Лекція 4. ЗАВАДОСТІЙКЕ КОДУВАННЯ ІНФОРМАЦІЇ

1. Класифікація завад
2. Принципи завадостійкого кодування
3. Циклічні коди

Еварист Галуа (фр. Évariste Galois) (*25 жовтня 1811 — †31 травня 1832) французький математик, засновник сучасної алгебри.

1. Класифікація завад [3]

Завада - сторонній вплив, що діє в системі передачі і заважає правильному прийому сигналів. Джерела завад можуть знаходитись як зовні, так і всередині самої системи передачі.

Якщо завада регулярна і відома, то боротьба з нею не викликає ускладнень. Однак, ситуація ускладнюється, якщо завади носять випадковий характер.

В загальному випадку дія завади ξ на сигнал, що передається, може бути виражена таким оператором [5]:

$$x = V(s, \xi), \qquad (3.1)$$

де s – повідомлення, яке передається;

ξ – завада;

x – прийняте повідомлення.

В окремому випадку, якщо оператор V вироджується в суму

$$x = s + \xi, \qquad (3.2)$$

то завада, яка діє в каналі, називається адитивною. Адитивну заваду часто називають шумом.

А якщо ж оператор V може бути поданий у вигляді

$$x = \nu \cdot s \ , \qquad (3.3)$$

де випадковий процес $\nu(t)$ невід'ємний, то заваду ν називають мультиплікативною.

Якщо ν змінюється повільно у порівнянні з $s(t)$, то явище, викликане мультиплікативною завадою, називають *завмиранням каналу (федінг)*.

В більш загальному випадку оператор V не може бути приведений до основних форм (3.2), (3.3). При одночасній наявності шуму і мультипликативної завади зручно ввести два випадкових процеси, тобто оператор V подається так:

$$x = \nu \cdot s + \xi. \qquad (3.4)$$

З фізичної точки зору випадкові завади породжуються різного роду флуктуаціями. Флуктуаціями у фізиці називають випадкові відхилення тих або інших випадкових величин від їх середніх значень. Так, джерелом шуму в електричних ланцюгах постійного струму можуть бути флуктуації струму навколо середнього значення, які викликані дискретною природою носіїв зарядів (іонів і електронів). Це явище носить назву дробового ефекту.

Найбільш універсальною причиною шуму є флуктуації, обумовлені тепловими рухами. Випадковий тепловий рух носіїв заряду в будь-якому провіднику викликає випадкову різницю потенціалів на його кінцях. Ця різниця потенціалів флуктує навколо середнього значення рівного нулю, її середній квадрат пропорційний абсолютній температурі. Завада, що виникає, називається тепловим шумом.

Є ще одне джерело принципово неусовного шуму, яке викликано дискретною природою електромагнітного випромінювання. Як відомо, випромінювання виконується дискретними порціями – квантами, які називають фотонами. При високих частотах шуми, викликані дискретною фотонною структурою випромінювання, можуть перевищити всі інші завади.

Природа мультпликативної завади полягає у випадкових змінах параметрів каналу передачі. Причому ці зміни можуть носити як лінійний, так нелінійний характер.

Одним із способів боротьби з випадковими завадами в системах передачі та обробки інформації є застосування кодів, що контролюють помилки. Теорія завадостійкого кодування для кожного конкретного каналу дозволяє вибрати найбільш ефективний метод виявлення і виправлення помилок. Існують два взаємодоповнювальних методи боротьби з завадами [19]:

- кодування з виправленням помилок (коректуючі коди) - приймач виявляє і виправляє помилки;
- кодування з виявленням помилок - приймач розпізнає помилки і, у разі потреби, проводить запит на повторну передачу помилкового блока.

Останній метод припускає наявність каналу зворотного зв'язку і знаходить своє застосування в каналах з достатньо малою імовірністю помилки у випадку, якщо цю імовірність помилки необхідно ще знизити. Така ситуація часто виникає в обчислювальних мережах і в інтернеті.

Теорема Шеннона про кодування для каналу із завадами

Теорема Шеннона задає умови, при яких можлива передача інформації по каналу зв'язку із завадами з як завгодно малою імовірністю помилки. На цій теоремі ґрунтується теорія завадостійкого кодування.

