

Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет Комп'ютерної інженерії та управління
(повна назва)

Кафедра Автоматизації проектування обчислювальної техніки
(повна назва)

АТЕСТАЦІЙНА РОБОТА
Пояснювальна записка

рівень вищої освіти другий (магістерський)
(рівень вищої освіти)

Декодер Вітербі на програмованих логічних інтегральних схемах
для системи цифрового наземного телебачення стандарту DVB-T
(тема)

Виконав: студент 2 курсу, групи СКСм-19-1
Кравцов К.Р.
(прізвище, ініціали)

Спеціальність 123 – Комп'ютерна інженерія
(код і повна назва спеціальності)

Тип програми освітньо-професійна
(освітньо-професійна або освітньо-наукова)

Освітня програма Спеціалізовані комп'ютерні системи
(повна назва освітньої програми)

Керівник доц. Філіппенко І.В.
(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри

(підпис)

Чумаченко С.В.
(прізвище, ініціали)

2020 р.

Харківський національний університет радіоелектроніки

Факультет Комп'ютерної інженерії та управління
Кафедра Автоматизації проектування обчислювальної техніки
Рівень вищої освіти другий (магістерський)
Спеціальність 123 – Комп'ютерна інженерія
Тип програми Освітньо-професійна
Освітня програма Спеціалізовані комп'ютерні системи

ЗАТВЕРДЖУЮ:
Зав. кафедри _____
(підпис)
« ____ » _____ 2020 р.

ЗАВДАННЯ
НА АТЕСТАЦІЙНУ РОБОТУ (ПРОЕКТ)

Студентові Кравцову Кирилу Романовичу
(прізвище, ім'я, по батькові)

- 1.Тема роботи (проекту) Декодер Вітербі на програмованих логічних інтегральних схемах для системи цифрового наземного телебачення стандарту DVB-T
затверджена наказом по університету від " 30 " 10 2020 р. № 1489 Ст.006
2. Термін подання студентом роботи (проекту) 13.12.2020
3. Вихідні дані до роботи (проекту) _____
Мова опису апаратури VHDL,
САПР Active-HDL,
САПР XILINX Vivado,
Matlab
4. Зміст пояснювальної записки (перелік питань, що потрібно розробити) _____
Аналіз предметної галузі та постановка задачі проектування.
Розробка кодера завадостійкого кодування.
Розробка декодера завадостійкого кодування.
Розробка VHDL моделей кодеку.
Моделювання розробленої моделі з використанням інструментальних засобів Active-HDL.
Синтез апаратної реалізації пристрою з використанням САПР Xilinx Vivado.
Проведення діагностичного експерименту.

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслеників, плакатів) 18 слайдів


6. Консультанти розділів роботи (проекту)

Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата

7. Дата видачі завдання 30.10.2020

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи (проекту)	Термін виконання етапів проекту (роботи)	Примітка
1	Видача теми проекту, узгодження і затвердження теми	30.10.2020 – 01.11.2020	
2	Аналіз проблемної галузі, постановка задачі, вибір інструментальних засобів	02.11.2020 – 04.11.2020	
3	Розробка моделі кодера	05.11.2020 – 07.11.2020	
4	Розробка моделі декодера	08.11.2020 – 11.11.2020	
5	Розробка VHDL опису кодека	12.11.2020 – 17.11.2020	
6	Проведення діагностичних експериментів	18.11.2020 – 22.11.2020	
7	Оформлення пояснювальної записки	23.11.2020 – 06.12.2020	
8	Перевірка виконаного проекту керівником, допуск до захисту	07.12.2020 – 13.12.2020	
9	Захист проекту	14.12.2020 – 27.12.2020	

Студент 
(підпис)

Керівник роботи (проекту) _____ доц. Філіппенко І.В.
(підпис) (посада, прізвище, ініціали)

РЕФЕРАТ

Записка пояснювальна містить 53 сторінок, 29 рисунків, 1 таблицю, 16 джерел за переліком посилань.

ДЕКОДЕР ВІТЕРБІ, САПР, ЗАВАДОСТІЙКЕ КОДУВАННЯ, МОДЕЛЮВАННЯ, ОПТИМІЗАЦІЯ

В роботі розглянуті питання проектування системи DVB-T. Предметом дослідження є вибір і обґрунтування структури декодера. Зроблено порівняльний аналіз ПЛІС в залежності від параметрів.

Мета роботи полягає в забезпеченні більшої гнучкості декодера Вітербі для системи DVB-T, завдяки переходу до програмованої елементній базі. Був проведений аналіз реалізації декодера від швидкості, різної бітності м'яких рішень.

На мові VHDL був спроектований декодер Вітербі, що відповідає стандарту DVB-T, гнучкість проекту якого полягає в можливості зміни його параметрів, таких як швидкість згортальності кода, породжують поліноми згортального кода, розрядність м'яких рішень і глибину декодування.

ABSTRACT

Explanatory note contains 53 pages, 29 figures, 1 table, 16 sources.

VITERBY DECODER, CAD, PRE-CODE, MODELING, OPTIMIZATION

The issues of DVB-T system design are considered in the work. The subject of research is the choice and justification of the structure of the decoder. The comparative analysis of FPGA depending on parameters is made.

The purpose of the work is to provide greater flexibility of the Viterbi decoder for the DVB-T system, due to the transition to a programmable element base. The implementation of the decoder implementation was analyzed from the speed, different bit rates of soft solutions.

The VHDL language has been designed with a Viterbi decoder that complies with the DVB-T standard, the flexibility of which is to change its parameters, such as the speed of code convolution, generate convolutional code polynomials, bit solutions and decoding depth.

ЗМІСТ

ЗМІСТ.....	6
ПЕРЕЛІК СКОРОЧЕНЬ, УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ І ТЕРМІНІВ.....	8
ВСТУП.....	9
1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ.....	11
1.1 Характеристики системи DVB-T	11
1.2 Згорткове кодування	20
1.3 Декодування згорткових кодів.....	24
1.4 QPSK модулятор.....	27
1.5 QPSK демодулятор.....	29
1.6 Програмована елементна база	30
1.7 Мова опису апаратури інтегральних схем VHDL	31
2 РОЗРОБКА АЛГОРИТМІВ ТА ВИБІР АППАРАТНОЇ ПЛАТФОРМИ.....	32
2.1 Гнучкість декодера Вітербі.....	32
2.2 Реалізація декодера на ПЛІС	34
2.3 Порівняння завадостійкості декодера Вітербі при різній розрядності м'яких рішень.....	35
2.4 Розробка QPSK модулятора	37
2.5 Моделювання системи з корекцією помилок DVB-T в середовищі Matlab	42
ВИСНОВКИ.....	52
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ.....	54
ДОДАТОК А.....	56
ДОДАТОК Б.....	58
ДОДАТОК В.....	60
ДОДАТОК Г.....	61
ДОДАТОК Д.....	63
ДОДАТОК Ж.....	64

ПЕРЕЛІК СКОРОЧЕНЬ, УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ
І ТЕРМІНІВ

ASIC – Application Specific Integrated Circuit;

AWGN – Additive White Gaussian Noise;

DVB-T – Digital Video Broadcasting – Terrestrial;

QAM – Quadrature Amplitude Modulation;

QPSK – Quadrature Phase Shift Keying;

VHDL – Very high speed integrated circuits Hardware Description Language;

ПЛІС – програмовані логічні інтегральні схеми;

САПР – система автоматизованого проектування;

ЗК – згортковий код.

ВСТУП

Матеріальною основою сучасних інформаційно-комунікаційних технологій є системи зв'язку, які створюються як взаємопов'язані системи інформаційного обміну та телекомунікацій з використанням елементної бази нового покоління.

Основними напрямками вдосконалення сучасних цифрових систем зв'язку, безперечно, є методи, спрямовані на поліпшення спектральної і енергетичної ефективності таких систем. Окремо кожне із зазначених напрямків характеризується своїми оптимальними параметрами, але прямий синтез цих двох технологій з метою подальшого поліпшення показників нової системи важкодоступний через їх суперечливості. Тому пошук компромісів при вирішенні подібних задач для різнотипних цифрових систем зв'язку і особливо цифрового радіозв'язку заслуговує особливої уваги. Одним з перспективних напрямків схожого роду є розвиток м'яких методів обробки перешкодостійких кодів.

При передачі даних по каналах зв'язку через пошкодження сигналів і впливу перешкод можливий помилковий прийом. Число виникаючих помилок визначається характеристиками каналу, і зазвичай, їх неприпустимо багато. Методи боротьби з помилками різноманітні.

Як відомо, будь-яка повторна передача даних веде до зменшення її швидкості, тому ефективніше використовувати методи боротьби з помилками без повторення. До них відносяться методи, які використовують коди, що виправляють помилки, зокрема, коди Ріда-Соломона[1], Боуза-Чоудхурі-хоквінгема (БЧХ), згорткові коди і багато інших.

Проте кожний тип кодування має свої переваги та недоліки, що ставить питання вибору кодування актуальним.

Згорткові коди широко використовуються в сучасних системах передачі даних в якості схем завадостійкого кодування. Популярність даних

кодів обумовлена можливістю їх декодування за допомогою алгоритму Вітербі, що забезпечує оптимальне декодування за критеріями максимальної правдоподібності при відносно невеликій (порівняно з іншими класами кодів) обчислювальній складності.

Зокрема, згортковий код (133, 177)[2] використовується в стандарті цифрового наземного телебачення DVB-T, як внутрішнього коду каскадної схеми кодування.

Важливим питанням при проектуванні системи DVB-T є вибір і обґрунтування структури декодера. В існуючих реалізаціях в якості елементної бази для реалізації схеми завадостійкого кодування стандарту DVB-T використовуються інтегральні схеми ASIC. ASIC мають ряд недоліків:

- відсутня можливість перепрограмування;
 - велика вартість систем проектування і складність виконання;
 - проектів, що призводять до високої вартості самої ASIC;
 - високий ризик проектних помилок, що призводить до необхідності перепроєктування;
 - на розробку і виготовлення ASIC потрібно в 2 рази більше часу, ніж для ПЛІС.

Мета роботи полягає в забезпеченні більшої гнучкості декодера Вітербі для системи DVB-T, завдяки переходу до програмованої елементній базі (ПЛІС).

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Характеристики системи DVB-T

Важливим напрямком в телекомунікації є цифрове наземне телебачення. З позиції ефективного використання радіочастотного спектру цифрове наземне телевізійне мовлення орієнтоване на систему DVB-T. DVB-T – європейський стандарт ефірного цифрового телевізійного мовлення з сімейства DVB. Серед всіх систем цифрового наземного телевізійного мовлення європейська система DVB-T розвивається досить динамічно і завойовує все більше прихильників, забезпечуючи найбільш високу якість у всіх можливих застосуваннях.

Для забезпечення завадостійкості в системі DVB-T використовується каскадне кодування. Для зовнішнього кодування використовується блоковий код Ріда-Соломона (204,188)[3]. При цьому до кожного пакету 188 байт вхідного потоку додається 16 перевірочних байт, що дозволяє скоригувати до 8 помилкових байт. Для захисту від пакетних помилок великої тривалості вводиться перемеження даних.

Внутрішнє кодування – це побітове кодування згортковим кодом з кодовим обмеженням, рівним 7, заданим породжуючими поліномами (171,133) 8 . На відміну від зовнішнього кодування, що має постійні параметри, для внутрішнього кодування можна міняти швидкість згорткових кодів. Є п'ять варіантів швидкості коду: 1/2, 2/3, 3/4, 5/6, 7/8. За стандартом DVB-T[4] використовується один з трьох видів модуляції QPSK, QAM16 і QAM64. Критерієм вибору служить або швидкість передачі даних, або стійкість.

Підвищення достовірності прийому сигналів в цифровому телебаченні вимагає використання завадостійкого кодування. При каналному кодуванні використовується каскадне кодування із зовнішнім і внутрішнім кодами.

Зовнішній код – це код, застосований на передавальній стороні першим, але декодується на приймальній стороні останнім. Внутрішній код – код, застосований на передавальній стороні останнім, але декодований на приймальній стороні в першу чергу. У цифровому ТБ дані транспортного пакета перемежаються, а потім кодуються за допомогою завадостійкого кодування каскадним кодом. В якості зовнішнього в DVB - T використовується код Ріда-Соломона, а в якості внутрішнього – згортковий код.

Основний принцип побудови коригувальних кодів полягає в тому, що в кожному передану кодову комбінацію, яка містить k інформаційних двійкових символів, вводять p додаткових двійкових символів. В результаті виходить нова кодова комбінація, яка містить $n = k + p$ двійкових символів. Такий код позначається як (n, k) . Частка інформаційних символів в ньому характеризується відносною швидкістю коду, яка визначається співвідношенням $R = k / n = k / k + p$ [5].

Для оцінки здатності коду виявляти і виправляти помилки використовується поняття кодової відстані d . Лінійні коди – це цифрові комбінації, що володіють важливою властивістю: два кодових слова можна скласти і отримати третій кодове слово. До них застосовні методи лінійної алгебри.

Наступна ступень класифікації – систематичні і несистематичні коди. У систематичних кодах k старших коефіцієнтів сформованого коду $z(x)$ містять вихідне повідомлення. У процесі кодування до блоку k інформаційних символів в кінці кодового слова додається блок з $2t$ перевірочних символів, де t – число виправляємих помилок. Це зручно при декодуванні, тому в телебаченні використовується систематичний код РС. Несистематичні коди не містять вихідного повідомлення в явній формі (наприклад, згортковий код). Циклічним кодом є лінійний код, що володіє наступною властивістю: якщо в кодовому слові виконана циклічна перестановка, то отримане слово

також є кодовим словом. Розглянемо коди РС і згорткові коди, які використовуються в стандарті DVB-T.

Коди РС або коди Ріда-Соломона – це недвійкові блокові циклічні коди, що дозволяють виправляти помилки в блоках даних. Недвійкові вони, тому що елементами кодового слова не є біти, а групи бітів. У телебаченні це – байти. Для восьмибітних символів повна довжина кодового слова становить $28-1 = 255$ символів, з них 239 інформаційних і 16 перевірочних. На рис. 1.1 показано розміщення байтів в пакеті даних MPEG 2 з захистом від помилок блоковим кодом РС. У стандарті цифрового телебачення DVB використовується код РС, який записується як (204, 188, 8). Тут 188 – кількість інформаційних байтів в пакеті транспортного потоку MPEG-2, 204 – кількість байт в пакеті після додавання перевірочних символів, 8 – число коректованих байт. Таким чином, в якості кодових комбінацій беруться цілі пакети, що містять 188 байт, а пакети, які додаються, – це перевірочні символи, що містять 16 байт.

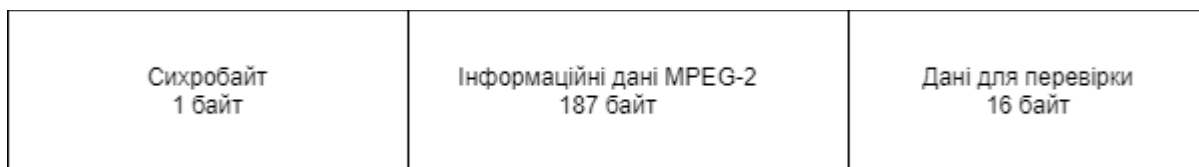


Рисунок 1.1 – Пакет даних MPEG-2 з захистом від помилок

Побудова кодового слова, захищеного кодами РС відбувається наступним чином. Формування циклічних кодових слів породжує многочлен $g(x)$, максимальний ступінь якого дорівнює числу перевірочних символів nk (для телебачення $nk = 2t = 16$)[6]. Процедура знаходження кодового слова $z(x)$ полягає в множенні вихідного інформаційного многочлена $i(x)$ на породжувачий многочлен коду: $z(x) = g(x) i(x)$. Синдроми помилок виходять розподілом кодового слова на породжувачі многочлени. Якщо кодові слова діляться на породжувачий многочлен без залишку, то

помилки немає. Прийняті слова, які не діляться без залишку на породжувачий многочлен, не є дозволеними і, отже, містять помилки. При наявності залишку ділення – включається процедура виправлення помилки.

