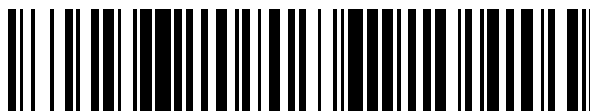


19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 517 418**

51 Int. Cl.:

H04N 5/92 (2006.01)
G11B 20/10 (2006.01)
G11B 27/00 (2006.01)
G11B 27/34 (2006.01)
H04N 5/93 (2006.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

96 Fecha de presentación y número de la solicitud europea: **17.08.2006 E 06782790 (7)**

97 Fecha y número de publicación de la concesión europea: **01.10.2014 EP 1919210**

54 Título: **Dispositivo de reproducción, método de reproducción, programa, soporte de almacenamiento de programa, estructura de datos y método de fabricación de un soporte de registro**

30 Prioridad:

25.08.2005 JP 2005244612
12.09.2005 JP 2005264041
26.10.2005 JP 2005311477

45 Fecha de publicación y mención en BOPI de la traducción de la patente:
03.11.2014

73 Titular/es:

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

72 Inventor/es:

IWASE, AYAKO;
KATO, MOTOKI y
HATTORI, SHINOBU

74 Agente/Representante:

LEHMANN NOVO, María Isabel

ES 2 517 418 T3

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

DESCRIPCIÓN

Dispositivo de reproducción, método de reproducción, programa, soporte de almacenamiento de programa, estructura de datos y método de fabricación de un soporte de registro

CAMPO TÉCNICO

La presente invención se refiere a aparatos de reproducción, métodos de reproducción, programas, soporte de almacenamiento de programas, estructura de datos y métodos de fabricación de soporte de registro. En particular, la presente invención se refiere a un aparato de reproducción, un método de reproducción, un programa, un medio de memorización de programa, una estructura de datos y un método de fabricación de soporte de registro que se utilizan, de forma preferida, para reproducir datos de flujos.

ANTECEDENTES DE LA TÉCNICA ANTERIOR

Con el fin de escuchar/ver diferentes contenidos simultáneamente, existe una tecnología en donde datos de imágenes sin comprimir se combinan después de que se decodifiquen datos de vídeo de múltiples contenidos, siendo luego los datos combinados sometidos a una conversión D/A (digital/analógica), proporcionándose los datos convertidos a un terminal de salida de vídeo para su visualización en un aparato de presentación visual externo (p.e., Documento de Patente 1).

[Documento de Patente 1] Solicitud de patente japonesa no examinada con número de publicación 200520242

A modo de ejemplo, un aparato de reproducción se describe en el documento WO 2004/109467 A2.

SUMARIO DE LA INVENCION

Existe una demanda para combinar múltiples datos de vídeo para su salida, no solamente cuando los datos de vídeo de múltiples contenidos se combinan para la presentación visual, según se describió anteriormente, sino también cuando un contenido incluye múltiples datos de vídeo.

Cuando múltiples datos de vídeo se combinan para la salida, resulta difícil para un productor de contenidos, distribuidor de contenidos o similar, en lugar de depender de un aparato de reproducción, establecer un método para visualizar una pantalla de presentación visual que sirve como una subpantalla relativa a una pantalla de presentación visual principal, a modo de ejemplo, para cada contenido o cada parte de reproducción predeterminada de un contenido.

Es un objetivo de la presente invención dar a conocer un aparato de reproducción, un método de reproducción, un programa y un soporte de almacenamiento de programa, con los que la señal de vídeo de un fichero de flujo principal y la señal de vídeo de un fichero de flujo secundario pueden combinarse en conformidad con una especificación de un productor de contenidos, distribuidor de contenidos o similar.

Este objetivo se consigue por un aparato de reproducción, un método de reproducción, un programa y un soporte de almacenamiento de programas en conformidad con las reivindicaciones independientes adjuntas. Características ventajosas de la presente invención se definen en las reivindicaciones secundarias correspondientes.

Con la presente invención, un productor de contenidos o un distribuidor de contenidos puede definir adecuadamente, a modo de ejemplo, el tamaño y la posición de visualización de una pantalla de presentación visual secundaria durante la visualización, imagen en imagen, independiente de las especificaciones del aparato de reproducción. Además, en particular, proporcionando un umbral para información que indique un brillo para procesamiento de transparencia es posible proporcionar una señal de vídeo secundaria mínima para visualizarse haciendo así posible impedir que la señal de vídeo primaria se haga difícil de ver. Además, de este modo es posible combinar, de forma flexible, señales de vídeo secundarias y señales de vídeo primarias.

El término "red" se refiere a un sistema en el que al menos dos aparatos están conectados con el fin de permitir a un aparato transmitir información a otro aparato. Los aparatos que realizan la comunicación a través de la red pueden ser independientes entre sí o pueden ser bloques internos incluidos en un solo aparato.

Además, la comunicación puede incluir la comunicación inalámbrica y la comunicación cableada. De este modo, la comunicación puede ser una comunicación que utilice una combinación de comunicación inalámbrica y de comunicación cableada. Es decir, la comunicación inalámbrica puede realizarse en una sección y la comunicación cableada puede realizarse en otra sección. Además, una comunicación desde un primer aparato a un segundo aparato puede realizarse utilizando la comunicación inalámbrica y una combinación desde el segundo aparato al primer aparato puede realizarse utilizando una comunicación inalámbrica.

Un aparato de registro puede ser un aparato independiente o puede ser parte de un aparato de

registro/reproducción. Además, a continuación la información de presentación visual, según se menciona en conjunto de reivindicaciones, se denomina también "información relacionada con el estado de presentación visual".

BREVE DESCRIPCIÓN DE LOS DIBUJOS

- 5 La Figura 1 es un diagrama que ilustra, a modo de ejemplo, un formato de aplicación de un soporte de registro a cargarse en un aparato de reproducción según la presente invención.
- 10 La Figura 2 es un diagrama que ilustra las estructuras de una ruta principal y rutas secundarias.
- La Figura 3 es un diagrama que ilustra, a modo de ejemplo, una ruta principal y una ruta secundaria.
- La Figura 4 es un diagrama que ilustra, a modo de otro ejemplo, una ruta principal y una ruta secundaria.
- 15 La Figura 5 es un diagrama que ilustra una pantalla de presentación visual principal y una pantalla de presentación visual secundaria.
- La Figura 6 es un diagrama que ilustra, a modo de ejemplo, el sistema de ficheros de ficheros de datos que se pueden reproducir.
- 20 La Figura 7 ilustra la estructura de datos de un fichero de lista de reproducción que se memoriza en un directorio denominado PLAYLIST y que tiene una extensión ".mpls"
- La Figura 8 ilustra la sintaxis de un parámetro PlaylistExtensionData().
- 25 La Figura 9 ilustra, a modo de primer ejemplo la sintaxis de pip_metadata que se establece en data_block.
- La Figura 10 ilustra el valor y el significado de pip_metadata_type que se ilustra en la Figura 9.
- 30 La Figura 11 ilustra, a modo de segundo ejemplo, la sintaxis de pip_metadata establecido en el bloque de datos data_block.
- La Figura 12 ilustra, a modo de tercer ejemplo, la sintaxis de pip_metadata establecida en el bloque data_block.
- 35 La Figura 13 ilustra el valor y el significado de pip_scale que se ilustra en la Figura 12.
- La Figura 14 ilustra el valor y el significado de pip_timeline_type que se ilustra en la Figura 12.
- La Figura 15 ilustra el significado de pip_metadata_time_stamp cuando pip_timeline_type que se ilustra en la Figura 40 12 es 1.
- La Figura 16 ilustra el significado de pip_metadata_time_stamp cuando pip_meline_type que se ilustra en la Figura 12, es 2.
- 45 La Figura 17 ilustra el significado de pip_metadata_time_stamp cuando pip_timeline_type que se ilustra en la Figura 12 es 3.
- La Figura 18 ilustra la sintaxis de Playlist().
- 50 La Figura 19 ilustra la sintaxis de SubPath().
- La Figura 20 ilustra SubPath_type.
- La Figura 21 ilustra la sintaxis de SubPlayItem(i).
- 55 La Figura 22 ilustra la sintaxis de PlayItem().
- La Figura 23 ilustra la sintaxis de STN_table().
- 60 La Figura 24 ilustra la sintaxis de stream_entry().
- La Figura 25 ilustra la sintaxis de stream_attribute().
- La Figura 26 ilustra stream_coding_type.
- 65 La Figura 27 ilustra video_format.

La Figura 28 ilustra la tasa de trama `frame_rate`.

La Figura 29 ilustra la relación de aspecto `aspect_ratio`.

5 La Figura 30 ilustra el tipo de presentación de audio `audio_presentation_type`.

La Figura 31 ilustra la frecuencia de muestreo `sampling_frequency`.

La Figura 32 ilustra el código de caracteres.

10 La Figura 33 ilustra la sintaxis de `STN_table()`.

La Figura 34 ilustra la sintaxis `STN_table()`.

15 La Figura 35 es un diagrama de bloques que ilustra, a modo de ejemplo, la configuración de un aparato de reproducción según la presente invención.

La Figura 36 es un diagrama de flujo que ilustra el procesamiento de reproducción realizado por el AP adhesión reproducción ilustrado en la Figura 35.

20 La Figura 37 es un diagrama de flujo que ilustra el procesamiento de reproducción de señal de vídeo primario.

La Figura 38 es un diagrama de flujo que ilustra el procesamiento de reproducción de señal de vídeo primario.

25 La Figura 39 es un diagrama de flujo que ilustra el procesamiento de reproducción de señal de vídeo primario.

La Figura 40 es un diagrama de flujo que ilustra el procesamiento de reproducción de señales de vídeo primarias y secundarias.

30 La Figura 41 es un diagrama de flujo que ilustra el procesamiento de reproducción de señales de vídeo primarias y secundarias.

La Figura 42 es un diagrama de flujo que ilustra el procesamiento de reproducción de señales de vídeo primarias y secundarias.

35 La Figura 43 es una vista que ilustra el posicionamiento y la puesta en escala.

La Figura 44 es una vista que ilustra la combinación de `luma_keying`

40 La Figura 45 es una vista que ilustra la combinación denominada `luma_keying`

La Figura 46 es un diagrama que ilustra la fabricación de un soporte de registro sobre el que pueden reproducirse datos por el aparato de reproducción que se registran.

45 La Figura 47 es un diagrama que ilustra la fabricación de un medio de registro en el que pueden reproducirse datos por el aparato de reproducción que se registran.

La Figura 48 es un diagrama que ilustra la configuración de un ordenador personal.

50 Referencias numéricas

1 pantalla de presentación visual principal, 2 pantalla de presentación visual secundaria, 20 aparato de reproducción, 31 soporte de almacenamiento, 32 conmutador, 33 unidad de decodificador AV, 34 controlador, 51 a 54, memorias intermedias 55 y 56, filtros de PID, 57 a 59 conmutadores, 71 decodificador de fondo, 72-1 primer decodificador de vídeo, 72-2 segundo decodificador de vídeo, 73 decodificador de gráficos de presentación, 74 decodificador de gráficos interactivos, 75 decodificador de audio, 76 composición de texto-St, 77 conmutador, 91 generador de plano de fondo, 92 generador de plano de vídeo, 93 generador de plano de gráficos de presentación, 94 generador de plano de gráficos interactivos, 95 memoria intermedia, 96 procesador de datos de vídeo, 97 procesador de mezcla, 101 mezcla.

60 FORMAS DE REALIZACIÓN PREFERIDAS DE LA INVENCION

Una forma de realización de la presente invención se describirá a continuación haciendo referencia a los dibujos adjuntos.

65 La Figura 1 es un diagrama que ilustra, a modo de ejemplo, un formato de aplicación de un soporte de registro a

cargarse en un aparato de reproducción 20 (descrito a continuación con referencia a la Figura 35) según la presente invención. El medio de registro puede ser no solamente un disco óptico descrito a continuación, sino también un disco magnético o una memoria de semiconductores.

El formato de aplicación tiene dos capas para PlayLists (listas de reproducción) y Clips (videoclips) para controlar los flujos de AV (audiovisuales). En este caso, un par de un flujo AV e información de Clip, que sea información que acompañe al flujo AV, se considerará como un solo objeto y se referirá colectivamente como un "Clip". En adelante, un flujo AV puede referirse también como un "fichero de flujo AV". La información de Clip puede referirse también como un "fichero de información de Clip".

En general, los ficheros utilizados para un ordenador, etc., se procesan como cadenas de bytes y el contenido de un fichero de flujo AV se expande en una cronología y un punto de acceso en un Clip se especifica, por, principalmente, una marca temporal en una lista Playlist. Es decir, los parámetros "Playlist" y "Clip" son capas para controlar un flujo AV.

Cuando un punto de acceso en un Clip se indica por una marca temporal en una lista de reproducción Playlist, el fichero de información de Clip se utiliza para encontrar la información de una dirección en la que se inicia la decodificación en el fichero de flujo AV, sobre la base de la marca temporal.

Playlist es un conjunto de segmentos de reproducción de flujo AV. Un segmento de reproducción, en un flujo AV, se denomina un "PlayItem", lo que se expresa por un par de un punto de entrada In-point (punto de inicio de reproducción) y un punto de salida OUT (punto de final de reproducción) de un segmento de reproducción en la cronología. De este modo, Playlist incluye uno o múltiples PlayItem, según se ilustra en la Figura 1.

En la Figura 1, una primera lista Playlist desde la izquierda incluye dos PlayItems, que se refieren, respectivamente a una primera parte mitad y una última parte mitad de un flujo AV contenido en un Clip en el lado de la izquierda. Una segunda lista Playlist, desde la izquierda, incluye un PlayItem, que se refiere a un flujo AV completo contenido en un Clip en el lado derecho. Además, una tercera lista Playlist, desde la izquierda, incluye dos PlayItems, que se refieren respectivamente a una parte del flujo AV contenido en el Clip en el lado izquierdo y una parte del flujo AV contenido en el Clip en el lado derecho.

A modo de ejemplo, cuando un programa de navegación de disco, ilustrado en la Figura 1, especifica el PlayItem en el lado izquierdo contenido en la primera Playlist desde la izquierda como información que indica una posición de reproducción en este punto en el tiempo, la primera parte mitad que se refiere por PlayItem y que está contenida en el flujo AV incluido en el Clip del lado izquierdo es objeto de reproducción. De esta manera, la lista Playlist se utiliza como una información de control de reproducción para controlar la reproducción del fichero de flujo AV.

El programa de navegación de disco tiene una función para controlar la secuencia de reproducción de Playlist y la reproducción interactiva de Playlist. El programa de navegación de disco tiene también, a modo de ejemplo, una función para visualizar una pantalla de menú mediante la cual un usuario emite una instrucción para realizar varios tipos de reproducción. El programa de navegación de disco es objeto de escritura en un lenguaje de programación, tal como Java (marca registrada) y se prepara en un soporte de registro.

En la presente forma de realización, en una lista Playlist, una ruta de reproducción que se realiza de un conjunto matricial de uno o más PlayItems (PlayItems continuos) se refiere como una ruta principal (Main Path). En una lista Playlist, una ruta de reproducción que es paralela (concurrente) con la ruta principal y que está constituida por un conjunto matricial de una o más rutas secundarias (que pueden ser SubPlayItems discontinuos o continuos) se refiere como una ruta secundaria (Sub Path). Es decir, en el formato de aplicación del soporte de registro a cargarse en el aparato de reproducción 20 (descrito a continuación con referencia a la Figura 35), una lista Playlist contiene una ruta secundaria o rutas secundarias (Sub Path(s)) a reproducirse en asociación con (en conformidad con) la ruta principal.

