



(51) МПК
H04N 19/186 (2014.01)
H04N 19/60 (2014.01)
H04N 19/122 (2014.01)

ФЕДЕРАЛЬНАЯ СЛУЖБА
 ПО ИНТЕЛЛЕКТУАЛЬНОЙ СОБСТВЕННОСТИ

(12) ОПИСАНИЕ ИЗОБРЕТЕНИЯ К ПАТЕНТУ

(21)(22) Заявка: 2015115514/08, 27.09.2013

(24) Дата начала отсчета срока действия патента:
 27.09.2013

Приоритет(ы):

(30) Конвенционный приоритет:
 28.09.2012 AU 2012232992

(43) Дата публикации заявки: 20.11.2016 Бюл. № 32

(45) Опубликовано: 20.12.2016 Бюл. № 35

(56) Список документов, цитированных в отчете о поиске: US 2012/0183080 A1, 19.07.2012. US 2008/0170615 A1, 17.07.2008. US 2012/0177116 A1, 12.07.2012. US 2010/0086034 A1, 08.04.2010. DAVID FLYNN, "BoG report: Extended chroma formats", Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11, 10th Meeting: Stockholm, SE, 11-20 July 2012, Document: JCTVC-J0581, опубл. 19.07.2012, с. 1-7. RU 2426269 C2, 10.08.2011.

(85) Дата начала рассмотрения заявки РСТ на национальной фазе: 28.04.2015

(86) Заявка РСТ:
 AU 2013/001116 (27.09.2013)

(87) Публикация заявки РСТ:
 WO 2014/047693 (03.04.2014)

Адрес для переписки:

129090, Москва, ул. Б. Спасская, 25, стр. 3, ООО
 "Юридическая фирма Городисский и Партнеры"

(72) Автор(ы):

РОУЗВОРН Кристофер Джеймс (AU),
 КОЛЕСНИКОВ Володимир (AU)

(73) Патентообладатель(и):

КЭНОН КАБУСИКИ КАЙСЯ (JP)

(54) СПОСОБ, УСТРОЙСТВО И СИСТЕМА ДЛЯ КОДИРОВАНИЯ И ДЕКОДИРОВАНИЯ ЕДИНИЦ ПРЕОБРАЗОВАНИЯ ЕДИНИЦЫ КОДИРОВАНИЯ

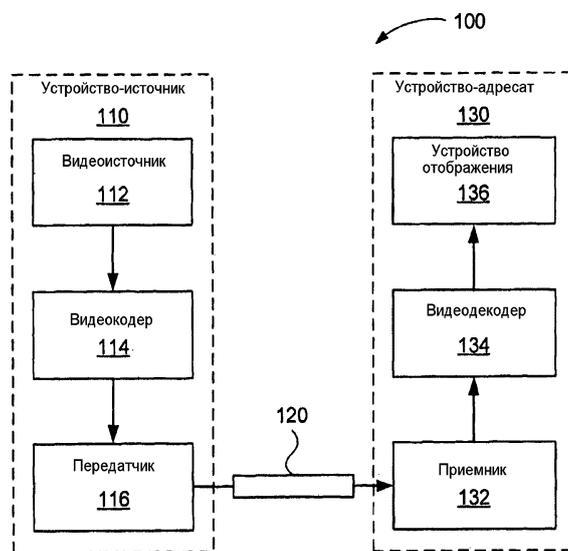
(57) Реферат:

Изобретение относится к области кодирования/декодирования видео. Технический результат - обеспечение повышения эффективности кодирования видео. Способ декодирования единицы преобразования содержит: определение иерархического уровня для единицы преобразования в пределах единицы

кодирования из флагов преобразования разделения; определение размера преобразования единицы преобразования для единственного канала цветности; декодирование множества значений флага кодированного блока из битового видеопотока для единственного канала цветности единицы преобразования, при этом единица

преобразования имеет множество массивов остаточных коэффициентов сигнала цветности; декодирование каждого из множества массивов остаточных коэффициентов сигнала цветности согласно значению флага кодированного блока; выбор квадратного обратного преобразования для декодированных массивов остаточных

коэффициентов сигнала цветности; применение выбранного квадратного обратного преобразования к каждому из декодированных массивов остаточных коэффициентов сигнала цветности для получения остаточных отсчетов сигнала цветности для канала цветности единицы преобразования. 6 н. и 14 з.п. ф-лы, 21 ил.



ФИГ.1



FEDERAL SERVICE
FOR INTELLECTUAL PROPERTY

(51) Int. Cl.
H04N 19/186 (2014.01)
H04N 19/60 (2014.01)
H04N 19/122 (2014.01)

(12) **ABSTRACT OF INVENTION**

(21)(22) Application: **2015115514/08, 27.09.2013**

(24) Effective date for property rights:
27.09.2013

Priority:

(30) Convention priority:
28.09.2012 AU 2012232992

(43) Application published: **20.11.2016** Bull. № 32

(45) Date of publication: **20.12.2016** Bull. № 35

(85) Commencement of national phase: **28.04.2015**

(86) PCT application:
AU 2013/001116 (27.09.2013)

(87) PCT publication:
WO 2014/047693 (03.04.2014)

Mail address:

**129090, Moskva, ul. B. Spasskaja, 25, str. 3, OOO
"JUrIdicheskaja firma Gorodisskij i Partnery"**

(72) Inventor(s):

**ROUZVORN Kristofer Dzhejms (AU),
KOLESNIKOV Volodimir (AU)**

(73) Proprietor(s):

KENON KABUSIKI KAJSSJA (JP)

(54) **METHOD, APPARATUS AND SYSTEM FOR ENCODING AND DECODING TRANSFORMATION UNITS OF CODING UNITS**

(57) Abstract:

FIELD: information technologies.

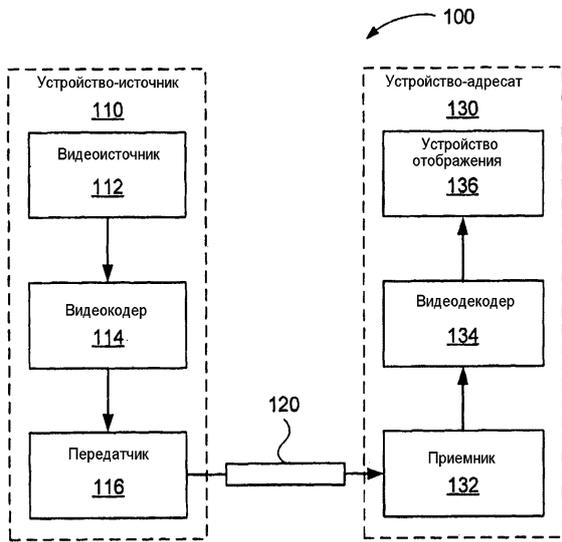
SUBSTANCE: invention relates to encoding/decoding video. Method of decoding transformation units comprises: determination of hierarchical level for transformation unit within coding unit from separation transformation flags; determination of transformation size of transformation unit for single colour channel; decoding multiple flag values of coded unit from bit video stream for a single colour channel of transformation unit, wherein transformation unit has a plurality of arrays of residual coefficients of colour signal; decoding each of plurality of arrays of residual coefficients of colour signal according to flag value of encoded unit; selection of square inverse transformation for decoded arrays of residual coefficients of colour signal; using selected square inverse transformation to each of decoded arrays of residual coefficients of colour signal to obtain residual colour signal readings for colour channel of transformation unit.

EFFECT: technical result is high efficiency of video coding.

20 cl, 21 dwg

RU 2 604 986 C 2

RU 2 604 986 C 2



ФИГ.1

ПЕРЕКРЕСТНАЯ ССЫЛКА НА СВЯЗАННЫЕ ЗАЯВКИ

[0001] Данная заявка в соответствии с §119(e) 35 U.S.C. испрашивает приоритет по дате подачи австралийской патентной заявки № 2012232992, поданной 28 сентября 2012 г., которая тем самым включена в настоящий документ посредством ссылки во всей ее полноте как если бы она была полностью сформулирована в настоящем документе.

ОБЛАСТЬ ТЕХНИКИ

[0002] Настоящее изобретение в целом относится к обработке сигнала цифрового видео и в частности к способу, устройству и системе для кодирования и декодирования остаточных коэффициентов единицы преобразования (TU), где единица преобразования (TU) включает в себя одну или более единиц преобразования (TU) и может конфигурироваться для множества форматов цветности, в том числе формат цветности 4:2:2.

УРОВЕНЬ ТЕХНИКИ

[0003] В настоящее время существует множество приложений для видеокодирования, в том числе приложения для передачи и хранения видеоданных. Также было разработано множество стандартов видеокодирования, и новые стандарты находятся в настоящий момент в разработке. Недавние разработки в стандартизации видеокодирования привели к формированию группы, названной «Объединенная Совместная Команда по Видеокодированию» (JCT-VC). Объединенная Совместная Команда по Видеокодированию (JCT-VC) включает в себя членов Исследовательской группы 16, Вопрос 6 (SG16/Q6) Сектора стандартизации электросвязи (ITU-T) Международного Телекоммуникационного Объединения (ITU), известной как Группа Экспертов по Видеокодированию (VCEG), а также членов Международных организаций по Стандартизации/Совместный Технический Комитет 1 Международной электротехнической комиссии/Подкомиссия 29/Рабочая группа 11 (ISO/IEC JTC1/SC29/WG11), также известной как Группа Экспертов по Кинофильмам (MPEG).

[0004] Объединенная Совместная Команда по Видеокодированию (JCT-VC) ставит своей целью создание нового стандарта видеокодирования для того, чтобы значительно превзойти существующий стандарт видеокодирования, известный как «H.264/MPEG-4 AVC». Стандарт H.264/MPEG-4 AVC сам по себе является большим улучшением по сравнению с предыдущими стандартами видеокодирования, такими как MPEG-4 и ITU-T H.263. Новый разрабатываемый стандарт видеокодирования был назван «высокоэффективным видеокодированием (HEVC)». Объединенная Совместная Команда по Видеокодированию JCT-VC также рассматривает проблемы реализации, являющиеся результатом технологии, предложенной для высокоэффективного видеокодирования (HEVC), которые создают трудности при масштабировании реализаций стандарта для работы с высокими разрешениями в реальном времени или с высокими частотами кадров. Одной проблемой реализации является сложность и размер логики, используемой для поддержки множества размеров «преобразования» для того, чтобы преобразовывать видеоданные между частотной областью и пространственной областью.

СУЩНОСТЬ ИЗОБРЕТЕНИЯ

[0005] Задачей настоящего изобретения является существенно преодолеть, или по меньшей мере уменьшить, один или более недостатков существующих схем.

[0006] В соответствии с одним аспектом настоящего изобретения предлагается способ декодирования единицы преобразования, содержащей остаточные коэффициенты цветности, из битового видеопотока, причем единица преобразования содержит по меньшей мере один массив остаточных коэффициентов сигнала цветности, связанный с единственным каналом цветности, способ содержит:

определение размера единицы преобразования, причем размер связан с иерархическим уровнем единицы преобразования в соответствующей единице кодирования;

идентификацию максимального количества обратных преобразований, используемых для преобразования упомянутого по меньшей мере одного массива остаточных

5 коэффициентов сигнала цветности в соответствии с определенным размером;

декодирование из битового видеопотока упомянутого по меньшей мере одного массива остаточных коэффициентов сигнала цветности с использованием идентифицированного максимального количества преобразований для канала цветности единицы преобразования;

10 выбор обратного преобразования для декодированных массивов остаточных коэффициентов сигнала цветности, причем обратное преобразование выбирается из predetermined набора обратных преобразований; и

применение выбранного обратного преобразования к каждому из массивов остаточных коэффициентов сигнала цветности для декодирования остаточных отсчетов сигнала цветности для канала цветности единицы преобразования.

15 [0007] В соответствии с другим аспектом настоящего раскрытия предлагается способ кодирования в битовый видеопоток единицы преобразования, содержащей остаточные отсчеты сигнала цветности, связанные с единственным каналом цветности, причем единица преобразования содержит по меньшей мере один массив остаточных отсчетов

20 сигнала цветности, способ содержит:

определение размера единицы преобразования, причем размер связан с иерархическим уровнем единицы преобразования в соответствующей единице кодирования;

идентификацию максимального количества predetermined прямых преобразований, используемых для преобразования упомянутого по меньшей мере

25 одного массива остаточных отсчетов сигнала цветности в соответствии с определенным размером;

выбор прямого преобразования для массивов остаточных отсчетов сигнала цветности, причем прямое преобразование выбирается из predetermined набора прямых преобразований;

30 применение выбранного прямого преобразования к каждому из массивов остаточных отсчетов сигнала цветности для преобразования по меньшей мере одного из массивов остаточных отсчетов сигнала цветности в соответствующий массив остаточных коэффициентов сигнала цветности для канала цветности единицы преобразования; и

35 кодирование массивов остаточных коэффициентов сигнала цветности для канала цветности единицы преобразования;

[0008] Предпочтительно максимальное количество преобразований равно одному или двум. Предпочтительно это количество равно двум и применяется в формате цветности 4:2:2 к имеющей размер 32×16 области цветности единицы преобразования.

40 [0009] Предпочтительно количество преобразований выбирается из набора из одного, двух и четырех. В конкретной реализации это количество равно четырем и применяется в формате цветности 4:4:4 к имеющей размер 32×32 области цветности единицы преобразования с размером 32×32.

[00010] Предпочтительно применяется единственное сканирование, покрывающее идентифицированное количество преобразований. Желательно, чтобы коэффициенты

45 идентифицированного количества преобразований перемежались.

[00011] В одной реализации единица преобразования, имеющая размер 4×8, сканируется с использованием схемы сканирования подблоков 4×4.

[00012] В другой реализации количество применяемых преобразований определяется

с использованием по меньшей мере флага блока кода. Желательно, чтобы количество применяемых преобразований определялось с использованием идентифицированного максимального количества преобразований и значения флага кодированного блока для каждого преобразования.

5 [00013] В соответствии с другим аспектом настоящего изобретения предлагается способ декодирования единицы преобразования, содержащей остаточные отсчеты сигнала цветности, из битового видеопотока, причем единица преобразования содержит по меньшей мере один массив остаточных коэффициентов сигнала цветности, связанный с единственным каналом цветности в формате цветности 4:2:2, способ содержит:

10 определение иерархического уровня для единицы преобразования в пределах единицы кодирования из флагов преобразования разделения, присутствующих в битовом видеопотоке, при этом размер единицы кодирования находится в диапазоне от наименьшей единицы кодирования до наибольшей единицы кодирования;

15 определение размера преобразования единицы преобразования для единственного канала цветности, причем размер преобразования связан с определенным иерархическим уровнем единицы преобразования и размером единицы кодирования;

20 определение множества значений флага кодированного блока из битового видеопотока для единственного канала цветности единицы преобразования, при этом единица преобразования имеет множество массивов остаточных коэффициентов сигнала цветности для единственного цветового канала, и каждое значение флага кодированного блока из упомянутого множества значений флага кодированного блока соответствует одному массиву остаточных коэффициентов сигнала цветности из массивов остаточных коэффициентов сигнала цветности;

25 декодирование из битового видеопотока каждого из множества массивов остаточных коэффициентов сигнала цветности согласно соответствующему значению флага кодированного блока из упомянутого множества значений флага кодированного блока;

30 выбор квадратного обратного преобразования для декодированных массивов остаточных коэффициентов сигнала цветности, причем квадратное обратное преобразование выбирается из predetermined набора квадратных обратных преобразований согласно определенному размеру преобразования; и

применение выбранного квадратного обратного преобразования к каждому из декодированных массивов остаточных коэффициентов сигнала цветности для получения остаточных отсчетов сигнала цветности для канала цветности единицы преобразования.

35 [00014] В соответствии с другим аспектом настоящего изобретения предлагается способ кодирования единицы преобразования, содержащей остаточные отсчеты сигнала цветности, в битовый видеопоток, причем единица преобразования содержит по меньшей мере один массив остаточных коэффициентов сигнала цветности, связанный с единственным каналом цветности в формате цветности 4:2:2, способ содержит:

40 кодирование флагов преобразования разделения в битовый видеопоток на основе принятого иерархического уровня для единицы преобразования в пределах единицы кодирования, при этом размер единицы кодирования находится в диапазоне от наименьшей единицы кодирования до наибольшей единицы кодирования;

45 прием размера преобразования единицы преобразования для единственного канала цветности, причем размер преобразования связан с иерархическим уровнем единицы преобразования и размером единицы кодирования;

кодирование множества значений флага кодированного блока в битовый видеопоток для единственного канала цветности единицы преобразования, при этом единица преобразования имеет множество массивов остаточных коэффициентов сигнала

цветности для единственного цветового канала, и каждое значение флага кодированного блока из упомянутого множества значений флага кодированного блока соответствует одному массиву остаточных коэффициентов сигнала цветности из массивов остаточных коэффициентов сигнала цветности;

5 выбор квадратного прямого преобразования для массивов остаточных коэффициентов сигнала цветности, причем квадратное прямое преобразование выбирается из predetermined набора квадратных прямых преобразований согласно принятому размеру преобразования;

10 применение выбранного квадратного прямого преобразования к каждому из декодированных массивов остаточных коэффициентов сигнала цветности для получения остаточных отсчетов сигнала цветности для канала цветности единицы преобразования;
и

15 кодирование в битовый видеопоток каждого из упомянутого множества массивов остаточных коэффициентов сигнала цветности согласно соответствующему значению флага кодированного блока из упомянутого множества значений флага кодированного блока.

[00015] Также раскрываются и другие аспекты.

КРАТКОЕ ОПИСАНИЕ ЧЕРТЕЖЕЙ

20 [00016] По меньшей мере один вариант осуществления настоящего изобретения будет теперь описан со ссылкой на следующие чертежи, в которых:

[00017] Фиг.1 представляет собой блок-схему, показывающую систему кодирования и декодирования видео;

25 [00018] Фиг. 2А и 2В образуют схематическую блок-схему компьютерной системы общего назначения, на которой может быть осуществлена система по меньшей мере одного из кодирования и декодирования видео, изображенная на Фиг.1;

[00019] Фиг.3 представляет собой блок-схему, показывающую функциональные модули видеокодера;

[00020] Фиг.4 представляет собой блок-схему, показывающую функциональные модули видеодекодера;

30 [00021] Фиг. 5А и 5В схематично иллюстрируют форматы цветности для представления данных кадра;

[00022] Фиг.6А является схематическим представлением примерного дерева преобразований единицы кодирования;

35 [00023] Фиг.6В является схематическим представлением примерного дерева преобразований, расположенного на сетке отсчетов сигнала яркости;

[00024] Фиг.6С является схематическим представлением примерного дерева преобразований, расположенного на сетке отсчетов сигнала цветности;

[00025] Фиг.7 представляет собой схематическую иллюстрацию структуры данных, представляющей канал яркости примерного дерева преобразований;

40 [00026] Фиг.8 иллюстрирует структуру данных, представляющую канал цветности примерного дерева преобразований;

[00027] Фиг. 9А и 9В схематично показывают структуру потока битов, которая кодирует примерное дерево преобразований;

45 [00028] Фиг. 9С и 9D схематично показывают альтернативную структуру потока битов, которая кодирует примерное дерево преобразований;

[00029] Фиг.10 представляет собой блок-схему последовательности операций, показывающую способ для кодирования примерного дерева преобразований;

[00030] Фиг.11 представляет собой блок-схему последовательности операций,

показывающую способ для декодирования примерного дерева преобразований; и [00031] Фиг. 12А-12С схематично показывают схемы сканирования остатка единицы преобразования размером 4×8.

ПОДРОБНОЕ ОПИСАНИЕ, ВКЛЮЧАЮЩЕЕ В СЕБЯ ЛУЧШИЙ ВАРИАНТ ОСУЩЕСТВЛЕНИЯ

[00032] Там, где ссылочные цифры, обозначающие этапы и/или особенности, в любом одном или более сопроводительных чертежей являются одинаковыми, эти этапы и/или особенности в целях данного описания представляют собой одинаковые функции или операции, если явно не указано противоположное.

[00033] Фиг.1 представляет собой блок-схему, показывающую функциональные модули системы 100 кодирования и декодирования видео, которая может использовать способы кодирования синтаксических элементов, характерных для предполагаемого подразделения единиц преобразования на многократные преобразования для канала цветности. Система 100 включает в себя устройство-источник 110 и устройство-адресат 130. Канал 120 связи используется для передачи закодированной видеoinформации от устройства-источника 110 к устройству-адресату 130. В некоторых случаях устройство-источник 110 и устройство-адресат 130 могут включать в себя соответствующие мобильные телефоны, и в этом случае канал 120 связи является беспроводным каналом. В других случаях устройство-источник 110 и устройство-адресат 130 могут включать в себя оборудование для организации видеоконференций, и в этом случае канал 120 связи обычно является проводным каналом, таким как интернет-соединение. Кроме того, устройство-источник 110 и устройство-адресат 130 могут включать в себя любое из широкого диапазона устройств, в том числе устройства, поддерживающие эфирное телевидение, приложения кабельного телевидения, приложения интернет-видео и в том числе приложения, в которых закодированное видео записывается на некотором носителе данных или файловом сервере.

[00034] Как было проиллюстрировано, устройство-источник 110 включает в себя видеоисточник 112, видеокодер 114 и передатчик 116. Видеоисточник 112 обычно включает в себя источник данных захватываемого видеокадра, такой как датчик формирователя изображений, ранее захваченную видео последовательность, сохраненную на невременном (“non-transitory”) носителе записи, или видеопоток от удаленного датчика формирователя изображений. Примеры устройств-источников 110, которые могут включать в себя датчик формирователя изображений в качестве видеоисточника 112, включают в себя смартфоны, видеокамеры и сетевые видеокамеры. Видеокодер 114 преобразует захваченные от видеоисточника 112 данные кадра в закодированные видеоданные и будет описан далее со ссылкой на Фиг.3. Закодированные видеоданные обычно передаются передатчиком 116 по каналу 120 связи в виде закодированной видеoinформации. Для закодированных видеоданных также возможно сохранение в некотором устройстве хранения данных, таком как флэш-память или жесткий диск, до того, как они позже будут переданы по каналу 120 связи.

[00035] Устройство-адресат 130 включает в себя приемник 132, видеodeкодер 134 и устройство 136 отображения. Приемник 132 принимает закодированную видеoinформацию из канала 120 связи и передает принятые видеоданные видеodeкодеру 134. Видеodeкодер 134 затем выводит декодированные данные кадра на устройство 136 отображения. Примеры устройства 136 отображения включают в себя электронно-лучевую трубку, жидкокристаллическое устройство отображения, такое как в смартфонах, планшетных компьютерах, компьютерных мониторах или в автономных телевизионных приемниках. Также возможно воплощение функциональности каждого

из устройства-источника 110 и устройства-адресата 130 в одном устройстве.

