

ФАКУЛЬТЕТ АВТОМАТИКИ, ТЕЛЕМЕХАНІКИ ТА ЗВ'ЯЗКУ

Кафедра транспортного зв'язку

МЕТОДИЧНІ ВКАЗІВКИ

до практичних занять

з дисциплін

***«СИСТЕМИ ПЕРЕДАЧІ В ЕЛЕКТРОЗВ'ЯЗКУ»,
«БАГАТОКАНАЛЬНІ СИСТЕМИ ПЕРЕДАЧІ ІНФОРМАЦІЇ»***

Харків - 2014

Методичні вказівки розглянуто і рекомендовано до друку на засіданні кафедри транспортного зв'язку 3 лютого 2013 р., протокол № 7.

У методичних вказівках пояснюється розв'язання типових задач, що розглядаються на практичних заняттях з навчальних дисциплін «Системи передачі в електрозв'язку» та «Багатоканальні системи передачі інформації». Також наведені варіанти завдань для самостійного рішення. Рекомендуються для студентів усіх форм навчання факультету АТЗ з напрямів підготовки «Автоматизація та комп'ютерно-інтегровані технології» та «Телекомунікації» і слухачів ІППК.

Укладачі:

доценти К.А. Трубочанінова,
О.С. Жученко

Рецензент

проф. О.А. Серков

МЕТОДИЧНІ ВКАЗІВКИ

до практичних занять
з дисциплін

*«СИСТЕМИ ПЕРЕДАЧІ В ЕЛЕКТРОЗВ'ЯЗКУ»,
«БАГАТОКАНАЛЬНІ СИСТЕМИ ПЕРЕДАЧІ ІНФОРМАЦІЇ»*

Відповідальний за випуск Трубочанінова К.А.

Редактор Етколо О.О.

Підписано до друку 17.03.14 р.

Формат паперу 60x84 1/16. Папір писальний.

Умовн.-друк.арк. 2,5. Тираж 100. Замовлення №

Видавець та виготовлювач Українська державна академія залізничного транспорту,
61050, Харків-50, майдан Фейербаха, 7.
Свідоцтво суб'єкта видавничої справи ДК № 2874 від 12.06.2007 р.

Українська державна академія залізничного транспорту

ФАКУЛЬТЕТ АВТОМАТИКИ, ТЕЛЕМЕХАНІКИ ТА
ЗВ'ЯЗКУ

Кафедра «Транспортний зв'язок»

МЕТОДИЧНІ ВКАЗІВКИ

до практичних занять
з дисциплін

*«СИСТЕМИ ПЕРЕДАЧІ В ЕЛЕКТРОЗВ'ЯЗКУ»
«БАГАТОКАНАЛЬНІ СИСТЕМИ ПЕРЕДАЧІ ІНФОРМАЦІЇ»*

К.А. Трубчанінова, О.С. Жученко

*Декан факультету АТЗ
Прогонний
Завідувач каф. ТЗ
Голова науково-метод. комісії*

*О.М.
С.І. Приходько
Н.А. Корольова*

Автори:

К.А. Трубчанінова
О.С. Жученко

Харків 2014

Методичні вказівки розглянуто і рекомендовано до друку на засіданні кафедри «Транспортний зв'язок», 3 лютого 2013 р., протокол № 7

У методичних вказівках пояснюється розв'язання типових задач, що розглядаються на практичних заняттях з навчальних дисциплін «Системи передачі в електрозв'язку» та «Багатоканальні системи передачі інформації». Також наведені варіанти завдань для самостійного рішення. Рекомендуються для студентів усіх форм навчання факультету АТЗ з напрямів підготовки «Автоматизація та комп'ютерно-інтегровані технології» та «Телекомунікації» і слухачів ІППК.

Укладачі:

доценти К.А. Трубчанінова,
О.С. Жученко

Рецензент

проф. О.А. Серков

ЗМІСТ

Вступ.....	4
Завдання 1. Компандування ІКМ-сигналів.....	5
1.1 Компандування за А - законом.....	7
1.1.1 Алгоритм 1: Пряме кодування.....	9
1.1.2 Алгоритм 2: Перетворення на основі лінійного коду. .	10
1.2 Компандування за μ - законом.....	12
1.2.1 Алгоритм 1: Пряме кодування.....	13
1.2.2 Алгоритм 2: Перетворення на основі лінійного коду. .	15
Контрольні питання.....	17
Завдання 2. Цифрове лінійне кодування сигналів.....	18
2.1 Вимоги до методів цифрового кодування.....	18
2.2 Методи цифрового кодування.....	20
2.3 Логічне кодування.....	27
Контрольні питання.....	35
Завдання 3. Об'єднання та роз'єднання	
цифрових потоків.....	35
3.1 Способи об'єднання цифрових потоків.....	35
3.2 Об'єднання синхронних цифрових потоків.....	36
3.3 Об'єднання асинхронних цифрових потоків.....	40
3.4 Структурна схема об'єднання (роз'єднання)	
асинхронних цифрових потоків.....	43
Контрольні питання.....	47
Список літератури.....	48
Додаток А. Вихідні дані.....	49

ВСТУП

У методичних вказівках розглядається розв'язання типових задач, що вирішуються на практичних заняттях та в години самостійної роботи з дисциплін «Системи передачі в електровз'язку» та «Багатоканальні системи передачі інформації». Для кожної задачі додаються варіанти завдань, які повинні бути вирішені студентами в години самостійної роботи.

Необхідно відмітити, що в методичних вказівках у вигляді прикладу розв'язання типових задач використовуються вихідні дані варіанта 0.

Задачі, що наведені в методичних вказівках, також можуть бути використані при проведенні поточного контролю знань студентів, модульного контролю та на екзамені.

ЗАВДАННЯ 1. Компандування ІКМ-сигналів

У системі з імпульсно-кодовою модуляцією (ІКМ) і рівномірним квантуванням розмір кожного кроку квантування визначається вимогами до відношення сигнал/шум (ВСШ) по найменшому рівню кодованого сигналу. Великі значення сигналу кодуються з тим же кроком. ВСШ збільшується з ростом амплітуди сигналу. Наприклад, якщо ВСШ = 26 дБ для малих сигналів, а динамічний діапазон дорівнює 30 дБ, то для сигналів з максимальною амплітудою отримуємо ВСШ = 56 дБ. Тому система з ІКМ і рівномірним квантуванням великих сигналів має зайву якість. Більше того, якщо ймовірність появи сигналу з великою амплітудою невелика, то кодовий простір у такій системі використовується неефективно [1].

Процедура дискретизації може стати ефективнішою, якщо крок квантування зробити не однаковим, а зростаючим разом зі значеннями відліків. Одним з таких способів є компандування. Якщо кроки квантування будуть пропорційні значенням відліків, то ВСШ буде постійним для всіх рівнів сигналу. Такий метод дискретизації потребує менше бітів на відлік для забезпечення заданого рівня ВСШ і забезпечує необхідний динамічний діапазон для великих сигналів. При неоднакових кроках

квантування між відліками і відповідними кодовими словами існує нелінійний зв'язок.

Сам принцип компандування полягає в тому, що діапазон значень амплітуди від максимального до мінімального розбивається на сегменти. Ті з них, які відповідають меншим значенням сигналу, квантуються дрібнішими рівнями квантування, а для великих значень вибираються великі рівні квантування, величина яких зростає з номером сегмента (рисунок 1).

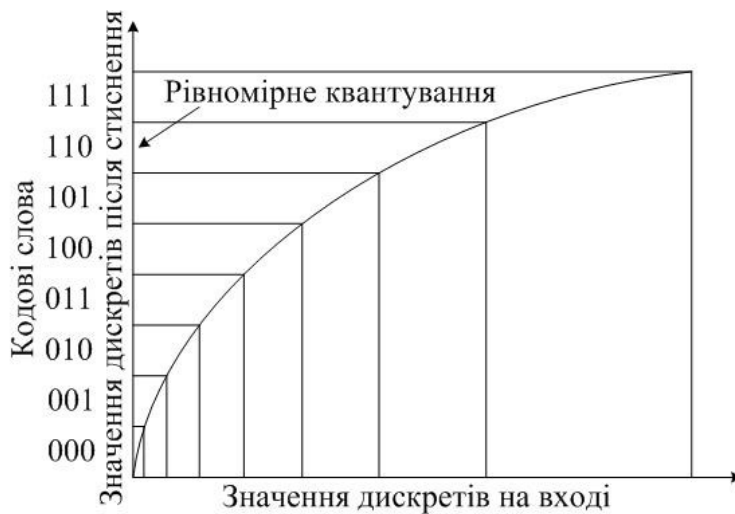


Рисунок 1 – Характеристика стиснення

Використовуються два закони нерівномірного кодування: μ -компандування і А- компандування.

Вони відрізняються невеликими подробицями, які ми розглянемо пізніше. Весь діапазон амплітуд розбивається на 8 сегментів, включаючи нульовий. Вони нумеруються від 0 до 7 або, у двійковій системі, 000 - 111.

У кожному сегменті для вимірювання застосовуються 16 кроків квантування різної величини (у двійковій системі вони нумеруються від 0000 до 1111). При переході від сегмента до сегмента значення величини кроку квантування всередині сегмента збільшується у два рази. Якщо крок квантування в нульовому сегменті прийняти за 2 кванти, то в першому сегменті при квантуванні за А-законом значення кванта 2, у другому сегменті цей крок становить 4 кванти, у третьому – 8 і останньому, восьмому, – 16 квантів. Якщо розглядати величину

кроків квантування з урахуванням зменшення максимального значення сигналу в даному сегменті, точність квантування зростає у два рази. Відповідно зменшується абсолютне значення помилки квантування (нагадаємо, що вона дорівнює половині кроку квантування). Відносна помилка на всіх кроках приблизно однакова і визначається розкидом амплітуд конкретного сигналу в даному сегменті. Зауважимо, що при застосуванні квантування число кроків квантування в кожному сегменті становить 16. Величина кожного кроку різна. Код, який передається в лінію, містить в 1-му розряді знак комбінації значення сигналу (додатне / від'ємне), наступні 3 розряди – номер одного з восьми сегментів і 4 розряди – номер кроку всередині сегмента. Таким чином, замість передачі 13-розрядних комбінацій, утворених на вході при 8192 (включаючи нульовий) кванти, в лінію передаються 8 розрядів. Виходячи зі сказаного вище формат, що проходить у лінію, містить один байт, структура полів якого показана на рисунку 2.

1 біт	3 біти	4 біти
Знак	Номер сегменту	Номер шагу

Рисунок 2. – Структура байта для компандованого відліку ІКМ

Поле знак має значення:

0 – при додатній величині,

1 – при від'ємній.

Кодування інших полів буде розглянуто нижче.

Цю операцію називають стисненням діапазону переданої потужності (компресія, або стиснення). На приймальному кінці ця інформація приблизно відновлюється (розширюється – експандується). Тому пристрій, який здійснює цю операцію (зазвичай це спеціальний діод з нелінійною характеристикою), називають компандер, а сам процес – компандування (компресія – експандування).

У цей час компресія включається до складу аналого-цифрового перетворювача (кодера), а експандування – до складу декодера (рисунок 3).



Рисунок 3 – ІКМ з компандуванням

Для реалізації компандера можуть бути використані різноманітні нелінійні характеристики стиснення – розтягування. Збільшуючи ступінь стиснення, ми збільшуємо динамічний діапазон, зменшуючи тим самим ВСШ для сигналів з більшою амплітудою. На рисунку 4 наведено приклад звукової хвилі до і після стиснення з нелінійним компандуванням.

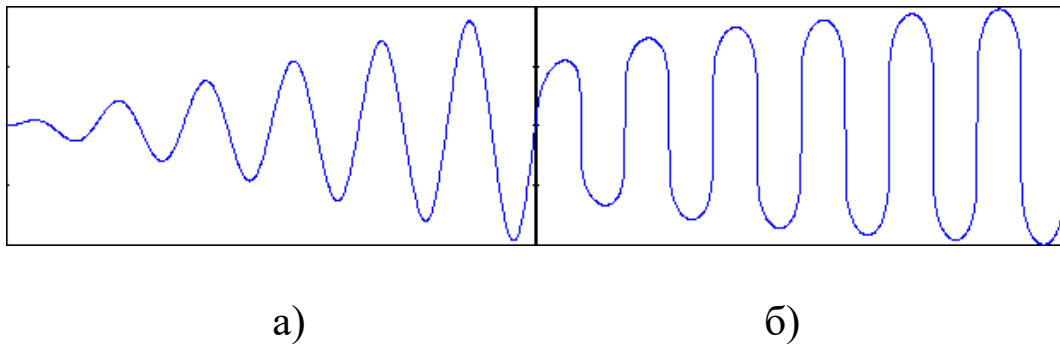


Рисунок 4 – Звукова хвиля до (а) і після (б) стиснення з нелінійним компандуванням

1.1 Компандування за А - законом

Ця характеристика компандування рекомендована для Європи та країн СНД.

Нормована характеристика закону визначається як

$$F_A(x) = \begin{cases} \text{sign}(x) \frac{A|x|}{1 + \ln A}, & 0 \leq |x| \leq 1/A; \\ \text{sign}(x) \frac{1 + \ln |Ax|}{1 + \ln A}, & 1/A \leq |x| \leq 1. \end{cases}$$

Зворотна характеристика (для відновлення сигналу) визначається як

$$F_A^{-1}(y) = \begin{cases} \text{sign}(y) \frac{|y| (1 + \ln A)}{A}, & 0 \leq |y| \leq \frac{1}{1 + \ln A}, \\ \text{sign}(y) \frac{e^{|y|(1 + \ln A)} - 1}{A}, & \frac{1}{1 + \ln A} \leq |y| \leq 1, \end{cases}$$

де $y = F_A(x)$; $A = 87,6$.

При цьому законі весь діапазон амплітуд вимірюється за допомогою 4096 квантів (у цьому випадку вони виконують функції одиниць вимірювання амплітуди) [1].

Умовно (щоб зберегти загальний підхід до двох законів) вважається, що є 16 сегментів (8 від'ємних і 8 додатних).

