

ФАКУЛЬТЕТ АВТОМАТИКИ, ТЕЛЕМЕХАНІКИ ТА ЗВ'ЯЗКУ

Кафедра «Транспортний зв'язок»

ДОСЛІДЖЕННЯ ЗАВАДОСТІЙКОГО КОДУВАННЯ

МЕТОДИЧНІ ВКАЗІВКИ

до лабораторних робіт

з дисципліни

«ТЕОРІЯ ЕЛЕКТРИЧНОГО ЗВ'ЯЗКУ»

Частина 2

Харків 2009

Методичні вказівки розглянуто та рекомендовано до друку на засіданні кафедри "Транспортний зв'язок" 11 грудня 2008 р., протокол № 5.

Рекомендуються для студентів факультету АТЗ спеціальності «Телекомунікаційні системи та мережі» та спеціалізації «Автоматизовані системи технологічного зв'язку на залізничному транспорті» всіх форм навчання.

Укладачі:

доц. О.П. Батаєв,

асп. Д.О. Бойко

Рецензент

доц. С.В. Кошевий

ДОСЛІДЖЕННЯ ЗАВАДОСТІЙКОГО КОДУВАННЯ

МЕТОДИЧНІ ВКАЗІВКИ

до лабораторних робіт
з дисципліни

«ТЕОРІЯ ЕЛЕКТРИЧНОГО ЗВ'ЯЗКУ»

Частина 2

Відповідальний за випуск Батаєв О.П.

Редактор Решетилова В.В.

Підписано до друку 06.02.09 р.

Формат паперу 60x84 1/16 . Папір писальний.

Умовн.-друк.арк. 1,75. Обл.-вид.арк. 2,0.

Замовлення № Тираж 150. Ціна

Видавництво УкрДАЗТу, свідоцтво ДК № 2874 від. 12.06.2007 р.

Друкарня УкрДАЗТу,
61050, Харків - 50, пл. Фейербаха, 7

ВСТУП

Останнім часом в різноманітних галузях народного господарства України, в тому числі і на залізничному транспорті впроваджуються сучасні телекомунікаційні технології для підвищення пропускної спроможності систем передачі інформації, а також для підвищення завадостійкості.

У свою чергу для підвищення завадостійкості систем передачі знаходять широке застосування різноманітні способи завадостійкого кодування. Тема «Завадостійке кодування» є однією з найважливіших у дисципліні «Теорія електричного зв'язку». При вивченні її студенти різних форм навчання повинні наочно досліджувати послідовності перетворень над довільними вихідними інформаційними послідовностями в кодові комбінації завадостійкого коду при кодуванні, а також зворотних перетворень при декодуванні.

Всі роботи передбачають при виконанні досліджень використання програмного середовища Matlab (переважно бібліотеки Simulink), в якому виконані структурні схеми, що пояснюють послідовність різноманітних перетворень.

У цих методичних вказівках наведені рекомендації щодо виконання лабораторних робіт за темами: «Дослідження циклічних кодів» і «Дослідження згорткових кодів», метою яких є закріплення теоретичних знань, отриманих з лекційного курсу та відповідної рекомендованої літератури.

1 Лабораторна робота 3

ДОСЛІДЖЕННЯ ЦИКЛІЧНИХ КОДІВ

1.1 Мета роботи

1.1.1 Засвоїти алгоритми процесів кодування та декодування циклічних кодів.

1.1.2 Розрахунковим і експериментальним шляхом дослідити процеси кодування та декодування циклічних кодів.

1.1.3 Навчитись та вміти синтезувати функціональні схеми кодеків.

1.2 Програма роботи

1.2.1 Аналітичне дослідження процесів кодування та декодування циклічних кодів.

1.2.2 Експериментальне дослідження процесів кодування та декодування циклічних кодів.

1.3. Підготовка до роботи

1.3.1 За рекомендованою літературою [1, 2, 3, 5] та конспектом лекцій у позааудиторний час засвоїти:

- мету, програму і вказівки щодо виконання роботи;
- теоретичні положення за темою роботи: поняття та алгоритми побудови кодових комбінацій циклічних кодів, виправлення помилок за їх допомогою, принципи синтезу кодеків циклічних кодів.

1.3.2 Виконати аналітичне дослідження процесів кодування та декодування циклічних кодів, в результаті чого побудувати таблиці зсуву, синтезувати кодер та декодер, використовуючи теоретичні відомості даних методичних вказівок (п. 1.4). Результати оформити у вигляді виконання домашнього завдання, виданого викладачем.

1.3.3 Підготувати бланк звіту з лабораторної роботи.

1.3.4 Підготувати відповіді на контрольні запитання.

1.4 Короткі відомості з теорії

Поняття циклічних кодів зводиться до наступного: підпростір V наборів довжини n називається циклічним підпростором або циклічним кодом, якщо для будь-якого вектора $v = (a_{n-1}, a_{n-2}, \dots, a_0)$ з підпростору V вектор $v' = (a_0, a_{n-1}, a_{n-2}, \dots, a_1)$, що одержується в результаті циклічного зсуву компонент вектора v на один розряд вправо і перестановки останнього елемента a_0 на місце першого a_{n-1} , також належить підпростору V .

Кожному набору $(a_{n-1}, a_{n-2}, \dots, a_0)$ довжини n відповідає багаточлен $f(x) = a_{n-1}x^{n-1} + a_{n-2}x^{n-2} + \dots + a_0$ степені, меншої n . Комбінації циклічних кодів при поданні їх багаточленами можуть бути задані породжувальним (твірним) багаточленом $g(x)$ степені $r = n - k$, на який без залишку ділиться багаточлен виду $f(x) = x^n + 1$.

Циклічний код може бути заданий перевірочним багаточленом

$$h(x) = \frac{(x^n + 1)}{g(x)} = h_k x^k + \dots + h_1 x + 1. \quad (1.1)$$

При необхідності за багаточленами $g(x)$ і $h(x)$ можуть бути складені породжувальна і перевірна матриці циклічного коду

$$G = \begin{Bmatrix} g(x) \\ xg(x) \\ \vdots \\ x^{k-1}g(x) \end{Bmatrix}; \quad H = \begin{Bmatrix} h(x) \\ xh(x) \\ \vdots \\ x^{r-1}h(x) \end{Bmatrix}. \quad (1.2)$$

Рядки матриць лінійно незалежні й отримані циклічним зсувом багаточленів $g(x)$ і $h(x)$. Як і для лінійних кодів, $yH^T = 0$. Варто тільки враховувати, що скалярне множення векторів не збігається із множенням багаточленів, що відображають ці вектори. Тому при матричному поданні циклічних кодів для виконання умови $yH^T = 0$ необхідно символи перевірочної (або породжувальної) матриці записувати у зворотному порядку.

Коригувальні властивості циклічних кодів залежать від степені й числа не рівних нулю коефіцієнтів породжувального багаточлена

$$g(x) = \alpha_r x^r + \alpha_{r-1} x^{r-1} + \dots + \alpha_1 x^1 + \alpha_0 x^0, \quad (1.3)$$

де α_i — двійкові символи кодової комбінації. Якщо $\alpha_0 = 1$, а інші коефіцієнти дорівнюють нулю, то $g(x) = \alpha_0 x^0 = 1$, тобто породжувальний багаточлен має нульову степінь і відповідає простому коду. Якщо $\alpha_0 = \alpha_1 = 1$, а інші коефіцієнти дорівнюють нулю, то $g(x) = x + 1$. В додатку А (таблиця А.1) наведені багаточлени, що не зводяться, до 5 ступеня включно.

Комбінації циклічних кодів будують за алгоритмом, що визначається породжувальним багаточленом. Помножимо кожен послідовність інформаційних символів $A(x)$ довжиною k символів на одночлен x^{n-k} , а потім добуток $x^{n-k} A(x)$ розділимо на породжувальний багаточлен $g(x)$ степені $n - k$. В результаті множення $A(x)$ на x^{n-k} степінь кожного доданка x , що входить в інформаційний багаточлен $A(x)$, підвищиться на $n - k$. При діленні добутку $x^{n-k} A(x)$ на $g(x)$ одержимо частку й залишок у вигляді багаточлена $R(x)$ степені не вище $n - k - 1$.

Таким чином,

$$\frac{x^{n-k} A(x)}{g(x)} = Q(x) + \frac{R(x)}{g(x)}. \quad (1.4)$$

Помножуючи обидві частини рівності (1.4) на $g(x)$, після переносу вліво добутку $Q(x) g(x)$ одержимо

$$F(x) = Q(x)g(x) = x^{n-k} A(x) + R(x). \quad (1.5)$$

Знак мінус при переносі замінений на плюс, тому що операції додавання й віднімання по модулю 2 збігаються. Степінь частки $Q(x)$ така ж, як і інформаційного багаточлена вихідного коду, тобто $Q(x)$ є також багаточленом вихідного коду.

