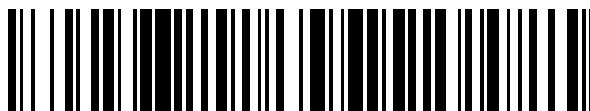


19



OFICINA ESPAÑOLA DE  
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 463 840**

51 Int. Cl.:

**G10L 19/008** (2013.01)

**G10L 19/02** (2013.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

96 Fecha de presentación y número de la solicitud europea: **28.10.2010 E 10776017 (5)**

97 Fecha y número de publicación de la concesión europea: **19.03.2014 EP 2510515**

54 Título: **Descodificación de trenes de bits codificados de audio multicanal usando transformación híbrida adaptativa**

30 Prioridad:

**07.12.2009 US 267422 P**

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

**29.05.2014**

73 Titular/es:

**DOLBY LABORATORIES LICENSING  
CORPORATION (100.0%)  
100 Potrero Avenue  
San Francisco, CA 94103-4813, US**

72 Inventor/es:

**RAMAMOORTHY, KAMALANATHAN**

74 Agente/Representante:

**PÉREZ BARQUÍN, Eliana**

**ES 2 463 840 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

Descodificación de trenes de bits codificados de audio multicanal usando transformación híbrida adaptativa

5 **Referencia cruzada a solicitudes relacionadas**

Esta solicitud reivindica prioridad respecto a la solicitud provisional de patente de Estados Unidos nº 61/267.422 presentada el 7 de diciembre de 2009.

10 **Campo técnico**

La presente invención se refiere en general a sistemas de codificación de audio y se refiere más específicamente a métodos y dispositivos que decodifican señales digitales de audio codificadas.

15 **Técnica antecedente**

El United States Advanced Television Systems Committee (ATSC), Inc., que fue formado por organizaciones miembros del Joint Committee on InterSociety Coordination (JCIC), desarrolló un conjunto coordinado de estándares nacionales para el desarrollo de los servicios de la televisión nacional de EE.UU. Estos estándares, incluyendo estándares relevantes de codificación/descodificación de audio, se exponen en varios documentos incluyendo el Documento A/52B titulado "Digital Audio Compression Standard (AC-3, E-AC-3)", Revisión B, publicado el 14 de junio de 2005. El algoritmo de codificación de audio especificado en el documento A/52B se denomina "AC-3". Una versión mejorada de este algoritmo, que se describe en el Anexo E del documento, se denomina "E-AC-3". Estos dos algoritmos se denominan en este documento como "AC-3" y los estándares pertinentes se denominan en este documento como los "estándares ATSC".

El documento A/52B no especifica muchos aspectos de diseño del algoritmo sino que, más bien, describe una "sintaxis del tren de bits" que define características estructurales y sintácticas de la información codificada que un descodificador compatible debe ser capaz de descodificar. Muchas aplicaciones que cumplen los estándares ATSC transmitirán información de audio digital codificada como datos binarios en serie. Como resultado, los datos codificados a menudo se denominan como tren de bits pero son permisibles otras disposiciones de los datos. Por facilidad de discusión, el término "tren de bits" se usa en este documento para denominar una señal de audio digital codificada independientemente del formato o la técnica de grabación o transmisión que se use.

Un tren de bits que cumpla los estándares ATSC está dispuesto en una serie de "tramas de sincronización". Cada trama es una unidad del tren de bits que es capaz de ser totalmente descodificada en uno o más canales de datos de audio digital modulados por impulsos codificados (PCM). Cada trama incluye "bloques de audio" y metadatos de trama que están asociados con los bloques de audio. Cada uno de los bloques de audio contiene datos de audio codificado que representan muestras de audio digital para uno o más canales de audio y metadatos de bloque asociados a los datos de audio codificados.

Aunque los detalles del diseño algorítmico no se especifican en los estándares ATSC, ciertas características algorítmicas han sido ampliamente adoptadas por los fabricantes de equipo profesional y para el consumidor. Una característica universal de implementación para los descodificadores que pueden descodificar trenes de bits AC-3 mejorado generados por codificadores E-AC-3 es un algoritmo que descodifica todos los datos codificados en una trama para un canal respectivo antes de descodificar los datos para otro canal. Este planteamiento se ha usado para mejorar el rendimiento de las implementaciones en procesadores de un solo chip que tienen poca memoria integrada en el chip porque algunos procesos de descodificación requieren datos para un canal dado procedentes de cada uno de los bloques de audio de una trama. Procesando los datos codificados en el orden del canal, las operaciones de descodificación pueden realizarse usando la memoria integrada en el chip para un canal particular. Los datos del canal descodificados pueden ser transferidos posteriormente a memoria externa al chip para liberar recursos integrados en el chip para el siguiente canal.

Un tren de bits que cumpla los estándares ATSC puede ser muy complejo porque es posible un gran número de variaciones. Unos pocos ejemplos aquí mencionados sólo incluyen brevemente acoplamiento de canales, rematrización de canales, normalización de diálogo, compresión de gama dinámica, submezcla de canales y conmutación de longitud de bloque para trenes de bits AC-3 estándar, y múltiples trenes independientes, subtrenes dependientes, extensión espectral y transformación híbrida adaptativa para trenes de bits AC-3 mejorado. Los detalles para estas características pueden obtenerse del documento A/52B.

Procesando cada canal independientemente, pueden simplificarse los algoritmos requeridos para estas variaciones. Los procesos complejos subsiguientes como el filtrado de síntesis pueden realizarse sin preocupación para estas variaciones. Los algoritmos más sencillos parecerían proporcionar un beneficio al reducir los recursos computacionales necesarios para procesar una trama de datos de audio.

Por desgracia, este planteamiento requiere que el algoritmo de descodificación lea y examine datos en todos los

bloques de audio dos veces. Cada iteración de lectura y examen de datos de bloques de audio en una trama se denomina en este documento como una “pasada” sobre los bloques de audio. La primera pasada realiza cálculos extensivos para determinar la ubicación de los datos de audio descodificados en cada bloque. La segunda pasada realiza muchos de estos mismos cálculos a medida que realiza los procesos de descodificación. Ambas pasadas requieren considerables recursos computacionales para calcular las ubicaciones de los datos. Si puede eliminarse la pasada inicial, puede ser posible reducir los recursos totales de procesamiento necesarios para descodificar una trama de datos de audio.

### Exposición de la invención

Un objeto de la presente invención es reducir los recursos computacionales requeridos para descodificar una trama de datos de audio en trenes de bits codificados dispuestos en unidades jerárquicas como las tramas y los bloques de audio mencionados anteriormente. El texto precedente y la siguiente descripción se refieren a trenes de bits codificados que cumplen los estándares ATSC pero la presente invención no está limitada al uso sólo con estos trenes de bits. Los principios de la presente invención pueden aplicarse esencialmente a cualquier tren de bits codificado que tenga características estructurales similares a las tramas, bloques y canales usados en algoritmos de codificación AC-3.

Según un aspecto de la presente invención, que se define por las reivindicaciones adjuntas, un método descodifica una trama de una señal de audio digital codificada recibiendo la trama y examinando la señal de audio digital codificada en una única pasada para descodificar los datos de audio codificados para cada bloque de audio en orden por bloque. Cada trama comprende metadatos de trama y una pluralidad de bloques de audio. Cada bloque de audio comprende metadatos de bloque y datos de audio codificados para uno o más canales de audio. Los metadatos de bloque comprenden información de control que describe las herramientas de codificación usadas por un proceso de codificación que produjo los datos de audio codificados. Una de las herramientas de codificación es el procesamiento de transformada híbrida que aplica un banco de filtros de análisis implementado por una transformada primaria a uno o más canales de audio para generar coeficientes espectrales que representan el contenido espectral del uno o más canales de audio, y aplica una transformada secundaria a los coeficientes espectrales para al menos algunos del uno o más canales de audio para generar coeficientes de la transformada híbrida. La descodificación de cada bloque de audio determina si el proceso de codificación usó procesamiento de transformada híbrida adaptativa para codificar cualquiera de los datos de audio codificados. Si el proceso de codificación usó procesamiento de transformada híbrida adaptativa, el método obtiene todos los coeficientes de la transformada híbrida para la trama a partir de los datos de audio codificados en el primer bloque de audio en la trama y aplica una transformada secundaria inversa a los coeficientes de la transformada híbrida para obtener coeficientes de la transformada secundaria inversa y obtiene coeficientes espectrales a partir de los coeficientes de la transformada secundaria inversa. Si el proceso de codificación no usó procesamiento de transformada híbrida adaptativa, los coeficientes espectrales se obtienen a partir de los datos de audio codificados en el bloque de audio respectivo. Se aplica una transformada primaria inversa a los coeficientes espectrales para generar una señal de salida que representa el uno o más canales en el bloque de audio respectivo.

Las diversas características de la presente invención y sus realizaciones preferentes pueden entenderse mejor haciendo referencia a la siguiente discusión y los dibujos adjuntos en los que los mismos números de referencia se refieren a los mismos elementos en las varias figuras. Los contenidos de la siguiente discusión y los dibujos se presentan únicamente como ejemplos y no debería entenderse que representan limitaciones sobre el ámbito de la presente invención.

### Breve descripción de los dibujos

La figura 1 es un diagrama de bloques esquemático de implementaciones ejemplares de un codificador.

La figura 2 es un diagrama de bloques esquemático de implementaciones ejemplares de un descodificador.

Las figuras 3A y 3B son ilustraciones esquemáticas de tramas en trenes de bits que cumplen las estructuras sintácticas estándar y mejoradas.

Las figuras 4A y 4B son ilustraciones esquemáticas de bloques de audio que cumplen las estructuras sintácticas estándar y mejoradas.

Las figuras 5A a 5C son ilustraciones esquemáticas de trenes de bits ejemplares que contienen datos con extensiones de programa y de canal.

La figura 6 es un diagrama de bloques esquemático de un proceso ejemplar implementado por un descodificador que procesa datos de audio codificados en el orden del canal.

La figura 7 es un diagrama de bloques esquemático de un proceso ejemplar implementado por un descodificador que procesa datos de audio codificados en el orden del bloque.

La figura 8 es un diagrama de bloques esquemático de un dispositivo que puede usarse para implementar diversos aspectos de la presente invención.

## 5 Modos de llevar a cabo la invención

### A. Visión general del sistema de codificación

10 Las figuras 1 y 2 son diagramas de bloques esquemáticos de implementaciones ejemplares de un codificador y un descodificador para un sistema de codificación de audio en el que el descodificador puede incorporar diversos aspectos de la presente invención. Estas implementaciones se ajustan a lo que se desvela en el documento A/52B citado anteriormente.

