



ФЕДЕРАЛЬНАЯ СЛУЖБА  
ПО ИНТЕЛЛЕКТУАЛЬНОЙ СОБСТВЕННОСТИ

(12) ОПИСАНИЕ ИЗОБРЕТЕНИЯ К ПАТЕНТУ

(52) СПК

G06T 9/00 (2020.01); H04N 19/124 (2020.01)

(21)(22) Заявка: 2017102010, 12.01.2012

(24) Дата начала отсчета срока действия патента:  
12.01.2012

Дата регистрации:  
17.04.2020

Приоритет(ы):

(30) Конвенционный приоритет:

24.01.2011 JP 2011-011861;

11.07.2011 JP 2011-153183

Номер и дата приоритета первоначальной заявки,  
из которой данная заявка выделена:

2013132921 24.01.2011

(45) Опубликовано: 17.04.2020 Бюл. № 11

Адрес для переписки:

101000, Москва, ул. Мясницкая, д. 13, стр. 5,  
ООО "Союзпатент"

(72) Автор(ы):

КОНДО Кэндзи (JP)

(73) Патентообладатель(и):

СОНИ КОРПОРЕЙШН (JP)

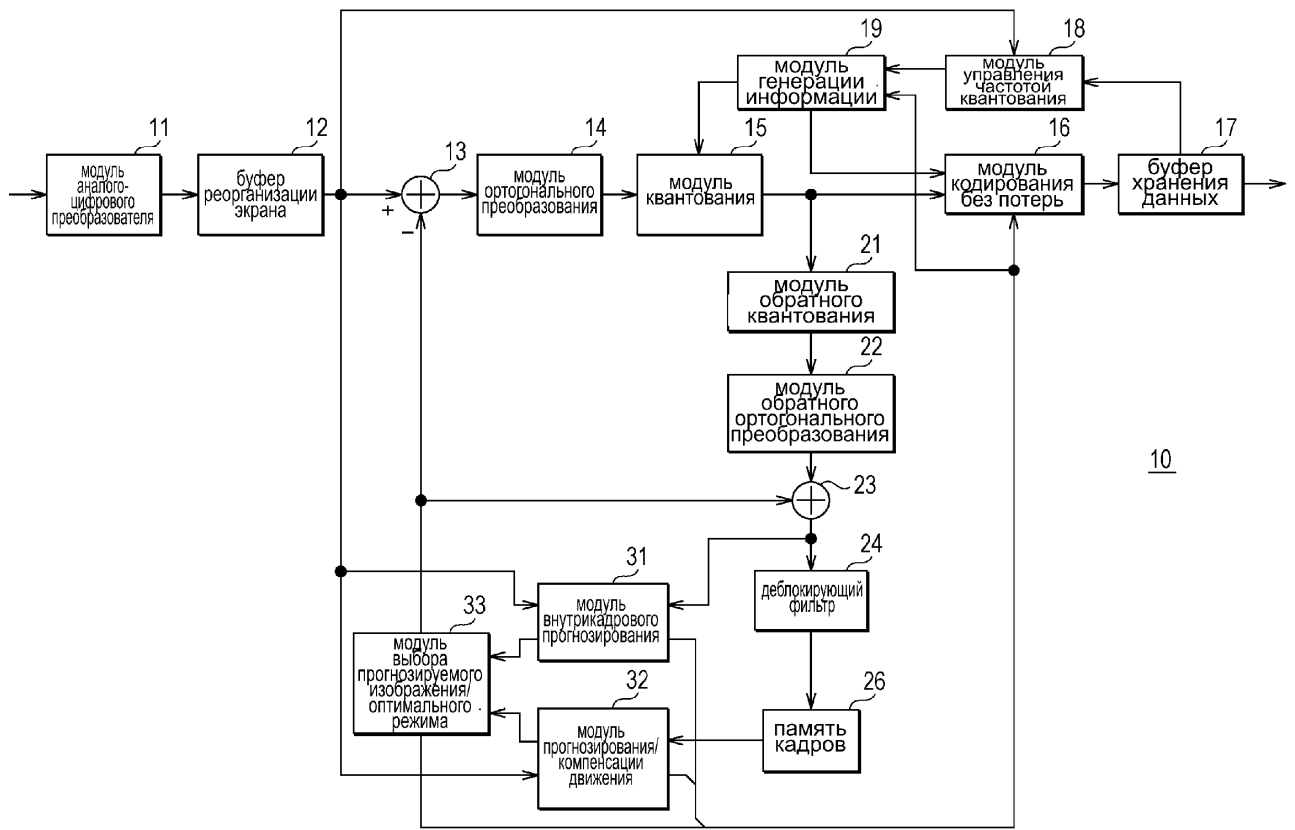
(56) Список документов, цитированных в отчете  
о поиске: WO 2009/158113 A2, 30.12.2009. US  
2005/0053151 A1, 10.03.2005. EP 2141927 A1,  
06.01.2010. CN 101232618 A, 30.07.2008. KR  
1020100074276 A, 01.07.2010. RU 2408161 C1,  
27.12.2010.

(54) СПОСОБ КОДИРОВАНИЯ ИЗОБРАЖЕНИЯ И УСТРОЙСТВО ОБРАБОТКИ ИНФОРМАЦИИ

(57) Реферат:

Изобретение относится к области вычислительной техники. Технический результат заключается в повышении точности кодирования параметров квантования. Технический результат достигается за счет того, что устанавливают, когда блок, соседствующий слева с текущим блоком, и блок, соседствующий сверху с текущим блоком, являются доступными, прогнозный параметр квантования, выводимый из среднего значения параметра квантования для блока, соседствующего слева, и параметра квантования для блока, соседствующего сверху;

устанавливают, когда блок, соседствующий слева с текущим блоком, и блок, соседствующий сверху с текущим блоком, являются недоступными, прогнозный параметр квантования, выводимый из параметра квантования модуля обработки, не соседствующего с текущим блоком; генерируют разностную информацию, указывающую разность между установленным прогнозным параметром квантования и параметром квантования текущего блока; и генерируют потоковую информацию, включающую в себя разностную информацию. 2 н. и 10 з.п. ф-лы, 33 ил.



Фиг. 3



FEDERAL SERVICE  
FOR INTELLECTUAL PROPERTY

(12) **ABSTRACT OF INVENTION**

(52) CPC

*G06T 9/00 (2020.01); H04N 19/124 (2020.01)*(21)(22) Application: **2017102010, 12.01.2012**(24) Effective date for property rights:  
**12.01.2012**Registration date:  
**17.04.2020**

Priority:

(30) Convention priority:  
**24.01.2011 JP 2011-011861;**  
**11.07.2011 JP 2011-153183**Number and date of priority of the initial application,  
from which the given application is allocated:  
**2013132921 24.01.2011**(45) Date of publication: **17.04.2020 Bull. № 11**Mail address:  
**101000, Moskva, ul. Myasnitskaya, d. 13, str. 5,**  
**OOO "Soyuzpatent"**

(72) Inventor(s):

**KONDO Kendzi (JP)**

(73) Proprietor(s):

**SONI KORPOREJSHN (JP)**(54) **IMAGE ENCODING METHOD AND INFORMATION PROCESSING DEVICE**

(57) Abstract:

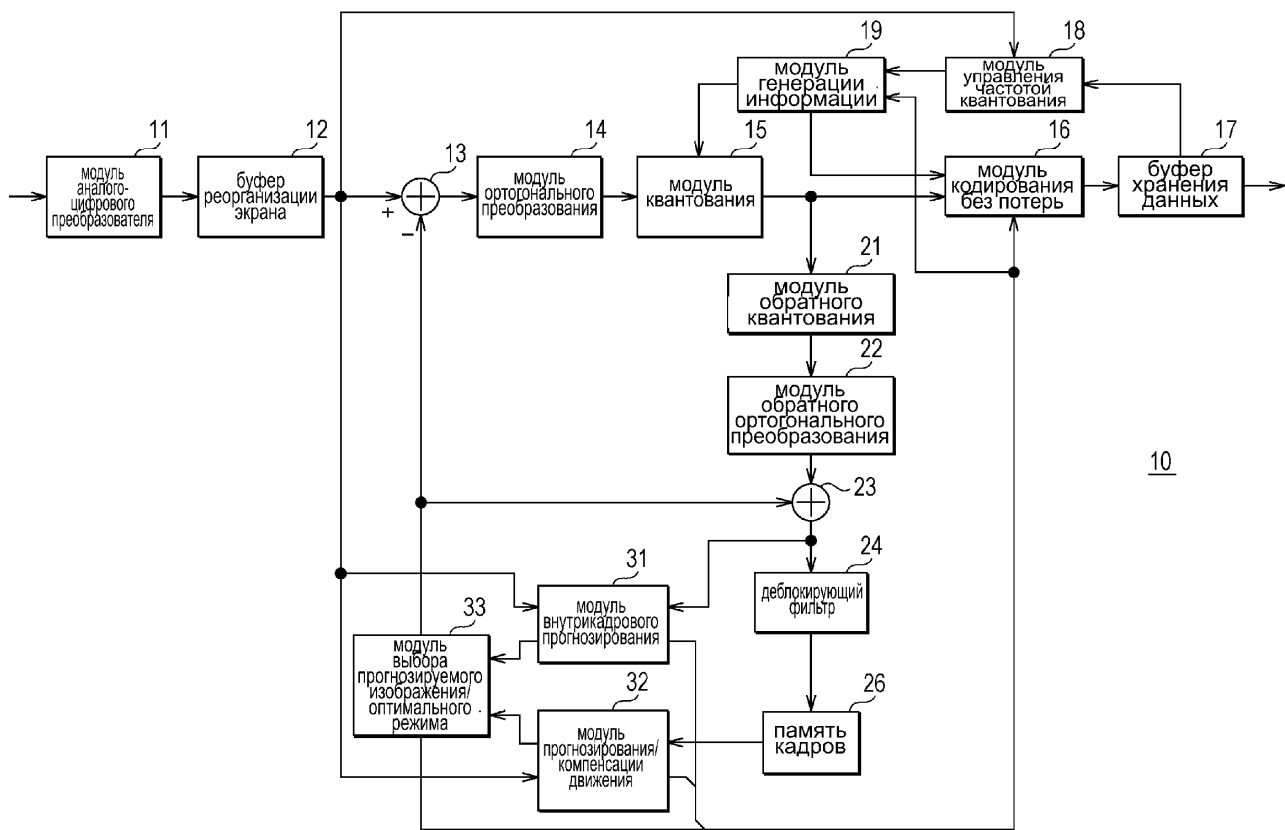
FIELD: computer equipment.

SUBSTANCE: technical result is achieved by the fact that when the unit adjacent to the left with the current unit and the unit adjoining from the top with the current unit are installed, a predictive quantisation parameter output from the average quantisation parameter value for the left-neighboring unit and the quantisation parameter for the neighboring unit from above; setting, when the block adjoining on the left with the current unit, and the block adjoining from the top with the current block, are inaccessible, a predictive

quantisation parameter output from the processing unit quantisation parameter not adjacent to the current block; generating difference information indicating a difference between the predetermined prediction quantization parameter and the current unit quantization parameter; and generating stream information including differential information.

EFFECT: technical result consists in improvement of quantization parameters encoding accuracy.

12 cl, 33 dwg



Фиг. 3

Область техники, к которой относится изобретение

Предлагаемая технология относится к устройству декодирования изображения, устройству кодирования изображения и способу работы устройств. Более конкретно, технология позволяет повысить эффективность кодирования параметров квантования.

Уровень техники

В последние годы стали широко применяться устройства для обработки информации изображений в цифровой форме с целью осуществления высокоэффективной передачи и хранения информации, например, в соответствии с форматами, такими как форматы группы MPEG или аналогичные форматы, с целью сжатия изображения посредством ортогонального преобразования, такого как дискретное косинусное преобразование или аналогичное преобразование, и компенсации движения, как в системах вещания, так и в бытовой аппаратуре в целом.

В частности, формат MPEG2 (ISO/IEC 13818-2) определен в качестве формата кодирования изображения общего назначения и широко применяется вплоть до сегодняшнего дня самыми разнообразными приложениями, как в профессиональной сфере, так и в потребительской сфере. При использовании формата MPEG2 сжатия назначают объем кода (частота передачи данных) от 4 до 8 Мбит/с для передачи изображения с чересстрочной разверткой и стандартным разрешением 720 × 480 пикселей, например, что позволяет реализовать высокую степень сжатия и хорошее качество изображения. Кроме того, назначают объем кода (частота передачи данных) от 18 до 22 Мбит/с для передачи изображения с чересстрочной разверткой и высоким разрешением 1920 × 1088 пикселей, что также позволяет реализовать высокую степень сжатия и хорошее качество изображения.

Кроме того, была выполнена стандартизация Совместной модели (Joint Model) Видео кодирования с повышенным коэффициентом сжатия (Enhanced-Compression Video Coding), которая реализует более высокую эффективность кодирования за счет увеличения объема вычислений, необходимых для кодирования и декодирования изображения, и стала международным стандартом, именуемым H.264 и MPEG-4 Part 10 (в дальнейшем будет называться "H.264/AVC (Усовершенствованное видео кодирование (Advanced Video Coding))").

При использовании таких стандартов MPEG и H.264/AVC можно при кодировании макроблоков изменять величину шага квантования таким образом, чтобы коэффициент сжатия оставался постоянным. Кроме того, в стандарте MPEG используют параметры квантования, пропорциональные величине шага квантования, а в стандарте H.264/AVC используют такие параметры квантования, что величина параметра возрастает на "6", когда шаг квантования удваивается. В стандартах MPEG и H.264/AVC параметры квантования кодируют (см. PTL 1).

Список литературы

Патентная литература

PTL 1: Публикация нерассмотренной Заявки на выдачу патента Японии No. 2006-094081

Раскрытие изобретения

Техническая проблема

Теперь, в процессе кодирующей обработки параметров квантования, если порядок декодирования совпадает с порядком растрового сканирования, как показано на Фиг. 1, например, начальное значение параметра квантования SliceQP<sub>Y</sub> используют для головного макроблока в срезе. В последующем обработку выполняют в порядке декодирования, обозначенном стрелками, а параметры квантования для этого

макроблока обновляют на величину разности (mb\_qp\_delta) относительно параметров квантования для макроблока, расположенного слева. Соответственно, возникают случаи, в которых, когда порядок квантования переходит от блока на правом краю к блоку на левом краю, величина разности становится большой, поскольку меняется изображение, и эффективность кодирования становится низкой. Кроме того, эффективность кодирования становится низкой, если величина разности относительно макроблока, расположенного слева, также велика.

Кроме того, в области технологии сжатия изображения проводятся исследования с целью стандартизации алгоритма HEVC (высокоэффективное видео кодирование (High Efficiency Video Coding)), который реализует даже более высокую эффективность, чем формат H.264/AVC. В рамках этого стандарта HEVC базовые единицы именуются единицами кодирования (CU: Coding Unit), что является расширением концепции макроблоков. Если каждый блок, показанный на Фиг. 2, является единицей кодирования, порядок декодирования представляет собой порядок блоков с номерами, последовательно увеличивающимися от "0". Если при таком подходе порядок декодирования не является порядком растрового сканирования, переход от блока "7" к блоку "8", например, или от блока "15" к блоку "16" может, как это можно представить, привести к большей величине разности, поскольку пространственное расстояние велико.

Соответственно, целью предлагаемой технологии является повышение эффективности кодирования параметров квантования.

#### Решение проблемы

Первым аспектом предлагаемой технологии является устройство декодирования изображения, включающее в себя: модуль получения информации, выполненный с возможностью получения параметров квантования декодированных блоков, соседствующих с блоком, подлежащим декодированию, в пространстве или во времени, в качестве кандидатов для выбора и извлечение из потоковой информации разностной информации, указывающей разность относительно прогнозного параметра квантования, выбранного из указанных кандидатов для выбора; и модуль вычисления параметра квантования, выполненный с возможностью вычисления на основе указанного прогнозного параметра квантования и разностной информации параметра квантования для блока, подлежащего декодированию.

Согласно предлагаемой технологии разностную информацию, указывающую разность относительно прогнозного параметра квантования, выбранного из кандидатов для выбора, представляющих собой параметры квантования декодированных блоков, соседствующих в пространстве или во времени с блоком, подлежащим декодированию, извлекают из потоковой информации. Кроме того, в таком устройстве декодирования изображения по меньшей мере блоки, в которых параметры квантования являются избыточными, или блоки, для которых не выполнялось обратное квантование с использованием параметров квантования, исключают из параметров квантования декодированных блоков, соседствующих в пространстве или во времени с блоком, подлежащим декодированию, и получают кандидатов для выбора. Для задания прогнозного параметра квантования выбирают параметр квантования в порядке, указанном идентификационной информацией, включенной в потоковую информацию потока, где указанные соседствующие декодированные блоки расположены в заданном, например, в альтернативном варианте кандидатов для выбора определяют в порядке, заданном заранее, и задают прогнозный параметр квантования на основе результата определения. В качестве еще одной альтернативы выбирают один из указанных выше вариантов обработки – задают в качестве прогнозного параметра квантования параметр

квантования в порядке, указанном идентификационной информацией, включенной в потоковую информацию, или определяют кандидатов для выбора в порядке, заданном заранее, и задают прогнозный параметр квантования на основе результата определения, включенного в потоковую информацию. Далее, в устройстве декодирования изображения параметр квантования для блока, подлежащего декодированию, вычисляют путем добавления разности, указанной посредством разностной информации, к прогнозному параметру квантования. Кроме того, если нет кандидатов для выбора, в качестве прогнозного параметра квантования принимают параметр квантования начального значения для среза. Кроме того, в кандидаты для выбора включают также параметр квантования, обновленный последним.

Вторым аспектом предлагаемой технологии является способ декодирования изображения, включающий в себя процесс получения параметров квантования декодированных блоков, соседствующих в пространстве или во времени с блоком, подлежащим декодированию, в качестве кандидатов для выбора и извлечение из потоковой информации разностной информации, указывающей разность относительно прогнозного параметра квантования, выбранного из указанных кандидатов для выбора; и процесс вычисления параметра квантования для блока, подлежащего декодированию, на основе указанных прогнозного параметра квантования и разностной информации.

Третьим аспектом предлагаемой технологии является устройство кодирования изображения, включающее в себя: модуль управления, выполненный с возможностью задания параметра квантования для блока, подлежащего кодированию; модуль генерирования информации, выполненный с возможностью получения параметров квантования кодированных блоков, соседствующих в пространстве или во времени с блоком, подлежащим кодированию, в качестве кандидатов для выбора, выбора из указанных кандидатов для выбора одного прогнозного параметра квантования в соответствии с заданным параметром квантования и генерирования разностной информации, указывающей разность между прогнозным параметром квантования и заданным параметром квантования; и модуль кодирования, выполненный с возможностью включения разностной информации в потоковую информацию, генерируемую в результате кодирующей обработки блока, подлежащего кодированию, с использованием указанного заданного параметра квантования.

При использовании предлагаемой технологии по меньшей мере блоки, в которых параметры квантования являются избыточными, или блоки, для которых не выполнялось квантование с использованием параметров квантования, исключают из указанных параметров квантования кодированных блоков, соседствующих в пространстве или во времени с блоком, подлежащим кодированию, и получают кандидатов для выбора. Кроме того, в совокупность кандидатов для выбора включают также параметр квантования, обновленный последним, или аналогичный параметр. Параметр квантования, разность между которым и заданным параметром квантования из указанных кандидатов для выбора является минимальной, выбирают в качестве прогнозного параметра квантования и генерируют идентификационную информацию для выбора прогнозного параметра квантования из указанных кандидатов для выбора. Например, идентификационная информация представляет собой порядок блоков, соответствующий выбранному параметру квантования, в котором соседствующие кодированные блоки располагаются в заданном порядке. Кроме того, заданный порядок представляет собой такой порядок, в котором приоритет отдан кодированному блоку, соседствующему слева, или кодированному блоку, соседствующему сверху, или кодированному блоку, соседствующему во времени. Кроме того, порядок следования

соседствующих кодированных блоков можно изменять. Кроме того, параметры квантования кодированных блоков, соседствующих во времени, можно переупорядочивать в соответствии со значениями этих параметров, причем порядок  
 5 выбранных параметров квантования используется в качестве идентификационной информации. Кроме того, определение кандидатов для выбора может быть выполнено в заранее заданном порядке, и прогнозный параметр квантования выбирают на основе результата такого определения. Далее, в устройстве кодирования изображения генерируют разностную информацию, указывающую разность между прогнозным  
 10 параметром квантования и заданным параметром квантования. Кроме того, если нет кандидатов для выбора, генерируют разностную информацию, указывающую разность между начальным значением параметра квантования в срезе и заданными параметрами квантования. Также может быть сделан выбор между обработкой задания параметра квантования, для которого разность относительно заданного параметра квантования является наименьшей, в качестве прогнозного параметра квантования, с одной стороны,  
 15 и обработкой определения кандидатов для выбора в заданном заранее порядке, и выбором прогнозного параметра квантования на основе результата такого определения, с другой стороны, после чего генерируют информацию определения, указывающую выбранную обработку. Сформированные разностную информацию, идентификационную информацию и информацию определения включают в потоковую информацию, генерируемую посредством кодирующей обработки блока, подлежащего кодированию,  
 20 с использованием заданного параметра квантования.

Четвертым аспектом предлагаемой технологии является способ кодирования изображения, содержащий: процесс задания параметра квантования для блока, подлежащего кодированию; процесс получения параметров квантования для  
 25 кодированных блоков, соседствующих в пространстве или во времени с блоком, подлежащим кодированию, в качестве кандидатов для выбора, выбора прогнозного параметра квантования из указанной совокупности кандидатов для выбора в соответствии с заданным параметром квантования и генерирования разностной информации, указывающей разность между прогнозным параметром квантования и заданными параметрами квантования; и процесс включения разностной информации в потоковую информацию, генерируемую путем выполнения кодирующей обработки блока, подлежащего кодированию, с использованием заданного параметра квантования.  
 30

#### Полезные результаты изобретения

Согласно предлагаемой технологии параметры квантования кодированных блоков, соседствующих в пространстве и во времени с блоком, подлежащим кодированию,  
 35 принимают в качестве кандидатов для выбора, и выбирают прогнозный параметр квантования из указанных кандидатов для выбора в соответствии с параметром квантования, заданным для блока, подлежащего кодированию. Далее генерируют разностную информацию, указывающую разность между прогнозным параметром квантования и параметрами квантования, заданными для блока, подлежащего кодированию. Соответственно, можно не допустить получения большой величины разности параметров квантования и повысить эффективность кодирования параметров квантования.  
 40

Кроме того, в случае декодирования потоковой информации, куда включена  
 45 указанная разностная информация, выбирают прогнозный параметр квантования из параметров квантования декодированных блоков, соседствующих в пространстве или во времени с блоком, подлежащим декодированию, и вычисляют параметр квантования для блока, подлежащего декодированию, на основе прогнозного параметра квантования



и разностной информации. Соответственно, даже если потоковая информация была сгенерирована при повышенной эффективности кодирования параметров квантования, эти параметры квантования могут быть восстановлены на основе прогнозного параметра квантования и разностной информации при декодировании информации потока, и декодирующая обработка может быть выполнена правильно.

Краткое описание чертежей

Фиг. 1 представляет диаграмму, иллюстрирующую случай, когда порядок декодирования является порядком сканирования раstra.

Фиг. 2 представляет диаграмму, иллюстрирующую случай, когда порядок декодирования не является порядком сканирования раstra.

Фиг. 3 представляет схему, иллюстрирующую конфигурацию устройства для кодирования изображения.

Фиг. 4 представляет схему, иллюстрирующую конфигурацию модуля генерации информации.

Фиг. 5 представляет диаграмму, иллюстрирующую пример иерархической структуры единицы кодирования.

Фиг. 6 представляет логическую схему, иллюстрирующую операции устройства для кодирования изображения.

Фиг. 7 представляет логическую схему, иллюстрирующую процесс прогнозирования.

Фиг. 8 представляет логическую схему, иллюстрирующую процесс внутрикадрового прогнозирования.

Фиг. 9 представляет логическую схему, иллюстрирующую процесс межкадрового прогнозирования.

Фиг. 10 представляет диаграмму, описывающую операции модуля генерации информации.

Фиг. 11 представляет диаграмму, иллюстрирующую пример операций модуля генерации информации.

Фиг. 12 представляют логическую схему, иллюстрирующую обработку параметров квантования в процессе кодирования.

Фиг. 13 представляет диаграмму, иллюстрирующую пример набора параметров последовательности.

Фиг. 14 представляют логическую схему, иллюстрирующую процесс кодирующей обработки кадра.

Фиг. 15 представляет диаграмму, иллюстрирующую пример набора параметров изображения.

Фиг. 16 представляет диаграмму, иллюстрирующую пример заголовка среза.

Фиг. 17 представляют логическую схему, иллюстрирующую процесс кодирующей обработки среза.

Фиг. 18 представляет схему, иллюстрирующую конфигурацию устройства для декодирования изображения.

Фиг. 19 представляет схему, иллюстрирующую конфигурацию модуля вычисления параметра квантования.

Фиг. 20 представляют логическую схему, иллюстрирующую операции устройства для декодирования изображения.

Фиг. 21 представляют логическую схему, иллюстрирующую процесс генерации прогнозного изображения.

Фиг. 22 представляют логическую схему, иллюстрирующую процесс декодирования параметров квантования.

Фиг. 23 представляют логическую схему для описания других операций устройства для декодирования изображения.

Фиг. 24 представляет диаграмму, иллюстрирующую пример операций для случая неявного прогнозирования параметров квантования.

5 Фиг. 25 представляют логическую схему, иллюстрирующую пример для случая неявного прогнозирования параметров квантования.

Фиг. 26 иллюстрирует другой пример операций для случая неявного прогнозирования параметров квантования.

Фиг. 27 представляет диаграмму, иллюстрирующую пример программы.

10 Фиг. 28 представляют логическую схему для описания других операций устройства для декодирования изображения.

Фиг. 29 представляет схему, иллюстрирующую пример упрощенной конфигурации компьютера.

15 Фиг. 30 представляет схему, иллюстрирующую пример упрощенной конфигурации телевизионного приемника.

Фиг. 31 представляет схему, иллюстрирующую пример упрощенной конфигурации сотового телефона.

Фиг. 32 представляет схему, иллюстрирующую пример упрощенной конфигурации устройства записи/воспроизведения.

20 Фиг. 33 представляет схему, иллюстрирующую пример упрощенной конфигурации устройства для считывания изображения.

Осуществление изобретения

Далее будут описаны варианты настоящего изобретения. Отметим, что описание будет дано в следующем порядке.

- 25 1. Конфигурация устройства кодирования изображения  
 2. Операции устройства кодирования изображения  
 3. Операции генерации идентификационной информации и разностной информации на основе параметров квантования  
 4. Конфигурация устройства декодирования изображения  
 30 5. Операции устройства декодирования изображения  
 6. Другие операции устройства кодирования изображения и устройства декодирования изображения  
 7. Случай программной обработки  
 8. Случай применения к электронному оборудованию  
 35 1. Конфигурация устройства кодирования изображения

Фиг. 3 иллюстрирует конфигурацию устройства для кодирования изображения. Это устройство 10 для кодирования изображения содержит модуль 11 аналого-цифрового (A/D) преобразователя, буфер 12 реорганизации экрана, модуль 13 вычитания, модуль 14 ортогонального преобразования, модуль 15 квантования, модуль 16 кодирования  
 40 без потерь, промежуточный буфер 17 и модуль 18 управления частотой квантования. Кроме того, устройство 10 для кодирования изображения содержит модуль 21 обратного квантования, модуль обратного ортогонального преобразования 22, модуль 23 суммирования, деблокирующий фильтр 24, память 26 кадров, селектор 26, модуль 31 внутрикадрового прогнозирования, модуль 32 прогнозирования/компенсации движения  
 45 и модуль 33 выбора прогнозного изображения/оптимального режима.

