

19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 792 116**

51 Int. Cl.:

G10L 19/025 (2013.01)

G10L 19/00 (2013.01)

G10L 19/008 (2013.01)

G10L 19/24 (2013.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

96 Fecha de presentación y número de la solicitud europea: **09.01.2009 E 18193700 (4)**

97 Fecha y número de publicación de la concesión europea: **11.03.2020 EP 3435375**

54 Título: **Códec de audio multicanal sin pérdida que usa segmentación adaptativa con capacidad de conjunto de parámetros de predicción múltiple (MPPS)**

30 Prioridad:

30.01.2008 US 1108908

45 Fecha de publicación y mención en BOPI de la traducción de la patente:

10.11.2020

73 Titular/es:

**DTS, INC. (100.0%)
5220 Las Virgenes Road
Calabasas, CA 91302, US**

72 Inventor/es:

FEJZO, ZORAN

74 Agente/Representante:

VALLEJO LÓPEZ, Juan Pedro

ES 2 792 116 T3

Aviso: En el plazo de nueve meses a contar desde la fecha de publicación en el Boletín Europeo de Patentes, de la mención de concesión de la patente europea, cualquier persona podrá oponerse ante la Oficina Europea de Patentes a la patente concedida. La oposición deberá formularse por escrito y estar motivada; sólo se considerará como formulada una vez que se haya realizado el pago de la tasa de oposición (art. 99.1 del Convenio sobre Concesión de Patentes Europeas).

DESCRIPCIÓN

Códec de audio multicanal sin pérdida que usa segmentación adaptativa con capacidad de conjunto de parámetros de predicción múltiple (MPPS)

5

Remisiones a solicitudes relacionadas

Esta solicitud reivindica beneficio de prioridad según 35 U.S.C. 120 como una continuación parcial (CIP) de Solicitud de Estados Unidos N.º 10/911.067 titulada " Lossless Multi-Channel Audio Codec" presentada el 4 de agosto de 2004.

10

Antecedentes de la invención

Campo de la invención

15

Esta invención se refiere códecs de audio sin pérdida y más específicamente a un códec de audio multicanal sin pérdida usando segmentación adaptativa con capacidad de conjunto de parámetros de predicción múltiple (MPPS).

Descripción de la técnica relacionada

20

Números de sistemas de codificación de audio con pérdida de tasa de bits baja están en uso en la actualidad en una amplia variedad de productos y servicios de reproducción de audio de consumo y profesionales. Por ejemplo, sistema de codificación de audio Dolby AC3 (Dolby digital) es una norma mundial para la codificación de pistas de sonido de audio de canal 5.1 y estéreo para Laser Disc, video DVD codificado en NTSC y ATV, usando tasas de bits de hasta 640 kbit/s. Las normas de codificación de audio MPEG I y MPEG II se usan ampliamente para codificación de pistas de sonido estéreo y multicanal para video DVD codificado en PAL, radiodifusión de radio digital terrestre en Europa y radiodifusión por satélite en los Estados Unidos, en tasas de bits hasta 768 kbit/s. Sistema de codificación de audio de Acústica Coherente de DTS (Sistemas de Cine Digital) se usa frecuentemente para pista de sonido de audio de canal 5.1 de calidad de estudio para Disco Compacto, video DVD, Radiodifusión por Satélite en Europa y Laser Disc y tasas de bits de hasta 1536 kbit/s.

25

30

En la actualidad, muchos consumidores han mostrado interés en estos así llamados códecs "sin pérdida". Códecs "sin pérdida" se basan en algoritmos que comprimen datos sin descartar ninguna información y producen una señal decodificada que es idéntica a la señal fuente (digitalizada). Este rendimiento tiene un coste: tales códecs habitualmente requieren más ancho de banda que códecs con pérdida y comprimen los datos en un menor grado.

35

La Figura 1 es una representación de diagrama de bloques de las operaciones implicadas en la compresión sin pérdida de un único canal de audio. Aunque los canales en audio multicanal generalmente no son independientes, la dependencia a menudo es débil y difícil de tener en cuenta. Por lo tanto, los canales habitualmente se comprimen de forma separada. Sin embargo, algunos codificadores intentarán eliminar la correlación formando una señal residual simple y codificando (Ch1, Ch1-CH2). Enfoques más sofisticados toman, por ejemplo, varias etapas de proyección ortogonales sucesivas en la dimensión de canal. Todas las técnicas se basan en el principio de primero eliminar la redundancia de la señal y a continuación codificar la señal resultante con un esquema de codificación digital eficiente. Códecs sin pérdida incluyen MPL (DVD Audio), Monkey's audio (aplicaciones informáticas), Apple sin pérdida, Windows Media Pro sin pérdida, AudioPak, DVD, LTAC, MUSICcompress, OggSquish, Philips, Shorten, Sonarc y WA. Una evaluación de muchos de estos códecs se proporciona por Mat Hans, Ronald Schafer "Lossless Compression of Digital Audio" Hewlett Packard, 1999.

40

45

La alineación de tramas **10** se introduce para proporcionar capacidad de edición, el enorme volumen de datos prohíbe la descompresión repetitiva de toda la señal que precede la región a editar. La señal de audio se divide en tramas independientes de igual duración. Esta duración no debería ser demasiado corta, ya que puede resultar sobrecarga significativa del encabezamiento que se prefija a cada trama. A la inversa, la trama duración no debería ser demasiado larga, ya que esto limitaría la capacidad de adaptación temporal y haría la edición más difícil. En muchas aplicaciones, el tamaño de trama se restringe por la tasa de bits pico del medio en el que se transfiere el audio, la capacidad de memoria intermedia del decodificador y deseo de tener cada trama que se pueda codificar independientemente.

50

55

La decorrelación intra canal **12** elimina la redundancia decorrelacionando las muestras de audio en cada canal dentro de una trama. La mayoría de los algoritmos eliminan la redundancia mediante algún tipo de modelado predictivo lineal. En este enfoque, se aplica un predictor lineal a las muestras de audio en cada trama resultando en una secuencia de muestras de error de predicción. Un segundo enfoque, menos común, es obtener una representación cuantificada de tasa de bits baja o sin pérdida de la señal y a continuación comprimir sin pérdida la diferencia entre la versión con pérdida y la versión original. La codificación por entropía **14** elimina redundancia del error de la señal residual sin perder ninguna información. Métodos típicos incluyen codificación de Huffman, codificación por longitud de serie y codificación de Rice. El resultado es una señal comprimida que puede ser reconstruida sin pérdida.

60

65

La especificación de DVD existente y la especificación de HD DVD preliminar establecen un límite estricto en el tamaño de una unidad de acceso de datos, que representan una parte del flujo de audio que una vez extraído puede decodificarse totalmente y las muestras de audio reconstruidas enviadas a las memorias intermedias de salida. Lo que esto significa para un flujo sin pérdida es que la cantidad de tiempo que cada unidad de acceso puede representar tiene que ser lo suficientemente pequeño que el peor caso de tasa de bits pico, la carga útil codificada no excede el límite estricto. La duración de tiempo también debe reducirse para tasas de muestreo aumentadas y número aumentado de canales, que aumentan la tasa de bits pico.

Para garantizar compatibilidad, estos codificadores existentes tendrán que establecer la duración de toda una trama para ser lo suficientemente corta para no exceder el límite estricto en un canal de peor caso/frecuencia de muestreo/ configuración de ancho de bit. En la mayoría de las configuraciones, esto se excederá y puede degradar seriamente el rendimiento de compresión. Adicionalmente, este enfoque de peor caso no escala bien con canales adicionales.

El documento US 5 956 674 A describe un codificador de audio de subbanda que emplea filtros de reconstrucción perfectos/no perfectos, codificación de subbanda predictiva/no predictiva, análisis transitorio, y asignación de bits psicoacústica/error medio cuadrado mínimo (mmse) a lo largo del tiempo, frecuencia y los múltiples canales de audio para codificar/decodificar un flujo de datos para generar audio reconstruido de alta fidelidad. El codificador de audio abre la señal de audio multicanal de modo que el tamaño de la trama, es decir, número de bytes, está obligado a permanecer en el intervalo deseado, y formatea los datos codificados para que las subtramas individuales puedan reproducirse a medida que se reciben, reduciendo así la latencia.

El documento US 2004/044534 A1 discute un esquema de compresión de audio sin pérdida que está adaptado para su uso en un esquema unificado de compresión de audio con pérdida y sin pérdida. En la compresión sin pérdidas, la tasa de adaptación de un filtro adaptativo varía según la detección transitoria, tal como aumentar la tasa de adaptación donde se detecta un transitorio. Una compresión sin pérdida multicanal utiliza un filtro adaptativo que procesa muestras de múltiples canales en la codificación predictiva de una muestra actual en un canal actual.

30 Sumario de la invención

La invención proporciona un método para codificar audio multicanal en un flujo de bits de audio de tasa de bits variable sin pérdida con las características de la reivindicación independiente 1, un método para decodificar un flujo de bits de audio multicanal con tasa de bits variable sin pérdida con las características de la reivindicación 10 independiente y un decodificador de audio multicanal para decodificar un flujo de bits de audio multicanal con tasa de bits variable sin pérdida con las características de la reivindicación independiente 17. Las realizaciones preferentes de la invención se identifican en las reivindicaciones dependientes.

Todas las siguientes ocurrencias de la palabra "realización(es)", si se refieren a combinaciones de características diferentes de las definidas por las reivindicaciones, se refieren a ejemplos que se presentaron originalmente pero que no representan realizaciones de la invención reivindicada actualmente; estos ejemplos todavía se muestran solo con fines ilustrativos.

La presente invención proporciona un códec de audio que genera un flujo de datos de variable tasa de bits (VBR) sin pérdida con capacidad de conjunto de parámetros de predicción múltiple (MPPS) particionada para mitigar los efectos transitorios.

Esto se consigue con una técnica de segmentación adaptativa que determina puntos de inicio de segmento para garantizar restricciones de límite en segmentos impuestos por uno o varios transitorios deseados en la trama y selecciona una duración de segmento óptima en cada trama para reducir carga útil de trama codificada sometida a una restricción de carga útil de segmento codificado. En general, las restricciones de límite especifican que un transitorio debe encontrarse dentro de un cierto número de bloques de análisis del inicio de un segmento. En una realización ilustrativa en la que segmentos dentro de una trama son de la misma duración y una potencia de dos de la duración de bloque de análisis, se determina una duración de segmento máxima para garantizar que se cumplen las condiciones deseadas. MPPS son particularmente aplicables para mejorar rendimiento general para duraciones de trama más largas.

Un flujo de bits de audio VBR sin pérdidas se codifica con MPPS particionados para que los transitorios detectados se ubiquen dentro de los primeros bloques de análisis L de un segmento en sus respectivos canales. En cada trama sucesiva se detecta hasta un transitorio por canal por conjunto de canales y su ubicación dentro de la trama. Los parámetros de predicción se determinan para cada partición considerando el (los) punto(s) de inicio del segmento impuestos por el (los) transitorio(s). Las muestras en cada partición se comprimen con el conjunto de parámetros respectivos. La segmentación adaptativa se emplea en las muestras residuales para determinar la duración de un segmento y los parámetros de codificación por entropía para cada segmento para minimizar la carga útil de trama codificada sujeta a las restricciones de inicio de segmento impuestas por el (los) transitorio(s) y las restricciones de carga útil del segmento codificado. Los parámetros transitorios que indican la existencia y ubicación del primer segmento transitorio (por canal) y los datos de navegación se empaquetan en el encabezado. Un decodificador

desempaqueta el encabezado de la trama para extraer los parámetros transitorios y el conjunto adicional de parámetros de predicción. Para cada canal en un conjunto de canales, el decodificador usa el primer conjunto de parámetros de predicción hasta que se encuentra el segmento transitorio y cambia al segundo conjunto para el resto del segmento. Aunque la segmentación de la trama es la misma en todos los canales y en múltiples conjuntos de canales, la ubicación de un transitorio (si lo hay) puede variar entre conjuntos y dentro de los conjuntos. Esta construcción permite que un decodificador cambie los conjuntos de parámetros de predicción en o muy cerca del inicio de los transitorios detectados con una resolución de subtrama. Esto es particularmente útil con duraciones de trama más largas para mejorar la eficiencia general de codificación.

Rendimiento de compresión puede mejorarse adicionalmente formando $M/2$ canales de decorrelación para M canales de audio. El triplete de canales (base, correlacionado, decorrelacionado) proporciona dos posibles combinaciones de pares (base, correlacionado) y (base, decorrelacionado) que pueden considerarse durante la segmentación y optimización de codificación por entropía para mejorar adicionalmente rendimiento de compresión. Los pares de canales pueden especificarse por segmento o por trama. En una realización ilustrativa, el codificador encuadra los datos de audio y a continuación extrae pares de canales ordenados que incluyen un canal base y un canal correlacionado y genera un canal decorrelacionado para formar al menos un triplete (base, correlacionado, decorrelacionado). Si el número de canales es impar, se procesa un canal base extra. Predicción polinomial adaptativa o fija se aplica a cada canal para formar señales residuales. Para cada triplete, el par de canales (base, correlacionado) o (base, decorrelacionado) con la menor carga útil codificada se selecciona. Usando el par de canales seleccionado, puede determinarse un conjunto global de parámetros de codificación para cada segmento por todos los canales. El codificador selecciona el conjunto global o distintos conjuntos de parámetros de codificación basándose en cuál tiene la menor carga útil codificada total (encabezamiento y datos de audio).

En cualquiera de los enfoques, una vez que se han determinado el conjunto óptimo de parámetros de codificación y pares de canales para la división actual (duración de segmento), el codificador calcula la carga útil codificada en cada segmento a través de todos los canales. Suponiendo que se satisfacen las restricciones en inicio de segmento y tamaño de carga útil de segmento máximo para cualquier transitorio detectado, el codificador determina si la carga útil codificada total para toda la trama para la división actual es menor que la óptima total para una división más temprana. Si es verdadero, el conjunto actual de parámetros de codificación y carga útil codificada se almacena y la duración de segmento se aumenta. El algoritmo de segmentación se inicia adecuadamente dividiendo la trama en los tamaños de segmento mínimos iguales al tamaño de bloque de análisis y aumenta la duración de segmento por una potencia de dos en cada etapa. Este proceso se repite hasta que cualquiera de los tamaños de segmento viola la restricción de tamaño máximo o la duración de segmento crece a la duración de segmento máxima. La habilitación de las características de RAP y la existencia de un transitorio detectado dentro de una trama puede provocar que la rutina de segmentación adaptativa elija una duración de segmento más pequeña de lo que sería de otra manera.

Estas y otras características y ventajas de la invención serán evidentes a los expertos en la materia a partir de la siguiente descripción detallada o realizaciones preferentes, tomadas en conjunto con los dibujos adjuntos, en los que:

Breve descripción de los dibujos

La Figura 1, como se describe anteriormente, es un diagrama de bloques para un codificador de audio sin pérdida estándar;

Las Figuras 2a y 2b son diagramas de bloques de un codificador y decodificador de audio sin pérdida, respectivamente, de acuerdo con la presente invención;

La Figura 3 es un diagrama de información de encabezamiento según se relaciona con segmentación y selección de código por entropía;

Las Figuras 4a y 4b son diagramas de bloques del procesamiento de ventana de análisis y procesamiento de ventana de análisis inverso;

La Figura 5 es un diagrama de flujo de decorrelación de canal transversal;

Las Figuras 6a y 6b son diagramas de bloques de análisis y procesamiento de predicción adaptativa y procesamiento de predicción adaptativa inversa;

Las Figuras 7a y 7b son un diagrama de flujo de segmentación óptima y selección de código por entropía;

Las Figuras 8a y 8b son diagramas de flujo de selección de código por entropía para un conjunto de canales;

La Figura 9 es un diagrama de bloques de un núcleo más códec de extensión sin pérdida;

La Figura 10 es un diagrama de una trama de un flujo de bits en el que cada trama incluye un encabezamiento y una pluralidad de segmentos;

Las Figuras 11a y 11b son diagramas de información de encabezamiento adicional relacionada con la especificación de RAP y MPPS;

La Figura 12 es un diagrama de flujo para determinar límites de segmento o una duración de segmento máxima para RAP deseados o transitorios detectados;

La Figura 13 es un diagrama de flujo para determinar MPPS;

La Figura 14 es un diagrama de una trama que ilustra la selección de puntos de inicio de segmento o una duración de segmento máxima;

Las Figuras 15a y 15b son diagramas que ilustran el flujo de bits y decodificación del flujo de bits en un segmento de RAP y un transitorio; y

La Figura 16 es un diagrama que ilustra segmentación adaptativa basándose en la carga de segmento máxima y restricciones de duración de segmento máxima.

5

Descripción detallada de la invención

La presente invención proporciona un algoritmo de segmentación adaptativa que genera un flujo de datos de variable tasa de bits (VBR) sin pérdida con capacidad de punto de acceso aleatorio (RAP) para iniciar decodificación sin pérdida en un segmento especificado dentro de una trama y/o capacidad de conjunto de parámetros de predicción múltiple (MPPS) dividido para mitigar efectos transitorios. La técnica de segmentación adaptativa determina y fija puntos de inicio de segmento para garantizar que se cumplen condiciones de límite impuestas por RAP deseados y/o transitorios detectados y selecciona una duración de segmento óptima en cada trama para reducir carga útil de trama codificada sometida a una restricción de carga útil de segmento codificado y los puntos de inicio de segmento fijos. En general, las restricciones de límite especifican que un RAP deseado o transitorio debe encontrarse dentro de un cierto número de bloques de análisis del inicio de un segmento. El RAP deseado puede ser más o menos el número de bloques de análisis desde el inicio de segmento. El transitorio se encuentra dentro del primer número de bloques de análisis del segmento. En una realización ilustrativa en la que segmentos dentro de una trama son de la misma duración y una potencia de dos de la duración de bloque de análisis, se determina una duración de segmento máxima para garantizar las condiciones deseadas. RAP y MPPS son particularmente aplicables para mejorar rendimiento general para duraciones de trama más largas.

10

CÓDEC DE AUDIO SIN PÉRDIDA

Como se muestra en las Figuras 2a y 2b, los bloques operacionales esenciales son similares a codificadores y decodificadores sin pérdida existentes con la excepción de modificaciones al procesamiento de ventanas de análisis para establecer condiciones de inicio de segmento para RAP y/o transitorios y la segmentación y selección de código por entropía. Un procesador de ventanas de análisis somete el audio multicanal de PCM **20** a procesamiento de ventana de análisis **22**, que bloquea los datos en tramas de una duración constante, fija puntos de inicio de segmento basándose en RAP deseados y/o transitorios detectados y elimina redundancia decorrelacionando las muestras de audio en cada canal dentro de una trama. Se realiza decorrelación usando predicción, que se define ampliamente para ser cualquier proceso que usa muestras de audio reconstruidas antiguas (un historial de predicción) para estimar un valor para una muestra original actual y determinar un residual. Técnicas de predicción abarcan fija o adaptativa y lineal o no lineal entre otras. En lugar de codificar por entropía las señales residuales directamente, un segmento adaptativo realiza una segmentación óptima y proceso de selección de código por entropía **24** que segmenta los datos en una pluralidad de segmentos y determina la duración de segmento y parámetros de codificación, por ejemplo, la selección de un codificador por entropía particular y sus parámetros, para cada segmento que minimiza la carga útil codificada para toda la trama sometida a la restricción de que cada segmento debe ser decodificable totalmente y sin pérdida, menor que un número máximo de bytes menor que el tamaño de trama, menor que la duración de trama, y que cualquier RAP deseado y/o transitorio detectado debe encontrarse dentro de un número especificado de bloques de análisis (resolución de subtrama) desde el inicio de un segmento. Los conjuntos de parámetros de codificación se optimizan para cada canal distinto y pueden optimizarse para un conjunto global de parámetros de codificación. Un codificador por entropía codifica por entropía **26** cada segmento de acuerdo con su conjunto de particular de parámetros de codificación. Un empaquetador empaqueta **28** datos codificados e información de encabezamiento en un flujo de bits **30**.

15

20

25

30

35

40

45

50

55

60

65

Como se muestra en la Figura 2b, para realizar la operación de decodificación, el decodificador navega a un punto en el flujo de bits **30** en respuesta a, por ejemplo, selección de usuario de una escena de video o capítulo o navegación de usuario, y un desempaquetador desempaqueta el flujo de bits **40** para extraer la información de encabezamiento y datos codificados. El decodificador desempaqueta información de encabezamiento para determinar el siguiente segmento de RAP en el que puede empezar la decodificación. El decodificador que navega al segmento de RAP e comienza la decodificación. El decodificador deshabilita predicción para un cierto número de muestras a medida que encuentra cada segmento de RAP. Si el decodificador detecta la presencia de un transitorio en una trama, el decodificador usa un primer conjunto de parámetros de predicción para decodificar una primera división y a continuación usa un segundo conjunto de parámetros de predicción para decodificar desde el transitorio hacia delante dentro de la trama. Un decodificador por entropía realiza una decodificación por entropía **42** en cada segmento de cada canal de acuerdo con los parámetros de codificación asignados para reconstruir sin pérdida las señales residuales. Un procesador de ventanas de análisis inverso somete estas señales a procesamiento de ventana de análisis inverso **44**, que realiza predicción inversa para reconstruir sin pérdida el audio original de PCM **20**.

