



ФЕДЕРАЛЬНАЯ СЛУЖБА  
ПО ИНТЕЛЛЕКТУАЛЬНОЙ СОБСТВЕННОСТИ

## (12) ОПИСАНИЕ ИЗОБРЕТЕНИЯ К ПАТЕНТУ

(21)(22) Заявка: 2015117520, 20.09.2013

(24) Дата начала отсчета срока действия патента:  
20.09.2013Дата регистрации:  
11.10.2017

Приоритет(ы):

(30) Конвенционный приоритет:  
08.10.2012 US 61/711,098;  
30.07.2013 US 13/954,758

(43) Дата публикации заявки: 27.11.2016 Бюл. № 33

(45) Опубликовано: 11.10.2017 Бюл. № 29

(85) Дата начала рассмотрения заявки РСТ на  
национальной фазе: 08.05.2015(86) Заявка РСТ:  
US 2013/060940 (20.09.2013)(87) Публикация заявки РСТ:  
WO 2014/058600 (17.04.2014)Адрес для переписки:  
129090, Москва, ул. Б. Спасская, 25, стр. 3, ООО  
"Юридическая фирма Городиский и Партнеры"(72) Автор(ы):  
ВАН Е-Куй (US)(73) Патентообладатель(и):  
КВЭЛКОММ ИНКОРПОРЕЙТЕД (US)(56) Список документов, цитированных в отчете  
о поиске: US 2010/0142613 A1, 10.06.2010. М.  
М. HANNUKSELA et al. "AHG9: Operation  
points in VPS and nesting SEI", JCTVC-K0180  
(version 1), опубли. 01.10.2012 на 5 страницах  
[найден 02.02.2017], размещено в Интернет  
по адресу URL: [http://phenix.it-sudparis.eu/jct/doc\\_end\\_user/current\\_document.php?id=6551](http://phenix.it-sudparis.eu/jct/doc_end_user/current_document.php?id=6551).  
WO 2012/018951 A1, 09.02.2012. WO 2008/  
085433 A2, (см. прод.)

## (54) ПРИМЕНИМОСТЬ БИТОВОГО СУБПОТОКА К ВЛОЖЕННЫМ SEI-СООБЩЕНИЯМ ПРИ КОДИРОВАНИИ ВИДЕО

(57) Реферат:

Изобретение относится к вычислительной технике. Технический результат заключается в использовании элемента синтаксиса идентификатора слоя для определения рабочей точки, применимой к масштабируемым вложенным сообщениям с дополнительной улучшающей информацией (SEI). Способ обработки видеоданных, в котором определяют, по меньшей мере, частично на основе элемента синтаксиса в SEI-сообщении, которое инкапсулировано посредством единицы уровня сетевой абстракции (NAL) SEI и инкапсулирует одно или более вложенных SEI-сообщений, то, являются или нет одно или более вложенных SEI-сообщений применимыми к битовому субпотoku

по умолчанию из кодированного битового потока видео, который содержит кодируемые изображения видеоданных, при этом элемент синтаксиса не находится в каком-либо из одного или более вложенных SEI-сообщений и битовый субпоток по умолчанию является представлением рабочей точки для рабочей точки, заданной посредством идентификатора слоя, точно определенного в заголовке NAL-единицы для SEI NAL-единицы, и временного идентификатора, точно определенного в заголовке NAL-единицы; и когда одно или более вложенных SEI-сообщений применимы к битовому субпотoku по умолчанию, используют одно или более вложенных SEI-сообщений при операции над битовым

субпотоком по умолчанию. 8 н. и 38 з.п. ф-лы, 9 ил., 5 табл.

(56) (продолжение):

17.07.2008. RU 2409910 C2, 20.01.2011.

RU 2 6 3 3 1 0 0 C 2

RU 2 6 3 3 1 0 0 C 2



FEDERAL SERVICE  
FOR INTELLECTUAL PROPERTY

(51) Int. Cl.

*H04N 19/46* (2014.01)*H04N 19/30* (2014.01)*H04N 19/70* (2014.01)(12) **ABSTRACT OF INVENTION**(21)(22) Application: **2015117520, 20.09.2013**(24) Effective date for property rights:  
**20.09.2013**Registration date:  
**11.10.2017**

Priority:

(30) Convention priority:  
**08.10.2012 US 61/711,098;**  
**30.07.2013 US 13/954,758**(43) Application published: **27.11.2016** Bull. № 33(45) Date of publication: **11.10.2017** Bull. № 29(85) Commencement of national phase: **08.05.2015**(86) PCT application:  
**US 2013/060940 (20.09.2013)**(87) PCT publication:  
**WO 2014/058600 (17.04.2014)**Mail address:  
**129090, Moskva, ul. B. Spasskaya, 25, str. 3, OOO**  
**"Yuridicheskaya firma Gorodisskij i Partnery"**

(72) Inventor(s):

**VAN E-Kuj (US)**

(73) Proprietor(s):

**KVELKOMM INKORPOREJTED (US)**(54) **APPLICABILITY OF BIT SUBSTREAM TO ATTACHED SEI-MESSAGES AT VIDEO ENCODING**

(57) Abstract:

FIELD: information technology.

SUBSTANCE: video data processing method that is determined at least partially based on a syntax element in a SEI-message that is encapsulated by network abstraction level (NAL) SEI unit and encapsulates one or more nested SEI-messages, whether or not one or more nested SEI-messages are applicable to the default bit substream from the encoded video bit stream that contains encoded video data images. The syntax element is not in any of the one or more nested SEI-messages, and the default bit substream is the work

point representation for the work point specified by the layer identifier specified exactly in NAL unit header for the SEI NAL unit and the temporary identifier, exactly defined in NAL unit header; and when one or more nested SEI-messages are applicable to the default bit substream, one or more nested SEI-messages are used for the default bit substream operation.

EFFECT: using the layer ID syntax element to define a work point applicable to scalable nested messages with additional enhancement information.

46 cl, 9 dwg, 5 tbl

[0001] Данная заявка испрашивает приоритет предварительной заявки на патент (США) №61/711098, поданной 8 октября 2012 года, содержимое которой полностью содержится в данном документе по ссылке.

## ОБЛАСТЬ ТЕХНИКИ, К КОТОРОЙ ОТНОСИТСЯ ИЗОБРЕТЕНИЕ

[0002] Данное раскрытие сущности относится к кодированию и декодированию видео.

## УРОВЕНЬ ТЕХНИКИ

[0003] Поддержка цифрового видео может быть включена в широкий диапазон устройств, включающих в себя цифровые телевизионные приемники, системы цифровой прямой широкополосной передачи, беспроводные широкополосные системы, персональные цифровые устройства (PDA), переносные или настольные компьютеры, планшетные компьютеры, устройства для чтения электронных книг, цифровые камеры, цифровые записывающие устройства, цифровые мультимедийные проигрыватели, устройства видеоигр, консоли для видеоигр, сотовые или спутниковые радиотелефоны, так называемые "смартфоны", устройства видеоконференц-связи, устройства потоковой передачи видео и т.п. Цифровые видеоустройства реализуют такие технологии сжатия видео, как технологии сжатия видео, описанные в стандартах, заданных посредством разрабатываемых в настоящее время стандартов MPEG-2, MPEG-4, ITU-T H.263, ITU-T H.264/MPEG-4, часть 10, усовершенствованное кодирование видео (AVC), стандарта высокоэффективного кодирования видео (HEVC), и расширений таких стандартов. Видеоустройства могут передавать, принимать, кодировать, декодировать и/или сохранять цифровую видеоинформацию более эффективно посредством реализации таких технологий сжатия видео.

[0004] Технологии сжатия видео выполняют пространственное (внутрикадровое) прогнозирование и/или временное (межкадровое) прогнозирование для того, чтобы уменьшать или удалять избыточность, внутренне присущую в видеопоследовательностях. Для кодирования видео на основе блоков видеослайс (т.е. видеокادر или часть видеокадра) может быть сегментирован на видеоблоки. Видеоблоки в внутренне-кодированном (I-) слайсе изображения кодируются с использованием пространственного прогнозирования относительно опорных выборок в соседних блоках в идентичном изображении. Видеоблоки во внешне-кодированном (P- или B-) слайсе изображения могут использовать пространственное прогнозирование относительно опорных выборок в соседних блоках в идентичном изображении или временное прогнозирование относительно опорных выборок в других опорных изображениях. Изображения могут упоминаться как кадры, и опорные изображения могут упоминаться как опорные кадры.

[0005] Пространственное или временное прогнозирование приводит в результате к прогнозному блоку для блока, который должен быть кодирован. Остаточные данные представляют пиксельные разности между исходным блоком, который должен быть кодирован, и прогнозным блоком. Внешне-кодированный блок кодируется согласно вектору движения, который указывает на блок опорных выборок, формирующих прогнозный блок, и остаточные данные указывают разность между кодированным блоком и прогнозным блоком. Внутренне кодированный блок кодируется согласно режиму внутреннего кодирования и остаточным данным. Для дополнительного сжатия остаточные данные могут быть преобразованы из пиксельной области в область преобразования, приводя к остаточным коэффициентам, которые затем могут быть квантованы. Квантованные коэффициенты, первоначально размещаемые в двумерном массиве, могут сканироваться для того, чтобы формировать одномерный вектор

коэффициентов, и может применяться энтропийное кодирование с тем, чтобы достигать еще большего сжатия.

[0006] Битовый поток многовидового кодирования может формироваться посредством кодирования видов, например, с нескольких перспектив. Разработаны некоторые трехмерные видеостандарты, которые используют аспекты многовидового кодирования. Например, различные виды могут передавать виды для левого и правого глаза, чтобы поддерживать трехмерное видео. Альтернативно, некоторые процессы кодирования трехмерного видео могут применять так называемое многовидовое кодирование с учетом глубины. При многовидовом кодировании с учетом глубины битовый поток трехмерного видео может содержать не только компоненты видов текстуры, но также и компоненты видов глубины. Например, каждый вид может содержать один компонент вида текстуры и один компонент вида глубины.

### СУЩНОСТЬ ИЗОБРЕТЕНИЯ

[0007] В общем, это раскрытие сущности описывает передачу в служебных сигналах параметров гипотетического эталонного декодера (HRD) и вложение сообщений с дополнительной улучшающей информацией (SEI) при кодировании видео. Более конкретно, видеокодер может включать в масштабируемое вкладывающее SEI-сообщение, инкапсулированное посредством единицы уровня сетевой абстракции (NAL) SEI, элемент синтаксиса, который указывает то, является или нет вложенное SEI-сообщение, инкапсулированное посредством масштабируемого вкладывающего SEI-сообщения, применимым к битовому субпотoku по умолчанию. Битовый субпоток по умолчанию является представлением рабочей точки для рабочей точки, заданной посредством идентификатора слоя, точно определяемого в заголовке NAL-единицы для SEI NAL-единицы, и временного идентификатора, точно определяемого в заголовке NAL-единицы. Кроме того, устройство может определять, по меньшей мере, частично на основе элемента синтаксиса в масштабируемом вкладывающем SEI-сообщении то, является или нет вложенное SEI-сообщение, инкапсулированное посредством масштабируемого вкладывающего SEI-сообщения, применимым к битовому субпотoku по умолчанию. Когда вложенное SEI-сообщение является применимым к битовому субпотoku по умолчанию, устройство может использовать вложенное SEI-сообщение при операции над битовым субпотком по умолчанию.

[0008] В одном примере, это раскрытие сущности описывает способ обработки видеоданных. Способ содержит определение, по меньшей мере, частично на основе элемента синтаксиса в масштабируемом вкладывающем SEI-сообщении, инкапсулированном посредством SEI NAL-единицы, того, является или нет вложенное SEI-сообщение, инкапсулированное посредством масштабируемого вкладывающего SEI-сообщения, применимым к битовому субпотoku по умолчанию из кодированного битового потока видео. Битовый субпоток по умолчанию является представлением рабочей точки для рабочей точки, заданной посредством идентификатора слоя, точно определяемого в заголовке NAL-единицы для SEI NAL-единицы, и временного идентификатора, точно определяемого в заголовке NAL-единицы. Помимо этого, способ содержит, когда вложенное SEI-сообщение является применимым к битовому субпотoku по умолчанию, использование вложенного SEI-сообщения при операции над битовым субпотком по умолчанию.

[0009] В другом примере, это раскрытие сущности описывает устройство, содержащее один или более процессоров, сконфигурированных с возможностью определять, по меньшей мере, частично на основе элемента синтаксиса в масштабируемом вкладывающем SEI-сообщении, инкапсулированном посредством SEI NAL-единицы,

то, является или нет вложенное SEI-сообщение, инкапсулированное посредством масштабируемого вкладывающего SEI-сообщения, применимым к битовому субпотoku по умолчанию из кодированного битового потока видео. Битовый субпоток по умолчанию является представлением рабочей точки для рабочей точки, заданной посредством идентификатора слоя, точно определяемого в заголовке NAL-единицы для SEI NAL-единицы, и временного идентификатора, точно определяемого в заголовке NAL-единицы. Один или более процессоров имеют такую конфигурацию, в которой, когда вложенное SEI-сообщение является применимым к битовому субпотoku по умолчанию, один или более процессоров используют вложенное SEI-сообщение при операции над битовым субпотком по умолчанию.

[0010] В другом примере, это раскрытие сущности описывает устройство, содержащее средство для определения, по меньшей мере, частично на основе элемента синтаксиса в масштабируемом вкладывающем SEI-сообщении, инкапсулированное посредством SEI NAL-единицы, того, является или нет вложенное SEI-сообщение, инкапсулированное посредством масштабируемого вкладывающего SEI-сообщения, применимым к битовому субпотoku по умолчанию из кодированного битового потока видео. Битовый субпоток по умолчанию является представлением рабочей точки для рабочей точки, заданной посредством идентификатора слоя, точно определяемого в заголовке NAL-единицы для SEI NAL-единицы, и временного идентификатора, точно определяемого в заголовке NAL-единицы. Устройство также содержит средство для использования, когда вложенное SEI-сообщение является применимым к битовому субпотoku по умолчанию, вложенного SEI-сообщения при операции над битовым субпотком по умолчанию.

[0011] В другом примере, это раскрытие сущности описывает считываемый компьютером носитель данных, который сохраняет инструкции, которые при выполнении посредством одного или более процессоров устройства конфигурируют устройство с возможностью определять, по меньшей мере, частично на основе элемента синтаксиса в масштабируемом вкладывающем SEI-сообщении, инкапсулированное посредством SEI NAL-единицы, то, является или нет вложенное SEI-сообщение, инкапсулированное посредством масштабируемого вкладывающего SEI-сообщения, применимым к битовому субпотoku по умолчанию из кодированного битового потока видео. Битовый субпоток по умолчанию является представлением рабочей точки для рабочей точки, заданной посредством идентификатора слоя, точно определяемого в заголовке NAL-единицы для SEI NAL-единицы, и временного идентификатора, точно определяемого в заголовке NAL-единицы. Инструкции при выполнении конфигурируют устройство с возможностью того, что когда вложенное SEI-сообщение является применимым к битовому субпотoku по умолчанию, устройство использует вложенное SEI-сообщение при операции над битовым субпотком по умолчанию.

[0012] В другом примере, это раскрытие сущности описывает способ кодирования видеоданных. Способ содержит включение в масштабируемое вкладывающее SEI-сообщение, инкапсулированное посредством SEI NAL-единицы, элемента синтаксиса, который указывает то, является или нет вложенное SEI-сообщение, инкапсулированное посредством масштабируемого вкладывающего SEI-сообщения, применимым к битовому субпотoku по умолчанию из кодированного битового потока видео. Битовый субпоток по умолчанию является представлением рабочей точки для рабочей точки, заданной посредством идентификатора слоя, точно определяемого в заголовке NAL-единицы для SEI NAL-единицы, и временного идентификатора, точно определяемого в заголовке NAL-единицы. Способ также содержит передачу в служебных сигналах

масштабируемого вкладывающего SEI-сообщения в кодированном битовом потоке видео.

[0013] В другом примере, это раскрытие сущности описывает устройство кодирования видео, содержащее один или более процессоров, сконфигурированных с возможностью включать в масштабируемое вкладывающее SEI-сообщение, инкапсулированное посредством SEI NAL-единицы, элемент синтаксиса, который указывает то, является ли вложенное SEI-сообщение, инкапсулированное посредством масштабируемого вкладывающего SEI-сообщения, применимым к битовому субпотoku по умолчанию из кодированного битового потока видео. Битовый субпоток по умолчанию является представлением рабочей точки для рабочей точки, заданной посредством идентификатора слоя, точно определяемого в заголовке NAL-единицы для SEI NAL-единицы, и временного идентификатора, точно определяемого в заголовке NAL-единицы. Один или более процессоров также сконфигурированы с возможностью сигнализировать масштабируемое вкладывающее SEI-сообщение в кодированном битовом потоке видео.

[0014] В другом примере, это раскрытие сущности описывает устройство кодирования видео, содержащее средство для включения в масштабируемое вкладывающее SEI-сообщение, инкапсулированное посредством SEI NAL-единицы, элемента синтаксиса, который указывает то, является ли вложенное SEI-сообщение, инкапсулированное посредством масштабируемого вкладывающего SEI-сообщения, применимым к битовому субпотoku по умолчанию из кодированного битового потока видео. Битовый субпоток по умолчанию является представлением рабочей точки для рабочей точки, заданной посредством идентификатора слоя, точно определяемого в заголовке NAL-единицы для SEI NAL-единицы, и временного идентификатора, точно определяемого в заголовке NAL-единицы. Устройство кодирования видео также содержит средство для сигнализации масштабируемого вкладывающего SEI-сообщения в кодированном битовом потоке видео.

[0015] В другом примере, это раскрытие сущности описывает считываемый компьютером носитель данных, который сохраняет инструкции, которые при выполнении посредством устройства кодирования видео, конфигурируют устройство кодирования видео с возможностью включать, в масштабируемое вкладывающее SEI-сообщение, инкапсулированное посредством SEI NAL-единицы, элемент синтаксиса, который указывает то, является ли вложенное SEI-сообщение, инкапсулированное посредством масштабируемого вкладывающего SEI-сообщения, применимым к битовому субпотoku по умолчанию из кодированного битового потока видео. Битовый субпоток по умолчанию является представлением рабочей точки для рабочей точки, заданной посредством идентификатора слоя, точно определяемого в заголовке NAL-единицы для SEI NAL-единицы, и временного идентификатора, точно определяемого в заголовке NAL-единицы. Инструкции при выполнении также конфигурируют устройство кодирования видео с возможностью сигнализировать масштабируемое вкладывающее SEI-сообщение в кодированном битовом потоке видео.

[0016] Подробности одного или более примеров данного раскрытия сущности изложены на прилагаемых чертежах и в нижеприведенном описании. Другие признаки, цели и преимущества должны становиться очевидными из описания, чертежей и формулы изобретения.

#### КРАТКОЕ ОПИСАНИЕ ЧЕРТЕЖЕЙ

[0017] Фиг. 1 является блок-схемой, иллюстрирующей примерную систему кодирования видео, которая может использовать технологии, описанные в этом раскрытии сущности.

[0018] Фиг. 2 является блок-схемой, иллюстрирующей примерный видеокодер, который может реализовывать технологии, описанные в этом раскрытии сущности.

[0019] Фиг. 3 является блок-схемой, иллюстрирующей примерный видеодекодер, который может реализовывать технологии, описанные в этом раскрытии сущности.

5 [0020] Фиг. 4 является блок-схемой последовательности операций способа, иллюстрирующей примерную работу видеокодера, в соответствии с одной или более технологий этого раскрытия сущности.

[0021] Фиг. 5 является блок-схемой последовательности операций способа, иллюстрирующей примерную работу устройства, в соответствии с одной или более  
10 технологий этого раскрытия сущности.

[0022] Фиг. 6 является блок-схемой последовательности операций способа, иллюстрирующей примерную работу видеокодера, в соответствии с одной или более технологий этого раскрытия сущности.

[0023] Фиг. 7 является блок-схемой последовательности операций способа, иллюстрирующей примерную работу устройства, в соответствии с одной или более  
15 технологий этого раскрытия сущности.

[0024] Фиг. 8 является блок-схемой последовательности операций способа, иллюстрирующей примерную работу видеокодера, в соответствии с одной или более технологий этого раскрытия сущности.

20 [0025] Фиг. 9 является блок-схемой последовательности операций способа, иллюстрирующей примерную работу устройства, в соответствии с одной или более технологий этого раскрытия сущности.

#### ПОДРОБНОЕ ОПИСАНИЕ ИЗОБРЕТЕНИЯ

[0026] Видеокодер может формировать битовый поток, который включает в себя  
25 кодированные видеоданные. Поскольку битовый поток включает в себя кодированные видеоданные, битовый поток может упоминаться в данном документе как кодированный битовый поток видео. Битовый поток может содержать последовательность единиц уровня сетевой абстракции (NAL). NAL-единицы могут включать в себя NAL-единицы слоя кодирования видео (VCL) и не-VCL NAL-единицы. VCL NAL-единицы могут  
30 включать в себя кодированные слайсы изображений. Не-VCL NAL-единица может включать в себя набор параметров видео (VPS), набор параметров последовательности (SPS), набор параметров изображения (PPS), дополнительную улучшающую информацию (SEI) или другие типы данных. VPS представляет собой синтаксическую структуру, которая может содержать элементы синтаксиса, которые применяются к  
35 нулю или более из всех кодированных видеопоследовательностей. SPS представляет собой синтаксическую структуру, которая может содержать элементы синтаксиса, которые применяются к нулю или более из всех кодированных видеопоследовательностей. Один VPS может быть применимым к нескольким SPS. PPS представляет собой синтаксическую структуру, которая может содержать элементы  
40 синтаксиса, которые применяются к нулю или более из всех кодированных изображений. Один SPS может быть применимым к нескольким PPS.

[0027] Устройство, такое как устройство для работы в сети доставки контента (CDN), мультимедийно-ориентированный сетевой элемент (MANE), видеокодер или  
45 видеодекодер, может извлекать битовый субпоток из битового потока. Устройство может выполнять процесс извлечения битовых субпоточков посредством удаления определенных NAL-единиц из битового потока. Результирующий битовый субпоток включает в себя оставшиеся, неудаленные NAL-единицы битового потока. В некоторых примерах, видеоданные, декодированные из битового субпоточка, могут иметь более



низкую частоту кадров и/или могут представлять меньше видов по сравнению с исходным битовым потоком.

[0028] Стандарты кодирования видео могут включать в себя различные признаки для того, чтобы поддерживать процесс извлечения битовых субпотоков. Например, видеоданные битового потока могут быть разделены на набор слоев. Для каждого из слоев, данные в нижнем слое могут декодироваться независимо от данных в любом верхнем слое. NAL-единицы только инкапсулируют данные одного слоя. Таким образом, NAL-единицы, инкапсулирующие данные наибольшего оставшегося слоя битового потока, могут удаляться из битового потока без влияния на декодируемость данных в оставшихся слоях битового потока. При масштабируемом кодировании видео (SVC), верхние слои могут включать в себя улучшающие данные, которые повышают качество изображений в нижних слоях (масштабируемость на основе качества), укрупняют пространственный формат изображений в нижних слоях (пространственную масштабируемость) или увеличивают временную скорость изображений в нижних слоях (временную масштабируемость). При многовидовом кодировании (MVC) и кодировании трехмерного видео (3DV), верхние слои могут включать в себя дополнительные виды.

[0029] Каждая NAL-единица может включать в себя заголовок и рабочие данные. Заголовки NAL-единиц могут включать в себя элементы `nuh_reserved_zero_6bits` синтаксиса. Элемент `nuh_reserved_zero_6bits` синтаксиса NAL-единицы равен 0, если NAL-единица связана с базовым слоем при MVC, 3DV-кодировании или SVC. Данные в базовом слое битового потока могут декодироваться независимо от данных в любом другом слое битового потока. Если NAL-единица не связана с базовым слоем при MVC, 3DV или SVC, элемент `nuh_reserved_zero_6bits` синтаксиса может иметь другие ненулевые значения. В частности, если NAL-единица не связана с базовым слоем при MVC, 3DV или SVC, элемент `nuh_reserved_zero_6bits` синтаксиса NAL-единицы указывает идентификатор слоя, который идентифицирует слой, ассоциированный с NAL-единицей.

[0030] Кроме того, некоторые изображения в слое могут декодироваться независимо от других изображений в идентичном слое. Таким образом, NAL-единицы, инкапсулирующие данные определенных изображений слоя, могут удаляться из битового потока без влияния на декодируемость других изображений в слое. Например, изображения с четными значениями номера в последовательности изображений (POC) могут быть декодируемыми независимо от изображений с нечетными POC-значениями. Удаление NAL-единиц, инкапсулирующих данные таких изображений, позволяет уменьшать частоту кадров битового потока. Поднабор изображений в слое, который может декодироваться независимо от других изображений в слое, может упоминаться в данном документе как "подслой" или "временной подслой".

