



(19)



OFICINA ESPAÑOLA DE  
PATENTES Y MARCAS

ESPAÑA

(11) Número de publicación: **2 316 679**

(51) Int. Cl.:  
**G10L 19/00** (2006.01)  
**G10L 19/02** (2006.01)

(12)

TRADUCCIÓN DE PATENTE EUROPEA

T3

(96) Número de solicitud europea: **03020111 .5**  
(96) Fecha de presentación : **04.09.2003**  
(97) Número de publicación de la solicitud: **1400955**  
(97) Fecha de publicación de la solicitud: **24.03.2004**

(54) Título: **Cuantificación y cuantificación inversa para señales audio.**

(30) Prioridad: **04.09.2002 US 408517 P**  
**15.08.2003 US 642551 P**

(45) Fecha de publicación de la mención BOPI:  
**16.04.2009**

(45) Fecha de la publicación del folleto de la patente:  
**16.04.2009**

(73) Titular/es: **MICROSOFT CORPORATION**  
**One Microsoft Way**  
**Redmond, Washington 98052, US**

(72) Inventor/es: **Thumpudi, Naveen y**  
**Chen, Wei-ge**

(74) Agente: **Carpintero López, Mario**

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

Cuantificación y cuantificación inversa para señales audio.

5 **Campo técnico**

La presente invención se refiere al procesamiento de información de señales audio en la codificación y decodificación. Específicamente, la presente invención se refiere a la cuantificación y a la cuantificación inversa en la codificación y decodificación de señales audio.

10 **Antecedentes**

Con la introducción de discos compactos, redes telefónicas e inalámbricas digitales, y la distribución audio por Internet, el audio digital se ha convertido en algo habitual. Los expertos utilizan diversas técnicas para procesar eficazmente la señal audiodigital manteniendo al tiempo la calidad de la señal audio digital. Para comprender estas técnicas, es útil comprender cómo se representa y procesa la información audio en una computadora.

I. *Representación de la Información Audio en una Computadora*

Una computadora procesa una información audio como una serie de números que representan la información audio. Por ejemplo, un solo número puede representar una muestra audio, la cual es un valor de amplitud (esto es, de intensidad sonora) en un momento determinado. Diversos factores afectan a la calidad de la información audio, incluyendo la profundidad de la muestra, la velocidad de la muestra, y el modo de canal.

La profundidad de la muestra (o precisión) indica el rango de números utilizado para representar una muestra. Cuantos más valores posibles haya para la muestra, más alta será la calidad porque el número puede captar variaciones de amplitud más sutiles. Por ejemplo, una muestra de 8 bits tiene 256 valores posibles, mientras que una muestra de 16 bits tiene 65.536 posibles valores. Una muestra de 24 bits puede captar con gran precisión variaciones de intensidad sonora normales, y puede también captar una intensidad sonora insólitamente alta.

La velocidad de muestreo (generalmente medida como el número de muestras por segundo) también afecta a la calidad. Cuánto más alta sea la velocidad de muestreo, más alta será la calidad porque pueden ser representadas más frecuencias de sonido. Algunas velocidades de muestreo de velocidades son 8.000, 11.025, 22.050, 32.000, 44.100, 48.000 y 96.000 muestras/segundo.

Mono y estéreo son dos modos de canal habituales para audio. En el modo mono, la información audio está presente en un canal. En el modo estéreo, la información audio está presente en dos canales generalmente designados como canales izquierdo y derecho. También son posibles otros modos con más canales, como por ejemplo un sonido envolvente de un canal 5.1, de un canal 7.1, o de un canal 9.1 (el "1" indica un canal de efectos de "sub-woofer" o de baja frecuencia). La tabla 1 muestra varios formatos de audio con diferentes niveles de calidad, junto con los costes brutos de velocidad de transmisión de bits.

TABLA 1

*Velocidades Transmisión de Bits para una información audio de calidad diferente*

<b>Calidad</b>	<b>Profundidad Muestra (bits / muestra)</b>	<b>Velocidad muestreo (muestras / segundo)</b>	<b>Modo</b>	<b>Velocidad Bits bruta (bits / segundo)</b>
Telefonía Internet	8	8.000	mono	64,000
Teléfono	8	11.025	mono	88,200
CD audio	16	44.100	estéreo	1,411,200

La señal audio de sonido envolvente típicamente tiene una velocidad de transmisión de bits bruta más alta. Como muestra la Tabla 1, el coste de la información audio de alta calidad es una velocidad de transmisión de bits alta. La información audio de alta calidad consume grandes cantidades de capacidad de transmisión y almacenaje informáticas. Sin embargo, las empresas y los consumidores dependen cada vez más de las computadoras, para crear, distribuir, y reproducir contenidos audio multicanal de alta calidad.

## II. Procesamiento de Información Audio en una Computadora

Muchas computadoras y redes informáticas carecen de los recursos para procesar señales audiodigitales en bruto. La compresión (también llamada codificación) reduce el coste de la información audio de almacenaje y transmisión mediante la conversión de la información en una forma de velocidad de transmisión de bits más baja. La compresión puede ser sin pérdidas (en la cual la calidad no se resiente) o con pérdidas (en la cual la calidad se resiente pero es más grave la reducción de la velocidad de transmisión de bits derivada de la compresión sin pérdidas subsecuente). La descompresión (también llamada decodificación) extrae una versión reconstruida de la información original a partir de la forma comprimida.

### A. Codificadores y Decodificadores Audio Perceptivos Estándar

En general, el objetivo de la compresión audio es representar digitalmente señales audio para proporcionar una calidad máxima de las señales con la menor cantidad de bits posible. Un sistema de codificador/decodificador audio convencional [“codec”] utiliza una codificación de subbanda/transformación, cuantificación, control de velocidad, y una codificación de longitud variable para conseguir su compresión. La cuantificación y otras técnicas de compresión con pérdidas introducen un ruido potencialmente audible dentro de una señal audio. La audibilidad del ruido depende de la cantidad de ruido que existe y de la cantidad de ruido que el oyente percibe. El primer factor se refiere principalmente a la calidad objetiva, mientras que el segundo factor depende de la percepción humana del sonido.

La figura 1 muestra un diagrama genérico de un codificador (100) audio perceptual basado en una transformación. La figura 2 muestra un diagrama genérico de un decodificador audio correspondiente (200) de acuerdo con la técnica anterior. Aunque el sistema codec mostrado en las figuras 1 y 2 es genérico, presenta características que se encuentran en diversos sistemas codec del mundo real, incluyendo las versiones del codificador y decodificador Audio Media Windows [“WMA”] de la empresa Microsoft Corporation. Otros sistemas codec se proporcionan o especifican con el estándar Audio Layer 3 [“MP3”] del Motion Picture Experts Group, el estándar Advanced Audio Coding [“AAC”], del Motion Picture Experts Group 2, y el Dolby AC3. Para más información acerca de los sistemas codec, véanse los respectivos estándares o publicaciones técnicas.

#### 1. Codificador Audio Sensorial

En términos generales, el codificador (100) recibe una serie temporal de muestras de entrada audio (105), comprime las muestras audio (105), y multiplexa la información producida por los diversos módulos del codificador (100) para emitir de salida un flujo de bits (195). El codificador (100) incluye un transformador de frecuencia (110), un transformador multicanal (120), un modelador perceptual (130), un ponderador (140), un cuantificador (150), un codificador de entropía (160), un controlador (170), y un multiplexor de flujo de bits [“MUX”] (180).

El transformador de frecuencia (110) recibe las muestras audio (105) y las convierte en datos en el ámbito de la frecuencia. Por ejemplo, el transformador de frecuencia (110) divide las muestras audio (105) en bloques, bloques que tienen tamaño variable para posibilitar una resolución temporal variable. Unos bloques pequeños posibilitan una mayor preservación de detalle de tiempo en segmentos de transición cortos pero activos en las muestras audio de entrada (105), pero a expensas de una cierta resolución de frecuencia. Por el contrario, unos bloques grandes tienen una mejor resolución de frecuencia y una peor resolución de tiempo, y generalmente posibilitan una mayor eficiencia de compresión en segmentos más largos y menos activos. Los bloques pueden superponerse para reducir las discontinuidades perceptibles entre bloques que en otro caso serían introducidas mediante una cuantificación posterior. Para señales audio multicanal, el transformador de frecuencia (110) utiliza el mismo patrón de ventanas para cada canal de una trama concreta. Este transformador de frecuencia (110) emite de salida unos bloques de datos de coeficientes de frecuencia hasta el transformador multicanal (120) y emite de salida la información colateral, como por ejemplo los tamaños de los bloques, hasta el MUX (180).

Para datos audio multicanal, a menudo se correlacionan múltiples canales de datos de coeficientes de frecuencia convertidos por el transformador de frecuencia (110). Para explotar esta correlación, el transformador multicanal (120) puede convertir los canales originales múltiples, independientemente codificados, en canales conjuntamente codificados. Por ejemplo, si la entrada es de modo estéreo, el transformador multicanal (120) puede convertir los canales izquierdo y derecho en canales de suma y diferencia:

$$X_{\text{Sum}}[k] = \frac{X_{\text{izquierdo}}[k] + X_{\text{derecho}}[k]}{2} \quad (1),$$

$$X_{\text{Dif}}[k] = \frac{X_{\text{izquierdo}}[k] - X_{\text{derecho}}[k]}{2} \quad (2).$$

O, el transformador multicanal (120) puede atravesar los canales izquierdo y derecho como canales independientemente codificados. La decisión de utilizar canales codificados de manera independiente o conjunta se predetermina o efectúa durante la codificación de acuerdo con las condiciones específicas. Por ejemplo, el codificador (100) determina si para codificar conjunta o independientemente canales estéreo con una decisión de selección de bucle abierto que

considera la a) separación de energía entre los canales de codificación con o sin la transformación multicanal y b) la disparidad en los patrones de excitación entre los canales de entrada izquierdo y derecho. Dicha decisión puede efectuarse sobre una base de ventana por ventana o solo una vez por trama para simplificar la decisión. El transformador multicanal (120) emite una información lateral hasta el MUX (180) que indica el modo de canal utilizado.

El codificador (100) puede aplicar una rematrización multicanal a un bloque de datos audio después de una transformación multicanal. Para una velocidad de transmisión de bits baja, datos multicanal en canales codificados conjuntamente, el codificador (100) suprime de manera selectiva la información de determinados canales (por ejemplo, el canal diferencial) para mejorar la calidad del(de los) canal(es) restante(s) (por ejemplo, el canal de suma). Por ejemplo, el codificador (100) cambia de escala el canal diferencial mediante un factor de escala  $\rho$ :

$$\overline{X}_{\text{Dif}}[k] = \rho \cdot X_{\text{Dif}}[k] \quad (3),$$

donde el valor de  $\rho$  se basa en: a) los niveles medios actuales de una medición de calidad de audio perceptual como la Noise to Excitation Ratio [“NER”], b) la plenitud actual de una memoria intermedia virtual, c) los ajustes de la velocidad de transmisión de bits y de la velocidad de muestreo del codificador (100), y d) la separación de canal en los canales de entrada izquierdo y derecho.

El modelador de percepción (130) procesa los datos audio de acuerdo con un modelo del sistema auditivo humano para mejorar la calidad percibida de la señal audio reconstruida para una velocidad de transmisión de bits determinada. Por ejemplo, un modelo auditivo típicamente considera los límites de la audición humana y las bandas críticas. El sistema nervioso humano integra unos subrangos de frecuencias. Por esta razón, un modelo auditivo puede organizar y procesar información audio mediante bandas críticas. Diferentes modelos auditivos utilizan un número diferente de bandas críticas (por ejemplo, 25, 32, 55 o 109) y/o frecuencias de cierre diferentes para las bandas críticas. Las bandas Bark son un ejemplo bien conocido de bandas críticas. Junto con las bandas límite y críticas, las interacciones entre las señales audio pueden afectar seriamente la percepción. Una señal audio claramente perceptible, si se presenta sola puede ser completamente inaudible en presencia de otra señal audio, llamada enmascaradora o señal de enmascaramiento. El oído humano es relativamente insensible a la distorsión o a otras pérdidas de la fidelidad, (esto es, el ruido), de la señal enmascarada, de forma que la señal enmascarada puede incluir más distorsiones sin degradar la calidad audio percibida. Así mismo, un modelo auditivo puede considerar una pluralidad de otros factores relacionados con aspectos físicos o neuronales de la percepción humana del sonido.

El modelador de percepción (130) emite de salida una información que el ponderador (140) utiliza para conformar el ruido de los datos audio para reducir la audibilidad del ruido. Por ejemplo, utilizando cualquiera de las múltiples técnicas existentes, el ponderador (140) genera unos factores de ponderación (a veces llamados factores de escala) para las matrices de cuantificación (a veces llamadas máscaras) en base a la información recibida. Los factores de ponderación de una matriz de cuantificación incluyen una ponderación para cada una de las múltiples bandas de cuantificación de los datos audio, donde las bandas de cuantificación son gamas de frecuencia de coeficientes de frecuencia. El número de bandas de cuantificación puede ser el mismo que o menor al número de bandas críticas. Así, los factores de ponderación indican las proporciones en las cuales el ruido se difunde a través de las bandas de cuantificación con el objetivo de reducir al mínimo la audibilidad del ruido mediante la introducción de más ruido en las bandas donde es menos audible, y viceversa. Los factores de ponderación pueden variar de bloque a bloque en amplitudes y número de bandas de cuantificación. El ponderador (140) aplica a continuación los factores de ponderación a los datos recibidos del transformador multicanal (120).

En una implementación, el ponderador (140) genera un conjunto de factores de ponderación para cada ventana de cada canal audio multicanal, o comparte un solo conjunto de factores de ponderación para ventanas paralelas de canales conjuntamente codificados. El ponderador (140) emite de salida unos bloques ponderados de coeficientes al cuantificador (150) y emite de salida una información colateral, como por ejemplo los conjuntos de factores de ponderación hasta el MUX (180).

Un conjunto de factores de ponderación puede ser comprimido para una representación más eficiente utilizando una compresión directa. En la técnica de compresión directa, el codificador (100) cuantifica de manera uniforme cada elemento de una matriz de cuantificación. El codificador a continuación codifica de manera diferencial los elementos cuantificados con respecto a los elementos diferentes de la matriz, y codifica con el Código de Huffman los elementos codificados de manera diferencial. En algunos casos (por ejemplo, cuando todos los coeficientes de las bandas de cuantificación concretas han sido cuantificados o truncados hasta un valor de 0), el decodificador (200) no requiere factores de ponderación para todas las bandas de cuantificación. En dichos casos, el codificador (100) proporciona unos valores a uno o más factores de ponderación no requeridos que son idénticos al valor del siguiente factor de ponderación requerido de una serie, lo que hace más eficaz la codificación diferencial de los elementos de la matriz de cuantificación.

O, para aplicaciones de velocidad de transmisión de datos baja, el codificador (100) puede comprimir paramétricamente una matriz de cuantificación para representar la matriz de cuantificación como un conjunto de parámetros, por ejemplo, utilizando una Linear Predictive Coding [“LPC”] de parámetros de pseudoautocorrelación computados a partir de la matriz de cuantificación.

El cuantificador (150) cuantifica la salida del ponderador (140), transmitiendo los datos de los coeficientes cuantificados al codificador de entropía (160) y la información colateral que incluye el tamaño de la etapa de cuantificación hasta el MUX (180). La cuantificación cartografía los límites de los valores de entrada en valores únicos, introduciendo la pérdida irreversible de información, pero posibilitando también que el codificador (100) regule la calidad y la velocidad de transmisión de bits del flujo de bits de salida (195) en combinación con el controlador (170). En la Figura 1, el cuantificador (150) es un cuantificador escalar, uniforme, adaptativo. El cuantificador (150) aplica el mismo tamaño de paso de cuantificación a cada coeficiente de frecuencia, pero el tamaño de paso de cuantificación propiamente dicho puede cambiar de una repetición de un bucle de cuantificación al siguiente para afectar a la velocidad de transmisión de bits de la salida del cuantificador de entropía (160). Otros tipos de cuantificación son la cuantificación no uniforme, la cuantificación de vectores y/o la cuantificación no adaptativa.

El codificador de entropía (160) comprime sin pérdidas los datos de los coeficientes cuantificados recibidos del cuantificador (150). El codificador de entropía (160) puede computar el número de bits empleados en la información audio de decodificación y pasar esta información al controlador (170) de la velocidad/calidad.

El controlador (170) opera con el cuantificador (150) para regular la velocidad de transmisión de bits y/o la calidad de la salida del codificador (100). El controlador (170) recibe la información procedente de otros módulos del codificador (100) y la procesa la información recibida para determinar un tamaño de etapa de cuantificación deseado dadas las condiciones actuales. El controlador (170) emite de salida el tamaño de etapa de cuantificación al cuantificador (150) con el objetivo de satisfacer los condicionamientos de la velocidad de transmisión de bits y de la calidad.

El codificador (100) puede aplicar una sustitución de ruido y/o un truncamiento de banda a un bloque de datos audio. A velocidades de transmisión de bits medias y bajas, el codificador audio (100) puede utilizar la sustitución de ruido para transmitir información en determinadas bandas. En el truncamiento de bandas, si la calidad medida de un bloque indica una calidad eficiente, el codificador (100) puede eliminar completamente los coeficientes de determinadas bandas (de frecuencia generalmente más alta) para mejorar la calidad global de las bandas restantes.

El MUX (180) multiplexa la información colateral recibida de los otros módulos del codificador (100) audio junto con los datos codificados por entropía recibidos del codificador de entropía (160). El MUX (180) emite de salida la información en un formato que reconoce el decodificador audio. El MUX (180) incluye una memoria intermedia virtual que almacena el flujo de bits (195) que van a ser emitidos de salida por el codificador (100) con el fin de suavizar las fluctuaciones a corto plazo de la velocidad de transmisión de bits debidas a la complejidad de los cambios del audio.

## 2. Descodificador Audio Perceptual

En términos generales, el decodificador (200) recibe un flujo de bits (205) de información audio comprimida que incluye datos codificados por entropía así como información colateral, a partir de la cual el decodificador (200) reconstruye muestras audio (295). El decodificador audio (200) incluye un desmultiplexador ["DEMUX"] (210) del flujo de bits, un descodificador de entropía (220), un cuantificador inverso (230), un generador de ruido (240), un ponderador inverso (250), un transformador multicanal inverso (260), y un transformador de frecuencia inversa (270).

El DEMUX (210) analiza la información del flujo de bits (205) y envía la información a los módulos del descodificador (200). El DEMUX (210) incluye una o más memorias intermedias para compensar las variaciones a corto plazo de la velocidad de transmisión de bits debidas a las fluctuaciones de la complejidad de la señal audio, las fluctuaciones de la red, y/u otros factores.

El descodificador de entropía (220) descomprime sin pérdidas los códigos de entropía recibidos del DEMUX (210), produciendo unos datos de coeficientes de frecuencia cuantificados. El descodificador de entropía (220) típicamente aplica la técnica inversa de codificación de entropía utilizada en el codificador.

El cuantificador inverso (230) recibe un tamaño de paso de cuantificación del DEMUX (210) y recibe los datos de los coeficientes de frecuencia cuantificados del descodificador de entropía (220). El cuantificador inverso (230) aplica el tamaño de paso de cuantificación a los datos de los coeficientes de frecuencia cuantificados para reconstruir parcialmente los datos de los coeficientes de frecuencia.

Del DEMUX (210), el generador de ruido (240) recibe información indicativa de las bandas de un bloque de datos que son sustituidas por ruido así como cualquier parámetro de la forma del ruido. El generador de ruido (240) genera las pautas de las bandas indicadas, y pasa la información al ponderador inverso (250).

El ponderador inverso (250) recibe los factores de ponderación del DEMUX (210), establece los patrones de cualquier banda de ruido sustituido desde el generador de ruido (240), y los datos de los coeficientes de frecuencia parcialmente reconstruidos a partir del cuantificador inverso (230). En caso necesario, el ponderador inverso (250) descomprime los factores de ponderación, por ejemplo, la codificación de entropía, la codificación inversa de forma diferencial, y lleva a cabo la cuantificación inversa de los elementos de la matriz de cuantificación. El ponderador inverso (250) aplica los factores de ponderación a los datos de los coeficientes de frecuencia parcialmente reconstruidos de las bandas que no han sido reemplazadas en cuanto al ruido. El ponderador inverso (250) a continuación suma los patrones de ruido recibidos del generador (240) de ruidos de las bandas substituidas por ruido sustituido.

El transformador inverso multicanal (260) recibe los datos de frecuencia reconstruidos del ponderador inverso (250) y la información del modo de canal desde el DEMUX (210). Si la señal audio multicanal está en los canales independientemente codificados, el transformador inverso multicanal (260) pasa a través de los canales. Si los datos multicanal están en los canales conjuntamente codificados, el transformador inverso multicanal (260) convierte los  
5 datos en canales independientemente codificados.

El transformador de frecuencia inversa (270) recibe los datos de los coeficientes de frecuencia emitidos de salida por el transformador multicanal (260), así como información colateral, como por ejemplo los tamaños de los bloques desde el DEMUX (210). El transformador de frecuencia inversa (270) aplica la inversa de la transformación de  
10 frecuencia utilizada en el codificador y emite de salida los bloques de las muestras audio reconstruidas (295).

## B. Desventajas de los Codificadores y Decodificadores Audio Perceptuales Estándar

Aunque los codificadores y decodificadores preceptuales de acuerdo con lo anteriormente descrito tienen un  
15 rendimiento global satisfactorio en muchas aplicaciones, presentan diversos inconvenientes, especialmente respecto de la compresión y descompresión de señales audio multicanal. Los inconvenientes limitan en algunos casos la calidad de la señal audio multicanal reconstruida, por ejemplo, cuando la velocidad de transmisión de bits disponible es pequeña con respecto a los canales audio de entrada.

### 20 1. Inflexibilidad de la Tabicación por Tramas de la Señal Audio Multicanal

En diversos aspectos, la tabicación por tramas llevada a cabo por el codificador (100) de la Figura 1 es inflexible.

Como se indicó anteriormente, el transformador de frecuencia (110) divide una trama de muestras audio de entrada  
25 (105) en una o más ventanas superpuestas para la transformación de frecuencia, donde unas ventanas mayores proporcionan una mejor resolución de frecuencia y la supresión de redundancias y ventanas más pequeñas proporcionan una mejor resolución de tiempo. La mejor resolución de tiempo ayuda al control de los artefactos preeco audibles introducidos cuando la señal efectúa la transición desde la energía baja a la energía alta, pero la utilización de ventanas más pequeñas reduce la compresibilidad de forma que el codificador debe ponderar estas consideraciones al considerar los  
30 tamaños de ventana. Para el canal multiaudio, el transformador de frecuencia (110) tabica los canales de una trama de forma idéntica (esto es, configuraciones de ventana idéntica de los canales), lo que puede ser ineficiente en algunos casos, como se ilustra en las figuras 3a-3c.

La Figura 3a muestra las formas de onda (300) de una señal audioestéreo ejemplar. La señal del canal 0 incluye una  
35 actividad transitoria, mientras que la señal en el canal 1 es relativamente fija. El codificador 100 detecta la transición en el canal 0 y, para reducir el preeco, divide la trama en ventanas moduladas superpuestas más pequeñas (301), como se muestra en la Figura 3b. En aras de la sencillez, la Figura 3c muestra la configuración de ventanas superpuestas (302) en cuadrículas, con líneas de puntos que delimitan los límites de las tramas. Figuras posteriores siguen también este sistema convencional.

Un inconveniente de forzar a todos los canales a que tengan una configuración de ventana idéntica es que una señal  
40 fija en uno o más canales (por ejemplo, el canal 1 en las Figuras 3a - 3c) puede dividirse en ventanas más pequeñas, disminuyendo las ventajas de la codificación. Como una alternativa, el codificador (100) puede forzar a los canales a usar ventanas más grandes, introduciendo el preeco en uno o más canales que tienen transeúntes. Este problema se  
45 exacerba cuando tienen que codificarse más de dos canales.

El AAC posibilita el agrupamiento de canales por pares para las transformaciones multicanal. Entre los canales izquierdo, derecho, central, izquierdo trasero, y derecho trasero, por ejemplo, los canales derecho e izquierdo deben  
50 ser agrupados para su codificación estéreo, y los canales izquierdo trasero y derecho trasero, podrían ser agrupados para su codificación estéreo. Grupos diferentes pueden tener configuraciones de ventana diferentes, pero ambos canales de un grupo determinado tienen la misma configuración de ventana si se utiliza una codificación estéreo. Esto limita la flexibilidad de la tabicación de las transformaciones multicanal en el sistema AAC, como lo hace el uso de agrupamientos solo por pares.

### 55 2. Inflexibilidad de las Transformaciones Multicanal

El codificador (100) de la Figura 1 muestra una determinada redundancia intercanal, pero es inflexible en diversos  
aspectos en términos de transformaciones multicanal. El codificador (100) posibilita dos tipos de transformaciones:  
60 a) una transformación de identidad (que es equivalente a ninguna transformación) o b) una codificación de diferencia de suma de pares estéreo. Estas limitaciones constriñen la codificación multicanal de más de dos canales. Incluso en el AAC, que puede trabajar con más de dos canales, una transformación multicanal se limita a un solo par de canales cada vez.

Varios grupos han experimentado con transformaciones multicanal para canales de sonido envolvente. Por ejemplo,  
65 véase Yang *et al.*, "Una Propuesta de Supresión de Redundancias Intercanal para una Compresión Audio Multicanal de Alta Calidad" ["An Inter-Channel Redundancy Removal Approach For High-Quality Multichannel Audio Compression"], Centésimo Novena Convención de la AES, Los Angeles, Septiembre 2000 ["Yang"], y Wang *et al.*, "Un Algoritmo de Codificación Audio Multicanal para la Supresión de Redundancias Multicanal" ["A Multichannel Audio

Coding Algorithm for Inter-Channel Redundancy Removal”], Centésimo Décima Convención de la AES, Ámsterdam, Países Bajos, Mayo 2001 [“Wang”]. El sistema Yang utiliza una Transformación Karthunen-Loeve [“KLT”] a través de los canales para descorrelacionar los canales para unos factores de compresión satisfactorios. El sistema Wang utiliza una Transformación discreta de coseno [“DCT”]. Ambos sistemas proporcionan resultados satisfactorios, pero

5 presentan todavía diversas limitaciones.

En primer lugar, la utilización de una KLT en muestras audio (ya sea a través del dominio temporal o del dominio de frecuencia, como en el sistema Yang) no controla la distorsión introducida en la reconstrucción. La KLT del sistema Yang no se utiliza de modo satisfactorio para codificaciones audio preceptuales de señal audio multicanal. El sistema

10 Yang no controla la cantidad de fugas procedentes de un canal codificado (por ejemplo, fuertemente cuantificado) a través de múltiples canales reconstruidos en la transformación inversa multicanal. Este inconveniente es señalado en Kuo *et al.*, “Un Estudio de Por Qué la Predicción de Canal Transversal no es Aplicable a la Percepción Audio Perceptual” [“A Study of Why Cross Channel Prediction Is Not Applicable to Perceptual Audio Coding”], IEEE Signal Proc. Letters, vol. 8, no. 9, Septiembre 2001. En otras palabras, la cuantificación que es “inaudible” en un canal codificado

15 puede convertirse en audible cuando se difunde en múltiples canales reconstruidos, dado que la ponderación inversa se lleva a cabo antes de la transformación inversa multicanal. El sistema Wang supera este problema colocando la transformación multicanal después de la ponderación y cuantificación en el codificador (y colocando la transformación inversa multicanal antes de la cuantificación inversa y de la ponderación inversa en el decodificador). El sistema Wang, sin embargo, tiene otros defectos. La ejecución de la cuantificación antes de la transformación multicanal significa que la transformación multicanal debe ser de entero a entero, limitando el número de transformaciones posibles y limitando la supresión de redundancias a través de los canales.

En segundo lugar, el sistema Yang se limita a las transformaciones KLT. Aunque las transformaciones KLT adaptan los datos audio que están siendo comprimidos, la flexibilidad del sistema Yang para utilizar diferentes tipos de transformaciones es limitada. De modo similar, el sistema Wang utiliza una DCT de entero a entero para transformaciones multicanal, los que no es tan satisfactorio como en las DCTs convencionales en términos de compresión de energía, y la flexibilidad del sistema Wang para usar diferentes tipos de transformaciones es limitada.

En tercer lugar, en los sistemas Yang y Wang, no hay ningún mecanismo para controlar cuáles son los canales

30 que se transforman conjuntamente, ni existe un mecanismo para agrupar selectivamente diferentes canales en momentos diferentes para una transformación multicanal. Dicho control contribuye a limitar las fugas de contenidos a través de canales totalmente incompatibles. Así mismo, canales que pueden ser compatibles en conjunto pueden ser incompatibles durante algunos periodos.

En cuarto lugar, en el sistema Yang, el transformador multicanal carece de control en cuanto a la aplicación de la transformación multicanal en el nivel de bandas de frecuencias. Incluso entre canales que son compatibles en conjunto, los canales podrían no ser compatibles en algunas frecuencias, o en algunas bandas de frecuencias. De modo similar, la transformación multicanal del codificador (100) de la Figura 1 carece de control al nivel de subcanal; no controla qué bandas de datos de coeficientes de frecuencia son transformados en el multicanal, lo que ignora las ineficiencias que pueden producirse cuando menos de la totalidad de las bandas de frecuencia de los canales de entrada se correlacionan.

En quinto lugar, incluso cuando canales de origen son compatibles hay a menudo necesidad de controlar el número de canales transformados conjuntamente, para limitar el sobreflujo de datos y reducir los accesos de memoria mientras se implementa al tiempo la transformación. En particular, la KLT del sistema Yang es computacionalmente compleja. Por otro lado, la reducción del tamaño de la transformación reduce también potencialmente la ganancia de codificación en comparación con transformaciones mayores.

En sexto lugar, el envío de información que especifique las transformaciones multicanal puede ser costoso en

50 términos de velocidad de transmisión de bits. Esto es particularmente cierto para la KLT del sistema Yang, en cuanto los coeficientes de transformación para la matriz de covarianza enviados son números reales.

En séptimo lugar para una señal audio multicanal de una velocidad de transmisión de datos baja, la calidad de los canales reconstruidos es muy limitada. A parte de los requisitos de codificación de la velocidad de transmisión de datos baja, ello se debe en parte a la incapacidad del sistema para recortar de modo selectivo y con elegancia el número de canales para los cuales la información está realmente codificada.

### 3. Ineficiencias de la Cuantificación y la Ponderación

En el codificador (100) de la Figura 1, el ponderador (140) conforma la distorsión a través de las bandas en datos audio y el cuantificador (150) fija los tamaños de las etapas de cuantificación para cambiar la amplitud de la distorsión de una trama y, de esta forma, equilibrar la calidad con respecto a la velocidad de transmisión de bits. Aunque el codificador (100) consigue un equilibrio satisfactorio de la calidad y de la velocidad de transmisión de bits en la mayoría de las aplicaciones, el codificador (100) presenta todavía diversos inconvenientes.

En primer lugar, el codificador (100) carece de control directo respecto de la calidad del nivel de los canales. Los factores de ponderación conforman la distorsión global a través de las bandas de cuantificación para un canal determinado. El tamaño uniforme de paso de cuantificación escalar afecta a la amplitud de la distorsión a través de

todas las bandas y canales de frecuencia para una trama. Lejos de imponer una calidad muy alta o muy baja sobre todos los canales, el codificador (100) carece de control directo sobre un ajuste de la calidad igual o al menos comparable en la salida reconstruida de todos los canales.

En segundo lugar, cuando los factores de ponderación son comprimidos con pérdidas, el codificador (100) carece de control sobre la resolución de la cuantificación de los factores de ponderación. Para una compresión directa de una matriz de cuantificación, el codificador (100) cuantifica de manera uniforme los elementos de la matriz de cuantificación, a continuación utiliza la codificación diferencial y la codificación de Huffman. La cuantificación uniforme de los elementos de máscara no se adapta a los cambios de la complejidad de la velocidad de la transmisión de bits o de la señal. Como resultado de ello, en algunos casos la matriz de cuantificación son codificadas con más resolución de la necesaria dada la baja calidad global de la señal audio reconstruida, y en otros casos las matrices de cuantificación son codificadas con menos resolución de la que debería utilizarse dada la alta calidad de la señal audio reconstruida.

En tercer lugar, la compresión directa de las matrices de cuantificación del codificador (100) no consigue explotar las redundancias temporales de las matrices de cuantificación. La compresión directa elimina la redundancia dentro de una matriz de cuantificación concreta, pero ignora la redundancia temporal en una serie de matrices de cuantificación.

### C. Canales Audio de Reconversión

Aparte de la codificación y decodificación multicanal, el Dolby Pro-Logic y otros diversos sistemas llevan a cabo una reconversión de la señal audio multicanal para facilitar la compatibilidad con las configuraciones de altavoz con diferentes números de altavoces. En la reconversión Dolby Pro-Logic, por ejemplo, cuatro canales son reconvertidos en dos canales, teniendo cada uno de los dos canales alguna combinación de los datos audio de los cuatro canales originales. Los dos canales pueden ser emitidos de salida sobre el equipamiento de canal estéreo, o los cuatro canales pueden ser reconstruidos a partir de los dos canales para su emisión de salida para el equipo de los cuatro canales.

Aunque la reconversión de esta naturaleza resuelve algunos problemas de compatibilidad, está limitada a diversas configuraciones determinadas, por ejemplo, la reconversión de cuatro a dos canales. Así mismo, las fórmulas de reconversión están predeterminadas y no permiten cambios a lo largo del tiempo para adaptarse a la señal.

El documento EP-A-0 669 724 divulga un procedimiento de codificación de gran eficacia que distribuye bits variables entre canales a las muestras en las regiones de tiempo o en las regiones de frecuencia de una pluralidad de canales. El procedimiento de codificación de gran eficacia cuantifica la cantidad de distribución de bits de los canales que distribuye la cantidad de bits mayor que una cantidad de referencia predeterminada (por ejemplo, 147 kbps) descomponiéndola en una primera cantidad de distribución de bits que no excede de 147 kbps y una segunda cantidad restante de distribución de bits (que excede de 147 kbps).