NAVEGACIÓN DE FLUJO DE BITS Y FORMATO DE ENCABEZAMIENTO

Como se muestra en la Figura 10, una trama **500** en flujo de bits **30** incluye un encabezamiento **502** y una pluralidad de segmentos **504**. El encabezamiento **502** incluye un sincronizador **506**, un encabezamiento común **508**, un sub-encabezamiento **510** para el uno o más conjuntos de canales y datos de navegación **512**. En esta realización, los

70

datos de navegación **512** incluyen una porción de NAVI **514** y código de corrección de error CRC16 **516**. La porción de NAVI preferentemente descompone los datos de navegación en las más pequeñas porciones del flujo de bits para habilitar navegación total. La porción incluye segmentos de NAVI **518** para cada segmento y cada segmento de NAVI incluye un tamaño de carga útil de NAVI Ch Set **520** para cada conjunto de canales. Entre otras cosas, esto permite que el decodificador navegue al comienzo del segmento de RAP para cualquier especificado conjunto de canales. Cada segmento **504** incluye los residuales codificados por entropía **522** (y muestras originales en las que predicción deshabilitada para RAP) para cada canal en cada conjunto de canales.

El flujo de bits incluye información de encabezamiento y datos codificados para al menos uno y preferentemente múltiples diferentes conjuntos de canales. Por ejemplo, un primer conjunto de canales puede ser una configuración 2.0, un segundo conjunto de canales puede ser unos adicionales 4 canales que constituyen a presentación de canal 5.1, y un tercer conjunto de canales puede ser unos adicionales 2 canales de envolventes que constituyen presentación de canal 7.1 general. Un decodificador de 8 canales extraerá y decodificará los 3 conjuntos de canales que producen una presentación de canal 7.1 en sus salidas. Un decodificador de 6 canales extraerá y decodificará el conjunto de canales 1 y conjunto de canales 2 ignorando completamente el conjunto de canales 3 que produce la presentación de canal 5.1. Un decodificador de 2 canales extraerá y decodificará únicamente el conjunto de canales 1 e ignorará los conjuntos de canales 2 y 3 que producen una presentación de 2 canales. Tener el flujo estructurado de esta manera permite la escalabilidad de la complejidad de decodificador.

Durante la codificación, un codificador de tiempo realiza una así llamada "mezcla descendente embebida" de tal forma que la mezcla descendente 7.1->5.1 está fácilmente disponible en canales 5.1 que se codifican en conjuntos de canales 1 y 2. De forma similar una mezcla descendente 5.1->2.0 está fácilmente disponible en canales 2.0 que se codifican como un conjunto de canales 1. Un decodificador de 6 canales decodificando los conjuntos de canales 1 y 2 obtendrá mezcla descendente 5.1 después de deshacer la operación de embebido de mezcla descendente 5.1->2.0 realizada en el lado de codificación. De forma similar un decodificador de 8 canales totales obtendrá presentación de 7.1 original decodificando conjuntos de canales 1, 2 y 3 y deshaciendo la operación de embebido de mezcla descendente 7.1->5.1 y 5.1->2.0 realizada en el lado de codificación.

Como se muestra en la Figura 3, el encabezamiento **32** incluye información adicional más allá de lo que se proporciona ordinariamente para un códec sin pérdida para implementar la segmentación y selección de código por entropía. Más específicamente, el encabezamiento incluye información de encabezamiento común **34** tal como el número de segmentos (NumSegments) y el número de muestras en cada segmento (NumSamplesInSegm), información de encabezamiento de conjunto de canales **36** tal como los coeficientes de decorrelación cuantificados (QuantChDecorrCoeff[]) e información de encabezamiento de segmento **38** tal como el número de bytes en segmento actual para el conjunto de canales (ChSetByteCOns), una bandera de optimización global (AllChSameParamFlag) y banderas de codificador por entropía (RiceCodeFlag[], CodeParam[]) que indican si se usa codificación de Rice o Binaria y el parámetro de codificación. Esta configuración de encabezamiento particular supone segmentos de igual duración dentro de una trama y segmentos que son una potencia de dos de la duración de bloque de análisis. Segmentación de la trama es uniforme a través de canales dentro de un conjunto de canales y a través de conjuntos de canales.

Como se muestra en la Figura 11a, el encabezamiento incluye adicionalmente parámetros de RAP **530** en el encabezamiento común que especifican la existencia y ubicación de un RAP dentro de una trama dada. En esta realización, el encabezamiento incluye una bandera de RAP = VERDADERO si RAP está presente. El ID de RAP especifica el número de segmento del segmento de RAP para iniciar decodificación cuando se accede al flujo de bits en el RAP deseado. Como alternativa, podría usarse una RAP_MASK para indicar segmentos que son y no son un RAP. El RAP será consistente a través de todos los conjuntos de canales.

Como se muestra en la Figura 11b, el encabezamiento incluye AdPredOrder[0][ch] = orden del predictor adaptativo o FixedPredOrder[0][ch] = orden del predictor fijo para el canal ch o bien en toda la trama o bien en caso de transitorio una primera división de la trama antes de un transitorio. Cuando se selecciona predicción adaptativa (AdPredOrder[0][ch]>0) se codifican coeficientes de predicción adaptativa y empaquetan en AdPredCodes[0][ch][AdPredOrder[0][ch]].

En caso de MPPS el encabezamiento incluye adicionalmente parámetros transitorios **532** en el conjunto de canales información de encabezamiento. En esta realización, cada conjunto de canales encabezamiento incluye una bandera ExtraPredSetsPrsent[ch] = VERDADERO si transitorio se detecta en el canal ch, StartSegment[ch] = índice que indica el segmento de inicio de transitorio para el canal ch y AdPredOrder[1][ch] = orden del predictor adaptativo o FixedPredOrder [1][ch] = orden del predictor fijo para el canal ch aplicable a segunda división en la trama posterior y que incluye un transitorio. Cuando se selecciona predicción adaptativa (AdPredOrder[1][ch]>0) se codifica un segundo conjunto de coeficientes de predicción adaptativa y empaqueta en AdPredCodes[1][ch][AdPredOrder[1][ch]]. La existencia y ubicación de un transitorio puede variar a través de los canales dentro de un conjunto de canales y a través de conjuntos de canales.

65 PROCESAMIENTO DE VENTANAS DE ANÁLISIS

Como se muestra en las Figuras 4a y 4b, una realización ilustrativa de procesamiento de ventanas de análisis **22** selecciona o bien predicción adaptativa **46** o bien predicción polinomial fija **48** para decorrelacionar cada canal, que es un enfoque bastante común. Como se describirá en detalle con referencia a la Figura 6a, se estima un orden de predictor óptimo para cada canal. Si el orden es mayor que cero, se aplica predicción adaptativa. De otra manera se usa la predicción polinomial fija más simple. De manera similar, en el decodificador el procesamiento de ventanas de análisis inverso **44** selecciona o bien predicción adaptativa inversa **50** o bien predicción polinomial fija inversa **52** para reconstruir audio de PCM a partir de las señales residuales. Los órdenes de predictor adaptativo e índices de coeficiente de predicción adaptativa y órdenes de predictor fijo se empaquetan **53** en el conjunto de canales información de encabezamiento.

Decorrelación de canal transversal

De acuerdo con la presente invención, el rendimiento de compresión puede mejorarse adicionalmente implementando decorrelación de canal transversal **54**, que ordena los M canales de entrada en pares de canales de acuerdo con una medida de correlación entre los canales (una diferente "M" que la M restricción de bloque de análisis en un punto RAP deseado). Uno de los canales se designa como el canal "base" y el otro se designa como el canal "correlacionado". Se genera un canal decorrelacionado para cada par de canales para formar un "tripleto" (base, correlacionado, decorrelacionado). La formación del tripleto proporciona dos posibles combinaciones de pares (base, correlacionado) y (base, decorrelacionado) que pueden considerarse durante la segmentación y optimización de codificación por entropía para mejorar adicionalmente rendimiento de compresión (véase la Figura 8a).

La decisión entre (base, correlacionado) y (base, decorrelacionado) puede realizarse o bien antes de (basándose en alguna medida de energía) o bien integrarse con la segmentación adaptativa. El primer enfoque reduce la complejidad mientras el segundo aumenta la eficiencia. Puede usarse un enfoque 'híbrido' donde para tripletes que tienen un canal decorrelacionado con varianza considerablemente más pequeña (basándose en un umbral) que el canal correlacionado se usa una sustitución simple del canal correlacionado por el canal decorrelacionado antes de segmentación adaptativa mientras para todos los demás tripletes la decisión acerca de canal de codificación correlacionada o decorrelacionada se deja al proceso de segmentación adaptativa. Esto simplifica algo la complejidad del proceso de segmentación adaptativa sin sacrificar la eficiencia de codificación.

El M-ch PCM **20** original y el M/2-ch PCM decorrelacionado **56** se reenvían ambos a las operaciones de predicción adaptativa y predicción polinomial fija, que generan señales residuales para cada uno de los canales. Como se muestra en la Figura 3, índices (OrigChOrder[]) que indican el orden original de los canales antes de la clasificación realizada durante el proceso de decorrelación de pares y una bandera PWChDecorrFlag[] para cada par de canales que indica la presencia de un código para coeficientes de decorrelación cuantificados se almacenan en el conjunto de canales encabezamiento **36** en la Figura 3.

Como se muestra en la Figura 4b, para realizar la operación de decodificación de procesamiento de ventana de análisis inverso **44** la información de encabezamiento se desempaqueta **58** y los residuales (muestras originales en el inicio de segmento de RAP) se pasan a través de o bien predicción polinomial fija inversa **52** o bien predicción adaptativa inversa **50** de acuerdo con la información de encabezamiento, a saber los órdenes de predictor adaptativo y fijo para cada canal. En la presencia de un transitorio en a canal, el conjunto de canales tendrá dos conjuntos diferentes de parámetros predichos para ese canal. El audio de PCM decorrelacionada de canal M (M/2 canales se descartan durante segmentación) se pasa a través de decorrelación inversa de canal transversal **60**; que lee los índices OrigChOrder[] y bandera PWChDecorrFlagg [] del conjunto de canales encabezamiento y reconstruye sin pérdida el audio de PCM del canal M **20**.

En la Figura se ilustra 5 un proceso ilustrativo para realizar decorrelación de canal transversal **54**. A modo de ejemplo, el audio de PCM se proporciona como M=6 canales distintos, L, R, C, Ls, Rs y LFE, que también directamente corresponde a una configuración de conjunto de canales almacenada en la trama. Otros conjuntos de canales pueden ser, por ejemplo, izquierda de envolvente central trasero y derecho de envolvente central trasero para producir audio envolvente 7.1. El proceso comienza iniciando un bucle de trama e iniciando un bucle de conjunto de canales (etapa **70**). Se calculan la estimada de autocorrelación de retardo cero para cada canal (etapa **72**) y la estimada de correlación transversal de retardo cero para todas las posibles combinaciones de pares de canales en el conjunto de canales (etapa **74**). A continuación, se estiman coeficientes de correlación de par de canales CORCOEF como la estimada de correlación transversal de retardo cero dividida por el producto de la estimadas de autocorrelación de retardo cero para los canales implicados en el par (etapa **76**). Los CORCOEF se clasifican desde el mayor valor absoluto al menor y se almacenan en la tabla (etapa **78**). Comenzando desde la parte superior de la tabla, se extraen correspondientes índices de pares de canales hasta que todos los pares se han configurado (etapa **80**). Por ejemplo, los 6 canales pueden emparejarse basándose en sus CORCOEF como (L,R), (Ls,Rs) y (C, LFE).

El proceso comienza un bucle de par de canales (etapa **82**) y selecciona un canal "base" como el que tiene la menor estimada de autocorrelación de retardo cero, que es indicativa de una menor energía (etapa **84**). En este ejemplo, los canales L, Ls y C forman los canales base. El coeficiente de decorrelación de par de canales (ChPairDecorrCoeff) se calcula como la estimada de correlación transversal de retardo cero dividida por estimada de

autocorrelación de retardo cero del canal base (etapa **86**). El canal decorrelacionado se genera multiplicando las muestras de canal base con el $ChPairDecorrCoeff$ y sustrayendo ese resultado de las correspondientes muestras del canal correlacionado (etapa **88**). Los pares de canales y sus canales decorrelacionados asociados definen "tripletes" $(L,R,R-ChPairDecorrCoeff[1]*L)$, $(Ls,Rs,Rs-ChPairDecorrCoeff[2]*Ls)$, $(C,LFE,LFE-ChPairDecorrCoeff[3]*C)$ (etapa **89**). El $ChPairDecorrCoeff[]$ para cada par de canales (y cada conjunto de canales) y los índices de canal que definen la configuración de pares se almacenan en el conjunto de canales información de encabezamiento (etapa **90**). Este proceso se repite para cada conjunto de canales en una trama y a continuación para cada trama en el audio de PCM en ventana (etapa **92**).

10 Determinar punto de inicio de segmento para RAP y transitorios

En las Figuras 12 a 14 se ilustra un enfoque ilustrativo para determinar inicio de segmento y restricciones de duración para acomodar RAP deseados y/o transitorios detectados. El bloque mínimo de datos de audio que se procesa se denomina como un "bloque de análisis". Bloques de análisis son únicamente visibles en el codificador, el decodificador únicamente procesa segmentos. Por ejemplo, un bloque de análisis puede representar 0,5 ms de datos de audio en una trama de **32** ms que incluye **64** bloques de análisis. Los segmentos se comprenden de uno o más bloques de análisis. Idealmente, la trama se divide de modo que un RAP deseado o transitorio detectado se encuentra en el primer bloque de análisis del RAP o segmento transitorio. Sin embargo, dependiendo de la ubicación del RAP deseado o transitorio para garantizar esta condición puede formar una sub-segmentación óptima (duraciones de segmento demasiado cortas) que aumenta demasiado la carga útil de trama codificada. Por lo tanto, una compensación es especificar que cualquier RAP deseado debe encontrarse dentro de M bloques de análisis (diferente "M" que los M canales en la rutina de decorrelación de canal) del inicio del segmento de RAP y cualquier transitorio debe encontrarse dentro de los primeros L bloques de análisis que siguen al inicio del segmento transitorio en el correspondiente canal. M y L son menores que el número total de bloques de análisis en la trama y se eligen para garantizar una toleración de alineación deseada para cada condición. Por ejemplo, si una trama incluye **64** bloques de análisis, M y/o L podría ser 1, 2, 4, 8 o 16. Normalmente, alguna potencia de dos menor que el total y habitualmente una pequeña fracción de la misma (no más del 25 %) para proporcionar verdadera resolución de subtrama. Adicionalmente, aunque la duración de segmento puede permitirse que varíe dentro de una trama, hacerlo complica enormemente el algoritmo de segmentación adaptativa y aumenta bits de encabezamiento de sobrecarga con una mejora relativamente pequeña en eficiencia de codificación. Consecuentemente, una realización típica restringe los segmentos para que sean de igual duración dentro de una trama y de una duración igual a una potencia de dos de la duración de bloque de análisis, por ejemplo, duración de segmento = 2^P * bloque de análisis duración donde $P = 0, 1, 2, 4, 8$ etc. En el caso más general, el algoritmo especifica el inicio del RAP o segmentos transitorios. En el caso restringido, el algoritmo especifica una duración de segmento máxima para cada trama que asegura que se cumplen las condiciones.

Como se muestra en la Figura 12, se proporciona un código de sincronización de codificación que incluye RAP deseados tal como un código de sincronización de video que especifica comienzos de capítulo o escena mediante la capa de aplicación (etapa **600**). Se proporcionan tolerancias de alineación que dictan los valores máximos de M y L anteriores (etapa **602**). Las tramas se bloquean en una pluralidad de bloques de análisis y sincronizan con el código de sincronización para alinear RAP deseados a bloques de análisis (etapa **603**). Si un RAP deseado se encuentra dentro de la trama, el codificador fija el inicio de un segmento de RAP en el que el bloque de análisis de RAP debe encontrarse dentro de M bloques de análisis antes o después del inicio del segmento de RAP (etapa **604**). Obsérvese, el RAP deseado puede encontrarse en realidad en el segmento que precede a el segmento de RAP dentro de M bloques de análisis del inicio del segmento de RAP. El enfoque comienza el análisis de predicción adaptativa/fija (etapa **605**), comienza el bucle de conjunto de canales (etapa **606**) e comienza el análisis de predicción adaptativa/fija en el conjunto de canales (etapa **608**) llamando a la rutina ilustrada en la Figura 13. El bucle de conjunto de canales finaliza (etapa **610**) con la rutina devolviendo el conjunto de parámetros de predicción ($AdPredOrder[0] []$, $FixedPredOrder[0] []$ y $AdPredCodes[0] []$) para el caso cuando $ExtraPredSetsPresent[] = \text{FALSO}$ o dos conjuntos de parámetros predichos ($AdPredOrder[0] []$, $FixedPredOrder[0] []$, $AdPredCodes[0] []$, $AdPredOrder[1] []$, $FixedPredOrder[1] []$ y $AdPredCodes[1] []$) para el caso cuando $ExtraPredSetsPresent[] = \text{VERDADERO}$, los residuales y la ubicación de cualquier transitorio detectado ($StartSegment[]$) por canal (etapa **612**). La etapa **608** se repite para cada conjunto de canales que se codifica en el flujo de bits. Puntos de inicio de segmento para cada trama se determinan a partir del punto de inicio de segmento de RAP y/o puntos de inicio de segmento de transitorio detectados y se pasan al algoritmo de segmentación adaptativa de las Figuras 16 y 7a-7b (etapa **614**). Si las duraciones de segmento se restringen para ser uniformes y una potencia de dos de la longitud de bloque de análisis, se selecciona una duración de segmento máxima basándose en los puntos de inicio fijo y pasa al algoritmo de segmentación adaptativa (etapa **616**). La duración de segmento máxima restricción mantiene los puntos de inicio fijo más añadiendo una restricción en la duración.

Se proporciona una realización ilustrativa del inicio de análisis de predicción adaptativa/fija en una rutina de conjunto de canales (etapa **608**) en la Figura 13. La rutina comienza bucle de canal indexado por ch (etapa **700**), calcula coeficientes de predicción basados en tramas y coeficientes de predicción basados en división (si un transitorio se detecta) y selecciona el enfoque con la mejor eficiencia de codificación por canal. Es posible que incluso si se detecta un transitorio, la codificación más eficiente es ignorar el transitorio. La rutina devuelve los conjuntos de parámetros de predicción, residuales y la ubicación de cualquier transitorio codificado.

Más específicamente, la rutina realiza un análisis de predicción basado en trama llamando a la rutina de predicción adaptativa diagramada en la Figura 6a (etapa **702**) para seleccionar un conjunto de parámetros de predicción basados en trama (etapa **704**). Este único conjunto de parámetros se usa a continuación para realizar predicción en la trama de muestras de audio considerando el inicio de cualquier segmento de RAP en la trama (etapa **706**). Más específicamente, se deshabilita la predicción en el inicio del segmento de RAP para las primeras muestras hasta el orden de la predicción. Una medida de la norma residual basada en trama, por ejemplo la energía residual, se estima a partir de los valores residuales y las muestras originales en las que se deshabilita la predicción.

En paralelo, la rutina detecta si existe cualquier transitorio en la señal original para cada canal dentro de la trama actual (etapa **708**). Un umbral se usa para equilibrar entre detección falsa y detección perdida. Los índices del bloque de análisis que contienen un transitorio se registran. Si se detecta un transitorio, la rutina fija el punto de inicio de un segmento transitorio que se coloca para garantizar que el transitorio se encuentra dentro de los primeros L bloques de análisis del segmento (etapa **709**) y divide la trama en primera y segunda divisiones con la segunda división coincidente con el inicio del segmento transitorio (etapa **710**). La rutina a continuación llama a la rutina de predicción adaptativa diagramada en la Figura 6a (etapa **712**) dos veces para seleccionar primer y segundo conjuntos de parámetros de predicción basados en división para la primera y segunda divisiones (etapa **714**). Los dos conjuntos de parámetros se usan a continuación para realizar predicción en la primera y segunda divisiones de muestras de audio, respectivamente, también considerando el inicio de cualquier segmento de RAP en la trama (etapa **716**). Una medida de la norma residual basada en división (por ejemplo energía residual) se estima a partir de los valores residuales y las muestras originales en las que se deshabilita la predicción.

La rutina compara la norma residual basada en trama a la norma residual basada en división multiplicada por un umbral para tener en cuenta la información de encabezamiento aumentada requerida para múltiples divisiones para cada canal (etapa **716**). Si la energía residual basada en trama es menor, entonces los residuales basados trama y parámetros de predicción se devuelven (etapa **718**) de otra manera los residuales basados división, dos conjuntos de parámetros de predicciones y los índices de los transitorios registrados se devuelven para ese canal (etapa **720**). El bucle de canal indexado por canal (etapa **722**) y análisis de predicción adaptativa/fija en un conjunto de canales (etapa **724**) iteran en los canales en un conjunto y todos los conjuntos de canales antes de finalizar.