З рівності 1.5 видно, що комбінації циклічного коду довжиною $n - k + r$ можуть бути утворені двома способами:

1) множенням кожної k -символьної комбінації вихідного коду, вираженої багаточленом $A(x)$, на породжувальний багаточлен $g(x)$ степені $n - k$;

2) множенням кожної k -символьної комбінації, вираженої багаточленом $A(x)$, на одночлен x^{n-k} і додаванням до добутку $x^{n-k} A(x)$ залишку $R(x)$ від ділення $x^{n-k} A(x)$ на породжувальний багаточлен степені $n - k$.

Обидва способи побудови комбінацій циклічних кодів еквівалентні. Однак перший спосіб приводить до утворення нероздільного циклічного коду, тобто в кодових комбінаціях за інформаційними й перевірочними символами немає закріплених позицій. Циклічний код, утворений за другим способом, роздільний. Багаточлен $x^{n-k} A(x)$ відповідає інформаційним символам, а багаточлен залишку $R(x)$ — перевірочним. Множина дозволених комбінацій циклічного коду в обох випадках кодування дорівнює числу комбінацій вихідного коду $N_p = 2^k$, а число заборонених $N_c = 2^n - 2^k$ тим більше, чим вище ступінь багаточлена.

Всі дозвалені комбінації циклічного коду без залишку діляться на утворюючий багаточлен. При діленні заборонених комбінацій з'являється залишок, який являється синдромом. Цю властивість комбінацій циклічного коду використовують для виявлення й виправлення помилок.

Комбінації роздільного циклічного коду можуть бути отримані множенням багаточлена вихідного коду $A(x)$ на одночлен x^{n-k} і додаванням до цього добутку залишку

$$R(x) = \frac{x^{n-k} A(x)}{g(x)}$$
. Дію кодера розглянемо за структурною схемою,

наведеною в додатку А (рисунок А.1). Її основою є регістр, що виконує ділення багаточлена вихідного коду на породжувальний багаточлен $g(x) = x^3 + x + 1$. Так як заповнення кодового фільтра (схеми ділення на основі регістра зсуву зі зворотними зв'язками) відбувається за $n - k$ тактів, а весь процес ділення триває n тактів, то процес формування залишку від ділення здійснюється за $n - (n - k) = k$ тактів. За k тактів повинні бути відправлені в лінію

зв'язку інформаційні символи, які необхідно затримати на $n - k$ тактів для того, щоб не було розриву між інформаційними й перевірочними символами. Тому дана схема містить елемент затримки на $n - k = 3$ такти.

Значне спрощення декодерів для виправлення помилок можливо при декодуванні комбінацій, сформованих діленням комбінації вихідного коду на породжувальний багаточлен, тобто у випадку роздільного циклічного коду. Розглянемо саме цей випадок.

У додатку А (рисунок А.2) наведена структурна схема декодера для роздільного циклічного коду $(7, 4)$ з породжувальним багаточленом $g(x) = x^3 + x + 1$. Прийнята кодова комбінація одночасно надходить у реєстри зсуву 1 і 2. Регістр зсуву 1 виконує роль накопичувача прийнятої комбінації, а регістр зсуву 2 — операцію ділення прийнятої комбінації на породжувальний багаточлен. При відсутності помилок у прийнятій комбінації залишок в елементах реєстра зсуву 2 дорівнює нулю й далі символи комбінації зчитуються з реєстра зсуву 1 через суматор М2-3.

При помилках у прийнятій кодовій комбінації залишок в елементах реєстра зсуву 2 не дорівнює нулю. Для виправлення помилки вхід пристрою відключається від каналу, і з реєстра зсуву 1 через суматор М2-3 символи кодової комбінації подаються на вихід. Одночасно із цим у реєстрі зсуву 2 відбувається зчитування символів отриманого залишку, які надходять на дешифратор. Під час проходження спотвореної послідовності через суматор М2-3 під дією тактових імпульсів 2 дешифратор виробляє сигнал (виправляючий вектор), що виправляє певний спотворений символ інформаційної послідовності в залежності від отриманого синдрому.

Основу кодерів і декодерів роздільних циклічних кодів становлять реєстри зсуву, що здійснюють операцію ділення кодового багаточлена на породжувальний багаточлен. Тому у двосторонніх системах, що працюють по черзі для схемного спрощення, функції кодування й декодування виконує один кодоперетворювач, що одержав назву кодек.

Припустимо, що на вхід кодера приходить інформаційна послідовність, що відповідає багаточлену $A(x) = x^3 + x - 1010$.

Спочатку на вхід елемента І1 надходить тактовий імпульс 1 (елемент відкритий), а на вхід елемента І2 тактовий імпульс 2 не надходить (елемент закритий). Протягом перших семи тактів інформаційна послідовність через елемент АБО надходить у канал зв'язку й паралельно через суматор по модулю 2 у регістр зсуву. В елементах регістра від ділення $x^3 A(x) = x^6 + x^4$ на $g(x) = x^3 + x + 1$ утвориться залишок $R(x) = x + 1$ — 011. Потім елемент І1 закривається, а І2 відкривається й через елемент АБО починається зчитування комбінації залишку 011 у канал зв'язку (див. таблицю А.2).

При прийомі комбінацій елемент І1 протягом чотирьох тактів відкритий й інформаційні символи надходять одночасно в регістри зсуву 1 і 2. Потім елемент І1 закривається й у наступні три такти перевірочні символи надходять тільки в регістр зсуву 2. Після сьомого такту на елементи І2 -4 подається тактовий імпульс, вони відкриваються й значення символів залишку зчитуються в елемент АБО. У випадку рівності нулю символів залишку, тобто 000, починається зчитування інформаційних символів з регістра 1. Процес ділення багаточлена циклічного коду, прийнятого без помилок, $F(x) = x^6 + x^4 + x + 1$ — 1010011 на багаточлен $g(x) = x^3 + x + 1$ показаний у таблиці А.2.

Якщо в комбінації циклічного коду, наприклад, у другому символі, сталася одиночна помилка, то при діленні багаточлена $F(x) = x^6 + x^5 + x^4 + x + 1$, 1 — 1110011 на породжувальний багаточлен $g(x) = x^3 + x + 1$ в елементах регістра зсуву 2 буде залишок $R(x) = x^2 + x + 1$ — 111, що не дорівнює нулю.

На виході елемента АБО з'являється сигнал «Помилка» і значення символів в елементах регістрів зсуву 1 і 2 стираються. Ділення багаточлена з помилкою $F(x) = x^6 + x^5 + x^4 + x + 1$ на $g(x) = x^3 + x + 1$ відбувається за сім тактів (див. таблицю А.2).

У декодерах циклічних кодів використаний більш простий у порівнянні з лінійними спосіб виправлення помилок, тому що синдромами помилок цих кодів є залишки від ділення багаточленів помилок $E(x)$ на породжувальний багаточлен $g(x)$ (у зв'язку з цим варто зазначити, що лише багаточлен, що не зводиться, дає найбільшу кількість залишків при діленні на нього багаточленів помилок).

Для циклічного коду (7, 4) з породжувальним багаточленом $g(x) = x^3 + x + 1$ таблиця відповідності комбінацій однократних E_i помилок і залишків (синдромів) S_i має такий вигляд:

$$\begin{array}{ll}
 E_1 = x^6 \rightarrow 1000000, & S_1 = x^2 + 1 \rightarrow 101; \\
 E_2 = x^5 \rightarrow 0100000, & S_2 = x^2 + x + 1 \rightarrow 111; \\
 E_3 = x^4 \rightarrow 0010000, & S_3 = x^2 + x \rightarrow 110; \\
 E_4 = x^3 \rightarrow 0001000, & S_4 = x + 1 \rightarrow 011; \\
 E_5 = x^2 \rightarrow 0000100, & S_5 = x^2 \rightarrow 100; \\
 E_6 = x^1 \rightarrow 0000010, & S_6 = x^1 \rightarrow 010; \\
 E_7 = x^0 \rightarrow 0000001, & S_7 = x^0 \rightarrow 001;
 \end{array} \tag{1.6}$$

Алгоритм виправлення помилки в цьому випадку полягає в наступному. Якщо залишок дорівнює S_i то i -й розряд прийнятої комбінації підсумовується з одночленом E_i , чим досягається виправлення однократної помилки, а якщо залишок S_i — 000, то комбінація вважається прийнятою без помилок.

