(19) OFICINA ESPAÑOLA DE PATENTES Y MARCAS

ESPAÑA

(11) Número de publicación: **2 686 349**

(51) Int. Cl.:

G10L 19/26 (2013.01)
G10L 19/18 (2013.01)

(12) TRADUCCIÓN DE PATENTE EUROPEA T3

(86) Fecha de presentación y número de la solicitud internacional: **06.07.2015** **PCT/FR2015/051864**

(87) Fecha y número de publicación internacional: **14.01.2016** **WO16005690**

(96) Fecha de presentación y número de la solicitud europea: **06.07.2015** **E 15742373 (2)**

(97) Fecha y número de publicación de la concesión europea: **06.06.2018** **EP 3167447**

(54) Título: **Actualización de los estados de un postratamiento a una frecuencia de muestreo variable según la trama**

(30) Prioridad:

11.07.2014 FR 1456734

(45) Fecha de publicación y mención en BOPI de la traducción de la patente:
17.10.2018

(73) Titular/es:

ORANGE (100.0%)
78, rue Olivier de Serres
75015 Paris, FR

(72) Inventor/es:

DANIEL, JÉRÔME y
KOVESI, BALAZS

(74) Agente/Representante:

ISERN JARA, Jorge

ES 2 686 349 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

Actualización de los estados de un postratamiento a una frecuencia de muestreo variable según la trama

La presente invención hace referencia al tratamiento de una señal de audiofrecuencia para su transmisión o su almacenamiento. Más particularmente, la invención hace referencia a una actualización de los estados de un postratamiento de una señal de audiofrecuencia decodificada, cuando la frecuencia de muestreo varía de una trama de señal a la otra.

La invención se aplica más particularmente al caso de una decodificación por predicción lineal como la decodificación de tipo CELP (para "Coded Excitation Linear Prediction"). Los códecs por predicción lineal, tales como los códecs de tipo ACELP (para "Algebraic Coded Excitation"), se consideran como adaptados a las señales vocales, de las que modelizan bien la producción.

La frecuencia de muestreo a la que funciona el algoritmo de codificación CELP está predeterminada, en general, y es idéntica en cada trama codificada; unos ejemplos de frecuencias de muestreo son:

- 8 kHz en los codificadores CELP definidos en UIT-T G.729, G.723.1, G.729.1
- 12,8 kHz para la parte CELP de los codificadores 3GPP AMR-WB, UIT-T G.722.2, G.718
- 16 kHz en los codificadores descritos, por ejemplo, en los artículos de G. Roy, P. Kabal, "Wideband CELP speech coding at 16 kbits/sec", ICASSP 1991, y de C. Laflamme et al., 16 kbps "wideband speech coding technique based on algebraic CELP", ICASSP 1991.

Se señalará, por otra parte, que en el caso de un códec tal como se describe en la recomendación UIT-T G.718, está presente un módulo de tratamiento de mejora de la señal decodificada por reducción del ruido de baja frecuencia. Se denomina "bass post-filter" en inglés (BPF) o "posfiltrado de baja frecuencia". Se aplica a la misma frecuencia de muestreo que la decodificación CELP. La finalidad de este postratamiento es eliminar el ruido de baja frecuencia entre los primeros armónicos de una señal vocal sonora. Este postratamiento es importante, sobre todo, para las voces de mujer agudas en que la distancia entre los armónicos es mayor y el ruido está menos enmascarado.

A pesar del hecho de que el nombre habitual de este postratamiento en el campo de la codificación es "posfiltrado de baja frecuencia", no se trata, de hecho, de un sencillo filtrado, sino más bien de un postratamiento bastante complejo que contiene, en general, unos módulos de "rastreo del pitch" ("Pitch Tracking" en inglés), de "refuerzo del pitch" ("Pitch Enhancer" en inglés), de "filtrado de paso bajo " ("Low-pass filtering" o "LP-filtering" en inglés) y unos módulos de adiciones. Este tipo de postratamiento se describe en detalle, por ejemplo, en la recomendación G.718 (06/2008) "Frame error robust narrowband and wideband embedded variable bit-rate coding of speech and audio from 8-32 kbit/s", capítulo 7.14.1. El esquema de bloque de este postratamiento se ilustra en la figura 29 de este mismo documento.

Recordamos solamente en este caso los principios y los elementos necesarios para la comprensión de este presente documento. La técnica descrita utiliza una descomposición en dos bandas de frecuencia, banda baja y banda alta. Se aplica un filtrado adaptativo sobre la banda baja, determinada para referirse a las frecuencias inferiores a los primeros armónicos de la señal sintetizada. Este filtrado adaptativo se parametriza, de este modo, por el periodo *T* de la señal vocal, denominado "pitch". En efecto, las ecuaciones de las operaciones hechas por el módulo de "refuerzo del pitch" son las siguientes: la señal con pitch reforzado $\hat{s}_T(n)$ se obtiene como

$$\hat{s}_T(n) = (1-\alpha)\hat{s}(n) + \alpha s_p(n)$$

donde

$$s_p(n) = 0{,}5\hat{s}(n-T) + 0{,}5\hat{s}(n+T)$$

Y $\hat{s}(n)$ es la señal decodificada.

Este tratamiento requiere una memoria de la señal pasada cuyo tamaño debe cubrir los diferentes valores posibles de pitch T (para conocer el valor ŝ(*n-T)).* El valor del pitch T no se conoce para la trama siguiente, de este modo, en general, para cubrir el peor caso posible, MAXPITCH+1 muestras de la señal decodificada pasada se memorizan para el postratamiento. MAXPITCH da la longitud máxima del pitch a la frecuencia de muestreo dada, por ejemplo, en general, este valor es de 289 a 16 kHz o 231 a 12,8 kHz. A menudo, se memoriza una muestra suplementaria para, a continuación, efectuar un filtrado de desacentuación de orden 1. No se detallará en este caso este filtrado de desacentuación que no es el objeto de la presente invención.

Cuando la frecuencia de muestreo de la señal a la entrada o a la salida del códec no es idéntica a la frecuencia interna de codificación CELP, se implementa un remuestreo. Por ejemplo:

- En los códecs 3GPP AMR-WB, UIT-T G.722.2, la señal de entrada y de salida de banda ancha se muestrea a 16 kHz, pero la codificación CELP funciona a la frecuencia de 12,8 kHz. Se señalará que los códecs UIT-T

G.718 y G.718 Anexo C funcionan, igualmente, con unas frecuencias de entrada/salida de 8 y/o 32 kHz, con un núcleo CELP a 12,8 kHz.

- En el códec UIT-T G.729.1, la señal de entrada es normalmente de banda ancha (a 16 kHz) y la banda baja (0-4 kHz) se obtiene por un banco de filtros de tipo QMF para obtener una señal muestreada a 8 kHz antes de codificación por un algoritmo CELP derivado de los códecs UIT-T G.729 y G.729 Anexo A.