La determinación de los puntos de inicio de segmento o duración de segmento máxima para una única trama **800** se ilustra en la Figura 14. Se supone que la trama **800** es 32 ms y contiene 64 bloques de análisis **802** de 0,5 ms cada uno en duración. Un código de sincronización de video **804** especifica un RAP deseado **806** que se encuentra dentro del 9º bloque de análisis. Transitorios **808** y **810** se detectan en CH 1 y 2 que se encuentran dentro del 5º y 18º bloques de análisis respectivamente. En el caso no restringido, la rutina puede especificar puntos de inicio de segmento en bloques de análisis 5, 9 y 18 para garantizar que el RAP y transitorios se encuentran en el 1º bloque de análisis de sus respectivos segmentos. El algoritmo de segmentación adaptativa podría dividir adicionalmente la trama para cumplir otras restricciones y minimizar carga útil de trama siempre que estos puntos de inicio se mantienen. El algoritmo de segmentación adaptativa puede alterar los límites de segmento y aún cumplir con la condición de que el RAP deseado o transitorio se encuentra dentro de un número especificado de bloques de análisis para cumplir con otras restricciones u optimizar mejor la carga útil.

En el caso restringido, la rutina determina una duración de segmento máxima que, en este ejemplo, satisface las condiciones en cada del RAP deseado y los dos transitorios. Ya que el RAP deseado **806** se encuentra dentro del 9º bloque de análisis, la duración máxima de segmento que asegura el RAP se encontraría en el 1º bloque de análisis del segmento de RAP es 8x (escalado por la duración del bloque de análisis). Por lo tanto, los tamaños de segmento permitidos (como un múltiplo de dos del bloque de análisis) son 1, 2, 4 y 8. De manera similar, ya que el transitorio de Ch 1 **808** se encuentra dentro del 5º bloque de análisis la duración de segmento máxima es 4. Transitorio **810** en CH 2 es más problemático en que para garantizar que se produzca en el primer bloque de análisis requiere una duración de segmento igual al bloque de análisis (IX). Sin embargo, si el transitorio puede colocarse en el segundo bloque de análisis entonces la duración máxima de segmento es 16x. Con estas restricciones, la rutina puede seleccionar una duración de segmento máxima de 4 permitiendo de este modo que el algoritmo de segmentación adaptativa seleccione 1x, 2x y 4x para minimizar carga útil de trama y satisfacer las otras restricciones.

En una realización alternativa, el primer segmento de cada n^{ésima} trama puede ser por defecto un segmento de RAP a no ser que el código de sincronización especifique un segmento de RAP diferente en esa trama. El RAP por defecto puede ser útil, por ejemplo, para permitir que un usuario salte o "navegue" dentro del flujo de datos de audio en lugar de restringirse a únicamente esos RAP especificados por el código de sincronización de video.

Predicción adaptativa

Análisis de predicción adaptativa y generación de residuales

Predicción lineal intenta eliminar la correlación entre las muestras de una señal de audio. El principio básico de predicción lineal es predecir un valor de muestra $s(n)$ usando las muestras previas $s(n-1)$, $s(n-2)$, ... y sustraer el

valor predicho $\hat{s}(n)$ de la muestra original $s(n)$. La señal residual resultante $e(n) = s(n) - \hat{s}(n)$ idealmente no estará correlacionada y en consecuencia tendrá un espectro de frecuencia plano. Además, la señal residual tendrá una varianza más pequeña que la señal original implicando que son necesarios menos bits para su representación digital.

5 En una realización ilustrativa del códec de audio, se describe un modelo de predictor FIR mediante la siguiente ecuación:

$$e(n) = s(n) - Q \left\{ \sum_{k=1}^M a_k * s(n-k) \right\}$$

10 en la que $Q\{\}$ indica la operación de cuantificación, M indica el orden de predictor y a_k son coeficientes de predicción cuantificados. Una cuantificación particular $Q\{\}$ es necesaria para compresión sin pérdida ya que la señal original se reconstruye en el lado de decodificación, usando diversas arquitecturas de procesador de precisión finita. La definición de $Q\{\}$ está disponible tanto en el codificador como decodificador y la reconstrucción de la señal original se obtiene simplemente mediante:

$$s(n) = e(n) + Q \left\{ \sum_{k=1}^M a_k * s(n-k) \right\}$$

20 en la que se supone que los mismos a_k coeficientes de predicción cuantificados están disponible para tanto el codificador como decodificador. Un nuevo conjunto de parámetros de predictor se transmite por cada ventana de análisis (trama) permitiendo que el predictor se adapte a la estructura de señal de audio que varía con el tiempo. En el caso de detección de transitorio, se transmiten dos nuevos conjuntos de parámetros predichos para la trama para cada canal en el que se detecta un transitorio; uno para decodificar residuales antes del transitorio y uno para decodificar residuales que incluyen y posteriores al transitorio.

25 Los coeficientes de predicción se diseñan para minimizar predicción residual cuadrática media. La cuantificación $Q\{\}$ hace el predictor un predictor no lineal. Sin embargo en la realización ilustrativa la cuantificación se hace con precisión de 24 bits y es razonable suponer que los efectos no lineales resultantes pueden ignorarse durante la optimización de coeficiente de predictor. Ignorando la cuantificación $Q\{\}$, el problema de optimización subyacente puede representarse como un conjunto de ecuaciones lineales que implican los retardos de secuencia de autocorrelación de señales y los coeficientes de predictor desconocidos. Este conjunto de ecuaciones lineales puede resolverse eficientemente usando el algoritmo de Levinson-Durbin (LD).

30 Los coeficientes de predicción lineal (LPC) resultantes necesitan cuantificarse, de tal forma que pueden transmitirse eficientemente en un flujo codificado. Desafortunadamente la cuantificación directa de LPC no es el enfoque más eficiente ya que los errores de cuantificación pequeños pueden provocar grandes errores espectrales. Una representación alternativa de LPC es la representación de coeficiente de reflexión (RC), que exhibe menos sensibilidad a los errores de cuantificación. Esta representación también puede obtenerse a partir del algoritmo de LD. Mediante la definición del algoritmo de LD los RC se garantizan que tenga una magnitud ≤ 1 (ignorando errores numéricos). Cuando el valor absoluto de los RC está cerca de 1 la sensibilidad de predicción lineal a los errores de cuantificación presentes en RC cuantificados se vuelven alta. La solución es realizar cuantificación no uniforme de RC con etapas de cuantificación más fina alrededor de la unidad. Esto puede conseguirse en dos etapas:

45 1) transformar RC a una representación de relación área-logaritmo (LAR) por medio de función de correlación

$$LAR = \log \frac{1 + RC}{1 - RC}$$

en la que \log indica logaritmo base natural.

50 2) cuantificar uniformemente las LAR

La transformación RC->LAR deforma la escala de amplitud de parámetros de tal forma que el resultado de las etapas 1 y 2 es equivalente a cuantificación no uniforme con etapas de cuantificación más finas alrededor de la unidad.

55 Como se muestra en la Figura 6a, en una realización ilustrativa de análisis de predicción adaptativa se usan parámetros de LAR cuantificados para representar parámetros de predictor adaptativos y transmitidos en el flujo de bits codificado. Se procesan muestras en cada canal de entrada independientes entre sí y en consecuencia la descripción considerará únicamente procesamiento en un único canal.

60 La primera etapa es calcular la secuencia de autocorrelación sobre la duración de ventana de análisis (toda la trama

o divisiones antes o después de un transitorio detectado) (etapa **100**). Para minimizar los efectos de bloqueo que se provocan mediante discontinuidades en la trama los datos de límite primero se basan en ventanas. La secuencia de autocorrelación para un número especificado (igual a orden LP máximo +1) de retardos se estiman a partir de los bloques de datos en ventana.

5 El algoritmo de Levinson-Durbin (LD) se aplica al conjunto de retardos de autocorrelación estimados y se calcula el conjunto de coeficientes de reflexión (RC), hasta el orden LP máximo (etapa **102**). Un resultado intermedio del algoritmo (LD) es un conjunto de varianzas estimadas de residuos de predicción para cada orden de predicción lineal hasta el orden LP máximo. En el siguiente bloque, usando este conjunto de varianzas residuales, se selecciona el orden de predictor lineal (AdPredOrder) (etapa **104**).

10 Para el orden de predictor seleccionado se transforma el conjunto de coeficientes de reflexión (RC), al conjunto de parámetros de relación logaritmo-área (LAR) usando la función de correlación indicada anteriormente (etapa **106**). Una limitación del RC se introduce antes de transformación para evitar la división por 0:

15

$$RC = \begin{cases} Tresh & \forall RC > Tresh \\ -1 & \forall RC < -1 \\ RC & \text{de lo contrario} \end{cases}$$

en la que *Tresh* indica número cercano pero menor de 1. Los parámetros LAR se cuantifican (etapa **108**) de acuerdo con la siguiente regla:

20

$$QLARInd = \begin{cases} \left\lfloor \frac{LAR}{q} \right\rfloor & \forall LAR \geq 0 \\ -\left\lfloor \frac{-LAR}{q} \right\rfloor & \forall LAR < 0 \end{cases}$$

en la que *QLARInd* indica los índices LAR cuantificados, $\lfloor x \rfloor$ indica operación de encontrar el mayor valor entero más pequeño o igual a *x* y *q* indica tamaño de etapa de cuantificación. En la realización ilustrativa, la región [-8 a 8] se

25 codifica usando 8 bits es decir, $q = \frac{2 * 8}{2^8}$ y en consecuencia *QLARInd* se limita de acuerdo con:

$$QLARInd = \begin{cases} 127 & \forall QLARInd > 127 \\ -127 & \forall QLARInd < -127 \\ QLARInd & \text{de lo contrario} \end{cases}$$

P *QLARInd* se traducen desde valores firmados a no firmados usando la siguiente correlación:

30

$$AdPredCodes = \begin{cases} 2 * QLARInd & \forall QLARInd \geq 0 \\ 2 * (-QLARInd) - 1 & \forall QLARInd < 0 \end{cases}$$

En el bloque "RC LUT", se hace una cuantificación inversa de parámetros LAR y una traducción a parámetros RC en una única etapa usando una tabla de correspondencia (etapa **112**). La tabla de correspondencia consiste en valores cuantificados de la correlación de RC -> LAR inversa es decir, correlación de LAR -> RC dada por:

35

$$RC = \frac{e^{LAR} - 1}{e^{LAR} + 1}$$

La tabla de correspondencia se calcula en valores cuantificados de LAR igual a 0, 1,5*q, 2,5*q,... 127,5*q. Los correspondientes valores de RC, después de escalado por 2^{16} , se redondean a enteros no firmados de 16 bits y almacenan como Q16 números de puntos fijos no firmados en una tabla de entrada **128**.

40

Parámetros de RC cuantificados se calculan a partir de la tabla y los índices de LAR de cuantificación *QLARInd* como

45

$$QRC = \begin{cases} TABLE[QLARInd] & \forall QLARInd \geq 0 \\ -TABLE[-QLARInd] & \forall QLARInd < 0 \end{cases}$$

Los parámetros de RC cuantificados QRC_{ord} para $ord = 1$, AdPredOrder se traducen a los parámetros de predicción lineales cuantificados (LP_{ord} para $ord = 1, \dots$ AdPredOrder) de acuerdo con el siguiente algoritmo (etapa 114):

```

For ord = 0 to AdPredOrder - 1 do
  For m = 1 to ord do
     $C_{ord+1,m} = C_{ord,m} + (QRC_{ord+1} * C_{ord,ord+1-m} + (1 \ll 15)) \gg 16$ 
  end
   $C_{ord+1,ord+1} = QRC_{ord+1}$ 
end
For ord = 0 to AdPredOrder - 1 do
   $LP_{ord+1} = C_{AdPredOrder,ord+1}$ 
end

```

5 Ya que los coeficientes RC cuantificados se representaron en formato de puntos fijos firmados Q16 el algoritmo anterior generará los coeficientes de LP también en formato de puntos fijos firmados Q16. La trayectoria de cálculo de decodificador sin pérdida se diseña para soportar hasta resultados intermedios de 24 bits. Por lo tanto es necesario realizar una comprobación de saturación después de que se calcula cada $C_{ord+1,m}$. Si la saturación se produce en cualquier etapa del algoritmo la bandera de saturación se establece y el orden de predictor adaptativo AdPredOrder, para un canal particular, se reinicializa a 0 (etapa 116). Para este canal particular con AdPredOrder=0 se realizará una predicción de coeficiente fijo en lugar de la predicción adaptativa (véase predicción de coeficiente fijo). Obsérvese que los índices de cuantificación de LAR no firmados ($PackLARInd[n]$ para $n=1, \dots$ AdPredOrder [Ch]) se empaquetan en el flujo codificado únicamente para los canales con $AdPredOrder[Ch]>0$.

10 Finalmente para cada canal con $AdPredOrder>0$ la predicción adaptativa lineal se realiza y los residuales de predicción $e(n)$ se calculan de acuerdo con las siguientes ecuaciones (etapa 118):

$$\overline{s(n)} = \left[\left\{ \sum_{k=1}^{AdPredOrder} LP_k * s(n-k) \right\} + (1 \ll 15) \right] \gg 16$$

$$\text{Limitar } \overline{s(n)} \text{ a intervalo de 24 bits } (-2^{23} \text{ a } 2^{23} - 1)$$

$$e(n) = s(n) + \overline{s(n)}$$

$$\text{Limitar } e(n) \text{ a intervalo de 24 bits } (-2^{23} \text{ a } 2^{23} - 1)$$

$$\text{para } n = AdPredOrder + 1, \dots, NumSamples$$

20 Ya que el objetivo de diseño en la realización ilustrativa es que un segmento de RAP específico de ciertas tramas son "puntos de acceso aleatorio", el historial de muestras no se transfiere desde el segmento precedente al segmento de RAP. En su lugar la predicción se acopla únicamente en la muestra AdPredOrder+1 en el segmento de RAP.

25 Los residuales de predicción adaptativa $e(n)$ se codifican por entropía adicionalmente y empaquetan en el flujo de bits codificado.

Predicción adaptativa inversa en el lado de decodificación

30 En el lado de decodificación, la primera etapa en la realización de predicción adaptativa inversa es desempaquetar la información de encabezamiento (etapa 120). Si el decodificador está intentado iniciar decodificación de acuerdo con un código de sincronización de reproducción (por ejemplo selección de usuario de un capítulo o navegación), el decodificador accede al flujo de datos de audio cercano pero antes de ese punto y busca el encabezamiento de la siguiente trama hasta que encuentra una $RAP_Flag = VERDADERO$ que indica la existencia de un segmento de RAP en la trama. El decodificador a continuación extrae el segmento de RAP número (ID de RAP) y datos de navegación (NAVI) para navegar al comienzo del segmento de RAP, deshabilita predicción hasta que índice $> pred_orden$ e comienza decodificación sin pérdida. El decodificador decodifica los segmentos restantes en las tramas y posteriores tramas, deshabilitando predicción cada vez que se encuentra un segmento de RAP. Si se encuentra un $ExtfaPredSetsPrsnt = VERDADERO$ en una trama para un canal, el decodificador extrae el primer y segundo conjuntos de parámetros predichos y el segmento de inicio para el segundo conjunto.

Se extraen los órdenes de predicción adaptativa AdPredOrder[Ch] para cada canal Ch=1, ... NumCh. A continuación para los canales con AdPredOrder[Ch]>0, se extrae la versión no firmada de índices de cuantificación de LAR (AdPredCodes[n] para n=1, AdPredOrder[Ch]). Para cada canal Ch con orden de predicción AdPredOrder[Ch]>0 los AdPredCodes[n] no firmados se correlacionan con los valores firmados QLARInd[n] usando la siguiente correlación:

5

$$QLARInd[n] = \begin{cases} AdPredCodes[n] \gg 1 & \forall AdPredCodes[n] \text{ pares} \\ -(AdPredCodes[n] \gg 1) - 1 & \forall AdPredCodes[n] \text{ impares} \end{cases}$$

para $n = 1, \dots, AdPredOrder[Ch]$

en la que >> indica una operación de desplazamiento derecho de entero.

10 Una cuantificación inversa de parámetros de LAR y una traducción a parámetros de RC se hace en una única etapa usando un Quant RC LUT (etapa 122). Esta es la misma tabla de correspondencia TABLE{} como se define en el lado de codificación. Los coeficientes de reflexión cuantificados para cada canal Ch (QRC[n] para n= 1, ... AdPredOrder [Ch]) se calculan a partir de la TABLE{} y los índices de LAR de cuantificación QLARInd[n], como

$$QRC[n] = \begin{cases} TABLE[QLARInd[n]] & \forall QLARInd[n] \geq 0 \\ -TABLE[-QLARInd[n]] & \forall QLARInd[n] < 0 \end{cases}$$

15

para $n = 1, \dots, PrOr[Ch]$ ³¹

Para cada canal Ch, los parámetros de RC cuantificados QRC_{ord} para ord=1, ... AdPredOrder [Ch] se traducen a los parámetros de predicción lineales cuantificados (LP_{ord} para ord = 1, ... AdPredOrder[Ch]) de acuerdo con el siguiente algoritmo (etapa 124) :

20

```

For ord = 0 to AdPredOrder - 1 do
  For m = 1 to ord do
    Cord+1,m = Cord,m + (QRCord+1 * Cord,ord+1-m + (1 << 15)) >> 16
  end
  Cord+1,ord+1 = QRCord+1
end
For ord = 0 to AdPredOrder - 1 do
  LPord+1 = CAdPredOrder,ord+1
end
    
```

25

Cualquier posibilidad de saturación de resultados intermedios se elimina en el lado de codificación. Por lo tanto en el lado de decodificación no hay necesidad de realizar comprobación de saturación después del cálculo de cada C_{ord+1,m}.

30

Finalmente para cada canal con AdPredOrder[Ch]>0 se realiza una predicción adaptativa inversa lineal (etapa 126). Suponiendo que residuales de predicción e(n) anteriormente se extraen y decodifican por entropía, las señales originales reconstruidas s(n) se calculan de acuerdo con las siguientes ecuaciones:

$$\overline{s(n)} = \left[\left\{ \sum_{k=1}^{AdPredOrder[Ch]} LP_k * s(n-k) \right\} + (1 \ll 15) \right] \gg 16$$

Limitar $\overline{s(n)}$ a intervalo de 24 bits $(-2^{23} \text{ a } 2^{23} - 1)$

$$e(n) = s(n) - \overline{s(n)}$$

para $n = AdPredOrder[Ch]+1, \dots, NumSamples$

Ya que el historial de muestras no se mantiene en un segmento de RAP la predicción adaptativa inversa comenzará desde la muestra (AdPredOrder[Ch]+1) en el segmento de RAP.

35

Predicción de coeficiente fijo

Una forma de coeficiente fijo muy simple del predictor lineal se ha encontrado útil. Los coeficientes de predicción fijos se obtienen de acuerdo con un método de aproximación polinomial muy simple propuesto por primera vez por Shorten (T.. Robinson. SHORTEN: Simple lossless and near lossless waveform compression. Informe Técnico 156. Departamento de Ingeniería de la Universidad de Cambridge, Trumpington Street, Cambridge CB2 1PZ, Reino Unido, diciembre de 1994). En este caso los coeficientes de predicción son los especificados encajando un orden p polinomial a los últimos p puntos de datos. Expandiéndose en cuatro aproximaciones.

$$\hat{s}_0[n] = 0$$

$$\hat{s}_1[n] = s[n-1]$$

$$\hat{s}_2[n] = 2s[n-1] - s[n-2]$$

$$\hat{s}_3[n] = 3s[n-1] - 3s[n-2] + s[n-3]$$

Una propiedad interesante de estas aproximaciones polinomiales es que la señal residual resultante, $e_k[n] = s[n] - \hat{s}_k[n]$ puede implementarse eficientemente de la manera recursiva siguiente.

$$e_0[n] = s[n]$$

$$e_1[n] = e_0[n] - e_0[n-1]$$

$$e_2[n] = e_1[n] - e_1[n-1]$$

$$e_3[n] = e_2[n] - e_2[n-1]$$

...

El análisis de predicción de coeficiente fijo se aplica es una base por trama y no se basa en muestras calculadas en la trama previa ($e_k[-1] = 0$). El conjunto residual con la menor magnitud de suma por toda la trama se define como la mejor aproximación. El orden residual óptimo se calcula para cada canal de forma separada y empaqueta en el flujo como orden de predicción fijo (FPO[Ch]). Los residuales $e_{FPO[Ch]}[n]$ la trama actual se codifican por entropía adicionalmente y empaquetan en el flujo.

El proceso de predicción de coeficiente fijo reverso, en el lado de decodificación, se define mediante una fórmula recursiva de orden para el cálculo de $k^{\text{ésimo}}$ residual de orden en la instancia de muestreo n:

$$e_k[n] = e_{k+1}[n] + e_k[n-1]$$

en la que la señal original deseada $s[n]$ se proporciona mediante

$$s[n] = e_0[n]$$

y en la que para cada $k^{\text{ésimo}}$ residual de orden $e_k[-1] = 0$.