Згідно з прийнятим правилом компресії, кожен сегмент містить 16 кроків квантування, які разом з номером сегмента дають змогу встановити значення відліку з точністю, що визначається номером сегмента. Нульові сегменти (додатний і від'ємний) розглядаються як один сегмент, що має 31 крок квантування. При обох законах компандування є один крок, який перекриває початок координат. Кодове слово 0 000 0000 позначає значення, що менші +1, а 1 000 0000 позначає значення, що більші -1. Це призводить до числа сегментів 15.

Нульовий і перші два сегменти кожного знака мають один і той же крок квантування (2 кванти) і можуть розглядатися як один сегмент додатного, а інший – від'ємного знака. Тому часто квантування за А-законом називають «13-сегментним компандуванням». Значення інших сегментів і їх нумерація буде залежати тільки від величини амплітуди відліків.

Однак для спрощення алгоритмів кодування використовується подання характеристики компандування у вигляді 16 сегментів. Таким чином, для нумерації сегментів (без урахування знака) потрібно три біти (значення номерів від 000 до 111). Конкретні коди кодування / декодування за А-законом наведені в таблиці 1.

Процес компандування при сучасних параметрах мікросхем може здійснюватися за допомогою постійних запам'ятовуючих пристроїв прямим табличним перетворенням. Однак існують алгоритми, що дають змогу робити інше перетворення шляхом нескладного перерахунку.

Таблиця 1 – Кодування/декодування згідно із А-законом

Кінцеві точки квантування коду сегмента S								Код квантування	Q
000	001	010	011	100	101	110	111		
0	32	64	128	256	512	1024	2048	0000	0
2	34	68	136	272	544	1088	2176	0001	1
4	36	72	144	288	576	1152	2304	0010	2
6	38	76	152	304	608	1216	2432	0011	3
8	40	80	160	320	640	1280	2560	0100	4
10	42	84	168	336	672	1344	2688	0101	5
12	44	88	176	352	704	1408	2816	0110	6
14	46	92	184	368	736	1472	2944	0111	7
16	48	96	192	384	768	1536	3072	1000	8
18	50	100	200	400	800	1600	3200	1001	9
20	52	104	208	416	832	1664	3328	1010	10
22	54	108	216	432	864	1728	3456	1011	11
24	56	112	224	448	896	1792	3584	1100	12
26	58	116	232	464	928	1856	3712	1101	13
28	60	120	240	480	960	1920	3840	1110	14
30	62	124	248	496	992	1984	3968	1111	15
32	64	128	256	512	1024	2048	4096		

1.1.1 Алгоритм 1: Пряме кодування

Для заданого відліку з амплітудою x перший крок процесу кодування амплітуди полягає у визначенні номера сегмента S . Кінцеві точки сегментів при А-законі дорівнюють 32, 64, 128, 256, 512, 1024, 2048 і 4096. Тому для абсолютного значення x ідентифікатор головного сегмента S може бути визначений як

$$x < 32 \cdot 2^S, S = 0, 1, \dots, 7. \quad (1)$$

Після того, як визначено базовий сегмент, що містить значення відліку, повинен бути ідентифікований конкретний інтервал квантування всередині цього сегмента. На першому кроці визначаємо залишок r як різницю між амплітудою вхідного сигналу і нижньою кінцевою точкою сегмента

$$r = \begin{cases} x, & S = 0; \\ x - 16 \cdot 2^S, & S = 1, 2, \dots, 7. \end{cases} \quad (2)$$

Тепер можна визначити Q як номер інтервалу квантування, що містить залишок r .

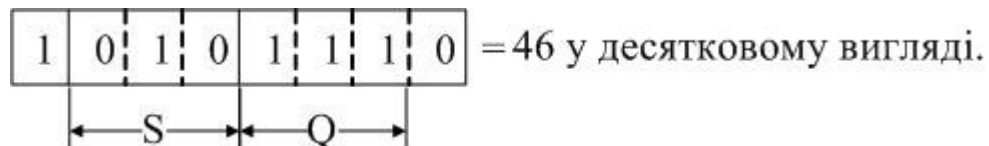
$$r < \begin{cases} 2(Q+1), & S = 0; \\ 2^S(Q+1), & S = 1, 2, \dots, 7. \end{cases} \quad (3)$$

Після того, як визначені S і Q , вони кодуються двійковими кодами з трьома і чотирма бітами відповідно. Об'єднання S і Q дає семибітове слово, яке може бути цілим числом між 0 і 127. По суті, ці цілі числа вказують один з 128 інтервалів квантування для стисненої (компресованої) амплітуди сигналу.

Процес декодування включає присвоєння зазначеної полярності в біті «знака» аналоговому вихідному відліку в середній точці n -го інтервалу квантування, $n = 0, 1, 2, \dots, 127$. Безпосередньо використовуючи значення S і Q , можемо визначити дискретне значення вихідного відліку:

$$y_n < \begin{cases} 2Q+1, & S = 0; \\ 2^S(Q+16.5), & S = 1, 2, \dots, 7. \end{cases} \quad (4)$$

ПРИКЛАД 1. Вхідний відлік з амплітудою, що дорівнює -121 В, формує такий код:



Перший біт коду містить «1», що відображає від'ємний знак вхідної величини. Наступні три (010) визначають величину сегмента S (1), 5 – 8 біти відображають значення номера рівня квантування Q (3). Таким чином, вхідний відлік -121 після процесу компандування набуває значення -46.

На виході декодера утворюється число

$$y_{46} = 2^S(14 + 16.5) = 122\text{В},$$

що відповідно до таблиці 1 є середньою точкою інтервалу квантування від 120 до 124.

1.1.2 Алгоритм 2: Перетворення на основі лінійного коду

Спочатку відлік у десятковому вигляді необхідно перетворити в 12-розрядний двійковий код. Потім за допомогою таблиці 2 перетворити 12-бітовий лінійний код безпосередньо в стиснений за А-законом код. Сегмент числа S може бути визначений як 7 мінус число провідних нулів. Поле даних Q виходить з чотирьох бітів ($wxyz$), що безпосередньо йдуть за ведучою одиницею, за винятком $S = 0$, у цьому випадку поле Q міститься в чотирьох бітах, наступних за сімома провідними нулями. Усі наступні біти ($a-d$) просто ігноруються при генерації в стиснений код.

Таблиця 2 – Кодування за А-законом

Лінійний код												Стиснений код						
0	0	0	0	0	0	0	w	x	y	z	a	0	0	0	w	x	y	z
0	0	0	0	0	0	1	w	x	y	z	a	0	0	1	w	x	y	z
0	0	0	0	0	1	w	x	y	z	a	b	0	1	0	w	x	y	z
0	0	0	0	1	w	x	y	z	a	b	c	0	1	1	w	x	y	z
0	0	0	1	w	x	y	z	a	b	c	d	1	0	0	w	x	y	z
0	0	1	w	x	y	z	a	b	c	d	e	1	0	1	w	x	y	z
0	1	w	x	y	z	a	b	c	d	e	f	1	1	0	w	x	y	z
1	w	x	y	z	a	b	c	d	e	f	g	1	1	1	w	x	y	z

Таблиця 3 служить для генерації лінійного коду безпосередньо зі стисненого коду. Вихідне значення відповідає середній точці інтервалу квантування у відповідності до значень S і Q .

Таблиця 3 – Декодування за А-законом

Стиснений код								Лінійний вихідний код			
0	0	0	w	x	y	z		0	0	0	0
0	0	1	w	x	y	z		0	0	0	0
0	1	0	w	x	y	z		0	0	0	0
0	1	1	w	x	y	z		0	0	0	0
1	0	0	w	x	y	z		0	0	0	1
1	0	1	w	x	y	z		0	0	1	w
1	1	0	w	x	y	z		0	1	w	x

1	1	1	w	x	y	z
---	---	---	---	---	---	---

1	w	x	y
---	---	---	---

У кожній з цих таблиць 12 бітів абсолютного значення лінійного коду стискаються в семибітове абсолютне значення. Відзначимо, що останній значущий біт при кодуванні завжди ігнорується. Тому кодер вимагає розрядності, рівної 11 бітам, якщо всі вихідні значення безпосередньо стискаються. Однак, якщо перед стисненням виконується обробка сигналів (наприклад додавання двох сигналів), то додатковий біт корисний для зменшення загальної помилки квантування.

ПРИКЛАД 2. Подаємо значення вхідного відліку -121В у двійковій формі як 000001111001. У таблиці кодування (таблиця 2, лінійний код) здійснюємо пошук рядка, що містить код з першими бітами 000001 (рядок 3), йому відповідає стиснений код 010wxuz, де $S = 010$ і $wxyz = Q = 1110$. Тому після стиснення отримуємо стиснений код

0	0	1	1	0	0	0	1
---	---	---	---	---	---	---	---

Використовуючи таблицю декодування (таблиця 3), лінійний код на виході може бути визначений як 000001111010, що дорівнює десятковому значенню 122В.

1.2 Компандування за μ -законом

Ця характеристика компандування рекомендована для Північної Америки і Японії.

Нормована характеристика закону визначається як

$$F_{\mu}(x) = \text{sign}(x) \frac{\ln(1 + \mu|x|)}{\ln(1 + \mu)}$$

де x – амплітуда вхідного сигналу ($-1 \leq x \leq 1$); $\text{sign}(x)$ – знак x ; μ – параметр, що характеризує величину стиснення, $\mu = 255$.

Внаслідок математичного характеру μ -закону ІКМ зі стисненням іноді називають логарифмічною ІКМ. Логарифмічна крива стиснення ідеальна в тому сенсі, що кроки квантування, а

отже, і шум квантування пропорційні амплітуді відліку. Інвертована характеристика для розширення сигналів у компандер з μ -законом визначається як

$$F_{\mu}^{-1}(y) = \text{sign}(y) \frac{1}{\mu} \left[\left(1 + \mu \right)^{|y|} - 1 \right],$$

де $y = F_{\mu}(x)$ - стиснене значення сигналу ($-1 \leq y \leq 1$); $\text{sign}(y)$ – знак y ; μ – параметр компандування [1].

Цей закон відзначається великим числом дискрет для кодування сигналу. Їх 8159, що дає змогу більш точно кодувати слабкі сигнали. За статистикою таких більше, ніж сигналів з великою амплітудою. Ця обставина підвищує якість мови (але, як показала практика, незначно). При цьому кроки квантування змінюються в кожному сегменті і рівні 1, 2, 4, 8, 16, 32, 64, 128, 256. Нижче наводиться таблиця кодування (таблиця 4).

1.2.1 Алгоритм 1: Пряме кодування (таблиця 1)

Для заданого відліку з амплітудою x перший крок процесу кодування амплітуди полягає у визначенні номера сегмента S . Базові сегменти ідентифікуються за їх кінцевими точками 31, 95, 223, 479, 991, 2015, 4063 і 8159. Таким чином, S визначається як мінімальна кінцева точка, яка більша за відлік x . Тут S дорівнює

$$x < 64 \cdot 2^S - 33, S = 0, 1, \dots, 7.$$

Таблиця 4 – Кодування/декодування згідно із μ -законом

Кінцеві точки квантування коду сегмента S								Код квантування	Q
000	001	010	011	100	101	110	111		
0	31	95	223	479	991	2015	4063	0000	0
1	35	103	239	511	1055	2143	4319	0001	1
3	39	111	255	543	1119	2271	4575	0010	2
5	43	119	271	575	1183	2399	4831	0011	3
7	47	127	287	607	1247	2527	5087	0100	4
9	51	135	303	639	1311	2655	5343	0101	5
11	55	143	319	671	1375	2783	5599	0110	6

13	59	151	335	703	1439	2911	5855	0111	7
15	63	159	351	735	1503	3039	6111	1000	8
17	67	167	367	767	1567	3167	6367	1001	9
19	71	175	383	799	1631	3295	6623	1010	10
21	75	183	399	831	1695	3423	6879	1011	11
23	79	191	415	863	1759	3551	7135	1100	12
25	83	199	431	895	1823	3679	7391	1101	13
27	87	207	447	927	1887	3807	7647	1110	14
29	91	215	463	959	1951	3935	7903	1111	15
31	95	223	479	991	2015	4063	8159		

Відліки масштабуються за максимальним значенням 8159. Від'ємні відліки кодується у форматі «знак - абсолютне значення» з піковим бітом, рівним 1. При реальній передачі коди інвертуються, для того щоб збільшити щільність одиниць у разі, коли кодується сигнал з малою амплітудою. Аналоговий відлік декодується як значення, рівне центру інтервалу квантування. Помилка квантування дорівнює різниці між відновленим значенням на виході і вихідним значенням відліку на вході

Інтервал квантування всередині цього сегмента r дорівнює

$$r = \begin{cases} x, & S = 0; \\ x - (32 \cdot 2^S - 33), & S = 1, 2, \dots, 7. \end{cases}$$

Тепер можна визначити Q як номер інтервалу квантування, що містить залишок r .

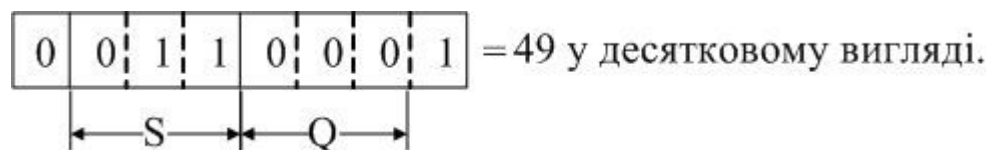
$$r = \begin{cases} 2Q + 1, & S = 0; \\ 2^{S+1}(Q + 1), & S = 1, 2, \dots, 7. \end{cases}$$

Тепер, як і у випадку кодування за А-законом, абсолютне значення, отримане за μ -законом, можна подати як ціле $n = 0, 1, \dots, 127$, утворене шляхом конкатенації трьох бітів S і чотирьох бітів Q . Абсолютне значення на виході y_n може бути виражене як

$$y_n = (2Q + 33)2^S - 33,$$

де n – ціле число, що отримане конкатенацією S та Q .

ПРИКЛАД 3. Вхідний відлік +242 породжує таке кодове слово:



На виході декодера отримуємо

$$y_n = (2 \cdot 1 + 33)2^3 - 33 = 247B,$$

що являє собою середню точку 49-го інтервалу квантування від 239 до 255 В.

1.2.2. Алгоритм 2: Перетворення на основі лінійного коду

Перший алгоритм забезпечує реалізацію кодера за законом μ -255 з ІКМ, який використовує 13-бітове рівномірне кодування і подальше стиснення за допомогою цифрової логіки. Другий алгоритм показує, як реалізувати функцію декодування для першого розширення стисненого коду в 13-бітовий лінійний код, який використовується для генерації вихідних відліків.