15 El propósito del sistema de codificación es generar una representación codificada de señales de audio de entrada que puede ser grabada o transmitida y descodificada posteriormente para producir señales de audio de salida que suenan esencialmente idénticas a las señales de audio de entrada en tanto que usando una cantidad mínima de información digital para representar la señal codificada. Los sistemas de codificación que cumplen los estándares ATSC básicos son capaces de codificar y descodificar información que puede representar desde uno a los denominados 5.1 canales de señales de audio, donde se entiende que 5.1 quiere decir cinco canales que pueden transportar señales de ancho de banda completo y un canal de ancho de banda limitado que está pensado para transportar señales para efectos de baja frecuencia (LFE).

20 Las siguientes secciones describen implementaciones del codificador y el descodificador, y algunos detalles de la estructura del tren de bits codificado y los procesos de codificación y descodificación relacionados. Estas descripciones se proporcionan de manera que los diversos aspectos de la presente invención puedan describirse de manera más sucinta y entenderse más claramente.

#### 1. Codificador

30 Haciendo referencia a la implementación ejemplar de la figura 1, el codificador recibe una serie de muestras moduladas por impulsos codificados (PCM) que representan uno o más canales de entrada de señales de audio procedentes del trayecto de la señal de entrada 1, y aplica un banco de filtros de análisis 2 a la serie de muestras para generar valores digitales que representan la composición espectral de las señales de audio de entrada. Para las realizaciones que cumplen los estándares ATSC, el banco de filtros de análisis está implementado por una transformada Discreta del Coseno Modificada (MDCT) descrita en el documento A/52B. La MDCT se aplica a segmentos o bloques superpuestos de muestras para cada canal de entrada de la señal de audio para generar bloques de coeficientes de la transformada que representan la composición espectral de esa señal de canal de entrada. La MDCT es parte de un sistema de análisis/síntesis que usa funciones de ventana especialmente diseñadas y procesos de superposición/adición para cancelar el solapamiento en el dominio del tiempo. Los coeficientes de la transformada en cada bloque se expresan en forma de coma flotante de bloque (BFP) que comprende exponentes de coma flotante y mantisas. Esta descripción se refiere a datos de audio expresados como exponentes de coma flotante y mantisas porque esta forma de representación se usa en trenes de bits que cumplen los estándares ATSC; sin embargo, esta representación particular es simplemente un ejemplo de representaciones numéricas que usan factores de escala y valores a escala asociados.

45 Los exponentes de BFP para cada bloque proporcionan conjuntamente una envolvente espectral aproximada para la señal de audio de entrada. Estos exponentes son codificados por modulación delta y otras técnicas de codificación para reducir las necesidades de información, pasados al formateador 5, e introducidos en un modelo psicoacústico para estimar el umbral de enmascaramiento psicoacústico de la señal que se codifica. Los resultados del modelo son usados por el asignador de bits 3 para asignar información digital en forma de bits para la cuantificación de las mantisas de tal manera que el nivel de ruido producido por la cuantificación se mantenga por debajo del umbral de enmascaramiento psicoacústico de la señal que se codifica. El cuantificador 4 cuantifica las mantisas según las asignaciones de bits recibidas del asignador de bits 3 y pasadas al formateador 5.

50 El formateador 5 multiplexa o ensambla los exponentes codificados, las mantisas cuantificadas y otra información de control, a veces denominada como metadatos de bloque, en bloques de audio. Los datos para seis bloques de audio sucesivos se ensamblan en unidades de información digital llamadas tramas. Las propias tramas también contienen información de control o metadatos de trama. La información codificada para tramas sucesivas se genera como un tren de bits a lo largo del trayecto 6 para la grabación en un medio de almacenamiento de información o para la transmisión a lo largo de un canal de comunicación. Para los codificadores que cumplen los estándares ATSC, el formato de cada trama del tren de bits cumple la sintaxis especificada en el documento A/52B.

65 Los algoritmos de codificación usados por los codificadores típicos que cumplen los estándares ATSC son más complicados que el que se ilustra en la figura 1 y se describió anteriormente. Por ejemplo, se insertan códigos de detección de errores dentro de las tramas para permitir que un descodificador de recepción valide el tren de bits. Puede usarse una técnica de codificación conocida como conmutación de longitud de bloque, a veces denominada

de manera más sencilla como conmutación de bloques, para adaptar la resolución temporal y espectral del banco de filtros de análisis para optimizar su rendimiento con características cambiantes de la señal. Los exponentes de coma flotante pueden codificarse con resolución variable de tiempo y frecuencia. Pueden combinarse dos o más canales en una representación compuesta usando una técnica de codificación conocida como acoplamiento de canales. Otra técnica de codificación conocida como rematrización de canales puede usarse de manera adaptativa para señales de audio de dos canales. Pueden usarse técnicas de codificación adicionales que no se mencionan aquí. Unas pocas de estas otras técnicas de codificación se analizan más adelante. Se omiten muchos otros detalles de implementación porque no son necesarios para entender la presente invención. Estos detalles pueden obtenerse del documento A/52B según se desee.

## 2. Descodificador

El descodificador realiza un algoritmo de descodificación que es esencialmente el inverso del algoritmo de codificación que se realizó en el codificador. Haciendo referencia a la implementación ejemplar de la figura 2, el descodificador recibe un tren de bits codificado que representa una serie de tramas procedente del trayecto de la señal de entrada 11. El tren de bits codificado puede recuperarse de un medio de almacenamiento de información o recibirse de un canal de comunicación. El desformateador 12 desmultiplexa o desensambla la información codificada para cada trama en metadatos de trama y seis bloques de audio. Los bloques de audio se desensamblan en sus respectivos metadatos de bloque, exponentes codificados y mantisas cuantificadas. Los exponentes codificados son usados por un modelo psicoacústico en el asignador de bits 13 para asignar información digital en forma de bits para la descuantificación de las mantisas cuantificadas de la misma manera que se asignaron los bits en el codificador. El descuantificador 14 descuantifica las mantisas cuantificadas según las asignaciones de bits recibidas del asignador de bits 13 y pasa las mantisas descuantificadas al banco de filtros de síntesis 15. Los exponentes codificados son descodificados y pasados al banco de filtros de síntesis 15.

Los exponentes descodificados y las mantisas descuantificadas constituyen una representación en BFP del contenido espectral de la señal de audio de entrada tal como fue codificado por el codificador. El banco de filtros de síntesis 15 se aplica a la representación del contenido espectral para reconstruir una réplica inexacta de las señales de audio de entrada originales, que se pasa a lo largo del trayecto de la señal de salida 16. Para las realizaciones que cumplen los estándares ATSC, el banco de filtros de síntesis se implementa mediante una Transformada Discreta del Coseno Modificada Inversa (IMDCT) descrita en el documento A/52B. La IMDCT es parte de un sistema de análisis/síntesis antes mencionado brevemente que se aplica a bloques de coeficientes de la transformada para generar bloques de muestras de audio que se superponen y suman para cancelar el solapamiento en el dominio del tiempo.

El algoritmo de descodificación usado por los descodificadores típicos que cumplen los estándares ATSC es más complicado que el que se ilustra en la figura 2 y se describió anteriormente. Unas pocas técnicas de descodificación que son las inversas de las técnicas de codificación descritas anteriormente incluyen detección de errores para corrección u ocultación de errores, conmutación de longitud de bloque para adaptar la resolución temporal y espectral del banco de filtros de síntesis, desacoplamiento de canales para recuperar la información del canal de las representaciones compuestas acopladas, y operaciones de matriz para recuperación de representaciones de dos canales rematrizados. La información acerca de otras técnicas y detalles adicionales puede obtenerse del documento A/52B según se desee.

## B. Estructura del tren de bits codificado

### 1. Trama

Un tren de bits codificado que cumpla los estándares ATSC comprende una serie de unidades de información codificadas llamadas "tramas de sincronización" que a veces se denominan de manera más sencilla como tramas. Tal como se mencionó anteriormente, cada trama contiene metadatos de trama y seis bloques de audio. Cada bloque de audio contiene metadatos de bloque y exponentes de BFP codificados y mantisas para un intervalo simultáneo de uno o más canales de señales de audio. La estructura para el tren de bits estándar se ilustra esquemáticamente en la figura 3A. La estructura para un tren de bits AC-3 mejorado tal como se describe en el Anexo E del documento A/52B se ilustra en la figura 3B. La porción de cada tren de bits dentro del intervalo marcado desde SI hasta CRC es una trama.

Se incluye un patrón de bits especial o palabra de sincronización en la información de sincronización (SI) que está provista al comienzo de cada trama de manera que un descodificador pueda identificar el comienzo de una trama y mantener la sincronización de sus procesos de descodificación con el tren de bits codificado. Una sección de información de tren de bits (BSI) que sigue inmediatamente a la SI transporta parámetros que se necesitan por parte del algoritmo de descodificación para descodificar la trama. Por ejemplo, la BSI especifica el número, tipo y orden de los canales que están representados por información codificada en la trama, y la información de compresión de gama dinámica y de normalización de diálogo que ha de usarse por parte del descodificador. Cada trama contiene seis bloques de audio (AB0 a AB5) que pueden estar seguidos por datos auxiliares (AUX) si se desea. La información de detección de errores en forma de una palabra de comprobación de redundancia cíclica (CRC) está

provista al final de cada trama.

Una trama en el tren de bits AC-3 mejorado también contiene datos de trama de audio (AFRM) que contienen banderas y parámetros que pertenecen a técnicas de codificación adicionales que no están disponibles para su uso en la codificación de un tren de bits estándar. Algunas de las técnicas adicionales incluyen el uso de extensión espectral (SPX), también conocida como replicación espectral, y transformada híbrida adaptativa (AHT). Más adelante se analizan diversas técnicas de codificación.

## 2. Bloques de audio

Cada bloque de audio contiene representaciones codificadas de exponentes de BFP y mantisas cuantificadas para 256 coeficientes de la transformada, y metadatos de bloque necesarios para descodificar los exponentes codificados y las mantisas cuantificadas. Esta estructura se ilustra esquemáticamente en la figura 4A. La estructura para el bloque de audio en un tren de bits AC-3 mejorado tal como se describe en el Anexo E del documento A/52B se ilustra en la figura 4B. Aquí no se analiza una estructura de bloque de audio en una versión alternativa del tren de bits tal como se describe en el Anexo D del documento A/52B porque sus características únicas no guardan relación con la presente invención.

Algunos ejemplos de metadatos de bloque incluyen banderas y parámetros para conmutación de bloque (BLKSW), compresión de gama dinámica (DYNRNG), acoplamiento de canales, rematrización de canales (REMAT), técnica o estrategia de codificación de exponentes (EXOSTR) usada para codificar los exponentes de BFP, los exponentes de BFP codificados (EXP), información de asignación de bits (BA) para las mantisas, ajustes para asignación de bits conocida como información de asignación de bits delta (DBA), y las mantisas cuantificadas (MANT). Cada bloque de audio en un tren de bits AC-3 mejorado puede contener información para técnicas de codificación adicionales incluyendo la extensión espectral (SPX).

