

19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 394 730**

51 Int. Cl.:

G10L 13/08 (2006.01)

H04N 7/24 (2011.01)

G06F 17/22 (2006.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

96 Fecha de presentación y número de la solicitud europea: **10.01.2008 E 08706987 (8)**

97 Fecha y número de publicación de la solicitud europea: **02.12.2009 EP 2126733**

54 Título: **Dispositivo y procedimiento para generar una corriente de datos y dispositivo y procedimiento para leer una corriente de datos**

30 Prioridad:

16.02.2007 DE 102007007830

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

05.02.2013

73 Titular/es:

**FRAUNHOFER-GESELLSCHAFT ZUR
FÖRDERUNG DER ANGEWANDTEN
FORSCHUNG E.V. (100.0%)
Hansastraße 27c
80686 München, DE**

72 Inventor/es:

**ZINK, ALEXANDER;
PROSCH, MARKUS;
KORTE, OLAF;
KELLERMANN, CHRISTIAN y
LINZ, BERND**

74 Agente/Representante:

PONTI SALES, Adelaida

ES 2 394 730 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

Dispositivo y procedimiento para generar una corriente de datos y dispositivo y procedimiento para leer una corriente de datos

5 **[0001]** La presente invención se refiere a la transmisión de datos y en particular a una transmisión más fácil de datos con texto para receptores simples y texto más datos para receptores más complejos, en los que ambos tipos de receptores pueden evaluar la corriente de datos.

10 **[0002]** Por ahora, existe una pluralidad de terminales de receptor individuales, todos capaces de recibir la información transmitida de forma móvil. Es característico de este amplio espectro que las capacidades de procesamiento y de salida de dichos receptores de información diferentes varíen fuertemente. Un ordenador portátil, por ejemplo, tiene capacidades de procesado y de salida muy elevadas como receptor móvil de información transmitida, puesto que un ordenador portátil dispone de importantes recursos de procesador y de memoria. Por otra parte, por ejemplo, un teléfono móvil que también participa en un servicio de información, tal como un canal de radiodifusión de datos, tiene recursos de procesamiento y de salida muy limitados. Un pequeño receptor de radiodifusión móvil que ni siquiera está pensado como un teléfono móvil, pero sólo como un receptor de radiodifusión móvil para la recepción de datos, tales como los resultados de la liga de fútbol o los resultados de otros deportes, titulares de periódicos, noticias, clima, etc, tiene recursos de procesamiento y de salida aún más limitados si se utiliza un servicio de difusión de datos respectivo.

15 **[0003]** Este servicio de información basado texto para difusión digital, que es adecuado para la recogida y reutilización simple de datos, así como para la transmisión de difusión muy eficiente, existe bajo la designación "Journaline". Este servicio de datos a soporte a una gama muy amplia de tipos de receptor, que van desde soluciones rentables con una pequeña pantalla de texto de gama alta con receptores de interfaz gráfica de usuario y la reproducción texto-voz opcional.

20 **[0004]** El usuario puede procesar toda la información proporcionada por la estación de radio inmediatamente y de forma interactiva. A este respecto, el servicio puede ser comparado con el videotexto para la televisión. La información básica se proporciona en una forma de texto simple, en la que, sin embargo, al mismo tiempo, debe activarse la opción de representaciones gráficas más complejas, incluyendo una extensión a elementos multimedia tales como secuencias de vídeo o de imágenes y otras extensiones funcionales.

25 **[0005]** Por otra parte, es importante que una corriente de datos, por ejemplo de difusión por un transmisor DAB, sea compatible con versiones anteriores, lo que significa que al mismo pueda ser leído y procesado igualmente por receptores simples y por receptores más complejos o por receptores de base que sean simples, y por receptores de extensión, que son los receptores más complejos.

30 **[0006]** WO 02/052857 A23 describe el uso de dos códigos derivados diferentes, en el que uno de los dos se utiliza con más frecuencia y, en consecuencia provoca la compresión. El código de rama y el código de posición se combinan y se transmite la longitud de bits del código de rama. Esto resulta en una simplificación de una función de búsqueda de tal manera que una primera parte simplemente especifica el tipo del elemento referenciado y además se obtiene una mejora con respecto a la capacidad de expansión a partir de un número de versión a transmitir, y en estrategias de expansión estrictamente predeterminadas que son conocidas por un decodificador.

35 **[0007]** Es objeto de la presente invención proporcionar un concepto para generar y procesar una corriente de datos que puedan ser procesados tanto por receptores simples como complejos, y que permita una alta flexibilidad de transmisión y procesamiento de datos, y además donde los datos de texto sean interpretables por igual por ambos receptores simples y complejos.

40 **[0008]** Este objeto se resuelve mediante un dispositivo para generar una corriente de datos de acuerdo con la reivindicación 1, un dispositivo para leer una corriente de datos de acuerdo con las reivindicaciones 4 ó 5, un procedimiento para generar una corriente de datos de acuerdo con la reivindicación 11, un procedimiento para leer una corriente de datos según las reivindicaciones 12 ó 13, o un programa de ordenador según la reivindicación 14 o una corriente de datos de acuerdo con la reivindicación 15.

45 **[0009]** Aparte de datos de texto, una corriente de datos comprende una secuencia de inicio de escape que define un primer número de unidades de datos que deben ser saltadas por un descodificador de base y ser interpretadas por un descodificador de extensión, el primer número de unidades de datos, definiendo una secuencia de continuación de escape un segundo número de unidades de datos que de nuevo deben ser saltadas por un descodificador de base y ser interpretadas por un descodificador de extensión junto con el primer número de unidades de datos, así como el segundo número de unidades de datos y finalmente, si es necesario, datos de texto.

50 **[0010]** De esta manera, se garantiza que, por un lado, se pueden utilizar secuencias de inicio cortas, ya que el número de unidades de datos con la que está relacionada una secuencia de inicio de escape es, a lo sumo, tan alta como o menor que el número máximo de unidades de datos que puede ser señalizada por la secuencia de inicio de escape. Para los muy pocos casos, por lo general, donde el número de unidades de datos considerados meramente para

5 decodificadores de mayor calidad es mayor que el número de unidades de datos interpretables por la secuencia de inicio de escape, aún más, se proporciona una secuencia de continuación de escape que define un número de unidades de datos que han de ser interpretados conjuntamente con el primer número de unidades de datos por un decodificador de extensión. Así, para bloques de datos más cortos con un número menor de unidades de datos, siempre se requiere solamente la secuencia de inicio de escape, que es un código corto, ya que esta no está destinada a codificar una longitud arbitrariamente larga, sino simplemente una longitud limitada de unidades de datos. La flexibilidad se obtiene gracias a que, por otra parte, son posibles las introducciones de datos arbitrariamente largos en la corriente de datos por una repetición arbitrariamente frecuente de secuencias de continuación de escape y segundas unidades de datos después de una secuencia de continuación de escape.

10 **[0011]** En otras palabras, debido a la corriente de datos flexible la cantidad de datos que pueden ser introducidos en el texto para un decodificador de extensión es ilimitada. Sin embargo, esto no afecta a la longitud de la secuencia de inicio de escape, ya que la tarea de señalar la longitud de inserciones de datos muy largas se divide de hecho en varias secuencias, es decir, en la secuencia de inicio de escape y una secuencia de continuación de escape que ocurren más tarde en la corriente de datos y posiblemente, otras secuencias de inicio de escape, mientras que solamente se necesita una secuencia de inicio de escape muy corta para inserciones de datos cortas que ocurren con cierta frecuencia. Así, cuando el descodificador de extensión encuentra una secuencia de continuación de escape en la corriente de datos, sabe que las unidades de datos que se muestran con esta secuencia pertenecen a las primeras unidades de datos. Con respecto a flexibilización adicional, cuando se utiliza la corriente de datos, un indicador de tipo de datos se encuentra en las unidades de datos el cual indica el tipo de datos y por lo tanto el procesamiento que se realiza con estos datos, tanto de las primeras unidades de datos como de las segundas. El indicador de tipo de datos se encuentra, por ejemplo, en frente de las primeras unidades de datos referenciadas por la secuencia de inicio de escape, y en frente de las segundas unidades de datos referenciadas por la secuencia de continuación de escape.

15 **[0012]** A continuación, se describirán unas realizaciones preferidas de la presente invención con referencia a los dibujos que se acompañan. Estos muestran:

20 La figura 1 un diagrama de bloques de un dispositivo de una corriente de datos;

La figura 2a un diagrama de bloques de un receptor de base;

25 La figura 2b un diagrama de bloques de un receptor de extensión;

La figura 3a una ilustración de una corriente de datos;

30 La figura 3b una ilustración ampliada de una secuencia de inicio de escape y una secuencia de continuación de escape de acuerdo con un aspecto;

35 La figura 3c una ilustración de los datos referenciados por la secuencia de inicio de escape y la secuencia de continuación de escape de la figura 3a o la figura 3b, junto con un indicador del tipo de datos interpretables por un decodificador de extensión;

40 La figura 4a un ejemplo de corriente de datos sin código de continuación;

La figura 4b un ejemplo de codificación con código de continuación;

45 La figura 5 un diagrama de flujo de un procedimiento para generar una corriente de datos;

La figura 6A un diagrama de flujo ejecutado por un receptor de base;

50 La figura 6B un diagrama de flujo ejecutado por un receptor de extensión; y

La figura 7 diferentes tipos de datos tal como pueden ser indicados por un indicador de tipo de datos y como pueden ser procesados.

55 **[0013]** La figura 1 muestra un dispositivo para generar una corriente de datos que comprende medios 10 para introducir información en la corriente de datos, que es emitida en una salida 12. La corriente de datos se proporciona a los medios 14 para transmitir y/o almacenar la corriente de datos para transmitirla a un receptor o un decodificador de la corriente de datos, respectivamente, a través de una ruta de transmisión de espacio libre 16 en el ejemplo de una transmisión de radiodifusión. Alternativamente, la salida 16 de los medios 14 para transmitir o almacenar está conectada con un medio de memoria legible por ordenador, tal como una tarjeta de memoria conectada a la salida de los medios o proporcionada en el otro extremo de una trayectoria de transmisión, tal como en un receptor, cuando este receptor simplemente almacena los datos, pero no los procesa o los almacena y los procesa simultáneamente. Por lo tanto, un medio de memoria legible por ordenador puede ser una tarjeta de memoria no volátil en un receptor o un decodificador, o un disco duro de un ordenador portátil, o una memoria de trabajo de un decodificador, que retiene sus datos siempre y cuando los mismos se suministran con corriente o tensión, respectivamente.