Модуль 11 аналого-цифрового преобразователя осуществляет преобразование аналоговых сигналов изображения в цифровые данные изображения и передает эти данные в буфер 12 реорганизации экрана.

Буфер 12 реорганизации экрана осуществляет реорганизации (переупорядочение) кадров данных изображения, поступающих от модуля 11 аналого-цифрового преобразователя. Этот буфер 12 реорганизации экрана осуществляет такую реорганизацию кадров в соответствии со структурой GOP (группа изображений (Group of Pictures)), относящейся к кодирующей обработке, и передает данные изображения после реорганизации в модуль 13 вычитания, модуль 18 управления частотой квантования, модуль 31 внутрикадрового прогнозирования и модуль 32 прогнозирования/компенсации движения.

Данные изображения с выхода буфера 12 реорганизации экрана и данные прогнозного изображения, выбранные в описываемом позднее модуле 33 выбора прогнозного изображения/оптимального режима, поступают в модуль 13 вычитания. Этот модуль 13 вычитания вычисляет данные погрешности прогнозирования, представляющие собой разность между данными изображения с выхода буфера 12 реорганизации экрана и прогнозированным изображением от модуля 33 выбора прогнозного изображения/оптимального режима, и передает результат в модуль 14 ортогонального преобразования.

Указанный модуль 14 ортогонального преобразования осуществляет ортогональное преобразование, такое как дискретное косинусное преобразование (DCT: Discrete Cosine Transform), преобразование Карунена-Лоэва (Karhunen-Loève) или подобное преобразование, применительно к данным погрешности прогнозирования, поступающим от модуля 13 вычитания. Этот модуль 14 ортогонального преобразования передает данные коэффициентов преобразования, полученные в результате ортогонального преобразования, в модуль 15 квантования.

Данные коэффициентов преобразования с выхода модуля 14 ортогонального преобразования и параметр квантования (масштаб квантования) от описываемого позднее модуля 19 генерации информации поступают в модуль 15 квантования. Этот модуль 15 квантования осуществляет квантование указанных данных коэффициентов преобразования и передает квантованные данные в модуль 16 кодирования без потерь и в модуль 21 обратного квантования. Кроме того, модуль 15 квантования изменяет частоту передачи квантованных данных на основе параметров квантования, заданных в модуле 18 управления частотой квантования.

Указанные квантованные данные с выхода модуля 15 квантования, идентификационная информация и разностная информация от описываемого позднее модуля 19 генерации информации, информация режима прогнозирования от модуля 31 внутрикадрового прогнозирования, а также информация режима прогнозирования и разностная информация вектора движения и другая подобная информация от модуля 32 прогнозирования/компенсации движения поступают в модуль 16 кодирования без потерь. Кроме того, информация, указывающая, какой режим – режим внутрикадрового прогнозирования или режим межкадрового прогнозирования, является оптимальным режимом, поступает от модуля 33 выбора прогнозного изображения/оптимального режима. Отметим, что упомянутая информация режима прогнозирования включает информацию о режиме прогнозирования и о размере блоков и т.п. в соответствии с тем, какой режим – внутрикадрового прогнозирования или межкадрового прогнозирования выбран.

Указанный модуль 16 кодирования без потерь осуществляет кодирование квантованных данных без потерь, например, с использованием кодирования в коде переменной длины, арифметического кодирования или другого подобного кодирования для генерации информации потока и передает эту информацию в промежуточный буфер

17. Кроме того, если оптимальным режимом является режим внутрикадрового прогнозирования, модуль 16 кодирования без потерь осуществляет кодирование без потерь применительно к информации режима прогнозирования, поступающей от модуля 31 внутрикадрового прогнозирования. Кроме того, если оптимальным режимом является режим внутрикадрового прогнозирования, модуль 16 кодирования без потерь осуществляет кодирование без потерь применительно к информации режима прогнозирования и разностным векторам движения и другой подходящей информации, поступающей от модуля 32 прогнозирования/компенсации движения. Далее, модуль 16 кодирования без потерь осуществляет кодирование без потерь применительно к информации, относящейся к параметрам квантования, такой как разностная информация, например. Модуль 16 кодирования без потерь включает информацию, полученную в результате кодирования без потерь, в информацию потока.

Указанный промежуточный буфер 17 записывает кодированный поток, поступающий модуля 16 кодирования без потерь. Кроме того, этот промежуточный буфер 17 передает записанный в нем кодированный поток на выход со скоростью, соответствующей скорости передачи данных в передающем тракте.

Модуль 18 управления частотой квантования контролирует величину доступного свободного пространства в промежуточном буфере 17 и задает параметры квантования так, чтобы если в буфере свободного места мало, частота передачи квантованных данных уменьшалась, а если свободного места в буфере достаточно, частота передачи квантованных данных увеличивалась. Кроме того, модуль 18 управления частотой квантования определяет степень сложности изображения, например параметр активности, представляющий собой информацию, указывающую изменчивость величин пикселей, например, с использованием данных изображения, поступающих от буфера 12 реорганизации экрана. Этот модуль 18 управления частотой квантования выбирает параметры квантования таким образом, чтобы реализовать грубое квантование для участков изображения, где изменчивость величин пикселей мала, и «тонкое» квантование для других участков, например, на основе результатов определения степени сложности изображения. Модуль 18 управления частотой квантования передает заданные им параметры квантования в модуль 19 генерации информации.

Указанный модуль 19 генерации информации передает параметры квантования, поступившие от модуля 18 управления частотой квантования, в модуль 15 квантования. Этот модуль 19 генерации информации принимает также параметры квантования кодированных блоков, соседствующих в пространстве или во времени с блоком, подлежащим кодированию, в качестве кандидатов для отбора. Модуль 19 генерации информации выбирает параметр квантования из совокупности кандидатов для отбора в соответствии с параметром квантования, заданным в модуле 18 управления частотой квантования, и принимает выбранный параметр в качестве прогнозного параметра квантования. Далее, модуль 19 генерации информации формирует идентификационную информацию, соответствующую выбранным параметрам квантования, т.е. идентификационную информацию для выбора прогнозируемых параметров квантования из совокупности кандидатов для отбора, и разностную информацию, обозначающую разность между прогнозируемыми параметрами квантования и заданными параметрами квантования.

Фиг. 4 иллюстрирует конфигурацию модуля генерации информации. Этот модуль 19 генерации информации имеет модуль 191 памяти параметров квантования и модуль 192 вычисления разности. Модуль 19 генерации информации передает в модуль 15 квантования параметры квантования, поступающие от модуля 18 управления частотой

квантования. Кроме того, модуль 19 генерации информации передает параметры квантования, поступившие от модуля 18 управления частотой квантования, в модуль 191 памяти параметров квантования и модуль 192 вычисления разности.

Указанный модуль 191 памяти параметров квантования сохраняет поступающие к нему параметры квантования. Модуль 192 вычисления разности считывает из совокупности параметров квантования кодированных блоков, сохраняемых в модуле 191 памяти параметров квантования, параметры квантования кодированных блоков, расположенных в пространстве или во времени по периферии блока, подлежащего кодированию, в качестве кандидатов для отбора. Кроме того, по меньшей мере блоки, в которых параметры квантования являются избыточными, и блоки, для которых квантование с использованием параметров квантования не проводилось, такие как блоки, где данные коэффициентов преобразования, подлежащие квантованию в модуле 15 квантования, все равны "0", например, исключают из совокупности кандидатов для отбора. Кроме того, модуль 192 вычисления разности исключает из совокупности кандидатов для отбора блоки (далее именуемые «пропускаемые блоки» ("skip blocks")), для которых было определено, что их нужно пропустить, на основе информации от описываемых позднее модуля 32 прогнозирования/компенсации движения и модуля 33 выбора прогнозного изображения/оптимального режима.

Упомянутый модуль 192 вычисления разности выбирает параметр квантования из указанной совокупности параметров квантования в соответствии с параметром квантования для блока, подлежащего кодированию, т.е. параметра квантования, поступивший от модуля 18 управления частотой квантования, в качестве прогнозного параметра квантования. Этот модуль 192 вычисления разности далее генерирует разностную информацию, обозначающую разность между идентификационной информацией для выбора прогнозируемых параметров квантования из совокупности кандидатов для отбора и параметром квантования для блока, подлежащего кодированию, и передает эту разностную информацию в модуль 16 кодирования без потерь.

Возвращаясь к Фиг. 3, указанный модуль 21 обратного квантования осуществляет обратное квантование применительно к квантованным данным, поступающим от модуля 15 квантования. Этот модуль 21 обратного квантования передает данные коэффициентов преобразования, полученные в результате обратного квантования, в модуль 22 обратного ортогонального преобразования.

Указанный модуль 22 обратного ортогонального преобразования осуществляет обратное преобразование данных коэффициентов преобразования, поступающих от модуля 21 обратного квантования, и передает полученные в результате данные в модуль 23 суммирования.

Этот модуль 23 суммирования осуществляет суммирование данных, поступивших от модуля 22 обратного ортогонального преобразования, с данными прогнозного изображения, поступающими от модуля 33 выбора прогнозного изображения/оптимального режима, для генерации данных декодированного изображения и передает эти данные декодированного изображения в деблокирующий фильтр 24 и в память 26 кадров. Отметим, что данные декодированного изображения используются в качестве данных опорного изображения.

Указанный деблокирующий фильтр 24 осуществляет фильтрацию с целью уменьшения искажений блоков, возникающих во время кодирования изображения. Этот деблокирующий фильтр 24 осуществляет фильтрацию для удаления искажений блоков из данных декодированного изображения, поступающих от модуля 23 суммирования,

и передает эти данные декодированного изображения после фильтрации в память 26 кадров.

Эта память 26 кадров сохраняет данные декодированного изображения после фильтрации, поступившие от деблокирующего фильтра 24. Указанные данные декодированного изображения, сохраняемые в памяти 26 кадров, передают в модуль 32 прогнозирования/компенсации движения в качестве данных опорного изображения.

Модуль 31 внутрикадрового прогнозирования осуществляет прогнозирование во всех режимах-кандидатах внутрикадрового прогнозирования с использованием данных входного изображения, поступающих от буфера 12 реорганизации экрана, и данных опорного изображения, поступающих от модуля 23 суммирования, и определяет оптимальный режим внутрикадрового прогнозирования. Указанный модуль 31 внутрикадрового прогнозирования вычисляет величину целевой функции для каждого режима внутрикадрового прогнозирования, и принимает режим внутрикадрового прогнозирования, для которого эффективность кодирования является наилучшей на основе вычисленной величины целевой функции, в качестве оптимального режима внутрикадрового прогнозирования. Модуль 31 внутрикадрового прогнозирования передает данные прогнозного изображения, сформированные в оптимальном режиме внутрикадрового прогнозирования, и величину целевой функции для оптимального режима внутрикадрового прогнозирования в модуль 33 выбора прогнозного изображения/оптимального режима. Далее, модуль 31 внутрикадрового прогнозирования передает информацию режима прогнозирования, указывающую режим внутрикадрового прогнозирования, в модуль 16 кодирования без потерь.

Модуль 32 прогнозирования/компенсации движения осуществляет прогнозирование по всем режимам-кандидатах межкадрового прогнозирования с использованием данных входного изображения, подлежащего кодированию, от буфера 12 реорганизации экрана и данных опорного изображения, поступающих от памяти 26 кадров, и определяет оптимальный режим межкадрового прогнозирования. Модуль 32 прогнозирования/компенсации движения вычисляет величину целевой функции в каждом режиме межкадрового прогнозирования, например, и принимает режим межкадрового прогнозирования, для которого эффективность кодирования является наилучшей на основе вычисленной величины целевой функции, в качестве оптимального режима межкадрового прогнозирования. Этот модуль 32 прогнозирования/компенсации движения передает данные прогнозного изображения, сформированные в оптимальном режиме межкадрового прогнозирования, и величину целевой функции для оптимального режима межкадрового прогнозирования в модуль 33 выбора прогнозного изображения/оптимального режима. Далее, модуль 32 прогнозирования/компенсации движения передает информацию режима прогнозирования, относящуюся к оптимальному режиму межкадрового прогнозирования, в модуль 16 кодирования без потерь и в модуль 19 генерации информации.

Модуль 33 выбора прогнозного изображения/оптимального режима сравнивает величину целевой функции, поступившую от модуля 31 внутрикадрового прогнозирования, с величиной целевой функции, поступившей от модуля 32 прогнозирования/компенсации движения, и выбирает режим, для которого величина целевой функции меньше, чем в другом режиме, в качестве оптимального режима, в котором эффективность кодирования будет наилучшей. Кроме того, модуль 33 выбора прогнозного изображения/оптимального режима передает данные прогнозного изображения, сформированные в оптимальном режиме, в модуль 13 вычитания и в модуль 23 суммирования. Далее, модуль 33 выбора прогнозного изображения/

оптимального режима передает информацию, указывающую, является ли оптимальный режим указанным режимом внутрикадрового прогнозирования или режимом межкадрового прогнозирования, в модуль 16 кодирования без потерь и в модуль 19 генерации информации. Отметим, что модуль 33 выбора прогнозного изображения/

5 оптимального режима осуществляет переключение внутрикадрового прогнозирования или межкадрового прогнозирования от среза к срезу.

## 2. Операции устройства для кодирования изображения

В рассматриваемом устройстве для кодирования изображения кодирующая обработка осуществляется для размера макроблока, увеличенного сверх того, что задано,

10 например, а формате стандарта H.264/AVC. На Фиг. 5 изображен пример иерархической структуры единицы кодирования. Отметим, что Фиг. 5 иллюстрирует случай, когда максимальный размер составляет 128 пикселей  $\times$  128 пикселей, а глубина иерархии (глубина (Depth)) равна "5". Например, когда глубина иерархии равна "0", единица CU0 кодирования представляет собой блок размером  $2N \times 2N$  ( $N = 64$  пикселя). Кроме того,

15 когда флаг разбиения (split flag) = 1, единицу CU0 кодирования разбивают на четыре независимых блока размером  $N \times N$ , где эти блоки  $N \times N$  представляют собой блоки, находящиеся на один иерархический уровень ниже. Иными словами, глубина иерархии равна "1", а единицами CU1 кодирования являются блоки размером  $2N \times 2N$  ( $N = 32$  пикселя). Точно таким же образом, когда флаг разбиения = 1, каждый из этих блоков

20 разбивают на четыре независимых блока. Далее, когда глубина равна "4", что является самым глубоким иерархическим уровнем, единицами CU4 кодирования являются блоки размером  $2N \times 2N$  ( $N = 4$  пикселя), а размер 8 пикселей  $\times$  8 пикселей является наименьшим размером для единиц CU кодирования. Кроме того, согласно алгоритму HEVC, определяют единицу прогнозирования (PU: Prediction Unit), которая является

25 базовой единицей для разбиения единиц кодирования и для прогнозирования, и единицу преобразования TU: Transform Unit), которая является базовой единицей для преобразования и для квантования.

Далее, операции устройства для кодирования изображения будут описаны со ссылками на логическую схему, показанную на Фиг. 6. На этапе ST11 модуль 11 аналого-цифрового преобразования осуществляет аналого-цифровое преобразование сигналов

30 входного изображения.

На этапе ST12 буфер 12 реорганизации экрана осуществляет реорганизацию изображения. Этот буфер 12 реорганизации экрана сохраняет данные изображения, поступающие от модуля 11 аналого-цифрового преобразования, и изменяет порядок

35 кадров изображения от порядка для представления на дисплее к порядку для кодирования.

На этапе ST13 модуль 13 вычитания генерирует данные погрешности прогнозирования. Этот модуль 13 вычитания вычисляет разность между данными изображений, реорганизованных на этапе ST12, и данными прогнозного изображения, выбранного модулем 33 выбора прогнозного изображения/оптимального режима, с целью генерации данных погрешности прогнозирования. Объем данных погрешности прогнозирования меньше объема данных исходного изображения. Таким образом, объем данных может быть сжат по сравнению со случаем, когда изображение кодируют так, как оно есть.

40 На этапе ST14 модуль 14 ортогонального преобразования осуществляет обработку данных для выполнения ортогонального преобразования. Этот модуль 14 ортогонального преобразования осуществляет ортогональное преобразование применительно к данным погрешности прогнозирования, поступающим от модуля 13

вычитания. В частности, ортогональное преобразование, такое как дискретное косинусное преобразование, преобразование Карунена-Лоэва или другое подобное преобразование, выполняют применительно к данным погрешности прогнозирования для передачи на выход данных коэффициентов преобразования.

5 На этапе ST15 модуль 15 квантования осуществляет квантование данных. Этот модуль 15 квантования осуществляет квантование данных коэффициентов преобразования. Во время квантования производится управление частотой дискретизации, как это иллюстрирует описываемая позднее обработка данных на этапе ST25.

10 Модуль 21 обратного квантования осуществляет обработку обратного квантования на этапе ST16. Этот модуль 21 обратного квантования выполняет обратное квантование применительно к данным коэффициентов преобразования, квантованным в модуле 15 квантования и имеющим характеристики, соответствующие характеристикам модуля 15 квантования.

15 На этапе ST17 модуль 22 обратного ортогонального преобразования осуществляет обработку обратного ортогонального преобразования. Этот модуль 22 обратного ортогонального преобразования выполняет обратное ортогональное преобразование применительно к данным коэффициентов преобразования, подвергнутым обратному квантованию в модуле 21 обратного квантования и имеющим характеристики, соответствующие характеристикам модуля 14 ортогонального преобразования.

20 На этапе ST18 модуль 23 суммирования генерирует данные опорного изображения. Этот модуль 23 суммирования осуществляет суммирование данных прогнозного изображения, поступающих от модуля 33 выбора прогнозного изображения/оптимального режима, с данными после обратного ортогонального преобразования, относящимися к соответствующей позиции прогнозного изображения, для генерации декодированных данных (данных опорного изображения).

25 На этапе ST19 указанный деблокирующий фильтр 24 осуществляет фильтрационную обработку. Этот деблокирующий фильтр 24 осуществляет фильтрацию данных декодированного изображения, поступающих от модуля 23 суммирования, и устраняет искажения блоков.

30 На этапе ST20 память 26 кадров сохраняет данные опорного изображения. Эта память 26 кадров сохраняет данные декодированного изображения после фильтрационной обработки (данные опорного изображения).

35 На этапе ST21 модуль 31 внутрикадрового прогнозирования и модуль 32 прогнозирования/компенсации движения осуществляют каждый прогнозирование. Иными словами модуль 31 внутрикадрового прогнозирования осуществляет внутрикадровое прогнозирование в режиме внутрикадрового прогнозирования, а модуль 32 прогнозирования/компенсации движения выполняет прогнозирование/компенсацию движения в режиме межкадрового прогнозирования. Такое прогнозирование будет описано ниже со ссылками на Фиг. 7, где прогнозирование осуществляют в каждом из всех режимов-кандидатов прогнозирования и вычисляют величины целевой функции для каждого из этих всех режимов-кандидатов прогнозирования в ходе такого прогнозирования. Далее, на основе вычисленных величин целевой функции выбирают оптимальный режим внутрикадрового прогноза и оптимальный режим межкадрового прогнозирования и передают прогнозируемое изображение, величину целевой функции и информацию режима прогнозирования, сформированные в выбранном режиме прогнозирования, в модуль 33 выбора прогнозного изображения/оптимального режима.



На этапе ST22 модуль 33 выбора прогнозного изображения/оптимального режима выбирает данные прогнозного изображения. Этот модуль 33 выбора прогнозного изображения/оптимального режима принимает решение об оптимальном режиме, в котором эффективность кодирования является наилучшей, на основе каждой из величин целевой функции, поступающих от модуля 31 внутрикадрового прогнозирования и модуля 32 прогнозирования/компенсации движения. Иными словами, этот модуль 33 выбора прогнозного изображения/оптимального режима выбирает единицу кодирования, для которой эффективность кодирования является наилучшей, на каждом из иерархических уровней, показанных на Фиг. 5, например, размер блока единицы прогнозирования в такой единице кодирования, а также, какое именно прогнозирование – внутрикадровое прогнозирование или межкадровое прогнозирование, следует выполнить. Далее, модуль 33 выбора прогнозного изображения/оптимального режима передает данные прогнозного изображения для выбранного оптимального режима в модуль 13 вычитания и в модуль 23 суммирования. Эти данные прогнозного изображения используют для вычислений на этапах ST13 и ST18, как описано выше.

На этапе ST23 указанный модуль 16 кодирования без потерь осуществляет кодирующую обработку в режиме без потерь. Этот модуль 16 кодирования без потерь выполняет кодирование без потерь применительно к квантованным данным с выхода модуля 15 квантования. Иными словами, кодирование без потерь, такое как кодирование в коде переменной длины или арифметическое кодирование, осуществляется применительно к квантованным данным с целью произвести сжатие этих данных. Кроме того, модуль 16 кодирования без потерь осуществляет кодирование без потерь применительно к информации режима прогнозирования и другой подобной информации, соответствующей данным прогнозного изображения, выбранного на этапе ST22, и включает данные, кодированные в режиме без потерь, такие как информация режима прогнозирования и другая подобная информация, в информацию потока, генерируемого в процессе кодирования квантованных данных без потерь.

На этапе ST24 промежуточный буфер 17 сохраняет данные. Этот промежуточный буфер 17 сохраняет информацию потока, поступающую от модуля 16 кодирования без потерь. Такую информацию потока, сохраненную в промежуточном буфере 17, считывают должным образом и передают на декодирующую сторону по тракту передачи данных.

На этапе ST25 модуль 18 управления частотой квантования осуществляет управление частотой квантования. Этот модуль 18 управления частотой квантования регулирует, в случае сохранения информации потока в промежуточном буфере 17, частоту дискретизации при выполнении операции квантования в указанном модуле 15 квантования таким образом, чтобы в промежуточном буфере 17 не было ни переполнения, ни недостаточного заполнения.

Далее, обработка данных в процессе прогнозирования на этапе ST21, показанном на Фиг. 6, будет описана со ссылками на логическую схему, изображенную на Фиг. 7.

На этапе ST31 модуль 31 внутрикадрового прогнозирования осуществляет обработку данных в режиме внутрикадрового прогнозирования. Этот модуль 31 внутрикадрового прогнозирования выполняет обработку в режиме внутрикадрового прогнозирования применительно к изображению, соответствующему единице прогнозирования, подлежащей кодированию, во всех режимах-кандидатах внутрикадрового прогнозирования. Отметим, что в качестве входных данных для такой прогнозирующей обработки декодированного изображения используются данные декодированного изображения, которые были подвергнуты деблокирующей фильтрации посредством

деблокирующего фильтра 24. В ходе такой прогнозирующей обработки осуществляют внутрикадровое прогнозирование во всех режимах-кандидатах внутрикадрового прогнозирования и вычисляют величину целевой функции для каждого из всех этих режимов-кандидатов внутрикадрового прогнозирования. После этого выбирают один режим внутрикадрового прогнозирования, в котором эффективность кодирования оказалась наилучшей, из совокупности всех режимов внутрикадрового прогнозирования на основе вычисленных величин целевой функции.

На этапе ST32 модуль 32 прогнозирования/компенсации движения осуществляет обработку данных в режиме межкадрового прогнозирования. Этот модуль 32 прогнозирования/компенсации движения выполняет межкадровое прогнозирование во всех режимах-кандидатах межкадрового прогнозирования с использованием данных декодированного изображения после деблокирующей фильтрации, сохраняемых в памяти 26 кадров. В ходе такой прогнозирующей обработки осуществляют межкадровое прогнозирование во всех режимах-кандидатах межкадрового прогнозирования и вычисляют величину целевой функции для каждого из всех этих режимов-кандидатов межкадрового прогнозирования. После этого выбирают один режим межкадрового прогнозирования, в котором эффективность кодирования оказалась наилучшей, из совокупности всех режимов внутрикадрового прогнозирования на основе вычисленных величин целевой функции.

Обработка данных в режиме внутрикадрового прогнозирования на этапе ST31, показанном на Фиг. 7, будет описана со ссылками на логическую схему, представленную на Фиг. 8.

На этапе ST41 модуль 31 внутрикадрового прогнозирования осуществляет внутрикадровое прогнозирование в каждом из режимов прогнозирования. Этот модуль 31 внутрикадрового прогнозирования генерирует данные прогнозного изображения в каждом из режимов внутрикадрового прогнозирования с использованием данных декодированного изображения, полученных до и после деблокирующей фильтрации.

На этапе ST42 модуль 31 внутрикадрового прогнозирования вычисляет величины целевой функции в каждом из режимов прогнозирования. Вычисление величин целевой функции производится, как описано в совместной модели (JM (Joint Model)), которая является базовым пакетом программного обеспечения для формата H.264/AVC, на основе одного из вариантов – варианта Высокой сложности (High Complexity mode) или варианта низкой сложности (Low Complexity mode).

Иными словами, в варианте Высокой сложности предварительно выполняют операции вплоть до кодирования без потерь во всех режимах-кандидатах прогнозирования и вычисляют величину целевой функции, представленную следующим Выражением (1), в каждом из режимов прогнозирования.

$$\text{Cost}(\text{Mode} \in \Omega) = D + \lambda \cdot R \dots (1)$$

Здесь параметр  $\Omega$  обозначает все множество режимов-кандидатов прогнозирования, проверяемых с целью кодирования изображения рассматриваемой единицы прогнозирования. Параметр  $D$  обозначает энергию разности (искажений) между декодированным изображением и входным изображением, если кодирование осуществлялось в соответствующем режиме прогнозирования. Параметр  $R$  обозначает объем сформированного кода, включая коэффициент ортогонального преобразования, информацию режима прогнозирования и другую подобную информацию, и параметр  $\lambda$  обозначает множитель Лагранжа, представляющий собой функцию параметра  $QP$  квантования. Иными словами, в варианте Высокой сложности предварительно выполняют операции вплоть до кодирования без потерь во всех режимах-кандидатах

прогнозирования, как это описано на этапе ST42, и вычисляют величину целевой функции, представленную Выражением (1) выше, в каждом из режимов прогнозирования.

С другой стороны, в варианте Низкой сложности, генерацию прогнозируемых изображений и генерацию битов заголовка, включающих разностные векторы движения и информацию режима прогнозирования, а также другую подобную информацию, осуществляют во всех режимах-кандидатах прогнозирования и вычисляют величины целевой функции в соответствии со следующим Выражением (2).

$$\text{Cost}(\text{Mode} \in \Omega) = D + \text{QP2Quant}(\text{QP}) \cdot \text{Header\_Bit} \dots (2)$$

Здесь параметр  $\Omega$  обозначает все множество режимов-кандидатов прогнозирования, проверяемых с целью кодирования изображения рассматриваемой единицы прогнозирования. Параметр D обозначает энергию разности (искажений) между декодированным изображением и входным изображением, если кодирование осуществлялось в соответствующем режиме прогнозирования. Параметр Header\_Bit представляет собой бит заголовка для соответствующего режима прогнозирования и параметр QP2Quant представляет собой функцию параметра QP квантования. Иными словами, в варианте Низкой сложности вычисляют величину целевой функции, представленную Выражением (2) выше, в каждом из режимов прогнозирования с использованием сформированных прогнозного изображения и битов заголовка, таких как векторы движения и информация режима прогнозирования, а также другая подобная информация, как при обработке данных на этапе ST42.

На этапе ST43 модуль 31 внутрикадрового прогнозирования определяет оптимальный режим внутрикадрового прогнозирования. Этот модуль 31 внутрикадрового прогнозирования выбирает режим внутрикадрового прогнозирования, в котором величина целевой функции является наименьшей, на основе вычисленных на этапе ST42 величин целевой функции и принимает его в качестве оптимального режима внутрикадрового прогнозирования.

Далее, обработка данных в режиме межкадрового прогнозирования на этапе ST32, показанном на Фиг. 7, будет описана со ссылками на логическую схему, представленную на Фиг. 9.

На этапе ST51 модуль 32 прогнозирования/компенсации движения выполняет обработку данных для определения движения. Этот модуль 32 прогнозирования/компенсации движения определяет векторы движения и переходит к этапу ST52.

