



ФЕДЕРАЛЬНАЯ СЛУЖБА
ПО ИНТЕЛЛЕКТУАЛЬНОЙ СОБСТВЕННОСТИ

(12) ОПИСАНИЕ ИЗОБРЕТЕНИЯ К ПАТЕНТУ

(52) СПК

H03M 13/036 (2006.01); H03M 13/1165 (2006.01); H03M 13/118 (2006.01); H03M 13/255 (2006.01); H03M 13/2707 (2006.01); H03M 13/271 (2006.01); H03M 13/2906 (2006.01); H03M 13/356 (2006.01); H03M 13/616 (2006.01); H03M 13/6552 (2006.01); H04L 1/0057 (2006.01); H04L 1/0071 (2006.01)

(21)(22) Заявка: 2015103856, 03.06.2014

(24) Дата начала отсчета срока действия патента:
03.06.2014

Дата регистрации:
06.06.2018

Приоритет(ы):

(30) Конвенционный приоритет:
12.06.2013 JP 2013-124187

(43) Дата публикации заявки: 27.08.2016 Бюл. № 24

(45) Опубликовано: 06.06.2018 Бюл. № 16

(85) Дата начала рассмотрения заявки РСТ на
национальной фазе: 05.02.2015

(86) Заявка РСТ:
JP 2014/064672 (03.06.2014)

(87) Публикация заявки РСТ:
WO 2014/199865 (18.12.2014)

Адрес для переписки:
109012, Москва, ул. Ильинка, 5/2, ООО
"Союзпатент"

(72) Автор(ы):

СИНОХАРА Юдзи (JP),
ЯМАМОТО Макико (JP)

(73) Патентообладатель(и):

СОНИ КОРПОРЕЙШН (JP)

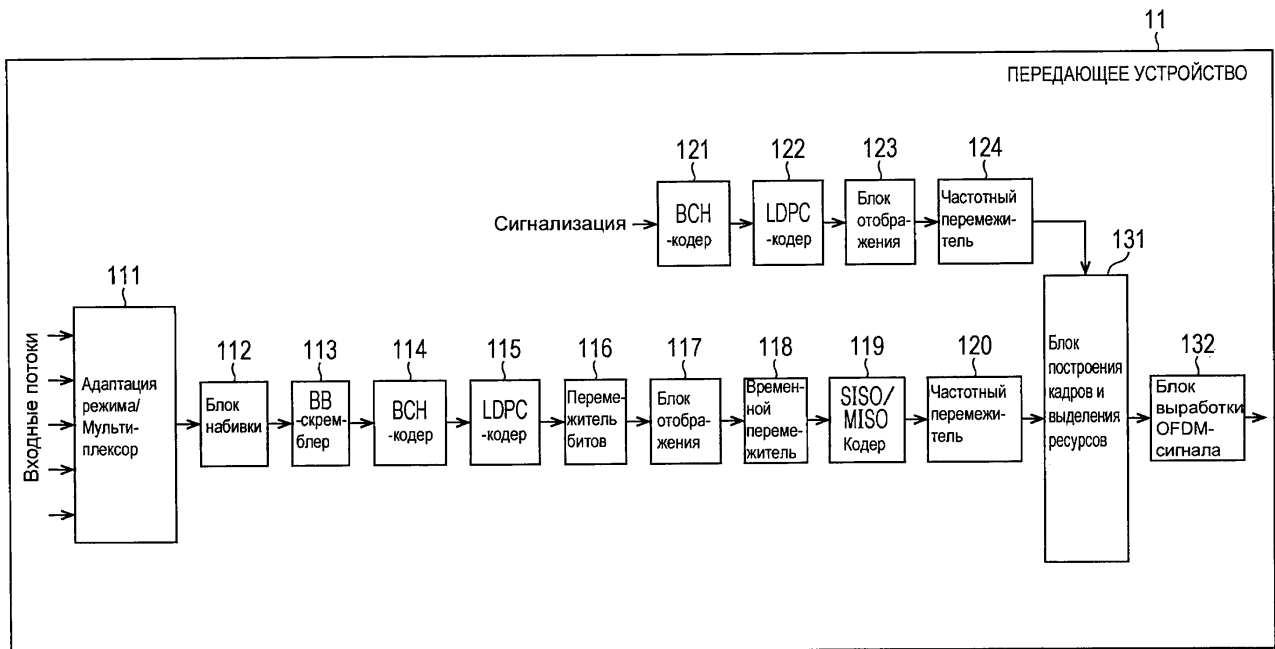
(56) Список документов, цитированных в отчете
о поиске: EP 2560310 A1, 20.02.2013. EP
2214320 A1, 04.08.2010. US 2010/0275100 A1,
28.10.2010. RU 2443053 C2, 20.02.2012. RU
2369008 C2, 27.09.2009.

(54) УСТРОЙСТВО ОБРАБОТКИ ДАННЫХ И СПОСОБ ОБРАБОТКИ ДАННЫХ

(57) Реферат:

Группа изобретений относится к области обработки данных и может быть использована для кодирования/декодирования с использованием LDPC-кода. Техническим результатом является уменьшение частоты ошибок при передаче данных. LDPC-код имеет длину кода 16200 битов и кодовую скорость 12/15. LDPC-код включает в себя информационный бит и бит четности, и матрица Н контроля четности выполнена с информационной частью

матрицы, соответствующей информационному биту LDPC-кода и части матрицы четности, соответствующей биту четности. Информационная часть матрицы Н контроля четности представлена таблицей начальных значений матрицы контроля четности, представляющих собой позицию элемента 1 в информационной части матрицы с интервалом 360 столбцов. 5 н. и 15 з.п. ф-лы, 60 ил.



Фиг. 8



FEDERAL SERVICE
FOR INTELLECTUAL PROPERTY

(12) **ABSTRACT OF INVENTION**

(52) CPC

H03M 13/036 (2006.01); *H03M 13/1165* (2006.01); *H03M 13/118* (2006.01); *H03M 13/255* (2006.01); *H03M 13/2707* (2006.01); *H03M 13/271* (2006.01); *H03M 13/2906* (2006.01); *H03M 13/356* (2006.01); *H03M 13/616* (2006.01); *H03M 13/6552* (2006.01); *H04L 1/0057* (2006.01); *H04L 1/0071* (2006.01)

(21)(22) Application: **2015103856, 03.06.2014**(24) Effective date for property rights:
03.06.2014Registration date:
06.06.2018

Priority:

(30) Convention priority:
12.06.2013 JP 2013-124187(43) Application published: **27.08.2016** Bull. № 24(45) Date of publication: **06.06.2018** Bull. № 16(85) Commencement of national phase: **05.02.2015**(86) PCT application:
JP 2014/064672 (03.06.2014)(87) PCT publication:
WO 2014/199865 (18.12.2014)Mail address:
109012, Moskva, ul. Ilinka, 5/2, OOO "Soyuzpatent"

(72) Inventor(s):

**SINOKHARA Yudzi (JP),
YAMAMOTO Makiko (JP)**

(73) Proprietor(s):

SONI KORPOREJSHN (JP)(54) **DATA PROCESSING APPARATUS AND DATA PROCESSING METHOD**

(57) Abstract:

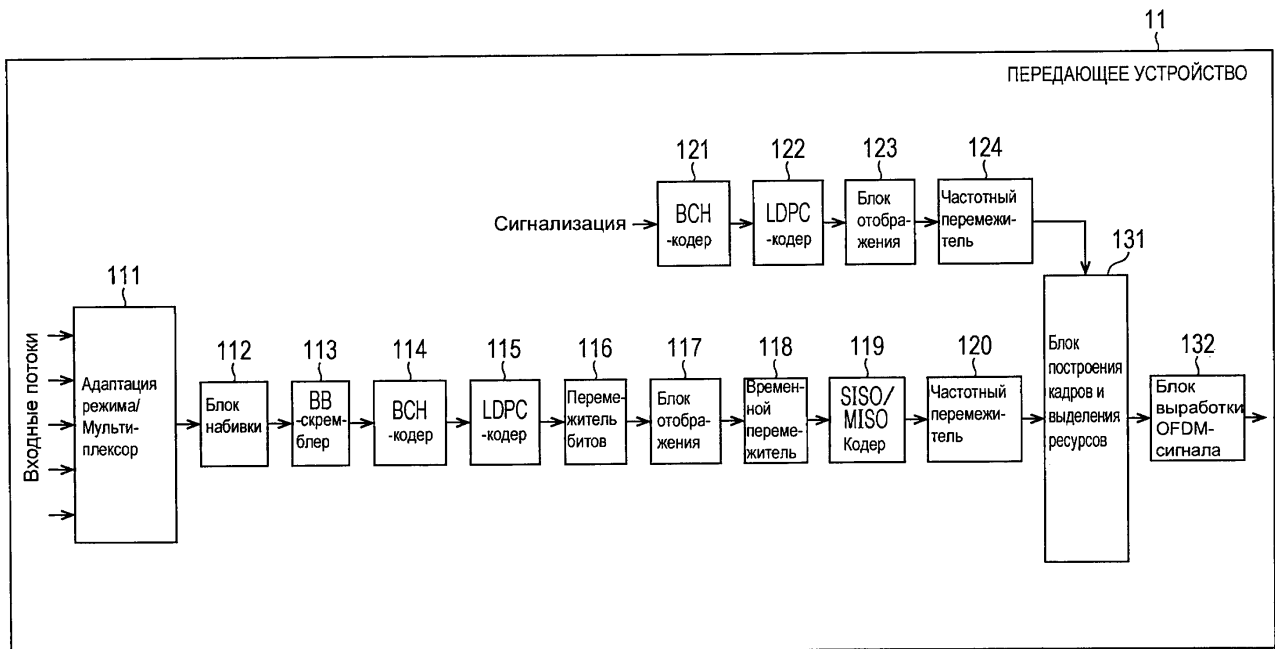
FIELD: information technology.

SUBSTANCE: group of inventions refers to the field of data processing and can be used for encoding / decoding using the LDPC code. LDPC code has a code length of 16,200 bits and a code rate of 12/15. LDPC code includes an information bit and a parity bit, and the parity check matrix H is executed with the information part of the matrix, corresponding to the information bit of the LDPC code and a part of the

parity matrix, corresponding to the parity bit. Informational part of the parity check matrix H is represented by a table of initial values of the parity check matrix representing the position of element 1 in the informational part of the matrix with an interval of 360 columns.

EFFECT: technical result is a reduction in the error rate in data transmission.

20 cl, 60 dwg



Фиг. 8

Область техники, к которой относится изобретение

Данная технология относится к устройству обработки данных и способу обработки данных и, в частности, относится к устройству обработки данных и способу обработки данных, которые способны обеспечить хорошее качество связи при передаче данных с использованием, например, LDPC-кода.

Уровень техники

Код разреженного контроля четности (LDPC) имеет высокую способность коррекции ошибки и в последние годы широко использовался в схеме передачи, включающей в себя цифровое ширококовещание, такое как, например, европейский стандарт цифрового телевизионного вещания (DVB)-S.2, DVB-T.2 и DVB-C.2 (например, смотри NPL 1).

Последние исследования показывают, что при увеличении длины кода LDPC-код имеет характеристики, близкие к пределу Шеннона и аналогичные турбокодам или т.п. В дополнение к этому, поскольку LDPC-код имеет свойства минимального расстояния, пропорционального длине кода, LDPC-код имеет преимущество в том, что характеристика вероятности блочной ошибки является хорошей, и так называемое явление минимального уровня ошибок, которое наблюдается как свойство декодирования турбо-кода или т.п., почти не возникает в виде свойства.

Перечень цитируемой литературы

Непатентная литература

NPL 1: DVB-S.2: ETSI EN 302307 V1.2.1 (2009-08)

Сущность изобретения

Техническая задача

При передаче данных с использованием, например, LDPC-кода, LDPC-код формируется в символе квадратурной модуляции (цифровой модуляции), такой как квадратурная фазовая манипуляция (QPSK) (символизируется), и символ передается путем отображения в сигнальную точку квадратурной модуляции.

Такая передача данных с использованием LDPC-кода становится широко распространенной во всем мире, и остается востребованным обеспечение хорошего качества связи.

Данная технология была выполнена с учетом таких обстоятельств, и задача изобретения состоит в том, чтобы обеспечить хорошее качество связи при передаче данных с использованием LDPC-кода.

Решение технической задачи

Первое устройство обработки данных (способ обработки данных) данной технологии включает в себя блок кодирования (этап кодирования), который кодирует информационный бит в LDPC-код, имеющий длину кода 16200 битов и кодовую скорость 12/15, на основании матрицы контроля четности кода разреженного контроля четности (LDPC), в которой LDPC-код включает в себя информационный бит и бит четности, матрица контроля четности включает в себя информационную часть матрицы, соответствующую информационному биту, и часть матрицы четности, соответствующую биту четности, информационная часть матрицы представлена таблицей начальных значений матрицы контроля четности, таблица начальных значений матрицы контроля четности представляет собой таблицу, представляющую позиции элементов 1 в информационной части матрицы с интервалом 360 столбцов, и представляет собой

3 394 1014 1214 1361 1477 1534 1660 1856 2745 2987 2991 3124 3155
59 136 528 781 803 928 1293 1489 1944 2041 2200 2613 2690 2847
155 245 311 621 1114 1269 1281 1783 1995 2047 2672 2803 2885 3014
79 870 974 1326 1449 1531 2077 2317 2467 2627 2811 3083 3101 3132
4 582 660 902 1048 1482 1697 1744 1928 2628 2699 2728 3045 3104
175 395 429 1027 1061 1068 1154 1168 1175 2147 2359 2376 2613 2682
1388 2241 3118 3148
143 506 2067 3148
1594 2217 2705
398 988 2551
1149 2588 2654
678 2844 3115
1508 1547 1954
1199 1267 1710
2589 3163 3207
1 2583 2974
2766 2897 3166
929 1823 2742
1113 3007 3239
1753 2478 3127

0 509 1811
 1672 2646 2984
 965 1462 3230
 5 3 1077 2917
 1183 1316 1662
 968 1593 3239
 10 64 1996 2226
 1442 2058 3181
 513 973 1058
 1263 3185 3229
 15 681 1394 3017
 419 2853 3217
 3 2404 3175
 2417 2792 2854
 20 1879 2940 3235
 647 1704 3060.

В первом устройстве обработки данных/способе обработки данных, информационный
 бит кодируется в LDPC-код, имеющий длину кода 16200 битов и кодовую скорость 12/
 25 15, на основании матрицы контроля четности кода разреженного контроля четности
 (LDPC). LDPC-код включает в себя информационный бит и бит четности, матрица
 контроля четности включает в себя информационную часть матрицы, соответствующую
 информационному биту, и часть матрицы четности, соответствующую биту четности,
 информационная часть матрицы представлена таблицей начальных значений матрицы
 30 контроля четности, таблица начальных значений матрицы контроля четности
 представляет собой таблицу, представляющую позиции элементов 1 в информационной
 части матрицы с интервалом 360 столбцов, и представляет собой

3 394 1014 1214 1361 1477 1534 1660 1856 2745 2987 2991 3124 3155
 35 59 136 528 781 803 928 1293 1489 1944 2041 2200 2613 2690 2847
 155 245 311 621 1114 1269 1281 1783 1995 2047 2672 2803 2885 3014
 79 870 974 1326 1449 1531 2077 2317 2467 2627 2811 3083 3101 3132
 4 582 660 902 1048 1482 1697 1744 1928 2628 2699 2728 3045 3104
 40 175 395 429 1027 1061 1068 1154 1168 1175 2147 2359 2376 2613 2682
 1388 2241 3118 3148
 143 506 2067 3148
 45 1594 2217 2705

398 988 2551

1149 2588 2654

678 2844 3115

5

1508 1547 1954

1199 1267 1710

2589 3163 3207

10

1 2583 2974

2766 2897 3166

929 1823 2742

1113 3007 3239

15

1753 2478 3127

0 509 1811

1672 2646 2984

20

965 1462 3230

3 1077 2917

1183 1316 1662

968 1593 3239

25

64 1996 2226

1442 2058 3181

513 973 1058

1263 3185 3229

30

681 1394 3017

419 2853 3217

3 2404 3175

35

2417 2792 2854

1879 2940 3235

647 1704 3060.

Второе устройство обработки данных (способ обработки данных) данной технологии включает в себя блок декодирования (этап декодирования), который декодирует LDPC-код, полученный из данных, которые передаются из передающего устройства, включающего в себя блок кодирования, который кодирует информационный бит в LDPC-код, имеющий длину кода 16200 битов и кодовую скорость 12/15, на основании матрицы контроля четности кода разреженного контроля четности (LDPC), в которой LDPC-код включает в себя информационный бит и бит четности, матрица контроля четности включает в себя информационную часть матрицы, соответствующую информационному биту, и часть матрицы четности, соответствующую биту четности, информационная часть матрицы представлена таблицей начальных значений матрицы

контроля четности, таблица начальных значений матрицы контроля четности представляет собой таблицу, представляющую позиции элементов 1 в информационной части матрицы с интервалом 360 столбцов, и представляет собой

5	3 394 1014 1214 1361 1477 1534 1660 1856 2745 2987 2991 3124 3155
	59 136 528 781 803 928 1293 1489 1944 2041 2200 2613 2690 2847
	155 245 311 621 1114 1269 1281 1783 1995 2047 2672 2803 2885 3014
	79 870 974 1326 1449 1531 2077 2317 2467 2627 2811 3083 3101 3132
10	4 582 660 902 1048 1482 1697 1744 1928 2628 2699 2728 3045 3104
	175 395 429 1027 1061 1068 1154 1168 1175 2147 2359 2376 2613 2682
	1388 2241 3118 3148
	143 506 2067 3148
15	1594 2217 2705
	398 988 2551
	1149 2588 2654
20	678 2844 3115
	1508 1547 1954
	1199 1267 1710
	2589 3163 3207
25	1 2583 2974
	2766 2897 3166
	929 1823 2742
30	1113 3007 3239
	1753 2478 3127
	0 509 1811
	1672 2646 2984
35	965 1462 3230
	3 1077 2917
	1183 1316 1662
	968 1593 3239
40	64 1996 2226
	1442 2058 3181
	513 973 1058
45	1263 3185 3229

681 1394 3017

419 2853 3217

3 2404 3175

5 2417 2792 2854

1879 2940 3235

647 1704 3060.

10 Во втором устройстве обработки данных/способе обработки данных декодируется LDPC-код, который получается из данных, переданных из передающего устройства, причем передающее устройство включает в себя блок кодирования (этап кодирования) который кодирует информационный бит в LDPC-код, имеющий длину кода 16200 битов и кодовую скорость 12/15, на основании матрицы контроля четности кода разреженного контроля четности (LDPC), в которой LDPC-код включает в себя информационный бит

15 и бит четности, матрица контроля четности включает в себя информационную часть матрицы, соответствующую информационному биту, и часть матрицы четности, соответствующую биту четности, информационная часть матрицы представлена

таблицей начальных значений матрицы контроля четности, таблица начальных значений матрицы контроля четности представляет собой таблицу, представляющую позиции

20 элементов 1 в информационной части матрицы с интервалом 360 столбцов, и представляет собой

3 394 1014 1214 1361 1477 1534 1660 1856 2745 2987 2991 3124 3155

59 136 528 781 803 928 1293 1489 1944 2041 2200 2613 2690 2847

25 155 245 311 621 1114 1269 1281 1783 1995 2047 2672 2803 2885 3014

79 870 974 1326 1449 1531 2077 2317 2467 2627 2811 3083 3101 3132

4 582 660 902 1048 1482 1697 1744 1928 2628 2699 2728 3045 3104

30 175 395 429 1027 1061 1068 1154 1168 1175 2147 2359 2376 2613 2682

1388 2241 3118 3148

143 506 2067 3148

1594 2217 2705

35 398 988 2551

1149 2588 2654

678 2844 3115

1508 1547 1954

40 1199 1267 1710

2589 3163 3207

1 2583 2974

45 2766 2897 3166

929 1823 2742

1113 3007 3239

1753 2478 3127

5

0 509 1811

1672 2646 2984

965 1462 3230

10

3 1077 2917

1183 1316 1662

968 1593 3239

64 1996 2226

15

1442 2058 3181

513 973 1058

1263 3185 3229

20

681 1394 3017

419 2853 3217

3 2404 3175

2417 2792 2854

25

1879 2940 3235

647 1704 3060.

Третье устройство обработки данных данной технологии включает в себя блок декодирования, который декодирует LDPC-код, имеющий длину кода 16200 битов и кодовую скорость 12/15, на основании матрицы контроля четности кода разреженного контроля четности (LDPC), в которой LDPC-код включает в себя информационный бит и бит четности, матрица контроля четности включает в себя информационную часть матрицы, соответствующую информационному биту, и часть матрицы четности, соответствующую биту четности, информационная часть матрицы представлена таблицей начальных значений матрицы контроля четности, таблица начальных значений матрицы контроля четности представляет собой таблицу, представляющую позиции элементов 1 в информационной части матрицы с интервалом 360 столбцов, и представляет собой

3 394 1014 1214 1361 1477 1534 1660 1856 2745 2987 2991 3124 3155

40

59 136 528 781 803 928 1293 1489 1944 2041 2200 2613 2690 2847

155 245 311 621 1114 1269 1281 1783 1995 2047 2672 2803 2885 3014

79 870 974 1326 1449 1531 2077 2317 2467 2627 2811 3083 3101 3132

45

4 582 660 902 1048 1482 1697 1744 1928 2628 2699 2728 3045 3104

175 395 429 1027 1061 1068 1154 1168 1175 2147 2359 2376 2613 2682

1388 2241 3118 3148

143 506 2067 3148

1594 2217 2705

5

398 988 2551

1149 2588 2654

678 2844 3115

10

1508 1547 1954

1199 1267 1710

2589 3163 3207

1 2583 2974

15

2766 2897 3166

929 1823 2742

1113 3007 3239

1753 2478 3127

20

0 509 1811

1672 2646 2984

965 1462 3230

25

3 1077 2917

1183 1316 1662

968 1593 3239

64 1996 2226

30

1442 2058 3181

513 973 1058

1263 3185 3229

35

681 1394 3017

419 2853 3217

3 2404 3175

2417 2792 2854

40

1879 2940 3235

647 1704 3060.

В третьем устройстве обработки данных LDPC-код, имеющий длину кода 16200 битов и кодовую скорость 12/15, декодируется на основании матрицы контроля четности кода разреженного контроля четности (LDPC). LDPC-код включает в себя

информационный бит и бит четности, матрица контроля четности включает в себя информационную часть матрицы, соответствующую информационному биту, и часть матрицы четности, соответствующую биту четности, информационная часть матрицы представлена таблицей начальных значений матрицы контроля четности, таблица

начальных значений матрицы контроля четности представляет собой таблицу, представляющую позиции элементов 1 в информационной части матрицы с интервалом 360 столбцов, и представляет собой

5	3 394 1014 1214 1361 1477 1534 1660 1856 2745 2987 2991 3124 3155
	59 136 528 781 803 928 1293 1489 1944 2041 2200 2613 2690 2847
	155 245 311 621 1114 1269 1281 1783 1995 2047 2672 2803 2885 3014
	79 870 974 1326 1449 1531 2077 2317 2467 2627 2811 3083 3101 3132
10	4 582 660 902 1048 1482 1697 1744 1928 2628 2699 2728 3045 3104
	175 395 429 1027 1061 1068 1154 1168 1175 2147 2359 2376 2613 2682
	1388 2241 3118 3148
	143 506 2067 3148
15	1594 2217 2705
	398 988 2551
	1149 2588 2654
20	678 2844 3115
	1508 1547 1954
	1199 1267 1710
	2589 3163 3207
25	1 2583 2974
	2766 2897 3166
	929 1823 2742
30	1113 3007 3239
	1753 2478 3127
	0 509 1811
	1672 2646 2984
35	965 1462 3230
	3 1077 2917
	1183 1316 1662
	968 1593 3239
40	64 1996 2226
	1442 2058 3181
	513 973 1058
45	1263 3185 3229

681 1394 3017

419 2853 3217

3 2404 3175

5 2417 2792 2854

1879 2940 3235

647 1704 3060.

10 В дополнение, устройство обработки данных может представлять собой независимое устройство или может представлять собой внутренний блок, конфигурирующий одно устройство.

Преимущественные эффекты изобретения

Согласно настоящему изобретению можно обеспечить хорошее качество связи при передаче данных с использованием LDPC-кода.

15 В дополнение к этому, эффекты, описанные в данном документе, являются только иллюстративными, при этом эффекты данной технологии не предназначены для ограничения эффектов, которые описаны в данном документе, и могут существовать дополнительные эффекты.

Краткое описание чертежей

20 Фиг. 1 - схема, изображающая матрицу Н контроля четности LDPC-кода.

Фиг. 2 - блок-схема последовательности операций, иллюстрирующая процедуру декодирования LDPC-кода.

Фиг. 3 - схема, иллюстрирующая пример матрицы контроля четности LDPC-кода.

Фиг. 4 - схема, иллюстрирующая график Таннера матрицы контроля четности.

25 Фиг. 5 - схема, иллюстрирующая переменный узел.

Фиг. 6 - схема, иллюстрирующая контрольный узел.

Фиг. 7 - схема, иллюстрирующая пример конфигурации варианта осуществления передающей системы, в которой применяется данная технология.

Фиг. 8 - схема, иллюстрирующая пример конфигурации передающего устройства 11.

30 Фиг. 9 - блок-схема, иллюстрирующая пример конфигурации перемежителя 116 битов.

Фиг. 10 - схема, иллюстрирующая матрица контроля четности.

Фиг. 11 - схема, иллюстрирующая матрицу четности.

35 Фиг. 12 - схема, иллюстрирующая матрицу контроля четности LDPC-кода, который определен в стандарте DVB-S.2.

Фиг. 13 - схема, изображающая матрицу контроля четности LDPC-кода, который определен в стандарте DVB-S.2.

Фиг. 14 - схема, иллюстрирующая размещение сигнальных точек 16QAM.

Фиг. 15 - схема, иллюстрирующая размещение сигнальных точек 64QAM.

40 Фиг. 16 - схема, иллюстрирующая размещение сигнальных точек 64QAM.

Фиг. 17 - схема, иллюстрирующая размещение сигнальных точек 64QAM.

Фиг. 18 - схема, иллюстрирующая размещение сигнальных точек, которое определено в стандарте DVB-S.2.

45 Фиг. 19 - схема, иллюстрирующая размещение сигнальных точек, которое определено в стандарте DVB-S.2.

Фиг. 20 - схема, иллюстрирующая размещение сигнальных точек, которое определено в стандарте DVB-S.2.

Фиг. 21 - схема, иллюстрирующая размещение сигнальных точек, которое определено

в стандарте DVB-S.2.

Фиг. 22 - схема, изображающая процесс, выполняемый демультимплексором 25.

Фиг. 23 - схема, изображающая процесс, выполняемый демультимплексором 25.

Фиг. 24 - схема, иллюстрирующая график Таннера для декодирования LDPC-кода.

5 Фиг. 25 - схема, иллюстрирующая матрицу H_T четности имеющий ступенчатую структуру и график Таннера, соответствующий матрице H_T четности.

Фиг. 26 - схема, иллюстрирующая матрицу H_T четности матрицы H контроля четности, соответствующей LDPC-коду после перемежения четности.

10 Фиг. 27 - схема, иллюстрирующая матрицу контроля четности преобразования.

Фиг. 28 - схема, изображающая процесс, выполняемый перемежителем 24 со скручиванием столбцов.

Фиг. 29 - схема, иллюстрирующая число столбцов и адрес позиции начала записи памяти 31, которая требуется для перемежения со скручиванием столбцов.

15 Фиг. 30 - схема, иллюстрирующая число столбцов и адрес позиции начала записи памяти 31, которая требуется для перемежения со скручиванием столбцов.

Фиг. 31 - блок-схема последовательности операций, иллюстрирующая процесс, выполняемый в перемежителе 116 битов и устройстве 117 отображения.

20 Фиг. 32 - схема, иллюстрирующая модель канала связи, используемую при моделировании.

Фиг. 33 - схема, иллюстрирующая зависимость между частотой появления ошибок, полученной путем моделирования, и доплеровской частотой f_d фазового дрожания.

Фиг. 34 - схема, иллюстрирующая зависимость между частотой появления ошибок, полученной путем модуляции, и доплеровской частотой f_d фазового дрожания.

25 Фиг. 35 - блок-схема, иллюстрирующая пример конфигурации LDPC-кодера 115.

Фиг. 36 - блок-схема последовательности операций, иллюстрирующая процесс LDPC-кодера 115.

Фиг. 37 - схема, иллюстрирующая пример таблицы начальных значений матрицы контроля четности, в которой кодовая скорость равна $1/4$, и длина кода равна 16200.

30 Фиг. 38 - схема, изображающая способ получения матрицы H контроля четности из таблицы начальных значений матрицы контроля четности.

Фиг. 39 - схема, иллюстрирующая пример таблицы начальных значений матрицы контроля четности кода $16k$ с $r=12/15$.

35 Фиг. 40 - схема, иллюстрирующая пример график Таннера ансамбля степенной последовательности, в которой вес столбца равен 3, и вес строки равен 6.

Фиг. 41 - схема, иллюстрирующая пример графика Таннера ансамбля многореберного типа.

Фиг. 42 - схема, иллюстрирующая минимальную длину цикла и порог производительности матрицы контроля четности кода $16k$ с $r=12/15$.

40 Фиг. 43 - схема, изображающая матрицу контроля четности кода $16k$ с $r=12/15$.

Фиг. 44 - схема, изображающая матрицу контроля четности кода $16k$ с $r=12/15$.

Фиг. 45 - схема, иллюстрирующая результат моделирования измерения BER/FER.

Фиг. 46 - блок-схема, иллюстрирующая пример конфигурации приемного устройства 12.

45 Фиг. 47 - блок-схема, иллюстрирующая пример конфигурации деперемежителя 165 битов.

Фиг. 48 - блок-схема последовательности операций, иллюстрирующая процесс, выполняемый устройством 164 обратного отображения, деперемежителем 165 битов

и LDPC-декодером 166.

Фиг. 49 - схема, иллюстрирующая пример матрицы контроля четности LDPC-кода.

Фиг. 50 - схема, иллюстрирующая матрицу (матрицу контроля четности преобразования), полученную путем выполнения перестановки строк и перестановки столбцов в отношении матрицы контроля четности.

Фиг. 51 - схема, иллюстрирующая матрицу контроля четности преобразования, которая разделена на 5×5 блоков.

Фиг. 52 - блок-схема, иллюстрирующая пример конфигурации устройства декодирования, которое выполняет вычисление узла в группах P.

Фиг. 53 - блок-схема, иллюстрирующая пример конфигурации LDPC-декодера 166.

Фиг. 54 - схема, изображающая процесс, выполняемый мультиплексором 54, конфигурирующим демультиплекситель 165 битов.

Фиг. 55 - схема, изображающая процесс, выполняемый демультиплексителем 55 со скручиванием столбцов.

Фиг. 56 - блок-схема, иллюстрирующая другой пример конфигурации демультиплексителя 165 битов.

Фиг. 57 - блок-схема, иллюстрирующая первый пример конфигурации приемной системы, в которой может применяться приемное устройство 12.

Фиг. 58 - блок-схема, иллюстрирующая второй пример конфигурации приемной системы, в которой может применяться приемное устройство 12.

Фиг. 59 - блок-схема, иллюстрирующая третий пример конфигурации приемной системы, в которой может применяться приемное устройство 12.

Фиг. 60 - блок-схема, иллюстрирующая пример конфигурации варианта осуществления компьютера, в котором применяется настоящая технология.

Подробное описание изобретения

Описание вариантов осуществления

Далее перед описанием вариантов осуществления данной технологии будет описан LDPC-код.

LDPC-код

Кроме того, LDPC-код представляет собой линейный код и не должен быть двоичным, но, в данном случае, описание будет приведено исходя из предположения того, что LDPC-код является двоичным.

Наиболее существенный признак LDPC-кода состоит в том, что матрица контроля четности, определяющая LDPC-код, является разреженной матрицей. В данном случае, разреженная матрица представляет собой матрицу, в которой число "1" элементов матрицы является очень маленьким (в основном элементы матрицы равны 0).

На фиг. 1 показана схема, иллюстрирующая матрицу H контроля четности LDPC-кода.

В матрице H контроля четности на фиг. 1, вес каждого столбца (вес столбца) (число "1") (вес) установлен на "3", и вес каждой строки (вес строки) установлен на "6".

При кодировании с использованием LDPC-кода (LDPC-кодирования), например, порождающая матрицы G, вырабатывается на основании матрицы H контроля четности, и кодовое слово (LDPC-код), вырабатывается путем умножения порождающей матрицы G на двоичный информационный бит.

В частности, устройство кодирования, которое выполняет LDPC-кодирование, сначала вычисляет порождающую матрицу G, в которой между матрицей H контроля четности и ее транспонированной матрицей H^T устанавливается уравнение $GH^T=0$. В данном случае, когда порождающая матрица G представляет собой матрицу размером

$K \times N$, устройство кодирования умножает битовую последовательность (вектор u) информационного бита, сформированного из K битов, с порождающей матрицей G для того, чтобы выработать кодовое слово $s (=uG)$, сформированное из N битов. Кодовое слово (LDPC-код), выработанное с помощью устройства кодирования, принимается

на приемной стороне через предварительно определенной канал связи. Декодирование LDPC-кода представляет собой алгоритм, который предложен и упоминается как вероятностное кодирование Галлагера, и алгоритм можно выполнить с помощью алгоритма передачи сообщений путем распространения по так называемому графу Таннера, который выполнен с переменным узлом (который также упоминается как узел сообщения) и проверочным узлом. Здесь и далее переменный узел и контрольный узел упоминаются при необходимости просто как узел.

На фиг. 2 показана блок-схема последовательности операций, иллюстрирующая процедуру декодирования LDPC-кода.

Здесь и далее при необходимости действительное число (принятое LLR), которое получено путем правдоподобия "0" значения кодового бита LDPC-кода (кодовое слово 1), принятое на приемной стороне в качестве логарифмического отношения правдоподобия, упоминается как принятое значение u_{0i} . Кроме того, предполагается, что сообщение, выходящее из проверочного узла, будет представлять собой u_j , и предполагается, что сообщение, выходящее из переменного узла, будет представлять собой v_i .

Сначала при декодировании LDPC-кода, как показано на фиг. 2, на этапе S11 принимается LDPC-код, сообщение u_j (сообщение проверочного узла) устанавливается в "0", переменная k , которая представляет собой целое число в качестве счетчика итерационного процесса, устанавливается в "0", и процесс переходит на этапе S12. На этапе S12 сообщение v_i (сообщение переменного узла) получается путем выполнения вычисления (вычисления переменного узла), показанного в уравнении (1), на основании принятого значения u_{0i} , полученного путем приема LDPC-кода, и сообщение u_j получается путем выполнения вычисления (вычисления проверочного узла), показанного в уравнении (2), на основании сообщения v_i .

[Формула 1]

$$v_i = u_{0i} + \sum_{j=1}^{d_v-1} u_j \quad \dots(1)$$

[Формула 2]

$$\tanh\left(\frac{u_j}{2}\right) = \prod_{i=1}^{d_c-1} \tanh\left(\frac{v_i}{2}\right) \quad \dots(2)$$

В данном случае, d_v и d_c в уравнении (1) и в уравнении (2) представляют собой, соответственно, параметры, показывающие число "1" в вертикальном направлении (в направлении столбца) и в горизонтальном направлении (в направлении строки) матрицы H контроля четности, которую можно выбрать произвольным образом. Например, в случае LDPC-кода, показанного на фиг. 1, в котором вес столбца равен 3, и вес строки равен 6 (LDPC-код (3, 6)), он устанавливается с параметрами $d_v=3$ и $d_c=6$.

В дополнение к этому, при вычислении переменного узла из уравнения (1) и

вычислении проверочного узла из уравнения (2) сообщение, которое вводится с края (линия, соединяющая переменный узел и проверочный узел), и из которого должно выводиться сообщение, не используется в качестве цели вычисления, и таким образом, диапазон вычисления равен $1 - D_v - 1$ или $1 - D_c - 1$. Кроме того, вычисление проверочного узла из уравнения (2) выполняется фактически путем предварительного создания таблицы из функции $R(v_1, v_2)$, показанной в уравнении (3), которая определена в качестве одного выхода для двух входов v_1 и v_2 и непрерывно (рекурсивно) использует таблицу, как показано в уравнении (4).

[Формула 3]

$$x = 2 \tanh^{-1} \{ \tanh(v_1/2) \tanh(v_2/2) \} = R(v_1, v_2) \dots (3)$$

[Формула 4]

$$u_j = R(v_1, R(v_2, R(v_3, \dots R(v_{d_c-2}, v_{d_c-1}))) \dots (4)$$

На этапе S12 переменная k получает приращение на "1", и процесс переходит на этап S13. На этапе S13 определяется, больше или нет переменная k , чем предварительно определенное итерационное число C декодирования. На этапе S13, если определено, что переменная k не будет больше, чем C , процесс возвращается на этап S12, и повторяется тот же самый процесс.

Кроме того, на этапе S13, если определено, что переменная k больше, чем C , процесс переходит на этап S14, сообщение v_i , в качестве результата декодирования, который будет окончательно выводиться, получается и выводится путем выполнения вычисления, показанного в уравнении (5), и процесс декодирования LDPC-кода заканчивается.

[Формула 5]

$$v_i = u_{0i} + \sum_{j=1}^{d_v} u_j \dots (5)$$

В данном случае, вычисление уравнения (5) отличается от вычисления переменного узла из уравнения (1) и выполняется с использованием сообщения u_j со всех ребер, связанных с переменным узлом.

На фиг. 3 показана схема, иллюстрирующая пример матрицы H контроля четности с LDPC-кодом (3, 6) (кодовая скорость равна $1/2$, длина кода равна 12).

В матрице H контроля четности, показанной на фиг. 3, вес столбца установлен на 3, и вес строки установлен на 6, аналогично фиг. 1.

На фиг. 4 показана схема, иллюстрирующая график Таннера матрицы H контроля четности (фиг. 3).

В данном случае, на фиг. 4, проверочные узлы показаны знаком плюс "+", и переменные узлы показаны знаком равно "=". Проверочные узлы и переменные узлы, соответственно, соответствуют строкам и столбцам матрицы H контроля четности. Соединительные линии между проверочными узлами и переменными узлами представляют собой ребро, и ребра соответствуют элементам "1" в матрице контроля четности.

Другими словами, если элемент в j -ой строке и i -ом столбце матрицы контроля четности равен 1 (фиг. 4), i -ый переменный узел (узел "=") сверху и j -ый проверочный узел (узел "+") сверху соединены с помощью ребра. Ребро представляет собой то, что

кодовый бит, соответствующий переменному узлу, имеет ограничения, соответствующие проверочному узлу.

В алгоритме произведения суммы, который представляет собой способ декодирования LDPC-кода, вычисление переменного узла и вычисление проверочного узла выполняются

На фиг. 5 показана схема, иллюстрирующая вычисление переменного узла, которое выполняется в переменном узле.

В переменном узле сообщение v_i , соответствующее ребру, подлежащему вычислению, получается путем вычисления переменного узла из уравнения (1) с использованием сообщений u_1 и u_2 из оставшихся ребер, которые соединены с переменным узлом, и принятого значения u_{0i} . Сообщения, соответствующие другим ребрам, получаются таким же образом.

На фиг. 6 показана схема, иллюстрирующая вычисление проверочного узла, которая выполняется в проверочном узле.

В данном случае, вычисление проверочного узла из уравнения (2) можно переписать в виде уравнения (6) с использованием функциональной зависимости уравнения $a \times b = \exp\{\ln(|a|) + \ln(|b|)\} \times \text{sign}(a) \times \text{sign}(b)$. В данном случае, $\text{sign}(x)$ равен 1 при $x \geq 0$, и $\text{sign}(x)$ равен -1 при $x < 0$.

[Формула 6]

$$\begin{aligned} u_j &= 2 \tanh^{-1} \left(\prod_{i=1}^{d_c-1} \tanh \left(\frac{v_i}{2} \right) \right) \\ &= 2 \tanh^{-1} \left[\exp \left\{ \sum_{i=1}^{d_c-1} \ln \left(\left| \tanh \left(\frac{v_i}{2} \right) \right| \right) \right\} \times \prod_{i=1}^{d_c-1} \text{sign} \left(\tanh \left(\frac{v_i}{2} \right) \right) \right] \\ &= 2 \tanh^{-1} \left[\exp \left\{ - \left(\sum_{i=1}^{d_c-1} - \ln \left(\tanh \left(\frac{|v_i|}{2} \right) \right) \right) \right\} \times \prod_{i=1}^{d_c-1} \text{sign}(v_i) \right] \end{aligned} \quad \dots(6)$$

Когда $x \geq 0$, если функция $\phi(x)$ определена в виде уравнения $\phi(x) = \ln(\tanh(x/2))$, устанавливается уравнение $\phi^{-1}(x) = 2 \tanh^{-1}(e^{-x})$ таким образом, что уравнение (6) можно преобразовать в уравнение (7).

[Формула 7]

$$u_j = \phi^{-1} \left(\sum_{i=1}^{d_c-1} \phi(|v_i|) \right) \times \prod_{i=1}^{d_c-1} \text{sign}(v_i) \quad \dots(7)$$

В проверочном узле вычисление проверочного узла из уравнения (2) выполняется в соответствии с уравнением (7).

Другими словами, в проверочном узле, как показано на фиг. 6, сообщение U_j , соответствующее ребру, которое будет вычисляться, получается с помощью вычисления проверочного узла из уравнения (7) с использованием сообщений v_1, v_2, v_3, v_4 , и v_5 из оставшихся ребер, которые соединены с проверочным узлом. Сообщения, соответствующие другим ребрам, получаются таким же образом.

В дополнение, функция $\phi(x)$ из уравнения (7) может быть представлена в виде уравнения $\phi(x) = \ln((e^x + 1)/(e^x - 1))$, когда $x > 0$, $\phi(x) = -\phi^{-1}(x)$. Когда функции $\phi(x)$ и $\phi^{-1}(x)$ реализованы в виде обратных средств, в некоторых случаях функции можно реализовать с использованием таблиц поиска (LUT), но LUT являются одинаковыми.

Пример конфигурации передающей системы, в которой применяется настоящая технология

На фиг. 7 показана схема, иллюстрирующая пример конфигурации варианта осуществления передающей системы, в которой применяется настоящая технология (система относится к таким системам, в которых множество устройств собирается логическим образом, и не имеет значение то, находится ли устройство, составляющее часть целого, в одном и том же корпусе).

На фиг. 7 передающая система включает в себя передающее устройство 11 и приемное устройство 12.

Передающее устройство 11 выполняет передачу (широковещание) (отправку), например, программ и т.п. телевизионного вещания. Другими словами, передающее устройство 11 кодирует, например, целевые данные, которые представляют собой цель передачи, такие как данные изображения и аудиоданные в виде программы в LDPC-код, и передает кодированные целевые данные через канал 13 связи, такой как, например, спутниковые линии, наземные волны, кабели (проводные линии).

Приемное устройство 12 принимает LDPC-код, который передается из передающего устройства 11 через канал 13 связи, декодирует LDPC-код в целевые данные и выводит данные.

В данном случае, известно, что LDPC-код, который используется в системе передачи, показанной на фиг. 7, проявляет в значительной степени высокую пропускную способность в канале связи с аддитивным белым гауссовым шумом (AWGN).

Между тем, канал 13 связи может вырабатывать пакетную ошибку или стирание пакета. Например, в частности, когда канал 13 связи представляет собой наземные волны, в системе ортогонального мультиплексирования с частотным разделением (OFDM) в многолучевой среде, в которой отношение желательного сигнала к нежелательному (D/U) равно 0 дБ (мощность нежелательного сигнала (эхо-сигнала) равна мощности желательного сигнала (сигнала, распространяющегося по основному пути)), мощность определенного символа становится равной 0 (стирание) в соответствии с задержкой эхо-сигнала (путь распространения отличается от основного пути распространения).

Кроме того, даже при фазовом дрожании (канал связи, в котором задержка равна 0, и добавляется эхо-сигнал из-за доплеровской частоты), если отношение D/U равно 0 дБ, мощность всех OFDM символов в специфический момент времени становится равной 0 (стирание) из-за Доплеровской частоты.

Кроме того, пакетная ошибка может возникать из-за статуса проводного соединения из приемного блока (не показан), такого как антенна, которая принимает сигналы из передающего устройства 11 на стороне приемного устройства 12 в приемное устройство 12, и нестабильности мощности приемного устройства 12.

Между тем, при декодировании LDPC-кода в столбце матрицы H контроля четности и переменного узла, соответствующего кодовому биту LDPC-кода, как показано на фиг. 5, поскольку выполняется вычисление переменного узла из уравнения (1) с добавлением кодового бита (его принятого значения u_{0j}) LDPC-кода, если в кодовом бите, используемом при вычислении переменного узла, возникает ошибка, точность полученного сообщения уменьшается.

Затем при декодировании LDPC-кода, поскольку вычисление проверочного узла из уравнения (7) в проверочном узле выполняется с использованием сообщения, которое получено в переменном узле, который подсоединен к проверочному узлу, если увеличивается число проверочных узлов, в которых множество переменных узлов,

соединенных с ним (кодировый бит LDPC-кода, соответствующий ему), одновременно становится ошибочным состоянием (включая стирание), производительность декодирования ухудшается.

Другими словами, например, если два или более переменных узлов, подсоединенных к проверочному узлу, стираются одновременно, проверочный узел возвращает сообщение, в котором вероятность наличия значения 0 и вероятность наличия значения 1 являются одинаковыми во всех переменных узлах. В этом случае, проверочный узел, которые возвращает сообщение с равной вероятностью, не вносит вклад ни в один процесс декодирования (один набор из вычисления переменного узла и вычисления проверочного узла), в результате требуется большое число итераций процесса декодирования, производительность декодирования ухудшается, и увеличивается потребляемая мощность приемного устройства 12, которая декодирует LDPC-код.

Таким образом, передающая система, показанная на фиг. 7, позволяет повысить стойкость к ошибке и стиранию пакета при сохранении производительности в канале связи AWGN (AWGN-канале).

Пример конфигурации передающего устройства 11

На фиг. 8 показана блок-схема, иллюстрирующая пример конфигурации передающего устройства 11 (фиг. 7).

В передающем устройстве 11 входные потоки из одного или более входных потоков в качестве целевых данных подаются в мультиплексор 111 с адаптацией режима.

Мультиплексор 111 с адаптацией режима выполняет процесс, такой как выбор режима и мультиплексирование одного или более входных потоков, подаваемых в него, при необходимости, и обеспечивают подачу результирующих данных в блок 112 набивки.

