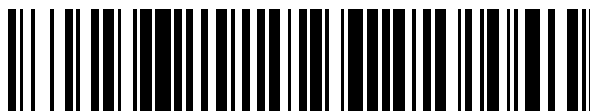


19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 421 515**

51 Int. Cl.:

G11B 27/034 (2006.01)

G11B 27/28 (2006.01)

G11B 27/30 (2006.01)

G11B 27/32 (2006.01)

H04N 21/235 (2011.01)

H04N 21/435 (2011.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

96 Fecha de presentación y número de la solicitud europea: **20.11.2001 E 10000153 (6)**

97 Fecha y número de publicación de la concesión europea: **29.05.2013 EP 2172938**

54 Título: **Aparato y método de reproducción de datos**

30 Prioridad:

08.03.2001 JP 2001065074

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

03.09.2013

73 Titular/es:

**SONY CORPORATION (100.0%)
1-7-1 KONAN MINATO-KU
TOKYO, JP**

72 Inventor/es:

**KATO, MOTOKI y
HAMADA, TOSHIYA**

74 Agente/Representante:

LEHMANN NOVO, María Isabel

ES 2 421 515 T3

Aviso: En el plazo de nueve meses a contar desde la fecha de publicación en el Boletín europeo de patentes, de la mención de concesión de la patente europea, cualquier persona podrá oponerse ante la Oficina Europea de Patentes a la patente concedida. La oposición deberá formularse por escrito y estar motivada; sólo se considerará como formulada una vez que se haya realizado el pago de la tasa de oposición (art. 99.1 del Convenio sobre concesión de Patentes Europeas).

DESCRIPCIÓN

Aparato y método de reproducción de datos

5 CAMPO TÉCNICO

En general, la presente invención se refiere a un aparato de grabación de datos. Más en particular, la presente invención se refiere a un aparato de grabación de datos capaz de gestionar adecuadamente datos memorizados en un medio de grabación y la información de reproducción aún cuando los datos memorizados en el medio de grabación sean objeto de edición.

ANTECEDENTES DE LA INVENCION

15 En los últimos años, una diversidad de discos ópticos se han propuesto como medios de grabación de información del tipo de disco extraíbles desde un aparato de grabación/reproducción. Dichos discos ópticos grabables se proponen como medios que tiene una gran capacidad de almacenamiento de varios gigabytes y siendo adaptables como medios para grabar una señal de AV (audiovisual) tal como una señal de vídeo. Las fuentes que suministran señales de AV digitales, a grabarse en dicho disco óptico susceptible de grabación, incluyen una estación de difusión vía satélite digital CS así como una estación de difusión digital BS y, en el futuro, una estación de difusión de televisión de ondas terrestres propuesta a este respecto.

25 En general, las señales de vídeo digitales suministradas por estas fuentes están sometidas a la compresión de imágenes que suele adoptar una técnica de MPEG (Grupo de Expertos en Imágenes en Movimiento)-2. Un aparato de grabación para grabar las señales tiene una tasa de grabación predeterminada única para ese aparato. Cuando una señal de vídeo digital generada por una estación de difusión digital se graba en el medio de memorización de imagen de consumo convencional, la señal de vídeo digital se decodifica y luego se somete a un proceso de limitación de banda si se adopta una técnica de grabación analógica. Si se adopta una técnica de grabación digital, por el contrario, la señal de vídeo digital es una vez decodificada y luego recodificada a una tasa de grabación peculiar para el aparato de grabación mediante la adopción de una técnica de codificación. Como representativas de la técnica de grabación digital se incluye una técnica de vídeo MPEG1, una técnica de vídeo MPEG2 y una técnica de DV (Video Digital).

35 Con dichas técnicas de grabación, sin embargo, un flujo continuo de bits suministrado se decodifica una vez antes de someterse a los procesos de limitación de banda y de grabación, por lo que se deteriora la calidad de la imagen. Si una tasa de transmisión de una señal digital de entrada, que realiza la compresión de imagen, no excede de una tasa de grabación de un aparato de grabación/reproducción en una operación para grabar la imagen digital, con el fin de reducir al mínimo los deterioros en la calidad de la imagen, es posible adoptar un método para grabar el flujo continuo de vídeo suministrado, tal como está, sin realizar procesos de decodificación y recodificación en la señal digital de entrada. Si la tasa de transmisión de la señal digital de entrada excede la tasa de grabación de un disco utilizado como el medio de grabación, por otro lado, la señal digital necesita decodificarse por el aparato de grabación/reproducción y luego, recodificarse de modo que la tasa de transmisión se haga inferior a un límite superior de la tasa de grabación antes de que la señal se memorice en el medio de grabación.

45 En el caso de una transmisión de señal digital que adopte una técnica de tasa variable, en donde la señal digital se transmita a una velocidad en bits que varía de un momento a otro, un aparato de grabación de disco capaz de grabar datos, que se guardan temporalmente en una memoria intermedia, en una operación de ráfaga, es capaz de utilizar la capacidad de almacenamiento de un disco utilizado como un medio de grabación de información de forma eficiente en comparación con una técnica de grabación de cinta con una tasa de grabación fija debido a una velocidad rotacional fija de una cabeza giratoria.

50 Según se describió anteriormente, en el futuro, en donde la difusión digital sea la más popular, existirá una demanda prevista para un aparato de grabación/reproducción que utilice un disco como un medio de grabación para memorizar una señal difundida como una señal digital sin ningún procesado que incluya procesos de decodificación y recodificación de la misma manera que un dispositivo de flujo continuo de datos (data streamer).

55 Según se describió anteriormente, cuando aumenta la capacidad de almacenamiento de un medio de grabación, el medio de grabación se puede utilizar para memorizar una mayor cantidad de datos, tales como datos de vídeo y datos de audio de un programa. De este modo, un disco se puede utilizar para grabar varios programas. En consecuencia, el usuario necesita realizar operaciones tales como edición para seleccionar solamente los programas deseados seleccionados desde varios programas grabados en el disco.

60 Si se realiza una operación de edición, sin embargo, se hace difícil gestionar adecuadamente los datos grabados en el disco y la información reproducida.

65 La técnica anterior incluye la patente de Estados Unidos US 6.169.843 B1 y de Japón JP 2000 041011A, que dan a conocer, cada una de ellas, un aparato de reproducción en el que las discontinuidades del reloj de tiempo de llegada

(ATC) de un flujo continuo de datos reproducido inicia operativamente una reposición del reloj de tiempo del sistema (STC) del aparato de reproducción.

DESCRIPCIÓN DE LA INVENCIÓN

5 Por lo tanto, es un objetivo de la presente invención resolver los problemas anteriormente descritos para proporcionar un aparato de grabación de datos capaz de gestionar adecuadamente los datos memorizados en un medio de grabación y la información reproducida aún cuando la descripción de los datos memorizados en un medio de grabación sea objeto de edición.

10 Los respectivos aspectos de la presente invención se definen según las reivindicaciones 1 y 2.

BREVE DESCRIPCIÓN DE LOS DIBUJOS

15 La Figura 1 es un diagrama explicativo que ilustra la estructura de un formato de aplicación en un medio de grabación utilizado en un sistema de grabación/reproducción al que se aplica la presente invención.

La Figura 2 es un diagrama explicativo que ilustra la estructura de directorios.

20 La Figura 3 es un diagrama que representa la estructura de un flujo continuo de transporte de DVR MPEG-2.

La Figura 4 es un diagrama que ilustra la sintaxis de paquete fuente.

25 La Figura 5 es un diagrama que ilustra la sintaxis de la cabecera TP_extra_header().

La Figura 6 es un diagrama de bloques que ilustra la configuración de un modelo de grabadora de flujo continuo de transporte DVR MPEG-2.

30 La Figura 7 es un diagrama de bloques que ilustra la configuración de un modelo de un reproductor de flujo continuo de transporte DVR MPEG-2.

La Figura 8 es un diagrama que ilustra la sintaxis de un archivo de información Clip (Fragmento).

35 La Figura 9 es un diagrama explicativo que ilustra la secuencia del ATC.

La Figura 10 es un diagrama explicativo que ilustra una relación entre las discontinuidades del ATC y las secuencias del ATC.

40 La Figura 11 es un diagrama explicativo que ilustra un intervalo de STC continuo.

La Figura 12 es un diagrama explicativo que ilustra una relación entre las discontinuidades de STC y las secuencias de STC.

45 La Figura 13 es un diagrama que ilustra la sintaxis de Información de secuencia SequenceInfo().

La Figura 14 es un diagrama explicativo que ilustra las secuencias de programas.

La Figura 15 es un diagrama que ilustra la sintaxis de información de programa ProgramInfo().

50 La Figura 16 es un diagrama que ilustra la sintaxis de StreamCodingInfo().

La Figura 17 es un diagrama explicativo que ilustra el tipo de codificación del flujo stream_coding_type.

55 La Figura 18 es un diagrama explicativo que ilustra el formato de vídeo, video_format.

La Figura 19 es un diagrama explicativo que ilustra la tasa de tramas frame_rate.

60 La Figura 20 es un diagrama explicativo que ilustra la relación de aspecto de la presentación visual display_aspect_ratio.

La Figura 21 es un diagrama explicativo que ilustra el tipo de presentación de audio, audio_presentation_type.

La Figura 22 es un diagrama explicativo que ilustra la frecuencia de muestreo sampling_frequency.

65 La Figura 23 es un diagrama que ilustra la sintaxis de CPI().

La Figura 24 es un diagrama explicativo que ilustra el mapa mapa_EP.

- La Figura 25 es un diagrama explicativo que ilustra el mapa_TU
- 5 La Figura 26 es un diagrama explicativo que ilustra la sintaxis de mapa_TU.
- La Figura 27 es un diagrama que ilustra la sintaxis del archivo PlayList (ListaReproducción).
- La Figura 28 es un diagrama que ilustra la sintaxis de PlayList().
- 10 La Figura 29 es un diagrama explicativo que ilustra la lista de reproducción (PlayList) de tipo mapa_EP.
- La Figura 30 es un diagrama explicativo que ilustra la lista de reproducción PlayList del tipo mapa_TU.
- 15 La Figura 31 es un diagrama explicativo que ilustra una relación entre la información temporal de PlayList de tipo de mapa_EP e información de dirección en un archivo de flujo continuo AV.
- La Figura 32 es un diagrama explicativo que ilustra una relación entre la información temporal de PlayList de tipo de mapa_TU e información de dirección en un archivo de flujo continuo AV.
- 20 La Figura 33 es un diagrama que ilustra la sintaxis de PlayItem().
- La Figura 34 es un diagrama explicativo que ilustra una relación que se establece entre Clip y PlayList cuando un flujo continuo AV se graba como un nuevo objeto Clip.
- 25 La Figura 35 es un diagrama explicativo que ilustra la creación de una lista de reproducción virtual.
- La Figura 36 es un diagrama explicativo que ilustra una relación que se establece entre Clip y PlayList cuando se elimina una parte de un intervalo de reproducción de la lista de reproducción real, Real PlayList.
- 30 La Figura 37 es un diagrama explicativo que ilustra la edición de minimización.
- La Figura 38 es un diagrama explicativo que ilustra las secuencias de ATC que se forman en Clip cuando se eliminan parcialmente datos de un flujo continuo AV Clip.
- 35 La Figura 39 es un diagrama explicativo que ilustra una relación que se establece entre una secuencia del ATC, una secuencia del STC y una secuencia de programas cuando se elimina parcialmente datos de un flujo continuo AV Clip.
- 40 La Figura 40 es un diagrama explicativo que ilustra una relación que se establece entre Clip y PlayList cuando se elimina una parte del flujo continuo AV Clip que tiene mapa_EP CPI.
- La Figura 41 es un diagrama explicativo que ilustra un caso en el que un archivo Clip (Fragmento) se divide en dos partes cuando se elimina una parte del flujo continuo AV Clip que tiene mapa_EP CPI.
- 45 La Figura 42 es un diagrama explicativo que ilustra una relación que se establece entre Clip y PlayList cuando se elimina una parte del flujo continuo AV Clip que tiene mapa_TU CPI.
- La Figura 43 es un diagrama de bloques que ilustra la configuración de un aparato de grabación/reproducción de imágenes en movimiento de la presente invención.
- 50 La Figura 44 ilustra un diagrama de flujo explicativo que representa operaciones para crear un Clip (Fragmento).
- La Figura 45 ilustra un diagrama de flujo explicativo que representa operaciones para crear información de secuencia SequenceInfo.
- 55 La Figura 46 ilustra un diagrama de flujo explicativo que representa operaciones para crear información del programa ProgramInfo.
- La Figura 47 ilustra un diagrama de flujo explicativo que representa operaciones para crear mapa_EP.
- 60 La Figura 48 ilustra un diagrama de flujo explicativo que representa diferentes métodos de creación de información de Fragmento para diferentes tipos CPI de Fragmento.
- La Figura 49 ilustra un diagrama de flujo explicativo que representa un método de creación de lista de reproducción real, Real PlayList.
- 65 La Figura 50 ilustra un diagrama de flujo explicativo que representa un método de crear una lista de reproducción

virtual, Virtual PlayList.

La Figura 51 ilustra un diagrama de flujo explicativo que representa un método de reproducción de una lista de reproducción del tipo de mapa_EP.

5 La Figura 52 ilustra un diagrama de flujo explicativo que representa un proceso de edición para minimizar la lista de reproducción del tipo de mapa_EP.

La Figura 53 es un diagrama explicativo que ilustra un procesado de minimización.

10 La Figura 54 es un diagrama explicativo que ilustra la eliminación de datos de flujo continuo innecesarios que preceden a IN_time (tiempo_ENTRADA) en un proceso de minimización.

15 La Figura 55 es un diagrama explicativo que ilustra la eliminación de datos de flujo continuo innecesarios después de OUT_time (tiempo_SALIDA) en un proceso de minimización.

La Figura 56 ilustra un diagrama de flujo explicativo que representa un método para reproducir la lista de reproducción del tipo mapa_TU.

20 La Figura 57 ilustra un diagrama de flujo explicativo que representa detalles de procesado en una etapa S303 del diagrama de flujo ilustrado en la Figura 56.

La Figura 58 ilustra un diagrama de flujo explicativo que representa un proceso de edición para minimizar la lista de reproducción del tipo de mapa_TU.

25 La Figura 59 ilustra un diagrama de flujo explicativo que representa detalles de procesado en una etapa S502 del diagrama de flujo ilustrado en la Figura 58.

30 La Figura 60 ilustra un diagrama de flujo explicativo que representa el procesado para actualizar el archivo de información de Fragmento en la edición para minimizar la lista de reproducción del tipo de mapa_EP y la lista de reproducción del tipo de mapa_TU.

35 La Figura 61 ilustra un diagrama explicativo que representa un caso en el que la lista de reproducción del tipo de mapa_EP se divide en dos objetos PlayItem (ElementoReproducción) en un entorno entre dos secuencias de ATC.

La Figura 62 ilustra un diagrama explicativo que representa un caso en el que la lista de reproducción de tipo de mapa_EP se divide en dos objetos PlayItem en un entorno entre dos secuencias STC en una secuencia del ATC continua.

40 La Figura 63 ilustra un diagrama de flujo que representa un proceso para crear la lista de reproducción del tipo de mapa_EP en el procesado para grabar un flujo continuo AV.

45 La Figura 64 es un diagrama explicativo que representa un caso en el que la listas de reproducción de tipo de mapa_TU se divide en dos objetos PlayItem en un entorno entre dos secuencias ATC.

La Figura 65 ilustra un diagrama de flujo que representa un proceso para crear la lista de reproducción de tipo de mapa_TU en el procesado para grabar un flujo continuo AV.

50 La Figura 66 ilustra un diagrama de flujo que representa la reproducción de una lista de reproducción del tipo de mapa_EP.

La Figura 67 ilustra un diagrama de flujo que representa la reproducción de la lista de reproducción del tipo de mapa_TU.

55 La Figura 68 es un diagrama explicativo que ilustra un área de grabación de un medio de grabación.

MEJOR MODO DE PONER EN PRÁCTICA LA INVENCIÓN

60 Formas de realización preferidas de la presente invención se explicarán haciendo referencia al diagrama correspondiente como sigue.

65 La Figura 1 es un diagrama que ilustra una estructura simplificada de un formato de aplicación en un medio de grabación (un medio de grabación 10 en la Figura 43, que se describirá más adelante). El formato tiene dos capas, esto es, PlayList (Lista de Reproducción) y Clip (Fragmento), que se utilizan para gestionar un flujo continuo AV. La información del volumen se utiliza para gestionar todos los objetos Clip y PlayList en el disco.

Un par que consiste en un flujo continuo AV y su información accesoria se considera como siendo un objeto referido como un objeto Clip. Un archivo de flujo continuo AV se denomina un archivo de flujo continuo AV Clip y su información accesoria se conoce como un archivo de información de Fragmento.

- 5 Un archivo del flujo continuo AV Clip se utiliza para memorizar los datos depositados en una estructura que prescribe un flujo continuo de transporte MPEG-2 en un formato de aplicación de DVR (Grabación Vídeo Digital).

10 En general, un archivo de datos utilizado en un aparato, tal como un ordenador, se trata como un conjunto matricial de bytes. Por el contrario, los contenidos de un archivo de flujo continuo AV Clip se expanden a lo largo de un eje de tiempos. Un objeto PlayList especifica los puntos de acceso en el objeto Clip principalmente como marcas de tiempo. Con las marcas de tiempo de los puntos de acceso en un objeto Clip dado por el objeto PlayList, el archivo de información de Fragmento es de utilidad para encontrar una dirección para iniciar una operación para decodificar un flujo en el archivo de flujo continuo AV Clip.

15 El objeto PlayList se introduce para la finalidad de permitir un intervalo de reproducción, que el usuario desea ver, a seleccionarse a partir del contenido del objeto Clip y el intervalo de reproducción a editarse con facilidad. Un objeto PlayList es un conjunto de intervalos de reproducción seleccionados a partir de un objeto Clip. En un objeto PlayList, un intervalo de reproducción en un objeto Clip se denomina PlayItem (ElementoReproducción), que se expresa por un par de puntos IN (ENTRADA) y OUT (SALIDA). De este modo, un objeto PlayList es un conjunto de objetos PlayItem.

Existen dos tipos de PlayList, esto es, PlayList real y PlayList virtual.

25 El PlayList real se puede considerar como un objeto PlayList que comparte partes del flujo continuo con un objeto Clip asociado con el objeto PlayList. Para expresarlo con más detalle, un objeto Real PlayList ocupa el área de disco para memorizar datos asociados con las partes de flujo compartidas con el objeto PlayList. Cuando se crea un flujo continuo AV como un nuevo objeto Clip, un objeto Real PlayList, que se refiere al intervalo de reproducción completo del objeto Clip, es también creado de forma automática. Si una parte del intervalo de reproducción del objeto Real PlayList se elimina, la parte del flujo de los datos del objeto Clip referidos por la parte eliminada del intervalo de reproducción del objeto Real PlayList es también objeto de eliminación.

30 Virtual PlayList puede considerarse como un objeto PlayList que no comparte ninguna parte de flujo con un objeto Clip asociado con el objeto PlayList. Aún cuando se elimine una parte del intervalo de reproducción del objeto Virtual PlayList, el objeto Clip no cambia en absoluto.

35 Conviene señalar que, en la descripción siguiente, Real PlayList y Virtual PlayList se refieren simplemente como PlayList, que es una denominación genérica.

Los directorios requeridos en un disco DVR se enumeran como sigue:

40 un directorio raíz que incluye un directorio "DVR" y el directorio "DVR" que aloja un directorio "PLAYLIST", un directorio "CLIPNF", un directorio "STREAM" y un directorio "DATA".

45 Un directorio distinto a los anteriores puede crearse bajo el directorio raíz. Sin embargo, dicho directorio creado se ignora en el formato de aplicación de DVR.

50 La Figura 2 es un diagrama que representa una estructura de directorios típica en un disco DVR. Según se ilustra en la Figura, el directorio raíz incluye solamente un directorio. Todos los archivos y directorios descritos en conformidad con "DVR", que es un formato de aplicación de DVR, se deben memorizar bajo el directorio de DVR.

El directorio de "DVR" incluye directorios descritos como sigue:

55 "PLAYLIST" es un directorio bajo el que deben colocarse los archivos de base de datos de Real PlayList y de Virtual PlayList. Este directorio debe existir aún cuando no exista ningún objeto PlayList en absoluto.

"CLIPINF" es un directorio bajo el que debe colocarse los archivos de información de Fragmento. Este directorio debe existir aún cuando no exista ningún objeto Clip en absoluto.

60 "STREAM" es un directorio bajo el que debe colocarse archivos de flujo continuo AV. Este directorio debe existir aún cuando no exista ningún archivo de flujo continuo AV en absoluto.

El directorio de "PLAYLIST" se utiliza para memorizar dos tipos de archivos PlayList, esto es, Real PlayList y Virtual PlayList.

65 "xxxx.rpls" es un archivo para memorizar información relacionada con un objeto Real PlayList. Es decir, un archivo "xxxx.rpls" se crea para cada objeto Real PlayList. "xxxx.rpls" es el nombre del archivo. "xxxx" es una cadena de

cinco caracteres numéricos que tienen, cada uno de ellos, un valor en el intervalo de 0 a 9. El nombre del archivo debe tener una extensión "rlps".

5 "yyyyy.vpls" es un archivo para memorizar información relacionada con un objeto Virtual PlayList. Es decir, un archivo "yyyyy.vpls" se crea para cada objeto Virtual PlayList. "yyyyy.vpls" es el nombre del archivo. "yyyyy" es una cadena de cinco caracteres numéricos que tienen, cada uno de ellos, un valor en el intervalo de 0 a 9. El nombre del archivo debe tener una extensión "vpls".

10 El directorio "CLIPNF" incluye tantos archivos de información de Fragmento como archivos de flujo de audio.

"zzzzz.clpi" es un archivo de información de Fragmento para un archivo de flujo continuo AV (un archivo de flujo continuo AV Clip o un archivo de flujo puente – AV Clip). "zzzzz.clpi" es el nombre del archivo. "zzzzz" es una cadena de cinco caracteres numéricos que tienen, cada uno de ellos, un valor en el intervalo de 0 a 9. El nombre del archivo debe tener una extensión "clpi".

15 El directorio "STREAM" se utiliza para alojar archivos de flujo continuo AV.

20 "zzzzz.m2ts" es un archivo de flujo continuo AV gestionado por el sistema DVR. Según se describió anteriormente, un archivo de flujo continuo AV puede ser un archivo de flujo continuo AV Clip o un archivo de flujo continuo AV Bridge-Clip (Puente-Fragmento). "zzzzz.m2ts" es el nombre del archivo. "zzzzz" es una cadena de cinco caracteres numéricos, que tienen, cada uno de ellos, un valor en el intervalo de 0 a 9. El nombre del archivo debe tener una extensión "m2ts".

25 Un archivo de flujos de AV debe tener la misma cadena de cinco caracteres numéricos "zzzzz" como el archivo de información de Fragmento asociado con el archivo de flujo continuo AV.

Puesto que los nombres de otros directorios y otros archivos no se requieren en la explicación de las formas de realización de la presente invención, se omite la descripción de los otros directorios.

30 A continuación, se explica la estructura del archivo de flujo continuo AV. El archivo de flujo continuo AV debe tener la estructura de un flujo continuo de transporte de DVR MPEG2, representado en la Figura 3. Un flujo continuo de transporte DVR MPEG2 tiene las características siguientes:

- 35 1) Un flujo continuo de transporte DVR MPEG2 tiene un número entero de unidades alineadas.
- 2) El tamaño de una unidad alineada es de 6.144 bytes (2.048 X 3 bytes).
- 3) Una unidad alineada se inicia con un primer byte de un paquete fuente.
- 40 4) Los paquetes fuente tienen, cada uno de ellos, una longitud de 192 bytes. Un paquete fuente incluye la cabecera TP_extra_header y un paquete de transporte. La cabecera TP_extra_header tiene una longitud de 4 bytes, mientras que el paquete de transporte tiene una longitud de 188 bytes.
- 45 5) Una unidad alineada incluye 32 paquetes fuente.
- 6) La última unidad alineada de un flujo continuo de transporte DVR MPEG-2 incluye también 32 paquetes fuente.
- 7) Si la última unidad alineada no está rellena completamente con paquetes de transporte del flujo continuo de transporte de entrada, el área de bytes restante debe rellenarse con paquetes fuente, que son, cada uno de ellos, un paquete nulo (un paquete de transporte con un PID de Ox1 FFF).

La sintaxis de un paquete fuente se representa en la Figura 4.

55 TP_extra_header() es una cabecera con una longitud de 4 bytes. transport_packet() es un paquete de transporte MPEG-2 de 188 bytes que está conforme con la norma ISO/IEC 13818-1.

La sintaxis de la cabecera TP_extra_header se ilustra en la Figura 5.

60 copy_permission_indicator es un número entero que representa un límite de copias de una carga útil del paquete de transporte.

arrival_time_stamp es una marca de tiempo que representa un tiempo en el que el paquete de transporte llega a un decodificador (el decodificador es un decodificador de AV 16 en la Figura 43, que se describirá más adelante). arrival_time_stamp es un número entero que tiene un valor especificado por arrival_time_stamp en la ecuación (1) que se describirá más adelante.

La Figura 6 es un diagrama que ilustra un modelo de una grabadora para un flujo continuo de transporte DVR MPEG-2. La grabadora corresponde al aparato de grabación/reproducción de imágenes en movimiento 1, cuya configuración se representa en la Figura 43 según se describirá más adelante. El modelo es un modelo de un concepto que prescribe un proceso de grabación. Un flujo continuo de transporte DVR MPEG-2 debe estar conforme con este modelo.

Una temporización de entrada de un flujo continuo de transporte MPEG-2 se describe como sigue.

- El flujo continuo de transporte de MPEG-2 de entrada es un flujo continuo de transporte completo o un flujo continuo de transporte parcial.
- El flujo continuo de transporte MPEG-2 de entrada debe estar conforme con la norma ISO/IEC 13818-1 o ISO/IEC 13818-9.
- El i-ésimo byte del flujo continuo de transporte MPEG-2 se suministra simultáneamente en un momento $t(i)$ a un T-STD 201 y un empaquetador fuente 204. El T-STD 201 es un decodificador objetivo del sistema de flujo continuo de transporte que está conforme con la norma ISO/IEC 13818-1. El T-STD 201 corresponde al decodificador de AV 16 representado en la Figura 43. El empaquetador fuente 204 es un empaquetador fuente 29 representado en la Figura 43.

Un PLL de 27 MHz 202 corresponde a un componente incorporado en una unidad de control 17 utilizada en el aparato de grabación/reproducción de imágenes en movimiento 1, cuya configuración se representa en la Figura 43. La frecuencia de reloj de 27 MHz debe fijarse con el valor de una PCR (Referencia de Reloj de Programa) del flujo continuo de transporte de MPEG-2.

Un reloj de tiempo de llegada se describe como sigue.

- Un contador de reloj de tiempo de llegada 203 es un contador binario para contar el número de pulsos de 27 MHz, a la salida, por el conmutador PLL de 27 MHz 202. El contador de reloj de tiempo de llegada 203 corresponde a un componente también incorporado en una unidad de control 17 utilizada en el aparato de grabación/reproducción de imágenes en movimiento 1, cuya configuración se representa en la Figura 43.
- $Arrival_time_clock(i)$ es el valor de conteo del contador de reloj de hora de llegada 203 en un momento $t(i)$.

El empaquetador fuente 204 crea un paquete fuente añadiendo la cabecera TP_extra_header a cada paquete de transporte.

- $Arrival_time_stamp$ es una marca de tiempo en la que el primer byte del paquete de transporte llega, a la vez, al T-STD 201 y al empaquetador fuente 204. Como resulta evidente a partir de la ecuación (1), $Arrival_time_stamp(k)$ es un valor de muestra de $arrival_time_clock(k)$.

$$time_stamp(k) = arrival_time_clock(k) \% 2^{30} \dots (1).$$

en donde la notación k indica el primer byte del paquete de transporte.

Una memoria intermedia de escritura 205 corresponde a un componente incorporado en una unidad de escritura 32 utilizada en el aparato de grabación/reproducción de imágenes en movimiento 1, cuya configuración se representa en la Figura 43. R_{max} es una velocidad en bits de entrada de un flujo de paquetes fuente que fluye desde el empaquetador fuente 204 a la memoria intermedia de escritura 205. R_{max} se calcula utilizando la ecuación siguiente:

$$R_{max} = TS_recording_rate \times 192/188$$

en donde la notación $TS_recording_rate$ es la velocidad en bits máxima del flujo continuo de transporte a la entrada.

R_{ud} es una velocidad de bits de salida desde la memoria intermedia de escritura 205 a una unidad DVR 206. La unidad DVR 206 corresponde a un componente también incorporado en la unidad de escritura 32 utilizada en el aparato de grabación/reproducción de imágenes en movimiento 1, cuya configuración se representa en la Figura 43.

- R_{ud} es una velocidad de bits de salida de un flujo de paquetes fuente desde la memoria intermedia de escritura 205 en un estado no vacío de la memoria intermedia de escritura 205. Cuando la memoria intermedia de escritura 205 está vacía, la velocidad en bits de salida es cero.

La unidad DVR 206 graba cada paquete desde la memoria intermedia de escritura 205 en un disco correspondiente a un medio de grabación 10 representado en la Figura 43. Cada paquete incluye un ATS añadido que indica un momento en el que el paquete llega al T-TSD 201.

La Figura 7 es un diagrama que ilustra un modelo de un reproductor para reproducir un flujo continuo de transporte de DVR MPEG-2. El reproductor corresponde al aparato de grabación/reproducción de imágenes en movimiento 1, cuya configuración se representa en la Figura 43. El modelo es un modelo de un concepto que prescribe un proceso de reproducción. Un flujo continuo de transporte de DVR MPEG-2 debe estar conforme con este modelo.

- Una memoria intermedia de lectura 222 corresponde a un componente incorporado en una unidad de lectura 11 utilizada en el aparato de grabación/reproducción de imágenes en movimiento 1, cuya configuración se representa en la Figura 43.

- Rud es una velocidad de bits de entrada desde una unidad DVR 221 a una memoria intermedia de lectura 222. (La unidad DVR 221 corresponde a un componente incorporado a una unidad de lectura 11 utilizada en el aparato de grabación/reproducción de imágenes en movimiento 1, cuya configuración se representa en la Figura 43).

- Rud es una velocidad en bits de entrada de un flujo de paquetes fuente a la memoria intermedia de lectura 222 en un estado no vacío de la memoria intermedia de lectura 222. Cuando la memoria intermedia de lectura 222 está llena, no se suministra ningún flujo continuo a la memoria de lectura 222.

Rmax es una velocidad de bits de salida desde la memoria intermedia de lectura 222 a un desempaquetador fuente 223 que corresponde a un desempaquetador fuente 14 utilizado en el aparato de grabación/reproducción de imágenes en movimiento 1, cuya configuración se representa en la Figura 43.

Un contador de reloj de tiempo de llegada 225 corresponde a un componente incorporado en la unidad de control 17 utilizada en el aparato de grabación/reproducción de imágenes en movimiento 1, cuya configuración se representa en la Figura 43.

El contador de reloj de tiempo de llegada 225 es un contador para contar el número de pulsos de 27 MHz generados por un oscilador de cristal 224 a una frecuencia de 27 MHz. El oscilador de cristal 224 corresponde a un componente también incorporado en la unidad de control 17, que se utiliza en el aparato de grabación/reproducción de imágenes en movimiento 1, cuya configuración se representa en la Figura 43.

- Si el zócalo de conexión fuente presente es el primer zócalo de conexión fuente del archivo de flujo continuo AV o un zócalo de conexión fuente indicado por SPN_ATC_start en SequenceInfo () a describirse más adelante, el contador de reloj de tiempo de llegada 225 es objeto de reposición al valor de la marca de tiempo de llegada del paquete.

arrival_time_clock(i) es el valor de conteo del contador de reloj de tiempo de llegada 225 en un momento t(i).

- Una temporización con la que el flujo continuo de transporte de MPEG-2 es objeto de salida se describe como sigue.

- Si la marca de tiempo de llegada arrival_time_stamp del presente paquete fuente es igual al valor de los 30 bits LSB de arrival_time_clock(i), el paquete de transporte del paquete fuente se extrae desde la memoria intermedia.

La siguiente descripción explica el formato de una base de datos para gestionar la información reproducida de un archivo de flujo continuo AV.

La Figura 8 es un diagrama que ilustra la sintaxis del archivo de información Clip. El archivo de información Clip incluye SequenceInfo(), ProgramInfo() y CPI ().

SequenceInfo_start_address es la dirección de inicio de SequenceInfo() relativa al primer byte del archivo zzzzz.clpi con un byte tomado como una unidad de dirección. Es decir, la primera dirección es una dirección relativa de 0.

ProgramInfo_start_address es la dirección de inicio de ProgramInfo() relativa al primer byte del archivo zzzzz.clpi con un byte tomado como una unidad de dirección. Es decir, la primera dirección es una dirección relativa de 0.

CPI_start_address es la dirección de inicio de CPI() relativa al primer byte del archivo zzzzz.clpi con un byte tomado como una unidad de dirección. Es decir, la primera dirección es una dirección relativa de 0.

Puesto que otros campos de sintaxis no son requeridos en la explicación de las formas de realización de la presente invención, se omite la descripción de los demás campos.

SequenceInfo() define la información de la secuencia del ATC y la secuencia del STC en el flujo continuo AV Clip.

La secuencia del ATC se describe como sigue. Una base de tiempos de llegada es un eje de tiempos basado en las marcas de tiempo de llegada (ATS) de paquetes fuente que están constituidos por un archivo de flujo continuo AV. El reloj a lo largo del eje de tiempos se denomina un ATC (reloj de tiempos de llegada). Una secuencia del ATC es una secuencia de paquetes fuente que no incluye ninguna discontinuidad de ATC (o discontinuidad de la base de tiempos de llegada).

La Figura 9 es un diagrama explicativo que ilustra una secuencia del ATC. Cuando un flujo continuo de transporte de entrada se graba recientemente como un archivo de flujo continuo AV Clip, su objeto de Clip no debe incluir una discontinuidad de ATC y solamente tiene una secuencia del ATC. Se supone que una discontinuidad de ATC se crea solamente cuando los datos del flujo de un archivo de flujo continuo AV Clip se eliminan parcialmente por un proceso tal como edición. Más detalles se proporcionarán más adelante.

La dirección de inicio de una secuencia del ATC, es decir, una dirección en la que se inicia una nueva marca de tiempo ATC en un archivo de flujo continuo AV, se memoriza en SequenceInfo(). Esta dirección se refiere como SPN_ATC_start.

Las secuencias de ATC distintas de la última en un archivo de flujo continuo AV, comienzan, cada una de ellas, en un paquete fuente apuntado, por su SPN_ATC_start y finalizan en un paquete fuente inmediatamente precedente a un paquete fuente apuntado por el siguiente SPN_ATC_start. La última secuencia del ATC comienza en un paquete fuente apuntado por su SPN_ATC_start y finaliza en el último paquete fuente del archivo de flujo continuo AV.

La Figura 10 es un diagrama explicativo que ilustra una relación entre las discontinuidades de ATC y las secuencias de ATC. En este ejemplo, un archivo de flujo continuo AV Clip tiene dos discontinuidades ATC y tres secuencias ATC.

A continuación, se explica la secuencia del STC (Reloj de Tiempos del Sistema). La definición de la secuencia del STC se prescribe en la norma MPEG-2. El STC es un reloj de base de tiempos del sistema, que es un eje de tiempos basado en una PCR (Referencia de Reloj del Programa) en el archivo de transporte. Un valor de STC es un valor de conteo de un contador binario de 33 bits que tiene una precisión de 90 kHz.

La Figura 11 es un diagrama explicativo que ilustra un intervalo STC continuo. El eje horizontal es el reloj de tiempos de llegada y el eje vertical es el STC (o base de tiempos del sistema). En el Caso-1, el STC está aumentando de forma monótona y el intervalo no contiene ninguna discontinuidad de STC. En el Caso-2, el contador STC de 33 bits del terminal de entrada digital está envuelto alrededor en la parte media. El punto de la envolvente del STC no es una discontinuidad. El STC es continuo aún cuando el contador STC funcione en el modo envolvente.

Una discontinuidad de STC se genera cuando una estación de difusión cambia el sistema de transmisión desde uno a otro, cuando la unidad grabadora cambia el canal de uno a otro, cuando el usuario realiza una operación de edición o en otros eventos operativos.

Una secuencia del STC es una secuencia de paquetes fuente que no incluye ninguna discontinuidad de STC (o ninguna discontinuidad de la base de tiempos del sistema). Conviene señalar que el mismo valor de STC aparece, ni mucho menos, más de una vez en la misma secuencia del STC. En consecuencia, la duración máxima de un objeto Clip no puede establecerse a un valor superior al del periodo de envolvente de 33 bits de STC (aproximadamente 26 horas).

La dirección de inicio de una secuencia del STC, es decir, una dirección en la que se inicia un nuevo STC, en un archivo de flujo continuo AV, se memoriza en SequenceInfo(). Esta dirección se refiere como SPN_STC_start.

Una secuencia del STC nunca se extiende a través de un contorno límite de una secuencia del ATC.

Las secuencias STC que no sean la última en un archivo de flujo continuo AV comienzan, cada una de ellas, en un paquete fuente apuntado por su SPN_STC_start y finalizan en un paquete fuente que precede inmediatamente a un paquete fuente apuntado por el siguiente SPN_STC_start. La última secuencia del STC comienza en un paquete fuente apuntado por su SPN_STC_start y finaliza en el último paquete fuente del archivo de flujo continuo AV.

La Figura 12 es un diagrama explicativo que ilustra una relación entre discontinuidades de STC y secuencias STC así como una relación entre secuencias STC y secuencias ATC. En este ejemplo, un archivo de flujo continuo AV Clip tiene tres STCs y tres secuencias STC. Una secuencia del STC nunca se extiende sobre un contorno límite de una secuencia del ATC.

Si un archivo de flujo continuo AV tiene discontinuidades de STC, un PTS puede aparecer más de una vez con el mismo valor en el archivo de flujo continuo AV. Por este motivo, cuando se utiliza el PTS como una base para apuntar a un momento determinado en un flujo continuo AV, el PTS de un punto de acceso solamente no es suficiente para identificar el punto. Además de PTS, se requiere el índice de una secuencia del STC que incluye el PTS. El índice se refiere como STC_id.

La Figura 13 es un diagrama que ilustra la sintaxis de SequenceInfo().

5 Longitud es el número de bytes que comienza con un byte inmediatamente después de que este campo de longitud y finaliza con el último byte de SequenceInfo().

num_of_ATC_sequences es el número de secuencias ATC en el archivo de flujo continuo AV.

10 SPN_ATC_start[atc_id] es una dirección en la que la secuencia del ATC apuntada por atc_id se inicia en el archivo de flujo continuo AV. SPN_ATC_start[atc_id] es un valor de conteo con un valor inicial de 0 correspondiente al primer paquete fuente del archivo de flujo continuo AV. El valor de conteo se incrementa en 1 para cada paquete fuente.

15 SPN_ATC_start[0] al principio de SequenceInfo() es cero. Los valores de SPN_ATC_start[atc_id] introducidos en SequenceInfo() están dispuestos en un orden ascendente. Es decir, SPN_ATC_start[atc_id] introducido en SequenceInfo() satisface la condición siguiente.

SPN_ATC_start[0] = 0

20 Para $0 < \text{atc_id} < \text{numero de secuencias ATC}$,

SPN_ATC_start[atc_id - 1] < SPN_ATC_start[atc_id]

número de secuencias de STC [atc_id] es el número de secuencias STC en la secuencia del ATC apuntada por atc_id.

25 offset_STC_id[atc_id] es el valor de un desplazamiento, offset, relativo a stc_id para la primera secuencia del STC en una secuencia del ATC apuntada por atc_id. Cuando un archivo de flujo continuo AV está recientemente grabado, offset_STC_id[atc_id] es cero.

30 El valor de stc_id para la primera secuencia del STC en una secuencia del ATC apuntada por atc_id se define por un orden prescrito como for_loop de stc_id en la sintaxis y comienza desde offset_STC_id[atc_id].

35 Para dos secuencias ATC consecutivas definidas en SequenceInfo(), stc_id para la última secuencia del STC en la anterior secuencia del ATC puede tener el mismo valor que stc_id para la primera secuencia del STC en la última secuencia del ATC. Si estos dos valores de stc_id son iguales entre sí, el mismo valor STC no aparecerá en las dos secuencias STC referidas por los dos valores de stc_id.

40 Los valores de stc_id introducidos en SequenceInfo() deben disponerse en un orden ascendente. offset_STC_id[atc_id] se establece en un valor tal que se satisfaga esta restricción.

PCR_PID[atc_id][stc_id] es el valor del PID de un paquete de transporte que tiene un PCR válido en una secuencia del STC apuntada por stc_id y situada en una secuencia del ATC apuntada por atc_id.

45 SPN_STC_start[atc_id][stc_id] es una dirección en la que una secuencia del STC apuntada por stc_id y situada en una secuencia del ATC apuntada por atc_id comienza en el archivo de flujo continuo AV.

SPN_STC_start[atc_id][stc_id] es un valor de conteo con un valor inicial de 0 correspondiente al primer paquete fuente del archivo de flujo continuo AV. El valor de conteo se incrementa en 1 para cada paquete fuente.

50 Los valores de SPN_STC_start[atc_id][stc_id] introducidos en SequenceInfo() deben disponerse en un orden ascendente. El primer valor de SPN_STC_start[atc_id][stc_id] en una secuencia del ATC apuntada por atc_id debe ser al menos igual a SPN_ATC_start[atc_id] o más. Es decir, debe satisfacerse la condición siguiente:

SPN_ATC_start[atc_id] \leq SPN_STC_start[atc_id][0]

55 presentation_start_time[atc_id][stc_id] es un tiempo de inicio de la presentación de los datos del flujo continuo AV en una secuencia del STC apuntada por stc_id y situada en una secuencia del ATC apuntada por atc_id. Este tiempo de inicio de presentación se deriva del STC de la secuencia del STC y tiene el periodo de una señal de 45 kHz utilizada como una unidad.

60 Por otro lado, presentation_end_time[atc_id][stc_id] es un tiempo de final de presentación de datos de flujo continuo AV en una secuencia del STC apuntada por stc_id y localizada en una secuencia del ATC apuntada por atc_id. Este tiempo de final de presentación se deriva del STC de la secuencia del STC y tiene el periodo de una señal de 45 kHz utilizada como una unidad.

