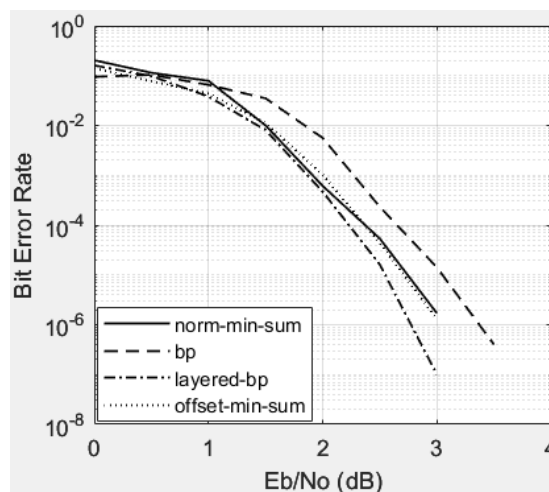


Рисунок 4.18 – Залежність коефіцієнту бітових помилок від відношення сигнал-шум системи зв'язку з кодуванням LDPC(648, 324). Алгоритм "norm-min-sum", довжина блоку 648, кодова швидкість 1/2.

Найкращу захищеність від завад має багаторівневий алгоритм поширення довіри, але для спрощення апаратної реалізації використовують алгоритм нормалізованої мінімальної суми.

Проведемо дослідження близькості коду LDPC до межі Шеннона. Для цього розглянемо коди, довжини яких затверджені стандартом IEEE 802.11ax і аналогічні коди, довжини яких збільшені у 24 рази.

Для модуляції BPSK код LDPC швидкістю 0,5 різної довжини. Вертикальною лінією проведена межа Шеннона на рівні -1,59 дБ.

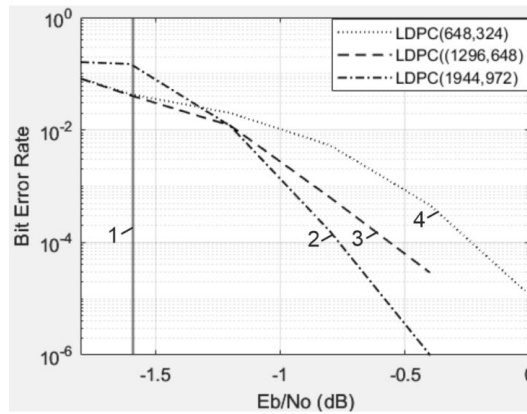


Рисунок 4.19 – Залежність BER від SNR системи зв'язку з кодуванням LDPC.

Довжина блоку: 2 – 1944; 3 – 1296; 4 – 648. 1 – межа Шеннона

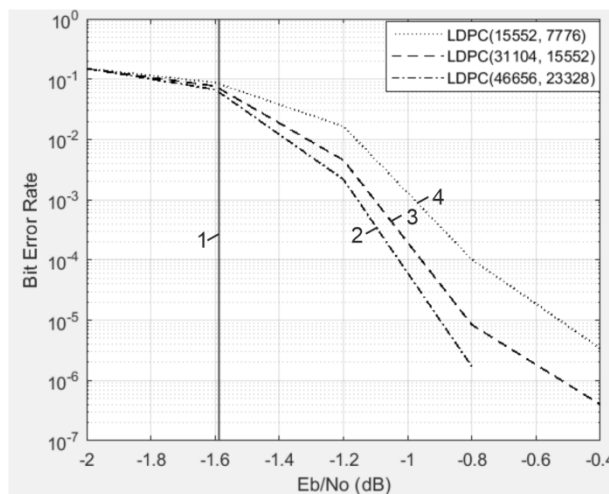


Рисунок 4.20 – Залежність BER від SNR системи зв'язку з кодуванням LDPC.

Довжина блоку: 2 – 46656; 3 – 31104; 4 – 15552. 1 – межа Шеннона

З отриманих залежностей можна зробити висновок, що код LDPC для стандарту 802.11ах, LDPC(1944,972) – з найбільшою довжиною блока даних, наближається до межі Шеннона на 1,2дБ. При збільшенні довжини блока у 24 рази (код LDPC(46656,23328)), це наближення складає 0,8 дБ, але потребує значного збільшення ресурсів апаратного забезпечення FPGA.

Сучасні схеми модуляції, такі як OFDM, об'єднують кілька окремих підносійних у канал фіксованої ширини, тому вони отримують більше даних в секунду із заданої ширини каналу, ніж теоретично припускав Шеннон, оскільки Шеннон обмежив свою модель каналу однією носійною.

#### 4.6 Апаратне забезпечення декодера LDPC

Розглянемо апаратну реалізацію гібридного декодера LDPC матриць перевірки парності, зазначених у стандарті IEEE 802.11n розміру блоку 648 [3]. Як показано на рисунку 4.21, апаратна архітектура декодера LDPC, має вигляд

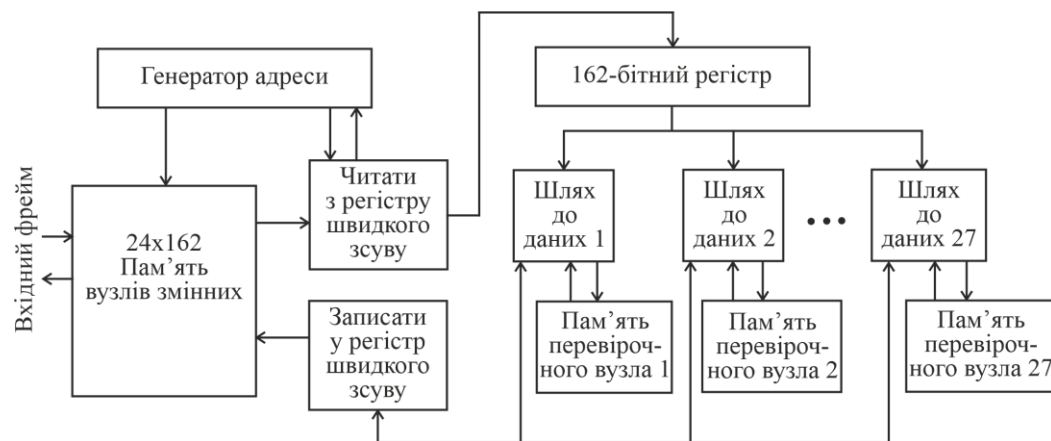


Рисунок 4.21 – Архітектура апаратного забезпечення декодера LDPC

Оскільки розмір матриці перевірки парності становить  $27 \times 27$ , ми використали 27 шляхів даних перевірювальних вузлів для паралельної реалізації алгоритму декодування мінімальної суми одного рівня. Після завершення оновлення змінних вузлів, обробляється наступний рівень матриці перевірки парності, що призводить до реалізації гібридного декодера LDPC.

Апаратна архітектура складається з пам'яті змінних вузлів  $24 \times 162$ , двох багато-регістрових схем циклічного зсуву, 27 каналів даних контрольних вузлів та 27 комірок пам'яті  $12 \times 38$  контрольних вузлів. Пам'ять розміром  $24 \times 162$  біта використовується для зберігання 648 повідомлень вузлів змінних, кожне з яких є 6-бітним, включаючи 1 знаковий біт. Пам'ять вузла змінних організована таким чином, що в кожному слові зберігається  $27 \times 6 = 162$ -бітові повідомлення для паралельного відправлення 27 повідомлень.

Оскільки підматриці стандарту IEEE 802.11n є зсунутими версіями ідентифікаційних матриць  $27 \times 27$ , перед відправкою повідомлень змінних

вузлів слово має бути зсунуте шляхом зчитування з багато-регістрової схеми циклічного зсуву, щоб відправити правильні повідомлення змінних вузлів у кожний шлях даних контрольного вузла. Потім оновлені повідомлення вузла змінної записуються назад у пам'ять після того, як вони зсунуті назад у вихідне положення за допомогою багато-регістрової схеми циклічного зсуву.

У матриці перевірки парності з довжиною блоку 648 і кодовою швидкістю 5/6 кожен контрольний вузол з'єднаний з 22 змінними вузлами. Тому для обчислення повідомлення контрольного вузла, як у рівнянні 2 кожному каналу даних контрольного вузла відправляються повідомлення змінного вузла за 22 циклу. У нашому апаратному декодері замість зберігання всіх повідомлень вузлів змінних для кожного контрольного вузла ми зберігаємо їх суму для кожного вузла змінних.

Потім, як показано на рисунку 4.22, на шляху даних контрольного вузла повідомлення контрольного вузла, відправлене на попередній ітерації, віднімається від загального повідомлення вузла змінної для отримання окремого повідомлення вузла змінної, як показано у рівнянні 4.5.

$$Q_{nm}^i = Q_n - R_{mn}^{i-1} \quad (4.5)$$

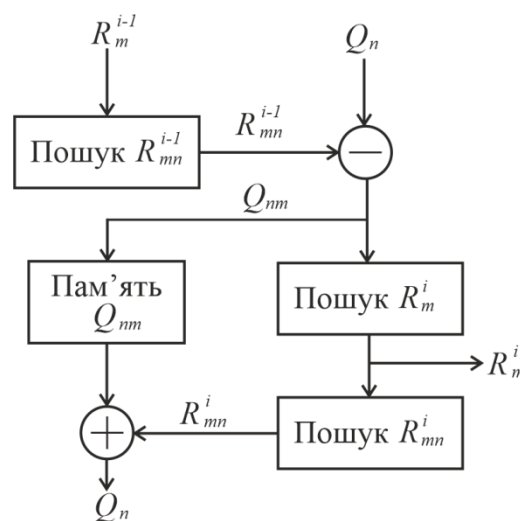


Рисунок 4.22 – Перевірка шляху до даних вузла

Після розрахунку  $Q_{nm}$  для всіх 22 повідомлень зі змінним вузлом блок « $R_m^i$  finder» знаходить мінімальне значення серед 22 повідомлень  $Q_{nm}$  і відправляє повідомлення довжиною 38 біт, що містить 4-бітове мінімальне значення, 5-бітний індекс мінімуму, 24-бітові знаки 24 вузлів змінних, з яких тільки 22 використовуються для кожного шару, і 1 біт для хог знаків 22 вузлів змінних. Це 38-бітове повідомлення зберігається в пам'яті контрольних вузлів розміром 4x38. «Пошук  $R_{mn}^{i-1}$ » і «Пошук  $R_{mn}^i$ » у тракті даних контрольного вузла використовуються для розпакування 38-бітових повідомлень  $R_m$  та пошуку окремих повідомлень вузла перевірки для змінної.

### **Висновки до четвертого розділу**

1. Досліджена система за стандартом 802.11ax, декодер LDPC за спрощеним алгоритмом мінімальної суми. Досліджений LDPC декодера в середовищі Simulink. Співвідношення BER і SNR можна покращити за рахунок збільшення розміру коду та збереження принципу паралелізму.

2. Сучасний стандарт 802.11ax використовує технології MIMO і Multi-User MIMO, вводиться режим ортогонального частотного мультиплексування OFDMA для покращення спектральної ефективності і модуляція 1024-QAM для чотирикратного збільшення пропускної спроможності, що потребує збільшення відношення сигнал-шум на 5 дБ. Підтримуються наступні кодові швидкості LDPC: 1/2, 2/3, 3/4, 5/6; підтримуються наступні довжини блоків LDPC коду: 648, 1296 та 1944 біта.

3. Розглянуто апаратне забезпечення LDPC-декодера, що реалізує багаторівневий алгоритм декодування мінімальної суми для стандарту бездротової локальної мережі IEEE 802.11. Для довжини блоку 648 і швидкості коду 1/2 FPGA Xilinx Virtex II апаратна реалізація декодера LDPC працює на частоті 83,5 МГц і може обробляти 60,68 Мбіт/с.

## ВИСНОВКИ

1. При збільшенні швидкості коду зменшується завадостійкість системи зв'язку. При збільшенні кодової швидкості з 1/2 до 5/6 необхідно підвищувати відношення сигнал-шум на 4 дБ.

2. Використання багатопозиційної модуляції потребує підвищення відношення сигнал-шум: від 1 (BPSK) до 2 (QPSK) біт на символ модуляції – на 3 дБ; від 2 (QPSK) до 4 (16-QAM) біт на символ модуляції – на 5 дБ. Подальше збільшення позиційності модуляції до 6, 8 і 10 біт на символ – підвищення відношення сигнал-шум на 5 дБ на кожні 2 біта на символ.

3. Код LDPC для стандарту 802.11ax, LDPC(1944,972) – з найбільшою довжиною блока даних, наближається до межі Шеннона на 1,2дБ.

4. Зі збільшенням швидкості передачі даних велике значення набуває величина вектора помилки (EVM) на радіочастотному інтерфейсі. При 1024 підносійних Wi-Fi 6 (802.11ax) сузір'я стає дуже щільним і системі важко відрізнити одну з точок від іншої. Для декодування (або демодуляції) цих точок сузір'я потрібно мати кращу EVM.

5. Стандарт IEEE 802.11ax використовує OFDMA (Orthogonal Frequency Division Multiple Access), або множинний доступ з ортогональним частотним поділом, який дозволяє блокам ресурсів (RU) надавати кільком абонентам однакову швидкість передачі даних.

6. У стандарті Wi-Fi 6 впроваджено технологію MU-MIMO, що дозволяє точці доступу передавати дані кільком терміналам одночасно, що значно підвищує пропускну спроможність.

## ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. M. S. Afaqui, E. Garcia-Villegas and E. Lopez-Aguilera, "IEEE 802.11ax: Challenges and Requirements for Future High Efficiency WiFi," in IEEE Wireless Communications, vol. 24, no. 3, pp. 130-137, June 2017.
2. A. Boudaoud, M. El Haroussi, E. Abdelmounim. VHDL Design and FPGA Implementation of LDPC Decoder for High Data Rate /International Journal of Advanced Computer Science and Applications(IJACSA), Volume 8 Issue 4, 2017.
3. Бойко Ю. М. Теоретичні аспекти підвищення завадостійкості й ефективності обробки сигналів в радіотехнічних пристроях та засобах телекомунікаційних систем за наявності завад : монографія / Ю. М. Бойко, В. А. Дружинін, С. В. Толюпа. - Київ : Логос, 2018. - 227 с.
4. Пятін І.С., Бойко Ю.М. Енергетична ефективність пропускнуої спроможності кодів 5G LDPC //IX Міжнародна НТК «Радіотехнічні поля, сигнали, апарати та системи» 16-22 листопада 2020 року
5. Пятін І.С. Методи генерації перевірочних матриць LDPC коду /І.С. Пятін, В.В. Мішан // ВОТТП. – 2020. - №2. – С. 18-24.
6. Пятін І. С. Дослідження енергетичної ефективності каналного кодування даних користувача кодами LDPC для систем зв'язку 5G [Текст] / І. С. Пятін, Ю. М. Бойко // Вісник Хмельницького національного університету. Технічні науки. – 2020. – №3. – С. 174-185,
7. Tahir B. BER comparison between Convolutional, Turbo, LDPC, and Polar codes," / B. Tahir, S. Schwarz, M. Rupp // 2017 24th International Conference on Telecommunications (ICT), Limassol 3-5 May 2017, Limassol - 2017, pp. 1-7.
8. J. Boiko, I. Pyatin and O. Eromenko, "Determining the Effectiveness of LDPC Codes for 5G Information Networks," 2021 IEEE 12th International Conference on Electronics and Information Technologies (ELIT), 2021, pp. 159-164.