На этапе ST52 модуль 32 прогнозирования/компенсации движения осуществляет компенсацию движения. Этот модуль 32 прогнозирования/компенсации движения производит компенсацию движения с использованием данных опорного изображения и на основе векторов движения, найденных на этапе ST51, и генерирует данные прогнозного изображения.

На этапе ST53 модуль 32 прогнозирования/компенсации движения вычисляет величины целевой функции. Этот модуль 32 прогнозирования/компенсации движения вычисляет такие величины целевой функции, как описано выше, с использованием в качестве входной информации прогнозного изображения, подлежащего кодированию, и данных прогнозного изображения, сформированных на этапе ST52, и другой подобной информации, после чего переходит к этапу ST54.

Модуль 32 прогнозирования/компенсации движения определяет оптимальный режим межкадрового прогнозирования. Этот модуль 32 прогнозирования/компенсации движения осуществляет обработку данных в соответствии с этапами ST51 – ST53 для каждого режима межкадрового прогнозирования. Указанный модуль 32 прогнозирования/компенсации движения различает опорный индекс, указывающий в

каком режиме величина целевой функции, вычисленная для каждого режима прогнозирования, является наименьшей, размер блока в единице кодирования и размер блока в единице прогнозирования в составе этой единицы кодирования и декодирует найденный оптимальный режим межкадрового прогнозирования. Отметим, что при  
 5 определении режима, в котором величина целевой функции является наименьшей, используют также величину целевой функции для случая межкадрового прогнозирования в режиме пропуска.

Кроме того, если в качестве оптимального режима прогнозирования в модуле 33 выбора прогнозного изображения/оптимального режима был выбран оптимальный  
 10 режим межкадрового прогнозирования, тогда модуль 32 прогнозирования/компенсации движения генерирует данные прогнозного изображения, так что данные прогнозного изображения для оптимального режима межкадрового прогнозирования могут быть переданы в модуль 13 вычитания и в модуль 23 суммирования.

3. Операции генерации идентификационной информации и разностной информации  
 15 на основе параметров квантования

В процессе описанной выше кодирующей обработки изображения устройство 10 для кодирования изображения задает параметры квантования таким образом, чтобы обеспечить адекватное квантование каждого блока в соответствии со степенью сложности изображения. Кроме того, устройство 10 для кодирования изображения  
 20 генерирует идентификационную информацию и разностную информацию и включает сформированную информацию в информацию потока с целью повысить эффективность кодирования параметров квантования, использованных в процессе квантования на этапе ST15.

Далее, будет дано описание, относящееся к идентификационной информации и разностной информации. Указанный модуль 18 управления частотой квантования задает параметры квантования с использованием формата управления объемом кода согласно, например, TM5 в стандарте MPEG2, например.

Далее будет рассмотрена обработка данных на этапах с 1 по 3 согласно TM5 в стандарте MPEG2.

На этапе 1 объем кода, который должен быть назначен каждому изображению в составе группы изображений (GOP (Group of Pictures)) распределяют для изображений, которые до рассматриваемого момента еще не были кодированы, включая изображения для назначения, на основе назначенного количества R битов. Это распределение повторяют в порядке кодированных изображений в группе GOP. В этот момент  
 35 назначение объема кода для каждого изображения осуществляют с использованием следующих двух допущений.

Первое допущение состоит в том, что произведение среднего кода масштаба (шкалы) квантования на объем сформированного кода, используемого во время кодирования каждого изображения, будет постоянным для каждого типа изображения, если не  
 40 изменится экран.

Соответственно, после кодирования каждого изображения параметры  $X_I$ ,  $X_P$  и  $X_B$  (глобальная мера сложности (Global Complexity Measure)), представляющие степень сложности экрана, обновляют в соответствии с Выражениями (3) – (5). Эти параметры позволяют оценить соотношение между кодом масштаба квантования и объемом  
 45 генерируемого кода.

$$X_I = S_I \cdot Q_I \dots (3)$$

$$X_P = S_P \cdot Q_P \dots (4)$$

$$X_B = S_B \cdot Q_B \dots(5)$$

Здесь величины  $S_I$ ,  $S_P$  и  $S_B$  представляют собой биты генерируемого кода в момент кодирования изображения, а величины  $Q_I$ ,  $Q_P$  и  $Q_B$  представляют собой средний код масштаба квантования в момент кодирования изображения. Кроме того, начальные значения иллюстрированы Выражениями (6), (7) и (8), использующими параметр частоты передачи битов данных  $bit\_rate$  [бит/с], который характеризует объем целевого кода.

$$X_I = 160 \times bit\_rate / 115 \dots(6)$$

$$X_P = 160 \times bit\_rate / 115 \dots(7)$$

$$X_B = 160 \times bit\_rate / 115 \dots(8)$$

Второе допущение состоит в том, что общее качество изображения будет постоянно оптимизированным, если отношения  $K_P$  и  $K_B$  кодов шкалы квантования для Р и В кадров, соответственно, к коду шкалы квантования для I-кадра, как для опоры, будут соответствовать Выражению (9).

$$K_P = 1,0; K_B = 1,4 \dots(9)$$

Иными словами, код шкалы квантования для В-кадров постоянно задан в 1,4 раза больше, чем код шкалы квантования для I и Р кадров. Это предполагает, что если сделать квантование В-кадров несколько более грубым по сравнению с I и Р кадрами и, тем самым, добавить объем кода, «сбереженный» В-кадрами, для кодирования I и Р кадров, качество изображения I и Р кадров будет улучшено, а также качество изображения В-кадров, «привязанное» к I и Р кадрам, тоже будет улучшено.

Согласно изложенным выше двум допущениям назначенные объемы кодов ( $T_I$ ,  $T_P$ ,  $T_B$ ) для каждого изображения в группе представляют собой величины, представленные Выражениями (10), (11) и (12). Отметим, что параметр  $picture\_rate$  (частота кадров) представляет число изображений, представляемых на дисплее за 1 с, в этой последовательности.

Математическое выражение 1

$$T_I = \max \left\{ \frac{R}{1 + \frac{N_P X_P}{X_I K_P} + \frac{N_B K_B}{X_I K_B}}, \frac{\text{bit\_rate}}{8 \times \text{picture\_rate}} \right\} \quad \dots (10)$$

$$T_P = \max \left\{ \frac{R}{N_P + \frac{N_B K_P X_B}{K_B X_P}}, \frac{\text{bit\_rate}}{8 \times \text{picture\_rate}} \right\} \quad \dots (11)$$

$$T_B = \max \left\{ \frac{R}{N_B + \frac{N_P K_B X_P}{K_P X_B}}, \frac{\text{bit\_rate}}{8 \times \text{picture\_rate}} \right\} \quad \dots (12)$$

Теперь, числа  $N_P$ ,  $N_B$  представляют собой числа  $P$  и  $B$  кадров, не кодированных в составе группы GOP. Иными словами, числа изображений, не кодированных в составе группы GOP, относительно числа изображений, для которых должно быть выполнено назначение, и изображений других типов, оценка сделана относительно того, сколько раз будет иметь место сформированный объем кода изображения для назначения объема кода, соответствующего этим изображениям, в описанных выше условиях оптимизации качества изображения. Далее, определяют число подлежащих кодированию изображений, составляющих объем кода, равный оценке генерируемого объема кода, эквивалентного всей совокупности некодированных изображений. Например, величина  $N_P X_P / X_I K_P$ , представляющая собой второе слагаемое в знаменателе первого аргумента в выражении, относящемся к  $T_I$ , выражает, какому числу  $I$  кадров эквивалентно по объему число  $N_P$  некодированных изображений в составе группы GOP. Кроме того, эту величину получают путем умножения числа  $N_P$  на долю  $S_P / S_I$  генерируемого объема кода для  $I$  кадров относительно генерируемого объема кода для  $P$  кадров и выражают посредством  $X_I$ ,  $X_P$  и  $X_B$ , как описано выше.

Число битов для назначения изображениям получают путем деления назначаемого объема  $R$  кода для некодированных изображений на число этих изображений. Отметим, однако, что этой величине приравнивают только нижний предел, учитывая, избыточный объем «служебного» кода («издержки») для заголовка и других подобных компонентов.

На основе полученного таким способом назначаемого объема кода обновляют объем  $R$  кода, который должен быть назначен некодированным изображениям в группе GOP, с использованием Выражения (13) каждый раз, когда кодируют каждое изображение после этапов 1 и 2.

$$R = R - S_{I,PB} \dots (13)$$

Кроме того, в момент кодирования первого изображения в группе GOP, величину R обновляют в соответствии со следующим Выражением (14).

Математическое выражение 2

$$R = \frac{\text{bit\_rate} \times N}{\text{picture\_rate}} - R \quad \dots (14)$$

где N – число изображений в составе группы GOP. Кроме того, исходная величина R в начале последовательности равна 0.

Теперь будет дано описание этапа 2. На этапе 2 получают код масштаба квантования для реального согласования назначаемых объемов кода ( $T_I$ ,  $T_P$ ,  $T_B$ ) для каждого изображения, полученных на этапе 1, с актуальным объемом кода. Код масштаба квантования получают с использованием обратной связи с шагом в единицах макроблоках для каждого типа изображения на основе емкостей трех независимо организованных виртуальных буферов.

Сначала, перед кодированием j-го макроблока получают объемы занятого пространства в виртуальных буферах в соответствии с выражениями с (15) по (17).

Математическое выражение 3

$$d_j^I = d_0^I + B_{j-1} - \frac{T_I \times (j-1)}{MB_{cnt}} \quad \dots (15)$$

$$d_j^P = d_0^P + B_{j-1} - \frac{T_P \times (j-1)}{MB_{cnt}} \quad \dots (16)$$

$$d_j^B = d_0^B + B_{j-1} - \frac{T_B \times (j-1)}{MB_{cnt}} \quad \dots (17)$$

здесь величины  $d_0^I$ ,  $d_0^P$  и  $d_0^B$  представляют собой первоначальные размеры занятого пространства в виртуальных буферах,  $B_j$  количество уже сформированных битов от начала изображения до j-го макроблока и  $MB_{cnt}$  число макроблоков в одном изображении (кадре).

Объемы занятого пространства в виртуальных буферах на момент окончания кодирования каждого изображения ( $dMB_{cnt}^I$ ,  $dMB_{cnt}^P$ ,  $dMB_{cnt}^B$ ) используют в качестве исходных величин объема занятого пространства в виртуальных буферах для следующего изображения ( $d_0^I$ ,  $d_0^P$ ,  $d_0^B$ ) такого же типа, соответственно.

Далее, вычисляют опорный код  $Q_j$  масштаба квантования для j-го макроблока согласно Выражению (18).

Математическое выражение 4

$$Q_j = \frac{d_j \times 31}{r} \quad \dots (18)$$

здесь  $r$  – параметр, управляющий скоростью отклика в группе обратной связи, именуемый параметром реакции и получаемый посредством Выражения (19).

Математическое выражение 5

$$r = 2 \times \frac{\text{bit\_rate}}{\text{picture\_rate}} \quad \cdot \cdot \cdot \quad (19)$$

Отметим, что исходный объем занятого пространства в виртуальном буфере в начале последовательности получают посредством Выражения (20).

Математическое выражение 6

$$d_0^I = 10 \times \frac{r}{31}, d_0^P = K_P d_0^I, d_0^B = K_B d_0^I \quad \cdot \cdot \cdot \quad (20)$$

Далее, будет описан этап 3. Параметр активности получают с использованием Выражений с (21) по (23) с применением величин пикселей яркостного сигнала исходного изображения, например, с применением величин пикселей восьми блоков, а именно четырех блоков размером  $8 \times 8$  в режиме дискретного косинусного преобразования (DCT) при кодировании кадра и четырех блоков размером  $8 \times 8$  в режиме дискретного косинусного преобразования (DCT) при кодировании поля.

Математическое выражение 7

$$\text{act}_j = 1 + \min_{\text{sblk}=1,8} (\text{var sblk}) \quad \cdot \cdot \cdot \quad (21)$$

$$\text{var sblk} = \frac{1}{64} \sum_{k=1}^{64} (P_k - \bar{P})^2 \quad \cdot \cdot \cdot \quad (22)$$

$$\bar{P} = \frac{1}{64} \sum_{k=1}^{64} P_k \quad \cdot \cdot \cdot \quad (23)$$

Величина  $\text{var\_sblk}$  в Выражении (21) представляет собой сумму квадратов разностей между данными изображения для каждого пикселя и средней величиной этих данных, так что чем сложнее изображения этих блоков размером  $8 \times 8$ , тем больше эта величина  $\text{var\_sblk}$ . Параметры  $P_k$  в выражениях (22) и (23) представляют величины пикселей яркостных сигналов исходного изображения, входящих в состав блока. Причина, по которой в Выражении (22) выбрана минимальная ( $\min$ ) величина, состоит в стремлении сделать квантование более «тонким» в случае, когда в пределах макроблока размером  $16 \times 16$  имеется даже частично гладкий участок. Далее, Выражение (24) дает нормированную величину активности  $N_{\text{actj}}$ , значение которой лежит в диапазоне от 0,5 до 2.

Математическое выражение 8

$$N_{\text{actj}} = \frac{2 \times \text{act}_j + \text{avg\_act}}{\text{act}_j + 2 \times \text{avg\_act}} \quad \cdot \cdot \cdot \quad (24)$$



Величина  $avg\_act$  представляет собой среднюю величину параметра активности, вычисленную вплоть до изображения, кодированного непосредственно перед рассматриваемым изображением, включительно. Код  $mquant_j$  масштаба квантования, учитывающий визуальные характеристики, получают посредством Выражения (25) на основе опорного кода  $Q_j$  масштаба квантования.

Математическое выражение 9

$$mquant_j = Q_j \times Nact_j \quad \cdot \cdot \cdot \quad (25)$$

Указанный модуль 18 управления частотой квантования передает на выход код  $mquant_j$  масштаба квантования, вычисленный, как описано выше, в качестве параметра квантования. Кроме того, параметр квантования генерируют для макроблока, расположенного на границе среза, таким же способом, как и для макроблока, расположенного не на границе среза. Отметим, что способ выбора параметра квантования не ограничивается случаями, когда выбор осуществляется на основе активности, как отмечено выше, а также возможен выбор по меньшему значению величины целевой функции.

Отметим, что вместе с приведенным выше описанием способа управления частотой дискретизации согласно ТМ5 в стандарте MPEG2 рассмотрен случай, когда обработка осуществляется с шагом, измеряемым в единицах макроблоков. Соответственно, если выполнять аналогичную обработку с шагом в единицах блоков, с которым можно менять параметры квантования, эти параметры квантования могут быть заданы для каждого из блоков, для которого такие параметры квантования можно изменять.

Далее, будет приведено описание операций для генерации информации, используемой для повышения эффективности кодирования параметров квантования. Указанный модуль 19 генерации информации принимает кодированные параметры квантования блоков, соседствующих в пространстве или во времени с блоком, подлежащим кодированию, в качестве кандидатов для отбора. Этот модуль 19 генерации информации выбирает также параметр квантования из совокупности кандидатов для отбора в соответствии с параметром квантования, заданным для блока, подлежащего кодированию, и принимает этот выбранный параметр в качестве прогнозного параметра квантования. Далее, модуль 19 генерации информации формирует идентификационную информацию для выбора прогнозного параметра квантования из совокупности кандидатов для отбора и разностную информацию, обозначающую разность между прогнозированным параметром квантования и параметром квантования, заданным для блока, подлежащего кодированию.

Фиг. 10 представляет диаграмму, описывающую операции модуля генерации информации и иллюстрирующую кадр, подлежащий кодированию, и кодированный кадр, ближайший во времени к кадру, подлежащему кодированию, в порядке представления кадров на экране. Пусть параметр квантования блока, подлежащего кодированию, в составе кадра, подлежащего кодированию, обозначен, например, "QP\_0". Пусть также параметр квантования блока, соседствующего с рассматриваемым кадром слева, обозначен, например, "QP\_A". Пусть, аналогичным образом, параметры квантования блоков, соседствующих сверху, сверху справа, сверху слева и снизу слева с блоком, подлежащим кодированию, обозначены, например, "QP\_B", "QP\_C", "QP\_D" и "QP\_E", соответственно. Пусть также параметр квантования блока, соседствующего во времени, обозначен "QP\_T". Отметим, что во время кодирования блока, подлежащего кодированию, в составе кадра, подлежащего кодированию, параметры квантования с

"QR\_A" по "QR\_E" и "QR\_T" хранятся в модуле 191 памяти параметров квантования. Кроме того, пусть каждый блок является наименьшим определяющим величину шага блоком, для которого параметры квантования могут быть изменены.

Указанный модуль 192 вычисления разности принимает параметры квантования кодированных блоков, соседствующих с блоком, подлежащим кодированию, в качестве кандидатов для отбора, выбирает из совокупности этих кандидатов для отбора параметр квантования, для которого разность относительно параметра квантования, заданного для блока, подлежащего кодированию, является наименьшей, и принимает выбранный параметр в качестве прогнозного параметра квантования. Этот модуль 192 вычисления разности генерирует идентификационную информацию для выбора прогнозного параметра из совокупности кандидатов для отбора и разностную информацию, обозначающую разность между прогнозированным параметром квантования и параметром квантования для блока, подлежащего кодированию.

Фиг. 11 представляет диаграмму, иллюстрирующую пример операций модуля генерации информации. Отметим, что случаи, когда для блока не заданы параметры квантования из-за того, что это пропускаемый блок, или нет остаточной информации, обозначены знаком "-".

Здесь блок, подлежащий кодированию, обозначен как ВК0, а параметры квантования соседствующих блоков равны "QR\_A = 32", "QR\_B = 40", "QR\_C = 40", "QR\_D = 35", "QR\_E = -" и "QR\_T = 31". Здесь, модуль 19 генерации информации исключает блоки, для которых параметр квантования не задан, поскольку это пропускаемый блок или блок без остаточной информации, и блоки, где параметры квантования являются избыточными, из совокупности кандидатов. Соответственно, кандидатами для отбора являются кодированные блоки с параметрами квантования "QR\_A = 32", "QR\_B = 40", "QR\_D = 35" и "QR\_T = 31". Кроме того, модуль 19 генерации информации перед этим задает идентификационную информацию, индексные номера, например, для совокупности кандидатов для отбора. Эта идентификационная информация может быть задана только для соседствующих кодированных блоков или может быть задана для параметров квантования соседствующих кодированных блоков.

В случае задания идентификационной информации для соседствующих кодированных блоков модуль 19 генерации информации задает индексные номера в порядке массива, где соседствующие кодированные блоки перечислены в заданном порядке массива. Этот заданный порядок массива представляет собой, например, порядок, когда одному из блоков – кодированному блоку, соседствующему слева, кодированному блоку, соседствующему сверху, или кодированному блоку, соседствующему во времени, отдан приоритет. Кроме того, модуль 19 генерации информации может быть способен переключать порядок массива. Если есть возможность переключать порядок массива, информацию, обозначающую, какого сорта этот порядок массива, включают в информацию потока. Кроме того, модуль 16 кодирования без потерь и модуль 19 генерации информации осуществляют задание идентификационной информации и кодирование этой информации без потерь таким образом, чтобы уменьшить объем кода при кодировании идентификационной информации для блока, которому отдан приоритет.

Указанный модуль 192 вычисления разности выбирает из совокупности кандидатов для отбора того кандидата, для которого разность относительно параметра квантования блока, подлежащего кодированию, является наименьшей, и использует идентификационную информацию, заданную для этого выбранного кандидата, генерируя тем самым идентификационную информацию для выбора прогнозного параметра

квантования из совокупности кандидатов для отбора. Кроме того, модуль 192 вычисления разности генерирует разностную информацию, обозначающую разность между прогнозным параметром квантования, представляющим собой выбранный параметр-кандидат квантования, и параметром квантования для блока, подлежащего кодированию. Например, если приоритет отдан кодированному блоку, соседствующему на Фиг. 11 с левой стороны, модуль 19 генерации информации задает "0 (индексный номер)": для блока QR\_A", "1: для блока QR\_B", "2: для блока QR\_B" и "3: для блока QR\_T". Кроме того, если принять, что параметр квантования для блока, подлежащего кодированию, равен, например, "33", тогда модуль 192 вычисления разности вписывает индексный номер блока с наименьшей разницей относительно параметра квантования блока, подлежащего кодированию, в идентификационную информацию "0 (index No.)". Кроме того, этот модуль 192 вычисления разности генерирует разностную информацию "1 (= 33 - 32)", обозначающую разность между прогнозным параметром квантования и параметром квантования блока, подлежащего кодированию.

Задавая идентификационную информацию для блоков, подлежащих кодированию, таким способом, можно повысить эффективность кодирования параметров квантования. Например, если приоритет отдан блоку с левой стороны, а порядок блоков представлен последовательностью параметров квантования "QR\_A", "QR\_B", "QR\_C", "QR\_D", "QR\_E", "QR\_T", объем данных будет небольшим для изображений, в которых больше блоков, подлежащих кодированию, аналогичны изображению указанного блока с левой стороны. Кроме того, если приоритет был отдан блоку, находящемуся сверху, а порядок блоков представлен последовательностью параметров квантования "QR\_B", "QR\_A", "QR\_C", "QR\_D", "QR\_E", "QR\_T", объем данных будет небольшим для изображений, в которых больше блоков, подлежащих кодированию, аналогичны изображению указанного блока, находящегося сверху. Далее, если приоритет был отдан блоку, соседствующему во времени, а порядок блоков представлен последовательностью параметров квантования "QR\_T", "QR\_A", "QR\_B", "QR\_C", "QR\_D", "QR\_E", объем данных будет небольшим для изображений, в которых больше блоков, подлежащих кодированию, аналогичны изображению указанного блока, соседствующего во времени, т.е. объем данных будет меньше для изображений более неподвижных.

В случае задания идентификационной информации для параметров квантования соседствующих кодированных блоков модуль 19 генерации информации задает индексные номера, так что соседствующие кодированные блоки располагаются в заданном порядке массива. Например, модуль 19 генерации информации может задавать индексные номера в порядке параметров квантования с небольшими величинами. Иными словами, в случае, изображенном на Фиг. 11, модуль 19 генерации информации задает индексные номера так, что "0 (индексный номер): 32 (параметр квантования)", "1:40", "2:35", "3:31".

Модуль 192 вычисления разности выбирает из совокупности кандидатов одного кандидата, для которого разность относительно параметра квантования блока, подлежащего кодированию, является наименьшей, и использует идентификационную информацию для выбранного кандидата, генерируя тем самым идентификационную информацию для выбора прогнозного параметра квантования из совокупности кандидатов для отбора. Кроме того, этот модуль 192 вычисления разности генерирует разностную информацию, обозначающую разность между прогнозным параметром квантования и параметром квантования для блока, подлежащего кодированию. Например, если параметр квантования для блока, подлежащего кодированию, равен "33", модуль 192 вычисления разности генерирует разностную информацию "1 (= 33

-32)" в качестве идентификационной информации.

Кроме того, если нет кандидатов для отбора, указанный модуль 192 вычисления разности генерирует разностную информацию, обозначающую разность между параметром SliceQP<sub>Y</sub> квантования, имеющим начальную величину в срезе, и заданным параметром квантования.

Фиг. 12 представляет логическую схему, иллюстрирующую обработку параметров квантования в процессе кодирования. На этапе ST61, устройство 10 для кодирования изображения генерирует информацию для получения минимального размера единицы параметров квантования (MinQpUnitSize). Этот минимальный размер единицы параметров квантования представляет собой наименьший размер, для которого параметр квантования можно адаптивно переключать.

Это устройство 10 для кодирования изображения использует в качестве информации для получения минимального размера единицы параметров квантования (MinQpUnitSize) разность относительно минимального размера единицы преобразования (MinTransformUnitSize), например.

Минимальный размер единицы параметров квантования (MinQpUnitSize) определяется Выражением (26).

$$\text{MinQpUnitSize} = 1 \ll (\log_2\_min\_transform\_unit\_size\_minus2 + \log_2\_min\_qp\_unit\_size\_offset + 2) \dots (26)$$

Отметим, что параметр "log2\_min\_transform\_unit\_size\_minus2" представляет собой параметр для определения минимального размера единицы преобразования (MinTransformUnitSize).

Минимальный размер единицы преобразования (MinTransformUnitSize) определяется Выражением (27).

$$\text{MinTransformUnitSize} = 1 \ll (\log_2\_min\_transform\_unit\_size\_minus2 + 2) \dots (27)$$

Разность между минимальным размером единицы параметров квантования (MinQpUnitSize) и минимальным размером единицы преобразования (MinTransformUnitSize), как может быть ясно видно из Выражений (26) и (27), эквивалентна параметру "log2\_min\_qp\_unit\_size\_offset". Отметим, что параметры квантования используются с шагом, эквивалентным единицам преобразования (TU). Иными словами, в пределах единицы преобразования параметр квантования остается неизменным.

Кроме того, минимальный размер единицы параметров квантования (MinQpUnitSize) может быть определен в соответствии с размером единицы кодирования. В этом случае, устройство 10 для кодирования изображения использует, например, информацию, устанавливающую минимальный размер единицы CU кодирования (log2\_min\_coding\_block\_size\_minus3) и максимальный размер единицы CU кодирования (log2\_diff\_max\_min\_coding\_block\_size). Отметим, что максимальный размер единицы CU кодирования "log2MaxCUsSize" представлен Выражением (28).

$$\log_2\text{MaxCUsSize} = \log_2\_min\_coding\_block\_size\_minus\_3 + 3 + \log_2\_diff\_max\_min\_coding\_block\_size \dots (28)$$

Логарифмическая величина минимального размера единицы параметра квантования (log2MinQpUnitSize) представлена Выражением (29).

$$\log_2\text{MinQpUnitSize} = \log_2\_min\_coding\_block\_size\_minus\_3 + 3 + \log_2\_diff\_max\_min\_coding\_block\_size - \log_2\_min\_qp\_unit\_size\_offset \dots (29)$$

Соответственно, увеличение параметра "log2\_min\_qp\_unit\_size\_offset" делает

минимальный размер единицы параметров квантования меньше. Например, если наименьший размер единицы CU кодирования равен "8 × 8", а наибольший размер равен "64 × 64", тогда установление параметра "log2\_min\_qp\_unit\_size\_offset" равным "1" делает минимальный размер единицы параметров квантования равным "32 × 32". Кроме того, установление величины "log2\_min\_qp\_unit\_size\_offset" равной "2" делает минимальный размер единицы параметров квантования равным "16 × 16".

На этапе ST62 устройство 10 для кодирования изображения выполняет обработку данных для включения сформированной информации в информацию потока. Устройство 10 для кодирования изображения включает "log2\_min\_qp\_unit\_size\_offset" и параметр "log2\_min\_transform\_unit\_size\_minus2", представляющий собой параметр для определения минимального размера единицы преобразования (MinTransformUnitSize), в информацию потока и переходит к этапу ST63. Кроме того, в случае определения минимального размера единицы параметра квантования в соответствии с размером единицы кодирования в информацию потока включают параметры "log2\_min\_coding\_block\_size\_minus3", "log2\_diff\_max\_min\_coding\_block\_size" и "log2\_min\_qp\_unit\_size\_offset".

Устройство 10 для кодирования изображения включает сформированную информацию в набор параметров последовательности (SPS: sequence parameter set), определенный в качестве синтаксиса RBSP (полезная нагрузка последовательности исходных байтов (raw byte sequence payload)), например. Отметим, что Фиг. 13 иллюстрирует пример набора параметров последовательности.

На этапе ST63 устройство 10 для кодирования изображения определяет, имеется ли кадр, подлежащий кодированию. Если такой кадр имеется, устройство 10 для кодирования изображения переходит к этапу ST 64 и осуществляет кодирующую обработку этого кадра, представленную на Фиг. 14, а если такого кадра нет, завершает кодирующую обработку.

В процессе кодирующей обработки кадра, показанной на Фиг 14, на этапе ST71 устройство 10 для кодирования изображения определяет, имеется ли срез, подлежащий кодированию. Если срез для кодирования имеется, устройство 10 для кодирования изображения переходит к этапу ST72, а если такого среза нет, завершает кодирующую обработку кадра.