La Figura 2 es un diagrama que ilustra las estructuras de una ruta principal y rutas secundarias. Una lista Playlist puede tener una ruta principal y una o más rutas secundarias. Una ruta principal está constituida por un conjunto matricial de uno o más PlayItems y una ruta secundaria está constituida por uno o más SubPlayItems.

En el caso de la realización, a modo de ejemplo, ilustrada en la Figura 2, la lista Playlist tiene una sola ruta principal, constituida por un conjunto matricial de tres PlayItems y tres rutas secundarias. Los PlayItems incluidos en la ruta principal son IDs (identificaciones) dadas de forma secuencial a partir de la primera Playlist. Más concretamente, la ruta principal está constituida por PlayItems con PlayItem_id=0, PlayItem_id=1 y PlayItem_id=2. Las rutas secundarias se proporcionan también con los identificadores IDs correspondientes, a saber, Subpath_id=0, Subpath_id=1 y Subpath_id=2, de forma secuencial desde la primera ruta secundaria. La ruta secundaria con Subpath_id=0 contiene un SubPlayItem, la ruta secundaria con SubPath_id=1 contiene dos SubPlayItems y la ruta secundaria con Subpath_id=2 contiene un SubPlayItem.

Un flujo que se refiere por el SubPlayItem contenido en la ruta secundaria con Subpath_id=0 se supone que es, a

modo de ejemplo, una señal de audio doblada en japonés para una película cinematográfica y se puede reproducir en lugar de un flujo de audio de un fichero de flujo AV referida por la ruta principal. Un flujo que se refiere por un SubPlayItem contenido en la ruta secundaria con Subpath_id=1 se supone que es, a modo de ejemplo, un corte por un director de película cinematográfica y puede ser un comentario del director que está contenido en una parte predeterminada del fichero de flujo AV referido por la ruta principal.

Un fichero de flujo AV de Clip que se refiere por un PlayItem contiene al menos datos de flujo de vídeo (datos de imagen principal). El fichero de flujo AV de Clip puede contener, o no, uno o más flujos de audio a reproducirse al mismo tiempo que (en sincronización con) el flujo de vídeo (datos de imagen principales) contenidos en el Figura de flujo AV de Clip. Además, el fichero de flujo AV de Clip puede contener, o no, uno o más flujos de captura de mapa de bits a reproducirse al mismo que el flujo de vídeo contenido en el fichero de flujo AV de Clip. Además, el fichero de flujo AV de Clip puede contener, o no, uno o más flujos de gráfico interactivos a reproducirse al mismo tiempo que el flujo de vídeo contenido en un fichero de flujo AV. El flujo de vídeo contenido en el fichero de flujo AV de Clip se multiplexa con el flujo de audio, el fichero de flujo de subtítulos de mapa de bits o el flujo de gráficos interactivos que han de reproducirse al mismo tiempo que el flujo de vídeo. Es decir, los datos de flujo de vídeo, cero o más flujos de audio, cero o más elementos de datos de flujo de subtítulos de mapa de bits y cero o más elementos de datos de flujos de gráficos interactivos que han de reproducirse en conjunción los datos de flujo de vídeo son objeto de multiplexación con un fichero de flujo AV de Clip referido por un PlayItem.

Es decir, un fichero de flujo AV de Clip que se refiere por PlayItem contiene múltiples tipos de flujos, tales como flujo de vídeo, flujo de audio, fichero de flujo de subtítulos de texto de mapeado de bits y flujo de gráficos interactivos.

Un solo SubPlayItem se refiere a datos de flujo de audio o datos de subtítulos de un flujo (otro flujo) que sea diferente de un fichero de flujo AV de Clip referido por un PlayItem.

Cuando una lista Playlist que tiene solamente una ruta principal ha de reproducirse, la operación de conmutación de audio o de conmutación de subtítulos, realizada por el usuario, permite que se seleccione la señal de audio o subtítulos a partir de solamente un flujo de audio y de un flujo de imagen secundario que son objeto de multiplexación con un Clip referido por la ruta principal. Por el contrario, cuando ha de reproducirse una Playlist que tenga una ruta principal y una ruta secundaria, un flujo de audio y un flujo de imagen secundario de un Clip, que se refieren por SubPlayItems, puede referirse también, además, a un flujo de audio y un flujo de imagen secundario que son objeto de multiplexación con un flujo AV de Clip referido por la ruta principal.

De este modo, puesto que la configuración en donde múltiples rutas secundarias están contenidas en una sola Playlist y SubPlayItems se refieren por las rutas secundarias correspondientes que se utiliza, es posible conseguir un flujo AV con una amplia posibilidad de expansión y libertad elevada. Es decir, en esta configuración, un SubPlayItem puede añadirse posteriormente a un flujo AV de Clip referido por una ruta principal.

La Figura 3 es un diagrama que ilustra, a modo de ejemplo, una ruta principal y una ruta secundaria. En la Figura 3, una ruta de reproducción de audio a reproducirse al mismo tiempo que (esto es, en sincronización con) la ruta principal se expresa utilizando una ruta secundaria.

Una Playlist, ilustrada en la Figura 3, contiene un PlayItem con PlayItem_id=0 como la ruta principal y un SubPlayItem como la ruta secundaria. Un PlayItem() con PlayItem_id=0 se refiere a un flujo AV principal ilustrado en la Figura 3. El SubPlayItem() contiene los datos siguientes. En primer lugar, el SubPlayItem() contiene Clip_Information_file_name para especificar un Clip que se refiere por la Sub Path (ruta secundaria) en la Playlist. En el caso ilustrado, a modo de ejemplo, en la Figura 3, el SubPlayItem se refiere a un flujo de audio auxiliar (un flujo de audio) con SubClip_entry_id=0. El SubPlayItem() contiene SubPlayItem_IN_time y SubPlayItem_OUT_time para especificar un segmento de reproducción de ruta secundaria en un Clip (el flujo de audio preliminar en este caso). El SubPlayItem() contiene, además, sync_PlayItem_id y sync_start_PTS_of_PlayItem para especificar el momento en el que ha de iniciarse en la cronología la reproducción de la ruta secundaria (playback timeline) de la ruta principal. En el caso de la realización, a modo de ejemplo, ilustrado en la Figura 3, un sync_PlayItem_id es 0 y sync_start_PTS_of_PlayItem es t1. Esto hace posible especificar el momento en el que ha de iniciarse la reproducción de la ruta secundaria en la cronología (playback timeline) de PlayItem_id=0 de la ruta principal. Es decir, en el caso de la ilustración, a modo de ejemplo, de la Figura 3, se muestra que el momento de iniciación de la reproducción t1 de la ruta principal y el momento de iniciación t1 de la ruta secundaria son los mismos.

En este caso, el flujo AV de Clip de audio, referido por la ruta secundaria, no debe contener puntos discontinuos de STC (puntos discontinuos basados en el tiempo del sistema). Una señal de reloj muestra de audio de Clip, utilizada para la ruta secundaria, se bloquea para una señal de reloj muestra de audio para la ruta principal.

Dicho de otro modo, el SubPlayItem() contiene información que especifica un Clip referido por la ruta secundaria, información que especifica un segmento de reproducción de la ruta secundaria e información que especifica el momento en que ha de iniciarse la reproducción de la ruta secundaria en la cronología de la ruta principal. El flujo AV de Clip utilizado por la ruta secundaria no contiene un STC. De este modo, sobre la base de la información contenida en el SubPlayItem() (esto es, la información que especifica el Clip referido por la ruta secundaria, la

información que especifica el segmento de reproducción de la ruta secundaria y la información que especifica el momento en que ha de iniciarse la reproducción de la ruta secundaria en la cronología de la ruta principal), la reproducción puede realizarse haciendo referencia al flujo de audio de un flujo AV de Clip que es diferente del flujo AV de Clip (flujo AV principal) referido por la ruta principal.

De esta manera, el PlayItem y el SubPlayItem controlan los ficheros de flujo AV de Clip correspondientes y en este caso, el fichero de flujo AV de Clip (el flujo AV principal) controlado por el PlayItem y el fichero de flujo AV de Clip controlado por el SubPlayItem son diferentes entre sí.

Como en la realización, a modo de ejemplo, ilustrada en la Figura 3, una ruta de reproducción de flujo de subtítulos, a reproducirse al mismo tiempo que la ruta principal puede expresarse también utilizando una ruta secundaria.

La Figura 4 es un diagrama que ilustra, a modo de otro ejemplo, una ruta principal y una ruta secundaria. En la Figura 4, una ruta de reproducción de vídeo y audio a reproducirse en el mismo tiempo que (esto es, en sincronización AV con) la ruta principal se expresa utilizando una ruta secundaria.

Un fichero de flujo que se refiere por PlayItem-1 de la ruta principal corresponde a las primeras mitades de un flujo de vídeo primario (primary) y un flujo de audio primario de Clip-0 y las primeras mitades de un flujo de PG (gráficos de presentación) y un flujo IG (gráficos interactivos) y un fichero de flujo AV principal que se refiere por PlayItem-2 corresponde a las últimas mitades del flujo de vídeo primario (primary) y el flujo de audio primario de Clip-0 y las últimas mitades del flujo PG y el flujo IG.

Un fichero de flujo que se refiere por un SubPlayItem de la ruta secundaria corresponde a un (2º) flujo de vídeo secundario y a un flujo de audio secundario de Clip-1.

Dicha configuración se utiliza, a modo de ejemplo, cuando el flujo de vídeo primario (primary), el flujo de audio primario, el flujo PG y el flujo IG, que se refieren por la ruta principal, constituyen un contenido de película única (contenido AV), el flujo de vídeo secundario y el flujo de audio secundario que se refieren por la ruta secundaria constituyen una pista adicional de bono que contiene un comentario del director sobre la película y los flujos de vídeo y de audio referidos por la ruta secundaria se mezclan con (superponen sobre) los flujos de vídeo y de audio referidos por la ruta principal.

En algunos casos un flujo secundario no está sincronizado con un flujo primario (esto es, la lista de reproducción Playlist no está establecida de modo que se reproduzcan siempre simultáneamente). Incluso en tal caso, se utiliza también la configuración anteriormente descrita, cuando el usuario introduce una instrucción para combinar y visualizar una pista adicional como bono para el aparato de reproducción (player) mientras que se visualiza una película y luego, las partes de vídeo y de audio del flujo primario referidas por una ruta principal y las partes de vídeo y audio del flujo secundario referidas por una ruta secundaria se mezclan y reproducen.

En la Figura 4, la ruta principal tiene dos PlayItems con PlayItem_ids=0 y 1 y la ruta secundaria (Subpath_id=0) tiene un SubPlayItem. El SubPlayItem (descrito a continuación con referencia a la Figura 21) a solicitarse por la ruta secundaria (que se describe más adelante con referencia a la Figura 19) contiene SubPlayItem_IN_time y SubPlayItem_out_time para especificar un segmento de reproducción de la ruta secundaria.

Cuando las partes de vídeo secundario y de audio secundario se superponen en las partes de vídeo primario y de audio primario descritas con referencia a la Figura 4 y se reproducen y son objeto de salida, las señales de audio se combinan y la salida y la señal de vídeo secundaria de los vídeos se superponen y visualizan como una pantalla de presentación visual secundaria 2 que tiene una magnitud predeterminada en una posición predeterminada en una pantalla de presentación visual principal 1 en la que se visualiza la señal de vídeo primaria, según se ilustra en la Figura 5.

En la Figura 4, lo que se refiere por la ruta principal es el flujo primario y lo que se refiere por la ruta secundaria es el flujo secundario. Sin embargo, cada uno del flujo primario y del flujo secundario se puede referir por la ruta principal o la ruta secundaria. A modo de ejemplo, a modo de ejemplo, el flujo primario y el flujo secundario pueden referirse por la ruta principal y ambos flujo primario y flujo secundario pueden referirse por la ruta secundaria.

De este modo, dicha configuración puede utilizarse, según se ilustra en la Figura 5, cuando se visualiza un flujo de vídeo primario en una pantalla principal y se visualiza un flujo secundario en una pantalla auxiliar dentro de la pantalla principal, es decir, cuando se realiza la visualización así denominada PinP (imagen – en-imagen: Picture-in-Picture).

La Figura 6 es un diagrama que ilustra, a modo de ejemplo, el sistema de ficheros de ficheros de datos que pueden reproducirse por el aparato de reproducción 20 (descrito a continuación con referencia a la Figura 35). La Figura 6 ilustra, a modo de ejemplo, un caso en el que ficheros de datos que pueden reproducirse por el aparato de reproducción 20 se suministran a través de un soporte de registro tal como un disco óptico. Este sistema de ficheros tiene una estructura de directorios. Es preferible que los ficheros de datos que se reproducen por el aparato de

reproducción 20 tengan un sistema de ficheros según se ilustra en la Figura 6, a modo de ejemplo, cuando los ficheros de datos que pueden reproducirse por el aparato de reproducción 20 se registran en un soporte de registro distinto de un disco óptico y se suministran desde dicho soporte o se registran en un medio de registro interno y se reproducen.

En este sistema de ficheros, un directorio al que se le proporciona un nombre "BDMV" se proporciona debajo de una "raíz" y contiene un fichero al que se proporciona un nombre "Index.bdmv" y un fichero al que se proporciona el nombre "NavigationObject.bdmv". Estos ficheros se referirán, a continuación, como un "fichero de índice" y un "fichero objeto de navegación", respectivamente, según sea necesario. Cada fichero se representará utilizando una forma en la que se añade un "fichero" a un "nombre de fichero" o cada directorio se representará utilizando una forma en la que se añade un "directorio" a un "nombre de directorio", cuando sea adecuado.

El fichero de índice es un fichero que en el que se describe la tabla de índices anteriormente citada y contiene información respecto a un menú para reproducir ficheros de datos que pueden reproducirse por el aparato de reproducción 20. El aparato de reproducción 20 hace que un dispositivo de presentación visualice una pantalla de menú de reproducción, sobre la base del fichero de índice. Realizaciones, a modo de ejemplo, de la pantalla de menú de reproducción incluyen una opción para visualizar la totalidad del contenido en los ficheros de datos que pueden reproducirse por el aparato de reproducción 20, una opción para reproducir solamente una sección concreta, una opción para una reproducción repetida y una opción para visualizar un menú inicial. Los objetos de navegación que han de realizarse cuando se selecciona cada opción, pueden incluirse en la tabla de índices en el fichero de índices. Cuando el usuario selecciona una de las opciones en la pantalla de menú de reproducción, el aparato de reproducción 20 ejecuta órdenes para los objetos de navegación establecidos en la tabla de índices en el fichero de índices.

El fichero objeto de navegación contiene los objetos de navegación. Los objetos de navegación incluyen órdenes para controlar la reproducción de una lista PlayList contenida en los ficheros de datos que pueden reproducirse por el aparato de reproducción 20. A modo de ejemplo, la selección y ejecución de uno de los objetos de navegación, contenidos en el sistema de ficheros, permite la reproducción del contenido.

