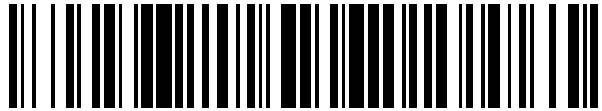


19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 693 559**

51 Int. Cl.:

G10L 19/02 (2013.01)

G06F 17/14 (2006.01)

G10L 21/038 (2013.01)

G10L 19/028 (2013.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

86 Fecha de presentación y número de la solicitud internacional: **22.08.2014 PCT/EP2014/067945**

87 Fecha y número de publicación internacional: **26.02.2015 WO15025052**

96 Fecha de presentación y número de la solicitud europea: **22.08.2014 E 14758321 (5)**

97 Fecha y número de publicación de la concesión europea: **18.07.2018 EP 3036738**

54 Título: **Aparato y procedimiento para procesar una señal de audio mediante una señal de error de aliasing**

30 Prioridad:

23.08.2013 EP 13181507

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

12.12.2018

73 Titular/es:

**FRAUNHOFER-GESELLSCHAFT ZUR
FÖRDERUNG DER ANGEWANDTEN
FORSCHUNG E.V. (100.0%)
Hansastraße 27c
80636 München, DE**

72 Inventor/es:

**DISCH, SASCHA;
NAGEL, FREDERIK;
GEIGER, RALF;
NEUKAM, CHRISTIAN y
EDLER, BERND**

74 Agente/Representante:

SALVÀ FERRER, Joan

ES 2 693 559 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

Aparato y procedimiento para procesar una señal de audio mediante una señal de error de *aliasing*

5 **[0001]** La presente invención está relacionada con el procesamiento de audio y particularmente con el procesamiento de audio en el contexto del procesamiento afectado por *aliasing* de señales de audio.

[0002] Durante un funcionamiento normal, la Transformada de coseno discreta modificada (MDCT) tiene características que la convierten en una herramienta muy adecuada para aplicaciones de codificación de audio.
 10 Genera una representación de señal espectral críticamente muestreada a partir de tramas de solapamiento y proporciona una reconstrucción perfecta. Esto significa que la señal de entrada se puede reconstruir a partir de coeficientes espectrales de una transformada directa al aplicar la transformada inversa y una operación de solapamiento y suma en las regiones de solapamiento. Sin embargo, si se aplica un procesamiento adicional en los coeficientes espectrales, la MDCT presenta algunos inconvenientes en comparación con las representaciones
 15 sobremuestreadas como el procesamiento solapado basado en la DFT. Incluso un control de ganancia dependiente del tiempo y la frecuencia relativamente simple, como el utilizado para el control de rango dinámico o la prevención de recorte, puede producir efectos secundarios no deseados. Por lo tanto, el posprocesamiento individual basado en la DFT para la decodificación de audio se aplica en varias aplicaciones que requieren este tipo de modificación de señal, aunque una representación espectral basada en la MDCT estaría disponible dentro del decodificador. Como
 20 desventaja, además de la complejidad computacional, está el retraso adicional introducido por dicho posprocesamiento.

[0003] Un enfoque común para la reducción de *aliasing* de dominio del tiempo de MDCT es recrear una transformada solapada compleja modulada (MCLT) sobremuestreada. La MCLT se obtiene a partir de la
 25 combinación de la MDCT con su contraparte compleja, la Transformada de seno discreta modificada (MDST). La MCLT proporciona características similares como una representación de DFT de una señal y, por lo tanto, su solidez contra el *aliasing* de dominio del tiempo (TDA) debido a la manipulación espectral es comparable con la representación de DFT. Pero desafortunadamente, el cálculo del espectro de MDST fuera del espectro de MDCT es computacionalmente muy complejo y produce un retraso de señal significativo. Por lo tanto, el estado de la técnica
 30 proporciona técnicas para reducir tanto el retraso como la complejidad [2 - 3]. En estos enfoques, se utiliza una transformada real a compleja (R2C) para aproximarse a los valores de MDST necesarios. A continuación, en el dominio de la MCLT, se aplica la manipulación de los coeficientes espectrales. Seguidamente, los valores complejos se transforman nuevamente en el dominio de la MDCT mediante una transformada compleja a real (C2R). Aunque este enfoque ofrece buenos resultados en términos de solidez de *aliasing*, tiene algunas desventajas. En primer
 35 lugar, los coeficientes de la MDST son estimados y su precisión se define por la cantidad de complejidad computacional. En segundo lugar, la cadena de transformadas R2C-C2R sigue produciendo retraso.

[1] H. S. Malvar, "A modulated complex lapped transform and its applications to audio processing", en Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP), Phoenix, marzo de 1999.

40 [2] Kuech, F.; Edler, B., "Aliasing Reduction for Modified Discrete Cosine Transform Domain Filtering and its Application to Speech Enhancement", en IEEE Workshop on Applications of Signal Processing to Audio and Acoustics, 21-24 de octubre de 2007.

[3] Edler, B., "Aliasing Reduction for Gain Control with Critically Sampled Filter Banks", en First International Conference on Communications and Electronics, ICCE '06, 10-11 de octubre de 2006.

45 [4] E. Larsen y R. M. Aarts. Audio Bandwidth Extension - Application to psychoacoustics, Signal Processing and Loudspeaker Design. John Wiley & Sons, Ltd, 2004.

[5] M. Dietz, L. Liljeryd, K. Kjörling y O. Kunz, "Spectral Band Replication, a novel approach in audio coding", en 112th AES Convention, Múnich, mayo de 2002.

50 [6] P. Ekstrand, "Bandwidth Extension of Audio Signals by Spectral Band Replication", en Proceedings of 1st IEEE Benelux Workshop on MPCA, Leuven, noviembre de 2002.

[0004] El documento US 2010283639 A1 proporciona la codificación / decodificación de una señal digital, en particular, mediante una transformada con solapamiento empleando ventanas de ponderación. Dos bloques sucesivos e iguales de muestras de la señal son ponderados por las diferentes ventanas sucesivas respectivas.
 55 Estas dos ventanas se pueden seleccionar independientemente entre sí de acuerdo con un criterio específico para las características de la señal (entropía, velocidad de datos / distorsión, etc.) que se determinan para cada uno de los dos bloques.

[0005] El documento US 2011173009 describe un aparato para codificar una señal de audio que incluye un

enventanador para enventanar un primer bloque mediante una ventana de análisis que tiene una porción de *aliasing* y la porción adicional. El aparato comprende además un procesador para procesar un primer subbloque de la señal de audio asociada con la porción de *aliasing* mediante la transformación del subbloque desde un dominio a un dominio posterior diferente para enventanar el primer subbloque para obtener el primer subbloque procesado y para procesar un segundo subbloque de la señal de audio asociada con la porción adicional transformando el segundo subbloque desde el dominio al dominio diferente antes del enventanado del segundo subbloque para obtener un segundo subbloque procesado. El aparato comprende, además, un transformador para convertir el primer subbloque procesado y el segundo subbloque procesado desde el dominio diferente en un dominio adicional diferente mediante la misma regla de transformada de bloque para obtener un primer bloque convertido que posteriormente puede comprimirse utilizando cualquier algoritmo de compresión de datos conocido. Por lo tanto, se puede obtener un conmutador críticamente muestreado entre dos modos de codificación, ya que las porciones de *aliasing* que ocurren en dos dominios diferentes se corresponden entre sí.

[0006] El documento WO 2012085410 A1 describe un procesamiento de una señal en forma de bloques de muestra consecutivos. El procesamiento comprende el filtrado en un dominio transformado de subbandas y, particularmente, el procesamiento de ecualización, aplicado a un bloque actual en el dominio de transformada y a un procesamiento de ajuste de filtrado que se aplica en el dominio de transformada a, al menos, un bloque adyacente al bloque actual.

[0007] La recomendación G.719 de la ITU-T sobre la codificación de audio en la banda completa y de baja complejidad para aplicaciones conversacionales de alta calidad describe un codificador de audio de acuerdo con la norma internacional G.719.

[0008] Es un objetivo de la presente invención proporcionar un concepto mejorado para el procesamiento de una señal de audio.

[0009] Este objetivo se logra mediante un aparato para procesar una señal de audio de acuerdo con la reivindicación 1, un procedimiento para procesar una señal de audio de acuerdo con la reivindicación 11 o un programa informático de acuerdo con la reivindicación 12. En un aspecto de un ejemplo comparativo que no forma parte de la invención, un aparato para procesar una señal de audio que comprende una secuencia de bloques de valores espectrales comprende un procesador para procesar la secuencia de bloques de valores espectrales mediante al menos un valor de modificación de un primer bloque para obtener una primera señal resultante con *aliasing* reducido o sin *aliasing* en un margen de solapamiento y mediante al menos un segundo valor de modificación diferente a partir de un segundo bloque de la secuencia de bloques de valores espectrales para obtener la segunda señal resultante con *aliasing* reducido o sin *aliasing* en el margen de solapamiento. Seguidamente, ambas señales en el margen de solapamiento, constituyendo ambas una cierta representación de una de las mismas porciones de señal, se combinan para obtener la señal procesada para el margen de solapamiento.

[0010] Preferentemente, esta combinación se basa en un fundido cruzado desde una señal resultante a la otra señal resultante, es decir, un desvanecimiento gradual de la primera señal resultante y al mismo tiempo una aparición gradual de la segunda señal resultante.

[0011] Preferentemente, el procesador comprende un modificador para modificar el primer bloque de la secuencia mediante al menos un primer valor de modificación para obtener un primer bloque modificado, para modificar el segundo bloque de la secuencia mediante al menos un segundo valor de modificación para obtener un segundo bloque modificado, para modificar el segundo bloque mediante al menos un primer valor de modificación para obtener un tercer bloque modificado y para modificar el primer bloque mediante al menos un segundo valor de modificación para obtener un cuarto bloque modificado. Un convertidor espectral en tiempo está configurado para convertir del primer al cuarto bloque modificado en representaciones temporales de los mismos y se proporciona un solapador y sumador para solapar y sumar el primero y el tercer bloque modificado para obtener una señal resultante y para solapar y sumar representaciones temporales del segundo y cuarto bloque modificado para obtener la segunda señal resultante. Por lo tanto, se realizan básicamente dos operaciones paralelas de transformada inversa, una con el primer valor de modificación de ganancia y otra con el segundo valor de modificación de ganancia.

[0012] Cuando el valor de modificación de ganancia es solo una variante de tiempo, es decir, es contante en todo el rango de frecuencia, solo se utiliza un único valor de modificación de ganancia para la modificación de un bloque. Sin embargo, si el valor de modificación de ganancia varía en frecuencia, cada bloque de valores espectrales se modifica con una pluralidad de valores de modificación de ganancia. Un valor de modificación de

ganancia puede aplicarse a cada valor espectral individual o a un grupo de valores espectrales, tales como 2, 3, 4 o 5 valores espectrales de frecuencia adyacentes. En el caso general, sin embargo, puede existir un único valor de modificación de ganancia para todas y cada una de las líneas espectrales, se modifica un bloque de valores espectrales con un número de valores de modificación de ganancia donde el número es igual al número de valores 5 espectrales en el bloque de valores espectrales.

[0013] Por lo tanto, de acuerdo con este aspecto, se generan dos señales de salida con *aliasing* reducido o sin *aliasing* para la región de solapamiento y seguidamente, se realiza un fundido cruzado entre dichas dos salidas o 10 señales resultantes.

[0014] De acuerdo con un aspecto de la presente invención, un aparato para procesar una señal de audio comprende un procesador para calcular una señal afectada por *aliasing* mediante al menos un primer valor de modificación para un primer bloque de la secuencia y al menos un segundo valor de modificación diferente para un 15 segundo bloque de la secuencia de bloques de valores espectrales. A continuación, se estima una señal de error de *aliasing* donde esta señal de error de *aliasing* se incluye en la señal de audio afectada por *aliasing* o sometida a *aliasing*. A continuación, la señal afectada por *aliasing* y la señal de error de *aliasing* se combinan de tal manera que la señal obtenida por la combinación es una señal con *aliasing* reducido o sin *aliasing*.

[0015] En otras palabras, el otro aspecto se basa en el cálculo de una señal afectada por *aliasing* y una señal 20 de error de *aliasing* y una combinación posterior de ambas señales para obtener la señal con *aliasing* reducido o sin *aliasing*. Preferentemente, el *aliasing* se anula mediante una reconstrucción de la señal de entrada en una región de solapamiento mediante una transformada de frecuencia-tiempo adicional con ganancias iguales para ambos bloques, una multiplicación con una función de ventana o una función relacionada, una inversión de tiempo, una multiplicación con una diferencia de ganancia entre los dos valores de ganancia o modificación y una resta de la 25 salida del procesamiento con la ganancia diferente.

[0016] Preferentemente, el algoritmo de transformada utilizado es una transformada de coseno discreta modificada MDCT y la operación de transformada inversa es una transformada de coseno discreta modificada inversa. Alternativamente, pueden utilizarse otras transformadas introductoras de *aliasing* similares, tal como una 30 MDST (transformada de seno discreta modificada) o una transformada de seno discreta modificada inversa (IMDST) u otra transformada similar, en la que, en el lado de análisis, un número de muestras en una porción temporal es mayor que un número de valores espectrales o indicados de manera alternativa, en la que hay una región de solapamiento entre dos porciones temporales posteriores dando como resultado dos bloques posteriores de valores espectrales, es decir, bloques de valores espectrales que son posteriores en el tiempo. En este caso, ambos 35 bloques de valores espectrales están relacionados al menos parcialmente con una y la misma región de solapamiento, es decir, la región de solapamiento entre dichas dos porciones temporales que finalmente han traído como resultado los dos bloques de valores espectrales posteriores en el tiempo. Esto significa que, en el lado de análisis, el número de muestras de dominio del tiempo en un bloque de muestras o en una trama es mayor que el número de valores de dominio de frecuencia en un bloque de representación de dominio de frecuencia, y en el lado 40 de la síntesis, el número de muestras de dominio del tiempo sintetizado es mayor que el número de valores espectrales en un bloque utilizado para sintetizar bloques de solapamiento de muestras de dominio del tiempo.

[0017] Sin embargo, como etapa final en el lado de la síntesis, se realiza un procesamiento de solapamiento y suma de modo que, al final, un número de muestras en el margen de solapamiento es menor que el número de 45 muestras de dominio del tiempo en una trama sintetizada y es preferentemente igual al número de valores espectrales de un bloque de valores espectrales. En este último caso, se obtiene una transformada críticamente muestreada y dichas transformadas son preferidas para la presente invención, pero la presente invención también puede aplicarse a transformadas no críticamente muestreadas, aunque estas transformadas tengan algún tipo de sobrecarga adicional en comparación con transformadas críticamente muestreadas. 50

[0018] Los aspectos de la invención no solo son útiles para compensar el *aliasing* debido a la modificación de ganancia variable en tiempo y frecuencia, sino también para la extensión de ancho de banda (BWE). En este caso de uso, el espectro replicado producido por la etapa de copiado del algoritmo de BWE tiene que estar conformado con una envolvente espectral que coincida lo más posible a la envolvente espectral original [4 - 6]. Esta envolvente 55 espectral es, en general, dependiente tanto del tiempo como de la frecuencia. Aunque en la mayoría de las técnicas de BWE del estado de la técnica el esquema de copiado es constante, es posible realizar un copiado variable en el tiempo que conduce a un *aliasing* adicional. La nueva técnica de anulación de *aliasing* propuesta también es capaz de manejar estos artefactos de BWE.

[0019] Las realizaciones preferidas de la presente invención se describirán a continuación en el contexto de los dibujos adjuntos, en los que:

- La Figura 1 es un ejemplo comparativo de un primer aspecto;
- 5 la Figura 1b es una realización preferida de un segundo aspecto;
- la Figura 1c es una representación esquemática de una secuencia de bloques de valores espectrales;
- la Figura 1d es una representación esquemática de la secuencia de porciones temporales de solapamiento que da como resultado la secuencia de bloques de la Figura 1c;
- la Figura 2a ilustra una implementación de una transformada directa introductora de *aliasing*;
- 10 la Figura 2b ilustra una implementación de una transformada inversa reductora de *aliasing*;
- la Figura 3a ilustra una realización adicional del segundo aspecto;
- la Figura 3b ilustra una realización adicional del segundo aspecto;
- la Figura 3c ilustra un diagrama esquemático de una aplicación de extensión de ancho de banda para el primer y segundo aspecto;
- 15 la Figura 4 ilustra un componente de error de *aliasing* para el segundo aspecto;
- la Figura 5a ilustra un ejemplo comparativo del primer aspecto;
- la Figura 5b ilustra un ejemplo comparativo del primer aspecto;
- la Figura 6 ilustra un término de *aliasing* generado por reconstrucción, inversión de tiempo y enventanado relacionado con el segundo aspecto;
- 20 la Figura 7 ilustra factores de ganancia variable en una trama;
- la Figura 8 ilustra una aplicación de factores de ganancia antes de un procesamiento de MDCT; y
- la Figura 9 ilustra señales relacionadas con el procesamiento de la Figura 8.

[0020] La Figura 1a ilustra un aparato para procesar una señal de audio que comprende una secuencia de bloques de valores espectrales de acuerdo con el primer aspecto. El aparato para el procesamiento comprende un procesador 100 para procesar la secuencia de bloques de valores espectrales mediante al menos un valor de modificación 102 para un primer bloque para obtener una primera señal resultante con *aliasing* reducido o sin *aliasing* 104. El procesador procesa además el segundo bloque de la secuencia de bloques, que sigue al primer bloque en el tiempo o que precede al primer bloque en el tiempo y que es inmediatamente adyacente al primer bloque que utiliza al menos un segundo valor de modificación diferente, es decir, un segundo valor de modificación 106, que es diferente del primer valor de modificación para obtener una segunda señal resultante con *aliasing* reducido o sin *aliasing* 108 en un margen de solapamiento. Por lo tanto, el procesador genera dos señales de dominio del tiempo sin *aliasing* o al menos con *aliasing* reducido para el mismo margen de solapamiento. Estas señales 108 se introducen en un combinador 110 para combinar la primera y la segunda señal resultante en el margen de solapamiento para obtener una señal procesada 112 para el margen de solapamiento. Preferentemente, el procesador está configurado para generar dos señales de salida sin *aliasing* para la región de solapamiento y luego preferentemente realiza un fundido cruzado entre dichas dos señales de salida sin *aliasing*. Una implementación para que el procesador logre esto es la provisión de un procesamiento de transformada inversa completa, tal como un procesamiento de IMDCT para ambos bloques o ambos valores de modificación diferentes para los bloques. En otras palabras, el procesador genera un procesamiento de IMDCT completo para el margen de solapamiento del primer bloque, uno y los mismos valores de modificación para el primer y el segundo bloque. Además, el procesador realiza un procesamiento de IMDCT completo para el primer y el segundo bloque, pero en este caso mediante segundos valores de modificación. Ambas operaciones de procesamiento de IMDCT completo dan como resultado las dos señales resultantes de dominio del tiempo sin *aliasing* o al menos con *aliasing* reducido en el margen de solapamiento, que luego son combinadas por el combinador.