[0031] NAL-единицы могут включать в себя элементы `nuh_temporal_id_plus1` синтаксиса. Элемент `nuh_temporal_id_plus1` синтаксиса NAL-единицы может указывать временной идентификатор NAL-единицы. Если временной идентификатор первой NAL-единицы меньше временного идентификатора второй NAL-единицы, данные, инкапсулированные посредством первой NAL-единицы, могут декодироваться независимо от данных, инкапсулированных посредством второй NAL-единицы.

[0032] Рабочие точки битового потока ассоциированы с набором идентификаторов слоев (т.е. набором значений `nuh_reserved_zero_6bits`) и временным идентификатором. Набор идентификаторов слоев может обозначаться как `OpLayerIdSet`, и временной идентификатор может обозначаться как `TemporalID`. Если идентификатор слоя NAL-единицы находится в наборе идентификаторов слоев для рабочей точки, и временной идентификатор NAL-единицы меньше или равен временному идентификатору рабочей

точки, NAL-единица ассоциирована с рабочей точкой. Представление рабочей точки представляет собой поднабор битовых потоков (т.е. битовый субпоток), который ассоциирован с рабочей точкой. Представление рабочей точки для рабочей точки может включать в себя каждую NAL-единицу, которая ассоциирована с рабочей точкой.

- 5 Представление рабочей точки не включает в себя VCL NAL-единицы, которые не ассоциированы с рабочей точкой.

[0033] Внешний источник может указывать набор идентификаторов целевого слоя для рабочей точки. Например, устройство для работы в сети доставки контента (CDN) может указывать набор идентификаторов целевого слоя. В этом примере, CDN-  
10 устройство может использовать набор идентификаторов целевого слоя для того, чтобы идентифицировать рабочую точку. CDN-устройство затем может извлекать представление рабочей точки для рабочей точки и передавать представление рабочей точки, вместо исходного битового потока, в клиентское устройство. Извлечение и передача представления рабочей точки в клиентское устройство позволяют уменьшать  
15 скорость передачи битов для битового потока.

[0034] Кроме того, стандарты кодирования видео указывают модели буферизации видео. Модель буферизации видео также может упоминаться в качестве "гипотетического эталонного декодера" или "HRD". HRD описывает то, как данные должны быть буферизованы для декодирования, и как декодированные данные буферизуются для  
20 вывода. Например, HRD описывает работу буфера кодированных изображений (CPB) и буфера декодированных изображений (DPB) в видеодекодере. CPB представляет собой буфер "первый на входе – первый на выходе", содержащий единицы доступа в порядке декодирования, указываемом посредством HRD. DPB представляет собой буфер, хранящий декодированные изображения для ссылки, переупорядочение вывода или  
25 задержку вывода, указываемую посредством HRD.

[0035] Видеокодер может сигнализировать набор HRD-параметров. HRD-параметры управляют различными аспектами HRD. HRD-параметры могут включать в себя начальную задержку удаления из CPB, CPB-размер, скорость передачи битов, начальную задержку DPB-вывода и DPB-размер. Эти HRD-параметры могут быть кодированы в  
30 синтаксической структуре `hrd_parameters()`, указываемой в VPS и/или SPS. HRD-параметры также могут указываться в SEI-сообщениях по периоду буферизации или в SEI-сообщениях по синхронизации изображений.

[0036] Как пояснено выше, представление рабочей точки может иметь другую частоту кадров и/или скорость передачи битов по сравнению с исходным битовым потоком.  
35 Это обусловлено тем, что представление рабочей точки может не включать в себя некоторые изображения и/или некоторые данные исходного битового потока. Следовательно, если видеодекодер должен удалять данные из CPB и/или DPB на конкретной скорости при обработке исходного битового потока, и если видеодекодер должен удалять данные из CPB и/или DPB на идентичной скорости при обработке  
40 представления рабочей точки, видеодекодер может удалять слишком много или слишком мало данных из CPB и/или DPB. Соответственно, видеокодер может сигнализировать различные наборы HRD-параметров для различных рабочих точек. В появляющемся стандарте высокоэффективного кодирования видео (HEVC) видеокодер может сигнализировать наборы HRD-параметров в VPS, либо видеокодер может  
45 сигнализировать наборы HRD-параметров в SPS.

[0037] Набор HRD-параметров необязательно включает в себя набор информации, которая является общей для всех временных подслоев. Временной подслоем представляет собой временной масштабируемый слой временного масштабируемого битового потока,

состоящего из VCL NAL-единиц с конкретным временным идентификатором и ассоциированных не-VCL NAL-единиц. В дополнение к набору общей информации, наборы HRD-параметров могут включать в себя набор элементов синтаксиса, которые являются конкретными для отдельных временных подслоев. Поскольку набор общей информации является общим для нескольких наборов из HRD-параметров, может быть 5 необязательным сигнализировать набор общей информации в нескольких наборах HRD-параметров. В некоторых проектах для HEVC общая информация может присутствовать в наборе HRD-параметров, когда набор HRD-параметров представляет собой первый набор HRD-параметров в VPS, или общая информация может 10 присутствовать в наборе HRD-параметров, когда набор HRD-параметров ассоциирован с первой рабочей точкой.

[0038] Тем не менее, когда предусмотрено несколько наборов из HRD-параметров в VPS, может быть желательным иметь несколько различных наборов общей информации для наборов HRD-параметров. Это может быть, в частности, истинным, 15 когда имеются большие числа синтаксических структур HRD-параметров в VPS. Таким образом, может быть желательным иметь наборы общей информации в синтаксических структурах HRD-параметров, отличных от первой синтаксической структуры HRD-параметров.

[0039] Технологии этого раскрытия сущности предоставляют схему для того, чтобы 20 предоставлять возможность явной сигнализации общей информации синтаксических структур HRD-параметров для любой синтаксической структуры HRD-параметров. Другими словами, технологии этого раскрытия сущности могут предоставлять возможность явной сигнализации информации, которая является общей для всех подслоев для любой синтаксической структуры `hrd_parameters()`. Это может повышать 25 эффективность кодирования.

[0040] Таким образом, в соответствии с одной или более технологий этого раскрытия сущности, устройство, такое как видеodeкодер или другое устройство, может определять, по меньшей мере, частично на основе элемента синтаксиса в VPS, который включает в себя множество синтаксических структур HRD-параметров, то, включает или нет 30 конкретная синтаксическая структура HRD-параметров в VPS в себя набор HRD-параметров, которые являются общими для каждого подслоя битового потока. Устройство может декодировать элемент синтаксиса из VPS. Одна или более синтаксических структур HRD-параметров могут возникать в порядке кодирования в VPS перед конкретной синтаксической структурой HRD-параметров. В ответ на 35 определение того, что конкретная синтаксическая структура HRD-параметров включает в себя набор HRD-параметров, которые являются общими для каждого подслоя битового потока, устройство может выполнять операцию с использованием конкретной синтаксической структуры HRD-параметров, включающей в себя набор HRD-параметров, которые являются общими для каждого подслоя битового потока.

[0041] Кроме того, видеокодер может формировать масштабируемые вкладывающиеся SEI-сообщения. Масштабируемое вкладывающееся SEI-сообщение содержит одно или более SEI-сообщений. SEI-сообщения, вложенные в масштабируемое вкладывающееся SEI-сообщение, могут включать в себя HRD-параметры или другую информацию, ассоциированную с рабочей точкой. Некоторые проекты для HEVC не предоставляют 45 возможность применения одного SEI-сообщения к нескольким рабочим точкам. Это может снижать эффективность по скорости передачи битов, поскольку это может приводить к тому, что видеокодер сигнализирует несколько SEI-сообщений с идентичной информацией. Соответственно, технологии этого раскрытия сущности могут

предоставлять возможность применения одного SEI-сообщения к нескольким рабочим точкам. Например, масштабируемое вкладывающее SEI-сообщение может включать в себя элементы синтаксиса, которые указывают несколько рабочих точек, применимых к SEI-сообщениям, вложенным в масштабируемом вкладывающем SEI-сообщении.

5 [0042] Помимо этого, SEI NAL-единицы, аналогично другим типам NAL-единиц, включают в себя заголовок NAL-единицы и тело NAL-единицы. Тело NAL-единицы для SEI NAL-единицы может включать в себя SEI-сообщение, к примеру, масштабируемое вкладывающее SEI-сообщение или другой тип SEI-сообщения. Аналогично другим NAL-единицам, заголовок NAL-единицы для SEI NAL-единицы может включать в себя  
10 элемент `nuh_reserved_zero_6bits` синтаксиса и элемент `nuh_temporal_id_plus1` синтаксиса. Тем не менее, в некоторых проектах для HEVC, элемент `nuh_reserved_zero_6bits` синтаксиса и/или элемент `nuh_temporal_id_plus1` синтаксиса заголовка NAL-единицы для SEI NAL-единицы не используются для того, чтобы определять рабочую точку, применимую к SEI-сообщению (или SEI-сообщениям), инкапсулированным посредством  
15 SEI NAL-единицы. Тем не менее, эти элементы синтаксиса заголовка SEI NAL-единицы могут быть многократно использованы для того, чтобы сокращать число сигнализируемых битов. Следовательно, в соответствии с технологиями этого раскрытия сущности, элемент синтаксиса может сигнализироваться в масштабируемом вкладывающем SEI-сообщении, чтобы указывать то, представляет собой или нет рабочая  
20 точка, применимая к вложенным SEI-сообщениям в SEI NAL-единице, рабочую точку, указываемую посредством идентификационной информации слоя, в заголовке NAL-единицы для SEI NAL-единицы. Идентификационная информация слоя в заголовке NAL-единицы для SEI NAL-единицы может включать в себя значение `nuh_reserved_zero_6bits` и значение `nuh_temporal_id_plus1` заголовка NAL-единицы.

25 [0043] Фиг. 1 является блок-схемой, иллюстрирующей примерную систему 10 кодирования видео, которая может использовать технологии этого раскрытия сущности. При использовании в данном документе, термин "видеокодер" относится обобщенно к видеокодерам и декодерам. В этом раскрытии сущности, термины "кодирование видео" или "кодирование" могут относиться обобщенно к кодированию видео или  
30 декодированию видео.

[0044] Как показано на фиг. 1, система 10 кодирования видео включает в себя устройство-источник 12 и устройство-адресат 14. Устройство-источник 12 формирует кодированные видеоданные. Соответственно, устройство-источник 12 может упоминаться в качестве устройства кодирования видео или устройства кодирования  
35 видео. Устройство-адресат 14 может декодировать кодированные видеоданные, сформированные посредством устройства-источника 12. Соответственно, устройство-адресат 14 может упоминаться в качестве устройства декодирования видео или устройства декодирования видео. Устройство-источник 12 и устройство-адресат 14 могут быть примерами устройств кодирования видео или устройств кодирования видео.

40 [0045] Устройство-источник 12 и устройство-адресат 14 могут содержать широкий диапазон устройств, включающих в себя настольные компьютеры, мобильные вычислительные устройства, ноутбуки (например, переносные компьютеры), планшетные компьютеры, абонентские приставки, телефонные трубки, к примеру, так называемые смартфоны, телевизионные приемники, камеры, устройства отображения,  
45 цифровые мультимедийные проигрыватели, консоли для видеоигр, встроенные в автомобиль компьютеры и т.п.

[0046] Устройство-адресат 14 может принимать кодированные видеоданные из устройства-источника 12 через канал 16. Канал 16 может содержать одну или более

сред или устройств, допускающих перемещение кодированных видеоданных из устройства-источника 12 в устройство-адресат 14. В одном примере, канал 16 может содержать одну или более сред связи, которые предоставляют возможность устройству-источнику 12 передавать кодированные видеоданные непосредственно в устройство-адресат 14 в реальном времени. В этом примере, устройство-источник 12 может модулировать кодированные видеоданные согласно стандарту связи, такому как протокол беспроводной связи, и может передавать модулированные видеоданные в устройство-адресат 14. Одна или более сред связи могут включать в себя беспроводные среды связи и/или проводные среды связи, к примеру, радиочастотный (RF) спектр или одну или более физических линий передачи. Одна или более сред связи могут составлять часть сети с коммутацией пакетов, такой как локальная вычислительная сеть, глобальная вычислительная сеть или глобальная сеть (например, Интернет). Один или более сред связи могут включать в себя маршрутизаторы, коммутаторы, базовые станции или другое оборудование, которое упрощает передачу из устройства-источника 12 в устройство-адресат 14.

[0047] В другом примере, канал 16 может включать в себя носитель хранения данных, который сохраняет кодированные видеоданные, сформированные посредством устройства-источника 12. В этом примере, устройство-адресат 14 может осуществлять доступ к носителю хранения данных, к примеру, через доступ к диску или доступ по карте. Носитель хранения данных может включать в себя множество локально доступных носителей хранения данных, таких как Blu-Ray-диски, DVD, CD-ROM, флэш-память или другие подходящие цифровые носители данных для сохранения кодированных видеоданных.

[0048] В дополнительном примере, канал 16 может включать в себя файловый сервер или другое промежуточное устройство хранения данных, которое сохраняет кодированные видеоданные, сформированные посредством устройства-источника 12. В этом примере, устройство-адресат 14 может осуществлять доступ к кодированным видеоданным, сохраненным на файловом сервере или другом промежуточном устройстве хранения данных через потоковую передачу или загрузку. Файловый сервер может представлять собой тип сервера, допускающего сохранение кодированных видеоданных и передачу кодированных видеоданных в устройство-адресат 14. Примерные файловые серверы включают в себя веб-серверы (например, для веб-узла), серверы по протоколу передачи файлов (FTP), устройства по протоколу системы хранения данных с подключением по сети (NAS) и локальные накопители на дисках.

[0049] Устройство-адресат 14 может осуществлять доступ к кодированным видеоданным через стандартное подключение для передачи данных, к примеру, Интернет-подключение. Примерные типы подключений для передачи данных могут включать в себя беспроводные каналы (например, Wi-Fi-подключения), проводные подключения (например, DSL, кабельный модем и т.д.) или комбинации вышеозначенного, которые являются подходящими для осуществления доступа к кодированным видеоданным, сохраненным на файловом сервере. Передача кодированных видеоданных из файлового сервера может представлять собой потоковую передачу, передачу на основе загрузки или комбинацию вышеозначенного.

[0050] Технологии этого раскрытия сущности не ограничены приложениями или настройками беспроводной связи. Технологии могут применяться к кодированию видео в поддержку множества мультимедийных приложений, таких как телевизионные широкоэмитательные передачи по радиоинтерфейсу, кабельные телевизионные передачи, спутниковые телевизионные передачи, потоковые передачи видео, например, через

Интернет, кодирование видеоданных для хранения на носителе хранения данных, декодирование видеоданных, сохраненных на носителе хранения данных, или другие приложения. В некоторых примерах, система 10 кодирования видео может быть сконфигурирована с возможностью поддерживать одностороннюю или двустороннюю передачу видео, чтобы поддерживать такие приложения, как потоковая передача видео, воспроизведение видео, широковещательная передача видео и/или видеотелефония.

[0051] Фиг. 1 является просто примером, и технологии этого раскрытия сущности могут применяться к настройкам кодирования видео (например, кодирования видео или декодирования видео), которые не обязательно включают в себя передачу данных между устройствами кодирования видео и декодирования. В других примерах, данные извлекаются из локального запоминающего устройства, передаются в потоковом режиме по сети и т.п. Устройство кодирования видео может кодировать и сохранять данные в запоминающем устройстве, и/или устройство декодирования видео может извлекать и декодировать данные из запоминающего устройства. Во многих примерах, кодирование и декодирование выполняется посредством устройств, которые не обмениваются данными друг с другом, а просто кодируют данные в запоминающее устройство и/или извлекают и декодируют данные из запоминающего устройства.

[0052] В примере по фиг. 1, устройство-источник 12 включает в себя видеоисточник 18, видеокодер 20 и интерфейс 22 вывода. В некоторых примерах, интерфейс 22 вывода может включать в себя модулятор/демодулятор (модем) и/или передающее устройство. Видеоисточник 18 может включать в себя устройство видеозахвата, например, видеокамеру, видеоархив, содержащий ранее захваченные видеоданные, интерфейс прямых видеотрансляций, чтобы принимать видеоданные от поставщика видеоконтента, и/или компьютерную графическую систему для формирования видеоданных либо комбинацию таких источников видеоданных.

[0053] Видеокодер 20 может кодировать видеоданные из видеоисточника 18. В некоторых примерах, устройство-источник 12 непосредственно передает кодированные видеоданные в устройство-адресат 14 через интерфейс 22 вывода. В других примерах, кодированные видеоданные также могут быть сохранены на носителе хранения данных или файловый сервер для последующего доступа посредством устройства-адресата 14 для декодирования и/или воспроизведения.

[0054] В примере по фиг. 1, устройство-адресат 14 включает в себя интерфейс 28 ввода, видеodeкодер 30 и устройство 32 отображения. В некоторых примерах, интерфейс 28 ввода включает в себя приемное устройство и/или модем. Интерфейс 28 ввода может принимать кодированные видеоданные по каналу 16. Устройство 32 отображения может быть интегрировано или может быть внешним для устройства-адресата 14. В общем, устройство 32 отображения отображает декодированные видеоданные. Устройство 32 отображения может содержать множество устройств отображения, таких как жидкокристаллический дисплей (LCD), плазменный дисплей, дисплей на органических светодиодах (OLED) или другой тип устройства отображения.

[0055] Видеокодер 20 и видеodeкодер 30 могут быть реализованы как любая из множества надлежащих схем, к примеру, один или более микропроцессоров, процессоров цифровых сигналов (DSP), специализированных интегральных схем (ASIC), программируемых пользователем вентильных матриц (FPGA), дискретная логика, аппаратные средства либо любые комбинации вышеозначенного. Если технологии реализуются частично в программном обеспечении, устройство может сохранять инструкции для программного обеспечения на подходящем энергонезависимом считываемом компьютере носителе данных и может выполнять инструкции в

аппаратных средствах с использованием одного или более процессоров, чтобы осуществлять технологии этого раскрытия сущности. Любое из вышеозначенного (включающее в себя аппаратные средства, программное обеспечение, комбинацию аппаратных средств и программного обеспечения и т.д.) может рассматриваться в качестве одного или более процессоров. Каждый из видеокодера 20 и видеодекодера 30 может быть включен в один или более кодеров или декодеров, любой из которых может быть интегрирован как часть комбинированного кодера/декодера (кодека) в соответствующем устройстве.

[0056] Это раскрытие сущности может, в общем, означать видеокодер 20, "сигнализирующий" определенную информацию в другое устройство, такое как видеодекодер 30. Термин "передача служебных сигналов" может, в общем, означать передачу элементов синтаксиса и/или других данных, используемых для того, чтобы декодировать сжатые видеоданные. Эта связь может осуществляться в реальном или практически в реальном времени. Альтернативно, эта связь может осуществляться в промежутке времени, к примеру, может осуществляться при сохранении элементов синтаксиса на считываемом компьютером носителе данных в кодированном битовом потоке во время кодирования, которые затем могут извлекаться посредством устройства декодирования видео в любое время после сохранения на этом носителе.

[0057] В некоторых примерах, видеокодер 20 и видеодекодер 30 работают согласно такому стандарту сжатия видео, как ISO/IEC MPEG-4 Visual и ITU-T H.264 (также известный как ISO/IEC MPEG-4 AVC), в том числе и согласно его расширению масштабируемого кодирования видео (SVC), расширению кодирования многовидового видео (MVC) и 3DV-расширению на основе MVC. В некоторых случаях, любой битовый поток, соответствующий 3DV на основе MVC, всегда содержит битовый субпоток, который является совместимым с профилем MVC, например, с основным стереопрофилем. Кроме того, прилагаются усилия для того, чтобы формировать расширение кодирования трехмерного видео (3DV) в H.264/AVC, а именно, 3DV на основе AVC. В других примерах, видеокодер 20 и видеодекодер 30 могут работать согласно ITU-T H.261, ISO/IEC MPEG-1 Visual, ITU-T H.262 или ISO/IEC MPEG-2 Visual и ITU-T H.264, ISO/IEC Visual.

[0058] В других примерах, видеокодер 20 и видеодекодер 30 могут работать согласно стандарту высокоэффективного кодирования видео (HEVC), в настоящее время разрабатываемому посредством Объединенной группы для совместной работы над видеостандартами (JCT-VC) Экспертной группы в области кодирования видео (VCEG) ITU-T и Экспертной группы по киноизображению (MPEG) ISO/IEC. Проект будущего HEVC-стандарта, называемый "рабочим проектом 8 HEVC", описывается в документе Bross и др. "High Efficiency Video Coding (HEVC) text specification draft 8", Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11, 10<sup>th</sup> Meeting, Стокгольм, Швеция, июль 2012 года, который с 13 июня 2013 доступен по адресу [http://phenix.int-evry.fr/jct/doc\\_end\\_user/documents/10\\_Sweden/wg11/JCTVC-J1003-v8.zip](http://phenix.int-evry.fr/jct/doc_end_user/documents/10_Sweden/wg11/JCTVC-J1003-v8.zip). Другой проект будущего HEVC-стандарта, называемый "рабочим проектом 9 HEVC", описывается в документе Bross и др. "High Efficiency Video Coding (HEVC) text specification draft 9", Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11, 11<sup>th</sup> Meeting, Шанхай, Китай, октябрь 2012 года, который с 13 июня 2013 доступен из [http://phenix.int-evry.fr/jct/doc\\_end\\_user/documents/11\\_Shanghai/wg11/JCTVC-K1003-v13.zip](http://phenix.int-evry.fr/jct/doc_end_user/documents/11_Shanghai/wg11/JCTVC-K1003-v13.zip). Кроме того, прилагаются усилия для того, чтобы формировать SVC-, MVC- и 3DV-расширения для HEVC. 3DV-расширение HEVC может упоминаться в качестве 3DV на основе HEVC или 3DV HEVC.

[0059] В HEVC и других стандартах кодирования видео, видеопоследовательность типично включает в себя последовательность изображений. Изображения также могут упоминаться в качестве "кадров". Изображение может включать в себя три массива выборок, обозначаемых как  $S_L$ ,  $S_{Cb}$  и  $S_{Cr}$ .  $S_L$  представляет собой двумерный массив (т.е. блок) выборок сигнала яркости.  $S_{Cb}$  представляет собой двумерный массив выборок Cb-сигнала цветности.  $S_{Cr}$  представляет собой двумерный массив выборок Cr-сигнала цветности. Выборки сигнала цветности также могут упоминаться в данном документе как выборки "сигнала цветности". В других случаях, изображение может быть монохромным и может включать в себя только массив выборок сигнала яркости.

[0060] Для того, чтобы формировать кодированное представление изображения, видеокодер 20 может формировать набор единиц дерева кодирования (CTU). Каждая из CTU может представлять собой блок дерева кодирования выборок сигнала яркости, два соответствующих блока дерева кодирования выборок сигнала цветности и синтаксические структуры, используемые для того, чтобы кодировать выборки блоков дерева кодирования. Блок дерева кодирования может представлять собой блок  $N \times N$  выборок. CTU также может упоминаться в качестве "древовидного блока" или наибольшей единицы кодирования (LCU). CTU HEVC в широком смысле могут быть аналогичными макроблокам других стандартов, таких как H.264/AVC. Тем не менее, CTU не обязательно ограничивается конкретным размером и может включать в себя одну или более единиц кодирования (CU). Слайс может включать в себя целое число CTU, упорядоченных последовательно в растровом сканировании.

[0061] Для того, чтобы формировать кодированную CTU, видеокодер 20 может рекурсивно выполнять сегментацию на дерево квадрантов для блоков дерева кодирования CTU для того, чтобы разделять блоки дерева кодирования на блоки кодирования, отсюда имя "единицы дерева кодирования". Блок кодирования представляет собой блок  $N \times N$  выборок. CU может представлять собой блок кодирования выборок сигнала яркости и два соответствующих блока кодирования выборок сигнала цветности изображения, которое имеет массив выборок сигнала яркости, массив Cb-выборок и массив Cr-выборок, и синтаксические структуры, используемые для того, чтобы кодировать выборки блоков кодирования. Видеокодер 20 может сегментировать блок кодирования CU на один или более прогнозных блоков. Прогнозный блок может представлять собой прямоугольный (т.е. квадратный или неквадратный) блок выборок, к которым применяется идентичное прогнозирование. Единица прогнозирования (PU) CU может представлять собой прогнозный блок выборок сигнала яркости, два соответствующих прогнозных блока выборок сигнала цветности изображения и синтаксические структуры, используемые для того, чтобы прогнозировать выборки прогнозных блоков. Видеокодер 20 может формировать прогнозирующие блоки сигналов яркости, прогнозирующие Cb-блоки и прогнозирующие Cr-блоки для прогнозных блоков сигналов яркости, прогнозных Cb-блоков и прогнозных Cr-блоков каждой PU CU.