En este caso, se presta interés a una categoría de códecs que soportan al menos dos frecuencias de muestreo internas, pudiendo la frecuencia de muestreo seleccionarse de forma adaptativa en el tiempo y variable de una trama a la otra. El códec MPEG-4 CELP de la norma ISO/IEC 14496-3:2001, susceptible de funcionar a las frecuencias de muestreo de 8 y 16 kHz, es un ejemplo de ello. En general, para una gama de velocidades "bajas", el codificador CELP funcionará a una frecuencia de muestreo menor, por ejemplo fs_1 = 12,8 kHz y para una gama de velocidades más elevadas, el codificador funcionará a una frecuencia superior, por ejemplo, fs_2 = 16 kHz. Un cambio de velocidad en el transcurso del tiempo, de una trama a otra, podrá en este caso ocasionar una conmutación entre estas dos frecuencias (fs_1 y fs_2) según la gama de velocidades cubiertas. Esta conmutación de frecuencias entre dos tramas puede ocasionar unos artefactos audibles y molestos, por varias razones.

Una de las razones que ocasionan estos artefactos es que la conmutación de frecuencias internas de decodificación impide que el posfiltrado de baja frecuencia funcione al menos en la primera trama después de la conmutación, ya que la memoria del postratamiento (es decir, la señal sintetizada pasada) se encuentra a una frecuencia de muestreo diferente de la señal nuevamente sintetizada.

Para remediar este problema, una opción consiste en desactivar el postratamiento sobre la duración de la trama de transición (la trama después del cambio de frecuencia de muestreo interna). Esta opción no produce un resultado deseable, en general, ya que el ruido que era posfiltrado reaparece de manera brusca sobre la trama de transición.

Otra opción es dejar el postratamiento activo, pero poniendo las memorias a cero. Con este método, la calidad obtenida es muy mediocre.

Otra posibilidad es también considerar una memoria a 16 kHz como si fuera a 12,8 kHz conservando solo las últimas 4/5 muestras de esta memoria o de manera inversa, considerar una memoria a 12,8 kHz como si fuera a 16 kHz, ya sea añadiendo 1/5 de ceros al principio (hacia el pasado) de esta memoria para tener la longitud buena, ya sea memorizando un 20 % más de las muestras a 12,8 kHz para tener bastantes de ellas en caso de cambio de frecuencia de muestreo interna. Las pruebas de escucha muestran que estas soluciones no dan una calidad satisfactoria.

Por lo tanto, existe una necesidad de encontrar una solución de mejor calidad para evitar una rotura del postratamiento en caso de cambio de frecuencia de muestreo de una trama a la otra.

La presente invención viene a mejorar la situación.

A tal efecto, propone un procedimiento de actualización de los estados de un postratamiento aplicado a una señal de audiofrecuencia decodificada. El procedimiento es tal que, para una trama corriente de señal decodificada, muestreada a una frecuencia de muestreo diferente de la trama anterior, incluye las etapas siguientes:

- obtención de una señal decodificada pasada, memorizada para la trama anterior;
- remuestreo a la frecuencia de muestreo de la trama corriente, por interpolación, de la señal decodificada pasada obtenida;
- utilización de la señal decodificada pasada remuestreada como memoria del postratamiento de la trama corriente.

De este modo, la memoria del postratamiento está adaptada a la frecuencia de muestreo de la trama corriente que se postrata. Esta técnica permite mejorar la calidad del postratamiento en las tramas de transición entre dos frecuencias de muestreo minimizando al mismo tiempo el aumento de la complejidad (carga de cálculo, memoria ROM, RAM y PROM).

Los diferentes modos particulares de realización mencionados a continuación pueden añadirse de manera independiente o en combinación los unos con los otros, a las etapas del procedimiento de remuestreo definido más arriba.

En un modo particular de realización, en el caso en que la frecuencia de muestreo de la trama anterior es superior a la frecuencia de muestreo de la trama corriente, la interpolación se efectúa partiendo de la muestra más reciente de la señal decodificada pasada e interpolando en el sentido cronológico inverso y en el caso en que la frecuencia de muestreo de la trama anterior es inferior a la frecuencia de muestreo de la trama corriente, la interpolación se efectúa partiendo de la muestra más antigua de la señal decodificada pasada e interpolando en el sentido cronológico.

Este modo de interpolación permite utilizar solo un solo cuadro de almacenamiento (de longitud correspondiente al máximo de periodo de señal para la frecuencia de muestreo mayor) para registrar la señal decodificada pasada antes y después de remuestreo. En efecto, en los dos sentidos de remuestreo, la interpolación está adaptada al hecho de que a partir del momento en que una muestra de la señal pasada se utiliza para una interpolación, ya no se utiliza para la interpolación siguiente. De este modo, puede sustituirse por la interpolada en el cuadro de almacenamiento.

De este modo, en un modo ventajoso de realización, la señal decodificada pasada, remuestreada se memoriza en una misma memoria tampón que la señal decodificada pasada antes de remuestreo.

De este modo, se optimiza la utilización de la memoria RAM del dispositivo que implementa este procedimiento.

En un modo particular de realización, la interpolación es de tipo lineal.

Este tipo de interpolación es de complejidad escasa.

Para una aplicación eficaz, la señal decodificada pasada es de longitud fijada en función de un máximo posible de periodo de señal vocal.

El procedimiento de actualización de los estados está particularmente adaptado al caso en que el postratamiento está aplicado a la señal decodificada sobre una banda de frecuencia baja para disminuir el ruido de baja frecuencia.

La invención hace referencia, igualmente, a un procedimiento de decodificación de una trama corriente de una señal de audiofrecuencia que incluye una etapa de selección de una frecuencia de muestreo de decodificación, una etapa de postratamiento. El procedimiento es tal que, en el caso en que la trama anterior se muestrea a una primera frecuencia de muestreo diferente de una segunda frecuencia de muestreo de la trama corriente, incluye una actualización de los estados del postratamiento según un procedimiento tal como se describe.

El tratamiento de baja frecuencia de la señal decodificada está adaptado, por lo tanto, a la frecuencia de muestreo interna del decodificador, estando, entonces, la calidad de este postratamiento mejorada.

La invención hace referencia a un dispositivo de tratamiento de una señal de audiofrecuencia decodificada, caracterizado por que incluye, para una trama corriente de señal decodificada, muestreada a una frecuencia de muestreo diferente de la trama anterior:

- un módulo de obtención de una señal decodificada pasada, memorizada para la trama anterior;
- un módulo de remuestreo para remuestrear a la frecuencia de muestreo de la trama corriente, por interpolación, la señal decodificada pasada obtenida;
- un módulo de postratamiento que utiliza la señal decodificada pasada remuestreada como memoria del postratamiento de la trama corriente.

Este dispositivo presenta las mismas ventajas que el procedimiento descrito anteriormente, que implementa.