65 A continuación, se explica el significado de ProgramInfo(). Un programa es un conjunto de flujos elementales. Para la

finalidad de una reproducción síncrona de estos flujos, los flujos comparten solamente una base de tiempos del sistema.

Para un aparato de reproducción correspondiente a la grabación/reproducción de imágenes en movimiento, el aparato de grabación/reproducción de imágenes en movimiento 1, ilustrado en la Figura 43 a describirse más adelante, es de utilidad un análisis del contenido de un flujo continuo AV antes del procesado para decodificar el flujo continuo AV. Los contenidos son información tal como el valor del PID de un paquete de transporte utilizado para transmitir un flujo elemental de vídeo o de audio y el tipo de componente del vídeo o audio. (Por ejemplo, el tipo de componente indica que el flujo elemental es un flujo continuo de vídeo de HDTV o un flujo de audio MPEG-2 AAC).

Esta información es de utilidad para la creación de una pantalla de menú utilizada para proporcionar al usuario una explicación del contenido de la lista de reproducción PlayList que se refiere al flujo continuo AV. Además, esta información es también de utilidad para el establecimiento de los estados inicial del decodificador de AV 16 y el demultiplexor 15 del aparato de grabación/reproducción de imágenes en movimiento, que se describirá más adelante haciendo referencia a la Figura 43. Por este motivo, el archivo de información de Fragmento incluye ProgramInfo que se utiliza para explicar el contenido de un programa.

El contenido de un programa en un archivo de flujo continuo AV utilizado para memorizar un flujo continuo de transporte de MPEG-2 puede cambiar. Por ejemplo, el PID de un paquete de transporte, para transmitir un flujo elemental de vídeo, cambia o el tipo de componente del flujo continuo de vídeo cambia desde SDTV a HDTV. El parámetro ProgramInfo se utiliza para memorizar información en puntos de cambio del contenido de un programa en el archivo de flujo continuo AV.

Una secuencia de paquetes fuente con contenidos de programa fijos, prescritos por el formato en el archivo de flujo continuo AV, se refiere como una secuencia de programa.

Una dirección en la que se inicia una nueva secuencia de programa en el archivo de flujo continuo AV se memoriza en ProgramInfo() como SPN_program_sequence_start.

Las secuencias de programas distintas de la última en un archivo de flujo continuo AV comienzan, cada una de ellas, en un paquete fuente apuntado por su SPN_program_sequence_start y finalizan en un paquete fuente que precede inmediatamente a un paquete fuente apuntado por el siguiente SPN_program_sequence_start. La última secuencia del programa comienza en un paquete fuente apuntado por su SPN_program_sequence_start y finaliza en el último paquete fuente del archivo de flujo continuo AV.

La Figura 14 es un diagrama explicativo que ilustra secuencias de programas. En este ejemplo, el archivo de flujo continuo AV Clip tiene tres secuencias de programas.

Una secuencia de programas puede extenderse sobre los contornos límite de la secuencia del ATC y de la secuencia del STC.

La Figura 15 es un diagrama que ilustra la sintaxis de ProgramInfo().

Longitud es el número de bytes que comienzan con un byte inmediatamente después de este campo de longitud y finaliza con el último byte de ProgramInfo().

num_of_program_sequences es el número de secuencias de programas en el archivo de flujo continuo AV.

SPN_pogram_sequence_start es una dirección en la que se inicia la secuencia de programas en el archivo de flujo continuo AV. SPN_program_sequence_start es un valor de conteo con un valor inicial de 0 correspondiente al primer paquete fuente del archivo de flujo continuo AV. El valor de conteo se incrementa en 1 para cada paquete fuente. Los valores de SPN_program_sequence_start introducidos en ProgramInfo() están dispuestos en un orden ascendente.

SPN_pogram_sequence_start se supone que apunta a un paquete fuente que tiene un primer PMT para su secuencia de programas. SPN_pogram_sequence_start se crea por la grabadora para grabar datos analizando PSI/SI en el flujo continuo de transporte. La grabadora corresponde al aparato de grabación/reproducción de imágenes en movimiento 1, cuya configuración se representa en la Figura 43. Para ser más concretos, una unidad de análisis de vídeo 24 o una unidad de análisis de flujos multiplexados 26 en el aparato de grabación/reproducción de imágenes en movimiento 1, cuya configuración se representa en la Figura 43, analiza el PSI/SI, que causa un retardo a la detección de un cambio en PSI/SI. De este modo, SPN_program_sequence_start puede apuntar a un paquete fuente dentro de un periodo predeterminado que comienza en el cambio real en PSI/SI.

program_map_PID es el valor del PID de un paquete de transporte que tiene una PMT (tabla de mapas de programas) aplicable a la secuencia de programas.

num_of_stream_in_ps es el número de flujos elementales definidos en la secuencia de programas.

5 num_of_groups es el número de grupos que incluye, cada uno de ellos, flujos elementales definidos en la secuencia de programas. num_of_groups tiene un valor al menos igual a 1 o más. Si el PSI/SI del flujo continuo de transporte tiene información sobre grupos que incluyen, cada uno de ellos, flujos elementales, num_of_groups se supone que tiene un valor al menos igual a 1. Cada uno de los grupos forma una vista en un programa de vista múltiple.

stream_PID es el valor de un PID para un flujo elemental definido en un PMT referido por program_map_PID de la secuencia de programas.

10 StreamCodingInfo() es información sobre un flujo elemental apuntado por stream_PID anteriormente descrito.

num_of_streams_in_group es el número de flujos elementales incluidos en un solo grupo que comprende flujos elementales.

15 stream_index es un valor que indica un flujo elemental incluido en un grupo de flujos elementales. El valor es igual al valor de stream_index definido en un for-loop en esta sintaxis.

La Figura 16 es un diagrama que ilustra la sintaxis de StreamCodingInfo().

20 Longitud es el número de bytes que comienza con un byte inmediatamente después de este campo de longitud y que finaliza con el último byte de StreamCodingInfo().

stream_coding_type es el tipo de codificación de un flujo elemental apuntado por stream_PID para este StreamCodingInfo(). Los significados de valores establecidos en stream_coding_type se ilustran en la Figura 17.

25 video_format es el formato de vídeo de un flujo continuo de vídeo apuntado por stream_PID para este StreamCodingInfo(). Los significados de los valores establecidos en video_format se ilustran en la Figura 18.

30 frame_rate es la tasa de tramas de un flujo continuo de vídeo apuntado por stream_PID para este StreamCodingInfo(). Los significados de los valores establecidos en frame_rate se ilustran en la Figura 19.

display_aspect_ratio es la relación de aspecto de presentación visual de un flujo continuo de vídeo apuntado por stream_PID para este StreamCodingInfo(). Los significados de los valores establecidos en video_aspect_ratio se ilustran en la Figura 20.

35 cc_flag es un indicador de si una señal de datos de subtítulos ocultos en un flujo continuo de vídeo apuntado por stream_PID para este StreamCodingInfo() se ha codificado, o no.

40 original_video_format_flag es un indicador de si original_video_format y original_display_aspect_ratio existen, o no, en StreamCodingInfo().

45 original_video_format es información sobre el formato de vídeo original antes de un proceso para codificar un flujo continuo de vídeo apuntado por stream_PID para este StreamCodingInfo(). Los significados de valores establecidos en original_video_format son los mismos que los de video_format anteriormente descritos.

50 original_display_aspect_ratio es la relación de aspecto de presentación visual original para un proceso para codificar un flujo continuo de vídeo apuntado por stream_PID para este StreamCodingInfo(). Los significados de los valores establecidos en original_display_aspect_ratio son los mismos que para display_aspect_ratio anteriormente descritos.

55 En el procesado para transcodificar un flujo continuo de transporte que incluye un flujo continuo de vídeo multiplexado con un flujo continuo de datos multimedia tal como un flujo de BML y subtítulos, el flujo continuo de vídeo se recodifica de modo que cambie su formato de vídeo (por ejemplo, desde 1080i a 480i). Sin embargo, existe un caso concebible en el que el flujo continuo de datos multimedia sigue siendo el mismo que su flujo original.

60 En tal caso, es probable que resulte una desadaptación de información entre el flujo continuo de vídeo recientemente recodificado y el flujo continuo de datos multimedia. Por ejemplo, el formato de vídeo del nuevo flujo continuo de vídeo recodificado cambia a pesar del hecho de que los parámetros de presentación visual del flujo continuo de datos multimedia se hayan determinado suponiendo el formato de vídeo del flujo continuo de vídeo original.

65 original_video_format y original_display_aspect_ratio que se utilizan, cada uno de ellos, para retener información sobre el flujo continuo de vídeo original. El aparato de reproducción crea una imagen a visualizarse desde el nuevo flujo continuo de vídeo recodificado y el flujo continuo de datos multimedia como sigue.

El flujo continuo de vídeo está sujeto a un proceso de muestreo ascendente para obtener un formato de vídeo

indicado por `original_video_format` y `original_display_aspect_ratio`. A continuación, una imagen obtenida como un resultado del proceso de muestreo ascendente se sintetiza con el flujo continuo de datos multimedia para obtener una imagen correcta para visualizarse.

5 `audio_presentation_type` es el tipo de presentación del flujo de audio apuntado por `stream_PID` para este `StreamCodingInfo()`. Los significados de los valores establecidos en `audio_presentation_type` se ilustran en la Figura 21.

10 `sampling_frequency` es la frecuencia de muestreo de un flujo de audio apuntado por `stream_PID` para este `StreamCodingInfo()`. Los significados de valores establecidos en `sampling_frequency` se ilustran en la Figura 22.

A continuación, se explica el significado de `CPI()`. La `CPI` (Información de Puntos Característicos) se utiliza para relacionar la información de tiempos de reproducción, en el flujo continuo AV, con las direcciones en el archivo de flujos de AV.

15 Existen dos tipos de `CPI`, a saber, `mapa_EP` y `mapa_TU`. Si el tipo de `CPI-type` en `CPI()` es el tipo de `mapa_EP`, `CPI()` incluye `mapa_EP`. Si el tipo de `CPI`, en `CPI()` es el tipo de `mapa_TU`, por el contrario, `CPI ()` incluye `mapa_TU`. Un archivo de flujo continuo AV tiene un `mapa_EP` o un `mapa_TU`.

20 `mapa_EP` es una lista de puntos de entrada (EPs). Dichos puntos de entrada se extraen desde flujos elementales y desde flujos de transporte. De este modo, la lista tiene información de dirección para encontrar el punto de entrada en el que ha de iniciarse la decodificación en el flujo continuo AV. Un elemento de datos de EP es un par de una marca de tiempo de presentación (PTS) y una dirección de datos de la unidad de acceso del flujo continuo AV correspondiente a la marca de tiempo PTS.

25 `mapa_EP` se utiliza para dos fines. En primer lugar, `mapa_EP` se utiliza para encontrar una dirección de datos de la unidad de acceso del flujo continuo AV referido utilizando una lista de reproducción `PlayList` de presentación. En segundo lugar, `mapa_EP` se utiliza para una reproducción de avance rápido y una reproducción de retroceso rápido. Si la sintaxis de un flujo continuo AV de entrada puede analizarse en una operación realizada por el aparato de grabación para grabar el flujo continuo AV de entrada, `mapa_EP` se crea y memoriza en el disco.

30 `mapa_TU` es una lista de unidades de tiempo (TU) basadas en las horas de llegada de los paquetes de transporte que entran a través de una interfaz digital. `mapa_TU` ilustra, de este modo, una relación entre las horas de llegada y las direcciones de datos en el flujo continuo AV. Si la sintaxis de un flujo continuo AV de entrada no se puede analizar en una operación realizada por el aparato de grabación para grabar el flujo continuo AV de entrada, `mapa_TU` se crea y memoriza en el disco.

La Figura 23 es un diagrama que ilustra la sintaxis de `CPI ()`.

40 Longitud es el número de bytes que comienza con un byte inmediatamente siguiente a este campo de longitud y que finaliza con el último byte de `CPI()`.

`CPI_type` es un indicador de 1 bit para indicar el tipo de la `CPI` de Clip.

45 `mapa_EP` tiene datos descritos a continuación para un flujo continuo de vídeo en el archivo del flujo continuo AV.

(1): `stream_PID` es el PID de un paquete de transporte para transmitir el flujo continuo de vídeo.

50 (2): `num_EP_entries` es el número de puntos de entrada para el flujo continuo de vídeo.

`mapa_EP` incluye datos de pares que consisten, cada uno de ellos, en `PTS_EP_start` y `SPN_EP_start` tantos como número de entradas EP.

55 (3): `PTS_EP_start` es el PTS de una unidad de acceso que comienza desde una cabecera de secuencia en el flujo continuo de vídeo.

60 (4): `SPN_EP_start` es una dirección en el archivo de flujo continuo AV. La dirección indicada por `SPN_EP_start` es la dirección de paquete fuente que incluye el primer byte de una unidad de acceso referida por `PTS_EP_start` anteriormente descrita. `SPN_EP_start` es un número incrementado en 1 para cada paquete fuente. `SPN_EP_start` tiene un valor inicial de 0 correspondiente al primer paquete fuente en el archivo de flujo continuo AV.

Si una pluralidad de flujos de vídeo existe en el archivo de flujo continuo AV, `mapa_EP` puede incluir los datos anteriormente descritos para cada flujo continuo de vídeo.

65 La Figura 24 es un diagrama explicativo que ilustra un ejemplo de `mapa_EP`. En este ejemplo, un flujo continuo de vídeo con `stream_PID = x` existe en el flujo continuo AV de Clip y existen `k` puntos de entrada (`num_EP_entries = k`).

Un ejemplo de un paquete fuente apuntado por SPN_EP_start se ilustra en la Figura. Una carga útil después de la cabecera TP_header de un paquete de transporte en cada paquete fuente comienza con una cabecera de paquete de PES. La cabecera de paquete de PES es seguida por una cabecera de secuencia (SQH), que se sigue por una cabecera GOP (GOPH). La cabecera GOPH está seguida por una imagen de I (I-PICH). El PTS de una unidad de acceso, que comienza con esta cabecera de secuencia, se codifica en la cabecera de paquete de PES.

A continuación, se explica el significado de mapa_TU.

La Figura 25 es un diagrama explicativo que ilustra mapa_TU, que se crea cuando un flujo continuo AV está recientemente grabado como Clip. Un eje de tiempos creado sobre la base de las horas de llegada de paquetes fuente, incluidos en una secuencia del ATC, se divide en unidades de tiempo predeterminadas. Estas unidades de tiempo son cada una de ellas referida como time_unit.

La dirección del primer paquete fuente de forma completa en el archivo de flujo continuo AV que entra en cada time_unit se memoriza en mapa_TU. La dirección se refiere como SPN_time_unit_start. La definición de un tiempo en una secuencia del ATC está basada en TU_unit_start. Esto se explica más adelante en términos de la semántica de SPN_time_unit_start.

La Figura 26 es un diagrama explicativo que ilustra la sintaxis de mapa_TU.

time_unit_size es el tamaño de una unidad de tiempo. La unidad del tamaño es el periodo de una señal de reloj de 45 kHz derivada de una señal de reloj de tiempos de llegada con una precisión de 27 MHz.

El valor del número de secuencias de ATC utilizado en un for-loop de atc_id, en la sintaxis, se define en SequenceInfo ().

offset_arrival_time[atc_id] es un tiempo de desplazamiento para una primera unidad de tiempo completo en una secuencia del ATC apuntada por atc_id. La unidad del tiempo de desplazamiento es el periodo de una señal de reloj de 45 kHz derivada de una señal de reloj de tiempos de llegada con una precisión de 27 MHz.

Cuando un flujo continuo AV se graba recientemente como Clip, el archivo de flujo continuo AV tiene solamente una secuencia del ATC y offset_arrival_time[atc_id] es cero.

Cuando una pluralidad de elementos de offset_arrival_time[atc_id] se graba para mapa_TU como entradas, se satisfacen las condiciones siguientes:

offset_arrival_time[0] = 0

Para atc_id que tenga un valor de $0 < \text{atc_id} < \text{número de secuencias de ATC}$, $\text{offset_arrival_time}[\text{atc_id}] > \text{offset_arrival_time}[\text{atc_id} - 1] + \text{time_unit} * \text{num_of_time_unit_entries}[\text{atc_id} - 1]$

en donde la notación num_of_time_unit_entries[atc_id] es el número de entradas de unidades de tiempo incluidas en una secuencia del ATC apuntada por atc_id.

SPN_time_unit_start[atc_id][i] es la dirección de inicio de la i-ésima unidad de tiempo en una secuencia del ATC apuntada por atc_id. SPN_time_unit_start[atc_id][i] es también un valor de conteo que se inicia a partir de un valor inicial de 0 correspondiente al primer paquete fuente en el archivo de flujo continuo AV. El valor de conteo se incrementa en 1 para cada paquete fuente.

Si ningún paquete fuente entra en la unidad de tiempo presente, el valor de SPN_time_unit_start para la presente unidad de tiempo es igual al valor de SPN_time_unit_start para la unidad de tiempo inmediatamente precedente.

Las entradas de los valores APN_time_unit_start en mapa_TU deben disponerse en un orden ascendente.

El tiempo de inicio de la i-ésima unidad de tiempo en una secuencia del ATC apuntada por atc_id se requiere como TU_start_time[atc_id][i], que se expresa como sigue:

$\text{TU_start_time}[\text{atc_id}][i] = \text{offset_arrival_time}[\text{atc_id}] + i * \text{time_unit_size} \dots (2)$

La Figura 27 es un diagrama que ilustra la sintaxis del archivo PlayList. Un archivo PlayList tiene PlayList().

PlayList_start_address es la dirección de inicio de PlayList() relativa al primer byte del archivo PlayList. La dirección de inicio se obtiene contando el número de bytes comenzando con el primer byte del archivo PlayList como un valor de conteo de 0.

Puesto que los otros campos de sintaxis no se requieren en la explicación de las formas de realización de la

presente invención, se omite la descripción de los demás campos.

La Figura 28 es un diagrama que ilustra la sintaxis de `PlayList()`.

5 Longitud es el número de bytes comenzando con un byte inmediatamente después de este campo de longitud y finalizando con el último byte de `PlayList()`.

`CPI_type` es un indicador de 1 bit para indicar el valor de `CPI_type` de Clip utilizado en `PlayItem ()`. `CPI_type` se define por el tipo de CPI del archivo de información de Fragmento.

10 `number_of_PlayItems` es el número de `PlayItem()` en `PlayList()`.

El valor de `PlayItem_id` para `PlayItem()` es el número de una iteración de bucle en el que `PlayItem()` aparece en un `PlayItem_id for_loop` en la sintaxis. `PlayItem_id` comienza desde 0.

15 Puesto que los otros campos de sintaxis no son requeridos en la explicación de las formas de realización de la presente invención, se omite la descripción de los demás campos.

A continuación, se explica el significado de `PlayItem`. Un objeto de `PlayItem` incluye básicamente los datos siguientes:

20 (1): `Clip_information_file_name` para especificar el nombre de archivo de Clip apuntado por `PlayItem`.

(2): Un par de `IN_time` (tiempo_ENTRADA) y `OUT_time` (tiempo_SALIDA), que identifican un intervalo de reproducción de este objeto de Clip (Fragmento).

25 (3): `connection_condition` (condición-conexión) que ilustra el estado de conexión entre dos objetos `PlayItem` objetivos en `PlayList`, esto es, el objeto de `PlayItem` precedente y el objeto de `PlayItem` presente.

30 La Figura 29 es un diagrama explicativo que ilustra la `PlayList` con el tipo de CPI de `mapa_EP`. Dicha lista `PlayList` se refiere como `mapa_EP-type PlayList`. En el caso de cómo `mapa_EP-type PlayList`, `IN_time` y `OUT_time` de `PlayItem` son tiempos base de PTS. `IN_time` y `OUT_time` son tiempos en la misma secuencia del STC. `ref_to_STC_id` se utiliza para indicar la secuencia del STC. `IN_time` y `OUT_time` son tiempos en un intervalo de reproducción indicado por `presentation_start_time` y `presentation_end_time`, que se definen para la secuencia del STC y se incluyen en `SequenceInfo`.

35 La Figura 30 es un diagrama explicativo que ilustra `PlayList` con el tipo de CPI de `mapa_TU`. Dicha lista `PlayList` se refiere como `mapa_TU-type PlayList`. En el caso de cómo `mapa_TU-type PlayList`, `IN_time` y `OUT_time` de `PlayList` son tiempos base de PTS. `IN_time` y `OUT_time` son tiempos en la misma secuencia del ATC.

40 La Figura 31 es un diagrama explicativo que ilustra una relación entre la información de tiempos de `mapa_EP-type PlayList` y la información de direcciones en un archivo de flujo continuo AV. La información de tiempos de `PlayList` es la información de PTS de una imagen o una trama de audio en el archivo de flujo continuo AV. `mapa_EP` y `SequenceInfo` del archivo de información de Fragmento relacionan la información de tiempos, en el flujo continuo AV, con las direcciones en el archivo.

45 La Figura 32 es un diagrama explicativo que ilustra una relación entre la información de tiempos de `mapa_TU-type PlayList` y la información de direcciones en un archivo de flujos de AV. La información de tiempos de `PlayList` es la información de tiempos de llegada en el archivo de flujo continuo AV. `mapa_TU` del archivo de información de Fragmento relaciona la información de tiempos en el flujo continuo AV con las direcciones en el archivo.

50 La Figura 33 es un diagrama que ilustra la sintaxis de `PlayItem()`.

Longitud es el número de bytes que comienza con un byte inmediatamente después de este campo de longitud y que finaliza con el último byte de `PlayItem()`.

55 `Clip_Information_file name` es el nombre de un archivo de información de Fragmento referido por `PlayItem`.

`connection_condition` es la información que indica si un objeto de `PlayItem` precedente está conectado, o no, a un objeto de `PlayItem` presente sin discontinuidad.

60 `ref_to_STC_id` es `stc_id` de una secuencia del STC de Clip referida por `PlayItem`. El valor de `stc_id` se define en `SequenceInfo`.

65 `IN_time` (tiempo_ENTRADA) es un tiempo de inicio de reproducción de `PlayItem`.

`OUT_time` (tiempo_SALIDA) es un tiempo de final de reproducción `PlayItem`.

Bridge_Clip_Information_file_name es información auxiliar de reproducción para un caso en el que un objeto PlayItem precedente está conectado a un objeto PlayItem presente sin discontinuidad.

5 A continuación, se explica el concepto de edición de PlayList. El procesado, descrito a continuación, se realiza normalmente por la unidad de control 17 utilizada en el aparato de grabación/reproducción de imágenes en movimiento 1, cuya configuración se representa en la Figura 43, sobre la base de operaciones realizadas por el usuario.

10 La Figura 34 es un diagrama explicativo que ilustra el concepto de una relación, que se establece entre Clip y PlayList cuando un flujo continuo AV se graba como un nuevo objeto de Clip. Cuando un flujo continuo AV se graba como un nuevo objeto de Clip, se crea una Real PlayList que se refiere a una gama reproducible completa del objeto de Clip.

15 La Figura 35 es un diagrama explicativo que ilustra el concepto de creación de Virtual PlayList. Cuando el usuario especifica IN_time y OUT_time a partir de los tiempos en una gama de reproducción de Real PlayList, se crean PlayItem de un intervalo de reproducción que el usuario desea observar y Virtual PlayList.

20 La Figura 36 es un diagrama explicativo que ilustra el concepto de una relación, que se establece entre Clip y PlayList cuando se elimina una parte de un intervalo de reproducción de Real PlayList. PlayItem de PlayList se cambia de tal modo que solamente una parte de reproducción referida del flujo continuo AV de Clip sea objeto de referencia. A continuación, se eliminan las partes de flujos innecesarias del flujo continuo AV de Clip. Según se ilustra en la Figura 36, aún cuando se eliminen los datos de una parte media del flujo continuo AV de Clip, el archivo de flujo continuo AV de Clip no se divide, sino que permanece como un solo archivo. Aún cuando los datos de un flujo continuo AV de Clip se eliminen en parte, las partes de datos restantes se recogen en un flujo continuo AV de Clip.

25 Cuando se cambia Real PlayList y se elimina una parte del flujo de Clip referida por Real PlayList, se pierde un objeto de Clip referido por Virtual PlayList que utiliza el mismo objeto de Clip, los que probablemente plantea un problema operativo. Con el fin de resolver el problema, la interfaz del usuario adopta las contramedidas siguientes.

30 La interfaz del usuario avisa al usuario emitiendo un mensaje de consulta: "Existe Virtual PlayList que se refiere a una parte del flujo de Clip referida por Real PlayList de modo que, si se elimina Real PlayList, también se elimina Virtual PlayList. ¿Permite la eliminación también de Virtual PlayList?". En lugar de eliminar Virtual PlayList, se realiza el procesado de minimización en Real PlayList.

35 La Figura 37 es un diagrama explicativo que ilustra el concepto de una relación establecida entre Clip, Real PlayList y Virtual PlayList como resultado de edición de minimización. La edición de minimización es el procedimiento para cambiar PlayList de Real PlayList, de modo que solamente se refieran a partes del flujo de Clip requeridas por Virtual PlayList. Es decir, se eliminan partes de flujo no requerida por Virtual PlayList.

40 Según se ilustra en la Figura 37, aún cuando se eliminen los datos de una parte media del flujo continuo AV de Clip, el archivo de flujo continuo AV de Clip no se divide, sino que permanece como un solo archivo. Aún cuando se elimine, en parte, los datos de un flujo continuo AV de Clip, las partes de datos restantes se recogen en un flujo continuo AV de Clip.

45 Sobre la base del concepto anteriormente descrito, el cambio del archivo de información de Fragmento, en el caso de que se eliminen en parte, los datos del flujo continuo AV de Clip, se describe a continuación.

50 Según se describió anteriormente, cuando un flujo continuo AV se graba como un archivo de Clip, el objeto de Clip no incluye las discontinuidades de ATC y por ello, solamente tiene una secuencia del ATC. Se supone que una discontinuidad de ATC se crea solamente cuando los datos de flujo de archivo de flujo continuo AV de Clip se eliminan, en parte, en un proceso de edición o similar. Es decir, según se ilustra en las Figuras 36 y 37, cuando se eliminan, en parte, los datos de un flujo continuo AV de Clip y se recogen partes de datos restantes en el flujo continuo AV de Clip único, el objeto de Clip tiene discontinuidades de ATC y una pluralidad de secuencias ATC. En un ejemplo de edición, ilustrado en la Figura 38, a modo de ejemplo, el Clip anterior a un proceso de edición no incluye ninguna discontinuidad de ATC y por ello, solamente tiene una secuencia del ATC. Entonces, según se ilustra en la Figura, cuando se eliminan datos de una parte media del flujo continuo AV de Clip, el Clip después del proceso de edición tiene dos secuencias ATC.

60 La Figura 39 es un diagrama explicativo que ilustra la relación que se establece entre una secuencia del ATC, una secuencia del STC y una secuencia de programas cuando se eliminan, en parte, datos de un flujo continuo AV de Clip. El Clip antes de un proceso de edición tiene una sola secuencia del ATC, una sola secuencia del STC y una secuencia de programas. Es decir, en Clip, no cambian los contenidos de la secuencia de programas. En este proceso de edición, se supone que se eliminan elementos de datos de flujos de AV, que se indican por partes oscuras en la Figura. En este caso, el Clip después del proceso de edición tiene 3 secuencias ATC, 3 secuencias STC y solamente una secuencia de programas, que permanece como un resultado del proceso de

65

edición. La secuencia de programas se extiende sobre los entornos límite de las secuencias ATC y secuencias STC.

La siguiente descripción explica una relación, que se establece entre Clip y PlayList cuando se eliminan, en parte, datos del flujo continuo AV.

5 La Figura 40 es un diagrama explicativo que ilustra una relación, que se establece entre Clip y PlayList cuando se elimina una parte del flujo continuo AV de Clip que tiene mapa_EP CPI. El Clip antes de un proceso de edición tiene una secuencia del ATC y tres secuencias STC. `offset_STC_id [0]` para la secuencia del ATC es cero. Se supone que una secuencia del STC con un `stc_id = 1` en Clip se utiliza en `PlayItem2` y `PlayItem3`. Según se ilustra en la Figura, se editan datos del flujo continuo AV de la secuencia del STC con `stc_id = 1`. Para ser más concretos, se eliminan datos de flujo continuo AV de una parte no utilizada en `PlayItem2` y `PlayItem3`.

10 Como resultado del proceso de edición, Clip tiene ahora dos secuencias ATC y la secuencia del STC con `stc_id = 1` se divide en dos secuencias STC. `offset_STC_id[0]`, para la primera secuencia del ATC, se pone a cero y `offset_STC_id[1]`, para la segunda secuencia del ATC, se pone a 1. De este modo, ambos `stc_id` de la última secuencia del STC en la primera secuencia del ATC y `stc_id` de la anterior secuencia del STC en la segunda secuencia del ATC son 1.

15 Por lo tanto, no es necesario cambiar el valor de `ref_to_STC_id` de `PlayItem3` de Virtual PlayList después de la edición y el valor de `ref_to_STC_id` del `PlayItem4` de Virtual PlayList después de la edición. Es decir, cuando se eliminan datos parciales del archivo de flujo continuo AV de Clip, no es necesario cambiar Virtual PlayList que no utiliza la parte eliminada.

20 Puesto que se puede crear una discontinuidad de ATC en un flujo continuo AV de Clip de esta manera, si se eliminan datos de flujo de una parte media del flujo continuo AV de Clip, no es necesario dividir el archivo de Clip en dos partes. Además, utilizando `offset_STC_id` para `stc_id` de la primera secuencia del STC en una secuencia del ATC para cada una de las secuencias ATC, no es necesario cambiar Virtual PlayList que no utilice una parte parcial del archivo de flujo continuo AV de Clip, que se elimina en una eliminación parcial del archivo de flujo continuo AV de Clip.

25 Con el fin de ayudar al usuario a entender este efecto, la Figura 41 se utiliza como un diagrama para explicar un caso en el que un archivo Clip se divide en dos partes cuando se elimina una parte de un flujo continuo AV de Clip que tiene mapa_EP CPI así como un diagrama para una relación establecida entre Clip y PlayList para el caso.

30 Muy similar al ejemplo ilustrado en la Figura 40, el Clip anterior a un proceso de edición tiene una secuencia del ATC y tres secuencias STC. `offset_STC_id[0]` para la secuencia del ATC es cero. Se supone que una secuencia del STC con `stc_id = 1` en Clip se utiliza en `PlayItem2` y `PlayItem3`. Según se ilustra en la Figura, se edita los datos de flujo continuo AV de la secuencia del STC con `stc = 1`. Para ser más concretos, se eliminan los datos de flujo continuo AV de una parte no utilizada en `PlayItem2` y `PlayItem3`.

35 Si ninguna discontinuidad de ATC se permite en Clip, después del proceso de edición, Clip se divide en dos archivos, esto es, Clip-A y Clip-B. Por ello, es necesario cambiar el nombre del archivo de Clip referido por `PlayItem3` y `PlayItem4`. Es decir, cuando se eliminan datos parciales del archivo de flujo continuo AV de Clip, en algunos casos, es necesario cambiar los contenidos de Virtual PlayList aún cuando PlayList no utilice los datos parciales suprimidos.

40 En comparación con un caso en el que se permitan discontinuidades de ATC en Clip, el caso en el que no se permite ninguna discontinuidad de ATC en Clip tiene los problemas siguientes:

45 (1): El número de archivos grabados en el disco se hace mayor. En consecuencia, al principio del procesado de reproducción de disco, se necesita un tiempo más largo para la lectura de todos los archivos de Clip y almacenar los archivos en la memoria del aparato de reproducción (correspondiente a una memoria incorporada en la unidad de control 17 empleada en el aparato de grabación/reproducción de imágenes en movimiento 1, cuya configuración se representa en la Figura 43). Además, si el límite superior para el número de archivos susceptible de grabación en el disco (correspondiente a un medio de grabación 10 empleado en el aparato de grabación/reproducción de imágenes en movimiento 1, cuya configuración se representa en la Figura 43) se establece en un valor predeterminado y el número de archivos de Clip aumenta debido a un proceso de edición o similar, que exceda el límite superior, ya no se pueden grabar datos en el disco a pesar del hecho de que existe todavía un área libre de almacenamiento en el disco.

50 (2): Cuando se eliminan datos parciales del archivo de flujo continuo AV de Clip, se tarda un tiempo prolongado en cambiar Virtual PlayList existente en el disco.

55 La presente invención resuelve los problemas anteriormente descritos. Para una descripción en detalle, en virtud de la presente invención, es posible acortar el tiempo requerido para la lectura de todos los archivos de Clip y almacenar los archivos en la memoria utilizada en el aparato de reproducción al inicio del procesado de reproducción de disco. Además, es posibles establecer un número más bajo como el máximo de archivos que se pueden grabar

en el disco. Además, también es posible reducir el tiempo, que se requiere para cambiar Virtual PlayList existente en el disco cuando se eliminan datos parciales del archivo de flujo continuo AV de Clip.

La Figura 42 es un diagrama explicativo que ilustra una relación que se establece entre Clip y PlayList, cuando se elimina una parte del flujo continuo AV de Clip que tiene mapa_TU CIP. Clip, antes de un proceso de edición, tiene una sola secuencia del ATC. `offset_arrival_time[0]` para esta secuencia del ATC, es cero. Se supone que `PlayItem1`, `PlayItem2`, `PlayItem3` y `PlayItem4` de Virtual PlayList se refieren a esta secuencia del ATC. Entonces, según se ilustra en la Figura, se editan datos de flujo continuo AV de esta secuencia del ATC. Para ser más concretos, se eliminan datos de flujo continuo AV no utilizados en cualquiera de los objetos de PlayList.

El Clip, después del proceso de edición, tiene dos secuencias ATC. `offset_arrival_time[0]` para la primera secuencia del ATC se establece en cero y `offset_arrival_time[1]` para la segunda secuencia del ATC se establece en un valor X, que es mayor que `OUT_time2` pero más pequeño que `IN_time3`. Es decir, después del proceso de edición, no es necesario cambiar los valores de `IN_time` y `OUT_time`, que pertenecen a `PlayItem3` de Virtual PlayList así como los valores de `IN_time` y `OUT_time`, que pertenecen a `PlayItem4` de Virtual PlayList.

No es necesario cambiar Virtual PlayList que no utiliza parte del archivo de flujo continuo AV de Clip, que se elimina en una eliminación parcial del archivo de flujo continuo AV de Clip.

En la reproducción de PlayList del tipo `mapa_TU`, el reproductor es capaz de encontrar una secuencia del ATC apuntada por `IN_time` y `OUT_time` comparando el valor de `IN_time` de `PlayItem` con el valor de `offset_arrival_time` de la secuencia del ATC. En el ejemplo ilustrado en la Figura 42, a modo de ejemplo, puesto que `IN_time3` de `PlayItem3` es mayor que `offset_arrival_time (= X)` de la segunda secuencia del ATC, es evidente que `IN_time3` y `OUT_time3` de `PlayItem3` apuntan a la segunda secuencia del ATC.

Haciendo referencia a un diagrama de bloques de la Figura 43 que ilustra el aparato de grabación/reproducción de imágenes en movimiento 1, la siguiente descripción explica un sistema de grabación/reproducción para grabar/reproducir datos con la estructura de aplicación de DVR.

Una unidad de lectura 11, utilizada en una unidad de reproducción 61 efectúa la lectura de la información grabada en un medio de grabación 10, que suele ser un disco óptico. Una unidad de demodulación 12 demodula datos objeto de lectura por la unidad de lectura 11 desde el medio de grabación 10 y suministra datos demodulados a una unidad de decodificación de ECC 13. La unidad de decodificación de ECC 13 divide los datos recibidos desde la unidad de demodulación 12 en un flujo continuo AV y una base de datos, que suministra el flujo continuo AV y la base de datos a un desempaqetador fuente 14 y la unidad de control 17, respectivamente.

El desempaqetador fuente 14 desempaqueta el flujo continuo AV de entrada y proporciona, a la salida, un resultado del desempaqetamiento a un demultiplexor 15. El demultiplexor 15 divide el resultado del desempaqetamiento, recibido desde el desempaqetador fuente 14, en datos de vídeo (V), datos de audio (A) y datos de sistemas (S), que proporcionan, a la salida, los datos de vídeo, los datos de audio y los datos de sistema para un decodificador de AV 16 y un multiplexor 25.

El decodificador de AV 16 decodifica los datos de vídeo y los datos de audio sobre la base de los datos del sistema, proporcionando a la salida señales de audio y de vídeo a partir de los terminales de vídeo y de audio, 18 y 19, respectivamente.

La entrada de señales de vídeo y de audio desde los terminales de entrada 21 y 22, respectivamente, se suministran a un codificador de AV 23 utilizando en una unidad de grabación 62. La señal de vídeo se suministra también a una unidad de análisis de vídeo 24. En lugar de la entrada de señal de vídeo desde el terminal de entrada de vídeo 21, una salida de señal de vídeo, por el decodificador de AV 16, puede suministrarse a un codificador de AV 23 y la unidad de análisis de vídeo 24, si fuera necesario.

El codificador de AV 23 codifica las señales de vídeo y de audio de entrada, proporcionando, a la salida, una señal de vídeo codificada (V), una señal de audio codificada (A) y datos de sistemas (S) para el proceso de codificación a un multiplexor 25.

La unidad de análisis de vídeo 24 analiza la señal de vídeo de entrada y proporciona a la salida, un resultado del análisis a la unidad de control 17.

Un terminal 33 recibe un flujo continuo de transporte desde una interfaz digital o un sintonizador de televisión digital. El flujo continuo de transporte se suministra al demultiplexor 15 o un conmutador 28 por intermedio de un conmutador 27. El flujo continuo de transporte suministrado al conmutador 28 se reenvía a una unidad de análisis de flujo multiplexado 26 y a un empaquetador fuente 29. Cambiando la posición de ajuste del conmutador 28, en lugar de suministrar el flujo continuo de transporte suministrado desde el conmutador 27 a la unidad de análisis de flujo multiplexado 26 y al empaquetador fuente 29, una salida de señal por el multiplexor 25 se puede suministrar a la unidad de análisis de flujo multiplexado 26 y el empaquetador fuente 29 por intermedio del conmutador 28.

5 La unidad de análisis de flujo multiplexado 26 analiza la señal de entrada y proporciona, a la salida, un resultado del análisis a la unidad de control 17. El empaquetador fuente 29 empaqueta la señal de entrada y suministra un resultado del empaquetamiento a una unidad de codificación de ECC 30. La unidad de codificación ECC 30 recibe también una base de datos desde la unidad de control 17.

10 La unidad de codificación ECC 30 añade códigos de corrección de errores a la entrada y codifica la entrada, suministrando datos codificados a una unidad de modulación 31. La unidad de modulación 31 modula los datos codificados recibidos desde la unidad de codificación de ECC 30 y proporciona, a la salida, datos modulados a una unidad de escritura 32. La unidad de escritura 32 realiza el procesado para la escritura de los datos modulados, recibidos desde la unidad de modulación 31, en el medio de grabación 10.

15 La unidad de control 17 tiene una unidad de almacenamiento 17A para memorizar varias clases de datos. La unidad de control 17 gestiona los formatos descritos anteriormente y controla otros componentes para grabar/reproducir datos en y desde el medio de grabación 10.

La unidad de control 17 está conectada a una unidad 41 para excitar un disco magnético 51, un disco óptico 52, un disco magneto-óptico 53 o una memoria de semiconductores 54.

20 Conviene señalar que el disco óptico 52 puede utilizarse en conjunción con el medio de grabación 10.

25 A continuación, se describen operaciones de grabación básicas explicando un caso en el que el aparato de grabación/reproducción de imágenes en movimiento 1 codifica y graba, por sí mismo, señales de audio y de vídeo a la entrada.

30 Las señales de vídeo y de audio se introducen a través de, respectivamente, los terminales de entrada de vídeo y de audio 21 y 22 de la unidad de grabación 62. La señal de vídeo se suministra a la unidad de análisis de vídeo 24 y al codificador de AV 23. El codificador de AV 23 recibe también la señal de audio. El codificador de AV 23 codifica las señales de vídeo y de audio de entrada, proporciona a la salida un flujo continuo de vídeo codificado (V), un flujo de audio codificado (A) e información del sistema (S) al multiplexor 25.

35 El flujo continuo de vídeo codificado (V) suele ser un flujo continuo de vídeo de MPEG-2 mientras que el flujo de audio codificado (A) suele ser un flujo de audio MPEG-1 o un flujo de audio de Dolby AC3 (marca registrada) o similar. La información del sistema (S) es la información temporal tal como información sobre la sincronización de AV e información sobre la codificación de vídeo/audio, incluyendo el número de bytes que constituyen una imagen codificada, el número de bytes que constituyen una trama de audio y un tipo de codificación de imágenes.

40 El multiplexor 25 efectúa la multiplexación de los flujos de entrada sobre la base de la información del sistema de entrada para generar un flujo multiplexado. El flujo multiplexado suele ser un flujo continuo de transporte de MPEG-2 o un flujo de programa MPEG-2. El flujo multiplexado se suministra a la unidad de análisis de flujo multiplexado 26 y al empaquetador fuente 29. En conformidad con el formato de aplicación del medio de grabación 10, el empaquetador fuente 29 codifica el flujo multiplexado de entrada en un flujo continuo AV que comprende paquetes fuente. La unidad de codificación de ECC 30 añade códigos de corrección de errores al flujo continuo AV antes de la modulación en la unidad de modulación 31, que proporciona, a la salida un flujo continuo AV modulado a la unidad de escritura 32. La unidad de escritura 32 graba, por último, un archivo de flujo continuo AV en el medio de grabación 10 en conformidad con una señal de control generada por la unidad de control 17.

50 La siguiente descripción explica las operaciones para grabar un flujo continuo de transporte, tal como una entrada de señal de difusión de TV digital desde uno de entre una interfaz digital y un sintetizador de TV digital, que no se ilustran en la Figura.

55 El terminal de entrada digital 33 es un terminal para recibir un flujo continuo de transporte. Existen dos métodos de grabación del flujo continuo de transporte a la entrada. Uno de los métodos es un método de grabación transparente. El otro método es una técnica mediante la cual el flujo se recodifica antes de grabarse para la finalidad de disminuir la velocidad en bits de grabación. La unidad de control 17 recibe información que indica el método de grabación a adoptarse desde un terminal 20, que se utiliza como una interfaz de usuario de entrada/salida. La unidad de control 17 controla el método de grabación.

60 En el caso del método de grabación transparente, el flujo continuo de transporte de entrada se suministra a la unidad de análisis del flujo multiplexado 26 y al empaquetador fuente 29. Las operaciones realizadas a continuación de la grabación del flujo continuo AV en el medio de grabación 10 son las mismas que las del procesado para codificar y grabar señales de audio y de vídeo de entrada según se describió anteriormente.

