



19



OFICINA ESPAÑOLA DE  
PATENTES Y MARCAS

ESPAÑA

11 Número de publicación: **2 347 473**

51 Int. Cl.:  
**G10L 19/14** (2006.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

96 Número de solicitud europea: **06850882 .9**

96 Fecha de presentación : **05.12.2006**

97 Número de publicación de la solicitud: **1958187**

97 Fecha de publicación de la solicitud: **20.08.2008**

54 Título: **Procedimiento y aparato de detección de componentes tonales de señales de audio.**

30 Prioridad: **05.12.2005 US 742846 P**

45 Fecha de publicación de la mención BOPI:  
**29.10.2010**

45 Fecha de la publicación del folleto de la patente:  
**29.10.2010**

73 Titular/es: **QUALCOMM Incorporated**  
**5775 Morehouse Drive**  
**San Diego, California 92121, US**

72 Inventor/es: **Manjunath, Sharath y**  
**Kandhadai, Ananthapadmanabhan**

74 Agente: **Carpintero López, Mario**

**ES 2 347 473 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

Procedimiento y aparato de detección de componentes tonales de señales de audio.

5 **Solicitudes relacionadas****Campo**

La presente divulgación se refiere al procesamiento de señales.

10 **Antecedentes**

La transmisión de voz mediante técnicas digitales se ha generalizado, particularmente en la telefonía de larga distancia, la telefonía por conmutación de paquetes tal como voz sobre IP (VoIP) y la telefonía de radio digital tal como la telefonía celular. Tal proliferación ha creado interés en determinar la cantidad mínima de información que puede enviarse por un canal manteniendo al mismo tiempo la calidad percibida de la voz reconstruida. Si la voz se transmite simplemente mediante muestro y digitalización, puede requerirse una velocidad de transmisión de datos del orden de sesenta y cuatro kilobits por segundo (kbps) para conseguir una calidad de voz comparable a la de un teléfono por cable analógico convencional. Sin embargo, a través del uso del análisis de voz, seguido de una codificación, transmisión y resíntesis apropiadas en el receptor, puede conseguirse una reducción significativa en la velocidad de transmisión de datos.

Los dispositivos que están configurados para comprimir la voz mediante la extracción de parámetros relacionados con un modelo de generación de voz humana se denominan “codificadores de voz”. Un codificador de voz incluye normalmente un codificador y un decodificador. El codificador divide la señal de voz entrante en bloques de tiempo (o “tramas”), analiza cada trama para extraer determinados parámetros relevantes y cuantifica los parámetros en una representación binaria, tal como un conjunto de bits o un paquete de datos binarios. Los paquetes de datos se transmiten por el canal de comunicación (es decir, una conexión de red cableada o inalámbrica) hasta un receptor que incluye un decodificador. El decodificador recibe y procesa paquetes de datos, los descuantifica para generar los parámetros y vuelve a crear tramas de voz utilizando los parámetros descuantificados.

La función del codificador de voz es comprimir la señal de voz digitalizada en una señal de baja velocidad binaria eliminando las redundancias naturales que son inherentes en la voz. La compresión digital se consigue representando la trama de voz de entrada con un conjunto de parámetros y utilizando cuantificación para representar los parámetros con un conjunto de bits. Si la trama de voz de entrada presenta un número de bits  $N_i$  y el paquete de datos correspondiente generado por el codificador de voz presenta un número de bits  $N_o$ , el factor de compresión conseguido por el codificador de voz es  $C_r = N_i/N_o$ . El reto es mantener una alta calidad de voz de la voz descodificada obteniendo al mismo tiempo el factor de compresión objetivo. El rendimiento de un codificador de voz depende de (1) la calidad del funcionamiento del modelo de voz, o de la combinación del proceso de análisis y el proceso de síntesis descritos anteriormente, y (2) de la calidad del proceso de cuantificación de parámetros llevado a cabo a la velocidad binaria objetivo de  $N_o$  bits por trama. El objetivo del modelo de voz es por tanto capturar el contenido de información de la señal de voz, para proporcionar una calidad de voz objetivo, con un pequeño conjunto de parámetros para cada trama.

Los codificadores de voz pueden implementarse como codificadores en el dominio de tiempo, los cuales tratan de capturar la forma de onda de la voz en el dominio de tiempo empleando un procesamiento de alta resolución en el tiempo para codificar pequeños segmentos de voz (normalmente subtramas de cinco milisegundos (ms)) a la vez. Para cada subtrama se obtiene un representante de alta precisión a partir de un espacio de libro de códigos mediante varios algoritmos de búsqueda conocidos en la técnica. Como alternativa, los codificadores de voz pueden implementarse como codificadores en el dominio de frecuencia, los cuales llevan a cabo un proceso de análisis para capturar el espectro de voz a corto plazo de la trama de voz de entrada con un conjunto de parámetros y utilizan un proceso de síntesis correspondiente para volver a crear la forma de onda de voz a partir de los parámetros espectrales. El cuantificador de parámetros conserva los parámetros representándolos con representaciones almacenadas de vectores de código según técnicas de cuantificación conocidas, tales como las descritas en el documento *Vector Quantization and Signal Compression* (1992) de A. Gersho y R.M. Gray.

Un codificador de voz en el dominio de tiempo ampliamente conocido es el codificador predictivo lineal excitado por código (CELP). Un ejemplo de un codificador de este tipo se describe en el documento *Digital Processing of Speech Signals* 396-453 (1978), de L.B. Rabiner y R.W. Schafer. En un codificador CELP se eliminan las correlaciones, o redundancias, a corto plazo de la señal de voz mediante un análisis de predicción lineal (LP), el cual obtiene los coeficientes de un filtro de formantes a corto plazo. Aplicando el filtro de predicción a corto plazo a la trama de voz entrante se genera una señal de residuo LP, la cual se modela y se cuantifica adicionalmente con parámetros de filtro de predicción a largo plazo y un libro de códigos estocástico posterior. Por lo tanto, la codificación CELP divide la tarea de codificar la forma de onda de voz de dominio de tiempo en distintas tareas de codificación de los coeficientes de filtro a corto plazo LP y de codificación del residuo LP. La codificación en el dominio de tiempo puede llevarse a cabo a una velocidad fija (es decir, usando el mismo número de bits  $N_o$  para cada trama) o a una velocidad variable (en la que se utilizan diferentes velocidades binarias para diferente tipos de contenidos de trama). Los codificadores de velocidad variable tratan de utilizar solamente la cantidad de bits necesaria para codificar los parámetros de códec

## ES 2 347 473 T3

a un nivel adecuado para obtener una calidad objetivo. Un codificador CELP de velocidad variable a modo de ejemplo se describe en la patente estadounidense número 5.414.796 (Jacobs *et al.*, expedida el 9 de mayo de 1995).

5 Los codificadores en el dominio de tiempo, tales como el codificador CELP, se basan normalmente en un alto número de bits  $N_0$  por trama para conservar la exactitud de la forma de onda de la voz en el dominio de tiempo. Tales codificadores proporcionan normalmente una excelente calidad de voz siempre que el número de bits  $N_0$  por trama sea relativamente grande (por ejemplo, 8 kbps o superior), y se utilizan de manera satisfactoria en aplicaciones comerciales de alta velocidad. Sin embargo, a bajas velocidades binarias (4 kbps o inferiores), un codificador en el dominio de tiempo puede fallar a la hora de conservar una alta calidad y un rendimiento robusto debido al número  
10 limitado de bits disponibles. Por ejemplo, el limitado espacio de libro de códigos disponible a una baja velocidad binaria puede afectar a la capacidad de correspondencia de formas de onda de un codificador convencional en el dominio de tiempo.

15 Un codificador de voz puede estar configurado para seleccionar un modo y/o una velocidad de codificación particulares según una o más cualidades de la señal que va a codificarse. Por ejemplo, un codificador de voz puede estar configurado para distinguir tramas que contienen voz de tramas que contienen señales que no son de voz, tales como tonos de señalización, y para utilizar diferentes modos de codificación para codificar las tramas de voz y las tramas que no son de voz.

20 La patente estadounidense número 4.689.760 (Lee *et al.*) da a conocer un procedimiento que puede aplicarse en la descodificación de audio para discriminar señales DTMF con respecto a señales de voz estableciendo umbrales de la energía residual de predicción lineal como una medida de ganancia de codificación. En particular, la velocidad del descenso de energía residual desde la primera hasta la última iteración de las iteraciones llevadas a cabo para una única trama de señal de audio se toma como una indicación para la discriminación voz-tono.

### 25 **Resumen**

Según la invención se proporciona un procedimiento de procesamiento de señales de audio como el expuesto en la reivindicación 1, un medio de almacenamiento de datos como el expuesto en la reivindicación 17 y un aparato  
30 para procesar señales de audio como el expuesto en la reivindicación 18. Realizaciones preferidas se exponen en las reivindicaciones dependientes.

### **Breve descripción de los dibujos**

35 La Figura 1 muestra un ejemplo de un espectro de una señal de voz.

La Figura 2 muestra un ejemplo de un espectro de una señal tonal.

40 La Figura 3 muestra un diagrama de flujo de un procedimiento M100 según una configuración desvelada.

La Figura 4A muestra un diagrama esquemático de una realización de forma directa de un filtro de síntesis.

La Figura 4B muestra un diagrama esquemático de una realización reticular de un filtro de síntesis.

45 La Figura 5 muestra un diagrama de flujo de una implementación M110 del procedimiento M100.

La Figura 6 muestra un listado de pseudocódigo de una implementación del algoritmo de Leroux-Gueguen.

50 La Figura 7 muestra un listado de pseudocódigo que incluye implementaciones de tareas T100 y T200.

La Figura 8 muestra un ejemplo de una estructura lógica para una tarea T300.

Las Figuras 9A y 9B muestran ejemplos de diagramas de flujo para la tarea T300.

55 La Figura 10 muestra un listado de pseudocódigo que incluye implementaciones de las tareas T100, T200 y T300.

La Figura 11 muestra un ejemplo de un módulo lógico para la tarea T300.

60 La Figura 12 muestra un ejemplo de un procedimiento de prueba para una configuración de una tarea T400.

La Figura 13 muestra un diagrama de flujo de una implementación de la tarea T400.

65 La Figura 14 muestra gráficos de una medida de ganancia  $G_i$  frente al índice de iteración  $i$  para cuatro ejemplos diferentes A a D de porciones de tiempo.

La Figura 15 muestra un ejemplo de una estructura lógica para la tarea T400.

La Figura 16A muestra un diagrama de bloques de un aparato A100 según una configuración desvelada.

## ES 2 347 473 T3

La Figura 16B muestra un diagrama de bloques de una implementación A200 del aparato A100.

La Figura 17 muestra un diagrama de un sistema de telefonía celular.

5 La Figura 18 muestra un diagrama de un sistema que incluye dos codificadores y dos descodificadores.

La Figura 19A muestra un diagrama de bloques de un codificador.

La Figura 19B muestra un diagrama de bloques de un descodificador.

10

La Figura 20 muestra un diagrama de flujo de tareas para la selección de modo.

La Figura 21 muestra un diagrama de flujo de otra implementación de la tarea T400.

15

La Figura 22 muestra un diagrama de flujo de una implementación adicional de la tarea T400.

### Descripción detallada

20 Sistemas, procedimientos y aparatos para la detección de señales que presentan picos espectrales con un ancho de banda estrecho (también denominados “componentes tonales” o “tonos”) se describen en este documento con referencia a las reivindicaciones adjuntas. El alcance de las configuraciones descritas incluye implementaciones que llevan a cabo tal detección utilizando parámetros de un esquema de análisis de codificación de predicción lineal (LPC) como el utilizado normalmente en codificadores de voz, reduciendo de ese modo la complejidad computacional a diferencia de un enfoque que utiliza un detector de tonos aparte.

25

A no ser que quede expresamente limitado por su contexto, el término “calcular” se utiliza en este documento para indicar cualquiera de sus significados habituales, tal como computar, generar y seleccionar a partir de una lista de valores. Cuando se utilice el término “comprender” en la presente descripción y en las reivindicaciones, no excluye otros elementos u operaciones. El término “A se basa en B” se utiliza para indicar cualquiera de sus significados habituales, incluyendo los casos (i) “A es igual a B” y (ii) “A se basa al menos en B”.

30

Ejemplos de tonos incluyen señales especiales encontradas con frecuencia en telefonía, tales como tonos de progreso de llamada (por ejemplo, un tono de llamada, una señal de ocupado, un tono de no disponibilidad de número, un tono de protocolo de facsímil u otro tono de señalización). Otros ejemplos de componentes tonales son las señales de multifrecuencia de doble tono (DTMF), las cuales incluyen una frecuencia del conjunto {697 Hz, 770 Hz, 852 Hz, 941 Hz} y una frecuencia del conjunto {1209 Hz, 1336 Hz, 1477 Hz, 1633 Hz}. Tales señales DTMF se utilizan comúnmente para la señalización de marcación por tonos. También es común que un usuario utilice un teclado para generar tonos DTMF durante una llamada telefónica para interactuar con un sistema automático en el otro extremo de la llamada, tal como un sistema de correo de voz u otro sistema que tenga un mecanismo de selección automático tal como un menú.

40

En general, una señal tonal se define por los inventores como una señal que contiene muy pocos tonos (por ejemplo, menos de ocho). La envolvente espectral de una señal tonal presenta picos agudos en las frecuencias de esos tonos, donde el ancho de banda de la envolvente espectral en torno a un pico de este tipo (como los mostrados en el ejemplo de la Figura 2) es mucho más pequeño que el ancho de banda de la envolvente espectral en torno a un pico normal de una señal de voz (como los mostrados en el ejemplo de la Figura 1). Por ejemplo, el ancho de banda de 3 dB de un pico correspondiente a un componente tonal puede ser inferior a 100 Hz y puede ser inferior a 50 Hz, 20 Hz, 10 Hz o incluso 5 Hz.

45

Puede ser deseable detectar si la entrada de señal en un codificador de voz es una señal tonal en lugar de algún tipo de señal de voz. Normalmente, las señales tonales no pasan muy bien a través de un codificador de voz, especialmente a bajas velocidades binarias, y el resultado después de la descodificación no suena normalmente como los tonos. Las envolventes espectrales de las señales tonales son diferentes de las envolventes de las señales de voz, y los procesos de clasificación tradicionales de los códecs de voz pueden fallar a la hora de seleccionar un modo de codificación adecuado para tramas que contengan componentes tonales. Por lo tanto, puede ser deseable detectar una señal tonal de manera que pueda usarse un modo apropiado para codificarla.

55

Por ejemplo, algunos códecs de voz utilizan un modo de predicción lineal excitada por ruido (NELP) para codificar tramas sordas. Aunque un modo NELP puede ser adecuado para formas de onda que se asemejen al ruido, es posible que tal modo produzca un mal resultado si se utiliza para codificar una señal tonal. Los modos de interpolación de forma de onda (WI), los cuales incluyen el modo de interpolación de forma de onda de prototipo (PWI) y el modo de periodo de tono de prototipo (PPP), son muy adecuados para codificar formas de onda que tengan un fuerte componente periódico. Sin embargo, en comparación con otro modo de codificación a la misma velocidad, un modo NELP o WI puede producir un mal resultado si se utiliza para codificar una señal que tenga dos o más componentes tonales, tal como una que incluye una señal DTMF. Es posible que el uso de tales modos de codificación a bajas velocidades binarias (tales como media velocidad (por ejemplo, 4 kbps), un cuarto de velocidad (por ejemplo, 2 kbps), o inferiores), las cuales pueden ser deseables para aumentar la capacidad del sistema, produzca un rendimiento incluso peor para señales tonales. Puede ser deseable utilizar un modo de codificación que pueda aplicarse de manera más general, tal

65

## ES 2 347 473 T3

como un modo de predicción lineal excitada por código (CELP) o un modo de codificación de voz sinusoidal, para codificar una señal tonal.

También puede ser deseable controlar la velocidad a la que se codifica una señal tonal. Tal control puede ser especialmente deseable en un codificador de voz de velocidad variable que elija una de entre una pluralidad de velocidades para codificar la trama de entrada. Por ejemplo, con el fin de conseguir una reproducción de alta calidad de una señal especial tal como un tono de llamada o un tono DTMF, un códec de voz de velocidad variable puede configurarse para usar la velocidad más alta posible, o una velocidad sustancialmente alta, o un modo de codificación especial para codificar una señal en la que se haya detectado la presencia de al menos uno tono.

Pueden surgir problemas cuando se lleva a cabo un esquema de codificación predictivo lineal (LPC) sobre una señal tonal. Por ejemplo, los fuertes picos espectrales de una señal tonal pueden volver inestable el filtro LPC correspondiente, pueden complicar la conversión de los coeficientes LPC a otra forma para la transmisión (tal como pares espectrales lineales, frecuencias espectrales lineales, o pares espectrales de inmitancia), y/o pueden reducir la eficacia de cuantificación. Por lo tanto, puede ser deseable detectar una señal tonal de manera que el esquema LPC pueda modificarse (por ejemplo, poniendo a cero parámetros del modelo LPC que estén por encima de un orden particular).

La Figura 3 muestra un diagrama de flujo de un procedimiento M100 según una configuración desvelada. La tarea T100 lleva a cabo una operación de codificación iterativa, tal como un análisis LPC, en una porción de tiempo de una señal de audio digitalizada (donde T100- $i$  indica la iteración  $i$ -ésima y  $r$  indica el número de iteraciones). La porción de tiempo, o "trama", se selecciona normalmente para que sea lo bastante corta como para que pueda esperarse que la envolvente espectral de la señal permanezca relativamente estacionaria. Una longitud de trama típica es de 20 milisegundos, lo que corresponde a 160 muestras a una tasa de muestreo típica de 8 kHz, aunque puede utilizarse cualquier longitud de trama o tasa de muestreo que se consideren adecuadas para la aplicación particular. En algunas aplicaciones, las tramas no están solapadas, mientras que en otras aplicaciones se utiliza un esquema de tramas solapadas. En un ejemplo de un esquema de tramas solapadas, cada trama se expande para incluir muestras de la trama anterior y de la trama futura adyacentes. En otro ejemplo, cada trama se expande solamente para incluir muestras de la trama anterior adyacente. En los ejemplos particulares descritos posteriormente se adopta un esquema de tramas no solapadas.

Un esquema de codificación de predicción lineal (LPC) modela una señal que va a codificarse  $s$  como una suma de una señal de excitación  $u$  y una combinación lineal de  $p$  muestras anteriores en la señal, como en la siguiente expresión:

$$s[n] = \sum_{i=1}^p a_i s[n-i] + Gu[n].$$

en la que  $G$  denota un factor de ganancia para la señal de entrada  $s$ , y  $n$  denota una muestra o índice de tiempo. Según un esquema de este tipo, la señal de entrada  $s$  puede modelarse como una señal origen de excitación  $u$  que activa un filtro todo polos (o autorregresivo) de orden  $p$  que presenta la siguiente forma:

$$H(z) = \frac{1}{1 - \sum_{i=1}^p a_i z^{-i}} \quad (1)$$

Para cada porción de tiempo (por ejemplo, trama) de la señal de entrada, la tarea T100 extrae un conjunto de parámetros de modelo que estiman la envolvente espectral a largo plazo de la señal. Normalmente, tal extracción se lleva a cabo a una velocidad de 50 tramas por segundo. La información que caracteriza estos parámetros se transfiere en alguna forma a un descodificador, posiblemente con otros datos tales como información que caracteriza a la señal de excitación  $u$ , donde se utiliza para volver a crear la señal de entrada  $s$ .