1.5 Вказівки щодо виконання роботи

1.5.1 Опис лабораторного макета

Так само, як в попередній лабораторній роботі, лабораторний макет на тему «Дослідження циклічних кодів» виконаний за допомогою пакета програм Simulink програмного середовища Matlab. Даний макет реалізує процес кодування і декодування роздільного циклічного коду (7, 4), приклад якого детально проілюстрований в пункті 1.4. Відмінність полягає в такому:

1) генератор інформаційної послідовності дає змогу ввести десяткове число I від 1 до 15. Цей спектр значень обумовлений тим, що максимальна кількість інформаційних комбінацій циклічного коду (7, 4) складає, як відомо, $2^k = 2^4 = 16$, однак через те, що нульова комбінація не використовується, то в макеті використовується лише 15 комбінацій;

2) генератор однократної помилки дозволяє обрати помилку у будь-якому розряді від 1 до 7 кодової комбінації шляхом введення десяткового числа K від 1 до 7 (цей спектр значень обумовлений тим, що довжина кодової комбінації циклічного (7, 4)- коду складає 7 елементів).

1.5.2 Вказівки щодо виконання лабораторної роботи

Як і в попередній лабораторній роботі, передбачається, що студенти перед виконанням даної лабораторної роботи отримують домашнє завдання у вигляді чотирирозрядної інформаційної послідовності, яку вони повинні закодувати і декодувати циклічним кодом з однократною помилкою в заданому розряді при передачі по каналу зв'язку, а також без помилки. Виконавши домашнє завдання у вигляді таблиць зсуву, процедура побудови яких докладно наведена у п. 1.4 та відповідній рекомендованій літературі, а також структурних схем кодерів та декодерів циклічного коду, студенти мають можливість закріпити теоретичні знання з даної теми, а також перевірити правильність виконання домашнього завдання.

Так, як і в лабораторній роботі «Дослідження лінійних кодів Хеммінга», розроблений лабораторний макет передбачає виконання однойменної лабораторної роботи в чотирьох режимах. Таким чином, розглянемо порядок виконання роботи в кожному з зазначених вище режимів.

Порядок виконання роботи в режимі 1 і 2:

- 1) відкрити файл `Cyclic_code`;
- 2) встановити КЛЮЧ 1 в положення 1;
- 3) натиснути два рази на блок ЗМІННИЙ ГЕНЕРАТОР ІНФОРМАЦІЙНОЇ ПОСЛІДОВНОСТІ і відповідно до певного варіанта ввести значення I від 1 до 15 в десятковій формі, в результаті чого блок автоматично перетворює це число в двійкову форму і генерує сигнал у вигляді послідовності логічних одиниць (1) і нуля (0);
- 4) встановити КЛЮЧ 3 в положення 2;
- 5) натиснути два рази на блок ВЕКТОР ОДНОКРАТНОЇ ПОМИЛКИ і відповідно до певного варіанта ввести значення $K = 0$ для режиму 1 та значення K від 1 до 7 в десятковій формі

для режиму 2, в результаті чого блок автоматично генерує сигнал завади у вигляді послідовності логічних одиниці (1) і нуля (0), який в залежності від введеного числа спотворює кодову послідовність у відповідному розряді;

6) послідовно натиснути: два рази на блок СКИДАННЯ РЕЗУЛЬТАТІВ, один раз на значок Start Simulation, два рази на блок ГЕНЕРАТОР ПОМИЛКИ, один раз на значок Start Simulation, два рази на блок ДЕШИФРАЦІЯ, один раз на значок Start Simulation, в результаті чого виконується запуск макета;

7) натиснути два рази на блок ОСЦИЛОГРАФ 1, де на новому вікні можна спостерігати сформовану інформаційну послідовність згідно з заданим варіантом (вона знаходиться в межах часу від 0 до 4 секунд, тобто перші 4 такти);

8) натиснути два рази на блок ОСЦИЛОГРАФ 2, де можна спостерігати кодову послідовність циклічного коду, яка затримана на 3 такти (вона знаходиться в межах часу від 3 до 10 секунд);

9) натиснути два рази на блок КОДЕР ЦИКЛІЧНОГО КОДУ і два рази на блок ОСЦИЛОГРАФ 5, де можна спостерігати процес ділення інформаційної послідовності на породжувальний багаточлен $g(x) = x^3 + x^2 + 1$ у вигляді станів ТРИГЕРІВ 1-3 (саме тут студент має можливість перевірити правильність складання таблиць зсуву). Слід зазначити, що значення станів ТРИГЕРІВ зсунуті на 1 такт (1 секунду на часовій осі) через використання елементів затримки (тригерів). Таким чином, процедуру ділення слід спостерігати з 1 по 8 секунди (всього 7 тактів);

10) натиснути два рази на блок ОСЦИЛОГРАФ 6, де можна спостерігати процес зчитування залишку від ділення інформаційної послідовності на породжувальний багаточлен $g(x) = x^3 + x^2 + 1$ у вигляді значень 5-7 розрядів кодової послідовності (він знаходиться в межах часу від 7 до 10 секунд);

11) натиснути два рази на блок ОСЦИЛОГРАФ 7, де можна спостерігати процес формування кодової послідовності;

12) закрити вікно блока КОДЕР ЦИКЛІЧНОГО КОДУ і натиснути 2 рази на блок ОСЦИЛОГРАФ 3, де можна спостерігати процес спотворення кодової послідовності завадою у вигляді однократної помилки;

13) натиснути два рази на блок ДЕКОДЕР ЦИКЛІЧНОГО КОДУ і 2 рази на блок ОСЦИЛОГРАФ 8, де можна спостерігати процес ділення кодової послідовності на породжувальний багаточлен $g(x) = x^3 + x^2 + 1$ у вигляді станів ТРИГЕРІВ 1-3 (тут студент також має можливість перевірити правильність складання таблиць зсуву). Слід зазначити, що значення станів ТРИГЕРІВ також зсунуті на 1 такт (1 секунду на часовій осі) через використання елементів затримки (тригерів). Таким чином, спостерігати процедуру ділення слід з 4 по 11 секунди (всього 7 тактів);

14) натиснути два рази на блок ОСЦИЛОГРАФ 9, де можна спостерігати процес зчитування залишку (синдрому) від ділення кодової послідовності на породжувальний багаточлен $g(x) = x^3 + x^2 + 1$ (він знаходиться в межах часу від 10 до 13 секунд);

15) натиснути два рази на блок ДЕШИФРАТОР і два рази на блоки ОСЦИЛОГРАФ 11-12, де можна спостерігати процес порівняння отриманого синдрому з можливими СИНДРОМАМИ 1-4, натиснути два рази на блок ОСЦИЛОГРАФ 13 і побачити, який саме виправляючий вектор генерується в результаті порівняння синдромів;

16) послідовно закрити вікна блоків ДЕШИФРАТОР і ДЕКОДЕР ЦИКЛІЧНОГО КОДУ і натиснути два рази на блок ОСЦИЛОГРАФ 4, де можна спостерігати ефект від виправлення циклічним кодом, а також без нього.

Порядок виконання роботи в режимі 3:

- 1) виконати пункти 1.1-1.3;
- 2) встановити КЛЮЧ 3 в положення 1, а КЛЮЧ 2 в положення 2;
- 3) виконати пункти 1.6-1.16.

Порядок виконання роботи в режимі 4:

- 1) виконати пункти 1.1-1.3;
- 2) встановити КЛЮЧ 3 в положення 1, а КЛЮЧ 2 в положення 1;
- 3) виконати пункти 1.6-1.16;
- 4) перейти до оформлення звіту з даної лабораторної роботи.

Передбачається, що студент в ході виконання роботи повинен фіксувати отримані результати в звіті, одночасно порівнюючи їх з власними результатами виконання домашнього завдання.

1.6 Зміст звіту

1.6.1 Назва і мета роботи.

1.6.2 Схема лабораторного макета.

1.6.3 Хід та результати виконання домашнього завдання у вигляді алгебраїчних виразів циклічних кодів, а також схем кодерів та декодерів.

1.6.4 Результати виконання практичного дослідження циклічних кодів в чотирьох режимах за допомогою лабораторного макета в пакеті Simulink.

1.6.5 Висновки з роботи.

Контрольні питання

- 1 Який порядок кодування циклічних кодів?
- 2 В чому полягає процедура декодування циклічних кодів?
- 3 Яким чином виявляються помилки в циклічних кодах?
- 4 Яким чином виправляються помилки в циклічних кодах?
- 5 Поясніть вибір породжувального полінома в циклічних кодах?
- 6 Як будується породжувальна матриця циклічного коду?
- 7 Як будується перевірна матриця циклічного коду?
- 8 Як за допомогою породжувального полінома задати циклічний код?
- 9 Від чого залежить здатність циклічного коду виправляти помилки?
- 10 Що складає основу кодерів та декодерів роздільних циклічних кодів?
- 11 Яким чином утворюється синдром помилки при декодуванні циклічних кодів?