## 3. Limitaciones del tren de bits

Los estándares ATSC imponen algunas limitaciones sobre los contenidos del tren de bits que guardan relación con la presente invención. Aquí se mencionan dos limitaciones: (1) el primer bloque de audio de la trama, el cual se denomina AB0, debe contener toda la información necesaria por el algoritmo de descodificación para empezar a descodificar todos los bloques de audio de la trama, y (2) siempre que el tren de bits empiece a transportar información codificada generada por acoplamiento de canales, el bloque de audio en el cual se usa en primer lugar acoplamiento de canales debe contener todos los parámetros necesarios para el desacoplamiento. Estas características se analizan más adelante. La información acerca de otros procesos no analizados aquí puede obtenerse del documento A/52B.

## C. Procesos y técnicas de codificación estándar

Los estándares ATSC describen un número de características sintácticas del tren de bits en cuanto a los procesos de codificación o "herramientas de codificación" que deben usarse para generar un tren de bits codificado. Un codificador no necesita emplear todas las herramientas de codificación pero un descodificador que cumpla el estándar debe poder responder a las herramientas de codificación que se consideren esenciales para la conformidad. Esta respuesta se implementa llevando a cabo una herramienta de descodificación apropiada que es esencialmente la inversa de la herramienta de codificación correspondiente.

Algunas de las herramientas de descodificación son particularmente relevantes para la presente invención porque su uso o falta de uso afecta a cómo deberían implementarse los aspectos de la presente invención. En los siguientes párrafos se analizan brevemente unos pocos procesos de descodificación y unas pocas herramientas de descodificación. Se omiten diversos detalles y características opcionales. Las descripciones sólo pretenden proporcionar una introducción de alto nivel para quienes no están familiarizados con las técnicas y para refrescar la memoria de quienes pueden haber olvidado qué técnicas describen estos términos.

Si se desea, pueden obtenerse detalles adicionales del documento A/52B, y de la patente de EE.UU. 5.583.962 titulada "Encoder/Decoder for Multi-Dimensional Sound Fields", de Davis y col., la cual fue concedida el 10 de diciembre de 1996.

### 1. Desempaquetado del tren de bits

Todos los descodificadores deben desempaquetar o desmultiplexar el tren de bits codificado para obtener parámetros y datos codificados. Este proceso está representado por el desformateador 12 analizado anteriormente. Este proceso es esencialmente uno que lee datos en el tren de bits entrante y copia porciones del tren de bits en registros, copia porciones en ubicaciones de memoria, o almacena punteros u otras referencias a los datos en el tren de bits que se almacenan en una memoria intermedia. Se requiere memoria para almacenar los datos y punteros y puede alcanzarse un compromiso entre el almacenamiento de esta información para uso posterior o la relectura del tren de bits para obtener la información siempre que sea necesario.

## 2. Decodificación de exponentes

Los valores de todos los exponentes de BFP son necesarios para desempaquetar los datos de los bloques de audio para cada trama porque estos valores indican indirectamente los números de bits que están asignados a las mantisas cuantificadas. Los valores de los exponentes del tren de bits son codificados, sin embargo, mediante técnicas de codificación diferencial que pueden aplicarse tanto a lo largo del tiempo como de la frecuencia. Como resultado, los datos que representan los exponentes codificados deben ser desempaquetados del tren de bits y descodificados antes de que puedan usarse para otros procesos de decodificación.

## 3. Procesamiento de asignación de bits

Cada una de las mantisas de BFP cuantificadas del tren de bits está representada por un número variable de bits que son una función de los exponentes de BFP y posiblemente otros metadatos contenidos en el tren de bits. Los exponentes de BFP son introducidos en un modelo especificado, el cual calcula una asignación de bits para cada mantisa. Si un bloque de audio también contiene información de asignación de bits delta (DBA), esta información adicional se usa para ajustar la asignación de bits calculada por el modelo.

## 4. Procesamiento de mantisas

Las mantisas de BFP cuantificadas constituyen la mayoría de los datos en un tren de bits codificado. La asignación de bits se usa tanto para determinar la ubicación de cada mantisa en el tren de bits para desempaquetar como para seleccionar la función de descuantificación apropiada para obtener las mantisas descuantificadas. Algunos datos del tren de bits pueden representar múltiples mantisas por un único valor. En esta situación, se deduce un número apropiado de mantisas a partir del único valor. Las mantisas que tienen una asignación igual a cero pueden reproducirse bien con un valor igual a cero o bien como un número pseudoaleatorio.

## 5. Desacoplamiento de canales

La técnica de codificación de acoplamiento de canales permite que un codificador represente múltiples canales de audio con menos datos. La técnica combina componentes espectrales procedentes de dos o más canales seleccionados, denominados como los canales acoplados, para formar un único canal de componentes espectrales compuestas, denominado como el canal de acoplamiento. Las componentes espectrales del canal de acoplamiento se representan en formato BFP. Un conjunto de factores de escala que describen la diferencia de energía entre el canal de acoplamiento y cada canal acoplado, conocidos como coordenadas de acoplamiento, se deduce para cada uno de los canales acoplados y se incluye en el tren de bits codificado. El acoplamiento se usa sólo para una porción especificada del ancho de banda de cada canal.

Cuando se usa el acoplamiento de canales, según se indica por parámetros en el tren de bits, un decodificador usa una técnica de decodificación conocida como desacoplamiento de canales para deducir una réplica inexacta de los exponentes de BFP y las mantisas para cada canal acoplado a partir de las componentes espectrales del canal de acoplamiento y las coordenadas de acoplamiento. Esto se hace multiplicando cada componente espectral del canal acoplado por la coordenada de acoplamiento apropiada. Pueden obtenerse detalles adicionales del documento A/52B.

## 6. Rematrización de canales

La técnica de codificación de rematrización de canales permite que un codificador represente señales de dos canales con menos datos usando una matriz para convertir dos canales de audio independientes en canales suma y diferencia. El exponente de BFP y las mantisas normalmente empaquetados dentro de un tren de bits para los canales de audio izquierdo y derecho representan en cambio los canales suma y diferencia. Esta técnica puede usarse ventajosamente cuando los dos canales tienen un alto grado de similitud.

Cuando se usa la rematrización, tal como se indica por una bandera en el tren de bits, un decodificador obtiene valores que representan los dos canales de audio aplicando una matriz apropiada a los valores de la suma y la diferencia. Pueden obtenerse detalles adicionales del documento A/52B.

## D. Procesos y técnicas de codificación mejorada

El Anexo E del documento A/52B describe características de la sintaxis del tren de bits AC-3 mejorado que permite el uso de herramientas de codificación adicionales. A continuación se describen brevemente unas pocas de estas herramientas y los procesos relacionados.

### 1. Procesamiento de transformada híbrida adaptativa

La técnica de codificación de transformada híbrida adaptativa (AHT) proporciona otra herramienta además de la

conmutación de bloques para adaptar la resolución temporal y espectral de los bancos de filtros de análisis y síntesis en respuesta a las características cambiantes de la señal aplicando dos transformadas en cascada. La información adicional para el procesamiento AHT puede obtenerse del documento A/52B y la patente de EE.UU. 7.516.064 titulada "Adaptive Hybrid Transform for Signal Analysis and Synthesis" de Vinton y col., la cual fue concedida el 7 de abril de 2009.

Los codificadores emplean una transformada primaria implementada por la transformada de análisis MDCT mencionada anteriormente delante de, y en cascada con una transformada secundaria implementada por una Transformada Discreta del Coseno de Tipo II (DCT-II) La MDCT se aplica a bloques superpuestos de muestras de la señal de audio para generar coeficientes espectrales que representan el contenido espectral de la señal de audio. La DCT-II puede conectarse y desconectarse del trayecto de procesamiento de la señal según se desee y, cuando está conectada, se aplica a bloques no superpuestos de los coeficientes espectrales de la MDCT que representan la misma frecuencia para generar los coeficientes de la transformada híbrida. En el uso típico, la DCT-II se activa cuando se considera que la señal de audio de entrada es suficientemente estacionaria porque su uso aumenta significativamente la resolución espectral efectiva del banco de filtros de análisis disminuyendo su resolución temporal efectiva de 256 muestras a 1536 muestras.

Los descodificadores emplean una transformada primaria inversa implementada por el banco de filtros de síntesis de IMDCT mencionado anteriormente que sigue y están en cascada con una transformada secundaria inversa implementada por una Transformada Discreta del Coseno Inversa de Tipo II (IDCT-II). La IDCT-II se conecta y desconecta del trayecto de procesamiento de la señal en respuesta a los metadatos proporcionados por el codificador. Cuando está conectada, la IDCT-II se aplica a bloques no superpuestos de los coeficientes de la transformada híbrida para obtener los coeficientes de la transformada secundaria inversa. Los coeficientes de la transformada secundaria inversa pueden ser coeficientes espectrales para introducción directa en la IMDCT si se usó ninguna otra herramienta de codificación como acoplamiento de canales o SPX. Alternativamente, los coeficientes espectrales de la MDCT pueden deducirse a partir de los coeficientes de la transformada secundaria inversa si se usaron herramientas de codificación como acoplamiento de canales o SPX. Después de obtenerse los coeficientes espectrales de la MDCT, la IMDCT se aplica a bloques de los coeficientes espectrales de la MDCT de manera convencional.

La AHT puede usarse con cualquier canal de audio incluyendo el canal de acoplamiento y el canal de LFE. Un canal que es codificado usando la AHT usa un proceso de asignación de bits alternativo y dos tipos de cuantificación diferentes. Un tipo es la cuantificación vectorial (VQ) y el segundo tipo es la cuantificación adaptativa de ganancia (GAQ). La técnica GAQ se analiza en la patente de EE.UU. 6.246.345, titulada "Using Gain-Adaptive Quantization and Non-Uniform Symbol Lengths for Improved Audio Coding", de Davidson y col., que fue concedida el 12 de junio de 2001.

El uso de la AHT requiere que un descodificador deduzca varios parámetros a partir de información contenida en el tren de bits codificado. El documento A/52B describe cómo pueden calcularse estos parámetros. Un conjunto de parámetros especifican el número de veces que son transportados los exponentes de BFP en una trama y se deducen examinando los metadatos contenidos en todos los bloques de audio de una trama. Otros dos conjuntos de parámetros identifican qué mantisas de BFP son cuantificadas usando GAQ y proporcionan palabras de control de ganancia para los cuantificadores y se deducen examinando los metadatos para un canal en un bloque de audio.

Todos los coeficientes de la transformada híbrida para AHT están contenidos en el primer bloque de audio, AB0, de una trama. Si la AHT se aplica a un canal de acoplamiento, las coordenadas de acoplamiento para los coeficientes de la AHT se distribuyen por todos los bloques de audio de la misma manera que para los canales acoplados sin AHT. Más adelante se describe un proceso para ocuparse de esta situación.

