

19



OFICINA ESPAÑOLA DE  
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 399 562**

51 Int. Cl.:

**H04S 3/00** (2006.01)

**H04S 5/02** (2006.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

96 Fecha de presentación y número de la solicitud europea: **15.10.2007 E 10171797 (3)**

97 Fecha y número de publicación de la concesión europea: **14.11.2012 EP 2299734**

54 Título: **Método y codificador para combinar conjuntos de datos digitales, método para descodificar y descodificador para tales conjuntos de datos digitales combinados y soporte de grabación para almacenar tales conjuntos de datos digitales combinados**

30 Prioridad:

**13.10.2006 US 829321 P**

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

**02.04.2013**

73 Titular/es:

**AURO TECHNOLOGIES (100.0%)  
Kievitstraat, 42  
2400 Mol, BE**

72 Inventor/es:

**VAN DEN BERGHE, GUIDO y  
VAN BAELEN, WILFRIED**

74 Agente/Representante:

**DE ELZABURU MÁRQUEZ, Alberto**

**ES 2 399 562 T3**

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

**DESCRIPCIÓN**

Método y codificador para combinar conjuntos de datos digitales, método para descodificar y descodificador para tales conjuntos de datos digitales combinados y soporte de grabación para almacenar tales conjuntos de datos digitales combinados

5 **Ámbito de la invención**

La invención se refiere a un método para combinar un primer conjunto de datos digitales de muestras con un primer tamaño y un segundo conjunto de datos digitales de muestras con un segundo tamaño en un tercer conjunto de datos digital de muestras con un tercer tamaño más pequeño que la suma del primer tamaño y el segundo tamaño.

**Técnica anterior**

10 Un método así se conoce a partir del documento EP1592008 en el que se describe un método para mezclar dos conjunto de datos digitales en un tercer conjunto de datos digitales. Con el fin de encajar dos conjuntos de datos digitales en un solo conjunto de datos digitales con un tamaño menor que la suma de los tamaños de los dos conjuntos de datos digitales, es necesario una reducción de la información en los dos conjuntos de datos digitales. El documento EP1592008 consigue esta reducción al definir una interpolación de muestras entre un primer conjunto de  
 15 posiciones predefinidas en el primer conjunto de datos digitales y en un conjunto no coincidente de muestras entre posiciones predefinidas en el segundo conjunto de datos digitales. El valor de las muestras entre las posiciones predefinidas de los conjuntos de datos digitales se establece como el valor de la interpolación. Después de realizar esta reducción de la información en los dos conjuntos de datos digitales, cada muestra del primer conjunto de datos digitales se suma con la muestra correspondiente del segundo conjunto de datos digitales. Esto da lugar a un tercer  
 20 conjunto de datos digital que comprende las muestras sumadas. Esta suma de las muestras, junto con una relación conocida del desplazamiento entre las posiciones predefinidas entre el primer conjunto de datos digitales y el segundo conjunto de datos digitales permite la recuperación del primer conjunto de datos digitales y el segundo conjunto de datos digitales, aunque sólo con las muestras interpoladas entre las posiciones predefinidas. Cuando se utiliza el método del documento EP 1592008 para secuencias de audio esta interpolación no es apreciable y el tercer  
 25 conjunto de datos digitales puede reproducirse como una representación mezclada de los dos conjuntos de datos digitales comprendidos. Con el fin de permitir la recuperación del primer y segundo conjunto de datos digitales con las muestras interpoladas, se debe conocer un valor inicial para los dos conjuntos primero y segundo de datos digitales y, por lo tanto, estos dos valores se almacenan también durante la mezcla para permitir que un descifrado posterior de los dos conjuntos de datos digitales a partir del tercer conjunto de datos digitales.

30 El método del documento EP1592008 tiene la desventaja de que necesita un procesamiento intensivo por el lado de la codificación.

**Resumen de la invención**

El objetivo de la presente invención es reducir el procesamiento necesario en el lado de la codificación.

Con el fin de lograr este objetivo, el método de la presente invención comprende las etapas de:

- 35 - igualar un primer subconjunto de muestras del primer conjunto de datos digitales con muestras vecinas de un segundo subconjunto de muestras del primer conjunto de datos digitales en el que se intercalan el primer subconjunto de muestras y el segundo subconjunto de muestras,
- igualar un tercer subconjunto de muestras del segundo conjunto de datos digitales con muestras vecinas de un cuarto subconjunto de muestras del segundo conjunto de datos digitales en el que se intercalan el tercer  
 40 subconjunto de muestras y el cuarto subconjunto de muestras,
- crear las muestras del tercer conjunto de datos digitales mediante la adición de las muestras del primer conjunto de datos digital en el dominio temporal a las muestras correspondientes del segundo conjunto de datos digitales,
- integrar una primera muestra semilla del primer conjunto de datos digitales y una segunda muestra semilla  
 45 del segundo conjunto de datos digitales en el conjunto de datos digitales.

Al sustituir la etapa de interpolación del método del documento EP1592008 por una etapa en la que los valores entre las posiciones predefinidas se establecen con el valor de una muestra adyacente, la intensidad de procesamiento se reduce considerablemente en el lado de la codificación. La señal resultante aún permite el descifrado (es decir, la extracción) de los dos conjuntos de datos digitales a partir del tercer conjunto de datos digitales. El tercer conjunto  
 50 de datos digitales, cuando se combinan dos secuencias digitales de audio en una única secuencia digital de audio, sigue siendo una buena representación mono de las dos secuencias combinadas de audio digital.

La invención se basa en la comprensión de que la interpolación es innecesaria en el lado de la codificación, ya que puede realizarse igualmente bien en el lado de la descodificación ya que el presente método de combinación y descifrado deja las muestras del primer y segundo conjunto de datos digitales en sus respectivas posiciones

predefinidas intactas y recuperables, permitiendo de este modo la interpolación de las muestras entre las muestras intactas después de la descodificación del tercer conjunto de datos digitales. El tercer conjunto de datos digitales de la reivindicación independiente de la presente invención difiere del tercer conjunto de datos digitales del documento EP1592008 en que por lo general existe un error más grande entre la suma real del primer y segundo conjunto de datos digitales y el tercer conjunto de datos digitales en el caso de la presente invención.

La igualación de un primer subconjunto de muestras del primer conjunto de datos digitales con muestras vecinas de un segundo subconjunto de muestras del primer conjunto de datos digitales en el que se intercala el primer subconjunto de muestras y el segundo subconjunto de muestras, realiza una reducción ejecutada fácilmente en la información del primer conjunto de datos digitales. La igualación de un tercer subconjunto de muestras del segundo conjunto de datos digitales con muestras vecinas de un cuarto subconjunto de muestras del segundo conjunto de datos digitales en el que se intercala el tercer subconjunto de muestras y el cuarto subconjunto de muestras, realiza una reducción ejecutada fácilmente en la información del segundo conjunto de datos digitales. Al hacer disponibles los valores originales del primer y segundo conjunto de datos digitales, en los que los valores originales pueden funcionar como un valor semilla, y al asegurar que el segundo y cuarto subconjunto también se intercalan, el primer y segundo conjunto de datos digitales pueden recuperarse a partir del tercer conjunto de datos digitales en el estado en el que se igualó el primer subconjunto de muestras del primer conjunto de datos digitales a muestras vecinas de un segundo subconjunto de muestras del primer conjunto de datos digitales y el tercer subconjunto de muestras del segundo conjunto de datos digitales con muestras vecinas de un cuarto subconjunto de muestras del segundo conjunto de datos digitales. Una vez que el primer y segundo conjunto de datos digitales se han recuperado en este estado, se puede utilizar interpolación o filtrado para restaurar con la mayor precisión posible los valores originales del primer subconjunto de muestras de la primera secuencia de datos digitales y el tercer subconjunto de muestras de la segunda secuencia de datos digitales. Por lo tanto el método que combina una primera secuencia de datos digitales y una segunda secuencia de datos digitales en una tercera secuencia de datos digitales permite la recuperación con alta precisión del segundo y cuarto subconjunto de muestras y la reconstrucción del primer y el tercer subconjunto de valores y, si es necesario, durante la descodificación se puede realizar la etapa de interpolación.

El dispositivo del usuario final que comprende el descodificador puede decidir qué nivel de calidad consigue de reconstrucción ya que la interpolación puede ser seleccionada y realizada por el descodificador en lugar de ser dictada por el codificador.

Al no imponer ninguna interpolación del primer y segundo conjunto de datos digitales, pero incluir una aproximación de error oculta en los bits menos significativos de la tercera secuencia de datos digitales, se consigue una ventaja porque la etapa de descodificación es libre de elegir que reconstrucción se ha de aplicar. Sin embargo, cuando la aproximación de error también se utilizó durante la composición del 3<sup>er</sup> conjunto digital (que es la mezcla de muestras de un 1<sup>er</sup> y un 2<sup>o</sup> conjunto digital, incluyendo los errores aproximados), los valores de aproximación de error ocultos en los bits menos significativos, tienen que ser utilizados también durante el proceso de descodificación con el fin de realizar la reconstrucción de los conjuntos originales de datos digitales, es decir, los canales originales de audio digital.

Se puede elegir que la reconstrucción durante la descodificación utilice la aproximación de error tal como se almacena en los bits menos significativos y realice una interpolación lineal entre los valores de las muestras en posiciones predefinidas, ya que son totalmente recuperables excepto por la pérdida de información en los bits menos significativos. De este modo, el sistema de codificación y descodificación se puede utilizar de manera más flexible.

La codificación puede minimizar sólo el procesamiento y juntar la primera y la segunda secuencia de datos digitales en la tercera secuencia de datos digitales sin añadir la aproximación de error y sólo ajustar los valores de las muestras entre las posiciones predeterminadas al valor de las muestras adyacentes, o la aproximación de error se puede seleccionar de un conjunto limitado de aproximaciones de error y ser añadida a los bits menos significativos del tercer conjunto de datos digitales.

En una realización del método el primer conjunto de datos digitales representa una primera señal de audio y el segundo conjunto de datos digitales representa una segunda señal de audio. Mediante la aplicación de la presente invención a las señales de audio no sólo se consigue que la primera y segunda señal de audio se puedan recuperar con una precisión aceptable, sino que la señal de audio combinada resultante, según se represente mediante el tercer conjunto de datos digitales, es una representación aceptable en sentido de percepción de la primera señal de audio cuando se mezcla con la segunda señal de audio. Por lo tanto, se consigue que el tercer conjunto de datos digitales resultante pueda ser reproducido correctamente en equipos que no son capaces de extraer la primera o la segunda señal de audio digital del tercer conjunto de datos digitales, mientras que los equipos capaces de realizar la extracción pueden extraer la primera y segunda señal de audio para la reproducción por separado o para un procesamiento posterior. Cuando se combinan, es decir, se mezclan, más de dos señales de audio utilizando esta invención también es posible extraer sólo una de las señales de audio, dejando combinadas las otras señales de audio. Estas señales de audio restantes todavía dan una señal de audio que se puede reproducir en representación de la mezcla de las señales de audio todavía combinadas, mientras que la señal de audio extraída puede ser procesada por sí misma.

Como herramienta para los ingenieros de grabación - es posible una emulación en tiempo real de la mezcla de pares de canales de audio en los canales individuales. Esto creará una salida de audio, durante la edición de la grabación como parte del proceso de creación, que representará la calidad mínima garantizada del proceso final de mezcla, así como una calidad mínima de los canales separados o descodificados. Una vez que se crea un conjunto básico de datos PCM (del inglés *Pulse Code Modulation*: Modulación por Impulsos Codificados) en múltiples canales AURO-fónicos, los parámetros de codificación adicionales para aumentar la calidad de las señales mezcladas se pueden calcular fuera de línea, eliminando la necesidad de procesamiento en tiempo real.

En otra realización del método la primera muestra semilla es la primera muestra del primer conjunto de datos digitales y la segunda muestra semilla es la segunda muestra del segundo conjunto de datos digitales. La selección de muestras semilla para el descifrado cerca del comienzo del conjunto de datos digitales permite que el comienzo del descifrado del primer y segundo conjunto de datos digitales comience tan pronto como se comienza a leer el tercer conjunto de datos digitales. Las muestras semilla también pueden integrarse, es decir situarse, además en el tercer conjunto de datos digitales de manera que se necesitaría un enfoque repetitivo para descifrar las muestras situadas antes de las muestras semilla. La selección de muestras semilla a partir del conjunto original de datos digitales en, o antes de, el inicio de ese conjunto simplifica el proceso de descifrado para recuperar el primer y segundo conjunto de datos digitales. En una realización adicional del método la primera muestra semilla y la segunda muestra semilla se integran en los bits menos significativos de las muestras del tercer conjunto de datos digitales. Mediante la integración de los valores semilla en los bits menos significativos de las muestras, las muestras afectadas sólo se desviarán ligeramente de los valores originales, lo que ha demostrado ser casi imperceptible, ya que sólo es necesario almacenar unos pocos valores semilla y, como tal, sólo se ven afectadas pocas muestras. Además la selección de los bits menos significativos garantiza que sólo se producen pequeñas desviaciones. Incluso cuando se utilizan los bits menos significativos de todas las muestras para integrar datos, esta desviación no es, o lo es apenas, perceptible ya que los bits menos significativos se eliminan de la muestra y esto resulta ser apenas perceptible. Esta eliminación de los bits menos significativos de las muestras reduce el espacio necesario para almacenar el conjunto de datos digitales en el que están comprendidas estas muestras, y de este modo se libera más espacio en el soporte de grabación o en el canal de transmisión o se permite la integración de datos adicionales tales como para fines de control. Separar la mezcla de las muestras PCM utilizando el método básico de la presente invención puede dar lugar a errores, cuando se produce un error de lectura cuando se están leyendo los datos adicionales codificados en los bits menos significativos de las muestras PCM o incluso como parte de los bits más significativos de las muestras de PCM utilizadas para el audio. La naturaleza de este proceso de descifrado es tal que estos errores - relacionados con una muestra (de audio/datos) - afectará a la operación de separación de las muestras posteriores. Sin embargo, para optimizar el uso de la zona de datos auxiliares para datos adicionales en la secuencia PCM, en el que la codificación avanzada utilizará esta zona de datos auxiliares para almacenar los errores (por reducción de la frecuencia de muestreo), y que tiene todos estos datos de corrección comprimidos, se añadirá una suma de comprobación CRC (del inglés *Cyclic Redundancy Check*: Comprobación de redundancia cíclica) al final de un bloque de datos para al descodificador verificar la integridad de todos los datos de dicho bloque. Al almacenar valores semilla a intervalos regulares, se pueden limitar los efectos provocados por los errores en las muestras de audio. Cuando se produce un error, el error sólo se propagará hasta la siguiente posición para la que se conocen los valores semilla ya que en ese punto se puede reiniciar el proceso de descifrado, finalizando de manera efectiva la propagación de errores. Además, cuando se produce un error de datos en los valores semilla almacenados en la zona de datos auxiliares de los bits menos significativos, el descifrado basado en esos malos valores semilla será erróneo, pero sólo hasta la siguiente posición para la que se conocen los valores semilla ya que en ese momento el proceso de descifrado puede reiniciarse. Mediante el almacenamiento de datos adicionales en la zona de datos auxiliares en los bits menos significativos de las muestras, la presente invención, la mezcla o "*multiplexado*" de los datos de audio mezclados (los bits de mayor precisión) y los datos de codificación/descodificación (normalmente 2, 4 ó 6 bits por muestra) no necesitan ningún espacio de grabación adicional más allá de los (ya disponibles) 24 bits por muestra en el caso de Blu-ray DVD o HD-DVD, y tampoco necesitan ningún tipo de información extra de la "navegación" de los datos sobre el disco (por ejemplo, no se necesitan marcas de tiempo de un capítulo o una secuencia). Como tal, no se necesitan cambios en el control de la lectura del disco (tal como se implementa mediante el software integrado de los reproductores de DVD). Además no se necesitan cambios ni añadidos a la norma de estos formatos multimedia con el fin de utilizar esta invención. Además, la reducción de la resolución de bits de la muestra de audio y el almacenamiento de los datos de descodificación/codificación de audio en los bits menos significativos será de tal manera que los usuarios no detectarán ningún artefacto sonoro durante la reproducción normal con un dispositivo o sistema (por ejemplo, reproductores de HD-DVD o BLU-Ray DVD) que no implemente los algoritmos de descodificación. En una realización adicional del método se integra un patrón de sincronización en una posición definida con respecto a una ubicación de la primera muestra semilla. Se integra un patrón de sincronización para permitir la recuperación de la primera muestra semilla porque cuando se detecta el patrón de sincronización se conoce la ubicación de la primera muestra semilla. Esto también se puede aplicar para localizar la segunda muestra semilla. El patrón de sincronización se puede mejorar aún más mediante la repetición del patrón de sincronización a intervalos regulares de manera que puede emplearse una detección de circuito compensador para detectar de manera fiable el patrón de sincronización. Esto divide el almacenamiento de los datos en los bits menos significativos en bloques, lo que permite aplicar el procesamiento bloque a bloque. En una realización adicional del método antes de la etapa de igualación de muestras, se aproxima un error, que es el resultado de la ecuación de la muestra, mediante la selección de una aproximación de error de un conjunto de aproximaciones de error. La etapa de igualar las muestras

es muy fácil de ejecutar durante la combinación del primer y segundo conjunto de datos digitales, pero también introduce un error. Con el fin de reducir este error se establece un valor de error que se selecciona de un conjunto limitado de aproximaciones de error para elegir. Este conjunto limitado de aproximaciones de error permite la reducción del error y, al mismo tiempo, se ahorra espacio ya que las aproximaciones de error sólo pueden seleccionarse de un conjunto limitado que se puede representar con menos bits que el error real encontrado durante la etapa de igualación. Los índices de las aproximaciones de error necesitan menos bits por muestra que el número de bits liberados durante el proceso de codificación. Esto es importante para garantizar la compresibilidad de los datos. Este espacio ahorrado permite la integración de información adicional, tal como los patrones de sincronización y las muestras semilla. Una reducción de la frecuencia de muestreo desde 96 kHz a 48 kHz ó desde 192 kHz a 96 kHz puede ser un problema ya que las mayores frecuencias de muestreo se introdujeron con el objetivo de recrear audio y no sólo como frecuencia de muestreo como tal, pero sobre todo para una reproducción de audio de alta fidelidad se requiere mucho más detalle de información de fase en comparación con grabaciones de audio en discos compactos. Los errores debidos a la reducción de la frecuencia de muestreo y los datos de corrección (aproximaciones de error) para eliminar estos errores (en la medida de lo posible) pueden ser el resultado de un algoritmo de optimización, en el que los criterios de optimización se pueden definir como una suma mínima de errores al cuadrado o incluso pueden incluir criterios basados en objetivos de percepción de audio.

En una realización adicional del método, después de que se ha establecido la aproximación de error para una muestra, el valor de la muestra vecina a la que la muestra se debe igualar se modifica de tal manera que la muestra, cuando se reconstruye la muestra a partir de la muestra igualada que incluye la aproximación de error, representa más estrechamente la muestra antes de la igualación. El error puede reducirse aún más si es necesario mediante la modificación del valor de una muestra adyacente de modo que cuando la muestra se iguala a la muestra adyacente la combinación del valor adyacente y la aproximación de error representa con más precisión el valor original de la muestra antes de realizar la igualación con su vecino.

En otra realización del método el conjunto de aproximaciones de error se indexa y un índice que representa a la aproximación de error se integra en las muestras a las que corresponde la aproximación de error. En otra realización del método las muestras se dividen en bloques y el índice se integra en las muestras en un primer bloque que precede a un segundo bloque que comprende las muestras a las que corresponde el índice. Se consigue una nueva reducción en el tamaño de la aproximación de error mediante la indexación de un conjunto limitado de aproximación de error y almacenando solamente el índice adecuado en los bits menos significativos de las muestras del tercer conjunto de datos digitales que precede a las muestras a las que corresponden. Al integrar el índice en las muestras de un bloque anterior el índice y por lo tanto las aproximaciones de error están disponibles cuando empieza el proceso de descifrado de las muestras correspondientes.

En otra realización del método las aproximaciones de error integradas se comprimen. Además de la indexación, se pueden emplear otros métodos de compresión tales como Lempel Ziff. Las aproximaciones de error provienen de un conjunto limitado de aproximaciones de error y por lo tanto pueden comprimirse, lo que permite el uso de menos espacio cuando se integran las aproximaciones de error en las muestras. Esto es especialmente beneficioso si otros datos integrados también están presentes en los bits menos significativos de las muestras. No hay necesariamente disponible una indexación para estos datos adicionales y se puede utilizar un esquema de compresión general. Pueden utilizarse combinaciones de indexación para la aproximación de error y compresión de los datos adicionales o puede utilizarse una compresión global de todos los datos integrados en los bits menos significativos, es decir aproximaciones de error y datos adicionales.

En otra realización del método los valores de error se integran con un desplazamiento predefinido. Un desplazamiento predefinido establece una relación definida entre las aproximaciones de error y las muestras a las que corresponden las aproximaciones de error. En el caso de que se utilice un índice para almacenar las aproximaciones de error, el índice se adapta para cada bloque y el índice adaptado se almacena también en cada bloque. Si es posible, el índice también puede elegirse por cada conjunto de datos digitales o ser fijado y almacenado en el codificador y descodificador, pero no se almacena en la secuencia de datos, a costa de flexibilidad. Cuando no se utilizan aproximaciones de error para mejorar la calidad de las señales de audio extraídas, no es necesario almacenar las aproximaciones de error. Esto no impide la integración y la compresión de otros datos en los bits menos significativos del conjunto de datos digitales.

En otra realización del método los valores de error se integran en una primera posición disponible con una posición variable con respecto a las muestras a las que corresponden los valores de error. Al comprimir los valores de error en las muestras tan pronto como haya espacio disponible se ahorra espacio de las muestras, dicho espacio puede utilizarse para permitir una expansión del conjunto limitado de valores de error más adelante, permitiendo a su vez una corrección más precisa de la muestras igualadas, lo que da como resultado una reproducción incluso mejor del conjunto de datos digitales.

Este podría haber sido un método para aprovechar el espacio ganado, pero preferiblemente se elige un enfoque diferente. El espacio ahorrado a partir de los valores de error comprimidos y la lista de índices se utiliza realmente para limitar el número de muestras del siguiente bloque con el que se mezclará. Dado que este número es menor que el bloque actual, la variedad de los errores será menor y por lo tanto se puede aproximar mejor con el mismo

número de valores de aproximación de error. Estos valores de error y los índices de referencia se comprimen de nuevo y el espacio ahorrado se emplea para limitar el número de muestras mezcladas en el bloque siguiente.