El documento EP-A-0 597 649 divulga un procedimiento y un aparato de codificación de gran eficacia que incluye la ejecución de una cuantificación teniendo en cuenta la correlación de las señales de entrada de varios canales en la compresión y/o la expansión de señales de entrada de los respectivos canales simultáneamente.

BOSI M. ET AL: "CODIFICACIÓN AUDIO AVANZADA DEL ISO/IEC MPEG-2" REVISTA DE LA SOCIEDAD DE INGENIERIA AUDIO, SOCIEDAD DE INGENIERIA AUDIO, NEW YORK, NY, EE.UU. vo. 45, no. 10, Octubre 1997 (1997-10), páginas 789 a 812 ["ISO/IEC MPEG-2 ADVANCED AUDIO CODING" JOURNAL OF THE AUDIO ENGINEERING SOCIETY, AUDIO ENGINEERING SOCIETY, NEW YORK, NY, US, vol. 45, no. 10, October 1997 (1997-10) pages 789-812], describen un sistema de codificación audio avanzado MPEG-2 que combina la eficacia de la codificación de un banco de filtros de gran resolución, de técnicas de predicción, y de la codificación de Huffman con funcionalidades adicionales destinadas a suministrar una gran calidad audio a una pluralidad de velocidades de transmisión de datos.

El documento "Tecnología de información - Codificación Genérica de Películas e Información audio asociada - Parte 7: Codificación Audio Avanzada (AAC)" ISO/IEC 13818-7, 1 de Diciembre de 1997 (1997-12-01) ["Information technology - Generic Coding of moving Pictures and associated audio Information - Part 7: Advanced Audio Coding (AAC)" ISO/IEC 13818-7, 1 December 1997 (1997-12-01)] describe el estándar MPEG-2 de codificación audio estándar avanzada.

Constituye el objetivo de la presente invención posibilitar una codificación y decodificación audio mejoradas que utilicen la cuantificación y la cuantificación inversa.

El objetivo se consigue mediante la materia objeto de las reivindicaciones independientes.

Determinadas formas de realización preferentes de la presente invención se definen mediante las reivindicaciones dependientes.



## Sumario

En resumen, la descripción detallada tiene por objeto determinadas estrategias de cuantificación y de cuantificación inversa en la codificación y decodificación de señales audio. Por ejemplo, un codificador audio utiliza una o más técnicas de cuantificación (por ejemplo, la ponderación) para mejorar la calidad y/o la velocidad de transmisión de bits de los datos audio. Ello mejora la sensación de audición global y convierte a los sistemas informáticos en una plataforma más convincente para crear, distribuir y reproducir señales audio de gran calidad. Las estrategias descritas en la presente memoria incluyen diversas técnicas y herramientas, las cuales pueden utilizarse en combinación o independientemente.

De acuerdo con un primer aspecto de las estrategias descritas en la presente memoria, un codificador audio cuantifica unos datos audio en múltiples canales, aplicando múltiples factores de cuantificación específicos de un canal para los múltiples canales. Por ejemplo, los factores de cuantificación específicos de un canal son modificadores de paso de cuantificador, los cuales proporcionan al codificador más control respecto del equilibrio de la calidad de la reconstrucción entre canales.

De acuerdo con un segundo aspecto de las estrategias descritas en la presente memoria, un cuantificador audio cuantifica unos datos audio, aplicando múltiples matrices de cuantificación. El codificador varía la resolución de las matrices de cuantificación. Ello permite, por ejemplo, al codificador cambia la resolución de los elementos de las matrices de cuantificación para utilizar más resolución si la calidad global es buena y utilizar menos resolución si la calidad global es deficiente.

De acuerdo con tercer aspecto de las estrategias descritas en la presente memoria, un codificador audio comprime una o más matrices de cuantificación utilizando una predicción temporal. Por ejemplo, el codificador computa una predicción para una matriz actual con respecto a otra matriz, a continuación computa un residuo a partir de la matriz actual y de predicción. De esta forma, el codificador reduce la velocidad de transmisión de bits asociada con las matrices de cuantificación.

Para los aspectos descritos anteriormente en términos de un codificador audio, un decodificador audio lleva a cabo el procesamiento y la decodificación inversas correspondientes.

Las diversas características y ventajas de la invención se pondrán de manifiesto a partir de la descripción detallada subsecuente de determinadas formas de realización que se exponen con referencia a los dibujos que se acompañan.

## Breve descripción de los dibujos

La Figura 1 es un diagrama de bloques de un codificador audio de acuerdo con la técnica anterior.

La Figura 2 es un diagrama de bloques de un decodificador audio de acuerdo con la técnica anterior.

Las Figuras 3a-3c son gráficos que muestran las configuraciones de ventana para una trama de datos audio estéreo de acuerdo con la técnica anterior.

La Figura 4 es un gráfico que muestra seis canales en una configuración de canal/altavoz 5.1.

La Figura 5 es un diagrama de bloques de un entorno informático apropiado en el cual pueden ser implementadas las formas de realización descritas.

La Figura 6 es un diagrama de bloques de un codificador audio en el cual pueden ser implementadas las formas de realización descritas.

La Figura 7 es un diagrama de bloques de un decodificador audio en el cual pueden ser implementadas las formas de realización descritas.

La Figura 8 es un diagrama de flujo que muestra una técnica generalizada de un preprocesamiento multicanal.

Las Figuras 9a-9e son gráficos que muestran matrices ejemplares de un preprocesamiento multicanal.

La Figura 10 es un diagrama de flujo que muestra una técnica de un preprocesamiento multicanal en el cual la matriz de transformación potencialmente cambia sobre una base de trama por trama.

Las Figuras 11a y 11b son gráficos que muestran configuraciones de mosaico ejemplares para señales audio multicanal.

La Figura 12 es un diagrama de flujo que muestra una técnica generalizada de configurar mosaicos de señales audio multicanal.

## ES 2 316 679 T3

La Figura 13 es un diagrama de flujo que muestra una técnica para configurar conjuntamente mosaicos y enviar información de mosaicos para señales multicanal de acuerdo con una sintaxis de flujo de bits concreta.

La Figura 14 es un diagrama de flujo que muestra una técnica generalizada para llevar a cabo una función multicanal después de una ponderación perceptual.

La Figura 15 es un diagrama de flujo que muestra una técnica generalizada para llevar a cabo una transformación multicanal inversa antes de la ponderación perceptual inversa.

La Figura 16 es un diagrama de flujo que muestra una técnica de agrupación de canales en un mosaico para una transformación multicanal en una implementación.

La Figura 17 es un diagrama de flujo que muestra una técnica para extraer información de un grupo de canales e información de una transformación multicanal para un mosaico a partir de un flujo de bits de acuerdo con una sintaxis de un flujo de bits concreta.

La Figura 18 es un diagrama de flujo que muestra una técnica para incluir de forma selectiva unas bandas de frecuencia de un grupo de canales de una transformación multicanal en una implementación.

La Figura 19 es un diagrama de flujo que muestra una técnica para extraer información de conexión/desconexión de bandas para una transformación multicanal para un grupo de canales de un mosaico a partir de un flujo de bits de acuerdo con una sintaxis de flujo de datos concreta.

La Figura 20 es un diagrama de flujo que muestra una técnica generalizada para emular una transformación multicanal que utiliza una jerarquía de transformaciones multicanal más sencillas.

La Figura 21 es un gráfico que muestra una jerarquía ejemplar de transformaciones multicanal.

La Figura 22 es un diagrama de flujo que muestra una técnica para extraer información para una jerarquía de transformaciones multicanal para grupos de canales a partir de un flujo de bits de acuerdo con una sintaxis de flujo de bits concreta.

La Figura 23 es un diagrama de flujo que muestra una técnica generalizada para la selección de un tipo de transformación multicanal de entre varios tipos disponibles.

La Figura 24 es un diagrama de flujo que muestra una técnica generalizada para extraer un tipo de información multicanal de entre varios tipos disponibles y llevar a cabo una transformación multicanal inversa.

La Figura 25 es un diagrama de flujo que muestra una técnica para extraer una información de transformación multicanal para un grupo de canales a partir de un flujo de bits de acuerdo con una sintaxis de flujo de bits concreta.

La Figura 26 es un gráfico que muestra la forma general de una matriz de rotación para rotaciones Givens para la representación de una matriz multicanal.

Las Figuras 27a-27c son gráficos que muestran matrices de rotación ejemplares para rotaciones Givens para la representación de una matriz multicanal.

La Figura 28 es un diagrama de flujo que muestra una técnica generalizada para la representación de una matriz de transformación multicanal que utiliza rotaciones de factorización Givens cuantificadas.

La Figura 29 es un diagrama de flujo que muestra una técnica para extraer información para una transformación unitaria genérica para un grupo de canales a partir de un flujo de bits de acuerdo con una sintaxis de flujo de bits concreta.

La Figura 30 es un diagrama de flujo que muestra una técnica para extraer un factor global de cuantificación de mosaico para un mosaico a partir de un flujo de bits de acuerdo con una sintaxis de flujo de bits concreta.

La Figura 31 es un diagrama de flujo que muestra una técnica generalizada para la computación de los modificadores de paso de cuantificación por canal para datos audio multicanal.

La Figura 32 es un diagrama de flujo que muestra una técnica para extraer los modificadores de canal de paso de cuantificación por canal a partir de un flujo de bits de acuerdo con una sintaxis de flujo de bits concreta.

La Figura 33 es un diagrama de flujo que muestra una técnica generalizada para regular adaptadamente un tamaño de paso de cuantificación para los elementos de las matrices de cuantificación.

La Figura 34 es un diagrama de flujo que muestra una técnica generalizada para extraer un tamaño de paso de cuantificación adaptativa para unos elementos de las matrices de cuantificación.

Las Figuras 35 y 36 son diagramas de flujo que muestran técnicas para la compresión de las matrices de cuantificación que utilizan una predicción temporal.

La Figura 37 es un gráfico que muestra un cartografiado de bandas para la predicción de unos elementos de las matrices de cuantificación.

La Figura 38 es un diagrama de flujo que muestra una técnica para extraer y decodificar las matrices de cuantificación comprimidas utilizando la predicción temporal de acuerdo con una sintaxis de flujo de bits concreta.

La Figura 39 es un diagrama de flujo que muestra una técnica generalizada de un postprocesamiento multicanal.

La Figura 40 es un gráfico que muestra una matriz ejemplar de un postprocesamiento multicanal.

La Figura 41 es un diagrama de flujo que muestra una técnica de postprocesamiento multicanal en la cual la matriz de transformación cambia potencialmente sobre una base de trama por trama.

La Figura 42 es un diagrama de flujo que muestra una técnica para la identificación y extracción de una matriz de transformación para un postprocesamiento multicanal de acuerdo con una sintaxis de flujo de bits concreta.

## 20 Descripción detallada

Las formas de realización descritas de la presente invención tienen por objeto determinadas técnicas y herramientas de procesamiento de información audio en la decodificación y codificación. En las formas de realización descritas, un codificador audio utiliza diversas técnicas para procesar señales audio durante la codificación. Un decodificador audio utiliza diversas técnicas para procesar señales audio durante la decodificación. Aunque las técnicas se describen en determinados puntos de la memoria como parte de un único sistema integrado, las técnicas pueden aplicarse por separado, eventualmente en combinación con otras técnicas. En formas de realización alternativas, una herramienta de procesamiento audio distinta de un codificador o decodificador implementa una o más de las técnicas.

En algunas formas de realización, un codificador lleva a cabo un preprocesamiento multicanal. Para una codificación con una velocidad de transmisión de bits baja, por ejemplo, un codificador opcionalmente rematiza muestras de señal audio de dominio temporal para incrementar artificialmente la correlación intercanal. Esto hace más eficaz la compresión subsecuente de los canales afectados mediante la reducción de la complejidad de la codificación. El preprocesamiento reduce la separación de canales, pero puede mejorar la calidad global.

En algunas formas de realización, un codificador y un decodificador trabajan con una señal audio multicanal configurada en mosaicos de ventanas. Por ejemplo, el codificador tabica tramas de señales audio multicanal sobre una base de por canal, de forma que cada canal puede tener una configuración de ventanas independiente de los otros canales. El codificador a continuación agrupa las ventanas de los canales tabicados en mosaicos para las transformaciones multicanal. Ello permite que el codificador aisle los transitorios que aparecen en un canal concreto de una trama con ventanas pequeñas (reduciendo artefactos preeco), pero utilizar grandes ventanas para la resolución de frecuencia y para la reducción temporal de redundancias en otros canales de la trama.

En algunas formas de realización, un codificador lleva a cabo una o más técnicas de transformación multicanal posibles. Un decodificador lleva a cabo las correspondientes técnicas de transformación multicanal inversa. En las primeras técnicas, el codificador lleva a cabo una transformación multicanal después de la ponderación perceptual en el codificador, lo que reduce las fugas de ruido de cuantificación audibles a través de los canales tras la reconstrucción. En las segundas técnicas, un codificador agrupa flexiblemente los canales para las transformaciones multicanal para incluir de manera selectiva canales en momentos diferentes. En unas terceras técnicas, un codificador incluye o excluye flexiblemente unas bandas de frecuencia concretas de las transformaciones multicanal, para incluir de forma selectiva bandas compatibles. En unas cuartas técnicas, un codificador reduce la velocidad de transmisión de bits asociada con las matrices de transformación mediante la utilización de forma selectiva de unas matrices predefinidas o la utilización de rotaciones Givens para parametrizar matrices de transformación personalizadas. En unas quintas técnicas, un codificador lleva a cabo unas transformaciones multicanal jerárquicas flexibles.

En algunas formas de realización, un codificador lleva a cabo una o más técnicas de cuantificación o ponderación mejoradas. Un codificador correspondiente lleva a cabo las técnicas correspondientes de cuantificación inversa o de ponderación inversa. En las primeras técnicas, un codificador computa y aplica unos modificadores de paso de cuantificación por canal, y proporciona al codificador más control respecto al equilibrio de la calidad de la reconstrucción entre canales. En las segundas técnicas, un codificador utiliza un tamaño flexible de paso de cuantificación para los elementos de las matrices de cuantificación, lo que posibilita que el codificador cambie la resolución de los elementos de las matrices de cuantificación. En las terceras técnicas, un codificador utiliza una predicción temporal en la compresión de matrices de cuantificación para reducir la velocidad de transmisión de bits.

En algunas formas de realización, un decodificador lleva a cabo un postprocesamiento multicanal. Por ejemplo, el decodificador opcionalmente rematiza muestras audio de dominio temporal para crear canales fantasma en la reproducción, llevar a cabo efectos especiales, reconvertir canales a estéreo para su reproducción sobre un menor número de altavoces, o para cualquier otra finalidad.

En las formas de realización descritas, la señal audio multicanal incluye seis canales de una configuración estándar canal/speaker 5.1 como se muestra en la matriz (400) de la Figura 4. Los “5” canales son el izquierdo, el derecho, el central, el trasero izquierdo, y el trasero derecho, y están orientados espacialmente de la manera habitual para el sonido envolvente. El canal “1” es el canal de efectos de *sub-woofer* o de baja frecuencia. En aras de la claridad, el orden de los canales mostrado en la matriz (400) se usa también para las matrices y ecuaciones del resto de la memoria descriptiva. Formas de realización alternativas utilizan señales audio multicanal con un número de ordenación diferente (por ejemplo, 7.1, 9.1, 2), y/o una configuración de canales.

En las formas de realización descritas, el codificador y el descodificador audio ejecutan diversas técnicas. Aunque las operaciones de estas técnicas se describen en un orden concreto, secuencial, con fines de presentación, debe entenderse que esta forma de descripción abarca redistribuciones menores en el orden de las operaciones, a menos que se requiera una ordenación particular. Por ejemplo, las operaciones descritas secuencialmente pueden en algunos casos redistribirse o llevarse a cabo conjuntamente. Así mismo, por razones de sencillez, los diagramas de flujo típicamente no muestran las diversas formas en las cuales pueden utilizarse las diversas técnicas en combinación con otras técnicas.

## 1. Entorno Informático

La Figura 5 ilustra un ejemplo generalizado de un entorno informático apropiado (500) en el cual las formas de realización descritas pueden ser implementadas. El entorno informático (500) no pretende sugerir limitación alguna en cuanto al alcance del uso o funcionalidad de la invención, en cuanto la presente invención puede ser implementada en entornos informáticos de propósito general o de propósito especial.

Con referencia a la Figura 5, el entorno informático (500) incluye al menos una unidad de procesamiento (510) y una memoria (520). En la Figura 5 la configuración más básica (530) está incluida dentro de una línea de puntos. La unidad de procesamiento (510) ejecuta unas instrucciones ejecutables por computadora y puede ser un procesador real o virtual. En un sistema multiprocesamiento, múltiples unidades de procesamiento ejecutan las instrucciones ejecutables de la computadora para incrementar la potencia de procesamiento. La memoria (520) puede ser una memoria volátil (por ejemplo registros, caché, RAM), una memoria no volátil (por ejemplo ROM, EEPROM, memoria flash, etc.), o alguna combinación de las dos. La memoria (520) almacena un software (580) que implementa técnicas de procesamiento audio de acuerdo con una o más de las formas de realización descritas.

Un entorno informático puede tener otras características. Por ejemplo, el entorno informático (500) incluye un almacenamiento (540), uno o más dispositivos de entrada (550), uno o más dispositivos de salida (560), y una o más conexiones de comunicación (570). Un mecanismo de interconexión (no mostrado) como por ejemplo un bus, un controlador, o una red interconecta los componentes del entorno informático (500). Típicamente, el software del sistema operativo (no mostrado) proporciona un entorno operativo para otro software ejecutivo del entorno informático (500), y coordina las actividades de los componentes del entorno informático (500).

El almacenamiento (540) puede ser extraíble o no extraíble, e incluye discos magnéticos, cintas magnéticas o casetes, CD-ROMs, CD-RWs, DVDs, o cualquier otro medio que pueda ser utilizado para almacenar la información y al que puede tenerse acceso dentro del sistema informático (500). El almacenamiento (540) almacena instrucciones destinadas al software (580) que implementan técnicas de procesamiento audio de acuerdo con una o más de las formas de realización descritas.

El(los) dispositivo(s) de entrada (550) puede(n) ser un dispositivo de entrada táctil, como por ejemplo un teclado, un ratón, un lápiz, una bola, un dispositivo de entrada de voz, un dispositivo de escaneo, una adaptador de red, u otro dispositivo que proporcione una entrada al entorno informático (500). Para la señal audio, el(los) dispositivo(s) de entrada (550) puede(n) ser una tarjeta de sonido o dispositivo similar que acepte una entrada audio de forma analógica o digital, un lector de CD-ROM/DVD que proporcione muestras audio al entorno informático. El(los) dispositivo(s) de salida (560) puede(n) ser una pantalla, una impresora, un altavoz, un escritor CD/DVD, un adaptador de red, u otro dispositivo que proporcione una salida desde el entorno informático (500).

La(s) conexión(es) de comunicación (570) permite(n) la comunicación a través de un medio de comunicación con otra entidad informática. El medio de comunicación transmite información, como por ejemplo instrucciones ejecutables por computadora, información audio comprimida, u otros datos en una señal de datos modulada. Una señal de datos modulada es una señal que tiene una o más de sus características dispuestas o modificadas de tal manera que se codifique una información de la señal. A modo de ejemplo, y no de limitación, los medios de comunicación incluyen técnicas por cable o sin cable implementadas con una portadora eléctrica, óptica, de RF, de infrarrojos, acústica, u otra portadora.

La invención puede ser descrita en el contexto general de unos medios legibles por computadora. Los medios legibles por computadora son cualquier medio disponible al que se pueda acceder dentro de un entorno informático. A modo de ejemplo, y no de limitación, con el entorno informático (500), los medios legibles por computadora incluyen una memoria (520), un almacenamiento (540), unos medios de comunicación y combinaciones de cualquiera de los anteriores.

La invención puede describirse en el contexto general de unas instrucciones ejecutables por computadora, como por ejemplo las incluidas en módulos de programas, que se ejecutan en un entorno informático sobre un procesador elegido

como objetivo real o virtual. Por regla general, los módulos de programas incluyen rutinas, programas, bibliotecas, objetos, clases, componentes, estructuras de datos, etc. que llevan a cabo tareas concretas o implementan tipos de datos abstractos concretos. La funcionalidad de los módulos de programas puede ser combinada o dividida entre módulos de programas de acuerdo con lo deseado en las distintas formas de realización. Las instrucciones ejecutables por computadora destinadas a los módulos de programas pueden ser ejecutadas dentro de un entorno informático local o distribuido.

Por razones de presentación, la descripción detallada utiliza términos como “determinar”, “generar”, “ajustar”, y “aplicar” para describir operaciones informáticas de un entorno informático. Estos términos son abstracciones de alto nivel para operaciones llevadas a cabo por una computadora, y no deben confundirse con actos llevados a cabo por un ser humano. Las operaciones informáticas reales correspondientes a estos términos varían dependiendo de la implementación.

## II. Codificador y Decodificador Audio Generalizados

La Figura 6 es un diagrama de bloques de un codificador audio generalizado (600) en el cual pueden ser implementadas las formas de realización descritas. La Figura 7 es un diagrama de bloques de un decodificador audio generalizado (700) en el cual pueden ser implementadas las formas de realización descritas.

Las relaciones mostradas entre módulos existentes dentro del codificador y el decodificador indican flujos de información dentro del codificador y el decodificador; no se muestran otras relaciones por razones de sencillez. Dependiendo de la implementación y del tipo de compresión deseada, pueden añadirse, suprimirse, o dividirse en múltiples módulos determinados módulos del codificador o decodificador, combinarse con otros módulos, y/o sustituirse por otros módulos. En formas de realización alternativas, los codificadores o decodificadores con diferentes módulos y/u otras configuraciones procesan los datos audio.

### A. Codificador Audio Generalizado

El codificador audio generalizado (600) incluye un seleccionador (608), un preprocesador multicanal (610), un configurador (620) de tabicador/mosaico, un transformador de frecuencia (630), un modelador de percepciones (640), un ponderador de bandas de cuantificación (642), un ponderador de canales (644), un transformador multicanal (650), un cuantificador (660), un codificador de entropía (670), un controlador (680), un codificador sin pérdidas mixto/puro (672) y un codificador de entropía asociado (674), y un multiplexor de flujo de bits [“MUX”] (690).

El codificador (600) recibe una serie temporal de muestras audio de entrada (605) a una cierta profundidad y velocidad de muestreo en un formato modulado por impulsos codificados [“PCM”]. Para la mayoría de las formas de realización descritas, las muestras audio de entrada 605 son para señales audio multicanal (p. ej. estéreo, envolvente), pero las muestras audio de entrada (605) pueden por el contrario ser mono. El codificador (600) comprime las muestras audio (605), y multiplexa la información producida por los diversos módulos del codificador (600) para emitir de salida un flujo de bits (695) en un formato que puede ser un formato Windows Media Audio [“WMA”] o un Advanced Streaming Format [“ASF”]. Como una alternativa, el codificador (600) trabaja con otros formatos de entrada y/o salida.

El seleccionador (608) selecciona entre múltiples modos de codificación para las muestras audio (605). En la Figura 6, el seleccionador (608) pasa de un modo de codificación sin pérdidas mixto/puro a un modo de codificación con pérdidas. El modo de codificación sin pérdidas incluye el codificador sin pérdidas mixto/puro (672) y se utiliza típicamente para una compresión de alta calidad (y una velocidad de transmisión de bits alta). El modo de codificación con pérdidas incluye componentes tales como el ponderador (642) y el cuantificador (660) y se utiliza típicamente para una compresión de la calidad ajustable (y una velocidad de transmisión de bits controlada). La decisión de selección en el seleccionador (608) depende de la entrada de usuario o de otros criterios. En determinadas circunstancias (por ejemplo, cuando la compresión con pérdidas no consigue suministrar una cantidad suficiente o produce un exceso de bits), el codificador (600) puede pasar de una codificación con pérdidas a una codificación sin pérdidas mixta/pura para una trama o conjunto de tramas.

Para una codificación con pérdidas de datos audio multicanal, el preprocesador multicanal (610) opcionalmente rematriza las muestras audio (605) de dominio temporal. En algunas formas de realización, el preprocesador multicanal (610) selectivamente rematriza las muestras audio (605) para abandonar uno o más canales codificados o incrementar la correlación intercanal del codificador (600), pero permitiendo la reconstrucción (de alguna forma) del codificador (700). Esto proporciona al codificador un control adicional respecto de la calidad en el nivel de los canales. El preprocesador multicanal (610) puede enviar una información colateral, como por ejemplo instrucciones para el postprocesamiento multicanal hasta el MUX (690). Para mayor detalle acerca del funcionamiento del preprocesador multicanal en algunas formas de realización, véase la sección titulada “Preprocesamiento Multicanal”. Como una alternativa, el codificador (600) lleva a cabo otra forma de preprocesamiento multicanal.

El configurador de tabicador/mosaico (620) tabica una trama de muestras de entrada audio (605) en bloques de subtramas (esto es, ventanas) con funciones de tamaño de tiempo variables y conformación de ventanas. Los tamaños y las ventanas de los bloques de subtramas dependen de la detección de las señales transitorias de la trama, del modo de codificación, así como de otros factores.

Si el codificador (600) pasa de la codificación con pérdidas a la codificación sin pérdidas mixta/pura, los bloques subtramas no necesitan en teoría solaparse o incorporar una función de gestión de ventanas (esto es, no superposición, bloques de ventanas rectangulares), pero las transiciones entre las tramas codificadas con pérdidas y las otras tramas pueden requerir especial tratamiento. El configurador de particionadores/mosaicos (620) emite de salida unos bloques de los datos particionados hasta el codificador sin pérdidas mixto/puro (672) y emite de salida una información colateral como por ejemplo los tamaños de los bloques hasta el MUX (690). Para mayor detalle acerca de la tabicación y de la gestión de ventanas para las tramas codificadas sin pérdidas mixtas o puras, véase la solicitud relacionada con el título “Compresión Audio Unificada con Pérdidas y sin Pérdidas” [“Unified Lossy and Lossless Audio Compression”].

Cuando el codificador (600) utiliza la codificación con pérdidas, las ventanas de tamaño variable posibilitan una resolución temporal variable. Los pequeños bloques posibilitan una mayor preservación del detalle del tiempo en segmentos de transición cortos pero activos. Los grandes bloques tienen una resolución de frecuencia mejor y una resolución de tiempo peor, y generalmente permiten una eficacia de compresión mayor en segmentos más largos y menos activos, en parte porque la cabecera de la trama y la información colateral es proporcionalmente menor que en los bloques pequeños, y en parte porque posibilita una mejor supresión de las redundancias. Los bloques pueden superponerse para reducir las discontinuidades perceptuales entre los bloques que, en otro caso, podrían ser introducidas mediante una cuantificación posterior. El configurador de tabicadores/mosaico (620) emite de salida determinados bloques de datos particionados hasta el transformador de frecuencias (630) y emite de salida información colateral como por ejemplo los tamaños de los bloques hasta el MUX (690). Para más información acerca de los criterios de detección y tabicación transitorios en algunas formas de realización, véase la Solicitud de Patente estadounidense con el número de serie 10/016,918 titulada “Selección Adaptativa del Tamaño de las Ventanas en la Codificación de Transformación” [“Adaptative Window-Size Selection in Transform Coding”], depositada el 14 de Diciembre de 2001. Como una alternativa, el configurador de tabicador/mosaico (620) utiliza otros criterios de tabicación o de tamaños de bloques al tabicar una trama en ventanas.

En algunas formas de realización, el configurador de tabicadores/mosaico (620) tabica tramas de señales audio multicanal sobre una base de por canal. El configurador de tabicador/mosaico (620) tabica independientemente cada canal de trama, si lo permite la calidad/velocidad de transmisión de bits. Ello permite, por ejemplo, que el configurador de tabicador/mosaico (620) aisle transitorios que aparecen en un canal concreto con ventanas más pequeñas, pero utilicen ventanas más grandes para una resolución de frecuencia o una eficacia de la compresión en otros canales. Ello puede mejorar la eficacia de la compresión mediante el aislamiento de los transitorios sobre una base de por canal, pero en muchos casos se necesita una información adicional que especifique las tabicaciones de los canales individuales. Las ventanas del mismo tamaño que están cosituadas en el tiempo pueden estar calificadas para una ulterior reducción de las redundancias a través de la transformación multicanal. Así, el configurador de tabicadores/mosaico (620) agrupa ventanas del mismo tamaño que están cosituadas en el tiempo como un mosaico. Para mayor detalle acerca de la disposición en mosaico en algunas formas de realización, véase la sección titulada “Configuración en Mosaico”.

El transformador de frecuencia (630) recibe las muestras audio y las convierte en datos en el dominio de frecuencia. El transformador de frecuencia (630) emite de salida unos bloques de datos de coeficientes de frecuencia hasta el ponderador (642) y emite de salida información colateral, como por ejemplo los tamaños de los bloques hasta el MUX (690). El transformador de frecuencias (630) emite de salida tanto los coeficientes de frecuencia como la información colateral hasta el modelador perceptual (640). En algunas formas de realización, el transformador de frecuencia (630) aplica una Transformación Lineal Ortogonal Modulada [“MLT”] de tiempo variable hasta los bloques de subtramas la cual opera como una DCT modulado por la(s) función(es) de ventanas sinusoidales de los bloques de subtramas. Formas de realización alternativas utilizan otras variantes de la MLT, o una DCT u otro tipo de transformación de frecuencia modulada o no modulada, superpuesta o no superpuesta, o utilizan una codificación de subbanda o de *wavelet*.

El modelador (640) de la percepción modela las propiedades del sistema auditivo humano para mejorar la calidad percibida de la señal audio reconstruida para una velocidad de transmisión de bits determinada. En términos generales, el modelador (640) de la percepción procesa los datos audio de acuerdo con un modelo auditivo, a continuación suministra la información al ponderador (642) el cual puede ser utilizado para generar factores de ponderación para los datos audio. El modelador (640) de la percepción utiliza cualquiera de los modelos auditivos y pasa la información de la pauta de excitación u otra información al ponderador (642).

El ponderador (642) de las bandas de cuantificación genera unos factores de ponderación para las matrices de cuantificación en base a la información recibida del modelador (640) de la percepción y aplica los factores de ponderación a los datos recibidos del transformador de frecuencia (630). Los factores de ponderación de una matriz de cuantificación incluyen un peso para cada una de las múltiples bandas de cuantificación de los datos audio. Las bandas de cuantificación pueden ser las mismas o diferentes en número o posición respecto de las bandas críticas utilizadas en cualquier parte del codificador (600), y los factores de ponderación pueden variar en amplitudes y número de bandas de cuantificación de bloque a bloque. El ponderador (642) de las bandas de cuantificación emite de salida unos bloques ponderados de datos de coeficientes hasta el ponderador de canales (644) y emite de salida determinada información colateral, como por ejemplo el conjunto de factores de ponderación al MUX (690). El conjunto de factores de ponderación puede ser comprimido para una representación más eficaz. Si los factores de ponderación son comprimidos con pérdidas, los factores de ponderación reconstruidos son típicamente utilizados para ponderar los bloques de datos de los coeficientes. Para mayor detalle acerca de la computación y compresión de los factores de ponderación en algunas formas de realización, véase la sección titulada “Cuantificación y Ponderación”. Como una alternativa, el codificador (600) omite la ponderación.

El ponderador de canales (644) genera unos factores de peso específicos de un canal (que son escalares) para canales basados en la información recibida del modelador (640) de la percepción y también en la calidad de la señal localmente reconstruida. Los pesos escalares (también llamados modificadores de paso de cuantificación) posibilitan que el codificador (600) ofrezca a los canales reconstruidos una calidad aproximadamente uniforme. Los factores de peso de los canales pueden variar en amplitudes de canal a canal y de bloque a bloque, o en algún otro nivel. El ponderador de canales (644) emite de salida los bloques ponderados de datos de los coeficientes al transformador multicanal (650) y emite de salida determinada información colateral, como por ejemplo el conjunto de los factores de peso de los canales hasta el MUX (690). El ponderador de canales (644) y el ponderador (642) de las bandas de cuantificación del diagrama de flujo pueden ser intercambiados o combinados entre sí. Para mayor detalle acerca de la computación y la compresión de los factores de ponderación en algunas formas de realización, véase la sección titulada “Cuantificación y Ponderación”. Como una alternativa, el codificador (600) omite la ponderación.

Para datos audio multicanal, los múltiples canales de datos de los coeficientes de frecuencia de ruido conformado producidos por el ponderador (644) de los canales a menudo se correlacionan, de forma que el transformador multicanal (650) puede aplicar una transformación multicanal. Por ejemplo, el transformador multicanal (650) aplica de forma selectiva y flexible la transformación multicanal a algunos pero no a todos los canales y/o las bandas de cuantificación del mosaico. Ello proporciona al transformador multicanal (650) un control más preciso respecto de la aplicación de la transformación sobre partes relativamente correlacionadas del mosaico. Para reducir la complejidad computacional, el transformador multicanal (650) puede utilizar una transformación jerárquica mejor que una transformación de un nivel. Para reducir la velocidad de transmisión de bits asociada con la matriz de transformación, el transformador multicanal (650) utiliza selectivamente matrices predefinidas (por ejemplo, de identidad/no transformación, Hadamard, DCT Tipo II) o matrices personalizadas, y aplica una compresión eficiente a las matrices personalizadas. Finalmente, dado que la transformación multicanal se encuentra corriente abajo del ponderador (642), la perceptibilidad del ruido (por ejemplo, debida a la cuantificación subsecuente) que se filtra entre canales después de la transformación multicanal inversa del descodificador (700) es controlada mediante ponderación inversa. Para más detalle acerca de las transformaciones multicanal en algunas formas de realización, véase la sección titulada “Transformaciones Multicanal Flexibles”. Como una alternativa, el codificador (600) utiliza otras formas de transformaciones multicanal o ninguna transformación. El transformador multicanal (650) envía una información colateral al MUX (690) indicativa de, por ejemplo, las transformaciones multicanal utilizadas y las partes multicanal transformadas de mosaicos.