[0021] Posteriormente, se describe un segundo aspecto en el contexto de la Figura 1b. La Figura 1b ilustra un aparato para procesar una señal de audio que comprende una secuencia de bloques de valores espectrales de acuerdo con una realización de la presente invención con respecto al segundo aspecto. El aparato comprende un procesador 150 para calcular una señal afectada por *aliasing* 154 mediante al menos un primer valor de modificación 102 para el primer bloque de la secuencia de bloques 114 de valores espectrales y mediante al menos un segundo valor de modificación diferente 106 para un segundo bloque de la secuencia de bloques 114 de valores espectrales. El procesador está configurado para calcular, además de la señal afectada por *aliasing* 154, una señal de error de *aliasing* 158. Además, el aparato comprende un combinador 152 para combinar la señal afectada por *aliasing* y la señal de error de *aliasing*, de modo que una señal de audio procesada 112 obtenida por la combinación del combinador 152 es una señal con *aliasing* reducido o sin *aliasing*.

[0022] Específicamente, de acuerdo con el segundo aspecto ilustrado en la Figura 1b, el procesamiento se realiza mediante los valores de modificación diferentes para los bloques pertenecientes al mismo margen de

solapamiento que da como resultado la señal afectada por *aliasing* 154. Sin embargo, a diferencia de la técnica anterior, donde esta señal afectada por *aliasing* se utiliza para un procesamiento adicional y se tolera el error inducido por *aliasing*, esto no se hace en la presente invención. En su lugar, el procesador 150 calcula además una señal de error de *aliasing* 158, la señal de error de *aliasing* se combina posteriormente con la señal afectada por *aliasing* típicamente restando o generalmente una combinación lineal ponderada para que la señal de audio procesada sea una señal sin *aliasing*, pero incluso cuando la combinación o la señal de error de *aliasing* específica no se calcule completamente con precisión, la combinación da como resultado, sin embargo, una señal de audio procesada que tiene menos error de *aliasing* que la señal afectada por *aliasing* 154.

10 **[0023]** En muchas aplicaciones, es deseable modificar los coeficientes espectrales mediante la aplicación de factores de ganancia antes de introducirlos en la IMDCT, de modo que:

$$Y_j(k) = g_j(k) \cdot X_j(k)$$

15 siendo $X_j(k)$ el espectro de MDCT de la j^{a} trama siendo k el índice de frecuencia, siendo $g_j(k)$ una función de ponderación espectral dependiente de tiempo y frecuencia y siendo $Y_j(k)$ el espectro de MDCT filtrado. Esta aplicación de factores de ganancia también se muestra en la Figura 8.

[0024] En particular, la Figura 8 ilustra un sistema de síntesis de MDCT típico de la técnica anterior, en el que los valores espectrales de un bloque anterior 800 que tiene un índice de tiempo $j-1$ son multiplicados por el al menos un factor de ganancia g_{j-1} por un multiplicador 802. Además, el bloque actual 804 con el índice de tiempo es multiplicado por un factor de ganancia g_j para el bloque actual indicado en 805. Específicamente, el factor de ganancia 801 para el bloque anterior con el índice de tiempo $j-1$ indicado en 801 y el factor de ganancia g_j para el bloque actual 805 son diferentes entre sí. Además, tal como se ha descrito en la Figura 8, el factor de ganancia puede ser dependiente de la frecuencia. Esto se ilustra con el índice de frecuencia k . Por lo tanto, cada valor espectral con el índice k es multiplicado por el correspondiente primer o segundo factor de ganancia 801, 805 dependiendo del bloque 800 u 804 al que pertenece el factor de ganancia. Por lo tanto, debido al hecho de que los factores de ganancia para los dos bloques son variables en el tiempo y / o variables en la frecuencia, un error de *aliasing* produce el procesamiento cuando se aplica el procesamiento de la Figura 8. Este procesamiento consiste en una operación de IMDCT 807 para el bloque anterior 800 y una operación de IMDCT adicional 808 para el bloque actual 804. La operación de IMDCT consiste en una transformada de tiempo de frecuencia y una operación de plegado posterior como se describirá con más detalle en el contexto de la Figura 2b. A continuación, el bloque 809 realiza un procesamiento de solapamiento y suma que comprende, además, antes de la suma real, un enventanador de síntesis, para obtener finalmente una señal afectada por *aliasing* $y(n)$ indicada en 810. Por lo tanto, la señal afectada por *aliasing* 810 puede ser típicamente la misma que la señal afectada por *aliasing* ilustrada en 154 en la Figura 1b.

[0025] Si las ganancias en dos tramas posteriores son diferentes, los términos de *aliasing* ya no se anulan entre sí, como se puede ver en la Figura 9 (derecha), donde los factores de ganancia son constantes en la frecuencia, pero no en el tiempo. En este ejemplo, son $g_0(k) = 1$ y $g_1(k) = 0,7$, de modo que el componente de *aliasing* restante es el *aliasing* de la trama 0 multiplicado por un factor de 0,3.

[0026] Debe observarse que, para un caso tan simple, que se ha seleccionado principalmente a modo ilustrativo, el procesamiento de dominio de frecuencia no sería necesario, ya que se podría lograr un efecto similar sin problemas de *aliasing* aplicando una envolvente temporal adecuada. Sin embargo, este ejemplo ayuda a explicar la idea básica del esquema de reducción de *aliasing*. En este caso, el componente de *aliasing* restante es el tiempo invertido y la señal de entrada enventanada multiplicado por la diferencia de los dos factores de ganancia, que es de 0,3 en nuestro ejemplo. Por lo tanto, el *aliasing* puede anularse mediante las siguientes etapas:

- 50 - reconstrucción de la señal de entrada en la región de solapamiento mediante IMDCT adicional con ganancias iguales para los dos bloques
- multiplicación con la función $w(n) \cdot w(N + n)$
- inversión de tiempo
- multiplicación con la diferencia de ganancia 0,3
- 55 - resta de la salida del procesamiento con ganancia diferente

[0027] La salida de las primeras tres etapas se ilustra en la Figura 6.

[0028] En el primer aspecto, se calculan dos señales de salida sin *aliasing* para la región de solapamiento y

luego se realiza un fundido cruzado entre las mismas. La Figura 5a muestra un diagrama de bloque de un ejemplo comparativo. La primera señal $p_{1,j}(n)$ se obtiene a partir del procesamiento de IMDCT de las dos tramas afectadas con los coeficientes espectrales $X_{j-1}(k)$ y $X_j(k)$ con conjuntos iguales de factores de ganancia $g_{j-1}(k)$. Los componentes de *aliasing* de dominio del tiempo de las dos tramas se anulan entre sí, ya que no hay diferencias de ganancia. En consecuencia, la segunda señal $p_{2,j}(n)$ se genera a partir de $X_{j-1}(k)$ y $X_j(k)$, pero ahora mediante factores de ganancia $g_j(k)$. La variación de la forma espectral se puede obtener ahora realizando un fundido cruzado desde $p_{1,j}(n)$ hacia $p_{2,j}(n)$ dentro de la región de solapamiento:

$$y(jN + n) = f(n) \cdot p_{1,j}(n) + (1 - f(n)) \cdot p_{2,j}(n)$$

10

siendo $f(n)$ típicamente decreciente de manera monótona desde 1 a 0 en el intervalo $0 \leq n < N$. Para el caso $f(n) = 1 - w^2(n)$, se puede demostrar que ambos enfoques producen exactamente la misma salida. La libertad de elegir funciones de fundido cruzado independientes de la ventana de MDCT, sin embargo, ofrece un beneficio a este enfoque, que se ilustra en la Figura 5a.

15

[0029] En cuanto a la variación de ganancia pura, un segundo procedimiento puede reducir los componentes de *aliasing* generando dos señales sin *aliasing* y realizando un fundido cruzado. En este caso, cada una de las dos señales se obtiene utilizando un parcheo equitativo y factores de ganancia equitativos en las dos tramas correspondientes (véase la Figura 5b). No es necesario hacer una distinción especial para parcheos constantes y fijos. El parcheo constante solo inferiría en el mapeo equitativo $P_i(k) = P(k)$ para todas las tramas.

20

[0030] El procesamiento se dificulta más si los factores de ganancia no solo varían en el tiempo, sino también en la frecuencia. En este caso, ya no es posible un reemplazo por una envolvente temporal y la generación de la señal de entrada de tiempo invertido tampoco proporciona una señal adecuada para la reducción del componente de *aliasing*. Esto se ilustra en el siguiente ejemplo, en el que las ganancias en la primera trama son constantes con respecto a la ganancia de frecuencia ($g_0(k) = 1$), pero en la segunda trama $g_1(k)$ varía como se muestra en la Figura 7.

25

[0031] Esto conduce a un componente de *aliasing* no anulado que, sin embargo, tiene una forma diferente a la que se muestra en la Figura 9 (derecha). Como se ve en el primer ejemplo, las diferencias de ganancia están causando componentes de *aliasing* no anulado. De ahí la razón del procedimiento que se muestra en la Figura 3a.

30

[0032] La reducción de *aliasing* se realiza mediante las siguientes etapas ilustradas en la Figura 3a:

35 - generar coeficientes espectrales adicionales mediante la ponderación (307, 308) de los coeficientes originales por las diferencias de ganancia:

$$U_{j-1}(k) = (g_j(k) - g_{j-1}(k)) \cdot X_{j-1}(k)$$

$$U_j(k) = (g_j(k) - g_{j-1}(k)) \cdot X_j(k)$$

40

- reconstrucción (303, 304, 306) de la señal de entrada en la región de solapamiento por IMDCT adicional con $U_{j-1}(k)$ y $U_j(k)$

- multiplicación (330) con la función $w(n) \cdot w(N + n)$

45

- inversión de tiempo (340)

- combinación como la suma y la resta a la salida del procesamiento con ganancia diferente

[0033] El orden del término de diferencia de ganancia ($g_j(k) - g_{j-1}(k)$) determina si debe sumarse o restarse la salida de la inversión de tiempo de la salida de IMDCT regular para la implementación de MDCT ilustrada específicamente. Para otras implementaciones de MDCT los signos pueden ser diferentes:

50

$(g_j(k) - g_{j-1}(k))$: sumar la salida de la inversión de tiempo a la salida de IMDCT regular

$(g_{j-1}(k) - g_j(k))$: restar la salida de la inversión de tiempo de la salida de IMDCT regular.

55

[0034] Por lo tanto, para la realización descrita anteriormente y para el caso ilustrado en la Figura 3a, en el que se ilustra $(g_j(k) - g_{j-1}(k))$, el combinador 341 debe implementarse como un sumador sumando ambas entradas.

[0035] Se debe observar que los signos requeridos pueden ser diferentes para diferentes implementaciones de las transformadas solapadas. Hay, por ejemplo, al menos cuatro realizaciones de la MDCT apilada de forma impar. Además, hay otras realizaciones de una MDCT apilada de forma par o una ELT con solapamiento múltiple. Para la MDCT apilada de forma impar, los signos para los términos de corrección son diferentes. Por lo tanto, la combinación realizada en el bloque 341 puede comprender una suma o una resta.

[0036] El término de corrección de *aliasing* para el ejemplo actual se muestra junto con la salida de IMDCT con *aliasing* en la Figura 4.

10 **[0037]** La etapa de copiado de un algoritmo de BWE se describe mediante una función de mapeo $P(k)$. Para copiar la mitad inferior de los coeficientes espectrales en la mitad superior, se haría de la manera siguiente:

$$P(k) = \begin{cases} k & \text{para } 0 \leq k < N/2 \\ k - N/2 & \text{para } N/2 \leq k < N \end{cases}$$

15 **[0038]** La función de ganancia sería constante e igual a 1 en la mitad inferior

$$g_j(k) = 1 \text{ para } 0 \leq k < N/2$$

[0039] Si los factores de ganancia variaran en la mitad superior del espectro, se produciría nuevamente un *aliasing* no anulado. Sin embargo, la reducción de *aliasing* podría realizarse exactamente de la misma manera que se describe en el primer enfoque descrito anteriormente con la única diferencia de que el parcheo también debe tenerse en cuenta en la generación de la señal de compensación. Esto se puede lograr utilizando los coeficientes espectrales mapeados a medida que se introducen en la IMDCT y ponderándolos apropiadamente con diferencias de ganancia. En este caso, todas las diferencias de ganancia en la mitad inferior serían cero.

25

[0040] Una extensión de ancho de banda más avanzada podría aplicar el parcheo que varía de trama en trama. Esto puede describirse definiendo una función de mapeo individual $P_j(k)$ para cada trama. En este caso, la reducción de *aliasing* debe tener en cuenta que pueden copiarse componentes diferentes al mismo índice de frecuencia en las dos tramas que influyen en la región de solapamiento común. Esto debe considerarse en la generación del componente de reducción de *aliasing*. Para este propósito, el parche en la primera trama se trata como arriba, teniendo una ganancia de $g_{j-1}(k)$ en la trama $j - 1$ y 0 en la trama j , y se supone que el parche en la trama j tenga una ganancia de 0 en la trama $j - 1$ y $g_j(k)$ en la trama j . Los coeficientes espectrales resultantes para la generación de la señal de reducción de *aliasing* son los siguientes:

35
$$U_j(k) = -g_{j-1}(k) \cdot X_{j-1}(P_{j-1}(k)) + g_j(k) \cdot X_{j-1}(P_j(k))$$

$$U_{j-1}(k) = -g_{j-1}(k) \cdot X_j(P_{j-1}(k)) + g_j(k) \cdot X_j(P_j(k))$$

[0041] El diagrama de bloque para esta configuración se muestra en la Figura 3b.

40

[0042] En la siguiente sección, se describen con más detalle los dos aspectos de las Figuras 3a y de las Figuras 5a y 5b con respecto a las similitudes.

[0043] La MDCT con una resolución de frecuencia de N coeficientes espectrales de una trama $x_j(n)$ de $2N$ muestras que comienzan en la posición de muestra jN está definida por:

45

$$X_j(k) = \sqrt{\frac{2}{N}} \sum_{n=0}^{2N-1} w(n)x_j(n) \cos\left(\frac{\pi}{N}\left(n + \frac{N}{2} + \frac{1}{2}\right)\left(k + \frac{1}{2}\right)\right), 0 \leq k < N$$

con $w(n)$ siendo una función de ventana de longitud $2N$, siendo k el índice de frecuencia, siendo n el índice de muestra en el dominio del tiempo. La trama $x_j(n)$ de la señal de tiempo $x(n)$ se define como:

$$x_j(n) = x(jN + n), 0 \leq n < 2N$$

5

[0044] Se obtiene una trama de salida intermedia $\tilde{y}_j(n)$, $0 \leq n < 2N$ a partir de los componentes espectrales $Y_j(k)$ con la transformada inversa:

$$\tilde{y}_j(n) = \sqrt{\frac{2}{N}} w(n) \sum_{k=0}^{N-1} Y_j(k) \cos\left(\frac{\pi}{N}\left(n + \frac{N}{2} + \frac{1}{2}\right)\left(k + \frac{1}{2}\right)\right), 0 \leq n < 2N$$

10

[0045] La salida final del procesamiento de la MDCT inversa (IMDCT) se calcula sumando segmentos solapados:

$$y(jN + n) = \tilde{y}_{j-1}(N + n) + \tilde{y}_j(n)$$

15

[0046] Con la abreviatura $c_k(x) = \cos\left(\frac{\pi}{N}\left(k + \frac{1}{2}\right)x\right)$ y con las condiciones de ventana

$$w(N - 1 - n) = w(N + n)$$

y

20

$$w^2(n) = 1 - w^2(N + n)$$

la salida de la IMDCT en la región de solapamiento después de la aplicación de los factores de ganancia es:

$$y(n) = \sum_{k=0}^{N-1} y_k(n)$$

25

con

$$y_k(n) = w(N + n)g_{j-1}(k)c_k\left(n + \frac{3N}{2} + \frac{1}{2}\right)X_{j-1}(k) + w(n)g_j(k)c_k\left(n + \frac{N}{2} + \frac{1}{2}\right)X_j(k)$$

30

[0047] El término de corrección de *aliasing* del segundo aspecto descrito con respecto a las Figuras 1b, 3a después del solapamiento y suma, enventanado e inversión de tiempo es:

$$r(n) = \sum_{k=0}^{N-1} r_k(n)$$

35

con

$$r_k(n) = w(n)w(N + n) \left[w(n) \left(g_{j-1}(k) - g_j(k) \right) c_k\left(\frac{5N}{2} - n - \frac{1}{2}\right) X_{j-1}(k) \right]$$

$$+ w(N+n) \left(g_{j-1}(k) - g_j(k) \right) c_k \left(\frac{3N}{2} - n - \frac{1}{2} \right) X_j(k) \Big]$$

[0048] Los términos de coseno tienen las siguientes simetrías:

$$c_k \left(\frac{5N}{2} - n - \frac{1}{2} \right) = c_k \left(\frac{3N}{2} + n + \frac{1}{2} \right)$$

$$c_k \left(\frac{3N}{2} - n - \frac{1}{2} \right) = -c_k \left(\frac{N}{2} + n + \frac{1}{2} \right)$$

5

[0049] Al sustituirlos se obtiene:

$$\begin{aligned} r_k(n) = & w(n)w(N+n) \left[w(n)g_{j-1}(k)c_k \left(\frac{3N}{2} + n + \frac{1}{2} \right) X_{j-1}(k) \right. \\ & - w(N+n)g_{j-1}(k)c_k \left(\frac{N}{2} + n + \frac{1}{2} \right) X_j(k) \\ & - w(n)g_j(k)c_k \left(\frac{3N}{2} + n + \frac{1}{2} \right) X_{j-1}(k) \\ & \left. + w(N+n)g_j(k)c_k \left(\frac{N}{2} + n + \frac{1}{2} \right) X_j(k) \right] \end{aligned}$$

10

$$\begin{aligned} r_k(n) = & w^2(n)w(N+n)g_{j-1}(k)c_k \left(\frac{3N}{2} + n + \frac{1}{2} \right) X_{j-1}(k) \\ & - w(n)w^2(N+n)g_{j-1}(k)c_k \left(\frac{N}{2} + n + \frac{1}{2} \right) X_j(k) \\ & - w^2(n)w(N+n)g_j(k)c_k \left(\frac{3N}{2} + n + \frac{1}{2} \right) X_{j-1}(k) \\ & + w(n)w^2(N+n)g_j(k)c_k \left(\frac{N}{2} + n + \frac{1}{2} \right) X_j(k) \end{aligned}$$

[0050] Restando $r_k(n)$ de $y_k(n)$ se obtienen los términos que construyen la salida de la reducción de *aliasing*:

$$\begin{aligned} y_k(n) - r_k(n) = & (1 - w^2(n)) \left[w(N+n)g_{j-1}(k)c_k \left(\frac{3N}{2} + n + \frac{1}{2} \right) X_{j-1}(k) \right. \\ & \left. + w(n)g_{j-1}(k)c_k \left(\frac{N}{2} + n + \frac{1}{2} \right) X_j(k) \right] \\ & + w^2(n) \left[w(N+n)g_j(k)c_k \left(\frac{3N}{2} + n + \frac{1}{2} \right) X_{j-1}(k) \right. \\ & \left. + w(n)g_j(k)c_k \left(\frac{N}{2} + n + \frac{1}{2} \right) X_j(k) \right] \end{aligned}$$

15

[0051] Estos corresponden a un fundido cruzado entre señales reconstruidas con ganancias $g_{j-1}(k)$ y $g_j(k)$ de acuerdo con el primer aspecto ilustrado y descrito con referencia a las Figuras 1a, 5a y 5b.