На этапе ST72 устройство 10 для кодирования изображения определяет параметр квантования для среза, подлежащего кодированию. Это устройство 10 для кодирования изображения определяет параметр квантования, имеющий начальную величину и такой размер, чтобы соответствовать целевому объему кода, и переходит к этапу ST73.

На этапе ST73 устройство 10 для кодирования изображения вычисляет параметр "slice\_qp\_delta". Параметр SliceQP<sub>Y</sub> квантования начальной величины в срезе определяется соотношением, представленным в Выражении (30), где параметр "pic\_init\_qp\_minus26" задает пользователь или этот параметр определяют другим подобным способом заранее. Соответственно устройство 10 для кодирования изображения вычисляет параметр "slice\_qp\_delta" в качестве параметра квантования, определяемого на этапе ST72, и переходит к этапу ST74.

$$\text{SliceQP}_Y = 26 + \text{pic\_init\_qp\_minus26} + \text{slice\_qp\_delta} \dots (30)$$

На этапе ST74 устройство 10 для кодирования изображения включает параметры "slice\_qp\_delta" и "pic\_init\_qp\_minus26" в информацию потока. Устройство 10 для кодирования изображения включает вычисленный параметр "slice\_qp\_delta" в срез заголовка, например, информации потока. Кроме того, устройство 10 для кодирования изображения включает параметр "pic\_init\_qp\_minus26", уже заданный к этому моменту, в набор параметров изображения, например, в составе информации потока. В результате

такого включения параметров "slice\_qp\_delta" и "pic\_init\_qp\_minus26" в информацию потока устройство для декодирования изображения, которое осуществляет декодирование информации потока, может вычислить имеющий начальную величину параметр SliceQPY квантования в срезе посредством Выражения (30). Отметим, что на Фиг. 15 показан пример набора параметров последовательности, а на Фиг. 16 – заголовок среза.

На этапе ST75 устройство 10 для кодирования изображения осуществляет кодирующую обработку среза. На Фиг. 17 приведена логическая схема, иллюстрирующая процесс кодирования среза.

На этапе ST81, показанном на Фиг. 17, устройство 10 для кодирования изображения определяет, имеется ли единица CU кодирования, подлежащая кодированию. Если в срезе, подлежащем кодированию, имеется единица кодирования, для которой еще не была проведена кодирующая обработка, устройство 10 для кодирования изображения переходит к этапу ST82. Кроме того, если кодирующая обработка всех единиц кодирования в составе среза завершена, устройство 10 для кодирования изображения заканчивает кодирующую обработку этого среза.

На этапе ST82 устройство 10 для кодирования изображения определяет, имеется ли единица TU преобразования в составе единицы CU кодирования, подлежащей кодированию. Если подобная единица преобразования имеется, устройство 10 для кодирования изображения переходит к этапу ST83, а если такой единицы преобразования нет, переходит к этапу ST87. Например, если все коэффициенты, подлежащие квантованию с использованием некоторого параметра квантования, равны "0", или в случае пропускаемого блока процесс переходит к этапу ST87.

На этапе ST83 устройство 10 для кодирования изображения определяет параметр квантования для единицы CU кодирования, подлежащей кодированию. Модуль 18 управления частотой квантования в составе устройства 10 для кодирования изображения определяет параметр квантования в соответствии со степенью сложности изображения в составе единицы кодирования, как описано выше, или таким образом, чтобы величина целевой функции была мала, и переходит к этапу ST84.

На этапе ST84 устройство 10 для кодирования изображения задает идентификационную информацию для кандидатов для отбора. Модуль 19 генерации информации в составе устройства 10 для кодирования изображения принимает параметры квантования кодированных единиц кодирования, расположенных в пространстве или во времени по периферии единицы кодирования, подлежащей кодированию, в качестве кандидатов для отбора. Кроме того, если для рассматриваемого блока не заданы параметры квантования из-за того, что это пропускаемый блок или этот блок не имеет остаточной информации, либо если какой-то параметр квантования равен другому кандидату, модуль 19 генерации информации исключает их из числа кандидатов для отбора. Указанное устройство 10 для кодирования изображения задает идентификационную информацию, например, индекс (ref\_qp\_block\_index) для рассматриваемых кандидатов для отбора и переходит к этапу ST85.

На этапе ST85 устройство 10 для кодирования изображения генерирует идентификационную информацию и разностную информацию. Указанный модуль 19 генерации информации в составе устройства 10 для кодирования изображения выбирает из совокупности кандидатов для отбора одного кандидата, для которого разность относительно параметра квантования для единицы кодирования, подлежащей кодированию, является наименьшей, и принимает его в качестве прогнозного параметра квантования. Этот модуль 19 генерации информации генерирует идентификационную

информацию с использованием индекса (ref\_qp\_block\_index) выбранного кандидата в качестве идентификационной информации для выбора прогнозного параметра квантования из совокупности кандидатов для отбора. Кроме того, модуль 19 генерации информации принимает разность (qb\_qp\_delta) между прогнозированным параметром квантования и параметром квантования для единицы кодирования, подлежащей кодированию, в качестве разностной информации и переходит к этапу ST86. Теперь, если прогнозированный параметр квантования, указываемый индексом (ref\_qp\_block\_index) найденного кандидата, обозначить "ref\_qp(ref\_qp\_block\_index)", параметр квантования для единицы кодирования, подлежащей кодированию, (CurrentQP) характеризуется соотношением, представленным Выражением (31).

$$\text{CurrentQP} = \text{qb\_qp\_delta} + \text{ref\_qp}(\text{ref\_qp\_block\_index}) \dots (31)$$

На этапе ST86 устройство 10 для кодирования изображения включает идентификационную информацию и разностную информацию в информацию потока. Модуль 16 кодирования без потерь в составе устройства 10 для кодирования изображения осуществляет кодирование без потерь применительно к идентификационной информации и разностной информации, сформированным в модуле 19 генерации информации, включает результат кодирования в информацию потока и переходит к этапу ST87.

На этапе ST87 устройство 10 для кодирования изображения использует найденный параметр квантования для осуществления квантования единицы кодирования в модуле 15 квантования и возвращается к этапу ST81.

Таким образом, устройство 10 для кодирования изображения выбирает из совокупности параметров квантования кодированных блоков, соседствующих в пространстве или во времени с блоком, подлежащим кодированию, одного кандидата, для которого разность относительно параметра квантования блока, подлежащего кодированию, является наименьшей, в качестве прогнозного параметра квантования. Кроме того, устройство 10 для кодирования изображения генерирует идентификационную информацию, соответствующую выбранному параметру квантования. Далее устройство 10 для кодирования изображения генерирует разностную информацию, обозначающую разность между прогнозированным параметром квантования и параметром квантования блока, подлежащего кодированию. Это устройство 10 для кодирования изображения включает сформированную идентификационную информацию и разностную информацию в состав информации потока. Таким образом, поскольку кандидат, разность для которого является наименьшей, выбран в качестве прогнозного параметра квантования, можно не допустить, чтобы разность между прогнозированным параметром квантования и параметром квантования блока, подлежащего кодированию, стала слишком большой. Соответственно, устройство 10 для кодирования изображения может повысить эффективность кодирования параметров квантования.

#### 4. Конфигурация устройства декодирования изображения

Далее будет описано устройство для декодирования изображения, осуществляющее декодирующую обработку информации потока, поступающей с выхода устройства для кодирования изображения. Кодированный поток, генерируемый в результате кодирования входного изображения, поступает в устройство для декодирования изображения по заданному тракту передачи данных, от носителя записи или другим подобным способом, и подвергается декодированию в этом устройстве.

На Фиг. 18 представлена схема, иллюстрирующая конфигурацию устройства для декодирования изображения, осуществляющего декодирование информации потока. Это устройство 50 для декодирования изображения содержит промежуточный буфер

51, модуль 52 декодирования без потерь, модуль 53 обратного квантования, модуль 54 обратного ортогонального преобразования, модуль 55 суммирования, деблокирующий фильтр 56, буфер 57 реорганизации экрана и модуль 58 цифро-аналогового преобразования. Более того, устройство 50 для декодирования изображения

5 содержит модуль 59 вычисления параметра квантования, память 61 кадров, модуль 71 внутрикадрового прогнозирования, модуль 72 компенсации движения и селектор 73.

Указанный промежуточный буфер 51 сохраняет переданную информацию потока. Модуль 52 декодирования без потерь осуществляет декодирование этой информации потока, поступившей из промежуточного буфера 51, согласно формату,

10 соответствующему формату кодирования в модуле 16 кодирования без потерь, показанном на Фиг. 3.

Указанный модуль 52 декодирования без потерь работает в качестве модуля получения информации и извлекает информацию различных типов из информации потока. Например, этот модуль 52 декодирования без потерь передает информацию

15 режима прогнозирования, полученную в результате декодирования потока, в модуль 71 внутрикадрового прогнозирования и в модуль 72 компенсации движения. Кроме того, модуль 52 декодирования без потерь передает разностные векторы движения, пороговые величины или информацию для генерации пороговых величин, полученные в результате декодирования информации потока, в модуль 72 компенсации движения.

20 Далее, этот модуль 52 декодирования без потерь передает полученную в результате декодирования информации потока информацию, относящуюся к параметрам квантования, например, разностную информацию и другую подобную информацию, в модуль 59 вычисления параметра квантования. Кроме того, модуль 52 декодирования без потерь передает данные квантования, полученные в результате декодирования

25 информации потока, в модуль 53 обратного квантования.

Указанный модуль 53 обратного квантования осуществляет обратное квантование применительно к данным квантования, декодированным в модуле 52 декодирования без потерь, согласно формату, соответствующему формату квантования в модуле 15 квантования, показанном на Фиг. 3. Указанный модуль 54 обратного ортогонального

30 преобразования осуществляет обратное ортогональное преобразование применительно к данным с выхода модуля 53 обратного квантования согласно формату, соответствующему формату ортогонального преобразования в модуле 14 ортогонального преобразования, показанном на Фиг. 3, и передает результат в модуль 55 суммирования.

Этот модуль 55 суммирования осуществляет суммирование данных после обратного ортогонального преобразования с данными прогнозного изображения, поступающими от селектора 73, для генерации данных декодированного изображения и передает результат в деблокирующий фильтр 56 и в модуль 71 внутрикадрового прогнозирования.

Указанный деблокирующий фильтр 56 осуществляет фильтрацию данных

40 декодированного изображения, поступающих от модуля 55 суммирования, удаляет искажения блоков и затем передает результат для сохранения в память 61 кадров, а также передает результат в буфер 57 реорганизации экрана.

Этот буфер 57 реорганизации экрана осуществляет реорганизацию (изменение порядка) изображений. Иными словами, порядок кадров, измененный буфером 12

45 реорганизации экрана, показанным на Фиг. 3, для расположения кадров в порядке кодирования, возвращается к первоначальному порядку кадров для представления их на дисплее, после чего данные передают в модуль 58 цифро-аналогового преобразования.



Этот модуль 58 цифро-аналогового преобразования осуществляет такое цифро-аналоговое преобразование данных изображения, поступающих из буфера 57 реорганизации экрана, чтобы представить потом это изображение на экране непоказанного дисплея.

Указанный модуль 59 вычисления параметра квантования восстанавливает параметры квантования на основе информации, поступающей от модуля 52 декодирования без потерь, и передает в модуль 53 обратного квантования. Фиг. 19 иллюстрирует конфигурацию модуля вычисления параметра квантования, имеющего вычислительный модуль 591 и модуль 592 памяти параметров квантования.

Вычислительный модуль 591 использует информацию, поступающую от модуля 52 декодирования без потерь, и параметры квантования, сохраняемые в модуле 592 памяти параметров квантования, для восстановления параметра квантования, использованного для квантования в процессе кодирования, которому был подвергнут блок, подлежащий декодированию, и передает результат в модуль 53 обратного квантования. Этот вычислительный модуль 591 сохраняет также параметр квантования блока, подлежащего декодированию, в модуле 592 памяти параметров квантования.

Вычислительный модуль 591 использует, например, параметр "pic\_init\_qr\_minus26", выделенный из набора параметров, и параметр "slice\_qr\_delta", выделенный из заголовка среза, для вычисления Выражения (30), вычисляет параметр SliceQP<sub>Y</sub> квантования и передает его в модуль 53 обратного квантования.

Указанный вычислительный модуль 591 использует также идентификационную информацию и разностную информации, поступающие от модуля 52 декодирования без потерь, и параметры квантования декодированных блоков, сохраняемые в модуле 592 памяти параметров квантования, и вычисляет параметр квантования блока, подлежащего декодированию. Этот вычислительный модуль 591 передает вычисленный параметр квантования в модуль 53 обратного квантования. В этом случае, вычислительный модуль 591 считывает из совокупности параметров квантования декодированных блоков, сохраняемых в модуле 592 памяти параметров квантования, параметры квантования декодированных блоков, расположенных в пространстве или во времени по периферии блока, подлежащего декодированию. Этот вычислительный модуль 591 задает совокупность кандидатов для отбора таким же способом, как это делается в модуле 192 вычисления разности. Например, вычислительный модуль 591 исключает по меньшей мере блоки, параметры квантования которых являются избыточными, или блоки, для которых обратное квантование с использованием параметров квантования не проводилось, и принимает оставшиеся параметры в качестве кандидатов для отбора. Далее, этот вычислительный модуль 591 задает идентификационную информацию, т.е. индекс (ref\_qr\_block\_index), как и модуль 192 вычисления разности, для параметров квантования каждого из кандидатов. Иными словами, вычислительный модуль 591 задает индекс (ref\_qr\_block\_index) с учетом соседствующих декодированных блоков в заданном порядке массива. Этот вычислительный модуль 591 вычисляет Выражение (31) с использованием параметра квантования "ref\_qr(ref\_qr\_block\_index)", соответствующего идентификационной информации, поступающей от модуля 52 декодирования без потерь, т.е.

прогнозируемому параметру квантования и разности, обозначенной разностной информацией (qb\_qr\_delta), поступающей от модуля 52 декодирования без потерь. Указанный вычислительный модуль 591 передает вычисленный параметр квантования (CurrentQP) в модуль 53 обратного квантования в качестве параметра квантования, подлежащего декодированию. Кроме того, если кандидатов для отбора нет,

вычислительный модуль 591 передает параметр квантования, имеющий начальную величину в срезе, в модуль 53 обратного квантования.

Кроме того, если из информации потока выделена информация, устанавливающая порядок массива блоков, вычислительный модуль 591 задает индекс (ref\_qp\_block\_index) с учетом декодированных блоков в заданном порядке массива. Соответственно, даже если в устройстве 10 для кодирования изображения порядок массива был изменен, все равно можно восстановить параметры квантования, использованные в устройстве 10 для кодирования изображения.

Возвращаясь к Фиг. 18, память 61 кадров сохраняет данные декодированного изображения после фильтрации, поступающие от деблокирующего фильтра 24.

Указанный модуль 71 внутрикадрового прогнозирования генерирует данные прогнозного изображения на основе информации режима прогнозирования, поступающей от модуля 52 декодирования без потерь, и данных декодированного изображения, поступающих от модуля 55 суммирования, и передает сформированные им данные прогнозного изображения селектору 73.

Указанный модуль 72 компенсации движения считывает данные опорного изображения из памяти 61 кадров на основе информации режима прогнозирования и вектора движения, поступающих от модуля 52 декодирования без потерь, и осуществляет компенсацию движения с целью генерации данных прогнозного изображения. Этот модуль 72 компенсации движения передает сформированные им данные прогнозного изображения в селектор 73. Кроме того, модуль 72 компенсации движения генерирует данные прогнозного изображения, переключая в процессе характеристики фильтра в соответствии с величиной векторов движения.

Указанный селектор 73 выбирает модуль 71 внутрикадрового прогнозирования в случае внутрикадрового прогнозирования или модуль 72 компенсации движения в случае межкадрового прогнозирования на основе информации режима прогнозирования, поступающей от модуля 52 декодирования без потерь. Этот селектор 73 передает данные прогнозного изображения, сформированные в выбранном им модуле 71 внутрикадрового прогнозирования или модуле 72 компенсации движения в модуль 55 суммирования.

#### 5. Операции устройства декодирования изображения

Далее, операции устройства 50 для декодирования изображения будут описаны со ссылками на логическую схему, представленную на Фиг. 20.

На этапе ST91 промежуточный буфер 51 сохраняет направленную ему информацию потока. На этапе ST92 модуль 52 декодирования без потерь осуществляет декодирование без потерь. Этот модуль 52 декодирования без потерь выполняет декодирование информации потока, поступающей из промежуточного буфера 51. Иными словами, модуль получает квантованные данные для каждого изображения, кодированного посредством модуля 16 кодирования без потерь, показанного на Фиг. 3. Кроме того, модуль 52 декодирования без потерь осуществляет декодирование без потерь информации режима прогнозирования, включенной в информацию потока, и, если полученная информация режима прогнозирования относится к режиму внутрикадрового прогнозирования, передает эту информацию режима прогнозирования в модуль 71 внутрикадрового прогнозирования. Кроме того, если информация режима прогнозирования относится к режиму межкадрового прогнозирования, этот модуль 52 декодирования без потерь передает информацию режима прогнозирования в модуль 72 компенсации движения. Далее, модуль 52 декодирования без потерь передает полученные в результате декодирования информации потока разностные векторы

движения, пороговые величины или информацию для генерации пороговых величин в модуль 72 компенсации движения.

На этапе ST93 модуль 53 обратного квантования осуществляет обратное квантование данных. Этот модуль 53 обратного квантования выполняет обратное квантование применительно к декодированным в модуле 52 декодирования без потерь квантованным данным с использованием характеристик, соответствующих характеристикам модуля 15 квантования, показанного на Фиг. 3.

На этапе ST94 модуль 54 обратного ортогонального преобразования осуществляет обратное ортогональное преобразование данных. Этот модуль 54 обратного ортогонального преобразования выполняет обратное ортогональное преобразование применительно к данным коэффициентов преобразования, подвергнутым обратному квантованию посредством модуля 53 обратного квантования, так что характеристики обратного ортогонального преобразования соответствуют характеристикам модуля 14 ортогонального преобразования, показанного на Фиг. 3.

На этапе ST95 модуль 55 суммирования генерирует данные декодированного изображения. Этот модуль 55 суммирования осуществляет суммирование данных, полученных в результате обратного ортогонального преобразования, с данными прогнозного изображения, выбранными на описываемом позднее этапе ST99, и генерирует данные декодированного изображения. Таким образом, исходное изображение оказывается декодировано.

На этапе ST96 деблокирующий фильтр 56 осуществляет фильтрацию. Этот деблокирующий фильтр 56 выполняет фильтрационную обработку данных декодированного изображения, поступающих от модуля 55 суммирования, и устраняет искажения блоков в составе декодированного изображения.

На этапе ST97 память 61 кадров сохраняет данные декодированного изображения. Отметим, что данные декодированного изображения, сохраненные в памяти 61 кадров, и данные декодированного изображения с выхода модуля 55 суммирования используют в качестве данных опорного изображения для генерации данных прогнозного изображения.

На этапе ST98 модуль 71 внутрикадрового прогнозирования и модуль 72 компенсации движения осуществляют прогнозирование. Каждый из этих модулей – модуль 71 внутрикадрового прогнозирования и модуль 72 компенсации движения выполняет прогнозирование в соответствии с информацией режима прогнозирования, поступающей от модуля 52 декодирования без потерь.

Иными словами, когда от модуля 52 декодирования без потерь поступает информация режима прогнозирования, указывающая режим внутрикадрового прогнозирования, модуль 71 внутрикадрового прогнозирования осуществляет внутрикадровое прогнозирование на основе информации режима прогнозирования и генерирует данные прогнозного изображения. Аналогично, когда от модуля 52 декодирования без потерь поступает информация режима прогнозирования, указывающая режим межкадрового прогнозирования модуль 72 компенсации движения осуществляет компенсацию движения на основе информации режима прогнозирования и генерирует данные прогнозного изображения.

На этапе ST99 селектор 73 выбирает данные прогнозного изображения. Этот селектор 73 выбирает данные прогнозного изображения, поступающие от модуля 71 внутрикадрового прогнозирования, и данные прогнозного изображения, поступающие от модуля 72 компенсации движения, и передает выбранные данные прогнозного изображения в модуль 55 суммирования, чтобы суммировать с данными с выхода

модуля 54 обратного ортогонального преобразования на этапе ST95, как описано выше.

На этапе ST100 буфер 57 реорганизации экрана осуществляет реорганизацию изображения. Иными словами, в буфере 57 реорганизации экрана, порядок кадров, измененный посредством буфера 12 реорганизации экрана в составе устройства 10 для кодирования изображения, показанном на Фиг. 3, вновь изменяют для возвращения к исходному порядку кадров для представления на дисплее.

На этапе ST101 модуль 58 цифро-аналогового преобразования осуществляет такое цифро-аналоговое преобразование данных изображения, поступающих от буфера 57 реорганизации экрана. Это изображение передают для представления на экране непоказанного дисплея.

Далее, процесс генерации прогнозного изображения на этапе ST98, показанном на Фиг. 20, будет описан со ссылками на логическую схему, представленную на Фиг. 21.

На этапе ST111 указанный модуль 52 декодирования без потерь определяет, был ли текущий блок закодирован способом внутрикадрового кодирования. Если информация режима прогнозирования, полученная в результате декодирования без потерь, является информацией режима внутрикадрового прогнозирования, модуль 52 декодирования без потерь передает эту информацию режима прогнозирования в модуль 53 внутрикадрового прогнозирования и переходит к этапу ST112. Аналогично, если эта информация режима прогнозирования не является информацией режима внутрикадрового прогнозирования, модуль 52 декодирования без потерь передает информацию режима прогнозирования в модуль 72 компенсации движения и переходит к этапу ST113.

На этапе ST112 модуль 71 внутрикадрового прогнозирования осуществляет внутрикадровое прогнозирование. Этот модуль 71 внутрикадрового прогнозирования выполняет внутрикадровое прогнозирование с использованием данных декодированного изображения до деблокирующей фильтрации и информации режима прогнозирования, поступающих от модуля 55 суммирования, и генерирует данные прогнозного изображения.

На этапе ST113 модуль 72 компенсации движения генерирует прогнозируемое изображение с применением межкадрового прогнозирования. Этот модуль 72 компенсации движения считывает данные опорного изображения из памяти 61 кадров и генерирует данные прогнозного изображения на основе информации, поступающей от модуля 52 декодирования без потерь, такой как информация режима прогнозирования или другая подобная информация.

Фиг. 22 представляет логическую схему, иллюстрирующую процесс декодирования параметров квантования. На этапе ST121 устройство 50 для декодирования изображения выделяет информацию для получения минимального размера единицы параметров квантования. Это устройство 50 для декодирования изображения извлекает такую информацию для получения минимального размера единицы параметров квантования, например, "log2\_min\_qp\_unit\_size\_offset", из информации потока и переходит к этапу ST122.

На этапе ST122 устройство 50 для декодирования изображения вычисляет минимальный размер единицы параметров квантования. Это устройство 50 для декодирования изображения вычисляет Выражение (26) с использованием параметра "log2\_min\_qp\_unit\_size\_offset" и параметра "log2\_min\_transform\_unit\_size\_minus2", определяющего минимальный размер единицы преобразования (MinTransformUnitSize), и вычисляет минимальный размер единицы параметров квантования (MinQpUnitSize).

Кроме того, устройство 50 для декодирования изображения может вычислить минимальный размер единицы параметров квантования (MinQpUnitSize) посредством Выражения (29).

На этапе ST123 устройство 50 для декодирования изображения определяет, имеется ли кадр, подлежащий декодированию. Если такой кадр, подлежащий декодированию, имеется, устройство 50 для декодирования изображения переходит к этапу ST124, а если кадров для декодирования нет, завершает обработку данных.

На этапе ST124 устройство 50 для декодирования изображения определяет, имеется ли срез, подлежащий декодированию. Если такой срез, подлежащий декодированию, имеется, устройство 50 для декодирования изображения переходит к этапу ST125, а если срезов для декодирования нет, возвращается к этапу ST123.

На этапе ST125 устройство 50 для декодирования изображения выделяет информацию для получения параметра квантования, имеющего начальную величину в срезе. Указанный модуль 52 декодирования без потерь в составе устройства 50 для декодирования изображения выделяет, например, параметр "pic\_init\_qp\_minus26" из набора параметров изображения (PPS: picture parameter set). Кроме того, устройство извлекает параметр "slice\_qp\_delta" из заголовка среза и переходит к этапу ST126.

На этапе ST126 устройство 50 для декодирования изображения вычисляет параметр квантования, имеющего начальную величину в срезе. Указанный модуль 59 вычисления параметра квантования в составе устройства 50 для декодирования изображения вычисляет Выражение (30) с использованием параметров "pic\_init\_qp\_minus26" и "slice\_qp\_delta", вычисляет параметр SliceQP<sub>Y</sub> квантования и переходит к этапу ST127.

На этапе ST127 устройство 50 для декодирования изображения определяет, имеется ли еще единица CU кодирования, подлежащая декодированию. Если такая единица, кодирования, подлежащая декодированию, имеется, устройство 50 для декодирования изображения переходит к этапу ST128, а если единиц кодирования, подлежащих декодированию, нет, возвращается к этапу ST124.

На этапе ST128 устройство 50 для декодирования изображения задает идентификационную информацию для кандидатов для отбора. Указанный модуль 59 вычисления параметра квантования в составе устройства 50 для декодирования изображения задает идентификационную информацию для кандидатов для отбора таким же образом, как это делает модуль 19 генерации информации в составе устройства 10 для кодирования изображения. Иными словами, этот модуль 59 вычисления параметра квантования принимает параметры квантования декодированных единиц кодирования, находящихся в пространстве или во времени по периферии единицы кодирования, подлежащей декодированию, в качестве кандидатов для отбора. Кроме того, если блоку не заданы параметры квантования из-за того, что этот блок является пропускаемым блоком или не имеет остаточной информации, или если какой-то параметр квантования равен другому кандидату, соответствующие параметры исключают из числа кандидатов для отбора. Модуль 59 вычисления параметра квантования задает идентификационную информацию, равную информации, задаваемой устройством 10 для кодирования изображения, например, индекс (ref\_qp\_block\_index), для параметров квантования из группы кандидатов и переходит к этапу ST129.

На этапе ST129 устройство 50 для декодирования изображения получает идентификационную информацию и разностную информацию. Указанный модуль 52 декодирования без потерь в составе устройства 50 для декодирования изображения выделяет идентификационную информацию и разностную информацию, включенную в состав информации потока устройством 10 для кодирования изображения, т.е. индекс

(ref\_qp\_block\_index) и разность (qb\_qp\_delta). Этот модуль 52 декодирования без потерь передает выделенные им идентификационную информацию и разностную информацию в модуль 59 вычисления параметра квантования и переходит к этапу ST130.

На этапе ST130 устройство 50 для декодирования изображения использует  
 5 идентификационную информацию и разностную информацию для вычисления параметров квантования. Указанный модуль 59 вычисления параметра квантования из состава устройства 50 для декодирования изображения вычисляет Выражение (31) с использованием параметра квантования "ref\_qp (ref\_qp\_block\_index)", соответствующего индексу (ref\_qp\_block\_index), являющемуся идентификационной информацией, и  
 10 параметра (qb\_qp\_delta), являющегося разностной информацией. Иными словами, суммируя указанную разность с прогнозным параметром квантования, вычисляют параметр квантования для единицы кодирования, подлежащей декодированию. Модуль 59 вычисления параметра квантования передает параметр квантования для единицы кодирования, подлежащей декодированию, (CurrentQP) в модуль 53 обратного  
 15 квантования и возвращается к этапу ST124.

Таким образом, используя идентификационную информацию и разностную информацию, включенные в информацию потока, можно восстановить параметры квантования, относящиеся к блоку, подлежащему декодированию, даже если параметры квантования для каждого из этих блоков не включены в информацию потока. Иными  
 20 словами, даже если эффективность кодирования параметров квантования была в устройстве 10 для кодирования изображения повышена за счет использования идентификационной информации и разностной информации, все равно в устройстве 50 для декодирования изображения можно восстановить и правильно декодировать параметры квантования, относящиеся к каждому из блоков, для генерации  
 25 декодированного изображения.