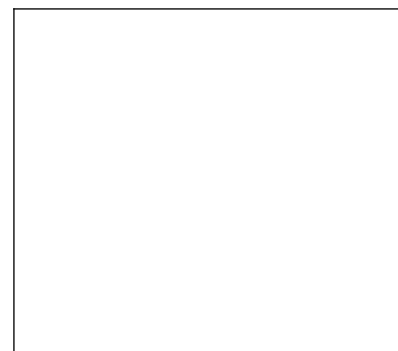
Простота перетворення з лінійного коду в стиснений код більш наочна, якщо лінійний код зміщений додатком 33 до абсолютного значення всіх відліків. Зазначимо, що це зміщення зсуває діапазон 0-8159 до діапазону 33-8192. Процес складання може бути реалізований безпосередньо з аналоговими відліками перед кодуванням або за допомогою цифрової логіки після кодування. У будь-якому випадку основна форма всіх зміщених комбінацій і відповідних стиснених кодів показана в таблиці 5.

З попередньої таблиці можна побачити, що на початку всіх зміщених кодів існує одиниця, яка вказує номер сегмента S . А саме, S дорівнює 7 мінус число нулів, що стоять перед одиницею. Величина Q безпосередньо відображається чотирма бітами (w, x, y, z), що безпосередньо йдуть за першою одиницею. Усі наступні біти ($a-h$) просто ігноруються, коли генерується стиснений код.

Таблиця 5 – Кодування при стисненні за законом μ -255

Зміщені лінійні коди	Стиснені коди
0 0 0 0 0 0 0 1 w x y	

0	0	0	0	0	0	1	w	x	y	z
0	0	0	0	0	1	w	x	y	z	a
0	0	0	0	1	w	x	y	z	a	b
0	0	0	1	w	x	y	z	a	b	c
0	0	1	w	x	y	z	a	b	c	d
0	1	w	x	y	z	a	b	c	d	e
1	w	x	y	z	a	b	c	d	e	f



Для зворотного перетворення використовується таблиця 6, яка показує, як отримати зміщений лінійний код зі стисненого коду.

Таблиця 6 – Декодування при стисненні за законом μ -255

Стиснені коди	Зміщені лінійні коди на виході
0 0 0 w x y z	0 0 0 0 0 0 0 1 w x y z 1
0 0 1 w x y z	0 0 0 0 0 0 1 w x y z 1 0
0 1 0 w x y z	0 0 0 0 0 1 w x y z 1 0 0
0 1 1 w x y z	0 0 0 0 1 w x y z 1 0 0 0
1 0 0 w x y z	0 0 0 1 w x y z 1 0 0 0 0
1 0 1 w x y z	0 0 1 w x y z 1 0 0 0 0 0
1 1 0 w x y z	0 1 w x y z 1 0 0 0 0 0 0
1 1 1 w x y z	1 w x y z 1 0 0 0 0 0 0 0

Обидві ці таблиці показують, що для подання абсолютної величини сигналу використані 13 бітів лінійного коду. Однак при кодуванні за законом μ -255 кодер має амплітудний діапазон, еквівалентний 12 бітам. Різниця виникає тому, що перший інтервал квантування має довжину, рівну 1, у той час як інші інтервали в першому сегменті мають довжину 2. Тому додатковий біт потрібен тільки для визначення першого інтервалу квантування. Зазначимо, що далі останній значущий біт у таблиці не несе інформації, але включений тільки для того, щоб полегшити отримання цілих значень. Зокрема останній значущий біт у таблиці кодування повністю ігнорується, коли розпізнається стиснений код (припускаємо, що зміщення додається до аналогового відліку). Крім того, останній значущий біт у вихідному коді повністю визначає номер сегмента S .

ПРИКЛАД 4. Вхідне кодове слово +242В зміщене до числа 275. Двійкове 13-розрядне подання цього числа 0000100010011 (зміщений лінійний код).

З кодової таблиці $S = 011$ і $wxyz = 0001$, тоді стиснений код має вигляд

0	0	1	1	0	0	0	1
---	---	---	---	---	---	---	---

Використовуючи таблицю стиснених кодів, отримуємо наступний зміщений лінійний код 0000100011000 (зміщений лінійний код на виході).

Значення попереднього коду у двійковому вигляді – 280В, це відповідає незміщеному сигналу +247В.

ІНДИВІДУАЛЬНЕ ЗАВДАННЯ. У відповідності до варіанта завдання (додаток А) здійснити компандування вхідних відліків за А- і μ -законами.

Контрольні питання

- 1 Призначення компандування ІКМ-сигналів. Переваги, недоліки.
- 2 Поясніть основний принцип компандування.
- 3 Скільки бітів складає формат кодового слова ІКМ за законами А- і μ -компандування?
- 4 У чому різниця перетворення сигналу на основі прямого та лінійного кодів? Поясніть на прикладі.
- 5 Визначте ВСШ компандованого за варіантом сигналу.

ЗАВДАННЯ 2. Цифрове лінійне кодування сигналів

При цифровому кодуванні дискретної інформації застосовують потенційні й імпульсні коди.

У *потенційних кодах* для подання логічних одиниць і нулів використовується тільки значення потенціалу сигналу, а його перепади, що формують закінчені імпульси, до уваги не беруться.

Імпульсні коди дають змогу подати двійкові дані або імпульсами певної полярності, або частиною імпульсу – перепадом потенціалу певного напрямку.

2.1 Вимоги до методів цифрового кодування

При використанні відеоімпульсів для передачі дискретної інформації необхідно вибрати такий спосіб кодування, який одночасно досягав би декількох цілей:

- мав при одній і тій же бітовій швидкості найменшу ширину спектра результуючого сигналу;
- забезпечував синхронізацію між передавачем і приймачем;
- мав здатність розпізнавати помилки;
- мав низьку вартість реалізації.

Більш *вузький спектр* сигналів дає змогу на одній і тій самій лінії (з однією і тією самою смугою пропускання) домагатися більш високої швидкості передачі даних. Крім того, часто до спектра сигналу ставиться вимога відсутності постійної складової, тобто наявності постійного струму між передавачем і приймачем.

Синхронізація передавача і приймача потрібна для того, щоб приймач точно знав, у який момент часу необхідно зчитувати нову інформацію з лінії зв'язку. Ця проблема в мережах вирішується складніше, ніж при обміні даними між близько розташованими пристроями, наприклад, між блоками всередині комп'ютера або ж між комп'ютером і принтером. На невеликих відстанях добре працює схема, основана на окремій синхронізації (рисунок 5), так, що інформація зчитується з лінії тільки в момент приходу тактового імпульсу. Використання цієї схеми у мережах викликає труднощі через неоднорідність характеристик провідників у кабелях. На великих відстанях нерівномірність швидкості поширення сигналу може призвести до того, що тактовий імпульс прийде настільки пізніше чи раніше відповідного сигналу даних, що біт даних буде пропущений або зчитаний повторно. Іншою причиною, через яку в мережах відмовляються від використання окремих ліній зв'язку синхронізації, є економія провідників у дорогих кабелях.

Тому в мережах застосовуються так звані самосинхронізовані коди, сигнали яких несуть для передавача вказівки про те, в який момент часу потрібно здійснювати розпізнавання чергового біта

(або декількох бітів, якщо код орієнтований більш ніж на два стани сигналу). Будь-який різкий перепад сигналу – так званий фронт – може служити хорошою вказівкою для синхронізації приймача з передавачем.

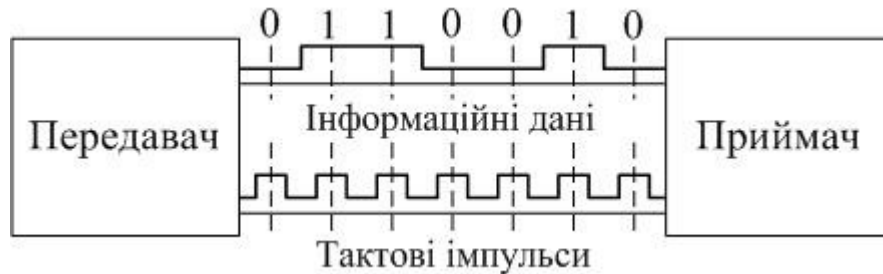


Рисунок 5 – Синхронізація передавача та приймача на невеликих відстанях

Розпізнавання і корекцію помилкових даних складно здійснити засобами фізичного рівня, тому найчастіше цю роботу беруть на себе протоколи, що лежать вище: канальний, мережний, транспортний або прикладний. З іншого боку, розпізнавання помилок на фізичному рівні економить час, так як приймач не чекає повного розміщення кадра в буфер, а відбраковує його відразу при розпізнаванні помилкових бітів усередині кадра.

Вимоги, що ставляться до методів кодування, є взаємно суперечливими, тому кожен з розглянутих нижче популярних методів цифрового кодування має свої переваги і свої недоліки порівняно з іншими [2].

2.2 Методи цифрового кодування

Потенційний код без повернення до нуля (*Non Return to Zero, NRZ*) – при передачі послідовності одиниць сигнал не повертається до нуля протягом такту (рисунок 6).

Метод NRZ простий у реалізації, має гарну розпізнаваність помилок (через два потенціали, що різко відрізняються), але не має властивості самосинхронізації. Другим недоліком є наявність низькочастотної складової, яка наближається до постійного значення сигналу при передачі довгих послідовностей одиниць

або нулів. Тому в мережах код NRZ в основному використовується у вигляді різних його модифікацій.

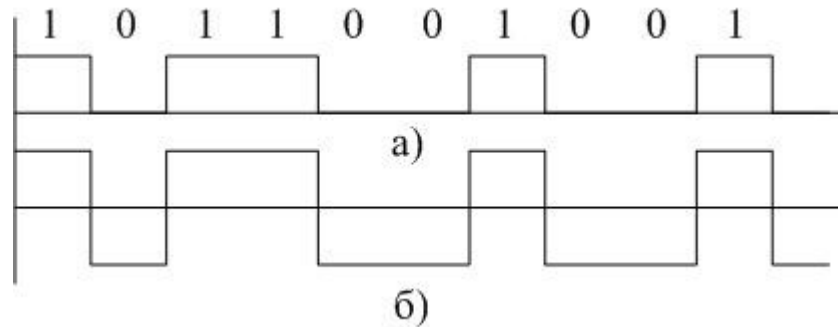


Рисунок 6 – Уніполярний (а) та полярний (б) коди передачі NRZ

Однією з модифікацій методу NRZ є **метод біполярного кодування з альтернативною інверсією** (*Alternate Mark Inversion, AMI*) (рисунок 7). У цьому методі застосовуються три рівні потенціалу – від’ємний, нульовий і додатний. Для кодування логічного нуля використовується нульовий потенціал, а логічна одиниця кодується або додатним потенціалом, або від’ємним, при цьому потенціал кожної нової одиниці протилежний потенціалу попередньої.

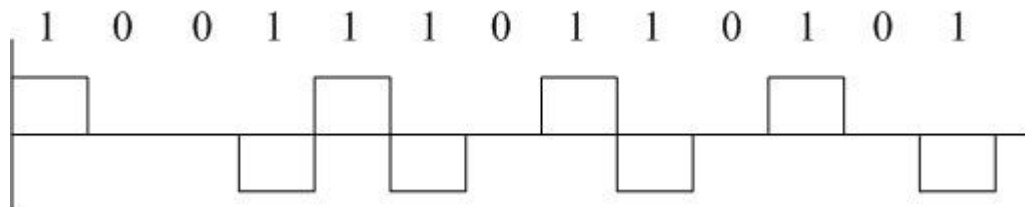


Рисунок 7 – Біполярний код АМІ

Біполярне кодування вирішує проблему усунення постійної складової, проте тільки при передачі довгих послідовностей одиниць, довгі ж послідовності нулів також небезпечні для коду АМІ, як і для коду NRZ, – сигнал вироджується в постійний потенціал нульової амплітуди.

Код АМІ дає також деякі можливості щодо розпізнання помилкових сигналів. Так, порушення строгого чергування полярності сигналів говорить про помилковий імпульс або зникнення в лінії коректного імпульсу. Сигнал з некоректною полярністю називається забороненим сигналом (signal violation).

При цьому код АМІ надлишковий, а отже, неефективний, тому що додатковий рівень вимагає збільшення потужності передавача приблизно на 3 дБ для забезпечення тієї ж достовірності приймання бітів на лінії. Це є загальним недоліком кодів з декількома станами сигналу порівняно з кодами, які розрізняють тільки два стани.

У локальних мережах до недавнього часу найпоширенішим методом кодування був так званий **манчестерський код** (рисунок 8). Він застосовується в технологіях Ethernet і Token Ring.

При манчестерському кодуванні кожен такт ділиться на дві частини. Інформація кодується перепадами потенціалу, що відбуваються всередині кожного такту. Одиниця кодується перепадом від низького рівня сигналу до високого, а нуль – зворотним перепадом. На початку кожного такту може відбуватися службовий перепад сигналу, якщо потрібно подати кілька одиниць або нулів підряд. Так як сигнал змінюється принаймні один раз за такт передачі одного біта даних, то манчестерський код має гарні самосінхронізовані властивості.

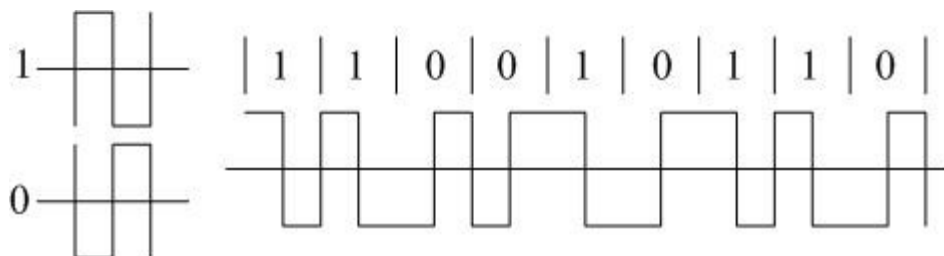


Рисунок 8 – Абсолютний двофазний (манчестерський) код передачі

Смуга пропускання манчестерського коду вужча, ніж біполярного імпульсного. У нього також немає постійної складової, а основна гармоніка в гіршому випадку (при передачі послідовності одиниць або нулів) має частоту N Гц, а в кращому (при передачі одиниць і нулів, що чергуються) вона дорівнює $N / 2$ Гц, як і у кодів АМІ або NRZ. У середньому ширина смуги манчестерського коду в півтора рази вужча, ніж біполярного імпульсного коду, а основна гармоніка коливається поблизу значення $3N / 4$. Манчестерський код має ще одну перевагу перед

біполярним імпульсним кодом. В останньому для передачі даних використовуються три рівні сигналу, а в манчестерському – два.