5 **[0014]** Los medios 10 para introducir incluyen datos de texto 11a y unidades de datos 11b. Preferentemente, se realizan cálculos en bytes. Por lo tanto, una unidad de datos es de 8 bits o 1 byte de largo. Esta granularidad se prefiere, ya que tanto el texto como los datos pueden ser fácilmente manejados con esta granularidad. En este contexto, se prefiere usar el formato de codificación UTF8 para la codificación de texto, en el que se codifican típicos caracteres ASCII con un byte cada uno, mientras que, por ejemplo, se codifican diéresis alemanas con 2 bytes y, por ejemplo, los caracteres chinos con 3, 4 o más bytes. De ello se deduce que un receptor base es capaz de decodificar una codificación UTF8, que se puede realizar, por ejemplo, mediante el depósito de una tabla de descodificación UTF8. Dependiendo de la decodificación, los medios para introducir obtienen datos de texto y las unidades de datos en paralelo. En este caso, los medios para introducir también reciben una señal de control en tiempo 11c, que determina en qué momento los datos de texto y las unidades de datos deben ser suministrados, por ejemplo, en la corriente de datos en serie 12. Como alternativa, los medios 10 para introducir ya pueden recibir datos de texto y unidades de datos a través de una única entrada, que ya proporciona datos de texto y unidades de datos en el correcto orden temporal o de bits deseado.

15 **[0015]** Los medios para introducir están implementados para introducir datos de texto en la corriente de datos. Cuando se deben introducir unas unidades de datos que deben ser saltadas por un simple decodificador, mientras las unidades de datos deben ser leídas o procesadas por un codificador más complejo, se introduce una secuencia de inicio de escape en la corriente de datos tal como se indica, por ejemplo, en la figura 3a. La secuencia de inicio de escape 31 define un primer número de unidades de datos que deben ser saltadas por un decodificador de base y ser interpretadas por un decodificador de extensión. Este primer número de unidades de datos también se introduce en la corriente de datos, tal como se puede ver en la figura 3a en 32. Si el bloque de datos que hay que introducir incluye más unidades de datos que los que se deben definir por la secuencia de inicio de escape, los medios 10 introducirán una secuencia de continuación de escape 33 que define un segundo número de unidades de datos que deben ser saltadas por un decodificador de base, aunque, sin embargo, serán interpretadas por un decodificador de extensión junto con el primer número de unidades de datos. Este segundo número de unidades de datos también se introduce en la corriente de datos, tal como se puede ver en la figura 3a en 34.

25 **[0016]** Dependiendo de la implementación, es decir, cuando el bloque de datos es un gran bloque de datos que tiene incluso más datos que pueden ser definidos por la secuencia de continuación de escape, se escribe otra secuencia de continuación de escape en la corriente de datos, etc., hasta que se han introducido que todas las unidades de datos a introducir en esta posición en los datos de texto que deben ser interpretadas conjuntamente por un decodificador de extensión.

30 **[0017]** Entonces, los medios 10 para introducir de la figura 1 típicamente volverán a introducir los datos de texto, que se indican en la figura 3a, por ejemplo, como segundos datos de texto 35, de tal manera que se produce una corriente de datos en la que están presentes los primeros datos de texto 30 y los segundos datos de texto 35, que, en cierto modo, encierran datos que deben ser saltados por el decodificador de base, en el que el decodificador de base es al menos capaz de leer la secuencia de inicio de escape y particularmente el número de unidades de datos, así como la secuencia de continuación de escape y al menos el número de unidades de datos incluidas con el fin de ser capaz de saltarse el número correcto de unidades de datos en la corriente de datos.

35 **[0018]** La figura 3b muestra un ejemplo de una secuencia de inicio de escape 31 que comprende un código de inicio de escape 31a, así como un código de longitud aguas abajo 31b. La secuencia de continuación de escape 33 se construye de manera análoga, con un código de continuación separado 33a y un código de longitud aguas abajo 33b. Por ejemplo, todos los códigos 31a, 31b, 33a, 33b son cada uno de 1 byte o, respectivamente, de una unidad de datos de largo, lo cual tiene el efecto de que 256 unidades de datos pueden ser codificadas por el código de longitud. Esto significa que, cuando un bloque de datos es de más de 256 unidades de datos de largo, el código de continuación 33a debe escribirse o introducirse en la corriente de datos después de las 256 unidades de datos con el fin de codificar las unidades de datos restantes del bloque de datos con los códigos de longitud posteriores 33b.

45 **[0019]** Si una unidad de datos fuera respectivamente más larga, un código de longitud que tiene una longitud de una unidad de datos podría codificar un número mayor de unidades de datos, lo que tendría el efecto de que en una corriente de datos se reduciría el número total de códigos de continuación. Si, sin embargo, se selecciona el código de longitud de menos de 8 bits, el número máximo de datos que pueden ser codificados por el código de longitud es respectivamente correspondientemente más pequeño, por lo que de otro modo con las mismas circunstancias aumentaría el número de códigos de continuación en la corriente de datos. Típicamente, hay un óptimo de cierta longitud de unidades de datos para una cierta longitud media de los bloques de datos. Si el código de longitud fuera demasiado largo, este procedimiento sería ineficiente, puesto que entonces tendría que escribirse toda la longitud del código de longitud en la corriente de datos, incluso para solamente una inserción de datos muy corta. Si, por otro lado, el código de longitud fuera demasiado corto, se escribiría un código de continuación para el gran número de bloques de datos, que no serían necesarios, en particular, en la implementación que se muestra en la figura 3b si se hubiera utilizado un código de longitud mayor desde el principio.

50 **[0020]** La figura 4a muestra una corriente de datos a modo de ejemplo que resulta en una reproducción de "This is a great test!", tal como se indica en la figura 4a. La corriente de datos asociada tiene inicialmente unos primeros datos de texto 30 con el texto "This is a ". Esto requiere 10 bytes, puesto que se requiere un byte para cada letra y para los espacios en blanco en la codificación UTF8. Por lo tanto, los primeros datos de texto son de 10 bytes de longitud.

5 **[0021]** Entonces, se tienen que insertar los datos. Para señalar esto, el código de inicio de escape 31a, que es 1A en el ejemplo mostrado en la figura 4a, se puede encontrar en el índice de byte (basado en cero) 10, en el que el prefijo "0x" indica una representación hexadecimal. Por supuesto, cualquier otro código que difiere del código de texto podría ser insertado por el código de inicio de escape. En otras palabras, el código de inicio de escape 31a, que es de 1 byte de longitud en el ejemplo en la figura 4a, debe diferir del código que define los caracteres (números, letras,...) visualizables en la pantalla en una codificación UTF8. Por lo tanto, los códigos de escape son, por ejemplo, caracteres UTF8 de control u otros caracteres fijos que difieren de los códigos de texto para caracteres visualizables en la pantalla.

10 **[0022]** Después del código de inicio de escape 31a hay un código de longitud 31b, que indica la longitud del campo de datos que sigue al código de longitud 31b. El campo de longitud incluye el código 4, de modo que, en consecuencia, las primeras unidades de datos 32 en la figura 4a comprenden 5 bytes. El concepto subyacente es que el campo de longitud incluye implícitamente el término "-1", ya que una longitud de cero no tendría sentido.

15 **[0023]** Entonces, hay otro campo de texto siguiente, a saber, la palabra "great" con 5 bytes, con lo cual de nuevo, en el índice del byte 22, se escribe un código de inicio de escape 31a, seguido de un código de longitud 31b que tiene un valor de 5, puesto que entonces los datos de 6-byte se escriben en los índices de byte 24 a 29 después del código de longitud. En el lado de salida, hay otro bloque de texto 35, que tiene una longitud de 6 bytes, puesto que se necesita 1 byte para cada letra de "test" y puesto que se necesita 1 byte para el punto de exclamación.

[0024] Esto resulta en una corriente de datos de 36-byte de largo que incluye 11 bytes de datos adicionales insertados en dos posiciones diferentes.

20 **[0025]** Un decodificador de base leerá como entrada y representará los datos y entonces, cuando encuentre el código de inicio de escape 31a, lo interpretará en el sentido de que buscará un código de longitud asociado al código de inicio de escape. Entonces, una interpretación de este código de longitud hará que el decodificador de base omita las unidades de datos referenciadas por el código de inicio de escape y el código de longitud, es decir, hará caso omiso de estas y para dejar de considerarlas más allá. Entonces, el decodificador base leerá como entrada los datos de texto en los índices de bytes 17 a 21 para reconocer de nuevo un código 31a como un código de inicio de escape y para buscar una indicación de longitud asociada en el código 31b, con el fin de saltarse de nuevo el número de bytes señalizados en el código 31b.

25 **[0026]** De este modo, se asegura de una manera simple que un decodificador de base puede incluso leer una corriente de datos escrita para decodificadores de extensión nuevos o más complejos. El hecho de que el decodificador de base no es capaz de tratar con los datos, pero puede interpretar correctamente la secuencia de inicio de escape con el código e indicación de longitud garantiza la compatibilidad con versiones anteriores. Sin embargo, un decodificador de extensión no sólo es capaz de interpretar la secuencia de principio, es decir, los códigos 31a y 31b, sino también no sólo saltarse los datos y procesarlos con el fin de ser capaz de realizar, además de la visualización de texto simple, funciones adicionales que pueden ser controladas a través de la corriente de datos y los datos asociados a una secuencia de escape.