9. Boiko J. Design and Evaluation of the Efficiency of Channel Coding LDPC Codes for 5G Information Technology / J. Boiko, I. Pyatin and O. Eromenko // Indonesian Journal of Electrical Engineering and Informatics (IJEI), vol. 9, no. 4, pp. 867-879, Dec. 2021.

10. I. Pyatin, J. Boiko and O. Eromenko, "Evaluating the Productivity of HDL Efficient Coding Models for 5G Information Networks," 2021 IEEE 8th International Conference on Problems of Infocommunications, Science and Technology (PIC S&T), 2021, pp. 305-308.

11. Пятін І. С. Порівняння продуктивності завадостійких кодів на основі програмного HDL моделювання для захищених інформаційних технологій / І. С. Пятін, Ю. М. Бойко // Інфокомунікаційні та комп'ютерні технології. – 2022. – № 1(03). – С. 39-62.

12. Пятін І.С. Підвищення пропускної спроможності мобільних телекомунікацій / І.С. Пятін, Ю.М. Бойко // Проблеми експлуатації та захисту інформаційно-комунікаційних систем : матеріали наук.-практ. конф., (Київ, 7 – 9 черв. 2022 р). – Київ : НАУ, 2022. – С. 95.

13. Diop, Idy, Sidi Mohamed Farssi, Marc Chaumont, Ousmane Khouma, H. B. Diouf, Khaly Tall and K. Sylla. "USING OF LDPC CODES IN STEGANOGRAPHY." / Journal of Theoretical and Applied Information Technology. January, 2012 38(1):103-109 (2012).

14. Бойко Ю.М. Моделі систем завадостійкого кодування у телекомунікаціях / Ю.М. Бойко, І.С. Пятін // Вісник Хмельницького національного університету. - 2020. - №4. - С. 174-183.

15. Boiko J. Design and Performance Evaluation of Channel Coding Schemes for Information Technologies / J. Boiko, I. Pyatin, R. Chorny, T. Davydova // 2021 IEEE 3rd International Conference on Advanced Trends in Information Theory (ATIT), Kyiv 15-17 Dec. 2021, Kyiv – 2021. - pp. 195-200.

16. Zhurakovskiy B. Calculation of Quality Indicators of the Future Multiservice Network / B. Zhurakovskiy, S. Toliupa, V. Druzhynin, A. Bondarchuk, M. Stepanov // Future Intent-Based Networking. – Springer, Cham,

2022. – С. 197-209.

17. Семенко А.І. Синтез широкопasmової телекомунікаційної системи з підвищеною конфіденційністю передачі інформації шляхом використання псевдовипадкових послідовностей на основі хаосу / А.І. Семенко, М. Я. Кушнір, Н. І. Бокла //Вісник університету «Україна». Серія: інформатика, обчислювальна техніка та кібернетика.- 2020. - №1(24), - С .65-75.

18. T. Palenik High throughput open-source implementation of Wi-Fi 6 and WiMAX LDPC encoder and decoder /Tomas Palenik, Viktor Szitkey. - International Journal Information Technology Applications (ITA). - Volume 11, Number 2, December 2022. arXiv:2306.12063.

19. Пятін І.С. Аналіз завадостійкості захищеної системи зв'язку 5G з полярним кодуванням /І.С. Пятін, Ю.М. Бойко, В.Є Гавронський, Д.В. Панасюк //ВОТТП № 3, 2023 С. 154-163

20. Li, Z., Chen, L., Zeng, L., et al. Efficient encoding of quasi-cyclic low-density parity-check codes. IEEE Transactions on Communications, 2006, vol. 54, no. 1, pp. 71-81.

21. Farkas, P., Rakus, M. Decoding five times extended reed solomon codes using syndromes. Computing and Informatics, 2020, no. 6, vol 39, pp.1311–1335. ISSN 2585-8807 (online).

22. Sowmya, G., Keerthi, K., Lalitkrushna, J. T., et al. An architecture for efficient encoding of quasi cyclic LDPC codes and its implementation in FPGA. In IEEE 11th International Conference on Communication Systems and Network Technologies (CSNT), Indore (India), 2022, pp. 136-140.

23. Liu, J., Feng, Q. A Miniaturized LDPC Encoder: Two-layer architecture for CCSDS near-earth standard. IEEE Transactions on Circuits and Systems II: Express Briefs, 2021, vol. 68, no. 7, p. 2384–2388.

24. Kang, J., Wang, B., Zhang, Y., An, J. Enhanced partially-parallel LDPC decoder for near earth applications. In IEEE 11th International Conference on Electronics Information and Emergency Communication (ICEIEC), Beijing (China), 2021, p. 12-16.

25. Rajagopalan, P., et al. Performance analysis of LDPC decoding algorithms for CCSDS telecommand space data link protocol. In 2nd International Conference for Emerging Technology (INCET), Belagavi (India), 2021, p. 1-5.
26. Rakus, M., Farkas, P. On possible energy savings with transmission supported via feedback channel in CubeSat transceiver. In Journal of Electrical Engineering, 2021, vol.72, no. 5, p. 337–342.
27. Geiselhart, M., Ebada, M., Elkelesh, A. et al., Automorphism ensemble decoding of quasi-cyclic LDPC codes by breaking graph symmetries. IEEE Communications Letters, 2022, vol. 26, no. 8, p. 1705-1709.
28. Nguyen, J., Wang, L., Hulse, C. et al., Neural normalized min-sum message-passing vs. viterbi decoding for the CCSDS line product code. In ICC 2022 - IEEE International Conference on Communications. Seoul (Korea), 2022, p. 2375–2380.
- 29 Farkas, P., Rakus, M. Adding RLL properties to four CCSDS LDPC codes without increasing their redundancy. Accepted for publication in Computing and Informatics, 2023, ISSN 1335-9150
30. Smarandache, R., Mitchell, D. G. M. A unifying framework to construct QC-LDPC Tanner graphs of desired girth. IEEE Transactions on Information Theory, 2022, vol. 68, no. 9, p. 5802-5822.
31. Jung, Y., Chung, C., Kim J., Jung, Y. 7.7Gbps encoder design for IEEE 802.11n/ac QC-LDPC codes. In International SoC Design Conference (ISOCC), Jeju (Island), 2012, pp. 215–218.
32. Mahdi A., Paliouras V. A low complexity-high throughput QC-LDPC encoder. IEEE Transactions on Signal Processing, 2014, vol. 62, no. 10, p. 2696–2708.
33. Kun, C., QI, Shengkai, L., Chengzhi, P. Implementation of encoder and decoder for LDPC codes based on FPGA. Journal of Systems Engineering and Electronics, 2019, vol. 30, no. 4, p. 642–650.
34. Nguyen, T., Nguyen, T., Lee, H. Efficient QC-LDPC encoder for 5G new radio. Electronics.2019, vol.8, no. 6, p.668, ISSN: 2079-9292.

35. Yao, X., Li, L., Liu, J., Li Q. A low complexity parallel QC-LDPC encoder. In IEEE MTT-S International Wireless Symposium (IWS). Nanjing (China), 2021, p. 1–3.

36. Le Gal, B., Jego, C. High-throughput multi-core LDPC decoders based on x86 processor. IEEE Transactions on Parallel and Distributed Systems, 2016, vol. 27, no. 5, p. 1373–1386.

Хмельницький національний університет  
Факультет інформаційних технологій  
Кафедра телекомунікацій, медійних та  
інтелектуальних технологій



## Дипломна робота

# Метод підвищення пропускної здатності безпроводних локальних мереж стандарту IEEE 802.11

Спеціальність 172 – Телекомунікації та радіотехніка

Виконав: Панасюк Д.В., гр. ТРМ-22-1

Керівник – д.т.н., проф. Бойко Ю.М.

**Мета і задачі дослідження.** Метою роботи є дослідження методів підвищення пропускної здатності безпроводних локальних мереж стандарту IEEE 802.11, моделювання системи обробки сигналів у процесорі основної смуги частот передавача і приймача.

Для досягнення цієї мети поставлені наступні завдання.

- провести огляд шляхів підвищення пропускної здатності безпроводних локальних мереж;
- дослідити модель системи безпроводного зв'язку;
- дослідити коефіцієнт бітових помилок від відношення сигнал-шум для системи безпроводного зв'язку;

**Об'єктом дослідження** є процес передачі сигналів у безпроводних локальних мереж стандарту IEEE 802.11.

**Предметом дослідження** є методи та засоби підвищення пропускної здатності безпроводного зв'язку стандарту IEEE 802.11.

## **Наукова новизна одержаних результатів :**

1. Набули подальшого розвитку методи підвищення пропускної здатності безпроводних локальних мереж.
2. Набули подальшого розвитку методи декодування LDPC кодів з різною кодовою швидкістю.

## **Практичне значення одержаних результатів:**

Проведене дослідження завадостійкості кодів LDPC різної кодової швидкості, сигнально-кодових конструкцій з різною швидкістю передавання інформації. Досліджені залежності коефіцієнту бітових помилок від відношення сигнал-шум для системи зв'язку з цифровою квадратурною модуляцією і кодуванням LDPC, побудований декодер LDPC за алгоритмом нормалізованої мінімальної суми з використанням HDL.

## Структурна схема системи зв'язку

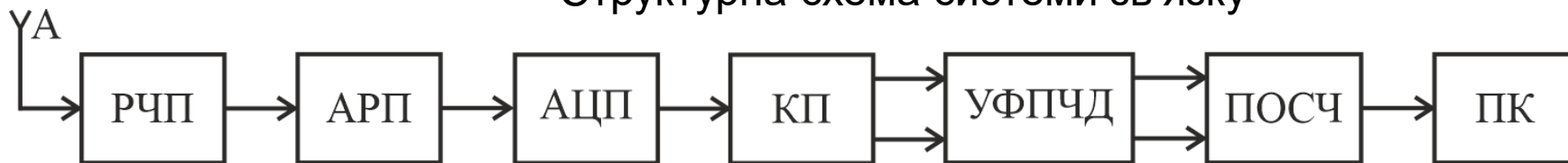


Рисунок 4.1 – SDR архітектура приймача(РЧП – радіочастотний приймач; АРП – автоматичне регулювання підсилення; АЦП – аналого-цифровий перетворювач; КП – квадратурний перетворювач; УФПЧД – узгоджений фільтр з пониженням частоти дискретизації; ПОСЧ – процесор основної смуги частот; ПК – персональний комп'ютер)

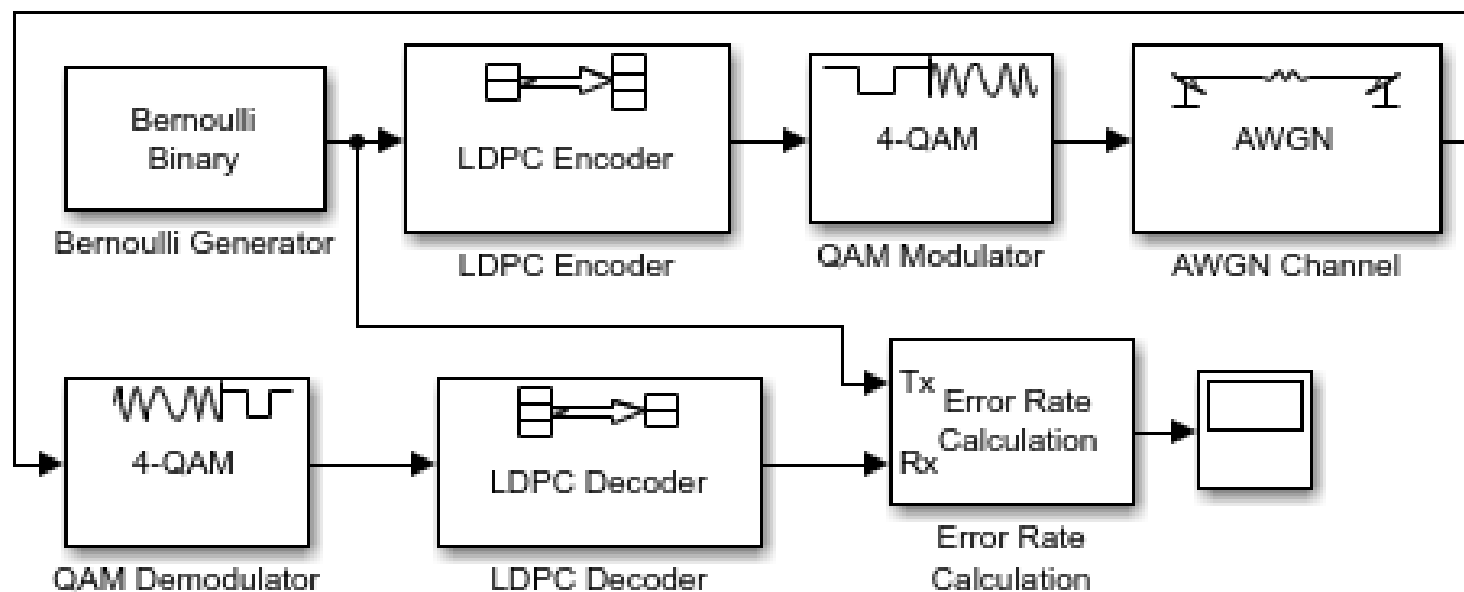


Рисунок 4.2 – Simulink модель дослідження ефективності обробки сигналів у процесорах основної смуги частот передавача і приймача

