

## ЛЕКЦІЯ 13. ТЕХНІЧНА РЕАЛІЗАЦІЯ КОДУЮЧИХ ТА ДЕКОДУЮЧИХ ПРИСТРОЇВ

Розглянемо деякі особливості побудови кодуючих та декодуючих пристроїв з урахуванням можливостей великих інтегральних схем (ВІС), що дозволяють помітно скоротити апаратні витрати.

Формування контрольного біта здійснюється суматором по модулю 2 ( $T$ -триггером). Передана в канал двійкова комбінація одночасно подається на вхід  $T$ -триггера. На відповідній часовій позиції імпульсом з розподільника відбувається опитування суматора й при одиничному стані останнього в канал подається "1" (доповнююча передану послідовність до парного числа одиниць), а при нульовому – "0". Аналогічна процедура відбувається на прийомній стороні. Нульовий стан  $T$ -триггера наприкінці прийому блоку свідчить про відсутність одиночних помилок.

Кодуючий пристрій ітеративного коду (рисунок 1) містить паралельний 8-розрядний суматор по модулю  $r$  ( $D1$ ), у якому формується перевірочний елемент стовпця  $q_j$ . Перевірочний біт разом з інформаційною кодовою комбінацією заноситься в паралельно-послідовний регістр  $D3$ . Синхронізація запису здійснюється подачею керуючого імпульсу на вхід  $C2$ . Сформований байт подається на вхід паралельно-послідовного суматора по модулю 2 ( $D4$ ), у якому здійснюється порядкове підсумовування переданих кодових комбінацій. Операція підсумовування в  $D4$  синхронізується імпульсом, що подається на вхід  $C$  у кінці кожного восьмого такту. Не слід забувати, що між синхроімпульсами запису в  $D3$  і  $D4$  повинен бути часовий інтервал, обумовлений затримкою в регістрі  $D3$ . Поелементна видача байта даних у дискретний канал відбувається під дією тактових імпульсів, що зрушують, подаваних на вхід  $C1$  і  $D3$ .

Момент часу подачі ПІ і їхня кількість визначаються керуючим потенціалом, що надходить із розподільника на вхід схеми збігу  $D2$ . Наприкінці інформаційного блоку контрольна сума  $r_1, r_2, \dots, r_7, q_{n+1}$  переписується в регістр зрушення  $D6$  при

наявності керуючого потенціалу із пристрою керування на другому вході  $D5$  і видається в послідовній формі на вхід ППС.