En otra realización del método cualquiera de los bits menos significativos de las muestras del tercer conjunto de datos digitales que no sea utilizado para integrar aproximaciones de error, u otros datos de control, se establece en un valor predefinido o pone a cero. Cualquiera de los bits menos significativos se puede poner a cero antes de la combinación de los conjuntos de datos digitales o después de la integración de la información integrada, tal como valores semilla, patrones de sincronización y valores de error. El valor predefinido o el valor cero pueden ayudar a distinguir los datos integrados ya que los datos integrados ya no estarán rodeados de datos aparentemente aleatorios. Además, permite la simplificación del proceso de combinación y descifrado ya que quedaría claro que estos bits no necesitan procesamiento. Cabe señalar que la selección del número de bits liberados en los bits menos significativos puede implementarse de forma dinámica, en otras palabras, basándose en el contenido de los conjuntos de datos digitales en ese momento. Por ejemplo, las partes en silencio de la música clásica pueden necesitar más bits de resolución de señal, mientras que las partes de volumen alto de la música pop podrían no necesitar tantos bits.

En una realización de la invención, la señal extraída o los datos de control integrados se pueden utilizar para controlar dispositivos externos que deben ser controlados de forma síncrona con la señal de audio, o controlar la reproducción de una señal de audio extraída, por ejemplo mediante la definición de la amplitud de la señal de audio extraída en relación con un nivel básico o en relación con los otros canales de audio no extraídos de la señal combinada, o en relación con la señal de audio combinada.

La presente invención describe una técnica para mezclar (y almacenar) pistas de audio PCM (las pistas PCM son conjuntos de datos digitales que representan canales de audio digital) - normalmente a partir de una grabación de audio de 3 dimensiones, pero no se limita a este uso - en un número de pistas que es menor que el número de pistas utilizadas en la grabación original. Esta combinación de canales se realiza mediante la mezcla de parejas de pistas de audio en pistas individuales, de una manera que soporten una operación inversa, es decir, una operación de descodificación que permita un descifrado de la señal combinada, para recrear las pistas de audio originales y separadas que serán de percepción idéntica a las pistas de audio originales de la grabación original, y al mismo tiempo la señal combinada proporciona una pista de audio que se puede reproducir mediante canales de reproducción normales y es de idéntica percepción a una mezcla de los canales de audio cuando se reproducen. Como tal, cuando se combinan los canales de una grabación de audio de 3 dimensiones en un conjunto de canales utilizados normalmente para una grabación de audio envolvente de 2 dimensiones y se reproducen los canales combinados sin aplicar la operación inversa, la grabación combinada, es decir mezclada con reducción (*downmixed*), sigue cumpliendo con los requisitos para recrear una grabación de audio envolvente realista de 2 dimensiones normalmente conocida como formatos de audio envolvente estéreo, 4.0, 5.1 o incluso 7.1, y que puede reproducirse como tal, sin la necesidad de un dispositivo adicional, un dispositivo modificado o un descodificador. Esto garantiza la compatibilidad de los canales combinados resultantes con equipos antiguos.

Es muy factible la ampliación a más de 2 conjuntos de datos digitales o dos señales de audio. La técnica se explica para 2 conjuntos de datos digitales, la ampliación de esta técnica a más de 2 conjuntos se puede hacer de una manera similar mediante el cambio de la intercalación de modo que para cada muestra del tercer conjunto de datos digitales solo un conjunto de datos digitales proporciona una muestra no igualada que se va a combinar con muestras igualadas de los otros conjuntos de datos digitales y que el conjunto de datos digitales que proporciona la muestra no igualada se elige de una manera alternante a partir de los conjuntos de datos digitales que proporcionan las muestras. Si se combinan más de 2 conjuntos de datos digitales, cada muestra  $n$  de cada conjunto de datos digitales se utiliza como muestras de igualación del primer subconjunto que contiene  $(n-1)$  por  $n$  muestras (iguales) del conjunto de datos, mientras que el segundo subconjunto contiene 1 muestra por  $n$  muestras del conjunto de datos. Por cada conjunto de datos, la posición de las muestras de igualación se traslada 1 posición en el dominio temporal.

Como tal se ha encontrado que las mezclas de audio digital de 3 canales a audio digital de 1 canal (mezcla 3 a 1) son ciertamente factibles dentro de la tasa de datos y la resolución proporcionadas por las actuales normas de audio digital. Las mezclas 4 a 1 también son posibles de esta manera.

Tales mezclas de canales de audio digital permiten el uso de un primer estándar de audio digital con un primer número de canales independientes de audio digital para el almacenamiento, la transmisión y la reproducción de un segundo estándar de audio digital con un segundo número de canales independientes de audio digital, en el que el segundo número de canales de audio digital es mayor que el primer número de canales digitales de audio. La invención consigue esto mediante la combinación de por lo menos dos canales de audio digital en un solo canal de audio digital utilizando el método de la invención o un codificador de acuerdo con la invención. Debido a la etapa de adición en el método la secuencia de audio digital resultante es una representación perceptivamente agradable de los dos canales de audio digital combinados. Al realizar esta combinación de múltiples canales se reduce el número de canales, por ejemplo desde una configuración 3D 9.1 a una configuración 2D 5.1. Esto puede conseguirse, por ejemplo, combinando el canal frontal inferior izquierdo y el canal frontal superior izquierdo del sistema 9.1 en un solo canal frontal izquierdo, que normalmente se puede almacenar, transmitir y reproducir a través del canal frontal izquierdo de un sistema 5.1. Por lo tanto, aunque las señales creadas utilizando la invención permiten la

recuperación de los canales originales 9.1 mediante descifrado de las señales combinadas, las señales combinadas son igualmente adecuadas para su uso por los usuarios que sólo tienen un sistema 5.1. Puede ser necesaria la atenuación de los dos canales antes de la mezcla o la codificación para un adecuado sistema mezclado con reducción (*downmixed*) a 5.1, de tal manera que durante la descodificación se necesitan datos de atenuación (inversa) de cada canal.

Las técnicas desarrolladas en esta invención se utilizan - pero no se limitan a este uso - para la creación de grabaciones de audio AURO-fónico que pueden almacenarse en soportes multimedia ya existentes o nuevos, como HD-DVD o DVD BLU-RAY, que se proporcionan solo como ejemplo, sin la necesidad de añadir ningún formato multimedia extra o añadidos a sus definiciones de formato multimedia, ya que estos estándares ya soportan datos PCM con varios canales, por ejemplo, 6 canales de audio PCM a 96 kHz 24 bits (HD-DVD) u 8 canales de audio PCM a 96 kHz 24 bits (Blu-ray DVD) o 6 canales de audio PCM a 192 kHz 24 bits (DVD Blu-ray). Para las grabaciones de audio AURO-fónico se necesitan más canales que los disponibles en estos soportes multimedia existentes o en otros nuevos. La presente invención permite el uso de estos soportes multimedia, u otros medios de transmisión en los que existe una falta de canales y permite el uso de un sistema así con un número inadecuado de canales para el almacenamiento o transmisión de audio 3D, y al mismo tiempo garantiza la compatibilidad con todos los equipos de reproducción existentes, al proporcionar de forma automática los canales de audio 3D en un sistema 2D como si fueran canales de audio 2D. Si hay un equipo de reproducción adaptado, puede proporcionarse el conjunto completo de canales de audio 3D utilizando el método de descodificación o el descodificador de acuerdo con la invención y puede proporcionar el audio 3D completo adecuadamente mediante el sistema después de extraer los canales de audio digital separados y reproducir estos canales individuales.

Aurofonía designa un sistema de reproducción de audio (o audio + vídeo) capaz de entregar correctamente las tres dimensiones de la sala de grabación - definida por sus ejes x, y, z -. Una grabación de sonido adecuada combinada con una distribución específica de los altavoces se ha demostrado que produce un sonido más natural.

Una grabación de audio 3D como Aurofonía también se puede definir como una configuración envolvente con altavoces en altura. Es esta adición de los altavoces en altura la que introduce la necesidad de más canales que los que pueden proporcionar los sistemas comúnmente utilizados en la actualidad ya que los sistemas 2D utilizados actualmente sólo permiten altavoces substancialmente al mismo nivel en una habitación. Cómo la Aurofonía junta y mezcla las características tonales de dos espacios está vinculado a determinados aspectos de la conciencia. El aumento del número de canales y la colocación de los altavoces, permite que cualquier grabación realizada sobre esta base permita una reproducción que utiliza todo el potencial de los aspectos naturales tridimensionales del audio. La tecnología de varios canales combinada con la colocación específica de los altavoces transporta acústicamente a los oyentes el mismo lugar del evento sonoro - a un espacio virtual - y les permite experimentar sus dimensiones espaciales de modo virtual. La anchura, profundidad y altura de este espacio son por primera vez percibidas tanto física como emocionalmente.

Además, los dispositivos como los reproductores de HD-DVD o de Blu-ray DVD implementan un mezclador de audio para mezclar los canales de audio externo durante la reproducción (no leído del disco) en la salida de audio, o mezclar efectos de audio generalmente de la operación de navegación del usuario para aumentar la experiencia del usuario. Sin embargo, también tienen modo de "película" real que elimina estos efectos de audio durante la reproducción. Este último modo es utilizado por estos reproductores a la salida de la mezcla PCM de varios canales a través de sus convertidores de audio (A/D) o para proporcionar la mezcla PCM de varios canales cifrada como una mezcla de audio de varios canales encapsulada en los datos que incluyen por ejemplo vídeo y enviarla fuera utilizando una interfaz HDMI (del inglés *High-Definition Multimedia Interface*, interfaz multimedia de alta definición) para su posterior procesamiento. El requisito de compresión sin pérdida, por ejemplo los datos de audio PCM de bits idénticos, utilizado durante la reproducción/grabación se mantiene válido para cualquier dispositivo de que proporciona o graba estas pistas de audio PCM de varios canales mezcladas (*downmixed*) cuando el descodificador - tal como se explica en esta invención - se utiliza para recrear la grabación de audio de 3 dimensiones o simplemente una grabación de audio "espacial" mejorada.

Además del almacenamiento más eficaz o eficiente de audio PCM mediante la combinación, de una manera reversible, de múltiples canales en un solo canal, un objetivo de aplicación o uso es la de una grabación y reproducción de audio de 3 dimensiones, que todavía mantenga la compatibilidad con formatos de audio según se proporcionan mediante los estándares de DVD, HD-DVD ó DVD Blu-Ray. Durante el la producción final (*mastering*) de grabación de audio envolvente o de audio de varios canales, los ingenieros de grabación actualmente tienen varias pistas de audio disponibles y utilizan plantillas para que sus herramientas de producción final creen una pista de audio estéreo o envolvente (2 Dimensiones), que puede crearse por ejemplo en un CD, SA-CD, DVD, DVD Blu-Ray o HD-DVD o simplemente ser almacenada digitalmente en un dispositivo de grabación (como por ejemplo una unidad de disco duro). Las fuentes de audio, que en el mundo real siempre se encuentran en un espacio de 3 dimensiones, hasta ahora se han grabado en su mayoría como fuentes definidas en un espacio de 2 dimensiones, a pesar de que para los ingenieros de grabación de audio estaba disponible la información tridimensional, o se podía haber añadido fácilmente (por ejemplo, efectos de sonido como aviones que vuelan sobre un público, o pájaros 'cantando' en el cielo) o ser grabada de una situación de la vida real. Hasta ahora no había disponible ningún formato de audio general, a excepción de los sistemas en los que las series adicionales de varias pistas de audio se almacenan de forma independiente en un sistema que proporciona un número suficiente de pistas para el

almacenamiento tal como en aplicaciones del cine. Estos canales adicionales sin embargo no se pueden almacenar en medios de grabación como HD-DVD o DVD Blu-Ray, ya que estos sistemas de almacenamiento proporcionan un número insuficiente de canales de audio. El objetivo de esta invención es crear estas pistas "virtuales" extra de una manera que no interfiera (o moleste) con la información estándar de audio en 2 canales o varios canales (2D), de una manera que para los ingenieros de grabación la evaluación básica en tiempo real esté disponible antes de finalizar la grabación de audio 3D y de una manera para incluso utilizar no más de las pistas "estándar" de varios canales en estos nuevos medios.

Cabe señalar que, si bien la presente invención se describe como que su objetivo son las aplicaciones de audio, se puede prever que los mismos principios se emplean en aplicaciones de vídeo, por ejemplo para crear una reproducción de vídeo en 3 dimensiones, por ejemplo, mediante el uso de 2 secuencias (ángulos) de vídeo simultáneas, cada una tomada desde una cámara con una pequeña diferencia angular, para crear un efecto 3-D, incluso combinar las dos secuencias de vídeo tal como se detalla en la presente invención y permitir de este modo el almacenamiento y la transmisión del vídeo en 3D de tal manera que todavía pueda reproducirse en un equipo de vídeo normal.

### 15 Ejemplos de Aplicaciones

Mezcla estéreo ("artística") incluida en Mezcla de Sonido Envoltente.

Durante la producción final de las grabaciones de audio, los ingenieros de sonido definen o utilizan plantillas de mezcla para, a partir de varias pistas de audio, crear una mezcla estéreo "verdadera" o "artística", así como una mezcla de sonido envolvente (por ejemplo, 4.0, 5.1,...). Aunque es posible la mezcla (*downmix*) por matriz de la mezcla envolvente a una mezcla estéreo, se pueden ilustrar fácilmente las deficiencias de tales técnicas de matrices para mezcla (*down-mix*). El estéreo mezclado (*downmixed*) por matriz diferirá sustancialmente de la mezcla estéreo "artística", ya que el contenido de tales señales estéreo mezcladas (*downmixed*) por matriz estará típicamente en el dominio I-D (señales fuera de fase), mientras que la mezcla estéreo "artística" verdadera estará principalmente en el dominio I+D (señales en fase) con una cantidad moderada en el dominio de I-R. Sólo como un ejemplo; el estéreo mezclado (*downmixed*) por matriz sonará mucho más tranquilo en mono debido a la gran cantidad de señales fuera de fase. Como consecuencia, las actuales grabaciones de audio envolvente producidas finalmente y codificadas con la mayor parte de la tecnología actual de codificación/descodificación de audio suelen ofrecer - si se tiene cuidado de una reproducción estéreo realista - una versión estéreo separada ("artística") verdadera de la grabación.

Con una aplicación basada en las técnicas de la invención actual, aquellos familiarizados con esta técnica, podrán construir fácilmente un sistema que produzca finalmente los canales de Audio Izquierdo (frontal) y Derecho (frontal) de la grabación artística en los canales Izquierdo y Derecho, y que cada uno de estos canales se mezcle con un (por ejemplo) Canal Delta de Audio atenuado 24 dB (I-artístico - I-envolvente) y (D-artístico - D-envolvente). Durante la reproducción de los canales I/D de una grabación de varios canales sin descodificador, la grabación de audio artística Izquierda/Derecha estará presente predominantemente, pero cuando se reproduce con un descodificador como se explica en esta invención, los canales mezclados se separarán en primer lugar, a continuación los canales (delta) se amplificarán (por ejemplo) 24 dB y se restarán de los canales "artísticos", para crear los canales Izquierdo y Derecho, según sea necesario para la mezcla envolvente, en ese momento también se reproducen los canales envolventes (I/D), así como los canales Centro y Altavoz de Graves (Subwoofer).

Mezcla de 3-Dimensiones ("AURO-fónica") incluida en la Mezcla Envoltente.

Utilizando la técnica de codificación tal como se explica en esta invención, se puede ver fácilmente que la mezcla de información de audio de 3 dimensiones se puede hacer simplemente mediante la mezcla en cada canal de una mezcla envolvente de 2 dimensiones 2.0, 4.0, 5.1 ó incluso 7.1 de otro canal de audio que represente el audio tal como se graba a una cierta altura por encima de esos altavoces de 2 dimensiones. Durante la mezcla, estos canales de audio de 3 dimensiones pueden ser atenuados, para evitar efectos de audio no deseados, cuando no se utiliza la grabación de varios canales con dicho descodificador tal como se define en esta invención. Durante la descodificación estos canales son separados y amplificados cuando sea necesario y son reproducidos por los altavoces principales.

Mezcla estéreo ("artística") y Mezcla 3-D ("AURO-fónica") incluida en la Mezcla Envoltente.

Si se busca generar una grabación todo en uno, por ejemplo, 6 canales a 96 kHz (HD-DVD) ó 192 kHz (Blu-ray DVD), útil para una reproducción estéreo artística, reproducción envolvente 2-D o reproducción AURO-fónica 3-D, se puede utilizar una aplicación basada en la invención. La invención puede utilizarse para mezclar 3 canales (o más) en un canal, mediante la reducción de la frecuencia de muestreo "inicial" por factor 3 (o más), y aproximar los errores generados durante esta reducción, para restaurar la señal original tanto como sea posible. Esto podría utilizarse para mezclar un canal a 96 kHz Frontal-Artístico Izquierdo, con un Delta Frontal Izquierdo (atenuado) a 96 kHz (I-artístico - I-envolvente), y con uno Superior Frontal Izquierdo (atenuado) de 96 kHz. Se puede aplicar un esquema de mezcla similar al canal Frontal Derecho. Se puede aplicar mezcla de 2 canales para el Envoltente Izquierdo y para el Envoltente Derecho. Incluso el canal Central se puede utilizar para mezclar un canal de audio Central Superior.



Producción de audio 3-D automatizada a partir de grabación "clásica" en 2-D.

La mayoría de las actuales producciones de audio o de vídeo existentes tienen pistas de audio de 2 dimensiones (envolvente). Aparte de la verdadera ubicación de la fuente de audio de 3 dimensiones - que se puede utilizar durante la producción final y la mezcla con un codificador tal como se explica en esta invención para utilizar esa información como canales adicionales mezclados (*down-mixed*) en una grabación de 2 dimensiones - el audio difuso tal como se presenta en las grabaciones de audio estándar de 2 dimensiones es el candidato para ser movido y traducido por los altavoces superiores de la configuración de audio de 3 dimensiones. Se puede pensar en procesos de audio automáticos (fuera de línea - o en tiempo no real), que extraerán el audio difuso de las grabaciones de 2 dimensiones, y se puede utilizar ese audio extraído para crear los canales que se mezclan (de acuerdo con el esquema de esta invención) con las pistas de audio "reducidas" de las grabaciones envolventes 2-D, de tal manera que se obtiene una grabación envolvente de varios canales que puede descodificarse como audio 3D. Dependiendo de los requisitos de cómputo, esta técnica de filtrado para extraer el audio difuso de los canales envolventes 2D se podría aplicar en tiempo real.

La invención se puede utilizar para varios dispositivos, que formen parte de un sistema de audio de 3 dimensiones.

Un Codificador Aurofónico - Programa añadido (*plug-in*) (software) de Aplicación Informática. Las herramientas de producción final y mezcla, comúnmente disponibles en el mundo de la grabación y producción final de audio/vídeo, permiten a terceros desarrollar programas añadidos de software. Por lo general ofrecen una interfaz común de datos/órdenes para activar los programas añadidos dentro de un conjunto completo de herramientas utilizadas por los ingenieros de mezcla y producción final. Como el núcleo del Codificador AUROFÓNICO es un simple ejemplo de Codificador, con varias entradas de canales de audio y una salida de canal de audio por un lado, y teniendo en cuenta la configuración del usuario como ubicación/atenuación de canal y calidad como parámetros por otra parte, se puede proporcionar un programa añadido en esas herramientas de producción final/mezcla de audio. Un Descodificador AUROFÓNICO. - Programa pequeño (software) de Aplicación Informática. Se puede desarrollar un descodificador de programa añadido de software como herramienta de verificación con las herramientas de producción final y mezcla, de forma similar al programa añadido de Codificador. Este descodificador de programa añadido de software también se puede integrar en los reproductores multimedia de los PC de los consumidores/usuarios finales (como Windows Media Player, o reproductores por software de DVD y más probablemente en reproductores por software de HD-DVD/Blu-Ray).

Un descodificador AUROFÓNICO - ASIC/DSP (del inglés *Application-Specific Integrated Circuit*: Circuito Integrado para Aplicaciones Específicas y *Digital Signal Processor*: procesador digital de señal) exclusivos incorporados en un reproductor de Blu-Ray o HD-DVD. Varios nuevos formatos multimedia de alta definición definen varias secuencias de audio PCM de alta frecuencia/alta resolución de bits que están disponibles (digitalmente) dentro de sus respectivos reproductores (del consumidor). Cuando se reproduce el contenido de estos discos, usando un modo con el que no se mezclan/juntan/atenuan/... los datos de audio PCM que se presentarán a los convertidores internos de audio digital analógico, estos datos de audio PCM (podrían ser datos codificados con AURO) puede ser interceptados por un ASIC o DSP exclusivo (cargado con el programa fijo del fabricante "*firmware*" del descodificador AURO) para descodificar todos los canales de audio mezclados y para generar un conjunto extra de salidas de audio para entregar, por ejemplo, audio artístico Izquierda/Derecha o, por ejemplo: un conjunto adicional de salidas Superiores I/D.

Un descodificador AUROFÓNICO - integrado como parte del programa fijo del fabricante del Blu-ray o HD-DVD. Siempre que tenga sentido un proceso de descodificación AUROFÓNICO durante la reproducción de un Blu-Ray o HD-DVD, el modo de reproducción de estos reproductores tiene que establecerse en el modo de Película-VERDADERA, para evitar que el mezclador de audio del reproductor estropee o modifique los datos originales de las secuencias PCM según la producción final en este disco. En este modo no se necesita toda la potencia de procesamiento del DSP o la CPU de los reproductores. Como tal, puede ser posible integrar el descodificador AUROFÓNICO como un proceso adicional de separación implementado como parte del software fijo del fabricante del DSP o la CPU del reproductor. Un descodificador AUROFÓNICO - accesorio ASIC/DSP en los dispositivos de audio con conmutadores de HDMI, USB o FireWire. El HDMI permite la transferencia de todo el ancho de banda de secuencias de audio de varios canales. (8 canales, 192 kHz, 24 bits). Los conmutadores HDMI regeneran los datos digitales de Audio/Vídeo, mediante descifrado en primer lugar, de tal manera que los datos de audio transmitidos a través de una interfaz HDMI son accesibles internamente en dicho conmutador. El audio codificado con AURO puede ser descodificado mediante un accesorio a bordo que implemente el descodificador AURO. Se puede utilizar una integración similar de accesorios (por lo general en las herramientas de grabación/reproducción de audio) para dispositivos de E/S de audio de varios canales con USB o FIREWIRE. Un codificador tal como el descrito en esta memoria descriptiva puede integrarse en un dispositivo más grande, tal como un sistema de grabación o puede ser un codificador independiente acoplado a un sistema de grabación o un sistema de mezcla. El codificador también puede implementarse como un programa informático, por ejemplo, para realizar los métodos de codificación de la presente invención cuando se ejecuta en un sistema informático adecuado para ejecutar dicho programa informático. Un descodificador tal como el descrito en esta memoria descriptiva puede integrarse en un dispositivo más grande, tal como un módulo de salida en un dispositivo de reproducción, un módulo de entrada en un dispositivo de amplificación o puede ser un descodificador independiente acoplado mediante su entrada a una fuente de la secuencia de datos codificados y combinados acoplado mediante su salida a un amplificador.

