



УКРАЇНА

(19) UA

(11) 122560

(13) C2

(51) МПК

G10L 19/16 (2013.01)

НАЦІОНАЛЬНИЙ ОРГАН
ІНТЕЛЕКТУАЛЬНОЇ
ВЛАСНОСТІ
ДЕРЖАВНЕ ПІДПРИЄМСТВО
"УКРАЇНСЬКИЙ ІНСТИТУТ
ІНТЕЛЕКТУАЛЬНОЇ
ВЛАСНОСТІ"

(12) ОПИС ДО ПАТЕНТУ НА ВИНАХІД

(21) Номер заявки:	a 2016 06019	(72) Винахідник(и):	Грант Майкл (US), Норкросс Скотт Грегорі (US), Рідміллер Джеффри (US), Вард Майкл (US)
(22) Дата подання заявки:	15.01.2014	(73) Володілець (володільці):	ДОЛБІ ЛАБОРАТОРІС ЛАЙСЕНЗІН КОРПОРЕЙШН, 1275 Market Street, San Francisco, California 94103, USA (US)
(24) Дата, з якої є чинними права інтелектуальної власності:	11.12.2020	(74) Представник:	Михайлюк Валентин Іванович, реєстр. №1
(31) Номер попередньої заявки відповідно до Паризької конвенції:	61/754,882, 61/824,010	(56) Перелік документів, взятих до уваги експертизою:	CA 2998405 A1, 07.06.2012 EP 1878010 A1, 16.01.2008 US 7624021 B2, 24.11.2004 EP 1587063 A2, 19.10.2005 WO 2012075246 A2, 07.06.2012
(32) Дата подання попередньої заявки відповідно до Паризької конвенції:	21.01.2013, 16.05.2013		
(33) Код держави-учасниці Паризької конвенції, до якої подано попередню заявку:	US, US		
(41) Публікація відомостей про заявку:	25.11.2016, Бюл.№ 22		
(46) Публікація відомостей про державну реєстрацію:	10.12.2020, Бюл.№ 23		
(62) Номер та дата подання попередньої заявки, з якої виділено заявку, позначену кодом (21):	a201503505, 15.01.2014		

(54) АУДІОКОДЕР І АУДІОДЕКОДЕР З МЕТАДАНИМИ ГУЧНОСТІ ТА ГРАНИЦІ ПРОГРАМИ

(57) Реферат:

Пристрій і способи формування кодованого бітового аудіопотоку включає в тому числі метадані гучності програми та аудіодані в бітовому потоці, а в деяких випадках також метадані границі програми щонайменше в одному сегменті (наприклад, фреймі) бітового потоку. Іншими аспектами є пристрій і способи для декодування такого бітового потоку, наприклад, у тому числі шляхом виконання адаптивної обробки гучності аудіоданих аудіопрограми, що вказана бітовим потоком, або шляхом перевірки дійсності та/або перевірки правильності метаданих і/або аудіоданих такої аудіопрограми. Іншим аспектом є блок обробки звукового сигналу (наприклад, кодер, декодер або постпроцесор), виконаний (наприклад, запрограмований) з можливістю виконання будь-якого варіанта здійснення способу або що включає буферний запам'ятовувальний пристрій, що зберігає щонайменше один фрейм бітового аудіопотоку, сформованого відповідно до будь-якого з варіантів здійснення способу.

UA 122560 C2

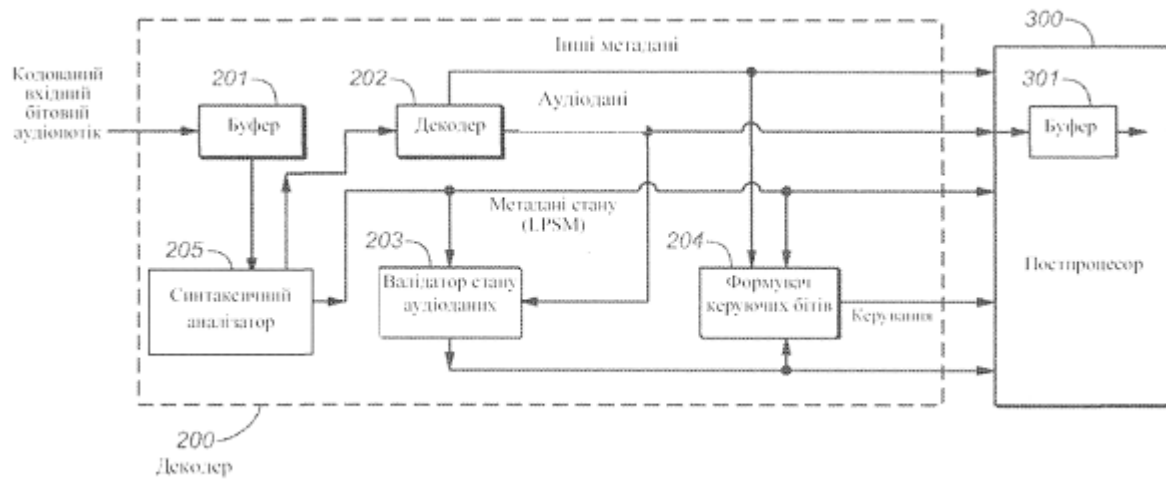


Fig. 3

Дана заявка заявляє пріоритет попередньої заявки на патент США № 61/754882, поданої 21 січня 2013 року, та попередньої заявки на патент США № 61/824010, поданої 16 травня 2013 року, кожна з яких включена в даний опис за допомогою посилання у всій повноті.

Винахід належить до обробки звукових сигналів, а більш конкретно, до кодування та декодування бітових потоків аудіоданих з метаданими, що є індикатором стану обробки гучності аудіоконтенту та розташування границь аудіопрограм, що вказані бітовим потоком. Деякі варіанти здійснення даного винаходу формують або декодують аудіодані в один з форматів, відомих як AC-3, Enhanced AC-3 або E-AC-3, або Dolby E.

Dolby, Dolby Digital, Dolby Digital Plus і Dolby E є товарними знаками Dolby Laboratories Licensing Corporation. Dolby Laboratories забезпечує запатентовані втілення AC-3 і E-AC-3, відомі як Dolby Digital і Dolby Digital Plus, відповідно.

Блоки обробки аудіоданих зазвичай працюють "наосліп" і не звертають увагу на характер протікання процесів з аудіоданими, які відбувалися до одержання даних. Це може працювати за умов обробки аудіоданих, коли один об'єкт повністю виконує обробку й кодування аудіоданих для різних цільових відтворюючих медіа-пристроїв, при тому, що цільовий відтворюючий медіа-пристрій повністю виконує декодування й відтворення кодованих аудіоданих. Проте, така "сліпа" обробка є неефективною (або взагалі непрацездатною) у ситуаціях, коли множина блоків обробки звукового сигналу рознесена по неоднорідній мережі або встановлена послідовно (тобто, у вигляді ланцюга) і, як очікується, повинна оптимально виконувати відповідні типи обробки звукових сигналів. Наприклад, деякі аудіодані можуть бути закодовані для медіасистем високої продуктивності й, можливо, їх доведеться перетворити в спрощену форму, що підходить для мобільного пристрою, що перебуває в тракті обробки медіаінформації. Відповідно, блоку обробки звукового сигналу немає необхідності здійснювати виконаний раніше вид обробки аудіоданих. Наприклад, блок авторегулювання гучності може виконувати обробку вхідного аудіокліпу, незалежно від того, чи було таке ж або аналогічне авторегулювання гучності для цього вхідного аудіокліпу виконане раніше. У результаті блок авторегулювання гучності може виконувати авторегулювання, навіть коли це не потрібно. Така зайва обробка також може бути причиною погіршення та/або видалення характерних особливостей при відтворенні контенту аудіоданих.

Типовий потік аудіоданих містить як аудіоконтент (наприклад, один або більше каналів аудіоконтенту), так і метадані, що вказують щонайменше одну характеристику аудіоконтенту. Наприклад, у бітовому потоці AC-3 є кілька параметрів метаданих звукового сигналу, які спеціально призначені для використання при зміні звуку програми, переданої в середовище прослуховування. Одним з параметрів метаданих є параметр DIALNORM, що призначений для вказівки середнього рівня діалогу, що зустрічається в аудіопрограмі, і використовується для визначення рівня відтворення звукового сигналу.

Під час відтворення бітового потоку, що містить послідовність різних сегментів аудіопроеграми (кожний з яких має різний параметр DIALNORM), AC-3 декодер використовує параметр DIALNORM кожного сегмента, щоб виконати тип обробки гучності, при якому він змінює рівень відтворення або гучність так, що сприймана гучність діалогу послідовності сегментів підтримується на постійному рівні. Кожний кодований аудіосегмент (елемент) у послідовності кодованих аудіоелементів буде (у загальному) мати різні параметри DIALNORM, і декодер буде масштабувати рівень кожного з елементів таким чином, щоб рівень відтворення або гучність діалогу кожного елемента була однаковою або дуже схожою, навіть якщо буде потрібно застосування коефіцієнтів підсилення різної величини для різних елементів під час відтворення.

DIALNORM зазвичай задається користувачем, і не формується автоматично, хоча існує значення DIALNORM за замовчуванням, якщо значення не задане користувачем. Наприклад, творець контенту може виконувати виміри гучності із зовнішнього пристрою відносно AC-3 кодера, а потім передати результат (що показує гучність розмовного діалогу аудіопроеграми) кодеру, щоб установити значення DIALNORM. Таким чином, правильність установки параметра DIALNORM залежить від творця контенту.

Існує кілька різних причин, через які параметр DIALNORM у бітовому потоці AC-3 може бути невірним. По-перше, кожний AC-3 кодер має значення DIALNORM за замовчуванням, що використовується при формуванні бітового потоку, якщо значення DIALNORM не задане творцем контенту. Це значення за замовчуванням може істотно відрізнятися від фактичної гучності діалогу аудіоконтенту. По-друге, навіть якщо творець контенту вимірює гучність і задає значення DIALNORM відповідним чином, алгоритм виміру гучності або вимірювальний прилад можуть не відповідати рекомендованому методу виміру гучності для AC-3, у результаті буде отримане невірне значення DIALNORM. По-третє, навіть якщо бітовий потік AC-3 був створений

із правильно виміряним і заданим творцем контенту значенням DIALNORM, він може бути змінений на невірне значення при передачі та/або зберіганні бітового потоку. Наприклад, у додатках телевізійного мовлення для бітових потоків AC-3, що підлягають декодуванню, модифікації, а потім перекодуванню, нерідким є використання інформації метаданих з неправильним DIALNORM. Таким чином, значення DIALNORM, включене в бітовий потік AC-3, може бути неправильним або неточним й, отже, може негативно впливати на якість звучання.

Крім цього, параметр DIALNORM не показує стан обробки гучності відповідних аудіоданих (наприклад, які тип(и) обробки гучності були виконані з аудіоданими). До даного винаходу, звуковий бітовий потік ніколи не включав метадані, що вказують стан обробки гучності (наприклад, застосований(і) тип(и) обробки гучності) аудіоконтенту бітового аудіопотоку або стан обробки гучності й гучність аудіоконтенту бітового потоку у форматі типу, що описаний у даному винаході. Метадані стану обробки гучності в такому форматі є корисними для забезпечення, зокрема, ефективною адаптивною обробки гучності бітового аудіопотоку та/або перевірки вірогідності стану обробки гучності та гучності аудіоконтенту.

Хоча даний винахід не обмежується використанням бітового потоку AC-3, бітового потоку E-AC-3 або бітового потоку Dolby E, для зручності він буде описаний у варіантах здійснення, у яких він генерує, декодує або іншим способом обробляє такі бітові потоки, які включають метадані стану обробки гучності.

Кодований бітовий потік AC-3 містить метадані та від одного до шести каналів аудіоконтенту. Аудіоконтент - це аудіодані, які були стиснуті з використанням перцепційного аудіокодування. Метадані містять кілька параметрів метаданих аудіоконтенту, які призначені для використання при зміні звучання програм, що передаються у середовище прослуховування.

Докладний опис AC-3 кодування (також відомого як Dolby Digital) добре відомий та викладений в багатьох опублікованих джерелах, у тому числі в наступних:

ATSC Standard A52/A: Digital Audio Compression Standard (AC-3), Revision A, Advanced Television Systems Committee, 20 Aug. 2001; і патенти США № 5583962; 5632005; 5633981; 5727119; і 6021386, кожний з яких включений у даний опис за допомогою посилання у всій своїй повноті.

Докладний опис кодування Dolby Digital Plus (E-AC-3) викладений в "Introduction to Dolby Digital Plus, an Enhancement to the Dolby Digital Coding System", AES Convention Paper 6196, 117th AES Convention, October 28, 2004.

Докладний опис кодування Dolby E викладений в "Efficient Bit Allocation, Quantization, and Coding in an Audio Distribution System", AES Preprint 5068, 107th AES Conference, August 1999 and "Professional Audio Coder Optimized for Use with Video", AES Preprint 5033, 107th AES Conference August 1999.

Кожний фрейм кодованого бітового аудіопотоку AC-3 містить аудіоконтент і метадані для 1536 семплів цифрового звукозапису. Що представляє 32 мілісекунди цифрового звукозапису або швидкість звукозапису 31,25 фреймів у секунду для частоти дискретизації 48 кГц.

Кожний фрейм кодованого бітового аудіопотоку E-AC-3 містить аудіоконтент і метадані для 256, 512, 768 або 1536 семплів цифрового звукозапису залежно від того, чи містить фрейм один, два, три або шість блоків аудіоданих відповідно. Для частоти дискретизації 48 кГц це представляє 5,333, 10,667, 16 або 32 мілісекунди цифрового аудіоапису відповідно або швидкість аудіоапису 189,9, 93,75, 62,5 або 31,25 фреймів у секунду відповідно.

Як показано на фіг. 4, кожний фрейм AC-3 ділиться на розділи (сегменти): розділ синхронізуючої інформації (SI), що містить (як показано на фіг. 5) синхрослово (SW) і перше із двох слів корекції помилок (CRC1); інформаційний розділ бітового потоку (BSI), що містить більшу частину метаданих; шість аудіоблоків (AB0-AB5), які містять дані стиснутого аудіоконтенту (а також можуть включати метадані); сегменти зайвих бітів (W), які містять всі невикористовувані біти, що залишилися після стиснення аудіоконтенту; допоміжний інформаційний розділ (AUX), який може містити додаткові метадані й друге із двох слів корекції помилок (CRC2).

Сегмент зайвих бітів (W) також може згадуватися як "поле ігнорованих даних".

Як показано на фіг. 7, кожний фрейм E-AC-3 ділиться на розділи (сегменти): розділ синхронізуючої інформації (SI), що містить (як показано на фіг. 5) синхрослово (SW); інформаційний розділ бітового потоку (BSI), що містить більшу частину метаданих; від одного до шести аудіоблоків (AB0 - AB5), які містять дані стиснутого аудіоконтенту (а також можуть містити метадані); сегмент зайвих бітів (W), що містить всі невикористовувані біти, що залишилися після стиснення аудіоконтенту (хоча показано тільки один сегмент зайвих бітів, як правило, за кожним аудіоблоком йдуть слідом інші сегменти зайвих бітів); допоміжний

інформаційний розділ (AUX), що може містити додаткові метадані та слово корекції помилок (CRC). Сегмент зайвих бітів (W) також може згадуватися як "поле ігнорованих даних".

У бітовому потоці AC-3 (або E-AC-3) є кілька параметрів метаданих звукозапису, які спеціально призначені для використання при зміні звучання програми, що передається в середовище прослуховування. Одним з параметрів метаданих є параметр DIALNORM, що входить у сегмент BSI.

Як показано на фіг. 6, сегмент BSI фрейму AC-3 містить п'ятибітовий параметр ("DIALNORM"), що вказує значення DIALNORM для програми. П'ятибітовий параметр ("DIALNORM2"), що вказує передане в тому ж фреймі AC-3 значення DIALNORM для другої аудіопрограми, міститься в бітовому потоці, якщо режим аудіокодування ("astmod") фрейму AC-3 дорівнює "0", вказуючи використання конфігурації каналу "дуальне моно" або "1+1".

Сегмент BSI також містить прапор ("addbsie"), що вказує наявність (або відсутність) додаткової інформації бітового потоку після біта "addbsie", параметр ("addbsil"), що вказує довжину будь-якої додаткової інформації бітового потоку, що йде слідом за значенням "addbsil", а також до 64 біт додаткової інформації потоку ("addbsi"), що йде слідом за значенням "addbsil".

Сегмент BSI містить інші значення метаданих, зокрема, не показаних на фіг. 6.

В одному класі варіантів здійснення даних винахід являє собою блок обробки звукового сигналу, що включає буферний запам'ятовувальний пристрій, аудіодекодер і синтаксичний аналізатор. Буферний запам'ятовувальний пристрій зберігає щонайменше один фрейм кодованого бітового аудіопотоку. Кодований бітовий аудіопотік включає аудіодані та контейнер метаданих. Контейнер метаданих містить заголовок, одне або кілька інформаційних наповнень метаданих і захисні дані. Заголовок включає синхрослово, що ідентифікує початок контейнера. Одне або кілька інформаційних наповнень метаданих описують аудіопрограму, пов'язану з аудіоданими. Захисні дані розташовуються після одного або декількох інформаційних наповнень метаданих. Захисні дані також можуть бути використані для перевірки цілісності контейнера метаданих і одного або декількох інформаційних наповнень у контейнері метаданих. Аудіодекодер підключений до буферного запам'ятовувального пристрою та здатний декодувати аудіодані. Синтаксичний аналізатор підключений до або інтегрований в аудіодекодер і здатний виконувати синтаксичний аналіз контейнера метаданих.

У типових варіантах здійснення винаходу спосіб включає прийом кодованого бітового аудіопотоку, причому кодований бітовий аудіопотік сегментований на один або кілька фреймів. Аудіодані, поряд з контейнером метаданих, добувають із кодованого бітового аудіопотоку. Контейнер метаданих містить заголовок з наступним одним або декількома інформаційними наповненнями метаданих, за якими йдуть захисні дані. І, нарешті, цілісність контейнера й одного або декількох інформаційних наповнень метаданих перевіряється за допомогою використання захисних даних. Одне або кілька інформаційних наповнень метаданих можуть включати інформаційне наповнення гучності програми, що містить дані, що вказують вимірювану гучність аудіопрограми, пов'язаної з аудіоданими.

Інформаційне наповнення метаданих гучності програми називають метаданими стану обробки гучності ("LPSM"), вбудованими в бітовий аудіопотік, для яких, відповідно до типових варіантів здійснення винаходу, може бути встановлена автентичність і підтверджена вірогідність, наприклад, для того, щоб об'єкти, що регулюють гучність, могли переконатися в тому, що гучність конкретної програми перебуває в межах встановленого діапазону, і що відповідні аудіодані не змінилися (забезпечуючи тим самим дотримання діючих регулюючих вимог). Для підтвердження цього замість повторного обчислення гучності може бути зчитане значення гучності, що міститься в блоці даних, який містить метадані стану обробки гучності. Завдяки LPSM, регулювальний орган без необхідності обчислення гучності аудіоконтенту може визначити, що відповідний аудіоконтент перебуває у відповідності (як вказано LPSM) із законодавчо встановленим рівнем гучності та/або з нормативними вимогами (наприклад, постановами, прийнятими відповідно до Закону про зменшення гучності комерційної реклами, також відомим як Закон "CALM").

Вимірювання гучності, які необхідні для дотримання законодавчо встановленого рівня гучності та/або деяких нормативних вимог (наприклад, постанов, прийнятих відповідно до Закону про зменшення гучності комерційної реклами), ґрунтуються на інтегральній гучності програми. Інтегральна гучність програми вимагає, щоб вимірювання гучності або рівня гучності діалогу, або рівня гучності змішаного звукозапису проводилися протягом всієї аудіопрограми. Таким чином, дуже важливо знати, які аудіодані (і метадані) визначають всю аудіопрограму, щоб виконати вимірювання гучності програми (наприклад, на різних етапах у віщальному ланцюзі) для перевірки відповідності дійсним вимогам законодавства, а це, як правило, вимагає

знання місця розташування початку й кінця програми (наприклад, під час обробки бітового потоку, що вказує послідовність аудіопрограм).

