



**ФЕДЕРАЛЬНАЯ СЛУЖБА  
ПО ИНТЕЛЛЕКТУАЛЬНОЙ СОБСТВЕННОСТИ**

**(12) ОПИСАНИЕ ИЗОБРЕТЕНИЯ К ПАТЕНТУ**

(21)(22) Заявка: 2013138354/08, 17.01.2012

(24) Дата начала отсчета срока действия патента:  
17.01.2012

Приоритет(ы):

(30) Конвенционный приоритет:  
18.01.2011 US 61/433,803;  
06.07.2011 EP 11172791.3

(43) Дата публикации заявки: 27.02.2015 Бюл. № 6

(45) Опубликовано: 20.02.2016 Бюл. № 5

(56) Список документов, цитированных в отчете о поиске: US 2007/0201514 A1, 30.08.2007. RU 2251750 C2, 10.05.2005. RU 2325046 C2, 20.05.2008. US 7783494 B2, 24.08.2010. US 7519538 B2, 14.04.2009. EP 1396843 A1, 10.03.2004.

(85) Дата начала рассмотрения заявки РСТ на национальной фазе: 19.08.2013

(86) Заявка РСТ:  
EP 2012/050613 (17.01.2012)

(87) Публикация заявки РСТ:  
WO 2012/098098 (26.07.2012)

Адрес для переписки:

129090, Москва, ул. Б. Спасская, 25, строение 3,  
ООО "Юридическая фирма Городиский и  
Партнеры"

(72) Автор(ы):

**КУНТЦ Ахим (DE),  
ДИШ Саша (DE),  
БЯКСТРЕМ Том (DE)**

(73) Патентообладатель(и):

**ФРАУНХОФЕР-ГЕЗЕЛЛЬШАФТ ЦУР  
ФЕРДЕРУНГ ДЕР АНГЕВАНДТЕН  
ФОРШУНГ Е.Ф. (DE)**

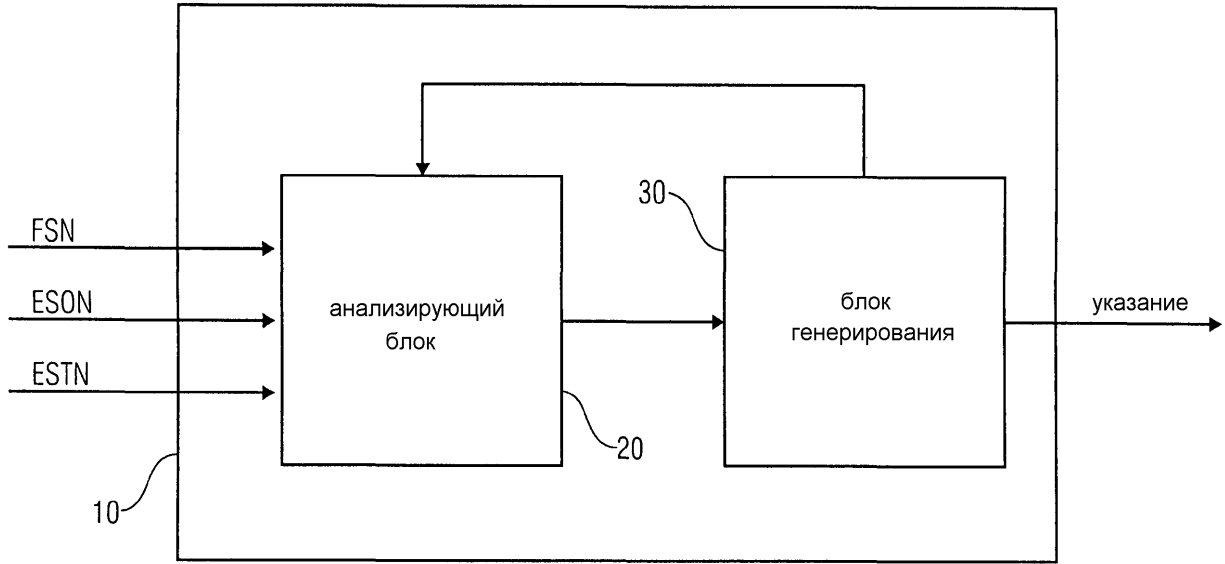
**(54) КОДИРОВАНИЕ И ДЕКОДИРОВАНИЕ ПОЗИЦИЙ СЛОТОВ С СОБЫТИЯМИ В КАДРЕ АУДИОСИГНАЛА**

(57) Реферат:

Изобретение относится к области обработки аудиосигнала и аудиокодирования. Технический результат - повышение точности кодирования и декодирования. Раскрыты устройство для декодирования, устройство для кодирования, способ декодирования и способ кодирования позиций слотов, содержащих события в кадре аудиосигнала, и соответствующие программы и кодированные сигналы, при этом устройство для декодирования содержит: анализирующий блок

для анализа числа слотов кадра, указывающего общее число слотов кадра аудиосигнала, числа слотов с событиями, указывающего число слотов, содержащих события кадра аудиосигнала, и числа состояний события и блок генерирования для генерирования указания множества позиций слотов, содержащих события в кадре аудиосигнала, с использованием числа слотов кадра, числа слотов с событиями и числа состояний события. 6 н. и 11 з.п. ф-лы, 25 ил., 6

табл.



Фиг.9А

RU 2575393 C2

RU 2575393 C2



FEDERAL SERVICE  
FOR INTELLECTUAL PROPERTY

(12) **ABSTRACT OF INVENTION**

(21)(22) Application: 2013138354/08, 17.01.2012

(24) Effective date for property rights:  
17.01.2012

Priority:

(30) Convention priority:  
18.01.2011 US 61/433,803;  
06.07.2011 EP 11172791.3

(43) Application published: 27.02.2015 Bull. № 6

(45) Date of publication: 20.02.2016 Bull. № 5

(85) Commencement of national phase: 19.08.2013

(86) PCT application:  
EP 2012/050613 (17.01.2012)

(87) PCT publication:  
WO 2012/098098 (26.07.2012)

Mail address:  
129090, Moskva, ul. B. Spasskaja, 25, stroenie 3,  
OOO "Juridicheskaja firma Gorodisskij i Partnery"

(72) Inventor(s):  
**KUNTTs Akhim (DE),**  
**DISH Sasha (DE),**  
**BJaKSTREM Tom (DE)**

(73) Proprietor(s):  
**FRAUNKhOFER-GEZELL'ShAFT T'SUR**  
**FERDERUNG DER ANGEVANDTEN**  
**FORShUNG E.F. (DE)**

(54) **ENCODING AND DECODING OF SLOT POSITIONS WITH EVENTS IN AUDIO SIGNAL FRAME**

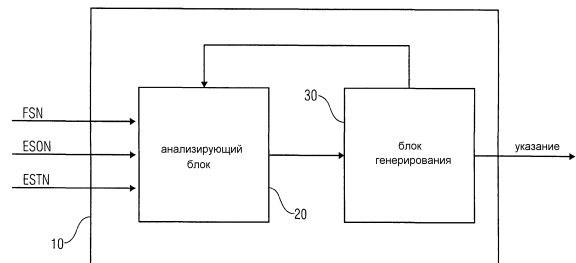
(57) Abstract:

FIELD: physics, acoustics.

SUBSTANCE: invention relates to audio signal processing and audio encoding. Disclosed is an apparatus for decoding, an apparatus for encoding, a method for decoding and a method for encoding positions of slots comprising events in an audio signal frame and respective programs and encoded signals, wherein the apparatus for decoding comprises: an analysing unit for analysing a frame slots number indicating the total of slots of the audio signal frame, an event slots number indicating the number of slots comprising the events of the audio signal frame, and an event state number, and a generating unit for generating an indication of a plurality of positions of

slots comprising the events in the audio signal frame using the frame slots number, the event slots number and the event state number.

EFFECT: high accuracy of encoding and decoding.  
17 cl, 25 dwg, 6 tbl



Фиг.9А

RU 2 575 393 C2

RU 2 575 393 C2

Настоящее изобретение относится к области обработки аудио и аудиокодирования, в частности к кодированию и декодированию позиций слотов с событиями в кадре аудиосигнала.

5 Обработка аудио и/или аудиокодирование продвинулись во многих отношениях. В частности, применения пространственного аудио становятся все более важными. Обработка аудиосигнала часто используется для декорреляции или воспроизведения сигналов. Более того, декорреляция и воспроизведение сигналов используется в процессе  
10 повышающего микширования моно-стерео (из моно в стерео), повышающего микширования из моно/стерео в многоканальный формат, для искусственной реверберации, расширения стерео или взаимодействующего с пользователем микширования/воспроизведения.

Некоторые системы обработки аудиосигнала используют декорреляторы. Важным примером является применение декорреляции сигналов в параметрических пространственных аудиодекодерах для восстановления конкретных свойств  
15 декорреляции между двумя или более сигналами, которые реконструируются из одного или нескольких сигналов, полученных понижающим микшированием. Применение декорреляторов значительно улучшает воспринимаемое качество выходного сигнала, например, при сравнении со стерео на основе интенсивности сигнала. Конкретно, использование декорреляторов обеспечивает возможность правильного синтеза  
20 пространственного звука с широкой фонограммой, несколькими одновременно действующими звуковыми объектами и/или окружением. Однако также известно, что декорреляторы вносят артефакты, такие как изменения во временной структуре сигнала, тембре и т.д.

Другими примерами декорреляторов при обработке аудио являются, например,  
25 генерирование искусственной реверберации для изменения пространственного впечатления или использование декорреляторов в многоканальных акустических системах эхоподавления для улучшения характера сходимости.

Одной важной схемой пространственного аудиокодирования является параметрическое стерео (PS). На фиг.1 проиллюстрирована структура моно-стерео  
30 декодера. Одиночный декоррелятор генерирует декоррелированный сигнал D ("обработанный" сигнал) из входного моносигнала M ("необработанного" сигнала). Декоррелированный сигнал D затем подается в микшер вместе с сигналом M. Затем микшер применяет матрицу H микширования к входным сигналам M и D для генерирования выходных сигналов L и R. Коэффициенты в матрице H микширования  
35 могут быть фиксированными, зависимыми от сигнала или управляемыми пользователем.

В качестве альтернативы, матрица микширования управляется дополнительной информацией, которая передается вместе с понижающим микшированием и содержит параметрическое описание того, как осуществить повышающее микширование сигналов  
40 понижающего микширования для формирования желаемого многоканального выходного сигнала. Пространственная дополнительная информация обычно генерируется во время процесса понижающего микширования до моно в соответствующем кодере сигнала.

Пространственное аудиокодирование, как описано выше, широко применяется, например, в параметрическом стерео. Типичная структура декодера параметрического  
45 стерео показана на фиг.2. На фиг.2 декорреляция выполняется в области преобразования. Пространственные параметры могут быть модифицированы пользователем или дополнительными инструментами, например, посредством пост-обработки для бинаурального воспроизведения/представления. В этом случае параметры повышающего

микширования объединяются с параметрами из бинауральных фильтров, чтобы вычислить входные параметры для матрицы микширования.

Выходной сигнал L/R матрицы H микширования вычисляется из входного моно сигнала M и декоррелированного сигнала D.

5

$$\begin{bmatrix} L \\ R \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \end{bmatrix} \begin{bmatrix} M \\ D \end{bmatrix}$$

10

В матрице микширования величиной декоррелированного звука, поданного на выход, управляют на основании переданных параметров, например межканальных разностей уровней частот (ILD), межканальной корреляции/когерентности (ICC) и/или фиксированных или заданных пользователем настроек.

15

Концептуально, выходной сигнал выхода D декоррелятора заменяет остаточный сигнал, который в идеале обеспечил бы возможность идеального декодирования первоначальных L/R сигналов. Использование выхода D декоррелятора вместо остаточного сигнала в повышающем микшере приводит к сбережению битовой скорости, которая иначе потребовалась бы для передачи остаточного сигнала. Целью декоррелятора, таким образом, является сгенерировать сигнал D из моно сигнала M, который показывает аналогичные свойства, как и остаточный сигнал, который заменяется посредством D. Сделана ссылка на документ:

20

[1] J. Breebaart, S. van de Par, A. Kohlrausch, E. Schuijers, "High-Quality Parametric Spatial

Audio Coding at Low Bitrates" in Proceedings of the AES 116<sup>th</sup> Convention, Berlin, Preprint 6072, May 2004.

25

Рассматривая MPEG Surround (MPS), структуры, аналогичные PS, называемые блоками преобразования из одного канала в два (ОТТ-блоки), используются в деревьях пространственного аудиодекодирования. Это может выглядеть как обобщение концепции повышающего микширования моно-стерео для схем многоканального пространственного аудиокодирования/аудиодекодирования. В MPS также существуют системы повышающего микширования из двух каналов в три (ТТТ-блоки), которые могут применять декорреляторы в зависимости от ТТТ-режима работы. Подробности описаны в документе:

30

[2] J. Herre, K. Kjolring, J. Breebaart, et al., "MPEG surround - the ISO/MPEG standard for

efficient and compatible multi-channel audio coding", in Proceedings of the 122<sup>th</sup> AES Convention, Vienna, Austria, May 2007.

35

Что касается направленного аудиокодирования (DirAC), DirAC относится к схеме параметрического кодирования звукового поля, которая не ограничена фиксированным числом каналов выходного аудиосигнала с фиксированными позициями громкоговорителей. DirAC применяет декорреляторы в блоке воспроизведения DirAC, т.е. в пространственном аудиодекодере для синтеза некогерентных составляющих звуковых полей. Направленное аудиокодирование дополнительно описано в:

40

[3] Pulkki, Ville: "Spatial Sound Reproduction with Directional Audio Coding", в J. Audio Eng. Soc., Vol. 55, № 6, 2007.

45

Касательно декорреляторов предшествующего уровня техники, сделана ссылка на документы:

[4] ISO/IEC International Standard "Information Technology - MPEG audio technologies - Part1: MPEG Surround", ISO/IEC 23003-1:2007.

[5] J. Engdegard, H. Purnhagen, J. Roden, L. Liljeryd, "Synthetic Ambience in Parametric Stereo Coding" in Proceedings of the AES 116<sup>th</sup> Convention, Preprint, May 2004.

Решетчатые частотнонезависимые ИР-структуры используются в качестве декорреляторов в пространственных аудиодекодерах, таких как MPS [2, 4]. Другие декорреляторы предшествующего уровня техники (потенциально частотнозависимые) применяют задержки к декоррелированным сигналам или свертывают входные сигналы, например, с экспоненциально затухающими шумовыми выбросами. Для обзора декорреляторов предшествующего уровня техники для пространственных аудиосистем повышающего микширования сделана ссылка на документ [5]: "Synthetic Ambience in Parametric Stereo Coding".

В общем, известно, что стерео или многоканальные подобные аплодисментам сигналы, кодированные/декодированные в параметрических пространственных аудиодекодерах, приводят к уменьшенному качеству сигнала. Подобные аплодисментам сигналы характеризуются содержанием достаточно частых примесей транзиентов с разных направлений. Примерами для таких сигналов являются аплодисменты, звук дождя, скачущие лошади и т.д. Подобные аплодисментам сигналы часто также содержат составляющие звука от удаленных источников звука, которые ощутимо сливаются с шумоподобным, сглаженным фоновым звуковым полем.

Решетчатые частотнонезависимые структуры, используемые в пространственных аудиодекодерах, таких как MPEG Surround, действуют как генераторы искусственной реверберации и поэтому хорошо подходят для генерирования однородных, сглаженных, шумоподобных, инверсивных звуков (таких как реверберационные хвосты в помещении). Однако они являются примерами звуковых полей с неоднородной пространственно-временной структурой, которые все же создают погружение слушателя: одним заметным примером являются подобные аплодисментам звуковые поля, которые создают охват слушателя не только посредством однородных шумоподобных полей, но также посредством достаточно частых последовательностей одиночных хлопков с разных направлений. Поэтому неоднородная составляющая звуковых полей аплодисментов может характеризоваться пространственно распределенной примесью транзиентов. Эти отличимые хлопки вовсе не являются однородными, сглаженными и шумоподобными.

Из-за их подобного реверберации поведения, решетчатые частотнонезависимые декорреляторы не способны генерировать иммерсивные звуковые поля с характеристиками, например, аплодисментов. Вместо этого, при применении к подобным аплодисментам сигналам они имеют склонность к временному размытию транзиентов в сигнале. Нежелательным результатом является шумоподобное иммерсивное звуковое поле без отличительной пространственно-временной структуры подобных аплодисментам звуковых полей. Кроме того, события транзиентов, такие как одиночные хлопки руками, могут вызывать звенящие артефакты фильтров декоррелятора.

USAC (унифицированное речевое и аудиокодирование) является стандартом аудиокодирования для кодирования речи и аудио и их смешения с разными битовыми скоростями.

Воспринимаемое качество USAC может быть дополнительно улучшено при стереокодировании аплодисментов и подобных аплодисментам звуков с битовыми скоростями в пределах 32 кбит/с, когда применимы методы параметрического стереокодирования. Элементы аплодисментов, кодированные с помощью USAC, имеют склонность показывать узкую звуковую сцену и недостаток охвата, если внутри кодека не применяется выделенная обработка аплодисментов. Во многом методы

стереокодирования USAC и их ограничения были унаследованы от MPEG Surround (MPS). Однако USAC предлагает выделенную адаптацию для требования правильной обработки аплодисментов. Вышеупомянутая адаптация называется управляющим транзидентами декоррелятором (TSD) и является вариантом осуществления этого изобретения.

Сигналы аплодисментов можно представить составленными из одиночных, отличимых близких хлопков, временно разделенных несколькими миллисекундами, и наложенного шумоподобного окружения, возникающего из-за очень частых отдаленных хлопков. В параметрическом стереокодировании с разумной скоростью дополнительной информации детализация наборов пространственных параметров (межканальной разности уровней частот, межканальной корреляции и т.д.) является чересчур низкой, чтобы гарантировать достаточное пространственное перераспределение одиночных хлопков, приводя к недостатку охвата. Дополнительно, хлопки подвергаются обработке решетчатым частотнонезависимым декоррелятором. Это неизбежно порождает временную дисперсию транзиевтов и дополнительно уменьшает субъективное качество.