Блок 112 набивки выполняет необходимую набивку нулями (вставку нулей) в отношении данных, полученных из мультиплексора 111 с адаптацией режима, и обеспечивает подачу результирующих данных в ВВ-скремблер 113.

ВВ-скремблер 113 выполняет основополосное скремблирование (ВВ-скремблирование) в отношении данных, поступающих из блока 112 набивки, и подает результирующие данные в ВСН-кодер 114.

ВСН-кодер 114 выполняет ВСН-кодирование-кодирование над данными, поступающими из ВВ-скремблера 113, и подает результирующие данные в виде целевых LDPC-данных, которые представляют собой цель LDPC-кодирования, в LDPC-кодер 115.

LDPC-кодер 115 выполняет LDPC-кодирование в соответствии с матрицей контроля четности, в которой матрица четности, которая представляет собой часть, соответствующую битам четности LDPC-кода, имеет ступенчатую структуру над целевыми LDPC-данными, поступающими из ВСН-кодера 114, и выводит LDPC-код с целевыми LDPC-данными в виде информационного бита.

Другими словами, LDPC-кодер 115 выполняет LDPC-кодирование, которое кодирует целевые LDPC-данные в LDPC-код (соответствующий матрице контроля четности), который определен в предварительно определенной стандартах, таких как, например, DVB-S.2, DVB-T.2 и DVB-C.2, или в LDPC-код (соответствующий матрице контроля четности), который определяется заранее, и выводит результирующий LDPC-код.

В данном случае, LDPC-код, который определен в стандартах, таких как DVB-S.2, DVB-T.2 и DVB-C.2, представляет собой нерегулярный код повторения-накопления (IRA), и матрица четности в матрице контроля четности LDPC-кода имеет ступенчатую структуру. Матрица четности и ступенчатая структура будут описаны позже. Кроме того, код IRA описан, например, в "Irregular Repeat-accumulate Codes," Н. Jin, А. Khandekar,

and R.J. McEliece, in Proceedings of 2nd International Symposium on Turbo codes and Related Topics, pp. 1-8, Sept. 2000.

LDPC-код, выводимый из кодера 115 LDPC, подается в перемежитель 116 битов.

5 Перемежитель 116 битов выполняет перемежение битов, которое будет описано позже, над LDPC-кодом, поступающим из LDPC-кодера 115, и подает LDPC-код после битового перемежения в блок 117 отображения.

10 Блок 117 отображения выполняет квадратурную модуляцию (многоуровневую модуляцию) путем отображения LDPC-кода, поступающего из перемежителя 116 битов, в сигнальную точку, представляющую один символ квадратурной модуляции, в блок кодового бита одного бита или более из LDPC-кода (символьного блока).

Другими словами, блок 117 отображения выполняет квадратурную модуляцию путем отображения LDPC-кода, поступающего из битового перемежителя 116, в сигнальные точки, которые определены в схеме модуляции, которая выполняет квадратурную модуляцию LDPC-кода, на плоскости IQ (созвездии IQ), которая определена в виде оси 15 I, представляющей собой компоненту I, которая находится в фазе с несущей волной, и оси Q, представляющей собой компоненту Q, которая ортогональна несущей волне.

В данном случае, пример схема модуляции квадратурной модуляции, выполняемой в блоке 117 отображения, включает в себя, например, схему модуляции, которая определена в стандартах, таких как DVB-S.2, DVB-T.2 и DVB-C.2, и другие схемы 20 модуляции, иными словами, например, двоичную фазовую манипуляцию (BPSK), квадратурную фазовую манипуляцию (QPSK), 8-позиционную фазовую манипуляцию (PSK), 16-позиционную амплитудно-фазовую манипуляцию (APSK), 32 APSK, 16-позиционную квадратурную амплитудную модуляцию (QAM), 64QAM, 256QAM, 1024QAM, 4096QAM и 4-позиционную импульсную амплитудную модуляцию (PAM). 25 В блоке 117 отображения схема модуляции, с помощью которой выполняется квадратурная модуляция, устанавливается заранее, например, в соответствии с операцией оператора передающего устройства 11.

Данные, которые получаются с помощью процесса, выполняемого в блоке 117 отображения (результат отображения, полученные путем отображения символа в 30 сигнальную точку), подаются во временной перемежитель 118.

Временной перемежитель 118 выполняет временное перемежение (перемежение в направлении времени) в блоке символа над данными, поступающими из блока 117 отображения, и подает результирующие данные в кодер 119 с одним входом и одним выходом/с множеством входов и одним выходом (SISO/MISO).

35 Кодер 119 SISO/MISO выполняет пространственно-временное кодирование над данными, поступающими из временного перемежителя 118, и подает результирующие данные в частотный перемежитель 120.

Частотный перемежитель 120 выполняет частотное перемежение (перемежение в частотном направлении) в блоке символа над данными, поступающими из кодера 119 40 SISO/MISO, и подает результирующие данные в блок 131 строителя кадров/выделения ресурсов.

Между тем, например, управляющие данные для управления передачей, такие как основополосная сигнализация (ВВ-сигнализация) (заголовок ВВ) подается в ВСН-кодер 121.

45 ВСН-кодер 121 аналогично ВСН-кодеру 114 выполняет ВСН-кодирование управляющих данных, подаваемых в него, и подает результирующие данные в LDPC-кодер 122.

LDPC-кодер 122, аналогично LDPC-кодеру 115, выполняет LDPC-кодирование над

данными, поступающими из BCH-кодера 121 в качестве целевых LDPC-данных, и подает результирующий LDPC-код в блок 123 отображения.

Блок 123 отображения, аналогично блоку 117 отображения, выполняет квадратурную модуляцию путем отображения LDPC-кода, поступающего из LDPC-кодера 122, в 5 сигнальную точку, представляющую один символ квадратурной модуляции, в блоке кодового бита из одного бита или более LDPC-кода (символьного блока), и подает результирующие данные в частотный перемежитель 124.

Частотный перемежитель 124, аналогично частотному перемежителю 120, выполняет частотное перемежение в символьном блоке над данными, поступающими из блока 10 123 отображения, и подает результирующие данные в блок 131 выделения ресурсов/построитель кадров.

Блок 131 выделения ресурсов/построитель кадров вставляет пилотный символ в необходимую позицию данных (символа), поступающих из частотных перемежителей 120 и 124, вырабатывает кадр, сформированный из символов с предварительно 15 определенным числом (например, кадр физического уровня (PL), кадр T2, кадр C2 и т.п.), из результирующих данных (символа) и подает кадр в блок 132 выработки OFDM-сигнала.

Блок 132 выработки OFDM-сигнала вырабатывает OFDM-сигнал, соответствующий кадру, из кадра, поступающего из блока 131 выделения ресурсов/построения кадров, 20 и передает OFDM-сигнал через канал 13 связи (фиг. 7).

В дополнение, передающее устройство 11 можно сконфигурировать без наличия некоторых блоков, показанных на фиг. 8, таких как, например, временной перемежитель 118, кодер 119 SISO/MISO, частотный перемежитель 120 и частотный перемежитель 124.

25 Фиг. 9 иллюстрирует пример конфигурации перемежителя 116 битов, показанного на фиг. 8.

Перемежитель 116 битов имеет функцию перемежения данных и выполнен с перемежителем 23 четности, перемежителем 24 со скручиванием столбцов и 30 демультиплексором (DEMUX) 25. В дополнение, перемежитель 116 битов можно выполнить без наличия одного или двух из перемежителя 23 четности и перемежителя 24 со скручиванием столбцов.

Перемежитель 23 четности выполняет перемежение битов четности, которое обеспечивает перемежение бита четности LDPC-кода, поступающего из LDPC-кодера 115, в позиции другого бита четности и подает LDPC-код после перемежения четности 35 в перемежитель 24 со скручиванием столбцов.

Перемежитель 24 со скручиванием столбцов выполняет перемежение со скручиванием столбцов над LDPC-кодом, поступающим из перемежителя 23 четности, и подает LDPC-код после перемежения со скручиванием столбцов в демультиплексор 25.

Другими словами, LDPC-код передается тогда, когда кодовые биты одного бита или 40 более LDPC-кода отображаются в сигнальную точку, представляющую один символ или квадратурную модуляцию с помощью блока 117 отображения (фиг. 8).

В перемежителе 24 со скручиванием столбцов, например, перемежение со скручиванием столбцов, которое будет описано позже, выполняется в качестве процесса перегруппировки, предназначенного для перегруппировки кодовых битов LDPC-кода, 45 поступающего из перемежителя 23 четности, таким образом, что множество кодовых битов LDPC-кода, соответствующего любой 1 в любой одной строке матрицы контроля четности, которая используется в LDPC-кодере 115, не включены в один символ.

Демультиплексор 25 получают LDPC-код, имеющей повышенную устойчивость к

AWGN и т.п., путем выполнения процесса замены для замены позиции двух или более кодовых битов LDPC-кода, который представляет собой символ, над LDPC-кодом, поступающим из перемежителя 24 со скручиванием столбцов. Затем демультимплексор 25 подает два или более кодовых битов LDPC-кода, который получается в процессе замены, в качестве символа, в блок 117 отображения (фиг. 8).

На фиг. 10 показана схема, иллюстрирующая матрицу H контроля четности, используемую при LDPC-кодировании в LDPC-кодере 115, показанном на фиг. 8.

Матрица H контроля четности имеет структуру порождающей матрицы малой плотности (LDGM) и может быть представлена уравнением $H=[H_A|H_T]$ (матрица, в которой элементы информационной матрицы H_A представляют собой элементы, расположенные слева, элементы матрицы H_T четности представляют собой элементы, расположенные справа) с помощью информационной матрицы H_A частей, соответствующих информационному биту, и матрицы H_T четности частей, соответствующих биту четности, из кодовых битов LDPC-кода.

В данном случае, число битов информационного бита и число битов бита четности из кодовых битов одного LDPC-кода (1 кодовое слово), соответственно, упоминаются как информационная длина K и длина M битов четности, и число кодовых битов одного LDPC-кода упоминается как длина $N (=K+M)$ кода.

Информационная длина K и длина M битов четности для LDPC-кода с определенной длиной N кода определяются с помощью кодовой скорости. Кроме того, матрица H контроля четности представляет собой матрицу типа строка $\times$ столбец размером $M \times N$. Затем информационная матрица H_A представляет собой матрицу размером $M \times K$, и матрица H_T четности представляет собой матрицу размером $M \times M$.

На фиг. 11 показана матрица H_T четности матрицы H контроля четности LDPC-кода, которая определена в стандартах DVB-S.2, DVB-T.2 и DVB-C.2.

Как показано на фиг. 11, матрица H_T четности матрицы H контроля четности LDPC-кода, который определен в стандарте, такой как DVB-T.2, представляет собой матрицу со ступенчатой структурой, в которой элементы 1 размещаются в ступенчатом виде (нижняя двухдиагональная матрица). Вес строки матрицы H_T четности равен 1 в первой строке и равен 2 во всех оставшихся строках. Кроме того, вес столбца равен 1 в последнем столбце и равен 2 во всех оставшихся столбцах.

Кроме того, как описано выше, можно легко выработать LDPC-код матрицы H контроля четности, чья матрица H_T четности имеет ступенчатую структуру, за счет использования матрицы H контроля четности.

Другими словами, LDPC-код (одно кодовое слово) представлен с помощью вектора-строки s , и вектор-столбец, полученный путем транспонирования вектора-строки, представлен s^T . Кроме того, часть информационного бита представлена с помощью вектора-строки s , и часть бита четности представлена с помощью вектора-строки T , в котором вектор-строка представляет собой LDPC-код.

В этом случае вектор-строка s представлен с помощью уравнения $s=[A|T]$ (вектор-строка, в котором элементы вектора-строки A представляют собой элементы, расположенные слева, и элементы вектора-строки T представляют собой элементы, расположенные справа) с использованием вектора-строки A в качестве информационного бита и вектора-строки T в качестве бита четности.

Необходимо, чтобы матрицы H контроля четности и вектор-строка $s=[A|T]$ в качестве

LDPC-кода удовлетворяли уравнению $Hc^T=0$, и если матрица H_T четности матрицы $H=[H_A|H_T]$ контроля четности имеет ступенчатую структуру, показанную на фиг. 11, вектор-строка T в качестве битов четности, конфигурирующих вектор-строку $c=[A|T]$,
 5 удовлетворяющую уравнению $Hc^T=0$, можно получить последовательно (по порядку) путем выработки элемента каждой строки 0 из элемента первой строки вектора-столбца Hc^T по порядку в уравнении $Hc^T=0$.

На фиг. 12 показана схема, изображающая матрицу H контроля четности LDPC-кода, которая определяется в стандарте DVB-T.2.

10 В матрице H контроля четности LDPC-кода, которая определяется в стандарте DVB-T.2, вес X столбца задается в первом столбце - KX -ом столбце, и вес столбца 3 задается в следующем $K3$ столбце, вес столбца 2 задается в следующем $M-1$ столбце, и вес столбца 1 задается в последнем столбце.

В данном случае, выражение $KX+K3+M-1+1$ равно длине N кода.

15 На фиг. 13 показана схема, иллюстрирующая число столбцов KX , $K3$ и M и вес X столбца для каждой скорости кодирования LDPC-кода, которая определена в стандартах DVB-T.2 и т.п.

В стандартах DVB-T.2 и т.п. определены LDPC-коды длиной N кода 64800 битов и 16200 битов.

20 Затем определяются 11 кодовых скоростей (номинальных скоростей) $1/4$, $1/3$, $2/5$, $1/2$, $3/5$, $2/3$, $3/4$, $4/5$, $5/6$, $8/9$ и $9/10$ для LDPC-кода длиной N кода 64800 битов, и определяются 10 кодовых скоростей $1/4$, $1/3$, $2/5$, $1/2$, $3/5$, $2/3$, $3/4$, $4/5$, $5/6$ и $8/9$ для LDPC-кода длиной N кода 16200 битов.

25 В данном случае, длина N кода 64800 битов упоминается как 64 кбит, и длина N кода 16200 битов упоминается как 16 кбит.

Для LDPC-кода кодовый бит, соответствующий столбцу, имеющему большой вес столбца матрицы H контроля четности, должен иметь, вероятно, низкую частоту появления ошибок.

30 В матрице H контроля четности, которая определена в стандарте DVB-T.2 и т.п., показанной на фиг. 12 и фиг. 13, чем ближе столбец находится к первой стороне (к левой стороне), тем, вероятно, будет больше вес столбца, поэтому для LDPC-кода, соответствующему матрице H контроля четности, кодовый бит на начальной стороне должен быть, вероятно, сильным по отношению к ошибке (то есть иметь устойчивость к ошибке), и кодовый бит на конечной стороне должен быть, вероятно, слабым по
 35 отношению к ошибке.

Фиг. 14 иллюстрирует пример размещения 16 символов (сигнальных точек, соответствующих им) на плоскости IQ, когда 16QAM выполняется блоком 117 отображения, показанным на фиг. 8.

40 Другими словами, поз. А на фиг. 14 показывает символы (сигнальные точки, соответствующие им) 16QAM стандарта DVB-T.2.

В 16QAM один символ представлен четырьмя битами, и существует $16 (=2^4)$ символов. Затем 16 символов размещаются таким образом, чтобы сформировать квадрат размером 4×4 square в направлении $I \times$ направлении Q , с началом координат плоскости IQ в качестве центра.

45 Теперь, если $(i+1)$ -ый бит из старшего значащего бита в битовой последовательности, представленной с помощью одного символа, представлен в виде бита y_i , четыре бита, которые представляют собой один бит 16QAM, можно представить в виде битов y_0, y_1 ,

y_2 и y_3 из старшего значащего бита по порядку. Если схемой модуляции является 16QAM, четыре бита кодовых битов LDPC-кода становятся (принимают вид символа) в символ (значения символа) из четырех битов y_0 - y_3 .

Поз. В на фиг. 14 показывает битовые границы для четырех битов y_0 - y_3 символьного бита y_i (который в дальнейшем упоминается как символьный бит), представленных символами 16QAM.

В данном случае, битовая граница для символьного бита y_i (на фиг. 14 $i=0, 1, 2$, и 3) означает границу между символом, чей символьный бит y_i равен 0, и символом, чей символьный бит y_i равен 1.

Как показано поз. В на фиг. 14, для верхнего символьного бита y_0 из четырех символьных битов y_0 - y_3 , представленных символами 16QAM, одно место по Q-оси в плоскости IQ представляет собой битовую границу, и для второго места (второго из старшего значащего бита) символьного бита y_1 только одно место по I-оси в плоскости IQ представляет собой битовую границу.

Кроме того, для третьего символьного бита y_2 , битовые границы расположены между первым столбцом и вторым столбцом и между третьим столбцом и четвертым столбцом слева в пределах 4×4 символов.

Кроме того, для четвертого символьного бита y_3 , битовые границы находятся между первой строкой и второй строкой и между третьей строкой и четвертой строкой сверху в пределах 4×4 символов.

Что касается символьного бита y_i , представленного символами, чем больше символов находится в стороне от битовой границы, тем меньше возникает ошибок (вероятность возникновения ошибок является низкой), и чем больше символов находится ближе к битовой границе символов, тем больше возникает ошибок (вероятность возникновения ошибок является высокой).

Теперь, если бит, в котором почти не возникает ошибка (сильный с точки зрения возникновения ошибок) упоминается как "сильный бит", и бит, в котором легко возникает ошибка (слабый с точки зрения возникновения ошибок), упоминается как "слабый бит", то для четырех символьных битов y_0 - y_3 для символа 16QAM, верхний символьный бит y_0 и второй символьный бит y_1 являются сильными битами, а третий символьный бит y_2 и четвертый символьный бит y_3 являются слабыми битами.

Фиг. 15 - фиг. 17 иллюстрируют пример размещения 64 символов (сигнальных точек, соответствующих им), другими словами, символов 16QAM DVB-T.2 на плоскости IQ, когда 64QAM выполняется с помощью блока 117 отображения, показанного на фиг. 8.

В 64QAM один символ представлен шестью битами, и существует $64 (=2^6)$ символа. Затем 64 символа размещаются таким образом, чтобы сформировать квадрат размером 8×8 в направлении I $\times$ направлении Q, с началом координат плоскости IQ в качестве центра.

Символьные биты одного символа 64QAM можно представить в виде битов y_0, y_1, y_2, y_3, y_4 и y_5 из старшего значащего бита и здесь по порядку. Если схемой модуляции является 64QAM, то шесть битов кодовых битов LDPC-кода представляет собой символ из символьных битов y_0 - y_5 шести битов.

В данном случае, фиг. 15 иллюстрирует битовую границу для каждого из верхнего

символьного бита y_0 и второго символьного бита y_1 , фиг. 16 иллюстрирует битовую границу для каждого из третьего символьного бита y_2 и четвертого символьного бита y_3 , и фиг. 17 иллюстрирует битовую границу для каждого из пятого символьного бита y_4 и шестого символьного бита y_5 среди символьных битов y_0 - y_5 символа 64QAM.

Как показано на фиг. 15, число битовых границ для каждого из верхнего символьного бита y_0 и второго символьного бита y_1 равно одному. Кроме того, как показано на фиг. 16, число битовых границ для каждого из третьего символьного бита y_2 и четвертого символьного бита y_3 равно двум, и как показано на фиг. 17, число битовых границ для каждого из пятого символьного бита y_4 и шестого символьного бита y_5 равно четырем.

Поэтому, для символьных битов y_0 - y_5 символа 64QAM верхний символьный бит y_0 и второй символьный бит y_1 являются самыми сильными битами, и третий символьный бит y_2 и четвертый символьный бит y_3 являются вторыми самыми сильными битами. Затем пятый символьный бит y_4 и шестой символьный бит y_5 являются слабыми битами.

Из фиг. 14 и фиг. 15 - фиг. 17 видно, что верхний бит должен быть, вероятно, сильным битом, и нижний бит должен быть, вероятно, слабым битом для символьных битов символа квадратурной модуляции.

Фиг. 18 иллюстрирует пример размещения четырех символов (сигнальных точек, соответствующих им) на плоскости IQ, другими словами, например, размещение сигнальных точек QPSK DVB-S.2, когда спутниковая линия используется в качестве канала 13 связи (фиг. 7), и QPSK выполняется блоком 117 отображения, показанным на фиг. 8.

В QPSK стандарта DVB-S.2 символ отображается в одну из четырех сигнальных точек, расположенных по окружности круга, имеющего радиус r , равный 1 с началом координат плоскости IQ в качестве центра.

Фиг. 19 иллюстрирует пример размещения восьми символов на плоскости IQ, другими словами, например, размещения сигнальных точек 8PSK DVB-S.2, когда спутниковая линия используется в качестве канала 13 связи (фиг. 7) и 8PSK выполняется блоком 117 отображения, показанным на фиг. 8.

В 8PSK стандарта DVB-S.2, символ отображается в один из восьми сигнальных точек, расположенных по окружности круга, имеющего радиус r , равный 1, с началом координат плоскости IQ в качестве центра.

Фиг. 20 иллюстрирует пример размещения 16 символов на плоскости IQ, другими словами, например, размещения сигнальных точек 16APSK стандарта DVB-S.2, когда спутниковая линия используется в качестве канала 13 связи (фиг. 7) и 16APSK выполняется блоком 117 отображения, показанным на фиг. 8.

Поз. А на фиг. 20 показывает созвездие 16APSK стандарта DVB-S.2.

В 16APSK стандарта DVB-S.2, символ отображается в одну из всех 16 сигнальных точек, включающих в себя четыре сигнальных точек, расположенных по окружности круга, имеющего радиус R_1 , и 12 сигнальных точек, расположенных по окружности круга, имеющего радиус R_2 ($>R_1$), с началом координат плоскости IQ в качестве центра.

Поз. В на фиг. 20 показывает отношение $\gamma = R_2/R_1$ радиуса R_2 к радиусу R_1 в созвездии 16APSK стандарта DVB-S.2.

Отношение γ радиуса R_2 к радиусу R_1 изменяется в зависимости от кодовой скорости в созвездии 16APSK стандарта DVB-S.2.

Фиг. 21 иллюстрирует пример размещения 32 символов на плоскости IQ, другими

словами, например, размещение сигнальных точек 32APSK стандарта DVB-S.2, когда спутниковая линия используется в качестве канала 13 связи (фиг. 7), и 32APSK выполняется блоком 117 отображения, показанным на фиг. 8.

Поз. А на фиг. 21 показывает созвездие 32APSK стандарта DVB-S.2.

5 В 32APSK стандарта DVB-S.2, символ отображается в одну из всех 32 сигнальных точек, включающих в себя четыре сигнальных точки, расположенных по окружности круга, имеющего радиус R_1 , 12 сигнальных точек, расположенных по окружности круга, имеющего радиус $R_2 (>R_1)$, и 16 сигнальных точек расположенных по окружности круга, имеющего радиус $R_3 (>R_2)$, с началом координат плоскости IQ в качестве центра.

10 Поз. В на фиг. 21 показывает отношение $\gamma_1=R_2/R_1$ радиуса R_2 к радиусу R_1 и отношение $\gamma_2=R_3/R_1$ радиуса R_3 к радиусу R_1 в созвездии 32APSK стандарта DVB-S.2.

Отношение γ_1 радиуса R_2 к радиусу R_1 и отношение γ_2 радиуса R_3 к радиусу R_1 изменяются в зависимости от кодовой скорости в созвездии 32APSK стандарта DVB-S.2.

15 Существуют сильные биты и слабые биты даже для символьных битов для символов соответствующих типов квадратурных модуляций (QPSK, 8PSK, 16APSK и 32APSK) стандарта DVB-S.2, чьи созвездия показаны на фиг. 18 - фиг. 21, аналогично случаю, показанному на фиг. 14 - фиг. 17.

20 В данном случае, как показано на фиг. 12 и фиг. 13, существуют кодовые биты, которые являются слабыми с точки зрения возникновения ошибок, и кодовые биты, которые являются сильными с точки зрения возникновения ошибок, для LDPC-кода, который выводит LDPC-кодер 115 (фиг. 8).

25 Кроме того, как показано на фиг. 14 - фиг. 21, существуют сильные биты и слабые биты для символьных битов символа квадратурной модуляции, выполняемой блоком 117 отображения.

Поэтому, если кодовый бит, который является слабым с точки зрения возникновения ошибок, LDPC-кода выделяется слабому символьному биту символа квадратурной модуляции, то, в целом, устойчивость к ошибке уменьшается.

30 Таким образом, в качестве тенденции выделения кодового бита, который является слабым к ошибке, LDPC-кода сильному биту (символьному биту) символа квадратурной модуляции предложен перемежитель для перемежения кодового бита LDPC-кода.

Демультимплексор 25, показанный на фиг. 9, может выполнять процесс, выполняемый перемежителем.

35 На фиг. 22 показана схема, изображающая процесс, выполняемый демультимплексором 25 (фиг. 9).

Другими словами, поз. А на фиг. 22 показывает пример функциональной конфигурации демультимплексора 25.

Демультимплексор 25 выполнен с памятью 31 и блоком 32 замены.

40 LDPC-код из LDPC-кодера 115 подается в память 31.

Память 31 имеет объем памяти для хранения mb битов в направлении строки (по горизонтали) и $N/(mb)$ битов в направлении столбца (по вертикали) и обеспечивает блок 32 замены кодовым битом LDPC-кода, который подается в нее, путем записи кодового бита в направлении столбца и считывания кодового бита в направлении строки.

В данном случае, N (= информационная длина K + длина M битов четности) представляет собой длину кода LDPC-кода, как описано выше.

Кроме того, m представляет собой число битов кодового бита LDPC-кода в качестве

одного символа, b представляет собой предварительно определенное положительное целое число, и m представляет собой множитель, который используется для целочисленного умножения. Демультимплексор 25 формирует (символизирует) кодовый бит LDPC-кода в символ в блоке с предварительно определенным числом битов m , и множитель b представляет собой число символов, полученных с помощью однократной символизации с помощью демультимплексора 25.

Поз. А на фиг. 22 иллюстрирует пример конфигурации демультимплексора 25, в котором схемой модуляции является 64QAM, которая отображает символ в любой из 64 сигнальных точек, и поэтому, числом m битов кодового бита LDPC-кода, которое равно одному символу, равно шести битам.

Кроме того, как показано поз. А на фиг. 22, множитель b равен 1, и поэтому память 31 имеет объем памяти $N/(6 \times 1) \times (6 \times 1)$ битов в направлении столбца x направлении строки.

В данном, случае, область хранения памяти 31, которая имеет один бит в направлении строки и продолжается в направлении столбца, в дальнейшем, при необходимости, упоминается как столбец. Как показано поз. А на фиг. 22, память 31 выполнена с 6 ($= 6 \times 1$) столбцами.

В демультимплексоре 25 запись кодового бита LDPC-кода в направлении сверху вниз столбцов (в направлении столбца), конфигурирующих память 31, выполняется в направлении столбца слева направо.

Затем, если запись кодового бита закончилась внизу крайнего правого столбца, кодовые биты считываются из первого столбца всех столбцов, конфигурирующих память 31, в направлении строки в блок из шести битов (mb битов) и подаются в блок 32 замены.

Блок 32 замены выполняет процесс замены, предназначенный для замены позиций кодовых битов из шести битов из памяти 31, и выводит шесть результирующих битов в качестве шести символьных битов u_0, u_1, u_2, u_3, u_4 и u_5 , которые представляют собой один символ 64QAM.

Другими словами, кодовые биты mb битов (в данном случае, шесть битов) считываются из памяти 31 в направлении строки, но если i -ый бит ($i=0, 1, \dots, mb-1$) из старшего значащего бита кодовых битов mb битов, которые считываются из памяти 31, представлен в качестве бита b_i , кодовые биты из шести битов, которые считываются из памяти 31 в направлении строки, можно представить в порядке в виде битов b_0, b_1, b_2, b_3, b_4 и b_5 от старшего значащего бита.

Из зависимости весов столбцов, показанной на фиг. 12 и фиг. 13, кодовый бит, расположенный в направлении бита b_0 , представляет собой кодовый бит, который является "сильным" по отношению к ошибке, и кодовый бит, расположенный в направлении бита b_5 представляет собой кодовый бит, который является "слабым" по отношению к ошибке.

Блок 32 замены может выполнять процесс замены, предназначенный для замены позиций кодовых битов b_0 - b_5 количеством шесть битов из памяти 31 таким образом, что кодовый бит, который является "слабым" по отношению к ошибке среди кодовых битов b_0 - b_5 количеством шесть битов из памяти 31 выделяется сильным битам среди символьных битов u_0 - u_5 одного символа 64QAM.

В данном случае, различные способы замены, которые позволяют заменять кодовые биты b_0 - b_5 количеством шесть битов из памяти 31 и выделять их каждому из шести

символьных битов y_0 - y_5 , представляющих один символ 64QAM, были предложены многими компаниями.

Поз. В на фиг. 22, поз. С на фиг. 22 и поз. D на фиг. 22 соответственно показывают первый способ замены, второй способ замены и третий способ замены.

В поз. В на фиг. 22 - поз. D на фиг. 22 (аналогичным образом даже на фиг. 23, который будет описан позже), линия, соединяющая бит b_i и y_i означает выделение кодового бита b_i символному биту y_i символа (заменяя его в позиции символного бита y_i).

В качестве первого способа замены (поз. В на фиг. 22) предлагается выбрать любой из трех типов способов замены, и в качестве второго способа замены (поз. С на фиг. 22) предлагается выбрать любой из двух типов способов замены.

В качестве третьего способа замены (поз. D на фиг. 22) предлагается выбрать и использовать надлежащим образом шесть типов способов замены.

Фиг. 23 иллюстрирует пример конфигурации демультимплексора 25, в котором схемой модуляции является 64QAM, которая отображает символ в любую из 64 сигнальных точек (поэтому число m битов кодового бита LDPC-кода, который отображается в один символ, равно шести битам также, как и на фиг. 22), и множитель b равен 2, и четвертый способ замены.

Когда множитель b равен 2, память 31 включает в себя объем памяти $N/(6 \times 2) \times (6 \times 2)$ битов в направлении столбца $\times$ направлении строки, и выполнена с 12 ($=6 \times 2$) столбцами.

Поз. А на фиг. 23 иллюстрирует порядок записи LDPC-кода в память 31.

В демультимплексоре 25, как показано на фиг. 22, запись кодового бита LDPC-кода в направлении сверху вниз столбцов (в направлении столбца), конфигурирующих память 31, выполняется в направлении столбца слева направо.

Затем, если запись кодового бита закончилась внизу крайнего правого столбца (при окончании записи одного кодового слова), кодовые биты считываются из первого столбца всех столбцов, конфигурирующих память 31, в направлении строки в блоке из 12 битов (m битов) и подаются в блок 32 замены.

Блок 32 замены выполняет процесс замены, предназначенный для замены позиций кодовых битов из 12 битов из памяти 31 с использованием четвертого способа замены, и выводит 12 результирующих битов в виде 12 битов представляющих собой два символа (b символов) 64QAM, другими словами, шесть символьных битов y_0, y_1, y_2, y_3, y_4 и y_5 , которые представляют один символ 64QAM, и шесть символьных битов y_0, y_1, y_2, y_3, y_4 и y_5 , которые представляют собой один следующий символ.

В данном случае, поз. В на фиг. 23 иллюстрирует четвертый способ замены процесса замены, выполняемого блоком 32 замены (поз. А на фиг. 23).

В дополнение к этому, когда множитель b равен 2 (то же самое относится к случаю, когда он равен 3 или более), в процессе замены кодовые биты m битов выделяются символным битам m битов последующих символов b . В дальнейшем для удобства объяснения, в том числе как показано на фиг. 23, $(i+1)$ -ый бит из старшего значащего бита символного бита m битов последующих символов b будет представлен в виде бита (символьного бита) y_i .

То, что способ замены кодовых битов является подходящим, другими словами, то, что дополнительно повышает частоту появления ошибок в канале связи AWGN и т.п., варьируется в зависимости от кодовой скорости и длины кода LDPC-кода, схемы модуляции и т.п.

Перемежевание битов четности

Далее, со ссылкой на фиг. 24 - фиг. 26, будет описано перемежевание битов четности,

выполняемое перемежителем 23 битов четности на фиг. 9.

Фиг. 24 иллюстрирует график Таннера (часть его) матрицы контроля четности LDPC-кода.

Как показано на фиг. 24, если множество (2 или т.п.) переменных узлов (кодовых битов, соответствующих им), соединенных с проверочным узлом, одновременно становятся ошибочными узлами, такими как блок стирания, проверочный узел возвращает сообщение, в котором вероятность наличия значения 0 и вероятность наличия значения 1 равны, во все переменные узлы, соединенные с проверочным узлом. Поэтому, если множество переменных узлов, соединенных с одним и тем же проверочным узлом, одновременно становятся устройством стирания, производительность декодирования ухудшается.

В данном случае, LDPC-код, который определен в стандарте DVB-S.2 и выводится с помощью LDPC-кодера 115 (фиг. 8), представляет собой IRA код, и матрица H_T четности матрицы H контроля четности имеет ступенчатую структуру, как показано на фиг. 11.

На фиг. 25 показана схема, иллюстрирующая матрицу H_T четности, имеющая ступенчатую структуру, и график Таннера, соответствующий матрице H_T четности.

Другими словами, поз. А на фиг. 25 показывает матрицу H_T четности, имеющую ступенчатую структуру, и поз. В на фиг. 25 показывает график Таннера, соответствующий матрице H_T четности (поз. А на фиг. 25).

В матрице H_T четности, имеющей ступенчатую структуру, элементы 1 находятся рядом в каждой строке (за исключением первой строки). Поэтому, на графике Таннера матрицы H_T четности два соседних переменных узла, соответствующих столбцам двух соседних элементов, значения которых в матрице H_T четности равны 1, соединены с тем же самым проверочным узлом.

Поэтому, если биты четности, соответствующие двум соседним переменным узлам, описанным выше, одновременно становятся ошибочными битами из-за пакетной ошибки или стирания, проверочный узел, соединенный с двумя переменными узлами (с переменными узлами для получения сообщения с использованием бита четности), соответствующими двум битам четности, которые становятся ошибочными битами, возвращает сообщение, в котором вероятность наличия значения 0 и вероятность наличия значения 1 равны, в переменный узел, соединенный с проверочным узлом таким образом, что производительность декодирования ухудшается. Затем, если длина пакета (число битов из битов четности, которые становятся ошибочными битами один за другим) увеличивается, то число проверочных узлов, возвращающих сообщение, показывающее равные вероятности, увеличивается, и производительность декодирования дополнительно ухудшается.

Таким образом, для того чтобы предотвратить снижение производительности декодирования, описанной выше, перемежитель 23 битов четности (фиг. 9) выполняет перемежение битов четности LDPC-кода, подаваемого из LDPC-кодера 115 в позицию другого бита четности.

На фиг. 26 показана схема, иллюстрирующая матрицу H_T четности матрицы H контроля четности, соответствующую LDPC-коду, после перемежения битов четности, выполняемого перемежителем 23 битов четности (фиг. 9).

В данном случае, информационная матрица H матрицы H контроля четности, соответствующая LDPC-коду, которая выводится кодером 115 LDPC и определена в

стандарте DVB-S.2, имеет циклическую структуру.

Циклическая структура относится к структуре, в которой определенный столбец получается путем циклического сдвига другого столбца, и включает в себя, например, структуру, в которой для каждого P столбцов позиция 1 в каждой строке P столбца представляет собой позицию, полученную в результате циклического сдвига первого столбца P столбца на значение, пропорциональное значению q , полученному путем деления длины M битов четности в направлении столбца. В дальнейшем, в случае необходимости, P столбец в циклической структуре упоминается как число столбцов блока циклической структуры.

Пример LDPC-кода, который определен в стандарте DVB-S.2, включает в себя два типа LDPC-кодов, в которых длина N кода равна 64800 битам и 16200 битам, как описано на фиг. 12 и фиг. 13, для обоих двух типов LDPC-кодов число P столбцов блока циклической структуры определено как 360, которое является одним из делителей за исключением 1 и M среди делителей, имеющих длину M битов четности.

Кроме того, длина M четности равна значению, отличному от простого числа, представленного с помощью уравнения $M=q \times P=q \times 360$, при использовании значения q , которое изменяется в зависимости от кодовой скорости. Поэтому, значение q , аналогичное числу P столбцов блока циклической структуры, представляет собой также другой один из делителей, за исключением 1 и M среди делителей с длиной M четности, и получается путем деления длины M четности на число P столбцов блока циклической структуры (произведение P и q , которые являются делителями длины M четности, представляют собой длину M битов четности).

Как описано выше, когда информационная длина равна K , и x представляет собой целое число, равное 0, или больше или меньше, чем P , если y представляет собой целое число, равное 0, или больше или меньше, чем q , перемежитель 23 битов четности выполняет перемежение $(K+qx+y+1)$ -го кодового бита среди кодовых битов LDPC-кода N битов в позиции $(K+Py+x+1)$ -го кодового бита, в качестве перемежения битов четности.

Поскольку как $(K+qx+y+1)$ -ый кодовый бит, так и $(K+Py+x+1)$ -ый кодовый бит представляют собой $(K+1)$ -ый и последующие кодовые биты в соответствии с перемежением битов четности, позиция бита четности LDPC-кода смещается.

Согласно такому перемежению битов четности переменные узлы (бит четности, соответствующий им), соединенные с одним и тем же проверочным узлом, разделяются на число P столбцов блока циклической структуры, другими словами, в данном случае, 360 бит, поэтому, когда длина пакета составляет менее чем 360 битов, можно избежать состояния, в котором множество переменных узлов, соединенных с одним и тем же проверочным узлом, одновременно становятся ошибочными узлами, и тем самым повышая устойчивость к пакетной ошибке.

В дополнение к этому, LDPC-код после перемежения битов четности, предназначенного для перемежения $(K+qx+y+1)$ -го кодового бита позиции $(K+Py+x+1)$ -го кодового бита, совпадает с LDPC-кодом матрицы контроля четности (которая в дальнейшем упоминается как матрица контроля четности преобразования), полученной с помощью перестановки столбцов, предназначенной для замены $(K+qx+y+1)$ -ого столбца исходной матрицы N контроля четности на $(K+Py+x+1)$ -ый столбец.

Кроме того, квазициклическая структура с P столбцами (на фиг. 26 360 столбцов) показана в виде блока в матрице четности матрицы контроля четности преобразования, как показано на фиг. 26.

В данном случае квазициклическая структура означает, что структура, в которой некоторые части, за исключением других частей, имеет циклическую структуру. В

матрице контроля четности преобразования, полученной путем выполнения перестановки столбцов, соответствующей перемежению битов четности, в матрице контроля четности LDPC-кода, который определен в стандарте DVB-S.2, отсутствует элемент 1 (становится элементом 0) на участке 360 строк $\times$ 360 столбцов на участке
 5 правого угла (сдвиговая матрица, которая будет описана позже), и с этого момента матрица контроля четности не имеет (законченную) циклическую структуру, а имеет так называемую квазициклическую структуру.

В дополнение к этому, матрица контроля четности преобразования, показанная на фиг. 26, представляет собой матрицу, полученную путем выполнения перестановки
 10 строк таким образом, чтобы сконфигурировать конфигурирующую матрицу, которая будет описана позже, а также перестановки столбцов, соответствующей перемежению битов четности, исходной матрицы H контроля четности.

Перемежение со скручиванием столбцов

Далее, со ссылкой на фиг. 27 - фиг. 30, будет описано перемежение со скручиванием
 15 столбцов в качестве процесса перегруппировки, выполняемого перемежителем 24 со скручиванием столбцов (фиг. 9).

Передающее устройство 11 (фиг. 8) передает один бит или более кодовых битов LDPC-кода в качестве одного символа. Другими словами, например, когда два бита
 20 кодового бита представляют собой один символ, например, QPSK используется в качестве схемы модуляции, и когда четыре бита кодового бита представляют собой один символ, например, 16APSK или 16QAM используется в качестве схемы модуляции.

Когда два или более битов кодового бита передаются в виде одного символа, если в любом символе имеет место стирание или т.п., все кодовые биты символа стираются.

Поэтому, необходимо избежать соединений переменного узла, соответствующего
 25 кодовым битам одного символа, с тем же самым проверочным узлом, чтобы уменьшить вероятность того, что множество переменных узлов, соединенных с одним и тем же проверочным узлом (кодовый бит, соответствующий ему), стираются в одно и то же время для того, чтобы повысить производительность декодирования.

Между тем, как описано выше, в матрице H контроля четности LDPC-кода, которая
 30 выводится LDPC-кодером 115 и определена в стандарте DVB-S.2, информационная матрица H_A имеет циклическую структуру, и матрица H_T четности имеет ступенчатую структуру. Затем, как показано на фиг. 26, в матрице контроля четности преобразования, которая представляет собой матрицу контроля четности LDPC-кода после перемежения битов четности, циклическая структура (а именно, как описано выше, квазициклическая
 35 структура) показана в матрице четности.

Фиг. 27 иллюстрирует матрицу контроля четности преобразования.

Другими словами, фиг. 27 иллюстрирует матрицу контроля четности преобразования матрицы H контроля четности LDPC-кода, в которой длина N кода равна 64800 битам, и кодовая скорость (r) равна $3/4$.

В поз. А на фиг. 27, в матрице контроля четности преобразования, позиции элементов, значения которых установлены на 1, представлены точками ($\cdot$).

Поз. В на фиг. 27 иллюстрирует процесс, выполняемый демультиплексором 25 (фиг. 9) в отношении LDPC-кода матрицы контроля четности преобразования (поз. А фиг. 27), другими словами, LDPC-кода после перемежения битов четности в качестве цели.

В поз. В на фиг. 27 в случае схемы модуляции, такой как схема отображения символа в 16 сигнальных точек, например 16APSK или 16QAM, кодовые биты LDPC-кода после перемежения битов четности записываются в четыре столбца, конфигурирующие память
 45 31 демультиплексора 25 в направлении столбца.

Кодовые биты, которые записаны в четырех столбцах, конфигурирующих память 31 в направлении столбца, считываются в четырех битовых блоках в направлении строки с тем, чтобы представлять собой один символ.

В этом случае, кодовые биты V_0 , V_1 , V_2 и V_3 четырех битов в качестве одного символа могут представлять собой кодовые биты, соответствующие 1 в любой одной строке матрицы контроля четности преобразования (поз. А на фиг. 27), и в этом случае переменные узлы, соответственно, соответствующие кодовым битам V_0 , V_1 , V_2 и V_3 , соединены с одним и тем же проверочным узлом.

Поэтому, когда кодовые биты V_0 , V_1 , V_2 и V_3 четырех битов одного символа представляют собой кодовые биты, соответствующие 1 в любой одной строке матрицы контроля четности преобразования, если в символе происходит стирание, нельзя получить подходящее сообщение из одного и того же проверочного узла, соединенного с переменными узлами, соответственно, соответствующих кодовым битам V_0 , V_1 , V_2 и V_3 , и, таким образом производительность декодирования ухудшается.

Даже при кодовой скорости, отличной от кодовой скорости $3/4$, аналогичным образом, множество кодовых битов, соответствующих множеству переменных узлов, соединенных с одним и тем же узлом, может представлять собой один символ 16APSK или 16QAM.

Таким образом, перемежитель 24 со скручиванием столбцов выполняет перемежение со скручиванием столбцов, предназначенное для перемежения кодового бита LDPC-кода после перемежения битов четности из перемежителя 23 битов четности таким образом, что множество кодовых битов, соответствующих 1 в любой одной строке матрицы контроля четности преобразования, не включено в один символ.

На фиг. 28 показана схема, изображающая перемежение со скручиванием столбцов. Другими словами, фиг. 28 иллюстрирует память 31 (фиг. 22 и фиг. 23) демультиплексора 25.

Как показано на фиг. 22, память 31 имеет объем памяти для хранения mb битов в направлении столбца (по вертикали) и $N/(mb)$ битов в направлении строки (по горизонтали) и выполнена с mb столбцами. Затем перемежитель 24 со скручиванием столбцов выполняет перемежение со скручиванием столбцов путем управления позиции начала записи для памяти 31 при записи кодового бита LDPC-кода в направлении столбца и при считывании кодового бита в направлении строки.

Другими словами, перемежитель 24 со скручиванием столбцов подходящим образом изменяет позицию начала записи, в которой начинается запись кодового бита, для каждого из множества столбцов, таким образом, что множество кодовых битов, конфигурирующих один символ, которые считываются в направлении строки, не становятся кодовыми битами, соответствующими 1, расположенными в определенной одной строке матрицы контроля четности преобразования (кодовые биты LDPC-кода перегруппировываются таким образом, что множество кодовых битов, соответствующих 1, расположенных в определенной одной строке матрицы контроля четности, не включаются в один и тот же символ).

В данном случае, фиг. 28 иллюстрирует пример конфигурации памяти 31, когда схемой модуляции является 16APSK или 16QAM, и множитель b , описанный на фиг. 22, равен 1. Поэтому, числом m битов кодового бита LDPC-кода, сконфигурированного в виде одного символа, равно четырем битам, и память 31 сконфигурирована с четырьмя ($=mb$) столбцами.

Перемежитель 24 со скручиванием столбцов (то есть демультиплексор 25 на фиг. 22)

выполняет запись кодового бита LDPC-кода в направлении сверху вниз четырех столбцов (в направлении столбца), конфигурирующего память 31 в направлении столбца в направлении слева направо.

Затем, если запись кодового бита заканчивается в крайнем правом столбце, перемежитель 24 со скручиванием столбцов считывает кодовый бит из первой строки всех столбцов, конфигурирующих память 31, в направлении строки, в блоке из четырех битов (mb битов), и выводит кодовый бит в качестве LDPC-кода после перемежения со скручиванием столбцов в блок 32 замены (фиг. 22 и фиг. 23) демультимплексора 25.

Однако, если предполагается, что адрес первой (верхней) позиции каждого столбца равен 0, и адрес каждой позиции в направлении столбца представлен целыми числами в возрастающем порядке, в перемежителе 24 со скручиванием столбцов, предполагается, что позиция начала записи крайнего левого столбца представляет собой позицию адреса 0, позиция начала записи второго (слева) столбца представляет собой позицию адреса 2, позиция начала записи третьего столбца представляет собой позицию адреса 4, и позиция начала записи четвертого столбца представляет собой позицию адреса 7.

В дополнение к этому, что касается столбцов, чьи позиции начала записи отличаются от позиции адреса 0, то после того, как кодовые биты записались в крайней нижней позиции, кодовые биты перемещаются в верхнюю позицию (в позицию адреса 0) и записываются вплоть до позиции, расположенной непосредственно перед позицией начала записи. После этого выполняется запись в следующем (в правом) столбце.

Путем выполнения перемежения со скручиванием столбцов, описанным выше, можно избежать формирования множества кодовых битов, соответствующих множеству переменных узлов, соединенных с одним и тем же проверочным узлом, в виде одного символа 16APSK или 16QAM (включения в один символ), для LDPC-кода, который определен в стандарте DVB-T.2, в результате можно повысить производительность декодирования в канале связи со стиранием.