# Коди LDPC

$$H = \begin{pmatrix} 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

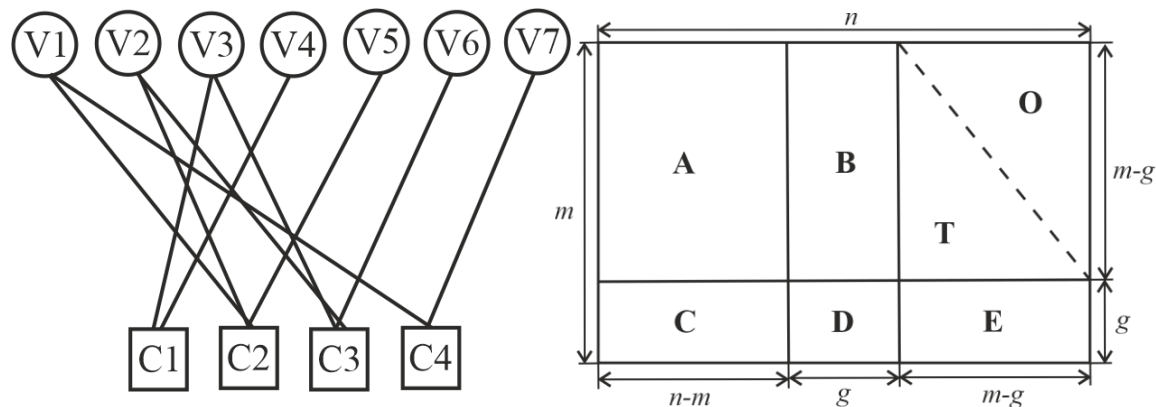


Рисунок 5.1 - Приклад матриці перевірки на парність (а)  
відповідний граф Таннера (б)

Рисунок 5.2 - Матриця перевірки парності, подана у нижній псевдотрикутній формі

## Реалізація складових LDPC декодера на FPGA

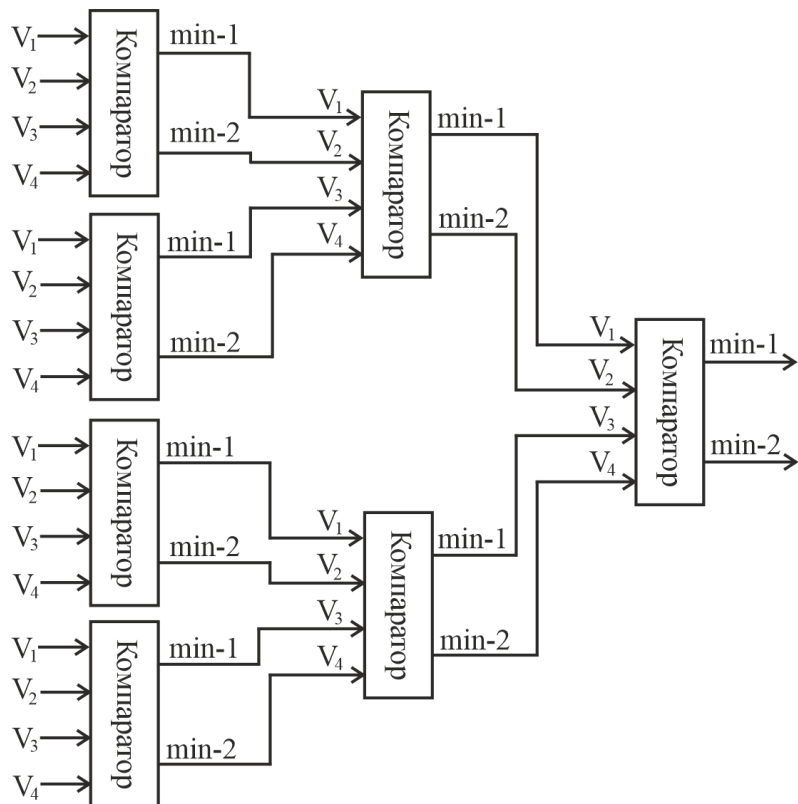


Рисунок 6.1 - Трирівневе дерево компараторів для знаходження першого і другого мінімумів

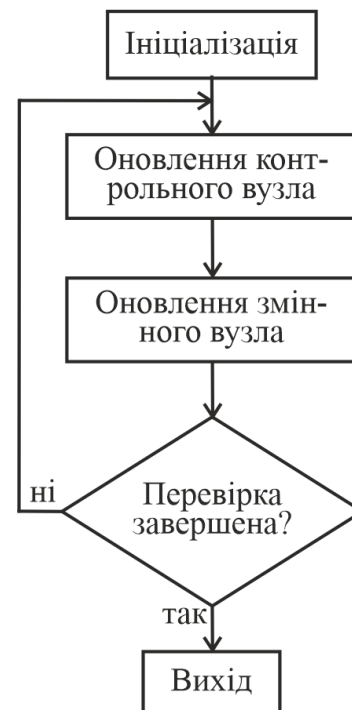


Рисунок 6.2 - Схема алгоритму нормалізованої мінімальної суми

# Схеми для оновлення вузлів змінних та вузлів перевірки

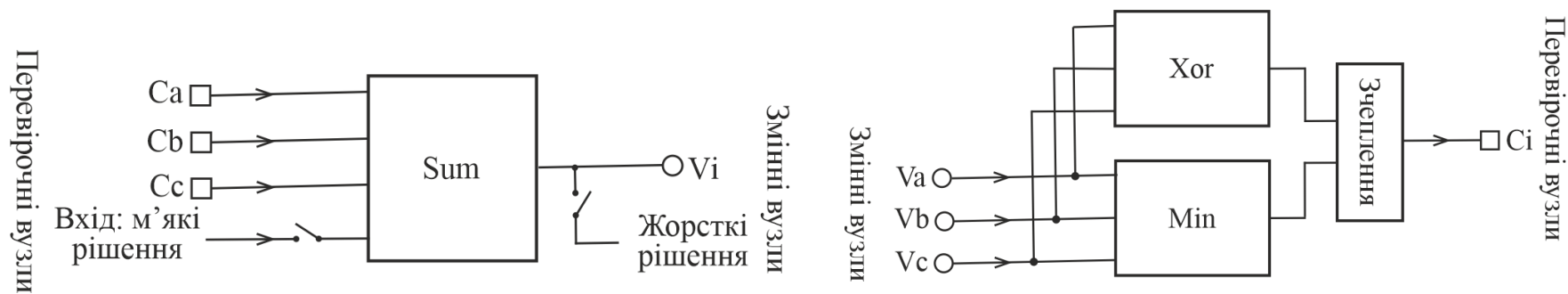


Рисунок 7.1 - Схема оновлення змінних вузлів

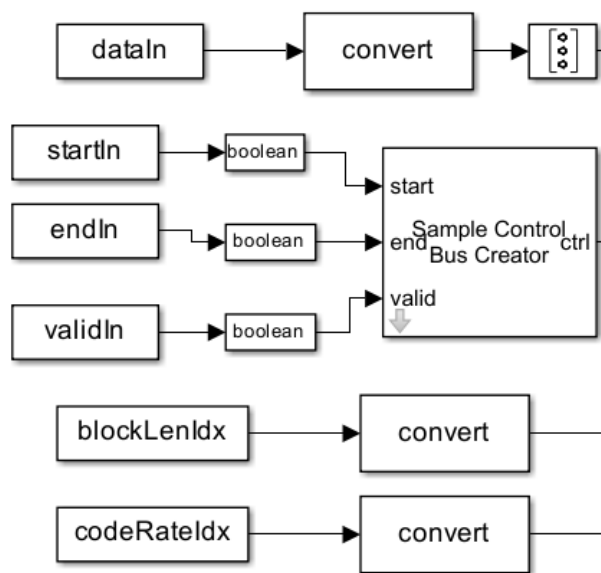
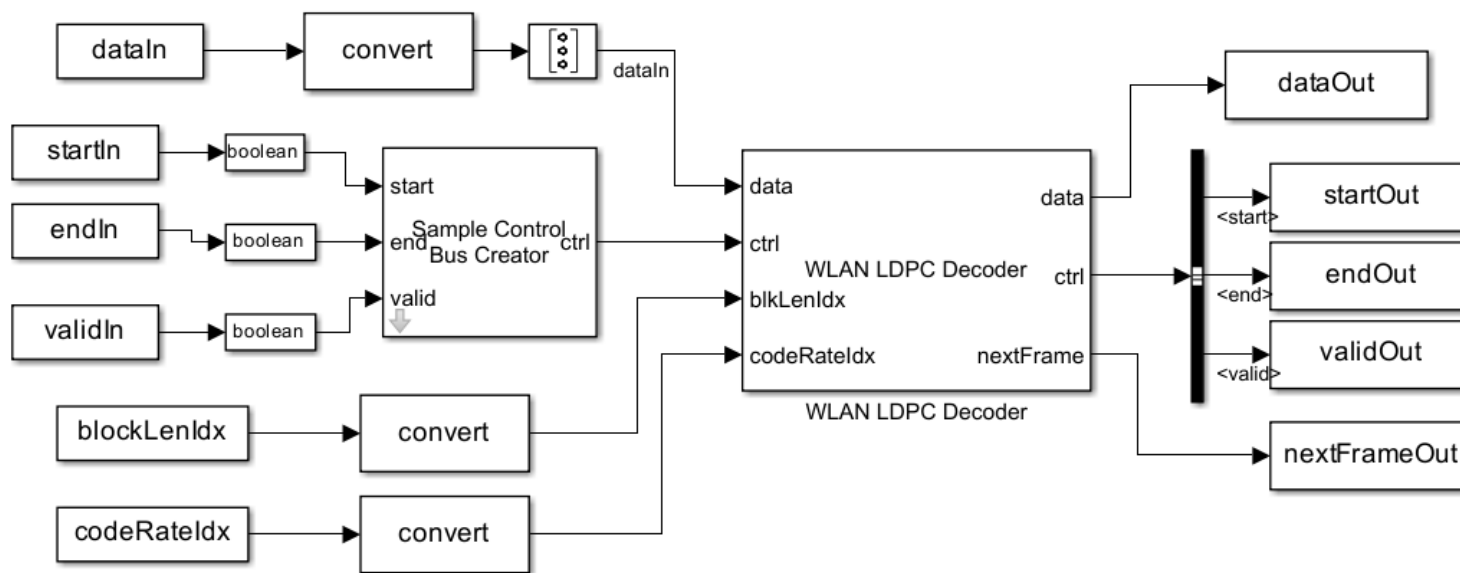
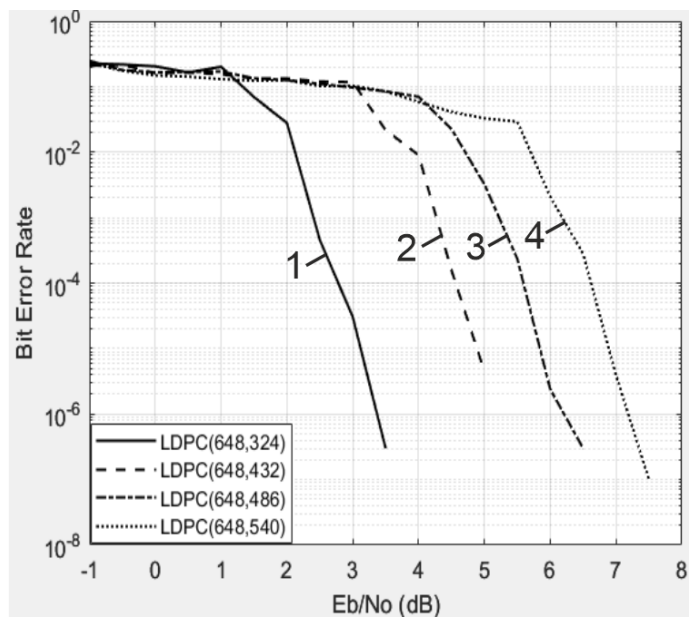


Рисунок 7.2 - Схема оновлення перевірочних вузлів

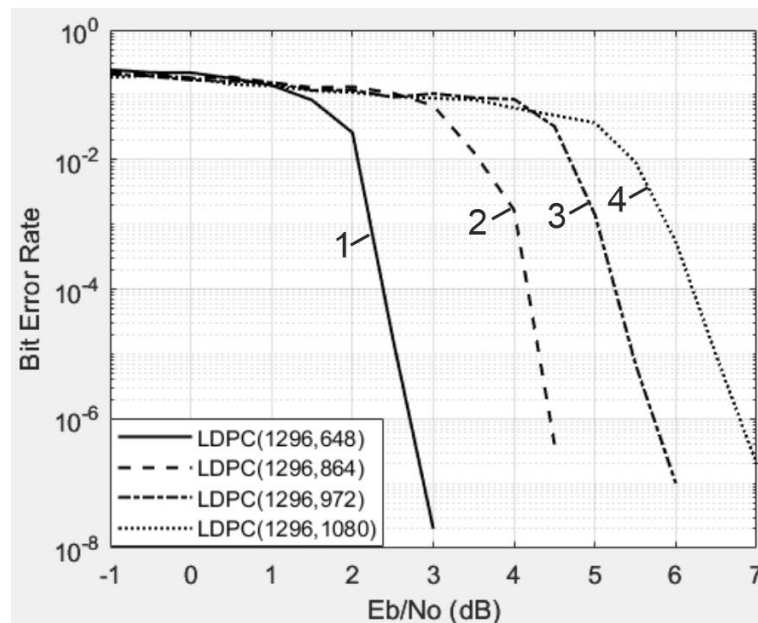
Рисунок 7.3 - Simulink модель LDPC декодера



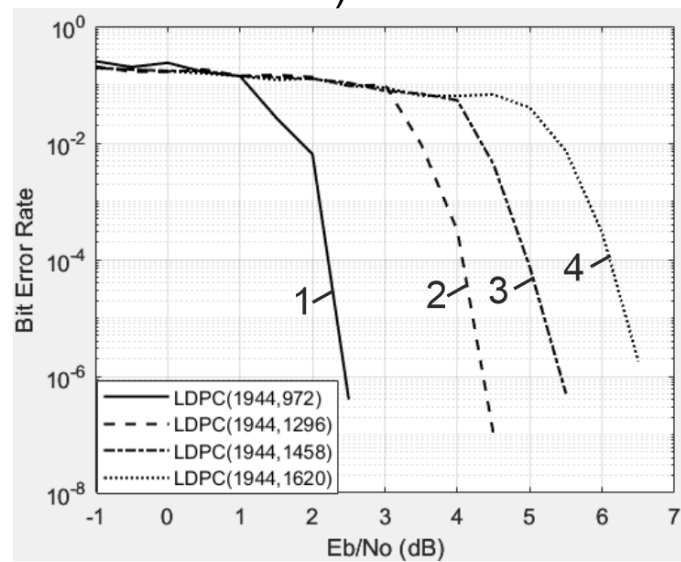
# Різна кодова швидкість



а)