La presente invención tiene como propósito, igualmente, un decodificador de señal de audiofrecuencia que incluye un módulo de selección de una frecuencia de muestreo de decodificación y al menos un dispositivo de tratamiento tal como se describe.

La invención tiene como propósito un programa informático que incluye unas instrucciones de código para la implementación de las etapas del procedimiento de actualización de los estados tal como se describe, cuando estas instrucciones se ejecutan por un procesador.

Finalmente, la invención hace referencia a un soporte de almacenamiento, legible por un procesador, integrado o no en el dispositivo de tratamiento, eventualmente amovible, que memoriza un programa informático que implementa un procedimiento de actualización de los estados tal como se ha descrito anteriormente.

Otras características y ventajas de la invención se pondrán de manifiesto de manera más clara con la lectura de la siguiente descripción, dada únicamente a título de ejemplo no limitativo y hecha con referencia a los dibujos adjuntos, en los que:

- la figura 1 ilustra en forma de organigrama un procedimiento de actualización de los estados de un postratamiento según un modo de realización de la invención;
- la figura 2 ilustra un ejemplo de remuestreo de 16 kHz a 12,8 kHz, según un modo de realización de la invención;
- la figura 3 ilustra un ejemplo de remuestreo de, 12,8 kHz a 16 kHz según un modo de realización de la invención;
- la figura 4 ilustra un ejemplo de decodificador que incluye unos módulos de decodificación que funcionan a unas frecuencias de muestreo diferentes y un dispositivo de tratamiento según un modo de realización de la invención; y

- la figura 5 ilustra una representación material de un dispositivo de tratamiento según un modo de realización de la invención.

La figura 1 ilustra en forma de organigrama las etapas implementadas en el procedimiento de actualización de los estados de un postratamiento según un modo de realización de la invención.

En este caso, se coloca uno en el caso en que la trama que antecede a la trama corriente que hay que tratar está a una primera frecuencia de muestreo fs_1, mientras que la trama corriente está a una segunda frecuencia de muestreo fs_2. En otros términos, en una aplicación relacionada con la decodificación, el procedimiento según un modo de realización de la invención, se aplica cuando la frecuencia interna de decodificación CELP en la trama corriente (fs_2) es diferente de la frecuencia interna de decodificación CELP de la trama anterior (*fs1*): $fs1 \neq fs_2$

En el modo de realización descrito en este caso, el codificador o decodificador CELP tiene dos frecuencias de muestreo internas: 12,8 kHz para las velocidades bajas y 16 kHz para las velocidades altas. Muy evidentemente, pueden estar previstas otras frecuencias de muestreo internas en el marco de la invención.

El procedimiento de actualización de los estados de postratamiento implementado sobre una señal de audiofrecuencia decodificada incluye una primera etapa E101 de recuperación en una memoria tampón también llamada buffer, de una señal decodificada pasada, memorizada durante la decodificación de la trama anterior. Como se ha mencionado anteriormente, esta señal decodificada de la trama anterior (Mem. fs_1) está a una primera frecuencia de muestreo interna fs_1.

La longitud de señal decodificada memorizada es función, por ejemplo, del valor máximo del periodo (o "pitch") de señal vocal.

Por ejemplo, a 16 kHz de frecuencia de muestreo el valor máximo del pitch codificado es de 289. La longitud de la señal decodificada memorizada es, entonces, *len_mem_16*=290 muestras.

Para una frecuencia interna a 12,8 kHz la señal decodificada memorizada tiene una longitud de *len_mem_12*=*(290/5)**4 = 232 muestras.

Para optimizar la memoria RAM se utiliza en este caso el mismo buffer de memoria de 290 muestras para los dos casos, a 16 kHz son necesarios todos los índices de 0 a 289, a 12,8 kHz solo son útiles los índices de 58 a 289. La última muestra de la memoria (del índice 289) contiene, por lo tanto, siempre la última muestra de la señal decodificada pasada, de manera independiente de la frecuencia de muestreo. Hay que señalar que a las dos frecuencias de muestreo (12,8 kHz o 16 kHz), la memoria cubre el mismo soporte temporal, 18,125 ms.

Hay que señalar, igualmente, que a 12,8 kHz es posible, igualmente, utilizar los índices de 0 a 231 e ignorar las muestras de 232 a 289. Son posibles, igualmente, unas posiciones intermedias, pero desde un punto de vista de programación estas soluciones no son prácticas. En la aplicación preferente de la invención, se utiliza la primera solución (índices de 58 a 289).

En la etapa E102, esta señal decodificada pasada se remuestrea a la frecuencia de muestreo interna de la trama corriente fs2. Este remuestreo se efectúa, por ejemplo, por un método de interpolación lineal de complejidad escasa. Podrán utilizarse otros tipos de interpolaciones, como la interpolación cúbica o de los "splines", por ejemplo.

En un modo particular ventajoso, la interpolación utilizada permite utilizar solo un solo cuadro de almacenamiento RAM (una sola memoria tampón).

El caso de un cambio de la frecuencia de muestreo interna de 16 kHz hacia 12,8 kHz se ilustra en la figura 2. Las longitudes representadas son reducidas, en este caso, para simplificar la descripción. En esta figura, la longitud de la memoria marcada "mem" es de *len_mem_16*=20 muestras a 16 kHz (marcadores cuadrados rellenos) y de *len_mem_12*=16 muestras a 12,8 kHz (marcadores redondos rellenos). El círculo vacío a 12,8 kHz a la derecha representa el inicio de la señal decodificada de la trama corriente. Las flechas en punteado dan para cada muestra de salida a 12.8 kHz las muestras de entrada a 16 kHz a partir de las cuales se interpolan en el caso de una interpolación lineal.

La figura ilustra, igualmente, cómo estas señales están almacenadas en la memoria tampón. En la parte a.), las muestras memorizadas a 12,8 kHz están alineadas con el final del buffer "mem" (según la aplicación preferente). Las cifras dan el índice de la ubicación en el cuadro de almacenamiento. Los marcadores redondos vacíos en punteado del índice 0 a 3 corresponden a las ubicaciones no utilizadas a 12,8 kHz.

Se puede observar que procediendo a la interpolación partiendo de la muestra más reciente (por lo tanto, la del índice 19 en la figura) e interpolando en el sentido cronológico inverso, se puede reescribir el resultado en el mismo cuadro, ya que el valor antiguo de esta ubicación ya no sirve para la interpolación siguiente. La flecha rellena

muestra el sentido de la interpolación, los números escritos en la flecha corresponden al orden en el que las muestras de salida se interpolan.

Se ve, igualmente, que los pesos de interpolación se repiten de manera periódica, por pasos de 5 muestras de entrada, esto es, 4 muestras de salida. De este modo, en un modo de realización particular, se puede proceder a la interpolación por bloques de 5 muestras de entrada y 4 muestras de salida. De este modo, hay *nb_bloc* = *len_mem_16*/5 = *len_mem_12* / 4 bloques que hay que tratar.