Фиг. 29 иллюстрирует требуемое число столбцов памяти 31 для перемежения со скручиванием столбцов и адрес позиции начала записи для каждой схемы модуляции для соответствующих LDPC-кодов с длиной N кодов, равной 64800, и 11 кодовыми скоростями, которые определены в стандарте DVB-T.2.

Если множитель b равен 1, и число m битов одного символа равно двум битам при использовании, например, QPSK в качестве схемы модуляции согласно фиг. 29, память 31 имеет два столбца, хранящих $2 \times 1 (=mb)$ битов в направлении строки, и хранит 64800/(2×1) битов в направлении столбца.

Затем среди двух столбцов памяти 31 предполагается, что позиция начала записи первого столбца представляет собой позицию адреса 0, и позиция начала записи второго столбца представляет собой позицию адреса 2.

В дополнение к этому, например, когда любой один из первого-третьего способа замены (фиг. 22) приняты в качестве способа замены для процесса замены демультимплексора 25 (фиг. 9), множитель b равен 1.

Если множитель b равен 2, и число m битов одного символа равно двум битам при использовании, например, QPSK в качестве схемы модуляции согласно фиг. 29, память 31 имеет четыре столбца, хранящих 2×2 битов в направлении строки, и хранит 64800/(2×2) битов в направлении столбца.

Затем среди четырех столбцов памяти 31 предполагается, что позиция начала записи первого столбца представляет собой позицию адреса 0, позиция начала записи второго столбца представляет собой позицию адреса 2, позиция начала записи третьего столбца представляет собой позицию адреса 4 и позиция начала записи четвертого столбца

представляет собой позицию адреса 7.

В дополнение к этому, например, когда четвертый способ замены (фиг. 23) принят в качестве способа замены для процесса замены демультимплексора 25 (фиг. 9), множитель b равен 2.

5 Если множитель b равен 1, и число m битов одного символа равно четырем битам при использовании, например, 16QAM в качестве схемы модуляции согласно фиг. 29, память 31 имеет четыре столбца, хранящих 4×1 битов в направлении строки, и хранит $64800/(4 \times 1)$ битов в направлении столбца.

10 Затем среди четырех столбцов памяти 31 предполагается, что позиция начала записи первого столбца представляет собой позицию адреса 0, позиция начала записи второго столбца представляет собой позицию адреса 2, позиция начала записи третьего столбца представляет собой позицию адреса 4 и позиция начала записи четвертого столбца представляет собой позицию адреса 7.

15 Если множитель b равен 2, и число m битов одного символа равно четырем битам при использовании, например, 16QAM в качестве схемы модуляции согласно фиг. 29, память 31 имеет восемь столбцов, хранящих 4×2 битов в направлении строки, и хранит $64800/(4 \times 2)$ битов в направлении столбца.

20 Затем среди восьми столбцов памяти 31 предполагается, что позиция начала записи первого столбца представляет собой позицию адреса 0, позиция начала записи второго столбца представляет собой позицию адреса 0, позиция начала записи третьего столбца представляет собой позицию адреса 2, позиция начала записи четвертого столбца представляет собой позицию адреса 4, позиция начала записи пятого столбца представляет собой позицию адреса 4, позиция начала записи шестого столбца представляет собой позицию адреса 5, позиция начала записи седьмого столбца
25 представляет собой позицию адреса 7, и позиция начала записи восьмого столбца представляет собой позицию адреса 7.

30 Если множитель b равен 1, и число m битов одного символа равно шести битам при использовании, например, 64QAM в качестве схемы модуляции согласно фиг. 29, память 31 имеет шесть столбцов, хранящих 6×1 битов в направлении строки, и хранит $64800/(6 \times 1)$ битов в направлении столбца.

35 Затем среди шести столбцов памяти 31 предполагается, что позиция начала записи первого столбца представляет собой позицию адреса 0, позиция начала записи второго столбца представляет собой позицию адреса 2, позиция начала записи третьего столбца представляет собой позицию адреса 5, позиция начала записи четвертого столбца представляет собой позицию адреса 9, позиция начала записи пятого столбца представляет собой позицию адреса 10 и позиция начала записи шестого столбца представляет собой позицию адреса 13.

40 Если множитель b равен 2, и число m битов одного символа равно шести битам при использовании, например, 64QAM в качестве схемы модуляции в соответствии с фиг. 29, память 31 имеет 12 столбцов, хранящих 6×2 битов в направлении строки, и хранит $64800/(6 \times 2)$ битов в направлении столбца.

45 Затем среди 12 столбцов памяти 31 предполагается, что позиция начала записи первого столбца представляет собой позицию адреса 0, позиция начала записи второго столбца представляет собой позицию адреса 0, позиция начала записи третьего столбца представляет собой позицию адреса 2, позиция начала записи четвертого столбца представляет собой позицию адреса 2, позиция начала записи пятого столбца представляет собой позицию адреса 3, позиция начала записи шестого столбца представляет собой позицию адреса 4, позиция начала записи седьмого столбца

представляет собой позицию адреса 4, позиция начала записи восьмого столбца представляет собой позицию адреса 5, позиция начала записи девятого столбца представляет собой позицию адреса 5, позиция начала записи десятого столбца представляет собой позицию адреса 7, позиция начала записи 11-го столбца представляет собой позицию адреса 8, и позиция начала записи 12-го столбца представляет собой позицию адреса 9.

Если множитель b равен 1, и число m битов одного символа равно 8 битам при использовании, например, 256QAM в качестве схемы модуляции согласно фиг. 29, память 31 имеет 8 столбцов, хранящих 8×1 битов в направлении строки, и хранит $64800/(8 \times 1)$ битов в направлении столбца.

Затем среди 8 столбцов памяти 31 предполагается, что позиция начала записи первого столбца представляет собой позицию адреса 0, позиция начала записи второго столбца представляет собой позицию адреса 0, позиция начала записи третьего столбца представляет собой позицию адреса 2, позиция начала записи четвертого столбца представляет собой позицию адреса 4, позиция начала записи пятого столбца представляет собой позицию адреса 4, позиция начала записи шестого столбца представляет собой позицию адреса 5, позиция начала записи седьмого столбца представляет собой позицию адреса 7, позиция начала записи восьмого столбца представляет собой позицию адреса 7.

Если множитель b равен 2, и число m битов одного символа равно 8 битам при использовании, например, 256QAM в качестве схемы модуляции согласно фиг. 29, память 31 имеет 16 столбцов, хранящих 8×2 битов в направлении строки, и хранит $64800/(8 \times 2)$ битов в направлении столбца.

Затем среди 16 столбцов памяти 31 предполагается, что позиция начала записи первого столбца представляет собой позицию адреса 0, позиция начала записи второго столбца представляет собой позицию адреса 2, позиция начала записи третьего столбца представляет собой позицию адреса 2, позиция начала записи четвертого столбца представляет собой позицию адреса 2, позиция начала записи пятого столбца представляет собой позицию адреса 2, позиция начала записи шестого столбца представляет собой позицию адреса 3, позиция начала записи седьмого столбца представляет собой позицию адреса 7, позиция начала записи восьмого столбца представляет собой позицию адреса 15, позиция начала записи девятого столбца представляет собой позицию адреса 16, позиция начала записи десятого столбца представляет собой позицию адреса 20, позиция начала записи 11-го столбца представляет собой позицию адреса 22, позиция начала записи 12-го столбца представляет собой позицию адреса 22, позиция начала записи 13-го столбца представляет собой позицию адреса 27, позиция начала записи 14-го столбца представляет собой позицию адреса 27, позиция начала записи 15-го столбца представляет собой позицию адреса 28, и позиция начала записи 16-го столбца представляет собой позицию адреса 32.

Если множитель b равен 1, и число m битов одного символа равно 10 битам при использовании, например, 1024QAM в качестве схемы модуляции согласно фиг. 29, память 31 имеет 10 столбцов, хранящих 10×1 битов в направлении строки, и хранит $64800/(10 \times 1)$ битов в направлении столбца.

Затем среди 10 столбцов памяти 31 предполагается, что позиция начала записи первого столбца представляет собой позицию адреса 0, позиция начала записи второго столбца представляет собой позицию адреса 3, позиция начала записи третьего столбца представляет собой позицию адреса 6, позиция начала записи четвертого столбца

представляет собой позицию адреса 8, позиция начала записи пятого столбца представляет собой позицию адреса 11, позиция начала записи шестого столбца представляет собой позицию адреса 13, позиция начала записи седьмого столбца представляет собой позицию адреса 15, позиция начала записи восьмого столбца представляет собой позицию адреса 17, позиция начала записи девятого столбца представляет собой позицию адреса 18, и позиция начала записи десятого столбца представляет собой позицию адреса 20.

Если множитель b равен 2, и число m битов одного символа равно 10 битам при использовании, например, 1024QAM в качестве схемы модуляции согласно фиг. 29, память 31 имеет 20 столбцов, хранящих 10×2 битов в направлении строки, и хранит $64800/(10 \times 2)$ битов в направлении столбца.

Затем среди 20 столбцов памяти 31 предполагается, что позиция начала записи первого столбца представляет собой позицию адреса 0, позиция начала записи второго столбца представляет собой позицию адреса 1, позиция начала записи третьего столбца представляет собой позицию адреса 3, позиция начала записи четвертого столбца представляет собой позицию адреса 4, позиция начала записи пятого столбца представляет собой позицию адреса 5, позиция начала записи шестого столбца представляет собой позицию адреса 6, позиция начала записи седьмого столбца представляет собой позицию адреса 6, позиция начала записи восьмого столбца представляет собой позицию адреса 9, позиция начала записи девятого столбца представляет собой позицию адреса 13, позиция начала записи десятого столбца представляет собой позицию адреса 14, позиция начала записи 11-го столбца представляет собой позицию адреса 14, позиция начала записи 12-го столбца представляет собой позицию адреса 16, позиция начала записи 13-го столбца представляет собой позицию адреса 21, позиция начала записи 14-го столбца представляет собой позицию адреса 21, позиция начала записи 15-го столбца представляет собой позицию адреса 23, позиция начала записи 16-го столбца представляет собой позицию адреса 25, позиция начала записи 17-го столбца представляет собой позицию адреса 25, позиция начала записи 18-го столбца представляет собой позицию адреса 26, позиция начала записи 19-го столбца представляет собой позицию адреса 28, и позиция начала записи 20-го столбца представляет собой позицию адреса 30.

Если множитель b равен 1, и число m битов одного символа равно 12 битам при использовании, например, 4096QAM в качестве схемы модуляции согласно фиг. 29, память 31 имеет 12 столбцов, хранящих 12×1 битов в направлении строки, и хранит $64800/(12 \times 1)$ битов в направлении столбца.

Затем среди 12 столбцов памяти 31 предполагается, что позиция начала записи первого столбца представляет собой позицию адреса 0, позиция начала записи второго столбца представляет собой позицию адреса 0, позиция начала записи третьего столбца представляет собой позицию адреса 2, позиция начала записи четвертого столбца представляет собой позицию адреса 2, позиция начала записи пятого столбца представляет собой позицию адреса 3, позиция начала записи шестого столбца представляет собой позицию адреса 4, позиция начала записи седьмого столбца представляет собой позицию адреса 4, позиция начала записи восьмого столбца представляет собой позицию адреса 5, позиция начала записи девятого столбца представляет собой позицию адреса 5, позиция начала записи десятого столбца представляет собой позицию адреса 7, позиция начала записи 11-го столбца представляет собой позицию адреса 8, и позиция начала записи 12-го столбца представляет собой

позицию адреса 9.

Если множитель b равен 2, и число m битов одного символа равно 12 битам при использовании, например, 4096QAM в качестве схемы модуляции согласно фиг. 29, память 31 имеет 24 столбца, хранящих 12×2 битов в направлении строки, и хранит
5 $64800/(12 \times 2)$ битов в направлении столбца.

Затем среди 24 столбцов памяти 31 предполагается, что позиция начала записи первого столбца представляет собой позицию адреса 0, позиция начала записи второго столбца представляет собой позицию адреса 5, позиция начала записи третьего столбца представляет собой позицию адреса 8, позиция начала записи четвертого столбца
10 представляет собой позицию адреса 8, позиция начала записи пятого столбца представляет собой позицию адреса 8, позиция начала записи шестого столбца представляет собой позицию адреса 8, позиция начала записи седьмого столбца представляет собой позицию адреса 10, позиция начала записи восьмого столбца представляет собой позицию адреса 10, позиция начала записи девятого столбца
15 представляет собой позицию адреса 10, позиция начала записи десятого столбца представляет собой позицию адреса 12, позиция начала записи 11-го столбца представляет собой позицию адреса 13, позиция начала записи 12-го столбца представляет собой позицию адреса 16, позиция начала записи 13-го столбца представляет собой позицию адреса 17, позиция начала записи 14-го столбца
20 представляет собой позицию адреса 19, позиция начала записи 15-го столбца представляет собой позицию адреса 21, позиция начала записи 16-го столбца представляет собой позицию адреса 22, позиция начала записи 17-го столбца представляет собой позицию адреса 23, позиция начала записи 18-го столбца представляет собой позицию адреса 26, позиция начала записи 19-го столбца
25 представляет собой позицию адреса 37, позиция начала записи 20-го столбца представляет собой позицию адреса 39, позиция начала записи 21-го столбца представляет собой позицию адреса 40, позиция начала записи 22-го столбца представляет собой позицию адреса 41, позиция начала записи 23-го столбца
30 представляет собой позицию адреса 41, и позиция начала записи 24-го столбца представляет собой позицию адреса 41.

Фиг. 30 иллюстрирует требуемое число столбцов памяти 31 для перемежения со скручиванием столбцов и адрес позиции начала записи для каждой схемы модуляции для соответствующих LDPC-кодов с длиной N кода, равной 16200, и 10 кодовыми скоростями, которые определены в стандарте DVB-T.2.

Если множитель b равен 1, и число m битов одного символа равно 2 битам при использовании, например, QPSK в качестве схемы модуляции согласно фиг. 30, память 31 имеет два столбца, хранящих 2×1 битов в направлении строки, и хранит $16200/(2 \times 1)$ битов в направлении столбца.

Затем среди двух столбцов памяти 31 предполагается, что позиция начала записи
40 первого столбца представляет собой позицию адреса 0, и позиция начала записи второго столбца представляет собой позицию адреса 0.

Если множитель b равен 2, и число m битов одного символа равно 2 битам при использовании, например, QPSK в качестве схемы модуляции согласно фиг. 30, память 31 имеет 4 столбца, хранящих 2×2 битов в направлении строки, и хранит $16200/(2 \times 2)$
45 битов в направлении столбца.

Затем среди четырех столбцов памяти 31 предполагается, что позиция начала записи первого столбца представляет собой позицию адреса 0, позиция начала записи второго столбца представляет собой позицию адреса 2, позиция начала записи третьего столбца

представляет собой позицию адреса 3, и позиция начала записи четвертого столбца представляет собой позицию адреса 3.

Если множитель b равен 1, и число m битов одного символа равно 4 битам при использовании, например, 16QAM в качестве схемы модуляции согласно фиг. 30, память 31 имеет 4 столбца, хранящих 4×1 битов в направлении строки, и хранит $16200/(4 \times 1)$ битов в направлении столбца.

Затем среди четырех столбцов памяти 31 предполагается, что позиция начала записи первого столбца представляет собой позицию адреса 0, позиция начала записи второго столбца представляет собой позицию адреса 2, позиция начала записи третьего столбца представляет собой позицию адреса 3, и позиция начала записи четвертого столбца представляет собой позицию адреса 3.

Если множитель b равен 2, и число m битов одного символа равно 4 битам при использовании, например, 16QAM в качестве схемы модуляции согласно фиг. 30, память 31 имеет 8 столбцов, хранящих 4×2 битов в направлении строки, и хранит $16200/(4 \times 2)$ битов в направлении столбца.

Затем среди восьми столбцов памяти 31 предполагается, что позиция начала записи первого столбца представляет собой позицию адреса 0, позиция начала записи второго столбца представляет собой позицию адреса 0, позиция начала записи третьего столбца представляет собой позицию адреса 0, позиция начала записи четвертого столбца представляет собой позицию адреса 1, позиция начала записи пятого столбца представляет собой позицию адреса 7, позиция начала записи шестого столбца представляет собой позицию адреса 20, позиция начала записи седьмого столбца представляет собой позицию адреса 20, и позиция начала записи восьмого столбца представляет собой позицию адреса 21.

Если множитель b равен 1, и число m битов одного символа равно 6 битам при использовании, например, 64QAM в качестве схемы модуляции согласно фиг. 30, память 31 имеет 6 столбцов, хранящих 6×1 битов в направлении строки, и хранит $16200/(6 \times 1)$ битов в направлении столбца.

Затем среди шести столбцов памяти 31 предполагается, что позиция начала записи первого столбца представляет собой позицию адреса 0, позиция начала записи второго столбца представляет собой позицию адреса 0, позиция начала записи третьего столбца представляет собой позицию адреса 2, позиция начала записи четвертого столбца представляет собой позицию адреса 3, позиция начала записи пятого столбца представляет собой позицию адреса 7, и позиция начала записи шестого столбца представляет собой позицию адреса 7.

Если множитель b равен 2, и число m битов одного символа равно 6 битам при использовании, например, 64QAM в качестве схемы модуляции согласно фиг. 30, память 31 имеет 12 столбцов, хранящих 6×2 битов в направлении строки, и хранит $16200/(6 \times 2)$ битов в направлении столбца.

Затем среди 12 столбцов памяти 31 предполагается, что позиция начала записи первого столбца представляет собой позицию адреса 0, позиция начала записи второго столбца представляет собой позицию адреса 0, позиция начала записи третьего столбца представляет собой позицию адреса 0, позиция начала записи четвертого столбца представляет собой позицию адреса 2, позиция начала записи пятого столбца представляет собой позицию адреса 2, позиция начала записи шестого столбца представляет собой позицию адреса 2, позиция начала записи седьмого столбца представляет собой позицию адреса 3, позиция начала записи восьмого столбца представляет собой позицию адреса 3, позиция начала записи девятого столбца

представляет собой позицию адреса 3, позиция начала записи десятого столбца представляет собой позицию адреса 6, позиция начала записи 11-го столбца представляет собой позицию адреса 7, и позиция начала записи 12-го столбца представляет собой позицию адреса 7.

5 Если множитель b равен 1, и число m битов одного символа равно 8 битам при использовании, например, 256QAM в качестве схемы модуляции согласно фиг. 30, память 31 имеет 8 столбцов, хранящих 8×1 битов в направлении строки, и хранит $16200/(8 \times 1)$ битов в направлении столбца.

10 Затем среди восьми столбцов памяти 31 предполагается, что позиция начала записи первого столбца представляет собой позицию адреса 0, позиция начала записи второго столбца представляет собой позицию адреса 0, позиция начала записи третьего столбца представляет собой позицию адреса 0, позиция начала записи четвертого столбца представляет собой позицию адреса 1, позиция начала записи пятого столбца представляет собой позицию адреса 7, позиция начала записи шестого столбца
15 представляет собой позицию адреса 20, позиция начала записи седьмого столбца представляет собой позицию адреса 20, и позиция начала записи восьмого столбца представляет собой позицию адреса 21.

Если множитель b равен 1, и число m битов одного символа равно 10 битам при использовании, например, 1024QAM в качестве схемы модуляции согласно фиг. 30,
20 память 31 имеет 10 столбцов, хранящих 10×1 битов в направлении строки, и хранит $16200/(10 \times 1)$ битов в направлении столбца.

Затем среди десяти столбцов памяти 31 предполагается, что позиция начала записи первого столбца представляет собой позицию адреса 0, позиция начала записи второго столбца представляет собой позицию адреса 1, позиция начала записи третьего столбца
25 представляет собой позицию адреса 2, позиция начала записи четвертого столбца представляет собой позицию адреса 2, позиция начала записи пятого столбца представляет собой позицию адреса 3, позиция начала записи шестого столбца представляет собой позицию адреса 3, позиция начала записи седьмого столбца представляет собой позицию адреса 4, позиция начала записи восьмого столбца
30 представляет собой позицию адреса 4, позиция начала записи девятого столбца представляет собой позицию адреса 5, и позиция начала записи десятого столбца представляет собой позицию адреса 7.

Если множитель b равен 2, и число m битов одного символа равно 10 битам при использовании, например, 1024QAM в качестве схемы модуляции согласно фиг. 30,
35 память 31 имеет 20 столбцов, хранящих 10×2 битов в направлении строки, и хранит $16200/(10 \times 2)$ битов в направлении столбца.

Затем среди 20 столбцов памяти 31 предполагается, что позиция начала записи первого столбца представляет собой позицию адреса 0, позиция начала записи второго столбца представляет собой позицию адреса 0, позиция начала записи третьего столбца
40 представляет собой позицию адреса 0, позиция начала записи четвертого столбца представляет собой позицию адреса 2, позиция начала записи пятого столбца представляет собой позицию адреса 2, позиция начала записи шестого столбца представляет собой позицию адреса 2, позиция начала записи седьмого столбца представляет собой позицию адреса 2, позиция начала записи восьмого столбца
45 представляет собой позицию адреса 2, позиция начала записи девятого столбца представляет собой позицию адреса 5, позиция начала записи десятого столбца представляет собой позицию адреса 5, позиция начала записи 11-го столбца представляет собой позицию адреса 5, позиция начала записи 12-го столбца представляет собой

представляет собой позицию адреса 10, позиция начала записи 21-го столбца представляет собой позицию адреса 10, позиция начала записи 22-го столбца представляет собой позицию адреса 10, позиция начала записи 23-го столбца представляет собой позицию адреса 10, и позиция начала записи 24-го столбца

5 представляет собой позицию адреса 11.

На фиг. 31 показана блок-схема последовательности операций, иллюстрирующая процесс, выполняемый LDPC-кодером 115, перемежителем 116 битов и блоком 117 отображения (фиг. 8).

LDPC-кодер 115 принимает целевые LDPC-данные, подаваемые из BCH-кодера 114, LDPC-кодирует целевые LDPC-данные в LDPC-код на этапе S101 и подает LDPC-код в перемежитель 116 битов, и процесс переходит на этап S102.

Перемежитель 116 битов на этапе S102 выполняет перемежение битов над LDPC-кодом, поступающим из LDPC-кодера 115, подает символ, полученный путем символизации LDPC-кода после перемежения битов в блок 117 отображения, и процесс

15 переходит на этап S103.

Другими словами, на этапе S102 в перемежитель 116 битов (фиг. 9), перемежитель 23 битов четности выполняет перемежение битов четности над LDPC-кодом, поступающим из LDPC-кодера 115, и подает LDPC-код после перемежения битов четности в перемежитель 24 со скручиванием столбцов.

Перемежитель 24 со скручиванием столбцов выполняет перемежение со скручиванием столбцов над LDPC-кодом, поступающим из перемежителя 23 битов четности, и подает его в демультиплексор 25.

Демультиплексор 25 выполняет процесс замены, предназначенный для замены кодового бита LDPC-кода, после перемежения со скручиванием столбцов, выполняемого

25 перемежителем 24 со скручиванием столбцов, и устанавливает кодовый бит после замены в качестве символьного бита символа (бит, который представляет собой символ).

В данном случае, процесс замены, выполняемый демультиплексором 25, можно выполнить согласно первому-четвертому способам замены, иллюстрированным на фиг. 22 и фиг. 23, и можно выполнить согласно другому способу замены.

Символ, полученный в процессе замены, выполняемой демультиплексором 25, подается в блок 117 отображения из демультиплексора 25.

На этапе S103 блок 117 отображения отображает символ, поступающий из демультиплексора 25, в сигнальную точку, которая определяется с помощью схемы модуляции квадратурной модуляции, выполняемой блоком 117 отображения, для того,

35 чтобы выполнить квадратурную модуляцию, и подает результирующие данные во временной перемежитель 118.

Можно улучшить устойчивость к стиранию или пакетной ошибке при передаче множества кодовых битов LDPC-кода в качестве одного символа путем выполнения перемежения битов четности и перемежения со скручиванием столбцов, как описано

40 выше.

В данном случае, на фиг. 9 для удобства описания перемежитель 23 битов четности, который представляет собой блок, выполняющий перемежение битов четности, и перемежитель 24 со скручиванием столбцов, который представляет собой блок, выполняющий перемежение со скручиванием столбцов, сконфигурированы отдельно,

45 но перемежитель 23 битов четности и перемежитель 24 со скручиванием столбцов можно сконфигурировать как единое целое.

Другими словами, как перемежение битов четности, так и перемежение со скручиванием столбцов можно выполнить путем считывания и записи кодовых битов

в память, и можно представить с помощью матрицы, которая преобразует адрес для записи кодовых битов (адрес записи) в адрес для считывания кодовых битов (адрес считывания).

Поэтому, если матрица, представляющая перемежение битов четности, и матрица, представляющая перемежение со скручиванием столбцов, перемножаются для того, чтобы получить матрицу и кодовый бит преобразуется с использованием матрицы, можно достичь результирующего LDPC-кода путем выполнения перемежения битов четности над LDPC-кодом и выполнения перемежения со скручиванием столбцов над LDPC-кодом, который был подвергнут перемежению битов четности.

Кроме того, можно совместно сконфигурировать демультимплексор 25 в дополнение к перемежителю 23 битов четности и перемежителю 24 со скручиванием столбцов.

Другими словами, можно представить процесс замены, выполняемый в демультимплексоре 25 с помощью матрицы, полученной путем преобразования адреса записи памяти 31, хранящей LDPC-код в адресе считывания.

Поэтому, если матрица получается путем мультиплексирования матрицы, представляющей собой перемежение битов четности, матрицы, представляющей собой перемежение со скручиванием столбцов, и матрицы представляющей собой процесс замены, можно обобщенно выполнить перемежение битов по четности, перемежение со скручиванием столбцов и процесс замены с использованием матрицы.

В дополнение к этому, можно выполнить только одно или не одно из перемежения битов четности и перемежения со скручиванием столбцов. Например, когда канал 13 связи (фиг. 7) представляет собой спутниковую линию или т.п., в которой не нужно рассматривать пакетную ошибку и фазовое дрожание и т.п., которая отличается от AWGN, в качестве DVB-S.2, перемежение битов четности и перемежение со скручиванием столбцов не выполняются.

Затем выполняется моделирование измерения частоты появления ошибок (частоты появления ошибочных битов), которое выполняется для передающего устройства 11, показанного на фиг. 8, будет описано со ссылкой на фиг. 32 и фиг. 34.

Моделирование выполняется при использовании канала связи, имеющего фазовое дрожание D/U 0 дБ.

На фиг. 32 показана схема, иллюстрирующая модель канала связи, используемую при моделировании.

Другими словами, поз. А на фиг. 32 иллюстрирует модель фазового дрожания, используемого при моделировании.

Кроме того, поз. В на фиг. 32 иллюстрирует модель канала связи, имеющего фазовое дрожание, представленное с помощью модели (поз. А на фиг. 32).

В дополнение к этому, в поз. В на фиг. 32 Н представляет собой модель фазового дрожания (фиг. 32). Кроме того, в поз. В на фиг. 32 N представляет собой интерференцию между, несущими (ICI), и при моделировании ожидаемое значение $E[N^2]$ мощности аппроксимируется AWGN.

На фиг. 33 и фиг. 34 показаны зависимости между частотой появления ошибок, полученный с помощью моделирования, и доплеровской частотой f_d фазового дрожания.

В дополнение к этому, на фиг. 33 показана зависимость между частотой появления ошибок и доплеровской частотой f_d , когда схемой модуляции является 16QAM, кодовая скорость (r) равна (3/4), и способ замены представляет собой первый способ замены. Кроме того, на фиг. 34 показана зависимость между частотой появления ошибок и доплеровской частотой f_d , когда схемой модуляции является 64QAM, кодовая скорость

(г) равна (5/6), и способ замены представляет собой первый способ замены.

Кроме того, на фиг. 33 и фиг. 34 жирная линия показывает зависимость между частотой появления ошибок и доплеровской частотой f_d в случае выполнения всех следующих процессов: перемежение битов четности, перемежение со скручиванием столбцов и процесс замены, и тонкая линия показывает зависимость между частотой появления ошибок и доплеровской частотой f_d в случае выполнения только процесса замены среди перемежения битов четности, перемежения со скручиванием столбцов и процесса замены.

На фиг. 33 или фиг. 34 установлено, что частота появления ошибок является более повышенной (пониженной) в случае выполнения всего из перемежения битов четности, перемежения со скручиванием столбцов и процесса замены, чем в случае выполнения только процесса замены.

Пример конфигурации LDPC-кодера 115

На фиг. 35 показана блок-схема, иллюстрирующая пример конфигурации LDPC-кодера 115 (фиг. 8).

В дополнение к этому, LDPC-кодер 122, показанный на фиг. 8, сконфигурирован аналогичным образом.

Как показано на фиг. 12 и фиг. 13, в стандартах DVB-S.2 и т.п. определены LDPC-коды двух типов длиной N кода 64800 битов и 16200 битов.

11 кодовых скоростей 1/4, 1/3, 2/5, 1/2, 3/5, 2/3, 3/4, 4/5, 5/6, 8/9 и 9/10 определены для LDPC-кода с длиной N кода 64800 битов, и 10 кодовых скоростей 1/4, 1/3, 2/5, 1/2, 3/5, 2/3, 3/4, 4/5, 5/6 и 8/9 определены для LDPC-кода с длиной N кода 16200 битов (фиг. 12 и фиг. 13).

LDPC-кодер 115 может выполнять кодирование (кодирование с коррекцией ошибок) с помощью LDPC-кода с каждой кодовой скоростью и длинами N кодов 64800 битов и 16200 битов для каждой кодовой длины N согласно матрице H контроля четности, предусмотренной для каждой кодовой скорости.

LDPC-кодер 115 выполнен с блоком 601 обработки кодирования и блоком 602 хранения.

Блок 601 обработки кодирования выполнен с блоком 611 установки кодовой скорости, блоком 612 считывания таблицы начальных значений, блоком 613 выработки матрицы контроля четности, блоком 614 считывания информационного бита, блоком 615 вычисления четности кодирования и блоком управления 616, выполняет LDPC-кодирование над целевыми LDPC-данными, которые подаются в LDPC-кодер 115, и подает результирующий LDPC-код в перемежитель 116 битов (фиг. 8).

Другими словами, блок 611 установки кодовой скорости устанавливает длину N кода и кодовую скорость LDPC-кода в ответ на, например, операции оператора или т.п.

Блок 612 считывания таблицы начальных значений считывает таблицу начальных значений матрицы контроля четности, которая будет описана позже и которая соответствует длине N кода и кодовой скорости, которые устанавливаются с помощью блока 611 установки кодовой скорости, из блока 602 хранения.

Блок 613 выработки матрицы контроля четности вырабатывает матрицу H контроля четности и хранит матрицу H в блоке 602 хранения путем размещения элементов 1 в информационной матрице H_A , соответствующей информационной длине K (= длина N кода - длина битов M четности) согласно длине N кода и кодовой скорости, которые устанавливаются с помощью блока 611 установки кодовой скорости на основании

таблицы начальных значений матрицы контроля четности, считываемой с помощью блока 612 считывания таблицы начальных значений в направлении столбца в периоде из 360 столбцов (число p столбцов блока циклической структуры).

Блок 614 считывания информационного бита считывает (извлекает) информационный бит с информационной длиной K , из целевых LDPC-данных поданный в LDPC-кодер 115.

Блок 615 вычисления четности кодирования вырабатывает кодовое слово (LDPC-код) путем считывания матрица H контроля четности, выработанная с помощью матрицы контроля четности генерирующая оп блок 613 из блока 602 хранения, и вычисления бит четности для информационного бита, который считывается с помощью блока 614 считывания информационного бита на основании предварительно определенной уравнение, с использованием матрицы H контроля четности.

Блок управления 616 управляет соответствующими блоками, конфигурирующими блок 601 обработки кодирования.

Блок 602 хранения хранит, например, множество таблиц начальных значений матрицы контроля четности, соответствующей каждой из множества кодовых скоростей, показанных на фиг. 12 и фиг. 13, для каждой длины N кода 64800 битов и 16200 битов. Кроме того, блок 602 хранения временно сохраняет данные, требуемые для процесса, выполняемого блоком 601 обработки кодирования.

На фиг. 36 показана блок-схема последовательности операций, иллюстрирующая процесс, выполняемый кодером 115 LDPC (фиг. 35).

На этапе S201 блок 611 установки кодовой скорости определяет (устанавливает) длину N кода и кодовую скорость r для выполнения LDPC-кодирования.

На этапе S202 блок 612 считывания таблицы начальных значений считывает таблицу начальных значений матрицы контроля четности, которая устанавливается заранее и которая соответствует длине N кода и кодовой скорости r , которые определяются с помощью блока 611 установки кодовой скорости, из блока 602 хранения.

На этапе S203 блок 613 выработки матрицы контроля четности получает (вырабатывает) матрицу H контроля четности LDPC-кода длиной N кода и кодовой скоростью r , которые определяются с помощью блока 611 установки кодовой скорости, с использованием таблицы начальных значений матрицы контроля четности, которая считывается из блока 602 хранения с помощью блока 612 считывания таблицы начальных значений, и подает и сохраняет матрицу H контроля четности в блоке 602 хранения.

На этапе S204 блок 614 считывания информационного бита считывает информационный бит с информационной длиной K ($=N \times r$), соответствующей длине N кода и кодовой скорости r , которые определяются с помощью блока 611 установки кодовой скорости из целевых LDPC-данных, которые подаются в LDPC-кодер 115, считывает матрицу H контроля четности, которая получается с помощью блока 613 выработки матрицы контроля четности, из блока 602 хранения, и подает информационный бит и матрицу H контроля четности в блок 615 вычисления четности кодирования.

На этапе S205 блок 615 вычисления четности кодирования последовательно вычисляет бит четности кодового слова s , удовлетворяющего уравнению (8), с использованием информационного бита и матрицы H контроля четности из блока 614 считывания информационного бита.

$$Hs^T = 0 \dots (8)$$

В уравнении (8) s представляет собой вектор-строку в качестве кодового слова

(LDPC-кода), и s^T представляет собой вектор, полученный путем транспонирования вектор-строки s .

В данном случае, как описано выше, когда часть информационного бита представлена вектор-строкой A , и часть бита четности представлена вектор-строкой T , в вектор-строке s , которая представляет собой LDPC-код (1 кодовое слово), вектор-строку s можно представить с помощью уравнения $s=[A|T]$, с помощью вектор-строки в качестве информационного бита и вектор-строки T в качестве бита четности.

Матрица H контроля четности и вектор-строка $s=[A|T]$ в виде LDPC-кода должны удовлетворять уравнению $Hs^T=0$, и когда матрица H_T четности матрицы $H=[H_A|H_T]$ контроля четности имеет ступенчатую структуру, показанную на фиг. 11, вектор-строку T в качестве бита четности, конфигурирующего вектор-строку $s=[A|T]$, удовлетворяющую уравнению $Hs^T=0$, можно получить последовательно путем выработки элементов каждой строки 0 надлежащим образом из элемента в первой строке вектора-столбца Hs^T в уравнении $Hs^T=0$.

Блок 615 вычисления четности кодирования получает бит четности T для информационный, бит из блока 614 считывания информационного бита, и выводит кодовое слово $s=[A|T]$, представленное с помощью информационного бита и бита четности T , в качестве LDPC-кодирования результат информационного бита A .

После этого на этапе S206 блок управления 616 определяет, закончено или нет LDPC-кодирование. На этапе S206, когда определено, что LDPC-кодирование не завершено, другими словами, например, когда еще есть LDPC целевые данные, которые необходимо подвергнуть LDPC-кодированию, процесс возвращается на этап S201 (или этап S204), и после этого процесс повторяется от этапа S201 (или этапа S204) до этапа S206.

Кроме того, на этапе S206, когда определено, что LDPC-кодирование закончено, другими словами, например, когда не существуют целевые LDPC-данные, которые необходимо подвергнуть LDPC-кодированию, LDPC-кодер 115 заканчивается процесс.

Как описано выше, когда таблица начальных значений матрицы контроля четности, соответствующая каждой длине N кода и каждой кодовой скорости r , подготовлена, LDPC-кодер 115 выполняет LDPC-кодирование с предварительно определенной кодовой скоростью r и предварительно определенной длиной N кода с использованием матрицы H контроля четности, выработанной из таблицы начальных значений матрицы контроля четности, соответствующей предварительно определенной длине N кода и предварительно определенной кодовой скорости r .

Пример таблицы начальных значений матрицы контроля четности

Таблица начальных значений матрицы контроля четности представляет собой таблицу, которая представляет собой позиции элементов 1 информационной матрицы H_A (фиг. 10), соответствующей информационной длине K , соответствующей длине N кода и кодовой скорости r LDPC-кода матрицы H контроля четности (LDPC-код, определенный с помощью матрицы H контроля четности) с интервалом 360 столбцов (число P столбцов в блоках циклической структуры), и создается заранее для каждой матрицы H контроля четности каждой длины N кода и каждой кодовой скорости r .

На фиг. 37 показана схема, иллюстрирующая пример таблицы начальных значений матрицы контроля четности.

Другими словами, на фиг. 37 показана таблица начальных значений матрицы контроля четности для матрицы H контроля четности, которая определена в стандарте DVB-T.2, и чья длина N кода равна 16200 битов, и кодовая скорость r (кодовая скорость

по определению DVB-T.2) равна 1/4.

Блок 613 выработки матрицы контроля четности (фиг. 35) получает матрицу Н контроля четности с использованием таблицы начальных значений матрицы контроля четности следующим образом.

5 На фиг. 38 показана схема, изображающая способ получения матрицы Н контроля четности из таблицы начальных значений матрицы контроля четности.

Другими словами, на фиг. 38 представлена таблица начальных значений матрицы контроля четности для матрицы Н контроля четности, которая определена в стандарте DVB-T.2, и чья длина N кода равна 16200 битов, и кодовая скорость r равна 2/3.

10 Как описано выше, таблица начальных значений матрицы контроля четности представляет собой таблицу, представляющую позиции элементов 1 в информационной матрице H_A (фиг. 10), соответствующей информационным битам с длиной K в соответствии с длиной N кода и кодовой скоростью r LDPC-кода с интервалом в 360 столбцов (число P столбцов в блоке циклической структуры), и номера строк (номера строк при условии, что номер строки первой строки матрицы Н контроля четности равен 0) элементов 1 в $(1+360 \times (i-1))$ -ом столбце матрицы Н контроля четности размещаются по числу веса столбца $(1+360 \times (i-1))$ -го столбца, в i-ой строке.

15 В данном случае, поскольку матрица H_T четности (фиг. 10), соответствующей длине битов четности M, матрицы Н контроля четности определяется как показано на фиг. 25, в соответствии с таблицей начальных значений матрицы контроля четности, информационная матрица H_A (фиг. 10), соответствующей информационной длине K, матрицы Н контроля четности получается.

20 Число k+1 строк таблицы начальных значений матрицы контроля четности изменяется в зависимости от информационной длины K.

25 Между информационными битами с длиной K и числом k+1 строк таблицы начальных значений матрицы контроля четности устанавливается зависимость согласно уравнению (9).

$$K=(k+1) \times 360 \dots (9)$$

30 В данном случае, 360 в уравнении (9) представляет собой число P столбцов блока циклической структуры, показанной на фиг. 26.

В таблице начальных значений матрицы контроля четности (фиг. 38) 13 чисел размещаются с первой строки по третью строку, и три числа размещаются с четвертой строки по (k+1)-ую строку (на фиг. 38 вплоть до 30-ой строки).

35 Поэтому вес столбца матрицы Н контроля четности, полученный из таблицы начальных значений матрицы контроля четности (фиг. 38), равен 13 с первой строки по $\{1+360 \times (3-1)-1\}$ -ую строку, и равен 3 с $(1+360 \times (3-1))$ -ой строки по K-ую строку.

40 0, 2084, 1613, 1548, 1286, 1460, 3196, 4297, 2481, 3369, 3451, 4620 и 2622 находятся в первой строке таблицы начальных значений матрицы контроля четности (фиг. 38), которая показывает, что элементы строк с номерами строк 0, 2084, 1613, 1548, 1286, 1460, 3196, 4297, 2481, 3369, 3451, 4620 и 2622 в первом столбце матрицы Н контроля четности равны 1 (другие элементы равны 0).

45 Кроме того, 1, 122, 1516, 3448, 2880, 1407, 1847, 3799, 3529, 373, 971, 4358 и 3108 находятся во второй строке таблицы начальных значений матрицы контроля четности (фиг. 38), которая показывает, что элементы строк с номерами строк 1, 122, 1516, 3448, 2880, 1407, 1847, 3799, 3529, 373, 971, 4358 и 3108 в 361 $(=1+360 \times (2-1))$ -ом столбце матрицы Н контроля четности равны 1.

Как описано выше, таблица начальных значений матрицы контроля четности

представляет собой позиции элементов 1 в информационной матрице H_A матрицы H контроля четности с интервалом 360 столбцов.

Столбцы, отличные от $(1+360 \times (i-1))$ -ых столбцов в матрице H контроля четности, другими словами, соответствующие столбцы с $(2+360 \times (i-1))$ -го столбца по $(360 \times i)$ -ый столбец размещаются путем периодического циклического сдвига элементов 1 в $(1+360 \times (i-1))$ -ом столбце, определенном с помощью таблицы начальных значений матрицы контроля четности, согласно длине M битов четности в направлении вниз (в направлении вниз столбца).

Другими словами, например, $(2+360 \times (i-1))$ -ый столбец получается путем циклического сдвига $(1+360 \times (i-1))$ -ый столбец на $M/360 (=q)$ в направлении вниз, и следующий $(3+360 \times (i-1))$ -ый столбец получается путем циклического сдвига $(2+360 \times (i-1))$ -ый столбец, который получается путем циклического сдвига $(1+360 \times (i-1))$ -ый столбец на $2 \times M/360 (=2 \times q)$ в направлении вниз, на $M/360 (=q)$ в направлении вниз.

В данном случае, если предполагается, что числовое значение j -го столбца (j -го слева) в i -ой строке (i -ой сверху) таблицы начальных значений матрицы контроля четности представлено $h_{i,j}$, и строка числом j -го элемента 1 в w -ом столбце матрицы H контроля четности представлена $H_{w,j}$, строка числом $H_{w,j}$ элемента 1 в w -ом столбце, который представляет собой столбец, отличный от $(1+360 \times (i-1))$ -го столбца матрицы H контроля четности, можно получить из уравнения (10).

$$H_{w,j} = \text{mod}\{h_{i,j} + \text{mod}((w-1), P) \times q, M\} \dots (10)$$

В данном случае, $\text{mod}(x, y)$ означает остаток при делении x на y .

Кроме того, P представляет собой число P столбцов блока циклической структуры, описанного выше, и равняется 360, например, в стандартах DVB-S.2, DVB-T.2 и DVB-C.2, как описано выше. Кроме того, q равно значению $M/360$, которое получается путем деления длина битов четности M на число $P (=360)$ столбцов блока циклической структуры.

Блок 613 выработки матрицы контроля четности (фиг. 35) точно определяет число строк с элементом 1 в $(1+360 \times (i-1))$ -ом столбце матрицы H контроля четности с помощью таблицы начальных значений матрицы контроля четности.

Кроме того, блок 613 выработки матрицы контроля четности (фиг. 35) получает номер строки $H_{w,j}$ элемента 1 в w -ом столбце, который представляет собой столбец, отличный от $(1+360 \times (i-1))$ -го столбца матрицы H контроля четности из уравнения (10), и вырабатывает матрицу H контроля четности, в которой элемент номера строки, полученный как описано выше, равен 1.

Новый LDPC-код

Между тем предполагается, что в ближайшем будущем будет увеличиваться спрос на передачу данных большого объема, таких как изображения с высоким разрешением. При передаче данных большого объема требуется LDPC-код с высокой кодовой скоростью (низкой избыточностью), но даже при использовании LDPC-кода с высокой кодовой скоростью желательно обеспечить хорошее качество связи.

Таким образом, в качестве LDPC-кода с высокой кодовой скоростью, при которой можно обеспечить хорошее качество связи, будет описан, например, LDPC-код (в дальнейшем упоминается как новый LDPC-код), чья кодовая скорость r равна $12/15$, и длина N кода равна 16 кбит.

В дополнение к этому, что касается нового LDPC-кода, матрица H_T четности матрицы H контроля четности имеет ступенчатую структуру (фиг. 11) аналогично LDPC-коду, который определен в стандарте DVB-S.2 и т.п., с точки зрения поддержания по

возможности максимальной аффинности (совместимости) с LDPC-кодом существующего стандарта, такого как DVB-S.2.

Кроме того, по отношению к новому LDPC-коду информационная матрица H_A матрицы H контроля четности имеет циклическую структуру, и число P столбцов блока циклической структуры равно 360 аналогично LDPC-коду, который определен в стандарте DVB-S.2 и т.п.

На фиг. 39 показана схема, иллюстрирующая пример таблицы начальных значений матрицы контроля четности нового LDPC-кода.

LDPC-кодер 115 (фиг. 8, фиг. 35) выполняет LDPC-кодирование с новым LDPC-кодом, используя матрицу H контроля четности, полученную из таблицы начальных значений матрицы контроля четности, показанной на фиг. 39.

В этом случае, таблица начальных значений матрицы контроля четности, показанной на фиг. 39 хранится в блоке 602 хранения LDPC-кодера 115 (фиг. 8).

Новый LDPC-код, полученный с использованием матрицы H контроля четности, полученной из таблицы начальных значений матрицы контроля четности, показанной на фиг. 39, представляет собой LDPC-код высокой производительности.

В данном случае, LDPC-код высокой производительности представляет собой LDPC-код, полученный из подходящей матрицы H контроля четности.

Кроме того, подходящей матрицей H контроля четности является матрица H контроля четности, удовлетворяющая предварительно определенному условию, в котором когда LDPC-код, полученный из матрицы H контроля четности, передается с низким отношением E_s/N_0 или E_b/N_0 (отношение мощности сигнала к мощности шума из расчета на один бит), BER (и FER) уменьшается.

Подходящую матрицу H контроля четности можно получить путем выполнения моделирования измерения BER, когда, например, LDPC-код, полученный из различных матриц контроля четности, удовлетворяющих предварительно определенным условиям, передается с низким отношением E_s/N_0 .

Предварительно определенное условие, которому будет удовлетворять подходящая матрица H контроля четности, состоит в том, что результат анализа, полученный с помощью способа анализа производительности кода, который называется эволюцией плотности, является хорошим, и отсутствует контур элементов 1, который называется циклом 4.

В данном случае, в информационной матрице H_A , если элемент 1 расположен с большой плотностью в цикле 4, известно, что производительность декодирования LDPC-кода ухудшается, поэтому требуется, чтобы отсутствовал цикл 4 в качестве предварительно определенного условия, которому будет удовлетворять подходящая матрица H контроля четности.