б)



в)

Рисунок 8.1 - Залежність BER від SNR для системи зв'язку з кодовою швидкістю: 1 - 1/2; 2 - 2/3; 3 - 3/4; 4 - 5/6. Довжина блоку: а – 648 біта; б – 1296 біта; в - 1944 біта

# Використання багатопозиційної модуляції

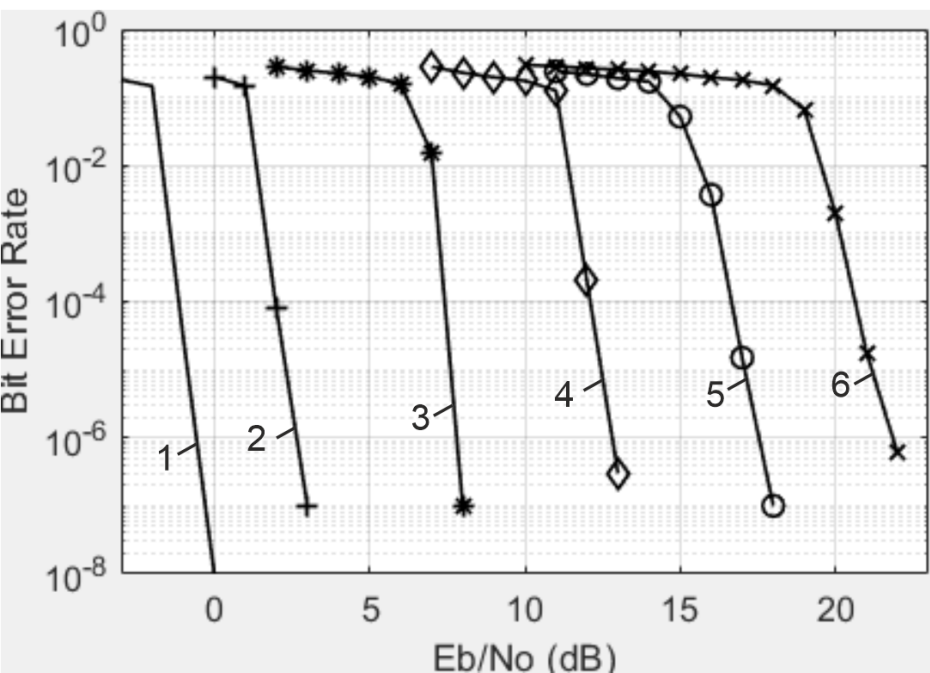


Рисунок 9.1 - Залежність BER від SNR для системи зв'язку з кодуванням LDPC, довжина блоку 1944, кодова швидкість 1/2. Модуляція: 1 – BPSK; 2 – QPSK; 3 – 16-QAM; 4 – 64-QAM; 5 – 256-QAM; 6 – 1024-QAM

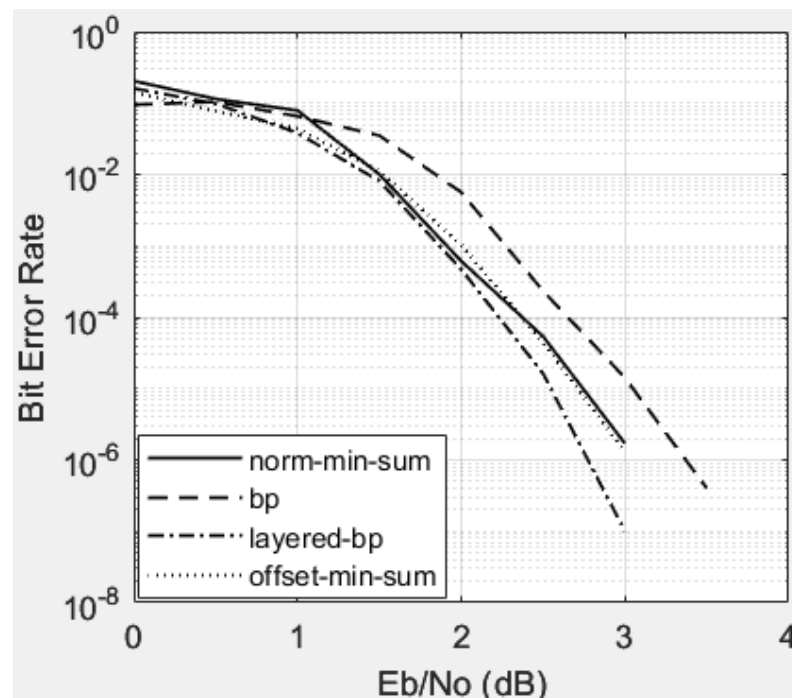


Рисунок 9.2 - Залежність BER від SNR для системи зв'язку з кодуванням LDPC(648, 324), довжина блоку 648, кодова швидкість 1/2.

# Близькість коду до межі Шеннона

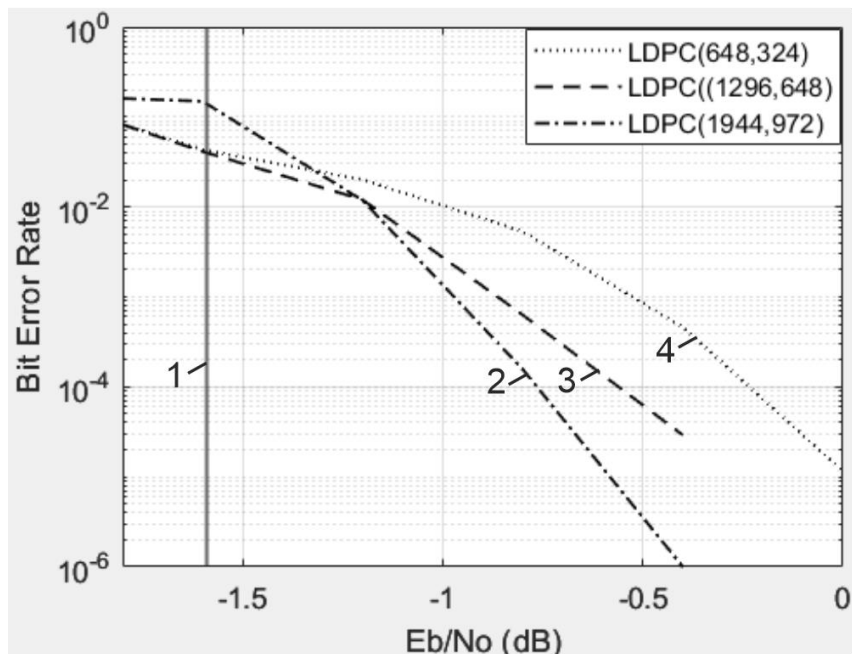


Рисунок 10.1 - Залежність BER від SNR для системи зв'язку з кодуванням LDPC, кодова швидкість 1/2. Довжина блоку: 2 – 1944; 3 – 1296; 4 – 648.  
1 – межа Шеннона

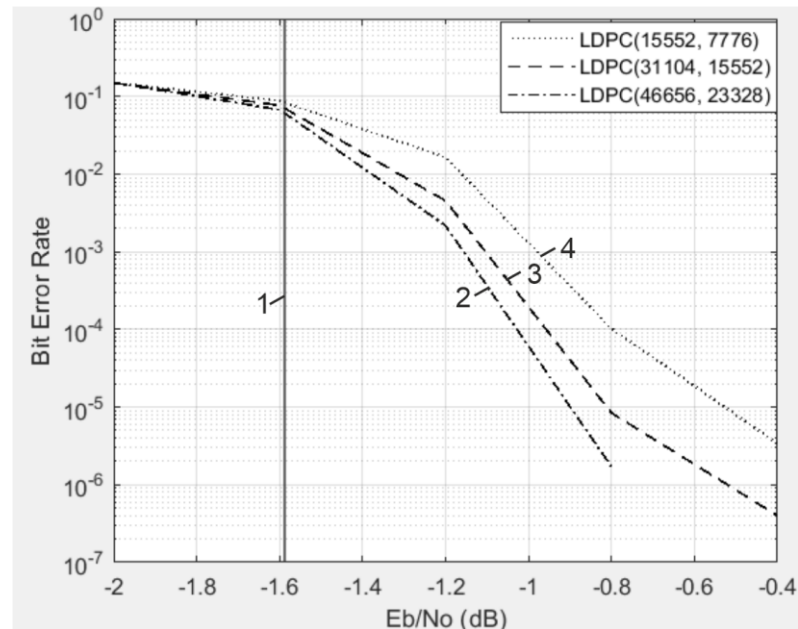


Рисунок 10.1 - Залежність BER від SNR для системи зв'язку з кодуванням LDPC, кодова швидкість 1/2. Довжина блоку: 2 – 46656; 3 – 31104; 4 – 15552.  
1 – межа Шеннона

1. При збільшенні швидкості коду зменшується завадостійкість системи зв'язку. При збільшенні кодової швидкості з  $1/2$  до  $5/6$  необхідно підвищувати відношення сигнал-шум на 4 дБ.
2. Використання багатопозиційної модуляції потребує підвищення відношення сигнал-шум: від 1 (BPSK) до 2 (QPSK) біт на символ модуляції – на 3 дБ; від 2 (QPSK) до 4 (16-QAM) біт на символ модуляції – на 5 дБ. Подальше збільшення позиційності модуляції до 6, 8 і 10 біт на символ – підвищення відношення сигнал-шум на 5 дБ на кожні 2 біта на символ.
3. Код LDPC для стандарту 802.11ax, LDPC(1944,972) – з найбільшою довжиною блока даних, наближається до межі Шеннона на 1,2дБ.
4. Зі збільшенням швидкості передачі даних велике значення набуває величина вектора помилки (EVM) на радіочастотному інтерфейсі. При 1024 підносійних Wi-Fi 6 (802.11ax) сузір'я стає дуже щільним і системі важко відрізнити одну з точок від іншої. Для декодування (або демодуляції) цих точок сузір'я потрібно мати кращу EVM.
5. Стандарт IEEE 802.11ax використовує OFDMA (Orthogonal Frequency Division Multiple Access), або множинний доступ з ортогональним частотним поділом, який дозволяє блокам ресурсів (RU) надавати кільком абонентам однакову швидкість передачі даних.
6. У стандарті Wi-Fi 6 впроваджено технологію MU-MIMO, що дозволяє точці доступу передавати дані кільком терміналам одночасно, що значно підвищує пропускну спроможність.

<https://doi.org/10.31891/2219-9365-2023-75-18>

УДК 621.396.969.1

ПЯТИН Ілля

Хмельницький політехнічний фаховий коледж національного університету «Львівська політехніка»

<https://orcid.org/0000-0003-1898-6755>

e-mail: [ilkhmel@ukr.net](mailto:ilkhmel@ukr.net)

БОЙКО Юлій

Хмельницький національний університет

<https://orcid.org/0000-0003-0603-7827>

e-mail: [boiko\\_julius@ukr.net](mailto:boiko_julius@ukr.net)

ГАВРОНСЬКИЙ Віталій

Хмельницький політехнічний фаховий коледж національного університету «Львівська політехніка»

<https://orcid.org/0000-0002-1529-1272>

e-mail: [gavrinskiy@gmail.com](mailto:gavrinskiy@gmail.com)

ПАНАСЮК Дмитро

Хмельницький національний університет

e-mail: [panasyuk-d@i.ua](mailto:panasyuk-d@i.ua)

## АНАЛІЗ ЗАВАДОСТІЙКОСТІ ЗАХИЩЕНОЇ СИСТЕМИ ЗВ'ЯЗКУ 5G З ПОЛЯРНИМ КОДУВАННЯМ

Передача інформації у мобільних телекомунікаціях через канал зв'язку супроводжується виникненням помилок. Полярні коди забезпечують гарне виправлення помилок при низькій складності декодування для використовуваних на практиці довжин блоків у поєднанні з більш досконалими алгоритмами декодування. Полярне кодування здійснює операцію поляризації каналу зв'язку. Метою роботи є дослідження системи зв'язку з декодером послідовного виключення, реалізація полярного кодера і декодера на FPGA System-on-Chip (SoC) Intel DE10-Standard Development Kit. Проведене дослідження кількості бітових помилок від відношення сигнал-шум для системи зв'язку з модуляцією BPSK і різною довжиною кодового блоку, циклічним надлишковим кодом і реверсуванням бітів, різним розміром списку послідовного виключення і різною кодовою швидкістю.

Ключові слова: система зв'язку, полярний кодер, поляризація каналу, декодер послідовного виключення, коефіцієнт бітових помилок, 5G.

PYATIN Ilya

Khmelnytskyi Polytechnic, Professional College by Lviv Polytechnic National University

BOIKO Juliy

Khmelnytskyi National University

GAVRONSKIY Vitaliy

Khmelnytskyi Polytechnic, Professional College by Lviv Polytechnic National University

PANASYUK Dmytro

Khmelnytskyi National University

## NOISE IMMUNITY ANALYSIS OF PROTECTED 5G COMMUNICATION SYSTEM WITH POLAR CODING

The transmission of information in mobile telecommunications through a communication channel is accompanied by the occurrence of errors. Polar codes are a family of codes that achieve the bandwidth of symmetric channels without memory. Polar codes provide good error correction at low decoding complexity for practical block lengths when combined with more sophisticated decoding algorithms. This made it possible to use polar codes in the 5G wireless communication standard. The purpose of the work is to study the communication system with the serial exclusion decoder, the implementation of this decoder on the FPGA, and the determination of the bit error rate using mathematical modeling in the MATLAB environment. Polar coding performs the operation of polarization of the communication channel. Channel polarization is an operation by which a polarized set of  $N$  channels is created from  $N$  independent copies of a given channel  $W$ , the bandwidth of which tends to zero or unity. The basic algorithm of polar decoding is a sequential elimination algorithm. The recovered data must be sorted similarly to the input data, so the encoder input and decoder output must mirror each other. Each stage consists of  $N/2$  nodes  $F$  and  $G$  connected together in a structure resembling a fast Fourier transform. Node  $G$  is a bitwise conditional adder/subtractor representing the modulo 2 partial sums of the previously evaluated bits. The rule for calculating partial sums is based on the structure that copies the corresponding polar coder. The sequential elimination decoding algorithm consists of two tasks, the first of which runs sequentially from the input stage to the output stage to compute the final LLR and make a hard decision. The second task recursively executes the predecessor nodes to backpropagate the decoded bits from the decoder output to the corresponding predecessor stage. The polar encoder and decoder are implemented on the FPGA System-on-Chip (SoC) Intel DE10-Standard Development Kit. A study of the number of bit error rate from the signal-to-noise ratio for a communication system with BPSK modulation and different code block lengths, cyclic redundant code and bit reversal, different sizes of the sequential exclusion list and different code rates was carried out.