Кодер Ріда – Соломона здійснює розподіл многочлена $v(x)$, що відображає прийняте слово, на породжувачий многочлен $g(x)$. Отриманий залишок від ділення приписується до $v(x)$ в якості молодших розрядів і це кодове слово йде на передачу. Розподіл в кодері реалізується за допомогою множення та складання.

Декодування кодів РС відбувається наступним чином. У приймачі для кожного прийнятого транспортного пакета, що містить 204 байта, обчислюються синдроми помилки і два полінома: "локатор" $L(x)$, корінь якого показує положення помилок, і "коректор", що дає значення помилок Y_i .

Обчислення синдрому помилки $S(x)$. Синдром є залишком ділення декодованого кодового слова $c(x)$ на породжувачий поліном $g(x)$. Якщо цей залишок дорівнює нулю, кодове слово вважається неспотвореним. Ненульовий залишок свідчить про наявність помилки.

Знаходження позицій символічних помилок (використовується алгоритм пошуку Ченя) виконується наступним чином. Відновлення даних здійснюється за допомогою рішення системи рівнянь з t невідомими. Для вирішення використовується швидкий алгоритм Форне[6].

Згорткові коди засновані на перетворенні вхідної послідовності двійкових символів в вихідну послідовність двійкових символів, у якій на кожен символ вхідної послідовності формується більш одного символу вихідної послідовності [1;3]. На відміну від блокових, вони не ділять інформацію на фрагменти і працюють з нею як із суцільним потоком даних.

Згорткове кодування – це ітеративна обробка потоку бітів, що створює залежність кожного біта від кількох попередніх, що дозволяє відтворювати передану інформацію.

Згорткове кодування задають за допомогою породжувачих многочленів, які визначають структуру двійкового згортального кодера. Кодове слово на

виході такого кодера складається у вигляді в двох послідовностей, які в двійковій формі представляють коефіцієнти відповідних породжуючих многочленів. Приклад згорткового кодера наведено на рис.1.2. Він містить трьохрозрядний зсувний регістр, на вхід якого надходить вхідна послідовність двійкових символів та 2 x суматорів за модулем 2.

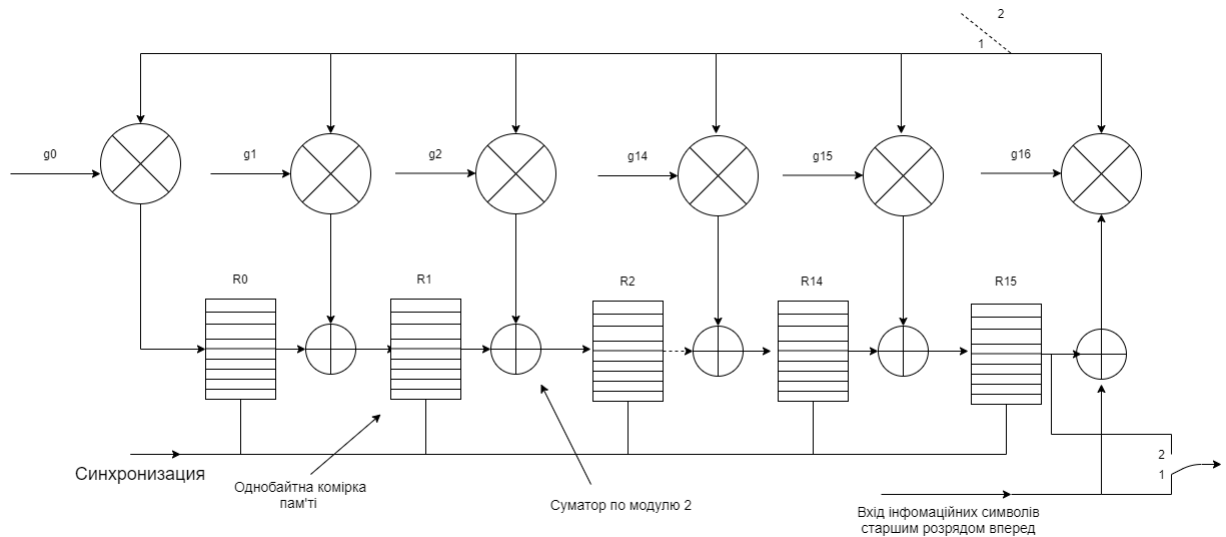


Рисунок 1.2 – Структурна схема кодера Ріда-Соломона:

$g_0 - g_{16}$ – коефіцієнти породжує многочлена $g(x)$

Інформаційна послідовність надходить на вхід регістра. Вміст регістра обробляється суматорами, на виходах яких утворюється кодова послідовність, яка символ за символом надходить на вихід. Відносна швидкість цього коду зменшується ($R = 1/2$), а тактова частота зростає в два рази.

Декодування згорткових кодів. Декодування реалізує алгоритм Вітербі, заснований на принципі максимальної правдоподібності. Алгоритм включає в себе обчислювальні заходи подібності (або відстані Хеммінга), між сигналом на вході декодера і на його виході. Декодер приймає рішення шляхом виключення менш ймовірних шляхів. Згорткові коди ефективно

працюють в каналі з білим шумом, але погано справляються з пакетами помилок.

Більш того, якщо декодер помиляється, на його виході виникає пакет помилок, який виправляється кодом РС.[7] Згорткові коди використовуються при низькому відношенні сигнал шум, коли виправляючої здатності блокових кодів при розумній довжині блоку виявляється недостатньо. В DVB - T2 також як і в DVB-T застосовується каскадне кодування. В якості зовнішнього коду використовується код Боуза Чоудхурі Хоквінгема (БЧХ), а в якості внутрішнього – код з малою щільністю перевірок на парність (LDPC). Коди Боуза Чоудхурі Хоквінгема, коди БЧХ, виконавчі, лінійні, циклічні, блокові виправляють кратні помилки, тобто: дві і більше помилок в кодовій комбінації.

Метод побудови кодів БЧХ відрізняється від метода звичайних циклічних, вибором визначального породжуючого полінома $g(x)$. Загальна властивість кодових слів циклічного коду: їх подільність без залишку на многочлен $g(x)$. Якщо при діленні є залишок, то сталася помилка. Систематичне кодування здійснюється шляхом "дописування" до кодованого слова залишку від ділення $x^n - k u(x)$ на $g(x)$, тобто $v(x) = x^n - k u(x) + [x^n - k u(x) \bmod g(x)]$.

В DVB T2 і DVB S2 застосовується систематичне кодування. При систематичному кодуванні кодове слово утворюється з двох підблоків: інформаційного та перевірного (як і в кодах РС)[7]. При цьому інформаційний підблок повторює вид інформаційного вихідного слова. У процесі систематичного кодування в кожному кодовому слові на перших k позиціях, відповідними старшими ступенями змінної x , будуть розташовані інформаційні символи, де n – загальне число символів в слові.

Для кодування досить знайти залишок $s(x)$ від ділення полінома $u(x)$ кодованого слова, помноженого на $x^n - k$, який породжується поліном $g(x)$: $s(x) = x^n - k u(x) \bmod g(x)$.

Потім цей залишок приписується слідом за останнім інформаційним символом. Для кодування кодами БЧХ застосовуються ті ж самі методи і схеми, що і для кодування іншими циклічними кодами. Відмінність полягає у виборі породжуючого полінома $g(x)$. БЧХ кодування здійснюється за схемою, подібною кодеру РС, але замість байтових осередків застосовуються D тригери.

Декодер БЧХ складається з логічних схем і обробних блоків, що реалізують такі задачі:

- 1) обчислення синдрому, шляхом обчислення значення прийнятого полінома в нулях коду;
- 2) знаходження коефіцієнту многочлена локаторів помилок $\sigma(x)$;
- 3) знаходження зворотних величин коренів локатора помилок $\sigma(x)$, тобто позицій помилок (за методом Ченя).
- 4) виправлення прийнятого слова на обчислених позиціях для значень помилок.

LDPC – код з малою щільністю перевірок на парність або нізкоплотностний код з перевіркою на парність. Це – лінійний блоковий систематичний код, що задається за допомогою породжуючої матриці і з ітеративним декодуванням.

Перевірка парності – це простий метод для виявлення помилок в переданому пакеті даних. З її допомогою можна відновити дані. Можна тільки виявити одиночну помилку. Тому при кодуванні і декодуванні LDPC для визначення розряду, в якому сталася помилка, використовують спеціальні перевірочні матриці, що дозволяють знайти синдроми помилок в будь-якому розряді кодового слова. Кодове слово LDPC формується множенням вихідної інформаційного слова на породжуючу матрицю G . LDPC коди описуються матрицею з низькою щільністю, що містить в основному нулі і відносно малу кількість одиниць.

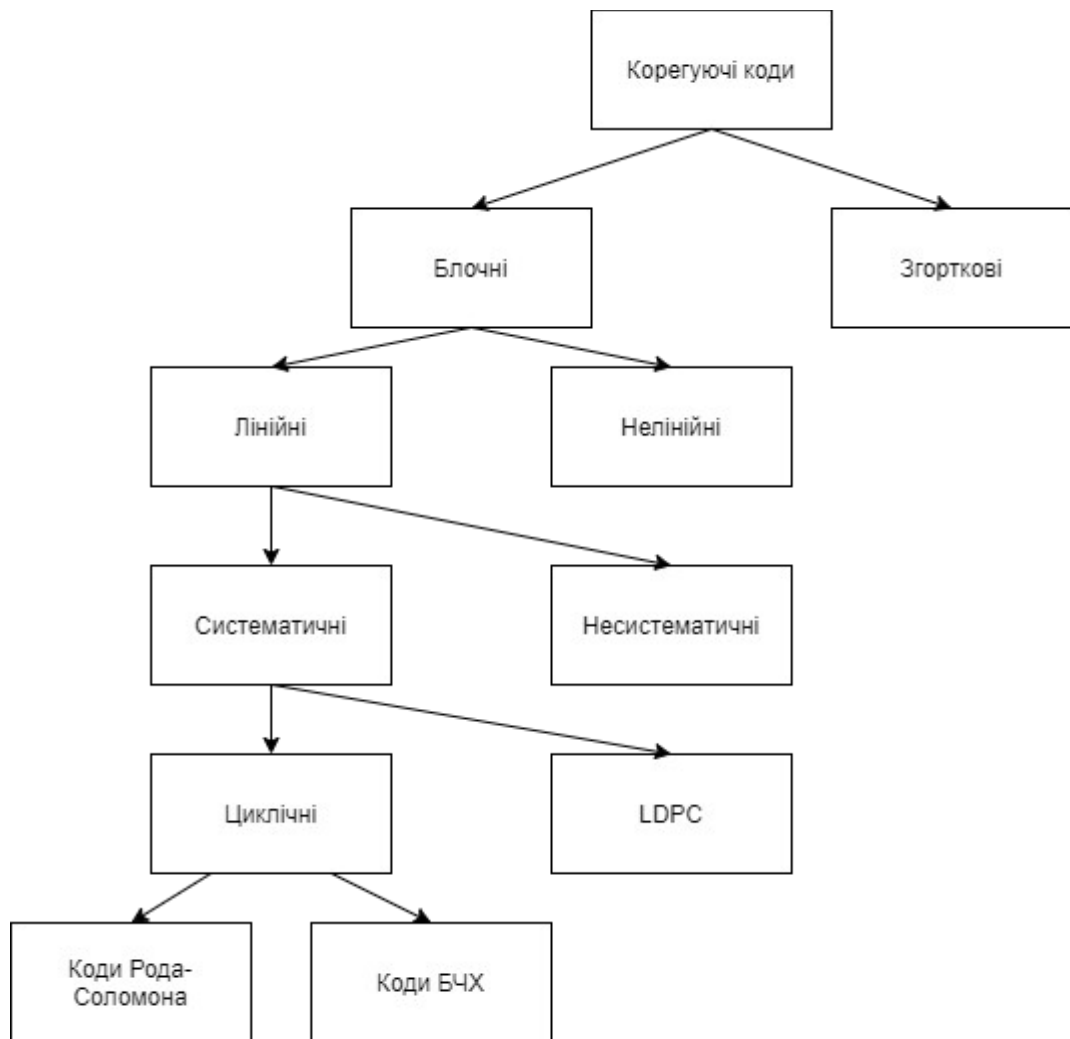


Рисунок 1.3 – Корегуючі коди

Коригуючий код в DVB T2 також є каскадним кодом. В якості зовнішнього коду в ньому застосований код БЧХ, а в якості внутрішнього – LDPC. Залежно від швидкості кодування LDPC, розмір вхідного блоку даних для БЧХ кодера може відрізнятися, однак вихідний розмір кодового слова після LDPC завжди становить 64800 біт (8100 байт) або 16200 біт (2025 байт).

LDPC коди ефективні в системах передачі інформації, що вимагають максимальної швидкості передачі при обмеженій смузі частот[8]. Використання LDPC / БЧХ кодів є більш ефективно, в порівнянні з каскадами кодів Ріда-Соломона і згорткового коду, і, тим самим, дає можливість знизити відношення сигнал/шум для надійної роботи системи.

При використанні LDPC і БЧХ вдається досягти виграшу в кілька децибел щодо сигнал/шуму. Але з іншого боку, LDPC/БЧХ-спосіб кодування вимагає більшої продуктивності і в першу чергу приймальних пристроїв, що призведе до більш високої вартості приймачів / декодерів.

Передані дані DVB-T2 і DVB-S2 пакуються в ВВ кадри, заголовок яких містить інформацію про характер даних. Кадр ВВ є контейнером для переданого потоку даних – транспортного потоку MPEG або будь-якого іншого.

Кадр ВВ разом зі своїм заголовком і вмістом розглядається як слово, до якого застосовуються коди LDPC і БЧХ. В результаті їх застосування виходить кадр FEC, який може мати довжину або 64800, або 16200 біт.

Застосування LDPC БЧХ-кодів в DVB-T2 дозволяє ефективно застосовувати режим модуляції 256QAM. Це збільшує ємність каналу на третину. Перешкодостійкість LDPC і БЧХ-кодів настільки висока, що вони справляються з компенсацією помилок, що виникають при використуванні 256QAM, без збільшення необхідного відношення сигнал/шум. Стандартом DVB-T2 є вимоги до завадостійкості які визначені величиною ймовірності помилки на біт $P_b = 10^{-7}$. Це значення досягається за допомогою каскадної конструкції LDPC і коду БЧХ.

Кодування істотно розширює тривалість кадру. В рамках стандарту DVB-T2 частка контрольних біт завадозахистних кодів може коливатися від 15 до 50%. Системи DVB-T2, DVB-S2 і DVB-C2 на базі LDPC / БЧХ показали істотне підвищення перешкодозахищеності в порівнянні з захистом, використовуваним в DVB-T. Виграш в рівні сигнал/шум за рахунок нового FEC може становити до 3 дБ для типового рівня помилок і при однаковій частці контрольних символів. Це поліпшення дозволяє підвищити пропускну здатність каналу приблизно на 30% (наприклад, за рахунок застосування більш високого рівня констеляції). Або 30-відсотковий приріст в потужності, в порівнянні зі стандартом DVB-S.