В дополнение к этому, можно соответствующим образом определить предварительно определенное условие, которое будет удовлетворено с помощью подходящей матрицы H контроля четности, с точки зрения повышения производительности декодирования LDPC-кода и облегчения (упрощения) процесс декодирования LDPC-кода.

На фиг. 40 и фиг. 41 показаны схемы, иллюстрирующие эволюцию плотности, с помощью которой результат анализа получается в виде предварительно определенного условия, которому будет удовлетворять подходящая матрица H контроля четности.

Эволюция плотности представляет собой способ анализа кода, предназначенный для вычисления ожидаемого значения вероятности ошибки для всего LDPC-кода (ансамбля), в котором длина N кода, характеризующаяся степенной

последовательностью, которая будет описана позже, представляет собой ∞ .

Например, ожидаемое значение вероятности ошибки определенного ансамбля равно 0 сначала на канале AWGN, но если дисперсия шума постепенно увеличивается от 0, и дисперсия шума приближается к определенному порогу или выходит за его пределы, ожидаемое значение не равно 0.

Согласно эволюция плотности можно определить качество и производительность ансамбля (адекватность матрицы контроля четности) путем сравнения порога дисперсии шума, при котором ожидаемое значение вероятности ошибки не равно 0 (которое в дальнейшем также упоминается как порог производительности).

В дополнение к этому, если ансамбль, которому принадлежит специфический LDPC-код, определен для LDPC-кода, и эволюция плотности выполняется для ансамбля, можно предсказать приблизительную производительность LDPC-кода.

Поэтому, если найден высокопроизводительный ансамбль, то высокопроизводительный LDPC-код можно найти среди LDPC-кодов, принадлежащих ансамблю.

В данном случае, описанная выше степенная последовательность представляет собой процент, переменный узел и проверочный узел которого присутствуют со своим весом для кодовой длины N LDPC-кода.

Например, регулярный (3, 6) LDPC-код, кодовая скорость которого равна $1/2$, принадлежит ансамблю, который характеризуется степенной последовательностью, в которой вес (вес столбца) всех переменных узлов равен 3, и вес (вес строки) всех проверочных узлов равен 6.

Фиг. 40 иллюстрирует график Таннера такого ансамбля.

На графике Таннера, показанном на фиг. 40, число переменных узлов, обозначенных знаками в виде круга (знак $\circ$), равно N , которое равно длине N кода, и число проверочных узлов, обозначенных знаками в виде квадрата (знак $\square$), равно $N/2$, которое получается путем умножения длины N кода на кодовую скорость $1/2$.

Три ребра, имеющие одинаковый вес столбца, соединены с каждым переменным узлом, поэтому, в общем, только $3N$ ребра соединены с N переменными узлами.

Шесть ребер, имеющих одинаковый вес столбца, соединены с каждым проверочным узлом, поэтому, в общем, только $3N$ ребер соединены с $N/2$ проверочными узлами.

Кроме того, на графике Таннера, показанном на фиг. 40, представлен один перемежитель.

Перемежитель случайным образом перегруппировывает $3N$ ребер, соединенных с N переменными узлами, и соединяет каждое перегруппированное ребро с любым из $3N$ ребер, соединенных с $N/2$ проверочными узлами.

В перемежителе картина перегруппировки для перегруппировки $3N$ ребер, соединенных с N переменными узлами, представлена в виде $(3N)! = (3N) \times (3N-1) \times \dots \times 1$. Поэтому, ансамбль, который характеризуется степенной последовательностью, в которой вес всех переменных узлов равен 3, и вес всех проверочных узлов равен 6, представляет собой агрегацию $(3N)!$ LDPC-кодов.

При моделировании получения высокопроизводительного LDPC-кода (подходящей матрицы контроля четности), в эволюции плотности используется ансамбль многореберного типа.

В многореберном типе перемежитель, через который проходят ребро, соединенное с переменным узлом, и ребро, соединенное с проверочным узлом, разделен на множество ребер таким образом, что характеристика ансамбля выполняется более строгим образом.

Фиг. 41 иллюстрирует пример графика Таннера ансамбля многореберного типа.

На графике Таннера, показанном на фиг. 41, существует два перемежителя: первый перемежитель и второй перемежитель.

Кроме того, на графике Таннера, показанном на фиг. 41, существует только v_1 переменных узлов, в которых число ребер, соединенных с первым перемежителем, равно 1, и число ребер, соединенных со вторым перемежителем, равно 0, существует только v_2 переменных узлов, в которых число ребер, соединенных с первым перемежителем, равно 1, и число ребер, соединенных со вторым перемежителем, равно 2, и существует только v_3 переменных узлов, в которых число ребер, соединенных с первым перемежителем, равно 0, и число ребер, соединенных со вторым перемежителем, равно 2.

Кроме того, на графике Таннера, показанном на фиг. 41, существует только s_1 проверочных узлов, в которых число ребер, соединенных с первым перемежителем, равно 2, и число ребер, соединенных со вторым перемежителем, равно 0, существует только s_2 проверочных узлов, в которых число ребер, соединенных с первым перемежителем, равно 2, и число ребер, соединенных со вторым перемежителем, равно 2, и существует только s_3 проверочных узлов, в которых число ребер, соединенных с первым перемежителем, равно 0, и число ребер, соединенных со вторым перемежителем, равно 3.

В данном случае, эволюция плотности и ее осуществление описаны, например, в работе "On the Design of Low-Density Parity-Check Codes within 0.0045 dB of the Shannon Limit", S.Y. Chung, G.D. Forney, T.J. Richardson, R. Urbanke, IEEE Communications Letters, VOL. 5, NO. 2, Feb 2001.

При моделировании получения нового LDPC-кода (его таблицы начальных значений матрицы контроля четности) находится ансамбль, в котором порог производительности имеет предварительно определенное значение или меньшее значение за счет использования эволюции плотности многореберного типа, и LDPC-код, BER которого уменьшается в случае использования одной или более квадратурных модуляций, таких как QPSK, среди LDPC-кодов, принадлежащих ансамблю, выбирается в качестве высокопроизводительного LDPC-кода, в данном случае, порог производительности равен E_b/N_0 (отношение мощности сигнала к мощности шума из расчета на один бит), в котором BER начинает падать (становится меньше).

Таблица начальных значений матрицы контроля четности нового LDPC-кода, описанная выше, представляет собой таблицу начальных значений матрицы контроля четности LDPC-кода, в которой длина N кода равна 16 кбит, и кодовая скорость r равна 12/15.

Поэтому, согласно новому LDPC-коду, полученному из таблицы начальных значений матрицы контроля четности, можно обеспечить хорошее качество связи при передаче данных.

На фиг. 42 показана схема, иллюстрирующая минимальную длину цикла и порог производительности матрицы H контроля четности, которые получаются из таблицы начальных значений матрицы контроля четности нового LDPC-кода на фиг. 39.

В данном случае, минимальная длина цикла (размер) означает минимальное значение длины (длина контура) контура, сконфигурированного с помощью элементов 1.

Для матрицы H контроля четности, полученной из таблицы начальных значений матрицы контроля четности нового LDPC-кода, минимальная длина цикла равна 6, и отсутствует цикл 4 (контур, сконфигурированный элементом 1, имеющей длину контура).

Кроме того, порог производительности нового LDPC-кода составляет 4,269922.

На фиг. 43 показана схема, изображающая матрицу H контроля четности (матрицу

Н контроля четности нового LDPC-кода) (которая получается из таблицы начальных значений матрицы контроля четности), показанную на фиг. 39.

В матрице Н контроля четности нового LDPC-кода вес X столбца задается в первом столбце - KX столбце, и вес столбца Y1 задается в следующих KY1 столбцах, вес столбца Y2 задается в следующих KY2 столбцах, вес столбца 2 задается в следующих M-1 столбцах, и вес столбца 1 задается в последнем одном столбце.

В данном случае, сумма $KX + KY1 + KY2 + M - 1 + 1$ равна длине N кода = 16200 битов.

На фиг. 44 показана схема, иллюстрирующая номера столбцов KX, KY1, KY2 и M (фиг. 43) и веса X, Y1 и Y2 столбцов для матрицы Н контроля четности нового LDPC-кода.

В матрице Н контроля четности нового LDPC-кода, аналогичной матрице контроля четности, показанной на фиг. 12 и фиг. 13, чем ближе к первой стороне (левой стороне) находится столбец, тем больше, вероятно, будет вес столбца, поэтому кодовый бит на стороне начала нового LDPC-кода, вероятно, будет сильным к возникновению ошибки (будет иметь устойчивость к ошибке).

На фиг. 45 показана схема, иллюстрирующая результат моделирования BER/FER нового LDPC-кода (фиг. 39).

При моделировании предполагается, что когда каналом (трактом) связи является AWGN, QPSK принят в качестве модуляции, и число 50 принято в качестве числа раз итерационного декодирования.

На фиг. 45 горизонтальная ось представляет собой E_s/N_0 , и вертикальная ось представляет собой BER/FER. В дополнение к этому, сплошная линия представляет собой BER, и пунктирная линия представляет собой FER.

Согласно фиг. 45 для нового LDPC-кода получается хорошее BER/FER, поэтому можно проверить, что хорошее качество связи обеспечивается при передаче новых данных с использованием LDPC-кода.

Пример конфигурации приемного устройства 12

На фиг. 46 показана блок-схема, иллюстрирующая пример конфигурации приемного устройства 12 (фиг. 7).

Блок 151 обработки OFDM-сигнала (операция OFDM) принимает OFDM-сигнал из передающего устройства 11 (фиг. 7), и выполняет процесс обработки сигнала в отношении OFDM-сигнала. Данные с помощью блока 151 обработки OFDM-сигнала, выполняющего обработку сигнала, подаются в блок 152 управления кадром.

Блок 152 управления кадром выполняет обработку кадра (анализ кадра), сконфигурированного с помощью данных, поданных из блока 151 обработки OFDM-сигнала, и подает результирующий сигнал целевых данных и сигнал управляющих данных в частотные демультиплексоры 161 и 153, соответственно.

Частотный демультиплексор 153 выполняет демультиплексирование по частоте в символьных блоках над данными, поступающими из блока 152 управления кадром, и подает результирующие данные в блок 154 обратного отображения.

Блок 154 обратного отображения выполняет обратное отображение (декодирование размещения сигнальных точек) и квадратурную демодуляцию данных (данных применительно к созвездию), поступающих из частотного демультиплексора 153, на основании размещения (созвездия) сигнальных точек, которые определяются в ходе квадратурной модуляции, которая будет выполняться на стороне передающего устройства 11, и подает результирующие данные (LDPC-код (правдоподобие)) в LDPC-декодер 155.

LDPC-декодер 155 выполняет LDPC-декодирование над LDPC-кодом, поступающим

из устройства обратного отображения 154, и подает результирующие целевые LDPC-данные (в данном случае, BCH-код) в BCH-декодер (BCH-декодер) 156.

BCH-декодер 156 выполняет BCH декодирование целевых LDPC-данных из LDPC-декодера 155, и выводит результирующие управляющие данные (сигнализацию).

5 Между тем, частотный деперемежитель 161 выполняет деперемежение по частоте в символьных блоках над данными, поступающими из блока 152 управления кадром, и подает результирующие данные в декодер 162 SISO/MISO.

Декодер 162 SISO/MISO выполняет пространственно-временное декодирование данных, поступающих из частотного деперемежителя 161, и подает результирующие
10 данные во временной деперемежитель 163.

Временной деперемежитель 163 выполняет временное деперемежение в символьных блоках над данными, поступающими из декодера 162 SISO/MISO, и подает результирующие данные в блок 164 обратного отображения.

Блок 164 обратного отображения выполняет обратное отображение (декодирование
15 размещения сигнальных точек) и квадратурную демодуляцию данных (данные в созвездии), поступающих из временного деперемежителя 163, на основании размещения (созвездия) сигнальных квадратурной модуляции, которая будет выполняться на стороне передающего устройства 11, и подает результирующие данные в деперемежитель 165 битов.

20 Деперемежитель 165 битов выполняет деперемежение битов данных, поступающих из блока 164 обратного отображения, обеспечивает LDPC-код (правдоподобие), который представляет собой данные, которые подвергаются деперемежению битов, и подает результирующие данные в LDPC-декодер 166.

LDPC-декодер 166 выполняет LDPC-декодирование над LDPC-кодом, поступающим
25 из деперемежителя 165 битов, и подает результирующие целевые LDPC-данные (в данном случае, BCH-код) в BCH-декодер 167.

BCH-декодер 167 выполняет BCH-декодирование целевых LDPC-данных, поступающих из LDPC-декодера 155, и выводит результирующие данные в ВВ-дескремблер 168.

30 ВВ-дескремблер 168 выполняет ВВ-дескремблирование над данными, поступающими из BCH-декодера 167, и подает результирующие данные в блок 169 удаления нулей.

Блок 169 удаления нулей удаляет нуль, вставленный блоком 112 набивки (фиг. 8), из данных, поступающих из ВВ-дескремблера 168, и подает результат в демультиплексор 170.

35 Демультиплексор 170 разделяет один или более потоков (данные объекта), полученных путем мультиплексирования данных, поступающих из блока 169 удаления нулей, в каждой потоке, выполняет необходимые процессы над ним и выводит результат в виде выходного потока.

В дополнение к этому, приемное устройство 12 можно сконфигурировать без
40 выполнения некоторых блоков, которые показаны на фиг. 46 как выполненные. Другими словами, например, если передающее устройство 11 (фиг. 8) сконфигурировано без временного перемежителя 118, кодера 119 SISO/MISO, частотного перемежителя 120 и частотного перемежителя 124, которые выполнены, приемное устройство 12 можно сконфигурировать без выполнения временного деперемежителя 163, декодера 162 SISO/
45 MISO, частотного деперемежителя 161 и частотного деперемежителя 153 соответственно, которые соответствуют временному перемежителю 118, кодеру 119 SISO/MISO, частотному перемежителю 120 и частотному перемежителю 124 передающего устройства 11.

На фиг. 47 показана блок-схема, иллюстрирующая пример конфигурации перемежителя 116 битов (фиг. 46).

Деперемежитель 165 битов выполнен с мультиплексором (MUX) 54 и деперемежителем 55 со скручиванием столбцов и выполняет деперемежение (битов) символьного бита символа, который представляет собой данные, поступающие из блока 164 обратного отображения (фиг. 46).

Другими словами, мультиплексор 54 выполняет обратный процесс замены (процесс, противоположный процессу замены), соответствующий процессу замены, выполняемой демultipлексором 25 (фиг. 9), символьного бита символа, поступающего из блока 164 обратного отображения, другими словами, обратный процесс замены, предназначенный для возврата позиции кодового бита (правдоподобия) LDPC-кода, который заменяется в процессе замены на свою исходную позицию, и подает результирующий LDPC-код в деперемежитель 55 со скручиванием столбцов.

Деперемежитель 55 со скручиванием столбцов выполняет деперемежение со скручиванием столбцов (процесс, противоположный перемежению со скручиванием столбцов), соответствующий перемежению со скручиванием столбцов в качестве процесса перегруппировки, выполняемого перемежителем 24 со скручиванием столбцов (фиг. 9) над LDPC-кодом, поступающим из мультиплексора 54, другими словами, например, деперемежение со скручиванием столбцов в качестве обратного процесса замены, предназначенного для возврата кодового бита LDPC-кода, последовательность которого изменяется путем перемежения со скручиванием столбцов в ходе процесса замены на свою первоначальную последовательность.

В частности, деперемежитель 55 со скручиванием столбцов выполняет деперемежение со скручиванием столбцов путем записи и считывания кодового бита LDPC-кода в память деперемежения, сконфигурированный аналогично памяти 31, показанной на фиг. 28.

В данном случае, в деперемежителе 55 со скручиванием столбцов запись кодового бита выполняется в направлении строки памяти деперемежения с использованием адреса считывания во время считывания кодового бита из памяти 31 в качестве адреса записи. Кроме того, считывание кодового бита выполняется с использованием адреса записи во время записи кодового бита в память 31 в качестве адреса считывания.

LDPC-код, полученный из деперемежителя со скручиванием столбцов, подается из деперемежителя 55 со скручиванием столбцов в LDPC-декодер 166.

В данном случае, когда перемежение битов четности, перемежения со скручиванием столбцов и процесс замены выполняются в отношении LDPC-кода, подаваемого из блока 164 обратного отображения в деперемежитель 165 битов, в деперемежителе 165 битов можно выполнить деперемежение битов четности, соответствующее перемежению битов четности (процессу, противоположному перемежению четности, другими словами, деперемежению битов четности для возврата кодового бита LDPC-кода, последовательность которого изменяется путем перемежения битов четности на свою первоначальную последовательность), обратный процесс замены, соответствующий процессу замены, и деперемежение со скручиванием столбцов, соответствующее перемежению со скручиванием столбцов.

Однако в деперемежителе 165 битов, показанном на фиг. 47, предусмотрены мультиплексор 54, выполняющий обратный процесс замены, соответствующий процессу замены, и деперемежитель 55 со скручиванием столбцов, выполняющий деперемежение со скручиванием столбцов, соответствующее перемежению со скручиванием столбцов, но не предусмотрен блок, выполняющий деперемежение битов четности,

соответствующее перемежению битов четности, и деперемежение битов четности не выполняется.

Поэтому, деперемежитель 165 битов (его деперемежитель 55 со скручиванием столбцов) выполняет обратный процесс замены и деперемежение со скручиванием столбцов применительно к LDPC-декодеру 166 и подает LDPC-код, который не подвергался деперемежению битов четности, в LDPC-декодер 166.

LDPC-декодер 166 выполняет LDPC-декодирование над LDPC-кодом, поступающим из деперемежителя 165 битов, с использованием матрицы контроля четности преобразования, полученной путем выполнения по меньшей мере замена столбцов, соответствующей перемежению битов четности применительно к матрице H контроля четности, используемой для LDPC-кодирования с помощью LDPC-кодера 115 (фиг. 8), и выводит результирующие данные в качестве результата декодирования целевых LDPC-данных.

На фиг. 48 показана блок-схема последовательности операций, иллюстрирующая процесс, выполняемый устройством 164 обратного отображения, деперемежителем 165 битов и LDPC-декодером 166 (фиг. 47).

На этапе S111 блок 164 обратного отображения выполняет обратное отображение и квадратурную демодуляцию над данными (данными применительно к созвездию, отображенной сигнальную точку), поступающими из временного деперемежителя 163, и подает результат в деперемежитель 165 битов, и процесс переходит на этап S112.

Деперемежитель 165 битов выполняет деперемежение (деперемежение битов) над данными, поступающими из блока 164 обратного отображения на этапе S112, и процесс переходит на этап S113.

Другими словами, на этапе S112 мультиплексор 54 в деперемежителе 165 битов выполняет обратный процесс замены над данными (соответствующими символному биту символа), поступающими из блока 164 обратного отображения, и подает кодовый бит результирующего LDPC-кода в деперемежитель 55 со скручиванием столбцов.

Деперемежитель 55 со скручиванием столбцов выполняет деперемежение со скручиванием столбцов над LDPC-кодом, поступающим из мультиплексора 54, и подает результирующий LDPC-код (правдоподобие) в LDPC-декодер 166.

На этапе S113 LDPC-декодер 166 выполняет LDPC-декодирование над LDPC-кодом, поступающим из деперемежителя 55 со скручиванием столбцов, используя матрицу H контроля четности, которая используется для LDPC-кодирования с помощью LDPC-кодера 115 (фиг. 8), другими словами, используя матрицу H контроля четности преобразование, полученную путем выполнения по меньшей мере замены столбцов, соответствующей перемежению битов четности применительно к матрице H контроля четности, и выводит результирующие данные в качестве результата декодирования целевых LDPC-данных в BCH-декодер 167.

В дополнение к этому, даже на фиг. 47, аналогично случаю, показанному на фиг. 9, для удобства описания, мультиплексор 54, который выполняет обратный процесс замены, и деперемежитель 55 со скручиванием столбцов, который выполняет деперемежение со скручиванием столбцов, конфигурируются по отдельности, но мультиплексор 54 и деперемежитель 55 со скручиванием столбцов можно сконфигурировать как единое целое.

Кроме того, когда перемежитель 116 битов (фиг. 9) не выполняет перемежение со скручиванием столбцов, деперемежитель 55 со скручиванием столбцов не нужно предусматривать в деперемежителе 165 битов (фиг. 47).

Затем будет дополнительно описано LDPC-декодирование, которое выполняет LDPC-

декодер 166, показанный на фиг. 46.

LDPC-декодер 166, показанный на фиг. 46, выполняет LDPC-декодирование над LDPC-кодом, поступающим из деперемежителя 55 со скручиванием столбцов, в котором выполняется обратный процесс замены и деперемежение со скручиванием столбцов, и деперемежение битов четности не выполняется, как описано выше, в отношении матрицы Н контроля четности, используемой для LDPC-кодирования с помощью LDPC-кодера 115 (фиг. 8) с использованием матрицы контроля четности преобразования, полученной путем выполнения по меньшей мере замены столбцов, соответствующей перемежению битов четности.

В данном случае, ранее было предложено LDPC-декодирование, которое позволяет подавить рабочую частоту в достаточно приемлемом диапазоне частот при уменьшении размера схемы путем выполнения LDPC-декодирования с использованием матрицы контроля четности преобразования (например, смотри патент Японии №4224777).

Таким образом, сначала, со ссылкой на фиг. 49 - фиг. 52, будет описано LDPC-декодирование с использованием матрицы контроля четности преобразования, которая была предложена ранее.

Фиг. 49 иллюстрирует пример матрицы Н контроля четности LDPC-кода, чья длина N кода равна 90, и кодовая скорость равна 2/3.

В дополнение к этому, на фиг. 49 (также на фиг. 50 и фиг. 51, которые будут описаны позже) 0 представлен в виде периода (.)

В матрице Н контроля четности, показанной на фиг. 49, матрица четности имеет ступенчатую структуру.

Фиг. 50 иллюстрирует матрицу Н' контроля четности, полученную путем выполнения перестановки строк уравнения (11) и перестановки столбцов уравнения (12) в отношении матрицы Н контроля четности (фиг. 49).

Перестановка строк: $(6s+t+1)$ -ая строка $\rightarrow (5t+s+1)$ -ая строка ... (11)

Перестановка столбцов: $(6x+y+61)$ -ый столбец $\rightarrow (5y+x+61)$ -ый столбец ... (12)

В данном случае, в уравнениях (11) и (12), s, t, x и y представляют собой целые числа в соответствующих интервалах $0 \leq s \leq 5$, $0 \leq t < 6$, $0 \leq x < 5$ и $0 \leq t < 6$.

В соответствии с перестановкой строк согласно уравнению (11), перестановка выполняется таким образом, что первая, седьмая, 13-ая, 19-ая и 25-ая строки, которые имеют остатки, равные 1, при делении их на 6, заменяются на первую, вторую, третью, четвертую и пятую строки, и вторая, восьмая, 14-ая, 20-ая и 26-ая строки, которые имеют остатки, равные 2, при делении их на 6, заменяются на шестую, седьмую, восьмую, девятую и десятую строки.

Кроме того, в соответствии с перестановкой столбцов согласно уравнению (12), перестановка выполняется таким образом, что 61-ый, 67-ой, 73-ий, 79-ый и 85-ый столбцы после 61-го столбца (матрицы четности), которые имеют остатки, равные 1, при делении их на 6, заменяются на 61-ый, 62-ой, 63-ий, 64-ый и 65-ый столбцы, и 62-ой, 68-ой, 74-ый, 80-ый и 86-ой столбцы, которые имеют остатки, равные 2, при делении их на 6, заменяются на 66-ой, 67-ой, 68-ой, 69-ый и 70-ый столбцы.

Таким образом, матрица, полученная путем выполнения перестановки строк и столбцов в матрице Н контроля четности (фиг. 49), представляет собой матрицу Н' контроля четности (фиг. 50).

В данном случае, даже в том случае, если перестановки строк выполняется в отношении матрицы Н контроля четности, она не влияет на последовательность кодового бита LDPC-кода.

Кроме того, перестановка столбцов согласно уравнению (12), соответствует

перемежению битов четности, которое обеспечивает перемежение $(K+qx+y+1)$ -го кодового бита, описанного выше в позиции $(K+Py+x+1)$ -го кодового бита, при соответствующей установке информационной длины K на 60, числа P столбцов блока циклической структуры на 5, и делителя q ($=M/P$) длины M битов четности (в данном случае, 30) на 6.

Поэтому, матрица H' контроля четности, показанная на фиг. 50, представляет собой матрицу контроля четности преобразования, полученную путем по меньшей мере выполнения перестановки столбца, предназначенной для замены $(K+qx+y+1)$ -го столбца матрицы контроля четности (фиг. 49) (которая в дальнейшем, при необходимости, упоминается как исходная матрица контроля четности H) на $(K+qx+x+1)$ -ый столбец.

Если LDPC-код исходной матрицы H контроля четности (фиг. 49) заменяется аналогичным образом на уравнение (12) и умножается на матрицу H' контроля четности преобразования (фиг. 50), выводится 0 вектор. Другими словами, если вектор-строка s , полученный путем выполнения перестановки столбцов согласно уравнению (12) в отношении вектора-строки s в качестве LDPC-код (кодированное слово 1) исходной матрицы H контроля четности, выражается как s' , Hs^T становится 0 вектором из-за согласно свойству матрицы H контроля четности, и, таким образом, $H's^T$ становится определенно 0 вектором.

Из вышесказанного следует, что матрица H' контроля четности преобразования (фиг. 50) представляет собой матрицу контроля четности LDPC-кода s' , полученную путем выполнения перестановки столбцов согласно уравнению (12) в отношении LDPC-кода s исходной матрицы H контроля четности.

Поэтому можно получить тот же самый результат декодирования, как и в случае декодирования LDPC-кода s исходной матрицы H контроля четности с использованием матрицы H контроля четности путем декодирования (LDPC-декодирования) LDPC-кода s' , который был подвергнут перестановке столбцов согласно уравнению (12) в отношении LDPC-кода исходной матрицы H контроля четности с использованием матрицы H' контроля четности преобразования (фиг. 50), и путем выполнения обратной перестановки для перестановки столбцов согласно уравнению (12) применительно к результату декодирования.

Фиг. 51 иллюстрирует матрицу H' контроля четности преобразования (фиг. 50) путем разнесения в блоке матрицы размером 5×5 .

На фиг. 51 матрица H' контроля четности преобразования представлена путем объединения единичной матрицы 5×5 ($=P \times P$), матрицы, имеющей 0 для одного или более элементов 1 в единичной матрице (которая в дальнейшем, при необходимости, упоминается как квазиединичная матрица), или матрицы, полученной путем циклического сдвига единичной матрицы или квазиединичной матрицы (которая в дальнейшем, при необходимости, упоминается как сдвиговая матрица), причем сумма из двух или более матриц из единичной матрицы, квазиединичной матрицы и сдвиговой матрицы (которая в дальнейшем, при необходимости, упоминается как суммарная матрица) и нулевой матрицы размером 5×5 .

Матрицу H' контроля четности преобразования, показанную на фиг. 51, можно сконфигурировать с помощью единичной матрицы 5×5 , квазиединичной матрицы, сдвиговой матрицы, суммарной матрицы и нулевой матрицы. Таким образом, матрицы 5×5 (единичная матрица, квазиединичная матрица, сдвиговая матрица, суммарная матрица и нулевая матрица), конфигурирующие матрицу H' контроля четности преобразования, в дальнейшем, при необходимости, упоминаются как конфигурирующие матрицы.

Для декодирования LDPC-кода матрицы контроля четности, выраженной с помощью конфигурирующих матриц размером $P \times P$, можно использовать архитектуру, которая одновременно выполняет число P раз вычисления проверочных узлов и вычисления переменных узлов.

На фиг. 52 показана блок-схема, иллюстрирующая пример конфигурации устройства декодирования, которое выполняет такое декодирование.

Другими словами, фиг. 52 иллюстрирует пример конфигурации устройства декодирования, которое выполняет декодирование LDPC-кода с использованием матрицы H' контроля четности преобразования (фиг. 51), которая получена путем выполнения по меньшей мере перестановки столбцов согласно уравнению (12) в отношении исходной матрицы H контроля четности (фиг. 49).

Устройство декодирования, показанное на фиг. 52, сконфигурировано с памятью 300 для хранения данных ребер, которая сконфигурирована с шестью FIFO 300_1 - 300_6 , селектора 301, который выбирает FIFO 300_1 - 300_6 , блока 302 вычисления проверочного узла, двух схем 303 и 308 циклического сдвига, памяти 304 для хранения данных ребер, которая сконфигурирована с 18 FIFO 304_1 - 304_{18} , селектора 305, который выбирает FIFO 304_1 - 304_{18} , памяти 306 данных приема, которая хранит данные приема, блока 307 вычисления переменного узла, блока 309 вычисления слова декодирования, блока 310 перегруппировки данных приема и блока 311 перегруппировки данных декодирования.

Сначала будет описан способ хранения данных в блоках 300 и 304 памяти для хранения данных ребер.

Память 300 для хранения данных ребер сконфигурирована с помощью шести FIFO 300_1 - 300_6 , в данном случае, шесть представляет собой число, полученное путем деления числа 30 строк матрицы H' контроля четности преобразования (фиг. 51) на число 5 строк конфигурирующей матрицы (число P столбцов блока циклической структуры). FIFO 300_y ($y=1, 2, \dots, 6$) сконфигурирована с областями хранения множества чисел каскадов, и можно одновременно считывать и записывать сообщение, соответствующее пяти ребрам, в данном случае, пять представляет собой число строк и число столбцов конфигурирующей матрицы (число P столбцов в блоке циклической структуры), в область памяти каждого каскада. Кроме того, число каскадов области памяти FIFO 300_y равно 9, которое равно максимальному числу 1 (вес Хэмминга) в направлении строки матрицы контроля четности преобразования (фиг. 51).

Части данных (сообщение v_i , полученное из переменного узла), соответствующие позиции 1 в первой строке - пятой строке матрицы H' контроля четности преобразования (фиг. 51) хранятся в FIFO 300_1 в виде набивки каждой строки в горизонтальном направлении (в форме игнорирования 0). Другими словами, если j -ая строка и i -ый столбец выражены как (j, i) , части данных, соответствующих позициям 1 в единичной матрице 5×5 (1, 1)-(5, 5) матрицы H' контроля четности преобразования, хранятся в области памяти первого каскада FIFO 300_1 . Части данных, соответствующие позициям 1 сдвиговой матрицы (в сдвиговой матрице, полученной путем циклического сдвига единичной матрицы 5×5 на три в направлении вправо) (1, 21)-(5, 25) матрицы H' контроля четности преобразования, хранятся в области памяти второго каскада. Аналогичным образом, данные хранятся в областях памяти третьего-восьмого каскада во взаимодействии с матрицей H' контроля четности преобразования. Затем части данных, соответствующие позициям 1 сдвиговой матрицы (в сдвиговой матрице, полученной с помощью 1 в первой строке единичной матрицы 5×5 , замененной на 0 и циклически

сдвинутой на единицу в направлении влево) (1, 86)-(5, 90) матрицы Н' контроля четности преобразования, сохраняются в области памяти девятого каскада.

Части данных, соответствующие позициям 1 в шестой строке - десятой строке матрицы Н' контроля четности преобразования (фиг. 51), хранятся в FIFO 300₂. Другими словами,

части данных, соответствующих позициям 1 в первой сдвиговой матрице, конфигурирующей суммарную матрицу (суммарную матрицу в качестве суммы первой сдвиговой матрицы и второй сдвиговой матрицы, которые получены путем соответствующего циклического сдвига единичной матрицы 5×5 на один и два в направлении вправо) (6, 1)-(10, 5) матрицы Н' контроля четности преобразования, хранятся в области памяти первого каскада FIFO 300₂. Части данных, соответствующие позициям 1 во второй сдвиговой матрице, конфигурирующей суммарную матрицу (6, 1)-(10, 5) матрицы Н' контроля четности преобразования, хранятся в области памяти второго каскада FIFO 300₂.

Другими словами, что касается конфигурирующей матрицы, имеющей вес 2 или более, когда конфигурирующая матрица представлена в виде суммы множества матриц из единичных матриц Р×Р, имеющих вес 1, квазиединичной матрицы, имеющей 0 для одного или более элементов 1 в единичной матрице, или сдвиговой матрицы, полученной путем циклического сдвига единичной матрицы или квазиединичной матрицы, частей данных, соответствующих позициям 1 в единичной матрице, имеющей вес 1, квазиединичной матрице или сдвиговой матрице (сообщения, соответствующие ребрам, принадлежащим единичной матрице, квазиединичной матрице или сдвиговой матрице) сохраняются в том же самом адресе (в том же самом FIFO среди FIFO 300₁-300₆).

В дальнейшем, данные хранятся во взаимодействии с матрицей Н' контроля четности преобразования даже в областях памяти третьего-девятого каскадов.

Аналогично FIFO 300₃-300₆, данные хранятся в связи с матрицей Н' контроля четности преобразования.

Память 304 для хранения данных ребер сконфигурирована с 18 FIFO 304₁-304₁₈, в данном случае 18 получается путем деления числа 90 столбцов матрицы Н' контроля четности преобразования на число 5 столбцов конфигурирующей матрицы (число Р столбцов в блоке с циклической структурой). FIFO 304_х (х=1, 2, ..., 18) сконфигурирована с областями памяти с множеством чисел каскадов, и можно одновременно считать и записать сообщение, соответствующее пяти ребрам, в данном случае, пять представляет собой число строк и число столбцов конфигурирующей матрицы (число Р столбцов в блоке циклической структуры), в область памяти каждого каскада.

Части данных (сообщение u_j из проверочного узла), соответствующие позициям 1 в первом столбце - пятом столбце матрицы Н' контроля четности преобразования (фиг. 51) хранятся в FIFO 304₁ в виде набивки каждого столбца в вертикальном направлении (в виде игнорирования 0). Другими словами, части данных, соответствующие позициям 1 в единичной матрице 5×5 (1, 1)-(5, 5) матрицы Н' контроля четности преобразования, хранятся в области памяти первого каскада FIFO 304₁. Части данных, соответствующие позициям 1 в первой сдвиговой матрице, конфигурирующей суммарную матрицу (суммарную матрицу в виде суммы первой сдвиговой матрицы и второй сдвиговой матрицы, которые получают соответственно путем циклического сдвига единичной матрицы 5×5 на один и два в направлении вправо) (6, 1)-(10, 5) матрицы Н' контроля четности преобразования, хранятся в области памяти второго каскада. Части данных, соответствующих позициям 1 во второй сдвиговой матрице, конфигурирующих

суммарную матрицу (6, 1)-(10, 5) матрицы H' контроля четности преобразования, хранятся в области памяти третьего каскада.

Другими словами, что касается конфигурирующей матрицы, имеющей вес 2 или более, когда конфигурирующая матрица представлена в виде суммы из множества матриц из единичной матрицы $P \times P$, имеющей вес 1, квазиединичной матрицы, имеющей 0 для одного или более элементов 1 в единичной матрице, или сдвиговой матрицы, полученной путем циклического сдвига единичной матрицы или квазиединичной матрицы, части данных, соответствующей позициям 1 в единичной матрице, имеющей вес 1, квазиединичной матрице или сдвиговой матрице (сообщения, соответствующие ребрам, принадлежащим единичной матрице, квазиединичной матрице или сдвиговой матрице) хранятся в том же самом адресе (в том же самом FIFO среди FIFO 304₁-304₁₈).

В дальнейшем, данные хранятся в связи с матрицей H' контроля четности преобразования, даже в областях памяти четвертого и пятого каскадов. Число каскадов области памяти FIFO 304₁ равно 5, что является максимальным числом 1 (вес Хэмминга) в направлении строки в первом столбце - пятом столбце матрицы H' контроля четности преобразования.

Данные хранятся в связи с матрицей H' контроля четности преобразования даже в FIFO 304₂ и 304₃, причем длина (числом каскадов) равна 5. Аналогичным образом, данные хранятся в связи с матрицей H' контроля четности преобразования даже в FIFO 304₄-304₁₂, причем каждая длина равна 3. Аналогичным образом, данные сохраняются в связи с матрицей H' контроля четности преобразования даже в FIFO 304₁₃-304₁₈, причем каждая длина равна 2.

Далее будет описана работа устройства декодирования, показанного фиг. 52. Память 300 для хранения данных ребер сконфигурирована с шестью FIFO 300₁-300₆ и выбирает FIFO, которая сохраняет данные, поступающие из FIFO 300₁-300₆, согласно информации (данным матрицы) D312 относительно того, к какой строке матрицы H' контроля четности преобразования (фиг. 51) принадлежат пять сообщений D311, которые подаются из схемы 308 циклического сдвига в предыдущем каскаде, и сохраняет по порядку сообщения D311 в группах из пяти сообщений в выбранной FIFO. Кроме того, во время считывания данных память 300 для хранения данных ребер считывает по порядку пять сообщений D300₁ из FIFO 300₁ и подает их в селектор 301 на следующий каскад. После того, как считывание сообщений из FIFO 300₁ закончилось, память 300 для хранения данных ребер считывает по порядку сообщения из FIFO 300₂-300₆ и подает сообщения в селектор 301.

Селектор 301 выбирает пять сообщений из FIFO, из которых в настоящий момент считываются данные, среди FIFO 300₁-300₆, в ответ на выбранный сигнал D301, и подает сообщения в качестве сообщения D302 в блок 302 вычисления проверочного узла.

Блок 302 вычисления проверочного узла сконфигурирован с пятью вычислителями 302₁-302₅ проверочного узла, выполняет вычисление проверочного узла согласно уравнению (7) с использованием сообщений D302, которые подаются из селектора 301 (D302₁-D302₅) (сообщение v_i в уравнении (7)), и подает пять сообщений D303 (D303₁-D303₅) (сообщение U_j в уравнении (7)), которые получены из вычисления проверочного узла в схеме 303 циклического сдвига.

Схема 303 циклического сдвига циклически сдвигает пять сообщений D303₁-D303₅, полученных с помощью блока 302 вычисления проверочного узла, основываясь на

информации (данных матрицы) D305 относительно числа раз циклического сдвига единичной матрицы (или квазиединичной матрицы), которая представляет собой основание в матрице H' контроля четности преобразования, которое выполняется на соответствующем ребре, и подает результат в качестве сообщения D304 в память 304 для хранения данных ребер.

Память 304 для хранения данных ребер сконфигурирована с 18 FIFO 304₁-304₁₈, и выбирает FIFO, которая сохраняет данные, поступающие из FIFO 304₁-304₁₈, согласно информации D305 относительно того, к какой строке матрицы H' контроля четности преобразования принадлежат пять сообщений D304, которые подаются из схемы 303 циклического сдвига в предыдущем каскаде, и размещает и сохраняет по порядку сообщения D304 в группах из пяти сообщений в выбранной FIFO. Кроме того, во время считывания данные, память 304 для хранения данных ребер считывает по порядку пять сообщений D306₁ из FIFO 304₁ и подает их в селектор 305 в следующий каскад. После окончания считывания данных из FIFO 304₁, память 304 для хранения данных ребер считывает по порядку сообщения из FIFO 304₂-304₁₈ и подает сообщения в селектор 305.

Селектор 305 выбирает пять сообщений из FIFO, из которого в данный момент времени считываются данные, среди FIFO 304₁-304₁₈, в ответ на сигнал D307 выбора, и подает сообщения в виде сообщения D308 в блок 307 вычисления переменного узла и в блок 309 вычисления слова декодирования.

Между тем, блок 310 перегруппировки данных приема перегруппировывает LDPC-код D313, соответствующий матрице H контроля четности (фиг. 49), принятый через канал 13 связи путем выполнения перестановки столбцов согласно уравнению (12), и подает перегруппированный LDPC-код в виде данных D314 приема в память 306 данных приема. Память 306 данных приема вычисляет и сохраняет логарифмическое отношение правдоподобия (LLR) приема из данных D314 приема, которые подаются из блока 310 перегруппировки данных приема, и подает LLR приема в группах из пяти, в качестве принятого значения D309, в блок 307 вычисления переменного узла и блок 309 вычисления слова декодирования.

Блок 307 вычисления переменного узла сконфигурирован с пятью вычислителями 307₁-307₅ переменного узла, выполняет вычисление переменного узла согласно уравнению (1), используя сообщение D308 (D308₁-D308₅) (сообщение u_j из уравнения (1)), которое подается через селектор 305, и пять принятых значений D309 (принятые значения u_{0i} из уравнения (1)), которые подаются из памяти 306 данных приема, и подает сообщение D310 (D310₁-D310₅) (сообщение v_i из уравнения (1)), которое получено путем вычисления в схеме 308 циклического сдвига.

Схема 308 циклического сдвига циклически сдвигает пять сообщений D310₁-D310₅, полученные с помощью блока 307 вычисления переменного узла, основываясь на информации относительно числа раз циклического сдвига единичной матрицы (или квазиединичной матрицы), которая является основанием в матрице H' контроля четности преобразования отношении, которая выполняется на соответствующем ребре, и подает результат в виде сообщения D311 в память 300 для хранения данных ребер.

Выполняя один цикл вышеописанных операций, можно выполнить одно декодирование LDPC-кодов (вычисление переменного узла и вычисление проверочного узла). После этого устройство декодирования (фиг. 52), декодирует LDPC-код предварительно определенное число раз и подает окончательно полученный результат

декодирования в блок 309 вычисления слова декодирования и блок 311 перегруппировки данных декодирования.

Другими словами, блок 309 вычисления слова декодирования конфигурируется с помощью пяти вычислителей 309_1 - 309_5 слова декодирования, вычисляет результат декодирования (слова декодирования) на основании уравнения (5) с использованием пяти сообщений $D308$ ($D308_1$ - $D308_5$) (сообщение u_j из уравнения (5)), которое выводится с помощью селектора 305 и пяти принятых значений $D309$ (принятых значений u_{0i} из уравнения (5)) поданных из памяти 306 данных приема, в качестве окончательного каскада с многочисленным числом раз декодирования, и подает результирующие данные $D315$ декодирования в блок 311 перегруппировки данных декодирования.

Блок 311 перегруппировки данных декодирования перегруппировывает и выводит порядок в качестве окончательного результата $D316$ декодирования путем выполнения обратной перестановки для перестановки столбцов согласно уравнению (12) в отношении данных $D315$ декодирования, которые подаются из блока 309 вычисления слова декодирования.

Матрица контроля четности преобразуется в матрицу контроля четности (матрицу контроля четности преобразования), которую можно выразить с помощью комбинации единичной матрицы $P \times P$, квазиединичной матрицы, имеющей 0 для одно или более элементов 1, или сдвиговой матрицы, полученной путем циклического сдвига единичной матрицы или квазиединичной матрицы, суммарной матрицы, которая представляет собой сумму из множества матриц, таких как единичная матрица, квазиединичная матрица или сдвиговая матрица, нулевой матрицы $P \times P$, другими словами, комбинации конфигурирующих матриц путем выполнения одной или двух из перестановки строк и перестановки столбцов в отношении матрицы контроля четности (исходной матрицы контроля четности) как описано выше, таким образом, что можно принять архитектуру одновременного выполнения вычислений проверочного узла и вычислений переменного узла P числом раз, в данном случае, P представляет собой число, которое меньше числа строк и числа столбцов матрицы контроля четности, для декодирования LDPC-кода. В случае принятия архитектуры одновременного выполнения узлов, вычисления (вычисления проверочного узла и вычисления переменного узла) P числом раз, в данном случае, P представляет собой число меньше числа строк и числа столбцов матрицы контроля четности, можно подавить рабочую частоту в приемлемом диапазоне и выполнить определенное число раз итерационное декодирование по сравнению со случаем одновременного выполнения вычислений узлов, число раз которых равно числу строк и числу столбцов матрицы контроля четности.

Предполагается, что LDPC-декодер 166, конфигурирующий приемное устройство 12 (фиг. 46) выполняет LDPC-декодирование путем одновременного выполнения вычислений проверочного узла и вычислений переменного узла P число раз, аналогично, например, устройству декодирования, показанному на фиг. 52.

Другими словами, чтобы упростить объяснение, предполагается, что матрица контроля четности LDPC-кода, которая выводится кодером 115 LDPC, конфигурирующим передающее устройство 11 (фиг. 8), представляет собой, например, матрицу H контроля четности, где матрица четности имеет ступенчатую структуру, показанную (фиг. 49), и в перемежителе 23 битов четности передающего устройства 11, перемежение битов четности, которое представляет собой перемежение $(K+qx+y+1)$ -го кодового бита в позиции $(K+Py+x+1)$ -го кодового бита, выполняется путем установки длины K информационных битов на 60, числа p столбцов блока циклической

структуры на 5 и делителя q с длиной M ($=M/P$) битов четности на 6.

Поскольку перемежение битов четности, как описано выше, соответствует перестановке столбцов уравнения (12), LDPC-декодеру 166 не нужно выполнять перестановку столбцов из уравнения (12).

5 По этой причине, в приемном устройстве 12 (фиг. 46), как описано выше, LDPC-код, который не подвергается деперемежению битов четности, другими словами, LDPC-код в состоянии перестановки столбцов из уравнения (12), которое выполняется, подается в LDPC-декодер 166 из деперемежителя 55 со скручиванием столбцов, и LDPC-декодер 166 выполняет такой же процесс, как и устройство декодирования (фиг. 52), за
10 исключением не выполнения перестановки столбцов уравнения (12).

Другими словами, фиг. 53 иллюстрирует пример конфигурации LDPC-декодера 166 (фиг. 46).

Поскольку LDPC-декодер 166 (фиг. 53) сконфигурирован аналогично устройству декодирования (фиг. 52) за исключением того, что не предусмотрен блок 310
15 перегруппировки данных приема (фиг. 52), и выполняет такой же процесс, как и устройство декодирования (фиг. 52) за исключением того, что не выполняется перестановка столбцов в уравнении (12), его описание будет опущено.

Как описано выше, LDPC-декодер 166 можно сконфигурировать без выполнения блока 310 перегруппировки данных приема таким образом, чтобы можно было
20 уменьшить размер в большей степени по сравнению с размером устройства декодирования (фиг. 52).

В дополнение к этому, на фиг. 49 - фиг. 53, с целью упрощения объяснения, предполагается, что длина N кода LDPC-кода равна 90, информационная длина K равна 60, число P столбцов блока циклической структуры (число строк и число столбцов
25 конфигурирующей матрицы) равно 5, и делитель q с длиной M ($=M/P$) битов четности равен 6, но длина N кода, информационная длина K , число P столбцов блока циклической структуры и делитель q ($=M/P$) соответственно ограничены приведенными выше значениями.