Відповідно до типових варіантів здійснення даного винаходу кодований бітовий аудіопотік вказує щонайменше одну аудіопрограму (наприклад, послідовність аудіопрограм), а метадані
 5 границі програми та LPSM, включені в бітовий потік, дозволяють скинути виміри гучності програми наприкінці програми й, тим самим, забезпечити автоматизований спосіб вимірювання інтегральної гучності програми. Типові варіанти здійснення даного винаходу включають ефективні метадані границі програми в кодованому бітовому аудіопотоці, що дозволяють виконати точне й надійне визначення щонайменше однієї границі між послідовними
 10 аудіопрограмми, вказаними бітовим потоком. Типові варіанти здійснення забезпечують точне й надійне визначення границі програми в тому розумінні, що вони дозволяють точно визначити границю програми, навіть у тих випадках, коли бітові потоки, що вказують різні програми, змонтовані один з одним (для формування бітового потоку відповідно до винаходу) таким чином, що обрізаний один або обидва змонтовані бітові потоки (і, таким чином, добуті метадані
 15 границі програми, які входили щонайменше в один з бітових потоків до монтажу).

У типових варіантах здійснення, метадані границі програми у фреймі бітового потоку; відповідно до винаходу, являють собою прапор границі програми, що вказує число фреймів. Як правило, прапор вказує кількість фреймів між поточним фреймом (фреймом, що містить прапор) і границею програми (початком або кінцем поточної аудіопрограми). У деяких
 20 переважних варіантах здійснення винаходу прапори програми розставляють симетрично, ефективним способом на початку й наприкінці кожного сегмента бітового потоку, що вказує одну програму (тобто у фреймах, що зустрічаються протягом деякого заданого числа фреймів після початку сегмента, і у фреймах, що зустрічаються протягом деякого заданого числа фреймів до кінця сегмента), таким чином, коли два таких сегменти бітового потоку з'єднуються (тобто буде присутня ознака послідовності двох програм), метадані границі програми можуть бути присутні
 25 (наприклад, симетрично) на обох сторонах границі між двома програмами.

Щоб обмежити збільшення швидкості передачі даних, що є результатом включення метаданих границі програми в кодований бітовий аудіопотік (який може містити ознаки однієї аудіопрограми або послідовності аудіопрограм), у типових варіантах здійснення винаходу
 30 прапори границь програми вставляють тільки в підмножині фреймів бітового потоку. Як правило, коефіцієнт розміщення прапора границі є не зростаючою функцією залежно від збільшення інтервалу між кожним із фреймів бітового потоку (в якому прапор установлений) і границею програми, що ближче до зазначеного фрейму, де "коефіцієнт розміщення прапора границі" є середнім значенням відношення кількості фреймів (що вказують програму), які містять прапори границь програми до числа фреймів (що вказують програму), які не містять
 35 прапор границь програми, де середнє значення є ковзним середнім кількості (наприклад, відносно невеликого числа) послідовних фреймів кодованого бітового аудіопотоку. У класі варіантів здійснення винаходу коефіцієнт розміщення прапора границі логарифмічно зменшується в міру збільшення інтервалу (від кожного місця вставки прапора) до найближчої
 40 границі програми, а для кожного фрейму, що містить прапор, що містить у собі один із прапорів, розмір прапора в зазначеному фреймі, що містить прапор, дорівнює або більше, ніж розмір кожного прапора у фреймі, розташованому ближче до найближчої границі програми, ніж зазначений фрейм, що містить прапор (тобто, розмір прапора границі програми в кожному фреймі, що містить прапор, є неспадною функцією від збільшення інтервалу від зазначеного
 45 фрейму, що містить прапор, до найближчої границі програми).

Інший аспект даного винаходу являє собою блок обробки звукового сигналу (APU), сконфігурований з можливістю виконання будь-якого варіанта здійснення способу відповідно до винаходу. В іншому класі варіантів здійснення винахід являє собою APU, що включає буферний
 50 запам'ятовувальний пристрій (буфер), що зберігає (наприклад, незмінним способом) щонайменше один фрейм кодованого бітового аудіопотоку, сформований будь-яким варіантом здійснення способу відповідно до винаходу. Приклади APU включають, але не обмежуються ними: кодери (наприклад, транскодери), декодери, кодеки, системи попередньої обробки (препроцесори), системи пост-обробки (постпроцесори), системи обробки бітового аудіопотоку та комбінації таких елементів.

В іншому класі варіантів здійснення винахід являє собою блок обробки звукового сигналу (APU), виконаний з можливістю генерації кодованого бітового аудіопотоку, що включає сегменти аудіоданих і сегменти метаданих, де сегменти аудіоданих є індикаторами аудіоданих, а кожний з щонайменше деяких сегментів метаданих включає метадані стану обробки гучності (LPSM) і, можливо, також метадані границі програми. Як правило, щонайменше один такий сегмент
 60 метаданих у фреймі бітового потоку включає щонайменше один сегмент LPSM, з ознакою, чи

був виконаний перший тип обробки гучності з аудіоданими фрейму (тобто, аудіоданими щонайменше в одному сегменті аудіоданих фрейму), і щонайменше один інший сегмент LPSM, що вказує гучність щонайменше деяких аудіоданих фрейму (наприклад, гучність діалогу щонайменше деяких з аудіоданих у фреймі, що має ознаки діалогу). В одному з варіантів здійснення винаходу в цьому класі, APU являє собою кодер, виконаний з можливістю кодування вхідного аудіосигналу для формування кодованого аудіосигналу, а сегменти аудіоданих містять у собі кодований аудіосигнал. У типових варіантах здійснення винаходу в цьому класі кожний із сегментів метаданих має переважний формат, що буде описаний далі.

У деяких варіантах здійснення винаходу кожний із сегментів метаданих кодованого бітового потоку (бітового потоку AC-3 або бітового потоку E-AC-3 у деяких варіантах здійснення винаходу), що включає LPSM (наприклад, LPSM і метадані границі програми), входить у сегмент зайвих бітів поля ігнорованих даних фрейму бітового потоку (наприклад, сегмент зайвих бітів W типу, показаного на фіг. 4 або фіг. 7). В інших варіантах здійснення винаходу кожний із сегментів метаданих кодованого бітового потоку (бітового потоку AC-3 або бітового потоку E-AC-3 у деяких варіантах здійснення винаходу), що включає FPSM (наприклад, FPSM і метадані границі програми), входить як додаткова інформація бітового потоку в поле "addbsi" інформаційного сегмента бітового потоку ("BSI") фрейму бітового потоку або в поле допоміжних даних (наприклад, сегмент AUX типу, показаного на фіг. 4 або фіг. 7) наприкінці фрейму бітового потоку. Кожний сегмент метаданих, що включає FPSM, може мати формат, визначений у даному описі з посиланням на таблицю 1 і таблицю 2, наведені нижче (тобто, він містить у собі основні елементи, зазначені в таблиці 1, або варіанти, після чого ідентифікатор інформаційного наповнення (ідентифікуючий метадані, такі як LPSM) і значення розміру інформаційного наповнення, а потім інформаційне наповнення (LPSM дані, які мають формат, показаний у таблиці 2, або формат, показаний у варіанті таблиці 2, наведений у даному описі). У деяких варіантах здійснення винаходу фрейм може включати один або два сегменти метаданих, кожний з яких включає LPSM, а якщо фрейм включає два сегменти метаданих, один може бути присутнім в полі фрейму addbsi, а інший - в полі фрейму AUX.

У класі варіантів здійснення винаходу даний винахід являє собою спосіб, що включає в себе етапи кодування аудіоданих для формування AC-3 або E-AC-3 кодованого бітового аудіопотоку, у тому числі за рахунок включення в сегмент метаданих (щонайменше одного фрейму бітового потоку) LPSM і метаданих границі програми й, можливо, також і інших метаданих для аудіопрограми, до якої належить цей фрейм. У деяких варіантах здійснення винаходу кожний такий сегмент метаданих включений у поле фрейму addbsi або поле допоміжних даних фрейму. В інших варіантах здійснення винаходу кожний такий сегмент метаданих включений у сегмент зайвих бітів фрейму. У деяких варіантах здійснення винаходу кожний сегмент, що містить метадані LPSM і метадані границі програми, містить заголовок фрейму (і в деяких випадках також додаткові основні елементи), і після заголовка фрейму (або заголовка фрейму й інших основних елементів) сегмент інформаційного наповнення LPSM (або контейнера), що має наступний формат: заголовок, як правило, що включає щонайменше одне ідентифікаційне значення (наприклад, версію формату LPSM, довжину, період, число і асоціативні значення вкладеного потоку даних, як зазначено в таблиці 2, наведений в даному описі), і після заголовка - LPSM і метадані границі програми. Метадані границі програми можуть містити в собі число фреймів до границі програми та значення коду (наприклад, значення "offset_exist"), що вказує, чи містить кадр тільки число фреймів до границі програми або число фреймів до границі програми й значення зсуву, і (у деяких випадках) значення зсуву. LPSM може включати:

- щонайменше одне значення, що вказує діалог, яке вказує одне із двох - відповідні аудіодані мають ознаку діалогу або не мають ознаки діалогу (наприклад, які канали відповідних аудіоданих мають ознаку діалогу). Значення, що вказує діалог, може вказати, чи є присутнім діалог у будь-якій комбінації каналів або у всіх каналах відповідних аудіоданих;

- щонайменше одне значення дотримання нормативних вимог по гучності, що вказує, чи відповідають відповідні аудіодані зазначеному пакету нормативних вимог щодо гучності;

- щонайменше одне значення обробки гучності, що вказує щонайменше один тип обробки гучності, що був виконаний з відповідними аудіоданими; і

- щонайменше одне значення гучності, що вказує щонайменше одну характеристику гучності (наприклад, пікове або середнє значення гучності) відповідних аудіоданих.

В інших варіантах здійснення винаходу кодований бітовий потік являє собою бітовий потік, що не є бітовим потоком AC-3 або бітовим потоком E-AC-3, а кожний із сегментів метаданих, що включає LPSM (і в деяких випадках також метадані границі програми), входить у сегмент (або поле, або слот) бітового потоку, зарезервований для зберігання додаткових даних. Кожний сегмент метаданих, що включає LPSM, може мати формат аналогічний або ідентичний

зазначеному в даному описі з посиланням на таблицю 1 і таблицю 2, які наведені нижче, (тобто, він містить у собі основні елементи, аналогічні або ідентичні тим, які зазначені в таблиці 1, за якими йде ID (ідентифікатор) інформаційного наповнення (ідентифікаційні метадані як LPSM) і значення обсягу інформаційного наповнення, а потім інформаційне наповнення (LPSM дані, які мають формат, аналогічний або ідентичний формату, зазначеному в таблиці 2, або у варіанті таблиці 2, наведеній в даному описі).

У деяких варіантах здійснення винаходу кодований бітовий потік містить послідовність фреймів, кожний із фреймів включає інформаційний сегмент бітового потоку ("BSI"), що включає поле "addbsi" (яке іноді називають сегмент або слот) і поле або слот допоміжних даних (наприклад, кодований бітовий потік є бітовим потоком AC-3 або бітовим потоком E-AC-3), і включає сегменти аудіоданих (наприклад, сегменти фрейму ABO - AB5, показані на фіг. 4) і сегменти метаданих, де сегменти аудіоданих є ознакою аудіоданих, причому кожний з щонайменше деяких сегментів метаданих включає метадані стану обробки гучності (LPSM) і у деяких випадках також метадані границі програми. LPSM присутні в бітовому потоці в наступному форматі. Кожний із сегментів метаданих, що включає LPSM, включений у поле "addbsi" сегмента BSI фрейму бітового потоку або в поле допоміжних даних бітового потоку, або в сегмент зайвих бітів фрейму бітового потоку. Кожний сегмент метаданих, що включає LPSM, містить сегмент інформаційного наповнення (або контейнера) LPSM, що має наступний формат: заголовок (зазвичай включає щонайменше одне ідентифікуюче значення, наприклад версію формату LPSM, довжину, період, число й асоціативні значення вкладеного потоку даних, зазначені нижче в таблиці 2); і після заголовка, LPSM і в деяких випадках також метадані границі програми. Метадані границі програми можуть містити в собі число фреймів до границі програми та значення коду (наприклад значення "offsetexist"), що вказує, чи містить фрейм тільки число фреймів до границі програми або й число фреймів до границі програми, і значення зсуву), і (у деяких випадках) значення зсуву. LPSM може містити:

- щонайменше одне значення, що вказує діалог (наприклад, параметр "Каналу(ів) діалогу" з таблиці 2), що вказує, що відповідні аудіодані вказують або не вказують діалог (наприклад, які канали відповідних аудіоданих вказують діалог). Значення, що вказує діалог, може вказати, чи є присутнім діалог у будь-якій комбінації каналів або у всіх каналах відповідних аудіоданих;

- щонайменше одне значення дотримання нормативних вимог по гучності (наприклад, параметр "Тип регулювання гучності" з таблиці 2), що вказує, чи відповідають відповідні аудіодані зазначеному пакету нормативних вимог щодо гучності;

- щонайменше одне значення обробки гучності (наприклад, один або кілька параметрів "Прапор корекції стробованої гучності діалогу", "Тип корекції гучності" з таблиці 2), що вказує щонайменше один тип обробки гучності, що був виконаний з відповідними аудіоданими; і

- щонайменше одне значення гучності (наприклад, один або декілька з параметрів: "Відносна стробована гучність за рекомендацією MCE (Міжнародний союз електрозв'язку)", "Стробована гучність за рекомендацією MCE", "Короткострокова гучність (3-секундний часовий інтервал) за рекомендацією MCE (EBU 3341)". і "Дійсне пікове значення", наведені в таблиці 2), що вказує щонайменше одну характеристику гучності (наприклад, пікову або середню гучність) відповідних аудіоданих.

У будь-якому з варіантів здійснення винаходу, що припускає, використовує або формує щонайменше одне значення гучності, що вказує відповідні аудіодані, значення гучності може вказувати на щонайменше одну вимірювальну характеристику рівня гучності, використовувану для обробки рівня гучності та/або динамічний діапазон аудіоданих.

У деяких втіленнях винаходу кожний із сегментів метаданих у полі "addbsi", або в полі допоміжних даних, або сегмента зайвих бітів фрейму бітового потоку має наступний формат:

заголовок фрейму (зазвичай включає синхрослово, що ідентифікує початок сегмента метаданих, а потім ідентифікаційні значення, наприклад, версію основного елемента, довжину й період, число розширених елементів, і асоціативні значення вкладеного потоку даних, зазначені в таблиці 1 нижче); і

після заголовка фрейму щонайменше одне захисне значення (наприклад, HMAC дайджест і значення цифрового відбитка, де HMAC дайджест може бути 256-бітним HMAC дайджестом (при використанні алгоритму SHA-2), що обчислюється за аудіоданими, основним елементом і всіма розширеними елементами, із усього фрейму, як зазначено в таблиці 1), що підходить щонайменше для одного з: декодування, аутентифікації, або перевірки достовірності щонайменше одного з метаданих стану обробки рівня гучності або відповідних аудіоданих); і

також після заголовка фрейму, якщо сегмент метаданих включає LPSM, ідентифікатор ("ID") інформаційного наповнення LPSM і значення обсягу (розмір) інформаційного наповнення LPSM, що ідентифікують наступні метадані як інформаційне наповнення LPSM, і вказує розмір

інформаційного наповнення LPSM. Сегмент інформаційного наповнення LPSM (переважно, що має формат, зазначений вище) йде за ID інформаційного наповнення LPSM і числовим значенням розміру інформаційного наповнення LPSM.

У деяких варіантах здійснення типу, описаного в попередньому абзаці, кожний сегмент метаданих у полі допоміжних даних (або в полі "addbsi", або в сегменті зайвих бітів) фрейму має три рівні структури:

структура високого рівня, що містить прапор, що вказує чи включає поле допоміжних даних (або поле addbsi) метадані, щонайменше одне значення ID, що вказує, який тип(и) метаданих є присутнім, і, як правило, також значення, що вказує, скільки біт метаданих (наприклад, кожного типу) є присутнім (якщо метадані присутні). Один тип метаданих, які можуть бути присутнім в LPSM, інший тип метаданих, які можуть бути присутні у метаданих границі програми, та інший тип метаданих, які можуть бути присутні у метаданих медіадосліджень;

структура середнього рівня, що містить основний елемент для кожного ідентифікованого типу метаданих (наприклад, заголовок фрейму, захисні значення, ID інформаційного наповнення й числове значення розміру інформаційного наповнення, наприклад, зі згаданого вище типу, для кожного ідентифікованого типу метаданих); і

структура низького рівня, що включає кожне інформаційне наповнення для одного з основних елементів (наприклад, інформаційне наповнення LPSM, якщо воно ідентифікується основним елементом як присутнє, і/або інформаційне наповнення метаданих іншого типу, якщо воно ідентифікується основним елементом як присутнє).

Значення даних в такій тривірневій структурі можуть бути вкладеними. Наприклад, захисні значення для інформаційного наповнення LPSM і/або іншого інформаційного наповнення метаданих, визначеного основним елементом, можуть бути включені після кожного інформаційного наповнення, визначеного основним елементом (і, таким чином, після заголовка фрейму основного елемента). В одному прикладі, заголовок фрейму може ідентифікувати інформаційне наповнення LPSM та інше інформаційне наповнення метаданих, ID інформаційного наповнення та значення розміру інформаційного наповнення для першого інформаційного наповнення (наприклад, інформаційного наповнення LPSM) може йти за заголовком фрейму, саме перше інформаційне наповнення може йти за ID і значеннями розміру, значення розміру інформаційного наповнення й ID інформаційного наповнення для другого інформаційного наповнення можуть йти за першим інформаційним наповненням, саме друге інформаційне наповнення може йти за цими ID і значеннями розміру, а захисне значення для одного або обох інформаційних наповнень (або для значень основного елемента одного або обох інформаційних наповнень) може йти за останнім інформаційним наповненням.

У деяких варіантах здійснення винаходу основний елемент сегмента метаданих у полі допоміжних даних (або в полі "addbsi" або в сегменті зайвих бітів) фрейму містить заголовок фрейму (як правило, що включає ідентифікаційні значення, наприклад, версію основного елемента), і після заголовка фрейму: значення, що вказують, чи входять дані цифрового відбитка в метадані сегмента метаданих, значення, що вказують чи існують зовнішні дані (пов'язані з аудіоданими, що відповідають метаданим сегмента метаданих), ID корисного наповнення й значення розміру корисного наповнення для кожного типу метаданих (наприклад, LPSM, і/або метадані типу, відмінного від LPSM), ідентифікованого основним елементом, і захисні значення щонайменше одного типу метаданих ідентифікованого основним елементом. Інформаційне наповнення(я) метаданих сегмента метаданих йде за заголовком фрейму, і (у деяких випадках) вкрито в значення основного елемента.

В іншому переважному форматі кодований бітовий потік є бітовим потоком Dolby E, а кожний із сегментів метаданих, що включає LPSM (а в деяких випадках метадані границі програми) входить в оточення перших N семплів захисного частотного інтервалу Dolby E.

В іншому класі варіантів здійснення винахід являє собою APU (наприклад, декодер), підключений і виконаний з можливістю прийому кодованого бітового аудіопотоку, що містить сегменти аудіоданих і сегменти метаданих, де сегменти аудіоданих вказують аудіодані, а кожний з, щонайменше, деяких сегментів метаданих включає метадані стану обробки гучності (LPSM) і, у деяких випадках, також метадані границі програми, для добування LPSM з бітового потоку, щоб сформувані декодовані аудіодані у відповідь на аудіодані й виконати щонайменше одну операцію адаптивної обробки гучності аудіоданих з використанням LPSM. Деякі варіанти здійснення винаходу в цьому класі також включають постпроцесор, підключений до APU, причому постпроцесор підключений і настроєний для виконання щонайменше однієї адаптивної операції обробки гучності аудіоданих за допомогою LPSM.

В іншому класі варіантів здійснення винахід являє собою блок обробки звукового сигналу (APU), що включає буферний запам'ятовувальний пристрій (буфер) і підсистему обробки,

з'єднану з буфером, причому APU підключений для одержання кодованого бітового аудіопотоку, що містить сегменти аудіоданих і сегменти метаданих, де сегменти аудіоданих є ознакою аудіоданих, а кожний з, щонайменше, деяких сегментів метаданих містить у собі метадані стану обробки гучності (LPSM) і, факультативно, також метадані границі програми, буфер зберігає (наприклад, незмінним способом) щонайменше один фрейм кодованого бітового аудіопотоку, і підсистема обробки настроєна з можливістю добування LPSM з бітового потоку, і виконання щонайменше однієї адаптивної операції обробки гучності аудіоданих з використанням LPSM. У типових варіантах здійснення винаходу в цьому класі, APU є одним з кодера, декодера і постпроцесора.

У деяких втіленнях способу, що є предметом запропонованого винаходу, сформований бітовий аудіопотік являє собою або кодований бітовий потік AC-3, або бітовий потік E-AC-3, або бітовий потік Dolby E, що включають метадані стану обробки гучності, а також інші метадані (наприклад, параметр метаданих DIALNORM, параметри метаданих регулювання динамічного діапазону і інші параметри метаданих). У деяких інших варіантах здійснення зазначеного способу, сформований бітовий аудіопотік являє собою кодований бітовий потік іншого типу.