2 Лабораторна робота ДОСЛІДЖЕННЯ ЗГОРТКОВИХ КОДІВ

2.1 Мета роботи

2.1.1 Засвоїти алгоритми процесів кодування та декодування згорткових кодів.

2.1.2 На основі цих алгоритмів синтезувати структурні схеми кодера

2.1.3 Розрахунковим і експериментальним шляхом дослідити процеси кодування та декодування згорткових кодів.

2.2 Програма роботи

2.2.1 Аналітичне дослідження процесів кодування та декодування згорткових кодів.

2.2.2 Експериментальне дослідження процесів кодування та декодування циклічних кодів.

2.3 Підготовка до роботи

2.3.1 За рекомендованою літературою [2, 5-9] та конспектом лекцій у позааудиторний час засвоїти:

- мету, програму і вказівки щодо виконання роботи;
- теоретичні положення за темою роботи: поняття та алгоритми побудови кодових комбінацій згорткових кодів, виправлення помилок за їх допомогою, принципи синтезу кодеків згорткових кодів.

2.3.2 Виконати аналітичне дослідження процесів кодування згорткових кодів, в результаті чого побудувати таблиці зсуву та синтезувати кодер, використовуючи теоретичні відомості даних методичних вказівок (п. 2.4). Результати оформити у вигляді виконання домашнього завдання, виданого викладачем.

2.3.3 Підготувати бланк звіту з лабораторної роботи.

2.3.4 Підготувати відповіді на контрольні запитання.

2.4 Короткі відомості з теорії

Згорткове кодування зручніше за все описувати, характеризуючи дію відповідного кодувального пристрою. Згортковий кодер являє собою пристрій, що сприймає за кожний такт роботи в загальному випадку k вхідних інформаційних символів, і видає на вихід за той же такт n вихідних символів, що підлягають передачі по каналу зв'язку.

Відношення $R = k/n$ називають відносною швидкістю коду. Вихідні символи, що створюються кодером на даному такті, залежать від m інформаційних символів, що надійшли на цьому й попередньому тактах. Таким чином, вихідні символи згорткового кодера однозначно визначаються його вхідним сигналом і станом, що залежить від $m-k$ попередніх інформаційних символів.

Основними елементами згорткового кодера є: реєстр зсуву, суматори по модулю 2 і комутатор (Км).

Комутатор здійснює послідовне зчитування символів, що надходять на його входи (контакти), і встановлює на виході черговість посилення кодівих символів у канал зв'язку.

Тактова частота переключення й число контактів комутатора у згорткових кодерах визначається відносною швидкістю коду $R = k/n$, де k – число інформаційних символів; n – число переданих у канал зв'язку символів за один такт надходження на кодер інформаційного символу. Відповідно до цього число контактів (комірок реєстра зсуву комутатора) повинне бути рівне n , а частота переключення повинна бути в n разів більше вхідної тактової частоти. Так, при швидкості $R = 1/2$ у комутатора повинно бути 2 контакти й переключення повинне проводитися з подвоєною тактовою частотою.

За аналогією із блоковими кодами, згорткові коди можна класифікувати на систематичні й несистематичні. Систематичним згортковим кодом є такий код, для якого у вихідній послідовності кодівих символів утримується без зміни послідовність інформаційних символів, яка її породила. У протилежному випадку згортковий код є несистематичним.

На рисунку Б.1, а й б додатка Б подано, відповідно, приклади кодерів систематичного й несистематичного згорткового коду для $R = 1/2$. У випадку систематичного

згорткового коду (рисунок Б.1, а) першим з вихідних кодових символів, одержуваних за кожний цикл опитування комутатора, завжди буде черговий інформаційний символ, що надійшов у регістр зсуву. З рисунка Б.1, б можна побачити, що вихідна послідовність кодових символів не містить вхідні інформаційні символи в незмінному вигляді, тому кодер буде породжувати несистематичний згортковий код.

У загальному випадку регістр зсуву кодера згорткового коду (рисунок 2.1) містить m комірок, а комутатор робить один цикл опитування при надходженні $1 < k < m$ чергових інформаційних символів, де m кратне k , опитуючи за один цикл $n > 2$ виходів кодера. При цьому вплив будь-якого вхідного інформаційного символу буде поширюватися на $l_0 = mn/k$ вихідних кодових символів. Ця величина називається повною довжиною кодового обмеження. При цьому довжина кодового обмеження й конкретний вибір зв'язків з комірками регістра зсуву на суматори по модулю 2 будуть визначати коригувальні властивості одержуваного згорткового коду.

Згортковий код зручно задавати за допомогою породжувальних багаточленів, як це робиться для лінійних блокових циклічних кодів. Породжувальні багаточлени повністю визначають структуру двійкового кодера згорткового коду.

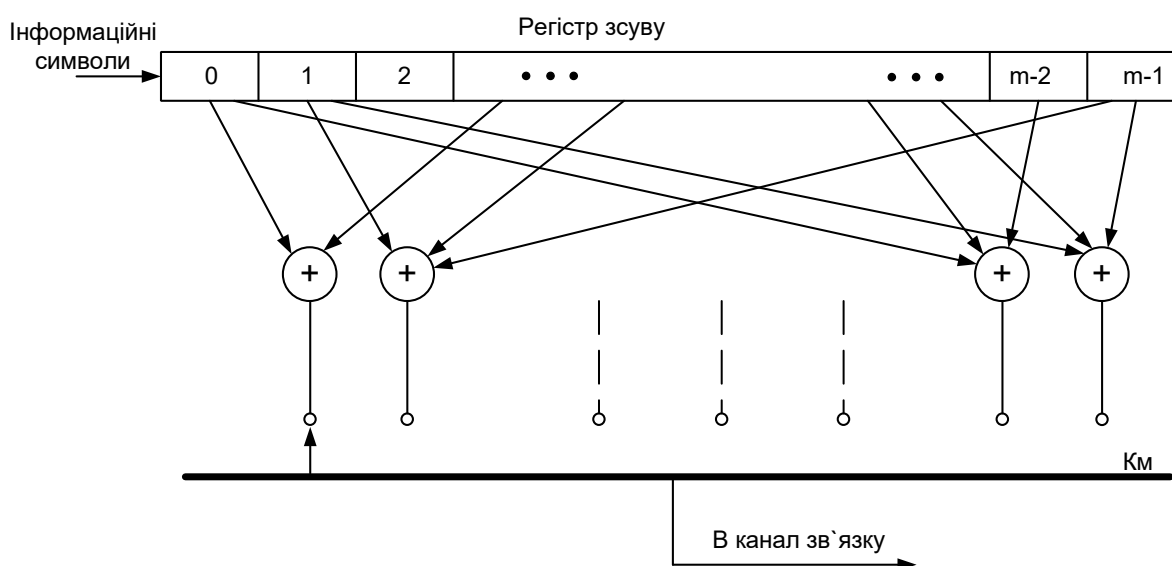


Рисунок 2.1 – Загальний вигляд двійкового згорткового кодера

Наприклад, для кодера систематичного згорткового коду (рисунок Б.1, а) породжувальні багаточлени будуть

$$G_1(X) = 1; \quad G_2(X) = 1 + X^2, \quad (2.1)$$

а для кодера несистематичного згорткового коду (рисунок Б.1, б)

$$G_1(X) = 1 + X + X^2; \quad G_2(X) = 1 + X^2. \quad (2.2)$$

Породжувальні багаточлени можуть бути об'єднані в матрицю $k \times n$ розміру, що називається породжувальною матрицею з багаточленів. Наприклад, породжувальні матриці для кодерів (рисунок 2.1), відповідно до (2.1) і (2.2), записуються у вигляді

$$\begin{aligned} G_1(X) &= [1 \quad 1 + X^2] \\ G_2(X) &= [1 + X + X^2 \quad 1 + X^2]. \end{aligned} \quad (2.3)$$

Рядок у матриці відповідає одному з k символів вхідної послідовності (у цьому випадку $k = 1$), а число багаточленів у рядку дорівнює числу схем підсумовування по модулю 2. При $k > 1$ деякі породжувальні багаточлени можуть дорівнювати нулю.

Розвиток теорії згорткових кодів відбувався в трьох напрямках відповідно до трьох найважливіших методів декодування згорткових кодів: методу порогового декодування, методу послідовного декодування і методу декодування за максимумом правдоподібності (алгоритм Вітербі).

Для повного розуміння процесів кодування та декодування згорткових кодів розглянемо приклад коду $(1/2)$, що є найпростішим кодом, у якого за кожним інформаційним знаходиться контрольний (перевірочний) символ. Довжина контрольних символів при цьому дорівнює довжині інформаційних символів: $m=k=n/2$, отже, надлишковість коду $R = (n-m)100/n = (n-0,5n) \times 100/n = 50\%$.