Другими словами, в передающем устройстве 11 (фиг. 8) LDPC-кодер 115 выводит,
30 например, LDPC-код, в котором длина N кода равна 64800, 16200 или т.п., информационная длина K равна $N-Pq$ ($=N-M$), число P столбцов блока циклической структуры равно 360, и делитель q равен M/P , но LDPC-декодер 166 (фиг. 53) одновременно выполняет вычисления проверочных узлов и вычисления переменных узлов P число раз над LDPC-кодом, и тем самым он может стать подходящим для случая
35 выполнения LDPC-декодирования.

На фиг. 54 показана схема, описывающая процесс, выполняемый в мультиплексоре 54 деперемежителя 165 битов (фиг. 47).

Другими словами, фиг. 54 иллюстрирует пример функциональной конфигурации мультиплексора 54.

40 Мультиплексор 54 выполнен с блоком 1001 обратной замены и памятью 1002.

Мультиплексор 54 выполняет обратный процесс замены (процесс, противоположный процессу замены), соответствующий процессу замены, выполняемому демультиплексором 25 передающего устройства 11 над символьным битом символа из блока 164 обратного отображения в предыдущем каскаде, другими словами, обратный
45 процесс замены, предназначенный для возврата позиции кодового бита (символьного бита) LDPC-кода, который заменяется в процессе замены на свою исходную позицию, и подает результирующий LDPC-код в следующий каскад в деперемежитель 55 со скручиванием столбцов.

Другими словами, символьные биты $y_0, y_1, \dots, y_{mb-1}$ mb битов b символов в блоке из (последующих) b символов подаются в блок 1001 обратной замены, находящийся в мультиплексоре 54.

Блок 1001 обратной замены выполняет обратную замену, предназначенную для
 5 возврата последовательности символьных битов y_0 - y_{mb-1} mb битов в размещение первоначальных кодовых битов $b_0, b_1, \dots, b_{mb-1}$ m битов (последовательности кодовых битов b_0 - b_{mb-1} перед выполнением замены с помощью блока 32 замены, конфигурирующего демультимплексор 25 передающего устройства 11), и выводит
 10 результирующие кодовые биты b_0 - b_{mb-1} mb битов.

Память 1002 имеет объем памяти для хранения mb битов в направлении строки (по горизонтали) и $N/(mb)$ битов в направлении столбца (по вертикали) аналогично памяти 31, конфигурирующую демультимплексор 25 на стороне передающего устройства 11. Другими словами, память 1002 выполнена с mb столбцами для хранения $N/(mb)$ битов.

15 В данном случае, запись кодового бита LDPC-кода, который выводится блоком 1001 обратной замены в память 1002, выполняется в направлении считывания кодового бита из памяти 31 демультимплексора 25 передающего устройства 11, и считывание кодового бита, который записывается в память 1002, выполняется в направлении записи кодового бита в памяти 31.

20 Другими словами, как показано поз. А на фиг. 54, мультиплексор 54 приемного устройства 12 последовательно выполняет запись кодового бита LDPC-кода, который выводится с помощью блока 1001 обратной замены в направлении нижней строки из первой строки памяти 1002.

Затем, если запись кодового бита одной длины кода закончено, мультиплексор 54
 25 считывает кодовый бит в направлении столбца из памяти 1002 и подает его в следующий каскад в деперемежитель 55 со скручиванием столбцов.

В данном случае, поз. В на фиг. 54 представляет собой схему, иллюстрирующую считывание кодового бита из памяти 1002.

В мультиплексоре 54, считывание кодового бита LDPC-кода в направлении сверху
 30 вниз столбцов (в направлении столбца), конфигурирующих память 1002, выполняется в направлении столбца слева направо.

На фиг. 55 показана схема, изображающая процесс, выполняемый в деперемежителе 55 со скручиванием столбцов, конфигурирующем деперемежитель 165 битов (фиг. 47).

Другими словами, фиг. 55 иллюстрирует пример конфигурации памяти 1002
 35 мультиплексора 54.

Память 1002 имеет объем памяти для хранения mb битов в направлении столбца (по вертикали) и $N/(mb)$ битов в направлении строки (по горизонтали) и выполнена с mb столбцами.

Деперемежитель 55 со скручиванием столбцов выполняет деперемежение со
 40 скручиванием столбцов путем управления позицией начала записи при записи кодового бита LDPC-кода в направлении строки в память 1002 и при считывании кодового бита в направлении столбца из памяти 1002.

Другими словами, деперемежитель 55 со скручиванием столбцов выполняет обратный процесс перегруппировки для возврата последовательности кодового бита, который
 45 переупорядочивается путем перемежения со скручиванием столбцов, в исходную последовательность путем подходящего изменения позиции начала считывания, в которой начинается считывание кодового бита, для каждого из множества столбцов.

В данном случае, фиг. 55 иллюстрирует пример конфигурации памяти 1002 в случае,

когда схемой модуляции (фиг. 28) является 16APSK, 16QAM, или т.п., и множитель b равен 1. В этом случае, число m битов одного символа равно четырем, и память 1002 выполнена с 4 ($=mb$) столбцами.

Деперемежитель 55 со скручиванием столбцов последовательно выполняет запись
 5 кодового бита LDPC-кода, который выводится с помощью блока замены 1001, а не мультиплексора 54, в направлении нижней строки из первой строки памяти 1002.

Затем, если запись кодового бита одной длины кода закончена, то деперемежитель 55 со скручиванием столбцов выполняет считывание кодового бита в направлении
 сверху вниз (в направлении столбца) памяти 1002 в направлении столбца слева направо.

10 В данном случае, деперемежитель 55 со скручиванием столбцов выполняет считывание кодового бита из памяти 1002, с использованием позиции начала записи, в которой перемежитель 24 со скручиванием столбцов передающего устройства 11 записывает кодовый бит в качестве позиции начала записи кодового бита.

Другими словами, если предполагается, что адрес первой (верхней) позиции каждого
 15 столбца равен 0, и адрес каждой позиции в направлении столбца представлен с помощью целых чисел в порядке возрастания, когда схемой модуляции является 16APSK или 16QAM, и множитель b равен 1, в деперемежителе 55 со скручиванием столбцов позиция начала записи для крайнего левого столбца представляет собой позицию адреса 0, позиция начала записи для второго столбца (слева) представляет собой позицию адреса
 20 2, позиция начала записи для третьего столбца представляет собой позицию адреса 4 и позиция начала записи для четвертого столбца представляет собой позицию адреса 7.

В дополнение к этому, что касается столбцов, чьи позиции начала записи отличаются от позиции адреса 0, то после считывания кодовых битов вплоть до нижней позиции,
 25 возвращаясь обратно в начало (в позицию адреса 0), кодовые биты считываются вплоть до позиции сразу же перед позицией начала записи. После этого выполняется считывание из следующего столбца (справа).

Путем выполнения деперемежения со скручиванием столбцов, как описано выше, последовательность кодового бита, который переупорядочивается путем перемежения
 30 со скручиванием столбцов, возвращается к исходной последовательности.

На фиг. 56 показана блок-схема, иллюстрирующая другой пример конфигурации деперемежителя 165 битов (фиг. 46).

В дополнение к этому, на чертеже части, соответствующие частям, показанным на
 35 фиг. 47, обозначены одинаковыми ссылочными позициями, и в случае необходимости их описание будет опущено.

Другими словами, деперемежитель 165 битов на фиг. 56 сконфигурирован аналогично случаю, показанному на фиг. 47, за исключением того, что снова предусмотрен деперемежитель 1011 битов четности.

На фиг. 56 деперемежитель 165 битов снабжен мультиплексором (MUX) 54,
 40 деперемежителем 55 со скручиванием столбцов и деперемежителем 1011 битов четности и выполняет деперемежение битов над кодовым битом LDPC-кода, поступающим из блока 164 обратного отображения.

Другими словами, мультиплексор 54 выполняет обратный процесс замены (процесс, противоположный процессу замены), соответствующий процессу замены, выполняемому
 45 демультимплексором 25 передающего устройства 11, другими словами, обратный процесс замены для возврата позиции кодового бита, который заменяется в процессе замены, на свою исходную позицию, над LDPC-кодом, поступающим из блока 164 обратного отображения, и подает результирующий LDPC-код в деперемежитель 55 со скручиванием

столбцов.

Деперемежитель 55 со скручиванием столбцов выполняет деперемежение со скручиванием столбцов, соответствующее перемежению со скручиванием столбцов в качестве процесса перегруппировки, выполняемого перемежителем 24 со скручиванием столбцов передающего устройства 11 над LDPC-кодом, поступающим из мультиплексора 54.

LDPC-код, полученный с помощью деперемежителя со скручиванием столбцов, подается из деперемежителя 55 со скручиванием столбцов в деперемежитель 1011 битов четности.

Деперемежитель 1011 битов четности выполняет деперемежение битов четности (процесс, противоположный перемежению четности), соответствующий перемежению битов четности, выполняемому перемежителем 23 битов четности передающего устройства И, другими словами, деперемежение битов четности, предназначенное для возврата кодового бита LDPC-кода, последовательность которого меняется путем перемежения битов четности на свою первоначальную последовательность, в отношении кодового бита, который подвергается деперемежению со скручиванием столбцов с помощью деперемежителя 55 со скручиванием столбцов.

LDPC-код, полученный с помощью деперемежения битов четности, подается из деперемежителя 1011 битов четности в LDPC-декодер 166.

Поэтому, в деперемежителе 165 битов (фиг. 56) LDPC-код, подвергнутый обратному процессу замены, деперемежению со скручиванием столбцов и деперемежению битов четности, другими словами, LDPC-код, полученный путем LDPC-кодирования в соответствии с матрицей Н контроля четности, подается в LDPC-декодер 166.

LDPC-декодер 166 выполняет LDPC-декодирование LDPC-кода, поступающего из деперемежителя 165 битов, с использованием матрицы Н контроля четности, используемой при LDPC-кодировании с помощью LDPC-кодера 115 передающего устройства 11. Другими словами, LDPC-декодер 166 выполняет LDPC-декодирование LDPC-кода, поступающего из деперемежителя 165 битов, с использованием матрицы Н контроля четности, используемой при LDPC-кодировании с помощью LDPC-кодера 115 передающего устройства 11, или с использованием матрицы контроля четности преобразования, полученной путем выполнения по меньшей мере перестановки столбцов, соответствующей перемежению битов четности в отношении матрицы Н контроля четности.

В данном случае, на фиг. 56, так как LDPC-код, полученный путем LDPC-кодирования в соответствии с матрицей Н контроля четности, подается из деперемежителя 165 битов (его деперемежителя 1011 битов четности) в LDPC-декодер 166, когда LDPC-кодер 115 передающего устройства 11 выполняет LDPC-декодирование LDPC-кода с использованием матрицы Н контроля четности, используемой при LDPC-кодировании, LDPC-декодер 166 может, например, представлять собой устройство декодирования, которое выполняет LDPC-декодирование согласно схеме полного последовательного декодирования, предназначенной для последовательного выполнения вычисления сообщения (сообщения проверочного узла и сообщения переменного узла) узел за узлом, или устройство декодирования, которое выполняет LDPC-декодирование согласно схеме полного параллельного декодирования, предназначенной для одновременного (параллельного) выполнения вычисления сообщения для всех узлов.

Кроме того, когда LDPC-декодер 166 выполняет LDPC-декодирование LDPC-кода в отношении матрицы Н контроля четности, используемой при LDPC-кодировании, с помощью LDPC-кодера 115 передающего устройства 11, используя матрицу контроля

четности преобразования, полученную по меньшей мере путем выполнения перестановки столбцов, соответствующей перемежению битов четности, LDPC-декодер 166 представляет собой устройство декодирования с архитектурой одновременного выполнения вычисления проверочных узлов и вычисления переменных узлов Р (или делителя Р, где делитель отличен от 1) число раз, и сконфигурировано устройством декодирования (фиг. 52), включающим в себя блок 310 перегруппировки данных приема, который перегруппировывает кодовый бит LDPC-кода, путем выполнения такой же перестановки столбцов, как и перестановка столбцов для получения матрицы контроля четности преобразования в отношении LDPC-кода.

В дополнение к этому, на фиг. 56, для удобства описания, мультиплексор 54, который выполняет обратный процесс замены, деперемежитель 55 со скручиванием столбцов, который выполняет деперемежение со скручиванием столбцов, и деперемежитель 1011 битов четности, который выполняет деперемежение битов четности конфигурируются, соответственно, по отдельности, но два или более из мультиплексора 54, деперемежителя 55 со скручиванием столбцов и деперемежителя 1011 битов четности можно сконфигурировать как единое целое, аналогично перемежителю 23 битов четности, перемежителю 24 со скручиванием столбцов и демultipлексору 25 передающего устройства 11.

Кроме того, когда перемежитель 116 битов (фиг. 8) передающего устройства 11 выполнен без обеспечения перемежителя 23 битов четности и перемежителя 24 со скручиванием столбцов, деперемежитель 165 битов можно сконфигурировать без обеспечения деперемежителя 55 со скручиванием столбцов и деперемежителя 1011 битов четности, как показано на фиг. 56.

Даже в этом случае, LDPC-декодер 166 можно сконфигурировать с устройством декодирования из схемы полного последовательного декодирования, которая выполняет LDPC-декодирование с использованием матрицы Н контроля четности, устройством декодирования из схемы полного параллельного декодирования, которая выполняет LDPC-декодирование с использованием матрицы Н контроля четности, или устройством декодирования (фиг. 52), включающим в себя блок 310 перегруппировки данных приема, который выполняет LDPC-декодирование путем Р одновременных вычислений проверочных узлов и вычислений переменных узлов с использованием матрицы Н' контроля четности преобразования.

Пример конфигурации приемной системы

На фиг. 57 показана блок-схема, иллюстрирующая первый пример конфигурации приемной системы, в которой применяется приемное устройство 12.

На фиг. 57 приемная система выполнена с блоком 1101 получения, блоком 1102 обработки канального декодирования и блоком 1103 обработки декодирования источника информации.

Блок 1101 получения получает сигналы, включающие в себя LDPC-код, полученный по меньшей мере путем LDPC-кодирования целевых LDPC-данных, таких как данные изображения и звуковые данные программы, через например, канал (канал связи), который не показан, такой как наземное цифровое ширококовещание, спутниковое цифровое ширококовещание, сеть CATV, Интернет и другие сети, и подает сигналы в блок 1102 обработки канального декодирования.

В данном случае, при трансляции сигналов, полученных с помощью блока 1101 получения, например, из ширококовещательной станции через наземные волны, спутниковые волны, кабельные телевизионные сети (CATV) и т.п., блок 1101 получения сконфигурирован с тюнером, телевизионной абонентской приставкой (STB) или т.п.

Кроме того, когда сигналы, полученные, например, из веб-сервера, с помощью блока 1101 получения, являются многоадресными, так как в телевидении по Интернет-протоколу (IPTV), блок 1101 получения сконфигурирован, например, с сетевым интерфейсом (IF), таким как карта сетевого интерфейса (NIC).

5 Блок 1102 обработки канального декодирования соответствует приемному устройству 12. Блок 1102 обработки канального декодирования выполняет процесс канального декодирования, включающий в себя по меньшей мере процесс исправления ошибки, возникающей в канале, в отношении сигналов, полученных с помощью блока 1101 получения через канал, и подает результирующий сигнал в блок 1103 обработки
10 декодирования источника информации.

Другими словами, сигналы, полученные с помощью блока 1101 получения через канал, представляют собой сигналы, полученные путем выполнения по меньшей мере кодирования с коррекцией ошибок для коррекции ошибок, возникающих в канале, и блок 1102 обработки канального декодирования выполняет, например, процесс
15 канального декодирования, такой как процесс коррекции ошибок в отношении таких сигналов.

В данном случае, примеры кодирования с коррекцией ошибок включают в себя LDPC-кодирование и BCH-кодирование. В данном случае, по меньшей мере LDPC-кодирование выполняется как кодирование с коррекцией ошибок.

20 Кроме того, процесс канального декодирования может включать в себя демодуляцию сигнала модуляции и т.п.

Блок 1103 обработки декодирования источника информации выполняет процесс декодирования источника информации, включающий в себя по меньшей мере процесс распаковки сжатой информации в исходную информации в отношении сигнала,
25 подвергнутому процессу канального декодирования.

Другими словами, сигналы, полученные с помощью блока 1101 получения через канал, можно подвергнуть кодированию со сжатием для сжатия информации, чтобы уменьшить надлежащим образом количество данных, такой как изображение и звук в качестве информации, в этом случае, блок 1103 обработки декодирования источника
30 информации выполняет процесс декодирования источника информации, такой как процесс распаковки (процесс, противоположный процессу сжатия) сжатой информации в исходную информации в отношении сигнала, подвергнутого процессу канального декодирования.

В дополнение к этому, когда сигналы, полученные с помощью блока 1101 получения
35 по каналу не подвергаются кодированию со сжатием, блок 1103 обработки декодирования источника информации не выполняет процесс распаковки сжатой информации в исходную информации.

В данном случае, пример процесса распаковки включает в себя MPEG-декодирование. Кроме того, процесс канального декодирования может включать в себя
40 дескремблирование и т.п., в дополнение к процессу распаковки.

В приемной системе, сконфигурированный как описано выше, блок 1101 получения получает, например, сигналы, полученные путем выполнения кодирования с сжатием, такого как MPEG-кодирование и кодирование с коррекцией ошибок, такое как LDPC-кодирование над данными, такими как изображение и звук, через канал, и подает их в
45 блок 1102 обработки канального декодирования.

В блоке 1102 обработки канального декодирования, например, такой же процесс, как и процесс, выполняемый приемным устройством 12, выполняется в качестве процесса канального декодирования в отношении сигнала, поступающего из блока 1101

получения, и результирующий сигнал подается в блок 1103 обработки декодирования источника информации.

В блоке 1103 обработки декодирования источника информации процесс декодирования источника информации, такой как MPEG-декодирование, выполняется в отношении сигнала, поступающего из блока 1102 обработки канального декодирования, и выводится результирующее изображение или звук.

Приемную систему, показанную на фиг. 57 и описанную выше, можно применить, например, в телевизионном тюнере, который принимает телевизионное вещание в качестве цифрового широко вещания.

В дополнение к этому, блок 1101 получения, блок 1102 обработки канального декодирования и блок 1103 обработки декодирования источника информации можно, соответственно, сконфигурировать в виде одного независимого устройства (аппаратного средства (интегральной схемы (ИС) или т.п.) или модуля программного обеспечения).

Кроме того, что касается блока 1101 получения, блока 1102 обработки канального декодирования и блока 1103 обработки декодирования источника информации, то набор из блока 1101 получения и блока 1102 обработки канального декодирования, набор из блока 1102 обработки канального декодирования и блока 1103 обработки декодирования источника информации и набор из блока 1101 получения, блока 1102 обработки канального декодирования и блока 1103 обработки декодирования источника информации, соответственно, сконфигурированы в виде одного независимого устройства.

На фиг. 58 показана блок-схема, иллюстрирующая второй пример конфигурации приемной системы, в которой применяется приемное устройство 12.

В дополнение к этому, части, показанные на чертеже и соответствующие частям, показанным на фиг. 57, обозначены одинаковыми ссылочными позициями, и поэтому в случае необходимости их описание будет опущено ниже.

Приемная система, показанная на фиг. 58, представлена, в общем, на фиг. 57 при наличии блока 1101 получения, блока 1102 обработки канального декодирования и блока 1103 обработки декодирования источника информации и отличается от случая, показанного на фиг. 57, в котором вновь выполнен блок 1111 вывода.

Например, блок 1111 вывода представляет собой устройство отображения, которое отображает изображение, или громкоговоритель, который выводит звук, и выводит изображение и звук или т.п. в качестве сигнал, выводимого из блока 1103 обработки декодирования источника информации. Другими словами, блок 1111 вывода отображает изображение или выводит звук.

Приемная система, показанная на фиг. 58 и описанная выше, может быть применена, например, в телевизионном приемнике (ТВ), принимающем телевизионное вещание в качестве цифрового вещания, радиоприемнике, принимающем цифровое широко вещание или т.п.

В дополнение к этому, когда кодирование сжатием не применяется к сигналу, полученному с помощью блока 1101 получения, сигнал, выводимый блоком 1102 обработки канального декодирования, подается в блок 1111 вывода.

На фиг. 59 показана блок-схема, иллюстрирующая третий пример конфигурации приемной системы, в которой может применяться приемное устройство 12.

В дополнение к этому, части, показанные на чертеже и соответствующие частям, показанным на фиг. 57, обозначены одинаковыми ссылочными позициями, и тем самым их описание при необходимости будет опущено ниже.

Приемная система (фиг. 59) представлена, в общем, на фиг. 57 при наличии блока

1101 получения и блока 1102 обработки канального декодирования.

Однако приемная система (фиг. 59) отличается от случая, показанного на фиг. 57, тем, что блок 1103 обработки декодирования источника информации не предусмотрен, и заново выполнен блок записи 1121.

5 Блок записи 1121 записывает (сохраняет) сигналы (например, TS пакеты TS MPEG), которые выводятся блоком 1102 обработки канального декодирования на носитель (информации) записи, такой как оптический диск, жесткий диск (магнитный диск) и флэш-память.

10 Приемная система, показанная на фиг. 59 и описанная выше, может применяться в видеомаягнитофоне, записывающем телевизионное вещание.

В дополнение к этому, как показано на фиг. 59, приемная система сконфигурирована путем обеспечения блока 1103 обработки декодирования источника информации и может записывать сигнал, подвергнутый процессу декодирования источника информации, выполняемому блоком 1103 обработки декодирования источника информации, другими словами, изображение и звук получают в результате декодирования в блоке 1121 записи.

Вариант осуществления компьютера

Затем ряд процессов, описанных выше, можно выполнить с помощью аппаратных средств или программного обеспечения. Когда ряд процессов выполняется с помощью 20 программного обеспечения, программа, образующая программное обеспечение, устанавливается на компьютер общего назначения или т.п.

Таким образом, фиг. 60 иллюстрирует пример конфигурации варианта осуществления компьютера, в котором установлена программа, исполняющая ряд процессов, описанных выше устанавливается.

25 Программу можно записать заранее на жесткий диск 705 или в ПЗУ 703 в качестве носителя записи, встроенного в компьютер.

Альтернативно, программу можно временно или постоянно сохранить (записать) на съемный носитель 711 информации, такой как гибкий диск, постоянное запоминающее устройство на основе компакт-диска (CD-ROM), магнитооптический (MO) диск, 30 цифровой универсальный диск (DVD), магнитный диск и полупроводниковая память. Такой съемный носитель 711 информации можно выполнить в виде называемого пакета программного обеспечения.

Дополнительно, программу помимо установки на компьютер со съемного носителя 711 информации, описанного выше, можно передать беспроводным способом в 35 компьютер из сайта загрузки через искусственный спутник для цифрового спутникового широковещания или передать в компьютер проводным способом через сеть, такую как локальная вычислительная сеть (LAN) или Интернет, и компьютер может принять программу, переданную как описано выше, с помощью блока 708 связи, и установить программу на встроенный жесткий диск 705.

40 Компьютер имеет встроенное центральное процессорное устройство (ЦПУ) 702. Интерфейс 710 ввода и вывода подсоединен к ЦПУ 702 через шину 701, и когда инструкция вводится с помощью блока 707 ввода, такого как клавиатура, мышь, микрофон или т.п., с которыми работает пользователь, через интерфейс 710 ввода и вывода, ЦПУ 702 исполняет программу, которая хранится в постоянном запоминающем 45 устройстве (ПЗУ) 703, в ответ на инструкцию. Альтернативно, ЦПУ 702 загружает программу, которая хранится на жестком диске 705, программу, которая передается со спутника или из сети, принимается с помощью блока 708 связи и устанавливается на жесткий диск 705, или программу, которая считывается со съемного носителя 711

информации, установленного в дисководе 709, и установлена на жестком диске 705, в оперативное запоминающее устройство (ОЗУ) 704, и исполняет программы. Таким образом, ЦПУ 702 исполняет процесс согласно блок-схеме последовательности операций, описанной выше, или процесс, выполняемый в соответствии с конфигурацией блок-схемы, описанный выше. Затем ЦПУ 702 побуждает выводить результат процесса так, как это требуется, например, через интерфейс 710 ввода и вывода, из блока 706 вывода, выполненного с жидкокристаллическим дисплеем (ЖКД), громкоговорителем и т.п., или передавать его из блока 708 связи или записывать на жесткий диск 705.

В данном документе и в данном описании этапы обработки, описывающие программу, побуждающую компьютер исполнять различные процессы, не обязательно будут обрабатываться во временной последовательности в соответствии с порядком, описанным в блок-схеме последовательности операций, и эти этапы обработки включают в себя процессы, которые должны обрабатываться параллельно или по отдельности (например, параллельная обработка или обработка с использованием объектов).

Кроме того, программа может быть обработана одним компьютером или может быть распределена и обработана множеством компьютеров. Кроме этого, программа может быть передана и исполнена в удаленном компьютере.

Кроме того, в описании, система означает набор из множества компонентов (устройств, модулей (изделий) и т.п.), и не имеет значения то, находятся ли все компоненты в одном корпусе. Поэтому, как множество устройств, которые размещены в отдельных корпусах и соединены через сеть, так и одно устройство, в котором множество модулей размещены в одном корпусе, являются системами.

В дополнение к этому, варианты осуществления данной технологии не ограничиваются вышеописанными вариантами осуществления, и различные модификации можно выполнить без отклонения от объема данной технологии.

Например, данная технология может принимать конфигурацию облачных вычислений, в которой одна функция используется совместно и обрабатывается совместно множеством устройств через сеть.

Кроме того, каждый этап, описанный в блок-схеме последовательности операций, представленной выше, выполняется с помощью одного устройства, и его можно выполнить путем совместного использования многочисленных устройств.

Кроме того, если множество процессов включены в один этап, то многочисленные процессы, включенные в один этап, выполняются с помощью одного устройства, и их можно выполнить путем совместного использования многочисленных устройств.

Кроме того, например, для нового LDPC-кода, описанного выше (его таблица начальных значений матрицы контроля четности), можно использовать спутниковые линии, наземные волны, кабели (проводные линии) и прочее в качестве канала 13 связи (фиг. 7). Кроме того, новый LDPC-код можно использовать для передачи данных, отличных от цифрового широко вещания.

Перечень ссылочных позиций

11 ПЕРЕДАЮЩЕЕ УСТРОЙСТВО

12 ПРИЕМНОЕ УСТРОЙСТВО

23 ПЕРЕМЕЖИТЕЛЬ ЧЕТНОСТИ

24 ПЕРЕМЕЖИТЕЛЬ СО СКРУЧИВАНИЕМ СТОЛБЦОВ

25 ДЕМУЛЬТИПЛЕКСОР

31 ПАМЯТЬ

32 БЛОК ЗАМЕНЫ

54 МУЛЬТИПЛЕКСОР

	55 ПЕРЕМЕЖИТЕЛЬ СО СКРУЧИВАНИЕМ СТОЛБЦОВ
	111 АДАПТАЦИЯ РЕЖИМА/МУЛЬТИПЛЕКСОР
	112 СУММАТОР
	113 ВВ-СКРЕМБЛЕР
5	114 ВСН-КОДЕР
	115 LDPC-КОДЕР
	116 ПЕРЕМЕЖИТЕЛЬ БИТОВ
	117 УСТРОЙСТВО ОТОБРАЖЕНИЯ
	118 ВРЕМЕННОЙ ПЕРЕМЕЖИТЕЛЬ
10	119 КОДЕР SISO/MISO
	120 ЧАСТОТНЫЙ ПЕРЕМЕЖИТЕЛЬ
	121 ВСН-КОДЕР
	122 LDPC-КОДЕР
	123 УСТРОЙСТВО ОТОБРАЖЕНИЯ
15	124 ЧАСТОТНЫЙ ПЕРЕМЕЖИТЕЛЬ
	131 БЛОК ВЫДЕЛЕНИЯ РЕСУРСОВ/ПОСТРОИТЕЛЬ КАДРОВ
	132 БЛОК ВЫРАБОТКИ OFDM-СИГНАЛА
	151 БЛОК ОБРАБОТКИ OFDM-СИГНАЛА
	152 БЛОК УПРАВЛЕНИЯ КАДРАМИ
20	153 ЧАСТОТНЫЙ ДЕПЕРЕМЕЖИТЕЛЬ
	154 БЛОК ОБРАТНОГО ОТОБРАЖЕНИЯ
	155 LDPC-ДЕКОДЕР
	156 ВСН-ДЕКОДЕР
	161 ЧАСТОТНЫЙ ДЕПЕРЕМЕЖИТЕЛЬ
25	162 ДЕКОДЕР SISO/MISO
	163 ВРЕМЕННОЙ ДЕПЕРЕМЕЖИТЕЛЬ
	164 БЛОК ОБРАТНОГО ОТОБРАЖЕНИЯ
	165 ДЕПЕРЕМЕЖИТЕЛЬ БИТОВ
	166 LDPC-ДЕКОДЕР
30	167 ВСН-ДЕКОДЕР
	168 ВВ-ДЕСКРЕМБЛЕР
	169 БЛОК УДАЛЕНИЯ НУЛЕЙ
	170 ДЕМУЛЬТИПЛЕКСОР
	210 БЛОК Tx
35	211 БЛОК FEC
	212 БЛОК ОТОБРАЖЕНИЯ
	213 БЛОК ПЕРЕДИСКРЕТИЗАЦИИ
	214 БЛОК ФИЛЬТРА НАЙКВИСТА
	220 БЛОК Rx
40	221 БЛОК AGC
	222 УМНОЖИТЕЛЬ
	223 БЛОК СГЛАЖИВАЮЩЕГО ФИЛЬТРА
	224 БЛОК СНИЖЕНИЯ ЧАСТОТЫ ВЫБОРКИ
	225 БЛОК CSI
45	226 БЛОК ОБРАТНОГО ОТОБРАЖЕНИЯ
	227 БЛОК FEC
	230 КАНАЛЬНЫЙ БЛОК
	231 БЛОК IBO

232 УМНОЖИТЕЛЬ

233 БЛОК TWTA

234 БЛОК AWGN

235 СУММАТОР

5 300 БЛОК ПАМЯТИ ДЛЯ ХРАНЕНИЯ ДАННЫХ РЕБЕР

301 СЕЛЕКТОР

302 БЛОК ВЫЧИСЛЕНИЯ ПРОВЕРОЧНОГО УЗЛА

303 СХЕМА ЦИКЛИЧЕСКОГО СДВИГА

304 БЛОК ПАМЯТИ ДЛЯ ХРАНЕНИЯ ДАННЫХ РЕБЕР

10 305 СЕЛЕКТОР

306 ПАМЯТЬ ДАННЫХ ПРИЕМА

307 БЛОК ВЫЧИСЛЕНИЯ ПЕРЕМЕННОГО УЗЛА

308 СХЕМА ЦИКЛИЧЕСКОГО СДВИГА

309 БЛОК ВЫЧИСЛЕНИЯ СЛОВА ДЕКОДИРОВАНИЯ

15 310 БЛОК ПЕРЕГРУППИРОВКИ ДАННЫХ ПРИЕМА

311 БЛОК ПЕРЕГРУППИРОВКИ ДАННЫХ ДЕКОДИРОВАНИЯ

601 БЛОК ОБРАБОТКИ КОДИРОВАНИЯ

602 БЛОК ХРАНЕНИЯ

611 БЛОК УСТАНОВКИ КОДОВОЙ СКОРОСТИ

20 612 БЛОК СЧИТЫВАНИЯ ТАБЛИЦЫ НАЧАЛЬНЫХ ЗНАЧЕНИЙ

613 БЛОК ВЫРАБОТКИ МАТРИЦЫ КОНТРОЛЯ ЧЕТНОСТИ

614 БЛОК СЧИТЫВАНИЯ ИНФОРМАЦИОННОГО БИТА

615 БЛОК ВЫЧИСЛЕНИЯ ЧЕТНОСТИ КОДИРОВАНИЯ

616 БЛОК УПРАВЛЕНИЯ

25 701 ШИНА

702 ЦПУ

703 ПЗУ

704 ОЗУ

705 ЖЕСТКИЙ ДИСК

30 706 БЛОК ВЫВОДА

707 БЛОК ВВОДА

708 БЛОК СВЯЗИ

709 ДИСКОВОД

710 ИНТЕРФЕЙС ВВОДА И ВЫВОДА

35 711 СЪЕМНЫЙ НОСИТЕЛЬ ИНФОРМАЦИИ

1001 БЛОК ОБРАТНОЙ ЗАМЕНЫ

1002 ПАМЯТЬ

1011 ДЕПЕРЕМЕЖИТЕЛЬ БИТОВ ЧЕТНОСТИ

1101 БЛОК ПОЛУЧЕНИЯ

40 1102 БЛОК ОБРАБОТКИ КАНАЛЬНОГО ДЕКОДИРОВАНИЯ

1103 БЛОК ОБРАБОТКИ ДЕКОДИРОВАНИЯ ИСТОЧНИКА ИНФОРМАЦИИ

1111 БЛОК ВЫВОДА

1121 БЛОК ЗАПИСИ

45 (57) Формула изобретения

1. Устройство обработки данных, содержащее:

блок кодирования, который кодирует информационный бит в LDPC-код, имеющий длину кода 16200 битов и кодовую скорость 12/15, на основании матрицы контроля

четности кода разреженного контроля четности (LDPC),

где LDPC-код включает в себя информационный бит и бит четности,

где матрица контроля четности включает в себя информационную часть матрицы, соответствующую информационному биту, и часть матрицы четности, соответствующую

5 биту четности,

где информационная часть матрицы представлена таблицей начальных значений матрицы контроля четности, и

где таблица начальных значений матрицы контроля четности представляет собой таблицу, представляющую позиции элементов 1 в информационной части матрицы с

10 интервалом 360 столбцов, и представляет собой

3 394 1014 1214 1361 1477 1534 1660 1856 2745 2987 2991 3124 3155

59 136 528 781 803 928 1293 1489 1944 2041 2200 2613 2690 2847

155 245 311 621 1114 1269 1281 1783 1995 2047 2672 2803 2885 3014

15 79 870 974 1326 1449 1531 2077 2317 2467 2627 2811 3083 3101 3132

4 582 660 902 1048 1482 1697 1744 1928 2628 2699 2728 3045 3104

175 395 429 1027 1061 1068 1154 1168 1175 2147 2359 2376 2613 2682

20 1388 2241 3118 3148

143 506 2067 3148

1594 2217 2705

398 988 2551

25 1149 2588 2654

678 2844 3115

1508 1547 1954

1199 1267 1710

30 2589 3163 3207

1 2583 2974

2766 2897 3166

35 929 1823 2742

1113 3007 3239

1753 2478 3127

40

45

0 509 1811

1672 2646 2984

965 1462 3230

5

3 1077 2917

1183 1316 1662

968 1593 3239

10

64 1996 2226

1442 2058 3181

513 973 1058

1263 3185 3229

15

681 1394 3017

419 2853 3217

3 2404 3175

2417 2792 2854

20

1879 2940 3235

647 1704 3060.

2. Устройство обработки данных по п. 1,

в котором, если строка исходной таблицы матрицы контроля четности представлена

25 i и длина битов четности LDPC-кода представлена M ,

$(2+360 \times (i-1))$ -й столбец матрицы контроля четности представляет собой столбец, полученный путем циклического сдвига $(1+360 \times (i-1))$ -го столбца матрицы контроля четности, для которой позиции элементов 1 представлены с помощью таблицы начальных значений матрицы контроля четности, на $q=M/360$ в направлении вниз.

30 3. Устройство обработки данных по п. 2,

в котором для $(1+360 \times (i-1))$ -го столбца матрицы контроля четности i -я строка таблицы начальных значений матрицы контроля четности представляет собой число строк элементов 1 в $(1+360 \times (i-1))$ -м столбце матрицы контроля четности, и

35 в котором для каждого столбца от $(2+360 \times (i-1))$ -го столбца до $(360 \times i)$ -го столбца, которые представляют собой столбцы, отличные от $(1+360 \times (i-1))$ -го столбца матрицы контроля четности, если числовое значение в i -й строке и j -м столбце таблицы начальных значений матрицы контроля четности представлено $h_{i,j}$, и номер строки j -го элемента 1 в w -м столбце матрицы H контроля четности представлен $H_{w,j}$, номер строки $H_{w,j}$ элемента 1 в w -м столбце, который представляет собой столбцы, отличные от $(1+360 \times (i-1))$ -го столбца матрицы контроля четности, выражен уравнением $H_{w,j} = \text{mod}\{h_{i,j} + \text{mod}((w-1), 360) \times M/360, M\}$.

4. Устройство обработки данных по п. 1, дополнительно содержащее:

45 блок перемежения битов четности, который выполняет перемежение только бита четности кодового бита LDPC-кода.

5. Устройство обработки данных по п. 1, дополнительно содержащее:

блок перемежения со скручиванием столбцов, который выполняет перемежение со скручиванием столбцов путем сохранения кодового бита LDPC-кода за счет сдвига в направлении столбца.

6. Устройство обработки данных по п. 1, дополнительно содержащее:

блок замены, который выполняет замену кодового бита LDPC-кода на символьный бит символа, соответствующего любому одному из сигнальных точек с предварительно определенным числом, которое определяется с помощью предварительно определенной

5 цифровой схемы модуляции.

7. Устройство обработки данных по п. 6, в котором блок замены заменяет кодовый бит, который хранится в направлении столбца и считывается в направлении строки.

8. Устройство обработки данных по п. 1,

в котором матрица контроля четности представляет собой матрицу контроля

10 четности, в которой отсутствует цикл 4.

9. Устройство обработки данных по п. 1,

в котором матрица контроля четности представляет собой матрицу контроля четности LDPC-кода, принадлежащего ансамблю LDPC-кода, в котором порог производительности представляет собой предварительно определенное значение или

15 меньшее, причем порог производительности представляет собой E_b/N_0 , при котором начинается падение BER.

10. Способ обработки данных, содержащий:

этап кодирования, на котором кодируют информационный бит в LDPC-код, имеющий

20 длину кода 16200 битов и кодовую скорость 12/15, на основании матрицы контроля четности кода разреженного контроля четности (LDPC),

где LDPC-код включает в себя информационный бит и бит четности,

где матрица контроля четности включает в себя информационную часть матрицы, соответствующую информационному биту, и часть матрицы четности, соответствующую

25 биту четности, где информационная часть матрицы представлена таблицей начальных значений матрицы контроля четности, и

где таблица начальных значений матрицы контроля четности представляет собой таблицу, представляющую позиции элементов 1 в информационной части матрицы с

30 интервалом 360 столбцов, и представляет собой

35

40

45

3 394 1014 1214 1361 1477 1534 1660 1856 2745 2987 2991 3124 3155
 59 136 528 781 803 928 1293 1489 1944 2041 2200 2613 2690 2847
 155 245 311 621 1114 1269 1281 1783 1995 2047 2672 2803 2885 3014
 5 79 870 974 1326 1449 1531 2077 2317 2467 2627 2811 3083 3101 3132
 4 582 660 902 1048 1482 1697 1744 1928 2628 2699 2728 3045 3104
 175 395 429 1027 1061 1068 1154 1168 1175 2147 2359 2376 2613 2682
 1388 2241 3118 3148
 10 143 506 2067 3148
 1594 2217 2705
 398 988 2551
 1149 2588 2654
 15 678 2844 3115
 1508 1547 1954
 1199 1267 1710
 20 2589 3163 3207
 1 2583 2974
 2766 2897 3166
 929 1823 2742
 25 1113 3007 3239
 1753 2478 3127
 0 509 1811
 1672 2646 2984
 30 965 1462 3230
 3 1077 2917
 1183 1316 1662
 968 1593 3239
 35 64 1996 2226
 1442 2058 3181
 513 973 1058
 40 1263 3185 3229
 681 1394 3017
 419 2853 3217
 3 2404 3175
 45 2417 2792 2854
 1879 2940 3235
 647 1704 3060.

11. Устройство обработки данных, содержащее:
 блок декодирования, который декодирует LDPC-код, который получается из данных,
 переданных из передающего устройства,

в котором передающее устройство включает в себя

5 блок кодирования, который кодирует информационный бит в LDPC-код, имеющий
 длину кода 16200 битов и кодовую скорость 12/15, на основании матрицы контроля
 четности кода разреженного контроля четности (LDPC),

в котором LDPC-код включает в себя информационный бит и бит четности,

10 где матрица контроля четности включает в себя информационную часть матрицы,
 соответствующую информационному биту, и часть матрицы четности, соответствующую
 биту четности,

где информационная часть матрицы представлена таблицей начальных значений
 матрицы контроля четности, и

15 где таблица начальных значений матрицы контроля четности представляет собой
 таблицу, представляющую позиции элементов 1 в информационной части матрицы с
 интервалом 360 столбцов, и представляет собой

3 394 1014 1214 1361 1477 1534 1660 1856 2745 2987 2991 3124 3155

59 136 528 781 803 928 1293 1489 1944 2041 2200 2613 2690 2847

20 155 245 311 621 1114 1269 1281 1783 1995 2047 2672 2803 2885 3014

79 870 974 1326 1449 1531 2077 2317 2467 2627 2811 3083 3101 3132

4 582 660 902 1048 1482 1697 1744 1928 2628 2699 2728 3045 3104

175 395 429 1027 1061 1068 1154 1168 1175 2147 2359 2376 2613 2682

25 1388 2241 3118 3148

143 506 2067 3148

1594 2217 2705

30 398 988 2551

1149 2588 2654

678 2844 3115

1508 1547 1954

35 1199 1267 1710

2589 3163 3207

1 2583 2974

40 2766 2897 3166

45

929 1823 2742

1113 3007 3239

1753 2478 3127

5

0 509 1811

1672 2646 2984

965 1462 3230

10

3 1077 2917

1183 1316 1662

968 1593 3239

64 1996 2226

15

1442 2058 3181

513 973 1058

1263 3185 3229

20

681 1394 3017

419 2853 3217

3 2404 3175

2417 2792 2854

25

1879 2940 3235

647 1704 3060.

12. Устройство обработки данных по п. 11,

в котором, если строка исходной таблицы матрицы контроля четности представлена i и длина битов четности LDPC-кода представлена M ,

30

$(2+360 \times (i-1))$ -й столбец матрицы контроля четности представляет собой столбец, полученный путем циклического сдвига $(1+360 \times (i-1))$ -го столбца матрицы контроля четности, для которой позиции элементов 1 представлены с помощью таблицы начальных значений матрицы контроля четности, на $q=M/360$ в направлении вниз.

13. Устройство обработки данных по п. 12,

35

в котором для $(1+360 \times (i-1))$ -го столбца матрицы контроля четности, i -я строка таблицы начальных значений матрицы контроля четности представляет собой номера строк элементов 1 в $(1+360 \times (i-1))$ -м столбце матрицы контроля четности, и

в котором для каждого столбца с $(2+360 \times (i-1))$ -го столбца по $(360 \times i)$ -й столбец, которые представляют собой столбцы, отличные от $(1+360 \times (i-1))$ -го столбца матрицы контроля четности, если числовое значение в i -ой строке и j -м столбце таблицы начальных значений матрицы контроля четности представлено $h_{i,j}$, и номер строки j -го элемента 1 в w -м столбце матрицы H контроля четности представлен $H_{w,j}$, номер строки H_{w-j} элемента 1 в w -м столбце, который представляет собой столбцы, отличные от $(1+360 \times (i-1))$ -го столбца матрицы контроля четности, выражен уравнением

45

$H_{w-j} = \text{mod}\{h_{i,j} + \text{mod}((w-1), 360) \times M/360, M\}$.

14. Устройство обработки данных по п. 11, дополнительно содержащее:

блок деперемежения со скручиванием столбцов, который при выполнении

перемежения со скручиванием столбцов путем сохранения кодового бита LDPC-кода за счет его сдвига в направлении столбца, возвращает кодовый бит LDPC-кода в исходную последовательность для того, чтобы выполнить деперемежение со скручиванием столбцов.

- 5 15. Устройство обработки данных по п. 11, дополнительно содержащее:
 блок обратной замены, который при выполнении процесса замены, предназначенного для замены кодового бита LDPC-кода на символьный бит символа, соответствующего любому одному из сигнальных точек с предварительно определенным числом, которое определяется с помощью предварительно определенной схемой цифровой модуляции,
 10 выполняет обратный процесс замены, предназначенный для возврата кодового бита, позиция которого заменяется в качестве символьного бита, в исходную позицию.

16. Устройство обработки данных по п. 15,
 в котором блок обратной замены выполняет обратный процесс замены, предназначенный для возврата кодового бита, который хранится в направлении строки
 15 и считывается в направлении столбца, в исходную позицию.

17. Устройство обработки данных по п. 11,
 в котором матрица контроля четности представляет собой матрицу контроля четности, в которой отсутствует цикл 4.

18. Устройство обработки данных по п. 11,
 20 в котором матрица контроля четности представляет собой матрицу контроля четности LDPC-кода, принадлежащего ансамблю LDPC-кода, в котором порог производительности представляет собой предварительно определенное значение или меньшее, которое обнаруживается с помощью эволюции плотности многореберного типа, причем в данном случае порог производительности представляет собой E_b/N_0 ,
 25 при котором начинается падение BER.

19. Способ обработки данных, содержащий:
 этап декодирования, на котором декодируют LDPC-код, который получается из данных, переданных из передающего устройства,
 в котором передающее устройство включает в себя
 30 блок кодирования, который кодирует информационный бит в LDPC-код, имеющий длину кода 16200 битов и кодовую скорость 12/15, на основании матрицы контроля четности кода разреженного контроля четности (LDPC),
 где LDPC-код включает в себя информационный бит и бит четности,
 где матрица контроля четности включает в себя информационную часть матрицы,
 35 соответствующую информационному биту, и часть матрицы четности, соответствующую биту четности,

где информационная часть матрицы представлена таблицей начальных значений матрицы контроля четности, и

- где таблица начальных значений матрицы контроля четности представляет собой
 40 таблицу, представляющую позиции элементов 1 в информационной части матрицы с интервалом 360 столбцов, и представляет собой

3 394 1014 1214 1361 1477 1534 1660 1856 2745 2987 2991 3124 3155
59 136 528 781 803 928 1293 1489 1944 2041 2200 2613 2690 2847
155 245 311 621 1114 1269 1281 1783 1995 2047 2672 2803 2885 3014
5 79 870 974 1326 1449 1531 2077 2317 2467 2627 2811 3083 3101 3132
4 582 660 902 1048 1482 1697 1744 1928 2628 2699 2728 3045 3104
175 395 429 1027 1061 1068 1154 1168 1175 2147 2359 2376 2613 2682
10 1388 2241 3118 3148
143 506 2067 3148
1594 2217 2705
398 988 2551
15 1149 2588 2654
678 2844 3115
1508 1547 1954
1199 1267 1710
20 2589 3163 3207
1 2583 2974
2766 2897 3166
25 929 1823 2742
1113 3007 3239
1753 2478 3127
0 509 1811
30 1672 2646 2984
965 1462 3230
3 1077 2917

35

40

45

1183 1316 1662

968 1593 3239

64 1996 2226

5 1442 2058 3181

513 973 1058

1263 3185 3229

10 681 1394 3017

419 2853 3217

3 2404 3175

2417 2792 2854

15 1879 2940 3235

647 1704 3060.