## 2. Procesamiento de extensión espectral

La técnica de codificación de extensión espectral (SPX) permite que un codificador reduzca la cantidad de información necesaria para codificar un canal de ancho de banda completo excluyendo las componentes espectrales de alta frecuencia del tren de bits codificado y haciendo que el descodificador sintetice las componentes espectrales desaparecidas de las componentes espectrales de frecuencia más baja que están contenidas en el tren de bits codificado.

Cuando se usa SPX, el descodificador sintetiza las componentes espectrales desaparecidas copiando los coeficientes de la MDCT de frecuencia más baja en ubicaciones de coeficientes de la MDCT de frecuencia más alta, añadiendo valores pseudoaleatorios o ruido a los coeficientes de la transformada copiados, y cambiando la escala de la amplitud según una envolvente espectral de SPX incluida en el tren de bits codificado. El codificador calcula la envolvente espectral de SPX y la inserta en el tren de bits codificado siempre que se use la herramienta de codificación SPX.

La técnica SPX se usa típicamente para sintetizar las bandas más altas de componentes espectrales para un canal. Puede usarse junto con acoplamiento de canales para un intervalo medio de frecuencias. Detalles adicionales del



procesamiento pueden obtenerse del documento A/52B.

### 3. Extensiones de canales y de programas

5 La sintaxis del tren de bits AC-3 mejorado permite que un codificador genere un tren de bits codificado que representa un único programa con más de 5.1 canales (extensión de canales), dos o más programas con hasta 5.1 canales (extensión de programas), o una combinación de programas con hasta 5.1 canales y más de 5.1 canales. La extensión de programas se implementa mediante un multiplex de tramas para múltiples trenes de datos independientes en un tren de bits codificado. La extensión de canales se implementa mediante un multiplex de tramas para uno o más subtrenes de datos dependientes que están asociados con un tren de datos independiente en implementaciones preferidas para extensión de programas, se informa a un descodificador de qué programa o programas descodificar y el proceso de descodificación se salta o ignora esencialmente los trenes y subtrenes que representan programas que no han de ser descodificados.

15 Las figuras 5A a 5C ilustran tres ejemplos de trenes de bits que contienen datos con extensiones de programas y canales. La figura 5A ilustra un tren de bits ejemplar con extensión de canales. Un único programa P1 está representado por un tren independiente S0 y tres subtrenes dependientes asociados SS0, SS1 y SS2. Una trama Fn para el tren independiente S0 viene seguido inmediatamente por tramas Fn para cada uno de los subtrenes dependientes asociados SS0 a SS3. Estas tramas vienen seguidas por la siguiente trama Fn+1 para el tren independiente S0, que a su vez viene seguido inmediatamente por tramas Fn+1 para cada uno de los subtrenes dependientes asociados SS0 a SS2. La sintaxis del tren de bits AC-3 mejorado permite hasta ocho subtrenes dependientes para cada tren independiente.

25 La figura 5B ilustra un tren de bits ejemplar con extensión de programas. Cada uno de cuatro programas P1, P2, P3 y P4 está representado por trenes independientes S0, S1, S2 y S3, respectivamente. Una trama Fn para el tren independiente S0 viene seguida inmediatamente por tramas Fn para cada uno de los trenes independientes S1, S2 y S3. Estas tramas vienen seguidas por la siguiente trama Fn+1 para cada uno de los trenes independientes. La sintaxis del tren de bits AC-3 mejorado deben tener al menos un tren independiente y permite hasta ocho trenes independientes.

30 La figura 5C ilustra un tren de bits ejemplar con extensión de programas y extensión de canales. El programa P1 está representado por datos en el tren independiente S0, y el programa P2 está representado por datos en el tren independiente S1 y los subtrenes dependientes asociados SS0 y SS1. Una trama Fn para el tren independiente S0 viene seguida inmediatamente por la trama Fn para el tren independiente S1, que a su vez viene seguida inmediatamente por tramas Fn para los subtrenes dependientes asociados SS0 y SS1. Estas tramas vienen seguidas por la siguiente trama Fn+1 para cada uno de los trenes independientes y los subtrenes dependientes.

40 Un tren independiente sin extensión de canales contiene datos que pueden representar hasta 5.1 canales de audio independientes. Un tren independiente con extensión de canales o, en otras palabras, un tren independiente que tiene uno o más subtrenes dependientes asociados, contiene datos que representan una submezcla de canales 5.1 de todos los canales para el programa. El término "submezcla" se refiere a una combinación de canales dentro de un número menor de canales. Esto se hace por compatibilidad con los descodificadores que no descodifican los subtrenes dependientes. Los subtrenes dependientes contienen datos que representan canales que o bien sustituyen o bien complementan los canales contenidos en el tren independiente asociado. La extensión de canales permite hasta catorce canales para un programa.

Detalles adicionales de la sintaxis del tren de bits y el procesamiento asociado pueden obtenerse del documento A/52B.

### 50 E. Procesamiento de prioridad de bloques

Se requiere una lógica compleja para procesar y descodificar correctamente las muchas variaciones en la estructura del tren de bits que se producen cuando se usan diversas combinaciones de herramientas de codificación para generar el tren de bits codificado. Tal como se mencionó anteriormente, los detalles del diseño algorítmico no están especificados en los estándares ATSC pero una característica universal de la implementación convencional de los descodificadores E-AC-3 es un algoritmo que descodifica todos los datos de una trama para un canal respectivo antes de descodificar los datos para otro canal. Este planteamiento tradicional reduce la cantidad de memoria on-chip necesaria para descodificar el tren de bits pero también requiere múltiples pasadas sobre los datos en cada trama para leer y examinar los datos en todos los bloques de audio de la trama.

60 El planteamiento tradicional se ilustra esquemáticamente en la figura 6. La componente 19 analiza sintácticamente las tramas de un tren de bits codificado recibido del trayecto 1 y extrae los datos de las tramas en respuesta a señales de control recibidas del trayecto 20. El análisis sintáctico se logra mediante múltiples pasadas sobre los datos de la trama. Los datos extraídos de una trama están representados por casillas debajo de la componente 19. Por ejemplo, la casilla con la etiqueta AB0-CH0 representa los datos extraídos para el canal 9 en el bloque de audio AB0 y la casilla con la etiqueta AB5-CH2 representa los datos extraídos para el canal 2 en el bloque de audio AB5.

Sólo se ilustran tres canales 0 a 2 y tres bloques de audio 0, 1 y 5 para simplificar el dibujo. La componente 19 también pasa los parámetros obtenidos de metadatos de trama a lo largo del trayecto 20 al canal que procesa las componentes 31, 32 y 33. Los trayectos de la señal y los conmutadores rotatorios a la izquierda de las casillas de datos representan la lógica realizada por los descodificadores tradicionales para procesar los datos de audio codificados en orden por canal. La componente del canal de proceso 31 recibe datos y metadatos de audio codificados a través del conmutador rotatorio 21 para el canal CH0, empezando con el bloque de audio AB0 y concluyendo con el bloque de audio AB5, descodifica los datos y genera una señal de salida aplicando un banco de filtros de síntesis a los datos descodificados. Los resultados de su procesamiento se pasan a lo largo del trayecto 41. La componente del canal de proceso 32 recibe datos para el canal CH1 para los bloques de audio AB0 a AB5 a través del conmutador rotatorio 22, procesa los datos y pasa su salida a lo largo del trayecto 42. La componente del canal de proceso 33 recibe datos para el canal CH2 para los bloques de audio AB0 a AB5 a través del conmutador rotatorio 23, procesa los datos y pasa su salida a lo largo del trayecto 43.

Las aplicaciones de la presente invención pueden mejorar la eficiencia de procesamiento eliminando múltiples pasadas sobre los datos de la trama en muchas situaciones. Se usan múltiples pasadas en algunas situaciones cuando se usan ciertas combinaciones de herramientas de codificación para generar el tren de bits codificado; sin embargo, los trenes de bits AC-3 mejorado generados por las combinaciones de herramientas de codificación analizadas más adelante pueden ser descodificados con una única pasada. Este nuevo planteamiento se ilustra esquemáticamente en la figura 7. La componente 19 analiza sintácticamente las tramas de un tren de bits codificado recibido del trayecto 1 y extrae los datos de las tramas en respuesta a señales de control recibidas del trayecto 20. En muchas situaciones, el análisis sintáctico se consigue mediante una única pasada sobre los datos de la trama. Los datos extraídos de una trama están representados por las casillas debajo de la componente 19 de la misma manera analizada anteriormente para la figura 6. La componente 19 pasa los parámetros obtenidos de metadatos de trama a lo largo del trayecto 20 al bloque que procesa las componentes 61, 62 y 63. La componente del bloque de proceso 61 recibe datos y metadatos de audio codificados a través del conmutador rotatorio 51 para todos los canales del bloque de audio AB0, descodifica los datos y genera una señal de salida aplicando un banco de filtros de síntesis a los datos descodificados. Los resultados de su procesamiento para los canales CH0, CH1 y CH2 se pasan a través del conmutador rotatorio 71 al trayecto de salida apropiado 41, 42 y 43, respectivamente. La componente del bloque de proceso 62 recibe datos para todos los canales del bloque de audio AB1 a través del conmutador rotatorio 52, procesa los datos y pasa su salida a través del conmutador rotatorio 72 al trayecto de salida apropiado para cada canal. La componente del bloque de proceso 63 recibe datos para todos los canales del bloque de audio AB5 a través del conmutador rotatorio 53, procesa los datos y pasa su salida a través del conmutador rotatorio 73 al trayecto de salida apropiado para cada canal.

Más adelante se analizan diversos aspectos de la presente invención y se ilustran con fragmentos de programa. Estos fragmentos de programa no pretenden ser implementaciones prácticas u óptimas sino sólo ejemplos ilustrativos. Por ejemplo, el orden de las sentencias del programa puede alterarse intercambiando algunas de las sentencias.

#### 1. Proceso general

En el siguiente fragmento de programa se muestra una ilustración de alto nivel de la presente invención:

```

(1.1)  determinar el comienzo de una trama del tren de bits S
(1.2)  para cada trama N del tren de bits S
(1.3)      desempaquetar los metadatos de la trama N
(1.4)      obtener los parámetros a partir de los metadatos de trama desempaquetados
(1.5)      determinar el comienzo del primer bloque de audio K de la trama N
(1.6)      para el bloque de audio K de la trama N
(1.7)          desempaquetar los metadatos del bloque K
(1.8)          obtener los parámetros a partir de los metadatos de bloque desempaquetados
(1.9)          determinar el comienzo del primer canal C del bloque K
(1.10)         para el canal C del bloque K
(1.11)             desempaquetar y descodificar los exponentes
(1.12)             desempaquetar y descuantificar las mantisas
(1.13)             aplicar el filtro de síntesis a los datos de audio descodificados para el canal C
(1.14)             determinar el comienzo del canal C+1 del bloque K
(1.15)         fin para
(1.16)         determinar el comienzo del bloque K+1 de la trama N
(1.17)     fin para
(1.18)     determinar el comienzo de la siguiente trama N+1 del tren de bits S
(1.19) fin para
    
```

La sentencia (1.1) escanea el tren de bits en busca de una cadena de bits que coincida con el patrón de sincronización transportado en la información de SI. Cuando se encuentra el patrón de sincronización, se ha determinado el comienzo de una trama en el tren de bits.