[0062] Видеокодер 20 может использовать внутреннее прогнозирование или внешнее прогнозирование для того, чтобы формировать прогнозирующие блоки для PU. Если видеокодер 20 использует внутреннее прогнозирование для того, чтобы формировать прогнозирующие блоки PU, видеокодер 20 может формировать прогнозирующие блоки PU на основе декодированных выборок изображения, ассоциированного с PU.

[0063] Если видеокодер 20 использует внешнее прогнозирование для того, чтобы формировать прогнозирующие блоки PU, видеокодер 20 может формировать



прогнозирующие блоки PU на основе декодированных выборок одного или более изображений, за исключением изображения, ассоциированного с PU. Видеокодер 20 может использовать унипрогнозирование или бипрогнозирование для того, чтобы

5 унипрогнозирование для того, чтобы формировать прогнозирующие блоки для PU, PU может иметь один вектор движения. Когда видеокодер 20 использует бипрогнозирование для того, чтобы формировать прогнозирующие блоки для PU, PU может иметь два вектора движения.

[0064] После того, как видеокодер 20 формирует прогнозирующие блоки сигналов яркости, прогнозирующие Сб-блоки и прогнозирующие Ср-блоки для одной или более PU CU, видеокодер 20 может формировать остаточный блок сигналов яркости для CU. Каждая выборка в остаточном блоке сигналов яркости CU указывает разность между

10 выборкой сигнала яркости в одном из прогнозирующих блоков сигналов яркости CU и соответствующей выборкой в исходном блоке кодирования сигналов яркости CU.

15 Помимо этого, видеокодер 20 может формировать остаточный Сб-блок для CU. Каждая выборка в остаточном Сб-блоке CU может указывать разность между Сб-выборкой в одном из прогнозирующих Сб-блоков CU и соответствующей выборкой в исходном Сб-блоке кодирования CU. Видеокодер 20 также может формировать остаточный Ср-блок для CU. Каждая выборка в остаточном Ср-блоке CU может указывать разность

20 между Ср-выборкой в одном из прогнозирующих Ср-блоков CU и соответствующей выборкой в исходном Ср-блоке кодирования CU.

[0065] Кроме того, видеокодер 20 может использовать сегментацию на дерево квадрантов для того, чтобы анализировать остаточные блоки сигналов яркости, остаточные Сб-блоки и остаточные Ср-блоки CU на один или более блоков

25 преобразования сигналов яркости, Сб-блоков преобразования и Ср-блоков преобразования. Блок преобразования может представлять собой прямоугольный блок выборок, к которым применяется идентичное преобразование. Единица преобразования (TU) CU может представлять собой блок преобразования выборок сигнала яркости, два соответствующих блока преобразования выборок сигнала цветности и

30 синтаксические структуры, используемые для того, чтобы преобразовывать выборки блока преобразования. Таким образом, каждая TU CU может быть ассоциирована с блоком преобразования сигналов яркости, Сб-блоком преобразования и Ср-блоком преобразования. Блок преобразования сигналов яркости, ассоциированный с TU, может представлять собой субблок остаточного блока сигналов яркости CU. Сб-блок

35 преобразования может представлять собой субблок остаточного Сб-блока CU. Ср-блок преобразования может представлять собой субблок остаточного Ср-блока CU.

[0066] Видеокодер 20 может применять одно или более преобразований к блоку преобразования сигналов яркости TU для того, чтобы формировать блок коэффициентов сигнала яркости для TU. Блок коэффициентов может представлять собой двумерный

40 массив коэффициентов преобразования. Коэффициент преобразования может быть скалярной величиной. Видеокодер 20 может применять одно или более преобразований к Сб-блоку преобразования TU для того, чтобы формировать Сб-блок коэффициентов для TU. Видеокодер 20 может применять одно или более преобразований к Ср-блоку преобразования TU для того, чтобы формировать Ср-блок коэффициентов для TU.

45 [0067] После формирования блока коэффициентов (например, блока коэффициентов сигнала яркости, Сб-блока коэффициентов или Ср-блока коэффициентов), видеокодер 20 может квантовать блок коэффициентов. Квантование, в общем, означает процесс, в котором коэффициенты преобразования квантуются, чтобы, возможно, уменьшать

объем данных, используемых для того, чтобы представлять коэффициенты преобразования, обеспечивая дополнительное сжатие. После того, как видеокодер 20 квантует блок коэффициентов, видеокодер 20 может энтропийно кодировать элементы синтаксиса, указывающие квантованные коэффициенты преобразования. Например, видеокодер 20 может выполнять контекстно-адаптивное двоичное арифметическое кодирование (CABAC) для элементов синтаксиса, указывающих квантованные коэффициенты преобразования. Видеокодер 20 может выводить энтропийно кодированные элементы синтаксиса в битовом потоке.

[0068] Видеокодер 20 может выводить битовый поток, который включает в себя энтропийно кодированные элементы синтаксиса. Битовый поток может включать в себя последовательность битов, которая формирует представление кодированных изображений и ассоциированных данных. Битовый поток может содержать последовательность единиц уровня сетевой абстракции (NAL). Каждая из NAL-единиц включает в себя заголовок NAL-единицы и инкапсулирует первичную байтовую последовательность данных (RBSP). Заголовок NAL-единицы может включать в себя элемент синтаксиса, который указывает код типа NAL-единицы. Код типа NAL-единицы, указываемый посредством заголовка NAL-единицы для NAL-единицы, указывает тип NAL-единицы. RBSP может представлять собой синтаксическую структуру, содержащую целое число байтов, которое инкапсулируется в NAL-единице. В некоторых случаях, RBSP включает в себя нулевые биты.

[0069] Различные типы NAL-единиц могут инкапсулировать различные типы RBSP. Например, первый тип NAL-единицы может инкапсулировать RBSP для набора параметров изображения (PPS), второй тип NAL-единицы может инкапсулировать RBSP для кодированного слайса, третий тип NAL-единицы может инкапсулировать RBSP для SEI и т.д. NAL-единицы, которые инкапсулируют RBSP для данных кодирования видео (в противоположность RBSP для наборов параметров и SEI-сообщений), могут упоминаться в качестве NAL-единиц слоя кодирования видео (VCL).

[0070] Видеодекодер 30 может принимать битовый поток, сформированный посредством видеокодера 20. Помимо этого, видеодекодер 30 может синтаксически анализировать битовый поток для того, чтобы декодировать элементы синтаксиса из битового потока. Видеодекодер 30 может восстанавливать изображения видеоданных, по меньшей мере, частично на основе элементов синтаксиса, декодированных из битового потока. Процесс для того, чтобы восстанавливать видеоданные, в общем, может быть обратным по отношению к процессу, выполняемому посредством видеокодера 20. Например, видеодекодер 30 может использовать векторы движения PU для того, чтобы определять прогнозирующие блоки для PU текущей CU. Помимо этого, видеодекодер 30 может обратно квантовать блоки коэффициентов преобразования, ассоциированные с TU текущей CU. Видеодекодер 30 может выполнять обратные преобразования для блоков коэффициентов преобразования для того, чтобы восстанавливать блоки преобразования, ассоциированные с TU текущей CU. Видеодекодер 30 может восстанавливать блоки кодирования текущей CU посредством суммирования выборок прогнозирующих блоков для PU текущей CU с соответствующими выборками блоков преобразования TU текущей CU. Посредством восстановления блоков кодирования для каждой CU изображения видеодекодер 30 может восстанавливать изображение.

[0071] При многовидовом кодировании могут быть несколько видов идентичной сцены с различных точек обзора. Термин "единица доступа" используется для того, чтобы означать набор изображений, которые соответствуют идентичному моменту

времени. Таким образом, видеоданные могут концептуализироваться в качестве последовательности единиц доступа, возникающих во времени. "Компонент вида" может быть кодированным представлением вида в одной единице доступа. В этом раскрытии сущности, "вид" может означать последовательность компонентов видов, ассоциированных с идентичным идентификатором вида.

[0072] Многовидовое кодирование поддерживает межвидовое прогнозирование. Межвидовое прогнозирование является аналогичным внешнему прогнозированию, используемому в HEVC, и может использовать идентичные элементы синтаксиса. Тем не менее, когда видеокодер выполняет межвидовое прогнозирование для текущей видеоединицы (к примеру, PU), видеокодер 20 может использовать, в качестве опорного изображения, изображение, которое находится в единице доступа, идентичной текущей видеоединице, но в другом виде. Напротив, традиционное внешнее прогнозирование использует только изображения в различных единицах доступа в качестве опорных изображений.

[0073] При многовидовом кодировании вид может упоминаться в качестве "базового вида", если видеодекодер (например, видеодекодер 30) может декодировать изображения в виде независимо от изображений в каком-либо другом виде. При кодировании изображения в одном из небазовых видов, видеокодер (к примеру, видеокодер 20 или видеодекодер 30) может добавлять изображение в список опорных изображений, если изображение находится в другом виде, но в идентичный момент времени (т.е. единице доступа), что и изображение, которое в данный момент кодирует видеокодер. Аналогично другим опорным изображениям внешнего прогнозирования, видеокодер может вставлять опорное изображение межвидового прогнозирования в любой позиции списка опорных изображений.

[0074] Стандарты кодирования видео указывают модели буферизации видео. В H.264/AVC и HEVC, модель буферизации упоминается в качестве "гипотетического эталонного декодера" или "HRD". В рабочем проекте 8 HEVC, HRD описывается в приложении С.

[0075] HRD описывает то, как данные должны быть буферизованы для декодирования, и как декодированные данные буферизуются для вывода. Например, HRD описывает работу CPB, буфера декодированных изображений (DPB) и процесс декодирования видео. CPB представляет собой буфер "первый на входе – первый на выходе", содержащий единицы доступа в порядке декодирования, указываемом посредством HRD. DPB представляет собой буфер, хранящий декодированные изображения для ссылки, переупорядочение вывода или задержку вывода, указываемую посредством HRD. Режимы работы CPB и DPB могут математически указываться. HRD может непосредственно налагать ограничения на синхронизацию, размеры буферов и скорости передачи битов. Кроме того, HRD может косвенно налагать ограничения на различные характеристики и статистику по битовому потоку.

[0076] В H.264/AVC и HEVC, соответствие битового потока и соответствие декодера указываются в качестве частей HRD-спецификации. Другими словами, HRD-модель указывает тесты с тем, чтобы определять то, соответствует или нет битовый поток стандарту, и тесты с тем, чтобы определять то, соответствует или нет декодер стандарту. Хотя HRD приводится в качестве некоторого типа декодера, видеокодеры типично используют HRD для того, чтобы гарантировать соответствие битового потока, в то время как видеодекодерам типично не требуется HRD.

[0077] H.264/AVC и HEVC указывают два типа соответствия битового потока или HRD-соответствия, а именно, тип I и тип II. Битовый поток типа I представляет собой поток NAL-единиц, содержащий только VCL NAL-единицы и NAL-единицу заполняющих

данных для всех единиц доступа в битовом потоке. Битовый поток типа II представляет собой поток NAL-единиц, который содержит, в дополнение к VCL NAL-единицам и NAL-единицам заполняющих данных для всех единиц доступа в битовом потоке по меньшей мере одно из следующего: дополнительные не-VCL NAL-единицы, отличные от NAL-единиц заполняющих данных; и все элементы `leading_zero_8bits`, `zero_byte`, `start_coded_prefix_one_3bytes` и `trailing_zero_8bits` синтаксиса, которые формируют байтовый поток из потока NAL-единиц.

[0078] Когда устройство выполняет тест на соответствие битового потока, который определяет то, соответствует или нет битовый поток стандарту кодирования видео, устройство может выбирать рабочую точку битового потока. Устройство затем может определять набор HRD-параметров, применимых к выбранной рабочей точке. Устройство может использовать набор HRD-параметров, применимых к выбранной рабочей точке, для того чтобы конфигурировать режим работы HRD. Более конкретно, устройство может использовать применимый набор HRD-параметров для того, чтобы конфигурировать режимы работы конкретных компонентов HRD, таких как гипотетический планировщик потоков (HSS), CPB, процесс декодирования, DPB и т.д. Затем, HSS может вводить закодированные видеоданные битового потока в CPB HRD согласно конкретному расписанию. Кроме того, устройство может активировать процесс декодирования, который декодирует закодированные видеоданные в CPB. Процесс декодирования может выводить декодированные изображения в DPB. Когда устройство перемещает данные через HRD, устройство может определять то, остается или нет удовлетворенным конкретный набор ограничений. Например, устройство может определять то, возникает или нет состояние переполнения или опустошенности в CPB или DPB, в то время как HRD декодирует представление рабочей точки для выбранной рабочей точки. Устройство может выбирать и обрабатывать каждую рабочую точку битового потока таким образом. Если ни одна рабочая точка битового потока не вызывает нарушения ограничений, устройство может определять то, что битовый поток соответствует стандарту кодирования видео.

[0079] Как H.264/AVC, так и HEVC указывают два типа соответствия декодера, а именно, соответствие декодера по синхронизации вывода и соответствие декодера по порядку вывода. Декодер, заявляющий соответствие конкретному профилю, ярусу и уровню, имеет возможность успешно декодировать все битовые потоки, которые соответствуют требованиям по соответствию битового потока стандарта кодирования видео, такого как HEVC. В этом раскрытии сущности, "профиль" может означать поднабор синтаксиса битового потока. "Ярусы" и "уровни" могут указываться в каждом профиле. Уровень яруса может представлять собой указанный набор ограничений, налагаемых на значения элементов синтаксиса в битовом потоке. Эти ограничения могут представлять собой простые пределы для значений. Альтернативно, ограничения могут принимать форму ограничений на арифметические комбинации значений (например, ширина изображения, умноженная на высоту изображения, умноженную на число изображений, декодированных в секунду). Типично, уровень, указываемый для нижнего яруса, является более ограниченным по сравнению с уровнем, указываемым для верхнего яруса.

[0080] Когда устройство выполняет тест на соответствие декодера, чтобы определять то, соответствует или нет тестируемый декодер (DUT) стандарту кодирования видео, устройство может предоставлять, как в HRD, так и в DUT, битовый поток, который соответствует стандарту кодирования видео. HRD может обрабатывать битовый поток способом, описанным выше относительно теста на соответствие битового потока.

Устройство может определять то, что DUT соответствует стандарту кодирования видео, если порядок вывода декодированных изображений посредством DUT совпадает с порядком вывода декодированных изображений посредством HRD. Кроме того, устройство может определять то, что DUT соответствует стандарту кодирования видео, если синхронизация, с которой DUT выводит декодированные изображения, совпадает с синхронизацией, с которой HRD выводит декодированные изображения.

[0081] В H.264/AVC и HEVC HRD-моделях, декодирование или удаление из CPB может быть основано на единице доступа. Иными словами, HRD предположительно должен декодировать полные единицы доступа одновременно и удалять полные единицы доступа из CPB. Кроме того, в H.264/AVC и HEVC HRD-моделях, предполагается, что декодирование изображений выполняется мгновенно. Видеокодер 20 может сигнализировать, в SEI-сообщениях по синхронизации изображений, времена декодирования для того, чтобы начинать декодирование единиц доступа. В практических вариантах применения, если соответствующий видеodeкодер строго следует временам декодирования, сигнализируемым, чтобы начинать декодирование единиц доступа, самое раннее время для того, чтобы выводить конкретное декодированное изображение, равно времени декодирования этого конкретного изображения плюс время, необходимое для декодирования этого конкретного изображения. Тем не менее, в реальном мире, время, необходимое для декодирования изображения, не может быть равно нулю.

[0082] HRD-параметры могут управлять различными аспектами HRD. Другими словами, HRD может основываться на HRD-параметрах. HRD-параметры могут включать в себя начальную задержку удаления из CPB, CPB-размер, скорость передачи битов, начальную задержку DPB-вывода и DPB-размер. Видеокодер 20 может сигнализировать эти HRD-параметры в синтаксической структуре `hrd_parameters()`, указываемой в наборе параметров видео (VPS) и/или в наборе параметров последовательности (SPS). Отдельные VPS и/или SPS могут включать в себя несколько синтаксических структур `hrd_parameters()` для различных наборов HRD-параметров. В некоторых примерах, видеокодер 20 может сигнализировать HRD-параметры в SEI-сообщениях по периоду буферизации или SEI-сообщениях по синхронизации изображений.

[0083] Как пояснено выше, рабочая точка битового потока ассоциирована с набором идентификаторов слоев (т.е. набором значений `nuh_reserved_zero_6bits`) и временным идентификатором. Представление рабочей точки может включать в себя каждую NAL-единицу, которая ассоциирована с рабочей точкой. Представление рабочей точки может иметь другую частоту кадров и/или скорость передачи битов по сравнению с исходным битовым потоком. Это обусловлено тем, что представление рабочей точки может не включать в себя некоторые изображения и/или некоторые данные исходного битового потока. Следовательно, если видеodeкодер 30 должен удалять данные из CPB и/или DPB на конкретной скорости при обработке исходного битового потока, и если видеodeкодер 30 должен удалять данные из CPB и/или DPB на идентичной скорости при обработке представления рабочей точки, видеodeкодер 30 может удалять слишком много или слишком мало данных из CPB и/или DPB. Соответственно, видеокодер 20 может сигнализировать различные наборы HRD-параметров для различных рабочих точек. Например, видеокодер 20 может включать в VPS несколько синтаксических структур `hrd_parameters()`, каждая из которых включает в себя HRD-параметры для различных рабочих точек.

[0084] В рабочем проекте 8 HEVC, набор HRD-параметров необязательно включает в себя набор информации, которая является общей для всех временных подслоев.

Другими словами, набор HRD-параметров необязательно может включать в себя набор общих элементов синтаксиса, которые применимы к рабочим точкам, которые включают в себя любые временные подслои. Временной подслоем может представлять собой временной масштабируемый слой временного масштабируемого битового потока, состоящего из VCL NAL-единиц с конкретным значением TemporalId и ассоциированных не-VCL NAL-единиц. В дополнение к набору общей информации, наборы HRD-параметров могут включать в себя набор элементов синтаксиса, которые являются конкретными для отдельных временных подслоев. Например, синтаксическая структура hrd\_parameters() необязательно может включать в себя набор информации, которая является общей для всех подслоев, и всегда включает в себя конкретную для подслоя информацию. Поскольку набор общей информации является общим для нескольких наборов из HRD-параметров, может быть необязательно сигнализировать набор общей информации в нескольких наборах HRD-параметров. Наоборот, в рабочем проекте 8 HEVC, общая информация может присутствовать в наборе HRD-параметров, когда набор HRD-параметров представляет собой первый набор HRD-параметров в VPS, или общая информация может присутствовать в наборе HRD-параметров, когда набор HRD-параметров ассоциирован с индексом первой рабочей точки. Например, рабочий проект 8 HEVC поддерживает присутствие общей информации, когда или синтаксическая структура hrd\_parameters() представляет собой первую синтаксическую структуру hrd\_parameters() в VPS, либо когда синтаксическая структура hrd\_parameters() ассоциирована с индексом первой рабочей точки.

[0085] Нижеприведенная таблица 1 представляет собой примерную синтаксическую структуру для синтаксической структуры hrd\_parameters() в HEVC.

ТАБЛИЦА 1	
HRD-параметры	
hrd_parameters (commonInfPresentFlag, MaxNumSubLayersMinus1) {	Дескриптор
if(commonInfPresentFlag) {	
timing_info_present_flag	u(1)
if(timing_info_present_flag) {	
num_units_in_tick	u(32)
time_scale	u(32)
}	
nal_hrd_parameters_present_flag	u(1)
vcl_hrd_parameters_present_flag	u(1)
if(nal_hrd_parameters_present_flag    vcl_hrd_parameters_present_flag) {	
sub_pic_cpb_params_present_flag	u(1)
if(sub_pic_cpb_params_present_flag) {	
tick_divisor_minus2	u(8)
du_cpb_removal_delay_length_minus1	u(5)
}	
bit_rate_scale	u(4)
cpb_size_scale	u(4)
initial_cpb_removal_delay_length_minus1	u(5)
cpb_removal_delay_length_minus1	u(5)
dpb_output_delay_length_minus1	u(5)
}	
}	
for(i=0; i<=MaxNumSubLayersMinus1; i++) {	
fixed_pic_rate_flag[i]	u(1)
if(fixed_pic_rate_flag[i])	
pic_duration_in_tc_minus1[i]	ue(v)
low_delay_hrd_flag[i]	u(1)

	cpb_cnt_minus1[i]	ue(v)
	if(nal_hrd_parameters_present_flag)	
	sub_layer_hrd_parameters(i)	
	if(vcl_hrd_parameters_present_flag)	
	sub_layer_hrd_parameters(i)	
5	}	
	}	

[0086] В примере вышеприведенной таблицы 1 и других синтаксических таблиц этого раскрытия сущности, элементы синтаксиса с дескриптором типа ue(v) могут быть целыми числами без знака переменной длины, кодированными с использованием экспоненциального кодирования кодом Голомба нулевого порядка (экспоненциального кодирования Голомба) с началом с левого бита. В примере таблицы 1 и следующих таблиц, элементы синтаксиса, имеющие дескрипторы формы u(n), где n является неотрицательным целым числом, являются значениями без знака длины n.

[0087] В примерном синтаксисе таблицы 1, элементы синтаксиса в блоке "if (commonInfPresentFlag) { ... }" представляют собой общую информацию синтаксических структур HRD-параметров. Другими словами, общая информация набора HRD-параметров может включать в себя элементы timing\_info\_present\_flag, num\_units\_in\_tick, time\_scale, nal\_hrd\_parameters\_present\_flag, vcl\_hrd\_parameters\_present\_flag, sub\_pic\_cpb\_parameters\_present\_flag, tick\_divisor\_minus2, du\_cpb\_removal\_delay\_length\_minus1, bit\_rate\_scale, cpb\_size\_scale, initial\_cpb\_removal\_delay\_length\_minus1, cpb\_removal\_delay\_length\_minus1 и dpb\_output\_delay\_length\_minus1 синтаксиса.

[0088] Кроме того, в примере таблицы 1, элементы fixed\_pic\_rate\_flag[i], pic\_duration\_in\_tc\_minus1[i], low\_delay\_hrd\_flag[i] и cpb\_cnt\_minus1[i] синтаксиса могут представлять собой набор конкретных для подслоя HRD-параметров. Другими словами, эти элементы синтаксиса синтаксической структуры hrd\_parameter() могут быть применимыми только к рабочим точкам, которые включают в себя конкретный подслой. Таким образом, HRD-параметры синтаксической структуры hrd\_parameters() могут включать в себя, в дополнение к необязательно включенной общей информации, набор конкретных для подслоя HRD-параметров, который является конкретным для конкретного подслоя битового потока.

[0089] Элемент fixed\_pic\_rate\_flag[i] синтаксиса может указывать то, что когда HighestTid равен i, временное расстояние между временами HRD-вывода любых двух последовательных изображений в порядке вывода ограничивается конкретным образом. HighestTid может представлять собой переменную, которая идентифицирует наибольший временной подслой (например, рабочей точки). Элемент pic\_duration\_in\_tc\_minus1[i] синтаксиса может указывать, когда HighestTid равен i, временное расстояние, в тактах синхросигнала, между временами HRD-вывода любых последовательных изображений в порядке вывода в кодированной видеопоследовательности. Элемент low\_delay\_hrd\_flag[i] синтаксиса может указывать режим работы HRD, когда HighestTid равен i, как указано в приложении C рабочего проекта 8 HEVC. Элемент cpb\_cnt\_minus1[i] синтаксиса может указывать число альтернативных CPB-спецификаций в битовом потоке кодированной видеопоследовательности, когда HighestTid равен i, при этом одна альтернативная CPB-спецификация означает одну конкретную CPB-операцию с конкретным набором CPB-параметров.

[0090] Видеокодер 20 может использовать SEI-сообщения для того, чтобы включать в битовый поток метаданные, которые не требуются для корректного декодирования выборочных значений изображений. Тем не менее, видеодекодер 30 или другие устройства могут использовать метаданные, включенные в SEI-сообщения, в различных

других целях. Например, видеodeкодер 30 или другое устройство может использовать метаданные в SEI-сообщениях для синхронизации вывода изображений, отображения изображения, обнаружения потерь и скрытия ошибок.