30 **[0027]** La figura 4b muestra un ejemplo de codificación con el código de continuación, en el que se supone en el ejemplo mostrado en la figura 4b que un bloque de datos es más largo que 256 bytes. Si se debe insertar un bloque de datos tan largo, primero, una secuencia de inicio de escape, que de nuevo comprende un código de inicio de escape 31a y un código de longitud aguas abajo 31b, se inserta en el índice byte de datos 0. Sin embargo, ahora, el valor de longitud en el código de longitud 31b no es cualquier número como en el índice de byte 11 o el índice de byte 23,

35 sino el máximo, a saber "FF". Luego, tras el código de longitud 31b, se escriben las 256 primeras unidades de datos 32 en la corriente de datos en los índices de byte 2 a 257. Puesto que hay muchas más unidades de datos, a saber 262 bytes en conjunto, el generador de corriente de datos determinará que aún no ha escrito todas las unidades de datos en la corriente de datos. Por consiguiente, él mismo escribe un código de continuación 33a en la corriente de datos, que diferirá del código de inicio de escape 31a, de modo que el decodificador sabe que las unidades de datos que pertenecen al código de continuación, el cual, por ejemplo, sigue inmediatamente al código de continuación, sigue perteneciendo a las unidades de datos que han sido procesadas o suministradas antes del código de continuación. En el ejemplo mostrado en la figura 4b, se deben introducir otras seis unidades de datos de modo que el código de longitud 33b que sigue al código de continuación señalice una longitud de 6 (código 0x05). Entonces, después de este código de longitud, la inserción en la corriente de datos se termina con la escritura de las segundas unidades de datos 34, y a continuación, a partir del índice de byte 266 que se muestra en la figura 4b, pueden seguir por ejemplo texto normales 35.

40 **[0028]** Debe tenerse en cuenta que, dependiendo de la aplicación, el código de continuación de escape 33a no difiere necesariamente del código de inicio de escape 31a. Esto no es importante para el decodificador base. Este tan sólo tiene que interpretar tanto el código de inicio de escape 31a como el código de continuación de escape 33a y leer el código de longitud aguas abajo esta interpretación, con el fin de saber cuántos datos deben omitirse. En este caso, un decodificador de extensión se implementaría de modo que el mismo asuma automáticamente, cuando un código de inicio de escape, seguido por un código de inicio de escape, seguido de un código de longitud de longitud mayor, es decir, por ejemplo, FF, ocurra por primera vez, de modo que las unidades de datos ulteriores sigan perteneciendo a las unidades de datos 32 y por lo tanto tengan que ser procesadas juntas. Por lo tanto, el significado del código de inicio de

escape se conmutaría por el hecho de que el código de longitud 31b que sigue al código de inicio de escape 31a indica una longitud mayor. Cuando el código de inicio de escape 31a y el código de continuación 33a tienen el mismo valor, el decodificador, cuando el código de longitud 31b tras el primer código de inicio de escape 31a no tiene los valores máximos, es decir FF, interpretaría el código de continuación tras las unidades de datos no como un código de continuación sino como un nuevo código de inicio, de modo que los datos que siguen al segundo código de inicio no se consideran como continuación de los datos que siguen al primer código de inicio sino como datos nuevos de un nuevo bloque de datos que pueden ser procesados de manera diferente. La interpretación de si los datos son datos de continuación o no son datos de continuación es de particular importancia cuando hay un indicador de tipo de datos en cada bloque de datos que consista en varias unidades de datos en una determinada posición, como por ejemplo al principio, que se lee cuando los datos no son datos de continuación, o cuando los datos son datos de continuación, no son esperados por un decodificador y tampoco leídos como salida, como se describirá a continuación con referencia a la figura 3c.

[0029] A continuación, con referencia a la figura 5, se ilustra un ejemplo de secuencia de etapas para generar una corriente de datos, ya que puede ser realizada por un generador de corriente de datos, que está estructurada tal como se muestra en la figura 1.

[0030] El diagrama de flujo de la figura 5 comienza con una etapa de introducción de texto 50. En la etapa 51, se comprueba si, después de la introducción de texto, existen datos que todavía tienen que ser introducidos. Si la respuesta a esta pregunta es no, el objeto de texto siguiente se introduce en la corriente de datos, tal como se ilustra en el bucle 52. Si, sin embargo, la respuesta a la pregunta es sí, los medios 10 para introducir introducen una secuencia de inicio de escape, tal como se ilustra en el etapa 53. La secuencia de inicio de escape incluye información acerca de la longitud de los datos, es decir, el número de unidades de datos. Después de introducir la secuencia de inicio de escape en la corriente de datos, se introducen las primeras unidades de datos, tal como se indica en 54. En una etapa 55, se comprueba si todos los datos pertenecientes a un bloque de datos se introducen en la corriente de datos. Si la respuesta a esta pregunta es no, es decir, que no hay más datos, se puede introducir de nuevo un objeto de texto o un código de escape, tal como se ilustra por un bucle 56.

[0031] Cabe señalar aquí que las secuencias de inicio de escape podrían estar delante de un texto, después de un texto o entre un texto. Como alternativa, sin embargo, una secuencia de inicio de escape también puede estar en frente de o detrás de otra secuencia de escape, donde otra secuencia de escape puede indicar cosas distintas de datos específicos.

[0032] Sin embargo, si se determina que aún hay datos, se introduce una secuencia de continuación de escape, tal como se muestra en la etapa 57. En esta secuencia de continuación de escape, también se determina la longitud de las unidades de datos que se introducen luego en una etapa 58. En una etapa 59, se comprueba si hay datos adicionales. Si este es el caso, es decir, si se introducen datos adicionales del bloque de datos, se escribe una secuencia de continuación de escape adicional, tal como se muestra en 60. Si, por el contrario, se introducen todos los datos, el texto se introduce de nuevo básicamente siempre de la misma manera tal como se muestra en la etapa 50. Por motivos de claridad, esta entrada de texto después de un bloque de datos se indica en la etapa 61 en la figura 5.

[0033] Cabe señalar que la verificación en la etapa 55 de si los datos están todavía presentes o en la etapa 59 de si aún hay datos no se va a realizar necesariamente cuando la longitud de los datos que se deben introducir se determina en paralelo o en un proceso separado, tal como se muestra en la figura 5 en el bloque 62. En el bloque 62, se determina la longitud de un bloque de datos a introducir. A partir del máximo número de unidades de datos a codificar en la secuencia de inicio de escape, el bloque 52 sabe inmediatamente cuántas secuencias de continuación se requieren. En la realización mostrada en la figura 4b, el bloque 62 determina que debe introducirse la longitud mayor en la secuencia de inicio de escape, y que entonces se introduce una secuencia de continuación cuya longitud también se determina, tal como se indica mediante las flechas de control 63a y 63b en la figura 5. En este caso, la verificación en el bloque 55 o bloque 59 no debe realizarse, tal como se indica mediante una flecha de conexión a trazos 64. Esta alternativa ya representa una solución en la que la secuencia de inicio de escape ya se ha acabado antes de introducir las unidades de datos, mientras que, en la segunda alternativa, la longitud sólo debe introducirse en la secuencia de inicio de escape después de haber introducido las primeras unidades de datos o las segundas unidades de datos, respectivamente. A continuación, con referencia a la figura 2a y a la figura 2b, se tratarán los receptores para la recepción o decodificación de la secuencia de datos, donde esta corriente de datos puede ser estructurada básicamente tal como se muestra en la figura 3a, a los efectos de que esta incluya datos de texto, una secuencia de inicio de escape, un número de unidades de datos, y a continuación una secuencia de continuación de escape y después de la secuencia de continuación de escape de nuevo un número de unidades de datos que puede ir seguido de datos de texto, o por otra secuencia de continuación de escape.

[0034] El receptor de base para la lectura de la corriente de datos comprende una interfaz de entrada 70 para la obtención de la corriente de datos 16. La corriente de datos se transmite entonces a un procesador 71, que está acoplado a una pantalla de texto 72 para leer el texto de la corriente de datos y visualizarlo, donde el procesador determina la longitud o el número de unidades de datos cuando este se encuentra con una secuencia de inicio de escape, para saltarse este número de unidades de datos, y donde el procesador, cuando el mismo se encuentra con una secuencia de continuación de escape, también se salta un número de unidades de datos que pertenecen a la secuencia de continuación, tal como se muestra en el bloque 71 de la figura 2a.

[0035] Un receptor de extensión tal como se muestra en la figura 2b comprende, además de los elementos del receptor de base de la figura 2a, un módulo o una funcionalidad del procesador 71 según el cual las unidades de datos no simplemente se omiten, si no que se ejecutan juntas.

5 **[0036]** Además de la visualización de texto 72, que puede funcionar o funcionará de la misma manera en el receptor de extensión que en el receptor base, el receptor de extensión realizará una interpretación de las unidades de datos después de la secuencia de inicio de escape y la secuencia de continuación de escape.

[0037] A continuación, con referencia a la figura 6A y la figura 6B, se hace una comparación de las funcionalidades de un receptor base sin opción de procesamiento de datos y un receptor de extensión con opción de procesamiento de datos.

10 **[0038]** Cuando tanto el receptor de base como el receptor extensión leen datos de texto, el procesador 71 de la figura 2a o figura 2b procesará los datos de texto, los decodificará y, a continuación los proporcionará a la pantalla de texto 72 y los mostrará, tal como se muestra en la etapa 80 de la figura 6A y la figura 6B. Si los receptores se encuentran con una secuencia de inicio de escape, esta secuencia de inicio de escape será leída, tal como se ilustra en una etapa 82. En particular, el receptor base verá en la etapa 82 cuantas unidades de datos se indican en la secuencia de inicio de escape, es decir, por ejemplo, por el código de longitud 31b o la figura 3b. En una etapa 83, el receptor base se saltará entonces el número de unidades de datos que ha determinado en la etapa 82. En contraste, en una etapa 84, el receptor de extensión no omitirá las unidades de datos que siguen a la secuencia de inicio de escape sino que los leerá. Por lo tanto, preferentemente, el receptor de extensión también necesita la información de longitud. Si los datos fueran, sin embargo, codificados de manera diferente que el texto, no se necesitaría forzosamente la información de longitud.