Частотний спектр манчестерського коду зображений на рисунку 9, на якому його можна порівняти зі спектром сигналу NRZ. Зазначимо, що перший спектральний нуль манчестерського сигналу перебуває в точці $2 / T$.



Рисунок 9 – Спектральна щільність потужності манчестерського коду

Отже, додаткові переходи й усунення дрейфу постійної складової реалізуються за рахунок використання більш високих частот. Однак у порівнянні з трирівневими біполярними кодами манчестерський код має більш низьку ймовірність помилки при тому ж відношенні сигнал / шум.

Існують коди, схожі на АМІ, але тільки з двома рівнями сигналу. При передачі нуля вони передають потенціал, який був установлений у попередньому такті (тобто не змінює його), а при передачі одиниці потенціал інвертується на протилежний. Такий тип кодування називається **диференційним**. Прикладом диференційного коду є код, що має назву потенційний код з інверсією при одиниці (*Non Return to Zero with ones Inverted, NRZI*) (рисунку 10). Цей код зручний у тих випадках, коли використання третього рівня сигналу дуже небажане, наприклад в оптичних кабелях, де стійко розпізнаються два стани сигналу – світло і темрява.

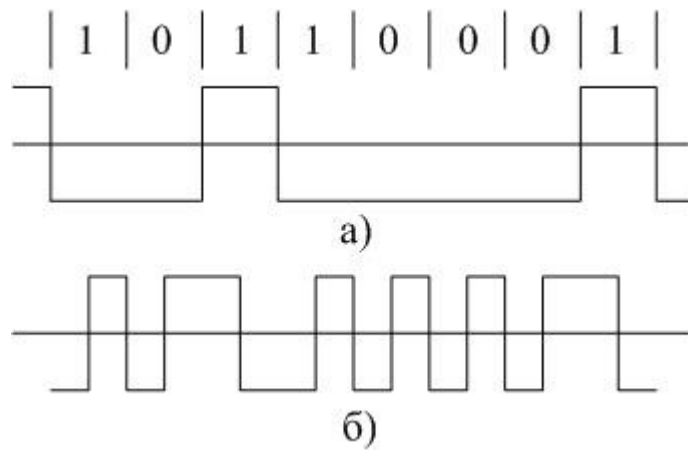


Рисунок 10 – Сигнали, що були отримані при диференційному кодуванні: а) код NRZ; б) манчестерський код

Для поліпшення потенційних кодів, подібних AMI і NRZI, використовуються два методи. Перший метод оснований на додаванні у вихідний код надлишкових бітів, що містять логічні одиниці. Очевидно, що в цьому випадку довгі послідовності нулів перериваються і код стає самосинхронізуючим для будь-яких переданих даних. Зникає також постійна складова, а отже, ще більш звужується спектр сигналу. Але цей метод знижує корисну пропускну здатність лінії, так як надлишкові одиниці користувальницької інформації не несуть. Інший метод оснований на попередньому «перемішуванні» вихідної інформації таким чином, щоб імовірність появи одиниць і нулів на лінії ставала близькою. Пристрої, або блоки, що виконують таку операцію, називаються скремблерами (scramble – звалище). При скремблюванні використовується відомий алгоритм, тому приймач, отримавши двійкові дані, передає їх на дескремблер, який відновлює вихідну послідовність бітів. Надлишкові біти при цьому по лінії не передаються. Обидва методи належать до логічного, а не фізичного кодування, так як форму сигналів на лінії вони не визначають.

Розроблено ряд кодів передачі, подібних до описаного вище абсолютного двофазного коду, зокрема, один з них **код з інверсією струмових посилок** (*Coded Mark Inversion, CMI*) (рисунок 11). Код CMI перетворює одиниці (струмові посилки), як і в коді NRZ, до рівня, протилежного попередньому, а нулі як

напівцикл прямокутного сигналу в певній фазі. У сигналі немає енергії постійного струму і є достатня кількість переходів, як і при двофазному коді. Крім того, немає неоднозначності розпізнавання одиниць і нулів. Усунення неоднозначності насправді призводить до значного недоліку коду: його чутливість до помилок на 3 дБ гірше, ніж чутливість двофазного коду. СМІ є специфікованою кодом інтерфейсу для четвертого рівня мультиплексованих сигналів МСЕ-Т (Е4) на швидкості 139,264 Мбіт/с. СМІ також використовується в електричних інтерфейсах сигналів SONET STS-3с.

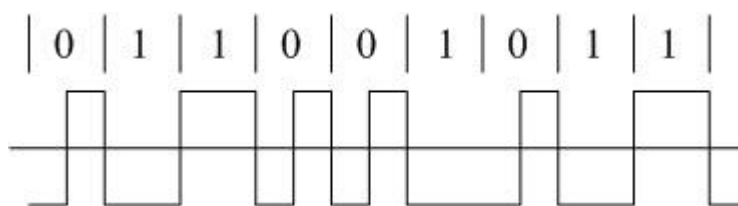


Рисунок 11 – Код з інверсією струмових посилок (СМІ)

Так як біполярний код використовує трійчастий кодовий простір для передачі двійкових даних, їх застосування не дає тієї високої здатності передачі, яка можлива при більш ефективному використанні кодового простору. Наприклад, восьмизначний трійчастий код здатний представляти $3^8 = 6561$ різних кодів, тоді як вісім бітів двійкових даних дають тільки $2^8 = 256$ різних кодів. Описані вище коди передачі не реалізують переваги більшого інформаційного змісту при використанні трійчастих кодів; їх вибирають виходячи з вимог синхронізації і спектральних властивостей.

Процедура **трійчастого кодування** складається з відображення послідовних груп з чотирьох бітів у три трійчасті цифри (4В3Т). Так як двійкові слова з чотирьох бітів потребують тільки 16 з 27 можливих тризначних трійчастих кодових слів, існує значна гнучкість у виборі трійчастих кодів. У таблиці 7 наведений один з можливих способів перетворення.

Таблиця 7 – Таблиця перетворення в код 4В3Т

Двійкове слово	Трійчасте слово (накопичений дисбаланс)		
	-	0	+
0000	- - -		+++

0001	-- 0		++ 0
0010	- 0 -		+ 0 +
0011	0 --		0 ++
0100	-- +		++ -
0101	- + -		+ - +
0110	+ - -		- + +
0111	- 0 0		+ 0 0
1000	0 - 0		0 + 0
1001	0 0 -		0 0 +
1010		0 + -	
1011		0 - +	
1100		+ 0 -	
1101		- 0 +	
1110		+ - 0	
1111		- + 0	

Трійчасті слова в середньому стовпці збалансовані за постійною складовою. Кодові слова з першого і третього стовпців для підтримки балансу постійної складової вибираються поперемінно. Якщо було передано більше додатних імпульсів, ніж від'ємних, то вибирається перший стовбець. Коли дисбаланс між додатними і від'ємними імпульсами змінюється на зворотний, вибирається третій стовбець. Зазначимо, що кодове слово, що складається з нулів, не використовується. Отже, підтримується значна частка синхронізуючих складових. Однак висока інформаційна ефективність принесена в жертву можливості контролю якості, крім того, потрібна кадрова синхронізація для визначення меж трійок.

На рисунку 12 показаний потенційний код з чотирма рівнями сигналу для кодування даних. Це **багаторівневий код 2В1Q**, назва якого відображає його суть – кожні два біти (2В) передаються за один такт сигналом, що має чотири стани (1Q). Парі бітів 00 відповідає потенціал - 2,5 В (-3), парі бітів 01 відповідає потенціал -0,833 В (-1), парі 11 - потенціал +0,833 В (+1), а парі 10 - потенціал +2,5 В (+3). При цьому способі кодування потрібні додаткові заходи для боротьби з довгими послідовностями однакових пар бітів, так як при цьому сигнал перетворюється в постійну складову. При випадковому чергуванні бітів спектр сигналу у два рази вузьчий, ніж у коді NRZ, так як при тій же бітовій швидкості тривалість такту

збільшується у два рази. Таким чином, за допомогою коду 2B1Q можна по одній і тій самій лінії передавати дані у два рази швидше, ніж за допомогою коду AMI або NRZI. Однак для його реалізації потужність передавача повинна бути вища, щоб чотири рівні чітко розрізнялися приймачем на тлі завад.

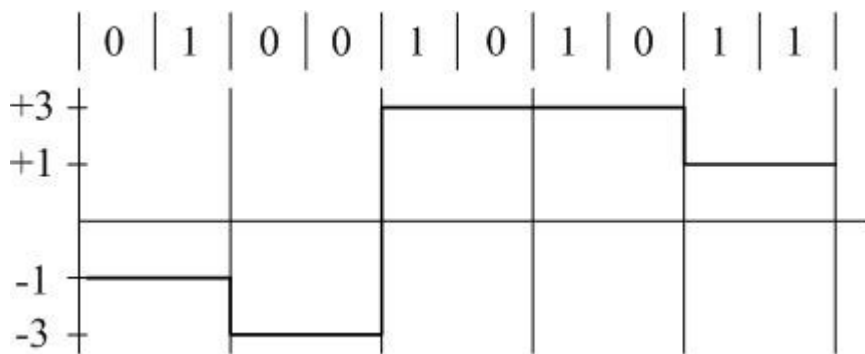


Рисунок 12 – Багаторівневий (чотирирівневий) код (2B1Q)

Восьмирівневий потенційний код, при якому досягається швидкість 3 біти на сигнальний інтервал, зображений на рисунку 13.

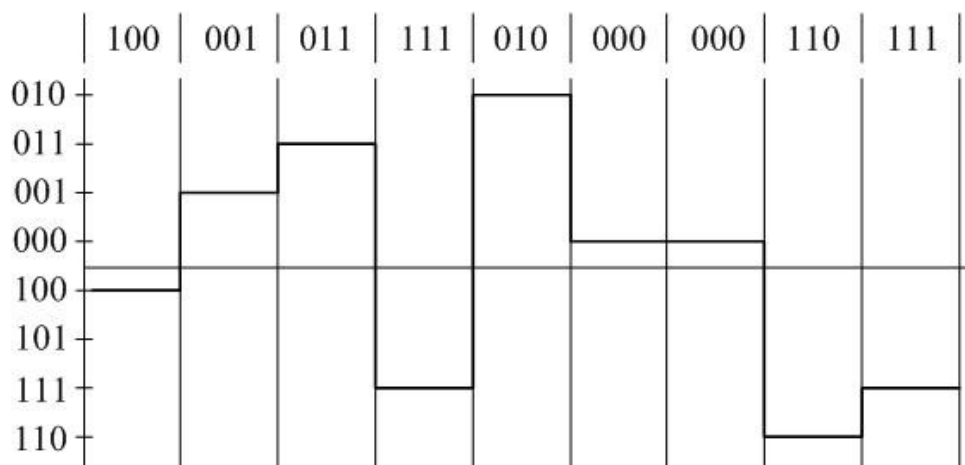


Рисунок 13 – Багаторівневий (восьмирівневий) код (3B1Q)

2.3 Логічне кодування

Логічне кодування використовується для поліпшення потенційних кодів типу AMI, NRZI або 2Q1B. Логічне кодування повинно замінити довгі послідовності бітів, що призводять до постійного потенціалу, вкрапленнями одиниць. Як уже

зазначалося вище, для логічного кодування характерні два методи – надлишкові коди і скремблювання.

Надлишкові коди основані на розбитті вихідної послідовності бітів на порції, які часто називають символами. Потім кожен вихідний символ замінюється на новий, який має більшу кількість бітів, ніж вихідний. Наприклад, логічний код 4В/5В, використовуваний у технологіях FDDI і Fast Ethernet, замінює вихідні символи довжиною в 4 біти на символи довжиною в 5 бітів. Так як результуючі символи містять надлишкові біти, то загальна кількість бітових комбінацій у них більша, ніж у вихідних. Так, у кодi 4В/5В результуючі символи можуть містити 32 бітових комбінації, у той час як вихідні символи – тільки 16. Тому в результуючому кодi можна відібрати 16 таких комбінацій, які не містять великої кількості нулів, а інші вважати забороненими кодами (code violation). Крім усунення постійної складової і надання коду властивості самосинхронізації, надлишкові коди дають змогу приймачу розпізнавати спотворені біти. Якщо приймач приймає заборонений код, отже, на лінії відбулося спотворення сигналу. Відповідність вихідних і результуючих кодів 4В/5В подано в таблиці 8.

Таблиця 8 – Відповідність вихідних і результуючих кодів 4В/5В

Вихідний код	Результуючий код	Вихідний код	Результуючий код
0000	11110	1000	10010
0001	01001	1001	10011
0010	10100	1010	10110
0011	10101	1011	10111
0100	01010	1100	11010
0101	01011	1101	11011
0110	01110	1110	11100
0111	01111	1111	11101

Код 4В/5В потім передається по лінії за допомогою фізичного кодування по одному з методів потенційного кодування, чутливому тільки до довгих послідовностей нулів. Літера В у назві коду означає, що елементарний сигнал має 2 стани – від англійського binary – двійковий. Є також коди з

трьома станами сигналу, наприклад, у кодi 8В/6Т для кодування 8 бітiв вихiдної iнформації використовується код з 6 сигналiв, кожен з яких має три стани. Надмiрність коду 8В/6Т вища, ніж коду 4В/5В, так як на 256 вихiдних кодiв припадає $3^6 = 729$ результуючих символiв.

Використання таблиці перекодування є дуже простою операцією, тому цей підхід не ускладнює мережні адаптери й iнтерфейсні блоки комутаторiв i маршрутизаторiв.