[0091] Видеокодер 20 может включать в себя одну или более SEI NAL-единиц в единице доступа. Другими словами, любое число SEI NAL-единиц может быть ассоциировано с единицей доступа. Кроме того, каждая SEI NAL-единица может содержать одно или более SEI-сообщений. HEVC-стандарт описывает синтаксис и семантику для различных типов SEI-сообщений. Тем не менее, HEVC-стандарт не описывает обработку SEI-сообщений, поскольку SEI-сообщения не влияют на нормативный процесс декодирования. Одна причина иметь SEI-сообщения в HEVC-стандарте состоит в том, чтобы обеспечивать возможность идентичной интерпретации вспомогательных данных в различных системах с использованием HEVC. Спецификации и системы с использованием HEVC могут требовать от видеокодеров формировать определенные SEI-сообщения или могут задавать конкретную обработку конкретных типов принимаемых SEI-сообщений. Нижеприведенная таблица 2 перечисляет SEI-сообщения, указываемые в HEVC, и кратко описывает их назначения.

Таблица 2	
Краткое описание SEI-сообщений	
SEI-сообщение	Назначение
Период буферизации	Начальные задержки для работы гипотетического эталонного декодера (HRD)
Синхронизация изображений	Время вывода изображений и время удаления изображений/субизображений для работы HRD
Прямоугольник сканирования с панорамированием	Отображение при другом соотношении сторон изображения (PAR) по сравнению с PAR выходных изображений
Заполняющие рабочие данные	Регулирование скорости передачи битов таким образом, что она удовлетворяет конкретным ограничениям
Зарегистрированные пользовательские данные Незарегистрированные пользовательские данные	SEI-сообщения, которые должны указываться посредством внешних объектов
Точка восстановления	Дополнительная информация для чистого произвольного доступа. Постепенное обновление при декодировании.
Информация сцен	Информация относительно смен и переходов сцен
Полнокадровый снимок экрана	Индикатор для того, чтобы пометить ассоциированное декодированное изображение в качестве снимка экрана в виде неподвижного изображения видеоконтента
Постепенный улучшающий сегмент	Указывает то, что определенные последовательные изображения представляют постепенное улучшение качества изображения, а не движущейся сцены
Характеристики зерна пленки	Обеспечивает возможность декодерам синтезировать зерно пленки
Настройки отображения с применением фильтра удаления блочности	Рекомендует то, должны или нет отображаемые изображения подвергаться процессу контурной фильтрации для удаления блочности
Подсказка по постфильтрации	Предоставляет предлагаемые коэффициенты постфильтрации или информацию корреляции для расчета постфильтра
Информация тонального преобразования	Повторное преобразование в другое цветовое пространство по сравнению с цветовым пространством, используемым или предполагаемым при кодировании
Компоновка пакетирования кадров	Пакетирование стереоскопического видео в HEVC-битовый поток
Ориентация отображения	Указывает зеркальное отражение и/или поворот, который должен применяться к выходным изображениям, когда они отображаются
Индикатор поля	Предоставляет информацию, связанную с видеоконтентом с чересстрочной разверткой и/или полевым кодированием, например, указывает то, представляет собой изображение построчный кадр, поле или кадр, содержащий два перемеженных поля
Хэш декодированного изображения	Контрольная сумма декодированного изображения, которая может использоваться для обнаружения ошибок
Синхронизация субизображений	Время удаления субизображений для работы HRD
Активные наборы параметров	Предоставляет информацию относительно активного VPS, SPS и т.д.
Структура описания изображений	Описывает временную структуру и структуру внешнего прогнозирования битового потока

[0092] Предварительная заявка на патент (США) 61/705102, поданная 24 сентября 2012 года, описывает различные способы для сигнализации и выбора HRD-параметров,



в том числе и сигнализации и выбора информации задержки и синхронизации в SEI-сообщениях. Документ Hannuksela и др. "AHG9: Operation points in VPS and nesting SEI", Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG11, 11<sup>th</sup> Meeting, Шанхай, CN, 10-19 октября 2012 года, документ № JCTVC-K0180v1, который с 13 июня 2013 года доступен по адресу [http://phenix.int-evry.fr/jct/doc\\_end\\_user/documents/11\\_Shanghai/wg11/JCTVC-K0180-v1.zip](http://phenix.int-evry.fr/jct/doc_end_user/documents/11_Shanghai/wg11/JCTVC-K0180-v1.zip), предоставляет другой способ для сигнализации HRD-параметров, а также механизм для вложения SEI-сообщений.

[0093] В существующих технологиях для сигнализации HRD-параметров имеется несколько проблем или недостатков. Например, существующие технологии не могут обеспечивать возможность совместного использования набора HRD-параметров посредством нескольких рабочих точек. Тем не менее, когда число рабочих точек является высоким, это может налагать нагрузку на видеокодер 20 или на другой модуль, который пытается обеспечивать соответствие битового потока для того, чтобы формировать различные наборы HRD-параметров для каждой рабочей точки. Наоборот, соответствие битового потока может быть обеспечено посредством проверки того, что каждая рабочая точка ассоциирована с набором HRD-параметров, но что конкретный набор HRD-параметров может совместно использоваться посредством нескольких рабочих точек. Одна или более технологий этого раскрытия сущности могут предоставлять схему для того, чтобы предоставлять возможность совместного использования одного набора HRD-параметров посредством нескольких рабочих точек. Другими словами, один набор HRD-параметров может быть применимым к нескольким рабочим точкам. Эта схема может давать возможность видеокодеру 20 или другому модулю, который пытается обеспечивать соответствие битового потока, находить компромисс между сложностью и производительностью.

[0094] В другом примере проблем или недостатков существующих технологий сигнализации HRD-параметров, когда предусмотрено несколько наборов из HRD-параметров в VPS, может быть желательным иметь несколько различных наборов общей информации для наборов HRD-параметров. Это может быть, в частности, истинным, когда имеются большие числа синтаксических структур HRD-параметров в VPS. Таким образом, может быть желательным иметь наборы общей информации в синтаксических структурах HRD-параметров, отличных от первой синтаксической структуры HRD-параметров. Например, для того чтобы предоставлять повышенную производительность, когда имеются несколько синтаксических структур `hrd_parameters()` в VPS, в частности, когда общее число синтаксических структур `hrd_parameters()` является относительно высоким, может быть желательным иметь различную общую информацию для синтаксических структур `hrd_parameters()`, отличающуюся от общей информации первой синтаксической структуры `hrd_parameters()` или отличающуюся от общей информации индекса первой рабочей точки.

[0095] Одна или более технологий этого раскрытия сущности предоставляют схему для того, чтобы предоставлять возможность явной сигнализации общей информации наборов HRD-параметров для любого набора HRD-параметров. Например, технологии этого раскрытия сущности могут предоставлять возможность явной сигнализации информации, которая является общей для всех подслоев для любой синтаксической структуры `hrd_parameters()`.

[0096] Таким образом, видеокодер 20 может сигнализировать в битовом потоке VPS, который включает в себя множество синтаксических структур HRD-параметров, каждая из которых включает в себя HRD-параметры. Для каждой соответствующей

синтаксической структуры HRD-параметров во множестве синтаксических структур HRD-параметров, VPS дополнительно включает в себя элемент синтаксиса, указывающий то, включают или нет HRD-параметры соответствующей синтаксической структуры HRD-параметров в себя общий набор HRD-параметров в дополнение к набору информации конкретных для подслоя HRD-параметров, конкретной для конкретного подслоя битового потока. Общий набор HRD-параметров является общим для всех подслоев битового потока.

[0097] Аналогично, видеodeкодер 30 или другое устройство может декодировать из битового потока VPS, который включает в себя множество синтаксических структур HRD-параметров, каждая из которых включает в себя HRD-параметры. Для каждой соответствующей синтаксической структуры HRD-параметров во множестве синтаксических структур HRD-параметров, VPS дополнительно может включать в себя элемент синтаксиса, указывающий то, включают или нет HRD-параметры соответствующей синтаксической структуры HRD-параметров в себя общий набор HRD-параметров. Видеodeкодер 30 или другое устройство может выполнять операцию с использованием HRD-параметров по меньшей мере одной из синтаксических структур HRD-параметров.

[0098] Кроме того, существующие способы для вложения SEI-сообщений могут иметь несколько проблем или недостатков. Например, существующие технологии сигнализации HRD-параметров не могут предоставлять возможность применения одного SEI-сообщения к нескольким рабочим точкам. Технологии этого раскрытия сущности могут предоставлять схему для того, чтобы предоставлять возможность применения одного SEI-сообщения к нескольким рабочим точкам.

[0099] В частности, масштабируемое вкладывающее SEI-сообщение может включать в себя элементы синтаксиса, которые указывают несколько рабочих точек, применимых к SEI-сообщениям, вложенным в масштабируемом вкладывающем SEI-сообщении. Другими словами, масштабируемые вкладывающие SEI-сообщения могут предоставлять механизм для ассоциирования SEI-сообщений с поднабором битовых потоков (например, представлением рабочей точки) или с конкретными слоями и подслоями.

[0100] Таким образом, видеodeкодер 20 может формировать масштабируемое вкладывающее SEI-сообщение, которое включает в себя множество элементов синтаксиса, которые идентифицируют множество рабочих точек, к которым применяется вложенное SEI-сообщение, инкапсулированное посредством масштабируемого вкладывающего SEI-сообщения. Кроме того, видеodeкодер 20 может сигнализировать масштабируемое вкладывающее SEI-сообщение в битовом потоке.

[0101] Таким образом, видеodeкодер 30 или другое устройство, в процессе кодирования видео, может декодировать, из масштабируемого вкладывающего SEI-сообщения, множество элементов синтаксиса, которые идентифицируют рабочие точки, к которым применяется вложенное SEI-сообщение, инкапсулированное посредством масштабируемого вкладывающего SEI-сообщения. Кроме того, видеodeкодер 30 или другое устройство может выполнять операцию, по меньшей мере, частично на основе одного или более элементов синтаксиса вложенного SEI-сообщения.

[0102] Другой пример проблем или недостатков существующих технологий вкладывающих SEI-сообщений связан с тем фактом, что существующие технологии вкладывающих SEI-сообщений не используют значение элемента синтаксиса идентификатора слоя (например, `nuh_reserved_zero_6bits`) в текущей SEI NAL-единице для того, чтобы определять рабочую точку, применимую к масштабируемым вложенным SEI-сообщениям, инкапсулированным посредством текущей SEI NAL-единицы.

[0103] Технологии этого раскрытия сущности предоставляют схему, которая сигнализирует то, представляет собой или нет рабочая точка, применимая к вложенным SEI-сообщениям в SEI NAL-единице, рабочую точку, указываемую посредством

5 Идентификационная информация слоя в заголовке NAL-единицы для SEI NAL-единицы может включать в себя значение `nuh_reserved_zero_6bits` и значение `nuh_temporal_id_plus1` заголовка NAL-единицы. Другими словами, технологии этого раскрытия сущности могут предоставлять схему для использования идентификационной информации слоя (например, значения `nuh_reserved_zero_6bits` и `nuh_temporal_id_plus1`) в заголовке NAL-единицы для текущей SEI NAL-единицы (т.е. SEI NAL-единицы, содержащей масштабируемое вкладывающее SEI-сообщение), посредством сигнализации того, применяются или нет вложенные SEI-сообщения к рабочей точке по умолчанию, идентифицированной посредством идентификационной информации слоя, включенной в заголовки NAL-единицы для текущей SEI NAL-единицы.

15 [0104] Таким образом, видеокодер 20 может включать в масштабируемое вкладывающее SEI-сообщение, инкапсулированное посредством SEI NAL-единицы, элемент синтаксиса, который указывает то, является или нет вложенное SEI-сообщение, инкапсулированное посредством масштабируемого вкладывающего SEI-сообщения, применимым к битовому субпотoku по умолчанию. Битовый субпоток по умолчанию  
20 может быть представлением рабочей точки для рабочей точки, заданной посредством идентификатора слоя, точно определяемого в заголовке NAL-единицы для SEI NAL-единицы, и временного идентификатора, точно определяемого в заголовке NAL-единицы. Кроме того, видеокодер 20 может выводить битовый поток, который включает в себя масштабируемое вкладывающее SEI-сообщение.

25 [0105] Аналогично, устройство, такое как видеодекодер 30 или другое устройство, может определять, по меньшей мере, частично на основе элемента синтаксиса в масштабируемом вкладывающем SEI-сообщении, инкапсулированном посредством SEI NAL-единицы, то, является или нет вложенное SEI-сообщение, инкапсулированное посредством масштабируемого вкладывающего SEI-сообщения, применимым к  
30 битовому субпотoku по умолчанию. Как указано выше, битовый субпоток по умолчанию может быть представлением рабочей точки для рабочей точки, заданной посредством идентификатора слоя, точно определяемого в заголовке NAL-единицы для SEI NAL-единицы, и временного идентификатора, точно определяемого в заголовке NAL-единицы. Когда вложенное SEI-сообщение является применимым к битовому субпотoku по  
35 умолчанию, устройство может использовать вложенное SEI-сообщение при операции над битовым субпотком по умолчанию. Например, вложенное SEI-сообщение может включать в себя один или более HRD-параметров. В этом примере, устройство может использовать один или более HRD-параметров для того, чтобы выполнять тест на соответствие битового потока, который определяет то, соответствует или нет битовый  
40 субпоток по умолчанию стандарту кодирования видео, такому как HEVC. Альтернативно, в этом примере, устройство может использовать один или более HRD-параметров для того, чтобы определять то, удовлетворяет или нет видеодекодер 30 тесту на соответствие декодера.

[0106] В другом примере проблем или недостатков существующих способов для  
45 вложения SEI-сообщений, явное кодирование идентификаторов слоев является неэффективным. Технологии этого раскрытия сущности позволяют повышать эффективность явного кодирования идентификаторов слоев через дифференциальное кодирование или кодирование с использованием флагов.

[0107] Фиг. 2 является блок-схемой, иллюстрирующий примерный видеокодер 20, который может реализовывать технологии этого раскрытия сущности. Фиг. 2 предоставляется для целей пояснения и не должен считаться ограничением технологий, как проиллюстрировано и описано в общих чертах в этом раскрытии сущности. Для  
 5 целей пояснения, это раскрытие сущности описывает видеокодер 20 в контексте HEVC-кодирования. Тем не менее, технологии этого раскрытия сущности могут быть применимыми к другим стандартам или способам кодирования.

[0108] В примере по фиг. 2, видеокодер 20 включает в себя процессор 100 прогнозирования, модуль 102 формирования остатков, процессор 104 преобразования,  
 10 модуль 106 квантования, модуль 108 обратного квантования, процессор 110 обратного преобразования, модуль 112 восстановления, модуль 114 фильтрации, буфер 116 декодированных изображений и модуль 118 энтропийного кодирования. Процессор 100 прогнозирования включает в себя процессор 120 внешнего прогнозирования и процессор 126 внутреннего прогнозирования. Процессор 120 внешнего прогнозирования  
 15 включает в себя модуль 122 оценки движения и модуль 124 компенсации движения. В других примерах, видеокодер 20 может включать в себя большее число, меньше число или другие функциональные компоненты.

[0109] Видеокодер 20 может принимать видеоданные. Видеокодер 20 может кодировать каждую STU в слайсе изображения видеоданных. Каждая из STU может  
 20 быть ассоциирована с блоками дерева кодирования (СТВ) сигнала яркости одинакового размера и соответствующими СТВ изображения. В качестве части кодирования STU, процессор 100 прогнозирования может выполнять сегментацию на дерево квадрантов для того, чтобы разделять СТВ STU на постепенно меньшие блоки. Меньшие блоки могут представлять собой блоки кодирования CU. Например, процессор 100  
 25 прогнозирования может сегментировать СТВ, ассоциированный с STU, на четыре субблока одинакового размера, сегментировать один или более субблоков на четыре субсубблока одинакового размера и т.д.

[0110] Видеокодер 20 может кодировать CU STU для того, чтобы формировать кодированные представления CU (т.е. кодированные CU). В качестве части кодирования  
 30 CU, процессор 100 прогнозирования может сегментировать блоки кодирования, ассоциированные с CU, из числа одной или более PU CU. Таким образом, каждая PU может быть ассоциирована с прогнозным блоком сигналов яркости и соответствующими прогнозными блоками сигнала цветности. Видеокодер 20 и видеодекодер 30 могут поддерживать PU, имеющие различные размеры. Как указано выше, размер CU может  
 35 означать размер блока кодирования сигналов яркости CU, и размер PU может означать размер прогнозного блока сигналов яркости PU. При условии, что размер конкретной CU составляет  $2N \times 2N$ , видеокодер 20 и видеодекодер 30 могут поддерживать PU-размеры  $2N \times 2N$  или  $N \times N$  для внутреннего прогнозирования и симметричные PU-размеры  $2N \times 2N$ ,  $2N \times N$ ,  $N \times 2N$ ,  $N \times N$  или аналогичные для внешнего прогнозирования. Видеокодер 20 и  
 40 видеодекодер 30 также могут поддерживать асимметричное сегментирование для PU-размеров  $2N \times nU$ ,  $2N \times nD$ ,  $nL \times 2N$  и  $nR \times 2N$  для внешнего прогнозирования.

[0111] Процессор 120 внешнего прогнозирования может формировать прогнозирующие данные для PU посредством выполнения внешнего прогнозирования для каждой PU CU. Прогнозирующие данные для PU могут включать в себя  
 45 прогнозирующие блоки, которые соответствуют PU, и информацию движения для PU. Процессор 120 внешнего прогнозирования может выполнять различные операции для PU CU в зависимости от того, находится PU в I-слайсе, P-слайсе или B-слайсе. В I-слайсе все PU внутренне прогнозируются. Следовательно, если PU находится в I-слайсе,

процессор 120 внешнего прогнозирования не выполняет внешнее прогнозирование для PU. Таким образом, для блоков, кодированных в I-режиме, прогнозирующий блок формируется с использованием пространственного прогнозирования из ранее кодированных соседних блоков в идентичном кадре.

5 [0112] Если PU находится в Р-слайсе, модуль 122 оценки движения может выполнять поиск в опорных изображениях в списке опорных изображений (например, в RefPicList0) на предмет опорной области для PU. Опорная область для PU может представлять собой область в опорном изображении, которая содержит блоки выборок, которые наиболее близко соответствуют блокам выборок PU. Модуль 122 оценки движения  
10 может формировать опорный индекс, который указывает позицию в RefPicList0 опорного изображения, содержащего опорную область для PU. Помимо этого, модуль 122 оценки движения может формировать вектор движения, который указывает пространственное смещение между блоком кодирования PU и опорным местоположением, ассоциированным с опорной областью. Например, вектор движения может представлять  
15 собой двумерный вектор, который предоставляет смещение от координат в текущем изображении на координаты в опорном изображении. Модуль 122 оценки движения может выводить опорный индекс и вектор движения в качестве информации движения PU. Модуль 124 компенсации движения может формировать прогнозирующие блоки PU на основе фактических или интерполированных выборок в опорном местоположении,  
20 указываемом посредством вектора движения PU.

[0113] Если PU находится в В-слайсе, модуль 122 оценки движения может выполнять унипрогнозирование или бипрогнозирование для PU. Чтобы выполнять унипрогнозирование для PU, модуль 122 оценки движения может выполнять поиск в опорных изображениях RefPicList0 или второго списка опорных изображений  
25 (RefPicList1) на предмет опорной области для PU. Модуль 122 оценки движения может выводить, в качестве информации движения PU, опорный индекс, который указывает позицию в RefPicList0 или RefPicList1 опорного изображения, которое содержит опорную область, вектор движения, который указывает пространственное смещение между прогнозным блоком PU и опорным местоположением, ассоциированным с опорной  
30 областью, и один или более индикаторов направления прогнозирования, которые указывают то, находится опорное изображение в RefPicList0 или в RefPicList1. Модуль 124 компенсации движения может формировать прогнозирующие блоки PU, по меньшей мере, частично на основе фактических или интерполированных выборок в опорной области, указываемой посредством вектора движения PU.

35 [0114] Чтобы выполнять двунаправленное внешнее прогнозирование для PU, модуль 122 оценки движения может выполнять поиск в опорных изображениях RefPicList0 на предмет опорной области для PU, а также может выполнять поиск в опорных изображениях RefPicList1 на предмет другой опорной области для PU. Модуль 122 оценки движения может формировать опорные индексы, которые указывают позиции  
40 в RefPicList0 и RefPicList1 опорных изображений, которые содержат опорные области. Помимо этого, модуль 122 оценки движения может формировать векторы движения, которые указывают пространственные смещения между опорными местоположениями, ассоциированными с опорными областями, и прогнозным блоком PU. Информация движения PU может включать в себя опорные индексы и векторы движения PU. Модуль  
45 124 компенсации движения может формировать прогнозирующие блоки PU, по меньшей мере, частично на основе фактических или интерполированных выборок в опорной области, указываемой посредством вектора движения PU.

[0115] Процессор 126 внутреннего прогнозирования может формировать

прогнозирующие данные для PU посредством выполнения внутреннего прогнозирования для PU. Прогнозирующие данные для PU могут включать в себя прогнозирующие блоки для PU и различные элементы синтаксиса. Процессор 126 внутреннего прогнозирования может выполнять внутреннее прогнозирование для PU в I-слайсах, R-слайсах и B-слайсах.

[0116] Чтобы выполнять внутреннее прогнозирование для PU, процессор 126 внутреннего прогнозирования может использовать несколько режимов внутреннего прогнозирования для того, чтобы формировать несколько наборов прогнозирующих данных для PU. Процессор 126 внутреннего прогнозирования может формировать прогнозирующий блок для PU на основе выборок соседних PU. Соседние PU могут располагаться выше, выше и справа, выше и слева или слева от PU, при условии порядка кодирования слева направо, сверху вниз для PU, CU и CTU. Процессор 126 внутреннего прогнозирования может использовать различные числа режимов внутреннего прогнозирования, например, 33 режима направленного внутреннего прогнозирования. В некоторых примерах, число режимов внутреннего прогнозирования может зависеть от размера прогнозирующих блоков PU.

[0117] Процессор 100 прогнозирования может выбирать прогнозирующие данные для PU CU из числа прогнозирующих данных, сформированных посредством процессора 120 внешнего прогнозирования для PU, или прогнозирующих данных, сформированных посредством процессора 126 внутреннего прогнозирования для PU. В некоторых примерах, процессор 100 прогнозирования выбирает прогнозирующие данные для PU CU на основе показателей искажения в зависимости от скорости передачи наборов прогнозирующих данных. Прогнозирующие блоки выбранных прогнозирующих данных могут упоминаться в данном документе как выбранные прогнозирующие блоки.

[0118] Модуль 102 формирования остатков может формировать, на основе блока кодирования сигналов яркости, Сб-блока кодирования и Ср-блока кодирования CU и выбранных прогнозирующих блоков сигналов яркости, прогнозирующих Сб-блоков и прогнозирующих Ср-блоков PU CU, остаточные блоки сигналов яркости, остаточные Сб-блоки и остаточные Ср-блоки CU. Например, модуль 102 формирования остатков может формировать остаточные блоки CU таким образом, что каждая выборка в остаточных блоках имеет значение, равное разности между выборкой в блоке кодирования CU и соответствующей выборкой в соответствующем выбранном прогнозирующем блоке PU CU.

[0119] Процессор 104 преобразования может выполнять сегментацию на дерево квадрантов для того, чтобы сегментировать остаточные блоки CU на блоки преобразования, ассоциированные с TU CU. Таким образом, TU может быть ассоциирована с блоком преобразования сигналов яркости и двумя соответствующими блоками преобразования сигналов цветности. Размеры и позиции блоков преобразования сигналов яркости и сигналов цветности TU CU могут быть основаны либо могут не быть основаны на размерах и позициях прогнозных блоков PU CU. Структура в виде дерева квадрантов, известная как "остаточное дерево квадрантов" (RQT), может включать в себя узлы, ассоциированные с каждой из областей. TU CU могут соответствовать концевым узлам RQT.

[0120] Процессор 104 преобразования может формировать блоки коэффициентов преобразования для каждой TU CU посредством применения одного или более преобразований к блокам преобразования TU. Процессор 104 преобразования может применять различные преобразования к блоку преобразования, ассоциированному с TU. Например, процессор 104 преобразования может применять дискретное косинусное

преобразование (DCT), направленное преобразование или концептуально аналогичное преобразование к блоку преобразования. В некоторых примерах, процессор 104 преобразования не применяет преобразования к блоку преобразования. В таких примерах, блок преобразования может трактоваться в качестве блока коэффициентов преобразования.

[0121] Модуль 106 квантования может квантовать коэффициенты преобразования в блоке коэффициентов. Процесс квантования может уменьшать битовую глубину, ассоциированную с некоторыми или всеми коэффициентами преобразования. Например,  $n$ -битовый коэффициент преобразования может округляться в меньшую сторону до  $m$ -битового коэффициента преобразования во время квантования, где  $n$  превышает  $m$ . Модуль 106 квантования может квантовать блок коэффициентов, ассоциированный с TU CU, на основе значения параметра квантования (QP), ассоциированного с CU. Видеокодер 20 может регулировать степень квантования, применяемого к блокам коэффициентов, ассоциированным с CU, посредством регулирования QP-значения, ассоциированного с CU. Квантование может вводить потери информации, в силу чего квантованные коэффициенты преобразования могут иметь меньшую точность по сравнению с исходными коэффициентами преобразования.

