



УКРАЇНА

(19) UA

(11) 44329

(13) C2

(51) 7 H04J3/06, H04L25/49

МІНІСТЕРСТВО ОСВІТИ  
І НАУКИ УКРАЇНИ

ДЕРЖАВНИЙ ДЕПАРТАМЕНТ  
ІНТЕЛЕКТУАЛЬНОЇ  
ВЛАСНОСТІ

## ОПИС ДО ПАТЕНТУ НА ВИНАХІД

видається під  
відповідальність  
власника  
патенту

### (54) СПОСІБ МУЛЬТИПЛЕКСУВАННЯ/ДЕМУЛЬТИПЛЕКСУВАННЯ

1

2

(21) 98020821

(22) 15 08 1996

(24) 15 02 2002

(86) PCT/GB96/01991, 15 08 1996

(31) 9517198 9

(32) 23 08 1995

(33) GB

(46) 15 02 2002, Бюл. № 2, 2002 р

(72) Беррі Алан Девід, GB

(73) МАРКОНІ КОММУНІКЕЙШНС ЛІМІТЕД, GB

(56) US 3622963 A, 23 11 1971

EP 0655850 A, 31 05 1995

(57) 1 Спосіб мультиплексування/демультиплексування, який передбачає, що в процесі мультиплексування, беруть щонайменше один біт у часовому інтервалі кожного з множини сигналів інформаційних даних для формування послідовності інформаційних даних, послідовно приєднують послідовності інформаційних даних до фіксованого двійкового коду для формування комбінованого набору даних, послідовно передають комбіновані набори даних у вигляді комбінованого сигналу даних і в процесі демультиплексування аналізують комбіно-

ваний сигнал даних у блоках даних довжиною, що дорівнює довжині комбінованого набору даних, для виділення інформаційних даних, причому код вибирають таким чином, що будь-який блок даних має для кожної можливої послідовності інформаційних даних один набір однозначно визначених значень незалежно від початкової точки блока у комбінованому сигналі даних

2 Спосіб мультиплексування/демультиплексування за п. 1, у якому у часовому інтервалі кожного з множини сигналів інформаційних даних беруть лише один біт

3 Спосіб мультиплексування/демультиплексування за п. 1, у якому два із сигналів інформаційних даних мають різні швидкості передачі, при цьому протягом згаданого часового інтервалу для формування послідовності інформаційних даних беруть різне число бітів із згаданих двох сигналів

4 Спосіб мультиплексування/демультиплексування за будь-яким із пп. 1 - 3, у якому набір комбінованих даних повторюють стільки разів, скільки необхідно для отримання необхідної швидкості передачі з мультиплексуванням

(13) C2

(11) 44329

(19) UA

Винахід належить до способу мультиплексування/демультиплексування, зокрема, для передачі множини двійкових цифрових сигналів у єдиному двійковому цифровому сигналі високої швидкості передачі, і для якого не потрібна процедура вирівнювання на приймальному боці для розкладання комбінованого сигналу на його складові частини і перенесення їх у відповідні належні місця призначення.

Відомі способи мультиплексування/демультиплексування використовують процедуру, яку часто називають "вирівнюванням", для однозначної ідентифікації складових частин комбінованого сигналу. Наприклад, деякі способи мультиплексування використовують бітове або байтове чергування сигналів більш низької швидкості передачі і уводять у сигнали, що передаються, деякі додаткові біти, наприклад, біти слова вирівнювання кадру, коди парності або коди Картера. Інші способи використовують блочне кодування з паралельними входами і послідовним виходом.

На приймальному боці, де повинно здійснюватися демультиплексування, прийнятий сигнал піддають дві, обернений чергуванню або декодують, причому повинні бути відомі початок і кінець кожного кадрового або кодового блока. Ця процедура часто називається "вирівнюванням" і містить пошук позицій вирівнювання в послідовній або в паралельній формі для коректного вирівнювання шляхом підрахунку кількості вірних і хибних послідовних позицій вирівнювання для полегшення зупинки або початку процедури пошуку, або використання іншого методу визначення початку і кінця кожного кадру або блока, виходячи з деякої властивості, наданої йому шляхом мультиплексування. Відомими прикладами є структури кадру, що визначаються в рекомендаціях ITU G 741 і G 751.