Las sentencias (1.2) y (1.19) controlan el proceso de descodificación que ha de realizarse para cada trama del tren de bits, o hasta que el proceso de descodificación sea detenido por algún otro medio. Las sentencias (1.3) a (1.18) realizan procesos que descodifican una trama en el tren de bits codificado.

5 Las sentencias (1.3) a (1.5) desempaquetan los metadatos de la trama, obtienen parámetros de descodificación de los metadatos desempaquetados, y determinan la ubicación en el tren de bits donde empiezan los datos para el primer bloque de audio K de la trama. La sentencia (1.16) determina el comienzo del siguiente bloque de audio en el tren de bits si en la trama está algún bloque de audio subsiguiente.

10 Las sentencias (1.6) y (1.17) hacen que el proceso de descodificación se realice para cada bloque de audio de la trama. Las sentencias (1.7) a (1.15) realizan procesos que descodifican un bloque de audio de la trama. Las sentencias (1.7) a (1.9) desempaquetan los metadatos del bloque de audio, obtienen parámetros de descodificación de los metadatos desempaquetados, y determinan dónde empiezan los datos para el primer canal.

15 Las sentencias (1.10) y (1.15) hacen que el proceso de descodificación se realice para cada canal del bloque de audio. Las sentencias (1.11) a (1.13) desempaquetan y descodifican los exponentes, usan los exponentes descodificados para determinar la asignación de bits para desempaquetar y descuantificar cada mantisa cuantificada, y aplican el banco de filtros de síntesis a las mantisas descuantificadas. La sentencia (1.14) determina la ubicación en el tren de bits donde comienzan los datos para el siguiente canal, si en la trama está algún canal subsiguiente.

20 La estructura del proceso varía para adaptarse a diferentes técnicas de codificación usadas para generar el tren de bits codificado. En fragmentos de programa de más adelante se analizan e ilustran varias variaciones. Las descripciones de los siguientes fragmentos de programa omiten algunos de los detalles que se describen para el fragmento de programa precedente.

## 2. Extensión espectral

30 Cuando se usa extensión espectral (SPX), el bloque de audio en el cual empieza el proceso de extensión contiene parámetros compartidos necesarios para la SPX en el bloque de audio inicial así como otros bloques de audio que usan SPX en la trama. Los parámetros compartidos incluyen una identificación de los canales que participan en el proceso, el intervalo de frecuencia de extensión espectral, y cómo se comparte la envolvente espectral de SPX para cada canal a lo largo del tiempo y la frecuencia. Estos parámetros son desempaquetados del bloque de audio que empieza el uso de SPX y almacenados en memoria o en registros informáticos para su uso al procesar la SPX en los bloques de audio subsiguientes en la trama.

35 Es posible que una trama tenga más de un bloque de audio inicial para la SPX. Un bloque de audio empieza la SPX si los metadatos para ese bloque de audio indican que se usa SPX y, o bien los metadatos para el bloque de audio precedente en la trama indican que no se usa SPX, o bien el bloque de audio es el primer bloque de una trama.

40 Cada bloque de audio que usa SPX o bien incluye la envolvente espectral de SPX, denominada como coordenadas de SPX, que se usan para el procesamiento de extensión espectral en ese bloque de audio, o bien incluye una bandera de "reutilización" que indica que han de usarse las coordenadas de SPX para un bloque previo. Las coordenadas de SPX en un bloque son desempaquetadas y conservadas para su posible reutilización por operaciones de SPX en bloques de audio subsiguientes.

El siguiente fragmento de programa ilustra un modo en que pueden procesarse los bloques de audio usando SPX.

50 (2.1) determinar el comienzo de una trama del tren de bits S  
 (2.2) para cada trama N del tren de bits S  
 (2.3) desempaquetar los metadatos de la trama N  
 (2.4) obtener los parámetros a partir de los metadatos de trama desempaquetados  
 (2.5) si están presentes parámetros de trama SPX entonces desempaquetar parámetros de trama SPX  
 55 (2.6) determinar el comienzo del primer bloque de audio K de la trama N  
 (2.7) para el bloque de audio K de la trama N  
 (2.8) desempaquetar los metadatos del bloque K  
 (2.9) obtener los parámetros a partir de los metadatos de bloque desempaquetados  
 (2.10) si están presentes parámetros de bloque SPX entonces desempaquetar parámetros de bloque SPX  
 60 (2.11) para el canal C del bloque K  
 (2.12) desempaquetar y descodificar los exponentes  
 (2.13) desempaquetar y descuantificar las mantisas  
 (2.14) si el canal C usa SPX entonces  
 (2.15) extender ancho de banda del canal C  
 65 (2.16) fin si  
 (2.17) aplicar el filtro de síntesis a los datos de audio descodificados para el canal C

- (2.18) determinar el comienzo del canal C+1 del bloque K
- (2.19) fin para
- (2.20) determinar el comienzo del bloque K+1 de la trama N
- (2.21) fin para
- 5 (2.22) determinar el comienzo de la siguiente trama N+1 del tren de bits S
- (2.23) fin para

10 La sentencia (2.5) desempaqueta los parámetros de la trama SPX a partir de los metadatos de trama si está presente alguno en esos metadatos. La sentencia (2.10) desempaqueta los parámetros del bloque SPX a partir de los metadatos de bloque si está presente alguno en los metadatos de bloque. Los parámetros SPX del bloque pueden incluir las coordenadas de SPX para uno o más canales del bloque.

15 Las sentencias (2.12) y (2.13) desempaquetan y descodifican los exponentes y usan los exponentes descodificados para determinar la asignación de bits para desempaquetar y descuantificar cada mantisa cuantificada. La sentencia (2.14) determina si el canal C del bloque de audio actual usa SPX. Si usa SPX, la sentencia (2.15) aplica procesamiento SPX para extender el ancho de banda del canal C. Este proceso proporciona las componentes espectrales para el canal C que se introducen en el banco de filtros de síntesis aplicado en la sentencia (2.17).

### 20 3. Transformada híbrida adaptativa

Cuando se usa la transformada híbrida adaptativa (AHT), el primer bloque de audio AB0 de una trama contiene todos los coeficientes de la transformada híbrida para cada canal procesado por la transformada DCT-II. Para todos los demás canales, cada uno de los seis bloques de audio de la trama contiene hasta 256 coeficientes espectrales generados por el banco de filtros de análisis MDCT.

25 Por ejemplo, un tren de bits codificado contiene datos para los canales izquierdo, central y derecho. Cuando los canales izquierdo y derecho son procesados por la AHT y el canal central no es procesado por la AHT, el bloque de audio AB0 contiene todos los coeficientes de la transformada híbrida para cada uno de los canales izquierdo y derecho y contiene hasta 256 coeficientes espectrales de MDCT para el canal central. Los bloques de audio AB1 a AB5 contienen coeficientes espectrales de MDCT para el canal central y no contienen coeficientes para los canales izquierdo y derecho.

El siguiente fragmento de programa ilustra un modo en que pueden procesarse los bloques de audio con coeficientes de la AHT.

- 35 (3.1) determinar el comienzo de una trama del tren de bits S
- (3.2) para cada trama N del tren de bits S
- (3.3) desempaquetar los metadatos de la trama N
- (3.4) obtener los parámetros a partir de los metadatos de trama desempaquetados
- 40 (3.5) determinar el comienzo del primer bloque de audio K de la trama N
- (3.6) para el bloque de audio K de la trama N
- (3.7) desempaquetar los metadatos del bloque K
- (3.8) obtener los parámetros a partir de los metadatos de bloque desempaquetados
- (3.9) determinar el comienzo del primer canal C del bloque K
- 45 (3.10) para el canal C del bloque K
- (3.11) si AHT está en uso para el canal C entonces
- (3.12) si K=0 entonces
- (3.13) desempaquetar y descodificar los exponentes
- (3.14) desempaquetar y descuantificar las mantisas
- 50 (3.15) aplicar la transformada secundaria inversa a los exponentes y las mantisas
- (3.16) almacenar exponentes y mantisas de MDTC en la memoria intermedia
- (3.17) fin si
- (3.18) obtener de la memoria intermedia exponentes y mantisas de MDTC para el bloque el K
- (3.19) si no
- 55 (3.20) desempaquetar y descodificar los exponentes
- (3.21) desempaquetar y descuantificar las mantisas
- (3.22) fin si
- (3.23) aplicar el filtro de síntesis a los datos de audio descodificados para el canal C
- (3.24) determinar el comienzo del canal C+1 del bloque K
- 60 (3.25) fin para
- (3.26) determinar el comienzo del bloque K+1 de la trama N
- (3.27) fin para
- (3.28) determinar el comienzo de la siguiente trama N+1 del tren de bits S
- (3.29) fin para
- 65

La sentencia (3.11) determina si la AHT está en uso para el canal C. Si está en uso, la sentencia (3.12) determina si

5 el primer bloque de audio ABO está siendo procesado. Si el primer bloque de audio está siendo procesado, entonces las sentencias (3.13) a (3.16) obtienen todos los coeficientes de la AHT para el canal C, aplican la transformada secundaria inversa o IDCT-II a los coeficientes de la AHT para obtener los coeficientes espectrales de la MDCT, y los almacenan en una memoria intermedia. Estos coeficientes espectrales corresponden a los exponentes y las mantisas descuantificadas que se obtienen mediante las sentencias (3.20) y (3.21) para los canales para los cuales no están en uso la AHT. La sentencia (3.18) obtiene los exponentes y las mantisas de los coeficientes espectrales de la MDCT que corresponden al bloque de audio K que está siendo procesado. Si el primer bloque de audio (K=0) está siendo procesado, por ejemplo, entonces se obtienen de la memoria intermedia los exponentes y las mantisas para el conjunto de coeficientes espectrales de MDCT para el primer bloque. Si el segundo bloque de audio (K=1) está siendo procesado, por ejemplo, entonces se obtienen de la memoria intermedia los exponentes y las mantisas para el conjunto de coeficientes espectrales de la MDCT para el segundo bloque.