Keywords: communication system, polar coder, channel polarization, successive cancellation decoder, bit error rate, 5G

**Постановка проблеми у загальному вигляді  
та її зв'язок із важливими науковими чи практичними завданнями**

Передача інформації у мобільних телекомунікаціях через канал зв'язку супроводжується виникненням помилок. Якщо приймач виправляє їх без звернення до передавача, то використовується схема прямої корекції помилок. Полярні коди дозволяють збільшити пропускну здатність системи зв'язку і мають спрощені процедури кодування і декодування. Алгоритм послідовного виключення (Successive Cancellation – SC) є одним з алгоритмів декодування, що найчастіше використовуються, через його низьку складність. Він має широкі можливості для удосконалення апаратної архітектури. Для полярного коду компроміс між великою затримкою та ресурсами FPGA (Field Programmable Gate Array) є вузьким місцем розробки полярного декодера з високою пропускну здатністю.

Метою роботи є дослідження системи зв'язку з декодером послідовного виключення, реалізація цього декодера на FPGA, визначення коефіцієнта бітових помилок за допомогою математичного моделювання у середовищі МАТЛАБ.

**Аналіз досліджень та публікацій**

Канали зв'язку схильні до помилок через випадковий шум, завади, дефекти пристроїв тощо, які спотворюють початковий потік даних на приймачі. Полярні коди – це сімейство кодів з виправленням помилок, які досягають пропускну спроможності симетричних каналів без пам'яті. Полярні коди забезпечують гарне виправлення помилок при низькій складності декодування для використовуваних на практиці довжин блоків у поєднанні з більш досконалішими алгоритмами декодування. Ці характеристики дали змогу використовувати полярні коди у стандарті бездротового зв'язку 5G.

Використовують декодер послідовного виключення (SC), який забезпечує оцінку кожного біта повідомлення по порядку, від 1 до  $N$  з використанням операції прийняття рішення. Хоча асимптотична продуктивність полярних кодів при SC-декодуванні відповідає пропускну здатності каналу зв'язку, продуктивність кодів кінцевої довжини незадовільна. Декодування списку послідовного виключення (SCL) [1-3] зменшує кількість бітових помилок. Також використовують поєднання полярних кодів (ПК) з іншими кодами, таким як коди Cyclic Redundancy Check (CRC) [4]. Також відомий рекурсивний алгоритм декодування, що базується на використанні параметрів Бгаттачар'ї.

Складність коду визначає кількість енергії, що споживає декодер, об'єм пам'яті, обчислювальну потужність і затримку. Канальне кодування використовує набір алгоритмічних операцій над початковим потоком даних у передавачі та інший набір операцій над прийнятим потоком даних у приймачі для виправлення помилок. Мета досліджень канального кодування: розробити високопродуктивний канальний код, який зменшує вплив помилок у каналі зв'язку. Актуальною проблемою канального кодування є розробка декодера з низькою складністю і затримкою, що забезпечує зниження вартості і підвищення швидкості обробки інформації.

**Полярний кодер**

При довжині коду  $N=2^n$  та  $K$  інформаційних бітах вхід кодера складається з  $K$  інформаційних бітів та  $N - K$  заморожених бітів. Кодове слово  $x$  можна отримати за виразом:

$$x = u \cdot G_N,$$

де  $G_N = B_N \cdot F_2^{\otimes n}$  - генераторна матриця;  $B_N$  - матриця перестановок,  $F_2^{\otimes n}$  -  $n$ -а степінь Кронекера,

$$F_2 = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}$$

Генераторна матриця для  $N = 8$  зображена на рис. 1: Схема кодування, що відповідає наведеній генераторній матриці, приведена на рис. 2.

$$G_8 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$

Рис. 1. Генераторна матриця для  $N = 8$

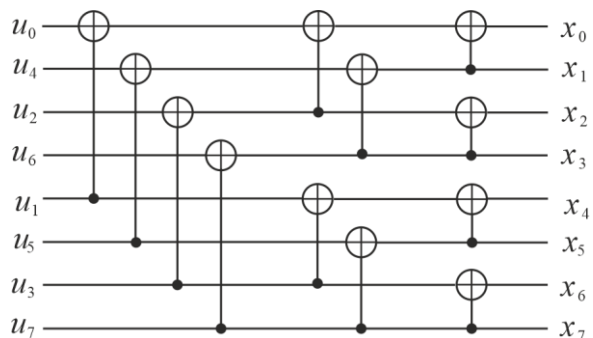


Рис. 2. Полярний кодер для  $N=8$

Полярний кодер реалізує алгоритм, який вибирає  $K$  найкращих серед  $N$  можливих поляризованих бітових каналів.

### Поляризація каналу

Розглянемо метод поляризації каналу, що використовують для побудови кодів, які досягають симетричної пропускної здатності  $I(W)$  дискретного каналу без пам'яті з двійковим входом (BDMC)  $W$  [4].

$W^N$  позначає канал, що відповідає  $N$  використанням каналу  $W$ ; таким чином,  $W^N : X^N \rightarrow Y^N$  з  $W^N(y_1^N | x_1^N) = \prod_{i=1}^N W(x_i, y_i)$ .

Для каналу B-DMC (binary-input discrete memoryless channel)  $W$  є два параметри, що часто використовуються [4]: симетрична пропускна здатність, що визначається виразом:

$$I(W) = \sum_{y \in Y} \sum_{x \in X} \frac{1}{2} W(y|x) \log_2 \frac{W(y|x)}{0,5[W(y|0) + W(y|1)]},$$

де  $W(y|x)$  - перехідні ймовірності між виходом ( $y$ ) і входом ( $x$ ), та параметр Бгаттачар'ї:

$$Z(W) = \sum_{y \in Y} \sqrt{W(y|0)W(y|1)}.$$

Ці параметри використовуються як міри швидкості та надійності відповідно.  $I(W)$  - це найвища швидкість, при якій можливий надійний зв'язок через  $W$ .  $Z(W)$  — це верхня межа ймовірності помилки рішення з максимальною правдоподібністю (ML).  $Z(W)$  та  $I(W)$  набувають значення  $[0, 1]$ .

Двійковий симетричний канал (BSC) та двійковий канал стирання (BEC) є прикладами симетричних каналів. Прикладом дії BEC на послідовність бітів приведена на рис. 3

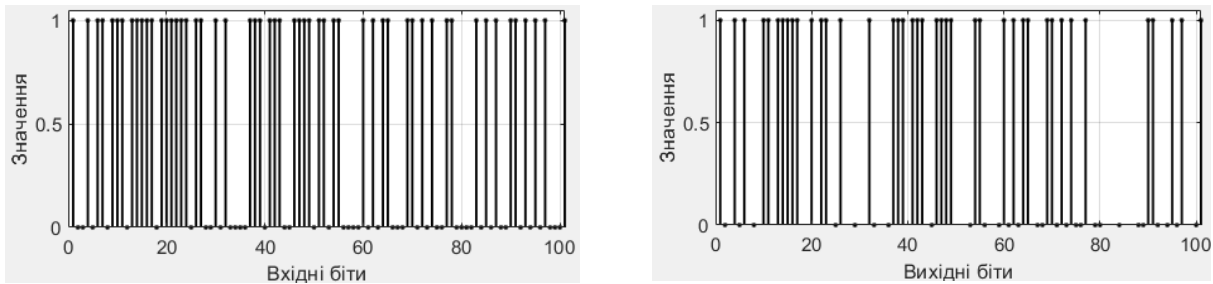


Рис. 3. Приклад дії двійкового каналу стирання (BEC) на послідовність бітів

Пропускна здатність  $I(W) \approx 1$  у випадку  $Z(W) \approx 0$ , і  $I(W) \approx 0$  у випадку  $Z(W) \approx 1$  [4].

$$I(W) \geq \log_2 \frac{2}{1 + Z(W)}, \quad I(W) \leq \sqrt{1 - Z(W)^2} \quad (1)$$

Приклад визначення пропускної здатності каналів для  $N=8$  приведено на рис. 4.

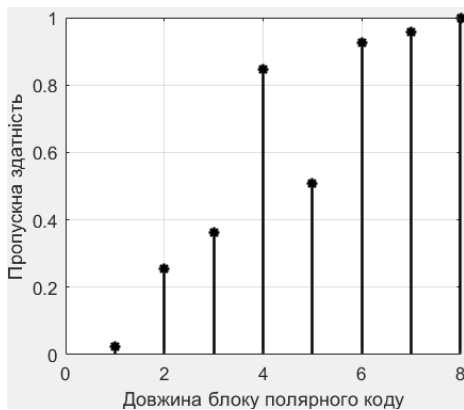


Рис. 4. Пропускна здатність за виразом (1) для  $N=8$

Наприклад, для довжини коду  $N=8$  і кодовій швидкості  $R=1/2$  маємо:

- кількість інформаційних біт  $K=N \cdot R=4$ ;
- кількість заморожених бітів  $F=N-K=4$ ;
- позиції інформаційних бітів  $A_c=(5,3,2,1)$ ;
- позиції заморожених бітів  $A=(8,7,6,4)$ .

Ідея кодування полярними кодами полягає у поділі індексів відсортованої за зростанням послідовності даних на два різні набори перед передачею. У перший набір входять індекси даних, що підлягають передачі каналами без завад. Інший набір включає індекси, що відповідають відомим замороженим бітам, які передаються по каналам з чистим шумом.

Двійковий симетричний канал (BSC) та двійковий канал стирання (BEC) є прикладами симетричних каналів. Поляризація каналу - це операція, за допомогою якої з  $N$  незалежних копій даного B-DMC каналу  $W$  створюється другий набір  $N$  каналів  $\{W_N^{(i)} : 1 \leq i \leq N\}$ , що демонструють ефект поляризації в тому сенсі, що коли  $N$  стає великим, симетричні члени пропускної спроможності  $I(W_N^{(i)})$  прагнуть до 0 або 1 для всіх індексів. Ця операція складається з фази об'єднання каналів та фази поділу каналів.

Ефект поляризації показано на рис. 5 для випадку, коли  $W$  є BEC із ймовірністю стирання  $\epsilon = 0,5$ .

На рис. 5 показано, що  $I(W^{(i)})$  має тенденцію бути близькою до 0 для малих  $i$  та до 1 для великих  $i$ .

На рис. 6 показано бінарне дерево поляризації каналів [4].

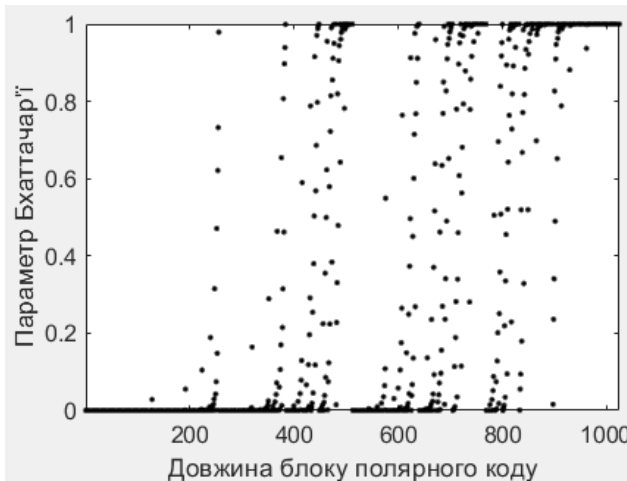


Рис. 5. Поляризація каналу

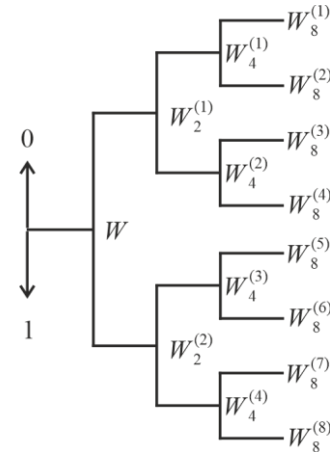


Рис. 6. Деревоподібний процес побудови рекурсивного каналу.

Кореневий вузол дерева пов'язаний з каналом  $W$ . Корінь  $W$  породжує верхній канал  $W_2^{(1)}$  і нижній канал  $W_2^{(2)}$ , який пов'язані з двома вузлами на рівні 1. Канал  $W_2^{(1)}$ , своєю чергою, породжує канали  $W_4^{(1)}$  і  $W_4^{(2)}$  і так далі. Канал  $W_2^{(i)}$  розташований на рівні  $n$  дерева у вузлі номер  $i$ , рахуючи зверху.

Верхня межа параметра Бхаттачар'я відповідає каналу з найбільшим рівнем шуму, а нижня межа - каналу з найнижчим рівнем шуму. Таким чином, для кращої роботи потрібно збільшити розрив між екстремумами параметра Бхаттачар'я. На боці передавача  $K$  інформаційних бітів вставляються у відповідні безшумні канали, а  $N-K$  заморожених бітів вставляються у відповідні канали з чистим шумом для створення вхідного вектора  $u_1^N$ , який має бути переданий каналом  $W$ .

Параметр Бхаттачар'я використовується як міра надійності, де він є верхньою межею ймовірності помилки рішення з максимальною ймовірністю для двійкового каналу зв'язку.

### Полярний декодер

Базовий алгоритм полярного декодування - це алгоритм послідовного виключення (SC). Відомі ще алгоритми декодування з більш високою продуктивністю для відносно коротких кодових слів, але з більшою складністю, такі як алгоритм декодування списку послідовного виключення (SCL), алгоритм декодування списку послідовного виключення за допомогою CRC (CA-SCL) [4] та алгоритм поширення переконань (PB) [5]. Алгоритм поширення переконань добре відомий своїм застосуванням для LDPC декодування, в якому між вузлами відбувається обмін м'якими повідомленнями. З іншого боку, алгоритм SCL зберігає список рішень та вибирає найкраще рішення, використовуючи сортування. Обидва вони мають кращу продуктивність, особливо алгоритм SCL забезпечує найкраще підсилення BER для полярних кодів із короткою довжиною блоку. Ми зосередимося на алгоритмі декодування SC через його простоту.

На рис. 7 наведена FPGA реалізація декодера послідовного виключення.