El orden  $p$  del modelo LPC puede ser cualquier valor considerado adecuado para la aplicación particular, tal como 4, 6, 8, 10, 12, 16, 20 ó 24. En algunas configuraciones, la tarea T100 está configurada para extraer los parámetros de modelo tal como un conjunto de  $p$  coeficientes de filtro  $a_i$ . En el descodificador, estos coeficientes pueden utilizarse para implementar un filtro de síntesis según una realización de forma directa como la mostrada en la Figura 4A. Como alternativa, la tarea T100 puede configurarse para extraer los parámetros de modelo como un conjunto de  $p$  coeficientes de reflexión  $k_i$ , los cuales pueden utilizarse en el descodificador para implementar un filtro de síntesis según una realización reticular como la mostrada en la Figura 4B. La realización de forma directa es normalmente

más sencilla y tiene un coste computacional más bajo, pero los coeficientes de filtro LPC son menos robustos para los errores de redondeo y de cuantificación que los coeficientes de reflexión, de modo que una realización reticular puede preferirse en un sistema que utilice un cálculo de punto fijo o que, de lo contrario, tenga una precisión limitada. (Debe observarse que en algunas descripciones de la técnica, los signos de los parámetros de modelo están invertidos en la expresión (1) anterior y en las implementaciones mostradas en las Figuras 4A y 4B).

Un codificador está configurado normalmente para transmitir los parámetros de modelo a través de un canal de transmisión en forma cuantificada. Los coeficientes de filtro LPC no están acotados y pueden tener un gran intervalo dinámico, y es habitual convertir estos coeficientes a otra forma antes de la cuantificación, tal como pares espectrales lineales (LSP), frecuencias espectrales lineales (LSF) o pares espectrales de inmitancia (ISP). Otras operaciones, tales como ponderación perceptual, también pueden llevarse a cabo sobre los parámetros de modelo antes de la conversión y/o la cuantificación.

También puede ser deseable que el codificador transmita información relacionada con la señal de excitación  $u$ . Algunos codificadores detectan y transmiten la frecuencia o periodo fundamentales de una señal de voz sonora, de manera que el descodificador utiliza un tren de impulsos en esa frecuencia como una excitación para la señal de voz sonora y una excitación de ruido aleatorio para señales de voz sordas. Otros codificadores o modos de codificación utilizan los coeficientes de filtro para extraer la señal de excitación  $u$  en el codificador y codificar la excitación utilizando uno o más libros de códigos. Por ejemplo, un modo de codificación CELP utiliza normalmente un libro de códigos fijo y un libro de códigos adaptativo para modelar la señal de excitación, de manera que la señal de excitación se codifica comúnmente como un índice para el libro de códigos fijo y como un índice para el libro de códigos adaptativo. Puede ser deseable utilizar un modo de codificación CELP para transmitir una señal tonal.

La tarea T100 puede configurarse según cualquiera de las diversas operaciones de codificación iterativas conocidas para calcular parámetros de modelo LPC tales como coeficientes de filtro y/o de reflexión. Tales operaciones de codificación están configuradas normalmente para resolver la expresión (1) de manera iterativa calculando un conjunto de coeficientes que minimicen un error cuadrático medio. Una operación de este tipo puede clasificarse generalmente como un procedimiento de autocorrelación o como un procedimiento de covarianza.

Un procedimiento de autocorrelación calcula el conjunto de coeficientes de filtro y/o de coeficientes de reflexión empezando a partir de valores de la función de autocorrelación de la señal de entrada. Una operación de codificación de este tipo incluye normalmente una tarea de inicialización en la que una función de división en ventanas  $w[n]$  se aplica a la porción de tiempo (por ejemplo, la trama) para poner a cero la señal fuera de la porción. Puede ser deseable utilizar una función de división en ventanas de sección decreciente que tenga bajas ponderaciones de muestras en cada extremo de la ventana, lo que puede ayudar a reducir el efecto de los componentes fuera de la ventana. Por ejemplo, puede ser deseable utilizar una ventana de coseno alzado, tal como la siguiente función de ventana de Hamming:

$$w[n] = \begin{cases} 0,54 & 0,46 \cos \frac{2\pi n}{N-1}, & 0 \leq n \leq N-1 \\ 0, & \text{en los demás casos} \end{cases}$$

en la que  $N$  es el número de muestras en la porción de tiempo.

Otras ventanas de sección decreciente que pueden utilizarse incluyen las ventanas de Hanning, de Blackman, de Kaiser y de Bartlett. La parte dividida en ventanas  $s_w[n]$  puede calcularse según una expresión como la siguiente:

$$s_w[n] = s[n]w[n]; \quad 0 \leq n \leq N-1.$$

La función de división en ventanas no necesita ser simétrica, de manera que una mitad de la ventana puede ponderarse de diferente manera que la otra mitad. También puede usarse una ventana híbrida, tal como una ventana de coseno de Hamming o una ventana que tenga dos mitades de ventanas diferentes (por ejemplo, dos ventanas de Hamming de diferentes tamaños).

Valores de la función de autocorrelación, de la porción de tiempo pueden calcularse según una expresión como la siguiente:

$$R(m) = \sum_{i=0}^{N-1-m} s_w[i]s_w[i+m], \quad 0 \leq m \leq (p-1).$$

## ES 2 347 473 T3

También puede ser deseable llevar a cabo una o más operaciones de preprocesamiento en los valores de autocorrelación antes de calcular las iteraciones. Por ejemplo, los valores de autocorrelación  $R(m)$  pueden suavizarse espectralmente llevando a cabo una operación como la siguiente:

5

10

$$R_u(m) = \begin{cases} 1,00003 R(m), & m = 0; \\ e^{\left[ \frac{1}{2} \left( \frac{40,2m}{8000} \right)^2 \right]} R(m), & 1 \leq m \leq (p-1). \end{cases}$$

15

El preprocesamiento de los valores de autocorrelación también puede incluir normalizar los valores (por ejemplo, con respecto al valor  $R(0)$ , el cual indica la energía total de la porción de tiempo).

20

Un procedimiento de autocorrelación para calcular parámetros de modelo LPC implica llevar a cabo un proceso iterativo para resolver una ecuación que incluya una matriz de Toeplitz. En algunas implementaciones de un procedimiento de autocorrelación, la tarea T100 está configurada para llevar a cabo una serie de iteraciones según cualquiera de los algoritmos recursivos ampliamente conocidos de Levinson y/o de Durbin para resolver tales ecuaciones. Tal y como se muestra en el siguiente listado de pseudocódigo, un algoritmo de este tipo genera los coeficientes de filtro  $a_i$  como los valores para  $1 \leq i \leq p$ , utilizando los coeficientes de reflexión  $k_i$  como intermediarios:

25

30

35

$$\begin{aligned} & E_0 = R(0); \\ & \text{para } (i = 1; i \leq p; i++) \{ \\ & \quad k_i = \left[ R(i) - \sum_{j=1}^{i-1} a_j^{(i-1)} R(i-j) \right] / E_{i-1}; \\ & \quad \text{para } (j = 1; j < i; j++) \quad a_j^{(i)} = a_j^{(i-1)} - k_i a_j^{(i-1)}; \\ & \quad E_i = (1 - k_i^2) E_{i-1}; \\ & \} \end{aligned} \tag{2}$$

40

en el que los valores de autocorrelación de entrada pueden preprocesarse tal y como se ha descrito anteriormente.

45

El término  $E_i$  indica la energía del error (o residuo) que permanece después de la iteración  $i$ . A medida que la serie de iteraciones se ejecuta, la energía residual se reduce progresivamente de manera que  $E_i \leq E_{i-1}$ . La Figura 5 muestra un diagrama de flujo de una implementación M110 del procedimiento M100 que incluye una implementación T110 de la tarea T100 configurada para realizar cálculos de  $k_i$ ,  $a_i$  y  $E_i$  según un algoritmo como el descrito anteriormente, donde T110-0 indica una o más tareas de inicialización y/o de preprocesamiento como las descritas en este documento, tales como la división en ventanas de la trama, el cálculo de los valores de autocorrelación, el suavizado espectral de los valores de autocorrelación, etc.

50

En otras implementaciones de un procedimiento de autocorrelación, la tarea T100 está configurada para llevar a cabo una serie de iteraciones para calcular los coeficientes de reflexión  $k_i$  (denominados también como coeficientes de correlación parcial (PARCOR), coeficientes PARCOR negativos, o parámetros de Schur-Szego) en lugar de los coeficientes de filtro  $a_i$ . Un algoritmo que puede utilizarse en la tarea T100 para obtener los coeficientes de reflexión es el algoritmo de Leroux-Gueguen, el cual utiliza estimaciones de respuesta de impulso  $e$  como intermediarias y se expresa en el siguiente listado de pseudocódigo:

60

65

$$\begin{aligned} & \text{para } (i = -(p-1); i \leq p; i++) \quad e_0(i) = R(i); \\ & \text{para } (m = 1; m \leq p; m++) \{ \\ & \quad k_m = -e_{m-1}(m) / e_{m-1}(0); \\ & \quad \text{para } (i = -(p-1) + m; i \leq p; i++) \\ & \quad \quad e_m(i) = e_{m-1}(i) + k_m e_{m-1}(m-i); \\ & \} \end{aligned} \tag{3}$$

## ES 2 347 473 T3

El algoritmo de Leroux-Gueguen se implementa normalmente utilizando dos matrices EP, EN en lugar de las matrices  $e$ . La Figura 6 muestra un listado de pseudocódigo para una implementación de este tipo que incluye calcular un término  $E(h)$  de error (o de energía residual) en cada iteración. Otros procedimientos iterativos ampliamente conocidos que pueden utilizarse para obtener los coeficientes de reflexión  $k_i$  a partir de los valores de autocorrelación incluyen los algoritmos recursivos de Schur, los cuales pueden configurarse para un cálculo en paralelo eficaz.

Tal y como se ha mencionado anteriormente, los coeficientes de reflexión pueden utilizarse para implementar una realización reticular del filtro de síntesis. Como alternativa, los coeficientes de filtro LPC pueden obtenerse a partir de los coeficientes de reflexión a través de una recursión como la mostrada en el siguiente listado de pseudocódigo:

```

para (i = 1; i ≤ p; i++) {
    ai(i) = ki;
    para (j = 1; j ≤ i; j++) aj(i) = aj(i-1) + kiai-j(i-1);
}

```

Los procedimientos de covarianza son otra clase de operaciones de codificación que pueden utilizarse en la tarea T100 para calcular de manera iterativa un conjunto de coeficientes para minimizar un error cuadrático medio. Un procedimiento de covarianza comienza a partir de valores de la función de covarianza de la señal de entrada y aplica normalmente una ventana de análisis a la señal de error en lugar de a la señal de voz de entrada. En este caso, la ecuación matricial que va a resolverse incluye una matriz definida positiva y simétrica en lugar de una matriz de Toeplitz, de manera que los algoritmos de Levinson-Durbin y de Leroux-Gueguen no son adecuados, pero puede utilizarse la descomposición de Cholesky para obtener los coeficientes de filtro  $a_i$  de una manera eficaz. Aunque un procedimiento de covarianza puede conservar una alta resolución espectral, no garantiza la estabilidad del filtro resultante. El uso de los procedimientos de covarianza es menos común que el uso de los procedimientos de autocorrelación.

Para algunas o todas las iteraciones de la operación de codificación, la tarea T200 calcula un valor correspondiente de una medida relacionada con una ganancia de la operación de codificación. Puede ser deseable calcular la medida de ganancia como una relación entre una medida de la energía de señal inicial (por ejemplo, la energía de la trama dividida en ventanas) y una medida de la energía del residuo actual. En un ejemplo de este tipo, la medida de ganancia  $G_i$  para la iteración  $i$  se calcula según la siguiente expresión:

$$G_i = \frac{E_0}{E_i}.$$

En este caso, el factor  $G_i$  representa hasta ahora la ganancia de predicción LPC de la operación de codificación. La ganancia de predicción también puede calcularse a partir de los coeficientes de reflexión  $k_i$  según la siguiente expresión:

$$G_i = \frac{1}{\prod_{j=1}^i (1 - k_j^2)}.$$

En otro ejemplo de este tipo, puede ser deseable calcular la medida de ganancia  $G_i$  para representar el error de predicción LPC actual, como en las siguientes expresiones:

$$G_i = \frac{E_i}{E_0} \quad \text{ó} \quad G_i = \prod_{j=1}^i (1 - k_j^2).$$

La medida de ganancia  $G_i$  también puede calcularse según otras expresiones que, por ejemplo, también incluyan el producto  $\prod_{j=1}^i (1 - k_j^2)$ , o una relación entre  $E_0$  y  $E_i$ , como un factor o un término. La medida de ganancia  $G_i$  puede expresarse en una escala lineal o en otro dominio, tal como en una escala logarítmica (por ejemplo,  $\log E_0/E_i$  ó  $\log E_i/E_0$ ). Implementaciones adicionales de la tarea T200 calculan la medida de ganancia en función de un cambio en la

## ES 2 347 473 T3

energía residual (por ejemplo,  $G_i = \Delta E_i = E_i - E_{i-1}$ ).

Normalmente, la medida de ganancia  $G_i$  se calcula en cada iteración (por ejemplo, las tareas T200-i mostradas en las Figuras 3 y 5), aunque también es posible implementar la tarea T200 de manera que la medida de ganancia  $G_i$  se calcule solamente cada dos iteraciones, cada tres iteraciones, etc. El siguiente listado de pseudocódigo muestra un ejemplo de una modificación del listado de pseudocódigo (2) anterior que puede utilizarse para llevar a cabo implementaciones en ambas tareas T100 y T200:

```

10       $E_0 = R(0);$ 
      para ( $i = 1; i \leq p; i ++$ ) {
15           $k_i = \left[ R(i) - \sum_{j=1}^{i-1} a_j^{(i-1)} R(i-j) \right] / E_{i-1};$ 
      para ( $j = 1; j < i; j ++$ )  $a_j^{(i)} = a_j^{(i-1)} - k_i a_{i-j}^{(i-1)};$           (4)
       $E_i = (1 - k_i^2) E_{i-1};$ 
       $G_i = E_0 / E_i;$ 
      }

```

La Figura 7 muestra un ejemplo de una modificación del listado de pseudocódigo de la Figura 6 que puede utilizarse para llevar a cabo implementaciones de ambas tareas T100 y T200.

Cuando uno o más tonos están presentes en la señal que está analizándose, la energía residual puede reducirse rápidamente entre dos de las iteraciones. La tarea T300 determina y registra una indicación de la primera iteración en la que se produce un cambio en un estado de una relación entre el valor de la medida de ganancia y un valor de umbral  $T$ . Para un caso en el que la medida de ganancia se calcula como  $E_0/E_i$ , por ejemplo, la tarea T300 puede configurarse para registrar una indicación de la primera iteración en la que un estado de la relación " $G_i > T$ " (o " $G_i \geq T$ ") cambia de falso a verdadero o, de manera equivalente, en la que un estado de la relación " $G_i \leq T$ " (o " $G_i < T$ ") cambia de verdadero a falso. Para un caso en el que la medida de ganancia se calcula como  $E_i/E_0$ , por ejemplo, la tarea T300 puede configurarse para registrar una indicación de la primera iteración en la que un estado de la relación " $G_i > T$ " (o " $G_i \geq T$ ") cambia de verdadero a falso o, de manera equivalente, en la que un estado de la relación " $G_i \leq T$ " (o " $G_i < T$ ") cambia de falso a verdadero.

La indicación almacenada de la primera iteración en la que se produce el cambio de estado pertinente también se denomina como una "orden de parada", y la operación que determina si se ha producido el cambio de estado pertinente también se denomina como "actualizar la orden de parada". Una orden de parada puede almacenar el valor de índice  $i$  de la iteración objetivo o puede almacenar alguna otra indicación del valor de índice  $i$ . En este documento se supone que la tarea T300 está configurada para inicializar cada orden de parada a un valor por defecto de cero, aunque también se contemplan expresamente configuraciones dadas a conocer en este documento en las que la tarea T300 está configurada para inicializar cada orden de parada a algún otro valor por defecto (por ejemplo,  $p$ ) o en las que el estado de una bandera de actualización respectiva se utiliza para indicar si la orden de parada contiene un valor válido. En este último tipo de configuración de la tarea T300, por ejemplo, si el estado de una bandera de actualización ha cambiado para impedir una actualización adicional, entonces se supone que la orden de parada correspondiente contiene un valor válido.

La tarea T300 puede configurarse para mantener más de una orden de parada (por ejemplo, dos o más). Es decir, la tarea T300 puede configurarse para determinar, para cada uno de una pluralidad de  $q$  umbrales  $T_j$  diferentes (donde  $1 \leq j \leq q$ ), la primera iteración en la que se produce un cambio en un estado de una relación entre el valor de la medida de ganancia y el valor de umbral  $T_j$ , y para almacenar una indicación de la iteración (por ejemplo, en una posición de memoria correspondiente). Para una configuración en la que  $G_i$  aumenta monótonicamente con  $i$  (por ejemplo,  $G_i = E_0/E_i$ ), puede ser deseable disponer los umbrales en una progresión de manera que  $T_j < T_{j+1}$ . Para una configuración en la que  $G_i$  disminuye monótonicamente con  $i$  (por ejemplo,  $G_i = E_i/E_0$ ), puede ser deseable disponer los umbrales en una progresión de manera que  $T_j > T_{j+1}$ . En un ejemplo particular, la tarea T300 está configurada para mantener tres órdenes de parada. Un ejemplo de un conjunto de umbrales  $T_j$  que puede utilizarse en un caso de este tipo es  $T_1 = 6,8$  dB,  $T_2 = 8,1$  dB y  $T_3 = 8,6$  dB (por ejemplo, para  $G_i = E_0/E_i$ ). Otro ejemplo de un conjunto de umbrales  $T_j$  que puede utilizarse en un caso de este tipo es  $T_1 = 15$  dB,  $T_2 = 20$  dB y  $T_3 = 30$  dB (por ejemplo, para  $G_i = E_i/E_0$ ).

La tarea T300 puede configurarse para actualizar la(s) orden(es) de parada cada vez que la tarea T200 calcule un valor para la medida de ganancia  $G_i$  (por ejemplo, en cada iteración de la tarea T 100), de manera que las órdenes de parada son actuales cuando la serie de iteraciones se completa. Como alternativa, la tarea T300 puede configurarse para actualizar la(s) orden(es) de parada después de que la serie de iteraciones se haya completado, por ejemplo, procesando de manera iterativa los valores de medida de ganancia  $G_i$  de las respectivas iteraciones que se hayan registrado por la

## ES 2 347 473 T3

tarea T200.