Спільне використання квадратурної фазової модуляції QPSK і каскадного кодування для каналу на базі коду РС і згоркових кодів в поєднанні з алгоритмом декодування Вітербі з м'яким рішенням забезпечує високу стійкість системи в умовах впливу шумових перешкод (DVB-S). При цьому гарантується не більше однієї помилки на годину, що еквівалентно ймовірності помилок близько 10^{-5} на вході демультимплексора MPEG-2 в приймальному-декодері, а в стандарті DVB-T2 - 10^{-12} . Застосування нових кодів LDPC і BCH замість кодів Ріда-Соломона і згорткових кодів істотно підвищує стійкість систем DVB-T2, DVB-S2 і DVB-C2.

1.2 Згорткове кодування

Кодування згортковим кодом є лінійним перетворенням інформаційної послідовності, в загальному випадку, необмеженої довжини. При цьому введення надмірності полягає в тому, що кожна розглянута група кодових символів виявляється функцією не тільки поточної групи інформаційних символів, але і ряду попередніх груп інформаційних символів.

При класичному згортковому кодуванні чергова група з k двійкових символів повідомлення (інформаційних символів) шляхом лінійних перетворень відображається в групу з двоїчних кодових символів ($n > k$). При цьому, на відміну від лінійних блокових кодів, при формуванні заданої групи кодових символів використовується не тільки дана група інформаційних символів, але також і $k'-1$ попередніх груп. Таким образом, згортковий код (ЗК) характеризується трьома основними параметрами: n , k , k' . Величину k' називають довжиною кодового обмеження (англ. Constraint length).

$k \times k'$ визначає максимальне число двійкових інформаційних символів, що беруть участь у формуванні групи, що складається з n двійкових кодових символів. Будемо позначати далі такий код як (n, k, k') [9]. При кодуванні вхідної послідовності необмеженої довжини кожен з інформаційних символів бере участь в формуванні кодових символів, складових k' груп, включаючи

групу, що формується безпосередньо в момент появи розглянутого інформаційного символу, і $K'-1$ наступної за нею групи. В цьому сенсі даний інформаційний символ виявляється «захищеним» кодовими групами, що і використовується при відповідних методах декодування [10].

Розглянемо ЗК з параметром $k = 1$. На рисунку 1.4 наведена структура кодера з параметрами $n = 2$, $k = 1$, $K' = 3$. Основу кодера становить регістр зсуву з $v = K' - 1 = 2$ осередками пам'яті, які позначені як z^{-1} .

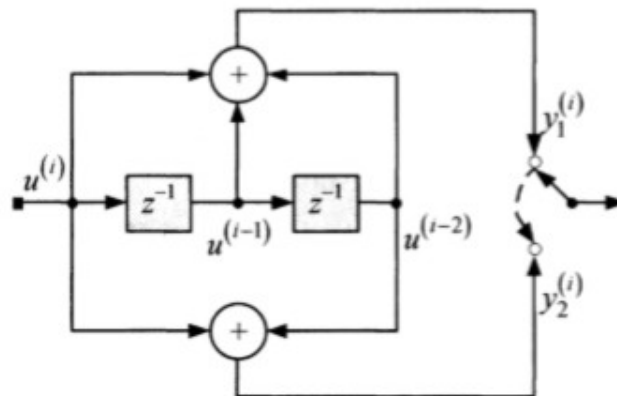


Рисунок 1.4 – Згортковий кодер ($n = 2$, $k = 1$, $K' = 3$)

У відповідь на кожен сигнал, що надійшов в кодер в момент часу t і інформаційний символ $u_1(i) = u(i)$ кодер формує групу з двох символів: $y_1(i)$ і $y_2(i)$ [10]. При цьому змінюється вміст комірок пам'яті і, тим самим, кодер переходить в новий стан.

Граф переходу між станами зображений на рисунку 1.5:

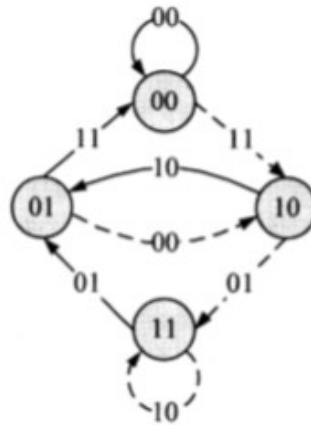


Рисунок 1.5 – Граф переходу згорткового кодера ($n = 2, k = 1, K = 3$)

Як видно з рисунку 1.5, значення кодових символів є лінійними комбінаціями (по модулю 2) даного вхідного символу і двох попередніх:

$$y_1(i) = u(i) + u(i-1) + u(i-2), \quad y_2(i) = u(i) + u(i-2).$$

Вид подібних комбінацій прийнято описувати породжуючими поліномами, кожен з яких відповідає певному кодовому символу. В даному випадку породжуючі поліноми мають вигляд:

$$g_1(z) = 1 + z^{-1} + z^{-2}, \quad g_2(z) = 1 + z^{-2}.$$

Також діаграму станів кодера можна навести у наступному виді (рисунок 1.6).

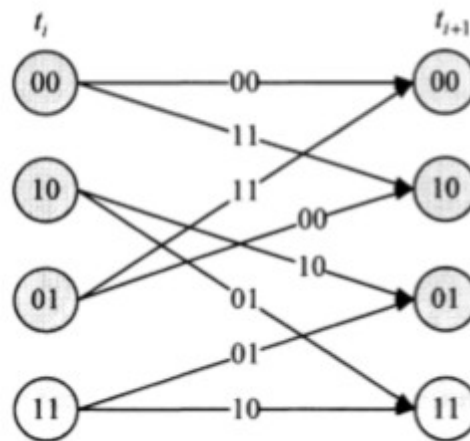


Рисунок 1.6 – Гратчастий граф діаграми станів кодера
($N = 2, k = 1, K' = 3$)

На підставі гратчастої форми діаграми станів кодера легко отримати зручне представлення процесу згорткового кодування, а саме гратчасту діаграму, зображену на рисунку 1.7:

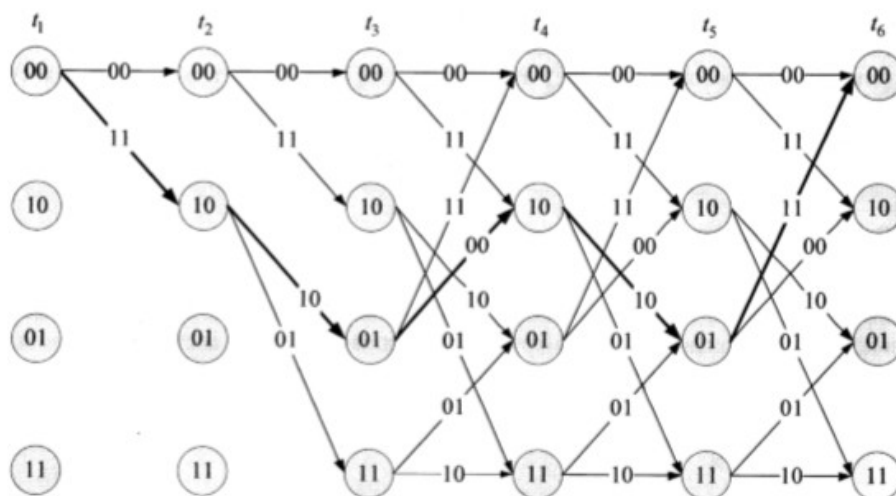


Рисунок 1.7 – Гратчаста діаграма станів кодера ($n = 2, k = 1, K' = 3$)

Вузли решітки, як і вузли на діаграмі станів, відповідають станам кодера, а вузли, що виходять з вузлів ребра – вхідним символам 0 (верхня гілка) і 1 (нижня)[11]. Жирною лінією на гратчастій діаграмі показаний шлях,

відповідний «ланцюжку», який формується з кодових комбінацій при згортковому кодуванні вхідної послідовності 10100.

1.3 Декодування згорткових кодів

Сутність алгоритму декодування Вітербі складається в покроковому порівнянні всіх можливих шляхів (послідовності переданих символів) на гратчастої діаграмі декодера, аналогічної з спостережуваної послідовністю «жорстких» або «м'яких» рішень демодулятора. При цьому кожен черговий крок роботи алгоритму безпосередньо пов'язаний з надходженням в декодер чергової групи рішень демодулятора, що відповідають групі переданих кодових символів. В результаті, на кожному кроці з усіх шляхів, що входять в кожен з вузлів діаграми декодера і відповідних можливим шляхах на діаграмі кодера, виключаються з подальшого аналізу (на наступних кроках) всі шляхи, окрім того, який виявляється на найменшій хеммінговій (або Евклідовій) відстані.

Рішення по переданому символу може бути прийнято при обмеженому числі кроків алгоритму. Саме така властивість коду дозволяє використовувати алгоритм Вітербі для декодування згортального коду на гратчастій діаграмі довільно великої довжини при обмеженому числі осередків пам'яті для зберігання збережених шляхів. При цьому число двійкових елементів пам'яті декодера прямо пропорційно числу станів кодера і значенню параметра W , має сенс розміри зображення решітки, часто званого глибиною декодування .

Для жорсткого (дворівневого) декодування кожен парю прийнятих кодових символів можна зобразити на площині у вигляді одного з кутів квадрата, як показано на рисунку 1.8 (а). Кути позначені двійковими числами $(0, 0)$, $(0, 1)$, $(1, 0)$ і $(1, 1)$, що представляють чотири можливих значення, які можуть приймати два кодових символи в жорсткій схемі прийняття рішень[12]. Аналогічно для 8-рівневого м'якого декодування кожен парю

кодових символів можна відобразити на площині у вигляді квадрата розміром 8×8 , що складається з 64 точок, як показано на рисунку 18 (б). В цьому випадку демодулятор більше не видає жорстких рішень; він видає квантовані сигнали з шумом (м'яка схема прийняття рішень).

Основна відмінність між м'яким і жорстким декодуванням за алгоритмом Вітербі полягає в тому, що в м'якій схемі не використовується метрика відстані Хеммінга, оскільки вона має обмежений дозвіл. Метрика відстаней, яка має потрібний дозвіл, називається евклідовою кодовою відстанню, тому далі, щоб полегшити її застосування, відповідним чином перетворимо двійкові числа з одиниць і нулів в восьмеричні числа від 0 до 7.

Це можна побачити на рисунку 1.8 (в), де відповідним чином позначені кути квадрата; тепер для опису будь-якої з 64 точок необхідно користуватися парами цілих чисел від 0 до 7. На рисунку 1.8 (в) також зображена точка (5,4), що представляє приклад пари значень кодових символів з шумом. Квадрат на рисунку 1.8 (в) зображений в координатах (x, y). евклідова кодова відстань між точкою з шумом (5,4) і точкою без шуму (0,0) дорівнює

$$\sqrt{(5-0)^2 + (4-0)^2} = \sqrt{41}$$

. Для того, щоб дізнатися евклідова кодова відстань між точкою з шумом (5,4) і точкою без шуму (7,7), необхідно обчислити

$$\sqrt{(5-7)^2 + (4-7)^2} = \sqrt{13}$$

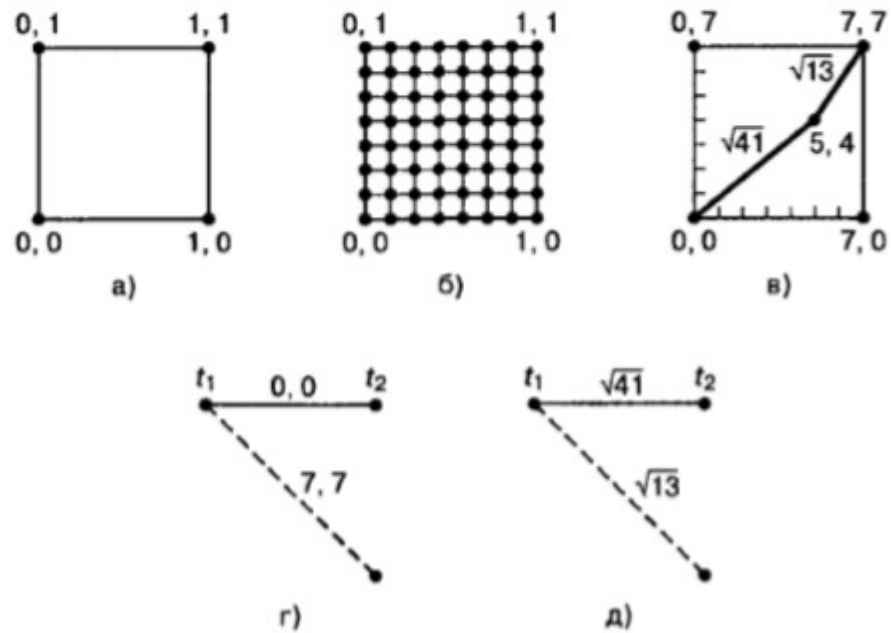


Рисунок 1.8 – Декодування Вітербі: а) площину жорсткої схеми прийняття рішень; б) 8-рівнева схема м'якої схеми прийняття рішень; в) приклад м'яких кодових символів; г) секція решітки кодування; д) секція решітки декодування

М'яке декодування за алгоритмом Вітербі, здебільшого, здійснюється так само, як і жорстке декодування. єдина відмінність полягає в тому, що тут не використовується відстань Хеммінга. Тому розглянемо м'яке декодування, здійснюване з евклідовим кодовою відстанню. На рисунку 1.8 (г) показана перша секція решітки кодування.

При цьому кодові слова перетворені з двоїчних в восьмиричні. Припустимо, що пара кодових символів, що надійшла на декодер під час першого переходу, згідно м'якої схемою декодування має значення 5 та 4. На рисунку 1.8 (д) показана перша секція решітки декодування. Метрика $(\sqrt{41})$, що представляє евклідова кодова відстань між прибулим кодовим словом 5,4 і кодовою словом 0,0, позначена суцільною лінією. Аналогічно метрика $(\sqrt{13})$ є евклідова кодове відстань між кодовим символом 5,4 і кодовим символом 7,7; яку показано пунктирною лінією.

Частина, що залишилася задачі декодування, яка зводиться до відсікання решітки та пошуку повної гілки, здійснюється аналогічно схемі жорсткого декодування. В реальних мікросхемах, призначених для згорткового декодування, евклідова кодова відстань насправді не застосовується, замість нього використовується монотонна метрика, яка володіє подібними властивостями, але значно простіше в реалізації. Прикладом такої метрики є квадрат евклідової кодової відстані, в цьому випадку виключається розглянута вище операція взяття квадратного кореня.

1.4 QPSK модулятор

Модулятор QPSK – це пристрій, що перетворює вхідні дані бітового потоку в форму сигналу, придатну для передачі по каналу зв'язку.

Принцип роботи модулятора QPSK – в квадратурно-фазовій маніпуляції (QPSK): два послідовні біти в послідовності даних групуються разом. Це зменшує бітову швидкість передачі сигналів (f_b) і, отже, зменшує пропускну здатність каналу.