20 **[0052]** Posteriormente, se hace referencia a las Figuras 1c y 1d para ilustrar la relación de porciones temporales y bloques tanto en el lado del codificador o del análisis como en el lado del decodificador o de la síntesis.

[0053] La Figura 1d ilustra una representación esquemática de una 0ª porción temporal a una tercera porción temporal y cada porción temporal de estas porciones temporales subsiguientes tiene un cierto margen de solapamiento 170. En base a estas porciones temporales, los bloques de la secuencia de bloques que representan porciones temporales de solapamiento son generados mediante el procesamiento descrito con más detalle con respecto a la Figura 2a que muestra un lado de análisis de una operación de transformada introductora de *aliasing*.

25

[0054] En particular, la señal de dominio del tiempo ilustrada en la Figura 1d, cuando la Figura 1d se aplica al

lado de análisis, es enventanada por un enventanador 201 que aplica una ventana de análisis. Por lo tanto, para obtener la 0^o porción temporal, por ejemplo, el enventanador aplica la ventana de análisis a, por ejemplo, 2048 muestras y específicamente de la muestra 1 a la muestra 2048. Por lo tanto, N es igual a 1024 y una ventana tiene una longitud de 2N muestras, que en el ejemplo es 2048. A continuación, el enventanador aplica una operación de análisis adicional, pero no para la muestra 2049 como la primera muestra del bloque, sino para la muestra 1025 como la primera muestra en el bloque para obtener la primera porción temporal. Por lo tanto, se obtiene el primer margen de solapamiento 170, que es de 1024 muestras de largo para un solapamiento del 50 %. Este procedimiento se aplica además a la segunda y a la tercera porción temporal, pero siempre con un solapamiento para obtener un cierto margen de solapamiento 170.

10

[0055] Se debe enfatizar que el solapamiento no tiene que ser necesariamente un solapamiento del 50 %, sino que el solapamiento puede ser superior o inferior y puede haber incluso un solapamiento múltiple, es decir, un solapamiento de más de dos ventanas, para que una muestra de la señal de audio de dominio del tiempo no contribuya a dos ventanas y, en consecuencia, bloques de valores espectrales solamente, sino que posteriormente una muestra contribuya a incluso más de dos ventanas / bloques de valores espectrales. Por otro lado, los expertos en la técnica han entendido además que existen otras formas de ventana que pueden ser aplicadas por el enventanador 201 de la Figura 2a, que tienen 0 porciones y / o porciones que tienen valores unitarios. Para dichas porciones que tienen valores unitarios, al parecer, dichas porciones se solapan normalmente con 0 porciones de ventanas anteriores o posteriores y, por lo tanto, una cierta muestra de audio ubicada en una porción constante de una ventana que tiene valores unitarios contribuye a un único bloque de valores espectrales solamente.

[0056] Las porciones temporales enventanadas, tal como se obtienen mediante la Figura 1d, se envían posteriormente a un plegador 202 para realizar una operación de plegado. Esta operación de plegado puede realizar, por ejemplo, un plegado de modo que a la salida del plegador 202 haya solo bloques de valores de muestreo que tienen N muestras por bloque. A continuación, después de la operación de plegado realizada por el plegador 202, se aplica un convertidor de tiempo y frecuencia que es, por ejemplo, un convertidor DCT-IV que convierte n muestras por bloque a la entrada en N valores espectrales a la salida del convertidor de tiempo y frecuencia 203.

[0057] Por lo tanto, la secuencia de bloques de valores espectrales obtenida a la salida del bloque 203 se ilustra en la Figura 1c, que muestra específicamente el primer bloque 191 que tiene asociado un primer valor de modificación ilustrado en 102 en la Figura 1a y 1b y que tiene un segundo bloque 192 que tiene asociado el segundo valor de modificación como 106 ilustrado en las Figuras 1a y 1b. Naturalmente, la secuencia tiene más bloques 193 o 194, que preceden al segundo bloque o incluso que conducen el primer bloque como se ilustra. El primer y segundo bloque 191, 192 se obtiene, por ejemplo, mediante la transformación de la primera porción temporal enventanada de la Figura 1d para obtener el primer bloque y el segundo bloque se obtiene mediante la transformación de la segunda porción temporal enventanada de la Figura 1d por el convertidor de tiempo y frecuencia 203 de la Figura 2a. Por lo tanto, ambos bloques de valores espectrales que son adyacentes en el tiempo en la secuencia de bloques de valores espectrales representan un margen de solapamiento cubierto por la primera porción temporal y la segunda porción temporal.

[0058] Posteriormente, se describe la Figura 2b para ilustrar un procesamiento del lado de la síntesis o del lado del decodificador del resultado del procesamiento del lado del codificador o del análisis de la Figura 2a. La secuencia de bloques de valores espectrales emitidos por el convertidor de frecuencia 203 de la Figura 2a se introduce en un modificador 211. Como se ha descrito, cada bloque de valores espectrales tiene N valores espectrales ilustrados en las Figuras 1c a 2b. Cada bloque ha asociado sus valores de modificación como 102, 104 ilustrados en las Figuras 1a y 1b. A continuación, en una operación típica de IMDCT o una transformada de síntesis reductora de redundancia, se realizan operaciones ilustradas por un convertidor de frecuencia-tiempo 212, un plegador 213 para desplegado, un enventanador 214 para aplicar una ventana de síntesis y una operación de solapador / sumador ilustrada por el bloque 215, para obtener la señal de dominio del tiempo en el margen de solapamiento. La misma tiene, en el ejemplo, 2N valores por bloque, de modo que se obtienen después de cada operación de solapamiento y suma, N muestras nuevas de dominio del tiempo sin *aliasing* siempre que los valores de modificación 102, 104 no sean variables en el tiempo o en la frecuencia. Sin embargo, si dichos valores son variables en el tiempo y en la frecuencia, entonces la señal de salida del bloque 215 no es una señal sin *aliasing*, pero este problema es abordado por el primer y el segundo aspecto de la presente invención tal como se ha descrito en el contexto de las Figuras 1b y 1a y como se ha descrito en el contexto de las otras figuras en la especificación.

[0059] Posteriormente, se proporciona una ilustración adicional de los procedimientos realizados por los bloques en la Figura 2a y en la Figura 2b.

[0060] La ilustración se ejemplifica por referencia a la MDCT, pero otras transformadas introductoras de *aliasing* pueden ser procesadas de manera similar y análoga. Como una transformada solapada, la MDCT es un poco inusual en comparación con otras transformadas relacionadas con Fourier porque tiene la mitad de las salidas con respecto a las entradas (en lugar del mismo número). En particular, es una función lineal $F: \mathbf{R}^{2N} \rightarrow \mathbf{R}^N$ (donde \mathbf{R} denota el conjunto de números reales). Los $2N$ números reales x_0, \dots, x_{2N-1} se transforman en los N números reales X_0, \dots, X_{N-1} según la fórmula:

$$X_k = \sum_{n=0}^{2N-1} x_n \cos \left[\frac{\pi}{N} \left(n + \frac{1}{2} + \frac{N}{2} \right) \left(k + \frac{1}{2} \right) \right]$$

10

[0061] (El coeficiente de normalización frente a esta transformada, aquí unidad, es una convención arbitraria y difiere entre tratamientos. Solo el producto de las normalizaciones de la MDCT y de la IMDCT, a continuación, está restringido.)

15 **[0062]** La MDCT inversa se conoce como la IMDCT. Debido a que hay diferentes números de entradas y salidas, a primera vista podría parecer que la MDCT no debe ser invertible. Sin embargo, se logra una invertibilidad perfecta al sumar las IMDCT solapadas de bloques solapados adyacentes en el tiempo, provocando la anulación de los errores y la recuperación de los datos originales; esta técnica se conoce como anulación de *aliasing* de dominio del tiempo (TDAC).

20

[0063] La IMDCT transforma N números reales X_0, \dots, X_{N-1} en $2N$ números reales y_0, \dots, y_{2N-1} según la fórmula:

$$y_n = \frac{1}{N} \sum_{k=0}^{N-1} X_k \cos \left[\frac{\pi}{N} \left(n + \frac{1}{2} + \frac{N}{2} \right) \left(k + \frac{1}{2} \right) \right]$$

25

(Al igual que para la DCT-IV, una transformada ortogonal, la inversa tiene la misma forma que la transformada directa.)

30 **[0064]** En el caso de una MDCT enventanada con la normalización de ventana habitual (véase a continuación), el coeficiente de normalización delante de la IMDCT debe multiplicarse por 2 (es decir, convertirse en $2/N$).

35 **[0065]** En aplicaciones típicas de compresión de señales, las propiedades de transformada se mejoran aún más mediante el uso de una función de ventana w_n ($n = 0, \dots, 2N-1$) que se multiplica con x_n y y_n en las fórmulas de MDCT e IMDCT, anteriores, para evitar discontinuidades en los límites $n = 0$ y $2N$ haciendo que la función vaya suavemente a cero en esos puntos. (Es decir, se enventanan los datos antes de la MDCT y después de la IMDCT.) En principio, x e y podrían tener funciones de ventana diferentes y la función de ventana también podría cambiar de un bloque a otro (especialmente para el caso en que se combinan bloques de datos de tamaños diferentes), pero para mayor simplicidad, se considera el caso común de funciones de ventana idénticas para bloques del mismo tamaño.

40 **[0066]** La transformada permanece invertible (es decir, la TDAC funciona), para una ventana simétrica $w_n = w_{2N-1-n}$, siempre que w cumpla la condición de Princen-Bradley:

$$w_n^2 + w_{n+N}^2 = 1$$

45

se utilizan varias funciones de ventana. Una ventana que produce una forma conocida como transformada solapada modulada viene dada por

$$w_n = \text{sen} \left[\frac{\pi}{2N} \left(n + \frac{1}{2} \right) \right]$$

y es utilizada para MP3 y MPEG-2 AACM, y

5

$$w_n = \text{sen} \left(\frac{\pi}{2} \text{sen}^2 \left[\frac{\pi}{2N} \left(n + \frac{1}{2} \right) \right] \right)$$

10 para Vorbis. AC-3 utiliza una ventana derivada de Kaiser-Bessel (KBD) y MPEG-2 AAC también puede utilizar una ventana de KBD.

[0067] Se debe tener en cuenta que las ventanas aplicadas a la MDCT son diferentes de las ventanas utilizadas para algunos otros tipos de análisis de señal, ya que deben cumplir la condición de Princen-Bradley. Una de las razones de esta diferencia es que las ventanas de MDCT se aplican dos veces, tanto para la MDCT (análisis) y la IMDCT (síntesis).

[0068] Como puede verse al revisar las definiciones, incluso para N, la MDCT es sustancialmente equivalente a una DCT-IV, donde la entrada es cambiada por N/2 y dos N bloques de datos se transforman a la vez. Al examinar esta equivalencia más cuidadosamente, se pueden derivar fácilmente propiedades importantes como la TDAC.

[0069] Para definir la relación precisa con la DCT-IV, se debe tener en cuenta que la DCT-IV corresponde a condiciones de límite par / impar alternativos: par en su límite izquierdo (alrededor de $n = -1/2$), impar en su límite derecho (alrededor de $n = N-1/2$) y así sucesivamente (en lugar de límites periódicos como para una DFT). Esto se deduce de las identidades

$$\cos \left[\frac{\pi}{N} \left(-n - 1 + \frac{1}{2} \right) \left(k + \frac{1}{2} \right) \right] = \cos \left[\frac{\pi}{N} \left(n + \frac{1}{2} \right) \left(k + \frac{1}{2} \right) \right]$$

y

30

$$\cos \left[\frac{\pi}{N} \left(2N - n - 1 + \frac{1}{2} \right) \left(k + \frac{1}{2} \right) \right] = -\cos \left[\frac{\pi}{N} \left(n + \frac{1}{2} \right) \left(k + \frac{1}{2} \right) \right].$$

[0070] Por lo tanto, si sus entradas son una matriz x de longitud N, se puede imaginar extender esta matriz a (x, -xR, -x, xR, ...) y así sucesivamente, donde xR indica x en orden inverso.

35

[0071] Se puede considerar una MDCT con 2N entradas y N salidas, donde se dividen las entradas en cuatro bloques (a, b, c, d), cada uno de tamaño N/2. Si se desplazan a la derecha por N/2 (del término +N/2 en la definición de MDCT), entonces (b, c, d) se extienden más allá del extremo de las N entradas de DCT-IV, por lo que se deben "plegar" nuevamente según las condiciones de límite descritas anteriormente.

40

[0072] Por lo tanto, la MDCT de 2N entradas (a, b, c, d) es exactamente equivalente a una DCT-IV de las N entradas: (-cR-d, a-bR), donde R indica la inversión como arriba.

[0073] Esto se ejemplifica para la función de ventana 202 en la Figura 2a donde a es la porción 204b, b es la porción 205a, c es la porción 205b y d es la porción 206a.

[0074] (De esta forma, cualquier algoritmo para calcular la DCT-IV se puede aplicar superficialmente a la MDCT). De forma similar, la fórmula de IMDCT anterior es precisamente 1/2 de la DCT-IV (que es su propia inversa), donde la salida se extiende (a través de las condiciones del límite) a una longitud 2N y es desplazada nuevamente a la izquierda por N/2. La DCT-IV inversa simplemente devolvería las entradas (-cRd, a-bR) desde arriba. Cuando esto se extiende a través de las condiciones de límite y se desplaza, se obtiene:

50

$$\text{IMDCT}(\text{MDCT}(a, b, c, d)) = (a-bR, b-aR, c+dR, d+cR) / 2.$$

5 **[0075]** La mitad de las salidas de IMDCT son, por lo tanto, redundantes, como $b-aR = -(a-bR)R$ e igualmente para los dos últimos términos. Si se agrupa la entrada en bloques más grandes A, B de tamaño N, donde $A = (a, b)$ y $B = (c, d)$, se puede escribir este resultado de una manera más simple:

$$\text{IMDCT}(\text{MDCT}(A, B)) = (A-AR, B+BR) / 2$$

10 **[0076]** Así se puede entender cómo funciona la TDAC, suponiendo que se calcule la MDCT de los 2N bloques (B, C) solapados al 50 % adyacentes en el tiempo. La IMDCT cederá de forma análoga a lo anterior: $(B-BR, C+Cr) / 2$. Cuando esto se suma al resultado de IMDCT anterior en la mitad de solapamiento, los términos invertidos se anulan y se obtiene simplemente B, recuperando los datos originales.

15 **[0077]** El origen del término “anulación de *aliasing* de dominio del tiempo” resulta evidente ahora. El uso de datos de entrada que se extienden más allá de los límites de la DCT-IV lógica provoca el *aliasing* de los datos de la misma manera que las frecuencias más allá de la frecuencia de Nyquist son sometidas al *aliasing* a frecuencias más bajas, excepto que este *aliasing* se produce en el dominio del tiempo en lugar de en el dominio de frecuencia: no se pueden distinguir las contribuciones de a y de bR a la MDCT de (a, b, c, d) o de manera equivalente, al resultado de
20 $\text{IMDCT}(\text{MDCT}(a, b, c, d)) = (a-bR, b-aR, c+dR, d+cR) / 2$. Las combinaciones c-dR y así sucesivamente, tienen precisamente los signos adecuados para la anulación de las combinaciones cuando se suman.

25 **[0078]** Para N impar (que en la práctica rara vez se utiliza), N/2 no es un número entero, por lo que la MDCT no es simplemente una permutación de cambio de una DCT-IV. En este caso, el cambio adicional a la mitad de una muestra significa que la MDCT / IMDCT pasa a ser equivalente a la DCT-III/II y el análisis es análogo al anterior.

[0079] Se ha visto anteriormente que la MDCT de 2N entradas (a, b, c, d) es equivalente a una DCT-IV de las N entradas (-cR-d, a-bR). La DCT-IV está diseñada para el caso en el que la función en el límite derecho es impar y, por lo tanto, los valores próximos al límite derecho están cerca de 0. Si la señal de entrada es suave, este es el
30 caso: los componentes más a la derecha de a y bR son consecutivos en la secuencia de entrada (a, b, c, d) y, por lo tanto, su diferencia es pequeña. Se puede echar un vistazo al medio del intervalo: si se reescribe la expresión anterior como $(-cR-d, a-bR) = (-d, a) - (b, c)R$, el segundo término, $(b, c)R$, da una transición suave al medio. Sin embargo, en el primer término, $(-d, a)$, hay una discontinuidad potencial donde el extremo derecho de -d se encuentra con el extremo izquierdo de a. Por esta razón se utiliza una función de ventana que reduce los
35 componentes próximos a los límites de la secuencia de entrada (a, b, c, d) hacia 0.

[0080] Anteriormente, se probó la propiedad de TDAC para la MDCT ordinaria, que muestra que la suma de IMDCT de bloques adyacentes en el tiempo en su mitad de solapamiento recupera los datos originales. La derivación de esta propiedad inversa para la MDCT enventanada es solo un poco más complicada.

40 **[0081]** Se considera el solapamiento de conjuntos consecutivos de 2N entradas (A, B) y (B, C) para bloques A, B, C de tamaño N. Se recuerda de lo anterior que cuando (A, B) y (B, C) se introducen en una MDCT, una IMDCT y se suman a su mitad de solapamiento, se obtiene $(B + BR) / 2 + (B - BR) / 2 = B$, los datos originales.

45 **[0082]** Ahora se supone que se multipliquen tanto las entradas de MDCT como las salidas de IMDCT por una función de ventana de longitud 2N. Al igual que arriba, se asume una función de ventana simétrica, que es por lo tanto de la forma (W, WR) donde W es un vector de longitud N y R indica la inversión como antes. Entonces, la condición de Princen-Bradley se puede escribir como $W^2 + W_R^2 = (1, 1, \dots)_1$, con los cuadrados y las sumas realizadas como elemento.

50 **[0083]** Por lo tanto, en lugar de realizar una MDCT (A, B), se realizan ahora MDCT (WA, WRB) con todas las multiplicaciones como elemento. Cuando esto se introduce en una IMDCT y se multiplica de nuevo (como elemento) por la función de ventana, la última mitad N se convierte en:

55
$$W_R'(W_R B + (W_R B)_R) = W_R'(W_R B + W B_R) = W_R^2 B + W W_R B_R$$

(Debe tenerse en cuenta que ya no se dispone de la multiplicación por 1/2, porque la normalización de IMDCT difiere por un factor de 2 en el caso enventanado).