Для забезпечення заданої пропускнуї здатності лінії передавач, що використовує надлишковий код, повинен працювати з підвищеною тактовою частотою. Так, для передачі кодiв 4В/5В зі швидкістю 100 Мбіт /с передавач повинен працювати з тактовою частотою 125 МГц. При цьому спектр сигналу на лінії розширюється порівняно з випадком, коли по лінії передається чистий, ненадлишковий код. Проте спектр надлишкового потенційного коду виявляється вужче спектра манчестерського коду, що виправдовує додатковий етап логічного кодування, а також роботу приймача i передавача на підвищеній тактовій частоті.

Перемішування даних скремблером перед передачею їх у лінію за допомогою потенційного коду є iншим способом логічного кодування.

Методи скремблювання полягають у побiтному обчисленні результуючого коду на підставі бітiв вихiдного коду та отриманих у попередніх тактах бітiв результуючого коду. Наприклад, скремблер може реалізовувати таке співвідношення:

$$V_i = A_i \oplus V_{i-3} \oplus V_{i-5},$$

де V_i – двійкова цифра результуючого коду, отримана на i -му такті роботи скремблера; A_i – двійкова цифра вихiдного коду, що надходить на i -му такті на вхід скремблера; V_{i-3} i V_{i-5} – двійкові цифри результуючого коду, отримані на попередніх тактах роботи скремблера відповідно на 3 i на 5 тактiв раніше поточного такту; \oplus – операція виключає АБО (додавання за модулем 2).

Наприклад, для вихiдної послiдовності 110110000001 скремблер дасть такий результуючий код:

$B_1 = A_1 = 1$ (перші три цифри результуючого коду будуть збігатися з вихідним, бо ще немає потрібних попередніх цифр)

$$B_2 = A_2 = 1$$

$$B_3 = A_3 = 0$$

$$B_4 = A_4 \oplus B_1 = 1 \oplus 1 = 0$$

$$B_5 = A_5 \oplus B_2 = 1 \oplus 1 = 0$$

$$B_6 = A_6 \oplus B_3 \oplus B_1 = 0 \oplus 0 \oplus 1 = 1$$

$$B_7 = A_7 \oplus B_4 \oplus B_2 = 0 \oplus 0 \oplus 1 = 1$$

$$B_8 = A_8 \oplus B_5 \oplus B_3 = 0 \oplus 0 \oplus 0 = 0$$

$$B_9 = A_9 \oplus B_6 \oplus B_4 = 0 \oplus 1 \oplus 0 = 1$$

$$B_{10} = A_{10} \oplus B_7 \oplus B_5 = 0 \oplus 1 \oplus 0 = 1$$

$$B_{11} = A_{11} \oplus B_8 \oplus B_6 = 0 \oplus 0 \oplus 1 = 1$$

$$B_{12} = A_{12} \oplus B_9 \oplus B_7 = 1 \oplus 1 \oplus 1 = 1$$

Таким чином, на виході скремблера з'явиться послідовність 110001101111, у якій немає послідовності з шести нулів, наявної у вихідному коді.

Після отримання результуючої послідовності приймач передає її дескремблеру, який відновлює вихідну послідовність на підставі зворотного співвідношення:

$$C_i = B_i \oplus B_{i-3} \oplus B_{i-5} = (A_i \oplus B_{i-3} \oplus B_{i-5}) \oplus B_{i-3} \oplus B_{i-5} = A_i.$$

Різні алгоритми скремблювання відрізняються кількістю доданків, що дають цифру результуючого коду, і зсувом між доданками. Так, у мережах ISDN при передачі даних від мережі до абонента використовується перетворення зі зсувами в 5-ї і 23-ї позицій, а при передачі даних від абонента в мережу – зі зсувами 18-ї і 23-ї позицій.

Існують і більш прості методи боротьби з послідовностями одиниць, також зараховують до класу скремблювання.

Для поліпшення коду АМІ використовуються два методи, основані на штучному перетворенні послідовності нулів забороненими символами.

Біполярний код із заміною N нулів (*Binary JV-Zero Substitution, BNZS*) – це вдосконалений біполярний код, у якому всі послідовності з N нулів замінюються спеціальним кодом довжини N, що містить кілька імпульсів, які спеціально

викликають порушення біполярності. Таким чином, кількість одиниць збільшується, а початкові дані отримуються розпізнаванням кодів з порушеною біполярністю і заміною їх у приймальному терміналі N нулями.

У коді V3ZS (рисунок 14) кожна послідовність з трьох нулів у вихідних даних замінюється комбінацією або 00V, або V0V. Комбінація 00V складається з двобітового інтервалу без імпульсів (00) і наступного за ним імпульсу з порушенням біполярності V. Комбінація V0V складається з одного імпульсу з біполярним чергуванням (V), наступного за ним інтервалу без імпульсу (0) і завершується імпульсом з порушенням біполярності (V). При кожному заміщенні порушення біполярності виникає як мінімум в одному біті з трьох замісних спеціальним кодом, що дає змогу виявити факт заміщення.

Рішення про те, яким із двох кодів (00V або V0V) замінити нулі, приймається таким чином, щоб число непорушених імпульсів (V), що містяться між імпульсами з порушеннями (V), було непарним. Отже, якщо з моменту останнього заміщення було передано непарне число одиниць, то для заміщення трьох нулів вибирається код 00V. Якщо ж число переданих одиниць парне – V0V.

Отже, всі спеціально зроблені порушення містять непарну кількість переданих біполярних імпульсів. Крім того, заглушенню біполярності поперемінно змінюють полярність, що запобігає придушення постійної складової. Парна кількість біполярних імпульсів між порушеннями виникає тільки в результаті помилки. Остаточний алгоритм заміни наведено в таблиці 9.

Таблиця 9 – Правила заміщення в коді V3ZS

Полярність попереднього імпульсу	Число біполярних імпульсів (одиниць) з моменту останнього заміщення	
	Непарне	Парне
-	00-	+0+
+	00+	-0-

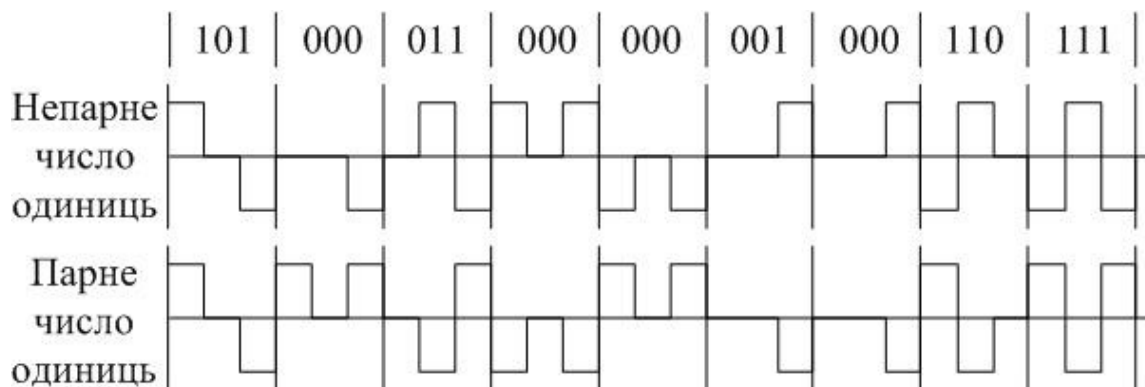


Рисунок 14 – Біполярний код з заміною трьох нулів B3ZS

Також існують біполярні коди із заміною шести (B6ZS) та восьми (B8ZS) нулів, у яких кожна послідовність з шести (восьми) нулів у вихідних даних замінюється комбінацією відповідно до таблиць 10 і 11.

Таблиця 10 – Правила заміщення в коді B6ZS

Полярність імпульсу, безпосередньо передує шести замінним нулям	Заміна
-	0-+0+-
+	0+-0-+

Таблиця 11 – Правила заміщення в коді B8ZS

Полярність попереднього імпульсу	Заміна
-	000-+0+-
+	000+-0-+

Ще один формат кодування із заміною N нулів, що носить назву **коду з високою щільністю** (*High-Density Bipolar, HDB*), рекомендований МСЕ. При реалізації первинного цифрового сигналу в стандарті E1 методом HDB послідовності з чотирьох нулів замінюються послідовностями, що містять порушення біполярності в останньому біті. Так як цей формат кодування запобігає послідовності, що складається більш ніж з трьох нулів, він також відомий як код HDB3. Перетворювальний алгоритм в

основному такий самий, як алгоритм B3ZS, описаний вище і поданий у таблиці 12. Зазначимо, що заміни породжують порушення тільки в четвертій бітовій позиції, а наступні заміни породжують порушення зі змінними полярностями.

Таблиця 12 – Правила заміщення в коді HDB3

Полярність попереднього імпульсу	Число біполярних імпульсів (одиниць) з моменту останнього заміщення	
	Непарне	Парне
-	000-	+00+
+	000+	-00-

Поліпшені потенційні коди мають досить вузьку смугу пропускання для будь-яких послідовностей одиниць і нулів, які зустрічаються в переданих даних. На рисунку 15 наведено спектри сигналів різних кодів, отриманих при передачі довільних даних, у яких різні поєднання нулів і одиниць у вихідному коді рівноймовірні. При побудові графіків спектр усереднювався по всіх можливих наборах вихідних послідовностей. Природно, що результуючі коди можуть мати й інший розподіл нулів та одиниць. З рисунка 15 видно, що потенційний код NRZ має гарний спектр з одним недоліком – у нього є постійна складова. Коди, отримані з потенційного шляхом логічного кодування, мають більш вузький спектр, ніж манчестерський, навіть при підвищеній тактовій частоті (на рисунку 15 спектр коду 4B/5B мав би приблизно збігатися з кодом B8ZS, але він зміщений в область більш високих частот, оскільки його тактова частота підвищена на 1/4 в порівнянні з іншими кодами). Цим пояснюється застосування потенційних надлишкових та скрембльованих кодів у сучасних технологіях, подібних FDDI, сімействі Ethernet, ISDN тощо замість манчестерського і біполярного імпульсного кодування.

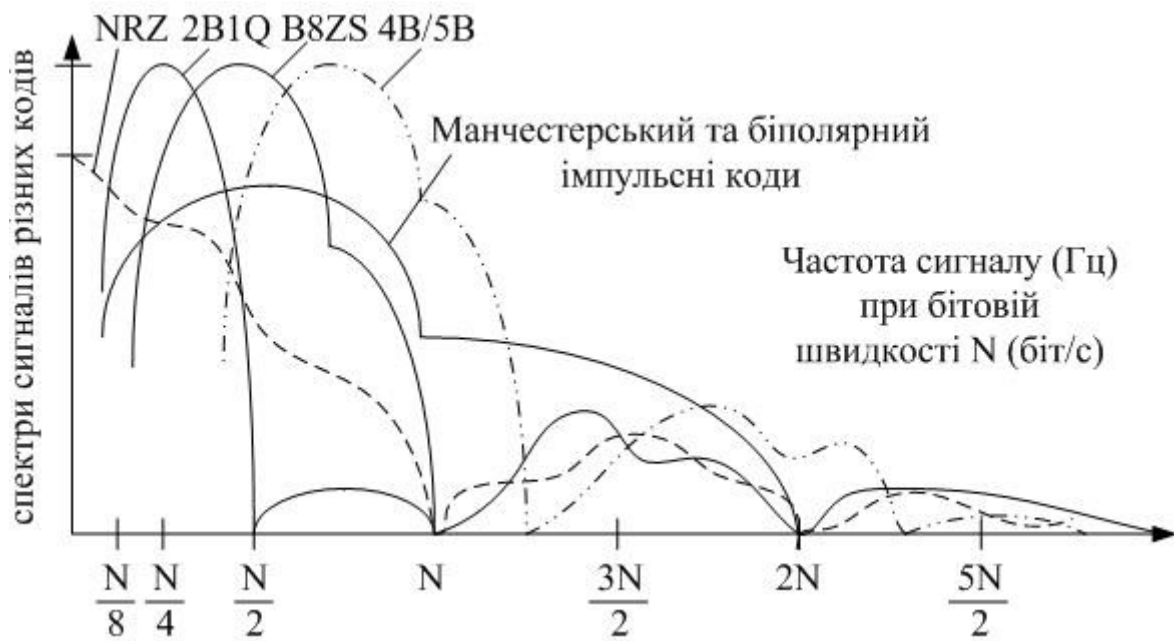


Рисунок 15 – Спектри потенційних та імпульсних кодів

На рисунку 16 наведений приклад кодування цифровими та логічними кодами інформаційної послідовності символів 110000110000001110000001100.

ІНДИВІДУАЛЬНЕ ЗАВДАННЯ. У залежності від варіанта завдання (додаток А) провести кодування заданої інформаційної послідовності, результати кодування зобразити у вигляді, наведеному на рисунку 16.

Контрольні питання

- 1 Призначення лінійного кодування. Вимоги до вибору типу лінійного кодування.
- 2 Наведіть типи цифрового (потенційного) кодування.
- 3 Наведіть типи логічного (імпульсного) кодування.
- 4 Скремблювання. Недоліки, переваги.
- 5 Які типи лінійних кодів використовуються в телефонних мережах, мережах передачі даних та телеметричних мережах?

ЗАВДАННЯ 3. Об'єднання та роз'єднання цифрових потоків

3.1 Способи об'єднання цифрових потоків

При часовому групоутворенні в передавальній частині цифрових систем передачі (ЦСП) здійснюється об'єднання компонентних цифрових потоків, які формуються системами передачі більш низького рівня, у груповий (агрегатний) потік, а в приймальній частині здійснюється роз'єднання агрегатного потоку на компонентні.

Об'єднання (мультиплексування) цифрових потоків може бути синхронним або асинхронним. Якщо задавальні генератори ЦСП, які створюють компонентні потоки, синхронізовані із задавальним генератором ЦСП, який формує агрегатний потік, то проводиться синхронне об'єднання компонентних цифрових потоків. Якщо ж зазначеної взаємної синхронізації немає, то здійснюється асинхронне об'єднання цифрових потоків.

У ЦСП синхронної цифрової ієрархії (SDH) об'єднуються потоки синхронні, а в ЦСП ієрархії PDH – майже синхронні (тактові частоти ЦСП, що формують компонентні потоки, за рахунок нестабільності генераторів незначно, але все-таки відрізняються один від одного).