Однополярний двійковий потік повідомлень має швидкість передачі даних R_b і спочатку перетворюється в біполярну послідовність неповернення до нуля (NRZ) за допомогою однополярного - біполярного перетворювача. Потім бітовий потік $m(t)$ розбивається на два бітові потоки $m_I(t)$ і $m_Q(t)$ (у фазовому та квадратурному потоках. Бітовий потік $m_I(t)$ називається «парний потік», а $m_Q(t)$ називається «непарним потоком». Дві двійкові послідовності модулюються окремо двома носіями $\varphi_1(t)$ і $\varphi_2(t)$, які знаходяться в квадратурі. Два модульованих сигнали, кожен з яких можна вважати сигналом BPSK, є підсумком для отримання сигналу QPSK. Фільтр на вихід модулятора обмежує спектр потужності сигналу QPSK в межах виділеної смуги. Це запобігає зменшенню енергії сигналу на сусідніх каналах, а також усуває помилки поза діапазоном сигнали, що генеруються в процесі модуляції. У більшості реалізацій формування імпульсів здійснюється за

адресою базової смуги для забезпечення належної ВЧ-фільтрації на вихід передавача.

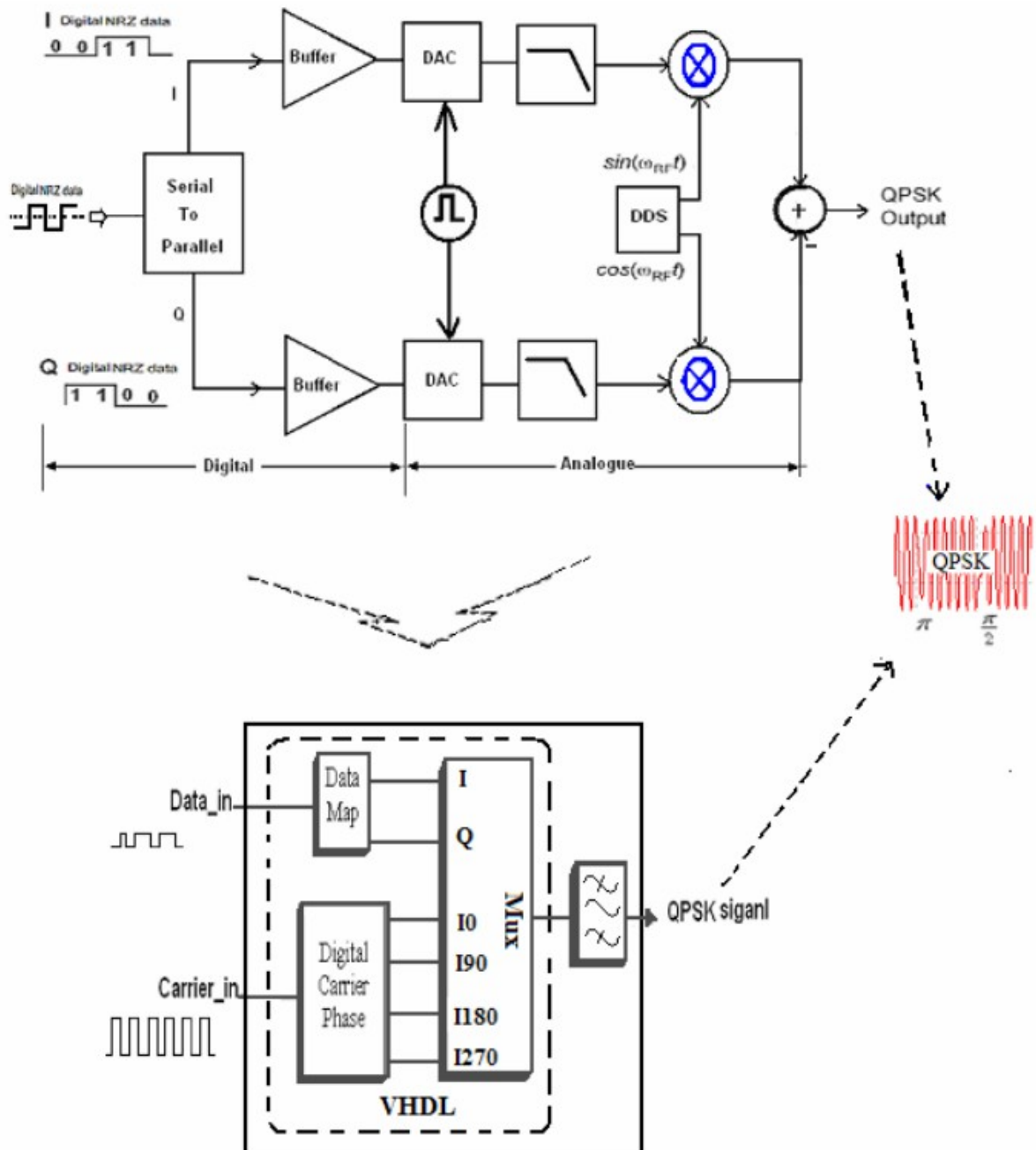


Рисунок 1.9 – Блочна діграма QPSK модулятора

На рис. 1.10 $m_1(t)$ і $m_2(t)$ – сигнали повідомлення, Q-ch і I-ch - сигнали Q-каналу, що відповідно (t) рівні модульованому сигналу.

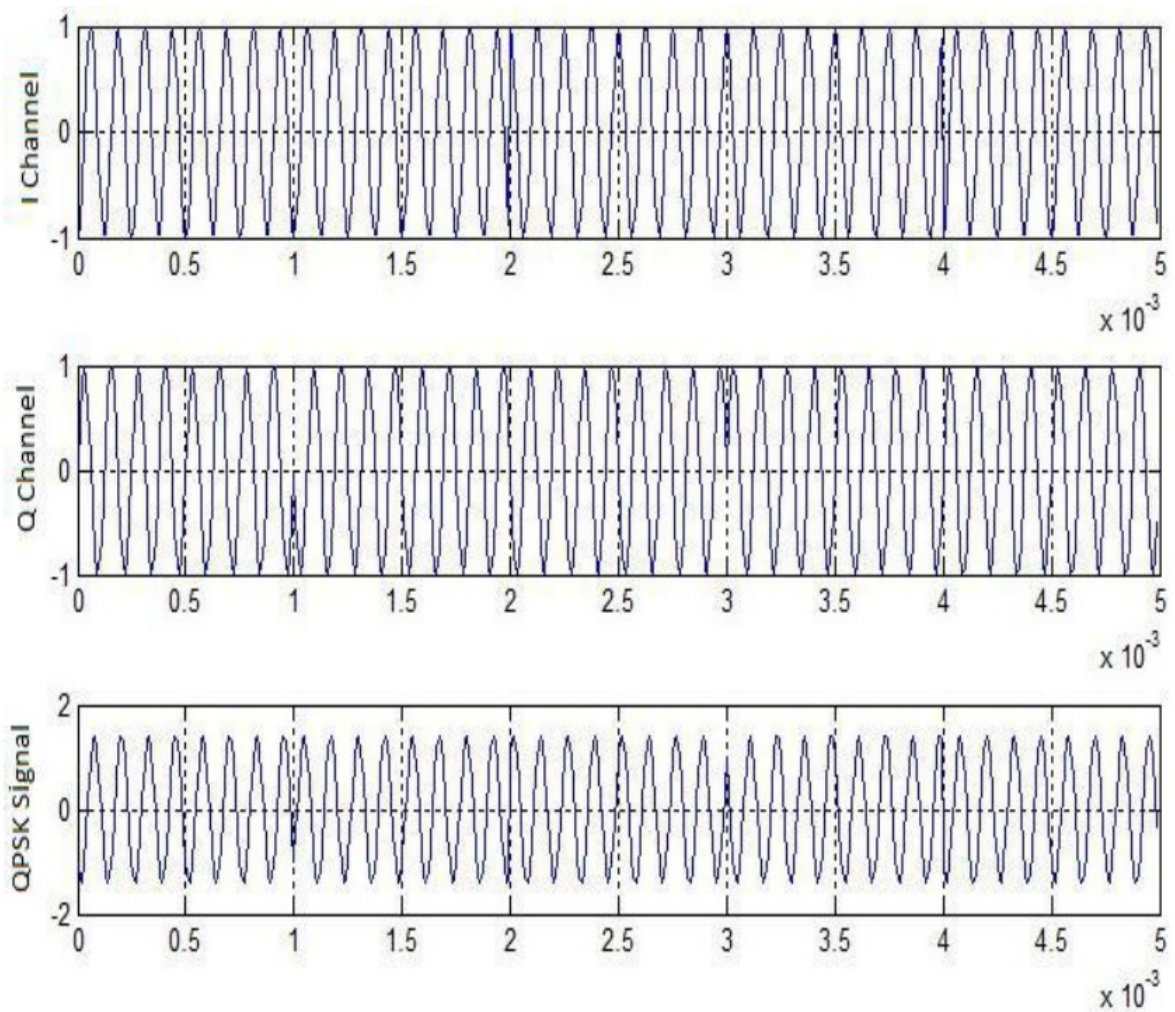


Рисунок 1.10 – Сигнал після модуляції QPSK

1.5 QPSK демодулятор

Демодуляція – це акт вилучення інформаційного сигналу із модульованої несучої.[14]

Перед початком демодуляції необхідно виділити позасмуговий шум за допомогою фільтру. Після цього відфільтрований вихід, розділений на дві частини, когерентно (окремо кожна) демодулюється за допомогою фазової та квадратурної несучої. Когерентна несуча, що використовуються для демодуляції, отримується з прийнятого сигналу за допомогою схем відновлення несучої. Виходи демодуляторів проходять через схеми прийняття рішень, які генерують фазу і квадратурні двійкові потоки. Ці два

компоненти потім мультиплекуються для відтворення вихідного двійкового потоку прийнятих даних.

1.6 Програмована елементна база

Декодери в DVB-T приймачах реалізовані у вигляді чіпсетів, а саме, набору спеціалізованих інтегральних схем ASIC (Application Specific Integrated Circuit). Всі методи реалізації систем-на- кристалі (SoC, System-on-Chip) лежать між двома полюсами – розробка ASIC і розробка системи на основі FPGA (Field Programmable Gate Arrays, програмована логічна інтегральна схема, ПЛІС). Вибір методу реалізації в кожному конкретному випадку визначається результатом оцінки цілого ряду факторів: передбачуваних обсягів виробництва, необхідної продуктивністю і функціональних характеристик мікросхем, тимчасових, інженерних і фінансових можливостей, а також ризиків, пов'язаних з тим або іншим способом реалізації.

До переваг ПЛІС можна віднести наявність готових бібліотек програм, описують складні алгоритми. Низька вартість засобів проектування роблять реалізацію проектів на основі FPGA вельми привабливим, а ризик проектування практично дорівнює нулю і зводиться тільки до перепрограмування ПЛІС.

Велика вартість систем проектування і складність виконання проектів, призводить до високої вартості самої ASIC. Тут дуже високий ризик проектних помилок, що призводять до необхідності перепроєктування. Таким чином, при використанні ASIC необхідно скорочувати час проектування і спрощувати процедуру розробки мікросхеми.

Час на розробку і виготовлення FPGA становить 10 тижнів, а на ASIC - 24 тижні. Також, хотілося б відзначити ще один недолік в існуючих реалізаціях. У декодера Вітербі вибираються досить високі значення глибини декодування, рівні 96 і 128.

У даній роботі показано, що при виборі глибини декодування, що дорівнює 64, декодер не програє по завадостійкості і при цьому, виходить зменшити використувані ресурси ПЛІС для реалізації пристрою в 2 рази, що дасть можливість вибору більш дешевої ПЛІС з меншим об'ємом пам'яті.

1.7 Мова опису апаратури інтегральних схем VHDL

Мова VHDL (Very high speed integrated circuits Hardware Description Language) є фактично міжнародним стандартом в області автоматизації проектування цифрових систем, це вхідна мова багатьох сучасних систем автоматизованого проектування (САПР). VHDL призначений, в першу чергу, для специфікації – точного опису проєктованих систем і їх моделювання на початкових етапах проектування.

З VHDL можна моделювати електронні схеми з урахуванням реальних часових затримок. VHDL – це потужна мова, вона дозволяє описувати поведінку, тобто алгоритми функціонування цифрових систем, а також проводити ієрархічне функціонально-структурний систем і в той же час володіє можливостями мови програмування високого рівня – дозволяє створювати свої типи даних, має широкий набір арифметичних, логічних операцій і т.д.

Останнім часом досить успішно розробляються і системи синтезу схем по специфікаціям на цій мові. Використання САПР Quartus II [15] дозволяє вирішувати це завдання для ПЛІС, що випускаються фірмою Altera. А серед ModelSim дозволяє провести моделювання описів, представлених на мові VHDL.

2 РОЗРОБКА АЛГОРИТМІВ ТА ВИБІР АППАРАТНОЇ ПЛАТФОРМИ

2.1 Гнучкість декодера Вітербі

Як оцінки завадостійкості, будемо використовувати залежність величини до величини BER, тобто енергетичні витрати на передачу одного біта інформації в умовах впливу перешкод з спектральною щільністю середньої потужності, де BER (Bit Error Rate) – ймовірність помилки двійкового символу. В результаті моделювання даного декодера при виборі різної глибини декодування отримали наступні характеристики (рисунок 2.1):

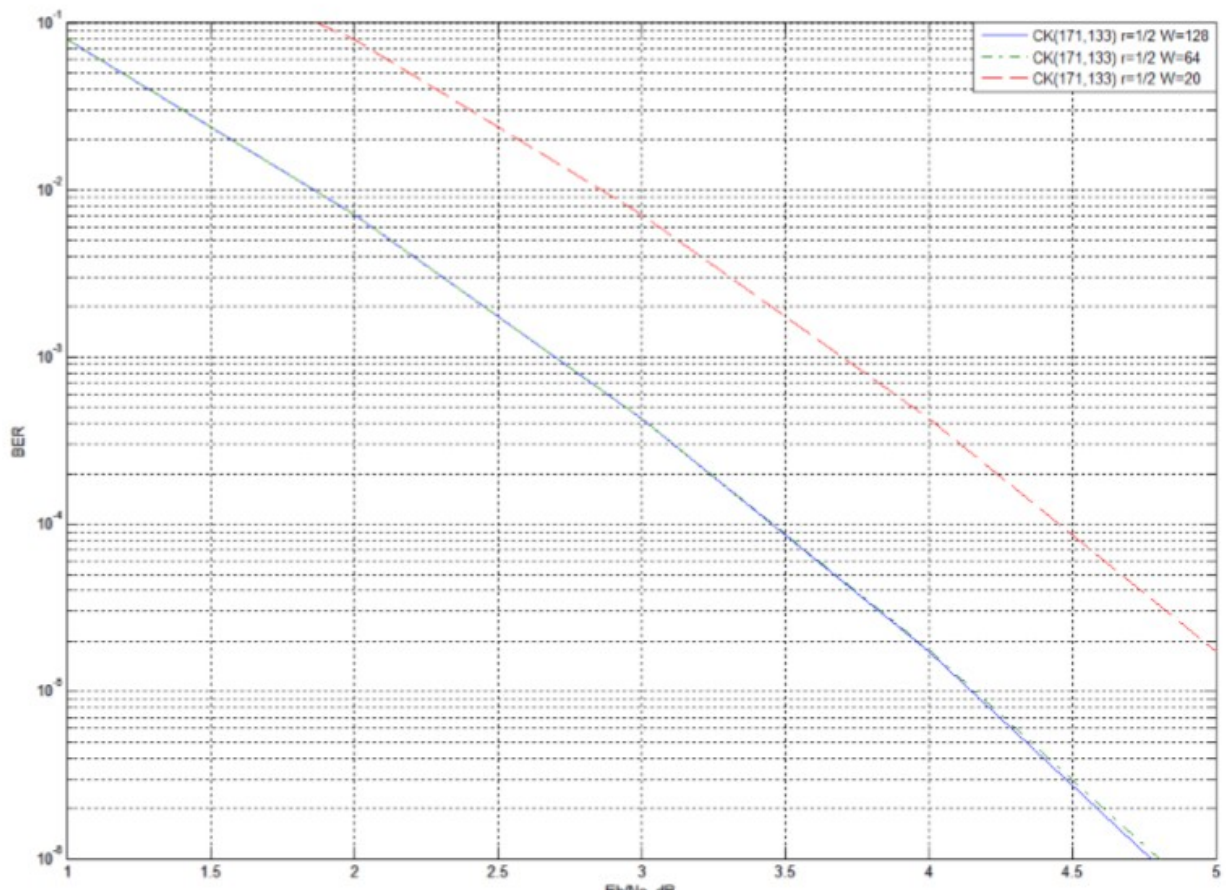


Рисунок 2.1 – Перешкодостійкість декодера Вітербі при різній глибині декодування (швидкість коду 1/2)

Наведені залежності були отримані при ймовірності 0,95 і інтервалі 0,8 дБ. З наведених кривих можна зробити наступний висновок, що глибину декодування W слід вибирати, використовуючи наступне співвідношення $W = (5..8) K'$, що узгоджується з теоретичними відомостями [10]. З графіка видно, що при значенні $W = 20$ отримуємо програвш 1дБ, в порівнянні з декодером, глибина якого дорівнює 128.