Принцип побудови згорткового коду ілюструється на рисунку 2.2. Послідовність контрольних символів (нижній рядок на рисунку 2.2) утворюється з послідовності інформаційних символів (верхній рядок на рисунку 2.2) шляхом додавання по модулю 2 інформаційних символів, що відстоять один від одного на постійну відстань l_0 .

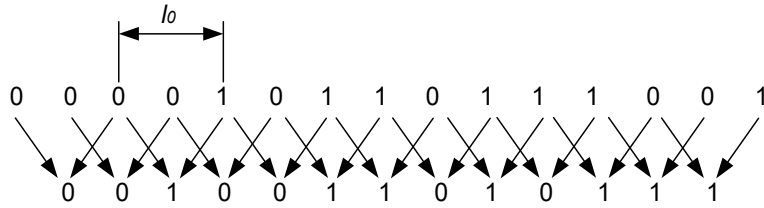


Рисунок 2.2 – Ілюстрація принципу побудови рекурентного коду

Схема систематичного кодера з чотирирозрядним регістром зсуву зі «зв'язком уперед» наведена на рисунку 2.3.

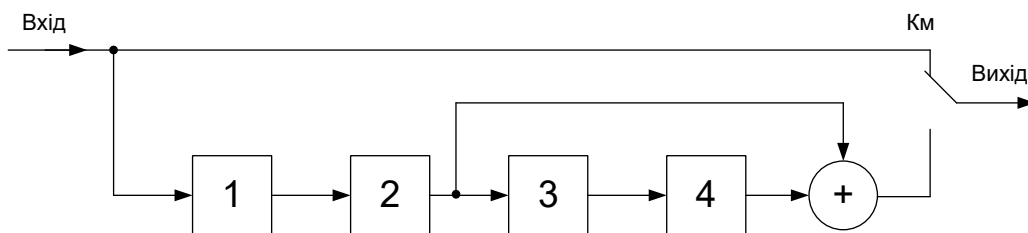


Рисунок 2.3 – Схема кодера згорткового коду

Слід зазначити, що цей кодер може виправляти пакети помилок завдяки перемеженню зі ступенем 2. Пристрій перемеження переставляє символи послідовності з одного місця на інше і таким чином «розносить» спотворені символи вздовж кодової послідовності, тим самим полегшуючи виправлення помилок (особливо пакетів помилок, тобто помилок, що йдуть одна за одною). При цьому швидкість коду не змінюється.

Якщо на вхід кодера подається та ж послідовність символів, що наведена на рисунку 2.2, тобто (2.4), то на виході регістра зсуву утвориться послідовність символів (2.5):

$$1 \ 0 \ 1 \ 1 \ 0 \ 1 \ 1 \ 1 \ 0 \ 0 \ 1 \quad (2.4)$$

$$0 \ 0 \ 1 \ 0 \ 0 \ 1 \ 1 \ 0 \ 1 \ 0 \ 1. \quad (2.5)$$

Утворення послідовності (2.5) ілюструється на рисунку 2.5 і в таблиці Б.1. Послідовність інформаційних символів (2.4) відповідає стану 1-ї комірки регістра в дискретні моменти часу T_i , а послідовність контрольних (перевірочних) символів (2.5) – символам на виході суматора по модулю 2.

Послідовність (2.5) виходить шляхом підсумовування сигналів на виходах другої й четвертої комірок регістра зсуву в попередньому такті, що відповідає підсумовуванню символів другої й четвертої позицій станів комірок регістра в попередньому рядку таблиці Б.1. Комутатор K_m на передавальній стороні видає на вихід послідовність (2.6), що складається з першого інформаційного символу, потім першого перевірного, потім другого інформаційного т.д.:

$$1\ 0\ 0\ 0\ 1\ 1\ 1\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 1\ 0\ 0\ 1\ 0\ 0\ 1\ 1. \quad (2.6)$$

Послідовність (2.6) і є вихідною послідовністю закодованих символів рекурентного коду (1/2).

Розглянемо декодер, що складається із двох частин: 1) схеми, що виробляє виправляючу послідовність; 2) схеми, що виконує виправлення (корекцію) коду. Перша схема наведена на рисунку 2.4. Комутатор K_m декодера працює синхронно й синфазно з комутатором кодера (рисунок 2.3).

Регістр зсуву декодера також складається із чотирьох розрядів (комірок). На вхід декодера подається послідовність символів (2.6), що комутатором K_m розділяється на інформаційну (2.4) послідовність, яка подається на вхід регістра зсуву, і на послідовність контрольних символів (2.5), котра подається на нижній вхід вихідного суматора. У цій схемі чотирирозрядний регістр зсуву має таку ж схему, як і регістр зсуву в кодері. Тому, якщо помилок немає, послідовність символів на виході верхнього суматора (рисунок 2.4) збігається з послідовністю контрольних символів (2.5), що подаються на вхід нижнього суматора. У цьому випадку на *Виході 2* суматора послідовність символів складається з одних нулів, а послідовність символів на *Виході 1* – з послідовності неспотворених символів (2.4).

Якщо в каналі зв'язку між кодером і декодером виникають помилки, то послідовність символів на *Виході 2* містить одиниці в певному сполученні, що дозволяє виправляти помилки. Отже, вона служить виправляючою послідовністю.

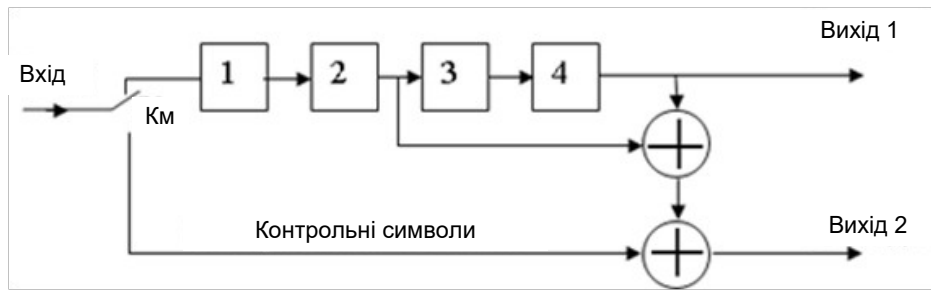


Рисунок 2.4 – Схема формування виправляючої послідовності

Розглянутий код дозволяє виправляти пакет помилок довжиною $l \leq 2l_0 = 4$. Розглянемо приклад виникнення помилок у найгіршому випадку – серії довжиною $2l_0 = 4$. Такий пакет помилок вражає тільки половину інформаційних символів довжиною $l_0 = 2$ і половину контрольних символів довжиною $l_0 = 2$. Припустимо, що відбулася послідовність помилок (2.7). Підсумовуючи послідовності символів (2.6) і (2.7), одержуємо прийняту послідовність (2.8):

$$0\ 0\ 0\ 0\ 0\ 1\ 0\ 1\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0 \quad (2.7)$$

$$1\ 0\ 0\ 0\ 1\ 0\ 1\ 1\ 1\ 0\ 1\ 1\ 1\ 1\ 1\ 0\ 0\ 1\ 0\ 0\ 1\ 1, \quad (2.8)$$

яка в розглянутому прикладі подається на вхід декодера.

Комутатор K_m у схемі на рисунку 2.4 розділяє послідовність (2.8) на інформаційні (2.9) і контрольні символи (2.10):

$$1\ 0\ 1\ 1\ 1\ 1\ 1\ 1\ 0\ 0\ 1 \quad (2.9)$$

$$0\ 0\ 0\ 1\ 0\ 1\ 1\ 0\ 1\ 0\ 1. \quad (2.10)$$

Послідовність символів (2.9) містить 5-й помилковий символ, а (2.10) містить 3-й і 4-й помилкові символи. Регістр зсуву (рисунок 2.4) видасть послідовність символів (2.11), яка в сумі з послідовністю (2.7) дасть виправляючу послідовність (2.12):

$$0\ 0\ 1\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 1, \quad (2.11)$$

$$0\ 0\ 1\ 1\ 0\ 0\ 1\ 0\ 1\ 0\ 0. \quad (2.12)$$

Автоматично операція виправлення послідовності (2.6), що виникає на *Виході 1* декодера за допомогою виправляючої послідовності (2.12) на *Виході 2* (рисунок 2.4), здійснюється за допомогою схеми виправлення помилок, що наведена на рисунку 2.5. Ця схема є продовженням схеми, яка наведена на рисунку 2.4 (*Вихід 1* з'єднується із *Входом 1*, а *Вихід 2* з'єднується із *Входом 2*).

Виправляюча послідовність (2.12) подається безпосередньо на інвертор, що перетворить символи 1 в 0, а 0 в 1 і подає їх на лівий вхід логічного елемента *I* у вигляді послідовності (2.13). Схема нижнього регістра зсуву така ж, як на рисунку 2.4.