Другий відомий спосіб, в якому не міститься обов'язкова вимога, щоб швидкість передачі перевищувала суму окремих швидкостей сигналів, які мультиплексуються, використовує бітове або байтове чергування, але залежить, щонайменше, від одного з індивідуальних сигналів, що мають відому властивість своєї бітової послідовності, яка відрізняється від інших, наприклад, слово кадру, щоб його можна було ідентифікувати. Цей спосіб можна розглядати як спеціальний випадок раніше описаного способу, в якому додаткові біти вносяться в один з індивідуальних сигналів перед мультиплексуванням. Але він все ж вимагає процедури вирівнювання в процесі демультиплексування.

Даний винахід, в одному з його аспектів, передбачає спосіб мультиплексування/демультиплексування, який містить в процесі мультиплексування, взяття, щонайменше, одного біту з часового інтервалу кожного з множини сигналів інформаційних даних для формування послідовності інформаційних даних, комбінування послідовності інформаційних даних з фіксованим двійковим кодом для формування комбінованого набору даних, послідовну передачу комбінованих наборів даних у вигляді одного сигналу, а також, в процесі демультиплексування, аналіз комбінованого сигналу даних в блоках з довжиною, що дорівнює довжині

комбінованого набору даних, для виділення інформаційних даних, при цьому код вибирають таким чином, що будь-який блок має для кожної можливої послідовності інформаційних даних один набір унікальних значень незалежно від початкової точки блока в комбінованому сигналі даних.

При використанні цього способу немає необхідності включати будь-які процедури вирівнювання в процесі демультиплексування.

Цей спосіб дає можливість передавати два, або більше, нормально синхронізованих двійкових цифрових сигналів у єдиному двійковому цифровому сигналі з більш високою швидкістю передачі, яка кратна номінальній швидкості передачі будь-якого з індивідуальних сигналів. Даний спосіб можна також використовувати в тих випадках, коли індивідуальні сигнали мають різні, наприклад, більш низькі швидкості передачі, і швидкість передачі мультиплексованого сигналу кратна найбільшому спільному дільнику всіх швидкостей індивідуальних сигналів. В цьому випадку під час будь-якого часового інтервалу різні кількості бітів можна брати з кожного індивідуального сигналу. Аналогічно, сигнал більш високої швидкості може бути лише допоміжним сигналом, що передається як частина сигналу ще більш високої швидкості з використанням того ж або іншого способу мультиплексування. В даному випадку поняття "часовий інтервал" означає час, протягом якого з кожного комбінованого сигналу можна взяти ціле число бітів, бажано найменше.

Для пояснення даного винаходу, варіанти його реалізації далі описуються з посиланнями на ілюструючі його креслення, де зображено передавач мультиплексування і приймач демультиплексування, які можуть бути використані для реалізації способу, що відповідає винаходу.

Приклад реалізації буде викладатися з посиланнями на креслення, де індивідуальні сигнали з швидкістю 64Кбіт/с, X і Y, перетворюються в сигнал з більш високою швидкістю передачі в 1024Кбіт/с. Мультиплексор, або передавач Tx, містить входи 1, 2 для приймання відповідних потоків трафіка або реальних даних X, Y. Кожний вхід 1, 2 з'єднано з відповідним тригером 3, від якого сигнали тактуються з швидкістю 64Кбіт/с в зсувний регістр 4 паралельного завантаження і послідовного виходу. Решта цифрових сигналів складається з фіксованого коду A, B, значення якого вибираються згідно з описаними нижче правилами. Мультиплексовані послідовні дані потім зчитуються з зсувного регістру 4 з швидкістю 1024Кбіт/с.

