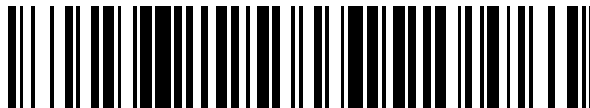


19



OFICINA ESPAÑOLA DE  
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 758 370**

51 Int. Cl.:

**G10L 19/028** (2013.01)

**G10L 21/038** (2013.01)

**G10L 19/02** (2013.01)

**G10L 19/00** (2013.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

96 Fecha de presentación y número de la solicitud europea: **14.09.2011 E 17208522 (7)**

97 Fecha y número de publicación de la concesión europea: **21.08.2019 EP 3319087**

54 Título: **Relleno de subvectores no codificados en señales de audio codificadas por transformada**

30 Prioridad:

**10.03.2011 US 201161451363 P**

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

**05.05.2020**

73 Titular/es:

**TELEFONAKTIEBOLAGET LM ERICSSON (PUBL)  
(100.0%)  
164 83 Stockholm, SE**

72 Inventor/es:

**GRANCHAROV, VOLODYA;  
NÄSLUND, SEBASTIAN y  
SVERRISSON, SIGURDUR**

74 Agente/Representante:

**ELZABURU, S.L.P**

**ES 2 758 370 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

Relleno de subvectores no codificados en señales de audio codificadas por transformada

### Campo técnico

5 La presente tecnología se refiere a la codificación de señales de audio, y especialmente al relleno de subvectores no codificados en señales de audio codificadas por transformada.

### Antecedentes

Un sistema típico de codificador/descodificador basado en codificación por transformada, ha sido ilustrado en la Figura 1.

Las etapas principales de la codificación por transformada son:

- 10 A. Transformar una trama de audio corta (20-40 ms) a un dominio de frecuencia, por ejemplo mediante la Transformada Coseno Discreta Modificada (MDCT);
- B. Dividir el vector  $X(k)$  de la MDCT en múltiples bandas (subvectores SV1, SV2, ...), según se ha ilustrado en la Figura 2. Típicamente, el ancho de las bandas se incrementa hacia frecuencias más altas [1];
- 15 C. Calcular la energía en cada banda. Esto proporciona una aproximación de la envolvente espectral, según se ha ilustrado en la Figura 3;
- D. La envolvente espectral se cuantifica, y los índices de cuantificación se transmiten al descodificador;
- E. Se obtiene un vector residual escalando el vector de MDCT con las ganancias de envolvente, por ejemplo el vector residual se forma mediante los subvectores (SV1, SV2, ...) de MDCT escalados a unidad de energía de Valor Cuadrático Medio (RMS);
- 20 F. Se asignan bits para la cuantificación de diferentes subvectores residuales en base a energías de envolvente. Debido a un presupuesto limitado de bits, no se asigna ningún bit a algunos de los subvectores. Esto ha sido ilustrado en la Figura 4, donde los subvectores correspondientes a ganancias de envolvente por debajo de un umbral TH no han sido asignados con ningún bit.
- G. Los subvectores residuales se cuantifican según los bits asignados, y los índices de cuantificación se transmiten al descodificador. La cuantificación residual puede ser llevada a cabo, por ejemplo, con el esquema de Codificación Factorial de Pulso (FPC) [2].
- 25 H. Los subvectores residuales con cero bits asignados no son codificados, pero en cambio se rellenan de ruido en el descodificador. Esto se consigue creando un Libro de Códigos Virtual (VC) a partir de subvectores codificados, concatenando los coeficientes perceptualmente relevantes del espectro descodificado. El VC crea contenidos en los subvectores residuales no codificados.
- 30 I. En el descodificador, el vector de MDCT se reconstruye mediante el escalonamiento ascendente de los subvectores residuales con ganancias de envolvente correspondientes, y la MDCT inversa se usa para reconstruir la trama de audio en el dominio del tiempo.

35 Un inconveniente del esquema de relleno de ruido convencional, por ejemplo según [1], consiste en que la etapa H crea distorsión audible en la señal de audio reconstruida, cuando se utiliza con el esquema de la FPC.

40 El documento US 2010/0241437 divulga un método para la descodificación espectral perceptual, donde un conjunto inicial de coeficientes espectrales es el relleno del espectro. El relleno del espectro comprende el relleno de ruido de huecos espectrales estableciendo coeficientes espectrales en el conjunto inicial de coeficientes espectrales que no han sido descodificados a partir de un flujo binario igual a los elementos derivados de los coeficientes espectrales descodificados. El conjunto de coeficientes espectrales reconstruido de un dominio de frecuencia formado por el relleno espectral se convierte en una señal de audio de un dominio de tiempo.

### Sumario

Un objeto general es un relleno mejorado de subvectores residuales no codificados de una señal de audio codificada por transformada.

45 Otro objeto es la generación de libros de código virtuales usados para rellenar los subvectores residuales no codificados.

Estos objetos se han alcanzado conforme a las reivindicaciones anexas.

Un primer aspecto de la presente tecnología incluye un aparato para el relleno de subvectores residuales no codificados de una señal de audio codificada por transformada. El aparato comprende:

- Medios para comprimir subvectores residuales codificados.
  - Medios para rechazar subvectores residuales comprimidos que no cumplan un criterio predeterminado;
  - Medios para concatenar los subvectores residuales cuantificados restantes para formar el primer libro de códigos virtual,
- 5
- Medios para combinar pares de coeficientes del primer libro de códigos virtual para formar un segundo libro de códigos virtual,
  - Medios para rellenar subvectores residuales no codificados por debajo de una frecuencia predeterminada con coeficientes procedentes del primer libro de códigos virtual, y para rellenar subvectores residuales no codificados por encima de la frecuencia predeterminada con coeficientes procedentes del segundo libro de códigos virtual.
- 10
- Un segundo aspecto de la presente tecnología incluye un descodificador de audio que comprende el aparato conforme al primer aspecto.
- Un tercer aspecto de la presente tecnología incluye un equipo de usuario que comprende el descodificador de audio conforme al segundo aspecto.
- 15
- Un cuarto aspecto de la presente tecnología incluye un método para el rellenado de subvectores residuales no codificados de una señal de audio codificada por transformada. El método comprende:
- Comprimir subvectores residuales codificados,
  - Rechazar subvectores residuales comprimidos que no cumplan un criterio predeterminado,
  - Concatenar los subvectores residuales comprimidos restantes para formar un primer libro de códigos virtual,
  - Combinar pares de coeficientes del primer libro de códigos virtual para formar un segundo libro de códigos virtual,
  - Rellenar subvectores residuales no codificados por debajo de una frecuencia predeterminada con coeficientes procedentes del primer libro de códigos virtual,
  - Rellenar subvectores residuales no codificados por encima de la frecuencia predeterminada con coeficientes procedentes del segundo libro de códigos virtual.
- 20
- 25
- Una ventaja de la presente tecnología de relleno espectral consiste en un perfeccionamiento perceptivo de señales de audio descodificadas en comparación con el relleno de ruido convencional.

**Breve descripción de los dibujos**

- La presente tecnología, junto con otros objetos y ventajas de la misma, podrá ser mejor comprendida haciendo referencia a la descripción que sigue tomada junto con los dibujos que se acompañan, en los que:
- 30
- La Figura 1 es un diagrama de bloques que ilustra un sistema típico de codificación/descodificación de audio basado en transformada;
- La Figura 2 es un diagrama que ilustra la estructura de un vector de MDCT;
- La Figura 3 es un diagrama que ilustra la distribución de energía en los subvectores de un vector de MDCT;
- La Figura 4 es un diagrama que ilustra el uso de la envolvente espectral para asignación de bits;
- 35
- La Figura 5 es un diagrama que ilustra un residual codificado;
- La Figura 6 es un diagrama que ilustra compresión de un residual codificado;
- La Figura 7 es un diagrama que ilustra el rechazo de subvectores residuales codificados;
- La Figura 8 es un diagrama que ilustra la concatenación de subvectores residuales sobrevivientes para formar un primer libro de códigos virtual;
- 40
- Las Figuras 9A-B son diagramas que ilustran la combinación de coeficientes a partir del primer libro de códigos virtual para formar un segundo libro de códigos virtual;
- La Figura 10 es un diagrama de bloques que ilustra un ejemplo de realización de un generador de libro de códigos virtual de baja frecuencia;
- La Figura 11 es un diagrama de bloques que ilustra un ejemplo de realización de un generador de libro de códigos

virtual de alta frecuencia;

La Figura 12 es un diagrama de bloques que ilustra un ejemplo de realización de un rellenedor de espectro;

La Figura 13 es un diagrama de bloques que ilustra un ejemplo de realización de un descodificador que incluye un rellenedor de espectro;

5 La Figura 14 es un diagrama de flujo que ilustra la generación de libro de códigos virtual de baja frecuencia;

La Figura 15 es un diagrama de flujo que ilustra la generación de libro de códigos virtual de alta frecuencia;

La Figura 16 es un diagrama de flujo que ilustra el relleno de espectro;

La Figura 17 es un diagrama de bloques que ilustra un ejemplo de realización de un generador de libro de códigos virtual de baja frecuencia;

10 La Figura 18 es un diagrama de bloques que ilustra un ejemplo de realización de un generador de libro de códigos virtual de alta frecuencia;

La Figura 19 es un diagrama de bloques que ilustra un ejemplo de realización de un rellenedor de espectro, y

La Figura 20 es un diagrama de bloques que ilustra un ejemplo de realización de un equipo de usuario.

### Descripción detallada

15 Con anterioridad a describir con mayor detalle la presente tecnología, se va a describir brevemente la codificación/descodificación basada en transformada, con referencia a las Figuras 1-7.

La Figura 1 es un diagrama de bloques que ilustra un sistema típico de codificación/descodificación de audio basado en transformada. Una señal de entrada  $x(n)$  se reenvía a un transformador de frecuencia, por ejemplo un transformador 10 de MDCT, donde tramas de audio cortas (20-40 ms) son transformadas en un dominio de frecuencia. La señal  $X(k)$  de dominio de frecuencia resultante se divide en múltiples bandas (subvectores SV1, SV2, ...), según se ha ilustrado en la Figura 2. Típicamente, el ancho de las bandas se incrementa hacia frecuencias más altas [1]. La energía de cada banda se determina en un calculador y cuantificador de envolvente 12. Esto da una aproximación de la envolvente espectral, según se ha ilustrado en la Figura 3. Cada subvector se normaliza según un subvector residual en un normalizador de subvector 14 escalando la inversa del valor de envolvente (ganancia) cuantificado correspondiente.

Un asignador de bits 16 asigna bits para la cuantificación de diferentes subvectores residuales en base a energías de envolvente. Debido a un presupuesto limitado de bits, no se asignan bits a algunos de los subvectores. Esto ha sido ilustrado en la Figura 4, donde los subvectores correspondientes a ganancias de envolvente por debajo de un umbral TH no han sido asignados con ningún bit. Los subvectores residuales son cuantificados en un cuantificador de subvector 18 conforme a los bits asignados. La cuantificación residual puede ser llevada a cabo, por ejemplo, con un esquema de Codificación de Pulso Factorial (FPC) [2]. Los índices de cuantificación de subvector residual y los índices de cuantificación de envolvente son transmitidos a continuación al descodificador a través de un multiplexor (MUX) 20.