El directorio BDMV tiene, además, un directorio al que se da un nombre "BACKUP" (esto es, un directorio de reserva BACKUP), un directorio al que se le da un nombre "PLAYLIST" (esto es, un directorio de lista de reproducción PLAYLIST), un directorio al que se le da un nombre "CLIPINF" (esto es, un directorio de información de Clip CLIPINF), un directorio al que se le da un nombre "STREAM" (esto es, un directorio de flujo STREAM) y un directorio al que se le proporciona un nombre "AUXDATA" (esto es, un directorio de AUXDATA).

Los ficheros y datos para la copia de seguridad de ficheros y datos que puedan reproducirse por el aparato de reproducción 20 se registran en el directorio BACKUP.

Los ficheros de PlayList se memorizan en el directorio PLAYLIST. Según se ilustra en la Figura, a cada fichero de PlayList se le proporciona un nombre que tiene un nombre de fichero de 5 dígitos al que se añade una extensión ".mpls". El fichero de PlayList se describe a continuación haciendo referencia a la Figura 7.

Los ficheros de información de Clip se memorizan en el directorio CLIPINF. Según se ilustra en la figura, a cada fichero de información de Clip se le proporciona un nombre de fichero de número de 5 dígitos al que se añade una extensión ".clpi".

Los ficheros de flujo AV de Clip y los ficheros de flujo secundario se memorizan en el directorio STREAM. Según se indica en la figura, a cada fichero de flujo se le proporciona un nombre que tiene un nombre de fichero de número de 5 dígitos al que se añade una extensión ".m2ts".

El directorio AUXDATA no contiene ningún fichero de flujo AV de Clip y fichero de flujo secundario y contiene ficheros, tales como los datos que se refieren por los ficheros de flujo AV de Clip y ficheros de flujo secundario y datos utilizados por separado, procedentes de los ficheros de flujo AV de Clip y ficheros de flujos secundarios. En la ilustración, a modo de ejemplo, de la Figura 6, el directorio AUXDATA contiene un fichero de fuente-subtítulos al que se proporciona un nombre "11111.otf" y datos de sonido de efecto acústico y por lo que se le da un nombre de "sound.bdmv".

Cuando los ficheros de datos que pueden reproducirse por el aparato de reproducción 20 se distribuyen a través de un disco óptico, se registran author_id y disc_id en receptáculos físicos como datos eléctricos seguros que no puedan sobrescribirse por los usuarios y así sucesivamente. Author_id es un identificador que se asigna a, a modo de ejemplo, un productor de contenido, tal como una empresa de producción o una empresa de distribución de películas o un autor de título correspondiente para identificar el autor del título, que es el proveedor del soporte de registro. Disc_id es un identificador que se asigna para identificar el tipo de disco óptico producido por el autor del título indizado por author_id.

Es también preferible que se proporcione una estructura de directorio similar a la ilustrada en la Figura 6, de módulo

que author_ID y un identificador ID correspondiente a author_ID sean asignados con el fin de hacerles distinguibles entre sí, cuando los ficheros de datos que pueden reproducirse por el aparato de reproducción 20 se registran en un soporte de registro extraíble que no sea un disco óptico, se registran en un soporte de registro en el interior de lapa de reproducción o se memorizan en una unidad de memorización interna. Sin importar si, o no, se asignan author_ID y un ID correspondiente a author_ID el fichero de datos que puede reproducirse por el aparato de reproducción 20 contiene un fichero al que se proporciona un nombre "Index.bdmv" y un fichero al que se proporciona un nombre "NavigationObject.bdmv", como en el caso descrito con referencia a la Figura 6. Además, el fichero de datos contiene un grupo o grupos de ficheros necesarios de un grupo de ficheros dado al que se le proporciona el nombre "BACKUP", un grupo de ficheros al que se le proporciona el nombre "PLAYLIST", un grupo de ficheros al que se le proporciona el nombre "CLIPINF", un grupo de ficheros al que se le proporciona el nombre "STREAM" y un grupo de ficheros al que se proporciona el nombre "AUXDATA", cuando sea necesario.

A continuación, se proporcionará una descripción de una estructura de datos (sintaxis) de cada fichero descrito con referencia a la Figura 6. Esta estructura de datos realiza estructuras específicas de las rutas principales y secundarias anteriores con referencia a las Figuras 2 a 4.

La Figura 7 ilustra la estructura de datos de un fichero PlayList que tiene la extensión ".mpls", estando el fichero PlayList memorizado en el directorio PLAYLIST.

Un fichero type_indicator proporciona información que indica el tipo de este fichero y debe codificarse en "MPLS" en conformidad con la norma ISO 646.

Un campo de version_number indica cuatro caracteres que muestra el número de versión de xxxx.mpls y version_number debe codificarse en conformidad con la norma ISO "0089".

La dirección de inicio PlayList_start_address indica la primera dirección de una PlayList() en términos del número relativo de bytes desde el principio del fichero PlayList.

La dirección PlayListMark_start_address indica la primera dirección de un PlayListMark() en términos del número relativo de bytes desde el inicio del fichero PlayList.

La dirección PlayListExtensionData_start_address indica la primera dirección de PlayListExtensionData() en términos del número relativo de bytes desde el inicio del fichero PlayList.

ApplInfoPlayList() contiene parámetros con respecto al control de reproducción de PlayList.

PlayList() contiene parámetros con respecto a la ruta principal y a las rutas secundarias en la PlayList. Detalles de PlayList() se describirán a continuación haciendo referencia a la Figura 18.

PlayListMark() contiene información de marca de PlayList. Datos privados pueden insertarse también en PlayListExtensionData().

La Figura 8 ilustra la sintaxis de PlayListExtensionData().

El parámetro Length indica el número de bytes de PlayListExtensionData() inmediatamente después de este campo de longitud al final de PlayListExtensionData().

La dirección Data_blocks_start_address indica la primera dirección del primer bloque de datos data_block() en términos del número relativo de bytes desde el inicio de PlayListExtensionData().

Number_of_PL_ext_data_entries indica el número de entradas de PlayListExtensionData contenidas en PlayListExtensionData(). Dos o más mismos identificadores IDs no pueden existir en PlayListExtensionData().

ID1 e ID2 proporcionan información (tal como identificadores) que pueden identificar los tipos de información indicados en el bloque de datos data_block de PlayListExtensionData().

La dirección PL_ext_data_start_address indica un número de bloque de datos data_block en el que se inicia PlayListExtensionData. Los primeros datos de PlayListExtensionData deben alinearse con el principio de data_block.

PL_ext_data_length indica el tamaño de los PlayListExtensionData en bytes.

Data_block es un área en el que se memorizan los datos de PlayListExtensionData. Todos los bloques de datos data_blocks en PlayListExtensionData() deben tener el mismo tamaño.

El bloque de datos Data_block puede proporcionar información que indica la posición de presentación visual y el tamaño de la pantalla de presentación visual secundaria 2 en la que se visualiza la señal de vídeo secundaria, la

pantalla de presentación visual secundaria 2 que se describe con referencia a la Figura 5. El bloque de datos `data_block` puede establecer los metadatos que indican el establecimiento de una presentación visual imagen-en-imagen (Picture-in-Picture).

- 5 La Figura 9 ilustra, a modo de primer ejemplo, la sintaxis de `pip_metadata` (metadatos que indican el establecimiento de la visualización denominada Picture-in-Picture, que significa 'superposición de imagen en imagen') que se establece en el bloque de datos `data_block`.

10 El parámetro `Length` indica el número de bytes de `pip_metadata( )` inmediatamente después de este campo de longitud al final de `pip_metadata( )`.

Un campo `pip_metadata_type` indica el tipo de `pip_metadata`. Según se ilustra en la Figura 10, para `pip_metadata_type=0x01`, un flujo de vídeo secundario, para el que este `pip_metadata( )` especifica la presentación visual *Picture-in-Picture* que se reproduce es un flujo de vídeo de tipo síncrono, es decir, se reproduce en sincronización con la cronología (playback timeline) de un `PlayItem` de una ruta principal. De este modo, el flujo de vídeo secundario se reproduce siempre y se proporciona en sincronización con el flujo primario. Para `pip_metadata_type=0x02`, un flujo de vídeo secundario, para el que este `pip_metadata( )` especifica la visualización de superposición imagen en imagen, *Picture-in-Picture*, es un flujo de vídeo de tipo asíncrono, es decir, se reproduce en sincronización con la cronología de un `SubPlayItem` de una ruta secundaria pero no requiere la sincronización con la cronología de un `PlayItem` de una ruta principal. De este modo, solamente cuando una entrada de operación de usuario emite una instrucción para visualizar el flujo secundario, el flujo secundario se reproduce y es objeto de salida.

25 Es decir, la "sincronización con `SubPlayItem`" expresada por `pip_metadata_type=0x02` significa que el flujo de vídeo secundario para el que este `pip_metadata( )` especifica la presentación visual *Picture-in-Picture* es asíncrono con la cronología de la ruta principal, pero es síncrono con la cronología de la ruta secundaria. Es decir, `pip_metadata( )` para `pip_metadata_type=0x02` indica la posición de visualización y el tamaño de la pantalla de presentación visual secundaria 2 durante la reproducción del `SubPlayItem`, sobre la base del momento en que se inicia la presentación visual de la señal de vídeo secundaria en respuesta a una acción del usuario (entrada de operación).

30 `Ref_to_PlayItem/SubPath_id` indica el valor de `PlayItem_id` de un `PlayItem` que utiliza `pip_metadata` o el valor de `SubPath_id` de un `SubPlayItem` que utiliza `pip_metadata`. Es decir, cuando el flujo de vídeo secundario para el que `pip_metadata( )` especifica una presentación visual *Picture-in-Picture* es síncrona con la ruta principal (esto es, cuando `pip_metadata_type` es `0x01`), se utiliza `ref_to_PlayItem` y cuando el flujo de vídeo secundario para el que este `pip_metadata( )` especifica la presentación visual *Picture-in-Picture* es asíncrono con la ruta secundaria (esto es, cuando `pip_metadata_type` es `0x02`), se utiliza `ref_to_SubPath_id`.

35 `Pip_metadata_time_stamp` representa una marca de tiempo para un `PlayItem` que utiliza `pip_metadata` y la señal de vídeo del flujo de vídeo secundario se visualiza en la pantalla de presentación visual secundaria 2 en la temporización indicada por `pip_metadata_time_stamp` y en la posición de presentación visual y en el tamaño que se especifican por `pip_metadata( )`.

45 Es decir, `pip_metadata_time_stamp` necesita especificar el tiempo de presentación entre `In_time` y `Out_time` del `PlayItem` objeto de referencia por `ref_to_PlayItem/SubPath_id`. Cuando el flujo de vídeo secundario para el que este `pip_metadata( )` especifica la presentación visual *Picture-in-Picture* es síncrona con la ruta secundaria (esto es, cuando `pip_metadata_type` es `0x02`), `ref_to_PlayItem` se utiliza, la señal de vídeo del flujo de vídeo secundario se visualiza en la pantalla de presentación visual secundaria 2 en una temporización que está incluida en la cronología del `SubPlayItem` y que se indica por `pip_metadata_time_stamp` y en la posición de la presentación visual y en el tamaño que se especifican por `pip_metadata( )`.

50 `Pip_entry_video_PID` indica el valor de un PID de la señal de vídeo secundaria utilizada para la visualización *Picture-in-Picture*.

55 `Pip_horizontal_position` indica, en la trama de vídeo primaria (la pantalla de presentación visual principal 1 ilustrada en la Figura 5), una coordenada X en la esquina superior izquierda de la pantalla de presentación visual secundaria 2 en la que se visualiza la señal de vídeo secundaria. `Pip_vertical_position` indica, en la trama de vídeo primaria (la pantalla de presentación visual principal 1 ilustrada en la Figura 5), una coordenada Y en la esquina superior izquierda de la pantalla de presentación visual secundaria 2 en la que se visualiza la señal de vídeo secundaria.

60 No es preciso indicar que un método distinto al que especifica las coordenadas X e Y en la esquina superior izquierda de la pantalla de presentación visual secundaria 2 utilizando `pip_horizontal_position` y `pip_vertical_position` pueden utilizarse también para especificar la posición de presentación visual de la pantalla de presentación visual secundaria 2 en la que se visualiza la señal de vídeo secundaria. Realizaciones, a modo de ejemplo, de dicho método incluyen la especificación de las coordenadas X e Y de la esquina inferior derecha de la pantalla de presentación visual secundaria 2 y la especificación de cualquiera de una pluralidad de posiciones de visualización predeterminadas.

Un campo `pip_scale` proporciona información que indica el tamaño de la pantalla de presentación visual secundaria 2 en la que se visualiza la señal de vídeo secundaria. A modo de ejemplo, `pip_scale` indica la relación de tamaños de la señal de vídeo secundaria a la señal de vídeo primaria, el ratio de escala del tamaño de vídeo secundario a su tamaño original o información que indica cuántos pixels se incluyen en la pantalla de presentación visual secundaria 2 en cada una de la dirección vertical y de la dirección horizontal.

La Figura 10 ilustra el valor y el significado de `pip_metadata_type` ilustrado en la Figura 9.

`Pip_metadata_type=0x00` es un valor que indica que el significado no se establece para futura expansión. Para `pip_metadata_type=0x01`, este `pip_metadata( )` es metadatos de tipo síncrono, es decir, es síncrono con la cronología del `PlayItem` de la ruta principal; por lo tanto, el flujo secundario se reproduce y proporciona siempre en sincronización con el flujo primario. Para `pip_metadata_type=0x02`, este `pip_metadata( )` es metadatos de tipo asíncrono, es decir, es asíncrono con la cronología del `SubPlayItem` de la ruta secundaria. Por lo tanto, la señal de vídeo del flujo de vídeo secundario se reproduce y se proporciona en la pantalla de presentación visual secundaria 2, solamente cuando una entrada de operación del usuario emite una instrucción para visualizar el flujo secundario. Exceptuado para los tres valores antes citados de `pip_metadata_type`, el significado no se establece para una futura extensión.

La Figura 11 ilustra una segunda realización, a modo de ejemplo, de la sintaxis de `pip_metadata` establecida en el bloque de datos `data_block`. La segunda realización, a modo de ejemplo, es diferente del caso ilustrado en la Figura 9. Las descripciones para campos de datos que tengan los mismos nombres y las mismas definiciones que las ilustradas en la Figura 9 se omiten a continuación, cuando sea adecuado.

Un campo `number_of_pip_entries` indica el número de aplicaciones de picture-in-picture.

El indicador `Synchronous_PIP_metadata_flag` indica con qué ruta se sincroniza una aplicación de picture-in-picture (esto es, un flujo de vídeo secundario). Es decir, cuando el valor del indicador `Synchronous_PIP_metadata_flag` es '0', esta aplicación de *Picture-in-Picture* se sincroniza con una ruta secundaria y el indicador `ref_to_SubPath_id` se registra para `ref_to_PlayItem/SubPath_id`. Por el contrario, cuando el valor del indicador `Synchronous_PIP_metadata_flag` es '1', esta aplicación de picture-in-picture se sincroniza con una ruta principal y `ref_to_PlayItem` se registra para `ref_to_PlayItem/SubPath_id`.

Es decir, el indicador `Synchronous_PIP_metadata_flag` ilustrado en la Figura 11, es información para definir un contenido que se prácticamente que el mismo que el de `pip_metadata_type` (Figura 10) en `pip_metadata` descrito con referencia a la Figura 9.