В одному часовому інтервалі швидкості передачі індивідуального сигналу містяться два біти інформації, що підлягає передачі, а саме, по одному від кожного з двох входів X і Y. Для передачі з більш високою швидкістю передачі в цьому інтервалі часу є 16 бітів. Сигнал з більш високою швидкістю передачі передають у вигляді чотирьох блоків 5 ABXY.

В приймачі Rx схема 10 виділення тактового сигналу виділяє тактовий сигнал частотою 1024КГц для завантаження вхідних послідовних даних в послідовний зсувний регістр 11. Паралель-

не завантаження потім тактують в буфері 13 перед тим, як здійснювати завантаження з тактовою швидкістю 64КГц в блочний декодер 14. Блок 5, що містить чотири послідовні біти даних, декодується в декодері 14, і реальні або інформаційні сигнали X, Y виходять з відповідних вихідних ліній 15 через тригери 16. Як буде викладено нижче, процедура декодування не вимагає будь-якого вирівнювання для виділення коректних значень X і Y, тобто завдяки фіксованому бітовому коду АВ не має значення, який з будь-яких чотирьох сусідніх бітів даних потрібно декодувати першим. В цьому випадку є дві задовільні комбінації А і В, а саме, 01 або 10. Чотири сусідні біти, що вибираються в декодері 14, можуть представляти будь-яку з чотирьох можливих комбінацій АВХУ, ВХУА, ХУАВ або УАВХ. В останньому випадку біти Х і У можуть бути з різних часових інтервалів початкових сигналів, якщо вибір роблять по межі набору з 16 бітів, але це вносить затримку лише в один біт в одному сигналі порівняно з іншим. Щоб показати, що декодування є однозначно визначеним, нижче перелічені всі можливі комбінації, а десяткове значення кожного блоку наведено як допоміжний засіб,

якщо  $X = 0\ 0\ 1\ 1$  (двійковий)

і  $Y = 0\ 1\ 0\ 1$  (двійковий)

і якщо  $A = 0$  і  $B = 1$ , то переданими комбінаціями будуть чотири блоки

$0100 = 4$   $0101 = 5$   $0110 = 6$   $0111 = 7$ ,

і блоками з чотирьох бітів, вибраними на приймальному боці, будуть

$0100 = 4$   $0101 = 5$   $0110 = 6$   $0111 = 7$

або  $1000 = 8$   $1010 = 10$   $1100 = 12$   $1110 = 14$

або  $0001 = 1$   $0101 = 5$   $1001 = 9$   $1101 = 13$

або  $0001 = 2$   $1010 = 10$   $0011 = 3$   $1011 = 11$

Звідси очевидно, що для кожної можливої інформаційної послідовності блок даних має однозначно визначений набір значень незалежно від початкової точки даних в блоці

Є два невикористані блоки, а саме - із значеннями 0 і 15, які могли виникнути внаслідок помилок в прийнятому сигналі, і їх переважно декодують як X, Y = 0, 0 і 1, 1, відповідно. Таблиця декодування має вигляд

Значення прийнятого блоку	X	Y
0, 1, 2, 4, 8	0	0
5, 10	0	1
3, 6, 9, 12	1	0
7, 11, 13, 14, 15	1	1

аналогічно, якщо  $A = 1$  і  $B = 0$ , то передані комбінації являють собою чотири блоки

$1000 = 8$   $1001 = 9$   $1010 = 10$   $1011 = 11$ ,

а блоки з чотирьох бітів, вибрані на приймальному боці, будуть мати вигляд

$1000 = 8$   $1001 = 9$   $1010 = 10$   $1011 = 11$

або  $0001 = 1$   $0011 = 3$   $0101 = 5$   $0111 = 7$

або  $0010 = 2$   $0110 = 6$   $1010 = 10$   $1110 = 14$

або  $0100 = 4$   $1100 = 12$   $0101 = 5$   $1101 = 13$

Отриманий результат схожий на викладений вище, таблиця декодування має вигляд

Значення прийнятого блоку	X	Y
0, 1, 2, 4, 8	0	0

3, 6, 9, 12	0	1
5, 10	1	0
7, 11, 13, 14, 15	1	1

Немає необхідності в тому, щоб повторювати блоки коду для отримання сигналу потрібної більш високої швидкості передачі. Як другий приклад можна навести випадок, коли два сигнали швидкості 64Кбп/с передають в сигналі більш високої швидкості в 256Кбп/с з цими ж блоками і згідно з тими ж правилами декодування