65 En el caso del método en el que el flujo continuo de transporte de entrada se recodifica antes de un proceso de grabación, el flujo continuo de transporte de entrada se suministra al demultiplexor 15. El demultiplexor 15 suministra

el flujo continuo de vídeo (V) al decodificador de AV 16. El decodificador de AV 16 decodifica el flujo continuo de vídeo y suministra una señal de vídeo reproducida obtenida como resultado del proceso de decodificación al codificador de AV 23. El codificador de AV 23 codifica la señal de vídeo reproducida y suministro el flujo continuo de vídeo codificado (V) al multiplexor 25.

5 Por otro lado, el flujo de audio (A) y la información del sistema (S), que se proporcionan a la salida por el demultiplexor 15, se suministran directamente al multiplexor 25. El multiplexor 25 realiza la multiplexación de los flujos de vídeo y de audio de entrada sobre la base de la información del sistema de entrada para generar un flujo multiplexado. Las operaciones realizadas, a continuación, para grabar el flujo continuo AV en el medio de grabación
10 son las mismas que las del procesado para codificar y grabar las señales de audio y de vídeo de entrada según se describió anteriormente.

15 El aparato de grabación/reproducción de imágenes en movimiento 1 graba no solamente un archivo de flujo continuo AV, sino también la información de base de datos de aplicaciones pertinente para el archivo. La información de base de datos de aplicaciones se crea por la unidad de control 17. La entrada de información a la unidad de control 17 incluye la información característica de la salida de imagen en movimiento por la unidad de análisis de vídeo 24. La información característica de la salida del flujo continuo AV por la unidad de análisis de flujo multiplexado 26 y una orden introducida por el usuario al terminal de entrada/salida de interfaz de usuario 20 que se utiliza como una interfaz de usuario.

20 La salida de información característica de imagen en movimiento por la unidad de análisis de vídeo 24 se genera en el aparato de grabación/reproducción de imágenes en movimiento 1 cuando dicho aparato de grabación/reproducción de imágenes en movimiento 1 codifica por sí mismo la señal de vídeo. La unidad de análisis de vídeo 24 analiza los datos transmitidos por una señal de vídeo de entrada para generar información sobre una imagen en los puntos de marcas característicos en la señal de imagen en movimiento a la entrada. Esta información es la que indica imágenes en puntos de marca característicos tales como los puntos de inicio del programa de la señal de vídeo de entrada, puntos de cambio de escena operativa y puntos de inicio/final de CM. Además, la información incluye miniaturas de las imágenes. La información que indica las imágenes se suministra al multiplexor
25 25 a través de la unidad de control 17.

30 Cuando el multiplexor 25 realiza la multiplexación de imágenes codificadas en los puntos de marcas indicados por la unidad de control 17, el multiplexor 25 reenvía direcciones, en donde las imágenes codificadas están localizadas en un flujo continuo AV, a la unidad de control 17. La unidad de control 17 memoriza la dirección, en donde cada imagen codificada está situada en el flujo continuo AV, asociando la dirección con el tipo de la imagen característica.

35 La información característica de la salida del flujo continuo AV por la unidad de análisis de flujo multiplexado 26 está relacionada con la información de codificación del flujo continuo AV a grabarse y generarse en el aparato de grabación/reproducción de imágenes en movimiento 1. La información característica de la salida de flujo continuo AV por la unidad de análisis de flujo multiplexado 26 incluye una marca de tiempo y un dirección de una imagen I en el flujo continuo AV, información sobre discontinuidades de STC en el flujo continuo AV, información sobre los cambios en los contenidos de programas en el flujo continuo AV así como una hora de llegada y una dirección en el flujo continuo AV.

40 La marca de tiempo y la dirección de una imagen I en el flujo continuo AV se procesan como datos a memorizarse en mapa_EP anteriormente descrito. La información sobre las discontinuidades de STC, en el flujo continuo AV, se procesa como datos a memorizarse en SequenceInfo según se describió anteriormente. La información sobre cambios en contenidos de programas, en el flujo continuo AV, se procesa como datos a memorizarse en ProgramInfo según se describió anteriormente. La hora de llegada y la dirección en la que se memoriza el flujo continuo AV en mapa_TU según se describió con anterioridad.

45 En el caso del método transparente para grabar una entrada de flujo continuo de transporte desde el terminal de entrada digital 33, la unidad de análisis de flujo multiplexado 26 detecta una imagen en un punto de marca característico en el flujo continuo AV, generando el tipo y la dirección de la imagen. El tipo y la dirección se procesan como datos a memorizarse en ClipMark.

50 La información característica de la salida de flujo continuo AV por la unidad de análisis de flujo multiplexado 26 se memoriza en una base de datos (información de Fragmento) del flujo continuo AV.

55 Una orden introducida por el usuario al terminal de entrada/salida de interfaz del usuario 20 incluye información que especifica un intervalo de reproducción deseado en el flujo continuo AV, un texto de caracteres que explica los contenidos del intervalo de reproducción, una marca de referencia a establecerse en una escena operativa deseada por el usuario y una marca de tiempo de un punto de reanudación en el flujo continuo AV. La orden introducida por el usuario se memoriza en la base de datos de PlayList.

60 La unidad de control 17 crea una base de datos (información de Fragmento) del flujo continuo AV, una base de datos de PlayList, información de gestión (information.dvr) de datos grabados en el medio de grabación 10 e información

de miniaturización sobre la base de la información de entrada anteriormente descrita. Estos elementos de información de base de datos se procesan por la unidad de codificación de ECC (corrección de errores) 30 y la unidad de modulación 31 de la misma manera que el flujo continuo AV y se suministra a la unidad de escritura 32. En conformidad con una señal de control generada por la unidad de control 17, la unidad de escritura 32 suministra la información de base de datos al medio de grabación 10 para grabarse como información de base de datos de aplicaciones.

A continuación, se explica el procesado de reproducción básico.

El medio de grabación 10 se utiliza para grabar archivos de flujos de AV y la información de base de datos de aplicaciones.

Ante todo, la unidad de control 17 demanda a la unidad de lectura 11, utilizada en la unidad de reproducción 61, que efectúe la lectura de la información de base de datos de aplicaciones desde el medio de grabación 10. A esta demanda, la unidad de lectura 11 efectúa la lectura de la información de base de datos de aplicaciones desde el medio de grabación 10. La información de base de datos de aplicaciones se procesa por la unidad de demodulación 12 y la unidad de decodificación ECC 13 antes de suministrar a la unidad de control 17.

La unidad de control 17 proporciona, a la salida, una lista de objetos de Playlist grabados en el medio de grabación 10 al terminal de entrada/salida de interfaz de usuario 20 sobre la base de la información de base de datos de aplicaciones. El usuario selecciona la Playlist a reproducirse a partir de la lista e introduce el objeto de Playlist seleccionado para reproducirse a la unidad de control 17. La unidad de control 17 demanda a la unidad de lectura 11 que efectúe la lectura de un archivo de flujo continuo AV que se requiere para la reproducción del objeto de Playlist seleccionado desde el medio de grabación 10. La unidad de lectura 11 efectúa la lectura del archivo de flujo continuo AV requerido desde el medio de grabación 10. El archivo de flujo continuo AV objeto de lectura desde el medio de grabación 10 se procesa por la unidad de demodulación 12 y la unidad de decodificación de ECC 13 antes de suministrarse al empaquetador fuente 14.

El empaquetador fuente 14 convierte el archivo de flujo continuo AV, que tiene un formato de aplicación para el medio de grabación, en un flujo que se puede suministrar al demultiplexor 15. El demultiplexor 15 suministra un flujo continuo de vídeo (V), un flujo de audio (A) e información del sistema(S), que constituyen el intervalo de reproducción del flujo continuo AV (PlayItem) especificado por la unidad de control 17 al decodificador de AV 16. El decodificador de AV 16 decodifica los flujos de vídeo y de audio para generar señales de vídeo y de audio reproducidas a suministrarse a los terminales de salida de vídeo y de audio 18 y 19, respectivamente.

Si se desea reproducir la lista Playlist del tipo de mapa_EP seleccionada por el usuario desde un punto medio en el tiempo, la unidad de control 17 demanda a la unidad de lectura 11 que efectúe la lectura de datos comenzando desde la dirección de una imagen I que tiene un PTS más próximo al punto especificado en el tiempo.

Además, si se desea reproducir la lista Playlist del tipo de mapa_TU seleccionada por el usuario desde un punto medio en el tiempo, la unidad de control 17 demanda a la unidad de lectura 11 que efectúe la lectura de los datos comenzando desde la dirección de paquete fuente que tiene un tiempo de llegada más próximo al punto especificado en el tiempo.

Además, cuando el usuario selecciona una marca entre el punto de apariencia de cabecera y los puntos de cambio de escena operativa del programa, la unidad de control 17 determina una localización para efectuar la lectura del flujo continuo AV desde el medio de grabación 10 sobre la base del contenido de la información de Fragmento y demanda a la unidad de lectura 11 que efectúe la lectura del flujo continuo AV desde el medio de grabación 10. El punto de apariencia de cabecera y los puntos de cambio de escena operativa se memorizan en la información ClipMark de Clip. El usuario suele seleccionar la marca a partir de una lista de imágenes de miniatura que representa los puntos de cambio de escena operativa del programa y el punto de apariencia de cabecera, que se memorizan en ClipMark de la información de Fragmento. La lista se visualiza en la interfaz de usuario.

A la demanda, la unidad de lectura 11 efectúa la lectura de datos de una imagen I en una dirección más próxima a una dirección en el flujo continuo AV. En la dirección del flujo continuo AV, se memoriza una imagen seleccionada por el usuario. Los datos objeto de lectura por la unidad de lectura 11, desde la dirección especificada, se procesan por la unidad de demodulación 12 y la unidad de decodificación de ECC 13 antes de suministrarse al decodificador de AV 16 por intermedio del demultiplexor 15. El decodificador de AV 16 decodifica los datos para reproducir datos de AV en la dirección de una imagen en el punto de marca.

La siguiente descripción explica un caso en el que el usuario realiza la edición de un flujo continuo AV.

Cuando el usuario desea crear una nueva ruta de reproducción especificando un intervalo de reproducción de un flujo continuo AV memorizado en el medio de grabación 10, la información de puntos de entrada y de salida en el intervalo de reproducción se suministra desde el terminal de entrada/salida de interfaz de usuario 20 a la unidad de control 17. La unidad de control 17 crea, entonces, una base de datos de Playlist, que es un grupo de intervalos de

reproducción (objetos de PlayItem) del flujo continuo AV.

5 Cuando el usuario desea eliminar una parte específica innecesaria de un flujo continuo AV memorizado en el medio de grabación 10, la información sobre un intervalo a eliminarse se suministra desde el terminal de entrada/salida de interfaz de usuario 20 a la unidad de control 17. La unidad de control 17 cambia la base de datos de PlayList con el fin de referirse a solamente una parte requerida del flujo continuo AV. La unidad de control 17 demanda también a la unidad de escritura 32 que efectúe la eliminación de la parte específica innecesaria del flujo continuo AV. Los contenidos del archivo de información de Fragmento se cambian sobre la base del cambio en el flujo continuo AV de Clip.

10 La siguiente descripción explica las operaciones para un caso en donde el usuario desea crear una nueva ruta de reproducción especificando con los intervalos de reproducción de un flujo continuo AV que se memorizan en el medio de grabación 10 y conectar los intervalos entre sí sin discontinuidad. En este caso, la unidad de control 17 crea una base de datos de PlayList, que es un grupo de intervalos de reproducción (objetos de PlayItem) del flujo continuo AV. Además, es también necesario recodificar parcialmente y remultiplexar las partes del flujo continuo de vídeo próximas a los puntos de unión de los intervalos de reproducción.

15 Ante todo, la información sobre las imágenes en los puntos de entrada y de salida de cada intervalo de reproducción se suministra desde el terminal de entrada/salida de interfaz de usuario 20 a la unidad de control 17. La unidad de control 17 demanda a la unidad de lectura 11 que efectúe la lectura de los datos, que se requieren para la reproducción de las imágenes en los puntos de entrada y de salida, desde el medio de grabación 10. La unidad de lectura 11 efectúa la lectura de los datos desde el medio de grabación 10. Los datos se suministran al demultiplexor 15 por intermedio de la unidad de demodulación 12, la unidad de codificación de ECC 13 y el desempaquetador fuente 14.

20 La unidad de control 17 analiza el flujo suministrado al demultiplexor 15 con el fin de determinar métodos para recodificar y remultiplexar el flujo continuo de vídeo, suministrando los métodos al codificador de AV 23 y al multiplexor 25. El método de recodificación incluye técnicas en cuanto a la forma en que se cambia el tipo de codificación de la imagen y como los conteos de bits codificados se asignan en el proceso de recodificación.

25 A continuación, el demultiplexor 15 divide el flujo de entrada en un flujo continuo de vídeo (V), un flujo de audio (A) e información del sistema (S). El flujo continuo de vídeo comprende datos a suministrarse al decodificador AV 16 y datos a suministrarse directamente al multiplexor 25. Los datos anteriores son datos para ser recodificados. Estos datos se decodifican por el decodificador de AV 16. Una imagen obtenida como resultado del proceso de decodificación se recodifica por el codificador de AV 23 en un flujo continuo de vídeo. Los últimos datos son los datos copiados desde el flujo original y no son recodificados. El flujo de audio y la información del sistema se suministran directamente al multiplexor 25.

30 El multiplexor 25 realiza la multiplexación del flujos de entrada para genera un flujo de multiplexado sobre la base de la información recibida desde la unidad de control 17. El flujo de multiplexado se procesa por la unidad de codificación de ECC 30 y la unidad de modulación 31 antes de suministrarse a la unidad de escritura 32. La unidad de escritura 32 graba el flujo continuo AV en el medio de grabación 10 en función de una señal de control recibida desde la unidad de control 17.

35 La Figura 44 ilustra un diagrama de flujo que representa las operaciones realizadas por el aparato de grabación/reproducción de imágenes en movimiento 1 para crear un archivo de flujo continuo AV de Clip y un archivo de información de Fragmento al procesar para grabación un flujo continuo AV según un nuevo objeto de Clip.

40 Según se ilustra en la Figura, el diagrama de flujo se inicia con una etapa S11 en la que la unidad de control 17 crea y graba un archivo de flujo continuo AV de Clip (Fragmento) para un flujo continuo de transporte obtenido como resultado de un proceso para codificar entradas de AV recibidas desde los terminales de entrada de vídeo y de audio 21 y 22 o una entrada de flujo continuo de transporte desde el terminal de interfaz digital 33.

45 A continuación, en la siguiente etapa S12, la unidad de control 17 crea ClipInfo ilustrado en la Figura 8 para el archivo de flujo continuo AV.

50 Posteriormente, en la siguiente etapa S13, la unidad de control 17 crea SequenceInfo, según se representa en la Figura 13, para el archivo de flujo continuo AV.

55 A continuación, en la siguiente etapa S14, la unidad de control 17 crea ProgramInfo ilustrado en la Figura 15 para el archivo de flujo continuo AV.

60 Posteriormente, en la siguiente etapa S15, la unidad de control 17 crea CPI (mapa_EP o mapa_TU) ilustrado en las Figuras 24, 25 y 26 para el archivo de flujo continuo AV.

65 A continuación, en la siguiente etapa S16, la unidad de control 17 crea ClipMark para el archivo de flujo continuo AV.

Posteriormente, en la siguiente etapa S17, la unidad de control 17 crea un archivo de información de fragmento (Clip) ilustrado en la Figura 8. El archivo de información de Fragmento se utiliza para la grabación de ClipInfo, SequenceInfo, ProgramInfo, CPI y ClipMark, que se indicaron con anterioridad.

5 Conviene señalar que, aunque la explicación anterior indica que los elementos de procesado se realizan secuencialmente a lo largo del eje de tiempos, los elementos de procesado son realmente realizados simultáneamente, al mismo tiempo, en las etapas S11 a S16.

10 Haciendo referencia a un diagrama de flujo ilustrado en la Figura 45, la siguiente descripción explica operaciones típicas para crear SequenceInfo, según se ilustra en la Figura 13, en el procesado para grabar un flujo continuo AV como nuevo Clip. Las operaciones se realizan por la unidad de análisis de flujo multiplexado 26 utilizada en el aparato de grabación/reproducción de imágenes en movimiento 1, cuya configuración se ilustra en la Figura 43.

15 Según se ilustra en la Figura, el diagrama de flujo comienza con una etapa S31 en la que la unidad de control 17 utiliza el primer paquete de transporte como el punto de partida de una secuencia del ATC. Es decir, la unidad de control 17 establece SPN_ATC_start. En ese momento, se establecen también atc_id y stc_id.

20 A continuación, en la siguiente etapa S32, la unidad de análisis de flujo multiplexado 26 analiza un PTS de una unidad de acceso incluida en el flujo continuo AV. Un ejemplo de la unidad de acceso es una imagen de una trama de vídeo.

25 Posteriormente, en la siguiente etapa S33, la unidad de análisis de flujo multiplexado 26 forma un juicio operativo en cuanto a si se ha recibido, o no, un paquete de PCR. Si el resultado de la determinación realizada en la etapa S33 indica que no se ha recibido un paquete de PCR, el flujo del procesado retrocede a la etapa S32. Si el resultado de la determinación realizada en la etapa S33 indica que se ha recibido un paquete de PCR, por el contrario, el flujo del procesado pasa a una etapa S34.

30 En la etapa S34, la unidad de análisis de flujo multiplexado 26 realiza una determinación en cuanto a si se ha detectado, o no, una discontinuidad de STC. Si el resultado de la determinación realizada en la etapa S34 indica que no se ha detectado una discontinuidad de STC, el flujo del procesado retrocede a la etapa S32. Si el resultado de la determinación realizada en la etapa S34 indica que se ha detectado una discontinuidad de STC, por el contrario, el flujo de procesado pasa a la etapa S35. En el caso del primer paquete de PCR recibido el flujo del procesado pasa siempre a la etapa S35.

35 En la etapa S35, la unidad de análisis de flujo multiplexado 26 adquiere el número (la dirección) de un paquete de transporte para transmitir un primer PCR de la nueva secuencia del STC.

40 A continuación, en la siguiente etapa S36, la unidad de control 17 utiliza el número de paquetes adquirido en la etapa S35 como el número de un paquete fuente al principio de la secuencia del STC. Es decir, se establece SPN_STC_start. Además se establece también un nuevo stc_id.

45 Posteriormente, en la siguiente etapa S37, la unidad de control 17 adquiere el PTS de inicio de visualización de la secuencia del STC y el PTS de final de visualización y establece el PTS de inicio de visualización y el PTS de final de visualización en presentation_start_time y presentation_end_time, respectivamente. La unidad de control 17 crea, entonces, la SequenceInfo representada en la Figura 13, sobre la base de la PTS de inicio de la presentación visual y la PTS de final de la presentación visual.

50 A continuación, en la siguiente etapa S38, la unidad de control 17 realiza una determinación en cuanto a si se ha recibido, o no, el último paquete de transporte. Si el resultado de la determinación realizada en la etapa S38 indica que no se ha recibido el último paquete de transporte, el flujo de procesado retrocede a la etapa S32. Si el resultado de la determinación realizada en la etapa S38 indica que se ha recibido el último paquete de transporte, por el contrario, se acaba el procesado para crear SequenceInfo.

55 Conviene señalar que, en el caso de Clip con mapa_TU CPI, solamente necesita crearse información de la secuencia del ATC. Por ello, no se requieren los elementos de procesado que se realizan en las etapas S32 a S37.

60 A continuación, se explican las operaciones típicas realizadas para crear ProgramInfo, según se ilustra en la Figura 15, haciendo referencia a un diagrama de flujo ilustrado en la Figura 46. Estas operaciones se realizan por la unidad de análisis de flujo multiplexado 26 que se utiliza en el aparato de grabación/reproducción de imágenes en movimiento 1, cuya configuración se representa en la Figura 43.

65 Conviene señalar que, en el caso de Clip con mapa_TU CPI, no se requiere la información de la secuencia de programas. Por ello, las operaciones representadas por el diagrama de flujo de la Figura 46 no son necesarias.

Según se ilustra en la Figura, el diagrama de flujo comienza con una etapa S51 en la que la unidad de análisis de

- flujo multiplexado 26 realiza una determinación en cuanto a si se ha recibido, o no, un paquete de transporte que incluye PSI/SI. Para exponerlo más concretamente, un paquete de transporte, que incluye PSI/SI, es un paquete PAT, PMT y SIT. Un paquete SIT es un paquete de transporte que describe información de servicio de un flujo continuo de transporte parcial prescrito por especificaciones de DVB. Si el resultado de la determinación realizada en la etapa S51 indica que no se ha recibido un paquete de transporte que incluye PSI/SI, el flujo de procesado retrocede a la etapa S51. Si el resultado de la determinación realizada en la etapa S51 indica que se ha recibido un paquete de transporte que incluye PSI/SI, por el contrario, el flujo del procesado pasa a una etapa S52.
- En la etapa S52, la unidad de análisis de flujo multiplexado 26 realiza una determinación en cuanto a si se ha cambiado, o no, los contenidos de PSI/SI. Con más detalle, la unidad de análisis de flujo multiplexado 26 realiza una determinación en cuanto a si se recibió anteriormente, o no, los contenidos de cada uno de los paquetes PAT, PMT y SIT que son diferentes, o no, de los contenidos recibidos con anterioridad. Si el resultado de la determinación realizada en la etapa S52 indica que no se han cambiado los contenidos, el flujo de procesado retrocede a la etapa S51. Si el resultado de la determinación realizada en la etapa S52 indica que se han cambiado los contenidos, por el contrario, el flujo de procesado pasa a la etapa S53. Conviene señalar que, al principio de una operación de grabación, el PSI/SI se recibe por primera vez. Por ello, en este caso, el flujo de procesado siempre pasa a la etapa S53.
- En la etapa S53, la unidad de control 17 adquiere el número (la dirección) del paquete de transporte para transmitir el PSI/SI y los contenidos del paquete.
- A continuación, en la siguiente etapa S54, la unidad de control 17 crea información de la secuencia de programas con el fin de formar ProgramInfo según se ilustra en la Figura 15.
- Posteriormente, en la siguiente etapa S55, la unidad de control 17 realiza una determinación en cuando a si el paquete de transporte recibido es, o no, el último paquete de transporte. Si el resultado de la determinación realizada en la etapa S55 indica que el paquete de transporte recibido no es el último paquete de transporte, el flujo de procesado retrocede a la etapa S51. Si el resultado de la determinación realizada en la etapa S55 indica que el paquete de transporte recibido es el último paquete de transporte, por el contrario, se acaba este procesado para crear ProgramInfo.
- A continuación, se aplican las operaciones realizadas para crear mapa_EP, según se ilustra en la Figura 24, haciendo referencia a un diagrama de flujo ilustrado en la Figura 47. Estas operaciones se realizan por la unidad de análisis de flujo multiplexado 26 utilizada en el aparato de grabación/reproducción de imágenes en movimiento 1, cuya configuración se representa en la Figura 43.
- Según se ilustra en la Figura, el diagrama de flujo se inicia con una etapa S71 en la que la unidad de análisis de flujo multiplexado 26 establece el PID del vídeo de un programa de AV a grabarse. Si el flujo continuo de transporte incluye una pluralidad de vídeos, la unidad de análisis de flujo multiplexado 26 establece el PID de cada uno de los vídeos.
- A continuación, en la siguiente etapa S72, la unidad de análisis de flujo multiplexado 26 recibe el paquete de transporte de vídeo.
- Posteriormente, en la siguiente etapa S73, la unidad de análisis de flujo multiplexado 26 realiza una determinación en cuanto a si la carga útil del paquete de transporte comienza, o no, con el primer byte de un paquete de PES. Una carga útil es una parte de paquete que sigue la cabecera del paquete. Obtenido como resultado del empaquetamiento de un flujo elemental, un paquete de PES es un paquete prescrito en las especificaciones de MPEG-2. La determinación se realiza examinando el valor de "payload_unit_start_indicator" incluido en la cabecera del paquete de transporte. Un valor de 1 indica que la carga útil del paquete de transporte se inicia con el primer byte de un paquete PES. Si el resultado de la determinación realizada en la etapa S73 indica que el paquete de transporte no se inicia por el primer byte de un paquete PES, el flujo de procesado retrocede a la etapa S72. Si el resultado de la determinación realizada en la etapa S73 indica que el paquete de transporte se inicia con el primer byte de un paquete PES, por el contrario, el flujo de procesado pasa a la etapa S74.
- En la etapa S74, la unidad de análisis de flujo multiplexado 26 realiza una determinación en cuanto a si la carga útil del paquete PES se inicia, o no, con el primer byte del código de cabecera de secuencia del vídeo MPEG.
- sequence_header_code es el código de "0x000001B3" que tiene una longitud de 32 bits. Si el resultado de la determinación realizada en la etapa S74 indica que la carga útil del paquete PES no se inicia con el primer byte del código de cabecera de secuencia sequence_header_code, el flujo de procesado retrocede a la etapa S72. Si el resultado de la determinación realizada en la etapa S74 indica que la carga útil del paquete PES se inicia con el primer byte del código de cabecera de secuencia, por el contrario, el flujo del procesado pasa a una etapa S75.
- En la etapa S75, la unidad de control 17 utiliza el paquete de transporte presente como un punto de entrada.

A continuación, en la siguiente etapa S76, la unidad de control 17 adquiere el número del paquete, el PTS de una imagen I que comienza desde el código de cabecera de secuencia y el PID de un vídeo al que pertenece el punto de entrada con el fin de crear mapa_EP.

5 Posteriormente, en la siguiente etapa S77, la unidad de análisis de flujo multiplexado 26 realiza una determinación en cuanto a si el paquete presente es, o no, el último paquete de transporte de entrada. Si el resultado de la determinación realizada en la etapa S77 indica que el paquete presente no es el último paquete de transporte de entrada, el flujo de procesado retrocede a la etapa S72. Si el resultado de la determinación realizada en la etapa S77 indica que el paquete presente es el último paquete de transporte de entrada, por el contrario, se termina el procesado para crear mapa_EP.

15 Conviene señalar que el método para crear un archivo de información de Fragmento varía en función del tipo de CPI del Clip. La Figura 48 es un diagrama de flujo explicativo que representa diferentes métodos de creación de un archivo de información de Fragmento para diferentes tipos de CPI de Clip. Según se ilustra en la Figura, el diagrama de flujo comienza con una etapa S101 en la que la unidad de control 17 realiza una determinación en cuanto a si mapa_EP ha de crearse, o no, como CPI. Si el resultado de la determinación realizada en la etapa S101 indica que mapa_EP ha de crearse como CPI, el flujo de procesado pasa a la etapa S102 para analizar la información de PTS, de STC y de PMT para los contenidos del flujo continuo AV. A continuación, en la siguiente etapa S103, la unidad de control 17 crea información de una secuencia del ATC, información de una secuencia del STC e información de una secuencia de programas. Por último, en la siguiente etapa S104, la unidad de control 17 crea mapa_EP.

25 Si el resultado de la determinación realizada en la etapa S101 indica que mapa_TU ha de crearse como CPI, por el contrario, el flujo de procesado pasa a una etapa S105 en la que ninguno de los contenidos del flujo continuo AV son analizados ni la información de una secuencia del STC y la información de una secuencia de programa son objeto de creación. A continuación, en la siguiente etapa S106, la unidad de control 17 crea información de una secuencia del ATC sobre la base de una temporización de entrada de un paquete de transporte. Por último, en la siguiente etapa S107, la unidad de control 17 crea mapa_TU. En uno u otro caso, un archivo de información de Fragmento se crea sin tener en cuenta el tipo de CPI según se describió anteriormente.

30 De este modo, un archivo de información de Fragmento se crea sea cual fuere el tipo de CPI.

Lo anterior tiene la repercusión siguiente. En el procesado para grabar un flujo continuo AV en un medio de grabación después de conocer el contenido de un flujo continuo AV, ATC, STC y secuencias de programas así como mapa_EP, se crean y graban en el medio de grabación. Ejemplos del procesado son una operación de grabación con conocimiento y una operación de grabación de auto-codificación. En la operación de grabación con conocimiento, los contenidos de un flujo continuo AV se analizan antes de que el flujo continuo AV se registre en un medio de grabación. En la operación de grabación de auto-codificación, por el contrario, una señal de vídeo de entrada se codifica por el propio aparato de grabación antes de grabarse en un medio de grabación. En el procesado para grabar un flujo continuo AV en un medio de grabación sin conocer el contenido del flujo, por el contrario, se crea una secuencia del ATC y mapa_TU y se graban en el medio de grabación. El procesado para grabar un flujo continuo AV en un medio de grabación sin conocer el contenido del flujo se refiere como una operación de grabación sin reconocimiento.

45 El procesado representado por el diagrama de flujo, ilustrado en la Figura 48 se puede interpretar como sigue. El diagrama de flujo se inicia con la etapa S101 para realizar una determinación en cuanto a si la operación de grabación es, o no, una operación de grabación con reconocimiento en donde los contenidos de un flujo continuo AV se analizan antes de que el flujo continuo AV se registre en un medio de grabación o una operación de grabación de auto-codificación en donde una señal de vídeo de entrada se codifica por el propio aparato de grabación antes de grabarse en un medio de grabación. Si la operación de grabación es una operación de grabación con reconocimiento o una operación de grabación de auto-codificación, en la etapa S102, se analizan los contenidos del flujo continuo AV, en la siguiente etapa S103, se crean una secuencia del ATC, una secuencia del STC y una secuencia del programa y en la etapa S104, se crea mapa_EP antes de un proceso para grabar los datos en el medio de grabación. Si el resultado de la determinación realizada en la etapa S101 indica que la operación de grabación es una operación de grabación sin reconocimiento, en donde un flujo continuo AV se graba en un medio de grabación, tal como está, sin conocer los contenidos del flujo, por el contrario, en la etapa S105, el flujo continuo AV no se somete a ningún análisis de los contenidos del flujo, en la siguiente etapa S106, se crea una secuencia del ATC y en la etapa final S107, se crea mapa_TU antes de un proceso para grabar los datos en el medio de grabación.

60 La Figura 49 es un diagrama de flujo explicativo que representa un método de creación de Real Playlist. El método se explica haciendo referencia a un diagrama de bloques de la Figura 43 que ilustra la configuración del aparato de grabación/reproducción de imágenes en movimiento 1.

Según se indica en la Figura, el diagrama de flujo comienza con una etapa S191 en donde la unidad de control 17 graba un flujo continuo AV de Clip.

65 A continuación, en la siguiente etapa S192, la unidad de control 17 crea Playlist() según se ilustra en la Figura 28.

- 5 La lista PlayLits() creada comprende objetos de PlayItem ilustrados en la Figura 33. Los objetos de PlayItem cubren todas las gamas reproducibles de Clip. Si Clip tiene mapa_EP, se crea mapa_EP-type PlayList ilustrado en la Figura 29. Si Clip tiene mapa_TU, por el contrario, se crea mapa_TU-type PlayList según se ilustra en la Figura 30. En el caso de mapa_EP-type PlayList, si Clip incluye discontinuidades de STC, de modo que PlayList() comprenda al menos 2 objetos de PlayList, la unidad de control 17 determina la condición_conexión entre los objetos de PlayList.
- 10 Posteriormente, en la siguiente etapa S193, la unidad de control 17 crea UIAppInfoPlayList(). UIAppInfoPlayList() incluye información utilizada para explicar los contenidos de PlayList al usuario. En esta forma de realización, se omite la explicación.
- 15 A continuación, en la siguiente etapa S194, la unidad de control 17 crea PlayList Mark. En esta forma de realización, se omite la explicación.
- Posteriormente, en la siguiente etapa S195, la unidad de control 17 crea MakersPrivateData. En esta forma de realización, se omite la explicación.
- 20 A continuación, en la siguiente etapa S196, la unidad de control 17 graba un archivo Real PlayList en el medio de grabación 10.
- Como es evidente a partir de la descripción anterior cada vez que se graba nuevamente un flujo continuo AV de Clip, se crea un archivo Real PlayList.
- La Figura 50 es un diagrama de flujo que representa un método de creación de Virtual PlayList.
- 25 Según se ilustra en la Figura, el diagrama de flujo comienza con una etapa S211 en donde el usuario introduce una demanda de reproducción de la Real PlayList grabada en el disco que sirve como el medio de grabación 10 a la interfaz del usuario. en la demanda, el usuario especifica un intervalo de reproducción indicado por los puntos IN y OUT introducidos en la interfaz de usuario.
- 30 A continuación, en la siguiente etapa S212, la unidad de control 17 realiza una determinación en cuanto a si el usuario ha acabado, o no, la operación para especificar intervalos de reproducción, es decir, las gamas a reproducirse. Si el usuario no ha acabado la operación, es decir, si el usuario desea especificar otro intervalo de reproducción siguiente a la gama especificada a reproducirse a continuación, el flujo del procesado retrocede a la etapa S211.
- 35 Si el resultado de la determinación realizada en la etapa S212 indica que el usuario ha acabado la operación para especificar intervalos de reproducción, por el contrario, el flujo de procesado pasa a una etapa S213.
- 40 En la etapa S213, se determina una condición de conexión entre dos intervalos de reproducción consecutivos a reproducirse continuamente por el usuario a través de la interfaz de usuario o por la unidad de control 17. Una condición de conexión entre dos intervalos de reproducción consecutivos se refiere como connection_condition.
- 45 A continuación, en la siguiente etapa S214, el usuario especifica información de sub-ruta (audio después de la grabación) a través de la interfaz del usuario. Si el usuario no desea crear una sub-ruta, se salta el procesado de esta etapa. La información de sub-ruta es información memorizada en SubPlayItem en PlayList. Puesto que la información de sub-ruta no es importante para la presente invención, sin embargo, se omite la explicación de la información de sub-ruta.
- 50 Posteriormente, en la siguiente etapa S215, la unidad de control 17 crea PlayList(), ilustrada en la Figura 28, sobre la información de las gamas de reproducción especificadas por el usuario y connection_condition.
- 55 A continuación, en la siguiente etapa S216, la unidad de control 17 crea UIAppInfoPlayList(). UIAppInfoPlayList() incluye información utilizada para explicar los contenidos de PlayList al usuario. En esta forma de realización, se omite la explicación.
- 60 A continuación, en la siguiente etapa S217, la unidad de control 17 crea PlayList Mark. En esta forma de realización, se omite la explicación.
- Posteriormente, en la siguiente etapa S218, la unidad de control 17 crea MakersPrivateData. En esta forma de realización, se omite la explicación.
- 65 A continuación, en la siguiente etapa S219, la unidad de control 17 graba un archivo Virtual PlayList en el medio de grabación 10.
- Como es evidente a partir de la descripción anterior, cada vez que el usuario selecciona intervalos de reproducción deseados desde una gama de reproducción de Real PlayList grabada en el medio de grabación 10, se crea un

archivo Virtual PlayList.

La Figura 51 ilustra un diagrama de flujo que representa un método de reproducción de mapa_EP-type PlayList.

- 5 Según se ilustra en la figura, el diagrama de flujo comienza con una etapa S231 en donde la unidad de control 17 adquiere información de Info.dvr, el archivo de información de Fragmento, un archivo de PlayList y un archivo de miniatura con el fin de crear una pantalla de GUI que ilustre una lista de los objetos de PlayList memorizados en el disco que sirve como medio de grabación 10. La unidad de control 17 visualiza, entonces, la pantalla de GUI a través de la interfaz de usuario.
- 10 A continuación, en la siguiente etapa S232, la unidad de control 17 visualiza información para explicar cada objeto de PlayList en la pantalla de GUI sobre la base de UIAppInfoPlayList() de cada uno de los objetos de PlayList.
- 15 Posteriormente, en la siguiente etapa S233 el usuario realiza una demanda de reproducción de un objeto de PlayList seleccionado de entre los visualizados en la pantalla de GUI a través de la interfaz de usuario.
- A continuación, en la siguiente etapa S234, la unidad de control 17 adquiere el número de paquete fuente con punto de entrada más próximo, desde el punto de vista temporal, y que precede a IN_time desde STC-ID y un IN_time PTS del PlayItem presente.
- 20 Posteriormente, en la siguiente etapa S235, la unidad de control 17 realiza la lectura de datos del flujo continuo AV a partir del paquete fuente indicado por el número adquirido y que tiene el punto de entrada y suministra los datos al decodificador.
- 25 A continuación, en la siguiente etapa S236, la unidad de control 17 realiza el procesado de la conexión visualizada con PlayItem, desde un punto temporal precediendo al presente objeto de PlayItem en función del parámetro connection_condition en caso de que exista dicho PlayItem precedente.
- Posteriormente, en la siguiente etapa S237, la unidad de control 17 emite una orden al decodificador de AV 16 para iniciar una presentación visual con una imagen en el IN_time PTS.
- 30 A continuación, en la siguiente etapa S238, la unidad de control 17 emite una orden al decodificador de AV 16 para continuar la operación de decodificación de un flujo continuo AV.
- 35 Posteriormente, en la siguiente etapa S239, la unidad de control 17 realiza una determinación en cuanto a si la imagen actualmente visualizada es, o no, una imagen en el OUT_time PTS. Si el resultado de la determinación realizada en la etapa S239 indica que la imagen actualmente visualizada no es una imagen en el OUT_time PTS, el flujo del procesado pasa a la etapa S240. En la etapa S240, se visualiza la imagen actual. A continuación, el flujo del procesado retrocede a la etapa S238. Si el resultado de la determinación realizada en la etapa S239 indica que la imagen actualmente visualizada es una imagen en el OUT_time PTS, por el contrario, el flujo del procesado pasa a una etapa S241.
- 40 En la etapa S241, la unidad de control 17 realiza una determinación en cuanto a si el objeto PlayItem actual es, o no, el último objeto PlayItem en PlayList. Si el resultado de la determinación realizada en la etapa S241 indica que el objeto PlayItem actual no es el último objeto PlayItem en la lista PlayList, el flujo de procesado retrocede a la etapa S234. Si el resultado de la determinación realizada en la etapa S241 indica que el objeto PlayItem actual es el último objeto PlayItem en PlayList, por el contrario, se termina el procesado para reproducir PlayList.
- 45 Haciendo referencia a la Figura 52 que ilustra un diagrama de flujo que representa un proceso de edición para reducir al mínimo mapa_EP-type PlayList, la siguiente descripción explica un procedimiento de un método para actualizar Clip y PlayList como parte del proceso de edición.
- 50 Según se ilustra en la Figura, el diagrama de flujo comienza con una etapa S261 en donde la unidad de control 17 busca la gama de reproducción de Real PlayList para al menos un intervalo de reproducción no utilizado en cualquier Virtual PlayList y trata dichos intervalos de reproducción como las gamas que han de eliminarse.
- 55 A continuación, en la siguiente etapa S262, la unidad de control 17 adquiere los tiempos de inicio y final del intervalo a eliminarse de la gama de reproducción de Real PlayList.
- 60 Posteriormente, en la siguiente etapa S263, la unidad de control 17 determina el paquete (dirección) de inicio de eliminación del flujo continuo AV de Clip y el paquete final de eliminación (dirección) correspondiente al intervalo de tiempo anterior sobre la base de mapa_EP.
- 65 A continuación, en la siguiente etapa S264, la unidad de control 17 añade una nueva secuencia del ATC que comienza con un paquete fuente inmediatamente siguiente al anterior paquete final de eliminación para SequenceInfo. Es decir, la unidad de control 17 establece el número del paquete fuente inmediatamente siguiente al anterior paquete final de eliminación en SPN_ATC_start.

5 Posteriormente, en la siguiente etapa S265, la unidad de control 17 actualiza el número (SPN_STC_start) de un paquete de inicio de una secuencia del STC existente en una secuencia del ATC en un flujo continuo AV después de la eliminación. Es decir, el valor de SPN_STC_start se cambia a uno nuevo que esté conforme con el flujo continuo AV después de la eliminación.

10 A continuación, en la siguiente etapa S266, la unidad de control 17 determina dicho offset_STC_id que el valor de stc_id para la secuencia del STC existente en la secuencia del ATC en el flujo continuo AV de Clip después de la eliminación que queda sin cambiar.

15 Posteriormente, en la siguiente etapa S267, se actualiza ProgramInfo para el flujo continuo AV de Clip después de la eliminación, si fuera necesario. Es decir, si la secuencia de programas comienza en la gama eliminada anteriormente descrita, el número del paquete fuente de inicio de la secuencia de programas se cambia al número de paquete fuente inmediatamente después del paquete final suprimido.

20 A continuación, en la siguiente etapa S268, la unidad de control 17 actualiza mapa_EP a uno nuevo que está conforme con el flujo continuo AV de Clip después de la eliminación. En este procesado, se elimina la entrada de mapa_EP que se refiere a un flujo en el intervalo eliminado y el valor de un número de paquete fuente en mapa_EP, es decir, SPN_EP_start de mapa_EP, se actualiza a uno nuevo que está conforme con el flujo continuo AV de Clip después de la eliminación.

Posteriormente, en la siguiente etapa S269, la unidad de control 17 elimina los datos del flujo continuo AV de Clip en el intervalo indicado por los paquetes de inicio y final anteriores.

25 A continuación, en la siguiente etapa S270, la unidad de control 17 actualiza el archivo de información de Fragmento para reflejar el procesado anteriormente descrito y graba el archivo de información de Fragmento actualizado. Según se ilustra en la Figura 8, el archivo de información de Fragmento incluye, entre otros parámetros, ClipInf(), SequenceInfo(), ProgramInfo() y CPI(). De este modo, la información sobre la secuencia del ATC antes citada y la información sobre la secuencia del STC antes citada se graban en el medio de grabación 10.

30 Posteriormente, en la siguiente etapa S271, la unidad de control 17 actualiza el archivo Real Playlist en cuanto a cubrir un intervalo de reproducción que no incluye ninguna gama de reproducción del intervalo suprimido anteriormente descrito y graba el archivo de Real Playlist actualizado.

35 A continuación, en la siguiente etapa S272, la unidad de control 17 realiza una determinación en cuanto a si el proceso de edición ha llegado, o no, a su fin, es decir, si todos los intervalos de reproducción buscados en la etapa S261 han sido suprimidos o no. Si el resultado de la determinación indica que el intervalo de reproducción buscado en la etapa S261 no ha sido suprimido en su totalidad, el flujo de procesado retrocede a la etapa S262. Si el resultado de la determinación indica que los intervalos de reproducción buscados en la etapa S261 han sido suprimidos en su totalidad, por el contrario, se termina el procesado de minimización.