Использование управляющего транзидентами декоррелятора (TSD) внутри USAC-декодера приводит к модификации MPS-обработки. Основополагающая идея такого подхода состоит в решении проблемы декорреляции аплодисментов, как следует ниже:

- Отделить транзиевты в QMF-области до решетчатого частотнонезависимого декоррелятора, т.е.: разделить входной сигнал декоррелятора на поток  $s_2$  с транзиевтами и поток  $s_1$  без транзиевтов.

- Подать поток с транзиевтами на другой управляемый параметрами декоррелятор, который хорошо подходит для примесей транзиевтов.

- Подать поток без транзиевтов на частотнонезависимый MPS-декоррелятор.

- Сложить выходные сигналы обоих декорреляторов  $D_1$  и  $D_2$ , чтобы получить декоррелированный сигнал  $D$ .

На фиг.3 проиллюстрирована конфигурация преобразования из одного канала в два (ОТТ) внутри USAC-декодера. U-образный блок обработки транзиевтов по фиг.3 содержит параллельный тракт сигнала, предлагаемый для обработки транзиевтов.

Два параметра, которые управляют TSD-процессом, передаются как частотнонезависимые параметры из кодера в декодер (см. фиг.3):

- Двоичное решение "транзиевты/без транзиевтов" детектора транзиевтов, выполняющегося в кодере, используется для управления отделением транзиевтов с детализацией временных слотов QMF в декодере. Эффективная схема кодирования без потерь используется для передачи данных о позициях слотов QMF с транзиевтами.

- Действительные параметры декоррелятора транзиевтов, которые нужны для декоррелятора транзиевтов, чтобы управлять пространственным распределением транзиевтов. Параметры декоррелятора транзиевтов обозначают угол между понижающим микшированием и его остатком. Эти параметры передаются только для временных слотов, в которых на кодере было обнаружено содержание транзиевтов.

Для того чтобы оценить качество вышеописанной технологии, были проведены две проверки на прослушивание MUSHRA в управляемой среде проверки на прослушивание с использованием высококачественных электростатических наушников STAX. Проверка выполнялась при конфигурации стерео 32 кбит/с и 16 кбит/с. Шестнадцать экспертных слушателей участвовали в каждой из проверок.

Так как набор для проверки USAC не содержит элементов аплодисментов, чтобы продемонстрировать преимущество предложенной технологии, были выбраны дополнительные элементы аплодисментов. Элементы, приведенные в таблице 1, были

включены в проверку:

Таблица 1 Элементы проверки на прослушивание	
Элемент	Свойства
5 ARL_applause	аплодисменты с частотой от низкой к средней (элемент набора для проверки MPS)
applause4s	очень частые аплодисменты, содержащие несколько отличимых хлопков
Applse_2ch	частые многоканальные аплодисменты - передние каналы (элемент набора для проверки MPS)
Applse_st	частые многоканальные аплодисменты - понижающее микширование до стерео (элемент набора для проверки MPS)
Klatschen	сигнал редких аплодисментов

10 Что касается обычных двенадцати элементов проверки на прослушивание MPEG USAC, TSD всегда неактивен. Однако эти элементы не остаются точно идентичными по битам, так как в битовый поток дополнительно включен бит включения TSD (указывающий, что TSD отключен), и, таким образом, это немного воздействует на битовый бюджет для базового кодера. Так как эти отличия очень малы, эти элементы  
15 не были включены в проверку на прослушивание. Предоставляются данные по размеру этих отличий, чтобы показать, что эти изменения ничтожны и незаметны.

Инструмент кодека, называемый inter-TES, является частью эталонной модели 8 (RM8) USAC. Так как этот метод был представлен для улучшения воспринимаемого качества транзиентов, включающих в себя подобные аплодисментам сигналы, inter-  
20 TES был всегда включен при каждом условии проверки. При такой установке гарантируется наилучшее возможное качество и демонстрируется ортогональность inter-TES и TSD.

Проверки системы имеют следующие конфигурации:

- RM8: система RM8 USAC
- 25 - CE: система RM8 USAC, расширенная посредством управляющего транзиентами декоррелятора (TSD)

На фиг.4 и 5 изображены баллы MUSHRA вместе с их 95%-ми доверительными интервалами для сценария проверки при 32 кбит/с. Для данных проверки было предложено t-распределение Стьюдента. Абсолютные баллы на фиг.4 показывают  
30 более высокий средний балл для всех элементов, для четырех из пяти элементов существует значительное улучшение в смысле 95% достоверности. Ни один элемент не ухудшился в отличие от RM8. Разностные баллы для USAC+TSD, оцененные в базовом эксперименте (CE) TSD по отношению к RM8 USAC, нанесены на фиг.5. Здесь может быть видно значительное улучшение для всех элементов.

35 Для схемы проверки при 16 кбит/с на фиг.6 и 7 изображены баллы MUSHRA вместе с их 95%-ми доверительными интервалами. Было предложено t-распределение Стьюдента для данных. Абсолютные баллы на фиг.6 показывают более высокий средний балл для каждого элемента. Для одного элемента может быть видна значимость в смысле 95% достоверности. Ни один элемент не имеет худшего балла, чем RM8. Разностные баллы  
40 нанесены на фиг.7. Снова было продемонстрировано значительное улучшение для всех элементов по отношению к разным данным.

Инструмент TSD включается посредством флага bsTsdEnable, передаваемого в битовом потоке. Если TSD включен, действительное разделение транзиентов управляется посредством флагов обнаружения транзиентов TsdSepData, которые также передаются  
45 в битовом потоке и которые кодируются в bsTsdCodedPos в случае, когда TSD включен.

В кодере флаг включения TSD bsTsdEnable генерируется посредством сегментного классификатора. Флаги обнаружения транзиентов TsdSepData задаются детектором транзиентов.

Как уже отмечено, TDS не активируется для двенадцати элементов проверки MPEG USAC. Для пяти дополнительных элементов аплодисментов активация TSD изображена на фиг.8, отображающей логическое состояние bsTsdEnable в зависимости от времени.

Если TSD активирован, транзиенты обнаруживаются в определенных временных слотах QMF, и они впоследствии подаются на выделенный декоррелятор транзиентов. Для каждого дополнительного элемента проверки таблица 2 приводит проценты слотов внутри активированных посредством TSD кадров, которые содержат транзиенты.

10

Элемент	Частота слотов с транзиентами (%)
ARL_applause	23,4
Applause4s	20,1
applse_2ch	24,7
applse_st	23,8
Klatschen	21,3

15

Передача решений отделения транзиентов и параметров декоррелятора из кодера в декодер требует определенной величины дополнительной информации. Однако эта величина с избытком компенсируется сбережениями битовой скорости, возникающими из-за передачи пространственных широкополосных меток внутри MPS.

20

Вследствие этого, средняя битовая скорость дополнительной информации MPS+TSD даже ниже, чем битовая скорость дополнительной информации простого MPS при простом USAC, как приведено в первом столбце таблицы 3. В предложенной конфигурации, использованной для оценки субъективного качества, средние битовые скорости, приведенные во втором столбце таблицы 3, были измерены для TDS:

25

Элемент	Средняя битовая скорость (бит/с) дополнительной информации MPS(+TSD)	
	простое USAC RM8	USAC с TSD
ARL_applause	2966	2345
Applause4s	2754	2278
applse_2ch	3000	2544
applse_st	2735	2253
Klatschen	2950	2495

30

Сложность вычисления TSD возникает из-за

- декодирования позиций слотов с транзиентами,
- сложности декоррелятора транзиентов.

35

Предполагая длину пространственного кадра MPEG Surround в 32 временных слота, декодирование позиций слотов требует (64 делений + 80 умножений) на каждый пространственный кадр в худшем случае, т.е.  $64 \cdot 25 + 80 = 1680$  операций на каждый пространственный кадр.

40

Игнорируя операции копирования и условные операторы, сложность декоррелятора транзиентов определяется одним комплексным умножением на каждый слот и гибридным диапазоном QMF.

Это приводит к следующим числам общей сложности TSD, показанным в сравнении с числами сложности простого USAC в таблице 4:

45

	Сложность простого USAC, в МОПС	TSD: сложность декоррелятора транзиентов, в МОПС	TSD: сложность декодера позиций слотов, в МОПС	$\Sigma$ (сложность TSD), в МОПС	$\Sigma$ (сложность TSD) относительно простого USAC

16 кбит/с стерео ( $f_s=28,8$ кГц)	8,7	0,117	0,024	0,141	1,62%
32 кбит/с стерео ( $f_s=40$ кГц)	13,2	0,163	0,033	0,196	1,48%

Итак, данные проверки на прослушивание ясно показывают значительное улучшение субъективного качества сигналов аплодисментов в разностных баллах всех элементов в обеих рабочих точках. В том, что касается абсолютных баллов, все элементы при условии TSD показывают более высокий средний балл. Для 32 кбит/с значительное улучшение существует для четырех из пяти элементов. Для 16 кбит/с один элемент показывает значительное улучшение. Ни один элемент не имеет худшего балла, чем RM8. Улучшение достигается при ничтожных вычислительных затратах, как может быть видно из данных о сложности. Это дополнительно подчеркивает преимущество инструмента TSD для USAC.

Вышеописанный управляющий транзиентами декоррелятор значительно улучшает обработку аудио при USAC. Однако, как было также видно выше, управляющий транзиентами декоррелятор требует информации о существовании или несуществовании транзиентов в конкретном слоте. При USAC, информация о временных слотах может быть передана на поккадровой основе. Кадр содержит несколько, например 32, временных слота. Вследствие этого следует осознавать, что кодер также передает информацию о том, какие слоты содержат транзиенты, на поккадровой основе. Уменьшение числа битов, которые должны быть переданы, является критичным при обработке аудиосигнала. Поскольку даже одиночная аудиозапись содержит огромное число кадров, это означает, что даже если число битов, которые должны быть переданы для каждого кадра, уменьшено всего лишь на несколько битов, общая скорость передачи битов может быть значительно уменьшена.

Проблема декодирования позиций слотов с событиями в кадре аудиосигнала, однако, не ограничивается проблемой декодирования транзиентов. Более того, было бы полезно декодировать также позиции слотов других событий, как, например, является ли слот кадра аудиосигнала тональным (или нет), содержит ли он шумы (или нет) и подобных. В действительности, устройство для эффективного кодирования и декодирования позиций слотов с событиями в кадре аудиосигнала было бы очень полезным для большого числа разного рода событий.

Когда этот документ ссылается на слоты или позиции слотов кадра аудиосигнала, слоты в этом смысле могут быть временными слотами, частотными слотами, частотно-временными слотами или любым другим типом слотов. Кроме того, следует понимать, что настоящее изобретение не ограничено обработкой аудио и кадров аудиосигнала при USAC, а вместо этого относится к любому типу кадров аудиосигнала и любому типу аудиоформатов, таких как MPEG1/2 уровень 3 ("MP3"), улучшенное аудиокодирование (AAC) и подобным. Эффективное кодирование и декодирование позиций слотов с событиями в кадре аудиосигнала было бы очень полезным для любого типа кадра аудиосигнала.

Вследствие этого, целью настоящего изобретения является предоставить устройство для кодирования позиций слотов с событиями в кадре аудиосигнала с помощью малого числа битов. Более того, целью настоящего изобретения является предоставить устройство для декодирования позиций слотов с событиями в кадре аудиосигнала, кодированных устройством для кодирования согласно настоящему изобретению. Цели настоящего изобретения достигаются посредством устройства для декодирования по п.1 формулы изобретения, устройства для кодирования по п.11 формулы изобретения, способа декодирования по п.14 формулы изобретения, способа кодирования по п.15

формулы изобретения, компьютерной программы для декодирования по п.16 формулы изобретения, компьютерной программы для кодирования по п.17 формулы изобретения и кодированного сигнала по п.18 формулы изобретения.

Согласно настоящему изобретению, предполагается, что число слотов кадра, указывающее общее число слотов кадра аудиосигнала, и число слотов с событиями, указывающее число слотов, содержащих события кадра аудиосигнала, могут быть доступны в декодирующем устройстве настоящего изобретения. Например, кодер может передавать число слотов кадра и/или число слотов с событиями на устройство для декодирования. Согласно варианту осуществления, кодер может указывать общее число слотов кадра аудиосигнала посредством передачи числа, которое является общим числом слотов кадра аудиосигнала минус 1. Кодер может дополнительно указывать число слотов, содержащих события кадра аудиосигнала, посредством передачи числа, которое является числом слотов, содержащих события кадра аудиосигнала, минус 1. В качестве альтернативы, декодер может сам определить общее число слотов кадра аудиосигнала и число слотов, содержащих события кадра аудиосигнала, без информации от кодера.

На основе этих предположений, согласно настоящему изобретению, данное число позиций слотов, содержащих события в кадре аудиосигнала, может быть кодировано и декодировано с использованием следующих результатов изысканий:

Пусть  $N$  будет общим числом слотов кадра аудиосигнала, и  $P$  будет числом слотов, содержащих события кадра аудиосигнала.

Предполагается, что как устройство для кодирования, так и устройство для декодирования осведомлены о значениях  $N$  и  $P$ .

Зная  $N$  и  $P$ , можно определить, что есть только  $\binom{N}{P}$  разных комбинаций позиций слотов, содержащих события в кадре аудиосигнала.

Например, если позиции слотов в кадре пронумерованы от 0 до  $N-1$  и если  $P=8$ , то первой возможной комбинацией позиций слотов с событиями будет (0, 1, 2, 3, 4, 5, 6, 7), второй комбинацией будет (0, 1, 2, 3, 4, 5, 6, 8) и так далее, до комбинации ( $N-8$ ,  $N-7$ ,  $N-6$ ,  $N-5$ ,  $N-4$ ,  $N-3$ ,  $N-2$ ,  $N-1$ ), так что в итоге есть  $\binom{N}{P}$  разных комбинаций.

Более того, настоящее изобретение использует дополнительные результаты изысканий, что число состояний события может быть кодировано устройством для кодирования и что число состояний события передается декодеру. Если каждая из возможных  $\binom{N}{P}$  комбинаций представлена уникальным числом состояний события и если устройство для декодирования осведомлено о том, какое число состояний события представляет какую комбинацию позиций слотов, содержащих события в кадре аудиосигнала (например, посредством применения соответствующего способа декодирования), то устройство для декодирования может декодировать позиции слотов, содержащие события, с использованием  $N$ ,  $P$  и числа состояний события. Для огромного количества типичных значений для  $N$  и  $P$  такой метод кодирования использует меньше битов для кодирования позиций слотов с событиями по сравнению с другими методами (например, использующими массив битов с одним битом для каждого слота кадра, в котором каждый бит указывает, встречается ли событие в этом слоте или нет).

Другими словами, проблема кодирования позиций слотов с событиями в кадре аудиосигнала может быть решена посредством кодирования дискретного числа  $P$  позиций  $p_k$  на диапазоне  $[0..N-1]$ , так что позиции не перекрываются,  $p_k \neq p_h$  для  $k \neq h$ , с как можно меньшим количеством битов. Так как порядок позиций не имеет значения,

следовательно, число уникальных комбинаций позиций является биномиальным коэффициентом  $\binom{N}{p}$ . Число требуемых битов, таким образом, составляет

$$bits = \text{ceil} \left( \log_2 \left( \binom{N}{p} \right) \right).$$

В варианте осуществления предоставляется устройство для декодирования, в котором данное устройство для декодирования выполнено с возможностью проведения проверки, сравнивающей число состояний события или обновленное число состояний события с пороговым значением. Такая проверка может быть использована для получения позиций слотов, содержащих события, из числа состояний события. Проверка сравнения числа состояний события с пороговым значением может быть проведена посредством сравнения, является ли число состояний события или обновленное число состояний события большим, большим или равным, меньшим или меньшим или равным пороговому значению. Кроме того, предпочтительно, чтобы устройство для декодирования было выполнено с возможностью обновления числа состояний события или обновленного числа состояний события в зависимости от результата проверки.

Согласно варианту осуществления, предоставляется устройство для декодирования, которое выполнено с возможностью проведения проверки, сравнивающей число состояний события или обновленное число состояний события относительно конкретного рассматриваемого слота, при этом пороговое значение зависит от числа слотов кадра, числа слотов с событиями и от позиции рассматриваемого слота внутри кадра. Кроме того, позиции слотов, содержащих события, могут быть определены на послотовой основе, с решением для каждого слота кадра, один за другим, содержит ли слот событие.

Согласно дополнительному варианту осуществления, предоставляется устройство для декодирования, которое выполнено с возможностью разбиения кадра на первый раздел кадра, содержащий первый набор слотов кадра, и на второй раздел кадра, содержащий второй набор слотов кадра, и в котором данное устройство для декодирования дополнительно выполнено с возможностью определения позиций, содержащих события, для каждого из разделов кадра по отдельности. Кроме того, позиции слотов, содержащих события, могут быть определены посредством повторного разбиения кадра или разделов кадра на даже меньшие разделы кадра.