[0122] Модуль 108 обратного квантования и процессор 110 обратного преобразования могут применять обратное квантование и обратные преобразования к блоку коэффициентов, соответственно, для того чтобы восстанавливать остаточный блок из блока коэффициентов. Модуль 112 восстановления может суммировать восстановленный остаточный блок с соответствующими выборками из одного или более прогнозирующих блоков, сформированных посредством процессора 100 прогнозирования, для того чтобы формировать восстановленный блок преобразования, ассоциированный с TU. Посредством восстановления блоков преобразования для каждой TU CU таким способом, видеокодер 20 может восстанавливать блоки кодирования CU.

[0123] Модуль 114 фильтрации может выполнять одну или более операций удаления блочности, чтобы уменьшать артефакты блочности в блоках кодирования, ассоциированных с CU. Буфер 116 декодированных изображений может сохранять восстановленные блоки кодирования после того, как модуль 114 фильтрации выполняет одну или более операций удаления блочности для восстановленных блоков кодирования. Процессор 120 внешнего прогнозирования может использовать опорное изображение, которое содержит восстановленные блоки кодирования, для того чтобы выполнять внешнее прогнозирование для PU других изображений. Помимо этого, процессор 126 внутреннего прогнозирования может использовать восстановленные блоки кодирования в буфере 116 декодированных изображений для того, чтобы выполнять внутреннее прогнозирование для других PU в изображении, идентичном изображению CU.

[0124] Модуль 118 энтропийного кодирования может принимать данные из других функциональных компонентов видеокодера 20. Например, модуль 118 энтропийного кодирования может принимать блоки коэффициентов из модуля 106 квантования и может принимать элементы синтаксиса из процессора 100 прогнозирования. Модуль 118 энтропийного кодирования может выполнять одну или более операций энтропийного кодирования для данных, чтобы формировать энтропийно кодированные данные. Например, модуль 118 энтропийного кодирования может выполнять операцию контекстно-адаптивного кодирования переменной длины (CAVLC), CABAC-операцию, операцию кодирования переменного-переменной (V2V) длины, операцию синтаксического контекстно-адаптивного двоичного арифметического кодирования (SBAC), операцию энтропийного кодирования на основе сегментирования на интервалы вероятности

(PIPE), операцию экспоненциального кодирования кодом Голomba или другой тип операции энтропийного кодирования для данных. Вideoкодер 20 может выводить битовый поток, который включает в себя энтропийно кодированные данные, сформированные посредством модуля 118 энтропийного кодирования. Например,

битовый поток может включать в себя данные, которые представляют RQT для CU. [0125] Как указано выше, технологии этого раскрытия сущности могут предоставлять схему для того, чтобы предоставлять возможность явной сигнализации общей информации синтаксических структур HRD-параметров для любой синтаксической структуры HRD-параметров в VPS. Чтобы предоставлять возможность явной сигнализации общей информации синтаксических структур HRD-параметров для любого HRD-параметра в VPS, видеокодер 20 может формировать синтаксические структуры VPS, которые соответствуют примерному синтаксису, показанному в нижеприведенной таблице 3.

15	Таблица 3	
	Синтаксическая структура VPS	
	video_parameter_set_rbsp( ) {	Дескриптор
	video_parameter_set_id	u(4)
	vps_temporal_id_nesting_flag	u(1)
	reserved_zero_2bits	u(2)
20	reserved_zero_6bits	u(6)
	vps_max_sub_layers_minus1	u(3)
	profile_tier_level(1, vps_max_sub_layers_minus1)	
	ext_essential_info_byte_offset/reserved_zero_12bits в базовой спецификации	u(12)
	for(i=0; i<=vps_max_sub_layers_minus1; i++) {	
	vps_max_dec_pic_buffering[i]	ue(v)
25	vps_max_num_reorder_pics[i]	ue(v)
	vps_max_latency_increase[i]	ue(v)
	}	
	num_ops_minus1	ue(v)
	for(i=1; i<=num_ops_minus1; i++)	
	operation_point(i)	
30	vps_num_hrd_parameters	ue(v)
	for(i=0; i<vps_num_hrd_parameters; i++) {	
	hrd_applicable_ops_minus1[i]	ue(v)
	for(j=0; j<=hrd_applicable_ops_minus1[i]; j++) {	
	hrd_op_idx[i][j]	ue(v)
	if(i==0)	
35	cprms_present_flag[i]	
	hrd_parameters(cprms_present_flag[i], vps_max_sub_layers_minus1)	
	}	
	vps_extension_flag	u(1)
	if(vps_extension_flag)	
40	while(more_rbsp_data( ) )	
	vps_extension_data_flag	u(1)
	rbsp_trailing_bits( )	
	}	

[0126] Курсивные части таблицы 3 указывают отличия между синтаксисом таблицы 3 и соответствующей таблицы из рабочего проекта 8 HEVC. Кроме того, в примерном синтаксисе таблицы 3, элемент num\_ops\_minus1 синтаксиса указывает число синтаксических структур operation\_point, присутствующих в VPS. Элемент hrd\_applicable\_ops\_minus1[i] синтаксиса указывает число рабочих точек, к которым применяется i-ая синтаксическая структура hrd\_parameters(). Элемент hrd\_op\_idx[i][j]



синтаксиса указывает j-ую рабочую точку, к которой применяется i-ая синтаксическая структура `hrd_parameters()` в VPS. Как вкратце упомянуто выше, технологии этого раскрытия сущности могут предоставлять возможность совместного использования одного набора HRD-параметров посредством нескольких рабочих точек. Элементы `hrd_applicable_ops_minus1[i]` и `hrd_op_idx[i][j]` синтаксиса могут служить для того, чтобы указывать рабочие точки, к которым применяется набор HRD-параметров. В некоторых примерах, в которых нескольким рабочим точкам не разрешается быть применимыми к одному набору HRD-параметров, элементы `hrd_applicable_ops_minus1[i]` синтаксиса и элементы `hrd_op_idx[i][j]` синтаксиса опускаются из таблицы 3.

[0127] В примерном синтаксисе таблицы 3, VPS может включать в себя набор флагов присутствия общих параметров (т.е. элементов синтаксиса), обозначаемый в таблице 3 как `sprms_present_flag[i]`. Элемент `sprms_present_flag[i]` синтаксиса, равный 1, указывает то, что HRD-параметры, которые являются общими для всех подслоев, присутствуют в i-ой синтаксической структуре `hrd_parameters()` в VPS. Элемент `sprms_present_flag[i]` синтаксиса, равный 0, указывает то, что HRD-параметры, которые являются общими для всех подслоев, не присутствуют в i-ой синтаксической структуре `hrd_parameters()` в VPS, а вместо этого извлекаются как идентичные (i-1)-ой синтаксической структуре `hrd_parameters()` в VPS.

[0128] `Sprms_present_flag[0]` может логически выводиться равным 1. Иными словами, устройство может автоматически определять (т.е. логически выводить) то, что первая (в порядке кодирования) синтаксическая структура `hrd_parameters()` в VPS включает в себя HRD-параметры, которые являются общими для всех подслоев. Следовательно, первая синтаксическая структура HRD-параметров, сигнализируемая в VPS, включает в себя общий набор HRD-параметров. Одна или более последующих синтаксических структур HRD-параметров в VPS могут включать в себя различные общие наборы HRD-параметров.

[0129] Как вкратце упомянуто выше, технологии этого раскрытия сущности могут предоставлять возможность явной сигнализации общей информации (т.е. HRD-параметров, общих для каждого из подслоев) синтаксических структур HRD-параметров для любой синтаксической структуры HRD-параметров. Элемент `sprms_present_flag[i]` синтаксиса таблицы 3 может предоставлять возможность видеodeкодеру 30 или другому устройству определять то, какая из синтаксических структур HRD-параметров включает в себя набор HRD-параметров, общих для каждого из подслоев. Таким образом, в то время как первая синтаксическая структура HRD-параметров может всегда включать в себя общий набор HRD-параметров, одна или более синтаксических структур HRD-параметров, сигнализируемых в VPS, не включают в себя общий набор HRD-параметров. Устройство может использовать элементы `sprms_present_flag[i]` синтаксиса, чтобы определять то, какая из синтаксических структур HRD-параметров VPS включает в себя общие наборы HRD-параметров.

[0130] Синтаксическая структура HRD-параметров (например, синтаксическая структура `hrd_parameters()`) может включать в себя набор конкретных для подслоя HRD-параметров независимо от того, включает или нет синтаксическая структура HRD-параметров в себя HRD-параметры, которые являются общими для всех подслоев. Когда видеodeкодер 30 или другое устройство определяет то, что конкретная синтаксическая структура HRD-параметров не включает в себя общий набор HRD-параметров, видеodeкодер 30 или другое устройство может выполнять операцию с использованием общего набора HRD-параметров, ассоциированных с предыдущей синтаксической структурой HRD-параметров, и набора конкретных для подслоя HRD-

параметров конкретной синтаксической структуры HRD-параметров. Предыдущая синтаксическая структура HRD-параметров может представлять собой набор HRD-параметров, сигнализируемый в VPS раньше, в порядке кодирования, конкретной синтаксической структуры HRD-параметров. Если предыдущая синтаксическая структура HRD-параметров включает в себя общий набор HRD-параметров, общий набор HRD-параметров, ассоциированных с предыдущей синтаксической структурой HRD-параметров, представляет собой общий набор HRD-параметров, включенных в предыдущую синтаксическую структуру HRD-параметров. Если предыдущая синтаксическая структура HRD-параметров не включает в себя общий набор HRD-параметров, устройство может определять то, что общий набор HRD-параметров, ассоциированных с предыдущей синтаксической структурой HRD-параметров, представляет собой общий набор HRD-параметров, ассоциированных с синтаксической структурой HRD-параметров, предшествующей, порядку кодирования, предыдущей синтаксической структуре HRD-параметров в порядке кодирования.

[0131] Как упомянуто выше, устройство может выполнять операцию с использованием общего набора HRD-параметров и конкретных для подслоя HRD-параметров. Во время этой операции, устройство может управлять работой CPB согласно одному или более HRD-параметров, декодировать видеоданные и управлять декодированными изображениями в DPB согласно одному или более HRD-параметров. В другом примере, общий набор HRD-параметров и конкретные для подслоя HRD-параметры могут использоваться для того, чтобы выполнять тест на соответствие битового потока или тест на соответствие декодера.

[0132] Кроме того, в некоторых примерах, масштабируемое вкладывающее SEI-сообщение предоставляет механизм для ассоциирования SEI-сообщений с поднаборами битовых потоков (например, представлениями рабочих точек) или с конкретными слоями и подслоями. В некоторых таких примерах, масштабируемое вкладывающее SEI-сообщение может содержать одно или более SEI-сообщений. SEI-сообщение, содержащееся в масштабируемом вкладывающем SEI-сообщении, может упоминаться в качестве вложенного SEI-сообщения. SEI-сообщение, не содержащееся в масштабируемом вкладывающем SEI-сообщении, может упоминаться в качестве невложенного SEI-сообщения. В некоторых примерах, вложенное SEI-сообщение в масштабируемом вкладывающем SEI-сообщении может включать в себя набор HRD-параметров.

[0133] В некоторых примерах, имеется несколько ограничений на то, какие типы сообщений могут быть вложены. Например, SEI-сообщение по периоду буферизации и SEI-сообщение любого другого типа не могут быть вложены в идентичное масштабируемое вкладывающее SEI-сообщение. SEI-сообщение по периоду буферизации может указывать начальные задержки для работы HRD. В другом примере, SEI-сообщение по синхронизации изображений и SEI-сообщение любого другого типа не могут быть вложены в идентичное масштабируемое вкладывающее SEI-сообщение. SEI-сообщение по синхронизации изображений может указывать время вывода изображений и время удаления изображений/субизображений для работы HRD. В других примерах, SEI-сообщение по синхронизации изображений и SEI-сообщение по синхронизации субизображений могут быть вложены в идентичное масштабируемое вкладывающее SEI-сообщение. SEI-сообщение по синхронизации субизображений может предоставлять информацию задержки удаления из CPB для декодированной единицы, ассоциированной с SEI-сообщением.

[0134] Как указано выше, одна или более технологий этого раскрытия сущности

могут предоставлять возможность применения одного SEI-сообщения к нескольким рабочим точкам. Кроме того, одна или более технологий этого раскрытия сущности могут предоставлять возможность видеокодеру 20 сигнализировать то, представляет собой или нет рабочая точка, применимая к вложенному SEI-сообщению в SEI NAL-единице, рабочую точку, указываемую посредством идентификационной информации слоя в заголовке NAL-единицы для SEI NAL-единицы. Помимо этого, одна или более технологий этого раскрытия сущности позволяют повышать эффективность явного кодирования идентификаторов слоев через дифференциальное кодирование. Примерный синтаксис, показанный в нижеприведенной таблице 4, и прилагаемая семантика позволяют реализовывать эти технологии.

Таблица 4		
Масштабируемые вкладывающие SEI-сообщения		
	scalable_nesting (payloadSize) {	Дескриптор
	bitstream_subset_flag	u(1)
15	if(bitstream_subset_flag) {	
	default_op_applicable_flag	u(1)
	nesting_num_ops_minus1	ue(v)
	} else	
	nesting_op_flag	u(1)
20	for(i=0; i<=nesting_num_ops_minus1; i++)	
	nesting_max_temporal_id_plus1[i]	u(3)
	if(nesting_op_flag)	
	for(i=0; i<=nesting_num_ops_minus1; i++)	
	nesting_op_idx[i]	ue(v)
	else {	
25	all_layers_flag	u(1)
	if(!all_layers_flag) {	
	nesting_num_layers_minus1	ue(v)
	for(i=0; i<=nesting_num_layers_minus1; i++)	
	nesting_layer_id_delta[i]	ue(v)
	}	
30	}	
	while(!byte_aligned( ))	
	nesting_zero_bit/*равен 0*/	u(1)
	Do	
	sei_message( )	
	while(more_rbsp_data( ))	
35	}	

[0135] В примере таблицы 4, курсивные части могут указывать отличия от рабочего проекта 8 HEVC. В частности, в примерном синтаксисе таблицы 4, элемент bitstream\_subset\_flag синтаксиса, равный 0, указывает то, что SEI-сообщения, вложенные в масштабируемое вкладывающее SEI-сообщениеб применяются к конкретным слоям и подслоям. Элемент bitstream\_subset\_flag синтаксиса, равный 1, указывает то, что SEI-сообщения, вложенные в масштабируемое вкладывающее SEI-сообщениеб применяются к битовому субпотoku, получающемуся в результате процесса извлечения битовых субпотокoв подраздела 10.1 рабочего проекта 8 HEVC, с вводами, указываемыми посредством элементов синтаксиса масштабируемого вкладывающего SEI-сообщения, как указано ниже. Подраздел 10.1 рабочего проекта 8 HEVC описывает операцию для извлечения битового субпотoka (т.е. представления рабочей точки) из битового потока. В частности, подраздел 10.1 рабочего проекта 8 HEVC предусматривает то, что битовый субпоток извлекается посредством удаления из битового потока всех NAL-единиц с

временными идентификаторами (например, TemporalID), превышающими tIdTarget, либо с идентификаторами слоев (например, nuh\_reserved\_zero\_6bits) не в числе значений в targetDecLayerIdSet. TIdTarget и targetDecLayerIdSet представляют собой параметры процесса извлечения битовых потоков. В некоторых примерах, если вложенные SEI-сообщения представляют собой SEI-сообщения буферизации изображений, SEI-сообщения по синхронизации изображений или SEI-сообщения по синхронизации субизображений, элемент bitstream\_subset\_flag синтаксиса равен 1. В противном случае, в таких примерах, элемент bitstream\_subset\_flag синтаксиса равен 0.

[0136] Кроме того, в примерном синтаксисе таблицы 4, масштабируемое вкладывающее SEI-сообщение включает в себя элемент default\_op\_applicable\_flag синтаксиса, если элемент bitstream\_subset\_flag синтаксиса равен 1. Элемент default\_op\_applicable\_flag синтаксиса, равный 1, указывает то, что вложенные SEI-сообщения (т.е. SEI-сообщения, вложенные в масштабируемое вкладывающее SEI-сообщение) применяются к битовому субпотoku по умолчанию, который является выводом процесса извлечения битовых субпотокa подраздела 10.1 рабочего проекта 8 HEVC, с вводами tIdTarget, равными временному идентификатору (TemporalId) текущей SEI NAL-единицы, и targetDecLayerIdSet, состоящего из всех значений nuh\_reserved\_zero\_6bits в диапазоне от 0 до nuh\_reserved\_zero\_6bits текущей SEI NAL-единицы, включительно. Таким образом, битовый субпоток по умолчанию может представлять собой битовый поток, который извлекается посредством удаления из битового потока всех NAL-единиц с временными идентификаторами, превышающими временной идентификатор текущей SEI NAL-единицы, или с идентификаторами слоев в диапазоне от 0 до идентификатора слоя (например, nuh\_reserved\_zero\_6bits) текущей SEI NAL-единицы, включительно. Например, битовый субпоток по умолчанию может представлять собой поднабор битового потока, и битовый субпоток по умолчанию может не включать в себя VCL NAL-единицы битового потока, которые имеют идентификаторы слоев, превышающие идентификатор слоя, указываемый посредством элемента синтаксиса идентификатора слоя заголовка NAL-единицы, или имеют временные идентификаторы, превышающие временной идентификатор, указываемый посредством элемента синтаксиса идентификатора временного слоя (например, nuh\_temporal\_id\_plus1) заголовка NAL-единицы. Элемент default\_op\_applicable\_flag синтаксиса, равный 0, указывает то, что вложенные SEI-сообщения не применяются к битовому субпотoku по умолчанию.

[0137] В примерном синтаксисе таблицы 4, масштабируемое вкладывающее SEI-сообщение включает в себя элемент nesting\_num\_ops\_minus1 синтаксиса, если элемент bitstream\_subset\_flag синтаксиса равен 1. Элемент nesting\_num\_ops\_minus1 синтаксиса, плюс 1, указывает число элементов nesting\_op\_idx[i] синтаксиса в масштабируемом вкладывающем SEI-сообщении. Таким образом, если элемент nesting\_num\_ops\_minus1 синтаксиса, плюс 1, превышает 0, элемент nesting\_num\_ops\_minus1 синтаксиса может указывать то, включает или нет масштабируемое вкладывающее SEI-сообщение в себя множество элементов синтаксиса, которые идентифицируют несколько рабочих точек, к которым применимы вложенные SEI-сообщения. Таким образом, устройство может декодировать, из масштабируемого вкладывающего SEI-сообщения, элемент синтаксиса (nesting\_num\_ops\_minus1), который указывает число рабочих точек, к которым применяется вложенное SEI-сообщение. Когда элемент nesting\_num\_ops\_minus1 синтаксиса не присутствует, значение nesting\_num\_ops\_minus1 может логически выводиться равным 0. Таким образом, если элемент bitstream\_subset\_flag синтаксиса равен 0, масштабируемое вкладывающее SEI-сообщение не включает в себя элементов

nesting\_op\_idx[i] синтаксиса.

[0138] Элемент nesting\_op\_flag синтаксиса, равный 0, указывает то, что nestingLayerIdSet[0] указывается посредством элемента all\_layers\_flag синтаксиса, и когда присутствует, элемента nesting\_layer\_id\_delta[i] синтаксиса для всех значений  $i$  в диапазоне от 0 до nesting\_num\_layers\_minus1, включительно. Элементы nestingLayerIdSet[ ] синтаксиса представляют собой массив идентификаторов слоев. Элемент nesting\_op\_flag синтаксиса, равный 1, указывает то, что nestingLayerIdSet[i] указывается посредством элемента nesting\_op\_idx[i] синтаксиса. Если не присутствует, значение nesting\_op\_flag логически выводится равным 1.

[0139] Элемент nesting\_max\_temporal\_id\_plus1[i] синтаксиса указывает переменную maxTemporalId[i]. В примерном синтаксисе таблицы 4, значение элемента nesting\_max\_temporal\_id\_plus1[i] синтаксиса превышает значение элемента nuh\_temporal\_id\_plus1 синтаксиса текущей SEI NAL-единицы (т.е. NAL-единицы, которая содержит масштабируемое вложенное SEI-сообщение). Переменная maxTemporalId[i] задается равной nesting\_max\_temporal\_id\_plus1[i]-1.

[0140] Элемент nesting\_op\_idx[i] синтаксиса используется для того, чтобы указывать набор nestingLayerIdSet[i]. Набор nestingLayerIdSet[i] может состоять из op\_layer\_id[nesting\_op\_idx][i] со всеми значениями  $i$  в диапазоне от 0 до op\_num\_layer\_id\_values\_minus1[nesting\_op\_idx], включительно. Активный VPS может указывать значения op\_layer\_id[ ][ ] и op\_num\_layer\_values\_minus1[ ].

[0141] Кроме того, в примерном синтаксисе таблицы 4, элемент all\_layers\_flag синтаксиса, равный 0, указывает то, что набор nestingLayerIdSet[0] состоит из nestingLayerId[i] для всех значений  $i$  в диапазоне от 0 до nesting\_num\_layers\_minus1, включительно. Переменная nestingLayerId[i] описывается ниже. Элемент all\_layers\_flag синтаксиса, равный 1, указывает то, что набор nestingLayerIdSet состоит из всех значений nuh\_reserved\_zero\_6bits, присутствующих в текущей единице доступа, которые равны или превышают nuh\_reserved\_zero\_6bits текущей SEI NAL-единицы.

[0142] Элемент nesting\_num\_layers\_minus1 синтаксиса, плюс 1, указывает число элементов nesting\_layer\_id\_delta[i] синтаксиса в масштабируемом вкладываемом SEI-сообщении. Элемент nesting\_layer\_id\_delta[i] синтаксиса, когда  $i$  равен 0, указывает разность между первым (т.е. 0-ым) значением nuh\_reserved\_zero\_6bits, включенным в набор nestingLayerIdSet[0], и элементом nuh\_reserved\_zero\_6bits синтаксиса текущей SEI NAL-единицы. Элемент nesting\_layer\_id\_delta[i] синтаксиса, когда  $i$  превышает 0, указывает разность между  $i$ -ым и  $(i-1)$ -ыми значениями nuh\_reserved\_zero\_6bits, включенными в набор nestingLayerIdSet[0].

[0143] Переменная nestingLayerId[i] может извлекаться следующим образом, при этом nuh\_reserved\_zero\_6bits исходит из заголовка NAL-единицы для текущей SEI NAL-единицы.

nestingLayerId[0]=nuh\_reserved\_zero\_6bits+ nesting\_layer\_id\_delta[0] for( $i=1$ ;  $i \leq$  nesting\_num\_layers\_minus1;  $i++$ ) nestingLayerId[i]=nestingLayerId[i-1]+nesting\_layer\_id\_delta[i] Набор nestingLayerIdSet[0] задается таким образом, что он состоит из nestingLayerId[i] для всех значений  $i$  в диапазоне от 0 до nesting\_num\_layers\_minus1, включительно. Когда элемент bitstream\_subset\_flag синтаксиса равен 0, вложенные SEI-сообщения применяются к NAL-единицам с nuh\_reserved\_zero\_6bits, включенным в набор nestingLayerIdSet[0] или равным nuh\_reserved\_zero\_6bits текущей SEI NAL-единицы, и с nuh\_temporal\_id\_plus1 в диапазоне от nuh\_temporal\_id\_plus1 текущей SEI NAL-единицы до maxTemporalId[0]+1, включительно. Когда элемент bitstream\_subset\_flag синтаксиса равен 1, вложенные SEI-сообщения применяются к выводу процесса извлечения битовых

субпоток подраздела 10.1 рабочего проекта 8 HEVC, с вводами `tIdTarget`, равными `maxTemporalId[i]`, и `targetDecLayerIdSet`, равными `nestingLayerIdSet[i]`, для каждого значения  $i$  в диапазоне от 0 до `nesting_num_ops_minus1`, включительно, и когда элемент `default_op_applicable_flag` синтаксиса равен 1, вложенные SEI-сообщения также

5 применяются к битовому субпотoku по умолчанию. Извлеченный битовый субпоток может получаться в результате удаления всех NAL-единиц с временными идентификаторами, превышающими `maxTemporalId[i]`, или с идентификаторами слоев в диапазоне от 0 до `nesting_num_ops_minus1`.