Взагалі, один біт з кожного часового інтервалу для двох або більше індивідуальних цифрових двоїчних сигналів комбінується з відповідним числом фіксованих бітів, і набір з N бітів передається послідовно з більш високою швидкістю. За бажанням цей набір можна повторити ціле число разів для забезпечення відповідності більш високій швидкості. Двійкові значення фіксованих бітів вибираються таким чином, щоб будь-які послідовні набори з N бітів, прийняті на більш високій швидкості, можна було однозначно декодувати визначеним чином для знаходження значень бітів складових сигналів більш низької швидкості передачі. Початок і кінець набору з N бітів, які повинні бути декодовані, не повинні обов'язково збігатися з бітами початкового набору. Якщо набір з N бітів повторюється для отримання в цілому K наборів, включаючи початковий, то достатньо буде декодувати лише один набір з N бітів з кожних прийнятих K наборів. Якщо сигнали з більш низькою швидкістю передачі не синфазні один одному, і якщо набори повторюються, щонайменше, один раз, то кожний набір з N бітів може містити самий останній біт кожного з каналів більш низької швидкості передачі, але в цьому випадку вихідні сигнали з більш низькою швидкістю передачі будуть мати однакову фазу, якщо декодовано лише один набір з наборів, тобто сигнали будуть мати різні відносні затримки, що вводяться у відповідні канали передачі. Або, якщо декодується кожний набір, то вихідну фазу сигналів більш низької швидкості передачі буде збережено на етапах квантування, що вводяться K наборами, на один часовий інтервал.

Якщо сигнали більш низької швидкості передачі не є точно синхронними з урахуванням кратності із сигналом більш високої швидкості передачі, то цей спосіб призведе або до пропуску бітів, або до появи етапів "тремтіння", квантованих K наборами на вищезгаданий часовий інтервал.

Нижче наведені інші приклади, з яких видно, що число каналів можна збільшувати згідно з довжиною фіксованого бітового коду відповідно до конкретних обставин.

Приклад 3. Два окремі сигнали з швидкістю 64Кбп/с передають в сигналі більш високої швидкості передачі 320Кбп/с. В одному часовому інтервалі швидкості передачі індивідуального сигналу передаються два біти інформації, X і Y, п'ять бітів передають на більш високій швидкості передачі. Сигнал високої швидкості передачі передають як послідовні блоки ABCXY, де A, B і C є фіксованими бітами, вибраними так, щоб забезпечувати можливість демультіплексування без виконання вирівнювання. В цьому випадку є чотири задовільні комбінації A, B і C, а саме - 001, 011, 100 і 110.

На приймаючій стороні достатньо вибрати

будь-які п'ять послідовних бітів і декодувати їх, як показано нижче. П'ять бітів, що вибираються, являтимуть собою будь-яку з п'яти можливих комбінацій ABCXY, BCXYA, CXYAB, XYABC, YABCX. В останньому випадку біти X і Y можуть бути з різних часових інтервалів початкових сигналів, але це вносить лише 1-бітову затримку в одному сигналі порівняно з іншим. Тут наведені всі можливі комбінації (десятькове значення кожного блоку наведено

або	00100 = 4	00101 = 5
або	01000 = 8	01010 = 10
або	10000 = 16	10100 = 20
або	00001 = 1	01001 = 9
або	00010 = 2	10010 = 18