Однак, при подальшому збільшенні глибини декодування, не вдається отримати енергетичного виграшу.

Розглянемо стійкість декодера на різних швидкостях згортальних кодів, при різній глибині декодування 64 і 128 (рисунок 2.2).

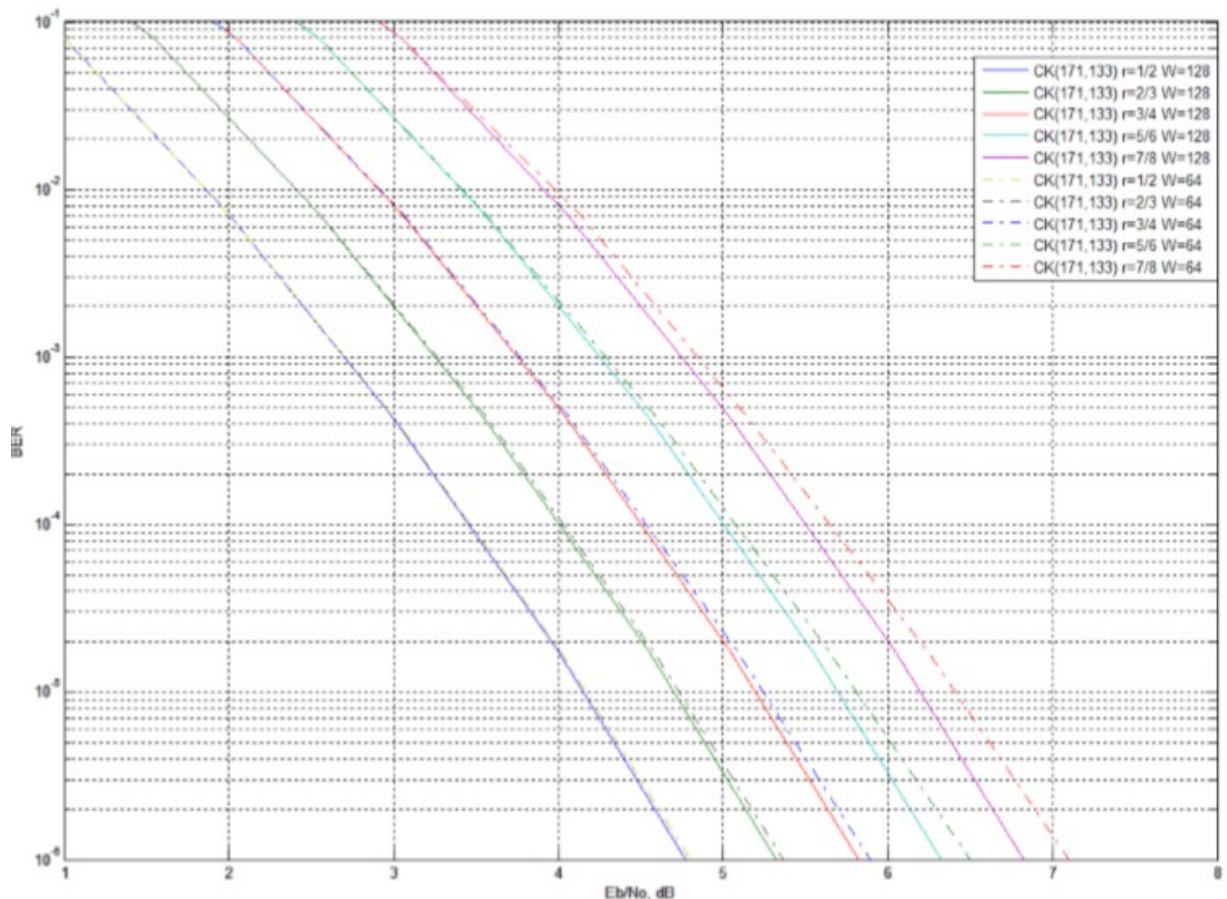


Рисунок 2.2 – Перешкодостійкість декодера при різних швидкостях згортальної коди і різній глибині декодування

Представлені залежності були отримані при ймовірності 0,95 і інтервалі 0,8 дБ. при обраному довірчому інтервалі моделювання не виявило відмінностей між кривими, отриманими при різній глибині декодування.

2.2 Реалізація декодера на ПЛІС

Величезна різноманітність сімейств ПЛІС таких популярних фірм як Altera, Xilinx, Actel, Lattice і ін. вимагає обґрунтування вибору фірми-виробника, а також сімейства для створення пристроїв, найкращим чином відповідають поставленим вимогам.

В результаті компіляції в середовищі QUARTUS II функції декодера Вітербі фірми Altera, при різній глибині декодування, отримали такі результати, наведені в таблиці 1. Компіляція проводилася на ПЛІС Cyclone II: EP2C15AF256A7.

Таблиця 2.1 – Результати компіляції

Глибина декодування	Використана пам'ять	Всього пам'яті на ПЛІС
60	57331	239616
80	76459	
100	95572	
120	114702	

Робимо висновок про те, що при зменшенні глибини декодування вдвічі, використовувана пам'ять ПЛІС зменшується в 2 рази.

Для реалізації декодера з глибиною декодування рівній 64 досить ПЛІС Cyclone, пам'яттю 59904 біт або Spartan-3, з пам'яттю 73728 біт.

Для реалізації декодера з глибиною декодування рівній 128 потрібні вже більш дорогі ПЛІС: Cyclone II, з пам'яттю 119808 біт і Spartan-3, з пам'яттю 221 184 біт.

Іншими словами перехід до меншої глибини декодування (від 128 до 64) знижує вартість елементної бази практично в 2 рази.

2.3 Порівняння завадостійкості декодера Вітербі при різній розрядності м'яких рішень.

В ході моделювання в Matlab декодера Вітербі при різній розрядності м'яких рішень були отримані результати, які наведені на рисунку 2.3.

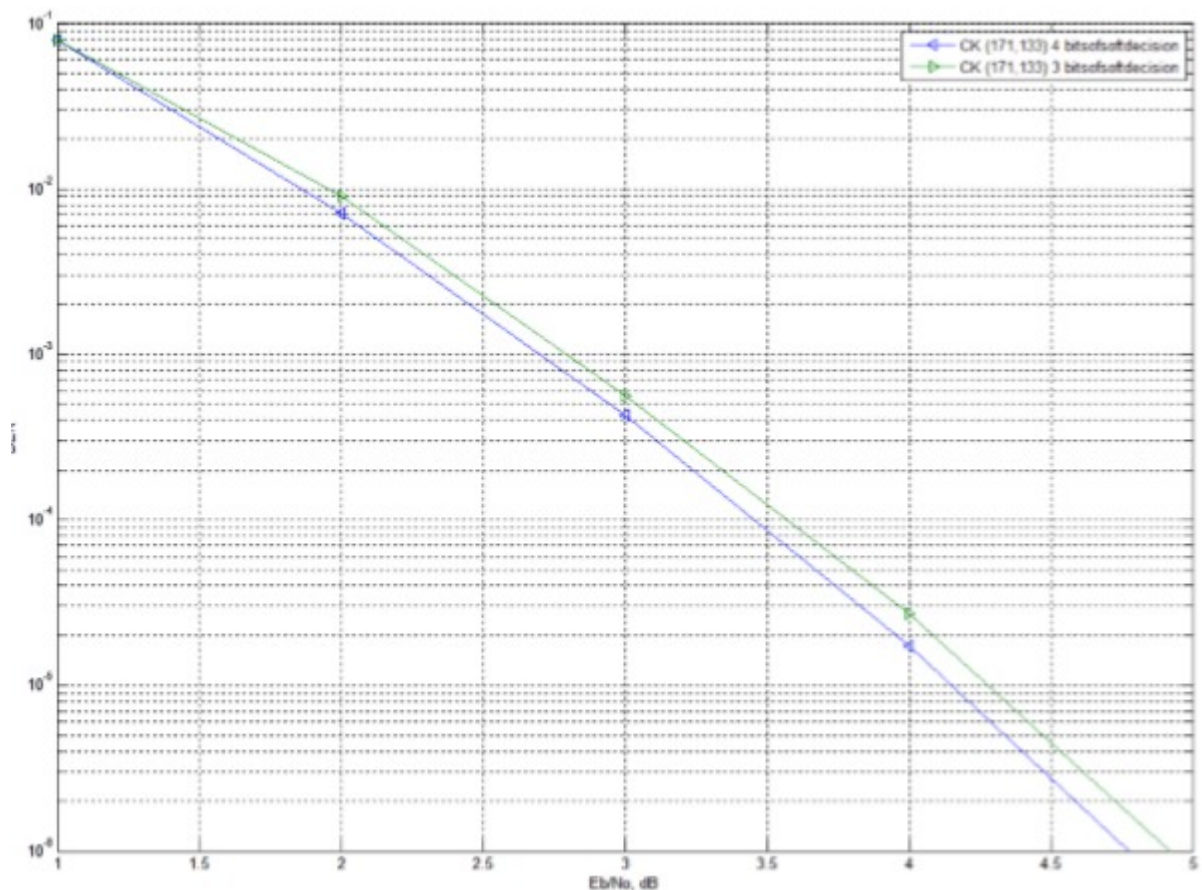


Рисунок 2.3 – Перешкодостійкість декодера при різній бітності м'яких рішень

Представлені залежності були отримані при довірчій ймовірності 0,95 і довірчому інтервалі 0,8 дБ[11]. З наведених залежностей можемо зробити висновок, що при переході від 3х-бітних м'яких рішень до 4х-бітовим не отримуємо енергетичного виграшу, при обраному довірчому інтервалі.

Формування згорткового кодування Вітербі за поліномами наведено в лістингу 2.1.

Лістинг 2.1 – Інтерфейс модулятора

```

if ( rising_edge(clk_for_in) ) then
    Iz0 <= I_In; Qz0 <= Q_In;
    Iz1 <= Iz0;   Qz1 <= Qz0;
    Iz2 <= Iz1;   Qz2 <= Qz1;
    Iz3 <= Iz2;   Qz3 <= Qz2;
    Iz4 <= Iz3;   Qz4 <= Qz3;
    Iz5 <= Iz4;   Qz5 <= Qz4;
    Iz6 <= Iz5;   Qz6 <= Qz5;

    if (mode = '1') then
        IS1 <= Iz0 xor Iz1 xor Iz2 xor Iz3 xor Iz6;
        IS2 <= Iz0 xor Iz2 xor Iz3 xor Iz5 xor Iz6;
        QS1 <= not(Qz0 xor Qz1 xor Qz2 xor Qz3 xor Qz6);
        QS2 <= not(Qz0 xor Qz2 xor Qz3 xor Qz5 xor Qz6);
    else
        IS1 <= Iz0 xor Iz1 xor Iz2 xor Iz3 xor Iz6;
        IS2 <= not(Iz0 xor Iz2 xor Iz3 xor Iz5 xor Iz6);
        QS1 <= Qz0 xor Qz1 xor Qz2 xor Qz3 xor Qz6;
        QS2 <= not(Qz0 xor Qz2 xor Qz3 xor Qz5 xor Qz6);
    end if;
    state <= '0';

end if;

```

2.4 Розробка QPSK модулятора

Всі аналогові або гібридні аналогово/цифрові модулятори QPSK працюють з кутом несучої фазового зсуву, як ключ модуляції. Фазовий сигнал є найважливішою частиною модулятора для отримання двох дискретних сигналів (синус і косинус). Однак формат є важливим для відображення I та Q [12, 13]. Аналоговий сигнал QPSK може бути представлений наступним чином:

$$QPSK(t) = I(t)\cos(2\pi fct) - Q(t)\sin(2\pi fct)$$

$$Q = \sqrt{\frac{2E}{T \sin\left[\frac{(2i-1)\pi}{4}\right]}}$$

QPSK модулятор був реалізований на мові VHDL. На листінгу 2.2 представлено загальну структуру модулятора.

Листінг 2.2 – Інтерфейс модулятора

```
entity QPSK_coder is
  Port (
    Fclk : in std_logic;
    StrDIBit : in std_logic;
    StrDIBit2 : in std_logic;
    I_In : in std_logic;
    Q_In : in std_logic;

    StrIQ_Out : out std_logic;
    I_Out : out std_logic;
    Q_Out : out std_logic
  );
```

```
end QPSK_coder;
```

Запропонований модулятор QHDK VHDL запрограмований генерувати несучу фазу, яка набуває чотирьох дискретних станів (0, 90, 180, 270). Два окремі потоки в фазі 'I' і квадратурна фаза Q для відображення даних для управління чотирма фазами різних несучих, пов'язаних з мультиплексором.

Вихід вибирається мультиплексором для забезпечення цифрового сигналу QPSK, який проходить через пасивний фільтр перед передачею (TX) для усунення високих частот.

Модулятор QPSK складається з несучого джерела для формування періодичного імпульсного сигналу (несучого сигналу), що подається на фазоперетворювач несучої; які зміщують вхідну несучу на чотири різні фазові сигнали (0°, 90°, 180°, 270°), пов'язані з мультиплексором. Поки data_in подається на відображення даних, щоб генерувати сигнали I та Q для впливу на чотири фази різних несучих.

Реалізація формування сигналу приведена на лістингу 2.3:

Лістинг 2.3 – Формування сигналу за поліномами

```
entity QPSK_MOD is
  port (
    -----Inputs-----
    rst      : in std_logic;
    clk      : in std_logic; --3/50 MHz
    I_in     : in std_logic;
    Q_in     : in std_logic;
    -----Outputs-----
    I_out    : out std_logic;
    Q_out    : out std_logic
  );
end QPSK_MOD;
```

Вихід вибирається мультиплексором, який забезпечує цифровий сигнал QPSK, цей сигнал фільтрується за допомогою аналогового фільтра перед передачею, щоб пропустити основну частоту і усуває більш високі частоти, пов'язані з квадратним сигналом.