La Figura 8 muestra un ejemplo de una estructura lógica que puede utilizarse por la tarea T300 para actualizar algún número  $q$  de órdenes de parada en serie y/o en paralelo. En este ejemplo, cada módulo  $j$  de la estructura determina si la medida de ganancia es mayor que (como alternativa, no inferior a) un valor de umbral  $T_j$  correspondiente para la orden de parada  $S_j$ . Si este resultado es verdadero, y la bandera de actualización para la orden de parada también tiene valor verdadero, entonces la orden de parada se actualiza para indicar el índice de la iteración, y el estado de la bandera de actualización se modifica para impedir una actualización adicional de la orden de parada.

Las Figuras 9A y 9B muestran ejemplos de diagramas de flujo que pueden reproducirse exactamente en implementaciones alternativas de la tarea T300 para actualizar cada una de un conjunto de órdenes de parada en serie y/o en paralelo. En estos ejemplos, el estado de la relación se evalúa solamente si la bandera de actualización respectiva todavía tiene valor verdadero. En el ejemplo de la Figura 9B, la orden de parada se incrementa en cada iteración hasta que la medida de ganancia  $G_i$  alcance el umbral  $T_j$  (como alternativa, lo supere), punto en el que la tarea T300 inhabilita un incremento adicional de la orden de parada modificando el estado de la bandera de actualización.

El siguiente listado de pseudocódigo muestra un ejemplo de una modificación del listado de pseudocódigo (4) anterior que puede utilizarse para llevar a cabo implementaciones de todas las tareas T100, T200 y T300:

```

20       $E_0 = R(0);$ 
      para ( $j = 1; j \leq q; j++$ ) {  $S\_actualizar(j) = 1; S_j = 0; \}$ 
25      para ( $i = 1; i \leq p; i++$ ) {
           $k_i = \left[ R(i) - \sum_{j=1}^{i-1} a_j^{(i-1)} R(i-j) \right] / E_{i-1};$ 
30      para ( $j = 1; j < i; j++$ )  $a_j^{(i)} = a_j^{(i-1)} - k_i a_{i-j}^{(i-1)};$ 
           $E_i = (1 - k_i^2) E_{i-1};$ 
           $G_i = E_0 / E_i;$ 
35
          para ( $j = 1; j \leq q; j++$ ) {
              si ( $S\_actualizar(j)$ ) {
40                   $S_j++;$ 
                  si ( $G_i > T_j$ )  $S\_actualizar(j) = 0;$ 
45              }
          }
50      }

```

(5)

En este ejemplo, el listado (5) incluye una implementación de la tarea T300 mostrada en la Figura 9B. La Figura 10 muestra un ejemplo de una modificación del listado de pseudocódigo de la Figura 7 que puede utilizarse para llevar a cabo implementaciones de todas las tareas T100, T200 y T300.

En algunas configuraciones, puede ser deseable que la tarea T300 actualice una orden de parada solamente después de que se hayan fijado los valores de las órdenes de parada anteriores a la misma. Por ejemplo, puede ser deseable que diferentes órdenes de parada tengan diferentes valores (por ejemplo, excepto para órdenes de parada que tengan el valor por defecto). La Figura 11 muestra un ejemplo de este tipo de un módulo que puede reproducirse exactamente en una implementación alternativa de la tarea T300 en la que la actualización de una orden de parada se suspende hasta que se haya fijado el valor de la orden de parada anterior.

La tarea T400 compara una o más de las órdenes de parada con un valor de umbral. La Figura 12 muestra un ejemplo de un procedimiento de prueba para una configuración de la tarea T400 que prueba las órdenes de parada secuencialmente en orden creciente. En este ejemplo, la tarea T400 compara cada orden de parada  $S_j$  con un par correspondiente de umbrales superior e inferior (excepto para la última orden de parada  $S_q$ , la cual se prueba solamente con un umbral inferior en este ejemplo particular) hasta que se tome una decisión en cuanto a la tonalidad de la porción de tiempo. La Figura 13 muestra un diagrama de flujo de una implementación de la tarea T400 que lleva a cabo tal

## ES 2 347 473 T3

procedimiento de prueba en serie para un caso en el que  $q$  es igual a tres. En otro ejemplo, una o más de las relaciones “<” en una tarea de este tipo se sustituye por la relación “≤”.

5 Tal y como se muestra en la Figura 12, un posible primer resultado de la prueba es que la orden de parada tenga un valor inferior a (como alternativa, no mayor que) el umbral inferior correspondiente. Un resultado de este tipo puede indicar que se obtuvo más ganancia de predicción a índices de iteración bajos de lo que se esperaría para una señal de voz. En este ejemplo, la tarea T400 está configurada para clasificar la porción de tiempo como una señal tonal.

10 Un posible segundo resultado de la prueba es que la orden de parada tenga un valor entre el umbral inferior y el umbral superior, lo que puede indicar que la distribución de energía espectral es típica de una señal de voz. En este ejemplo, la tarea T400 está configurada para clasificar la porción de tiempo como no tonal.

15 Un posible tercer resultado de la prueba es que la orden de parada tenga un valor mayor que (como alternativa, no inferior a) el umbral superior correspondiente. Un resultado de este tipo puede indicar que se obtuvo menos ganancia de predicción a índices de iteración bajos de lo que se esperaría para una señal de voz. En este ejemplo, la tarea T400 está configurada para continuar el procedimiento de prueba hasta la siguiente orden de parada en un caso de este tipo.

20 La Figura 14 muestra gráficos de la medida de ganancia  $G_i$  frente al índice de iteración  $i$  para cuatro ejemplos diferentes A a D de porciones de tiempo. En estos gráficos, el eje vertical indica la magnitud de la medida de ganancia  $G_i$ , el eje horizontal indica el índice de iteración  $i$ , y  $p$  tiene el valor 12. Tal y como se indica en los gráficos, en estos ejemplos los umbrales de medida de ganancia  $T_1$ ,  $T_2$  y  $T_3$  tienen asignados los valores 8, 19 y 34, respectivamente, y los umbrales de orden de parada  $T_{L1}$ ,  $T_{U1}$ ,  $T_{L2}$ ,  $T_{U2}$  y  $T_{L3}$  tienen asignados los valores 3, 4, 7, 8 y 11, respectivamente. (En general, no es necesario que  $T_{Li}$  sea adyacente a  $T_{Ui}$ , o que  $T_{Ui}$  sea inferior a  $T_{L(i+1)}$ , para cualquier índice  $i$ ).

25 Usando estos valores de umbral, todas las porciones de tiempo mostradas en los gráficos A a D se clasifican como tonales por la implementación particular de la tarea T400 mostrada en la Figura 13. La porción de tiempo del gráfico A se clasifica como tonal ya que  $S_1$  es inferior a  $T_{L1}$ . Las porciones de tiempo de los gráficos B y C se clasifican como tonales ya que para ambas porciones  $S_1$  es mayor que  $T_{U1}$  y  $S_2$  es inferior a  $T_{L2}$ . Debe observarse también que el gráfico C muestra un ejemplo en el que dos órdenes de parada diferentes tienen el mismo valor. La porción de tiempo del gráfico D se clasifica como tonal ya que  $S_1$  y  $S_2$  son mayores que  $S_{U1}$  y  $S_{U2}$ , respectivamente, y  $S_3$  es inferior a  $T_{L3}$ .

La Figura 15 muestra un ejemplo de una estructura lógica para la tarea T400 en la que las pruebas mostradas en la Figura 13 pueden llevarse a cabo en paralelo.

35 Puede apreciarse que en la implementación de la tarea T400 mostrada en la Figura 13, la secuencia de prueba termina una vez que se haya tomado una decisión de tonalidad, incluso si solo se ha examinado la primera de las órdenes de parada. El alcance de las implementaciones del procedimiento M100 también incluye configuraciones de la tarea T400 en las que la secuencia de pruebas continúa. En una configuración de este tipo, una porción de tiempo se clasifica como tonal si cualquiera de las órdenes de parada presenta un valor inferior a (como alternativa, no mayor que) el umbral inferior correspondiente. En otra configuración de este tipo, una porción de tiempo se clasifica como tonal si la mayoría de las órdenes de parada presentan valores inferiores a (como alternativa, no mayores que) los umbrales inferiores correspondientes.

45 La Figura 21 muestra un diagrama de flujo de otra implementación de la tarea T400 que prueba las órdenes de parada secuencialmente en orden decreciente. En este ejemplo se utilizan dos órdenes de parada (es decir,  $q = 2$ ). El intervalo de valores particulares que puede utilizarse en una implementación de este tipo incluye el conjunto  $T_1 = 15$  dB,  $T_2 = 30$  dB,  $T_{L1} = 4$ ,  $T_{L2} = 4$  y  $T_{U2} = 6$ . En otro ejemplo, una o más de las relaciones “<” en una tarea de este tipo se sustituye por la relación “≤”.

50 La Figura 22 muestra un diagrama de flujo de una implementación adicional de la tarea T400 que prueba las órdenes de parada secuencialmente en orden decreciente, comparándose cada orden de parada  $S_q$  con un umbral  $T_{S_q}$  correspondiente. En este ejemplo se utilizan dos órdenes de parada (es decir,  $q = 2$ ). El intervalo de valores particulares que puede utilizarse en una implementación de este tipo incluye el conjunto  $T_1 = 15$  dB,  $T_2 = 30$  dB,  $T_{S1} = 4$  y  $T_{S2} = 4$ . En otro ejemplo, una o más de las relaciones “<” en una tarea de este tipo se sustituye por la relación “≤”.

55 Esta implementación también ilustra un caso en el que el resultado de la tarea T400 puede ser contingente en una o más condiciones. Ejemplos de tales condiciones incluyen una o más cualidades de la porción de tiempo, tal como el estado de una relación entre la inclinación espectral (es decir, el primer coeficiente de reflexión) de la porción de tiempo y un valor de umbral. Ejemplos de tales condiciones incluyen además uno o más historiales de la señal, tal como el resultado de la tarea T400 para una o más de las porciones de tiempo anteriores.

65 Tal y como se muestra en las Figuras 3 y 5, la tarea T400 puede configurarse para ejecutarse después de se haya completado la serie de iteraciones. Sin embargo, el alcance contemplado de las implementaciones del procedimiento M100 incluye además implementaciones que están configuradas para llevar a cabo la tarea T400 si se actualiza una orden de parada e implementaciones que están configuradas para llevar a cabo la tarea T400 en cada iteración.

El alcance de las implementaciones del procedimiento M100 incluye además implementaciones que están configuradas para llevar a cabo una o más acciones en respuesta al resultado de la tarea T400. Por ejemplo, puede ser deseable

## ES 2 347 473 T3

truncar o terminar de otra manera una LP u otra operación de codificación de voz cuando la trama que está codificándose es tonal. Tal y como se ha indicado anteriormente, los altos picos espectrales de una señal tonal pueden provocar inestabilidad en un filtro LPC, y la conversión de los coeficientes LPC a otra forma para la transmisión (tal como pares espectrales lineales, frecuencias espectrales lineales o pares espectrales de inmitancia) también puede verse afectada si la señal tiene picos.

Algunas implementaciones del procedimiento M100 están configuradas para truncar el análisis LPC según el índice de iteración  $i$  indicado por la orden de parada en la que se alcanzó la clasificación de tonalidad en la tarea T400. Por ejemplo, un procedimiento de este tipo puede configurarse para reducir las magnitudes de los coeficientes LPC (por ejemplo, los coeficientes de filtro) para el índice  $i$  y superiores, por ejemplo, asignando valores de cero a esos coeficientes. Tal truncado puede llevarse a cabo después de que se haya completado la serie de iteraciones. Como alternativa, para una implementación de este tipo en la que la tarea T400 se lleva a cabo en cada iteración o cuando se actualiza una orden de parada, tal truncado puede incluir finalizar la serie de iteraciones de la tarea T100 antes de que se llegue a la iteración  $p$ -ésima.

Tal y como se ha indicado anteriormente, otras implementaciones del procedimiento M100 pueden estar configuradas para seleccionar un modo y/o una velocidad de codificación adecuados en función del resultado de la tarea T400. Un modo de codificación de propósito general, tal como una predicción lineal excitada por código (CELP) o un modo de codificación sinusoidal, puede dejar pasar cualquier forma de onda parecida. Por lo tanto, una manera de transferir el tono de manera satisfactoria al descodificador es obligar al codificador a que utilice un modo de codificación de este tipo (por ejemplo, CELP a velocidad completa). Un codificador de voz con módem aplica normalmente varios criterios para determinar cómo va a codificarse cada trama (tales como límites de velocidad), de manera que forzar un modo de codificación particular puede requerir invalidar muchas otras decisiones.

El alcance de las implementaciones del procedimiento M100 incluye además implementaciones que tienen tareas que están configuradas para identificar la frecuencia o el tipo del tono o tonos. En ese caso, puede ser deseable utilizar un modo de codificación especial para enviar esa información en lugar de codificar la porción de tiempo. Un procedimiento de este tipo puede iniciar la ejecución de una tarea de identificación de frecuencia (por ejemplo, en lugar de continuar un procedimiento de codificación de voz para esa trama) en función del resultado de la tarea T400. Por ejemplo, puede utilizarse una disposición de filtros de ranura para identificar las frecuencias del o de los componentes de frecuencia más fuertes de la porción de tiempo. Un filtro de este tipo puede configurarse para dividir el espectro de frecuencia (o alguna parte del mismo) en contenedores (*bins*) que tengan un ancho de, por ejemplo, 100 Hz o 200 Hz. La tarea de identificación de frecuencia puede examinar todo el espectro o la porción de tiempo o, como alternativa, solo contenedores o regiones de frecuencia seleccionados (tales como regiones que incluyan las frecuencias de tonos de señalización comunes tales como señales DTMF).

En caso de que se identifiquen los dos tonos de una señal DTMF, puede ser deseable utilizar un modo de codificación especial para transmitir un dígito correspondiente a la señal DTMF identificada en lugar de los propios tonos o una identificación de las frecuencias reales. La tarea de identificación de frecuencia también puede configurarse para detectar la duración de uno o más tonos, información que puede transmitirse al descodificador. Un codificador de voz que lleve a cabo una implementación de este tipo del procedimiento M100 también puede configurarse para transmitir información tal como la frecuencia, la amplitud y/o la duración de tono a un descodificador a través de un canal secundario de un esquema de canales de transmisión, tal como un canal de datos o de señalización, en lugar de a través de un canal de tráfico.

El procedimiento M100 puede utilizarse en el contexto de un codificador de voz o puede aplicarse de manera independiente (por ejemplo para proporcionar una detección de tono en un dispositivo diferente a un codificador de voz). La Figura 16A muestra un diagrama de bloques de un aparato A100 según una configuración desvelada que también puede utilizarse en un codificador de voz, tal como un detector de tono, y/o como parte de otro dispositivo o sistema.

El aparato A100 incluye un calculador de coeficientes A110 que está configurado para llevar a cabo una operación de codificación iterativa para calcular una pluralidad de coeficientes (por ejemplo, coeficientes de filtro y/o coeficientes de reflexión) a partir de una porción de tiempo de una señal de audio digitalizada. Por ejemplo, el calculador de coeficientes A110 puede configurarse para llevar a cabo una implementación de la tarea T100 descrita en este documento.

El calculador de coeficientes A110 puede configurarse para llevar a cabo la operación de codificación iterativa según un procedimiento de autocorrelación descrito en este documento. La Figura 16B muestra un diagrama de bloques de una implementación A200 del aparato A100 que también incluye un calculador de autocorrelaciones A105 configurado para calcular valores de autocorrelación de la porción de tiempo. El calculador de autocorrelaciones A105 también puede configurarse para llevar a cabo un suavizado espectral de los valores de autocorrelación tal y como se describe en este documento.

El aparato A100 incluye un calculador de medidas de ganancia A120 configurado para calcular, en cada una de la pluralidad ordenada de iteraciones, un valor de una medida relacionada con una ganancia de la operación de codificación. El valor de la medida de ganancia puede ser una ganancia de predicción o un error de predicción. El valor de la medida de ganancia puede calcularse en función de una relación entre una medida de la energía de la porción de

## ES 2 347 473 T3

tiempo y una medida de la energía residual en la iteración. Por ejemplo, el calculador de medidas de ganancia A120 puede configurarse para llevar a cabo una implementación de la tarea T200 descrita en este documento.

El aparato A100 incluye además una primera unidad de comparación A130 configurada para almacenar una indicación de la iteración, entre la pluralidad ordenada, en la que se produce un cambio en un estado de una primera relación entre el valor calculado y un primer valor de umbral. La indicación de la iteración puede implementarse como una orden de parada, y la primera unidad de comparación A130 puede configurarse para actualizar una o más órdenes de parada. Por ejemplo, la primera unidad de comparación A130 puede configurarse para llevar a cabo una implementación de la tarea T300 descrita en este documento.

El aparato A100 incluye además una segunda unidad de comparación A140 configurada para comparar la indicación almacenada con un segundo valor de umbral. La segunda unidad de comparación A140 puede configurarse para clasificar la porción de tiempo como tonal o como no tonal en función de un resultado de la comparación. Por ejemplo, la segunda unidad de comparación A140 puede configurarse para llevar a cabo una implementación de la tarea T400 descrita en este documento. Una implementación adicional del aparato A100 incluye una implementación de un selector de modo 202 descrito posteriormente que está configurado para seleccionar un modo de codificación y/o una velocidad de codificación en función de la salida de la segunda unidad de comparación A140.

Los diversos elementos de las implementaciones del aparato A100 pueden implementarse como dispositivos electrónicos y/u ópticos que residan, por ejemplo, en el mismo chip o entre dos o más chips de un conjunto de chips, aunque también se contemplan otras disposiciones sin tal limitación. Uno o más elementos de un aparato de este tipo pueden implementarse completamente o en parte como uno o más conjuntos de instrucciones dispuestas para ejecutarse en una o más disposiciones fijas o programables de elementos lógicos (por ejemplo, transistores, puertas) tales como microprocesadores, procesadores embebidos, núcleos IP, procesadores de señales digitales, FPGA (matrices de puertas programables de campo), ASSP (productos estándar de aplicación específica) y ASIC (circuitos integrados de aplicación específica).

Es posible utilizar uno o más elementos de una implementación del aparato A100 para llevar a cabo tareas o ejecutar otros conjuntos de instrucciones que no estén directamente relacionadas con una operación el aparato, tal como una tarea relacionada con otra operación de un dispositivo o sistema en el que esté embebido el aparato. También es posible que uno o más elementos de una implementación del aparato A100 tengan una estructura en común (por ejemplo, un procesador utilizado para ejecutar partes de código correspondientes a diferentes elementos en momentos diferentes, un conjunto de instrucciones ejecutadas para llevar a cabo tareas correspondientes a diferentes elementos en momentos diferentes, o una disposición de dispositivos electrónicos y/u ópticos que lleven a cabo operaciones para diferentes elementos en momentos diferentes). Tal y como se muestra en los listados de pseudocódigo (4) y (5) anteriores y en los listados de pseudocódigo de las Figuras 7 y 10, por ejemplo, uno o más elementos de una implementación del aparato A100 pueden implementarse incluso como diferentes partes del mismo bucle.