Аспекти даного винаходу містять у собі систему або пристрій, настроєний (наприклад, запрограмований) для виконання якого-небудь варіанта здійснення способу відповідно до винаходу, і машинопрочитуваний носій (наприклад, диск), що зберігає код (наприклад, незмінним способом) для втілення кожного з варіантів здійснення способу відповідно до винаходу або їхніх етапів. Наприклад, система відповідно до винаходу може являти собою або включати програмований процесор загального призначення, цифровий сигнальний процесор або мікропроцесор, запрограмований за допомогою програмних засобів або апаратно-програмних засобів і/або іншим способом настроєний для виконання будь-якої з множини операцій з даними, включаючи варіант здійснення способу відповідно до винаходу або його етапів. Такий процесор загального призначення може являти собою або включати комп'ютерну систему, що містить пристрої уведення, запам'ятовувальний пристрій і запрограмовану (і/або іншим способом настроєну) схему обробки для виконання варіанта здійснення способу (або його етапів) відповідно до винаходу у відповідь на затверджені дані.

На фіг. 1 представлена структурна схема варіанта здійснення системи, що може бути сконфігурована для виконання варіанта здійснення способу відповідно до винаходу.

На фіг. 2 представлена структурна схема кодера, що є варіантом здійснення блока обробки звукового сигналу відповідно до винаходу.

На фіг. 3 представлена структурна схема декодера, що є варіантом здійснення блока обробки звукового сигналу відповідно до винаходу й підключеного до нього постпроцесора, що є ще одним варіантом здійснення блока обробки звукового сигналу відповідно до винаходу.

На фіг. 4 представлена діаграма AC-3 фрейму, розділеного на сегменти.

На фіг. 5 представлена діаграма сегмента синхронізуючої інформації (SI) AC-3 фрейму, що включає сегменти, на які він розділений.

На фіг. 6 представлена діаграма інформаційного сегмента бітового потоку (BSI) AC-3 фрейму, що включає сегменти, на які він розділений.

На фіг. 7 представлена діаграма розділеного на сегменти E-AC-3 фрейму.

На фіг. 8 показана діаграма фреймів кодованого бітового аудіопотоку, що містить метадані границі програми у форматі, що відповідає варіанту здійснення даного винаходу.

На фіг. 9 показана діаграма інших фреймів кодованого бітового аудіопотоку на фіг. 9. Деякі із цих фреймів містять метадані границі програми у форматі, що відповідає варіанту здійснення даного винаходу.

На фіг. 10 представлена діаграма двох кодованих бітових аудіопотоків: бітового потоку (IEB), у якому границя програми (позначена як "границя") збігається з переходом між двома фреймами бітового потоку, та іншого бітового потоку (TB), у якому границя програми (позначена як "дійсна границя") зсунута на 512 семплів від переходу між двома фреймами бітового потоку.

На фіг. 11 представлений набір діаграм, що показують чотири кодованих бітових аудіопотоки. Бітовий потік у верхній частині фіг. 11 (позначений як "Сценарій 1") вказує на першу аудіопрограму (P1), що містить метадані границі програми, за якими йде друга аудіопрограма (P2), що також містить метадані границі програми; другий бітовий потік (позначений як "Сценарій 2") вказує на першу аудіопрограму (P1), що містить метадані границі програми, за якими йде друга аудіопрограма (P2), що не містить метаданих границі програми; третій бітовий потік (позначений як "Сценарій 3") вказує на обрізану першу аудіопрограму (P1), що містить метадані границі програми, і яка була змонтована із цілою другою аудіопрограмою (P2), що містить метадані границі програми; а також четвертий бітовий потік (позначений як "Сценарій 4"), що вказує на обрізану першу аудіопрограму (P1), що містить метадані границі

програми, і на обрізану другу аудіопрограму (P2), що містить метадані границі програми і яка була змонтована із частиною першої аудіопрограми.

Позначення й термінологія

У даному описі, включаючи формулу винаходу, вираз виконання операції "з" сигналом або даними (наприклад, фільтрація, масштабування, перетворення або застосування посилення для сигналу або даних) використовується в широкому сенсі для позначення виконання дії безпосередньо із сигналом або даними або з обробленим варіантом сигналу або даних (наприклад, з варіантом сигналу, що зазнав попередньої фільтрації або попередньої обробки до виконання дії з ним).

У даному описі, включаючи формулу винаходу, вираз "система" використовується в широкому сенсі для позначення пристрою, системи або підсистеми. Наприклад, підсистема, що реалізує декодер, може згадуватися як декодувальна система, а система, що включає таку підсистему (наприклад, система, що формує X вихідних сигналів у відповідь на множину вхідних сигналів, де підсистема формує M вхідних сигналів, а інші X-M вхідних сигналів одержують від зовнішнього джерела), також може згадуватися як декодувальна система.

У даному описі, включаючи формулу винаходу, термін "процесор" використовується в широкому сенсі для позначення системи або пристрою, що програмується або іншим чином налаштовується (наприклад, за допомогою програмного забезпечення або програмно-апаратних засобів) для виконання операцій з даними (наприклад, аудіо, або відео, або іншими даними зображень). Приклади процесорів включають програмовану користувачем вентиляну матрицю (або іншу інтегральну схему з конфігурацією, що переналаштовується, або чипсет), цифровий сигнальний процесор, запрограмований і/або іншим способом сконфігурований для виконання конвеєрної обробки аудіоданих або інших звукових даних, програмований процесор загального призначення або комп'ютер, а також програмовану мікропроцесорну велику інтегральну схему або чипсет.

У даному описі, включаючи формулу винаходу, вираз "аудіопроцесор" і "блок обробки звукового сигналу" використовуються взаємозамінним чином та для позначення системи в широкому сенсі, виконаної з можливістю обробки аудіоданих. Приклади блоків обробки звукового сигналу включають кодери (наприклад, транскодери), декодери, кодеки, системи попередньої обробки, системи постобробки та системи обробки бітового потоку (які іноді називають інструментами обробки бітового потоку), але не обмежуються ними.

У даному описі, включаючи формулу винаходу, вираз "метадані стану обробки" (наприклад, вираз "метадані стану обробки гучності") належить до окремих даних і даних, що відрізняються від відповідних аудіоданих (аудіоконтенту потоку аудіоданих, що також включає метадані стану обробки). Метадані стану обробки, пов'язані з аудіоданими, вказують стан обробки гучності відповідних аудіоданих (наприклад, те, який тип(и) обробки вже був виконаний з аудіоданими) і, як правило, також вказують щонайменше один параметр або характеристику аудіоданих. Зв'язок метаданих стану обробки з аудіоданими синхронізований за часом. Таким чином, дійсні (зовсім недавно отримані або оновлені) метадані стану обробки вказують, що відповідні аудіодані одночасно включають результати зазначеного типу(ів) обробки аудіоданих. У деяких випадках метадані стану обробки можуть включати послідовність подій обробки й/або деякі або всі параметри, які використовуються та/або виходять при зазначених видах обробки. Крім цього, метадані стану обробки можуть включати щонайменше одну ознаку або характеристику відповідних аудіоданих, які були обчислені або добуті з аудіоданих. Метадані стану обробки можуть також включати інші метадані, які не пов'язані з або не отримані при якій-небудь обробці відповідних аудіоданих. Наприклад, дані сторонніх виробників, дані супроводу, ідентифікатори, службова або стандартна інформація, дані приміток користувача, дані установок користувача та ін. можуть бути додані окремим блоком обробки звукового сигналу для передачі іншому блоку обробки звукового сигналу.

У даному описі, включаючи формулу винаходу, вираз "метадані стану обробки гучності" (або "LPSM") означає метадані стану обробки, що вказують стан обробки гучності відповідних аудіоданих (наприклад, тип(и) обробки гучності, виконаний з аудіоданими) і, як правило, також щонайменше один параметр або характеристику (наприклад, гучність) відповідних аудіоданих. Метадані стану обробки гучності можуть включати дані (наприклад, інші метадані), які не є (тобто, коли вони вважаються окремими) метаданими стану обробки гучності.

У даному описі, включаючи формулу винаходу, вираз "канал" (або "аудіоканал") позначає монофонічний звуковий сигнал.

У даному описі, включаючи формулу винаходу, вираз "аудіопрограма" означає набір з одного або декількох аудіоканалів і в деяких випадках також зв'язаних метаданих (наприклад,

метаданих, які описують бажане просторове звукове представлення, і/або LPSM, і/або метадані границі програми).

У даному описі, включаючи формулу винаходу, вираз "метадані границі програми" позначає метадані кодованого бітового аудіопотоку, де кодований бітовий аудіопотік є ознакою щонайменше однієї аудіопрограми (наприклад, двох або більше аудіопрограм). а метадані границі програми вказують місце розташування щонайменше однієї границі (початку та/або кінця) щонайменше однієї зазначеної аудіопрограми в бітовому потоці. Наприклад, метадані границі програми (кодованого бітового аудіопотоку, що вказує аудіопрограму) можуть включати метадані, що вказують місце розташування (наприклад, початок "N"-го фрейму бітового потоку або місце розташування "M"-го семпла в "TM"-ому фреймі бітового потоку) початку програми, а також додаткові метадані, що вказують місце розташування (наприклад, початок "J"-го фрейму бітового потоку або місце розташування "K"-го семпла в "J"-ому фреймі бітового потоку) кінця програми.

У даному описі, включаючи формулу винаходу, термін "з'єднувати" або "з'єднаний" використовується для позначення або прямого, або непрямого підключення. Таким чином, якщо перший пристрій з'єднаний із другим пристроєм, то підключення може бути прямим підключенням або непрямым підключенням через інші пристрої й підключення.

Відповідно до типових варіантів здійснення винаходу, інформаційне наповнення метаданих гучності програми називають метаданими стану обробки гучності ("LPSM"), і в деяких випадках також метадані границі програми убудовані в одне або декілька резервних полів (або слотів) сегментів метаданих бітового аудіопотоку, що також містить у собі аудіодані в інших сегментах (сегментах аудіоданих). Як правило, щонайменше один сегмент кожного фрейму бітового потоку включає LPSM, і щонайменше один сегмент фрейму включає відповідні аудіодані (тобто аудіодані, стан обробки гучності яких зазначено за допомогою LPSM). У деяких варіантах здійснення винаходу обсяг даних в LPSM може бути досить малим, і метадані передаються без впливу на швидкість передачі даних, відведеної для передачі аудіоданих.

Взаємодія метаданих стану обробки гучності в тракці обробки аудіоданих є особливо зручною, коли два або кілька блоків обробки звукового сигналу повинні працювати спільно один з одним у тракці обробки даних (або протягом життєвого циклу контенту). Без включення метаданих стану обробки гучності в бітовий аудіопотік можуть відбуватися серйозні проблеми медійної обробки, такі як зниження якості, рівня й просторового звучання, наприклад, коли два або декілька аудіокодеків використовуються послідовно й застосовується асиметричне авторегулювання гучності більш, ніж один раз протягом переміщення бітового потоку до сприймаючих медіа-пристроїв (або до точки відтворення аудіоконтенту бітового потоку).

На фіг. 1 представлена структурна схема зразкового тракту обробки аудіоданих (системи обробки аудіоданих), на якій один або декілька елементів системи можуть бути виконані відповідно до варіанта здійснення даного винаходу. Система включає наступні елементи, з'єднані, як показано на схемі: блок попередньої обробки, кодер, блок аналізу сигналів і корекції метаданих, транскодер, декодер і блок попередньої обробки. У варіантах показаної системи один або декілька елементів опущені або включені додаткові блоки обробки аудіоданих.

У деяких втіленнях блок попередньої обробки, наведений на фіг. 1, виконаний з можливістю приймати як вхідні дані ІКМ (імпульсно-кодова модуляція) семпли (часової області), що містять аудіоконтент, і виводити оброблені ІКМ семпли. Кодер може бути виконаний з можливістю прийому ІКМ семплів як вхідних даних й виведення кодованого (наприклад, стиснутого) бітового аудіопотоку, що вказує аудіоконтент. Дані бітового потоку, що вказують аудіоконтент, іноді згадуються в даному описі як "аудіодані". Якщо кодер виконаний відповідно до типового варіанта здійснення дійсного винаходу, вихідні дані бітового потоку кодера включають метадані стану обробки гучності (і зазвичай також інші метадані, у деяких випадках метадані границі програми), а також аудіодані.

Блок аналізу сигналів і корекції метаданих, наведений на фіг. 1, може приймати один або декілька кодованих бітових аудіопотоків як вхідні дані і визначати (наприклад, перевіряти правильність) чи є метадані стану обробки вірними в кожному кодованому бітовому аудіопотоці, виконуючи аналіз сигналу (наприклад, використовуючи метадані границі програми в кодованому бітовому аудіопотоці). Як правило, блок аналізу сигналів і корекції метаданих заміняє невірне (невірні) значення на вірне (вірні) значення, отримане шляхом аналізу сигналу, якщо виявляє, що включені метадані є недійсними. Таким чином, всі вихідні дані кодованого бітового аудіопотоку блока аналізу сигналів і корекції метаданих можуть включати скоректовані (або нескоректовані) метадані стану обробки, а також кодовані аудіодані.

Транскодер, наведений на фіг. 1, може приймати кодовані бітові потоки як вхідні сигнали і виводити у відповідь модифіковані (наприклад, кодовані іншим способом) бітові аудіопотоки

(наприклад, шляхом декодування вхідного потоку й перекодування декодованого потоку в інший формат кодування). Якщо транскодер виконаний відповідно до типового варіанта здійснення даного винаходу, вихідні дані бітового аудіопотоку транскодера містять метадані стану обробки гучності (і, як правило, також інші метадані), а також кодовані аудіодані. Метадані можуть бути включені в бітовий потік.

Декодер, наведений на фіг. 1, може приймати кодовані (наприклад, стиснуті) бітові аудіопотоки як вхідні дані і виводити (у відповідь) потоки декодованих ІКМ аудіосемплів. Якщо декодер виконаний відповідно до типового варіанта здійснення даного винаходу, вихідний сигнал декодера в номінальному режимі роботи являє собою або містить:

потік аудіосемплів і відповідний потік метаданих стану обробки гучності (і, як правило, також інших метаданих), добутий із вхідного кодованого бітового потоку; або

потік аудіосемплів і відповідний потік керуючих бітів, визначених за метаданими стану обробки гучності (і, як правило, також за іншими метаданими), добутий із вхідного кодованого бітового потоку; або

потік аудіосемплів без відповідного потоку метаданих стану обробки або керуючих бітів, визначених за метаданими стану обробки. В останньому випадку декодер може добувати метадані стану обробки гучності (і/або інші метадані) із вхідного кодованого бітового потоку й виконувати щонайменше одну операцію з добутими метаданими (наприклад, перевірку правильності), навіть якщо він не виводить добуті метадані або керуючі біти, визначені з них.

Блок постобробки, наведений на фіг. 1, виконаний відповідно до типового варіанта здійснення даного винаходу, блок постобробки виконаний з можливістю приймати потік декодованих ІКМ аудіосемплів, а також виконувати його постобробку (наприклад, авторегулювання гучності аудіоконтенту) за допомогою метаданих стану обробки гучності (і, як правило, також інших метаданих), отриманих із семплами, або керуючих бітів (визначених декодером за метаданими обробки стану гучності та, як правило, також за іншими метаданими), отриманих з семплами. Блок постобробки, як правило, також виконаний з можливістю відтворювати постоброблений аудіоконтент для програвання за допомогою однієї або декількох акустичних колонок.

Типові варіанти здійснення даного винаходу забезпечують поліпшений тракт обробки аудіоданих, у якому блоки обробки звукового сигналу (наприклад, кодери, декодери, транскодери, і блоки попередньої обробки й постобробки) погоджують свою відповідну обробку для застосування до аудіоданих відповідно до поточного стану медіаданих, що вказані метаданими стану обробки гучності, отриманих відповідно блоками обробки звукового сигналу.

Вхідні аудіодані будь-якого блока обробки звукового сигналу системи, наведеної на фіг. 1 (наприклад, кодера або транскодера, наведені на фіг. 1), можуть включати метадані стану обробки гучності (і в деяких випадках також інші метадані), а також аудіодані (наприклад, кодовані аудіодані). Ці метадані можуть бути включені у вхідні аудіодані іншим елементом системи, наведеної на фіг. 1 (або іншим джерелом, не показаним на фіг. 1), відповідно до варіанта здійснення даного винаходу. Блок обробки, що отримує вхідні аудіодані (з метаданими), може бути виконаний з можливістю здійснення щонайменше однієї операції з метаданими (наприклад, перевірки правильності) або у відповідь на метадані (наприклад, адаптивної обробки вхідних аудіоданих) і, як правило, також з можливістю включення метаданих у вихідні аудіодані, оброблений варіант метаданих або керуючі біти, визначені з метаданих.

Типовий варіант здійснення винаходу блока обробки звукового сигналу (або аудіопроцесор) виконаний з можливістю здійснення адаптивної обробки аудіоданих на основі стану аудіоданих, зазначеному метаданими стану обробки гучності, що відповідають аудіоданим. У деяких варіантах здійснення винаходу адаптивна обробка є (або включає) обробкою гучності (якщо метадані вказують, що обробка гучності або подібна їй обробка ще не була виконана з аудіоданими, але не є (і не включає) обробкою гучності (якщо метадані вказують, що така обробка гучності або подібна їй обробка вже була виконана з аудіоданими). У деяких варіантах здійснення винаходу адаптивна обробка є або включає перевірку правильності метаданих (наприклад, виконувати в підблоці перевірки правильності метаданих) для забезпечення виконання блоком обробки звукового сигналу іншої адаптивної обробки аудіоданих на основі стану аудіоданих, зазначеному метаданими стану обробки гучності. У деяких варіантах здійснення винаходу перевірка правильності визначає достовірність метаданих стану обробки гучності, пов'язаних з аудіоданими (наприклад, включеними в бітовий потік). Наприклад, якщо підтверджено достовірність метаданих, то результати раніше виконаної обробки аудіоданих можуть бути повторно використані, а виконання нової обробки аудіоданих такого ж типу можна уникнути. З іншого боку, якщо встановлено, що метадані були змінені (або з інших причин є

недостовірними), то передбачуваний раніше виконаний тип медійної обробки (як зазначено недостовірними метаданими) може бути повторений блоком обробки звукового сигналу та/або інша обробка метаданих та/або аудіоданих може бути виконана за допомогою блока обробки звукового сигналу. Блок обробки звукового сигналу може також бути виконаний з можливістю

повідомляти іншим блокам обробки звукового сигналу, розташованим далі в удосконаленому тракці медійної обробки, що метадані обробки стану гучності (наприклад, присутні в бітовому потоці медіаданих) є дійсними, якщо блок визначає, що метадані стану обробки дійсні (наприклад, на підставі збігу добутого значення й контрольного значення при криптографічній перевірці).

На фіг. 2 наведена блок-схема кодера (100), що є варіантом здійснення блока обробки звукового сигналу відповідно до винаходу. Будь-який з компонентів або елементів кодера 100 може бути втілений у вигляді одного або декількох процесів та/або однієї або декількох схем (наприклад, спеціалізованих інтегральних схем, ПКВМ (програмована користувачем вентильна матриця) або інших інтегральних схем), апаратних засобів, програмного забезпечення або комбінації апаратних засобів і програмного забезпечення. Кодер 100 містить буфер ПО фрейму, синтаксичний аналізатор 111, декодер 101, валідатор 102 стану аудіоданих, ланку 103 обробки гучності, ланку 104 вибору аудіопотоку, кодер 105, ланку 107 формувача швидкості передачі даних/пристрою форматування, ланку 106 формування метаданих. підсистему 108 вимірювання гучності діалогу та буфер 109 фрейму, з'єднані, як показано на схемі. Також, як правило, кодер 100 містить інші елементи обробки (не показані).

Кодер 100 (який є транскодером) виконаний з можливістю перетворення вхідного бітового аудіопотоку (який, наприклад, може бути бітовим потоком формату AC-3, або бітовим потоком формату E-AC-3, або бітовим потоком формату Dolby E) у кодований вихідний бітовий аудіопотік (який, наприклад, може бути іншим бітовим потоком формату AC-3, або бітовим потоком формату E-AC-3, або бітовим потоком формату Dolby E) у тому числі шляхом виконання адаптивної і автоматизованої обробки гучності з використанням метаданих стану обробки гучності, включених у вхідний бітовий потік. Наприклад, кодер 100 може бути виконаний з можливістю перетворення вхідного бітового потоку у форматі Dolby E (формат, як правило, використовується у виробничій і віщальній апаратурі, але не в побутових пристроях, які одержують аудіопрограми, передані їм) у кодований вихідний бітовий аудіопотік (що підходить для трансляції побутовими пристроями) у форматі AC-3 або E-AC-3.