20. Устройство обработки данных, содержащее:

блок декодирования, который декодирует LDPC-код, имеющий длину кода 16200 битов и кодовую скорость 12/15, на основании матрицы контроля четности кода

20 разреженного контроля четности (LDPC),

где LDPC-код включает в себя информационный бит и бит четности,

где матрица контроля четности включает в себя информационную часть матрицы, соответствующую информационному биту, и часть матрицы четности, соответствующую биту четности,

25 где информационная часть матрицы представлена таблицей начальных значений матрицы контроля четности, и

где таблица начальных значений матрицы контроля четности представляет собой таблицу, представляющую позиции элементов 1 в информационной части матрицы с интервалом 360 столбцов, и представляет собой

30 3 394 1014 1214 1361 1477 1534 1660 1856 2745 2987 2991 3124 3155

59 136 528 781 803 928 1293 1489 1944 2041 2200 2613 2690 2847

155 245 311 621 1114 1269 1281 1783 1995 2047 2672 2803 2885 3014

35 79 870 974 1326 1449 1531 2077 2317 2467 2627 2811 3083 3101 3132

4 582 660 902 1048 1482 1697 1744 1928 2628 2699 2728 3045 3104

175 395 429 1027 1061 1068 1154 1168 1175 2147 2359 2376 2613 2682

1388 2241 3118 3148

40 143 506 2067 3148

1594 2217 2705

398 988 2551

45

1149 2588 2654

678 2844 3115

1508 1547 1954

5

1199 1267 1710

2589 3163 3207

1 2583 2974

10

2766 2897 3166

929 1823 2742

1113 3007 3239

1753 2478 3127

15

0 509 1811

1672 2646 2984

965 1462 3230

3 1077 2917

20

1183 1316 1662

968 1593 3239

64 1996 2226

25

1442 2058 3181

513 973 1058

1263 3185 3229

681 1394 3017

30

419 2853 3217

3 2404 3175

2417 2792 2854

35

1879 2940 3235

647 1704 3060.

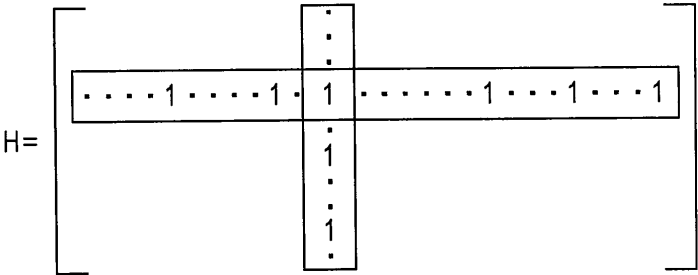
40

45

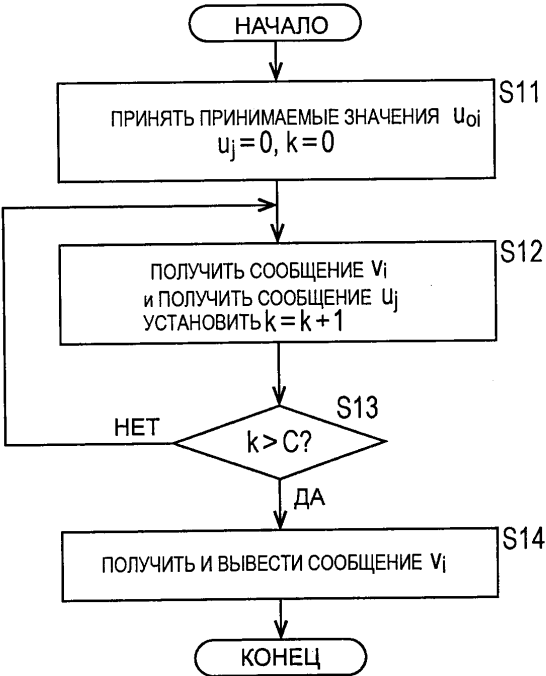
1

1 / 53

Фиг. 1



Фиг. 2



2

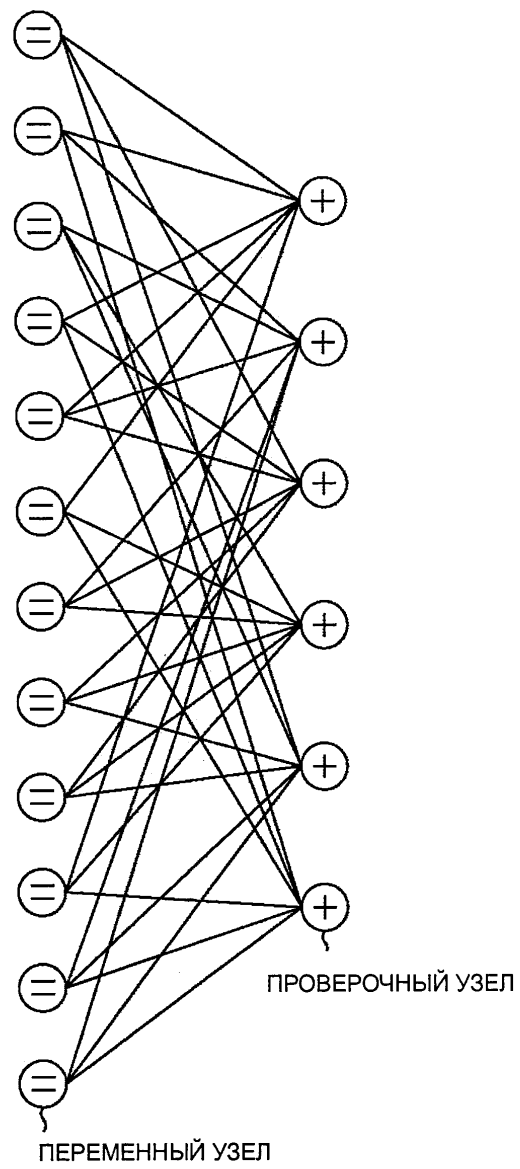
2/53

Фиг. 3

$$H = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 1 \end{bmatrix}$$

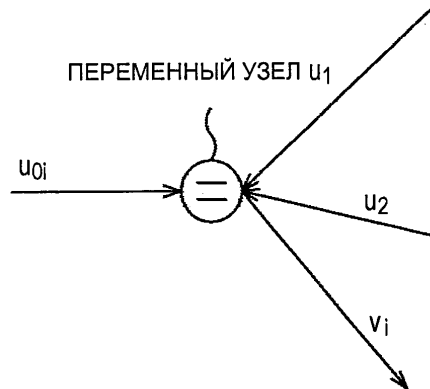
3/53

Фиг. 4

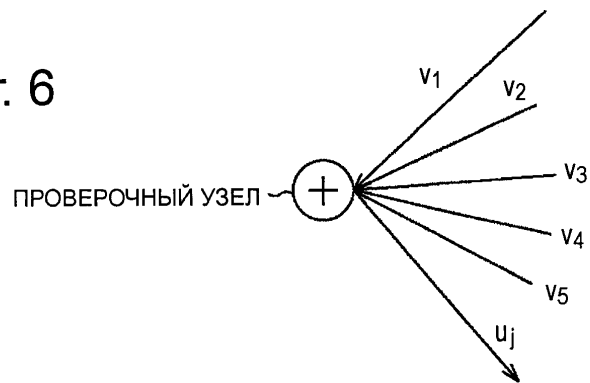


4/53

Фиг. 5



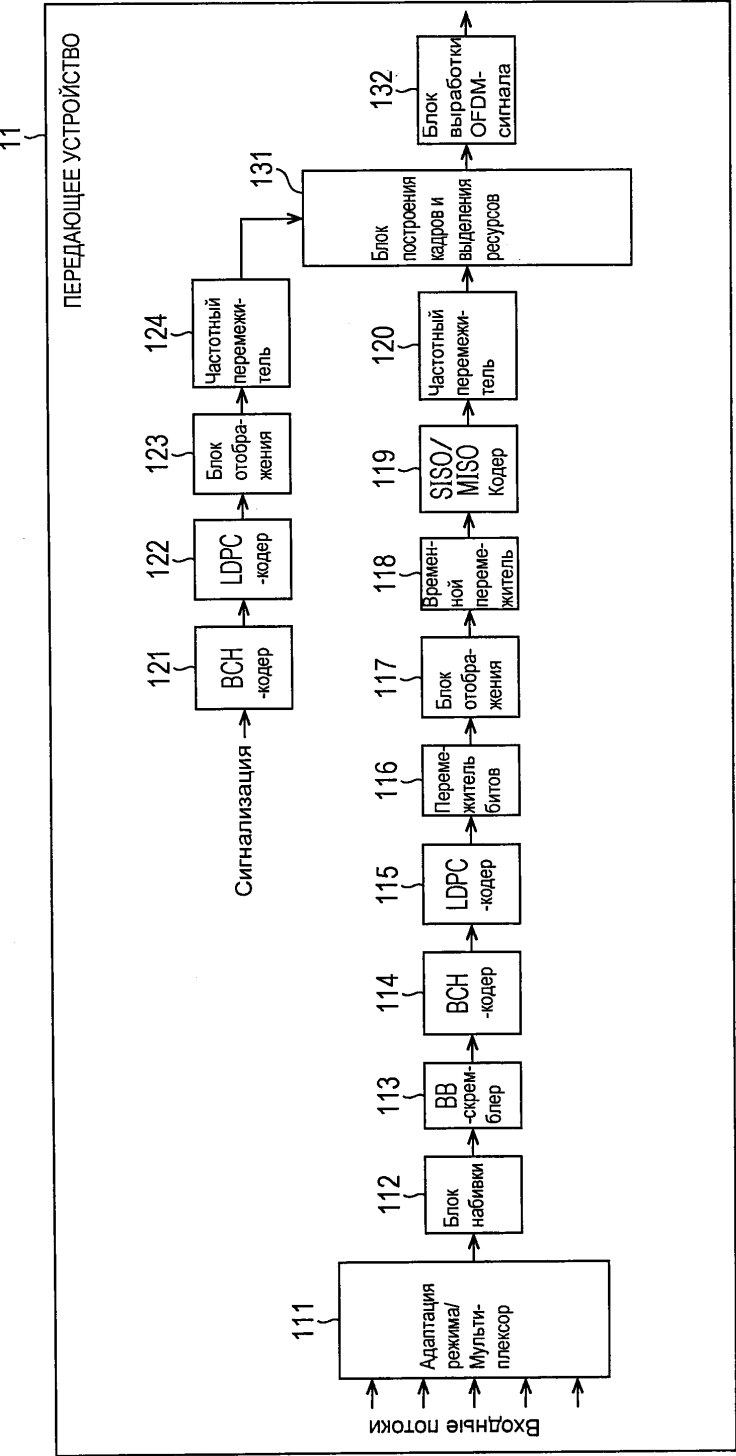
Фиг. 6



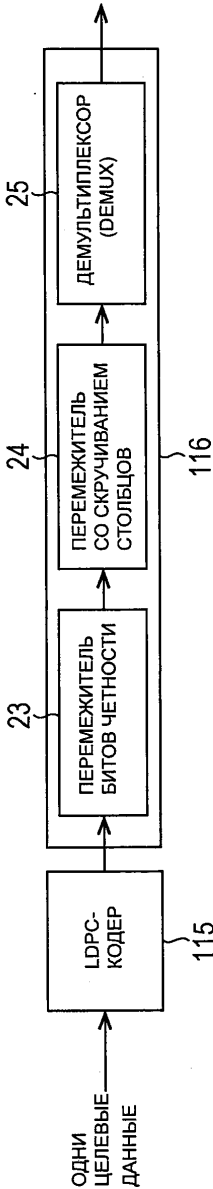
Фиг. 7



Фиг. 8

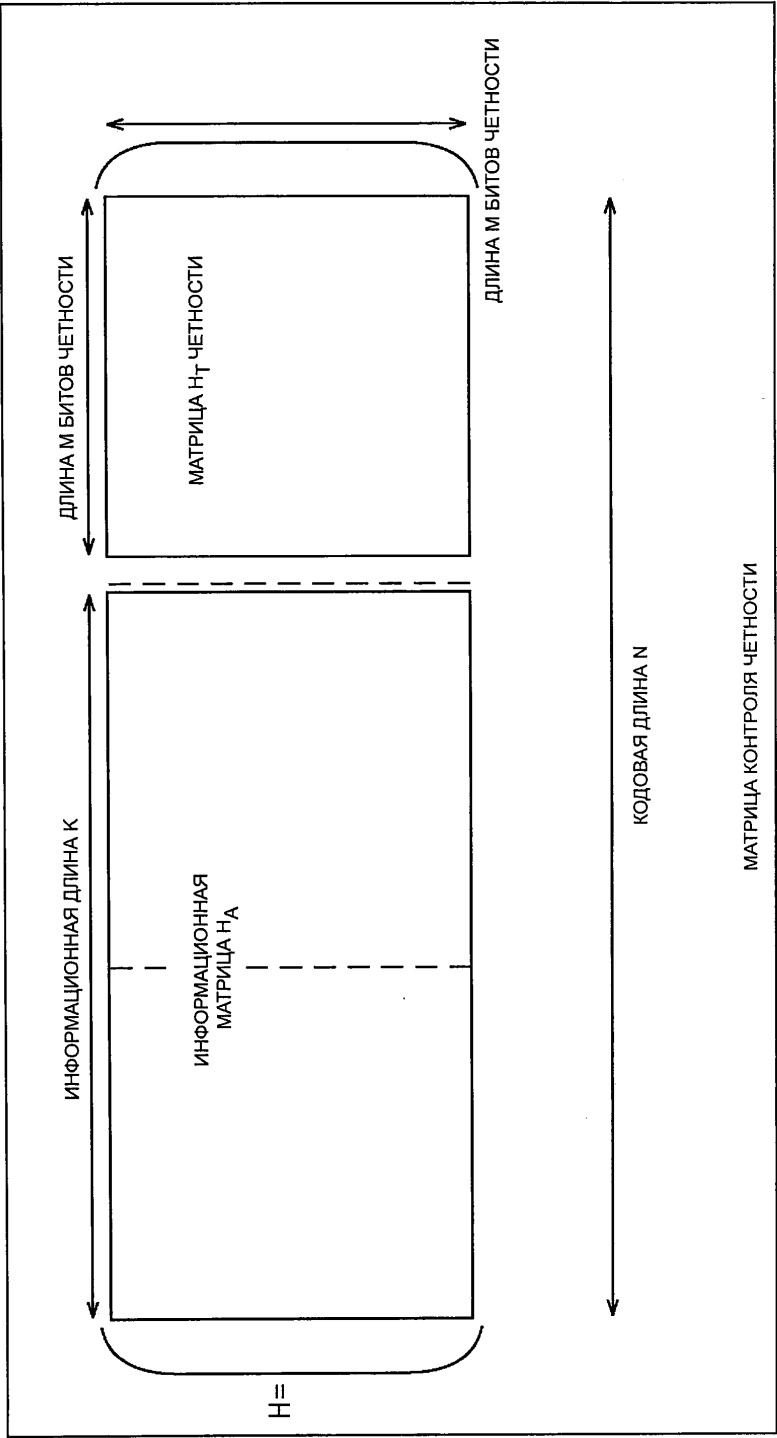


Фиг. 9



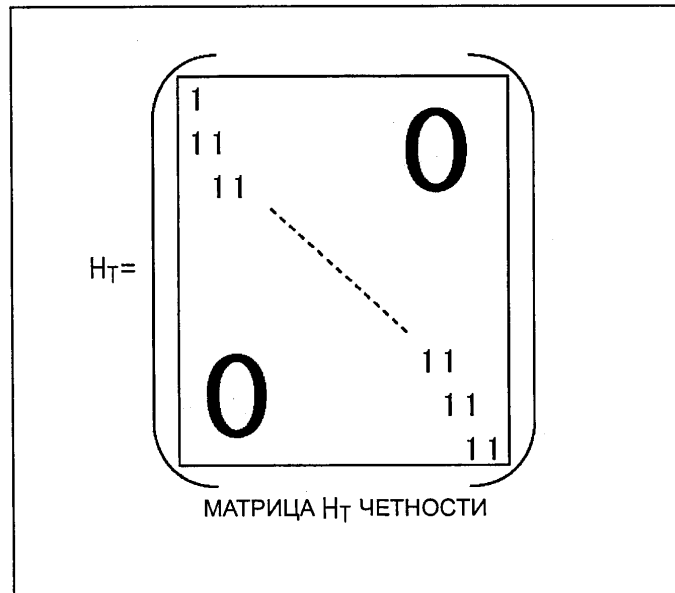
7/53

Фиг. 10

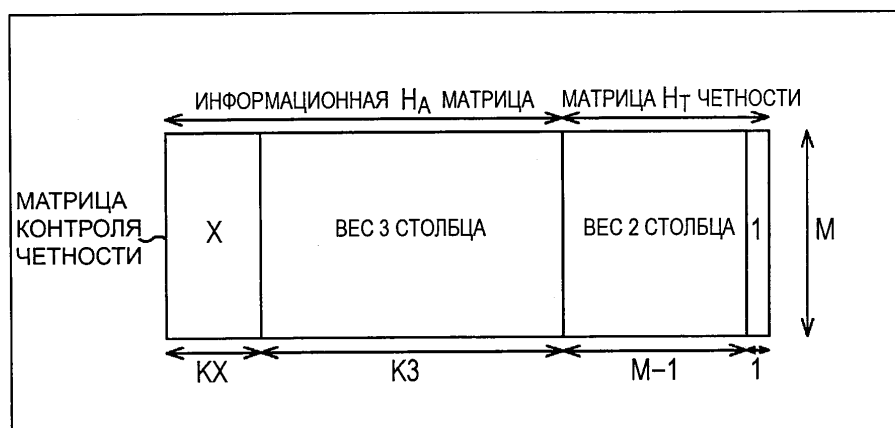


8/53

Фиг. 11



Фиг. 12

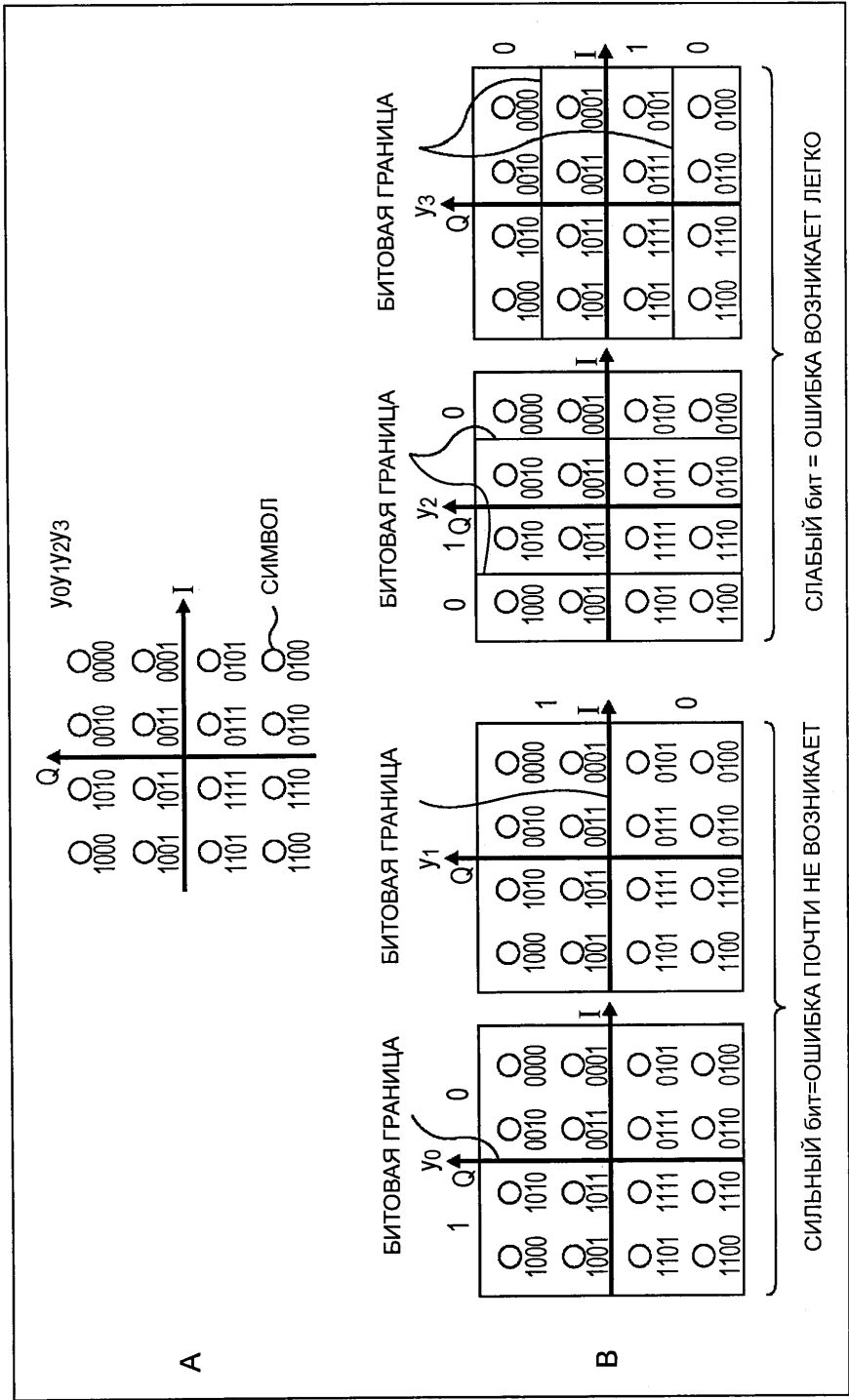


Фиг. 13

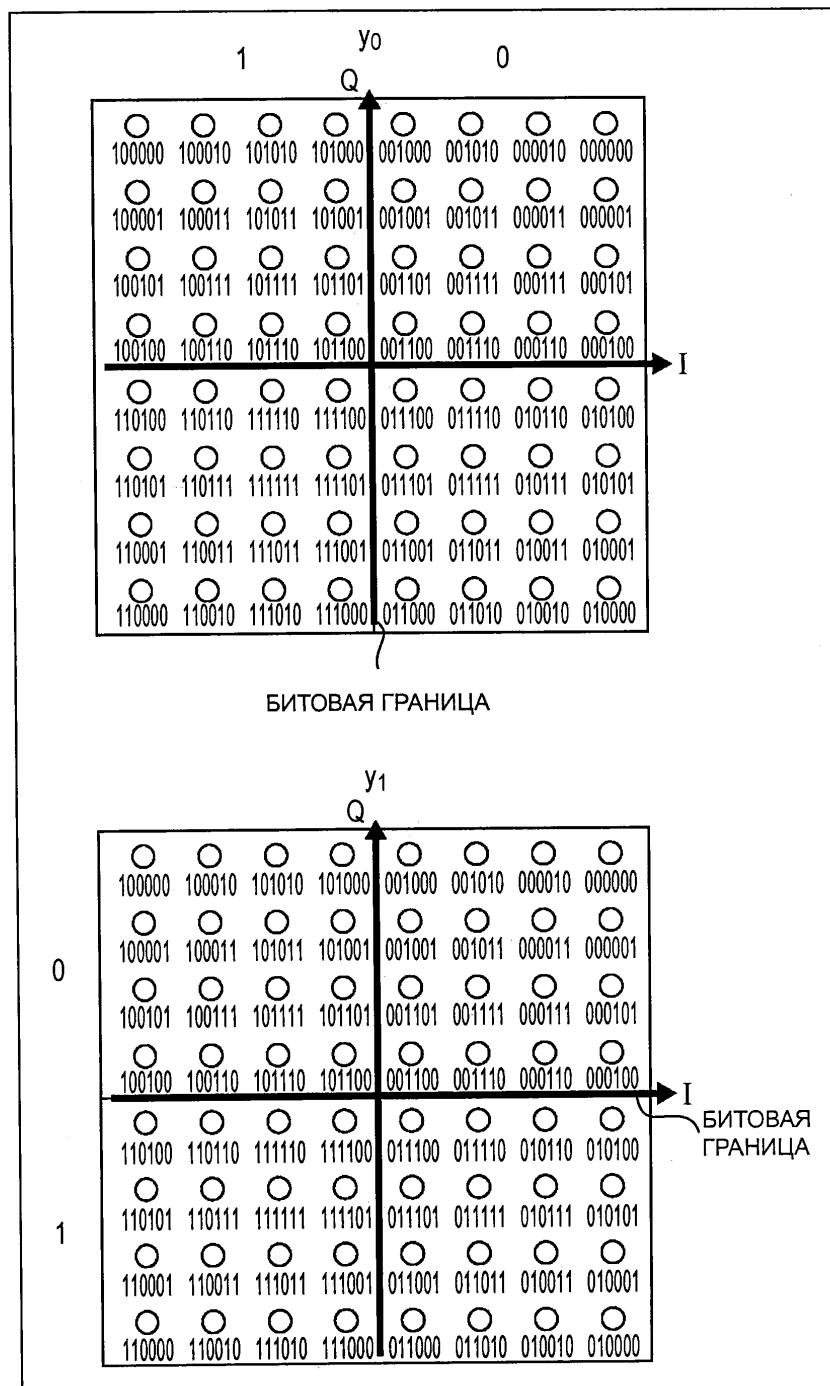
НОМИНАЛЬНАЯ КОДОВАЯ СКОРОСТЬ	N=64800					N=16200				
	X	KX	K3	M		X	KX	K3	M	
1/4	12	5400	10800	48600		12	1440	1800	12960	
1/3	12	7200	14400	43200		12	1800	3600	10800	
2/5	12	8640	17280	38880		12	2160	4320	9720	
1/2	8	12960	19440	32400		8	1800	5400	9000	
3/5	12	12960	25920	25920		12	3240	6480	6480	
2/3	13	4320	38880	21600		13	1080	9720	5400	
3/4	12	5400	43200	16200		12	360	11520	4320	
4/5	11	6480	45360	12960		-	0	12600	3600	
5/6	13	5400	48600	10800		13	360	12960	2880	
8/9	4	7200	50400	7200		4	1800	12600	1800	
9/10	4	6480	51840	6480		-	-	-	-	