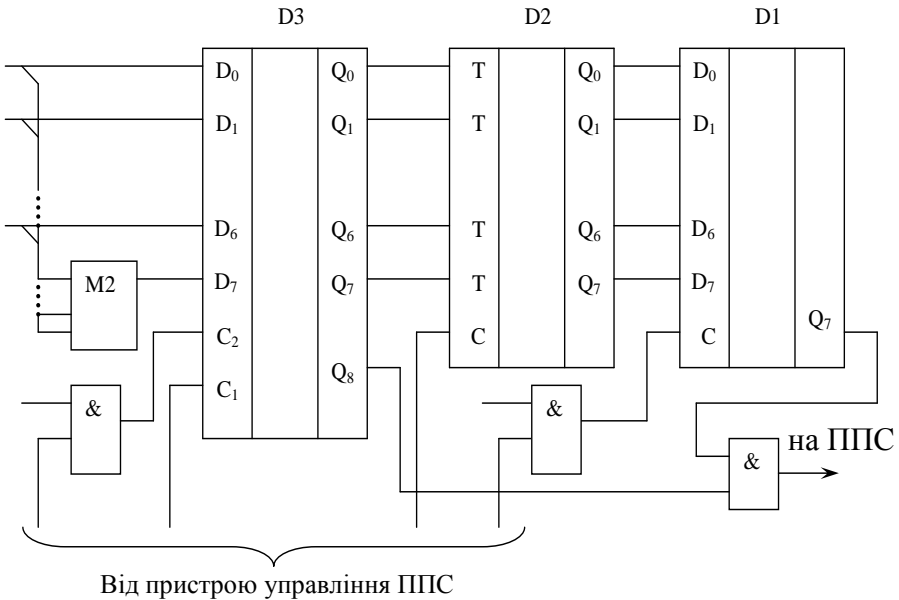


Рисунок 1 – Кодер ітеративного коду

В склад декодувального пристрою ітеративного коду також входять суматори по модулю 2, що здійснюють перевірку на парність поступаючих байтів по стовпцях і рядкам. Так як помилка може бути виявлена в будь-якому стовпці, то її необхідно фіксувати в момент появи. Недотримання ознаки парності по рядках виявляється тільки по закінченні блоку.

Основною операцією в кодуючих та декодуючих пристроях коду Хеммінга є підсумовування по модулю 2. Таким чином, їхні схеми від кодерів ітеративного коду відрізняються утворенням перевірочних елементів. Кодери і декодери кодів Хеммінга досить детально описані, тому відзначимо особливості, які доцільно враховувати при реалізації цих пристроїв. Як відзначалося вище, розташування перевірочних елементів при кодуванні не має принципового значення. Для

спрощення технічної реалізації (виключення багаторозрядних паралельних суматорів, вхідного накопичувача) варто спочатку посилати в канал інформаційні біти, а потім – перевірочні. При такому способі формування контрольних елементів можна здійснювати за допомогою однорозрядних послідовних суматорів по модулю 2 одночасно з передачею інформаційних розрядів. Щоб зберегти коригувальні властивості коду Хеммінга, необхідно зробити перестановку розрядів у перевірочних рівностях з урахуванням зміни номерів суммуємих елементів за рахунок винесення в кінець блоку перевірочних бітів. При такій перестановці рівняння перевірки будуть охоплювати наступні розряди:

$$\begin{aligned}
 P_1 &: 1, 2, 4, 5, 7, 9, 11, 12, 14, \dots; \\
 P_2 &: 1, 3, 4, 6, 7, 10, 11, 13, 14, \dots; \\
 P_3 &: 2, 3, 4, 8, 9, 10, 11, \dots; \\
 P_4 &: 5, 6, 7, 8, 9, 10, 11, \dots
 \end{aligned}
 \tag{1}$$

Біт першої перевірки буде розташовуватися на  $(k+1)$ -й позиції блоку, другий – на  $(k+2)$ -й, останній – на  $n$ -й позиції. На рисунку 16 наведена функціональна схема одного з варіантів формування перевірочних бітів коду Хеммінга, що може бути використана як у кодері, так і в декодері. Інформаційні елементи, що надходять від джерела, подаються на лічильні входи  $T$ -триггерів (суматорів по модулю 2) і через ППС – у канал зв'язку.

Кількість триггерів дорівнює числу контрольних елементів  $r$ . Синхронізація підсумовування здійснюється імпульсами з розподільника, які поєднуються схемами АБО. Входи першого елемента АБО з'єднуються з виходами розподільника відповідно до першої перевірки  $P_1$  (30), другий – перевіркою  $P_2$  й т.д.

Таким чином, забезпечується підсумовування на  $T$ -триггерах тих бітів, номера яких визначаються відповідними перевірками. Після приходу останнього  $k$ -го інформаційного елемента в суматорах буде зафіксовано  $r$  перевірочних елементів.