[00036] Невзирая на примерные устройства, упомянутые выше, каждое из устройства-источника 110 и устройства-адресата 130 может быть выполнено внутри вычислительной системы общего назначения, обычно путем комбинирования аппаратных и программных компонентов. Фиг.2А иллюстрирует такую компьютерную систему 200, которая включает в себя: компьютерный модуль 201; устройства ввода, такие как клавиатура 202, устройство 203 указателя мыши, сканер 226, камера 227, которые могут конфигурироваться как видеоисточник 112, и микрофон 280; а также устройства вывода, включающие в себя принтер 215, устройство 214 отображения, которое может конфигурироваться как устройство 136 отображения, и громкоговорители 217. Внешнее приемопередающее устройство 216 модулятора-демодулятора (модема) может использоваться компьютерным модулем 201 для передачи информации к и из сети 220 связи через соединение 221. Сеть 220 связи, которая может представлять собой канал 120 связи, может быть глобальной сетью (WAN), такой как Интернет, сотовой телекоммуникационной сетью или частной WAN. Когда соединение 221 является телефонной линией, модем 216 может быть традиционным модемом “коммутируемой линии передачи”. Альтернативно, когда соединение 221 является высокоскоростным (например, кабельным) соединением, модем 216 может быть широкополосным модемом. Беспроводной модем также может использоваться для беспроводного соединения с сетью 220 связи. Устройство 216 приемопередатчика может обеспечить функциональность передатчика 116 и приемника 132, а канал 120 связи может быть воплощен в соединении 221.

[00037] Компьютерный модуль 201 обычно включает в себя по меньшей мере один блок 205 процессора и блок 206 памяти. Например, блок 206 памяти может иметь полупроводниковую память произвольного доступа (RAM) и полупроводниковую память только для чтения (ROM). Компьютерный модуль 201 также включает в себя ряд интерфейсов ввода/вывода (I/O), в том числе: аудио-видео интерфейс 207, который соединяется с устройством 214 отображения, громкоговорителями 217 и микрофоном 280; интерфейс ввода-вывода 213, который соединяется с клавиатурой 202, мышью 203, сканером 226, камерой 227 и опционально с джойстиком или другим устройством интерфейса пользователя (не показано); а также интерфейс 208 для внешнего модема 216 и принтера 215. В некоторых реализациях модем 216 может быть включен в компьютерный модуль 201, например в пределах интерфейса 208. Компьютерный модуль 201 также имеет локальный сетевой интерфейс 211, который обеспечивает связь компьютерной системы 200 через соединение 223 с локальной сетью 222 связи, известной как локальная сеть (LAN). Как проиллюстрировано на Фиг.2А, локальная сеть 222 может также соединяться с глобальной сетью 220 через соединение 224, которое обычно включает в себя так называемое устройство «брандмауэра» или устройство подобной функциональности. Интерфейс 211 локальной сети может включать в себя плату Ethernet™, беспроводной интерфейс Bluetooth™ или беспроводной интерфейс IEEE 802,11; однако для интерфейса 211 могут быть осуществлены многочисленные другие типы интерфейсов. Локальный сетевой интерфейс 211 может также обеспечивать функциональность передатчика 116 и приемника 132, а канал 120 связи также может быть воплощен в локальной сети 222.

[00038] Интерфейс ввода-вывода 208 и 213 может обеспечивать одну или обе из последовательной и параллельной связи, причем первая обычно реализуется в соответствии со стандартами Универсальной Последовательной Шины (USB) и имеет соответствующие соединители USB (не показаны). Предусматриваемые устройства 209

хранения данных обычно включают в себя жесткий диск (HDD) 210. Другие устройства хранения данных, такие как дисковод и накопитель на магнитной ленте (не показаны) также могут использоваться. Оптический дисковод 212 обычно предусматривается для использования в качестве энергонезависимого источника данных. Портативные устройства памяти, например такие как оптические диски (например, CD-ROM, DVD, Blu-ray Disc™), USB-RAM, портативные внешние жесткие диски, и дискеты, могут использоваться в качестве подходящих источников данных для компьютерной системы 200. Как правило, любой из жесткого диска 210, накопителя на оптических дисках 212 и сетей 220 и 222 также может быть выполнено с возможностью работы в качестве видеоисточника 112, или в качестве места назначения для декодированных видеоданных, которые должны быть сохранены для воспроизведения через устройство 214 отображения.

[00039] Компоненты 205-213 компьютерного модуля 201 обычно обмениваются информацией через соединяющую их шину 204 таким способом, который приводит к обычному режиму работы компьютерной системы 200, известному специалистам в данной области техники. Например, процессор 205 связывается с системной шиной 204 с использованием соединения 218. Аналогичным образом, память 206 и оптический дисковод 212 связываются с системной шиной 204 с помощью соединений 219. Примеры компьютеров, на которых могут быть осуществлены описанные компоновки, включают в себя компьютеры типа IBM PC и совместимые с ними, Sun SPARCstations, Apple Mac™ или подобные компьютерные системы.

[00040] Когда это необходимо или желательно, видеокодер 114 и видеodeкодер 134, так же как и описанные ниже способы, могут быть реализованы с использованием компьютерной системы 200, в которой видеокодер 114, видеodeкодер 134 и процессы, которые изображены на Фиг. 10 - 13 и будут описаны позже, могут быть реализованы как одно или более программных приложений 233, исполняемых на компьютерной системе 200. В частности, видеокодер 114, видеodeкодер 134 и этапы описанных способов реализуются инструкциями 231 (см. Фиг.2В) в программном обеспечении 233, которые выполняются на компьютерной системе 200. Инструкции 231 программного обеспечения могут быть сформированы как один или более модулей кода, каждый из которых выполняет одну или более конкретных задач. Программное обеспечение также может быть поделено на две отдельные части, в которых первая часть и соответствующий ей модуль кода выполняет описанные способы, а вторая часть и соответствующий ей модуль кода управляет пользовательским интерфейсом между первой частью и пользователем.

[00041] Программное обеспечение может быть сохранено на считываемом компьютером носителе, в том числе, например, устройствах хранения данных, описанных ниже. Программное обеспечение загружается в компьютерную систему 200 со считываемого компьютером носителя, а затем выполняется компьютерной системой 200. Считываемый компьютером носитель, содержащий такое программное обеспечение или компьютерную программу, записанную на считываемом компьютером носителе, является компьютерным программным продуктом. Использование компьютерного программного продукта в компьютерной системе 200 предпочтительно осуществляет предпочтительное устройство для реализации видеокодера 114, видеodeкодера 134 и описанных способов.

[00042] Программное обеспечение 233 обычно хранится на жестком диске 210 или в памяти 206. Программное обеспечение загружается в компьютерную систему 200 со считываемого компьютером носителя, а затем выполняется компьютерной системой

200. Таким образом, например, программное обеспечение 233 может быть сохранено на оптически читаемом дисковом носителе данных (например, CD-ROM) 225, который читается оптическим дисководом 212.

[00043] В некоторых случаях программы 233 приложения могут поставляться пользователю закодированными на одном или более дисках CD-ROM 225 и читаться с помощью соответствующего дисковода 212, или альтернативно могут быть считаны пользователем из сетей 220 или 222. Кроме того, программное обеспечение также может быть загружено в компьютерную систему 200 с других считываемых компьютером носителей. Считываемые компьютером носители данных означают любой невременной материальный носитель данных, который обеспечивает записанные инструкции и/или данные для компьютерной системы 200 для выполнения и/или обработки. Примеры таких носителей данных включают в себя дискеты, магнитную ленту, CD-ROM, DVD, диск blu-ray, жесткий диск, ROM или интегральную схему, память USB, магнитно-оптический диск или считываемую компьютером карту, такую как карта PCMCIA и т.п., независимо от того, являются ли такие устройства внутренними или внешними по отношению к компьютерному модулю 201. Примеры временных или нематериальных считываемых компьютером носителей (сред), которые также могут участвовать в процессе доставки программного обеспечения, приложений, инструкций и/или видеоданных или закодированных видеоданных к компьютерному модулю 401, включают в себя радио- или инфракрасные каналы передачи, а также сетевое соединение с другим компьютером или сетевым устройством, а также Интернет или сети Интранет, в том числе передачу электронной почты и информацию, записанную на веб-сайтах и т.п.

[00044] Вторая часть приложений 233 и соответствующие упомянутые выше модули кода могут выполняться для реализации одного или более графических интерфейсов пользователя (GUI), которые должны быть воспроизведены или иным образом представлены на устройстве 214 отображения. Посредством манипулирования обычно клавиатурой 202 и мышью 203, пользователь компьютерной системы 200 и приложение может управлять интерфейсом функционально адаптируемым способом для того, чтобы обеспечить команды управления и/или ввод для приложений, связанных с GUI. Также могут быть реализованы другие формы функционально адаптируемых пользовательских интерфейсов, такие как аудиоинтерфейс, использующий вывод речевых подсказок через громкоговорители 217 и речевой ввод пользовательских команд через микрофон 280.

[00045] Фиг.2В представляет собой подробную блок-схему процессора 205 и «памяти» 234. Память 234 представляет собой логическую агрегацию всех модулей памяти (в том числе жесткого диска 209 и полупроводниковой памяти 206), к которым может осуществлять доступ компьютерный модуль 201 на Фиг.2А.

[00046] Когда изначально включается питание компьютерного модуля 201, выполняется программа 250 самотестирования при включении питания (POST). Программа 250 POST обычно хранится в ROM 249 полупроводниковой памяти 206, изображенной на Фиг.2А. Аппаратное устройство, такое как ROM 249, хранящее программное обеспечение, иногда упоминается как программно-аппаратное средство. Программа 250 POST проверяет аппаратные средства компьютерного модуля 201 для того, чтобы гарантировать надлежащее функционирование, и обычно проверяет на корректность работы процессор 205, память 234 (209, 206), и модуль 251 программного обеспечения базовой системы ввода - вывода (BIOS), также обычно хранимый в ROM 249. Если программа 250 POST завершилась успешно, BIOS 251 активирует жесткий

диск 210, изображенный на Фиг.2А. Активация жесткого диска 210 запускает выполнение процессором 205 программы 252 загрузчика операционной системы, которая является резидентным объектом на жестком диске 210. Она загружает операционную систему 253 в память RAM 206, после чего операционная система 253 начинает работу.

5 Операционная система 253 является приложением системного уровня, выполняемым процессором 205 для того, чтобы выполнять различные высокоуровневые функции, включая управление процессором, управление памятью, управление устройствами, управление хранением, обеспечение интерфейса с приложениями, а также обеспечение универсального пользовательского интерфейса.

10 [00047] Операционная система 253 управляет памятью 234 (209, 206) для того, чтобы гарантировать, что каждый процесс или приложение, выполняющееся на компьютерном модуле 201, имеет достаточно памяти для выполнения и не создает коллизий с памятью, выделенной другому процессу. Кроме того, различные типы памяти, доступной в компьютерной системе 200, изображенной на Фиг.2А, должны использоваться так, чтобы каждый процесс мог работать эффективно. Соответственно, агрегированная
15 память 234 предназначена не для иллюстрации того, как выделяются определенные сегменты памяти (если не указано иное), а скорее для того, чтобы показать общий вид памяти, доступной для компьютерной системы 200, а также то, как такая память используется.

20 [00048] Как показано на Фиг.2В, процессор 205 включает в себя ряд функциональных модулей, в том числе блок 239 управления, арифметический логический блок (ALU) 240, а также локальную или внутреннюю память 248, иногда называемую кэш-памятью. Кэш-память 248 обычно включает в себя ряд регистров 244-246 в секции регистров. Одна или более внутренних шин 241 функционально взаимосвязывают эти
25 функциональные модули. Процессор 205 обычно также имеет один или более интерфейсов 242 для того, чтобы обмениваться информацией с внешними устройствами через системную шину 204, используя соединение 218. Память 234 соединена с шиной 204 с использованием соединения 219.

[00049] Приложение 233 включает в себя последовательность инструкций 231, которая
30 может включать в себя инструкции условного перехода и цикла. Программа 233 может также включать в себя данные 232, которые используются при выполнении программы 233. Инструкции 231 и данные 232 хранятся в областях памяти 228, 229, 230 и 235, 236, 237, соответственно. В зависимости от относительного размера инструкций 231 и областей памяти 228-230, конкретная инструкция может храниться в одной области
35 памяти, как изображено инструкцией, показанной в области памяти 230. Альтернативно, инструкция может быть сегментирована на несколько частей, каждая из которых хранится в отдельной области памяти, как изображено сегментами инструкции, показанными в областях памяти 228 и 229.

[00050] В большинстве случаев процессору 205 дается набор инструкций, которые в
40 нем выполняются. Процессор 205 ожидает последующего ввода, на который процессор 205 реагирует путем выполнения другого набора инструкций. Каждый ввод может быть обеспечен от одного или более из ряда источников, в том числе данных, генерируемых одним или более устройствами ввода 202, 203, данных, принимаемых от внешнего источника через одну из сетей 220, 202, данных, принимаемых от одного из устройств
45 хранения данных 206, 209, или данных, принимаемых от носителя данных 225, вставленного в соответствующее средство 212 чтения, которые все изображены на Фиг.2А. Выполнение набора инструкций может в некоторых случаях привести к выводу данных. Выполнение может также включать в себя сохранение данных или переменных

в памяти 234.

[00051] Видеокодер 114, декодер 134 и описанные способы могут использовать входные переменные 254, которые хранятся в памяти 234 в соответствующих областях памяти 255, 256, 257. Видеокодер 114, декодер 134 и описанные способы могут
5 создавать выходные переменные 261, которые хранятся в памяти 234 в соответствующих областях памяти 262, 263, 264. Промежуточные переменные 258 могут храниться в областях памяти 259, 260, 266 и 267.

[00052] Что касается процессора 205, изображенного на Фиг.2В, регистры 244, 245, 246, арифметический логический блок (ALU) 240, и блок 239 управления работают
10 совместно для того, чтобы выполнять последовательности микроопераций, необходимых для выполнения циклов «выборка-декодирование-выполнение» для каждой инструкции в наборе инструкций, составляющем программу 233. Каждый цикл «выборка-декодирование-выполнение» включает в себя:

(а) операцию выборки, которая выбирает или считывает инструкцию 231 из области
15 памяти 228, 229, 230;

(b) операцию декодирования, в которой блок 239 управления определяет, какая именно инструкция была выбрана; и

(с) операцию выполнения, в которой блок 239 управления и/или ALU 240 выполняет эту инструкцию.

[00053] После этого может быть выполнен следующий цикл «выборка-декодирование-выполнение» для следующей инструкции. Аналогичным образом может быть выполнен цикл запоминания, с помощью которого блок 239 управления сохраняет или записывает значение в область памяти 232.

[00054] Каждый этап или подпроцесс в процессе, изображенном на Фиг. 10-13, которые
25 будут описаны позже, связаны с одним или более сегментами программы 233 и обычно выполняются совместно секцией регистра 244, 245, 247, арифметико-логическим блоком 240 и блоком 239 управления в процессоре 205 так, чтобы выполнять циклы выборки, декодирования и выполнения для каждой инструкции в наборе инструкций для отмеченных сегментов программы 233.

[00055] Фиг.3 представляет собой блок-схему, показывающую функциональные модули видеокодера 114. Фиг.4 представляет собой блок-схему, показывающую функциональные модули декодера 134. Видеокодер 114 и декодер 134 могут быть реализованы с использованием компьютерной системы 200 общего назначения, как показано на Фиг. 2А и 2В, где различные функциональные модули могут быть
35 реализованы выделенными аппаратными средствами в пределах компьютерной системы 200, исполнимым программным обеспечением в пределах компьютерной системы 200, таким как один или более модулей программного кода прикладной программы 233, находящейся на жестком диске 205 и выполняемой процессором 205, или альтернативно комбинацией выделенного аппаратного и программного обеспечения в пределах
40 компьютерной системы 200. Видеокодер 114, декодер 134 и описанные способы альтернативно могут быть реализованы в выделенных аппаратных средствах, таких как одна или более интегральных схем, выполняющих функции или подфункции описанных способов. Такие выделенные аппаратные средства могут включать в себя графические процессоры, цифровые сигнальные процессоры, специализированные
45 интегральные схемы (ASIC), программируемые пользователем логические матрицы (FPGA) или один или более микропроцессоров с соответствующей связанной с ними памятью. В частности, видеокодер 114 включает в себя модули 320-344, а декодер 134 включает в себя модули 420-434, каждый из которых может быть реализован как

один или более модулей программного кода программы 233 приложения.

[00056] Хотя видеокодер 114, изображенный на Фиг.3, является примером конвейера высокоэффективного видеокодирования (HEVC), этапы обработки, выполняемые модулями 320-344, являются характерными для других видеокодеров, таких как VC-1 или H.264/MPEG-4 AVC. Видеокодер 114 принимает данные захваченного кадра как серию кадров, каждый из которых включает в себя один или более цветовых каналов. Каждый кадр включает в себя одну сетку отсчетов на цветовой канал. Информация о цвете представляется с использованием «цветового пространства», такого как пространство в соответствии с рекомендациями ITU-R BT.709 («YUV»), хотя также возможны другие цветовые пространства. Когда используется цветовое пространство YUV, цветовые каналы включают в себя канал яркости («Y») и два канала цветности («U» и «V»). Более того, различное количество информации может быть включено в сетку отсчетов каждого цветового канала в зависимости от дискретизации изображения или приложения фильтрации, используемого для передискретизации данных захваченного кадра. Существует несколько подходов к дискретизации, известных как «форматы цветности», некоторые из которых будут описаны со ссылками на Фиг. 5А и Фиг. 5В.

[00057] Видеокодер 114 делит каждый кадр данных захваченного кадра, таких как данные 310 кадра, на области, обычно называемые «блоками дерева кодирования» (CTB). Каждый блок дерева кодирования (CTB) включает в себя иерархическое подразделение с использованием дерева квадрантов части кадра на совокупность «единиц кодирования» (CU). Блок дерева кодирования (CTB) обычно занимает область из 64×64 отсчетов сигнала яркости, хотя возможны и другие размеры, такие как 16×16 или 32×32. В некоторых случаях могут использоваться еще большие размеры, такие как 128×128. Блок дерева кодирования (CTB) может быть подразделен посредством разделения на четыре области равного размера для того, чтобы создать новый уровень иерархии. Разделение может быть применено рекурсивно, что приводит к иерархии дерева квадрантов. Поскольку размеры сторон блока дерева кодирования (CTB) всегда являются степенями числа два, и разделение с использованием дерева квадрантов всегда приводит к делению ширины и высоты на два, размеры сторон области также всегда являются степенями числа два. Когда дальнейшее разделение области не выполняется, говорят, что в области существует «единица кодирования» (CU). Когда разделение не выполняется на верхнем уровне блока дерева кодирования, область, занимающая весь блок дерева кодирования, содержит одну единицу кодирования (CU), которая обычно упоминается как «наибольшая единица кодирования» (LCU). Также существует минимальный размер для каждой единицы кодирования, такой как область, занимаемая отсчетами яркости 8×8, хотя также возможны другие минимальные размеры. Единицы кодирования такого размера обычно упоминаются как «наименьшие единицы кодирования» (SCU). В результате этой иерархии дерева квадрантов весь блок дерева кодирования (CTB) занимается одним или более единицами кодирования (CU).

[00058] Видеокодер 114 производит один или более массивов отсчетов, обычно называемых «единицами предсказания» (PU) для каждой единицы кодирования (CU). Возможны различные компоновки единиц предсказания (PU) в каждой единице кодирования (CU), с тем требованием, чтобы единицы предсказания (PU) не накладывались друг на друга и чтобы вся единица кодирования (CU) была занята одной или более единицами предсказания (PU). Эта схема гарантирует, что единицы предсказания (PU) покрывают всю область кадра.

[00059] Видеокодер 114 работает путем вывода из модуля 340 мультиплексора

единицы 382 предсказания (PU). Модуль 344 различия выводит различие между единицей 382 предсказания (PU) и соответствующим двумерным массивом отсчетов данных из единицы кодирования (CU) блока дерева кодирования (СТВ) данных 310 кадра, причем различие известно как «массив остаточных отсчетов» 360. Массив 360 остаточных отсчетов из модуля 344 различия принимается модулем 320 преобразования, который преобразует (или «кодирует») массив 360 остаточных отсчетов из пространственного представления в представление в частотной области путем применения «прямого преобразования». Модуль 320 преобразования создает коэффициенты 362 преобразования для каждого преобразования в единице преобразования (TU) в иерархическом подразделении единицы кодирования (CU) на одну или более единиц преобразования (TU), обычно называемых «деревом преобразований». Для разрабатываемого стандарта высокоэффективного видеокодирования (HEVC) преобразование в представление в частотной области реализуется с использованием модифицированного дискретного косинусного преобразования (DCT), в котором традиционное DCT модифицируется так, чтобы оно было реализованным с использованием сдвига и сложения. Возможны различные размеры для массива 360 остаточных отсчетов и коэффициентов 362 преобразования в соответствии с поддерживаемыми размерами преобразования. В разрабатываемом стандарте высокоэффективного видеокодирования (HEVC) преобразования выполняются на двумерных массивах отсчетов, имеющих конкретные размеры, такие как 32×32 , 16×16 , 8×8 и 4×4 . Таким образом, можно сказать, что существует предопределенный набор размеров преобразования, доступных видеокодеру 114. Кроме того, как упоминалось выше, набор размеров преобразования может различаться между каналом яркости и каналами цветности. Двумерные преобразования обычно конфигурируются так, чтобы они были «разделяемыми», позволяя реализацию в виде первого набора одномерных преобразований, выполняемого на двумерном массиве отсчетов в одном направлении (например, по строкам), с последующим вторым набором одномерных преобразований, выполняемым на двумерном массиве отсчетов, являющемся выходом первого набора одномерных преобразований, в другом направлении (например, по столбцам). Преобразования, имеющие одну и ту же ширину и высоту, обычно упоминаются как «квадратные преобразования». Дополнительные преобразования, имеющие отличающиеся ширины и высоты также возможны, и обычно упоминаются как «неквадратные преобразования». Оптимизированные реализации преобразований могут объединять одномерные преобразования строк и столбцов в конкретные аппаратные или программные модули, такие как модуль преобразования 4×4 или модуль преобразования 8×8 . Преобразования, имеющие более высокие размерности, требуют более сложных схем для реализации, даже при том, что они могут использоваться нечасто. Соответственно, в разрабатываемом стандарте высокоэффективного видеокодирования (HEVC) существует максимальный размер преобразования 32×32 . Интегрированная природа реализации преобразования также добавляет предпочтительность сокращения количества поддерживаемых размеров неквадратных преобразований, поскольку они будут обычно требовать реализации полностью новых аппаратных средств вместо того, чтобы повторно использовать существующую одномерную логику преобразования из соответствующих квадратных преобразований. Преобразования применяются как к каналам яркости, так и к каналам цветности. Различия между обработкой каналов яркости и каналов цветности относительно единиц преобразования (TU) существуют и будут обсуждены ниже со ссылками на Фиг. 5А и Фиг. 5В. Каждое дерево преобразований занимает одна единица

кодирования (CU) и определяется как разложение единицы кодирования (CU) в иерархию дерева квадрантов, содержащую одну единицу преобразования (TU) в каждом узле листа иерархии дерева преобразований (дерева квадрантов), при этом каждая единица преобразования (TU) может использовать преобразования поддерживаемых размеров преобразования. Аналогично блоку дерева кодирования (СТВ), необходимо, чтобы вся единица кодирования (CU) была занята одним или более единицами преобразования (TU). На каждом уровне иерархии квадрантов дерева преобразований «значение флага кодированного блока» сигнализирует возможное присутствие преобразования в каждом цветовом канале, либо на текущем уровне иерархии, когда никаких дальнейших разделений нет, либо сигнализирует о том, что более низкие уровни иерархии могут содержать по меньшей мере одно преобразование среди получающихся единиц преобразования (TU). Когда значение флага кодированного блока равно нулю, никаких преобразований не выполняется для соответствующего цветового канала любой единицы преобразования (TU) дерева преобразований, либо на текущем иерархическом уровне, либо на более низких иерархических уровнях. Когда значение флага кодированного блока равно единице, область содержит преобразование, которое должно иметь по меньшей мере один ненулевой остаточный коэффициент. Таким образом для каждого цветового канала ноль или больше преобразований могут покрыть часть области единицы кодирования (CU), изменяющуюся от нуля вплоть до всей единицы кодирования (CU). Для каждого цветового канала существуют отдельные кодированные значения флага блока. Каждое значение флага кодированного блока не обязано быть закодированным, поскольку существуют случаи, в которых имеется только одно возможное значение флага кодированного блока.