Ниже, варианты осуществления настоящего изобретения описаны более подробно относительно чертежей, на которых:

на фиг.1 показано типичное применение декоррелятора в повышающем микшере моно-стерео;

на фиг.2 показано дополнительное типичное применение декоррелятора в повышающем микшере моно-стерео;

на фиг.3 показан обзор системы преобразования одного канала в два (ОТТ), включающей в себя управляющий транзисторами декоррелятор (TSD);

на фиг.4 показана диаграмма, иллюстрирующая абсолютные баллы для 32 кбит/с стерео, сравнивающая RM8 USAC и USAC RM8+TSD в базовом эксперименте (CE) TSD;

на фиг.5 показана диаграмма, отображающая разностные баллы для 32 кбит/с стерео, сравнивающая USAC, использующее управляющий транзисторами декоррелятор, с системой простого USAC;

на фиг.6 показана диаграмма, отображающая абсолютные баллы для 16 кбит/с стерео, сравнивающая RM8 USAC и USAC RM8+TSD в базовом эксперименте (CE) TSD;

на фиг.7 показана диаграмма, отображающая разностные баллы для 16 кбит/с стерео, сравнивающая USAC, использующее управляющий транзиентами декоррелятор, с системой простого USAC;

на фиг.8 отображена активность TSD для пяти дополнительных элементов, изображенная как логическое состояние флага bsTsdEnable;

на фиг.9A проиллюстрировано устройство для декодирования позиций слотов, содержащих события в кадре аудиосигнала, согласно варианту осуществления настоящего изобретения;

на фиг.9B проиллюстрировано устройство для декодирования позиций слотов, содержащих события в кадре аудиосигнала, согласно дополнительному варианту осуществления настоящего изобретения;

на фиг.9C проиллюстрировано устройство для декодирования позиций слотов, содержащих события в кадре аудиосигнала, согласно еще одному варианту осуществления настоящего изобретения;

на фиг.10 показана схема последовательности операций, иллюстрирующая процесс декодирования, проводимый устройством для декодирования, согласно варианту осуществления настоящего изобретения;

на фиг.11 показан псевдокод, реализующий декодирование позиций слотов, содержащих события, согласно варианту осуществления настоящего изобретения;

на фиг.12 показана схема последовательности операций, иллюстрирующая процесс кодирования, проводимый устройством для кодирования, согласно варианту осуществления настоящего изобретения;

на фиг.13 проиллюстрирован псевдокод, изображающий процесс кодирования позиций слотов, содержащих события в кадре аудиосигнала, согласно дополнительному варианту осуществления настоящего изобретения;

на фиг.14 проиллюстрировано устройство для декодирования позиций слотов, содержащих события в кадре аудиосигнала, согласно дополнительному варианту осуществления настоящего изобретения;

на фиг.15 проиллюстрировано устройство для кодирования позиций слотов, содержащих события в кадре аудиосигнала, согласно варианту осуществления настоящего изобретения;

на фиг.16 изображен синтаксис данных MPS 212 USAC согласно варианту осуществления;

на фиг.17 проиллюстрирован синтаксис TsdData USAC согласно варианту осуществления;

на фиг.18 проиллюстрирована таблица nBitsTrSlots в зависимости от длины MPS-кадра;

на фиг.19 показана таблица, относящаяся к bsTempShapeConfig USAC согласно варианту осуществления;

на фиг.20 изображен синтаксис TempShapeData USAC согласно варианту осуществления;

на фиг.21 проиллюстрирован блок D декоррелятора в блоке OTT-декодирования согласно варианту осуществления;

на фиг.22 изображен синтаксис EcData USAC согласно варианту осуществления;

на фиг.23 показана схема прохождения сигналов для генерирования данных TSD.

На фиг.9A проиллюстрировано устройство 10 для декодирования позиций слотов,

содержащих события в кадре аудиосигнала, согласно варианту осуществления настоящего изобретения. Устройство 10 для декодирования содержит анализирующий блок 20 и блок 30 генерирования. Число слотов кадра FSN, указывающее общее число слотов кадра аудиосигнала, число слотов с событиями ESON, указывающее число слотов, содержащих события кадра аудиосигнала, и число состояний события ESTN подаются в устройство 10 для декодирования. Устройство 10 для декодирования затем декодирует позиции слотов, содержащих события, посредством использования числа слотов кадра FSN, числа слотов с событиями ESON и числа состояний события ESTN. Декодирование проводится анализирующим блоком 20 и блоком 30 генерирования, которые взаимодействуют в процессе декодирования. Тогда как анализирующий блок 20 ответственен за исполнение проверок, например сравнение числа состояний события ESTN с пороговым значением, блок 30 генерирования генерирует и обновляет промежуточные результаты процесса декодирования, например, обновленное число состояний события.

Кроме того, блок 30 генерирования генерирует указание множества позиций слотов, содержащих события в кадре аудиосигнала. Конкретное указание множества позиций слотов, содержащих события кадра аудиосигнала, может называться как "состояние указания".

Согласно варианту осуществления, указание множества позиций слотов, содержащих события в кадре аудиосигнала, может быть сгенерировано так, что в первый момент времени блок 30 генерирования указывает для первого слота, содержит ли слот событие или нет, во второй момент времени блок 30 генерирования указывает для второго слота, содержит ли слот событие или нет, и так далее.

Согласно дополнительному варианту осуществления, указание множества позиций слотов, содержащих события, может, например, быть массивом битов, указывающим для каждого слота кадра, содержит ли он событие.

Анализирующий блок 20 и блок 30 генерирования могут взаимодействовать так, что оба блока вызывают друг друга один или более раз в процессе декодирования, чтобы произвести промежуточные результаты.

На фиг.9В проиллюстрировано устройство 40 для декодирования согласно варианту осуществления настоящего изобретения. Устройство 40 для декодирования, среди прочего, отличается от устройства 10 для декодирования по фиг.9А в том, что дополнительно содержит процессор 50 аудиосигнала. Процессор 50 аудиосигнала принимает входной аудиосигнал и указание множества позиций слотов, содержащих события в кадре аудиосигнала, которое было сгенерировано блоком 45 генерирования. В зависимости от указания, процессор 50 аудиосигнала генерирует выходной аудиосигнал. Процессор 50 аудиосигнала может генерировать выходной аудиосигнал, например, посредством декоррелирования входного аудиосигнала. Кроме того, процессор 50 аудиосигнала может содержать решетчатый ИР-декоррелятор 54, декоррелятор 56 транзиентов и блок 52 отделения транзиентов для генерирования выходного аудиосигнала, как проиллюстрировано на фиг.3. Если указание множества позиций слотов, содержащих события в кадре аудиосигнала, указывает, что слот содержит транзиенты, то процессор 50 аудиосигнала будет декоррелировать входной аудиосигнал, относящийся к этому слоту, посредством декоррелятора 56 транзиентов. Если, однако, указание множества позиций слотов, содержащих события в кадре аудиосигнала, указывает, что слот не содержит транзиенты, то процессор аудиосигнала будет декоррелировать входной аудиосигнал S, относящийся к этому слоту, посредством использования решетчатого ИР-декоррелятора 54. Процессор аудиосигнала использует

блок 52 отделения транзиентов, который решает на основе указания, подан ли участок входного аудиосигнала, относящийся к слоту, в декоррелятор 56 транзиентов или в решетчатый ПР-декоррелятор 54, в зависимости от того, указывает ли указание, что конкретный слот содержит транзиенты (декорреляция посредством декоррелятора 56 транзиентов) или слот не содержит транзиенты (декорреляция посредством решетчатого ПР-декоррелятора 54).

На фиг.9С проиллюстрировано устройство 60 для декодирования согласно варианту осуществления настоящего изобретения. Устройство 60 для декодирования отличается от устройства 10 для декодирования по фиг.9А в том, что дополнительно содержит селектор 90 слотов. Декодирование делается на послотовой основе, с решением для каждого слота кадра, один за другим, содержит ли слот событие. Селектор 90 слотов решает, какой слот кадра рассматривать. Предпочтительным подходом был бы тот, где селектор 90 слотов выбирает слоты кадра один за другим.

Послотовое декодирование устройства 60 для декодирования по этому варианту осуществления основано на следующих результатах изысканий, которые могут быть применены для вариантов осуществления устройства для декодирования, устройства для кодирования, способа декодирования и способа кодирования позиций слотов, которые содержат события в кадре аудиосигнала. Следующие результаты изысканий также применимы для соответствующих компьютерных программ и кодированных сигналов.

Предположим, что  $N$  является (общим) числом слотов кадра аудиосигнала и  $P$  является числом слотов, содержащих события кадра (это означает, что  $N$  может быть числом слотов кадра FSN и  $P$  может быть числом слотов с событиями ESON).

Рассматривается первый слот кадра. Можно различить два случая.

Если первый слот является слотом, который не содержит событие, то, относительно оставшихся  $N-1$  слотов кадра, есть только  $\binom{N-1}{P}$  разных возможных комбинаций из  $P$  позиций слотов, содержащих событие, относительно оставшихся  $N-1$  слотов кадра.

Однако если первый слот является слотом, который содержит событие, то, относительно оставшихся  $N-1$  слотов кадра, есть только  $\binom{N-1}{P-1} = \binom{N}{P} - \binom{N-1}{P}$  разных возможных комбинаций из оставшихся  $P-1$  слотов, содержащих событие, относительно оставшихся  $N-1$  слотов кадра.

На основе этих результатов изысканий варианты осуществления дополнительно основаны на результатах изысканий, что все комбинации с первым слотом, где событие не встретилось, должны быть кодированы посредством чисел состояний события, которые меньше, чем пороговое значение, или равны ему. Кроме того, все комбинации с первым слотом, где встретилось событие, должны быть кодированы посредством чисел состояний события, которые больше, чем пороговое значение. В варианте осуществления все числа состояний события могут быть положительными целыми числами или 0, и подходящим пороговым значением касательно первого слота может быть  $\binom{N-1}{P}$ .

В варианте осуществления устройство для декодирования выполнено с возможностью определения того, содержит ли первый слот кадра событие, посредством проверки того, является ли число состояний события большим, чем пороговое значение. (В качестве альтернативы, может также быть реализован процесс кодирования/декодирования по вариантам осуществления, так что устройство для декодирования проверяет, является ли число состояний события большим или равным, меньшим или равным или меньшим, чем пороговое значение.) После анализа первого слота

декодирование продолжается для второго слота кадра с использованием отрегулированных значений. Помимо регулирования числа рассматриваемых слотов (которое уменьшено на один), число слотов, содержащих события, также, в конечном счете, уменьшается на один (если первый слот содержит событие), и число состояний события регулируется, в случае, когда число состояний события было больше, чем пороговое значение, чтобы удалить участок, относящийся к первому слоту, исходя из числа состояний события. Процесс декодирования может быть продолжен для дальнейших слотов кадра аналогичным образом.

В варианте осуществления кодируется дискретное число  $P$  из позиций  $p_k$  на диапазоне  $[0 \dots N-1]$ , так что позиции не перекрываются,  $p_k \neq p_h$  для  $k \neq h$ . Здесь, каждая уникальная комбинация позиций на данном диапазоне называется состоянием, и каждая возможная позиция в этом диапазоне называется слотом. Согласно варианту осуществления устройства для декодирования, рассматривается первый слот в диапазоне. Если слот не имеет назначенной ему позиции, то диапазон может быть уменьшен до  $N-1$ , и число возможных состояний уменьшается до  $\binom{N-1}{P}$ . Напротив, если состояние больше, чем  $\binom{N-1}{P}$ , то можно сделать вывод, что первый слот имеет назначенную ему позицию.

Следующий алгоритм декодирования может быть результатом этого:

Для каждого слота  $h$   
 Если состояние  $> \binom{N-h-1}{P}$ , то  
 Назначить позицию для слота  $h$   
 Обновить оставшееся состояние как  $state := state - \binom{N-h-1}{P}$   
 Уменьшить число позиций влево как  $P := P - 1$   
 Конец  
 Конец

Вычисление биномиального коэффициента на каждой итерации было бы затратным. Вследствие этого, согласно вариантам осуществления, следующие правила могут быть использованы для обновления биномиального коэффициента с использованием значения из предыдущей итерации:

$$\binom{N}{P} = \binom{N-1}{P} \cdot \frac{N}{N-P} \quad \text{и} \quad \binom{N}{P} = \binom{N}{P-1} \cdot \frac{N-P+1}{P}$$

Используя эти формулы, каждое обновление биномиального коэффициента стоит только одно умножение и одно деление, тогда как точная оценка стоила бы  $P$  умножений и делений на каждой итерации.

В этом варианте осуществления общая сложность декодера составляет  $P$  умножений и делений для инициализации биномиального коэффициента, для каждой итерации 1 умножение, деление и оператор "если", и для каждой закодированной позиции 1 умножение, сложение и деление. Следует отметить, что в теории было бы возможно уменьшить число делений, нужных для инициализации, до одного. На практике, однако, этот подход привел бы к очень большим целым числам, которые трудно обработать. Сложностью декодера в худшем случае является тогда  $N+2P$  делений и  $N+2P$  умножений,  $P$  сложений (могут быть проигнорированы, если используются MAC-операции) и  $N$  операторов "если".

В варианте осуществления алгоритм кодирования, используемый устройством для

кодирования, не должен выполнять итерацию во всех слотах, а только в тех, которые имеют назначенную им позицию. Вследствие этого

Для каждой позиции  $P_h$ ,  $h=1\dots P$

5 Обновить состояние как  $\text{state} := \text{state} + \binom{P_{h-1}}{h}$ .

Сложностью кодера в худшем случае является  $P \cdot (P-1)$  умножений и  $P \cdot (P-1)$  делений, а также  $P-1$  делений.

10 На фиг.10 иллюстрирован процесс декодирования, проводимый устройством для декодирования, согласно варианту осуществления настоящего изобретения. В этом варианте осуществления декодирование выполняется на послотовой основе.

15 На этапе 110 инициализируются значения. Устройство для декодирования хранит число состояний события, которое оно приняло как входное значение, в переменной  $s$ . Кроме того, число слотов, содержащих события кадра, как указано числом слотов с событиями, хранится в переменной  $p$ . Кроме того, общее число слотов, содержащихся в кадре, как указано числом слотов кадра, хранится в переменной  $N$ .

20 На этапе 120 значение  $\text{TsdSepData}[t]$  инициализируется со значением 0 для всех слотов кадра. Массив битов  $\text{TsdSepData}$  является выходными данными, которые должны быть сгенерированы. Он указывает для каждой позиции слота  $t$ , содержит ли событие слот с соответствующей позицией слота ( $\text{TsdSepData}[t]=1$ ) или не содержит ( $\text{TsdSepData}[t]=0$ ). На этапе 120 соответствующие значения всех слотов кадра инициализируются со значением 0.

25 На этапе 130 переменная  $k$  инициализируется со значением  $N-1$ . В этом варианте осуществления слоты кадра, содержащего  $N$  элементов, пронумерованы  $0, 1, 2, \dots, N-1$ . Задание  $k=N-1$  означает, что слот с наивысшим числом слота считается первым.

На этапе 140 рассматривается, верно ли  $k \geq 0$ . Если  $k < 0$ , декодирование позиций слотов было закончено, и процесс прекращается, иначе процесс продолжается на этапе 150.

30 На этапе 150 проверяется, верно ли  $p > k$ . Если  $p$  больше, чем  $k$ , это означает, что все оставшиеся слоты содержат событие. Процесс продолжается на этапе 230, в котором все значения поля  $\text{TsdSepData}$  оставшихся слотов  $0, 1, \dots, k$  заданы в значение 1, указывая, что каждый из оставшихся слотов содержит событие. В этом случае процесс после этого прекращается. Однако если на этапе 150 обнаружено, что  $p$  не больше, чем  $k$ , процесс декодирования продолжается на этапе 160.

35 На этапе 160 вычисляется значение  $c = \binom{k}{p}$ , с используется как пороговое значение.

На этапе 170 проверяется, является ли (в конечном счете, обновленное) число состояний события  $s$  большим, чем  $c$ , или равно ему, где  $c$  является пороговым значением, только что вычисленным на этапе 160.

40 Если  $s$  меньше, чем  $c$ , это означает, что рассматриваемый слот ( $s$  позицией слота  $k$ ) не содержит событие. В этом случае не нужно предпринимать дополнительное действие, так как  $\text{TsdSepData}[k]$  уже был задан в значение 0 для этого слота на этапе 140. Процесс тогда продолжается на этапе 220. На этапе 220  $k$  задается так, чтобы быть  $k:=k-1$ , и рассматривается следующий слот.

45 Однако если проверка на этапе 170 показывает, что  $s$  больше, чем  $c$ , или равно ему, это означает, что рассматриваемый слот  $k$  содержит событие. В этом случае число состояний события  $s$  обновляется и задается в значение  $s:=s-c$  на этапе 180. Кроме того,  $\text{TsdSepData}[k]$  задается в значение 1 на этапе 190 для указания, что слот  $k$  содержит событие. Более того, на этапе 200  $p$  задается в значение  $p-1$ , указывая, что оставшиеся слоты, которые должны быть исследованы, теперь содержат только  $p-1$  слотов  $s$

событиями.

На этапе 210 проверяется, является ли  $p$  равным 0. Если  $p$  равно 0, оставшиеся слоты не содержат события, и процесс декодирования заканчивается. Иначе, по меньшей мере один из оставшихся слотов содержит событие, и процесс продолжается на этапе 220, где процесс декодирования продолжается со следующим слотом ( $k-1$ ).

Процесс декодирования по варианту осуществления, проиллюстрированному на фиг.10, генерирует массив TsdSepData в качестве выходного значения, указывая для каждого слота  $k$  кадра, содержит ли слот событие ( $TsdSepData[k]=1$ ) или нет ( $TsdSepData[k]=0$ ).

Возвращаясь к фиг.9с, устройство 60 для декодирования по варианту осуществления, в котором данное устройство реализует процесс декодирования, проиллюстрированный на фиг.10, содержит селектор 90 слотов, который решает, какие слоты рассматривать. Относительно фиг.10, такой селектор слотов был бы выполнен с возможностью исполнения этапов 130 и 220 процесса по Фиг. 10. Подходящий анализирующий блок 70 этого варианта осуществления был бы выполнен с возможностью исполнения этапов 140, 150, 170 и 210 обработки по фиг.10. Блок 80 генерирования по такому варианту осуществления был бы выполнен с возможностью проведения всех остальных этапов обработки по фиг.10.

На фиг.11 показан псевдокод, реализующий декодирование позиций слотов, содержащих события, согласно варианту осуществления настоящего изобретения.