Система, наведена на фіг. 2, також включає підсистему 150 передачі кодованих аудіоданих (яка зберігає та/або передає кодовані бітові потоки, що виходять із кодера 100) і декодер 152. Кодований бітовий аудіопотік, що виходить із кодера 100, може бути збережений підсистемою 150 (наприклад, на DVD або Blu-Ray диску) або переданий підсистемою 150 (яка може здійснити канал передачі або мережу), або може бути і збережений, і переданий підсистемою 150. Декодер 152 виконаний з можливістю декодувати кодований бітовий аудіопотік (сформований кодером 100), який він одержує за допомогою підсистеми 150, у тому числі шляхом добування метаданих стану обробки гучності (LPSM) з кожного фрейму бітового потоку (і факультативно також шляхом добування метаданих границі програми з бітового потоку) і формування декодованих аудіоданих. Як правило, декодер 152 виконаний з можливістю здійснення адаптивної обробки гучності декодованих аудіоданих з використанням LPSM (і в деяких випадках також метаданих границі програми) і/або передачі декодованих аудіоданих і LPSM у постпроцесор, виконаний з можливістю здійснення адаптивної обробки гучності декодованих аудіоданих з використанням LPSM (і, у деяких випадках, метаданих границі програми). Як правило, декодер 152 включає буфер, що зберігає (наприклад, незмінним способом) кодований бітовий аудіопотік, отриманий від підсистеми 150.

Різні втілення кодера 100 і декодера 152 виконані з можливістю виконання різних варіантів здійснення способу відповідно до винаходу.

Буфер 110 фрейму є буферним запам'ятовувальним пристроєм, підключеним для одержання кодованого вхідного бітового аудіопотоку. У процесі роботи буфер 110 зберігає (наприклад, незмінним способом) щонайменше один фрейм кодованого бітового аудіопотоку й послідовність фреймів кодованого бітового аудіопотоку передається з буфера 110 у синтаксичний аналізатор 111.

Синтаксичний аналізатор 111 з'єднаний і виконаний з можливістю добування метаданих стану обробки гучності (LPSM) і, факультативно, метаданих границі програми (і/або інших метаданих) з кожного фрейму кодованих вхідних аудіоданих, у які включені такі метадані, для передачі, щонайменше, LPSM (і в деяких випадках також метаданих границі програми й/або інших метаданих) у валідатор 102 стану аудіоданих, ланку 103 обробки гучності, ланку 106 і підсистему 108, добування аудіоданих з кодованих вхідних аудіоданих і передачі аудіоданих у

декодер 101. Декодер 101 кодера 100 виконаний з можливістю декодування аудіоданих для формування декодованих аудіоданих і передачі декодованих аудіоданих у ланку 103 обробки гучності, ланку 104 вибору аудіопотоку, підсистему 108, а також, як правило, валідатор 102 стану.

5 Валідатор 102 стану виконаний з можливістю перевірки дійсності й перевірки правильності LPSM (і в деяких випадках інших метаданих), переданих йому. У деяких варіантах здійснення винаходу LPSM є (або включений в) блок даних, включений у вхідний бітовий потік (наприклад, відповідно до варіанта здійснення даного винаходу). Блок може містити криптографічний хеш (алгоритм перевірки дійсності за допомогою криптографічних хеш-функцій, об'єднаних із шифруванням із закритим ключем, або "HMAC") для обробки LPSM (і в деяких випадках також і інших метаданих) і/або основних аудіоданих (наданих декодером 101 валідатору 102). Блок даних може мати цифровий підпис у цих варіантах здійснення винаходу, таким чином, наступний в тракту блок обробки звукового сигналу може відносно легко перевірити дійсність і підтвердити правильність метаданих стану обробки.

15 Наприклад, HMAC використовується для формування дайджесту, а захисне (захисні) значення, включене в бітовий потік відповідно до винаходу, може включати дайджест. Дайджест може бути отриманий для AC-3 фрейму в такий спосіб:

1. Після того як AC-3 дані й LPSM кодовані, байти даних фрейму (зв'язані послідовно frame_data #1 і frame_data #2) і байти даних LPSM використовуються в якості вхідних даних для хеш-функції HMAC. Інші дані, які можуть бути присутні усередині поля допоміжних даних, не будуть прийняті до уваги при здійсненні розрахунку дайджесту. Такі інші дані можуть бути байтами, що не належать ні до даних AC-3, ні до даних LPSM. Біти захисту, включені в LPSM, не можуть бути враховані при здійсненні розрахунку HMAC дайджесту.

2. Після здійснення розрахунку дайджесту його записують у бітовий потік у зарезервоване для бітів захисту поле.

25 3. Останнім кроком формування повного AC-3 фрейму є розрахунок контрольної суми CRC. Вона записується в самому кінці фрейму, і враховуються всі дані, що належать цьому фрейму, включаючи біти LPSM.

30 Інші криптографічні методи, включаючи який-небудь один або декілька не HMAC криптографічних методів, але не обмежуючись ними, можуть бути використані для перевірки правильності LPSM (наприклад, у валідаторі 102) з метою забезпечення безпечної передачі й прийому LPSM та/або основних аудіоданих. Наприклад, перевірка правильності (з використанням такого криптографічного методу) може бути виконана в кожному блоці обробки звукового сигналу, що приймає варіант здійснення бітового аудіопотоку відповідно до винаходу, щоб визначити, чи включені метадані стани обробки гучності й відповідні аудіодані в бітовий потік, що пройшов (і/або виник у результаті) спеціальну обробку гучності (як зазначено в метаданих) і не був модифікований після виконання такої спеціальної обробки гучності.

35 Валідатор 102 стану передає керуючі дані ланці 104 вибору аудіопотоку, формувачу 106 метаданих і підсистемі 108 вимірювання гучності діалогу, щоб указати результати операції перевірки правильності. Під впливом керуючих даних ланка 104 може вибрати (і передати на кодер 105) або адаптивно оброблені вихідні дані ланки 103 обробки гучності (наприклад, коли LPSM указують, що вихідні дані декодера 101 не проходили спеціальний тип обробки гучності, а керуючі біти сигналу валідатора 102 указують, що LPSM дійсні); або вихідні аудіодані декодера 101 (наприклад, коли LPSM указують, що вихідні аудіодані декодера 101 уже пройшли певний тип обробки гучності, виконаний ланкою 103, а керуючі біти сигналу валідатора 102 показують, що LPSM дійсні).

40 Ланка 103 кодера 100 виконана з можливістю здійснення адаптивної обробки гучності вихідних декодованих аудіоданих декодера 101 на основі однієї або декількох характеристик аудіоданих, вказаних за допомогою LPSM, добутих декодером 101. Ланка 103 може бути керуючим процесором з адаптивною областю перетворення гучності й динамічного діапазону в режимі реального часу. Ланка 103 може приймати вхідні дані користувача (наприклад, цільову гучність користувача/значення динамічного діапазону або значення dialnorm) або інші вхідні метадані (наприклад, один або кілька типів даних сторонніх виробників, дані спостереження, ідентифікатори, службову або стандартну інформацію, дані приміток користувача, дані установок користувача та ін.) і/або інші вхідні дані (наприклад, дані цифрового відбитка), і використати такі вхідні дані для обробки вихідних декодованих аудіоданих декодера 101. Ланка 103 може виконувати адаптивну обробку гучності декодованих аудіоданих (переданих від декодера 101), що вказують одну аудіопрограму (як зазначено метаданими границі програми, добутими синтаксичним аналізатором 111), а може скинути обробку гучності в результаті прийому декодованих аудіоданих (переданих від декодера 101), що вказують іншу

аудіопрограму, як зазначено метаданими границі програми, добути синтаксичним аналізатором 111.

Підсистема 108 вимірювання гучності діалогу може функціонувати з метою визначення гучності сегментів декодованих аудіоданих (від декодера 101), які вказують діалог (або іншу мову), наприклад, за допомогою LPSM (і/або інших метаданих), добутих за допомогою декодера 101, коли керуючі біти валідатора 102 показують, що LPSM є недійсними. Функціонування підсистеми 108 вимірювання гучності діалогу може бути зупинено, коли LPSM вказують раніше визначену гучність сегментів діалогу (або іншої мови) декодованих аудіоданих (від декодера 101), якщо керуючі біти валідатора 102 показують, що LPSM є дійсними. Підсистема 108 може виконувати вимірювання гучності декодованих аудіоданих, що вказують одну аудіопрограму (що вказується метаданими границі програми, добути синтаксичним аналізатором 111), і може скидати вимірювання у відповідь на прийом декодованих аудіоданих, що вказують іншу аудіопрограму, що вказується метаданими границі цієї програми.

Для зручного й легкого вимірювання рівня діалогу в аудіоконтенті існують корисні інструменти (наприклад, вимірювач гучності Dolby LM100). Деякі варіанти здійснення APU, відповідно до винаходу (наприклад, ланка 108 кодера 100), виконані з можливістю включення (або виконання функцій) такого інструмента для вимірювання середньої гучності діалогу аудіоконтенту бітового аудіопотоку (наприклад, декодованого бітового потоку формату AC-3, переданого ланці 108 від декодера 101 кодера 100).

Якщо ланка 108 виконана з можливістю вимірювання дійсного середнього значення гучності діалогу аудіоданих, то вимірювання може включати етап виділення сегментів аудіоконтенту, переважно таких, що містять мову. Потім аудіосегменти, які в основному є мовою, обробляються відповідно до алгоритму вимірювання гучності. Для декодованих аудіоданих бітового потоку формату AC-3 таким алгоритмом може бути стандартне вимірювання гучності, зваженої за K (відповідно до міжнародного стандарту ITU-R BS.1770). Як альтернатива можуть бути використані інші показники гучності (наприклад, ті, які засновані на психоакустичних моделях гучності).

Виділення мовних сегментів не є істотним для вимірювання середньої гучності діалогу аудіоданих. Проте, воно поліпшує точність вимірювання та зазвичай забезпечує більшою мірою задовольняючим вимогам результати з погляду слухача. Так як не весь аудіоконтент містить діалог (мову), показник гучності всього аудіоконтенту може забезпечити досить точне наближення рівня діалогу звукозапису, як яби була присутня тільки мова.

Формувач метаданих 106 формує (і/або передає ланці 107) метадані, які будуть включені ланкою 107 у кодований бітовий потік і виведені кодером 100. Формувач 106 метаданих може передавати ланці 107 LPSM (і в деяких випадках також метадані границі програми й/або інші метадані), добути кодером 101 та/або синтаксичним аналізатором 111 (наприклад, коли керуючі біти валідатора 102 показують, що LPSM та/або інші метадані дійсні), або формувати нові LPSM (і в деяких випадках також метадані границі програми та/або інші метадані) і передавати нові метадані ланці 107 (наприклад, коли керуючі біти валідатора 102 показують, що LPSM і/або інші метадані, добути декодером 101, є недійсними, або він може передавати ланці 107 комбінацію метаданих, добутих декодером 101 і/або синтаксичним аналізатором 111, і знову сформованих метаданих. Формувач 106 метаданих може включати дані гучності, сформовані підсистемою 108, а також щонайменше одне значення, що вказують тип обробки гучності, виконаною підсистемою 108, в LPSM і передавати ланці 107 для включення в кодований бітовий потік, що виводиться кодером 100.

Формувач 106 метаданих може формувати захисні біти (які можуть складатися з або містити код перевірки дійсності за допомогою криптографічних хеш-функцій, об'єднаних із шифруванням із закритим ключем, або "HMAC"), застосовні щонайменше для однієї з наступних операцій: декодування, перевірки дійсності або перевірки достовірності LPSM (і в деяких випадках також інших метаданих) із включенням їх у кодований бітовий потік і/або включення основних аудіоданих у кодований бітовий потік. Формувач 106 метаданих може надати такі захисні біти ланці 107 для включення в кодований бітовий потік.

У номінальному режимі роботи підсистема 108 вимірювання гучності діалогу обробляє аудіодані, що виводяться з декодера 101, для формування в результаті значень гучності (наприклад, значень стробованої та нестробованої гучності діалогу) і значень динамічного діапазону. Завдяки цим значенням формувач 106 метаданих може формувати метадані стану обробки гучності (LPSM) для включення (за допомогою формувача швидкості передачі даних/пристрою форматування 107) у кодований бітовий потік, що виводиться кодером 100.

Крім цього, у деяких випадках або як альтернатива підсистеми 106 і/або 108 кодера 100 можуть виконувати додатковий аналіз аудіоданих для формування метаданих, що вказують

щонайменше одну характеристику аудіоданих для включення в кодований бітовий потік, що виводиться ланкою 107.

5 Кодер 105 кодує (наприклад, шляхом виконання стиснення) вихідні аудіодані ланки 104 вибору й передає кодовані аудіодані ланці 107 для включення в кодований бітовий потік, що виводиться ланкою 107.

Ланка 107 виконує мультиплексування кодованих аудіоданих з кодера 105 і метаданих (включаючи LPSM) з формувача 106 для формування кодованого бітового потоку, що виводиться ланкою 107, переважно, щоб кодований бітовий потік мав формат, зазначений у переважному варіанті здійснення даного винаходу.

10 Буфер 109 фрейму являє собою буферний запам'ятовувальний пристрій, що зберігає (наприклад, незмінним способом) щонайменше один фрейм кодованого бітового аудіопотоку, виведеного ланкою 107, а послідовність фреймів кодованого бітового аудіопотоку потім передається з буфера 109, що є виходом кодера 100, у систему 150 передачі.

15 LPSM, сформовані формувачем 106 метаданих і включені в кодований бітовий потік ланкою 107, показують стан обробки гучності відповідних аудіоданих (наприклад, який тип(и) обробки гучності був виконаний з аудіоданими) і гучність (наприклад, виміряну гучність діалогу, стробовану та/або нестробовану гучність і/або динамічний діапазон) відповідних аудіоданих.

20 У даному описі "стробування" гучності та/або вимірювання рівня сигналу, виконане з аудіоданими, відноситься до спеціального рівня або порога гучності, причому обчислене (обчислені) значення, що перевищує граничне значення, включають у кінцеве вимірювання (наприклад, зневажають значеннями короткочасної гучності нижче -60 dBFS у кінцевих виміряних значеннях). Стробування за абсолютною величиною приведе до фіксованого рівня або гучності, враховуючи це, стробування за відносним значенням приводить до значення, що залежить від поточного "нестробованого" виміряного значення.

25 У деяких втіленнях кодера 100 кодований бітовий потік, що накопичується в запам'ятовувальному пристрої 109 (і виводиться в систему 150 передачі), є бітовим потоком формату AC-3 або бітовим потоком формату E-AC-3 і включає сегменти аудіоданих (наприклад, сегменти фрейму AB0-AB5, показані на фіг. 4) і сегменти метаданих, причому сегменти аудіоданих відображають аудіодані, а кожний із щонайменше деяких сегментів метаданих включає метадані стану обробки гучності (LPSM). Ланка 107 вставляє LPSM (і в деяких випадках також метадані границі програми) у бітовий потік у наступному форматі. Кожний із сегментів метаданих, що включає LPSM (і в деяких випадках також метадані границі програми), входить у сегмент зайвих бітів бітового потоку (наприклад, сегмент "W" зайвих бітів, як показано на фіг. 4 або фіг. 7), або в поле "addbsi" інформаційного сегмента бітового потоку ("BSI") фрейму бітового потоку, або в поле допоміжних даних (наприклад, сегмент AUX, показаний на фіг. 4 або фіг. 7) наприкінці фрейму бітового потоку. Фрейм бітового потоку може включати один або два сегменти метаданих, кожний з яких включає LPSM. і якщо фрейм включає два сегменти метаданих, то один може бути присутнім у полі фрейму addbsi, а інший - в полі фрейму AUX. У деяких варіантах здійснення винаходу кожний сегмент метаданих, що містить LPSM, включає

40 сегмент інформаційного наповнення (або контейнера) LPSM, що має наступний формат:
заголовок (як правило, що включає синхрослово, що ідентифікує початок інформаційного наповнення LPSM, за яким йде щонайменше одне ідентифікуюче значення, наприклад, версія формату LPSM, довжина, період, число, асоціативні значення вкладеного потоку даних, зазначені нижче в таблиці 2); і після заголовка щонайменше одне значення, що вказує діалог (наприклад, параметр "Канал(и) діалогу", наведений у таблиці 2), що вказує, чи мають відповідні аудіодані ознаку діалогу або не мають ознаки діалогу (наприклад, які канали відповідних аудіоданих мають ознаку діалогу);

50 щонайменше одне значення нормативних вимог щодо гучності (наприклад, параметр "Тип регулювання гучності", наведений у таблиці 2), що вказує, чи відповідають відповідні аудіодані зазначеному пакету нормативних вимог щодо гучності;

щонайменше одне значення обробки гучності (наприклад, один або кілька параметрів: "Прапор корекції стробованої гучності діалогу", "Тип корекції гучності", наведені в таблиці 2), що вказує щонайменше один тип обробки гучності, що був виконаний з відповідними аудіоданими; і

55 щонайменше одне значення гучності (наприклад, один або кілька параметрів: "Відносна стробована гучність ITU", "Стробована гучність мови ITU", "Короткочасна гучність (3-секундний часовий інтервал) ITU (EBU 3341)" і "Дійсне пікове значення", наведені в таблиці 2), що вказує щонайменше одну характеристику гучності відповідних аудіоданих (наприклад, пікову або середню гучність).

60 У деяких варіантах здійснення винаходу кожний сегмент метаданих, що містить LPSM і метадані границі програми, містить заголовок фрейму (і в деяких випадках також додаткові

основні елементи), а після заголовка фрейму (або заголовка фрейму та інших основних елементів) сегмент інформаційного наповнення (або контейнера) LPSM, що має наступний формат:

заголовок, як правило, що включає щонайменше одне ідентифікуюче значення (наприклад, версію формату LPSM, довжину, період, число, асоціативні значення вкладеного потоку даних, зазначені в таблиці 2. наведеної в даному описі), і

після заголовка - LPSM і метадані границі програми. Метадані границі програми можуть включати число фреймів до границі програми, значення коду (наприклад, значення "ofFsetexist"), що вказує, чи містить фрейм тільки число фреймів до границі програми або й число фреймів до границі програми, і значення зсуву) і (у деяких випадках) значення зсуву.

У деяких втіленнях кожний сегмент метаданих, внесений ланкою 107 у сегмент зайвих бітів, або поле "addbsi", або поле допоміжних даних фрейму бітового потоку, має наступний формат:

заголовок фрейму (як правило, що включає синхрослово, що вказує початок сегмента метаданих, за яким йде ідентифікаційне значення, наприклад, версія основного елемента, довжина й період, число розширених елементів й асоціативні значення вкладеного потоку даних, зазначені нижче в таблиці 1); і

після заголовка фрейму щонайменше одне захисне значення (наприклад, HMAC дайджест і значення цифрового відбитка аудіоданих, наведені в таблиці 1), що підходить щонайменше для однієї з операцій: декодування, перевірки дійсності або перевірки правильності щонайменше одного з: метаданих стану обробки гучності або відповідних аудіоданих); і

також після заголовка фрейму, якщо сегмент метаданих включає LPSM, ідентифікатор інформаційного наповнення LPSM і значення розміру інформаційного наповнення LPSM, які ідентифікують наступні за ними метадані як інформаційне наповнення LPSM і вказують розмір інформаційного наповнення LPSM.

Сегмент інформаційного наповнення (або контейнера) LPSM (переважно, що має формат, зазначений вище) йде за ідентифікатором інформаційного наповнення LPSM і значеннями розміру інформаційного наповнення LPSM.

У деяких варіантах здійснення винаходу кожний сегмент метаданих у полі допоміжних даних (або в полі "addbsi") фрейму має три рівні структури:

структура високого рівня, що включає прапор, що вказує, чи включає поле допоміжних даних (або поле addbsi) метадані, щонайменше одне ідентифікуюче значення, що вказує який тип(и) метаданих є присутніми, і, як правило, також значення, що вказує, скільки біт метаданих (наприклад, кожного типу) присутні (якщо метадані присутні). Одним типом метаданих, які можуть бути присутніми, є LSPM, іншим типом метаданих, які можуть бути присутніми, є метадані границі програми, ще одним типом метаданих, які можуть бути присутніми, є метадані медіадосліджень (наприклад, метадані Nielsen Media Research);

структура середнього рівня, що включає основний елемент для кожного ідентифікованого типу метаданих (наприклад, заголовок фрейму, захисні значення, ідентифікатор інформаційного наповнення LPSM і значення розміру інформаційного наповнення LPSM, як згадувалося вище, для кожного ідентифікованого типу метаданих); і

структура низького рівня, що включає будь-яке інформаційне наповнення для одного основного елемента (наприклад, інформаційне наповнення LPSM, якщо основним елементом визначено його присутність і/або інформаційне наповнення метаданих іншого типу, якщо основним елементом визначена його присутність).