[00060] Коэффициенты 362 преобразования затем вводятся в модуль 322 масштабирования и квантования и масштабируются и квантуются согласно определенному параметру 384 квантования для того, чтобы получить массив 364 остаточных коэффициентов. Процесс масштабирования и квантования приводит к потере точности, зависящей от значения определенного параметра 384 квантования. Более высокое значение определенного параметра 384 квантования приводит к большей потере информации из коэффициентов преобразования. Это увеличивает степень сжатия, достигаемую видеокодером 114, за счет снижения визуального качества вывода из видеодекодера 134. Определенный параметр 384 квантования может быть адаптирован во время кодирования каждого кадра данных 310 кадра, или он может быть зафиксирован для части данных кадра 310, таких как весь кадр. Также возможны другие адаптации определенного параметра 384 квантования, такие как квантование различных остаточных коэффициентов с отдельными значениями. Массив 364 остаточных коэффициентов и определенный параметр 384 квантования берутся в качестве входа для модуля 326 обратного масштабирования, который инвертирует масштабирование, выполненное модулем 322 масштабирования и квантования, для того, чтобы получить массивы 366 перемасштабированных коэффициентов преобразования, которые являются перемасштабированными версиями массива 364 остаточных коэффициентов.

[00061] Массив 364 остаточных коэффициентов и определенный параметр 384 квантования также берутся в качестве входа для модуля 324 энтропийного кодера, который кодирует остаточные коэффициенты в закодированном потоке битов 312 (или в «битовом видеопотоке»). Массивы 364 остаточных коэффициентов каждого преобразования в каждой единице преобразования (TU) кодируются в группах, обычно называемых «подблоками». Подблоки предпочтительно должны иметь одинаковые размеры независимо от размера преобразования, поскольку это позволяет повторно

использовать логику, относящуюся к обработке подблока. Остаточные коэффициенты в пределах одного подблока обычно упоминаются как «группа коэффициентов», и для каждой группы коэффициентов обычно кодируется флаг группы коэффициентов для того, чтобы указать, является ли по меньшей мере один остаточный коэффициент в пределах группы коэффициентов ненулевым. В некоторых случаях флаг группы коэффициентов может быть выведен логически, и таким образом не кодируется. Флаг кодируется для каждого остаточного коэффициента, принадлежащего группе коэффициентов, имеющей значение флага группы коэффициентов, равное единице, чтобы указать, что остаточный коэффициент является ненулевым («значащим»), или нулю, чтобы указать, что остаточный коэффициент является нулевым («незначащим»). Из-за потери точности в результате работы модуля 322 масштабирования и квантования массивы 366 перемасштабированных коэффициентов преобразования не идентичны исходным коэффициентам 362 преобразования. Массивы 366 перемасштабированных коэффициентов преобразования из модуля 326 обратного масштабирования затем выводятся в модуль 328 обратного преобразования. Модуль 328 обратного преобразования выполняет обратное преобразование из частотной области в пространственную область для того, чтобы получить представление 368 в пространственной области массивов 366 перемасштабированных коэффициентов преобразования, идентичных представлению в пространственной области, которое формируется в видеодекодере 134.

[00062] Модуль 338 оценки движения формирует векторы 374 движения путем сравнения данных 310 кадра с предыдущими данными кадра из одного или более наборов кадров, сохраненных в модуле 332 буфера кадров, обычно конфигурируемом внутри памяти 206. Наборы кадров известны как «списки опорных изображений». Векторы 374 движения затем вводятся в модуль 334 компенсации движения, который формирует единицу 376 предсказания (PU) путем фильтрации отсчетов, сохраненных в модуле 332 буфера кадров, принимая во внимание пространственное смещение, полученное из векторов 374 движения. Непоказанные на Фиг.3 векторы 374 движения также передаются в качестве синтаксических элементов модулю 324 энтропийного кодера для кодирования в закодированном потоке битов 312. Модуль 336 внутрикадрового предсказания формирует внутренне(интра)-предсказанную единицу 378 предсказания (PU) с использованием отсчетов 370, полученных из модуля 342 суммирования, который суммирует единицу 382 предсказания (PU) из модуля 340 мультиплексора и представление 368 в пространственной области из модуля 328 обратного преобразования. Модуль 336 внутрикадрового предсказания также формирует режим 380 внутреннего предсказания, который отправляется энтропийному кодери 324 для кодирования в закодированный поток битов 312.

[00063] Единицы предсказания (PU) могут быть сгенерированы с использованием либо способа внутреннего предсказания, либо способа внешнего предсказания. Способы внутреннего предсказания используют отсчеты, смежные с единицей предсказания (PU), которые ранее были декодированы (обычно выше и левее от единицы предсказания) для того, чтобы сгенерировать опорные отсчеты в пределах единицы предсказания (PU). Возможны различные направления внутреннего предсказания, называемые «режимом внутреннего предсказания». Способы внешнего предсказания используют вектор движения для того, чтобы обратиться к блоку из выбранного опорного кадра. Поскольку блок может иметь любое выравнивание вплоть до дробной точности дискретизации, например, одной восьмой отсчета, необходима фильтрация для того, чтобы создать блок опорных отсчетов для единицы предсказания (PU). Решение о том,

какой способ использовать, принимается в соответствии с компромиссом скорость-искажение между желаемым битрейтом получающегося закодированного битового потока 312 и величиной искажения качества изображения, вводимого способом внутреннего предсказания или способом внешнего предсказания. Если используется внутреннее предсказание, один режим внутреннего предсказания выбирается из набора возможных режимов внутреннего предсказания, также в соответствии с компромиссом скорость-искажение. Модуль 340 мультиплексора выбирает либо внутренне-предсказанные опорные отсчеты 378 из модуля 336 внутрикадрового предсказания, либо внешне(интер)-предсказанную единицу 376 предсказания (PU) из блока 334 компенсации движения, в зависимости от решения, принятого алгоритмом компромисса скорость-искажение. Модуль 342 суммирования формирует сумму 370, которая вводится в модуль 330 деблокирующего фильтра. Модуль 330 деблокирующего фильтра выполняет фильтрацию вдоль границ блока, формируя деблокированные отсчеты 372, которые записываются в модуль 332 буфера кадров, конфигурируемый в пределах памяти 206. Модуль 332 буфера кадров является буфером с достаточной емкостью для того, чтобы удержать данные из одного или более прошлых кадров для обращения (ссылки) в будущем в качестве части списка опорных изображений.

[00064] Для разрабатываемого стандарта высокоэффективного видеокодирования (HEVC) закодированный поток битов 312, произведенный энтропийным кодером 324 описывается единицами уровня сетевой абстракции (NAL). Обычно, каждая часть кадра содержится в одной единице NAL. Энтропийный кодер 324 кодирует массив 364 остаточных коэффициентов, режим 380 внутреннего предсказания, векторы движения и другие параметры, совокупно называемые «синтаксическими элементами», в закодированный поток битов 312 путем выполнения алгоритма контекстно-адаптивного двоичного арифметического кодирования (CABAC). Синтаксические элементы группируются в «синтаксические структуры», и эти группировки могут содержать рекурсию для описания иерархических структур. В дополнение к порядковым значениям, таким как режим внутреннего предсказания, или целочисленным значениям, таким как вектор движения, синтаксические элементы также включают в себя флаги, например, указывающие разделение дерева квадрантов. Модуль 338 оценки движения и модуль 334 компенсации движения производят операцию над векторами 374 движения, с точностью в 1/8 отсчета сигнала яркости, что позволяет точно моделировать движения между кадрами в данных 310 кадра.

[00065] Хотя видеodeкодер 134, изображенный на Фиг.4, описывается в отношении декодирующего конвейера высокоэффективного видеокодирования (HEVC), этапы обработки, выполняемые модулями 420-434, характерны и для других видео кодеков, которые используют энтропийное кодирование, таких как H.264/MPEG-4 AVC, MPEG-2 и VC-1. Закодированная видеoinформация также может быть считана из памяти 206, с жесткого диска 210, с CD-ROM, с диска Blu-ray™ или с другого считываемого компьютером носителя данных. Альтернативно закодированная видеoinформация может быть принята от внешнего источника, такого как сервер, соединенный с сетью 220 связи или радиочастотный приемник.

[00066] Как видно на Фиг.4, принятые видеоданные, такие как закодированный поток битов 312, вводятся в видеodeкодер 134. Закодированный поток битов 312 может быть считан из памяти 206, с жесткого диска 210, с CD-ROM, с диска Blu-ray™ или с другого считываемого компьютером носителя данных. Альтернативно закодированный поток битов 312 может быть принят от внешнего источника, такого как сервер, соединенный с сетью 220 связи или радиоприемник. Закодированный поток битов 312

содержит закодированные синтаксические элементы, представляющие данные захваченного кадра, которые должны быть декодированы.

[00067] Закодированный поток битов 312 вводится в модуль 420 энтропийного декодера, который извлекает синтаксические элементы из закодированного потока битов 312 и передает значения синтаксических элементов другим блокам в видеодекодере 134. Модуль 420 энтропийного декодера применяет алгоритм контекстно-адаптивного двоичного арифметического кодирования (САВАС) для декодирования синтаксических элементов из закодированного потока битов 312. Декодированные синтаксические элементы используются для восстановления параметров в видеодекодере 134. Параметры включают в себя ноль или больше из массива 450 остаточных коэффициентов, векторов 452 движения и режима 454 предсказания. Массивы 450 остаточных коэффициентов передаются модулю 422 обратного масштабирования и преобразования, векторы 452 движения передаются модулю 434 компенсации движения, и режим 454 предсказания передается модулю 426 внутрикадрового предсказания и мультиплексору 428. Модуль 422 обратного масштабирования и преобразования выполняет обратное масштабирование над данными остаточных коэффициентов для того, чтобы создать восстановленные коэффициенты преобразования. Модуль 422 обратного масштабирования и преобразования затем применяет «обратное преобразование» для того, чтобы преобразовать (или «декодировать») восстановленные коэффициенты преобразования из представления в частотной области в представление в пространственной области, формируя массив 456 остаточных отсчетов. Обратное преобразование в модуле 422 обратного масштабирования и преобразования выполняет ту же самую операцию, что и обратное преобразование 328. Модуль 422 обратного масштабирования и преобразования поэтому должен быть выполнен с возможностью обеспечения predetermined набора размеров преобразования, требуемых для декодирования закодированного потока битов 312, который совместим с разрабатываемым стандартом высокоэффективного видеокодирования (HEVC).

[00068] Модуль 434 компенсации движения использует векторы 452 движения из модуля 420 энтропийного декодера, объединенные с данными 460 опорного кадра из блока 432 буфера кадров, сконфигурированного в памяти 206, для формирования внешне-предсказанной единицы 462 предсказания (PU) для единицы предсказания (PU), являющегося прогнозом данных выходного декодированного кадра. Когда режим 454 предсказания указывает, что текущий блок прогноза был закодирован с использованием внутреннего предсказания, модуль 426 внутрикадрового предсказания формирует внутренне-предсказанную единицу 464 предсказания (PU) для единицы предсказания (PU) с использованием отсчетов, пространственно граничащих с единицей предсказания (PU) и направлением прогноза, также обеспечиваемым режимом 454 предсказания. Пространственно соседние отсчеты получают из суммы 458, являющейся выходом модуля 424 суммирования. Модуль 428 мультиплексора выбирает внутренне-предсказанную единицу 464 предсказания (PU) или внешне-предсказанную единицу 462 предсказания (PU) для единицы 466 предсказания (PU) в зависимости от текущего режима 454 предсказания. Единица 466 предсказания (PU), выводимая из модуля 428 мультиплексора, добавляется к массиву 456 остаточных отсчетов из модуля 422 обратного масштабирования и преобразования с помощью модуля 424 суммирования для того, чтобы сформировать сумму 458, которая затем вводится в каждый из модуля 430 деблокирующего фильтра и модуля 426 внутрикадрового предсказания. Модуль 430 деблокирующего фильтра выполняет фильтрацию вдоль границ блока данных, таких как границы единицы преобразования (TU), для того, чтобы сгладить видимые

артефакты. Выходные данные модуля 430 деблокирующего фильтра записываются в модуль 432 буфера кадров, конфигурируемый в памяти 206. Модуль 432 буфера кадров обеспечивает достаточный объем памяти для того, чтобы сохранить один или более декодированных кадров для обращения в будущем. Декодированные кадры 412 также выводятся из модуля 432 буфера кадров на устройство отображения, такое как устройство 136 отображения.

[00069] Фиг. 5А и 5В показывают сетки отсчетов части 500 кадра и части 510 кадра, закодированных с использованием формата цветности 4:2:0 и 4:2:2 соответственно. Формат цветности определяется как параметр конфигурации видеокодера 114, и видеокодер 114 кодирует в закодированный поток битов 312 синтаксический элемент «chroma_format_idc», который определяет формат цветности. Видеodeкодер 134 декодирует синтаксический элемент «chroma_format_idc» из закодированного потока битов 312, чтобы определить используемый формат цветности. Например, когда используется формат цветности 4:2:0, значение chroma_format_idc равно единице, когда используется формат цветности 4:2:2, значение chroma_format_idc равно двум, и когда используется формат цветности 4:4:4, значение chroma_format_idc равно трем. На Фиг. 5А и 5В положения отсчетов сигнала яркости, такие как положение 501 отсчета сигнала яркости, иллюстрируются с использованием символов «Х», а положения отсчетов сигнала цветности, такие как положение 502 отсчета сигнала цветности, иллюстрируются с использованием символов «О». Путем дискретизации части 500 кадра в обозначенных точках получается сетка отсчетов для каждого цветового канала, когда применяется формат цветности 4:2:0. В каждом положении Х отсчета сигнала яркости осуществляется дискретизация канала яркости («Y»), и в каждом положении О отсчета сигнала цветности осуществляется дискретизация обоих каналов цветности («U» и «V»). Как показано на Фиг.5А, для каждого положения отсчета сигнала цветности существует компоновка 2×2 положений отсчетов сигнала яркости. Путем дискретизации отсчетов сигнала яркости в положениях отсчетов сигнала яркости и отсчетов сигнала цветности в положениях отсчетов сигнала цветности, обозначенных в части 510 кадра, получается сетка отсчетов для каждого цветового канала, когда применяется формат цветности 4:2:2. То же самое размещение отсчетов для цветовых каналов, что и в части 500 кадра, делается для части 510 кадра. В отличие от части 500 кадра, в части 510 кадра существует вдвое больше положений отсчетов сигнала цветности. В части 510 кадра положения отсчетов сигнала цветности совмещаются с каждым вторым положением отсчета сигнала яркости. Соответственно, на Фиг.5В для каждого положения отсчета сигнала цветности существует компоновка 2×1 положений отсчетов сигнала яркости.

[00070] Различные допустимые размеры единиц преобразования были описаны выше в единицах отсчетов сигнала яркости. Область, покрываемая преобразованием, применяемым к каналу яркости, будет таким образом иметь те же самые размеры, что и размеры единицы преобразования. Поскольку единицы преобразования также кодируют каналы цветности, применяемое преобразование для каждого канала цветности будет иметь размеры в соответствии с конкретным используемым форматом цветности. Например, когда используется формат цветности 4:2:0, единица преобразования (TU) размера 16×16 будет использовать преобразование размера 16×16 для канала яркости и преобразование размера 8×8 для каждого канала цветности. Одним особым случаем является случай, когда для канала яркости используется преобразование размера 4×4, и нет никакого соответствующего преобразования размера 2×2 (когда применяется формат цветности 4:2:0) или нет никакого соответствующего преобразования размера 4×2 (когда применяется формат цветности 4:2:2), которые

могли бы использоваться для каналов цветности. В этом особом случае преобразование размера 4×4 для каждого канала цветности может покрывать область, занимаемую несколькими преобразованиями яркости.

[00071] Фиг.6А изображает схематическое представление примерного дерева преобразований единицы 602 кодирования (CU) (обозначенной жирной границей) в пределах блока 600 дерева кодирования (СТВ) кадра. Единственное подразделение дерева квадрантов делит блок 600 дерева кодирования (СТВ) на четыре единицы кодирования (CU) размера 32×32, такие как единица 602 кодирования (CU). Примерное дерево преобразований существует внутри единицы 602 кодирования (CU). Примерное дерево преобразований включает в себя несколько подразделений дерева квадрантов, приводящее к десяти единицам преобразования (TU), пронумерованным на Фиг.6А от 1 до 10, например единица 604 преобразования #9 (TU). Единицы преобразования #1-#10 покрывают всю единицу 602 кодирования (CU). Каждое подразделение дерева квадрантов делит область пространственно на четыре квадранта, что дает четыре меньших области. Каждая единица преобразования (TU) имеет значение глубины преобразования, соответствующее иерархическому уровню единицы преобразования (TU) в пределах дерева преобразований. Иерархический уровень указывает количество подразделений дерева квадрантов, выполненных до завершения деления дерева квадрантов, приведшего к экземпляру единицы преобразования (TU), который занимает соответствующую область. Например, единица 604 преобразования #9 (TU) занимает одну четверть области единицы 602 кодирования (CU), и поэтому имеет глубину преобразования, равную единице. Каждая единица преобразования (TU) имеет связанный размер (или «размер преобразования»), обычно описываемый как размеры области, содержащей единицу преобразования (TU) на сетке отсчетов сигнала яркости. Размер зависит от размера единицы кодирования (CU) и глубины преобразования. Единицы преобразования (TU) с нулевой глубиной преобразования имеют размер, равный размеру соответствующей единицы кодирования (CU). Каждое приращение глубины преобразования приводит к делению на два размера единиц преобразования (TU), существующих в дереве преобразований на данной глубине преобразования. Поскольку кадр включает в себя канал яркости и каналы цветности, единица 602 кодирования (CU) занимает область как на сетке отсчетов сигнала яркости, так и на сетке отсчетов сигнала цветности, и таким образом каждая единица преобразования (TU) включает в себя информацию, описывающую как отсчеты сигнала яркости на сетке отсчетов сигнала цветности, так и отсчеты сигнала цветности на сетке отсчетов сигнала цветности.

Природа информации для каждой единицы преобразования (TU) зависит от этапа обработки видеокодера 114 или видеodeкодера 134. На входе модуля 320 преобразования и выходе модуля 422 обратного масштабирования и преобразования массив остаточных отсчетов 360 и 456 соответственно содержат информацию для каждой единицы преобразования (TU) в пространственной области. Массив остаточных отсчетов 360 и 456 может быть далее разделен на «массив остаточных отсчетов сигнала цветности» и «массив остаточных отсчетов сигнала яркости», вследствие различий в обработке между каналом яркости и каналами цветности. На выходе модуля 322 масштабирования и квантования и на входе модуля 422 обратного масштабирования и преобразования массив остаточных коэффициентов 364 и 450 соответственно содержат информацию для каждой единицы преобразования (TU) в частотной области. Массив остаточных коэффициентов 364 и 450 может быть далее разделен на «массив остаточных коэффициентов сигнала цветности» и «массив остаточных коэффициентов сигнала яркости», вследствие различий в обработке между каналом яркости и каналами

цветности.

[00072] Фиг.6В иллюстрирует примерное дерево 630 преобразования, соответствующее примерному дереву преобразований, изображенному на Фиг.6А, для канала яркости единицы кодирования (CU) размером 32×32, содержащего набор единиц преобразования (TU) и занимающего единицу 602 кодирования (CU), которая занимает массив отсчетов сигнала яркости размером 32×32 на сетке отсчетов сигнала яркости. Фиг.7 иллюстрирует структуру 700 данных, которая представляет примерное дерево 630 преобразований. На Фиг.6В квадраты 1-10 указывают, единицы преобразования, присутствующие в пределах области 632 (проиллюстрировано несколькими единицами 640 преобразования (TU)), и каждый квадрат содержится в области, которая больше не подразделяется (обозначено квадратом с пунктирной границей).

[00073] На Фиг.6В квадраты под номерами 1 и 9 содержат преобразования размера 16×16 для канала яркости, квадраты под номерами 2, 3 и 8 содержат преобразования размера 8×8 для канала яркости, и квадраты под номерами 4-7 содержат преобразования размера 4×4 для канала яркости. Соответствующая область (пунктирный квадрат) для каждого из этих квадратов имеет значение флага кодированного блока, равное единице, для того, чтобы указать на присутствие преобразования.

[00074] Присутствие или отсутствие преобразования для каждого цветового канала определяется отдельным значением флага кодированного блока, которое используется при кодировании и декодировании потока битов, но которое не обязательно должно передаваться в потоке битов, как будет обсуждено ниже. Следовательно, количество массивов 450 остаточных коэффициентов, выводимых из энтропийного декодера 420, зависит от значений флага кодированного блока. Когда ни в одном цветовом канале нет значимых коэффициентов, количество массивов 450 остаточных коэффициентов, выводимых из энтропийного декодера 420, равно нулю.

[00075] На Фиг.7 круги представляют значения флага преобразования разделения, где само значение флага преобразования разделения обозначено в соответствующем круге. На Фиг.7 треугольники представляют кодированные значения флага блока, где само значение флага кодированного блока обозначено в соответствующем треугольнике. Квадраты представляют блоки преобразования, причем каждое преобразование пронумеровано в соответствии с нумерацией преобразований, изображенной на Фиг.6В.