40 El procesado realizado en la etapa S263 se explica en detalle para Clip con CPI del tipo mapa_EP.

45 La Figura 53 es un diagrama explicativo que ilustra un archivo de flujo continuo AV original y un archivo de flujo continuo AV típico obtenido como resultado de un proceso de edición en donde el flujo de una gama de reproducción parcial se elimina del archivo original. Se supone que, antes del proceso de edición, Virtual Playlist apunta a IN_time y OUT_time, que están en el flujo continuo AV. Cuando parte del flujo no utilizada por Virtual Playlist se elimina en el así denominado proceso de edición de minimización, el flujo continuo AV original se cambia al flujo después de la edición ilustrado en la Figura 53. Según se ilustra en esta Figura, los datos desde el inicio del flujo continuo AV original al punto X y los datos desde el punto Y al final del flujo continuo AV original se eliminan del flujo continuo AV original. La siguiente descripción explica un método típico de determinación de puntos X e Y.

50 La Figura 54 es un diagrama explicativo que muestra la eliminación de los datos de flujo innecesarios que preceden a IN_time de un flujo continuo AV sin analizar los contenidos del flujo continuo AV. Playlist apunta a un punto IN en el flujo continuo AV original. La Figura ilustra también mapa_EP del flujo continuo AV. Con el fin de decodificar una imagen apuntada por el punto IN, una imagen I que comienza a partir de la dirección ISA2 se requiere a este respecto. Además, después del punto X, los paquetes PAT, PMT y PCR son requeridos. Pts1 es un PTS con SPN_EP_start = ISA1 y pts2 es un PTS con SPN_EP_start = ISA2. Si una diferencia en la base de tiempos del sistema entre pts1 y pts2 es al menos igual a 100 milisegundos, los paquetes PAT, PMT y PCR existen entre las direcciones ISA1 e ISA2. Esto se mantiene cierto para al menos los casos de SESF, DVB, ATSC e ISDB. Por ello, el punto X se determina para ser un punto antes de la dirección ISA1. Además, el punto X debe estar en el entorno límite de una unidad alineada.

60 Sin analizar los contenidos del flujo continuo AV, el aparato de grabación/reproducción de imagen en movimiento 1 es capaz de determinar el punto X utilizando mapa_EP mediante la ejecución de las etapas siguientes:

65

1) Encontrar SPN_EP_start que tiene un valor PTS de tiempo de visualización más próximo al que precede al PTS de un tiempo IN en la base de tiempos del sistema.

5 2) Encontrar SPN_EP_start que tiene un valor de PTS de tiempo de visualización que precede al valor PTS del tiempo de visualización encontrado en la etapa 1) en al menos 100 milisegundos.

3) Determinar el punto X en una posición que precede a SPN_EP_start encontrado en la etapa 2). El punto X debe estar en un entorno límite de una unidad alineada.

10 Este método es simple porque, con el fin de determinar el punto X, no es necesaria la lectura de los datos del flujo continuo AV ni analizar los datos. En algunos casos, sin embargo, datos innecesarios para la reproducción de Playlist pueden permanecer inevitablemente en el flujo continuo AV después de la edición. Si los datos del flujo continuo AV son objeto de lectura y se analiza en la determinación del punto X, se pueden eliminar efectivamente datos innecesarios para la reproducción de Playlist.

15 La Figura 55 es un diagrama explicativo que ilustra un método de eliminación de datos de flujos innecesarios después de un punto OUT sin analizar los datos del flujo continuo AV. Playlist apunta al punto OUT en el flujo continuo AV original. La Figura ilustra también mapa_EP del flujo continuo AV. Se supone que una secuencia de vídeo que comienza desde SPN_EP_start = ISA4 es la siguiente serie de tramas de imagen:

20 I2 B0 B1 P5 ...

25 en donde los símbolos I, P y B indican I, P y B imágenes, respectivamente. Los números unidos a los símbolos I, P y B, como sufijos son números del orden de la presentación visual. Si la unidad de grabación no analiza los datos del flujo continuo AV en este procesado, el aparato de grabación/reproducción de imagen en movimiento 1 no conocerá la información sobre una imagen referida por el PTS de OUT_time. La información incluye el tipo de codificación de la imagen y una referencia temporal. El PTS de OUT_time puede referirse a la imagen B0 o B1. Si los datos del flujo continuo AV no son analizados, el aparato de grabación/reproducción de imagen en movimiento 1 no tiene conocimiento del hecho de que el PTS de OUT_time se refiere a la imagen B0 o B1. En este caso, con el fin de decodificar la imagen B0 y B1, se requiere la imagen I2. De este modo, la PTS de la imagen I2 es mayor que la PTS de OUT_time. Es decir $OUT_time < pts4$ en donde la notación pts4 indica el PTS de la imagen I2. A pesar del hecho de que la PTS de la imagen I2 es mayor que la PTS de la imagen OUT_time, la imagen I2 se requiere para decodificar las imágenes B0 y B1.

35 De este modo, el punto Y se determina en una posición detrás de la dirección ISA5. ISA5 es el valor de SPN_EP_start que sucede inmediatamente a la dirección ISA4 en mapa_EP. El punto Y debe estar también en un entorno límite de una unidad alineada.

40 Sin analizar los contenidos del flujo continuo AV, el aparato de grabación/reproducción de imagen en movimiento 1 es capaz de determinar el punto Y utilizando mapa_EP mediante la ejecución de las etapas siguientes de:

1) Encontrar SPN_EP_start que tiene un valor de PTS del tiempo de visualización más próximo y sucediendo al PTS de un OUT_time en la base de tiempos del sistema.

45 2) Encontrar SPN_EP_start que tenga un valor de PTS de tiempo de visualización inmediatamente a continuación del valor de PTS de tiempo de visualización encontrado en la etapa 1

3) Determinar el punto Y en una posición siguiente a SPN_EP_start encontrado en la etapa 2.

50 El punto Y debe estar en un entorno límite de una unidad alineada.

55 Este método es simple, por cuanto que, con el fin de determinar el punto Y, no es necesario efectuar la lectura los datos del flujo continuo AV y analizar los datos. En algunos casos, sin embargo, pueden permanecer inevitablemente datos innecesarios para la reproducción de Playlist en el flujo continuo AV después de la edición. Si los datos del flujo continuo AV son objeto de lectura y análisis en la determinación del punto Y, se pueden eliminar efectivamente datos innecesarios para la reproducción de Playlist.

60 La Figura 56 ilustra un diagrama de flujo explicativo que representa un método para reproducir Playlist de un tipo de mapa_TU.

Los elementos de procesado realizados en las etapas S300 a S302 son los mismos que los realizados en las etapas S231 a S232 del diagrama de flujo ilustrado en la Figura 51.

65 Más en detalle, según se ilustra en la Figura 56, el diagrama de flujo comienza con la etapa S300 en donde la unidad de control 17 adquiere información de Info.dvr, un archivo de información de Fragmento, un archivo de Playlist y un archivo de miniatura para crear una pantalla de GUI para visualizar una lista de objetos de Playlist memorizados en

un disco utilizado como el medio de grabación 10.

A continuación, en la siguiente etapa S301 la unidad de control 17 visualiza cada uno de los objetos de PlayList visualizados en la pantalla de GUI sobre la base UIAppInfoPlayList() de cada objeto PlayList.

5 Posteriormente, en la siguiente etapa S302, el usuario realiza una demanda de reproducción de un objeto PlayList seleccionado de entre los visualizados en la pantalla de GUI a través de la interfaz de usuario.

10 A continuación, en la siguiente etapa S303, la unidad de control 17 adquiere la dirección de un punto de entrada del flujo continuo AV haciendo referencia a la información de TU_rmap. Para ser más concretos, la unidad de control 17 adquiere el número de un paquete fuente con un punto de entrada más próximo temporalmente y que precede al tiempo de llegada de IN_time presente objeto PlayItem. Detalles de este procesado se describirán más adelante. Posteriormente, en la siguiente etapa S304, la unidad de control 17 reproduce un paquete indicado por el número del paquete con dicho punto de entrada y suministra el paquete reproducido al decodificador AV 16.

15 A continuación, en la siguiente etapa S305, la unidad de control 17 realiza una determinación de si la marca de tiempo de llegada del presente paquete es, o no, la misma que o posterior a la del paquete de OUT_time. Si el resultado de la determinación indica que la marca de tiempo de llegada del presente paquete no es la misma ni es posterior a la del paquete de OUT_time, el flujo del procesado pasa a la etapa S306. En la etapa S306, la unidad de control 17 reproduce el siguiente paquete y suministra el siguiente paquete al decodificador de AV 16. A continuación, el flujo del procesado retrocede a la etapa S305. Si el resultado de la determinación realizada en la etapa S305 indica que la marca de tiempo de llegada del presente paquete es la misma o posterior a la del paquete de OUT_time, por el contrario, el flujo del procesado pasa a una etapa S307.

25 En la etapa S307, la unidad de control 17 realiza una determinación en cuanto a si el objeto de PlayItem presente, es o no, el último objeto de PlayItem. Si el resultado de la determinación indica que el objeto de PlayItem presente no es el último eje de PlayItem, el flujo de procesado retrocede a la etapa S303. Si el resultado de la determinación indica que el objeto de PlayItem presente es el último objeto de PlayItem, por el contrario, la unidad de control 17 finaliza la reproducción de objetos de PlayList.

30 A continuación, los detalles del procesado realizado en la etapa S303 del diagrama de flujo ilustrado en la Figura 56 se explican haciendo referencia a un diagrama de flujo ilustrado en la Figura 57.

35 Según se ilustra en la Figura 57, el diagrama de flujo comienza con S400 en donde la unidad de control 17 adquiere el valor máximo de atc_id que hace que se mantenga verdadera la siguiente relación entre IN_time de PlayItem y offset_arrival_time[atc_id] de mapa_TU():

$$\text{offset_arrival_time[atc_id]} \leq \text{IN_time}$$

40 Hace referencia a la sintaxis ilustrada en la Figura 26.

A continuación, en la siguiente etapa S401, la unidad de control 17 adquiere dicho valor de i que el tiempo de inicio de la i -ésima unidad de tiempo en una secuencia del ATC especificada por el anterior atc_id (TU_start_time[atc_id][i]) es más próximo temporalmente y procede a IN_time. Se hace referencia a la ecuación (2) dada anteriormente.

Posteriormente, en la siguiente etapa S402, la unidad de control 17 utiliza SPN_time_unit_start [atc_id][i] para el i anterior como la dirección de un punto de entrada. A continuación, se finaliza el procesado.

50 Haciendo referencia a un diagrama de flujo ilustrado en la Figura 58, la siguiente descripción explica un método para actualizar Clip y PlayList en un proceso de edición para minimizar mapa_TU-type PlayList.

Según se ilustra en la Figura, el diagrama de flujo comienza con S500 en donde la unidad de control 17 busca la gama de reproducción de Real PlayList para al menos un intervalo de reproducción no utilizado en cualquier Virtual PlayList y trata dichos intervalos de reproducción como gamas que han de eliminarse.

60 A continuación, en la siguiente etapa S501, la unidad de control 17 adquiere el tiempo de inicio (un tiempo de llegada) y el tiempo final (un tiempo de llegada) de un intervalo a eliminarse de la gama de reproducción de Real PlayList.

Posteriormente, en la siguiente etapa S502, la unidad de control 17 determina el paquete de inicio de eliminación (dirección) del flujo continuo AV de Clip y el paquete (dirección) de final de eliminación correspondiente al intervalo de tiempo anterior sobre la base de mapa_TU. Detalles de este procesado se describirán más adelante.

65 A continuación, en la siguiente etapa S503, la unidad de control 17 añade una nueva secuencia del ATC que comienza con un paquete fuente inmediatamente después del paquete de final de eliminación anterior para

SequencelInfo. Es decir, la unidad de control 17 establece el número del paquete fuente inmediatamente siguiente al paquete de final de eliminación anterior en SPN_ATC_start.

5 Posteriormente, en la siguiente etapa S504, la unidad de control 17 actualiza mapa_TU con el fin de estar conforme con un flujo continuo AV de Clip después de la eliminación como sigue:

- Una entrada de datos de SPN_time_unit_start para el intervalo del flujo continuo AV a eliminarse es objeto de eliminación.

10 - El tiempo de inicio de la primera unidad de tiempo en la nueva secuencia del ATC anterior se añade a mapa_TU como offset_arrival_time para esta secuencia del ATC.

15 - El valor del número de paquete fuente en mapa_TU es actualizado o, para ser más concretos, se cambia SPN_time_unit_start de mapa_TU con el fin de estar conforme con el flujo continuo AV de Clip después de la eliminación.

A continuación, en la siguiente etapa S505 la unidad de control 17 elimina los datos del flujo continuo AV de Clip en el intervalo indicado por los paquetes de inicio y final anteriores.

20 Posteriormente, en la siguiente etapa S506, la unidad de control 17 actualiza y graba el archivo de información de Fragmento para reflejar el procesado anterior. Según se indica en la Figura 8, el archivo de información de Fragmento incluye, entre otra información, ClipInf(), SequencelInfo(), ProgramInfo() y CPI(). De este modo, la información sobre la secuencia del ATC antes citada se graba en el medio de grabación 10.

25 A continuación, en la siguiente etapa S507, la unidad de control 17 actualiza y graba el archivo de Real Playlist con el fin de cubrir el intervalo de reproducción, con la excepción de la gama de reproducción del intervalo de reproducción suprimido.

30 Posteriormente, en la siguiente etapa S508, la unidad de control 17 realiza una determinación en cuanto a si todas las gamas examinadas en la etapa S500 como objeto de eliminación han sido eliminadas o no. Si el resultado de la determinación indica que las gamas examinadas para eliminarse no han sido eliminadas en su totalidad, el flujo del procesado retrocede a la etapa S501. Si el resultado de la determinación indica que las gamas examinada a eliminarse han sido eliminadas en su totalidad, por el contrario, se termina el procesado de minimización.

35 A continuación, se explican detalles del procesado realizado en la etapa S502 del diagrama de flujo ilustrado en la Figura 58, haciendo referencia a un diagrama de flujo ilustrado en la Figura 59.

40 Según se indica en la Figura 59, el diagrama de flujo comienza con una etapa S600 en donde la unidad de control 17 adquiere atc_id de una secuencia del ATC que incluye los tiempos de inicio y final de un intervalo que ha de eliminarse.

45 A continuación, en la siguiente etapa S601, la unidad de control 17 adquiere dicho valor de i que es el tiempo de inicio de la i -ésima unidad de tiempo en la secuencia del ATC especificada por atc_id anterior (TU_start_time[atc_id][i]) es el más próximo, en el tiempo, y siguiente al tiempo de inicio del intervalo a eliminarse. Se hace referencia a la ecuación (2) dada con anterioridad.

Posteriormente, en la siguiente etapa S602, la unidad de control 17 utiliza SPN_time_unit_start [atc_id] [i] para el valor anterior de i como la dirección de un paquete de inicio a eliminarse.

50 A continuación, en la siguiente etapa S603, la unidad de control 17 adquiere dicho valor de j que es el tiempo de inicio de la j -ésima unidad de tiempo en la secuencia del ATC especificada por el atc_id anterior (TU_start_time[atc_id][j]) es el más próximo, temporalmente y que precede al tiempo final del intervalo a eliminarse. Se hace referencia a la ecuación (2) dada con anterioridad.

55 Posteriormente, en la siguiente etapa S604, la unidad de control 17 utiliza SPN_time_unit_start [atc_id][j] para el valor anterior de j como la dirección de un paquete final a eliminarse.

La Figura 60 es un diagrama de flujo único que representa el procesado para actualizar un archivo de información de Fragmento en un proceso de edición para minimizar mapa_EP-type Playlist y mapa_TU-type Playlist.

60 Según se ilustra en la Figura 60, del diagrama de flujo comienzo con la etapa S701 en donde la unidad de control 17 realiza una determinación en cuanto a si Playlist es, o no, mapa_EP-type Playlist. Si el resultado de la determinación indica que Playlist es mapa_EP-type Playlist, el flujo de procesado pasa a una etapa S702. En la etapa S702, la unidad de control 17 actualiza el archivo de información de Fragmento con el fin de estar conforme con la eliminación de una parte del flujo continuo AV. Para más detalles, la unidad de control 17 actualiza:

65

La información de secuencia del ATC (en la etapa S264 del diagrama de flujo ilustrado en la Figura 52),

La información de secuencia del STC (en las etapas S265 y S266 del diagrama de flujo ilustrado en la Figura 52) y la información de secuencia de programas, si fuera necesario (en la etapa S267 del diagrama de flujo ilustrado en la Figura 52).

A continuación, en la siguiente etapa S703, la unidad de control 17 actualiza la información de mapa_EP con el fin de estar conforme con la eliminación de una parte del flujo continuo AV. En el diagrama de flujo ilustrado en la Figura 52, este procesado se realiza en la etapa S268. Entonces se finaliza el procesado.

Si el resultado de la determinación realizada en la etapa S701 indica que PlayList mapa_TU-type PlayList, por el contrario, el flujo del procesado pasa a una etapa S704. En la etapa S704, la unidad de control 17 actualiza la información de la secuencia del ATC del archivo de información de Fragmento con el fin de estar conforme con la eliminación de una parte del flujo continuo AV. En el diagrama de flujo ilustrado en la Figura 58, este procesado se realiza en la etapa S503.

Posteriormente, en la siguiente etapa S705, la unidad de control 17 actualiza la información de mapa_TU con el fin de estar conforme con la eliminación de una parte del flujo continuo AV. En el diagrama de flujo ilustrado en la Figura 58, este procesado se realiza en la etapa S504. Entonces, se finaliza el procesado.

La siguiente descripción explica un método de establecimiento de un valor del parámetro connection_condition ilustrado en la Figura 29 para datos de PlayItem también ilustrados en la Figura 29 de PlayList según se representa en la Figura 28 en un caso en que se generan discontinuidades de ATC y STC cuando se graba el flujo continuo AV de mapa_EP-type PlayList.

Ante todo, se explica una relación entre un flujo continuo AV que tiene discontinuidades de ATC y STC y PlayItem.

La Figura 61 es un diagrama explicativo que ilustra un caso en donde mapa_EP-type PlayList se divide en dos objetos de PlayItem en un entorno de separación entre las dos secuencias ATC. En el entorno límite entre dos secuencias ATC, la secuencia del STC es también dividida. Puesto que PlayItem se refiere a una secuencia del STC continua, dicho PlayItem se divide también en 2 objetos PlayItem en el entorno límite de la secuencia del STC. En este caso, el valor del parámetro connection_condition se pone a 1 para indicar que el objeto de PlayItem actual para el objeto de PlayItem anterior es dicho estado.

La Figura 62 es un diagrama explicativo que ilustra un caso en el que mapa_EP-type PlayList se divide en 2 objetos PlayItem en un entorno límite entre 2 secuencias STC en una secuencia del ATC continua. La secuencia del STC se divide en 2 secuencias STC en una discontinuidad de STC en la secuencia del ATC continua. Puesto que PlayItem se refiere a una secuencia del STC continua, dicho PlayItem se divide también en 2 objetos PlayItem en el entorno límite de la secuencia del STC. En este caso, el valor de parámetro connection_condition se pone en 2 para indicar el hecho de que el objeto de PlayItem actual está conectado al objeto de PlayItem anterior en dicho estado.

La Figura 63 es un diagrama de flujo que representa un método para crear datos de mapa_EP-type PlayList en un proceso para grabar un flujo continuo AV, en donde se generan discontinuidades de ATC y STC en el curso del proceso de grabación.

Según se indica en la Figura, el diagrama de flujo comienza con una etapa S800 en donde la unidad de control 17, ilustrada en la Figura 43, establece parámetros como sigue $n = 0$, $m = 0$ y $is_ATC_sequence = 1$, en donde el parámetro n es el número de una secuencia del ATC generada en el curso del proceso de grabación, el parámetro m es el número de una secuencia del STC generada en el curso del proceso de grabación y el parámetro $is_ATC_sequence$ es un indicador que muestra si se ha generado, o no, una discontinuidad de ATC.

A continuación, en la siguiente etapa S801, la unidad de control 17 inicia la n -ésima secuencia del ATC desde un paquete que se está grabando actualmente.

Posteriormente, en la siguiente etapa S802, la unidad de control 17 inicia la m -ésima secuencia del STC y el m -ésimo objeto PlayItem.

A continuación, en la siguiente etapa S803, la unidad de control 17 determina el parámetro connection_condition del m -ésimo objeto PlayItem como sigue:

Para $is_ATC_change = 1$, connection_condition se pone a 1.

Para $is_ATC_change = 0$, connection_condition se pone a 2.

Conviene señalar que, para el primer objeto PlayItem ($m = 0$), connection_condition se pone a 1 a pesar del hecho de que el estado operativo es diferente del ilustrado en la Figura 62.

Posteriormente, en la siguiente etapa S804, la unidad de control 17 analiza el PTS de un vídeo incluido en el flujo continuo AV que se está grabando. La información de PTS se utiliza como información para adquirir IN_time y OUT_time de PlayItem.

5 A continuación, en la siguiente etapa S805, la unidad de control 17 realiza una determinación en cuanto a si se ha detectado, o no, una discontinuidad. Si no se ha detectado ninguna discontinuidad, el flujo del procesado retrocede a la etapa S804 para continuar el procesado de esta etapa. Si se ha detectado una discontinuidad, por el contrario, el flujo del procesado pasa a una etapa S806.

10 En la etapa S806, la unidad de control 17 realiza una determinación en cuanto a si la discontinuidad detectada es, o no, una discontinuidad de STC. La discontinuidad se determina como siendo una discontinuidad de STC en la misma manera que se explicó anteriormente haciendo referencia a la Figura 45. Si el resultado de la determinación realizada en la etapa S806 indica que la discontinuidad detectada es una discontinuidad de STC, el flujo de procesado pasa a una etapa S807. En este caso, la unidad de control 17 determina que un estado operativo ilustrado en la Figura 62 ha sido obtenido como resultado. En este estado, se ha generado una discontinuidad de STC pero no se ha detectado ninguna generación de una discontinuidad de ATC. En la etapa S807, se realiza el procesado siguiente:

20 (1) IN_time y OUT_time del m-ésimo objeto de PlayItem se adquieren.

(2) m++

(3) is_ATC_change = 0

25 A continuación, el flujo de procesado retrocede a la etapa S802 en donde se realiza el procesado siguiente. En este momento, puesto que is_ATC_change = 0, el parámetro connection_condition de PlayItem se pone a 2 en la etapa S803.

30 Si el resultado de la determinación realizada en la etapa S806 indica que la discontinuidad detectada no es una discontinuidad de STC, por el contrario, el flujo de procesado pasa a una etapa S808. En la etapa S808, la unidad de control 17 realiza una determinación en cuanto a si la discontinuidad detectada ha sido causada, o no, por una pausa de grabación/liberación de pausa. La pausa de grabación/liberación de pausa es un evento operativo en donde el proceso de grabación es temporalmente suspendido antes de reanudarse más adelante.

35 Si el resultado de la determinación realizada en la etapa S808 indica que la discontinuidad detectada ha sido causada por una pausa de grabación/liberación de pausa, la unidad de control 17 determina que se ha generado una discontinuidad de ATC y se generará también una discontinuidad de STC muy similar al estado ilustrado en la Figura 61 debido a que ha hecho de que fue suspendido una vez el proceso de grabación. En este caso, el flujo de procesado pasa a una etapa S809 en donde se realiza el procesado siguiente:

(1) IN_time y OUT_time de m-ésimo objeto PlayItem se adquiere.

(2) m++

(3) n++

(4) is_ATC_change = 1

50 Entonces, el flujo de procesado retrocede a la etapa S801 en donde se realiza el procesado siguiente. En este momento, puesto que is_ATC_change = 1, el parámetro connection_condition de PlayItem se pone a 1 en la etapa S803.

55 Si el resultado de la determinación realizada en la etapa S808 indica que la discontinuidad detectada no ha sido causada por una pausa de grabación/liberación de pausa, por el contrario, se finaliza el procesado para grabar el flujo continuo AV.

60 La siguiente descripción explica un método de establecimiento de un valor de connection_condition, ilustrado en la Figura 29, para datos de PlayItem también conocidos en la Figura 29 de PlayList que se representa en la Figura 28 en un caso en donde una discontinuidad de ATC se genera cuando se graba el flujo continuo AV de un PlayList tipo mapa_TU.

Ante todo, se explica una relación entre un flujo continuo AV que tiene una discontinuidad de ATC y PlayItem.

65 La Figura 64 es un diagrama explicativo que ilustra un caso en donde mapa_TU-type PlayList se divide en 2 objetos PlayItem en un entorno límite entre 2 secuencias ATC. Puesto que PlayItem se refiere a una secuencia del

STC continua, dicho PlayItem se divide también en 2 objetos PlayItem en el entorno límite de la secuencia del STC. En este caso, el valor del parámetro connection_condition se pone a 1 para indicar el hecho de que el objeto de PlayItem actual está conectado al objeto PlayItem anterior en dicho estado.

5 La Figura 65 es un diagrama de flujo que representa un método para crear datos de mapa_TU-type Playlist en un proceso para grabar un flujo continuo AV, en donde se genera una discontinuidad de ATC en el curso del proceso de grabación.

10 Según se ilustra en la Figura, el diagrama de flujo comienza con una etapa S831 en donde la unidad de control 17, ilustrada en la Figura 43, establece un parámetro n en 0 ($n = 0$). El parámetro n es el número de una secuencia del ATC generada en el curso del proceso de grabación.

15 A continuación, en la siguiente etapa S832, la unidad de control 17 inicia la n-ésima secuencia del ATC a partir del paquete actualmente en grabación.

Posteriormente, en la siguiente etapa S833, la unidad de control 17 inicia el n-ésimo objeto PlayItem.

20 A continuación, en la siguiente etapa S834, la unidad de control 17 establece el parámetro connection_condition del n-ésimo objeto PlayItem en 1. Conviene señalar que, también para el primer objeto PlayItem ($n = 0$), el parámetro connection_condition se pone a 1 a pesar del hecho de que el estado operativo es diferente del ilustrado en la Figura 64.

25 Posteriormente, en la siguiente etapa S835, la unidad de control 17 adquiere la marca de tiempo de llegada del paquete del flujo continuo AV que se está grabando. La información de la marca de tiempo de llegada se utiliza para adquirir IN_time y OUT_time de PlayItem.

30 A continuación, en la siguiente etapa S836, la unidad de control 17 realiza una determinación en cuanto a si se ha detectado, o no, una discontinuidad. Si no se ha detectado ninguna discontinuidad, el flujo de procesado retrocede a la etapa S835 en donde se repite el procesado de esta etapa. Si el resultado de la determinación realizada en la etapa S836 indica que se ha detectado una discontinuidad, por el contrario, el flujo de procesado pasa a una etapa S837.

35 En la etapa S837, la unidad de control 17 realiza una determinación de si la discontinuidad detectada ha sido causada, o no, por una pausa de grabación/liberación de pausa. La pausa de grabación/liberación de pausa es un evento operativo en donde el proceso de grabación se suspende temporalmente antes de reanudarse más adelante.

40 Si el resultado de la determinación realizada en la etapa S837 indica que la discontinuidad detectada ha sido causada por una pausa de grabación/liberación de pausa, la unidad de control 17 determina que se ha generado una discontinuidad de ATC debido al hecho de que fue suspendido una vez el proceso de grabación. En este caso, el flujo de procesado pasa a una etapa S838 en donde se realiza el procesado siguiente:

(1) Se adquieren IN_time y OUT_time del n-ésimo PlayItem.

(2) $n++$

45 Entonces, el flujo de procesado retrocede a la etapa S832 en donde se realiza el procesado siguiente. En este momento, el parámetro connection_condition de PlayItem se pone a 1 en la etapa S834. (Referirse al estado ilustrado en la Figura 63).

50 Si el resultado de la determinación realizada en la etapa S837 indica que la discontinuidad detectada no ha sido causada por una pausa de grabación/liberación de pausa, por el contrario, el procesado para grabar el flujo continuo AV se finaliza en este momento.

55 La Figura 66 es un diagrama de flujo que representa un método para reproducir mapa_EP-type Playlist sobre la base del valor del parámetro connection_condition.

Según se ilustra en la Figura, el diagrama de flujo comienza con una etapa S851 en donde la unidad de control 17, ilustrada en la Figura 43, realiza la lectura de los datos de un archivo Playlist.

60 A continuación, en la siguiente etapa S852, la unidad de control 17 establece un parámetro K en 0 ($K = 0$). El parámetro k es el número de datos de PlayItem incluidos en Playlist como una entrada.

Posteriormente, en la siguiente etapa S853, la unidad de control 17 adquiere connection_condition del k-ésimo objeto de PlayItem actualmente destinado a reproducirse.

65

A continuación, en la siguiente etapa S854, la unidad de control 17 realiza una determinación en cuanto a si el valor de connection_condition es 2 o no lo es. Si el valor de connection_condition es 2 el flujo de procesado pasa a una etapa S855.

5 En la etapa S855, la unidad de control 17 tiene conocimiento de que es posible una lectura continua de datos de AV de una secuencia del ATC después de los datos de AV del (K-1)-ésimo objeto PlayItem. Con mayor detalle, la unidad de control 17 tiene conocimiento de que es posible la lectura continua de datos del flujo continuo AV a través de una discontinuidad de STC en la secuencia del ATC continua, puesto que PlayItem simplemente se divide en la discontinuidad de STC en un estado como el ilustrado en la Figura 62. De forma más concreta, es evidente que, en el modelo de reproducción ilustrado en la Figura 7, el valor del reloj de contador de tiempos de llegada 255 puede hacerse continuo incluso en el caso de un estado de extensión sobre una discontinuidad de STC.

Si el resultado de la determinación realizada en la etapa S854 indica que el valor de connection_condition no es 2, por el contrario, el flujo de procesado pasa a una etapa S856.

15 En la etapa S856, la unidad de control 17 tiene conocimiento de que es necesario efectuar la reposición del contador de ATC de la unidad de reproducción después de la lectura de datos de AV del (K-1)-ésimo objeto PlayItem, pero antes de iniciar la lectura de datos de AV del k-ésimo objeto PlayItem. Con más detalle, la unidad de control 17 tiene conocimiento de que es necesario efectuar la reposición del valor del reloj de contador de tiempos de llegada 255 en una discontinuidad de ATC en el modelo de reproducción ilustrado en la Figura 7, puesto que la discontinuidad de ATC existe en un borde PlayItem. En el caso del objeto PlayItem actual representado en la Figura 61, a modo de ejemplo, el valor del reloj de contador de tiempos de llegada 255 es objeto de reposición al valor de la marca de tiempo de llegada arrival_time_stamp de un paquete indicado por el punto de inicio SPN_ATC_start de ATC_Sequence2.

25 En una etapa S857, la unidad de control 17 realiza una determinación en cuanto a si el procesado del último objeto PlayItem se ha realizado completamente, o no. Si el procesado del último objeto PlayItem no se ha completado, el flujo de procesado pasa a una etapa S858 en donde el parámetro k se incrementa en 1. Si el resultado de la determinación realizada en la etapa S857 indica que se ha completado el procesado del último objeto PlayItem, por el contrario, se finaliza el procesado para reproducir PlayList.

La Figura 67 es un diagrama de flujo que representa un método para reproducir mapa_TU-type PlayList.

35 Según se ilustra en la Figura, el diagrama de flujo comienza con una etapa S871 en donde la unidad de control 17, utilizada en el aparato de grabación/reproducción de imágenes en movimiento 1, ilustrado en la Figura 43, efectúa la lectura de datos de un archivo PlayList.

A continuación, en la siguiente etapa S872, la unidad de control 17 establece un parámetro K a 0 ($K = 0$). El parámetro K es el número de datos de PlayItem incluidos en PlayList como una entrada.

40 Posteriormente, en la siguiente etapa S873, la unidad de control 17 adquiere connection_condition = 1 del k-ésimo PlayItem a reproducirse actualmente.

45 A continuación, en la siguiente etapa S874, la unidad de control 17 tiene conocimiento de que es necesario efectuar la reposición del contador de ATC de la unidad de reproducción después de la lectura de datos de AV del (K-1)-ésimo objeto PlayItem pero antes de iniciar la lectura de datos de AV del k-ésimo objeto PlayItem. Con más detalle, la unidad de control 17 tiene conocimiento de que es necesario efectuar la reposición del valor de contador de tiempos de llegada 255 en una discontinuidad de ATC en el modelo de reproducción ilustrado en la Figura 7, puesto que la discontinuidad de ATC existe en una frontera operativa de PlayItem. (En el caso del objeto PlayItem actual ilustrado en la Figura 64, a modo de ejemplo, el valor del reloj de contador de tiempos de llegada 255 se repone al valor de la marca de tiempo de llegada arrival_time_stamp de un paquete indicado por el punto de inicio SPN_ATC_start de ATC_Sequence2).

50 Posteriormente, en la siguiente etapa S875, la unidad de control 17 realiza una determinación en cuanto a si el procesado del último objeto PlayItem ha sido completado o no. Si no se ha completado el procesado del último objeto PlayItem, el flujo de procesado pasa a una etapa S876 en donde el parámetro k se incrementa en 1. Si el resultado de la determinación realizada en la etapa S875 indica que se ha completado el procesado del último objeto PlayItem, por el contrario, se finaliza el procesado para reproducir PlayList.

60 Según se describió anteriormente, sobre la base de la sintaxis, las estructuras de datos y sus reglas, es posible gestionar adecuadamente datos e información de reproducción, que se memorizan en el medio de grabación 10. Además, al usuario le está también permitido verificar adecuadamente datos memorizados en el medio de grabación 10 y reproducir los datos deseados a partir del medio de grabación 10 en una operación de reproducción.

65 El medio de grabación 10, para memorizar varias clases de información según se describió anteriormente, se suele realizar mediante un disco óptico. En el medio de grabación 10, se constituyen pistas de grabación en forma de

espiral o en forma concéntrica. Según se indica en la Figura 68, en un área de archivos agrupados 10A, los archivos agrupados y los archivos puestos bajo el directorio CLIPINF, ilustrado en la Figura 2, son objeto de memorización. Los archivos agrupados son archivos puestos bajo el directorio PLAYLIST, ilustrado en la Figura 2. Los archivos agrupados, a modo de ejemplo, son archivos que tienen un nombre de archivo *.rlps o *.vpls o que tienen la extensión rlps o vpls. A modo de ejemplo de los archivos puestos bajo el directorio CLIPINF son los archivos que tienen un nombre de archivo *.clpi o que tienen la extensión clpi. En otra área de grabación 10B, los archivos puestos bajo el directorio STREAM, ilustrado en la Figura 2, son memorizados. A modo de ejemplo de los archivos puestos bajo el directorio STREAM se indican los archivos que tienen un nombre de archivo *.m2ts o que tienen la extensión m2ts. Los archivos agrupados son archivos que necesitan ser objeto de lectura desde el medio de grabación 10 en un corto periodo de tiempo cuando el medio de grabación 10 está montado en el aparato de grabación/reproducción de imagen en movimiento 1.

Conviene señalar que, aunque las formas de realización anteriormente descritas adoptan un flujo continuo de transporte MPEG-2 como un flujo multiplexado típico, el alcance de la presente invención no está limitado a dicho flujo. A modo de ejemplo, la presente invención se puede aplicar a un flujo continuo de transporte DSS y a un flujo de programa MPEG-2.

Según se describió anteriormente, en conformidad con la presente invención, en un sistema para grabar paquete de un flujo continuo AV, tal como un flujo continuo de transporte en un medio de grabación, una marca de tiempo de llegada (arrival_time_stamp) que indica el tiempo de llegada de cada uno de los paquetes se graba también en el medio de grabación junto con los paquetes. Además, la información de una secuencia del ATC se graba también en el medio de grabación. La información representa la contigüidad de las marcas de tiempo de llegada. De forma más concreta, un conjunto matricial de paquetes grabados incluye una dirección de paquete grabado (SPN_ATC_start) en donde se inicia un eje de tiempos de la base de tiempos de llegada. La dirección se representa por el número de un paquete incluido en el conjunto matricial.

Cuando se graba recientemente un flujo continuo AV, a modo de ejemplo, el conjunto matricial de paquetes continuamente grabados no incluye una discontinuidad de la base de tiempos de llegada. Es decir, existe solamente un eje de tiempos de la base de tiempos de llegada. El eje de tiempos se inicia desde el primer paquete del conjunto matricial.

Es posible un caso en que un paquete de una parte innecesaria en el conjunto matricial de paquetes se elimine en un proceso de edición o similar y todos los paquetes restantes se pueden reagrupar en un nuevo conjunto matricial de paquetes. En este caso, el nuevo conjunto matricial de paquetes puede incluir una pluralidad de ejes de tiempos de la base de tiempos de llegada. En tal caso, la dirección de un paquete desde la que se inicia cada uno de los ejes de tiempos de la base de tiempos de llegada se graba también en el medio de grabación.

Además, en un sistema utilizado también para grabar información que representa la continuidad de las marcas de tiempo de llegada, la información que representa la continuidad de una base de tiempos del sistema, que se refiere a tiempos de reproducción de datos de AV, es también objeto de grabación. La información que representa la continuidad de la base de tiempos del sistema se refiere como una información de secuencia del STC, que incluye discontinuidades de la base de tiempos del sistema. Más concretamente, un conjunto matricial de paquetes grabados incluye una dirección de paquetes grabados (SPN_STC_start) en donde se inicia el eje de tiempos de la base de tiempos del sistema. La dirección se representa por el número de un paquete incluido en el conjunto matricial.

En un conjunto matricial de paquetes que no incluye ninguna discontinuidad de la base de tiempos del sistema, los datos anteriormente descritos se gestionan de tal manera que una secuencia del STC no se extienda sobre un entorno límite de una secuencia del ATC, que es un conjunto matricial de paquetes que no incluye ninguna discontinuidad de la base de tiempos de llegada.

De este modo, es posible gestionar correctamente la dirección de un paquete, en donde se inicia el eje de tiempos de la base de tiempos de llegada, en un solo conjunto matricial de paquetes grabados. En consecuencia, aún cuando aumente el número de secuencias ATC, no aumenta el número de archivos descrito. En consecuencia, los archivos se pueden gestionar con facilidad. Además, puesto que cada secuencia del STC se identifica por stc_id se puede editar PlayList también con facilidad.

Además, aún cuando el archivo de flujo continuo AV incluya una discontinuidad de la base de tiempos de llegada y/o una discontinuidad de la base de tiempos del sistema, se puede gestionar adecuadamente los tiempos de inicio y de final de la reproducción de datos de AV.

Conviene señalar que, aunque las formas de realización anteriormente descritas utilizan un flujo continuo de transporte MPEG-2, como un flujo multiplexado típico, el alcance de la presente invención no está limitado al flujo continuo de transporte MPEG-2. A modo de ejemplo, la presente invención se puede aplicar a un flujo continuo de transporte DSS y a un flujo de programas MPEG-2.

La serie de procesos anteriormente descritos puede realizarse por hardware o software. Si el procesado se realiza

por software, los programas que constituyen el software pueden instalarse en un ordenador incorporado en hardware especial o un ordenador de uso general desde una red o un medio de grabación. Instalando una diversidad de programas en un ordenador de uso general, ordenador puede activarse para realizar varias clases de procesado.

5 Según se ilustra en la Figura 43, el medio de grabación está distribuido por separado del aparato de grabación/reproducción de imágenes en movimiento como un medio para presentar programas memorizados en el medio de grabación al usuario. El medio de grabación suele ser medios de paquetes tales como el disco magnético 51 que incluye un disco flexible, el disco óptico 52 que incluye un CD-ROM (memoria de lectura solamente de disco compacto) y un DVD (disco versátil digital), un disco magneto-óptico 53 que incluye un MD (minidisco) y la memoria de semiconductores 54. Como alternativa, en lugar de presentar el programa al usuario a través de dichos medios de paquetes, los programas se pueden presentar también al usuario memorizándose por anticipado en una memoria ROM y en un disco duro, que se preinstalan en el aparato de grabación/reproducción de imágenes en movimiento.

15 Conviene señalar que, en esta especificación, las etapas que prescriben un programa memorizado en un medio de grabación pueden, por supuesto, ser elementos de procesado, que se realizan en el orden prescrito, a lo largo de la serie temporal, pero no tienen que ejecutarse de forma secuencial. Es decir, las etapas pueden ser elementos de procesado que se realizan de forma simultánea o individual.

20 Además, el término "sistema" utilizado en esta especificación significa el sistema completo que incluye una pluralidad de aparatos.

APLICABILIDAD INDUSTRIAL

25 Según se describió anteriormente, en conformidad con la presente invención, aún cuando se realice una operación de edición después de un proceso de grabación, la información de reproducción y datos se puede gestionar con facilidad así como adecuadamente.

30 Además, en conformidad con la presente invención, es posible poner en práctica un medio de grabación de información que permita que los datos memorizados y la reproducción de los datos se gestionen con facilidad y de forma adecuada, aún cuando se realice una operación de edición.

Además, en conformidad con la presente invención, un flujo continuo de datos se puede reproducir de forma continua y rápida sin perder ninguna parte deseada del flujo.

35 Además, en conformidad con la presente invención, aún cuando se elimine una parte de un flujo continuo de datos, no solamente puede reproducirse continuamente un flujo continuo de datos, sino que también se pueden gestionar con facilidad los datos después de la edición.

REIVINDICACIONES

5 **1.** Un aparato de reproducción de datos (1) para reproducir un flujo continuo de datos que incluye un conjunto matricial de paquetes procedente de un medio de grabación de información (10) para memorizar dicho flujo continuo de datos, que comprende:

10 un medio de reproducción (11) para reproducir la información que indica si existe, o no, una discontinuidad de la secuencia de reloj de tiempos de llegada (ATC) entre un primer elemento de reproducción que comprende un primer conjunto matricial de paquetes que no incluye ninguna discontinuidad de secuencia de reloj de tiempos del sistema (STC) y un segundo elemento de reproducción que comprende un segundo conjunto matricial de paquetes que sigue a dicho primer conjunto matricial de paquetes, que está separado de dicho primer conjunto matricial de paquetes por una discontinuidad de secuencia del STC y no incluye ninguna discontinuidad de secuencia del STC y

15 un medio de control (17) para controlar la reproducción de dicho flujo continuo de datos desde dicho medio de grabación de información sobre la base de dicha información reproducida, en donde

20 la información reproducida comprende una condición de conexión establecida en un primer valor, cuando existe la discontinuidad de secuencia del ATC y establecida en un segundo valor, diferente al primer valor, cuando no exista la discontinuidad de secuencia del ATC y

25 el medio de control es utilizable para realizar una determinación en cuanto a si el segundo elemento de reproducción es el último elemento de reproducción de una lista de reproducción y para finalizar la reproducción del flujo continuo de datos cuando el segundo elemento de reproducción es el último elemento de reproducción y para incrementar un número de datos de elementos de reproducción en 1 cuando el segundo elemento de reproducción no es el último elemento de reproducción, poniéndose a cero el número de datos de elementos de reproducción al principio de la reproducción del flujo continuo de datos.