Un campo de entrada `Number_of_pip_metadata_entries` indica el número total de elementos de metadatos PinP para el flujo de vídeo de `ref_to_PlayItem/ref_to_SubPath_id`.

`Pip_metadata_type` indica el tipo de metadatos de picture-in-picture y puede indicar, a modo de ejemplo, no solamente la información que indica la posición y el tamaño de la pantalla de presentación visual secundaria 2 en la que se visualiza el flujo de vídeo secundario sino también, a modo de ejemplo, información respecto a la designación de color y la rotación de la señal de vídeo visualizada en relación con la señal de vídeo original del flujo de vídeo secundario.

Es decir, `pip_metadata_type` que se ilustra en la Figura 11, es información que tiene una definición que es diferente de la que tiene `pip_metadata_type` en `pip_metadata` descrito con referencia a la Figura 9.

La Figura 12 ilustra, a modo de tercer ejemplo, la sintaxis de `pip_metadata` establecida en el bloque de datos `data_block`. Esta tercera realización, a modo de ejemplo, es diferente de los casos ilustrados en las Figuras 9 y 11.

Un campo de longitud proporciona información que indica el número de bytes de `pip_metadata( )` inmediatamente después de este campo de longitud al final de `pip_metadata( )`.

Un campo de entrada `number_of_metadata_block_entries` proporciona información que indica el número de entradas del bloque de metadatos que se contiene en `pip_metadata( )`.

`Metadata_block_header(k)( )` proporciona información que incluye información de cabecera relacionada con un bloque de metadatos.

Un campo de `ref_to_PlayItem_id[k]` proporciona información que indica el valor de `PlayItem_id` de un `PlayItem` que utiliza `pip_metadata`.

Un campo `ref_to_secondary_video_stream_id[k]` proporciona información que indica el valor del identificador ID de la señal de vídeo secundaria que utiliza `pip_metadata`, esto es, el valor de `secondary_video_stream_id` definido en la tabla `STN_table` (descrita a continuación haciendo referencia a la Figura 22) en un `PlayItem` referido por

ref_to_PlayItem_id.

Un campo `pip_timeline_type[k]` proporciona información que indica el tipo de cronología referida por `pip_metadata_time_stamp`. El valor y el significado de `pip_timeline_type[k]` se describen a continuación.

Cuando un indicador para `is_luma_key` se establece a '1', la función *luma keying* se aplica al flujo de vídeo secundario en conformidad con los valores de `lower_limit_luma_key` y de `upper_limit_luma_key`.

El campo `lower_limit_luma_key` proporciona información que indica el límite inferior del valor de luminancia de vídeo secundario para *luma keying*.

El campo `upper_limit_luma_key` proporciona información que indica el límite superior del valor de luminancia de vídeo secundario para *luma keying*.

El término "luma keying" significa aquí la supervisión y combinación de una imagen, a partir de la que se fijan partes indeseadas utilizando diferencias de componentes en brillo (valor de luminancia), con vídeo. Es decir, cuando el indicador para `is_luma_key` se establece a '1', una imagen que tiene un valor de luminancia especificado por la variación del límite inferior del valor de luminancia al límite superior del valor de luminancia, que se definen por `lower_limit_luma_key` y `upper_limit_luma_key`, se establece para ser transparente. De este modo, la señal de vídeo secundaria a partir de la que se extraen imágenes en el margen especificado por la información del brillo se superpone sobre y se combina con la señal de vídeo primaria.

Esta disposición permite que se visualice la señal de vídeo secundaria mínima, con lo que se hace posible impedir que la señal de vídeo primaria se haga difícil de ver. Además, es posible combinar, de forma flexible, la señal de vídeo secundaria y la señal de vídeo primaria.

En esta realización, a modo de ejemplo, se ha proporcionado la descripción de un caso en el que una imagen puede establecerse transparente a través del uso de dos valores umbrales, esto es, `lower_limit_luma_key` y `upper_limit_luma_key`. La presente invención, sin embargo, no está limitada a esta disposición. A modo de ejemplo, uno de los umbrales puede utilizarse para establecer solamente una parte que tenga un valor de luminancia menor o igual a `upper_limit_luma_key` o un valor de luminancia mayor o igual a `upper_limit_luma_key` para ser transparente.

Un campo de `metadata_block_data_start_address[k]` proporciona información que indica la primera dirección de los primeros `metadata_block_data[k]` () en términos del número relativo de bytes desde el primer byte de `pip_metadata` (). Además, la dirección de `metadata_block_data_start_address[k]` introducida en `pip_metadata` () debe registrarse con referencia a un valor de dirección.

Una palabra de relleno se inserta en `padding_word`, en función del valor de `metadata_block_data_start_address[k]`.

`Metadata_block_data(k)` () proporciona información que incluye información de datos de un bloque de metadatos.

Un campo `number_pip_metadata_entries[k]` proporciona información que indica el número total de elementos de `pip_metadata` en `metadata_block_data`.

Un campo `pip_metadata_time_stamp [i]` proporciona información incluyendo, a modo de ejemplo, una marca de tiempo de 45 kHz que indica el tiempo en el que se utiliza `pip_metadata`. Un intervalo mínimo de dos `pip_metadata_time_stamps` consecutivos es un segundo. El significado de `pip_metadata_time_stamp` correspondiente al valor de `pip_timeline_type` se describe a continuación.

Un campo de `pip_composition_metadata` () proporciona información respecto al tamaño y posición de la trama de vídeo-primario.

`Pip_horizontal_position[i]` indica, en la trama de vídeo primaria (la pantalla de presentación visual principal 1 ilustrada en la Figura 5), una coordenada X en la esquina superior izquierda de la pantalla de presentación visual secundaria 2 en la que se visualiza la señal de vídeo secundaria. `Pip_vertical_position[i]` indica, en la trama de vídeo primario (la pantalla de presentación visual principal 1 ilustra en la Figura 5) una coordenada Y en la esquina superior izquierda de la pantalla de presentación visual secundaria 2 en la que se visualiza la señal de vídeo secundaria.

Según se ilustra en la Figura 13, un campo `pip_scale[i]` proporciona información para establecer la relación de escalas del tamaño de la señal de vídeo secundaria a su tamaño de imagen original a 1, 1/2, 1/4, 1.5 o una pantalla completa (con lo que se visualiza la señal de vídeo secundaria en la pantalla de presentación visual principal completa 1 ilustrada en la Figura 5).

`Pip_composition_metadata` () en `pip_metadata_time_stamp[i]` es válida en el intervalo desde `pip_metadata_time_stamp[i]` a `pip_metadata_time_stamp [i+1]`. Sin embargo, el intervalo en el que el último `pip_composition_metadata` es válido es desde el último `pip_metadata_time_stamp` al momento de finalización de

presentación, presentation-end time, de ruta secundaria por ref_to_secondary_video_stream_id[k].

A continuación, el valor y el significado de pip_timeline_type se describirán con referencia a la Figura 14 y el significado de pip_metadata_time_stamp correspondiente al valor de pip_timeline_type se describirá con referencia a las Figuras 15 a 17.

Pip_timeline_type=0 es un valor que indica que el significado no se establece para una futura expansión.

Pip_timeline_type=1 significa que la ruta de presentación de picture-in-picture es una ruta de tipo síncrono. En este caso, el tipo de ruta secundaria especificado por ref_to_secondary_video_stream_id debe ser 5 o 7 (detalles del tipo de ruta secundaria se describen a continuación haciendo referencia a la Figura 20).

En este caso, según se ilustra en la Figura 15, pip_metadata_time_stamp se refiere a la cronología de un PlayItem especificado por ref_to_PlayItem_id e indica el segmento de reproducción del SubPlayItem asociado. Es decir, el segmento de reproducción de SubPlayItem se proyecta sobre la cronología de un PlayItem referido por ref_to_PlayItem_id[k]. Pip_metadata_time_stamp[0] indica el inicio del segmento de reproducción de SubPlayItem [0] en la ruta secundaria.

Pip_timeline_type=2 significa que la ruta de presentación de picture-in-picture (imagen en imagen) es una ruta de tipo asíncrono. En este caso, el tipo de ruta secundaria especificado por ref_to_secondary_video_stream_id debe ser 6. En este caso, la ruta secundaria que tiene SubPath_type=6 contiene solamente un SubPlayItem.

En este caso, según se ilustra en la Figura 16, pip_metadata_time_stamp se refiere a la cronología de la ruta secundaria e indica el segmento de reproducción de un SubPlayItem de la ruta secundaria que se especifica por ref_to_secondary_video_stream_id[k]. Pip_metadata_time_stamp[0] indica el inicio del segmento de reproducción del SubPlayItem.

Pip_timeline_type=3 significa que la ruta de presentación de *picture-in-picture* es una ruta de tipo asíncrono. En este caso, el tipo de ruta secundaria, especificado por ref_to_secondary_video_stream_id debe ser 6.

En este caso, según se ilustra en la Figura 17, pip_metadata_time_stamp se refiere a la cronología de un PlayItem especificado por ref_to_PlayItem_id e indica el segmento de reproducción del PlayItem especificado por ref_to_PlayItem_id[k]. Pip_metadata_time_stamp[0] indica el inicio del segmento de reproducción del PlayItem especificado por ref_to_PlayItem_id[k].

La Figura 18 ilustra la sintaxis de Playlist().

Un campo de longitud tiene un número entero sin signo de 32 bits que indican el número de bytes inmediatamente después de este campo de longitud al final de PlayItem(). Es decir, el campo de longitud indica el número de bytes desde reservado_para_futuro_uso al final de la Playlist. Un campo de reservado_para_futuro_uso de 16 bits se prepara después del campo de longitud. Number_of_PlayItems es un campo de 16 bits que indica el número de PlayItems en la Playlist. A modo de ejemplo, en el caso del ejemplo ilustrado en la Figura 2, el número de PlayItem es 3. Los valores de PlayItem_ids se asignan desde 0 en el orden en que aparecen los PlayItems() en la Playlist. A modo de ejemplo, según se ilustra en las Figuras 2 y 4, se asignan PlayItem_ids = 0, 1 y 2.

Number_of_SubPaths es un campo de 16 bits que indica el número de rutas secundarias (el número de entradas) en la Playlist. A modo de ejemplo, en el caso del ejemplo ilustrado en la Figura 2, el número de rutas secundarias es 3. Los valores de SubPath_ids se asignan desde 0 en el orden en que aparecen los SubPaths() en la Playlist. A modo de ejemplo, según se ilustra en la Figura 2, se asignan identificadores de ruta secundaria Subpath_ids 0, 1 y 2. En la posterior declaración de 'for', los PlayItems se refieren en función del número de PlayItem y las rutas secundarias se refieren en función del número de rutas secundarias.

La Figura 19 ilustra la sintaxis de SubPath().

Un campo de longitud tiene un número entero sin signo de 32 bits que indica el número de bytes inmediatamente después de este campo de longitud al final de SubPath(). Es decir, el campo de longitud indica el número de bytes desde reservado_para_futuro_uso al final de la Playlist. Después del campo de longitud, se prepara el campo de 16 bits reservado_para_futuro_uso. SubPath_type es un campo de 8 bits que indica el tipo de aplicación de ruta secundaria. SubPath_type se utiliza para indicar un tipo de ruta secundaria, a modo de ejemplo, de audio, de subtítulos de mapa de bits o de subtítulos de texto. Este SubPath_type se describe a continuación haciendo referencia a la Figura 20. Después de SubPath_type, se prepara 15 bits reservados para uso futuro. Is_repeat_SubPath es un campo de 1 bit que especifica el método de reproducción de la ruta secundaria e indica si la ruta secundaria se reproduce repetidamente o se reproduce solamente una vez durante la reproducción de la ruta principal. A modo de ejemplo, is_repeat_SubPath se utiliza cuando las temporizaciones de la reproducción de un flujo AV principal y de un Clip, que se especifica por la ruta secundaria, son diferentes entre sí (p.e., cuando se utiliza la ruta principal para una diapositiva de imagen fija y la ruta de audio de la ruta secundaria se utiliza para BGM

(música de fondo) para la ruta principal). Después de `is_repeat_SubPath`, se prepara 8 bits reservado para futuro uso. El número de `Number_of_SubPlayItems` es un campo de 8 bits que indica el número de `SubPlayItems` (el número de entradas) en una sola ruta secundaria. A modo de ejemplo, para `number_of_SubPlayItems`, el número de `SubPlayItems` con `SubPath_id=0`, ilustrado en la Figura 2, es uno y el número de `SubPlayItems` con `SubPath_id=1` es dos. En la declaración 'for' posterior, los `SubPlayItems` se refieren en función del número de `SubPlayItems`.

La Figura 20 es un diagrama que ilustra una realización, a modo de ejemplo, de `SubPath_types` (tipos de ruta secundaria). Es decir, se definen los tipos de ruta secundaria, a modo de ejemplo, según se ilustra en la Figura 20.

En los tipos de ruta secundaria, la declaración "Out-of-mux... type" indica un tipo (clase) de ruta secundaria en la que un TS (flujo de transporte), es decir, un Clip, que contiene un ES (flujo elemental) referido por la ruta secundaria es diferente de un TS (un Clip) que contiene un elemento de reproducción (uno o más ESs) referidos por la ruta principal es decir, indica un tipo (clase) de ruta secundaria en la que un ES referido por la ruta secundaria no es objeto de multiplexación con un TS (un Clip) que contiene un elemento de reproducción referido por la ruta principal. Dicho tipo se referirá, en adelante, como una "ruta no multiplexada de la ruta principal".

En la Figura 20, `SubPath_types=0` y 1 significa "reservado". `SubPath_type=2` significa la ruta de presentación de audio de las diapositivas susceptibles de exploración (ruta de presentación de audio de las diapositivas explorables). A modo de ejemplo, `SubPath_type=2` indica que, en la lista de reproducción, una ruta de presentación de audio referida por la ruta secundaria y una ruta principal referida por un elemento de reproducción no están sincronizadas entre sí.

`SubPath_type=3` significa un menú de presentación de gráficos interactivos (interactive graphics presentation menu). A modo de ejemplo, `SubPath_type=3` indica que, en la lista de reproducción, un menú de gráficos interactivos referidos por la ruta secundaria y una ruta principal referida por un elemento de reproducción no están sincronizadas entre sí.

`SubPath_type=4` significa una ruta de presentación de subtítulos de texto (text caption presentation path). A modo de ejemplo, `SubPath_type=4` indica que, en la lista de reproducción, una ruta de presentación de títulos de texto referidos por la ruta secundaria y una ruta principal referida por un elemento de reproducción están sincronizadas entre sí.

En este caso, cuando un ES referido por una ruta principal y un ES referido por una reelección no están sincronizados entre sí como en el caso de `SubPath_types=2` y 3, el tipo (clase) de la ruta secundaria se referirá, en adelante, como una "ruta de tipo asíncrono". Por el contrario, cuando un ES referido por una ruta principal y un ES referido por una ruta secundaria están sincronizados entre sí, en el caso de `SubPath_type=4`, el tipo (clase) de la ruta secundaria, se referirá, en lo sucesivo, como una "ruta de tipo asíncrono".