Формулювання теореми

1. При будь-якій продуктивності джерела інформації, меншій ніж пропускна здатність каналу, існує такий спосіб кодування, який дозволяє забезпечити передачу інформації, що створюється джерелом повідомлень з як завгодно малою ймовірністю помилки.
2. Не існує способу кодування, який дозволив би вести передачу інформації з як завгодно малою ймовірністю помилки, якщо продуктивність джерела повідомлення більша пропускної здатності каналу.

Теорема встановлює теоретичну межу можливої ефективності системи при достовірній передачі інформації. Теорема неконструктивна, в тому смислі, що в ній не торкаються питань про шляхи побудови кодів, які забезпечують вказану ідеальну передачу. Однак, обґрунтувавши принципову можливість такого кодування, вона мобілізувала зусилля вчених на розробку конкретних кодів.

2. Принципи завадостійкого кодування

Кодування з виявленням і виправленням помилок є метод обробки повідомлень, призначений для підвищення надійності передачі по цифрових каналах. Ідея виявлення помилок дуже проста. Для передачі використовуються не всі $N_0=m^n$ комбінацій, де m - основа коду, а n – значність (довжина) коду, а лише N комбінацій, причому $N<N_0$ [5].

Комбінації, які використовуються в даному коді, називаються *дозволеними*, а комбінації, що не використовуються, називаються *забороненими*. Якщо сукупність помилок в дозволеній кодовій комбінації перетворює її в іншу дозволену кодову комбінацію, то такі помилки не можуть бути виявлені.

Інтуїтивно зрозуміло, що чим більша відстань між дозволеними кодовими комбінаціями, тим більше помилок виправляє код. Для визначення відстані між двома кодовими комбінаціями прийнято використовувати таку метрику [4-5]:

$$d_{ij} = \sum_{k=1}^{n} \left| x_{ik} - x_{jk} \right|. \qquad (3.5)$$

Визначену таким чином відстань часто називають відстанню Хеммінга. Для двійкового коду відстань просто дорівнює кількості знаків в яких одна кодова комбінація відрізняється від іншої. Зазвичай її визначають шляхом додавання за модулем 2 двох кодових комбінацій і підрахунку кількості одиниць в цій сумі.

Найменшу відстань для даного коду називають *кодовою відстанню* і позначають символом "d_{min}".

Приклад 3.1. Визначити кодову відстань d_{min} для наступного коду: A_1=0000, A_2=0011, A_3=1100, A_4=1010.

Розв'зування. Знайдемо суми за модулем 2 всіх можливих кодових комбінацій і значення відстані між ними як кількість одиниць в отриманій сумі:

0000	0000	0000	0011	0011	1100
+	+	+	+	+	+
0011	1100	1010	1100	1010	1010
0011	1100	1010	1111	1001	0110
$d_{12} = d_{21} = 2$	$d_{13} = d_{31} = 2$	$d_{14} = d_{41} = 2$	$d_{23} = d_{32} = 4$	$d_{24} = d_{42} = 2$	$d_{34} = d_{43} = 2$

Таким чином, найменшим значенням d_{ij} є 2, тобто кодова відстань для даного коду така: d_{min}=2.

Для виявлення всіх помилок кратності q_d в кодовій комбінації необхідно і достатньо, щоб кодова відстань задовольняла таку нерівність [5]:

$$d_{min} \geq q_d + 1. \quad (3.6)$$

Під кратністю помилок розуміють кількість неправильно переданих символів в межах однієї кодової комбінації. Тобто, мінімальне значення $d_{min}=2$.

Для виправлення помилок кратності q_c необхідно і достатньо щоб кодова відстань задовольняла таку нерівність:

$$d_{min} \geq 2q_c + 1. \quad (3.7)$$

Тобто, мінімальне значення $d_{min}=3$.

Для виправлення помилок кратності q_c і виявлення помилок кратності q_d необхідно і достатньо щоб [5]:

$$d_{min} \geq q_c + q_d + 1, \qquad (3.8)$$

причому $q_d \geq q_c$.

Наведені вирази дозволяють будувати прості завадостійкі коди.