A título ilustrativo, un ejemplo de instrucciones de código de estilo lenguaje C, para efectuar esta interpolación, se da en Anexo 1,
donde *pf5* es un apuntador de cuadro (direccionamiento) para la señal de entrada a 16 kHz, *pf4* es un apuntador de cuadro para la señal de salida a 12,8 kHz. Al inicio, los dos apuntan al mismo lugar, al final del cuadro *mem* de longitud *len_mem_16* (los índices utilizados son de 0 a *len_mem_16*-1*). nb_bloc* contiene el número de bloques que hay que tratar en el bucle *for. pf4[0]* es el valor de cuadro apuntado por el apuntador *pf4, pf4[-1]* es el valor anterior y sucesivamente de este modo. Ocurre lo mismo para *pf5.* Al final de cada iteración, los apuntadores *pf5* y *pf4* retroceden por pasos de 5 y 4 muestras respectivamente.

Con esta solución el aumento de la complejidad (número de operaciones, PROM, ROM) es muy escaso y la asignación de un nuevo cuadro RAM no es necesaria.

La parte b.) de la figura 2 ilustra el caso en que las muestras a 12,8 kHz están alineadas con el inicio del buffer "mem" y las ubicaciones del índice 16 a 19 no se utilizan. En ese caso, como ilustra la flecha rellena, hay que proceder a la interpolación partiendo de la muestra más antigua para poder reescribir el resultado en el mismo cuadro.

De la misma forma, la figura 3 ilustra el caso de cambio de la frecuencia de muestreo interna de 12,8 kHz hacia 16 kHz, siempre con unas longitudes reducidas para simplificar la descripción: *len_mem_16*=20 muestras a 16 kHz (marcadores cuadrados rellenos) y *len_mem_12*= 16 muestras a 12,8 kHz (marcadores redondos rellenos). El cuadrado vacío a 16 kHz representa el inicio de la señal decodificada de la trama corriente. Hay que señalar que la primera muestra de la trama corriente a 16 kHz es idéntica a la de a 12,8 kHz (mismo momento temporal), esto está representado por un círculo vacío. Las flechas en punteado dan para cada muestra de salida a 16 kHz las muestras de entrada a 12,8 kHz a partir de las cuales se interpolan en el caso de una interpolación lineal. Para la interpolación de la última muestra de salida se debe utilizar, igualmente, la primera muestra de la trama corriente a 12,8 kHz, que se conoce bien como se ha explicado anteriormente. Esta dependencia se ilustra por una flecha discontinua en la figura 3.

La figura también muestra cómo estas señales están almacenadas en la memoria tampón, las cifras dan el índice de la ubicación en el cuadro. En la parte a.) las muestras memorizadas a 12,8 kHz están alineadas con el final del buffer "mem" (según la aplicación preferente). Los marcadores redondos vacíos en punteado del índice 0 a 3 corresponden a las ubicaciones no disponible (ya que no se utiliza) a 12,8 kHz

Se puede observar que esta vez, la interpolación se efectúa partiendo de la muestra más antigua (por lo tanto, la de índice 0 a la salida) para poder reescribir el resultado de la interpolación en el mismo cuadro de memoria, ya que el valor antiguo en estas ubicaciones no sirve para efectuar las interpolaciones siguientes. La flecha rellena muestra el sentido de la interpolación, los números escritos en la flecha corresponden al orden en el que las muestras de salida se interpolan.

Se ve, igualmente, que el peso de interpolación se repite de manera periódica por pasos de 4 muestras de entrada, esto es, 5 muestras de salida. De este modo, es ventajoso proceder a la interpolación por bloques de 4 muestras de entrada y 5 muestras de salida. Por lo tanto, hay siempre *nb_bloc* = len_mem_16/5 = *len_mem_12* / 4 bloques que hay que tratar, salvo que esta vez, el último bloque es particular, ya que utiliza, igualmente, el primer valor de la trama corriente. Es interesante, igualmente, observar que el índice de la primera muestra a 12,8 kHz en la memoria "mem" (4 en la figura 3) es igual al número de bloques que hay que tratar, *nb_bloc*, ya que entre las 2 frecuencias hay una muestra de desfase por bloque.

A título ilustrativo, un ejemplo de instrucciones de código de estilo lenguaje C se da en Anexo 2 para efectuar esta interpolación:

El último bloque se trata de manera separada, ya que depende, igualmente, de la primera muestra de la trama corriente anotada *syn[0].*

Por analogía con el caso anterior, *pf4* es un apuntador de cuadro para la señal de entrada a 12,8 kHz que apunta sobre el inicio de la memoria del filtro, esta memoria está almacenada a partir del *nb_bloc°* muestra del cuadro *mem. pf5* es un apuntador de cuadro para la señal de salida a 16 kHz, apunta sobre el primer elemento del cuadro *mem. nb_bloc* contiene el número de bloques que hay que tratar. *nb_bloc-1* bloques se tratan en el bucle *for,* luego se trata el último bloque de manera separada. *pf4[0]* es el valor de cuadro apuntado por el apuntador *pf4, pf4[1]* es el valor

siguiente y sucesivamente de este modo. Ocurre lo mismo para *pf5.* Al final de cada iteración, los apuntadores *pf5* y *pf4* avanzan por pasos de 5 y 4 muestras respectivamente. La señal decodificada de la trama corriente se almacena en el cuadro *syn, syn[0]* es la primera muestra de la trama corriente.

Con esta solución el aumento de la complejidad (número de operaciones, PROM, ROM) es muy escaso y la asignación de un nuevo cuadro RAM no es necesaria.

La parte b.) de la figura 3 ilustra el caso en que las muestras a 12,8 kHz están alineadas con el inicio del buffer "mem" y las ubicaciones del índice 16 a 19 no se utilizan. En ese caso, como ilustra la flecha rellena, hay que proceder a la interpolación partiendo de la muestra más reciente para poder reescribir el resultado en el mismo cuadro.

En este momento, se vuelve a la figura 1. Después de la etapa E102 de remuestreo de la memoria Mem. fs_1. a la frecuencia fs_2 se obtiene la memoria o señal decodificada pasada, remuestreada (Mem. fs_2). Esta señal decodificada pasada remuestreada se utiliza en la etapa E103 como nueva memoria del postratamiento de la trama corriente.

En un modo de realización particular, el postratamiento es similar al descrito en la recomendación UIT-T G.718. La memoria de la señal decodificada pasada remuestreada se utiliza, en este caso, para conocer los valores $\hat{s}(n-T)$ para n = 0...T-1 tal como se ha descrito anteriormente como recordatorio de la técnica "bass-post-filter" de G.718.

La figura 4 describe en este momento un ejemplo de decodificador que incluye un dispositivo de tratamiento 410 en un modo de realización de la invención. La señal de salida *y(n)* (mono), se muestrea a la frecuencia fs^{out} que puede tomar los valores de 8, 16, 32 o 48 kHz.