`SubPath_type=5` significa "fuera de multiplexación y tipo sincronizado de AV de una o más rutas de flujo elementales (ruta de audio primaria/PG/IG/ de audio secundaria)" y "fuera de multiplexación y tipo sincronizado AV de ruta de presentación de picture-in-picture que contiene una o más rutas de flujos elementales". Es decir, `SubPath_type=5` representa un tipo no multiplexado de TS de ruta principal y una ruta de tipo asíncrono y representa una o más rutas de ES (audio primaria/PG/IG/audio secundaria) o una ruta de presentación de picture-in-picture.

El término de "ruta de presentación picture-in-picture" significa en este caso, en el sistema de picture-in-picture anteriormente descrito, al menos uno de entre (un tipo de ruta secundaria) de un flujo de audio primario, una ruta de flujo secundario, un flujo de audio secundario y un flujo de subtítulos con respecto a un flujo de vídeo primario predeterminado (un flujo de vídeo referido por la ruta principal).

`SubPath_type=6` significa "fuera de multiplexación y tipo no sincronizado AV de ruta de presentación de picture-in-picture que contiene una o más rutas de flujos elementales". Es decir, `SubPath_type=6` significa un tipo no multiplexado de TS de ruta principal y una ruta de tipo asíncrono y una ruta de presentación de *picture-in-picture* (una o más rutas de ES).

`SubPath_type=7` significa "tipo en multiplexación y tipo sincronizado AV de ruta de presentación de picture-in-picture que contiene una o más rutas de flujos elementales".

En este caso, la declaración "In-mux type" indica un tipo (clase) de ruta secundaria en un caso en el que un TS (un Clip) que contiene un ES referido por la ruta secundaria, es diferente de un TS (un Clip) que contiene un elemento de reproducción (uno o más ES) referidos por la ruta principal, es decir, indica un tipo (clase) de ruta secundaria en el que un ES referido por la ruta secundaria no está multiplexado con un TS (un Clip) que contiene un elemento de reproducción referido por la ruta principal. Dicho tipo se referirá a continuación, como una "ruta multiplexada de ruta principal".

Es decir, `SubPath_type=7` significa un tipo multiplexado de TS de ruta principal y una ruta de tipo asíncrono y una

ruta de presentación de *picture-in-picture* (una o más rutas ES).

SubPath_types=8 a 255 significa "reservado".

5 La Figura 21 ilustra la sintaxis de SubPlayItem(i).

Un campo de longitud tiene un número entero sin signo de 16 bits que indica el número de bytes inmediatamente después de este campo de longitud hasta el final de SubPlayItem().

10 En la Figura 21, los campos se dividen en un caso en el que un SubPlayItem se refiere a un Clip y un caso en el que un SubPlayItem se refiere a múltiples Clips.

En primer lugar, se describirá un caso en el que un SubPlayItem se refiere a un solo Clip.

15 El SubPlayItem contiene Clip_Information_file_name[0] para especificar un Clip. El SubPlayItem contiene, además, Clip_codec_identifier[0] para especificar un sistema de codificación/decodificación de Clip, reservado_para_futuro_uso, is_multi_Clip_entries que los servidores como un indicador de si se registran, o no, múltiples Clips y ref_to_STC_id[0] que sirve como información respecto a puntos discontinuos de STC (puntos discontinuos basados en el tiempo del sistema). Cuando se establece el indicador de is_multi_Clip_entries, la
20 sintaxis para el caso en el que SubPlayItem se refiere a múltiples Clips se refiere a este respecto. El SubPlayItem contiene, además, SubPlayItem_IN_time y SubPlayItem_OUT_time para especificar un segmento de reproducción de ruta secundaria en un Clip. El SubPlayItem contiene, además, sync_PlayItem_id y sync_start_PTS_of_PlayItem para especificar en el momento en que ha de iniciarse la reproducción de la ruta secundaria en la cronología de la ruta principal. Según se describió anteriormente, sync_PlayItem_id y sync_start_PTS_of_PlayItem se utilizan para el
25 caso ilustrado en las Figuras 3 y 4 (esto es, cuando la temporización de reproducción del flujo AV principal y la temporización de reproducción del fichero indicado por la ruta secundaria son las mismas), y no se utilizan cuando la temporización de reproducción del flujo AV principal y la temporización de reproducción de un fichero indicado por la ruta secundaria son diferentes entre sí (p.e., cuando una imagen fija referida por la ruta principal y una señal de audio referida por la ruta secundaria no están sincronizadas entre sí, como BGM para una diapositiva que incluye
30 imágenes fijas). Los campos de SubPlayItem_IN_time, SubPlayItem_OUT_time, sync_PlayItem_id y sync_start_PTS_of_PlayItem se comparten en el Clip referido por el SubPlayItem.

A continuación, se proporcionará una descripción de un caso en el que el SubPlayItem se refiere a múltiples Clips (un caso de si (is_multi_Clip_entries=1b), es decir, un caso en el que se registran múltiples Clips). Más
35 concretamente, se proporcionará una descripción de un caso en el que el SubPlayItem se refiere a múltiples Clips según se ilustra en la Figura 4.

Num_of_Clip_entries indica el número de Clips y especifica los Clips y el número de Clip_Information_file_names[SubClip_entry_ids] especifica los Clips distintos de un Clips que tenga un nombre de
40 Clip_Information_file_name[0]. Es decir, num_of_Clip_entries especifica Clips que tengan Clip_Information_file_name[1], Clip_Information_file_name[2] y así sucesivamente que no sea un Clip que tenga Clip_Information_file_name[0]. El SubPlayItem contiene también Clip_codec_identifier[SubClip_entry_id] para especificar un sistema de codificación-decodificación de Clip, ref_to_STC_id[SubClip_entry_id] que es información respecto a los puntos discontinuos de STC (puntos discontinuos basados en el tiempo del sistema) y
45 reservado_para_futuro_uso.

SubPlayItem_IN_time, SubPlayItem_OUT_time, sync_PlayItem_id, y sync_start_PTS_of_PlayItem son compartidos por los múltiples Clips. En la realización, a modo de ejemplo, ilustrada en la Figura 4, SubPlayItem_IN_time, SubPlayItem_OUT_time, sync_PlayItem_id y sync_start_PTS_of_PlayItem son compartidos por SubClip_entry_id=0
50 y SubClip_entry_id=1. Los subtítulos basados en el texto para SubClip_entry_id seleccionada se reproducen sobre la base del tiempo SubPlayItem_IN_time, SubPlayItem_OUT_time, sync_PlayItem_id y sync_start_PTS_of_PlayItem.

Los valores de SubClip_ids se asignan desde 1 en el orden en el que aparecen
55 Clip_Information_file_names[SubClip_entry_ids] en el SubPlayItem. SubClip_entry_id de Clip_Information_file_name[0] es 0.

La Figura 22 ilustra la sintaxis de PlayItem().

60 Un campo de longitud tiene un número entero sin signo de 16 bits que indica el número de bytes inmediatamente después de este campo de longitud hasta el final de PlayItem(). Clip_Information_file_name[0] es un campo para especificar un Clip referido por el PlayItem. En la realización, a modo de ejemplo, ilustrada en la Figura 3, un flujo AV principal se refiere por Clip_Information_file_name[0]. El PlayItem() contiene también Clip_codec_identifier[0] para especificar un sistema de codificación-decodificación de clic, reservado_para_futuro_uso, es una condición de
65 conexión multi-ángulo is_multi_angle, connection_condition y ref_to_STC_id[0] que es información respecto a los puntos discontinuos de STC (puntos discontinuos basados en el tiempo del sistema). El PlayItem() contiene,

además, IN_time y OUT_time para especificar el segmento de reproducción del PlayItem en el Clip. En la realización, a modo de ejemplo ilustrada en la fi 3, IN_time y OUT_time representan el margen de reproducción del fichero de flujo AV de Clip principal. El PlayItem() contiene, además, UO_mask_table(), PlayItem_random_access_mode y still_mode. Una descripción para un caso en el que múltiples is_multi_angles están contenidos se omite en esta descripción, puesto que dicho caso no está directamente relacionado con la presente invención.

STN_table() en PlayItem() proporciona un sistema que permite la operación de conmutación de audio o conmutación de subtítulos realizada por el usuario para seleccionar audio o subtítulos desde un Clip referido por el PlayItem y los Clips referidos por una o más rutas secundarias, cuando un PlayItem se interés y una o más rutas secundarias a reproducirse en asociación con el PlayItem son objeto de preparación. STN_table() proporciona también un sistema que permite al usuario seleccionar una reproducción de mezcla de dos flujos de audio.

La Figura 23 ilustra, a modo de primer ejemplo, la sintaxis de STN_table(). STN_table() se establece como atributos de un PlayItem.

Un campo de longitud tiene un número entero sin signo de 16 bits que indica el número de bytes inmediatamente después de este campo de longitud al final de STN_table(). Después del campo de longitud, se prepara 16 bits reservados para futuro uso. Number_of_video_stream_entries indica el número de flujos que se introducen (registran) con video_stream_ids en STN_table(). Video_stream_id es información para identificar un flujo de vídeo y video_stream_number es un número de flujos de vídeo utilizado para la conmutación de vídeos y visto por el usuario.

Number_of_audio_stream_entries indica el número de flujos de primeros flujos de audio que se introduce (registran) con audio_stream_ids dados en STN_table(). Audio_stream_id es información para identificar un flujo de audio y audio_stream_number es un número de flujos de audio utilizados para la conmutación de audio y visto por el usuario.

Number_of_audio_stream2_entries indica el número de flujos de segundos flujos de audio que son audio_stream_id2s dados introducidos en STN_table(). Audio_stream_id2 es información para identificar un flujo de audio y audio_stream_number es un número de flujos de audio utilizado para conmutación de audio y vistos por el usuario. Más concretamente, los flujos de audio indicados por number_of_audio_stream_entries que se introducen en STN_table() son flujos de audio decodificados por un primer decodificador de audio 75-1 en el aparato de reproducción 20 ilustrado en la Figura 35 y descrito a continuación y los flujos de audio indicados por number_of_audio_stream2_entries introducidos en STN_table() son flujos de audio decodificados por un segundo de decodificador de audio 75-2 en el aparato de reproducción 20 ilustrado en la Figura 35 y descritos a continuación. De esta manera, los flujos de audio que han de decodificarse por los dos decodificadores de audio, respectivamente, pueden introducirse en STN_table() ilustrado en las Figuras 23 y 33 y 34 se aplican a continuación.

En adelante, los flujos de audio indicados por number_of_audio_stream_entries y decodificados por el primer decodificador de audio 75-1 en el aparato de reproducción 20 ilustrado en la Figura 35, esto es, flujos de audio primarios, se denomina "flujo de audio nº 1" y flujos de audio indicados por number_of_audio_stream2_entries y decodificados por el segundo decodificador de audio 75-2 en el aparato de reproducción 20 ilustrado en la Figura 35, esto es flujos de audio secundarios que se denominan "flujo de audio nº 2". El flujo de audio nº 1 se supone que es un flujo de audio, al que se proporciona prioridad sobre el flujo de audio nº 2.

Number_of_PG_txtST_stream_entries indica el número de flujos a los que se proporciona PG_txtST_stream_ids introducidos en STN_table(). En este caso, un flujo (PG: flujo de gráficos de presentación) y un siguiente fichero de subtítulos de texto (txtST), que se obtienen realizando una codificación de longitud de recorrido en subtítulos de mapa de bits tales como imágenes secundarias de DVD se introducen a este respecto. PG_txtST_stream_id es información para identificar un flujo de subtítulos y PG_txtST_stream_number es un número de flujos de subtítulos (un número de flujo de subtítulos de texto) que se utiliza para la conmutación de subtítulos de texto y visto por el usuario.

Number_of_IG_stream_entries indica el número de flujos que son IG_stream_ids dados introducidos en STN_table(). En este caso, se introduce un flujo de gráficos interactivos. IG_stream_id es información para identificar un flujo de gráficos interactivos e IG_stream_number indica un número de flujo de gráficos utilizado para la conmutación de gráficos y visto por el usuario.

La sintaxis de stream_entry() se describirá ahora haciendo referencia a la Figura 24.

Un campo de longitud tiene un número entero sin signo de 8 bits que indica el número de bytes inmediatamente después de este campo de longitud al final de stream_entry(). Type es un campo de 8 bits que indica el tipo de información requerida para identificar, de forma única, un flujo al que se proporciona el número de flujo anteriormente descrito.

Para `type=1`, se especifica un identificador ID de paquete de 16 bits (PID) con el fin de simplificar uno de múltiples flujos elementales multiplexados en un Clip (Clip principal) referidos por el `PlayItem`. Este PID se indica por `ref_to_stream_PID_of_mainClip`. Es decir, para `type=1`, que especifica solamente un PID en fichero de flujo AV de Clip principal que permite la determinación de un flujo.

Para `type=2`, cuando la ruta secundaria se refiere a múltiples Clips al mismo tiempo y múltiples flujos elementales se multiplexan en cada Clip, `SubPath_id`, un identificador Clip ID y un identificador ID de paquete (PID) de la ruta secundaria se especifican con el fin de identificar uno de los múltiples flujos elementales en un Clip (Clip secundario) referido por la ruta secundaria. `SubPath_id` se indica por `ref_to_SubPath_id`, estando el Clip ID indicado por `ref_to_SubClip_entry_id` y el PID se indica por `ref_to_stream_PID_of_SubClip`. De este modo, `type=2` se utiliza cuando múltiples Clips se refieren en el `SubPlayItem` y cada Clip se refiere por múltiples flujos elementales.

De esta manera, los tipos (dos tipos, esto es, `type=1` y `type=2`) se utilizan a este respecto. En consecuencia, cuando una o más rutas secundarias han de reproducirse en asociación con un `PlayItem` se preparan, un flujo elemental puede identificarse a partir de un Clip referido por el `PlayItem` y Clips referidos por una o más rutas secundarias. `Type=1` indica un Clip (Clip principal) referido por una ruta principal y `type=2` indica Clip (Clip secundario) referido por una ruta secundaria.

Haciendo referencia de nuevo a la Figura 23 se proporcionará una descripción de `STN_table()`. En el bucle 'for' de un ID de flujo de vídeo (`video_stream_id`), `video_stream_id`, que comienza desde 0, se proporciona secuencialmente a un flujo elemental de vídeo identificado para cada `stream_entry()`. Un número de flujo de vídeo (`video_stream_number`) puede utilizarse en lugar de ID de flujo de vídeo (`video_stream_id`). En este caso, `video_stream_number` se proporciona desde 1 y no desde 0. Es decir, `video_stream_number` se obtiene añadiendo 1 al valor de `video_stream_id`. El número de flujo de vídeo se utiliza para la conmutación de vídeo y se ve por el usuario y de este modo, se define desde 1.

De modo similar, en el bucle 'for' de un ID de flujo de audio (`audio_stream_id`), `audio_stream_id`, que comienza desde 0, se proporciona secuencialmente a un flujo elemental de audio identificado para cada entrada de flujo `stream_entry()`. De la misma manera que con los flujos de vídeo, un número de flujos de audio (`audio_stream_number`) puede utilizarse en lugar del identificador ID de flujo de audio (`audio_stream_id`). En este caso, `audio_stream_number` se proporciona desde 1 y no desde 0. Es decir, `audio_stream_number` se obtiene añadiendo 1 al valor de `audio_stream_id`. El número de flujo de audio se utiliza para la conmutación de audio y se ve por el usuario y de este modo, se define desde 1.