Таким чином, очевидно, що декодування блоків можна виконати без будь-якої неоднозначності, є дванадцять невикористаних блоків, а саме блоків значень 0, 11, 13, 15, 21, 22, 23, 26, 27, 29, 30 і 31, які можуть виникати внаслідок помилок у прийнятому сигналі, і їх переважно декодують, щоб отримати множення з мінімальною похибкою, як показано нижче. Тоді таблиця декодування буде мати наступний вигляд

Значення прийнятого блоку		X	Y
(без помилок)	(з помилками)		
1, 2, 4, 8, 16	0	0	0
5, 9, 10, 18, 20	11, 13, 21, 22, 26	0	1
3, 6, 12, 17, 24		1	0
7, 14, 19, 25, 28	15, 23, 27, 29, 30, 31	1	1

(аналогічні таблиці декодування можна скласти для ABC = 011, 100 або 110)

Приклад 4. Три індивідуальні сигнали з швидкістю 64Кбіт/сек передаються в сигналі більш високої швидкості передачі в 384Кбіт/с. В одному часовому інтервалі швидкості індивідуального сигналу є три інформаційні біти для передачі - X, Y і Z, є 6 бітів для передачі на більш високій

001000	001001	001010	001011
= 8	= 9	= 10	= 11

і блоків з шести бітів, вибраних на приймаючій стороні, будуть наступними

001000	001001	001010	001011
= 8	= 9	= 10	= 11

або

010000	010010	010100	010110
= 16	= 18	= 20	= 22

або

100000	100100	1010000	101100
= 32	= 36	= 40	= 44

або

000001	001001	010001	011001
= 1	= 9	= 17	= 25

або

000010	010010	100010	110010
= 2	= 18	= 34	= 50

000100	100100	000101	100101
= 4	= 36	= 5	= 37

Таким чином, очевидно, що декодування блоків виконано без будь-якої неоднозначності, єдиними повторами є блоки значень 9, 18 і 36, які коректно декодуються як XYZ = 001. Є 19 невикористаних блоків, а саме блоків значень 0, 21, 23, 27, 29, 31, 42, 43, 45, 46, 47, 53, 54, 55, 58,

Значення прийнятого блоку		X	Y	Z
(без помилок)	(з помилками)			

як допоміжний засіб)

якщо X = 0 0 1 1 (двійковий)

якщо Y = 0 1 0 1 (двійковий)

і якщо A = 0, B = 0 і C = 1, тоді передані комбінації являють собою блоки

00100 = 4 00101 = 5 00110 = 6 01111 = 7,

а блоки з п'яти бітів, вибрані на приймаючій стороні, будуть мати вигляд

00110 = 6	00111 = 7
01100 = 12	01110 = 14
11000 = 24	11100 = 28
10001 = 17	11001 = 25
00011 = 3	10011 = 19

швидкості. Сигнал високої швидкості передається у вигляді послідовних блоків ABCXYZ, де A, B і C є фіксованими бітами, вибраними таким чином, щоб можна було забезпечити можливість демультимплексування без вирівнювання. В цьому випадку є чотири можливі задовільні комбінації A, B і C, а саме - 001, 011, 100 і 110

На приймальному боці достатньо вибрати будь-які шість послідовних бітів і декодувати їх, як показано нижче. Шість вибраних бітів з будь-якою з шести можливих комбінацій ABCXYZ, BCXYZA, CXYZAB, XYZABC, YZABCX, ZABCSY. В останніх двох випадках біти X або Z можуть бути з різних часових інтервалів початкових сигналів, але це вносить лише 1-бітову затримку в один або два сигнали порівняно з іншим(ми). Нижче наведені всі можливі комбінації (десятькові значення кожного блоку наведено в якості допоміжного засобу)