[0144] Таким образом, по меньшей мере, для одной соответствующей рабочей точки

10 во множестве рабочих точек, к которым применяется вложенное SEI-сообщение, устройство (например, видеокoder 20, видеокoder 30 или другое устройство, такое как устройство сети доставки контента) может декодировать, из масштабируемого вкладывающего SEI-сообщения, первый элемент синтаксиса (например, `nesting_max_temporal_id_plus1[i]`) и второй элемент синтаксиса (например, `nesting_op_idx[i]`).

15 Кроме того, устройство может определять, по меньшей мере, частично на основе первого элемента синтаксиса, максимальный временной идентификатор соответствующей рабочей точки. Устройство может определять, по меньшей мере, частично на основе второго элемента синтаксиса, набор идентификаторов слоев соответствующей рабочей точки.

[0145] В примере таблицы 4, элемент `nesting_zero_bit` синтаксиса равен 0. Элемент `nesting_zero_bit` синтаксиса может служить для того, чтобы обеспечивать, что масштабируемое вкладывающее SEI-сообщение имеет байтовое совмещение. Масштабируемое вкладывающее SEI-сообщение может иметь байтовое совмещение, когда число битов в масштабируемом вкладывающем SEI-сообщении делится на 8.

[0146] Кроме того, в примере таблицы 4, синтаксические структуры `sei_message()` включают в себя SEI-сообщения. Таким образом, устройство может декодировать, из масштабируемого вкладывающего SEI-сообщения, множество вложенных SEI-сообщений, инкапсулированных посредством масштабируемого вкладывающего SEI-сообщения. Каждое из вложенных SEI-сообщений может быть применимым ко всем

30 рабочим точкам, идентифицированным посредством множества элементов синтаксиса (например, `nesting_max_temporal_id_plus1[i]`, `nesting_op_idx[i]` и т.д.).

[0147] В альтернативном примере, масштабируемые вкладывающие SEI-сообщения могут соответствовать примерному синтаксису нижеприведенной таблицы 5. В примерном синтаксисе таблицы 5, масштабируемое вкладывающее SEI-сообщение, в

35 соответствии с одной или более технологий этого раскрытия сущности, может повышать эффективность явного кодирования идентификаторов слоев с помощью кодирования флагов.

Таблица 5	Масштабируемые вкладывающие SEI-сообщения	
40	<code>scalable_nesting (payloadSize) {</code>	Дескриптор
	...	
	<code>if(bitstream_subset_flag) {</code>	
	<code>default_op_applicable_flag</code>	u(1)
	<code>nesting_num_ops_minus1</code>	ue(v)
45	<code>} else</code>	
	<code>nesting_op_flag</code>	u(1)
	<code>for(i=0; i&lt;=nesting_num_ops_minus1; i++)</code>	
	<code>nesting_max_temporal_id_plus1[i]</code>	u(3)
	<code>if(nesting_op_flag)</code>	

	for(i=0; i<=nesting_num_ops_minus1; i++)	
	nesting_op_idx[i]	ue(v)
	else {	
	all_layers_flag	u(1)
5	if(!all_layers_flag) {	
	nesting_max_layer_id	u(6)
	minLayerId=nuh_reserved_zero_6bits+1	
	for(i=0; i<nesting_max_layer_id-minLayerId; i++)	
	nesting_layer_id_included_flag[i]	ue(v)
	}	
10	}	
	while(!byte_aligned( ))	
	nesting_zero_bit/*равен 0*/	u(1)
	Do	
	sei_message( )	
	while(more_rbsp_data( ))	
15	}	

[0148] В примере таблицы 5, курсивные части показывают отличия от рабочего проекта 8 HEVC. Как показано в таблице 5, элемент *bitstream\_subset\_flag* синтаксиса, элемент *default\_op\_applicable\_flag* синтаксиса, элемент *nesting\_num\_ops\_minus1* синтаксиса, элемент *nesting\_max\_temporal\_id\_plus1* синтаксиса, элемент *nesting\_op\_idx[i]* синтаксиса и элемент *nesting\_zero\_bit* синтаксиса могут иметь идентичную семантику, как описано выше относительно таблицы 4.

[0149] Кроме того, в примере таблицы 5, переменная *minLayerId* задается равной *nuh\_reserved\_zero\_6bits+1*, при этом *nuh\_reserved\_zero\_6bits* исходит из заголовка NAL-единицы для текущей SEI NAL-единицы. Элемент *nesting\_op\_flag* синтаксиса, равный 0, указывает то, что набор *nestingLayerIdSet[0]* указывается посредством элемента *all\_layers\_flag* синтаксиса, и когда присутствует, *nesting\_layer\_id\_included\_flag[i]* для всех значений *i* в диапазоне от 0 до *nesting\_max\_layer\_id-minLayerId-1*, включительно. Элемент *nesting\_op\_flag* синтаксиса, равный 1, указывает то, что набор *nestingLayerIdSet[i]* указывается посредством элемента *nesting\_op\_idx[i]* синтаксиса. Когда элемент *nesting\_op\_flag* синтаксиса не присутствует, значение *nesting\_op\_flag* логически выводится равным 1.

[0150] В примере таблицы 5, элемент *all\_layers\_flag* синтаксиса, равный 0, указывает то, что набор *nestingLayerIdSet[0]* состоит из *nestingLayerId[i]* для всех значений *i* в диапазоне от 0 до *nesting\_max\_layer\_id-minLayerId*, включительно. Переменная *nestingLayerId[i]* описывается ниже. В примере таблицы 5, *all\_layers\_flag* равно 1 указывает то, что набор *nestingLayerIdSet* состоит из всех значений *nuh\_reserved\_zero\_6bits*, присутствующих в текущей единице доступа, которые превышают или равны элементу *nuh\_reserved\_zero\_6bits* синтаксиса текущей SEI NAL-единицы.

[0151] Кроме того, в примере таблицы 5, элемент *nesting\_max\_layer\_id* синтаксиса указывает самое большое значение *nuh\_reserved\_zero\_6bits* в наборе *nestingLayerIdSet[0]*. Элемент *nesting\_layer\_id\_included\_flag[i]* синтаксиса, равный 1, указывает то, что значение *nuh\_reserved\_zero\_6bits*, равное *i+minLayerId*, включено в набор *nestingLayerIdSet[0]*. Элемент *nesting\_layer\_id\_included\_flag[i]* синтаксиса, равный 0, указывает то, что значение *nuh\_reserved\_zero\_6bits*, равное *i+minLayerId*, не включено в набор *nestingLayerIdSet[0]*.

[0152] Переменная *nestingNumLayersMinus1* и переменные *nestingLayerId[i]* для *i* в диапазоне от 0 до *nestingNumLayersMinus1*, включительно, могут извлекаться следующим образом:

```

    for(i=0, j=0; i<nesting_max_layer_id; i++)
    if(nesting_layer_id_included_flag[i]) nestingLayerId[ j++] =i+minLayerId nestingLayerId[ j
]=nesting_max_layer_id
    nestingNumLayersMinus1=j

```

Набор nestingLayerIdSet[0] может задаваться таким образом, что он состоит из nestingLayerId[i] для всех значений i в диапазоне от 0 до nestingNumLayersMinus1, включительно.

[0153] Когда элемент bitstream\_subset\_flag синтаксиса равен 0, вложенные SEI-сообщения могут применяться к NAL-единицам с nuh\_reserved\_zero\_6bits, включенным в набор nestingLayerIdSet[0] или равным элементу nuh\_reserved\_zero\_6bits синтаксиса текущей SEI NAL-единицы, и с nuh\_temporal\_id\_plus1 в диапазоне от элемента nuh\_temporal\_id\_plus1 синтаксиса текущей SEI NAL-единицы до maxTemporalId[0]+1, включительно.

[0154] Когда элемент bitstream\_subset\_flag синтаксиса масштабируемого вложенного SEI-сообщения равен 1, вложенные SEI-сообщения могут применяться к выводу процесса извлечения битовых субпоток под раздела 10.1, с вводами tIdTarget, равными maxTemporalId[i], и targetDecLayerIdSet, равными nestingLayerIdSet[i] для каждого значения i в диапазоне от 0 до nesting\_num\_ops\_minus1, включительно, и когда default\_or\_applicable\_flag равен 1, вложенные SEI-сообщения также применяются к битовому субпотoku по умолчанию.

[0155] Фиг. 3 является блок-схемой, иллюстрирующей примерный видеodeкодер 30, который сконфигурирован с возможностью реализовывать технологии этого раскрытия сущности. Фиг. 3 предоставляется для целей пояснения и не является ограничением технологий, как проиллюстрировано и описано в общих чертах в этом раскрытии сущности. Для целей пояснения, это раскрытие сущности описывает видеodeкодер 30 в контексте HEVC-кодирования. Тем не менее, технологии этого раскрытия сущности могут быть применимыми к другим стандартам или способам кодирования.

[0156] В примере по фиг. 3, видеodeкодер 30 включает в себя модуль 150 энтропийного декодирования, процессор 152 прогнозирования, модуль 154 обратного квантования, процессор 156 обратного преобразования, модуль 158 восстановления, модуль 160 фильтрации и буфер 162 декодированных изображений. Процессор 152 прогнозирования включает в себя модуль 164 компенсации движения и процессор 166 внутреннего прогнозирования. В других примерах, видеodeкодер 30 может включать в себя большее число, меньше число или другие функциональные компоненты.

[0157] Буфер 151 кодированных изображений (CPB) может принимать и сохранять кодированные видеоданные (например, NAL-единицы) битового потока. Модуль 150 энтропийного декодирования может принимать NAL-единицы из CPB 151 и синтаксически анализировать NAL-единицы, чтобы декодировать элементы синтаксиса. Модуль 150 энтропийного декодирования может энтропийно декодировать энтропийно кодированные элементы синтаксиса в NAL-единицах. Процессор 152 прогнозирования, модуль 154 обратного квантования, процессор 156 обратного преобразования, модуль 158 восстановления и модуль 160 фильтрации могут формировать декодированные видеоданные на основе элементов синтаксиса, извлеченных из битового потока.

[0158] NAL-единицы битового потока могут включать в себя NAL-единицы кодированных слайсов. В качестве части декодирования битового потока, модуль 150 энтропийного декодирования может извлекать и энтропийно декодировать элементы синтаксиса из NAL-единиц кодированных слайсов. Каждый из кодированных слайсов может включать в себя заголовок слайса и данные слайса. Заголовок слайса может



содержать элементы синтаксиса, связанные со слайсом. Элементы синтаксиса в заголовке слайса могут включать в себя элемент синтаксиса, который идентифицирует PPS, ассоциированный с изображением, которое содержит слайс.

[0159] Помимо декодирования элементов синтаксиса из битового потока, видеodeкодер 30 может выполнять операцию восстановления для несегментированной CU. Чтобы выполнять операцию восстановления для несегментированной CU, видеodeкодер 30 может выполнять операцию восстановления для каждой TU CU. Посредством выполнения операции восстановления для каждой TU CU видеodeкодер 30 может восстанавливать остаточные блоки, ассоциированные с CU.

[0160] В качестве части выполнения операции восстановления для TU CU, модуль 154 обратного квантования может обратно квантовать, т.е. деквантовать, блоки коэффициентов, ассоциированные с TU. Модуль 154 обратного квантования может использовать QR-значение, ассоциированное с CU TU, для того чтобы определять степень квантования и, аналогично, степень обратного квантования для модуля 154 обратного квантования, которое должно применяться. Иными словами, коэффициент сжатия, т.е. отношение числа битов, используемых для того, чтобы представлять исходную последовательность, и числа сжатых битов, может управляться посредством регулирования значения QR, используемого при квантовании коэффициентов преобразования. Коэффициент сжатия также может зависеть от используемого способа энтропийного кодирования.

[0161] После того, как модуль 154 обратного квантования обратно квантует блок коэффициентов, процессор 156 обратного преобразования может применять одно или более обратных преобразований к блоку коэффициентов, чтобы формировать остаточный блок, ассоциированный с TU. Например, процессор 156 обратного преобразования может применять обратное DCT, обратное целочисленное преобразование, обратное преобразование Карунена-Лозва (KLT), обратное вращательное преобразование, обратное направленное преобразование или другое обратное преобразование к блоку коэффициентов.

[0162] Если PU кодируется с использованием внутреннего прогнозирования, процессор 166 внутреннего прогнозирования может выполнять внутреннее прогнозирование для того, чтобы формировать прогнозирующие блоки для PU. Процессор 166 внутреннего прогнозирования может использовать режим внутреннего прогнозирования для того, чтобы формировать прогнозирующие блоки сигналов яркости, прогнозирующие Cb-блоки и прогнозирующие Cr-блоки для PU на основе прогнозных блоков пространственно соседних PU. Процессор 166 внутреннего прогнозирования может определять режим внутреннего прогнозирования для PU на основе одного или более элементов синтаксиса, декодированных из битового потока.

[0163] Процессор 152 прогнозирования может составлять первый список опорных изображений (RefPicList0) и второй список опорных изображений (RefPicList1) на основе элементов синтаксиса, извлеченных из битового потока. Кроме того, если PU кодируется с использованием внешнего прогнозирования, модуль 150 энтропийного декодирования может извлекать информацию движения для PU. Модуль 164 компенсации движения может определять, на основе информации движения PU, одну или более опорных областей для PU. Модуль 164 компенсации движения может формировать, на основе блоков выборок в одном или более опорных блоках для PU, прогнозирующие блоки сигналов яркости, прогнозирующие Cb-блоки и прогнозирующие Cr-блоки для PU.

[0164] Модуль 158 восстановления может использовать блоки преобразования сигналов яркости, Cb-блоки преобразования и Cr-блоки преобразования,

ассоциированные с TU CU, и прогнозирующие блоки сигналов яркости, прогнозирующие Сб-блоки и прогнозирующие Ср-блоки PU CU, т.е. либо данные внутреннего прогнозирования, либо данные внешнего прогнозирования, при соответствующих условиях, для того чтобы восстанавливать блоки кодирования сигналов яркости, Сб-блоки кодирования и Ср-блоки кодирования CU. Например, модуль 158 восстановления может суммировать выборки блоков преобразования сигналов яркости, Сб-блоков преобразования и Ср-блоков преобразования с соответствующими выборками прогнозирующих блоков сигналов яркости, прогнозирующих Сб-блоков и прогнозирующих Ср-блоков для того, чтобы восстанавливать блоки кодирования сигналов яркости, Сб-блоки кодирования и Ср-блоки кодирования CU.

[0165] Модуль 160 фильтрации может выполнять операцию удаления блочности, чтобы уменьшать артефакты блочности, ассоциированные с блоками кодирования сигналов яркости, Сб-блоками кодирования и Ср-блоками кодирования CU.

Видеодекодер 30 может сохранять блоки кодирования сигналов яркости, Сб-блоки кодирования и Ср-блоки кодирования CU в буфере 162 декодированных изображений. Буфер 162 декодированных изображений может предоставлять опорные изображения для последующей компенсации движения, внутреннего прогнозирования и представления на устройстве отображения, к примеру, на устройстве 32 отображения по фиг. 1.

Например, видеодекодер 30 может выполнять, на основе блоков сигналов яркости, Сб-блоков и Ср-блоков в буфере 162 декодированных изображений, операции внутреннего прогнозирования или внешнего прогнозирования для PU других CU. Таким образом, видеодекодер 30 может извлекать, из битового потока, уровни коэффициентов преобразования блока значимых коэффициентов сигнала яркости, обратно квантовать уровни коэффициентов преобразования, применять преобразование к уровням коэффициентов преобразования, чтобы формировать блок преобразования, формировать, по меньшей мере, частично на основе блока преобразования, блок кодирования и выводить блок кодирования для отображения.

[0166] Фиг. 4 является блок-схемой последовательности операций способа, иллюстрирующей примерную работу 200 видеокодера 20, в соответствии с одной или более технологий этого раскрытия сущности. В примере по фиг. 4, видеокодер 20 может формировать VPS, который включает в себя множество синтаксических структур HRD-параметров, каждая из которых включает в себя HRD-параметры (202). Для каждой соответствующей синтаксической структуры HRD-параметров во множестве синтаксических структур HRD-параметров, VPS дополнительно включает в себя элемент синтаксиса, указывающий то, включают или нет HRD-параметры соответствующей синтаксической структуры HRD-параметров в себя общий набор HRD-параметров в дополнение к набору информации конкретных для подслоя HRD-параметров, конкретной для конкретного подслоя битового потока, при этом общий набор HRD-параметров является общим для всех подслоев битового потока. Кроме того, видеокодер 20 может сигнализировать VPS в битовом потоке (204).

[0167] Фиг. 5 является блок-схемой последовательности операций способа, иллюстрирующей примерную работу 250 устройства, в соответствии с одной или более технологий этого раскрытия сущности. Работа 250 может выполняться посредством видеокодера 20, видеодекодера 30 или другого устройства. Как проиллюстрировано в примере по фиг. 5, устройство может декодировать из битового потока VPS, который включает в себя множество синтаксических структур HRD-параметров, каждая из которых включает в себя HRD-параметры (252). Для каждой соответствующей синтаксической структуры HRD-параметров во множестве синтаксических структур

HRD-параметров, VPS дополнительно включает в себя элемент синтаксиса, указывающий то, включают или нет HRD-параметры соответствующей синтаксической структуры HRD-параметров в себя общий набор HRD-параметров.

[0168] Кроме того, устройство может выполнять операцию с использованием HRD-параметров по меньшей мере одной из синтаксических структур HRD-параметров (254). В некоторых примерах, битовый поток может содержать представление рабочей точки для конкретной рабочей точки, конкретная синтаксическая структура HRD-параметров может быть применимой к конкретной рабочей точке, и устройство может выполнять операцию с использованием HRD-параметров конкретной синтаксической структуры HRD-параметров. Например, устройство может использовать HRD-параметры для того, чтобы выполнять тест на соответствие битового потока, который определяет то, соответствует или нет рабочей точке, применимая к синтаксической структуре HRD-параметров, стандарту кодирования видео, такому как HEVC. В другом примере, устройство может использовать HRD-параметры для того, чтобы выполнять тест на соответствие декодера.

[0169] Общий набор HRD-параметров может быть общим для всех подслоев битового потока. В некоторых примерах, HRD-параметры каждой синтаксической структуры HRD-параметров включают в себя набор конкретных для подслоя HRD-параметров, который является конкретным для конкретного подслоя битового потока. В некоторых примерах, каждый из наборов конкретных для подслоя HRD-параметров включает в себя элемент синтаксиса (например, указывающий временное расстояние между временами HRD-вывода любых двух последовательных изображений в порядке вывода), элемент синтаксиса, указывающий число альтернативных спецификаций буфера кодированных изображений в битовом потоке кодированной видеопоследовательности. В некоторых примерах, когда устройство определяет то, что конкретная синтаксическая структура HRD-параметров не включает в себя общий набор HRD-параметров, устройство может выполнять операцию с использованием общего набора HRD-параметров, ассоциированных с предыдущей синтаксической структурой HRD-параметров, и набора конкретных для подслоя HRD-параметров конкретной синтаксической структуры HRD-параметров.

[0170] Фиг. 6 является блок-схемой последовательности операций способа, иллюстрирующей примерную работу 300 видеокодера 20, в соответствии с одной или более технологий этого раскрытия сущности. Как проиллюстрировано в примере по фиг. 6, видеокодер 20 может формировать масштабируемое вкладывающее SEI-сообщение, которое включает в себя множество элементов синтаксиса, которые идентифицируют множество рабочих точек, к которым применяется вложенное SEI-сообщение, инкапсулированное посредством масштабируемого вкладывающего SEI-сообщения (302). Кроме того, видеокодер 20 может сигнализировать масштабируемое вкладывающее SEI-сообщение в битовом потоке (304).

[0171] Фиг. 7 является блок-схемой последовательности операций способа, иллюстрирующей примерную работу 350 устройства, в соответствии с одной или более технологий этого раскрытия сущности. Видеокодер 20, видеodeкодер 30 или другое устройство может выполнять работу 350. Как проиллюстрировано в примере по фиг. 7, устройство может декодировать, из масштабируемого вкладывающего SEI-сообщения, множество элементов синтаксиса, которые идентифицируют множество рабочих точек, к которым применяется вложенное SEI-сообщение, инкапсулированное посредством масштабируемого вкладывающего SEI-сообщения (352). В некоторых примерах, устройство может декодировать, из масштабируемого вкладывающего SEI-сообщения,

элемент синтаксиса (например, `nesting_num_ops_minus1`), указывающий то, включает или нет масштабируемое вкладывающее SEI-сообщение в себя множество элементов синтаксиса, которые идентифицируют рабочие точки.

[0172] Кроме того, устройство может использовать один или более элементов синтаксиса вложенного SEI-сообщения для того, чтобы выполнять операцию относительно любой из рабочих точек, к которым применяется вложенное SEI-сообщение (354). Например, устройство может использовать элементы синтаксиса вложенного SEI-сообщения в тесте на соответствие битового потока, который определяет то, соответствует или нет какая-либо из рабочих точек, к которым применяется вложенное SEI-сообщение, стандарту кодирования видео, такому как HEVC. В другом примере, устройство может использовать элементы синтаксиса вложенного SEI-сообщения для того, чтобы выполнять тест на соответствие декодера.

[0173] Фиг. 8 является блок-схемой последовательности операций способа, иллюстрирующей примерную работу 400 видеокодера 20, в соответствии с одной или более технологий этого раскрытия сущности. Как проиллюстрировано в примере по фиг. 8, видеокодер 20 может включать в масштабируемое вкладывающее SEI-сообщение, инкапсулированное посредством SEI NAL-единицы, элемент синтаксиса (например, `default_or_applicable_flag`), который указывает то, является ли вложенное SEI-сообщение, инкапсулированное посредством масштабируемого вкладывающего SEI-сообщения, применимым к битовому субпотoku по умолчанию (402). Битовый субпоток по умолчанию является представлением рабочей точки для рабочей точки, заданной посредством идентификатора слоя, точно определяемого в заголовке NAL-единицы для SEI NAL-единицы, и временного идентификатора, точно определяемого в заголовке NAL-единицы. Первый элемент синтаксиса в заголовке NAL-единицы (например, `nuh_reserved_zero_6bits`) может указывать идентификатор слоя, а второй элемент синтаксиса в заголовке NAL-единицы (например, `nuh_reserved_temporal_id_plus1`) может указывать временной идентификатор.

[0174] В примере по фиг. 8, видеокодер 20 может включать в масштабируемое вкладывающее SEI-сообщение один или более дополнительных элементов синтаксиса, которые идентифицируют временной идентификатор дополнительной рабочей точки и максимальный идентификатор слоя дополнительной рабочей точки (404). Кроме того, видеокодер 20 может сигнализировать масштабируемое вкладывающее SEI-сообщение в битовом потоке (406). В некоторых примерах, элемент синтаксиса, который указывает то, является ли вложенное SEI-сообщение, инкапсулированное посредством масштабируемого вкладывающего SEI-сообщения, применимым к битовому субпотoku по умолчанию, может упоминаться в качестве первого элемента синтаксиса, и видеокодер 20 может включать второй элемент синтаксиса (например, `bitstream_subset_flag`) в масштабируемое вкладывающее SEI-сообщение. Второй элемент синтаксиса может указывать то, применяются ли вложенные SEI-сообщения, инкапсулированные посредством масштабируемого вкладывающего SEI-сообщения, к битовому субпотoku, извлеченному из битового потока, либо то, применяются ли вложенные SEI-сообщения к конкретным слоям и подслоям битового потока. Видеокодер 20 может включать в себя первый элемент синтаксиса только тогда, когда второй элемент синтаксиса указывает то, что вложенные SEI-сообщения применяются к битовому субпотoku, извлеченному из битового потока.

[0175] Фиг. 9 является блок-схемой последовательности операций способа, иллюстрирующей примерную работу 450 устройства, в соответствии с одной или более технологий этого раскрытия сущности. Видеокодер 20, видеокодер 30 или другое

устройство может выполнять работу 450. Как проиллюстрировано в примере по фиг. 9, устройство может определять, по меньшей мере, частично на основе первого элемента синтаксиса (например, `bitstream_subset_flag`) масштабируемого вкладывающего SEI-сообщения, то, применяется или нет вложенное SEI-сообщение, инкапсулированное посредством масштабируемого вкладывающего SEI-сообщения, к битовому субпотoku, извлеченному из битового потока (452). В ответ на определение того, что вложенное SEI-сообщение, инкапсулированное посредством масштабируемого вкладывающего SEI-сообщения, применяется к битовому субпотoku, извлеченному из битового потока ("Да" на 452), устройство может декодировать элемент синтаксиса рабочей точки по умолчанию (например, `default_or_applicable_flag`) в масштабируемом вкладывающем SEI-сообщении (454). Элемент синтаксиса рабочей точки по умолчанию может указывать, является или нет вложенное SEI-сообщение, инкапсулированное посредством масштабируемого вкладывающего SEI-сообщения, применимым к битовому субпотoku по умолчанию.