30 **2.** Un método de reproducción de datos, adoptado por un aparato de reproducción de datos para reproducir un flujo continuo de datos que incluye un conjunto matricial de paquetes desde un medio de grabación de información utilizado para memorizar dicho flujo continuo de datos, que comprende las etapas de:

35 reproducir información (S2853) que indica si existe, o no, una secuencia de reloj de tiempo de llegada (ATC) entre un primer elemento de reproducción que comprende un primer conjunto matricial de paquetes que no incluye ninguna discontinuidad de reloj de tiempos de sistema (STC) y un segundo elemento de reproducción que comprende un segundo conjunto matricial de paquetes que sigue a dicho primer conjunto matricial de paquetes, que está separado de dicho primer conjunto matricial de paquetes por una discontinuidad de la secuencia del STC y no incluye ninguna discontinuidad de secuencia del STC y

40 controlar la reproducción (S854) de dicho flujo continuo de datos desde dicho medio de grabación de información sobre la base de dicha información reproducida, en donde

45 la información reproducida comprende una condición de conexión establecida en un primer valor cuando existe la discontinuidad de ATC y para establecer un segundo valor, diferente al primer valor, cuando no exista la discontinuidad de secuencia del ATC y

50 la etapa de reproducción controladora comprende la realización de una determinación en cuanto a si el segundo elemento de reproducción es el último elemento de reproducción de una lista de reproducción y finalizar la reproducción del flujo continuo de datos cuando el segundo elemento de reproducción es el último elemento de reproducción e incrementando un número de datos de elementos de reproducción en 1 cuando el segundo elemento de reproducción no es el último elemento de reproducción, poniéndose a cero el número de datos de elementos de reproducción al principio de la reproducción del flujo continuo de datos.

55 **3.** Un programa a ejecutarse por un ordenador para controlar un aparato de reproducción de datos para reproducir un flujo continuo de datos que incluye un conjunto matricial de paquetes desde un medio de grabación de información utilizado para memorizar dicho flujo continuo de datos para realizar el método según la reivindicación 2.

FIG. 1

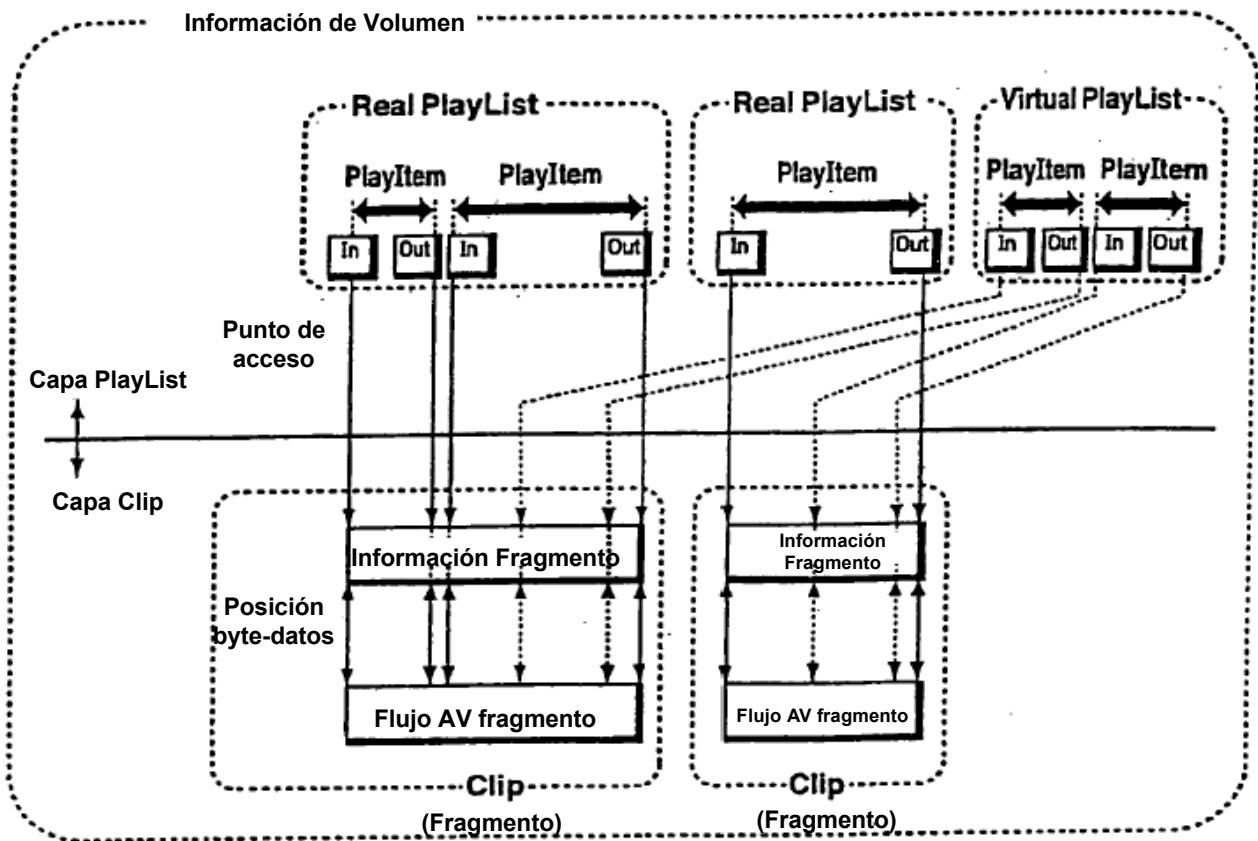


FIG. 2

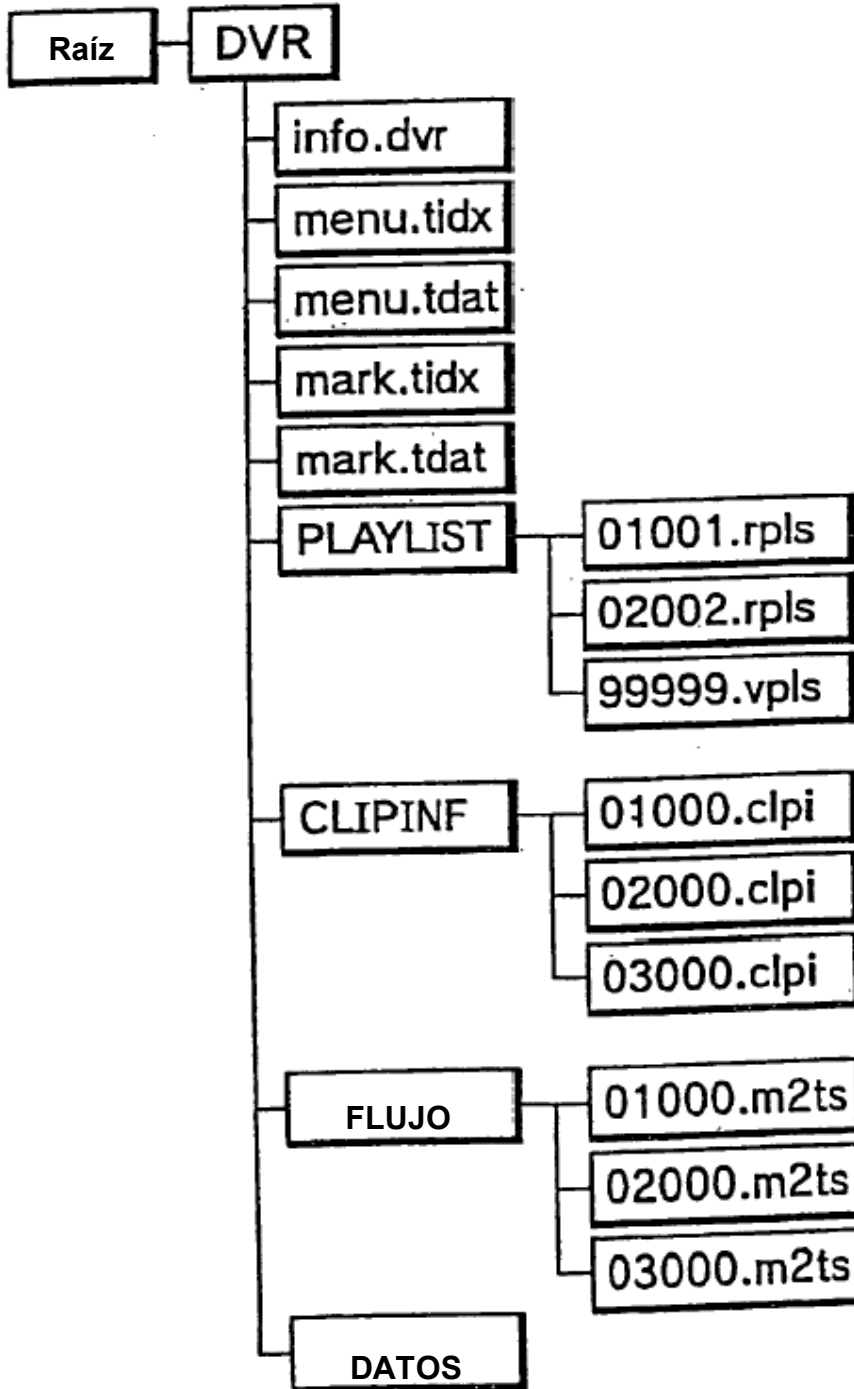


FIG. 3

Flujo continuo de transporte DVR MPEG-2

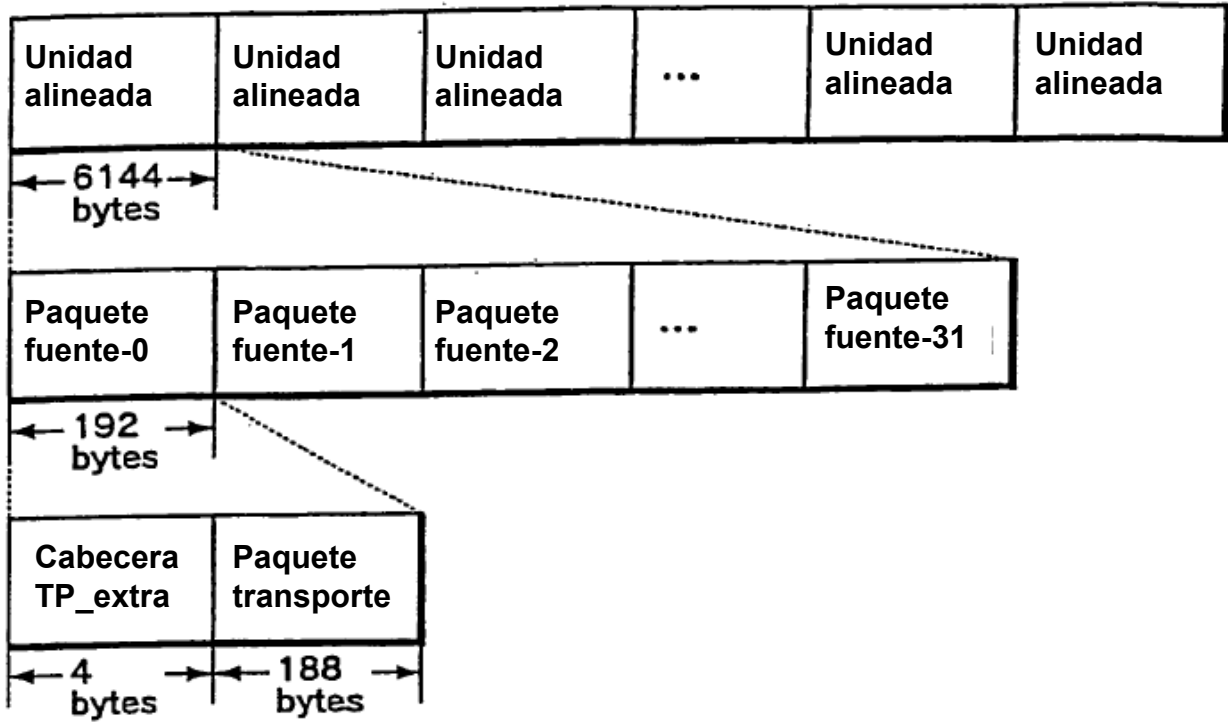


FIG. 4

Sintaxis	Nº de bits	Mnemónico
source packet() {		
TP_extra_header()		
transport_packet()		
}		

FIG. 5

Sintaxis	Nº de bits	Mnemónico
TP_extra_header() {		
copy_permission_Indicator	2	uimsbf
arrival_time_stamp	30	uimsbf
}		

FIG. 6

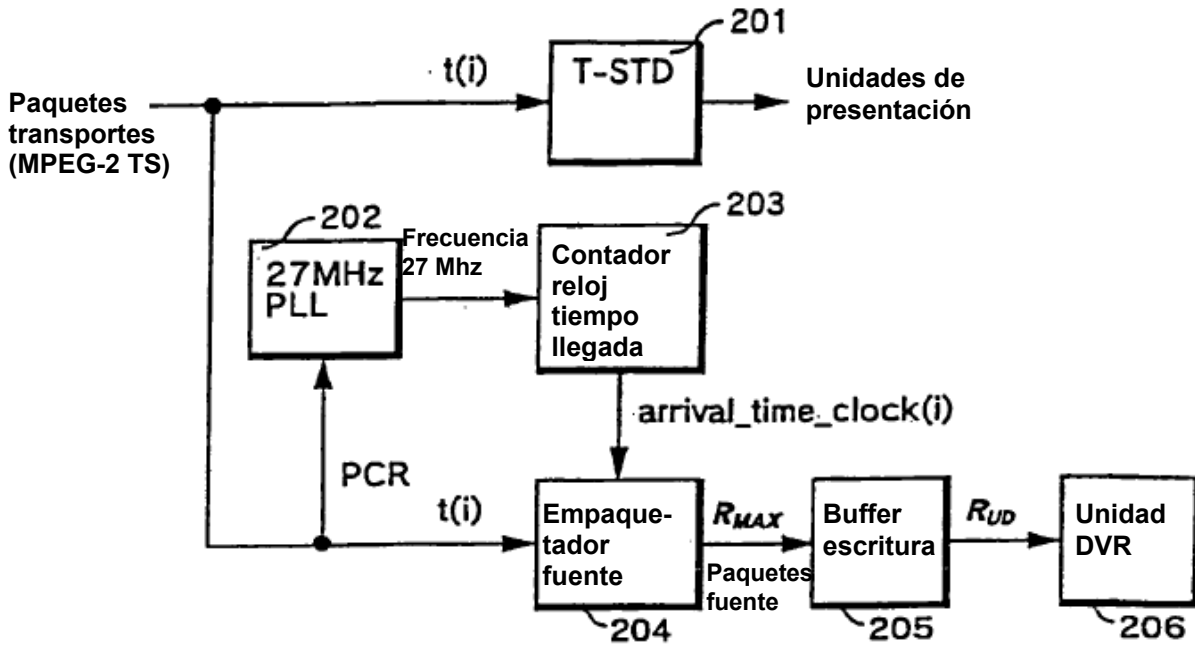
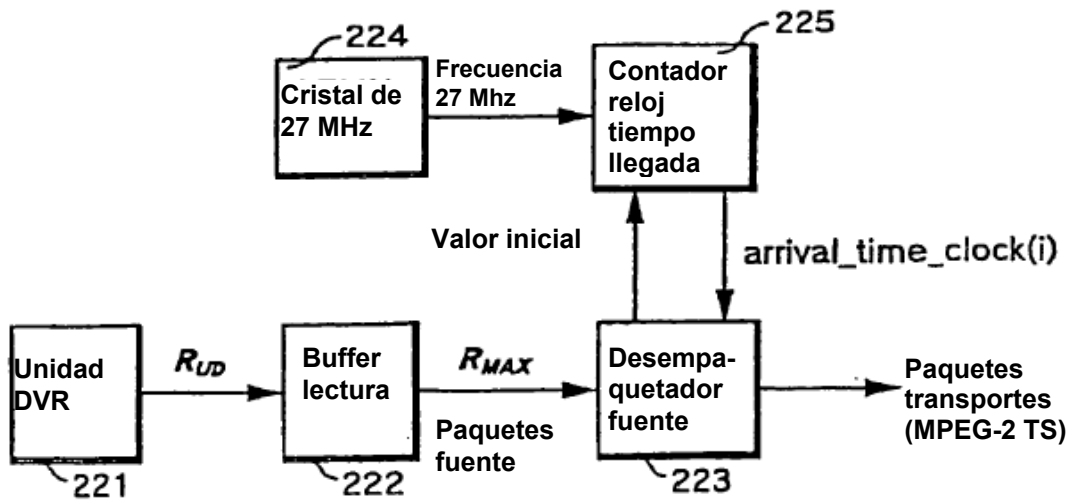


FIG. 7



F I G. 8

Sintaxis	Nº de bits	Mnemónico
zzzzz.cipi {		
version_number	8*4	bslbf
SequenceInfo_start_address	32	uimsbf
ProgramInfo_start_address	32	uimsbf
CPI_start_address	32	uimsbf
ClipMark_start_address	32	uimsbf
MarkersPrivateData_start_address	32	uimsbf
reserved_for_future_use	96	bslbf
ClipInfo()		
for(i=0; i<N1; i++){		
Padding_word	16	bslbf
}		
SequenceInfo()		
for(i=0; i<N2; i++){		
Padding_word	16	bslbf
}		
ProgramInfo()		
for(i=0; i<N3; i++){		
Padding_word	16	bslbf
}		
CPI()		
for(i=0; i<N4; i++){		
Padding_word	16	bslbf
}		
ClipMark()		
for(i=0; i<N5; i++){		
Padding_word	16	bslbf
}		
MarkersPrivateData()		
for(i=0; i<N6; i++){		
Padding_word	16	bslbf
}		
}		
}		

FIG. 9

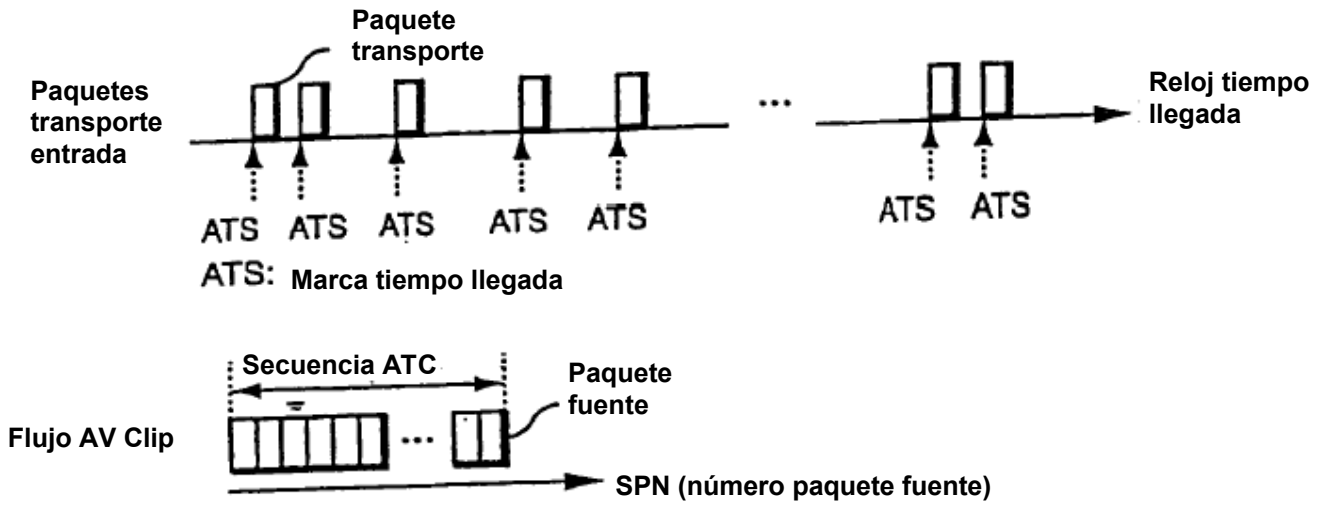


FIG. 10

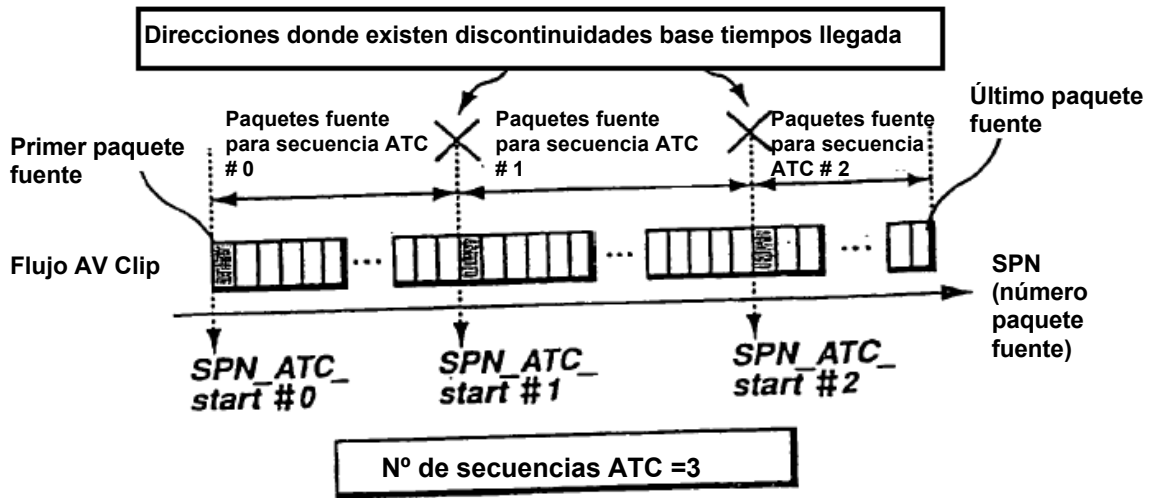
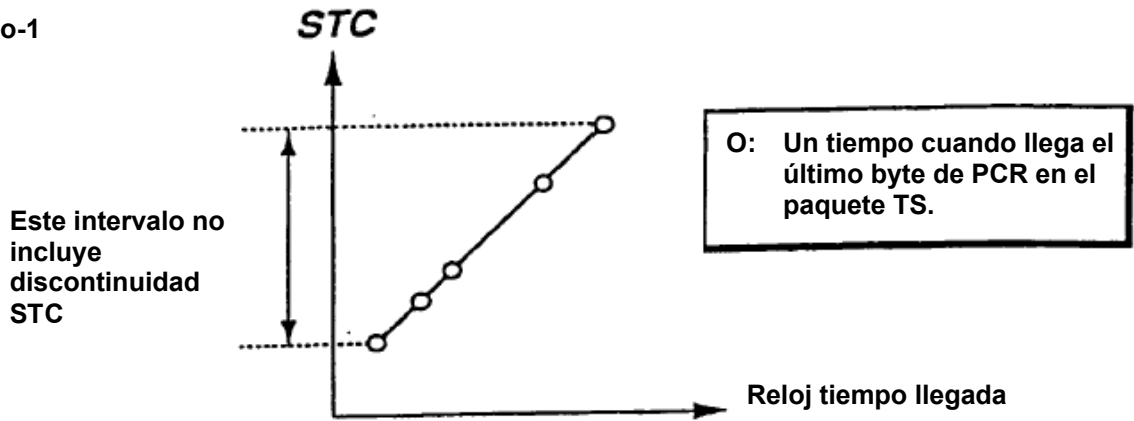


FIG. 11

Caso-1



Caso-2

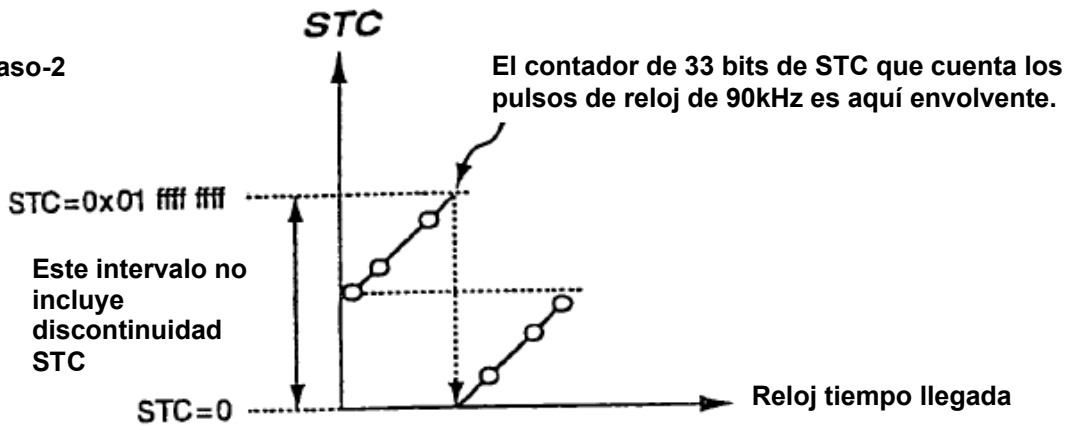


FIG. 12

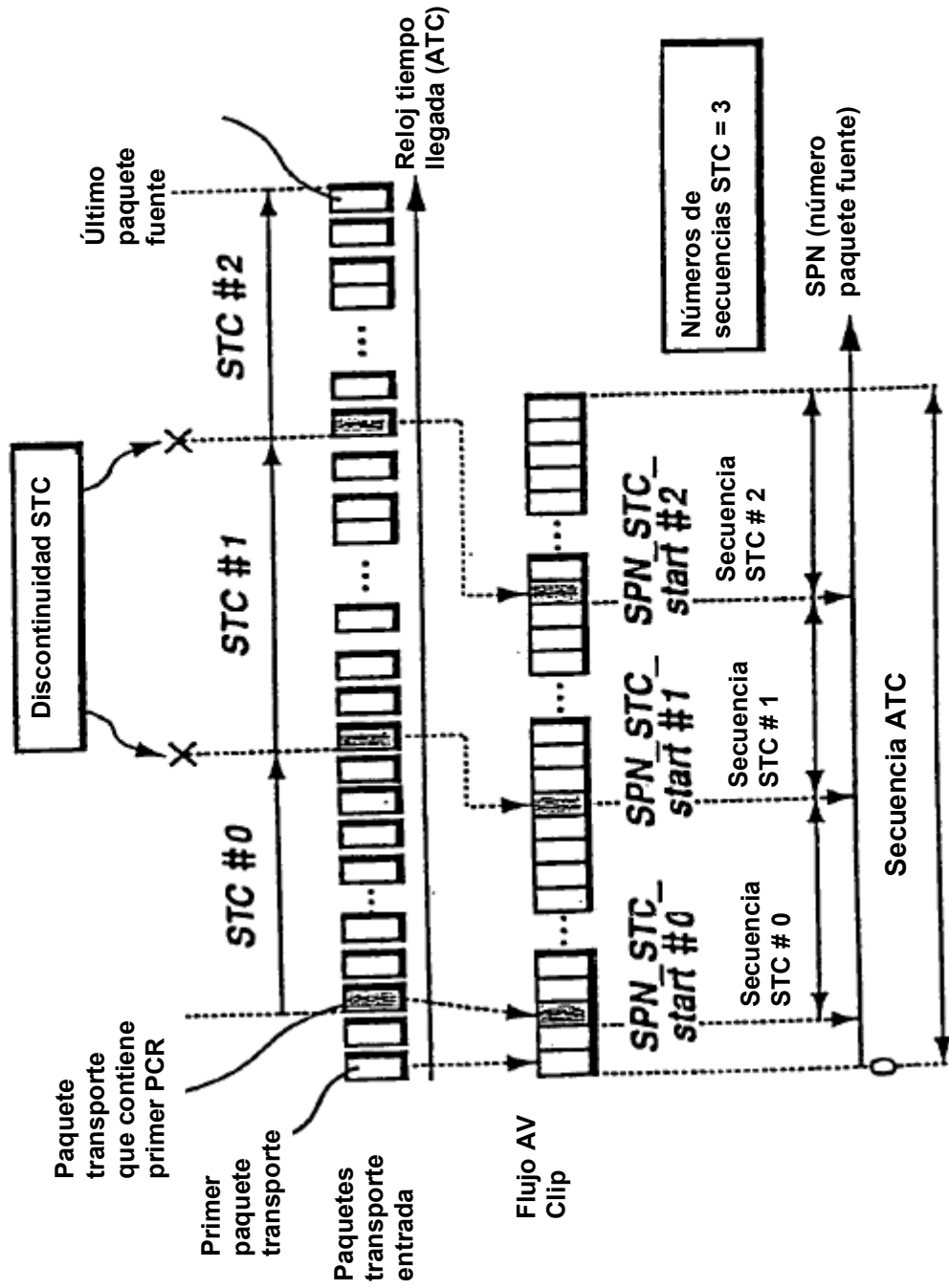


FIG. 13

Sintaxis	Nº de bits	Mnemónico
Sequenceinfo() {		
Longitud	32	uimsbf
reserved_for_word_align	8	bslbf
número de secuencias ATC	8	uimsbf
for(atc_id=0; atc_id<num_of_ATC_sequences; atc_id++){		
SPN_ATC_start[atc_id]	32	uimsbf
número de secuencias STC [atc_id]	8	uimsbf
offset_STC_id[atc_id]	8	uimsbf
for (stc_id = offset_STC_id[atc_id]; stc_id <(num_of_STC_sequences[atc_id]+offset_STC_id[atc_id]); stc_id++) {		
PCR_PID[atc_id][stc_id]	16	uimsbf
SPN_STC_start[atc_id][stc_id]	32	uimsbf
presentation_start_time[atc_id][stc_id]	32	uimsbf
presentation_end_time[atc_id][stc_id]	32	uimsbf
}		
}		
}		

FIG. 14

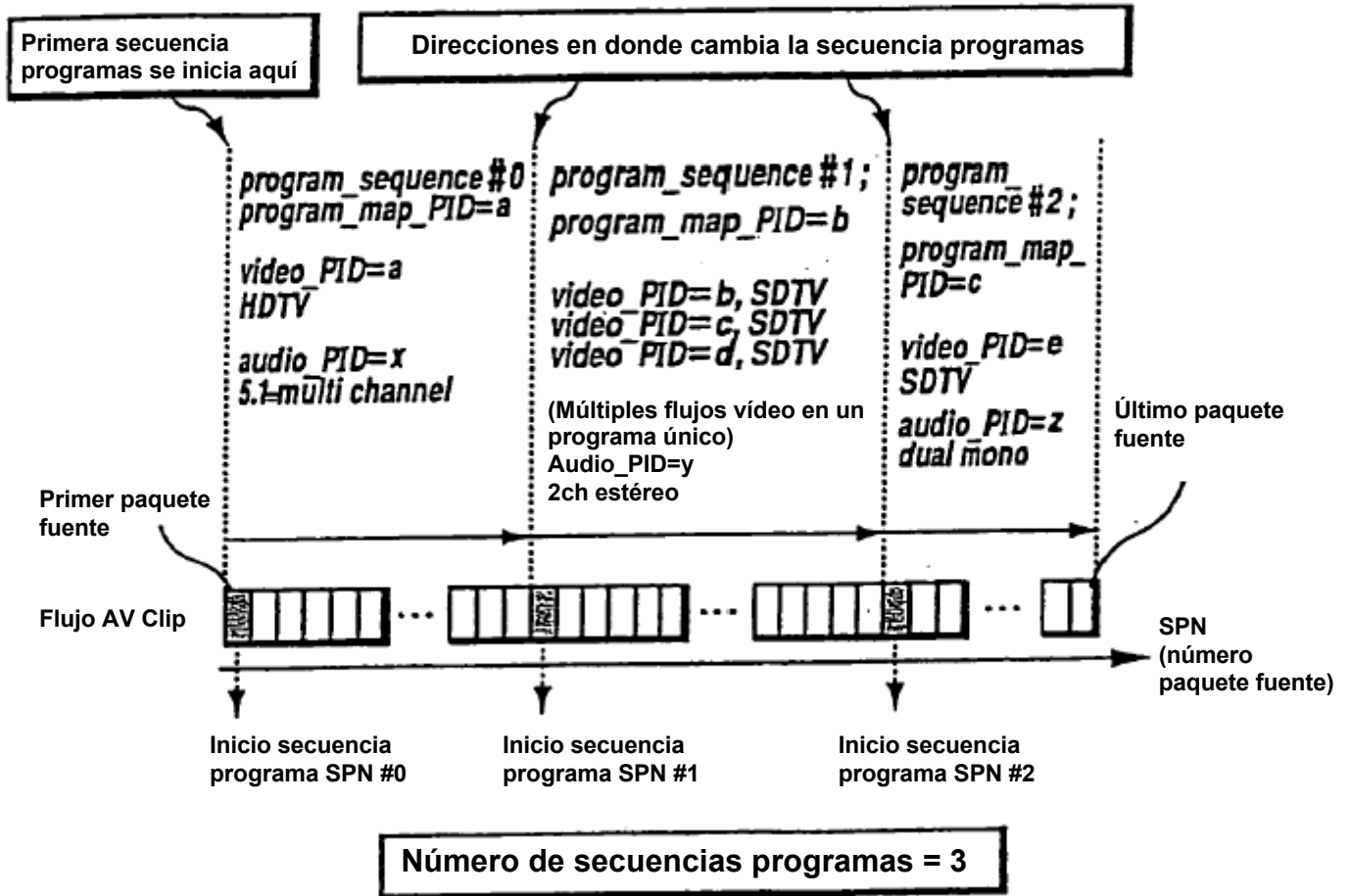


FIG. 15

Sintaxis	Nº de bits	Mnemónico
<code>programInfo() {</code>		
longitud	32	uimsbf
reserved_for_word_align	8	bslbf
número de secuencias de programas	8	uimsbf
for(i=0; i<num_of_program_sequences; i++){		
inicio secuencias programas SPN	32	uimsbf
PID mapa programas	16	bslbf
número de flujos en ps	8	uimsbf
número de grupos	8	uimsbf
for (stream_index=0; stream_index<num_of_streams_in_ps; stream_index++){		
PID flujo	16	uimsbf
Stream CodingInfo()		
}		
if (num_of_groups > 1){		
for(i=0; i<num_of_groups; i++){		
número de flujos en grupo	8	uimsbf
for (k=0; k<num_of_streams_in_group; k++){		
índice de flujos	8	uimsbf
}		
if (num_of_streams_in_group%2==0){		
reservado para alineación palabras	8	bslbf
}		
}		
}		
}		

FIG. 16

Sintaxis	Nº de bits	Mnemónico
StreamCodingInfo() {		
longitud	8	bslbf
tipo codificación flujo	8	uimsbf
if (<i>stream_coding_type</i> ==0x02) {		
formato vídeo	4	uimsbf
tasa tramas	4	uimsbf
relación aspecto visualización	4	uimsbf
reserved_for_word_align	2	bslbf
cc_flag	1	uimsbf
indicador formato vídeo original	1	
if (<i>original_video_format_flag</i> ==1) {		
formato vídeo original	4	uimsbf
relación aspecto visualización original	4	uimsbf
reserved_for_word_align	8	bslbf
}		
} else if (<i>stream_coding_type</i> ==0x03// <i>stream_coding_type</i> ==0x04// <i>stream_coding_type</i> ==0x0F// <i>stream_coding_type</i> ==0x80// <i>stream_coding_type</i> ==0x81) {		
tipo presentación audio	4	uimsbf
frecuencia de muestreo	4	uimsbf
reserved_for_word_align	8	bslbf
}		
}		

FIG. 17

Tipo codificación flujo	Significado
0x00 - 0x01	Reservado para uso futuro
0x02	Flujo video MPEG-1 o MPEG-2
0x03	Audio MPEG-1
004	Audio multicanal MPEG-2, retrocompatible para MPEG-1
0x05	Reservado para uso futuro
0x06	Teletexto definido en SESF o DVB o Subtítulo definido en ISDB
0x07 - 0x09	Reservado para uso futuro
0x0A	ISO/IEC 13818-6 tipo A
0x0 B	ISO/IEC 13818-6 tipo B
0x0 C	ISO/IEC 13818-6 tipo C
0x0 D	ISO/IEC 13818-6 tipo D
0x0 E	Reservado para uso futuro
0x0 F	Audio MPEG-2 AAC con sintaxis transporte ADTS
0x10-0x 7F	Reservado para uso futuro
0x80	Audio SESF LPCM
0x81	Audio Dolby AC-3
0x82-0x FF	Reservado para uso futuro

FIG. 18

Formato de video	Significado	Estándar de vídeo
0	480 i	ITU-R BT.601-4
1	576 i	ITU-R BT.601-4
2	480 p	SMPTE 293M
3	1080i	SMPTE 274M
4	720 p	SMPTE 296M
5-14	Reservado para uso futuro	
15	Sin información	

FIG. 19

Tasa de tramas	Significado
0	Reservado para uso futuro
1	24 000/1001 (23.976...)
2	24
3	25
4	30 000/1001 (29.97...)
5	30
6	50
7	60 000/1001 (59.94...)
8	60
9-14	Reservado para uso futuro
15	Sin información

F I G. 2 0

Relación aspecto visualización	Significado
0	Reservado para uso futuro
1	Reservado para uso futuro
2	Relación aspecto visualización 4:3
3	Relación aspecto visualización 16:9
4	Relación aspecto visualización 2.21:1
5-14	Reservado para uso futuro
15	Sin información

F I G. 2 1

Tipo presentación audio	Significado
0	Reservado para uso futuro
1	Monocanal único
2	Monocanal dual
3	estéreo (2 canales)
4	Multilingüe
5	Sonido envolvente
6	Multicanal
7-12	Reservado para uso futuro
13	Descripción de audio para el impedimento visual
14	Audio para el equipo de escucha
15	Sin información

FIG. 22

Frecuencia de muestreo	Significado
0	48 kHz
1	44,1 kHz
2	32 kHz
3-14	Reservado para uso futuro
15	Sin información

FIG. 23

Sintaxis	Nº de bits	Mnemónico
CPI() {		
Longitud	32	uimsbf
reserved_for_word_align	15	bslbf
Tipo de CPI	1	bslbf
if (CPI_type == 0) {		
EP_map()		
} else {		
TU_map()		
}		
}		

FIG. 24

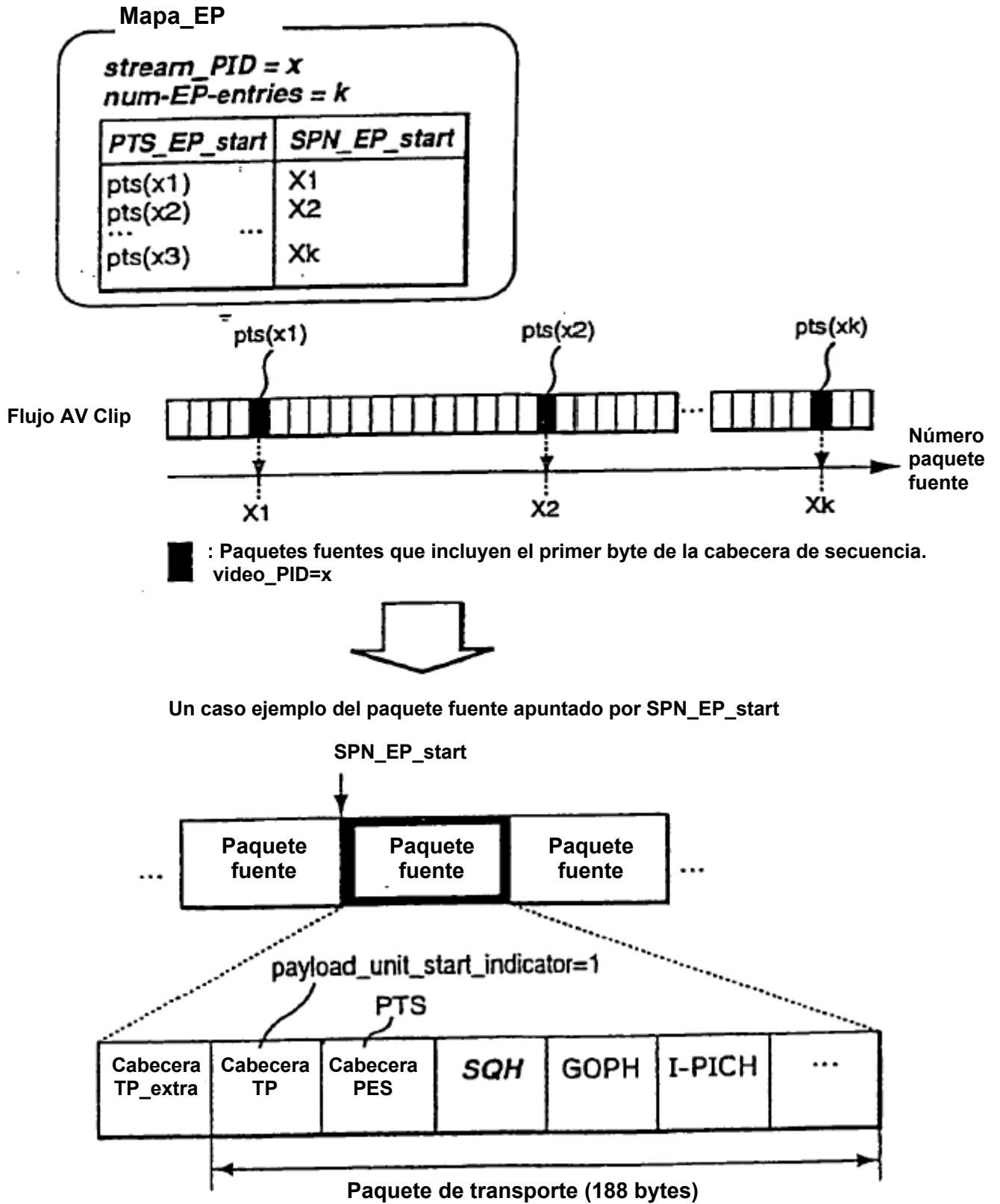


FIG. 25

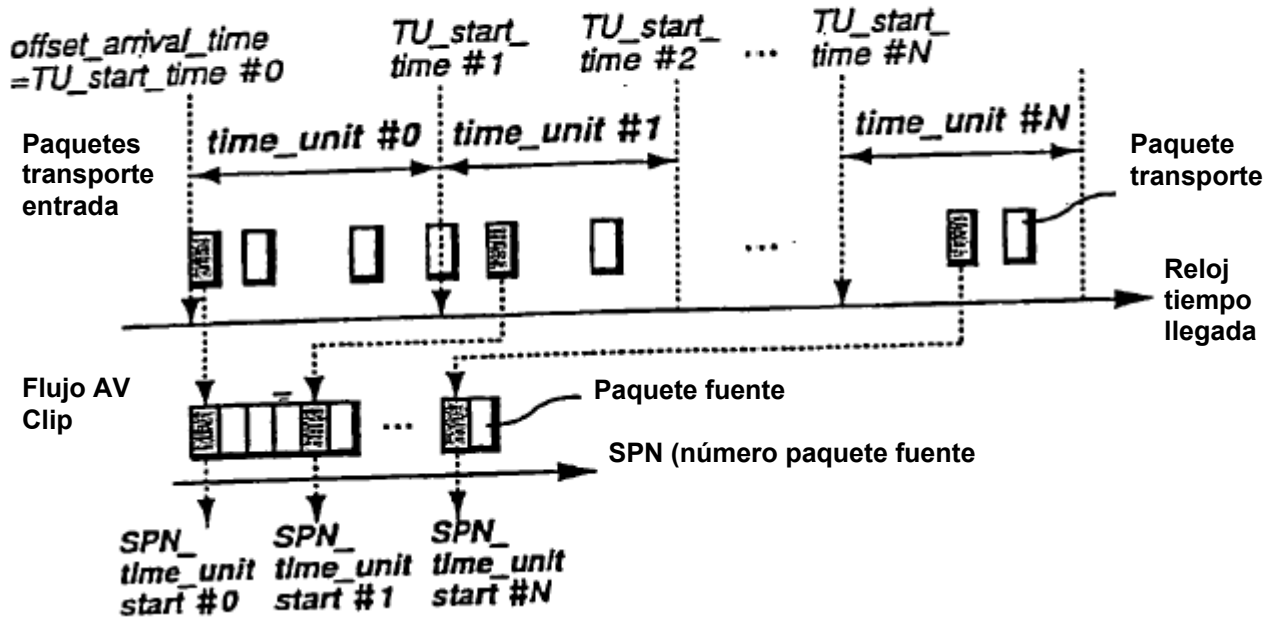


FIG. 26

Sintaxis	Nº de bits	Mnemónico
TU_map() {		
Tamaño unidad tiempo	32	uimsbf
for(atc_id=0; atc_id<num_of_ATC_sequences; atc_id++) {		
Tiempo llegada desplazamiento [atc_id]	32	bslbf
Número de entradas unidad tiempo [atc_id]	32	uimsbf
}		
for(atc_id=0; atc_id<num_of_ATC_sequences; atc_id++) {		
for(i=0; i<num_of_time_unit_entries[atc_id]; i++) {		
Inicio unidad tiempo SPN [atc_id][i]	32	uimsbf
}		
}		
}		