20 **[0039]** En una etapa 85, tanto el receptor base como el receptor extensión leen la secuencia de continuación de escape que sigue a las primeras unidades de datos 32 de la figura 3a. Aquí, el receptor base está principalmente interesado en el código de longitud con el fin de averiguar cuantos datos deben ser omitidos, lo cual se realiza a continuación por el receptor base en una etapa 86. Por otra parte, el receptor de extensión no omitirá el segundo número de unidades de datos en una etapa 87, sino que los leerá, tal como se ilustra en 87. Entonces, en una etapa 88, en la que no hay receptor base paralelo, el receptor de extensión activará la funcionalidad o el módulo 73 de la figura 2b, y logrará una ejecución común de las unidades de datos primeras y segundas. Cuando todas las unidades de datos han sido procesadas, tanto el receptor base como el receptor de extensión mostrarán los datos de texto en una segunda etapa 89, es decir, los datos 35 que siguen a la corriente de datos mostrada en la figura 3a. Si, sin embargo, el segundo código de longitud 33b también tiene la longitud mayor, es decir en este ejemplo "FF", el receptor de extensión puede leer como entrada un código de continuación de escape adicional, interpretar el código de longitud siguiente y a continuación añadir también aquellas unidades de datos referenciadas a las unidades de datos primeras y segundas con el fin de permitir de nuevo una ejecución común.

30 **[0040]** Posteriormente, con referencia a la figura 3c, se discutirá con más detalle lo que el receptor de extensión realiza en la etapa 88, o cómo se interpretan los datos que se han extraído después d el código de inicio de escape y el código de continuación.

35 **[0041]** En un ejemplo, el primer número de unidades de datos comprende un indicador de tipo de datos 90, que es, por ejemplo, de 1 byte. Este indicador de tipo de datos 90 se puede encontrar sólo en un cierto byte, por ejemplo el primer byte del primer número de unidades de datos, que se referencia mediante una secuencia de inicio de escape 31. En comparación, no hay este indicador de datos en el segundo número de unidades de datos, sino que el segundo número de unidades de datos 34 contribuye totalmente a que las unidades de datos se ejecuten conjuntamente, lo que se denomina "carga útil". Cuando se usa un indicador de tipo de datos 90, un descodificador de extensión interpretará los datos incluidos después del código de continuación como pertenecientes al mismo tipo que los datos que se incluyen en el primer número de unidades de datos. Esto permite que no se requieran señalizaciones para el segundo número de unidades de datos que siguen al código de continuación en la corriente de datos, que incluyen información acerca de si se trata de datos de continuación o qué tipo de datos tiene el segundo número de unidades de datos. En su lugar, el indicador de tipo de datos en el primer número de unidades de datos se utiliza también simplemente o se aplica al segundo número de unidades de datos, o los datos en el segundo número de unidades de datos se añaden simplemente al primer número de unidades de datos, respectivamente, como si nunca hubiese habido separación, de modo que entonces se ejecutarán o procesarán juntas.

40 **[0042]** Es una ventaja de la corriente de datos escalable de la invención que la misma se basa en las normas generales. Por lo tanto, los objetos pueden ser importados y tratarse en el formato XML.

45 **[0043]** La corriente de datos es especialmente adecuada para sistemas de radiodifusión digital, como un canal de datos adicional que aporta un valor adicional a los oyentes, ya que ahora tienen acceso inmediato a la información de texto donde quiera que estén, donde los receptores para la radiodifusión digital son simplemente receptores que representan al menos información de texto, o especialmente compleja y, por lo tanto, por supuesto, receptores de mayor coste que pueden realizar cualquier procesamiento del primer número de unidades de datos y el segundo número de unidades de datos. De ello se deduce que los receptores que tienen una pantalla de texto que son económicos y por lo tanto están disponibles como un producto de masa pueden generar un valor adicional para el oyente. Sin embargo, la corriente de datos también es adecuada para receptores de gama alta con una interfaz gráfica de usuario y una reproducción de voz

opcional. Todo esto se obtiene mediante implementaciones sencillas, incluso en los receptores de bajo coste y mediante un procedimiento particularmente simple de utilización donde el usuario no tiene que preocuparse de que los datos sean actualmente válidos. En su lugar, borrando los datos o ejecutando los datos, en función de lo que tenga el receptor, se realiza de forma totalmente automática sin que el usuario tenga que preocuparse de este.

5 **[0044]** Además, en ciertos ejemplos, los datos de texto se ilustran de una manera orientada a objetos, donde estos objetos son unidades independientes y autónomos. Por lo tanto, no hay estructuras de datos globales a montar o mantener en el receptor. Los objetos se transmiten en la forma de un carrusel de datos, y el almacenamiento en caché de datos se usa ventajosamente en el receptor. Los datos de texto que pueden ser, por ejemplo, diseños de menú, reportajes o noticias ticker se transmiten como los llamados objetos NML, donde NML significa News Service Mark Language y es similar a las representaciones de contenidos codificados binarios basados en XML.

10 **[0045]** Preferentemente, se utiliza el código hexadecimal 1A como código de inicio de escape 31a, y se utiliza el código hexadecimal 1B, por ejemplo, como código de continuación. El indicador de tipo de datos 90 de la figura 3c puede indicar varios tipos de datos, tal como se ilustra a continuación. Los siguientes datos de valores de los indicadores de tipo son meramente a título de ejemplo. Un indicador de "00" indica el relleno. Los datos incluidos llevan bytes de relleno, en el que este contenido es ignorado tanto por el decodificador de extensión como por el decodificador de base. Un tipo alternativo de conjunto de datos tiene un indicador de "01" y representa un tiempo de parada absoluto. Se define el período de tiempo de parada de presentación absoluto de un objeto NML. Cuando este tiempo de parada haya pasado, el objeto NML ya no se mostrará. Esta representación de código de secuencia de escape es necesaria para el marcado de objetos individuales NML. Se ilustra una alternativa de tiempo de parada general para todo el servicio,. La carga útil después del indicador del tipo de datos incluye el número de 15 minutos desde el 1.1.2000, como un número entero de 24 bits sin signos, que abarca más de 450 años. Un indicador del tipo de datos del tipo "02" indica un tiempo de parada relativo. Aquí, se define el período de tiempo de parada de presentación relativa de un objeto NML. Transcurrido este período, ya no se muestra el objeto NML. El tiempo de parada comienza con cada recepción de un objeto NML, incluso cuando el objeto está ya almacenado en la memoria caché. Este código de secuencia de escape sirve para el marcado de objetos individuales NML. Los datos de carga útil incluyen el número de minutos como un número entero de 16 bits sin signo, que abarca más de 45 días. Generalmente, los datos de tiempo de parada son procesados por el control de tiempo de parada 100 de la figura 7.

15 **[0046]** El indicador del tipo de datos "03" se refiere a un objetivo de enlace general. Un objetivo de enlace general es un objetivo que se presenta o se activa, por ejemplo, mediante el control de conexión 101, cuando el usuario exige explícitamente la ejecución de una acción, es decir, cuando se ilustra o proporciona un funcionalidad de "botón caliente", respectivamente, para la interacción con el usuario. Los objetivos generales de enlace se pueden definir para todos los tipos de objetos NML. La disponibilidad de un objetivo de enlace general para un objeto NML que se muestra actualmente se comunica al usuario de alguna u otra manera cuando tiene un receptor de extensión. Los datos de carga útil que se muestran, por ejemplo, en la figura 3c tienen el siguiente formato. 1 byte representa un tipo de conexión y n bytes representan la dirección de enlace. Los siguientes valores de tipo de enlace, por ejemplo, están disponibles. Cuando el tipo de enlace tiene un valor, por ejemplo, de "00", los 2 bytes siguientes son un ID de objeto de otro objeto NML en el mismo servicio de datos.

20 **[0047]** Otro byte de tipo de enlace, por ejemplo, de "01", seguido de una cadena URI, apunta a diferentes multiplexores DAB / DRM, servicios o elementos de servicio.

25 **[0048]** Otro tipo de conexión, como por ejemplo "02" que muestra una cadena de dirección URL, apunta a una dirección de Internet o un documento.

30 **[0049]** Otro tipo de enlace, por ejemplo, "03", seguido de un número de teléfono, apunta a un servicio de voz que puede ser logrado por teléfono. El número comienza aquí, por ejemplo, con un prefijo internacional, por ejemplo "+ [código internacional de país]".

35 **[0050]** Por lo general, un receptor de extensión está diseñado para hacer caso omiso de valores de tipo de enlace desconocidos.

40 **[0051]** Se considera otro indicador de tipo de datos "FF", por ejemplo, como un indicador propietario de tipo de datos, donde los datos que preceden solo conciernen a determinados receptores de extensión que pueden evaluar estos datos propietarios.

45 **[0052]** A diferencia de los tipos de gestión de objetos mencionados, las unidades de datos también pueden incluir tipos de gestión de contenidos. Un indicador de tipo de datos de "20", también conocido como palabra clave, marca una palabra clave siguiente junto con una descripción de palabras clave opcional. Puede utilizarse la palabra clave, por ejemplo, para la generación de un índice de búsqueda de basado en receptor, tal como se muestra por los medios de generación de consulta 102 en la figura 7. La sección de carga útil de datos tiene el siguiente formato. Primero viene un código de longitud que tiene la longitud de la palabra clave, donde el valor igual a la longitud -1 se indica como número entero sin signo. Se identifica el número de caracteres de texto visuales después de esta sección de datos que debe ser tratado como una palabra clave. A continuación viene una descripción con n bytes, en la que la descripción opcional puede ser indexada además de la palabra clave y / o se puede mostrar al usuario.

5 [0053] Otro indicador del tipo de datos, por ejemplo de "21", representa una definición de macro. Una macro permite la definición de las secciones de texto, incluyendo secuencias de escape opcionales que se pueden insertar varias veces en cualquier parte de la sección de contenidos con una simple referencia. Por ejemplo, la macro puede definir descripciones de voz que van a ser ilustradas además de los elementos de texto textuales. La sección de datos tiene un formato donde inicialmente una ID de macro (0 a 255) de un byte identifica las definiciones de las macros siguientes. A continuación viene una definición de macro con n bytes. El texto (que incluye secuencias de escape), que siempre debe insertarse cuando esta macro esta referenciada por su ID, está incluido en la definición de macro con n bytes. Cabe señalar que las macros no deben utilizarse necesariamente para la información esencial, ya que los receptores también pueden ignorarlas. Otro indicador del tipo de datos "22" sirve, por ejemplo, para una referencia de macro. La definición de macro referenciada por su ID se introduce virtualmente en esta posición de la secuencia de escape para mostrarla al usuario. La sección de datos incluye una ID de macro de 1 byte (0 a 255) que hace referencia a una definición de macro. Las macros son procesadas generalmente por el procesador macro 103 en la figura 7.