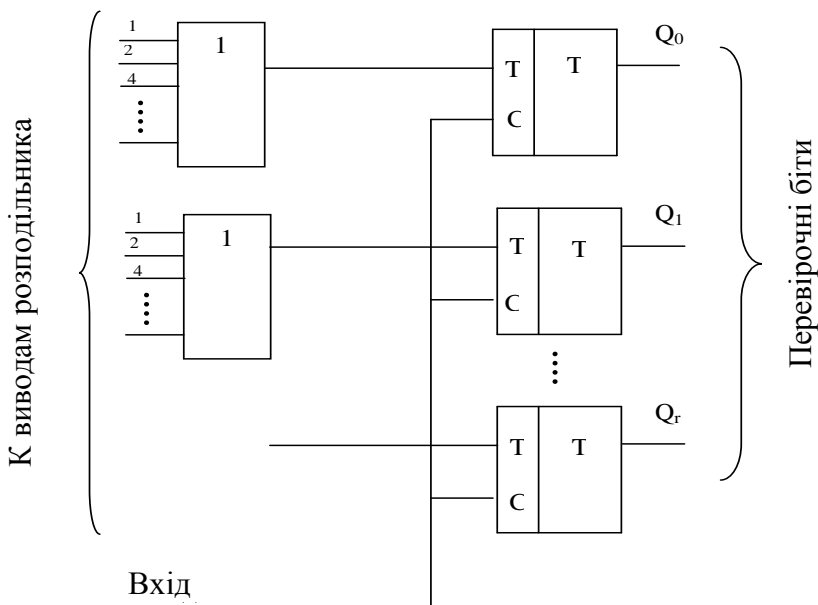


Рисунок 2 – Схема формування перевіірочних елементів коду Хеммінга

У кодуєчому пристрої, ці біти перетворюють у послідовну форму й через ППС надходять у канал зв'язку. На прийомній стороні разом з інформаційними підсумовуванню піддаються й перевіірочні елементи. Отримана  $r$ -розрядна кодова комбінація (синдром помилки) подається на дешифратор, що визначає номер розряду, у якому відбулася помилка. Даний дешифратор відрізняється від класичного (дешифратора коду Хеммінга) і повинен урахувувати перестановки контрольних елементів, зроблених при кодуванні.

Для побудови кодуєчого пристрою, циклічного коду необхідно виконати дві операції: помножити багаточлен  $Q(x)$  на  $x^r$  й отриманий добуток розділити на утворюючий поліном  $P(x)$ . Для виконання першої операції не потрібно спеціального пристрою, тому що множення багаточлена на  $x^r$  означає додавання до нього  $r$  нулів з боку молодшого розряду, тобто після передачі  $k$  інформаційних елементів за ними ідуть  $r$

перевірочних. Як дільники полінома на поліном у кодерах циклічних кодів застосовуються пристрої, побудовані на основі регістрів зрушення зі зворотними зв'язками і суматорів по модулю 2, причому схема дільника визначається видом утворюючого полінома. Кількість тригерів регістра зрушення вибирається рівним показнику степеня утворюючого полінома  $r$ , комірка регістра для старшого ступеня виключається, але завжди є присутнім тригер, що відповідає нульовому степеню  $x^0$ . Число суматорів по модулю 2 у регістрі повинне бути на одиницю менше кількості ненульових членів виразу  $P(x)$ . Суматори розташовують перед комірками регістра, що відповідають ненульовим членам утворюючого полінома. На перші входи суматорів подаються сигнали з попередніх комірок регістра, а на другі – з виходу дільника. Очевидно, що ставити суматори перед коміркою  $x^0$  необов'язково.

Структурна схема пристрою, що дозволяє отримати циклічний код за допомогою утворюючого полінома  $P(x) = x^5 + x^2 + 1$ , наведена на рисунку 3. Прямокутниками на схемі позначені комірки пам'яті, а кружками – суматори по модулю 2. У вихідному стані  $K_1$  замкнений, а  $K_2$  перебуває в положенні 2. Інформація, що підлягає кодуванню, надходить одночасно на вхід кодера й через суматор – на схему ділення на  $P(x)$ . Ділення починається з приходом першого інформаційного елемента й кінчається після видачі в дискретний канал  $k$ -го біта. Після цього схема керування ПЗП переводить  $K_2$  у положення 1, розмикає  $K_1$  і протягом наступних  $r$  тактів здійснюється видача в УПС залишку від ділення  $R(x)$ , що був зафіксований тригерами дільника. При виведення залишку з

дільника необхідно припинити введення даних із джерела. Функціональна схема кодера (рисунок 4) відповідає наведеній на рисунку 3.