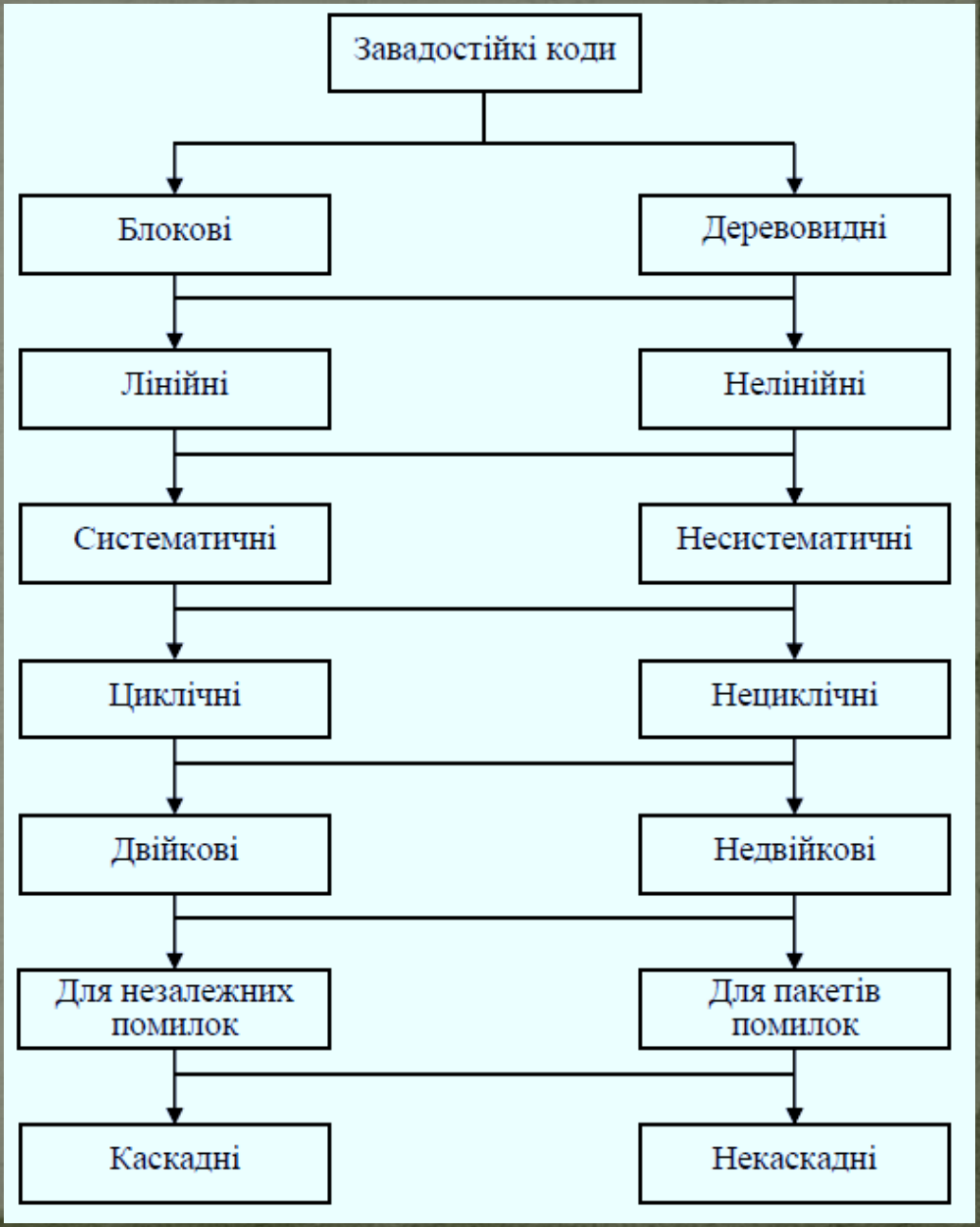
Хоча різні схеми кодування дуже несхожі одна на одну і основані на різних математичних теоріях, всім їм притаманні дві загальні властивості.

Перша - використання надмірності. Закодовані послідовності завжди містять додаткові, або надмірні, символи. Кількість символів в кодовій послідовності завжди більша, ніж необхідно для однозначного подання будь-якого повідомлення.