[00076] Самый верхний иерархический уровень примерного дерева 630 преобразования содержит область 632, занимающую единицу кодирования (CU) размером 32×32. Значение 702 флага преобразования разделения указывает, что область 632 подразделяется на четыре области размера 16×16, такие как область 634, таким образом определяя «неконцевой» узел примерного дерева 630 преобразования. Для каждой области размера 16×16 дополнительное значение флага преобразования разделения, такое как значение 704 флага преобразования разделения, указывает, что соответствующая область размера 16×16 должна быть далее подразделена на четыре области размера 8×8. Например, область 634 далее не подразделяется, что обозначено нулевым значением 704 флага преобразования разделения, определяющего таким образом «концевой» узел примерного дерева 630 преобразования. В противоположность этому, область 638 далее подразделяется на четыре области размером 4×4 (такие как область 636), как обозначено значением 712 флага преобразования разделения, равным единице. Рекурсивная структура разделения, существующая в дереве 630 преобразования, является аналогичной разделению дерева квадрантов, существующему в блоке дерева кодирования (СТВ). Для канала яркости в «концевых» узлах дерева квадрантов присутствие преобразования в единице преобразования (TU) сигнализируется значением

флага кодированного блока, например, значение 708 флага кодированного блока, равное единице, указывает на присутствие преобразования 710 в области 634.

[00077] Поскольку преобразование может использоваться для представления остаточных данных в каждой области, области не могут быть меньше, чем наименьший поддерживаемый размер преобразования, такой как 4×4 отсчета сигнала яркости для канала яркости. Дополнительно к этому, для областей, размер которых больше чем наибольший доступный размер преобразования, значение флага преобразования разделения автоматически устанавливается равным единице. Например, для дерева преобразований с верхним уровнем единицы кодирования 64×64 происходит автоматическое подразделение (то есть не сигнализируемое в закодированном потоке битов 312) на четыре области размера 32×32, когда наибольший поддерживаемый размер преобразования составляет 32×32 отсчета сигнала яркости.

[00078] Нижняя правая 16×16 область 642 содержит единицу преобразования (TU) (обозначенную цифрой 10 (десять) и заштрихованную) без преобразования для канала яркости, и поэтому имеет соответствующее значение 716 флага кодированного блока, равное нулю.

[00079] Фиг. 6С и Фиг. 8 иллюстрируют примерное дерево 630 преобразований, соответствующее примерному дереву преобразований, изображенному на Фиг.6А, для канала цветности, сконфигурированное для формата цветности 4:2:2 и содержащее набор преобразований для канала цветности, соответствующий дереву 630 преобразований для канала яркости и представленный структурой 800 данных. Поскольку иерархия дерева преобразований является общей на основании структуры Фиг.6А между каналом яркости и каналами цветности, значения флага преобразования разделения используются совместно структурами данных 700 и 800. В отличие от структуры 700 данных, структура 800 данных включает в себя значение флага кодированного блока для каждого значения флага преобразования разделения, равного единице (то есть на неконцевых узлах дерева преобразований). Например, значение 802 флага кодированного блока, равное единице, связывается с флагом 702 преобразования разделения. Если значение флага кодированного блока на неконцевом узле дерева преобразований равно нулю, значения флага кодированного блока на дочерних узлах логически устанавливаются равными нулю (и никакие соответствующие флаги кодированного блока не кодируются в закодированном потоке битов 312). Значения флага кодированного блока в неконцевых областях позволяют прекратить кодирование флагов кодированного блока на нижних уровнях дерева преобразований для каждого канала цветности, если никаких значимых остаточных коэффициентов не присутствует ни в одной из дочерних областей, даже при том, что в канале яркости могут присутствовать значимые остаточные коэффициенты. Это является общей ситуацией для типичных данных захваченных кадров, поскольку большинство информации присутствует в канале яркости.

[00080] Когда видеокодер 114 и видеodeкодер 134 конфигурируются для формата цветности 4:4:4, область цветности каждого канала цветности любой данной единицы преобразования (TU) с размером, который не является одним из предопределенного набора размеров единицы преобразования (TU), имеет размеры, идентичные областям яркости данной единицы преобразования (TU) (то есть когда логически определяемое разделение не имеет места). Когда видеокодер 114 и видеodeкодер 134 конфигурируются для формата цветности 4:4:4, область цветности каждого канала цветности любой данной единицы преобразования (TU) с размером, который является одним из предопределенного набора размеров единицы преобразования (TU), имеет размеры,

меньшие чем области яркости данной единицы преобразования (TU) (то есть когда логически определяемое разделение имеет место).

[00081] Когда используется формат цветности 4:2:2, это приводит к единице 602 кодирования (CU), включающей в себя изображенную на Фиг.6С 16×32 область 662 отсчетов сигнала цветности для каждого канала цветности и таким образом занимающему область 16×32 на сетке отсчетов сигнала цветности. Фиг.6С иллюстрирует области на сетке отсчетов сигнала цветности, изображенной как массив отсчетов сигнала цветности, в котором каждый отсчет сигнала цветности изображен в одинаковом масштабе по горизонтали и по вертикали (в отличие от Фиг.5В). Из-за использования формата цветности 4:2:2 каждая область цветности на Фиг.6С кажется горизонтально сжатой относительно соответствующей области яркости, изображенной на Фиг.6В. Значение 702 флага преобразования разделения, равное единице, делит 16×32 область 662, соответствующую единице 602 кодирования (CU), на четыре области размера 8×16, такие как 8×16 область 664. 8×16 область 664 имеет неквадратную форму, а также больше по размеру, чем другие неквадратные области, проиллюстрированные на Фиг.6С, такие как 4×8 область 670. Для каждой области размера 8×16 значение флага преобразования разделения, такое как значение 704 флага преобразования разделения, указывает, должна ли соответствующая область размера 8×16 быть далее подразделена на четыре меньших области размера 4×8, аналогично способу разделения дерева квадрантов, представленному в дереве 630 преобразований для массива отсчетов сигнала яркости. Верхняя правая 8×16 область 672 далее подразделяется на четыре области размером 4×8. Значение 804 флага кодированного блока, равное единице, указывает, что каждая из четырех областей размером 4×8 может содержать значимые остаточные коэффициенты. Флаг кодированного блока для каждой области размером 4×8 таким образом обязан указывать на присутствие преобразования для соответствующей области. Из этих четырех областей размером 4×8 нижняя левая область 674 с размером 4×8 (заштрихованная) содержит единицу преобразования (TU), но не содержит преобразования, и поэтому имеет значение 814 флага кодированного блока, равное нулю. Каждая из оставшихся областей с размером 4×8, такая как область 670, имеет преобразование и поэтому имеет соответствующее значение флага кодированного блока, равное единице. Верхняя левая область размера 8×16 подразделяется на две равных области размера 8×8. В отличие от подразделения дерева квадрантов никакого соответствующего флага преобразования разделения в закодированном потоке битов 312 не присутствует.

[00082] Разделение области канала, такого как канал цветности, единицы преобразования (TU) на несколько областей (каждая из которых может иметь преобразование) без сигнализации об этом в закодированном потоке битов 312 упоминается как «логически выведенное разделение». Логически выведенное разделение избавляет от необходимости вводить аппаратные средства, поддерживающие неквадратное преобразование для этого случая (8×16). Вместо этого используются такие преобразования, как первое преобразование 666 размера 8×8. Поскольку каждая из областей, получающихся в результате логически выведенного разделения, может содержать нулевую остаточную информацию, необходимо определить присутствие преобразования в каждой области, получающейся в результате логически выведенного разделения. Соответственно, отдельные значения флага кодированного блока требуются для каждой области, получающейся в результате логически выведенного разделения. В этом случае значения 806 и 808 флага кодированного блока соответствуют первому преобразованию 666 размера 8×8 и второму преобразованию 668 размера 8×8

соответственно. Для тех единиц преобразования (ТУ), в которых логически выведенное разделение не имеет места, значение флага кодированного блока для каждого канала цветности определяет присутствие или отсутствие преобразования для области, занятой единицей преобразования (ТУ) для канала цветности. Когда логически выведенное разделение имеет место, для каждой из получающихся областей требуется отдельное значение флага кодированного блока (не показанное на Фиг.8), однако реализации могут сохранять значение флага кодированного блока, приписываемое всей единице преобразования (ТУ). Отдельное значение флага кодированного блока может быть логически выведено как 1 во всех случаях, или отдельное значение флага кодированного блока может быть определено путем применения операции логического 'ИЛИ' к значению флага кодированного блока каждой области, получающейся в результате разделения. Если отдельное значение флага кодированного блока определяется из значения флага кодированного блока каждой области, получающейся в результате разделения, отдельное значение флага кодированного блока может быть закодировано в закодированном потоке битов 312 энтропийным кодером 324 и декодировано из закодированного потока битов 312 энтропийным декодером 420 в качестве дополнительного флага кодированного блока (не показанного на Фиг.9). В таком случае, когда отдельное значение флага кодированного блока является нулем, значение флага кодированного блока каждой области, получающейся в результате разделения, может быть логически выведено как ноль, а когда отдельное значение флага кодированного блока равно единице, флаги кодированного блока для каждой области, получающейся в результате разделения, кодируются в закодированном потоке битов 312 энтропийным кодером 324 и декодируются из закодированного потока битов 312 энтропийным декодером 420.

[00083] Нижняя левая 8×16 область 680 из области 662 размера 16×32 иллюстрирует логически выведенное разделение, в котором преобразование размера 8×8 присутствует в верхней логически выведенной области 682 размера 8×8 , но никакого преобразования размера 8×8 не присутствует в нижней логически выведенной области 684 размера 8×8 . Нижний правый массив 676 размера (заштрихованный) содержит единицу преобразования (ТУ), но не содержит преобразования в любой квадратной области размера 8×8 , полученной в результате логически выведенного разделения, и поэтому имеет значения 810, 812 флага кодированного блока, равные нулю.

[00084] Присутствие двух каналов цветности приводит к дублированию структуры, изображенной на Фиг.6С, с отдельными значениями флага кодированного блока, используемыми для определения присутствия преобразований для каждого канала цветности. В этой реализации разделение было логически выведено для размеров области для цветности, отличающихся от размера 4×8 , что приводит к использованию прямоугольного преобразования размера 4×8 , такого как 4×8 преобразование 816 (содержащееся в области 670), и к возможности повторного использования существующих квадратных преобразований в других случаях (например, 8×8 , 16×16). Таким образом, можно говорить о том, что существует набор предопределенных размеров области (таких как 8×16 и 16×32), для которых может использоваться разделение на две области и соответственно могут использоваться два преобразования (размеров 8×8 и 16×16). Другие определения предопределенного набора размеров области, для которых происходит логически выведенное разделение, также возможны, и позволят по-разному комбинировать существующие квадратные преобразования и прямоугольные преобразования, которые должны быть использованы. Для некоторых реализаций также возможно всегда логически вывести разделение, и в этом случае

никакого прямоугольного преобразования не предусматривается для цветовых каналов с цветностью 4:2:2. В таком случае predetermined набор размеров области, для которых происходит логически выведенное разделение, содержит все возможные размеры области цветности (например, 4×8, 8×16 и 16×32 для формата цветности 4:2:2 или 4×4, 8×8, 16×16 и 32×32 для формата цветности 4:4:4).

[00085] Когда используется формат цветности 4:2:0, логически выведенное разделение не имеет места ни для одной области цветности в единице преобразования (TU), поэтому максимальное количество преобразований для каждого канала цветности всегда равно единице (значение флага кодированного блока для каждого канала цветности определяет, происходит ли преобразование цветности).

[00086] Хотя видеокодер 114 и видеодекoder 134 описываются независимо от различий между каналами яркости и цветности, отличающиеся сетки отсчетов, следующие из форматов цветности, определяют необходимость различий в модулях. Практические реализации могут иметь отдельные «пути обработки» для канала яркости и для каналов цветности. Такая реализация может таким образом развязать обработку отсчетов сигнала яркости и отсчетов сигнала цветности. Поскольку закодированный поток битов 312 является единственным потоком битов и для канала яркости, и для каналов цветности, Энтропийный кодер 324 и энтропийный декодер 420 не разделяются. Дополнительно, единственный буфер кадров, такой как буфер 332, 432 кадра, содержит отсчеты сигнала яркости и отсчеты сигнала цветности, и таким образом не разделяется. Однако модули 322-330 и 334-340 и модули 422-430 и 434 могут иметь развязанную обработку яркости и цветности, включая реализации, имеющие отдельную логику для яркости и цветности, создавая таким образом «путь обработки яркости» и «путь обработки цветности».

[00087] Некоторые реализации могут логически выводить разделение для 16×32 области канала цветности единицы преобразования (TU) на две области размера 16×16, но не выводить разделение для случаев 8×16 и 4×8. Такие реализации избегают необходимости введения логики преобразования с 32 точками в путь обработки цветности, полагаясь вместо этого на логику преобразования с 4, 8 или 16 точками, хорошо известную в данной области техники.

[00088] Фиг. 9А и Фиг. 9В иллюстрируют синтаксическую структуру, которая может использоваться для того, чтобы закодировать или иным образом представить иерархический уровень дерева преобразований. В неконцевых узлах дерева преобразований синтаксическая структура 900 расширяется рекурсивно в соответствии со структурами данных, такими как структуры 700 и 800 данных, для того, чтобы определить синтаксические элементы, присутствующие в части закодированного потока битов 312, соответствующей дереву преобразований. В концевых узлах дерева преобразований (где никакого дальнейшего подразделения в дереве преобразований не происходит) синтаксическая структура 930 определяет синтаксические элементы, присутствующие в части закодированного потока битов 312. Как правило, присутствуют одна структура данных для яркости и две структуры данных для цветности, хотя возможны дополнительные структуры данных, такие как для кодирования альфа-канала или карты глубины. Альтернативно может быть использовано меньшее количество структур данных, как например в случае, где единственная структура данных совместно используется каналами цветности, и значения флага кодированного блока могут быть совместно использованы каналами цветности. Синтаксическая структура 902 неконцевого узла дерева преобразований определяет кодирование одного иерархического уровня дерева преобразований, такого как дерево 630 преобразований.

Флаг 910 преобразования разделения кодирует значение флага преобразования разделения как 1, так же как значение 702 флага преобразования разделения. Это значение указывает, что синтаксическая структура 902 неконцевого узла дерева преобразований включает в себя нижний иерархический уровень, который содержит
5 дополнительные экземпляры синтаксической структуры 902 неконцевого узла дерева преобразований, или синтаксическую структуру 932 конечного узла дерева преобразований, или «дочерние узлы». Флаг 912 кодированного блока кодирует значение 802 флага кодированного блока, равное единице, для канала цветности «U», и флаг 914 кодированного блока кодирует дополнительное значение флага
10 кодированного блока для канала цветности «V». Если синтаксическая структура 902 неконцевого узла дерева преобразований определяет верхний уровень иерархии дерева преобразований, тогда присутствуют флаги 912, 914 кодированного блока. Если синтаксическая структура 902 неконцевого узла дерева преобразований не определяет верхний уровень иерархии дерева преобразований, тогда присутствуют только флаги
15 912, 914 кодированного блока, если соответствующие флаги кодированного блока на родительском уровне иерархии дерева преобразований присутствуют и равны единице. Если в дереве 630 преобразований существует более низкий иерархический уровень (относительно верхнего иерархического уровня), производится подразделение дерева квадрантов. Это подразделение приводит к четырем синтаксическим структурам 916,
20 918, 920, 922 дерева преобразований, включаемым в синтаксическую структуру 902 неконцевого узла дерева преобразований.

[00089] Синтаксическая структура 930 определяет кодирование конечного узла 932 дерева преобразований (то есть такого узла, в котором никаких дальнейших подразделений не производится). Флаг 940 преобразования разделения кодирует
25 значение флага преобразования разделения как 0, так же как значение 704 флага преобразования разделения.

[00090] Флаг преобразования разделения кодируется только в том случае, если соответствующая область больше, чем минимальный размер. Например, область 636
30 имеет наименьший допустимый размер для области 4×4 отсчетов сигнала яркости (соответствующий наименьшему поддерживаемому размеру преобразования яркости), так что значение 714 флага преобразования разделения логически выводится как нулевое, и никакой флаг преобразования разделения не кодируется для соответствующей синтаксической структуры дерева преобразований.

[00091] Для области 636 остаточные отсчеты сигнала цветности преобразуются с
35 использованием преобразования цветности размера 4×8, следовательно логически выведенное преобразование разделения отсутствует. Флаги кодированного блока, такие как флаг 942 кодированного блока и флаг 946 кодированного блока, могут присутствовать для того, чтобы сигнализировать о присутствии преобразования для каждого из каналов цветности. Флаг 950 кодированного блока сигнализирует о
40 присутствии преобразования для канала яркости. Остаточные коэффициенты для каналов яркости и цветности (если они есть) присутствуют в единице преобразования (TU) синтаксической структуры 952. Если значение флага 950 кодированного блока равно единице, остаточный блок 954 яркости присутствует в закодированном потоке битов 312. Если значение флага кодированного блока для каждого канала цветности
45 равно единице, соответствующие остаточные блоки 956 и 960 цветности присутствуют в закодированном потоке битов 312.

[00092] Для области 664 остаточные отсчеты сигнала цветности преобразуются с использованием двух преобразований цветности размера 8×8, следовательно логически

выведенное преобразование разделения присутствует. Флаги 942 и 946 кодированного блока, если они присутствуют, сигнализируют о присутствии преобразований размера 8×8 для каждого канала цветности первого преобразования 666 размера 8×8. Флаг 944 кодированного блока и флаг 948 кодированного блока, если они есть, сигнализируют о присутствии преобразований размера 8×8 для каждого канала цветности второго преобразования 668 размера 8×8. Если значение флага 944 кодированного блока равно единице, остаточный блок 958 цветности присутствует в закодированном потоке битов 312. Если значение флага 948 кодированного блока равно единице, остаточный блок 962 цветности присутствует в закодированном потоке битов 312.

[00093] Синтаксическая структура 930, проиллюстрированная на Фиг.9В, показывает первое и второе преобразование каждого канала цветности, закодированные рядом для логически выведенного преобразования разделения. Альтернативно могут использоваться другие компоновки, такие как кодирование синтаксических элементов для каждого канала цветности рядом, или кодирование синтаксических элементов для каждого канала цветности, перемежаемое другими синтаксическими элементами.

[00094] Фиг. 9С и Фиг. 9D иллюстрируют синтаксическую структуру 9100, которая может использоваться для того, чтобы закодировать или иным образом представить иерархический уровень дерева преобразований. В неконцевых узлах дерева преобразований синтаксическая структура 9100 расширяется рекурсивно в соответствии со структурами данных, такими как структуры 700 и 800 данных, для того, чтобы определить синтаксические элементы, присутствующие в части закодированного потока битов 312, соответствующей дереву преобразований. Экземпляр альтернативной синтаксической структуры 9100 существует для каждого узла в дереве преобразований, в том числе листовые узлы, каждый из которых содержит единицу преобразования (TU). Когда происходит «логически выведенное разделение» для того, чтобы подразделить единицу преобразования (TU) для каждого канала цветности, синтаксическая структура 9130 определяет синтаксические элементы, существующие в части закодированного потока битов 312. Как правило, присутствуют одна структура данных для яркости и две структуры данных для цветности, хотя возможны дополнительные структуры данных, такие как для кодирования альфа-канала или карты глубины. Альтернативно может быть использовано меньшее количество структур данных, как например в случае, где единственная структура данных совместно используется каналами цветности, и значения флага кодированного блока могут быть совместно использованы каналами цветности. Синтаксическая структура 9102 определяет кодирование одного иерархического уровня дерева преобразований, такого как дерево 630 преобразований.

[00095] Для экземпляра синтаксической структуры 9102 дерева преобразований в неконцевом узле дерева преобразований, такого как дерево 630 преобразований, флаг 9110 преобразования разделения кодирует единичное значение флага преобразования разделения, такое как значение 702 флага преобразования разделения. Это значение указывает, что экземпляр синтаксической структуры 9102 дерева преобразований включает в себя нижний иерархический уровень, содержащий дополнительные экземпляры синтаксической структуры 9102 дерева преобразований или «дочерние узлы». Флаг 9112 кодированного блока содержит значение флага кодированного блока в соответствии с описанием флага 912 кодированного блока. Флаг 9114 кодированного блока содержит значение флага кодированного блока в соответствии с описанием флага 914 кодированного блока. Если в дереве 630 преобразований существует более низкий иерархический уровень (относительно верхнего иерархического уровня),

производится подразделение дерева квадрантов. Это подразделение приводит к четырем синтаксическим структурам 9116, 9118, 9120, 9122 дерева преобразований, включаемым в синтаксическую структуру 9102 неконцевого узла дерева преобразований. Каждая из синтаксических структур 9116, 9118, 9120, 9122 дерева преобразований является

5 другим экземпляром синтаксической структуры 9102 дерева преобразований. Флаг 9124 кодированного блока и часть 9126 единицы преобразования яркости будут отсутствовать в синтаксической структуре 9102 дерева преобразований.

[00096] Реализации могут также располагать синтаксическую структуру 9102 дерева преобразований так, чтобы флаг 9124 кодированного блока и часть 9126 единицы преобразования яркости (если они присутствуют) помещались ранее в синтаксической

10 структуре 9102 дерева преобразований, например между флагом 9114 кодированного блока и синтаксической структурой 9116 дерева преобразований.

[00097] Для экземпляра синтаксической структуры 9102 дерева преобразований в неконцевом узле дерева преобразований, такого как дерево 630 преобразований, флаг 9110 преобразования разделения кодирует нулевое значение флага преобразования

15 разделения, такое как значение 704 флага преобразования разделения. Экземпляр синтаксической структуры 9102 дерева преобразований таким образом соответствует единице преобразования (TU) в дереве 930 преобразований. Единица преобразования (TU) имеет размер, определяемый в соответствии с единицей кодирования (CU),

20 содержащей единицу преобразования (CU), такую как единица 602 кодирования (CU), и глубиной преобразования. Флаг 9112 кодированного блока кодирует единичное значение флага кодированного блока для того, чтобы указать, что любая из областей цветности, получающихся в результате логически выведенного разделения для канала цветности «U», может иметь значение флага кодированного блока, равное единице.