Las configuraciones descritas anteriormente pueden utilizarse en uno o más dispositivos (por ejemplo, codificadores de voz) de un sistema de comunicaciones de telefonía inalámbrica configurado para utilizar una interfaz inalámbrica CDMA (acceso múltiple por división de código). Sin embargo, los expertos en la técnica entenderán que los procedimientos y aparatos que incluyen las características descritas en este documento pueden residir en cualquiera de varios sistemas de comunicación utilizando una amplia gama de tecnologías conocidas por los expertos en la técnica. Por ejemplo, un experto en la técnica apreciará que los procedimientos y aparatos descritos anteriormente pueden aplicarse a cualquier sistema de comunicación digital, independientemente del esquema de transmisión físico y/u lógico, e independientemente de si tal sistema es cableado y/o inalámbrico, de conmutación de circuitos y/o de conmutación de paquetes, etc., y el uso de estos procedimientos y/o aparatos con tales sistemas se contempla y se da a conocer de manera expresa.

Tal y como se ilustra en Figura 17, un sistema de telefonía celular incluye generalmente una pluralidad de unidades de abonado móviles 10, una pluralidad de estaciones base 12, controladores de estación base (BSC) 14 y un centro de conmutación móvil (MSC) 16. El MSC 16 está configurado para interactuar con una red telefónica pública conmutada (PSTN) 18 convencional. El MSC 16 está configurado además para interactuar con los BSC 14. Los BSC 14 están acoplados a las estaciones base 12 a través de líneas de retroceso. Las líneas de retroceso pueden configurarse para soportar cualquiera de varias interfaces conocidas incluyendo, por ejemplo, E1/T1, ATM, IP, PPP, Frame Relay, HDSL, ADSL o xDSL. Debe entenderse que puede haber más de dos BSC 14 en el sistema. Cada estación base 12 incluye de manera ventajosa al menos un sector (no mostrado), comprendiendo cada sector una antena omnidireccional o una antena que apunta a una dirección particular radialmente desde la estación base 12. Como alternativa, cada sector puede comprender dos antenas para la recepción de diversidad. Cada estación base 12 puede estar diseñada de manera ventajosa para soportar una pluralidad de asignaciones de frecuencia. En un sistema CDMA, la intersección de un sector y de una asignación de frecuencia puede denominarse como un canal CDMA. Las estaciones base 12 también pueden conocerse como subsistemas de transceptor de estación base (BTS) 12. Como alternativa, "estación base" puede utilizarse en la industria para referirse colectivamente a un BSC 14 y a uno o más BTS 12. Los BTS 12 también pueden denotarse como "emplazamientos de célula" 12. Como alternativa, los sectores individuales de un BTS 12 dado pueden denominarse como emplazamientos de célula. Las unidades de abonado móviles 10 son normalmente teléfonos celulares o PCS 10. Un sistema de este tipo puede configurarse para utilizarse según la norma IS-95 u otra norma CDMA. Un sistema de este tipo también puede configurarse para transportar tráfico de voz a través de uno o más protocolos de conmutación de paquetes, tales como VoIP.

## ES 2 347 473 T3

Durante el funcionamiento típico del sistema de telefonía celular, las estaciones base 12 reciben conjuntos de señales de enlace inverso desde conjuntos de unidades móviles 10. Las unidades móviles 10 llevan a cabo llamadas telefónicas u otras comunicaciones. Cada señal de enlace inverso recibida por una estación base 12 dada se procesa en esa estación base 12. Los datos resultantes se reenvían a los BSC 14. Los BSC 14 proporcionan asignación de recursos de llamada y funcionalidad de gestión de movilidad incluyendo el control de los traspasos continuos entre las estaciones base 12. Los BSC 14 encaminan además los datos recibidos al MSC 16, el cual proporciona servicios de encaminamiento adicionales para interactuar con la PSTN 18. De manera similar, la PSTN 18 interactúa con el MSC 16, y el MSC 16 interactúa con los BSC 14, que a su vez controlan las estaciones base 12 para transmitir conjuntos de señales de enlace directo a conjuntos de unidades móviles 10.

La Figura 18 muestra un diagrama de un sistema que incluye dos codificadores 100, 106 que pueden configurarse para llevar a cabo una implementación de la tarea T400 descrita en este documento y/o que pueden configurarse para incluir una implementación del aparato A100 descrito en este documento. El primer codificador 100 recibe muestras de voz digitalizadas  $s(n)$  y codifica las muestras  $s(n)$  para su transmisión a través de un medio de transmisión y/o canal de comunicaciones 102 hasta un primer descodificador 104. El descodificador 104 descodifica las muestras de voz codificadas y sintetiza una señal de voz de salida  $sSYNTH(n)$ . Para la transmisión en el sentido contrario, un segundo codificador 106 codifica muestras de voz digitalizadas  $s(n)$ , las cuales se transmiten a través de un medio de transmisión y/o canal de comunicaciones 108. Un segundo descodificador 110 recibe y descodifica las muestras de voz codificadas, generando una señal de voz de salida sintetizada  $sSYNTH(n)$ . El codificador 100 y el descodificador 110 pueden implementarse conjuntamente en un transceptor tal como un teléfono celular. Asimismo, el codificador 106 y el descodificador 104 pueden implementarse conjuntamente en un transceptor tal como un teléfono celular.

Las muestras de voz  $s(n)$  representan señales de voz que se han digitalizado y cuantificado según cualquiera de varios procedimientos conocidos en la técnica incluyendo, por ejemplo, la modulación por impulsos codificados (PCM), y la ley A o ley  $\mu$  de compresión-expansión. Tal y como se conoce en la técnica, las muestras de voz  $s(n)$  se organizan en tramas de datos de entrada en las que cada trama comprende un número predeterminado de muestras de voz digitalizadas  $s(n)$ . En una configuración a modo de ejemplo se utiliza una tasa de muestreo de 8 kHz, comprendiendo cada trama de 20 milisegundos 160 muestras. En las configuraciones descritas posteriormente, la velocidad de transmisión de datos puede variar de manera ventajosa por cada trama entre velocidad completa, media velocidad, un cuarto de velocidad y un octavo de velocidad (correspondiéndose en el ejemplo a 13,2, 6,2, 2,6 y 1 kbps, respectivamente). Variar la velocidad de transmisión de datos es potencialmente ventajoso ya que las velocidades binarias más bajas pueden emplearse de manera selectiva para tramas que contengan relativamente menos información de voz. Tal y como entienden los expertos en la técnica, pueden utilizarse otras tasas de muestreo, tamaños de trama y velocidades de transmisión de datos.

El primer codificador 100 y el segundo descodificador 110 comprenden conjuntamente un primer codificador de voz, o códec de voz. El codificador de voz puede configurarse para utilizarse en cualquier tipo de dispositivo de comunicaciones para transmitir señales de voz a través de un canal cableado y/o inalámbrico, incluyendo, por ejemplo, las unidades de abonado, los BTS, o los BSC descritos anteriormente con referencia a la Figura 17. De manera similar, el segundo codificador 106 y el primer descodificador 104 comprenden conjuntamente un segundo codificador de voz. Los expertos en la técnica entenderán que los codificadores de voz pueden implementarse con un procesador de señales digitales (DSP), un circuito integrado de aplicación específica (ASIC), lógica de puertas discreta, firmware o cualquier módulo de software programable convencional y un microprocesador. El módulo de software puede residir en memoria RAM, memoria flash, registros, o cualquier otra forma de medio de almacenamiento grabable conocido en la técnica. Como alternativa, cualquier máquina de estados, controlador o procesador convencionales pueden sustituir al microprocesador. ASIC a modo de ejemplo diseñados específicamente para la codificación de voz se describen en las patentes estadounidenses números 5.727.123 (McDonough *et al.*, expedida el 10 de marzo de 1998) y 5.784.532 (McDonough *et al.*, expedida el 21 de julio de 1998).

En la Figura 19A, un codificador 200 que puede utilizarse en un codificador de voz incluye un selector de modo 202, un módulo de estimación de tono 204, un módulo de análisis LP 206, un filtro de análisis LP 208, un módulo de cuantificación LP 210 y un módulo de cuantificación de residuo 212. Las tramas de voz de entrada  $s(n)$  se proporcionan al selector de modo 202, al módulo de estimación de tono 204, al módulo de análisis LP 206 y al filtro de análisis LP 208. El selector de modo 202 genera una indicación de modo M en función de la periodicidad, energía, relación de señal a ruido (SNR) o tasa de cruce por cero, entre otras características, de cada trama de voz de entrada  $s(n)$ . El selector de modo 202 también puede configurarse para generar la indicación de modo M en función de un resultado de la tarea T400, y/o de una salida de la segunda unidad de comparación A140, correspondiente a la detección de una señal tonal.

El modo M puede indicar un modo de codificación tal como CELP, NELP o PPP como los descritos en este documento y también puede indicar una velocidad de codificación. En el ejemplo mostrado en la Figura 19A, el selector de modo 202 también genera un índice de modo  $I_M$  (por ejemplo, una versión codificada de la indicación de modo M para la transmisión). Varios procedimientos de clasificación de tramas de voz según la periodicidad se describen en la patente estadounidense número 5.911.128 (DeJaco, expedida el 8 de junio de 1999). Tales procedimientos también se incorporan en las normas provisionales TIA/EIA IS-127 y TIA/EIA IS-733 de la Asociación de la Industria de las Telecomunicaciones. Un esquema de decisión de modo a modo de ejemplo también se describe en la patente estadounidense número 6.691.084 (Manjunath *et al.*, expedida el 10 de febrero de 2004).

## ES 2 347 473 T3

El módulo de estimación de tono 204 genera un índice de tono  $I_p$  y un valor de retardo  $P_0$  en función de cada trama de voz de entrada  $s(n)$ . El módulo de análisis LP 206 lleva a cabo un análisis predictivo lineal en cada trama de voz de entrada  $s(n)$  para generar un conjunto de parámetros LP (por ejemplo, coeficientes de filtro  $a$ ). Los parámetros LP se reciben por el módulo de cuantificación LP 210, posiblemente después de la conversión a otra forma tal como LSP, LSF o LSP (como alternativa, tal conversión puede producirse en el módulo 210). En este ejemplo, el módulo de cuantificación LP 210 también recibe la indicación de modo M, llevando a cabo de ese modo el proceso de cuantificación de una manera dependiente del modo.

El módulo de cuantificación LP 210 genera un índice LP  $I_{LP}$  (por ejemplo, un índice en un libro de códigos de cuantificación) y un conjunto cuantificado de parámetros LP  $\hat{a}$ . El filtro de análisis LP 208 recibe el conjunto cuantificado de parámetros LP  $\hat{a}$  además de la trama de voz de entrada  $s(n)$ . El filtro de análisis LP 208 genera una señal de residuo LP  $u[n]$ , la cual representa el error entre las tramas de voz de entrada  $s(n)$  y la voz reconstruida en función de los parámetros estimados  $\hat{a}$  lineales y cuantificados. El residuo LP  $u[n]$  y la indicación de modo M se proporcionan al módulo de cuantificación de residuo 212. En este ejemplo, el conjunto cuantificado de parámetros LP  $\hat{a}$  también se proporciona al módulo de cuantificación de residuo 212. En función de estos valores, el módulo de cuantificación de residuo 212 genera un índice de residuo  $I_R$  y una señal de residuo cuantificada  $\hat{u}[n]$ . Cada uno de los codificadores 100 y 106 mostrados en la Figura 18 puede configurarse para incluir una implementación del codificador 200 junto con una implementación del aparato A100.

En la Figura 19B, un descodificador 300 que puede usarse en un codificador de voz incluye un módulo de descodificación de parámetros LP 302, un módulo de descodificación de residuo 304, un módulo de descodificación de modo 306 y un filtro de síntesis LP 308. El módulo de descodificación de modo 306 recibe y descodifica un índice de modo  $I_M$ , generando a partir del mismo una indicación de modo M. El módulo de descodificación de parámetros LP 302 recibe el modo M y un índice LP  $I_{LP}$ . El módulo de descodificación de parámetros LP 302 descodifica los valores recibidos para generar un conjunto cuantificado de parámetros LP  $\hat{a}$ . El módulo de descodificación de residuo 304 recibe un índice de residuo  $I_R$ , un índice de tono  $I_p$  y el índice de modo  $I_M$ . El módulo de descodificación de residuo 304 descodifica los valores recibidos para generar una señal de residuo cuantificada  $\hat{u}[n]$ . La señal de residuo cuantificada  $\hat{u}[n]$  y el conjunto cuantificado de parámetros LP  $\hat{a}$  se reciben por el filtro de síntesis LP 308, el cual sintetiza una señal de voz de salida descodificada  $\hat{s}[n]$  a partir de los mismos. Cada uno de los descodificadores 104 y 110 mostrados en la Figura 18 pueden configurarse para incluir una implementación del descodificador 300.

La Figura 20 muestra un diagrama de flujo de tareas de selección de modo que pueden llevarse a cabo por un codificador de voz que incluya una implementación del selector de modo 202. En la tarea 400, el selector de modo recibe muestras digitales de una señal de voz en tramas sucesivas. Tras recibir una trama dada, el selector de modo avanza hasta la tarea 402. En la tarea 402, el selector de modo detecta la energía de la trama. La energía es una medida de la actividad de voz de la trama. La detección de voz se lleva a cabo sumando los cuadrados de las amplitudes de las muestras de voz digitalizadas y comparando la energía resultante con un valor de umbral. La tarea 402 puede configurarse para adaptar este valor de umbral en función del nivel variable de ruido de fondo. Un detector de actividad de voz de umbral variable a modo de ejemplo se describe en la patente estadounidense mencionada anteriormente número 5.414.796. Algunos sonidos de voz sordos pueden ser muestras de energía extremadamente bajas que puede codificarse por error como ruido de fondo. Para reducir la posibilidad de un error de este tipo, la inclinación espectral (por ejemplo, el primer coeficiente de reflexión) de muestras de baja energía puede utilizarse para distinguir la voz sorda del ruido de fondo, tal y como se describe en la patente estadounidense mencionada anteriormente número 5.414.796.

Después de detectar la energía de la trama, el selector de modo avanza hasta la tarea 404. (Una implementación alternativa del selector de modo 202 está configurada para recibir la energía de trama de otro elemento del codificador de voz). En la tarea 404, el selector de modo determina si la energía de trama detectada es suficiente para clasificar la trama como conteniendo información de voz. Si la energía de trama detectada cae por debajo de un nivel de umbral predefinido, el codificador de voz avanza hasta la tarea 406. En la tarea 406, el codificador de voz codifica la trama como ruido de fondo (es decir, silencio). En una configuración, la trama de ruido de fondo se codifica a 1/8 de velocidad (por ejemplo, 1 kbps). Si en la tarea 404 la energía de trama detectada alcanza o supera el nivel de umbral predefinido, la trama se clasifica como voz y el selector de modo avanza hasta la tarea 408.

En la tarea 408, el selector de modo determina si la trama es voz sorda. Por ejemplo, la tarea 408 puede configurarse para examinar la periodicidad de la trama. Varios procedimientos conocidos de determinación de periodicidad incluyen, por ejemplo, el uso de cruces por cero y el uso de funciones de autocorrelación normalizadas (NACF). En particular, utilizar cruces por cero y NACF para detectar la periodicidad se describe en las patentes estadounidenses mencionadas anteriormente números 5.911.128 y 6.691.084. Además, los procedimientos anteriores utilizados para distinguir voz sonora de voz sorda se incorporan en las normas provisionales TIA/EIA IS-127 y TIA/EIA IS-733 de la Asociación de la Industria de las Telecomunicaciones. Si en la tarea 408 se determina que la trama es de voz sorda, el codificador de voz avanza hasta la tarea 410. En la tarea 410, el codificador de voz codifica la trama como voz sorda. En una configuración, las tramas de voz sordas se codifican a un cuarto de la velocidad (por ejemplo, a 2,6 kbps). Si en la tarea 408 se determina que la trama no es de voz sorda, el selector de modo avanza hasta la tarea 412.

En la tarea 412, el selector de modo determina si la trama es de voz de transición. La tarea 412 puede configurarse para utilizar procedimientos de detección de periodicidad conocidos en la técnica (por ejemplo, tal y como se describe en la patente estadounidense mencionada anteriormente número 5.911.128). Si se determina que la trama es de voz de

## ES 2 347 473 T3

transición, el codificador de voz avanza hasta la tarea 414. En la tarea 414, la trama se codifica como voz de transición (es decir, transición desde voz sorda a voz sonora). En una configuración, la trama de voz de transición se codifica según un procedimiento de codificación interpoladora de múltiples impulsos descrito en la patente estadounidense número 6.260.017 (Das *et al.*, expedida el 10 de julio de 2001). Un modo CELP también puede utilizarse para codificar 5 tramas de voz de transición. En otra configuración, la trama de voz de transición se codifica a velocidad completa (por ejemplo, 13,2 kbps).

Si en la tarea 412, el selector de modo determina que la trama no es de voz de transición, el codificador de voz avanza hasta la tarea 416. En la tarea 416, el codificador de voz codifica la trama como voz sonora. En una 10 configuración, las tramas de voz sonoras pueden codificarse a media velocidad (por ejemplo, 6,2 kbps), o a un cuarto de velocidad, utilizando un modo de codificación PPP. También es posible codificar tramas de voz sonoras a velocidad completa utilizando un modo de codificación PPP u otro modo de codificación (por ejemplo, 13,2 kbps u 8 kbps en un codificador CELP de 8k). Sin embargo, los expertos en la técnica apreciarán que codificar tramas sonoras a media o a un cuarto de velocidad permite al codificador ahorrar un ancho de banda valioso aprovechando la naturaleza de estado 15 estable de las tramas sonoras. Además, independientemente de la velocidad utilizada para codificar la voz sonora, la voz sonora se codifica de manera ventajosa utilizando información de tramas anteriores.

La descripción anterior de un códec de voz multimodo describe el procesamiento de una trama de entrada que contiene voz. Obsérvese que se utiliza un proceso de clasificación de los contenidos de la trama para seleccionar el mejor 20 modo mediante el cual codificar la trama. Varios modos de codificador/descodificador se describen en las siguientes secciones. Los diferentes modos de codificador/descodificador funcionan según diferentes modos de codificación. Determinados modos son más eficaces en partes de codificación de la señal de voz  $s(n)$  que presenten determinadas propiedades. Tal y como se ha indicado anteriormente, el selector de modo 202 puede configurarse para invalidar una decisión de codificación tal y como se muestra en la Figura 20 (por ejemplo, generada por la tarea 408 y/o 412), en 25 función del resultado de la tarea T400 y/o de una salida de la segunda unidad de comparación A140.

En una configuración se elige un modo “predictivo lineal excitado por código” (CELP) para codificar tramas clasificadas como voz de transición. El modo CELP excita un modelo de tracto vocal predictivo lineal con una versión 30 cuantificada de la señal residual de predicción lineal. De todos los modos de codificador/descodificador descritos en este documento, CELP produce generalmente la reproducción de voz más precisa pero requiere la mayor velocidad binaria. En una configuración, el modo CELP lleva a cabo una codificación a 8500 bits por segundo. En otra configuración, la codificación CELP de una trama se lleva a cabo a una velocidad seleccionada entre la velocidad completa y la mitad de velocidad. Un modo CELP también puede seleccionarse según un resultado de la tarea T400 y/o una salida 35 de la segunda unidad de comparación A140, correspondiente a la detección de una señal tonal.