Відновлені дані повинні бути відсортовані аналогічно до вхідних даних, тому вхід кодера і вихід декодера повинні відображати один одного. Отже, кодер повинен мати «великих» метеликів на вході, а декодер - «великих» метеликів на виході. Кожен етап складається з  $N/2$  вузлів  $F$  і  $G$ , з'єднаних між собою у структуру, що нагадує швидке перетворення Фур'є (ШПФ). Ці два вузли реалізують дві основні функції,  $f(a,b)$  і  $g(\hat{u}_s, a,b)$ , де  $a$  і  $b$  - відношення правдоподібності:

$$f(a,b) = \frac{1+ab}{a+b} \quad (2)$$

$$g(\hat{u}_s, a, b) = a^{1-2\hat{u}_s} b \quad (3)$$

Оціночні значення розраховуються з використанням наступного виразу:

$$\hat{u}_i = \begin{cases} 0, & \frac{\Pr(y|\hat{u}_0^{i-1}, u_i=0)}{\Pr(y|\hat{u}_0^{i-1}, u_i=1)} \geq 1 \\ 1, & \text{інакше} \end{cases}$$

де  $y$  – вихід із каналу.

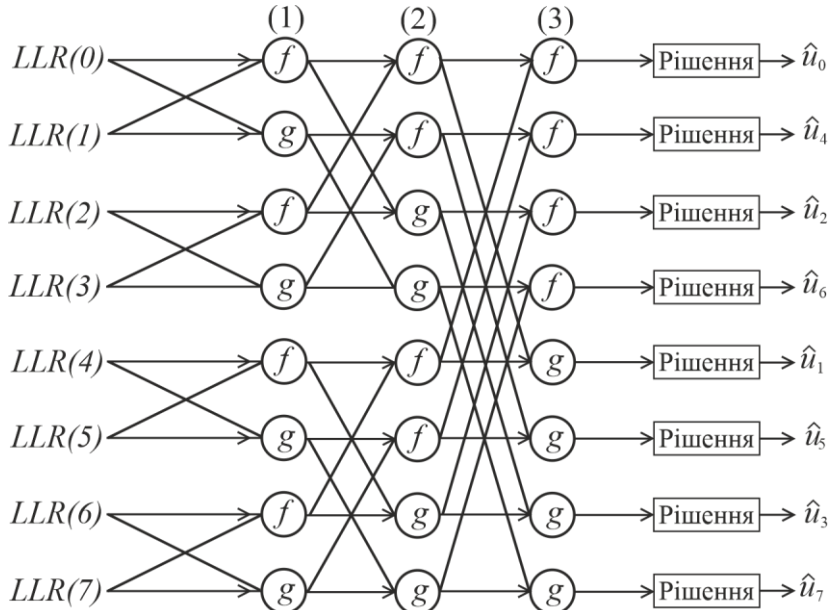


Рис. 7. Декодер полярного коду,  $N=8$

Щоб реалізувати ці два вузли апаратно, (2) і (3) перетворюються у логарифмічній області до:

$$f(a,b) = 2\text{arth} \left[ \text{th} \left( \frac{a}{2} \right) \cdot \text{th} \left( \frac{b}{2} \right) \right], \quad (4)$$

$$g(\hat{u}_s, a, b) = (1 - 2\hat{u}_s) a + b, \quad (5)$$

де  $a, b$  - логарифмічні відношення правдоподібності (LLR). Рівняння (4) можна замінити наближенням мінімальної суми, що описується виразом:

$$f(a,b) = \min(|a|, |b|). \quad (6)$$

Це наближення використовується для розрахунку контрольного вузла для коду LDPC. Вузол  $G$ , який реалізує (5), насправді є умовним додаванням/відніманням, що залежить від значення біта рішення  $\hat{u}_s$  де  $\hat{u}_s$  є бітом, що представляє часткову суму за модулем 2 раніше оцінених бітів. Правило обчислення часткових сум  $\hat{u}_s$  засноване на структурі, що копіює відповідний полярний кодер. У середині DFG декодера «захований» DFG кодер. Наприклад, часткова сума  $\hat{u}_s$  для вузла  $G_{1,2}$  дорівнює  $\hat{u}_2 \otimes \hat{u}_3$ , а відповідна часткова сума  $G_{3,4}$  дорівнює  $\hat{u}_6$ , що означає, що для оновлення вузла  $G$  його вихід має вже оцінити  $\hat{u}_6$ . Отже правило оновлення для жорстких повідомлень можна виразити наступним чином:

$$\hat{s}_{i+1,j} = \begin{cases} s_{i,j} \oplus s_{i,j+2^{i-1}}, & \left| \frac{j-1}{2^{i-1}} \right| \bmod 2 = 0 \\ s_{i,j}, & \text{інакше} \end{cases} \quad (7)$$

в той час, як правило оновлення для м'яких повідомлень може бути визначено наступним виразом:

$$LLR_{i,j} = \begin{cases} \min(|LLR_{i+1,j}|, |LLR_{i+1,j+2^{i-1}}|), & \left| \frac{j-1}{2^{i-1}} \right| \bmod 2 = 0 \\ (1 - 2s_{i,j-2^{i-1}})LLR_{i+1,j-2^{i-1}} + LLR_{i+1,j}, & \text{інакше} \end{cases} \quad (8)$$

де  $i = 1, 2, K, n$  та  $j = 1, 2, K, N$

Алгоритм декодування послідовного виключення складається з двох завдань, перше з яких проходить послідовно від вхідного етапу до вихідного для визначення LLR і прийняття жорсткого рішення. Друге завдання рекурсивно виконує вузли-попередники для зворотного розповсюдження декодованих  $\hat{u}_s$  з виходу на відповідну попередню стадію.

Алгоритм декодування SCL є значним удосконаленням алгоритму SC, де на кожному етапі декодування розглядаються  $L$  шляхів декодування. Зокрема алгоритм SCL подвоює кількість шляхів декодування для кожного інформаційного біта, таким чином переслідуючи обидва біти оцінки  $u_i = 0$  і  $u_i = 1$ , а потім використовує процедуру укорочення, щоб відкинути все, крім  $L$  найбільш ймовірних шляхів. Наприкінці процесу декодування як кодове слово вибирається найімовірніший шлях. Продуктивність алгоритму декодування SCL набагато краща порівняно з алгоритмом SC, особливо при нижчому рівні шуму, де оцінки ненадійні через малі значення LLR. Алгоритм декодування SCL складніше, ніж SC, і декодер SCL має більшу затримку, ніж реалізований декодер SC.

Алгоритм CA-SCL є удосконаленням відповідного алгоритму SCL для коротких чи середніх блоків. Коди CRC є найбільш широко використовуваною схемою кодування для виявлення помилок в практичних системах зв'язку [6].  $K$ -бітний вхідний блок кодера з виправленням помилок складається з інформаційних бітів  $k$  і  $m$ -бітної послідовності CRC. Декодер CA-SCL направляє послідовності-кандидати в детектор CRC, а останній повертає результати перевірки, щоб допомогти у визначенні кодового слова. Продуктивність такої реалізації полярного декодера покращується [6].

Арікан також запропонував полярний декодер поширення переконань [4], який використовує такий самий DFG (data flow graph), що й SC. Замість передачі жорстких повідомлень BP Polar Decoder передає програмні повідомлення між вузлами перевірки та вузлами змінних. Продуктивність помилок полярного декодера BP має додатковий вииграш у кодуванні порівняно з SC, але використовувати паралелізм складно. Крім того, полярний декодер BP відрізняється більш високою складністю реалізації та меншою пропускну здатністю, ніж декодер SC Polar [3].

### Реалізація полярного кодування

DFG алгоритм декодування SC складається з  $2^n$  бінарних дерев, і для максимального використання обладнання SoC FPGA пропонується реалізація одного бінарного дерева з вузлами, що налаштовуються [6]. Ця архітектура Polar Decoder передбачає, що на кожному етапі пара вузлів  $F$  і  $G$  використовує однакову пару LLR, тому такі вузли  $F$  і  $G$  групуються в одному процесорному блоці. Також активується лише одне бінарне дерево для оцінки  $\hat{u}_i$ . Отже, цей полярний декодер містить  $2^n - 1$  блоків обробки, а блок жорстких рішень оцінює LLR вихідного каскаду. Набір бітів рішення, необхідних для вузла  $G$  усередині блоків обробки, зберігається у пам'яті та використовується у відповідний час.

Як приклад для 8-точкового полярного коду реалізовані два комбінаційних декодери  $N = 4$ , щоб зробити обчислення паралельними, у той час як додатковий етап  $N = 8$  DFG використовується як вхідні дані вихідні сигнали цих двох декодерів. Для обчислення рішень для кожного вузла  $G$  було реалізовано Polar Encoder. На вхід кодера подаються правильно оцінені біти, а вихід кодера використовується як рішення для вузлів  $G$  на додатковому етапі.

У поданні двійкового дерева декодера SC деякі блоки обробки неактивні, оскільки процес декодування використовує блоки обробки наступного етапу. Поки ці блоки обробки неактивні, вставляється новий набір LLR каналу, і це триває за умови, що залежності даних задоволені. Наприклад, у 8-точковому полярному декодері з використанням цієї реалізації [7] три кодові слова були декодовані з тією ж затримкою, що і в полярному декодері уявлення двійкового дерева в [8]. Це значне покращення пропускну здатності, але збільшується вартість апаратного ускладнення.

### Вузли $F$ і $G$

Щоб  $F$ -вузол був реалізований із наближенням minSum, цей компонент містить схему, що реалізує функцію ABS, яка обчислює абсолютне значення своїх входів. Також використовується компаратор, який визначає мінімальне значення. Також включені два мультиплексори два-до-одного. Перший вибирає, яке з двох абсолютних значень має мінімальне значення, а його сигнал вибору є виходом компаратора. Вихід

першого мультиплектора та його додатковий код використовуються як входи для другого мультиплектора два-до-одного, як показано на рис. 8. Сигналом вибору цього мультиплектора є вихідний сигнал вентиля XOR (виключене АБО). Наближення minSum визначається згідно виразу:

$$output = sign(l_a) * sign(l_b) * \min(abs(l_a), abs(l_b))$$

Блок-схема вузла F показано на рис. 8.

Функція G є умовним додаванням/відніманням, що залежить від значення біта  $\hat{u}_s$  у відповідній точці DFG декодера. Вузол G визначається наступним виразом:

$$g(\hat{u}_s, a, b) = (1 - 2\hat{u}_s)a + b$$

Залежно від значення  $\hat{u}_s$ , в якості одної з складових вибирається a, або додатковий код a. Цей компонент реалізує суматор, який складає раніше обраний доданок з іншим входом, як показано на рис. 9.

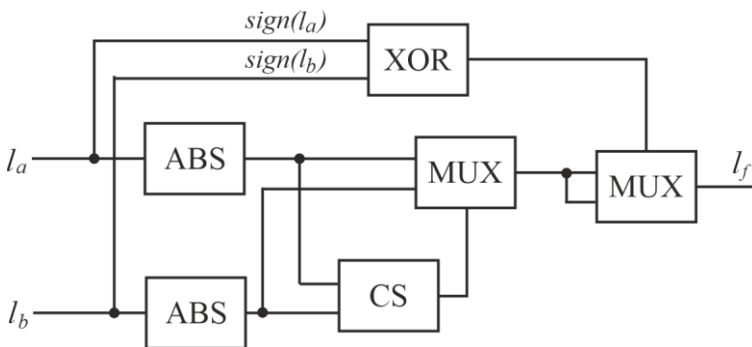


Рис. 8. Блок-схема вузла F (ABS – Абсолютне значення; CS – Порівняти і вибрати, MUX – Мультиплексор)

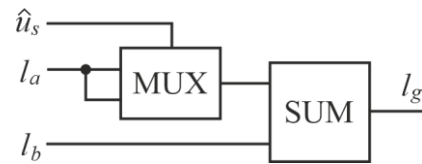


Рис. 9. Блок-схема вузла G (MUX – Мультиплексор, SUM – Підсумовувач)

### Експериментальні дослідження

Проведемо дослідження системи зв'язку за рис. 10 у середовищі MATLAB.

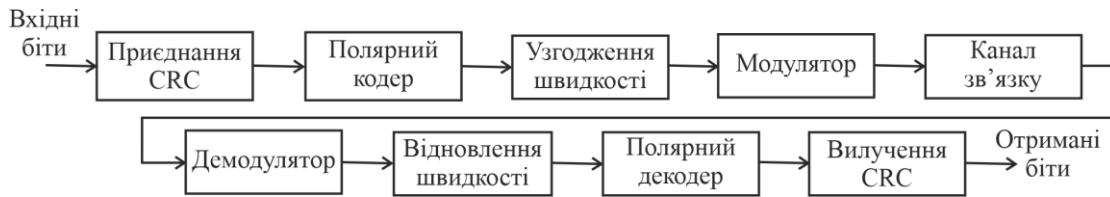


Рис. 10. Структурна схема системи зв'язку

Полярний кодер і декодер реалізовані на FPGA System-on-Chip (SoC) Intel DE10-Standard Development Kit. Процесор Cortex-A9 має два вбудованих ядра з програмованою логікою. SoC Cyclone V SE 5CSXFC6D6F31C6N інтегрує систему жорсткого процесора (HPS) на базі ARM, що складається з процесора, периферійних пристроїв та інтерфейсів пам'яті, пов'язаних із матрицею FPGA.

На рис. 10 показано, як використовується полярний код для передачі інформаційного повідомлення. Спочатку до інформаційного повідомлення приєднується циклічний надлишковий код (CRC) для підвищення завадостійкості, як показано на рис.11.



Рис. 11. Приєднання CRC до інформаційних бітів

$$g_{CRC-11}(D) = D^{11} + D^{10} + D^9 + D^5 + 1$$

За наведеним виразом формується наступна бітова послідовність: CRC11=[1 1 1 0 0 0 1 0 0 0 0 1]; CRC ефективний як зовнішній код для виявлення помилок.

Якщо обрана довжина коду M не відповідає довжині закодованої послідовності ( $2^{\log_2(M)}$ ), необхідна операція узгодження швидкості, яка полягає у виколюванні N-M позицій у кодовому слові довжини N. Далі відбувається BPSK-модуляція сигналу і передача його у середовище розповсюдження [9, 10].

У приймачі виконуються наступні операції (рис. 10):

- BPSK-демодуляція;
- відновлення швидкості, що полягає у відновленні за розміром закодованої послідовності (N) і кількістю виколотих позицій (P) кодового слова, що подається на декодер;

- полярне декодування, що полягає у відновленні переданого повідомлення [11];
- вилучення CRC.