25 Если флаг 9112 кодированного блока имеет нулевое значение, то значение флага кодированного блока для каждой области цветности, получающейся в результате логически выведенного разделения для канала цветности «U», имеет значение флага кодированного блока, логически выведенное как ноль. Даже когда флаг 9112 кодированного блока имеет значение, равное единице, реализации могут кодировать

30 флаг кодированного блока, имеющий нулевое значение для каждой области цветности, получающейся в результате логически выведенного разделения. Следовательно, реализации могут опускать флаг 9112 кодированного блока в закодированном потоке битов 312, а вместо этого всегда логически выводить значение флага кодированного блока, равное единице, для опущенного флага 9112 кодированного блока. Флаг 9114 кодированного блока кодирует дополнительное значение флага кодированного блока

35 для канала цветности «V» аналогично флагу 9112 кодированного блока. Для единицы преобразования (TU) размеры, которые согласуются с размерами, для которых происходит логически выведенное разделение на четыре области цветности (максимальное количество массивов остаточных коэффициентов сигнала цветности равно четырем), четыре синтаксические структуры 9116, 9118, 9120, 9122 дерева преобразований включаются в синтаксическую структуру 9102 узла дерева преобразований. Для единицы преобразования (TU) размеры, которые согласуются с теми, для которых происходит логически выведенное разделение на две области

40 цветности (максимальное количество массивов остаточных коэффициентов сигнала цветности равно двум), две синтаксические структуры дерева преобразований, такие как синтаксические структуры 9116, 9118 дерева преобразований, включаются в синтаксическую структуру 9102 узла дерева преобразований. Каждая из синтаксических структур 9116, 9118, 9120, 9122 дерева преобразований является экземпляром дерева

преобразований для синтаксической структуры 9132 цветности. Флаг 9124 кодированного блока кодирует значение флага кодированного блока, такое как значение 708 флага кодированного блока, определяя присутствие или отсутствие преобразования для канала яркости единицы преобразования (TU). Часть яркости блока 9126 преобразования кодирует массив остаточных коэффициентов сигнала яркости как синтаксические элементы 9128 остаточной яркости.

[00098] Дерево преобразований для синтаксической структуры 9132 цветности, существующей только для каждой области цветности, когда имеет место логически выведенное разделение, включает в себя сокращенный набор синтаксиса синтаксической структуры 930 дерева преобразований. Флаг 9142 кодированного блока содержит значение флага кодированного блока для канала цветности «U» области цветности. Флаг 9144 кодированного блока содержит значение флага кодированного блока для канала цветности «V» области цветности. Часть цветности единицы преобразования (TU) 9146 кодирует подмножество единицы преобразования (TU) синтаксической структуры 952. Часть цветности единицы преобразования (TU) 9146 кодирует массив остаточных коэффициентов сигнала цветности как синтаксические элементы 9150 остаточной цветности для канала цветности «U», если значение флага 9142 кодированного блока равно единице. Часть цветности единицы преобразования (TU) 9146 кодирует массив остаточных коэффициентов сигнала цветности как синтаксические элементы 9152 остаточной цветности для канала цветности «V», если значение флага 9144 кодированного блока равно единице.

[00099] Синтаксическая структура 9130, изображенная на Фиг.9D, показывает первый и второй флаг кодированного блока, закодированные рядом, сопровождаемые первым и вторым массивом остаточных коэффициентов сигнала цветности каждого канала цветности для логически выведенного преобразования разделения. Альтернативно могут использоваться другие компоновки, такие как кодирование флага кодированного блока и массива остаточных коэффициентов сигнала цветности рядом для каждого канала цветности.

[000100] Хотя логически выведенное преобразование разделения иллюстрируется областью 664 размера 8×16 , разделяемой на две области размера 8×8 , альтернативные реализации могут выполнять разделение для других областей. Например, некоторые реализации могут логически выводить разделение области размером 16×32 на две области размером 16×16 . Такие реализации выгодно избегают необходимости в одномерном преобразовании с 32 точками в пути обработки цветности. Поскольку одномерное преобразование с 32 точками не требуется для пути обработки цветности, когда применяется формат цветности 4:2:0, требование одномерного преобразования с 32 точками полностью удаляется из пути обработки цветности. Реализации, которые используют отдельные схемные решения обработки для того, чтобы развязать каналы яркости и цветности, могут таким образом достичь более низкой стоимости реализации в схеме обработки цветности.

[000101] Существует формат цветности 4:4:4, где имеется одно положение отсчета цветности для каждого положения отсчета сигнала яркости. Соответственно, при использовании этого формата преобразования для канала цветности и канала яркости могут иметь одинаковые размеры. При самом большом размере преобразования 32×32 в пути обработки яркости это потребовало бы введения преобразования размером 32×32 в путь обработки цветности для развязанной реализации. Конкретные реализации могут логически выводить разделение для каждого канала цветности, чтобы разделить область размера 32×32 на четыре области размера 16×16 , позволяя повторно

использовать существующее преобразование размером 16×16 в пути обработки цветности. Поскольку в пути обработки цветности для формата цветности 4:4:4 использовалось бы только преобразование размера 32×32 , логическое выведение разделения для каждого канала цветности для разделения области размера 32×32 на четыре области размера 16×16 позволило бы удалить преобразование размером 32×32 из пути обработки цветности, уменьшая требуемое схемное решение для обработки. Такие реализации потребовали бы четырех значений флага кодированного блока для каждого канала цветности, и таким образом до четырех флагов кодированного блока, закодированных в синтаксической структуре 930 для каждого канала цветности в закодированном потоке битов 312.

[000102] Реализации, поддерживающие формат цветности 4:2:2, могут также логически выводить разделение для каждого канала цветности для того, чтобы разделить область размера 32×16 на четыре области размера 8×16 . Такие реализации требуют четырех значений флага кодированного блока для каждого канала цветности, и таким образом четырех флагов кодированного блока, закодированных в синтаксической структуре 930, для каждого канала цветности в закодированном потоке битов 312, таким образом флаг кодированного блока «CU3», «CU4», «CV3» и «CV4» (не показанный на Фиг.9В), может быть введен в синтаксическую структуру 952 единицы преобразования (TU). Такие реализации позволяют избежать введения логики преобразования с 32 точками в путь обработки цветности и, когда области размером 8×16 больше не подразделяются, могут повторно использовать логику преобразования 8×16 , требуемую для единиц преобразования (TU) размера 16×16 (в канале яркости), которые требуют преобразования размера 8×16 для каналов цветности.

[000103] Фиг.10 представляет собой блок-схему последовательности операций, показывающую способ 1000 для кодирования единицы преобразования (TU) путем кодирования синтаксической структуры 902 неконцевого узла дерева преобразований и синтаксической структуры 932 конечного узла дерева преобразований. Способ 1000 описывается в отношении канала цветности единицы преобразования (TU), однако способ 1000 может быть применен к любому каналу цветности единицы преобразования (TU). Поскольку синтаксическая структура 902 неконцевого узла дерева преобразований и синтаксическая структура 932 конечного узла дерева преобразований описывают один узел в дереве преобразований, способ 1000 кодирует один узел дерева преобразований в закодированный поток битов 312. Способ 1000 может быть реализован в аппаратных средствах или программным обеспечением, выполняемым, например, на процессоре 205. Способ 1000 первоначально вызывается для верхнего уровня дерева преобразований и может вызывать сам себя (рекурсивно) для того, чтобы закодировать дочерние узлы дерева преобразований. Этап 1002 определения размера единицы преобразования определяет размер единицы преобразования (TU) в дереве преобразований в соответствии с размером единицы кодирования (CU), которая содержит дерево преобразований, и значением глубины преобразования единицы преобразования (TU). Когда способ 1000 вызывается на верхнем уровне дерева преобразований, значение глубины преобразования обнуляется, в противном случае значение глубины преобразования обеспечивается родительским экземпляром способа 1000. Значение флага преобразования разделения, такое как значение 702 флага преобразования разделения, кодируется в закодированном потоке битов 312 как флаг 910 преобразования разделения, если значение глубины преобразования меньше, чем максимально допустимая глубина преобразования.

[000104] Когда значение флага преобразования разделения равно единице, флаги

912 и 914 кодированного блока цветности кодируются для каждого канала цветности только в том случае, если у родительского узла иерархии дерева преобразований соответствующее значение флага кодированного блока равно единице. Способ 1000 затем вызывает новый экземпляр способа 1000 для каждого дочернего узла дерева преобразований (представленного в части закодированного потока битов 312 синтаксическими структурами 916, 918, 920 и 922 дерева преобразований). Каждому экземпляру способа 1000, вызванному для дочерних узлов, предоставляется значение глубины преобразования, равное значению глубины преобразования текущего экземпляра способа 1000, увеличенному на единицу.

10 [000105] Когда значение флага преобразования разделения является нулем, этап 1004 идентификации максимального количества прямых преобразований определяет максимальное количество (n) преобразований для каждого канала цветности закодированной области. Когда логически выводимое разделение не производится, это число n будет иметь значение 1. Когда используется формат цветности 4:2:2, и
15 встречается прямоугольная область канала цветности, такая как область 664 размера 8×16, и размер области является одним из predetermined набора размеров области (например, 16×32 и 8×16), производится логически выводимое разделение, и максимальное количество преобразований будет равно двум (в противном случае количество преобразований будет равно одному). В противном случае (размер области
20 не является одним из predetermined набора размеров области) максимальное количество преобразований будет равно одному. Например, если 4×8 не является одним из predetermined набора размеров области, то максимальное количество преобразований будет равно одному. Когда используется формат цветности 4:4:4, и размер области является одним из predetermined набора размеров области
25 (например 32×32), производится логически выведенное разделение, и максимальное количество преобразований будет равно четырем. В противном случае (размер области не является одним из predetermined набора размеров области) максимальное количество преобразований будет равно одному. Например, если 8×8 не является одним из predetermined набора размеров области, то максимальное количество преобразований будет равно одному. Хотя predetermined набор размеров области
30 включает 8×16, возможны другие predetermined наборы размеров области, такие как только 16×32, когда используется формат цветности 4:2:2, или 32×32, когда используется формат цветности 4:4:4.

[000106] Для каждого канала цветности, если у родительского узла значение флага кодированного блока равнялось единице, то для каждого из n флаг кодированного
35 блока кодируется в закодированном потоке битов 312. Например, когда количество преобразований равно двум, флаги 942 и 944 кодированного блока указывают присутствие преобразования для каждой из этих двух областей, полученных путем логически выведенного разделения. Этап 1006 выбора прямого преобразования
40 выбирает прямое преобразование из predetermined набора прямых преобразований для каждого из максимального количества преобразований, основываясь на размере единицы преобразования (TU), который в свою очередь зависит от глубины преобразования, и таким образом обращается к иерархическому уровню единицы преобразования в наибольшей единице кодирования. Когда глубина преобразования
45 равна нулю, размер единицы преобразования (TU) равен размеру единицы кодирования (CU). Для каждого приращения глубины преобразования размер единицы преобразования (TU) делится на два. Для размера единицы кодирования (CU), равного 32×32, глубина преобразования равна нулю и при использовании формата цветности

4:2:2 размер единицы преобразования (TU) таким образом будет 32×32, а размер преобразования для цветности таким образом будет 16×32. Например, когда максимальное количество преобразований равно двум, и размер области для цветности составляет 16×32, тогда для каждой из областей размером 16×16 для цветности, полученных в результате логически выведенного разделения, выбирается прямое преобразование 16×16.

[0100] Этап 1008 применения прямого преобразования выполняет прямое преобразование для каждого из максимального количества преобразований над соответствующей областью, у которой значение флага кодированного блока равно единице. Этап 1008 кодирования массивов остаточных отсчетов сигнала цветности обычно выполняется модулем 320 преобразования. Это приводит к преобразованию каждого массива остаточных отсчетов сигнала цветности (представление в пространственной области) в массив остаточных коэффициентов сигнала цветности (представление в частотной области).

[0101] Этап 1010 кодирования массивов остаточных коэффициентов сигнала цветности кодирует массив остаточных коэффициентов сигнала цветности для каждой из максимального количества областей преобразования каждого канала цветности, имеющей значение флага кодированного блока, равное единице, в закодированный поток битов 312. Количество массивов остаточных коэффициентов сигнала цветности, кодируемых для данной единицы преобразования для данного канала цветности, зависит от значения флага кодированного блока каждого преобразования, и таким образом будет изменяться от нуля до (самое большее) максимального количества преобразований. Например, когда количество преобразований равно двум и оба канала цветности имеют значения флага кодированного блока, равные единице, для каждого из значений счетчика, тогда остаточные блоки 956, 958, 960 и 962 цветности кодируются в закодированном потоке битов 312. Если значение флага кодированного блока для каждого преобразования для данного канала цветности равно нулю, то никаких остаточных блоков цветности не кодируется в закодированном потоке битов 312 для этого канала цветности. Этап 1010 кодирования массивов остаточных коэффициентов сигнала цветности обычно выполняется энтропийным кодером 324.

[0102] Фиг.11 представляет собой блок-схему последовательности операций, показывающую способ 1100 для декодирования единицы преобразования (TU) путем декодирования синтаксической структуры 902 неконцевого узла дерева преобразований и синтаксической структуры 932 конечного узла дерева преобразований. Способ 1100 описывается в отношении канала цветности единицы преобразования (TU), однако способ 1100 может быть применен к любому каналу цветности единицы преобразования (TU). Поскольку синтаксическая структура 902 неконцевого узла дерева преобразований и синтаксическая структура 932 конечного узла дерева преобразований описывают один узел в дереве преобразований, способ 1100 декодирует один узел дерева преобразований из закодированного потока битов 312. Способ 1100 может быть выполнен в подходящих аппаратных средствах или альтернативно в программном обеспечении, например исполняемом процессором 205. Способ 1100 первоначально вызывается для верхнего уровня дерева преобразований и может вызывать сам себя (рекурсивно) для того, чтобы декодировать дочерние узлы дерева преобразований. Этап 1102 определения размера единицы преобразования (TU) определяет размер единицы преобразования (TU) способом, идентичным этапу 1002 определения размера единицы преобразования. Этап 1102 определения размера единицы преобразования определяет размер единицы преобразования (TU) в дереве преобразований в

соответствии с размером единицы кодирования (CU), который содержит дерево преобразований, и значением глубины преобразования единицы преобразования (TU). Когда способ 1100 вызывается на верхнем уровне дерева преобразований, значение глубины преобразования обнуляется, в противном случае значение глубины преобразования обеспечивается родительским экземпляром способа 1100. Значение флага преобразования разделения, такое как значение 702 флага преобразования разделения, декодируется из закодированного потока битов 312 как флаг 910 преобразования разделения, если значение глубины преобразования меньше, чем максимально допустимая глубина преобразования.

[0103] Когда значение флага преобразования разделения равно единице, флаги 912 и 914 закодированного блока цветности декодируются для каждого канала цветности только в том случае, если у родительского узла иерархии дерева преобразований соответствующее значение флага закодированного блока равно единице. Способ 1100 затем вызывает новый экземпляр способа 1100 для каждого дочернего узла дерева преобразований (представленного в части закодированного потока битов 312 синтаксическими структурами 916, 918, 920 и 922 дерева преобразований). Каждому экземпляру способа 1100, вызванному для дочерних узлов, предоставляется значение глубины преобразования, равное значению глубины преобразования текущего экземпляра способа 1100, увеличенному на единицу.

[0104] Когда значение флага преобразования разделения является нулем, этап идентификации максимального количества обратных преобразований определяет (максимальное) количество (n) преобразований для каждого из по меньшей мере одного массива остаточных коэффициентов сигнала цветности, присутствующих в каждом канале цветности декодируемой области, способом, идентичным этапу 1004 идентификации максимального количества (n) прямых преобразований. Когда логически выводимое разделение не производится, это число n будет иметь значение 1. Когда используется формат цветности 4:2:2, и встречается прямоугольная область канала цветности, такая как область 664 размера 8×16, и размер области является одним из predetermined набора размеров области (например, 16×32 и 8×16), производится логически выводимое разделение, и максимальное количество преобразований будет равно двум (в противном случае количество преобразований будет равно одному). В противном случае (размер области не является одним из predetermined набора размеров области) максимальное количество преобразований будет равно одному. Например, если 4×8 не является одним из predetermined набора размеров области, то максимальное количество преобразований будет равно одному. Когда используется формат цветности 4:4:4, и размер области является одним из predetermined набора размеров области (например 32×32), производится логически выведенное разделение, и максимальное количество преобразований будет равно четырем. В противном случае (размер области не является одним из predetermined набора размеров области) максимальное количество преобразований будет равно одному. Например, если 8×8 не является одним из predetermined набора размеров области, то максимальное количество преобразований будет равно одному. Хотя predetermined набор размеров области включает 8×16, возможны другие predetermined наборы размеров области, такие как только 16×32, когда используется формат цветности 4:2:2, или 32×32, когда используется формат цветности 4:4:4. Для каждого канала цветности, если у родительского узла значение флага закодированного блока равно единице, то для каждого из (n) преобразований флаг закодированного блока декодируется из закодированного потока битов 312. Например, когда максимальное количество преобразований равно

двум, флаги 942 и 944 кодированного блока указывают на присутствие преобразования для каждой из этих двух областей, полученных путем логически выведенного разделения.

[0105] Этап 1106 декодирования массивов остаточных коэффициентов сигнала цветности затем декодирует массив остаточных коэффициентов для каждой из
5 максимального количества областей преобразований каждого канала цветности из закодированного потока битов 312, имеющей значение флага кодированного блока, равное единице. Количество массивов остаточных коэффициентов, декодируемых для данной единицы преобразования для данного канала цветности, зависит от значения флага кодированного блока каждого преобразования, и таким образом будет изменяться
10 от нуля до (самое большое) «количества (n) преобразований». Например, когда количество преобразований равно двум и оба канала цветности имеют флаги кодированного блока, равные единице, для каждого из значений счетчика, тогда остаточные блоки 956, 958, 960 и 962 цветности декодируются из закодированного потока битов 312. Этап 1106 декодирования массивов остаточных коэффициентов
15 сигнала цветности обычно выполняется энтропийным декодером 420 для каждого массива остаточных коэффициентов сигнала цветности, имеющего значение флага кодированного блока, равное единице.

[0106] Этап 1108 выбора обратного преобразования затем выбирает обратное преобразование из predetermined набора обратных преобразований для каждого
20 из максимального количества преобразований, имеющих значение флага кодированного блока, равное единице, для каждого канала цветности. Например, когда максимальное количество преобразований равно двум и размер области составляет 16×32 , и значение флага кодированного блока для каждого из этих двух преобразований равно единице, тогда обратное преобразование размера 16×16 выбирается для каждой из областей
25 размера 16×16 , полученных в результате логически выведенного разделения.

[0107] Этап 1110 применения обратного преобразования затем выполняет обратное преобразование для каждой из максимального количества областей преобразований над соответствующей областью, имеющей значение флага кодированного блока, равное
30 единице. Это приводит к преобразованию каждого массива остаточных коэффициентов сигнала цветности (представление в частотной области) в массив остаточных отсчетов сигнала цветности (представление в пространственной области), представляющих декодированный видеокادر. Этап 1110 применения обратного преобразования обычно выполняется модулем 422 обратного масштабирования и преобразования.

[0108] Фиг.12А показывает схему 1201 диагонального сканирования, Фиг.12В
35 показывает схему 1202 горизонтального сканирования, и Фиг.12С показывает схему 1203 вертикального сканирования, каждая для единицы 1200 преобразования размера 4×8 . Реализации, которые сканируют единицу 1200 преобразования размером 4×8 с использованием проиллюстрированных схем сканирования, обладают тем свойством, что остаточные коэффициенты группируются в блоки 4×4 , известные как «подблоки». Флаг «группа коэффициентов», присутствующий в закодированном потоке битов 312,
40 может поэтому использоваться для того, чтобы указать для каждого подблока присутствие по меньшей мере одного значимого (ненулевого) остаточного коэффициента. Применение размера подблока 4×4 для преобразования 4×8 достигает непротиворечивости со схемой сканирования, существующей в других размерах
45 преобразования, где коэффициенты всегда группируются в подблоки.

[0109] Конкретные реализации могут применять флаг группы коэффициентов для того, чтобы сигнализировать о присутствии по меньшей мере одного ненулевого остаточного коэффициента в каждом подблоке. Эти схемы сканирования выгодно

позволяют повторно использовать управляющее программное обеспечение или цифровую схему, которая обрабатывает остаточные коэффициенты, путем повторного использования обработки подблока для всех размеров преобразований. Конкретная используемая схема сканирования может быть выбрана согласно таким критериям, как направление внутреннего предсказания соотнесенной единицы предсказания (PU). Когда преобразование кодирует отсчеты сигнала цветности на сетке отсчетов формата цветности 4:2:2, отношение между направлением внутреннего предсказания и схемой сканирования изменяется, потому что каждый отсчет сигнала цветности отображается на неквадратный (2×1) массив отсчетов сигнала яркости, влияя на «направление» или угол режима внутреннего предсказания. Сканирование показано в «обратном» направлении, заканчивающемся на коэффициенте DC, расположенном в верхнем левом углу единицы преобразования (TU). Далее, сканирование не обязано начинаться в нижнем правом углу единицы преобразования (TU). Из-за преобладания ненулевых остаточных коэффициентов в верхней левой области единицы преобразования (TU) сканирование может начаться с «положения последнего значимого коэффициента» и прогрессировать в обратном направлении до тех пор, пока не будет достигнут верхний левый коэффициент.

[0110] Другие реализации могут применять единственное сканирование к данной области для того, чтобы закодировать остаточные коэффициенты, а затем применять более одного преобразования к этим остаточным коэффициентам. В этом случае только один флаг кодированного блока используется для области, и поэтому она для всех преобразований покрывается схемой сканирования. Флаг кодированного блока устанавливается в единицу, если по меньшей мере один значимый остаточный коэффициент существует в каком-либо из сканирований. Например, схемы сканирования 4×8, изображенные на Фиг. 12А-12С, могут быть применены для того, чтобы закодировать остаточные коэффициенты двух преобразований 4×4. Два массива 4×4 остаточных коэффициентов могут быть объединены для того, чтобы сформировать массив 4×8, подходящий для схемы сканирования. Поскольку над массивом выполняется единственное сканирование, одно положение «последнего значимого коэффициента» кодируется в потоке битов для схемы сканирования, и для массива достаточно единственного значения флага кодированного блока. Свойство энергетического уплотнения модифицированного дискретного косинусного преобразования (DCT) дает преимущество перед другими схемами, такое как чередование коэффициентов каждого квадратного преобразования вдоль пути схемы сканирования в прямоугольный массив коэффициентов. Это дает то преимущество, что плотность значений остаточных коэффициентов в каждом массиве 4×4 остаточных коэффициентов приблизительно выравнивается в объединенном массиве 4×8, обеспечивая более высокую эффективность сжатия энтропийным кодером 324 для последующего декодирования энтропийным декодером 420.