6. Другие операции устройства кодирования изображения и устройства для декодирования изображения

При выполнении описанных выше операций в устройстве для кодирования изображения и в устройстве для декодирования изображения, параметры квантования  
 30 кодированных блоков, соседствующих в пространстве или во времени с блоком, подлежащим кодированию, принимают в качестве кандидатов для отбора. Кроме того, параметр квантования, выбранный из совокупности кандидатов для отбора в соответствии с параметром квантования, заданным для блока, подлежащего кодированию, принимают в качестве прогнозного параметра квантования. Далее,  
 35 эффективность кодирования параметров квантования повышается посредством включения, в состав информации потока, идентификационной информации для выбора прогнозного параметра квантования из совокупности кандидатов для отбора и разностной информации, обозначающей разность между прогнозным параметром квантования и параметром квантования, заданным для блока, подлежащего  
 40 кодированию.

Однако совокупность кандидатов для отбора не ограничивается параметрами квантования для кодированных блоков, соседствующих в пространстве или во времени с блоком, подлежащим кодированию, так что в совокупность кандидатов для отбора может быть включен также параметр квантования, обновленный последним по времени.  
 45 Как будет описано позднее, даже если для некоторого блока кодированные блоки, соседствующие с этим блоком в пространстве или во времени, не используют обратного квантования, параметр квантования для блока, расположенного рядом с блоком, подлежащим кодированию, может быть задан в качестве прогнозного параметра

квантования. Далее, параметры квантования могут представлять собой неявно или явно прогнозируемые параметры квантования для кандидатов для отбора, на основе которых можно генерировать разностную информацию, обозначающую разность между прогнозируемыми параметрами квантования и параметром квантования блока, подлежащего кодированию.

Далее будет дано описание для случая минимального размера единицы параметров квантования (MinQpUnitSize) в соответствии с размером единицы кодирования и неявного или явного выбора прогнозного параметра квантования среди параметров квантования для совокупности кандидатов для отбора в качестве другой операции устройства для кодирования изображения и устройства для декодирования изображения. Отметим, что ниже будет дано описание, относящееся к частям, отличным от устройства для кодирования изображения и устройства для декодирования изображения, описанным выше.

В случае неявного или явного выбора прогнозного параметра квантования среди параметров квантования для совокупности кандидатов для отбора, устройство для кодирования изображения включает информацию различия "qr\_explicit\_flag", обозначающую явный или неявный способ определения параметров квантования. Кроме того, алгоритм работы устройства для кодирования изображения и устройства для декодирования изображения может быть построен так, чтобы информацию с указанием способа неявного или явного определения параметров квантования декодировать предварительно.

Под неявным определением параметров квантования понимают способ выбора прогнозного параметра квантования, равного параметру, выбранному в устройстве для кодирования изображения, в устройстве для декодирования изображения без передачи идентификационной информации для выбора прогнозируемых параметров квантования из совокупности кандидатов для отбора от устройства для кодирования изображения в устройство для декодирования изображения. В частности, есть способ выбора параметра квантования из совокупности кандидатов для отбора на основе порядка приоритетности, определенного заранее, и затем определения прогнозного параметра квантования, способ принятия стохастической величины параметров квантования для совокупности кандидатов для отбора в качестве прогнозного параметра квантования, способ взвешивания параметров квантования для совокупности кандидатов для отбора в зависимости от расстояния до текущего блока и принятия стохастической величины взвешенных параметров квантования в качестве прогнозного параметра квантования или аналогичный способ.

Под явным определением параметров квантования понимают способ выбора прогнозного параметра квантования, равного параметру, выбранному в устройстве для кодирования изображения, в устройстве для декодирования изображения посредством передачи идентификационной информации для выбора прогнозируемых параметров квантования из совокупности кандидатов для отбора от устройства для кодирования изображения в устройство для декодирования изображения. В частности, имеется способ вычисления индексной информации, определяющей кандидата для отбора, в устройстве для кодирования изображения и включения этой информации в информацию потока, после чего используют параметр квантования выбранного кандидата, указанного в индексной информации, в качестве прогнозного параметра квантования в устройстве для декодирования изображения, способ, когда индексная информация не включена для блоков, к которым квантование не применялось, или другой подобный способ.

Фиг. 23 представляют логическую схему для описания других операций устройства для декодирования изображения, иллюстрирующую кодирующую обработку среза.

На этапе ST141 устройство 10 для кодирования изображения определяет, имеется ли какая-либо единица CU кодирования, подлежащая кодированию. Если в срезе, подлежащем кодирующей обработке, имеется какая-либо единица кодирования, которая еще не была кодирована, устройство 10 для кодирования изображения переходит к этапу ST142. Если кодирующая обработка завершена уже для всех единиц кодирования в составе среза, устройство 10 для кодирования изображения заканчивает кодирующую обработку этого среза.

На этапе ST142 устройство 10 для кодирования изображения разбивает единицу CU кодирования. Это устройство 10 для кодирования изображения разбивает единицу CU кодирования, как показано на Фиг. 5, определяет размер единицы кодирования, для которой величина целевой функции мала, и переходит к этапу ST143. Кроме того, чтобы можно было определить размер единицы кодирования, где величина целевой функции мала, устройство 10 для кодирования изображения включает в информацию потока, например, в параметры "Coding tree syntax", параметр "split\_coding\_unit\_flag", эквивалентный флагу разбиения, показанному на Фиг. 5.

На этапе ST143 устройство 10 для кодирования изображения определяет, выполняется ли обратное квантование применительно к единице кодирования. Если единица CU кодирования, подлежащая кодированию, представляет собой блок в режиме, в котором не требуется обратное квантование с использованием параметров квантования, чтобы осуществить декодирование, например, блок в режиме пропуска, блок в режиме I\_PCM или блок в прямом режиме (параметр «структура кодированных блоков» (CBP (Coded Block Pattern)) = 0), устройство 10 для кодирования изображения возвращается к этапу ST141, а в случае блока, для которого выполняют обратное квантование, переходит к этапу ST144.

На этапе ST144 устройство 10 для кодирования изображения определяет, является ли размер единицы CU кодирования равным "log2MinQpUnitSize" или больше. Если размер единицы CU кодирования равен "log2MinQpUnitSize" или более, устройство 10 для кодирования изображения переходит к этапу ST145. Кроме того, если размер единицы CU кодирования меньше "log2MinQpUnitSize", устройство 10 для кодирования изображения переходит к этапу ST152.

На этапе ST145 устройство 10 для кодирования изображения определяет параметр QR квантования для единицы CU кодирования, подлежащей кодированию. Указанный модуль 18 управления частотой квантования в составе устройства 10 для кодирования изображения определяет параметр квантования в соответствии со степенью сложности изображения единицы кодирования, как описано выше, или так, чтобы величина целевой функции была мала, и переходит к этапу ST146.

На этапе ST146 устройство 10 для кодирования изображения определяет, равен ли "1" параметр информации различения "qr\_explicit\_flag", позволяющий идентифицировать, следуют ли параметры квантования прогнозировать неявным или явным способом. Если параметр "qr\_explicit\_flag" информации различения равен "1", так что параметры квантования следует прогнозировать в явном виде, устройство 10 для кодирования изображения переходит к этапу ST147. Кроме того, если параметр "qr\_explicit\_flag" информации различения равен "0", так что параметры квантования следует прогнозировать в неявном виде, устройство 10 для кодирования изображения переходит к этапу ST149. Это устройство 10 для кодирования изображения сравнивает величину целевой функции в случае, когда параметр "qr\_explicit\_flag" информации различения



задан равным "1", и величину целевой функции, когда этот параметр "qr\_explicit\_flag" информации различения задан равным "0", например. Устройство 10 для кодирования изображения задает величину параметра "qr\_explicit\_flag" информации различения таким образом, чтобы сделать эффективность кодирования выше, на основе результатов

5 сравнения. Кроме того, если пользователь может сам задать параметр "qr\_explicit\_flag" информации различения, устройство 10 для кодирования изображения задает этот параметр "qr\_explicit\_flag" информации различения в соответствии с командой пользователя.

На этапе ST147 устройство 10 для кодирования изображения генерирует

10 идентификационную информацию. Это устройство 10 для кодирования изображения выбирает кандидата из совокупности кандидатов для отбора таким образом, чтобы его разность относительно параметра квантования для единицы кодирования, подлежащей кодированию, была наименьшей, в модуле 19 генерации информации, как описано выше, и принимает выбранного кандидата в качестве прогнозного параметра

15 квантования. Устройство 10 для кодирования изображения принимает, например, параметры квантования для кодированных блоков, соседствующих в пространстве или во времени с блоком, подлежащим кодированию, последний по времени обновленный параметр квантования и процедуры обработки данных, заданные в блоке заголовка среза, в качестве кандидатов для отбора. Это устройство 10 для кодирования

20 изображения выбирает одного кандидата из совокупности кандидатов для отбора, для которого разность относительно параметра квантования для единицы кодирования, подлежащей кодированию, является наименьшей, и принимает выбранного кандидата в качестве прогнозного параметра квантования. Далее, модуль 19 генерации информации принимает индекс (ref\_qr\_block\_index) выбранного кандидата в качестве

25 идентификационной информации для выбора прогнозного параметра квантования из совокупности кандидатов для отбора и переходит к этапу ST148.

На этапе ST148 устройство 10 для кодирования изображения включает идентификационную информацию в составе информации потока. Это устройство 10 для кодирования изображения включает идентификационную информацию,

30 сформированную на этапе ST147, и переходит к этапу ST150.

На этапе ST149 устройство 10 для кодирования изображения определяет прогнозный параметр dQP квантования неявным способом. Иными словами, устройство 10 для кодирования изображения прогнозирует параметр квантования таким же способом, как и устройство 50 для декодирования изображения. Что касается способа

35 прогнозирования параметра квантования, прогнозируемые параметры квантования декодируют на основе порядка приоритетности, определенного заранее, например. Кроме того, в качестве прогнозного параметра квантования может быть принята стохастическая величина для нескольких параметров-кандидатов квантования. Кроме того, может быть использован способ взвешивания параметров квантования для

40 совокупности кандидатов для отбора в соответствии с расстоянием от текущего блока и принятия стохастической величины взвешенных параметров квантования в качестве прогнозного параметра квантования, или другой подобный способ. Это устройство 10 для кодирования изображения вычисляет прогнозный параметр квантования и переходит к этапу ST150.

На этапе ST150 устройство 10 для кодирования изображения генерирует разностную информацию. Это устройство 10 для кодирования изображения вычисляет разность между прогнозным параметром квантования, обозначенным идентификационной информацией, сформированной на этапе ST147, и параметром квантования,

определенным на этапе ST145, или разность между прогнозным параметром квантования, определенным на этапе ST149, и параметром квантования, определенным на этапе ST145. Устройство 10 для кодирования изображения генерирует разностную информацию, обозначенную вычисленной разностью, и переходит к этапу ST151.

5 На этапе ST151 устройство 10 для кодирования изображения включает разностную информацию и информацию различения в информацию потока. Это устройство 10 для кодирования изображения включает в информацию потока разностную информацию, сформированную на этапе ST150, и информацию различения "qr\_explicit\_flag",  
10 указанную информацию различения в один из наборов параметров, например, набор параметров последовательности, набор параметров изображения, заголовок среза или другой подобный набор и переходит к этапу ST152.

На этапе ST152 устройство 10 для кодирования изображения осуществляет квантование единицы CU кодирования. Это устройство 10 для кодирования изображения  
15 выполняет квантование единицы кодирования с использованием найденного параметра квантования и возвращается к этапу ST141.

Фиг. 24 представляет пример операций для случая неявного прогнозирования параметров квантования, а Фиг. 25, иллюстрирует пример логической схемы для случая неявного прогнозирования параметров квантования. Отметим, что здесь для упрощения  
20 описания рассмотрен случай трех кандидатов для отбора.

Как показано на Фиг. 24 (А), параметр квантования для блока, подлежащего кодированию, в кадре, подлежащем кодированию, обозначен, например, "QP\_0". Кроме того показаны три кандидата, а именно параметр "QP\_A" квантования для  
25 кодированного блока, соседствующего слева, параметр "QP\_B" квантования для кодированного блока, соседствующего сверху, и параметр "QP\_LS" квантования для декодированной единицы кодирования.

Как показано на Фиг. 25, на этапе ST161 устройство 10 для кодирования изображения определяет, можно ли использовать параметры "QP\_A" и "QP\_B" квантования в качестве опоры. Если кодированный блок, соседствующий слева, и кодированный блок,  
30 соседствующий сверху, не являются блоками в режиме, в котором не требуется обратное квантование с использованием параметров квантования, чтобы осуществить декодирование, например, блок в режиме пропуска, блок в режиме I\_PCM или блок в прямом режиме (параметр CBP (Coded Block Pattern) = 0), это устройство 10 для кодирования изображения определяет, что использование в качестве опоры возможно,  
35 и переходит к этапу ST162. Кроме того, если по меньшей мере один из параметров – параметр "QP\_A" квантования и параметр "QP\_B" квантования, относится к режиму, для которого не требуется обратное квантование, устройство переходит к этапу ST163.

На этапе ST162 устройство 10 для кодирования изображения принимает среднюю величину параметров "QP\_A" и "QP\_B" квантования в качестве прогнозного параметра  
40 dQP квантования. Иными словами, как показано на Фиг. 24 (В), если можно использовать параметры "QP\_A" и "QP\_B" квантования в качестве опоры, тогда в качестве прогнозного параметра dQP квантования принимают среднюю величину параметров "QP\_A" и "QP\_B" квантования, т.е.  $(QP\_A + QP\_B + 1) / 2$ .

На этапе ST163 устройство 10 для кодирования изображения определяет, можно ли  
45 использовать параметр "QP\_A" квантования в качестве опоры. Если кодированный блок, соседствующий слева, не относится к режиму, для которого не требуется обратное квантование, устройство 10 для кодирования изображения определяет, что этот блок можно использовать в качестве опоры, и переходит к этапу ST164. Кроме того, если

кодированный блок, соседствующий слева, относится к режиму, для которого не требуется обратное квантование, устройство 10 для кодирования изображения определяет, что этот блок нельзя использовать в качестве опоры и переходит к этапу ST165.

5 На этапе ST164 устройство 10 для кодирования изображения принимает параметр "QP\_A" квантования в качестве прогнозного параметра dQP квантования. Иными словами, если параметр "QP\_A" квантования можно использовать в качестве опоры, а параметр "QP\_B" квантования использовать в качестве опоры нельзя, как показано на Фиг. 24 (C), тогда параметр "QP\_A" квантования принимают в качестве прогнозного  
10 параметра dQP квантования. Отметим, что на Фиг. 24 и на описываемом позднее Фиг. 26 блоки в режиме, в котором обратное квантование не требуется, т.е. блоки, которые нельзя использовать в качестве опоры, обозначены штриховкой.

На этапе ST165 устройство 10 для кодирования изображения определяет, можно ли использовать параметр "QP\_B" квантования в качестве опоры. Если кодированный  
15 блок, соседствующий сверху, не относится к режиму, для которого не требуется обратное квантование, устройство 10 для кодирования изображения определяет, что этот можно использовать в качестве опоры, и переходит к этапу ST166. Кроме того, если кодированный блок, соседствующий сверху, относится к режиму, для которого не требуется обратное квантование, устройство 10 для кодирования изображения  
20 определяет, что этот блок нельзя использовать в качестве опоры, и переходит к этапу ST167.

На этапе ST166 устройство 10 для кодирования изображения принимает параметр "QP\_B" квантования в качестве прогнозируем параметра dQP квантования. Иными словами, если параметр "QP\_B" квантования можно использовать в качестве опоры, а  
25 параметр "QP\_A" квантования нельзя использовать в качестве опоры, как показано на Фиг. 24 (D), тогда параметр "QP\_B" квантования принимают в качестве прогнозного параметра dQP квантования.

На этапе ST167 устройство 10 для кодирования изображения принимает параметр "QP\_LS" квантования в качестве прогнозного параметра dQP квантования. Как показано  
30 на Фиг. 24 (E), если кодированный блок, соседствующий слева, и кодированный блок, соседствующий сверху, относятся к режиму, в котором обратное квантование не требуется, тогда параметр "QP\_LS" квантования принимают в качестве прогнозного параметра dQP квантования.

Фиг. 26 иллюстрирует другой пример операций для случая неявного прогнозирования  
35 параметров квантования. Например, как показано на Фиг. 24 (E), если кодированный блок, соседствующий слева, и кодированный блок, соседствующий сверху, относятся к режиму, для которого не требуется обратное квантование, заданный параметр квантования может быть сформирован путем увеличения числа кандидатов для отбора. Например, как показано на Фиг. 26 (A), к совокупности кандидатов для отбора могут  
40 быть добавлены параметр "QP\_C" квантования кодированного блока, соседствующего сверху справа, параметр "QP\_D" квантования кодированного блока, соседствующего сверху слева, и параметр "QP\_E" квантования кодированного блока, соседствующего снизу слева.

Если параметры "QP\_C", "QP\_D" и "QP\_E" квантования можно использовать в  
45 качестве опоры, как показано на Фиг. 26 (B), устройство 10 для кодирования изображения принимает среднюю величину  $(QP\_C + QP\_D + 1) / 2$  параметров "QP\_C" и "QP\_D" квантования или медианное значение в качестве прогнозного параметра dQP квантования.

Если параметры "QP\_C" и "QP\_D" квантования можно использовать в качестве опоры, как показано на Фиг. 26 (C), устройство 10 для кодирования изображения принимает среднюю величину  $(QP\_C + QP\_D + 1) / 2$  параметров "QP\_C" и "QP\_D" квантования в качестве прогнозного параметра dQP квантования.

5 Если параметры "QP\_D" и "QP\_E" квантования можно использовать в качестве опоры, как показано на Фиг. 26 (D), устройство 10 для кодирования изображения принимает среднюю величину  $(QP\_D + QP\_E + 1) / 2$  параметров "QP\_D" и "QP\_E" в качестве прогнозного параметра dQP квантования.

10 Если параметры "QP\_C" и "QP\_E" квантования можно использовать в качестве опоры, как показано на Фиг. 26 (E), устройство 10 для кодирования изображения принимает среднюю величину  $(QP\_C + QP\_E + 1) / 2$  параметров "QP\_C" и "QP\_E" в качестве прогнозного параметра dQP квантования.

15 Если параметры "QP\_C", "QP\_D" и "QP\_E" квантования нельзя использовать в качестве опоры, как показано на Фиг. 26 (F), устройство 10 для кодирования изображения принимает параметр "QP\_LS" квантования в качестве прогнозного параметра dQP квантования. Отметим, что Фиг. 27 иллюстрирует программу для выполнения операций (B) – (D) на Фиг. 24 и (B) – (F) на Фиг. 26.

20 Кроме того, если число параметров квантования, которые можно использовать в качестве опоры, равно единице, как показано на Фиг. 26 (G) – (I), этот параметр квантования, который можно использовать в качестве опоры, может быть использован в качестве прогнозного параметра dQP квантования.

25 Таким образом, устройство 10 для кодирования изображения принимает параметры квантования кодированных блоков, соседствующих в пространстве или во времени с блоком, подлежащим кодированию, в качестве кандидатов для отбора и выбирает прогнозный параметр квантования из совокупности кандидатов для отбора в соответствии с заданным параметром квантования. Кроме того, устройство 10 для кодирования изображения генерирует идентификационную информацию для выбора прогнозного параметра квантования из совокупности кандидатов для отбора. Далее, это устройство 10 для кодирования изображения генерирует разностную информацию, 30 обозначающую разность между прогнозным параметром квантования и параметром квантования, заданным для блока, подлежащего кодированию. Устройство 10 для кодирования изображения включает сформированную им идентификационную информацию и разностную информацию в информацию потока. В результате такой обработки можно не допустить, чтобы разность между параметром квантования блока, 35 подлежащего кодированию, и прогнозным параметром квантования стала большой. Соответственно, устройство 10 для кодирования изображения может повысить эффективность кодирования параметров квантования.

40 Кроме того, в случае неявного прогнозирования параметров квантования, прогнозный параметр квантования, равный параметру квантования, найденному в устройстве 10 для кодирования изображения, может быть использован в устройстве 50 для декодирования изображения без включения идентификационной информации для выбора прогнозируемых параметров квантования из совокупности кандидатов для отбора в информацию потока. Более того, включение информации различия в информацию потока позволит адаптивно переключаться между режимом явного 45 прогнозирования необходимых прогнозируемых параметров квантования и режимом неявного прогнозирования необходимых прогнозируемых параметров квантования.

Фиг. 28 представляют логическую схему для описания других операций устройства для декодирования изображения. На этапе ST127, показанном на Фиг. 22, если

определено, что имеется единица кодирования, подлежащая декодированию, устройство 50 для декодирования изображения выполняет обработку данных, начиная с этапа ST171, и осуществляет декодирование этой единицы кодирования.

На этапе ST171 устройств 50 для декодирования изображения выделяет информацию. Это устройство 50 для декодирования изображения выделяет информацию из информации потока с целью использования выделенной информации для декодирования указанной единицы кодирования. Например, извлекает информацию "Coding tree syntax" («Синтаксис дерева кодирования»), позволяющую определить размер единицы кодирования, информацию "log2\_min\_qp\_unit\_size\_offset", позволяющую определить минимальный размер единицы параметра квантования, информацию "qp\_explicit\_flag" различения и другую подобную информацию и переходит к этапу ST172.

На этапе ST172 устройство 50 для декодирования изображения разбивает рассматриваемую единицу CU кодирования. Это устройство 50 для декодирования изображения разбивает единицу CU кодирования на основе параметра "split\_coding\_unit\_flag" и другой соответствующей информации, включенной в информацию потока, и переходит к этапу ST173.

На этапе ST173 устройство 50 для декодирования изображения определяет, использует ли единица CU кодирования, подлежащая декодированию, обратное квантование. Если единица CU кодирования, подлежащая декодированию, относится к режиму, в котором осуществляется обратное квантование с использованием параметров квантования, устройство 50 для декодирования изображения переходит к этапу ST174, а если это блок, для которого обратное квантование с использованием параметра квантования не требуется, процесс завершается.

На этапе ST174 устройство 50 для декодирования изображения определяет, является ли размер единицы CU кодирования равным или больше "log2MinQpUnitSize". Если размер единицы CU кодирования равен или больше "log2MinQpUnitSize", устройство 50 для декодирования изображения переходит к этапу ST175. Кроме того, если размер единицы CU кодирования меньше "log2MinQpUnitSize", устройство 50 для декодирования изображения переходит к этапу ST180.

На этапе ST175 устройство 50 для декодирования изображения определяет, равен ли "1" параметр "qp\_explicit\_flag" информации различения. Если этот параметр "qp\_explicit\_flag" информации различения в составе информации потока равен "1", а параметры квантования следует прогнозировать явным способом, устройство 50 для декодирования изображения переходит к этапу ST176. Кроме того, если параметр "qp\_explicit\_flag" информации различения равен "0", а параметры квантования следует прогнозировать неявным способом, устройство 50 для декодирования изображения переходит к этапу ST178.

На этапе ST176 устройство 50 для декодирования изображения выделяет индекс (ref\_qp\_block\_index) из информации потока и переходит к этапу ST177.

На этапе ST177 устройство 50 для декодирования изображения определяет прогнозный параметр dQP квантования. Это устройство 50 для декодирования изображения выбирает параметр квантования на основе индекса (ref\_qp\_block\_index) среди параметров квантования из совокупности кандидатов для отбора, как и устройство 10 для кодирования изображения, принимает выбранный параметр квантования в качестве прогнозного параметра dQP квантования и переходит к этапу ST179.

На этапе ST178 устройство 50 для декодирования изображения неявным способом определяет прогнозный параметр dQP квантования. Это устройство 50 для декодирования изображения определяет прогнозный параметр квантования таким же

способом, как в устройстве 10 для кодирования изображения. Что касается способа прогнозирования параметра квантования, определить параметр квантования можно на основе порядка приоритетности, выбранного заранее, например. Кроме того, в качестве прогнозного параметра квантования можно принять стохастическую величину для параметров квантования. Далее, можно использовать способ, согласно которому взвешивают параметры квантования для совокупности кандидатов для отбора в соответствии с расстоянием от текущего блока и принимают стохастическую величину для взвешенных параметров квантования в качестве прогнозного параметра квантования, или другой аналогичный способ. Устройство 50 для декодирования изображения прогнозирует параметр квантования и переходит к этапу ST179.

На этапе ST179 устройство 50 для декодирования изображения вычисляет параметр квантования QR для текущей единицы CU кодирования. Это устройство 50 для декодирования изображения получает разностную информацию "qb\_qr\_delta" из информации потока, суммирует эту разностную информацию с прогнозным параметром dQR квантования, вычисляет параметр квантования для единицы кодирования, подлежащей декодированию, и переходит к этапу ST180.

На этапе ST180 устройство 50 для декодирования изображения осуществляет обратное квантование обрабатываемой единицы кодирования. Это устройство 50 для декодирования изображения осуществляет обратное квантование рассматриваемой единицы кодирования с использованием декодированного параметра квантования.

Соответственно, устройство 50 для декодирования изображения может осуществить декодирование изображений с использованием параметров квантования, равных параметрам квантования, использованным в устройстве для кодирования изображения.

#### 7. Случай программной обработки

Последовательность операций обработки данных, описанных выше, может быть выполнена с использованием аппаратной конфигурации, программной конфигурации или обеих конфигураций. Если обработка данных осуществляется посредством программного обеспечения, программу, в которой записана последовательность обработки данных, инсталлируют в памяти компьютера, встроенного в специализированную аппаратуру, и выполняют. В альтернативном варианте, программа может быть инсталлирована в компьютере общего назначения, способном выполнять различные виды обработки.

Фиг. 29 представляет схему, иллюстрирующую пример упрощенной конфигурации компьютера, выполняющего описанную выше последовательность операций обработки данных в соответствии с программой. Центральный процессор CPU 801 компьютера 80 выполняет различные виды обработки данных в соответствии с программой, записанной в постоянном запоминающем устройстве (ПЗУ (ROM)) 802 или в записывающем устройстве 808.

Программы, выполняемые процессором CPU 801, данные и прочую информацию записывают в запоминающем устройстве с произвольной выборкой (ЗУПВ (RAM)) 803 по мере необходимости. Процессор CPU 801, ПЗУ ROM 802, и ЗУПВ RAM 803 соединены одно с другим шиной 804.

Интерфейс 805 ввода/вывода также соединен с процессором CPU 801 посредством шины 804. С процессором CPU 801 соединены также модуль 806 ввода, такой как сенсорная панель, клавиатура, мышь, микрофон или другое подобное устройство, и модуль 807 вывода, образованный дисплеем или аналогичным устройством. Процессор CPU 801 выполняет различные виды обработки данных в соответствии с командами, поступающими от модуля 806 ввода. Процессор CPU 801 затем передает результаты

обработки данных модулю 807 вывода.

Записывающее устройство 808, соединенное с интерфейсом 805 ввода/вывода, построено на основе, например, жесткого диска и записывает программы, выполняемые процессором CPU 801, и различного рода данные. Модуль 809 связи поддерживает

5 связь с внешними устройствами по кабельным или беспроводным линиям связи, таким как линии сетей связи типа Интернет, или локальных сетей связи, или систем цифрового вещания, или других подобных систем. Кроме того, компьютер 80 может получать программы через модуль 809 связи и записывать их в ПЗУ ROM 802 или в запоминающем устройстве 808.

10 Если сменный носитель 85 записи, такой как магнитный диск, оптический диск, магнитооптический диск или полупроводниковое запоминающее устройство или другой подобный носитель, установить в приводе накопителя 810, с него можно получить программы, данные или подобную информацию, записанные на носителе. Полученные программы и данные переписывают в ПЗУ ROM 802 или в ЗУПВ RAM 803 или в

15 записывающее устройство 808 по мере необходимости.