Como un ejemplo recursiones para el 3^{er} orden predicción de coeficiente fijo se presentan en las que los residuales $e_3[n]$ se codifican, transmitidos en el flujo y desempaquetados en el lado de decodificación:

$$e_2[n] = e_3[n] + e_2[n-1]$$

$$e_1[n] = e_2[n] + e_1[n-1]$$

$$e_0[n] = e_1[n] + e_0[n-1]$$

$$s[n] = e_0[n]$$

5 La predicción lineal inversa, adaptativa o fija, realizada en la etapa **126** se ilustra para un caso en el que el m+1 segmento es un segmento de RAP **900** en la Figura 15a y en el que el m+1 segmento es un segmento transitorio **902** en la Figura 15b. Se usa un predictor de 5 derivaciones **904** para reconstruir las muestras de audio sin pérdida. En general, el predictor recombina las 5 muestras reconstruidas sin pérdida previas para generar un valor predicho **906** que se añade al residual actual **908** para reconstruir sin pérdida la muestra actual **910**. en el RAP ejemplo, las 1^a 5 muestras en el flujo de datos de audio comprimido **912** son muestras de audio sin comprimir. En consecuencia, el predictor puede iniciar decodificación sin pérdida en el segmento m+1 sin ningún historial de la muestra previa. En otras palabras, el segmento m+1 es un RAP del flujo de bits. Obsérvese, si también se detectó un transitorio en el segmento m+1 los parámetros de predicción para el segmento m+1 y el resto de la trama diferirá de los usados en los segmentos 1 a m. En el ejemplo de transitorio, todas las muestras en los segmentos m y m+1 son residuales, ninguna RAP. Se ha iniciado decodificación y está disponible un historial de predicción para el predictor. Como se muestra, para reconstruir muestras de audio sin pérdida en los segmentos m y m+1 se usan conjuntos diferentes de parámetros predichos. Para generar la 1^a muestra sin pérdida en el segmento m+1, el predictor usa los parámetros para el segmento m+1 usando las últimas cinco muestras reconstruidas sin pérdida del segmento m. Obsérvese, si el segmento m+1 también fuera un segmento de RAP, las primeras cinco muestras de segmento m+1 serían muestras originales, no residuales. En general, una trama dada puede no contener ni un RAP ni transitorio, de hecho ese es el resultado más típico. Como alternativa, una trama puede incluir un segmento de RAP o un segmento transitorio o incluso ambos. Un segmento puede ser tanto un segmento de RAP como transitorio.

Porque las condiciones de inicio de segmento y duración de segmento máxima se establecen basándose en la ubicación permitida de un RAP deseado o transitorio detectado dentro de un segmento, la selección de la duración de segmento óptima puede generar un flujo de bits en el que el RAP deseado o transitorio detectado realmente se encuentran dentro de segmentos posteriores a los segmentos de RAP o transitorios. Esto podría suceder si los límites M y L son relativamente grandes y la duración de segmento óptima es menor que M y L. El RAP deseado puede encontrarse en realidad en un segmento que precede al segmento de RAP pero que aún está dentro de la tolerancia especificada. Las condiciones en tolerancia de alineación en el lado de codificación aún se mantienen y el decodificador no sabe la diferencia. El decodificador simplemente accede a los segmentos de RAP y transitorios.

SEGMENTACIÓN Y SELECCIÓN DE CÓDIGO POR ENTROPÍA

35 El problema de optimización restringido abordado por el algoritmo de segmentación adaptativa se ilustra en la Figura 16. El problema es codificar uno o más conjuntos de canales de audio multicanal en un flujo de bits VBR de tal manera para minimizar la carga útil de trama codificada sometida a las restricciones de que cada segmento de audio es decodificable totalmente y sin pérdida con carga útil de segmento codificado menor que un número máximo de bytes. El número máximo de bytes es menor que el tamaño de trama y habitualmente se establece por el tamaño máximo de unidad de acceso para leer el flujo de bits. El problema se limita adicionalmente para acomodar acceso aleatorio y transitorios requiriendo que los segmentos se seleccionen de modo que un RAP deseado debe encontrarse a más o menos M bloques de análisis del inicio del segmento de RAP y un transitorio debe encontrarse dentro de los primeros L bloques de análisis de un segmento. La duración de segmento máxima puede restringirse adicionalmente por el tamaño de la memoria intermedia de salida de decodificador. En este ejemplo, los segmentos dentro de una trama se restringen para ser de la misma longitud y una potencia de dos de la duración de bloque de análisis.

Como se muestra en la Figura 16, la duración de segmento óptima para minimizar carga útil de trama codificada **930** equilibra mejoras en ganancia de predicción para un número mayor de segmentos de menor duración contra el coste de bits de sobrecarga adicionales. En este ejemplo, 4 segmentos por trama proporciona una menor carga útil de trama que o bien 2 o bien 8 segmentos. La solución de dos segmentos se descalifica porque la carga útil de segmento para el segundo segmento excede la restricción de carga de segmento máxima **932**. La duración de segmento para ambas dos y cuatro divisiones de segmento excede una duración de segmento máxima **934**, que se establece mediante alguna combinación de, por ejemplo, el tamaño de memoria intermedia de salida de decodificador, ubicación de un punto de inicio de segmento de RAP y/o ubicación de un punto de inicio de segmento transitorio. En consecuencia, el algoritmo de segmentación adaptativa selecciona los 8 segmentos **936** de igual duración y los parámetros de predicción y codificación por entropía optimizados para esa división.

En las Figuras 7a-b y 8a-b se ilustra una realización ilustrativa de segmentación y selección de código por entropía **24** para el caso restringido (segmentos uniformes, potencia de dos de bloque de análisis duración). Para establecer la duración de segmento óptima, parámetros de codificación (selección de código por entropía y parámetros) y pares de canales, los parámetros de codificación y pares de canales se determinan para una pluralidad de diferentes duraciones de segmento hasta la duración de segmento máxima y de entre esos candidatos se selecciona el que tiene la carga útil codificada por trama mínima que satisface las restricciones de que cada segmento debe ser decodificable totalmente y sin pérdida y no excede un tamaño máximo (número de bytes). La segmentación "óptima", parámetros de codificación y pares de canales se someten por supuesto a las restricciones del proceso de

codificación así como la restricción en tamaño de segmento. Por ejemplo, en el proceso ilustrativo, la duración de tiempo de todos los segmentos en la trama es igual, la búsqueda de la duración óptima se realiza en una cuadrícula diádica que comienza con una duración de segmento igual a la duración de bloque de análisis y aumenta por potencias de dos, y el par de canales selección es válido en toda la trama. En el coste de complejidad de codificador adicional y bits de sobrecarga, la duración de tiempo puede permitirse que varíe dentro de una trama, la búsqueda de la duración óptima podría resolverse de forma más fina y el par de canales selección podría hacerse en una base por segmento. En este caso 'restringido', la restricción de que asegura que cualquier RAP deseado o transitorio detectado se alinea al inicio de un segmento dentro de una resolución especificada se incorpora en la duración de segmento máxima.

El proceso ilustrativo comienza inicializando parámetros de segmento (etapa **150**) tal como el número mínimo de muestras en un segmento, en tamaño de carga útil codificada permitido máximo de un segmento, número de segmentos máximo y el número máximo de divisiones y la duración de segmento máxima. Posteriormente, el procesamiento comienza un bucle de división que se indexa desde 0 hasta el número máximo de divisiones menos uno (etapa **152**) e inicializa los parámetros de división que incluyen el número de segmentos, número de muestras en un segmento y el número de bytes consumidos en una división (etapa **154**). En esta realización particular, los segmentos tienen igual duración y el número de segmentos se escala como una potencia de dos con cada iteración de división. El número de segmentos se inicializa preferentemente al máximo, por lo tanto duración de tiempo mínima, que es igual a un bloque de análisis. Sin embargo, el proceso podría usar segmentos de duración de tiempo variable, que podrían proporcionar mejor compresión de datos de audio pero a expensas de sobrecarga adicional y complejidad adicional para satisfacer las condiciones de RAP y transitorio. Adicionalmente, el número de segmentos no tiene que limitarse a potencias de dos o buscarse a partir de mínima a máxima duración. En este caso, los puntos de inicio de segmento determinados por el RAP deseado y transitorios detectados son restricciones adicionales en el algoritmo de segmentación adaptativa.

Una vez que se inicializan, los procesos inician un bucle de conjunto de canales (etapa **156**) y determinan los parámetros de codificación por entropía óptimos y par de canales selección para cada segmento y el correspondiente consumo de bytes (etapa **158**). Los parámetros de codificación PWChDecorrFlag[], AllChSameParamFlag[], RiceCodeFlag[], CodeParam[] y ChSetByteCons[] se almacenan (etapa **160**). Esto se repite para cada conjunto de canales hasta que el bucle de conjunto de canales finaliza (etapa **162**).

El proceso comienza un bucle de segmento (etapa **164**) y calcula el consumo de bytes (SegmByteCons) en cada segmento en todos los conjuntos de canales (etapa **166**) y actualiza el consumo de bytes (ByteConsInPart) (etapa **168**). En este punto, el tamaño del segmento (carga útil de segmento codificado en bytes) se compara al tamaño máximo restricción (etapa **170**). Si se viola la restricción se descarta la división actual. Adicionalmente, porque el proceso comienza con la menor duración de tiempo, una vez que un tamaño de segmento es demasiado grande el bucle de división termina (etapa **172**) y la mejor solución (duración de tiempo, pares de canales, parámetros de codificación) a ese punto se empaqueta en el encabezamiento (etapa **174**) y el proceso se mueve a la siguiente trama. Si la restricción falla en el tamaño de segmento mínimo (etapa **176**), entonces el proceso termina y notifica un error (etapa **178**) porque el tamaño máximo restricción no puede satisfacerse. Suponiendo que la restricción se satisface, este proceso se repite para cada segmento en la división actual hasta que el bucle de segmento finaliza (etapa **180**).

Una vez que el bucle de segmento se ha completado y el consumo de bytes para toda la trama calculado como se representa mediante ByteConsInPart, esta carga útil se compara a la carga útil mínima actual (MinByteInPart) a partir de una iteración de división previa (etapa **182**). Si la división actual representa una mejora entonces la división actual (PartInd) se almacena como la división óptima (OptPartInd) y se actualiza la carga útil mínima (etapa **184**). Estos parámetros y los parámetros de codificación almacenados se almacenan a continuación como la solución total óptima (etapa **186**). Esto se repite hasta que el bucle de división finaliza con la duración de segmento máxima (etapa **172**), en cuyo punto la información de segmentación y los parámetros de codificación se empaquetan en el encabezamiento (etapa **150**) como se muestra en las Figuras 3 y 11a y 11b.

Una realización ilustrativa para determinar los parámetros de codificación óptimos y consumo de bits asociado para un conjunto de canales para una división actual (etapa **158**) se ilustra en las Figuras 8a y 8b. El proceso comienza un bucle de segmento (etapa **190**) y bucle de canal (etapa **192**) en el que los canales para nuestro ejemplo actual son:

Ch1: L,
 Ch2: R
 Ch3: R- ChPairDecorrCoeff[1]*L
 Ch4: Ls
 Ch5: Rs
 Ch6: Rs - ChPairDecorrCoeff[2]*Ls
 Ch7: C
 Ch8: LFE
 Ch9: LFE- ChPairDecorrCoeff[3]*C

El proceso determina el tipo de código por entropía, correspondiente parámetro de codificación y correspondiente consumo de bits para los canales base y correlacionados (etapa **194**). En este ejemplo, el proceso calcula parámetros de codificación óptimos para un código binario y un código de Rice y a continuación selecciona el que tiene menor consumo de bits para el canal y cada segmento (etapa **196**). En general, la optimización puede realizarse para uno, dos o más posibles códigos por entropía. Para los códigos binarios el número de bits se calcula a partir del valor absoluto máximo de todas las muestras en el segmento del canal actual. El parámetro de codificación de Rice se calcula a partir del valor absoluto medio de todas las muestras en el segmento del canal actual. Basándose en la selección, se establece la RiceCodeFlag, se establece la BitCons y se establece la CodeParam o bien al NumBitsBinary o bien al RiceKParam (etapa **198**).

Si el canal actual que se procesa es un canal correlacionado (etapa **200**) entonces la se repite misma optimización para el correspondiente canal decorrelacionado (etapa **202**), se selecciona el mejor código por entropía (etapa **204**) y se establecen los parámetros de codificación (etapa **206**). El proceso se repite hasta que el bucle de canal finaliza (etapa **208**) y el bucle de segmento finaliza (etapa **210**).

En este punto, se han determinado los parámetros de codificación óptimos para cada segmento y para cada canal. Estos parámetros de codificación y cargas útiles podrían devolverse para los pares de canales (base, correlacionado) desde audio de PCM original. Sin embargo, rendimiento de compresión puede mejorarse seleccionando entre los canales (base, correlacionado) y (base, decorrelacionado) en los tripletes.

Para determinar qué pares de canales (base, correlacionado) o (base, no correlacionado) para los tres tripletes, se inicia un bucle de par de canales (etapa **211**) y se calcula la contribución de cada canal correlacionado (Ch2, Ch5 y Ch8) y cada canal decorrelacionado (Ch3, Ch6 y Ch9) al consumo de bits de trama general (etapa **212**). Las contribuciones de consumo de trama para cada canal correlacionado se compara con las contribuciones de consumo de trama para correspondientes canales decorrelacionados, es decir, Ch2 a Ch3, Ch5 a Ch6 y Ch8 a Ch9 (etapa **214**). Si la contribución del canal decorrelacionado es mayor que el canal correlacionado, la PWChDecorrFlag se establece a falso (etapa **216**). De otra manera, el canal correlacionado se sustituye con el canal decorrelacionado (etapa **218**) y PWChDecorrFlag se establece a verdadero y los pares de canales se configuran como (base, decorrelacionado) (etapa **220**).

Basándose en estas comparaciones el algoritmo seleccionará:

1. O bien Ch2 o bien Ch3 como el canal que se emparejará con correspondiente canal base Ch1;
2. O bien Ch5 o bien Ch6 como el canal que se emparejará con correspondiente canal base Ch4; y
3. O bien Ch8 o bien Ch9 como el canal que se emparejará con correspondiente canal base Ch7.

Estas etapas se repiten para todos los pares de canales hasta que el bucle finaliza (etapa **222**).

En este punto, se han determinado los parámetros de codificación óptimos para cada segmento y cada canal distinto y los pares de canales óptimos. Estos parámetros de codificación para cada distinto pares de canales y cargas útiles podrían devolverse al bucle de división. Sin embargo, rendimiento de compresión adicional puede estar disponible calculando un conjunto de parámetros de codificación globales para cada segmento a través de todos los canales. En el mejor caso, la porción de datos codificada de la carga útil será del mismo tamaño que los parámetros de codificación optimizados para cada canal y más probablemente algo mayor. Sin embargo, la reducción en bits de sobrecarga puede más que compensar la eficacia de codificación de los datos.

Usando los mismos pares de canales, el proceso comienza un bucle de segmento (etapa **230**), calcula los consumos de bits (ChSetByteCons[seg]) por segmento para todos los canales usando los distintos conjuntos de parámetros de codificación (etapa **232**) y almacena ChSetByteCons[seg] (etapa **234**). Se determina a continuación un conjunto global de parámetros de codificación (selección de código por entropía y parámetros) para el segmento a través de todos los canales (etapa **236**) usando los mismos cálculos de código binario y código de Rice que antes excepto que a través de todos los canales. Los mejores parámetros se seleccionan y se calcula el consumo de bytes (SegmByteCons) (etapa **238**). El SegmByteCons se compara al CHSetByteCons[seg] (etapa **240**). Si usando parámetros globales no se reduce el consumo de bits, la AllChSamParamFlag[seg] se establece a falso (etapa **242**). De otra manera, la AllChSameParamFlag[seg] se establece a verdadero (etapa **244**) y los parámetros globales de codificación y correspondiente consumo de bits por segmento se guardan (etapa **246**). Este proceso se repite hasta que se alcanza el final del bucle de segmento (etapa **248**). Todo el proceso se repite hasta que el bucle de conjunto de canales termina etapa **250**.

El proceso de codificación se estructura de una forma que puede deshabilitarse diferente funcionalidad mediante el control de unas pocas banderas. Por ejemplo una única bandera controla si el análisis de decorrelación de canal de par tiene que realizarse o no. Otra bandera controla si el análisis de predicción adaptativa (aún otra bandera para predicción fija) tiene que realizarse o no. Además una única bandera controla si la búsqueda de global parámetros en todos los canales tiene que realizarse o no. La segmentación es también controlable estableciendo el número de divisiones y duración de segmento mínima (en la forma más simple puede ser una única división con duración de

segmento predeterminada). Una bandera indica la existencia de un segmento de RAP y otra bandera indica la existencia de un segmento transitorio. En esencia estableciendo unas pocas banderas en el codificador el codificador puede colapsar a alineación de tramas simple y codificación por entropía.

5 CÓDECO DE AUDIO SIN PÉRDIDA COMPATIBLE HACIA ATRÁS

El códec sin pérdida puede usarse como un "codificador de extensión" en combinación con un codificador de núcleo con pérdida. Un flujo de código de núcleo "con pérdida" se empaqueta como un flujo de bits de núcleo y una señal de diferencia codificada sin pérdida se empaqueta como un flujo de bits de extensión separado. Tras decodificación en un decodificador con características sin pérdida extendidas, los flujos con pérdida y sin pérdida se combinan para construir una señal reconstruida sin pérdida. En un decodificador de generación anterior, el flujo sin pérdida se ignora y el flujo "con pérdida" de núcleo se decodifica para proporcionar una señal audio multicanal y de alta calidad con el ancho de banda y característica de relación señal a ruido del flujo de núcleo.

La Figura 9 muestra una vista de nivel de sistema de un codificador sin pérdida compatible hacia atrás **400** para un canal de una señal multicanal. Una señal de audio digitalizada, adecuadamente muestras de audio de PCM de M bits, se proporciona en la entrada **402**. Preferentemente, la señal de audio digitalizada tiene una tasa de muestreo y ancho de banda que excede la de un núcleo codificador con pérdida modificado **404**. En una realización, la tasa de muestreo de la señal de audio digitalizada es 96 kHz (correspondiente a un ancho de banda de 48 kHz para el audio de muestra). También debería entenderse que el audio de entrada puede ser, y preferentemente es, una señal multicanal en la que cada canal se muestrea a 96 kHz. La descripción a continuación se concentrará en el procesamiento de un único canal, pero la extensión a múltiples canales es sencilla. La señal de entrada se duplica en el nodo **406** y trata en ramas paralelas. En una primera rama de la trayectoria de señal, un codificador de banda ancha con pérdida modificado **404** codifica la señal. El codificador de núcleo modificado **404**, que se describe en detalle a continuación, produce un flujo de bits de núcleo codificado **408** que se transporta a un empaquetador o multiplexor **410**. El flujo de bits de núcleo **408** se comunica también a un decodificador de núcleo modificado **412**, que produce como salida una señal de núcleo reconstruida y modificada **414**.

Mientras tanto, la señal de audio digitalizada de entrada **402** en la trayectoria paralela sufre un retardo de compensación **416**, sustancialmente igual al retardo introducido en el flujo de audio reconstruido (mediante codificación modificada y decodificadores modificados), para producir un flujo de audio digitalizado retardado. El flujo de audio **400** se sustrae del flujo de audio digitalizado retardado **414** en el nodo de adición **420**.

El nodo de adición **420** produce una señal de diferencia **422** que representa la señal original y la señal de núcleo reconstruida. Para lograr codificación "sin pérdida" puramente, es necesario codificar y transmitir la señal de diferencia con técnicas de codificación sin pérdida. Por consiguiente, la señal de diferencia **422** se codifica con un codificador sin pérdida **424**, y el flujo de bits de extensión **426** se empaqueta con el flujo de bits de núcleo **408** en el empaquetador **410** para producir un flujo de bits de salida **428**.

Obsérvese que la codificación sin pérdida produce un flujo de bits de extensión **426** que está en una tasa de bits variable, para acomodar las necesidades del codificador sin pérdida. El flujo empaquetado se somete a continuación opcionalmente a capas adicionales de codificación que incluyen codificación de canal, y a continuación transmite o registra. Obsérvese que para propósitos de esta divulgación, el registro puede considerarse como transmisión a través de un canal.

El codificador de núcleo **404** se describe como "modificado" porque en una realización capaz de manejar ancho de banda extendida el codificador de núcleo requeriría modificación. Un banco de filtros de análisis de 64 bandas **430** dentro del codificador descarta la mitad de sus datos de entrada **432** y un codificador de subbanda de núcleo **434** codifica únicamente las menores 32 bandas de frecuencia. Esta información descartada no preocupa a decodificadores heredados que serían incapaces de reconstruir la mitad superior del espectro de señal en cualquier caso. La información restante se codifica de conformidad con el codificador no modificado para formar un flujo de salida de núcleo compatible hacia atrás. Sin embargo, en otra realización que opera en o por debajo de tasa de muestreo de 48 kHz, el codificador de núcleo podría ser una versión sustancialmente no modificada de un núcleo codificador anterior. De manera similar, para operación por encima de la tasa de muestreo de decodificadores heredados, el decodificador de núcleo modificado **412** incluye un decodificador de subbanda de núcleo **436** que decodifica muestras en las 32 subbandas inferiores. El decodificador de núcleo modificado toma las muestras de subbanda de las 32 subbandas inferiores y pondrá a cero a las muestras de subbanda no transmitidas para las 32 bandas superiores **438** y reconstruye las 64 bandas usando un filtro de síntesis QMF de 64 bandas **440**. Para operación en tasa de muestreo convencional (por ejemplo, 48 kHz e inferior) el codificador de núcleo podría ser una versión sustancialmente no modificada de un anterior codificador de núcleo o equivalente. En algunas realizaciones la elección de tasa de muestreo podría hacerse en el momento de codificación y los módulos de codificación y decodificación reconfigurados en ese momento por software como se desee.