Para cada trama recibida, el tren binario se demultiplexa en 401 y se decodifica. El decodificador determina en 402, en este caso, en función de la velocidad de la trama corriente, a qué frecuencia fs_1 o fs_2 decodificar las informaciones procedentes de un codificador CELP. En función de la frecuencia de muestreo, ya sea el módulo de decodificación 403 para la frecuencia fs_1 ya sea el módulo de decodificación 404 para la frecuencia fs_2 se implementa para decodificar la señal recibida.

El decodificador CELP que funciona a la frecuencia fs_1 = 12,8 kHz (bloque 403) es una extensión multivelocidades del algoritmo de decodificación UIT-T G.718 inicialmente definido entre 8 y 32 kbit/s. En particular, comprende la decodificación de la excitación CELP y un filtrado de síntesis de predicción lineal $1/\hat{A}(z)$.

El decodificador CELP que funciona a la frecuencia fs_2 = 16 kHz (bloque 404) es una extensión multivelocidades a 16 kHz del algoritmo de decodificación UIT-T G.718 inicialmente definido a entre 8 y 32 kbit/s y a 12,8 kHz.

La implementación de la decodificación CELP a 16 kHz no está detallada en este caso, ya que rebasa el marco de la invención.

En este caso, no se presta interés al problema de la actualización de los estados de decodificador CELP durante una conmutación de la frecuencia fs_1 a la frecuencia fs_2.

La salida del decodificador CELP en la trama corriente se posfiltra, a continuación, por el dispositivo de tratamiento 410 que implementa el procedimiento de actualización de los estados de postratamiento descrito con referencia a la figura 1. Este dispositivo incluye unos módulos de postratamiento 420 y 421 adaptados a las frecuencias de muestreo respectivas fs_1 y fs_2 adecuadas para efectuar un postratamiento de tipo disminución del ruido de baja frecuencia también llamado posfiltrado de baja frecuencia, de forma similar al "bass post-filter" (BPF) del códec UIT-T G.718, utilizando las memorias de postratamiento remuestreadas por el módulo de remuestreo 422. En efecto, el dispositivo de tratamiento incluye, igualmente, un módulo de remuestreo 422 que efectúa el remuestreo de una señal decodificada pasada, memorizada para la trama anterior, por interpolación. De este modo, la señal decodificada pasada de la trama anterior (Mem. fs_1), muestreada a la frecuencia fs_1 se remuestrea a la frecuencia fs_2 para obtener una señal decodificada pasada remuestreada (Mem. fs_2) utilizada como memoria del postratamiento de la trama corriente.

De manera inversa, la señal decodificada pasada de la trama anterior (Mem. Fs_2), muestreada a la frecuencia fs_2 se remuestrea a la frecuencia fs_1 para obtener una señal decodificada pasada remuestreada (Mem. fs_1) utilizada como memoria del postratamiento de la trama corriente.

La señal postratada por el dispositivo de tratamiento 410 se remuestrea, a continuación, a la frecuencia de salida fs^{out} por los módulos de remuestreo 411 y 412, con, por ejemplo, fs^{out}=32 kHz. Esto equivale a efectuar ya sea un remuestreo de fs_1 a fs^{out} en 411, ya sea un remuestreo de fs_2 *a* fs^{out} en 412.

En unas variantes, otras operaciones de postratamiento (filtrado de paso alto, etc.) podrán realizarse como complemento o como sustitución de los bloques 420 y 421.

Según la frecuencia de salida fs^{out}, una señal de banda alta (remuestreada a la frecuencia fs^{out}) decodificada por el módulo de decodificación 405 puede añadirse en 406 a la señal de banda baja remuestreada.

El decodificador prevé, igualmente, la utilización de modos suplementarios de decodificación, tal como una decodificación por transformada de frecuencia inversa (bloque 430) en el caso en que la señal de entrada que hay que codificar se ha codificado por un codificador por transformada. En efecto, el codificador analiza el tipo de señal que hay que codificar y elige la técnica de codificación mejor adaptada a esta señal. La codificación por transformada se utiliza, sobre todo, para las señales musicales que están generalmente mal codificadas por un codificador predictivo de tipo CELP.

La figura 5 representa un ejemplo de realización material de un dispositivo de tratamiento 500 según un modo de realización de la invención. Este puede formar parte integrante de un decodificador de señal de audiofrecuencia o de un equipo que recibe unas señales de audiofrecuencias. Puede estar integrado en un terminal de comunicación, un decodificador de salón de tipo set-top box o también de una pasarela doméstica.

Este tipo de dispositivo incluye un procesador PROC 506 que coopera con un bloque de memoria BM que incluye una memoria de almacenamiento y/o de trabajo MEM.

Un dispositivo de este tipo incluye un módulo de entrada 501 adecuado para recibir unas tramas de señal de audio y, en concreto, una parte memorizada (Buf_{prec}) de una trama anterior a una primera frecuencia de muestreo fs_1.

Incluye un módulo de salida 502 adecuado para transmitir una trama corriente de señal de audio frecuencia postratada s'(n).

El procesador PROC controla el módulo de obtención 503 de una señal decodificada pasada, memorizada para la trama anterior. Típicamente, la obtención de esta señal decodificada pasada se efectúa por sencilla lectura en una memoria de tipo memoria tampón, comprendida en el bloque de memoria BM. El procesador controla, igualmente, un módulo de remuestreo 504 para remuestrear por interpolación la señal decodificada pasada obtenida en 503.

También controla un módulo de postratamiento 505 que utiliza la señal decodificada pasada remuestreada como memoria de postratamiento para efectuar un postratamiento de la trama corriente.

El bloque de memoria puede incluir, ventajosamente, un programa informático que incluye unas instrucciones de código para la implementación de las etapas del procedimiento de actualización de los estados de postratamiento en el sentido de la invención, cuando estas instrucciones se ejecutan por el procesador PROC y, en concreto, las etapas de obtención de una señal decodificada pasada, memorizada para la trama anterior, de remuestreo por interpolación de la señal decodificada pasada obtenida y de utilización de la señal decodificada pasada remuestreada como memoria del postratamiento de la trama corriente.

Típicamente, la descripción de la figura 1 retoma las etapas de un algoritmo de un programa informático de este tipo. El programa informático puede estar almacenado, igualmente, sobre un soporte de memoria legible por un lector del dispositivo o descargable en el espacio de memoria de este.

La memoria MEM registra de manera general, todos los datos necesarios para la implementación del procedimiento.