[0084] De manera similar, la MDCT y la IMDCT enventanadas de (B, C) dan, en su primera mitad N:

$$W \cdot (WB - W_R B_R) = W^2 B - WW_R B_R$$

5

[0085] Cuando se suman estas dos mitades, se recuperan los datos originales. La reconstrucción también es posible en el contexto de conmutación de ventana, cuando las dos mitades de ventanas solapadas cumplen la condición de Princen-Bradley. La reducción de *aliasing* podría en este caso realizarse exactamente de la misma manera que la descrita anteriormente. Para transformadas con solapamiento múltiple, podrían requerirse más de dos
10 ramas con todos los valores de ganancia involucrados.

[0086] Posteriormente, el primer aspecto se describe más detalladamente con referencia a las Figuras 5a y 5b. Específicamente, el procesador 100 ilustrado en la Figura 1a puede incluir todos o solo una parte de los elementos 501 a 506 ilustrados en la Figura 5a en el cuadro definido por el número de referencia 100.
15 Preferentemente, el procesador 100 comprende un modificador para modificar el primer bloque de la secuencia ilustrada como X_{j-1} mediante al menos un primer valor de modificación 102 para obtener un primer bloque modificado 551. Esta modificación se realiza preferentemente mediante un multiplicador de ganancia 510, que puede implementarse de forma digital o análoga o de cualquier otra manera adecuada. Además, el modificador está configurado para modificar el segundo bloque indicado como $K_j(k)$ mediante al menos un segundo valor de
20 modificación 106 para obtener un segundo bloque modificado 552. Esta modificación se realiza de nuevo preferentemente por un multiplicador 509 que puede implementarse de la misma manera que el multiplicador 510 o de una manera diferente. Además, el modificador está configurado para modificar el segundo bloque $X_j(k)$ mediante al menos un primer valor de modificación 102 para obtener un tercer bloque modificado 553, donde esta modificación puede ser realizada nuevamente por un multiplicador 508 que puede implementarse de la misma
25 manera o de una manera diferente con respecto a los multiplicadores 510 o 509.

[0087] Además, el modificador está configurado para modificar el primer bloque X_{j-1} mediante al menos un segundo valor de modificación 106, es decir $X_j(k)$ para obtener un cuarto bloque modificado 554. Preferentemente, el cuarto bloque modificado 554 es generado nuevamente por un multiplicador 507 que puede implementarse de la
30 misma manera o de una manera diferente con respecto a los multiplicadores 510, 509, 508.

[0088] Además, el procesador 100 comprende preferentemente un convertidor de espectro-tiempo para convertir del primero al cuarto bloque modificado 551 a 554 en representaciones temporales correspondientes 561, 562, 563, 564. En particular, el convertidor de espectro-tiempo se implementa comprendiendo los bloques de IMDCT
35 501, 502, 503, 504 que generan del primero al cuarto bloque modificado correspondiente 561 a 564. El convertidor de espectro-tiempo puede implementarse como un algoritmo de IMDCT que comprende elementos reales 212 (convertidor de frecuencia-tiempo), 213 (plegador para un plegado) y 214 (un enventanador de síntesis) de la Figura 2b. Sin embargo, el convertidor de espectro-tiempo puede implementarse como cualquier otro transformador de reducción de *aliasing* que da como resultado, en una salida, un bloque de valores de muestreo de dominio del
40 tiempo que tiene un mayor número de muestras en comparación con el número de muestras en una entrada de los mismos.

[0089] Alternativamente, para una implementación computacionalmente eficiente, no puede calcularse toda la IMDCT o la transformada inversa reductora de *aliasing* entera, sino solamente la sección temporal que contiene la
45 señal de *aliasing* reducido o de anulación de *aliasing*. En esta línea, por ejemplo, en el caso de la IMDCT, podría omitirse una operación de plegado y la mitad de la operación de enventanado de síntesis. Por lo tanto, el procesador puede configurarse para realizar transformadas solapadas que tienen un margen de solapamiento y donde el procesador está configurado solo para realizar operaciones que influyen en los valores del margen de solapamiento. Con respecto a la Figura 2b, las operaciones que no influyen en el margen de solapamiento son las operaciones de
50 plegado de bloque 213 que influyen en la primera mitad del bloque anterior y en la segunda mitad del bloque actual. Además, las operaciones de enventanado correspondientes para la primera mitad del bloque anterior y la segunda mitad del bloque actual tampoco son necesarias para esta implementación eficiente. Esto se debe al hecho de que solo se necesita la segunda mitad del bloque anterior y la primera mitad del bloque actual para el margen de solapamiento.
55

[0090] En el ejemplo comparativo de la Figura 5a, el número de muestras introducidas en el bloque de IMDCT es igual a N y el número de muestras emitidas por el bloque de IMDCT es 2N. Sin embargo, pueden implementarse otras relaciones numéricas correspondientes a otros factores de solapamiento siempre y cuando el

número de muestras emitidas por el convertidor de espectro-tiempo sea mayor que el número de muestras de dominio espectral introducidas en el convertidor de espectro-tiempo correspondiente.

[0091] Además, el convertidor de espectro-tiempo puede implementarse como conteniendo convertidores de espectro en tiempo individuales para cada señal individual que será convertida o puede comprender un convertidor de tiempo de un solo espectro, como solamente el bloque 501 y un controlador de secuencia correspondiente para transformar secuencialmente un bloque modificado después del otro. Además, el procesador 100 comprende un solapador-sumador para solapar y sumar representaciones del primer bloque modificado 561 y del tercer bloque modificado 563 para obtener la primera señal resultante 104 sin *aliasing* o al menos con *aliasing* reducido.

[0092] Además, el solapador-sumador está configurado para solapar y sumar representaciones temporales del segundo bloque modificado 562 y del cuarto bloque modificado 564 para obtener la segunda señal resultante 108 que también es una señal sin *aliasing* o al menos con *aliasing* reducido. Esta operación de solapamiento y suma para la primera y la tercera representación temporal se realiza mediante el bloque de solapamiento y suma 505 y la otra operación de solapamiento y suma para las representaciones temporales del segundo y del cuarto bloque modificado, es decir, para la señal en la línea 562 y 564 se realiza mediante un bloque de solapamiento y suma adicional 506. Una vez más, el sumador de solapamiento puede tener dichos bloques separados o un solo bloque y un control de secuencia correspondiente de cualquier otra forma imaginable para obtener el resultado definido. Preferentemente, cada uno de los bloques 505, 506 se implementa en el bloque 215 correspondiente descrito en el contexto de la Figura 2b.

[0093] Preferentemente, el combinador 110 en la Figura 5a está configurado para combinar la primera señal resultante 104 y la segunda señal resultante 108 mediante el desvanecimiento gradual de la primera señal resultante y mediante la aparición gradual de la segunda señal resultante. Para este fin, se proporciona un bloque de desvanecimiento gradual 520 para la primera señal resultante 104 y el bloque de aparición gradual 521 para la segunda señal resultante. La combinación real puede realizarse mediante un sumador 522 ilustrado en la Figura 5a como un elemento separado. Sin embargo, debe enfatizarse que la operación real del combinador 510 es preferentemente una combinación lineal ponderada donde, para cada muestra, la función de desvanecimiento gradual 520 proporciona un factor de ponderación determinado y posteriormente se suma la muestra correspondiente ponderada por este factor de ponderación a una muestra correspondiente a partir de la otra señal resultante que es ponderada con un factor de ponderación proporcionado por la función de aparición gradual 521 para la muestra correspondiente.

[0094] Tal como se ha descrito, el procesador 100 está configurado para realizar una operación de IMDCT al realizar una conversión de espectro-tiempo y esta operación de IMDCT puede comprender las funcionalidades de los bloques 212, 213, 214, pero la operación de IMDCT puede implementarse de cualquier otra manera y se conocen algoritmos de IMDCT de eficiencia numérica que obtienen básicamente el mismo resultado que lo descrito en el contexto de las Figuras 2a y 2b.

[0095] Además, el procesador 100 está configurado para realizar una operación de solapamiento y suma 505, 506 mediante el cálculo de la primera y la segunda señal resultante en un margen de solapamiento ilustrado en 170 en la Figura 1d. Además, el combinador 104 está configurado para combinar la primera y la segunda señal resultante en un margen de combinación, es decir, en un margen de solapamiento y suma donde este margen de combinación es igual al margen de solapamiento.

[0096] Por lo tanto, debe enfatizarse que la Figura 1d puede considerarse también para representar la operación de solapamiento y suma realizada por el bloque 215 en el lado de la síntesis. Además, cada "porción temporal" en la Figura 1d representa un bloque emitido por el inventanador de síntesis 214 de la Figura 2b y se suman muestras en el margen de solapamiento de dicho bloque inventanado a muestras en el margen de solapamiento del próximo bloque inventanado. Además, las muestras correspondientes se ponderan con factores de ponderación por una función de desvanecimiento gradual tal como 520 y una función de aparición gradual 522 según sea necesario. Por ejemplo, cuando la primera porción temporal corresponde a un bloque inventanado, entonces, en el margen de solapamiento 170 de la Figura 1d, la primera porción temporal se desvanecería gradualmente durante el margen de solapamiento y al mismo tiempo, la segunda porción temporal aparecería gradualmente dentro del margen de solapamiento. Por lo tanto, una función de desvanecimiento gradual proporcionaría factores decrecientes de 1 a 0 preferentemente de forma lineal y distribuidos equitativamente a través del número de muestras de 1 a N. Por lo tanto, cuando la longitud de un bloque inventanado es 2N, el intervalo entre 1 y 0 se separaría en N intervalos iguales y para cada intervalo, el factor de desvanecimiento gradual sería determinado como el centro de cada intervalo, por ejemplo. Del mismo modo, una función de aparición gradual sería

una función que proporciona linealmente factores de aparición gradual crecientes de 0 a 1 nuevamente con N intervalos iguales.

5 **[0097]** Sin embargo, se pueden aplicar otras funciones aparte de las funciones lineales, y se prefiere que, para cada muestra, la suma del factor de aparición gradual para la muestra y el factor de desvanecimiento gradual para la muestra sea igual a 1, por lo que la aparición gradual / desvanecimiento gradual o, generalmente, el fundido cruzado no da como resultado una variación de amplitud o intensidad sonora de la señal de audio. Por lo tanto, una suma de una parte de desvanecimiento gradual y la parte de aparición gradual para cada muestra en un rango de fundido cruzado es constante y preferentemente igual a 1.

10 **[0098]** Preferentemente, el primer aspecto del ejemplo comparativo se aplica en el contexto de las funcionalidades de llenado de ancho de banda en el contexto de la Figura 5b. La extensión de ancho de banda significa que el ancho de banda de la señal de entrada se extiende de manera que típicamente la señal de salida generada por la tecnología de extensión de ancho de banda tiene un ancho de banda mayor que la señal de
15 entrada. Por otro lado, sin embargo, también existen tecnologías de llenado de ancho de banda, que no necesariamente aumentan el ancho de banda, pero que llenan los agujeros espectrales dentro de la señal de entrada. Cuando la frecuencia superior se considera un "agujero espectral", entonces una funcionalidad de llenado de ancho de banda es similar a una tecnología de extensión de ancho de banda. Sin embargo, si un agujero espectral de la señal de entrada que se encuentra, con respecto a la frecuencia, por debajo de un rango para el que existen valores espectrales, la funcionalidad de llenado de ancho de banda no extiende el ancho de banda, pero el
20 resultado de la tecnología tiene el mismo ancho de banda que la entrada. En este contexto, SBR, por ejemplo, es un ejemplo de una tecnología de extensión de ancho de banda y el llenado inteligente de intervalos (IGF) es un ejemplo para una funcionalidad de llenado de ancho de banda sin tener que aumentar necesariamente el ancho de banda de la señal de entrada.

25 **[0099]** Preferentemente, el procesador 100 está configurado para aplicar una funcionalidad de llenado de ancho de banda que tiene una función de parcheo para parchear valores espectrales de un rango de origen 300 a un rango de destino 334, y el procesador está configurado para aplicar la función de parcheo en el cálculo de la primera y la segunda señal resultante. A modo de ejemplo, la Figura 3c ilustra un parcheador para generar desde una
30 secuencia de bloques de valores espectrales que tienen una alta resolución o que representan un rango de origen, una secuencia de bloques de valores espectrales 334 en un rango de llenado de ancho de banda o en un rango de destino. El parcheador está indicado en 332 en la Figura 3c y puede implementarse para aplicar una función de parcheo P(k) como se ilustra. Opcionalmente, como se ilustra en la Figura 3c y en el caso de una funcionalidad de llenado de ancho de banda aplicada, la construcción del procesador 100 y el combinador 110 es la misma que si no se aplicara el llenado de ancho de banda, aparte del hecho de que la secuencia de bloques de valores espectrales
35 indicada en 334 en la Figura 5b es una salida del parcheador en la Figura 3c y los factores de ganancia 102, 106 o, generalmente, los factores de modificación para cada bloque, son como se definen por la funcionalidad de llenado de ancho de banda determinada, tal como la replicación de banda espectral, el llenado inteligente de intervalo o cualquier otra funcionalidad de llenado de ancho de banda. Por lo tanto, el parcheador 332 puede ser parte del
40 procesador o puede implementarse como una etapa de preprocesamiento aplicada en la entrada al procesador.

[0100] Por lo tanto, el aparato para procesar una señal de audio comprende un parcheador, ya sea como parte del procesador 100 o como un bloque aplicado en la dirección de procesamiento de señal, antes del
45 procesador 100 de la Figura 1a, donde este parcheador está configurado para generar el primer bloque en un rango de llenado de ancho de banda que utiliza valores espectrales de un rango de frecuencia diferente, es decir, de un rango de origen de acuerdo con una función de parcheo para el primer bloque y el parcheador está configurado además para generar el segundo bloque de valores espectrales en un rango de llenado de ancho de banda o en un rango de destino mediante un rango espectral de una región de frecuencia diferente o una región de origen de acuerdo con una función de parcheo para el segundo bloque, que puede ser la misma función de parcheo o una
50 función de parcheo diferente.

[0101] Además, como ya se ha descrito en el contexto de la Figura 5a o 5b, el procesador está configurado para realizar una funcionalidad de multiplicación 510, 509, 508, 507 para modificar el primer y segundo bloque mediante funciones de ganancia o valores de ganancia como valores de modificaciones.

55 **[0102]** Se describen otras realizaciones del segundo aspecto de la presente invención en el contexto de las Figuras 3a y 3b.

[0103] Las Figuras 3a y 3b muestran un aparato para el procesamiento de una señal de audio que

comprende una secuencia de bloques de los valores espectrales 114. Cada implementación comprende un procesador 150 para calcular una señal afectada por *aliasing* 154 mediante al menos un primer valor de modificación 102 para un primer bloque de la secuencia de bloques 114 y al menos un segundo valor de modificación diferente para un segundo bloque de la secuencia de bloques. Preferentemente, la funcionalidad del procesador para calcular una señal afectada por *aliasing* comprende un modificador de ganancia para modificar el primer bloque mediante al menos un primer valor de modificación para obtener un primer bloque modificado 351. Esta modificación se realiza preferentemente por un multiplicador 310, pero puede implementarse como se ha descrito en el contexto del correspondiente multiplicador 510 de la Figura 5. Además, el modificador de ganancia está configurado para modificar el segundo bloque $X_j(k)$ mediante al menos un segundo valor de modificación 106 para obtener un segundo bloque modificado 352. Esta modificación una vez más puede realizarse por un multiplicador 309, que nuevamente puede implementarse como se ha descrito en el contexto del multiplicador 510. Para generar la señal afectada por *aliasing*, el procesador 150 comprende un convertidor de espectro-tiempo para convertir el primer y el segundo bloque modificado en representaciones de dominio del tiempo 361, 362 y, adicionalmente, un solapador-sumador está configurado para solapar y sumar las representaciones de dominio del tiempo del primer y el segundo bloque, es decir, 361 y 362, para obtener la señal afectada por *aliasing* 154.

[0104] Además, el procesador también está configurado para estimar la señal de error de *aliasing*. Para este fin, el procesador 150 comprende una funcionalidad adicional del modificador de ganancia ilustrada por un multiplicador 308 y 307 para modificar el primer bloque y el segundo bloque mediante al menos un primer valor de modificación 102 o al menos un segundo valor de modificación 106 para obtener un tercer bloque modificado 353 y un cuarto bloque modificado 354.

[0105] Además, se ilustra un convertidor de espectro-tiempo en 303 y 304 para convertir el tercer bloque modificado 353 y el cuarto bloque modificado 354 en una representación de dominio del tiempo 363 y 364, respectivamente, y a continuación, este tercer y cuarto bloque modificado son procesados por un solapador-sumador para solapar y sumar las representaciones de dominio del tiempo del tercer y cuarto bloque modificado para obtener la señal de error de *aliasing* 158.

[0106] Para manipular la señal de error de *aliasing* 158 para obtener una buena combinación con la señal afectada por *aliasing* 158, el combinador comprende un enventanador 330 para aplicar una función de ventana y un bloque de inversión de tiempo 340 para la inversión de tiempo de una señal.

[0107] La diferencia entre los valores de modificación de ganancia se aplica en el dominio de frecuencia, es decir, en el procesador antes de realizar la conversión de tiempo espectral en los bloques 303 y 304. Con este fin, se hace referencia a la Figura 3a. En particular, en esta realización, el procesador comprende el modificador de ganancia, que está configurado para modificar el primer bloque X_{j-1} a través del multiplicador 307 mediante una diferencia entre al menos un primer valor de modificación y al menos un segundo valor de modificación, donde esta diferencia se calcula preferentemente por valor de frecuencia o valor espectral como se indica por el índice k en la Figura 3a. Además, el modificador de ganancia está configurado para modificar el segundo bloque mediante la diferencia 125 dentro de un multiplicador 308 con el fin de obtener el tercer bloque modificado 353 y el cuarto bloque modificado 354. En esta realización, que se ilustra en la Figura 3a, el combinador comprende la ventana 330 y la inversión de tiempo 340.

[0108] Aunque la secuencia de operaciones de procesamiento dentro del combinador se indica de modo que el enventanador 330 sea operado en la dirección del flujo de señal antes de la inversión de tiempo 340, queda claro que el orden de las operaciones de estos elementos también puede invertirse.

[0109] Por lo tanto, el procesador 150 en la Figura 3a es tal que el modificador de ganancia modifica el primer bloque y el segundo bloque mediante al menos un primer valor de modificación o al menos un segundo valor de modificación. La modificación en la Figura 3a incurre en ambos valores de modificación debido al hecho de que la diferencia entre ambos valores de modificación es utilizada realmente, por ejemplo, por los multiplicadores 308, 307, donde la diferencia de ganancia se indica en 125 en la Figura 3a.

[0110] Además, como se ha descrito, la operación de enventanado 330 y la operación de inversión de tiempo 340 son preferentemente aplicadas. Sin embargo, para otras implementaciones, por ejemplo, cuando se implementa la transformada de tiempo espectral de una manera diferente, puede que no sea necesario en absoluto aplicar una inversión de tiempo 340. Además, cuando no se aplican ventanas de análisis o de síntesis, pero cuando se aplican solamente, por ejemplo, "ventanas rectangulares", también puede prescindirse del enventanado 330.