Значення даних у такій трирівневій структурі можуть бути вкладеними. Наприклад, захисне (захисні) значення для інформаційного наповнення LPSM і/або іншого інформаційного наповнення метаданих, ідентифікованого основним елементом, може бути включене після кожного інформаційного наповнення, ідентифікованого основним елементом (і, таким чином, після заголовка фрейму основного елемента). В одному прикладі заголовок фрейму може ідентифікувати інформаційне наповнення LPSM і інше інформаційне наповнення метаданих, ідентифікатор інформаційного наповнення та значення розміру інформаційного наповнення для першого інформаційного наповнення (наприклад, інформаційного наповнення LPSM) може йти за заголовком фрейму, саме перше інформаційне наповнення може йти за ідентифікатором і значенням розміру, ідентифікатор інформаційного наповнення й значення розміру інформаційного наповнення для другого інформаційного наповнення може йти за першим інформаційним наповненням, саме друге інформаційне наповнення може йти за цими ідентифікатором і значенням розміру, а захисні біти для обох інформаційних наповнень (або для значень основного елемента й обох інформаційних наповнень) можуть йти за останнім інформаційним наповненням.

У деяких варіантах здійснення винаходу, якщо декодер 101 одержує бітовий аудіопотік, сформований відповідно до варіанта здійснення винаходу із криптографічним хешем, то декодер виконується з можливістю синтаксичного аналізу й добування криптографічного хеша із блока даних, визначеного бітовим потоком, зазначений блок містить метадані стану обробки гучності (LPSM) і в деяких випадках також метадані границі програми. Валідатор 102 може використовувати криптографічний хеш для перевірки отриманого бітового потоку та/або пов'язаних з ними метаданих. Наприклад, якщо валідатор 102 встановлює дійсність LPSM на основі збігу еталонного криптографічного хеша та криптографічного хеша, добутого із блока даних, то він може відключити обробку процесором 103 відповідних аудіоданих та ініціювати передачу (без змін) аудіоданих ланкою 104 вибору. Крім цього в деяких випадках або як альтернатива замість методу, заснованого на криптографічному хеші, можуть бути використані інші типи криптографічних методів.

Кодер 100, наведений на фіг. 2, може визначити (завдяки LPSM і в деяких випадках також метаданим границі програми, добутих за допомогою декодера 101), що блок постобробки/попередньої обробки виконав тип обробки гучності аудіоданих, які будуть закодовані (елементами 105, 106, і 107) і, отже, можна створювати (у формувачі 106) метадані стану обробки гучності, які включають спеціальні параметри, використовувані в і/або отримані від проведеної раніше обробки гучності. У деяких втіленнях кодер 100 може створити (і включати у вихідний кодований бітовий потік) метадані стану обробки, що вказують послідовність подій обробки аудіоконтенту, поки кодер розпізнає типи обробки, які були проведені з аудіоконтентом.

На фіг. 3 наведена структурна схема декодера (200), який є варіантом здійснення блока обробки звукового сигналу відповідно до винаходу, і постпроцесора (300), з'єднаного з ним. Постпроцесор (300) також є варіантом здійснення блока обробки звукового сигналу відповідно до винаходу. Будь-який з компонентів або елементів декодера 200 і постпроцесора 300 може бути втілений як один або кілька процесів і/або одна або кілька схем (наприклад, спеціалізовані інтегральні схеми, програмовані користувачем вентильні матриці або інші інтегральні схеми) в апаратних засобах, програмному забезпеченні або комбінації апаратних засобів і програмного забезпечення. Декодер 200 містить буфер 201 фрейму, синтаксичний аналізатор 205, аудіодекодер 202, ланку 203 перевірки стану аудіоданих (валідатор) і ланку 204 формування керуючих бітів, з'єднані, як показано на схемі. Також зазвичай декодер 200 включає інші елементи обробки (не показані).

Буфер 201 фрейму (буферний запам'ятовувальний пристрій) зберігає (наприклад, незмінним способом) щонайменше один фрейм кодованого бітового аудіопотоку, отриманого декодером 200. Послідовність фреймів кодованого бітового аудіопотоку передається з буфера 201 у синтаксичний аналізатор 205.

Синтаксичний аналізатор 205 з'єднаний і виконаний з можливістю добування метаданих стану обробки гучності (LPSM), у деяких випадках також метаданих границі програми та інших метаданих з кожного фрейму кодованого вхідного аудіосигналу, з можливістю передачі щонайменше LPSM (і метаданих границі програми, якщо вони добуті) валідатору 203 стану аудіоданих і ланці 204, з можливістю передачі LPSM (і в деяких випадках також метаданих границі програми) як вихідних даних (наприклад, постпроцесору 300), з можливістю добування аудіоданих з кодованого вхідного аудіосигналу й передачі добутих аудіоданих у декодер 202.

Кодований бітовий аудіопотік, що входить у декодер 200, може бути або бітовим потоком формату AC-3, або бітовим потоком формату E-AC-3, або бітовим потоком формату Dolby E.

Система, наведена на фіг. 3 також включає постпроцесор 300. Постпроцесор 300 містить буфер 301 фрейму та інші елементи обробки (не показані), що включають щонайменше один елемент обробки, з'єднаний з буфером 301. Буфер 301 фрейму зберігає (наприклад, незмінним способом) щонайменше один фрейм декодованого бітового аудіопотоку, отриманого постпроцесором 300 від декодера 200. Елементи обробки постпроцесора 300 з'єднані й виконані з можливістю прийому й адаптивної обробки послідовності фреймів вихідного декодованого бітового аудіопотоку буфера 301 за рахунок використання метаданих (у тому числі значень LPSM), що виводяться з декодера 202, і/або керуючих бітів, що виводяться з ланки 204 декодера 200. Як правило, постпроцесор 300 виконаний з можливістю здійснення адаптивної обробки гучності аудіоданих, що декодуються, за рахунок використання значень LPSM і в деяких випадках також метаданих границі програми (наприклад, на основі стану обробки гучності й/або однієї або декількох характеристик аудіоданих, зазначених LPSM для аудіоданих, що вказують одну аудіопрограму).

Різні втілення декодера 200 і постпроцесора 300 виконані з можливістю виконання різних варіантів здійснення способу відповідно до винаходу.

Аудіодекодер 202 декодера 200 виконаний з можливістю декодування аудіоданих, добутих за допомогою синтаксичного аналізатора 205 для формування декодованих аудіоданих. і передачі декодованих аудіоданих як вихідних даних (наприклад, постпроцесора 300).

Валідатор 203 стану виконаний з можливістю перевірки дійсності й перевірки правильності LPSM (і в деяких випадках інших метаданих), переданих йому. У деяких варіантах здійснення винаходу LPSM є (або включені в) блоком даних, що був включений у вхідний бітовий потік (наприклад, відповідно до варіанта здійснення даного винаходу). Блок може містити криптографічний хеш (код перевірки дійсності за допомогою криптографічних хеш-функцій, об'єднаних із шифруванням із закритим ключем, або "HMAC") для обробки LPSM (і в деяких випадках також і інших метаданих) і/або основних аудіоданих (наданих синтаксичним аналізатором 205 і/або декодером 202 валідатору 203). У цих варіантах здійснення винаходу блок даних може мати цифровий підпис, таким чином, наступний в тракці блок обробки звукового сигналу може відносно легко визначити достовірність і підтвердити правильність метаданих стану обробки.

Інші криптографічні методи, включаючи, але не обмежуючись яким-небудь одним або декількома криптографічними методами без HMAC, можуть бути використані для перевірки правильності LPSM (наприклад, у валідаторі 203) при забезпеченні безпечної передачі й прийому LPSM і/або основних аудіоданих. Наприклад, перевірка правильності (з використанням такого криптографічного методу) може бути виконана в кожному блоці обробки звукового сигналу, що приймає варіант здійснення бітового аудіопотоку відповідно до винаходу для визначення, чи проходили метадані стану обробки гучності й відповідні аудіодані, включені в бітовий потік, (і/або виникли внаслідок) спеціальну обробку гучності (як зазначено метаданими) і чи не були модифіковані після виконання такої спеціальної обробки гучності.

Валідатор 203 стану передає дані керування формувачу 204 керуючих бітів і/або передає дані керування як вихідні (наприклад, у постпроцесор 300) з метою вказівки результатів операції перевірки правильності. У відповідь на дані керування (і в деяких випадках також інші метадані, добути із вхідного бітового потоку) ланка 204 може формувати (і передавати в постпроцесор 300) або:

керуючі біти, що вказують, що декодовані аудіодані, виведені з декодера 202, пройшли спеціальний тип обробки гучності (коли LPSM показують, що аудіодані, виведені з декодера 202, пройшли спеціальний тип обробки гучності, а керуючі біти валідатора 203 указують, що LPSM дійсні); або

керуючі біти, що вказують, що декодовані аудіодані з декодера 202 повинні пройти спеціальний тип обробки гучності (наприклад, коли LPSM указують, що аудіодані, що виводяться з декодера 202, не проходили спеціального типу обробки гучності, або коли LPSM указують, що аудіодані, що виводяться з декодера 202, пройшли спеціальний тип обробки гучності, але керуючі біти валідатора 203 показують, що LPSM недійсні).

Як альтернатива декодер 200 передає метадані, добути декодером 202, із вхідного бітового потоку й LPSM (і в деяких випадках також метадані границі програми), добути синтаксичним аналізатором 205 із вхідного бітового потоку, постпроцесору 300, а постпроцесор 300 виконує обробку гучності декодованих аудіоданих за рахунок використання LPSM (і в деяких випадках також метаданих границі програми) або виконує перевірку правильності LPSM, а потім виконує обробку гучності декодованих аудіоданих за рахунок використання LPSM (і в деяких випадках також метаданих границі програми), якщо перевірка правильності показує, що LPSM є дійсними.

У деяких варіантах здійснення винаходу, якщо декодер 200 отримує бітовий аудіопотік, сформований відповідно до одного з варіантів здійснення винаходу із криптографічним хешем, то декодер виконується з можливістю аналізувати й добувати криптографічний хеш із блоку даних, визначених з бітового потоку, причому зазначений блок містить метадані стану обробки гучності (LPSM).

Валідатор 203 може використовувати криптографічний хеш для перевірки правильності прийнятого бітового потоку та/або пов'язаних з ними метаданих. Наприклад, якщо валідатор 203 виявляє на підставі збігу еталонного криптографічного хеша й криптографічного хеша, добутого із блока даних, що LPSM можуть вважатися дійсними, те це може сигналізувати наступному у тракці блока обробки звукового сигналу (наприклад, постпроцесора 300, що може бути включати блок авторегулювання гучності) передавати (не змінити) аудіодані бітового потоку. Крім цього, у деяких випадках або як альтернатива інші типи криптографічних методів можуть бути використані замість методу, заснованого на криптографічному хеші.

У деяких втіленнях декодера 200 кодований бітовий потік, отриманий (і накопичений у запам'ятовувальному пристрої 201), є бітовим потоком формату AC-3 або бітовим потоком формату E-AC-3 і включає сегменти аудіоданих (наприклад, сегменти AB0-AB5 фрейму,

показані на фіг. 4) і сегменти метаданих, причому сегменти аудіоданих є ознакою аудіоданих, а кожний із щонайменше деяких сегментів метаданих включає метадані стану обробки гучності (LPSM) і в деяких випадках також метадані границі програми. Ланка 202 декодера (і/або синтаксичного аналізатора 205) виконана з можливістю добування з бітового потоку LPSM (і в деяких випадках також метаданих границі програми), що мають наступний формат. Кожний із сегментів метаданих, що включає LPSM (і в деяких випадках також метадані границі програми), входить у сегмент зайвих бітів фрейму бітового потоку, або в поле "addbsi" інформаційного сегмента бітового потоку ("BSI") фрейму бітового потоку, або в поле допоміжних даних (наприклад, у сегмент AUX, показаний на фіг. 4) наприкінці фрейму бітового потоку. Фрейм бітового потоку може включати один або два сегменти метаданих, кожний з яких може включати LPSM, якщо фрейм включає два сегменти метаданих, то один може бути присутнім у полі addbsi фрейму, а інший - в полі AUX фрейму. У деяких варіантах здійснення винаходу кожний сегмент метаданих, що включає LPSM, включає сегмент інформаційного наповнення (або контейнера) LPSM, що має наступний формат:

заголовок (як правило, що включає синхрослово, що визначає початок інформаційного наповнення LPSM, а потім ідентифікаційні значення, наприклад, версію формату LPSM, довжину, період, число, асоціативні значення вкладеного потоку даних, зазначені нижче в таблиці 2); і після заголовка

щонайменше одне значення, що вказує діалог (наприклад, параметр "Канал(и) діалогу", наведений у таблиці 2), що вказує, мають або не мають ознаку діалогу відповідні аудіодані (наприклад, які канали відповідних аудіоданих мають ознаку діалогу);

щонайменше одне значення дотримання нормативних вимог щодо гучності (наприклад, параметр "Тип регулювання гучності", наведений у таблиці 2), що вказує чи відповідають відповідні аудіодані зазначеному пакету нормативних вимог щодо гучності;

щонайменше одне значення обробки гучності (наприклад, один або кілька параметрів "Прапор корекції стробованої гучності діалогу", "Тип корекції гучності", наведені в таблиці 2), що вказує щонайменше один тип обробки гучності, що був виконаний з відповідними аудіоданими; і

щонайменше одне значення гучності (наприклад, один або кілька параметрів "Відносна стробована гучність ITU", "Стробована гучність мови ITU", "Короткочасна гучність (3-секундний часовий інтервал) ITU (EBU 3341)" і "Дійсне пікове значення", наведені в таблиці 2), що вказує щонайменше одну характеристику гучності (наприклад, пікову або середню гучність) відповідних аудіоданих.

У деяких варіантах здійснення винаходу кожний сегмент, що містить метадані LPSM і метадані границі програми, містить заголовок фрейму (і в деяких випадках також додаткові основні елементи), а після заголовка фрейму (або заголовка фрейму та інших основних елементів) сегмент інформаційного наповнення (або контейнера) LPSM, що має наступний формат:

заголовок, як правило, що включає щонайменше одне ідентифікаційне значення (наприклад, версію формату LPSM, довжину, період, число, асоціативні значення вкладеного потоку даних, зазначені нижче в таблиці 2), і після заголовка - LPSM і метадані границі програми. Метадані границі програми можуть включати число фреймів до границі програми та значення коду (наприклад, значення "offsetexist"), що вказує, чи містить фрейм тільки число фреймів до границі програми або й число фреймів до границі програми, і значення зсуву), і (у деяких випадках) значення зсуву.

У деяких втіленнях аналізатор 205 (і/або ланка 202 декодера) виконаний з можливістю добування із сегмента зайвих бітів або поля "addbsi", або поля допоміжних даних фрейму бітового потоку, причому кожний сегмент метаданих має наступний формат:

заголовок фрейму (як правило, що включає синхрослово, що вказує початок сегмента метаданих, за яким йде щонайменше одне ідентифікаційне значення, наприклад, версія основного елемента, довжина й період, число розширених елементів й асоціативні значення вкладеного потоку даних, зазначені нижче в таблиці 1); і

після заголовка фрейму щонайменше одне захисне значення (наприклад, HMAC дайджест і значення цифрового відбитка аудіоданих, наведені в таблиці 1), що підходить щонайменше для однієї з операцій: декодування, перевірки дійсності або перевірки правильності щонайменше одного з: метаданих стану обробки гучності або відповідних аудіоданих); і

також після заголовка фрейму, якщо сегмент метаданих включає LPSM, ідентифікатор інформаційного наповнення LPSM і значення розміру інформаційного наповнення LPSM, які ідентифікують наступні за ними метадані як інформаційне наповнення LPSM і вказують розмір інформаційного наповнення LPSM.

Сегмент інформаційного наповнення (або контейнера) LPSM (переважно, що має формат, зазначений вище) йде за ідентифікатором інформаційного наповнення LPSM і значеннями розміру інформаційного наповнення LPSM.

У цілому кодований бітовий аудіопотік, сформований переважним варіантами здійснення винаходу, має структуру, що забезпечує механізм позначення елементів метаданих і підлеглих елементів як основних (обов'язкових) або розширених (необов'язкових елементів). Це дозволяє масштабувати в численних додатках швидкість передачі даних бітового потоку (що включає метадані). Основні (обов'язкові) елементи переважного синтаксису бітового потоку також повинні бути здатні подати сигнал про те, що розширені (додаткові) елементи, пов'язані з аудіоконтентом, присутні (усередині смуги) і/або перебувають у віддаленому місці (за межами смуги).

Основний елемент(и) повинен бути присутнім у кожному фреймі бітового потоку. Деякі підлеглі елементи основних елементів є необов'язковими й можуть бути представлені в будь-якій комбінації. Розширені елементи не повинні бути присутніми у кожному фреймі (щоб обмежити бітрейт-навантаження). Таким чином, розширені елементи в деяких фреймах можуть бути присутніми, а в інших - ні. Деякі підлеглі елементи розширеного елемента є необов'язковими та можуть бути представлені в будь-якій комбінації, при цьому деякі підлеглі елементи розширеного елемента можуть бути обов'язковими (тобто, якщо розширений елемент присутній у фреймі бітового потоку).

У класі варіантів здійснення винаходу формується (наприклад, за допомогою блока обробки звукового сигналу, що втілює винахід) кодований бітовий аудіопотік, що містить послідовність сегментів аудіоданих і сегментів метаданих. Сегменти аудіоданих є ознакою аудіоданих, кожний із щонайменше деяких сегментів метаданих включає метадані стану обробки гучності (LPSM) і в деяких випадках також метадані границі програми, а сегменти аудіоданих мультиплекуються із сегментами метаданих з часовим розділенням. У переважних варіантах цього класу кожний із сегментів метаданих має переважний формат, описуваний у даному описі.

В одному переважному форматі кодований бітовий потік є бітовим потоком формату AC-3 або бітовим потоком формату E-AC-3, а кожний із сегментів метаданих, що включає LPSM, входить (наприклад, за допомогою ланки 107 переважного втілення кодера 100) в якості додаткової інформації бітового потоку в полі "addbsi" (показаного на фіг. 6) інформаційного сегмента бітового потоку ("BSI") фрейму бітового потоку, або в полі допоміжних даних фрейму бітового потоку, або в сегменті зайвих бітів фрейму бітового потоку.