Un dispositivo de procesamiento de señal digital se entiende esta memoria que es un dispositivo en la sección de grabación de la cadena de grabación, transmisión y reproducción, tal como una mesa de mezcla de audio, un dispositivo de grabación para grabar en un soporte de grabación tal como discos ópticos o discos duros, un dispositivo de procesamiento de señales o un dispositivo de captura de señales.

- 5 Un dispositivo de reproducción en esta memoria descriptiva se entiende que es un dispositivo en la sección de reproducción de la cadena de grabación, transmisión y reproducción, tal como un amplificador de audio o un dispositivo de reproducción para recuperar datos de un medio de almacenamiento.

El descodificador o dispositivo de reproducción puede integrarse ventajosamente en un vehículo tal como un coche o un autobús. En un vehículo los pasajeros están normalmente rodeados por un habitáculo de pasajeros.

- 10 El habitáculo permite la fácil colocación de los altavoces mediante los que se va a reproducir el audio de varios canales. Por lo tanto un diseñador es capaz de adaptar específicamente el ambiente de audio para adaptar la reproducción de audio de varios canales en 3 dimensiones u otras en el interior del habitáculo de pasajeros. Otra ventaja es que el cableado necesario para los altavoces puede esconderse fácilmente de la vista, al igual que el otro cableado está oculto de la vista. El conjunto inferior de altavoces del sistema de altavoces de 3 dimensiones se coloca en la parte inferior del habitáculo de pasajeros, al igual que muchos altavoces se montan actualmente, por ejemplo en el panel de la puerta, en el salpicadero o cerca del piso. El conjunto superior de los altavoces del sistema de altavoces de 3 dimensiones se puede colocar en la parte superior del habitáculo de pasajeros, por ejemplo cerca del techo o en otra posición más alta que el panel de instrumentos o el salpicadero o por lo menos más alto que el conjunto inferior de altavoces. También es beneficioso permitir al usuario cambiar el dispositivo de reproducción desde un primer estado en el que el descodificador descifra canales de audio y pasa los canales de audio descifrados al amplificador a un segundo estado en el que los canales de audio combinados se pasan al amplificador. Se puede conseguir una conmutación entre la reproducción de 3 dimensiones y la reproducción de 2 dimensiones sin pasar por el descodificador. En otra configuración también se prevé una conmutación entre la reproducción de 2 dimensiones y la reproducción estéreo. Los requisitos para la reproducción de audio de 2 y 3 dimensiones, tal como la colocación de los altavoces, no forman parte de esta invención y como tal no se describirán con detalle. Sin embargo, debe tenerse en cuenta que la invención es adaptable a cualquier configuración de canales que un diseñador de un dispositivo de reproducción de varios canales de audio pueda elegir, por ejemplo, al configurar un coche para la reproducción adecuada de audio de varios canales.

**Breve descripción de las figuras**

- 30 La invención se describirá basándose en figuras.
- La figura 1 muestra un codificador de acuerdo con la invención para la combinación de dos canales.
- La figura 2 muestra un primer conjunto de datos digitales que se convierten mediante la igualación de muestras
- La figura 3 muestra un segundo conjunto de datos digitales que se convierten mediante la igualación de muestras
- 35 La figura 4 muestra la codificación de los dos conjuntos de datos digitales resultantes en un tercer conjunto de datos digitales.
- La figura 5 muestra la descodificación del tercer conjunto de datos digitales de nuevo en dos conjuntos separados de datos digitales.
- La figura 6 muestra una conversión mejorada del primer conjunto de datos digitales.
- La figura 7 muestra una conversión mejorada del segundo conjunto de datos digitales.
- 40 La figura 8 muestra la codificación de los dos conjuntos de datos digitales resultantes en un tercer conjunto de datos digitales.
- La figura 9 muestra la descodificación del tercer conjunto de datos digitales de nuevo en dos conjuntos separados de datos digitales.
- 45 La figura 10 muestra un ejemplo en el que se representan las muestras de la primera secuencia A tal como se obtienen mediante la codificación que se describe en la figura 6.
- La figura 11 muestra un ejemplo en el que se representan las muestras de la primera secuencia B tal como se obtienen mediante la codificación que se describe en la figura 7.
- La figura 12 muestra las muestras de la secuencia mezclada C.
- La figura 13 muestra los errores introducidos en la secuencia PCM por la invención.

La figura 14 muestra el formato de la zona de datos auxiliares en los bits menos significativos de las muestras del conjunto de datos digitales combinados.

La figura 15 muestra más detalles de la zona de datos auxiliares.

La figura 16 muestra una situación en la que la adaptación conduce a bloque de datos AURO de longitud variable.

5 La figura 17 proporciona una visión general de una combinación de las etapas de proceso tal como se explica en las secciones anteriores.

La figura 18 muestra un dispositivo Codificador Aurofónico

La figura 19 muestra un dispositivo Descodificador Aurofónico

### Descripción de realizaciones

10 La figura 1 muestra un codificador de acuerdo con la invención para la combinación de dos canales. El codificador 10 comprende una primera unidad de igualación 11a y una segunda unidad de igualación 11b. Cada unidad de igualación 11a, 11b, recibe un conjunto de datos digitales desde una entrada respectiva del codificador 10. La primera unidad de igualación 11a selecciona un primer subconjunto de muestras del primer conjunto de datos digitales y iguala cada muestra de este primer subconjunto con muestras vecinas de un segundo subconjunto de muestras del primer conjunto de datos digitales en el que el primer subconjunto de muestras y el segundo subconjunto de muestras se intercalan como se explicará con detalle en la figura 2. El conjunto de datos digitales resultante que comprende las muestras no afectadas del segundo subconjunto y las muestras igualadas del primer subconjunto se puede pasar a un primer reductor 12a de tamaño de muestra opcional o se puede pasar directamente al elemento de combinación 13. La segunda unidad de igualación 11b selecciona un tercer subconjunto de muestras del segundo conjunto de datos digitales y iguala cada muestra de este tercer subconjunto con muestras vecinas de un cuarto subconjunto de muestras del segundo conjunto de datos digitales en el que el tercer subconjunto de muestras y el cuarto subconjunto de muestras se intercalan como se explicará con detalle en la figura 3. El conjunto de datos digitales resultante que comprende las muestras del cuarto subconjunto y las muestras igualadas del tercer subconjunto se puede pasar a un segundo reductor 12b de tamaño de muestra opcional o se puede pasar directamente al elemento de combinación 13. El primer y segundo reductor de tamaño de muestra eliminan un número determinado de bits menores de las muestras de sus respectivos conjuntos de datos digitales, por ejemplo reduciendo las muestras de 24 bits a 20 bits mediante la eliminación de los cuatro bits menos significativos. La igualación de las muestras como la realizada por las unidades de igualación 11a, 11b introduce un error. Opcionalmente, este error se aproxima con el dispositivo de aproximación 15 de errores mediante la comparación de las muestras igualadas con las muestras originales. Esta aproximación de error puede utilizarse por el descodificador para restaurar con mayor precisión los conjuntos originales de datos digitales, como se explica a continuación. El elemento de combinación 13 añade las muestras del primer conjunto de datos digitales a las muestras correspondientes del segundo conjunto de datos digitales, tal como se proporcionan a sus entradas, y suministra las muestras resultantes del tercer conjunto de datos de digitales mediante su salida a un elemento para dar formato 14, que integra datos adicionales tales como valores semilla de los dos conjuntos de datos digitales y las aproximaciones de error tal como se reciben del elemento 15 de aproximación de errores en los bits menos significativos del tercer conjunto de datos digitales y proporciona el conjunto de datos digitales resultante a una salida del codificador 10.

40 Con el fin de explicar el principio, las realizaciones se explican utilizando dos secuencias de entrada, pero la invención también puede utilizarse con tres o más secuencias de entrada que se combinan en una sola secuencia de salida.

La figura 2 muestra un primer conjunto de datos digitales que se convierten mediante la igualación de muestras El primer conjunto 20 de datos digitales comprende una secuencia de valores de muestra  $A_0, A_1, A_2, A_3, A_4, A_5, A_6, A_7, A_8, A_9$ . El primer conjunto de datos digitales se divide en un primer subconjunto de muestras  $A_1, A_3, A_5, A_7, A_9$  y un segundo subconjunto de muestras  $A_0, A_2, A_4, A_6, A_8$ .

Posteriormente cada uno de los valores de cada muestra  $A_1, A_3, A_5, A_7, A_9$  del primer subconjunto de muestras se iguala con el valor de la muestra vecina  $A_0, A_2, A_4, A_6, A_8$  del segundo subconjunto según lo indicado por las flechas en la figura 2. En particular, esto significa que el valor de la muestra  $A_1$  se sustituye por el valor de la muestra vecina  $A_0$ , es decir, el valor de la muestra  $A_1$  se iguala con el valor de la muestra  $A_0$ . Esto da lugar a un primer conjunto intermedio 21 de datos digitales tal como se muestra, que comprende los valores de muestra  $A_0'', A_1'', A_2'', A_3'', A_4'', A_5'', A_6'', A_7'', A_8'', A_9''$ , etc., donde el valor  $A_0''$  es igual al valor  $A_0$  y  $A_1''$  es igual al valor  $A_0$  etc. En la figura 6 se muestra una realización en la que  $A_0''$  ya no es igual a  $A$  debido a una reducción del número de bits de la muestra.

La figura 3 muestra un segundo conjunto de datos digitales que se convierten mediante la igualación de muestras. El segundo conjunto 30 de datos digitales comprende una secuencia de valores de muestras  $B_0, B_1, B_2, B_3, B_4, B_5, B_6, B_7, B_8, B_9$ . El segundo conjunto de datos digitales se divide en un tercer subconjunto de muestras  $B_0, B_2, B_4, B_6, B_8$  y un cuarto subconjunto de muestras  $B_1, B_3, B_5, B_7, B_9$ . Posteriormente cada uno de los valores de cada muestra  $B_0, B_2, B_4, B_6, B_8$  del tercer subconjunto de muestras se iguala con el valor de la muestra vecina  $B_1, B_3, B_5, B_7, B_9$  del

cuarto subconjunto según lo indicado por las flechas de la figura 3. En particular, esto significa que el valor de la muestra  $B_2$  se sustituye por el valor de la muestra vecina  $B_1$ , es decir, el valor de la muestra  $B_2$  se iguala con el valor de la muestra  $B_1$ . Esto da lugar a un segundo conjunto intermedio 31 de datos digitales como se muestra, que comprende los valores de muestra  $B_0$ ,  $B_1$ ,  $B_2$ ,  $B_3$ ,  $B_4$ ,  $B_5$ ,  $B_6$ ,  $B_7$ ,  $B_8$ ,  $B_9$ , donde el valor  $B_1$  es igual al valor  $B_1$  y  $B_2$  es igual al valor  $B_1$ , etc. En la figura 7 se muestra una realización en la que  $B_1$  ya no es igual a  $B_1$  debido a una reducción del número de bits en la muestra.

La figura 4 muestra la codificación de los dos conjuntos de datos digitales resultantes en un tercer conjunto de datos digitales. El primer conjunto intermedio 21 de datos digitales y el segundo conjunto intermedio 31 de datos digitales se combinan ahora mediante la adición de las muestras correspondientes. Por ejemplo, la segunda muestra  $A_1$  del primer conjunto intermedio 21 de datos digitales se añade a la segunda muestra  $B_1$  del segundo conjunto intermedio 31 de datos digitales. La primera muestra combinada resultante  $C_1$  se coloca en la segunda posición del tercer conjunto 40 de datos digitales y tiene un valor de  $A_1 + B_1$ .

La tercera muestra  $A_2$  del primer conjunto intermedio 21 de datos digitales se añade a la tercera muestra  $B_2$  del segundo conjunto intermedio 31 de datos digitales. La segunda muestra combinada resultante  $C_2$  se coloca en la tercera posición del tercer conjunto 40 de datos digitales y tiene un valor de  $A_2 + B_2$ .

La figura 5 muestra la descodificación del tercer conjunto de datos digitales de nuevo en dos conjuntos separados de datos digitales. El tercer conjunto 40 de datos digitales se suministra a un descodificador para descifrar los dos conjuntos 31, 32 de datos digitales comprendidos en el tercer conjunto 40 de datos digitales. La primera posición del tercer conjunto 40 de datos digitales se muestra conteniendo el valor  $A_0$ , que es un valor semilla necesario durante la descodificación. Este valor semilla se puede almacenar en otro lugar, pero se muestra en la primera posición por conveniencia durante la explicación. La segunda posición contiene la primera muestra combinada con un valor de  $A_0 + B_0$ . Debido a que el descodificador conoce el valor semilla  $A_0$ , tal como se recupera de la primera posición, el valor de la muestra del segundo conjunto intermedio de datos digitales se pueden establecer restando  $C_0 - A_0 = (A_0 + B_0) - A_0 = B_0$ . Este valor  $B_0$  de muestra recuperado se utiliza para reconstruir el segundo conjunto intermedio de datos digitales, pero también se utiliza para recuperar una muestra del primer conjunto intermedio de datos digitales. Dado que el valor  $A_0$  es ahora conocido, y se sabe que su muestra vecina  $A_1$  tiene el mismo valor, ya se puede calcular la muestra del 2º conjunto intermedio de datos digitales:

$$C_1 - A_1 = (A_1 + B_1) - A_1 = B_1.$$

Este valor  $B_1$  de muestra recuperado se utiliza para reconstruir el segundo conjunto intermedio de datos digitales, pero también se utiliza para recuperar una muestra del primer conjunto intermedio de datos digitales.

Dado que el valor  $B_1$  es ahora conocido, y se sabe que su muestra vecina  $B_2$  tiene el mismo valor, ya se puede calcular la muestra del 1er conjunto intermedio de datos digitales:

$$C_2 - B_2 = (A_2 + B_2) - B_2 = A_2.$$

Este valor  $A_2$  de muestra recuperado se utiliza para reconstruir el primer conjunto intermedio de datos digitales, pero también se utiliza para recuperar una muestra del 2º conjunto intermedio de datos digitales.

Esto se puede repetir como se muestra en la figura 5 para las muestras restantes.

Con el fin de aproximar el primer conjunto 20 de datos digitales el primer conjunto intermedio recuperado de datos digitales puede procesarse utilizando la información acerca de la señal conocida por sistema, por ejemplo para una señal de audio las muestras perdidas por la codificación y descodificación (las muestras igualadas) puede reconstruirse mediante interpolación u otros métodos conocidos de reconstrucción de señales. Como se verá a continuación, también es posible almacenar la información sobre el error introducido por la igualación de la señal y utilizar esta información de error para reconstruir las muestras cerca del valor que tenían antes de la igualación, es decir, cerca del valor que tenían en el conjunto original 21 de datos digitales. Lo mismo se puede realizar, por supuesto, para cada uno de los conjuntos intermedios recuperados de datos digitales con el fin de restablecer las muestras igualadas a un valor lo más cercano posible al valor original de las muestras en el conjunto original de datos digitales.

En la siguiente descripción de las figuras 6, 7 y 8, los 2 canales originales se reducen en resolución de bits por ejemplo de 24 bits por muestra a 18 bits. Junto a la reducción de la resolución de la muestra, se reduce la frecuencia de muestreo a la mitad de la frecuencia de muestreo original (en este ejemplo se empieza a partir de 2 canales de audio cada uno con la misma resolución de bits y frecuencia de muestreo). Son posibles otras combinaciones, como empezando a partir de X bits y reduciendo a Y bits (por ejemplo,  $X/Y = 24/22, 24/20, 24/16$ , etc. ó  $20/18, 20/16$  ó  $16/15, 16/14, \dots$ ) dados los requisitos del audio de alta fidelidad, no se debe reducir una muestra a una resolución de bits por debajo de 14 bits. Si se mezclan más canales, la técnica básica descrita en esta memoria descriptiva necesita que la frecuencia de muestreo sea dividida por el número de canales, que deben ser mezclados en un solo canal. Cuantos más canales se mezclan, menor será la frecuencia de muestreo real de los canales (antes de la mezcla). En HD-DVD o DVD BLU-Ray la frecuencia de muestreo inicial puede ser de hasta 96 kHz o incluso (BLU-Ray) de hasta 192 kHz. Si se empieza a partir de 2 canales con una frecuencia de muestreo cada uno de 96 kHz, y

se reducen ambos a 48 kHz todavía se deja una frecuencia de muestreo en la gama de audio de alta fidelidad. Incluso 3 canales mezclados y reducidos a 32 kHz es aceptable para la calidad del audio de películas/televisión (esta es una frecuencia como la utilizada por el audio digital de televisión transmitida NICAM). Si se empieza partir de la grabación verdadera a 192 kHz, permite mezclar 4 canales, reduciendo la frecuencia de muestreo de 48 kHz.

5 La figura 6 muestra una conversión mejorada del primer conjunto de datos digitales. En la conversión mejorada los bits menos significativos de las muestras ya no representan a la muestra original, pero se utilizan para almacenar información adicional, tal como valores semilla, patrones de sincronización, información acerca de los errores provocados por la igualación de las muestras u otra información de control. El primer conjunto 20 de datos digitales comprende una secuencia de valores de muestra  $A_0, A_1, A_2, A_3, A_4, A_5, A_6, A_7, A_8, A_9$ . Cada muestra  $A_0, A_1, A_2, A_3,$   
10  $A_4, A_5, A_6, A_7, A_8, A_9$  se trunca dando lugar a muestras truncadas o redondeadas  $A_0', A_1', A_2', A_3', A_4', A_5', A_6', A_7', A_8', A_9'$ . Este conjunto 60 de muestras truncada  $A_0', A_1', A_2', A_3', A_4', A_5', A_6', A_7', A_8', A_9'$ , en las que se consideran los bits menos significativos, o en realidad ya no tienen información acerca de la muestra se procesan posteriormente como se explica en la figura 2. El conjunto 60 de muestras truncadas se divide en un primer subconjunto de muestras  $A_1', A_3', A_5', A_7', A_9'$  y un segundo subconjunto de muestras  $A_0', A_2', A_4', A_6', A_8'$ . Posteriormente cada uno de los valores de cada muestra  $A_1', A_3', A_5', A_7', A_9'$  del primer subconjunto de muestras se iguala con el valor de la muestra vecina  $A_0', A_2', A_4', A_6', A_8'$  del segundo subconjunto según lo indicado por las flechas en la figura 6. En particular, esto significa que el valor de la muestra  $A_1'$  se sustituye por el valor de la muestra vecina  $A_0'$ , es decir, el valor de la muestra  $A_1'$  se iguala con el valor de  $A_0'$  muestra. Esto da lugar a un primer conjunto intermedio 61 de datos digitales tal como se muestra, que comprende los valores de muestra  $A_0'', A_1'', A_2'', A_3'', A_4'', A_5'', A_6'', A_7'', A_8'', A_9''$ , etc., donde el valor  $A_0''$  es igual al valor  $A_0'$  y  $A_1''$  es igual al valor  $A_0'$  etc. Cabe señalar que, debido al truncamiento, es decir el redondeo de las muestras, se crea una zona reservada 62 en el primer conjunto intermedio 61 de datos digitales.

La figura 7 muestra una conversión mejorada del segundo conjunto de datos digitales. De la misma manera que para el primer conjunto de datos digitales, la conversión puede mejorarse porque los bits menos significativos de las muestras ya no representan a la muestra original, pero se utilizan para almacenar información adicional, tal como valores semilla, patrones de sincronización, información acerca de los errores provocados por la igualación de las muestras u otra información de control. El segundo conjunto 30 de datos digitales comprende una secuencia de valores de muestras  $B_0, B_1, B_2, B_3, B_4, B_5, B_6, B_7, B_8, B_9$ . Cada muestra  $B_0, B_1, B_2, B_3, B_4, B_5, B_6, B_7, B_8, B_9$  se trunca dando lugar a muestras truncadas o redondeadas  $B_0', B_1', B_2', B_3', B_4', B_5', B_6', B_7', B_8', B_9'$ . Este conjunto 70 de muestras truncada  $B_0', B_1', B_2', B_3', B_4', B_5', B_6', B_7', B_8', B_9'$ , en las que se consideran los bits menos significativos, o en realidad ya no tienen información acerca de la muestra se procesan posteriormente como se explica en la figura 3. El conjunto 70 de muestras truncadas  $B_0', B_1', B_2', B_3', B_4', B_5', B_6', B_7', B_8', B_9'$  se divide en un tercer subconjunto de muestras  $B_0', B_2', B_4', B_6', B_8'$  y un cuarto subconjunto de las muestras  $B_1', B_3', B_5', B_7', B_9'$ . Posteriormente cada uno de los valores de cada muestra  $B_0', B_2', B_4', B_6', B_8'$  del tercer subconjunto de muestras se iguala con el valor de la muestra vecina  $B_1', B_3', B_5', B_7', B_9'$  del cuarto subconjunto según lo indicado por las flechas de la figura 3. En particular, esto significa que el valor de la muestra  $B_2'$  se sustituye por el valor de la muestra vecina  $B_1'$ , es decir, el valor de la muestra  $B_2'$  se iguala con el valor de la muestra  $B_1'$ . Esto da lugar a un segundo conjunto intermedio 71 de datos digitales como se muestra, que comprende los valores de muestra  $B_0'', B_1'', B_2'', B_3'', B_4'', B_5'', B_6'', B_7'', B_8'', B_9''$ , donde el valor  $B_2''$  es igual al valor  $B_1'$  y  $B_1''$  es igual al valor  $B_1'$ , etc. Cabe señalar que, debido al truncamiento, es decir el redondeo de las muestras, se crea una zona reservada 72 en el segundo conjunto intermedio 71 de datos digitales.

La reducción de la resolución introducida por el redondeo, como se explica en la figura 6 y 7, en principio es "irrecuperable", pero se pueden aplicar técnicas para aumentar la frecuencia de la muestra percibida. Si se requiere más resolución de bits, la invención permite aumentar el valor de Y (bits utilizados realmente) a costa de menos "espacio" disponible para los datos codificados o X bits por muestra. Por supuesto, la aproximación de error almacenada en el bloque de datos en la zona de datos auxiliares permite una reducción sustancial de la percepción de pérdida de resolución. Para una secuencia de audio PCM de 24 bits, con un formato 18/6 y que mezcla 2 canales se tienen muestras de audio de 18 bits y muestras de datos de 6 bits, cada bloque de datos se inicia con una sincronización (sincr) de 6 muestras de datos (6 bits cada una), se utilizan 2 muestras de datos (12 bits en total) para almacenar la longitud del bloque de datos y, finalmente, se utilizan 2x3 muestras de datos (2x18bit) para almacenar duplicados de las muestras de audio. Para otros formatos (ejemplos):

- 16/8: sincronización de 8 muestras de datos, 2 muestras de datos (16 bits, sólo se utilizan 12 bits) para la longitud y 2x2 muestras de datos (2x16bit) para duplicados de las muestras de audio;
- 20/4: sincronización de 4 muestras de datos, 3 muestras de datos (12 bits en total) para la longitud y 2x5 muestras de datos (2x20bit) para duplicados de las muestras de audio
- 55 - 22/2: sincronización de 2 muestras de datos, 6 muestras de datos (12 bits en total) para la longitud y 2x11 muestras de datos (2x22bit) para duplicados de las muestras de audio

Para otros formatos (por ejemplo, audio PCM de 16 bits, con formato 14/2) se pueden definir estructuras similares.