El cuantificador (660) cuantifica la salida del transformador multicanal (650), enviando los datos de los coeficientes cuantificados al codificador de entropía (670) y la información colateral que incluye los tamaños de paso de cuantificación hasta el MUX (690). En la Figura 6, el cuantificador (660) es un cuantificador escalar uniforme adaptativa que computa un factor de cuantificación por mosaico. El factor de cuantificación de mosaicos puede cambiar de una iteración de un bucle de cuantificación a la siguiente para afectar a la velocidad de transmisión de bits de la salida del codificador de entropía (660), y los modificadores de paso de cuantificación por canal pueden ser utilizados para calibrar la calidad de la reconstrucción entre canales. Para más detalle acerca de la cuantificación en algunas formas de realización véase la sección titulada “Cuantificación y Ponderación”. En formas de realización alternativas, el cuantificador es un cuantificador no uniforme, un cuantificador de vectores, y/o un cuantificador no adaptativo, o utiliza una forma diferente de cuantificación escalar, uniforme, adaptativa. En otras formas de realización alternativas, el cuantificador (660), el ponderador (642) de las bandas de cuantificación, el ponderador (644) de los canales y el transformador multicanal (650) están fundidos y el módulo fundido determina los diversos pesos de una sola vez.

El codificador de entropía (670) comprime sin pérdidas los datos de los coeficientes cuantificados recibidos del cuantificador (660). En algunas formas de realización, el codificador de entropía (670) utiliza una codificación de entropía adaptativa tal como la descrita en la solicitud relacionada con el título, “Codificación de Entropía mediante Codificación Adaptativa entre Modos de Nivel y Longitud de Recorrido/Nivel” [“Entropy Coding by Adapting Coding Between Level and Run Length/Level Modes”]. Como una alternativa, el codificador de entropía 670 utiliza alguna otra forma o combinación de codificación de longitud de recorrido de nivel, de codificación de longitud variable a variable, de codificación de longitud de recorrido, de codificación de Huffman, de codificación de diccionario, codificación aritmética, de codificación LZ, o alguna otra técnica de codificación de entropía. El codificador de entropía (670) puede computar el número de bits gastados codificando la información audio y pasar esta información al controlador de la velocidad/calidad (680).

El controlador (680) trabaja con el cuantificador (660) para regular la velocidad de transmisión de bits y/o la calidad de la salida del codificador (600). El controlador (680) recibe la información de otros módulos del codificador (600) y procesa la información recibida para determinar los factores de cuantificación deseados dadas las actuales condiciones. El controlador (670) emite de salida los factores de cuantificación hasta el cuantificador (660) con el objetivo de satisfacer los condicionamientos de calidad y/o de velocidad de transmisión de datos.

El codificador sin pérdidas mixto/puro (672) y el codificador de entropía asociado (674) comprimen los datos audio para el modo de codificación sin pérdidas mixto/puro. El codificador (600) utiliza el modo de codificación sin pérdidas mixto/puro para una entera secuencia o cambia de modo de codificación sobre una base de trama por trama, bloque por bloque, mosaico por mosaico u otra. Para mayor detalle acerca del modo de codificación sin pérdidas mixto/puro véase la solicitud relacionada con el título “Compresión Audio Unificada con Pérdidas y sin Pérdidas” [“Unified Lossy and Lossless Audio Compression”]. Como una alternativa, el codificador (600) utiliza otras técnicas para la codificación sin pérdidas mixta y/o pura.

El MUX (690) multiplexa la información colateral recibida de los otros módulos del codificador audio (600) junto con los datos codificados por entropía recibidos de los codificadores de entropía (670), (674). El MUX (690) emite de salida la información en un formato WMA u otro formato que reconoce un decodificador audio. El MUX (690) incluye una memoria intermedia virtual que almacena el flujo de bits (695) que va a ser emitido de salida por el codificador (600). La memoria intermedia virtual a continuación emite de salida los datos a una velocidad de transmisión de bits relativamente constante, aunque la calidad puede cambiar debido a los cambios de la complejidad de la entrada. La plenitud actual u otras características de la memoria intermedia, pueden ser utilizadas por el controlador (680) para regular la calidad y/o la velocidad de transmisión de bits. Como una alternativa, la velocidad de transmisión de bits de salida puede variar con el tiempo, y la calidad se mantiene relativamente constante. O, la velocidad de transmisión de bits de salida se restringe únicamente para que sea menor de una velocidad de transmisión de bits determinada, la cual es o bien constante o bien varía con el tiempo.

#### B. Decodificador Audio Generalizado

Con referencia a la Figura 7, el decodificador audio generalizado (700) incluye un desmultiplexor ["DEMUX"] de flujo de bits (710), uno o más codificadores de entropía (720), un decodificador sin pérdidas mixto/puro (722), un decodificador de configuración de mosaicos (730), un transformador multicanal inverso (740), un cuantificador/ponderador inverso (750), un transformador de frecuencia inverso (760), un solapador/sumador (770), y un post-procesador multicanal (780). El decodificador (700) es algo más sencillo que el codificador (700) porque el decodificador (700) no incluye módulos para el control de la velocidad/calidad o para la modelación de la percepción.

El decodificador (700) recibe un flujo de bits (705) de información audio comprimida en un formato WMA u otro formato. El flujo de bits (705) incluye datos codificados por entropía así como información colateral a partir de la cual el decodificador (700) reconstruye unas muestras audio (795).

El DEMUX (710) analiza la información del flujo de bits (705) y envía la información a los módulos del decodificador (700). El DEMUX (710) incluye una o más memorias intermedias para compensar las variaciones a corto plazo de la velocidad de transmisión de bits debidas a las fluctuaciones en la complejidad del audio, la fluctuación de red, y/u otros factores.

Los uno o más codificadores de entropía (720) descomprimen sin pérdidas los códigos de entropía recibidos del DEMUX (710). El decodificador de entropía (720) típicamente aplica la inversa de la técnica de codificación de entropía utilizada en el codificador (600). Por razones de sencillez, en la Figura 7 se muestra solo un módulo de decodificador de entropía, aunque pueden ser utilizados diferentes decodificadores de entropía para módulos de codificación con pérdidas o sin pérdidas, o incluso dentro de los modos. Así mismo, por razones de sencillez, la Figura 7 no muestra una lógica de la selección de los modos. Al decodificar los datos de decodificación comprimidos en el modo de configuración con pérdidas, el decodificador de entropía (720) produce unos datos de coeficiente de frecuencia cuantificados.

El decodificador sin pérdidas mixto/puro (722) y el(los) decodificador(es) de entropía asociado(s) (720) descomprime(n) los datos audio codificados sin pérdidas destinados al modo de codificación sin pérdidas mixto/puro. Para más detalle acerca de la descompresión del modo de decodificación sin pérdidas mixto/puro, véase la solicitud relacionada con el título "Compresión Audio Codificada con Pérdidas y sin Pérdidas" ["Unified Lossy and Lossless Audio Compression"]. Como una alternativa, el decodificador 700 utiliza otras técnicas de decodificación sin pérdidas mixtas y/o puras.

El decodificador de configuración de mosaicos (730) recibe y, en caso necesario, decodifica la información indicativa de los patrones de los mosaicos para las tramas procedentes del DEMUX (790). La información de la pauta de los mosaicos puede ser codificada por entropía o parametrizada de cualquier otra forma. El decodificador de configuración de mosaicos (730) a continuación pasa la información de la pauta de los mosaicos a otros diversos módulos del decodificador (700). Para mayor detalle acerca de la configuración de mosaicos en algunas formas de realización, véase la sección titulada "Configuración de Mosaicos": Como una alternativa, el decodificador (700) utiliza otras técnicas para parametrizar patrones de ventanas en tramas.

El transformador multicanal inverso (740) recibe los datos de los coeficientes de frecuencias cuantificados del decodificador de entropía (720) así como la información de la pauta de mosaicos procedente del decodificador de configuración de mosaicos (730) y la información colateral procedente del DEMUX (710) indicativa, por ejemplo, de la transformación multicanal utilizada y de las partes transformadas de los mosaicos. Utilizando esta información, en transformador multicanal inverso (740) descomprime la matriz de transformación la medida necesaria, y aplica de forma selectiva y flexible una o más transformaciones multicanal inversas a los datos audio. El emplazamiento del transformador multicanal inverso (740) con respecto al cuantificador/ponderador inverso (750) contribuye a transformar el ruido de cuantificación que puede filtrarse a través de los canales. Para mayor detalle acerca de las transformaciones multicanal inversas en algunas formas de realización, véase la sección titulada "Transformaciones Multicanal Flexibles".

El cuantificador/ponderador inverso (750) recibe los factores de cuantificación de mosaicos y canales así como las matrices de cuantificación del DEMUX (710) y recibe los datos de los coeficientes de frecuencia cuantificados del transformador multicanal inverso (740). El cuantificador/ponderador inverso (750) lleva a cabo la cuantificación



y ponderación inversas. Para mayor detalle acerca de la cuantificación y ponderación inversas en algunas formas de realización, véase la sección titulada “Cuantificación y Ponderación”.

El transformador de frecuencia inversa (760) recibe los datos de los coeficientes de frecuencia emitidos de salida por el cuantificador/ponderador inverso (750) así como la información colateral procedente del DEMUX (710) y la información de la pauta de los mosaicos procedentes del descodificador de configuración de mosaicos (730). El transformador de frecuencia inversa (770) aplica la inversa de la transformación de frecuencia utilizada en el codificador y emite de salida unos bloques hasta el solapador/sumador (770).

Además de recibir la información de la pauta de los mosaicos procedente del descodificador de configuración de mosaicos (730), el solapador/sumador (770) recibe la información descodificada del transformador de frecuencia inversa (760) y/o el descodificador sin pérdidas mixto/puro (722). El solapador/sumador (770) solapa y suma los datos audio de la forma necesaria e intercala las tramas u otras secuencias de datos audio codificadas con los diferentes modos. Para mayor detalle acerca del solapamiento, suma, e intercalación de tramas codificadas sin pérdidas mixtas o puras, véase la solicitud relacionada con el título “Compresión Audio Unificada con Pérdidas y sin Pérdidas”. Como una alternativa, el descodificador (700) utiliza otras técnicas de solapamiento, suma, y tramas de intercalación.

El postprocesador multicanal (780) opcionalmente rematriza las muestras audio de dominio temporal emitidas de salida por el solapador/sumador (770). El postprocesador multicanal rematriza selectivamente los datos audio para crear unos canales fantasma de reproducción, llevar a cabo efectos especiales, como por ejemplo la rotación espacial de canales entre altavoces, reconvertir canales para su reproducción en un menor número de altavoces, o para cualquier otra finalidad. Para un postprocesamiento controlado del flujo de bits, las matrices de transformación de postprocesamiento varían con el tiempo o se señalan o se incluyen en el flujo de bits (705). Para mayor detalle acerca del funcionamiento del postprocesador multicanal en algunas formas de realización véase la sección titulada “Postprocesamiento Multicanal”. Como una alternativa, el descodificador (700) lleva a cabo otra forma de procesamiento multicanal.

### III. Preprocesamiento Multicanal

En algunas formas de realización, un codificador, como por ejemplo el codificador (600) de la Figura 6, lleva a cabo un preprocesamiento multicanal sobre muestras audio de entrada en el dominio temporal.

En general, cuando hay unos canales audio de origen  $N$  como entrada, el número de canales codificados producidos por el codificador es también  $N$ . Los canales codificados pueden corresponderse de uno en uno con los canales origen, o los canales codificados pueden ser canales de transformación codificada multicanal. Cuando la complejidad de la codificación del origen hace difícil la compresión o cuando la memoria intermedia del codificador está llena, sin embargo, el codificador puede alterar o soltar (esto es, no codificar) uno o más de los canales audio de entrada originales. Esto puede llevarse a cabo para reducir la complejidad de la codificación y mejorar la calidad global percibida de la señal audio. Para un preprocesamiento junto motivado por la calidad, el codificador lleva a cabo el preprocesamiento multicanal en reacción a la calidad audio medida para controlar suavemente la calidad audio global y la separación de canales.

Por ejemplo, el codificador puede alterar la imagen audio multicanal para hacer que uno o más canales sean menos críticos de forma que los canales sean eliminados en el codificador ya reconstruidos en el descodificador como canales “fantasma”. La supresión directa de canales puede tener un efecto dramático sobre la calidad, de forma que se lleva únicamente a cabo cuando la complejidad de la codificación es muy alta o la memoria intermedia está tan llena que no puede conseguirse por otros medios una buena calidad de reproducción.

El codificador puede indicar al descodificador que acción adoptar cuando el número de canales codificados es inferior al número de canales para emitir de salida. A continuación, puede utilizarse una transformación de posprocesamiento multicanal en el descodificador para crear canales fantasma, de acuerdo con lo descrito más adelante en la sección titulada “Postprocesamiento Multicanal”. O, el codificador puede señalar al descodificador que lleve a cabo el postprocesamiento multicanal para otra finalidad.

La Figura 8 muestra otra técnica generalizada (800) de preprocesamiento multicanal. El codificador lleva a cabo (810) el preprocesamiento multicanal sobre los datos audio multicanal (805) de dominio temporal, produciendo unos datos audio transformados (815) en el dominio temporal. Por ejemplo, el preprocesamiento implica una transformación general  $N$  a  $N$ , donde  $N$  es el número de canales. El codificador multiplica unas muestras  $N$  con una matriz  $A$ .

$$y_{\text{pre}} = A_{\text{pre}} \cdot x_{\text{pre}} \quad (4),$$

donde  $x_{\text{pre}}$  e  $y_{\text{pre}}$  son la entrada de los canales  $N$  hasta la salida desde el preprocesamiento, y  $A_{\text{pre}}$  es una matriz de transformación  $N \times N$  con elementos valorados reales (esto es, continuos). La matriz  $A_{\text{pre}}$  puede escogerse para incrementar la correlación intercanal en  $y_{\text{pre}}$  en comparación con  $x_{\text{pre}}$ . Esto reduce la complejidad para el resto del codificador, pero al coste de la separación de los canales perdidos.

La salida  $y_{pre}$  es a continuación alimentada al resto del codificador, el cual codifica (820) los datos utilizando las técnicas mostradas en la Figura 6 u otras técnicas de compresión, produciendo los datos multicanal codificados (825).

La sintaxis utilizada por el codificador y el decodificador posibilita la descripción de matrices de transformación multicanal de postprocesamiento generales o predefinidas, las cuales pueden variar o ser activadas/o desactivadas sobre una base de trama por trama. El codificador utiliza esta flexibilidad para limitar las discapacidades de imagen estéreo/envolvente, sacrificando la separación de canales para una calidad global en determinadas circunstancias mediante un incremento artificial de la correlación intercanal. Como una alternativa, el decodificador y el codificador utilizan otra sintaxis para el pre y postprocesamiento multicanal, por ejemplo, uno que posibilita cambios en las matrices de transformación sobre una base distinta de trama a trama.

Las Figuras 9a-9e muestran unas matrices de transformación de preprocesamiento multicanal (900-904) utilizadas para incrementar artificialmente la correlación entre canales bajo determinadas circunstancias del codificador. El codificador conmuta entre matrices de preprocesamiento para cambiar cuánta relación intercanal es artificialmente incrementada entre los canales izquierdo, derecho, y central, y entre los canales trasero izquierdo y trasero derecho, en un entorno de reproducción del canal 5.1.

En una implementación, a velocidades de transmisión de bits baja, el codificador evalúa la calidad de la señal audio reconstruida con respecto a un periodo de tiempo y, dependiendo del resultado, selecciona una de las matrices de preprocesamiento. La medición de la calidad evaluada por el codificador es una Relación Ruido a Excitación [“NER”], la cual es la relación de la energía de la pauta de ruido para un clip de audio reconstruido con respecto a la energía del clip de audio digital original. Unos valores *NER* bajos indican una buena calidad, y unos valores *NER* altos indican una calidad deficiente. El codificador evalúa la *NER* respecto una o más tramas previamente codificadas. Para una información adicional acerca de la *NER* y otras mediciones de la calidad, véase la Solicitud de Patente estadounidense con el número de serie 10/017,861, titulada “Técnicas de Medición de la Calidad Audio Perceptual” [“Techniques for Measurement of Perceptual Audio Quality”], depositada el 14 de Diciembre de 2001. Como una alternativa, el codificador utiliza otra medición de la calidad, la totalidad de la memoria descriptiva y/o algunos otros criterios para seleccionar una matriz de transformación de preprocesamiento y el codificador evalúa un periodo diferente de señal audio multicanal.

Volviendo a los ejemplos mostrados en las Figuras 9a-9e, a velocidades de transmisión de bits bajas, el codificador lentamente cambia la matriz de transformación de preprocesamiento en base a la *NER*  $n$  de una extensión particular del clip de audio. El codificador compara el valor de  $n$  con los valores de umbral  $n_{bajo}$  y  $n_{alto}$ , los cuales dependen de la implementación. En una implementación,  $n_{bajo}$  y  $n_{alto}$  tienen los valores predeterminados  $n_{bajo} = 0,05$  y  $n_{alto} = 0,1$ . Como una alternativa,  $n_{bajo}$  y  $n_{alto}$  tienen valores diferentes o valores que cambian con el tiempo como reacción a la velocidad de transmisión de bits u otros criterios, o el codificador cambia entre un número diferente de matrices.

Un valor bajo de  $n$  (por ejemplo,  $n \leq n_{bajo}$ ) indica una codificación de buena calidad. De forma que, el codificador utiliza la matriz de identidad  $A_{bajo}$  (900) mostrada en la Figura 9a efectivamente desactivando el preprocesamiento.

Por otro lado, un valor alto de  $n$  (por ejemplo,  $n \geq n_{alto}$ ) indica una codificación de calidad deficiente. De forma que, el codificador utiliza la matriz  $A_{alto, 1}$  (902) mostrada en la Figura 9c. La matriz  $A_{alto, 1}$  (902) introduce una severa distorsión de imágenes envolventes, pero al mismo tiempo impone una correlación muy alta entre los canales izquierdo, derecho, y central, lo que mejora la subsecuente eficacia de la codificación mediante la reducción de la complejidad. El canal central transformado multicanal es la media de los canales izquierdo, central izquierdo, derecho y central originales. La matriz  $A_{alto, 1}$  (902) comprende también la separación de canales entre los canales traseros - los canales trasero izquierdo y trasero derecha introducidos son promediados.

Un valor intermedio de  $n$  (por ejemplo  $n_{bajo} < n < n_{alto}$ ) indica una codificación de calidad intermedia, de forma que, el codificador puede utilizar la matriz intermedia  $A_{inter, 1}$  (901) mostrada en la Figura 9b. En la matriz intermedia  $A_{inter, 1}$  (901), el factor  $\alpha$  mide la posición relativa de  $n$  entre  $n_{bajo}$  y  $n_{alto}$ .

$$\alpha = \frac{n - n_{bajo}}{n_{alto} - n_{bajo}}$$

La matriz intermedia  $A_{inter, 1}$  (901) gradualmente transiciona desde la matriz de identidad  $A_{bajo}$  (900) a la matriz de baja calidad  $A_{alto, 1}$  (902).

Para las matrices  $A_{inter, 1}$  (901) y  $A_{alto, 1}$  (902) mostradas en las Figuras 9b y 9c, el codificador explota más adelante la redundancia entre los canales para los cuales el codificador artificialmente incrementó la correlación entre canales, y el codificador no necesita instruir al decodificador para llevar a cabo ningún postprocesamiento multicanal para esos canales.

Cuando el decodificador tiene la capacidad para llevar a cabo el postprocesamiento multicanal, el codificador puede delegar la reconstrucción del canal central al decodificador. En este caso, cuando el valor *NER*  $n$  indica una codificación de calidad deficiente, el codificador utiliza la matriz  $A_{alto, 2}$  (904) mostrada en la Figura 9e, con la cual el canal central de entrada se fuga por el interior de los canales izquierdo y derecho. En la salida, el canal central es cero, reduciendo la complejidad de la codificación.

$$\begin{bmatrix} \left( \frac{a}{1.5} + \frac{.5 \cdot c}{1.5} \right) \\ \left( \frac{b}{1.5} + \frac{.5 \cdot c}{1.5} \right) \\ 0 \\ d \\ \frac{e+f}{2} \\ \frac{e+f}{2} \end{bmatrix} = A_{\text{alto}, 2} \cdot \begin{bmatrix} a \\ b \\ c \\ d \\ e \\ f \end{bmatrix}$$

Cuando el codificador utiliza la matriz de transformación de preprocesamiento  $A_{\text{alto}, 2}$  (904), el codificador (a través del flujo de bits) instruye al decodificador para crear un centro fantasma mediante la promediación de los canales izquierdo y derecho descodificados. Las transformaciones multicanal posteriores del codificador pueden explotar la redundancia entre los canales trasero izquierdo y trasero derecho (sin postprocesamiento), o el codificador puede instruir al decodificador para que lleve a cabo algún postprocesamiento multicanal para los canales trasero izquierdo y derecho.

Cuando el valor  $NER\ n$  indica una codificación de calidad intermedia, el codificador puede utilizar la matriz intermedia  $A_{\text{inter}, 2}$  (903) mostrada en la Figura 9d para llevar a cabo la transición de las matrices mostrada en las Figuras 9a y 9e.

La Figura 10 muestra una técnica (1000) de preprocesamiento multicanal en la cual la matriz de transformación potencialmente cambia sobre una base de trama por trama. El cambio de la matriz de transformación puede conducir a un ruido audible (por ejemplo, pum) en la salida final si no se maneja cuidadosamente. Para evitar la introducción de ruidos pum, el codificador gradualmente efectúa una transición de una matriz de transformación a otra entre tramas.

El codificador primeramente establece (1010) la matriz de transformación de preprocesamiento, de acuerdo con lo descrito anteriormente. El codificador a continuación determina (1020) si la matriz de la trama actual es diferente de la matriz de la trama anterior (si había una trama anterior). Si la trama actual es la misma o no hay una matriz anterior, el codificador aplica (1030) la matriz a las muestras audio de entrada para la trama actual. De no ser así, el codificador aplica (1040) una matriz de transformación mezclada a las muestras audio de entrada para la trama actual. La función de mezcla depende de la implementación. En una implementación, en la muestra  $i$  de la trama actual, el codificador utiliza una matriz mezclada a corto plazo  $A_{\text{pre}, i}$ .

$$A_{\text{pre}, i} = \frac{\text{NumSamples} - i}{\text{NumSamples}} A_{\text{pre}, \text{prev}} + \frac{i}{\text{NumSamples}} A_{\text{pre}, \text{actual}} \quad (6),$$

donde  $A_{\text{pre}, \text{prev}}$  y  $A_{\text{pre}, \text{actual}}$  son las matrices de preprocesamiento para las tramas previa y actual, respectivamente, y  $\text{NumMuestras}$  es el número de muestras de la trama actual. Como una alternativa, el codificador utiliza otra función de mezcla para suavizar las discontinuidades de las matrices de transformación de preprocesamiento.

A continuación, el codificador codifica (1050) los datos audio multicanal para la trama, utilizando las técnicas mostradas en la Figura 6 u otras técnicas de compresión. El codificador repite la técnica (1000) sobre una base de trama por trama. Como una alternativa, el codificador cambia el preprocesamiento multicanal sobre alguna otra base.

#### IV. Configuración de Mosaicos

En algunas formas de realización, un codificador, como por ejemplo el codificador (600) de la Figura 6 agrupa ventanas de la señal audio multicanal en mosaicos para su codificación subsecuente. Esto proporciona al codificador la flexibilidad de utilizar configuraciones de ventanas diferentes para los diferentes canales de una trama, aunque permitiendo también transformaciones multicanal en varias combinaciones de canales para la trama. Un decodificador, como por ejemplo el decodificador (700) de la Figura 7, trabaja con mosaicos durante la descodificación.

Cada canal puede tener una configuración de ventanas independiente de los demás canales. Las ventanas que tienen unos tiempos de partida y final idénticos se consideran como parte de un mosaico. Un mosaico puede tener uno o más canales, y el codificador lleva a cabo transformaciones multicanal para los canales de un mosaico.

## ES 2 316 679 T3

La Figura 11a muestra una configuración de mosaicos ejemplar (1100) para una trama de audio estéreo. En la Figura 11a, cada mosaico incluye una sola ventana. Ninguna ventana en uno u otro canal del audio estéreo empieza o finaliza al mismo tiempo que otra ventana del otro canal.

La Figura 11b muestra una configuración de mosaicos ejemplar (1101) para una trama de señal audio de un canal 5.1. La configuración de mosaicos (1101) incluye siete mosaicos, numerados del 0 al 6. El mosaico 0 incluye unas muestras procedentes de los canales 0, 2, 3 y 4 y abarca el primer cuarto de la trama. El mosaico 1 incluye las tramas procedentes del canal 1 y abarca la primera mitad de la trama. El mosaico 2 incluye las tramas procedentes del canal 5 y abarca la entera trama. El mosaico 3 es como el mosaico 0 pero abarca el segundo cuarto de la trama. Los mosaicos 4 y 6 incluyen las muestras de los canales 0, 2 y 3, y abarcan los tercero y cuarto cuartos, respectivamente, de la trama. Finalmente, el mosaico 5 incluye las muestras procedentes de los canales 1 y 4 y abarca la última mitad de la trama. Como se muestra en la Figura 11b, un mosaico concreto puede incluir ventanas de canales no contiguos.

La Figura 12 muestra una técnica generalizada (1200) para configurar mosaicos de una trama de señal audio multicanal. El codificador establece (1210) las configuraciones de ventanas para los canales de la trama, tabicando cada canal en ventanas de tamaño variable para efectuar un compromiso entre la resolución de tiempo y la resolución de frecuencia. Por ejemplo, un configurador de tabicadores/mosaicos del codificador tabica cada canal independientemente de los demás canales de la trama.

El codificador a continuación agrupa (1220) las ventanas procedentes de los diferentes canales en mosaicos para la trama. Por ejemplo, el codificador coloca las ventanas procedentes de diferentes canales en un solo mosaico si las ventanas tienen posiciones de partida idénticas y posiciones finales idénticas. Como una alternativa, el codificador utiliza criterios distintos de o además de las posiciones de partida/final para determinar qué secciones de los diferentes canales se agrupan conjuntamente formando un mosaico.

En una implementación, el codificador lleva a cabo el agrupamiento de mosaicos (1220) después (e independientemente) de la regulación (1210) de las configuraciones de ventanas para una trama. En otras implementaciones, el codificador conjuntamente regula (1210) las configuraciones de ventanas y agrupa (1220) las ventanas en mosaicos, por ejemplo, para favorecer la correlación del tiempo (que utiliza ventanas más largas) o la correlación de los canales (poniendo más canales en los mosaicos simples), o para controlar el número de mosaicos obligando a las ventanas a encajar dentro de un conjunto concreto de mosaicos.

El codificador a continuación envía (1230) la información de la configuración de mosaicos para la trama para la emisión de salida con los datos audio codificados. Por ejemplo, el configurador de tabicadores/mosaico del codificador envía la información de los miembros de los canales y del tamaño de los mosaicos para los mosaicos hasta un MUX. Como una alternativa, el codificador envía otra información que especifique las configuraciones de los mosaicos. En una implementación, el codificador envía (1230) la información de la configuración de mosaicos después del agrupamiento (1220) de los mosaicos. En otras implementaciones, el codificador lleva a cabo estas acciones simultáneamente.

La Figura 13 muestra una técnica (1300) de configurar mosaicos y enviar información de la configuración de mosaicos para una trama de señales audio multicanal de acuerdo con una sintaxis de flujo de bits concreta. La Figura 13 muestra la técnica (1300) llevada a cabo por el codificador para introducir la información en el flujo de bits; el decodificador lleva a cabo una técnica correspondiente (lectura de indicadores, obtención de información de la configuración para los mosaicos concretos, etc.) para extraer la información de la configuración de los mosaicos para la trama de acuerdo con la sintaxis del flujo de bits. Como una alternativa, el decodificador y el codificador utilizan otras sintaxis para una o más de las opciones mostradas en la Figura 13, por ejemplo, una que utiliza indicadores diferentes o una ordenación diferente.

El codificador inicialmente verifica (1310) si ninguno de los canales de la trama está dividido en ventanas. Si es así, el codificador envía (1312) un bit indicador (que indica que no hay ningún canal dividido), y a continuación sale. Así, un único bit indica si una trama determinada tiene un solo mosaico o tiene múltiples mosaicos.

Por otro lado, si al menos un canal está dividido en ventanas, el codificador verifica (1320) si todos los canales de la trama tienen la misma configuración de ventanas. Si es así, el codificador envía (1322) un bit indicador (que indica que todos los canales tienen la misma configuración de ventanas - cada mosaico de ventana tiene todos los canales) y una secuencia de los tamaños de los mosaicos, y a continuación sale. Así, el único bit indica si los canales tienen todos la misma configuración (como en un flujo de bits de un codificador convencional) o tienen una configuración de mosaicos flexible.

Si al menos algunos canales tienen configuraciones de ventanas diferentes, el codificador escanea las posiciones de las muestras de la trama para identificar las ventanas que tienen tanto la misma posición de partida como la misma posición final. Pero primeramente, el codificador marca (1330) todas las posiciones de muestra de la trama como no agrupadas. El codificador a continuación escanea (1340) la siguiente posición de muestras no agrupadas de la trama de acuerdo con un patrón de escaneo por canal/tiempo. En una implementación, el codificador escanea todos los canales en un momento determinado buscando las posiciones de muestra no agrupadas, a continuación repite para la siguiente posición de muestra en el tiempo, etc. En otras implementaciones, el codificador utiliza otra pauta de escaneo.

Para la posición de muestra no agrupadas detectadas, el codificador agrupa (1350) las mismas ventanas juntas en un mosaico. En particular, el codificador agrupa las ventanas que empiezan en la posición de partida de la ventana que incluye la posición de muestras no agrupadas detectadas, y que también terminan en la misma posición que la ventana que incluye la posición de muestras no agrupadas detectadas. En la trama mostrada en la Figura 11b, por ejemplo, el

5 codificador detectaría primeramente la posición de las muestras al principio del canal 0. El codificador agruparía las ventanas con una longitud de un cuarto de trama a partir de los canales 0, 2, 3 y 4 conjuntamente en un mosaico, dado que estas ventanas tienen cada una la misma posición de partida y la misma posición final que las otras ventanas del mosaico.

10 El codificador a continuación envía (1360) la información de la configuración de los mosaicos que especifica el mosaico para la emisión de salida con los datos audio codificados. La información de la configuración de los mosaicos incluye el tamaño de los mosaicos y un mapa indicativo de qué canales con posiciones de las muestras no agrupadas de la trama en ese punto están dentro del mosaico. El mapa de canales incluye un bit por canal posible para el mosaico. En base a la secuencia de información de los mosaicos, el descodificador determina donde un mosaico empieza y

15 termina en una trama. El codificador reduce la velocidad de transmisión de bits para el mapa de los canales teniendo en cuenta qué canales puede haber en el mosaico. Por ejemplo, la información para el mosaico 0 de la Figura 11b incluye el tamaño de mosaico, y un patrón binario "101110" para indicar que los canales 0, 2, 3, y 4 son parte de mosaico. Después de ese punto, solo las posiciones de las muestras de los canales 1 y 5 no están agrupadas. De forma que la información del mosaico 1 incluye el tamaño de mosaico y la pauta binaria "10" para indicar que el canal 1 es

20 parte del mosaico pero que el canal 5 no lo es. Esto ahorra cuatro bits de la pauta binaria. La información de mosaico del mosaico 2 a continuación incluye únicamente el tamaño de mosaico (y no el mapa de canal), dado que el canal 5 es el único canal que puede tener una ventana de partida en el mosaico 2. La información de mosaico para el mosaico 3 incluye el tamaño de mosaico y la pauta binaria "1111" dado que los canales 1 y 5 han agrupado las posiciones dentro del rango del mosaico 3. Como una alternativa, el codificador y el descodificador utilizan otra técnica para señalar

25 pautas de canal en la sintaxis.

El codificador a continuación marca (1370) las posiciones de las muestras de las ventanas en el mosaico como agrupadas y determina (1380) si continuar o no. Si no hay más posiciones de las muestras no agrupadas en la trama, el codificador sale. En otro caso, el codificador escanea (1340) la siguiente posición de las muestras no agrupadas de

30 la trama de acuerdo con la pauta de escaneo canal/tiempo.

## V. Transformaciones Flexibles Multicanal

En algunas formas de realización, un codificador, como por ejemplo el codificador (600) de la Figura 6, lleva a

35 cabo unas transformaciones flexibles multicanal que aprovechan eficazmente la correlación entre canales. Un descodificador, como por ejemplo el descodificador (700) de la Figura 7, lleva a cabo las correspondientes transformaciones multicanal inversas.

Específicamente, el codificador y el descodificador llevan a cabo una o más de las siguientes funciones para mejorar

40 las transformaciones multicanal en diferentes situaciones.

1. El codificador lleva a cabo la transformación multicanal después de la ponderación sensorial; y el descodificador lleva a cabo la correspondiente transformación multicanal inversa antes de la ponderación inversa. Ello reduce el desenmascaramiento del ruido de cuantificación a través de los canales después de la transformación multicanal

45 inversa.

2. El codificador y el descodificador agrupan los canales para las transformaciones multicanal para limitar qué canales son transformados conjuntamente.

3. El codificador y el descodificador selectivamente activan/desactivan las transformaciones multicanal en el nivel

50 de bandas de frecuencia para controlar que bandas son transformadas conjuntamente.

4. El codificador y el descodificador utilizan unas transformaciones multicanal jerárquicas para limitar la complejidad computacional (especialmente del descodificador).

55

5. El codificador y el descodificador utilizan matrices de transformación multicanal predefinidas para reducir la velocidad de transmisión de bits utilizada para especificar las matrices de transformación.

6. El codificador y el descodificador utilizan parámetros de factorización de Givens cuantificados basados en la

60 rotación para especificar las matrices de transformación multicanal para la eficacia de los bits.

## A. Transformación Multicanal sobre Señales Audio Multicanal Ponderadas

En algunas formas de realización, el codificador sitúa la transformación multicanal después de la ponderación

65 perceptual (y el descodificador sitúa la transformación multicanal inversa antes de la ponderación inversa) de forma que la señal fugada a través del canal es controlada, es susceptible de medición, y tiene un espectro como la señal original.