Un modo de “periodo de tono de prototipo” (PPP) puede elegirse para codificar tramas clasificadas como voz sonora. La voz sonora contiene componentes periódicos que varían lentamente en el tiempo y que se utilizan por 40 el modo PPP. El modo PPP codifica solamente un subconjunto de los periodos de tono dentro de cada trama. Los periodos restantes de la señal de voz se reconstruyen mediante interpolación entre estos periodos de prototipo. Al utilizar la periodicidad de la voz sonora, PPP puede conseguir una velocidad binaria más baja que CELP y producir aun así la señal de voz de una manera perceptivamente precisa. En una configuración, el modo PPP lleva a cabo una codificación a 3900 bits por segundo. En otra configuración, la codificación PPP de una trama se lleva a cabo a una velocidad seleccionada entre una velocidad completa, la mitad de la velocidad y un cuarto de la velocidad. Un modo 45 de “interpolación de forma de onda” (WI) o de “interpolación de forma de onda de prototipo” (PWI) también puede utilizarse para codificar tramas clasificadas como voz sonora.

Un modo “predictivo lineal excitado por ruido” (NELP) puede elegirse para codificar tramas clasificadas como voz sorda. NELP utiliza una señal filtrada de ruido pseudoaleatorio para modelar la voz sorda. NELP utiliza el modelo 50 más sencillo para la voz codificada y, por lo tanto, consigue la velocidad binaria más baja. En una configuración, el modo NELP lleva a cabo una codificación a 1500 bits por segundo. En otra configuración, la codificación NELP de una trama se lleva a cabo a una velocidad seleccionada entre la mitad de velocidad y un cuarto de velocidad.

Puede utilizarse frecuentemente la misma técnica de codificación a diferentes velocidades binarias, con niveles 55 de rendimiento variables. Por lo tanto, los diferentes modos de codificador/descodificador pueden representar diferentes técnicas de codificación o la misma técnica de codificación funcionando a diferentes velocidades binarias, o combinaciones de lo anterior. Los expertos en la técnica reconocerán que aumentando el número de modos de codificador/descodificador se permitirá una mayor flexibilidad cuando se elija un modo, lo que puede dar como resultado una velocidad binaria media más baja, pero aumentará la complejidad del sistema global. La combinación particular utilizada en cualquier sistema dado quedará dictaminada por los recursos disponibles del sistema y por el entorno 60 específico de la señal. Un codificador de voz u otro aparato que lleve a cabo una implementación de la tarea T400 descrita en este documento y/o que incluya una implementación del aparato A100 descrito en este documento, puede configurarse para seleccionar una velocidad de configuración particular (por ejemplo, velocidad completa o media velocidad) según un resultado de la tarea T400 y/o una salida de la segunda unidad de comparación A140, que indica la detección de una señal tonal.

La presentación anterior de las configuraciones descritas se proporciona para permitir que cualquier experto en la 65 técnica realice o utilice los procedimientos y otras estructuras dados a conocer en este documento. Los diagramas de flujo y otras estructuras mostradas y descritas en este documento son solamente ejemplos, y otras variantes de estas

## ES 2 347 473 T3

estructuras también están dentro del alcance de la invención. Son posibles varias modificaciones de estas configuraciones, y los principios genéricos presentados en este documento también pueden aplicarse a otras configuraciones.

5 Cada una de las configuraciones descritas en este documento puede implementarse en parte o totalmente como un circuito cableado, como una configuración de circuitos fabricada en un circuito integrado de aplicación específica, o como un programa de firmware cargado en memoria no volátil o como un programa de software cargado desde o en un medio de almacenamiento de datos como código legible por máquina, siendo tal código instrucciones ejecutables por una disposición de elementos lógicos tales como un microprocesador u otra unidad de procesamiento de señales digitales. El medio de almacenamiento de datos puede ser una disposición de elementos de almacenamiento  
10 tales como una memoria de semiconductor (que puede incluir sin limitación una RAM (memoria de acceso aleatorio) dinámica o estática, una ROM (memoria de solo lectura), y/o una RAM flash), o una memoria ferroeléctrica, magnetorresistiva, ovónica, polimérica o de cambio de fase; o un medio de disco tal como un disco magnético u óptico. Debe entenderse que el término "software" incluye código fuente, código de lenguaje ensamblador, código máquina, código binario, firmware, macrocódigo, microcódigo, uno o más conjuntos o secuencias de instrucciones ejecutables  
15 por una disposición de elementos lógicos, y cualquier combinación de tales ejemplos.

Cada uno de los procedimientos dados a conocer en este documento también pueden realizarse de manera tangible (por ejemplo, en uno o más medios de almacenamiento de datos como los enumerados anteriormente) como uno o más conjuntos de instrucciones legibles y/o ejecutables por una máquina que incluya una disposición de elementos lógicos (por ejemplo, un procesador, un microprocesador, un microcontrolador u otra máquina de estados finitos). Por lo tanto, la presente invención no pretende limitarse a las configuraciones mostradas anteriormente sino que en cambio debe concedérsele el alcance más amplio compatible con los principios y características novedosas dados a conocer de cualquier manera en este documento, incluyendo las reivindicaciones adjuntas como campo, las cuales forman una  
25 parte de la invención original.

25

30

35

40

45

50

55

60

65

REIVINDICACIONES

1. Un procedimiento de procesamiento de señales de audio, comprendiendo dicho procedimiento:

llevar a cabo una operación de codificación en una porción de tiempo de una señal de audio digitalizada, en el que dicha operación de codificación incluye una pluralidad ordenada de iteraciones;

en cada una de la pluralidad ordenada de iteraciones, calcular un valor de una medida de ganancia de la operación de codificación;

para cada uno de una primera pluralidad de valores de umbral, determinar la iteración, entre la pluralidad ordenada, en la que se produce un cambio en un estado de una primera relación entre el valor calculado y el valor de umbral, y almacenar una indicación de la iteración; y

comparar al menos una de las indicaciones almacenadas con al menos un valor de umbral correspondiente.

2. El procedimiento de procesamiento de señales de audio según la reivindicación 1, en el que dicha comparación de al menos una de las indicaciones almacenadas con al menos un valor de umbral correspondiente incluye comparar la al menos una de las indicaciones almacenadas con un valor correspondiente de una segunda pluralidad de valores de umbral.

3. El procedimiento de procesamiento de señales de audio según la reivindicación 1, en el que la operación de codificación es una operación de codificación predictiva lineal.

4. El procedimiento de procesamiento de señales de audio según la reivindicación 1, en el que dicha realización de una operación de codificación incluye calcular una pluralidad de coeficientes de filtro relacionados con la porción de tiempo.

5. El procedimiento de procesamiento de señales de audio según la reivindicación 4, comprendiendo dicho procedimiento, en respuesta a un resultado de dicha comparación, reducir la magnitud de al menos uno de los coeficientes de filtro.

6. El procedimiento de procesamiento de señales de audio según la reivindicación 1, en el que dicha realización de una operación de codificación incluye calcular una pluralidad de coeficientes de reflexión relacionados con la porción de tiempo.

7. El procedimiento de procesamiento de señales de audio según la reivindicación 6, en el que dicho cálculo de un valor de una medida de ganancia incluye calcular el valor en función de al menos uno de la pluralidad de coeficientes de reflexión.

8. El procedimiento de procesamiento de señales de audio según la reivindicación 1, en el que la medida de ganancia de la operación de codificación es una entre A) una ganancia de predicción y B) un error de predicción.

9. El procedimiento de procesamiento de señales de audio según la reivindicación 1, en el que dicha comparación de al menos una de las indicaciones almacenadas con al menos un valor de umbral correspondiente incluye comparar al menos una de las indicaciones almacenadas con cada uno de un valor de umbral superior correspondiente y un valor de umbral inferior correspondiente.

10. El procedimiento de procesamiento de señales de audio según la reivindicación 1, en el que la medida de ganancia de la operación de codificación está basada en una relación entre A) la energía de la porción de tiempo y B) la energía de un residuo de la iteración correspondiente de la operación de codificación.

11. El procedimiento de procesamiento de señales de audio según la reivindicación 1, en el que, para cada uno de la primera pluralidad de valores de umbral, el estado de la primera relación entre el valor calculado y el valor de umbral presenta A) un primer valor cuando el valor calculado es mayor que el valor de umbral y B) un segundo valor, diferente del primer valor, cuando el valor calculado es inferior al valor de umbral.

12. El procedimiento de procesamiento de señales de audio según la reivindicación 1, comprendiendo dicho procedimiento seleccionar, en función de un resultado de dicha comparación, un modo de codificación para la porción de tiempo.

13. El procedimiento de procesamiento de señales de audio según la reivindicación 1, comprendiendo dicho procedimiento, en respuesta a un resultado de dicha comparación, utilizar al menos un índice de libro de códigos para codificar una señal de excitación de la porción de tiempo.

## ES 2 347 473 T3

14. El procedimiento de procesamiento de señales de audio según la reivindicación 1, comprendiendo dicho procedimiento, en respuesta a un resultado de dicha comparación, identificar una señal de multifrecuencia de doble tono incluida en la porción de tiempo.

5 15. El procedimiento de procesamiento de señales de audio según la reivindicación 1, comprendiendo dicho procedimiento, en respuesta a un resultado de dicha comparación, determinar una frecuencia de cada uno de los al menos dos componentes de frecuencia de la porción de tiempo.

10 16. El procedimiento de procesamiento de señales de audio según la reivindicación 1, comprendiendo dicho procedimiento, en función de al menos una de las indicaciones almacenadas, decidir que la porción de tiempo es una de entre A) una señal de voz y B) una señal tonal, donde dicha decisión incluye dicha comparación de al menos una de las indicaciones almacenadas con al menos un valor de umbral correspondiente.

15 17. Un medio de almacenamiento de datos que contiene instrucciones legibles por máquina que describen el procedimiento según la reivindicación 1.

18. Un aparato de procesamiento de señales de audio, comprendiendo dicho aparato:

20 medios para llevar a cabo una operación de codificación en una porción de tiempo de una señal de audio digitalizada, en el que dicha operación de codificación incluye una pluralidad ordenada de iteraciones;

medios para calcular, en cada una de la pluralidad ordenada de iteraciones, un valor de una medida de ganancia de la operación de codificación;

25 medios para determinar, para cada uno de una primera pluralidad de valores de umbral, la iteración, entre la pluralidad ordenada, en la que se produce un cambio en un estado de una primera relación entre el valor calculado y el valor de umbral, y para almacenar una indicación de la iteración; y

30 medios para comparar al menos una de las indicaciones almacenadas con al menos un valor de umbral correspondiente.

19. El aparato de procesamiento de señales de audio según la reivindicación 18, en el que dichos medios para comparar al menos una de las indicaciones almacenadas con al menos un valor de umbral correspondiente están configurados para comparar la al menos una de las indicaciones almacenadas con un valor correspondiente de una segunda pluralidad de valores de umbral.

20. El aparato de procesamiento de señales de audio según la reivindicación 18, en el que la medida de ganancia de la operación de codificación es una entre A) una ganancia de predicción y B) un error de predicción.

40 21. El aparato de procesamiento de señales de audio según la reivindicación 18, en el que la medida de ganancia de la operación de codificación está basada en una relación entre A) la energía de la porción de tiempo y B) la energía de un residuo de la iteración correspondiente de la operación de codificación.

45 22. El aparato de procesamiento de señales de audio según la reivindicación 18, en el que dichos medios para comparar al menos una de las indicaciones almacenadas con al menos un valor de umbral correspondiente están configurados para comparar al menos una de las indicaciones almacenadas con cada uno de un valor de umbral superior correspondiente y un valor de umbral inferior correspondiente.

50 23. El aparato de procesamiento de señales de audio según la reivindicación 18, en el que, para cada uno de la primera pluralidad de valores de umbral, el estado de la primera relación entre el valor calculado y el valor de umbral presenta A) un primer valor cuando el valor calculado es mayor que el valor de umbral y B) un segundo valor, diferente del primer valor, cuando el valor calculado es inferior al valor de umbral.

55 24. El aparato de procesamiento de señales de audio según la reivindicación 18, comprendiendo dicho aparato medios para seleccionar, en función de una salida de dichos medios de comparación, un modo de codificación para la porción de tiempo.

60 25. El aparato según la reivindicación 18, en el que los medios para llevar a cabo una operación de codificación es un calculador de coeficientes (A110), configurado para llevar a cabo una operación de codificación para calcular una pluralidad de coeficientes; los medios de cálculo son un calculador de medidas de ganancia (A120); y los medios de comparación comprenden una primera unidad de comparación (A130) configurada para determinar, para cada uno de una primera pluralidad de valores de umbral, la iteración, entre la pluralidad ordenada, en la que se produce un cambio en un estado de una primera relación entre el valor calculado y el valor de umbral, y para almacenar una indicación de la iteración; y

65 una segunda unidad de comparación (A140) configurada para comparar al menos una de las indicaciones almacenadas con al menos un valor de umbral correspondiente.

## ES 2 347 473 T3

26. El aparato de procesamiento de señales de audio según la reivindicación 25, en el que dicha segunda unidad de comparación (A140) está configurada para comparar la al menos una de las indicaciones almacenadas con un valor correspondiente de una segunda pluralidad de valores de umbral.

5 27. El aparato de procesamiento de señales de audio según la reivindicación 25, en el que dicha segunda unidad de comparación (A140) está configurada para comparar al menos una de las indicaciones almacenadas con cada uno de un valor de umbral superior correspondiente y un valor de umbral inferior correspondiente.

10 28. El aparato de procesamiento de señales de audio según la reivindicación 25, comprendiendo dicho aparato un selector de modo (202) configurado para seleccionar, en función de una salida de dicha segunda unidad de comparación (A140), un modo de codificación para la porción de tiempo.

15 29. Un teléfono celular que incluye el aparato según la reivindicación 25 y configurado para llevar a cabo, en función de una salida de dicha segunda unidad de comparación (A140), al menos uno de entre A) seleccionar un modo de codificación para la porción de tiempo y B) reducir una magnitud de al menos uno de entre la pluralidad de coeficientes.

20 30. Un codificador de voz que incluye el aparato según la reivindicación 25 y configurado para llevar a cabo, en función de una salida de dicha segunda unidad de comparación, al menos uno de entre A) seleccionar un modo de codificación para la porción de tiempo y B) reducir una magnitud de al menos uno de entre la pluralidad de coeficientes.