La figura 8 muestra la codificación de los dos conjuntos de datos digitales resultantes en un tercer conjunto de datos digitales. La codificación se realiza de la misma manera a como se describe en la figura 4. Ahora que el primer

conjunto intermedio 61 de datos digitales tiene una zona reservada 62 y el segundo conjunto intermedio 71 de datos digitales también tiene una zona reservada 72, la suma de ambos conjuntos de datos digitales da como resultado ahora un tercer conjunto 80 de datos digitales con una zona 81 de datos auxiliares. En esta zona 81 de datos auxiliares se pueden colocar datos adicionales. Cuando el tercer conjunto 80 de datos digitales se reproduce mediante un equipo que no tiene conocimiento de la presencia de esta zona 81 de datos auxiliares, los datos de esta zona 81 de datos auxiliares serán interpretados por el equipo como los bits menos significativos del conjunto de datos digitales a reproducir. Los datos situados en esta zona 81 de datos auxiliares, por lo tanto, introducirá un ligero ruido en la señal que en gran medida es imperceptible. Esta imperceptibilidad, por supuesto, depende del número de bits menos significativos elegidos para ser reservados para esta zona 81 de datos auxiliares y es fácil para un experto elegir la cantidad adecuada de bits menos significativos a utilizar con el fin de equilibrar el requisito de almacenamiento de datos en la zona 81 de datos auxiliares y la pérdida resultante de calidad en el conjunto de datos digitales. Es evidente que en un sistema de audio de 24 bits el número de bits menos significativos dedicados a la zona 81 de datos auxiliares puede ser más alto que en un sistema de audio de 16 bits. Con el fin de que estos canales de audio mezclados permitan a la operación inversa (o separación), se almacenan copias duplicadas de un número limitado de muestras. Aunque en los ejemplos anteriores sólo se utiliza y se almacena una sola muestra de valor semilla, es decir, copia duplicada de una muestra, el almacenamiento de múltiples muestras de valor semilla tiene la ventaja de que se proporciona redundancia. Esta redundancia se debe a la vez a la naturaleza repetitiva de los valores semilla almacenados que permiten la recuperación de los errores al proporcionar nuevos puntos de partida en la secuencia y al hecho de que se pueden almacenar dos valores semilla para cada posición inicial. Los valores semilla A0 y B1 permiten la verificación de la posición inicial ya que el cálculo que empieza con A0 dará el valor B0 que entonces puede compararse con el valor semilla almacenado para su verificación. Otra ventaja es que el almacenamiento a la vez de A0 y B1 permite una búsqueda de la correcta posición inicial a la que pertenecen los dos valores semilla, lo que permite una sincronización por sí mismos entre los valores semilla y el conjunto C de datos digitales, ya que es probable que en una posición en la que se descodifica usando el valor semilla A0 se dará lugar a exactamente un valor B1 que es igual al valor semilla almacenado B1. Al empezar, como ejemplo, a partir de una señal de muestreo de 24 (Z) bits a 96 kHz reducida a 18 (Y) bits 48 kHz, y al crear un duplicado de una muestra por valor de ms., es decir, un valor semilla por ms., 1000 duplicados de muestra de 18 bits, es decir, los valores semilla, por canal mezclado. Si esta mezcla incluye 2 canales, se necesitan 2x1000x18bits ó 36 kbits de "almacenamiento" para duplicados de muestra por segundo. Como se creó el primer "espacio" extra - 6 (X) bits por muestra a 96K por segundo – hay disponibles 6x96 = 576K bits por segundo en la zona de datos auxiliares formada por los bits menos significativos, en la que se pueden almacenar fácilmente estas copias duplicadas de los valores de muestra. De hecho, hay 16 veces la memoria disponible para almacenar estas copias y, como tal, sería posible almacenar duplicados de las muestras de estos 2 canales a una velocidad de 16 veces por ms si no se fuera a almacenar otra información en esta zona de datos auxiliares. Si se seleccionan otros valores para Z/Y/X, por ejemplo, 24/20/4 a 96 kHz ó 16/14/2 a 44,1 kHz la cantidad de la zona "libre" creada de datos auxiliares mediante el uso de los bits menos significativos será diferente. Los siguientes casos se dan como ejemplos, pero la invención no se limita a estos otros casos de uso; 2 canales a 24/20/4 @ 96 kHz y 4x96 = 392K bits por segundo memoria que requiere 2x1000x20 = 4Kbits para duplicados de muestras por ms, es posible almacenar duplicados de muestras a una velocidad de 9,6 veces por ms. 2 canales a 16/14/2 @ 44,1 kHz y 2x44,1 = 88,2K bits por segundo requisito de memoria 2x1000x14 = 28Kbits para duplicados de muestras por ms, es posible almacenar duplicados de muestras a una velocidad de 3,15 por ms. Los ejemplos mencionados en esta memoria utilizan la zona de datos auxiliares formada por los bits menos significativos de las muestras exclusivamente para la duplicación de las muestras a partir de la secuencia de audio original (resolución y frecuencia reducidas). Debido a la naturaleza y las características de la técnica que se utiliza en esta memoria, es beneficioso no sólo utilizar esta zona "libre" de datos auxiliares para el almacenamiento de duplicados de muestras, aunque estos duplicados de muestras son información esencial utilizada por el proceso de separación o el descodificador.

En la técnica Básica, como se explica en las figuras 2-8, 2 secuencias de audio PCM A (A<sub>0</sub>, A<sub>1</sub>, A<sub>2</sub>) y B (B<sub>0</sub>, B<sub>1</sub>, B<sub>2</sub>), se reducen en primer lugar en cuanto a resolución de bits, para generar 2 nuevas secuencias A' (A'<sub>0</sub>, A'<sub>1</sub>, A'<sub>2</sub>) y B' (B'<sub>0</sub>, B'<sub>1</sub>, B'<sub>2</sub>). A continuación la frecuencia de muestreo de estas secuencias se reduce a la mitad de la frecuencia de muestreo original, dando A'' (A''<sub>0</sub>, A''<sub>1</sub>, A''<sub>2</sub>) y B'' (B''<sub>0</sub>, B''<sub>1</sub>, B''<sub>2</sub>). Esta última operación introduce Errores, con A''<sub>2i</sub> = A''<sub>2i+1</sub> = A'<sub>2i</sub> generando un Error E<sub>2i+1</sub> = A'<sub>2i+1</sub>-A'<sub>2i</sub> y B''<sub>2i+1</sub> = B''<sub>2i+2</sub> = B'<sub>2i+1</sub> (B''<sub>0</sub> = B'<sub>0</sub>) generando un error E<sub>2i+2</sub> = B'<sub>2i+2</sub> - B'<sub>2i+1</sub> (E<sub>0</sub> = 0). Esta Serie de Errores (E<sub>0</sub>, E<sub>1</sub>, E<sub>2</sub>, E<sub>3</sub> ...) contiene Errores con índice par debido a la reducción de muestreo de la secuencia de audio B y con errores con índice impar de muestreo debido a la reducción de muestreo de la secuencia de audio A. La codificación avanzada aproximará estos Errores y utilizará estas aproximaciones para reducir los errores antes de la mezcla. Los Errores aproximados (que se representan como los inversos de los Errores reales) E' se añaden como un canal independiente establecido en la zona de datos auxiliares en los bits menos significativos de las muestras como parte de la mezcla. Como tal, la señal mezclada se define por Z = A''+B''+E' con muestras (Z<sub>i</sub> = A''<sub>i</sub>+B''<sub>i</sub>+E'<sub>i</sub>). Si la secuencia de Errores se puede aproximar exactamente entonces E' = E con Z<sub>2i</sub> = A''<sub>2i</sub>+B''<sub>2i</sub>+E<sub>2i</sub> = A'<sub>2i</sub>+B'<sub>2i-1</sub>+B'<sub>2i</sub>-B'<sub>2i-1</sub> = A'<sub>2i</sub>+B'<sub>2i</sub> y Z<sub>2i+1</sub> = A''<sub>2i+1</sub>+B''<sub>2i+1</sub>+E<sub>2i+1</sub> = A'<sub>2i</sub>+B'<sub>2i+1</sub>+A'<sub>2i+1</sub>-A'<sub>2i</sub> = A'<sub>2i+1</sub>+B'<sub>2i+1</sub>. En tal caso, no se generan errores de reducción de muestreo en secuencia de mezcla final.

La figura 9 muestra la descodificación del tercer conjunto de datos digitales de nuevo en dos conjuntos separados de datos digitales. La descodificación del conjunto 80 de datos digitales obtenidos por la codificación mejorada, es decir con los bits menos significativos 81 utilizados para almacenar datos adicionales, se lleva a cabo al igual que la descodificación regular descrita en la figura 5, pero sólo son proporcionados por el descodificador los bits relevantes

de cada muestra  $A_0, A_1, A_2, A_3, A_4, A_5, A_6, A_7, A_8, A_9, B_0, B_1, B_2, B_3, B_4, B_5, B_6, B_7, B_8, B_9$ , es decir, no los bits menos significativos. El decodificador puede además recuperar los datos adicionales almacenados en la zona 81 de datos auxiliares en los bits menos significativos. Estos datos adicionales posteriormente se pueden pasar a lo largo del destino de los datos adicionales como se explica en la figura 20. Una vez que el decodificador tiene estos duplicados de muestras, los valores semilla, reconstruidos, estos duplicados de muestras (valores semilla) se utilizan entonces para separar el canal mezclado. El canal mezclado es, por ejemplo, una mezcla de las secuencias PCM A" y B", con  $A''_{2i} = A''_{2i+1} = A'_{2i}$  y  $B''_{2i+1} = B''_{2i+2} = B'_{2i+1}$ .  $A'_0$  y  $B'_1$  se utilizarán como duplicados de muestras y se codificarán en el bloque de datos.

La separación de las señales (mono) a partir de A"+B" se puede hacer, alternativa al método explicado en la figura 5, cuando sólo se utiliza un valor semilla, de la siguiente manera: Las muestras A"+B" son:  $A''_0+B''_0, A''_1+B''_1, A''_2+B''_2, A''_3+B''_3, A''_4+B''_4, A''_5+B''_5$ . Debido a que se tiene una copia de  $A'_0 = A'_0$  y  $B'_1 = B'_1$  se pueden reconstruir las secuencias A" y B".

1. con  $A''_0+B''_0 - (A''_0=A'_0)$  se obtiene  $B''_0$  y se consigue  $A''_0$  del duplicado de muestra
2. con  $A''_1+B''_1 - (A''_1=A'_1)$  se obtiene  $A''_1$  y se consigue  $B''_0$  del duplicado de muestra
3. con  $A''_2+B''_2 - (B''_2=B''_1)$  se obtiene  $A''_2$  y  $B''_2 = B''_1$
4. con  $A''_3+B''_3 - (A''_3=A''_2)$  se obtiene  $B''_3$  y  $A''_3 = A''_2$
5. con  $A''_4+B''_4 - (B''_4=B''_3)$  se obtiene  $A''_4$  y  $B''_4 = B''_3$
6. con  $A''_5+B''_5 - (A''_5=A''_4)$  se obtiene  $B''_5$  y  $A''_5 = A''_4$

En los formatos multimedia como HD-DVD o Blu-Ray DVD el audio de varios canales de puede almacenar como una mezcla múltiple (*multiplex*) de secuencias de audio PCM. Utilizando la técnica de mezcla/separación como se explicó anteriormente en cada uno de estos canales, se puede duplicar fácilmente el número de canales (de 6 u 8 a 12 ó 16). Esto permite almacenar o crear una 3ª dimensión de la grabación o la reproducción de audio mediante la adición de un altavoz superior por encima de todos los altavoces de tierra, pero no es necesario que el usuario tenga un decodificador para escuchar la versión de "2 dimensiones" del audio, ya que el audio almacenado en las pistas de audio de varios canales sigue siendo audio PCM "que se puede reproducir" al 100%. En este último modo de reproducción, el efecto de la 3ª dimensión no se creará, pero tampoco se degradará la calidad perceptible de la grabación de audio de 2 dimensiones.

La figura 10 muestra un ejemplo en el que se representan las muestras de la primera secuencia A tal como se obtienen mediante la codificación que se describe en la figura 6. A modo de ejemplo, se supone que se van a procesar 2 secuencias A y B de audio digital mono a 96 kHz de 24 bits. A = muestras originales (24 bits), A' = muestras redondeadas (18H bits significativos y 6L bits = 0), A" = Muestras con Frecuencia de muestreo reducida. En la Figura 10, se muestra una primera secuencia de audio A en el gráfico como una línea gris oscura. Las muestras de A son:  $A_0, A_1, A_2, A_3, A_4, A_5, \dots$ . La resolución de cada muestra es de 24 (Z) bits por muestra, representa como un valor entero con signo de 24 bits, por lo que los valores oscilan entre  $-2^{(Z-1)}$  a  $2^{(Z-1)-1}$ . A partir de esta serie de muestras, se reduce la resolución a 18 (Y) bits, despejando los 6 (X) bits menos significativos para crear "espacio" para los datos codificados. La reducción se consigue mediante el redondeo de todas las muestras de Z bits a su representación más cercana utilizando sólo los Y bits más significativos de un total de Z. Hasta aquí cada muestra se incrementa con  $(2^{(X-1)}-1)$ , cada total se limita a  $(2^{(Z-1)}-1)$  o se representa como  $[(2^{(Z-1)}-1)$ . A continuación se establecen los 6 (X) bits menos significativos a 0 mediante operación AND bit a bit con  $((2^{(Y)}-1)$  bit a bit desviados X bits a la izquierda), por lo tanto se genera una nueva secuencia A' (gris claro). Las muestras de A' son:  $A'_0, A'_1, A'_2, \dots$  con

$$A'_i = [A_i + (2^{(X-1)} - 1)]_{(2^{(Z-1)} - 1)} \text{ AND } ((2^{(Y)} - 1) \ll X)$$

Tras la reducción de la resolución de la muestra también se reduce la frecuencia de muestreo por un factor de 2 (en caso de que se mezclen más de 2 canales se necesita reducir la frecuencia de muestreo por un factor igual al número de canales mezclados). Aquí se repiten todas las muestras pares de la secuencia original A'. Después de la reducción de la frecuencia de la muestra se obtiene una nueva secuencia A". Las muestras de A" son:  $A''_0, A''_1, A''_2, \dots$  con

$$A''_{2i} = A''_{2i+1} = A'_i$$

Todas las muestras pares de A" con índice 2i son idénticas a los datos originales de A' con índice 2i y todas las muestras impares de A" con índice 2i + 1 son duplicados de la muestra anterior de A" con índice 2i.

La figura 11 muestra un ejemplo en el que se representan las muestras de la primera secuencia B tal como se obtienen mediante la codificación que se describe en la figura 7. B = muestras originales (24 bits), B' = muestras redondeadas (18H bits significativos y 6L bits = 0), B" = Muestras con Frecuencia de muestreo reducida. En la Figura 11, se muestra una segunda secuencia de audio B en el gráfico como una línea gris oscura. A esta secuencia se

aplica la misma reducción de resolución de muestra. Las muestras de B son:  $B_0, B_1, B_2, B_3, B_4, B_5, \dots$ . A partir de esta serie de muestras, se genera una nueva secuencia  $B'$  (gris claro). Las muestras de  $B'$  son:  $B'_0, B'_1, B'_2, \dots$  con

$$B'_i = [B_i + (2^{(X-1)} - 1)]_{(2^{(Z-1)} - 1)} \text{ AND } ((2^{(Y)} - 1) \ll X)$$

Tras la reducción de la resolución de la muestra que también se reduce la frecuencia de muestreo de manera similar por un factor de 2 y se obtiene una nueva secuencia  $B''$ . Las muestras de  $B''$  son:  $B''_0, B''_1, B''_2, \dots$

con

$$B''_{2i+1} = B''_{2i+2} = B'^{2i+1}$$

Todas las muestras impares de  $B''$  con índice  $2i+1$  son idénticas a los datos originales de  $B'$  con índice  $2i+1$  y todas las muestras pares de  $B''$  con índice  $2i+2$  son duplicados de la muestra anterior de  $B''$  con índice  $2i+1$ .

10 La figura 12 muestra las muestras de la secuencia mezclada C.  $A+B$  = muestras originales (24 bits),  $A'+B'$  = muestras redondeadas (18H bits significativos y 6L bits = 0),  $A''+B''$  = Muestras con Frecuencia de muestreo reducida.

15 Ambas secuencias  $A+B$  se mezclan (suman) para obtener una nueva secuencia (gris oscuro). Se mezclan (suman) las secuencias  $A''$  y  $B''$  y se obtiene otra secuencia (gris claro).  $A''+B''$  será diferente de  $A+B$  y de  $A'+B'$  para cada muestra, ya que  $A''$  o  $B''$  pueden ser diferentes de las muestras originales  $A$  y  $B$  debido a la reducción de la resolución de bits (redondeo), y pueden diferir de las muestras con resolución reducida debido a la reducción de muestreo, pero en general, todavía se tiene una buena aproximación de percepción de la secuencia original  $A+B$  (gris oscuro) debido a la alta resolución de bits y la alta frecuencia de muestreo originales.

20 La figura 13 muestra los errores introducidos en la secuencia de PCM por la invención. Error = Errores debidos a las muestras de redondeo, Error' = Errores debidos a las muestras de redondeo + reducción de frecuencia.

25 La figura 14 muestra el formato de la zona de datos auxiliares en los bits menos significativos de las muestras del conjunto de datos digitales combinados. Por último, para permitir que el decodificador separe los datos de audio PCM mezclados, el decodificador necesita tener las muestras duplicadas de las muestras de audio PCM ANTES de recibir las muestras de audio PCM, de manera que la operación de separación se pueda realizar en tiempo real con la secuencia de audio PCM. Aquí es necesario poner estos datos de un bloque de datos (que contienen duplicados de muestras de las muestras de audio, patrones de sincronización, parámetro de longitud...) en las muestras (Z bits) que también llevan la información de audio PCM relacionada con el bloque de los datos anterior. Para dar tiempo al decodificador para decodificar estos bloques de datos, se puede incluso terminar varias muestras de audio PCM antes de las muestras de audio PCM que se utilizaron para tomar sus duplicados. El número de muestras de audio PCM entre el final de un bloque de datos y las muestras de audio PCM que se utilizaron para copiar como duplicados de muestras es el Desplazamiento, que es otro parámetro almacenado en el bloque de datos. A veces, este desplazamiento puede ser negativo, lo que indica que la posición de las muestras duplicadas en la secuencia de audio PCM está dentro de las muestras de audio PCM utilizadas para llevar a ese bloque de datos. Para el desplazamiento también utilizará un valor de 12 bits (valor entero con signo).

35 Un bloque de datos comprende:

1. Un patrón de sincronización (Sincr)
2. Una longitud de bloque de datos
3. Un desplazamiento de muestra de audio PCM con referencia al final de ese bloque de datos.
4. Duplicados de las muestras de audio PCM (una para cada canal mezclado)

40 Una ventaja adicional se consigue mediante la inclusión de información de corrección que permite una invalidación (parcial) del error introducido por la igualación de las muestras.

45 En la figura 14, en el momento 0 el codificador comienza a leer  $2x$  U muestras de X bits, que se reducen a Y bits para crear la zona de datos auxiliares para contener los bloques de datos. La reducción de la frecuencia de muestreo crea errores, que se aproximan y se sustituyen por una lista de referencias a estas aproximaciones. Aparte de estos datos - que se comprimen con efectividad - se generan las cabeceras de los bloques de datos (sincronización, longitud, desplazamiento, etc...) dando lugar a una longitud de bloque de datos de muestras  $U'$ . Estas muestras de datos se colocan dentro de la sección de datos de las primeras U muestras. En una siguiente etapa, el codificador lee las  $U'$  ( $<U$ ) muestras, lo que da lugar a un bloque de datos que (descomprimido) necesita U muestras, pero después de la compresión  $U''$ . Una vez más este bloque de datos se une al bloque de datos anterior y en este ejemplo (todavía) utiliza algunas muestras de las U muestras iniciales (Xbit). El proceso del codificador al leer  $U'$  muestras X bits y generar los bloques de datos correspondientes continúa hasta que se hayan procesado todos los datos.



La figura 15 muestra más detalles de la zona de datos auxiliares.

El Formato de Soporte de Datos AUROFÓNICOS cumple con la siguiente estructura; es una secuencia de datos/audio de bits precisos, normalmente una secuencia PCM 150, en la que los datos se dividen en secciones 158, 159 de Z muestras. Cada muestra en la sección 158, 159 consiste en X bits. (X será por lo general 16 bits para datos de audio de CD/DVD, o 24 bits para datos de audio de Blu-Ray/HD-DVD). Los bits más significativos (Primeros Y bits, por ejemplo, para Blu-Ray normalmente 18 ó 20 bits) contienen los datos de audio (pueden ser datos de audio PCM), los bits menos significativos (los últimos Q bits, por ejemplo para el formato Blu-Ray normalmente 6 ó 4 bits) contienen los datos de decodificación AURO.

Los datos AURO adicionales tal como se utilizan durante la decodificación de cada bloque de datos 156, 157 se organizan de la siguiente manera; comprenden una sección de sincronización "Sincr" 151, una sección 154 de datos de decodificación de uso general, opcionalmente una Lista de Índices 152 y una Tabla 153 de errores y finalmente un valor de CRC 155. La sección Sincr 151 se define previamente como un patrón de bits de rodadura (el tamaño depende del número de bits Q utilizados para la anchura de datos AURO). Los datos de uso general 154 incluyen información acerca de la longitud del bloque de datos AURO, el desplazamiento exacto (en relación a la posición de sincronización 151) de los primeros datos de audio (PCM) 158 en los que se han de aplicar los datos 156 de decodificación AURO, las copias de la primera muestra de datos de audio (PCM) (una para cada canal codificado), los datos de atenuación y otros datos. Opcionalmente (dependiendo de la selección de calidad AURO durante el proceso de codificación), estos datos de codificación AURO 156, 157 pueden incluir también una Lista 152 de índices y una tabla 153 de errores que contiene aproximaciones de todos los Errores generados durante la etapa de codificación. Además, también opcionalmente, La Lista 152 de índices y la Tabla 153 de Errores pueden comprimirse. La sección 154 de datos de decodificación de uso general indicará si está presente dicha Lista 152 de índices y Tabla 153 de Errores, incluyendo información acerca de la compresión aplicada. Por último, el valor de CRC 155 es un CRC calculado utilizando tanto los datos de Audio PCM (Y bits) como los datos AURO Q (bits).