De modo similar, en el bucle 'for' de un identificador ID2 de flujo de audio (`audio_stream_id2`), `audio_stream_id2`, que comienza desde 0, se proporciona secuencialmente a un flujo elemental de audio identificado para cada `stream_entry()`. Como en el caso de los flujos de vídeo, un número de flujo de audio 2 (`audio_stream_number2`) puede utilizarse en lugar del ID2 de flujo de audio (`audio_stream_id2`). En este caso, `audio_stream_number2` se proporciona desde 1 y no desde 0. Es decir, `audio_stream_number2` se obtiene añadiendo 1 al valor de `audio_stream_id2`. El número de flujo de audio 2 se utiliza para la conmutación de audio y se ve por el usuario y de este modo, se define desde 1.

Es decir, en `STN_table()` ilustrado en la Figura 23, se definen los flujos de audio especificados por `number_of_audio_stream_entries` (flujo de audio nº 1) y los flujos de audio especificados por `number_of_audio_stream2_entries` (flujo de audio nº 2) se definen a este respecto. Dicho de otro modo, es posible introducir un flujo de audio nº 1 y un flujo de audio nº 2 utilizando `STN_table()`, de modo que el usuario pueda seleccionar dos flujos de audio que han de reproducirse en sincronización entre sí.

De modo similar, en el bucle 'for' de un ID de flujo de subtítulos de texto (`PG_txtST_stream_id`), `PG_txtST_stream_id`, que se inicia desde 0, se proporciona secuencialmente a un flujo elemental de subtítulos de mapa de bits o subtítulos de texto identificados para cada `stream_entry()`. Como en el caso de los flujos de vídeo, un número de flujo de subtítulos (`PG_txtST_stream_number`) puede utilizarse en el lugar del ID de flujo de subtítulos (`PG_txtST_stream_id`). En este caso, `PG_txtST_stream_number` se proporciona desde 1 y no desde 0. Es decir, `PG_txtST_stream_number` se obtiene añadiendo 1 al valor de `PG_txtST_stream_id`. El número de flujo de subtítulos es un número de flujo de subtítulos (un número de flujo de subtítulos de texto) utilizado para la conmutación de subtítulos y visto por el usuario y en consecuencia, se define desde 1.

De modo similar, en el bucle 'for' de un ID de flujo de gráficos (`IG_stream_id`), `IG_stream_id`, que se inicia desde 0, se proporciona secuencialmente a un flujo elemental de gráficos interactivos identificado para cada `stream_entry()`. De la misma manera que con los flujos de vídeo, un número de flujos de gráficos (`IG_stream_number`) puede utilizarse en lugar del ID de flujo de gráficos (`IG_stream_id`). En este caso, `IG_stream_number` se proporciona desde 1 y no desde 0. Es decir, `IG_stream_number` se obtiene añadiendo 1 al valor de `IG_stream_id`. El número de flujo de gráficos se utiliza para la conmutación de gráficos y se ve por el usuario y en consecuencia, se define desde 1.

A continuación, se proporcionará una descripción del atributo `stream_attribute()` en `STN_table()` ilustrado en la Figura 23.

En las declaraciones 'for' posteriores a reservado_para_futuro_uso los flujos de vídeo se refieren en función del número de flujos de audio, los flujos de audio se refieren en función de la ruta principal y de las rutas secundarias que se establecen para flujos de audio por el fabricante de disco, los flujos de PG texto-ST se refieren en función del número de flujos de PG texto-ST y los flujos IG se refieren en función del número del flujos IG.

Stream_attribute() en el bucle 'for' del ID flujo de vídeo (video_stream_id) proporciona información de atributos de flujo de un flujo elemental de vídeo identificado para cada entrada de flujo stream_entry(). Es decir, este stream_attribute() proporciona la información de atributo de flujo del flujo elemental de vídeo identificado para cada stream_entry().

De modo similar, el atributo stream_attribute() en el bucle 'for' del ID de flujo de audio (audio_stream_id) proporciona información de atributo de flujo de un flujo elemental de audio identificado para cada stream_entry(). Es decir, este atributo stream_attribute() proporciona la información de atributo de flujo de un flujo elemental de audio identificado para cada stream_entry(). A modo de ejemplo, puesto que el número de flujos elementales de audio identificados por type=1 o type=2 en stream_entry() ilustrado en la Figura 24 es 1, stream_attribute() proporciona la información de atributo de flujo del flujo elemental de audio único.

De modo similar, el atributo stream_attribute() en el bucle 'for' del ID2 de flujo de audio (audio_stream_id2) proporciona información de atributo de flujo del flujo elemental de audio identificado para cada stream_entry(). Es decir, este stream_attribute() proporciona la información de atributo de flujo de un flujo elemental de audio identificado para cada stream_entry(). A modo de ejemplo, puesto que el número de flujos elementales de audio identificados por type=1 o type=2 en stream_entry() ilustrado en la Figura 24 es 1, stream_attribute() proporciona la información de atributo de flujo del flujo elemental de audio único.

De modo similar, stream_attribute() en el bucle 'for' del ID de flujo de subtítulos (PG_txtST_stream_id) proporciona información de atributo de flujo de un flujo elemental de subtítulos de mapa de bit o flujo elemental de subtítulos de texto identificado para cada stream_entry(). Es decir, este stream_attribute() proporciona la información de atributo de flujo de un flujo elemental de subtítulos de mapa de bits identificado stream_entry().

De modo similar, stream_attribute() en el bucle 'for' del ID de flujo de gráficos (IG_stream_id) proporciona información de atributo de flujo de un flujo elemental de gráficos interactivos identificado para cada stream_entry(). Es decir, este stream_attribute() proporciona la información de atributo de flujo de un flujo elemental de ras interactivos identificado para cada stream_entry().

La sintaxis de stream_attribute() se describirá ahora con referencia a la Figura 25.

Un campo de longitud tiene un número entero sin signo de 16 bits que indica el número de bytes inmediatamente después de este campo de longitud al final de stream_attribute().

Según se ilustra en la Figura 26, stream_coding_type indica el tipo de codificación de un flujo elemental. Como el tipo de codificación de un flujo elemental, se proporcionan el flujo de vídeo MPEG-2, audio HDMV LPCM, Audio Dolby AC-3, audio dts, flujo de gráficos de presentación, flujo de gráficos interactivos y flujos de subtítulos de texto.

Según se ilustra en la Figura 27, video_format indica el formato de vídeo de un flujo elemental de vídeo. Como el formato de vídeo de un flujo elemental de vídeo, se proporciona 480i, 576i, 480p, 1080i, 720p y 1080p.

Según se ilustra en la Figura 28, frame_rate indica la tasa de tramas de un flujo elemental de vídeo. Como la tasa de tramas de un flujo elemental de vídeo, se proporciona 24000/1001, 24, 25, 30000/1001, 50 y 60000/1001.

Según se ilustra en la Figura 29, aspect_ratio indica la información de ratio de aspecto de un flujo elemental de vídeo. Como la información de ratio de aspecto de un flujo elemental de vídeo, se proporciona un ratio de aspecto de pantalla 4:3 y un ratio de aspecto de pantalla 16:9.

Según se ilustra en la Figura 30, audio_presentation_type indica la información de tipo de presentación de un flujo elemental de audio. Como la información de tipo de presentación de un flujo elemental de audio, se proporciona los así denominados canal mono único, canal mono dual, canal estéreo (2 canales) y múltiples canales.

Según se ilustra en la Figura 31, sampling_frequency indica la frecuencia de muestreo de un flujo elemental de audio. Como la frecuencia de muestreo de un flujo elemental de audio, se proporciona 48 kHz y 96 kHz.

Audio_language_code indica el código de lenguaje (p.e., japonés, coreano o chino) de un flujo elemental de audio.

PG_language_code indica el código de lenguaje (p.e., japonés, coreano o chino) de un flujo elemental de subtítulos de mapa de bits.

IG_language_code indica el código de lenguaje (p.e., japonés, coreano o chino) de un flujo elemental de gráficos

interactivos.

TextST_language_code indica el código de lenguaje (p.e., japonés, coreano o chino) de un flujo elemental de subtítulos de texto.

Según se ilustra en la Figura 32, character_code indica el código de caracteres de un flujo elemental de subtítulos. Como el código de caracteres de un flujo elemental de subtítulos de texto, Unicode VI.1 (ISO 10646-1), Shift JIS (japonés), KSC 5601-1987 incluyendo KSC 5653 para caracteres románicos (coreano), GB 180302000 (chino), GB2312 (chino) y BIG5 (chino) se proporcionan a este respecto.

La sintaxis de stream_attribute() ilustrada en la Figura 25 se describirá concretamente a continuación haciendo referencia a las Figuras 25 y 26 a 32.

Cuando el tipo de codificación (stream_coding_type ilustrado en la Figura 25) de un flujo elemental es el flujo de vídeo MPEG-2 (Figura 26), stream_attribute() contiene el formato de vídeo (Figura 27), la tasa de trama (Figura 28) y la información de ratio de aspecto (Figura 29) del flujo elemental.

Cuando el tipo de codificación (stream_coding_type ilustrado en la Figura 25) de un flujo elemental es HDMV LPCM, Dolby AC-3, o dts (Figura 26), stream_attribute() contiene la información de tipo de presentación (Figura 30), la frecuencia de muestreo (Figura 31) y el código de lenguaje del flujo elemental de audio.

Cuando el tipo de codificación (stream_coding_type ilustrado en la Figura 25) de un flujo elemental es el flujo de gráficos de presentación (Figura 26), stream_attribute() contiene el código de lenguaje del flujo elemental de subtítulos de mapa de bits.

Cuando el tipo de codificación (stream_coding_type ilustrado en la Figura 25) de un flujo elemental es un flujo de gráficos interactivos (Figura 26), stream_attribute() contiene el código de lenguaje del flujo elemental de gráficos interactivos.

Cuando el tipo de codificación (stream_coding_type ilustrado en la Figura 25) de un flujo elemental es un flujo de subtítulos de texto (Figura 26), stream_attribute() contiene el código de caracteres (Figura 32) y el código de lenguaje del siguiente flujo elemental de subtítulos de texto.

La información de atributo anteriormente descrita no está limitada a lo que se describió con anterioridad.

De esta manera, cuando un PlayItem y una o más rutas secundarias a reproducirse en asociación con el PlayItem son objeto de preparación, la información de atributo de un flujo elemental identificado por stream_entry() puede conocerse a partir de un Clip referido por el PlayItem y Clips referidos por la una o más rutas secundarias sobre la base de stream_attribute().

Comprobando la información de atributo (stream_attribute()), el aparato de reproducción puede comprobar si tiene, o no, una función para reproducir el flujo elemental. Además, comprobando la información de atributo, el aparato de reproducción puede seleccionar un flujo elemental correspondiente a la información inicial del establecimiento del lenguaje del aparato de reproducción.

A modo de ejemplo, se supone que el aparato de reproducción tiene solamente una función para reproducir flujos elementales de subtítulos de mapa de bits y carece de una función para reproducir flujos elementales de subtítulos de texto. Cuando el usuario emite una instrucción de conmutación de idiomas al aparato de reproducción, el aparato de reproducción selecciona secuencialmente solamente flujos elementales de subtítulos de mapa de bits desde el bucle 'for' del ID de flujo de subtítulos (PG_txtST_stream_id) y reproduce los flujos elementales de subtítulos de mapa de bits seleccionados.

A modo de ejemplo, se supone que la información inicial de establecimiento de lenguajes del aparato de reproducción es japonés. Cuando el usuario emite una instrucción de conmutación de audio al aparato de reproducción, el aparato de reproducción selecciona secuencialmente solamente flujos elementales de audio cuyo código de lenguaje es japonés desde el bucle 'for' del ID de flujo de audio (Audio_stream_id) y reproduce los flujos elementales de audio seleccionados.

A modo de ejemplo, para reproducir un flujo AV (p.e., una película) que se refiera por una ruta principal y que está constituida por un flujo de vídeo y flujos de audio, cuando el usuario emite una instrucción de conmutación de audio al aparato de reproducción para especificar (seleccionar) el flujo de audio nº 1 (salida de audio en una película típica) y el flujo de audio nº 2 (comentarios del director e intérpretes) como salidas de audio a reproducir, el aparato de reproducción mezcla (superpone) el flujo de audio nº 1 y el flujo de audio nº 2 y reproduce el flujo de audio resultante junto con el flujo de vídeo.

Como puede entenderse haciendo referencia a STN_table() ilustrado en la Figura 23, ambos flujos de audio nº 1 y

flujo de audio nº 2 pueden ser flujos de audio contenidos en un Clip referido por una ruta principal. Como alternativa, uno de entre el flujo de audio nº 1 y el flujo de audio nº 2 puede ser un flujo de audio contenido en un Clip referido por una ruta principal y el otro flujo de audio puede ser un flujo de audio contenido en un Clip referido por una ruta secundaria. De esta manera, dos flujos de flujos de audio superpuestos en un flujo AV principal referidos por un flujo principal pueden seleccionarse, mezclarse y reproducirse.

De esta manera, cuando PlayItem y una o más rutas secundarias a reproducirse en asociación con el PlayItem son objeto de preparación, STN_table() en PlayItem() proporciona un sistema que permite la operación de conmutación de audio o de conmutación de subtítulos realizada por el usuario para seleccionar audio o subtítulos desde un Clip referido por el PlayItem y los Clips referidos por una o más rutas secundarias. De este modo, se puede realizar una operación interactiva en un flujo y fichero de datos que sean diferentes de un flujo AV que contiene un flujo AV principal y que ha de reproducirse.

Puesto que la configuración en la que se utilizan múltiples rutas secundarias en una Playlist y SubPlayItems se refieren por las rutas secundarias correspondientes, es posible conseguir un flujo AV con una alta capacidad de expansión y una libertad elevada. Es decir, en esta configuración, se puede añadir posteriormente un SubPlayItem. A modo de ejemplo, cuando un fichero de flujo AV de Clip se refiere por una ruta principal y una Playlist asociada con el fichero de flujo AV de Clip existen y esta Playlist se sobrescribe por una Playlist a la que se añade otra ruta secundaria, es posible realizar la reproducción haciendo referencia al fichero de flujo AV de Clip referido por la ruta principal y un fichero de flujo AV de Clip que es diferente del fichero de flujo AV de Clip referido por la ruta principal. De este modo, la disposición operativa puede tener capacidad de extensión.

Además, STN_table() en PlayItem() proporciona un sistema en el que el flujo de audio nº 1 decodificado por el primer decodificador de audio 75-1 en el aparato de reproducción 20 ilustrado en la Figura 35 y descrito a continuación y el flujo de audio nº 2 decodificado por el segundo decodificador de audio 75-2 pueden mezclarse y reproducirse. STN_table() proporciona un sistema en el que, a modo de ejemplo, cuando PlayItem() y una o más rutas secundarias a reproducirse en asociación mutua son objeto de preparación, un flujo de audio de un Clip referido por el PlayItem, siendo el flujo de audio considerado como flujo de audio nº 1 y un flujo de audio de un Clip referido por una ruta secundaria, siendo el flujo de audio considerado como flujo de audio nº 2, pueden mezclarse y reproducirse. STN_table() proporciona también un sistema en el que, a modo de ejemplo, dos flujos de audio contenidos en un Clip (Clip principal) referidos por un PlayItem, siendo considerados los dos flujos de audio como flujo de audio nº 1 y flujo de audio nº 2, pueden mezclarse y reproducirse. Con esta disposición operativa, un flujo de audio (p.e., un flujo de comentarios del director) que sea diferente del flujo de audio de un flujo principal de reproducción en el que un flujo AV principal se registra puede superponerse y reproducirse. Dos flujos de audio nº 1 y nº 2 superpuestos en un flujo AV principal pueden también superponerse (mezclarse) y reproducirse.