Ya que el codificador sin pérdida se está usando para codificar la señal de diferencia, puede verse que un código por entropía simple sería suficiente. Sin embargo, debido a las limitaciones de tasa de bits en los códecs de núcleo con pérdida existentes, aún permanece una cantidad considerable de los bits totales requeridos para proporcionar un

flujo de bits sin pérdida. Adicionalmente, debido a las limitaciones de ancho de banda del códec de núcleo el contenido de información por encima de **24** kHz en la señal de diferencia aún se correlaciona. Por ejemplo muchos de los componentes armónicos que incluyen trompeta, guitarra, triángulo... alcanzan mucho más allá de 30 kHz). Por lo tanto códecs sin pérdida más sofisticados que mejoran el rendimiento de compresión añaden valor. Además,

5 en algunas aplicaciones los flujos de bits de núcleo y de extensión deben aún satisfacer la restricción de que las unidades decodificables no deben exceder un tamaño máximo. El códec sin pérdida de la presente invención proporciona tanto rendimiento de compresión mejorado como flexibilidad mejorada para satisfacer estas restricciones.

10 A modo de ejemplo, 8 canales de audio de PCM de 24 bits a 96 KHz requieren 18,5 Mbps. Compresión sin pérdida puede reducir esto a aproximadamente 9 Mbps. Acústica Coherente decodificaría el núcleo a 1,5 Mbps, dejando una señal de diferencia de 7,5 Mbps. Para tamaño de segmento máximo de 2 kByte, la duración de segmento media es $2048 \cdot 8 / 7500000 = 2,18$ mseg o aproximadamente **209** muestras a 96 kHz. Un tamaño de trama típico para el núcleo con pérdida para satisfacer el tamaño máximo está entre 10 y 20 mseg.

15 A un nivel de sistema, el códec sin pérdida y el códec sin pérdida compatible hacia atrás pueden combinarse para codificar sin pérdida canales de audio extra en un ancho de banda expandido mientras mantiene compatibilidad hacia atrás con códecs con pérdida existentes. Por ejemplo, 8 canales de audio de 96 kHz a 18,5 Mbps pueden codificarse sin pérdida para incluir canales 5.1 de audio de 48 kHz a 1,5 Mbps. El codificador de núcleo más sin

20 pérdida se usarían para codificar los canales 5.1. El codificador sin pérdida se usará para codificar las señales de diferencia en los canales 5.1. Los restantes 2 canales se codifican en un conjunto de canales separado usando el codificador sin pérdida. Ya que todos los conjuntos de canales no necesitan considerarse cuando se intenta optimizar la duración de segmento, todas las herramientas de codificación se usarán de una forma u otra. Un decodificador compatible decodificaría los 8 canales y reconstruiría sin pérdida la señal de audio de 96 kHz 18,5

25 Mbps. Un decodificador más antiguo decodificaría únicamente los canales 5.1 y reconstruiría la 48 kHz 1,5Mbps.

En general, pueden proporcionarse más de un conjunto de canales sin pérdida puro para el propósito de escalar la complejidad del decodificador. Por ejemplo, para una mezcla original 10.2 los conjuntos de canales podrían organizarse de tal forma que:

- 30
- CHSET1 transporta 5.1 (con mezcla descendente de 10.2 a 5.1 embebida) y se codifica usando núcleo+sin pérdida
 - CHSET1 y CHSET2 transportan 7.1 (con descendente de 10.2 a 7.1 embebida) en los que CHSET2 codifica 2 canales usando sin pérdida
- 35
- CHSET1+CHSET2+CHSET3 transportan mezcla de 10.2 discreta total en la que CHSET3 codifica canales 3.1 restantes usando sin pérdida únicamente

Un decodificador que es capaz de decodificación solo 5.1 decodificará únicamente CHSET1 e ignorará todos los demás conjuntos de canales. Un decodificador que es capaz de decodificación solo 7.1 decodificará CHSET1 y CHSET2 e ignorará todos los demás conjuntos de canales...

40 Adicionalmente, el núcleo con pérdida más sin pérdida no se limita a 5.1. Implementaciones actuales soportan hasta 6.1 usando con pérdida (núcleo+XCh) y sin pérdida y pueden soportar un genérico m.n canales organizados en cualquier número de conjuntos de canales. La codificación con pérdida tendrá un núcleo compatible hacia atrás con

45 5.1 y todos los otros canales que se codifican con el códec con pérdida irán en la extensión XXCh. Esto proporciona el codificado sin pérdida general con considerable flexibilidad de diseño para permanecer compatible hacia atrás con decodificadores existentes mientras soportan canales adicionales.

50 Mientras varias realizaciones ilustrativas de la invención se han mostrado y descrito, numerosas variaciones y realizaciones alternativas se ocurrirán a los expertos en la materia. Tales variaciones y realizaciones alternativas se contemplan y pueden hacerse sin alejarse del alcance de la invención como se define en las reivindicaciones adjuntas.

REIVINDICACIONES

1. Un método de codificación de audio multicanal, en un flujo de datos de audio de tasa de bits variable sin pérdida, VBR, que comprende:

bloquear el audio multicanal que incluye al menos un canal establecido en tramas de igual duración, incluyendo cada trama un encabezamiento y una pluralidad de segmentos, teniendo cada uno de dichos segmentos una duración de uno o más bloques de análisis;
para cada trama sucesiva,

detectar la existencia de un transitorio en un bloque de análisis transitorio en la trama para cada canal del conjunto de canales;

particionar la trama de modo que cualquier bloque de análisis transitorio esté ubicado dentro de los primeros bloques de análisis L de un segmento en sus canales correspondientes;

determinar un primer conjunto de parámetros de predicción para segmentos anteriores y no incluidos al bloque de análisis transitorio y un segundo conjunto de parámetros de predicción para segmentos que incluyen y posteriores al bloque de análisis transitorio para cada canal en el conjunto de canales;

comprimir los datos de audio usando el primer y segundo conjuntos de parámetros de predicción en la primera y segunda partición, respectivamente, para generar señales de audio residuales;

determinar una duración de segmento y parámetros de codificación por entropía para cada segmento de las muestras de audio residuales para reducir una carga útil codificada de tamaño variable de la trama sometida a restricciones de que cada segmento deba ser decodificable totalmente y sin pérdida, tenga una duración menor que la duración de trama y tenga una carga útil de segmento codificado menor que un número máximo de bytes menor que el tamaño de trama;

empaquetar la información del encabezado, incluida la duración del segmento, parámetros transitorios que indican la existencia y ubicación del transitorio, parámetros de predicción, parámetros de codificación por entropía y datos de navegación de flujo de bits en el encabezado de trama en el flujo de bits; y
empaquetar los datos de audio comprimidos y codificados por entropía para cada segmento en los segmentos de cuadro en el flujo de bits.

2. El método de la reivindicación 1, que comprende, además, para cada canal en el conjunto de canales:

determinar un tercer conjunto de parámetros de predicción para toda la trama;

comprimir los datos de audio utilizando el tercer conjunto de parámetros de predicción en todo el cuadro para generar señales de audio residuales; y

seleccionar el tercer conjunto o el primer y segundo conjuntos de parámetros de predicción en función de una medida de eficiencia de codificación de sus respectivas señales de audio residuales,
en donde si se selecciona dicho tercer conjunto, inhabilitar la restricción en la duración del segmento con respecto a la ubicación del transitorio dentro de los bloques de análisis L del comienzo de un segmento.

3. El método de la reivindicación 1, que comprende, además:

recibir un código de tiempo que especifica los puntos de acceso aleatorio deseados, RAP, en el flujo de bits de audio;

determinar hasta un bloque de análisis RAP dentro de la trama a partir del código de sincronización;

fijar el inicio de un segmento RAP para que el bloque de análisis RAP se encuentre dentro de los bloques de análisis M del inicio;

considerar el límite del segmento impuesto por el segmento RAP al particionar el marco para determinar el primer y el segundo conjuntos de parámetros de predicción;

deshabilitar dicha predicción para las primeras muestras hasta el orden de predicción después del inicio del segmento RAP para generar muestras de audio originales precedidas y/o seguidas de muestras de audio residuales para dicho primero y segundo, y terceros conjuntos de parámetros de predicción;

determinar la duración del segmento que reduce la carga útil de la trama codificada mientras satisface las restricciones de que un bloque de análisis RAP se encuentra con los bloques de análisis M del inicio del segmento RAP y los bloques de análisis transitorios deben estar dentro de los primeros bloques de análisis L de un segmento; y

empaquetar parámetros RAP que indican la existencia y ubicación de los datos de navegación RAP y flujo de bits en el encabezado de la trama.

4. El método de la reivindicación 1, que comprende, además:

usar la ubicación detectada del bloque de análisis transitorio para determinar la duración máxima del segmento como una potencia de dos de la duración del bloque de análisis de manera que dicho transitorio se encuentre dentro de los primeros bloques de análisis L de un segmento,

en donde se determina una duración de segmento uniforme que es una potencia de dos de la duración del bloque de análisis y que no excede la duración máxima del segmento para reducir la carga útil de trama

codificada sujeta a las restricciones.

5. El método de la reivindicación 1, en donde el número máximo de bytes para la carga útil del segmento codificado se impone mediante una restricción de tamaño de unidad de acceso del flujo de bits de audio.

6. El método de la reivindicación 1, en donde dicho flujo de bits incluye conjuntos de canales primero y segundo, seleccionando dicho método el primer y segundo conjuntos de parámetros de predicción para cada canal en cada conjunto de canales basándose en la detección de transitorios en diferentes ubicaciones para al menos un canal en los conjuntos de canales respectivos, en donde dicha duración del segmento se determina de modo que cada uno de dichos transitorios se encuentre dentro de los primeros bloques de análisis L de un segmento en el que ocurre el transitorio.

7. El método de la reivindicación 1, en donde los parámetros transitorios incluyen una bandera transitoria que indica la existencia de un transitorio y una ID transitoria que indica el número de segmento en el que ocurre el transitorio.

8. El método de la reivindicación 1, que comprende además generar un canal decorrelacionado para pares de canales para formar un triplete que incluye una base, correlacionado y canales decorrelacionado, seleccionar o un primer par de canales que incluye una base y un canal correlacionado o un segundo par de canales que incluye una base y un canal decorrelacionado, y entropía que codifica los canales en los pares de canales seleccionados.

9. El método de la reivindicación 8, en el que los pares de canales se seleccionan mediante:

Si la varianza del canal decorrelacionado es menor que la varianza del canal correlacionado por un umbral, seleccionar el segundo par de canales antes de determinar la duración de segmento; y

De lo contrario, diferir la selección del primer o del segundo pares de canales hasta la determinación de la duración de segmento basándose en qué par de canales contribuye con menos bits a la carga útil codificada.

10. Un método de decodificación de un flujo de bits de audio multicanal de tasa de bits variable sin pérdida, VBR, que comprende:

recibir un flujo de bits de audio multicanal de VBR sin pérdida como una secuencia de tramas divididas en una pluralidad de segmentos que tienen una carga útil de trama de longitud variable y que incluye al menos un conjunto de canales que pueden reconstruirse sin pérdida y descodificarse independientemente incluyendo una pluralidad de canales de audio para una señal de audio multicanal, comprendiendo cada trama información de encabezamiento que incluye la duración del segmento, información de encabezado de conjunto de canales que incluyen parámetros de transitorio que indican la existencia y la ubicación de un segmento de transitorio, en cada coeficiente de predicción de canal para cada dicho canal que incluye un conjunto único de coeficientes de predicción basados en trama si ningún transitorio está presente y primeros y segundos conjuntos de coeficientes de predicción basados en partición si hay un transitorio presente en cada dicho conjunto de canales e información de encabezamiento de segmento para cada dicho conjunto de canales que incluye al menos una bandera de código por entropía y al menos un parámetro de codificación por entropía, y señales de audio multicanal comprimidas y codificadas por entropía almacenadas en dicho número de segmentos;

desempaquetar el encabezamiento para extraer la duración de segmento;

desempaquetar el encabezamiento para el al menos un dicho conjunto de canales para extraer la bandera de código y el parámetro de codificación por entropía y las señales de audio multicanal comprimidas y codificadas por entropía para cada segmento y realizar una decodificación por entropía en cada segmento usando un código por entropía seleccionada y el parámetro de codificación por entropía para generar señales de audio comprimidas para cada segmento;

desempaquetar el encabezamiento para el al menos un dicho conjunto de canales para extraer los parámetros transitorios para determinar la existencia y la ubicación de segmentos transitorios en cada canal en el conjunto de canales;

desempaquetar el encabezamiento para el al menos un dicho conjunto de canales para extraer el conjunto único de coeficientes de predicción basados en tramas o primer y segundo conjuntos de coeficientes de predicción basados en división para cada canal dependiendo de la existencia de un transitorio; y

para cada canal en el conjunto de canales, aplicar o bien el conjunto único de coeficientes de predicción a las señales de audio comprimidas para todos los segmentos en la trama para reconstruir sin pérdida audio de modulación de código de pulso PCM o bien aplicar el primer conjunto de coeficientes de predicción a las señales de audio comprimidas comenzando en el primer segmento y aplicar el segundo conjunto de coeficientes de predicción a las señales de audio comprimidas comenzando en el segmento transitorio.

11. El método de la reivindicación 10, en el que el flujo de bits comprende además información de encabezamiento de conjunto de canales que incluye una bandera de decorrelación de canal de pares, un orden de canales original y coeficientes de decorrelación de canales cuantificados, generando dicha reconstrucción audio de PCM decorrelacionada, comprendiendo además el método:

desempaquetar el encabezamiento para extraer el orden de canales original, la bandera de decorrelación de canal de pares y los coeficientes de decorrelación de canales cuantificados y realizar una decorrelación inversa de canal

transversal para reconstruir audio de PCM para cada canal de audio en dicho conjunto de canales.

- 5 12. El método de la reivindicación 11, en el que la bandera de decorrelación de canal de pares indica si se codificó un primer par de canales que incluye un canal base y uno correlacionado o un segundo par de canales que incluye el canal base y uno decorrelacionado para un triplete que incluye los canales base, correlacionado y decorrelacionado, comprendiendo además el método:
si la bandera indica un segundo par de canales, multiplicar el canal base por el coeficiente de decorrelación del canal cuantificado y añadirlo al canal decorrelado para generar audio PCM en el canal correlacionado.
- 10 13. El método de la reivindicación 10, que comprende, además:
recibir una trama que tiene información de encabezado que incluye un punto de acceso aleatorio, RAP, parámetros que indican la existencia y la ubicación de hasta un segmento RAP y datos de navegación;
desempaquetar el encabezado del siguiente cuadro en el flujo de bits para extraer los parámetros RAP, si se
15 intenta iniciar la decodificación en RAP saltando al siguiente cuadro hasta que se detecte un cuadro que tenga un segmento RAP y utilizar los datos de navegación para navegar al comienzo del segmento RAP; y
cuando se encuentra un segmento RAP, desactivar la predicción para las primeras muestras de audio hasta el orden de predicción para reconstruir sin pérdidas el audio PCM.
- 20 14. El método de la reivindicación 10, en donde el número y la duración de los segmentos varía de trama a trama para minimizar la carga útil de longitud variable de cada cuadro sujeto a restricciones de que la carga útil del segmento codificado sea menor que un número máximo de bytes menor que el tamaño de la trama y reconstruible sin pérdidas.
- 25 15. Uno o más medios legibles por ordenador que comprenden instrucciones ejecutables por ordenador que, cuando se ejecutan, realizan el método de acuerdo con la reivindicación 1 o la reivindicación 10.
- 30 16. Uno o más dispositivos de semiconductores que comprenden circuitos digitales configurados para realizar el método de acuerdo con la reivindicación 1 o la reivindicación 10.
- 35 17. Un decodificador de audio multicanal para la decodificación de un flujo de bits de audio multicanal de tasa de bits variable sin pérdida, VBR, en donde dicho decodificador está configurado para:
recibir un flujo de bits de audio multicanal VBR sin pérdidas como una secuencia de tramas divididas en una
pluralidad de segmentos que tienen una carga útil de trama de longitud variable e incluye al menos un conjunto
de canales independientemente decodificables y reconstruibles sin pérdidas que incluye una pluralidad de
canales de audio para un audio multicanal señal, comprendiendo cada trama información de encabezado,
incluida la duración del segmento, información de encabezado del conjunto de canales que incluye parámetros
transitorios que indican la existencia y la ubicación de un segmento transitorio en cada canal, coeficientes de
40 predicción para cada uno de dichos canales, incluido un conjunto único de coeficientes de predicción basados en tramas si no hay transitorios y primer y segundo conjuntos de coeficientes de predicción basados en particiones si hay un transitorio presente en cada conjunto de canales, e incluyendo la información del encabezado del segmento para cada conjunto de canales al menos una bandera de código por entropía y al menos un parámetro de codificación por entropía, y señales de audio multicanal comprimidas codificadas por entropía almacenadas en dicho número de segmentos;
desempaquetar el encabezado para extraer la duración del segmento;
desempaquetar el encabezado para el al menos un conjunto de canales para extraer la bandera de código por entropía y el parámetro de codificación y las señales de audio multicanal comprimidas codificadas por entropía para cada segmento y realizar una decodificación por entropía en cada segmento utilizando un código por entropía seleccionado y un parámetro de codificación para generar señales de audio comprimidas para cada
50 segmento;
desempaquetar el encabezado para el al menos un dicho conjunto de canales para extraer los parámetros transitorios para determinar la existencia y la ubicación de segmentos transitorios en cada canal en el conjunto de canales;
desempaquetar el encabezado para el al menos uno de dichos conjuntos de canales para extraer el conjunto único de coeficientes de predicción basados en cuadros o el primer y segundo conjuntos de coeficientes de predicción basados en particiones para cada canal, dependiendo de la existencia de un transitorio; y
para cada canal en el conjunto de canales, aplicar el conjunto único de coeficientes de predicción a las señales de audio comprimidas para todos los segmentos en la trama para reconstruir sin pérdida audio de modulación del código de pulso, PCM, o aplicar el primer conjunto de coeficientes de predicción a las señales de audio comprimidas que comienzan en el primer segmento y aplicar el segundo conjunto de coeficientes de predicción a las señales de audio comprimidas que comienzan en el segmento transitorio.
- 60