Процес об'єднання як синхронних, так і асинхронних цифрових потоків передбачає запис компонентних потоків у запам'ятовуючий пристрій (ЗП) відведеного для кожного з потоків блоку сполучення (БС). Запис здійснюється з тактовою частотою компонентного потоку, а зчитування – з частотою, кратною тактовій частоті агрегатного потоку. Мультиплексування компонентних потоків в агрегатний, в принципі, може бути побітовим (порозрядним), побайтовим (поканальним) і поцикловим (посистемним). На практиці використовується побітове об'єднання, що вимагає найменшого обсягу пам'яті ЗП. Спрощена структурна схема обладнання об'єднання цифрових потоків наведено на рисунку 17.

Цифрові потоки від компонентних ЦСП надходять на вхід ЗП відповідних БС. У кожному БС сигнали запису формуються видільниками тактової частоти (ВТЧ), а сигнали зчитування формуються генераторним обладнанням агрегатної системи. Послідовно сигнали компонентних систем, що зчитуються з ЗП з доданими до них службовими сигналами, об'єднуються в загальний агрегатний потік [4].

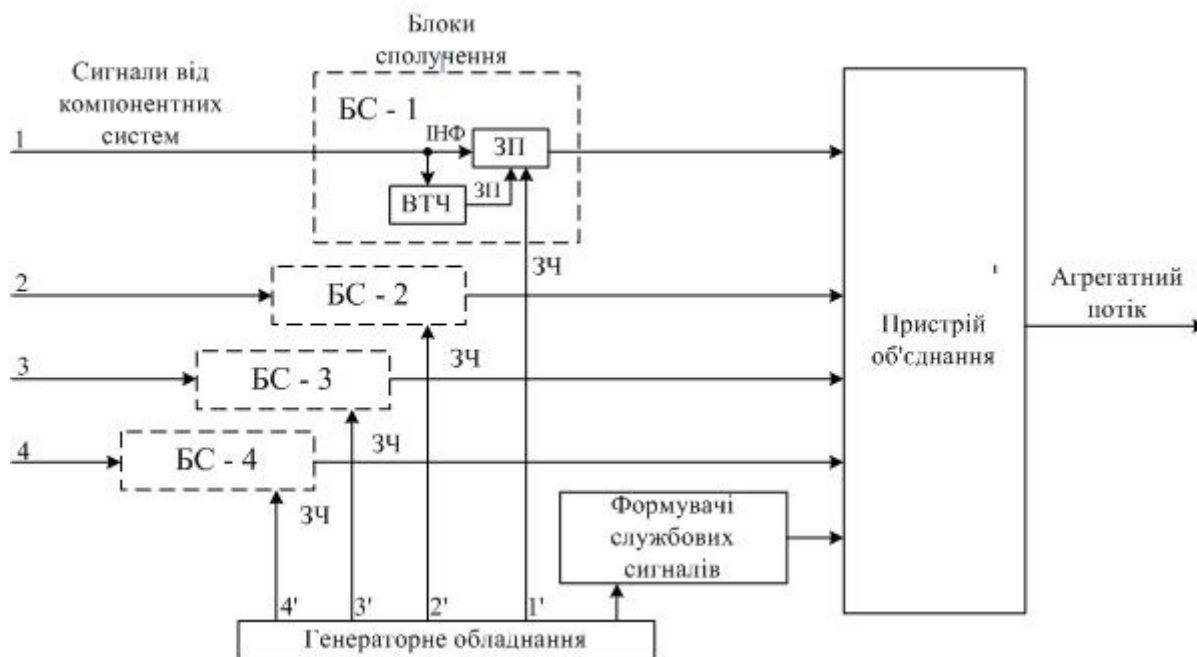


Рисунок 17 – Спрощена структурна схема обладнання об'єднання цифрових потоків

3.2 Об'єднання синхронних цифрових потоків

На рисунку 18 зображені послідовності імпульсів запису і зчитування. У принципі різниця між частотами запису $f_{зп}$ й зчитування $f_{зч}$ $\Delta f = f_{зп} - f_{зч}$ може бути як постійною, так і змінною величиною.

Якщо $f_{зп} = f_{зч}$ ($\Delta T_{зп} = \Delta T_{зч}$), то існуючий часовий інтервал між імпульсами запису і зчитування Δt буде постійною величиною.

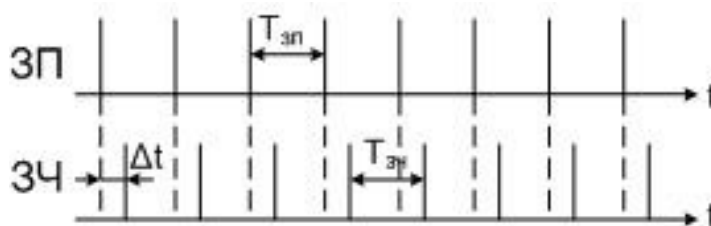


Рисунок 18 – Послідовності імпульсів запису і зчитування

Якщо ж частоти $f_{зп}$ і $f_{зч}$ відрізняються на постійну величину ($\Delta f = f_{зп} - f_{зч} = \text{const}$), то після кожного зчитування цей часовий інтервал Δt між моментами запису і зчитування змінюватиметься.

Якщо $f_{зч} > f_{зп}$ ($T_{зч} < T_{зп}$), то величина Δt зменшується від деякого максимального значення до нуля, а при черговому зчитуванні величина Δt знову виявляється максимальною. На рисунку 19 наведені послідовності сигналів запису і зчитування (для конкретності прийнято $T_{зч} = (3/4) T_{зп}$, відповідно $f_{зч} = (4/3) f_{зп}$).

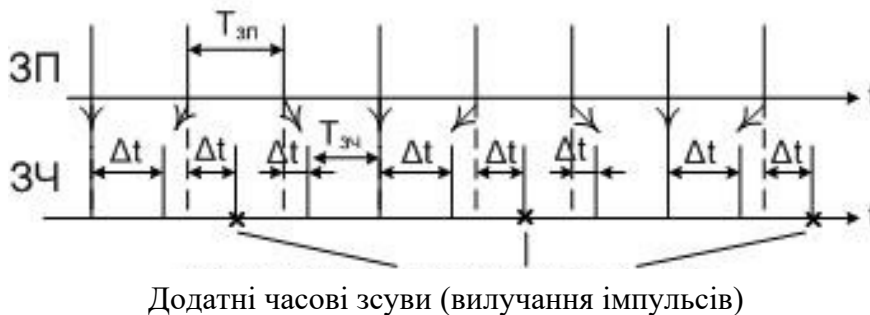
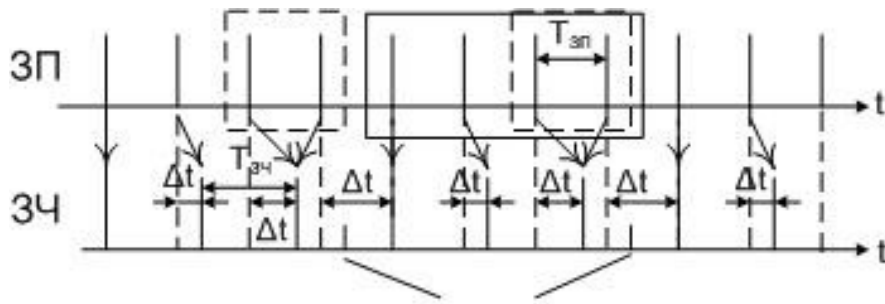


Рисунок 19 – Послідовності сигналів запису і зчитування для $T_{зч} = (3/4) T_{зп}$ ($f_{зч} = (4/3) f_{зп}$)

Оскільки частота імпульсів зчитування $f_{зч}$ перевищує частоту імпульсів запису $f_{зп}$, то ЗП «спустошується» до моменту $\Delta t = 0$, тобто деякі імпульси зчитування виявляються зайвими і їх необхідно вилучити, інакше будуть зчитані «нулі» і передані як інформаційні, яких реально немає у загальному інформаційному потоці. Вивільнені часові позиції (додатні часові зсуви) можна використовувати для передачі службової інформації. На приймальній стороні службові сигнали виділяються за ознакою постійності їх частоти проходження.

Якщо $f_{зч} < f_{зп}$ ($T_{зч} > T_{зп}$), то часовий інтервал Δt між моментами запису і зчитування збільшується до деякого максимального значення, а при черговому зчитуванні він виявляється мінімальним. На рисунку 20 наведено послідовності сигналів запису і зчитування для варіанта $T_{зч} = (4/3) T_{зп}$ ($f_{зч} = (3/4) f_{зп}$).



Від’ємні часові зсуви (повторне зчитування імпульсу)

Рисунок 20 – Послідовності сигналів запису і зчитування для $T_{зч} = (4/3) T_{зп}$ ($f_{зч} = (3/4) f_{зп}$)

Оскільки при $f_{зч} < f_{зп}$ ЗП переповнюється, то виникають моменти, які характеризуються двома інформаційними імпульсами, що припадають на один імпульс зчитування. Для забезпечення нормального процесу об’єднання потоків у цьому випадку необхідно в потік імпульсів зчитування вводити додаткові імпульси (від’ємні часові зсуви). Сталість частоти проходження часових зсувів дає змогу правильно відновлювати інформаційні сигнали, передані в моменти виникнення від’ємних часових зсувів [4].

Частота часових зсувів у зчитаній послідовності імпульсів залежить від співвідношення частот запису і зчитування. Чим більше різняться частоти запису і зчитування, тим частіше формуються часові зсуви. Кількість інформаційних символів R між сусідніми часовими зсувами визначається співвідношенням

$$R = \left\lceil \frac{T_{зч}}{T_{зп} - T_{зч}} \right\rceil = \left\lceil \frac{f_{зп}}{f_{зч} - f_{зп}} \right\rceil, \quad (5)$$

де $\lceil x \rceil$ – ціла частина x з надлишком.

Період виникнення часових зсувів $T_{вз}$ визначається співвідношенням

$$T_{вз} = T_{зч} \left(1 + \left\lceil \frac{T_{зч}}{T_{зп} - T_{зч}} \right\rceil \right) = T_{зч} (1 + R), \quad (6)$$

а частота їх виникнення $f_{вз}$ – співвідношенням

$$f_{вз} = f_{зч} \left(1 + \left\lceil \frac{f_{зп}}{f_{зч} - f_{зп}} \right\rceil \right) = \frac{f_{зч}}{1 + R}. \quad (7)$$

Так, для наведених на рисунках 19 і 20 послідовностей з додатними і від'ємними часовими зсувами значення R , $T_{вз}$ і $f_{вз}$ відповідно дорівнюють

$$R = \left[\frac{T_{зч}}{T_{зн} - T_{зч}} \right] = \left[\frac{3/4 T_{зн}}{T_{зн} - 3/4 T_{зн}} \right] = 3; T_{вз} = 4 T_{зч} \text{ та } f_{вз} = \frac{f_{зч}}{4}$$

для додатних часових зсувів (рисунок 19) та

$$R = \left[\frac{T_{зч}}{T_{зн} - T_{зч}} \right] = \left[\frac{4/3 T_{зн}}{T_{зн} - 4/3 T_{зн}} \right] = -4; T_{вз} = |-4+1| T_{зч} = 3 T_{зч} \text{ та } f_{вз} = \frac{f_{зч}}{3}$$

для від'ємних часових зсувів (рисунок 20).

Таким чином, при об'єднанні синхронних цифрових потоків співвідношення між частотами запису і зчитування незмінні. Тому при об'єднанні цих потоків часові зсуви формуються через певне і строго постійне число інформаційних імпульсів ($R = \text{const}$). Частота і періодичність часових зсувів також незмінні ($T_{вз} = \text{const}$; $f_{вз} = \text{const}$).

Потоки, у яких часові зсуви (СТАФІНГ) формуються строго через визначене і незмінне число інформаційних імпульсів, є однорідними.

3.3 Об'єднання асинхронних цифрових потоків

Цифрові потоки плезіосинхронної ієрархії не синхронні. Тому в реальних умовах відношення $T_{зч}/T_{зн}$ змінюється за рахунок нестабільності частот зчитування та запису. У результаті строга періодичність появи часових зсувів порушується і, як наслідок, виникають неоднорідності, які порушують сталість числа інформаційних імпульсів між сусідніми часовими зсувами.

При цьому значення $\frac{T_{зч}}{T_{зн} - T_{зч}}$ є дробовим числом. Зазначені неоднорідності з'являються з періодичністю, яка визначається співвідношенням

$$\frac{T_{зч}}{T_{зн} - T_{зч}} - R = \pm \frac{n}{L}, \quad (8)$$

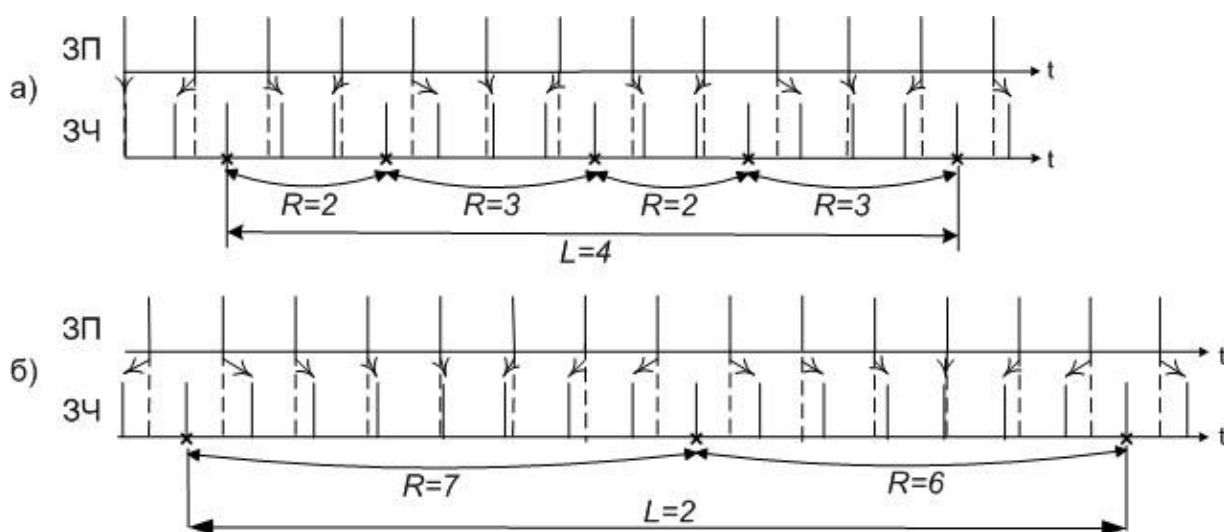
де L – число часових зсувів, що становлять цикл неоднорідностей; n – число неоднорідностей у цьому циклі.