En el descodificador, se desmultiplexa la corriente de bits recibida en índices de cuantificación de subvector residual e índices de cuantificación de envolvente en un desmultiplexor (DEMUX) 22. Los índices de cuantificación de subvector residual son descuantificados en subvectores residuales en un descuantificador de subvector 24, y los índices de cuantificación de envolvente son descuantificados en ganancias de envolvente en un descuantificador de envolvente 26. Un asignador de bits 28 usa las ganancias de envolvente para controlar la descuantificación de subvector residual.

40 Los subvectores residuales con cero bits asignados, no han sido codificados en el codificador, y en cambio son rellenados con ruido mediante un rellenedor de ruido 30 en el descodificador. Esto se consigue creando un Libro de Códigos Virtual (VC) a partir de los subvectores codificados, concatenando los coeficientes perceptiblemente relevantes del espectro descodificado ([1] sección 8.4.1). De ese modo, el VC crea contenidos en los subvectores residuales no codificados.

45 En el descodificador, el vector  $\hat{X}(n)$  de MDCT es reconstruido a continuación mediante escalado ascendente de subvectores residuales con ganancias de envolvente correspondientes en un conformador de envolvente 32, y transformando el vector  $\hat{X}(k)$  de dominio de frecuencia resultante en un transformador de MDCT inverso 34.

Un inconveniente del esquema convencional de relleno de ruido descrito con anterioridad consiste en que crea distorsión audible en la señal de audio reconstruida, cuando se usa con el esquema de FPC. La razón principal es que algunos de los vectores codificados pueden ser demasiado dispersos, lo que crea problemas de desajuste energético en las bandas rellenadas con ruido. Adicionalmente, algunos de los vectores codificados pueden contener demasiada estructura (color), lo que conduce a degradaciones perceptivas cuando el relleno de ruido se

realiza a frecuencias elevadas.

La descripción que sigue va a ser enfocada a una realización de un procedimiento mejorado para la generación de libro de códigos virtual en la etapa H anterior.

Un residual codificado  $\hat{X}(k)$ , ilustrado en la Figura 5, se comprime o cuantifica conforme a:

$$Y(k) = \begin{cases} 1 & \text{si } \hat{X}(k) > 0 \\ 0 & \text{si } \hat{X}(k) = 0 \\ -1 & \text{si } \hat{X}(k) < 0 \end{cases} \quad (1)$$

según se ha ilustrado en la Figura 6. Esta etapa garantiza que no exista una estructura excesiva (tal como periodicidad a altas frecuencias) en las regiones rellenas de ruido.

Adicionalmente, la forma específica del residual comprimido  $Y(k)$  permite una baja complejidad en las etapas subsiguientes.

10 Como alternativa, el residual codificado  $\hat{X}(k)$  puede ser comprimido o cuantificado conforme a:

$$Y(k) = \begin{cases} 1 & \text{si } \hat{X}(k) > T \\ 0 & \text{si } -T \leq \hat{X}(k) \leq T \\ -1 & \text{si } \hat{X}(k) < -T \end{cases} \quad (2)$$

donde  $T$  es un número pequeño positivo. El valor de  $T$  puede ser usado para controlar la cantidad de compresión. Esta realización es también útil para señales que hayan sido codificadas por medio de un codificador que cuantifique simétricamente en torno a 0 pero no incluye el valor 0 real.

15 El libro de códigos virtual se construye solamente a partir de subvectores  $M$ -dimensionales "poblados". Si un subvector residual codificado no cumple el criterio:

$$\sum_{k=1}^M |Y(k)| \geq 2 \quad (3)$$

se considera disperso, y se rechaza. Por ejemplo, si el subvector tiene dimensión 8 ( $M=8$ ), la ecuación (3) garantiza que un subvector particular será rechazado desde el libro de códigos virtual si tiene más de 6 ceros. Esto ha sido ilustrado en la Figura 7, donde el subvector SV3 es rechazado, debido a que tiene 7 ceros. Un libro de códigos virtual VC1 se forma concatenando los subvectores restantes o sobrevivientes, según se ha ilustrado en la Figura 8. Puesto que la longitud de los subvectores es un múltiplo de  $M$ , el criterio (3) puede ser usado también para subvectores más largos. En este caso, las partes que no cumplan el criterio son rechazadas.

25 En general, un subvector comprimido se considera "poblado" si contiene más de un 20-30% de componentes distintos de cero. En el ejemplo anterior con  $M=8$ , el criterio es "más de un 25% de componentes distintos de cero".

Un segundo libro de códigos virtual VC2 se crea a partir del libro de códigos virtual VC1 obtenido. Este segundo libro de códigos virtual VC2 está incluso más "poblado" y se usa para frecuencias de relleno superiores a 4,8 kHz (también son por supuesto posibles otras frecuencias de transición; típicamente, la frecuencia de transición está entre 4 y 6 kHz). El segundo libro de códigos virtual VC2 se forma conforme a:

$$Z(k) = Y(k) \oplus Y(N - k), \quad k = 0 \dots N - 1 \quad (4)$$

30 donde  $N$  es el tamaño (número total de coeficientes  $Y(k)$ ) del primer libro de códigos virtual VC1, y la operación combinadora  $\oplus$  se define como:

$$Z(k) = \begin{cases} \text{sign}(Y(k)) \times (|Y(k)| + |Y(N - k)|) & \text{si } Y(k) \neq 0 \\ Y(N - k) & \text{si } Y(k) = 0 \end{cases} \quad (5)$$

35 Esta etapa de combinación o fusión, ha sido ilustrada en las Figuras 9A-B. Se debe apreciar que el mismo par de coeficientes  $Y(k)$ ,  $Y(N - k)$  se utiliza dos veces en el proceso de fusión, una vez en la mitad inferior (Figura 9A) y una vez en la mitad superior (Figura 9B).

Los subvectores no codificados pueden ser rellenados pasando cíclicamente a través del libro de códigos virtual respectivo, VC1 o VC2, dependiendo de si el subvector que va a ser rellenado está por debajo o por encima de la frecuencia de transición, y copiando el número requerido de coeficientes del libro de códigos en el subvector vacío. De ese modo, si los libros de códigos son cortos y existen muchos subvectores para ser rellenados, se reutilizarán los mismos coeficientes para rellenar más de un subvector.

Un ajuste de energía de los subvectores rellenados se lleva a cabo preferentemente sobre la base de un subvector. Esto explica el hecho de que tras el rellenado espectral, los subvectores residuales pueden no tener la energía de RMS unitaria esperada. El ajuste se puede llevar a cabo conforme a:

$$D(k) = \frac{\alpha}{\sqrt{\frac{1}{M} \sum_{k=1}^M Z(k)^2}} Z(k) \quad (6)$$

donde  $\alpha \leq 1$ , por ejemplo,  $\alpha = 0,8$  es un factor de atenuación optimizado perceptualmente. Un motivo para la atenuación perceptual es que la operación de rellenado de ruido da con frecuencia como resultado estadísticas significativamente diferentes del vector residual y es deseable atenuar tales regiones "imprecisas".

En un esquema más avanzado, el ajuste de energía de un subvector particular puede estar adaptado al tipo de subvectores vecinos: si las regiones vecinas están codificadas a una tasa de bits alta, la atenuación del subvector actual es más agresiva (alfa tiende hacia cero). Si las regiones vecinas están codificadas a una tasa de bits baja o rellenas con ruido, la atenuación del subvector actual está limitada (alfa tiende hacia uno). Este esquema impide la atenuación de grandes regiones espectrales continuas, lo que podría conducir a pérdida de sonoridad acústica. Al mismo tiempo, si la región espectral que debe ser atenuada es estrecha, una atenuación incluso muy intensa no afectará a la sonoridad global.

La tecnología descrita proporciona un relleno de ruido mejorado. Las mejoras perceptivas han sido medidas por medio de pruebas de escucha. Estas pruebas indican que el procedimiento de relleno de espectro descrito con anterioridad fue el preferido por un 83% de oyentes de las pruebas, mientras que el procedimiento de relleno de ruido convencional fue el preferido en un 17% de las pruebas.

La Figura 10 es un diagrama de bloques que ilustra un ejemplo de realización de un generador de libro de códigos virtual 60 de baja frecuencia. Los subvectores residuales son enviados a un compresor de subvector 42, el cual está configurado para comprimir subvectores residuales actualmente codificados (es decir, subvectores a los que se haya asignado realmente bits para codificación), por ejemplo conforme a la ecuación (1). Los subvectores comprimidos son reenviados a un rechazador de subvector 44, el cual está configurado para rechazar los subvectores residuales comprimidos que no cumplan con un criterio de dispersión predeterminado, por ejemplo el criterio (3). Los subvectores comprimidos restantes son recopilados en un colector de subvector 46, el cual está configurado para concatenarlos para formar el libro de códigos virtual VC1 de baja frecuencia.

La Figura 11 es un diagrama de bloques que ilustra un ejemplo de realización de un generador de libro de códigos virtual 70 de alta frecuencia. Los subvectores residuales son reenviados a un compresor de subvector 42, el cual está configurado para comprimir subvectores residuales actualmente codificados (es decir, subvectores que han sido actualmente asignados con bits para codificación), por ejemplo según la ecuación (1). Los subvectores comprimidos son reenviados a un rechazador de subvector 44, el cual está configurado para rechazar subvectores residuales comprimidos que no cumplan un criterio de dispersión predeterminado, como por ejemplo el criterio (3). Los restantes subvectores comprimidos son recopilados en un colector de subvector 46, el cual está configurado para concatenarlos para formar el libro de códigos virtual VC1 de baja frecuencia. De ese modo, hasta ese punto, el generador de libro de códigos virtual 70 de alta frecuencia incluye los mismos elementos que el generador de libro de códigos virtual 60 de baja frecuencia. Los coeficientes del libro de códigos virtual VC1 de baja frecuencia son reenviados a un combinador de coeficientes 48, el cual está configurado para combinar pares de coeficientes para formar el libro de códigos virtual VC2 de alta frecuencia, por ejemplo conforme a la ecuación (5).