У переважному форматі кожний із фреймів у полі addbsi (або сегменті зайвих бітів) фрейму включає основний елемент, що має формат, наведений нижче в таблиці 1:

Таблиця 1

Параметр	Опис	Обов'язковий/ Необов'язковий
SYNC [ID]	Синхрослово може бути 16-бітним із установленим значенням 0x5838	Обов'язковий
Версія основного елемента		Обов'язковий
Довжина основного елемента		Обов'язковий
Період основного елемента (xxx)		Обов'язковий
Число розширених елементів	Показує кількість розширених елементів метаданих, пов'язаних з основним елементом. Це значення може збільшуватися/зменшуватися в міру проходження бітового потоку від виробітку до розподілу й кінцевого поширення.	Обов'язковий
Зв'язок вкладеного потоку даних	Описує, з яким вкладеним потоком (потоками) даних пов'язаний основний елемент.	Обов'язковий
Підпис (HMAC дайджест)	256-бітний HMAC дайджест (з використанням алгоритму SHA-2), обчислений за аудіоданими, основним елементом та всіма розширеними елементами усього фрейму.	Обов'язковий
Відлік у зворотному	Поле присутнє тільки в деякому числі фреймів у	Необов'язковий

порядку до границі програми	голові або хвості файлу/ потоку аудіопрограми. Таким чином, зміна версії основного елемента може бути використана для сигналізації про включення цього параметра.	
Цифровий відбиток аудіоданих	Цифровий відбиток аудіоданих береться по деякій кількості ІКМ аудіосемплів, представлених полем періоду основного елемента.	Необов'язковий
Цифровий відбиток відеоданих	Цифровий відбиток відеоданих береться по деякій кількості стиснутих відеосемплів (якщо такі є), представлених полем періоду основного елемента.	Необов'язковий
URL/UUID	Дане поле визначене для змісту URL і/або UUID (може бути надлишковим при цифровому відбитку), що посиляється на зовнішнє розміщення додаткового змісту програми (суті) і/або метаданих, пов'язаних з бітовим потоком.	Необов'язковий

У переважному форматі кожне з полів addbsi (або полів допоміжних даних) або сегментів зайвих бітів, які містять LPSM, що містять заголовок фрейму (і в деяких випадках також додаткові основні елементи), а після заголовка фрейму (або заголовка фрейму та інших основних елементів) - наступні значення LPSM (параметри):

5 ідентифікатор інформаційного наповнення (що визначає метадані як LPSM), що йде за значеннями основного елемента (наприклад, як зазначено в таблиці 1);

розмір інформаційного наповнення (що вказує розмір інформаційного наповнення LPSM), що йде за ідентифікатором інформаційного наповнення; і

10 дані LPSM (що йдуть за ідентифікатором інформаційного наповнення й значенням розміру інформаційного наповнення), що мають формат, показаний у наступній таблиці (таблиці 2):

Таблиця 2

Параметр LPSM [Інтелектуальна гучність]	ОПИС	Кількість унікальних станів	Обов'язковий/ Необов'язковий	Коефіцієнт розміщення (період оновлення параметра)
Версія LPSM			Обов'язковий	
LPSM період (XXX)	Застосовується тільки до XXX полів		Обов'язковий	
Число LPSM			Обов'язковий	
Зв'язаність вкладеного потоку LPSM			Обов'язковий	

Канал(и) діалогу	Указує, яка комбінації L, C & R аудіоканалів містить мову протягом попередніх 0,5 секунд Коли мова не присутня в будь-якій L, C або R комбінації, то цей параметр вказує "немає діалогу"	8	Обов'язковий	~0,5 секунди (типовий)
Тип регулювання гучності	Указує, що відповідний аудіопотік даних відповідно де визначеного набору правил (наприклад ATSC A/85 або EBL R128)	8	Обов'язковий	Фрейм
Прапор корекції стробованої гучності діалогу	Вказує, чи був скоректований зв'язаний аудіопотік на основі стробування діалогу	2	Необов'язковий (є присутнім тільки тоді, якщо тип регулювання гучності вказує, що відповідний звук без виправлення)	Фрейм
Тип корекції гучності	Указує, чи був скоректований аудіопотік за допомогою нескінченного прогнозування (на основі файлів) або за допомогою контролера гучності й динамічного діапазону, що працює в режимі реального часу.	2	Необов'язковий (є присутні тільки тоді, якщо тип регулювання гучності вказує що відповідні аудіодані не скоректовані)	Фрейм
Відносна стробована гучність ITU (INF)	Указує інтегральну гучність за ITU-R BS. 1770-3 зв'язаного аудіопотоку без застосування метаданих (наприклад, 7 біт: -58 → +5,5 LKFS із кроком 0,5 LKFS)	128	Необов'язковий	1 с
Стробована гучність мови ITU (INF)	Указує 1/3 інтегрально гучності мови/ діалогу за ITU-R BS. 1770-3 зв'язаного аудіопотоку без застосування метаданих (наприклад 7 біт: -58 → +5,5 LKFS із кроком 0,5 LKFS)	128	Необов'язковий	1 с

Короткочасна (3-секундний часовий інтервал) гучність за ITU (EBU3341)	Указує нестробовану гучність протягом 3 секунд за ITU (ITU-BS.1771-1) зв'язаного аудіопотоку без застосування метаданих (ковзне вікно) @ ~ оновлення вимірювання 10 Гц (наприклад, 8 біт 116 → + 11,5 LKFS із кроком 0,5 LKFS)	256	Необов'язковий	0,1 с
Дійсне пікове значення	Указує значення TruePeak (дБ TP) за ITU-R BS. 1770-3 Додаток 2, зв'язаного аудіопотоку без застосування метаданих (тобто, найбільше значення за період фрейму, повідомлене в поле періоду елемента) 116→ 11,5 LKFS із кроком 0,5 LKFS	256	Необов'язковий	0,5 с
Зсув понижувального мікшування	Указує зсув понижувального мікшування гучності			
Границя програми	Указує у фреймах, коли зустрінеться або зустрілася границя програми. Коли границя програми не є границею фрейму, додатковий зсув семплів вкаже, як далеко у фреймі перебуває границя поточної програми.			

В іншому переважному форматі кодованого бітового потоку, згенерованого відповідно до винаходу, бітовий потік являє собою бітовий потік формату AC-3 або бітовий потік формату E-AC-3, і кожний із сегментів метаданих, що включає LPSM (і факультативно також метадані границі програми), включається (наприклад, ланкою 107 переважного варіанта здійснення кодера 100) у кожне з: сегмента зайвих бітів фрейму бітового потоку; або поля "addbsi" (зображено на фіг. 6) інформаційного сегмента бітового потоку ("BSI") фрейму бітового потоку; або поля допоміжних даних (наприклад, сегмента AUX, зображеного на фіг. 4) наприкінці фрейму бітового потоку. Фрейм може включати один або два сегменти метаданих, кожний з яких включає FPSM, а якщо фрейм включає два сегменти метаданих, один може бути присутнім у полі фрейму addbsi, а інший - в полі фрейму AUX. Кожний сегмент метаданих, що включає FPSM, має формат, визначений вище з посиланням на таблиці 1 і 2, наведені вище (тобто він включає основні елементи, зазначені в таблиці 1, з наступним ID інформаційного наповнення (ідентифікуючим метадані як FPSM) і значення розміру інформаційного наповнення, визначені вище, за яким іде інформаційне наповнення (дані FPSM, які мають формат, зазначений у таблиці 2).

В іншому переважному форматі кодований бітовий потік є бітовим потоком формату Dolby E, а кожний із сегментів метаданих, що включає FPSM (а в деяких випадках метадані границі програми), входить в оточення перших N семплів захисного частотного інтервалу Dolby E. Бітовий потік формату Dolby E, що включає такий сегмент метаданих, що включає FPSM, переважно включає значення, що вказує довжину інформаційного наповнення FPSM, що повідомляється в слові Pd преамбули згідно з SMPTE 33 7M (частота повторення слова Pa згідно з SMPTE 337M переважно залишається ідентичною зв'язаній частоті зміни відеокадрів).

У переважному форматі, у якому кодований бітовий потік являє собою бітовий потік формату E-AC-3. кожний із сегментів метаданих, що містить FPSM (і факультативно також

метадані границі програми), включається (наприклад, ланкою 107 переважного варіанта реалізації кодера 100) як додаткова інформація бітового потоку в сегмент зайвих бітів, у поле "addbsi" інформаційного сегмента бітового потоку ("BSr") фрейму бітового потоку. Далі описуються додаткові аспекти кодування бітового потоку формату E-AC-3 з LPSM у цьому переважному форматі:

1. під час створення бітового потоку формату E-AC-3, у той час як кодер E-AC-3 (який вставляє значення LPSM у бітовий потік) "активний", для кожного створеного фрейму (синхрофрейму), бітовий потік повинен містити блок метаданих (що включає LPSM), розташований у полі addbsi (або сегменті зайвих бітів) фрейму. Біти, які повинні нести блок метаданих, не повинні збільшувати бітрейт кодера (довжину фрейму);

2. Кожний блок метаданих (що містить LPSM) повинен містити наступну інформацію:

прапор_типу_корекції_гучності: де "1" означає, що гучність відповідних аудіоданих була скоректована до кодера, а "0" означає, що гучність була скоректована коректором гучності, вбудованим у кодер (наприклад, процесором 103 корекції гучності кодера 100, зображеному на фіг. 2);

мовний канал: відображає, який вихідний канал(и) містить мова (за попередні 0,5 с). Якщо мова не виявлена, то це повинне бути зазначене як таке;

гучність_мови: позначає інтегральну гучність мови кожного відповідного аудіоканалу, що містить мову (за попередні 0,5 с);

гучність_ITU: позначає інтегровану гучність згідно ITU BS. 1770-3 кожного відповідного аудіоканалу; і

посилення: складене (складені) посилення гучності для зворотної дії в декодері (для демонстрації здатності до зворотної дії);

3. У той час, коли кодер формату E-AC-3 (який вставляє значення LPSM у бітовий потік) "активний" і приймає фрейм формату AC-3 із прапором "надійності", контролер гучності в кодері (наприклад, процесор 103 корекції гучності кодера 100, зображеного на фіг. 2) повинен бути обійдений. Значення dialnorm і DRC "надійного" джерела повинні бути пропущені через (наприклад, генератором 106 кодера 100) у компонент кодера E-AC-3 (наприклад, ланку 107 кодера 100). Генерування блоків LPSM триває, і прапор_типу_корекції_гучності встановлюється в "1". Послідовність обходу контролера гучності повинна бути синхронізована з початком декодованого фрейму формату AC-3. де з'являється прапор "надійності". Послідовність обходу контролера гучності повинна здійснюватися в такий спосіб: регулятор ступеня підтримувача гучності зменшується зі значення 9 до значення 0 протягом 10 періодів аудіоблоків (тобто 53,3 мс), регулятор кінцевого вимірювача підтримувача гучності розміщується в обхідний режим (ця операція повинна привести до плавного переходу). Термін "надійний" обхід регулятора рівня означає, що значення dialnorm бітового потоку джерела також повторно використовується на виході кодера (наприклад, якщо "надійний" бітовий потік джерела має значення dialnorm, рівне -30, то вихід кодера повинний використовуватися -30 для вихідного значення dialnorm);

4. У той час, коли кодер формату E-AC-3 (який вставляє значення LPSM у бітовий потік) "активний" і приймає фрейм формату AC-3 без прапора "надійності", контролер гучності в кодері (наприклад, процесор 103 корекції гучності кодера 100, зображеного на фіг. 2) повинен бути активний. Генерування блоків LPSM триває, і прапор_типу_корекції_гучності встановлюється в "0". Активаційна послідовність контролера гучності повинна бути синхронізована з початком декодованого фрейму формату AC-3, де зникає прапор "надійності". Активаційна послідовність контролера гучності повинна здійснюватися в такий спосіб: регулятор ступеня підтримувача гучності збільшується від значення 0 до значення 9 протягом 1 періоду аудіоблоків. (тобто 5,3 мс) і регулятор кінцевого вимірювача підтримувача гучності приводиться в "активний" режим (ця операція повинна привести до плавного переходу й включати інтегральне скидання кінцевого вимірювача); і

5. під час кодування, графічний інтерфейс користувача (GUI) повинен відображати користувачеві наступні параметри: "Вхідна аудіопрограма:

[Надійна/Ненадійна]" - стан цього параметра заснований на наявності прапора "надійності" у вхідному сигналі; і "Корекція гучності в реальному часі: [Включена/Виключена]" - стан цього параметра заснований на тому, чи активний цей контролер гучності, вбудований у кодер. При декодуванні бітового потоку формату AC-3 або E-AC-3, що має LPSM (у переважному форматі), включений у сегмент зайвих бітів, або поле "addbsi" інформаційного сегмента бітового потоку ("BSr"), кожного фрейму бітового потоку, декодер повинен піддавати синтаксичному аналізу дані блоку LPSM (у сегменті зайвих бітів або полі addbsi) і проводити всі з добутих значень LPSM у графічний інтерфейс користувача (GUI). Набір добутих значень LPSM оновлюється кожного фрейму.

В іншому переважному форматі кодованого бітового потоку, згенерованому відповідно до винаходу, кодований бітовий потік являє собою бітовий потік формату AC-3 або бітовий потік формату E-AC-3, і кожний із сегментів метаданих, що включає LPSM, включається (наприклад, ланкою 107 переважної реалізації кодера 100) у сегмент зайвих бітів, або в сегмент Aux, або як

5 додаткова інформація бітового потоку в поле "addbsi" (показаному на фіг. 6) інформаційного сегмента бітів потік ("BSI") фрейму бітового потоку. У цьому форматі (який є різновидом формату, описаного вище з посиланнями на таблиці 1 і 2), кожне з полів addbsi (або Aux, або зайвих бітів), що містить LPSM, містить наступні значення LPSM:

10 основні елементи, наведені в таблиці 1, за якими йде ID інформаційного наповнення (що ідентифікує метадані як LPSM), і значення розміру інформаційного наповнення, за якими йде інформаційне наповнення (LPSM дані), що має наступний формат (подібний до обов'язкових елементів, наведених у таблиці 2 вище):

версія інформаційного наповнення LPSM: 2-бітне поле, що відображає версію інформаційного наповнення LPSM;

15 dialchan: 3-бітне поле, що відображає чи містить розмовний діалог Лівий, Правий і/або Центральний канали відповідних аудіоданих. Розподіл бітів поля dialchan може бути наступним: біт 0, що відображає

20 наявність діалогу в лівому каналі, зберігається в найстаршому біті поля dialchan; і біт 2, що відображає наявність діалогу в центральному каналі, зберігається в наймолодшому біті поля dialchan.

Кожний біт поля dialchan устанавлюється в "1", якщо відповідний канал містить розмовний діалог під час попередніх 0,5 с програми;

loudregtyp: 4-бітне поле, що відображає те, якому стандарту нормативних вимог щодо гучності задовольняє гучність програми. Установка поля "loudregtyp" в "000" указує на те, що LPSM не відображає відповідність нормативним вимогам щодо гучності. Наприклад, одне значення цього поля (наприклад, 0000) може означати, що відповідність стандарту нормативних вимог по гучності не зазначена, інше значення цього поля (наприклад, 0001) може вказувати на те, що аудіодані програми відповідають стандарту ATSC A/85, і інше значення цього поля (наприклад, 0010) указувати на те, що аудіодані програми відповідають стандарту EBU R128. У зазначеному прикладі, якщо поле встановлене в будь-яке інше значення, відмінне від "0000",

30 поля loudcorrldialgat і loudcorrtyp повинні йти слідом в інформаційному наповненні; loudcorrldialgat: однобітне поле, що вказує, якщо була застосована корекція стробованої гучності діалогу. Якщо гучність програми була скоректована з використанням стробування діалогу, значення поля loudcorrldialgat устанавлюється в "1". Інакше воно встановлюється в "0";

35 loudcorrtyp: однобітне поле, що вказує тип корекції гучності, застосовуваної до програми. Якщо гучність програми була скоректована за допомогою операції нескінченної прогнозової (файлової) корекції гучності, значення поля loudcorrtyp устанавлюється в "0". Якщо гучність програми була скоректована з використанням комбінації вимірювання гучності в реальному часі й керування динамічним діапазоном, значення цього поля встановлюється в "1";

40 loudrelgate: однобітне поле, що вказує чи існують дані відносної стробованої гучності (ITU). Якщо поле loudrelgate устанавлюється в "1", 7-бітне поле ituloudrelgat повинне йти слідом в інформаційному наповненні;

45 loudrelgat: 7-бітне поле, що вказує відносну стробовану гучність програми (ITU). Це поле вказує інтегральну гучність аудіопрограми, виміряну відповідно до ITU-R BS. 1770-3 без яких-небудь регулювань посилення внаслідок застосування dialnorm і стиснення динамічного діапазону. Значення від 0 до 127 інтерпретуються як від -58 LKFS до +5,5 LKFS, із кроком 0,5 LKFS;

50 loudspchgate: однобітне поле, що вказує чи існують дані стробованої гучності мови (ITU). Якщо поле loudspchgate устанавлюється в "1", 7-бітне поле loudspchgat повинне йти слідом в інформаційному наповненні;

55 loudspchgat: 7-бітне поле, що вказує стробовану гучність мови програми. Це поле вказує інтегральну гучність всієї відповідної аудіопрограми, виміряну відповідно до формули (2) згідно з ITU-R BS.1770-3 і без яких-небудь регулювань посилення внаслідок застосування dialnorm і стиснення динамічного діапазону. Значення від 0 до 127 інтерпретуються як від -58 до +5,5 UKFS, із кроком 0,5 UKFS;

loudstnn3se: однобітне поле, що вказує чи існують дані короткострокової (3 с) гучності. Якщо поле встановлене в "1", 7-бітне поле loudstrm3s повинне йти слідом в інформаційному наповненні;

60 loudstrm3s: 7-бітне поле, що вказує нестробовану гучність за попередні 3 секунди відповідної аудіопрограми, виміряну відповідно до ITU-R BS. 1771-1 і без яких-небудь

регулювань посилення внаслідок застосування `dialnorm` і стиснення динамічного діапазону. Значення від 0 до 256 інтерпретуються як від -116 UKFS до + 11.5 UKFS, із кроком 0,5 UKFS;

`truepk`: однобітне поле, що вказує чи існують дані гучності за дійсними піками. Якщо поле `truepk` установлене в "1", 8-бітне поле `truepk` повинне йти слідом в інформаційному наповненні;

5 i

`truepk`: 8-бітне поле, що вказує вибіркове значення дійсного піка програми, виміряне відповідно до Annex 2 згідно з ITU-R BS. 1770-3 і без яких-небудь регулювань посилення внаслідок застосування `dialnorm` і стиснення динамічного діапазону. Значення від 0 до 256 інтерпретуються як від -116 LKFS до +11,5 LKFS, із кроком 0,5 LKFS.

10

У деяких варіантах здійснення основний елемент сегмента метаданих у сегменті зайвих бітів або в полі `auxdata` (або `addbsi`) фрейму бітового потоку формату AC-3 або бітового потоку формату E-AC-3 містить заголовок фрейму (зазвичай включає ідентифікаційні значення, наприклад, версію основного елемента), і після заголовка фрейму: значення, що вказують чи включаються дані цифрового відбитка (або інші захисні значення) у метадані сегмента метаданих, значення, що вказують чи існують зовнішні дані (пов'язані з аудіоданими, що відповідають метаданим сегмента метаданих), ID інформаційного наповнення й значення розміру інформаційного наповнення для кожного типу метаданих (наприклад, LPSM і/або метадані типу, відмінного від LPSM), ідентифікованих основним елементом, і захисні значення для щонайменше одного типу метаданих, ідентифікованих основним елементом. Інформаційне (інформаційні) наповнення метаданих сегмента метаданих йдуть за заголовком фрейму, і (у деяких випадках) вкладене в значення основного елемента.

15

20

Типові варіанти здійснення даного винаходу включають ефективні метадані границі програми в кодованому бітовому аудіопотоці, що дозволяють виконати точне й надійне визначення щонайменше однієї границі між послідовними аудіопрограмами, вказаними бітовим потоком. Типові варіанти здійснення забезпечують точне й надійне визначення границі програми в тому сенсі, що вони дозволяють точно визначити границю програми, навіть у тих випадках, коли бітові потоки, що вказують різні програми, змонтовані один з одним (для формування бітового потоку відповідно до винаходу) таким чином, що обрізаний один або обидва змонтованих бітових потоки (і, таким чином, добуті метадані границі програми, які входили щонайменше в один з бітових потоків до монтажу).

25

30

У типових варіантах здійснення метадані границі програми у фреймі бітового потоку, відповідно до винаходу, являють собою прапор границі програми, що вказує число фреймів. Як правило, прапор вказує кількість фреймів між поточним фреймом (фреймом, що містить у собі прапор) і границею програми (початком або кінцем поточної аудіопрограми). У деяких переважних варіантах здійснення винаходу прапори границі програми розставляють симетрично, ефективним способом на початку й наприкінці кожного сегмента бітового потоку, що вказує одну програму (тобто у фреймах, що зустрічаються протягом деякого заданого числа фреймів після початку сегмента, і у фреймах, що зустрічаються протягом деякого заданого числа фреймів до кінця сегмента), таким чином, коли два таких сегменти бітового потоку з'єднуються (тобто буде присутня ознака послідовності двох програм), метадані границі програми можуть бути присутніми (наприклад, симетрично) на обох сторонах границі між двома програмами.

35

40

Максимальна надійність може бути досягнута за рахунок вставки прапора границі програми в кожний фрейм бітового потоку, що вказує програму, але це зазвичай неможна застосувати на практиці внаслідок відповідного збільшення швидкості даних.

45

У типових варіантах здійснення прапори границі програми вставляються тільки у вкладений набір фреймів кодованого бітового аудіопотоку (який може вказувати одну аудіопрограму або послідовність аудіопрограм), і коефіцієнт розміщення прапора границі є незростаючою функцією в залежності від збільшення інтервалу між кожним із фреймів бітового потоку (у якому прапор установлений) і границею програми, що ближче до зазначеного фрейму, де "коефіцієнт розміщення прапора границі" є середнім значенням відношення кількості фреймів (що вказують програму), які містять у собі прапори границі програми до числа фреймів (що вказують програму), які не містять у собі прапор границі програми, де середнє значення є ковзним середнім кількості (наприклад, відносно невеликого числа) послідовних фреймів кодованого бітового аудіопотоку.

50

55

Збільшення коефіцієнта розміщення прапора (наприклад, у місцях у бітовому потоці ближче до границі програми) збільшує швидкість даних, необхідну для доставки бітового потоку. Щоб компенсувати це, розмір (кількість біт) кожного вставленого прапора переважно зменшується при збільшенні коефіцієнта розміщення прапора (наприклад, так, що розмір прапора границі програми в "N"-ому фреймі бітового потоку, де N - ціле число, являє собою не зростаючу

60

функцію відстані (кількості фреймів) між "N"-м фреймом і найближчою границею програми). У класі варіантів здійснення винаходу коефіцієнт розміщення прапора границі логарифмічно спадає по мірі збільшення інтервалу (від кожного місця вставки прапора) до найближчої границі програми, а для кожного фрейму, що містить прапор, що містить у собі один із прапорів, розмір прапора в зазначеному фреймі, що містить прапор, дорівнює або більше, ніж розмір кожного прапора у фреймі, розташованому ближче до найближчої границі програми, ніж зазначений фрейм, що містить прапор. Як правило, розмір кожного прапора визначається зростаючою функцією кількості фреймів від місця вставки прапора до найближчої границі програми.