[0111] En una realización preferida, sin embargo, el enventanador y el operador del inversor de tiempo se ilustran y se posicionan en el orden ilustrado.

[0112] Posteriormente, se describe más detalladamente la Figura 3b. La Figura 3b es análoga a la Figura 3a porque la aplicación de la "diferencia" de ganancia se aplica en el dominio de frecuencia. Sin embargo, debido al hecho de que se aplica una operación de parcheo, la diferencia de ganancia no puede aplicarse explícitamente, pero las diferentes operaciones de parcheo se tienen en cuenta preferentemente.

[0113] Por lo tanto, para obtener la señal de error de *aliasing*, en la operación de parcheo en el contexto de una funcionalidad de llenado de banda ancha como se ha descrito en el contexto de la Figura 3c, se aplican preferentemente los siguientes procedimientos. Primero que nada, se calcula el bloque de valores espectrales 306a, que es el primer bloque, pero el primer bloque se parchea a través de la operación de parcheo para el segundo bloque y este primer bloque 306a se multiplica posteriormente por el factor de ganancia 106 para el segundo bloque.

[0114] Además, como se ilustra, se genera el primer bloque de valores espectrales 306b, que es el mismo que el primer bloque 304b, es decir, el primer bloque al que se aplica la operación de parcheo asociada con el primer bloque y la señal se multiplica, posteriormente por el primer factor de modificación 102. A continuación, las señales generadas por los multiplicadores se combinan, así como se restan o se suman juntas en 329, lo que se puede implementar como un restador o un sumador con una entrada negada, etc. Luego, se obtiene el tercer bloque modificado que corresponde básicamente al bloque 353. De manera similar, se obtiene el bloque 354, es decir, mediante el parcheo del segundo bloque X_j con la operación de parcheo para el segundo bloque, es decir, mediante el bloque 304b y multiplicando este bloque con el segundo factor de modificación 106. Sin embargo, el segundo bloque también está sujeto a un algoritmo de parcheo asociado con el primer bloque como se indica en el elemento 306d y el resultado se multiplica luego por el primer factor de multiplicación a través del multiplicador 307b. Las señales de salida de los multiplicadores 307a y 307b se suman posteriormente en el sumador 328 para obtener finalmente el cuarto bloque modificado 354. A continuación, el tercer y cuarto bloque modificado 353 y 354 se someten a una transformada de tiempo espectral como se describe en el contexto de la Figura 3a y luego se solapan y suman mediante el bloque 306 como se ilustra en la Figura 3b. Seguidamente, se realizan las mismas operaciones en el combinador, tales como el enventanado 330, la inversión de tiempo 340 y finalmente la combinación 152, con el fin de obtener la señal sin *aliasing* 112.

[0115] Aunque la presente invención se ha descrito en el contexto de diagramas de bloques donde los bloques representan componentes de hardware reales o lógicos, la presente invención también puede implementarse mediante un procedimiento implementado por ordenador. En este último caso, los bloques representan las etapas del procedimiento correspondiente, donde estas etapas presentan las funcionalidades realizadas por bloques de hardware lógicos o físicos.

[0116] Aunque algunos aspectos se han descrito en el contexto de un aparato, está claro que estos aspectos también representan una descripción del procedimiento correspondiente, donde un bloque o dispositivo corresponde a una etapa de procedimiento o una característica de una etapa de procedimiento. De manera análoga, los aspectos descritos en el contexto de una etapa de procedimiento representan una descripción de un bloque correspondiente o elemento o característica de un aparato correspondiente. Algunas o todas las etapas de procedimiento se puede ejecutar mediante (o utilizando) un aparato de hardware como, por ejemplo, un microprocesador, un ordenador programable o un circuito electrónico. En algunas realizaciones, una o más de las etapas de procedimiento más importantes pueden ser ejecutadas por dicho aparato.

[0117] La señal transmitida o codificada de la invención puede almacenarse en un medio de almacenamiento digital o puede transmitirse en un medio de transmisión tal como un medio de transmisión inalámbrico o un medio de transmisión por cable como internet.

[0118] Dependiendo de ciertos requisitos de implementación, las realizaciones de la invención pueden implementarse en hardware o software. La implementación puede realizarse utilizando un medio de almacenamiento digital, por ejemplo, un disquete, un DVD, un Blu-Ray, un CD, un ROM, un PROM y EPROM, un EEPROM o una memoria FLASH que tienen señales de control electrónicamente legibles almacenadas en los mismos, que cooperan (o son capaces de cooperar) con un sistema informático programable de modo que se realiza el procedimiento respectivo. Por lo tanto, el medio de almacenamiento digital puede ser legible por ordenador.

[0119] Algunas realizaciones de acuerdo con la invención comprenden un soporte de datos que tiene señales de control electrónicamente legibles, que son capaces de cooperar con un sistema informático programable, de

modo que se realiza uno de los procedimientos descritos en la presente memoria.

5 **[0120]** Generalmente, las realizaciones de la presente invención pueden implementarse como un producto de programa informático con un código de programa, estando operativo dicho código de programa para realizar uno de los procedimientos cuando el producto de programa informático se ejecuta en un ordenador. El código de programa puede almacenarse, por ejemplo, en un soporte legible por máquina.

10 **[0121]** Otras realizaciones comprenden el programa informático para realizar uno de los procedimientos descritos en la presente memoria, almacenado en un soporte legible por máquina.

15 **[0122]** En otras palabras, una realización del procedimiento de la invención es, por lo tanto, un programa informático que tiene un código de programa para realizar uno de los procedimientos descritos en la presente memoria, cuando el programa informático se ejecuta en un ordenador.

20 **[0123]** Otra realización del procedimiento de la invención es, por lo tanto, un soporte de datos (o un medio de almacenamiento no transitorio tal como un medio de almacenamiento digital o un medio legible por ordenador) que comprende, grabado en el mismo, el programa informático para realizar uno de los procedimientos descritos en la presente memoria. El soporte de datos, el medio de almacenamiento digital o el medio grabado son típicamente tangibles y / o no transitorios.

25 **[0124]** Otra realización del procedimiento de la invención es, por lo tanto, un flujo de datos o una secuencia de señales que representan el programa informático para realizar uno de los procedimientos descritos en la presente memoria. El flujo de datos o la secuencia de señales pueden configurarse, por ejemplo, para ser transferidos a través de una conexión de comunicación de datos, por ejemplo, a través de internet.

30 **[0125]** Otra realización comprende un medio de procesamiento, por ejemplo, un ordenador o un dispositivo lógico programable, configurado para, o adaptado para, realizar uno de los procedimientos descritos en la presente memoria.

35 **[0126]** Otra realización comprende un ordenador que tiene instalado en el mismo el programa informático para realizar uno de los procedimientos descritos en la presente memoria.

[0127] Otra realización de acuerdo con la invención comprende un aparato o un sistema configurado para transferir (por ejemplo, electrónicamente u ópticamente) un programa informático para realizar uno de los procedimientos descritos en la presente memoria a un receptor. El receptor puede ser, por ejemplo, un ordenador, un dispositivo móvil, un dispositivo de memoria o similar. El aparato o sistema puede comprender, por ejemplo, un servidor de archivos para transferir el programa informático al receptor.

40 **[0128]** En algunas realizaciones, un dispositivo lógico programable (por ejemplo, una matriz de puerta programable por campo) puede ser utilizado para realizar algunas o todas las funcionalidades de los procedimientos descritos en la presente memoria. En algunas realizaciones, una matriz de puerta programable por campo puede cooperar con un microprocesador con el fin de realizar uno de los procedimientos descritos en la presente memoria. Generalmente, los procedimientos son realizados preferentemente por cualquier aparato de hardware.

45 **[0129]** Las realizaciones descritas anteriormente son meramente ilustrativas de los principios de la presente invención. Se entiende que las modificaciones y variaciones de las disposiciones y los detalles descritos en la presente memoria serán evidentes para otros expertos en la materia. Por lo tanto, la intención es limitarse solo por el alcance de las reivindicaciones de la inminente patente y no por los detalles específicos presentados a modo de descripción y explicación de las realizaciones de la presente memoria.

50

REIVINDICACIONES

1. Un aparato para procesar una señal de audio que comprende una secuencia de bloques de valores espectrales, que comprende:
- 5 un procesador (150) para calcular una señal afectada por *aliasing* (154) mediante al menos un primer valor de modificación (102) para un primer bloque de la secuencia de bloques y mediante al menos un segundo valor de modificación diferente (106) para un segundo bloque de la secuencia de bloques y para estimar una señal de error de *aliasing* (158) que representa un error de *aliasing* en la señal afectada por *aliasing* (154); y
- 10 un combinador (152) para combinar la señal afectada por *aliasing* (154) y la señal de error de *aliasing* (158) de manera que una señal procesada obtenida mediante la combinación es una señal con *aliasing* reducido o sin *aliasing* (112), donde el procesador (150) comprende:
- un modificador (308, 307, 310) para modificar el primer bloque mediante al menos un primer valor de modificación (102) para obtener un primer bloque modificado, para modificar el segundo bloque mediante al menos un segundo valor de modificación (106) para obtener un segundo bloque modificado (352), y para modificar el primer bloque y el segundo bloque mediante al menos un primer valor de modificación o al menos un segundo valor de modificación para obtener un tercer bloque modificado (353, 354, 351) y un cuarto bloque modificado (352);
- 15 un convertidor de espectro-tiempo (301, 302, 303, 304) para convertir del primer al cuarto bloque modificado (351-354) en representaciones de dominio del tiempo (361, 362, 363, 364); y
- 20 un solapador-sumador para solapar y sumar (305) las representaciones de dominio del tiempo (361, 362) del primer y del segundo bloque modificado (351, 352) para obtener la señal afectada por *aliasing* y para solapar y sumar (306) las representaciones de dominio del tiempo (363, 364) del tercer y del cuarto bloque modificado (353, 354) para obtener la señal de error de *aliasing*; y
- 25 donde el combinador (152) comprende un manipulador de señal de *aliasing* para manipular la señal de error de *aliasing*, comprendiendo el manipulador de señal de *aliasing* un enventanador para aplicar una función de ventana y un inversor de tiempo (340) para aplicar una operación de inversión de tiempo, y
- donde el combinador (152) comprende un sumador para sumar una señal de error de *aliasing* manipulada y la señal afectada por *aliasing* o
- 30 donde el procesador (150) comprende:
- un modificador de ganancia para modificar (310) el primer bloque mediante al menos un primer valor de modificación (102) para obtener un primer bloque modificado (351), para modificar (309) el segundo bloque mediante al menos un segundo valor de modificación (106) para obtener un segundo bloque modificado (352), para modificar (308) el segundo bloque mediante una combinación entre al menos un primer valor de modificación y al menos un segundo valor de modificación (125) para obtener un tercer bloque modificado (353) y para modificar (307) el primer bloque mediante la diferencia (125) de al menos un primer valor de modificación y al menos un segundo valor de modificación para obtener un cuarto bloque modificado (354);
- 35 un convertidor de espectro-tiempo (301, 302, 303, 304) para convertir del primer al cuarto bloque modificado (351-354) en representaciones de dominio del tiempo (361, 362, 363, 364); y
- una suma de solapador-sumador para solapar y sumar (305) las representaciones de dominio del tiempo (361, 362) del primer y segundo bloque modificado (351, 352) para obtener la señal afectada por *aliasing* y para solapar y sumar las representaciones de dominio del tiempo (363, 364) del tercer y cuarto bloque modificado (353, 354) para
- 40 obtener la señal de error de *aliasing*; y
- 45 donde el combinador (152) comprende un manipulador de señal de *aliasing* para manipular la señal de error de *aliasing*, comprendiendo el manipulador de señal de *aliasing* un enventanador para aplicar una función de ventana y un inversor de tiempo (340) para aplicar una operación de inversión de tiempo, y
- donde el combinador (152) comprende un sumador o restador (341) para sumar o restar una señal de *aliasing* manipulada y la señal afectada por *aliasing* o
- 50 donde el procesador (150) comprende:
- un modificador de ganancia para modificar el primer bloque (304a) mediante al menos un primer valor de modificación, donde el primer bloque se genera mediante una función de parcheo asociada con el primer bloque para obtener un primer bloque modificado (351), para modificar (309) el segundo bloque (304b) mediante al menos un segundo valor de modificación (106), donde el segundo bloque se genera mediante una operación de parcheo asociada con el segundo bloque para obtener un segundo bloque modificado (352), para modificar (308a) el primer bloque mediante al menos un segundo valor de modificación (106), donde el primer bloque se ha obtenido mediante

- una operación de parcheo asociada con el segundo bloque para obtener un primer tercer bloque intermedio modificado, para modificar (308b) el primer bloque obtenido al aplicar la operación de parcheo asociada con el primer bloque mediante al menos un primer valor modificado (102) para obtener un segundo tercer bloque intermedio modificado, para combinar (329) el primer y el segundo tercer bloque intermedio modificado para obtener el tercer bloque modificado (353), y para modificar (307) el primer bloque mediante la diferencia (125) del al menos primer valor de modificación y del al menos segundo valor de modificación para obtener un cuarto bloque modificado (354);
- un convertidor de espectro-tiempo (301, 302, 303, 304) para convertir del primer al cuarto bloque modificado (351-354) en representaciones de dominio del tiempo (361, 362, 363, 364); y
- 10 una suma de solapador-sumador para solapar y sumar (305) las representaciones de dominio del tiempo (361, 362) del primer y del segundo bloque modificado (351, 352) para obtener la señal afectada por *aliasing* y para solapar y sumar (306) las representaciones de dominio del tiempo (363, 364) del tercer y cuarto bloque modificado (353, 354) para obtener la señal de error de *aliasing*; y
- 15 donde el combinador (152) comprende un manipulador de señal de *aliasing* para manipular la señal de error de *aliasing*, comprendiendo el manipulador de señal de *aliasing* un enventanador para aplicar una función de ventana y un inversor de tiempo (340) para aplicar una operación de inversión de tiempo, y
- donde el combinador (152) comprende un sumador o restador (341) para sumar o restar una señal de error de *aliasing* y la señal afectada por *aliasing* o
- 20 donde el procesador (150) comprende:
- un modificador de ganancia para modificar el primer bloque (304a) mediante al menos un primer valor de modificación, donde el primer bloque se genera mediante una función de parcheo asociada con el primer bloque para obtener un primer bloque modificado (351), para modificar (309) el segundo bloque (304b) mediante al menos
- 25 un segundo valor de modificación (106), donde el segundo bloque se genera mediante una operación de parcheo asociada con el segundo bloque para obtener un segundo bloque modificado (352), para modificar (308a) el primer bloque mediante al menos un segundo valor de modificación (106), donde el primer bloque se ha obtenido mediante la operación de parcheo asociada con el segundo bloque para obtener un primer tercer bloque intermedio modificado, para modificar (308b) el primer bloque obtenido al aplicar la operación de parcheo asociada con el
- 30 primer bloque mediante al menos un primer valor de modificación (102) para obtener un segundo tercer bloque intermedio modificado, para combinar (329) el primer y el segundo tercer bloque intermedio modificado (353), para modificar (307a) el segundo bloque obtenido al aplicar la operación de parcheo asociada con el segundo bloque mediante al menos un segundo valor de modificación (106) para obtener un primer cuarto bloque intermedio modificado, para modificar (307b) el segundo bloque obtenido al aplicar la operación de parcheo asociada con el
- 35 primer bloque mediante al menos un primer valor de modificación (102) para obtener un segundo cuarto bloque intermedio modificado y para combinar (328) el primer y el segundo cuarto bloque intermedio modificado para obtener el cuarto bloque modificado (354);
- un convertidor de espectro-tiempo (301, 302, 303, 304) para convertir del primer al cuarto bloque modificado (351-354) en representaciones de dominio del tiempo (361, 362, 363, 364); y
- 40 una suma de solapador-sumador para solapar y sumar (305) las representaciones de dominio del tiempo (361, 362) del primer y segundo bloque modificado (351, 352) para obtener la señal afectada por *aliasing* y para solapar y sumar (306) las representaciones de dominio del tiempo (363, 364) del tercer y cuarto bloque modificado (353, 354) para obtener la señal de error de *aliasing*; y
- 45 donde el combinador (152) comprende un manipulador de señal de *aliasing* para manipular la señal de error de *aliasing*, comprendiendo el manipulador de señal de *aliasing* un enventanador para aplicar una función de ventana y un inversor de tiempo (340) para aplicar una operación de inversión de tiempo, y
- donde el combinador (152) comprende un sumador para sumar una señal de error de *aliasing* manipulada y la señal afectada por *aliasing*.
- 50 2. Un aparato según la reivindicación 1, donde el combinador (152) está configurado para sumar la señal de error de *aliasing* (158) a la señal afectada por *aliasing* (154), cuando una diferencia entre al menos el primer valor de modificación (102) y al menos un segundo valor de modificación (106) se calcula restando al menos el primer valor de modificación del al menos segundo valor de modificación, y donde el combinador está configurado para sumar la señal de error de *aliasing* (158) a la señal afectada por *aliasing* (154), cuando, para el cálculo de la
- 55 diferencia, al menos un segundo valor de modificación (106) se resta del al menos primer valor de modificación (102).
3. Un aparato según una de las reivindicaciones anteriores, donde el procesador (100) está configurado para realizar una operación de transformada de coseno discreta inversa mediante la realización de una conversión

de espectro-tiempo.