La Figura 14 muestra una técnica (1400) para llevar a cabo una o más transformaciones multicanal después de la ponderación perceptual del codificador. El codificador pondera perceptualmente (1410) la señal multicanal, por ejemplo, aplicando factores de ponderación a la señal audio multicanal en el dominio de frecuencia. En algunas implementaciones, el codificador aplica tanto los factores de ponderación como los modificadores de paso de cuantificación por canal a los datos audio multicanal antes de la(s) transformación(es) multicanal.

El codificador a continuación lleva a cabo (1420) una o más transformaciones multicanal sobre los datos audio ponderados; por ejemplo, de acuerdo con lo descrito más adelante. Finalmente, el modificador cuantifica (1430) los datos audio multicanal transformados.

La Figura 15 muestra una técnica (1500) para llevar a cabo una transformación multicanal inversa antes de la ponderación inversa del decodificador. El decodificador lleva a cabo (1510) una o más transformaciones multicanal inversas sobre los datos audio cuantificados, por ejemplo, de acuerdo con lo descrito más adelante. En particular, el decodificador recoge muestras de los múltiples canales en un índice de frecuencias concreto formando un vector  $x_{mc}$  y lleva a cabo la transformación multicanal inversa  $A_{mc}$  para generar la salida  $y_{mc}$ .

$$y_{mc} = A_{mc} \cdot x_{mc} \quad (7).$$

A continuación, el decodificador efectúa una cuantificación inversa y una ponderación inversa (1520) de la señal audio multicanal, coloreando la salida de la transformación multicanal inversa con máscara(s). Así, el escape que se produce a través de los canales (debido a la cuantificación) es espectralmente conformado de tal manera que la audibilidad de la señal fugada puede medirse y controlarse, y la fuga de los otros canales en un canal reconstruido determinado es conformado espectralmente como la señal no corrompida original del canal determinado. (En algunas implementaciones, los modificadores de paso de cuantificación por canal permite también que el codificador haga que la señal reconstruida tenga una calidad aproximadamente igual a través de todos los canales reconstruidos).

#### B. Grupos de Canales

En algunas formas de realización, el codificador y el decodificador agrupan los canales de las transformaciones multicanal para limitar qué canales se transforman conjuntamente. Por ejemplo, en formas de realización que utilizan una configuración de mosaicos, el codificador determina qué canales situados dentro de un mosaico se correlacionan, y agrupan los canales correlacionados. Como una alternativa, un codificador y un decodificador no utilizan una configuración de mosaicos, pero siguen agrupando los canales para las tramas o en algún otro nivel.

La Figura 16 muestra una técnica (1600) para agrupar canales de un mosaico para la transformación multicanal en una implementación. En la técnica (1600), el codificador considera correlaciones por pares de entre las señales de los canales así como las correlaciones entre bandas en algunos casos. Como una alternativa, un codificador considera factores distintos y/o adicionales al agrupar canales para una transformación multicanal.

En primer lugar, el codificador obtiene (1610) los canales para un mosaico. Por ejemplo, en la configuración de mosaicos mostrada en la Figura 11b, el mosaico 3 tiene cuatro canales en ella: 0, 2, 3, y 4.

El codificador computa (1620) las correlaciones por pares entre las señales de los canales y a continuación agrupa (1630) los canales en la medida correspondiente. Supóngase que para el mosaico 3 de la Figura 11b, los canales 0 y 2 están correlacionados por pares, pero ninguno de esos canales está correlacionado por pares con el canal 3 o el canal 4, y el canal 3 no está correlacionado por pares con el canal 4. El codificador agrupa (1630) los canales 0 y 2 entre sí, sitúa el canal 3 en un grupo separado y sitúa el canal 4 en otro grupo separado.

Un canal que no está correlacionado por pares con ninguno de los canales de un grupo puede seguir siendo compatible con ese grupo. De esta forma, para los canales que son incompatibles con un grupo, el codificador opcionalmente verifica (1640) la compatibilidad en el nivel de bandas y ajusta (1650) los uno o más grupos de canales de la forma correspondiente. En particular, esto identifica los canales que son compatibles con un grupo en algunas bandas, pero incompatibles en algunas otras bandas. Por ejemplo, supóngase que el canal 4 del mosaico 3 de la Figura 11b es efectivamente compatible con los canales 0 y 2 en la mayoría de las bandas, pero que la incompatibilidad en unas pocas bandas oblicúa los resultados de la correlación por pares. El codificador ajusta (1650) los grupos para situar los canales 0, 2, y 4 conjuntamente, dejando el canal 3 en su propio grupo. El codificador puede también llevar a cabo dicha comprobación cuando dichos canales están (globalmente) correlacionados, pero tienen bandas incompatibles. La desconexión de la transformación en esas bandas incompatibles mejora la correlación de esas bandas que de hecho obtienen una transformación multicanal mejorada, y por tanto mejora la eficacia de la codificación.

Un canal de un mosaico determinado pertenece a un grupo de canales. Los canales de un grupo de canales no necesitan ser contiguos. Un solo mosaico puede incluir múltiples grupos de canales y cada grupo de canales puede tener una transformación multicanal asociada diferente. Después de decidir qué canales son compatibles, el codificador introduce la información del grupo de canales dentro del flujo de bits.

La Figura 17 muestra una técnica (1700) para extraer una información de un grupo de canales y de una información de una transformación multicanal para un mosaico a partir de un flujo de bits de acuerdo con una sintaxis de flujo de bits concreta, con independencia de cómo el codificador computa los grupos de canales. La Figura 17 muestra la técnica (1700) llevada a cabo por el descodificador y extrae la información a partir del flujo de bits; el codificador

5 lleva a cabo una técnica correspondiente a la información del grupo de canales para formatear la información de la transformación multicanal para el mosaico de acuerdo con la sintaxis del grupo de bits. Como una alternativa, el descodificador y el codificador utilizan otra sintaxis para una o más de las opciones mostradas en la Figura 17.

En primer lugar, el descodificador inicializa diversas variables utilizadas en la técnica (1700). El descodificador

10 establece (1710) *#ChannelsToVisit* igual al número de canales del mosaico *#ChannelsInTile* y fija (1712) el número de grupos de canales *#ChannelsGroups* en 0.

El descodificador verifica (1720) si *#ChannelsToVisit* es mayor de 2. Si no, el descodificador verifica (1730) si *#ChannelsToVisit* es igual a 2. Si es así, el descodificador descodifica (1740) la transformación multicanal para el

15 grupo de dos canales, por ejemplo, utilizando una técnica descrita más adelante. La sintaxis posibilita que cada grupo de canales tenga una transformación multicanal diferente. Por otro lado, si *#ChannelsToVisit* es igual a 1 o 0, el descodificador sale sin descodificar una transformación multicanal.

Si *#ChannelsToVisit* es mayor de 2, el descodificador descodifica (1750) la máscara de los canales para un grupo

20 del mosaico. Específicamente, el descodificador lee *#ChannelsToVisit* a partir del flujo de bits para la máscara de los canales. Cada bit de la máscara de los canales indica si un canal concreto está o no en el grupo de canales. Por ejemplo, si la máscara de los canales es "10110" entonces el mosaico incluye 5 canales, y los canales 0, 2, y 3 están en el grupo de canales.

El descodificador a continuación cuenta (1760) el número de canales del grupo y descodifica (1770) la transformación multicanal para el grupo, por ejemplo, utilizando una técnica descrita más adelante. El descodificador actualiza

25 (1780) *#ChannelsToVisit* mediante la sustracción del número contado de canales del grupo de canales actual, incrementa (1790) *#ChannelsGroup*, y verifica (1720) si el número de canales dejados de visitar *#ChannelsToVisit* es mayor de 2.

Como una alternativa, en formas de realización que no usan configuraciones de mosaicos, el descodificador extrae la información del grupo de canales y la información de la transformación multicanal para una trama o en algún otro nivel.

### 35 C. Control de Activación/Desactivación de Bandas para una Transformación Multicanal

En algunas formas de realización, el codificador y el descodificador selectivamente activan/desactivan las transformaciones multicanal en el nivel de bandas de frecuencia para controlar qué bandas son transformadas conjuntamente. De esta forma, el codificador y el descodificador selectivamente excluyen las bandas que no son compatibles en las transformaciones multicanal. Cuando la transformación multicanal es desactivada para una banda concreta, el codificador y el descodificador utilizan la transformación de identidad para esa banda, pasando a través de los datos en esa banda sin alterarlos.

Las bandas de frecuencia son bandas críticas o bandas de identificación. El número de bandas de frecuencia, se refiere a la frecuencia de muestreo de los datos audio y al tamaño de los mosaicos. En general, cuanto mayor es la frecuencia de muestreo o mayor es el tamaño de los mosaicos, mayor es el número de las bandas de frecuencia.

En algunas implementaciones, el codificador selectivamente activa/desactiva las transformaciones multicanal en el nivel de bandas de frecuencia para canales de un grupo de canales de un mosaico. El codificador puede activar/desactivar las bandas cuando el codificador agrupa los canales para un mosaico o después del agrupamiento de canales para el mosaico. Como una alternativa, un codificador y un descodificador no utilizan una configuración de mosaicos, pero siguen activando/desactivando las transformaciones multicanal en bandas de frecuencia para una trama o en algún otro nivel.

La Figura 18 muestra una técnica (1800) para incluir selectivamente bandas de frecuencia de un grupo de canales de una transformación multicanal de una implementación. En la técnica (1800), el codificador considera unas correlaciones por pares entre las señales de los canales en una banda para determinar si se habilita o deshabilita la transformación multicanal para la banda. Como una alternativa, un codificador considera factores distintos y/o adicionales al activar o desactivar selectivamente las bandas de frecuencia para una transformación multicanal.

En primer lugar, el codificador obtiene (1810), los canales para un grupo de canales, por ejemplo, de acuerdo con lo descrito con referencia a la Figura 16. El codificador a continuación computa (1820) las correlaciones por pares entre las señales de los canales para bandas de frecuencia diferentes. Por ejemplo, si el grupo de canales incluye dos canales, el codificador computa unas correlaciones por pares entre algunos o todos los pares respectivos de los canales

65 en cada banda de frecuencia.

El codificador a continuación activa o desactiva (1830) para la transformación multicanal para el grupo de canales. Por ejemplo, si el grupo de canales incluye dos canales, el codificador posibilita la transformación multicanal para

una banda si la correlación por pares en la banda satisface un concreto umbral. O, si el grupo de canales incluye dos o más canales, el codificador posibilita la transformación multicanal para una banda si cada uno o una mayoría de las correlaciones por pares en la banda satisface un concreto umbral. En formas de realización alternativas, en lugar de activar o desactivar una banda de frecuencia concreta para todos los canales, el codificador activa la banda para algunos canales y la desactiva para los demás canales.

Después de decidir qué bandas están incluidas en las transformaciones multicanal, el codificador introduce la información de activación/desactivación de las bandas dentro del flujo de bits.

La Figura 19 muestra una técnica (1900) para extraer la información de activación/desactivación de las bandas para una transformación multicanal para un grupo de canales de un mosaico a partir de un flujo de bits de acuerdo con una sintaxis de flujo de bits concreta, con independencia de cómo el codificador decide si activar o desactivar las bandas. La Figura 19 muestra la técnica (1900) llevada a cabo por el descodificador para extraer información a partir del grupo de bits; el codificador lleva a cabo la técnica correspondiente a la información para formatear la activación/desactivación de las bandas para el grupo de canales de acuerdo con la sintaxis del flujo de bits. Como una alternativa, el codificador y el descodificador utilizan otra sintaxis para una o más de las opciones mostradas en la Figura 19.

En algunas implementaciones, el descodificador lleva a cabo la técnica (1900) como parte de la descodificación de la transformación multicanal (1740 o 1770) de la técnica (1700). Como una alternativa, el descodificador lleva a cabo la técnica (1900) separadamente.

El descodificador obtiene (1910) un bit y verifica (1920) el bit para determinar si todas las bandas están habilitadas para el grupo de canales. Si es así, el descodificador habilita (1930) la transformación multicanal para todas las bandas del grupo de canales.

Por otro lado, si el bit indica que todas las bandas no están habilitadas para el grupo de canales, el descodificador descodifica (1940) la máscara de las bandas para el grupo de canales. Específicamente, el descodificador lee un número de bits entre el flujo de bits, donde el número es el número de bandas para el grupo de canales. Cada bit de la máscara de bandas indica si una banda concreta está activada o desactivada para el grupo de canales. Por ejemplo, si la máscara de bandas es "111111110110000" entonces el grupo de canales incluye 15 bandas, y las bandas 0, 1, 2, 3, 4, 5, 6, 7, 9, y 10 están activadas para la transformación multicanal. El descodificador a continuación habilita (1950) la transformación multicanal para las bandas indicadas.

Como una alternativa, en formas de realización que no utilizan las configuraciones de mosaicos, el descodificador extrae la información de la activación/desactivación de las bandas para una trama o en algún otro nivel.

#### D. Transformaciones Jerárquicas Multicanal

En algunas formas de realización, el codificador y el descodificador utilizan transformaciones jerárquicas multicanal para limitar la complejidad computacional, especialmente en el codificador. Con la transformación jerárquica, un codificador divide una transformación global en múltiples etapas, reduciendo la complejidad computacional de las etapas individuales y en algunos casos reduciendo la cantidad de información requerida para especificar la(s) transformación(es) multicanal. Utilizando esta estructura en cascada, el codificador emula la transformación global mayor con transformaciones menores, hasta una cierta precisión. El descodificador lleva a cabo una transformación inversa jerárquica correspondiente.

En algunas formas de realización, cada etapa de la transformación jerárquica es idéntica en estructura y, en el flujo de bits, cada etapa se describe de forma independiente de las una o más distintas etapas. En particular, cada etapa tiene sus propios grupos de canales y una matriz de transformación multicanal por grupo de canales. En implementaciones alternativas, diferentes etapas tienen estructuras diferentes, el codificador y el descodificador utilizan una sintaxis de flujo de bits diferente, y/o las etapas utilizan otra configuración para los canales y las transformaciones.

La Figura 20 muestra una técnica generalizada (2000) para emular una transformación multicanal que utiliza una jerarquía de transformaciones multicanal más sencillas. La Figura 20 muestra una jerarquía de etapas  $n$ , donde  $n$  es el número de etapas de transformación multicanal. Por ejemplo, en una implementación,  $n$  es 2. Como una alternativa,  $n$  es más de 2.

El codificador determina (2010) una jerarquía de transformaciones multicanal para una transformación global. El codificador decide los tamaños de la transformación (esto es, el tamaño del grupo de canales) en base a la complejidad del descodificador que llevará a cabo las transformaciones inversas. O el codificador considera el perfil del descodificador/nivel del descodificador escogido como objetivo o algún otro criterio.

La Figura 21 es un gráfico que muestra una jerarquía ejemplar (2100) de transformaciones multicanal. La jerarquía (2100) incluye 2 etapas. La primera etapa incluye unos grupos y transformaciones de canales  $N + 1$ , numeradas de 0 a  $N$ ; la segunda etapa incluye grupos y transformaciones de canales  $M + 1$ , numeradas de 0 a  $M$ . Cada grupo de canales incluye uno o más canales. Para cada una de las transformaciones  $N + 1$  de la primera etapa, los canales de entrada son alguna combinación de los canales introducidos en el transformador multicanal. No todos los canales de entrada deben ser transferidos en la primera etapa. Uno o más canales de entrada pueden pasar inalterados por la primera etapa



(por ejemplo, el codificador puede incluir dichos canales en un grupo de canales que utilice una matriz de identidad). Para cada una de las transformaciones  $M + 1$  de la segunda etapa, los canales de entrada son alguna combinación de los canales de salida, entre la primera etapa, incluyendo canales que pueden haber pasado inalterados por la primera etapa.

Volviendo a la Figura 20, el codificador lleva a cabo (2020) la primera etapa de transformaciones multicanal, lleva a cabo la etapa siguiente de transformaciones multicanal, llevando a cabo finalmente (2030) la *enésima* etapa de transformaciones multicanal. Un descodificador lleva a cabo las correspondientes transformaciones multicanal inversas durante la descodificación.

En algunas implementaciones, los grupos de canales son los mismos en las múltiples etapas de la jerarquía, pero las transformaciones multicanal son diferentes. En dichos casos, y también en otros casos determinados, el codificador puede combinar la información de activación/desactivación de bandas de frecuencia para las múltiples transformaciones multicanal. Por ejemplo, supóngase que hay dos transformaciones multicanal y los mismos tres canales en el grupo de canales para cada uno. El codificador puede no especificar ninguna transformación/transformación de identidad en ambas etapas para la banda 0, únicamente la etapa 1 de transformación multicanal para la banda 1 (ninguna transformación de la etapa 2), únicamente una etapa 2 de transformación multicanal para la banda 2 (ninguna transformación de la etapa 1), ambas etapas de transformaciones multicanal para la banda 3, ninguna transformación en ambas bandas para la etapa 4, etc.

La Figura 22 muestra una técnica (2200) para extraer información para una jerarquía de transformaciones multicanal para grupos de canales entre un flujo de bits de acuerdo con una sintaxis de flujo de bits concreta. La Figura 22 muestra la técnica (2200) llevada a cabo por el descodificador para analizar el flujo de bits; el codificador lleva a cabo una técnica correspondiente para formatear la jerarquía de las transformaciones multicanal de acuerdo con la sintaxis del flujo de bits. Como una alternativa, el descodificador y el codificador utilizan otras sintaxis, por ejemplo, una que incluya indicadores adicionales y bits de señalización para más de dos etapas.

El descodificador primeramente fija (2210) un valor temporal  $iTmp$  igual al siguiente bit del flujo de bits. El descodificador a continuación verifica (2220) el valor del valor temporal, el cual señala si el descodificador debe o no descodificar (2230) del grupo de canales y la información de transformación multicanal para un grupo de la etapa 1.

Después de que el descodificador descodifica (2230) el grupo de canales y la información de transformación multicanal para un grupo de la etapa 1, el descodificador fija (2240) un  $iTmp$  igual al siguiente bit del flujo de bits. El descodificador verifica de nuevo (2220) el valor de  $iTmp$ , el cual señala si el flujo de bits incluye o no el grupo de canales y la información de transformación multicanal para cualquier grupo más de la etapa 1. Solo los grupos de canales con transformaciones de no identidad se especifican en la porción de la etapa 1 del flujo de bits; los canales que no se describen en la parte de la etapa 1 del flujo de bits se supone que son parte de un grupo de canales que utiliza una transformación de identidad.

Si el flujo de bits no incluye más grupos de canales y la información de transformación de grupos de la etapa 1, el descodificador (2250), el grupo de canales y la información de la transformación multicanal para todos los grupos de la etapa 2.

#### E. Transformaciones Predefinidas o Multicanal Personalizadas

En algunas formas de realización, el codificador y el descodificador utilizan matrices de transformaciones multicanal predefinidas para reducir la velocidad de transformación de bits utilizada para especificar las matrices de las transformaciones. El codificador selecciona entre múltiples tipos de matrices predefinidas disponibles y señala la matriz seleccionada del grupo de bits con un número pequeño (por ejemplo, 1, 2) de bits. Algunos tipos de matrices no requieren señalización adicional en el flujo de bits, pero otros grupos de matrices requieren señalización adicional. El descodificador extrae la información indicativa del tipo de matriz y (si es necesario) la información adicional que especifica la matriz.

En algunas implementaciones, el codificador y el descodificador utilizan los siguientes tipos de matrices predefinidas: de identidad, de Hadamard, de DCT tipo II, o una unidad arbitraria. Como una alternativa, el codificador y el descodificador utilizan tipos de matrices diferentes y/o predefinidas adicionales.

La Figura 9a muestra un ejemplo de una matriz de identidad para 6 canales en otro contexto. El codificador eficazmente especifica una matriz de identidad del flujo de bits utilizando bits indicadores, suponiendo que el número de dimensiones de la matriz de identidad es conocido tanto para el codificador como para el descodificador a partir de otra información (por ejemplo, el número de canales de un grupo).

Una matriz Hadamard tiene la forma siguiente:

$$A_{Hadamard} = \rho \begin{pmatrix} 0,5 & -0,5 \\ 0,5 & 0,5 \end{pmatrix} \quad (8),$$

donde  $\rho$  es un normalizador escalar ( $\sqrt{2}$ ). El codificador especifica eficazmente una matriz Hadamar para datos estéreo del flujo de datos utilizando bits indicadores.

Una matriz DCT tipo II tiene la forma siguiente:

$$A_{DCT, II} = \begin{pmatrix} a_{0,0} & a_{0,1} & \dots & a_{0,N-1} \\ a_{1,0} & a_{1,1} & \dots & a_{1,N-1} \\ \dots & \dots & \dots & \dots \\ a_{N-1,0} & a_{N-1,1} & \dots & a_{N-1,N-1} \end{pmatrix} \quad (9),$$

donde

$$a_{n,m} = k_m \cdot \cos \left[ \frac{m \cdot (n + 0,5) \pi}{N} \right] \quad (10),$$

y donde

$$k_m = \begin{cases} \sqrt{\frac{1}{N}} & m = 0 \\ \sqrt{\frac{2}{N}} & m > 0 \end{cases} \quad (11).$$

Para más información acerca de las matrices DCT tipo II, véase Rao *et al.*, *Transformación de Coseno Discreto*, Academic Press (1990). La matriz DCT tipo II puede tener cualquier tamaño (esto es, trabajar para cualquier tipo de canales). El codificador especifica eficazmente una matriz DCT tipo II del flujo de bits utilizando bits indicadores, suponiendo que el número de dimensiones para la matriz DCT tipo II es conocido tanto para el codificador como para el descodificador a partir de otra información (por ejemplo, el número de canales de un grupo).

## ES 2 316 679 T3

Una matriz cuadrada  $A_{\text{cuadrada}}$  es unitaria si su transposición es su inversa.

$$A_{\text{cuadrada}} \cdot A_{\text{cuadrada}}^T = A_{\text{cuadrada}}^T \cdot A_{\text{cuadrada}} = I \quad (12),$$

donde  $I$  es la matriz de identidad. El codificador utiliza matrices unitarias arbitrarias para especificar las transformaciones KLT para una supresión eficaz de la redundancia. El codificador especifica de manera eficaz la matriz unitaria del flujo de bits utilizando bits indicadores y una parametrización de la matriz. En algunas formas de realización, el codificador parametriza la matriz utilizando las rotaciones de factorización Givens cuantificadas, de acuerdo con lo descrito más adelante. Como una alternativa, el codificador utiliza otra parametrización.

La Figura 23 muestra una técnica (2300) de selección de un tipo de transformación multicanal entre varios tipos disponibles. El codificador selecciona un tipo de transformación sobre una base de grupos de canales grupo por grupo o en algún otro nivel.

El codificador selecciona (2310) un tipo de transformación multicanal entre múltiples tipos disponibles. Por ejemplo, los tipos disponibles incluyen la de identidad, la de Hadamard, la de DCT tipo II, la de unitaria arbitraria. Como una alternativa, los tipos incluyen tipos de matriz diferentes y/o adicionales. El codificador utiliza una matriz de identidad, Hadamard, o de DCT tipo II (mejor que una matriz unitaria arbitraria) si es posible o si se requiere para reducir los bits necesarios para especificar la matriz de transformación. Por ejemplo, el codificador utiliza una matriz de identidad, Hadamard o un DCT tipo II si la supresión de la redundancia es comparable o lo suficientemente próxima (mediante algunos criterios) a la supresión de la redundancia con la matriz unitaria arbitraria. O, el codificador utiliza una matriz de identidad, Hadamard, o DCT tipo II si el codificador debe reducir la velocidad de transmisión de bits. En una situación general, sin embargo, el codificador utiliza una matriz unitaria arbitraria para la máxima eficacia de la compresión.

El codificador a continuación aplica (2320) una transformación multicanal del tipo seleccionado a los datos audio multicanal.

La Figura 24 muestra una técnica (2400) para extraer un tipo de transformación multicanal entre varios tipos disponibles y llevar a cabo una transformación multicanal inversa. El decodificador extrae la información del tipo de transformación sobre una base de grupos de grupo por grupo de canales o en algún otro nivel.

El decodificador extrae (2410) un tipo de transformación multicanal entre múltiples tipos disponibles. Por ejemplo, los tipos disponibles incluyen la matriz de identidad, Hadamard, DCT tipo II, y unitaria arbitraria. Como una alternativa los tipos incluyen tipos de matrices diferentes y/o adicionales. Si es necesario, el decodificador extrae información adicional que especifica la matriz.

Después de la reconstrucción de la matriz, el decodificador aplica (2420) una transformación multicanal inversa del tipo seleccionado a los datos audio multicanal.

La Figura 25 muestra una técnica (2500) para extraer la información de la transformación multicanal para un grupo de canales entre un flujo de bits de acuerdo con una sintaxis de flujo de bits concreta. La Figura 25 muestra la técnica (2500) llevada a cabo por el decodificador para analizar el flujo de bits; el codificador lleva a cabo una técnica correspondiente para formatear la información de la transformación multicanal de acuerdo con la sintaxis del flujo de bits. Como una alternativa, el decodificador y el codificador utilizan otra sintaxis, por ejemplo una que utilice bits indicadores diferentes, una ordenación diferente, o diferentes tipos de información.

Inicialmente, el decodificador verifica (2510) si el número de canales del grupo *#ChannelsInGroup* es mayor de 1. Si no, el grupo de canales es para las señales audio mono y el decodificador utiliza (2512) una transformación de identidad para el grupo.

Si *#ChannelsInGroup* es mayor de 1, el decodificador verifica (2520) si *#ChannelsInGroup* es mayor de 2. Si no, el grupo de canales es para el audio estéreo, y el decodificador fija (2522) un valor temporal *iTmp* igual al bit siguiente del flujo de bits. El decodificador a continuación verifica (2524) el valor del valor temporal, el cual señala si el decodificador debe utilizar (2530) una transformación Hadamard para el grupo de canales. Si no, el decodificador fija (2526) un *iTmp* igual al bit siguiente del flujo de bits y verifica (2528) el valor de *iTmp*, el cual señala si el decodificador debe utilizar (2550) una transformación de identidad para el grupo de canales. Si no, el decodificador descodifica (2570) una transformación unitaria genérica para el grupo de canales.

Si *#ChannelsInGroup* es mayor de 2, el grupo de canales es para señales audio de sonido envolvente, y el decodificador fija (2540) un valor temporal *iTmp* igual al bit siguiente del flujo de bits. El decodificador verifica (2542) el valor del valor temporal, el cual señala si el decodificador debe utilizar (2550) una transformación de identidad del tamaño *#ChannelsInGroup* para el grupo de canales. Si no, el decodificador fija (2560) un *iTmp* igual al bit siguiente del flujo de bits, y verifica (2562) el valor del *iTmp*. El bit señala si el decodificador debe descodificar (2570) una transformación unitaria genérica para el grupo de canales o utilizar (2580) una transformación DCT tipo II del tamaño del *#ChannelsInGroup* para el grupo de canales.

Cuando el decodificador utiliza una matriz Hadamard, una DCT tipo II, o una transformación genérica unitaria para el grupo de canales, el decodificador descodifica (2590) la información de activación/desactivación de bandas de transformación multicanal para la matriz, y a continuación sale.

#### 5 F. Representación de Rotación Givens de Matrices de Transformación

En algunas formas de realización, el codificador y el decodificador utilizan parámetros cuantificados de factorización basados en la rotación Givens para especificar una matriz de transformación unitaria arbitraria para la eficacia de los bits.

En general, una matriz de transformación unitaria puede ser representada utilizando rotaciones de factorización Givens. Utilizando esta especificación, una matriz de factorización unitaria puede representarse como:

$$A_{\text{unitaria}} = \Theta_{0, N-2} \dots \Theta_{0,1} \Theta_{0,0} \Theta_{1, N-3} \dots \Theta_{1,1} \Theta_{1,0} \dots \Theta_{N-2,0} \begin{pmatrix} \alpha_0 & 0 & \dots & 0 \\ 0 & \alpha_1 & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & \alpha_{N-1} \end{pmatrix} \quad (13),$$

donde  $\alpha_i$  es +1 o -1 (signo de rotación), y cada  $\Theta$  es la forma de la matriz de rotación (2600) mostrada en la Figura 26. La matriz de rotación (2600) es casi como una matriz de identidad, pero tiene cuatro términos de seno/coseno con posiciones variables. Las Figuras 27a-27c muestran matrices de rotación ejemplares para rotaciones Givens para representar una matriz de transformación multicanal. Los dos términos cosenoidales están siempre sobre la diagonal, los dos términos sinusoidales están en la misma fila/columna que los dos términos cosenoidales. Cada  $\Theta$  tiene un ángulo de rotación, y su valor puede tener un rango de  $-\frac{\pi}{2} \leq \omega_k < \frac{\pi}{2}$ . El número de dichas matrices de rotación  $\Theta$  necesario para describir completamente una matriz unitaria  $N \times N$   $A_{\text{unitaria}}$  es:

$$\frac{N(N-1)}{2} \quad (14).$$

Para más información acerca de las rotaciones de factorización Givens, véase Vaidyanathan, *Sistemas Multivelocidad y Bancos de Filtros* ["Multirate Systems and Filter Banks"], Capítulo 14.6, "Factorización de Matrices Unitarias" ["Factorization of Unitary Matrices"], Prentice Hall (1993).

En algunas formas de realización, el codificador cuantifica los ángulos de rotación de la factorización Givens para reducir la velocidad de transmisión de bits. La Figura 28 muestra una técnica (2800) de representación de una matriz de transformación multicanal que utiliza rotaciones cuantificadas de factorización Givens.

Como una alternativa, un codificador o herramienta de procesamiento utiliza rotaciones cuantificadas de factorización Givens para representar una matriz unitaria para alguna finalidad distinta de la transformación multicanal de canales audio.

El codificador primeramente computa (2810) una matriz unitaria arbitraria para una transformación multicanal. El codificador a continuación computa (2820) las rotaciones de factorización Givens para la matriz unitaria.

Para reducir la transmisión de bits, el codificador cuantifica (2830) los ángulos de rotación. En una implementación, el codificador cuantifica de manera uniforme cada ángulo de rotación hasta uno de los 64 ( $2^6 = 64$ ) posibles valores. Los signos de rotación se indican con un bit cada uno, de forma que el codificador utiliza el número siguiente de bits para representar la matriz unitaria  $N \times N$ .

$$6 \cdot \frac{N(N-1)}{2} + N = 3N^2 - 2N \quad (15).$$

Este nivel de cuantificación posibilita que el codificador represente la matriz unitaria  $N \times N$  para la transformación multicanal con un muy buen grado de precisión. Como una alternativa, el codificador utiliza algún otro nivel y/o tipo de cuantificación.

5 La Figura 29 muestra una técnica (2900) para extraer información para una transformación unitaria genérica para un grupo de canales entre un flujo de datos de acuerdo con una sintaxis de grupo de datos concreta. La Figura 29 muestra la técnica (2900) llevada a cabo por el decodificador para analizar el flujo de bits; el codificador lleva a cabo una técnica correspondiente para formatear la información para la transformación unitaria genérica de acuerdo con la sintaxis del flujo de bits. Como una alternativa, el decodificador y el codificador utilizan otra síntesis, por ejemplo,  
10 una que utiliza una ordenación o resolución diferentes para los ángulos de rotación.

En primer lugar, el decodificador inicializa diversas variables utilizadas en el resto de la descodificación. Específicamente, el decodificador fija (2910) el número de ángulos que hay que descodificar *#AnglesToDecode* en base al número de canales del grupo de canales *#ChannalesInGroup* como se muestra en la Ecuación 14. El decodificador fija también (2912) el número de signos que hay que descodificar *#SignsToDecode* en base a *#ChannelsInGroup*.  
15 El decodificador vuelve a fijar también (2914, 2916) un contador de ángulos descodificados *iAnglesDecoded* y un contador de signos descodificados *iSignsDecoded*.

El decodificador verifica (2920) si hay cualquier ángulo que descodificar, y, si es así, fija (2922) el valor para el  
20 ángulo de rotación siguiente, reconstruyendo el ángulo de rotación a partir del valor cuantificado de 6 bits.

$$\text{RotationAngle}[\text{iAnglesDecoded}] = \pi * (\text{getBits}(6) - 32)/64 \quad (16).$$

25 El decodificador a continuación incrementa (2924) el contador de ángulos descodificados y verifica (2920) si hay cualquier ángulo adicional que descodificar.

Cuando no hay más ángulos que descodificar, el decodificador verifica (2940) si hay muchos signos que descodificar, y si es así, fija (2942) el valor del siguiente signo, reconstruyendo el signo a partir del valor de 1 bit.

$$\text{RotationSing}[\text{iSignsDecoded}] = (2 * \text{getBits}(1)) - 1 \quad (17).$$

30 El decodificador a continuación incrementa (2944) el contador de signos descodificados y verifica (2940) si hay cualquier signo adicional que descodificar. Cuando no hay más signos que descodificar, el decodificador sale.

## VI. Cuantificación y Ponderación

En algunas formas de realización, un codificador, como por ejemplo el codificador (600) de la Figura 6, lleva a cabo  
40 la cuantificación y ponderación sobre los datos audio utilizando las diversas técnicas descritas a continuación. Para una señal audio multicanal configurada en mosaicos, el codificador computa y aplica las matrices de cuantificación para canales de mosaicos, para los modificadores de paso de cuantificación por canal, y para los factores de mosaicos de cuantificación global. Ello posibilita que el codificador conforme el ruido de acuerdo con un modelo auditivo, equilibre el ruido entre canales, y controle la distorsión global.

45 Un decodificador correspondiente, como por ejemplo el decodificador (700) de la Figura 7, lleva a cabo una cuantificación inversa en una ponderación inversa. Para señales audio multicanal configuradas en mosaicos, el decodificador descodifica y aplica unos factores de mosaicos de cuantificación global, unos modificadores de paso de cuantificación por canal, y unas matrices de cuantificación para los canales de los mosaicos. La cuantificación inversa  
50 y la ponderación inversa se funden en un único paso.