La Figura 12 es un diagrama de bloques que ilustra un ejemplo de realización de un rellenedor de espectro 40. Los subvectores residuales son reenviados a un compresor de subvector 42, el cual está configurado para comprimir subvectores residuales actualmente codificados (es decir, subvectores que han sido asignados realmente con bits para codificación), por ejemplo conforme a la ecuación (1). Los subvectores comprimidos son reenviados a un rechazador de subvector 44, el cual está configurado para rechazar subvectores residuales comprimidos que no cumplan un criterio de dispersión predeterminado, por ejemplo el criterio (3). Los subvectores comprimidos restantes son recopilados en un colector de subvector 46, el cual está configurado para concatenarlos para que formen un primer libro de códigos virtual VC1 (baja frecuencia). Los coeficientes procedentes del primer libro de códigos virtual VC1 son reenviados a un combinador de coeficientes 48, el cual está configurado para combinar pares de coeficientes para formar un segundo libro de códigos virtual VC2 (alta frecuencia), por ejemplo conforme a la ecuación (5). De ese modo, hasta ese punto, el rellenedor de espectro 40 incluye los mismos elementos que el generador de libro de códigos virtual 70 de alta frecuencia. Los subvectores residuales son reenviados también a un rellenedor de subvector 50, el cual está configurado para rellenar subvectores residuales no codificados por debajo

de una frecuencia predeterminada con coeficientes procedentes del primer libro de códigos virtual VC1, y para rellenar subvectores residuales no codificados por encima de la frecuencia predeterminada con coeficientes procedentes del segundo libro de códigos virtual. En una realización preferida, el rellenador de espectro 40 incluye también un ajustador de energía 52 configurado para ajustar la energía de los subvectores residuales no codificados rellenos, para obtener una atenuación perceptiva, según se ha descrito con anterioridad.

La Figura 13 es un diagrama de bloques que ilustra un ejemplo de realización de un descodificador 300 que incluye un rellenador de espectro 40. La estructura general del descodificador 300 es la misma que la del descodificador de la Figura 1, pero con el rellenador de ruido 30 reemplazado por el rellenador de espectro 40.

La Figura 14 es un diagrama de flujo que ilustra la generación de libro de códigos virtual de baja frecuencia. La etapa S1 comprime subvectores residuales actualmente codificados, por ejemplo conforme a la ecuación (1). La etapa S2 rechaza subvectores residuales comprimidos que sean demasiado dispersos, es decir subvectores residuales comprimidos que no cumplan un criterio de dispersión predeterminado, por ejemplo el criterio (3). La etapa S3 concatena los subvectores residuales comprimidos restantes para formar el libro de códigos virtual VC1.

La Figura 15 es un diagrama de flujo que ilustra la generación de libro de códigos virtual de alta frecuencia. La etapa S1 comprime subvectores residuales actualmente codificados, por ejemplo conforme a la ecuación (1). La etapa S2 rechaza subvectores residuales comprimidos que sean demasiado dispersos, es decir subvectores residuales comprimidos que no cumplan un criterio de dispersión predeterminado, tal como el criterio (3). La etapa S3 concatena los subvectores residuales comprimidos restantes para formar un primer libro de códigos virtual VC1. De ese modo, hasta ese punto, la generación de libro de códigos virtual de alta frecuencia incluye las mismas etapas que la generación de libro de códigos virtual de baja frecuencia. La etapa S4 combina pares de coeficientes del primer libro de códigos virtual VC1, por ejemplo conforme a la ecuación (5), formando con ello el libro de códigos virtual VC2 de alta frecuencia.

La Figura 16 es un diagrama de flujo que ilustra el relleno de espectro. La etapa S1 comprime subvectores residuales actualmente codificados, por ejemplo conforme a la ecuación (1). La etapa S2 rechaza subvectores residuales comprimidos que sean demasiado dispersos, es decir, subvectores residuales comprimidos que no cumplan un criterio de dispersión predeterminado, tal como el criterio (3). La etapa S3 concatena los subvectores residuales comprimidos restantes para formar un primer libro de códigos virtual VC1. La etapa S4 combina pares de coeficientes del primer libro de códigos virtual VC1, por ejemplo conforme a la ecuación (5), para formar un segundo libro de códigos virtual VC2. De ese modo, hasta ese punto, el relleno de espectro incluye las mismas etapas que la generación de libro de códigos virtual de alta frecuencia. La etapa S5 rellena subvectores residuales no codificados por debajo de una frecuencia predeterminada con coeficientes procedentes del primer libro de códigos virtual VC1. La etapa S6 rellena subvectores residuales no codificados por encima de una frecuencia predeterminada con coeficientes procedentes del segundo libro de códigos virtual VC2. La etapa S7 opcional ajusta la energía de subvectores residuales no codificados rellenos para obtener una atenuación perceptiva, según se ha descrito con anterioridad.

La Figura 17 es un diagrama de bloques que ilustra un ejemplo de realización de un generador de libro de códigos virtual 60 de baja frecuencia. Esta realización se basa en un procesador 110, por ejemplo un microprocesador, que ejecuta una componente de software 120 para comprimir subvectores residuales actualmente codificados, una componente de software 130 para rechazar subvectores residuales comprimidos que sean demasiado dispersos, y una componente de software 140 para concatenar los restantes subvectores residuales comprimidos para formar el libro de códigos virtual VC1. Estas componentes de software están almacenadas en la memoria 150. El procesador 110 comunica con la memoria a través de un bus de sistema. Los subvectores residuales son recibidos por el controlador 160 de entrada/salida (E/S) que controla un bus de E/S, al que están conectados el procesador 110 y la memoria 150. En esta realización, los subvectores residuales recibidos por el controlador de E/S 160 son almacenados en la memoria 150, donde éstos son procesados por medio de las componentes de software. La componente de software 120 puede implementar la funcionalidad del bloque 42 en la realización descrita con referencia a la Figura 10 anterior. La componente de software 130 puede implementar la funcionalidad del bloque 44 en la realización descrita con referencia a la Figura 10 anterior. La componente de software 140 puede implementar la funcionalidad del bloque 46 en la realización descrita con referencia a la Figura 10 anterior. El libro de códigos virtual VC1 obtenido a partir de la componente de software 140 es presentado a la salida desde la memoria 150 por el controlador de E/S 160 a través del bus de E/S o se almacena en la memoria 150.

La Figura 18 es un diagrama de bloques que ilustra un ejemplo de realización de un generador de libro de códigos virtual 70 de alta frecuencia. Esta realización se basa en un procesador 110, por ejemplo un microprocesador, que ejecuta una componente de software 120 para comprimir subvectores residuales actualmente codificados, una componente de software 130 para rechazar subvectores residuales comprimidos que sean demasiado dispersos, una componente de software 140 para concatenar los restantes subvectores residuales comprimidos para formar el libro de códigos virtual VC1 de baja frecuencia, y una componente de software 170 para combinar pares de coeficientes a partir del libro de códigos VC1 para formar el libro de códigos virtual VC2 de alta frecuencia. Estas componentes de software se almacenan en la memoria 150. El procesador 110 comunica con la memoria a través de un bus de sistema. Los subvectores residuales son recibidos por un controlador 160 de entrada/salida (E/S) que controla un bus de E/S, al que están conectados el procesador 110 y la memoria 150. En esta realización, los

subvectores residuales recibidos por el controlador de E/S 160 son almacenados en la memoria 150, donde son procesados por las componentes de software. La componente de software 120 puede implementar la funcionalidad del bloque 42 en la realización descrita con referencia a la Figura 11 anterior. La componente de software 130 puede implementar la funcionalidad del bloque 44 en las realizaciones descritas con referencia a la Figura 11 anterior. La componente de software 140 puede implementar la funcionalidad del bloque 46 en la realización descrita con referencia a la Figura 11 anterior. La componente de software 170 puede implementar la funcionalidad del bloque 48 en la realización descrita con referencia a la Figura 11 anterior. El libro de códigos virtual VC1 obtenido a partir de la componente de software 140 se almacena preferentemente en la memoria 150 para este propósito. El libro de códigos virtual VC2 obtenido a partir de la componente de software 170 se presenta a la salida desde la memoria 150 por parte del controlador de E/S 160 a través del bus de E/S o se almacena en la memoria 150.

La Figura 19 es un diagrama de bloques que ilustra un ejemplo de realización de un rellenedor de espectro 40. Esta realización se basa en un procesador 110, por ejemplo un microprocesador, que ejecuta una componente de software 180 para generar un libro de códigos virtual VC1 de baja frecuencia, una componente de software 190 para generar un libro de códigos virtual VC2 de alta frecuencia, una componente de software 200 para rellenar subvectores residuales no codificados por debajo de una frecuencia predeterminada a partir del libro de códigos virtual VC1, y una componente de software 210 para rellenar subvectores residuales no codificados por encima de una frecuencia predeterminada a partir del libro de códigos virtual VC2. Estas componentes de software se almacenan en la memoria 150. El procesador 110 comunica con la memoria a través de un bus de sistema. Los subvectores residuales son recibidos por un controlador 160 de entrada/salida (E/S) que controla un bus de E/S, al que están conectados el procesador 110 y la memoria 150. En esta realización, los subvectores residuales recibidos por el controlador de E/S 160 se almacenan en la memoria 150, donde son procesados mediante las componentes de software. La componente de software 180 puede implementar la funcionalidad de los bloques 42-46 en la realización descrita con referencia a la Figura 12 anterior. La componente de software 190 puede implementar la funcionalidad del bloque 48 en las realizaciones descritas con referencia a la Figura 12 anterior. Las componentes de software 200, 210 pueden implementar la funcionalidad del bloque 50 en la realización descrita con referencia a la Figura 12 anterior. Los libros de códigos virtuales VC1, VC2 obtenidos a partir de las componentes de software 180 y 190 se almacenan con preferencia en la memoria 150 para este propósito. Los subvectores residuales rellenos obtenidos a partir de las componentes de software 200, 201 son presentados a la salida desde la memoria 150 por medio del controlador de E/S 160 a través del bus de E/S o se almacenan en la memoria 150.

La tecnología descrita en lo que antecede está destinada a ser usada en un descodificador de audio, el cual puede ser usado en un dispositivo móvil (por ejemplo, teléfono móvil, ordenador portátil) o en un PC estacionario. En esta memoria, el término Equipo de Usuario (UE) será usado como nombre genérico de tales dispositivos. Un descodificador de audio con el esquema de relleno de espectro que se ha propuesto puede ser usado en escenarios de comunicación en tiempo real (dirigidos principalmente a habla) o en escenarios de transmisión (dirigidos principalmente a música).

La Figura 20 ilustra una realización de un equipo de usuario conforme a la presente tecnología. Éste incluye un descodificador 300 equipado con un rellenedor de espectro 40 conforme a la presente tecnología. Esta realización ilustra un terminal de radio, pero también son factibles otros nodos de red. Por ejemplo, si se usa voz sobre IP (Protocolo de Internet) en la red, el equipo de usuario puede comprender un ordenador.

En el equipo de usuario de la Figura 20, una antena 302 recibe una señal de audio codificada. Una unidad de radio 304 transforma la señal en parámetros de audio, los cuales son reenviados al descodificador 300 para generar una señal de audio digital, según se ha descrito con referencia a las diversas realizaciones anteriores. La señal de audio digital se convierte a continuación en D/A y se amplifica en una unidad 306 y finalmente se envía a un altavoz 308.

Los expertos en la materia comprenderán que se pueden realizar diversas modificaciones y cambios en la presente tecnología sin apartarse del alcance de la misma, el cual se define en las reivindicaciones anexas.

#### Referencias:

[1] ITU-T Rec. G.719, "Low-complexity full-band audio coding for high-quality conversational applications", 2008, Secciones 8.4.1, 8.4.3.

[2] Mittal, J. Ashley, E. Cruz-Zeno, "Low Complexity Factorial Pulse Coding of MDCT Coefficients using Approximation of Combinatorial Functions", ICASSP 2007.