На рис. 12 приведені результати дослідження кількості бітових помилок від відношення сигнал-шум для системи зв'язку з модуляцією BPSK і різною довжиною кодового блоку [12].

Реверсування бітів відсутнє, приєднаний CRC довжиною 11 бітів. Для полярного коду з довжиною блоку  $M=32$  і кодовою швидкістю  $R=1/2$ , кількість інформативних бітів з урахуванням CRC:  $K=16$ . Кількість закодованих бітів на виході кодера полярного коду  $N=32$  (має визначатись степенем числа 2). В цьому випадку виколування позицій полярного коду для узгодження швидкості відсутнє [13].

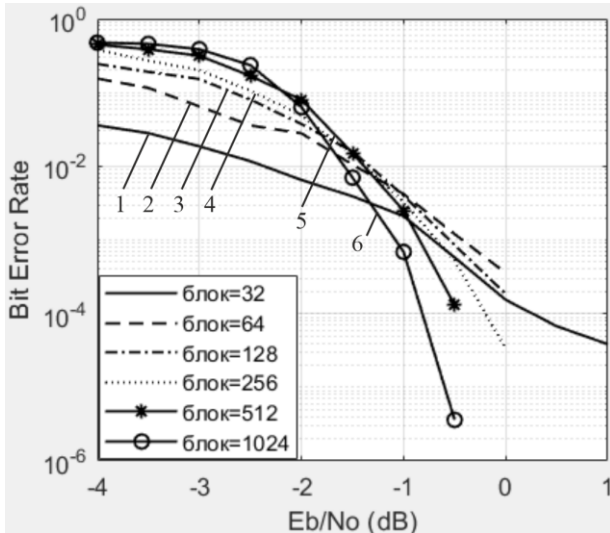


Рис. 12. Залежність BER від  $E_b/N_0$  для системи зв'язку з різною довжиною кодового блоку: (1 – 32 біта; 2 – 64 біта; 3 – 128 біт; 4 – 256 біт; 5 – 512 біт; 6 – 1024 біт)

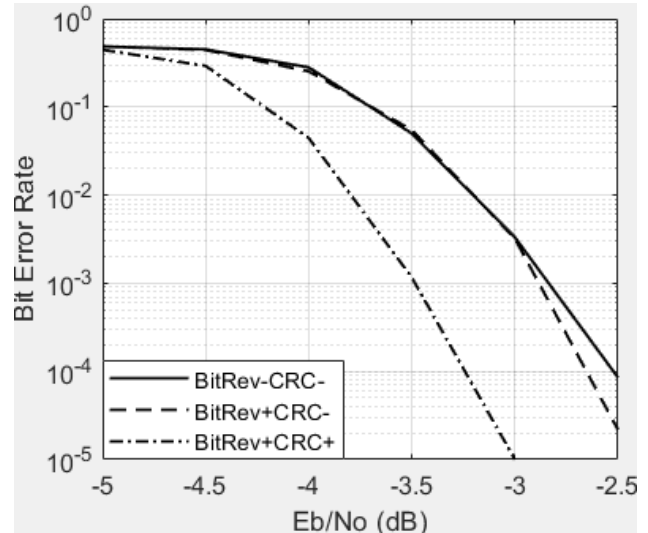


Рис. 13. Залежність кількості бітових помилок від відношення сигнал-шум для системи зв'язку з циклічним надлишковим кодом і реверсуванням бітів

З отриманих результатів можна зробити висновок, що зі збільшенням розміру блоку даних (кодового слова) від 64 біт до 1024 біт завадостійкість системи зв'язку підвищується на 1 дБ.

На рис. 13 приведені результати дослідження кількості бітових помилок від відношення сигнал-шум для системи зв'язку з циклічним надлишковим кодом і реверсуванням бітів. З отриманих результатів можна зробити висновок, що наявність реверсування бітів пришвидшує загасання BER на 0,2 дБ. Приєднання CRC підвищує завадостійкість на 0,5 дБ.

Проведемо дослідження кількості бітових помилок від відношення сигнал-шум для системи зв'язку з різним розміром списку послідовного виключення (рис. 14) і різною кодовою швидкістю (рис. 15).

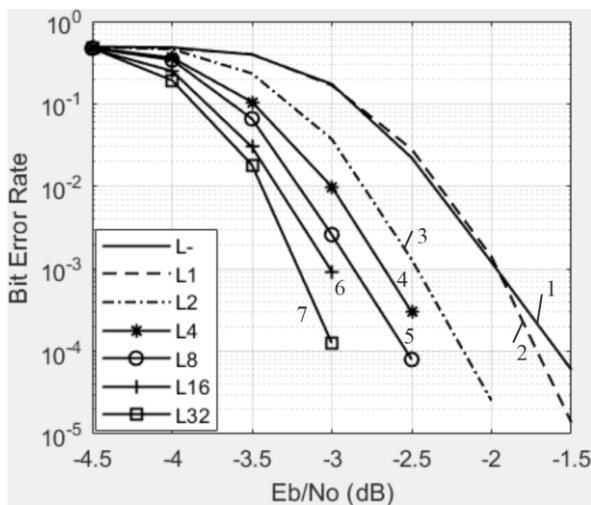


Рис. 14. Залежність BER від  $E_b/N_0$  системи зв'язку з різною довжиною списку декодера послідовного виключення (1 – список відсутній; 2 –  $L=1$ ; 3 –  $L=2$ ; 4 –  $L=4$ ; 5 –  $L=8$ ; 6 –  $L=16$ ; 7 –  $L=32$ )

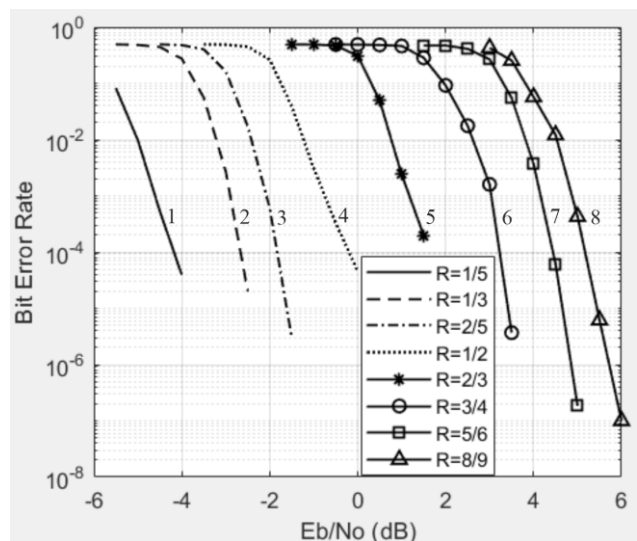


Рис. 15. Залежність BER від  $E_b/N_0$  системи зв'язку з різною кодовою швидкістю (1 –  $R=1/5$ ; 2 –  $R=1/3$ ; 3 –  $R=2/5$ ; 4 –  $R=1/2$ ; 5 –  $R=2/3$ ; 6 –  $R=3/4$ ; 7 –  $R=5/6$ ; 8 –  $R=8/9$ )

Для моделювання було використано довжина кодового слова  $M=4000$ , довжина кодового слова на виході кодеру  $N=4096$ , кількість виколотих бітів  $P=96$  [13].

З отриманих залежностей можна зробити висновок, що декодер списку послідовного виключення дозволяє підвищити завадостійкість системи зв'язку: навіть розмір списку  $L=2$  підвищує завадостійкість на 0,7 дБ у порівнянні з класичним декодером послідовного виключення. Подальше збільшення розміру списку не дає суттєвого покращення продуктивності: збільшення розміру списку з  $L=2$  до  $L=32$  збільшує завадостійкість на 0,8 дБ, але додає суттєву затримку. Збільшення кодової швидкості має більший вплив на коефіцієнт бітових помилок: збільшення кодової швидкості від 1/5 до 8/9 потребує збільшення відношення сигнал-шум на 10 дБ, але дозволяє збільшити швидкість передавання даних у 4,5 рази.

### Висновки з даного дослідження і перспективи подальших розвідок у даному напрямі

Проведене дослідження системи зв'язку з полярним кодуванням. Розглянуті кодер, декодер полярного коду, принципи поляризації каналу. Проаналізована дія двійкового каналу стирання (BEC) на послідовність бітів. Визначена пропускна здатність  $I(W)$  кожного віртуального поляризованого каналу і принципи поділу цих каналів на «погані і «хороші» за значенням параметра Бгаттачар'ї.

Проведене дослідження кількості бітових помилок від відношення сигнал-шум для системи зв'язку з модуляцією BPSK і різною довжиною кодового блоку, циклічним надлишковим кодом і реверсуванням бітів, різним розміром списку послідовного виключення і різною кодовою швидкістю. З отриманих результатів можна зробити висновок, що зі збільшенням розміру блоку даних (кодового слова) від 64 біт до 1024 біт завадостійкість системи зв'язку підвищується на 1 дБ. Наявність реверсування бітів пришвидшує загасання BER на 0,2 дБ. Приєднання CRC підвищує завадостійкість на 0,5 дБ.

Використання декодера списку послідовного виключення з розміром списку  $L=2$  підвищує завадостійкість системи зв'язку на 0,7 дБ у порівнянні з класичним декодером послідовного виключення. Подальше збільшення розміру списку до  $L=32$  збільшує завадостійкість на 0,8 дБ, але додає суттєву затримку. Збільшення кодової швидкості має більший вплив на коефіцієнт бітових помилок: збільшення кодової швидкості від 1/5 до 8/9 потребує збільшення відношення сигнал-шум на 10 дБ, але дозволяє збільшити швидкість передавання даних у 4,5 рази.

### Література

1. Пятін І.С. Методика полярного кодування в 5G мобільних засобах телекомунікацій з багатопозиційною модуляцією /І.С. Пятін, Ю.М. Бойко //Вимірювальна та обчислювальна техніка в технологічних процесах. – 2020. - №1 - С.67-76.
2. Пятін І. С. Порівняння продуктивності завадостійких кодів на основі програмного HDL моделювання для захищених інформаційних технологій / І. С. Пятін, Ю. М. Бойко // Інфокомунікаційні та комп'ютерні технології. – 2022. – № 1(03). – С. 39-62.
3. C. Andriakopoulos and V. Paliouras, "Data representation and hardware aspects in a fully-folded successive-cancellation polar decoder," 2018 7th International Conference on Modern Circuits and Systems Technologies (MOCASST), Thessaloniki, Greece, 2018, pp. 1-4, doi: 10.1109/MOCASST.2018.8376633.
4. E. Arikan, "Channel Polarization: A Method for Constructing Capacity-Achieving Codes for Symmetric Binary-Input Memoryless Channels," in IEEE Transactions on Information Theory, vol. 55, no. 7, pp. 3051-3073, July 2009, doi: 10.1109/TIT.2009.2021379.
5. H. Ochiai, P. Mitran and H. Vincent Poor, "Capacity-Approaching Polar Codes With Long Codewords and Successive Cancellation Decoding Based on Improved Gaussian Approximation," in IEEE Transactions on Communications, vol. 69, no. 1, pp. 31-43, Jan. 2021, doi: 10.1109/TCOMM.2020.3030057.
6. W. Tan, A. Wang, Y. Xu and Y. Lao, "Area-Efficient Pipelined VLSI Architecture for Polar Decoder," 2020 IEEE Computer Society Annual Symposium on VLSI (ISVLSI), Limassol, Cyprus, 2020, pp. 352-357, doi: 10.1109/ISVLSI49217.2020.00071.
7. Federico G. Krasser, Mónica C. Liberatori, Leonardo Coppolillo, Fast and efficient FPGA implementation of Polar Codes and SoC test bench, Microprocessors and Microsystems, Volume 84, 2021. <https://doi.org/10.1016/j.micpro.2021.104264>.
8. C. Leroux, I. Tal, A. Vardy and W. J. Gross, "Hardware architectures for successive cancellation decoding of polar codes," 2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Prague, Czech Republic, 2011, pp. 1665-1668, doi: 10.1109/ICASSP.2011.5946819.
9. J. Boiko, I. Pyatin and O. Eromenko, "Simulation of the Transport Channel With Polar Codes for the 5G Mobile Communication," 2020 IEEE International Conference on Problems of Infocommunications. Science and Technology (PIC S&T), Kharkiv, Ukraine, 2020, pp. 182-186.
10. Z. B. Kaykac Egilmez, L. Xiang, R. G. Maunder and L. Hanzo, "The Development Operation and Performance of the 5G Polar Codes", IEEE Communications Surveys & Tutorials, vol. 22, no. 1, pp. 96-122, 2020.

11. Бойко Ю. Transmission of control information in 5G broadband telecommunication systems / Ю.Бойко, І. П'ятін, Л. Карпова, І. Пархомей //Адаптивні системи автоматичного управління. – 2021. – Т. 1. – №. 38. – С. 82-95.

12. Mordiyuk V. Методичні аспекти визначення основних параметрів сигнально-кодових конструкцій системи автоматизованого встановлення зв'язку у сеансах з кореспондентами /V. Mordiyuk, Y. Synytskiy // Системи управління, навігації та зв'язку. Збірник наукових праць. – Полтава: ПНТУ, 2023. – Т. 2 (72). – С.199-203.

13. Бойко Ю. Особливості формування кодової надлишковості у каналах передачі інформації / Ю.Бойко, А. Семенко, І. П'ятін, //Інфокомунікаційні та комп'ютерні технології. – 2022 - Т.2, №04. - С. 12-25.

### References

1. Piatin I. Polar coding technique in 5G mobile telecommunications with multi-position modulation /I. Piatin, J. Boiko //Measuring and computing devices in technological processes. – 2020. – No. 1 - S.67-76.

2. Piatin I. Comparison the performance of error-control code based on software HDL modeling for information security technologies / I. Piatin, J. Boiko //Infocommunication and computer technologies. – 2022. – Vol. 1, No. 3. – S. 39-62.

3. C. Andriakopoulos and V. Paliouras, "Data representation and hardware aspects in a fully-folded successive-cancellation polar decoder," 2018 7th International Conference on Modern Circuits and Systems Technologies (MOCASST), Thessaloniki, Greece, 2018, pp. 1-4, doi: 10.1109/MOCASST.2018.8376633.