число столбцов
для каждого
веса столбца

Фиг. 14

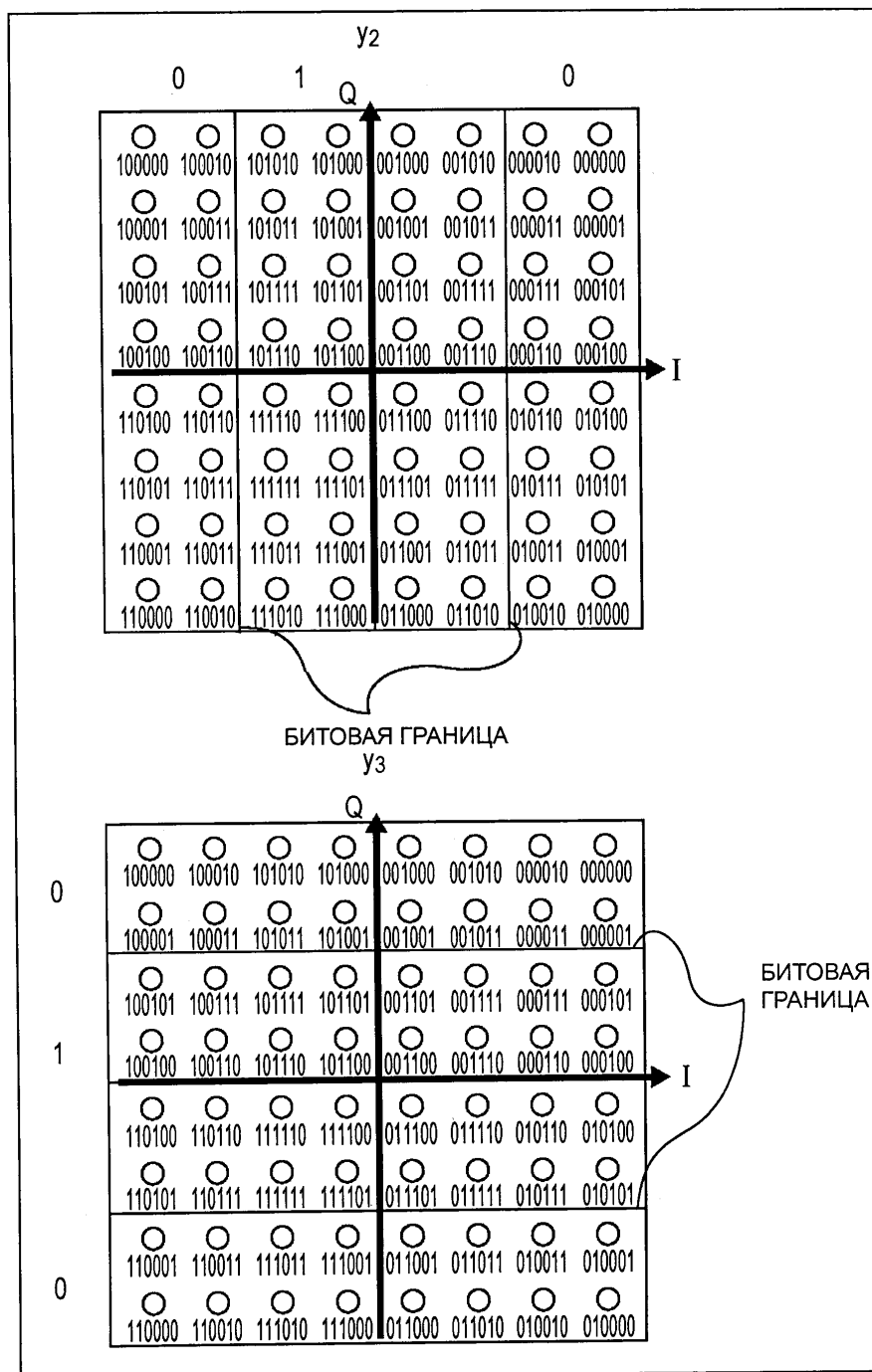


11/53
Фиг. 15

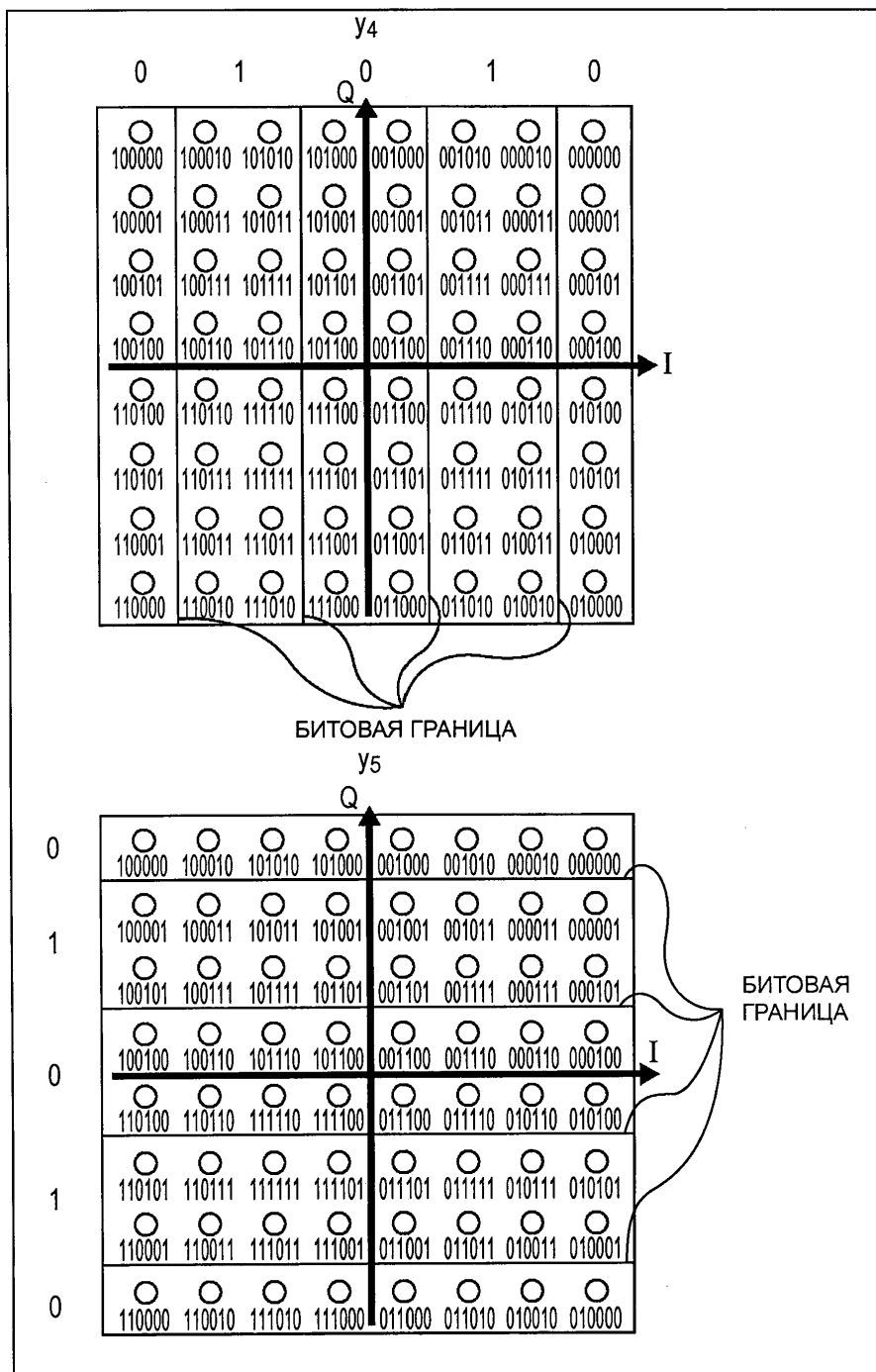


12/53

Фиг. 16

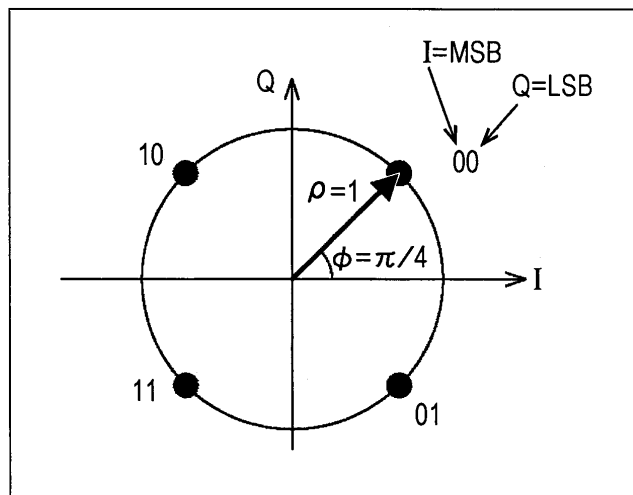


13/53
Фиг. 17

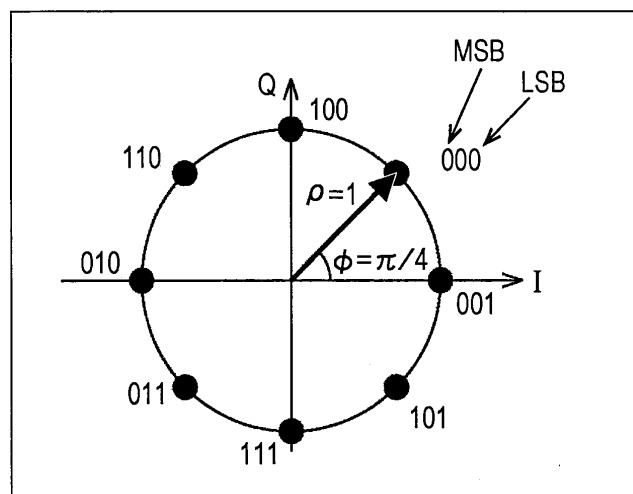


14/53

Фиг. 18

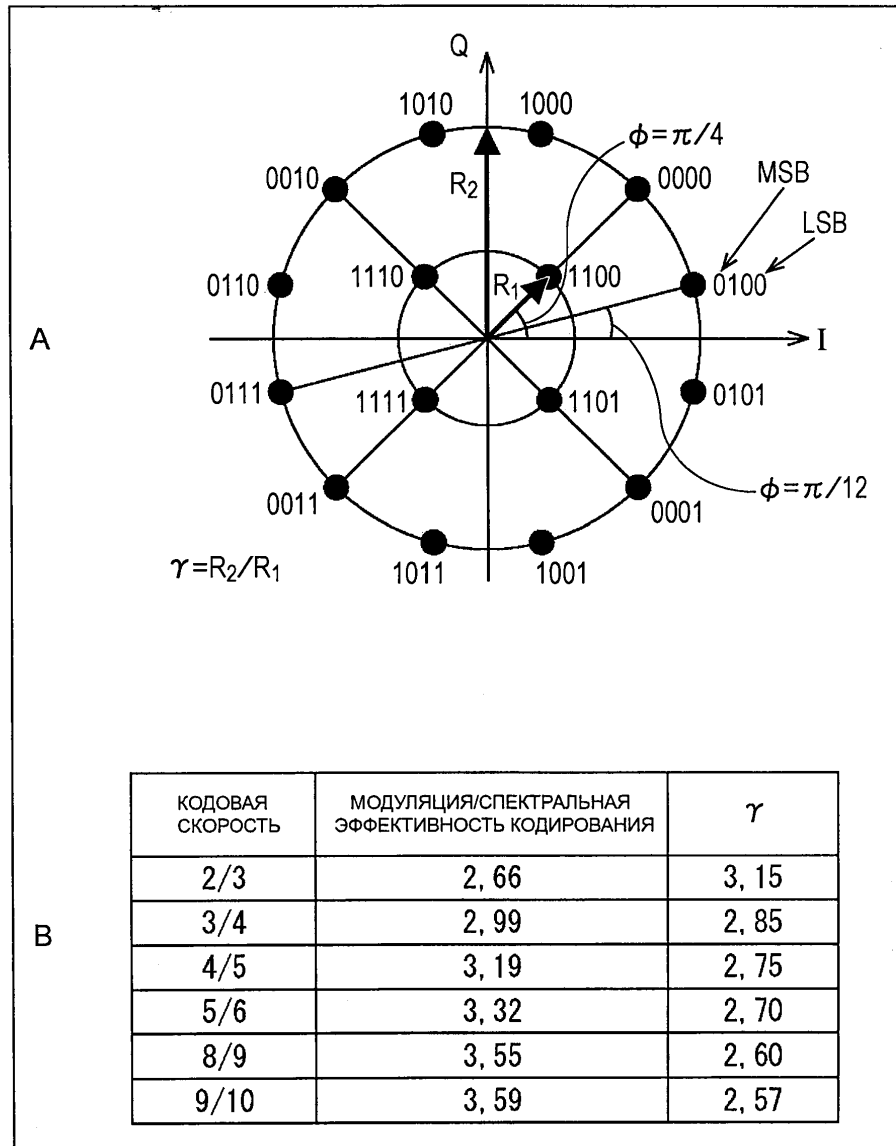


Фиг. 19



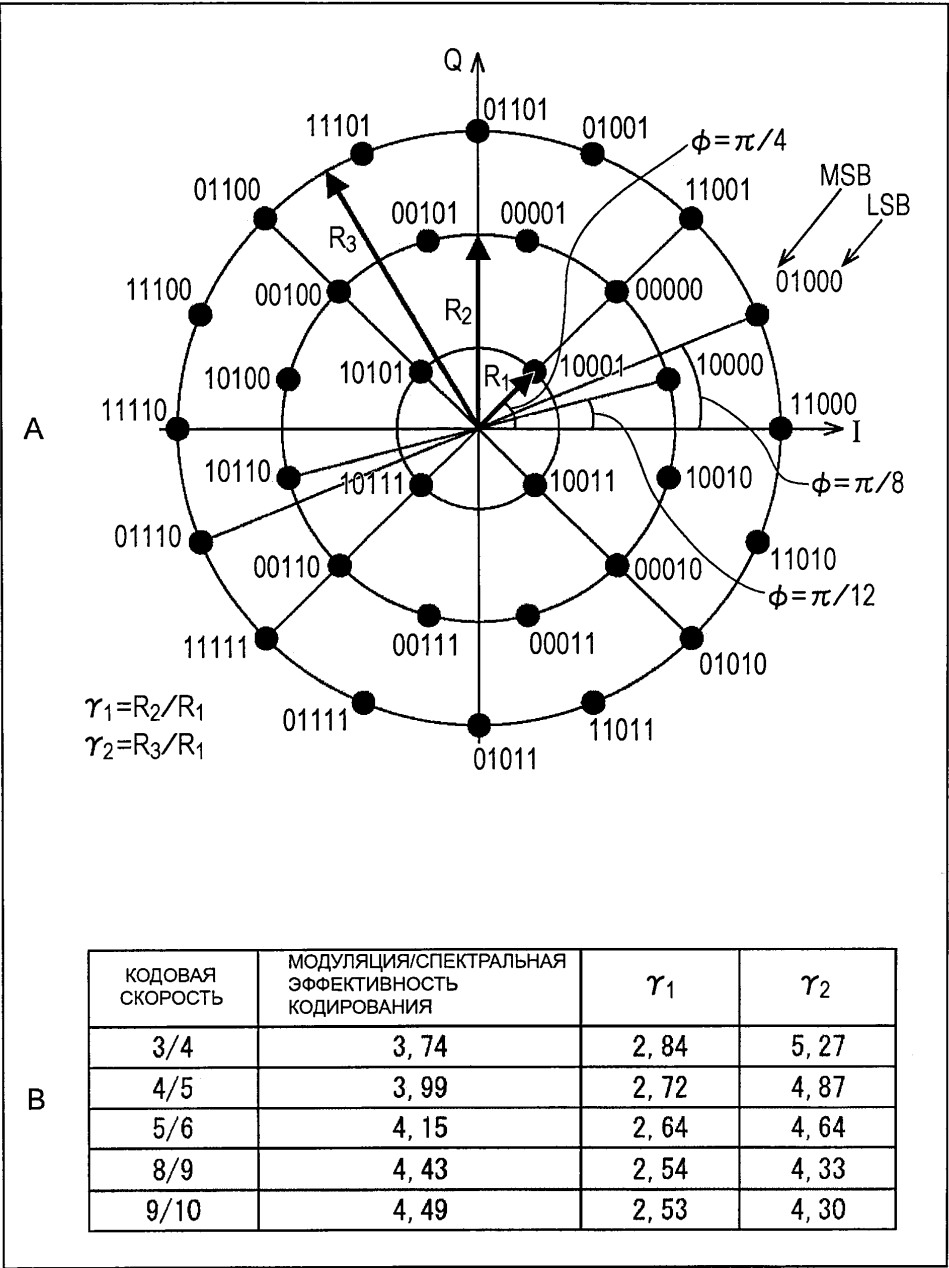
15/53

Фиг. 20

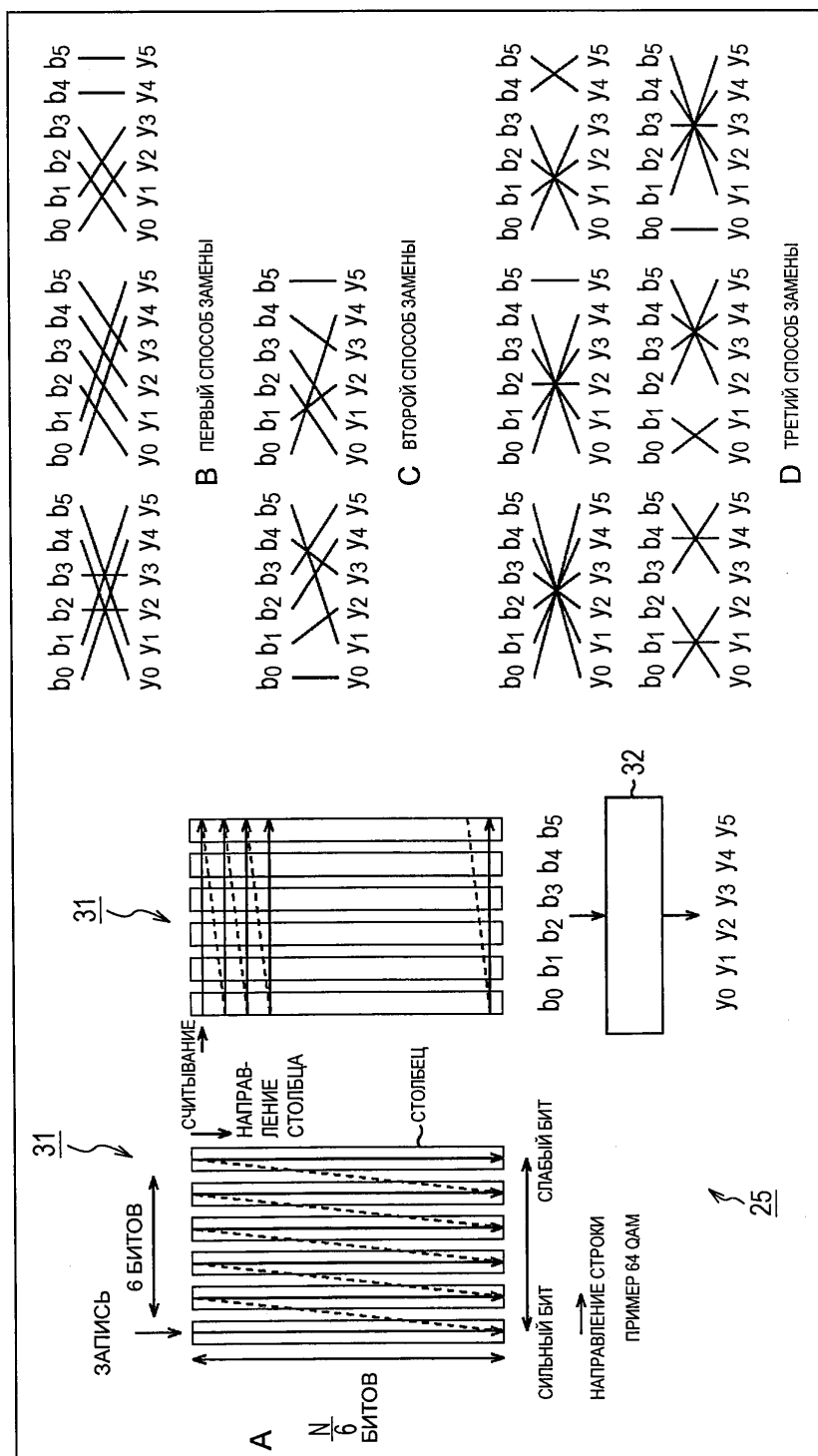


16/53

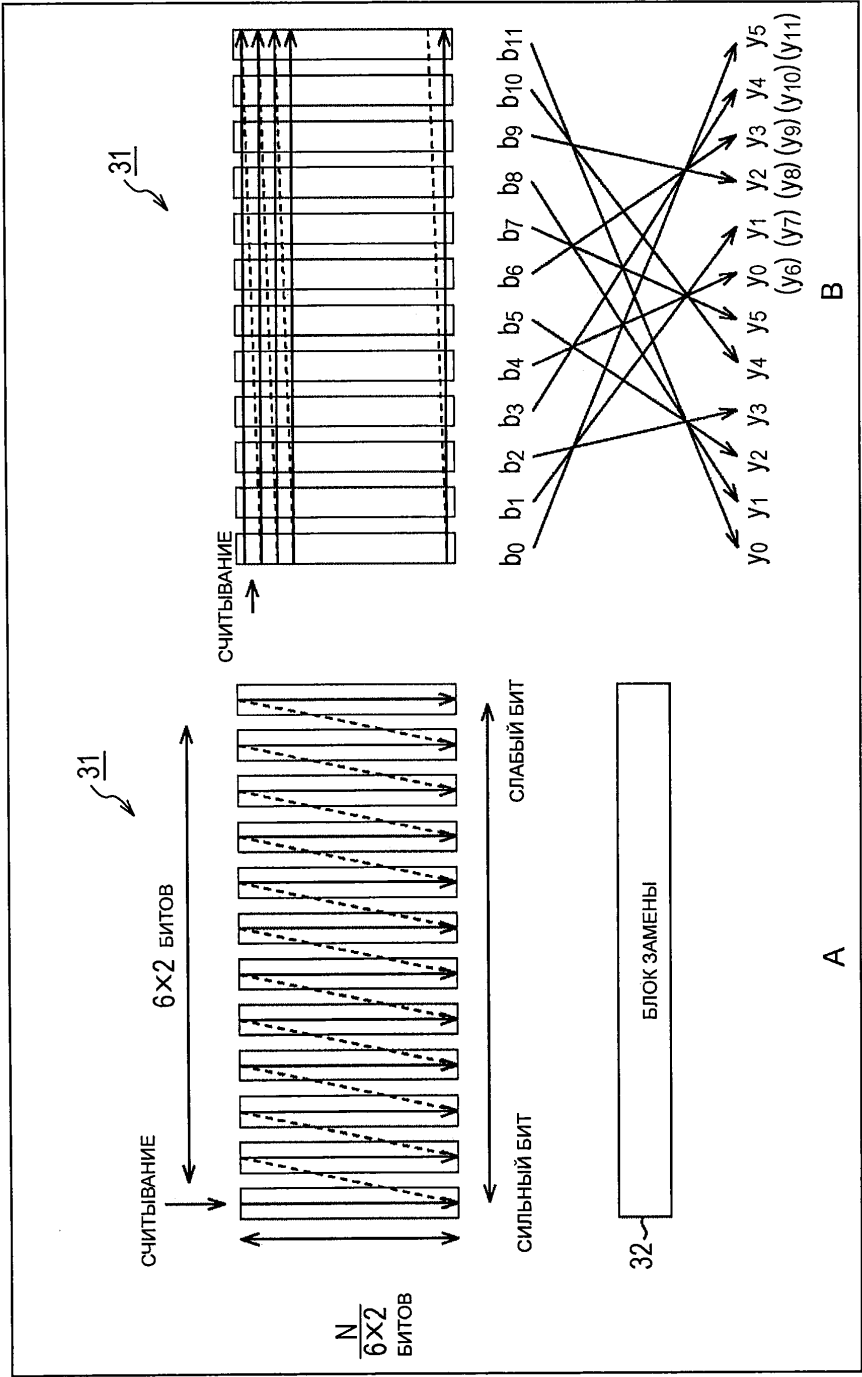
Фиг. 21



Фиг. 22

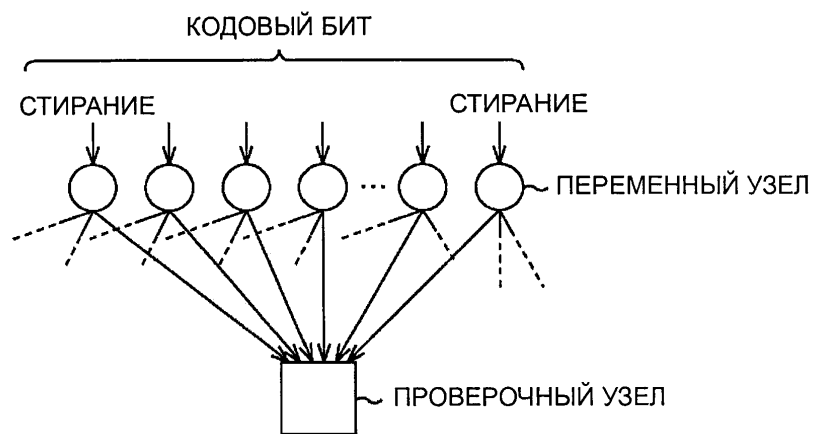


Фиг. 23



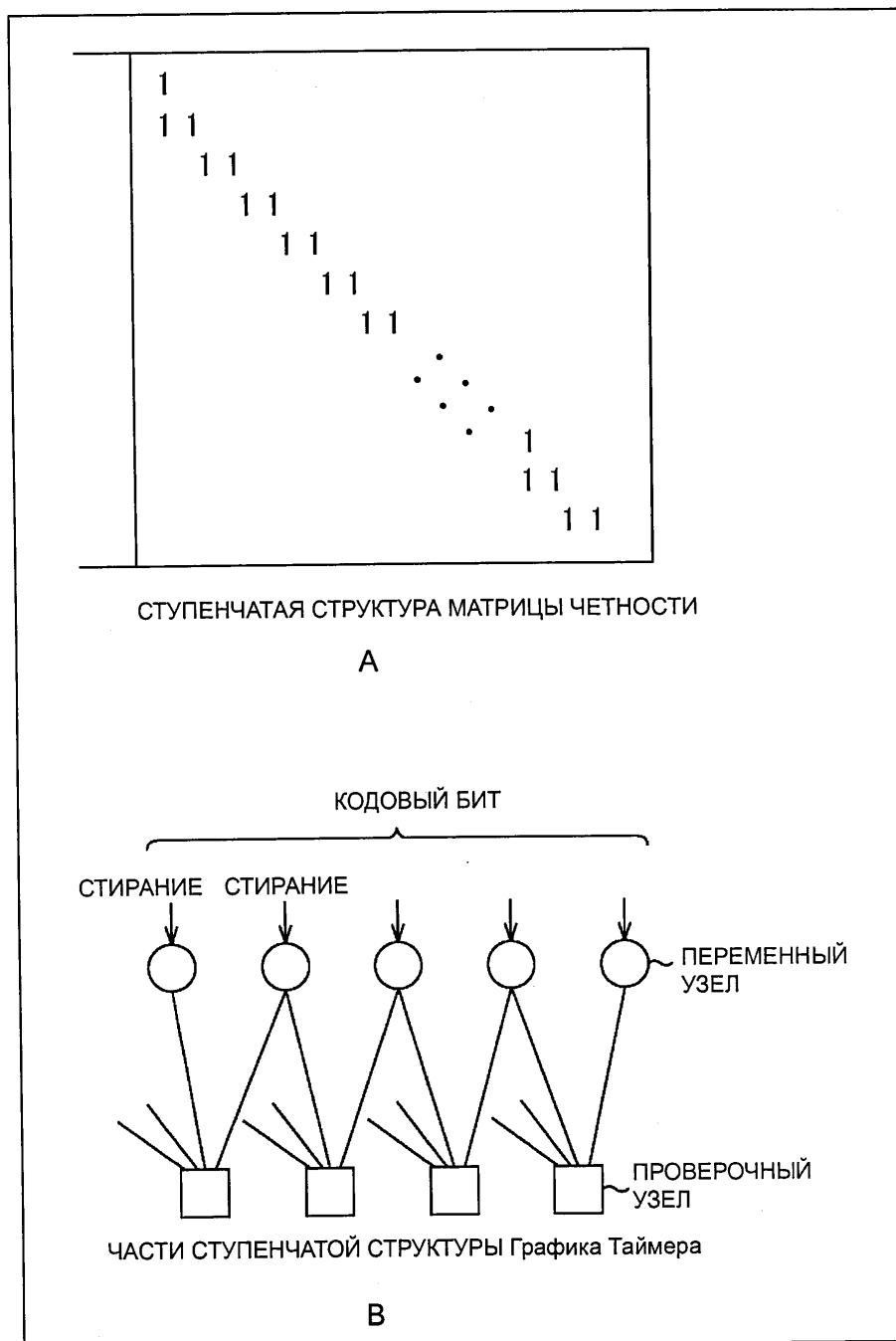
19/53

Фиг. 24

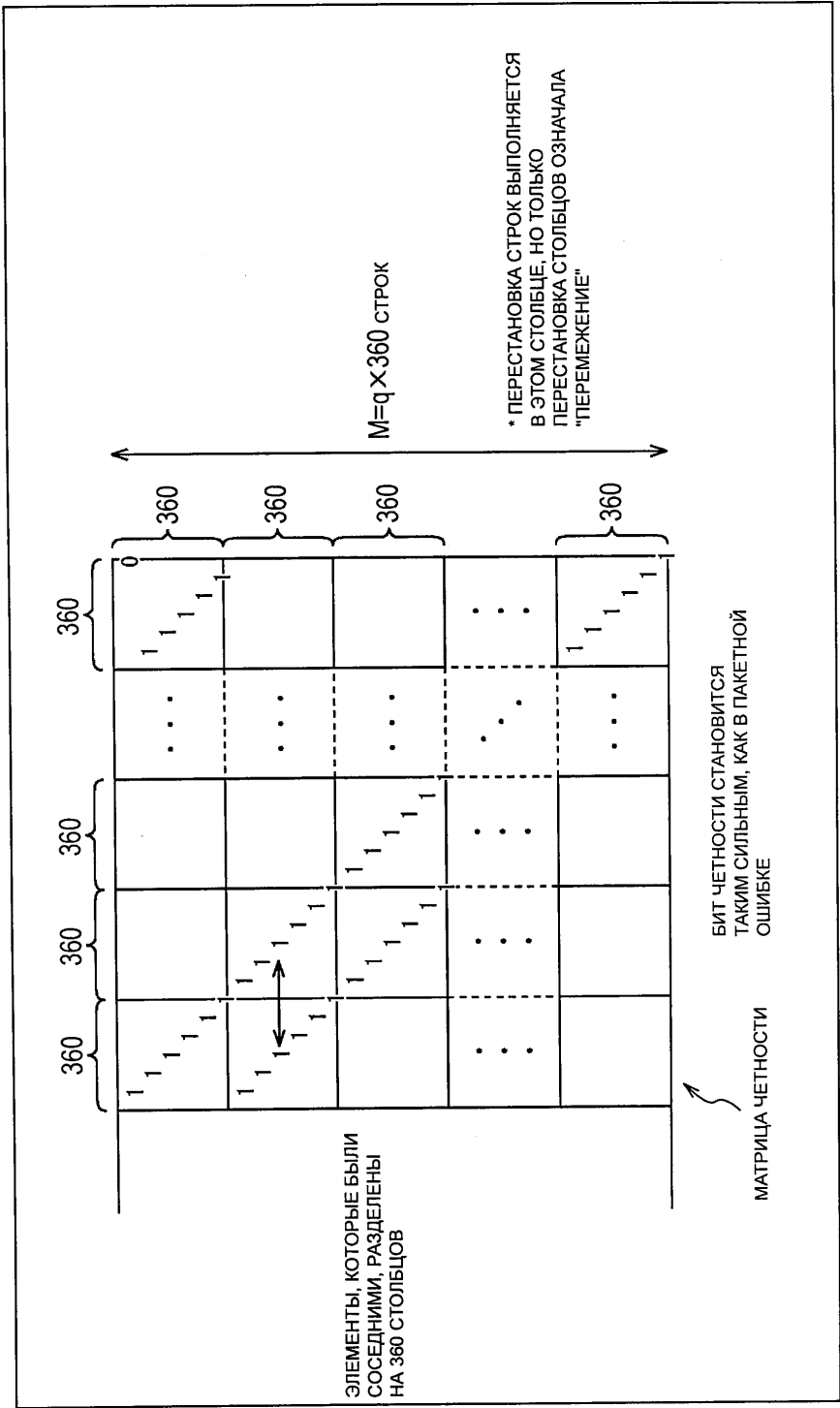


20 / 53

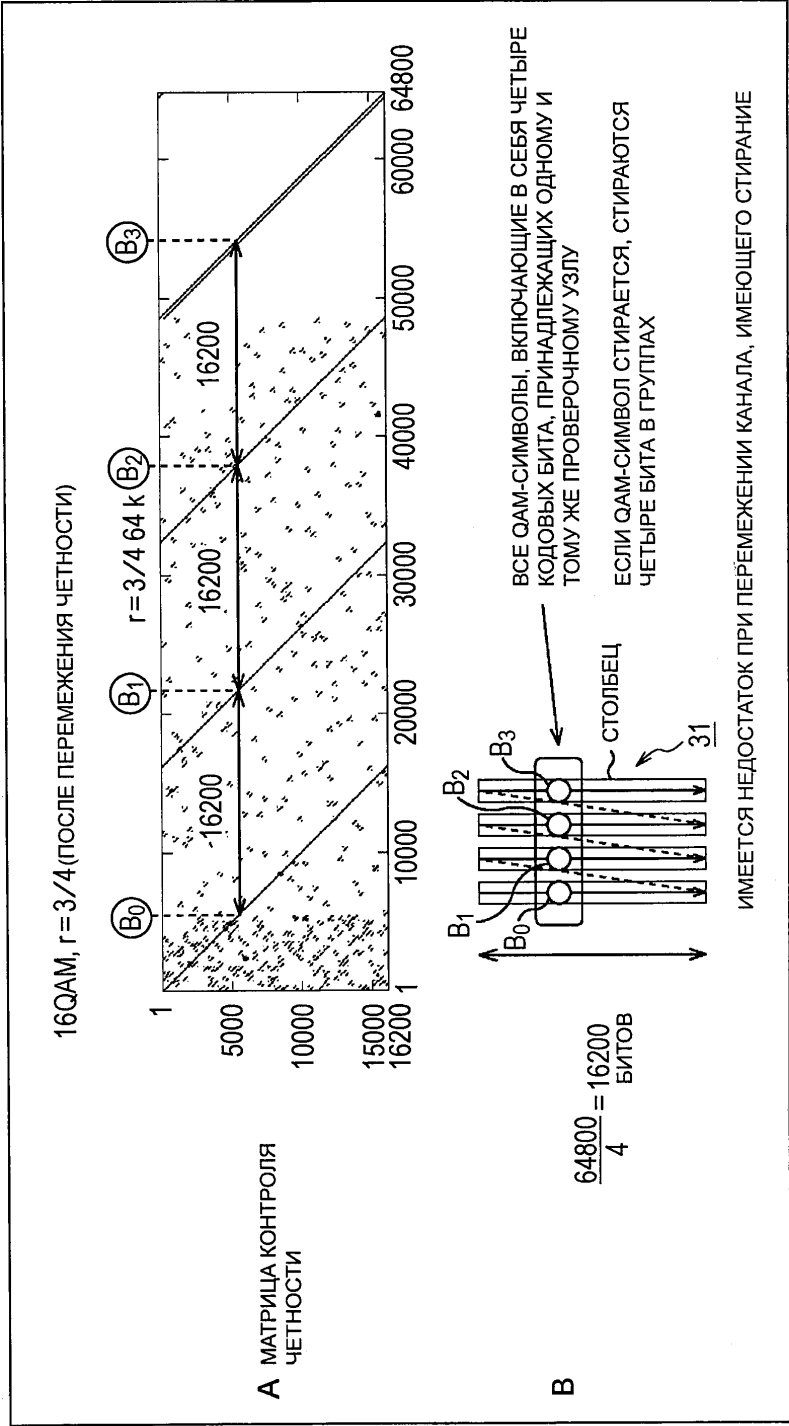
Фиг. 25



Фиг. 26

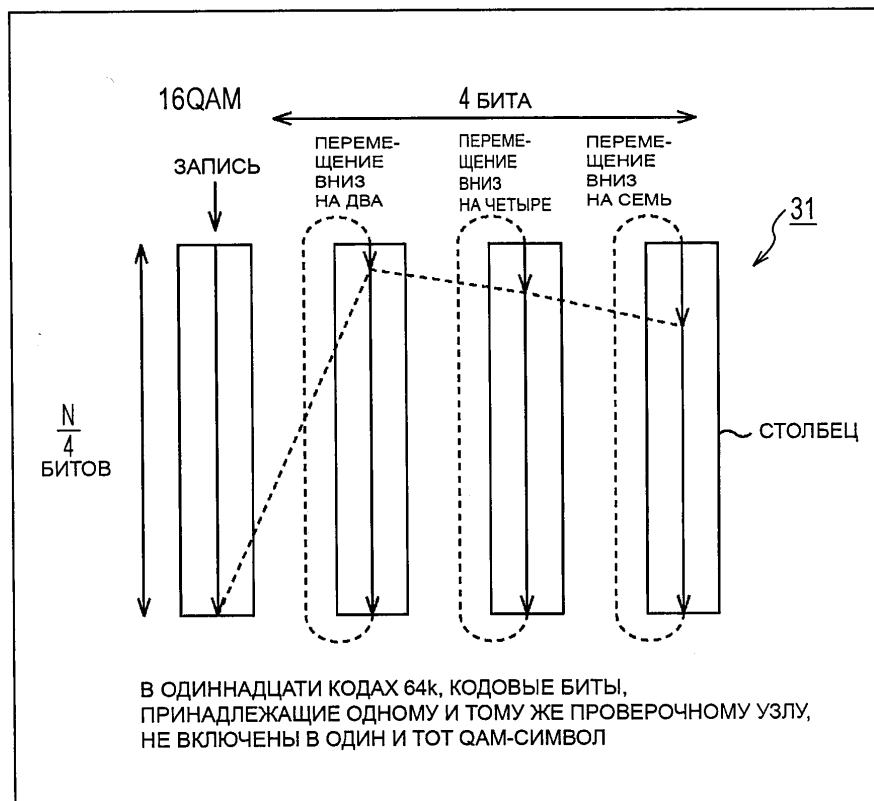


Фиг. 27



23/53

Фиг. 28



Фиг. 29

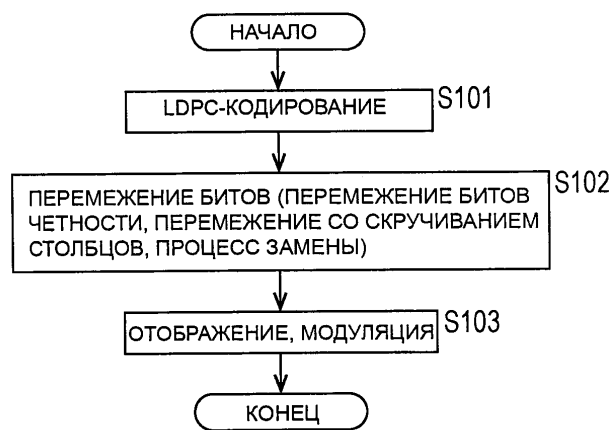
			ПОЗИЦИЯ НАЧАЛА ЗАПИСИ СООТВЕТСТВУЮЩИХ mb СТОЛБЦОВ																							
НЕОБХОДИ- МОЕ ЧИСЛО mb СТОЛБЦОВ ПАМЯТИ	b = 1 (ПЕРВЫЙ – ТРЕТИЙ СПОСОБЫ ЗАМЕНЫ))	b = 2 (ЧЕТВЕРТЫЙ СПОСОБ ЗАМЕНЫ))	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
2	QPSK		0	2																						
4	16QAM	QPSK	0	2	4	7																				
6	64QAM		0	2	5	9	10	13																		
8	256QAM	16QAM	0	0	2	4	4	5	7	7																
10	1024QAM		0	3	6	8	11	13	15	17	18	20														
12	4096QAM	64QAM	0	0	2	2	3	4	4	5	5	7	8	9												
16		256QAM	0	2	2	2	2	3	7	15	16	20	22	22	27	27	28	32								
20		1024QAM	0	1	3	4	5	6	6	9	13	14	14	16	21	21	23	25	25	26	28	30				
24		4096QAM	0	5	8	8	8	8	10	10	10	12	13	16	17	19	21	22	23	26	37	39	40	41	41	41

Фиг. 30

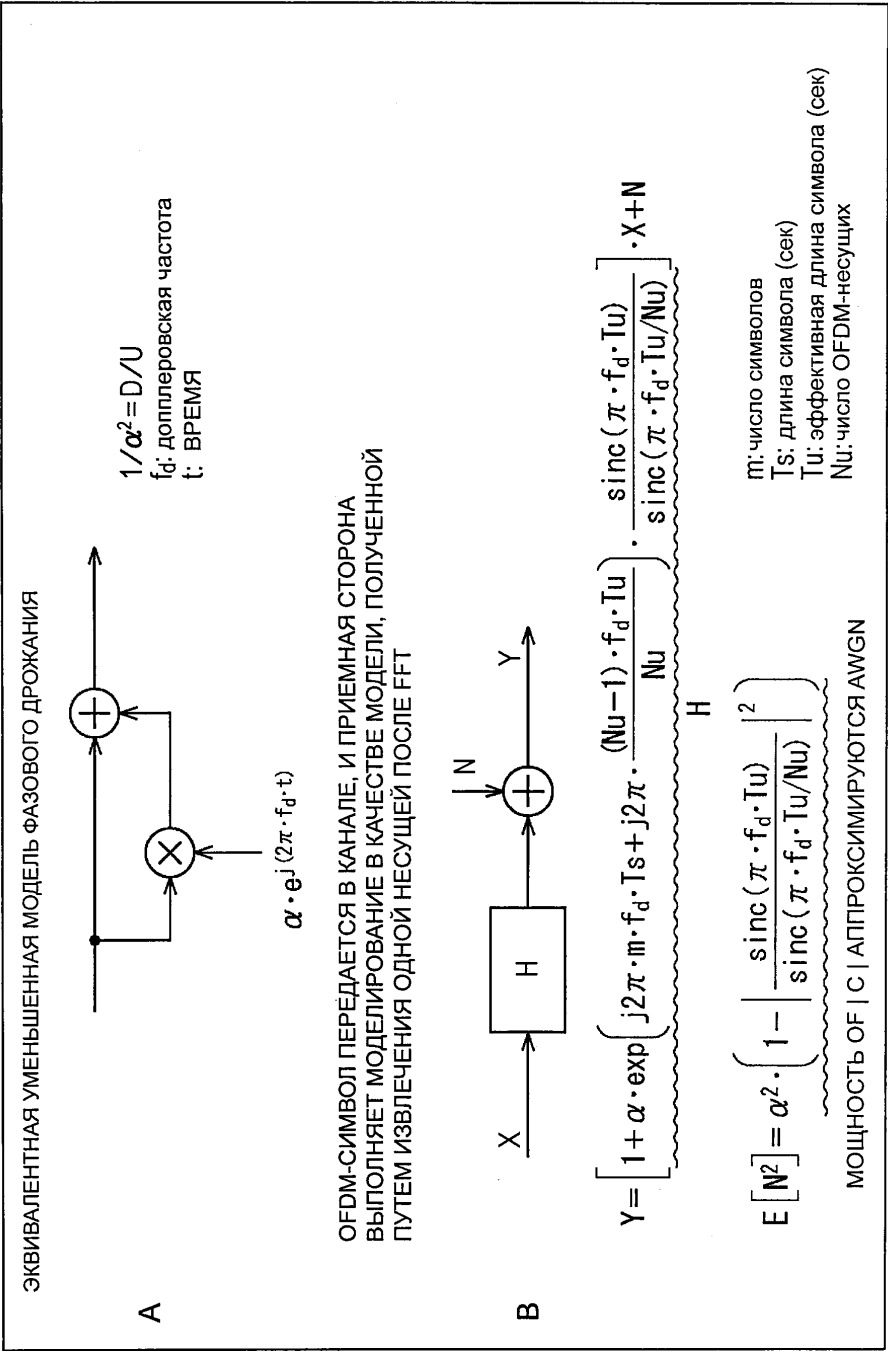
ПОЗИЦИЯ НАЧАЛА ЗАПИСИ СООТВЕТСТВУЮЩИХ mb СТОЛБЦОВ																											
НЕОБХОДИ- МОЕ ЧИСЛО mb СТОЛБЦОВ ПАМЯТИ	b = 1 (ПЕРВЫЙ – ТРЕТИЙ СПОСОБЫ ЗАМЕНЬ)	b = 2 (ЧЕТВЕРТЫЙ СПОСОБ ЗАМЕНЬ)	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	
2	QPSK		0	0																							
4	16QAM	QPSK	0	2	3	3																					
6	64QAM		0	0	2	3	7	7																			
8	256QAM	16QAM	0	0	0	1	7	20	20	21																	
10	1024QAM		0	1	2	2	3	3	4	4	5	7															
12	4096QAM	64QAM	0	0	0	2	2	2	3	3	3	6	7	7													
20		1024QAM	0	0	0	2	2	2	2	2	5	5	5	5	5	7	7	7	7	8	8	10					
24		4096QAM	0	0	0	0	0	0	0	1	1	1	2	2	2	3	7	9	9	10	10	10	10	10	10	11	

26 / 53

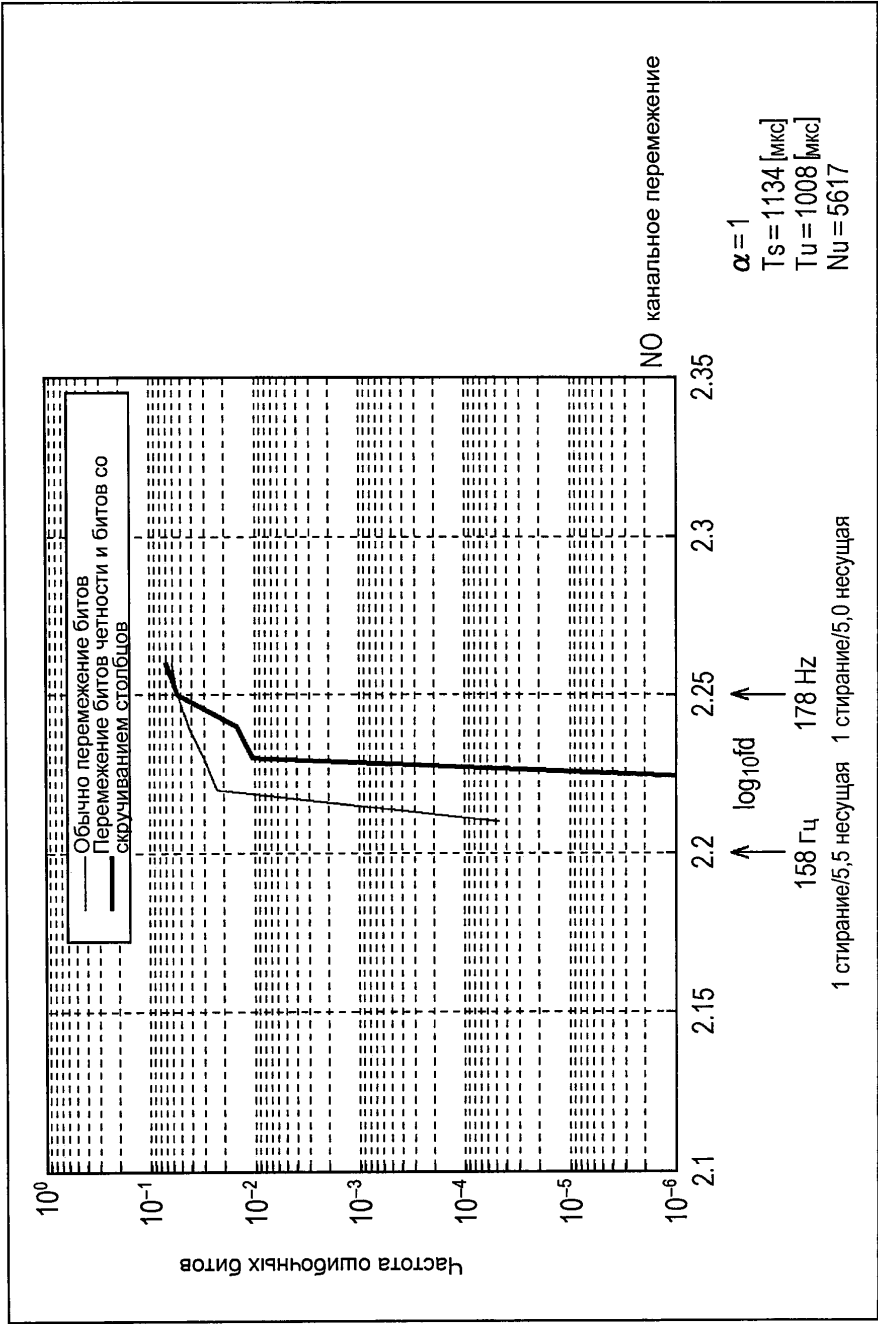
Фиг. 31



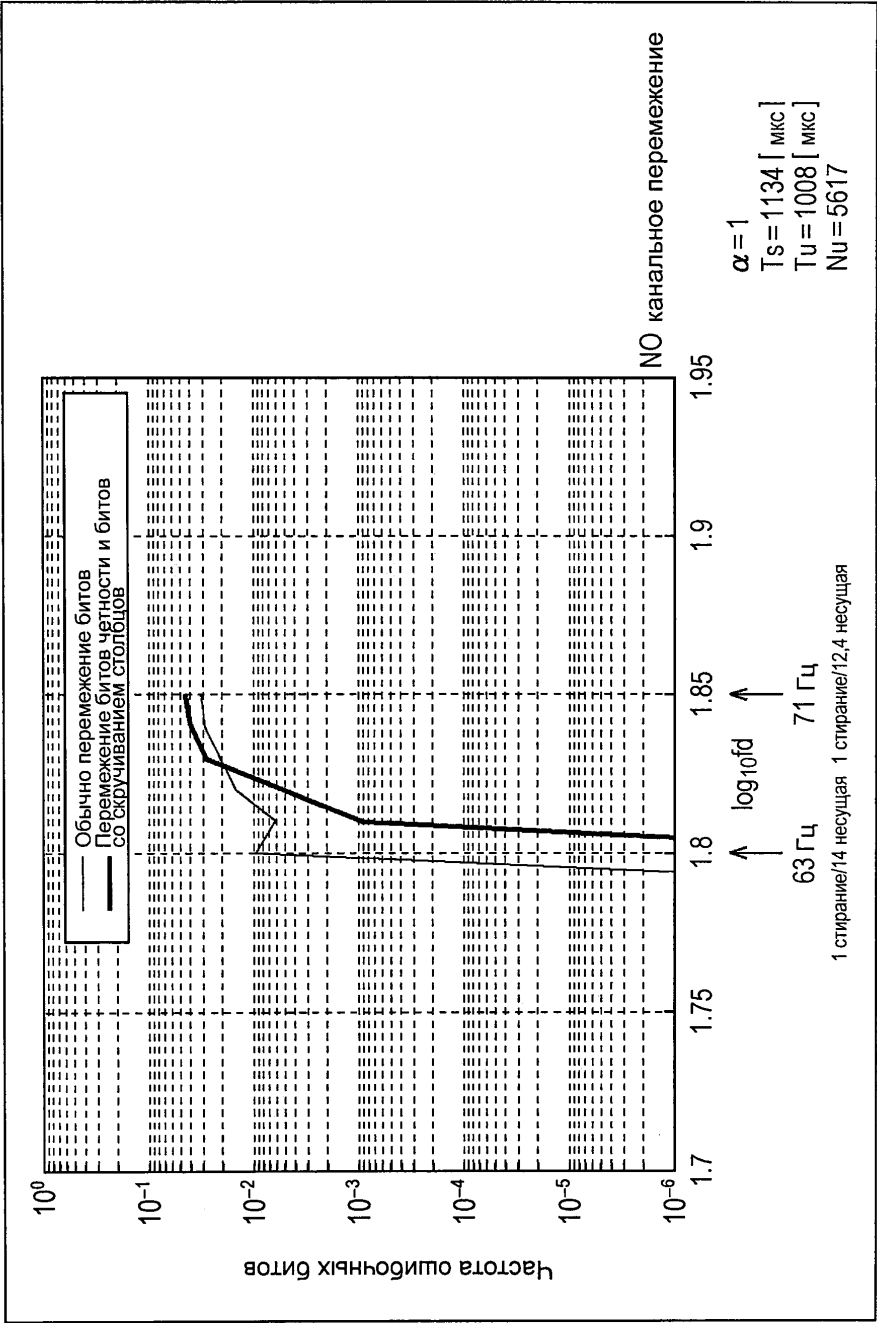
Фиг. 32



Фиг. 33

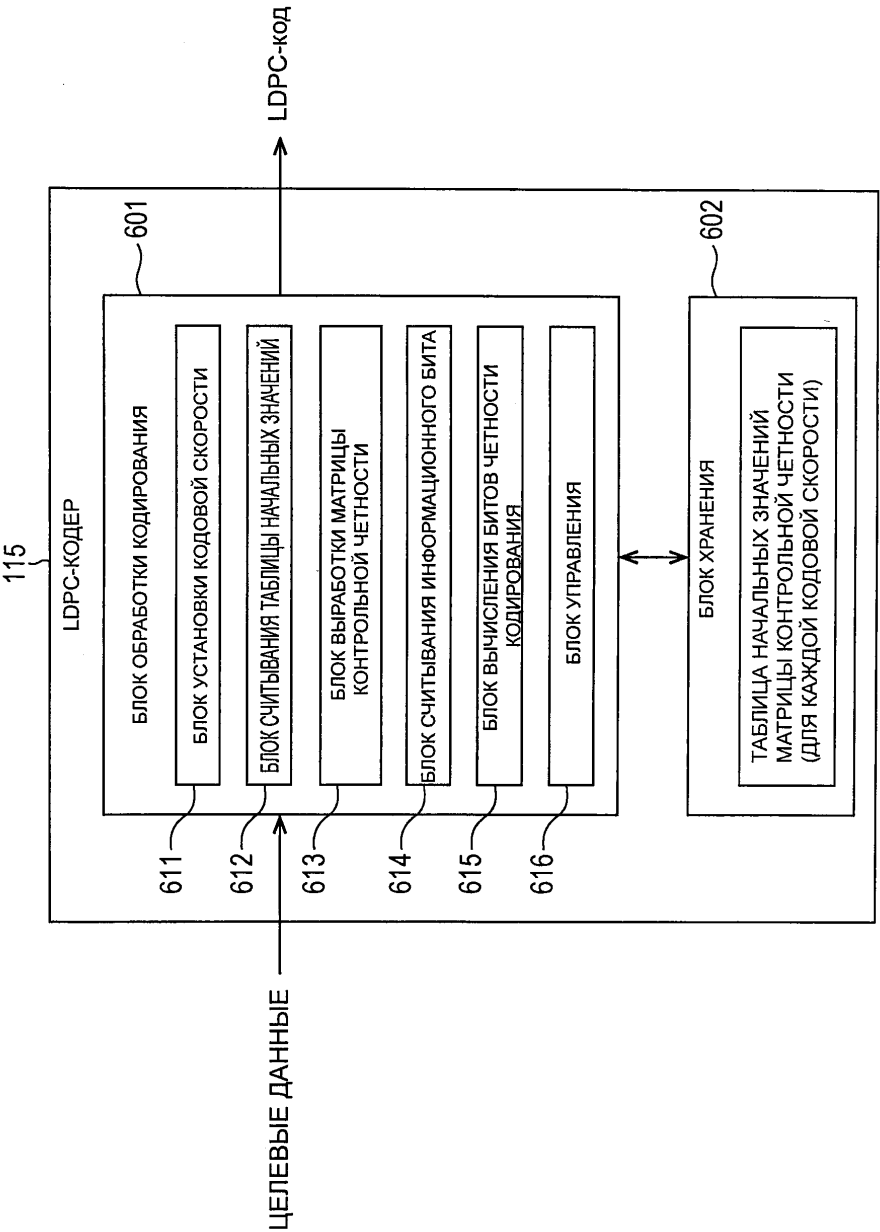


Фиг. 34



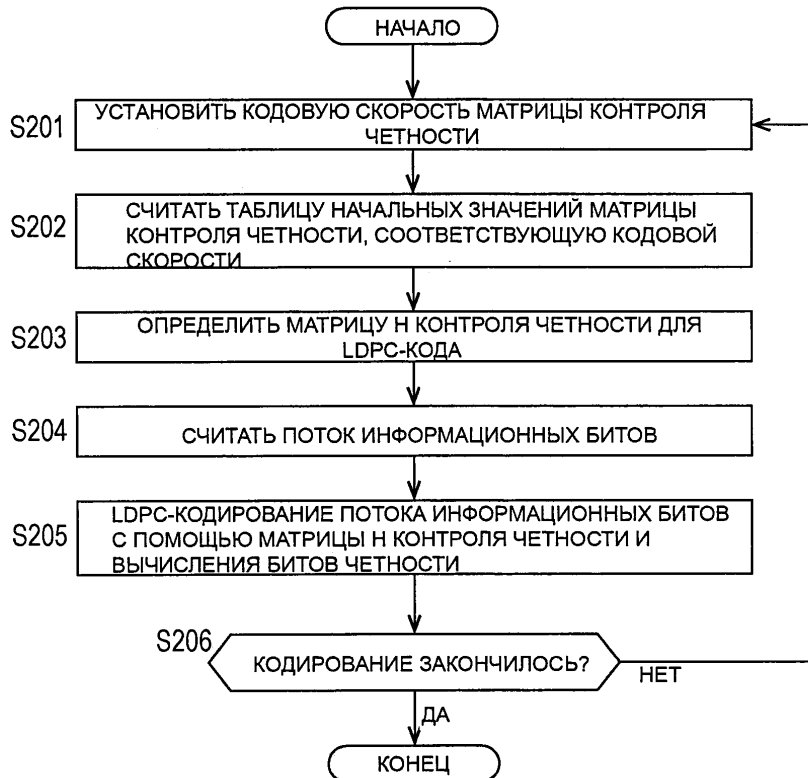
30/53

Фиг. 35



31 / 53

Фиг. 36

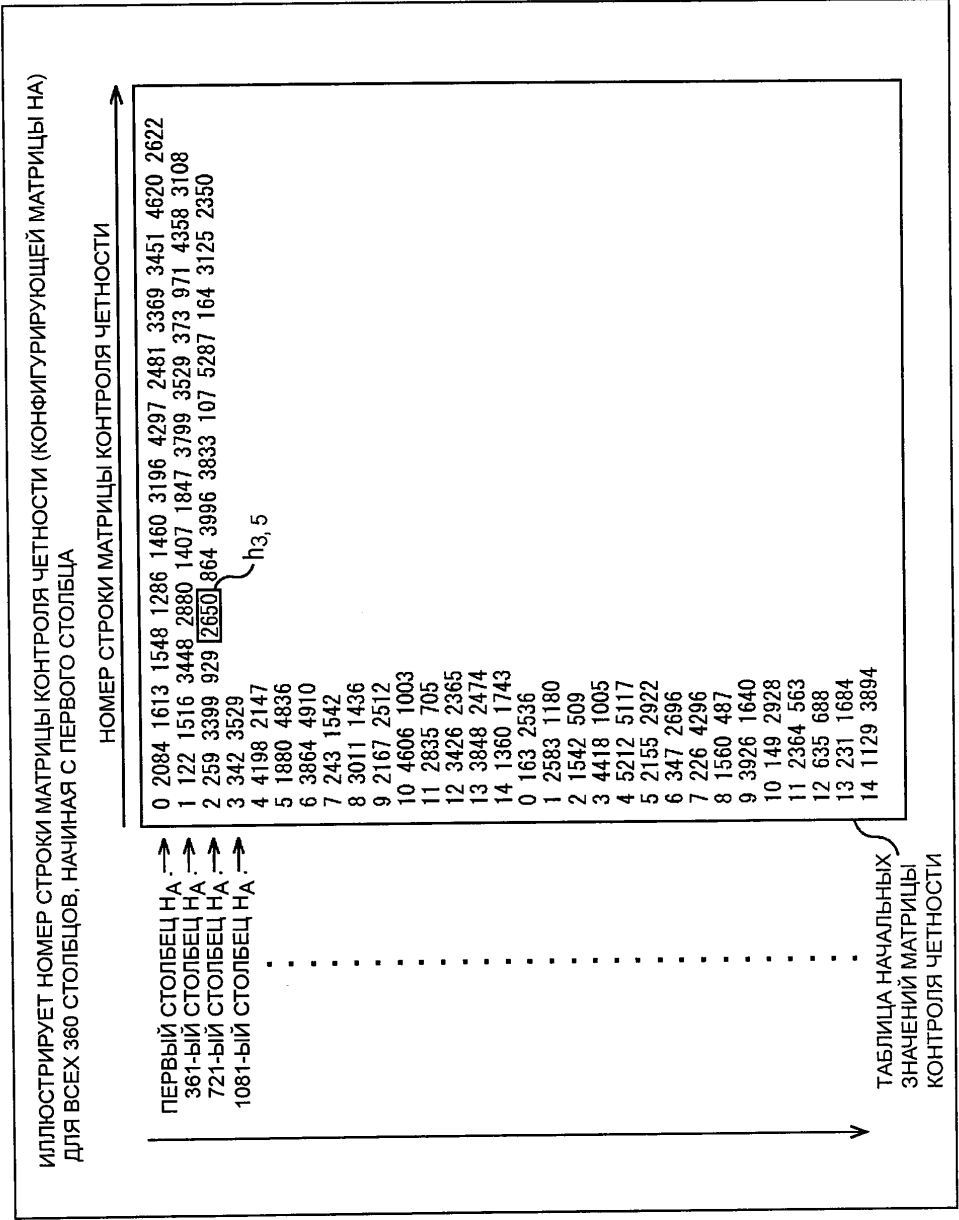


Фиг. 37

$r1/4\ 16k$

6295	9626	304	7695	4839	4936	1660	144	11203	5567	6347	12557
10691	4988	3859	3734	3071	3494	7687	10313	5964	8069	8296	11090
10774	3613	5208	11177	7676	3549	8746	6583	7239	12265	2674	4292
11869	3708	5981	8718	4908	10650	6805	3334	2627	10461	9285	11120
7844	3079	10773									
3385	10854	5747									
1360	12010	12202									
6189	4241	2343									
9840	12726	4977									

Фиг. 38



33/53

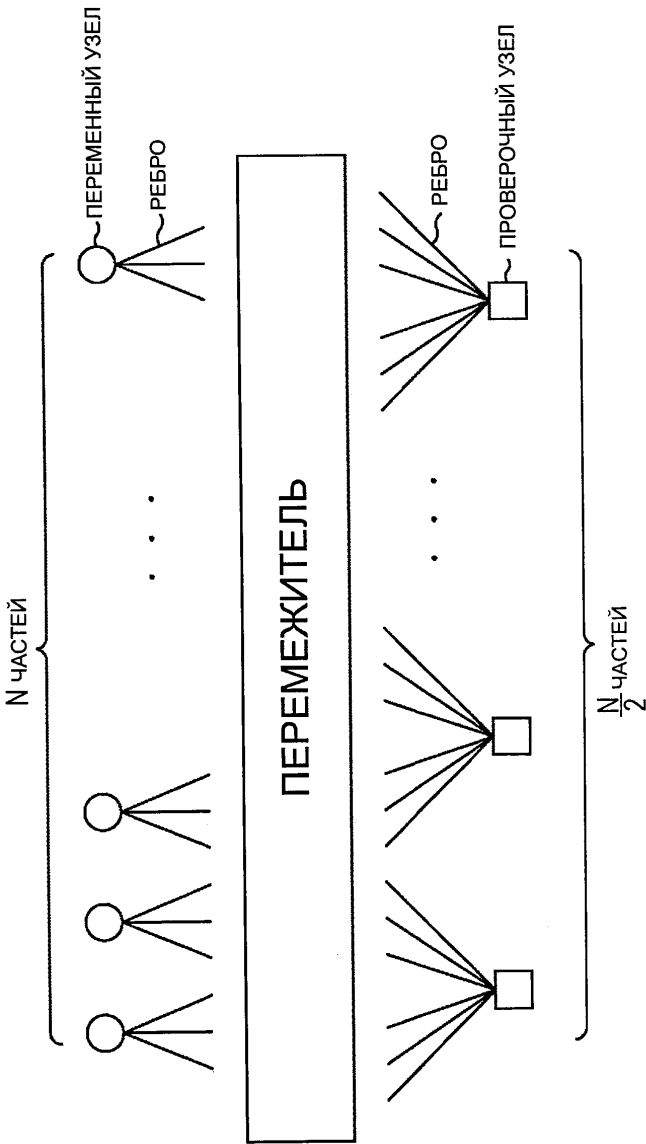
Фиг. 39

ТАБЛИЦА НАЧАЛЬНЫХ ЗНАЧЕНИЙ МАТРИЦЫ КОНТРОЛЯ ЧЕТНОСТИ

N=16200 и r=12/15

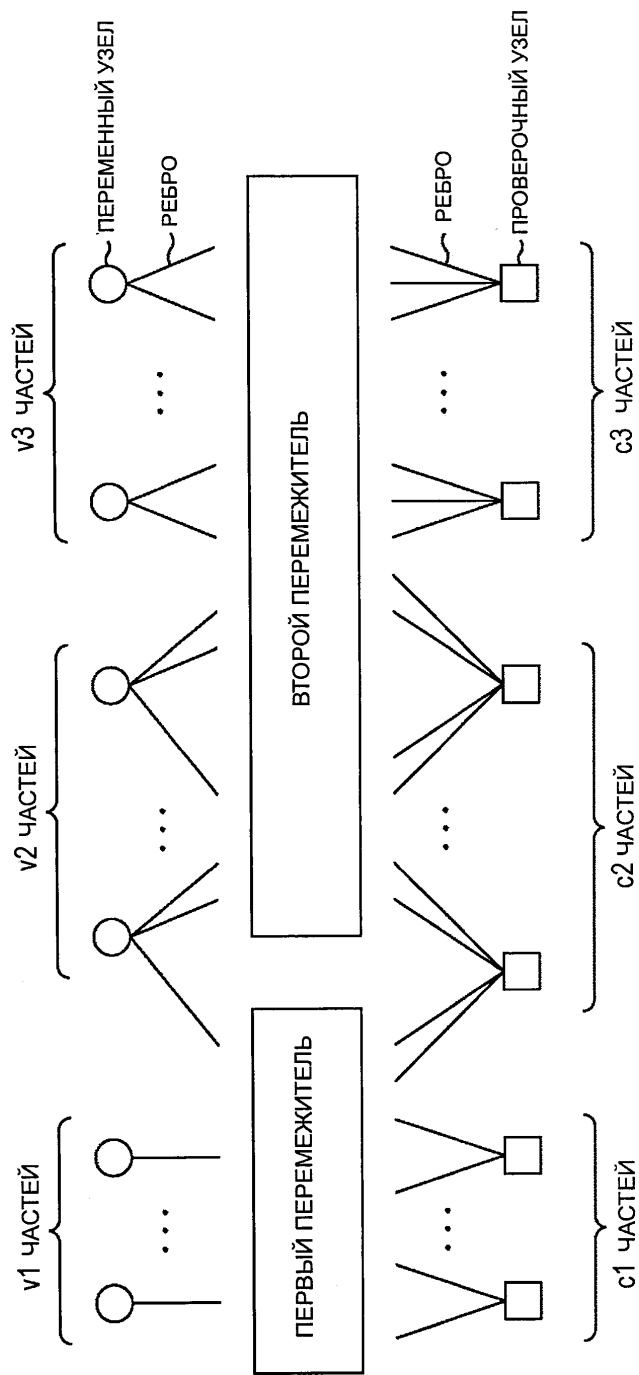
3	394	1014	1214	1361	1477	1534	1660	1856	2745	2987	2991	3124	3155
59	136	528	781	803	928	1293	1489	1944	2041	2200	2613	2690	2847
155	245	311	621	1114	1269	1281	1783	1995	2047	2672	2803	2885	3014
79	870	974	1326	1449	1531	2077	2317	2467	2627	2811	3083	3101	3132
4	582	660	902	1048	1482	1697	1744	1928	2628	2699	2728	3045	3104
175	395	429	1027	1061	1068	1154	1168	1175	2147	2359	2376	2613	2682
1388	2241	3118	3148										
143	506	2067	3148										
1594	2217	2705											
398	988	2551											
1149	2588	2654											
678	2844	3115											
1508	1547	1954											
1199	1267	1710											
2589	3163	3207											
1	2583	2974											
2766	2897	3166											
929	1823	2742											
1113	3007	3239											
1753	2478	3127											
0	509	1811											
1672	2646	2984											
965	1462	3230											
3	1077	2917											
1183	1316	1662											
968	1593	3239											
64	1996	2226											
1442	2058	3181											
513	973	1058											
1263	3185	3229											
681	1394	3017											
419	2853	3217											
3	2404	3175											
2417	2792	2854											
1879	2940	3235											
647	1704	3060											