#### Abreviaturas

FPC	Codificación de Pulso Factorial
MDCT	Transformada Coseno Discreta Modificada
RMS	Valor Cuadrático Medio
UE	Equipo de Usuario
VC	Libro de Códigos Virtual



**REIVINDICACIONES**

1. Un aparato (40) para rellenar subvectores residuales no codificados de una señal de audio codificada por transformada, comprendiendo el aparato:

medios (42) para comprimir subvectores residuales codificados;

5 medios (44) para rechazar subvectores residuales comprimidos que no cumplan un criterio de dispersión predeterminado;

medios (46) para concatenar los subvectores residuales comprimidos restantes para formar un primer libro de códigos virtual (VC1);

10 medios (48) para combinar pares de coeficientes del primer libro de códigos virtual (VC1) para formar un segundo libro de códigos virtual (VC2),

medios (50) para rellenar subvectores residuales no codificados por debajo de una frecuencia predeterminada con coeficientes procedentes del primer libro de códigos virtual (VC1), y para rellenar subvectores residuales no codificados por encima de la frecuencia predeterminada con coeficientes procedentes del segundo libro de códigos virtual (VC2), caracterizado por que los medios de compresión (42) están configurados para comprimir componentes

15  $\hat{X}(k)$  de los subvectores residuales codificados conforme a:

$$Y(k) = \begin{cases} 1 & \text{si } \hat{X}(k) > 0 \\ 0 & \text{si } \hat{X}(k) = 0 \\ -1 & \text{si } \hat{X}(k) < 0 \end{cases}$$

o

$$Y(k) = \begin{cases} 1 & \text{si } \hat{X}(k) > T \\ 0 & \text{si } -T \leq \hat{X}(k) \leq T \\ -1 & \text{si } \hat{X}(k) < -T \end{cases}$$

20 donde  $Y(k)$  son las componentes de los subvectores residuales comprimidos y  $T$  es un número pequeño positivo que controla la cantidad de compresión.

2. El aparato conforme a la reivindicación 1, en el que los medios (44) de rechazo están configurados para rechazar subvectores residuales comprimidos que tienen menos de un porcentaje predeterminado de componentes distintas de cero.

25 3. El aparato conforme a la reivindicación 1 o 2, en donde los subvectores residuales comprimidos que no cumplan el criterio:

$$\sum_{k=0}^{M-1} |Y(k)| \geq 2,$$

donde la dimensión  $M$  del subvector es 8, son rechazados.

4. El aparato conforme a una cualquiera de las reivindicaciones 1 a 3, en el que los medios de combinación (48) están configurados para combinar pares de coeficientes  $Y(k)$  del primer libro de códigos virtual (VC1) conforme a:

$$Z(k) = \begin{cases} \text{sign}(Y(k)) \times (|Y(k)| + |Y(N-k)|) & \text{si } Y(k) \neq 0 \\ Y(N-k) & \text{si } Y(k) = 0 \end{cases} \quad k = 0 \dots N-1$$

30 donde  $N$  es el tamaño del primer libro de códigos virtual (VC1) y  $Z(k)$  son las componentes del segundo libro de códigos virtual (VC2).

5. El aparato conforme a una cualquiera de las reivindicaciones 1 a 4, que comprende además medios (52) para ajustar la energía de los subvectores residuales no codificados rellenados para obtener una atenuación perceptual.

6. Un descodificador (300) de audio, que comprende el aparato (40) conforme a una cualquiera de las reivindicaciones 1 a 5.

5 7. Un equipo de usuario (UE) que comprende el descodificador de audio conforme a la reivindicación 6.

8. Un método para rellenar subvectores residuales no codificados de una señal de audio codificada por transformada, comprendiendo el método:

comprimir (S1) subvectores residuales codificados;

rechazar (S2) subvectores residuales comprimidos que no cumplan un criterio de dispersión predeterminado;

10 concatenar (S3) los subvectores residuales comprimidos restantes para formar un primer libro de códigos virtual (VC1);

combinar (S4) pares de coeficientes del primer libro de códigos virtual (VC1) para formar un segundo libro de códigos virtual (VC2);

15 rellenar (S5) subvectores residuales no codificados por debajo de una frecuencia predeterminada con coeficientes procedentes del primer libro de códigos virtual (VC1);

rellenar (S6) subvectores residuales no codificados por encima de la frecuencia predeterminada con coeficientes procedentes del segundo libro de códigos virtual,

caracterizado por que las componentes  $\hat{X}(k)$  de los subvectores residuales codificados se comprimen (S1) conforme a:

$$Y(k) = \begin{cases} 1 & \text{si } \hat{X}(k) > 0 \\ 0 & \text{si } \hat{X}(k) = 0 \\ -1 & \text{si } \hat{X}(k) < 0 \end{cases}$$

20

o

$$Y(k) = \begin{cases} 1 & \text{si } \hat{X}(k) > T \\ 0 & \text{si } -T \leq \hat{X}(k) \leq T \\ -1 & \text{si } \hat{X}(k) < -T \end{cases}$$

donde  $Y(k)$  son las componentes de los subvectores residuales comprimidos y  $T$  es un número pequeño positivo que controla la cantidad de compresión.