ANEXO 1:

```
pf4 = &mem[len_mem_16-1];
      pf5 = pf4;
      nb_bloc =  len_mem_16 / 5;
      for (c=0; c<nb_bloc; c++)
      {
            pf4[0] = 0.75f * pf5[0] + 0.25f * pf5[-1];
            pf4[-1] = 0.50f * pf5[-1] + 0.50f * pf5[-2];
            pf4[-2] = 0.25f * pf5[-2] + 0.75f * pf5[-3];
            pf4[-3] = pf5[-4];
            pf5 -= 5;
            pf4 -= 4;
      }
```

ANEXO 2:

```
nb_bloc =  len_mem_16 / 5;
pf4 = & mem[nb_bloc];
pf5 = &mem[0];
for (c=0; c< nb_bloc-1; c++)
{
      pf5[0] = pf4[0];
      pf5[1] = 0.2f * pf4[0] + 0.8f * pf4[1];
      pf5[2] = 0.4f * pf4[1] + 0.6f * pf4[2];
      pf5[3] = 0.6f * pf4[2] + 0.4f * pf4[3];
      pf5[4] = 0.8f * pf4[3] + 0.2f * pf4[4];
      pf4 += 4;
      pf5 += 5;
}
pf5[0] = pf4[0];
pf5[1] = 0.2f * pf4[0] + 0.8f * pf4[1];
pf5[2] = 0.4f * pf4[1] + 0.6f * pf4[2];
pf5[3] = 0.6f * pf4[2] + 0.4f * pf4[3];
pf5[4] = 0.8f * pf4[3] + 0.2f * syn[0];
```

REIVINDICACIONES

1. Procedimiento de actualización de los estados de un postratamiento aplicado a una señal de audiofrecuencia decodificada, caracterizado por que, para una trama corriente de señal decodificada, muestreada a una frecuencia de muestreo diferente de la trama anterior, el procedimiento incluye las etapas siguientes:

- obtención (E101) de una señal decodificada pasada, memorizada para la trama anterior;
- remuestreo (E102) a la frecuencia de muestreo de la trama corriente, por interpolación, de la señal decodificada pasada obtenida;
- utilización de la señal decodificada pasada remuestreada como memoria (E103) del postratamiento de la trama corriente.

2. Procedimiento según la reivindicación 1, caracterizado por que, en el caso en que la frecuencia de muestreo de la trama anterior es superior a la frecuencia de muestreo de la trama corriente, la interpolación se efectúa partiendo de la muestra más reciente de la señal decodificada pasada e interpolando en el sentido cronológico inverso y en el caso en que la frecuencia de muestreo de la trama anterior es inferior a la frecuencia de muestreo de la trama corriente, la interpolación se efectúa partiendo de la muestra más antigua de la señal decodificada pasada e interpolando en el sentido cronológico.

3. Procedimiento según la reivindicación 1 o 2, caracterizado por que la señal decodificada pasada, remuestreada se memoriza en una misma memoria tampón que la señal decodificada pasada antes de remuestreo.

4. Procedimiento según una de las reivindicaciones 1 a 3, caracterizado por que la interpolación es de tipo lineal.

5. Procedimiento según la reivindicación 1, caracterizado por que la señal decodificada pasada es de longitud fijada en función de un máximo posible de periodo de señal vocal.

6. Procedimiento según una de las reivindicaciones anteriores, caracterizado por que el postratamiento se aplica a la señal decodificada sobre una banda de frecuencia baja para disminuir el ruido de baja frecuencia.

7. Procedimiento de decodificación de una trama corriente de una señal de audiofrecuencia que incluye una etapa de selección de una frecuencia de muestreo de decodificación, una etapa de postratamiento, caracterizado por que, en el caso en que la trama anterior se muestrea a una primera frecuencia de muestreo diferente de una segunda frecuencia de muestreo de la trama corriente, el procedimiento incluye una actualización de los estados del postratamiento según un procedimiento conforme a una de las reivindicaciones 1 a 6.

8. Dispositivo de tratamiento de una señal de audiofrecuencia decodificada, caracterizado por que incluye, para una trama corriente de señal decodificada, muestreada a una frecuencia de muestreo diferente de la trama anterior:

- un módulo de obtención (422, 503) de una señal decodificada pasada, memorizada para la trama anterior;
- un módulo de remuestreo (422, 504) para remuestrear a la frecuencia de muestreo de la trama corriente, por interpolación, la señal decodificada pasada obtenida;
- un módulo de postratamiento (420, 421, 505) que utiliza la señal decodificada pasada remuestreada como memoria del postratamiento de la trama corriente.

9. Decodificador de señal de audiofrecuencia caracterizado por que incluye un módulo de selección de una frecuencia de muestreo de decodificación y un dispositivo de tratamiento conforme a la reivindicación 8.

10. Programa informático que incluye unas instrucciones de código para la implementación de las etapas del procedimiento de actualización de los estados según una de las reivindicaciones 1 a 6, cuando estas instrucciones se ejecutan por un procesador.

11. Soporte de almacenamiento legible por un procesador, sobre el que está almacenado un programa informático que comprende unas instrucciones de código para la ejecución de las etapas del procedimiento de actualización de los estados según una de las reivindicaciones 1 a 6.