Процессор CPU 801 считывает и исполняет программу для осуществления описанной выше последовательности операций и производит кодирующую обработку сигналов изображения, записанных в записывающем устройстве 808 или на сменном носителе 85 записи, либо сигналов изображения, поступающих через модуль 809 связи, или

20 декодирующую обработку информации потока.

#### 8. Случай применения к электронному оборудованию

Кроме того, в изложенном выше описании в качестве формата кодирования/декодирования использован формат H.264/AVC, но предлагаемая технология может быть также применена в устройстве для кодирования изображения / устройстве для

25 декодирования изображения, использующих формат кодирования/декодирования, предполагающий другой способ обработки данных для прогнозирования/компенсации движения.

Более того, предлагаемая технология может быть применена в устройстве для кодирования изображения и устройстве для декодирования изображения, используемых

30 во время приема информации потока, полученной путем кодирующей обработки, такой как MPEG, H.26x или другой подобной обработки, через сеть связи, такую как сеть спутникового вещания, сеть кабельного телевидения, Интернет, сеть сотовой телефонной связи, или во время обработки на носителе записи, таком как оптический диск или магнитный диск или флэш-память.

35 Далее будет дано описание электронного устройства, в котором применены описанные выше устройство 10 для кодирования изображения и устройство 50 для декодирования изображения.

Фиг. 30 представляет схему, иллюстрирующую пример упрощенной конфигурации телевизионного устройства, в котором применена предлагаемая технология.

40 Телевизионное устройство 90 содержит антенну 901, тюнер 902, демультимплексор 903, декодер 904, модуль 905 обработки видеосигнала, модуль 906 дисплея, модуль 907 обработки аудио сигнала, громкоговоритель 908 и модуль 909 внешнего интерфейса. Кроме того, телевизионное устройство содержит модуль 910 управления, модуль 911 интерфейса пользователя или другой подобный модуль.

45 Тюнер 902 осуществляет демодуляцию путем выбора сигнала нужного канала из совокупности сигналов вещания, принимаемых антенной 901, и передает полученный поток данных в демультимплексор 903.

Этот демультимплексор 903 выделяет пакеты данных аудио и видео составляющих

программы, выбранной для просмотра, из потока и передает данные выделенных пакетов в декодер 904. Кроме того, демультимплексор 903 передает пакеты данных, такие как электронная программа передач (EPG (Electronic Program Guide)), в модуль 910 управления. Отметим, что если было выполнено скремблирование, тогда дескремблирование осуществляется в демультимплексоре или в другом подходящем устройстве.

Декодер 904 осуществляет декодирование пакетов и передает видеоданные, полученные в результате декодирования, в модуль 905 обработки видеосигнала, и аудиоданные в модуль 907 обработки аудио сигнала.

Модуль 905 обработки видеосигнала обрабатывает видеоданные в соответствии с требованиями подавления шумов и настройками пользователя. Этот модуль 905 обработки видеосигнала генерирует видеоданные для представления программ на экране модуля 906 дисплея и данные изображения, обработанные в соответствии с приложениями, поступающими через сеть связи. Кроме того, модуль 905 обработки видеосигнала генерирует видеоданные для представления экранных меню или другой подобной информации и накладывает эти видеоданные на видеоданные просматриваемой программы. Указанный модуль 905 обработки видеосигнала генерирует сигналы управления на основе видеоданных, сформированных таким способом, и управляет модулем 906 дисплея.

Модуль 906 дисплея управляет дисплейным устройством (например, жидкокристаллическим или иным подобным дисплейным устройством) на основе драйверного сигнала от модуля 905 обработки видеосигнала для представления видео составляющей программы на дисплее.

Модуль 907 обработки аудио сигнала осуществляет заданную обработку аудиоданных, такую как подавление шумов, и передает аудио сигнал на выход громкоговорителю 908 после осуществления цифро-аналогового преобразования аудиоданных и усиления полученного сигнала.

Модуль 909 внешнего интерфейса представляет собой интерфейс для соединения с внешним оборудованием или сетью и осуществляет передачу и прием данных, таких как видеоданные или аудиоданные.

Модуль 911 интерфейса пользователя соединен с модулем 910 управления. Этот модуль 911 интерфейса пользователя конфигурирован в составе переключателя режимов работы или приемника сигналов дистанционного управления или другого подобного устройства и передает сигналы управления режимами работы в соответствии с действиями пользователя в модуль 910 управления.

Модуль 910 управления конфигурирован с использованием центрального процессора CPU (Central Processing Unit), запоминающего устройства или другого аналогичного устройства. Запоминающее устройство сохраняет программы, которые должен выполнять процессор CPU, и различные данные, необходимые процессору CPU для выполнения этих программ, данные электронной программы передач (EPG), данные, получаемые через сеть связи, и другую подобную информацию. Программу, сохраняемую в запоминающем устройстве, процессор CPU считывает в заданные моменты времени, например при включении телевизионного устройства 90, и выполняет эту программу. Выполняя программу, процессор CPU управляет работой каждой части телевизионного устройства 90, так что это устройство работает в соответствии с действиями пользователя.

Отметим, что в телевизионном устройстве 90 имеется шина 912, соединяющая тюнер 902, демультимплексор 903, модуль 905 обработки видеосигнала, модуль 907 обработки



аудио сигнала, модуль 909 внешнего интерфейса и модуль 910 управления.

В имеющем такую конфигурацию телевизионном устройстве функция устройства для декодирования изображения (способ декодирования изображения) отдана декодеру 904. Поэтому, даже если на стороне вещательной станции была произведена кодирующая  
5 обработка изображения с целью уменьшить объем кода, необходимый для передачи параметров квантования, это телевизионное устройство может правильно восстановить параметры квантования и сформировать декодированное изображение.

Фиг. 31 представляет схему, иллюстрирующую пример упрощенной конфигурации сотового телефона, в котором была применена предлагаемая технология. Сотовый  
10 телефон 92 содержит модуль 922 связи, аудио кодек 923, модуль 926 видеокамеры, модуль 927 обработки изображения, модуль 928 разделения мультимплексированных данных, модуль 929 воспроизведения/записи, модуль 930 дисплея и модуль 931 управления. Эти модули соединены один с другими посредством шины 933 данных.

Кроме того, антенна 921 соединена с модулем 922 связи, и громкоговоритель 924 и  
15 микрофон 925 соединены с аудио кодеком 923. Кроме того, с модулем 931 соединен операционный модуль 932.

Сотовый телефон 92 выполняет различные операции, такие как передачу и прием аудио сигналов, передачу и приема сообщений электронной почты и данных изображения, съемку изображения, запись данных и другие подобные операции, в  
20 различных режимах, таких как режим голосового вызова или режим передачи данных.

В режиме голосового вызова аудио сигналы, генерируемые микрофоном 925, преобразуют в аудиоданные и сжимают эти данные в аудио кодеке 923, после чего передают в модуль 922 связи. Этот модуль 922 связи осуществляет модуляционную  
25 обработку аудиоданных и преобразование частоты аудиоданных для генерации передаваемых сигналов. Кроме того, модуль 922 связи направляет передаваемые сигналы в антенну 921 для передачи на неизвестную базовую станцию. Кроме того, модуль 922 связи осуществляет усиление, преобразование частоты и демодуляционную обработку сигналов, принимаемых антенной 921, и передает полученные в результате аудиоданные в аудио кодек 923. Этот аудио кодек 923 расширяет аудиоданные и  
30 преобразует в аналоговые аудио сигналы, после чего передает эти аудио сигналы громкоговорителю 924.

Кроме того, в режиме передачи данных, в случае передачи сообщений электронной почты модуль 931 управления принимает текстовые данные, вводимые посредством  
35 операционного модуля 932, и представляет вводимый текст на экране дисплейного модуля 930. Кроме того, модуль 931 управления генерирует данные электронной почты на основе команд пользователя, вводимых через операционный модуль 932, и передает в модуль 922 связи. Этот модуль 922 связи осуществляет модуляционную обработку, преобразование частоты и другую необходимую обработку данных электронной почты и передает полученные передаваемые сигналы через антенну 921. Кроме того, модуль  
40 922 связи осуществляет усиление, преобразование частоты и демодуляционную обработку сигналов, принимаемых антенной 921, и восстанавливает данные электронной почты. Эти данные электронной почты передают дисплейному модулю 930 для представления содержания сообщения электронной почты на экране дисплея.

Отметим, что сотовый телефон 92 может сохранять принятые данные электронной  
45 почты на носителе записи в модуле 929 записи/воспроизведения. В качестве носителя записи можно использовать любой носитель записи, на котором можно записывать/считывать информацию. Например, носитель записи может представлять собой полупроводниковое запоминающее устройство, такое как ЗУПВ RAM или встроенная

флэш-память, сменный носитель записи, такой как жесткий диск, магнитный диск, магнитооптический (МО) диск, оптический диск, USB-память, карту памяти или другой подобный носитель.

В случае передачи данных изображения в режиме передачи данных, такие данные изображения, генерируемые модулем 926 видеокамеры, передают в модуль 927 обработки изображения. Этот модуль 927 обработки изображения осуществляет кодирование данных изображения и генерирует информацию потока.

Модуль 928 разделения мультимплексированных данных осуществляет мультимплексирование информации потока, генерируемой в модуле 927 обработки изображения, и аудиоданных, поступающих от аудио кодека 923, в заданном формате и передает в модуль 922 связи. Этот модуль 922 связи осуществляет модуляцию, преобразование частоты и другую подходящую обработку применительно к мультимплексированным данным и передает полученные в результате передаваемые сигналы через антенну 921. Кроме того, модуль 922 связи осуществляет усиление, преобразование частоты и демодуляцию или другую подобную обработку сигналов, принимаемых антенной 921, и восстанавливает мультимплексированные данные. Эти мультимплексированные данные передают в модуль 928 разделения мультимплексированных данных. Указанный модуль 928 разделения мультимплексированных данных осуществляет разделение мультимплексированных данных и передает информацию потока в модуль 927 обработки изображения, а аудиоданные в аудио кодек 923.

Модуль 927 для обработки изображения осуществляет декодирующую обработку кодированных данных и генерирует данные изображения. Эти данные изображения поступают в модуль 930 дисплея для представления принятого изображения на экране модуля. Аудио кодек 923 преобразует аудиоданные в аналоговые аудио сигналы и передает громкоговорителю 924 для воспроизведения принятой звуковой составляющей.

В сотовом телефоне, сконфигурированном таким образом, модуль 927 для обработки изображения имеет функции согласно настоящей заявке. Соответственно можно уменьшить объем данных во время выполнения кодирующей обработки и передачи изображений, например. Кроме того, в процессе декодирующей обработки принятого изображения можно восстановить параметры квантования и генерировать декодированное изображение.

Фиг. 32 представляет схему, иллюстрирующую пример упрощенной конфигурации устройства записи/воспроизведения, в котором применена предлагаемая технология. Устройство 94 записи/воспроизведения записывает, например, аудиоданные и видеоданные из состава принятой вещательной программы на носителе записи и представляет записанные данные пользователю в моменты времени, указанные командами пользователя. Кроме того, можно построить систему, в которой устройство 94 записи/воспроизведения может, например, получать аудиоданные и видеоданные от других устройств, чтобы записать полученные данные на носителе записи. Более того, можно построить систему, в которой устройство 94 записи/воспроизведения может, посредством декодирования аудиоданных и видеоданных, записанных на носителе записи, передавать изображение для представления на дисплее и аудио сигнал для воспроизведения звука устройствам-мониторам.

Устройство 94 записи/воспроизведения содержит тюнер 941, модуль 942 внешнего интерфейса, модуль 943 кодирования, накопитель 944 на жестком диске (HDD (Hard Disk Drive)), дисковод 945, селектор 946, декодер 947, модуль 948 экранного меню (OSD (On-Screen Display)), модуль 949 управления и модуль 950 интерфейса пользователя.

Тюнер 941 выбирает станцию, передающую сигнал нужного канала, из совокупности сигналов вещания, принимаемых непоказанной антенной. Этот тюнер 941 передает кодированный поток, полученный в результате демодуляции принимаемых сигналов нужного канала, селектору 946.

5 Модуль 942 внешнего интерфейса конфигурирован по меньшей мере на основе одного из следующих интерфейсов – интерфейса IEEE1394, модуля сетевого интерфейса, USB-интерфейса, интерфейса флэш-памяти и или другого подходящего интерфейса. Этот модуль 942 внешнего интерфейса представляет собой интерфейс для соединения с внешним устройством, сетью связи, картой памяти или другим подобным устройством  
10 и осуществляет прием данных, таких как видеоданные и аудиоданные, для записи.

Модуль 943 кодирования осуществляет кодирующую обработку, если видеоданные и аудиоданные, поступающие от модуля 942 внешнего интерфейса, не были кодированы в заданном формате, и передает полученную информацию потока селектору 946.

Накопитель 944 на жестком диске (HDD) записывает данные контента, такие как  
15 видео или аудио, различных программ, другие данные или иную подходящую информацию на встроенном жестком диске, а также считывает эти данные с жесткого диска во время воспроизведения.

Дисковод 945 осуществляет запись или воспроизведение сигналов на установленном оптическом диске. Оптический диск представляет собой DVD-диск (DVD-Video, DVD-  
20 RAM, DVD-R, DVD-RW, DVD+R, DVD+RW или другой подобный диск) или диск Блю-рей или другой подобный диск.

Селектор 946 выбирает, во время записи видео или аудио программ, поток либо от тюнера 941, либо от модуля 943 кодирования и передает его либо накопителю HDD 944, либо дисководу 945. Кроме того, селектор 946 передает, во время воспроизведения  
25 видео и аудио программ, поток с выхода накопителя HDD 944 или дисковода 945 в декодер 947.

Декодер 947 осуществляет декодирование потока. Этот декодер 947 передает видеоданные модулю OSD 948 после выполнения декодирования. Кроме того, декодер 947 передает на выход полученные им в результате декодирования аудиоданные.

30 Модуль OSD 948 генерирует видеоданные для представления экранного меню или другой подобной информации, например, для выбора позиций этого меню, накладывает эти видеоданные на видеоданные от декодера 947 и передает на выход.

Модуль 950 интерфейса пользователя соединен с модулем 949 управления. Этот модуль 950 интерфейса пользователя конфигурирован в составе операционного  
35 переключателя или приемника сигналов дистанционного управления или другого подобного устройства и передает операционные сигналы в ответ на операции пользователя модулю 949 управления.

Модуль 949 конфигурирован с использованием процессора CPU и запоминающего устройства. В запоминающем устройстве сохраняют программу, выполняемую  
40 процессором CPU, и различные данные, необходимые процессору CPU для выполнения обработки. Сохраняемую в памяти программу считывают и выполняют в заданные моменты времени, такие как момент включения устройства 94 записи/воспроизведения, в процессоре CPU. Этот центральный процессор CPU управляет каждой частью таким образом, что устройство 94 записи/воспроизведения работает в соответствии с  
45 операциями пользователя, выполняя некую программу.

В конфигурированном таким образом устройстве записи/воспроизведении функции предлагаемой Заявки отданы модулю 943 кодирования. Поэтому объем данных может быть уменьшен в процессе выполнения кодирующей обработки и записи изображений,

например. Кроме того, в процессе декодирования записанного изображения можно восстановить параметры квантования и сформировать декодированное изображение.

Фиг. 33 представляет схему, иллюстрирующую пример упрощенной конфигурации устройства для считывания изображения, в котором применена настоящая Заявка.

5 Устройство 96 для считывания изображения считывает изображение предмета, чтобы представить изображение этого предмета на экране модуля дисплея и записать это изображение на носителе записи в виде данных изображения.

10 Устройство 96 для считывания изображения содержит оптический блок 961, модуль 962 для считывания изображения, модуль 963 обработки сигнал видеокамеры, модуль 964 обработки данных изображения, модуль 965 дисплея, модуль 966 внешнего интерфейса, модуль 967 памяти, привод 968 носителя записи, модуль 969 экранного меню (OSD) и модуль 970 управления. Кроме того, с модулем 970 управления соединен модуль 971 интерфейса пользователя. Более того, модуль 964 обработки данных изображения и модуль 966 внешнего интерфейса, модуль 967 памяти, привод 968 носителя записи, модуль 969 экранного меню, модуль управления и другие подобные устройства соединены один с другими посредством шины 972 данных.

20 Оптический блок 961 конфигурирован в составе фокусирующей линзы, механизма диафрагмы и других подобных элементов. Оптический блок 961 проецирует оптическое изображение предмета на входную поверхность модуля 962 для считывания изображения. Этот модуль 962 для считывания изображения конфигурирован в составе формирователя сигналов изображения на основе приборов с зарядовой связью (ПЗС (CCD)) или КМОП (CMOS) структур, генерирующего электрические сигналы, соответствующие оптическому изображению, в результате фотоэлектрического преобразования и передающего эти сигналы в модуль 963 обработки сигналов видеокамеры.

25 Этот модуль 963 обработки сигнала видеокамеры осуществляет разнообразную обработку, такую как коррекция изгиба характеристики (KNEE correction) и гамма-коррекция, цветовая коррекция и другая подобная обработка, применительно к электрическим сигналам, поступающим от модуля 962 считывания изображения. Модуль 30 963 для обработки сигналов видеокамеры передает данные изображения после обработки сигналов видеокамеры в модуль 964 обработки данных изображения.

Модуль 964 обработки данных изображения осуществляет кодирующую обработку данных изображения, поступающих от модуля 963 обработки сигналов видеокамеры. Этот модуль 964 обработки данных изображения передает информацию потока, 35 генерируемую в результате кодирующей обработки, в модуль 966 внешнего интерфейса и модулю 968 привода носителя записи. Кроме того, модуль 964 обработки данных изображения осуществляет декодирующую обработку информации потока, поступающей от модуля 966 внешнего интерфейса и модуля 968 привода носителя записи. Модуль 964 обработки данных изображения передает сформированные данные изображения, 40 поступающие от модуля 963 обработки сигналов видеокамеры в модуль 965 дисплея после декодирования. Кроме того, модуль 964 обработки данных изображения осуществляет обработку данных для передачи данных изображения, поступающих от модуля 963 обработки сигналов видеокамеры, в модуль 965 дисплея и обработку данных для наложения данных для представления на дисплее, полученных от модуля 969 экранного меню, на данные изображения и передачи результата в модуль 965 дисплея.

Модуль 969 экранного меню (OSD) генерирует данные для представления на дисплее, такие как экранное меню или иконка, построенная из знаков, текста или фигур, и передает эти данные в модуль 964 для обработки данных изображения.

Например, модуль 966 внешнего интерфейса конфигурирован в составе входного и выходного USB-портов, а в случае выполнения распечатки изображения соединен с принтером. Кроме того, при необходимости с этим модулем 966 внешнего интерфейса может быть соединен привод, на котором установлен подходящий сменный носитель записи, такой как магнитный диск, оптический диск или другой носитель, и программа, записанная на этом носителе, может быть инсталлирована в устройстве, если нужно. Более того, модуль 966 внешнего интерфейса содержит сетевой интерфейс, соединенный с заданной сетью связи, такой как локальная сеть связи LAN или Интернет. Например, по команде от модуля 971 интерфейса пользователя модуль управления 970 считывает информацию потока из модуля 967 памяти, чтобы передать ее другим устройствам, присоединенным через сеть, от модуля 966 внешнего интерфейса. Кроме того, модуль 970 управления получает информацию потока и данные изображения, поступающие от других устройств из сети, через модуль 966 внешнего интерфейса, чтобы передать эти данные в модуль 964 для обработки данных изображения.

Например, в качестве носителя записи, приводимого в действие приводом 968 носителя записи, может быть использован любой сменный носитель записи, на котором можно записывать и считывать информацию, такой как магнитный диск, магнитооптический диск, оптический диск или полупроводниковое запоминающее устройство. Кроме того, что касается носителя записи, тип сменного носителя также может быть разный – это может быть магнитофон, диск или карта памяти. Само собой разумеется, что это может быть бесконтактная карточка с интегральной микросхемой или аналогичное устройство.

Кроме того, можно построить систему, в которой привод 968 носителя записи и носитель записи выполнены заодно и, например, конфигурированы в виде непортативного носителя записи, такого как встроенный накопитель на жестком диске или твердотельный накопитель (SSD (Solid State Drive)) или другое подобное устройство.

Модуль 970 управления конфигурирован с использованием процессора CPU, запоминающего устройства и другого подобного устройства. Запоминающее устройство сохраняет программы, которые должен выполнять процессор CPU, и разного рода данные, необходимые процессору CPU для выполнения обработки данных. Программы, записанные в запоминающем устройстве, считывает процессор CPU в заданные моменты времени, например в момент включения устройства 96 для считывания изображения, и выполняет эти программы. Процессор CPU, выполняя эту программу, управляет каждой частью таким образом, что операции устройства 96 для считывания изображения соответствуют операциям пользователя.

В конфигурированном таким образом устройстве для считывания изображения функции согласно настоящей Заявке отданы модулю 964 обработки данных изображения. Поэтому можно уменьшить объем данных во время кодирования и записи данных изображения в модуле 967 памяти или на носителе записи. Кроме того, в процессе декодирующей обработки записанного изображения можно восстановить параметры квантования и сформировать декодированное изображение.

Более того, не следует интерпретировать предлагаемую технологию, как ограниченную описанными выше вариантами. Эти варианты описаны только в качестве примеров, и совершенно понятно, что специалист в рассматриваемой области может осуществить модификации и замены в этих вариантах, не отклоняясь от существа предлагаемой технологии. Другими словами, для определения сущности предлагаемой технологии следует рассматривать Формулу изобретения.

Кроме того, устройство для декодирования изображения и устройство для

кодирования изображения согласно предлагаемой технологии могут принимать следующие конфигурации.

(1) Устройство декодирования изображения, содержащее:

5 модуль получения информации, конфигурированный для принятия параметров квантования декодированных блоков, соседствующих в пространстве или во времени с блоком, подлежащим декодированию, в качестве кандидатов для отбора и для выделения, из информации потока, разностной информации, обозначающей разность относительно прогнозного параметра квантования, выбранного из совокупности кандидатов для отбора; и

10 модуль вычисления параметра квантования, конфигурированный для вычисления, на основе прогнозного параметра квантования и разностной информации, параметра квантования для блока, подлежащего декодированию.

(2) Устройство декодирования изображения согласно (1), отличающееся тем, что модуль вычисления параметра квантования задает в качестве прогнозного параметра квантования один из параметров квантования в порядке, обозначенном

15 идентификационной информацией, включенной в состав информации потока, где соседствующие декодированные блоки указаны в заданном порядке.

(3) Устройство декодирования изображения согласно (1), отличающееся тем, что модуль вычисления параметра квантования определяет кандидатов для отбора в

20 порядке, заданном заранее, и задает прогнозный параметр квантования на основе результатов такого определения.

(4) Устройство декодирования изображения согласно (1), отличающееся тем, что модуль вычисления параметра квантования выбирает, на основе информации определения, включенной в состав информации потока, выполнение одной из двух

25 процедур обработки данных – процедуры задания в качестве прогнозного параметра квантования одного из параметров квантования в порядке, обозначенном идентификационной информацией, включенной в состав информации потока, или процедуры определения кандидатов для отбора в порядке, заданном заранее, и задания прогнозного параметра квантования на основе результатов определения.

30 (5) Устройство декодирования изображения согласно любому из (1) – (4), отличающееся тем, что модуль вычисления параметра квантования принимает кандидатов для отбора из совокупности соседствующих декодированных блоков, из которой исключены по меньшей мере блоки, где параметры квантования являются избыточными, или блоки, для которых обратное квантование с использованием

35 параметров квантования не проводилось.

(6) Устройство декодирования изображения согласно любому из (1) – (5), отличающееся тем, что, если нет кандидатов для отбора, модуль вычисления параметра квантования принимает параметр квантования, имеющий начальную величину в срезе, в качестве прогнозного параметра квантования.

40 (7) Устройство декодирования изображения согласно любому из (1) – (6), отличающееся тем, что модуль вычисления параметра квантования включает параметр квантования, обновленный самым последним, в совокупность кандидатов для отбора.

(8) Устройство декодирования изображения согласно любому из (1) – (7), отличающееся тем, что модуль вычисления параметра квантования вычисляет параметр квантования для блока, подлежащего декодированию, путем суммирования разности, которую обозначает разностная информация, с прогнозным параметром квантования.

45

(9) Устройство кодирования изображения, содержащее:

модуль управления, конфигурированный с целью задания параметра квантования

для блока, подлежащего кодированию;

модуль генерации информации, конфигурированный для принятия параметров квантования кодированных блоков, соседствующих в пространстве или во времени с блоком, подлежащим кодированию, в качестве кандидатов для отбора, выбора из совокупности кандидатов для отбора одного прогнозного параметра квантования в соответствии с заданным параметром квантования и генерацию разностной информации, обозначающей разность между прогнозированным параметром квантования и заданными параметрами квантования; и

модуль кодирования, конфигурированный для включения этой разностной информации в состав информации потока, генерируемой в результате кодирующей обработки блока, подлежащего кодированию, с использованием заданного параметра квантования.

(10) Устройство кодирования изображения согласно (9), отличающееся тем, что модуль генерации информации выбирает параметр квантования, для которого разность относительно заданного параметра квантования является наименьшей, в качестве прогнозного параметра квантования.

(11) Устройство кодирования изображения согласно (10), отличающееся тем, что модуль генерации информации генерирует идентификационную информацию, указывающую порядок блоков относительно выбранного параметра квантования, где соседствующие кодированные блоки перечисляют в заданном порядке;

и отличающееся тем, что модуль кодирования включает эту идентификационную информацию в состав информации потока.

(12) Устройство кодирования изображения согласно (11), отличающееся тем, что модуль генерации информации принимает порядок массива, где приоритет отдан одному из кодированных блоков – кодированному блоку, соседствующему слева, кодированному блоку, соседствующему сверху, или кодированному блоку, соседствующему во времени.

(13) Устройство кодирования изображения согласно (11) или (12), отличающееся тем, что модуль генерации информации может переключать порядок массива соседствующих кодированных блоков.

(14) Устройство кодирования изображения согласно (9), отличающееся тем, что модуль генерации информации определяет кандидатов для отбора в порядке, заданном заранее, и выбирает прогнозированный параметр квантования на основе результатов определения.

(15) Устройство кодирования изображения согласно (9), отличающееся тем, что модуль генерации информации способен выбирать между процедурой выбора параметра квантования, для которого разность относительно заданного параметра квантования является наименьшей, в качестве прогнозного параметра квантования, и процедурой определения кандидатов для отбора в порядке, заданном заранее, и выбора прогнозного параметра квантования на основе результатов определения и генерирует информацию определения, обозначающую выбранную процедуру;

и отличающееся тем, что модуль кодирования включает информацию определения в состав информации потока.

(16) Устройство кодирования изображения согласно любому из (9) – (15), отличающееся тем, что модуль генерации информации принимает кандидатов для отбора из совокупности соседствующих декодированных блоков, из которой исключены по меньшей мере блоки, где параметры квантования являются избыточными, или блоки, для которых обратное квантование с использованием параметров квантования не

проводилось.

(17) Устройство кодирования изображения согласно любому из (9) – (16), отличающееся тем, что, если нет кандидатов для отбора, модуль генерации информации генерирует разностную информацию, обозначающую разность между параметром квантования, имеющим начальную величину в срезе, и заданными параметрами квантования.

(18) Устройство кодирования изображения согласно любому из (9) – (17), отличающееся тем, что модуль генерации информации включает параметр квантования, обновленный самым последним, в совокупность кандидатов для отбора.