### A. Factor Global de Cuantificación de Mosaicos

En algunas formas de realización, para controlar la calidad y/o la velocidad de transmisión de bits de los datos audio  
55 de un mosaico, un cuantificador de un codificador computa un tamaño de paso de cuantificación  $Q_i$  para el mosaico. El cuantificador puede trabajar en conjunción con un controlador de velocidad/calidad para evaluar los diferentes tamaños de paso de cuantificación para el mosaico antes de seleccionar un tamaño de paso de cuantificación de mosaicos que satisfaga los condicionamientos de la velocidad de transmisión de bits y/o de la calidad. Por ejemplo, el cuantificador y el controlador operan de acuerdo con lo descrito en la Solicitud de Patente estadounidense con el número de serie  
60 10/017,694, titulada (Estrategia de Control de la Calidad y la Velocidad para Señales Audio Digitales) ["Quality and Rate Control Strategy for Digital Audio"], depositada el 14 de Diciembre de 2001.

La Figura 30 muestra una técnica (3000) para extraer un factor de cuantificación de mosaicos global de un flujo de bits de acuerdo con una sintaxis de flujo de bits concreta. La Figura 30 muestra la técnica (3000) llevada a cabo  
65 por el decodificador para analizar el flujo de bits; el codificador lleva a cabo una técnica para formatear el factor de cuantificación de mosaicos de acuerdo con la sintaxis del flujo de bits. Como una alternativa, el decodificador y el codificador utilizan otra sintaxis, por ejemplo, una que trabaje con rangos diferentes para el factor de cuantificación de mosaicos, utilice una lógica diferente para modificar el factor de mosaicos o descodifique el factor de los mosaicos.

## ES 2 316 679 T3

En primer lugar, el descodificador inicializa (3010) el tamaño de paso de cuantificación  $Q_t$  para el mosaico. En una implementación, el descodificador fija  $Q_t$  en:

$$Q_t = 90 \cdot \text{ValidBitsPerSample}/16 \quad (18),$$

donde *ValidBitsPerSample* es un número  $16 \leq \text{ValidBitsPerSample} \leq 24$  que se fija para el descodificador o el clip de audio, o se fija en algún otro nivel.

A continuación el codificador (3020) obtiene seis bits indicativos de la primera modificación de  $Q_t$  con respecto al valor inicializado de  $Q_t$ , y almacena el valor  $-32 \leq Tmp \leq 31$  en la variable temporal *Tmp*. La función *SignExtend()* determina un valor con signo a partir de un valor sin signo. El descodificador suma (3030) el valor de *Tmp* al valor inicializado de  $Q_t$ , a continuación determina (3040) el signo de la variable *Tmp*, el cual es almacenado en la variable *SignofDelta*.

El descodificador verifica (3050) si el valor de *Tmp* igual a  $-32$  o  $31$ . Si no, el descodificador sale. Si el valor de *Tmp* es igual a  $-32$  o  $31$ , el codificador puede haber señalado que  $Q_t$  debe ser modificado en mayor medida. La dirección (positiva o negativa) de la(s) modificación(es) adicionales se indica mediante *SignofDelta*, y el descodificador obtiene (3060) los siguientes cinco bits para determinar la magnitud  $0 \leq Tmp \leq 31$  de la siguiente modificación. El descodificador cambia (3070) el valor actual de  $Q_t$  en la dirección de *SignofDelta* por el valor de *Tmp*, a continuación verifica (3080) si el valor de *Tmp* es  $31$ . Si no, el descodificador sale. Si el valor de *Tmp* es  $31$ , el descodificador obtiene (3060) los siguientes cinco bits y continúa desde ese punto.

En formas de realización que no utilizan configuraciones de mosaico, el codificador computa un tamaño de paso de cuantificación global para una trama u otra porción de datos audio.

### B. Modificadores de Paso de Cuantificación por Canal

En algunas formas de realización, un codificador computa un modificador de paso de cuantificación para cada canal de un mosaico:  $Q_{c,0}, Q_{c,1}, \dots, Q_{c,\#ChannelTile-1}$ . El codificador generalmente computa estos factores de cuantificación específicos de un canal para equilibrar la calidad de la reconstrucción para todos los canales. Incluso en formas de realización que no utilizan configuraciones de mosaicos, el codificador puede también computar los factores de cuantificación por canal para los canales de una trama u otra unidad de datos audio. Por el contrario, las técnicas de cuantificación anteriores, como por ejemplo las utilizadas en el codificador (100) de la Figura 1, utilizan un elemento de matriz de cuantificación por banda de una ventana de un canal, pero no tienen un modificador global para el canal.

La Figura 31 muestra una técnica generalizada (3100) para la computación de los modificadores de paso de cuantificación por canal para datos audio multicanal. El codificador utiliza varios criterios para computar los modificadores de paso de cuantificación. En primer lugar, el codificador busca aproximadamente la misma calidad a través de todos los canales de los datos audio reconstruidos. En segundo lugar, si las posiciones de los altavoces son conocidas, el codificador favorece los altavoces que son más importantes para la percepción en usos típicos para la configuración de los altavoces. En tercer lugar, si son conocidos los tipos de altavoces, el codificador favorece los mejores altavoces de la configuración de altavoces.

El codificador empieza mediante la regulación (3110) de los modificadores de paso de cuantificación para los canales. En una implementación, el codificador fija (3110) los modificadores en base a la energía de los respectivos canales. Por ejemplo, para un canal con relativamente más energía (esto es, más alto) que los otros canales, los modificadores de paso de cuantificación para los otros canales se ejecutan relativamente más altos. O, el codificador puede fijar (3110) los modificadores para igualar los valores inicialmente (en base a la evaluación de resultados “en bucle cerrado” para converger en los valores finales de los modificadores).

El codificador cuantifica (3120) los datos audio multicanal utilizando los modificadores audio de cuantificación así como otros factores de cuantificación (incluyendo la ponderación) si dichos otros factores no han sido ya aplicados.

Después de la subsecuente reconstrucción, el codificador evalúa (3130) la calidad de los canales de la señal reconstruida utilizando la *NER* o alguna otra medición de la calidad. El codificador verifica (3140) si la señal audio reconstruida satisface los criterios de calidad (y/u otros criterios) y, si es así, sale. Si no el codificador fija (3110) unos nuevos valores para los modificadores de paso de cuantificación, ajustando los modificadores a la vista de los resultados evaluados. Como una alternativa, para una regulación en bucle abierto de un paso de los modificadores de paso, el codificador omite la evaluación (3130) y la verificación (3140).

Los modificadores de paso de cuantificación por canal tienden a cambiar de ventana/mosaico a ventana/mosaico. El codificador codifica los modificadores de paso de modificación como códigos de longitud literales o variables, y a continuación los empaqueta en el flujo de bits con los datos audio.

La Figura 32 muestra una técnica (3200) para extraer modificadores de paso de cuantificación por canal a partir de un flujo de bits de acuerdo con una sintaxis de un flujo de bits concreta. La Figura 32 muestra la técnica (3200) llevada a cabo por el descodificador para analizar el flujo de bits; el codificador lleva a cabo una técnica correspondiente (in-

dicadores de regulación, datos de empaquetado para los modificadores de paso de cuantificación, etc.) para formatear los modificadores de paso de cuantificación de acuerdo con la sintaxis del flujo de bits. Como una alternativa, el descodificador y el codificador utilizan otras sintaxis, por ejemplo, una que trabaje con indicadores o lógicas diferentes para codificar los modificadores de paso de cuantificación.

La Figura 32 muestra la extracción de los modificadores de paso de cuantificación por canal para un mosaico. Como una alternativa, en formas de realización que no utilizan mosaicos, el descodificador extrae los modificadores de paso por canal para las tramas u otras unidades de datos audio.

Para empezar, el descodificador verifica (3210) si el número de canales del mosaico es mayor de 1. Si no, los datos audio son mono. El descodificador fija (3212) el modificador de paso de cuantificación para el canal mono en 0 y sale.

Para señales audio multicanal, el descodificador inicializa las diversas variables. El descodificador obtiene (3220) unos bits indicativos del número de bits por modificador de paso de cuantificación (*#BitsPerQ*) para el mosaico. En una implementación, el descodificador obtiene tres bits. El descodificador a continuación fija (3222) un contador de canales *iChannelsDone* en 0.

El descodificador verifica (3230) si el contador de canales es menor que el número de canales del mosaico. Si no, todos los modificadores de paso de cuantificación de los canales del mosaico han sido extraídos, y el descodificador sale.

Por otro lado, si el contador de canales es inferior al número de canales del mosaico, el descodificador obtiene (3232) un bit y verifica (3240) el bit para determinar si el modificador de paso de cuantificación del canal actual es 0. Si es así, el descodificador fija (3242) el modificador de paso de cuantificación para el canal actual en 0.

Si el modificador de paso de cuantificación para el canal actual no es 0, el descodificador verifica (3250) si *#bitsPerQ* es mayor de 0, para determinar si el modificador de paso de cuantificación para el canal actual es 1. Si es así, el descodificador fija (3252) el modificador de paso de cuantificación para el canal actual en 1.

Si *#BitsPerQ* es mayor de 0, el descodificador obtiene los siguientes *#BitsPerQ* del flujo de bits, añade 1 (dado que el valor de 0 desencadena una condición de salida anterior), y fija (3260) el modificador de paso de cuantificación para el canal actual respecto del resultado.

Después de que el descodificador fija el modificador de paso de cuantificación para el canal actual, el descodificador incrementa (3270) el contador de canales y verifica (3230) si el contador de canales es menor que el número de canales del mosaico.

### C. Codificación y Descodificación de Matrices de Cuantificación

En algunas formas de realización, un codificador computa una matriz de cuantificación para cada canal de un mosaico. El codificador mejora en relación con las técnicas de cuantificación anteriores, como por ejemplo las utilizadas en el codificador (100) de la Figura 1. Para una compresión con pérdidas de matrices de cuantificación, el codificador utiliza un tamaño de paso flexible para los elementos de la matriz de cuantificación, lo cual posibilita que el codificador cambie la resolución de los elementos de las matrices de cuantificación. Al margen de esta característica, el codificador aprovecha la correlación temporal de los valores de las matrices de cuantificación durante la compresión de las matrices de cuantificación.

De acuerdo con lo anteriormente expuesto, una matriz de cuantificación sirve como una formación de tamaños de paso, un valor de paso por banda de frecuencia bark (u otra banda de cuantificación tabica de otra forma) para cada canal de un mosaico. El codificador utiliza unas matrices de cuantificación para “colorear” la señal audio reconstruida para tener una forma espectral comparable con la de la señal original. El codificador generalmente determina las matrices de cuantificación en base a las psicoacústica y comprime las matrices de cuantificación para reducir la velocidad de transmisión de bits. La compresión de las matrices de cuantificación puede ser con pérdidas.

Las técnicas descritas en esta sección se describen con referencia a las matrices de cuantificación para canales de mosaicos. Para la anotación, supongamos que  $Q_{m,iChannel,iBand}$  representan el elemento de matriz de cuantificación para el canal *iChannel* para la banda *iBand*. En determinadas formas de realización que no utilizan configuraciones de mosaicos, el codificador puede utilizar también un tamaño de paso para los elementos de las matrices de cuantificación y/o aprovechar la correlación temporal de los valores de las matrices de cuantificación durante la compresión.

#### 1. Tamaño de Paso de Cuantificación Flexible para una Información de Máscara

La Figura 33 muestra una técnica generalizada (3300) para regular de forma adaptativa un tamaño de paso de cuantificación para los elementos de las matrices de cuantificación. Esto posibilita que el codificador cuantifique la información de máscara de forma burda o precisa. En una implementación, el codificador fija el tamaño de paso de cuantificación para los elementos de la matriz de cuantificación sobre una base de canal por canal para un mosaico (esto es, la base matriz por matriz cuando cada canal del mosaico tiene una matriz).

## ES 2 316 679 T3

Como una alternativa, el codificador fija el tamaño de paso de cuantificación para los elementos de la máscara sobre una base de mosaico por mosaico o de trama por trama, para una entera secuencia audio, o en algún otro nivel.

5 El codificador empieza por fijar (3310) un tamaño de paso de cuantificación para una (o más) máscara(s). (El número de máscaras afectadas dependen del nivel en el cual el codificador asigna el tamaño de paso de cuantificación flexible). En una implementación, el codificador evalúa la señal audio reconstruida durante algún periodo de tiempo y, dependiendo del resultado, selecciona el tamaño de paso de cuantificación para que sea 1, 2, 3, o 4 dB para la información de la máscara. La medición de la calidad evaluada por el codificador es la *NER* para una o más tramas previamente codificadas. Por ejemplo, si la calidad global es deficiente, el codificador puede fijar (3310) un valor más alto para el tamaño de paso de cuantificación para la identificación de la máscara, dado que la resolución de la matriz de cuantificación no es un uso eficaz de la velocidad de transmisión de bits. Por otro lado, si la calidad global es buena, el codificador puede fijar (3310) un valor más bajo para el tamaño de paso de cuantificación para la identificación de la máscara, dado que una mejor resolución de la matriz de cuantificación puede eficazmente mejorar la calidad percibida. Como una alternativa, el codificador utiliza otra medición de la calidad, la evaluación a lo largo de un periodo diferente, y/u otros criterios en una estimación en bucle abierto del tamaño de paso de cuantificación. El codificador puede también utilizar tamaños de paso de cuantificación diferentes o adicionales. O, el codificador puede omitir la estimación en bucle abierto, en lugar de basarse en resoluciones de resultados en bucle cerrado para converger en el valor final del tamaño de paso.

20 El codificador cuantifica (3320) las una o más matrices de cuantificación utilizando el tamaño de paso de cuantificación de los elementos de la máscara, y pondera y cuantifica los datos audio multicanal.

Después de la subsecuente reconstrucción, el codificador evalúa (3330) la calidad de la señal audio reconstruida utilizando una *NER* o alguna otra medición de la calidad. El codificador verifica (3340) si la calidad de la señal audio reconstruida justifica la regulación actual del tamaño de paso de cuantificación para la identificación de la máscara. Si no, el codificador puede fijar (3310) un valor más alto o más bajo de tamaño de paso de cuantificación para la información de máscara. En otro caso, el codificador sale. Como una alternativa, para una regulación de un bucle de un paso de tamaño de paso de la cuantificación para la información de la máscara, el codificador omite la evaluación (3330) y la verificación (3340).

30 Después de la selección, el codificador indica el tamaño de paso de cuantificación para la información de la máscara en el nivel apropiado del flujo de bits.

35 La Figura 34 muestra una técnica generalizada (3400) para extraer un tamaño de paso de cuantificación para los elementos de la matriz de cuantificación. El decodificador puede así cambiar el tamaño de paso de cuantificación para los elementos de paso sobre una base de canal por canal para un mosaico sobre una base de mosaico por mosaico o de trama por trama para una secuencia de audio o en algún otro nivel.

40 El decodificador empieza por obtener (3410) un tamaño de paso de cuantificación para una (o más) máscara (s). (El número de máscaras afectadas depende del nivel en el cual el codificador asignó el tamaño de paso de cuantificación flexible). En una implementación, el tamaño de paso de cuantificación es 1, 2, 3, o 4 dB para la información de la máscara. Como una alternativa, el codificador y el decodificador utilizan tamaños de paso de cuantificación diferentes o adicionales para la información de la máscara.

45 El decodificador a continuación efectúa una codificación inversa (3420) de las una o más matrices de cuantificación utilizando el tamaño de paso de cuantificación para la información de la máscara, y reconstruye los datos audio multicanal.

### 2. Predicción Temporal de Matrices de Cuantificación

50 La Figura 35 muestra una técnica generalizada (3500) de compresión de matrices de cuantificación utilizando una predicción temporal. Con la técnica (3500), el codificador aprovecha una correlación temporal de los valores de máscara. Ello reduce la velocidad de transmisión de bits asociada con las matrices de cuantificación.

55 Las Figuras 35 y 36 muestran la predicción temporal para matrices de cuantificación en un canal de una trama de datos audio. Como una alternativa, un codificador comprime las matrices de cuantificación utilizando una predicción temporal entre múltiples tramas a través de alguna otra secuencia de audio, o para una configuración diferente de matrices de cuantificación.

60 Con referencia a la Figura 35, el codificador obtiene (3510) unas matrices de cuantificación para una trama. Las matrices de cuantificación para una trama. Las matrices de cuantificación de un canal tienden a ser las mismas de ventana a ventana, convirtiéndolas en buenos candidatos para una codificación predictiva.

65 El codificador a continuación codifica (3520) las matrices de cuantificación utilizando una predicción temporal. Por ejemplo, el codificador utiliza la técnica (3600) mostrada en la Figura 36. Como una alternativa, el codificador utiliza otra técnica con predicción temporal.



## ES 2 316 679 T3

El codificador determina (3530) si hay más matrices que comprimir y, en caso contrario, sale. En otro caso, el codificador obtiene las matrices de cuantificación siguientes. Por ejemplo, el codificador verifica si las matrices de la trama siguiente están disponibles para la codificación.

5 La Figura 36 muestra una técnica más detallada (3600) de compresión de matrices de cuantificación en un canal que utiliza una predicción temporal en una implementación.

La predicción temporal utiliza un proceso de nuevo muestreo a través de mosaicos de diferentes tamaños de ventana y utiliza una codificación de *run-level* sobre los residuales de predicción para reducir la velocidad de transmisión de bits.

El codificador inicia (3610) la compresión para la siguiente matriz de cuantificación que va a ser comprimida y verifica (3620) si está disponible una matriz de anclaje, lo cual generalmente depende de si la matriz es la primera de su canal. Si una matriz de anclaje no está disponible, el codificador directamente comprime (3630) la matriz de cuantificación. Por ejemplo, el codificador codifica de forma diferencial los elementos de la matriz de cuantificación (donde la diferencia de un elemento es relativa respecto del elemento de la banda anterior) y asigna los códigos de Huffman a los diferenciales. Para el primer elemento de la matriz (esto es, el elemento de la máscara para la banda 0), el codificador utiliza una constante de predicción que depende del tamaño de paso de cuantificación para los elementos de la máscara.

20

$$\text{PredConst} = 45/\text{MaskQuantMultiplier}_{i\text{Channel}} \quad (19).$$

Como una alternativa, el codificador utiliza otra técnica de compresión para la matriz de anclaje.

25

El codificador a continuación fija (3640) la matriz de cuantificación como matriz de anclaje como canal de la trama. Cuando el codificador utiliza mosaicos, el mosaico que incluye la matriz de anclaje para un canal puede llamarse el mosaico de anclaje. El codificador anota el tamaño de la matriz de anclaje o el tamaño del mosaico para el mosaico de anclaje, que puede ser utilizado para formar predicciones para las matrices con un tamaño diferente.

30

Por otro lado, si está disponible una matriz de anclaje, el codificador comprime la matriz de codificación utilizando una predicción temporal. El codificador computa (3650) una predicción para la matriz de cuantificación en base a la matriz de anclaje para el canal. Si la matriz de cuantificación que está siendo comprimida tiene el mismo número de bandas que la matriz de anclaje, la predicción es los elementos de la matriz de anclaje. Si la matriz de cuantificación que está siendo comprimida tiene un número diferente de bandas que la matriz de anclaje, sin embargo, el codificador vuelve a muestrear la matriz de anclaje para computar la predicción.

35

El proceso de nuevo muestreo utiliza el tamaño de la matriz que está siendo comprimida/el tamaño del mosaico actual y el tamaño de la matriz de anclaje/tamaño del mosaico de anclaje.

40

$$\text{MaskPrediction}[i\text{Band}] = \text{AnchorMask}[i\text{ScaleBand}] \quad (20),$$

donde *iScaleBand* es la banda de la matriz de anclaje que incluye la frecuencia representativa (por ejemplo, la media) de *iBand*. *iBand* es en términos de la matriz de cuantificación actual/tamaño de mosaico actual mientras que *iScaleBand* es en términos de la matriz de anclaje/tamaño del mosaico de anclaje.

45

La Figura 37 ilustra una técnica de nuevo muestreo de la matriz de anclaje cuando el codificador utiliza mosaicos. La Figura 37 muestra un cartografiado ejemplar (3700) de bandas de un mosaico actual respecto de bandas de un mosaico de anclaje para formar una predicción. Las frecuencias en la mitad de los límites (3720) de las bandas de la matriz de cuantificación del mosaico actual son cartografiadas (3730) respecto de las frecuencias de la matriz de anclaje del mosaico de anclaje. Los valores para la predicción de la máscara son fijados dependiendo de cuándo las frecuencias cartografiadas son relativas con respecto a los límites (3710) de las bandas de la matriz de anclaje del mosaico de anclaje. Como una alternativa, el codificador utiliza una predicción actual con respecto a la matriz de cuantificación precedente del canal o alguna otra matriz precedente, o utiliza otra técnica de nuevo muestreo.

55

Volviendo a la Figura 36, el codificador computa (3660) un residual de la matriz de cuantificación con respecto a la predicción. En teoría, la predicción es perfecta y el residual no tiene energía. Si es necesario, sin embargo, el codificador codifica (3670) el residual. Por ejemplo, el codificador utiliza una codificación *run-level* u otra técnica de compresión para el residual de la predicción.

60

El codificador a continuación determina (3680) si hay más matrices que deben ser comprimidas y, si no, sale. En otro caso, el codificador obtiene (3610) la siguiente matriz de cuantificación y continúa.

La Figura 38 muestra una técnica (3800) para extraer y descodificar matrices de cuantificación comprimidas utilizando una predicción temporal de acuerdo con una sintaxis de flujo de bits concreta. Las matrices de cuantificación son para los canales de un único mosaico de una trama. La Figura 38 muestra una técnica (3800) llevada a cabo por el descodificador para analizar información dentro del flujo de bits; el codificador lleva a cabo una técnica correspon-

65

diente. Como una alternativa, el decodificador y el codificador utilizan otra sintaxis para una o más de las opciones mostradas en la Figura 38, por ejemplo, una que utilice o una ordenación diferente, o una que no utilice mosaicos.

El decodificador verifica (3810) si el codificador ha llegado al principio de una trama. Si es así, el decodificador marca (3812) todas las matrices de anclaje para la trama como que no están siendo fijadas.

El decodificador a continuación verifica (3820) si la matriz de anclaje está disponible en el canal de la siguiente matriz de cuantificación que va a ser modificada. Si no hay ninguna matriz disponible, el decodificador obtiene (3830) el tamaño de paso de cuantificación de la matriz de cuantificación del canal. En una implementación, el decodificador obtiene el valor 1, 2, 3, o 4 dB.

$$\text{MaskQuantMultiplier}_{i\text{Channel}} = \text{obtenerBits} (2) + 1 \quad (21).$$

El decodificador a continuación descodifica (3832) la matriz de anclaje del canal. Por ejemplo, el decodificador Huffman descodifica los elementos codificados de manera diferencial de la matriz de anclaje (donde la diferencia para un elemento es relativa respecto del elemento de la banda anterior) y reconstruye los elementos. Para el primer elemento, el decodificador utiliza la constante de predicción utilizada en el codificador.

$$\text{PredConst} = 45/\text{MaskQuantMultiplier}_{i\text{Channel}} \quad (22).$$

Como una alternativa, el decodificador utiliza otra técnica de descompresión para la matriz de anclaje de un canal de la trama.

El decodificador a continuación fija (3834) la matriz de cuantificación como matriz de anclaje del canal de la trama y fija los valores de la matriz de cuantificación del canal para los de la matriz de anclaje.

$$Q_{m, i\text{Channel}, i\text{Band}} = \text{AnchorMask} [i\text{Band}] \quad (23).$$

El decodificador anota también el tamaño de mosaico del mosaico de anclaje el cual puede ser utilizado para formar predicciones para las matrices de los mosaicos con un tamaño diferente del mosaico de anclaje.

Por otro lado, si una matriz de anclaje está disponible para el canal, el decodificador descomprime la matriz de cuantificación utilizando una predicción temporal. El decodificador computa (3840) una predicción para la matriz de cuantificación en base a la matriz de anclaje para el canal. Si la matriz de cuantificación para el mosaico actual tiene el mismo número de bandas que la matriz de anclaje, la predicción es los elementos de la matriz de anclaje. Si la matriz de cuantificación para el mosaico actual tiene un número de bandas diferentes que la matriz de anclaje, sin embargo, el codificador vuelve a muestrear la matriz de anclaje para obtener la predicción, por ejemplo, utilizando el tamaño de mosaico actual y el tamaño de mosaico de anclaje como se muestra en la Figura 37.

$$\text{MaskPrediction} [i\text{Band}] = \text{AnchorMask} [i\text{ScaledBand}] \quad (24).$$

Como una alternativa, el decodificador utiliza una predicción temporal con respecto a la matriz de cuantificación precedente del canal o alguna otra matriz precedente, o utiliza otra técnica de nuevo muestreo.

El decodificador obtiene (3842) el bit siguiente del flujo de bits y verifica (3850) si el flujo de bits incluye un residual para la matriz de cuantificación. Si no hay una actualización de máscara para este canal en el mosaico actual, el residual de predicción de máscara es 0, de forma que:

$$Q_{m, i\text{Channel}, i\text{Band}} = \text{MaskPrediction} [i\text{Band}] \quad (25).$$

Por otro lado, si hay un residual de predicción, el decodificador descodifica (3852) el residual, por ejemplo, utilizando una descodificación de nivel de ejecución o alguna otra técnica de descompresión. El decodificador a continuación suma (3854) el residual de predicción a la predicción para reconstruir la matriz de cuantificación. Por ejemplo, la adición es una adición escalar simple sobre una base de banda por banda para obtener el elemento para la banda  $i\text{Band}$  para el canal actual  $i\text{Channel}$ :

$$Q_{m, i\text{Channel}, i\text{Band}} = \text{MaskPrediction} [i\text{Band}] + \text{MaskPredResidual} [i\text{Band}] \quad (26).$$

El decodificador a continuación verifica (3860) si las matrices de cuantificación para todos los canales del mosaico actual han sido descodificadas y, si es así, sale. En otro caso, el decodificador continúa descodificando para la siguiente matriz de cuantificación del mosaico actual.

## D. Cuantificación Inversa y Ponderación Inversa Combinadas

Una vez que el descodificador extrae toda la información de cuantificación y ponderación necesaria, el descodificador efectúa una cuantificación inversa y una ponderación inversa de los datos audio. En una implementación, el descodificador lleva a cabo la cuantificación inversa y la ponderación inversa en un paso, lo cual se muestra en las ecuaciones de abajo por razones de claridad de impresión.

$$\text{CombinedQ} = Q_1 + Q_{c, iChannel} - (\text{Max}(Q_{m, iChannel}, \bullet) - Q_{m, iChannel, lband}) \cdot \text{MaskQuantMultiplier}_{iChannel} \quad (27a),$$

$$y_{iqw}[n] = 10^{\text{CombinedQ}/20} \cdot x_{iqw}[n] \quad (27b).$$

donde  $x_{iqw}$  es la entrada (por ejemplo un coeficiente inverso MC transformado) del canal  $iChannel$ , y  $n$  es un índice de coeficiente de la banda  $iBand$ .  $\text{Max}(Q_{m, iChannel}, \bullet)$  es el valor de máscara máximo del canal  $iChannel$  por todas las bandas. (La diferencia entre los factores de ponderación mayores y más pequeños para una máscara es típicamente mucho menor que el rango de los valores potenciales para los elementos de la máscara, de forma que la cantidad del ajuste de cuantificación por factor de ponderación se computa con respecto al máximo).  $\text{MaskQuantMultiplier}_{iChannel}$  es el multiplicador de paso de cuantificación de la máscara para la matriz de cuantificación del canal  $iChannel$  e  $y_{iqw}$  es la salida de este paso.

Como una alternativa, el descodificador lleva a cabo la cuantificación y la ponderación inversas separadamente o utilizando técnicas diferentes.

## VII. Postprocesamiento Multicanal

En algunas formas de realización, un descodificador, como por ejemplo el descodificador (700) de la Figura 7, lleva a cabo un postprocesamiento multicanal sobre las muestras audio reconstruidas en el dominio temporal.

El postprocesamiento multicanal puede ser utilizado para muchas finalidades distintas. Por ejemplo, el número de canales descodificados puede ser inferior al número de canales de salida (por ejemplo, porque el codificador eliminó uno o más canales de entrada o canales transformados multicanal para reducir la complejidad de la codificación o la plenitud de la memoria intermedia). En este caso, puede utilizarse una transformación de postprocesamiento multicanal para crear uno o más canales fantasma en base a los datos reales de los canales descodificados. O, incluso si el número de canales descodificados es igual al número de canales de salida, la transformación de postprocesamiento puede utilizarse para una rotación espacial arbitraria de la presentación, mediante el recartografiado de los canales de salida entre las posiciones de los altavoces, u otros efectos espaciales o especiales. O, si el número de canales descodificados es mayor que el número de canales de salida (por ejemplo, reproduciendo una señal de audio de sonido envolvente en un equipo estéreo), la transformación de postprocesamiento puede utilizarse para “reconvertir” canales. En algunas formas de realización, los coeficientes de reconversión varían potencialmente con el tiempo -el postprocesamiento multicanal es controlado por el flujo de bits. Las matrices de transformación para estos escenarios y aplicaciones pueden ser suministradas o señaladas por el codificador.

La Figura 39 muestra una técnica generalizada (3900) de un postprocesamiento multicanal. El descodificador descodifica (3910) los datos audio multicanal codificados (3905) utilizando las técnicas mostradas en la Figura 7 u otras técnicas de descompresión, produciendo unos datos audio multicanal de dominio temporal reconstruidos (3915).

El descodificador a continuación lleva a cabo (3920) un postprocesamiento multicanal sobre los datos audio multicanal de dominio temporal (3915). Por ejemplo, cuando el codificador produce unos canales descodificados  $M$  y el descodificador emite de salida unos canales  $N$ , el postprocesamiento implica una transformación general de  $M$  a  $N$ . El descodificador toma unas muestras cosituadas (en el tiempo)  $M$ , una de cada uno de los canales codificados  $M$  reconstruidos, a continuación rellena cualquier canal que falte (esto es, los canales suprimidos  $N - M$  por el codificador) con ceros. El descodificador multiplica las muestras  $N$  con una matriz  $A_{\text{post}}$ .

$$y_{\text{post}} = A_{\text{post}} \cdot x_{\text{post}} \quad (28),$$

donde  $x_{\text{post}}$  e  $y_{\text{post}}$  son la entrada de los canales  $N$  y la salida procedente del postprocesamiento multicanal,  $A_{\text{post}}$  es una matriz de transformación  $N \times N$  general y  $x_{\text{post}}$  es relleno con ceros para corresponderse con la longitud del vector de salida  $N$ .

La matriz  $A_{\text{post}}$  puede ser una matriz con elementos predeterminados, o puede ser una matriz general con elementos especificados por el codificador. El codificador señala al descodificador el uso de una matriz predeterminada (por ejemplo, con uno o más bits indicadores) y envía los elementos de una matriz general al descodificador, o el descodificador puede ser configurado para utilizar siempre la misma matriz  $A_{\text{post}}$ . La matriz  $A_{\text{post}}$  no necesita poseer especiales características como el ser simétrica o invertible. Para obtener una flexibilidad adicional, el postprocesamiento multicanal puede ser activado/desactivado sobre una base de trama por trama u otra base (en cuyo caso, el descodificador puede utilizar una matriz de identidad para dejar inalterados los canales).

La Figura 40 muestra una matriz ejemplar  $A_{p\text{-center}}$  (4000) utilizada para crear un canal central fantasma de los canales izquierdo y derecho en un entorno de reproducción del canal 5.1 con los canales ordenados como se muestra en la Figura 4. La matriz ejemplar  $A_{p\text{-center}}$  (4000) pasa a los otros canales inalterada. El decodificador obtiene muestras cosituadas en el tiempo desde los canales izquierdo, derecho, de *subwoofer*, trasero izquierdo, y trasero derecho, y rellena el canal central con 0s. El codificador a continuación multiplica las seis muestras de entrada por la matriz  $A_{p\text{-center}}$  (4000).

$$\begin{pmatrix} a \\ b \\ a+b \\ \hline 2 \\ d \\ e \\ f \end{pmatrix} = A_{P\text{-Center}} \begin{pmatrix} a \\ b \\ 0 \\ d \\ e \\ f \end{pmatrix} \quad (29).$$

Como una alternativa, el decodificador utiliza una matriz con coeficientes diferentes o un número diferente de canales. Por ejemplo, el decodificador utiliza una matriz para crear unos canales fantasma en un canal 7.1 o un canal 9.1, u otro entorno de reproducción a partir de canales codificados para una señal audio multicanal 5.1.

La Figura 41 muestra una técnica (4100) para un postprocesamiento multicanal en el cual la matriz de transformación potencialmente cambia sobre una base de trama por trama. El cambio de la matriz de transformación puede conducir a un ruido audible (por ejemplo, pum) en la salida final si no se maneja cuidadosamente. Para evitar introducir el ruido de pum, el decodificador gradualmente efectúa una transición de una matriz de transformación a otra entre tramas.

El decodificador primeramente descodifica (4110) los datos audio multicanal codificados para una trama, utilizando las técnicas mostradas en la Figura 7 u otras técnicas de descompresión y produce datos audio multicanal de dominio temporal reconstruidos. El decodificador a continuación obtiene (4120) la matriz de postprocesamiento para la trama, por ejemplo, como se muestra en la Figura 42.

El decodificador determina (4130) si la matriz de la trama actual es diferente de la matriz de la trama anterior (si es que había una trama anterior). Si la matriz actual es la misma o no hay una matriz anterior, el decodificador aplica (4140) la matriz a las muestras audio reconstruidas para la trama actual. En otro caso, el decodificador aplica (4150) una matriz de transformación mezclada a las muestras audio reconstruidas para la trama actual. La función de mezcla depende de la implementación. En una implementación, en la muestra  $i$  de la trama actual el decodificador utiliza una matriz mezclada a corto plazo  $A_{\text{post}, i}$ .

$$A_{\text{post}, i} = \frac{\text{NumSamples} - i}{\text{NumSamples}} A_{\text{post}, \text{prev}} + \frac{i}{\text{NumSamples}} A_{\text{post}, \text{actual}} \quad (30),$$

donde  $A_{\text{post}, \text{prev}}$  y  $A_{\text{post}, \text{actual}}$  son las matrices postprocesamiento para las tramas previa y actual, respectivamente, y  $\text{NumSamples}$  es el número de muestras de la trama actual. Como una alternativa, el decodificador utiliza otra función de mezcla para suavizar las discontinuidades de las matrices de transformación postprocesamiento.