Фиг. 40



35/53

Фиг. 41

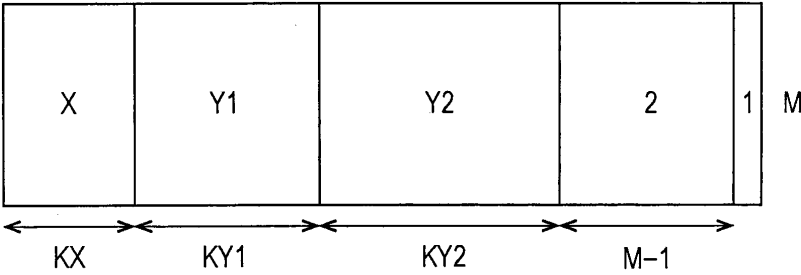


36 / 53

Фиг. 42

КОДОВАЯ СКОРОСТЬ	МИНИМАЛЬНАЯ ДЛИНА ЦИКЛА	ПОРОГ ПРОИЗВОДИТЕЛЬНОСТИ Eb/No
12/15	6	4,269922

Фиг. 43



37/53

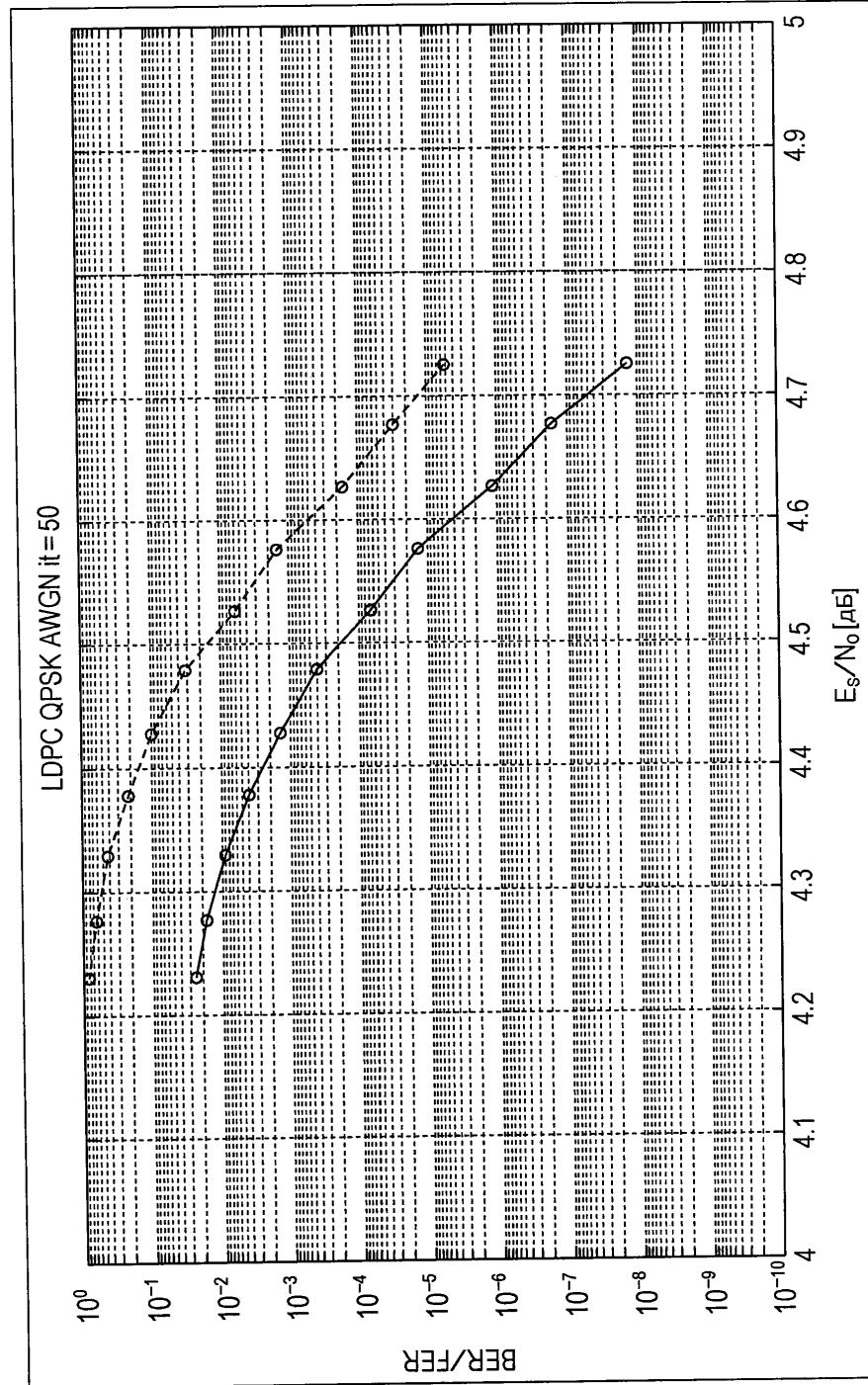
Фиг. 44

Скорость	X	KX	Y1	KY1	Y2	KY2	M
12/15	14	2160	4	720	3	10080	3240

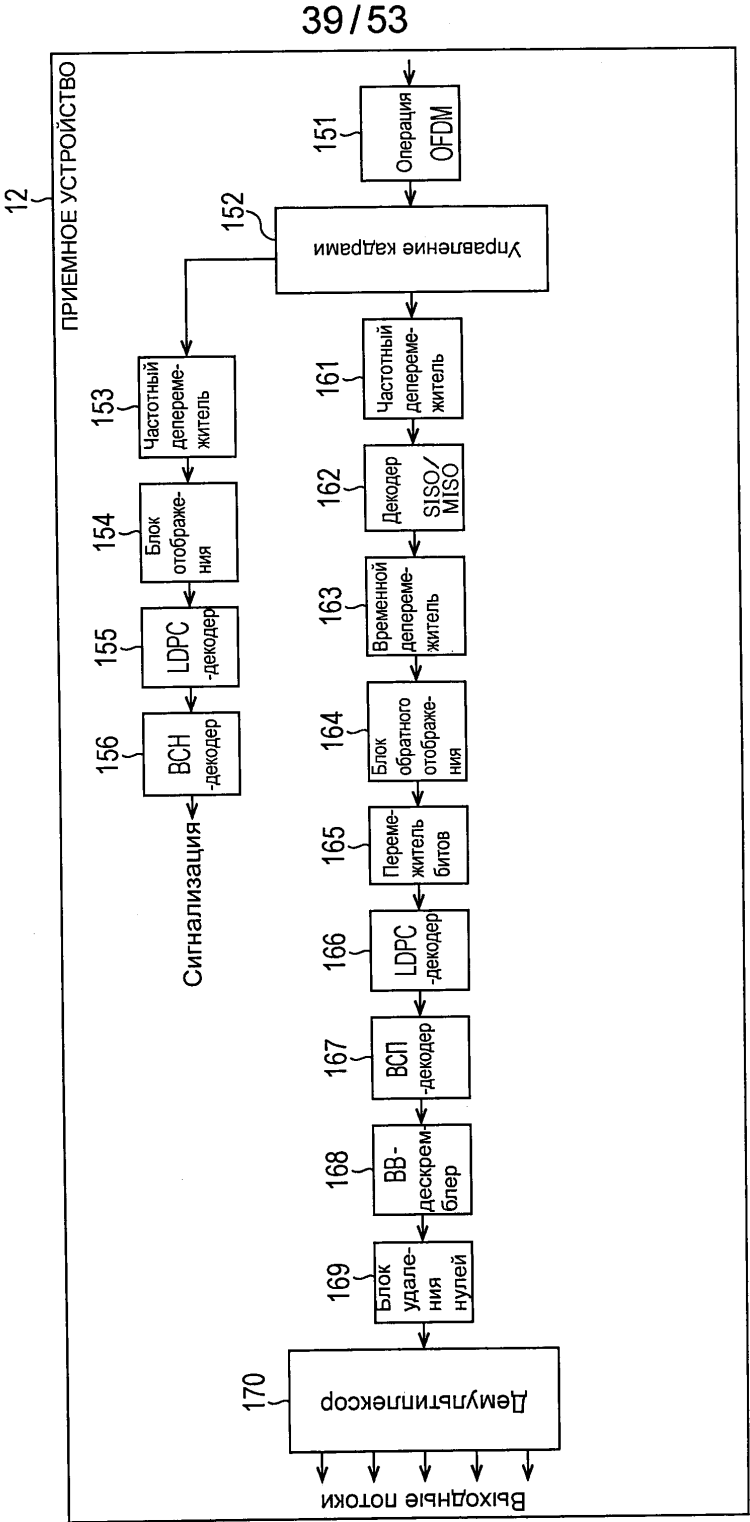
38/53

Фиг. 45

LDPQ QPSK AWGN it=50

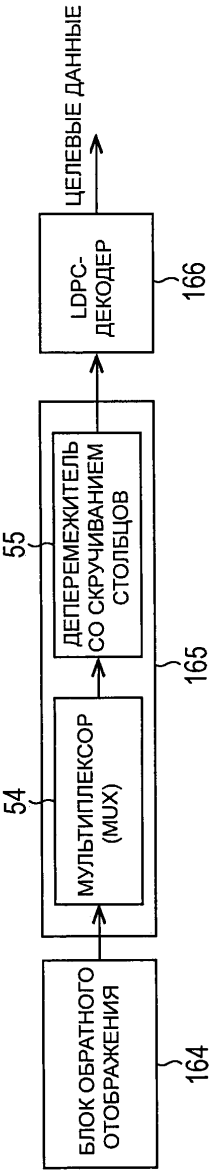


Фиг. 46



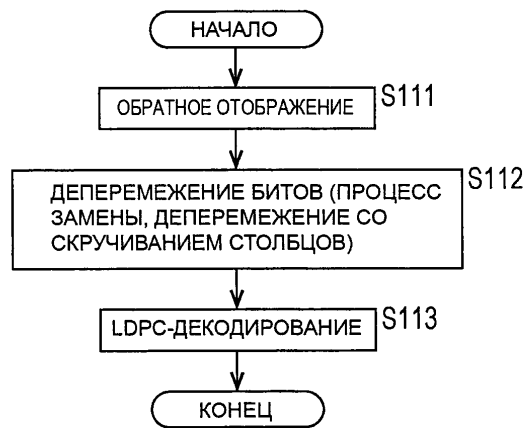
40/53

Фиг. 47



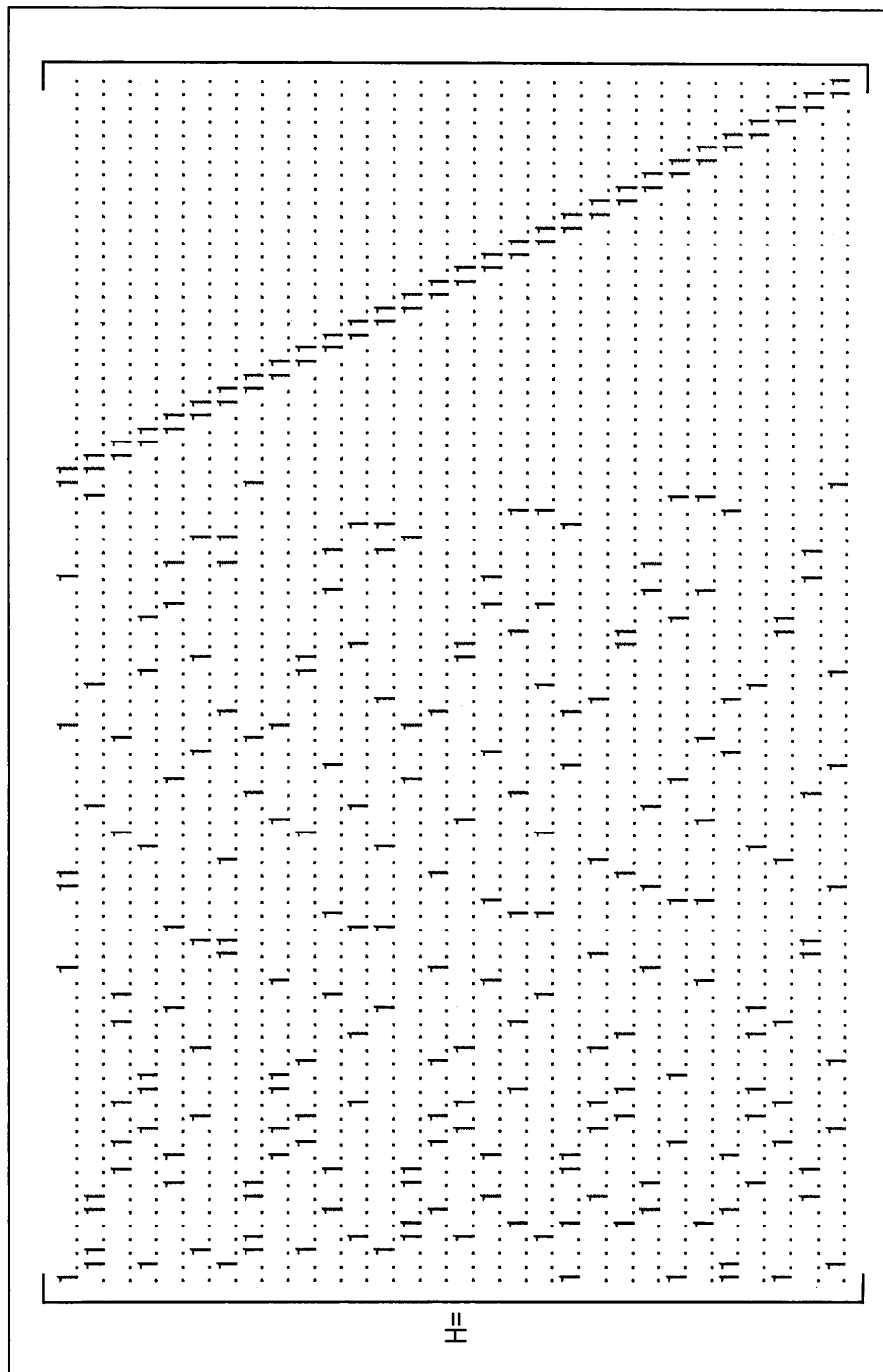
41 / 53

Фиг. 48



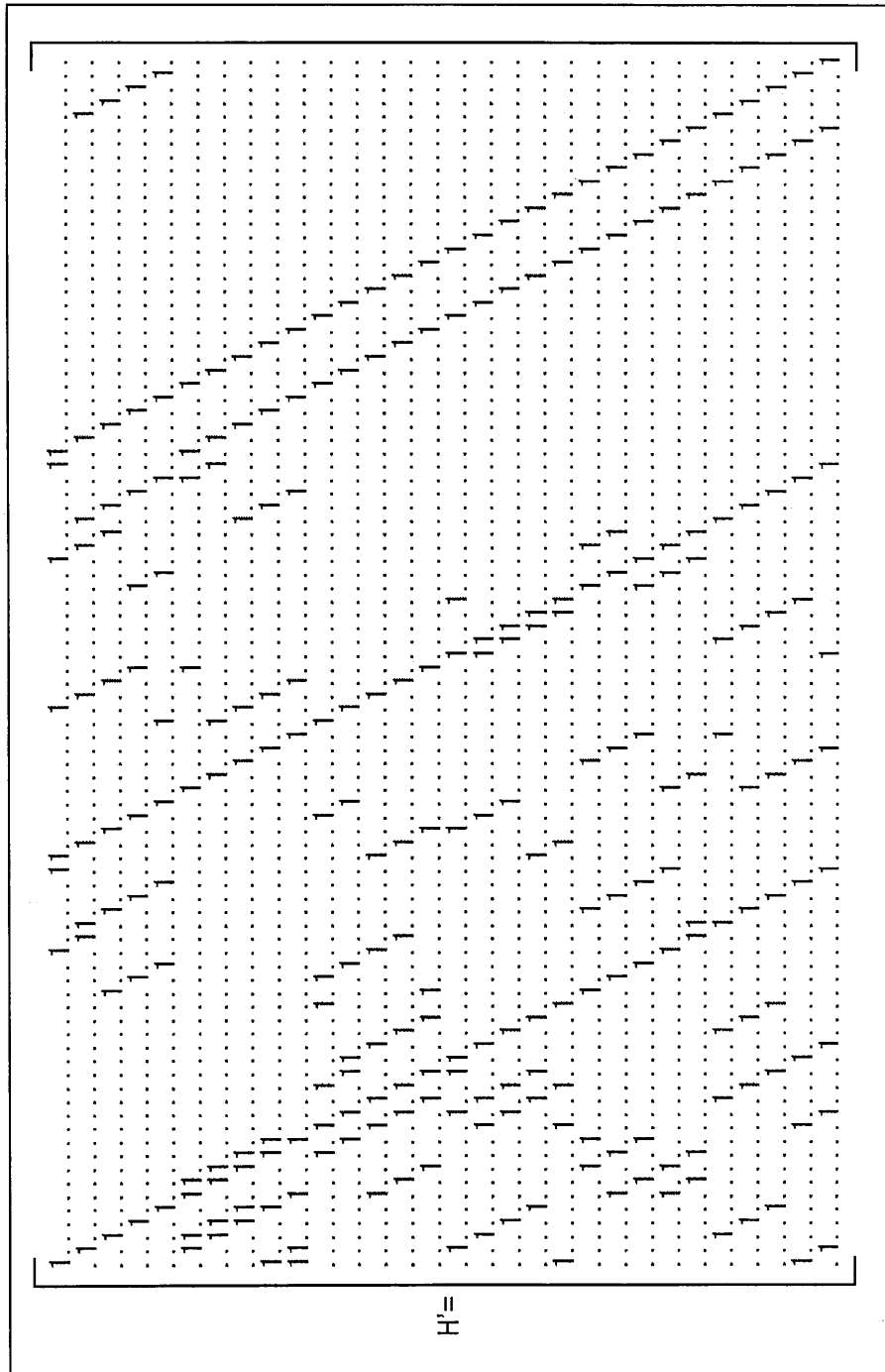
42/53

Фиг. 49



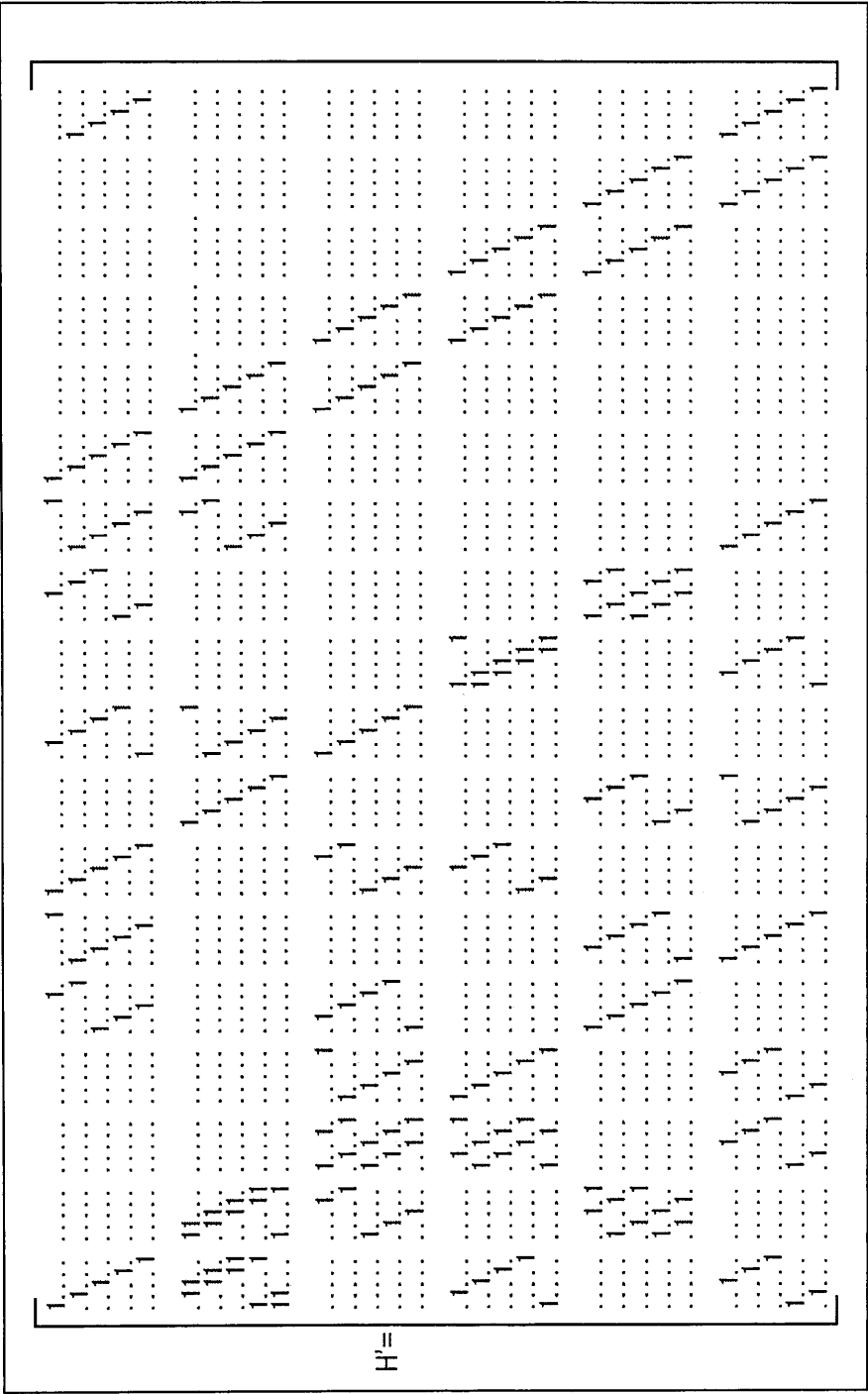
43/53

Фиг. 50

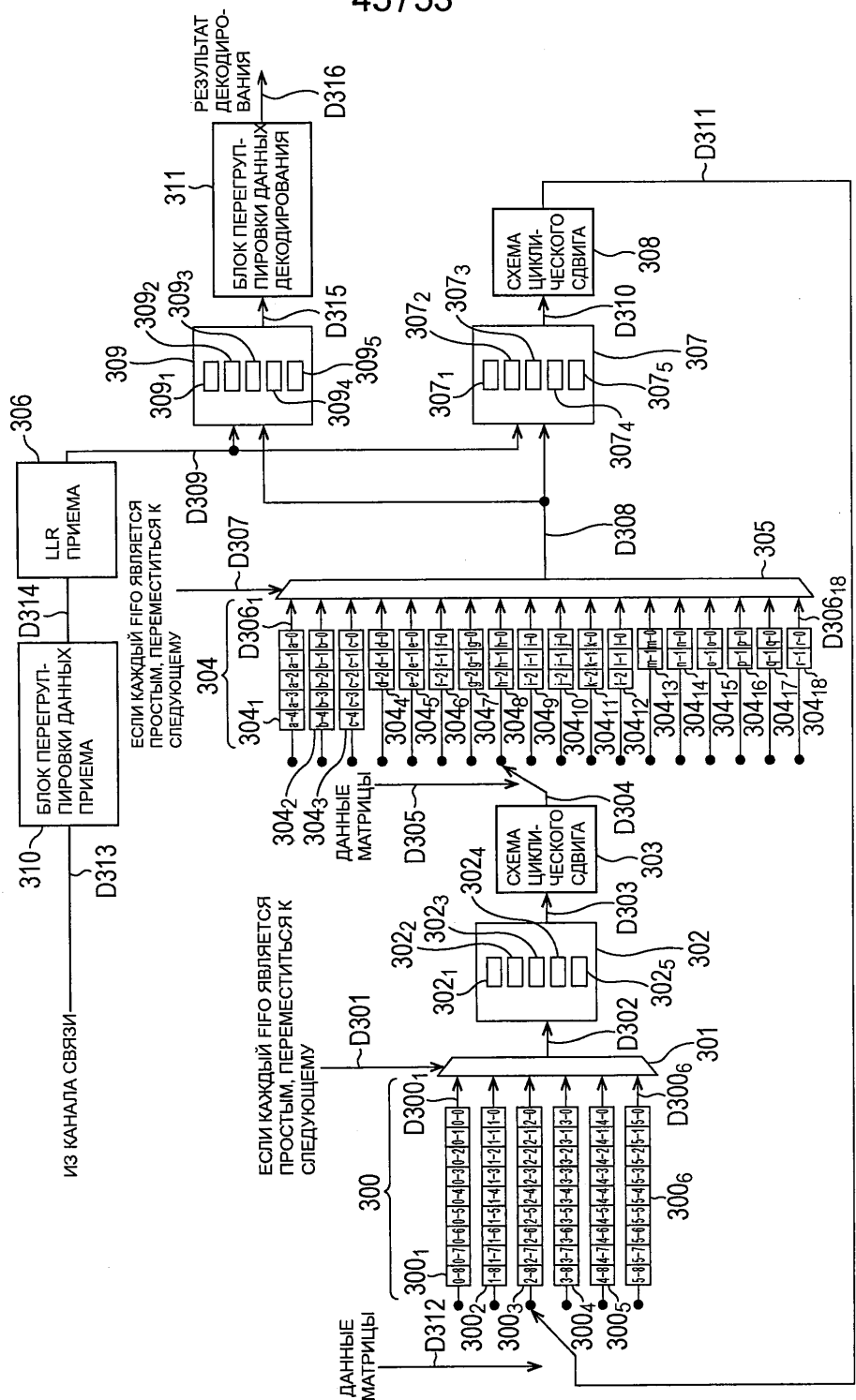


44/53

Фиг. 51

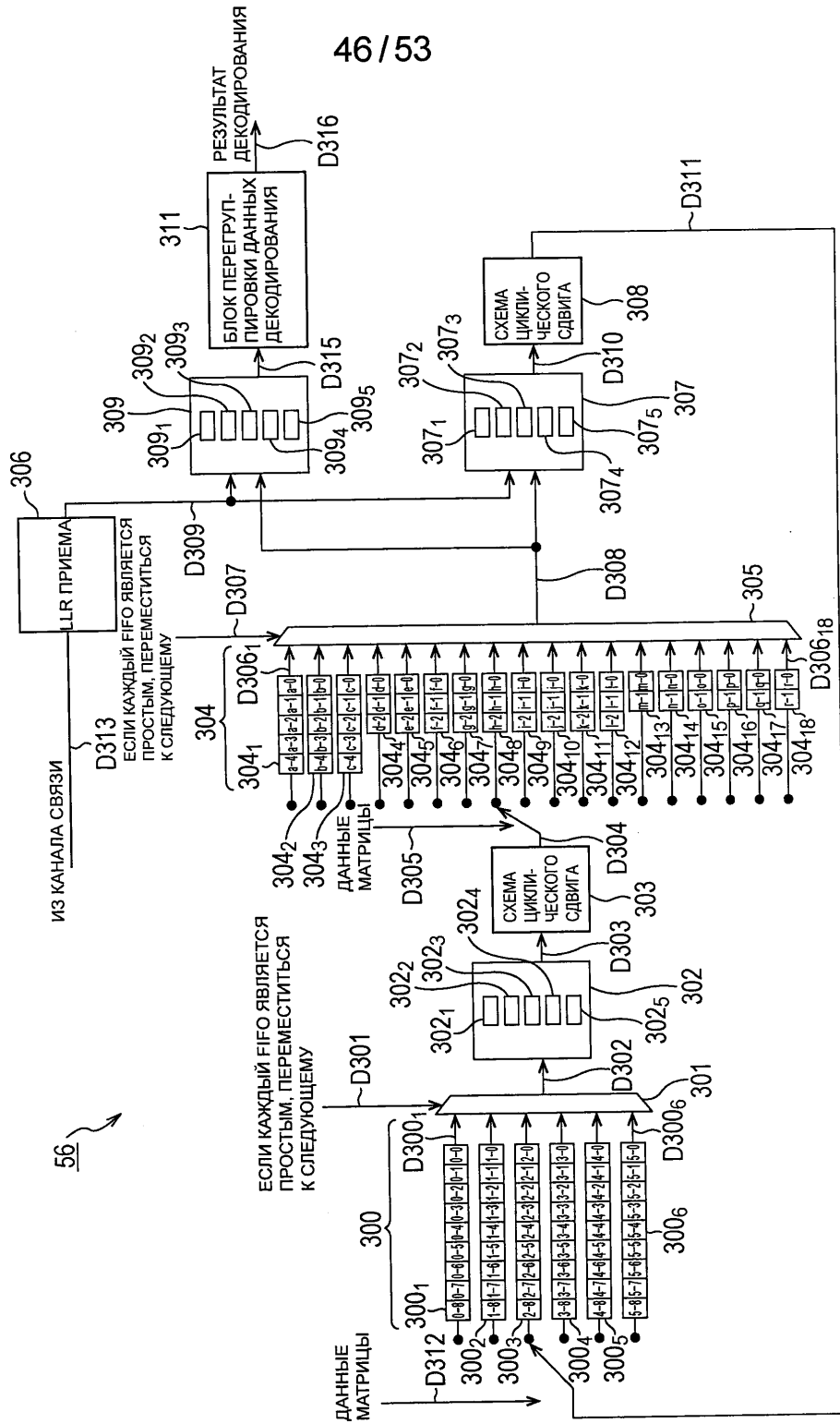


Фиг. 52



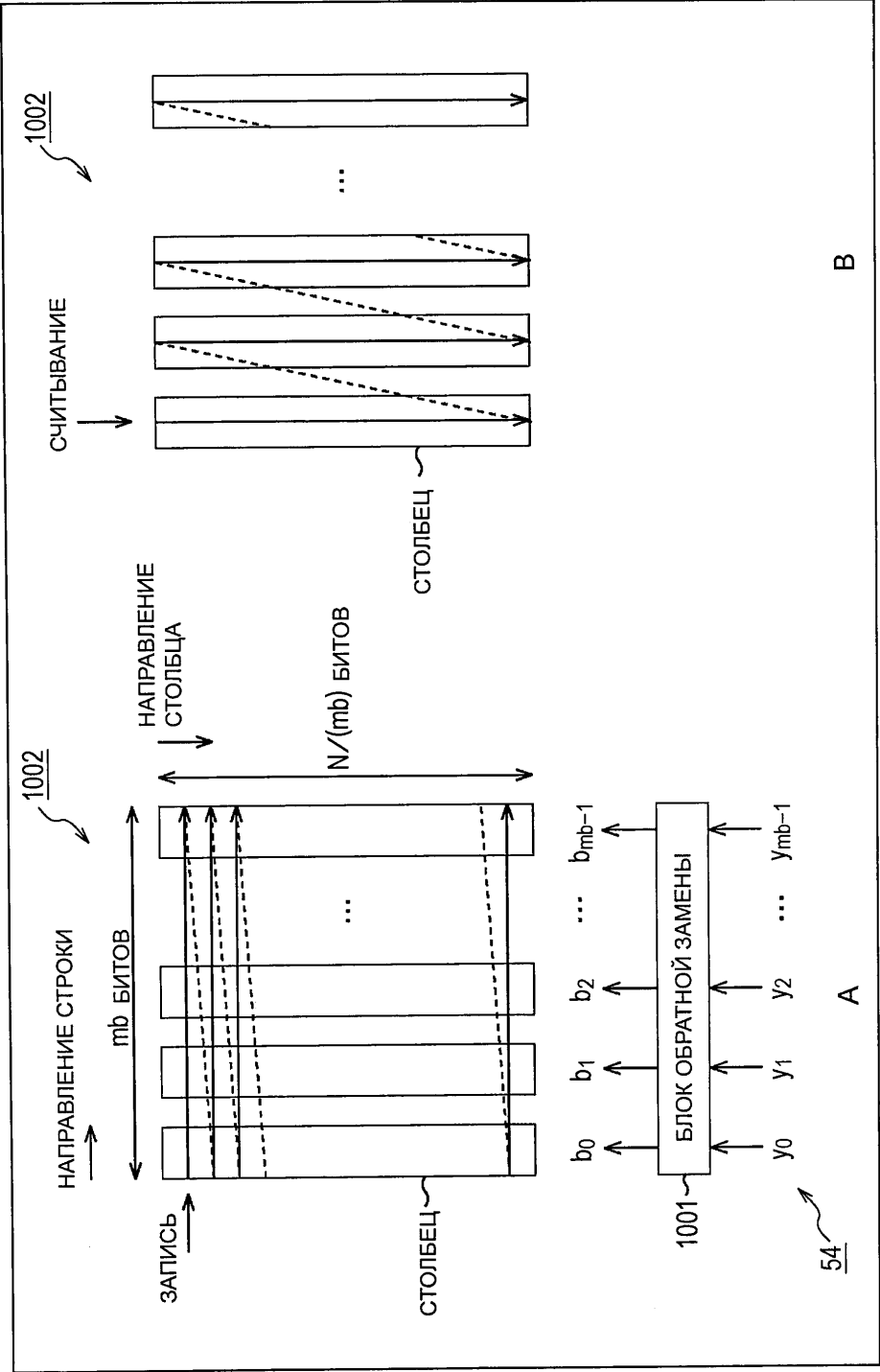
46/53

Фиг. 53



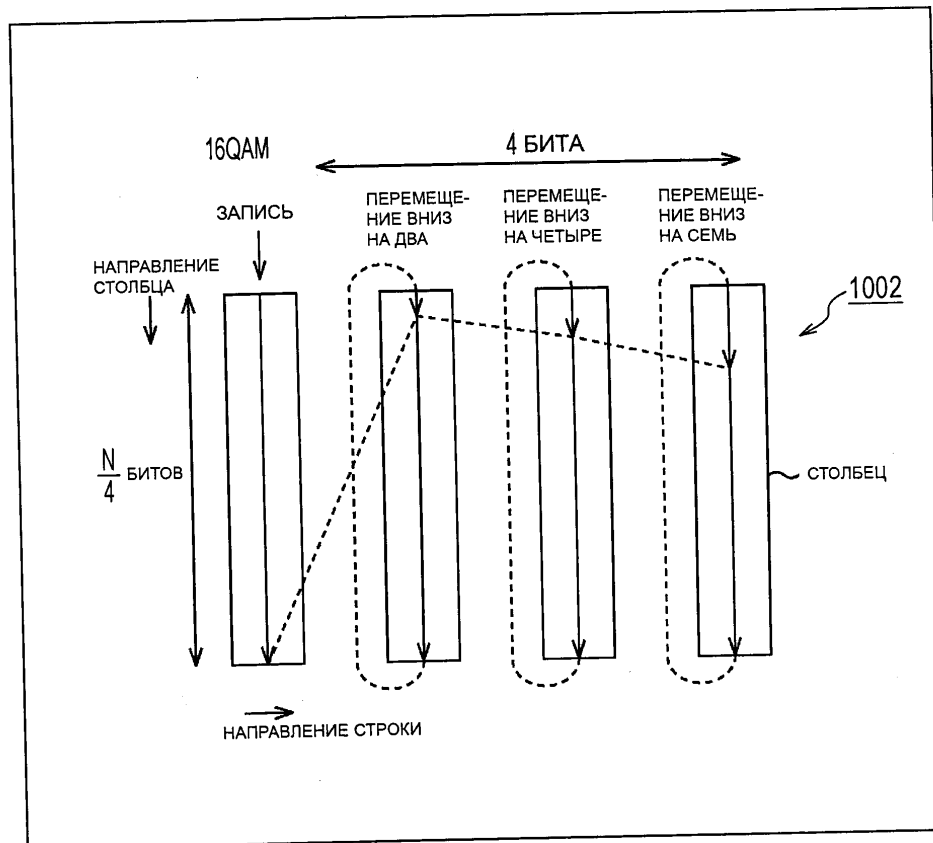
47/53

Фиг. 54



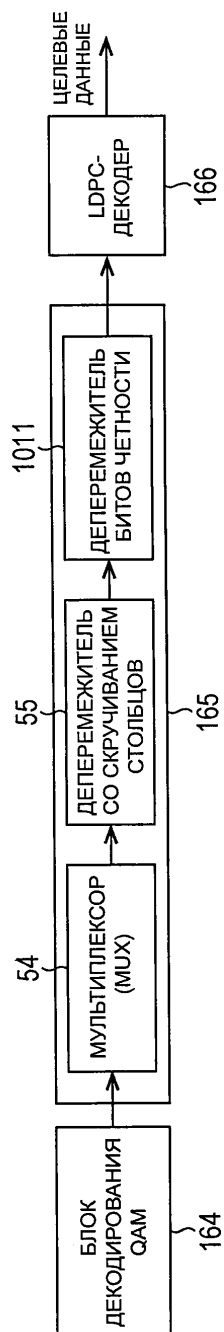
48/53

Фиг. 55



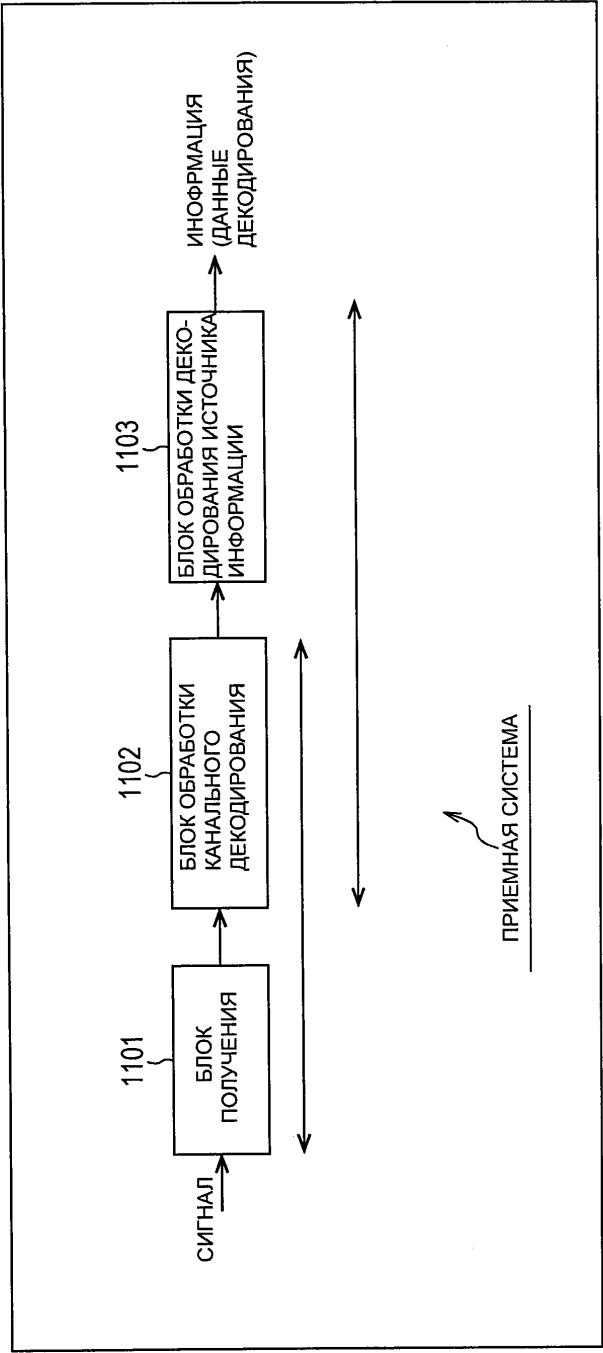
49/53

Фиг. 56



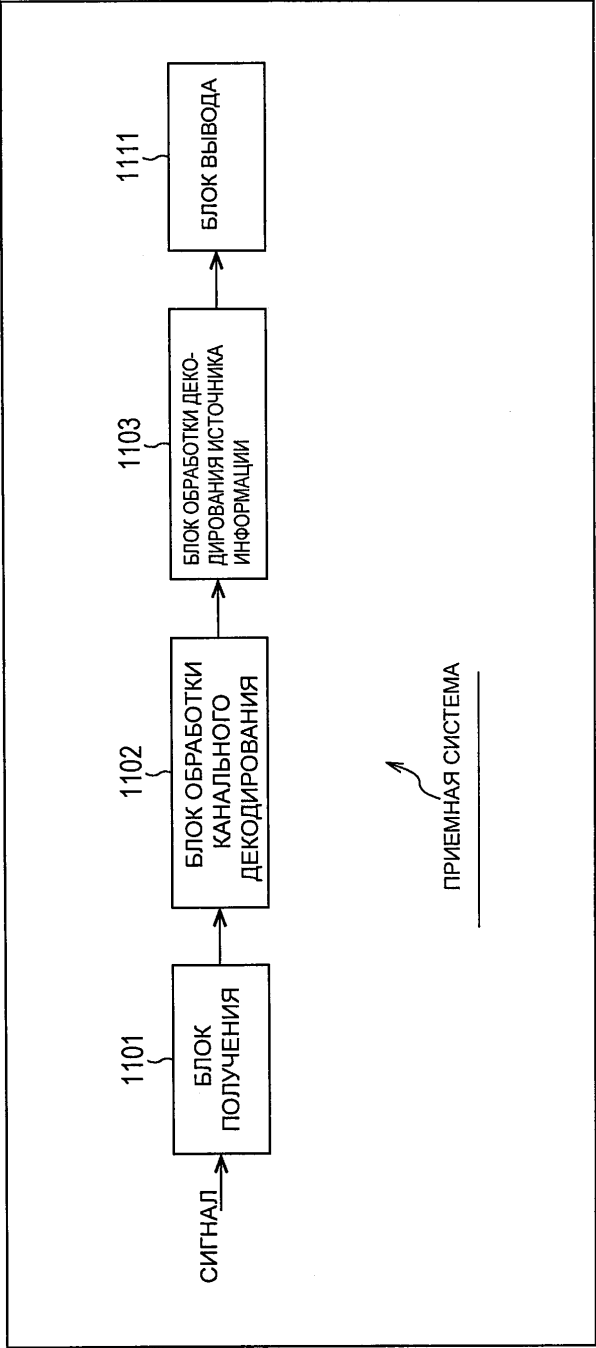
50/53

Фиг. 57

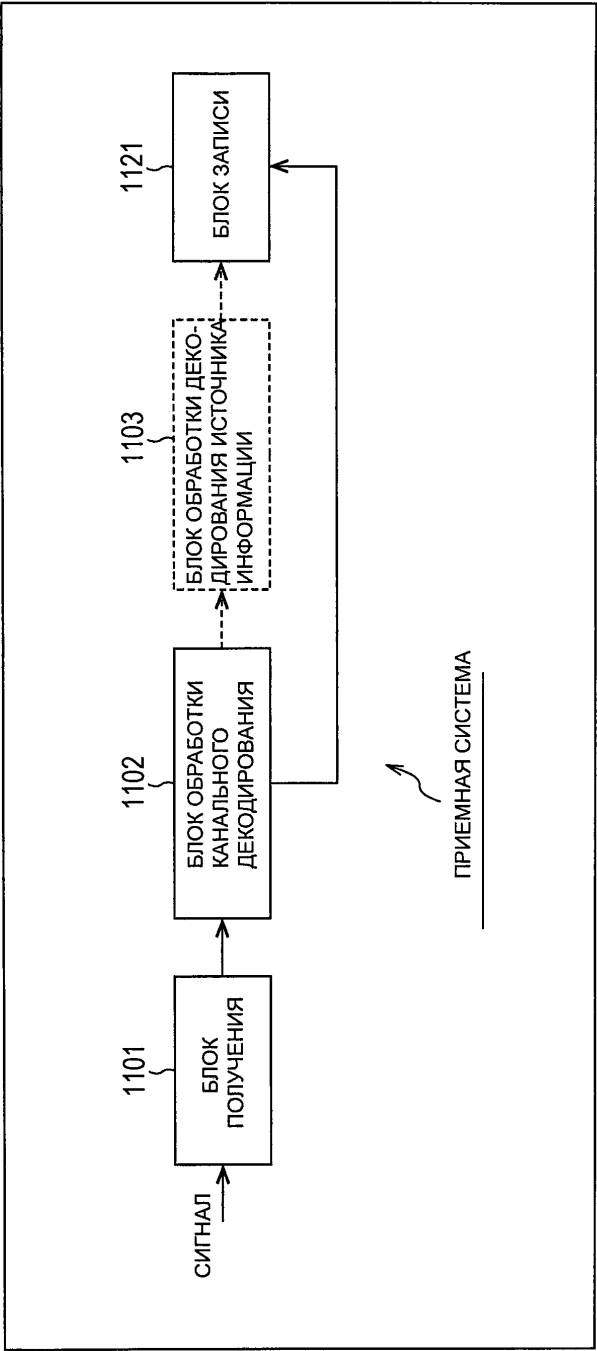


51/53

Фиг. 58



Фиг. 59



53/53

Фиг. 60