[0176] Битовый субпоток по умолчанию может быть представлением рабочей точки для рабочей точки, заданной посредством идентификатора слоя, точно определяемого в заголовке NAL-единицы для SEI NAL-единицы, и временного идентификатора, точно определяемого в заголовке NAL-единицы. В некоторых примерах, первый элемент синтаксиса в заголовке NAL-единицы (например, `nuh_reserved_zero_6bits`) указывает идентификатор слоя, а второй элемент синтаксиса в заголовке NAL-единицы (например, `nuh_reserved_temporal_id_plus1`) указывает временной идентификатор. Битовый субпоток по умолчанию может представлять собой поднабор битового потока, и битовый субпоток по умолчанию не включает в себя VCL NAL-единицы битового потока, которые имеют идентификаторы слоев, превышающие идентификатор слоя, указываемый посредством первого элемента синтаксиса заголовка NAL-единицы, или имеют временные идентификаторы, превышающие временной идентификатор, указываемый посредством второго элемента синтаксиса заголовка NAL-единицы.

[0177] Кроме того, устройство может определять, по меньшей мере, частично на основе элемента синтаксиса (например, `default_or_applicable_flag`) в масштабируемом вкладывающем SEI-сообщении, инкапсулированном посредством SEI NAL-единицы, то, является или нет вложенное SEI-сообщение, инкапсулированное посредством масштабируемого вкладывающего SEI-сообщения, применимым к битовому субпотoku по умолчанию битового потока (456). В некоторых примерах, масштабируемое вкладывающее SEI-сообщение инкапсулирует множество вложенных SEI-сообщений. В таких примерах, устройство может определять, на основе элемента синтаксиса (например, `default_or_applicable_flag`), то, является или нет каждое вложенное SEI-сообщение в масштабируемом вкладывающем SEI-сообщении применимым к битовому субпотoku по умолчанию.

[0178] Когда вложенное SEI-сообщение является применимым к битовому субпотoku по умолчанию ("Да" на 456), устройство может использовать вложенное SEI-сообщение при операции над битовым субпотком по умолчанию (458). Например, вложенное SEI-сообщение может включать в себя набор HRD-параметров. В этом примере, устройство может использовать HRD-параметры во вложенном SEI-сообщении при операции, которая тестирует то, соответствует или нет битовый субпоток по умолчанию стандарту кодирования видео, такому как HEVC. В другом примере, устройство может использовать HRD-параметры во вложенном SEI-сообщении в тесте на соответствие декодера. В другом примере, устройство может использовать вложенное SEI-сообщение при операции декодирования над битовым субпотком по умолчанию. В другом

примере, начальные задержки удаления из СРВ могут использоваться для того, чтобы направлять систему таким образом, чтобы устанавливать надлежащую начальную сквозную задержку, и времена DPB-вывода могут использоваться для того, чтобы извлекать временные метки RTP, когда видео транспортируется по RTP.

5 [0179] В противном случае, когда вложенное SEI-сообщение не является применимым в битовый субпоток по умолчанию ("Нет" на 456) или когда масштабируемое вкладывающее SEI-сообщение не применяется к битовому субпотoku, извлеченному из битового потока ("Нет" на 452), устройство не использует вложенное SEI-сообщение при операции над битовым субпотком по умолчанию (460). Например, устройство  
10 может определять, на основе одного или более дополнительных элементов синтаксиса в масштабируемом вкладывающем SEI-сообщении (например, `nesting_max_temporal_id_plus1[i]`, `nesting_op_idx[i]` и т.д.), временной идентификатор второй рабочей точки и максимальный идентификатор слоя второй рабочей точки. В этом примере, устройство может использовать вложенное SEI-сообщение при операции  
15 над дополнительным битовым субпотком, причем дополнительный битовый субпоток является представлением рабочей точки второй рабочей точки.

[0180] В одном или более примеров, описанные функции могут быть реализованы в аппаратных средствах, программном обеспечении, микропрограммном обеспечении или любой комбинации вышеозначенного. При реализации в программном обеспечении,  
20 функции могут быть сохранены или переданы, в качестве одной или более инструкций или кода, по считываемому компьютером носителю и выполнены посредством аппаратного процессора. Считываемые компьютером носители могут включать в себя считываемые компьютером носители данных, которые соответствуют материальному носителю, такие как носители данных, или среды связи, включающие в себя любой  
25 носитель, который упрощает перенос компьютерной программы из одного места в другое, например, согласно протоколу связи. Таким образом, считываемые компьютером носители, в общем, могут соответствовать (1) материальному считываемому компьютером носителю хранения данных, который является  
30 энергонезависимым, или (2) среде связи, такой как сигнал или несущая. Носители данных могут представлять собой любые доступные носители, к которым может осуществляться доступ посредством одного или более компьютеров или одного или более процессоров, с тем чтобы извлекать инструкции, код и/или структуры данных для реализации технологий, описанных в этом раскрытии сущности. Компьютерный программный продукт может включать в себя считываемый компьютером носитель.

35 [0181] В качестве примера, а не ограничения, эти считываемые компьютером носители данных могут содержать RAM, ROM, EEPROM, CD-ROM или другое устройство хранения на оптических дисках, устройство хранения на магнитных дисках или другие магнитные устройства хранения, флэш-память либо любой другой носитель, который может быть использован для того, чтобы сохранять требуемый программный код в форме  
40 инструкций или структур данных, и к которому можно осуществлять доступ посредством компьютера. Так же, любое подключение корректно называть считываемым компьютером носителем. Например, если инструкции передаются из веб-узла, сервера или другого удаленного источника с помощью коаксиального кабеля, оптоволоконного кабеля, "витой пары", цифровой абонентской линии (DSL) или беспроводных технологий, таких как инфракрасные, радиопередающие и микроволновые среды, то коаксиальный кабель, оптоволоконный кабель, "витая пара", DSL или беспроводные технологии, такие как инфракрасные, радиопередающие и микроволновые среды, включаются в  
45 определение носителя. Тем не менее, следует понимать, что считываемые компьютером

носители данных и носители данных не включают в себя соединения, несущие, сигналы или другие энергозависимые носители, а вместо этого направлены на энергонезависимые материальные носители данных. Диск (disk) и диск (disc) при использовании в данном документе включают в себя компакт-диск (CD), лазерный диск, оптический диск, универсальный цифровой диск (DVD), гибкий диск и диск Blu-Ray, при этом диски (disk) обычно воспроизводят данные магнитно, тогда как диски (disc) обычно воспроизводят данные оптически с помощью лазеров. Комбинации вышеперечисленного также следует включать в число считываемых компьютером носителей.

[0182] Инструкции могут выполняться посредством одного или более процессоров, например, одного или более процессоров цифровых сигналов (DSP), микропроцессоров общего назначения, специализированных интегральных схем (ASIC), программируемых пользователем вентильных матриц (FPGA) либо других эквивалентных интегральных или дискретных логических схем. Соответственно, термин "процессор" при использовании в данном документе может означать любую вышеуказанную структуру или другую структуру, подходящую для реализации технологий, описанных в данном документе. Помимо этого, в некоторых аспектах функциональность, описанная в данном документе, может быть предоставлена в пределах специализированных программных и/или аппаратных модулей, сконфигурированных с возможностью кодирования или декодирования либо встроенных в комбинированный кодек. Кроме того, технологии могут быть полностью реализованы в одной или более схем или логических элементов.

[0183] Технологии этого раскрытия сущности могут быть реализованы в широком спектре устройств или приборов, в том числе в беспроводном переносном телефоне, в интегральной схеме (IC) или в наборе IC (к примеру, в наборе микросхем). Различные компоненты, модули или блоки описываются в этом раскрытии сущности для того, чтобы подчеркивать функциональные аспекты устройств, сконфигурированных с возможностью осуществлять раскрытые технологии, но необязательно требуют реализации посредством различных аппаратных модулей. Наоборот, как описано выше, различные модули могут быть комбинированы в аппаратный модуль кодека или предоставлены посредством набора взаимодействующих аппаратных модулей, включающих в себя один или более процессоров, как описано выше, в сочетании с надлежащим программным обеспечением и/или микропрограммным обеспечением.

[0184] Описаны различные примеры. Эти и другие примеры находятся в пределах объема прилагаемой формулы изобретения.

#### (57) Формула изобретения

1. Способ обработки видеоданных, при этом способ содержит этапы, на которых:  
 - определяют, по меньшей мере, частично на основе элемента синтаксиса в сообщении с дополнительной улучшающей информацией (SEI), которое инкапсулировано посредством единицы уровня сетевой абстракции (NAL) SEI и инкапсулирует одно или более вложенных SEI-сообщений, то, являются или нет одно или более вложенных SEI-сообщений применимыми к битовому субпотoku по умолчанию из кодированного битового потока видео, который содержит кодируемые изображения видеоданных, при этом элемент синтаксиса не находится в каком-либо из одного или более вложенных SEI-сообщений и битовый субпоток по умолчанию является представлением рабочей точки для рабочей точки, заданной посредством идентификатора слоя, точно определенного в заголовке NAL-единицы для SEI NAL-единицы, и временного идентификатора, точно определенного в заголовке NAL-единицы; и

- когда одно или более вложенных SEI-сообщений применимы к битовому субпотoku по умолчанию, используют одно или более вложенных SEI-сообщений при операции над битовым субпотком по умолчанию.

2. Способ по п. 1, в котором первый элемент синтаксиса в заголовке NAL-единицы указывает идентификатор слоя, а второй элемент синтаксиса в заголовке NAL-единицы указывает временной идентификатор.

3. Способ по п. 2, в котором битовый субпоток по умолчанию представляет собой поднабор кодированного битового потока видео и битовый субпоток по умолчанию не включает в себя NAL-единицы слоя кодирования видео (VCL) кодированного битового потока видео, которые имеют идентификаторы слоев, превышающие идентификатор слоя, указываемый посредством первого элемента синтаксиса в заголовке NAL-единицы, или имеют временные идентификаторы, превышающие временной идентификатор, указываемый посредством второго элемента синтаксиса в заголовке NAL-единицы.

4. Способ по п. 1, в котором вложенное SEI-сообщение включает в себя набор параметров гипотетического эталонного декодера (HRD).

5. Способ по п. 4, в котором использование одного или более вложенных SEI-сообщений содержит этап, на котором используют HRD-параметры в конкретном вложенном SEI-сообщении при операции, которая тестирует то, соответствует или нет битовый субпоток по умолчанию стандарту кодирования видео.

6. Способ по п. 1, в котором использование одного или более вложенных SEI-сообщений содержит этап, на котором используют вложенное SEI-сообщение, инкапсулированное посредством SEI-сообщения, при операции декодирования над битовым субпотком по умолчанию.

7. Способ по п. 1, в котором:

- SEI-сообщение инкапсулирует множество вложенных SEI-сообщений, и
- определение того, является ли одно или более вложенных SEI-сообщений применимыми к битовому субпотку по умолчанию, содержит этап, на котором определяют, по меньшей мере, частично на основе элемента синтаксиса, то, является ли или нет каждое из вложенных SEI-сообщений применимым к битовому субпотку по умолчанию.

8. Способ по п. 1, в котором:

- элемент синтаксиса в SEI-сообщении является первым элементом синтаксиса, и
- способ дополнительно содержит этап, на котором определяют, по меньшей мере, частично на основе второго элемента синтаксиса в SEI-сообщении, указывающем, что одно или более вложенных SEI-сообщений, инкапсулированных SEI-сообщением, применяются к битовому субпотку, извлеченному из кодированного битового потока видео, то, что SEI-сообщение включает в себя первый элемент синтаксиса.

9. Способ по п. 1, при этом способ дополнительно содержит этапы, на которых:

- определяют, на основе одного или более дополнительных элементов синтаксиса в SEI-сообщении, временной идентификатор второй рабочей точки и максимальный идентификатор слоя второй рабочей точки; и
- используют одно или более вложенных SEI-сообщений при операции над дополнительным битовым субпотком, причем дополнительный битовый субпоток является представлением рабочей точки второй рабочей точки.

10. Устройство обработки видеоданных, содержащее:

носитель данных, сконфигурированный с возможностью хранения видеоданных; один или более процессоров, сконфигурированных с возможностью:



- определять, по меньшей мере, частично на основе элемента синтаксиса в сообщении с дополнительной улучшающей информацией (SEI), которое инкапсулировано посредством единицы уровня сетевой абстракции (NAL) SEI и инкапсулирует одно или более вложенных SEI-сообщений, то, являются или нет одно или более вложенных SEI-сообщений применимыми к битовому субпотoku по умолчанию из кодированного битового потока видео, который содержит кодируемые изображения видеоданных, при этом элемент синтаксиса не находится в каком-либо из одного или более вложенных SEI-сообщений и битовый субпоток по умолчанию является представлением рабочей точки для рабочей точки, заданной посредством идентификатора слоя, точно определенного в заголовке NAL-единицы для SEI NAL-единицы, и временного идентификатора, точно определенного в заголовке NAL-единицы; и

- когда одно или более вложенных SEI-сообщений применимы к битовому субпотoku по умолчанию, использовать одно или более вложенных SEI-сообщений при операции над битовым субпотком по умолчанию.

11. Устройство по п. 10, в котором первый элемент синтаксиса в заголовке NAL-единицы указывает идентификатор слоя, а второй элемент синтаксиса в заголовке NAL-единицы указывает временной идентификатор.

12. Устройство по п. 11, в котором битовый субпоток по умолчанию представляет собой поднабор кодированного битового потока видео и битовый субпоток по умолчанию не включает в себя NAL-единицы слоя кодирования видео (VCL) кодированного битового потока видео, которые имеют идентификаторы слоев, превышающие идентификатор слоя, указываемый посредством первого элемента синтаксиса в заголовке NAL-единицы, или имеют временные идентификаторы, превышающие временной идентификатор, указываемый посредством второго элемента синтаксиса в заголовке NAL-единицы.

13. Устройство по п. 10, в котором конкретное вложенное SEI-сообщение, инкапсулированное SEI-сообщением, включает в себя набор параметров гипотетического эталонного декодера (HRD).

14. Устройство по п. 13, в котором один или более процессоров сконфигурированы с возможностью использовать HRD-параметры в конкретном вложенном SEI-сообщении при операции, которая тестирует то, соответствует или нет битовый субпоток по умолчанию стандарту кодирования видео.

15. Устройство по п. 10, в котором один или более процессоров сконфигурированы с возможностью использовать вложенное SEI-сообщение, инкапсулированное SEI-сообщением, при операции декодирования над битовым субпотком по умолчанию.

16. Устройство по п. 10, в котором:

- SEI-сообщение инкапсулирует множество вложенных SEI-сообщений, и

- один или более процессоров сконфигурированы с возможностью определять, по меньшей мере, частично на основе элемента синтаксиса то, является или нет каждое из вложенных SEI-сообщений применимым к битовому субпотoku по умолчанию.

17. Устройство по п. 10, в котором:

- элемент синтаксиса в SEI-сообщении является первым элементом синтаксиса, и

- один или более процессоров дополнительно сконфигурированы с возможностью определять, по меньшей мере, частично на основе второго элемента синтаксиса в SEI-сообщении, указывающем, что одно или более вложенных SEI-сообщений, инкапсулированных SEI-сообщением, применяются к битовому субпотoku, извлеченному из кодированного битового потока видео, то, что SEI-сообщение включает в себя первый элемент синтаксиса.

18. Устройство по п. 10, в котором один или более процессоров дополнительно сконфигурированы с возможностью:

- определять, на основе одного или более дополнительных элементов синтаксиса в SEI-сообщении, временной идентификатор второй рабочей точки и максимальный идентификатор слоя второй рабочей точки; и

- использовать одно или более вложенных SEI-сообщений при операции над дополнительным битовым субпоток, причем дополнительный битовый субпоток является представлением рабочей точки второй рабочей точки.

19. Устройство обработки видеоданных, содержащее:

- средство для определения, по меньшей мере, частично на основе элемента синтаксиса в сообщении с дополнительной улучшающей информацией (SEI), которое инкапсулировано посредством единицы уровня сетевой абстракции (NAL) SEI и которое инкапсулирует одно или более вложенных SEI-сообщений, того, являются или нет одно или более вложенных SEI-сообщений применимыми к битовому субпоток по умолчанию из кодированного битового потока видео, который содержит кодируемые изображения видеоданных, при этом элемент синтаксиса не находится в каком-либо из одного или более вложенных SEI-сообщений и битовый субпоток по умолчанию является представлением рабочей точки для рабочей точки, заданной посредством идентификатора слоя, точно определенного в заголовке NAL-единицы для SEI NAL-единицы, и временного идентификатора, точно определенного в заголовке NAL-единицы, при этом SEI-сообщение содержит одно или более SEI-сообщений; и

- средство для использования, когда одно или более вложенных SEI-сообщений применимы к битовому субпоток по умолчанию, вложенных SEI-сообщений при операции над битовым субпоток по умолчанию.

20. Устройство по п. 19, в котором:

- первый элемент синтаксиса в заголовке NAL-единицы указывает идентификатор слоя, а второй элемент синтаксиса в заголовке NAL-единицы указывает временной идентификатор, и

- битовый субпоток по умолчанию представляет собой поднабор кодированного битового потока видео и битовый субпоток по умолчанию не включает в себя NAL-единицы слоя кодирования видео (VCL) кодированного битового потока видео, которые имеют идентификаторы слоев, превышающие идентификатор слоя, указываемый посредством первого элемента синтаксиса в заголовке NAL-единицы, или имеют временные идентификаторы, превышающие временной идентификатор, указываемый посредством второго элемента синтаксиса в заголовке NAL-единицы.

21. Устройство по п. 19, в котором:

- вложенное SEI-сообщение, инкапсулированное SEI-сообщением, включает в себя набор параметров гипотетического эталонного декодера (HRD), и

- устройство содержит средство для использования HRD-параметров во вложенном SEI-сообщении при операции, которая тестирует то, соответствует или нет битовый субпоток по умолчанию стандарту кодирования видео.

22. Считываемый компьютером носитель данных, который хранит инструкции, которые, при выполнении посредством одного или более процессоров устройства, конфигурируют устройство:

- определять, по меньшей мере, частично на основе элемента синтаксиса в сообщении с дополнительной улучшающей информацией (SEI), которое инкапсулировано посредством единицы уровня сетевой абстракции (NAL) SEI и инкапсулирует одно или более вложенных SEI-сообщений, то, являются или нет одно или более вложенных SEI-

сообщений применимыми к битовому субпотoku по умолчанию из кодированного битового потока видео, который содержит кодируемые изображения видеоданных, при этом элемент синтаксиса не находится в каком-либо из одного или более вложенных SEI-сообщений и битовый субпоток по умолчанию является представлением рабочей точки для рабочей точки, заданной посредством идентификатора слоя, точно определенного в заголовке NAL-единицы для SEI NAL-единицы, и временного идентификатора, точно определенного в заголовке NAL-единицы; и

- когда одно или более вложенных SEI-сообщений применимы к битовому субпотoku по умолчанию, использовать одно или более вложенных SEI-сообщений при операции над битовым субпотком по умолчанию.

23. Считываемый компьютером носитель данных по п. 22, в котором:

- первый элемент синтаксиса в заголовке NAL-единицы указывает идентификатор слоя, а второй элемент синтаксиса в заголовке NAL-единицы указывает временной идентификатор, и

- битовый субпоток по умолчанию представляет собой поднабор кодированного битового потока видео и битовый субпоток по умолчанию не включает в себя NAL-единицы слоя кодирования видео (VCL) кодированного битового потока видео, которые имеют идентификаторы слоев, превышающие идентификатор слоя, указываемый посредством первого элемента синтаксиса в заголовке NAL-единицы, или имеют временные идентификаторы, превышающие временной идентификатор, указываемый посредством второго элемента синтаксиса в заголовке NAL-единицы.

24. Считываемый компьютером носитель данных по п. 22, в котором:

- вложенное SEI-сообщение, инкапсулированное SEI-сообщением, включает в себя набор параметров гипотетического эталонного декодера (HRD), и

- инструкции дополнительно конфигурируют устройство использовать HRD-параметры во вложенном SEI-сообщении при операции, которая тестирует то, соответствует или нет битовый субпоток по умолчанию стандарту кодирования видео.

25. Способ кодирования видеоданных, при этом способ содержит этапы, на которых:

- включают в сообщение с дополнительной улучшающей информацией (SEI), которое инкапсулировано посредством единицы уровня сетевой абстракции (NAL) SEI и которое инкапсулирует одно или более вложенных SEI-сообщений, элемент синтаксиса, который указывает то, являются ли одно или более вложенных SEI-сообщений применимыми к битовому субпотoku по умолчанию из кодированного битового потока видео, который содержит кодируемые изображения видеоданных, при этом элемент синтаксиса не находится в каком-либо из одного или более вложенных SEI-сообщений и битовый субпоток по умолчанию является представлением рабочей точки для рабочей точки, заданной посредством идентификатора слоя, точно определенного в заголовке NAL-единицы для SEI NAL-единицы, и временного идентификатора, точно определенного в заголовке NAL-единицы; и

- сигнализируют SEI-сообщение в кодированном битовом потоке видео.

26. Способ по п. 25, в котором первый элемент синтаксиса в заголовке NAL-единицы указывает идентификатор слоя, а второй элемент синтаксиса в заголовке NAL-единицы указывает временной идентификатор.

27. Способ по п. 26, в котором битовый субпоток по умолчанию представляет собой поднабор кодированного битового потока видео и битовый субпоток по умолчанию не включает в себя NAL-единицы слоя кодирования видео (VCL) кодированного битового потока видео, которые имеют идентификаторы слоев, превышающие идентификатор слоя, указываемый посредством первого элемента синтаксиса в заголовке

NAL-единицы, или имеют временные идентификаторы, превышающие временной идентификатор, указываемый посредством второго элемента синтаксиса в заголовке NAL-единицы.

28. Способ по п. 25, в котором конкретное вложенное SEI-сообщение, инкапсулированное SEI-сообщением, включает в себя набор параметров гипотетического эталонного декодера (HRD).

29. Способ по п. 28, дополнительно содержащий этап, на котором используют HRD-параметры в конкретном вложенном SEI-сообщении при операции, которая тестирует то, соответствует или нет битовый субпоток по умолчанию стандарту кодирования видео.

30. Способ по п. 25, в котором:

- SEI-сообщение инкапсулирует множество вложенных SEI-сообщений, и
- элемент синтаксиса указывает то, является или нет каждое из вложенных SEI-сообщений применимым к битовому субпотoku по умолчанию.

31. Способ по п. 25, в котором:

- элемент синтаксиса в SEI-сообщении является первым элементом синтаксиса в SEI-сообщении,

- способ дополнительно содержит этап, на котором включают второй элемент синтаксиса в SEI-сообщение, причем второй элемент синтаксиса указывает то, применяются или нет вложенные SEI-сообщения, инкапсулированные SEI-сообщением, к битовому субпотoku, извлеченному из кодированного битового потока видео, либо то, применяются или нет вложенные SEI-сообщения к конкретным слоям и подслоям кодированного битового потока видео, и

- SEI-сообщение включает в себя первый элемент синтаксиса только тогда, когда второй элемент синтаксиса указывает то, что вложенные SEI-сообщения применяются к битовому субпотoku, извлеченному из кодированного битового потока видео.

32. Способ по п. 25, в котором:

- рабочая точка, идентифицированная посредством одного или более элементов синтаксиса заголовка NAL-единицы для SEI NAL-единицы, представляет собой первую рабочую точку, и

- способ дополнительно содержит этап, на котором включают в SEI-сообщение один или более дополнительных элементов синтаксиса, которые идентифицируют временной идентификатор второй рабочей точки и максимальный идентификатор слоя второй рабочей точки.

33. Устройство кодирования видео, содержащее:

- носитель данных, сконфигурированный для хранения видеоданных; и
- один или более процессоров, сконфигурированных с возможностью:

- включать в сообщение с дополнительной улучшающей информацией (SEI), которое инкапсулировано посредством единицы уровня сетевой абстракции (NAL) SEI и

- инкапсулирует одно или более вложенных SEI-сообщений, элемент синтаксиса, который указывает то, являются или нет одно или более вложенных SEI-сообщений применимыми к битовому субпотoku по умолчанию из кодированного битового потока видео, который содержит кодируемые изображения видеоданных, при этом элемент синтаксиса не находится в каком-либо из одного или более вложенных SEI-сообщений и битовый субпоток по умолчанию является представлением рабочей точки для рабочей точки, заданной посредством идентификатора слоя, точно определенного в заголовке NAL-единицы для SEI NAL-единицы, и временного идентификатора, точно определенного в заголовке NAL-единицы; и

- сигнализировать SEI-сообщение в кодированном битовом потоке видео.