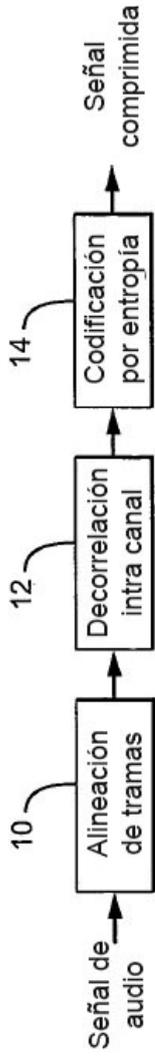


FIG. 1
TÉCNICA ANTERIOR

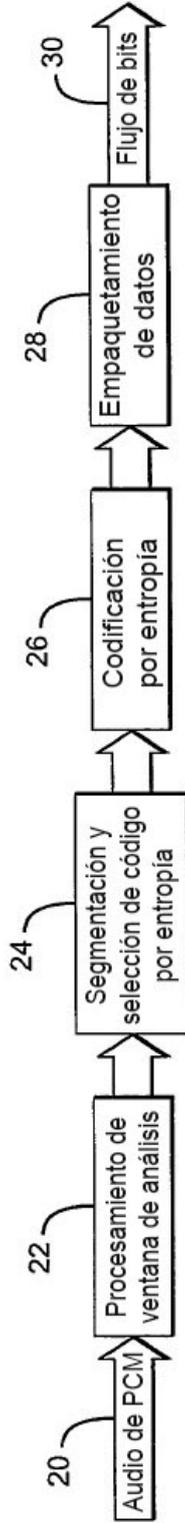


FIG. 2a

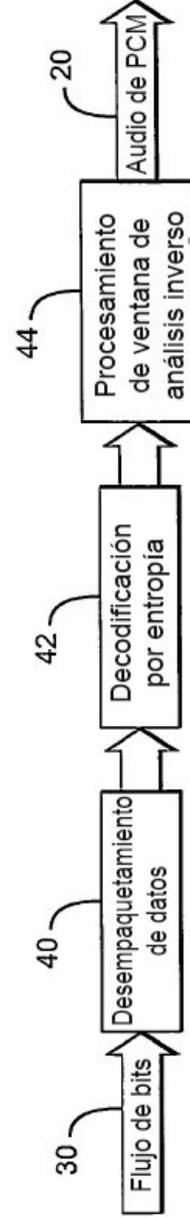
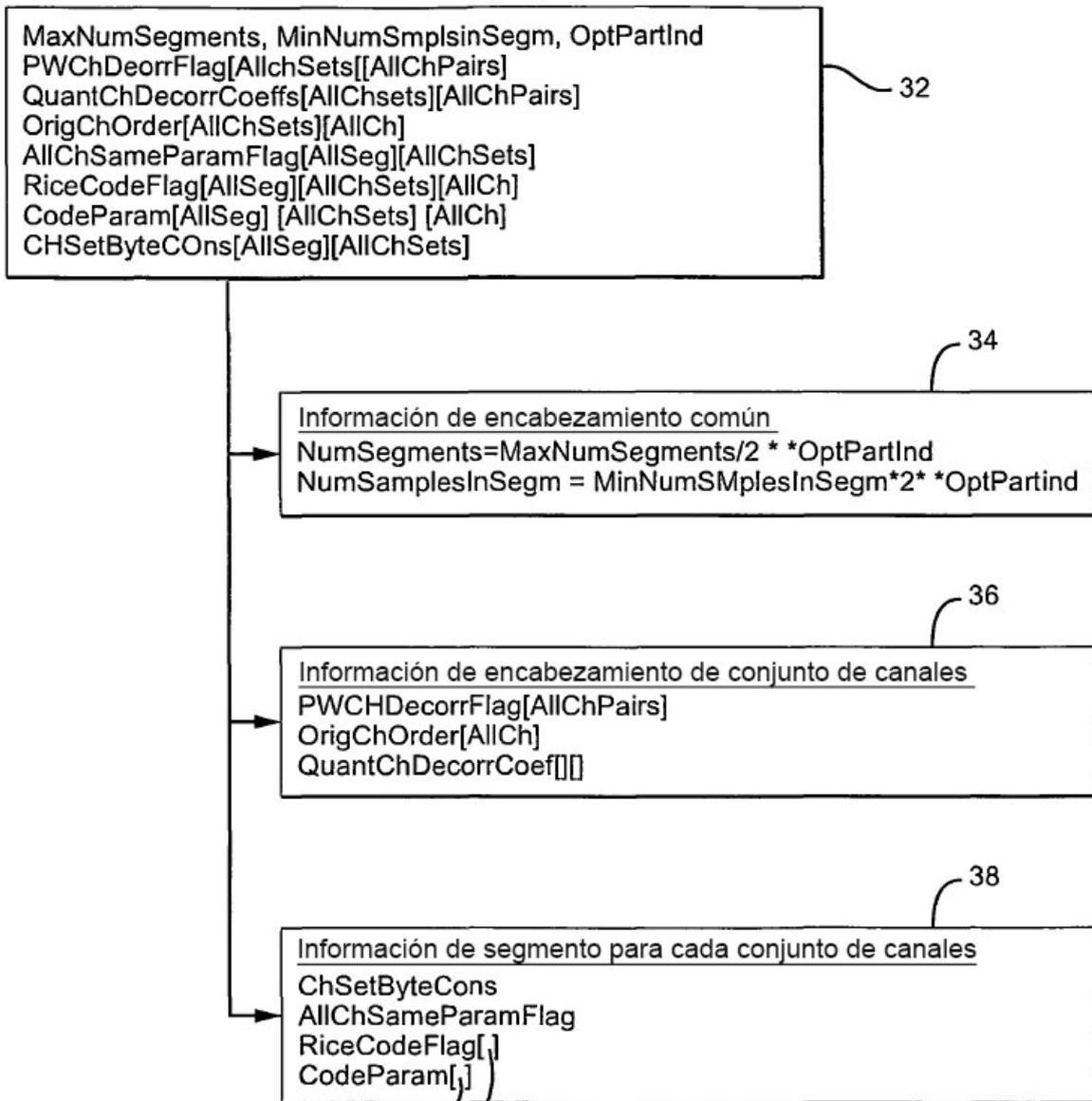


FIG. 2b



Si AllChSameParamFlag == verdadero
 - única RiceCodeFlag se transmite por conjunto de canales
 - único CodeParam se transmite por conjunto de canales
 De otro modo
 - RiceCodeFlag se transmite para cada canal en el conjunto de canales
 - CodeParam se transmite para cada canal en el conjunto de canales

FIG. 3

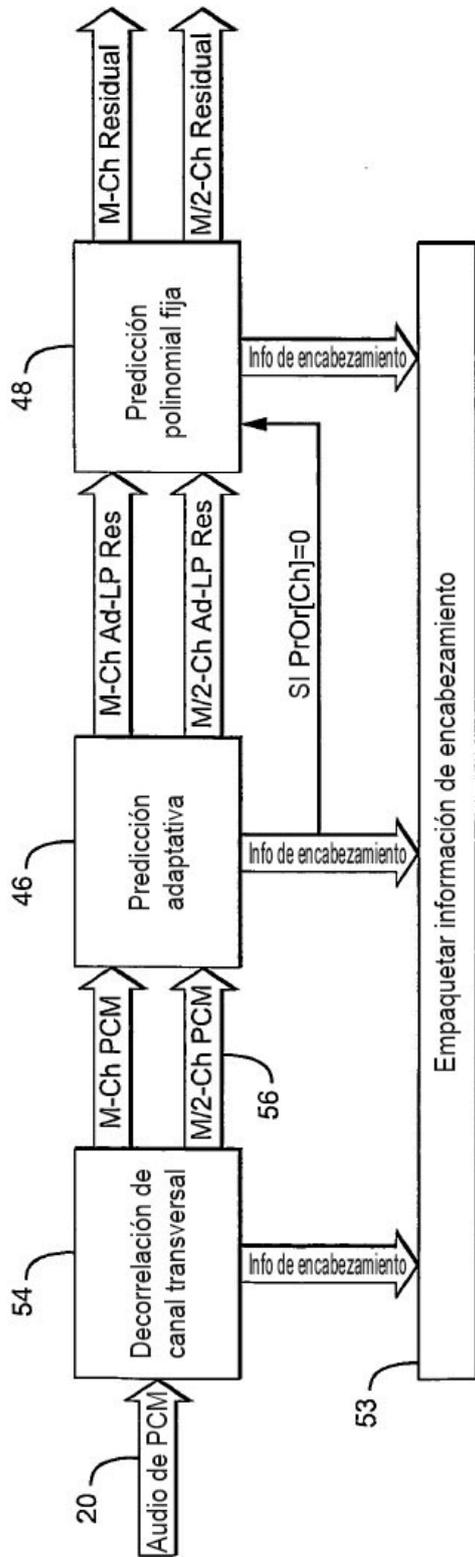
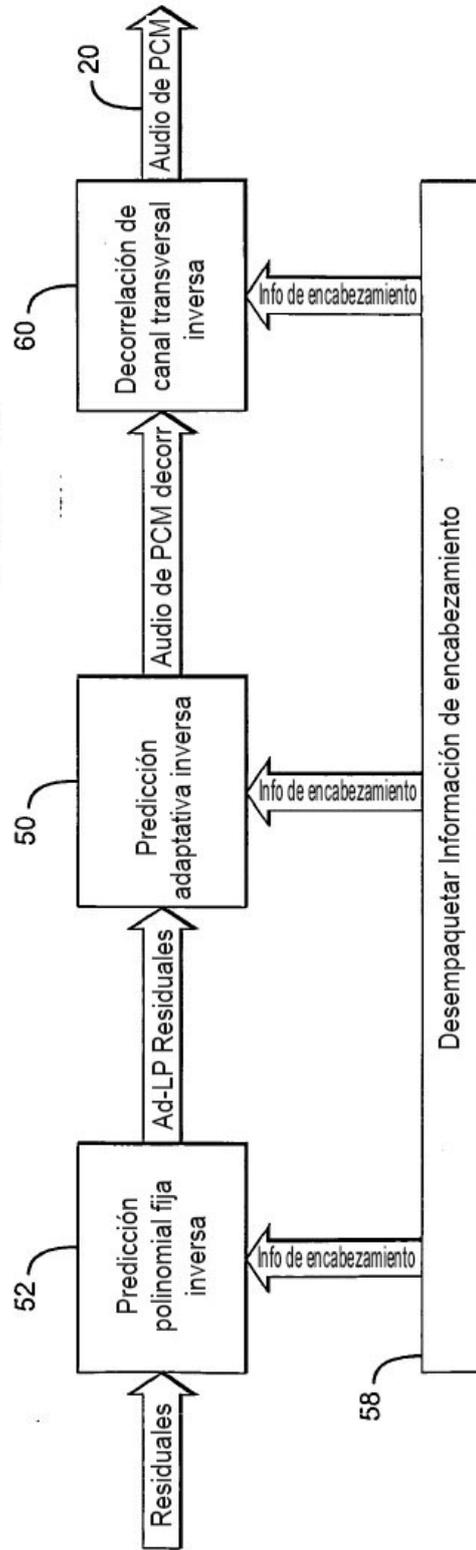


FIG. 4a

FIG. 4b



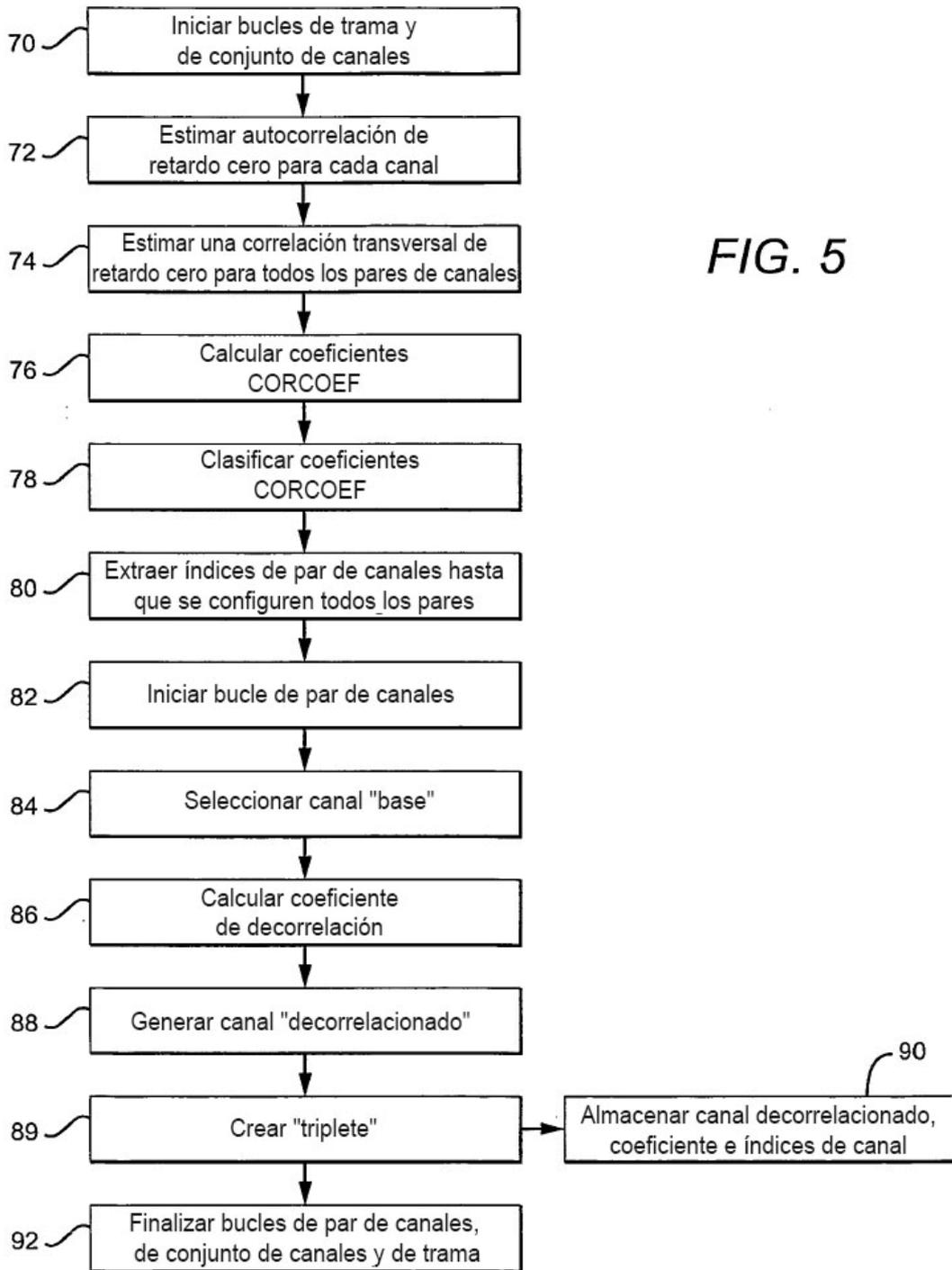


FIG. 5

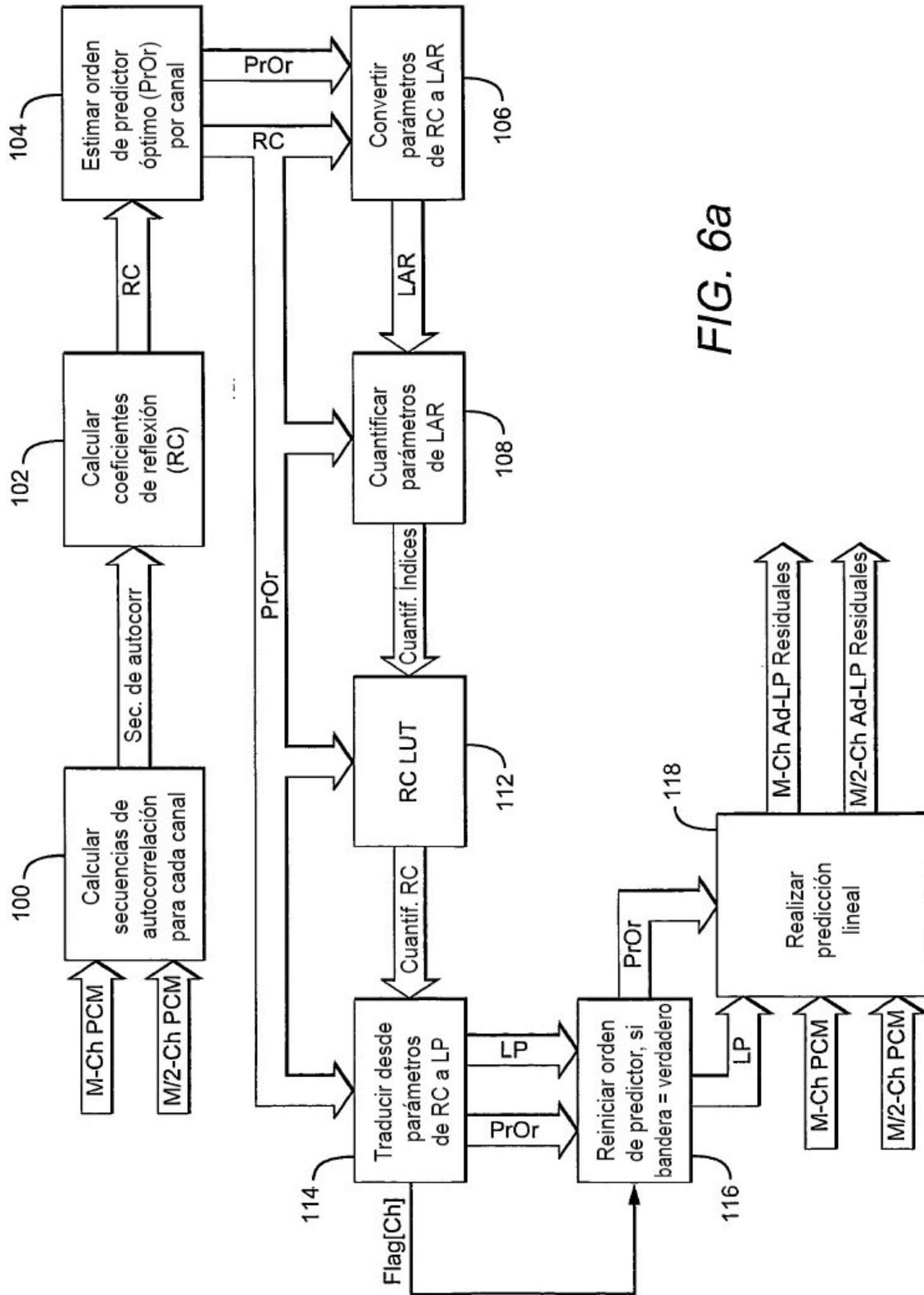


FIG. 6a

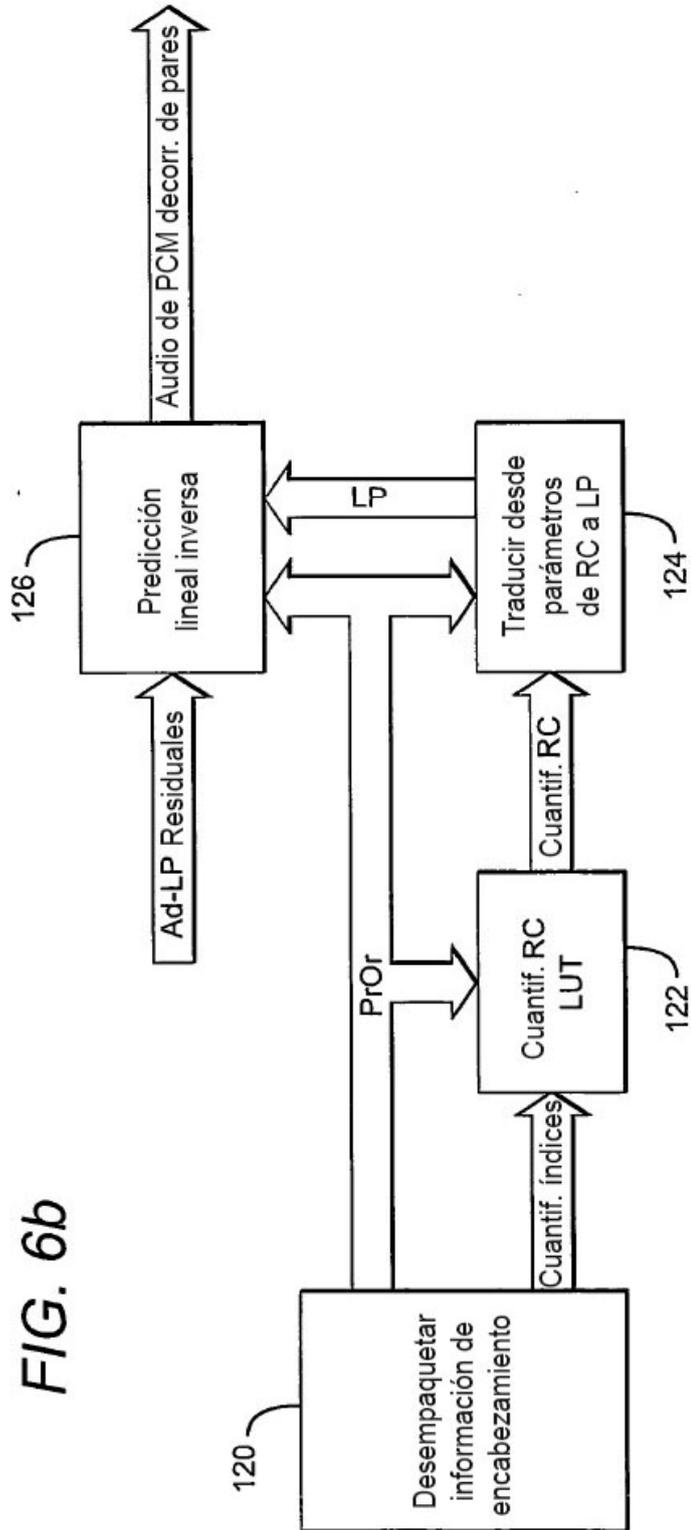
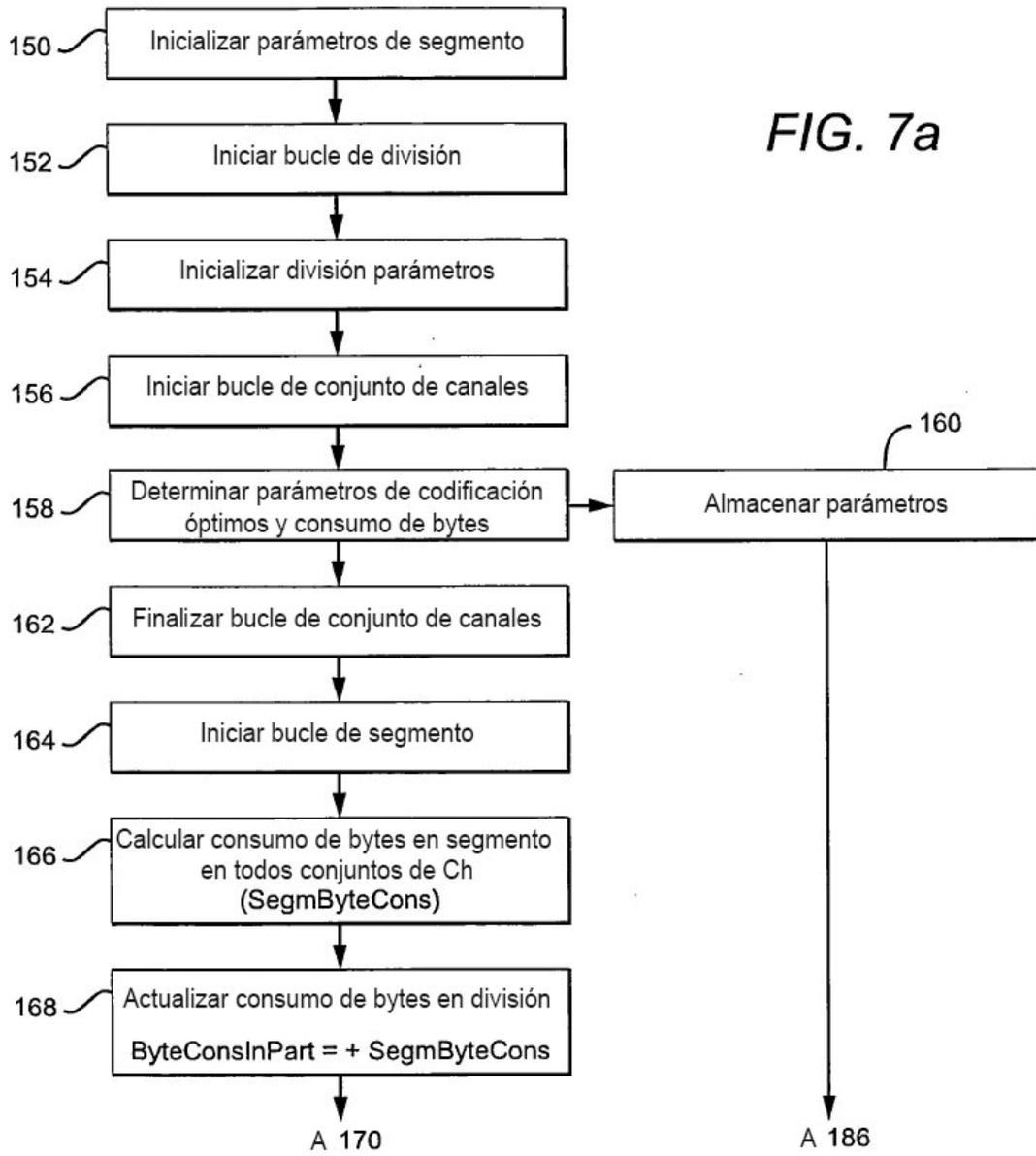