15 [0054] Otro grupo de tipos de datos puede comprender tipos de soporte de voz. Un indicador del tipo de datos de, por ejemplo "A0", define un lenguaje estándar o un idioma predefinido. Aquí, el idioma predeterminado del objeto NML se describe o referencia, respectivamente. La sección de datos, es decir, la carga útil de la figura 3c, tiene un código de lenguaje de tres letras ISO de letras minúsculas.

20 [0055] Otro tipo de soporte del voz, por ejemplo, referenciado con "A1" es la sección de idioma. Esta define el lenguaje de un número especificado de una sección de texto o una parte específica de un objeto NML. La sección de carga útil tiene el siguiente formato. Hay una longitud de texto que tiene un byte, cuyo valor es igual al valor del número de unidades de datos de esta longitud de texto -1, como número entero sin signo. Esto identifica el número de caracteres de texto visuales que siguen a esta sección de datos para el que se aplica la definición de lenguaje. Entonces, viene un grupo de 3 bytes con una definición de lenguaje, que lleva un código de lenguaje de 3 letras ISO de letras minúsculas.

25 [0056] Otro tipo de idioma está indexado, por ejemplo, por el indicador de tipo de datos "A2". Se refiere a los fonemas del voz y define una descripción de fonema de una sección de texto utilizando el alfabeto fonético internacional (IPA). La sección de carga útil tiene un formato que tiene una primera longitud de texto con un byte cuyo valor es un número entero. Este byte indica el número de caracteres de texto visuales después de esta sección de datos, que han de ser representados por una definición fonética. Entonces viene un grupo de n bytes con los fonemas del IPA sigue. Este grupo incluye las definiciones de fonemas de los fonemas como notaciones IPA.

30 [0057] Otro tipo de soporte de la voz, por ejemplo referenciado por el indicador de tipo de datos "A3", comprende un corte de discurso, que define un corte para los procesadores de texto a voz que se debe insertar en la posición de la secuencia de escape. La sección de datos lleva 1 byte como un número entero sin signo, en el que este byte define un período de voz en unidades de 0,1 segundos.

35 [0058] Otro tipo de soporte de voz que puede, por ejemplo, tener el indicador de tipo de datos "A4" define los caracteres y en particular varios de los caracteres de texto visuales tras la secuencia de escape que han de ser tratados como caracteres individuales o números en lugar de palabras o números continuos por el procesador texto a voz. La sección de carga útil transporta 1 byte que define el número de caracteres de texto visuales como un número entero sin signo que tiene un valor respectivo.

40 [0059] Cabe señalar que todos estos tipos de datos, dependiendo de la cantidad máxima que puede ser representada por el código de longitud, o bien pueden ser representados en el primer número de unidades de datos 32 de la figura 3a o, cuando el número de unidades de datos no es suficiente para la escritura de todos los datos, se incluyen en el segundo número de unidades de datos que pueden cada una ser iniciadas con una secuencia de continuación de escape. De ello se desprende que no necesariamente todos los tipos de datos que están referenciados por un indicador del tipo de datos necesitan tener unidades de datos tanto después de una secuencia de inicio de escape como de una secuencia de continuación de escape. En cambio, incluso los tipos de datos cortos pueden incluir simplemente datos que no requieren continuación ya que el número de unidades de datos de estos datos es menor que el número máximo de unidades de datos que pueden representarse por el código de longitud 31b.

50 [0060] Dependiendo de las circunstancias, cada procedimiento de la invención puede ser implementado en hardware o en software. La implementación puede realizarse en un medio de memoria digital, en particular un disco o un CD con señales de control legibles electrónicamente que pueden cooperar con un sistema informático programable de tal manera que se lleve a cabo el procedimiento. Así, generalmente, la invención también consiste en un producto de programa de ordenador con un código de programa almacenado en un soporte legible por máquina para la realización de un procedimiento de la invención cuando el producto de programa de ordenador se ejecuta en un ordenador. En otras palabras, la invención se puede realizar como un programa informático con un código de programa para realizar el procedimiento, cuando el programa informático se ejecuta en un ordenador.

55 [0061] De acuerdo con una realización del dispositivo para generar, la corriente de datos es una corriente de datos en serie y los medios 10 para introducir están implementados para introducir la secuencia de inicio de escape 31 en la corriente de datos después o en frente de los datos de texto 30 y para introducir la secuencia de continuación de escape 33 después del primer número 32 de unidades de datos en la corriente de datos.

- [0062]** Según una realización preferida del dispositivo para generar, los medios 10 para introducir están implementado para colocar primero los datos de texto 30, entonces la secuencia de inicio de escape 31, luego el primer número de unidades de datos, luego la secuencia de continuación de escape 33 y luego el segundo número 34 de unidades de datos en la corriente de datos.
- 5 **[0063]** Según una realización preferida del dispositivo para generar, el código de longitud 31b se introduce en la corriente de datos inmediatamente después del código de inicio de escape 31a, o el código de longitud 33b se introduce en la corriente de datos inmediatamente después del código de continuación 33a.
- [0064]** Según una realización preferida del dispositivo para generar, el código de longitud 33b introducido en la corriente de datos después del código de continuación 33a y el código de longitud 31b introducido en la corriente de datos
10 después del código de inicio de escape 31a se origina a partir de la misma tabla de codificación para la codificación de longitud.
- [0065]** Según una realización preferida del dispositivo para generar, las unidades de datos en la corriente de datos son idénticos y comprenden una pluralidad predefinida de bits cada una.
- 15 **[0066]** En una realización preferida del dispositivo para la generación, la pluralidad predefinida de bits es igual a 8, de tal manera que una unidad de datos es de 1 byte.
- [0067]** Según una realización preferida del dispositivo para generar, la corriente de datos tiene que ser legible por un primer tipo de receptores base y por un segundo tipo de receptores de extensión, en el que los medios 10 para introducir están implementados para utilizar códigos de datos de texto que pueden ser detectados por los receptores base y los receptores de extensión.
- 20 **[0068]** Según una realización preferida del dispositivo para generar, el código de inicio de escape 31a tiene la misma longitud que un código de continuación 33a dentro de una secuencia de continuación de escape 33, en el que esta longitud es igual a la longitud de una unidad de datos.
- [0069]** Según una realización preferida del dispositivo para generar, el código de longitud está implementado para codificar una longitud de contenido de 1 a 256 bytes.
- 25 **[0070]** Según una realización preferida del dispositivo para generar, el indicador de tipo de datos 90 es tal que el mismo no tiene que ser interpretable por un decodificador de base, sino interpretable por un decodificador de extensión.
- [0071]** Según una realización preferida del dispositivo para generar, los medios 10 para introducir está implementado para introducir, después del indicador de tipo de datos, datos que tienen el tipo de datos indicado por el indicador del tipo de datos 90.
- 30 **[0072]** Según una realización preferida del dispositivo para generar, los medios 10 para introducir están implementados para introducir el indicador de tipo de datos 90 en la corriente de datos en el primer número de unidades de datos 32 inmediatamente después de un código de longitud 31b de la secuencia de inicio de escape 31.
- [0073]** Según una realización preferida del dispositivo para leer, la corriente de datos está asegurada en una memoria búfer y el procesador 71 está implementado para borrar el primer número de unidades de datos 32 y el segundo número
35 de unidades de datos 34 de la memoria búfer y para leer continuamente la memoria búfer, o para controlar la memoria búfer de manera que un área de la memoria búfer donde se almacenan el primer número de unidades de datos 32 y el segundo número de unidades de datos 34 se omite cuando se lee en la memoria.
- [0074]** Según una realización preferida del dispositivo para leer, el primer número 32 de unidades de datos comprende un indicador de tipo de datos 90 para indicar un tipo de datos del primer número de unidades de datos 32 y el segundo
40 número de unidades de datos 34, y el procesador 71 está implementado para extraer indicador de tipo de datos 90 y para procesar el primer número de unidades de datos 32 y el segundo número de unidades de datos 34 conjuntamente según el indicador de tipo de datos 90.
- [0075]** Según una realización preferida del dispositivo para leer, el objetivo es un objeto con datos de texto en la corriente de datos, un multiplex DAB / DRM, un servicio o elemento de servicio, una dirección de Internet, un documento
45 sobre Internet o un número de teléfono.
- [0076]** Según una realización preferida del dispositivo para leer, el procesador 71 está implementado para ignorar las primeras unidades de datos 32 o las segundas unidades de datos 34 cuando el mismo define un destino no interpretable.

5 **[0077]** Según una realización preferida del dispositivo para leer, el primer número de unidades de datos 32, el segundo número de unidades de datos 34 u otras unidades de datos tienen un indicador de tipo de datos 90 que comprende datos de soporte de voz para un procesador de voz 73, que indican un lenguaje de una sección de texto, comprenden fonemas del voz, están relacionados con un corte del voz o tiene caracteres de lenguaje, y el procesador 71 está implementado para emitir como salida datos de soporte de voz y para suministrarlos al procesador de voz para generar o influenciar una salida de voz.

[0078] En una realización preferida, la secuencia de datos se almacena en un medio legible por ordenador.