[0111] Некоторые реализации, кодирующие каналы цветности, могут использовать первое преобразование для того, чтобы закодировать остаточные отсчеты в положениях отсчетов сигнала цветности, соответствующих сетке отсчетов сигнала цветности 4:2:0, и второе преобразование для того, чтобы закодировать остаточные отсчеты в дополнительных положениях отсчетов сигнала цветности, введенных в сетку отсчетов сигнала цветности 4:2:2 относительно сетки отсчетов сигнала цветности 4:2:0. Такие реализации могут выгодно использовать упрощенное преобразование для второго преобразования, такое как преобразование Адамара, с добавлением (или с иным объединением) выхода второго преобразования к остаточным отсчетам для первого

преобразования с тем, чтобы сформировать остаточные отсчеты для второго преобразования. Этап предварительной обработки, реализующий такое преобразование, как преобразование Хаара, может выгодно использоваться для дискретизации сетки отсчетов сигнала цветности для формата цветности 4:2:2 в сетку отсчетов сигнала цветности для формата цветности 4:2:0. Такие конфигурации должны передавать дополнительные остаточные коэффициенты от этапа предварительной обработки как побочную информацию, с применением такого остатка к каждой наибольшей единицы кодирования (LCU) в случае, когда преобразование предварительной обработки применяется на уровне наибольшей единицы кодирования (LCU).

[0112] Реализации, имеющие многократные преобразования для данной области, могут использовать либо единственное объединенное сканирование, покрывающее всю область, либо отдельное сканирование для каждого преобразования. Если сканирование для нескольких преобразований объединяется в одно сканирование, то только один флаг кодированного блока требуется для каждой сканируемой области. Те реализации, которые используют одно объединенное сканирование, могут достичь более высокого сжатия остаточных коэффициентов путем чередования остаточных коэффициентов каждого преобразования, такого как чередование на основе «коэффициент за коэффициентом», для того, чтобы объединить рядом остаточные коэффициенты от каждого преобразования, имеющие сходные спектральные свойства.

[0113] Приложение А иллюстрирует возможный «текст» для разрабатываемого стандарта высокоэффективного видеокodирования (HEVC), который соответствует синтаксической структуре 900 и синтаксической структуре 930. Каждый экземпляр функции `transform_tree()` в приложении А изображается как часть синтаксической структуры, помеченной как «ТТ» на Фиг. 9А и Фиг. 9С, и каждый экземпляр функции `transform_unit()` в приложении А изображается как часть синтаксической структуры, помеченной как «ТУ» на Фиг. 9А и Фиг. 9В. Текст, приведенный в Приложении А, является одним примером текста, который согласуется с синтаксическими структурами 900 и 930, и возможны другие примеры. Текст, который согласуется с синтаксическими структурами 900 и 930, подразумевает, что видеокodер 114 выполняет способ 1000 для того, чтобы закодировать поток битов, а видеокodер 134 выполняет способ 1100 для того, чтобы декодировать поток битов.

[0114] Приложение В иллюстрирует возможный текст для разрабатываемого стандарта высокоэффективного видеокodирования (HEVC), который соответствует синтаксической структуре 9100 и синтаксической структуре 9130. Каждый экземпляр функции `transform_tree()` в приложении В изображается как часть синтаксической структуры, помеченной как «ТТ» на Фиг. 9С и Фиг. 9D, и каждый экземпляр функции `transform_unit()` в приложении А изображается как часть синтаксической структуры, помеченной как «ТУ» на Фиг. 9С и Фиг. 9D. Текст, приведенный в Приложении В, является одним примером текста, который согласуется с синтаксическими структурами 9100 и 9130, и возможны другие примеры. Текст, который согласуется с синтаксическими структурами 9100 и 9130, также подразумевает, что видеокodер 114 выполняет способ 1000 для того, чтобы закодировать поток битов, а видеокodер 134 выполняет способ 1100 для того, чтобы декодировать поток битов.

[0115] Текст в Приложении А и в Приложении В приводит к реализации, посредством которой область цветности размера 32×32, которая находится в единице преобразования (TU) размера 32×32, сконфигурированном для формата цветности 4:4:4, приводит к (максимум) четырем применяемым преобразованиям цветности размера 16×16, и область цветности размера 16×32, которая находится в единице преобразования (TU) размера

32×32, сконфигурированном для формата цветности 4:2:2, приводит к (максимум) двум применяемым преобразованиям цветности размера 16×16. Реализация, следующая из текста, приведенного в Приложении А и в Приложении В, при применении к единицам преобразования (TU) меньшего размера для формата цветности 4:2:2, приводит
5 (максимум) к одному преобразованию цветности. Например, преобразование размером 8×16 применяется к области цветности размера 8×16 и преобразование размером 4×8 применяется к области цветности размера 4×8.

ПРОМЫШЛЕННАЯ ПРИМЕНИМОСТЬ

[0116] Описанные компоновки применимы к компьютеру и отраслям обработки
10 данных и в частности для цифровой обработки сигнала для кодирования и декодирования сигналов, таких как видеосигналы.

[0117] Вышеизложенное описывает лишь некоторые варианты осуществления настоящего изобретения, являющиеся иллюстративными, а не ограничивающими, и модификации и/или изменения в нем могут быть сделаны без отступления от области
15 охвата и духа настоящего изобретения.

[0118] В контексте данного описания словосочетание «включающий в себя» означает «включающий преимущественно, но не обязательно исключительно» или «имеющий» или «включающий в себя», а не «состоящий только из». Вариации словосочетания «включающий в себя», такие как «включают в себя» и «включает в себя», имеют
20 соответственно измененные значения.

Приложение А

transform_tree() и transform_unit() реализуют логически выведенное разделение цветности, используя конструкцию цикла

7.3.11 Синтаксис дерева преобразований

25

30

35

40

45

		Дескриптор
	<code>transform_tree(x0, y0, xBase, yBase, log2TrafoSize, trafoDepth, blkIdx) {</code>	
	<code>if(log2TrafoSize <= Log2MaxTrafoSize &&</code>	
	<code>log2TrafoSize > Log2MinTrafoSize &&</code>	
	<code>trafoDepth < MaxTrafoDepth && !(IntraSplitFlag && trafoDepth == 0))</code>	
5	<code>split_transform_flag[x0][y0][trafoDepth]</code>	ae(v)
	<code>if(trafoDepth == 0 log2TrafoSize > 2) {</code>	
	<code>if(trafoDepth == 0 cbf_cb[xBase][yBase][trafoDepth - 1]) {</code>	
	<code>for(tldx = 0; tldx < TrafoCrCbCnt; tldx++) {</code>	
10	<code>cbf_cb[x0 + ((1 << log2CrCbTrafoHorSize) * (tldx mod</code>	ae(v)
	<code>TrafoCrCbHorCnt)][y0 + (1 << log2CrCbTrafoVertSize) * (tldx div</code>	
	<code>TrafoCrCbVertCnt))][trafoDepth + (TrafoCrCbCnt > 1)]</code>	
	<code>}</code>	
	<code>cbf_cb[x0][y0][trafoDepth] = (TrafoCrCbCnt > 1)</code>	
	<code>}</code>	
	<code>if(trafoDepth == 0 cbf_cr[xBase][yBase][trafoDepth - 1]) {</code>	
15	<code>for(tldx = 0; tldx < TrafoCrCbCnt; tldx++) {</code>	
	<code>cbf_cr[x0 + ((1 << log2CrCbTrafoHorSize) * (tldx mod</code>	ae(v)
	<code>TrafoCrCbHorCnt)][y0 + (1 << log2CrCbTrafoVertSize) * (tldx div</code>	
	<code>TrafoCrCbVertCnt))][trafoDepth + (TrafoCrCbCnt > 1)]</code>	
	<code>}</code>	
20	<code>cbf_cr[x0][y0][trafoDepth] = (TrafoCrCbCnt > 1)</code>	
	<code>}</code>	
	<code>}</code>	
	<code>if(split_transform_flag[x0][y0][trafoDepth]) {</code>	
	<code>x1 = x0 + ((1 << log2TrafoSize) >> 1)</code>	
25	<code>y1 = y0 + ((1 << log2TrafoSize) >> 1)</code>	
	<code>transform_tree(x0, y0, x0, y0, log2TrafoSize - 1, trafoDepth + 1, 0)</code>	
	<code>transform_tree(x1, y0, x0, y0, log2TrafoSize - 1, trafoDepth + 1, 1)</code>	
	<code>transform_tree(x0, y1, x0, y0, log2TrafoSize - 1, trafoDepth + 1, 2)</code>	
	<code>transform_tree(x1, y1, x0, y0, log2TrafoSize - 1, trafoDepth + 1, 3)</code>	
	<code>} else {</code>	
30	<code>if(PredMode[x0][y0] == MODE_INTRA trafoDepth != 0 </code>	
	<code>cbf_cb[x0][y0][trafoDepth] cbf_cr[x0][y0][trafoDepth])</code>	
	<code>cbf_luma[x0][y0][trafoDepth]</code>	ae(v)
	<code>transform_unit(x0, y0, xBase, yBase, log2TrafoSize, trafoDepth, blkIdx)</code>	
	<code>}</code>	
35	<code>}</code>	

7.3.12 Синтаксис единицы преобразования

40

45

	transform_unit(x0, y0, xBase, yBase, log2TrafoSize, trafoDepth, blkIdx) {	Дескриптор
	if(cbf_luma[x0][y0][trafoDepth] cbf_cb[x0][y0][trafoDepth] cbf_cr[x0][y0][trafoDepth]) {	
	if(cu_qp_delta_enabled_flag && !IsCuQpDeltaCoded) {	
5	cu_qp_delta_abs	ae(v)
	if(cu_qp_delta_abs)	
	cu_qp_delta_sign	ae(v)
	}	
	if(cbf_luma[x0][y0][trafoDepth])	
10	residual_coding(x0, y0, log2TrafoSize, 0)	
	if(log2TrafoSize > 2) {	
	if(cbf_cb[x0][y0][trafoDepth])	
	for (tIdx = 0; tIdx < TrafoCrCbCnt; tIdx++) {	
15	residual_coding(x0 + ((1 << log2CrCbTrafoHorSize) * (tIdx mod TrafoCrCbHorCnt), y0 + (1 << log2CrCbTrafoVertSize) * (tIdx div TrafoCrCbVertCnt)), log2TrafoSize, 1)	
	}	
	if(cbf_cr[x0][y0][trafoDepth])	
	for (tIdx = 0; tIdx < TrafoCrCbCnt; tIdx++) {	
20	residual_coding(x0 + ((1 << log2CrCbTrafoHorSize) * (tIdx mod TrafoCrCbHorCnt), y0 + (1 << log2CrCbTrafoVertSize) * (tIdx div TrafoCrCbVertCnt)), log2TrafoSize, 2)	
	}	
	} else if(blkIdx == 3) {	
	if(cbf_cb[xBase][yBase][trafoDepth])	
25	residual_coding(xBase, yBase, log2TrafoSize, 1)	
	if(cbf_cr[xBase][yBase][trafoDepth])	
	residual_coding(xBase, yBase, log2TrafoSize, 2)	
	}	
	}	
30	}	

7.4.8.1 Общая семантика единицы кодирования

Переменные TrafoCrCbHorCnt и TrafoCrCbVertCnt получаются следующим образом:

- Если log2TrafoSize равен 5, и split_transform_flag равен 0, TransformIdxMax получается следующим образом:

- 35
- Если chroma_format_idc равен 1, TrafoCrCbHorCnt и TrafoCrCbVertCnt равны 1.
 - Если chroma_format_idc равен 2, TrafoCrCbHorCnt равен 1 и TrafoCrCbVertCnt равен 2.
 - 2. - Если chroma_format_idc равен 3, TrafoCrCbHorCnt и TrafoCrCbVertCnt равны 2.
 - Иначе TrafoCrCbHorCnt и TrafoCrCbVertCnt равны 1.

40 Переменная TrafoCrCbCnt получается как TrafoCrCbHorCnt * TrafoCrCbVertCnt.

Переменные log2CrCbTrafoHorSize и log2CrCbTrafoVertSize получаются следующим образом:

- Если chroma_format_idc равен 1, log2CrCbTrafoHorSize, и log2CrCbTrafoVertSize равны log2TrafoSize - 1.
- 45 - Иначе, если chroma_format_idc равен 2, log2CrCbTrafoHorSize равен log2TrafoSize и log2CrCbTrafoVertSize равен min(log2TrafoSize - 1, 4).
- Иначе, если chroma_format_idc равен 3, log2CrCbTrafoHorSize и log2CrCbTrafoVertSize равны min(log2TrafoSize, 4).

Конец Приложения А

Приложение В

Процедура transform_tree() вызывается один раз для пары каналов цветности для каждого преобразования цветности, следующего из ЛОГИЧЕСКИ выведенного

5 разделения.

7.3.11 Синтаксис дерева преобразований

	Дескриптор
	transform_tree(x0, y0, xBase, yBase, log2TrafoSize, trafoDepth, blkIdx, chromaOnly) {
10	if(log2TrafoSize <= Log2MaxTrafoSize && log2TrafoSize > Log2MinTrafoSize && trafoDepth < MaxTrafoDepth && !(IntraSplitFlag && trafoDepth == 0) && !chromaOnly)
	split_transform_flag[x0][y0][trafoDepth]
	ac(v)
	if(trafoDepth == 0 log2TrafoSize > 2) {
	if(trafoDepth == 0 cbf_cb[xBase][yBase][trafoDepth - 1])
15	if(TrafoCrCbCnt > 1) {
	cbf_cb[x0][y0][trafoDepth] = 1
	} else {
	cbf_cb[x0][y0][trafoDepth]
	ac(v)
	}
20	if(trafoDepth == 0 cbf_cr[xBase][yBase][trafoDepth - 1])
	if(TrafoCrCbCnt > 1) {
	cbf_cr[x0][y0][trafoDepth] = 1
	} else {
	cbf_cr[x0][y0][trafoDepth]
	ac(v)
25	}
	}
	if(split_transform_flag[x0][y0][trafoDepth] TrafoCrCbCnt > 1) {
	x1 = x0 + ((1 << log2TrafoSize) >> 1)
	y1 = y0 + ((1 << log2TrafoSize) >> 1)
30	transform_tree(x0, y0, x0, y0, log2TrafoSize - 1, trafoDepth + 1, 0, TrafoCrCbCnt > 1)
	if(chroma_format_idc != 2) {
	transform_tree(x1, y0, x0, y0, log2TrafoSize - 1, trafoDepth + 1, 1, TrafoCrCbCnt > 1)
	}
35	transform_tree(x0, y1, x0, y0, log2TrafoSize - 1, trafoDepth + 1, 2, TrafoCrCbCnt > 1)
	if(chroma_format_idc != 2) {
	transform_tree(x1, y1, x0, y0, log2TrafoSize - 1, trafoDepth + 1, 3, TrafoCrCbCnt > 1)
40	}
	}

45

	}	
	else-if (!split_transform_flag[x0][y0][trafoDepth] && TrafoCrCbCnt > 1) {	
	if((PredMode[x0][y0] == MODE_INTRA trafoDepth != 0 cbf_cb[x0][y0][trafoDepth] cbf_cr[x0][y0][trafoDepth]) && !chromaOnly)	
5	cbf_luma [x0][y0][trafoDepth]	ae(v)
	transform_unit(x0, y0, xBase, yBase, log2TrafoSize, trafoDepth, blkIdx, chromaOnly)	
	}	
	}	

7.3.12 Синтаксис единицы преобразования

	transform_unit (x0, y0, xBase, yBase, log2TrafoSize, trafoDepth, blkIdx, chromaOnly) {	Дескриптор
	if(cbf_luma[x0][y0][trafoDepth] cbf_cb[x0][y0][trafoDepth] cbf_cr[x0][y0][trafoDepth]) {	
15	if(cu_qp_delta_enabled_flag && !IsCuQpDeltaCoded && !chromaOnly) {	
	cu_qp_delta_abs	ae(v)
	if(cu_qp_delta_abs)	
	cu_qp_delta_sign	ae(v)
	}	
20	if(cbf_luma[x0][y0][trafoDepth])	
	residual_coding(x0, y0, log2TrafoSize, 0)	
	if(log2TrafoSize > 2) {	
	if(cbf_cb[x0][y0][trafoDepth])	
	residual_coding(x0, y0, log2TrafoSize, 1)	
25	if(cbf_cr[x0][y0][trafoDepth])	
	residual_coding(x0, , log2TrafoSize, 2)	
	} else if(blkIdx == 3) {	
	if(cbf_cb[xBase][yBase][trafoDepth])	
	residual_coding(xBase, yBase, log2TrafoSize, 1)	
30	if(cbf_cr[xBase][yBase][trafoDepth])	
	residual_coding(xBase, yBase, log2TrafoSize, 2)	
	}	
	}	
	}	

7.4.8.1 Общая семантика единицы кодирования

Переменные TrafoCrCbHorCnt и TrafoCrCbVertCnt получаются следующим образом:

- Если log2TrafoSize равен 5, и split_transform_flag равен 0, TransformIdxMax получается следующим образом:

- Если chroma_format_idc равен 1, TrafoCrCbHorCnt и TrafoCrCbVertCnt равны 1.

- Если chroma_format_idc равен 2, TrafoCrCbHorCnt равен 1 и TrafoCrCbVertCnt равен 2.

- Если chroma_format_idc равен 3, TrafoCrCbHorCnt и TrafoCrCbVertCnt равны 2.

- Иначе TrafoCrCbHorCnt и TrafoCrCbVertCnt равны 1.

Переменная TrafoCrCbCnt получается как TrafoCrCbHorCnt * TrafoCrCbVertCnt.

Конец Приложения В.

Формула изобретения

1. Способ декодирования единицы преобразования, содержащей остаточные отсчеты сигнала цветности, из битового видеопотока, причем единица преобразования содержит

по меньшей мере один массив остаточных коэффициентов сигнала цветности, связанный с единственным каналом цветности в формате цветности 4:2:2, причем способ содержит:

определение иерархического уровня для единицы преобразования в пределах единицы кодирования из флагов преобразования разделения, присутствующих в битовом видеопотоке, при этом размер единицы кодирования находится в диапазоне от

наименьшей единицы кодирования до наибольшей единицы кодирования;

определение размера преобразования единицы преобразования для единственного канала цветности, причем размер преобразования связан с определенным иерархическим уровнем единицы преобразования и размером единицы кодирования;

декодирование множества значений флага кодированного блока из битового видеопотока для единственного канала цветности единицы преобразования, при этом единица преобразования имеет множество массивов остаточных коэффициентов сигнала цветности для единственного цветового канала, и каждое значение флага кодированного блока из упомянутого множества значений флага кодированного блока соответствует одному массиву остаточных коэффициентов сигнала цветности из массивов остаточных коэффициентов сигнала цветности;

декодирование из битового видеопотока каждого из множества массивов остаточных коэффициентов сигнала цветности согласно соответствующему значению флага кодированного блока из упомянутого множества значений флага кодированного блока;

выбор квадратного обратного преобразования для декодированных массивов остаточных коэффициентов сигнала цветности, причем квадратное обратное преобразование выбирается из predetermined набора квадратных обратных преобразований согласно определенному размеру преобразования; и

применение выбранного квадратного обратного преобразования к каждому из декодированных массивов остаточных коэффициентов сигнала цветности для получения остаточных отсчетов сигнала цветности для канала цветности единицы преобразования.

2. Способ кодирования единицы преобразования, содержащей остаточные отсчеты сигнала цветности, в битовый видеопоток, причем единица преобразования содержит по меньшей мере один массив остаточных коэффициентов сигнала цветности, связанный с единственным каналом цветности в формате цветности 4:2:2, причем способ содержит:

кодирование флагов преобразования разделения в битовый видеопоток на основе принятого иерархического уровня для единицы преобразования в пределах единицы кодирования, причем размер единицы кодирования находится в диапазоне от наименьшей единицы кодирования до наибольшей единицы кодирования;

прием размера преобразования единицы преобразования для единственного канала цветности, причем размер преобразования связан с иерархическим уровнем единицы преобразования и размером единицы кодирования;

кодирование множества значений флага кодированного блока в битовый видеопоток для единственного канала цветности единицы преобразования, при этом единица преобразования имеет множество массивов остаточных коэффициентов сигнала цветности для единственного цветового канала, и каждое значение флага кодированного блока из упомянутого множества значений флага кодированного блока соответствует одному массиву остаточных коэффициентов сигнала цветности из массивов остаточных коэффициентов сигнала цветности;

выбор квадратного прямого преобразования для массивов остаточных коэффициентов сигнала цветности, причем квадратное прямое преобразование выбирается из predetermined набора квадратных прямых преобразований согласно принятому размеру преобразования;

применение выбранного квадратного прямого преобразования к каждому из массивов остаточных коэффициентов сигнала цветности для получения остаточных отсчетов сигнала цветности для канала

цветности единицы преобразования; и

5 кодирование в битовый видеопоток каждого из упомянутого множества массивов остаточных коэффициентов сигнала цветности согласно соответствующему значению флага кодированного блока из упомянутого множества значений флага кодированного блока.

3. Способ по п. 1, в котором значение флага кодированного блока, равное нулю, 10 указывает, что все остаточные коэффициенты соответствующего массива остаточных коэффициентов равны нулю, а значение флага кодированного блока, равное единице, указывает, что по меньшей мере один остаточный коэффициент соответствующего массива остаточных коэффициентов является ненулевым.

4. Способ по п. 1, в котором размер преобразования является одним из 32×32 , 16×16 , 15 8×8 или 4×4 .

5. Способ по п. 1, в котором существуют два массива остаточных коэффициентов сигнала цветности для единственного цветового канала, и два значения флага кодированного блока декодируются из битового видеопотока для единственного 20 цветового канала.

6. Способ по п. 2, в котором значение флага кодированного блока, равное нулю, указывает, что все остаточные коэффициенты соответствующего массива остаточных 25 коэффициентов равны нулю, а значение флага кодированного блока, равное единице, указывает, что по меньшей мере один остаточный коэффициент соответствующего массива остаточных коэффициентов является ненулевым.

7. Способ по п. 2, в котором размер преобразования является одним из 32×32 , 16×16 , 30 8×8 или 4×4 .

8. Способ по п. 2, в котором существуют два массива остаточных коэффициентов сигнала цветности для единственного цветового канала, и два значения флага кодированного блока кодируются в битовый видеопоток для единственного цветового 35 канала.