#### 4. Extensión espectral y transformada híbrida adaptativa

15 La SPX y la AHT pueden usarse para generar datos codificados para los mismos canales. La lógica analizada anteriormente por separado para el procesamiento de extensión espectral y transformada híbrida puede combinarse para procesar canales para los cuales está en uso la SPX, está en uso la AHT, o están en uso tanto la SPX como la AHT.

20 El siguiente fragmento de programa ilustra un modo en que pueden procesarse los bloques de audio con coeficientes de la SPX y la AHT.

```

(4.1)  comienzo de una trama del tren de bits S
(4.2)  para cada trama N del tren de bits S
25  (4.3)  desempaquetar los metadatos de la trama N
      (4.4)  obtener los parámetros a partir de los metadatos de trama desempaquetados
      (4.5)  si están presentes parámetros de trama SPX entonces desempaquetar parámetros de trama SPX
      (4.6)  determinar el comienzo del primer bloque de audio K de la trama N
      (4.7)  para el bloque de audio K de la trama N
30  (4.8)  desempaquetar los metadatos del bloque K
      (4.9)  obtener los parámetros a partir de los metadatos de bloque desempaquetados
      (4.10) si están presentes parámetros de bloque SPX entonces desempaquetar parámetros de bloque SPX
      (4.11) para el canal C del bloque K
      (4.12) si está en uso AHT para el canal C entonces
35  (4.13) si K=0 entonces
      (4.14) desempaquetar y descodificar los exponentes
      (4.15) desempaquetar y descuantificar las mantisas
      (4.16) aplicar la transformada secundaria inversa a los exponentes y las mantisas
      (4.17) almacenar exponentes y mantisas de transformada secundaria inversa en la memoria intermedia
40  (4.18) fin si
      (4.19) obtener de la memoria intermedia exponentes y mantisas de transformada secundaria inversa para el bloque K
      (4.20) si no
      (4.21) desempaquetar y descodificar los exponentes
      (4.22) desempaquetar y descuantificar las mantisas
45  (4.23) fin si
      (4.24) si el canal C usa SPX entonces
      (4.25) extender ancho de banda del canal C
      (4.26) fin si
      (4.27) aplicar el filtro de síntesis a los datos de audio descodificados para el canal C
50  (4.28) determinar el comienzo del canal C+1 del bloque K
      (4.29) fin para
      (4.30) determinar el comienzo del bloque K+1 de la trama N
      (4.31) fin para
      (4.32) determinar el comienzo de la siguiente trama N+1 del tren de bits S
55  (4.33) fin para

```

60 La sentencia (4.5) desempaqueta los parámetros de la trama SPX a partir de los metadatos de trama si está presente alguno en esos metadatos. La sentencia (4.10) desempaqueta los parámetros del bloque SPX a partir de los metadatos de bloque si está presente alguno en los metadatos de bloque. Los parámetros SPX del bloque pueden incluir las coordenadas de SPX para uno o más canales del bloque.

65 La sentencia (4.12) determina si la AHT está en uso para el canal C. Si la AHT está en uso para el canal C, la sentencia (4.13) determina si este es el primer bloque de audio. Si es el primer bloque de audio, las sentencias (4.14) a (4.17) obtienen todos los coeficientes de la AHT para el canal C, aplican la transformada secundaria inversa o la IDCT-II a los coeficientes de la AHT para obtener los coeficientes de la transformada secundaria inversa, y los almacenan en una memoria intermedia. La sentencia (4.19) obtiene los exponentes y las mantisas de los

coeficientes de la transformada secundaria inversa que corresponden al bloque de audio K que está siendo procesado.

5 Si la AHT no está en uso para el canal C, las sentencias (4.21) y (4.22) desempaquetan y obtienen los exponentes y las mantisas para el canal C del bloque K tal como se analizó anteriormente para las sentencias de programa (1.11) y (1.12).

10 La sentencia (4.24) determina si el canal C del bloque de audio actual usa SPX. Si usa SPX, la sentencia (4.25) aplica procesamiento SPX a los coeficientes de la transformada secundaria inversa para extender el ancho de banda, obteniendo así los coeficientes espectrales de la MDCT del canal C. Este proceso proporciona las componentes espectrales para el canal C que se introducen en el banco de filtros de síntesis aplicado en la sentencia (4.27). Si no se usa el procesamiento SPX para el canal C, los coeficientes espectrales de la MDCT se obtienen directamente de los coeficientes de la transformada secundaria inversa.

15 5. Acoplamiento y transformada híbrida adaptativa

El acoplamiento de canales y la AHT pueden usarse para generar datos codificados para los mismos canales. Esencialmente, la misma lógica analizada anteriormente para el procesamiento de extensión espectral y transformada híbrida puede usarse para procesar trenes de bits usando acoplamiento de canales y la AHT porque los detalles del procesamiento SPX analizados anteriormente se aplican al procesamiento realizado para acoplamiento de canales.

El siguiente fragmento de programa ilustra un modo en que pueden procesarse los bloques de audio con acoplamiento y coeficientes de AHT.

25 (5.1) comienzo de una trama del tren de bits S  
 (5.2) para cada trama N del tren de bits S  
 (5.3) desempaquetar los metadatos de la trama N  
 (5.4) obtener los parámetros a partir de los metadatos de trama desempaquetados  
 30 (5.5) si están presentes parámetros de trama de acoplamiento entonces desempaquetar parámetros de trama de acoplamiento  
 (5.6) determinar el comienzo del primer bloque de audio K de la trama N  
 (5.7) para el bloque de audio K de la trama N  
 (5.8) desempaquetar los metadatos del bloque K  
 (5.9) obtener los parámetros a partir de los metadatos de bloque desempaquetados  
 35 (5.10) si están presentes parámetros de bloque de acoplamiento entonces desempaquetar parámetros de bloque de acoplamiento  
 (5.11) para el canal C del bloque K  
 (5.12) si está en uso AHT para el canal C entonces  
 (5.13) si K=0 entonces  
 (5.14) desempaquetar y descodificar los exponentes  
 40 (5.15) desempaquetar y descuantificar las mantisas  
 (5.16) aplicar la transformada secundaria inversa a los exponentes y las mantisas  
 (5.17) almacenar exponentes y mantisas de transformada secundaria inversa en la memoria intermedia  
 (5.18) fin si  
 (5.19) obtener de la memoria intermedia exponentes y mantisas de transformada secundaria inversa para el bloque K  
 45 (5.20) si no  
 (5.21) desempaquetar y descodificar los exponentes para el canal C  
 (5.22) desempaquetar y descuantificar las mantisas para el canal C  
 (5.23) fin si  
 (5.24) si el canal C usa acoplamiento entonces  
 50 (5.25) si el canal C es el primer canal en usar acoplamiento entonces  
 (5.26) si está en uso AHT para el canal de acoplamiento entonces  
 (5.27) si K=0 entonces  
 (5.28) desempaquetar y descodificar los exponentes del canal de acoplamiento  
 (5.29) desempaquetar y descuantificar las mantisas del canal de acoplamiento  
 55 (5.30) aplicar la transformada secundaria inversa al canal de acoplamiento  
 (5.31) almacenar exponentes y mantisas de canal de acoplamiento de transformada secundaria inversa en la memoria intermedia  
 (5.32) fin si  
 (5.33) obtener de la memoria intermedia exponentes y mantisas de canal de acoplamiento para el bloque K  
 60 (5.34) si no  
 (5.35) desempaquetar y descodificar los exponentes del canal de acoplamiento  
 (5.36) desempaquetar y descuantificar las mantisas del canal de acoplamiento  
 (5.37) fin si  
 (5.38) fin si  
 65 (5.39) obtener el canal acoplado C del canal de acoplamiento  
 (5.40) fin si

- (5.41) aplicar el filtro de síntesis a los datos de audio descodificados para el canal C
- (5.42) determinar el comienzo del canal C+1 del bloque K
- (5.43) fin para
- (5.44) determinar el comienzo del canal C+1 del bloque K
- 5 (5.45) fin para
- (5.46) determinar el comienzo de la siguiente trama N+1 del tren de bits S
- (5.47) fin para

10 La sentencia (5.5) desempaqueta los parámetros de acoplamiento de canales a partir de los metadatos de trama si está presente alguno en esos metadatos. La sentencia (5.10) desempaqueta los parámetros de acoplamiento de canales a partir de los metadatos de bloque si está presente alguno en los metadatos de bloque. Si están presentes, se obtienen las coordenadas de acoplamiento para los canales acoplados en el bloque.

15 La sentencia (5.12) determina si la AHT está en uso para el canal C. Si la AHT está en uso, la sentencia (5.13) determina si es el primer bloque de audio. Si es el primer bloque de audio, las sentencias (5.14) a (5.17) obtienen todos los coeficientes de la AHT para el canal C, aplican la transformada secundaria inversa o la IDCT-II a los coeficientes de la AHT para obtener los coeficientes de la transformada secundaria inversa, y los almacenan en una memoria intermedia. La sentencia (5.19) obtiene los exponentes y las mantisas de los coeficientes de la transformada secundaria inversa que corresponden al bloque de audio K que está siendo procesado.

20 Si la AHT no está en uso para el canal C, las sentencias (5.21) y (5.22) desempaquetan y obtienen los exponentes y las mantisas para el canal C del bloque K tal como se analizó anteriormente para las sentencias de programa (1.11) y (1.12).

25 La sentencia (5.24) determina si el canal de acoplamiento está en uso para el canal C. Si está en uso, la sentencia (5.25) determina si el canal C es el primer canal del bloque en usar acoplamiento. Si lo es, se obtienen los exponentes y las mantisas para el canal de acoplamiento o bien a partir de una aplicación de una transformada secundaria inversa a los exponentes y las mantisas del canal de acoplamiento tal como se muestra en las sentencias (5.26) a (5.33), o bien a partir de los datos del tren de bits tal como se muestra en las sentencias (5.35) y (5.36). Los datos que representan las mantisas del canal de acoplamiento están situados en el tren de bits inmediatamente después de los datos que representan las mantisas del canal C. La sentencia (5.39) deduce el canal acoplado C a partir del canal de acoplamiento usando las coordenadas de acoplamiento apropiadas para el canal C. Si no se usa acoplamiento de canales para el canal C, los coeficientes espectrales de la MDCT se obtienen directamente de los coeficientes de la transformada secundaria inversa.

35 6. Extensión espectral, acoplamiento y transformada híbrida adaptativa

La extensión espectral, el acoplamiento de canales y la AHT pueden usarse todos para generar datos codificados para los mismos canales. La lógica analizada anteriormente para combinaciones de procesamiento AHT con extensión espectral y acoplamiento puede combinarse para procesar canales usando cualquier combinación de las tres herramientas de codificación incorporando la lógica adicional necesaria para ocuparse de ocho situaciones posibles. El procesamiento para desacoplamiento de canales se realiza antes de realizar el procesamiento SPX.