REIVINDICACIONES

- 5 1. Dispositivo para generar una corriente de datos, que comprende medios (10) para introducir datos de texto (30) en la corriente de datos, para introducir una secuencia de inicio de escape (31) en la corriente de datos, en el que la secuencia de inicio de escape define un primer número de unidades de datos que deben ser saltadas por un descodificador de base y ser interpretadas por un descodificador de extensión, para introducir el primer número (32) de unidades de datos en la corriente de datos, para introducir una secuencia de continuación de escape (33) en la corriente de datos, en el que la secuencia de continuación de escape define un segundo número de unidades de datos que deben ser saltadas por un descodificador de base y ser interpretadas por un descodificador de extensión junto con el primer número de unidades de datos, y para introducir el segundo número (34) de unidades de datos en la corriente de datos,
- 10 en el que la secuencia de inicio de escape es un código para codificar una cantidad limitada de unidades de datos y en el que son posibles inserciones de datos de cualquier longitud mediante cualquier número de repeticiones de secuencias de continuación de escape e introducir unas segundas unidades de datos asociadas y
- 15 en el que la secuencia de inicio de escape (31) comprende un código de inicio de escape (31 a) y un primer código de longitud (31b), y la secuencia de continuación de escape (33) comprende un código de continuación (33a) y un segundo código de longitud (33b), en el que el código de inicio de escape (31a) difiere del código de continuación (33a), o en el que el código de inicio de escape (31a) y el código de continuación (33a) son idénticos y el primer código de longitud (31b) de longitud mayor señala que el siguiente código en la corriente de datos es un código de continuación (33a).
- 20 2. Dispositivo según la reivindicación anterior, en el que el código de inicio de escape (31a) o el código de continuación (33a) difieren de los códigos de texto.
- 20 3. Dispositivo según cualquiera de las reivindicaciones anteriores, en el que los medios (10) están configurados para introducir, según una relación determinada con la secuencia de inicio de escape (31), un indicador de tipo de datos (90) en la corriente de datos que indica un tipo de datos tanto del primer número de unidades de datos como del segundo número de unidades de datos (34).
- 25 4. Dispositivo para leer una corriente de datos que tiene datos de texto (30), definiendo una secuencia de inicio de escape (31) un primer número de unidades de datos (32), el primer número de unidades de datos (32), definiendo una secuencia de continuación de escape (33) un segundo número de unidades de datos, el segundo número de unidades de datos (34), que comprende medios (72) para reproducir datos de texto (30); y un procesador (71) para interpretar la secuencia de inicio de escape (31) de modo que a partir de la secuencia de inicio de escape se determina el primer número de unidades de datos, para saltarse el primer número de unidades de datos y para interpretar la secuencia de continuación de escape (33) para determinar el segundo número de unidades de datos (34) a partir de la secuencia de continuación de escape, saltarse el segundo número de unidades de datos (34), en el que la secuencia de inicio de escape (31) comprende un código de inicio de escape (31a) y un primer código de longitud (31b) y la secuencia de continuación de escape (33) comprende un código de continuación (33a) y un segundo código de longitud (33b), en el que el código de inicio de escape (31a) difiere del código de continuación (33a) o en el que el código de inicio de escape (31a) y el código de continuación (33a) son idénticos y el primer código de longitud (31b) de longitud mayor señala que el siguiente código en la corriente de datos es un código de continuación (33a).
- 30 30
- 35 35
- 40 5. Dispositivo para leer una corriente de datos que tiene datos de texto (30), definiendo una secuencia de inicio de escape (31) un primer número de unidades de datos (32), el primer número de unidades de datos (32), definiendo una secuencia de continuación de escape (33) un segundo número de unidades de datos, el segundo número de unidades de datos (34), que comprende medios (72) para reproducir los datos de texto (30); un procesador (71) para interpretar la secuencia de inicio de escape (31), de modo que el primer número de unidades de datos (32) sea leído, para interpretar la secuencia de continuación de escape (33) de modo que el segundo número de unidades de datos (34) sea leído y para procesar conjuntamente (73) el primer número de unidades de datos (32) y el segundo número de unidades de datos (34) además de o en lugar de representar los datos de texto (30), en el que la secuencia de inicio de escape (31) comprende un código de inicio de escape (31a) y un primer código de longitud (31b) y la secuencia de continuación de escape (33) comprende un código de continuación (33a) y un segundo código de longitud (33b), en el que el código de inicio de escape (31a) difiere del código de continuación (33a), o en el que el código de inicio de escape (31a) y el código de continuación (33a) son idénticos y el primer código de longitud (31b) de longitud mayor señala que el siguiente código en la corriente de datos es un código de continuación (33a).
- 45 45
- 50 6. Dispositivo según la reivindicación 5, en el que el primer número de unidades de datos (32) y el segundo número de unidades de datos (34) definen un objetivo de ensamblado, y en el que el procesador (71) está implementado para ejecutar un ensamblado de datos según el objetivo.
- 55 7. Dispositivo según cualquiera de las reivindicaciones 5 y 6, en el que el primer número de unidades de datos (32) o el segundo número de unidades de datos (34) u otro número de unidades de datos definen un periodo de tiempo de parada absoluta o relativa para un objeto, y en el que el procesador (71) está implementado para ejecutar un control del tiempo de parada (100) que presenta una comparación entre el tiempo de parada y un tiempo de procesador para

reproducir los datos de texto y contenidos multimedia extendidos solamente cuando el tiempo de parada no ha sido alcanzado por el tiempo de procesador.

5 **8.** Dispositivo según cualquiera de las reivindicaciones 5 a 7, en el que el primer número de unidades de datos (32) y el segundo número de unidades de datos (34) u otras unidades de datos definen caracteres de texto que deben ser interpretados como palabras clave, y en el que el procesador (71) comprende medios de generación de solicitud de búsqueda (102) para generar una interrogación a la base de datos en el dispositivo para decodificar a partir de la palabra clave.

10 **9.** Dispositivo según cualquiera de las reivindicaciones 5 a 8, en el que el primer número de unidades de datos (32) o el segundo número de unidades de datos (34) u otras unidades de datos tienen macrodatos y el procesador (71) está implementado para realizar una funcionalidad de procesador de macro (103) para ejecutar una macro definida por los macrodatos.

15 **10.** Dispositivo según cualquiera de las reivindicaciones 5 a 9, en el que el primer número de unidades de datos (32), el segundo número de unidades de datos (34) u otro número de unidades de datos tienen un indicador de tipo de datos (90), que representan una descripción de voz, en el que el procesador (71) comprende un procesador de voz (73, 104) que puede ser controlado por las unidades de datos para influenciar una salida de voz, en el que el procesador (71) está implementado para emitir como salida datos de soporte de voz y suministrarlos al procesador de voz para generar una salida de voz o tener influencia en esta.

20 **11.** Procedimiento para generar una corriente de datos, que comprende introducir datos de texto (30) en la corriente de datos; introducir una secuencia de inicio de escape (31) en la corriente de datos, definiendo la secuencia de inicio de escape un primer número de unidades de datos que deben ser saltadas por un descodificador de base y ser interpretadas por un descodificador de extensión, introducir el primer número (32) de unidades de datos en la corriente de datos, introducir una secuencia de continuación de escape (33) en la corriente de datos, definiendo la secuencia de continuación de escape un segundo número de unidades de datos que deben ser saltadas por un descodificador de base y ser interpretadas por un descodificador de extensión junto con el primer número de unidades de datos; e introducir el segundo número (34) de unidades de datos en la corriente de datos, en el que la secuencia de inicio de escape es un código para codificar una cantidad limitada de unidades de datos y en el que son posibles inserciones de datos de cualquier longitud mediante cualquier número de repeticiones de secuencias de continuación de escape e introducir unas segundas unidades de datos asociadas y en el que la secuencia de inicio de escape (31) comprende un código de inicio de escape (31a) y un primer código de longitud (31b), y la secuencia de continuación de escape (33) comprende un código de continuación (33a) y un segundo código de longitud (33b), en el que el código de inicio de escape (31a) difiere del código de continuación (33a), o en el que el código de inicio de escape (31a) y el código de continuación (33a) son idénticos y el primer código de longitud (31b) de longitud mayor señala que el siguiente código en la corriente de datos es un código de continuación (33a).

35 **12.** Procedimiento para leer una corriente de datos que tiene datos de texto (30), definiendo una secuencia de inicio de escape (31) un primer número de unidades de datos (32), el primer número de unidades de datos (32), definiendo una secuencia de continuación de escape (33) un segundo número de unidades de datos, el segundo número de unidades de datos (34), que comprende representar (72) datos de texto (30); interpretar la secuencia de inicio de escape (31) de manera que se determina el primer número de unidades de datos a partir de la secuencia de inicio de escape; saltarse el primer número de unidades de datos a partir de la etapa de interpretar la secuencia de inicio de escape; interpretar la secuencia de continuación de escape (33) de modo que el segundo número de unidades de datos (34) se determina a partir de la secuencia de continuación de escape, saltarse el segundo número de unidades de datos (34) a partir de la etapa de interpretar la secuencia de continuación de escape; en el que la secuencia de inicio de escape (31) comprende un código de inicio de escape (31 a) y un primer código de longitud (31b) y la secuencia de continuación de escape (33) comprende un código de continuación (33a) y un segundo código de longitud (33b), en el que el código de inicio de escape (31a) difiere del código de continuación (33a) o en el que el código de inicio de escape (31a) y el código de continuación (33a) son idénticos y el primer código de longitud (31b) de longitud mayor señala que el siguiente código en la corriente de datos es un código de continuación (33a).

50 **13.** Procedimiento para leer una corriente de datos que tiene datos de texto (30), definiendo una secuencia de inicio de escape (31) un primer número de unidades de datos (32), el primer número de unidades de datos (32), definiendo una secuencia de continuación de escape (33) un segundo número de unidades de datos, el segundo número de unidades de datos (34), que comprende interpretar la secuencia de inicio de escape (31) de modo que se determina el primer número de unidades de datos (32), leer como entrada el primer número de unidades de datos a partir de la etapa de interpretar la secuencia de inicio de escape; interpretar la secuencia de continuación de escape (33) de modo que se determina el segundo número de unidades de datos (34); leer como entrada el segundo número de unidades de datos a partir de la etapa de interpretar la secuencia de continuación de escape; y procesar conjuntamente (73) el primer número de unidades de datos (32) y el segundo número de unidades de datos (34) además de o en lugar de representar los datos de texto (30), en el que la secuencia de inicio de escape (31) comprende un código de inicio de escape (31 a) y un primer código de longitud (31b) y la secuencia de continuación de escape (33) comprende un código de continuación (33a) y un segundo código de longitud (33b), en el que el código de inicio de escape (3 1 a) difiere del código de continuación (33a) o en el que el código de inicio de escape (31a) y el código de continuación (33a) son

idénticos y el primer código de longitud (31b) de longitud mayor señala que el siguiente código en la corriente de datos es un código de continuación (33a).