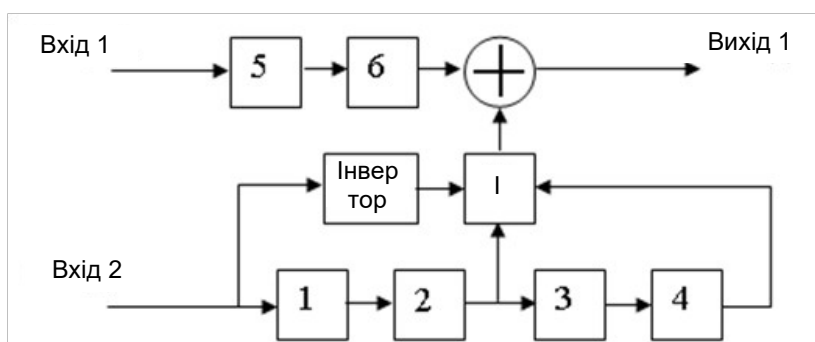


Рисунок 2.5 – Схема декодера, що виправляє помилки

З виходу комірки 2 виправляюча послідовність (2.12) подається зі зсувом $l_0=2$ кроки на нижній вхід елемента *I* у вигляді послідовності (2.14), а з виходу комірки 4 регістра – на правий вхід елемента *I* у вигляді послідовності (2.15), зсунутої на $2l_0=4$ кроки. У результаті на виході елемента *I* одержимо послідовність (2.16)

$$1 \ 1 \ 0 \ 0 \ 1 \ 1 \ 0 \ 1 \ 0 \ 1 \ 1 \dots \quad (2.13)$$

$$\cdot \ \cdot \ 0 \ 0 \ 1 \ 1 \ 0 \ 0 \ 1 \ 0 \ 1 \dots \quad (2.14)$$

$$\cdot \ \cdot \ \cdot \ \cdot \ 0 \ 0 \ 1 \ 1 \ 0 \ 0 \ 1 \dots \quad (2.15)$$

$$\cdot \ \cdot \ \cdot \ \cdot \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 1 \dots \quad (2.16)$$

Крапки в послідовностях ліворуч позначають зсув символів. Одиниця на виході елемента *I* виникає тільки в тих випадках, коли на всі його три входи подаються одиниці. Вона використовується як команда на виправлення помилки.

Видача цієї команди повинна бути узгоджена за часом із вхідною (помилковою) послідовністю інформаційних символів, що надходить на *Вхід 1* декодера. Для цієї мети служать комірки 5 і 6 регістра зсуву на Вході 1, що забезпечують $l_0=2$.

Виправлена послідовність формується на виході схеми у вигляді суми послідовностей символів (2.17) і (2.18), які є послідовностями (2.16) і (2.19), зсунутими відповідно на $2l_0=4$ і $3l_0=6$ кроків відповідно до числа комірок регістрів зсуву, включених послідовно:

$$\begin{array}{cccccccccccc} \dots & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{array} \quad (2.17)$$

$$\begin{array}{cccccccccccc} \dots & \dots & \dots & 1 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 1 \end{array} \quad (2.18)$$

$$\begin{array}{cccccccccccc} & & & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 1 \end{array} \quad (2.19)$$

Після автоматичного виправлення послідовність (2.19) збігається з послідовністю (2.4).

Як видно з (2.19), на шляху інформаційних символів включено $3l_0=6$ комірок регістрів зсуву. Складність апаратури для рекурентного коду оцінюється за числом комірок регістрів зсуву. Кодер має $2l_0=4$ комірки регістра, а декодер - $5l_0=10$ комірок, з яких виправляюча схема має $3l_0=6$ комірок. Для збільшення захищеності й пакета помилок, що виправляються, збільшують n , тобто використовують коди $[(n-1)/n]$. Схеми таких кодерів та декодерів трохи ускладнюються.

2.5 Порядок виконання роботи

Лабораторний макет на тему «Згорткові коди» виконаний за допомогою пакета програм Simulink програмного середовища Matlab. Він реалізує процес кодування і декодування систематичного згорткового коду, приклад якого детально проілюстрований в п. 2.4.

Так само, як і в попередніх роботах, передбачається, що студенти перед виконанням даної лабораторної роботи отримують домашнє завдання у вигляді семирозрядної інформаційної послідовності, яку вони повинні закодувати і декодувати згортковим кодом з пакетом помилок довжиною 4 символи, однократною помилкою в заданому розряді при

передачі по каналу зв'язку, а також без помилки. Подібно до попередніх, розроблений лабораторний макет передбачає виконання однойменної лабораторної роботи в чотирьох режимах:

1) режим роботи кодера та декодера без спотворення переданих інформаційних та контрольних послідовностей і відповідно без помилок;

2) режим роботи кодера та декодера зі спотворенням переданих інформаційних та контрольних послідовностей в заданому (фіксованому) дослідником розряді, що призводить до однократних помилок (пакету помилок);

3) режим роботи кодера та декодера зі спотворенням переданих інформаційних та контрольних послідовностей в певному розряді, що визначається випадковим чином і призводить до однократних помилок (пакету помилок);

4) режим роботи кодера та декодера зі спотворенням переданих інформаційних та контрольних послідовностей пакетом помилок довжиною більше 4 символів, що призводить до неправильного декодування на прийомі.

Таким чином, розглянемо порядок виконання роботи в кожному з зазначених вище режимів.

Порядок виконання роботи в режимі 1 і 2:

1) відкрити файл Convolution_code;

2) встановити КЛЮЧ 1 в положення 1;

3) натиснути два рази на блок ЗМІННИЙ ГЕНЕРАТОР ІНФОРМАЦІЙНОЇ ПОСЛІДОВНОСТІ і відповідно до певного варіанта ввести значення I від 1 до 127 в десятковій формі, в результаті чого блок автоматично перетворює це число в двійкову форму і генерує сигнал у вигляді послідовності логічних одиниці (1) і нуля (0);

4) встановити КЛЮЧ 2 в положення 1 і КЛЮЧ 3 в положення 2 для введення однократної помилки в інформаційній послідовності, або встановити КЛЮЧ 6 в положення 1 і КЛЮЧ 7 в положення 2 для введення однократної помилки в контрольній послідовності;

5) натиснути два рази на блок ВЕКТОР ОДНОКРАТНОЇ ПОМИЛКИ 1-2 і відповідно до певного варіанта ввести значення K (або L) = 0 для режиму 1 та значення K (або L) від 1 до 7 в

десятковій формі для режиму 2, в результаті чого блок автоматично генерує сигнал завади у вигляді послідовності логічних одиниці (1) і нуля (0), який в залежності від введеного числа спотворює кодову послідовність у відповідному розряді;

6) послідовно натиснути: два рази на блок СКИДАННЯ РЕЗУЛЬТАТІВ, один раз на значок Start Simulation, два рази на блок ГЕНЕРАТОР ПОМИЛКИ, один раз на значок Start Simulation, в результаті чого виконується запуск макету;

7) натиснути два рази на блок ОСЦИЛОГРАФ 1, де на вікні можна спостерігати сформовану інформаційну послідовність згідно з заданим варіантом (вона знаходиться в межах часу від 0 до 11 секунд, тобто перші 11 тактів);

8) натиснути два рази на блок ОСЦИЛОГРАФ 2, де можна спостерігати кодову послідовність згорткового коду, яка складається з накладених один на одній (об'єднаних) сигналів, причому інформаційна послідовність жовтого кольору, а контрольна фіолетова (вона знаходиться в межах часу від 0 до 11 секунд);

9) натиснути два рази на блок КОДЕР ЦИКЛІЧНОГО КОДУ і два рази на блок ОСЦИЛОГРАФ 6, де можна спостерігати процес формування контрольної послідовності у вигляді станів ТРИГЕРІВ 1-4 і СУМАТОРА ПО МОДУЛЮ 2 (саме тут студент має можливість перевірити правильність складання таблиць зсуву). Слід зазначити, що значення станів ТРИГЕРІВ і СУМАТОРА ПО МОДУЛЮ 2 зсунуті на 1 такт (1 секунду на часовій осі) через використання елементів затримки (тригерів). Таким чином, процедуру ділення слід спостерігати з 1 по 12 секунди (всього 11 тактів);

10) натиснути два рази на блок ОСЦИЛОГРАФ 7, де можна спостерігати інформаційну і контрольну послідовності (вони знаходяться в межах часу від 1 до 11 секунд);

11) закрити вікно блока ЗГОРТКОВИЙ КОДЕР і натиснути два рази на блок ОСЦИЛОГРАФ 3, де можна спостерігати процес спотворення інформаційної (контрольної) послідовності завадою у вигляді однократної помилки, та ОСЦИЛОГРАФ 4, де зображено вектор відповідних помилок;