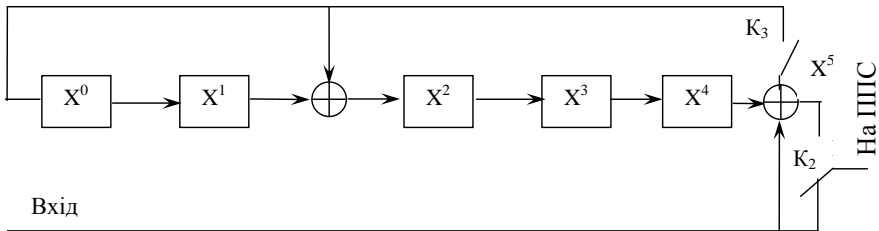


Рисунок 3 – Структурна схема пристрою, який формує циклічний код

Основу декодуючих пристроїв циклічних кодів також становлять дільники багаточленів на утворюючий поліном. Ознакою наявності помилок у прийнятій послідовності є ненульовий залишок від ділення цієї послідовності на  $P(x)$ . До завершення процесу ділення необхідно запам'ятовувати блок, що надійшов, у буферному накопичувачі. Після закінчення циклу відбувається опитування дільника, і у випадку помилки прийнятий блок стирають. При нульовому залишку блок виводиться одержувачеві через ключовий елемент  $Kл$ , а на його місце записується наступний. Структурна схема декодера зображена на рисунку 5. При реалізації схеми варто пам'ятати, що в буферний регістр потрібно записувати тільки інформаційні елементи, а на дільник повинні надходити всі символи блоку, які брали участь у процесі ділення в кодері.

Для виправлення помилок у прийомній частині ПЗП спочатку знаходять помилкові розряди, а потім змінюють їхнє значення, тобто інвертують. Для цього:

1) прийнятий блок ділять на утворюючий поліном  $P(x)$ . Наявність ненульового залишку свідчить про те, що послідовність прийнята спотвореною;

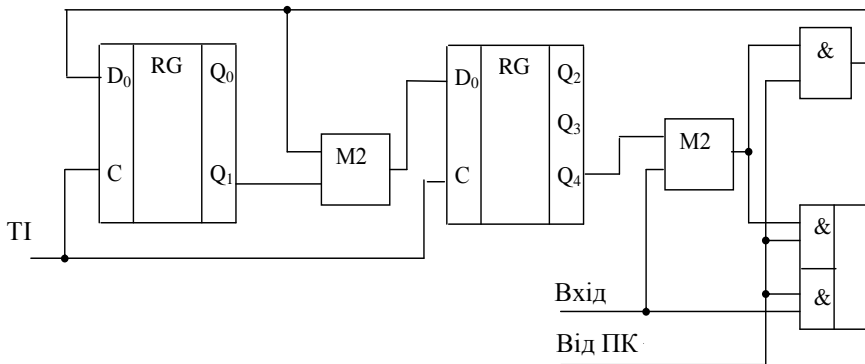


Рисунок 4 – Функціональна схема кодера циклічного коду

2) підраховують кількість одиниць у залишку (вагу залишку)  $W$ . Якщо  $W \leq t_u$ , де  $t_u$  – припустиме число помилок, що виправляють даним кодом, то прийняту послідовність сумують по модулю 2 з отриманим залишком з боку молодших розрядів. У результаті підсумовування одержують виправлений блок даних;

3) при  $W > t_u$  роблять циклічне зрушення прийнятої послідовності  $F(x)$  на один біт убік старших розрядів (уліво) і отриману комбінацію знову ділять на утворюючий поліном. Визначають вагу залишку й при  $W < t_u$  ділене підсумують із залишком і потім циклічно зрушують його у зворотну сторону на один розряд (повертають на колишнє місце). У результаті формують виправлену комбінацію;

4) якщо після зрушення на один біт як і раніше  $W > t_u$ , то роблять додаткові циклічні зрушення вліво. При цьому після кожного зрушення отриману комбінацію ділять на  $P(x)$  й перевіряють вагу залишку. При  $W < t_u$  комбінацію, отриману в результаті останнього циклічного зрушення, сумують із залишком від ділення цієї комбінації на утворюючий багаточлен, а потім роблять циклічне зрушення вправо на

стільки розрядів, на скільки була зрушена сумують з останнім залишком комбінація щодо вихідної послідовності. У результаті одержують виправлений інформаційний блок.