25

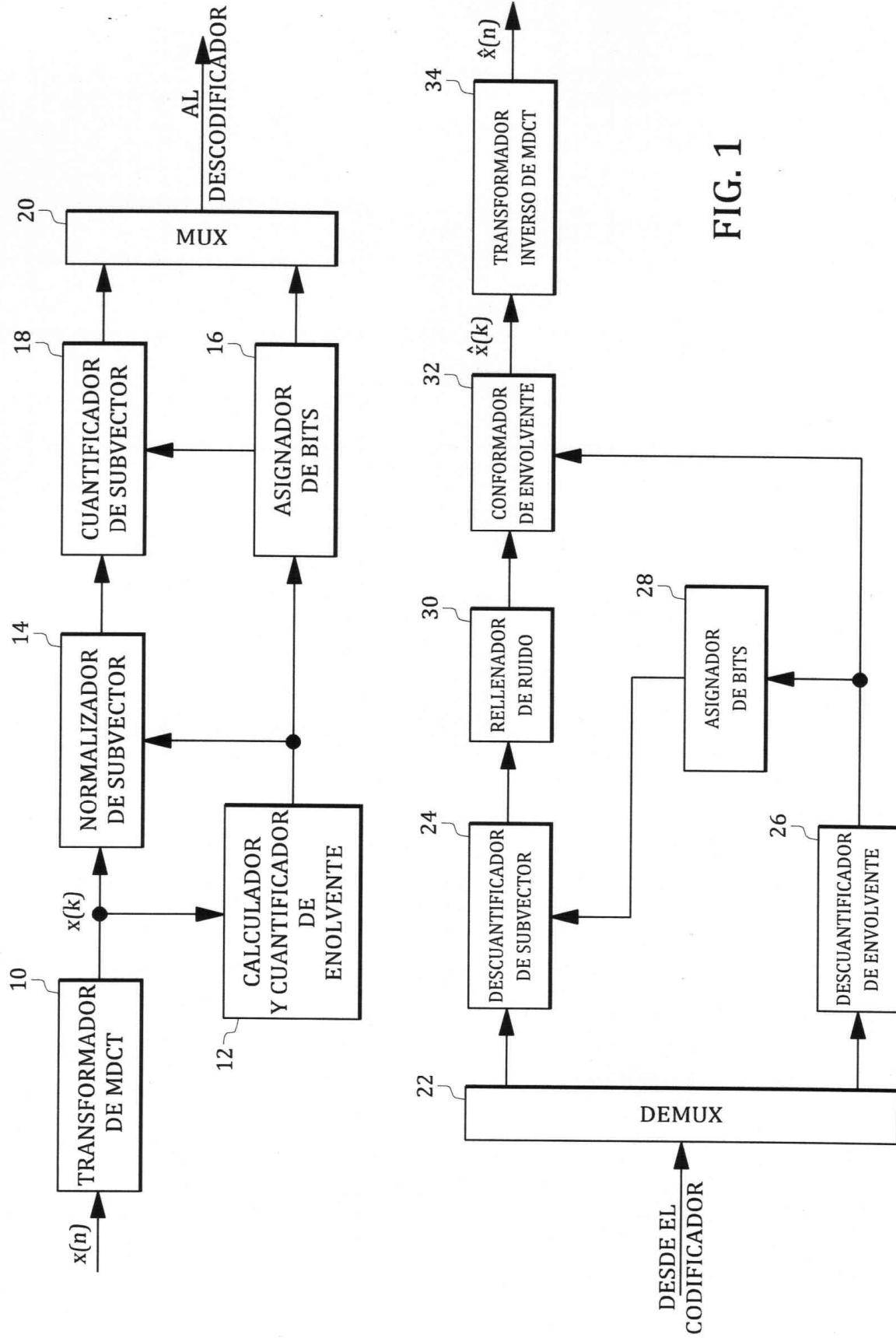


FIG. 1

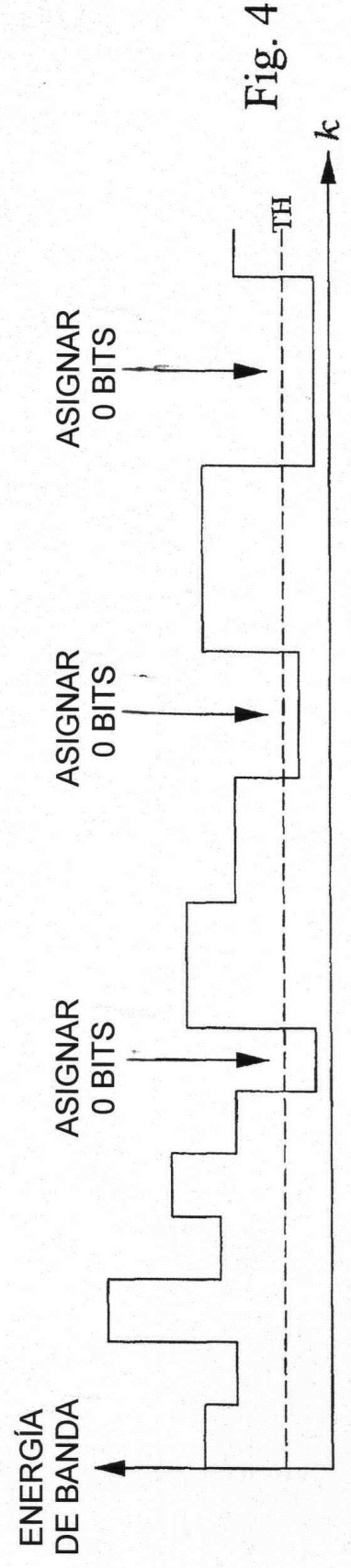
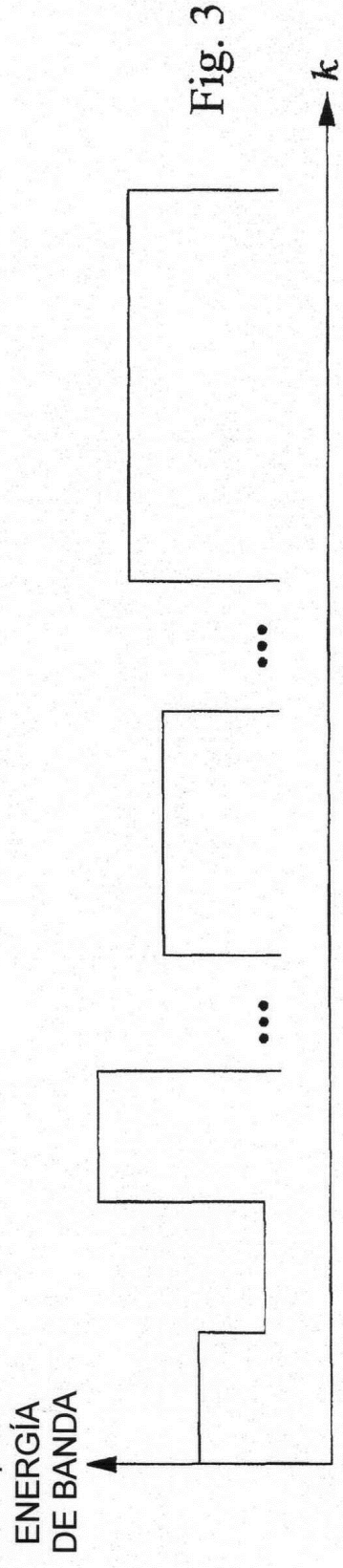
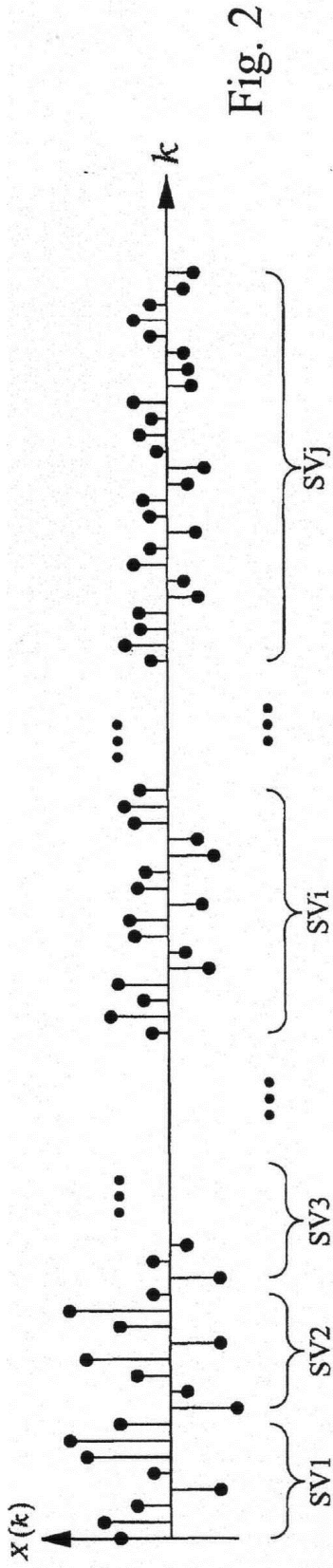