Наприклад, розглянемо варіант здійснення на фіг. 8 і 9, у якому кожна колонка, позначена номером фрейму (у верхньому ряді), відображає фрейм кодованого бітового аудіопотоку. Бітовий потік відображає аудіопрограму, що має першу границю програми (що вказує початок програми), що йде відразу ліворуч від колонки, позначеної номером фрейму "17" з лівої сторони на фіг. 9, і другу границю програми (що вказує кінець програми), що йде відразу праворуч від колонки, позначеної номером фрейму "1" із правої сторони на фіг. 8. Прапори границі програми, включені у фрейми, зображені на фіг. 8, відраховують у зворотному порядку кількість фреймів між поточним фреймом і другою границею програми. Прапори границі програми, включені у фрейми, зображені на фіг. 9, відраховують у прямому порядку кількість фреймів між поточним фреймом і першою границею програми.

У варіанті здійснення згідно з фіг. 8 і 9 прапор границі програми вставляється тільки в кожний з 2^N -х фреймів перших X фреймів кодованого бітового потоку після початку аудіопрограми, що відображається бітовим потоком, і в кожний з 2^N -х фреймів (з останніх X фреймів бітового потоку), які є найближчими до кінця програми, що відображається бітовим потоком, де програма містить Y фреймів, X - ціле, що менше або дорівнює $Y/2$, і N - позитивне ціле в діапазоні від 1 до $\log_2(X)$. У такий спосіб (як показано на фіг. 8 і 9), прапор границі програми вставляється в другий фрейм (N=1) бітового потоку (найближчий до початку програми фрейм, що містить прапор), у четвертий фрейм (N=2), у восьмий фрейм (N=3), і так далі, і у восьмий фрейм від кінця бітового потоку, у четвертий фрейм від кінця бітового потоку, і в другий фрейм від кінця бітового потоку (найближчий до кінця програми фрейм, що містить прапор). У цьому прикладі прапор границі програми в 2^N -ому фреймі від початку (або кінця) програми містить $\log_2(2^{N+2})$ двійкових розрядів, як відображено на фіг. 8 і 9. Таким чином, прапор границі програми в другому фреймі (N=1) від початку (або кінця) програми містить $\log_2(2^{N+2}) = \log_2(2^3) = 3$ двійкових розрядів, і прапор у четвертому фреймі (N=2) від початку (або кінця) програми містить $\log_2(2^{N+2}) = \log_2(2^4) = 4$ двійкових розрядів, і так далі.

У прикладі на фіг. 8 і 9 формат кожного прапора границі програми є наступним. Кожний прапор границі програми складається з початкового "Г"-ного біта, послідовності "0"-вих бітів (або без "0"-го біта або з одним або декількома послідовними "0"-ми бітами) після початкового біта й двобітного кінцевого коду. Як показано на фіг. 8, кінцевий код становить "11" для прапорів в останніх X фреймах бітового потоку (фреймах, найближчих до кінця програми). Як показано на фіг. 9, кінцевий код становить "10" для прапорів у перших X фреймах бітового потоку (фреймах, найближчих до початку програми). Таким чином, для зчитування (декодування) кожного прапора враховується кількість нулів між початковим "1"-им бітом і кінцевим кодом. Якщо кінцевий код визначений як "11", прапор вказує, що між поточним фреймом (фреймом, що містить прапор) і кінцем програми присутні (2^Z+1-1) фреймів, де Z - кількість нулів між початковим "1"-м бітом і кінцевим кодом прапора. Декодер може бути ефективно реалізований для ігнорування першого й останнього біта кожного такого прапора, для визначення інверсії послідовності інших (проміжних) бітів прапора (наприклад, якщо послідовність проміжних бітів являє собою "0001" з "1"-м бітом, що є останнім бітом у послідовності, інвертована послідовність проміжних бітів являє собою "1000" з "Г"-м бітом, що є першим бітом в інвертованій послідовності), і для визначення двійкового значення інвертованої послідовності проміжних бітів як індекс поточного фрейму (фрейму, у який включений зазначений прапор) відносно кінця програми. Наприклад, якщо інвертована послідовність проміжних бітів являє собою "1000", ця інвертована послідовність має двійкове значення $2^4=16$, і фрейм визначається як 16-й фрейм перед кінцем програми (як показано в колонці на фіг. 8, що описує фрейм "0").

Якщо кінцевий код визначений як "10", прапор вказує, що між початком програми і поточним фреймом (фреймом, що містить прапор) присутні $(2^{Z+1}-1)$ фреймів, де Z - кількість нулів між початковим "1"-им бітом і кінцевим кодом прапора. Декодер може бути ефективно реалізований для ігнорування першого й останнього біта кожного такого прапора, для визначення інверсії послідовності проміжних бітів прапора (наприклад, якщо послідовність проміжних бітів являє собою "0001" з "1"-м бітом, що є останнім бітом у послідовності, інвертована послідовність проміжних бітів являє собою "1000" з "1"-м бітом, що є першим бітом в інвертованій

послідовності), і для визначення двійкового значення інвертованої послідовності проміжних бітів в якості індексу поточного фрейму (фрейму, у який включений зазначений прапор) відносно початку програми. Наприклад, якщо інвертована послідовність проміжних бітів являє собою "1000", ця інвертована послідовність має двійкове значення $2^4=16$, і фрейм визначається як 16-й фрейм після початку програми (як показано в колонці на фіг. 9, що описує фрейм "32").

У прикладі на фіг. 8 і 9, прапор границі програми присутній тільки в кожному з 2^N -х фреймів перших X фреймів кодованого бітового потоку після початку аудіопрограми, що відображується бітовим потоком, і в кожному з 2^N -х фреймів (з останніх X фреймів бітового потоку), найближчих до кінця програми, що

відображається бітовим потоком, де програма містить Y фреймів, X - ціле, що менше або дорівнює $Y/2$, і N - позитивне ціле в діапазоні від 1 до $\log_2(X)$. Включення прапорів границі програми додає тільки середнє значення бітрейта, рівне 1,875 біт/фрейм, до бітрейта, необхідному для передачі бітового потоку без прапорів.

У типовій реалізації варіанта здійснення з фіг. 8 і 9, у якій бітовий потік являє собою кодований бітовий аудіопотік формату AC-3, кожний фрейм містить аудіоконтент і метадані для 1536 семплів цифрового звукозапису. Це представляє 32 мілісекунди цифрового звукозапису або швидкість звукозапису 31,25 фреймів в секунду для частоти дискретизації 48 кГц. Таким чином, у такому варіанті здійснення прапор границі програми у фреймі, відділеному деякою кількістю фреймів (" X " фреймів) від границі програми, указує, що границя виникає через $32X$ мілісекунди після кінця фрейму, що містить прапор (або за $32X$ мілісекунди перед початком фрейму, що містить прапор).

У типовій реалізації варіанта здійснення з фіг. 8 і 9, у якій бітовий потік являє собою кодований бітовий аудіопотік формату E-AC-3, кожний фрейм бітового потоку містить аудіоконтент і метадані для 256, 512, 768 або 1536 семплів цифрового звукозапису, залежно від того, чи містить фрейм один, два, три або шість блоків аудіоданих відповідно. Для частоти дискретизації 48 кГц це представляє 5,333, 10,667, 16 або 32 мілісекунди цифрового аудіозапису відповідно або швидкість аудіозапису 189,9, 93,75, 62,5 або 31,25 фреймів у секунду відповідно. Таким чином, у такому варіанті здійснення (за умови, що кожний фрейм представляє 32 мілісекунди цифрового звукозапису) прапор границі програми у фреймі, відділеному деякою кількістю фреймів (" X " фреймів) від границі програми, указує, що границя перебуває за $32X$ мілісекунди після кінця фрейму, що містить прапор (або за $32X$ мілісекунди перед початком фрейму, що містить прапор).

У деяких варіантах здійснення, в яких границя програми може перебувати усередині фрейму бітового аудіопотоку (тобто, не збігається з початком або кінцем фрейму), метадані границі програми, включені у фрейм бітового потоку, містять число фреймів до границі програми (тобто метадані, що вказують кількість повних фреймів між початком або кінцем фрейму, що містить число фреймів, і границею програми) і деяке значення зсуву. Значення зсуву вказує зсув (звичайне число семплів) між початком або кінцем фрейму, що містить границю програми, і дійсним положенням границі програми усередині фрейму, що містить границю програми.

Кодований бітовий аудіопотік може відображати послідовність програм (звукових доріжок) відповідної послідовності відеопрограм, і границі таких аудіопрограм, як правило, перебувають на границях відеофреймів, а не на границях аудіофреймів. Також деякі аудіокодеки (наприклад, кодеки формату E-AC-3) використовують розміри аудіофреймів, які не збігаються з відеофреймами. Також у деяких випадках спочатку кодований бітовий аудіопотік піддається перекодуванню для створення перекодованого бітового потоку, і спочатку кодований бітовий потік має розмір фрейму, що відрізняється від розміру фрейму перекодованого бітового потоку, так що границя програми (визначена за спочатку кодованим бітовим потоком) не обов'язково буде перебувати на границі фрейму перекодованого бітового потоку. Наприклад, якщо спочатку кодований бітовий потік (наприклад, бітовий потік "ГЕВ" на фіг. 10) має розмір фрейму, рівний 1536 семплам на фрейм, і перекодований бітовий потік (наприклад, бітовий потік "ТВ" на фіг. 10) має розмір фрейму, рівний 1024 семплам на фрейм, процес перекодування може привести до того, що дійсна границя програми буде перебувати не на границі фрейму перекодованого бітового потоку, а десь у його фреймі (наприклад, 512 семпл у фреймі перекодованого бітового потоку, як показано на фіг. 10), внаслідок відмінності розмірів фреймів різних кодеків. Варіанти здійснення даного винаходу, у яких метадані границі програми включені у фрейм кодованого бітового аудіопотоку, містять значення зсуву, а також число фреймів до границі програми, корисні в трьох випадках, відзначених у цьому розділі (а також в інших випадках).

Варіант здійснення, описаний вище з посиланням на фіг. 8 і 9, не містить значення зсуву (наприклад, поле зсуву) у якому-небудь із фреймів кодованого бітового потоку. У різновидах цього варіанта здійснення значення зсуву включається в кожний фрейм кодованого бітового

аудіопотоку, що включає прапор границі програми (наприклад, у фреймах, що відповідають фреймам під номерами 0, 8, 12 і 14 на фіг. 8, і фреймах під номерами 18, 20, 24 і 32 на фіг. 9).

У класі варіантів здійснення структура даних (у кожному фреймі кодованого бітового потоку, що містить метадані границі програми відповідно до винаходу) містить значення коду, що вказує, чи містить фрейм тільки число фреймів до границі програми або як число фреймів до границі програми, так і значення зсуву. Наприклад, значення коду може являти собою значення однобітного поля (поле, що згадується тут як "offsetexist"), значення "offset_exist" = 0 може вказувати на те, що у фрейм не включене значення зсуву, а значення "offsetexist" = 1 може вказувати на те, що у фрейм включене як число фреймів до границі програми, так і значення зсуву.

У деяких варіантах здійснення щонайменше один фрейм кодованого бітового аудіопотоку формату AC-3 або E-AC-3 містить сегмент метаданих, що містить LPSM і метадані границі програми (і факультативно також інші метадані) для аудіопрограми, утвореної бітовим потоком. Кожний такий сегмент метаданих (який може бути включений у поле addbsi, або поле допоміжних даних, або сегмент зайвих бітів бітового потоку) містить заголовок фрейму (і в деяких випадках також додаткові основні елементи), і після заголовка фрейму (або заголовка фрейму й інших основних елементів) сегмент інформаційного наповнення LPSM (або контейнер), що має наступний формат:

заголовок, як правило, що включає щонайменше одне ідентифікаційне значення (наприклад, версію формату LPSM, довжину, період, число і асоціативні значення вкладеного потоку даних), і

після заголовка, метадані границі програми (які можуть включати число фреймів до границі програми, значення коду (наприклад, значення "offsetexist"), що вказує, чи містить фрейм тільки число фреймів до границі програми або як число фреймів до границі програми, так і значення зсуву, і в деяких випадках значення зсуву) і LPSM. LPSM може включати:

щонайменше одне значення, що вказує діалог, що вказує одне із двох - відповідні аудіодані мають ознаку діалогу або не мають ознаки діалогу (наприклад, які канали відповідних аудіоданих мають ознаку діалогу). Значення, що вказує діалог, може вказати, чи є присутнім діалог у будь-якій комбінації каналів або у всіх каналах відповідних аудіоданих;

щонайменше одне значення дотримання нормативних вимог щодо гучності, що вказує чи відповідають відповідні аудіодані зазначеному пакету нормативних вимог щодо гучності:

щонайменше одне значення обробки гучності, що вказує щонайменше один тип обробки гучності, що був виконаний з відповідними аудіоданими; і

щонайменше одне значення гучності, що вказує щонайменше одну характеристику гучності (наприклад, пікове або середнє значення гучності) відповідних аудіоданих.

У деяких варіантах здійснення сегмент інформаційного наповнення LPSM містить значення коду (значення "offsetexist"), що вказує, чи містить фрейм тільки число фреймів до границі програми або як число фреймів до границі програми, так і значення зсуву. Наприклад, в одному такому варіанті здійснення, коли таке значення коду вказує (наприклад, коли offset_exist=1), що фрейм містить число фреймів до границі програми й значення зсуву, сегмент інформаційного наповнення LPSM може містити значення зсуву, що представляє 11 - бітне беззнакове ціле (тобто значення, що має, від 0 до 2048) і яке вказує кількість додаткових аудіосемплів між сигнальною границею фрейму (границею фрейму, що містить границю програми) і дійсною границею програми. Якщо число фреймів до границі програми вказує кількість фреймів (при поточній частоті фреймів) до фрейму, що містить границю програми, точне положення (в одиницях кількості семплів) границі програми (відносно початку або кінця фрейму, що включає сегмент інформаційного наповнення LPSM) буде розраховуватися як:

$$S = (\text{лічильник_фреймів} \times \text{розмір фрейму}) + \text{зсув}$$
, де S - кількість семплів до границі програми (від початку або кінця фрейму, що включає сегмент інформаційного наповнення LPSM), "лічильник_фреймів" - число фреймів, що вказується числом фреймів до границі програми, "розмір фрейму" - кількість семплів на фрейм, і "зсув" - кількість семплів, що вказується значенням зсуву.

Деякі варіанти здійснення, у яких коефіцієнт розміщення прапорів границі програми збільшується біля дійсної границі програми, реалізують правило, що полягає в тому, що значення зсуву ніколи не включається у фрейм, якщо фрейм відстоїть від фрейму, що включає границю програми, на число фреймів, яке менше або дорівнює деякому числу ("Y"). Як правило, Y=32. Для кодера формату E-AC-3, що реалізує це правило (з Y=32), кодер ніколи не вставляє значення зсуву в кінцеву секунду аудіопрограми. У цьому випадку приймаючий пристрій відповідає за організацію таймера й, таким чином, виконання свого власного обчислення (у відповідь на метадані границі програми, включаючи значення зсуву, у фреймі кодованого

бітового потоку, що відстоїть від фрейму, що містить границю програми, на більш, ніж γ фреймів).

Для програм, чиї аудіопрограми відомі як "співпадаючі за фреймами" з відеофреймами відповідних відеопрограм (наприклад, звичайної передачі у вигляді кодованого звукозапису у форматі Dolby E), буде зайвим включати значення зсуву в кодовані бітові потоки, що вказують аудіопрограми. Таким чином, значення зсуву, як правило, не будуть включатися в такі кодовані бітові потоки.

З посиланням до фіг. 11, далі розглянемо випадки, у яких кодовані бітові аудіопотоки монтуються разом, щоб створювати варіант здійснення бітового аудіопотоку відповідно до винаходу.

Бітовий потік угорі фіг. 11 (підписаний як "Сценарій 1") вказує всю першу аудіопрограму (P1), що включає метадані границі програми (прапори границі програми, F), за якою йде вся друга аудіопрограма (P2), що також включає метадані границі програми (прапори границі програми, F). Прапори границі програми в кінцевій частині першої програми (деякі з яких показані на фіг. 11) такі ж або схожі на ті, які описані з посиланням на фіг. 8, і визначають положення границі між двома програмами (тобто границі на початку другої програми). Прапори границі програми в початковій частині другої програми (деякі з яких показані на фіг. 11) такі ж або схожі на ті, які описані з посиланням на фіг. 9, і вони також визначають положення границі. У типових варіантах здійснення кодер або декодер реалізує таймер (калібрований прапорами в першій програмі), що відраховує у зворотному порядку до границі програми, і той же таймер (калібрований прапорами в другій програмі) відраховує в прямому порядку від тієї ж границі програми. Як показано на графіку таймера границі в сценарії 1 на фіг. 11, такий відлік таймера у зворотному порядку (калібрований прапорами в першій програмі) досягає нуля на границі, і відлік таймера в прямому порядку (калібрований прапорами в другій програмі) відображає те ж положення границі.

Другий зверху бітовий потік фіг. 11 (підписаний як "Сценарій 2") вказує всю першу аудіопрограму (P1), що включає метадані границі програми (прапори границі програми, F), за якою йде вся друга аудіопрограма (P2), що не включає метадані границі програми. Прапори границі програми в кінцевій частині першої програми (деякі з яких показані на фіг. 11) такі ж або схожі на ті, які описані з посиланням на фіг. 8, і визначають положення границі між двома програмами (тобто границі на початку другої програми), так само, як у сценарії 1. У типових варіантах здійснення кодер або декодер реалізує таймер (калібрований прапорами в першій програмі), що відраховує у зворотному порядку до границі програми, і той же таймер (далі без калібрування) продовжує відлік у прямому напрямку від границі програми (як показано на графіку таймера границі в сценарії 2 на фіг. 11).

Третій зверху бітовий потік фіг. 11 (підписаний як "Сценарій 3") вказує обрізану першу аудіопрограму (P1), що включає метадані границі програми (прапори границі програми, F), і яка змонтована з усією другою аудіопрограмою (P2), що також включає метадані границі програми (прапори границі програми, F). Монтаж видалив останні "N" фреймів першої програми. Прапори границі програми в початковій частині другої програми (деякі з яких показані на фіг. 11) такі ж або схожі на ті, які описані з посиланням на фіг. 9, і вони визначають положення границі (монтажу) між обрізаною першою програмою й всією другою програмою. У типових варіантах здійснення кодер або декодер реалізує таймер (калібрований прапорами в першій програмі), що відраховує у зворотному порядку до кінця необрізаної програми, і той же таймер (калібрований прапорами в другій програмі) відраховує в прямому порядку від початку другої програми. У сценарії 3 початок другої програми є границею програми. Як показано на графіку таймера границі в сценарії 3 на фіг. 11, такий відлік таймера у зворотному порядку (калібрований метаданими границі програми в першій програмі) скидається (у відповідь на метадані границі програми в другій програмі) перед тим, як він досягне нуля (у відповідь на метадані границі програми в першій програмі). Таким чином, хоча обрізання першої програми (за рахунок монтажу) перешкоджає визначенню таймером границі програми між обрізаною першою програмою й початком другої програми у відповідь на (тобто при калібруванні ними) метадані границі програми тільки в першій програмі, метадані програми в другій програмі скидають таймер, так що таймер зі скиданням коректно вказує (в якості положення, що відповідає "нульовому" відліку таймера зі скиданням) положення границі програми між обрізаною першою програмою й початком другої програми.

Четвертий бітовий потік (підписаний як "Сценарій 4") відображає обрізану першу аудіопрограму (P1), що включає метадані границі програми (прапори границі програми, F), і обрізану другу аудіопрограму (P2), що включає метадані границі програми (прапори границі програми, F) і яка була змонтована із частиною (необрізаною частиною) першої аудіопрограми.