#### Промышленная применимость

При использовании устройства для декодирования изображения, устройства для кодирования изображения и способа работы этих устройств согласно предлагаемой технологии параметры квантования кодированных блоков, соседствующих в пространстве или во времени с блоком, подлежащим кодированию, принимают в качестве кандидатов для отбора и выбирают прогнозный параметр квантования из совокупности кандидатов для отбора в соответствии с заданным параметром квантования. Генерируют разностную информацию, обозначающую разность между прогнозным параметром квантования и параметрами квантования, заданными для блока, подлежащего кодированию. Соответственно, можно не допустить, чтобы величина разности между параметрами квантования стала большой, и можно повысить эффективность кодирования параметров квантования. Кроме того, в случае декодирования информации потока, куда включена разностная информация, выбирают прогнозный параметр квантования из совокупности параметров квантования для декодированных блоков, соседствующих в пространстве или во времени с блоком, подлежащим декодированию, и вычисляют параметр квантования для блока, подлежащего декодированию, на основе прогнозного параметра квантования и разностной информации. Соответственно, даже если информацию потока генерируют при повышенной эффективности кодирования параметров квантования, при декодировании эти параметры квантования могут быть восстановлены на основе прогнозного параметра квантования и разностной информации, так что процедура декодирования может быть выполнена правильно. Соответственно, все это может быть применено в оборудовании, которое передает/принимает информацию потока, получаемую путем кодирования с шагом, равным блоку, через сеть связи, такую как сеть спутникового вещания, сеть кабельного телевидения, Интернет, сеть сотовой телефонной связи или другую подобную сеть, а также в оборудовании, которое обрабатывает такую информацию на носителях записи, таких как оптические диски, магнитные диски, флэш-память и другие подобные носители.

#### Список позиционных обозначений

- 10 устройство кодирования изображения
- 11 модуль аналого-цифрового преобразователя
- 12, 57 буфер реорганизации экрана
- 13 модуль вычитания
- 14 модуль ортогонального преобразования
- 15 модуль квантования
- 16 модуль кодирования без потерь
- 17, 51 промежуточный буфер
- 18 модуль управления частотой квантования
- 19 модуль генерации информации



- 21, 53 модуль обратного квантования
- 22, 54 модуль обратного ортогонального преобразования
- 23, 55 модуль суммирования
- 24, 56 деблокирующий фильтр
- 5 26, 61 память кадров
- 31, 71 модуль внутрикадрового прогнозирования
- 32 модуль прогнозирования/компенсации движения
- 33 модуль выбора прогнозного изображения/оптимального режима
- 50 устройство декодирования изображения
- 10 52 модуль декодирования без потерь
- 58 модуль цифро-аналогового преобразования
- 59 модуль вычисления параметра квантования
- 62, 73 селектор
- 72 модуль компенсации движения
- 15 80 компьютер
- 90 телевизионное устройство
- 92 сотовый телефон
- 94 устройство записи/воспроизведения
- 96 устройство для считывания изображения
- 20 191 модуль памяти параметров квантования
- 192 модуль вычисления разности
- 591 вычислительный модуль
- 592 модуль памяти параметров квантования

25 (57) Формула изобретения

1. Способ кодирования изображения, содержащий этапы, на которых:  
устанавливают, когда блок, соседствующий слева с текущим блоком, и блок,  
соседствующий сверху с текущим блоком, являются доступными, прогнозный параметр  
квантования, выводимый из среднего значения параметра квантования для блока,  
30 соседствующего слева, и параметра квантования для блока, соседствующего сверху;  
устанавливают, когда блок, соседствующий слева с текущим блоком, и блок,  
соседствующий сверху с текущим блоком, являются недоступными, прогнозный  
параметр квантования, выводимый из параметра квантования модуля обработки, не  
соседствующего с текущим блоком;  
35 генерируют разностную информацию, указывающую разность между установленным  
прогнозным параметром квантования и параметром квантования текущего блока; и  
генерируют потоковую информацию, включающую в себя разностную информацию.
2. Способ кодирования изображения по п. 1, в котором параметр квантования модуля  
обработки, не соседствующего с текущим блоком, является начальным значением  
40 параметра квантования в срезе.
3. Способ кодирования изображения по п. 1, в котором, когда блок, соседствующий  
слева, и блок, соседствующий сверху, являются недоступными, выводят параметр  
квантования предыдущего блока для установки прогнозного параметра квантования.
4. Способ кодирования изображения по п. 1, в котором блок является единицей  
45 кодирования, причем единица кодирования имеет иерархическую структуру,  
определяемую иерархической глубиной.
5. Способ кодирования изображения по п. 4, дополнительно содержащий этап, на  
котором

выполняют квантование посредством модуля преобразования, включенного в единицу кодирования, на основе параметра квантования текущего блока.

6. Способ кодирования изображения по п. 1, дополнительно содержащий этап, на котором:

5 кодируют разностную информацию.

7. Устройство обработки информации, содержащее схему, выполненную с возможностью

устанавливать, когда блок, соседствующий слева с текущим блоком, и блок, соседствующий сверху с текущим блоком, являются доступными, прогнозный параметр квантования, выводимый из среднего значения параметра квантования для блока, соседствующего слева, и параметра квантования для блока, соседствующего сверху; устанавливать, когда блок, соседствующий слева с текущим блоком, и блок, соседствующий сверху с текущим блоком, являются недоступными, прогнозный параметр квантования, выводимый из параметра квантования модуля обработки, не соседствующего с текущим блоком;

15 генерировать разностную информацию, указывающую разность между установленным прогнозным параметром квантования и параметром квантования текущего блока; и

генерировать потоковую информацию, включающую в себя разностную информацию.

20 8. Устройство обработки информации по п. 7, в котором

параметр квантования модуля обработки, не соседствующего с текущим блоком, равен начальному значению параметра квантования в срезе.

9. Устройство обработки информации по п. 7, характеризующееся тем, что выполнено с возможностью, когда блок, соседствующий слева, и блок, соседствующий сверху, являются недоступными, выводить параметр квантования предыдущего блока для установки прогнозного параметра квантования.

10. Устройство обработки информации по п. 7, в котором блок является единицей кодирования, причем единица кодирования имеет иерархическую структуру, определяемую иерархической глубиной.

30 11. Устройство обработки информации по п. 10, в котором схема обработки дополнительно выполнена с возможностью

выполнять квантование посредством модуля преобразования, включенного в единицу кодирования, на основе параметра квантования текущего блока.

12. Устройство обработки информации по п. 7, в котором схема обработки дополнительно выполнена с возможностью:

кодировать разностную информацию.

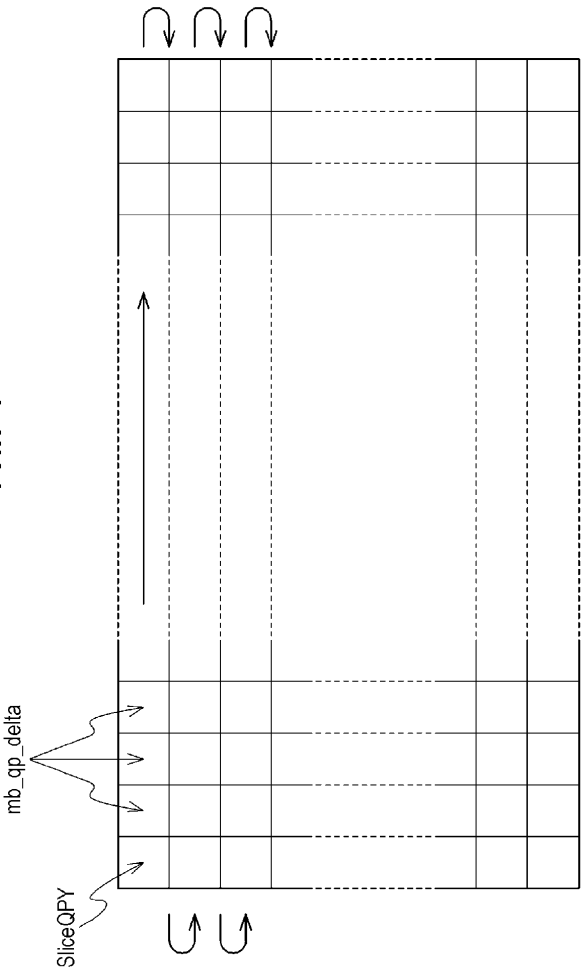
40

45

1

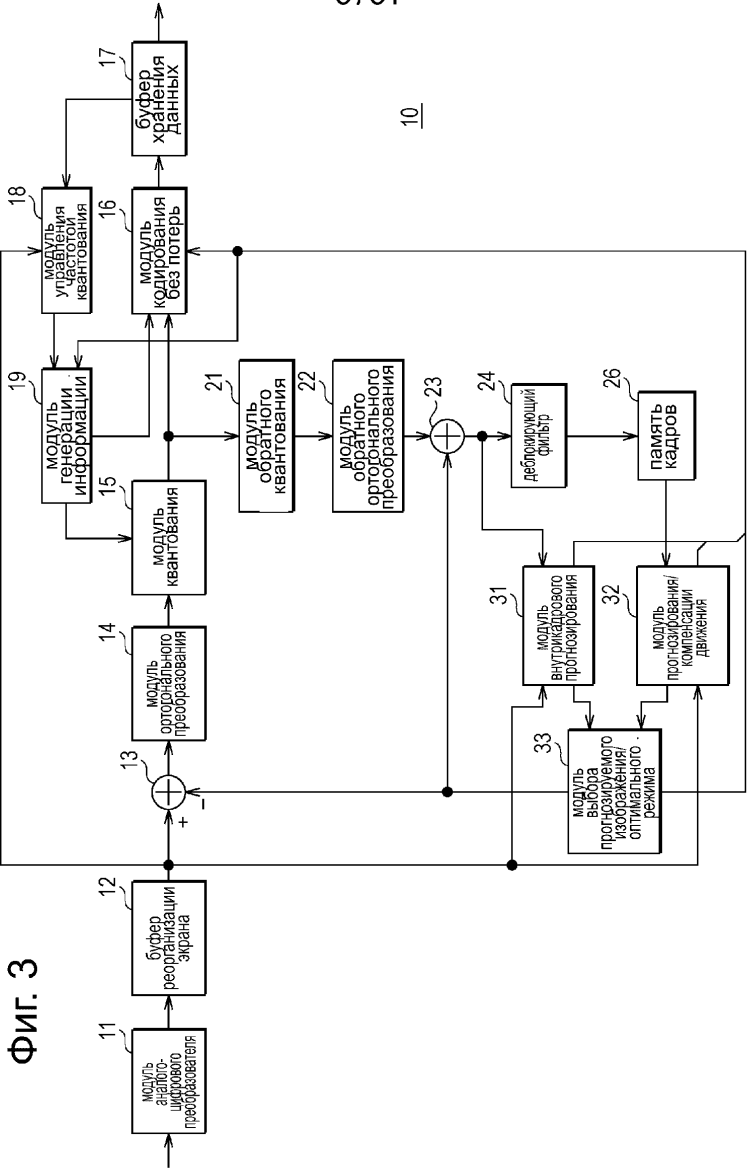
1/31

ФИГ. 1



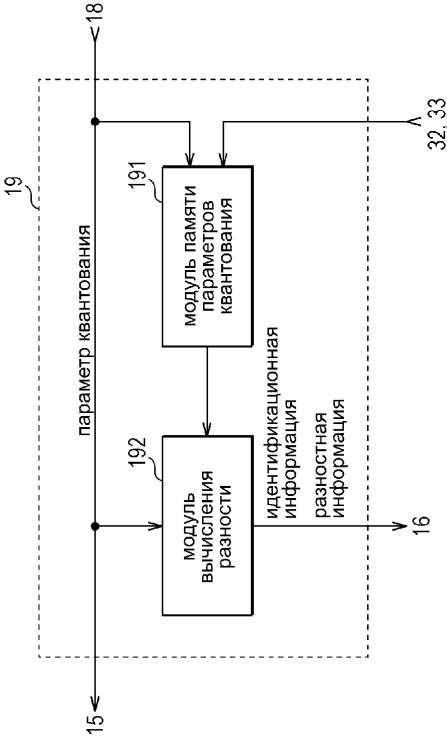
2





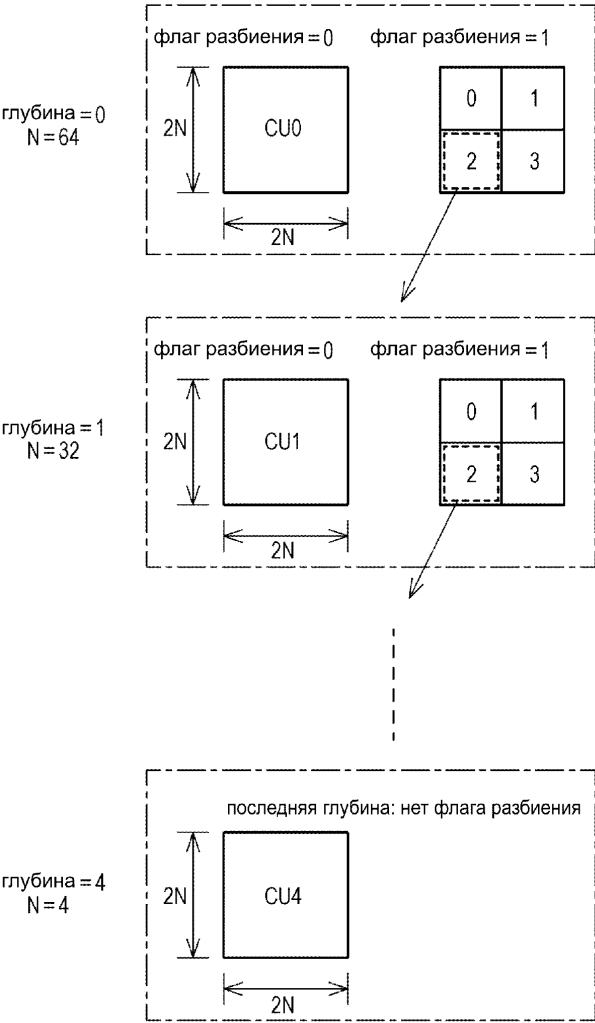
10

Фиг. 4



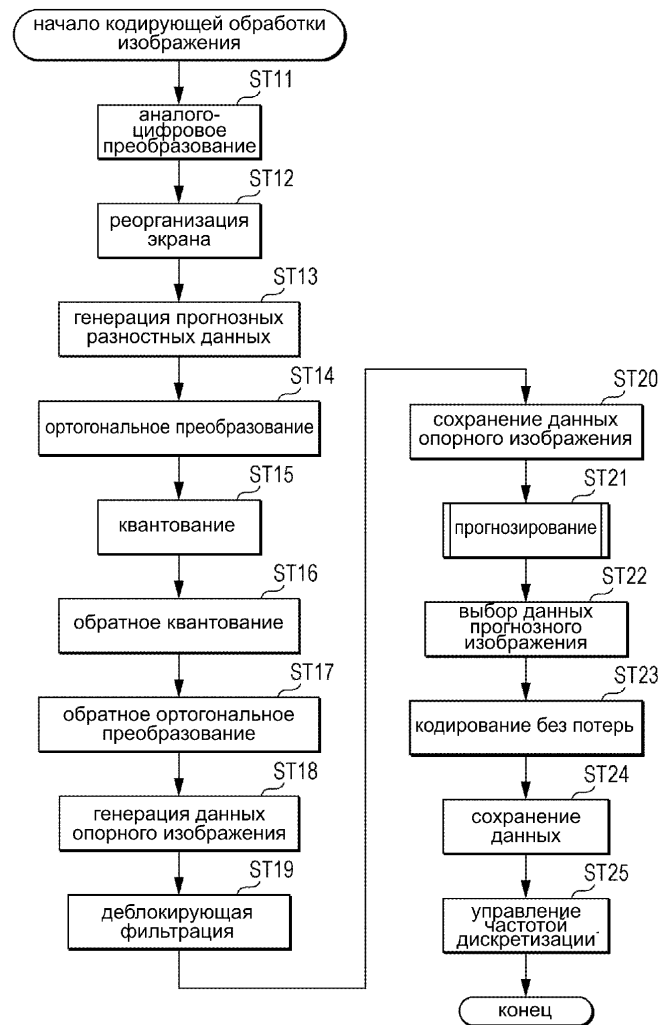
5/31

Фиг. 5



6/31

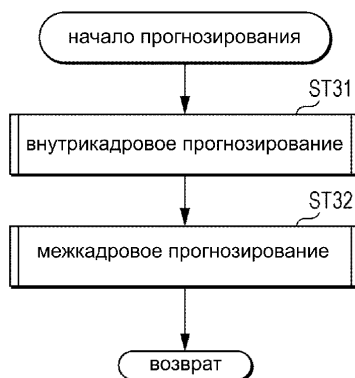
Фиг. 6



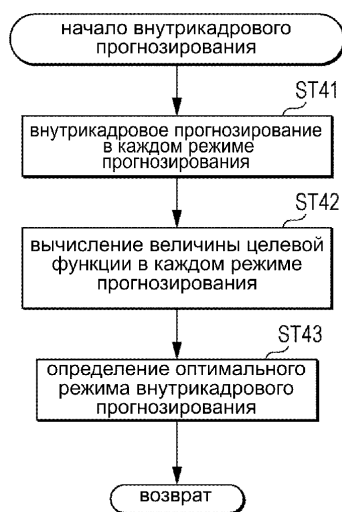


7/31

Фиг. 7

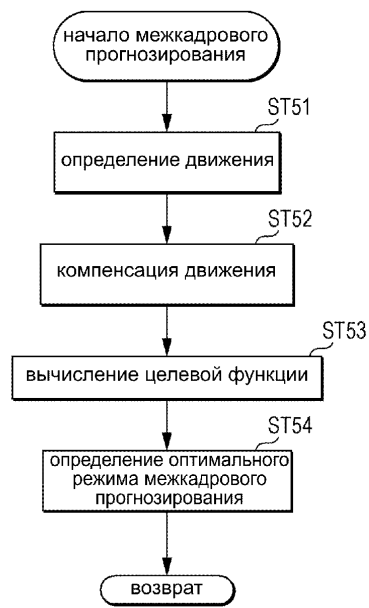


Фиг. 8

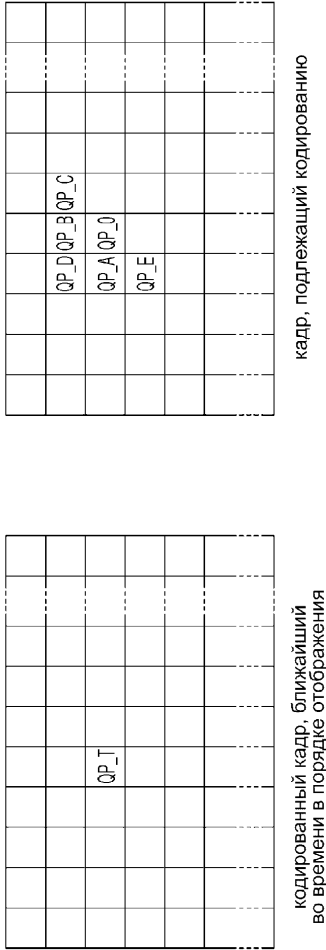


8 / 31

Фиг. 9



Фиг. 10



Фиг. 11

|    |    |    |    |    |    |    |    |    |  |  |
|----|----|----|----|----|----|----|----|----|--|--|
| 31 | -  | 31 | 31 | 31 | 31 | 31 | 31 | 31 |  |  |
| 31 | 33 | 31 | -  | 31 | -  | 31 | 31 | 31 |  |  |
| 31 | 31 | -  | 31 | 31 | 31 | 33 | 31 |    |  |  |
| 31 | 33 | 31 | 31 | 31 | -  | 31 | 31 |    |  |  |
|    |    |    |    |    |    |    |    |    |  |  |
|    |    |    |    |    |    |    |    |    |  |  |

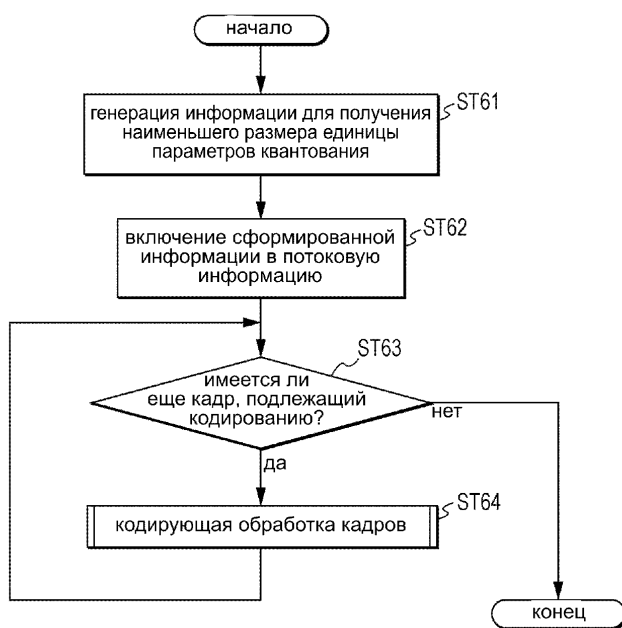
|    |    |    |    |           |    |    |    |  |  |
|----|----|----|----|-----------|----|----|----|--|--|
| 32 | 32 | 32 | 32 | 32        | -  | 40 | 46 |  |  |
| 32 | -  | 32 | 35 | 40        | 40 | 45 | 46 |  |  |
| -  | 32 | 32 | 32 | ВКО<br>33 |    |    |    |  |  |
| 32 | 32 | 32 | -  |           |    |    |    |  |  |
|    |    |    |    |           |    |    |    |  |  |
|    |    |    |    |           |    |    |    |  |  |

кодированный кадр, ближайший  
во времени в порядке отображения

кадр, подлежащий кодированию

11 / 31

Фиг. 12



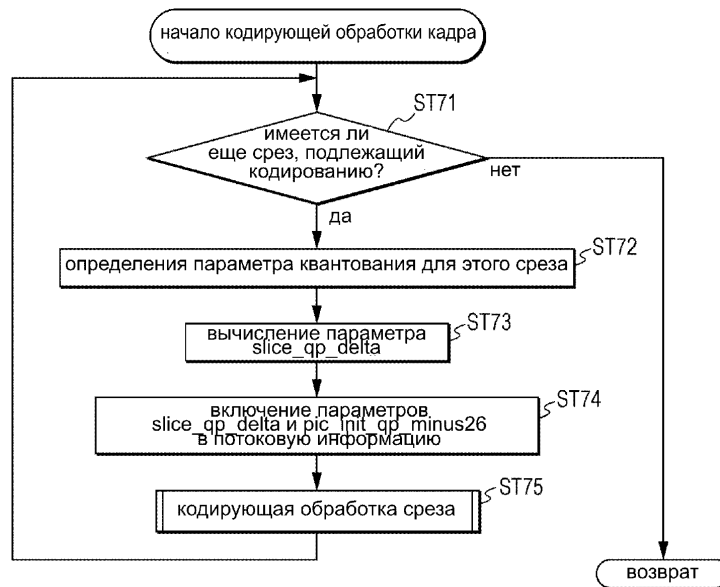
12/31

Фиг. 13

|  |            |
|--|------------|
| seq_parameter_set_rbsp( ) {                                  | дескриптор |
| profile_idc  | u(8)       |
| reserved_zero_8bits /* equal to 0 */                         | u(8)       |
| level_idc  | u(8)       |
| seq_parameter_set_id   | ue(v)      |
| pic_width_in_luma_samples                                    | u(16)      |
| pic_height_in_luma_samples                                   | u(16)      |
| bit_depth_luma_minus8  | ue(v)      |
| bit_depth_chroma_minus8                                      | ue(v)      |
| log2_max_frame_num_minus4                                    | ue(v)      |
| pic_order_cnt_type   | ue(v)      |
| if( pic_order_cnt_type == 0 )                                |            |
| log2_max_pic_order_cnt_lsb_minus4                            | ue(v)      |
| else if( pic_order_cnt_type == 1 ) {                         |            |
| delta_pic_order_always_zero_flag                             | u(1)       |
| offset_for_non_ref_pic                                       | se(v)      |
| num_ref_frames_in_pic_order_cnt_cycle                        | ue(v)      |
| for( i = 0; i < num_ref_frames_in_pic_order_cnt_cycle; i++ ) |            |
| offset_for_ref_frame[ i ]                                    | se(v)      |
| }  |            |
| max_num_ref_frames   | ue(v)      |
| gaps_in_frame_num_value_allowed_flag                         | u(1)       |
| log2_min_coding_block_size_minus3                            | ue(v)      |
| log2_diff_max_min_coding_block_size                          | ue(v)      |
| log2_min_transform_block_size_minus2                         | ue(v)      |
| log2_diff_max_min_transform_block_size                       | ue(v)      |
| max_transform_hierarchy_depth_inter                          | ue(v)      |
| max_transform_hierarchy_depth_intra                          | ue(v)      |
| adaptive_loop_filter_enabled_flag                            | u(1)       |
| cu_qp_delta_enabled_flag                                     | u(1)       |
| rbsp_trailing_bits( )  |            |
| }  |            |

13/31

Фиг. 14



Фиг. 15

|  |
|--|
| pic_parameter_set_rbsp( ) {              |
| pic_parameter_set_id                     |
| seq_parameter_set_id                     |
| entropy_coding_mode_flag                 |
| num_ref_idx_l0_default_active_minus1     |
| num_ref_idx_l1_default_active_minus1     |
| pic_init_qp_minus26 /* relative to 26 */ |
| constrained_intra_pred_flag              |
| rbsp_trailing_bits( )                    |
| }  |

Фиг. 16

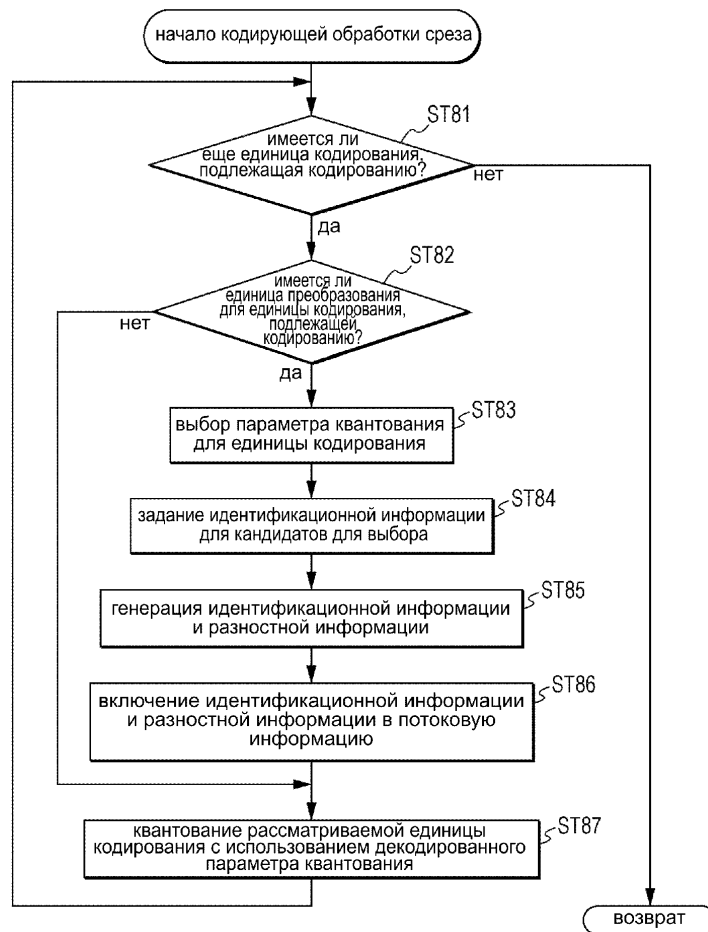
14/31

| slice_header( ) {                                 | дескриптор |
|---|------------|
| first_tb_in_slice                                 | ue(v)      |
| entropy_slice_flag                                | u(1)       |
| if( !entropy_slice_flag ) {                       |            |
| slice_type  | ue(v)      |
| pic_parameter_set_id                              | ue(v)      |
| frame_num   | u(v)       |
| if( !idrPicFlag )                                 |            |
| idr_pic_id  | ue(v)      |
| if( pic_order_cnt_type == 0 )                     |            |
| pic_order_cnt_lsb /*                              | u(v)       |
| if( slice_type == P    slice_type == B ) {        |            |
| num_ref_idx_active_override_flag                  | u(1)       |
| if( num_ref_idx_active_override_flag ) {          |            |
| num_ref_idx_l0_active_minus1                      | ue(v)      |
| if( slice_type == B )                             |            |
| num_ref_idx_l1_active_minus1                      | ue(v)      |
| }   |            |
| }   |            |
| ref_pic_list_modification( )                      |            |
| ref_pic_list_combination( )                       |            |
| if( nal_ref_idc != 0 )                            |            |
| dec_ref_pic_marking( )                            |            |
| if( entropy_coding_mode_flag && slice_type != I ) |            |
| cabac_init_idc                                    | ue(v)      |
| slice_qp_delta                                    | se(v)      |
| if( adaptive_loop_filter_enabled_flag )           |            |
| alf_param( )                                      |            |
| if( deblocking_filter_control_present_flag ) {    |            |
| disable_deblocking_filter_idc                     |            |
| if( disable_deblocking_filter_idc != 1 ) {        |            |
| slice_alpha_c0_offset_div2                        |            |
| slice_beta_offset_div2                            |            |
| }   |            |
| }   |            |
| if( slice_type == B )                             |            |
| collocated_from_l0_flag                           | u(1)       |
| } else  |            |
| if( entropy_coding_mode_flag && slice_type != I ) |            |
| cabac_init_idc                                    | ue(v)      |
| }   |            |