45 F. Implementación

Los dispositivos que incorporan diversos aspectos de la presente invención pueden implementarse de diversas maneras incluyendo software para ejecución por un ordenador o algún otro dispositivo que incluya componentes más especializados tal como un sistema de circuitos de procesador de señales digitales (DSP) acoplado a componentes similares a los que se encuentran en un ordenador de propósito general. La figura 8 es un diagrama de bloques esquemático de un dispositivo 90 que puede usarse para implementar aspectos de la presente invención. El procesador 92 proporciona recursos informáticos. La RAM 93 es la memoria de acceso aleatorio (RAM) del sistema usada por el procesador 92 para el procesamiento. La ROM 94 representa alguna forma de almacenamiento persistente tal como memoria de sólo lectura (ROM) para almacenar los programas necesarios para operar el dispositivo 90 y posiblemente para llevar a cabo diversos aspectos de la presente invención. El control de I/O (entrada/salida) 95 representa el sistema de circuitos de interconexión para recibir un transmitir señales por medio de los canales de comunicación 1, 16. En la realización mostrada, todos los componentes principales del sistema se conectan al bus 91, el cual puede representar más de un bus físico o lógico; sin embargo, no se requiere una arquitectura de bus para implementar la presente invención.

60 En las realizaciones implementadas por un sistema de ordenador de propósito general, pueden incluirse componentes adicionales para interconexión a dispositivos tales como un teclado o un ratón y una pantalla, y para controlar un dispositivo de almacenamiento que tiene un medio de almacenamiento tal como una cinta magnética o un disco, o un medio óptico. El medio de almacenamiento puede usarse para grabar programas de instrucciones para sistemas operativos, utilidades y aplicaciones, y puede incluir programas que implementan diversos aspectos de la presente invención.

Las funciones requeridas para poner en práctica diversos aspectos de la presente invención pueden realizarse mediante componentes que se implementan en una amplia variedad de modos incluyendo componentes lógicos discretos, circuitos integrados, uno o más ASIC y/o procesadores controlados por un programa. La manera en que se implementen estos componentes no es importante para la presente invención.

- 5 Las implementaciones de software de la presente invención pueden ser transportadas por una diversidad de medios legibles por una máquina tales como trayectos de comunicación de banda de base o modulada a lo largo de todo el espectro incluyendo desde frecuencias supersónicas a ultravioletas, o medios de almacenamiento que transportan la información usando esencialmente cualquier tecnología de grabación incluyendo cinta, tarjetas o disco magnéticos,
- 10 tarjetas o disco ópticos, y marcas detectables en medios incluyendo papel.



**REIVINDICACIONES**

1. Un método para descodificar una trama de una señal de audio digital codificada, en el que:

- 5 • la trama comprende metadatos de trama, un primer bloque de audio y uno o más bloques de audio subsiguientes;  
y
- 10 • cada uno del primer bloque de audio y los subsiguientes comprende metadatos de bloque y datos de audio codificados para dos o más canales de audio, en el que:
- los datos de audio codificados comprenden factores de escala y valores a escala que representan el contenido espectral de los dos o más canales de audio, estando cada valor a escala asociado con uno de los factores de escala respectivo; y
- 15 - los metadatos de bloque comprenden información de control que describe las herramientas de control usadas por un proceso de codificación que produjo los datos de audio codificados, incluyendo las herramientas de control procesamiento de transformada híbrida adaptativa que comprende:
- aplicar un banco de filtros de análisis implementado por una transformada primaria a los dos o más canales de audio para generar coeficientes de la transformada primaria, y
- 20 • aplicar una transformada secundaria a los coeficientes de la transformada primaria para al menos alguno de los dos o más canales de audio para generar coeficientes de la transformada híbrida;
- 25 y en el que el método comprende:
- (A) recibir la trama de la señal de audio digital codificada; y
- (B) examinar la señal de audio digital codificada de la trama en una única pasada para descodificar los datos de audio codificados para cada bloque de audio en orden por bloque, en el que la descodificación de cada bloque de audio respectivo comprende:
- 30 (1) determinar para cada canal respectivo de los dos o más canales si el proceso de codificación usó procesamiento de transformada híbrida adaptativa para codificar cualquiera de los datos de audio codificados;
- 35 (2) si el proceso de codificación usó el procesamiento de transformada híbrida adaptativa para el canal respectivo:
- (a) si el bloque de audio respectivo es el primer bloque de audio de la trama:
- 40 (i) obtener todos los coeficientes de la transformada híbrida del canal respectivo para la trama a partir de los datos de audio codificados del primer bloque de audio, y
- (ii) aplicar una transformada secundaria inversa a los coeficientes de la transformada híbrida para obtener coeficientes de la transformada secundaria inversa, y
- 45 (b) obtener coeficientes de la transformada primaria a partir de los coeficientes de la transformada secundaria para el canal respectivo del bloque de audio respectivo;
- (3) si el proceso de codificación no usó el procesamiento de transformada híbrida adaptativa para el canal respectivo, obtener coeficientes de la transformada primaria para el canal respectivo descodificando los datos codificados del bloque de audio respectivo; y
- 50 (C) aplicar una transformada primaria inversa a los coeficientes de la transformada primaria para generar una señal de salida que representa el canal respectivo del bloque de audio respectivo.
- 55
2. El método de la reivindicación 1, en el que la trama de la señal de audio digital codificada cumple la sintaxis del tren de bits AC-3 mejorado.
3. El método de la reivindicación 2, en el que las herramientas de codificación incluyen procesamiento de extensión espectral y la descodificación de cada bloque de audio respectivo comprende además:
- 60 determinar si el proceso de descodificación debería usar procesamiento de extensión espectral para descodificar cualquiera de los datos de audio codificados; y
- 65 si debería usarse procesamiento de extensión espectral, sintetizar una o más componentes espectrales a partir de los coeficientes de la transformada secundaria inversa para obtener coeficientes de la transformada primaria con un

ancho de banda extendido.

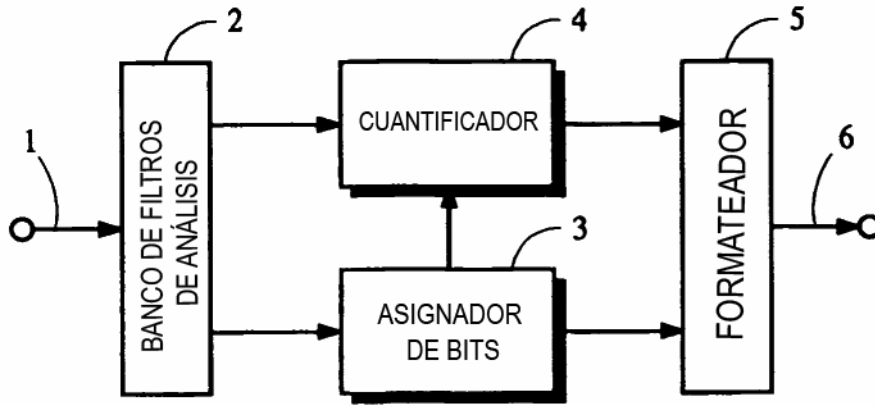
4. El método de la reivindicación 2 o 3, en el que las herramientas de codificación incluyen acoplamiento de canales y la decodificación de cada bloque de audio respectivo comprende además:

5 determinar si el proceso de codificación usó acoplamiento de canales para codificar cualquiera de los datos de audio codificados; y

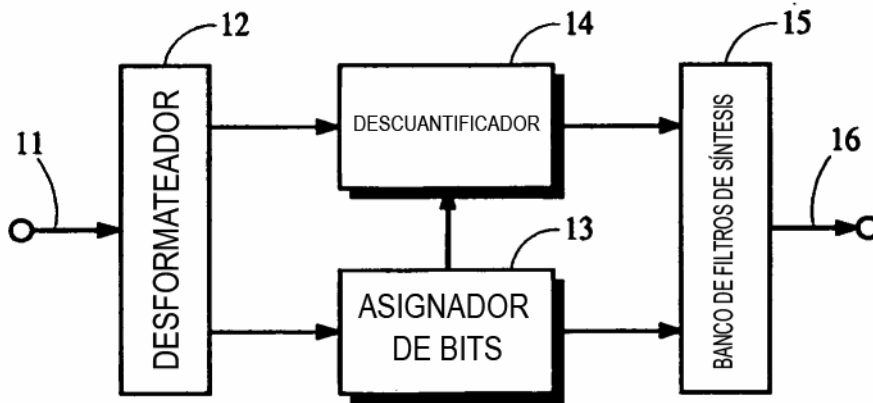
10 si el proceso de codificación usó acoplamiento de canales, deducir las componentes espectrales a partir de los coeficientes de la transformada secundaria inversa para obtener coeficientes de la transformada primaria para canales acoplados.

15 5. Un aparato para decodificar una trama de una señal de audio digital codificada, en el que el aparato comprende medios para realizar todas las etapas de una cualquiera de las reivindicaciones 1 a 4.

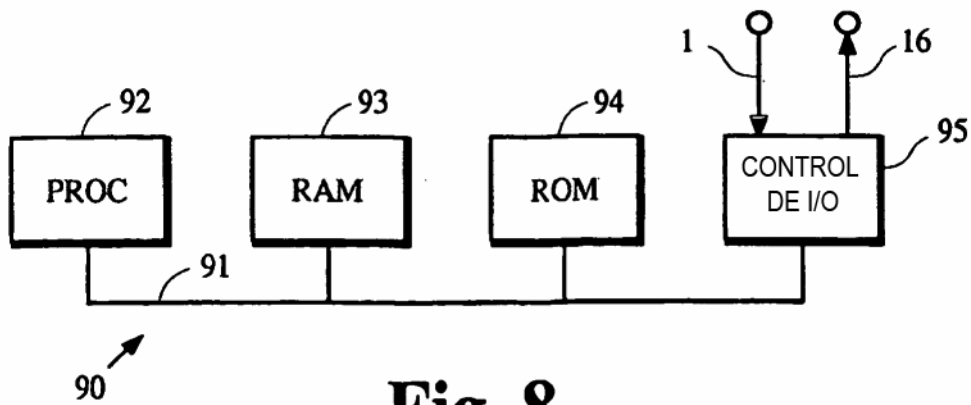
6. Un medio de almacenamiento que graba un programa de instrucciones que es ejecutable por un dispositivo para realizar un método para decodificar una trama de una señal de audio digital codificada, en el que el método comprende todas las etapas de una cualquiera de las reivindicaciones 1 a 4.