25

30

35

40

45

50

55

60

65

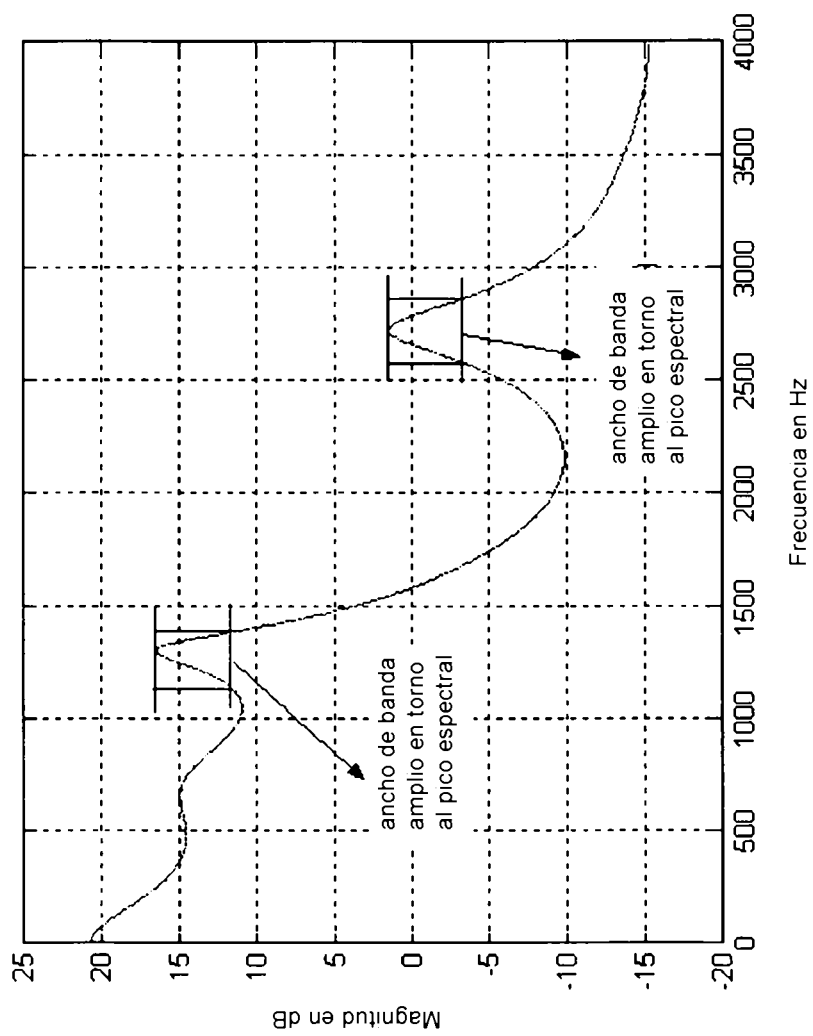


FIG. 1

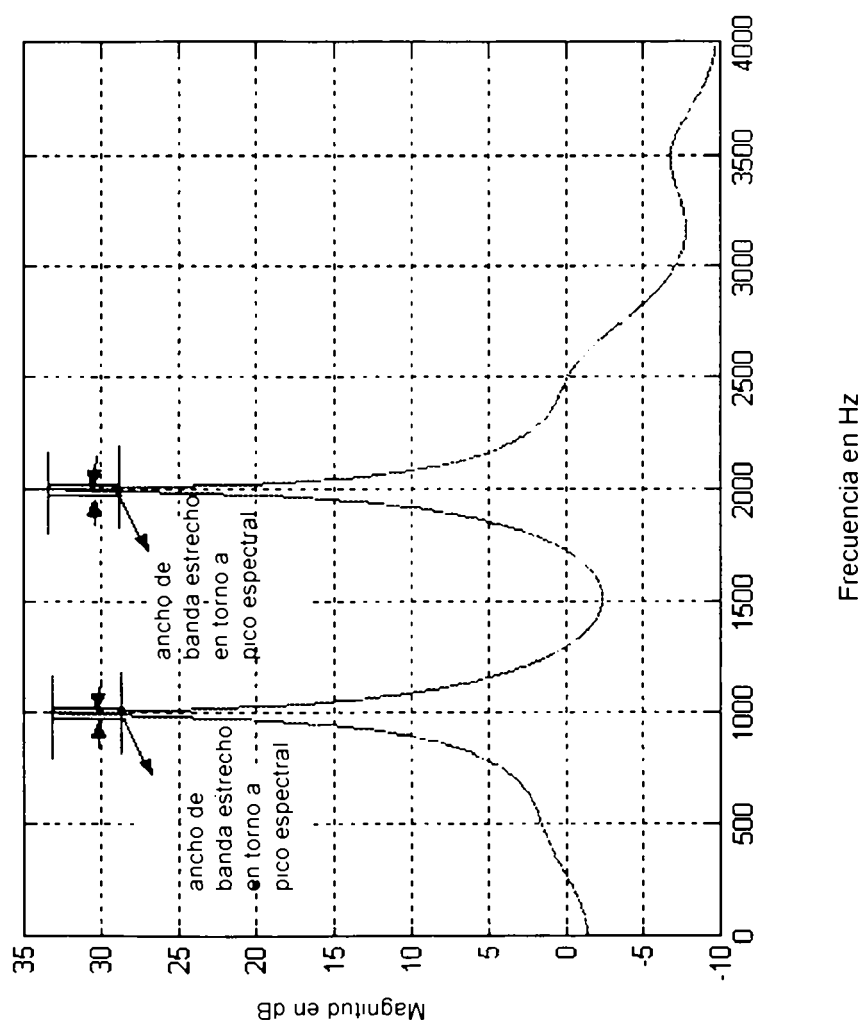


FIG. 2

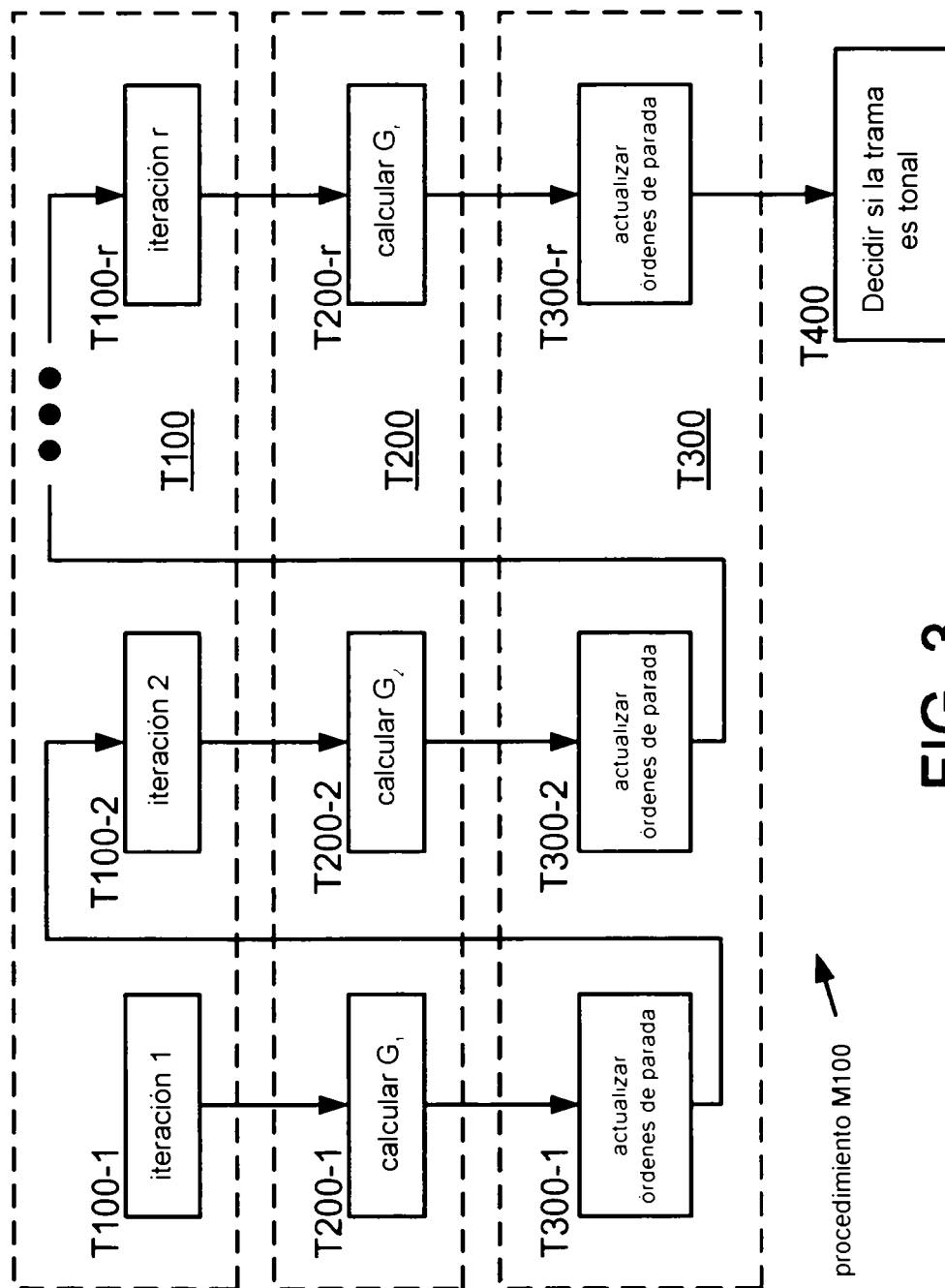


FIG. 3

FIG. 4a

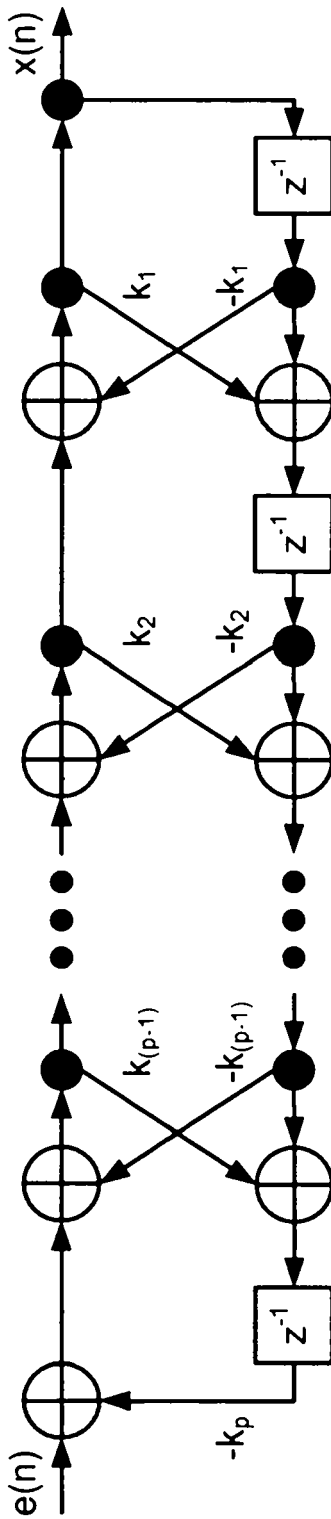
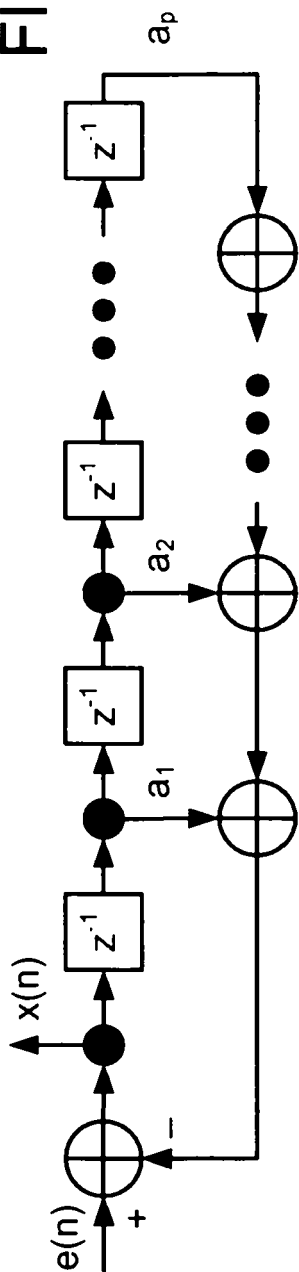


FIG. 4b

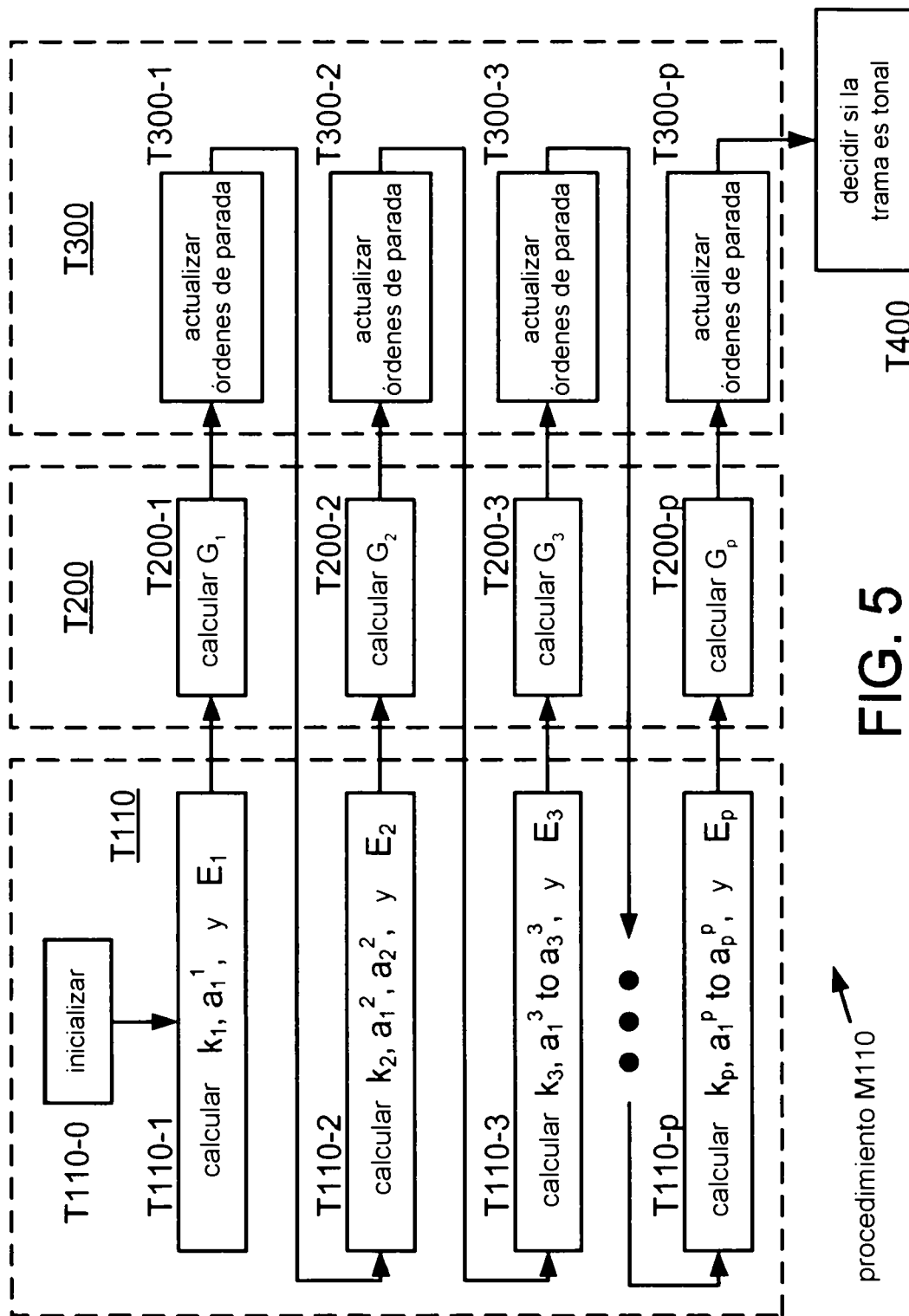


FIG. 5

```

para (m=1; m <= p; m++) {
    EP(m) = R(m);
    EN(m) = R(m-1);
}

para (h = 0; ; h++) {
    k(h+1) = -EP(h+1) / EN(1);
    E(h) = EN(1);
    EN(1) += k(h-1) * EP(h+1);

    si (h == p-1) break;

    para (m = h+2; m < p; m++) {
        EX = EP(m) + k(h+1)*EN(m-h);
        EN(m-h) += k(h+1) * EP(m);
        EP(m) = EX;
    }
}

```

FIG. 6

```

para (m = 1; m <= p; m++) {
    EP(m) = R(m);
    EN(m) = R(m-1);
}

para (h = 0; ; h++) {
    k(h+1) = -EP(h+1) / EN(1);
    E(h) = EN(1);
    EN(1) += k(h+1) * EP(h+1);
    G(h+1) = E(0) / EN(1);

    si (h == p-1) break;

    para (m = h+2; m < p; m++) {
        EX = EP(m) + k(h+1)*EN(m-h);
        EN(m-h) += k(h+1) * EP(m);
        EP(m) = EX;
    }
}

```

FIG. 7

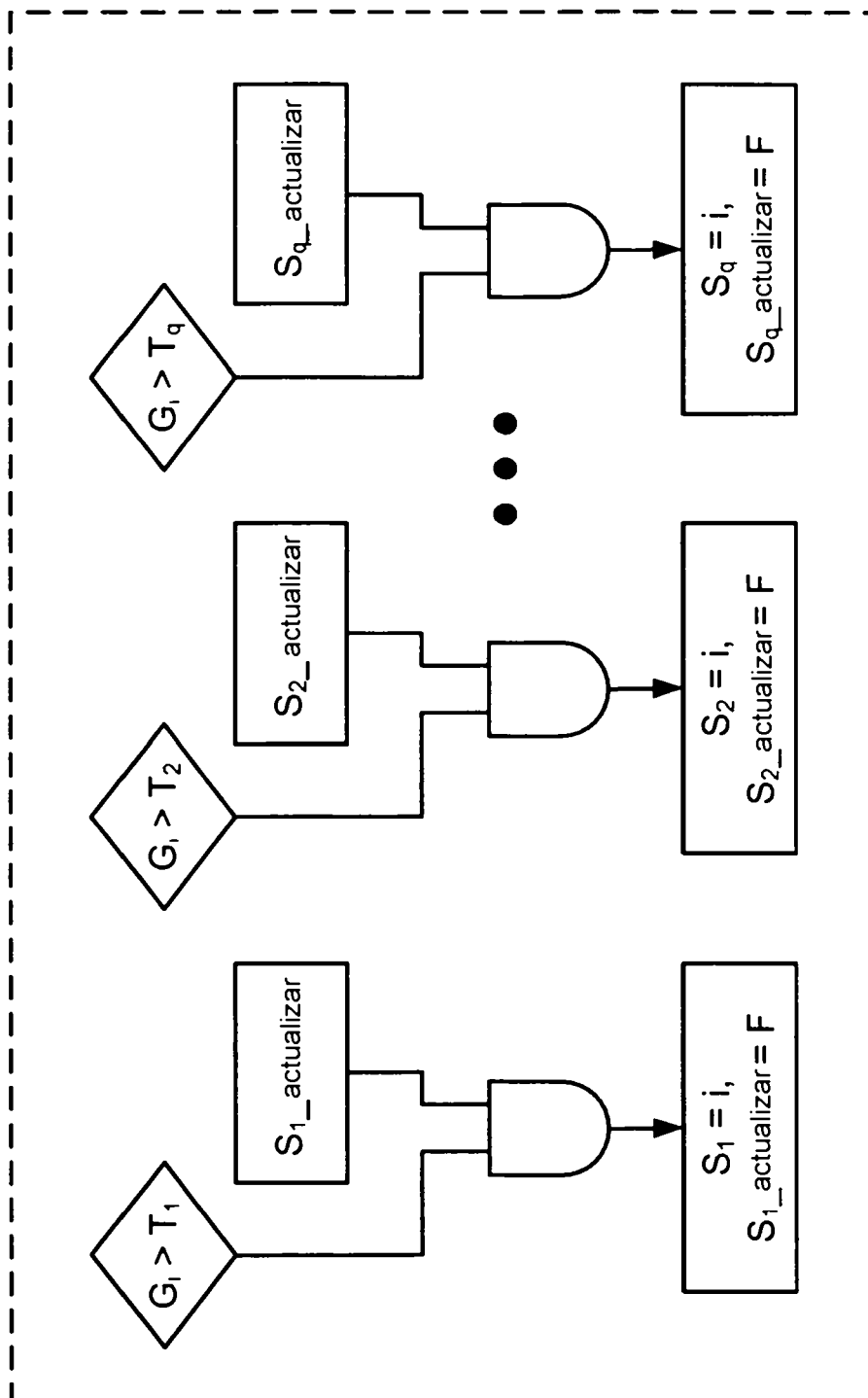


FIG. 8

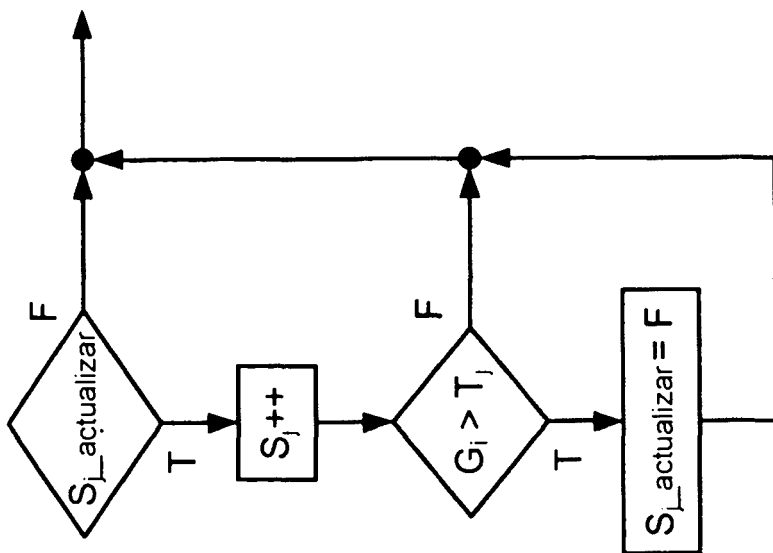


FIG. 9B

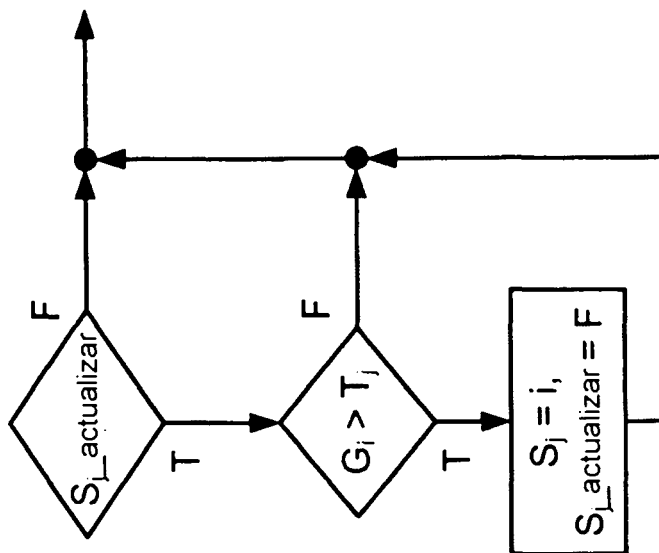


FIG. 9A

```

para (m = 1; m <= p; m++) { EP(m) = R(m); EN(m) = R(m-1); }

para (j = 1; j <= q; j++) { S_actualizar(j) = 1; S(j) = 0; }

para (h = 0; ; h++) {
    k(h+1) = -EP(h+1) / EN(1);
    E(h) = EN(1);
    EN(1) += k(h+1) * EP(h+1);
    G(h+1) = E(0) / EN(1);

    para (j = 1; j <= q; j++) {
        si (S_actualizar(j)) {
            S(j)++;
            si (G(h+1) > T(j)) S_actualizar(j) = 0;
        }
    }

    si (h == p-1) break;

    para (m = h+2; m < p; m++) {
        EX = EP(m) + k(h+1) * EN(m-h);
        EN(m-h) += k(h+1) * EP(m);
        EP(m) = EX;
    }
}