Друга - властивість усереднювання, це означає, що надмірні символи залежать від декількох інформаційних символів, тобто інформація, що міститься в кодовій послідовності, перерозподіляється також і на надмірні символи [19].

Класифікація завадостійких кодів та їх основні параметри

Всі коди, які виявляють і виправляють помилки, можна розділити на *імовірнісні* та *алгебраїчні*. В імовірнісних кодах здатність виявляти і виправляти помилки ґрунтується на визначені імовірності появи символів (наприклад, в кодах Вагнера). Однак вони не знайшли широкого застосування. Алгебраїчні коди характеризуються тим, що здатність виявляти і виправляти помилки є властивістю їх алгебраїчної структури. Алгебраїчні завадостійкі коди можна розділити на *блокові* та *деревовидні*. [3].



Характеристики завадостійких кодів

1. Довжина коду – n;
2. Довжина інформаційної послідовності - k;
3. Довжина перевірної послідовності - $m=n-k$;
4. Кодова відстань - d_{min};
5. Швидкість коду - $v=k/n$;
6. Надмірність коду –$1/v$;
7. Імовірність виявлення помилки - $p_{вп}$;
8. Імовірність не виявлення помилки (спотворення) - $p_{нв}$.

Способи подання лінійних блокових кодів.

Більшість практичних схем кодування ґрунтуються на лінійних кодах.

Блоковий код довжиною n з 2k словами, які утворюють k-вимірний підпростір Vk, що породжується базисом з k лінійно незалежних векторів, векторного n-вимірного простору Vn, називається лінійним (n, k) кодом. Ці k лінійно незалежних векторів утворюють рядки породжувальної матриці (n, k) коду:

$$G = \begin{bmatrix} g_0 \\ g_1 \\ \cdot \\ \cdot \\ \cdot \\ g_{k-1} \end{bmatrix} = \begin{bmatrix} g_{0,0} & g_{0,1} & \cdots g_{0,n-1} \\ g_{1,0} & g_{1,1} & \cdots g_{1,n-1} \\ \cdots & \cdots & \cdots \\ g_{k,0} & g_{k,1} & \cdots g_{k-1,n-1} \end{bmatrix}$$

Інформаційна послідовність розбивається на повідомлення

$$U = (u_1 , u_2 , \ldots, u_к)$$

довжини k, а процес кодування полягає у відображенні цих повідомлень в кодові слова

$$C = (c_1 , c_2 , \ldots, c_n)$$ довжини n $C = U*G$

Породжувальна (твірна) матриця, наприклад для циклічного коду

$$G = \begin{bmatrix} g_0 & g_1 & \cdots g_{n-k} & 0 & \cdots & 0 \\ 0 & g_0 & g_1 & \cdots g_{n-k} & \cdots 0 \\ \cdots & \cdots & & \cdots & \cdots & \cdots \\ 0 & \cdots & 0 & g_0 & g_1 & \cdots & g_{n-k} \end{bmatrix}.$$

Для компактного запису циклічного (n,k) коду часто використовується породжувальний або твірний поліном g(x) степені n-k, з коефіцієнтами з поля GF(q) (для двійкових кодів q={0,1}):

$$g(x) = g_0 + g_1 x + g_2 x^2 + \ldots + g_{n-k} x^{n-k}.$$

Якщо у вигляді полінома подати інформаційне повідомлення U(x), то кодові слова коду утворюються шляхом множення полінома, що відповідає інформаційному повідомленню на породжувальний поліном коду:

$$C(x) = U(x)g(x).$$

Коефіцієнти твірних поліномів об'єднують в двійкове слово і подають у восьмиричній системі числення, а також твірні поліноми можна записувати шляхом перерахування степенів з ненульовими коефіцієнтами. Наприклад, запис $g(x) = 1 + x + x^4 + x^6$: $g = 1010011_2 = 123_8$; g = (0,1,4,6).

У лінійному двійковому блоковому коді довжиною n символів $2k$ кодових слів утворюють k-вимірний підпростір векторного простору n-послідовностей двійкового поля GF(2).

Поле – це множина математичних об'єктів, з якими можна виконувати такі математичні операції: додавати, віднімати, множити і ділити. Поле, що містить скінченну кількість елементів, називається полем Галуа (G). Для поля, що складається з двох елементів – нуля «0» і одиниці «1» (поле Галуа, позначається GF(2)).