14. Un programa de ordenador que tiene un código de programa para ejecutar el procedimiento según la reivindicación 11, 12 ó 13, cuando el programa de ordenador se ejecuta en un ordenador.

5 **15.** Corriente de datos que tiene datos de texto (30), definiendo una secuencia de inicio de escape (31) un primer número de unidades de datos (32), el primer número de unidades de datos (32), definiendo una secuencia de continuación de escape (33) un segundo número de unidades de datos, el segundo número de unidades de datos (34), en el que el segundo número de unidades de datos puede ser interpretado por un descodificador de extensión junto con el primer número de unidades de datos (32), en el que la secuencia de inicio de escape (31) comprende un código de inicio de escape (31 a) y un primer código de longitud (31b), y en el que la secuencia de continuación de escape (33) comprende un código de continuación (33a) y un segundo código de longitud (33b), en el que el código de inicio de escape (31a) difiere del código de continuación (33a), o en el que el código de inicio de escape (31a) y el código de continuación (33a) son idénticos y el primer código de longitud (31b) de longitud mayor señala que el siguiente código en la corriente de datos es un código de continuación (33a).

10

15

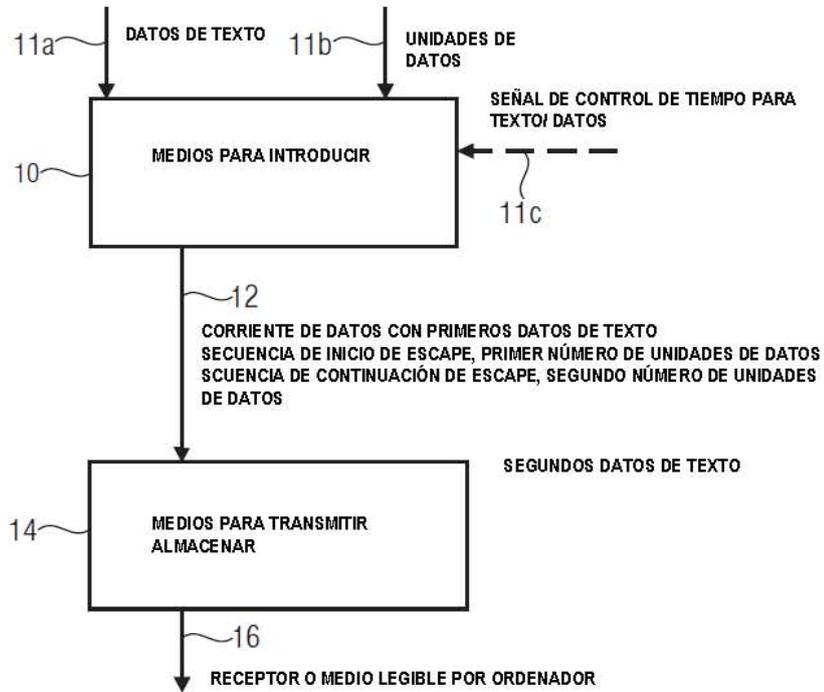


FIGURA 1

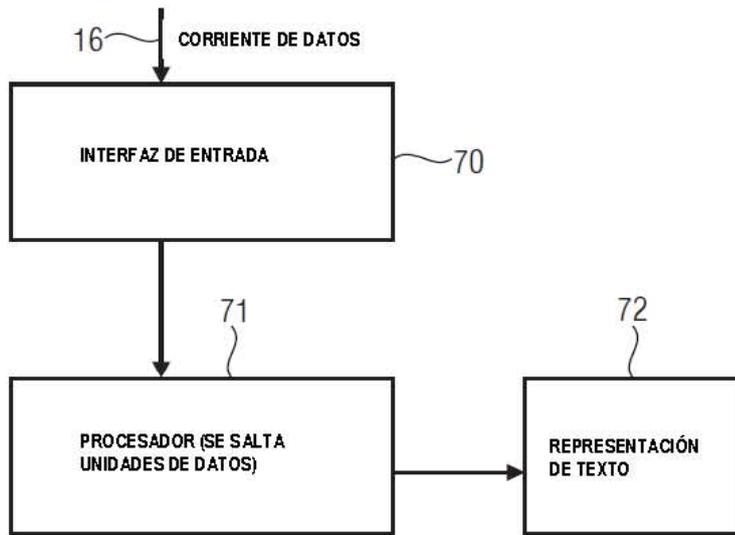


FIGURA 2A
(receptor de base)

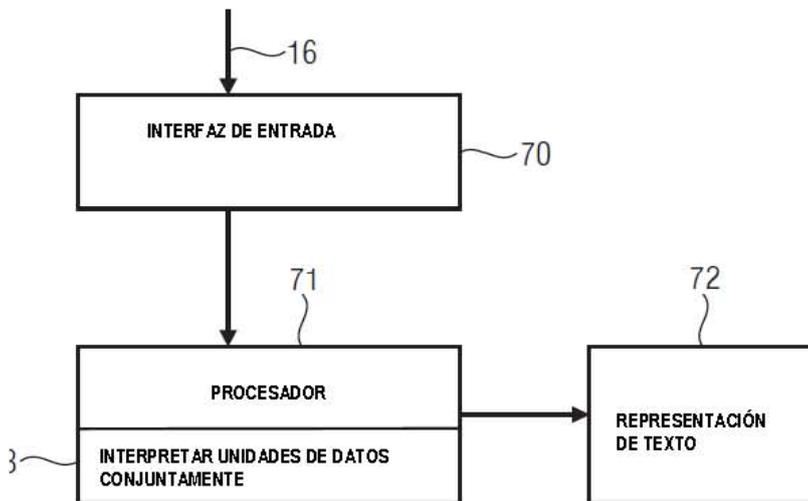
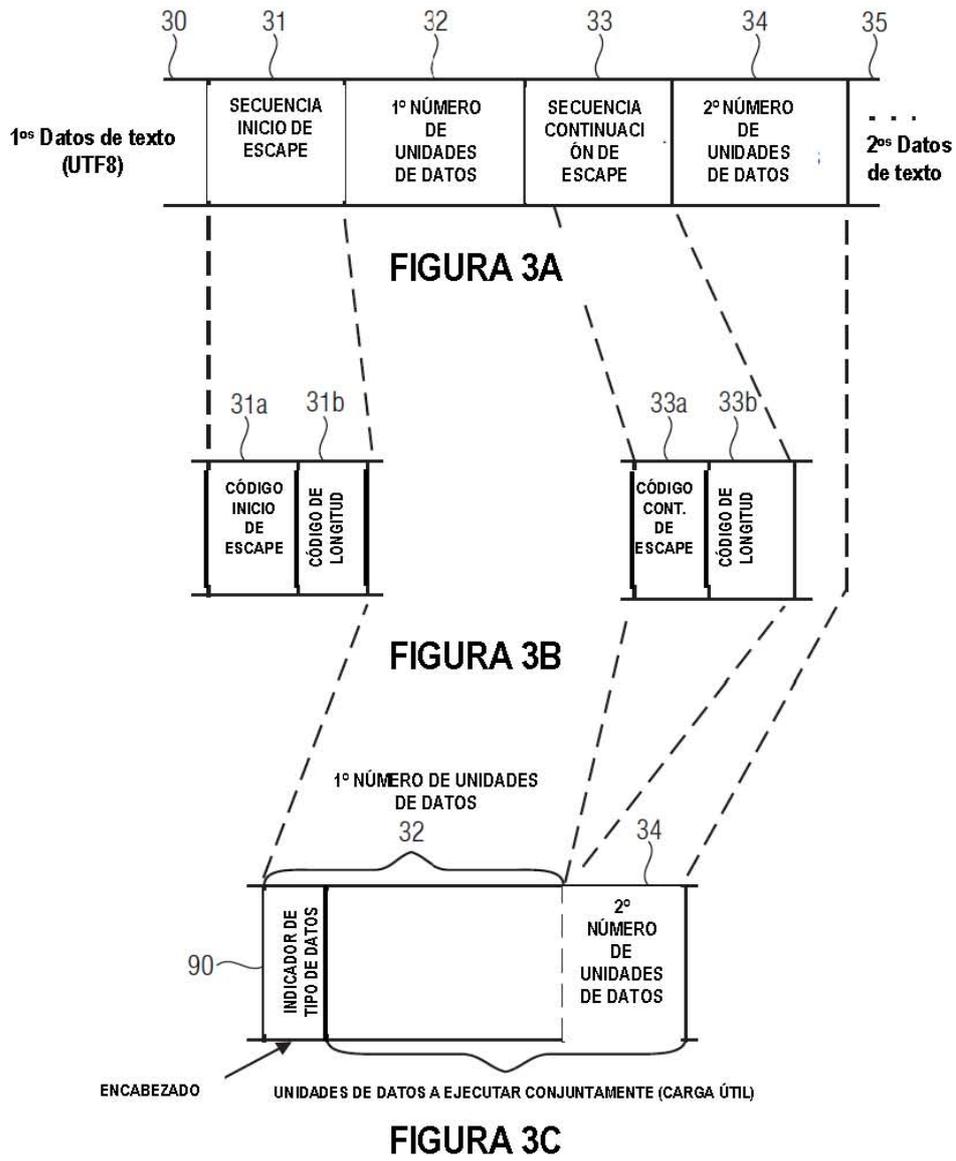


FIGURA 2A
(receptor de extensión)



Representación: This is a great test!