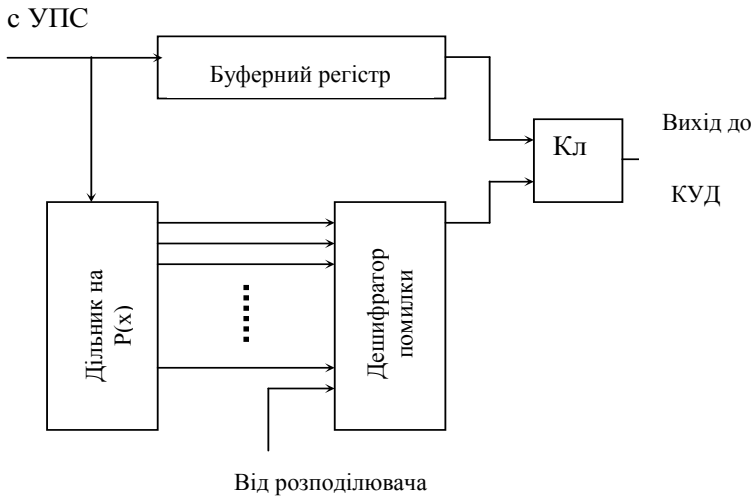


Рисунок 5 – Структурна схема декодера циклічного коду

**Приклад 1.** На вхід прийомної частини ПЗП надходять комбінації циклічного коду 15,7, одержувані за допомогою утворюючого полінома  $P(x) = x^8 + x^7 + x^6 + x^4 + 1$ . У процесі впливу перешкод два біти кодової комбінації можуть бути спотворені. Необхідно виявити наявність помилок в 15-розрядній послідовності, що надійшла, і зробити їхнє виправлення. Розглянути на конкретному прикладі виправлення двократних помилок і побудувати функціональну схему декодера.

**Рішення.** Нехай у дискретний канал передана комбінація 100 000 011 101 000, що на виході прийомної частини УПС прийняла вид 111000 011 101 000. Розділимо прийняту послідовність на утворюючий поліном  $P(x) = 111 010 001$ :



$$\begin{array}{r|l}
 + & 111\ 000\ 011\ 101\ 000 \\
 & \underline{111\ 010\ 001} \\
 & +\ 10\ 010\ 101\ 0 \\
 & \quad \underline{11\ 101\ 000\ 1} \\
 & \quad +\ 1\ 111\ 101\ 10 \\
 & \quad \quad \underline{1\ 110\ 100\ 01} \\
 & \quad \quad \quad 1\ 001\ 110.
 \end{array}$$

Таким чином,  $R(x) = 01001110$ , тобто вага залишку  $W = 4$ , що свідчить про наявність помилки в прийнятій комбінації.

Зрушуємо циклічно цю комбінацію вліво на один біт, у результаті чого вона приймає вид  $110\ 000\ 111\ 010\ 001$ . Розділимо отриману послідовність знову на  $P(x)$ . Залишок від ділення дорівнює  $10011100$ , тобто  $W = 4$ . Зрушуємо циклічно комбінацію ще раз, після чого вона приймає вид  $100\ 001\ 110\ 100\ 011$ . Розділивши її на  $111\ 010\ 001$ , одержимо залишок  $11101001$ , вага якого  $W = 5$ . Після третього зрушення й ділення на утворюючий поліном вага залишку  $00000011$  стає рівним двом. Так як  $W \leq t_u = 2$ , то складаємо кодову комбінацію, отриману в результаті останнього зрушення із залишком:  $000\ 011\ 101\ 000\ 111 + 000\ 000\ 11 = 000\ 011\ 101\ 000\ 100$ . Зробивши циклічне зрушення отриманої суми вправо, маємо виправлену послідовність  $100\ 000\ 011\ 101\ 000$ , що збігається з переданою в канал зв'язку. **Приклад закінчено.**

Декодування циклічних кодів методом обчислення залишку застосовується, якщо  $k \cdot t_u > n$ ; при  $k \cdot t_u < n$  код тільки виявляє, але не виправляє помилки.