Формування вихідного сигналу наведено в лістингу 2.4:

Листінг 2.4 – Формування вихідного сигналу за стробом

```
architecture QPSK_MOD_Arch of OQPSK_MOD is
    signal I_buff: std_logic;
    signal Q_buff: std_logic;

begin
    process(clk)
    begin
        if (rst = '1')then
            I_buff <= '0';
            Q_buff <= '0';
        elsif(rising_edge(clk))then
            I_buff <= I_in;
        end if;

        if (rst = '1')then
        elsif(falling_edge(clk))then
            Q_buff <= Q_in;
        end if;
    end process;
    I_out <= I_buff and I_buff;
    Q_out <= Q_buff ;
end QPSK_MOD_Arch;
```

Результати моделювання наведені на рисунку 2.4. Несуча частота 5 МГц генерувалася з локального тактового сигналу. Сигнал даних був зменшений до 5 кГц за допомогою дільника частоти, який потім подавався в генератор псевдовипадкових послідовностей. Сформований VHDL “Поведінковий” RTL модулятора QPSK наведений на рис 2.4.

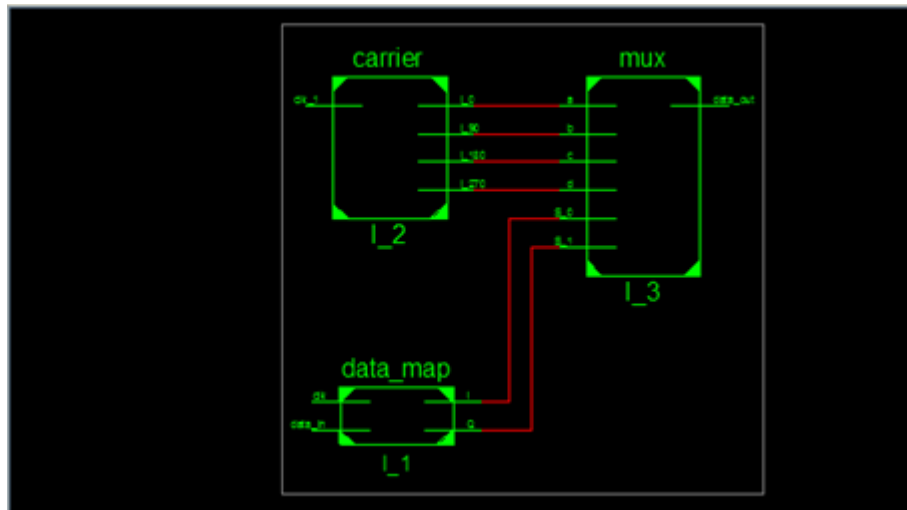


Рисунок 2.4 – Схема RTL для модулятора QPSK

На рисунку 2.5 показані результати моделювання по зростаючому і спадаючому краю вхідного сигналу[14]. Clk_2 – це сигнал carrier, clk_3 – тактовий імпульс для управління швидкістю передачі даних вхідного сигналу, data_in - вхідний сигнал, а data_out - вихідний сигнал.

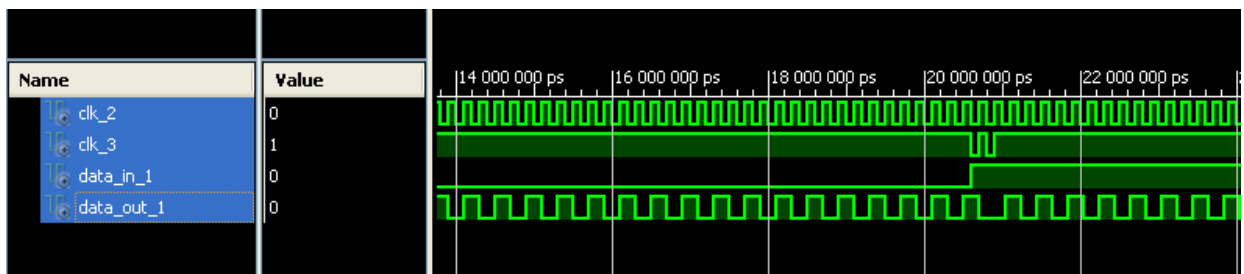


Рисунок 2.5 – Вихід на верхньому фронті

Тривалість імпульсу для clk_2 і clk_3 дорівнює 200 нс, а тривалість імпульсу для data_out – 400 нс. Залежність фазомодульованого сигналу від data_in при різних змінах фази наступна:

- 1) коли data_in дорівнює 00, фаза модульованого сигнал дорівнює 0° ;
- 2) коли data_in дорівнює 01, фаза модульованого сигналу дорівнює 90° ;
- 3) коли data_in дорівнює 10, фаза модульованого сигналу дорівнює 180° ;
- 4) коли data_in дорівнює 11, фаза модульованого сигналу дорівнює 270° .

Сигнал проходить як цифровий QPSK через пасивний ФНЧ для розділення гармонік. С.

На виході мультиплікатора формується форма «квадратного сигналу» цифрового сигналу QPSK. Важливо використовувати фільтр для завершення процесу модулятора. Для цього було розроблено аналоговий пасивний фільтр, оскільки він має нульове споживання енергії.

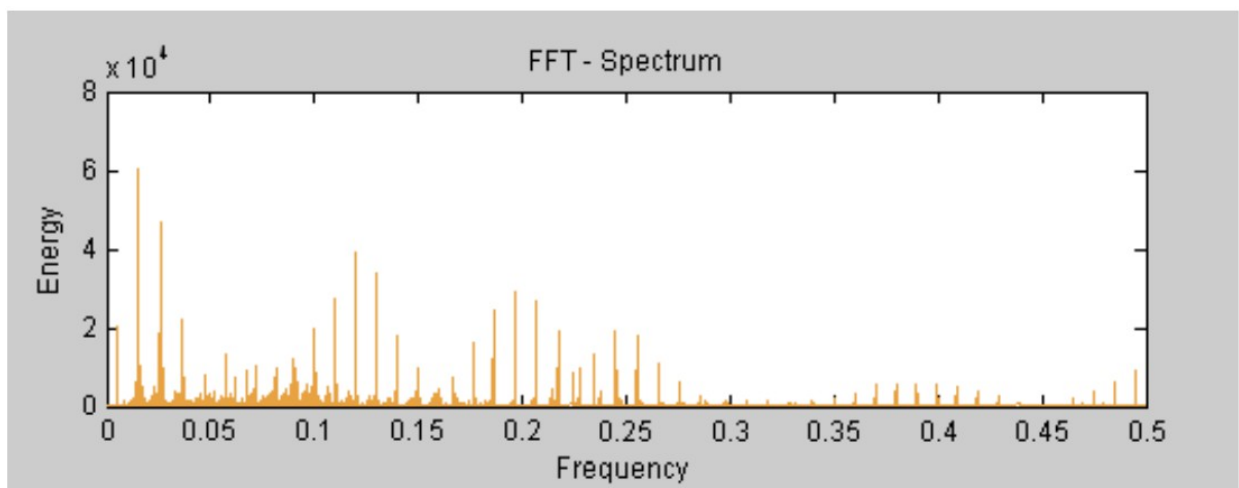


Рисунок 2.6 – Wavelet LPF

Були досліджені два типи фільтрів: фільтра низьких частот (Wavelet LPF) та смугового фільтра (Butterworth BPF другого порядку[15]) для усунення гармонік із цифрового сигналу QPSK(I). Моделювання BPF

Баттерворта Вибраним прототипом аналогового фільтра є фільтр Баттерворта 2-го порядку для фільтрації вхідного сигналу QPSK.

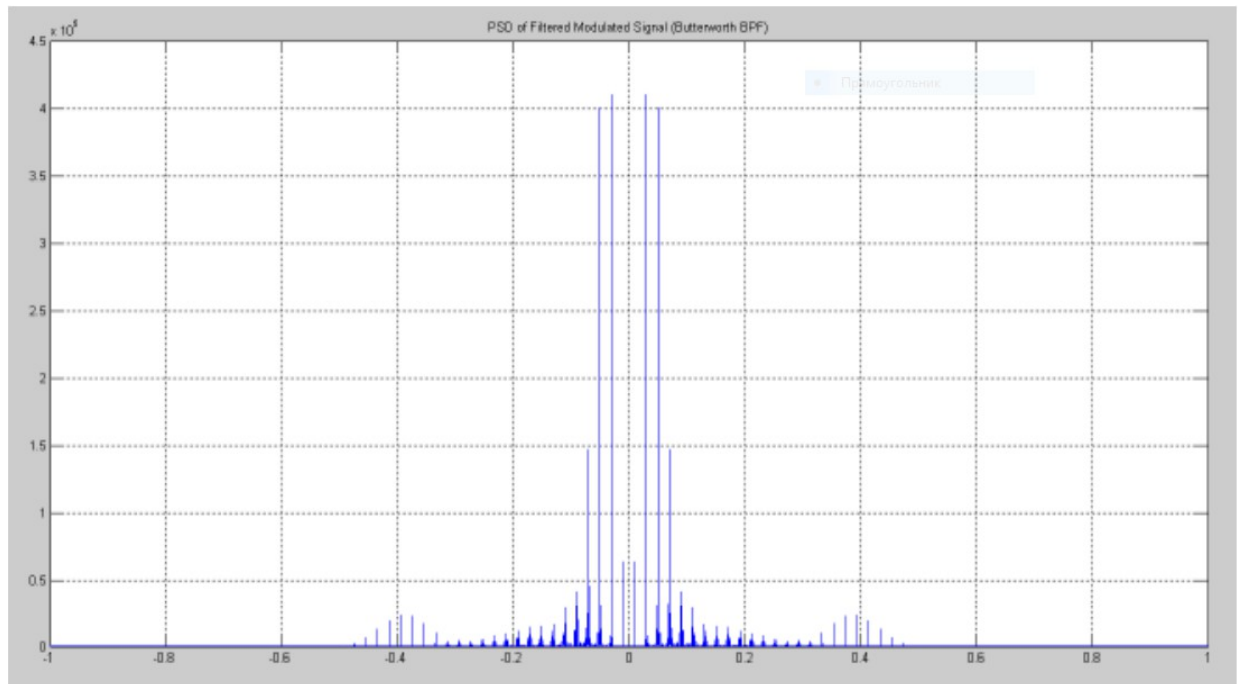


Рисунок 2.7 – Фільтрація Баттервортом

2.5 Моделювання системи з корекцією помилок DVB-T в середовищі Matlab

Перевірка спільної роботи декодера Вітербі і Ріда – Соломона була виконана в середовищі Matlab.

На рисунку 2.8 представлена модель цієї системи. Вона містить в собі джерело даних, кодер, канал зв'язку, декодер і осцилограф Scope.

Таблиця 2.2 – Порівняння симуляцій

Тип фільтру	Частота даних	Несуча частота	Вихідна частота
Butterworth BPF	500 Kbps	5 MHz	2 MHz
Wavelet	500 Kbps	5 MHz	3 MHz
Butterworth LPF	500 Kbps	5 MHz	1 MHz

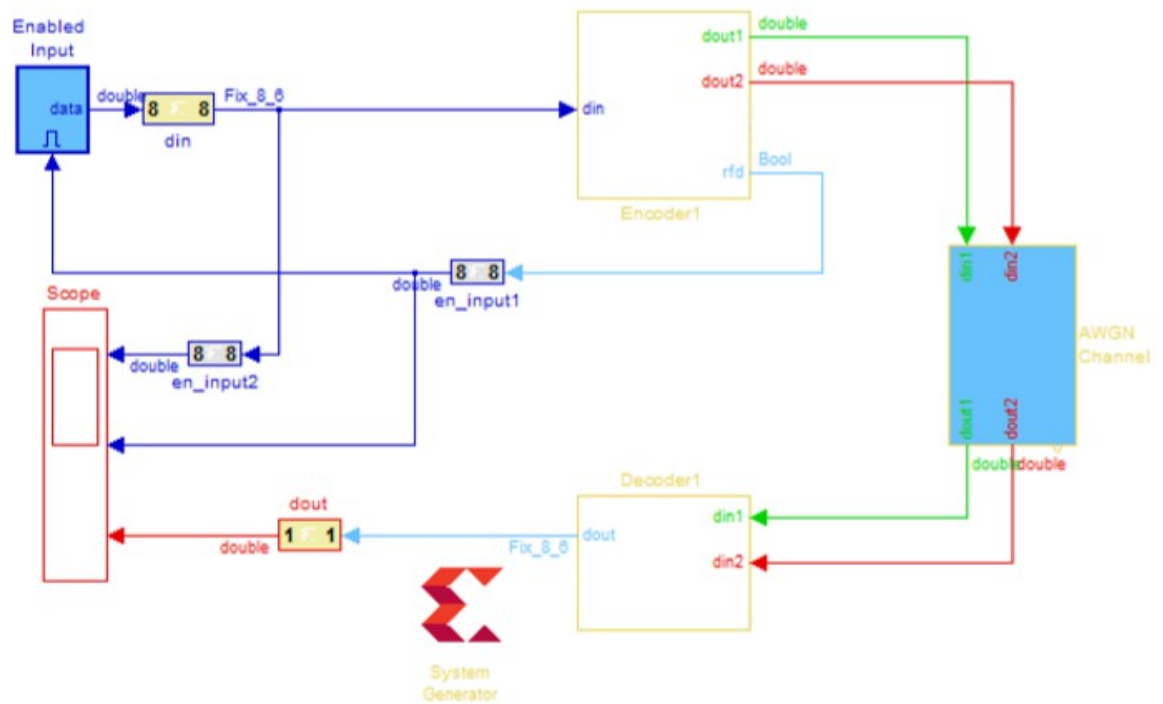


Рисунок. 2.8 – Модель системи з корекцією помилок

Джерело формує дискретизований і квантований за рівнями синусоїдальний сигнал.

Блок кодера Encoder1 складається з декількох функціональних блоків (рисунок 2.9). Тобто з кодера Ріда-Соломона (Outer Encoder), переміжника (Interleaver), згорткового кодера (Inner Encoder) і puncturing.

Блок затримки (Enabled Delay) необхідний для обліку часу виконання процесу кодування.

Блок Outer Encoder являє собою кодер укороченого коду РС (204,188,8), що формується в кодері повного коду Ріда - Соломона (255, 239, 8) шляхом додавання перед початком пакету групи з 51 байта, що складається з одних нулів. Одержаний пакет з 239 байт проходить перешкодостійке кодування, після закінчення якого, додані позиції в 51 байт відкидаються.

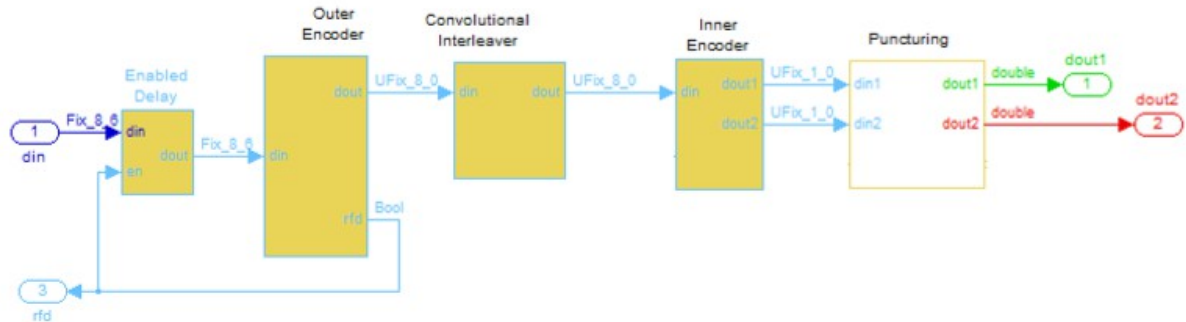


Рисунок 2.9 – Блок кодера Encoder1

Блок Inner Encoder являє собою згортковий кодер, наступної структури (рис.2.10).