34. Устройство кодирования видео по п. 33, в котором первый элемент синтаксиса в заголовке NAL-единицы указывает идентификатор слоя, а второй элемент синтаксиса в заголовке NAL-единицы указывает временной идентификатор.

35. Устройство кодирования видео по п. 34, в котором битовый субпоток по умолчанию представляет собой поднабор кодированного битового потока видео и битовый субпоток по умолчанию не включает в себя NAL-единицы слоя кодирования видео (VCL) кодированного битового потока видео, которые имеют идентификаторы слоев, превышающие идентификатор слоя, указываемый посредством первого элемента синтаксиса в заголовке NAL-единицы, или имеют временные идентификаторы, превышающие временной идентификатор, указываемый посредством второго элемента синтаксиса в заголовке NAL-единицы.

36. Устройство кодирования видео по п. 33, в котором конкретное вложенное SEI-сообщение, инкапсулированное SEI-сообщением, включает в себя набор параметров гипотетического эталонного декодера (HRD).

37. Устройство кодирования видео по п. 36, в котором один или более процессоров дополнительно сконфигурированы с возможностью использовать HRD-параметры в конкретном вложенном SEI-сообщении при операции, которая тестирует то, соответствует или нет битовый субпоток по умолчанию стандарту кодирования видео.

38. Устройство кодирования видео по п. 33, в котором:

- SEI-сообщение инкапсулирует множество вложенных SEI-сообщений, и
- элемент синтаксиса указывает то, является ли нет каждое из вложенных SEI-сообщений применимым к битовому субпотoku по умолчанию.

39. Устройство кодирования видео по п. 33, в котором:

- элемент синтаксиса в SEI-сообщении является первым элементом синтаксиса в SEI-сообщении,

- один или более процессоров дополнительно сконфигурированы с возможностью включать второй элемент синтаксиса в SEI-сообщение, причем второй элемент синтаксиса указывает то, применяются ли нет вложенные SEI-сообщения,

инкапсулированные SEI-сообщением, к битовому субпотoku, извлеченному из кодированного битового потока видео, либо то, применяются ли нет вложенные SEI-сообщения к конкретным слоям и подслоям кодированного битового потока видео, и

- SEI-сообщение включает в себя первый элемент синтаксиса только тогда, когда второй элемент синтаксиса указывает то, что вложенные SEI-сообщения применяются к битовому субпотoku, извлеченному из кодированного битового потока видео.

40. Устройство кодирования видео по п. 33, в котором:

- рабочая точка, идентифицированная посредством одного или более элементов синтаксиса заголовка NAL-единицы для SEI NAL-единицы, представляет собой первую рабочую точку, и

- один или более процессоров дополнительно сконфигурированы с возможностью включать в SEI-сообщение один или более дополнительных элементов синтаксиса, которые идентифицируют временной идентификатор второй рабочей точки и максимальный идентификатор слоя второй рабочей точки.

41. Устройство кодирования видео, содержащее:

- средство для включения, в сообщение с дополнительной улучшающей информацией (SEI), которое инкапсулировано посредством единицы уровня сетевой абстракции (NAL) SEI и инкапсулирует одно или более вложенных SEI-сообщений, элемента синтаксиса, который указывает то, являются ли нет одно или более вложенных SEI-сообщений

применимыми к битовому субпотoku по умолчанию из кодированного битового потока видео, который содержит кодируемые изображения видеоданных, при этом элемент синтаксиса не находится в каком-либо из одного или более вложенных SEI-сообщений и битовый субпоток по умолчанию является представлением рабочей точки для рабочей точки, заданной посредством идентификатора слоя, точно определенного в заголовке NAL-единицы для SEI NAL-единицы, и временного идентификатора, точно определенного в заголовке NAL-единицы; и

- средство для сигнализации SEI-сообщения в кодированном битовом потоке видео.

42. Устройство кодирования видео по п. 41, в котором:

- первый элемент синтаксиса в заголовке NAL-единицы указывает идентификатор слоя, а второй элемент синтаксиса в заголовке NAL-единицы указывает временной идентификатор, и

- битовый субпоток по умолчанию представляет собой поднабор кодированного битового потока видео и битовый субпоток по умолчанию не включает в себя NAL-единицы слоя кодирования видео (VCL) кодированного битового потока видео, которые имеют идентификаторы слоев, превышающие идентификатор слоя, указываемый посредством первого элемента синтаксиса в заголовке NAL-единицы, или имеют временные идентификаторы, превышающие временной идентификатор, указываемый посредством второго элемента синтаксиса в заголовке NAL-единицы.

43. Устройство кодирования видео по п. 41, в котором:

- вложенное SEI-сообщение, инкапсулированное SEI-сообщением, включает в себя набор параметров гипотетического эталонного декодера (HRD), и

- устройство кодирования видео содержит средство для использования HRD-параметров во вложенном SEI-сообщении при операции, которая тестирует то,

соответствует или нет битовый субпоток по умолчанию стандарту кодирования видео.

44. Считываемый компьютером носитель данных, который хранит инструкции, которые, при выполнении посредством устройства кодирования видео, конфигурируют устройство кодирования видео с возможностью:

- включать в сообщение с дополнительной улучшающей информацией (SEI), которое инкапсулировано посредством единицы уровня сетевой абстракции (NAL) SEI и инкапсулирует одно или более вложенных SEI-сообщений, элемент синтаксиса, который указывает то, являются ли одно или более вложенных SEI-сообщений применимыми к битовому субпотoku по умолчанию из кодированного битового потока видео, который содержит кодируемые изображения видеоданных, при этом элемент синтаксиса не находится в каком-либо из вложенных SEI-сообщений и битовый субпоток по умолчанию является представлением рабочей точки для рабочей точки, заданной посредством идентификатора слоя, точно определенного в заголовке NAL-единицы для SEI NAL-единицы, и временного идентификатора, точно определенного в заголовке NAL-единицы; и

- сигнализировать SEI-сообщение в кодированном битовом потоке видео.

45. Считываемый компьютером носитель данных по п. 44, в котором:

- первый элемент синтаксиса в заголовке NAL-единицы указывает идентификатор слоя, а второй элемент синтаксиса в заголовке NAL-единицы указывает временной идентификатор, и

- битовый субпоток по умолчанию представляет собой поднабор кодированного битового потока видео и битовый субпоток по умолчанию не включает в себя NAL-единицы слоя кодирования видео (VCL) кодированного битового потока видео, которые имеют идентификаторы слоев, превышающие идентификатор слоя, указываемый

посредством первого элемента синтаксиса в заголовке NAL-единицы, или имеют временные идентификаторы, превышающие временной идентификатор, указываемый посредством второго элемента синтаксиса в заголовке NAL-единицы.

46. Считываемый компьютером носитель данных по п. 44, в котором:

- 5       - конкретное вложенное SEI-сообщение, инкапсулированное SEI-сообщением, включает в себя набор параметров гипотетического эталонного декодера (HRD), и
- инструкции дополнительно конфигурируют устройство кодирования видео с возможностью использовать HRD-параметры в конкретном вложенном SEI-сообщении при операции, которая тестирует то, соответствует или нет битовый субпоток по
- 10       умолчанию стандарту кодирования видео.

15

20

25

30

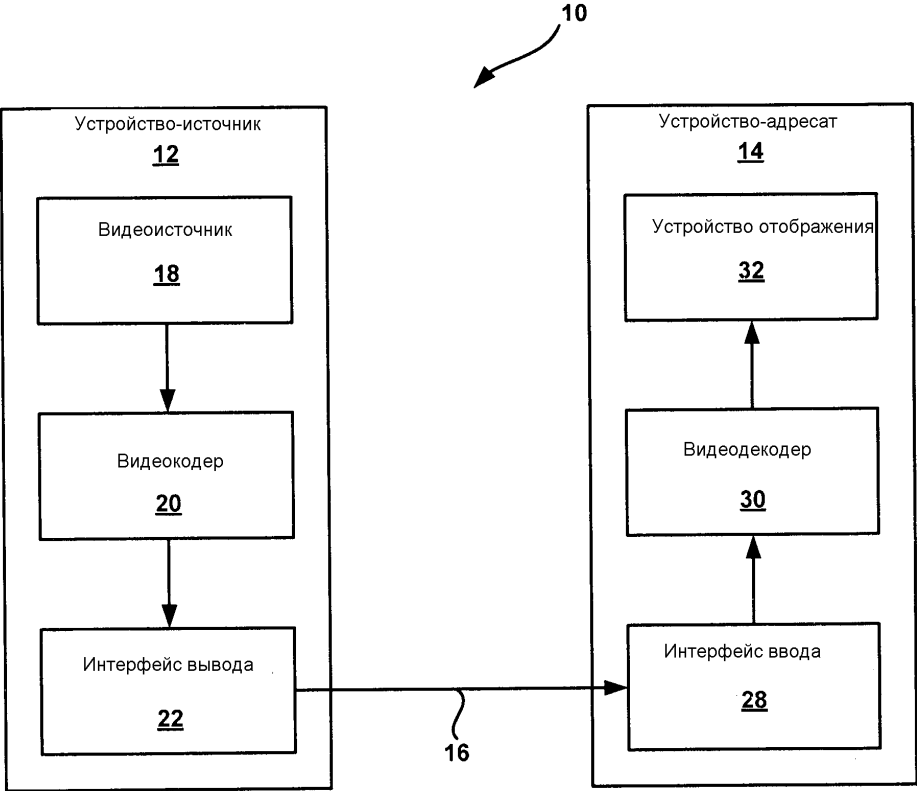
35

40

45

1

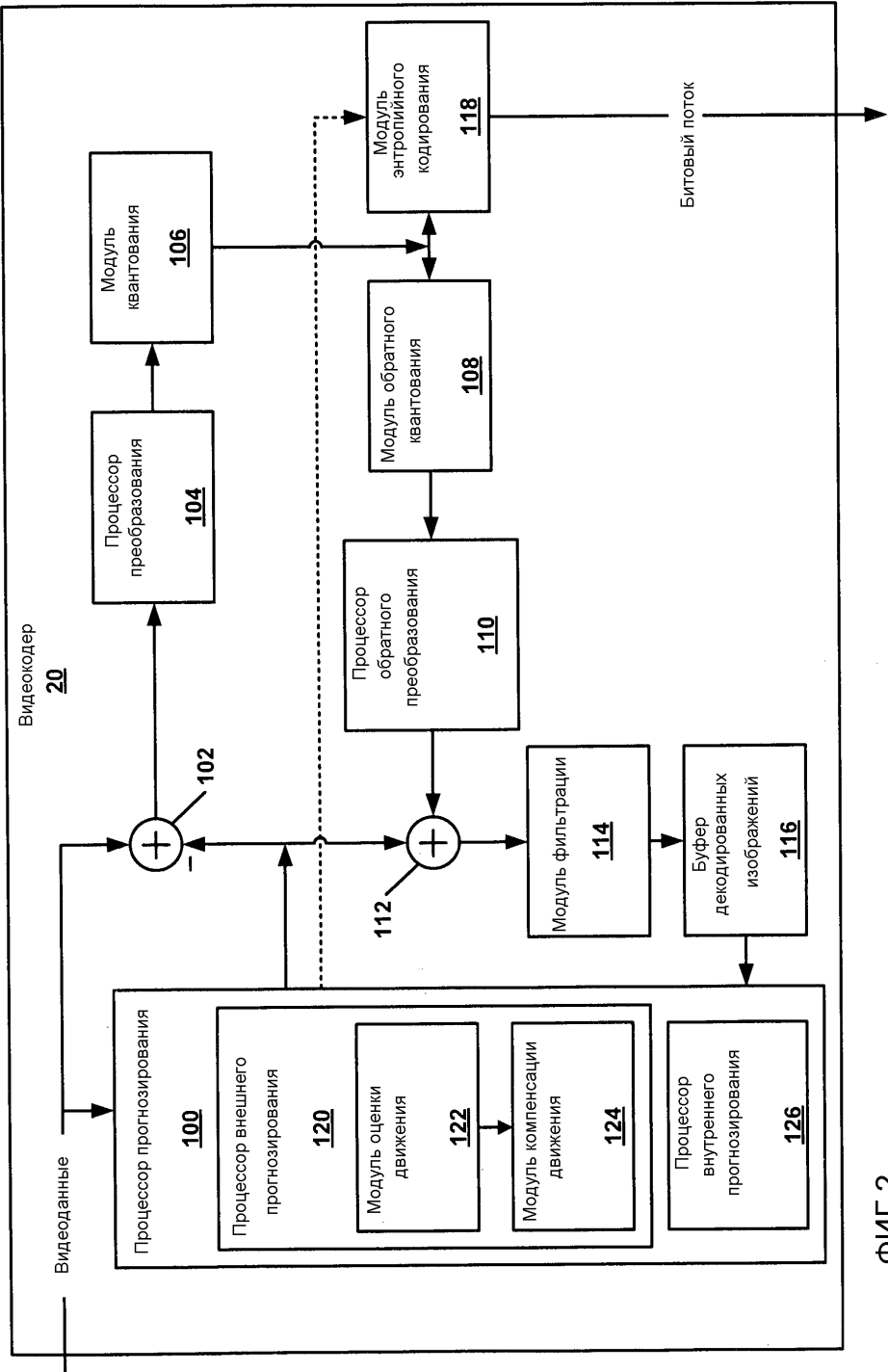
1/9

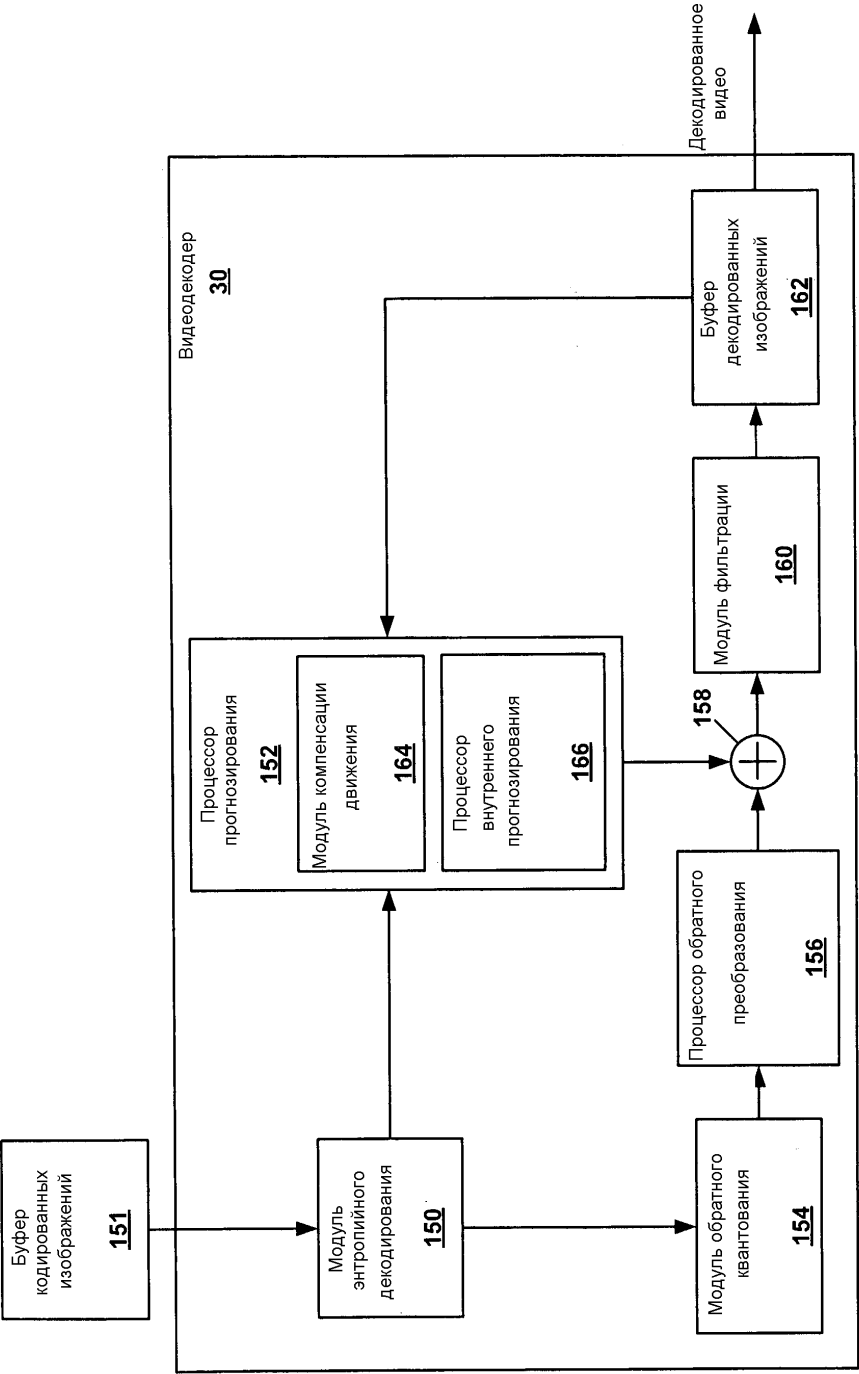


ФИГ.1

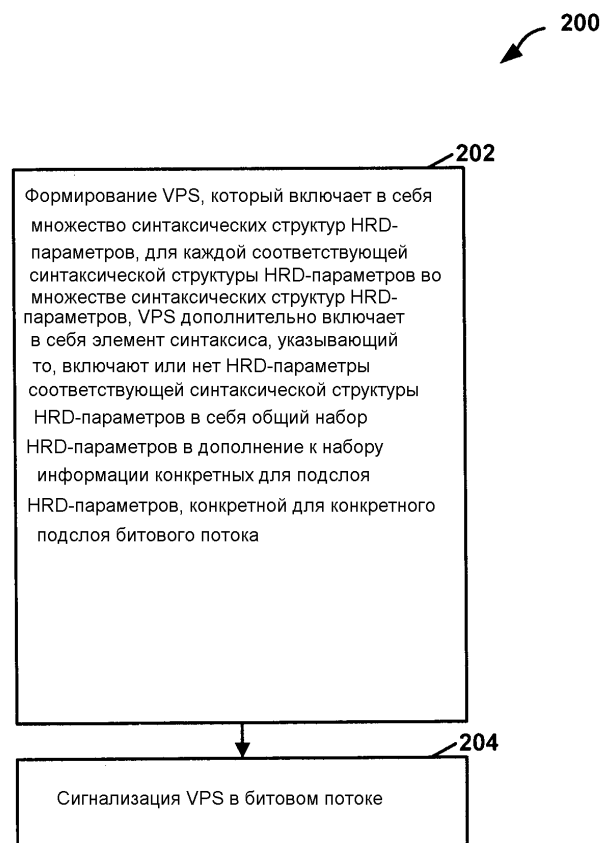
2





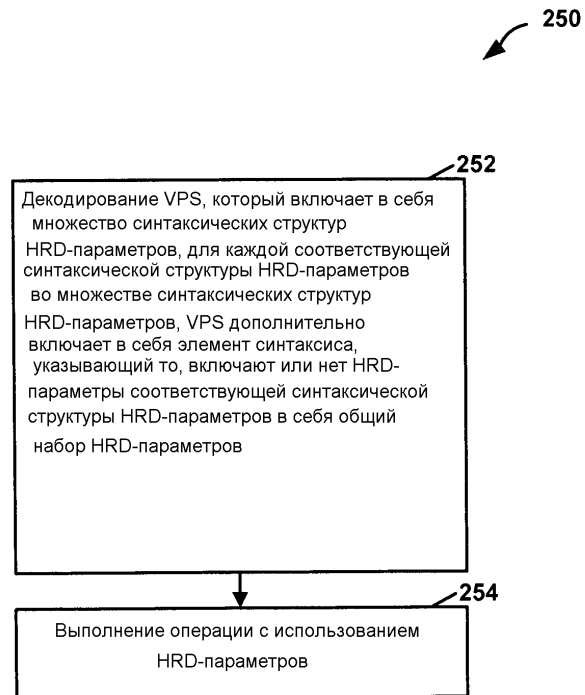


ФИГ.3



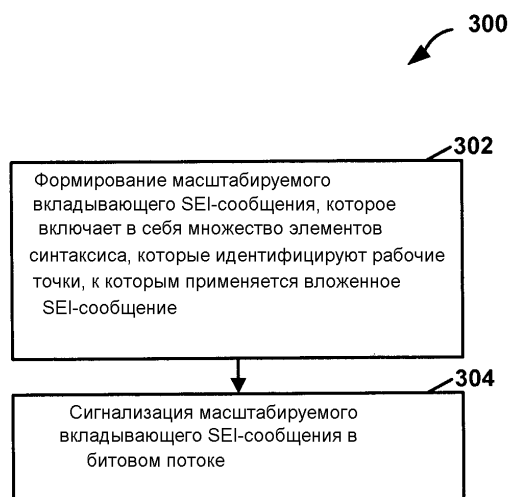
ФИГ.4

5/9

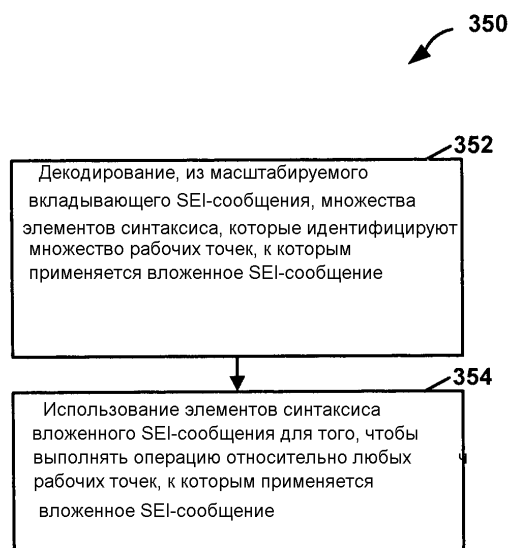


ФИГ.5

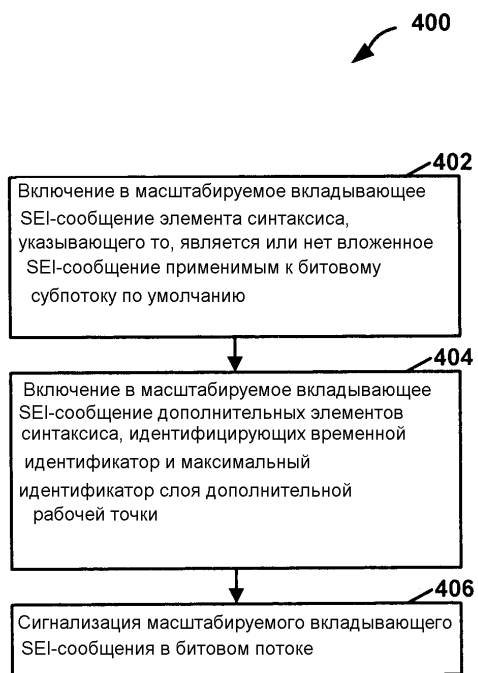
6/9



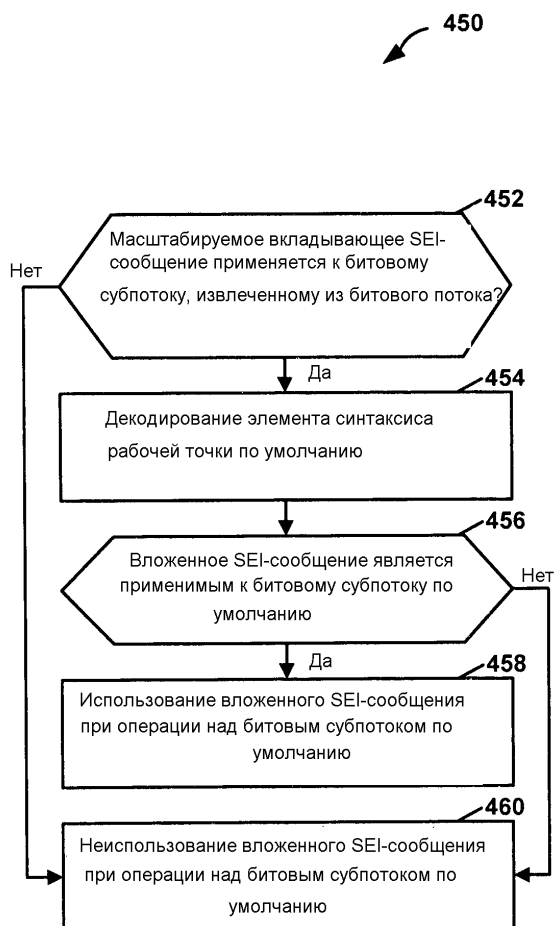
ФИГ.6



ФИГ.7



ФИГ.8



ФИГ.9