На фиг.12 иллюстрирован процесс кодирования, проводимый устройством для кодирования, согласно варианту осуществления настоящего изобретения. В этом варианте осуществления кодирование выполняется на послотовой основе. Целью процесса кодирования согласно варианту осуществления, проиллюстрированному на фиг.12, является сгенерировать число состояний события.

На этапе 310 инициализируются значения.  $p\_s$  инициализируется со значением 0. Число состояний события генерируется посредством успешного обновления переменной  $p\_s$ . Когда процесс кодирования закончен,  $p\_s$  будет нести число состояний события. Этап 310 также инициализирует переменную  $k$  посредством задания  $k$  в значение  $k:=$  число слотов, содержащих события в кадре,  $-1$ .

На этапе 320 переменная "slots" задается в значение  $slots:=tsdPos[k]$ , где  $tsdPos$  является массивом, удерживающим позиции слотов, содержащих события. Позиции слотов в массиве хранятся в возрастающем порядке.

На этапе 330 проводится проверка, проверяющая, верно ли  $k \geq slots$ . Если это так, процесс прерывается. Иначе, процесс продолжается на этапе 340.

На этапе 340 вычисляется значение  $c = \binom{slots}{k+1}$ .

На этапе 350 переменная  $p\_s$  обновляется и задается в значение  $p\_s:=p\_s+c$ .

На этапе 360  $k$  задается в значение  $k:=k-1$ .

Затем, на этапе 370 проводится проверка, проверяющая, верно ли  $k \geq 0$ . В этом случае рассматривается следующий слот  $k-1$ . Иначе, процесс прерывается.

На фиг.13 изображен псевдокод, реализующий кодирование позиций слотов, содержащих события, согласно варианту осуществления настоящего изобретения.

На фиг.14 проиллюстрировано устройство 410 для декодирования позиций слотов, содержащих события в кадре аудиосигнала, согласно дополнительному варианту осуществления настоящего изобретения. Снова, как на фиг.9а, число слотов кадра FSN, указывающее общее число слотов кадра аудиосигнала, число слотов с событиями ESON, указывающее число слотов, содержащих события кадра аудиосигнала, и число состояний события ESTN подаются в устройство 410 для декодирования. Устройство 410 для

декодирования отличается от устройства по фиг.9а в том, что дополнительно содержит блок 440 разделения кадра. Блок 440 разделения кадра выполнен с возможностью разбиения кадра на первый раздел кадра, содержащий первый набор слотов кадра, и на второй раздел кадра, содержащий второй набор слотов кадра, и в котором позиции слотов, содержащие события, определяются по отдельности для каждого раздела кадра. Кроме того, позиции слотов, содержащих события, могут быть определены посредством повторного разбиения кадра или разделов кадра на даже меньшие разделы кадра.

Декодирование "на основе разделения" устройства 410 для декодирования по этому варианту осуществления основано на следующих концепциях, которые могут быть применены для вариантов осуществления устройства для декодирования, устройства для кодирования, способа декодирования и способа кодирования позиций слотов, которые содержат события в кадре аудиосигнала. Следующие концепции также применимы для соответствующих компьютерных программ и кодированных сигналов:

Декодирование на основе разделения основано на идее, что кадр разбивается на два раздела А и В кадра, причем каждый раздел кадра содержит набор слотов, в котором раздел А кадра содержит  $N_a$  слотов и в котором раздел В кадра содержит  $N_b$  слотов, и что  $N_a + N_b = N$ . Кадр может быть произвольно разбит на два раздела предпочтительно так, чтобы разделы А и В имели примерно одинаковое общее число слотов (например, так, чтобы  $N_a = N_b$  или  $N_a = N_b - 1$ ). Посредством разбиения кадра на два раздела задача определения позиций слотов, где встретились события, также разбивается на две подзадачи, а именно определение позиций слотов, где встретились события в разделе А кадра, и определение позиций слотов, где встретились события в разделе В кадра.

В этом варианте осуществления снова предполагается, что устройство для декодирования осведомлено о числе слотов кадра, числе слотов, содержащих событие кадра, и числе состояний события. Для решения обеих подзадач устройство для декодирования должно быть осведомлено о числе слотов каждого раздела кадра, числе слотов, где встретились события касательно каждого раздела кадра, и числе состояний события каждого раздела кадра (такое число состояний события раздела кадра теперь называется как "число подсостояний события").

Так как устройство для декодирования само разбивает кадр на два раздела кадра, оно само по себе знает, что раздел А кадра содержит  $N_a$  слотов, и раздел В кадра содержит  $N_b$  слотов. Определение числа слотов, содержащих события, для каждого из двух разделов кадра основано на следующих результатах изысканий.

Так как кадр был разбит на два раздела, каждый из слотов, содержащих события, теперь расположен либо в разделе А, либо в разделе В. Кроме того, предполагается, что Р является числом слотов, содержащих события раздела кадра, и N является общим числом слотов раздела кадра, и что  $f(P, N)$  является функцией, которая возвращает число разных комбинаций позиций слотов с событиями раздела кадра, тогда число разных комбинаций позиций слотов с событиями всего кадра (который был разбит на раздел А и раздел В) составляет:

Число слотов, содержащих события, в разделе А	Число слотов, содержащих события, в разделе В	Число разных комбинаций в целом кадре аудиосигнала с этой конфигурацией
0	P	$f(0, N_a) \bullet f(P, N_b)$
1	P-1	$f(1, N_a) \bullet f(P-1, N_b)$
2	P-2	$f(2, N_a) \bullet f(P-2, N_b)$
...	...	...
P	0	$f(P, N_a) \bullet f(0, N_b)$

На основе вышеприведенных размышлений, согласно варианту осуществления, все комбинации с первой конфигурацией, где раздел А имеет 0 слотов, содержащих события, и где раздел В имеет Р слотов, содержащих события, должны быть кодированы с числом состояний события, меньшим, чем первое пороговое значение. Число состояний события может быть кодировано как целочисленное значение, являющееся положительным или 0. Так как есть только  $f(0, N_a) \bullet f(P, N_b)$  комбинаций с первой конфигурацией, подходящим первым пороговым значением может быть  $f(0, N_a) \bullet f(P, N_b)$ .

Все комбинации со второй конфигурацией, где раздел А имеет 1 слот, содержащий события, и где раздел В имеет Р-1 слотов, содержащих события, должны быть кодированы с числом состояний события, большим, чем первое пороговое значение, или равным ему, но меньшим, чем второе значение, или равным ему. Так как есть только  $f(1, N_a) \bullet f(P-1, N_b)$  комбинаций со второй конфигурацией, подходящим вторым значением может быть  $f(0, N_a) \bullet f(P, N_b) + f(1, N_a) \bullet f(P-1, N_b)$ . Число состояний события для комбинаций с другими конфигурациями определяется аналогично.

Согласно варианту осуществления, декодирование выполняется посредством разделения кадра на два раздела А и В кадра. Затем, проверяется, является ли число состояний события меньшим, чем первое пороговое значение. В предпочтительном варианте осуществления первым пороговым значением может быть  $f(0, N_a) \bullet f(P, N_b)$ .

Если число состояний события меньше, чем первое пороговое значение, то может быть сделан вывод, что раздел А содержит 0 слотов, содержащих события, и раздел В содержит все Р слотов кадра, где встретились события. Декодирование тогда проводится для обоих разделов с соответственно определенным числом, представляющим число слотов, содержащих события соответствующего раздела. Кроме того, первое число состояний события определяется для раздела А, и второе число состояний события определяется для раздела В, которые соответственно используются как новое число состояний события. В рамках этого документа, число состояний события раздела кадра называется как "число подсостояний события".

Однако если число состояний события больше, чем первое пороговое значение, или равно ему, число состояний события может быть обновлено. В предпочтительном варианте осуществления число состояний события может быть обновлено посредством вычитания значения из числа состояний события, предпочтительно посредством вычитания первого порогового значения, например  $f(0, N_a) \bullet f(P, N_b)$ . На следующем этапе проверяется, является ли обновленное число состояний события меньше, чем второе пороговое значение. В предпочтительном варианте осуществления вторым пороговым значением может быть  $f(1, N_a) \bullet f(P-1, N_b)$ . Если число состояний события меньше, чем второе пороговое значение, то может быть получено, что раздел А содержит 1 слот, содержащий события, и раздел В содержит Р-1 слотов, содержащих события. Декодирование тогда проводится для обоих разделов с соответственно определенным числом слотов, содержащих события каждого раздела. Первое значение подсостояний события используется для декодирования раздела А, и второе значение подсостояний события используется для декодирования раздела В. Однако если число состояний события больше, чем второе пороговое значение, или равно ему, число состояний события может быть обновлено. В предпочтительном варианте осуществления число состояний события может быть обновлено посредством вычитания значения из числа состояний события, предпочтительно  $f(1, N_a) \bullet f(P-1, N_b)$ . Процесс декодирования аналогично применяется для оставшихся возможностей распределения слотов,

содержащих события касательно двух разделов кадра.

В варианте осуществления значение подсостояний события для раздела А и значение подсостояний события для раздела В могут быть использованы для декодирования

5 разделов А и раздела В, где оба значения подсостояний события определяются

значением состояний события  $f$ (число слотов, содержащих события раздела В,  $N_b$ ).

Предпочтительно, число подсостояний события раздела А является целочисленной частью вышеприведенного деления, и число подсостояний события раздела В является остатком этого деления. Число состояний события, используемое в этом делении, может

10 быть первоначальным числом состояний события кадра или обновленным числом состояний события, например обновленным посредством вычитания одного или более пороговых значений, как описано выше.

Чтобы проиллюстрировать вышеописанную концепцию декодирования на основе

разделения, рассмотрена ситуация, где кадр имеет два слота, содержащих события.

15 Кроме того, если  $f(p, N)$  снова является функцией, которая возвращает число разных комбинаций позиций слотов с событиями раздела кадра, в которой  $p$  является числом слотов, содержащих события раздела кадра, и  $N$  является общим числом слотов этого

раздела кадра, тогда, для каждого из возможных распределений позиций в результате получается следующее число возможных комбинаций:

20

Позиции в разделе А	Позиция в разделе В	Число комбинаций в этой конфигурации
0	2	$f(0, N_a) \cdot f(2, N_b)$
1	1	$f(1, N_a) \cdot f(1, N_b)$
2	0	$f(2, N_a) \cdot f(0, N_b)$

25 Таким образом, может быть сделан вывод, что если кодированное число состояний события кадра меньше, чем  $f(0, N_a) \cdot f(2, N_b)$ , то слоты, содержащие события, должны быть распределены как 0 и 2. Иначе,  $f(0, N_a) \cdot f(2, N_b)$  вычитается из числа состояний

30 события, и результат сравнивается с  $f(1, N_a) \cdot f(1, N_b)$ . Если он меньше, то позиции распределены как 1 и 1. Иначе, осталось только распределение 2 и 0, и позиции

распределены как 2 и 0.

Ниже предоставлен псевдокод согласно варианту осуществления для декодирования

35 позиций слотов, содержащих определенные события (здесь: "pulses") в кадре аудиосигнала. В этом псевдокоде "pulses\_a" является (предполагаемым) числом слотов, содержащих события в разделе А, и "pulses\_b" является (предполагаемым) числом

40 слотов, содержащих события в разделе В. В этом псевдокоде (в конечном счете, обновленное) число состояний события называется "state". Числа подсостояний события разделов А и В по-прежнему кодируются совместно в переменной "state". Согласно

схеме совместного кодирования по варианту осуществления, число подсостояний события из А (в настоящем документе называемое "state\_a") является целочисленной

частью деления  $state/f(pulses_b, N_b)$ , и число подсостояний события из В (в настоящем документе называемое "state\_b") является остатком этого деления. Кроме того, длина

(общее число слотов раздела) и число кодированных позиций (число слотов, содержащих события в разделе) обоих разделов могут быть декодированы посредством одинакового

45 подхода:

Функция  $x = \text{decodestate}(state, pulses, N)$

1. Разбить вектор на два раздела длиной  $N_a$  и  $N_b$ .

2. Для pulses\_a от 0 до pulses

- a.  $pulses\_b = pulses - pulses\_a$
- b. если  $state < f(pulses\_a, Na) * f(pulses\_b, Nb)$ , то прекратить цикл "для".
- c.  $state := state - f(pulses\_a, Na) * f(pulses\_b, Nb)$

5 3. Число возможных состояний для раздела В равно  $no\_states\_b = f(pulses\_b, Nb)$

4. Состояния  $state\_a$  и  $state\_b$  разделов А и В, соответственно, являются целой частью и остатком от деления  $state / no\_states\_b$ .

10 5. Если  $Na > 1$ , то декодированный вектор раздела А получается рекурсивно как  $xa = decodestate(state\_a, pulses\_a, Na)$

Иначе ( $Na == 1$ ), и вектор  $xa$  является скаляром, и мы можем задать  $xa = state\_a$ .

15 6. Если  $Nb > 1$ , то декодированный вектор раздела В получается рекурсивно как  $xb = decodestate(state\_b, pulses\_b, Nb)$

Иначе ( $Nb == 1$ ), и вектор  $xb$  является скаляром, и мы можем задать  $xb = state\_b$ .

20 7. Конечный выходной сигнал  $x$  получается путем объединения  $xa$  и  $xb$  как  $x = [xa\ xb]$ .

Выходным сигналом этого алгоритма является вектор, который имеет единицу (1) в каждой кодированной позиции (т.е. позиции слота для слота, содержащего событие) и ноль (0) где-либо еще (т.е. в позициях слотов, которые не содержат события).

25 Ниже предоставлен псевдокод согласно варианту осуществления для кодирования позиций слотов, содержащих события в кадре аудиосигнала, который использует аналогичные имена переменных с аналогичным значением, как приведено выше:

Функция  $state = encodestate(x, N)$

1. Разбить вектор на два раздела  $xa$  и  $xb$  длиной  $Na$  и  $Nb$ .

30 2. Подсчитать  $pulses$  в разделах А и В в  $pulses\_a$  и  $pulses\_b$  и задать  $pulses = pulses\_a + pulses\_b$ .

3. Задать состояние в 0.

4. Для  $k$  от 0 до  $pulses\_a - 1$

35 a.  $state := state + f(k, Na) * f(pulses - k, Nb)$ .

5. Если  $Na > 1$ , кодировать раздел А как  $state\_a = encodestate(xa, Na)$ ;

Иначе ( $Na == 1$ ), задать  $state\_a = xa$ .

6. Если  $Nb > 1$ , кодировать раздел В как

40  $state\_b = encodestate(xb, Nb)$ ;

Иначе ( $Nb == 1$ ), задать  $state\_b = xb$ .

7. Кодировать состояния совместно

$state := state + state\_a * f(pulses\_b, Nb) + state\_b$ .

45 Здесь, предполагается, что аналогично алгоритму декодера каждая кодированная позиция (т.е. позиция слота для слота, содержащего событие) идентифицируется посредством единицы (1) в векторе  $x$ , и все другие элементы являются нулем (0) (т.е. в позициях слотов, которые не содержат события.)

Вышеприведенные рекурсивные способы, сформулированные в псевдокоде, могут

легко быть реализованы нерекурсивным образом с использованием стандартных способов.

Согласно варианту осуществления настоящего изобретения, функция  $f(p, N)$  может быть реализована как таблица соответствия. Когда позиции не перекрываются, как, например, в текущем контексте, то функция числа состояний  $f(p, N)$  является просто биномиальной функцией, которая может быть вычислена по сети. Имеется

$$f(p, N) = \frac{N(N-1)(N-2)\dots(N-k)}{k(k-1)(k-2)\dots 1}.$$

Согласно варианту осуществления настоящего изобретения, и кодер, и декодер имеют цикл со счетчиком, где произведение  $f(p-k, N_a) \cdot f(k, N_b)$  вычисляется для последовательных значений из  $k$ . Для эффективного вычисления это может быть записано как

$$\begin{aligned} f(p-k, N_a) f(k, N_b) &= \frac{N_a(N_a-1)(N_a-2)\dots(N_a-p+k)}{(p-k)(p-k-1)(p-k-2)\dots 1} \cdot \frac{N_b(N_b-1)(N_b-2)\dots(N_b-k)}{k(k-1)(k-2)\dots 1} \\ &= \frac{N_a(N_a-1)(N_a-2)\dots(N_a-p-k+1)}{(p-k+1)(p-k)(p-k-1)\dots 1} \cdot \frac{N_b(N_b-1)(N_b-2)\dots(N_b-k+1)}{(k-1)(k-2)\dots 1} \cdot \frac{p-k+1}{N_a-p-k+1} \cdot \frac{N_a-k}{k} \\ &= f(p-k+1, N_a) f(k-1, N_b) \cdot \frac{p-k+1}{N_a-p-k+1} \cdot \frac{N_a-k}{k}. \end{aligned}$$

Другими словами, последовательные члены для вычитания/сложения (на этапе 2b и 2c в декодере и на этапе 4a в кодере) могут быть вычислены посредством трех умножений и одного деления за итерацию.

Аналогично, как в способе, описанном ранее, состояние длинного вектора (многословный кадр) может быть очень большим целым числом, легко продлевающим длину представления в стандартных процессорах. Вследствие этого, будет необходимо использовать арифметические функции, способные обрабатывать очень большие целые числа.

Касательно сложности, рассмотренный здесь способ, в отличие от вышеприведенных пословных процессов, является алгоритмом типа "разделяй и властвуй". Предполагая, что длина входного вектора имеет степень двойки, тогда рекурсия имеет глубину  $\log_2(N)$ .

Так как число импульсов остается постоянным на каждой глубине рекурсии, то число итераций цикла со счетчиком является одинаковым при каждой рекурсии. Из этого следует, что число циклов составляет  $\text{pulses} \cdot \log_2(N)$ .