Una característica del decodificador AURO es su latencia extremadamente baja. Para la decodificación sólo es necesario un retraso de procesamiento de 2 muestras AURO (PCM). La información del bloque de datos AURO 156, 157 debe transmitirse y procesarse (por ejemplo, descomprimirse) antes de transmitir los datos de audio PCM 158 a los que han de aplicarse los datos de decodificación AURO. Como consecuencia de ello, el bloque de datos AURO 156, 157 (bits menos significativos) se combina con los datos de Audio PCM 159 (bits más significativos) de tal manera que la última información de los datos AURO 154, 155 de un bloque no está más tarde que la primera muestra de datos de audio (PCM) a la que se aplica esa información de datos AURO.

El decodificador que implementa la operación de separación de los canales utiliza patrones de sincronización para que pueda localizar, por ejemplo, las muestras duplicadas y relacionarlas con las muestras originales coincidentes. Estos patrones de sincronización se pueden colocar además en los 6 (X) bits por muestra y deben ser fácilmente detectables por el decodificador. Un patrón de "sincronización" puede ser un patrón repetido de una secuencia de varias "claves" de 6 (X) bits de largo. Por ejemplo, teniendo una desviación de un solo bit desde la posición menos significativa a la posición más significativa, o binario representado como: 000001, 000010, 000100, 001000, 010000, 100000. Pueden seleccionarse otros patrones de bits en base a las características de las muestras con el fin de evitar que los patrones de sincronización afecten a las muestras de una manera perceptible, o que las muestras afecten a la detección de los patrones de sincronización. Como tal, se pueden definir patrones de sincronización uniforme para todas las combinaciones diferentes de resoluciones de muestra. (24/22/2, 24/20/4/, 24/18/6, 24/16/8, 16/14/2,...) Estos patrones también se pueden optimizar para eliminar el "ruido" generado a partir de los bits menos significativos de las muestras de audio, cuando se reproducen mediante un reproductor de DVD que no utiliza un decodificador AURO-Fónico.

La figura 16 muestra una situación en la que la adaptación conduce a bloques de datos AURO de longitud variable. Además, es necesario que el decodificador reciba la información de los bloques de datos antes de procesar las muestras de audio mezcladas, como tiene que decodificar los bloques de datos (incluyendo la descompresión) y tiene que acceder a estos Errores (aproximados) con el fin de llevar a cabo la operación de separación. Las muestras de secuencia de Errores (de ese 2º bloque) se aproximarán (utilizando algoritmos K-Medias o Localización de Instalaciones (*Facility Location*)) con una tabla que contiene aproximaciones y una lista de referencias para vincular cada muestra de esa sección de secuencia de Errores a un elemento de esa tabla de aproximación. Esta lista de referencias forma la secuencia aproximada de Errores. Tanto la lista como la tabla con los valores de aproximación se comprimen mediante un compresor, los otros elementos restantes de la estructura de datos se definen mediante un elemento para dar formato (como patrón de sincronización, longitud del bloque de datos, desplazamiento, duplicados de muestras de audio, atenuación, etc.) de tal manera que (muy probablemente) uno terminaría con menos de U muestras de datos, un número de muestras al que se hará referencia como W ( $W \leq U$ ). Se puede esperar que el valor W sea normalmente de 20-50% más pequeño que U. A continuación este bloque de datos se coloca en el espacio de datos de las primeras U muestras mediante el elemento para dar formato. Esto garantiza que estas muestras de datos estarán disponibles para el decodificador antes de que reciba las muestras de audio coincidentes. Como es posible que se haya ahorrado en muestras de datos, (U-W) para su uso posterior, la siguiente sección de audio a codificar (esta es la aproximación de mezcla y error) debe contener sólo W muestras de audio ( $\leq U$ ). Incluso si el bloque de datos para esta sección (de W muestras de audio) necesitara U muestras de datos, se garantiza que tiene el final de este bloque de datos antes de la primera muestra de audio a la se refiere.

Además, debido a un menor número de muestras de audio ( $W \leq U$ ) se puede esperar que la aproximación del error de reducción de frecuencia de muestreo sea mejor, ya que se tiene que aproximar un menor número de valores de Error. Como tal, la ganancia de compresión se utiliza mediante una mejor aproximación de la siguiente sección de muestras de audio. Una vez más, esta última sección del bloque de datos podría ser menor que  $U$ , por ejemplo,  $W' (\leq U)$  de tal manera que el siguiente número de muestras de audio a codificar podría a su vez también limitarse a  $W'$ .

Se entiende además que el tamaño del bloque de datos puede variar, dependiendo de la calidad de compresión. Como consecuencia, el parámetro de desplazamiento (parte de la estructura del bloque de datos) es un parámetro importante para vincular los bloques de datos de tamaño variable con la correspondiente primera muestra de audio. La longitud del bloque de datos en sí coincide con el número de muestras necesarias durante la descodificación de audio, empezando desde la primera muestra de audio que se vinculó con el bloque de datos con el parámetro de desplazamiento. Este parámetro de desplazamiento puede aumentarse aún más si es necesario (y el bloque de datos ser desviado más hacia atrás en el tiempo) cuando en determinados casos el descodificador necesita más tiempo para iniciar la descodificación del bloque de datos en relación con el momento en el que recibe la primera muestra de audio coincidente. Se entiende además que la descodificación del bloque de datos se debe ejecutar por lo menos en tiempo real por el descodificador, ya que dichos retrasos no pueden aumentar.

Otra característica de esta invención es que el descodificador se mantendrá fácilmente en sincronía con las referencias de sincronización y, además, detectará automáticamente el formato de codificación utilizado (detecta el número de bits de una muestra de audio utilizada para los patrones de sincronización/duplicados de muestra). A esto se incluye el número de muestras entre cada primera palabra de un patrón de sincronización como parte de los datos codificados. También se requiere que los patrones de sincronización se repitan después de como mucho  $4096 \times 2$  ( $2 =$  el número de canales mixtos) muestras. Esto reduce la longitud máxima de un bloque de datos (patrón de sincronización + datos de duplicado de muestra) a  $4096 \times 2$  muestras que requieren 12 bits para almacenar la longitud de cada bloque de datos. Usando esta información, y teniendo en cuenta las diferentes resoluciones de codificación por ejemplo para muestras PCM de 24 bits: 22/2, 20/4, 18/6, 16/8, el descodificador debe ser capaz de identificar automáticamente el formato de codificación, detectar los patrones de sincronización y sus repeticiones fácilmente.

La integración de datos auxiliares en la zona de datos formada por los bits menos significativos de las muestras se puede utilizar independientemente del mecanismo de combinación/descifrado. También en una única secuencia de audio esta zona de datos puede crearse sin afectar de manera audible a la señal en el que se integran los datos auxiliares. La integración de aproximaciones de error para los errores debidos a la reducción de la frecuencia de la muestra (equivalencia de muestras) sigue siendo beneficiosa si no se lleva a cabo la combinación, ya que también permite la reducción de la frecuencia de muestreo (con el consiguiente ahorro de espacio de almacenamiento), sin embargo permite una buena reconstrucción de la señal original usando las aproximaciones de error como se ha explicado para combatir los efectos de la reducción de la frecuencia de muestras.

La figura 17 muestra la codificación incluyendo todas las mejoras de las realizaciones. Los bloques mostrados corresponden tanto a las etapas del método como por igual a bloques del equipo físico del codificador y muestran el flujo de datos entre los bloques de equipo físico, así como entre las etapas del método.

#### Etapas de Procesamiento de la Codificación

En la primera etapa las secuencias de Audio A, B, primero se reducen mediante redondeo de muestras de audio ( $24 \rightarrow 18/6$ ) a  $A'$ ,  $B'$ . En la segunda etapa, las secuencias reducidas tienen un mezclado previo (utilizando datos de atenuación) aplicando compresión dinámica de estas secuencias para evitar el recorte del audio ( $A'^C$ ,  $B'^C$ ). En la tercera etapa la frecuencia de muestreo se reduce por un factor igual al número de canales mezclados ( $A'^C$ ,  $B'^C$ ) introduciendo una secuencia de Errores E. En la cuarta etapa la secuencia de errores E se aproxima por  $E'$ : utilizando  $2^{(z-1)}$  centros (por ejemplo, aproximación de K-Medias) y una lista de referencia a estos centros. En la quinta etapa la tabla y las referencias se comprimen, la atenuación se muestrea (inicio de muestras de audio), las cabeceras de bloque se definen (sincr, longitud, ..., crd). En la sexta etapa las secuencias ( $A'^C$ ,  $B'^C$ ,  $E'$ ) se mezclan incluyendo una comprobación final contra el recorte (rebasado de audio) - esta comprobación puede necesitar cambios de menor importancia. En la séptima etapa la sección del bloque de datos (muestras de 6 bits) se fusiona con muestras de audio.

La figura 17 proporciona una visión general de una combinación de las etapas de procesamiento tal como se explica en las secciones anteriores. Se entiende que este proceso de codificación funciona más fácilmente cuando se aplica en una situación fuera de línea, el codificador tiene acceso a las muestras de las secciones correspondientes de todas las secuencias que tiene que procesar en cualquier momento. Por lo tanto, es necesario que las secciones de las secuencias de audio se almacenen por lo menos temporalmente por ejemplo en un disco duro de tal manera que el proceso del codificador pueda buscar (ida y vuelta) para utilizar los datos que necesita para el procesamiento de esa sección. En la explicación de la figura 17, se usa como ejemplo un caso de una muestra de 24 bits ( $X/Y/Z$ ) =  $(24/18/6)$  que se divide en un valor de muestra de 18 bits y un valor de datos de 6 bits que forma parte de la zona de datos auxiliares que contiene los datos de control y los valores semilla.

La longitud del bloque - con el fin de generalización - recibe la referencia U.

Una primera etapa <1> del proceso de codificación es (como se explica en la sección acerca de la técnica básica) la reducción de la resolución de la muestra a la vez en la secuencia A 161a y la secuencia B 161b, por ejemplo de 24 a 18 bits mediante los reductores de tamaño de muestra, por redondeo de cada muestra a su representación más cercana de 18 bits. Estas secuencias 163a, 163b, que son el resultado de este redondeo reciben las referencias secuencia A' 163a y secuencia B' 163b. Paralelamente, la atenuación es determinada por un controlador de atenuador que recibe un valor deseado de atenuación 161c a partir de una entrada.

La segunda etapa <2> es una simulación de mezcla de estas secuencias 163a, 163b mediante un manipulador de atenuación para analizar si la mezcla podría causar recorte. Si es necesario atenuar una secuencia 163b, por lo general la secuencia de audio con 3ª dimensión en el caso de la codificación AURO-FÓNICA, antes de la mezcla, esta atenuación debe ser tenida cuenta en esta simulación de mezcla por el manipulador de atenuación. Si a pesar de esta atenuación, la mezcla de ambas secuencias (96 kHz) 163a, 163b generara recorte, esta etapa del proceso de codificación realizada por el manipulador de atenuación se realizará una compresión suave (atenuación aumentada gradualmente de las muestras de audio hacia el punto de recorte y a continuación una disminución gradual de las mismas). Esta compresión se puede aplicar a ambas secuencias 163a, 163b por el manipulador de atenuación, pero esto no es necesario, ya que (más) compresión en una secuencia 163b también podría eliminar este recorte. Cuando se aplica a estas secuencias A' 163a y B' 163b, el controlador de atenuación genera una nueva secuencia A<sup>c</sup> 165a y secuencia B<sup>c</sup> 165b. El efecto de esta atenuación para evitar el recorte será persistente en la secuencia final mezclada 169, así como en las secuencias sin mezclar. En otras palabras, el descodificador no compensará esta atenuación para generar la secuencia original A' 163a o la secuencia original B' 163b, pero su objetivo será generar A<sup>c</sup> 165a y B<sup>c</sup> 165b. Durante la producción final de estas grabaciones (Aurofónicas), el ingeniero de grabación puede definir - si es necesario - el nivel de atenuación 161c y proporcionarlo mediante una entrada al controlador de atenuación para controlar la atenuación de la segunda secuencia 163b (normalmente la secuencia de audio de 3ª dimensión), que se desea cuando se mezclan hacia una reproducción de audio de 2 dimensiones.

En la siguiente etapa <3> la Frecuencia de muestreo se reduce mediante un reductor de frecuencia por un factor igual al número de canales mezclados (A<sup>c</sup> W<sup>c</sup>) introduciendo una secuencia de Errores E 167. La reducción de frecuencia se puede realizar por ejemplo como se explica en la figura 2 y 3 ó 6 y 7. En la siguiente etapa <4> la secuencia de error E167 se aproxima por E' 162 generado por un dispositivo de aproximación de error: utilizando  $2^{(z-1)}$  centros (por ejemplo, aproximación de K-Medias) y una lista de referencia a estos centros. En la sección de codificación/descodificación avanzada, se explicó que los errores 167 (debidos a la reducción de la frecuencia de muestras) en la operación de mezclado y separación podrían evitarse en la situación en la que esta secuencia de Errores 167 se aproximara sin errores. En este ejemplo en particular (X/Y/Z) = (24/18/6) y V = 32 ( $2^{(z-1)}$ ) aproximaciones, lo más probable sería que no hubiera errores (aparte de las limitaciones debidas a la representación de 12 bits de los Errores) cuando sólo se tienen V muestras en un bloque de datos de tal manera que hay una asignación uno a uno de estos Errores con estas "aproximaciones". En el otro extremo también se define la longitud máxima U del bloque de datos, que en ningún caso garantiza que la lista de referencia de Errores y la tabla de aproximaciones se podrían "codificar" en un bloque de datos como ese. Por lo tanto esta etapa de la codificación inicialmente necesitará un número de U muestras desde ambas secuencias A<sup>c</sup> 165a y B<sup>c</sup> 165b y de la secuencia de errores E 167.

En primer lugar se selecciona la anchura de la muestra de Errores (esta es el número de bits utilizados para representar esta información de errores). Dado que la secuencia básica son datos PCM procedentes de una grabación de audio, se puede esperar que los Errores o las diferencias entre 2 muestras adyacentes sean relativamente pequeños en comparación con la muestra Máxima (o Mínimo). En una señal de audio de (por ejemplo) 96 kHz, este Error puede ser relativamente grande sólo cuando la secuencia de audio contiene señales con frecuencias muy altas. Como se explicó anteriormente, en esta memoria descriptiva, se utiliza una secuencia de 24 bits PCM, reducida a 18 bits para el audio y que crea un espacio para 6 bits de datos por muestra. Estos bits de datos se utilizan, como se explica en la técnica básica, para almacenar el patrón de sincronización, la longitud de un bloque de datos, el desplazamiento, los parámetros a definir, 2 muestras duplicadas (cuando se mezclan 2 canales), una "lista de índices a Errores" comprimida, una tabla de Errores comprimida y una suma de verificación. La "lista de índices de errores" y la tabla de Errores se explican a continuación. En el ejemplo de 24/18/6, hay disponibles 6 bits por muestra para la zona de datos auxiliares y los 6 bits por muestra en teoría podrían definir una tabla con  $2^6 = 64$  errores cuando fuera necesario. En este ejemplo de 24/18/6, las representaciones de Error se limitarán a un número entero con signo de 2x6 bits.

Parte del contenido de un bloque de datos en la zona de datos auxiliares con U muestras de 6 bits (24/18/6 - para cada muestra del bloque de datos, hay una muestra (mezclada) de audio), es una tabla con las aproximaciones de los Errores debidos a la reducción de frecuencia de muestras de estas secuencias. Como se mencionó anteriormente un Error se aproximará utilizando 2 muestras de datos de 6 bits. Dado que no hay suficiente "espacio" para almacenar una aproximación para cada Error, hay que definir un número limitado de valores de error, que se aproxima - lo más cerca posible - a todos estos Errores. A continuación se crea una lista que incluye referencias a estos Errores' aproximados para cada elemento de la "secuencia" de Errores en el bloque de datos en la zona de datos auxiliares. Además de la sincronización, la longitud, desplazamiento, duplicados de muestra, etc. se necesita

espacio para almacenar una tabla con los Errores' aproximados en el bloque de datos. Esta tabla puede comprimirse, para limitar la memoria utilizada por el bloque de datos, y además la lista de referencias también puede comprimirse.

5 En primer lugar se explorará el camino para aproximar estos elementos a partir de la secuencia de Errores. Lo que hay que definir es un número K de valores, de manera que cada elemento de la secuencia (pero por lo general una parte de esa secuencia a la que corresponden los datos en el bloque de datos) puede estar asociada con uno de estos valores y de tal manera que la suma total de los errores (esta es la diferencia absoluta de cada elemento de la secuencia de Errores con su mejor (más cercano) valor aproximado de Error') es tan pequeña como sea posible. Se pueden utilizar otros factores de "ponderación" en lugar del valor absoluto, como el cuadrado de este valor absoluto o una definición que tenga en cuenta características perceptivas de audio. La búsqueda de números K de una serie de valores - en este caso se define como Errores debidos a la reducción de frecuencia de muestreo de los 2 canales mezclados - se define como el objetivo de K-Medias. Deben agruparse grupos de elementos de la secuencia de Errores y los K centros deben ser identificados de forma que la suma de las distancias desde cada punto a su más cercano se minimice. Problemas similares y sus soluciones también se conocen en la literatura tales como los algoritmos de Localización de Instalaciones. Además, dentro de este contexto se deben considerar soluciones de "streaming" (transmisión secuencial), así como las soluciones de transmisión no secuencial. El primero significaría que el "codificador" sólo tiene un momento y un pase de acceso a los Errores generados en vivo (y en tiempo real) resultantes de la mezcla de las secuencias de audio en vivo. Este último (no secuencial) significaría que un codificador tiene acceso "fuera de línea" y continuo a los datos que necesita para su procesamiento. Debido a la estructura de la secuencia de salida de datos digitales (una secuencia de audio PCM con muestras de audio de 18 bits y datos de 6 bits) un bloque de datos de la zona de datos auxiliares se envía antes que las muestras de audio a las que corresponde, se crea una situación de no transmisión secuencial en caso de algoritmos K-Medias o Localización de Instalaciones. El objetivo de esta invención no es definir un nuevo algoritmo de Agrupamiento de Datos, ya que muchos de estos están disponibles en la literatura de dominio público, sino más bien hacer referencia a estos como una solución para que un experto lo implemente. [Véase por ejemplo Agrupamiento de Secuencias de Datos (*Clustering Data Streams*): Teoría y Práctica, IEEE TRANSACCIONES DEL CONOCIMIENTO e INGENIERÍA DE DATOS, VOL. 15, n.º. 3, MAYO/JUNIO de 2003]. Una vez que se han definido estos K centros o aproximaciones de error, se genera una lista en la que los L elementos de las secuencia de errores de la mezcla se sustituyen por L referencias a elementos de esa tabla, que contienen las K aproximaciones (o centros). Como hay disponibles 6 bits de datos para cada muestra de audio, se puede - para una determinada sección de una secuencia de Error - definir K = 64 aproximaciones diferentes para todos los diferentes Errores en esa sección. A continuación se puede confiar en la compresión sin pérdida de esa lista de L referencias, de tal manera que después de la compresión se termina con M muestras de datos x 6bit y N muestras de los datos "libres" de 6bit con  $L = M + N$ . El espacio libre de la zona de datos auxiliares se utilizaría para almacenar las aproximaciones de Errores, así como el patrón de sincronización, la longitud del bloque de datos, etc. Sin embargo, dado que los valores de esta lista de L referencias podría ser una serie de números aleatorios verdaderos, no se debe confiar en la compresión de esta lista, sino más bien garantizar que esta lista sea compresible. Por lo tanto, en un caso de X/Y/Z que en este ejemplo X=24, Y=18, Z=6, se utilizan no más de  $32 = 2(Z-1)$  aproximaciones. Como tal, sólo se necesitan (Z-1) bits para hacer referencia a esta tabla y puede comprobarse fácilmente que esta lista de referencias es compresible;  $5 * 6\text{bit}$  muestras de datos pueden contener 6 referencias a esta tabla (necesitando cada una 5 bits). En el caso de 24/18/6, como se explica en la sección de la técnica básica, se necesita por lo menos un total de 86 muestras de datos para almacenar todos los datos sin incluir la lista de referencias. (6 muestras (6bit) para la sincronización, 2 muestras (6bit) para la longitud del bloque de datos, 2 muestras (6bit) para el desplazamiento, 6 muestras (6bit) para dos duplicados de muestras de audio cada una de 18 bits, 2 (6bit) para la Atenuación, 2 (6bit) datos a definir, a lo sumo 64 muestras (6bit) para 32 aproximaciones de error si es incompresible, 2 muestras (6bit) para CRC). Dada una relación de compresión de por lo menos 6 comprimida a 5 (entregando 1 muestra de datos libre), como mucho se necesitan  $6 * 86 = 516$  muestras. Este total también define la longitud máxima de un bloque de datos para este modo de 24/18/6. Restringir el número de aproximaciones, por ejemplo a 16, conduce a una reducción del total de 86 a 54, la relación de compresión mínima de la lista de referencia de por lo menos 6 comprimida a 4 y la longitud máxima del bloque de datos de  $3 * 54 = 162$  muestras de los datos. O bien, mediante la ampliación de la anchura de los errores de  $3 * 6\text{bit}$ , la creación de 118 muestras de datos para almacenar todos los datos excepto la lista de referencias. (Esto requeriría un total de  $708 = 6 * 118$ ). Sin embargo, en la mayoría de los casos una mayor compresión que comprima aún más estos datos es realista ya que lo anterior sólo considera el peor escenario posible; por ejemplo, una compresión en un 25% (4 bits reducidos a 3 bits) que es una relación típica para la tabla de aproximación de errores. Para una aproximación con 32 aproximaciones de error, esta relación extra reduciría la longitud del bloque de datos en más de un 50%; las 64 muestras de datos de las (32) aproximaciones de error se reducirían a 48 muestras de datos, de tal manera que el total (sin la lista de referencias) se reduce a 70. Más aún una compresión adicional de 20%-25% en la lista de referencias, comprimiría la lista de 6 bits a 5 bits, aún más hasta 4 bits, lo que da como resultado un total de longitud de bloque de datos de  $3 * 70 = 210$  muestras de datos. El resultado es que la secuencia de errores de 210 Errores de la reducción de muestras de las secuencias de audio mezcladas, se puede aproximar mediante una secuencia de referencias a 32 aproximaciones de Errores. Para un caso 24/18/6 con sólo 16 aproximaciones de Errores, y tomando relaciones de compresión comparables, da como resultado una secuencia de errores que necesitan  $3 * 46 = 138$  muestras de datos.

Para concluir - basado en los ejemplos anteriores - pero no limitado a estos ejemplos - el esquema de compresión introducido aquí, permite que la secuencia de errores sea aproximada de tal manera que esta aproximación puede tenerse en cuenta en el momento de mezclar las secuencias de audio de frecuencia de muestra reducida, lo que reducirá sustancialmente los errores debidos a esta reducción de la frecuencia de muestras. El uso de estas aproximaciones de errores comprimidas permite la reconstrucción de las dos secuencias PCM mezcladas con notable precisión, lo que hace en gran medida imperceptible el error introducido por la combinación y el descifrado de las dos secuencias PCM.