FIG. 27

Sintaxis	Nº de bits	Mnemónico
xxxx.rpls / yyyy.vpls {		
Número de versión	8*4	bslbf
Dirección inicio PlayList	32	uimsbf
Dirección inicio PlayListMark	32	uimsbf
Dirección inicio MakersPrivateData	32	uimsbf
Reservado para uso futuro	160	bslbf
UIAppInfoPlayList()		
for (i=0; i<N1; i++) {		
Palabra de relleno	16	bslbf
}		
PlayListst()		
for (i=0; i<N2; i++) {		
Palabra de relleno	16	bslbf
}		
PlayListstMark()		
for (i=0; i<N3; i++) {		
Palabra de relleno	16	bslbf
}		
MakersPrivateData()		
for (i=0; i<N4; i++) {		
Palabra de relleno	16	bslbf
}		
}		

FIG. 28

Sintaxis	Nº de bits	Mnemónico
PlayList() {		
Longitud	32	uimsbf
Reservado para alineación de palabra	15	bslbf
Tipo de CPI	1	bslbf
Número de PlayItems	16	uimsbf
if (<Virtual-PlayList> && CPI_type==0) {		
Número de SubPlayItems	16	uimsbf
} else {		
Reservado para alineación de palabra	16	bslbf
}		
for (PlayItem_id=0; PlayItem_id<number_of_PlayItems; PlayItem_id++) {		
PlayItem()		
}		
if (<Virtual-PlayList> && CPI_type==0) {		
for (i=0; i<number_of_ SubPlayItems; i++) {		
SubPlayItem()		
}		
}		
}		

FIG. 29

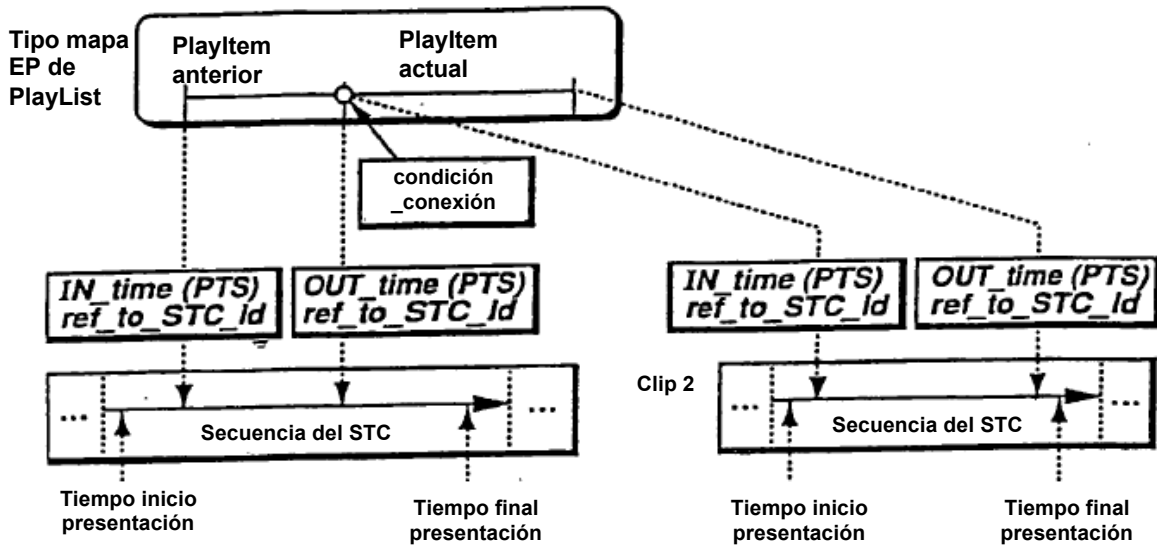


FIG. 30

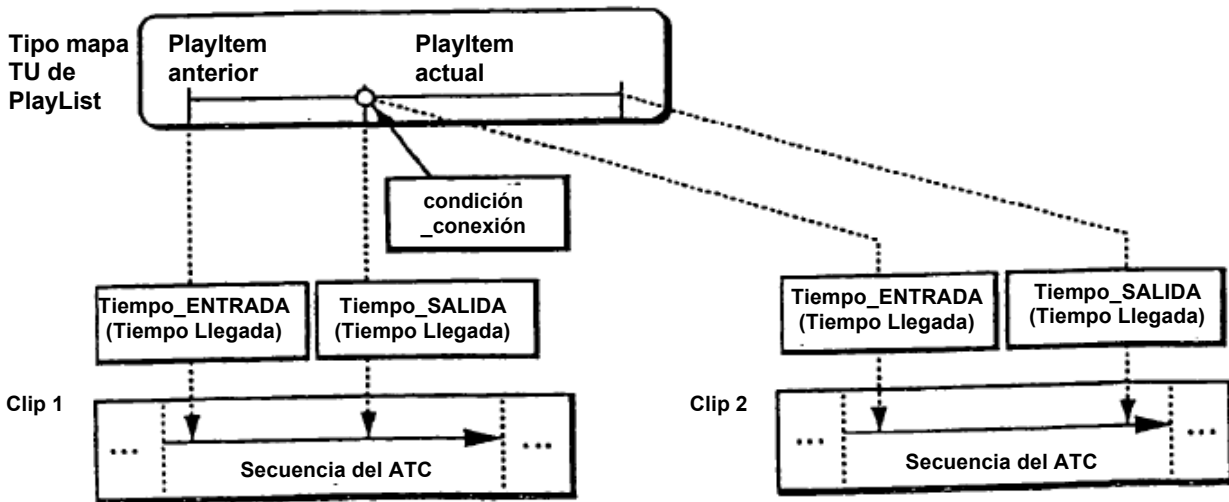


FIG. 31

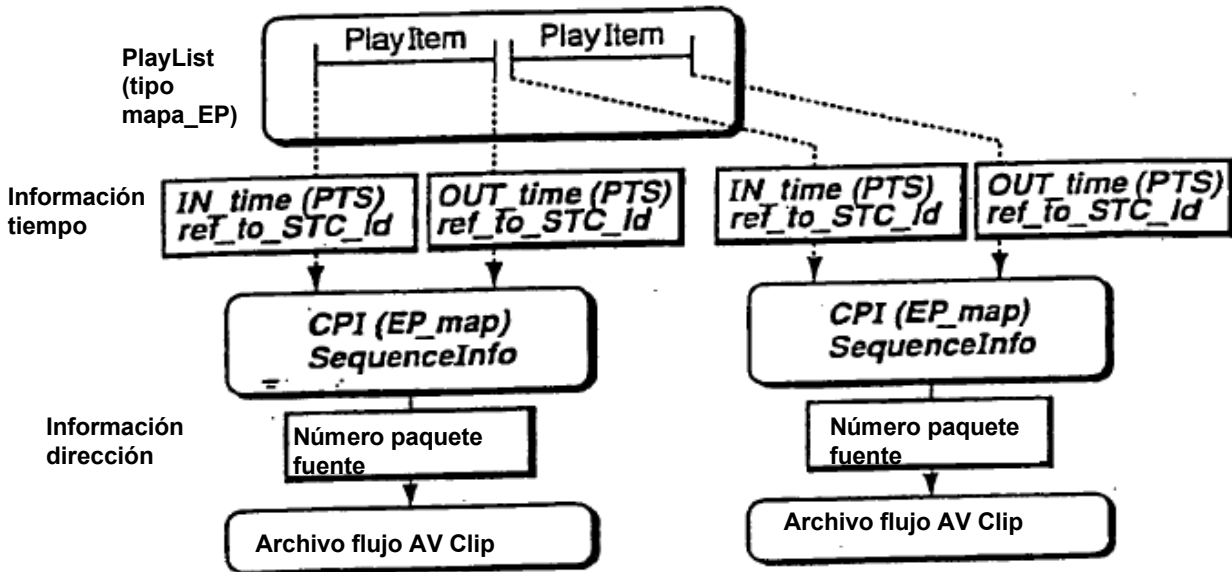


FIG. 32

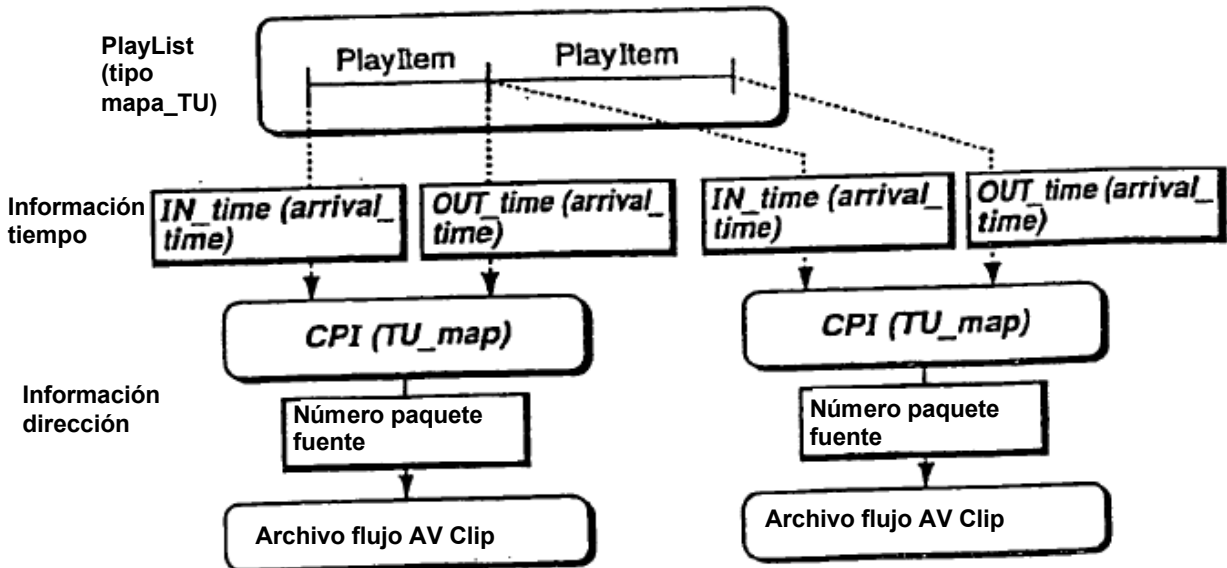


FIG. 33

Sintaxis	Nº de bits	Mnemónico
PlayItem() {		
longitud	16	uimsbf
nombre archivo información Clip	8*10	bslbf
reservado para alineación de palabra	6	bslbf
condición conexión	2	bslbf
si (tipo_CPI==0= { /* el tipo_CPI se define en la PlayList() */		
Ref_a_id_STC	8	uimsbf
} else {		
reservado para alineación de palabra	8	bslbf
}		
Tiempo_ENTRADA	32	uimsbf
Tiempo_SALIDA	32	uimsbf
if (<Virtual-PlayList> connection_condition=='10') {	&&	
Nombre_archivo_información_puente_Clip		
}		
}		

FIG. 34

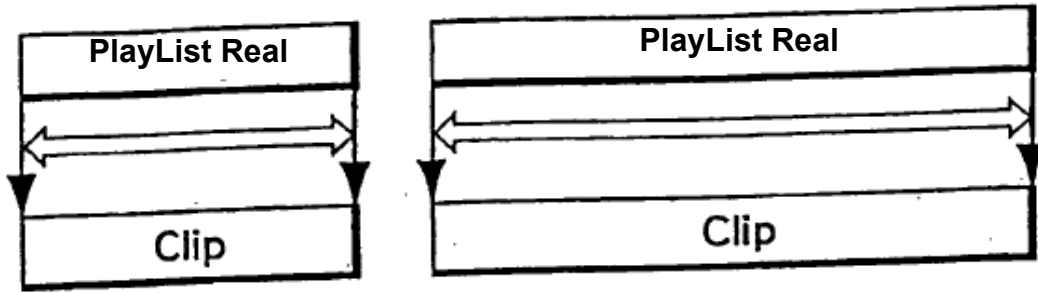


FIG. 35

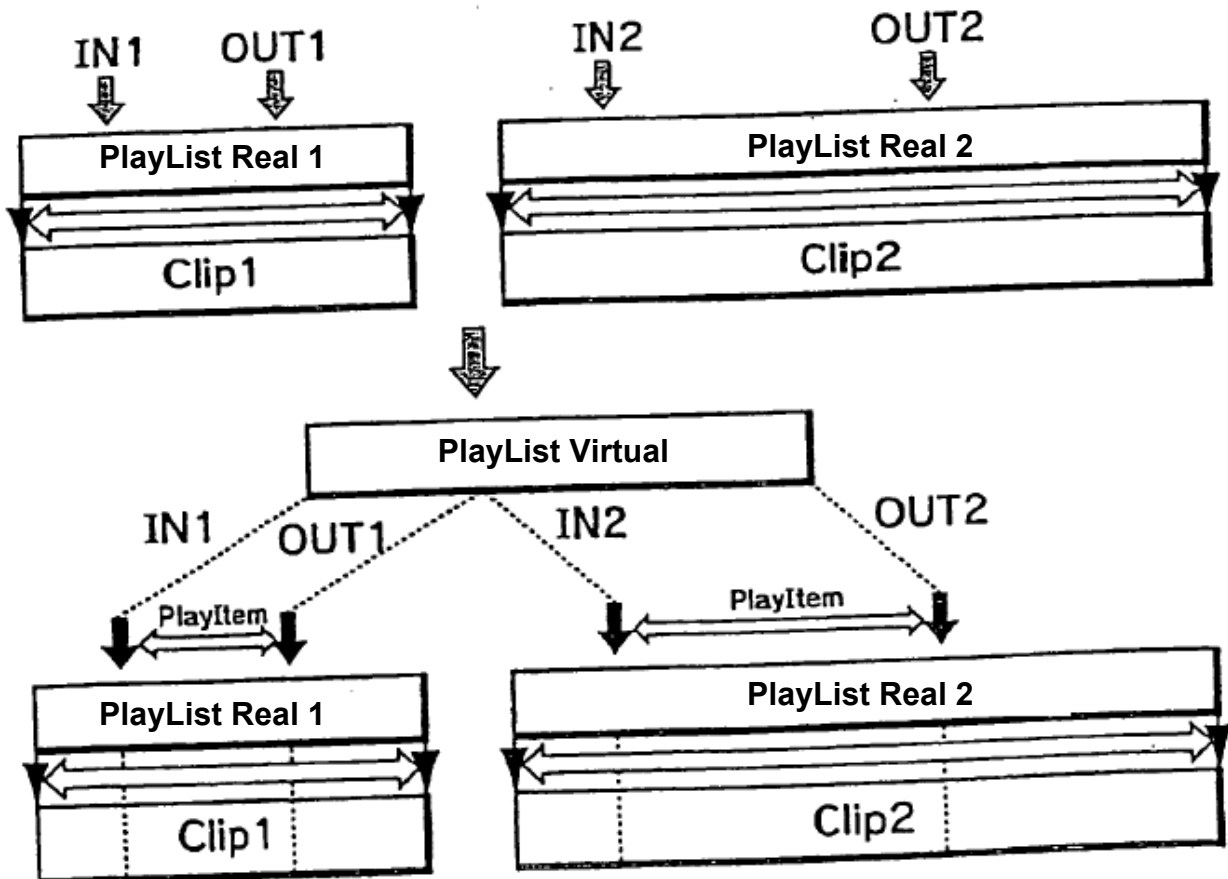


FIG. 36

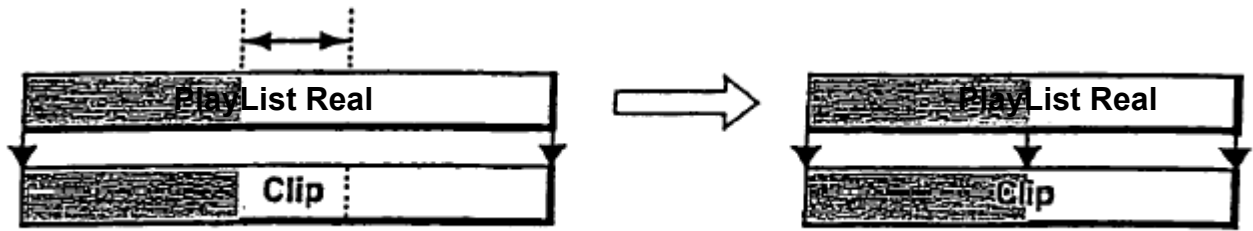
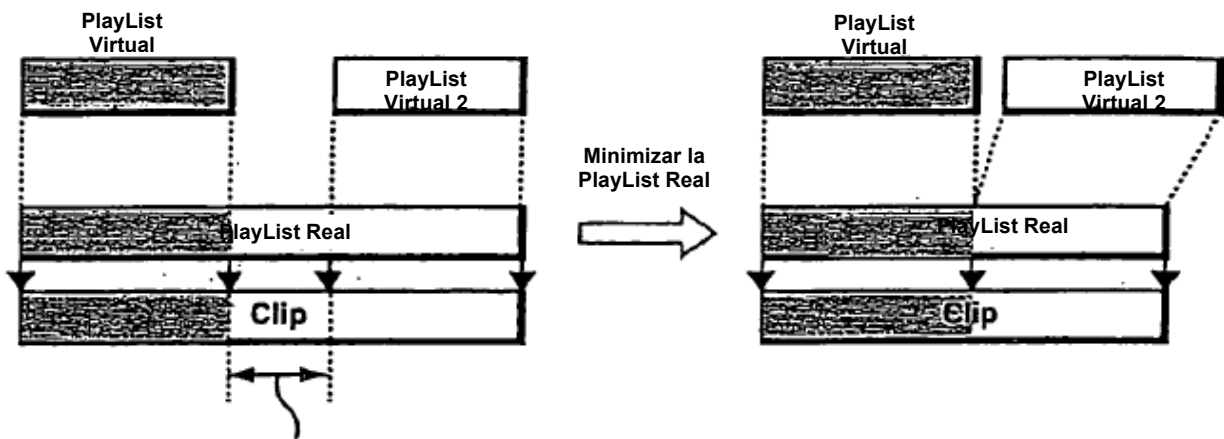


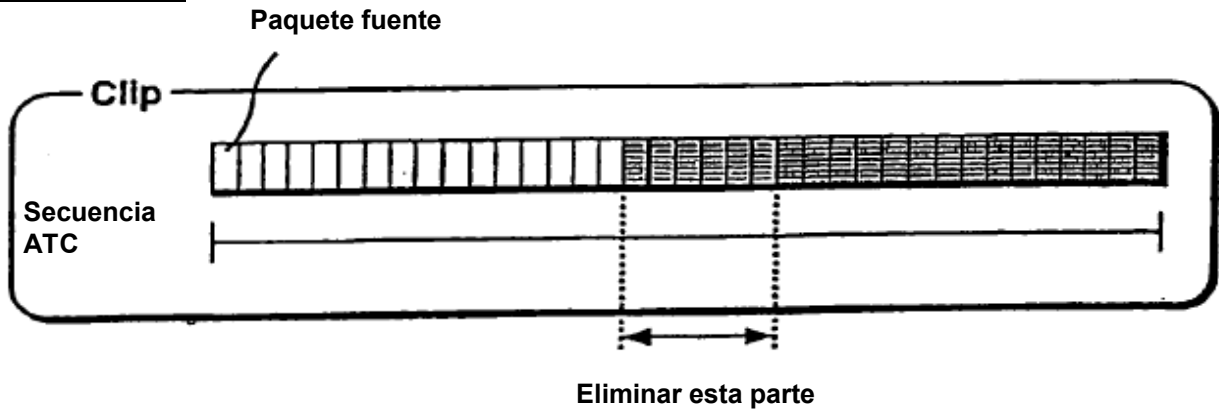
FIG. 37



Esta parte del Clip no se utiliza por cualquier PlayLists Virtual

FIG. 38

Antes de edición



Después de edición

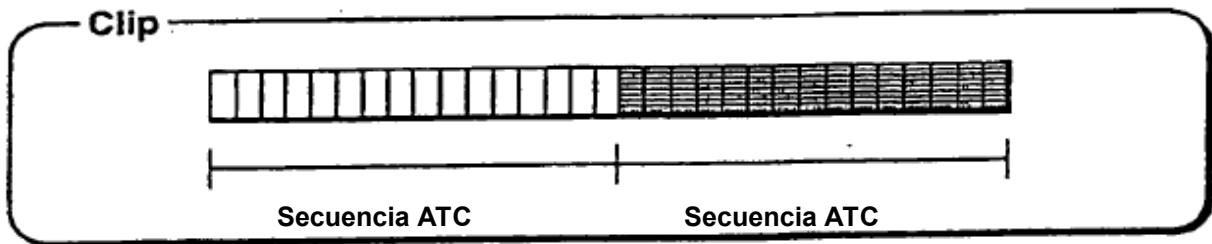
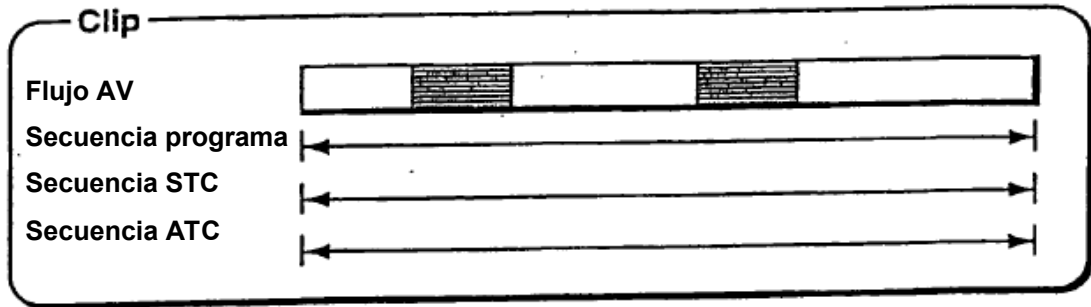


FIG. 39

Antes de edición



Eliminación partes sombreadas del flujo continuo AV

Después de edición

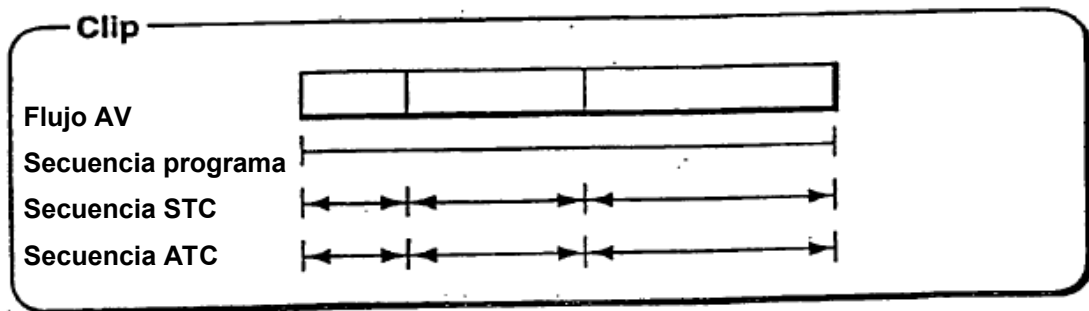
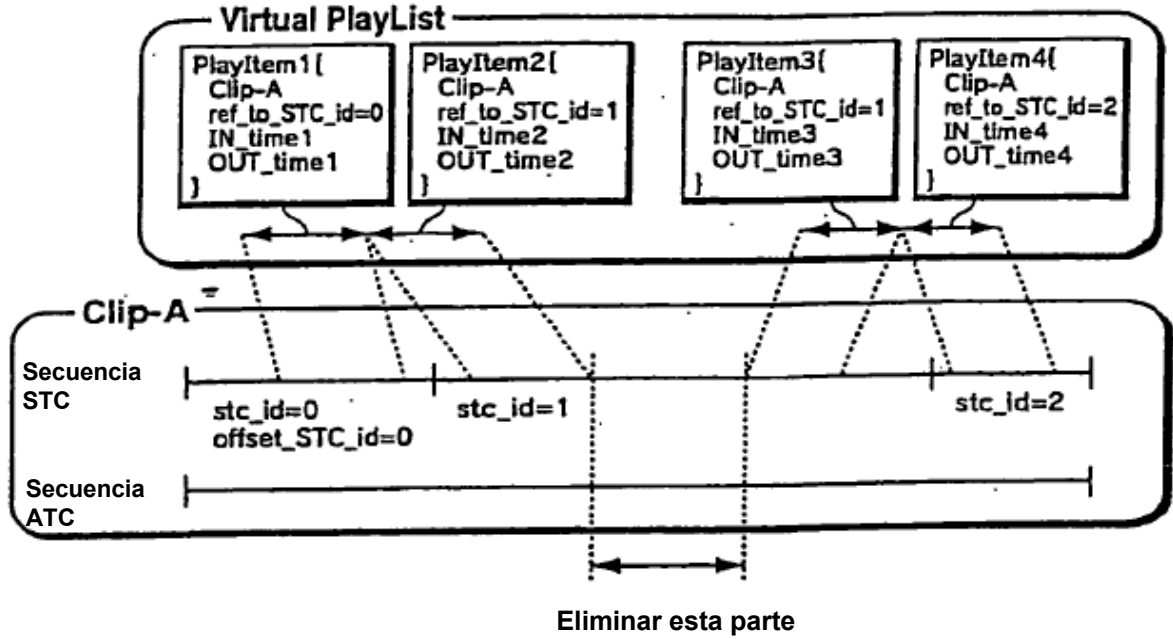


FIG. 40

Antes de edición



Después de edición: PlayItem3 y PlayItem 4 no cambian

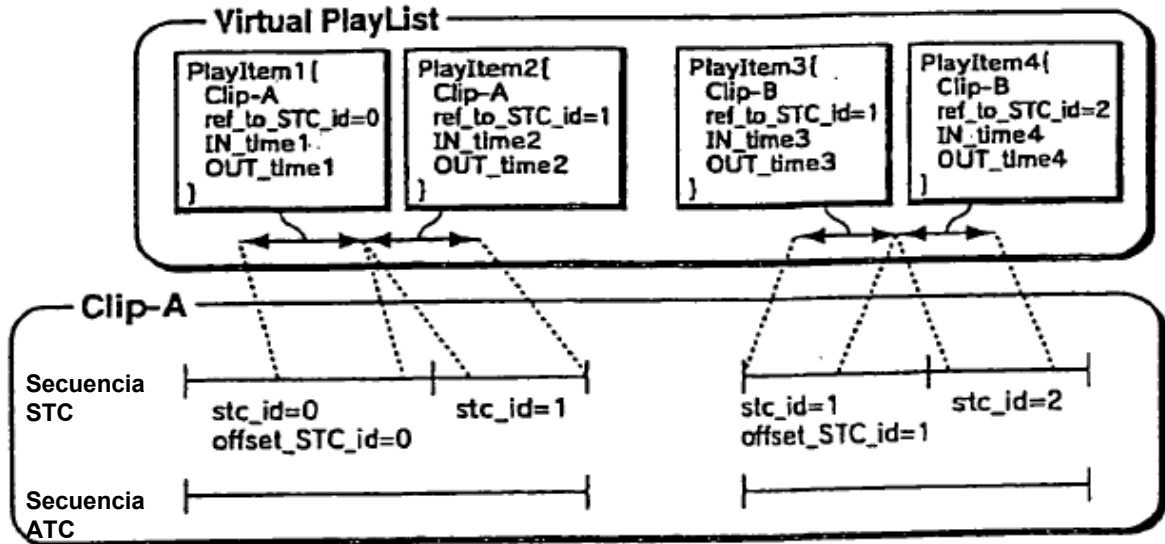
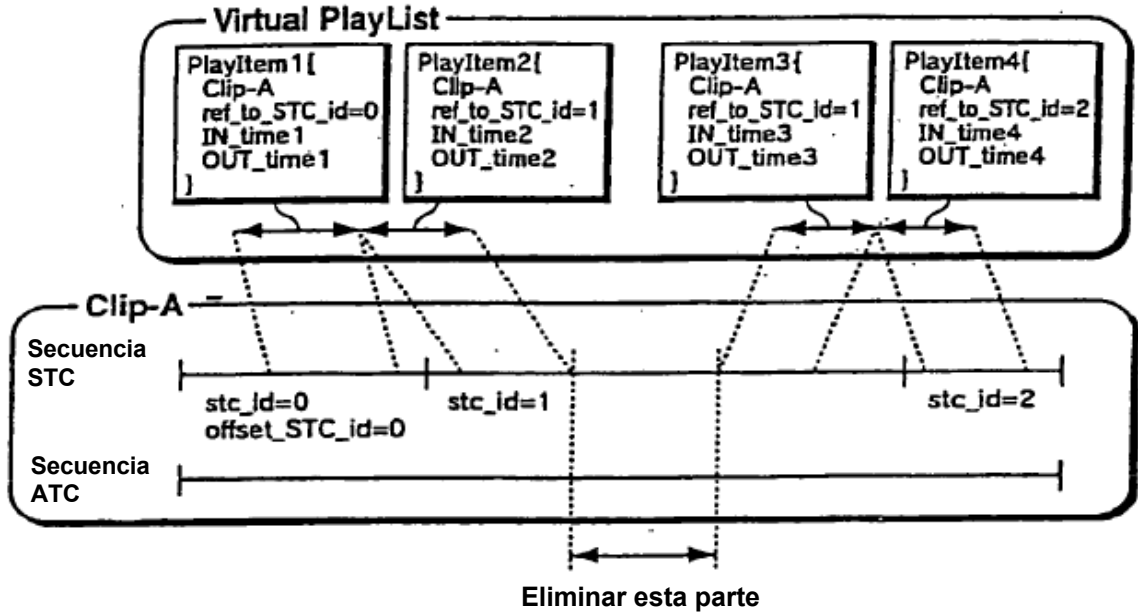


FIG. 41

Antes de edición



Después de edición: PlayItem3 y PlayItem 4 cambian

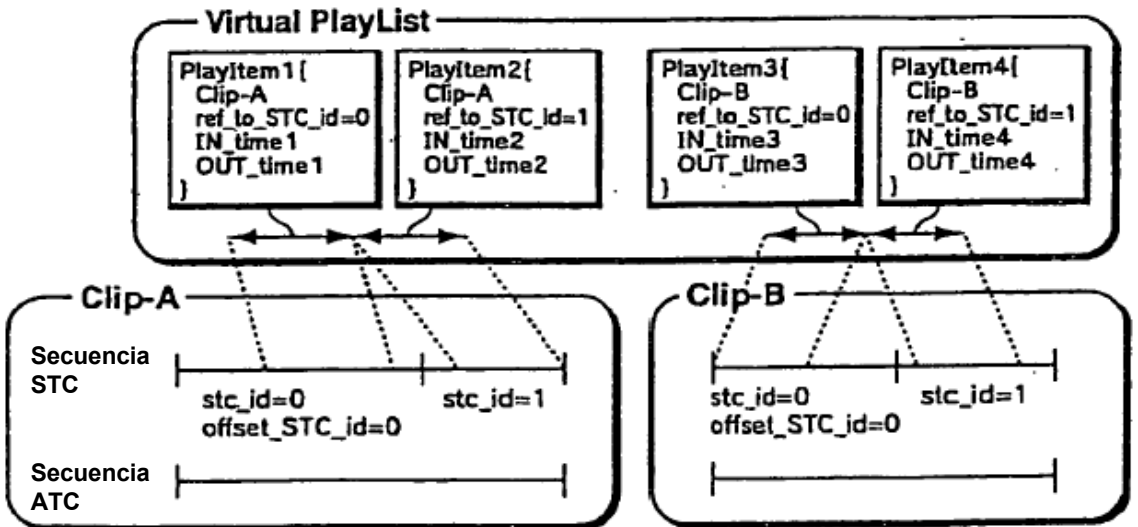
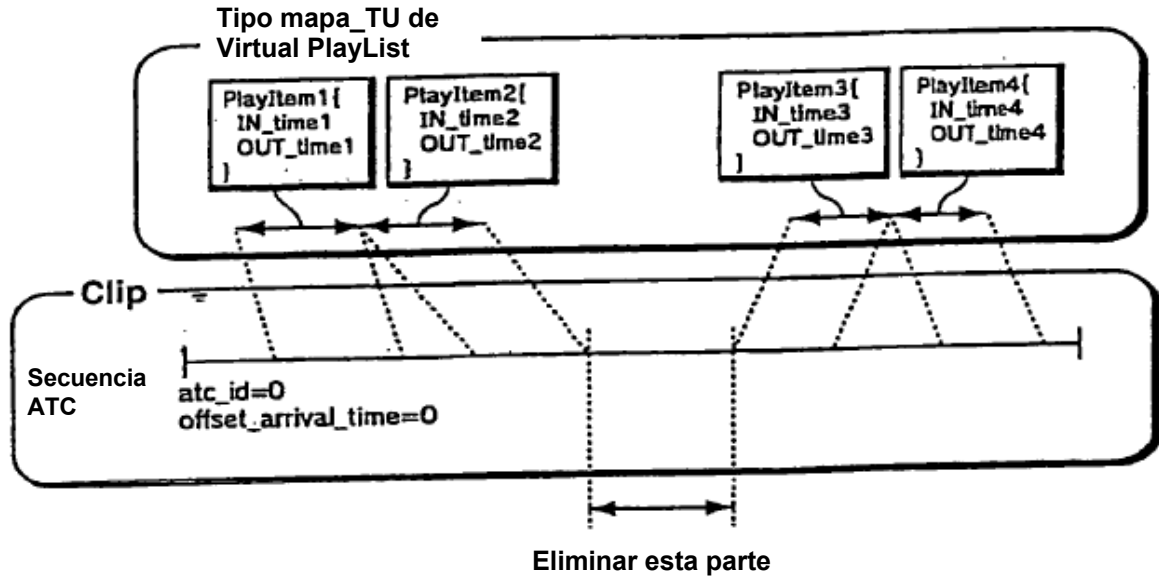