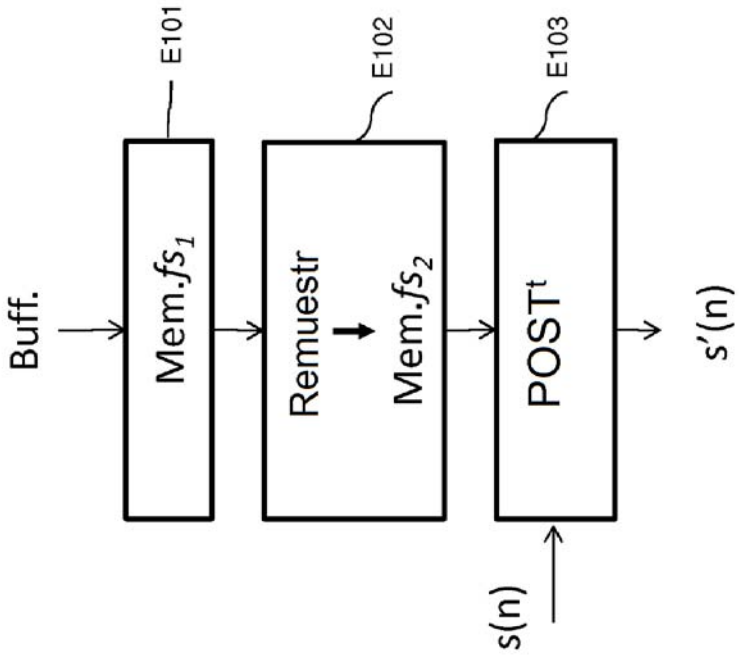

Fig.1

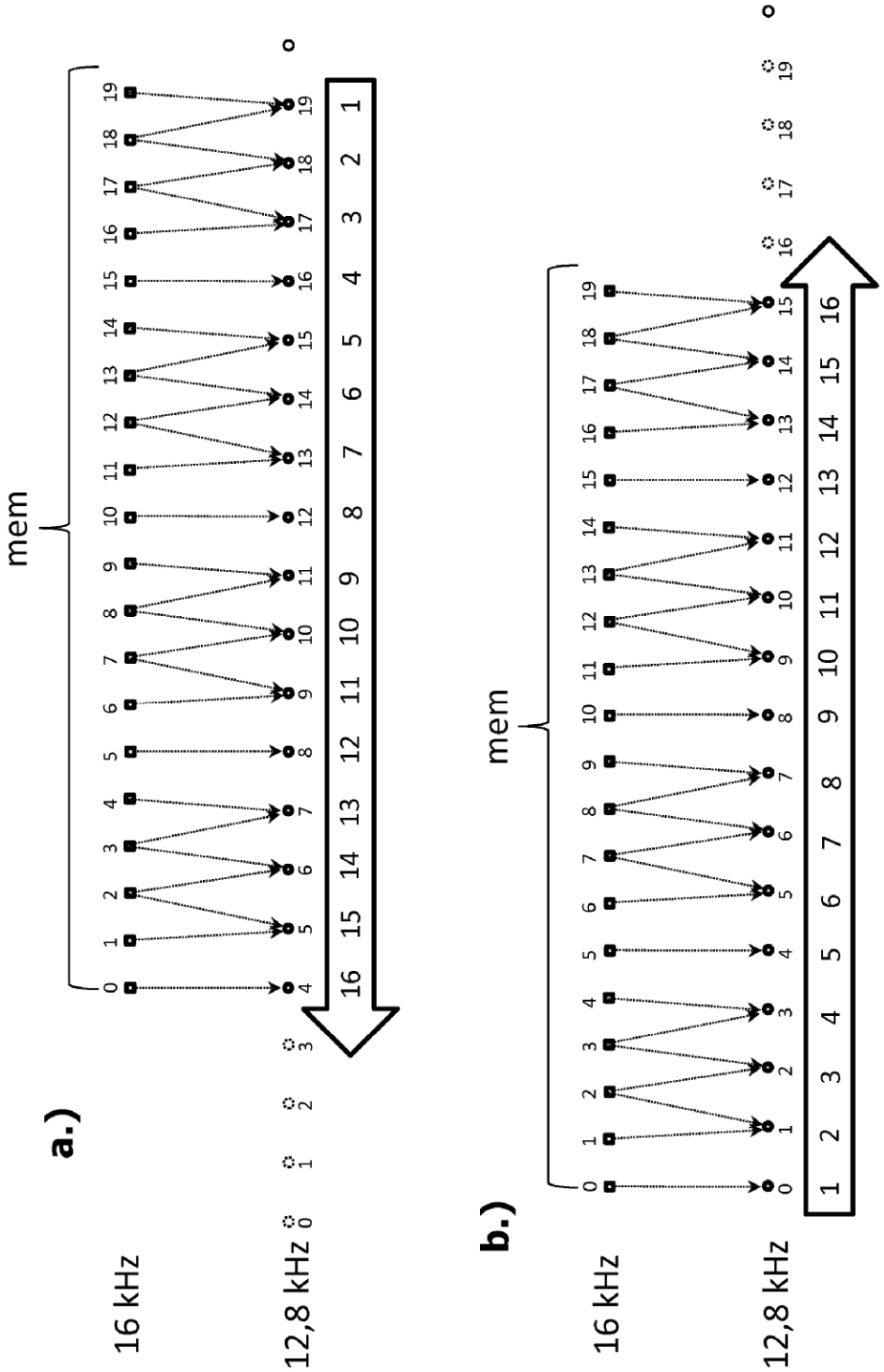

Fig.2

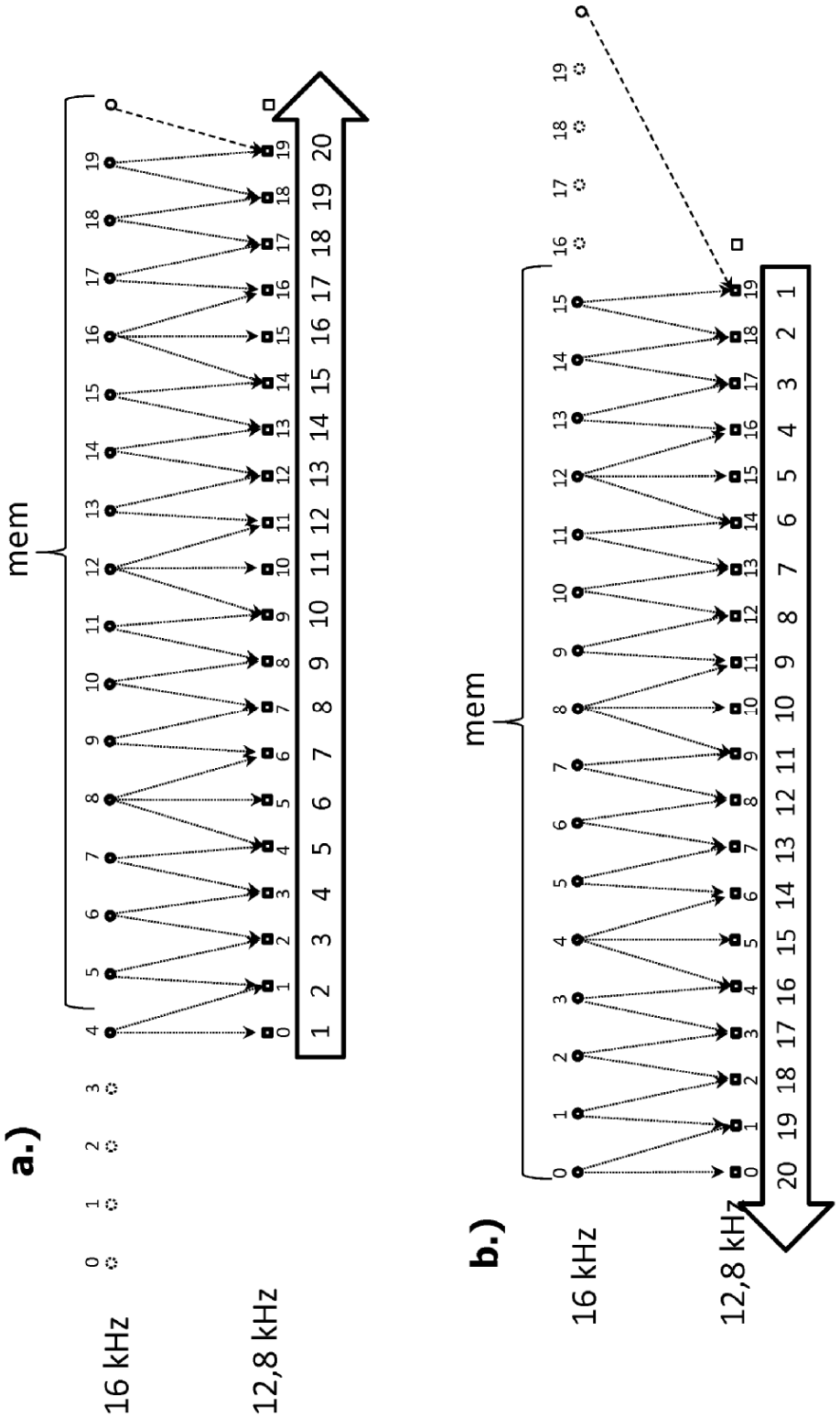

Fig.3

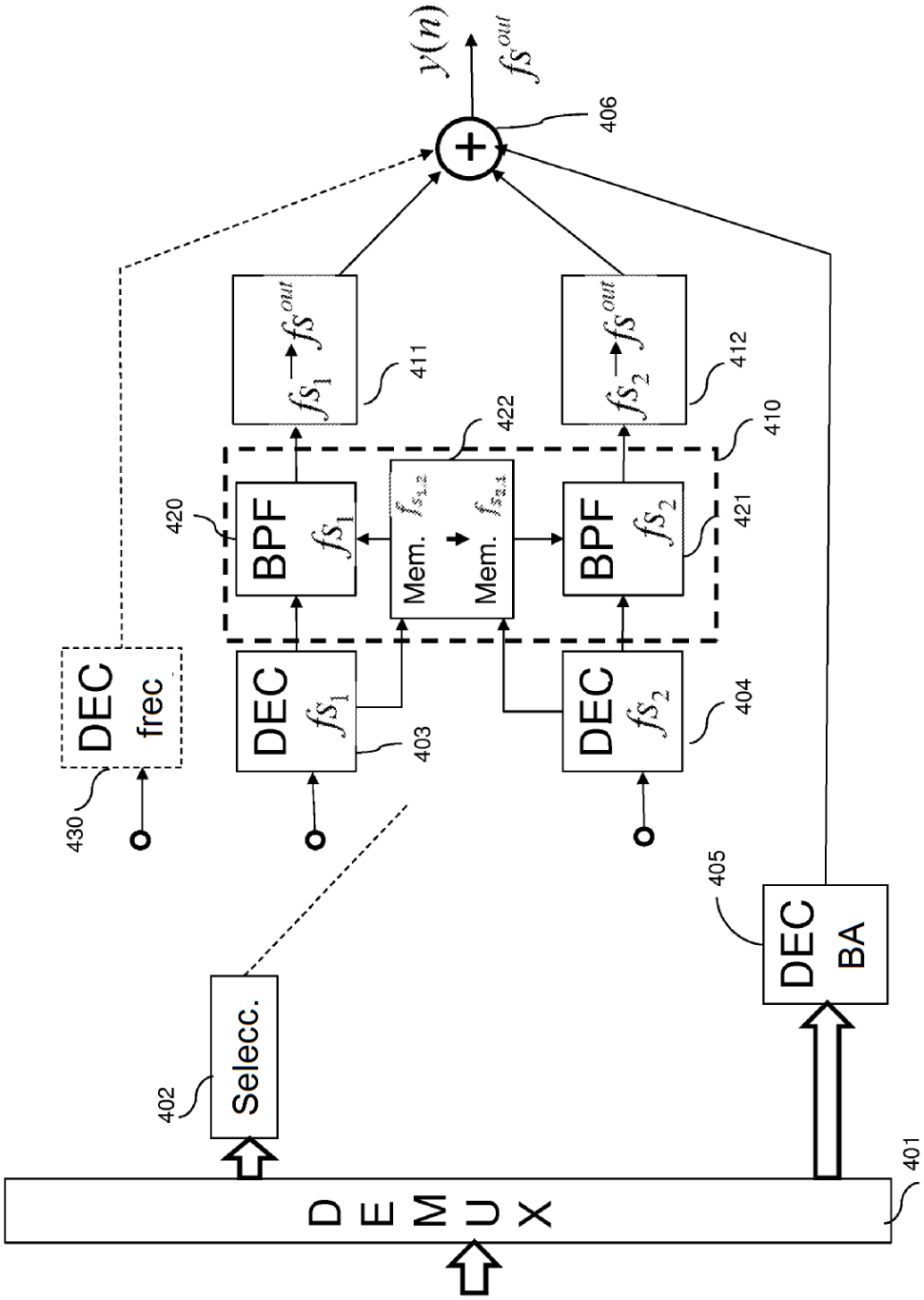
DEMUX
401
Selecc.
402
DEC
BA
405
DEC
fs_1
403
DEC
fs_2
404
DEC
frec
430
BPF