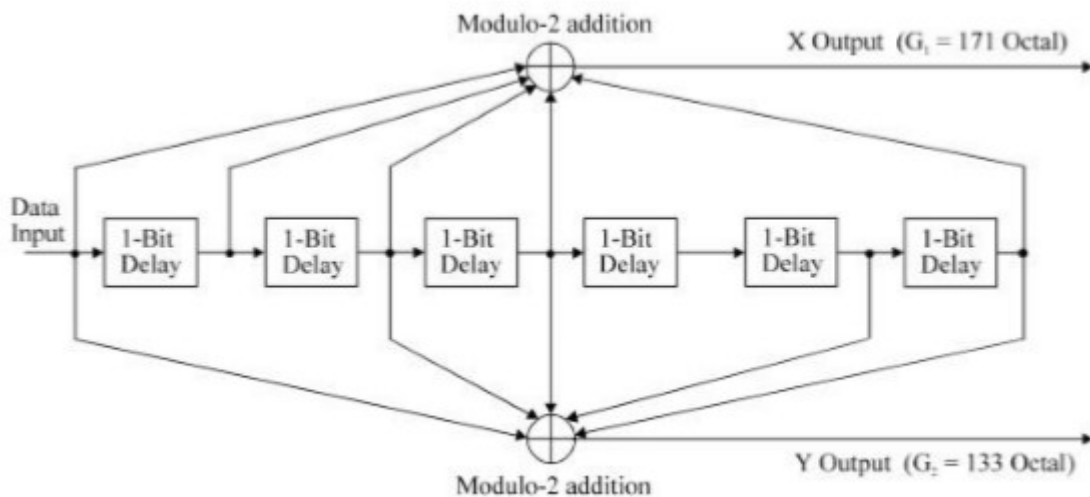


Рисунок 2.10 – Структура згорткового кодера (171,133)

Для захисту від пакетних помилок великої тривалості вводиться перемеження даних.

Глибина перемеження, тобто перестановка двох сусідніх байт пакету після кодування, становить 12 байт. Відповідно до цього параметр кодера-перемежника складається з 12 синхронно комутованих по входу і виходу паралельних гілок.

У кожному положенні комутатора в реєстри записується один байт інформації, і за один цикл комутації в перемежник буде введено 12 байт

інформації. Таким чином, пакет в 204 байта буде повністю оброблений за 204 байта / 12 гілок = 17 циклів комутації. Кожна гілка складається з регістрів зсуву різної довжини, які забезпечують затримку даних на величину: M (байт) $\times n$, де n - порядковий номер кожної гілки ($n = 0 \dots 11$). При цьому значення M визначається зі співвідношення: $M = \text{тривалість пакета в байтах (204 байт)} / \text{на число гілок перемешника (n = 12)} = 17 \text{ байт}$ і має бути цілим числом[16]. Таким чином, гілка з номером $n = 0$ затримку даних не вносить, а гілка з $n = 11$ вносить максимальну затримку: $17 \text{ байт} \times 11 = 187 \text{ байт}$.

Деперемешник будується аналогічним чином, але зі зворотним порядком включення регістрів затримки - гілка з номером $n = 0$ забезпечує максимальну затримку в 187 байт, а гілка з номером $n = 11$ затримки не вносить. Блок AWGN channel моделює канал з адитивним білим гауссовским шумом. Блок Decoder1 здійснює операцію декодування (рисунок 2.11).

Блок Depuncturing виконує операцію квантування за рівнями, які використовуються в декодері Вітербі (Inner Decoder) в якості м'яких рішень, а також здійснює процес девікалівання.

Блок Inner Decoder реалізує схему, що виконує декодування по алгоритму Вітербі. Блок Convolutional Deinterleaver виконує операцію деперемешення.

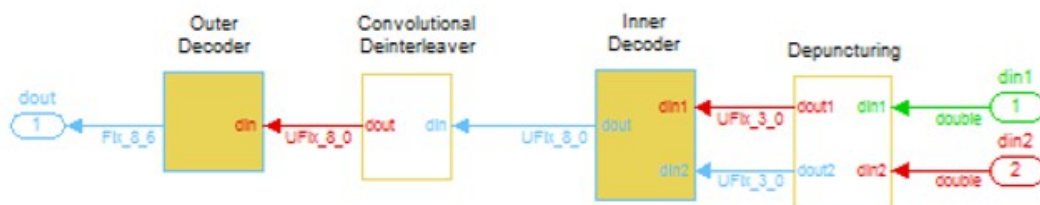


Рисунок 2.11 – Схема блоку Decoder1

На рисунку 2.12 представлені результати моделювання – діаграми каскадної схеми кодування.

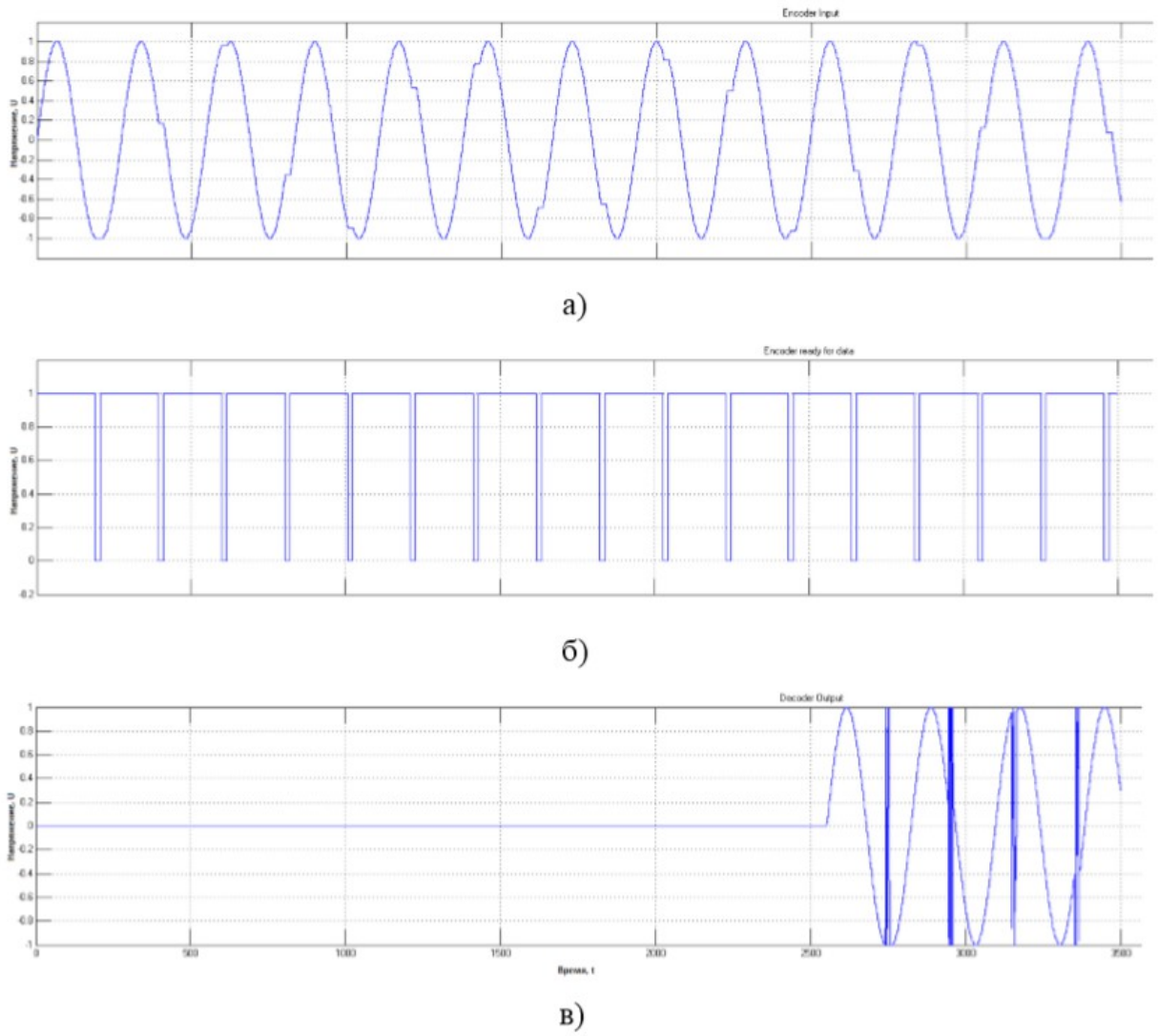


Рисунок 2.12 – Діаграми каскадної схеми кодування:

- а) сигнал перед кодуванням; б) сигнал про стан вхідного буфера;
в) сигнал після декодування

На рисунку 2.13 представлено аналогічні діаграми, що і на рисунку 2.12, але з моменту початку декодування.

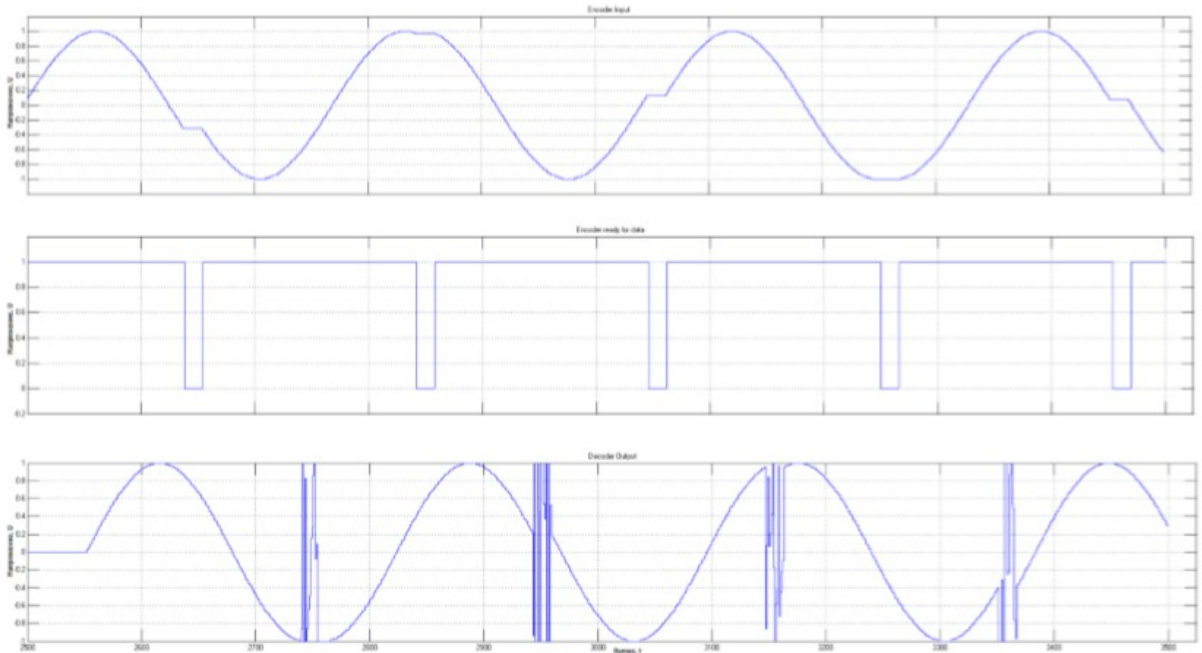
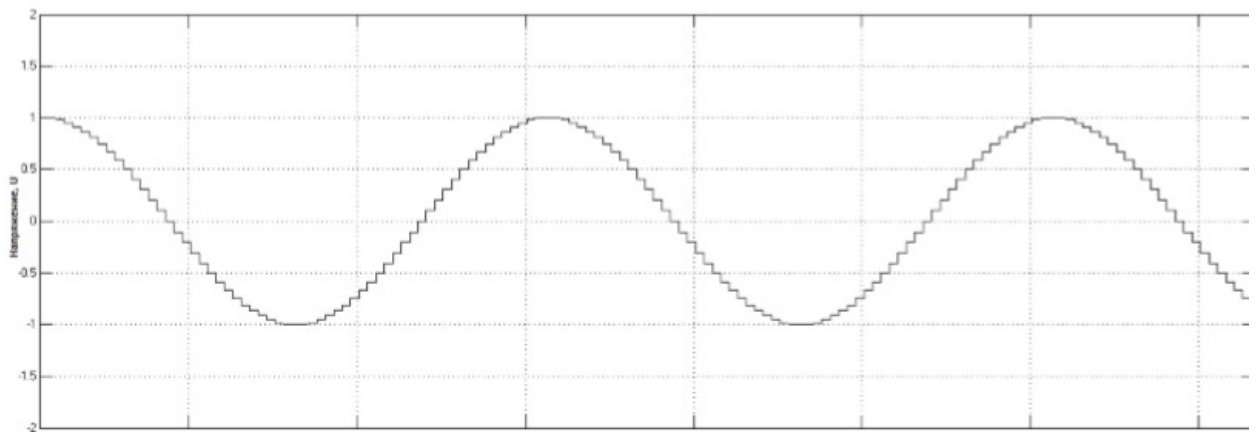


Рисунок 2.13 – Діаграми каскадної схеми кодування (з моменту декодування): а) сигнал перед кодуванням; б) сигнал про стан вхідного буфера; в) сигнал після декодування

На рисунку 2.14 представлений сигнал після процедури успішного декодування і в разі виникнення помилок.



а)

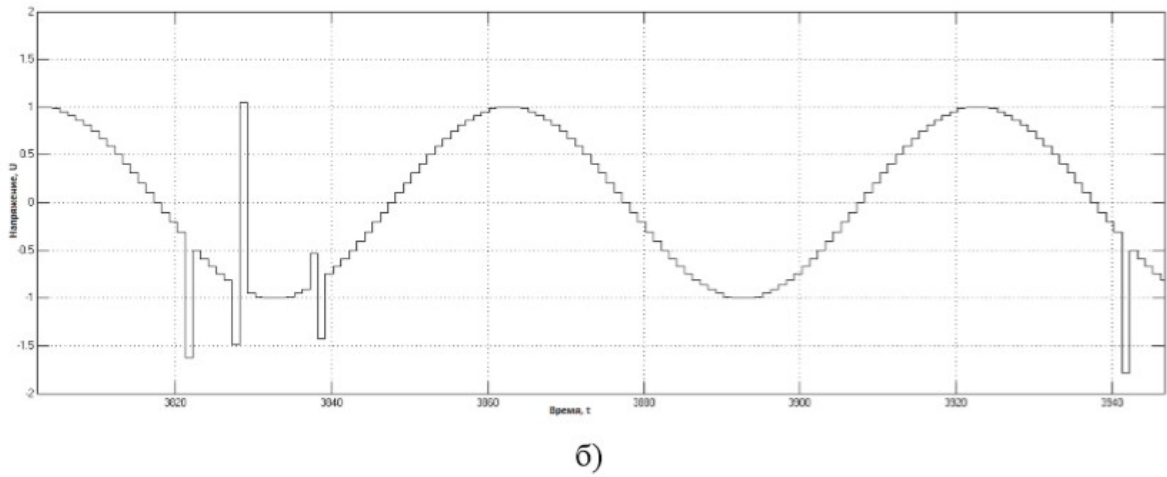


Рисунок 2.14 – Сигнал після декодування: а) безпомилкове декодування;
б) декодування в разі виникнення помилок

Далі, було здійснено моделювання каскадних схем кодування при різних видах модуляції і швидкості згортального коду.