Además, es necesario que el descodificador reciba la información de los bloques de datos antes de procesar las muestras de audio mezcladas, ya que tiene que descodificar los bloques de datos (incluyendo la descompresión) y es necesario el acceso a estos errores (aproximados) con el fin de llevar a cabo la operación de separación. Como tal, en una primera fase de esta etapa de codificación, también se necesita un segundo bloque de un número de U muestras (= una sección) de la secuencia A<sup>ci</sup> 165a y B<sup>ci</sup> 165b y de la secuencia de Errores E 167. Las muestras de la secuencia de Errores (de ese 2º bloque) se aproximarán (utilizando algoritmos K-Medias o Localización de Instalaciones) con una tabla que contiene V (=32) aproximaciones de 12 bits y una lista de referencias para vincular cada muestra de esa sección de secuencia de Errores a un elemento de esa tabla de aproximación. Esta lista de referencias forma la secuencia aproximada de Errores E' 162.

En la etapa de combinación <6> las secuencias (A<sup>ci</sup>, B<sup>ci</sup>, E') se mezclan mediante un elemento de combinación/elemento para dar formato. Este elemento de combinación/elemento para dar formato comprende además un analizador de recortes para realizar una comprobación final para evitar los recortes (rebasado de audio) - esta comprobación podrá necesitar cambios de menor importancia. El elemento de combinación/elemento para dar formato añade datos adicionales tales como la atenuación, los valores semilla y aproximaciones de errores en la zona de datos auxiliares del bloque de datos apropiado en la secuencia de datos combinada creada por los reductores de tamaño de muestra, y proporciona la secuencia de salida 169 que comprende las secuencias combinada, la sección de bloque de datos fusionada con muestras de audio a una salida del codificador.

Reducción de errores que se introduciría por recortes. Otro aspecto de esta invención es el procesamiento previo de las secuencias de audio antes de ser mezcladas realmente. Dos o más secuencias podrían generar recortes cuando estas señales se mezclan entre sí. En tal caso, una etapa de procesamiento previo incluye un limitador/compresor dinámico de audio en uno de los canales que se mezclan o incluso en ambos canales. Esto puede hacerse aumentando gradualmente la atenuación antes de estos acontecimientos específicos, y después de esos acontecimientos disminuyendo gradualmente la atenuación. Este enfoque se aplicaría principalmente en un modo de transmisión no secuencial del procesador de codificación, ya que requiere (antes de tiempo) los valores de muestra que generarían estos rebasados/recortes. Estas atenuaciones se podrían procesar en las propias secuencias de audio y, como tal, evitar los recortes saturación de una manera que - cuando están separadas - estos efectos del compresor seguirán siendo parte de las secuencias separadas. Además de evitar recortes del audio (mezclado), la grabación de audio mezclada reduciendo (*down-mixed*) de 3D a 2D tiene que ser utilizable cuando no haya descodificador (como se describe en esta invención). Por esa razón, se utiliza una compresión (o atenuación) dinámica de señal de audio en la secuencia de audio mezclada para reducir la posibilidad de que el audio adicional (de la 3ª dimensión) interfiera demasiado con el audio básico de 2 dimensiones, pero por el almacenamiento de estos parámetros de atenuación se puedan realizar las operaciones inversas después de la separación de manera que restablecen los niveles adecuados de señal. Como se mencionó anteriormente, la estructura del bloque de datos de la zona de datos auxiliares formada por los bits menos significativos de las muestras contiene una sección para contener este parámetro dinámico de compresión (atenuación) de audio de por lo menos 8 bits. Además, a partir del análisis (véase Corrección de Errores por Reducción de Frecuencia de Muestreo), se puede concluir que la longitud máxima de un bloque de datos para un caso típico de 24/18/6 con una tabla de errores de 32 elementos y un ancho de error de 12 bits sería aproximadamente de 500 muestras. Con una tasa de muestreo de 96 kHz una sección así es de unos 5 ms. de audio, que se convierte de este modo en el grado de detalle temporal de los parámetros de atenuación. El propio valor de atenuación se representa con un valor de 8 bits, cuando diferentes niveles de atenuación dB se asignan a cada valor (por ejemplo: 0 = 0 dB, 1 = (-0,1) dB, 2 = (-0,2) dB...) se puede confiar en estos valores y las etapas temporales, para implementar una curva de compresión suave, que se puede utilizar a la inversa durante la operación de descodificación para restablecer los niveles adecuados de señal relativa. El almacenamiento de valores de atenuación en los bits menos significativos de una secuencia de audio, por supuesto, también puede aplicarse a una sola secuencia en la que algunos bits de resolución, en este caso, se sacrifican para aumentar la gama dinámica global de la señal en la secuencia. Como alternativa, en una secuencia mezclada se pueden almacenar varios valores de atenuación en el bloque de datos de manera que cada secuencia de datos tenga un valor de atenuación asociado, definiendo de este modo los niveles de reproducción para cada señal individualmente, pero conservando la resolución incluso en los niveles de señal bajos para cada señal. Además los parámetros de atenuación pueden utilizarse para mezclar información de audio de 3 dimensiones, de tal manera que los consumidores que no utilizan esta información de audio de 3 dimensiones no oyen la señal de audio adicional de 3 dimensiones, ya que esta señal adicional se atenúa en relación con la señal principal de 2 dimensiones, a la vez que conocer el valor de atenuación permite a un descodificador que recupera la señal adicional de 3 dimensiones restaurar el componente de la señal atenuada de la 3ª dimensión a su nivel de señal original. Normalmente, esto requiere que una secuencia de audio de 3ª dimensión sea atenuada por ejemplo en 18 dB antes de mezclarla en la secuencia de audio PCM de 2 dimensiones para evitar que esta información de audio

"domine" a la secuencia de audio PCM "normal". Esto requiere un parámetro adicional de (8 bits) para definir la atenuación (por cada sección de la secuencia - se define como la longitud del bloque de datos) utilizada en una secuencia de audio de 3 dimensiones antes de que se mezcle con la otra secuencia. La atenuación de 18 bits se puede invalidar después de la descodificación mediante la amplificación de la secuencia de audio de 3ª dimensión.

5 La figura 18 muestra un Dispositivo Codificador AUROFÓNICO

El dispositivo Codificador AUROFÓNICO 184 comprende varios ejemplos de codificador AURO 181, 182, 183, cada uno mezclando 1 o más canales de audio PCM utilizando la técnica descrita en las figuras 1-17. Por cada canal de salida Aurofónica se activa un ejemplo de codificador AURO 181, 182, 183. Cuando sólo se proporciona 1 canal no hay nada que mezclar y el ejemplo de codificador no debe activarse.

10 Las entradas del Codificador Aurofónico 184 son múltiples canales de audio (PCM) (canal de Audio 1 a canal de audio X). Para cada canal, se adjunta información (pos/atenuación) acerca de su posición (3D) y su atenuación utilizada cuando se mezcla reduciendo (*down-mixed*) en canales menores. Otras entradas del codificador Aurofónico consisten en la Selección de Matriz de Audio 180 que decide qué canales de audio PCM se mezclan reduciendo (*down-mixed*) en qué canales de salida Aurofónicos y el indicador de Calidad del Codificador Aurofónico que se proporciona a cada codificador AURO 181, 182, 183.

15 Canales de entrada típicos del codificador 3D son I (Frontal Izquierdo), IC (Frontal Izquierdo Centro), C (Frontal Centro), DC (Frontal Derecho Centro), D (Frontal Derecho), LFE (del inglés *Low Frequency Effects*: Efectos de Baja Frecuencia), le (Izquierda Envolvente), De (Derecho Envolvente), SI (Superior Frontal Izquierdo), SC (Superior Frontal Centro), SD (Superior Frontal Derecho), Sle (Superior Izquierdo Envolvente), SDe (Superior Derecho Envolvente), AI (Artístico-Izquierdo), AD (Artístico-Derecho)...

20 Canales de salida típicos según los proporciona el codificador y que son compatibles con un formato de reproducción de 2D son AURO-I (Izquierda) (Canal Aurofónico 1), AURO-C (Centro) (canal Aurofónico 2), AURO-D (Derecha) (canal Aurofónico...), AURO-le (Envolvente Izquierdo) (canal Aurofónico...), AURO-De (Envolvente Derecha) (canal Aurofónico...), AURO-LFE (Efectos de Baja Frecuencia) (canal Aurofónico Y).

25 Ejemplos de canales codificados con AURO según son proporcionados por la salida del codificador 184: (AURO-I, AURO-D, AURO-le, AURO-De). AURO-I puede contener los canales de audio PCM originales I (Frontal Izquierdo), SI (Frontal Superior Izquierdo) y AI (Artístico-Izquierdo), AURO-D sería similar pero para los canales de audio frontal derecho, AURO-le contiene los canales de audio PCM le (Envolvente Izquierdo) y Sle (Superior Envolvente Izquierdo), AURO-De los canales derechos equivalentes.

30 La figura 19 muestra un dispositivo Descodificador Aurofónico

El Descodificador AUROFÓNICO 194 comprende varios ejemplos de descodificador AURO 191, 192, 193, que separan 1 o más canales de audio PCM utilizando la técnica descrita en las figuras 5 y 10. Para cada canal de entrada AURO se activa un ejemplo de descodificador AURO 191, 192, 193. Cuando un canal AURO consiste en una mezcla de sólo 1 canal de audio, el ejemplo de descodificador no debe ser activado.

35 Las entradas del descodificador AUROFÓNICO reciben canales Aurofónicos (PCM) canal Aurofónico 1...canal Aurofónico X. Para cada canal Aurofónico 1...canal Aurofónico X, un descodificador de zona de datos auxiliares que forma parte del descodificador detectará automáticamente la presencia de los patrones de sincronización del bloque de datos AURO de los canales PCM. Cuando se detectan sincronizaciones coherentes, el descodificador AURO 191, 192, 193 empieza a separar las partes de audio de los canales AURO (PCM) y, al mismo tiempo, descomprimir (si es necesario) la Lista de Índices y la Tabla de Errores, y aplicar esta corrección a los canales de audio separados. Los datos AURO también incluyen parámetros como la atenuación (compensada por el descodificador) y la posición en 3D. La posición 3D se utiliza en la Sección 190 de Selección de Salida de audio para redirigir el canal de audio separado a la salida correcta del descodificador 194. El usuario selecciona el grupo de canales de salida de audio.

45 La figura 20 muestra un descodificador de acuerdo con la invención.

Ahora que todos los aspectos de la invención se han explicado se puede describir un descodificador, incluyendo las realizaciones ventajosas.

50 El descodificador 200 para descodificar la señal obtenida por la invención debe detectar preferiblemente de forma automática si se ha encontrado "audio" (por ejemplo, 24 bits) de acuerdo a las técnicas detalladas en las secciones anteriores. Esto se puede lograr, por ejemplo, mediante un detector de sincronización 201 que busca en la secuencia de datos recibidos un patrón de sincronización en los bits menos significativos. El detector de sincronización 201 tiene la capacidad de sincronizarse con los bloques de datos en la zona de datos auxiliares formada por los bits menos significativos de las muestras mediante la búsqueda de los patrones de sincronización. Como se explicó anteriormente, el uso de patrones de sincronización es opcional, pero ventajoso. Los patrones de sincronización pueden, por ejemplo para un tamaño de muestra de 24 bits, ser muestras de 2, 4, 6 u 8 bits (Z-bits) de ancho, y 2, 4, 6 u 8 de largo. (2 bits: LSB = 01, 10; (4 bits: LSB = 0001, 0010, 0100, 1000; 6 bits: 000001,....

100000; 8 bits: 00000001, ..., 10000000). Una vez que el detector de sincronización 201 ha encontrado alguno de estos patrones de coincidencia, "espera" hasta que se detecta un patrón similar. Una vez que se ha detectado el patrón, el detector de sincronización 201 se pone en un estado-candidato-SINCR. Basándose en el patrón de sincronización detectado el detector de sincronización 201 también puede determinar si se utilizaron 2, 4, 6 u 8 bits por muestra para la zona de datos auxiliares.

En el 2º patrón de sincronización, el descodificador 200 explorará por el bloque de datos para descodificar la longitud del bloque y verificará con el siguiente patrón de sincronización si hay una coincidencia entre la longitud del bloque y el inicio del siguiente patrón de sincronización. Si éstos dos coinciden, el descodificador 200 se pone en estado-Sincr. Si esta prueba falla, el descodificador 200 reiniciará su proceso de sincronización desde el principio. Durante la operación de descodificación, el descodificador 200 siempre compara la longitud del bloque con el número de muestras entre el inicio de cada bloque de sincronización sucesivo. Tan pronto como se detecta una discrepancia, el descodificador 200 sale del estado-sincr y el proceso de sincronización tiene que empezar de nuevo.

Como se explica en la figura 15 y la 16, se puede aplicar un código de corrección de errores a los bloques de datos de la zona de datos auxiliares para proteger los datos presentes. Este código de corrección de errores también se puede utilizar para la sincronización si se conoce el formato de los bloques de código de corrección de errores, y se conoce la posición de los datos auxiliares en los bloques de código de corrección de error. Por lo tanto, en la figura 20, el detector de sincronización y el detector de error se muestran como si estuvieran combinados en el bloque 201 para mayor comodidad, pero también pueden implementarse por separado. El detector de errores calcula el valor de CRC (utilizando todos los datos de este bloque de datos, excepto los sincron) y compara este valor de CRC con el valor que se encuentra al final del bloque de datos. Si hay una discrepancia, el descodificador se dice que está en estado de Error-CRC. El detector de errores de sincronización proporciona información al recuperador 202 de valores semilla, el recuperador 203 de aproximaciones de error y el controlador auxiliar 204 permiten que el recuperador 202 de valores semilla, el recuperador 203 de aproximaciones de error y el controlador auxiliar 204 extraigan los datos relevantes de la zona de datos auxiliares tal como se reciben de la entrada del descodificador 200. Una vez que el detector de sincronización se sincroniza con las cabeceras de los bloques de datos, el recuperador de valores semilla explora los datos en el bloque de datos para determinar el desplazamiento, es decir, el número de muestras entre el final de un bloque de datos y la primera muestra duplicada de audio (este número podría teóricamente ser negativo) y para leer estas muestras duplicadas (de audio).

El recuperador 202 de valores semilla recupera uno o más valores semilla de la zona de datos auxiliares del conjunto de datos digitales recibidos y suministra los valores semilla recuperados al descifrador 206. El descifrador 206 realiza el descifrado básico de los conjuntos de datos digitales utilizando el valor(es) semilla como se explica en la figura 5 y la 9. El resultado de este descifrado son varios conjuntos de datos digitales o un solo conjunto de datos digitales con uno o más conjuntos de datos digitales eliminados del conjunto de datos combinados digitales. Esto se indica en la figura 20 mediante las tres flechas que conectan el descifrador 206 a las salidas del descodificador 200.

Como se explicó anteriormente, el uso de las aproximaciones de errores es opcional, ya que el audio tal como se descifra mediante el descifrador 206 es ya muy aceptable sin necesidad de utilizar las aproximaciones de errores para reducir los errores introducidos por la igualación realizada por el codificador. El recuperador 203 de aproximaciones de errores descomprimirá la lista de referencia y la tabla de aproximaciones, si es necesario. Si se van a utilizar las aproximaciones de errores para mejorar el conjunto o conjuntos de datos digitales descifrados, el descifrador 206 aplica las aproximaciones de errores recibidas del recuperador 203 de aproximaciones de errores a los correspondientes conjuntos de datos digitales y suministra el conjunto(s) de datos digitales resultantes a la salida del descodificador. Siempre que el descodificador 200 permanezca sincronizado con las cabeceras de los bloques de datos, el recuperador 203 de aproximaciones de errores seguirá descomprimiendo las listas de referencias y las tablas de aproximaciones, y suministrará estos datos al descifrador 206 para separar las muestras de audio mezcladas de acuerdo a  $C = A+B+E$  o  $C-E = A+B$ . El descifrador 206 utiliza las muestras de audio duplicadas para iniciar la separación en las muestras "A" y muestras "B". Para un conjunto de datos digitales combinado en el que se han combinado dos conjuntos de datos digitales, las muestras indexadas pares de  $A'_{2i}$  coinciden con estas de  $A'_{2i}$  y  $A'_{2i+1}$  se corrigen mediante la adición de un  $E'_{2i+1}$ . Del mismo modo, las muestras indexadas impares de  $B'_{2i+1}$  coinciden con las de  $B'_{2i+1}$  y  $B'_{2i+2}$  se corrigen mediante la adición de  $E'_{2i+2}$ . La atenuación inversa se aplica a la segunda secuencia de audio (B) y las dos muestras de audio (A' y B') se convierten a su anchura de bits original mediante el traslado de estas muestras de Z bits a la izquierda, mientras que se rellenan con ceros en el lado de bits menos significativos. Las muestras reconstruidas se envían como secuencias de audio independientes no relacionadas.

Otro elemento opcional del descodificador 200 es el controlador auxiliar 204. El controlador auxiliar 204 recupera los datos de control auxiliares de la zona de datos auxiliares y procesa los datos de control auxiliares recuperados y proporciona el resultado, por ejemplo en forma de datos de control a sistemas de accionamiento mecánicos de control, instrumentos musicales o luces, a una salida auxiliar del descodificador. De hecho, el descodificador puede ser despojado del descifrador 206, el recuperador 202 de valores semilla y el recuperador 203 de aproximaciones de errores en caso de que el descodificador sólo tenga que proporcionar los datos de control auxiliares, por ejemplo, para a los sistemas de accionamiento mecánico de control de manera que corresponda a la secuencia de audio en el conjunto de datos digitales combinados

## ES 2 399 562 T3

- 5 Cuando el descodificador se pone en un estado de Error-CRC, el usuario puede definir el comportamiento del descodificador, por ejemplo, puede querer desvanecer la segunda salida hasta un nivel de silencio, y una vez que el descodificador se recupera desde su estado Error-CRC, desvanecer la segunda salida de nuevo. Otro comportamiento podría ser duplicar la señal mezclada en ambas salidas, pero estos cambios de audio presentados en las salidas del descodificador no deben provocar nunca chasquidos o chapoteo del audio.



**REIVINDICACIONES**

1. Un método para reducir un conjunto (20) de datos digitales de muestras ( $A_0, A_1, A_2, A_3, A_4, A_5, A_6, A_7, A_8, A_9$ ) con un primer tamaño en un segundo conjunto (40) de datos digitales de muestras ( $C_0, C_1, C_2, C_3, C_4$ ) con un segundo tamaño más pequeño que el primer tamaño, que comprende las etapas de:
  - 5 - igualar cada una muestra de un primer subconjunto de muestras ( $A_1, A_3, A_5, A_7, A_9$ ) del primer conjunto (20) de datos digitales con una muestra vecina de un segundo subconjunto de muestras ( $A_0, A_2, A_4, A_6, A_8$ ) del primer conjunto (20) de datos digitales, en el que se intercala el primer subconjunto de muestras ( $A_1, A_3, A_5, A_7, A_9$ ) y el segundo subconjunto de muestras ( $A_0, A_2, A_4, A_6, A_8$ );
  - aproximar un error para cada muestra igualada, el error es una diferencia entre la muestra del primer subconjunto de muestras ( $A_1, A_3, A_5, A_7, A_9$ ) y la correspondiente muestra igualada;
  - 10 - extraer las muestras igualadas desde el conjunto de datos digitales de muestras ( $A_0, A_1, A_2, A_3, A_4, A_5, A_6, A_7, A_8, A_9$ ) que tiene como resultado el segundo conjunto (40) de datos digitales; e
  - integrar los errores aproximados en los bits menos significativos de las muestras.
2. Un método según la reivindicación 1, en donde el primer conjunto (20) de datos digitales representa una primera señal de audio y el segundo conjunto (40) de datos digitales representa una segunda señal de audio, la segunda señal de audio es una representación de la primera señal de audio.
3. Un método según la reivindicación 1 o 2, en donde se integra un patrón de sincronización (SYNC) en los bits menos significativos de ( $C_0, C_1, C_2, C_3, C_4$ ).
4. Un método según cualquiera de las reivindicaciones precedentes, en donde se indexa el conjunto de aproximaciones de errores y un índice que representa la aproximación de error se integra en una zona (81) de datos auxiliares formada por los bits menos significativos de las muestras a las que corresponde la aproximación de errores.
5. Un método para recrear un primer conjunto (20) de datos digitales de ( $A_0, A_1, A_2, A_3, A_4, A_5, A_6, A_7, A_8, A_9$ ) a partir de un segundo conjunto (40) de datos digitales de muestras ( $C_0, C_1, C_2, C_3, C_4$ ), el segundo conjunto (40) de datos digitales se crea utilizando un método según cualquiera de las reivindicaciones precedentes, dicho método comprende las etapas de:
  - 25 - recuperar una aproximación de error para cada muestra igualada desde los bits menos significativos de las muestras ( $C_0, C_1, C_2, C_3, C_4$ ) del conjunto (40) de datos digitales;
  - recuperar un primer subconjunto de muestras ( $A_0, A_2, A_4, A_6, A_8$ ) del primer conjunto (20) de datos digitales que es igual a las muestras ( $C_0, C_1, C_2, C_3, C_4$ ) del segundo conjunto (40) de datos digitales; y
  - 30 - calcular un segundo subconjunto de muestras ( $A_1, A_3, A_5, A_7, A_9$ ) mediante la igualación de las muestras con las muestras vecinas del primer subconjunto de muestras ( $A_0, A_2, A_4, A_6, A_8$ ) y aplicando la aproximación de error a la correspondiente muestra igualada.
6. Un método según la reivindicación 5, en donde el primer conjunto (20) de datos digitales representa una primera señal de audio y el segundo conjunto (40) de datos digitales representa una segunda señal de audio, la segunda señal de audio es una representación reducida de frecuencia de muestra de la primera señal de audio.
7. Un método según la reivindicación 6, en donde se extrae un tercer conjunto (80) de datos digitales, que representa una tercera señal de audio, el tercer conjunto (80) de datos digitales comprende una combinación del primer conjunto (20) de datos digitales y el segundo conjunto (40) de datos digitales.
8. Un método según la reivindicación 7, en donde las aproximaciones de error se recuperan de una zona (81) de datos auxiliares formada por los bits menos significativos de las muestras del tercer conjunto (80) de datos digitales y se utiliza un patrón de sincronización (SYNC) para definir una posición de la primera aproximación de error relativa a un valor de semilla ( $A_0$ ).
9. Un método según cualquiera de las reivindicaciones 5 a 8, en donde tras la etapa de recuperar el primer conjunto (20) de datos digitales, un error resultante de la ecuación de la muestra durante la codificación se compensa añadiendo una aproximación de error recuperada.
10. Un codificador (10) dispuesto para ejecutar el método según cualquiera de las reivindicaciones 1 a 4, que comprende:
  - 50 - unos primeros medios de igualación (11a) para igualar cada muestra de un primer subconjunto de muestras ( $A_1, A_3, A_5, A_7, A_9$ ) del primer conjunto (20) de datos digitales con una muestra vecina de un segundo subconjunto de

- muestras ( $A_0, A_2, A_4, A_6, A_8$ ) del primer conjunto (20) de datos digitales en el que se intercala el primer subconjunto de muestras ( $A_1, A_3, A_5, A_7, A_9$ ) y el segundo subconjunto de muestras ( $A_0, A_2, A_4, A_6, A_8$ );
- un dispositivo de aproximación de errores para aproximar un error para cada muestra igualada, el error es una diferencia entre la muestra del primer subconjunto de muestras ( $A_1, A_3, A_5, A_7, A_9$ ) y la correspondiente muestra igualada;
  - unos medios para extraer el primer subconjunto igualado de muestras ( $A_1, A_3, A_5, A_7, A_9$ ) desde el conjunto de datos digitales de muestras ( $A_0, A_1, A_2, A_3, A_4, A_5, A_6, A_7, A_8, A_9$ ); y
  - unos medios para integrar los errores aproximados en los bits menos significativos del segundo subconjunto de muestras ( $A_0, A_2, A_4, A_6, A_8$ ).
- 10 11. Un decodificador dispuesto para ejecutar el método según cualquiera de las reivindicaciones 5 a 9, que comprende:
- un procesador (206) para recuperar el primer conjunto (20) de datos digitales que comprende un primer subconjunto de muestras ( $A_1, A_3, A_5, A_7, A_9$ ) y un segundo subconjunto de muestras ( $A_0, A_2, A_4, A_6, A_8$ );
  - un recuperador (203) de aproximación de error para recuperar una aproximación de error para cada muestra igualada a partir de los bits menos significativos de las muestras ( $C_0, C_1, C_2, C_3, C_4$ ) del conjunto (40) de datos digitales;
  - unos medios para recuperar un segundo subconjunto de muestras ( $A_0, A_2, A_4, A_6, A_8$ ) del primer conjunto (20) de datos digitales que son iguales a las muestras ( $C_0, C_1, C_2, C_3, C_4$ ) del segundo conjunto (40) de datos digitales; y
  - unos medios para calcular (204) un segundo subconjunto de muestras ( $A_1, A_3, A_5, A_7, A_9$ ) mediante la igualación de las muestras con las muestras vecinas del primer subconjunto de muestras ( $A_0, A_2, A_4, A_6, A_8$ ) y aplicar la aproximación de error a la correspondiente muestra igualada.
- 15 12. Un programa informático que comprende unos medios de codificación para ejecutar el método según cualquiera de las reivindicaciones precedentes 1 a 9 cuando se ejecuta en un ordenador que proporciona un entorno adecuado para la ejecución del programa informático.
- 20 13. Un conjunto reducido (40) de datos digitales según se puede obtener mediante el método de cualquiera de las reivindicaciones 1 a 4.
- 25 14. Una portadora de registros que comprende un conjunto reducido (40) de datos digitales según la reivindicación 13.
- 30 15. Un registrador que comprende un codificador según la reivindicación 10.
16. Un dispositivo de reproducción que comprende un decodificador según la reivindicación 11.