Операції додавання і множення:

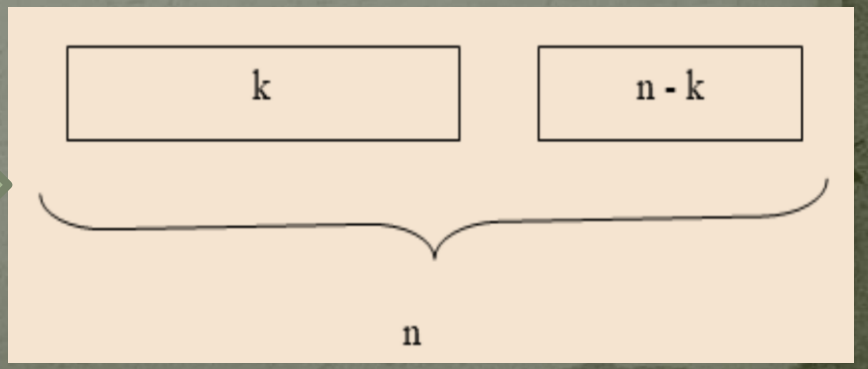
$$0 + 0 = 0, \quad 0 \cdot 0 = 0;$$
$$0 + 1 = 1, \quad 0 \cdot 1 = 0;$$
$$1 + 0 = 1, \quad 1 \cdot 0 = 0;$$
$$1 + 1 = 0, \quad 1 \cdot 1 = 1.$$

Такі операції додавання і множення називаються додаванням за модулем 2 (mod 2) і множенням за модулем 2.

З рівності $1+1 = 0$ витікає, що $1=-1$ і $1+1=1-1$, а з рівності $1 \cdot 1=1$, що $1:1=1$.

До поля GF(2) можна застосувати будь-які методи лінійної алгебри, зокрема матричні операції, а дії в двійкових кодах виконуються за модулем 2.

Отже, лінійний двійковий блоковий систематичний (n,k)-код містить незмінну інформаційну частину завдовжки k символів і надмірну (перевірну), довжиною $n - k$ символів



Прості коди з перевіркою на парність (непарність) – високошвидкісні коди з досить слабкими коригувальними характеристиками. До k інформаційних біт додається $(k+1)$-й біт перевірки на парність, щоб загальна кількість одиниць в кодовому слові була парною *(parity)*. Це $(k+1, k)$-код або $(n, n-1)$**-**код. Для цього коду кодова відстань дорівнює двом**,** тому одиничні помилки тільки виявляються, але не виправляються. Біт парності обчислюється як сума за модулем 2 інформаційних бітів:

$$U_i = (u_1, u_2, \ldots, u_k) \qquad U_{k+1} = u_1 + u_2 + \ldots + u_k ,$$

де знак «+» – операція додавання за модулем 2; $u_1, u_2, \ldots, u_k$ – інформаційні біти повідомлення; u_{k+1} – біт перевірки на парність. Якщо під час передачі послідовність $(u_1, u_2, \ldots, u_k, u_{k+1})$ довжини $k+1$, що містить парне число символів «1», виникне помилка в одному з символів, то кількість одиниць стане непарною і сума

$$u_1 + u_2 + \ldots + u_k + u_{k+1}$$

буде дорівнювати одиниці, що і дозволяє виявляти виникнення одиничних помилок.

Прості коди з повторенням. Це низькошвидкісні коди з високими корегувальними характеристиками. Інформаційний символ повторюється n раз (звичайно n непарне), тобто це (n,1)-коди. Наприклад (3,1)-код формується так:

$$0 – 000, \quad 1 – 111.$$

Кодова відстань дорівнює 3, а кількість помилок, які може виправляти цей код, дорівнює:

$$q_d = (n-1) / 2 = (3-1) / 2 = 1.$$

Приклад. Нехай передається повідомлення, що складається з чотирьох символів A_1, A_2, A_3, A_4, які подано такими кодовими комбінаціями $A_1 = 00$, $A_2 = 01$, $A_3 = 10$, $A_4 = 11$. Побудувати код з перевіркою на парність.