9. Видеодекодер для декодирования единицы преобразования, содержащей остаточные отсчеты сигнала цветности, из битового видеопотока, причем единица преобразования содержит по меньшей мере один массив остаточных коэффициентов сигнала цветности, 40 связанный с единственным каналом цветности в формате цветности 4:2:2, причем декодер содержит:

первый определитель для определения иерархического уровня для единицы преобразования в пределах единицы кодирования из флагов преобразования разделения, присутствующих в битовом видеопотоке, при этом размер единицы кодирования находится в диапазоне от наименьшей единицы кодирования до наибольшей единицы 45 кодирования;

второй определитель для определения размера преобразования единицы преобразования для единственного канала цветности, причем размер преобразования связан с определенным иерархическим уровнем единицы преобразования и размером единицы кодирования;

45 первый декодер для декодирования множества значений флага кодированного блока из битового видеопотока для единственного канала цветности единицы преобразования, при этом единица преобразования имеет множество массивов остаточных коэффициентов сигнала цветности для единственного цветового канала, и

каждое значение флага кодированного блока упомянутого множества значений флага кодированного блока соответствует одному массиву остаточных коэффициентов сигнала цветности из массивов остаточных коэффициентов сигнала цветности;

5 второй декодер для декодирования из битового видеопотока каждого из множества массивов остаточных коэффициентов сигнала цветности согласно соответствующему значению флага кодированного блока из упомянутого множества значений флага блока кодирования;

средство выбора для выбора квадратного обратного преобразования для декодированных массивов остаточных коэффициентов сигнала цветности, причем
10 квадратное обратное преобразование выбирается из predetermined набора квадратных обратных преобразований согласно определенному размеру преобразования; и

средство применения для применения выбранного квадратного обратного преобразования к каждому из декодированных массивов остаточных коэффициентов
15 сигнала цветности для получения остаточных отсчетов сигнала цветности для канала цветности единицы преобразования.

10. Декодер по п. 9, в котором значение флага кодированного блока, равное нулю, указывает, что все остаточные коэффициенты соответствующего массива остаточных
20 коэффициентов равны нулю, а значение флага кодированного блока, равное единице, указывает, что по меньшей мере один остаточный коэффициент соответствующего массива остаточных коэффициентов является ненулевым.

11. Декодер по п. 9, в котором размер преобразования является одним из 32×32 , 16×16 , 8×8 или 4×4 .

12. Декодер по п. 9, в котором существуют два массива остаточных коэффициентов
25 сигнала цветности для единственного цветового канала, и два значения флага кодированного блока декодируются из битового видеопотока для единственного цветового канала.

13. Невременной считываемый компьютером носитель данных, имеющий записанную на нем программу, причем программа исполняется процессором для декодирования
30 единицы преобразования, содержащей остаточные отсчеты сигнала цветности, из битового видеопотока, причем единица преобразования содержит по меньшей мере один массив остаточных коэффициентов сигнала цветности, связанный с единственным каналом цветности в формате цветности 4:2:2, причем программа содержит:

код для определения иерархического уровня для единицы преобразования в пределах
35 единицы кодирования из флагов преобразования разделения, присутствующих в битовом видеопотоке, при этом размер единицы кодирования находится в диапазоне от наименьшей единицы кодирования до наибольшей единицы кодирования;

код для определения размера преобразования единицы преобразования для
40 единственного канала цветности, причем размер преобразования связан с определенным иерархическим уровнем единицы преобразования и размером единицы кодирования;

код для декодирования множества значений флага кодированного блока из битового видеопотока для единственного канала цветности единицы преобразования, причем
единица преобразования имеет множество массивов остаточных коэффициентов
сигнала цветности для единственного цветового канала, и каждое значение флага
45 кодированного блока из упомянутого множества значений флага кодированного блока соответствует одному массиву остаточных коэффициентов сигнала цветности из массивов остаточных коэффициентов сигнала цветности;

код для декодирования из битового видеопотока каждого из множества массивов

остаточных коэффициентов сигнала цветности согласно соответствующему значению флага кодированного блока из упомянутого множества значений флага кодированного блока;

код для выбора квадратного обратного преобразования для декодированных массивов остаточных коэффициентов сигнала цветности, квадратное обратное преобразование выбирается из predetermined набора квадратных обратных преобразований согласно определенному размеру преобразования; и

код для применения выбранного квадратного обратного преобразования к каждому из декодированных массивов остаточных коэффициентов сигнала цветности для получения остаточных отсчетов сигнала цветности для канала цветности единицы преобразования.

14. Невременной считываемый компьютером носитель данных по п. 13, в котором значение флага кодированного блока, равное нулю, указывает, что все остаточные коэффициенты соответствующего массива остаточных коэффициентов равны нулю, а значение флага кодированного блока, равное единице, указывает, что по меньшей мере один остаточный коэффициент соответствующего массива остаточных коэффициентов является ненулевым.

15. Невременной считываемый компьютером носитель данных по п. 13, в котором размер преобразования является одним из 32×32, 16×16, 8×8 или 4×4.

16. Невременной считываемый компьютером носитель данных по п. 13, в котором существуют два массива остаточных коэффициентов сигнала цветности для единственного цветового канала, и два значения флага кодированного блока декодируются из битового видеопотока для единственного цветового канала.

17. Видеокодер для кодирования единицы преобразования, содержащей остаточные отсчеты сигнала цветности, в битовый видеопоток, причем единица преобразования содержит по меньшей мере один массив остаточных коэффициентов сигнала цветности, связанный с единственным каналом цветности в формате цветности 4:2:2, причем кодер содержит:

первый кодер для кодирования флагов преобразования разделения в битовый видеопоток на основе принятого иерархического уровня для единицы преобразования в пределах единицы кодирования, при этом размер единицы кодирования находится в диапазоне от наименьшей единицы кодирования до наибольшей единицы кодирования;

приемник для приема размера преобразования единицы преобразования для единственного канала цветности, причем размер преобразования связан с иерархическим уровнем единицы преобразования и размером единицы кодирования;

второй кодер для кодирования множества значений флага кодированного блока в битовый видеопоток для единственного канала цветности единицы преобразования, при этом единица преобразования имеет множество массивов остаточных коэффициентов сигнала цветности для единственного цветового канала, и каждое значение флага кодированного блока упомянутого множества значений флага кодированного блока соответствует одному массиву остаточных коэффициентов сигнала цветности из массивов остаточных коэффициентов сигнала цветности;

средство выбора для выбора квадратного прямого преобразования для массивов остаточных коэффициентов сигнала цветности, причем квадратное прямое преобразование выбирается из predetermined набора квадратных прямых преобразований согласно принятому размеру преобразования;

средство применения для применения выбранного квадратного прямого преобразования к каждому из массивов остаточных коэффициентов сигнала цветности

для получения остаточных отсчетов сигнала цветности для канала цветности единицы преобразования; и

третий кодер для кодирования в битовый видеопоток каждого из упомянутого множества массивов остаточных коэффициентов

5 сигнала цветности согласно соответствующему значению флага кодированного блока из упомянутого множества значений флага кодированного блока.

18. Невременной считываемый компьютером носитель данных, имеющий записанную на нем программу, причем программа исполняется процессором для кодирования единицы преобразования, содержащей остаточные отсчеты сигнала цветности, в битовый
10 видеопоток, причем единица преобразования содержит по меньшей мере один массив остаточных коэффициентов сигнала цветности, связанный с единственным каналом цветности в формате цветности 4:2:2, причем программа содержит:

код для кодирования флагов преобразования разделения в битовый видеопоток на основе принятого иерархического уровня для единицы преобразования в пределах
15 единицы кодирования, при этом размер единицы кодирования находится в диапазоне от наименьшей единицы кодирования до наибольшей единицы кодирования;

код для приема размера преобразования единицы преобразования для единственного канала цветности, причем размер преобразования связан с иерархическим уровнем единицы преобразования и размером единицы кодирования;

20 код для кодирования множества значений флага кодированного блока в битовый видеопоток для единственного канала цветности единицы преобразования, при этом единица преобразования имеет множество массивов остаточных коэффициентов сигнала цветности для единственного цветового канала, и каждое значение флага кодированного блока из упомянутого множества значений флага кодированного блока соответствует
25 одному массиву остаточных коэффициентов сигнала цветности из массивов остаточных коэффициентов сигнала цветности;

код для выбора квадратного прямого преобразования для массивов остаточных коэффициентов сигнала цветности, причем квадратное прямое преобразование выбирается из предопределенного набора квадратных прямых преобразований согласно
30 принятому размеру преобразования;

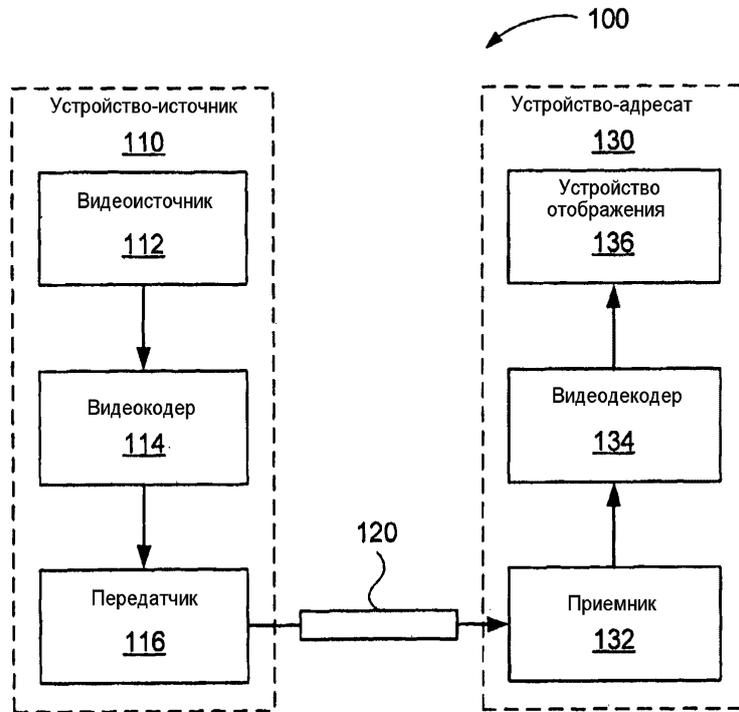
код для применения выбранного квадратного прямого преобразования к каждому из массивов остаточных коэффициентов

сигнала цветности для получения остаточных отсчетов сигнала цветности для канала цветности единицы преобразования; и

35 код для кодирования в битовый видеопоток каждого из упомянутого множества массивов остаточных коэффициентов сигнала цветности согласно соответствующему значению флага кодированного блока из упомянутого множества значений флага кодированного блока.

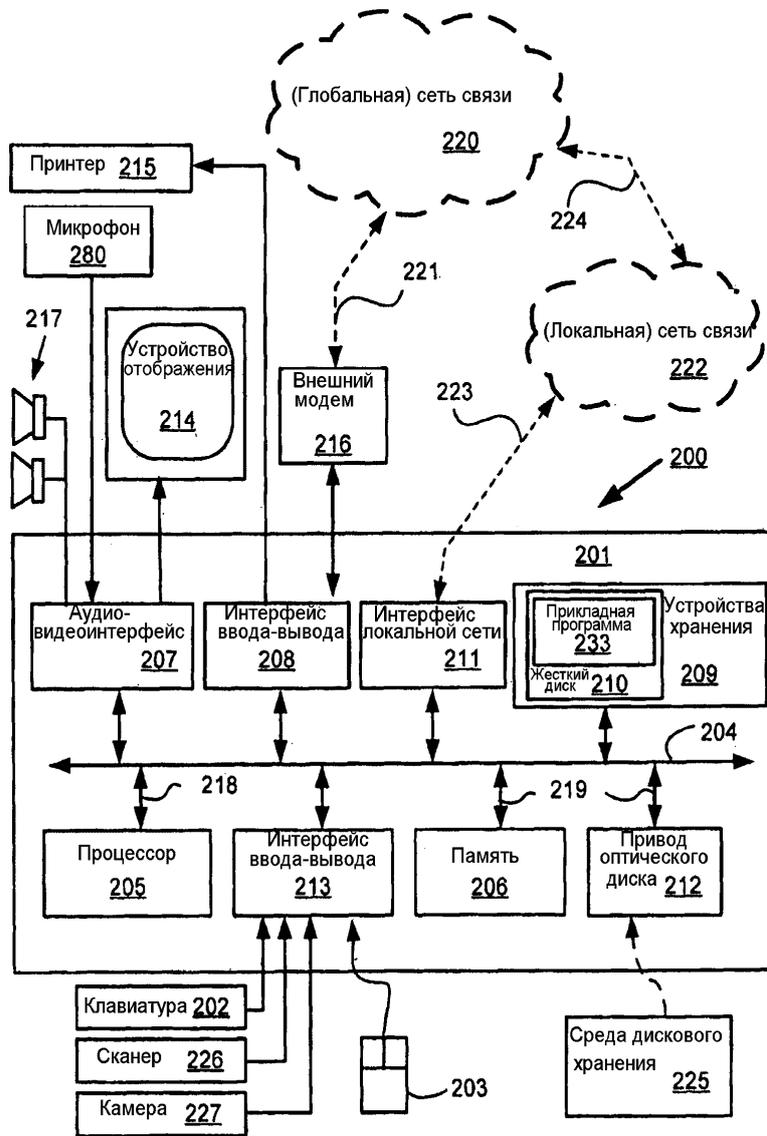
19. Невременной считываемый компьютером носитель данных по п. 18, в котором значение флага кодированного блока, равное нулю, указывает, что все остаточные коэффициенты соответствующего массива остаточных коэффициентов равны нулю, а значение флага кодированного блока, равное единице, указывает, что по меньшей мере один остаточный коэффициент соответствующего массива остаточных коэффициентов является ненулевым.

45 20. Невременной считываемый компьютером носитель данных по п. 18, в котором существуют два массива остаточных коэффициентов сигнала цветности для единственного цветового канала, и два значения флага кодированного блока декодируются из битового видеопотока для единственного цветового канала.