якщо X = 0 0 0 0 0 1 1 1 1

і Y = 0 0 1 1 0 0 1 1

і Z = 0 1 0 1 0 1 0 1

і якщо A = 0, B = 0 і C = 1, тоді передані комбінації блоків

001100	001101	001110	001111
= 12	= 13	= 14	= 15

001100	001101	001110	001111
= 12	= 13	= 14	= 15

011000	011010	011100	011110
= 24	= 26	= 28	= 30

110000	110100	111000	111100
= 48	= 52	= 56	= 60

100001	101001	110001	111001
= 33	= 41	= 49	= 57

000011	010011	100011	110011
= 3	= 19	= 35	= 51

або

000110	100110	000111	100111
= 6	= 38	= 7	= 39

59, 61, 62 і 63, які можуть виникнути внаслідок помилок у прийнятому сигналі, і їх бажано декодувати, для отримання множення з мінімальною похибкою, як показано нижче. Тоді таблиця декодування матиме вигляд

1, 2, 4, 8, 16, 32	0	0	0	0
9, 18, 36		0	0	1
5, 10, 17, 20, 34, 40	21, 42	0	1	0
11, 22, 25, 37, 44, 50	27, 45, 54 [див Прим 1]	0	1	1
3, 6, 12, 24, 33, 48		1	0	0
13, 19, 26, 38, 41, 52	23, 29, 43, 46, 53, 58 [2]	1	0	1
7, 14, 28, 35, 49, 56		1	1	0
15, 30, 39, 51, 57, 60	31, 47, 55, 59, 61, 62, 63	1	1	1

Примітка 1 для множення з мінімальною похибкою будь-які з цих 3 значень блоків могли бути декодовані як XYZ = 101 замість 011

Примітка 2 для множення з мінімальною похибкою будь-які з цих 6 значень блоків могли бути декодовані як XYZ = 011 або 110 замість 101

(Аналогічні таблиці декодування можна скласти для ABC = 011, 100 або 110)

#### Приклад 5

Чотири індивідуальні сигнали швидкості 64Кбіт/сек передаються в сигналі з більш високою швидкістю 512Кбіт/сек. В одному часовому інтервалі швидкості передачі індивідуального сигналу є чотири інформаційні біти для передачі W, X, Y і Z, 8 бітів передаються з більш високою швидкістю. Сигнал більш високої швидкості передається у вигляді послідовних блоків ABCDWXYZ, де A, B, C і D є фіксованими бітами, вибраними таким чином, щоб забезпечити можливість демультиплексування без вирівнювання. В цьому випадку є шість можливих задовільних комбінацій A, B, C і D, а саме 0001, 0011, 0111, 1000, 1100 і 1110.

Вісім бітів, що вибираються на приймальному боці, являтимуть собою будь-яку з восьми можливих комбінацій ABCDWXYZ, BCDWXYZA, CDWXYZAB, DWXYZABC, WXYZABCD, XYZABCDW, YZABCDWX, ZABCDWXY. В останніх трьох випадках деякі з бітів W, X, Y або Z можуть бути з різних часових інтервалів початкових сигналів, але це вносить лише 1-бітову затримку в деяких сигналах порівняно з іншим(ми). Щоб показати, що декодування є однозначно визначеним, наведено всі можливі комбінації, після прикладу 6 нижче, для випадку, коли ABCD є 0001 (десятькове значення кожного блоку наведено лише тільки для того, щоб уникнути непотрібної довжини еквівалентного двоїчного представлення).

#### Приклад 6

Чотири індивідуальні сигнали швидкості 64Кбіт/с передаються в сигналі більш високої швидкості 1024Кбіт/с. В одному часовому інтервалі швидкості індивідуального сигналу є чотири біти інформації, що передається W, X, Y і Z, є 16 бітів для передачі на більш високій швидкості. Сигнал високої швидкості передається у вигляді двох послідовних блоків ABCDWXYZ, де A, B, C і D є фіксованими бітами, вибраними для забезпечення можливості демультиплексування без вирівнювання. В цьому випадку знову є шість можливих сполучень A, B, C і D, які є задовільними, а саме 0001, 0011, 0111, 1000, 1100 і 1110.