FIG. 42

Antes de edición



Después de edición

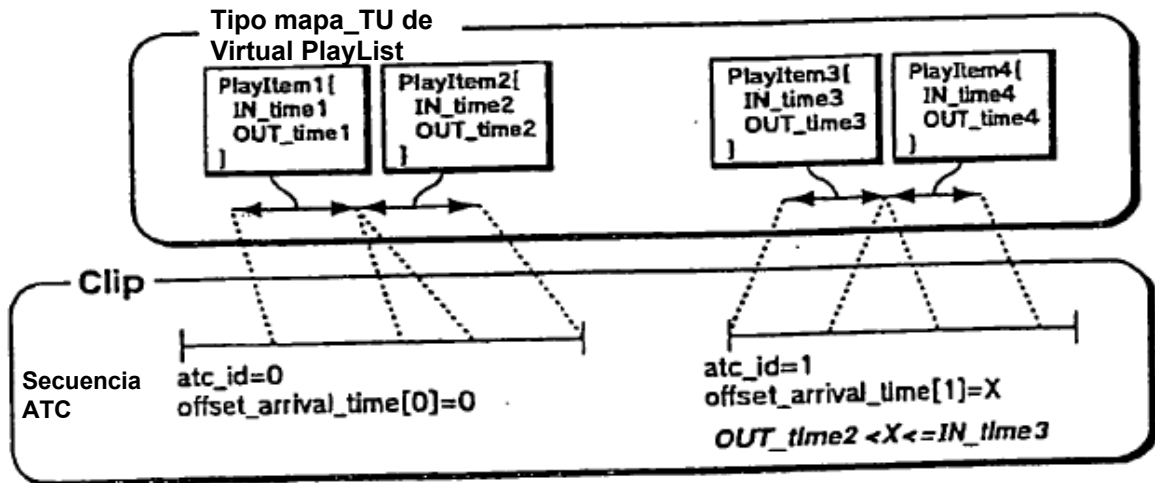


FIG. 43

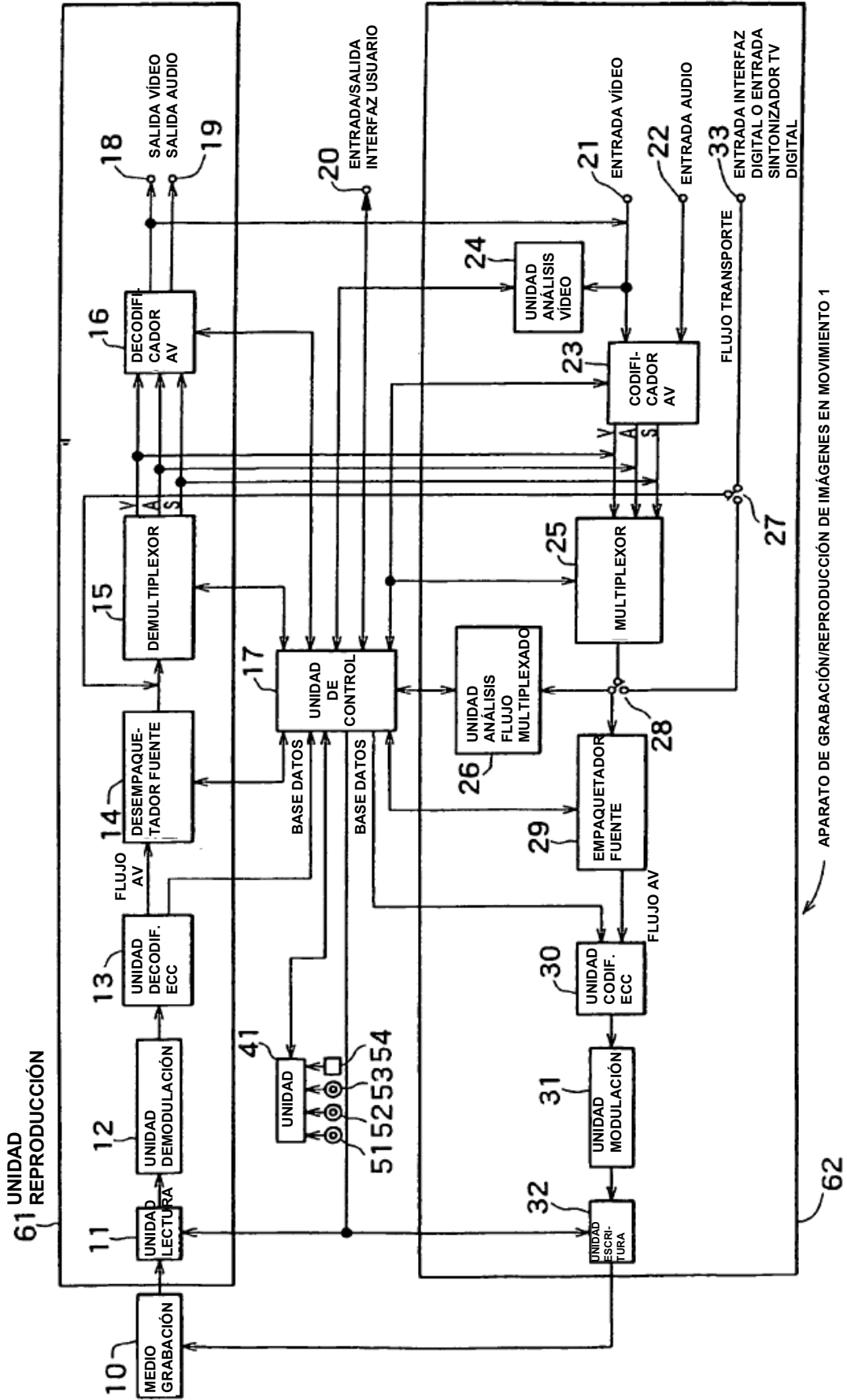


FIG. 44

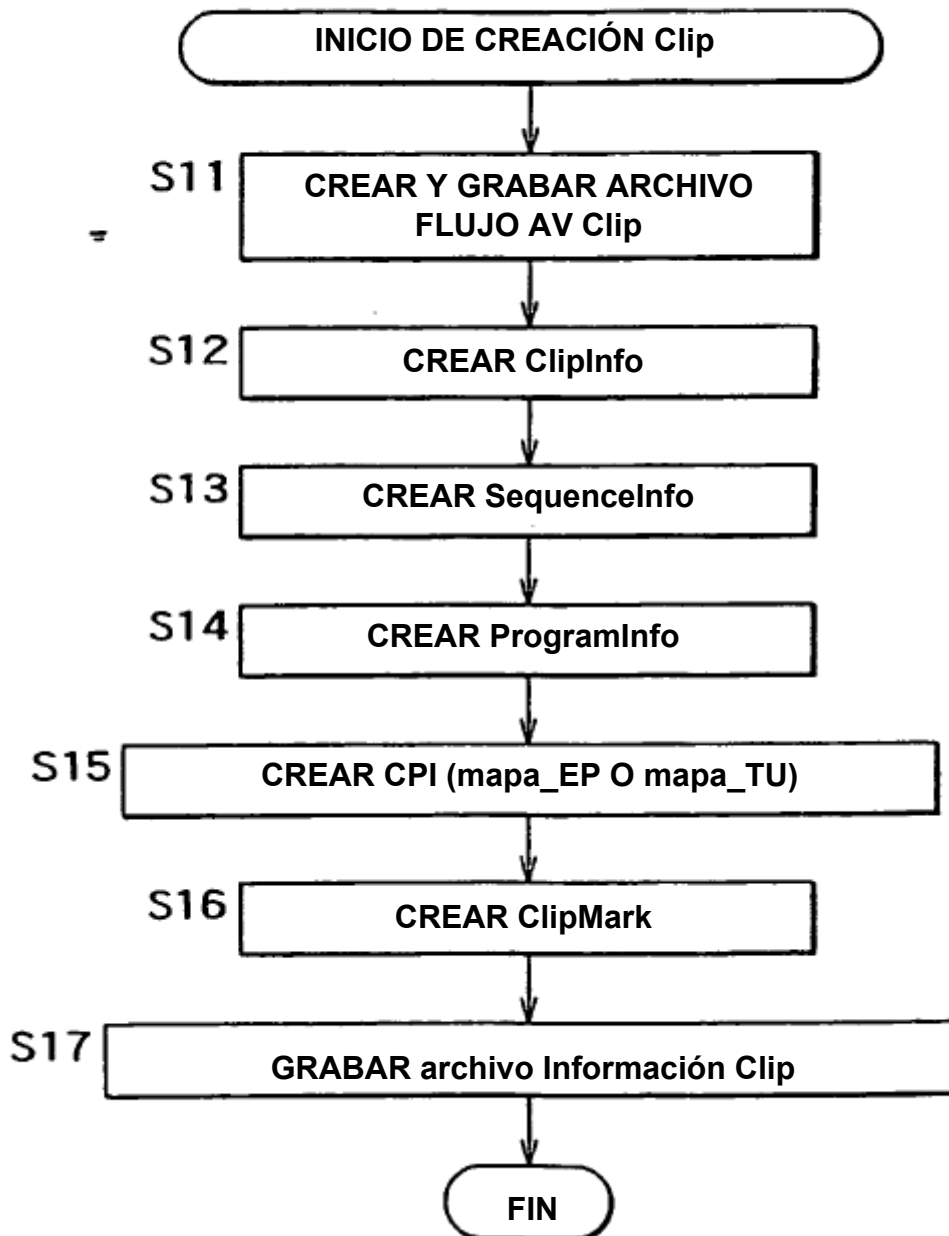


FIG. 45

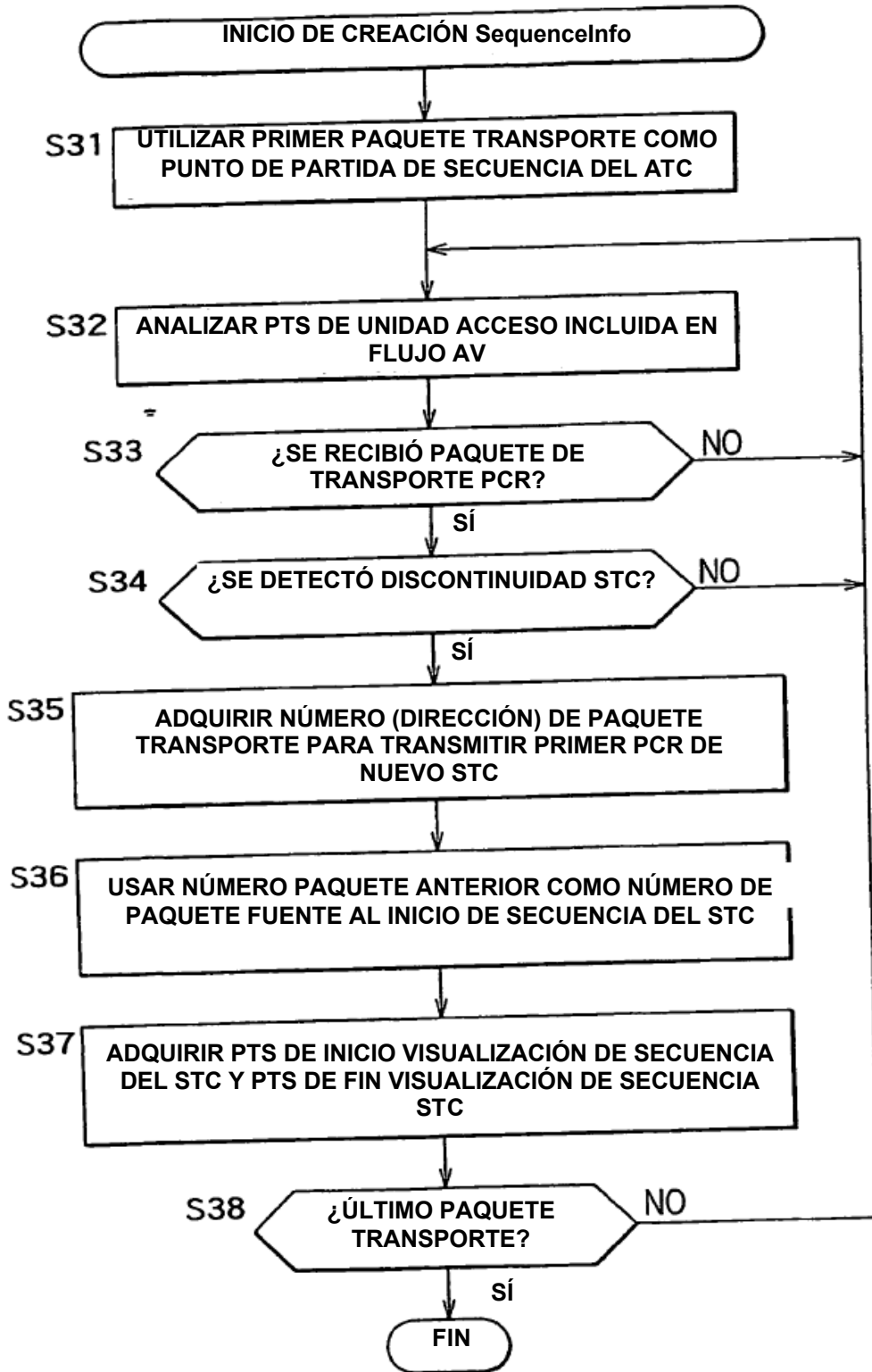


FIG. 46

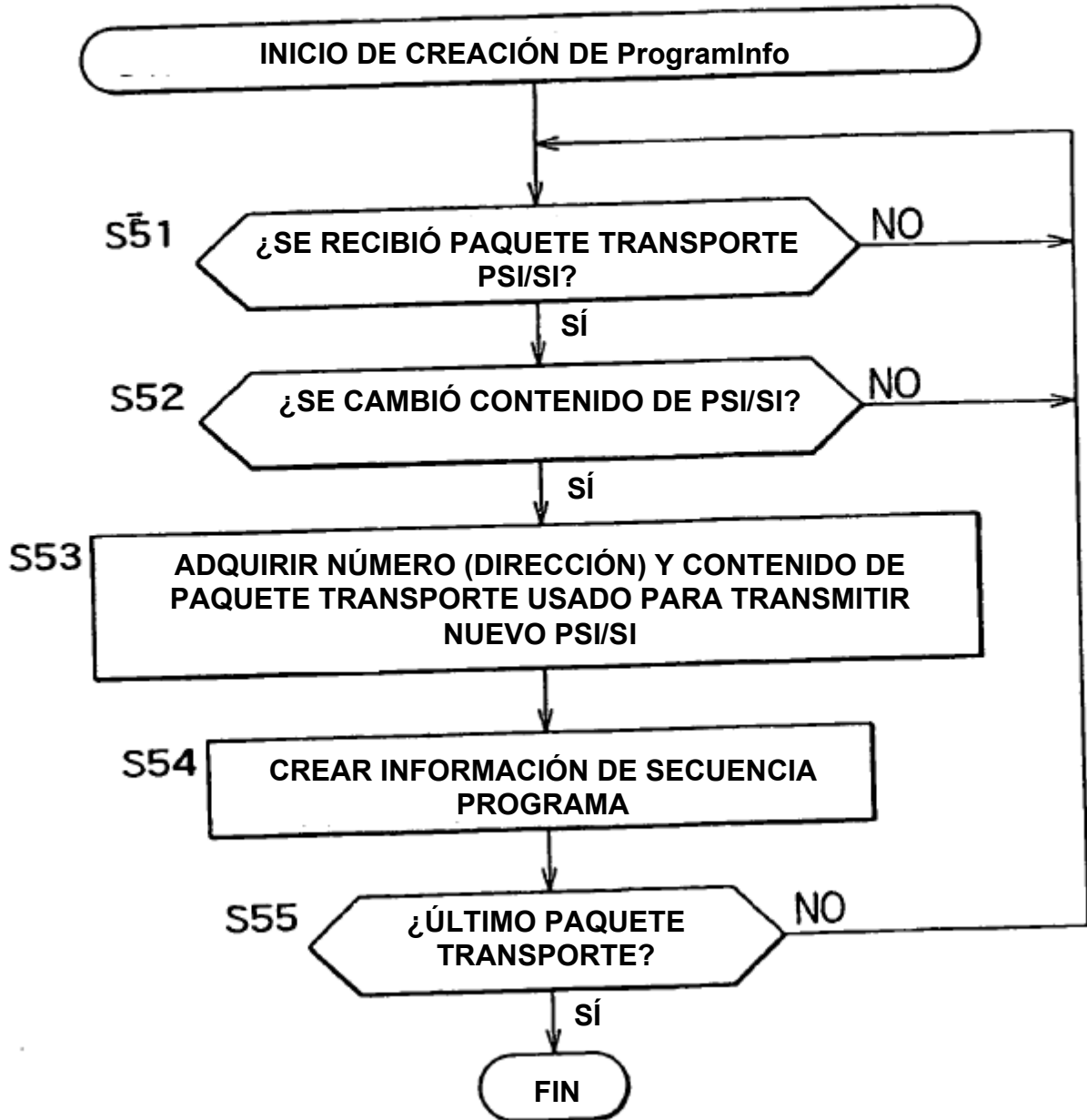


FIG. 47

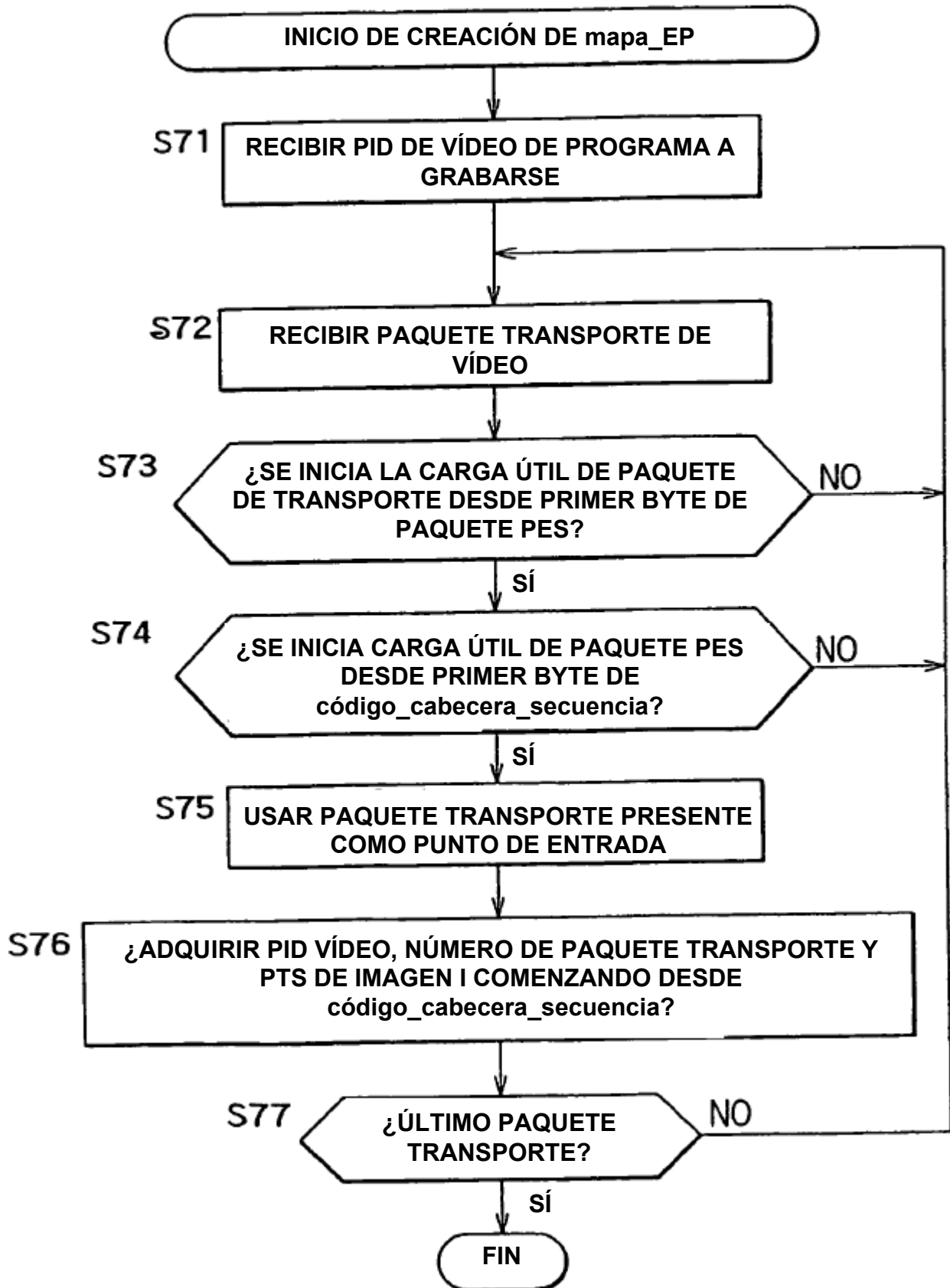


FIG. 48

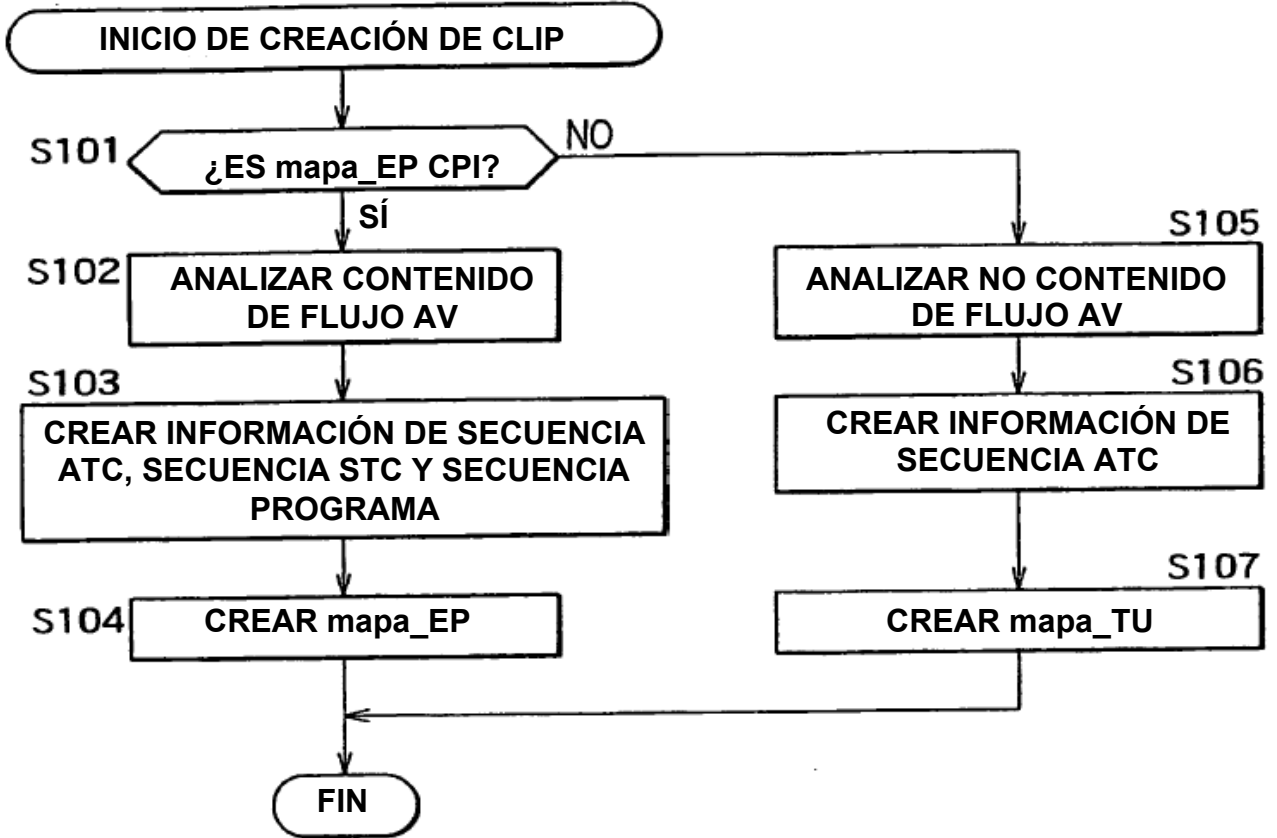


FIG. 49

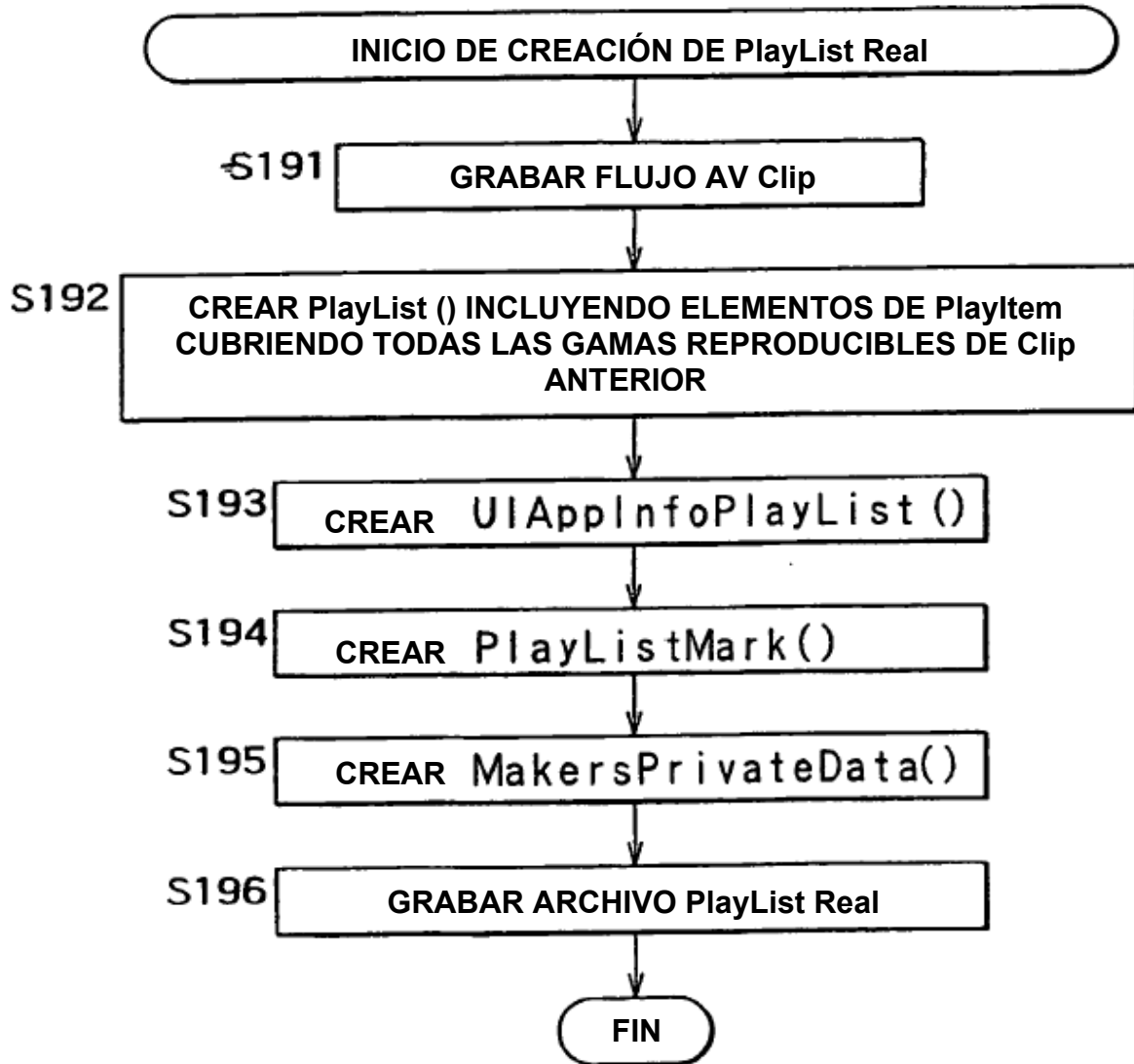


FIG. 50

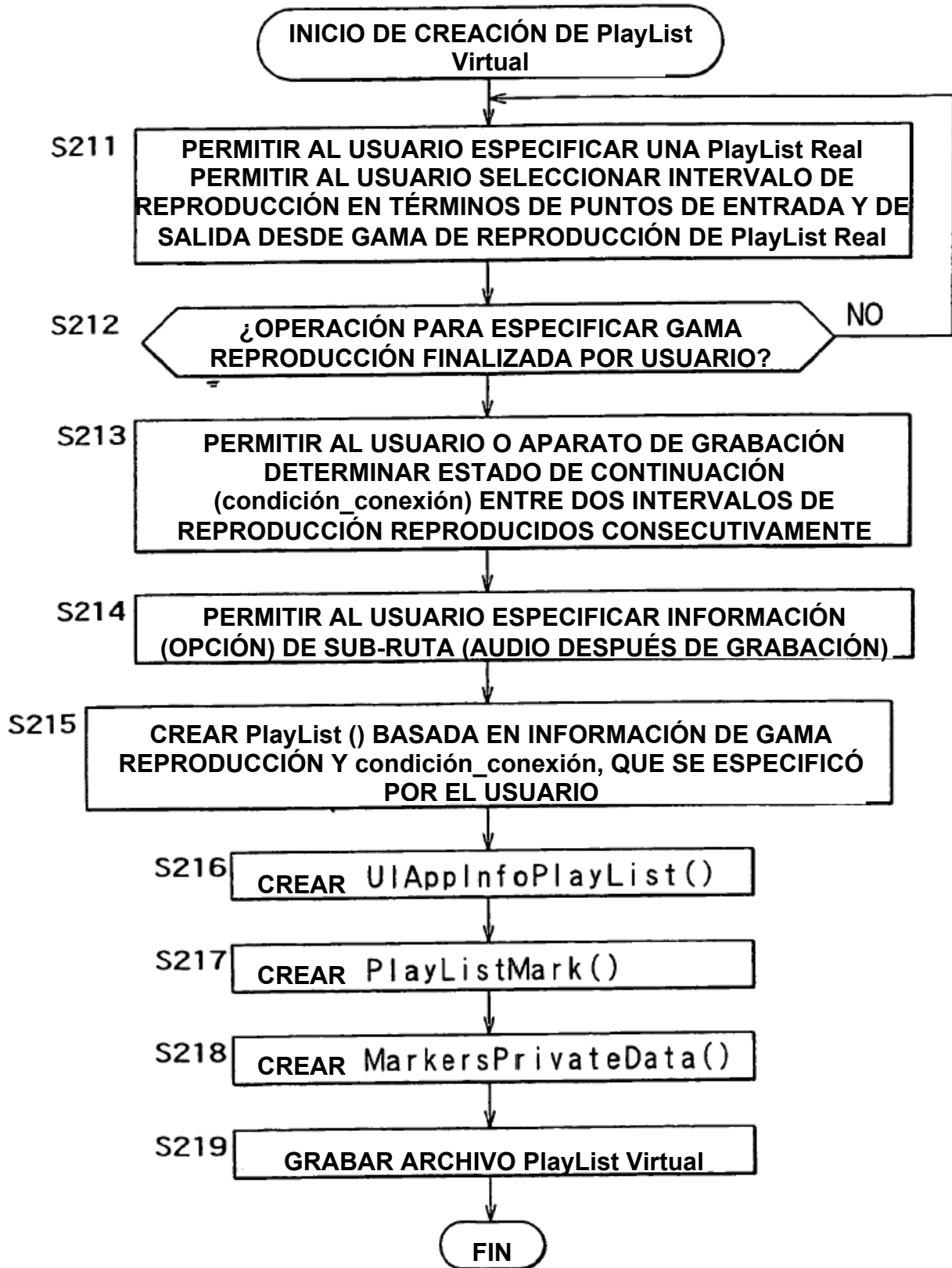


FIG. 51

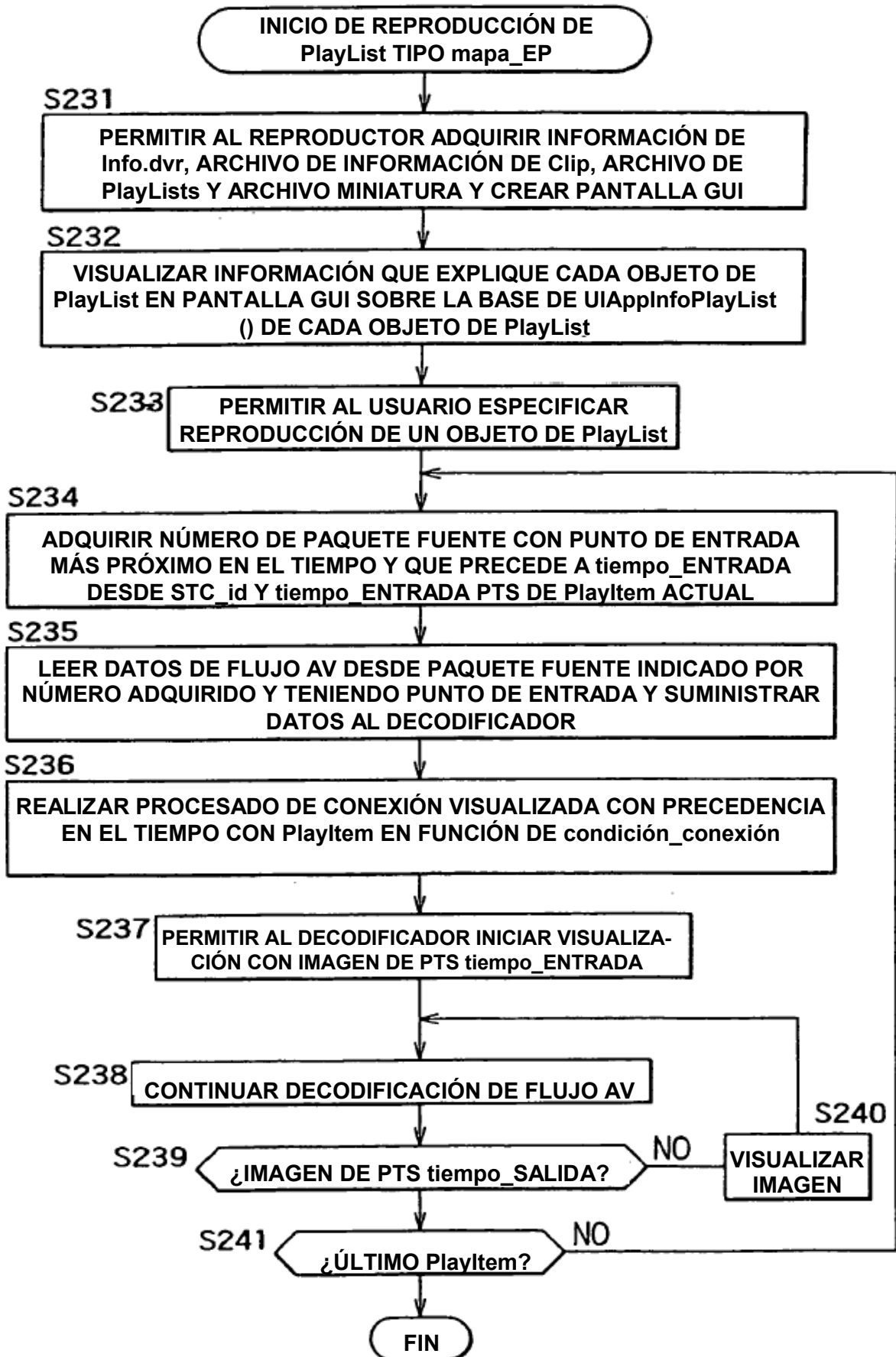


FIG. 52

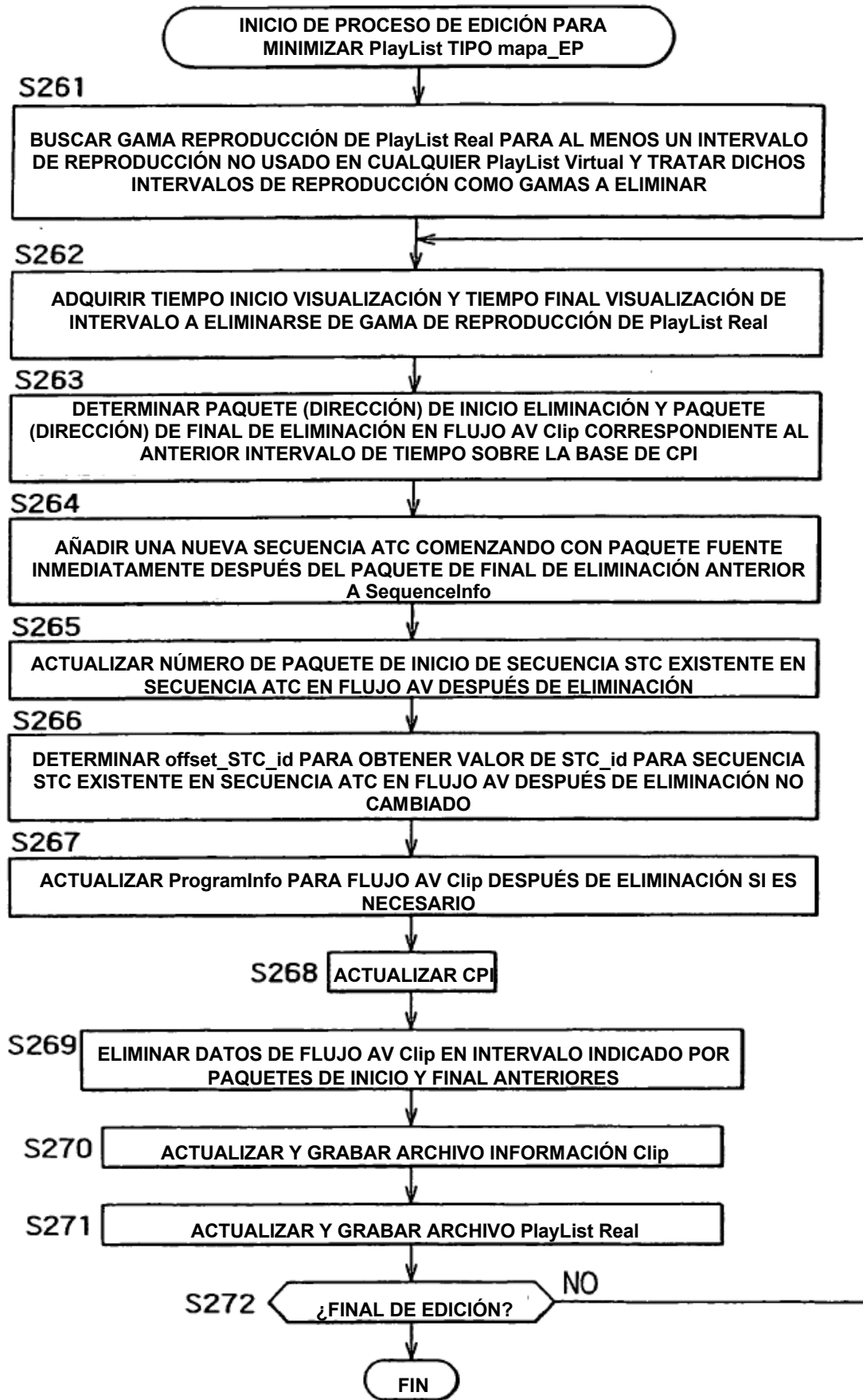


FIG.53

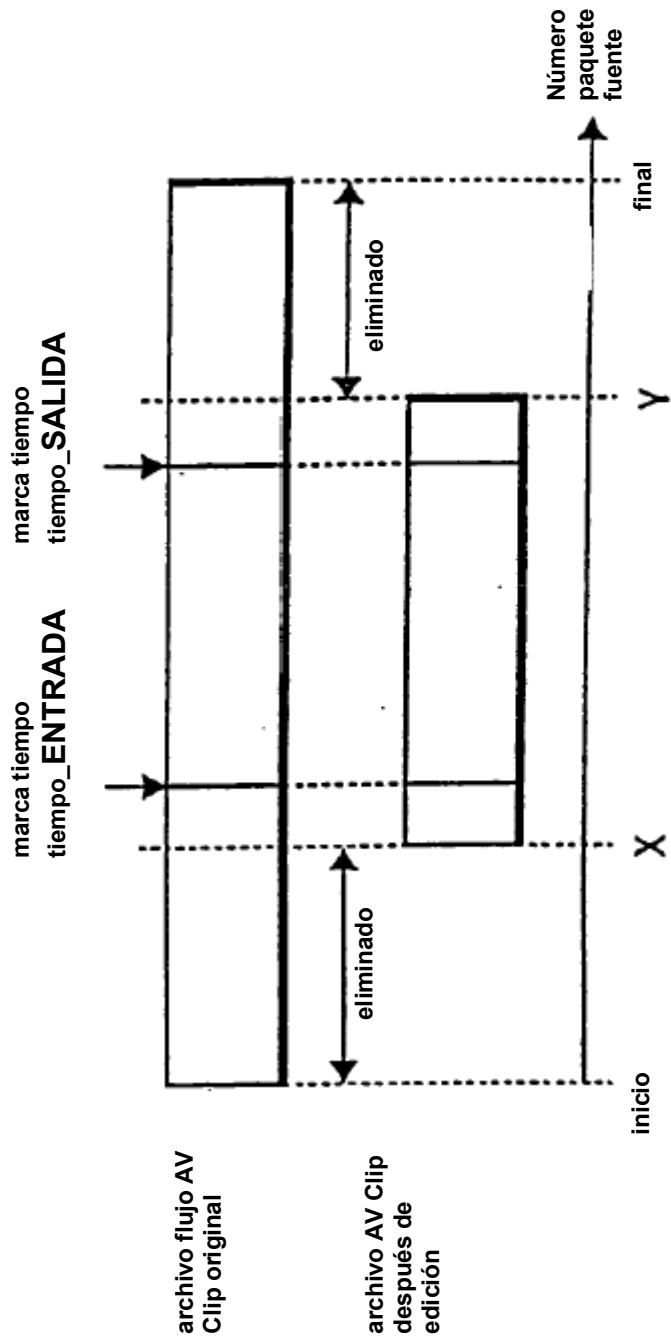


FIG. 54

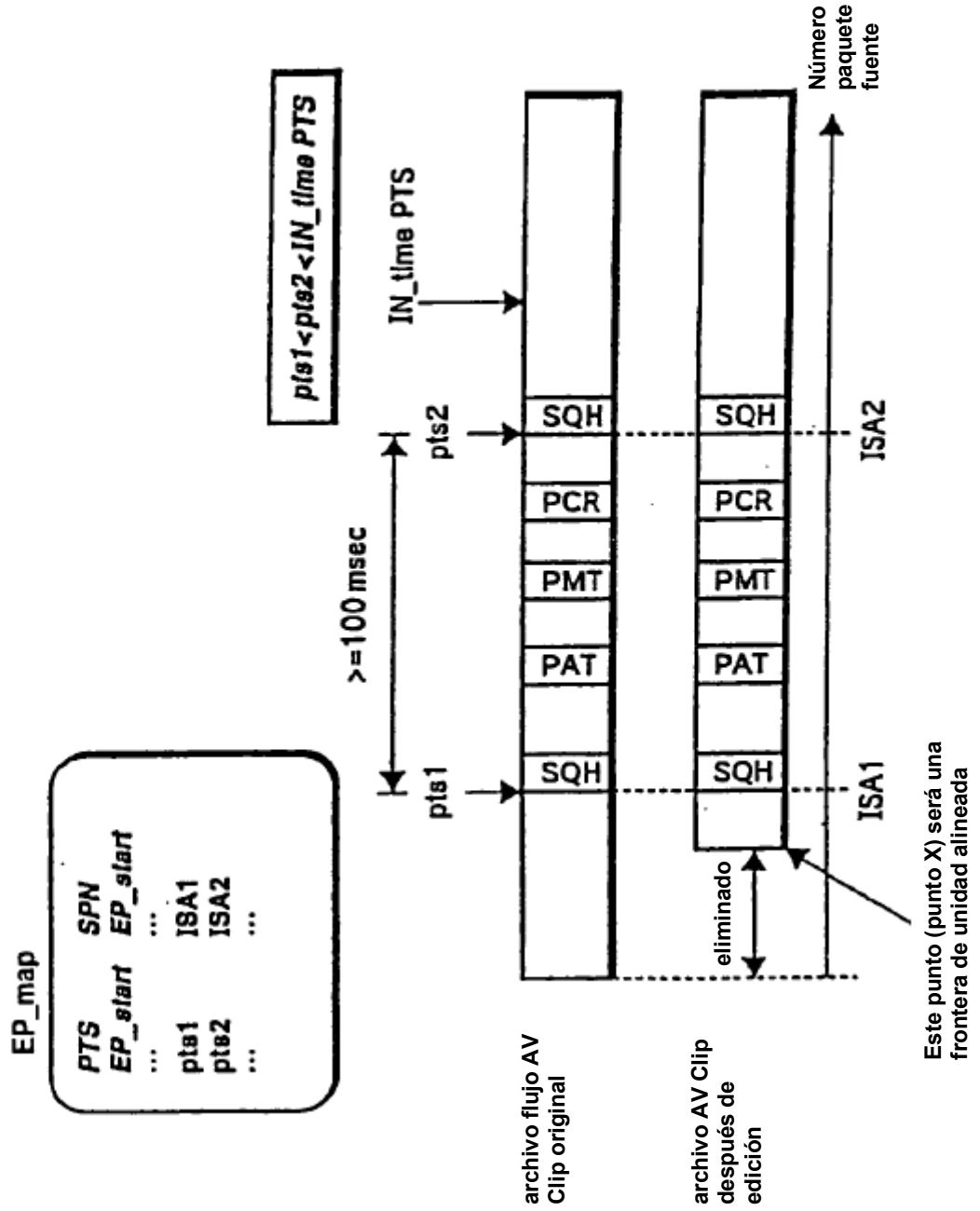