4. Un aparato según una de las reivindicaciones anteriores, donde el procesador (100) está configurado para realizar una operación de procesamiento de solapamiento y suma (505, 506) al calcular la primera y segunda
5 señal resultante en el margen de solapamiento, y donde el combinador (110) está configurado para combinar la primera y la segunda señal resultante en un margen de combinación, donde el margen de combinación es idéntico al margen de solapamiento (170).
5. Un aparato según una de las reivindicaciones anteriores, donde el procesador (100) está configurado
10 para aplicar una operación de llenado de ancho de banda (332) que tiene una función de parcheo para parchear valores espectrales desde un rango de origen (300) a un rango de destino (334), y donde el procesador (100) está configurado para aplicar la función de parcheo al calcular la primera y la segunda señal resultante.
6. Un aparato según una de las reivindicaciones anteriores, un parcheador (332) para generar el primer
15 bloque en un rango de llenado de ancho de banda mediante valores espectrales de un rango de frecuencia diferente de acuerdo con una función de parcheo para el primer bloque, y para generar el segundo bloque en un rango de llenado de ancho de banda que tiene valores espectrales para una región de frecuencia diferente de acuerdo con una función de parcheo asociada con el segundo bloque.
- 20 7. Un aparato según una de las reivindicaciones anteriores, donde el procesador (100) está configurado para realizar una operación de multiplicación al primer y segundo bloque mediante las funciones de ganancia como valores de modificación.
8. Un aparato según una de las reivindicaciones anteriores, donde el procesador (100) está configurado
25 para aplicar una operación de conversión de espectro-tiempo que tiene una parte de transformada de espectro-tiempo (212) y una parte de ventana de síntesis (214) aplicadas a una salida de la parte de transformada de espectro-tiempo (212, 213).
9. Un aparato según una de las reivindicaciones anteriores, donde el procesador (100) está configurado
30 para operar mediante valores de modificación que varían en tiempo y en frecuencia.
10. Un aparato según una de las reivindicaciones anteriores, donde el procesador está configurado para realizar transformadas solapadas que tienen un margen de solapamiento, y donde el procesador está configurado solo para realizar operaciones que no influyen en los valores en el margen de solapamiento.
35
11. Un procedimiento para procesar una señal de audio que comprende una secuencia de bloques de valores espectrales, que comprende:
- calcular (150) una señal afectada por *aliasing* 8154) mediante al menos un primer valor de modificación (102) para
40 un primer bloque de la secuencia de bloques y mediante al menos un segundo valor de modificación diferente (106) para un segundo bloque de la secuencia de bloques y estimar una señal de error de *aliasing* (158) que representa un error de *aliasing* en la señal afectada por *aliasing* (154); y
combinar (152) la señal afectada por *aliasing* (154) y la señal de error de *aliasing* (158) de tal manera que una señal procesada obtenida mediante la combinación sea una señal con *aliasing* reducido o sin *aliasing* (112), donde el
45 cálculo (150) comprende:
- modificar (308, 307, 310) el primer bloque mediante al menos un primer valor de modificación (102) para obtener un primer bloque modificado, modificar el segundo bloque mediante al menos un segundo valor de modificación (106) para obtener un segundo bloque modificado (352) y modificar el primer bloque y el segundo bloque mediante al
50 menos un primer o al menos un segundo valor de modificación para obtener un tercer bloque modificado (353, 354, 351) y un cuarto bloque modificado (352);
convertir (301, 302, 303, 304) del primer al cuarto bloque modificado (351-354) en representaciones de dominio del tiempo (361, 362, 363, 364); y
solapar y sumar las representaciones de dominio del tiempo (361, 362) del primer y segundo bloque modificado
55 (351, 352) para obtener la señal afectada por *aliasing* y solapar y sumar (306) las representaciones de dominio del tiempo (363, 364) del tercer y cuarto bloque modificado (353, 354) para obtener la señal de error de *aliasing*; y
donde la combinación (152) comprende manipular la señal de error de *aliasing*, comprendiendo la manipulación aplicar una función de ventana y aplicar una operación de inversión de tiempo, y
donde la combinación (152) comprende sumar una señal de error de *aliasing* manipulada y la señal afectada por

aliasing o,

donde el cálculo (150) comprende:

5 modificar (310) el primer bloque mediante al menos un primer valor de modificación (102) para obtener un primer bloque modificado (351), modificar (309) el segundo bloque mediante al menos un segundo valor de modificación (106) para obtener un segundo bloque modificado (352), modificar (308) el segundo bloque mediante una combinación entre al menos un primer valor de modificación y al menos un segundo valor de modificación (125) para obtener un tercer bloque modificado (353), y modificar (307) el primer bloque mediante la diferencia (125) de al
 10 menos un primer valor de modificación y al menos un segundo valor de modificación para obtener un cuarto bloque modificado (354); convertir (301, 302, 303, 304) del primer al cuarto bloque modificado (351-354) en representaciones de dominio del tiempo (361, 362, 363, 364); y solapar y sumar (305) las representaciones de dominio del tiempo (361, 362) del primer y segundo bloque modificado (351, 352) para obtener la señal afectada por *aliasing* y solapar y sumar (306) las representaciones de dominio del tiempo (363, 364) del tercer y cuarto bloque
 15 modificado (353, 354) para obtener la señal de error de *aliasing*; y
 donde la combinación (152) comprende manipular la señal de error de *aliasing*, comprendiendo la manipulación aplicar una función de ventana y aplicar una operación de inversión de tiempo, y
 donde la combinación (152) comprende sumar o restar una señal de error de *aliasing* manipulada y la señal afectada por *aliasing* o,

20

donde el cálculo (150) comprende:

modificar el primer bloque (304a) mediante al menos un primer valor de modificación, donde el primer bloque se genera mediante una función de parcheo asociada con el primer bloque para obtener un primer bloque modificado
 25 (351), modificar (309) el segundo bloque (304b) mediante al menos un segundo valor de modificación (106), donde el segundo bloque se genera mediante una operación de parcheo asociada con el segundo bloque para obtener un segundo bloque modificado (352), modificar (308a) el primer bloque mediante al menos un segundo valor de modificación (106), donde el primer bloque se ha obtenido mediante la operación de parcheo asociada con el segundo bloque para obtener un primer tercer bloque intermedio modificado, modificar (308b) el primer bloque
 30 obtenido al aplicar la operación de parcheo asociada con el primer bloque mediante al menos un primer valor modificado (102) para obtener un segundo tercer bloque intermedio modificado, combinar (329) el primer y el segundo tercer bloque intermedio modificado para obtener el tercer bloque modificado (353), modificar (307a) el segundo bloque obtenido al aplicar la operación de parcheo asociada con el segundo bloque mediante al menos un segundo valor de modificación (106) para obtener un primer cuarto bloque intermedio modificado, modificar (307b) el
 35 segundo bloque que se ha obtenido al aplicar la operación de parcheo asociada con el primer bloque mediante al menos un primer valor de modificación (102) para obtener un segundo cuarto bloque intermedio modificado y combinar (328) el primer y el segundo cuarto bloque intermedio modificado para obtener el cuarto bloque modificado (354);
 convertir (301, 302, 303, 304) del primer al cuarto bloque modificado (351-354) en representaciones de dominio del
 40 tiempo (361, 362, 363, 364); y
 solapar y sumar (305) las representaciones de dominio del tiempo (361, 362) del primer y segundo bloque modificado (351, 352) para obtener la señal afectada por *aliasing* y solapar y sumar (306) las representaciones de dominio del tiempo (363, 364) del tercer y cuarto bloque modificado (353, 354) para obtener la señal de error de *aliasing*; y
 45 donde la combinación (152) comprende manipular la señal de error de *aliasing*, comprendiendo la manipulación aplicar una función de ventana y aplicar una operación de inversión de tiempo, y
 donde la combinación (152) comprende sumar una señal de error de *aliasing* manipulada y la señal afectada por *aliasing*.

50 12. Un programa informático para realizar, cuando se ejecuta en un ordenador o en un procesador, el procedimiento de la reivindicación 11.