```

FIG. 10

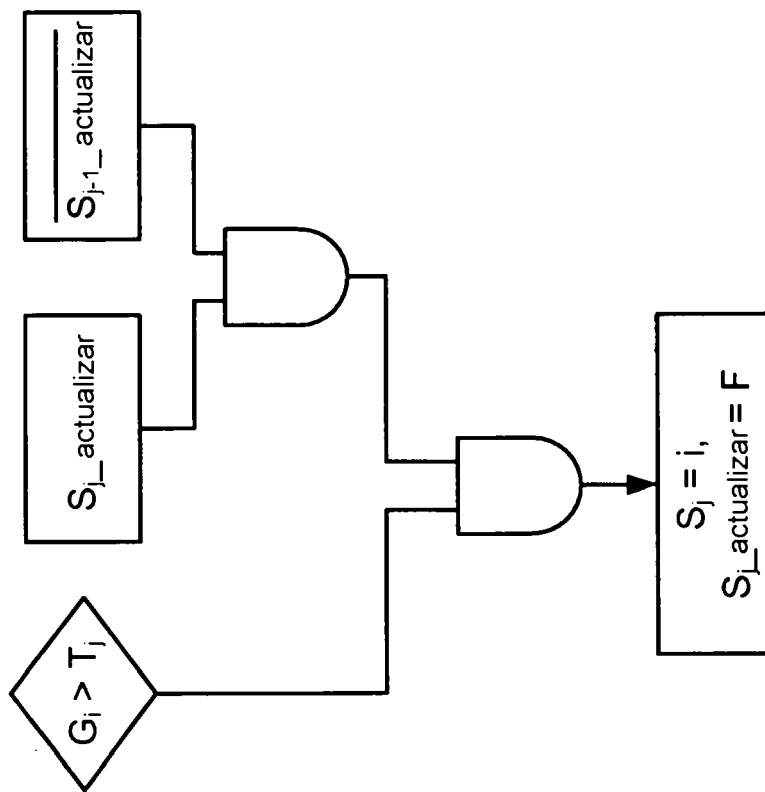


FIG. 11

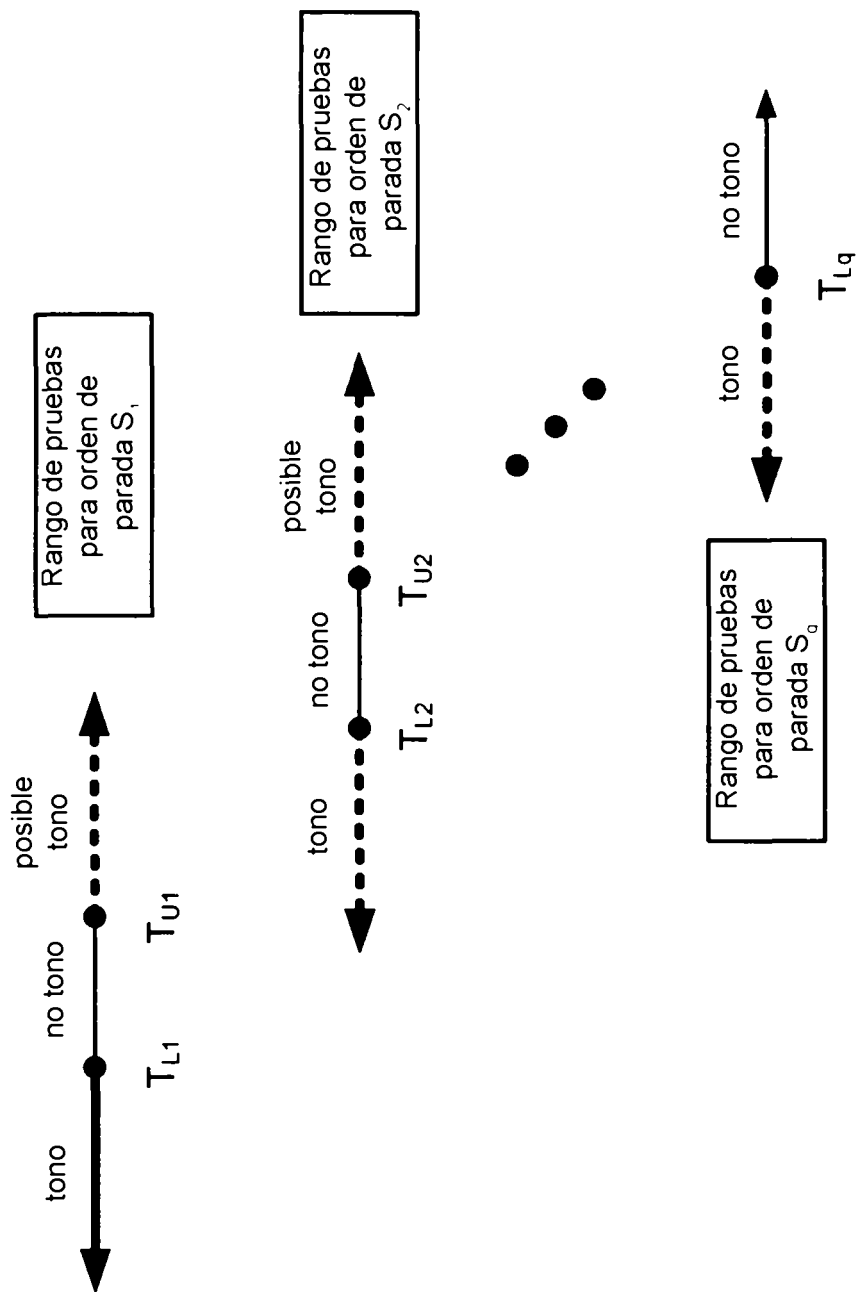


FIG. 12

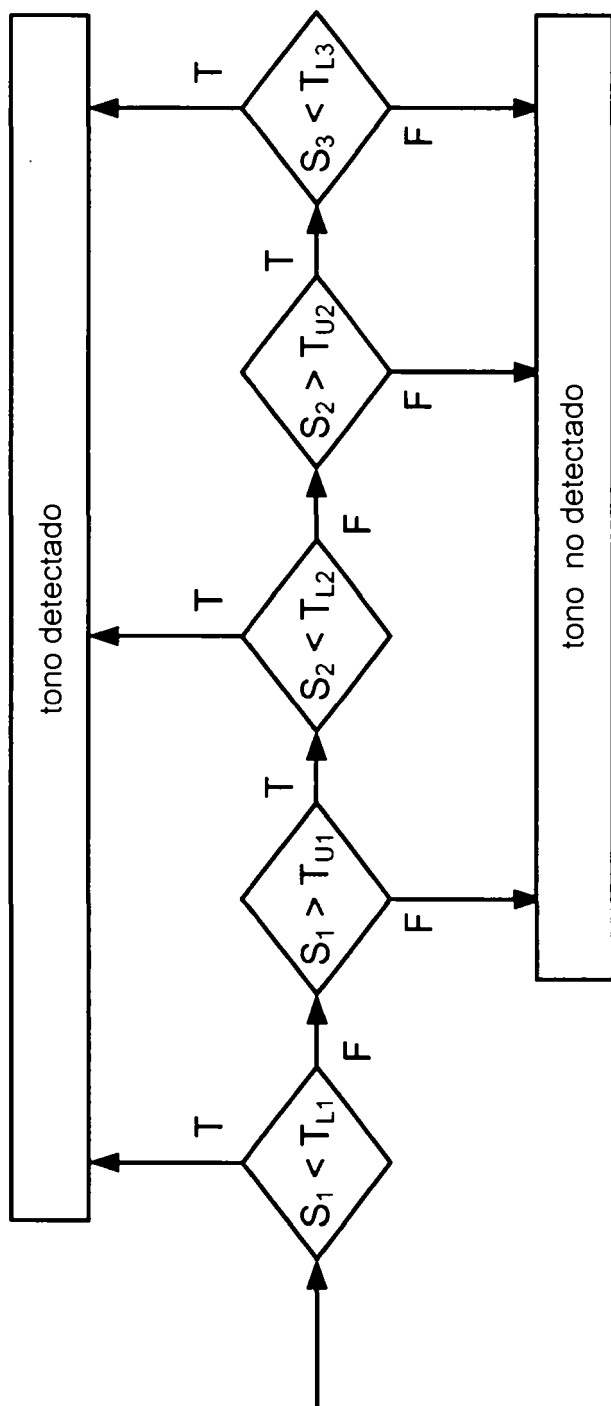


FIG. 13

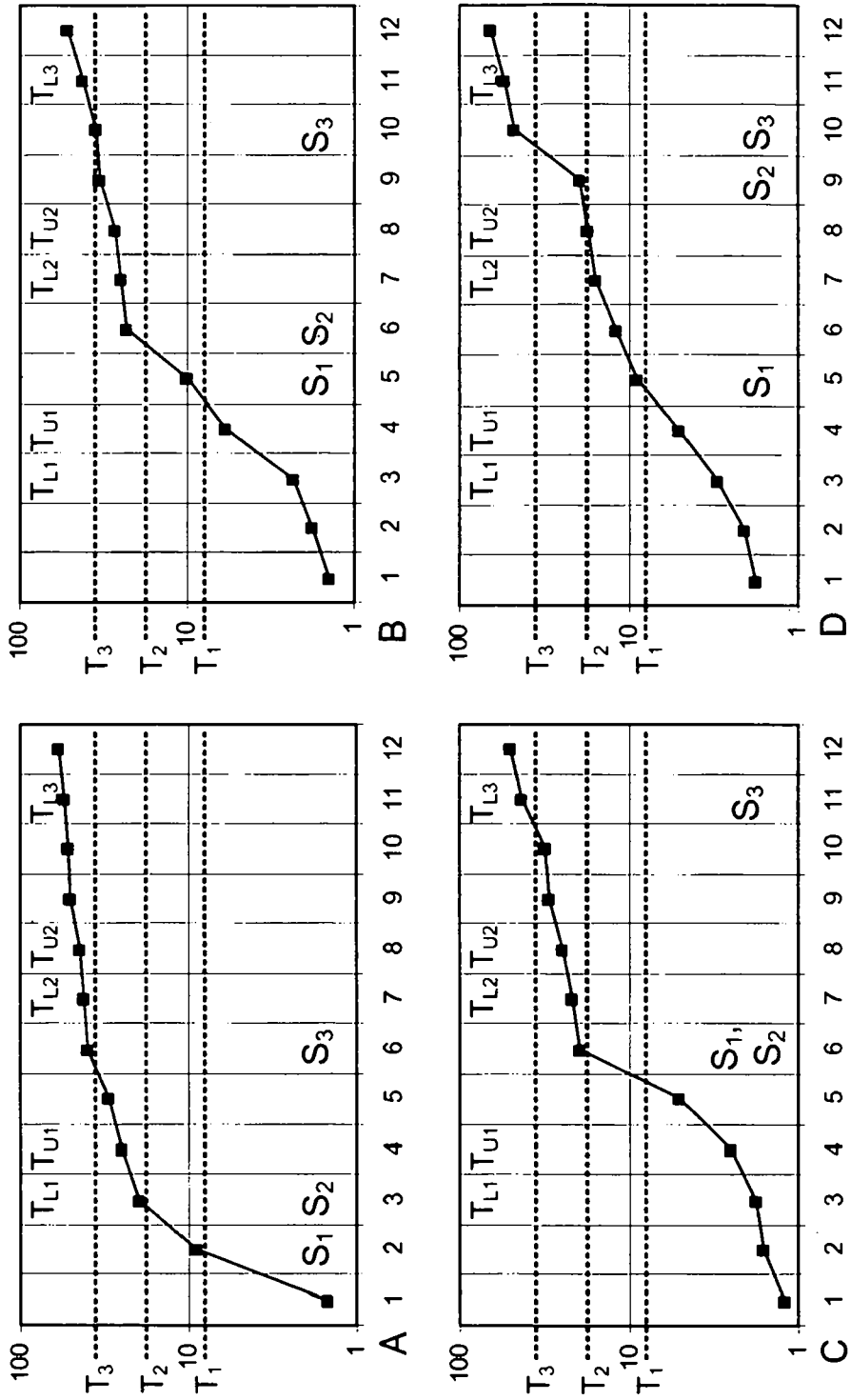


FIG. 14

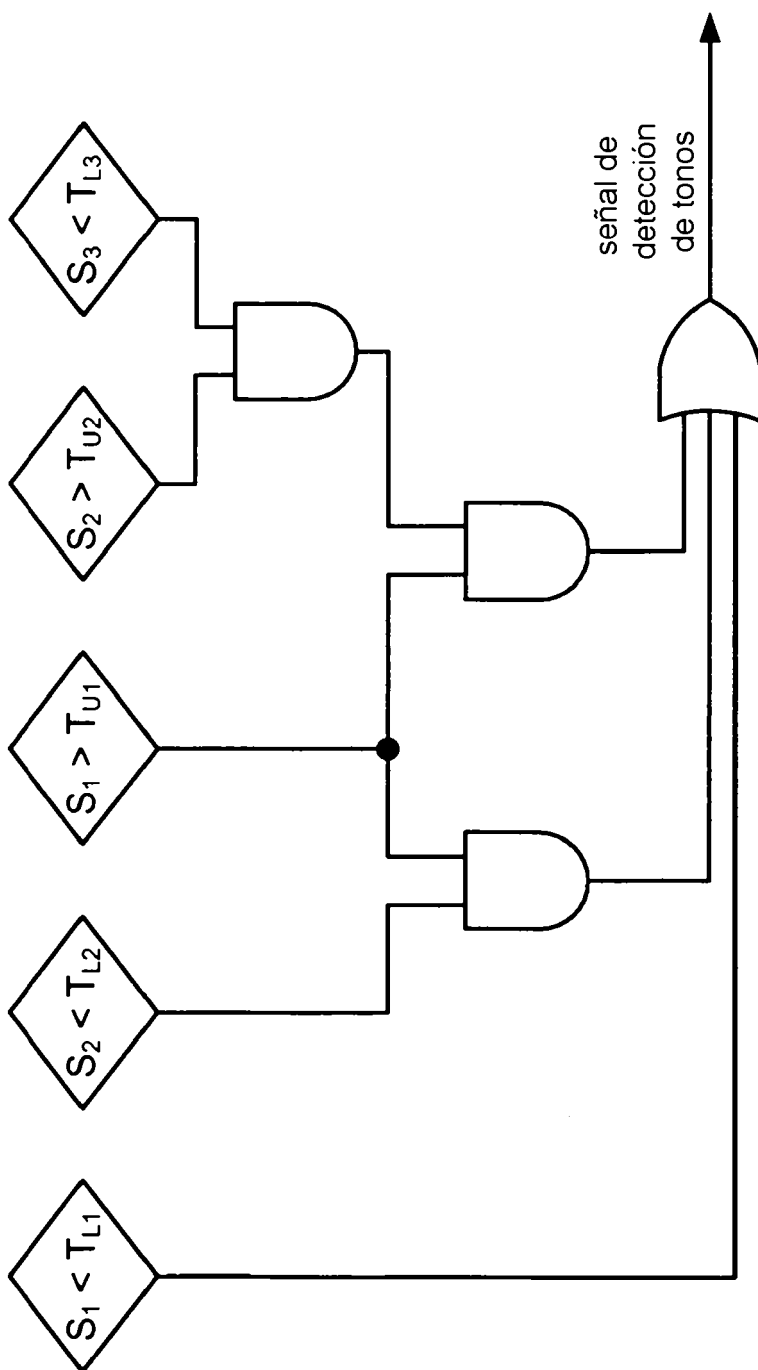


FIG. 15

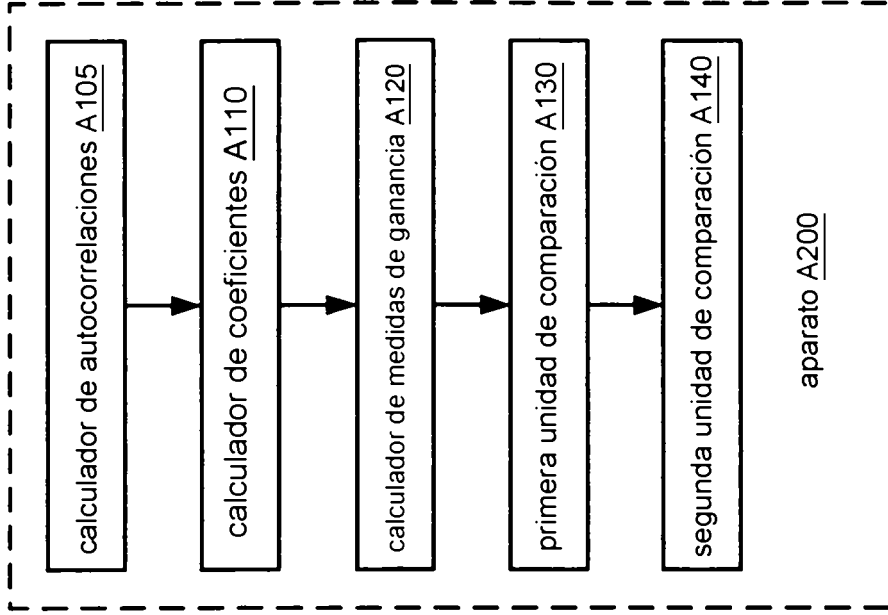


FIG. 16B

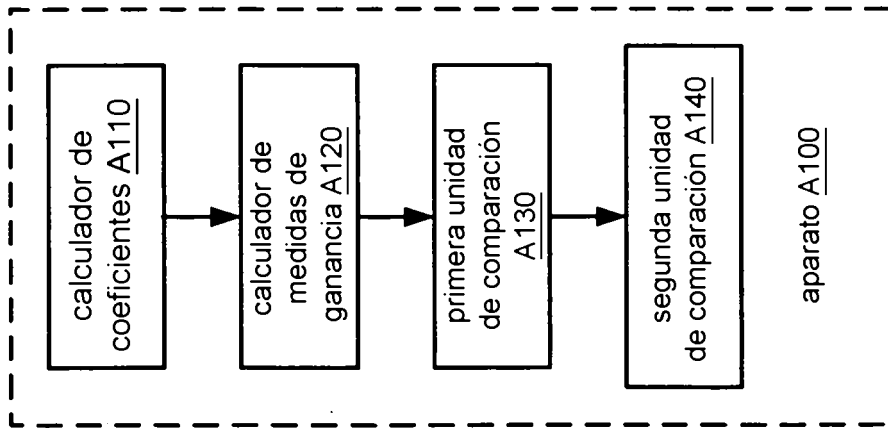