Знак указує напрямок зміни часового інтервалу між часовими зсувами. Додатний знак вказує на збільшення, а від’ємний – на зменшення цього інтервалу. Такі потоки є неоднорідними. На рисунку 21 наведено послідовності імпульсів запису і зчитування для двох варіантів співвідношення між частотами запису й зчитування $f_{зч} = (15/11) f_{зп}$ ($T_{зч} = (11/15) T_{зп}$) і $f_{зч} = (15/13) f_{зп}$ ($T_{зч} = (13/15) T_{зп}$) відповідно.

З рисунка 21, а видно, що при $T_{зч} = (11/15) T_{зп}$ величина

$$R = \left\lfloor \frac{T_{зч}}{T_{зп} - T_{зч}} \right\rfloor = \left\lfloor \frac{11}{4} \right\rfloor = 3, \text{ а } \frac{T_{зч}}{T_{зп} - T_{зч}} - R = \frac{11}{4} - 3 = -\frac{1}{4}.$$

Таким чином, між сусідніми часовими зсувами $R = 3$ імпульси, цикл виникнення неоднорідності $L = 4$, включаючи одну неоднорідність у циклі ($n = 1$). Від’ємний знак указує на зменшення інтервалу між сусідніми часовими зсувами. У розглянутому прикладі значення R змінюється від трьох до двох.



а – $f_{зч} = (15/11) f_{зп}$ ($T_{зч} = (11/15) T_{зп}$);

б – $f_{зч} = (15/13) f_{зп}$ ($T_{зч} = (13/15) T_{зп}$)

Рисунок 21 – Послідовності імпульсів запису і зчитування для двох варіантів співвідношення між частотами запису і зчитування

При $T_{зч} = (13/15) T_{зп}$ значення $R = \left\lceil \frac{T_{зч}}{T_{зп} - T_{зч}} \right\rceil = \left\lceil \frac{13}{2} \right\rceil = 7$, а

$$\frac{T_{зч}}{T_{зп} - T_{зч}} - R = \frac{13/15}{13/15 - 1} - 7 = \frac{13}{2} - 7 = -\frac{1}{2}.$$

Цьому відповідає потік, наведений на рисунку 21, б, у якому між сусідніми часовими зсувами число інформаційних імпульсів $R = 7$, цикл виникнення неоднорідностей $L = 2$ часових зсувів, включаючи одну неоднорідність у циклі. Значення R змінюється від семи до шести.

Таким чином, положення часових зсувів і кількість неоднорідностей змінюється при зміні співвідношення між частотами запису і зчитування. Якщо на рисунку 21, а при заданому співвідношенні між частотою запису і зчитування виникає одна неоднорідність, то при іншому співвідношенні цих частот з'являється інше число неоднорідностей.

Наприклад, при $T_{сч} = (13/18) T_{зп}$ величина $R = \left\lceil \frac{13}{5} \right\rceil = 3$, а

$$\frac{T_{сч}}{T_{зп} - T_{сч}} - R = \frac{13/18}{1 - 13/18} - 3 = \frac{13}{5} - 3 = -\frac{2}{5}.$$

Отримані результати вказують на те, що цикл появи неоднорідності $L = 5$, у складі якого міститься $n = 2$ неоднорідності. Значення R змінюється від трьох до двох.

Слід зауважити, що в реальних умовах співвідношення між частотами запису і зчитування змінюється в невеликих межах. Разом з тим очевидно, що зміщення положення часових зсувів у послідовності імпульсів необхідно компенсувати, щоб забезпечити розміщення і передачу службових символів на певних і незмінних часових позиціях. Зазначена компенсація можлива або вилученням на передавальній стороні «зайвих» імпульсів зчитування інформаційних символів з ЗП (при $f_{зч} > f_{зп}$), або їх додаванням (при $f_{зч} < f_{зп}$). У результаті відбувається узгодження швидкості компонентного потоку зі швидкістю агрегатного в перерахунку на один компонентний потік. Оповіщення приймальної сторони про всі операції (вилучення-додавання імпульсів зчитування) здійснюється передачею команд узгодження швидкостей (КУШ). Крім того, в асинхронних системах для циклової синхронізації приймального обладнання

агрегатної системи у складі службових сигналів передаються синхросигнали. Прийнятий агрегатний потік розділяється на компонентні, кожен з яких записується у «свій» ЗП тактами агрегатної системи в розрахунку на компонентну, а зчитування здійснюється тактами компонентної системи. У європейському варіанті плезіосинхронної цифрової ієрархії (PDH) значення кратності частоти зчитування дорівнює чотирьом, тобто $f_{зч1} = \frac{f_A}{4}$, де f_A – тактова частота групового (агрегатного) потоку. Так, наприклад, при об'єднанні чотирьох первинних потоків (E1) у вторинний (E2) частота запису кожного з потоків E1 складає значення $f_{зп} = f_{мчк} = 2048$ кГц, а частота зчитування кожного з цих потоків $f_{зч1} = \frac{f_{ГЧА}}{4} = \frac{8448}{4} = 2112$ кГц. Перевищення частоти зчитування над частотою запису дає можливість передавати в агрегатному потоці службову інформацію, яка необхідна для забезпечення нормальної роботи агрегатної системи. Таким чином, у розрахунку на один компонентний потік $f_{зч1} = f_{інф} + f_{сл1}$, де $f_{інф} = f_{зп}$, а $f_{сл1}$ – частота (швидкість) передачі службових сигналів у розрахунку на один компонентний потік.

Наприклад, в ЦСП ІКМ-120 для кожної компонентної системи (ІКМ-30) частота запису $f_{зп} = f_{інф} = 2048$ кГц, а частота зчитування $f_{зч1} = 2112$ кГц. Тому частота передачі службових сигналів $f_{сл1} = f_{зч1} - f_{інф} = 2112 - 2048 = 64$ кГц у розрахунку на один компонентний потік. В агрегатному потоці частота проходження службових сигналів у чотири рази вища: $f_{слА} = 4 f_{сл1} = 256$ кГц, а швидкість передачі агрегатного потоку складає $B = 4 * 2048 + 256 = 8448$ кбіт / с.

Слід підкреслити, що номінальна частота зчитування при об'єднанні асинхронних цифрових потоків завжди вибирається вищою від частоти запису. Часові зсуви, які з'являються за рахунок різниці номінальної частоти зчитування і частоти запису, є нормованими, що не потребують передачі інформації про їх наявність. Як неоднорідності сприймаються й усуваються ті часові зсуви, які порушують нормоване співвідношення частоти запису і номінальної частоти зчитування.

При побудові апаратури об'єднання цифрових потоків передбачена можливість як додатного, так і від'ємного

узгодження швидкостей, тобто двостороннього (додатно-від'ємного) узгодження швидкостей.

3.4 Структурна схема об'єднання (роз'єднання) асинхронних цифрових потоків

Структурна схема устаткування об'єднання асинхронних цифрових потоків наведена на рисунку 22.

У кожному з блоків асинхронного сполучення (БАС-1 ... БАС-4) вхідний компонентний потік записується в ЗП тактами компонентних потоків (систем передачі). Сигнали зчитування надходять до ЗП кожного з БАС з відповідних виходів генераторного обладнання (ГО) передавача агрегатної системи.

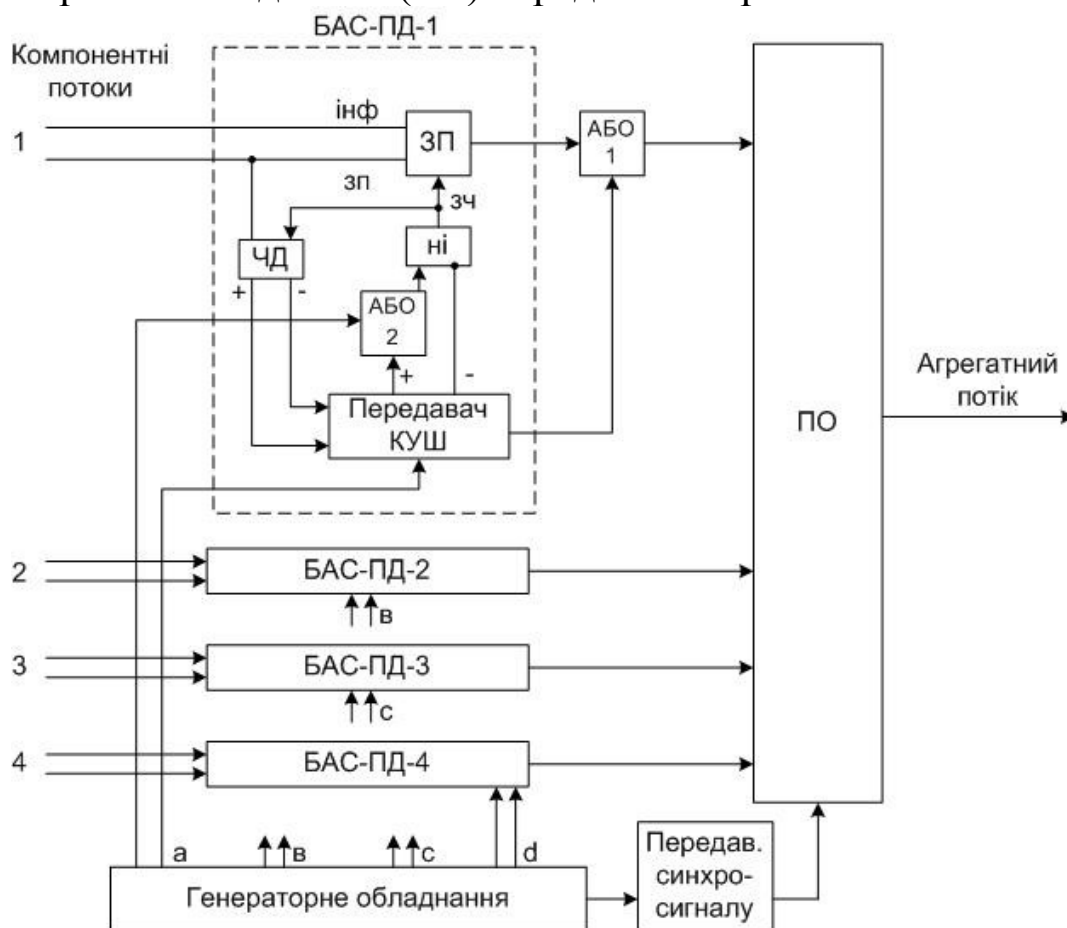


Рисунок 22 – Структурна схема устаткування об'єднання асинхронних цифрових потоків

Якщо співвідношення між частотами запису і зчитування ($f_{зч1} = f_{інф} + f_{сл1}$) незмінно, то сигнали зчитування надходять на вхід БАС через елементи АБО-2 і НІ без будь-яких змін.

Якщо ж це співвідношення змінюється в ту або іншу сторону і різниця між послідовностями імпульсів запису і зчитування досягає періоду зчитування, то часовий дискримінатор (ЧД), який визначає моменти появи і знак неоднорідності, видає відповідний сигнал у передавач КУШ. За цим сигналом у потік імпульсів зчитування або додається імпульс через схему АБО-2, або вилучається один з імпульсів зчитування шляхом заборони за командою з передавача КУШ проходження імпульсу через схему НІ.

Формування агрегатного потоку здійснюється шляхом об'єднання вихідних сигналів блоків асинхронного сполучення і службових сигналів.

На приймальній стороні агрегатний потік розділяється на компонентні потоки. Структурна схема устаткування роз'єднання прийнятого агрегатного потоку на компонентні наведена на рисунку 23.

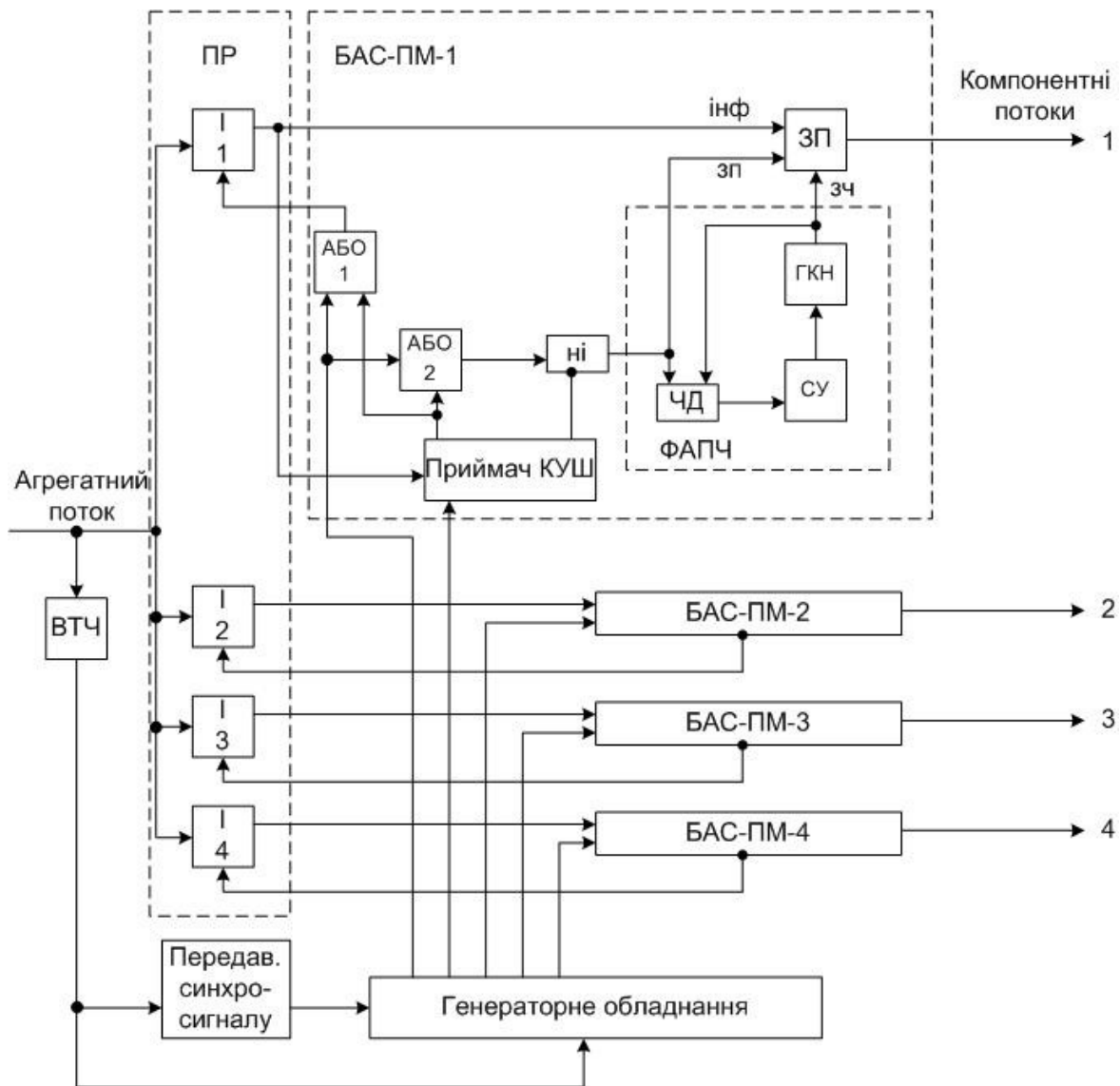


Рисунок 23 – Структурна схема устаткування роз'єднання прийнятого агрегатного потоку на компонентні