PRIMER ASPECTO

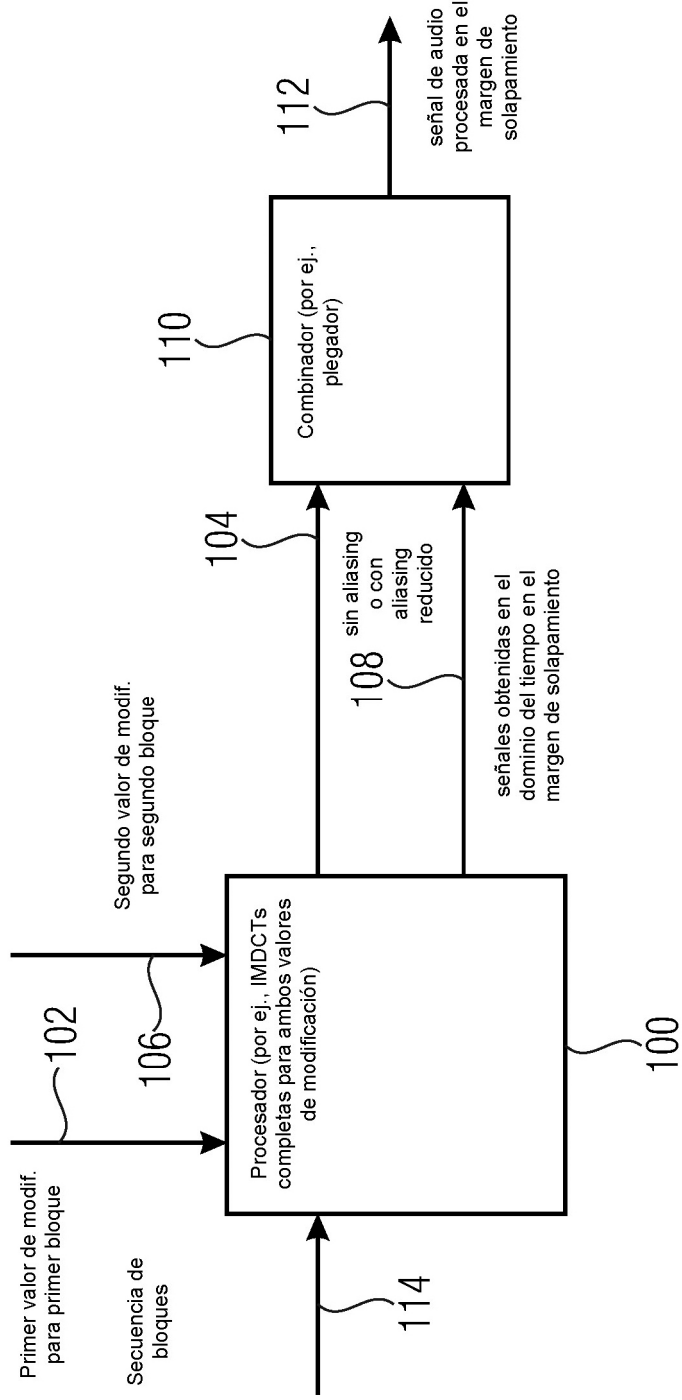


FIG 1A

SEGUNDO ASPECTO

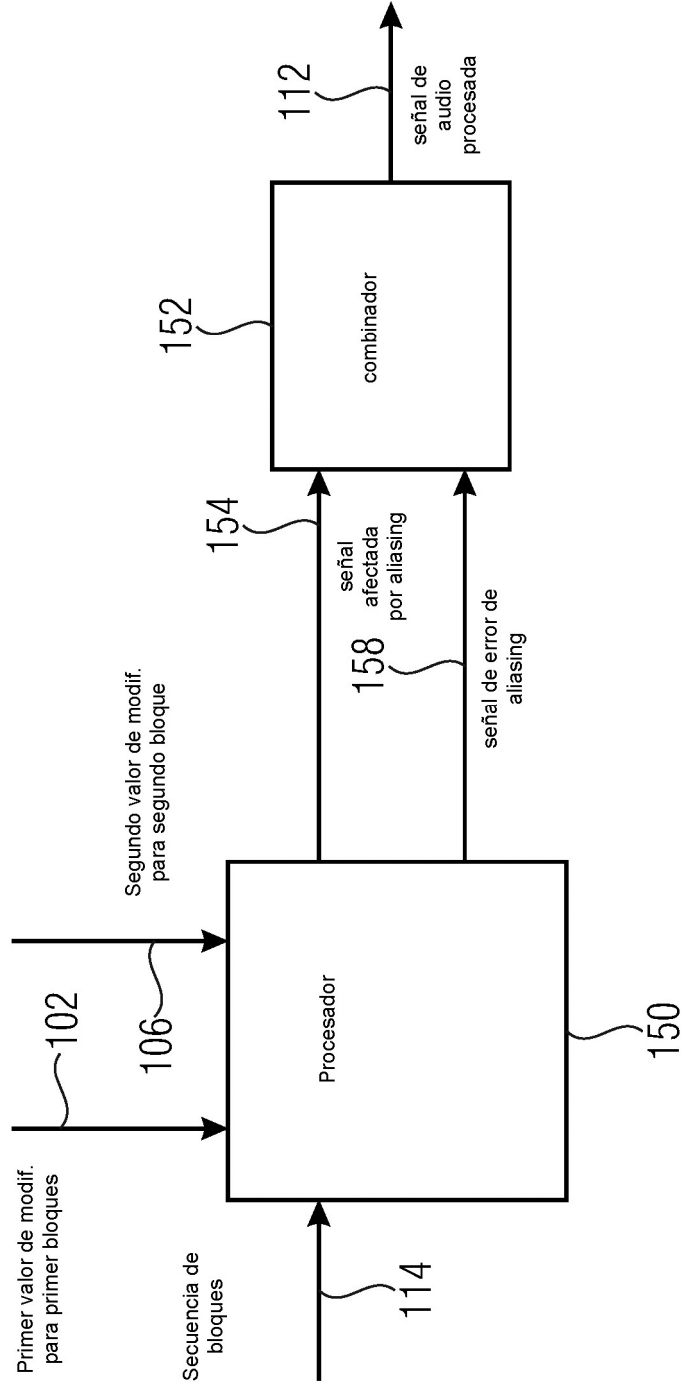


FIG 1B

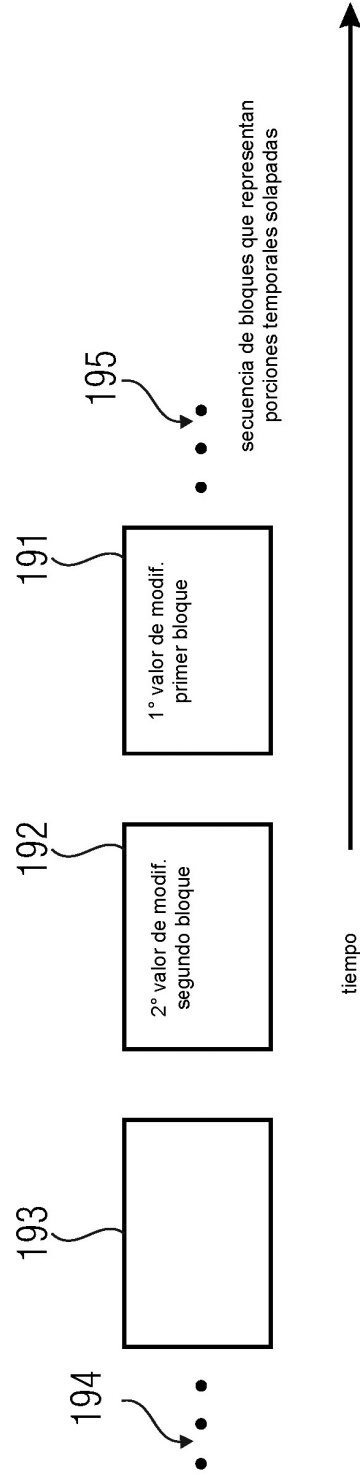


FIG 1C

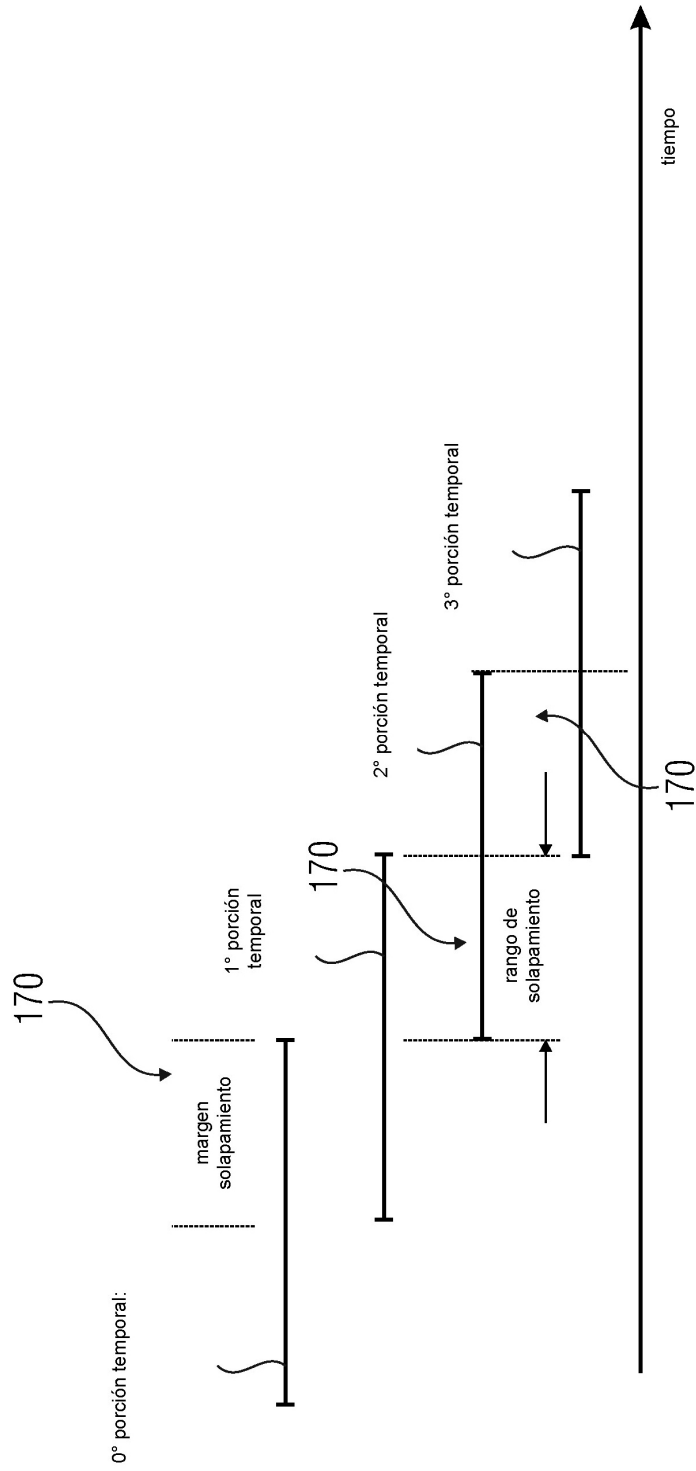


FIG 1D

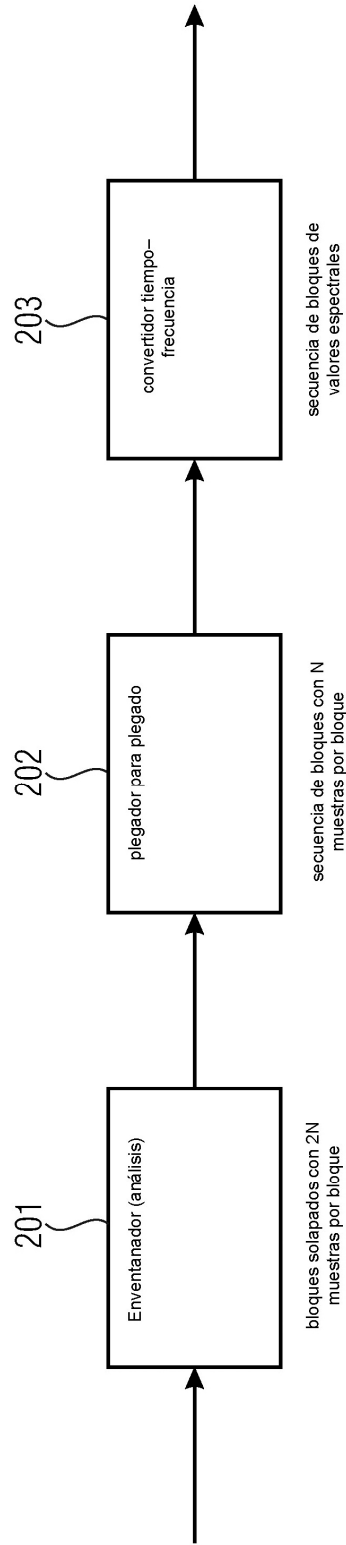


FIG 2A

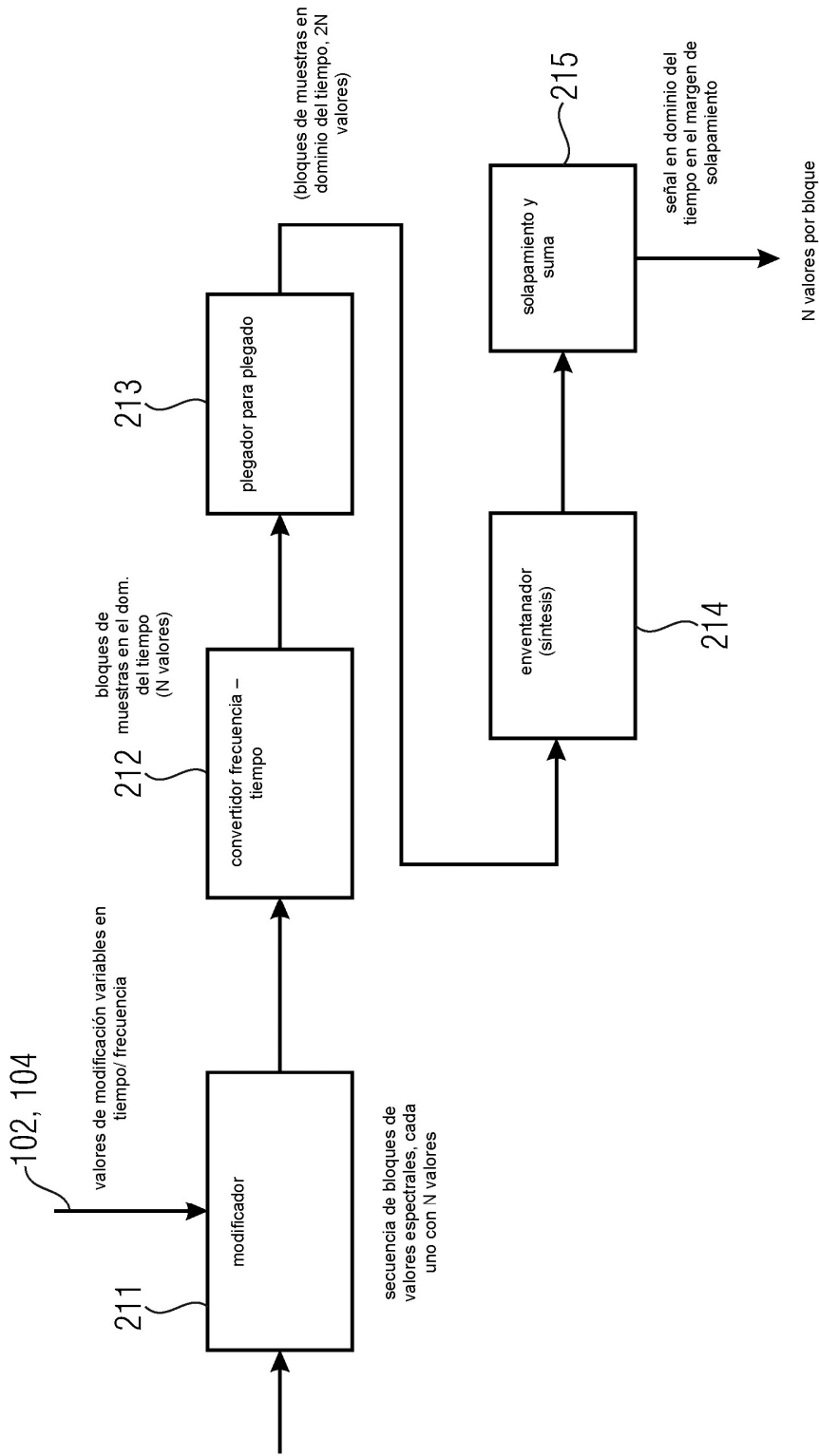
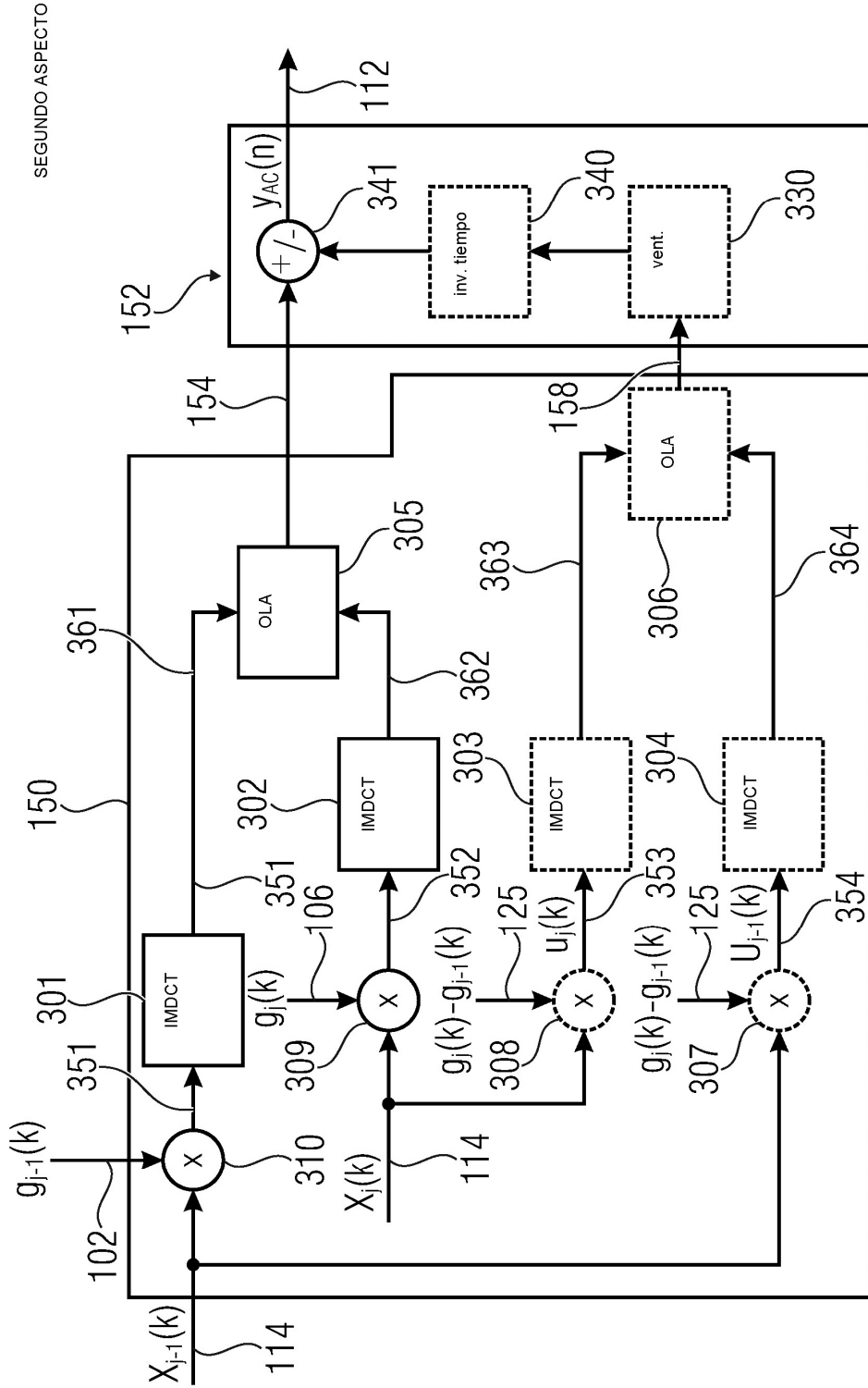


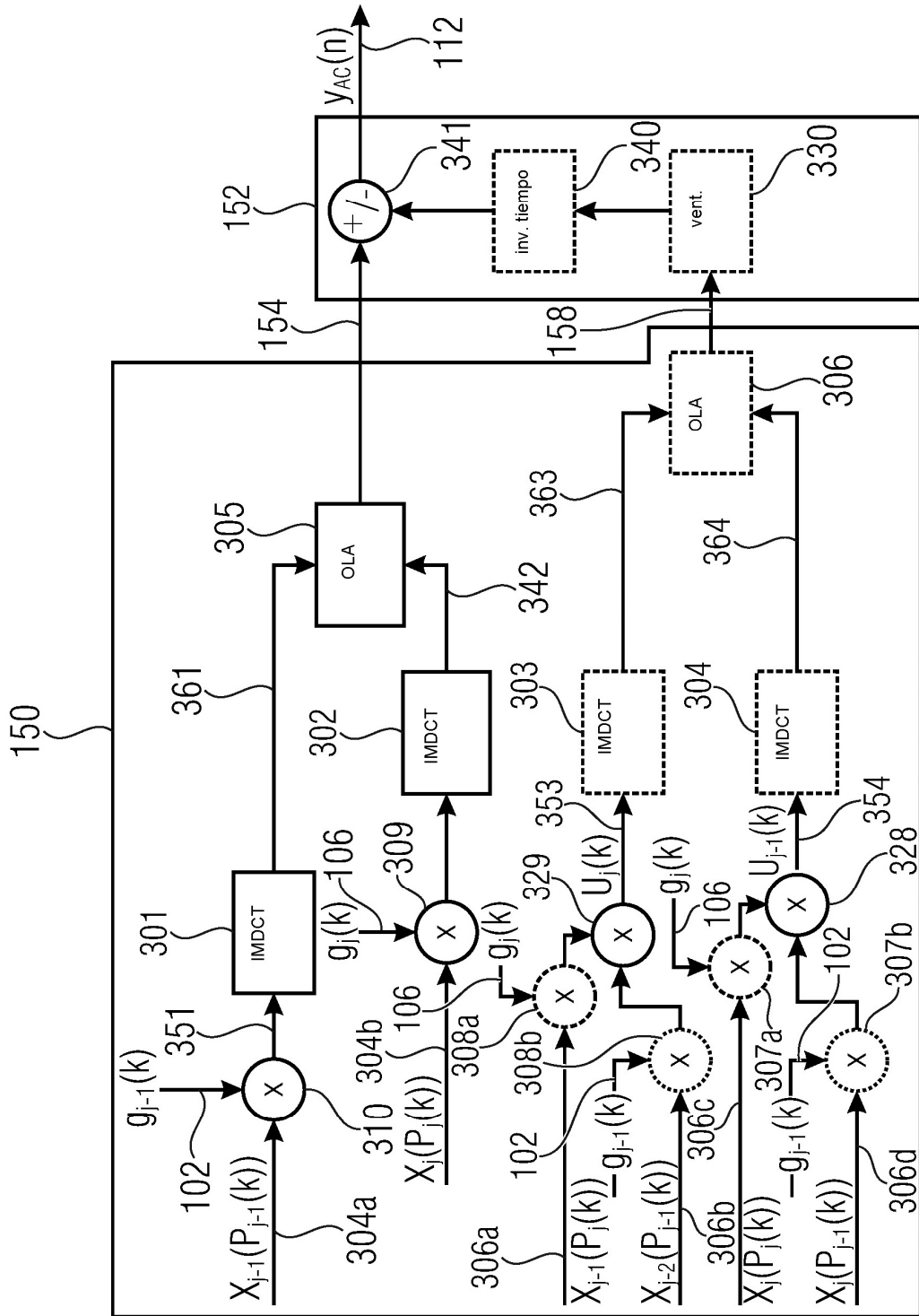
FIG 2B



Reducción de componentes de aliasing en el dominio del tiempo correspondientes a factores de ganancia que varían en el tiempo y la frecuencia.

FIG 3A

SEGUNDO ASPECTO



Reducción de componentes de aliasing en el dominio del tiempo correspondientes a factores de ganancia que varían en el tiempo y la frecuencia.

FIG 3B

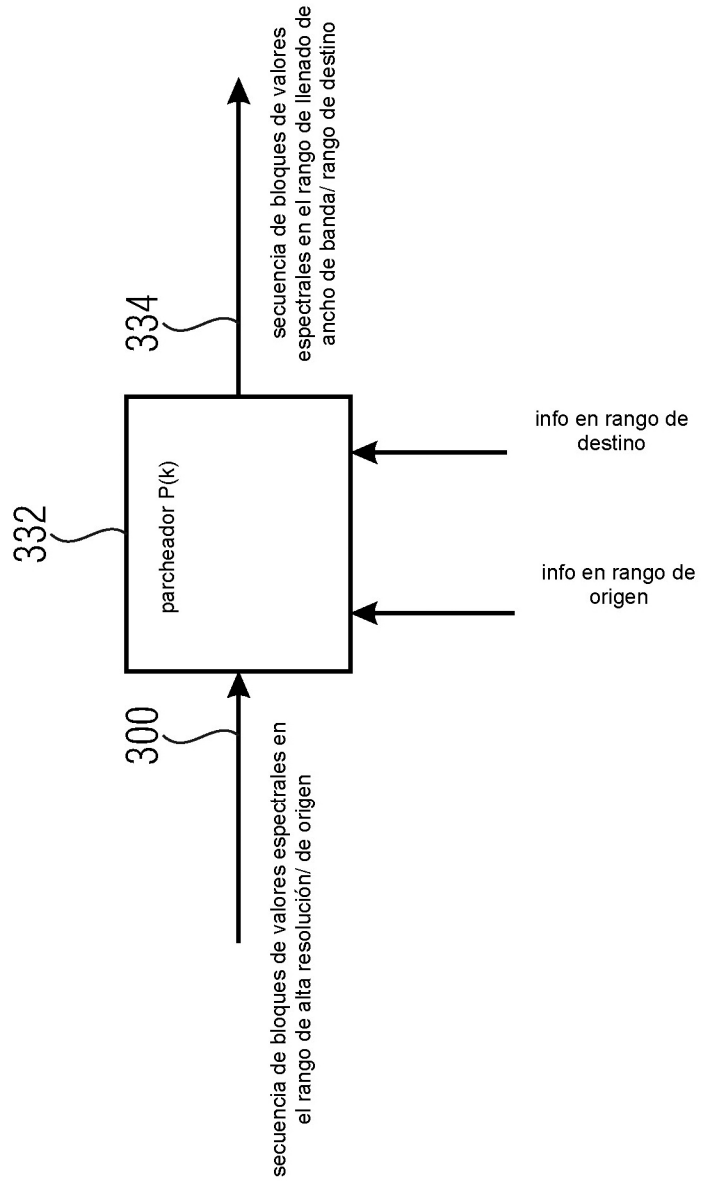
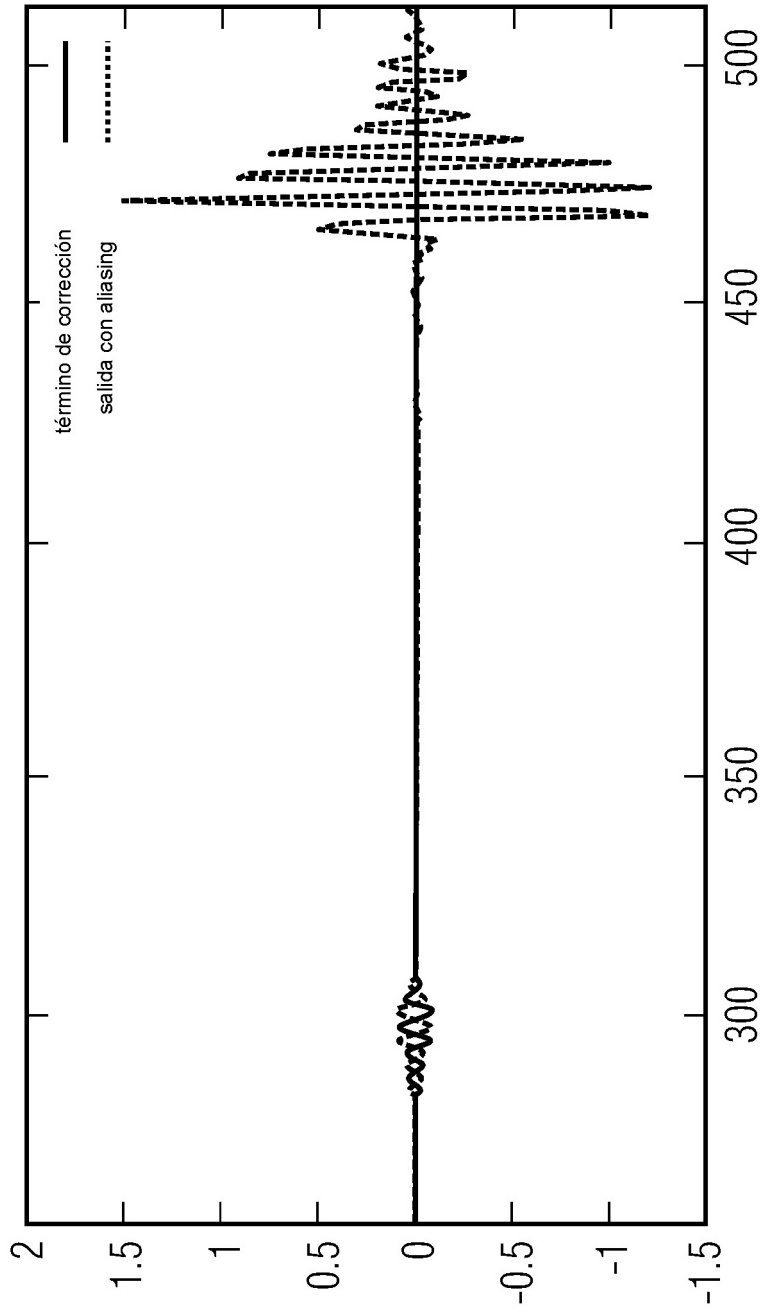
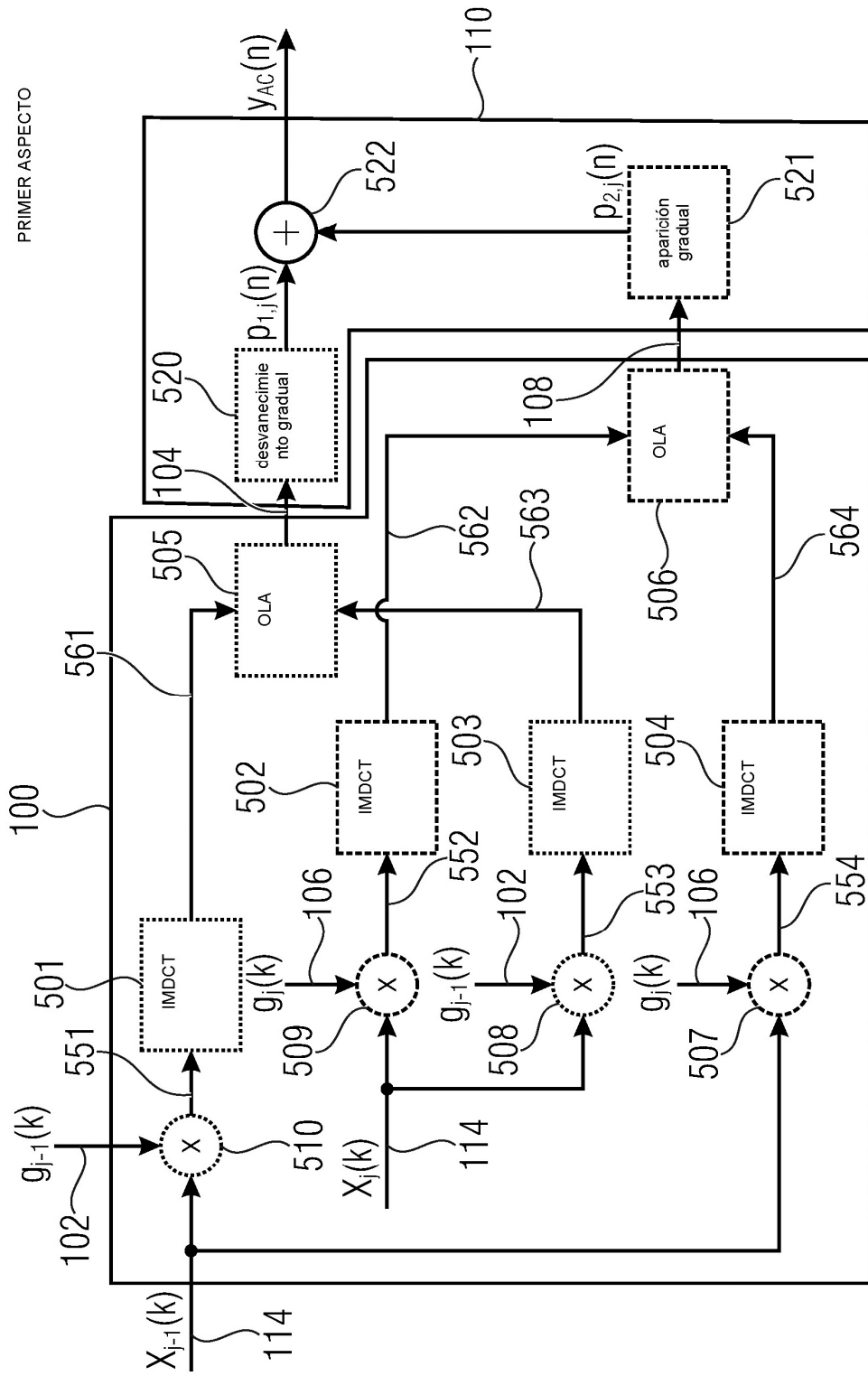