Как разъяснено выше, каждое обновление  $f(p-k, N_a) \cdot f(k, N_b)$  может быть сделано с помощью трех умножений и одного деления.

Следует отметить, что вычитания и сравнения в декодере могут предполагаться как одна операция.

Может быть легко видно, что разделы объединяются  $\log_2(N)-1$  раз. При совместном кодировании состояний в кодере, таким образом, необходимо умножать и складывать  $\log_2(N)-1$  раз. Аналогично, при совместном декодировании состояний в декодере необходимо делить  $\log_2(N)-1$  раз.

Следует отметить, что из делений, только при совместном кодировании состояний

в декодере нужны деления, где знаменателем является длинное целое число. Другие деления всегда имеют относительно короткие целые числа в знаменателе. Так как деления с длинными знаменателями являются наиболее сложными операциями, их нужно избегать, когда возможно.

5 Итак, данное число арифметических операций с длинными целыми числами происходит в декодере

	Умножения	$(3 \bullet \text{pulses} + 1) \bullet \log_2(N) - 1$
	Деления	$(\text{pulses} + 1) \bullet \log_2(N) - 1$
10	из которых делений с длинным знаменателем	$\log_2(N) - 1$
	Сложения и вычитания	$\text{pulses} \bullet \log_2(N)$
	Аналогично, в кодере происходят	
	Умножения	$(3 \bullet \text{pulses} + 1) \bullet \log_2(N) - 1$
	Деления	$(\text{pulses} + 1) \bullet \log_2(N) - 1$
	из которых делений с длинным знаменателем	0
	Сложения и вычитания	$(\text{pulses} + 2) \bullet \log_2(N)$

15 Требуется только  $\log_2(N) - 1$  делений с длинным знаменателем.

В дополнительных вариантах осуществления вышеописанные варианты осуществления, которые содержат или которые выполнены с возможностью использования этапов рекурсивной обработки, модифицированы так, что некоторые или все из этапов рекурсивной обработки реализованы нерекурсивным образом с использованием стандартных способов.

На фиг.15 проиллюстрировано устройство (510) для кодирования позиций слотов, содержащих события в кадре аудиосигнала, согласно варианту осуществления. Устройство (510) для кодирования содержит генератор (530) числа состояний события, который выполнен с возможностью кодирования позиций слотов посредством кодирования числа состояний события. Кроме того, устройство содержит блок (520) информации слота, выполненный с возможностью предоставления числа слотов кадра и числа слотов с событиями в генератор (530) числа состояний события. Генератор числа состояний события может реализовать один из вышеописанных способов кодирования.

30 В дополнительном варианте осуществления предоставлен кодированный аудиосигнал. Кодированный аудиосигнал содержит число состояний события. В другом варианте осуществления кодированный аудиосигнал, кроме того, содержит число слотов с событиями. Более того, кадр кодированного аудиосигнала может также содержать число слотов кадра. В кадре аудиосигнала позиции слотов, содержащих события в кадре аудиосигнала, могут быть декодированы согласно одному из вышеописанных способов декодирования. В варианте осуществления передаются число состояний события, число слотов с событиями и число слотов кадра, так что позиции слотов, содержащих события в кадре аудиосигнала, могут быть декодированы посредством использования одного из вышеописанных способов.

40 Изобретенный кодированный аудиосигнал может быть сохранен в цифровом запоминающем носителе или некратковременном запоминающем носителе или может быть передан в среде передачи, такой как среда беспроводной передачи или среда проводной передачи, такая как Интернет.

45 Нижеследующее разъясняет синтаксические определения USAC, выполненные с возможностью поддержки управляющего транзиентами декоррелятора (TSD) согласно варианту осуществления.

На фиг.16 проиллюстрированы данные MPS (MPEG Surround) 212. Данные MPS 212 являются блоком данных, содержащих полезные данные для стереомодуля MPS 212.

Данные MPS 212 содержат данные TSD.

На фиг.17 изображен синтаксис данных TSD. Они содержат число слотов с транзидентами ( $bsTsdNumTrSlots$ ) и данные фазы транзиев TSD ( $bsTsdTrPhaseData$ ) для слотов в кадре данных MPS 212. Если слот содержит данные транзиев ( $TsdSepData$  [ts] задано в значение 1),  $bsTsdTrPhaseData$  содержит данные фазы, иначе  $bsTsdTrPhaseData$  [ts] задано в значение 0.

$nBitsTrSlots$  задает число битов, используемых для переноса числа слотов с транзидентами ( $bsTsdNumTrSlots$ ).  $nBitsTrSlots$  зависит от числа слотов в кадре данных MPS 212 ( $numSlots$ ). На фиг.18 проиллюстрирована взаимосвязь числа слотов в кадре данных MPS 212 и числа битов, используемых для переноса числа слотов с транзидентами.

На фиг.19 задается значение  $tempShapeConfig$ .  $tempShapeConfig$  указывает режим работы временного формирования (STP или GES) или активацию управляющей транзидентами декорреляции в декодере. Если  $tempShapeConfig$  задан в значение 0, временное формирование совсем не применяется; если  $tempShapeConfig$  задан в значение 1, применяется временная обработка области поддиапазона (STP); если  $tempShapeConfig$  задан в значение 2, применяется управляемое формирование огибающей (GES); и если  $tempShapeConfig$  задан в значение 3, применяется управляющая транзидентами декорреляция (TSD).

На фиг.20 проиллюстрирован синтаксис данных  $TempShapeData$ . Если  $bsTempShapeConfig$  задан в значение 3,  $TempShapeData$  содержит  $bsTsdEnable$ , указывающий, что TSD включена в кадре.

На фиг.21 проиллюстрирован блок D декоррелятора согласно варианту осуществления. Блок D декоррелятора в блоке ОТТ-декодирования содержит блок разделения сигнала, две структуры декоррелятора и блок объединения сигналов.

$D_{AP}$  означает: частотннезависимый декоррелятор, как задано в подразделе 7.11.2.5 (частотннезависимый декоррелятор).

$D_{TR}$  означает: декоррелятор транзиев.

Если инструмент TSD активен в текущем кадре, т.е. если ( $bsTsdEnable==1$ ), входной сигнал разделяется на поток  $v_{X,Tr}^{n,k}$  с транзидентами и поток  $v_{X,nonTr}^{n,k}$  без транзиев согласно:

$$v_{X,Tr}^{n,k} = \begin{cases} v_X^{n,k}, & \text{если } TsdSepData(n) = 1, 7 \leq k \\ 0 & \text{, в ином случае} \end{cases}$$

$$v_{X,nonTr}^{n,k} = \begin{cases} 0, & \text{если } TsdSepData(n) = 1, 7 \leq k \\ v_X^{n,k} & \text{, в ином случае} \end{cases}$$

Флаг послотового отделения транзиев  $TsdSepData(n)$  декодируется из кодового слова переменной длины  $bsTsdCodedPos$  посредством  $TsdTrPos\_dec()$ , как описано ниже. Длина кодового слова из  $bsTsdCodedPos$ , т.е.  $nBitsTsdCW$ , вычисляется согласно:

$$nBitsTsdCW = ceil\left(\log_2\left(\frac{bsFrameLength}{bsTsdNumTrSlots + 1}\right)\right)$$

Возвращаясь к фиг.11, на фиг.11 проиллюстрировано декодирование данных отделения слота с транзиентами  $bsTsdCodedPos$  в  $TsdSepData[n]$  согласно варианту осуществления. Массив длины  $numSlots$ , состоящий из '1' для кодированных позиций с транзиентами и '0' для других, задается, как проиллюстрировано на фиг.11.

5 Если инструмент TSD отключен в текущем кадре, т.е. если ( $bsTsdEnable==0$ ), входной сигнал обрабатывается так, как если бы  $TsdSepData(n)=0$  для всех  $n$ .

Составляющие сигнала с транзиентами обрабатываются в структуре декоррелятора транзиентов  $D_{TR}$ , как следует ниже:

$$10 \quad d_{X,Tr}^{n,k} = \begin{cases} e^{j\varphi} \cdot v_{X,Tr}^{n,k} & , \text{ если } bsTsdEnable = 1 \\ 0 & , \text{ в ином случае} \end{cases} ,$$

где

$$15 \quad \varphi_{TSD}^n = \pi \cdot 0.25 \cdot bsTsdTrPhaseData(n) .$$

Составляющие сигнала без транзиентов обрабатываются в частотнонезависимом декорреляторе  $D_{AP}$ , как задано в следующем подразделе, выдающем выходной сигнал декоррелятора для составляющих сигнала без транзиентов

$$d_{X,nonTr}^{n,k} = \mathbf{D}_{AP} \left\{ v_{X,nonTr}^{n,k} \right\} .$$

25 Выходные сигналы декоррелятора складываются для образования декоррелированного сигнала, содержащего составляющие как с транзиентами, так и без транзиентов

$$30 \quad d_X^{n,k} = d_{X,Tr}^{n,k} + d_{X,nonTr}^{n,k} .$$

На фиг.22 проиллюстрирован синтаксис  $EcData$ , содержащего  $bsFrequencyResStrideXXX$ . Синтаксический элемент  $bsFreqResStride$  обеспечивает возможность для использования широкополосных меток в MPS. XXX должно быть заменено значением типа данных (CLD, ICC, IPD).

35 Управляющий транзиентами декоррелятор в структуре OTT-декодера предоставляет возможность применения специализированного декоррелятора к составляющим с транзиентами подобных аплодисментам сигналов. Активация этой функции TSD управляется сгенерированным кодером флагом  $bsTsdEnable$ , который передается раз за кадр.

40 Данные TSD в модуле преобразования двух каналов в один канал (R-OTT) кодера генерируются, как следует ниже:

- Запустить семантический классификатор сигналов, который обнаруживает подобные аплодисментам сигналы. Результат классификации передается раз за кадр: флаг  $bsTsdEnable$  задается в значение 1 для подобных аплодисментам сигналов, иначе он задается в значение 0.

- Если  $bsTsdEnable$  задан в значение 0 для текущего кадра, для этого кадра дополнительные данные TSD не генерируются/не передаются.

- Если  $bsTsdEnable$  задан в значение 1 для текущего кадра, выполнить следующее:

- Включить широкополосный расчет пространственных ОТТ-параметров.
- Обнаружить транзиенты в текущем кадре (двоичное решение по каждому временному слоту MPS).

○ Кодировать позиции слотов с транзиентами  $tsdPosLen$  в векторе  $tsdPos$  согласно следующему псевдокоду, где позиции слотов в  $tsdPos$  предполагаются в возрастающем порядке. На фиг.13 проиллюстрирован псевдокод для кодирования позиций слотов с транзиентами в  $tsdPosLen$ .

○ Передать число слотов с транзиентами ( $bsTsdNumTrSlots=(\text{число обнаруженных слотов с транзиентами})-1$ ).

○ Передать закодированные позиции с транзиентами ( $bsTsdCodedPos$ ).

○ Для каждого слота с транзиентами вычислить величину фазы, которая представляет разность фаз при широкополосной передаче между сигналом, полученным понижающим микшированием, и остаточным сигналом.

○ Для каждого слота с транзиентами кодировать и передать величину разности фаз при широкополосной передаче ( $bsTsdTrPhaseData$ ).

Наконец, на фиг.23 проиллюстрирована схема прохождения сигналов для генерирования данных TSD в модуле преобразования двух каналов в один канал (R-ОТТ).

Хотя некоторые аспекты были описаны в контексте устройства, ясно, что эти аспекты также представляют описание соответствующего способа, где блок или устройство соответствует этапу способа, или признака этапа способа. Аналогично, аспекты, описанные в контексте этапа способа, также представляют описание соответствующего блока, или элемента, или признака соответствующего устройства.

В зависимости от определенных требований реализации, варианты осуществления данного изобретения могут быть реализованы в аппаратном обеспечении или программном обеспечении. Данная реализация может быть выполнена, используя цифровой запоминающий носитель, например флорпи-диск, DVD, CD, ROM, PROM, EPROM, EEPROM или флэш-память, содержащий электронно-считываемые сигналы управления, хранящиеся на нем, которые взаимодействуют (или способны взаимодействовать) с программируемой компьютерной системой так, чтобы выполнялся соответствующий способ.

Некоторые варианты осуществления согласно данному изобретению содержат носитель данных, содержащий электронно-считываемые сигналы управления, которые способны взаимодействовать с программируемой компьютерной системой так, чтобы выполнялся один из способов, описанных в настоящем документе.

В общем, варианты осуществления настоящего изобретения могут быть реализованы как компьютерный программный продукт с программным кодом, причем программный код функционирует для выполнения одного из способов, когда компьютерный программный продукт выполняется на компьютере. Программный код может, например, храниться на машиночитаемом носителе.

Другие варианты осуществления содержат компьютерную программу для выполнения одного из способов, описанных в настоящем документе, хранящихся на машиночитаемом носителе или в некрatkовременном запоминающем носителе.

Другими словами, вариант осуществления изобретенного способа вследствие этого является компьютерной программой, содержащей программный код для выполнения одного из способов, описанных в настоящем документе, когда компьютерная программа выполняется на компьютере.

Дополнительный вариант осуществления изобретенных способов вследствие этого

является носителем данных (или цифровым запоминающим носителем или компьютерно-читаемым носителем), содержащим записанную на нем компьютерную программу для выполнения одного из способов, описанных в настоящем документе.

5 Дополнительный вариант осуществления изобретенного способа вследствие этого является потоком данных или последовательностью сигналов, представляющими компьютерную программу для выполнения одного из способов, описанных в настоящем документе. Поток данных или последовательность сигналов могут, например, быть выполненными с возможностью быть перенесенными через соединение передачи данных, например через Интернет.

10 Дополнительный вариант осуществления содержит средство обработки, например компьютер или программируемое логическое устройство, сконфигурированное с возможностью или выполненное с возможностью выполнения одного из способов, описанных в настоящем документе.

15 Дополнительный вариант осуществления содержит компьютер, содержащий установленную на нем компьютерную программу для выполнения одного из способов, описанных в настоящем документе.

В некоторых вариантах осуществления программируемое логическое устройство (например, программируемая пользователем вентиляционная матрица) может быть использовано для выполнения некоторых или всех функциональных возможностей 20 способов, описанных в настоящем документе. В некоторых вариантах осуществления программируемая пользователем вентиляционная матрица может взаимодействовать с микропроцессором для выполнения одного из способов, описанных в настоящем документе. В общем, способы предпочтительно выполняются любым аппаратным устройством.

25 Вышеописанные варианты осуществления являются лишь иллюстративными для принципов настоящего изобретения. Следует понимать, что модификации и вариации данных компоновок и подробности, описанные в настоящем документе, будут очевидны специалистам в данной области техники. Соответственно, подразумевается, что изобретение следует ограничивать только объемом прилагаемой формулы изобретения, 30 но не конкретными подробностями, представленными с целью описания и разъяснения вариантов осуществления в настоящем документе.

#### Список литературы

[1] J. Breebaart, S. van de Par, A. Kohlrausch, E. Schuijers, "High-Quality Parametric Spatial Audio Coding at Low Bitrates" in Proceedings of the AES 116<sup>th</sup> Convention, Berlin, Preprint 35 6072, May 2004.

[2] J. Herre, K. Kjolring, J. Breebaart et al., "MPEG surround - the ISO/MPEG standard for efficient and compatible multi-channel audio coding", in Proceedings of the 122<sup>th</sup> AES Convention, Vienna, Austria, May 2007.

40 [3] Pulkki, Ville: "Spatial Sound Reproduction with Directional Audio Coding", в J. Audio Eng. Soc., Vol. 55, № 6, 2007.

[4] ISO/IEC International Standard "Information Technology - MPEG audio technologies - Part1: MPEG Surround", ISO/IEC 23003-1:2007.

45 [5] J. Engdegard. H. Purnhagen, J. Roden, L. Liljeryd, "Synthetic Ambience in Parametric Stereo Coding" in Proceedings of the AES 116<sup>th</sup> Convention, Berlin, Preprint, May 2004.

#### Формула изобретения

1. Устройство (10; 40; 60; 410) для декодирования кодированного аудиосигнала,

содержащего кадр аудиосигнала, содержащий слоты и события, связанные с этими слотами, содержащее:

анализирующий блок (20; 42; 70; 420) для анализа числа слотов кадра, указывающего общее число слотов кадра аудиосигнала, числа слотов с событиями, указывающего число слотов, содержащих события кадра аудиосигнала, и числа состояний события;

и блок (30; 45; 80; 430) генерирования для генерирования указания множества позиций слотов, содержащих события в кадре аудиосигнала, с использованием числа слотов кадра, числа слотов с событиями и числа состояний события.

2. Устройство (10; 40; 60; 410) для декодирования по п. 1, при этом устройство (10; 40; 60; 410) для декодирования выполнено с возможностью декодирования позиций слотов с транзиентами в кадре аудиосигнала.

3. Устройство (10; 40; 60; 410) для декодирования по п. 1, в котором анализирующий блок (20; 42; 70; 420) выполнен с возможностью проведения проверки, сравнивающей число состояний события или обновленное число состояний события с пороговым значением.

4. Устройство (10; 40; 60; 410) для декодирования по п. 3, в котором анализирующий блок (20; 42; 70; 420) выполнен с возможностью проведения проверки посредством сравнения, является

ли число состояний события или обновленное число состояний события большим, большим или равным, меньшим или меньшим или равным пороговому значению, и в котором блок (30; 45; 80; 430) генерирования дополнительно выполнен с возможностью обновления числа состояний события или обновленного числа состояний события в зависимости от результата проверки.

5. Устройство (10; 40; 60) для декодирования по п. 3, при этом устройство (10; 40; 60) для декодирования дополнительно содержит селектор (90) слотов,

при этом селектор (90) слотов выполнен с возможностью выбора слота в качестве рассматриваемого слота,

при этом анализирующий блок (20; 42; 70) выполнен с возможностью проведения проверки по отношению к рассматриваемому слоту,

и при этом пороговое значение зависит от числа слотов кадра, числа слотов с событиями и от позиции рассматриваемого слота внутри кадра.