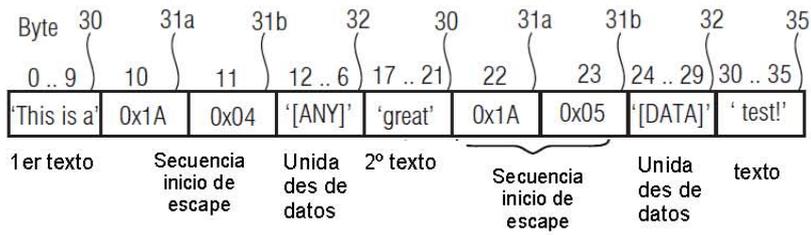


FIGURA 4A

Ejemplo de corriente de datos sin código de continuación

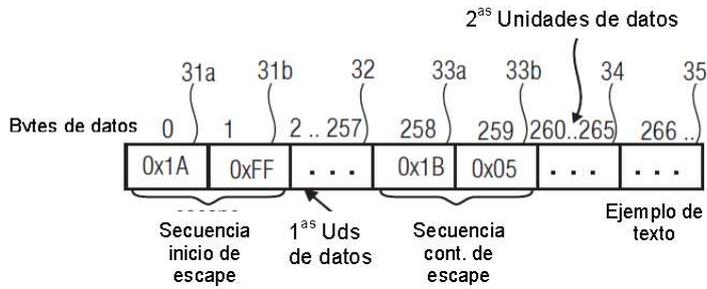


FIGURA 4A

Ejemplo de codificación con código de continuación

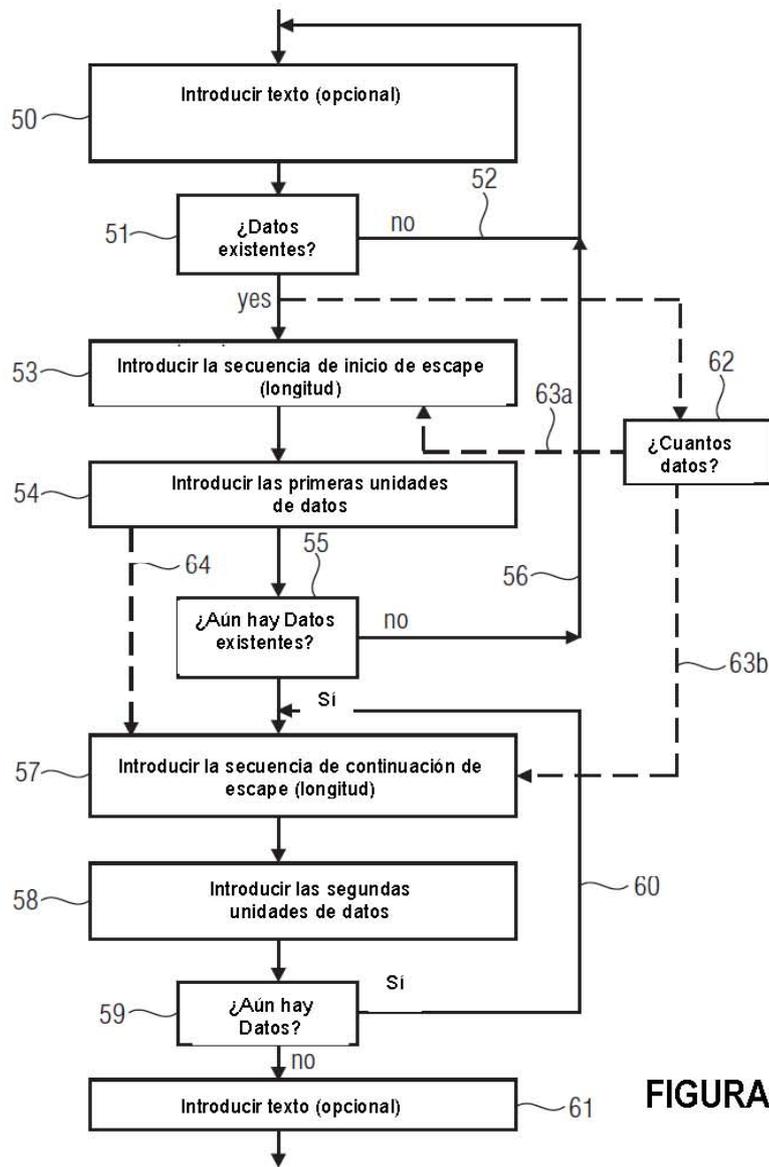
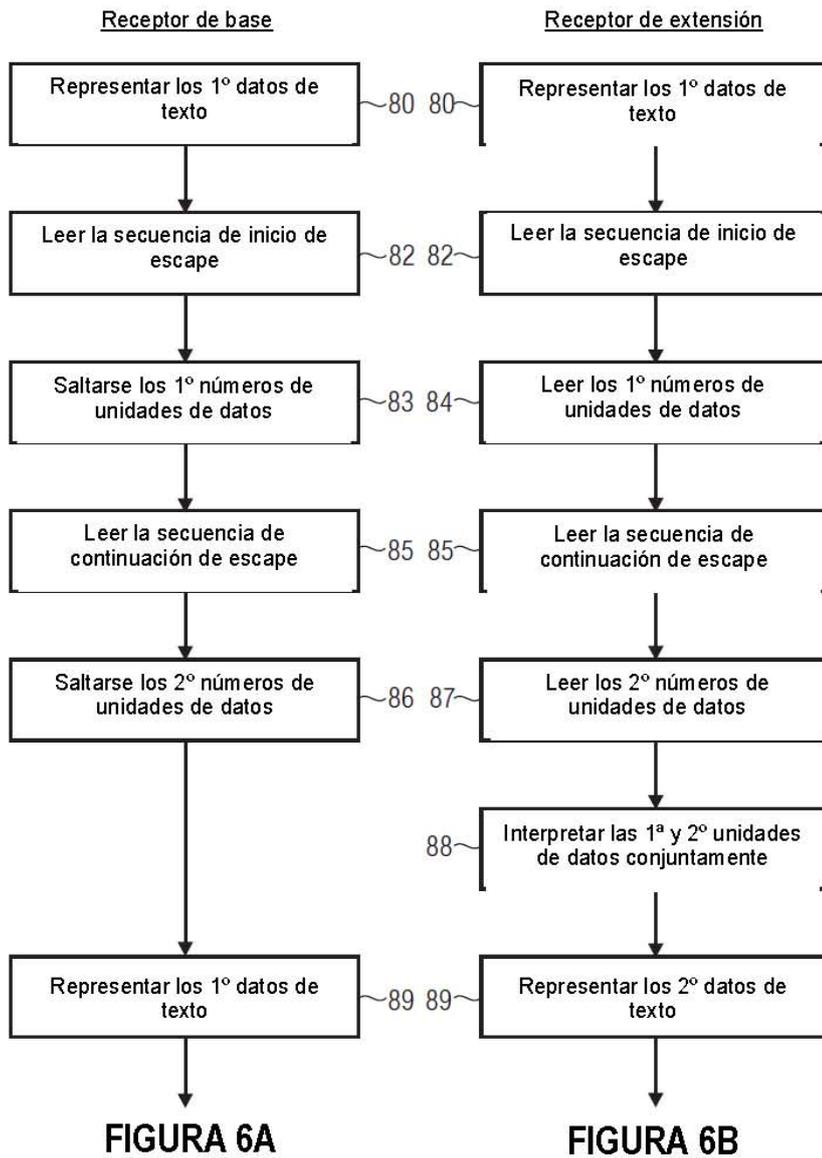


FIGURA 5



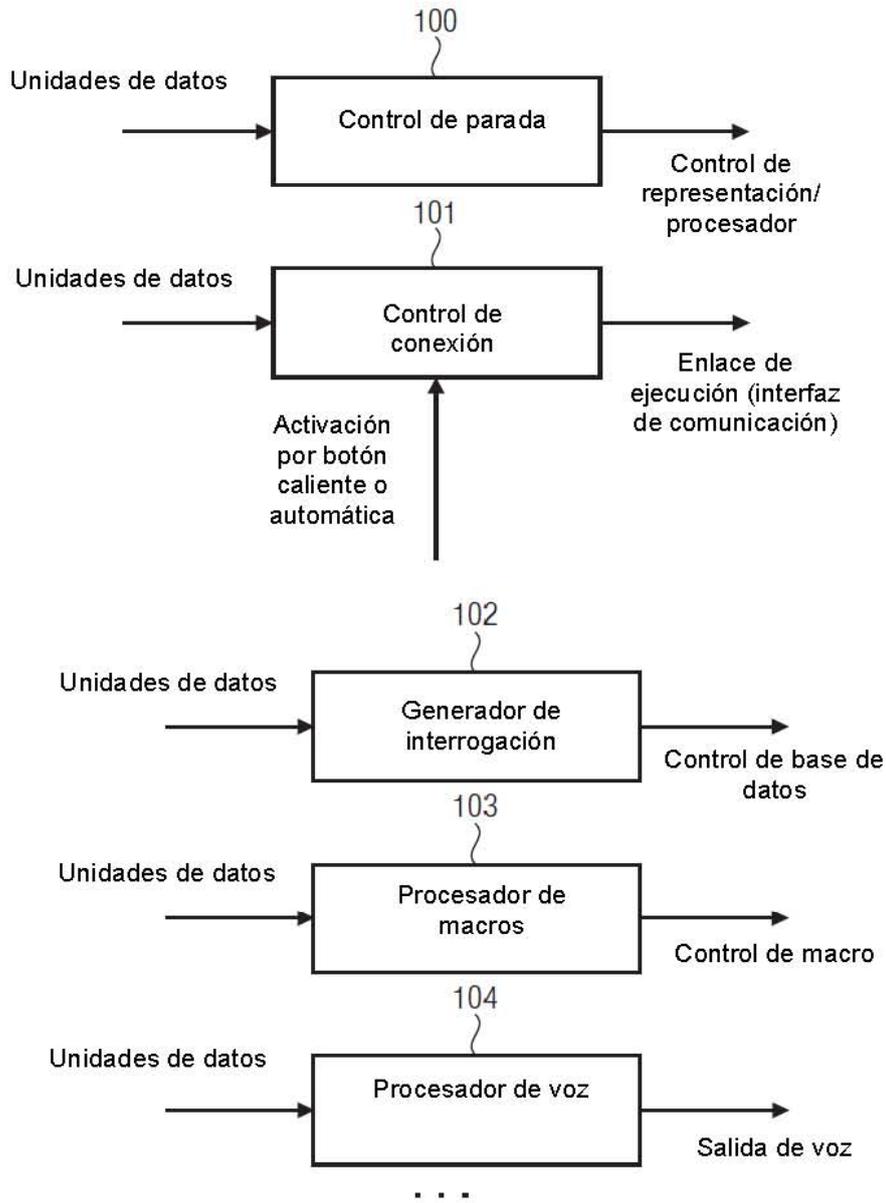


FIGURA 7