FIG 3C



Componente de reducción de aliasing correspondiente a la variación de los factores de ganancia en el tiempo y la frecuencia

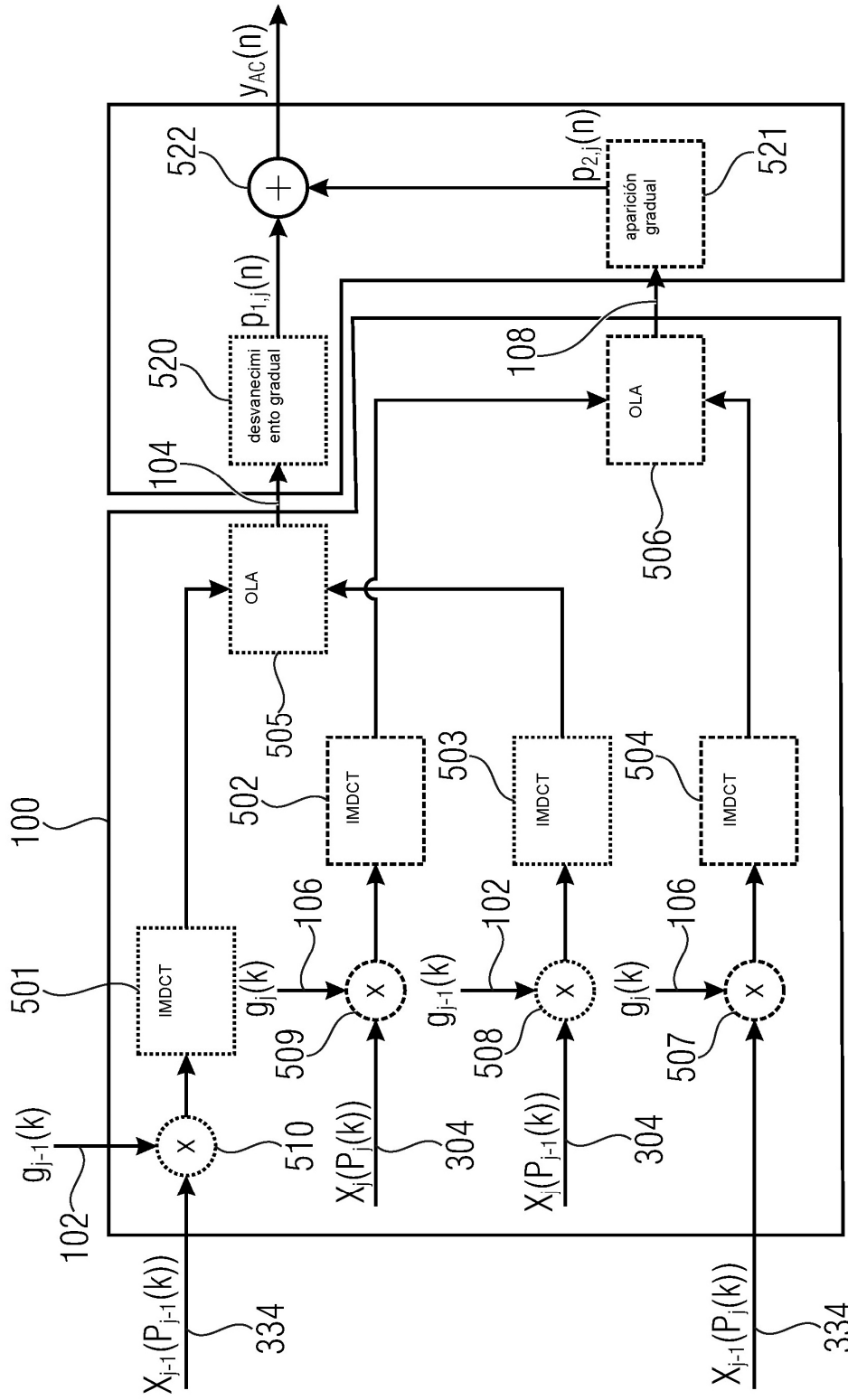
FIG 4



Reducción de componentes de aliasing en el dominio del tiempo con factores de ganancia variantes en el tiempo y la frecuencia

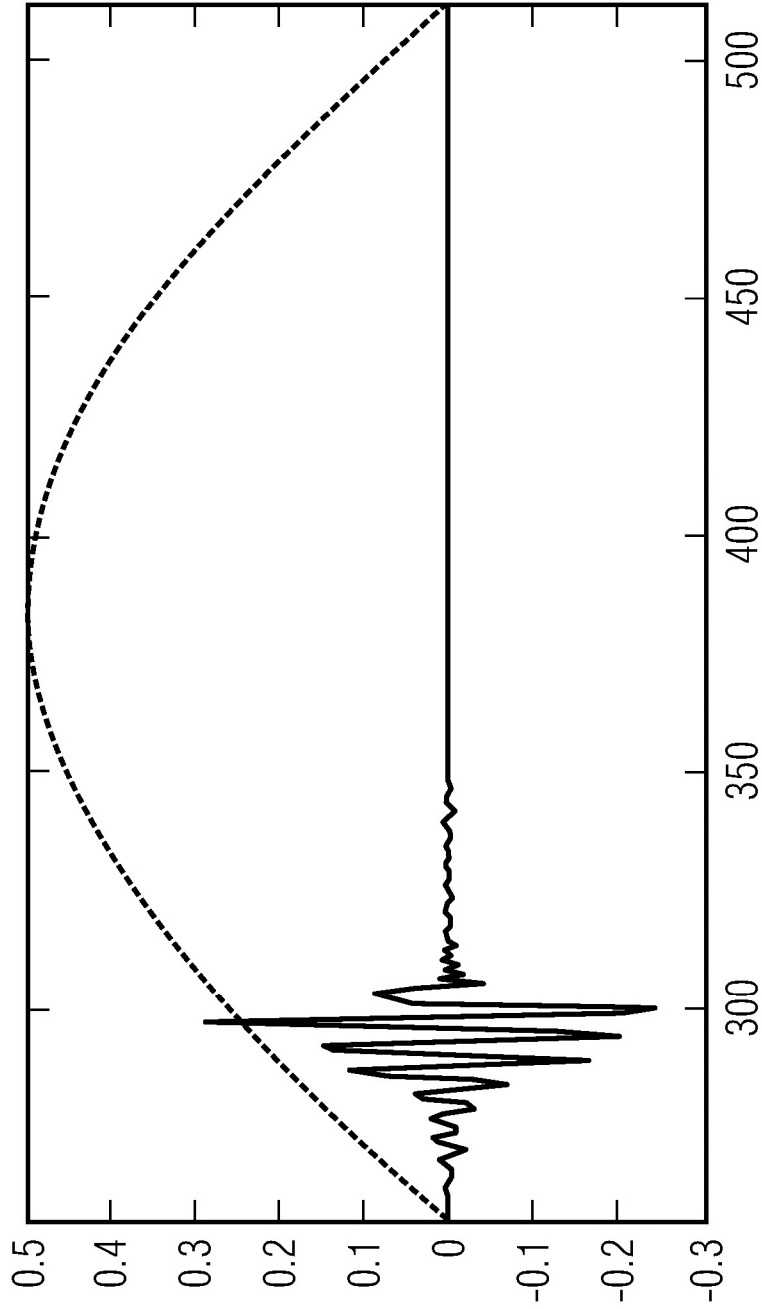
FIG 5A

PRIMER ASPECTO



Reducción de aliasing para extensión de ancho de banda con parcheo variable

FIG 5B



Término de aliasing generado por reconstrucción, inversión en tiempo y enventanado

FIG 6

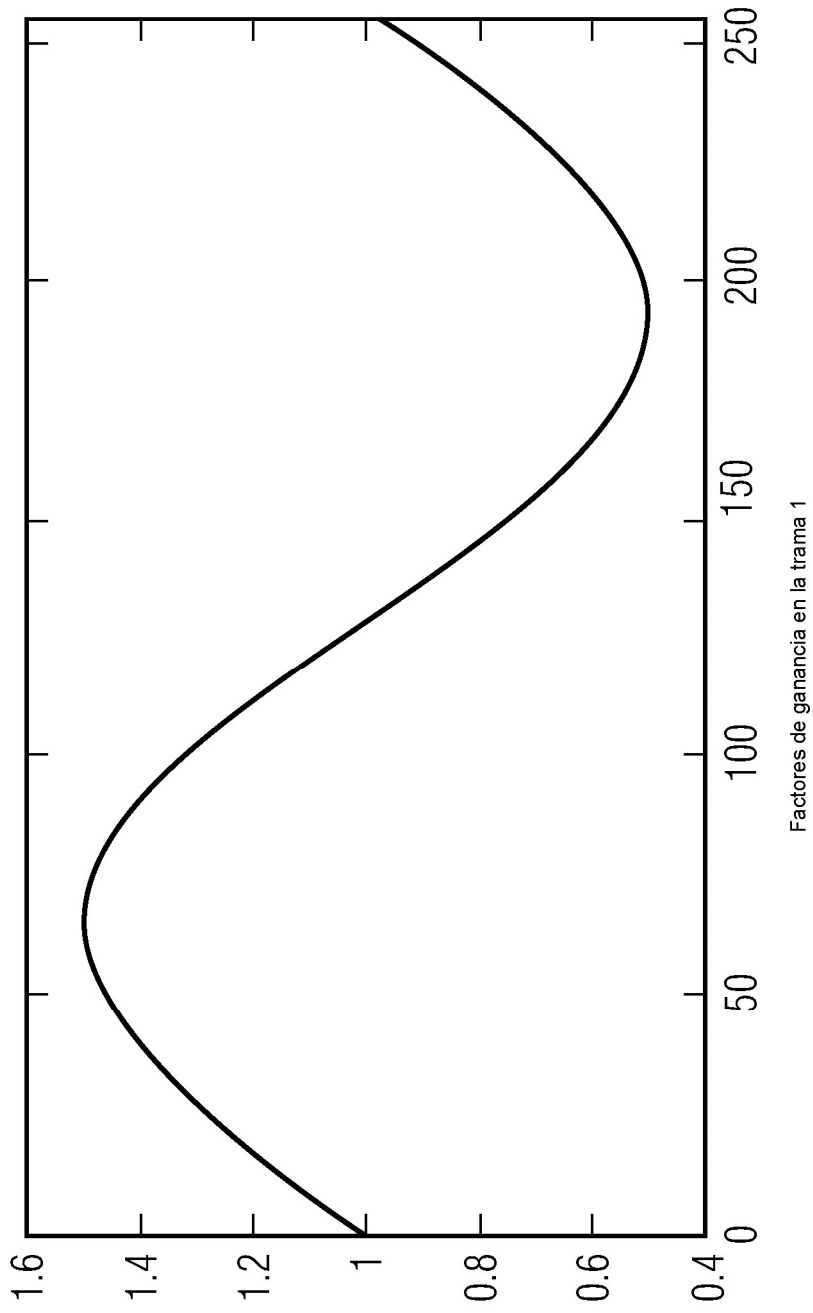
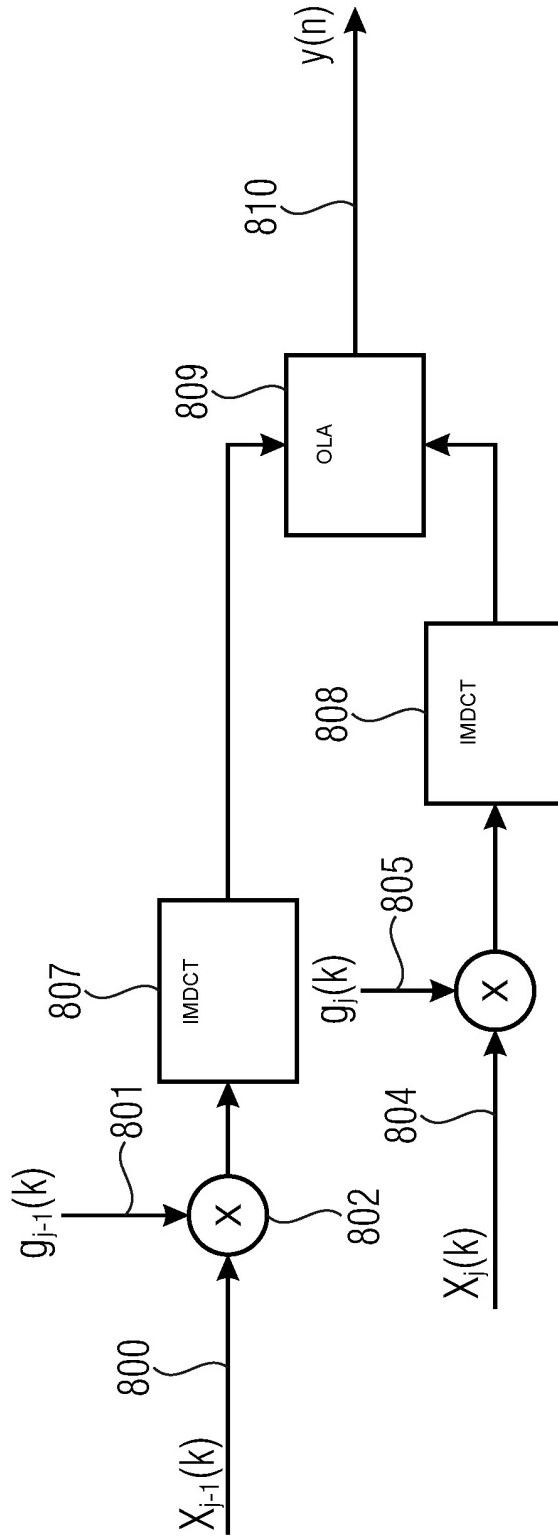
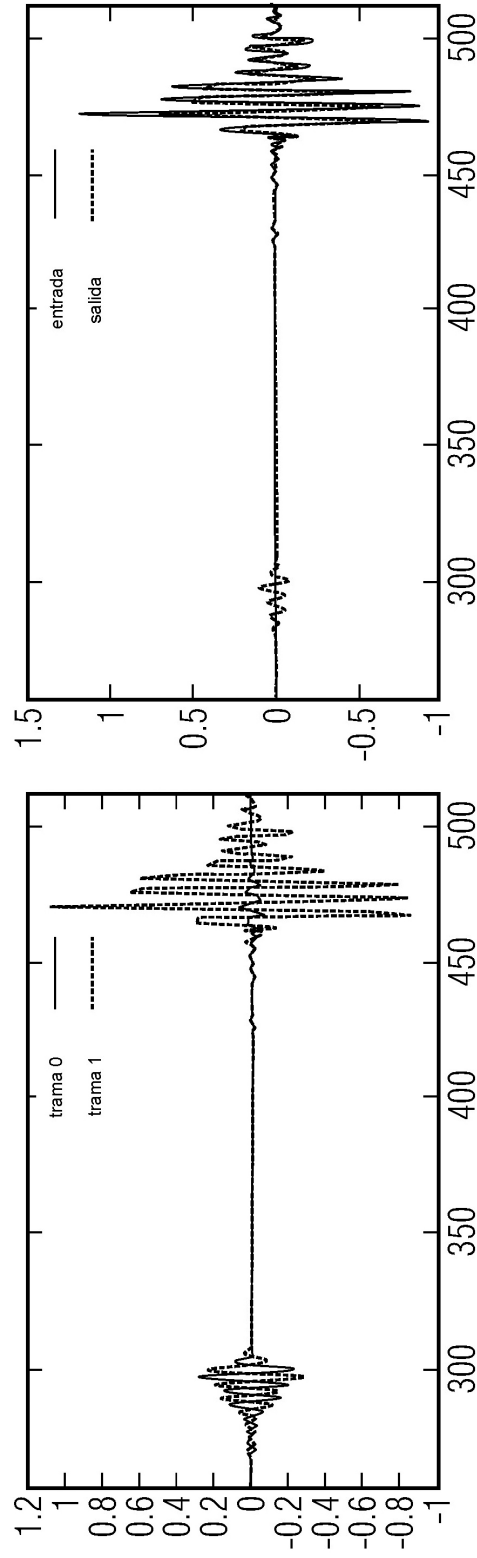


FIG 7



Aplicación de factores de ganancia con anterioridad al procesamiento por IMDCT.

FIG 8
(Técnica Anterior)



izquierda: señales de salida después de las transformadas directa e inversa
derecha: Aliasing no cancelado debido a la variación de los factores de ganancia en el tiempo

FIG 9