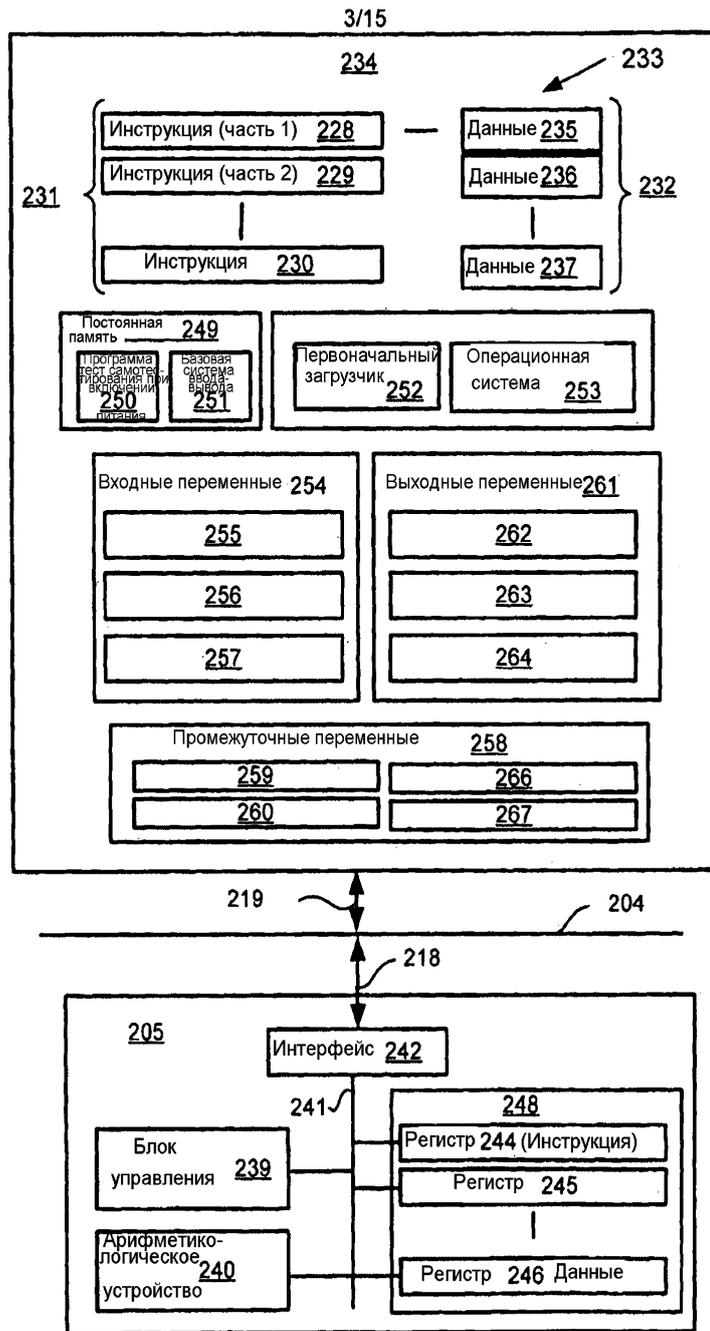


ФИГ.1

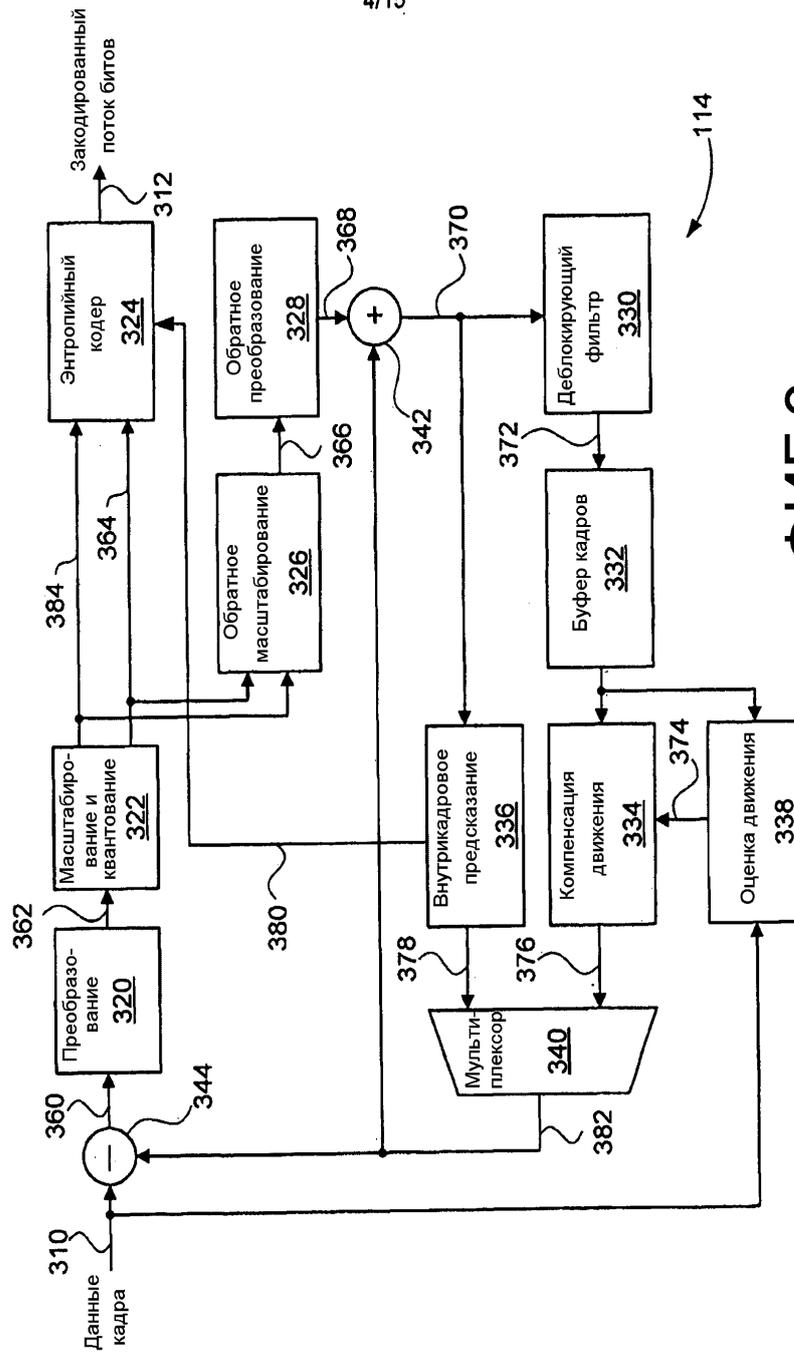
2/15



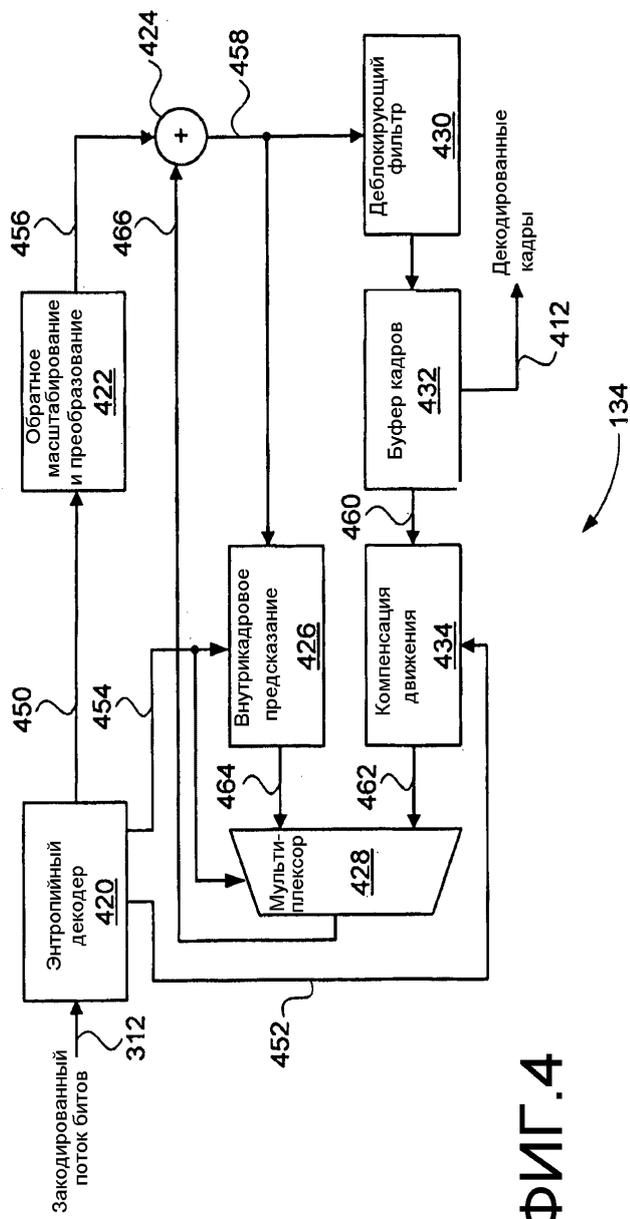
ФИГ.2А



ФИГ.2В

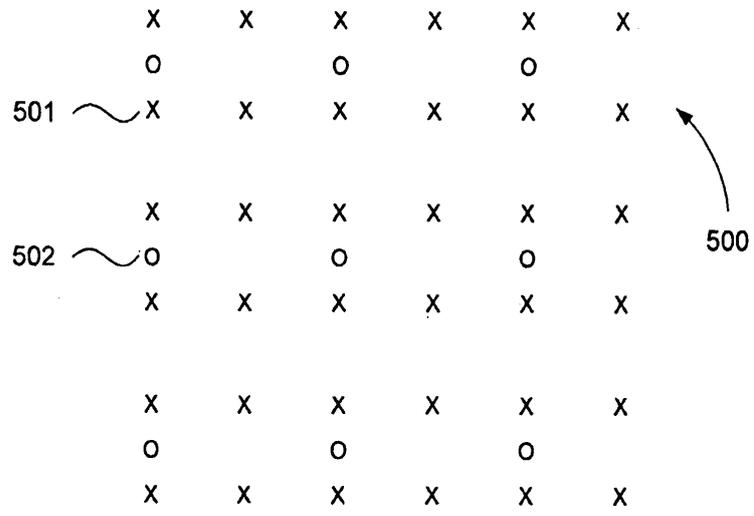


ФИГ.3

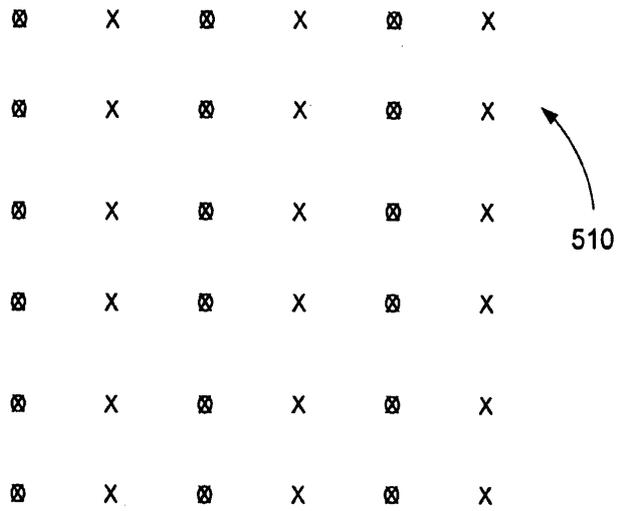


ФИГ.4

6/15

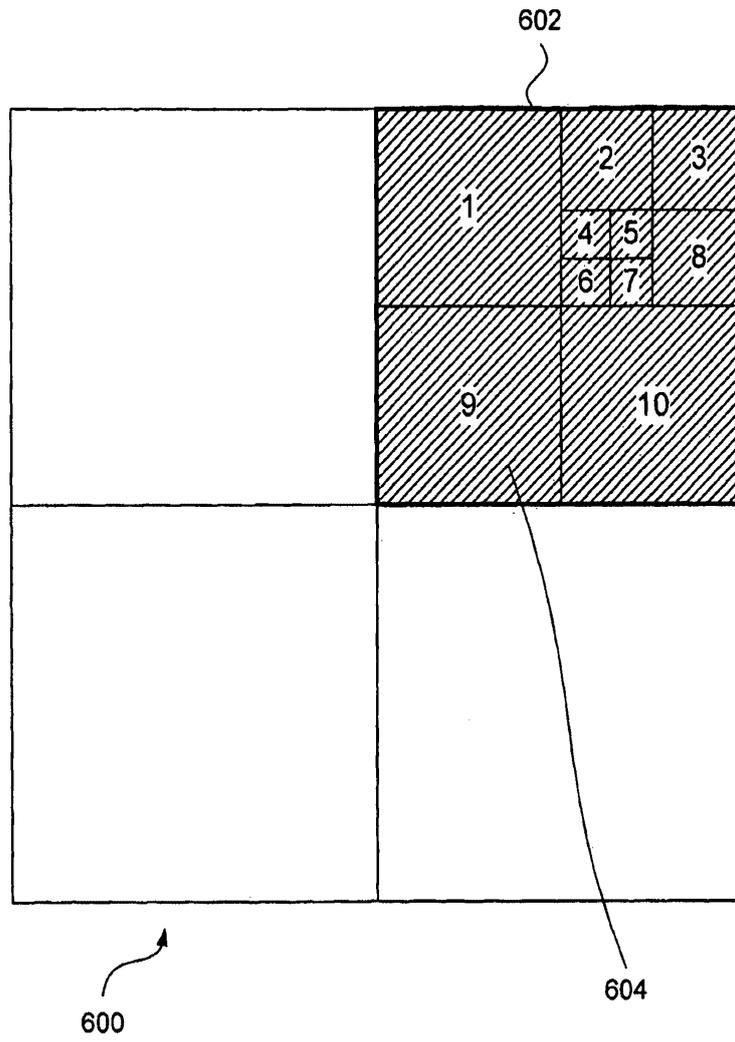


ФИГ.5А



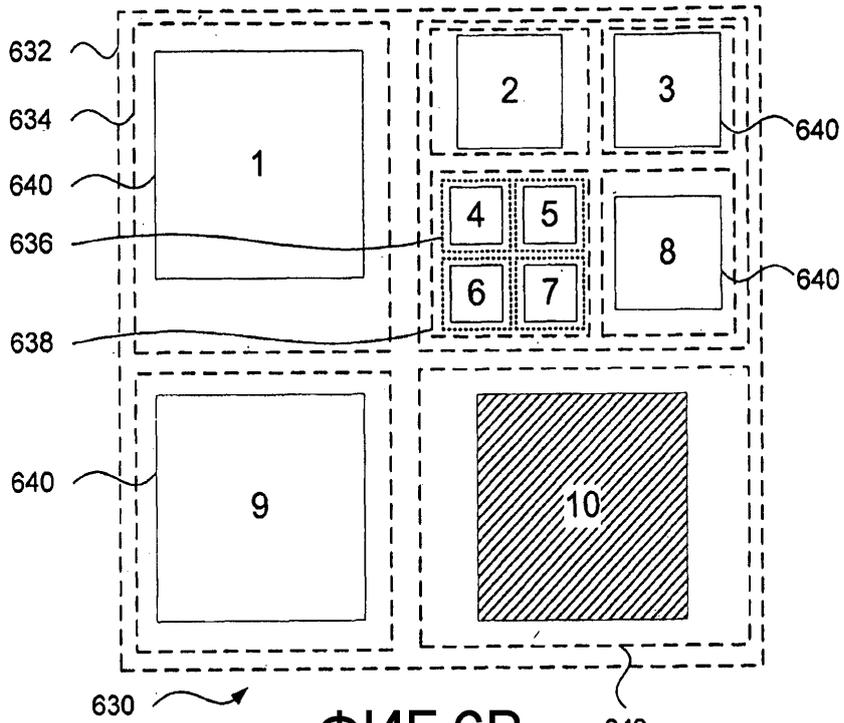
ФИГ.5В

7/15

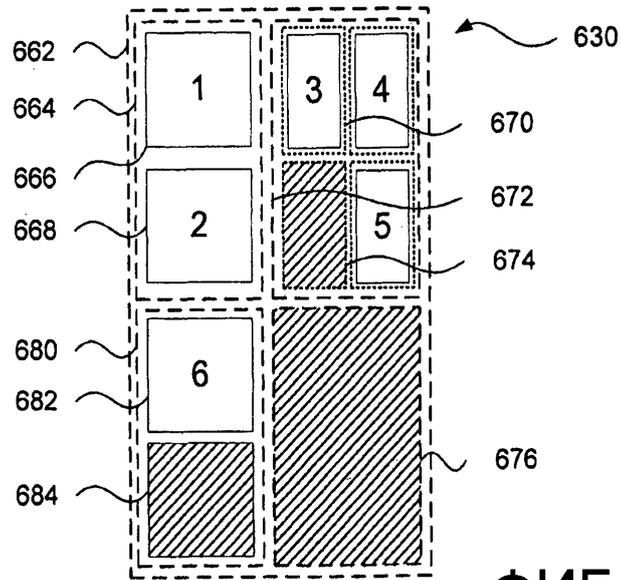


ФИГ.6А

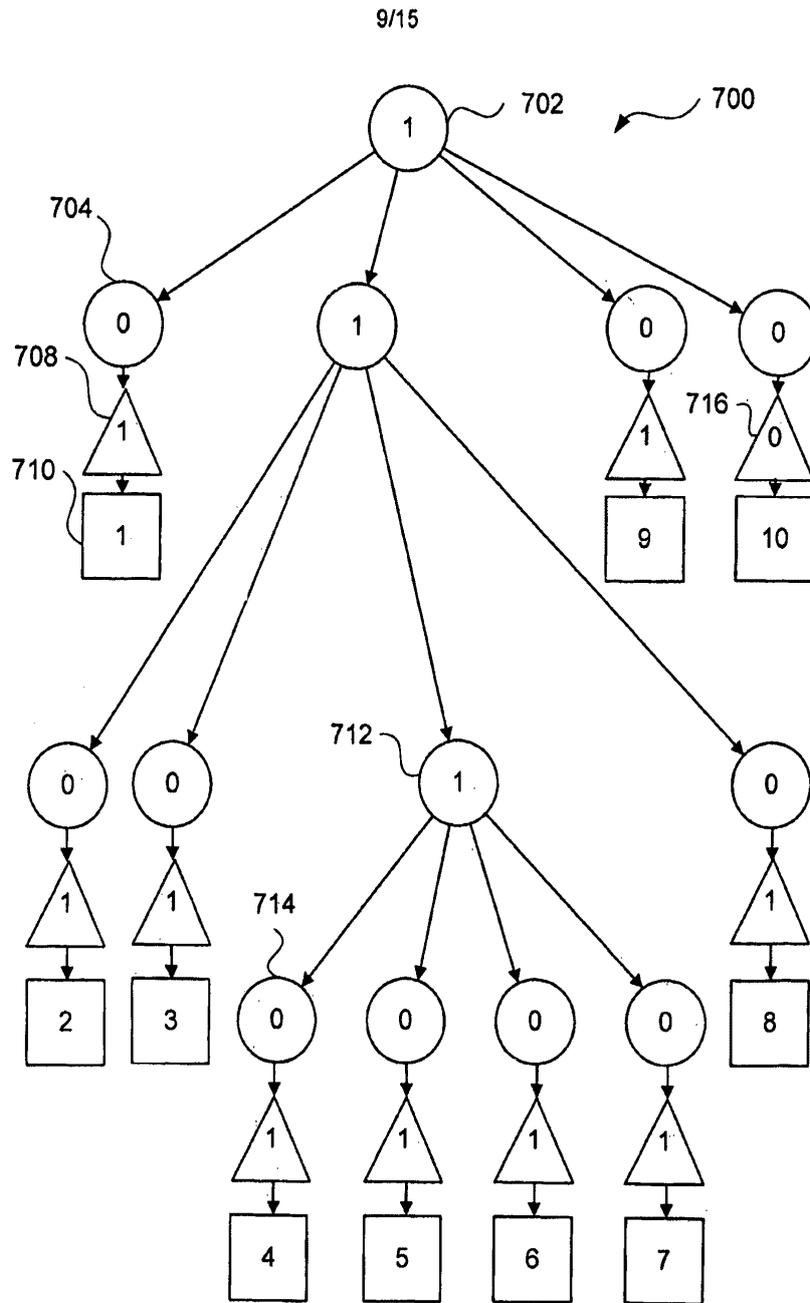
8/15



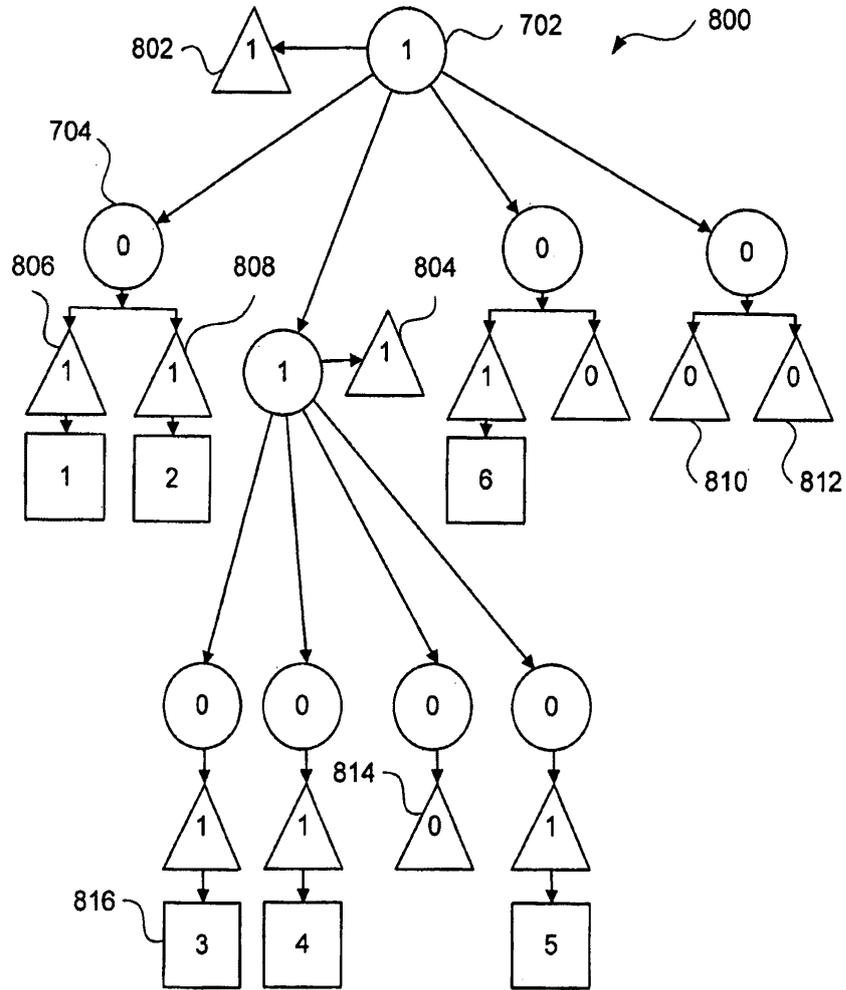
ФИГ.6В



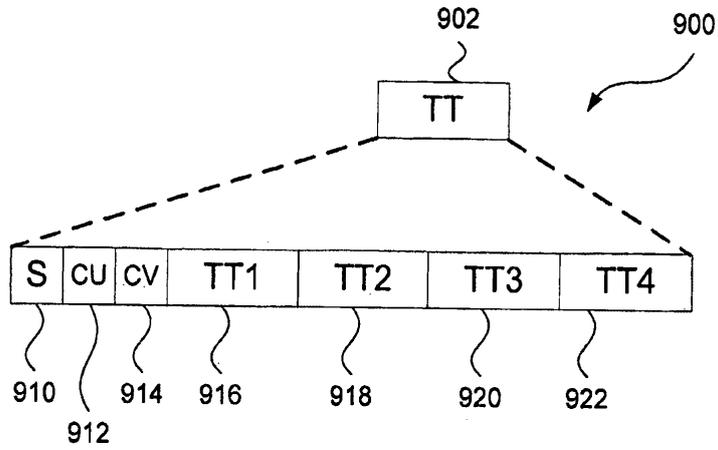
ФИГ.6С



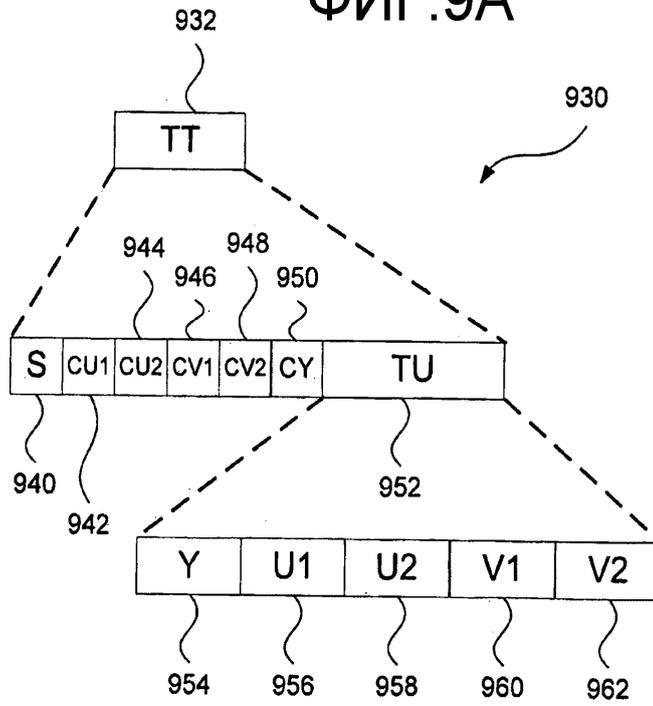
ФИГ.7



ФИГ.8

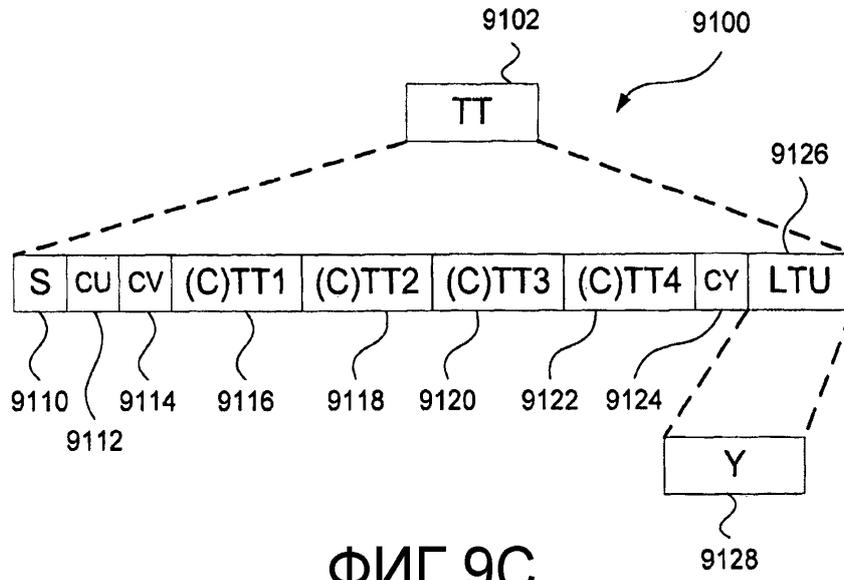


ФИГ.9А

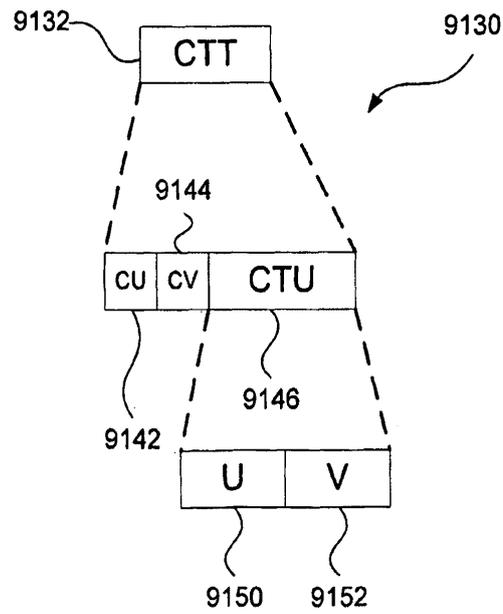


ФИГ.9В

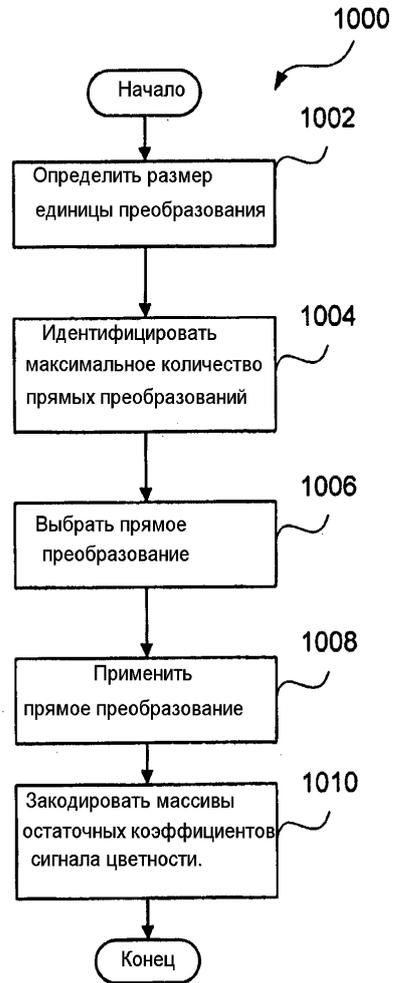
12/15



ФИГ.9С

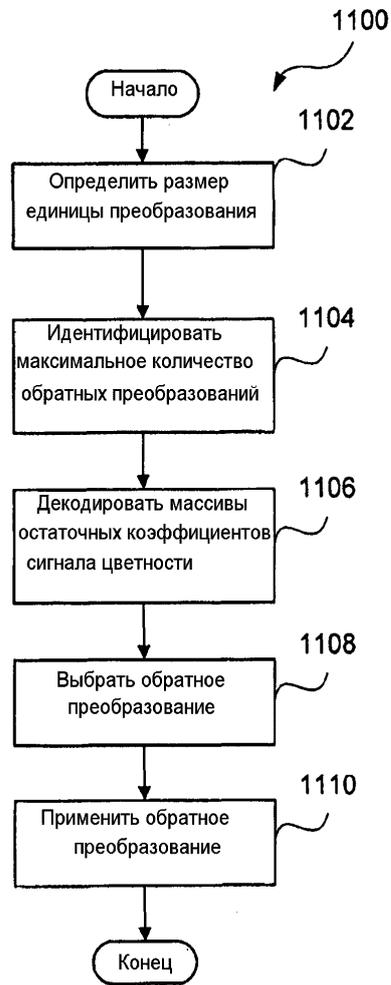


ФИГ.9D

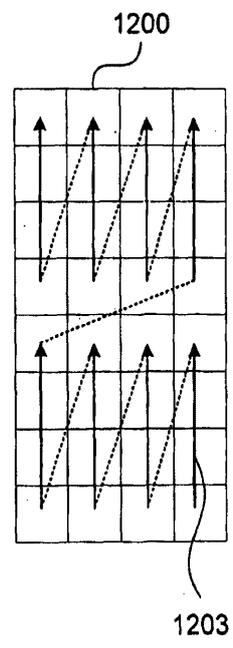
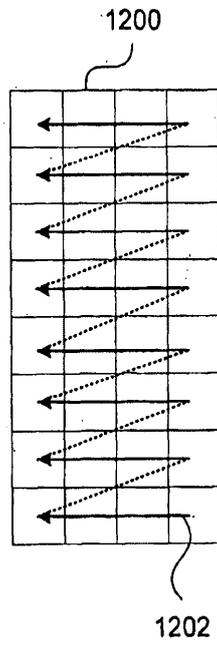
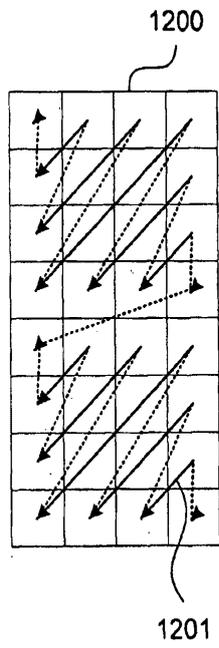


ФИГ.10

14/15



ФИГ.11



ФИГ.12А

ФИГ.12В

ФИГ.12С