Декодувальний пристрій (рисунок б), реалізує описаний алгоритм виправлення помилок для циклічного коду, розглянутого в прикладі 5. Декодер складається з дільника, виконаного для ділення на багаточлен  $P(x) = x^8 + x^7 + x^6 + x^4 + 1$ , і запам'ятовуючого пристрою (ЗП) ємністю  $n = 15$  біт, розділеного на дві частини: одна для зберігання контрольних  $r$ , а друга для інформаційних  $k$  розрядів. Між цими частинами включений суматор по модулю 2, що може підсумувати побітно

дані з регістра перевірочних елементів  $r$  і регістра дільника. Результат додавання надходить у регістр  $k$  інформаційних елементів. Ключі  $ДО_1-ДО_4$  забезпечують роботу декодера в режимах виявлення, виправлення помилок і виведення інформації споживачеві. Відкриття й замикання ключів здійснюються відповідно до алгоритму декодування в певні такти сигналами пристрою керування (на рисунку 20 не показано), що забезпечує також формування імпульсів зрушення інформації в регістрах.

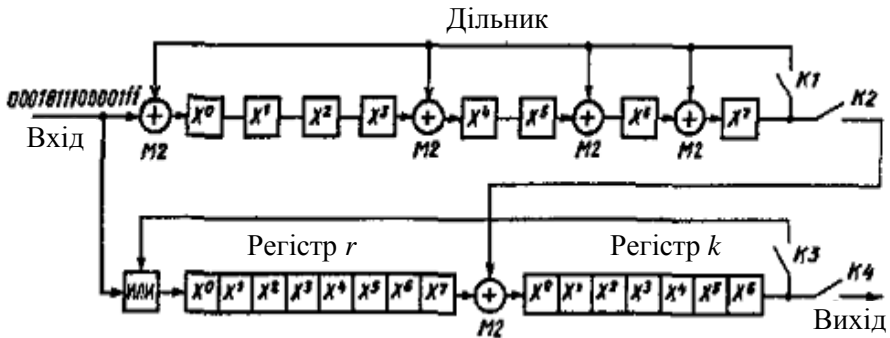


Рисунок 6 – Декодувальний пристрій

У вихідному стані всі ключі декодера розімкнуті. Протягом перших восьми тактів відбувається заповнення дільника, а його комірки пам'яті приймають стани, показані в таблиці 3. За цей час у ЗП будуть занесені вісім старших розрядів комбінації, що надійшла, два з яких спотворені.

На 9-му такті замикається  $K_1$  і протягом семи тактів, що залишилися, здійснюється ділення прийнятої комбінації на утворюючий поліном. По закінченні циклу наприкінці 15-го такту в регістрі дільника сформується залишок від ділення (синдром) 0100 1110 (див. таблицю 1). Аналізуючи отриманий синдром, пристрій керування ухвалює рішення щодо циклічного зрушення прийнятої послідовності на один біт убік старших розрядів (на схемі рисунку 20 – вправо) і про продовження ділення. Для цього ключ  $ДО_3$  замикається, а подача інформації

Таблиця 1 – Стани комірок пам'яті дільника

Номер такту	Ділене	Стан осередків дільника								Вага залишку
		$x^0$	$x^1$	$x^2$	$x^3$	$x^4$	$x^5$	$x^6$	$x^7$	
8	1	1	0	0	0	0	1	1	1	
9	1	0	1	0	0	1	0	0	0	
10	1	1	0	1	0	0	1	0	0	
11	0	0	1	0	1	0	0	1	0	
12	1	1	0	1	0	1	0	0	1	
13	0	1	1	0	1	1	1	1	1	
14	0	1	1	1	0	0	1	0	0	
15	0	0	1	1	1	0	0	1	0	4
1		0	0	1	1	1	0	0	1	4
2		1	0	0	1	0	1	1	1	5
3		1	1	0	0	0	0	0	0	2