El decodificador repite la técnica (4100) sobre una base de trama por trama. Como una alternativa, el decodificador cambia el postprocesamiento multicanal sobre otra base.

La Figura 42 muestra una técnica (4200) para la identificación y extracción de una matriz de transformación para un postprocesamiento multicanal de acuerdo con una sintaxis de un flujo de bits concreto. La sintaxis posibilita la especificación de las matrices de transformación predefinidas así como de las matrices personalizadas para el postprocesamiento multicanal. La Figura 42 muestra la técnica (4200) llevada a cabo por el decodificador para analizar el flujo de bits; el codificador lleva a cabo una técnica correspondiente (establecimiento de indicadores, empaquetado de datos para elementos, etc.) para formatear la matriz de transformación de acuerdo con la sintaxis del flujo de bits. Como una alternativa, el decodificador y el codificador utilizan otras sintaxis para una o más de las opciones mostradas en la Figura 42, por ejemplo, una que utilice diferentes indicadores o una ordenación diferente.

## ES 2 316 679 T3

En primer lugar, el descodificador determina (4210) si el número de canales *#Channels* es mayor de 1. Si *#Channels* es 1, los datos audio son mono, y el descodificador utiliza (4212) una matriz de identidad (esto es, no lleva a cabo ningún postprocesamiento multicanal propiamente dicho).

- 5 Por otro lado, si *#Channels* es  $> 1$ , el descodificador fija (4220) un valor temporal *iTmp* igual al bit siguiente del flujo de bits. El descodificador a continuación verifica (4230) el valor del valor temporal, el cual señala si el descodificador debe o no utilizar (4232) una matriz de identidad.

- 10 Si el descodificador utiliza algo distinto de una matriz de identidad para la señal audio multicanal, el descodificador fija (4240) el valor temporal *iTmp* igual al bit siguiente del flujo de bits. El descodificador a continuación verifica (4250) el valor del valor temporal, el cual señala si el descodificador debe o no utilizar (4252) una matriz de transformación multicanal predefinida. Si el descodificador utiliza (4252) una matriz predefinida, el descodificador puede obtener uno o más bits adicionales a partir del flujo de bits (no mostrado) que indiquen cuál de las diversas matrices predefinidas disponibles del descodificador debe utilizarse.

- 15 Si el descodificador no utiliza una matriz predefinida, el descodificador inicializa diversos valores temporales para la descodificación de una matriz personalizada. El descodificador fija (4260) un contador *iCoefsDone* para coeficientes llevados a cabo hasta 0 y fija (4262) el número de coeficientes *#CoefsToDo* para descodificar hasta igualar el número de elementos de la matriz ( $\#Channels^2$ ). Para matrices conocidas que tengan propiedades particulares (por ejemplo, simétricas), el número de coeficientes que descodificar puede reducirse. El descodificador a continuación determina (4270) si todos los coeficientes han sido extraídos del flujo de datos y, si es así, finaliza. En otro caso, el descodificador obtiene (4272) el valor del siguiente elemento *A[iCoefsDone]* de la matriz e incrementa (4274) *iCoefsDone*. La forma en que los elementos son codificados y empaquetados dentro del flujo de bits depende de la implementación. En la Figura 42, la sintaxis permite cuatro bits de precisión por elemento de la matriz de transformación, y el valor absoluto para cada elemento es menor de o igual a 1. En otras implementaciones, la precisión por elemento es diferente, el codificador y el descodificador utilizan la compresión para explotar pautas de redundancia de la matriz de transformación, y/o la sintaxis difiere de alguna otra forma.

- 20 Después de describir e ilustrar los principios de la invención con referencia a las formas de realización descritas, debe advertirse que las formas de realización descritas pueden ser modificadas en cuanto a disposición y detalle sin apartarse de dichos principios. Debe entenderse que los programas, procesos, o procedimientos descritos en la presente memoria no están relacionados o limitados con cualquier tipo concreto de entorno informático, a menos que se indique lo contrario. Pueden utilizarse diversos tipos de entornos informáticos de propósito general o especializado o llevar a cabo operaciones de acuerdo con las enseñanzas descritas en la presente memoria. Los elementos de las formas de  
35 realización descritas mostrados en software pueden ser implementados en hardware y viceversa.

40

45

50

55

60

65

# REIVINDICACIONES

1. Un procedimiento de codificación audio implementado por computadora que comprende:

la recepción de los datos audio en una pluralidad de canales;

la tabicación de cada canal de la pluralidad de canales en ventanas de tamaño variable, en el que la configuración de ventana de cada canal de los diversos canales es independiente de los otros canales de la pluralidad de canales;

la agrupación de las ventanas en una pluralidad de mosaicos, en el que para cada una de la pluralidad de mosaicos las ventanas agrupadas en el mosaico tienen unas posiciones de partida idénticas y unas posiciones finales idénticas; y

la cuantificación de los datos audio, que incluye para un mosaico de la pluralidad de mosaicos la aplicación de un factor de cuantificación específica de un canal para cada canal de la pluralidad de canales para las ventanas agrupadas en el mosaico y la aplicación de un factor de cuantificación general de mosaico para el mosaico.

2. El procedimiento de la reivindicación 1 en el que la pluralidad de canales se compone de dos canales.

3. El procedimiento de la reivindicación 1 en el que la pluralidad de canales se compone de más de dos canales.

4. El procedimiento de la reivindicación 1 en el que los factores de cuantificación específicos de un canal son modificadores de paso de cuantificación específicos de un canal.

5. El procedimiento de la reivindicación 4 en el que la aplicación de los modificadores equilibra la calidad de la reconstrucción perceptual a través de la pluralidad de canales.

6. El procedimiento de la reivindicación 1 que comprende también, en el codificador, la computación de los factores de cuantificación en base al menos en parte a uno o más criterios.

7. El procedimiento de la reivindicación 6 en el que los criterios incluyen la igualdad de la calidad de la reconstrucción a través de la pluralidad de canales.

8. El procedimiento de la reivindicación 6 en el que los criterios incluyen el favorecimiento de uno o más de la pluralidad de canales que son más importantes que los otros canales en términos perceptuales.

9. El procedimiento de la reivindicación 6 en el que la computación se basa al menos en parte en las energías respectivas de la pluralidad de canales.

10. El procedimiento de la reivindicación 1 que comprende también, en el codificador, la computación de los factores de cuantificación mediante una estimación en bucle abierto.

11. El procedimiento de la reivindicación 1 que comprende también, en el codificador, la computación de los factores de cuantificación mediante una evaluación en bucle cerrado.

12. El procedimiento de la reivindicación 1 en el que el factor global de cuantificación de mosaico es un tamaño de paso de cuantificación de mosaico.

13. Un medio legible por computadora que almacena unas instrucciones ejecutables por computadora para provocar que una computadora así programada lleve a cabo el procedimiento de la reivindicación 1.

14. Un procedimiento de decodificación audio implementado por computadora que comprende:

la recepción de datos audio codificados en una pluralidad de canales;

la extracción de información para uno o más tamaños globales de paso de cuantificación de mosaico, y una pluralidad de modificadores de paso de cuantificadores específicos de un canal para uno o más mosaicos, agrupando cada uno de los uno o más mosaicos una pluralidad de ventanas que:

están en canales diferentes de la pluralidad de canales; y

tienen unas posiciones de partida idénticas y unas posiciones finales idénticas; y

la decodificación de los datos audio, que incluye para un mosaico de los uno o más mosaicos la aplicación de los uno o más tamaños globales de paso de cuantificación de mosaico para el mosaico y la aplicación de uno de los modificadores de paso de cuantificador específicos de un canal para cada canal de la pluralidad de canales para las ventanas agrupadas en el mosaico en una cuantificación inversa.

## ES 2 316 679 T3

15. El procedimiento de la reivindicación 14 en el que la pluralidad de canales está compuesta por dos canales.

16. El procedimiento de la reivindicación 14 en el que la pluralidad de canales está compuesta por más de dos canales.

17. El procedimiento de la reivindicación 14 en el que la extracción incluye la obtención de una pluralidad de bits que indica la precisión de la pluralidad de modificadores de paso de cuantificador específicos de un canal.

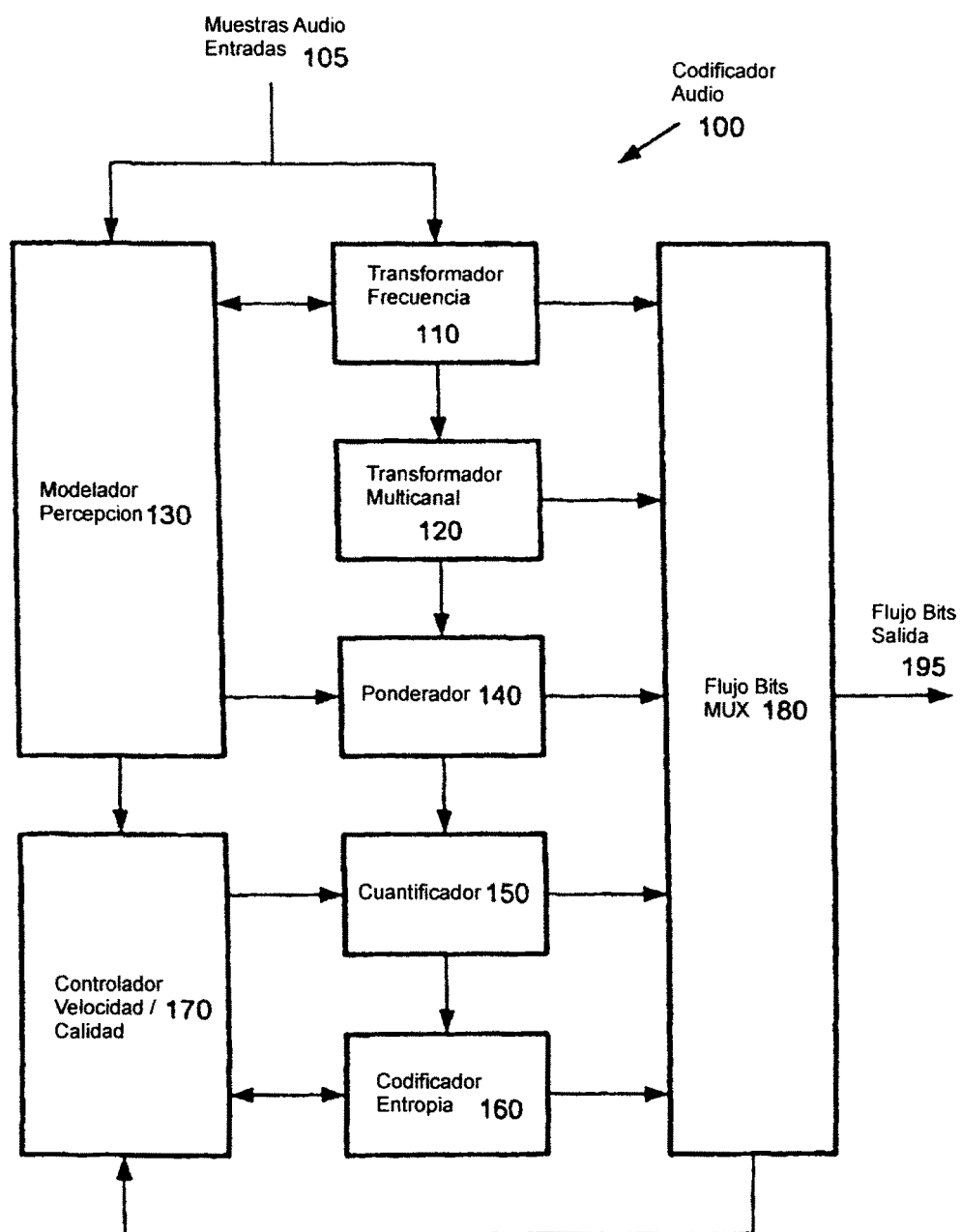
18. El procedimiento de la reivindicación 14 en el que la extracción incluye la obtención de un solo bit por modificador para indicar si el modificador tiene un valor de cero.

19. El procedimiento de la reivindicación 14 en el que la aplicación es parte de un paso combinado para la cuantificación, en el que el paso combinado para la cuantificación lleva a cabo una cuantificación inversa y una ponderación inversa, y en el que para cada uno de los diversos coeficientes de los datos audio el paso combinado incluye una única multiplicación por una cantidad total de cuantificación.

20. Un medio legible por computadora que almacena unas instrucciones ejecutables por computadora para provocar que una computadora así programada lleve a cabo el procedimiento de la reivindicación 14.

21. El procedimiento de la reivindicación 14 en el que la extracción incluye, para un mosaico de los uno o más mosaicos, la obtención de una pluralidad de bits que indica una modificación de un tamaño de paso de cuantificación de mosaico global inicial para el mosaico.

**Figura 1**  
**Tecnica Anterior**





**Figura 2**  
**Tecnica Anterior**

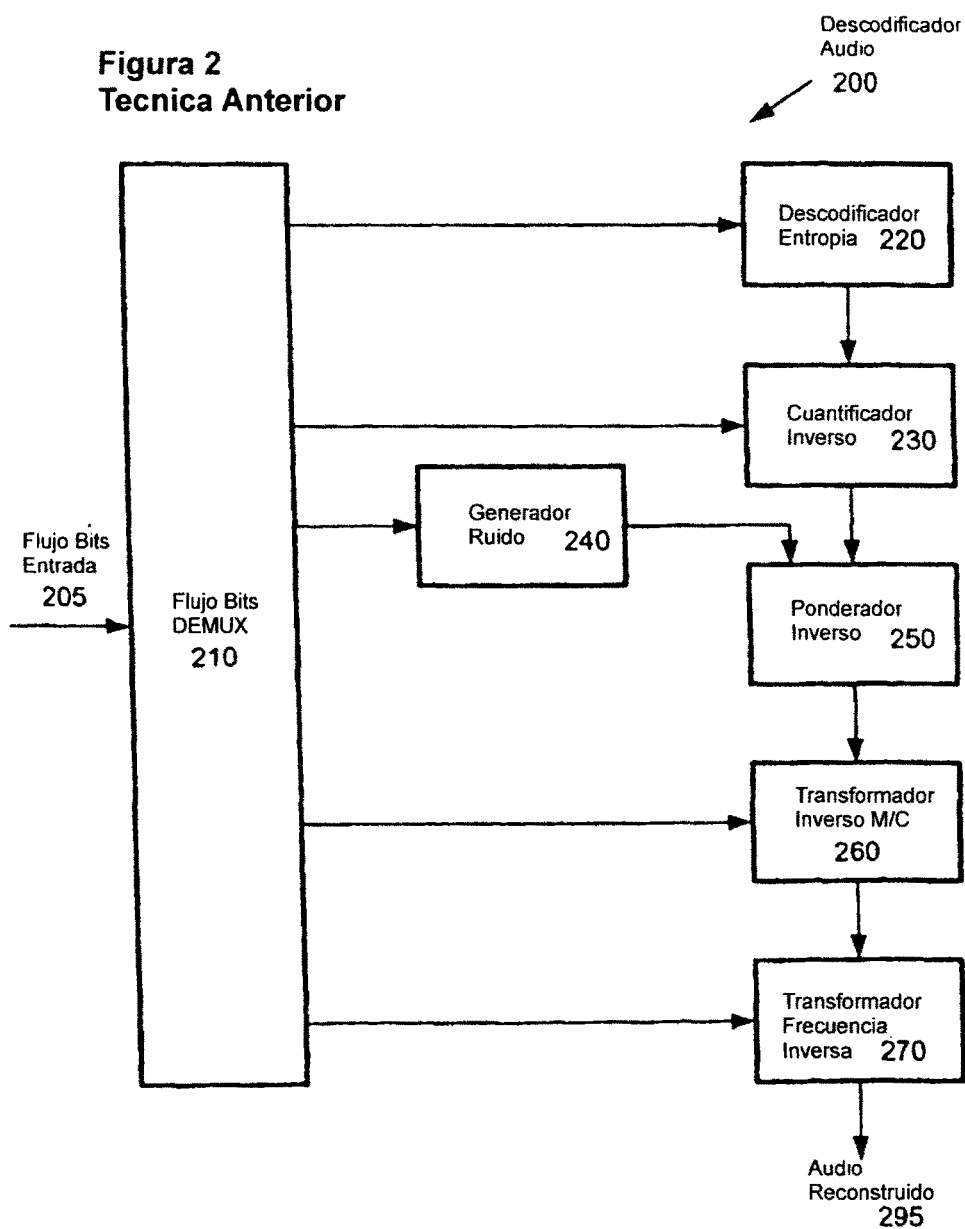


Figura 3a, Técnica Anterior

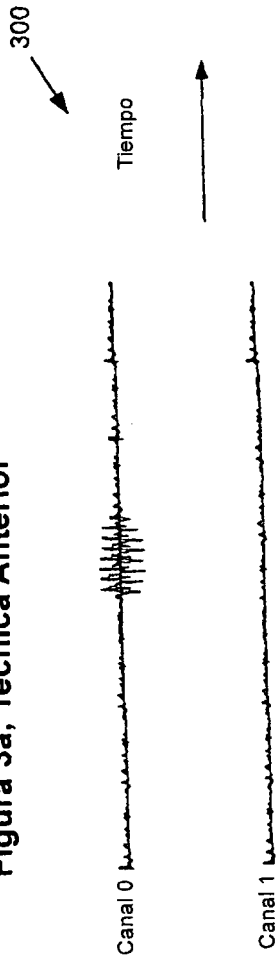
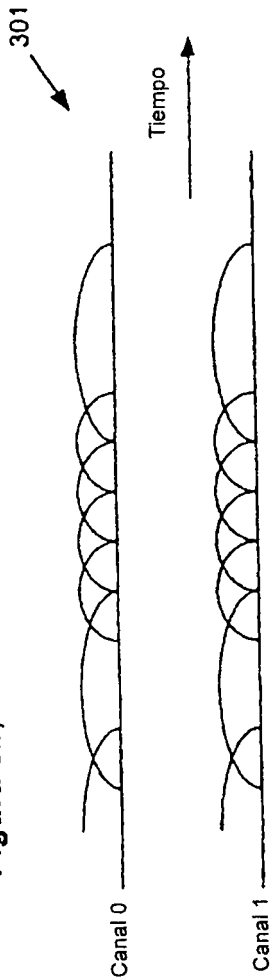
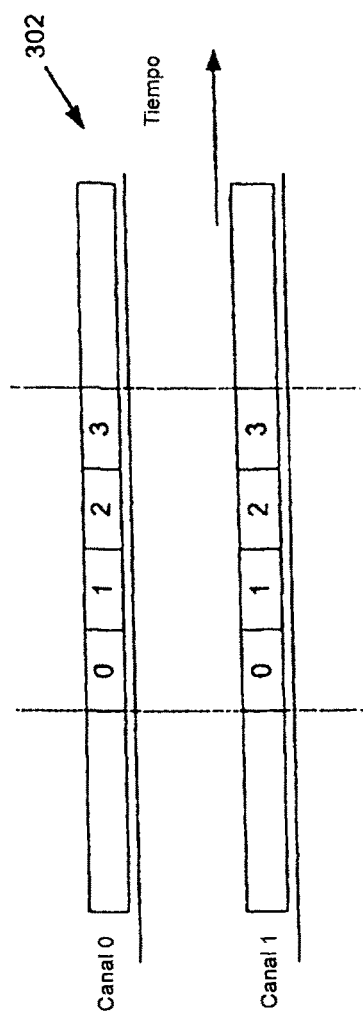


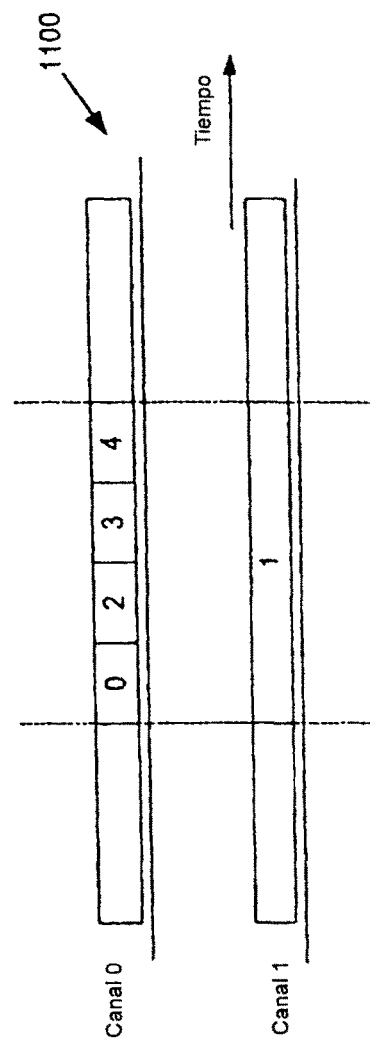
Figura 3b, Técnica Anterior



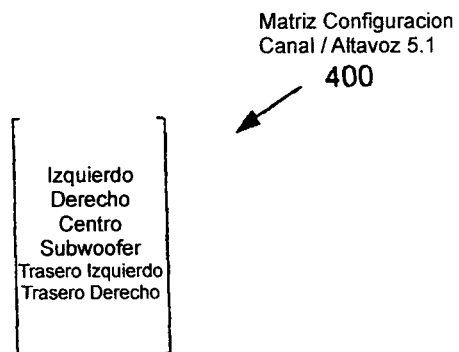
**Figura 3c, Técnica Anterior**



**Figura 11a**



**Figura 4**



**Figura 5**

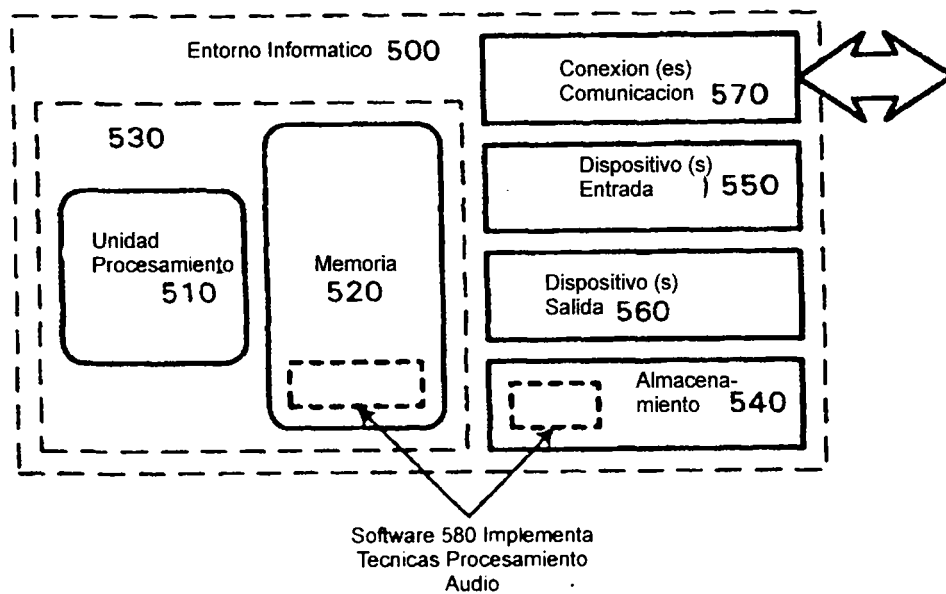
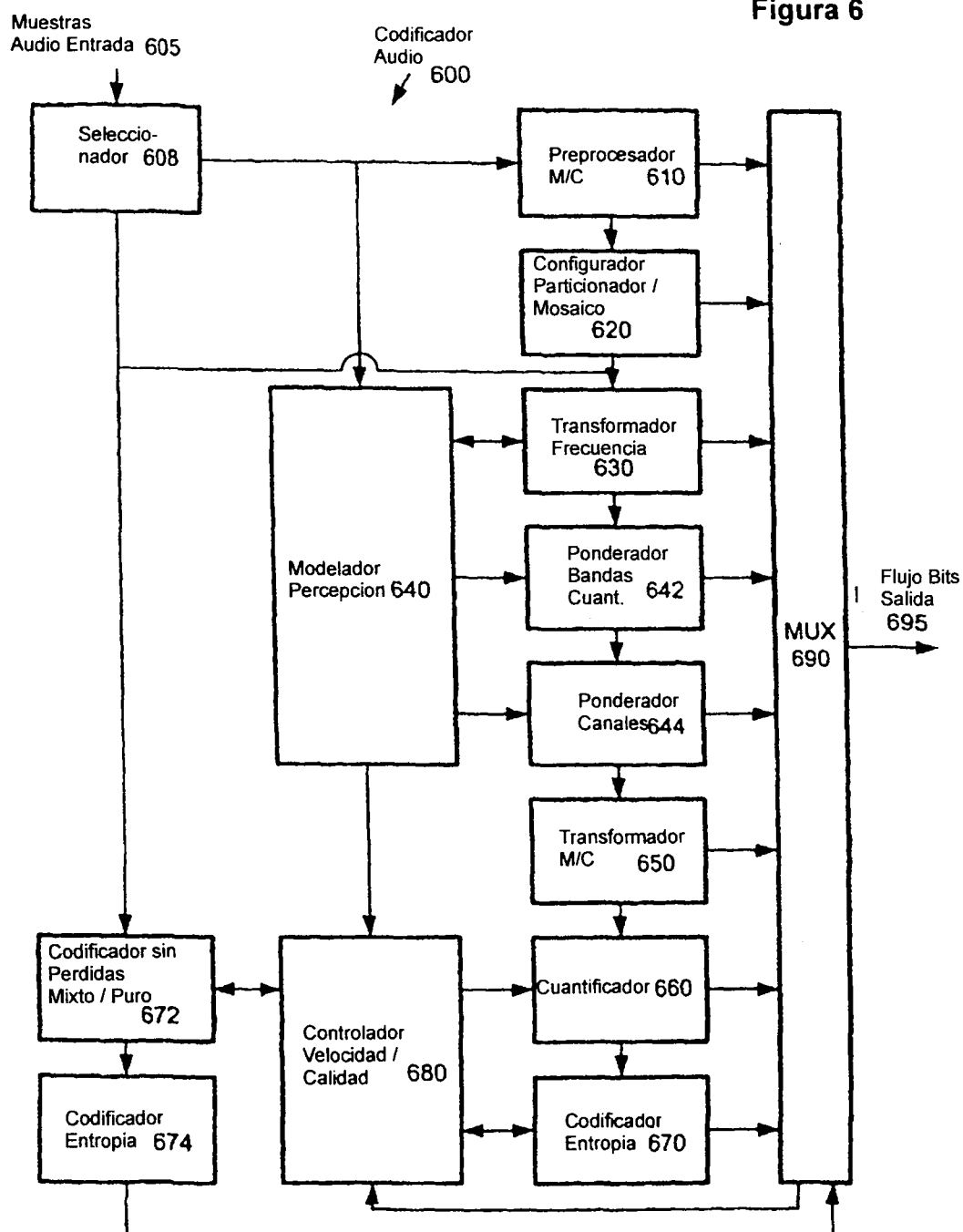


Figura 6



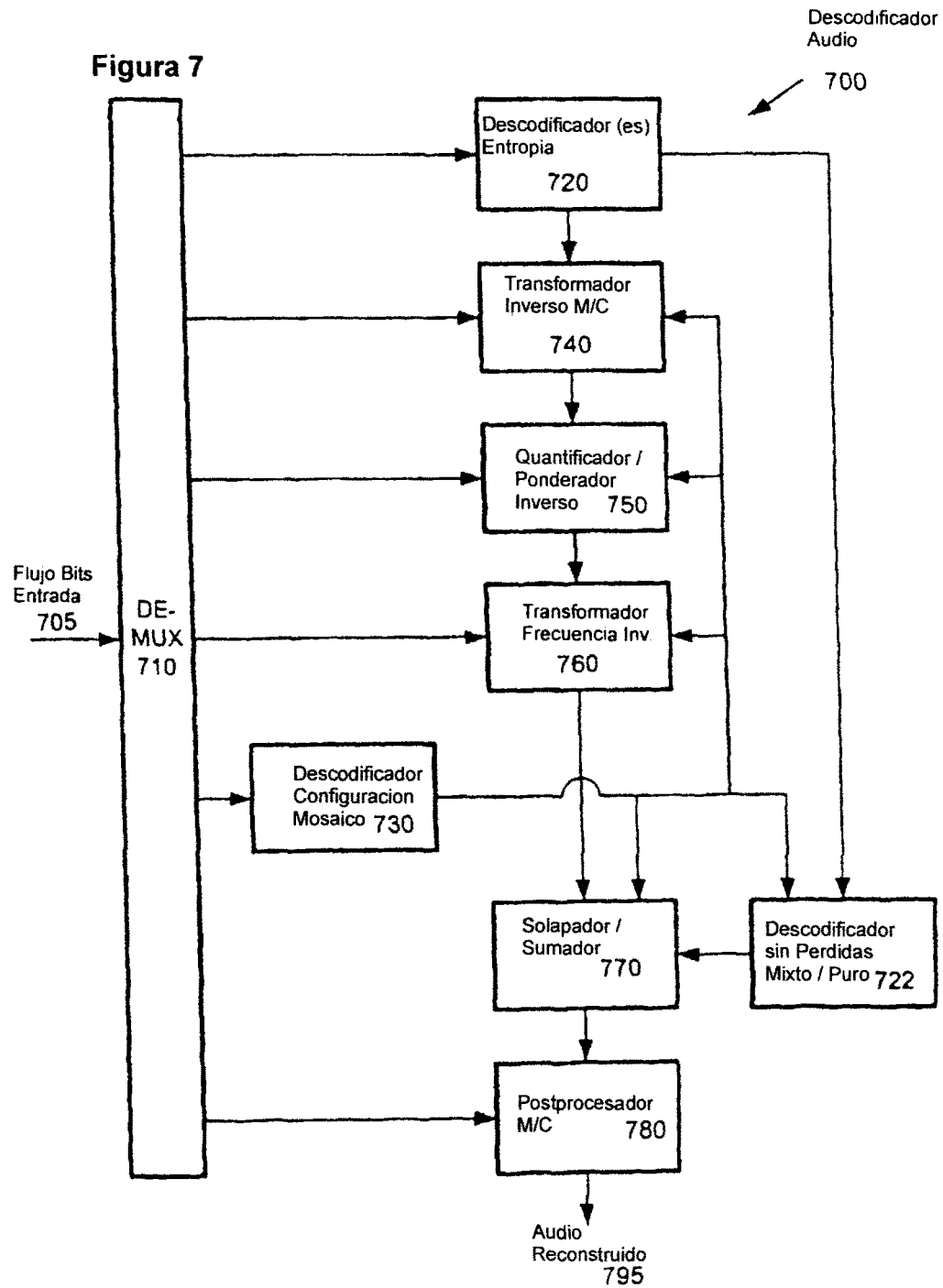


Figura 8

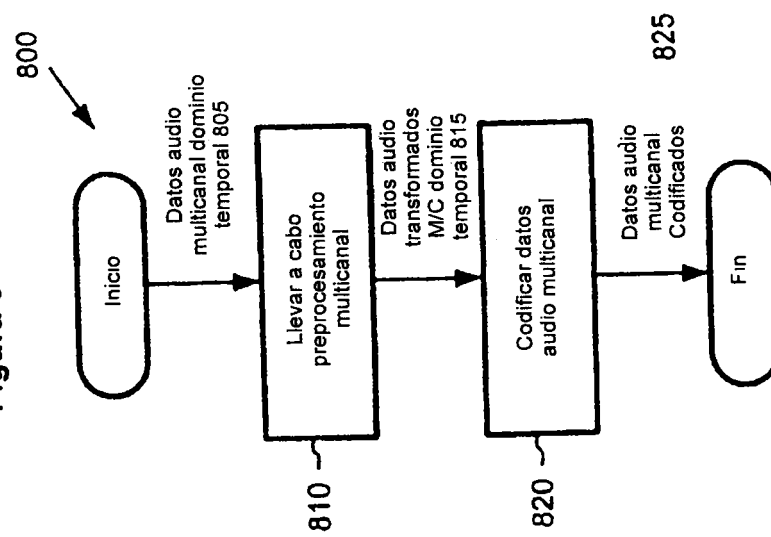
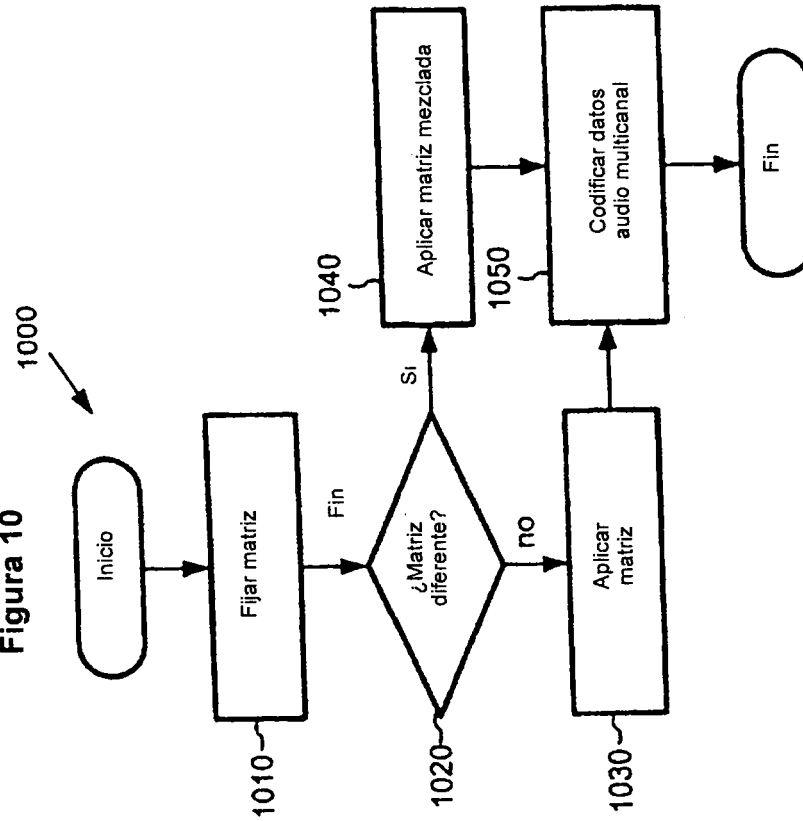