Además, STM_table() puede definir una combinación de un flujo de vídeo secundario, un flujo de audio primario, un flujo de audio secundario y un flujo de subtítulos que han de reproducirse simultáneamente en combinación con un flujo de vídeo primario. Las Figuras 33 y 34 ilustran una segunda realización, a modo de ejemplo, de la sintaxis de STN_table().

Las Figuras 33 y 34 ilustran, a modo de ejemplo, la sintaxis de STN_table() cuando se define una combinación de un flujo de vídeo secundario, un flujo de audio primario, un flujo de audio secundario y un flujo de subtítulos que han de reproducirse simultáneamente en combinación con un flujo de vídeo primario. En las Figuras, para las partes que sean las mismas que las ilustradas en la Figura 23, se omitirá las descripciones redundantes correspondientes.

En la realización, a modo de ejemplo, ilustrada en las Figuras 33 y 34, se define una combinación de un flujo de vídeo secundario, un flujo de audio primario, un flujo de audio secundario y un flujo de subtítulos que han de reproducirse simultáneamente en combinación con un flujo de vídeo primario como sigue. Es decir, en primer lugar, uno o más flujos de vídeo secundarios que han de reproducirse simultáneamente en combinación con un flujo de vídeo primario son objeto de definición. Posteriormente, flujos de audio (un flujo de audio primario y un flujo de audio secundario) y un flujo de subtítulos, que han de reproducirse simultáneamente, son definidos con respecto a cada uno de los uno o más flujos de vídeo secundarios.

Más concretamente, en la Figura 33, number_of_video_stream2_entries indica el número de flujos cuyos video_stream_id2s dados se introducen (registran) en STN_table(). Video_stream_id2 es información para identificar un flujo de vídeo secundario y video_stream_number2 es un número de flujo de vídeo secundario utilizado para la conmutación de vídeo y visto por el usuario.

En la Figura 34, en el bucle 'for' de un ID2 de flujo de vídeo (video_stream_id2), video_stream_id2, que se inicia desde 0, se proporciona secuencialmente a un flujo elemental de vídeo (un dato de flujo elemental de vídeo que sirve como un flujo de vídeo secundario) identificado para cada stream_entry().

En este caso, number_of_Audio_combinations_for_video2 se proporciona a video_stream_id2 y en la declaración 'for' posterior, audio_stream_ids y audio_stream_id2s se proporcionan en función del número de number_of_Audio_combinations_for_video2s. Number_of_Audio_combinations_for_video2 y la declaración 'for' posterior

sirven como información que define una combinación de flujos de audio que han de reproducirse simultáneamente con los flujos de vídeo secundarios, esto es, una combinación de un flujo de audio primario identificado por `audio_stream_id` y un flujo de audio secundario identificado por `audio_stream_id2`. El número de conjuntos de flujos de audio que pueden combinarse con los flujos de vídeo secundarios identificados por `video_stream_id2` (esto es, conjuntos de flujos de audio primarios y flujos de audio secundarios) se expresan por `number_of_Audio_combinations_for_video2`. Como un conjunto de flujos de audio que pueden combinarse con los flujos de vídeo secundarios codificados por `video_stream_id2`, `audio_stream_id` que identifica los flujos de audio primarios y `audio_stream_id2` que identifica los flujos de audio secundarios se definen en una declaración 'for' posterior a `number_of_Audio_combinations_for_video2`.

Además, `number_of_Subtitle_combinations_for_video2` se proporciona al mismo `video_stream_id2` y en la declaración 'for' posterior, `PG_textST_stream_ids` se proporciona en función del número de establecido por `number_of_Subtitle_combinations_for_video2s`. `Number_of_Subtitle_combinations_for_video2` y la declaración 'for' posterior sirven como información que define una combinación de flujos de subtítulos a reproducirse simultáneamente con los flujos de vídeo secundarios, esto es, una combinación de flujos de subtítulos identificada por `PG_textST_stream ID`. El número de flujos de subtítulos que pueden combinarse con los flujos de vídeo secundarios identificados por `video_stream_id2` se expresa por `number_of_Subtitle_combinations_for_video2`. Además, `PG_textST_stream_id video_stream_id2` que identifica un flujo de subtítulos que pueden combinarse con el flujo de vídeo secundario identificado por `video_stream_id2` se define en una declaración 'for' posterior a `number_of_Subtitle_combinations_for_video2`.

Como en el caso descrito con referencia a la Figura 23, se pueden utilizar números en lugar de los identificadores IDs, a modo de ejemplo, un número de flujo de audio (`audio_stream_number`) puede utilizarse en lugar de `audio_stream_id` y un número de flujo de audio 2 (`audio_stream_number2`) puede utilizarse en lugar de `audio_stream_id2`. Lo mismo es cierto para un flujo de vídeo y un flujo de subtítulos.

En las Figuras 33 y 34, un flujo de vídeo secundario a reproducirse simultáneamente con un flujo de vídeo primario pueden definirse mediante el uso de `video_stream_id2` y una combinación de un flujo de audio primario, un flujo de audio secundario y un flujo de subtítulos que han de reproducirse simultáneamente con el flujo de vídeo secundario pueden definirse también mediante el uso de `video_stream_id2`, `audio_stream_id`, `audio_stream_id2` y `PG_textST_stream_id`. Es decir, es posible definir una combinación de un flujo de vídeo secundario, un flujo de audio primario, un flujo de audio secundario y un flujo de subtítulos que se reproducen simultáneamente con un flujo de vídeo primario.

De este modo, una combinación contradictoria no se define como una combinación de un flujo de vídeo secundario, un flujo de audio primario, un flujo de audio secundario y un flujo de subtítulos que han de reproducirse simultáneamente con un flujo de vídeo primario de modo que el usuario pueda seleccionar, con facilidad, una combinación de flujos que puedan reproducirse simultáneamente con un flujo de vídeo primario.

`STN_table( )` en `PlayItem( )` define que, cuando este `PlayItem` y una o más rutas secundarias a reproducirse en asociación con el `PlayItem` son objeto de preparación, `Subpath_type` de las rutas secundarias se definen como siendo 5 a 7, esto es, una ruta o rutas de presentación denominada *picture-in-picture* (una o más rutas de un flujo de vídeo secundario, un flujo de audio secundario y un flujo de gráficos de presentación). Con esta disposición operativa, visualizar un flujo de vídeo secundario en un flujo de vídeo primario que se reproduce y visualiza como un flujo principal se hace posible realizar una presentación visual del tipo *picture-in-picture* según se describe con referencia a la Figura 5.

El establecimiento de presentación visual *picture-in-picture* (superposición de imagen en imagen) se proporciona en `pip_metadata`, según se describe con referencia a la Figura 9, 11, o 12; por lo tanto, el tamaño y la posición de pantalla de la pantalla de presentación visual secundaria 2 descrita con referencia a la Figura 5 puede establecerse arbitrariamente por el productor de contenidos o el proveedor de contenidos, sin depender de un aparato de reproducción. El establecimiento de presentación visual del tipo *picture-in-picture* se proporciona en `pip_metadata` según se describe con referencia a las Figuras 9, 11, o 12, en lugar de los propios datos de un flujo de vídeo (un flujo de vídeo secundario) visualizado en un formato de *picture-in-picture*. Es decir, cuando se desea cambiar el tamaño y/o la posición de pantalla de la pantalla de presentación visual secundaria 2 descrita con referencia a la Figura 5, es suficiente modificar la declaración de `pip_metadata`, sin tener que modificar un flujo de vídeo (un flujo de vídeo secundario) visualizado en un formato de *picture-in-picture*.

El establecimiento de presentación visual del tipo *picture-in-picture* se proporciona en `pip_metadata`, según se describe con referencia a las Figuras 9, 11, o 12; por lo tanto, el tamaño de la señal de vídeo visualizada en la pantalla de presentación visual secundaria 2 descrita con referencia a la Figura 5 puede establecerse arbitrariamente por el productor de contenidos o el proveedor de contenidos, sin depender del tamaño de imagen original del flujo de vídeo secundario.

Un aparato de reproducción, según la presente invención, se describirá a continuación. La Figura 35 es un diagrama de bloques que ilustra, a modo de ejemplo, la configuración de un aparato de reproducción 20 según la presente

invención. Este aparato de reproducción 20 es el mismo que el aparato de reproducción 20 para reproducir la lista de reproducción PlayList anteriormente descrita que tiene una ruta principal y rutas secundarias.

El aparato de reproducción 20 tiene una unidad de memorización 31, un conmutador 32, una unidad de decodificador AV 33 y un controlador 34.

En el caso, a modo de ejemplo, ilustrado en la Figura 35, el controlador 34 efectúa, en primer lugar, la lectura de un fichero PlayList mediante la unidad de memorización 31 y sobre la base de la información del fichero PlayList, el controlador 34 efectúa la lectura de un flujo AV y datos AV desde un soporte de registro, tal como un disco duro HDD, disco Blu-ray, o DVD, mediante la unidad de memorización 31. Utilizando una interfaz de usuario, el usuario puede emitir una instrucción para la conmutación de señales de audio, subtítulos, etc., al controlador 34. La información inicial del establecimiento del lenguaje del aparato de reproducción 20 se suministra desde una unidad de memorización, que no se ilustra o unidad similar al controlador 34.

El fichero PlayList contiene STN_table() además de la información de una ruta principal y de rutas secundarias. El controlador 34 efectúa la lectura, desde el soporte de registro o dispositivo similar, mediante la unidad de memorización 31, un fichero de flujo AV de Clip (en lo sucesivo referido como un "Clip principal") referido por un PlayItem contenido en el fichero PlayList, un fichero de flujo AV de Clip (en adelante referido como un "Clip secundario") referido por un SubPlayItem y datos de subtítulos de texto referidos por un SubPlayItem. El Clip principal referido por el PlayItem y el Clip secundario referido por el SubPlayItem puede memorizarse en diferentes soportes de registro. A modo de ejemplo, el Clip principal puede registrarse en un soporte de registro y un Clip secundario correspondiente puede suministrarse a través de una red, que no se ilustra, y memorizarse en un disco duro HDD. El controlador 34 realiza también el control para seleccionar un flujo elemental correspondiente a la función de reproducción del aparato (el aparato de reproducción 20) para realizar dicha reproducción o realiza el control para seleccionar solamente un flujo elemental correspondiente a la información inicial del establecimiento de lenguaje del aparato de reproducción 20 para realizar la reproducción.

Cuando el controlador 34 se refiere a información (o un identificador) que se describe en ID1/ID2 de PlayListExtensionData() en el fichero PlayList y detecta que data_block de PlayListExtensionData() proporciona información (pip_metadata) con respecto a la presentación visual del tipo picture-in-picture, se refiere a pip_metadata descrito con referencia a las Figuras 9 o 11 para obtener el establecimiento de presentación visual de un vídeo secundario a visualizarse en la pantalla de presentación visual secundaria 2 descrita con referencia a la Figura 5. El controlador 34 controla, entonces, la combinación de las señales de vídeo del flujo de vídeo primario y de los flujos de vídeo secundarios, realizándose la combinación por un generador de planos de vídeo 92 en la unidad del decodificador AV 33.

La unidad del decodificador AV 33 incluye las memorias intermedias 51 a 54, filtros de PID 55 y 56, conmutadores 57 a 59, un filtro de PID 60, un decodificador de fondo 71, un primer decodificador de vídeo 72-1, un segundo decodificador de vídeo 72-2, un decodificador de gráficos de presentación 73, un decodificador de gráficos interactivos 74, el primer decodificador de audio 75-1, el segundo decodificador de audio 75-2, una composición de Texto-ST 76, un conmutador 77, un generador de planos de fondo 91, un generador de planos de vídeo 92, un generador de planos de gráficos de presentación 93, un generador de planos de gráficos interactivos 94, una memoria intermedia 95, un procesador de datos-vídeo 96 y procesadores de mezcla 97 y 98.

El primer decodificador de vídeo 72-1 decodifica un flujo de vídeo primario y el segundo decodificador de vídeo 72-1 decodifica un flujo de vídeo secundario. El primer decodificador de audio 75-1 decodifica el flujo de audio nº 1 (un flujo de audio primario) y el segundo decodificador de audio 75-2 decodifica el flujo de audio nº 2 (un flujo de audio secundario). Más concretamente, en STN_table() ilustrado en la Figura 23 y las Figuras 33 y 34, el primer decodificador de vídeo 72-1 decodifica un flujo de vídeo especificado por video_stream_id, el segundo decodificador de vídeo 72-2 decodifica un flujo de vídeo especificado por video_stream_id2, el primer decodificador de audio 75-1 decodifica un flujo de audio especificado por audio_stream_id y el segundo decodificador de audio 75-2 decodifica un flujo de audio especificado por audio_stream_id2.

Según se describió con anterioridad, el aparato de reproducción 20 tiene dos decodificadores de vídeo (el primer decodificador de vídeo 72-1 y el segundo decodificador de vídeo 72-2) para decodificar dos flujos de vídeo y tiene dos decodificadores de audio (el primer decodificador de audio 75-1 y el segundo decodificador de audio 75-2) para decodificar dos flujos de audio. El primer decodificador de vídeo 72-1 y el segundo decodificador de vídeo 72-2 se denominarán "decodificador de vídeo 72" en adelante a no ser que tengan que distinguirse particularmente entre sí y el primer decodificador de audio 75-1 y el segundo decodificador de audio 75-2 se denominarán "decodificador de audio 75" a no ser que hayan de distinguirse particularmente entre sí.

Los datos de ficheros leídos por el controlador 34 se demodulan por una unidad de demodulación y decodificación - ECC que no se ilustra, y el flujo multiplexado demodulado se somete a una corrección de errores. Bajo el control del controlador 34, el conmutador 32 selecciona los datos demodulados y con errores corregidos para cada tipo de flujo y luego, suministra los datos seleccionados a las memorias intermedias correspondientes 51 a 54. Más concretamente, bajo el control del controlador 34, el conmutador 32 realiza la conmutación con el fin de suministrar

datos de imágenes de fondo a la memoria intermedia 51, suministrar datos de Clip principal a la memoria intermedia 52, suministrar datos del Clip secundario a la memoria intermedia 53 y suministrar datos de Texto-ST a la memoria intermedia 54. La memoria intermedia 51 memoriza los datos de imágenes de fondo, la memoria intermedia 52 memoriza los datos de Clip principal, la memoria intermedia 53 memoriza los datos de Clip secundario y la memoria intermedia 54 memoriza los datos de Texto-ST.

El Clip principal es un flujo (p.e., un flujo de transporte) obtenido mediante la multiplexación de vídeo y al menos un flujo de vídeo, audio, subtítulos de mapa de bits (un flujo de gráficos de presentación) y gráficos interactivos. El Clip secundario es un flujo obtenido mediante la multiplexación de al menos un flujo de vídeo, subtítulos de mapa de bits, gráficos interactivos y audio. Los datos del fichero de datos de subtítulos de texto (Text_ST) pueden tener, o no, el formato de un flujo multiplexado, tal como un flujo de transporte.