6. Устройство (10; 40) для декодирования по п. 5, в котором анализирующий блок (20; 42; 70) выполнен с возможностью проведения проверки, сравнивающей число состояний события или обновленное число состояния

события с пороговым значением, где пороговое значение составляет  $\binom{N-h-1}{P}$ ,

где N является общим числом слотов кадра аудиосигнала, где P является числом слотов, содержащих события кадра аудиосигнала или рассматриваемого участка кадра аудиосигнала, и где h

является позицией рассматриваемого слота внутри кадра.

7. Устройство (10; 40; 410) для декодирования по п. 1,

при этом устройство для декодирования (10; 40; 410) дополнительно содержит блок (440) разделения кадра,

при этом блок (440) разделения кадра выполнен с возможностью разбиения кадра на первый раздел кадра, содержащий первый набор слотов кадра, и на второй раздел

кадра, содержащий второй набор слотов кадра, и при этом устройство (10; 40; 410) для декодирования дополнительно выполнено с возможностью определения позиций слотов, содержащих события для каждого из разделов кадра в отдельности.

5 8. Устройство (10; 40; 60; 410) для декодирования по п. 1, дополнительно содержащее: процессор (50) аудиосигнала для генерирования выходного аудиосигнала с использованием указания множества позиций слотов, содержащих события в кадре аудиосигнала, с использованием числа слотов кадра, числа слотов с событиями и числа состояний события.

10 9. Устройство (10; 60; 410) для декодирования по п. 8, в котором процессор (50) аудиосигнала выполнен с возможностью генерирования выходного аудиосигнала согласно первому способу, если указание множества позиций слотов, содержащих события, находится в первом состоянии указания, и в котором процессор (50) аудиосигнала выполнен с возможностью генерирования выходного аудиосигнала согласно отличающемуся второму способу, если указание множества 15 позиций слотов, содержащих события, находится во втором состоянии указания, которое отличается от первого состояния указания.

10. Устройство (10; 40; 60; 410) для декодирования по п. 9, в котором процессор (50) аудиосигнала выполнен так, что первый способ содержит этап, на котором для декодирования слота используют декоррелятор (56) транзиевтов, 20 если первое состояние указания указывает, что слот содержит транзиевт, и в котором второй способ содержит этап, на котором для декодирования слота используют второй декоррелятор (54), если второе состояние указания указывает, что слот не содержит транзиевт.

25 11. Устройство (510) для кодирования позиций слотов, содержащих события в кадре аудиосигнала, содержащее:

генератор (530) числа состояний события для кодирования позиций слотов посредством кодирования числа состояний события; и

30 блок (520) информации слота, выполненный с возможностью предоставления числа слотов кадра, указывающего общее число слотов кадра аудиосигнала, и числа слотов с событиями, указывающего число слотов, содержащих события кадра аудиосигнала, в генератор (530) числа состояний события,

при этом число состояний события, число слотов кадра и число слотов с событиями вместе указывают множество позиций слотов, содержащих события в кадре аудиосигнала.

35 12. Устройство (510) для кодирования по п. 11, в котором генератор (530) числа состояний события выполнен с возможностью генерирования числа состояний события посредством прибавления положительного целочисленного значения для каждого слота, содержащего событие.

40 13. Устройство (510) для кодирования по п. 11, в котором генератор (530) числа состояний события выполнен с возможностью генерирования числа состояний события посредством определения первого числа подсостояний события для первого раздела кадра, посредством определения второго числа подсостояний события для второго раздела кадра и посредством объединения первого и второго числа состояний события, чтобы сгенерировать число состояний 45 события.

14. Способ декодирования позиций слотов, содержащих события в кадре аудиосигнала, содержащий этапы, на которых:

анализируют число слотов кадра, указывающее общее число слотов кадра

аудиосигнала, число слотов с событиями, указывающее число слотов, содержащих события кадра аудиосигнала, и число состояний события; и

генерируют указание множества позиций слотов, содержащих события в кадре аудиосигнала, с использованием числа слотов кадра, числа слотов с событиями и числа состояний события.

15. Способ кодирования позиций слотов, содержащих события в кадре аудиосигнала, содержащий этапы, на которых:

принимают или определяют число слотов кадра, указывающее общее число слотов кадра аудиосигнала,

принимают или определяют число слотов с событиями, указывающее число слотов, содержащих события кадра аудиосигнала,

кодируют число состояний события на основе числа состояний события, числа слотов кадра и числа слотов с событиями, так что указание множества позиций слотов, содержащих события в кадре аудиосигнала, может быть кодировано с использованием числа слотов кадра, числа слотов с событиями и числа состояний события.

16. Компьютерно-читаемый носитель, содержащий записанную на нем компьютерную программу, которая, при исполнении в компьютере, предписывает компьютеру выполнять способ декодирования позиций слотов с событиями в кадре аудиосигнала по п. 14.

17. Компьютерно-читаемый носитель, содержащий записанную на нем компьютерную программу, которая, при исполнении в компьютере, предписывает компьютеру выполнять способ кодирования позиций слотов с событиями в кадре аудиосигнала по п. 15.

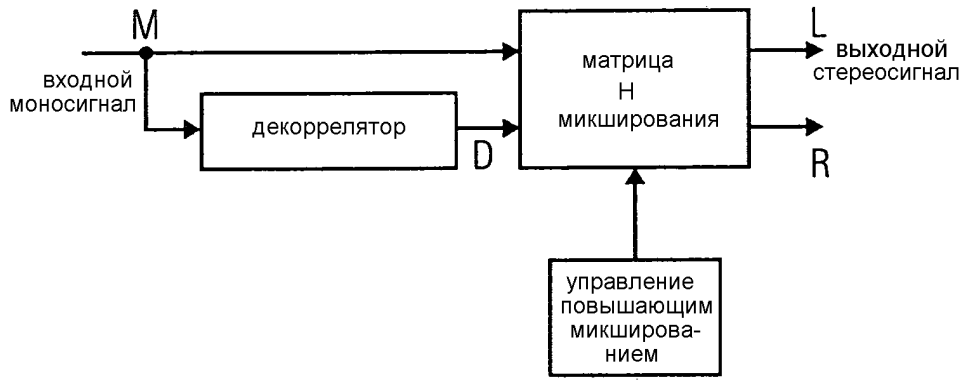
25

30

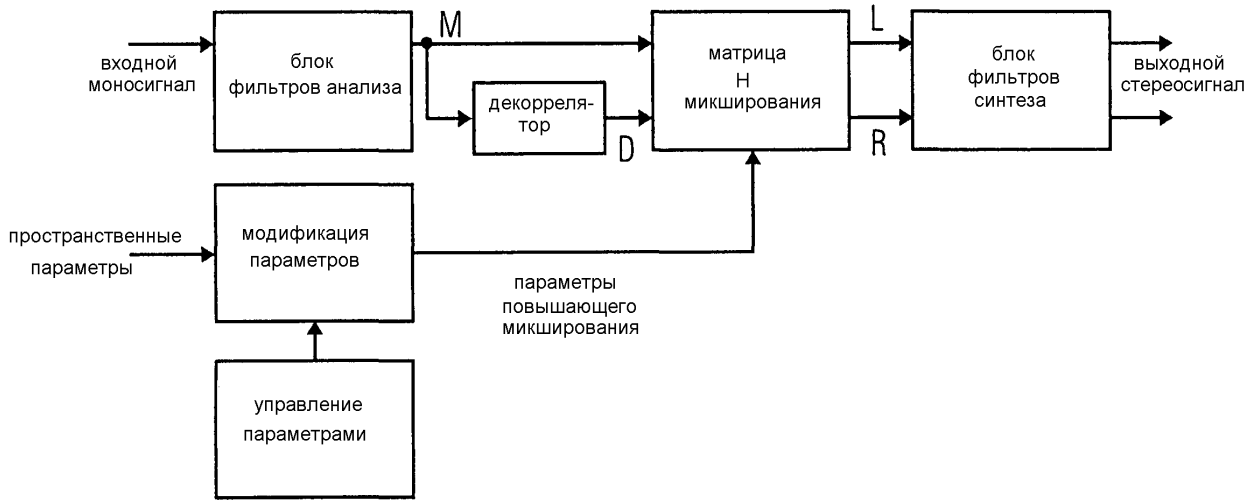
35

40

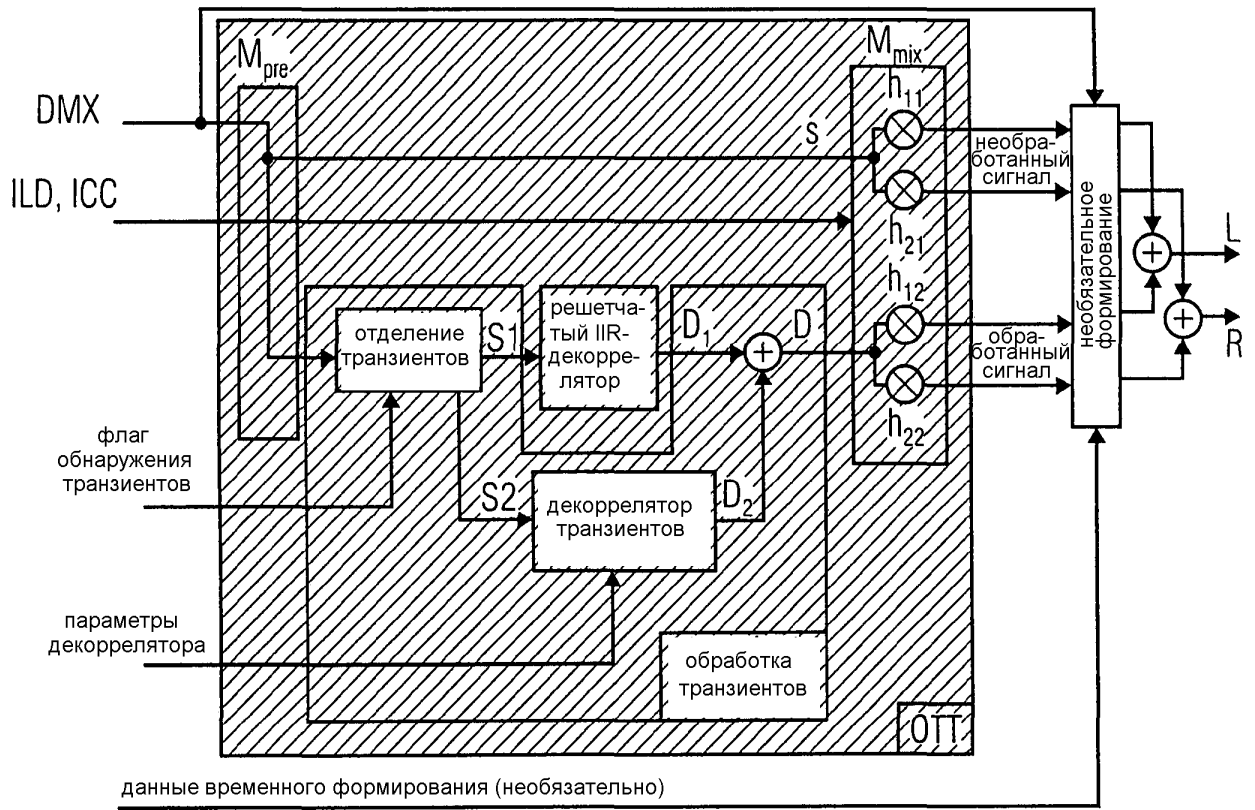
45



Фиг. 1

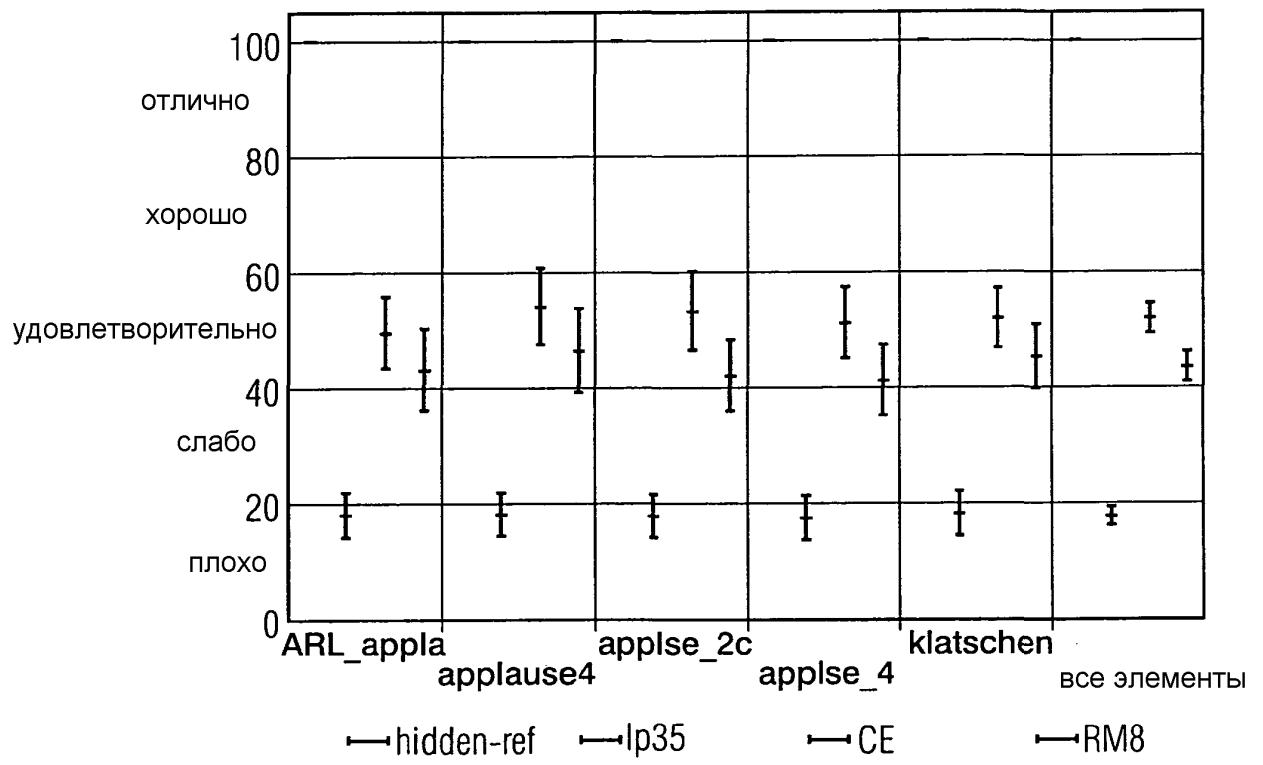


Фиг. 2



Фиг. 3

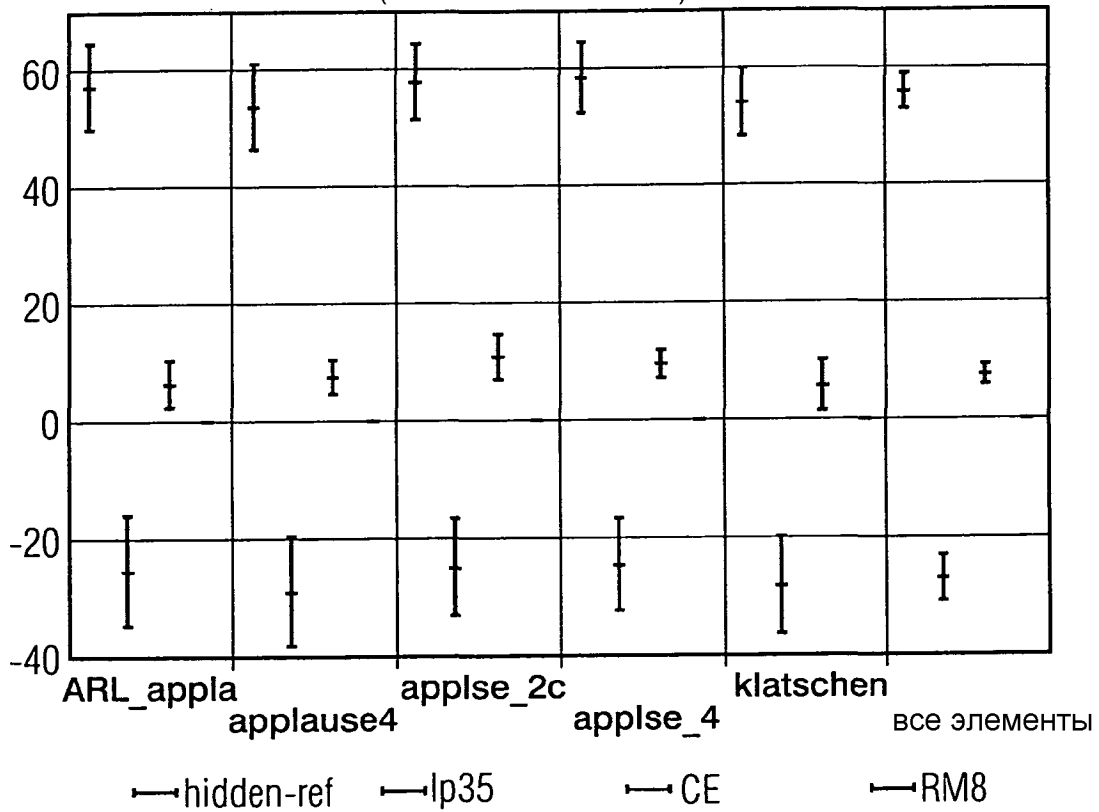
средние значения и 95%-е доверительные интервалы