Figura 10



**Figura 9a**

$$A_{\text{bajo}} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Matriz transformacion  
preprocesamiento buena  
calidad

900

**Figura 9b**

$$A_{\text{inter. 1}} = \begin{bmatrix} \left( \frac{1}{1+0,5 \times \alpha} \right) & 0 & \left( \frac{0,5 \times \alpha}{1+0,5 \times \alpha} \right) & 0 & 0 & 0 \\ 0 & \left( \frac{1}{1+0,5 \times \alpha} \right) & \left( \frac{0,5 \times \alpha}{1+0,5 \times \alpha} \right) & 0 & 0 & 0 \\ \left( \frac{\alpha}{1+2\alpha} \right) & \left( \frac{\alpha}{1+2\alpha} \right) & \left( \frac{1}{1+2\alpha} \right) & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & \left( \frac{1}{1+\alpha} \right) & \left( \frac{\alpha}{1+\alpha} \right) \\ 0 & 0 & 0 & 0 & \left( \frac{\alpha}{1+\alpha} \right) & \left( \frac{1}{1+\alpha} \right) \end{bmatrix}$$

Primera matriz  
transformacion  
preprocesamiento  
calidad intermedia

901

**Figura 9c**

$$A_{\text{alto. 1}} = \begin{bmatrix} \left( \frac{1}{1,5} \right) & 0 & \left( \frac{0,5}{1,5} \right) & 0 & 0 & 0 \\ 0 & \left( \frac{1}{1,5} \right) & \left( \frac{0,5}{1,5} \right) & 0 & 0 & 0 \\ \left( \frac{1}{3} \right) & \left( \frac{1}{3} \right) & \left( \frac{1}{3} \right) & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0,5 & 0,5 \\ 0 & 0 & 0 & 0 & 0,5 & 0,5 \end{bmatrix}$$

Primera matriz  
transformacion  
preprocesamiento  
calidad deficiente

902



**Figura 9d**

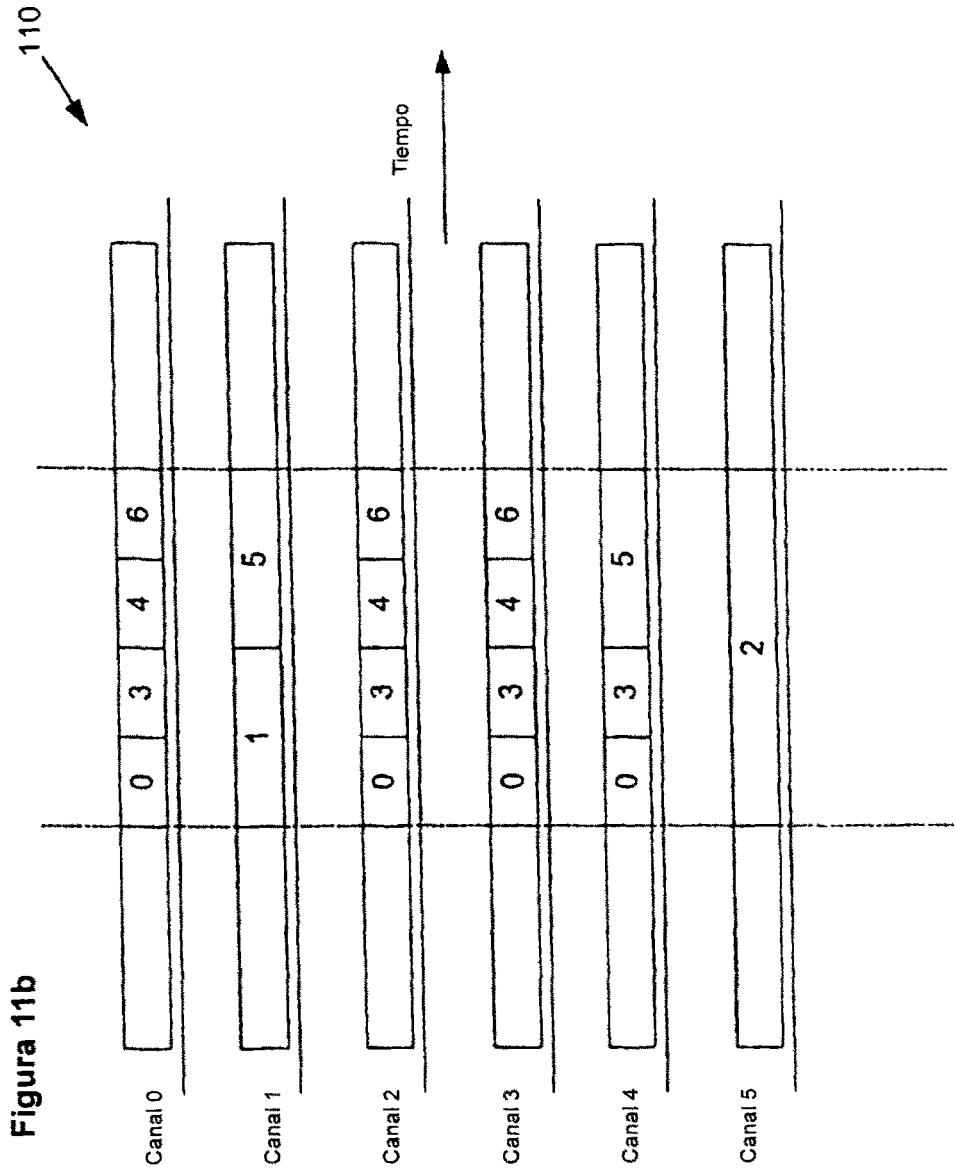
$$A_{\text{inter}, 2} = \begin{bmatrix} \left( \frac{1}{1+0,5\alpha} \right) & 0 & \left( \frac{0,5\alpha}{1+0,5\alpha} \right) & 0 & 0 & 0 \\ 0 & \left( \frac{1}{1+0,5\alpha} \right) & \left( \frac{0,5\alpha}{1+0,5\alpha} \right) & 0 & 0 & 0 \\ 0,5\alpha & 0,5\alpha & 1-\alpha & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & \left( \frac{1}{1+\alpha} \right) & \left( \frac{\alpha}{1+\alpha} \right) \\ 0 & 0 & 0 & 0 & \left( \frac{\alpha}{1+\alpha} \right) & \left( \frac{1}{1+\alpha} \right) \end{bmatrix}$$

Segunda matriz  
transformacion  
preprocesamiento  
calidad intermedia  
903

**Figura 9e**

$$A_{\text{ato}, 2} = \begin{bmatrix} \left( \frac{1}{1,5} \right) & 0 & \left( \frac{0,5}{1,5} \right) & 0 & 0 & 0 \\ 0 & \left( \frac{1}{1,5} \right) & \left( \frac{0,5}{1,5} \right) & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0,5 & 0,5 \\ 0 & 0 & 0 & 0 & 0,5 & 0,5 \end{bmatrix}$$

Segunda matriz  
transformacion  
preprocesamiento  
calidad deficiente  
904



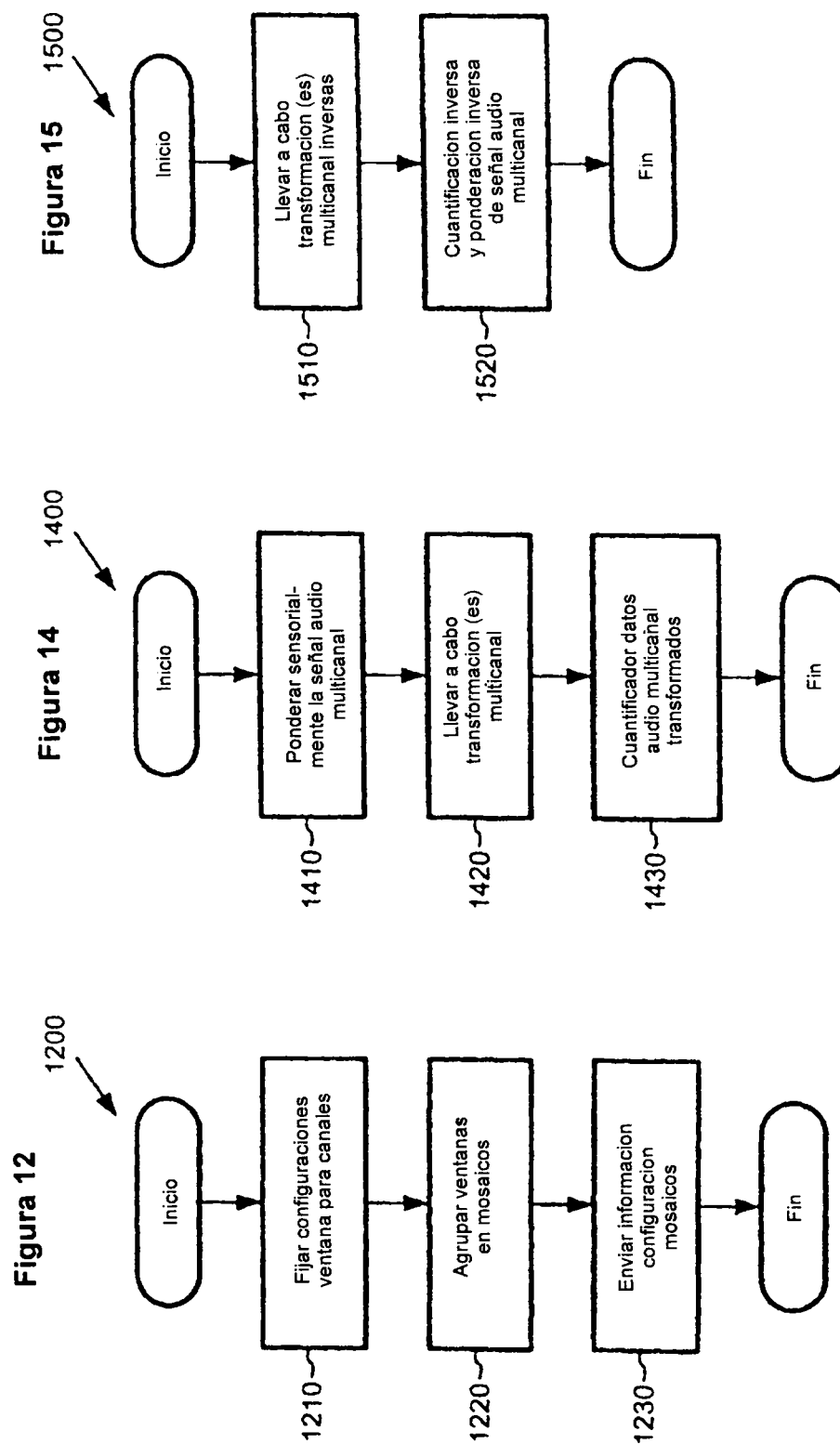


Figura 13

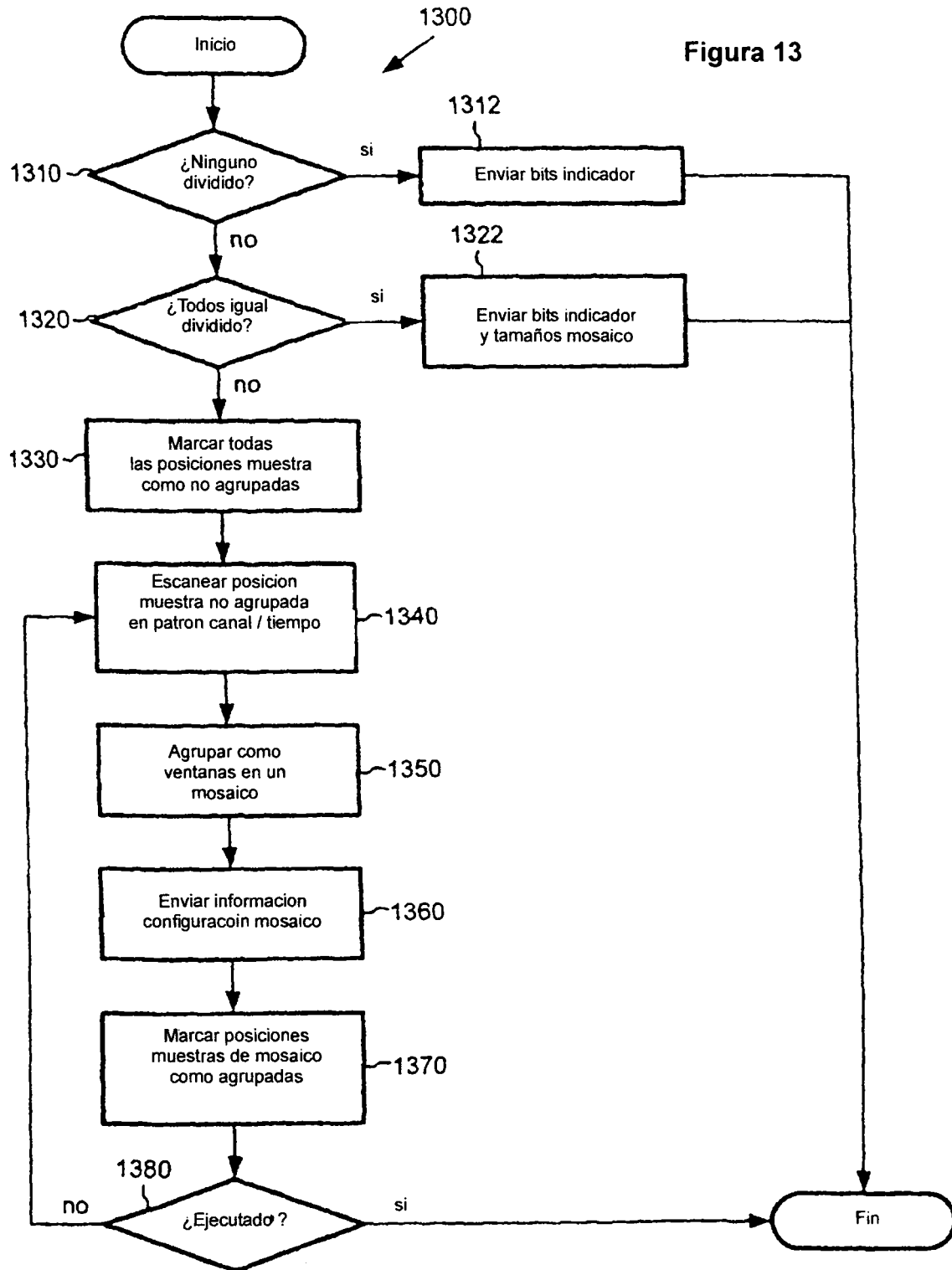


Figura 16

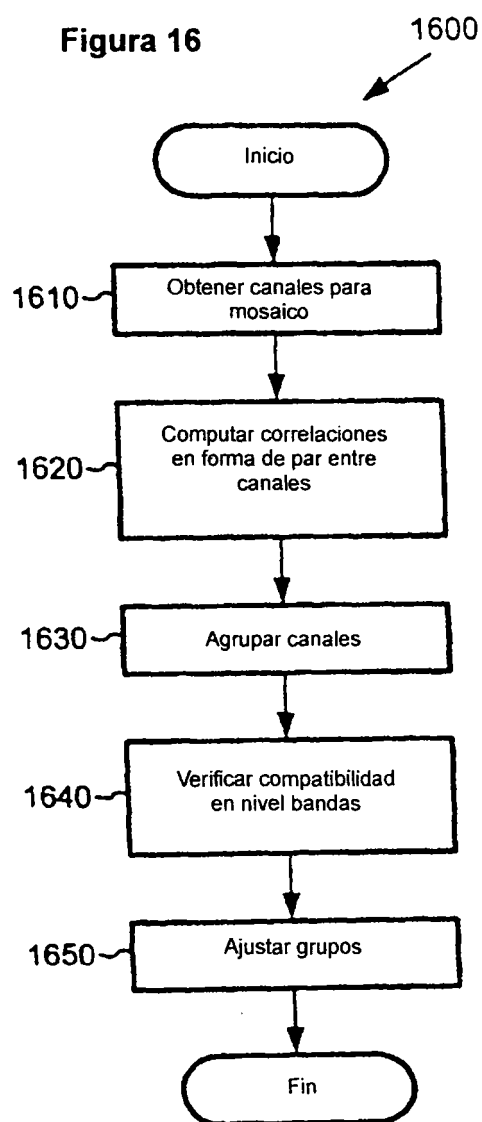


Figura 17

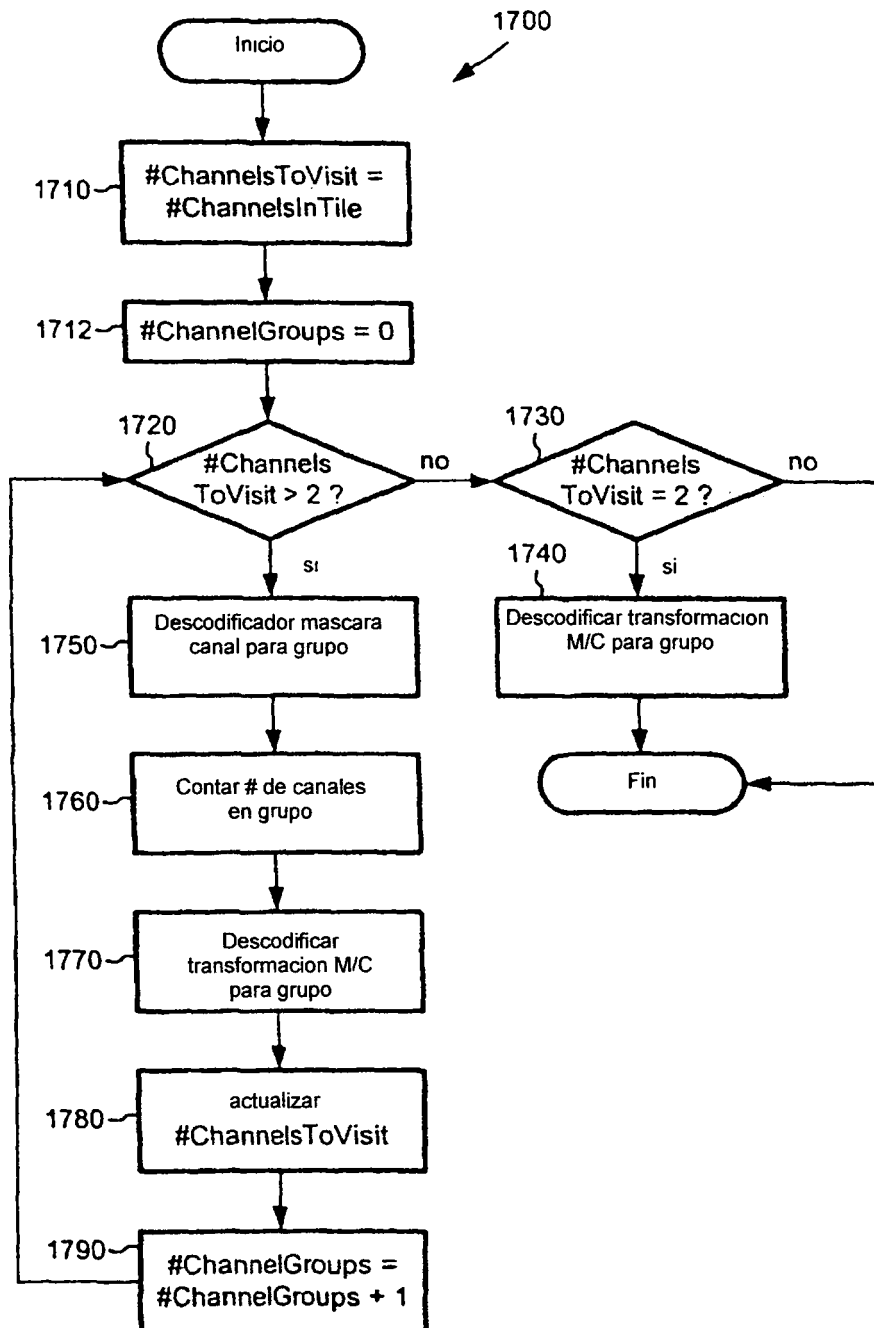


Figura 18

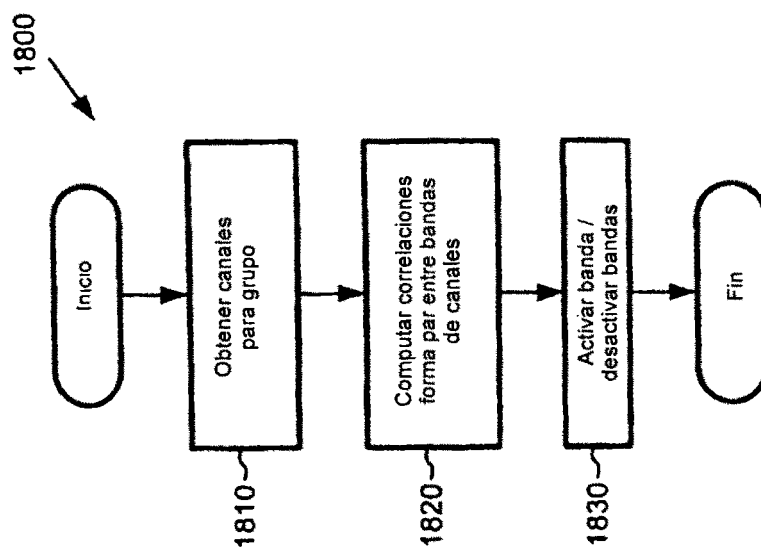


Figura 19

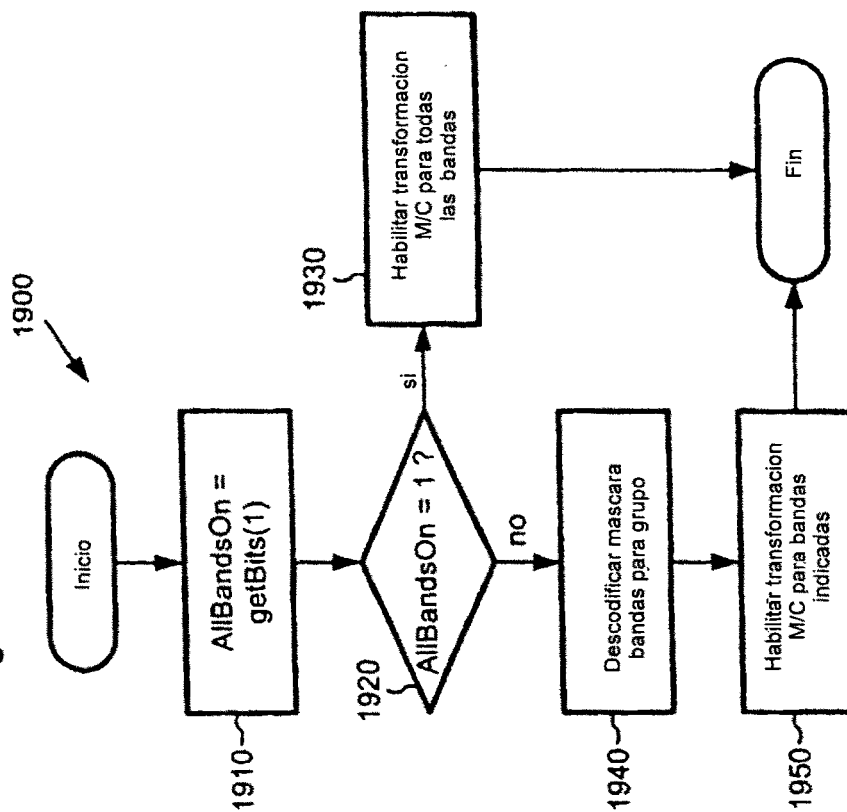


Figura 20

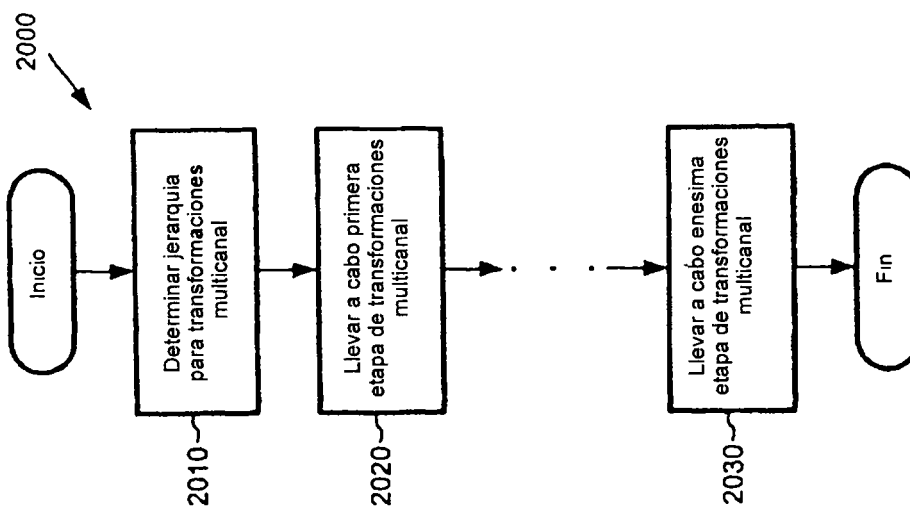
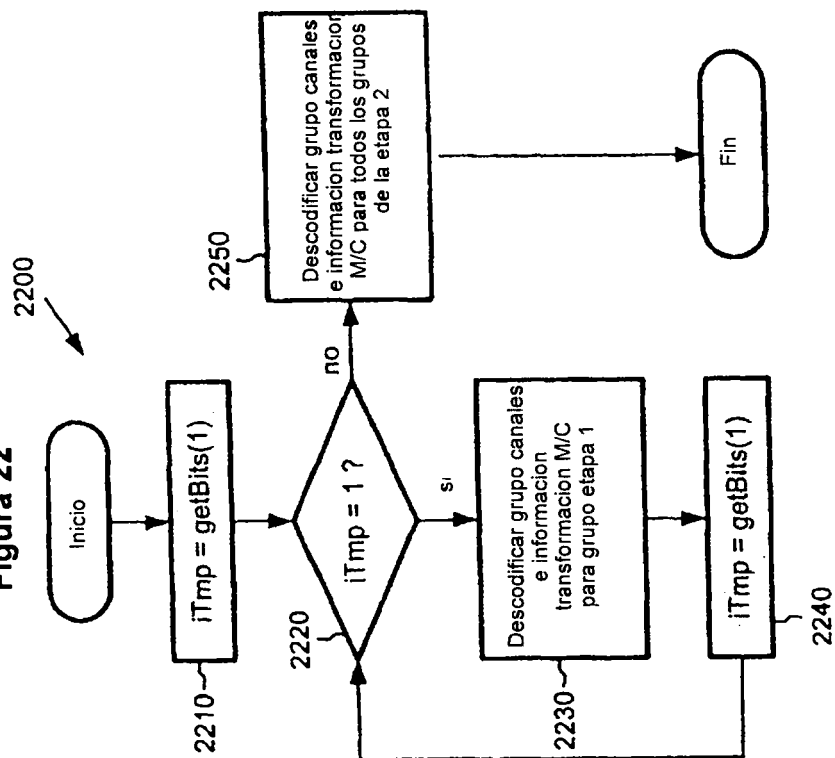