Fig. 5

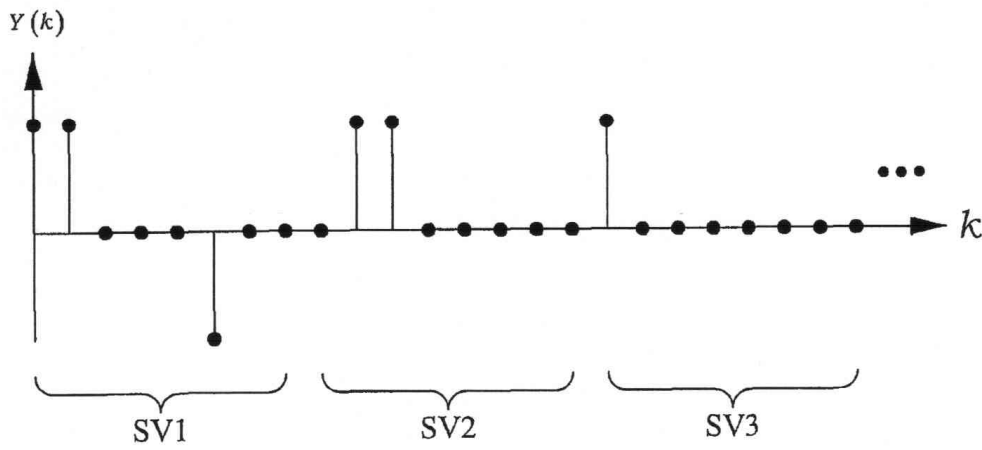


Fig. 6

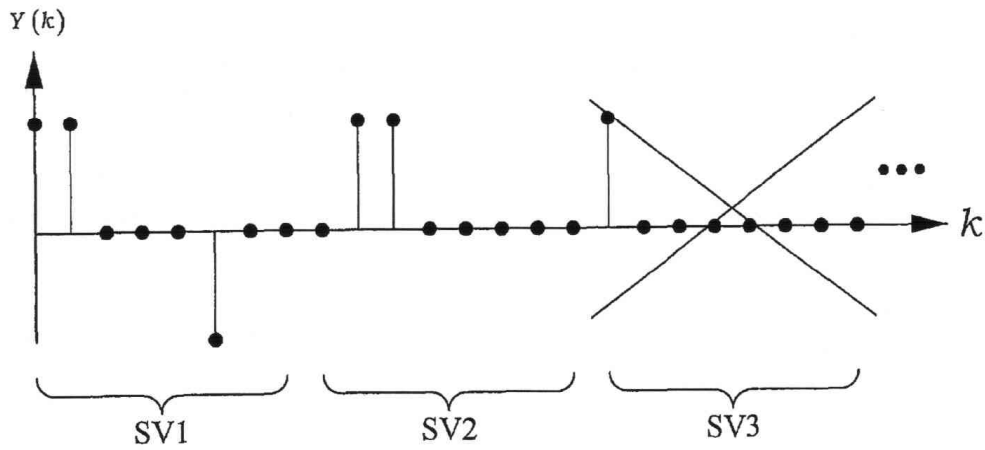


Fig. 7

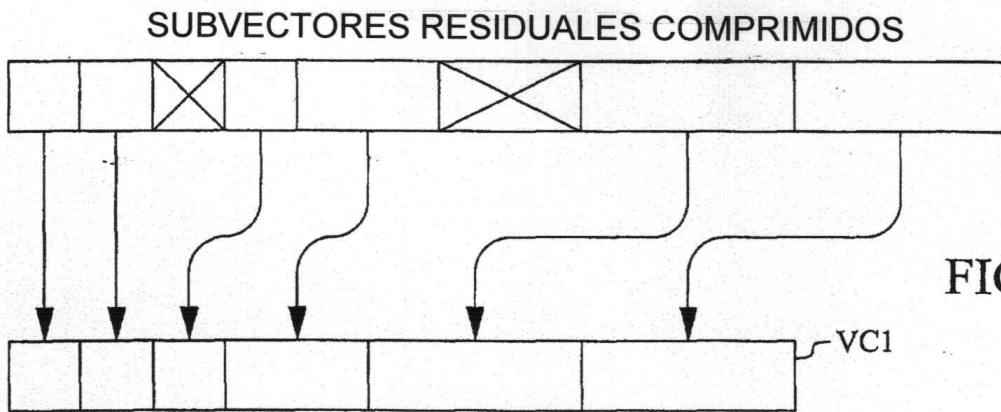


FIG. 8

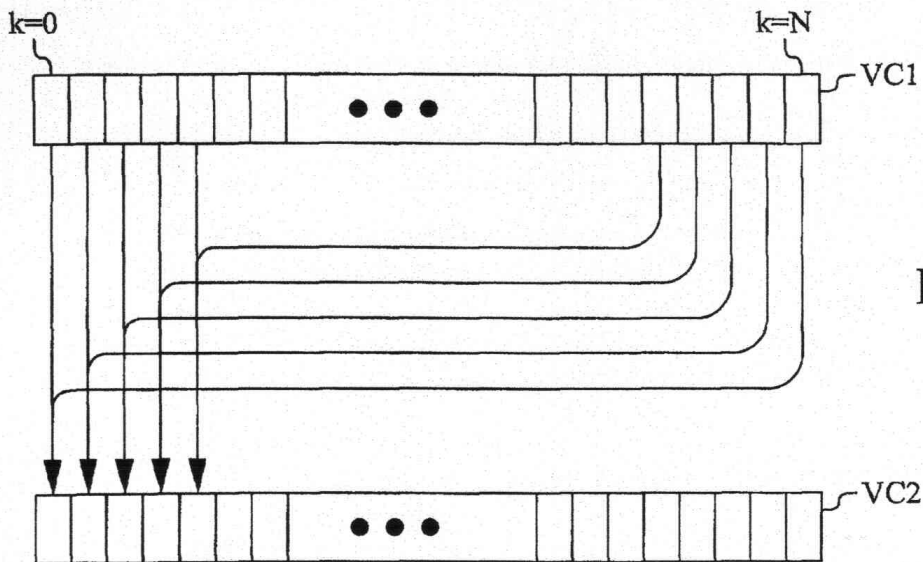


FIG. 9A

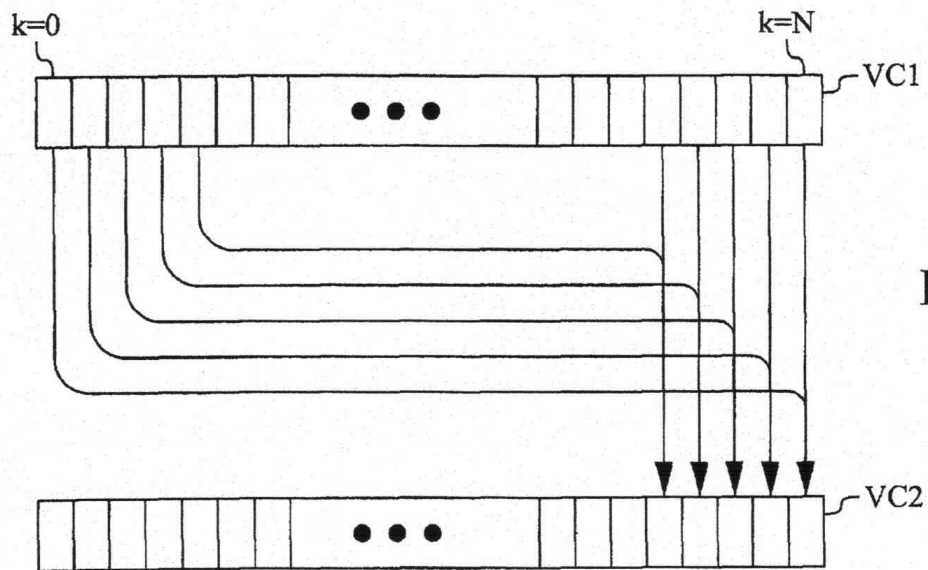


FIG. 9B