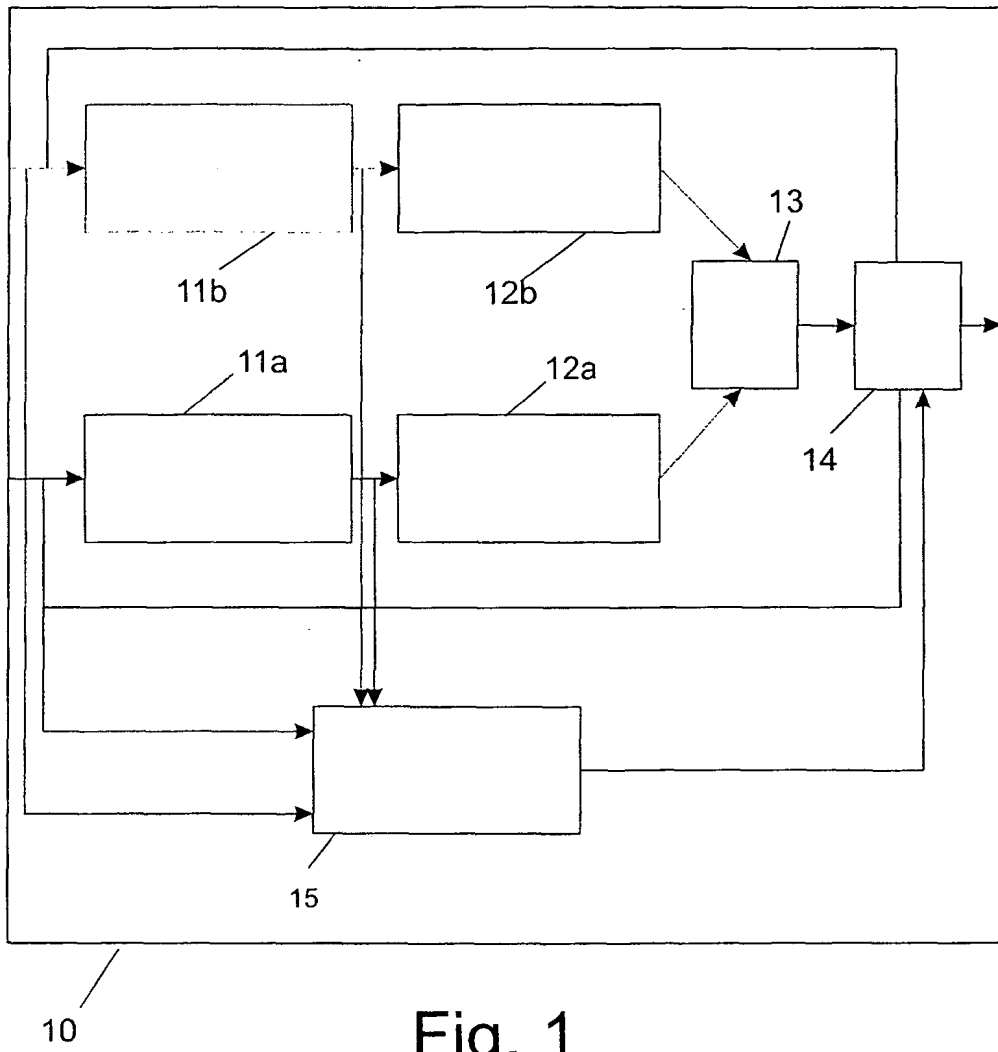


Fig. 1

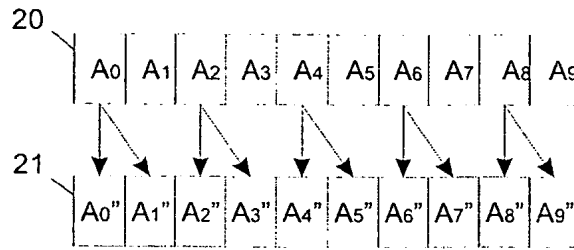


Fig. 2

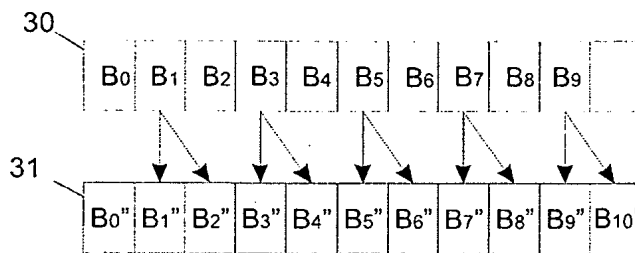


Fig. 3

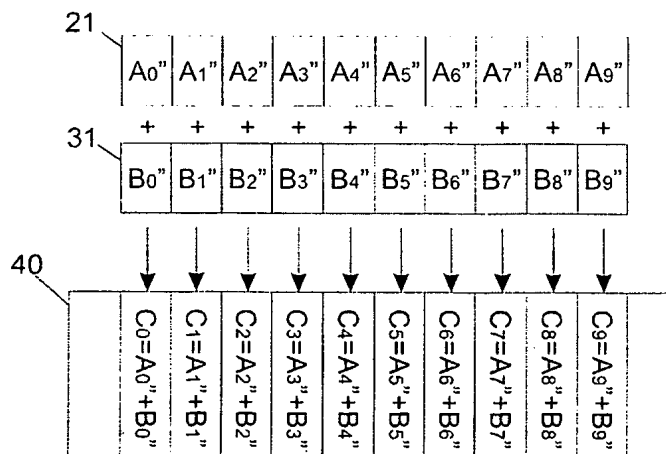


Fig. 4

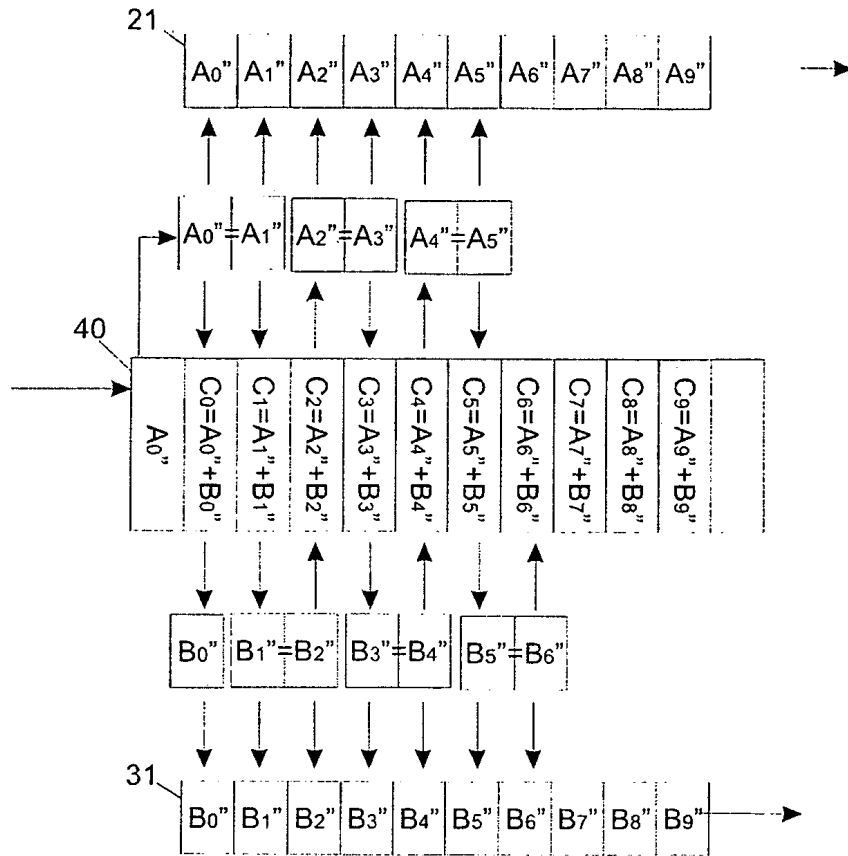


Fig. 5

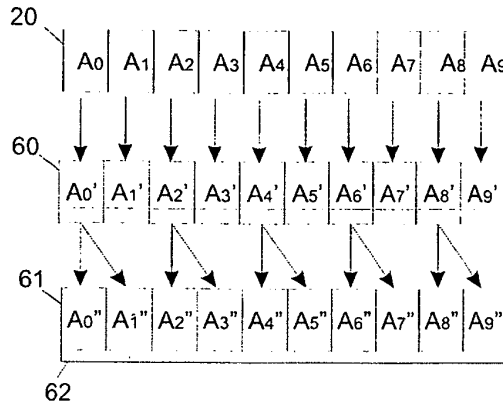


Fig. 6

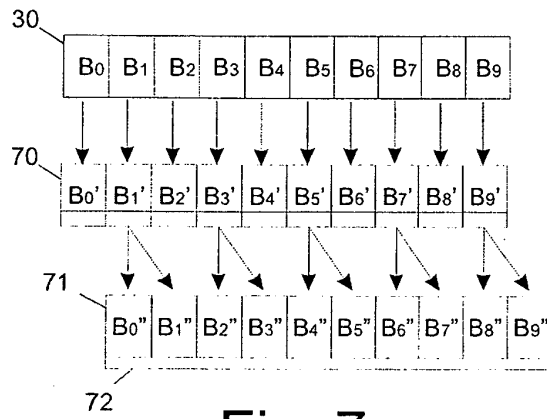


Fig. 7

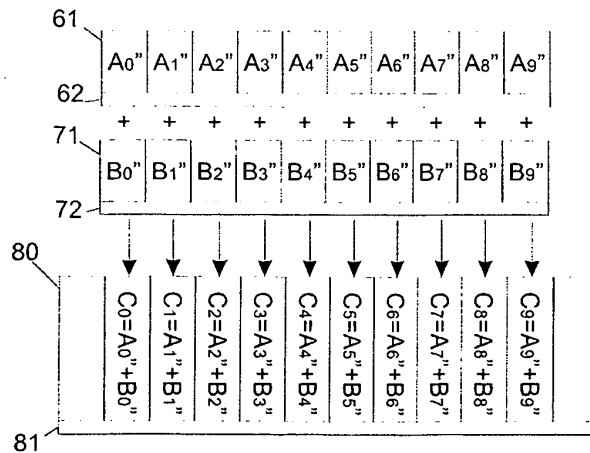


Fig. 8

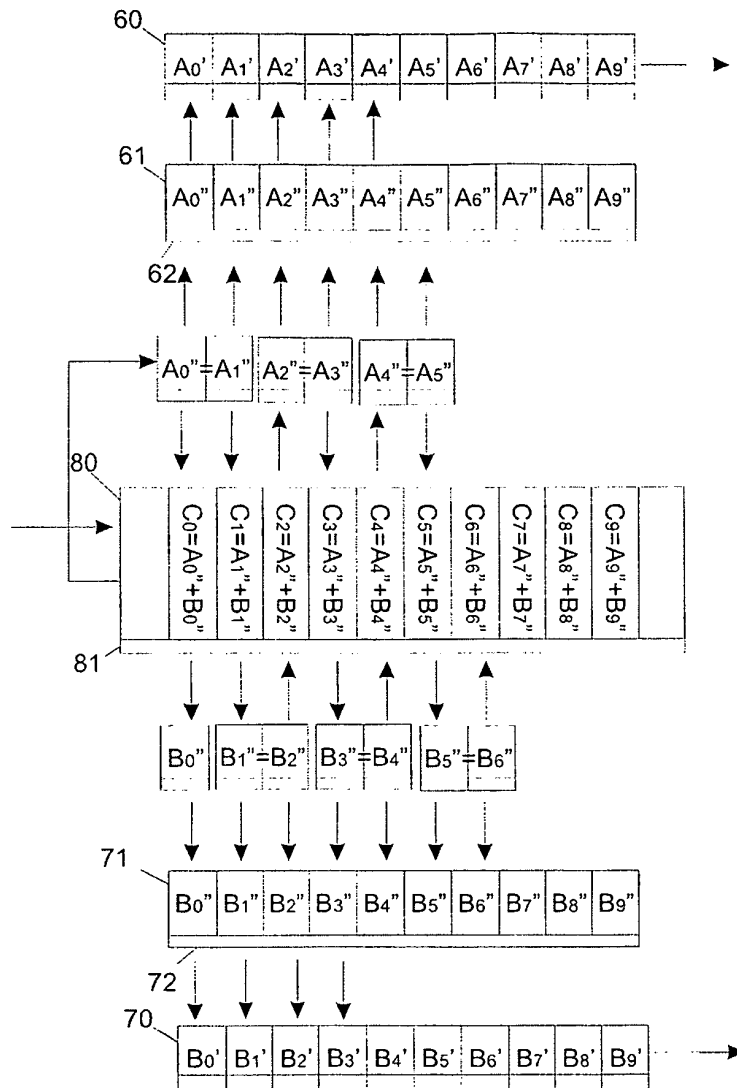


Fig. 9

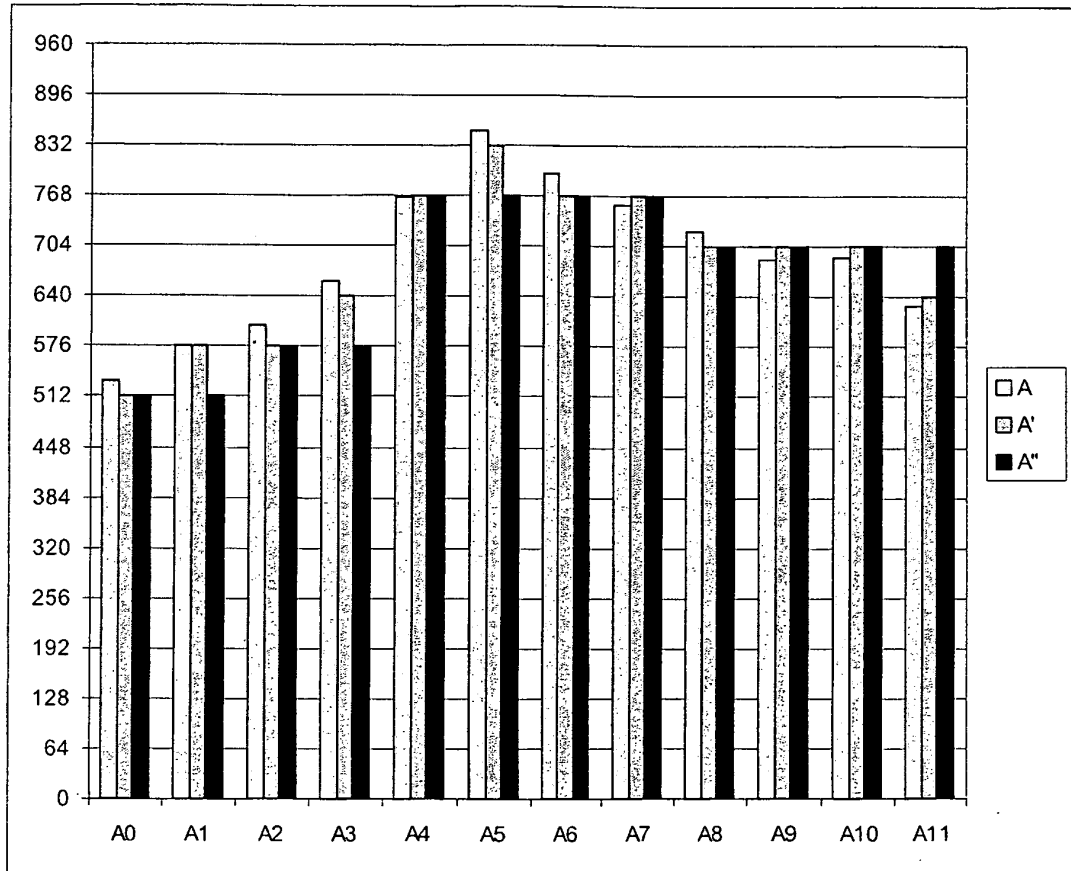