FIG. 16A

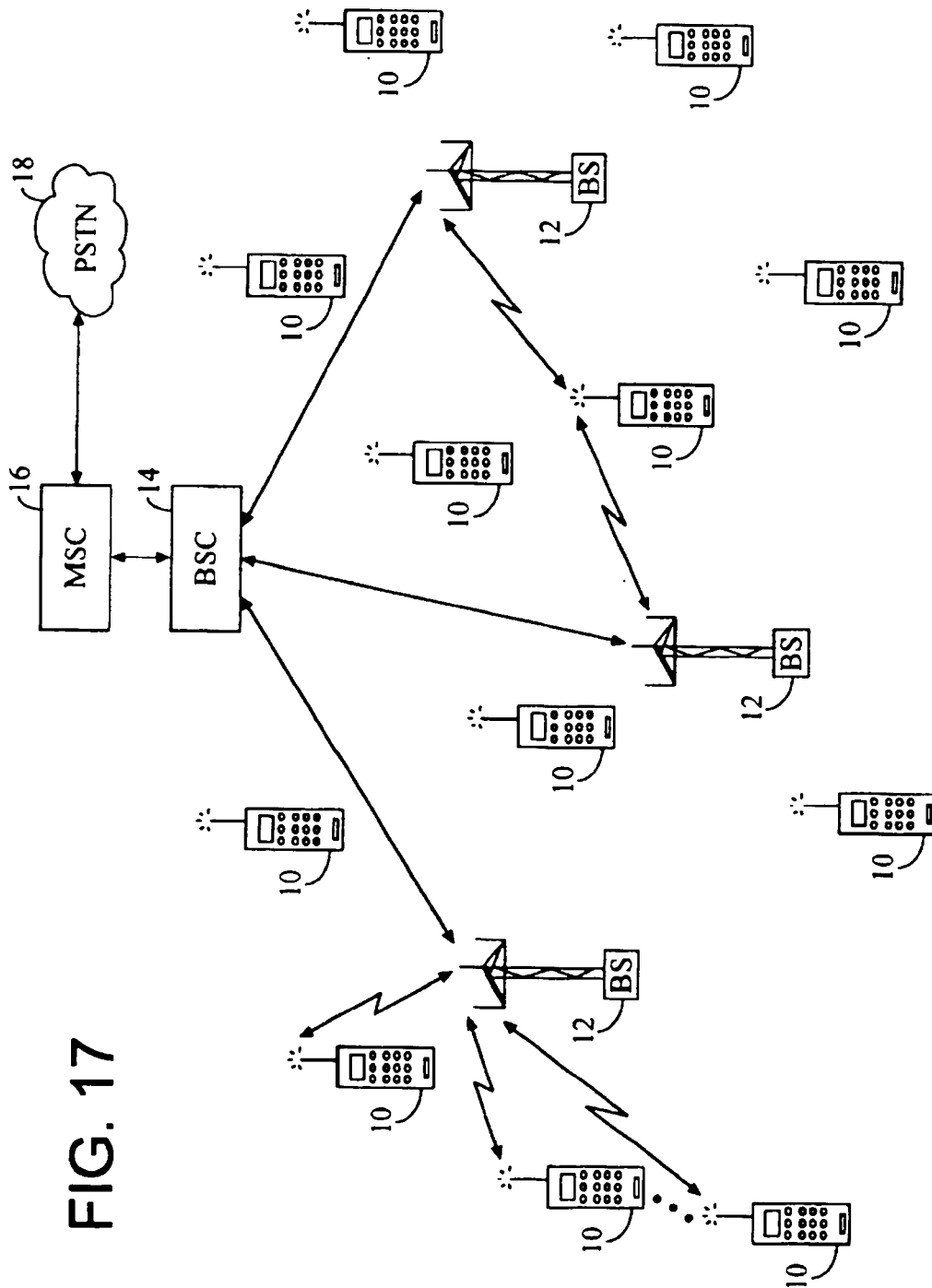


FIG. 17

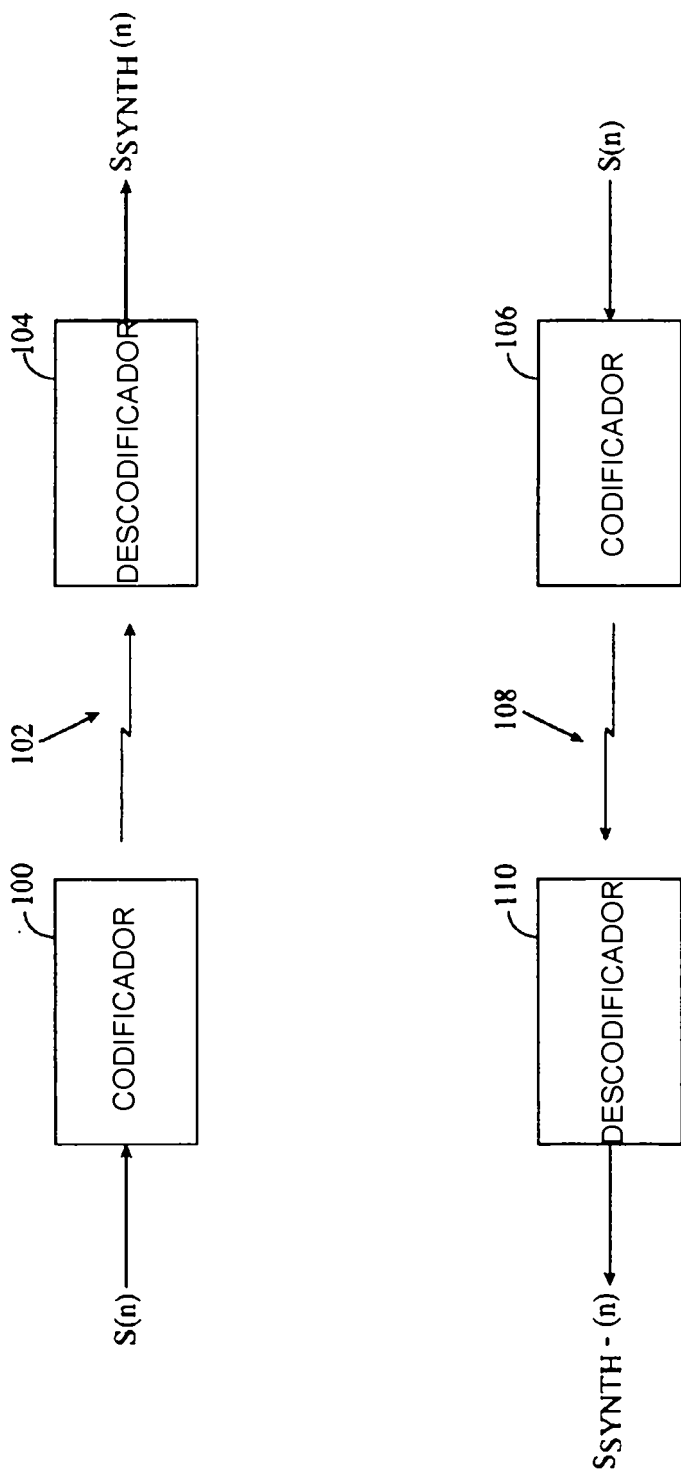


FIG. 18

FIG. 19A

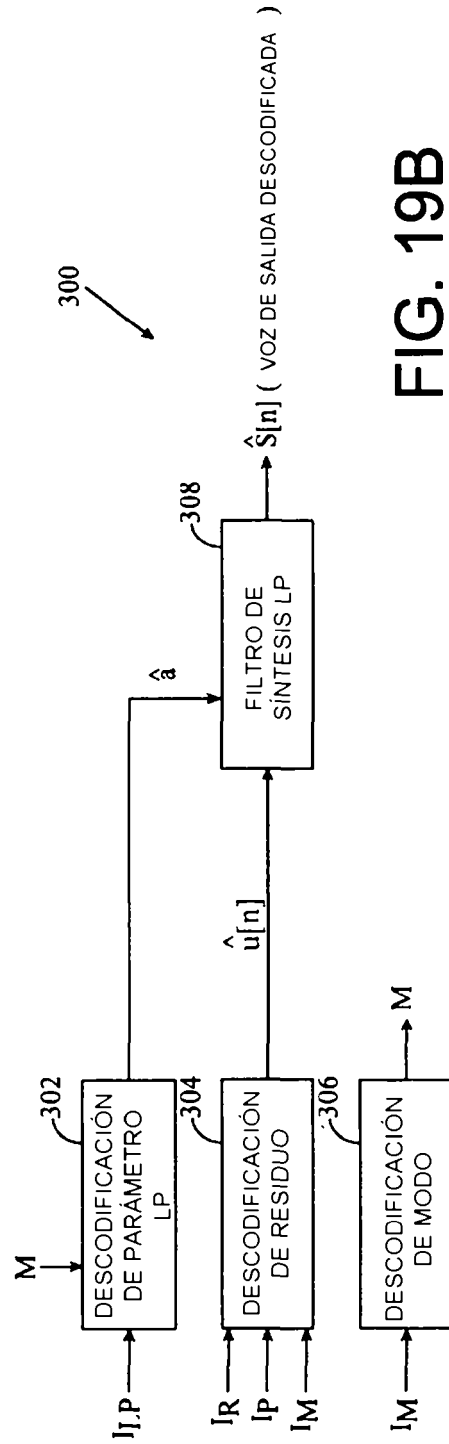
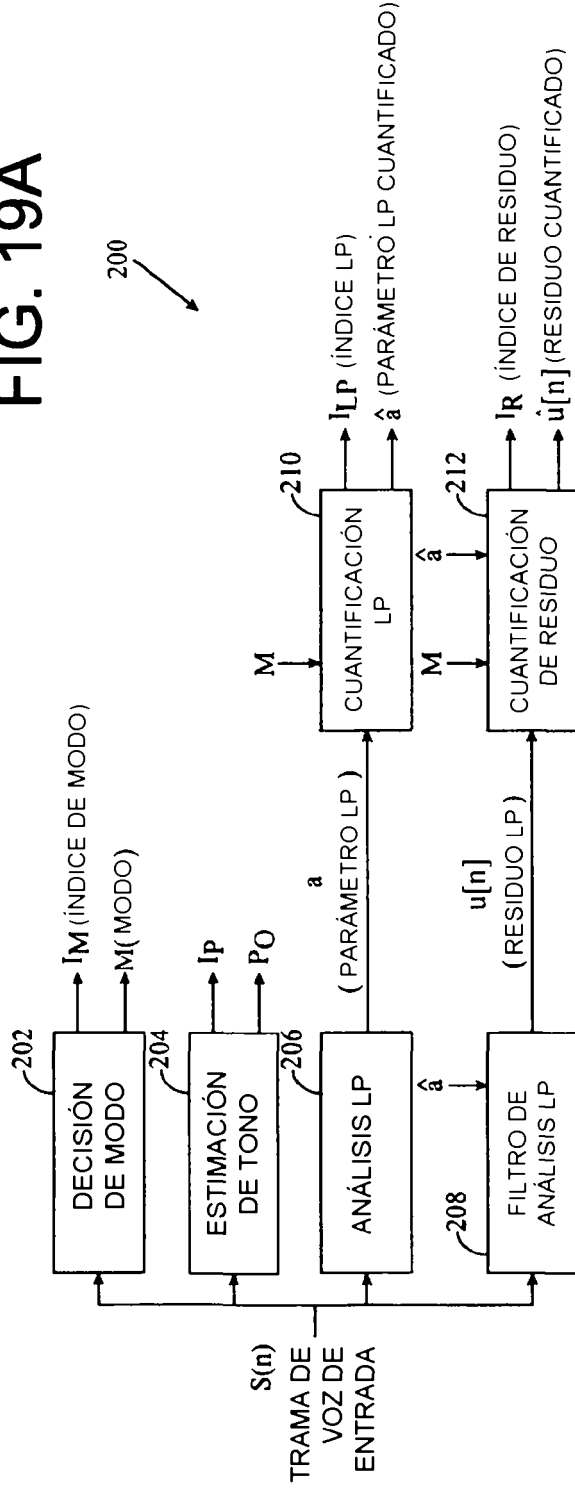
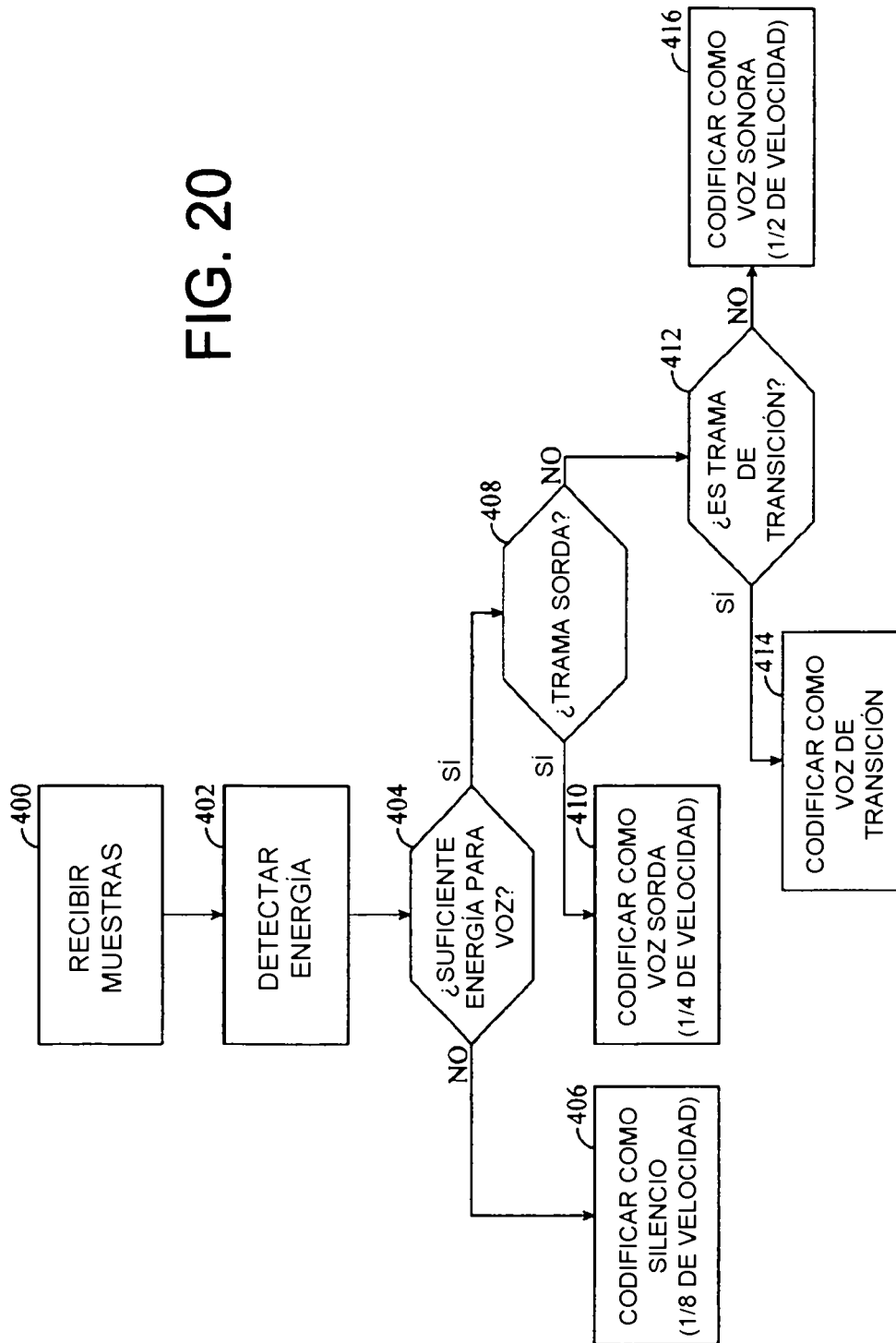


FIG. 19B

FIG. 20



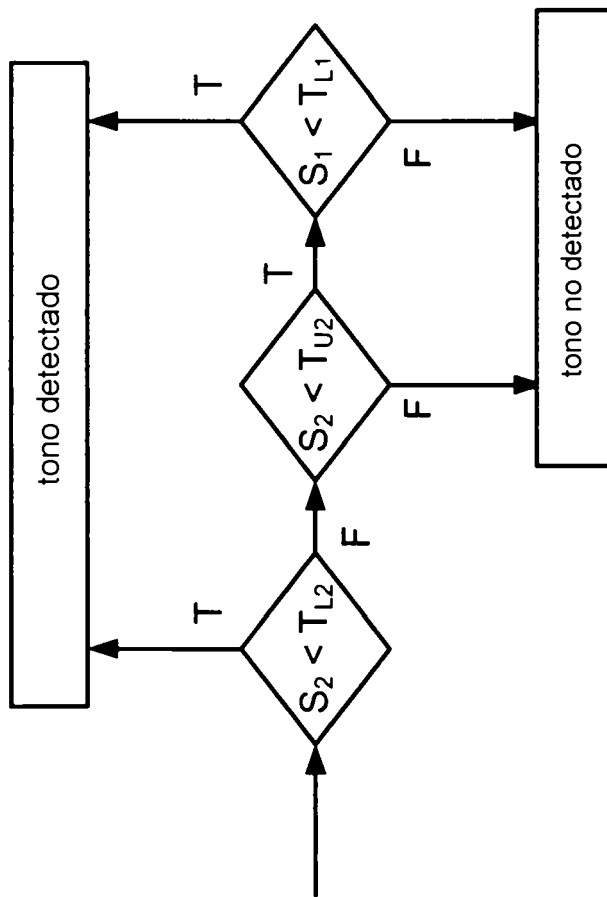


FIG. 21

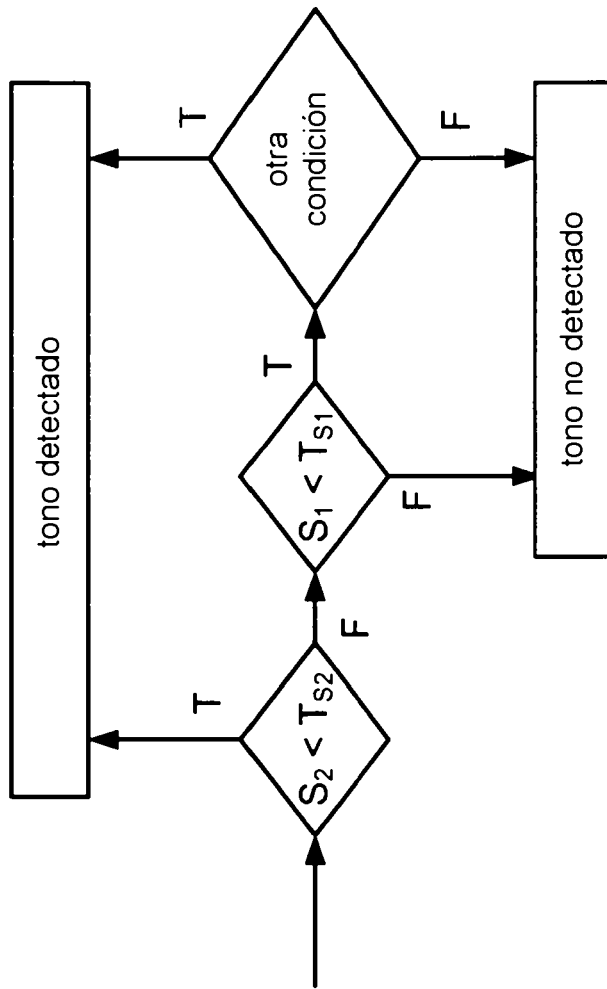


FIG. 22