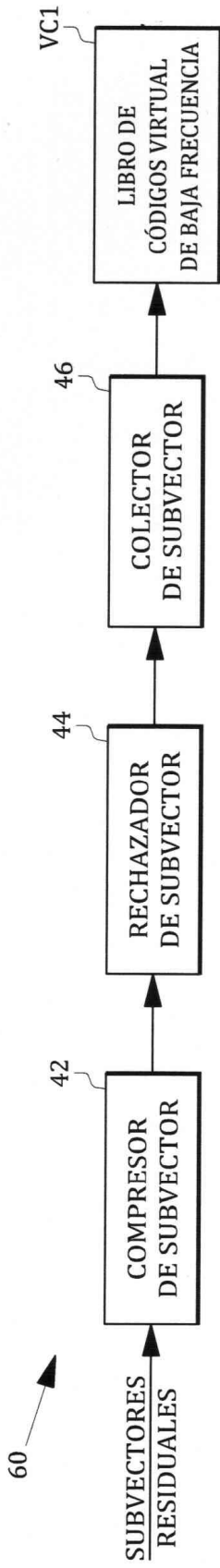


FIG. 10

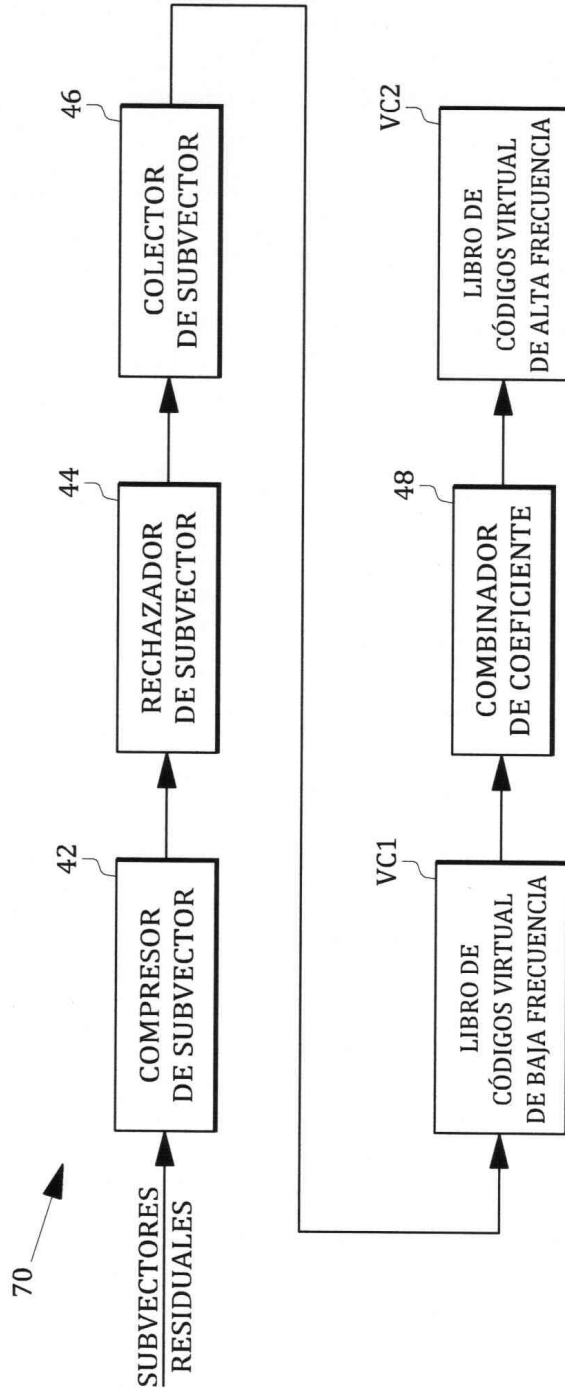


FIG. 11

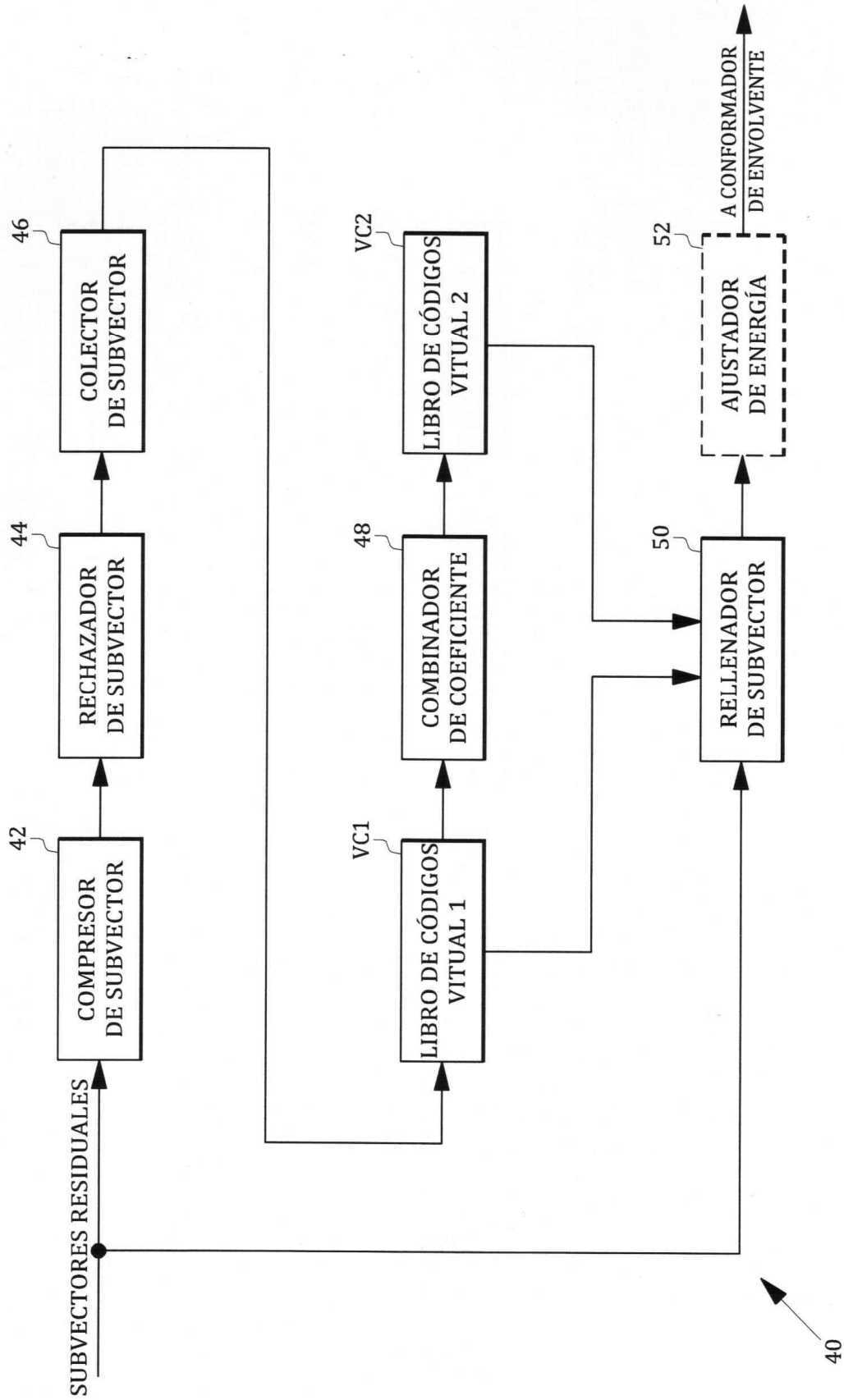


FIG. 12



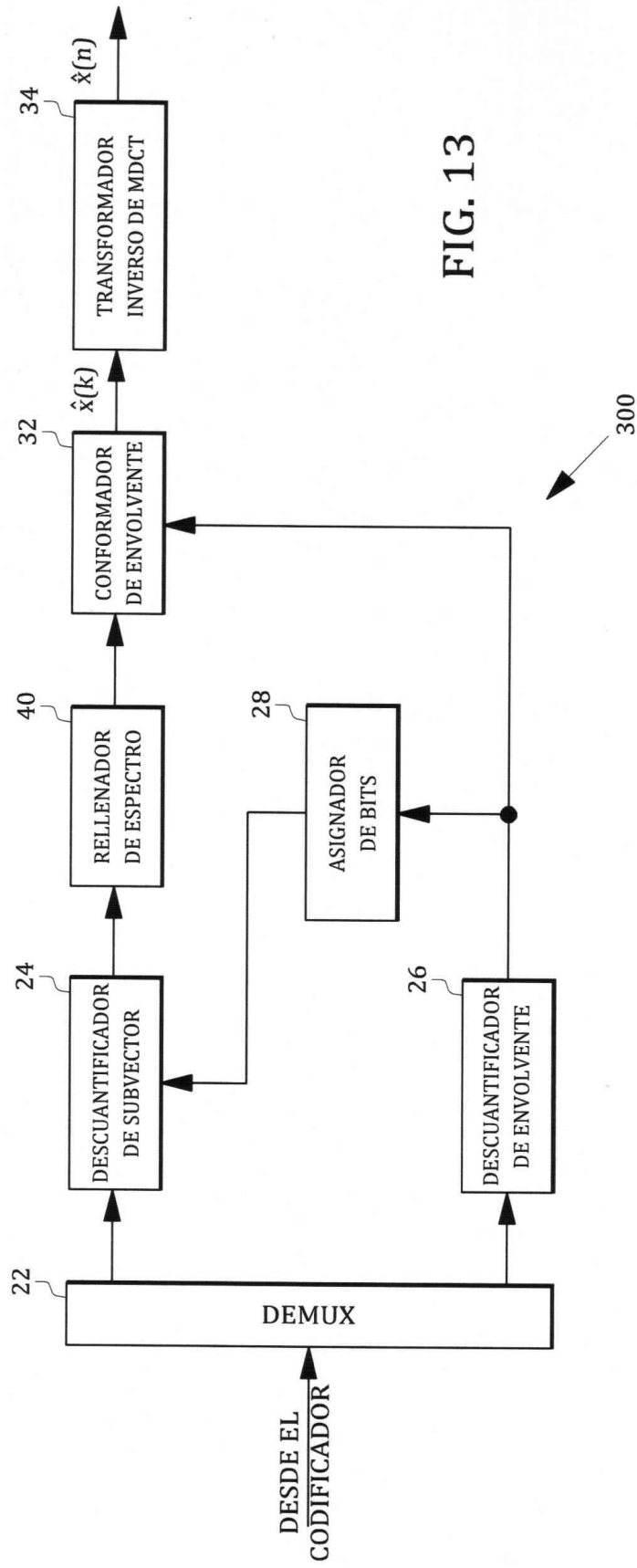


FIG. 13

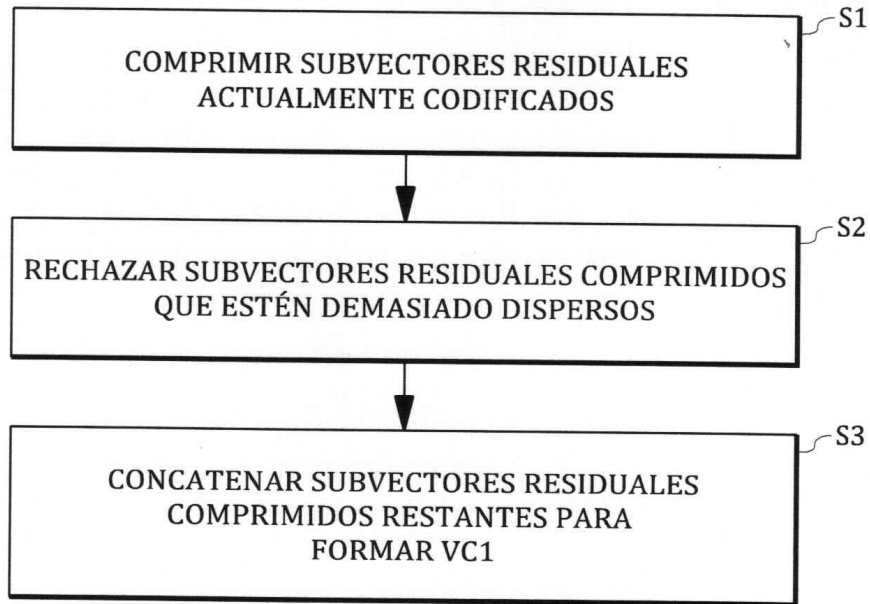


FIG. 14

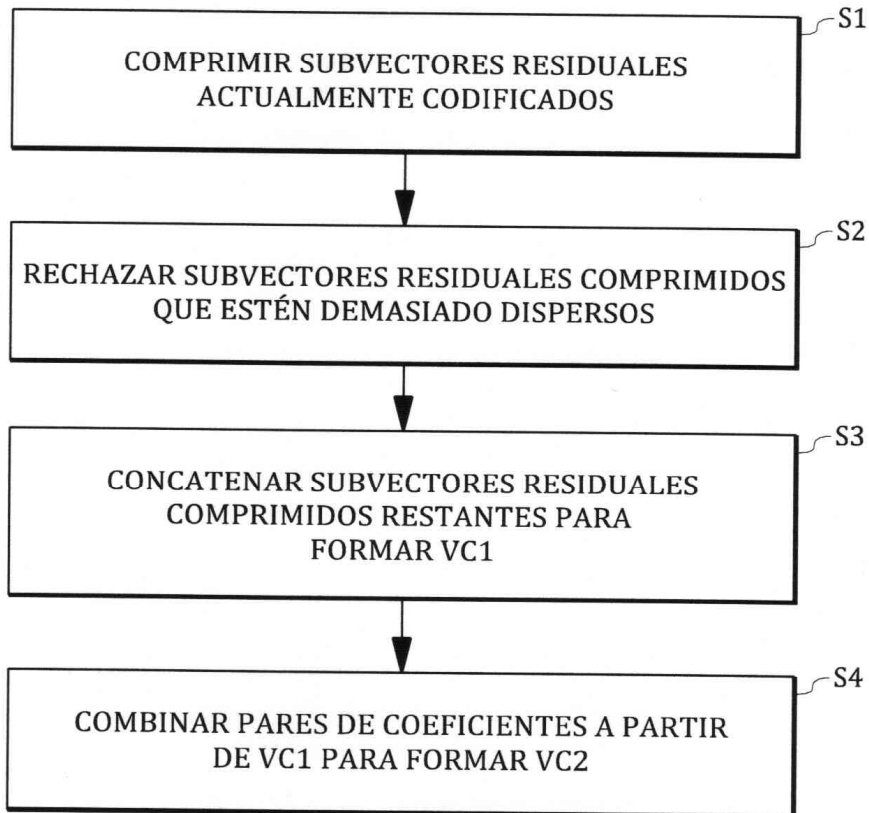