FIG. 6b



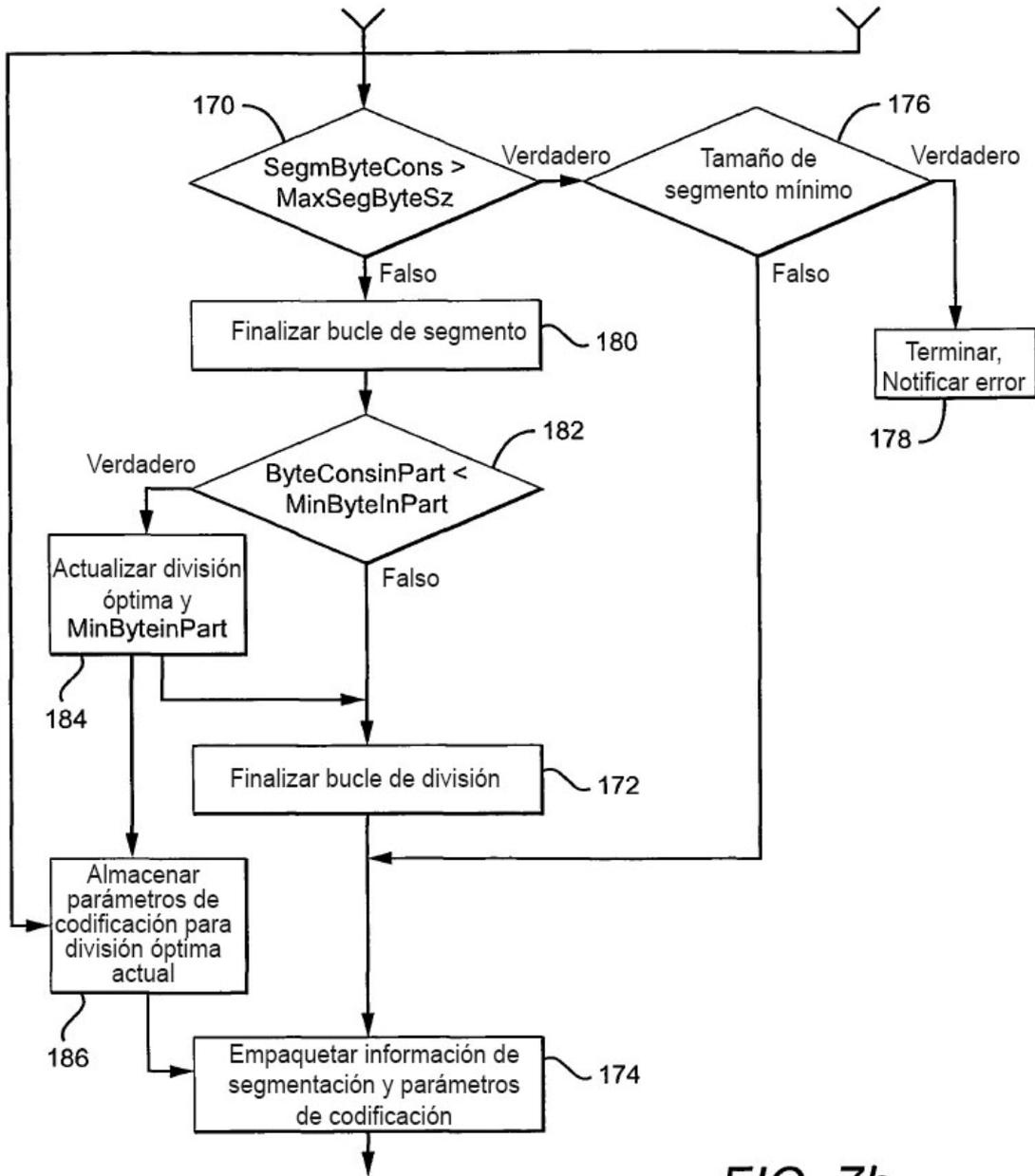
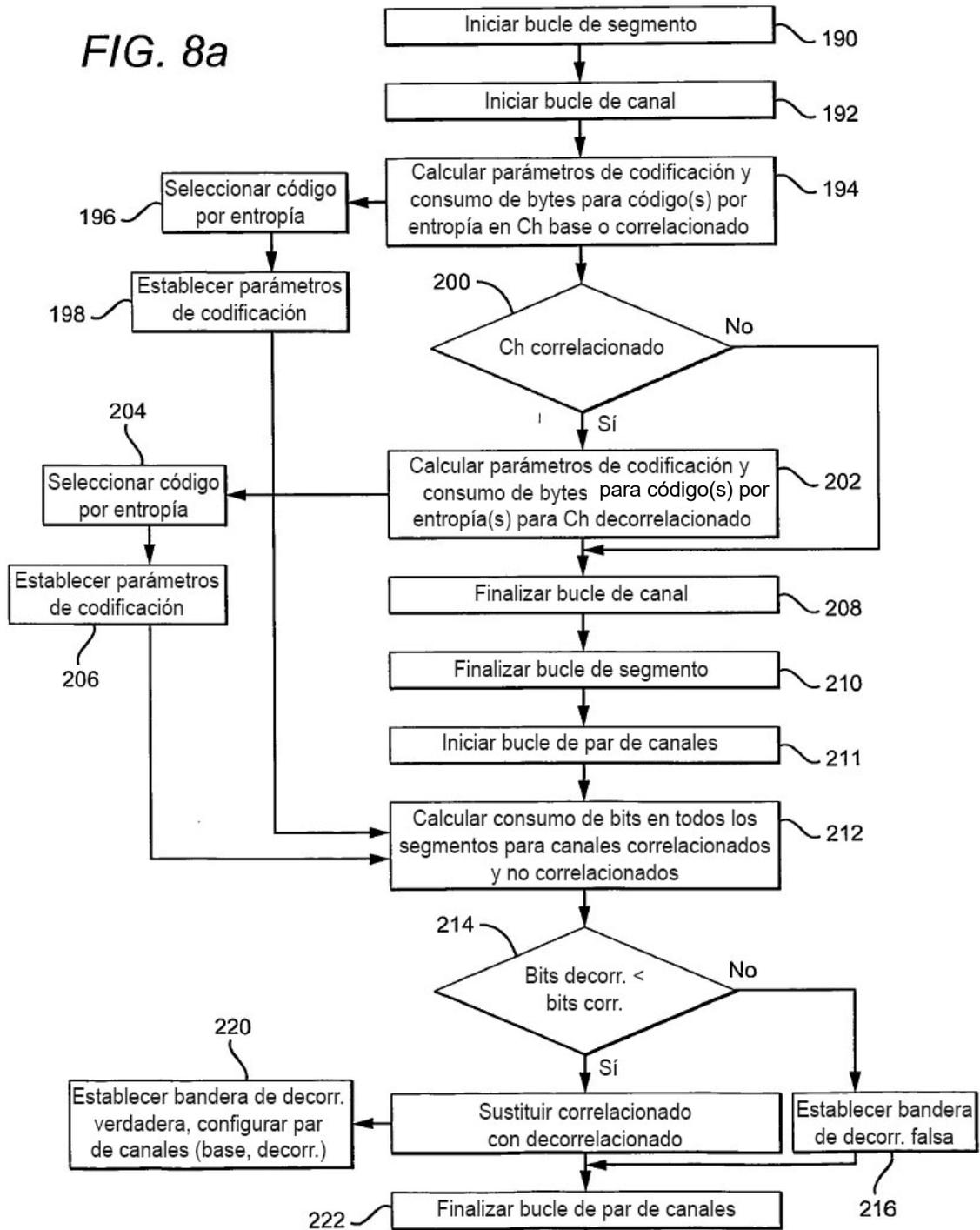


FIG. 7b

FIG. 8a



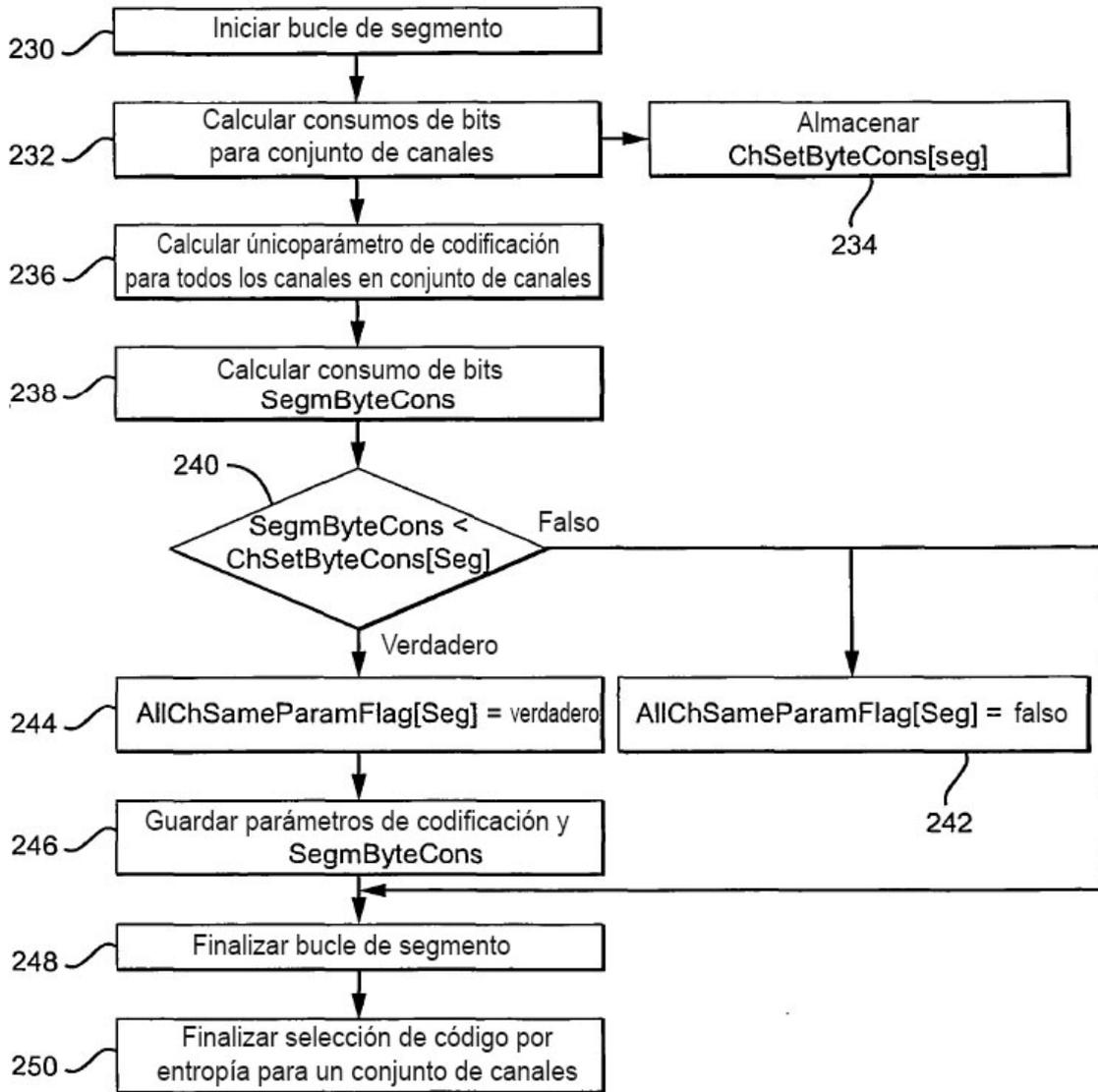
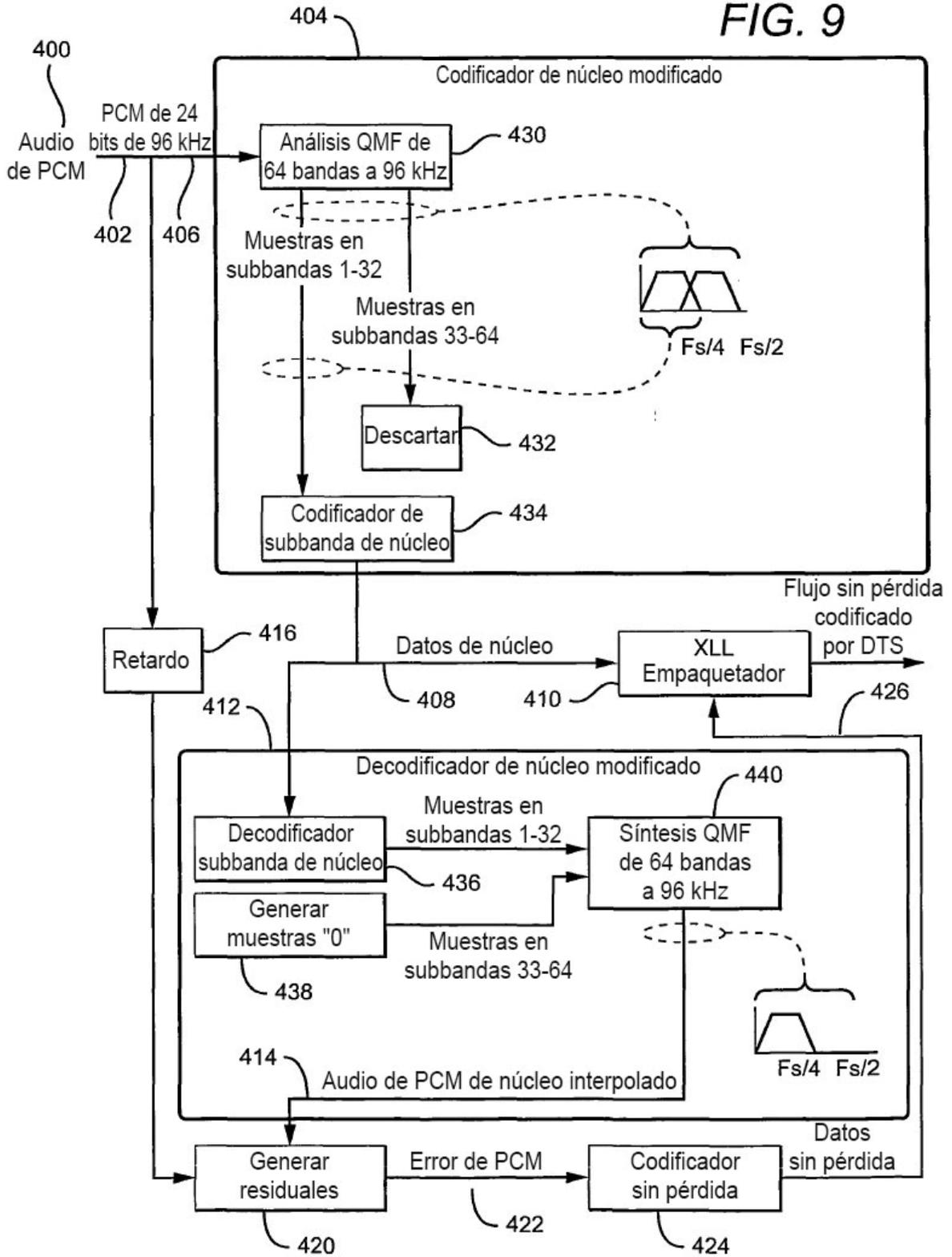