Прапори границі програми в початковій частині всієї (попереднє обрізання) другої програми (деякі з яких показані на фіг. 11) такі ж або схожі на ті, які описані з посиланням на фіг. 9, і прапори границі програми в кінцевій частині всієї (попереднє обрізання) першої програми (деякі з яких показані на фіг. 11) такі ж або схожі на ті, які описані з посиланням на фіг. 8. Монтаж видалив останні "N" фреймів першої програми (і, таким чином, деякі із прапорів границі програми, які перебували там перед монтажем) і перші "M" фреймів другої програми (і, таким чином, деякі із прапорів границі програми, які перебували в них перед монтажем). У типових варіантах здійснення кодер або декодер реалізує таймер (калібрований прапорами в обрізаній першій програмі), що відраховує у зворотному порядку до кінця необрізаної програми, і той же таймер (калібрований прапорами в обрізаній другій програмі) відраховує в прямому порядку від початку необрізаної другої програми. Як показано на графіку таймера границі в сценарії 4 на фіг. 11, такий відлік таймера у зворотному порядку (калібрований метаданими границі програми в першій програмі) скидається (у відповідь на метадані границі програми в другій програмі) перед тим, як він досягне нуля (у відповідь на метадані границі програми в першій програмі). Обрізання першої програми (за рахунок монтажу) запобігає визначенню таймером границі програми між обрізаною першою програмою й початком обрізаної другої програми у відповідь на (тобто при калібрування ними) метадані границі програми тільки в першій програмі. Однак таймер зі скиданням коректно не вказує положення границі програми між кінцем обрізаної першої програми й початком обрізаної другої програми. Таким чином, обрізання обох змонтованих бітових потоків може перешкоджати точному визначенню границі між ними.

Варіанти здійснення даного винаходу можуть бути реалізовані на апаратних, програмних, програмно-апаратному рівнях або їхніх комбінаціях (наприклад, у вигляді програмованої логічної матриці). Якщо не зазначено інше, алгоритми або процеси, включені як частина винаходу, по своїй суті, не пов'язані з будь-яким конкретним комп'ютером або іншим пристроєм. Зокрема, різні машини загального застосування можуть бути використані із програмами, написаними відповідно до описаних в даному документі ідей, або більше зручним може бути створення більш спеціалізованого пристрою (наприклад, інтегральних схем) для здійснення необхідних етапів способу. Таким чином, винахід може бути реалізований в одній або декількох комп'ютерних програмах, виконуваних на одній або декількох програмованих комп'ютерних системах (наприклад, реалізації кожного з елементів, зображених на фіг. 1, або кодера 100, зображеного на фіг. 2, (або його елементи), або декодера 200, зображеного на фіг. 3, (або його елементу), або постпроцесора 300, зображеного на фіг. 3, (або його елемента), кожна з яких містить щонайменше один процесор, щонайменше одну систему зберігання даних (що містить постійний запам'ятовувальний й тимчасовий запам'ятовувальний пристрій і/або запам'ятовувальні елементи), щонайменше один пристрій введення й щонайменше один пристрій виведення або порт. Програмний код застосовується до вхідних даних для здійснення функцій, описаних у даному документі й генерування вихідної інформації. Вихідна інформація застосовується до одного або декількох вихідним пристроїв відомим способом.

Кожна така програма може бути реалізована на будь-якій бажаній мові програмування (включаючи машинну мову, мову асемблера або високорівневу процедурну мову, мову логіки або об'єктно-орієнтовану мову програмування) для взаємодії з комп'ютерною системою. У будь-якому разі мова може являти собою трансльовану або інтерпретовану мову.

Наприклад, при реалізації за допомогою послідовностей інструкцій комп'ютерного програмного забезпечення, різні функції й етапи варіантів здійснення винаходу можуть бути реалізовані за допомогою багатопотокових послідовностей інструкцій програмного забезпечення, виконуваних на придатному апаратному забезпеченні для цифрової обробки сигналів, на випадок чого різні пристрої, етапи й функції варіантів здійснення можуть відповідати частинам програмних інструкцій.

Кожна така комп'ютерна програма переважно зберігається або завантажується в носій або пристрій зберігання даних (наприклад, твердотільну пам'ять або носій, або магнітний, або оптичний носій), що читається програмованим комп'ютером загального або спеціального призначення, для конфігурування й керування комп'ютером, коли носій або пристрій зберігання даних зчитується комп'ютерною системою для здійснення процедур, описаних у даному документі. Система відповідно до винаходу також може бути реалізована як машинопрочитуваний носій даних, сконфігурований (тобто що зберігає) комп'ютерною програмою, де носій даних, сконфігурований таким чином, змушує комп'ютерну систему працювати певним і попередньо заданим чином, щоб здійснювати функції, описані в даному документі.

Була описана деякий кількість варіантів здійснення винаходу. Проте, варто розуміти, що без відступу від суті й обсягу даного винаходу можуть бути виконані різні модифікації. У світлі

вищевикладених ідей можливі численні модифікації й варіанти даного винаходу. Варто розуміти, що в межах обсягу прикладеної формули винаходу, винахід може бути здійснений інакше, ніж конкретно описано тут.

5

ФОРМУЛА ВИНАХОДУ

1. Блок обробки звукового сигналу, який містить:

буферний запам'ятовувальний пристрій, призначений для зберігання щонайменше одного фрейму кодованого бітового аудіопотоку, при цьому кодований бітовий аудіопотік містить аудіодані та контейнер метаданих, причому контейнер метаданих включає заголовок й одне або декілька інформаційних наповнень метаданих після заголовку, причому одне або декілька інформаційних наповнень метаданих містять метадані, що вказують стан обробки гучності аудіоданих, і причому метадані, що вказують стан обробки гучності аудіоданих, являють собою або містять метадані, що вказують щонайменше один тип обробки гучності, яка виконується над аудіоданими;

синтаксичний аналізатор, з'єднаний з буферним запам'ятовувальним пристроєм та призначений для виконання синтаксичного аналізу кодованого бітового аудіопотоку; та підсистему, з'єднану з синтаксичним аналізатором та призначену для адаптивної обробки гучності з використанням щонайменше деяких метаданих, що вказують стан обробки гучності аудіоданих.

2. Блок обробки звукового сигналу за п. 1, який **відрізняється** тим, що метадані, що вказують стан обробки гучності аудіоданих, містять метадані, що вказують щонайменше одну характеристику гучності або динамічного діапазону аудіоданих.

3. Блок обробки звукового сигналу за п. 1, який **відрізняється** тим, що адаптивна обробка гучності являє собою або включає виконання керування динамічним діапазоном.

4. Блок обробки звукового сигналу за п. 1, який **відрізняється** тим, що також містить: аудіодекодер, підключений та призначений для декодування аудіоданих, внаслідок чого генеруються декодовані аудіодані.

5. Блок обробки звукового сигналу за п. 4, який **відрізняється** тим, що з'єднана з синтаксичним аналізатором підсистема також з'єднана з аудіодекодером, причому зазначена підсистема призначена для виконання адаптивної обробки гучності над щонайменше деякими декодованими аудіоданими з використанням щонайменше деяких метаданих, що вказують стан обробки гучності аудіоданих.

6. Блок обробки звукового сигналу за п. 5, який **відрізняється** тим, що адаптивна обробка гучності являє собою або включає виконання керування динамічним діапазоном.

7. Блок обробки звукового сигналу за п. 4, який **відрізняється** тим, що одне або декілька інформаційних наповнень метаданих містять інформаційне наповнення гучності програми, що містить дані, які вказують вимірювану гучність аудіопрограми, пов'язаної з аудіоданими, причому підсистема, з'єднана з синтаксичним аналізатором, також з'єднана з аудіодекодером, і зазначена підсистема призначена для виконання адаптивної обробки гучності над щонайменше деякими аудіоданими з використанням інформаційного наповнення гучності програми.

8. Блок обробки звукового сигналу за п. 1 який **відрізняється** тим, що одне або декілька інформаційних наповнень метаданих містять інформаційне наповнення гучності програми, що містить дані, які вказують вимірювану гучність аудіопрограми, пов'язаної з аудіоданими.

9. Блок обробки звукового сигналу за п. 8, який **відрізняється** тим, що інформаційне наповнення гучності програми містить поле, що вказує спосіб вимірювання гучності, який був використаний для формування даних про гучність, включених в інформаційне наповнення гучності програми.

10. Спосіб обробки звукового сигналу, який включає етапи:

отримання кодованого бітового аудіопотоку, при цьому кодований бітовий аудіопотік сегментований на один або декілька фреймів;

добування аудіоданих та контейнера метаданих із кодованого бітового аудіопотоку, причому контейнер метаданих містить заголовок і одне або декілька інформаційних наповнень метаданих після заголовка, причому одне або декілька інформаційних наповнень метаданих містять метадані, що вказують стан обробки гучності аудіоданих, і причому метадані, що вказують стан обробки гучності аудіоданих, являють собою або містять метадані, що вказують щонайменше один тип обробки гучності, яка виконується над аудіоданими; та

виконання адаптивної обробки гучності з використанням щонайменше деяких метаданих, що вказують стан обробки гучності аудіоданих.

11. Спосіб за п. 10, який **відрізняється** тим, що метадані, що вказують стан обробки гучності аудіоданих, містять метадані, що вказують щонайменше одну характеристику гучності або динамічного діапазону аудіоданих.

12. Спосіб за п. 10, який **відрізняється** тим, що аудіодані являють собою кодовані аудіодані, а також включає етап: декодування кодованих аудіоданих для генерування декодованих аудіоданих.

13. Спосіб за п. 12, який **відрізняється** тим, що етап виконання адаптивної обробки гучності включає виконання адаптивної обробки гучності над щонайменше деякими декодованими аудіоданими з використанням зазначених щонайменше деяких метаданих, що вказують стан обробки гучності аудіоданих.

14. Спосіб за п. 13, який **відрізняється** тим, що адаптивна обробка гучності являє собою або включає виконання керування динамічним діапазоном.

15. Спосіб за п. 10, який **відрізняється** тим, що одне або декілька інформаційних наповнень метаданих містять інформаційне наповнення гучності програми, яке містить дані, що вказують вимірювану гучність аудіопрограми, яка пов'язана з аудіоданими, і при цьому етап виконання адаптивної обробки гучності включає виконання адаптивної обробки гучності над щонайменше деякими аудіоданими, добутими із кодованого бітового аудіопотоку, із використанням інформаційного наповнення гучності програми.

16. Спосіб за п. 10, який **відрізняється** тим, що адаптивна обробка гучності являє собою або включає виконання керування динамічним діапазоном.

17. Спосіб за п. 10, який **відрізняється** тим, що одне або декілька інформаційних наповнень метаданих містять інформаційне наповнення гучності програми, яке містить дані, що вказують вимірювану гучність аудіопрограми, яка пов'язана з аудіоданими, і при цьому інформаційне наповнення гучності програми містить поле, що вказує спосіб вимірювання гучності, який був використаний для формування даних про гучність, включених в інформаційне наповнення гучності програми.

18. Носій даних, на якому зберігається щонайменше один сегмент бітового аудіопотоку, що містить аудіодані та контейнер метаданих, причому контейнер метаданих містить заголовок, і одне або декілька інформаційних наповнень метаданих після заголовка, причому одне або декілька інформаційних наповнень метаданих містять метадані, що вказують стан обробки гучності аудіоданих, і метадані, що вказують стан обробки гучності аудіоданих, являють собою або містять метадані, що вказують щонайменше один тип обробки гучності, яка виконується над аудіоданими.

19. Носій даних за п. 18, який **відрізняється** тим, що метадані, що вказують стан обробки гучності аудіоданих, містять метадані, що вказують щонайменше одну характеристику гучності або динамічного діапазону аудіоданих.

20. Носій даних за п. 18, який **відрізняється** тим, що зазначений носій даних являє собою машинозчитуваний носій даних.



Fig. 1

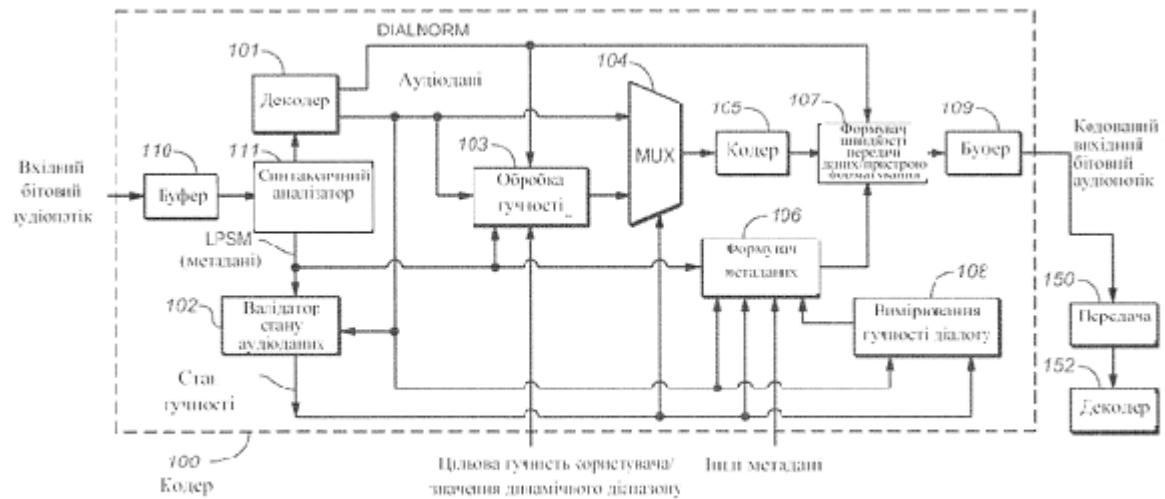


Fig. 2

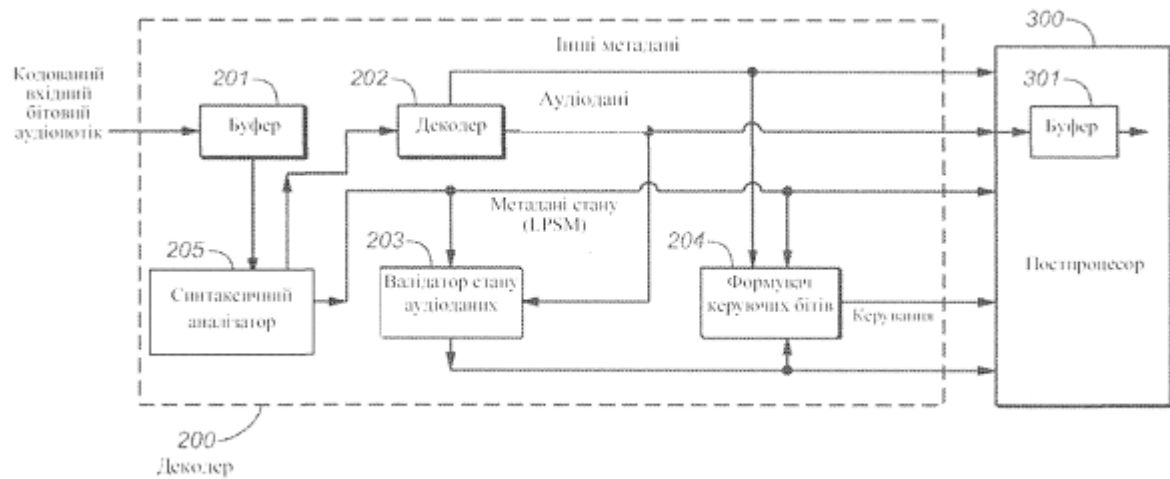


Fig. 3

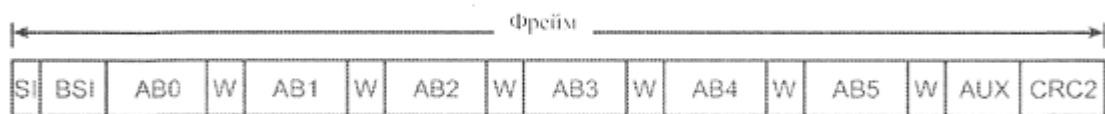


Fig. 4

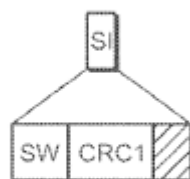


Fig. 5

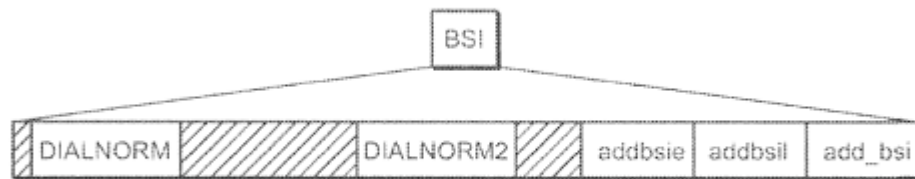


Fig. 6

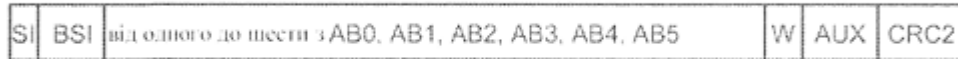


Fig. 7

№ фрейму	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	
Фрейм до границі	-16	-15	-14	-13	-12	-11	-10	-9	-8	-7	-6	-5	-4	-3	-2	-1	B
Зміст прапору границі програми	100011	0	0	0	0	0	0	0	10011	0	0	0	1011	0	111	0	B
																	B
Наявність	1	0	0	0	0	0	0	0	1	0	0	0	1	0	1	0	B
Лічильник фреймів	0001->16								001->8				01->4		1->2		B
Відлік в прямому порядку (+) / зворотному порядку	1								1						1		B

Синтаксис прапору границі програми, що віраховується в зворотному порядку до границі програми

Fig. 8

№ фрейму	B	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Фрейм до границі	B	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Зміст прапору границі програми	B	0	110	0	1010	0	0	0	10010	0	0	0	0	0	0	0	10010
	B																
Наявність	B	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	
Лічильник фреймів	B		1->2		01->4				001->8								0001->16
Відлік в прямому порядку (+) / зворотному порядку	B		0		0				0								0

Синтаксис прапору границі програми, що віраховується в прямому порядку для початку програми

Fig. 9

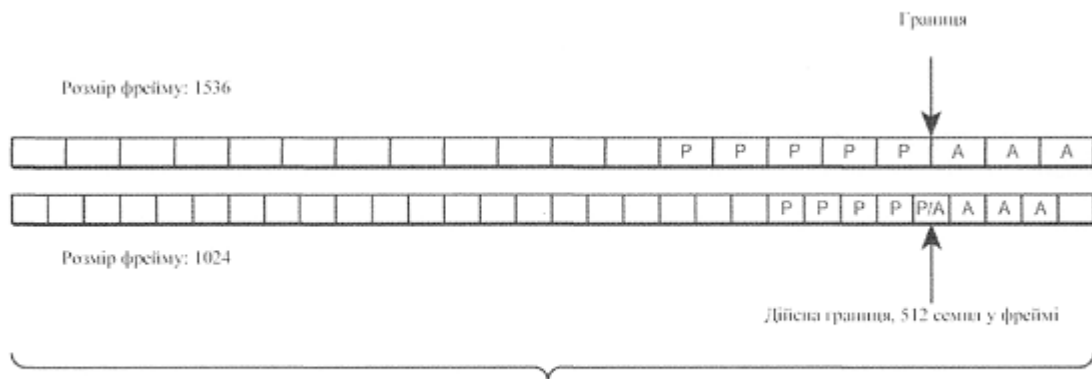
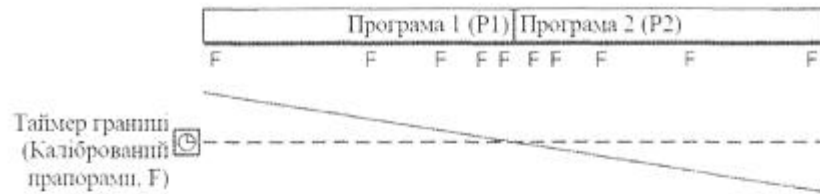
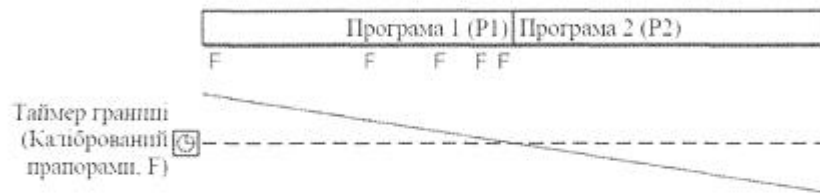


Fig. 10

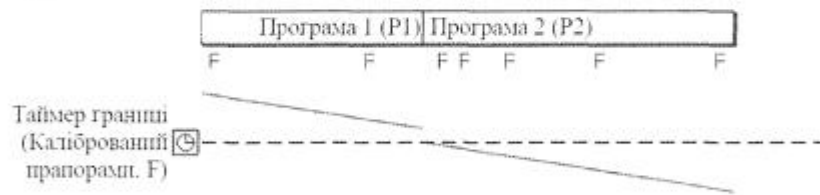
Сценарій 1: і P1, і P2 мають прапори



Сценарій 2: тільки P1 має прапори



Сценарій 3: Кінець P1 змонтований



Сценарій 4: Кінець P1, початок P2 змонтовані



Fig. 11