на вхід декодера припиняється. На 1-му такті наступного циклу здійснюється ділення зрушеної послідовності на  $P(x)$  й знову аналізується залишок від ділення. Так як  $W > 2$ , то інформація в регістрах зрушується ще на один такт. Така операція буде повторюватися, поки вага залишку не буде  $W \leq 2$ . Для нашого приклада це відбудеться на 3-му такті (див. таблицю 1). Стан комірок регістрів  $r$  і  $k$  в кінці 15-го такту першого циклу й п'ятнадцяти другого показані в таблиці 4 (перекручені біти в ній відзначені крапками). Під дією керуючих сигналів  $DO_1$  розмикається,  $K_2$  – замикається, а інформація продовжує зрушуватися по комітках регістрів. Починаючи с 4-го такту, біти синдрому з виходу дільника надходять на суматор по модулю 2 запам'ятовуючого пристрою. З метою зменшення апаратних витрат підсумовування здійснюється побітно послідовно. Перші шість нулів синдрому, що надходять на суматор, не впливають на вміст ЗП. Лише під час тактових інтервалів 10 і 11 дві одиниці залишку, складаючись по модулю 2 із двома помилковими одиницями інформаційних елементів, інвертують їх, тобто виправляють помилки. Дані в ЗП продовжують зрушуватися до закінчення 15-го такту другого циклу, наприкінці якого  $K_2$  і  $DO_3$  розмикаються, а  $DO_4$  замикається: починається видача виправленої комбінації споживачеві й одночасно запис нової.

Зверніть увагу, на відміну від розглянутого чисельного приклада, інформація після підсумовування зрушується не у зворотному напрямку, а продовжує зміщатися циклічно у бік старших розрядів доти, поки всі біти послідовності не стануть на свої місця, тобто замість повернення на  $i$  кроків триває циклічне зрушення в колишньому напрямку на  $n-i$  біт. Так як регістр замкнений у кільце, то обидві ці процедури еквівалентні й відрізняються тільки часом затримки видачі даних споживачеві. Зрушення інформації в одну сторону викликане тим, що в більшості регістрів зрушення здійснюється тільки в одному напрямку. Зрушення інформації в ЗП може бути реалізовано на основі реверсивних регістрів, однак схема при цьому трохи ускладниться.

Таблиця 2 – Стан комірок регістрів  $r$  і  $k$  в кінці 15-го такту першого циклу й п'ятнадцяти другого

Номер такту	Регістр $r$								Регістр $k$							
	$x^{10}$	$x^1$	$x^2$	$x^3$	$x^4$	$x^5$	$x^6$	$x^7$	$x^0$	$x^1$	$x^2$	$x^3$	$x^4$	$x^5$	$x^6$	
15	0	0	0	1	0	1	1	1	0	0	0	0	1	1	1	
1	1	0	0	0	1	0	1	1	1	0	0	0	0	1	1	
2	1	1	0	0	0	1	0	1	1	1	0	0	0	0	1	
3	1	1	1	0	0	0	1	0	1	1	1	0	0	0	0	
4	0	1	1	1	0	0	0	1	0	1	1	1	0	0	0	
5	0	0	1	1	1	0	0	0	1	0	1	1	1	0	0	
6	0	0	0	1	1	1	0	0	0	1	0	1	1	1	0	
7	0	0	0	0	1	1	1	0	0	0	1	0	1	1	1	
8	1	0	0	0	0	1	1	1	0	0	0	1	0	1	1	
9	1	1	0	0	0	0	1	1	1	0	0	0	1	0	1	
10	1	1	1	0	0	0	0	1	0	1	0	0	0	1	0	
11	0	1	1	1	0	0	0	0	0	0	1	0	0	0	1	
12	1	0	1	1	1	0	0	0	0	0	0	1	0	0	0	
13	0	1	0	1	1	1	0	0	0	0	0	0	1	0	0	
14	0	0	1	0	1	1	1	0	0	0	0	0	0	1	0	
15	0	0	0	1	0	1	1	1	0	0	0	0	0	0	1	