fs_1
420
Mem.
Mem.
422
BPF
fs_2
421
410
$fs_1 \rightarrow fs^{out}$
411
$fs_2 \rightarrow fs^{out}$
412
406
$y(n)$
fs^{out}

Fig.4

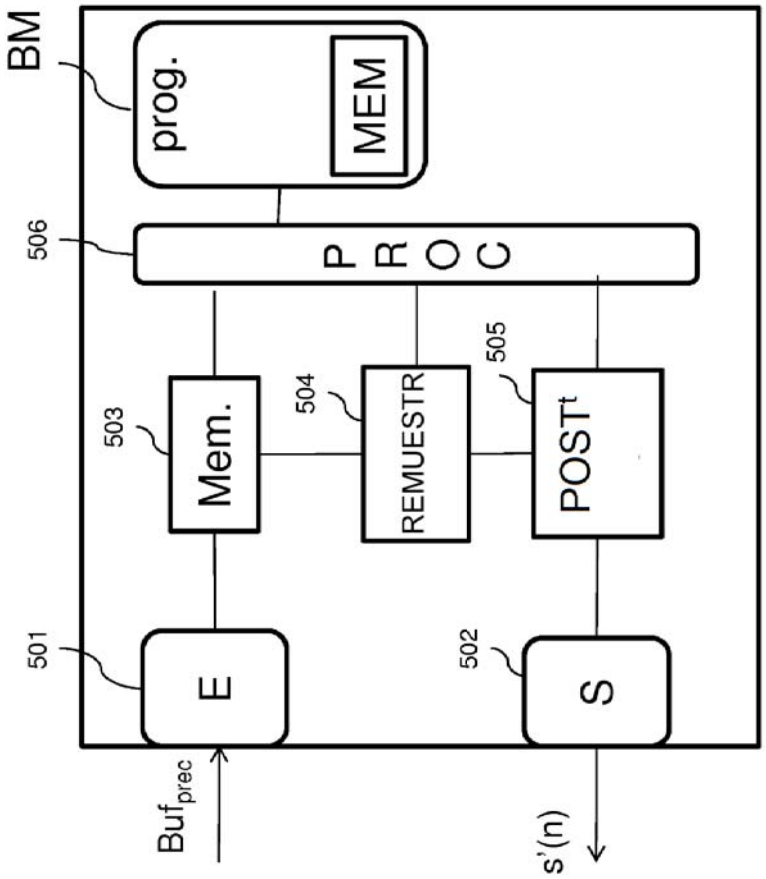
BM
prog.
MEM
506
P R O C
503
Mem.
504
REMUESTR
505
POST'
501
E
502
S
Buf$_{prec}$
s'(n)

Fig.5