Figura 22





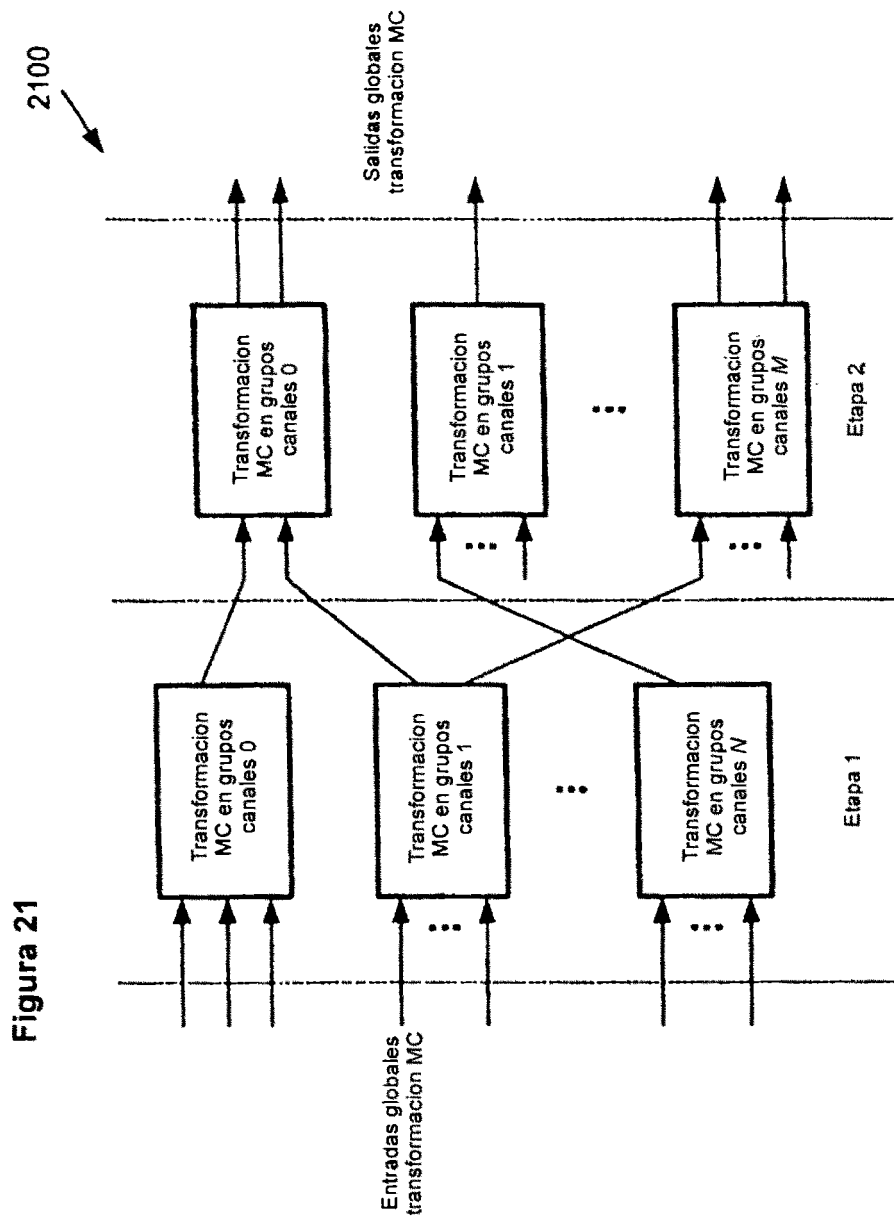


Figura 21

Figura 23

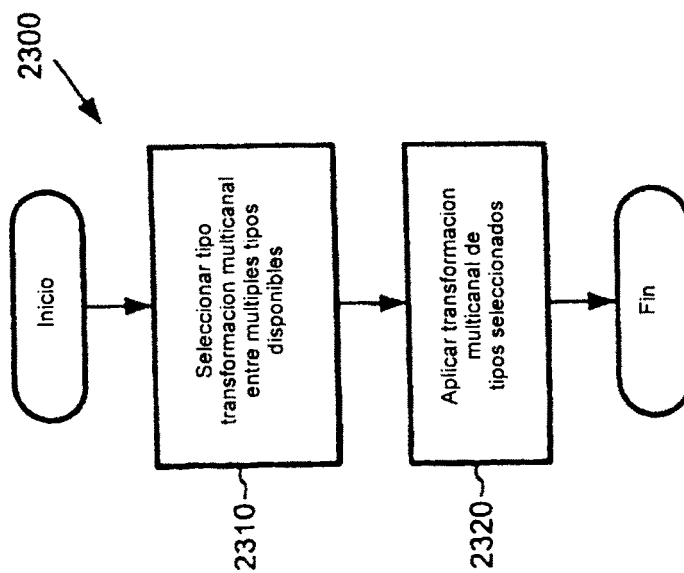


Figura 24

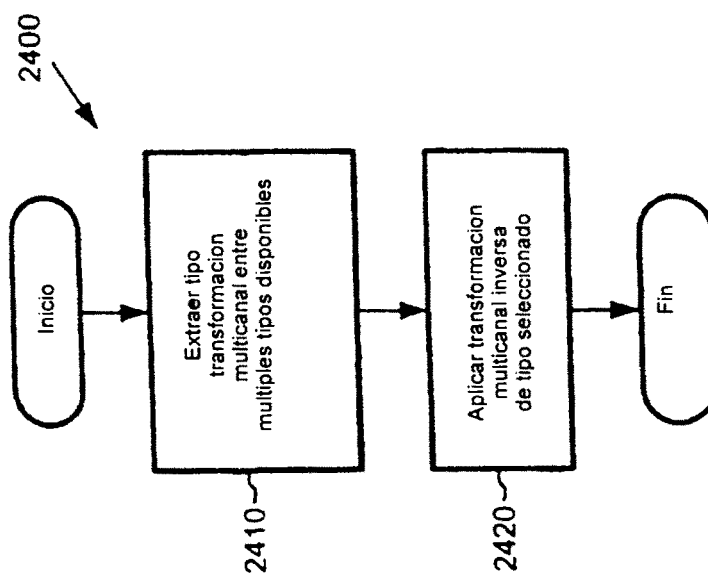


Figura 25

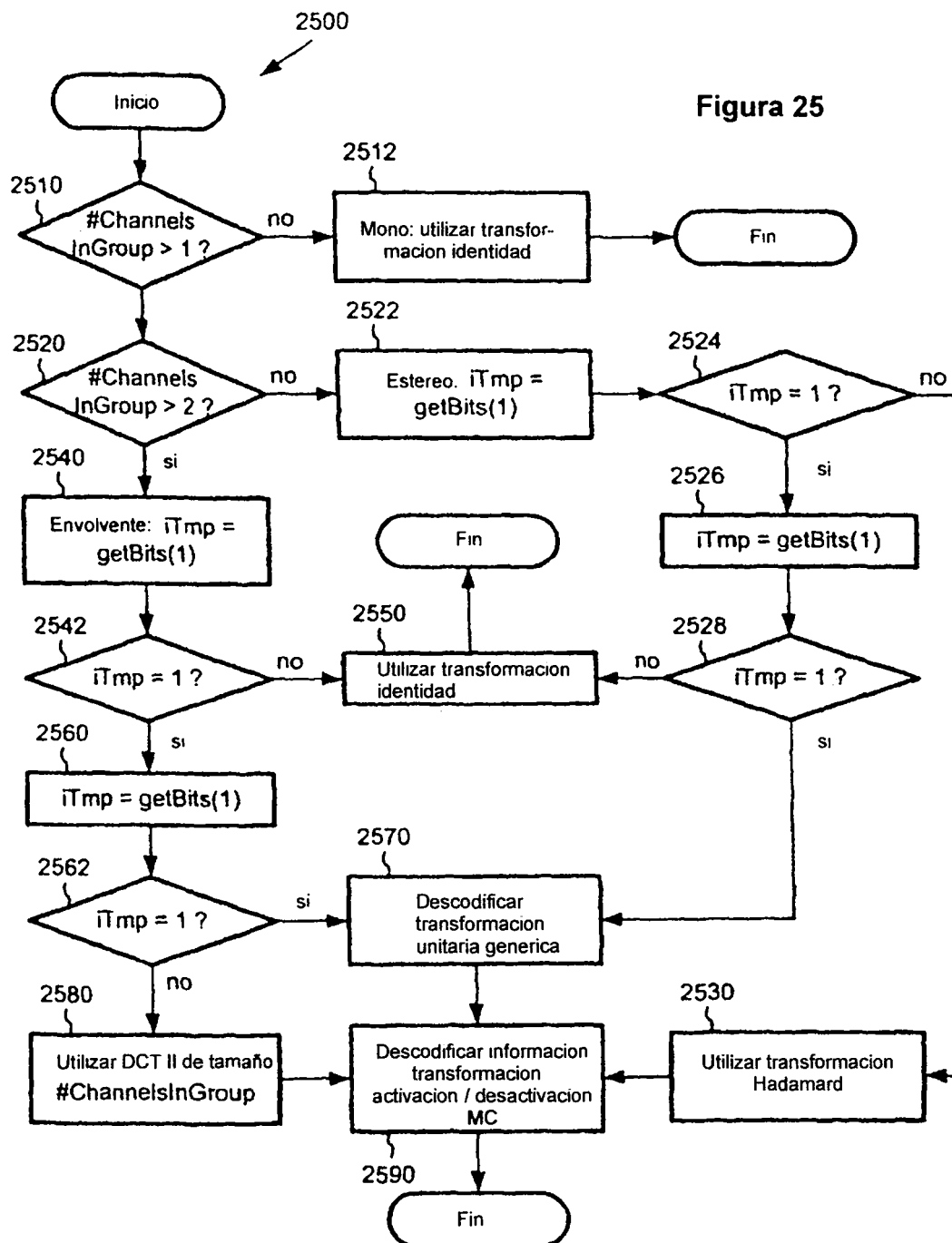


Figura 28

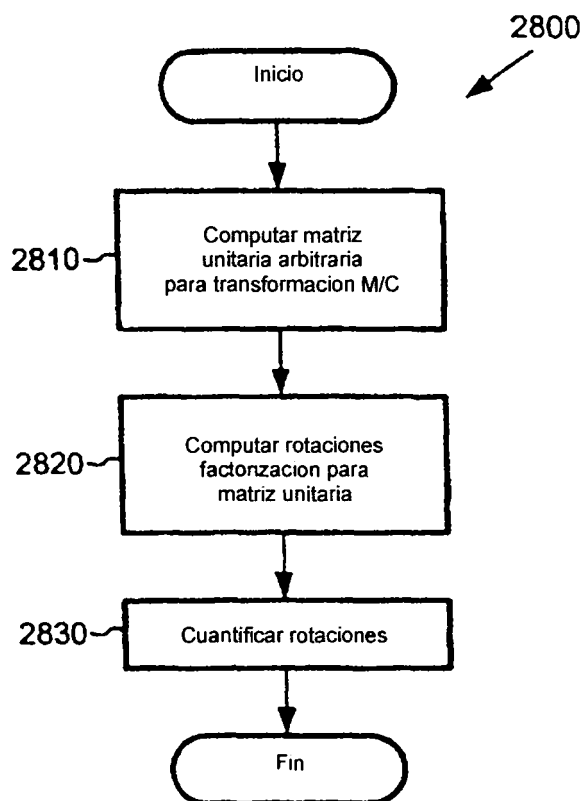


Figura 26

2600

$$\begin{bmatrix}
 1 & 0 & 0 & 0 & \dots & 0 & 0 & 0 & 0 \\
 0 & 1 & 0 & 0 & \dots & 0 & 0 & 0 & 0 \\
 0 & 0 & \cos \omega_k & 0 & \dots & \sin \omega_k & 0 & 0 & 0 \\
 0 & 0 & 0 & 1 & \dots & 0 & 0 & 0 & 0 \\
 \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\
 0 & 0 & -\sin \omega_k & 0 & \dots & \cos \omega_k & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & \dots & 0 & 1 & 0 & 0 \\
 0 & 0 & 0 & 0 & \dots & 0 & 0 & 1 & 0 \\
 0 & 0 & 0 & 0 & \dots & 0 & 0 & 0 & 1
 \end{bmatrix}$$

Figura 27a

$$\Theta_1 = \begin{bmatrix} \cos \varpi_1 & \text{sen } \varpi_1 & 0 & 0 & 0 & 0 & 0 & \dots & 0 \\ -\text{sen } \varpi_1 & \cos \varpi_1 & 0 & 0 & 0 & 0 & 0 & \dots & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & \dots & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & \dots & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & \dots & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \dots & 1 \end{bmatrix} \quad \swarrow 2700$$

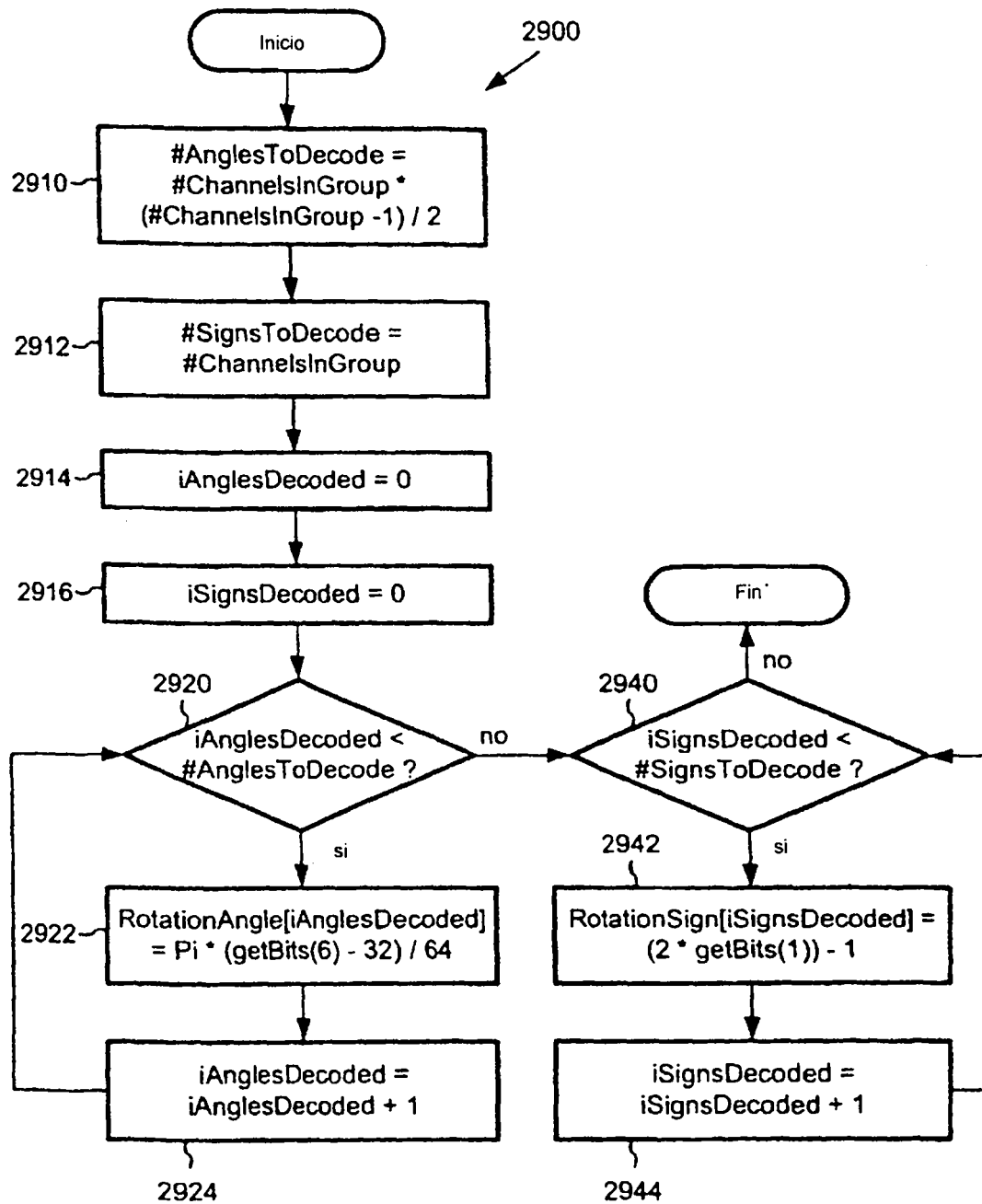
Figura 27b

$$\Theta_2 = \begin{bmatrix} \cos \varpi_2 & 0 & \text{sen } \varpi_2 & 0 & 0 & 0 & 0 & \dots & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & \dots & 0 \\ -\text{sen } \varpi_2 & 0 & \cos \varpi_2 & 0 & 0 & 0 & 0 & \dots & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & \dots & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & \dots & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \dots & 1 \end{bmatrix} \quad \swarrow 2701$$

Figura 27c

$$\Theta_3 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & \dots & 0 \\ 0 & \cos \varpi_3 & \text{sen } \varpi_3 & 0 & 0 & 0 & 0 & \dots & 0 \\ 0 & -\text{sen } \varpi_3 & \cos \varpi_3 & 0 & 0 & 0 & 0 & \dots & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & \dots & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & \dots & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \dots & 1 \end{bmatrix} \quad \swarrow 2702$$

Figura 29



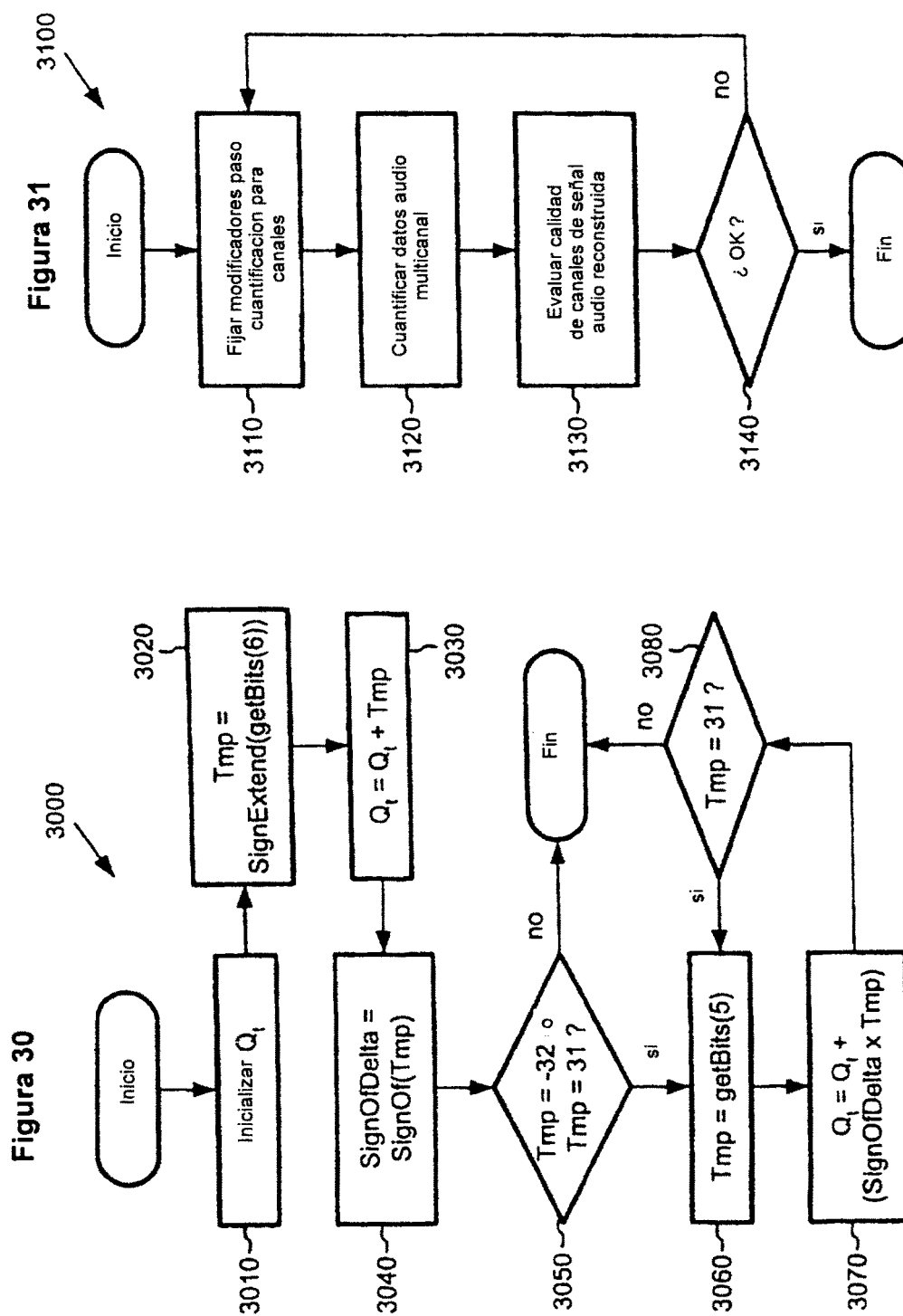
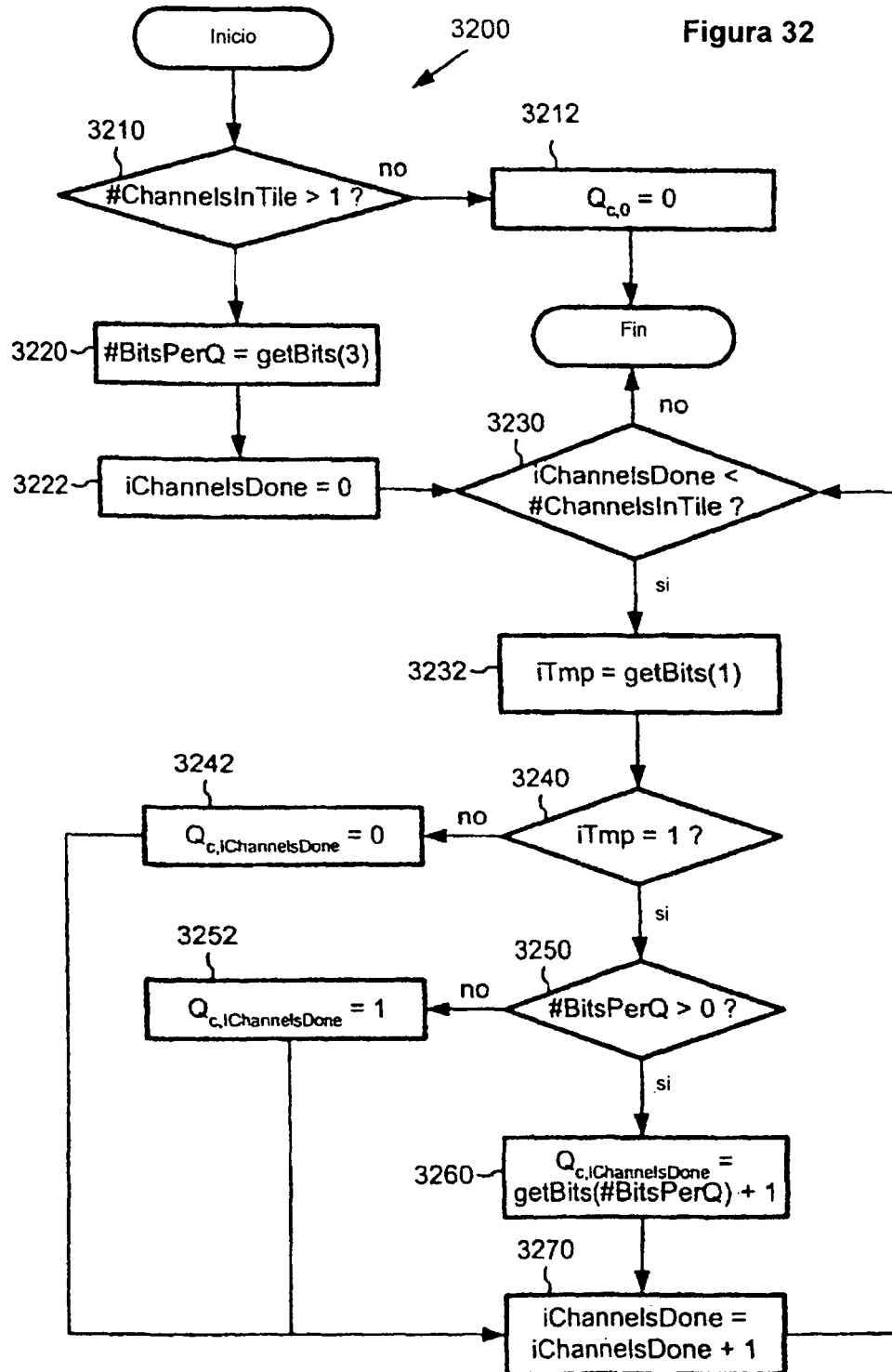


Figura 32





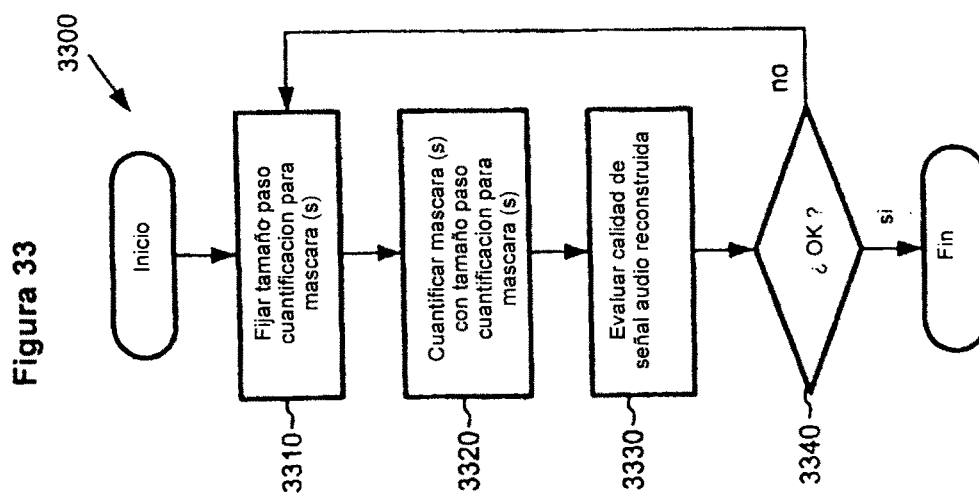
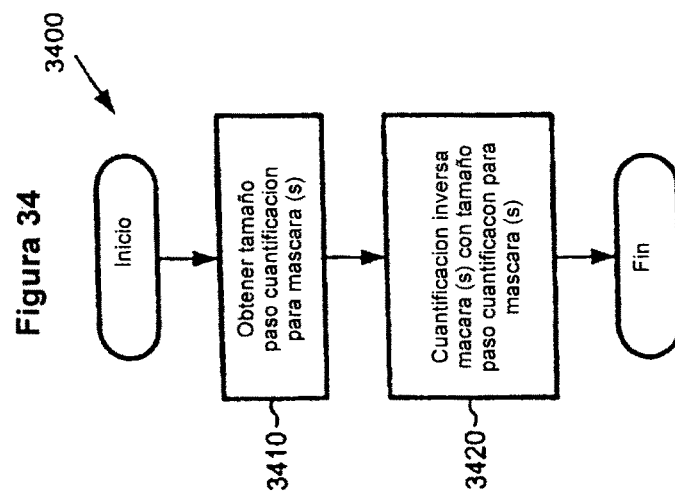


Figura 35

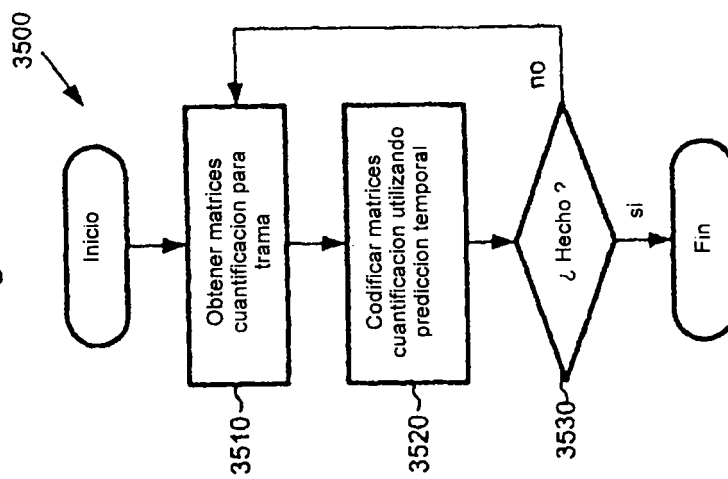
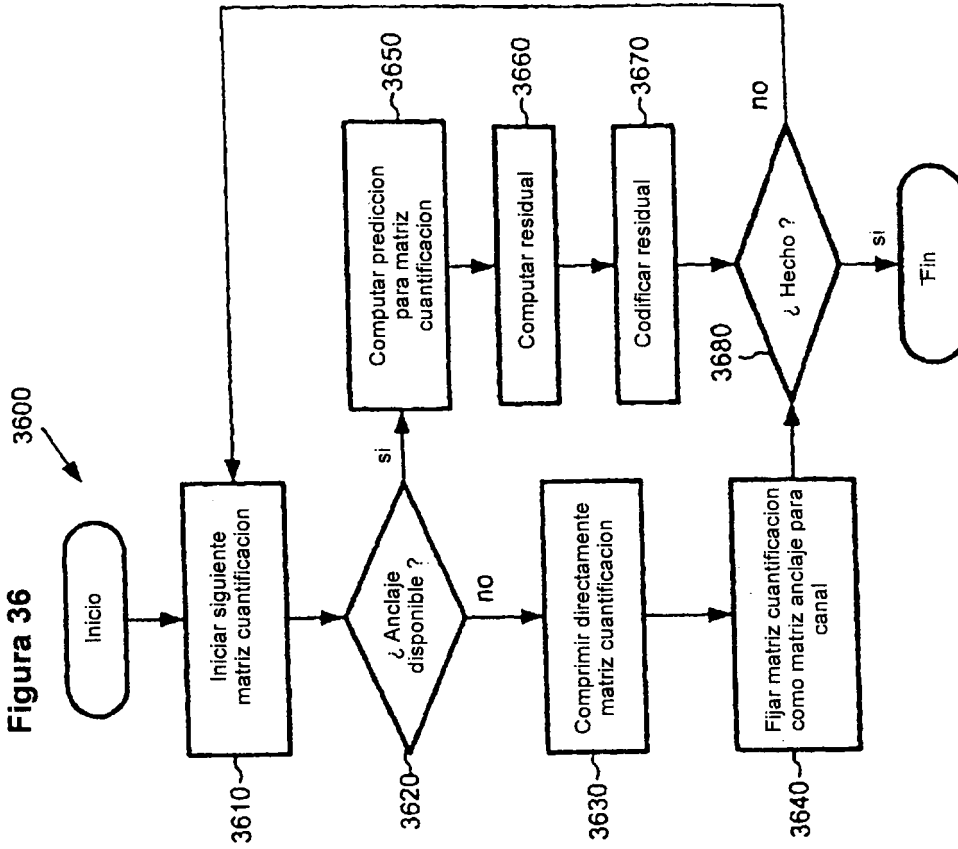
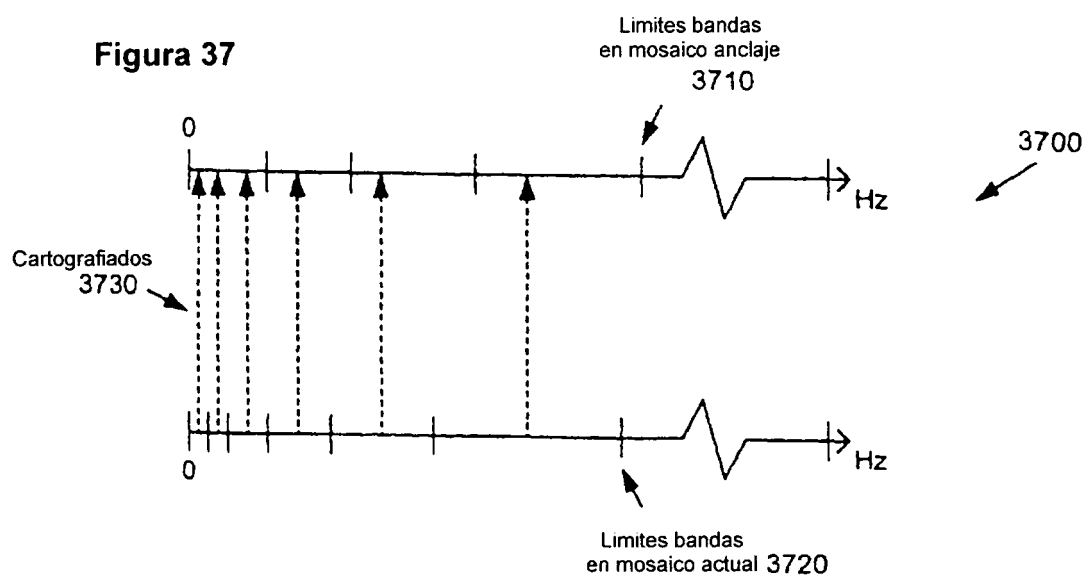


Figura 36



**Figura 37**

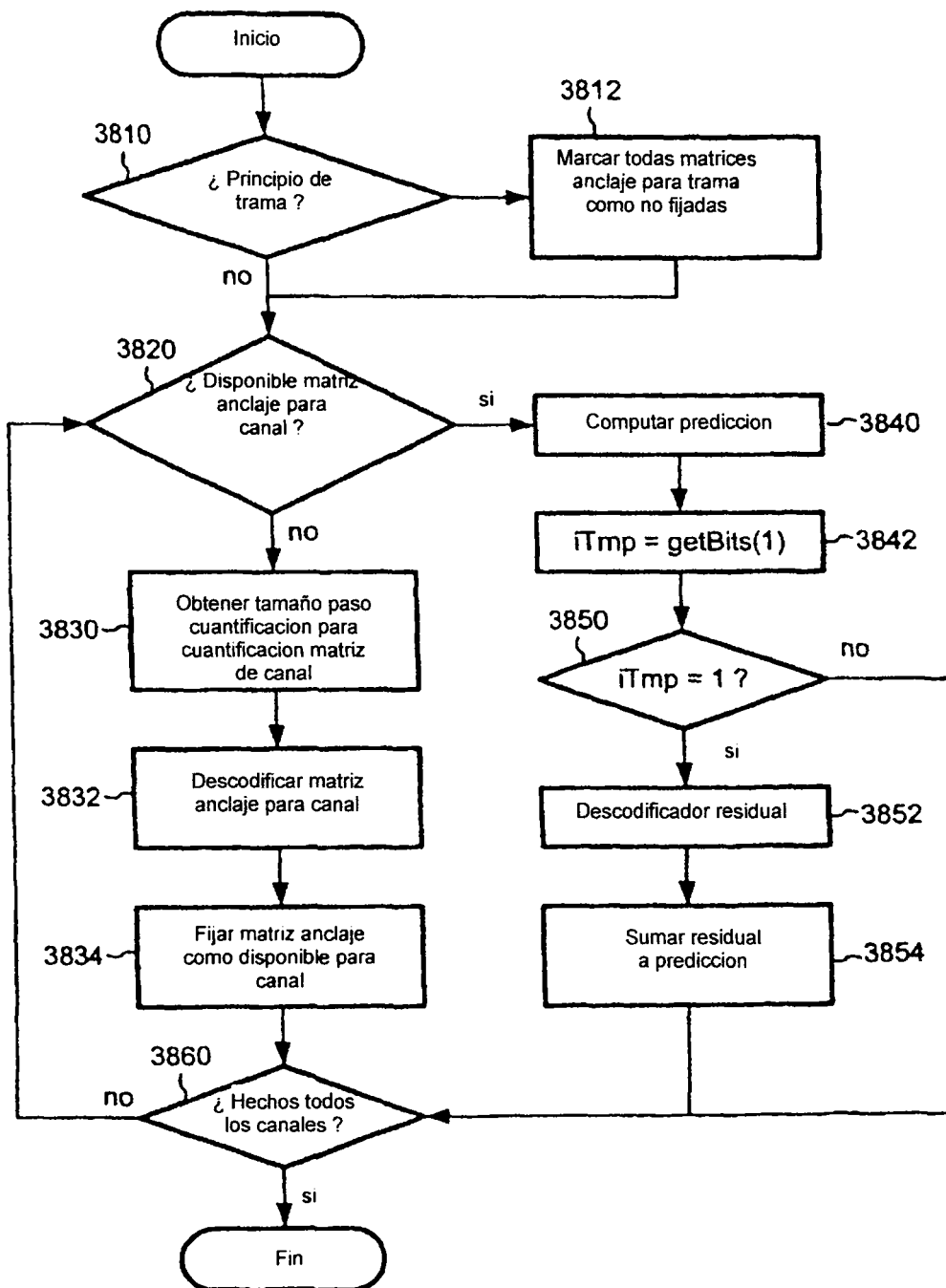


**Figura 40**

$$A_{P\text{-Center}} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0,5 & 0,5 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Matriz transformacion postprocesamiento 4000

Figura 38



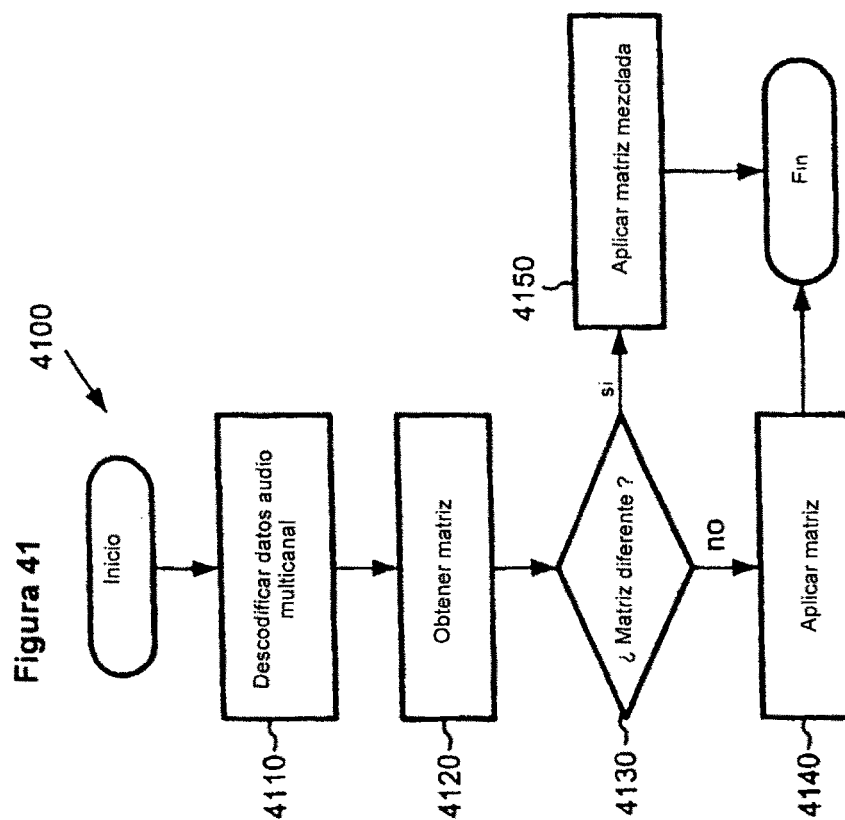
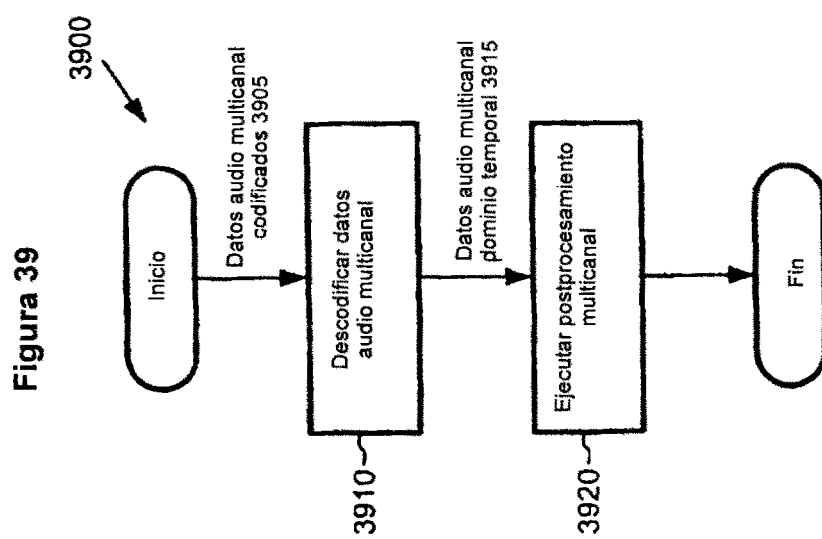


Figura 42