FIG. 55

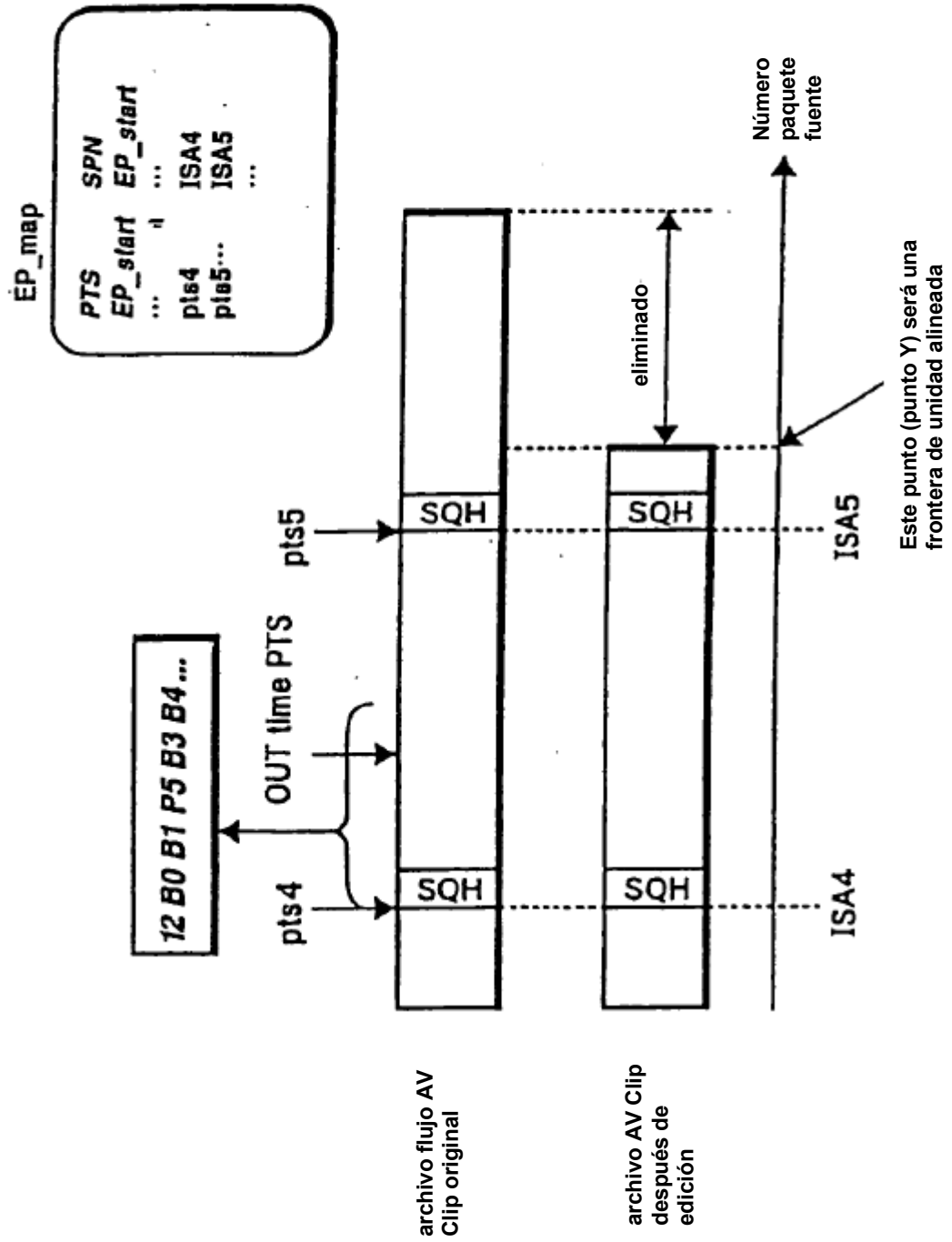


FIG. 56

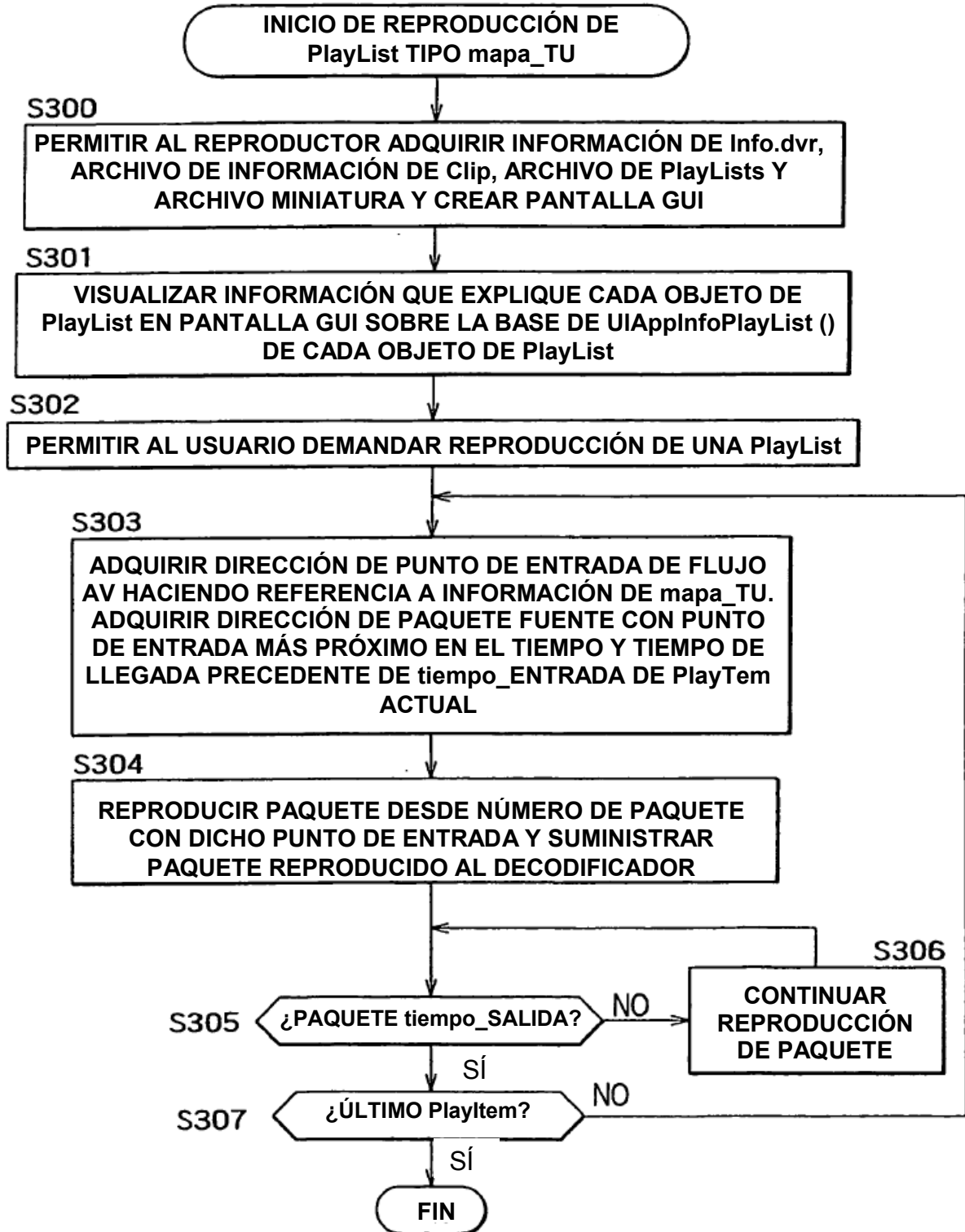


FIG. 57

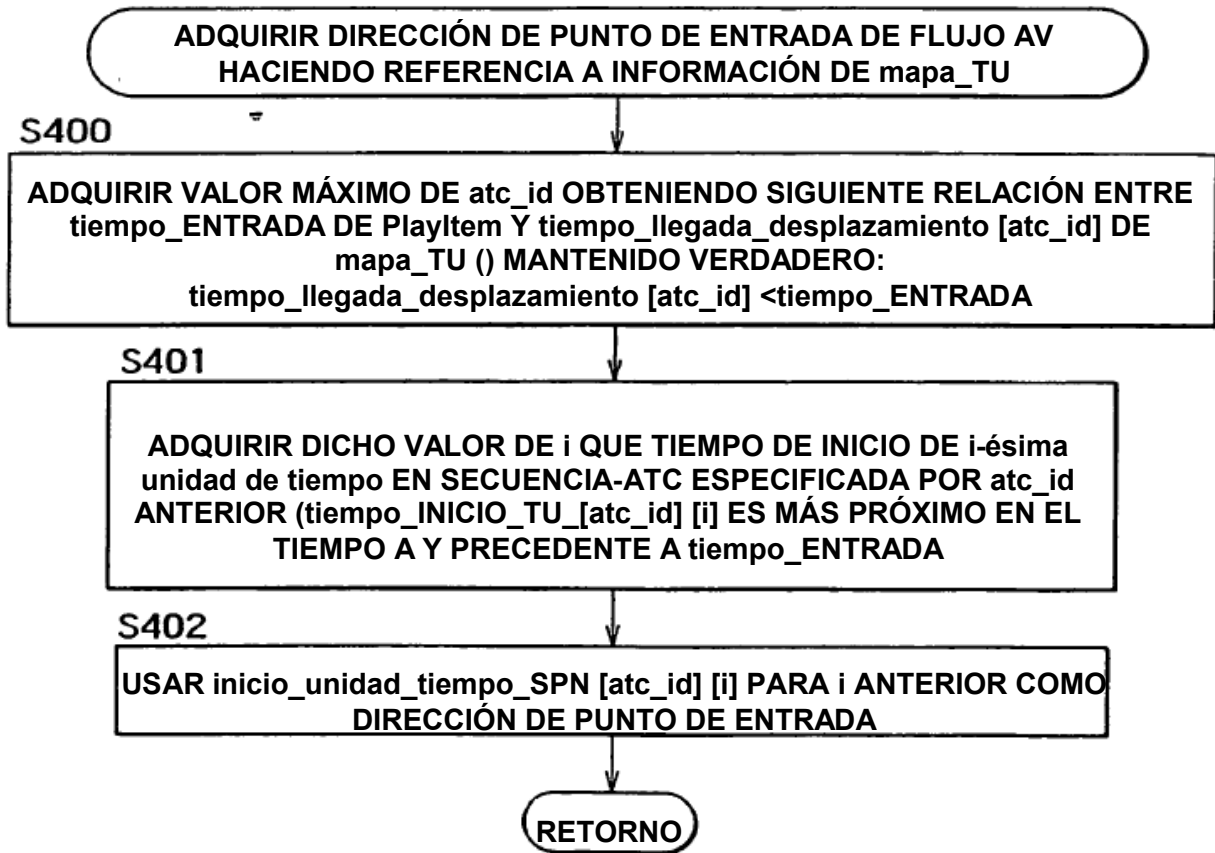


FIG. 58

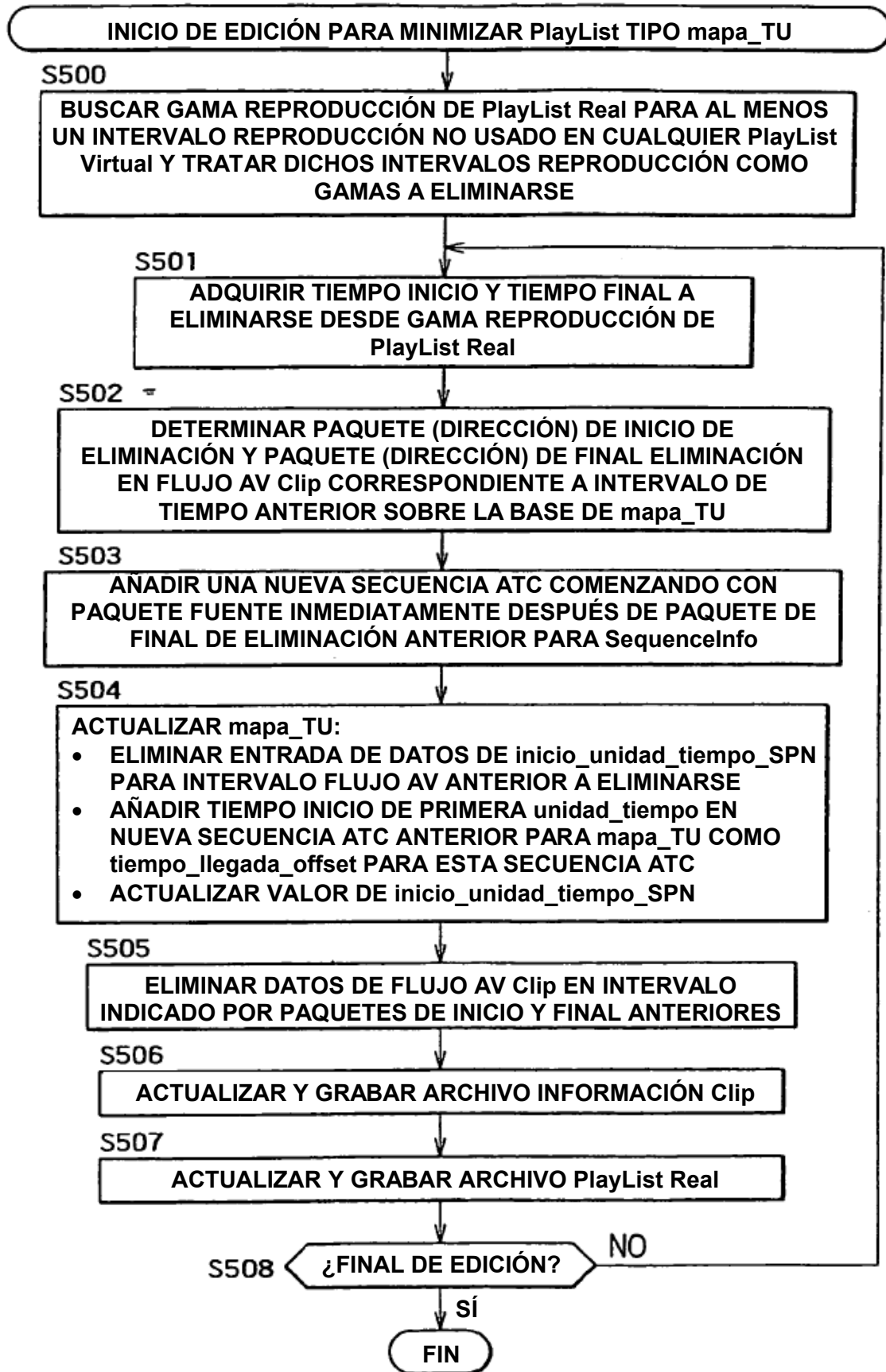


FIG. 59

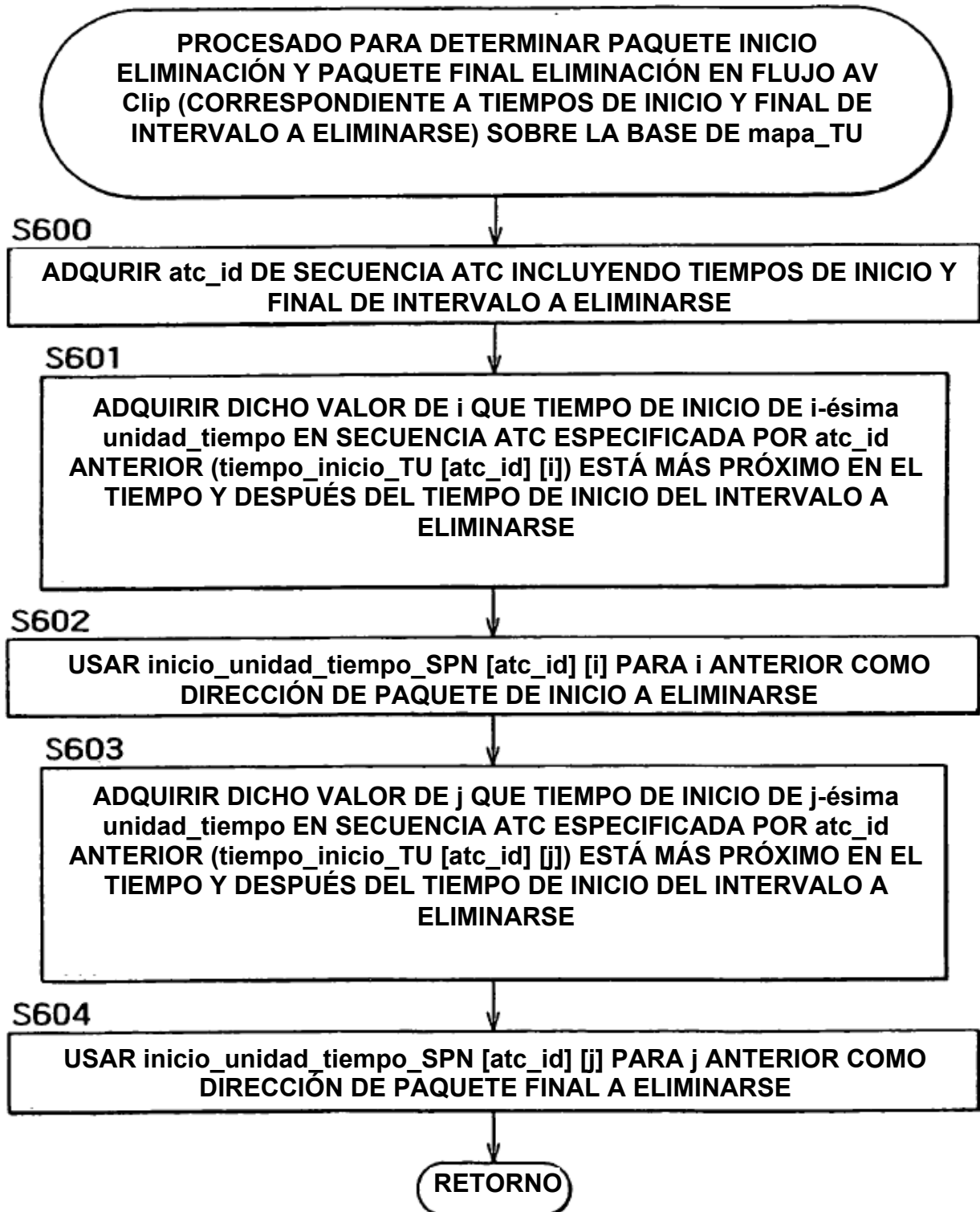


FIG. 60

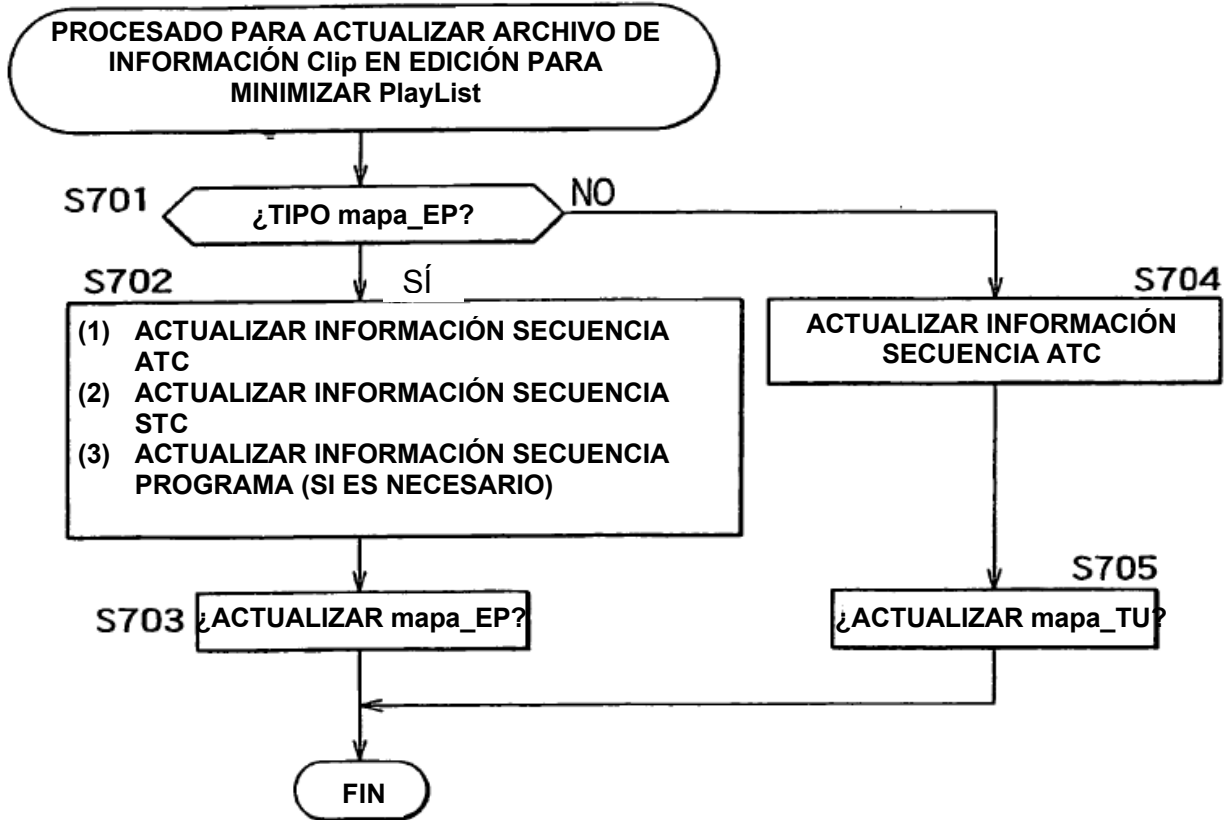


FIG. 61

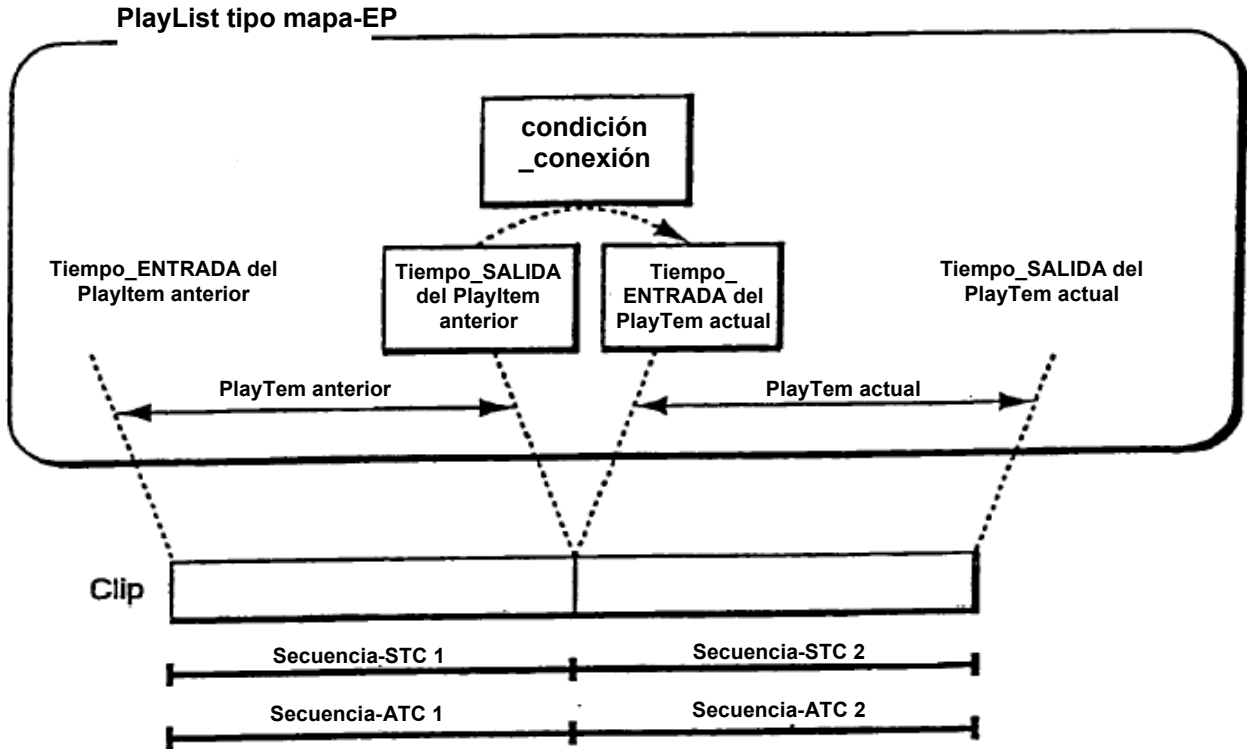


FIG. 62

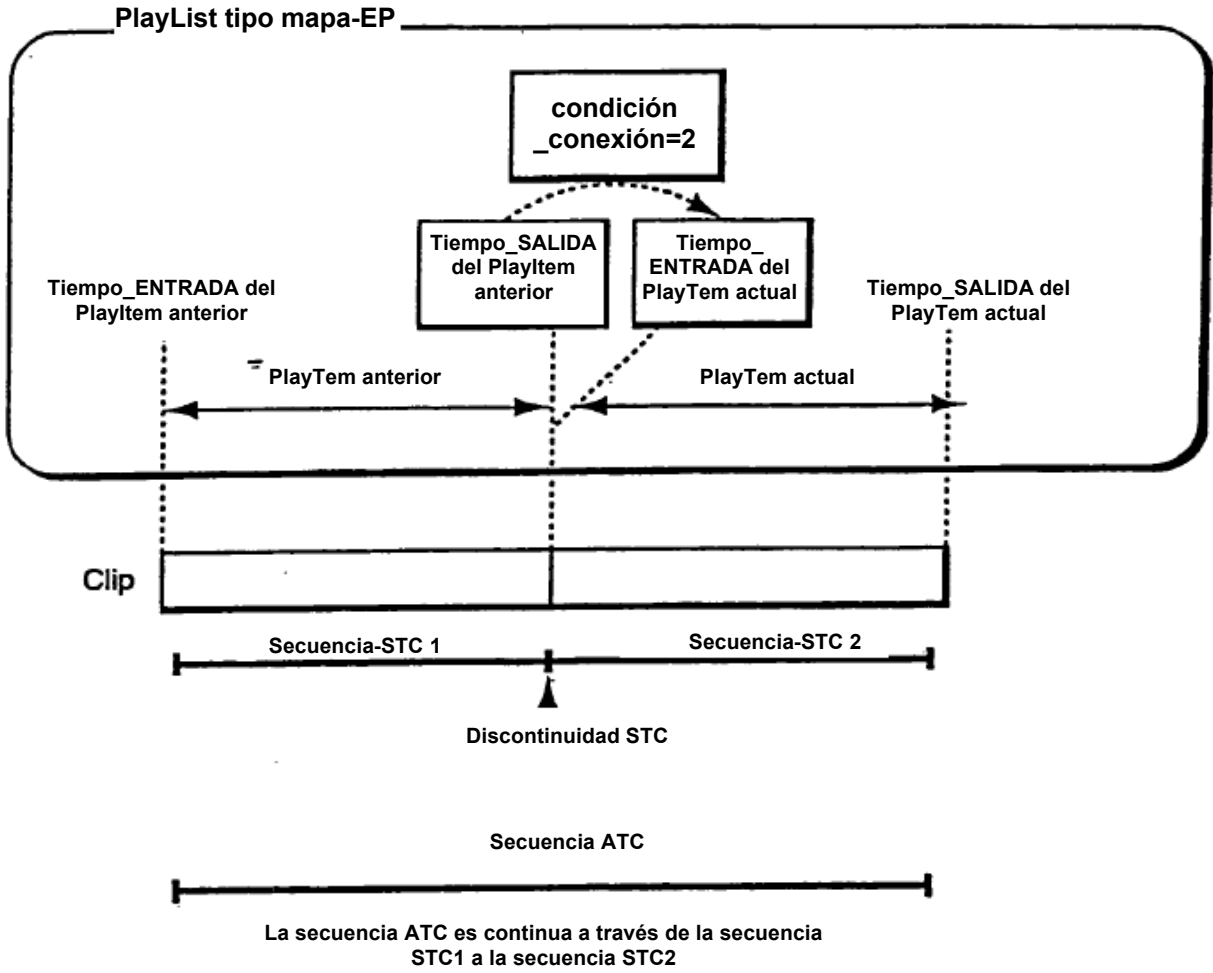


FIG. 63

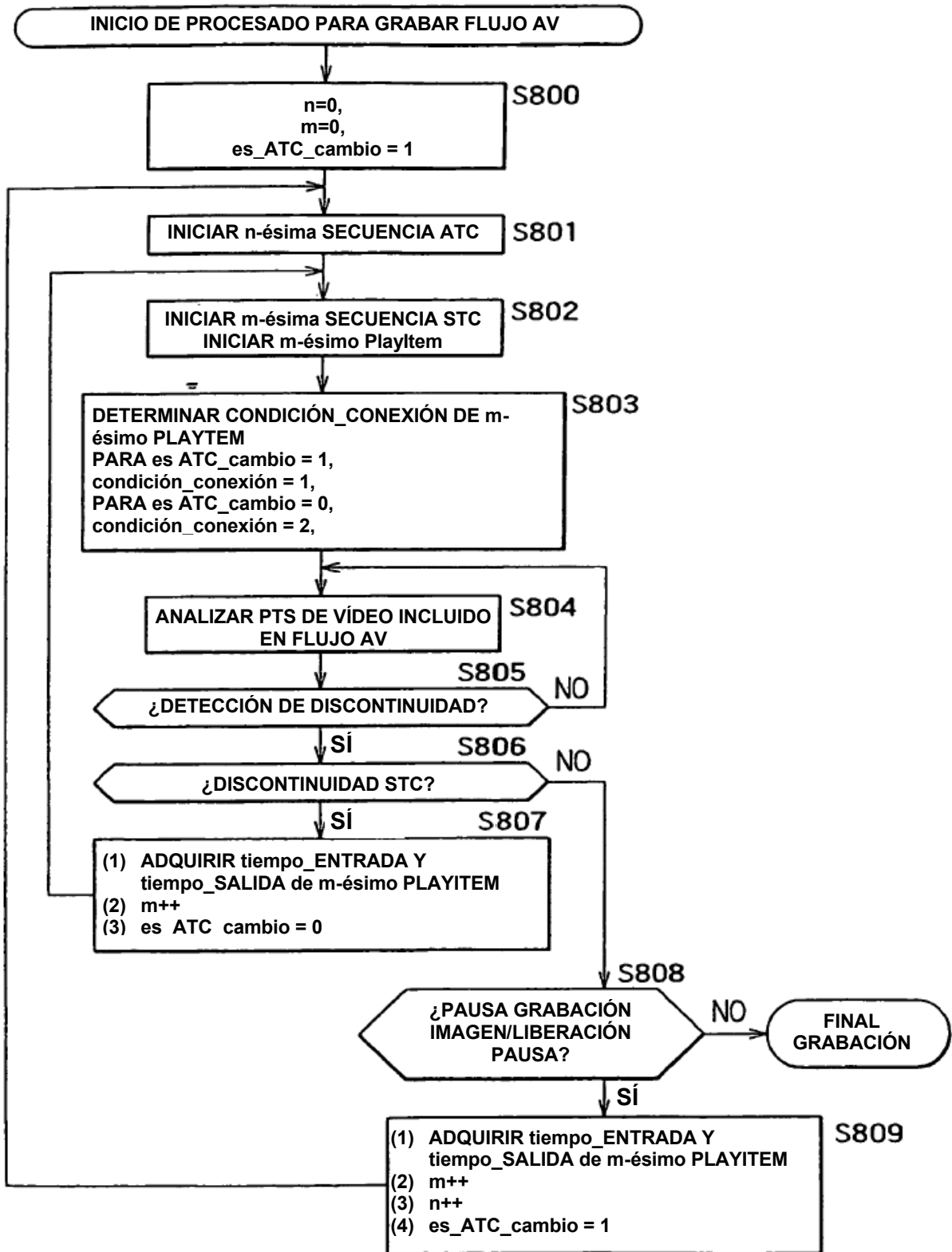


FIG. 64

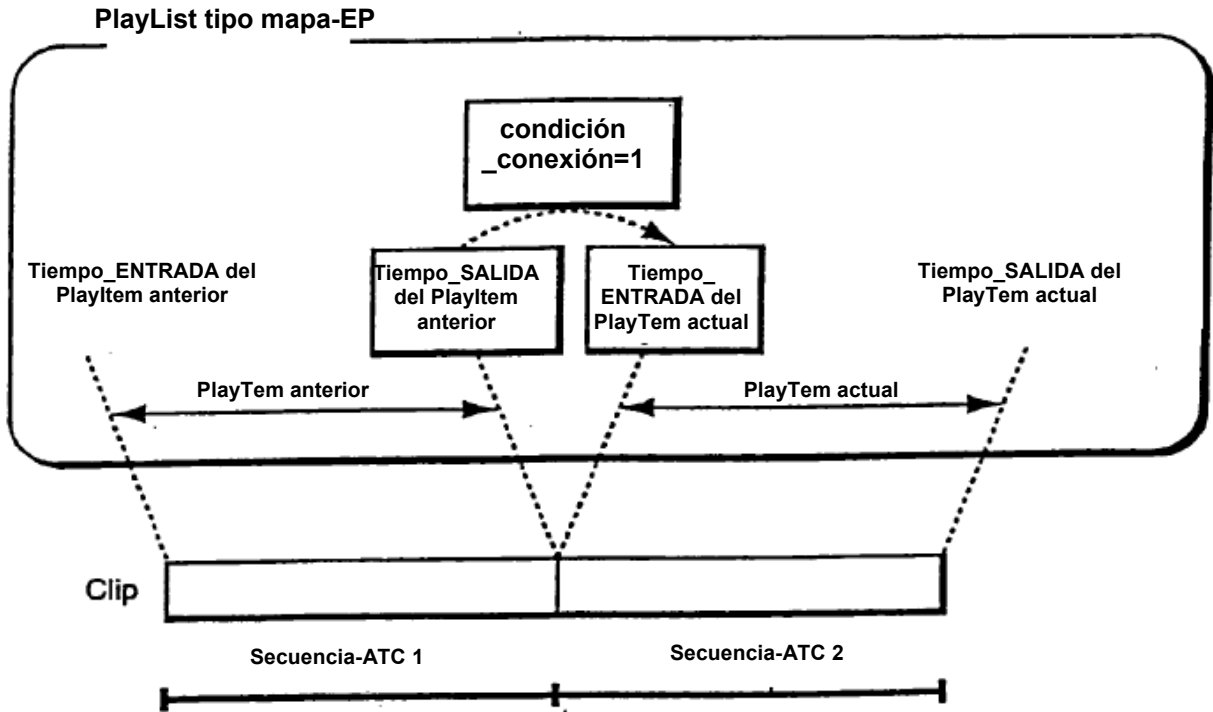


FIG. 65

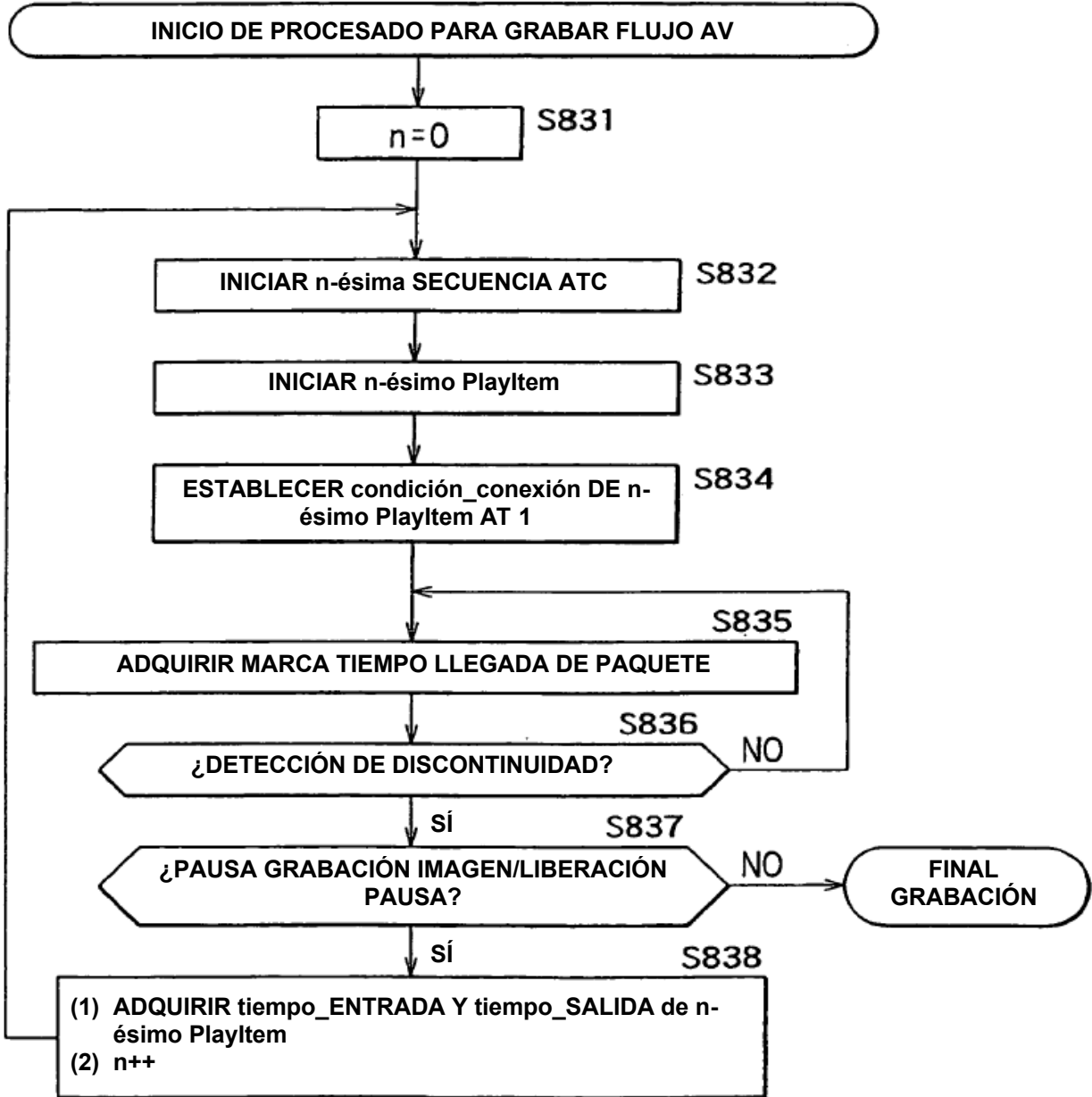


FIG. 66

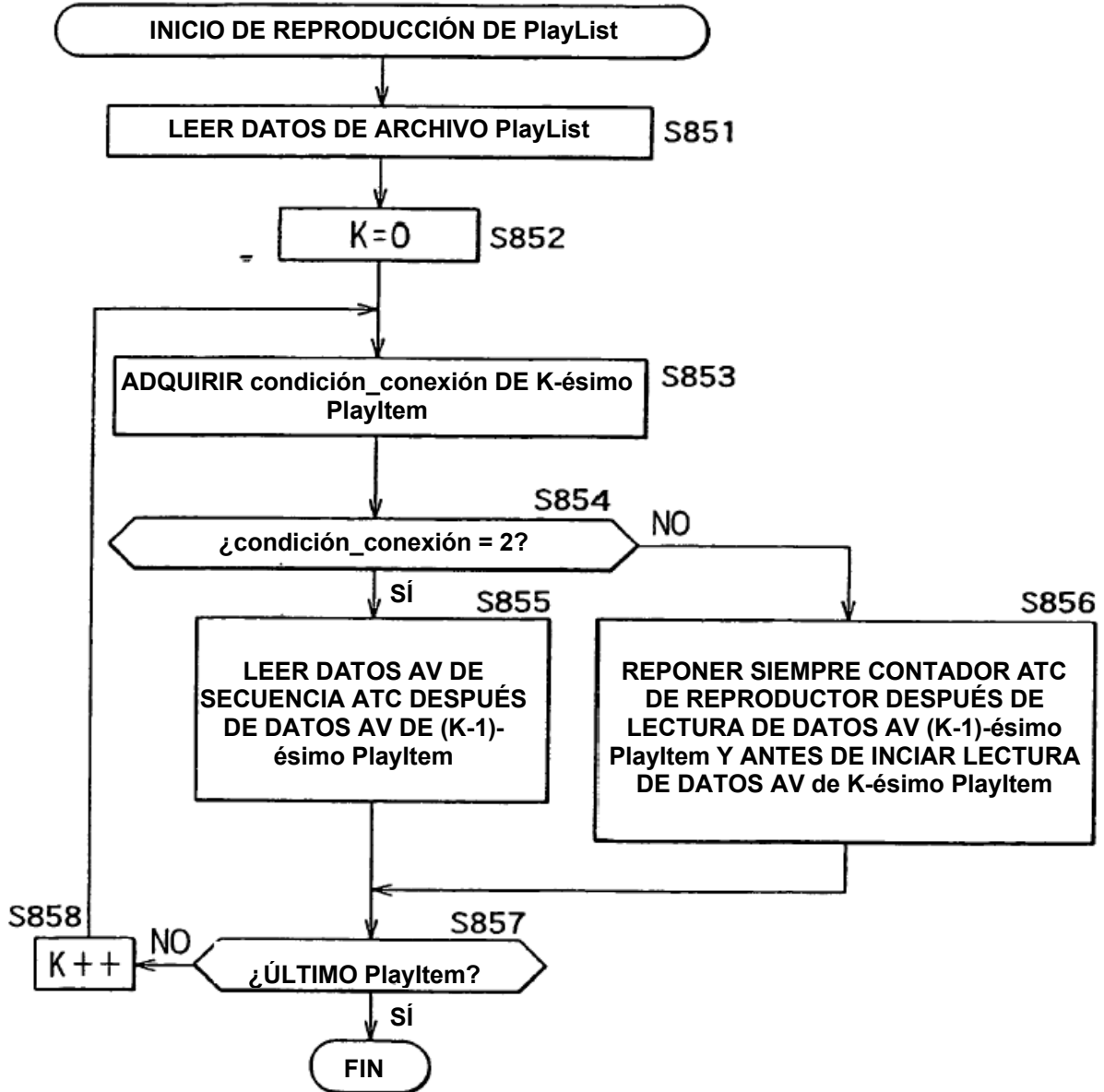


FIG. 67

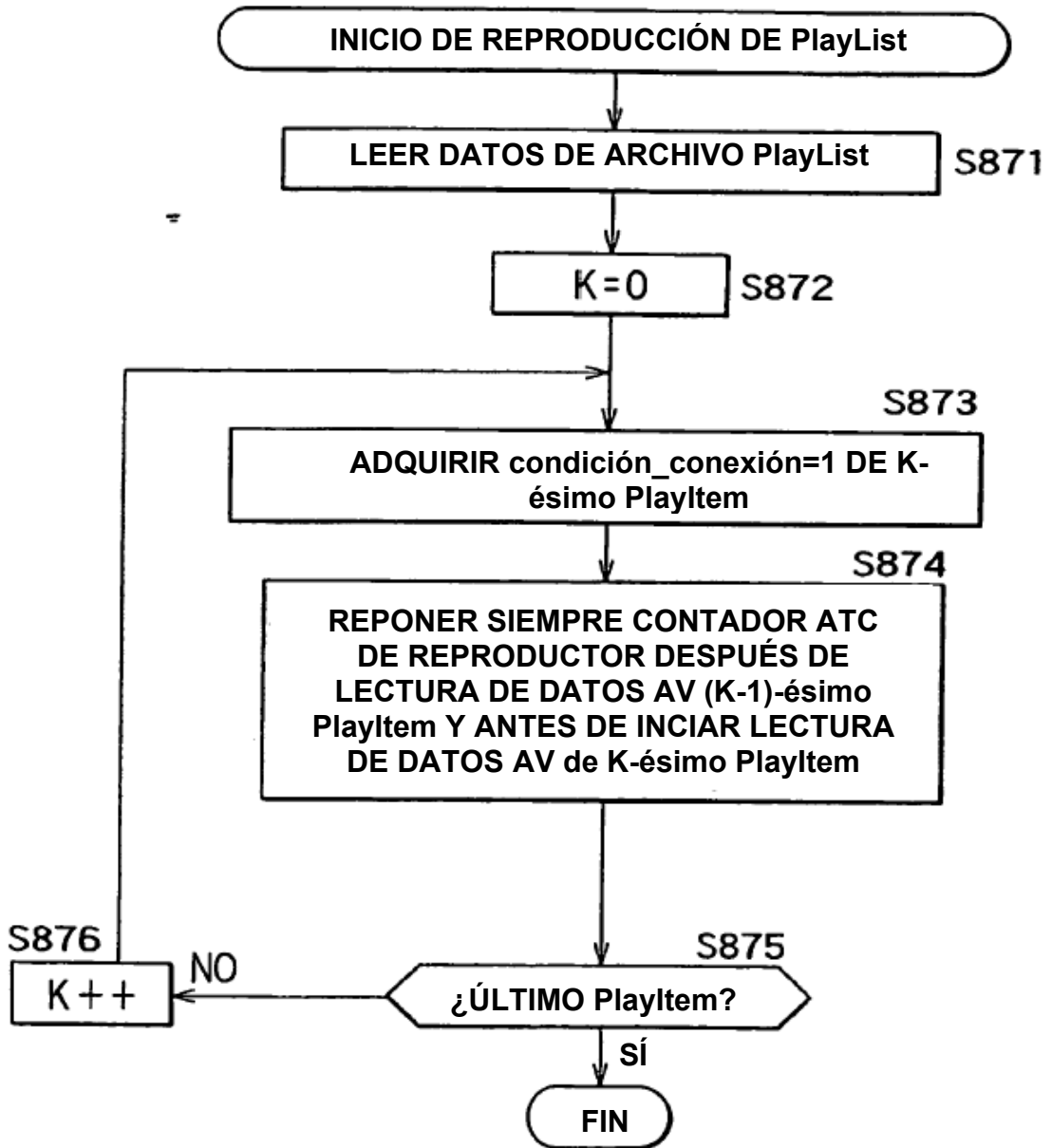


FIG. 68