На приймальному боці достатньо вибрати будь-які вісім послідовних бітів з шістнадцяти і декодувати їх, як показано нижче. Вісім вибраних бітів будуть будь-якими з восьми можливих комбінацій ABCDWXYZ, BCDWXYZA, CDWXYZAB, DWXYZABC, WXYZABCD, XYZABCDW, YZABCDWX, ZABCDWXY. В останніх трьох випад-

ках деякі з бітів W, X, Y або Z можуть бути з різних часових інтервалів початкових сигналів, якщо вибір зроблено по межі одного з початкових наборів з 16 бітів, але це вносить тільки 1-бітову затримку в деяких сигналах порівняно з іншим(ми). Щоб показати, що декодування є однозначно визначеним, тут наведені всі комбінації для випадку, коли ABCD є 0001, (наведено лише десяткове значення кожного блоку),

(це відноситься до обох прикладів 5 і 6)

якщо W = 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 (двійковий)

і X = 0 0 0 0 1 1 1 1 0 0 0 0 1 1 1 1 (двійковий)

і Y = 0 0 1 1 0 0 1 1 0 0 1 1 0 0 1 1 (двійковий)

і Z = 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 (двійковий)

і якщо A, B, C, D = 0, 0, 0, 1 - тоді комбінації, що передаються, являють собою блоки, десятковими значеннями яких є

16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31,

а блоки з восьми бітів, вибраних на приймальному боці, мають значення

16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31

або

32 34 36 38 40 42 44 46 48 50 52 54 56 58 60 62

або

64 68 72 76 80 84 88 92 96 100 104 108 112 116

120 124

або

128 136 144 152 160 168 176 184 192 200 208

216 224 232 240 248

або

1 17 33 49 65 81 97 113 129 145 161 177 193

209 225 241

або

2 34 66 98 130 162 194 226 3 35 67 99 131 163

195 227

або

4 68 132 196 5 69 133 197 6 70 134 198 7 71 135

199

або

8 136 9 137 10 138 11 139 12 140 13 141 14 142

15 143

Таким чином, очевидно, що декодування блоків виконано без будь-якої неоднозначності, єдиними повторами, що мають місце, є блоки значень 17, 34, 68 і 136, які кодовані коректно як WXYZ = 0001. Є 132 невикористані блоки, які могли виникнути внаслідок помилок у прийнятому сигналі, і знову пропонується, що вони повинні декодуватися з забезпеченням множення з мінімальною похибкою. Тоді таблиця декодування для блоків, що не мають помилок, буде мати вигляд

11

44329

12

Значення прийнятого блоку (без помилок)	W	X	Y	Z
1, 2, 4, 8, 16, 32, 64, 128	0	0	0	0
17, 34, 68, 136	0	0	0	1
9, 18, 36, 72, 144	0	0	1	0
19, 38, 76, 152, 304	0	0	1	1
5, 10, 20, 40, 80, 160	0	1	0	0
21, 42, 84, 168, 336	0	1	0	1
11, 22, 44, 88, 176, 352	0	1	1	0
23, 46, 92, 184, 368	0	1	1	1
3, 6, 12, 24, 48, 96, 192	1	0	0	0
25, 50, 100, 200, 400	1	0	0	1

13, 26, 52, 104, 208, 416	1	0	1	0
27, 54, 108, 216, 432	1	0	1	1
7, 14, 28, 56, 112, 224	1	1	0	0
29, 58, 116, 232, 464	1	1	0	1
15, 30, 60, 120, 240	1	1	1	0
31, 62, 124, 248, 496	1	1	1	1

(аналогічні таблиці декодування можна генерувати для ABCD = 0011, 0111, 1000, 1100 або 1110)

Незважаючи на надмірну складність останніх прикладів, все ж очевидно, що для кожної можливої послідовності реальних даних можна вибрати фіксовані бітові коди, які забезпечують для комбінованого сигналу наявність набору особливих значень незалежно від початкової точки блоку даних, що декодуються