Труднощі реалізації кодерів та декодерів циклічних кодів на основі мікропроцесорів полягає в тому, що операція ділення багаточленів займає багато часу. Так, при діленні 32-розрядного слова на 16-розрядне центральний процесор серії КР580 затрачає час близько 2,5 мс. А подальше збільшення розрядності діленого

практично неможливо через обмежене число робочих реєстрів процесора. З метою спрощення кодування й декодування циклічних кодів розроблені спеціальні методи, що дозволяють скоротити витрати машинного часу. Один з таких методів [2] припускає здійснювати ділення  $k$ -розрядних послідовностей по частинам. Сутність його полягає в тому, що вихідний  $k$ -розрядний блок розбивається на  $l_p$ -розрядні комбінації. Тому що в загальному випадку  $k$  не кратно  $l_p$ , то виходить  $S_c$  частин, для яких виконується умова  $k = l_p(S_c - 1) + m_0$ , де  $m_0 < l_p$ . При діленні виділеної частини на утворюючий поліном  $r$ -й степеня повинне виконуватися співвідношення  $l_p < r$ , причому  $l_p$  доцільно вибирати рівним довжині машинного слова. У постійний запам'ятовувальний пристрій (ПЗУ) мікропроцесора заноситься заздалегідь розрахована таблиця  $r$ -розрядних залишків, отриманих у результаті ділення  $l_p$ -розрядної комбінації доповненої з боку молодших розрядів  $r$  нулями на утворюючий поліном  $P(x)$ . Так як кожному значенню  $l_p$ -розрядного слова відповідає свій залишок, то розмір таблиці становить  $2^{l_p}$ -розрядних слів. Причому для спрощення адресації таблиці  $l_p$ -розрядне число доцільно використати як зсув.

Таким чином, операція одержання залишку шляхом ділення багаточленів замінюється читанням залишку із ПЗУ, що значно скорочує витрати машинного часу. Алгоритм кодування зводиться до наступного.

1. З вихідної  $k$ -розрядної інформаційної послідовності виділяється комбінація довжиною  $l_p$  біт (з боку старших розрядів блоку) і з таблиці вибирається відповідний їй залишок.

2. Отриманий залишок сумується по модулю 2 зі старшими розрядами частини, що залишилася, блоку  $k - l_p$  довжиною біт.

3. З отриманої суми виділяється наступна  $l_p$ -розрядна комбінація, для якої з таблиці зчитується відповідний залишок, і т.д.

4. Через  $S_q - 1$  таких кроків з отриманої суми виділяються  $m_0$  старших розрядів, виділена комбінація доповнюється з боку старших розрядів  $l_p - m_0$  нулями й для сформованої  $l_p$ -розрядної комбінації вибирається відповідний залишок з таблиці.

5. Отриманий залишок підсумується по модулю 2 із бітами, що залишилися (після виділення  $m_0$  розрядів). Ця сума є комбінацією перевірочних розрядів циклічного коду.

*Методичні вказівки до виконання курсового проекту "Пристрій захисту від помилок" по курсу "Системи передачі даних" / Укладачі Кулик І.А., Зубань О.Ю. – Суми, Видавництво СумДУ, 2008. – 73 с. (розділ "Технічна реалізація кодуєчих та декодуєчих пристроїв")*

*Чернега В.С., Василенко В.А., Бондарев В.Н. Расчет и проектирование технических средств обмена и передачи информации: Учеб. пособие для вузов. – М.: Высш. шк., 1990. – 224 с. (с. 129-139).*

*Жураковский Ю.П. Передача информации в ГАП. – Киев: Вища школа, 1991. – 216 с. (с. 171-190)*