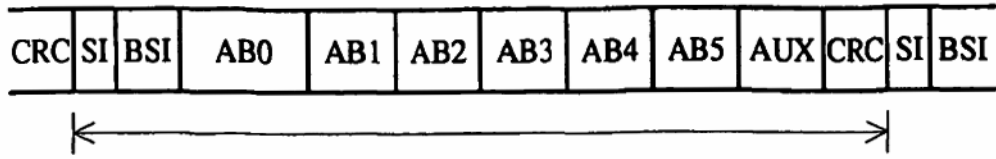
**Fig. 1**



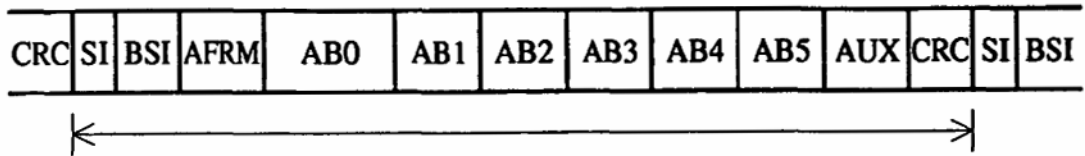
**Fig. 2**



**Fig. 8**



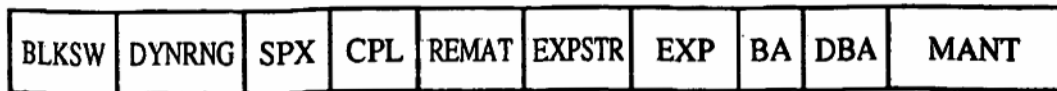
**Fig. 3A**



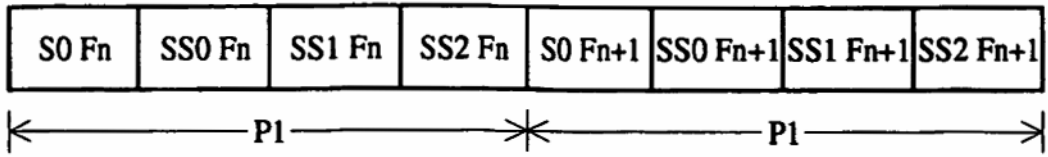
**Fig. 3B**



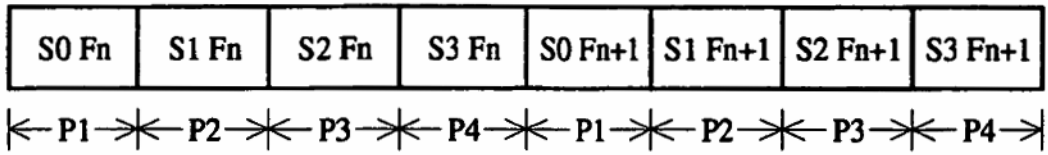
**Fig. 4A**



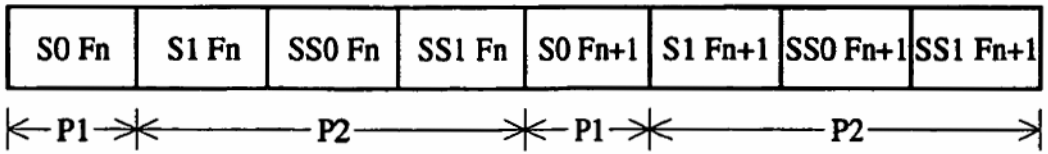
**Fig. 4B**



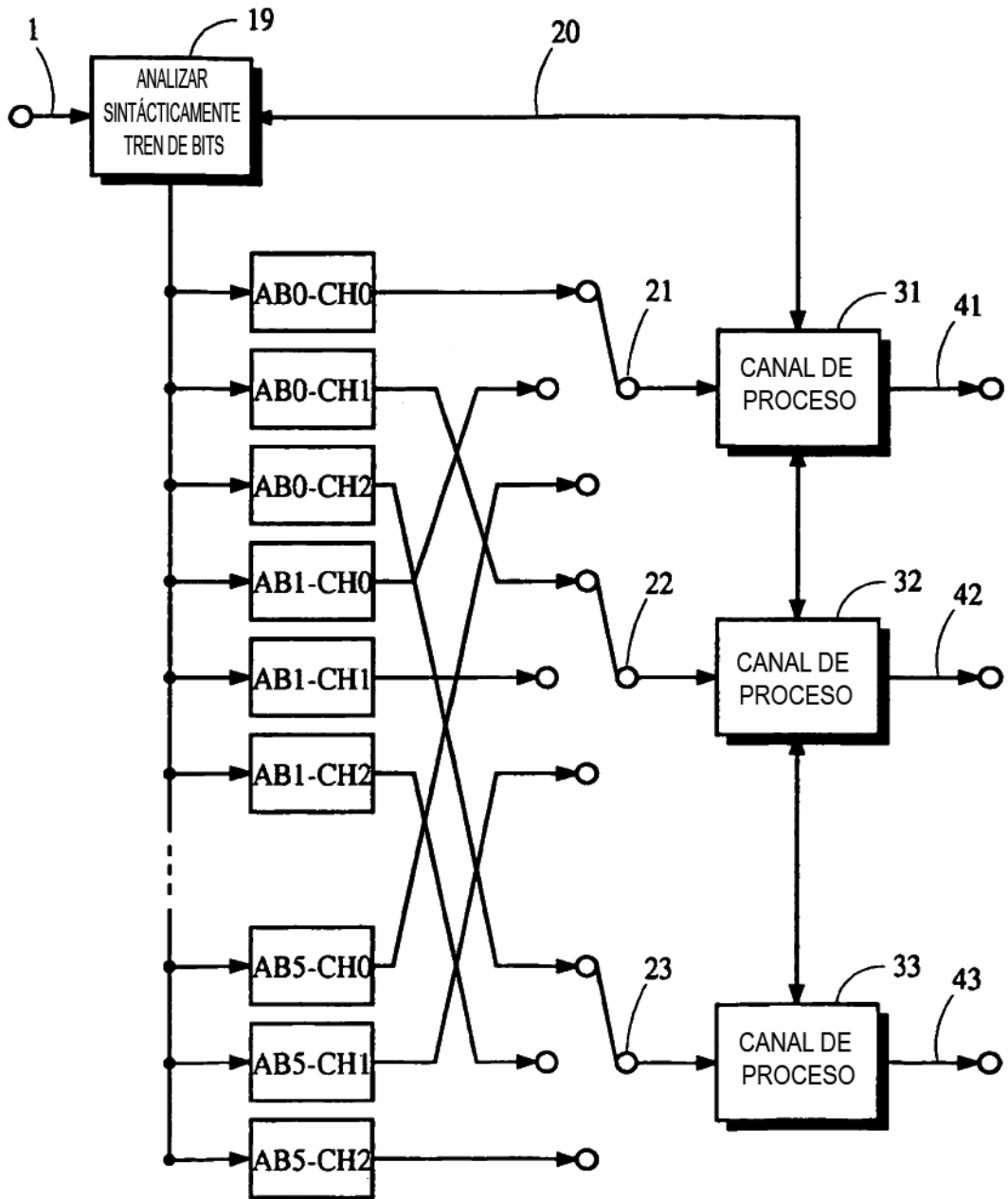
**Fig. 5A**



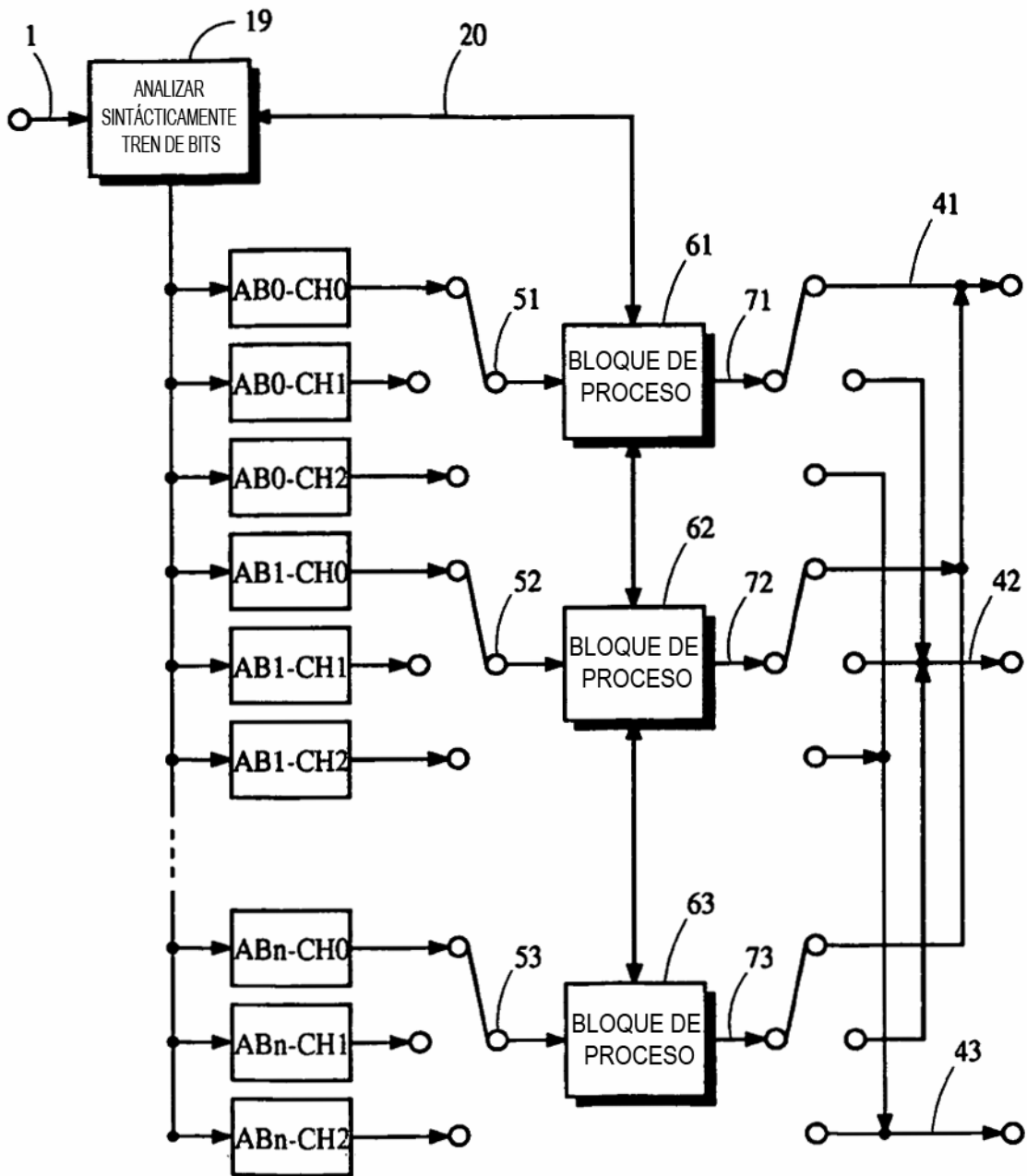
**Fig. 5B**



**Fig. 5C**



**Fig. 6**



**Fig. 7**