FIG. 8b

FIG. 9



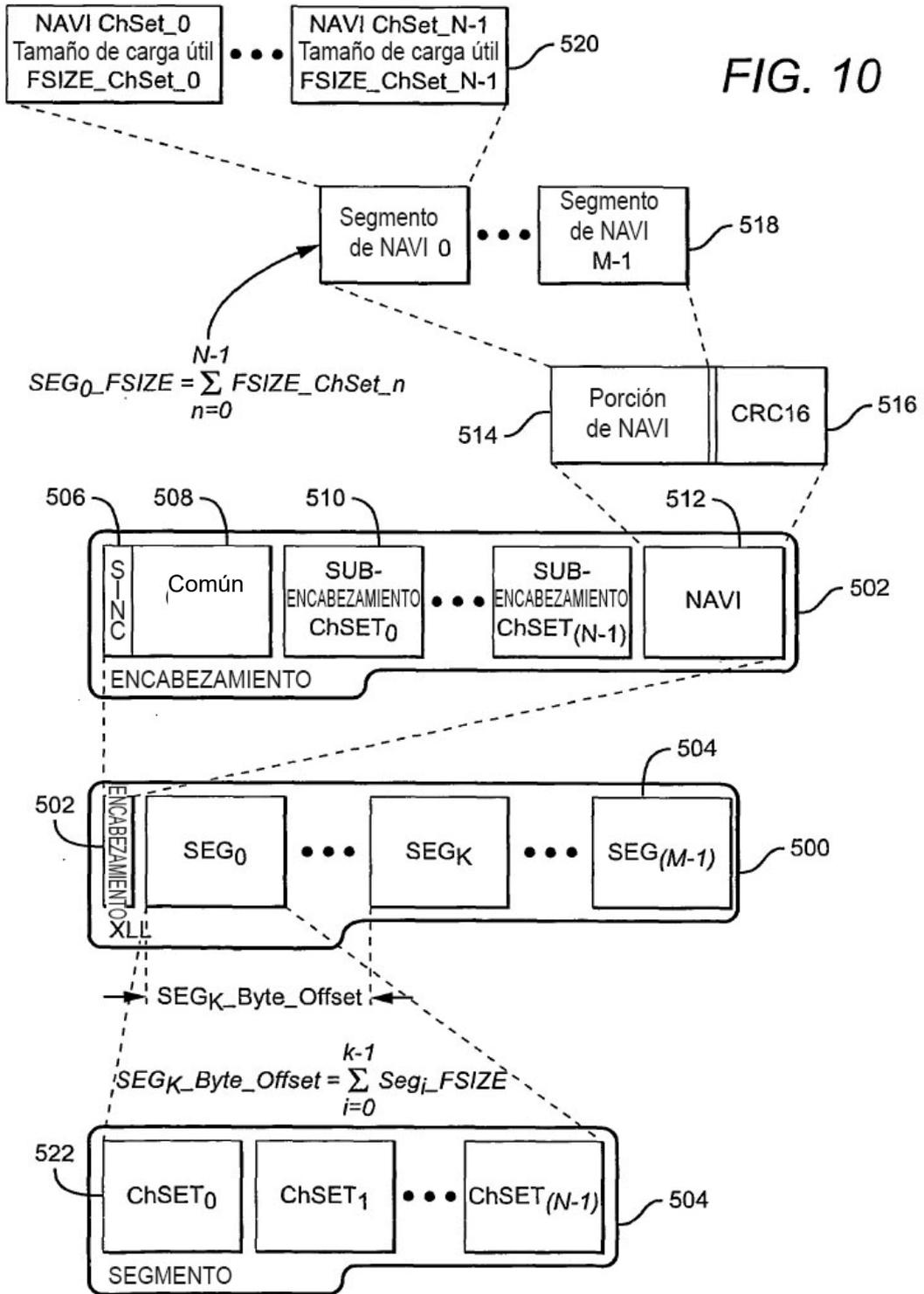


FIG. 11a

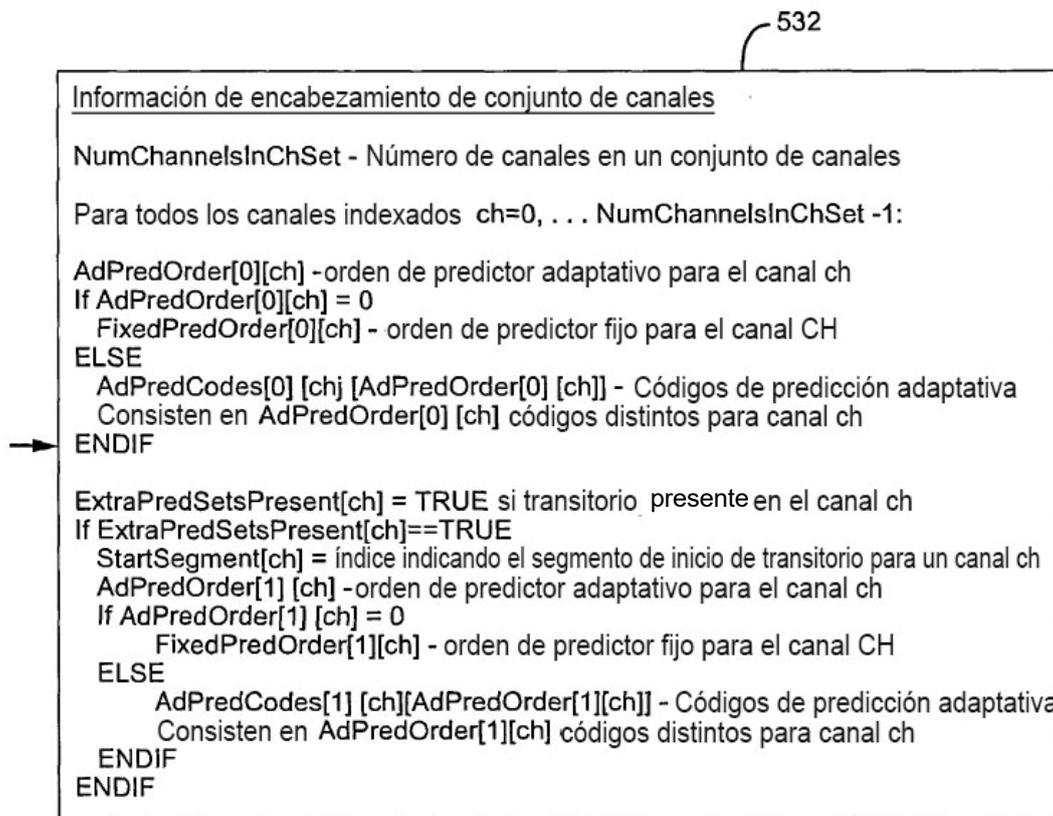
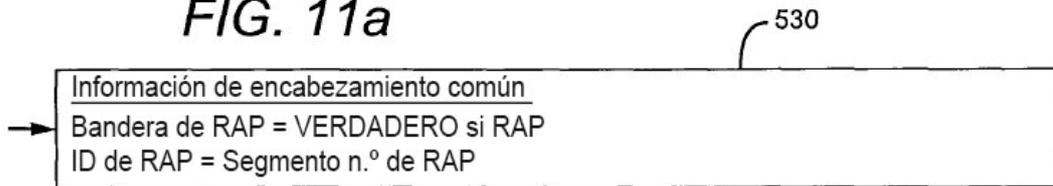


FIG. 11b

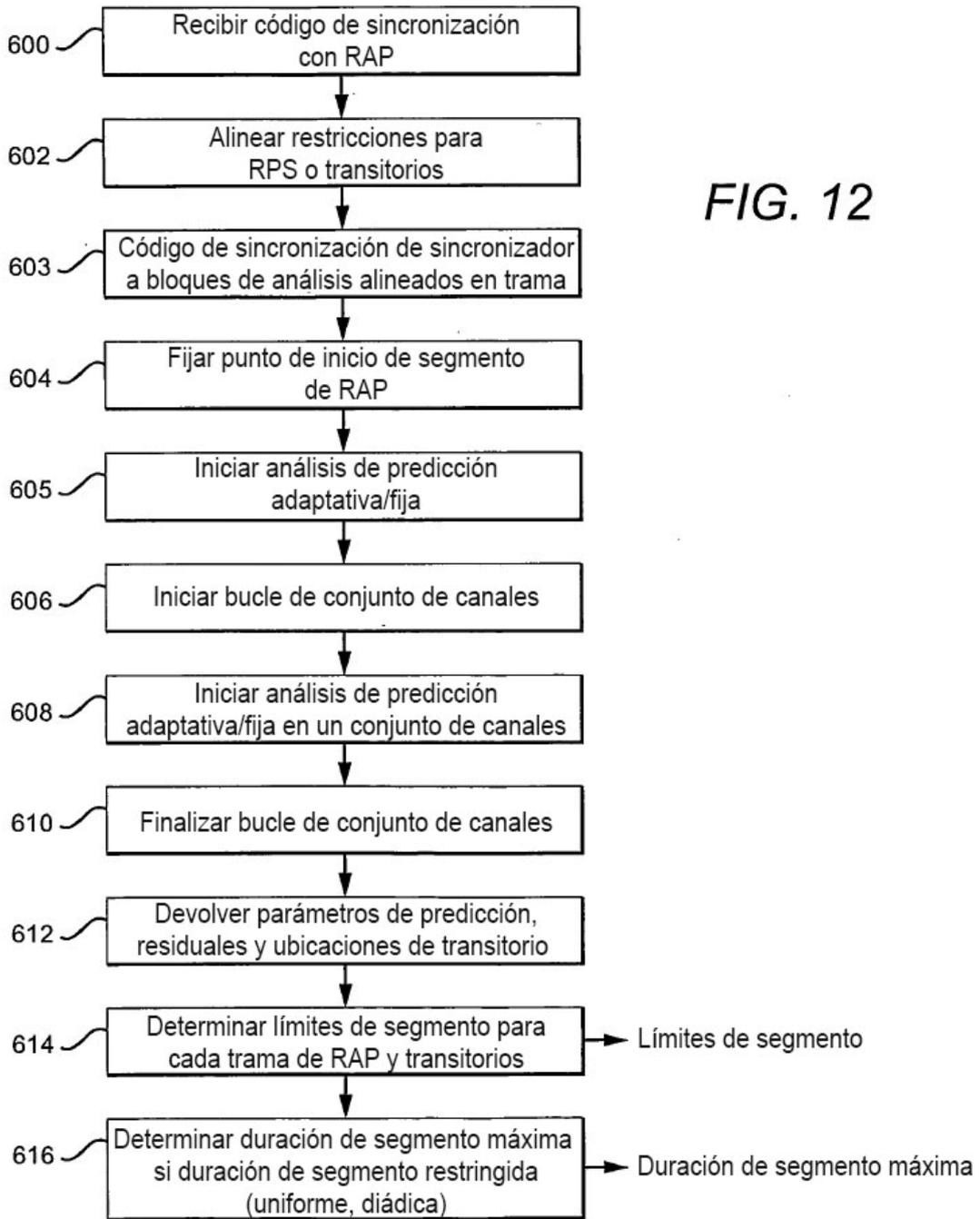


FIG. 12

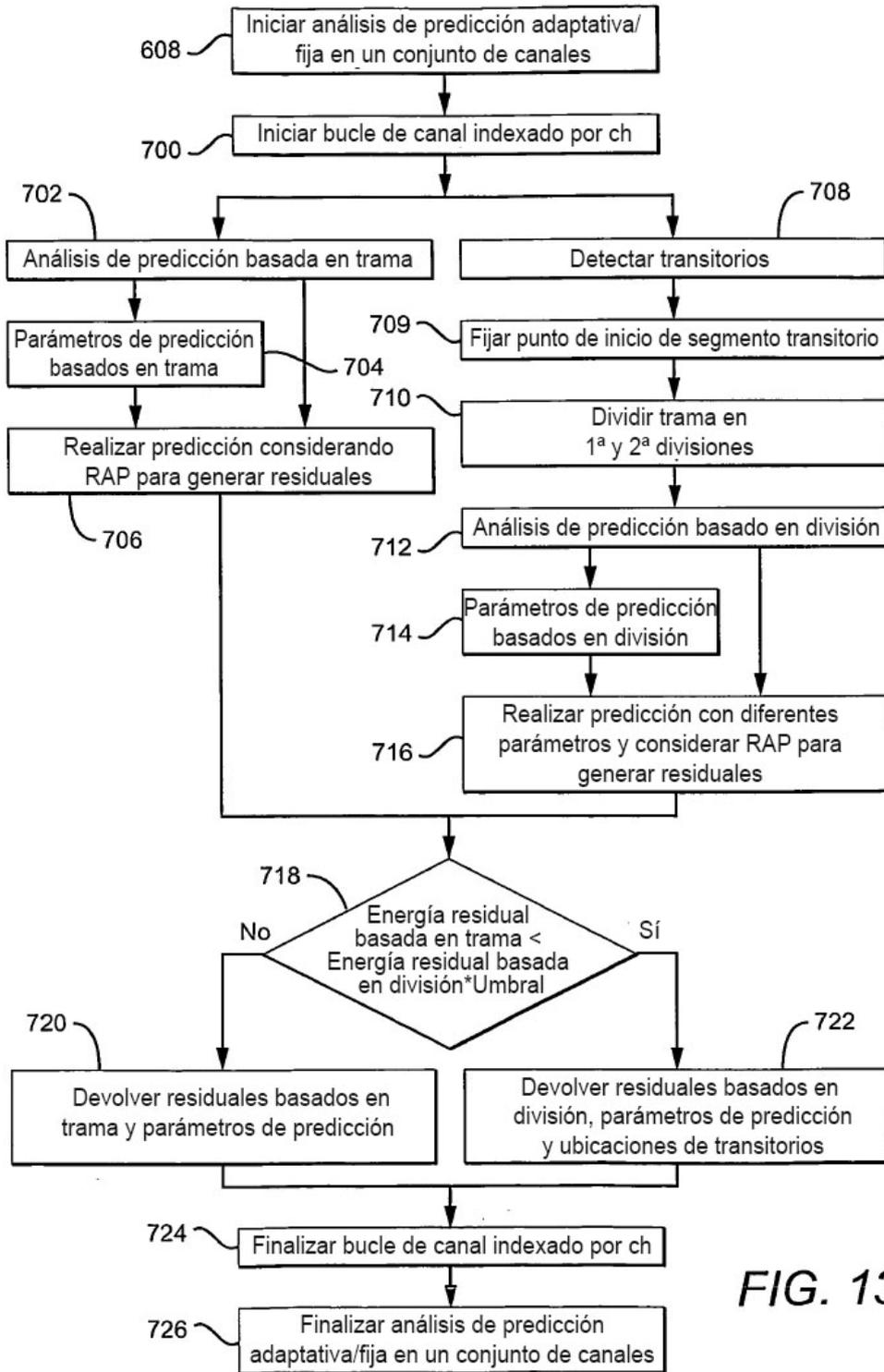
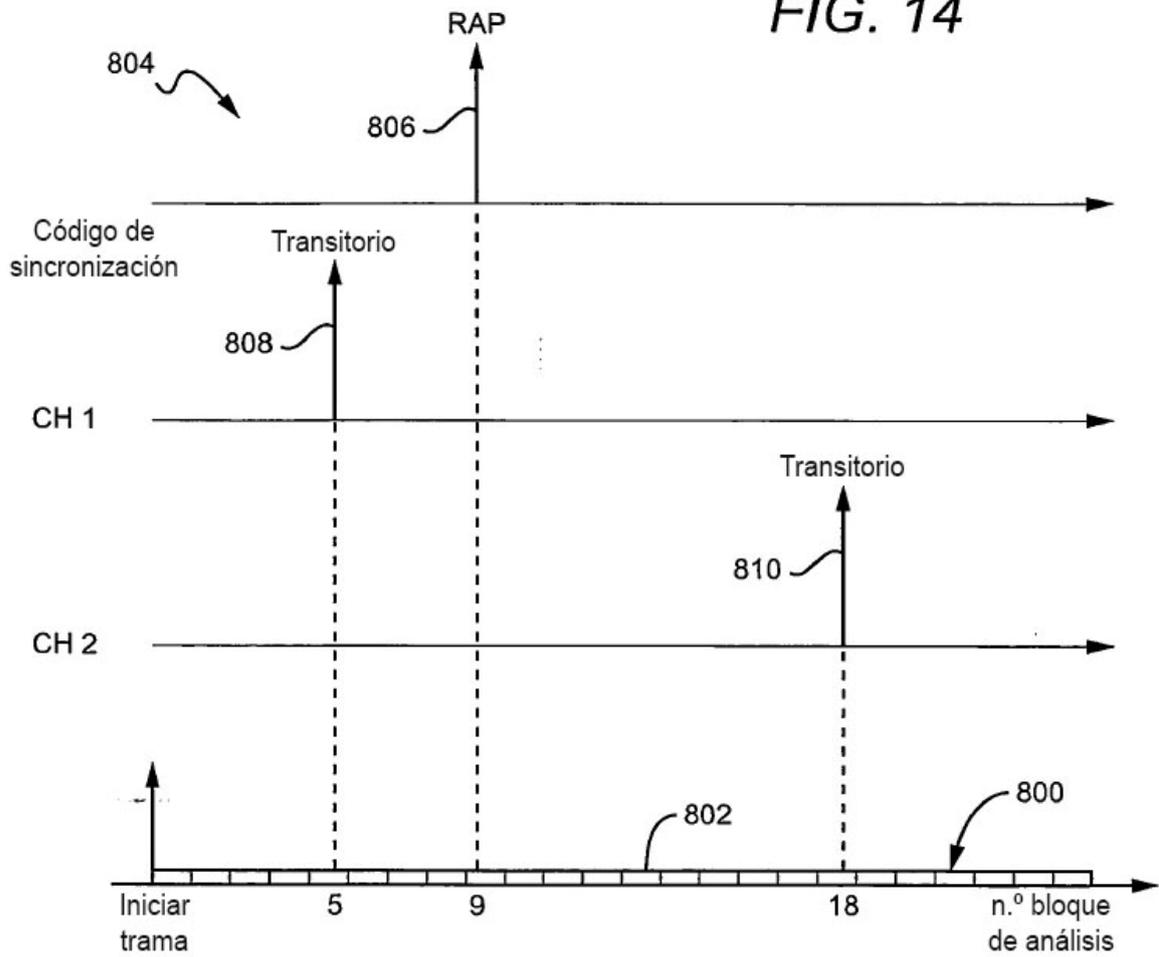


FIG. 13

FIG. 14



Duración de segmento restringido

Duración máxima de RAP = 8 ABs

CH 1 máx = 4 ABs

CH 2 máx = 1 AB, si 2º AB Válido entonces 16

Elegir duración de trama máxima = 4 x AB_dur.

Duración de segmento no restringido

Inicio de segmento 1: 1er segmento

Inicio de segmento 2: 5º segmento

Inicio de segmento 3: 9º segmento

Inicio de segmento 4: 18º segmento

FIG. 15a

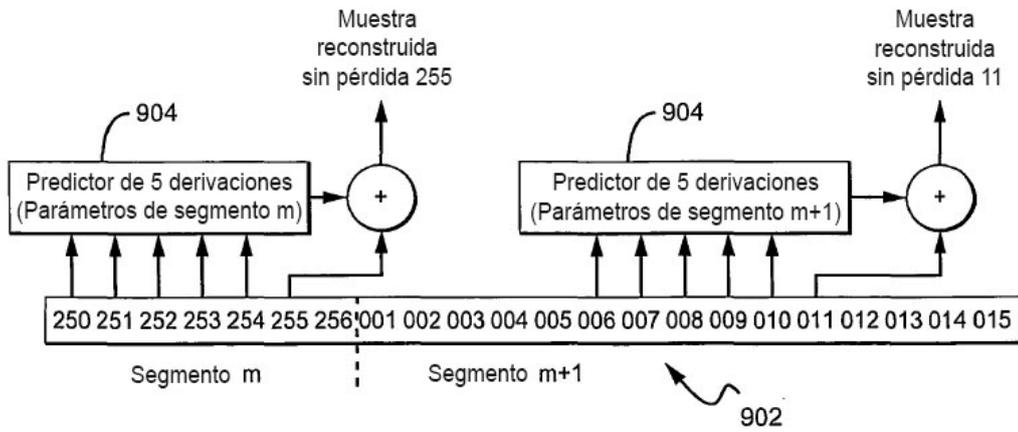
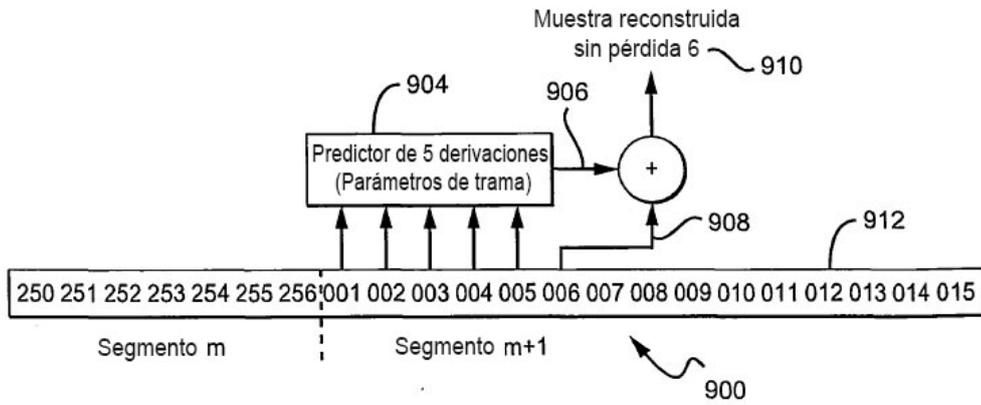


FIG. 15b

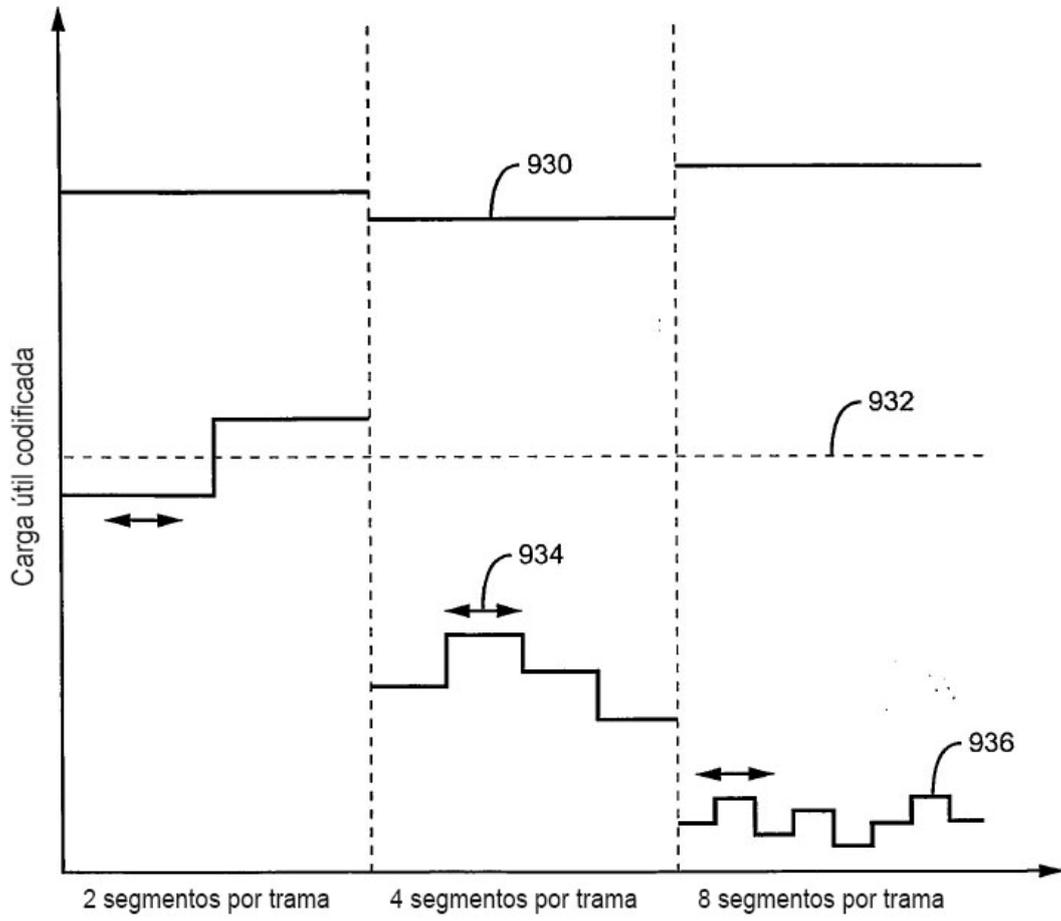


FIG. 16