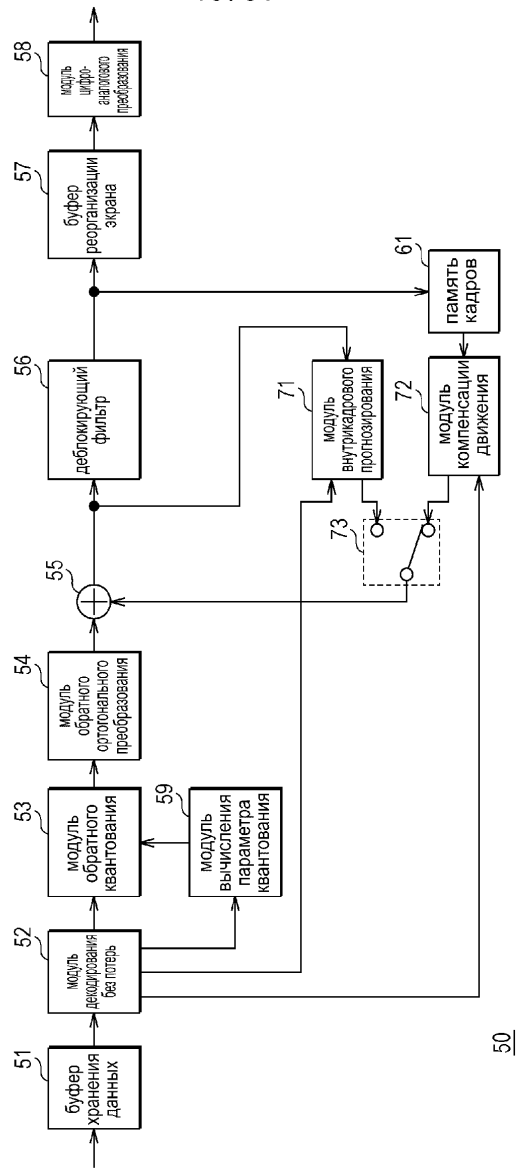


15/31

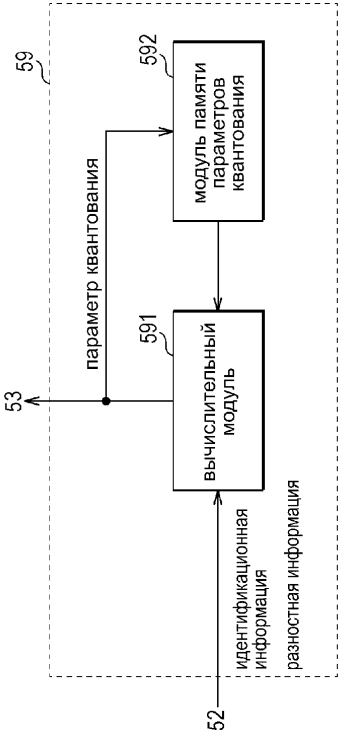
Фиг. 17

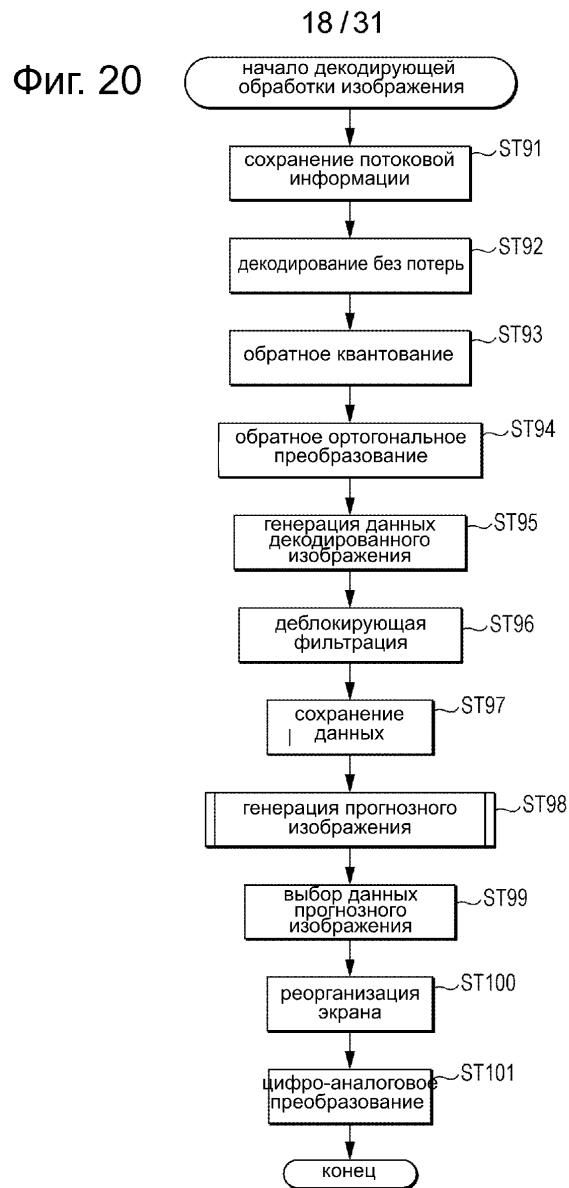


Фиг. 18



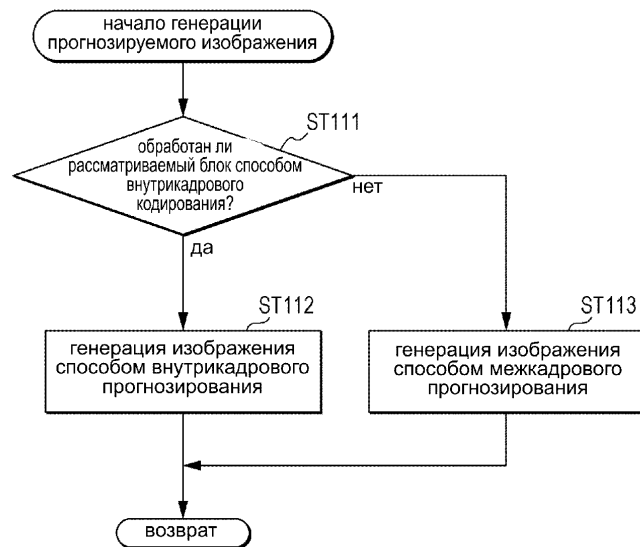
Фиг. 19



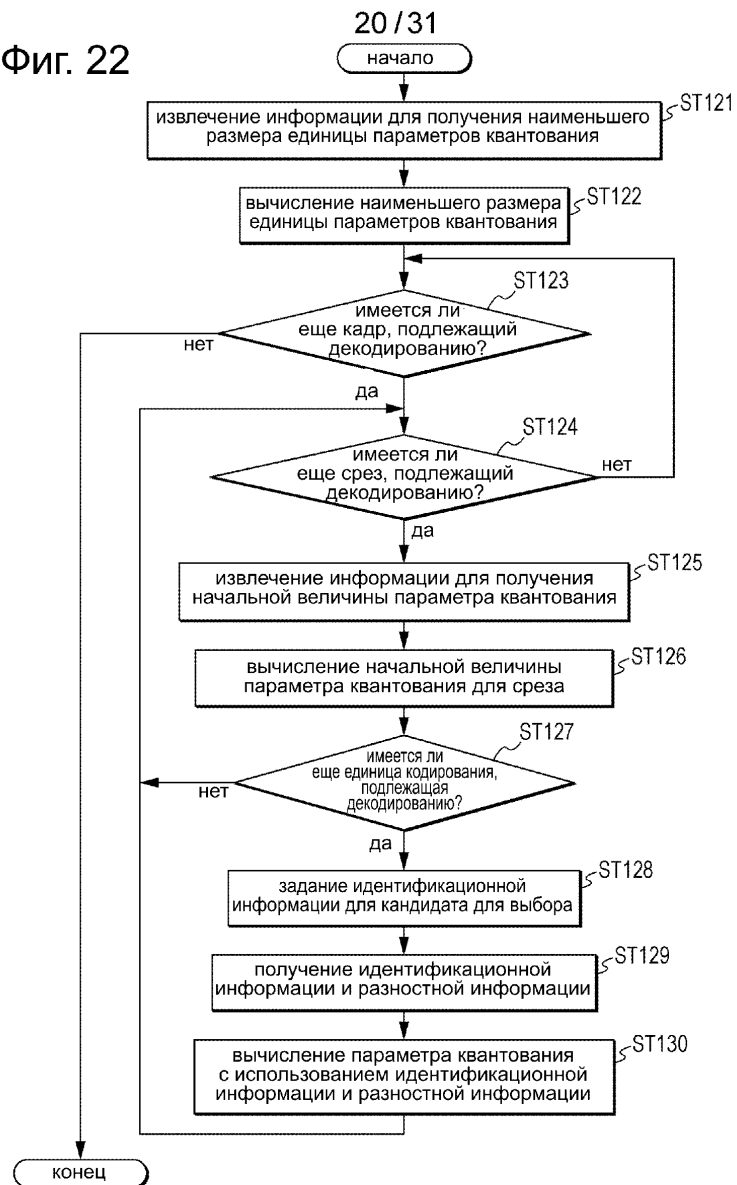


19/31

Фиг. 21

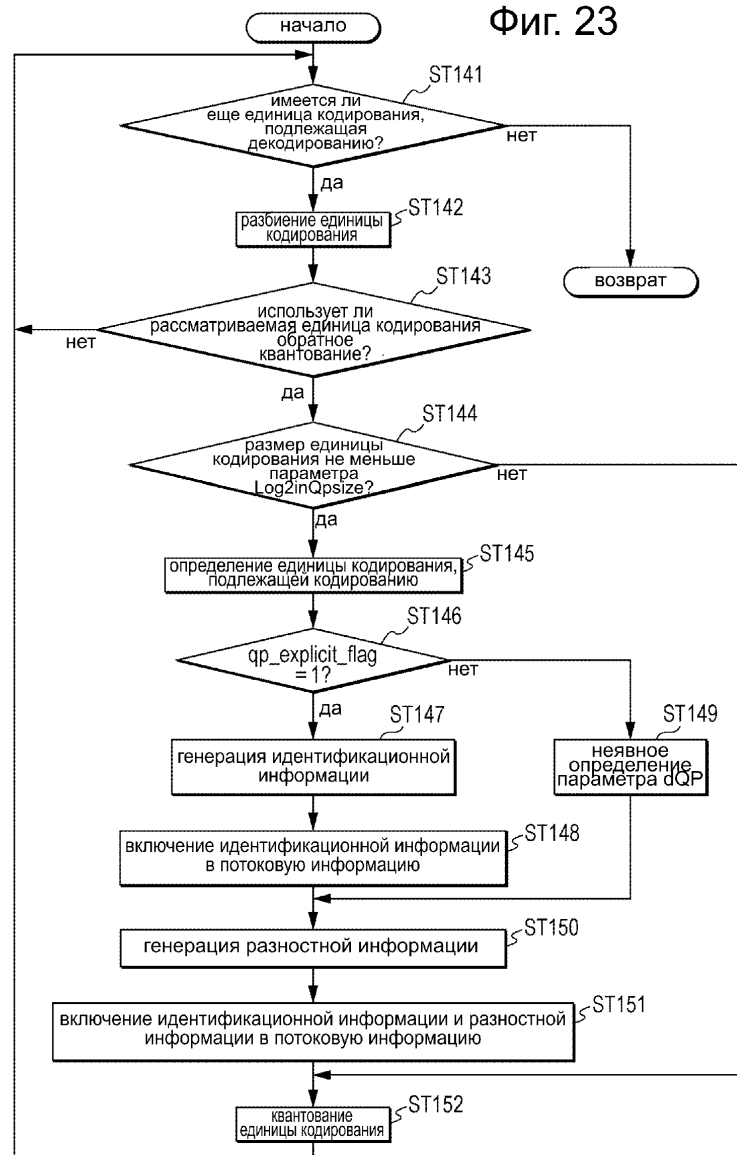


Фиг. 22

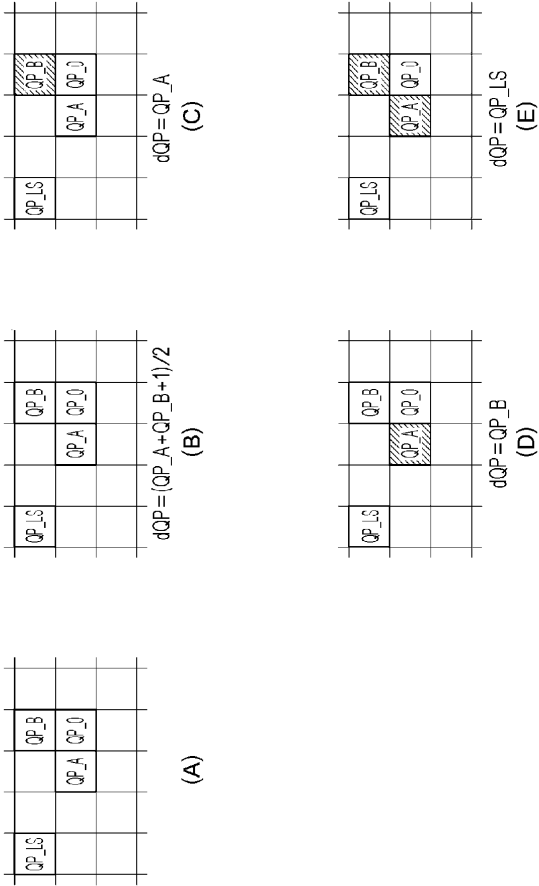


21/31

Фиг. 23

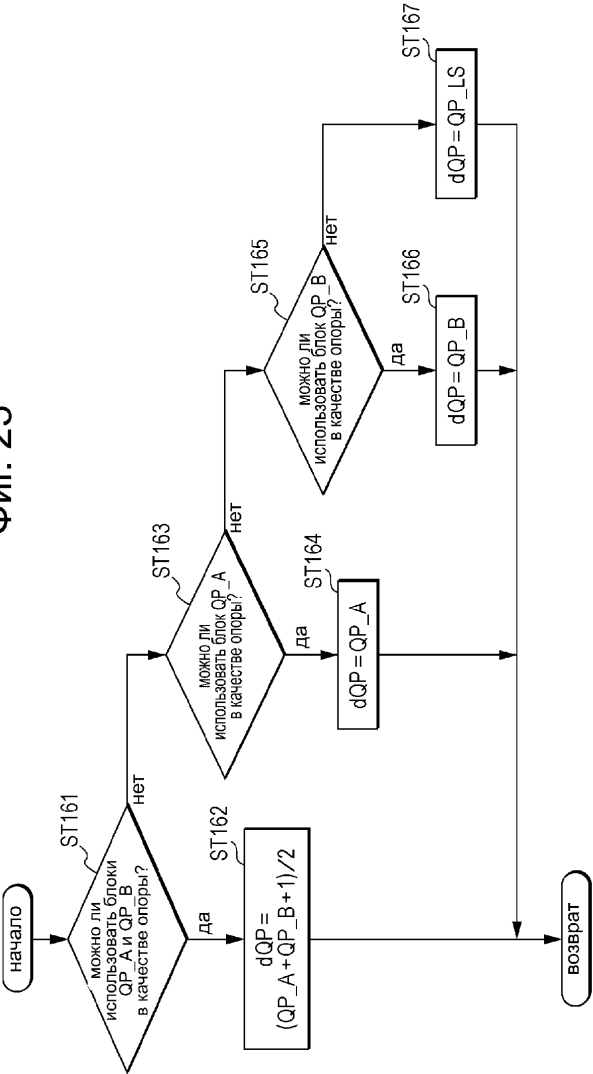


Фиг. 24

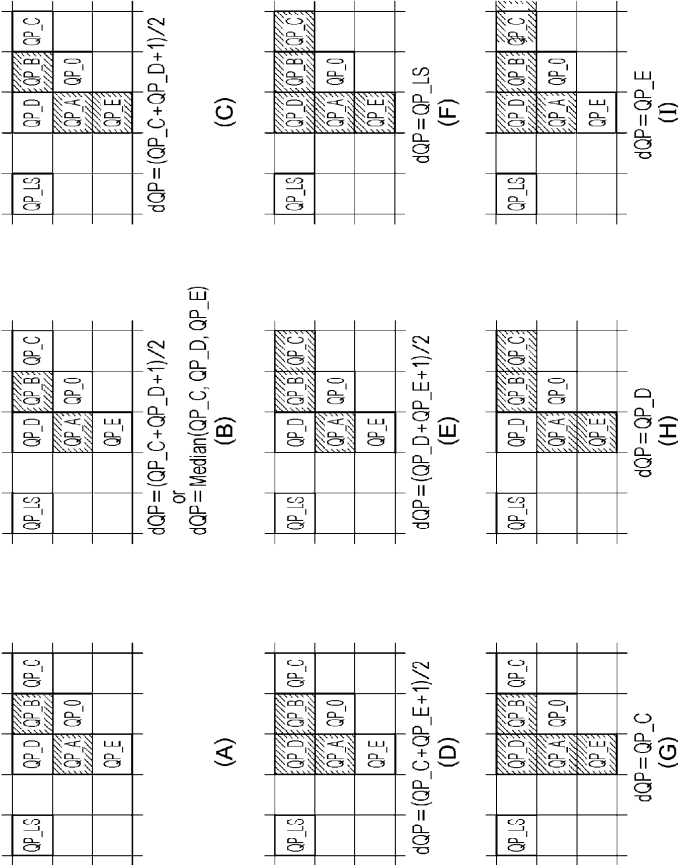




Фиг. 25



Фиг. 26



25/31

## Фиг. 27

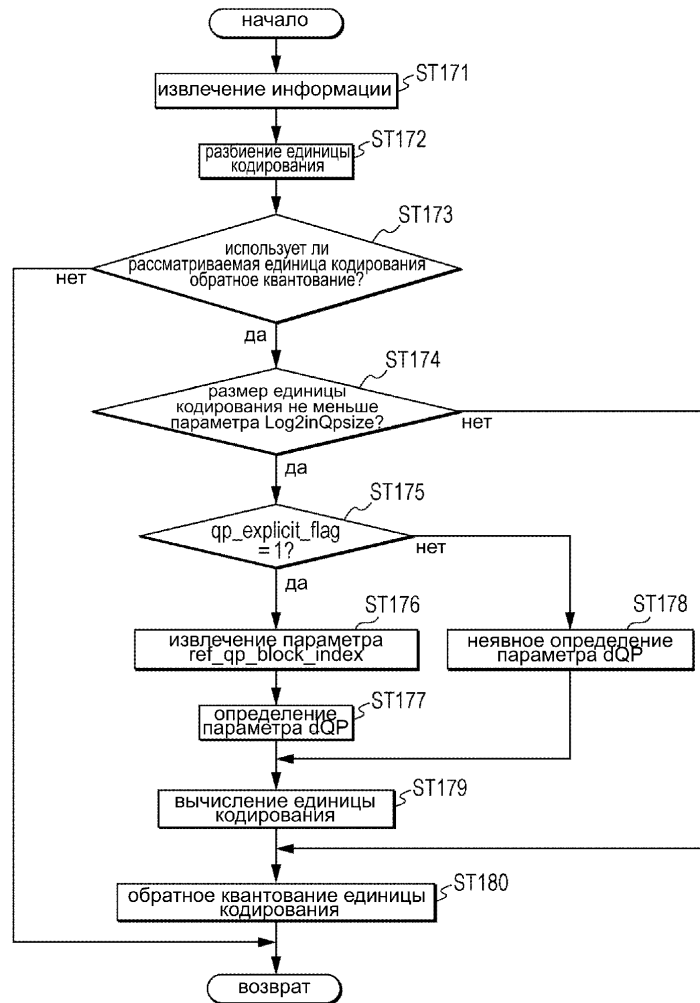
```

if ( left & above ) dQP = Avg.(QP_A, QP_B)
else if ( left ) dQP = QP_A
else if ( above ) dQP = QP_B
else if (above-left & above-right & below-left )
    dQP = Median or Avg.(QP_C, QP_D))
else if (above-left & above-right ) dQP = Avg.(QP_C, QP_D)
else if (above-right & below-left ) dQP = Avg.(QP_C, QP_E)
else if (above-left & below-left ) dQP = Avg.(QP_D, QP_E)
else dQP = QP_LS

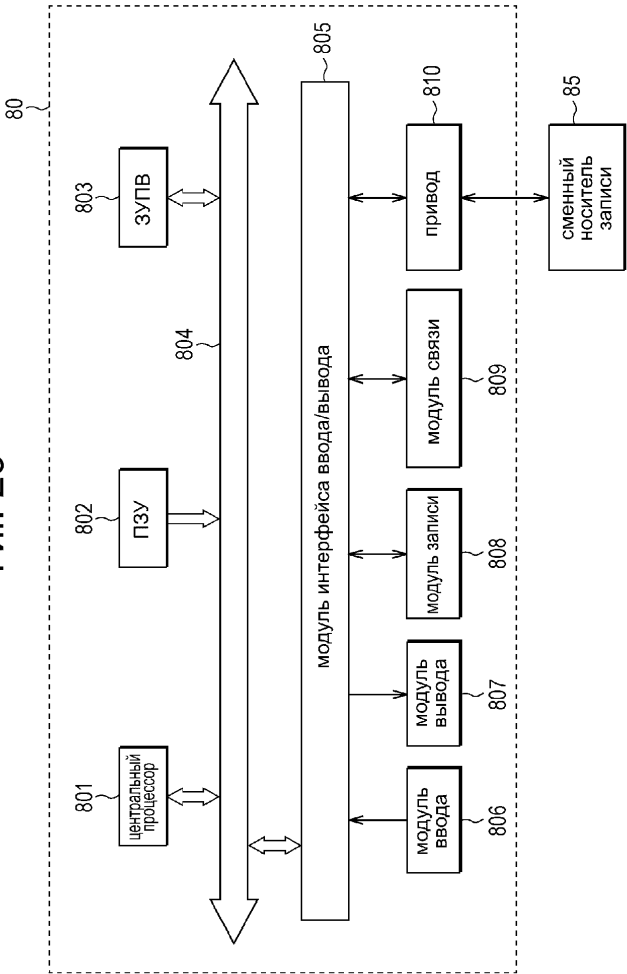
```

26 / 31

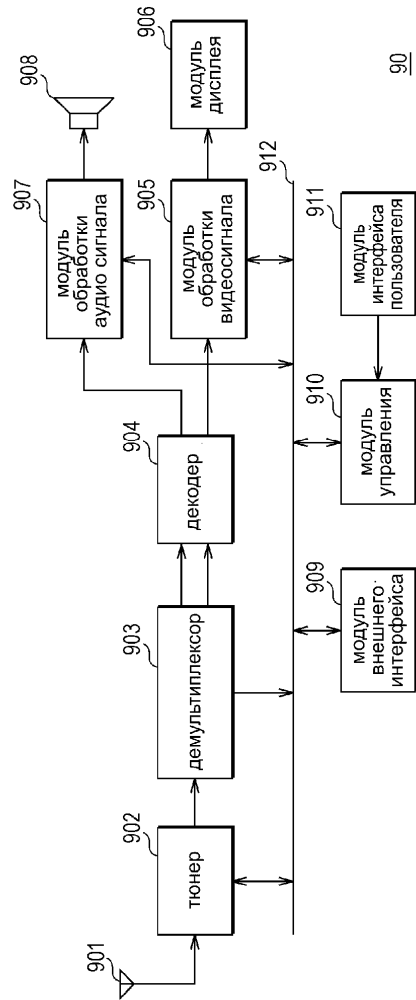
Фиг. 28



Фиг. 29

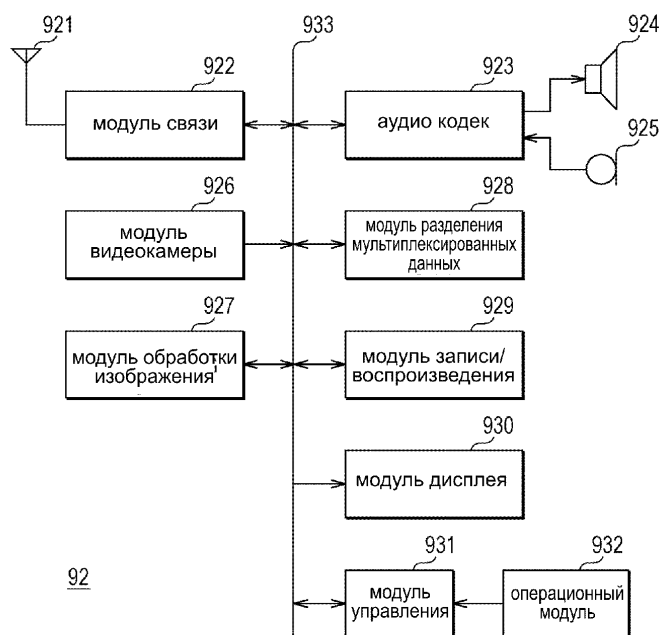


Фиг. 30

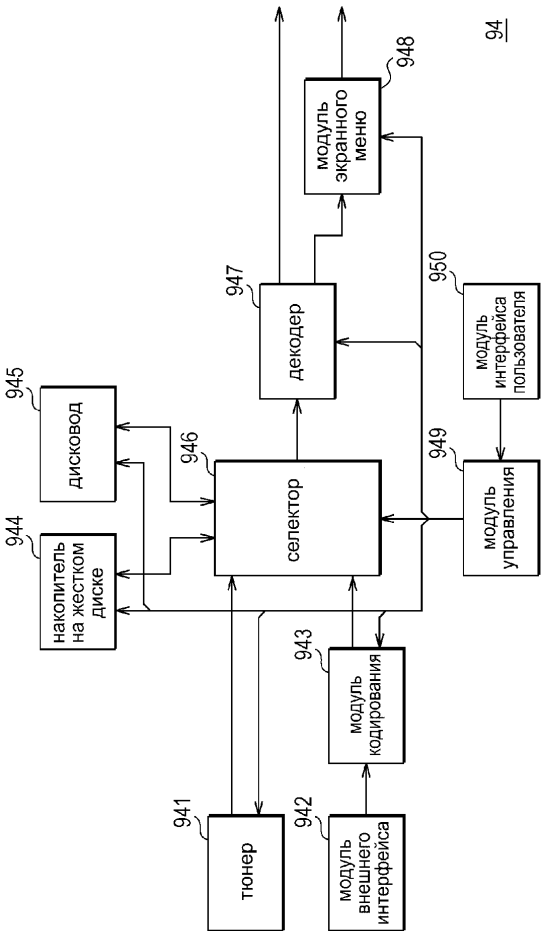


29/31

Фиг. 31



Фиг. 32





Фиг. 33