Агрегатний потік надходить на пристрій розділення, яке складається зі схем збігу I1 ... I4. Під дією імпульсних послідовностей з ГО відбувається розподіл прийнятого потоку між приймальними блоками асинхронного сполучення (БАС-ПМ-1 ... БАС-ПМ-4). У цих блоках здійснюється відновлення швидкості (тактової частоти) компонентних потоків. Для цього в кожному БАС-ПМ цифрова послідовність, виділена пристроєм поділу, надходить до ЗП. Сигнали запису надходять в ЗП з відповідних виходів ГО через АБО 2 і Ні. Зчитування з ЗП здійснюється сигналами, які формується пристроєм фазового

автопідстроювання частоти (ФАПЧ). Частота зчитування дорівнює середньому значенню частоти запису. До складу ФАПЧ входять часовий дискримінатор (ЧД), генератор, керований напругою (ГКН), і схема управління (СУ). ФАПЧ формує сигнал, що відповідає поточному значенню часового інтервалу між моментами запису і зчитування.

Якщо в складі прийнятого цифрового потоку відсутні КУШ, то сигнали записи з ГО надходять до ЗП без будь-яких змін. У разі появи КУШ про наявність позитивного часового зсуву (один з імпульсів зчитування вилучений на передавальній стороні) приймач КУШ видає сигнал на заборонний вхід схеми НІ, у результаті чого забороняється запис у ЗП. При отриманні КУШ про наявність від'ємного часового зсуву (на передавальній стороні здійснене зчитування інформації із ЗП додатковим імпульсом) приймач КУШ формує сигнал (імпульс), який проходить через АБО1, відкриває елемент І1 по одному з входів, що забезпечує приймання додатково зчитаного інформаційного імпульсу. Цей же імпульс з приймача КУШ через АБО 2 і НІ надходить до ЗП як додатковий сигнал запису. Крім того, цей же сигнал запису надходить на вхід ЧД схеми ФАПЧ. На другий вхід ЧД надходять сигнали зчитування з ГКН. На виході ЧД формується напруга постійної складової сигналу, яким і здійснюється управління частотою ГКН.

Об'єднання синхронних цифрових потоків є окремим випадком об'єднання асинхронних цифрових потоків, при якому частота зчитування кратна частоті запису. При цьому імпульсні позиції часових зсувів можна повністю використовувати для передачі службової інформації.

Перехід у синхронний режим роботи дає змогу спростити обладнання об'єднання цифрових потоків. Зокрема, оскільки в цьому режимі підтримується незмінне співвідношення між частотами запису і зчитування, то в передавальній частині обладнання (рисунок 22) виявляються непотрібними передавач КУШ, схема АБО-2. На заборонний вхід схеми НІ керуючий сигнал надходить з часового дискримінатора. У приймальному обладнанні (рисунок 23) прийняті сигнали після поділу надходять до ЗП, у якому зчитування здійснюється з частотою, рівною частоті запису цих сигналів у ЗП передавальної частини

обладнання. В обладнанні об'єднання синхронних цифрових потоків відпадає необхідність у пристрої ФАПЧ, приймачі КУШ, елементах І, ІІ. При цьому здійснюється рівномірний розподіл часових зсувів, сформованих у передавальному пристрої.

ІНДИВІДУАЛЬНЕ ЗАВДАННЯ. У відповідності до варіанта завдання (додаток А) провести об'єднання асинхронних потоків за формулами (5) – (8), розрахувати кількість інформаційних символів R між сусідніми часовими зсувами і величину періодичності неоднорідностей $\left(\frac{T_{сч}}{T_{зн} - T_{сч}}\right)$, зобразити послідовності імпульсів запису і зчитування варіанта завдання подібно до рисунка 21.

Контрольні питання

- 1 Наведіть способи об'єднання цифрових потоків. У чому збіжність та різниця між способами?
- 2 Призначення пристрою ЗП у блоці сполучення БС в обладнанні об'єднання цифрових потоків?
- 3 Які цифрові потоки є неоднорідними?
- 4 Які способи використовуються при об'єднанні цифрових потоків для вилучення неоднорідностей потоку?
- 5 Розрахуйте кількість інформаційних символів між сусідніми часовими зсувами ЦСП ІКМ-120, 480.

СПИСОК ЛІТЕРАТУРИ

- 1 Беллами, Дж. Цифровая телефония [Текст]: пер. с англ. / под. ред. А.Н. Берлина, Ю.Н. Чернышова. – М.: Эко-Трендз, 2004. – 640 с.
- 2 Олифер, В.Г. Компьютерные сети. Принципы, технологии, протоколы. [Текст] / В.Г. Олифер, Н.А. Олифер. – 3-е изд. – С.Пб.: Питер, 2006. – 985 с.

3 Скляр, Бернад. Цифрова зв'язь. Теоретическіе основи і практичне застосування [Текст]: пер. с англ. / Бернад Скляр. – 2-е изд., испр. – М.: Издательский дом "Вильямс", 2003. – 1104 с.

4 Бірюков, Н.Л. Транспортні мережі і системи електросв'язі. Системи мультиплексування [Текст]: учеб. для студ. вузів по спеціальності "Телекомунікації" / Н.Л. Бірюков, В.К. Стеклов. – К., 2003. – 352 с.

5 Гитлиц, М.В. Теоретическіе основи багатоканалної зв'язі [Текст]: учеб. посібник для вузів зв'язі / М.В. Гитлиц, А.Ю. Лев. — М.: Радио и связь, 1985. — 248 с.

6 Кириллов, В.И. Многоканальные системы передачи [Текст]: учебник / В.И. Кириллов. — М.: Новое знание, 2002. — 751 с.

7 Крухмалев, В.В. Цифровые системы передачи [Текст]: учеб. посібник для высших учебных заведений / В.В. Крухмалев, В.Н. Гордиенко, А.Д. Моченов. – М.: Горячая линия-Телеком, 2007. — 351 с.

8 Гордиенко, В.Н. Многоканальные телекоммуникационные системы [Текст] / В.Н. Гордиенко, М.С. Тверецкий. – М.: Горячая линия-Телеком, 2005. — 416 с.

9 Современные телекоммуникации [Электронный ресурс]. – Режим доступа: <http://www.velcom.ru/ar385.html>. - Загл. с экрана. (Дата обращения: 3.08.2013).

10 Козар, Л.М. Методичний посібник з додержання вимог нормоконтролю у студентській навчальній звітності [Текст] / Л.М. Козар, Є.В. Коновалов. – Харків: УкрДАЗТ, 2004. – 38 с.

ДОДАТОК А

Вихідні дані

Варіант	Значення вхідного сигналу	Інформаційна послідовність	Співвідношення між інтервалами запису та зчитування
1	2	3	4
0	-121/+242	110000110000001110000001100	$T_{сч} = (11/15) T_{зп}$ $T_{сч} = (13/15) T_{зп}$

1	+120	111000011110000001110000001100 1	Тсч = (11/14) Тзп Тсч = (14/11) Тзп
2	-200	100000001110001000110100000000 1	Тсч = (7/9) Тзп Тсч = (9/7) Тзп
3	+430	100011010000000011000000111000	Тсч = (7/11) Тзп Тсч = (11/7) Тзп
4	-690	11100010000001011010000000011	Тсч = (2/3) Тзп Тсч = (3/2) Тзп
5	+33	00000011100000000011011000000	Тсч = (10/11) Тзп Тсч = (11/10) Тзп
6	-1280	11110000111000000001010010000	Тсч = (11/17) Тзп Тсч = (17/11) Тзп
7	+720	01100000011100100000000100001	Тсч = (7/10) Тзп Тсч = (10/7) Тзп
8	-1040	11000111000010000000001110000	Тсч = (9/11) Тзп Тсч = (11/9) Тзп
9	+301	01010000110000000010001100000	Тсч = (4/9) Тзп Тсч = (9/4) Тзп
10	-890	11110001110000000001000110000	Тсч = (11/13) Тзп Тсч = (13/11) Тзп
11	+2350	101011000011000000001000010000	Тсч = (5/7) Тзп Тсч = (7/5) Тзп
12	-580	01010000000011000111000000000	Тсч = (13/18) Тзп Тсч = (18/13) Тзп
13	+1200	101111000010000000011000000000	Тсч = (7/11) Тзп Тсч = (11/7) Тзп
14	-760	010000110000000111110000000001 1	Тсч = (2/3) Тзп Тсч = (3/2) Тзп
15	+1890	111000011000000001000000001000	Тсч = (11/14) Тзп Тсч = (14/11) Тзп
16	-2080	000000011110000100000000100010	Тсч = (7/9) Тзп Тсч = (9/7) Тзп
17	+377	111100000000011100010000110000	Тсч = (4/9) Тзп Тсч = (9/4) Тзп

Продовження додатка А

1	2	3	4
18	-999	00001110001100000010000110000	Тсч = (11/13) Тзп Тсч = (13/11) Тзп
19	+1500	110000010000000011000011000000	Тсч = (5/7) Тзп Тсч = (7/5) Тзп
20	-150	11110000111000100000000010000	Тсч = (13/18) Тзп Тсч = (18/13) Тзп

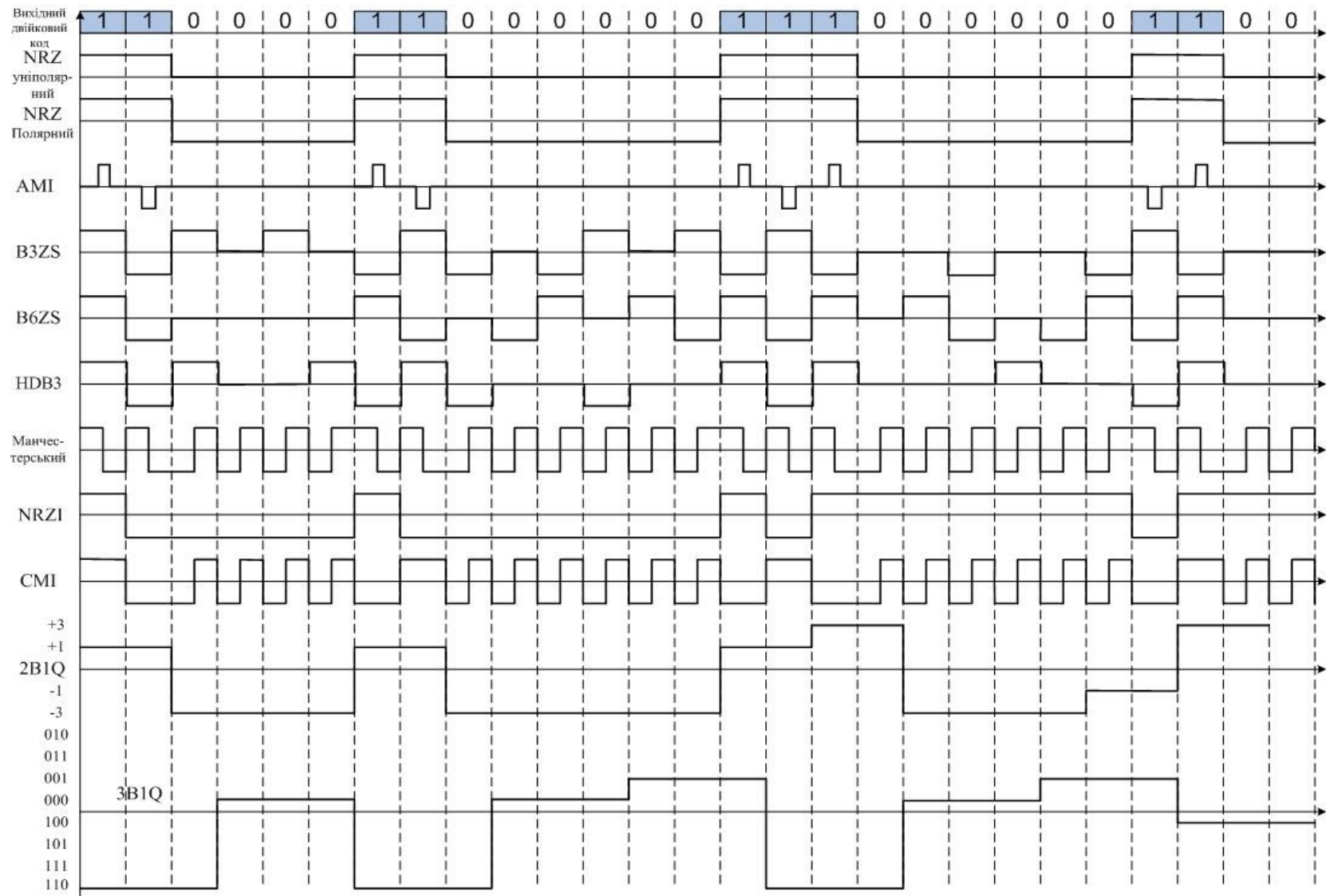


Рисунок 16 – Приклад кодування цифровими та логічними кодами