12) натиснути два рази на блок ЗГОРТКОВИЙ ДЕКОДЕР і 2 рази на блок ОСЦИЛОГРАФ 8, де можна спостерігати інформаційну послідовність, затриману на 4 такти. Таким чином, спостерігати слід з 4 по 15 секунди (всього 11 тактів);

13) натиснути два рази на блок ОСЦИЛОГРАФ 9, де можна спостерігати контрольні символи, які отримані з каналу зв'язку та контрольні символи, отримані в результаті перетворення отриманих з каналу зв'язку інформаційних символів (так само, як при кодуванні в кодері). Вони знаходяться в межах часу від 0 до 11 секунд;

14) натиснути два рази на блок ОСЦИЛОГРАФ 10, де можна спостерігати виправляючу послідовність;

15) натиснути два рази на блок ОСЦИЛОГРАФ 11, де можна спостерігати процес виправлення спотвореної інформаційної послідовності (виправлення відбувається шляхом підсумовування по модулю 2 спотвореної інформаційної послідовності та виправляючого вектора, отриманого в результаті перетворення в логічній схемі виправляючої послідовності);

16) закрити вікно блока ЗГОРТКОВИЙ ДЕКОДЕР і натиснути два рази на блок ОСЦИЛОГРАФ 5, де можна побачити виправлену інформаційну послідовність, затриману на 6 тактів (через 4 тригери в схемі формування виправляючої послідовності та 2 тригери в схемі виправлення помилок).

Порядок виконання роботи в режимі 3-4:

1) виконати пункти 1.1-1.3 для режимів 1-2;

2) встановити КЛЮЧІ 2-9 в положення 1 або 2 згідно з робочим завданням студента, яке видає викладач;

3) виконати пункти 1.6-1.16 для режимів 1-2.

Порядок виконання роботи в режимі 4:

1) виконати пункти 1.1-1.3 для режимів 1-2;

2) встановити КЛЮЧІ 2-9 в положення 1 або 2 згідно з робочим завданням студента, яке видає викладач (довжина вибраного пакету помилок повинна перевищувати 4 символи);

3) виконати пункти 1.6-1.16 для режимів 1-2.

4) перейти до оформлення звіту з даної лабораторної роботи.

2.6 Зміст звіту

2.6.1 Назва і мета роботи.

2.6.2 Схема лабораторного макета

2.6.3 Хід та результати виконання домашнього завдання у вигляді алгебраїчних виразів згорткових кодів, а також схем кодерів.

2.6.4 Результати виконання експериментального дослідження згорткових кодів в чотирьох режимах за допомогою лабораторного макета в пакеті Simulink.

2.6.5 Висновки з роботи.

Контрольні питання

- 1 Який код називається згортковим і де його застосовують?
- 2 Поясніть основні відмінності згорткових кодів від блокових.
- 3 Поясніть, як утворюється згортка двійкових послідовностей?
- 4 Поясніть основні параметри згорткового коду.
- 5 Як будується породжувальна матриця згорткового коду?
- 6 Поясніть, як будується граф потоку станів кодера згорткового (2,1,3)-коду?
- 7 Як будується таблиця переходів для згорткового (2,1,3)-коду?
- 8 Побудуйте діаграму станів згорткового (3,1,2)-коду.
- 9 Поясніть сутність декодування в згорткових кодах за максимумом правдоподібності.
- 10 Поясніть сутність метода порогового декодування.
- 11 Поясніть сутність метода послідовного декодування.
- 12 Знайдіть породжувальний поліном двійкового згорткового коду, що має максимальну вільну відстань, зі швидкістю $R=1/2$ і довжиною кодового обмеження $n_c = 6$.
- 13 Побудуйте схему кодера для питання 12.
- 14 Проведіть кодування інформаційної послідовності $\{u[n]\} = \{1, 0, 0, 1, 1, 0, 1\}$ за допомогою діаграми станів.

СПИСОК ЛИТЕРАТУРИ

1 Кудряшов В.А., Семенюта Н.Ф. Передача дискретной информации на железнодорожном транспорте: Учеб. для вузов ж.-д. трансп. – 3-е изд., перераб. и доп. – М.: Транспорт, 1986. – 295 с.

2 Кларк Дж., мл., Кейн Дж. Кодирование с исправлением ошибок в системах цифровой связи / Пер. с англ. – М.: Радио и связь, 1987. – 392 с.

3 Теория электрической связи: Учеб. для вузов / А.Г. Зюко, Д.Д. Кловский, В.И. Коржик, М.В. Назаров; Под ред. Д.Д. Кловского. – М.: Радио и связь, 1999. – 432 с.

4 Емельянов Г.А., Шварцман В.О. Передача дискретной информации: Учеб. для вузов. – М.: Радио и связь, 1982. – 240 с.

5 Питерсон У., Уэлдон Э. Коды, исправляющие ошибки / Пер. с англ. Под ред. Р.Л. Добрушина и С.И. Самойленко. – М.: Мир, 1976. – 594 с.

6 Никитин Г.И. Сверточные коды: Учеб. пособие. – С.Пб.: ГУАП, 2001. – 80 с.

7 Месси Дж. Пороговое декодирование. – М.: Мир, 1966. – 132 с.

8 Зигангиров К.Ш. Процедуры последовательного декодирования. – М.: Связь, 1974. – 208 с.

9 Теория кодирования / Т. Касам., Н. Токура, Е. Ивадари, Я. Инагаки. – М.: Мир, 1978. – 576 с.

Додаток А

Таблиця А.1 – Багаточлени, що не зводяться, до 5 ступеня включно

| Степінь | Вид багаточлена | $b = n = 2^m - 1$ |
|---------|---|-------------------|
| 1 | $x + 1$ | 1 |
| 2 | $x^2 + x + 1$ | 3 |
| 3 | $x^3 + x^2 + 1$; $x^3 + x + 1$ | 7 |
| 4 | $x^4 + x + 1$; $x^4 + x^3 + 1$; $x^4 + x^3 + x^2 + x + 1$ | 15 |
| 5 | $x^5 + x^2 + 1$; $x^5 + x^3 + 1$; $x^5 + x^3 + x^2 + x + 1$; | 31 |