Фиг. 4

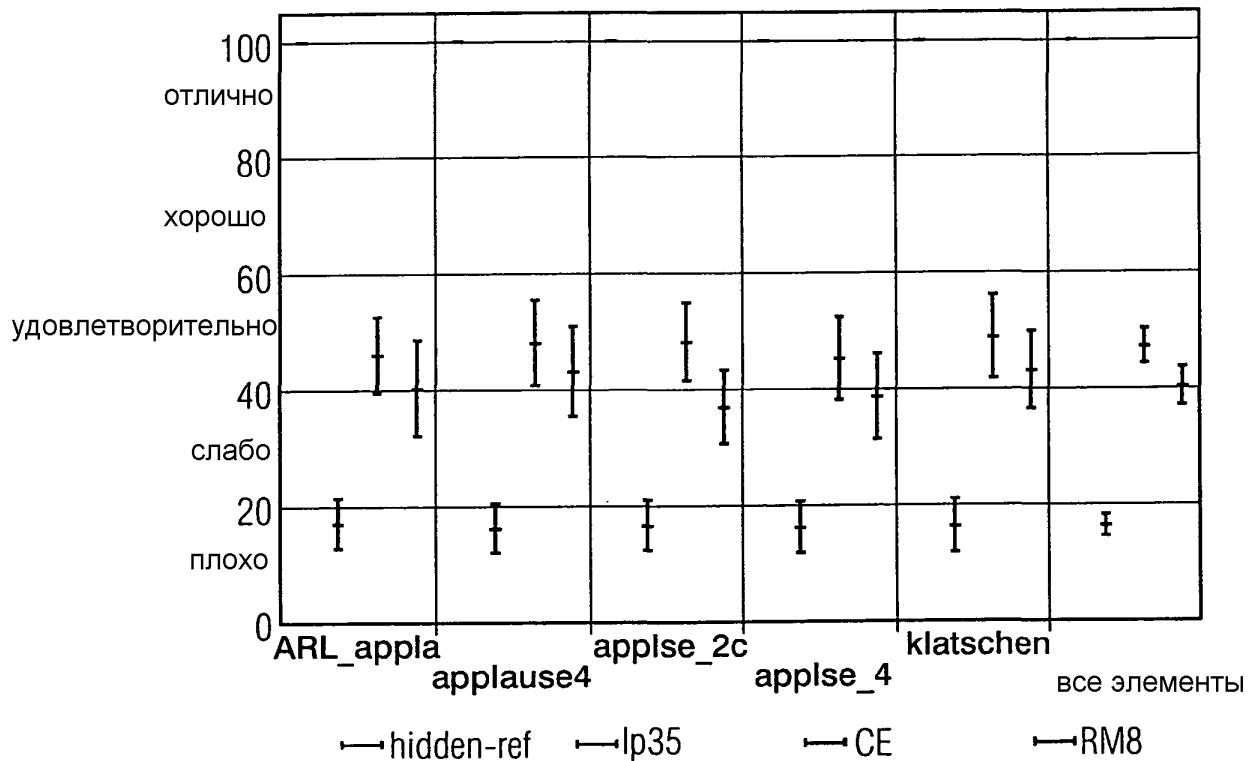
средние значения и 95%-е доверительные интервалы

(отличия: Cond.-RM8)



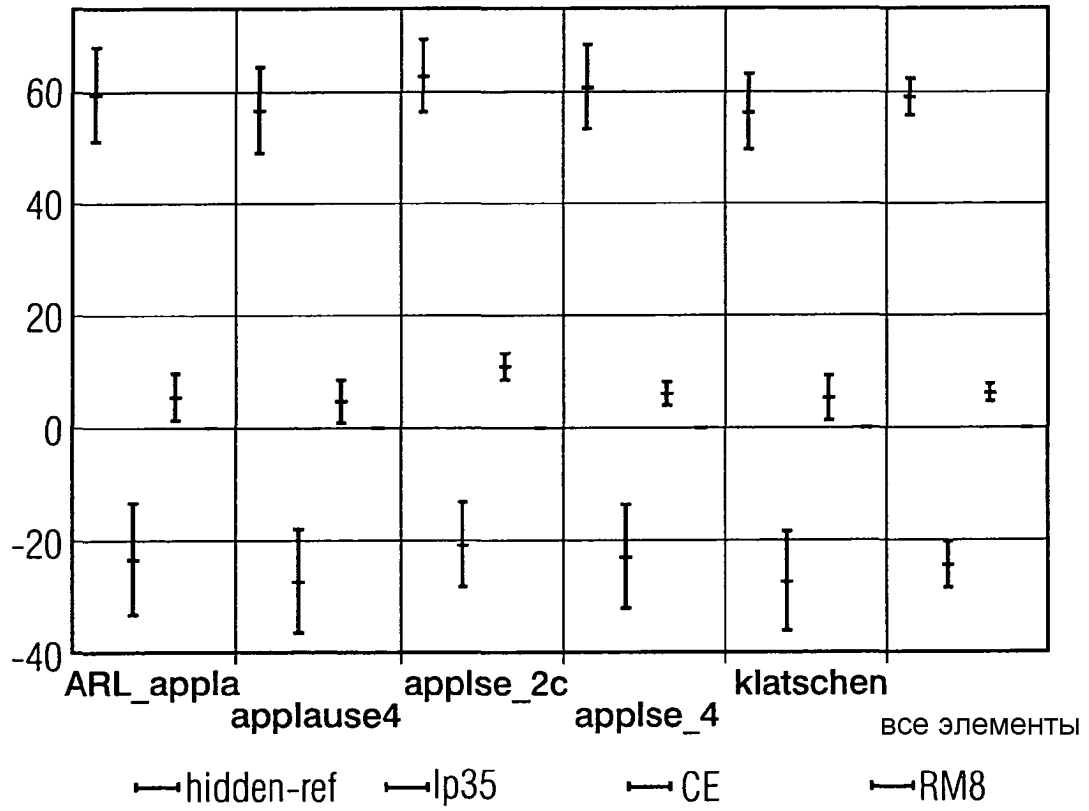
Фиг. 5

средние значения и 95%-е доверительные интервалы

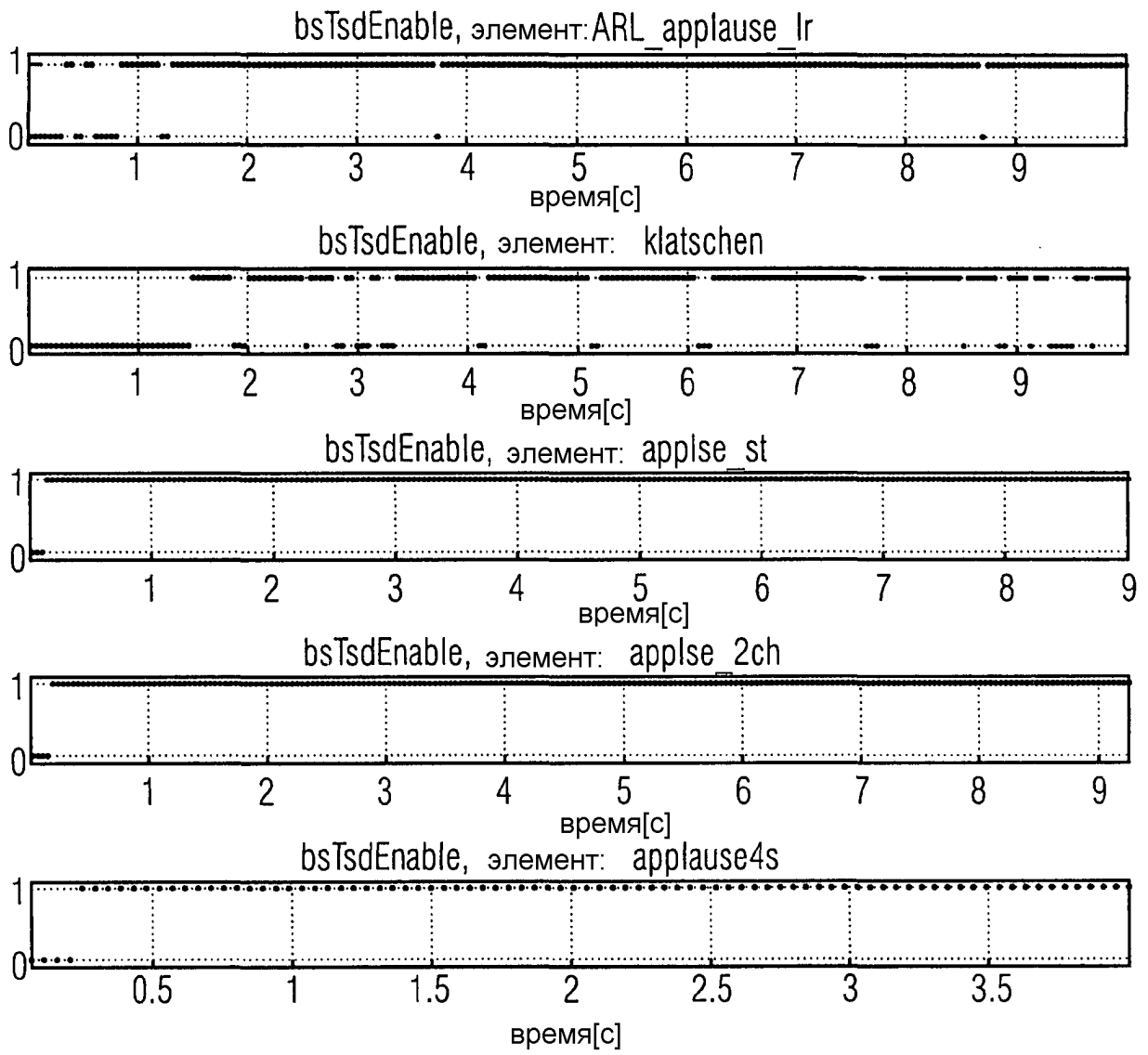


Фиг. 6

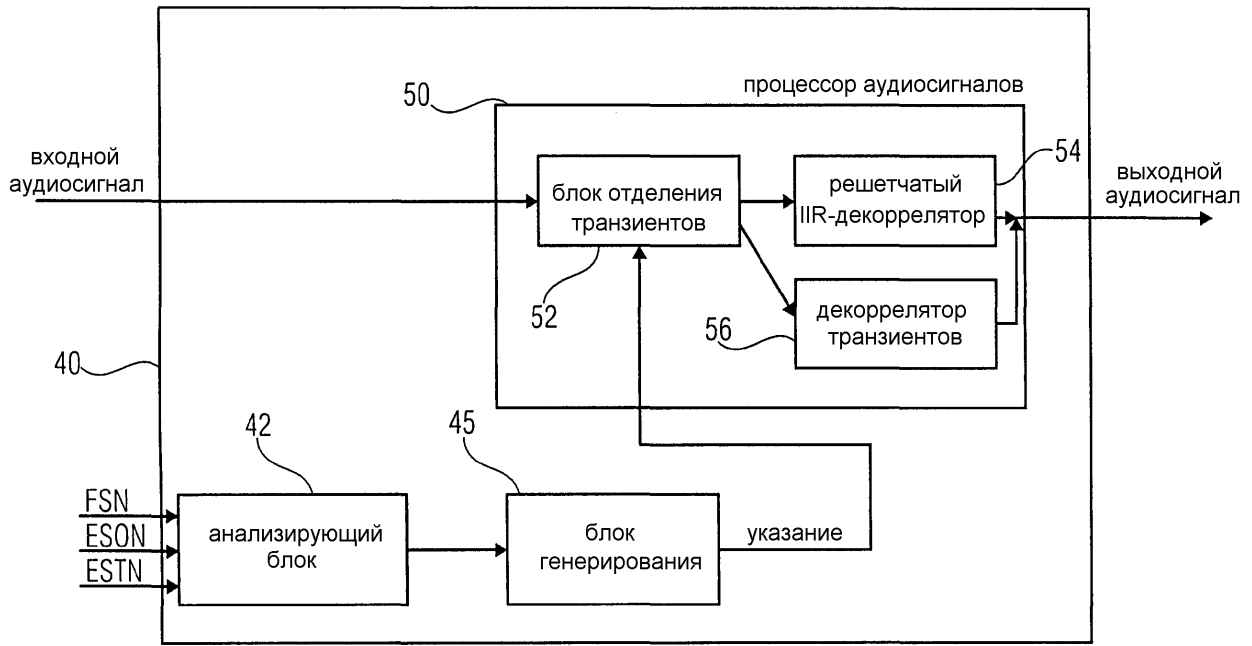
средние значения и 95%-е доверительные интервалы  
(отличия: Cond.-RM8)