Розв'язання сформуємо біт перевірки на парність для кожної кодової комбінації:

$$A_1 - 0 \oplus 0 = 0,$$
$$A_2 - 0 \oplus 1 = 1,$$
$$A_3 - 1 \oplus 0 = 1,$$
$$A_4 - 1 \oplus 1 = 0.$$

До заданих k = 2 інформаційних бітів дописується (k+1)-й додатковий біт перевірки на парність, отримаємо новий код:

$A_1 = 0\ 0\ 0,$
$A_2 = 0\ 1\ 1,$
$A_3 = 1\ 0\ 1,$
$A_4 = 1\ 1\ 0.$

Визначимо кодову відстань d_{min} для заданого коду. Для цього визначимо кількість одиниць в сумі за модулем 2 всіх пар кодових комбінацій:

000	000	000	011	011	101
+	+	+	+	+	+
011	101	110	101	110	110
011	101	110	110	101	011
$d_1 = 2$	$d_2 = 2$	$d_3 = 2$	$d_4 = 2$	$d_5 = 2$	$d_6 = 2$

Таким чином, $d_{min}=2$ і заданий код може виявляти лише одну помилку в кодовій комбінації (блоці). Якщо виникне дві помилки в межах одного блока, то вони переведуть помилкову комбінацію в іншу дозволену комбінацію з парною кількістю одиниць.

3. Циклічні коди

Циклічні коди (ЦК) були розроблені Хаффманом у середині 50-х років минулого століття на основі теорії полів Галуа двійкових багаточленів $GF(2_n)$. В полях Галуа коефіцієнти багаточленів можуть приймати значення $c_i \in \{0,1\}$, операція додавання елементів поля проводиться за модулем 2. Такими кодами є коди Боуза-Чоудхурі-Хоквінгема (БЧХ) і коди Ріда-Соломона (РС).

Циклічні коди привабливі з двох причин:

- по-перше, операції кодування виконуються дуже просто з використанням регістрів зсуву;
- по-друге, цим кодам властива алгебраїчна структура, і можна знайти прості та ефективні способи їх декодування.

Назву ЦК одержали через найпростіший метод одержання їхніх комбінацій – шляхом циклічного зсуву вихідної *n* - розрядної комбінації вліво з переносом зі старшого розряду в молодший. Так, якщо вихідна комбінація даного ЦК 1100101, то послідовний зсув дає наступні його комбінації: 1001011, 0010111 і т.д. Є також спосіб запису комбінацій ЦК у вигляді багаточленів фіктивної змінної x:

$$F(x) = c_{n-1}x^{n-1} + c_{n-2}x^{n-2} + \ldots + c_1x^1 + c_0x^0$$

де коефіцієнти *C* набувають значення «1» або «0» відповідно до символів кодових комбінацій.

Основні властивості циклічних (n,k)-кодів :

1. Кожен ненульовий поліном має мати степінь, принаймні (n-k), але не більше n-1.
2. Існує тільки один кодовий поліном степені (n-k), який називається породжувальним поліномом коду:

$$g(x)=1+g_1x+g_2x^2+\ldots+g_{n-k}x^{n-k},$$

3. Кожен поліном u(x), що відповідає закодованим даним, є кратним породжувальному поліному g(x), оскільки:

$$u(x)=m(x)\cdot g(x).$$

4. Породжувальний поліном циклічного коду ділить без залишку двочлен x^n+1, тобто $x^n+1=g(x)h(x)+0$. Поліном h(x) – частка від ділення x^n+1 на g(x) – називається перевірним поліномом. Оскільки h(x) і g(x) однозначно пов'язані, то за допомогою перевірного полінома h(x) теж можна проводити кодування, тобто він також визначає код.

Над двійковими поліномами виконують операції додавання, віднімання, множення та ділення.

Відповідно з цим наведена вище комбінація ЦК 1100101 записується як $F(x)_1 = x^6 + x^5 + x^2 + 1$; її циклічні зсуви одержуємо послідовним множенням $F(x)_1$ на x з урахуванням того, що $x^{n-1} \cdot x = x^0 = 1$. Тоді $F(x)_2 = F(x)_1 \cdot x = x^6 + x^3 + x + 1$, $F(x)_3 \cdot x = x^4 + x^2 + x + 1$ і т.д. Підсумовуючи наведені кодові комбінації ЦК одержуємо зокрема, $F(x)_1 + F(x)_2 = x^5 + x^3 + x^2 + x$ або у двійковому вираженні 0101110.