4. E. Arıkan, "Channel Polarization: A Method for Constructing Capacity-Achieving Codes for Symmetric Binary-Input Memoryless Channels," in IEEE Transactions on Information Theory, vol. 55, no. 7, pp. 3051-3073, July 2009, doi: 10.1109/TIT.2009.2021379.

5. H. Ochiai, P. Mitran and H. Vincent Poor, "Capacity-Approaching Polar Codes With Long Codewords and Successive Cancellation Decoding Based on Improved Gaussian Approximation," in IEEE Transactions on Communications, vol. 69, no. 1, pp. 31-43, Jan. 2021, doi: 10.1109/TCOMM.2020.3030057.

6. W. Tan, A. Wang, Y. Xu and Y. Lao, "Area-Efficient Pipelined VLSI Architecture for Polar Decoder," 2020 IEEE Computer Society Annual Symposium on VLSI (ISVLSI), Limassol, Cyprus, 2020, pp. 352-357, doi: 10.1109/ISVLSI49217.2020.00071.

7. Federico G. Krasser, Mónica C. Liberatori, Leonardo Coppelillo, Fast and efficient FPGA implementation of Polar Codes and SoC test bench, Microprocessors and Microsystems, Volume 84, 2021. <https://doi.org/10.1016/j.micpro.2021.104264>.

8 C. Leroux, I. Tal, A. Vardy and W. J. Gross, "Hardware architectures for successive cancellation decoding of polar codes," 2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Prague, Czech Republic, 2011, pp. 1665-1668, doi: 10.1109/ICASSP.2011.5946819.

9. J. Boiko, I. Pyatin and O. Eromenko, "Simulation of the Transport Channel With Polar Codes for the 5G Mobile Communication," 2020 IEEE International Conference on Problems of Infocommunications. Science and Technology (PIC S&T), Kharkiv, Ukraine, 2020, pp. 182-186.

10. Z. B. Kaykac Egilmez, L. Xiang, R. G. Maunder and L. Hanzo, "The Development Operation and Performance of the 5G Polar Codes", IEEE Communications Surveys & Tutorials, vol. 22, no. 1, pp. 96-122, 2020.

11. Boiko J. Transmission of control information in 5G broadband telecommunication systems /J. Boiko, I. Pyatin, L. Karpova, I. Parkhomye //Adaptive systems of automatic control. – 2021. – Vol. 1. – No. 38. – S. 82-95.

12. Mordiyuk V. Methodological aspects of determining the main parameters of the signal-code constructions of the automated contact establishment system in sessions with correspondents /V. Mordiyuk, Y. Synytskiy // Control, Navigation and Communication Systems. Academic Journal. – Poltava: PNTU, 2023. – VOL. 2 (72). – PP. 199-203.

13. Boiko J. Features of code redundancy formation in information transmission channels /J. Boiko, A. Semenکو, I. Pyatin //Infocommunication and computer technologies. – 2022 - Vol.2, № 04. - P. 12-25.

Завідувачу кафедри телекомунікацій,  
медійних та інтелектуальних технологій  
д.т.н., професору ПІДЧЕНКУ Сергію  
здобувача вищої освіти  
ПАНАСЮКА Дмитра  
ФІТ, гр. ТРМ-22-1


### ЗАЯВА

З правилами чинного Положення «Про дотримання академічної доброчесності в Хмельницькому національному університеті» від 26.09.2020 (зі змінами від 26.11.2020), згідно з яким виявлення плагіату є підставою для відмови в допуску кваліфікаційної роботи до захисту та застосування заходів дисциплінарної та академічної відповідальності, ознайомлений. Про використання програмно-технічних засобів для перевірки кваліфікаційних робіт здобувачів вищої освіти на плагіат оповіщений (а) та надаю свою згоду на обробку та збереження університетом моєї роботи в інституційному репозитарії університету.

Також надаю університету право на передачу моєї дипломної роботи «Метод підвищення пропускнуої здатності безпроводних локальних мереж стандарту IEEE 802.11» роботи для обробки та збереження в базах даних програмно-технічних засобів (Unicheck та Anti-Plagiarism) та використання роботи для виявлення плагіату в інших роботах, які перевіряються програмно-технічними засобами та користувачами, що мають доступ до цих програмно-технічних засобів, виключно в обмежених цілях для виявлення плагіату в текстах робіт.

Робота для перевірки університетом надається в друкованому та електронному варіанті. Електронна версія моєї роботи збігається (ідентична) з друкованою.

17 листопада 2023 р.  
дата

  
підпис

## Anti-Plagiarism v-15.257

**Максимальное совпадение с одним документом 1.0%**

**Словари проверки: en\_US, ru\_RU, ua\_UA. Ошибок в документах: 11%**

ID: 121335 Название: Метод підвищення пропускної здатності безпроводних локальних мереж стандарту IEEE 802.11 Добавлено в БД: 2023-11-28 Авторы: Панасюк Дмитро Володимирович Руководители: Бойко Юлій Миколайович Консультанты: Опоненты:	Документ		Суммарное совпадение по Базе Данных	
	Символы	Лексемы	Символы	Лексемы
	108598	1723	1843 (2%)	29 (2%)



Имя пользователя:  
Kafedra TMIT KhNU

Дата проверки:  
29.11.2023 14:34:57 EET

Дата отчета:  
29.11.2023 17:46:00 EET

ID проверки:  
1015952456

Тип проверки:  
Doc vs Internet + Library

ID пользователя:  
100005657

Название файла: Панасюк\_ТРМ-22

Количество страниц: 121 Количество слов: 21237 Количество символов: 148258 Размер файла: 6.47 MB ID файла: 1015628446

335 слов помечены как "исключенные" и не учитываются в подсчете слов

## 7.95% Совпадения

Наибольшее совпадение: 2.36% с Интернет-источником (<https://arxiv.org/pdf/2306.12063.pdf>)

7.84% Источники из Интернета

720

Страница 123

2.59% Источники из Библиотеки

69

Страница 128

## 0.34% Цитат

Цитаты

7

Страница 129

Не найдено ни одной ссылки

## 0% Исключений

Некоторые источники исключены автоматически (фильтры исключения: количество найденных слов мень...

## РІШЕННЯ

### кафедри телекомунікацій, медійних та інтелектуальних технологій

#### ПРО ДОПУСК ДИПЛОМНОЇ РОБОТИ ДО ЗАХИСТУ

Підтверджуємо ознайомлення з результатом звіту подібності щодо роботи, генерованого системою виявлення текстових збігів/ідентичності/схожості:

Назва: Метод підвищення пропускнув здатності безпроводних локальних мереж стандарту IEEE 802.11

Автор: Панасюк Дмитро Володимирович

Спеціальність: 172 Телекомунікації та радіотехніка

Науковий керівник: д.т.н., професор Бойко Юлій Миколайович

Після аналізу звіту подібності зроблено такий висновок:

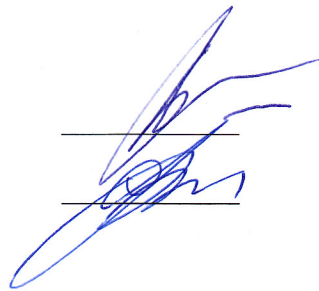
№	Висновок	Позначка про відповідність
1	Запозичення, виявлені в роботі, є законними і не є плагіатом (далі – зазначаються підстави віднесення запозичень до правомірних). Робота приймається до захисту.	-
2	Виявлені запозичення не є плагіатом, розміщені в розділах, які не описують безпосередньо авторське дослідження, але кількість цитат перевищує обсяг, виправданий поставленою метою роботи (далі – зазначаються детальні та аргументовані підстави віднесення запозичень до правомірних). Робота приймається до захисту, але має бути відкоригована. Відкоригований варіант має бути поданий на кафедру за 2 дні до захисту, разом із заявою щодо самостійності виконання письмової роботи та ідентичності друкованої та електронної версії роботи	відповідає
3	Виявлені запозичення не є плагіатом, але частково розміщені в розділах, які описують безпосередньо авторське дослідження, а кількість цитат перевищує обсяг, виправданий поставленою метою роботи. В зв'язку з цим мета роботи та поставлені завдання не були досягнені. Робота може бути допущена до захисту (наступного року) після того як буде відкоригована та допрацьована і успішно пройде повторну перевірку на академічний плагіат.	-
4	Робота містить навмисні текстові спотворення, передбачувані спроби укріття запозичень або інші прояви академічного плагіату. Робота містить фабрикацію або фальсифікацію даних. Робота не допускається до захисту.	-
5	Інше:	-

Підтвердження: Виявленні запозичення не є плагіатом так як розміщені в розділах, які не описують безпосередньо авторське дослідження (є власні терміни, визначення тощо), складають 7,95% та мають посилання на приведений список літературних джерел.

«29» листопада 2023 р.

Науковий керівник

Завідувач кафедрою ТМІТ



Юлій БОЙКО

Сергій ПІДЧЕНКО

## В І Д Г У К

на дипломну роботу магістра Панасюка Д.В.

«Метод підвищення пропускної здатності безпроводних локальних мереж стандарту IEEE 802.11»

Використання пристроїв з підтримкою Wi-Fi збільшується з кожним роком. Споживачам потрібно більшої швидкості, більшої надійності та більшої енергоефективності. Стандарт 802.11ac, або Wi-Fi п'ятого покоління надає кращий рівень зв'язку. Стандарт 802.11ac забезпечує гігабітну пропускну здатність і працює в діапазоні 5 ГГц. Він використовує нові методи: формування променя на боці передвача та використання кодів LDPC у приймачі. Стандарт 802.11ax, або Wi-Fi 6 забезпечує підвищення пропускної здатності безпроводних локальних мереж стандарту IEEE 802.11 за рахунок використання модуляції 1024-QAM, використання технології MU-MIMO, збільшення кількості просторових потоків на боці передавача і приймача.

В першому розділі дипломної роботи здійснено огляд принципів побудови безпроводних локальних мереж за стандартом IEEE 802.11.

У другому розділі роботи представлені математичні моделі та способи побудови кодеру і декодеру LDPC.

В третьому розділі дипломної роботи розглянуті квазіциклічні коди LDPC, їх апаратна реалізація у FPGA.

Четвертий розділ кваліфікаційної роботи присвячений дослідженню пропускної здатності безпроводних локальних мереж, дослідженні Simulink моделі системи зв'язку, визначенні коефіцієнту бітових помилок при зміні позиційності цифрової модуляції, швидкості коду, розміру фрейму.

Результати дипломної роботи опубліковані у журналі «Вимірювальна та обчислювальна техніка в технологічних процесах», №3, 2023 р.

Дипломна робота магістра виконана на високому науково-технічному рівні і заслуговує оцінку «відмінно», а Панасюка Д.В. – присвоєння кваліфікації магістра з спеціальності «Телекомунікації та радіотехніка».

Доктор техн. наук, проф.



Бойко Ю.М.

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

РЕЦЕНЗІЯ НА ДИПЛОМНУ РОБОТУ

Дипломник: Панасюк Дмитро Володимирович

Тема роботи: Метод підвищення пропускної здатності безпроводних локальних мереж стандарту IEEE 802.11

Спеціальність 172 «Телекомунікації та радіотехніка»

Обсяг дипломної роботи

Кількість листів креслень 11 Кількість сторінок записки 98

1. Короткий зміст роботи та прийнятих рішень в результаті виконаного наукового дослідження Дипломна робота присвячена дослідженню енергетичного виграшу кодування при використанні кодів LDPC і багатопозиційної цифрової модуляції. Збільшення кодової швидкості з 1/2 до 5/6 потребує підвищення відношення сигнал-шум на 4 дБ, і дозволяє підвищити швидкість передавання інформації на 33%. Перехід з модуляції QPSK на 1024-QAM потребує підвищення відношення сигнал-шум на 20 дБ і дозволяє підвищити швидкість передавання інформації з 2 біт на символ до 10 біт на символ. Використання більш довгої довжини блоку підвищує швидкість передавання інформації і завадостійкість системи зв'язку.

2. Висновок про відповідність роботи дипломному завданню Дипломна робота відповідає виданому завданню

3. Характеристика виконання кожного розділу, ступінь використання останніх досягнень науки та техніки і передових методів роботи: В першому розділі дипломної роботи здійснено огляд принципів побудови безпроводних локальних мереж за стандартом IEEE 802.11. У другому розділі роботи представлені математичні моделі та способи побудови кодеру і декодеру LDPC. В третьому розділі дипломної роботи розглянуті квазіциклічні коди LDPC, їх апаратна реалізація у FPGA. Четвертий розділ присвячений дослідженню пропускної здатності безпроводних локальних мереж, дослідженні Simulink моделі системи зв'язку, визначенні коефіцієнту бітових помилок при зміні позиційності цифрової модуляції, швидкості коду, розміру фрейму.

4. Позитивні сторони роботи: Проведене дослідження залежностей коефіцієнту бітових помилок безпроводної локальної мережі стандарту IEEE 802.11 від різних факторів, що впливають на пропускну здатність, досліджено декодер LDPC з використанням спрощеного алгоритму мінімальної суми.

Сучасний стандарт 802.11ax використовує технологію MIMO, вводить режим ортогонального частотного мультиплексування OFDMA для покращення спектральної ефективності і модуляція 1024-QAM

5. Негативні сторони роботи: У роботі бажано було би провести дослідження залежностей коефіцієнту бітових помилок від відношення сигнал-шум для системи зв'язку з цифровою квадратурною модуляцією і кодуванням LDPC на радіочастоті для оцінки реального комплексу завад у каналі зв'язку. Присутні невеликі граматичні помилки. Однак, ці недоліки не мають принципового значення, суттєво не впливають на кінцевий результат і не знижують загального враження від проведеної роботи.

6. Оцінка графічного оформлення та пояснювальної записки роботи: немає

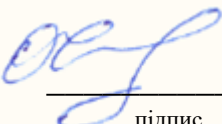
7. Відгук про роботу в цілому: Дипломна робота виконана на високому технічному рівні, має безперечну актуальність в області безпроводних локальних мереж стандарту IEEE 802.11. Результати дослідження мають важливе практичне застосування при налаштуванні безпроводового з'єднання.

8. Інші зауваження: немає

9. Оцінка дипломної роботи: Дипломна робота відповідає встановленим вимогам і заслуговує оцінки «відмінно» (5.00/А), а її автору Панасюку Д.В. присвоєння кваліфікації магістра зі спеціальності «Телекомунікації та радіотехніка»

10. Рецензент (прізвище, ім'я, по батькові, місце роботи) Єрмоменко Олександр Іванович – к.т.н., доцент кафедри фізики та електротехніки

«30» листопада 2023р.



підпис