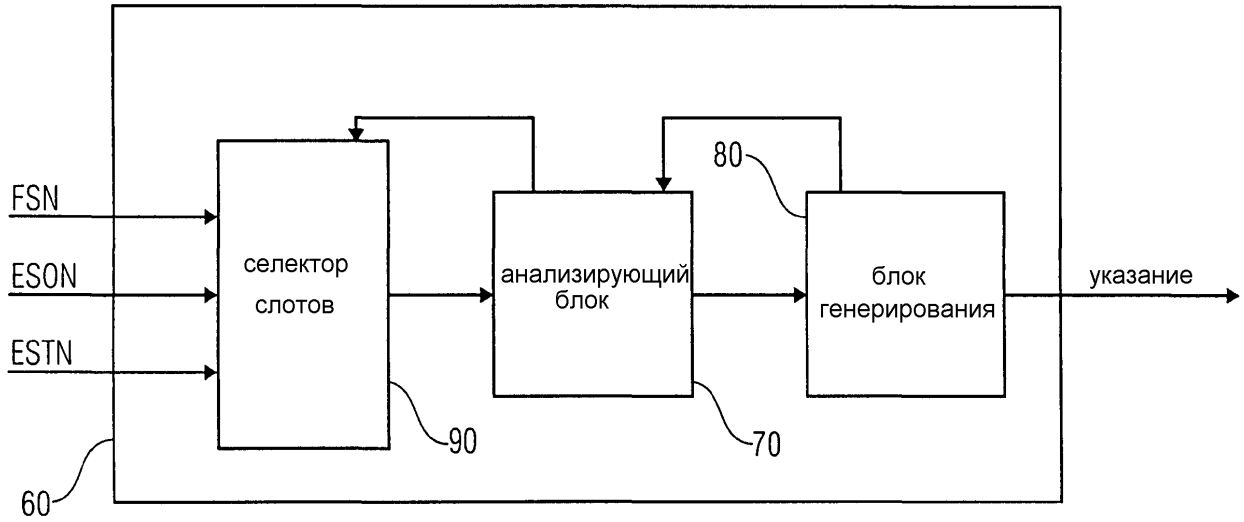
Фиг. 7



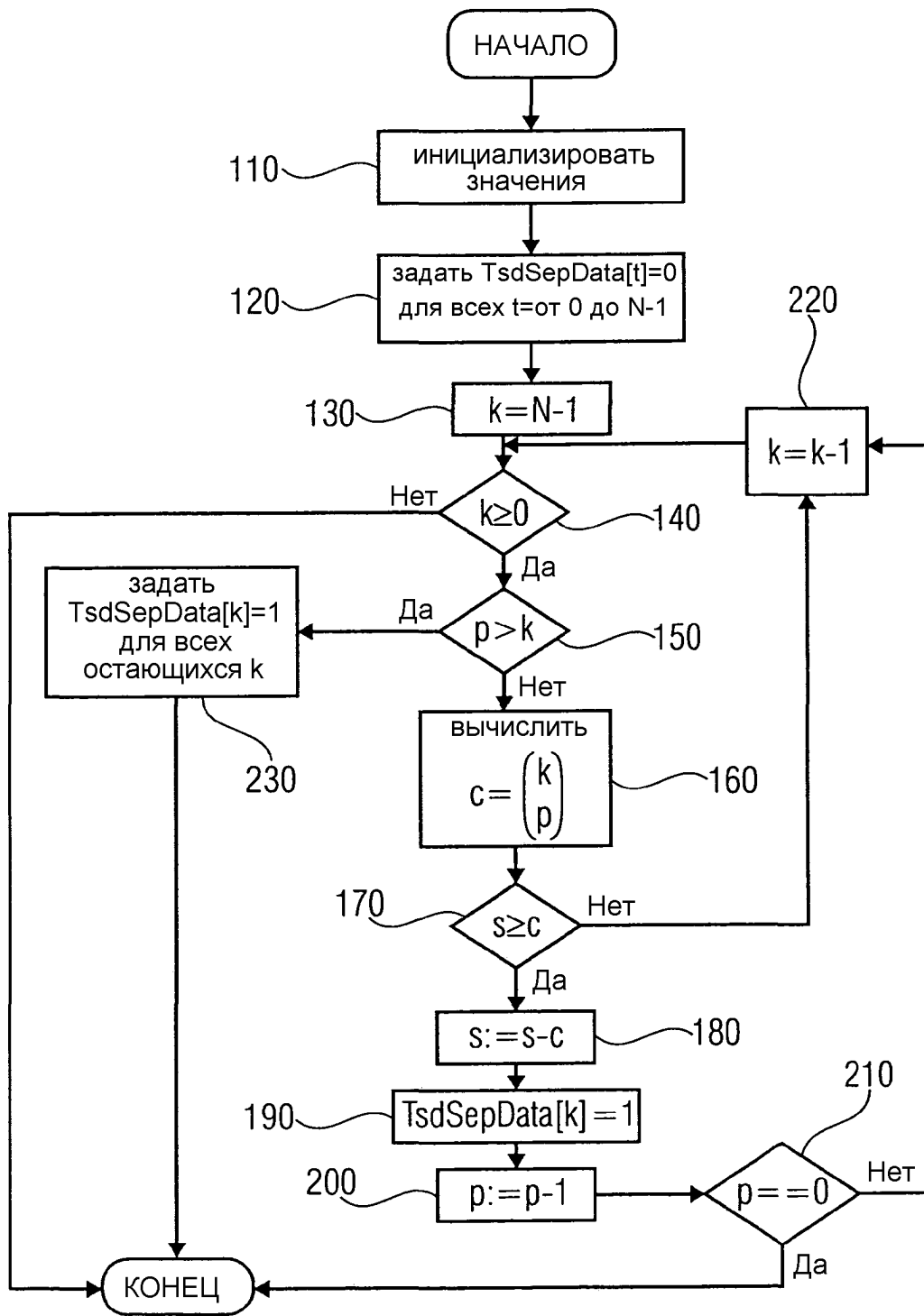
Фиг.8



Фиг. 9В



Фиг. 9С



Фиг. 10

```

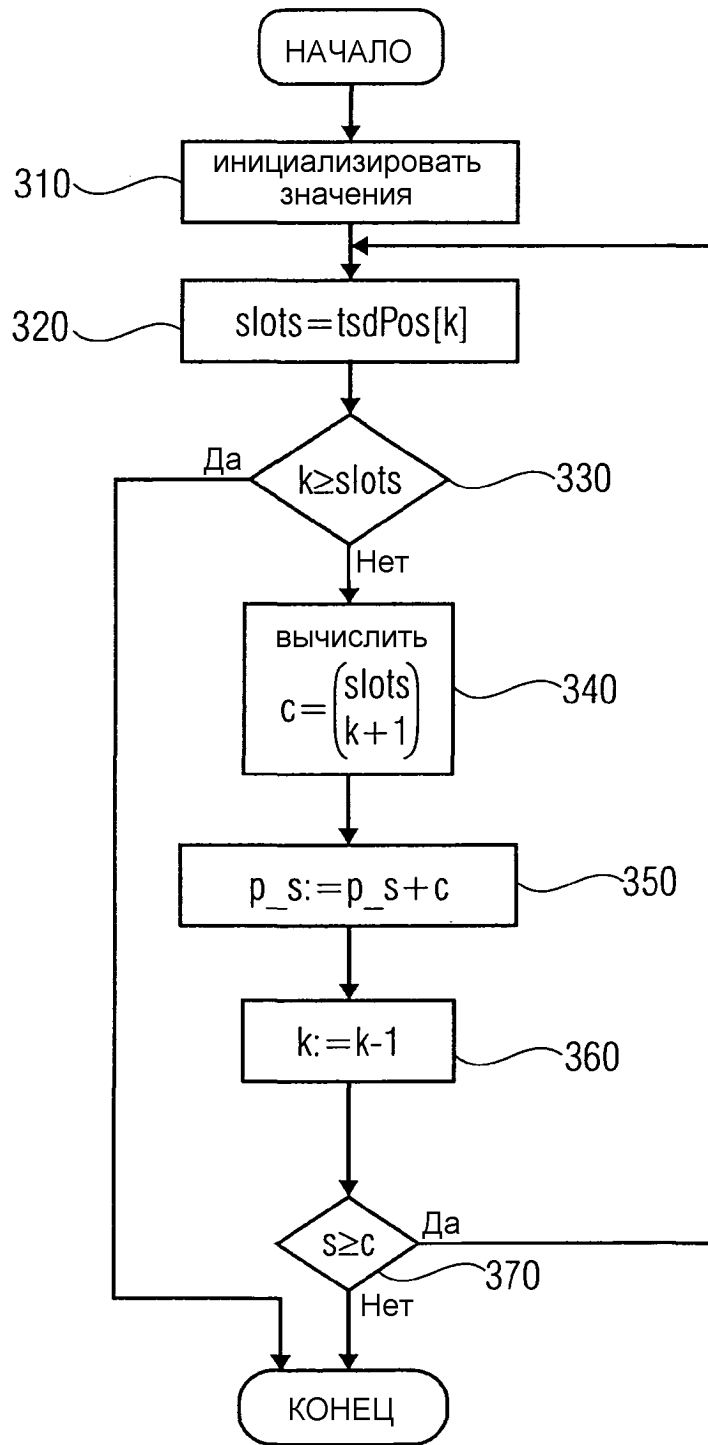
s = bsTsdCodedPos;
p = bsTsdNumTrSlots + 1;
N = numSlots

for (k=0; k<N; k++)
{
    TsdSepData[k]=0;
}

for (k=N-1; k>=0; k--)
{
    if (p > k) {
        for (;k>=0; k--)
            TsdSepData[k]=1;
        break;
    }
    c = k-p+1;
    for (h=2; h<=p; h++) {
        c *= k - p + h;
        c /= h;
    }
    if (s >= (int)c) { /* c is long long for up to 32 slots */
        s -= c;
        TsdSepData[k]=1;
        p--;
        if (p == 0)
            break;
    }
}

```

Фиг. 11



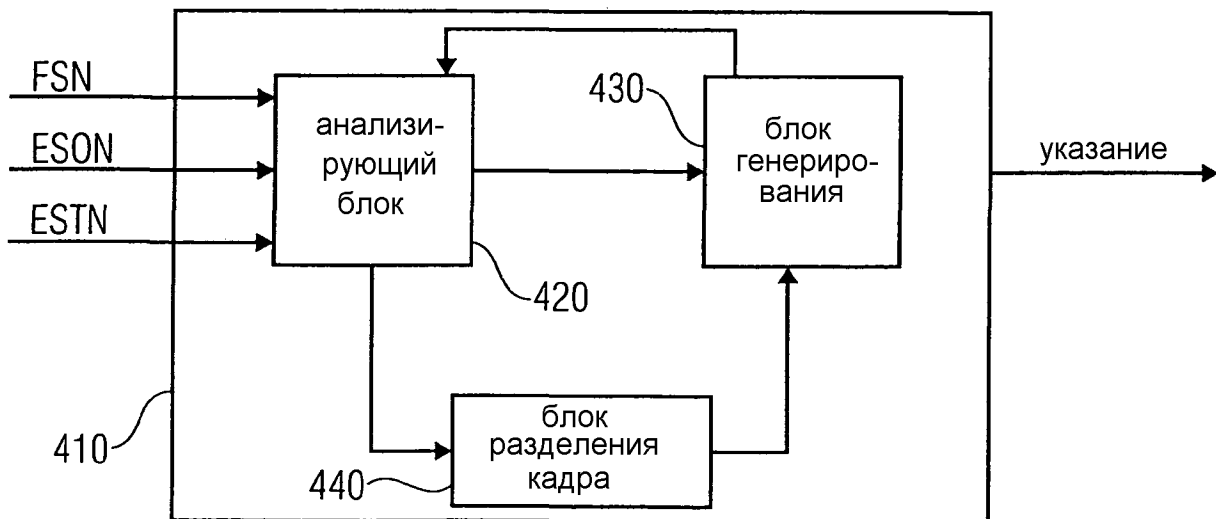
Фиг. 12

```

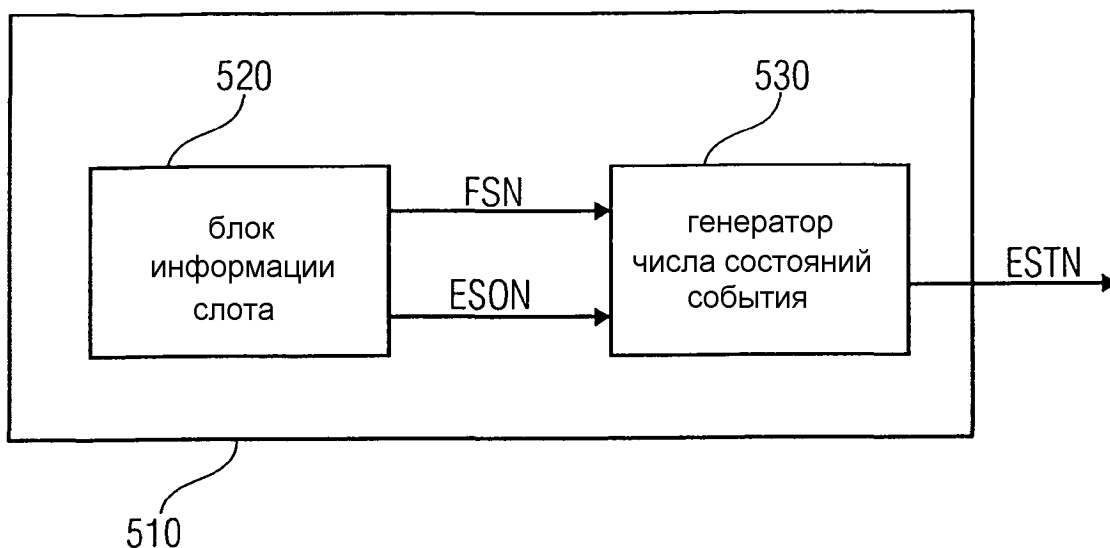
p_s = 0;
for (k=tsdPosLen-1; k >= 0; k--){
    slots = tsdPos[k];
    if (k >= slots) {
        break;
    }
    c = 1; /* c is long long for up to 32 slots */
    for (h=1; h <= (k+1); h++) {
        c *= slots-k-1 + h;
        c /= h;
    }
    p_s += c;
}
bsTsdCodedPos = p_s;

```

Фиг.13



Фиг. 14



Фиг.15

синтаксис	число битов	мнемоника
Mps212Data(indepFlag)		
{		
FrameingInfo();		
if (indepFlag) {		
bsIndependencyFlag=1;		
} else {		
<b>bsIndependencyFlag;</b>	<b>1</b>	<b>uimsbf</b>
}		
OttData();		
SmgData();		
TempShapeData();		
if (bsTsdEnable == 1) {		
TsdData();		
}		
}		

Фиг.16

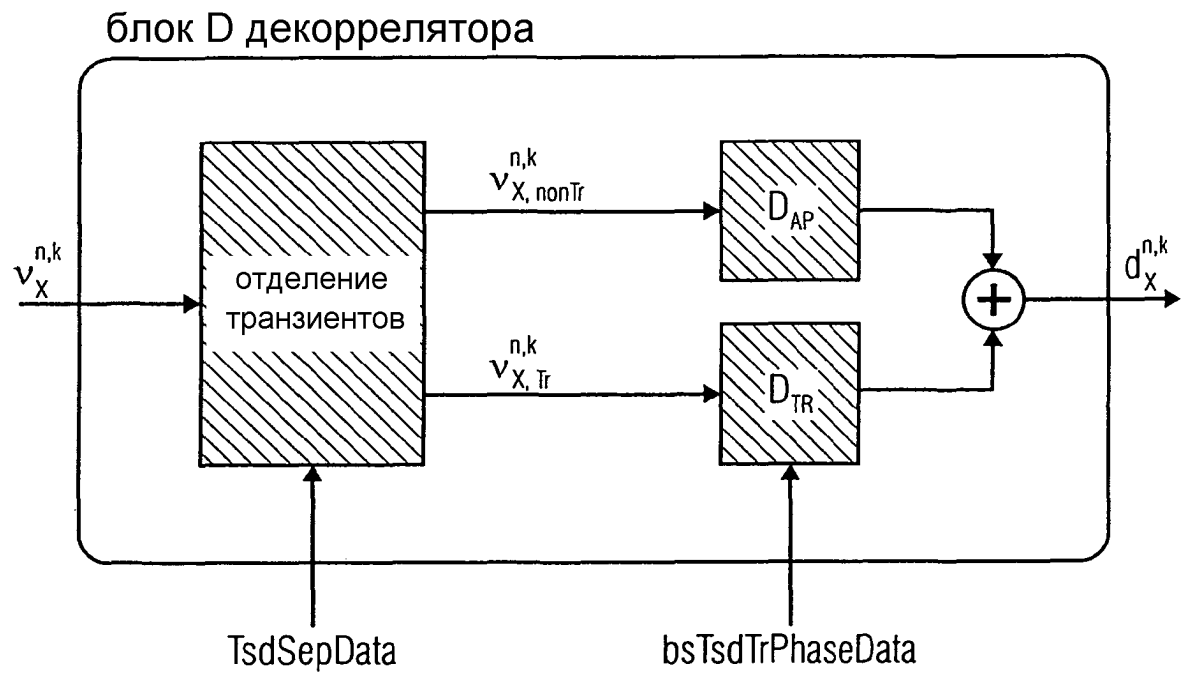


bsTempShapeConfig	Значение
0	не применять временное формирование
1	применить STP
2	применить GES
3	применить TSD

Фиг. 19

СИНТАКСИС	число битов	МНЕМОНИКА
<pre> TempShapeData() {     bsTsdEnable = 0;     if (bsTempShapeConfig == 3) {         <b>bsTsdEnable;</b>     }else if ( (bsTempShapeConfig == 1)    (bsTempShapeConfig == 2) ) {         <b>bsTempShapeEnable;</b>         if (bsTempShapeEnable) {             for (ch=0; ch&lt;numTempShapeChan; ch++) {                 <b>bsTempShapeEnableChannel[ch];</b>             }             if (bsTempShapeConfig == 2) {                 EnvelopeReshapeHuff(bsTempShapeEnableChannel);             }         }     } } </pre>	<p>1</p> <p>1</p> <p>1</p>	<p>uimsbf</p> <p>uimsbf</p> <p>uimsbf</p> <p>примечание 1</p>
примечание 1: numTempShapeChan имеет значение 2, как задано в 6.2.13.2		

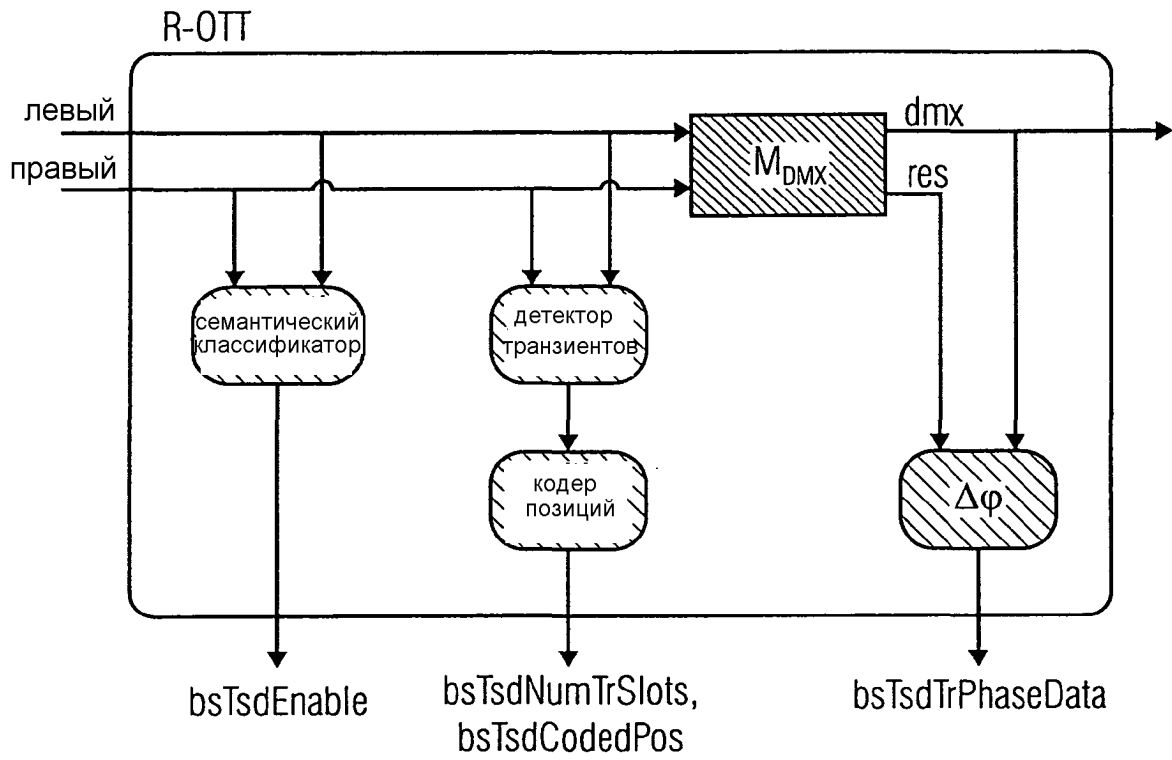
Фиг. 20



Фиг. 21

синтаксис	число битов	мнемоника
EcData(dataType, paramIdx, startBand, stopBand)		примечание 1
{		
dataSets = 0;		
for (ps=0; ps<numParamSets; ps++) {		примечание 2
<b>bsXXXdataMode</b> [paramIdx][ps];	2	<b>uimsbf</b>
if (bsXXXdataMode[paramIdx][ps] == 3) {		
dataSets++;		
}		
}		
setIdx = 0;		
while (setIdx < dataSets) {		
<b>bsDataPairXXX</b> [paramIdx][setIdx];	1	<b>uimsbf</b>
<b>bsQuantCoarseXXX</b> [paramIdx][setIdx];	1	<b>uimsbf</b>
<b>bsFreqResStrideXXX</b> [paramIdx][setIdx];	2	<b>uimsbf</b>
dataBands = (stopBand-startBand-1)/pbStride+1;		примечание 3
EcDataPair(dataType, paramIdx, setIdx, dataBands,		
<b>bsDataPairXXX</b> [paramIdx][setIdx] ,		
<b>bsQuantCoarseXXX</b> [paramIdx][setIdx]);		
if (bsDataPairXXX[paramIdx][setIdx]) {		
bsQuantCoarseXXX[paramIdx][setIdx+1] = bsQuantCoarseXXX[paramIdx][setIdx];		
bsFreqResStrideXXX[paramIdx][setIdx+1] = bsFreqResStrideXXX[paramIdx][setIdx];		
}		
setIdx += bsDataPairXXX[paramIdx][setIdx] + 1;		
}		
startBandXXX[paramIdx] = startBand;		
stopBandXXX[paramIdx] = stopBand;		
}		
<p>примечание 1: XXX должно быть заменено на значение типа данных (CLD, ICC, IPD)</p> <p>примечание 2: numParamSets задано посредством numParamSets= bsNumParamSets+1.</p> <p>примечание 3: pbStride задано в ISO/IEC 23003 - 1:2007, Таблице 70 и зависит от bsFreqResStride[[]].</p> <p>Кроме того, деление должно быть интерпретировано как деление целого числа C ANSI.</p>		

Фиг. 22



Фиг. 23