Методика побудови систематичного ЦК

Операція циклічного зсуву – це множення полінома $G(x)$ на x у відповідному степені та приведення результату до стандартної форми. Якщо вихідна кодова комбінація джерела $G(x)$ складається з k розрядів, то процес формування кодової комбінації $F(x)$ циклічного коду, що виправляє одиничні помилки, має вигляд [2]:

$$\left[G(x) \cdot x^r\right] : P(x) = Q(x) + R(x);$$
$$F(x) = G(x) \cdot x^r \oplus R(x),$$

де $P(x)$ – породжувальний поліном степеня r; $R(x)$ – залишок.

Породжувальні поліноми вибирають із таблиці

Поліноми, що не приводяться

Степінь твірного полінома r	Вид полінома, що не приводиться $P(x)$	Значність коду $n = l = 2^r - 1$
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 $x^5+x^4+x^2+1$ $x^5+x^3+x^2+x+1$ $x^5+x^4+x^2+x+1$ $x^5+x^4+x^3+x+1$ $x^5+x^4+x^3+x^2+1$	31

Фрагмент таблиці породжувальних поліномів

Код	Поліном	Код	Поліном
11	$x+1$	10001	x^4+1
101	x^2+1	10011	x^4+x+1
111	x^2+x+1	10101	x^4+x^2+1
1001	x^3+1	10111	x^4+x^2+x+1
1011	x^3+x+1	11001	x^4+x^3+1
1101	x^3+x^2+1	11011	x^4+x^3+x+1
1111	x^3+x^2+x+1	11111	$x^4+x^3+x^2+x+1$

На приймальному кінці лінії зв'язку в декодері виконуються операції перевірки прийнятих комбінацій $\hat{F}(x)$ за таким правилом:

$$\hat{F}(x)/P(x)=Q^*(x)+R^*(x).$$

Якщо залишок $R^*(x)=0$, то це означає, що кодова комбінація прийнята правильно або помилка не виявляється, коли число спотворених символів більше $g=1$.

При $R^*(x)\neq 0$ необхідно виправляти помилку.

Приклад 5.1. У радіосистемі передачі дискретних повідомлень застосовується завадостійке кодування циклічним кодом, здатним виправляти одиничні помилки. Загальна кількість елементарних повідомлень, що передаються, дорівнює M. Визначити кодову комбінацію для повідомлення за номером 10, якщо $M = 16$.

Розв'язання. Визначимо кількість інформаційних і контрольних символів у кодовій комбінації:

$$k = \log M = \log 16 = 4, \quad 2^r - 1 \geq 4 + r, \quad r = 3.$$

Кодова комбінація для повідомлення за номером 10 має вигляд 1010 або подана у вигляді двійкового полінома:

$$G(x) = x^3 + x.$$

Для $r = 3$ обираємо з дод. 2 породжувальний поліном

$$P(x) = x^3 + x^2 + 1.$$

Визначаємо комбінацію циклічного коду за формулою

$$F(x) = G(x) \cdot x^r \oplus R(x),$$

де $R(x)$ – залишок від ділення добутку $G(x) \cdot x^r$ на породжувальний поліном $P(x)$.

$$G(x)\cdot x^r=(x^3+x)\cdot x^3=x^6+x^4,$$

$$\begin{array}{r|l}
x^6+x^4 & x^3+x^2+1 \\
\hline
+\quad x^3+x^2+1 & \\
\hline
x^6+x^5+x^3 & \\
\hline
+\quad x^5+x^4+x^3 & \\
\hline
x^5+x^4+x^2 & \\
\hline
+\quad x^3+x^2 & \\
\hline
x^3+x^2+1 & \\
\hline
R(x)=1 &
\end{array}$$

$F(x)=x^6+x^4+1$, або 1010001.