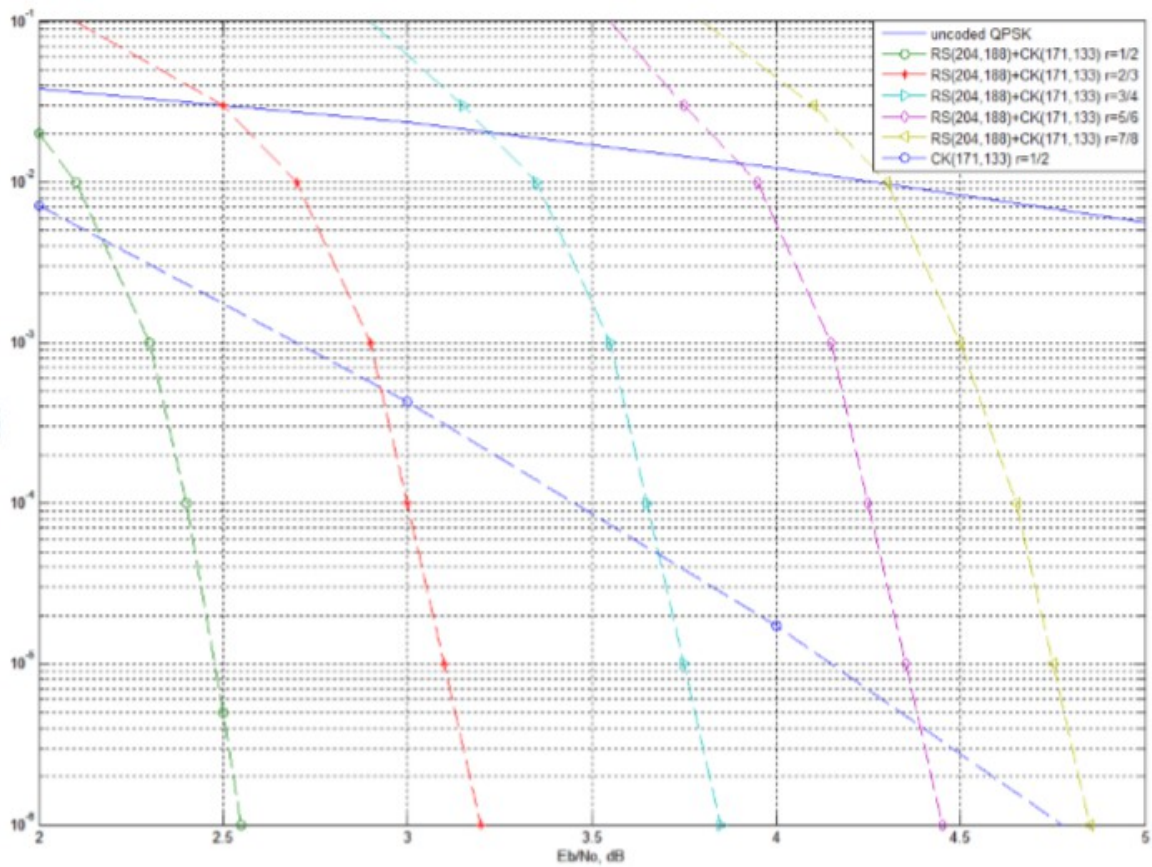


Рисунок 2.15 – Перешкодостійкість каскадної схеми кодування при різних швидкостях згортального коду при модуляції QPSK

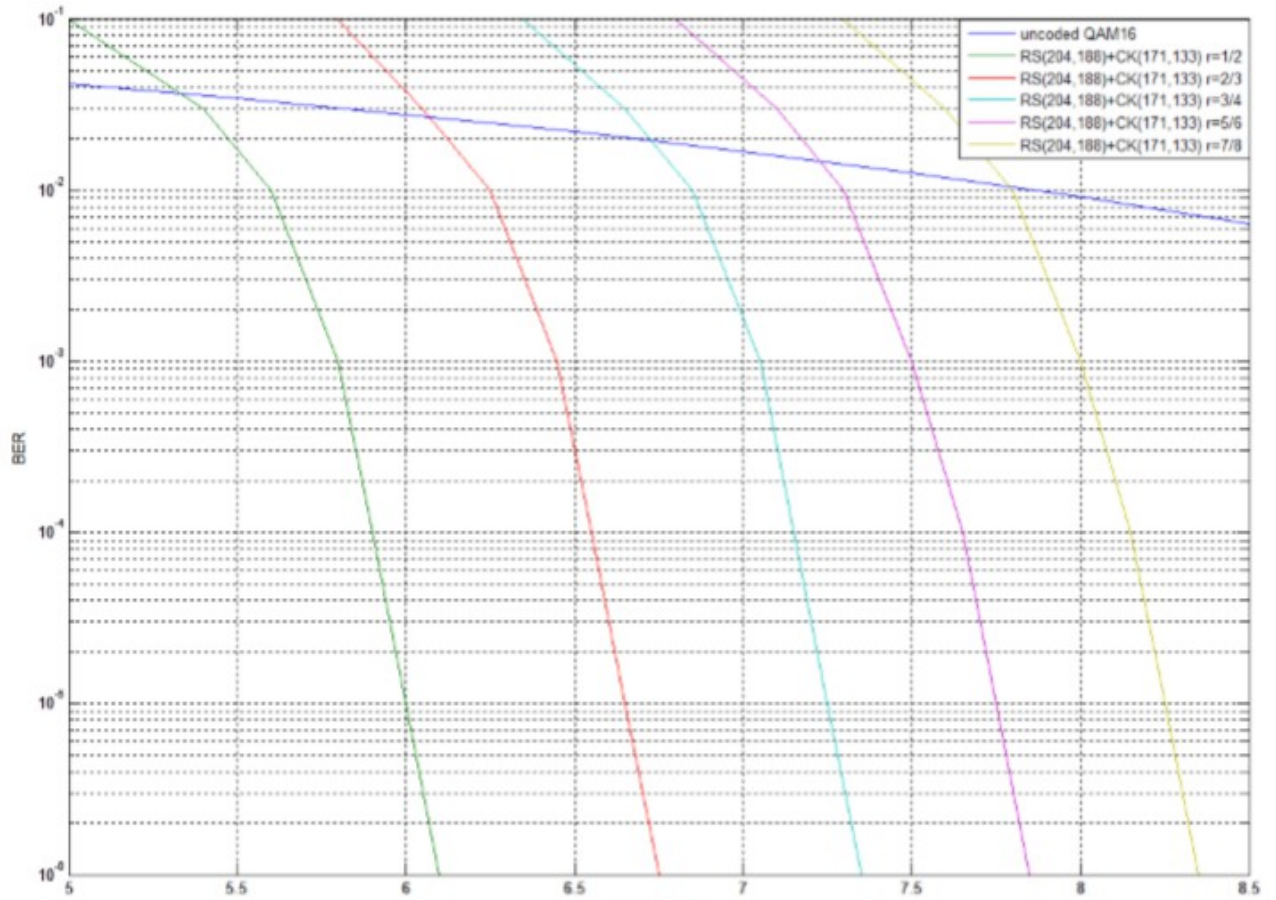


Рисунок 2.16 – Перешкодостійкість каскадної схеми кодування при різних швидкостях згортального коду при модуляції QAM16

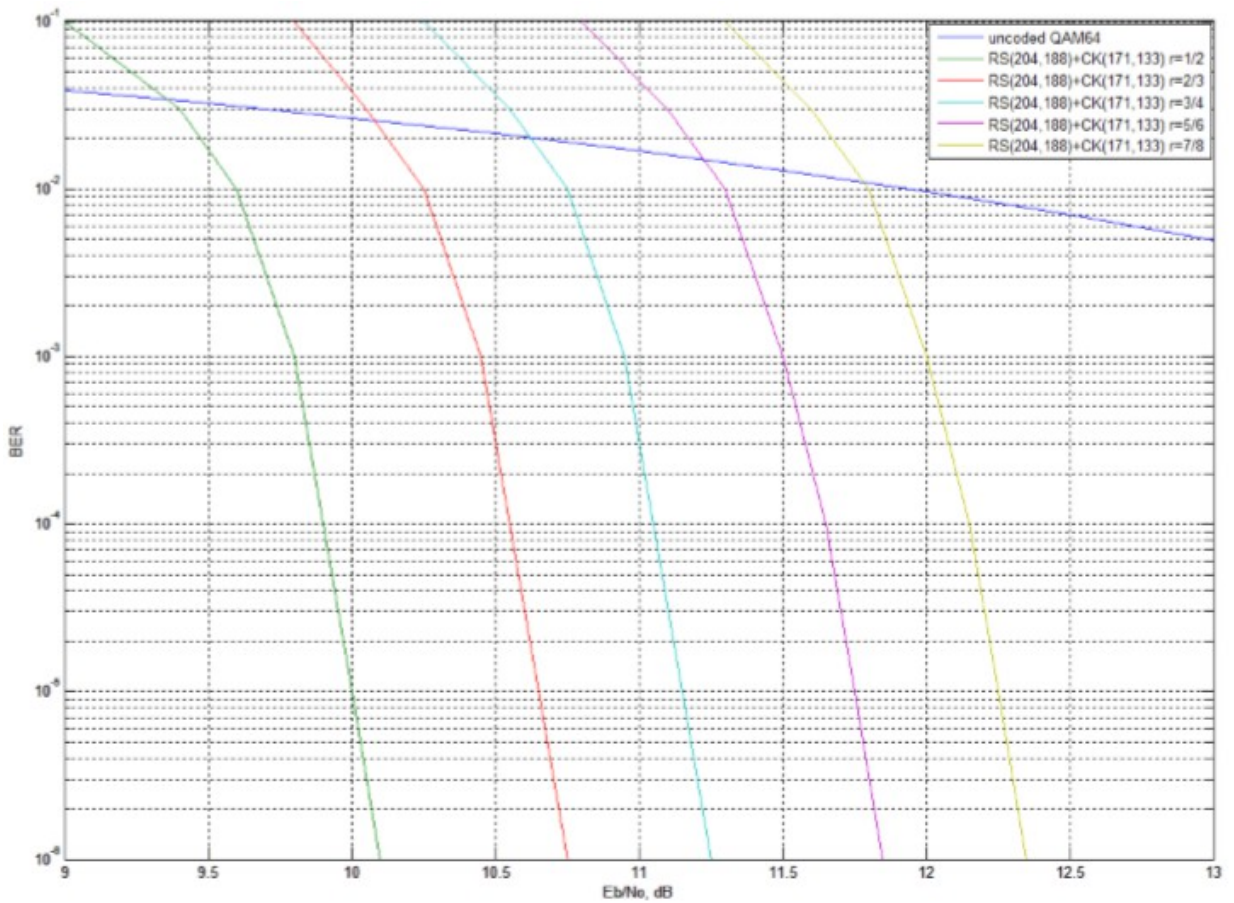


Рисунок 2.17 – Перешкодостійкість каскадної схеми кодування при різних швидкостях згортального коду при модуляції QAM64

Результат синтезу, тобто синтезована RTL – схема, наведено на рисунку 2.18.

ВИСНОВКИ

В роботі було проаналізовано способи кодування в системах стандарту DVB-T.

В ході роботи було виконано моделювання декодера Вітербі в середовищі Matlab. Були отримані залежності завадостійкості декодера при різній глибині декодування і розрядності м'яких рішень. Було визначено обсяг використовуваної пам'яті ПЛІС при різних глибинах декодування.

Також був розроблений проект декодера мовою VHDL, що відповідає стандарту DVB-T і володіє можливістю гнучкого налаштування його параметрів.

В результаті компіляції функції декодера Вітербі було визначено необхідний обсяг пам'яті ПЛІС для реалізації декодера в залежності від глибини декодування. Були обрані кілька ПЛІС, на яких реалізація декодера була б найбільш вигідна з економічних міркувань.

Моделювання не виявило відмінностей між кривими завадостійкості декодера Вітербі, отриманих при різних глибинах декодування (64 і 128), при довірчому інтервалі 0,8 дБ і довірчій ймовірності 0,95. При збільшенні глибини декодування, починаючи з величини більшої, ніж 8 довжин кодового обмеження, не вдається отримати енергетичного виграшу.

При виборі глибини декодування меншою, ніж 5 довжин кодового обмеження, було отримано енергетичний програш приблизно в 1дБ. При переході від 3-х бітних м'яких рішень до 4-х бітовим, не отримуємо енергетичного виграшу, з урахуванням довірчого інтервалу 0,8дБ і довірчій ймовірності 0,95.

Гнучкість декодера Вітербі була досягнута, завдяки використанню ПЛІС, за рахунок можливості зміни таких параметрів, як швидкості

згортальних кодів, розрядності м'яких рішень, глибини декодування, довжини блоків даних.

Проведено порівняння проектів декодерів з різною глибиною декодування (64 і 128). Були визначені програмовані логічні інтегральні схеми, на яких можливе виконання таких декодерів.

Розроблений модуль можна використовувати для реалізації системи цифрового наземного телебачення стандарту DVB-T, так і в інших декодерах такого типу.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Прокис, Д. Цифровая связь / Д. Прокис. – М. : Радио и связь, 2000. – 800 с.
2. Волков, Л. Н. Системы цифровой радиосвязи: базовые методы и характеристики [Текст] / Л. Н. Волков, М. С. Немировский, Ю. С. Шинаков. – М. : Эко-Трендз, 2005. – 392с.
3. Вернер, М. Основы кодирования [Текст] : пер. с нем. – К. : Зигангилова; М. : ТЕХНОСФЕРА, 2004. – 288с.
4. Short block length LDPC codes for TC synchronization and channel coding [Текст] – М. : Consultative Committee for Space Data Systems, 2015. – 39 p.
5. TM synchronization and channel coding: Recommended standard 131.0-B-2. [Текст] – М. : CCSDS, 2011. – 93 p.
6. Морелос-Сарагоса, Р. Искусство помехоустойчивого кодирования. Методы, алгоритмы, применение [Текст] / Р. Морелос-Сарагоса. – М. : Техносфера, 2005. – 320 с.
7. Шульгин, В. И. Основы теории передачи информации. Часть 1 Экономное кодирование [Текст] / В. И. Шульгин. – М. : ХАИ, 2003. - 103с.
8. Шульгин, В. И. Основы теории передачи информации. Часть 2 Помехоустойчивое кодирование [Текст] / В. И. Шульгин. – М. : ХАИ, 2003. - 88с.
9. Золотарев, В. В. Помехоустойчивое кодирование. Методы и алгоритмы. Справочник. [Текст] / В. В. Золотарев, Г. В. Овечкин. – М. : Горячая линия - Телеком, 2004. - 126с.
10. Березюк, А. Г. Кодирование информации (двоичные коды) [Текст] / А. Г. Березюк, Н. Т. Андрущенко, С. С. Моощицкий. – М. : Издательское объединение «Вища школа», 1978. – 252 с.

11. Файнштейн, А. Основы теории информации [Текст] / А. Файнштейн. – М. : Изд-во иностранной литературы, 1960. - 144с.
12. Галагер, Р. Д. Коды з низькою щільністю перевірок на парність [Текст]: Пер. з англ. – М. : Мир, 1966. – 145 с.
13. Горюнов, А. Г. Телеконтроль и управление [Текст] / А. Г. Горюнов, С. Н. Ливенцов, А. А. Лисенок. – М. : ТПУ, 2008. – 160 с.
14. Хэмминг, Р. В. Теория кодирования и теория информации [Текст] : пер. с англ. – М. : Радио и связь, 1983. - 176с.
15. Галлагер, Р. Теория информации и надежная связь [Текст] / Р. Галлагер. – М. : Советское радио, 1974. - 720с.
16. Radio frequency and modulation systems – part 1 earth stations and spacecraft: Recommended standard 401.0-B. [Текст] – М. : CCSDS, 2013. – 120с.