FIG. 15

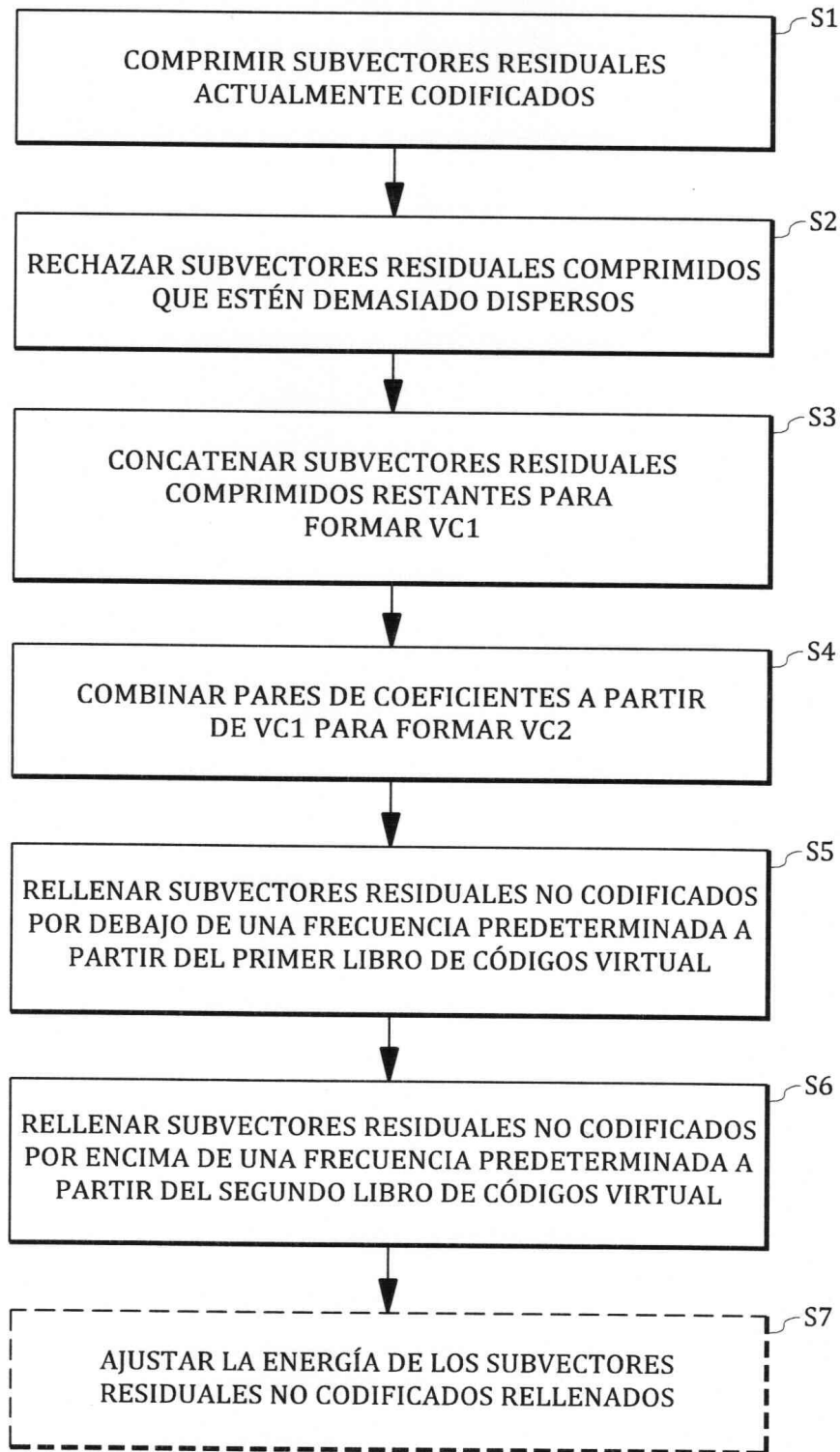


FIG. 16

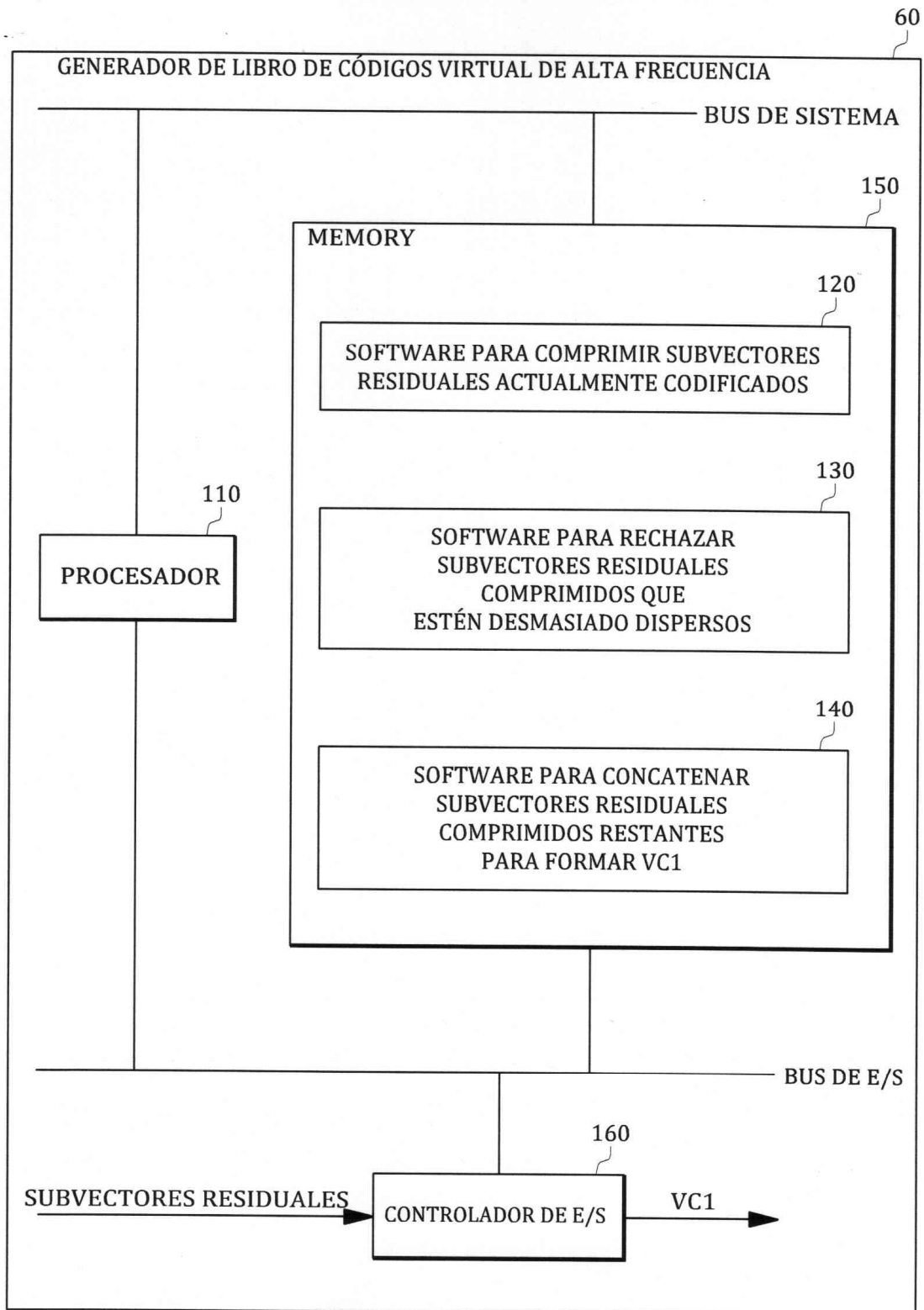


FIG. 17

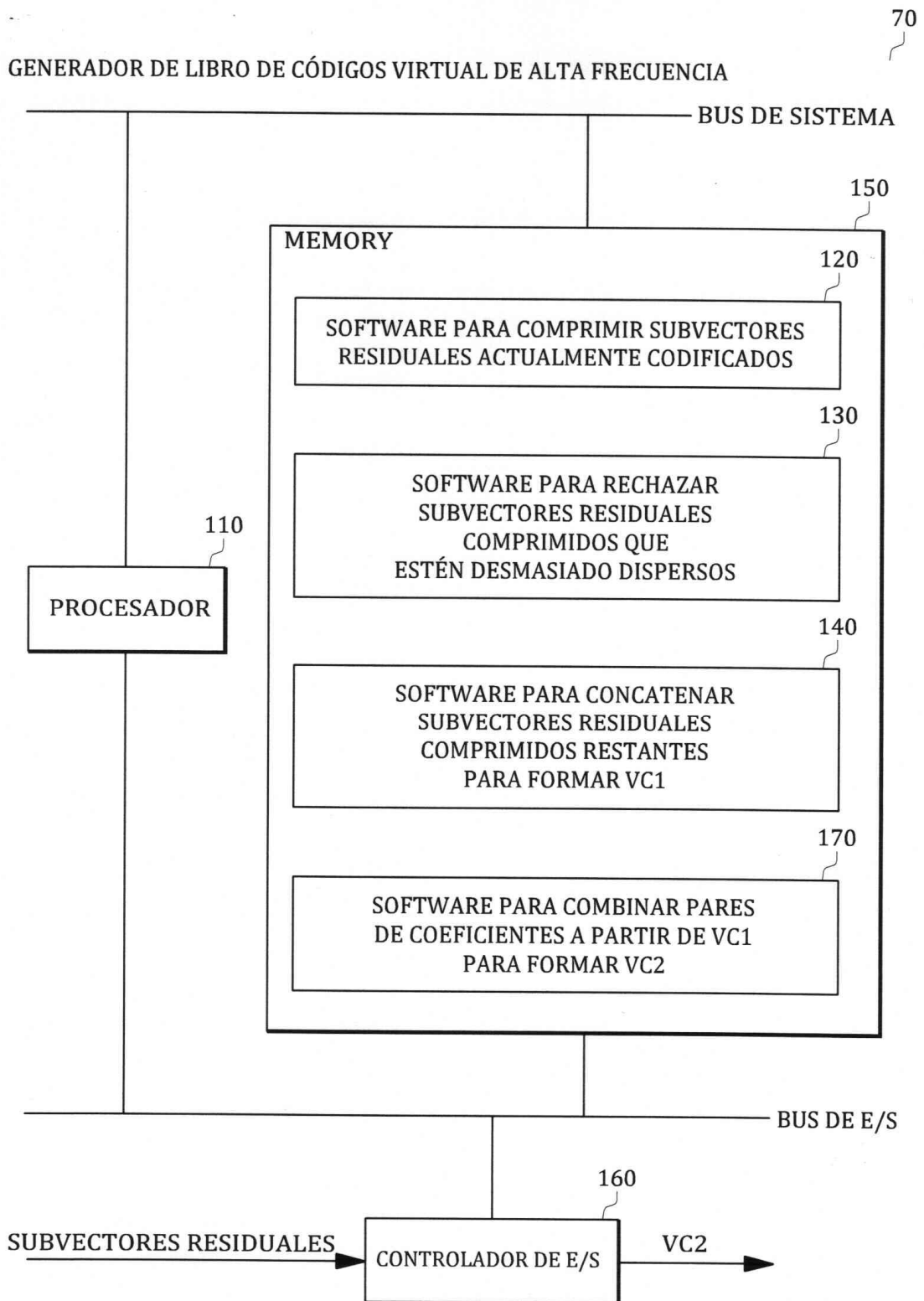


FIG. 18

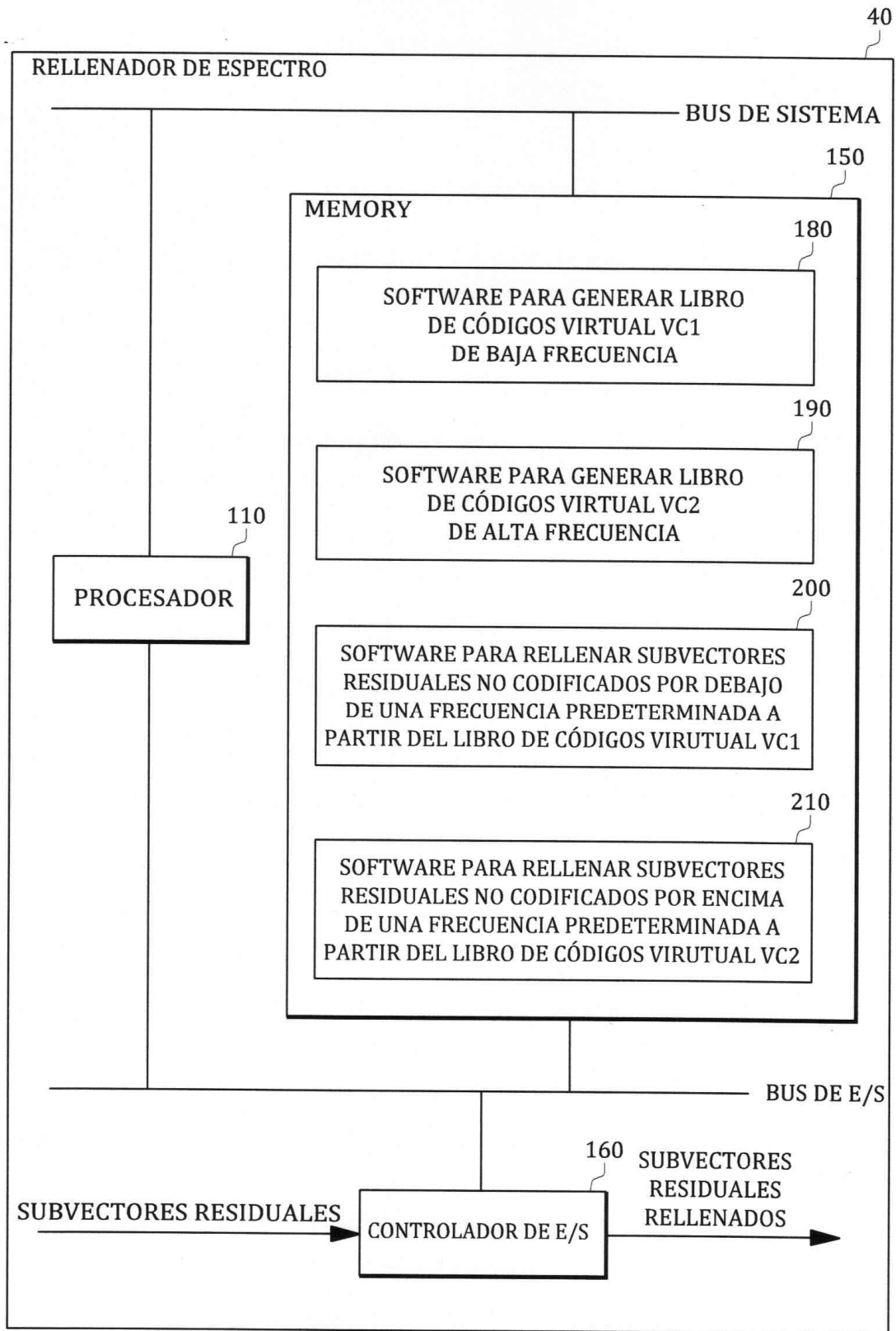


FIG. 19

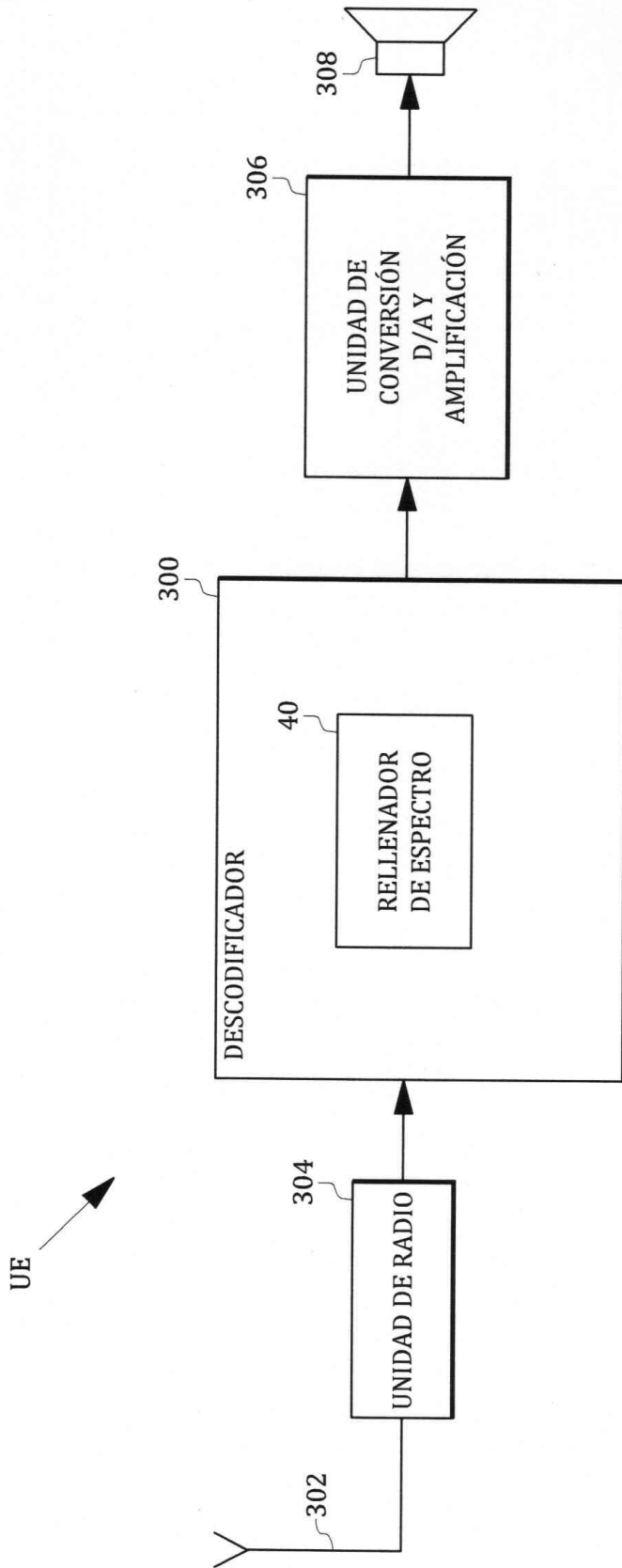


FIG. 20