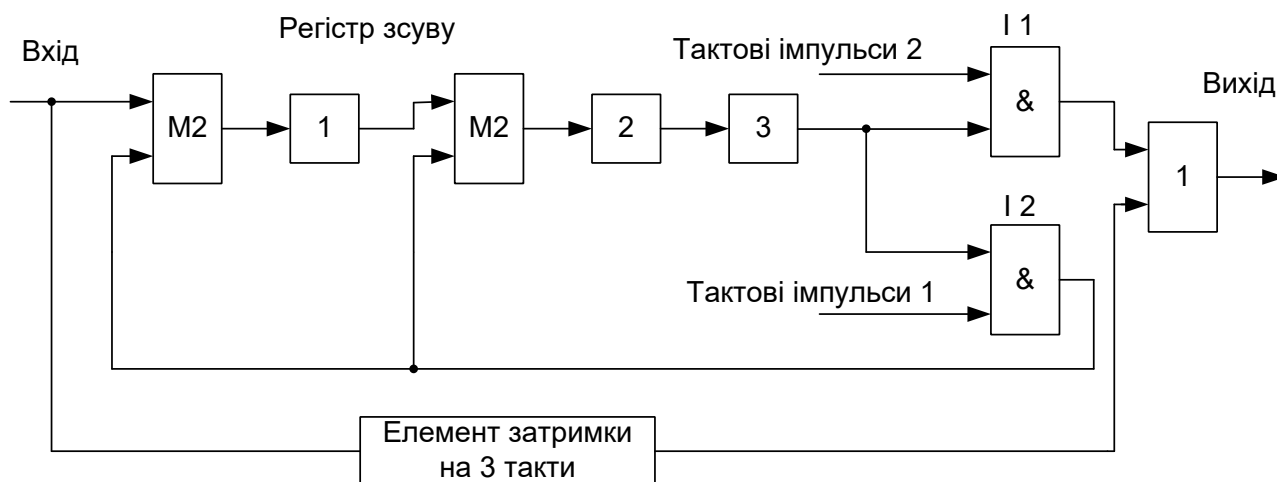


Рисунок А.1 – Структурна схема кодера роздільного циклічного коду

Таблиця А.2 – Проміжні стани елементів регістра при кодуванні

| Такт | Вхід | Стан елементів | | | Вихід | Символ багаточленів |
|---|------|----------------|---|---|-------|---------------------|
| | | 1 | 2 | 3 | | |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| Кодування комбінації вихідного коду | | | | | | |
| 1 | 1 | 1 | 0 | 0 | - | - |
| 2 | 0 | 0 | 1 | 0 | - | - |
| 3 | 1 | 1 | 0 | 1 | - | - |
| 4 | 0 | 1 | 0 | 0 | 1 | x^6 |
| 5 | 0 | 0 | 1 | 0 | 0 | x^5 |
| 6 | 0 | 0 | 0 | 1 | 1 | x^4 |
| 7 | 0 | 1 | 1 | 0 | 0 | x^3 |
| 8 | 0 | 0 | 1 | 1 | 0 | x^2 |
| 9 | 0 | 0 | 0 | 1 | 1 | x^1 |
| 10 | 0 | 0 | 0 | 0 | 1 | x^0 |
| Ділення роздільної кодової комбінації без помилок | | | | | | |
| 1 | 1 | 1 | 0 | 0 | - | - |
| 2 | 0 | 0 | 1 | 0 | - | - |
| 3 | 1 | 1 | 0 | 1 | - | - |
| 4 | 0 | 1 | 0 | 0 | - | - |
| 5 | 0 | 0 | 1 | 0 | - | - |
| 6 | 1 | 1 | 0 | 1 | - | - |
| 7 | 1 | 0 | 0 | 0 | - | - |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| Ділення роздільної кодової комбінації з помилкою | | | | | | |
| 1 | 1 | 1 | 0 | 0 | - | - |
| 2 | 1 | 1 | 1 | 0 | - | - |
| 3 | 1 | 1 | 1 | 1 | - | - |
| 4 | 0 | 1 | 0 | 1 | - | - |
| 5 | 0 | 1 | 0 | 0 | - | - |
| 6 | 1 | 1 | 1 | 0 | - | - |
| 7 | 1 | 1 | 1 | 1 | - | - |

Додаток Б

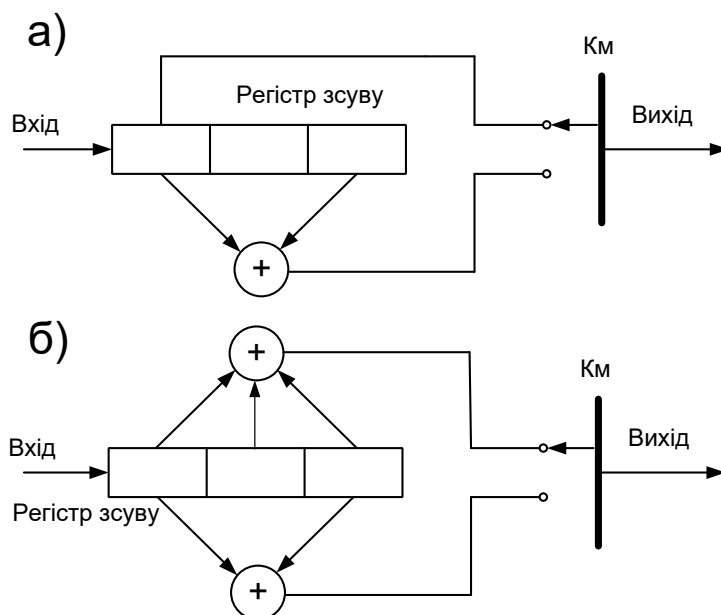


Рисунок Б.1 – Приклади кодерів систематичного (а) і несистематичного (б) згорткового коду

Таблиця Б.1 – Процес кодування у вигляді станів елементів кодера

| Дискретний час (такт) | Стан розрядів регістра | | | | На виході суматора |
|-----------------------|------------------------|-----|-----|-----|--------------------|
| | 1-й | 2-й | 3-й | 4-й | |
| 1 | 1 | 0 | 0 | 0 | 0 |
| 2 | 0 | 1 | 0 | 0 | 0 |
| 3 | 1 | 0 | 1 | 0 | 1 |
| 4 | 1 | 1 | 0 | 1 | 0 |
| 5 | 0 | 1 | 1 | 0 | 0 |
| 6 | 1 | 0 | 1 | 1 | 1 |
| 7 | 1 | 1 | 0 | 1 | 1 |
| 8 | 1 | 1 | 1 | 0 | 0 |
| 9 | 0 | 1 | 1 | 1 | 1 |
| 10 | 0 | 0 | 1 | 1 | 0 |
| 11 | 1 | 0 | 0 | 1 | 1 |

МЕТОДИЧНІ ВКАЗІВКИ

до виконання лабораторних робіт з дисципліни
«Теорія електричного зв'язку»

Відповідальний за випуск Батаєв О.П.

Редактор

Підписано до друку 15.12.08 р.

Формат паперу 60x84 1/16. Папір писальний.

Умовн.-друк.арк. 3,0 Обл.-вид.арк. 3,25.

Замовлення № , Тираж Ціна

Видавництво УкрДАЗТу, свідоцтво ДК № 112 від 06.07.2000 р.

Друкарня УкрДАЗТу,
61050, Харків – 50, пл. Фейєрбаха, 7

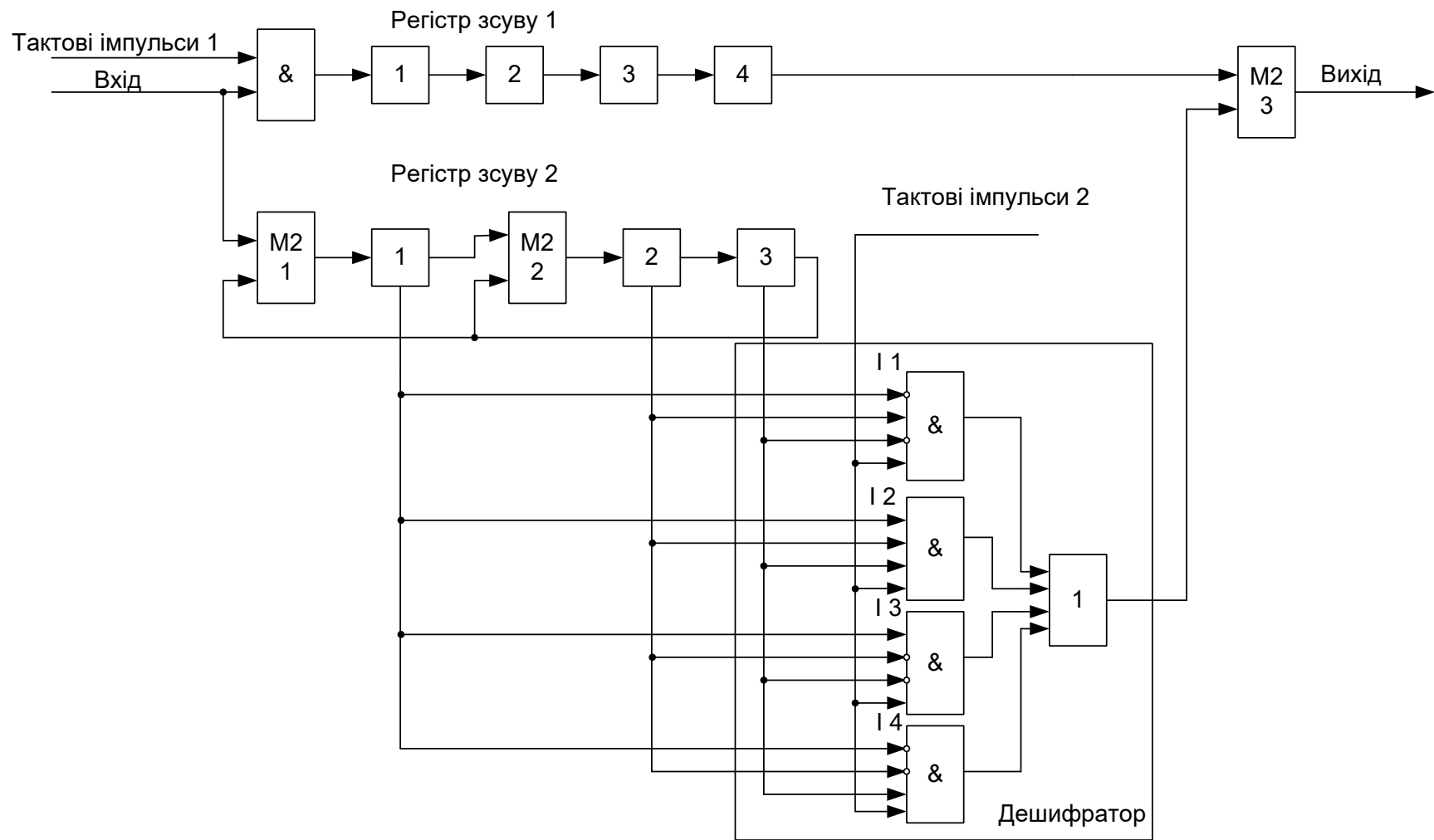


Рисунок А.2 – Структурна схема декодера роздільного циклічного коду (7, 4)