Fig. 10



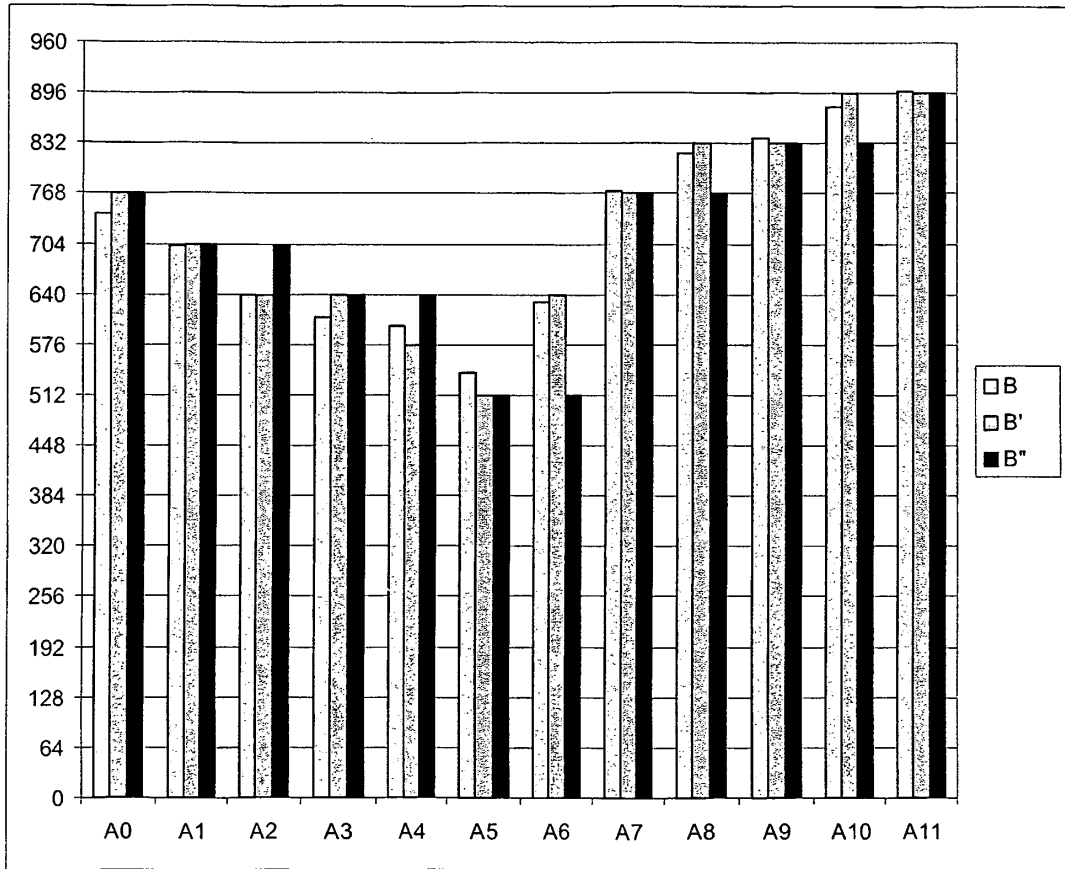


Fig. 11

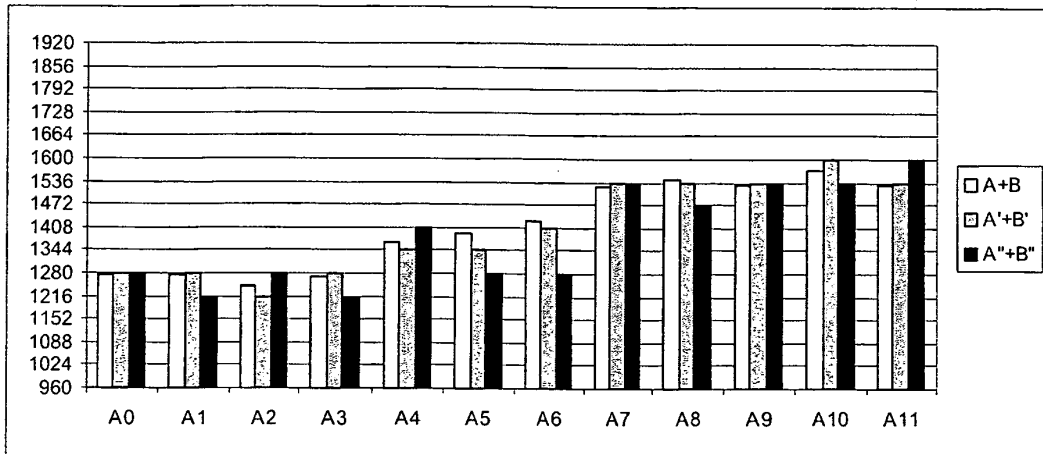


Fig. 12

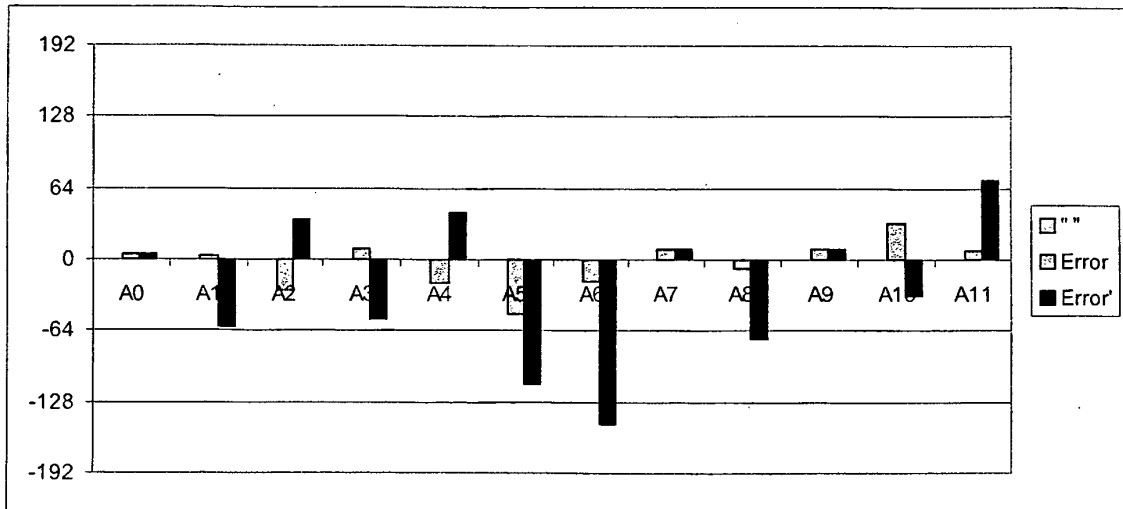


Fig. 13

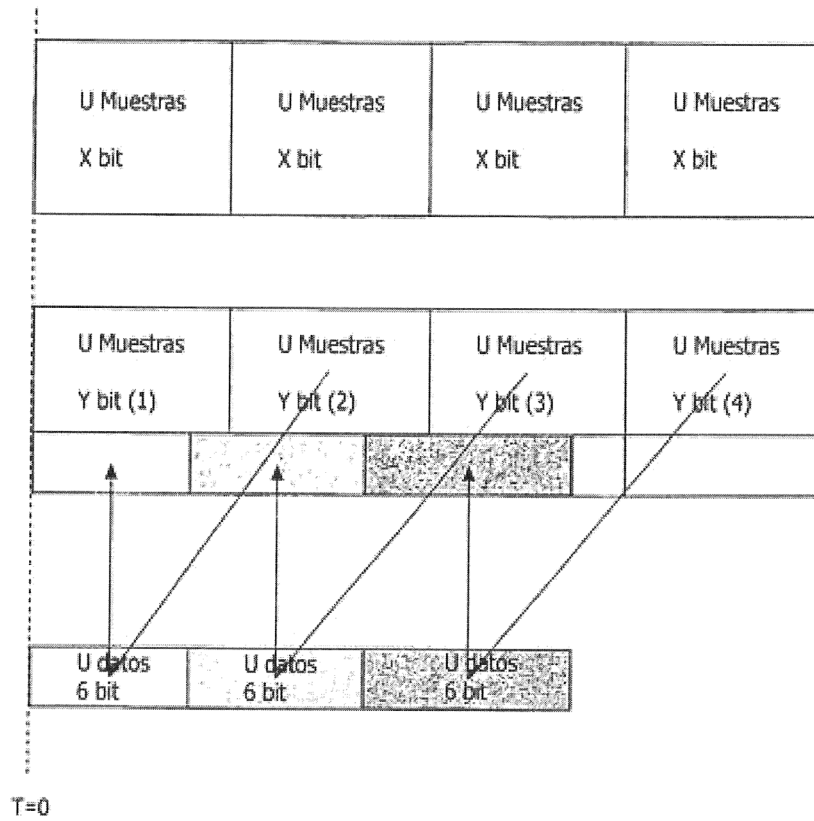
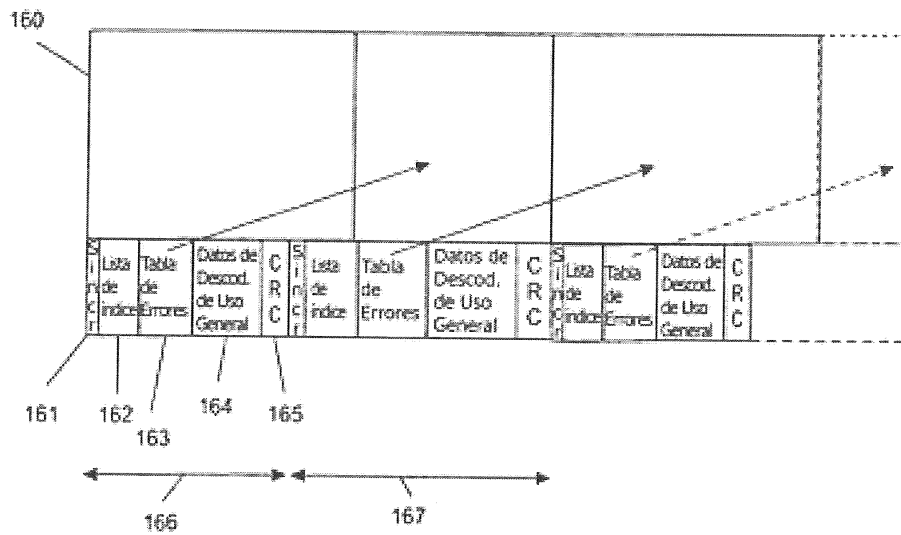
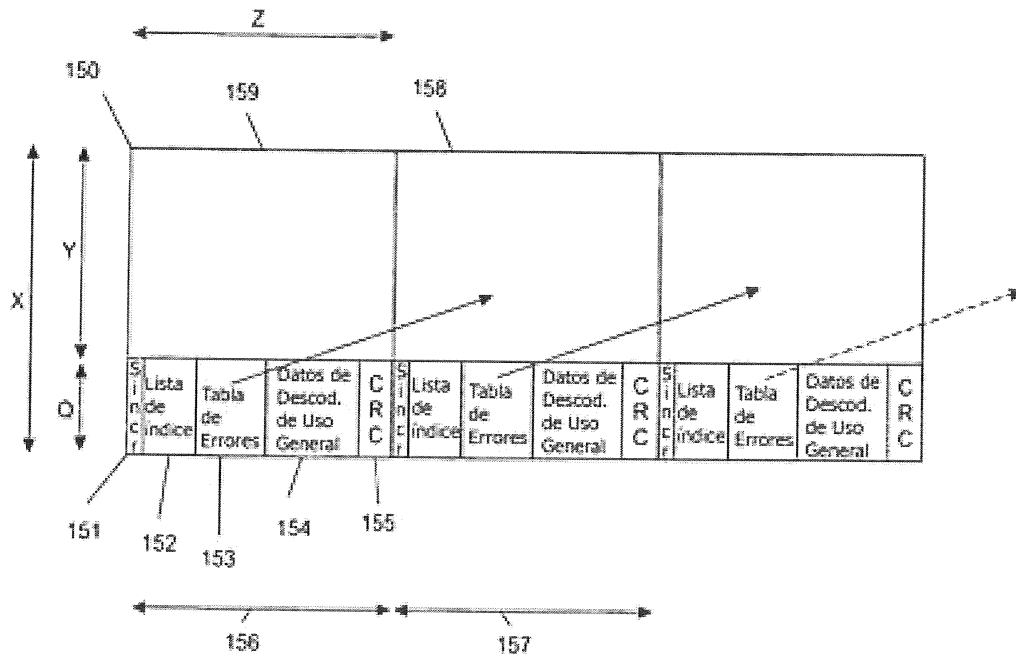


Fig. 14



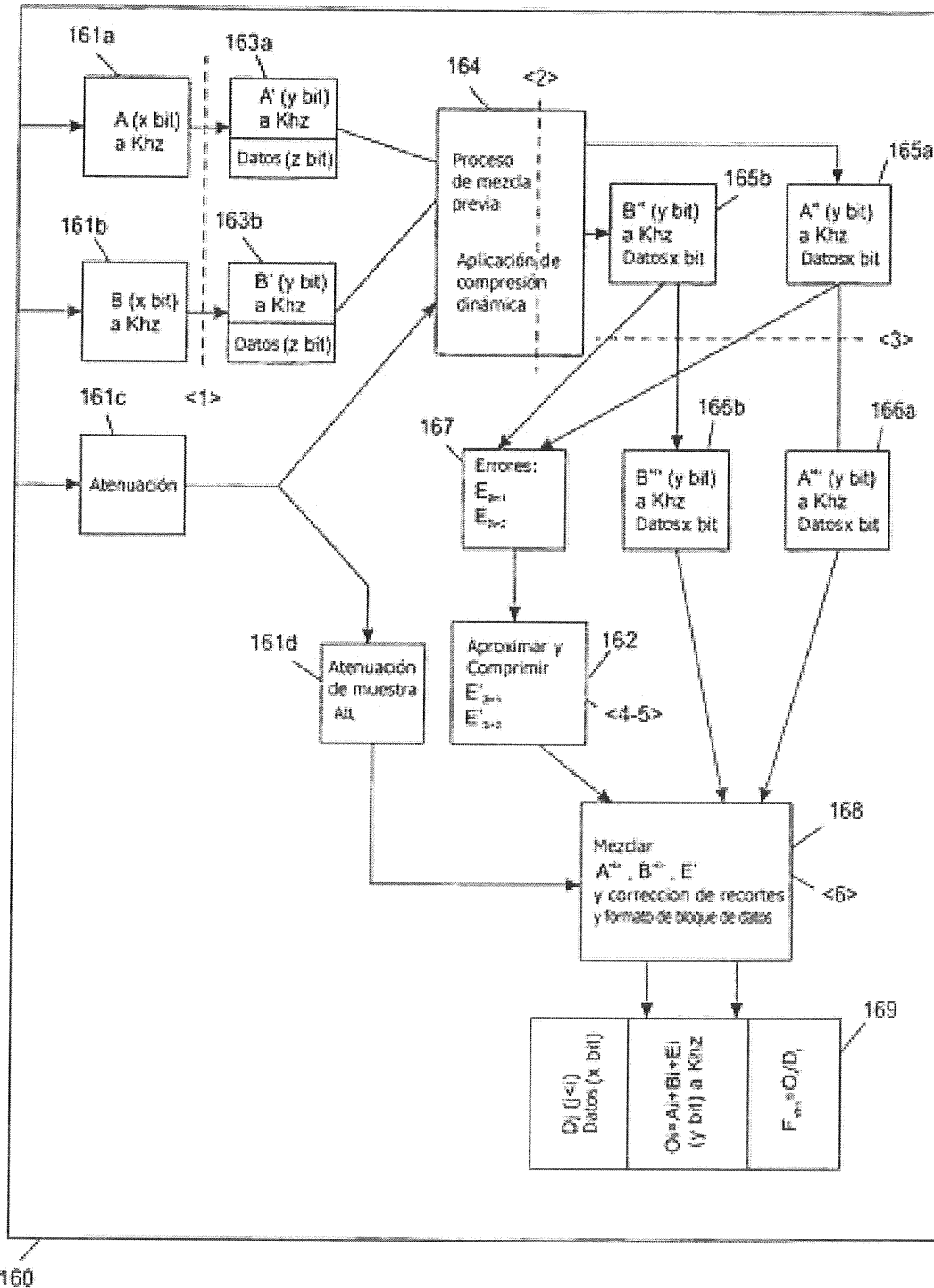


Fig. 17

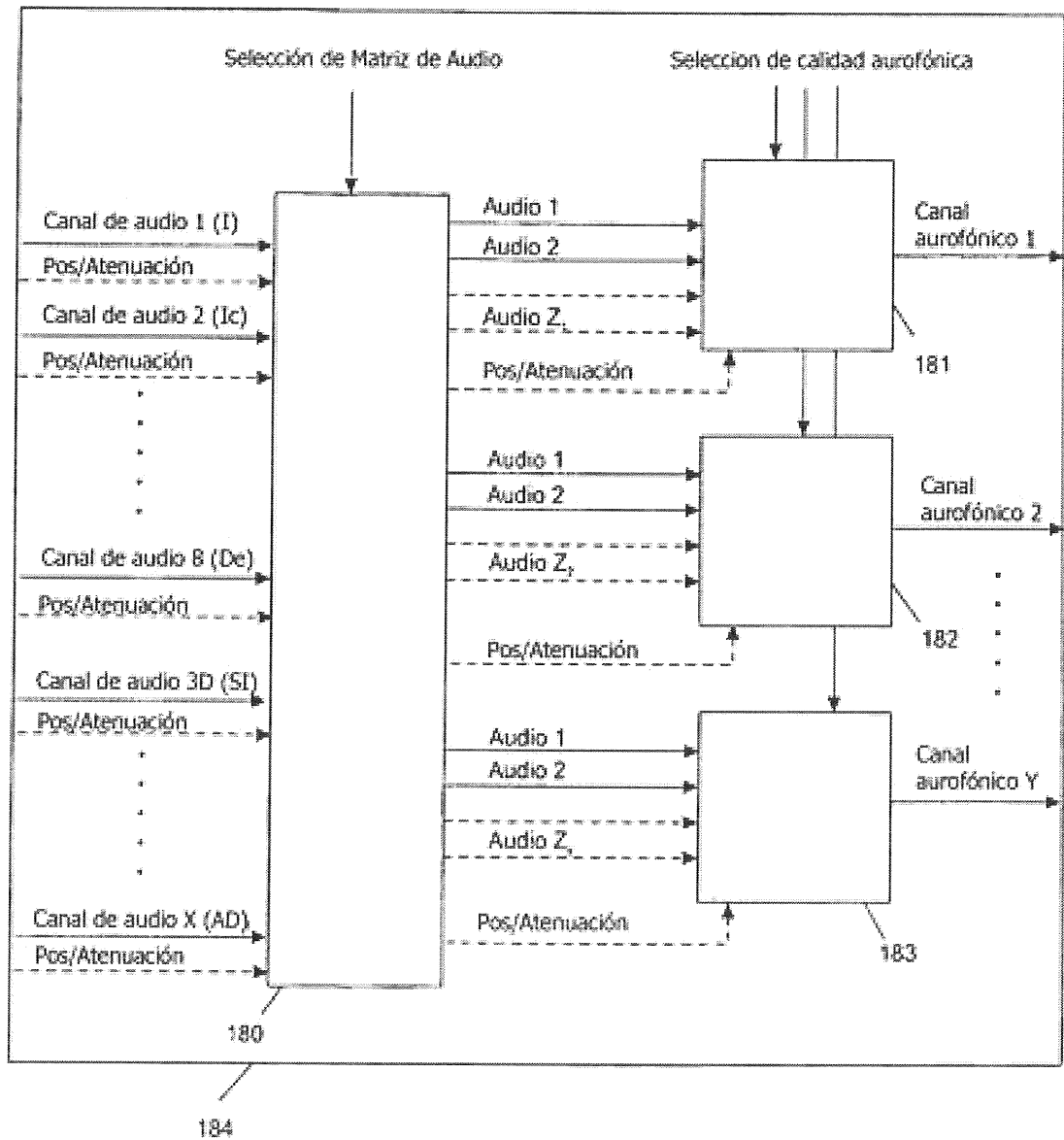


Fig. 18

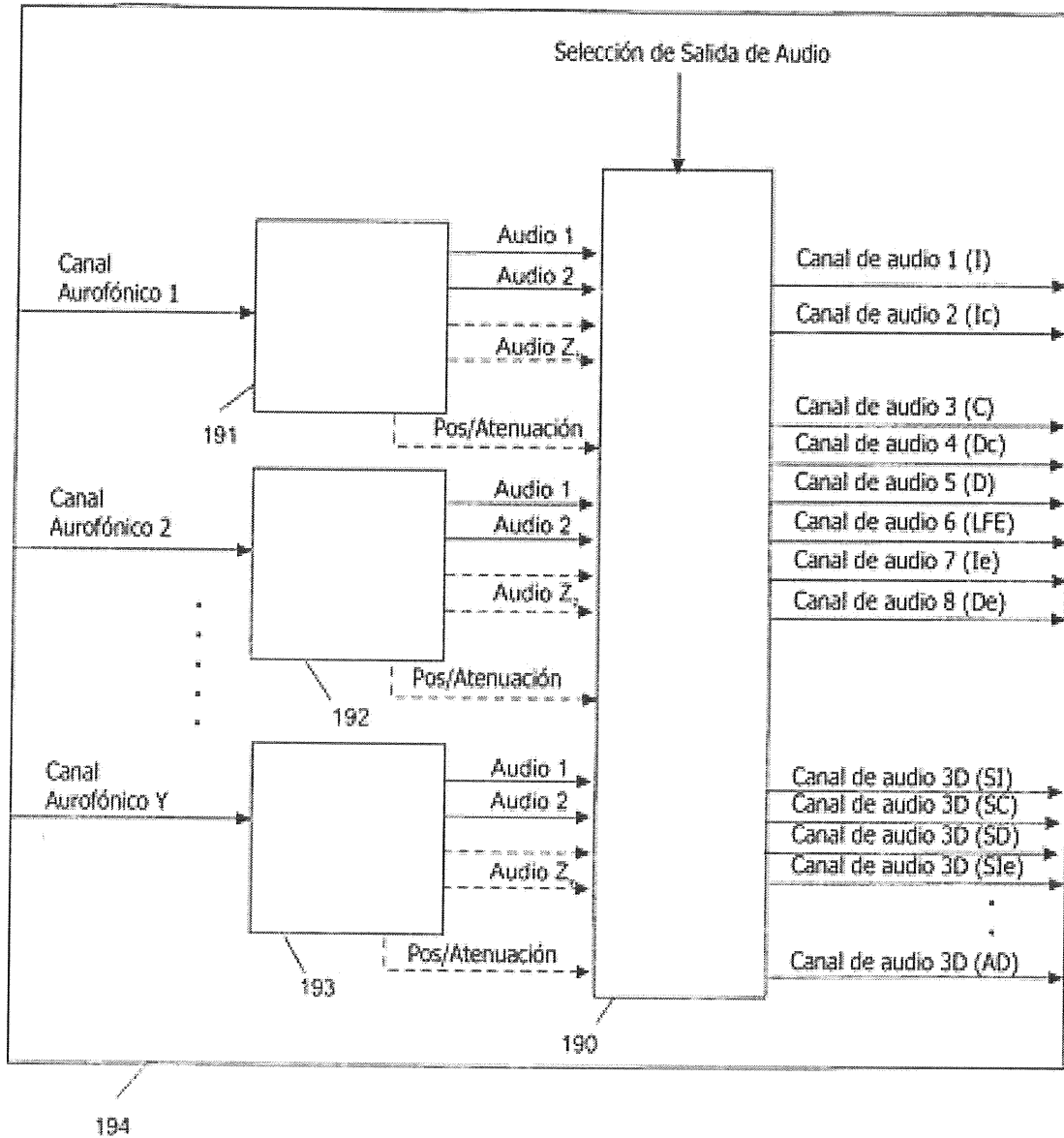


Fig. 19

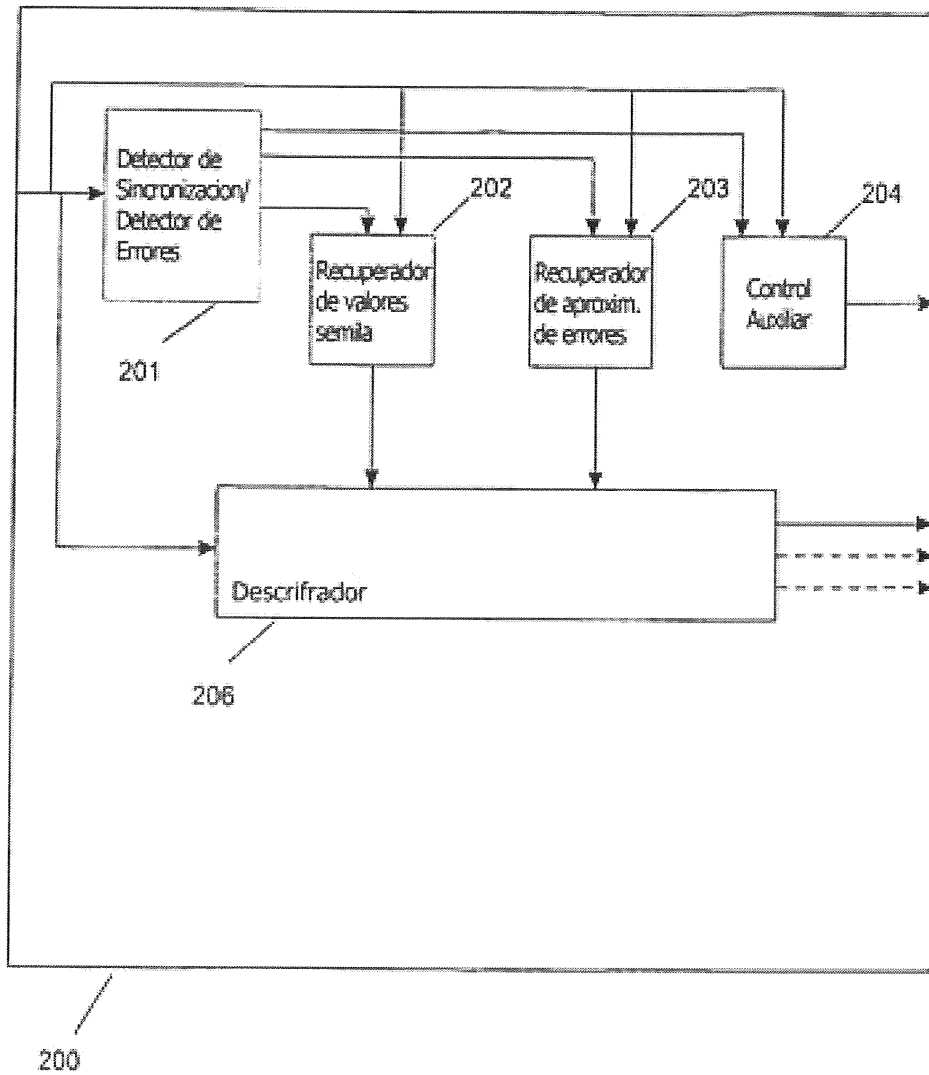


Fig. 20