Приклад 5.2. Цифрова система радіозв'язку використовує завадостійке кодування циклічним кодом, що виправляє одиничні помилки. Довжина кодової комбінації за умов $n = 7$. Прийнята кодова комбінація 1110001. Перевірити, чи правильно прийнята комбінація і за необхідності виправити її, якщо породжувальний поліном має вид:

$$P(x) = x^3 + x^2 + 1.$$

Розв'язання. Запишемо прийняту кодову комбінацію у вигляді двійкового полінома

$$\hat{F}(x) = x^6 + x^5 + x^4 + 1.$$

Перевіримо комбінацію шляхом ділення її на породжувальний поліном $\hat{F}(x) / P(x)$:

$$\begin{array}{l} x^6 + x^5 + x^4 + 1 \\ + \\ \quad x^6 + x^5 + x^3 \\ \hline \quad\quad x^4 + x^3 + 1 \\ \quad + \\ \quad\quad x^4 + x^3 + x \\ \hline R^*(x) = x + 1 \end{array} \qquad \begin{array}{l|l} x^3 + x^2 + 1 & \\ \hline x^3 + x & \end{array}$$

Наявність ненульового залишку $R^*(x)$ свідчить про помилку в прийнятій кодовій комбінації. Для виправлення помилки необхідно знайти вектор помилки $E(x)=x^i,\quad i=\overline{0,n-1}$ і виконати операцію $\hat{F}(x)\oplus E(x)=F(x)$, де $F(x)$ — передана комбінація. Запропоновані різні варіанти розв'язання цієї задачі. Розглянемо найпростіший. Знайдемо еталонний залишок, який відповідає помилці в першому (старшому) розряді кодової комбінації. Для цього поділимо вектор помилки в першому розряді на породжувальний поліном $E(x)/P(x)$:

$$
\begin{array}{r|l}
x^6 & x^3+x^2+1 \\
\hline
+\quad x^6+x^5+x^3 & x^3+x^2+x \\
\hline
\quad x^5+x^3 & \\
\hline
+\quad x^5+x^4+x^2 & \\
\hline
\quad x^4+x^3+x & \\
\hline
\quad R_{em}(x)=x^2+x &
\end{array}
$$

Зафіксуємо значення еталонного залишку.

Залишок, який був одержаний при діленні прийнятої кодової комбінації на породжувальний поліном, не збігається з еталонним залишком:

$$R_i^*(x) \oplus R_{em}(x) = (x+1) \oplus (x^2+x) = x^2+1.$$

Отже, помилка не в першому розряді.

Здійснимо циклічний зсув прийнятої кодової комбінації на один розряд справа наліво і знову поділимо на породжуючий поліном $\left[\hat{F}(x)\cdot x\right] / P(x)$.

$$\hat{F}(x)\cdot x = (x^6+x^5+x^4+1)\cdot x = x^6+x^5+x+1$$

$$\begin{array}{r|l} x^6+x^5+x+1 & x^3+x^2+1 \\ + & \hline x^3+1 \\ \hline x^6+x^5+x^3 & \\ \hline x^3+x+1 & \\ + & \\ \hline x^3+x^2+1 & \\ \hline R_2^*(x) = x^2+x & \end{array}$$

$$R_2^*(x) \oplus R_{em}(x) = (x^2+x) \oplus (x^2+x) = 0$$

Рівність еталонного та одержаного залишків свідчить про помилку в другому розряді. Якщо рівність не досягнута, описаний процес циклічного зсуву і ділення необхідно продовжувати до досягнення результату $R_i^*(x) \oplus R_{em}(x) = 0$. Номер спотвореного розряду визначається як кількість кроків зсуву плюс одиниця.

Висновки:

- завади поділяють на адитивні та мультиплікативні
- спосіб боротьби з перешкодами – застосування завадостійких кодів
- принципи завадостійкого кодування:
  - застосування надмірності при формуванні кодових комбінацій;
  - визначення дозволених і заборонених кодових комбінацій;
  - використання спеціальної метрики у вигляді кодової відстані (Хемінга)

До основних характеристик завадостійких кодів відносять: довжина коду – n; довжина інформаційної послідовності - k; довжина перевірної послідовності - $m=n-k$; кодова відстань - $dmin$; швидкість коду - $v=k/n$; надмірність коду $–1/v$;

- кодові слова містять інформаційну і контрольну складові; вони утворюються за допомогою твірних поліномів.