Cuando el Clip principal, el Clip secundario y los datos de subtítulos de texto son objeto de lectura desde la unidad de memorización 31 (soporte de registro), los ficheros respectivos pueden ser objeto de lectura alternativa en una manera de tiempo compartido. Como alternativa, antes de que el Clip secundario y los datos de subtítulos de texto se lean desde el Clip principal, todos los ficheros pueden precargarse para la memoria intermedia (la memoria intermedia 53 o la memoria intermedia 54).

El aparato de reproducción 20 efectúa la lectura de los datos de dichos ficheros desde el soporte de registro por intermedio de la unidad de memorización 31 y reproduce la señal de vídeo, los subtítulos de mapa de bits, los gráficos interactivos y las señales de audio.

Más concretamente, los datos de flujos objeto de lectura desde la memoria 52, que es una memoria intermedia de lectura de Clip principal, se proporciona, a la salida, al filtro de PID (ID de paquete) 55 en la etapa posterior en una temporización predeterminada. El filtro de PID 55 clasifica el Clip principal de entrada en conformidad con los PIDs (IDs de paquetes) y proporciona los flujos clasificados a los decodificadores de flujos elementales correspondientes en la etapa posterior. Es decir, el filtro de PID 55 suministra un flujo de vídeo al filtro de PID 60 para suministrarlo a uno de entre el primer decodificador de vídeo 72-1 y el segundo decodificador de vídeo 72-2, suministra un flujo de gráficos de presentación al conmutador 57 que sirve como una fuente de suministro para el decodificador de gráficos de presentación 73, suministra un flujo de gráficos interactivos al conmutador 58 que sirve como una fuente de suministro para el decodificador de gráficos interactivos 74 y suministra un flujo de audio al conmutador 59 que sirve como una fuente de suministro para uno de entre el primer decodificador de audio 75-1 y el segundo decodificador de audio 75-2.

El flujo de gráficos de presentación es, a modo de ejemplo, datos de subtítulos de mapa de bits y los datos de subtítulos de texto es, a modo de ejemplo, datos de subtítulos de texto.

Los datos de flujos objeto de lectura desde la memoria intermedia 53, que es una memoria intermedia de lectura de Clip secundario, se proporciona al filtro de PID (ID de paquete) 56 en la etapa posterior en una temporización predeterminada. El filtro de PID 56 clasifica el Clip secundario de entrada en función de los PIDs (IDs de paquetes) y proporciona, a la salida, los flujos clasificados a los decodificadores de flujo elementales correspondientes en la etapa posterior. Es decir, el filtro de PID 56 suministra el flujo de vídeo suministrado al filtro de PID 60 para suministrarlo a uno de entre el primer decodificador de vídeo 72-1 y el segundo decodificador de vídeo 72-2, suministra un flujo de gráficos de presentación al conmutador 57 que sirve como una fuente de suministro para el decodificador de gráficos de presentación 73, suministra un flujo de gráficos interactivos al conmutador 58 que sirve como una fuente de suministro para el decodificador de gráficos interactivos 74 y suministra un flujo de audio al conmutador 59 que sirve como una fuente de suministro para uno de entre el primer decodificador de audio 75-1 y el segundo decodificador de audio 75-2.

Los datos objeto de lectura desde la memoria intermedia 51, que memoriza datos de imágenes de fondo, se suministra al decodificador de fondo 71 en una temporización predeterminada. El decodificador de fondo 71 decodifica los datos de imágenes de fondo y suministra los datos de imágenes de fondo decodificados al generador de plano de fondo 91.

El conmutador 57 selecciona uno de entre el flujo de gráficos de presentación, contenido en el Clip principal y suministrado desde el filtro de PID 55 y el flujo de gráficos de presentación, que se contiene en el Clip secundario y suministra el flujo de gráficos de presentación seleccionado al decodificador de gráficos de presentación 73 en la etapa posterior. El decodificador de gráficos de presentación 73 decodifica el flujo de gráficos de presentación y suministra los datos del flujo de gráficos de presentación decodificados al conmutador 77, que sirve como una fuente de suministro para el generador de planos de gráficos de presentación 93.

El conmutador 58 selecciona uno de entre el flujo de gráficos interactivos, contenido en el Clip principal y suministrado desde el filtro de PID 55 y el flujo de gráficos interactivos, contenido en el Clip secundario, y suministra el flujo de gráficos interactivos seleccionado al decodificador de gráficos interactivos 74 en la etapa posterior. Es decir, el flujo de gráficos interactivos suministra simultáneamente al decodificador de gráficos interactivos 74 es un flujo separado del Clip principal o del Clip secundario. El decodificador de gráficos interactivos 74 decodifica el flujo

de gráficos interactivos y suministra los datos del flujo de gráficos interactivos para generador de plano de gráficos interactivos 94.

Además, el conmutador 59 selecciona uno de entre el flujo de audio, contenido en el Clip principal y suministrado desde el filtro de PID 55 y el flujo de audio, contenido en el Clip secundario, y suministra el flujo de audio seleccionado al primer decodificador de audio 75-1 o al segundo decodificador de audio 75-2 en la etapa posterior. En este caso, el flujo de audio introducido simultáneamente al primer decodificador de audio 75-1 es un flujo separado del Clip principal o del Clip secundario. De forma similar, el flujo de audio simultáneamente introducido al segundo decodificador de audio 75-2 es un flujo separado del Clip principal o del Clip secundario. A modo de ejemplo, cuando el flujo de audio nº 1 y el flujo de audio nº 2 están contenidos en el Clip principal, el filtro de PID 55 filtra el flujo de audio nº 1 y el flujo de audio nº 2 sobre la base de los PIDs de los flujos de audio y suministra el flujo de audio resultante al conmutador 59.

A modo de ejemplo, el conmutador 59 selecciona una conexión con el fin de suministrar el flujo de audio nº 1, suministrado desde el filtro de PID 55, al primer decodificador de audio 75-1 y para suministrar el flujo de audio nº 2, suministrado desde el filtro de PID 55, al segundo decodificador de audio 75-2.

El filtro de PID 60 recibe el flujo de vídeo contenido en el Clip principal y suministrado desde el filtro de PID 55 o el flujo de vídeo contenido en el Clip secundario y suministrado desde el filtro de PID 56. Bajo el control del controlador 34, el filtro de PID 60 determina si el flujo de vídeo recibido es un flujo de vídeo primario o un flujo de vídeo secundario y suministra el flujo de vídeo primario al primer decodificador de vídeo 72-1 y suministra el flujo de vídeo secundario al segundo decodificador de vídeo 72-2.

El flujo de vídeo clasificado por el filtro de PID 60 se suministra al primer decodificador de vídeo 72-1 o al segundo decodificador de vídeo 72-2 en la etapa posterior. El primer decodificador de vídeo 72-1 o el segundo decodificador de vídeo 72-2 decodifica el flujo de vídeo suministrado y proporciona a la salida, los datos de vídeo decodificados al generador de plano de vídeo 92.

Cuando el generador de plano de vídeo 92 recibe los datos de vídeo desde el primer decodificador de vídeo 72-1 y el segundo decodificador de vídeo 72-2, el generador de plano de vídeo 92 combina los datos de vídeo suministrados bajo el control del controlador 34 que refiere a pip_metadata. El generador de plano de vídeo 92 genera, entonces, un plano de vídeo que incluye una pantalla de presentación visual principal 1 y una pantalla de presentación visual secundaria 2, según se describe con referencia a la Figura 5 y suministra el plano de vídeo al procesador de datos de vídeo 96. Cuando los datos de vídeo desde solamente el primer decodificador de vídeo 72-1 se suministran al generador de plano de vídeo 92, el generador de plano de vídeo 92 genera un plano de vídeo utilizando los datos de vídeo suministrados y suministra el plano de vídeo generado al procesador de datos de vídeo 96. La combinación de dos elementos de datos de vídeo se denomina también, a modo de ejemplo, "mezcla" o "superposición",

El primer decodificador de audio 75-1 decodifica el flujo de audio y suministra los datos del flujo de vídeo decodificado al procesador de mezcla 101. El segundo decodificador de audio 75-2 decodifica el flujo de audio y suministra los datos del flujo de audio decodificado al procesador de mezcla 101.

En este caso, cuando el flujo de audio nº 1 y el flujo de audio nº 2 se superponen para la reproducción (cuando dos flujos de audio se seleccionan por el usuario como flujos de audio a reproducirse), el flujo de audio nº 1 decodificado por el primer decodificador de audio 75-1 y el flujo de audio nº 2 decodificado segundo decodificador de audio 75-2 se suministran al procesador de mezcla 101.

El procesador de mezcla 101 mezcla (superpone) los datos de audio desde el primer decodificador de audio 75-1 y los datos de audio desde el segundo decodificador de audio 75-2 y proporciona, a la salida, los datos resultantes al procesador de mezcla 97 en la etapa posterior. En la presente forma de realización, la mezcla (superposición) de la salida de datos de audio desde el primer decodificador de audio 75-1 y la salida de datos de audio desde el segundo decodificador de audio 75-2 se denomina también "combinación". El término "combinación" se refiere también a "mezclar dos elementos de datos de audio".

Los datos de sonido seleccionados por el conmutador 32 se suministran a la memoria intermedia 95 y es objeto de memorización. La memoria intermedia 95 suministra los datos de sonido al procesador de mezcla 97 en una temporización predeterminada. En este caso, los datos de sonido son datos de efectos acústicos resultantes de la selección de menú o de una procedencia similar. El procesador de mezcla 97 mezcla (superpone o combina) los datos de audio procesados por el procesador de mezcla 101 (es decir, los datos de audio resultantes de la mezcla de la salida de datos de audio desde el primer decodificador de audio 75-1 y la salida de datos de audio desde el segundo decodificador de audio 75-2) y los datos de sonido suministrados desde la memoria intermedia 95 y proporciona, a la salida, los datos resultantes como señales de audio.

Los datos objeto de lectura desde la memoria intermedia 54, que es una memoria intermedia de lectura de subtítulos de texto, se proporciona, a la salida, a la composición (decodificador) de subtítulos de texto 76 en la etapa posterior

en una temporización predeterminada. La composición de subtítulos de texto 76 decodifica los datos de Texto-ST y suministra los datos decodificados al conmutador 77.

El conmutador 77 selecciona uno de los flujos de gráficos de presentación decodificados por el decodificador de gráficos de presentación 73 y el Texto-ST (datos de subtítulos de texto) y suministra los datos seleccionados al generador de plano de gráficos de presentación 93. Es decir, una imagen de subtítulos simultáneamente suministrada al generador de plano de gráficos de presentación 93 se proporciona, a la salida, desde el decodificador de gráficos de presentación 73 o desde la composición de subtítulos de texto (Texto-ST) 76. El flujo de gráficos de presentación introducido simultáneamente al decodificador de gráficos de presentación 73 es un flujo separado del Clip principal o del Clip secundario (es decir, se selecciona por el conmutador 57). De este modo, la imagen de subtítulos proporcionada simultáneamente al generador de plano de gráficos de presentación 93 es la salida decodificada de los flujos de gráficos de presentación desde el Clip principal, el flujo de gráficos de presentación desde el Clip secundario o los datos de subtítulos de texto.

Sobre la base de los datos de imágenes de fondo suministrados desde el decodificador de fondo 71, el generador de plano de fondo 91 genera un plano de fondo, que sirve como, a modo de ejemplo, una imagen de papeles pintados cuando se visualiza una imagen de vídeo en un tamaño reducido y suministra el plano de fondo generado al procesador de datos de vídeo 96.

Sobre la base de los datos (el flujo de gráficos de presentación o datos de subtítulos de texto) seleccionados por el conmutador 77 y suministrados, el generador de plano de gráficos de presentación 93 genera un plano de gráficos de presentación, que sirve como, a modo de ejemplo, una imagen de presentación y suministra el plano de gráficos de presentación generado al procesador de datos de vídeo 96. Sobre la base de los datos de flujos de gráficos interactivos suministrados desde el decodificador de gráficos interactivos 74, el generador de plano de gráficos interactivos 94 genera un plano de gráficos interactivos y suministra el plano de gráficos interactivos generado al procesador de datos de vídeo 96.

El procesador de datos de vídeo 96 combina el plano de fondo desde el generador de plano de fondo 91, el plano de vídeo desde el generador de plano de vídeo 92, el plano de gráficos de presentación desde el generador de plano de gráficos de presentación 93 y el plano de gráficos interactivos desde el generador de plano de gráficos interactivos 94 y proporciona, a la salida, el plano combinado como señales de vídeo. El procesador de mezcla 97 mezcla (combina o superpone) los datos de audio desde el procesador de mezcla 101 (esto es, los datos de audio resultantes de la mezcla de los datos de audio decodificados por el primer decodificador de audio 75-1 y los datos de audio decodificados por el segundo decodificador de audio 75-2) y los datos de sonido suministrados desde la memoria intermedia 95 y proporciona, a la salida, los datos resultantes como señales de audio.

Los conmutadores 57 a 59 y el conmutador 77, realizan la conmutación sobre la base de una selección del usuario a través de la interfaz de usuario o en un fichero que contiene datos de interés. A modo de ejemplo, cuando solamente el fichero de flujo AV de Clip secundario contiene un flujo de audio, el conmutador 59 conmuta la conexión al lado secundario.

A continuación, el procesamiento de reproducción realizado por el aparato de reproducción 20, ilustrado en la Figura 35, se describirá haciendo referencia al diagrama de flujo ilustrado en la Figura 36. Este procesamiento se inicia cuando el usuario emite una instrucción para reproducir un flujo AV predeterminado por intermedio de la interfaz de usuario.

En la etapa S1, el controlador 34 efectúa la lectura de un fichero de PlayList, registrado en un soporte de registro o una HDD (una de disco duro), que no se ilustra, por intermedio de la unidad de memorización 31. A modo de ejemplo, el fichero de PlayList (xxxxx.mpls) anteriormente descrito con referencia a la Figura 7 es objeto de lectura.

En la etapa S2, el controlador 34 determina si existe, o no, un vídeo secundario en un flujo para reproducirse, sobre la base del fichero PlayList objeto de lectura.

Más concretamente, fuera de PlayItems en el fichero PlayList objeto de lectura, el controlador 34 realiza el bucle de vídeo secundario (el bucle 'for' de video_stream_id2) en STN_table() anteriormente descrito con referencia a las Figuras 33 y 34, para determinar, de este modo, si existe, o no, un vídeo secundario en el flujo que se va a reproducir.

Cuando se determina, en la etapa S2, que no existe un flujo de vídeo secundario, el procesamiento de reproducción del vídeo primario, que se describe a continuación con referencia a las Figuras 37 a 39, es realizado en la etapa S3.

En la etapa S4, el controlador 34 determina si está acabada, o no, la reproducción del flujo correspondiente al fichero PlayList objeto de lectura. Cuando se determina, en la etapa S4, que no está terminada la reproducción del flujo, el proceso retorna a la etapa S3 y se repite el procesamiento posterior. Cuando se determina que se está concluida la reproducción del flujo, finaliza el procesamiento.

Quando se determina, en la etapa S2, que existe un flujo de vídeo secundario, el controlador 34 efectúa la lectura de pip_metadata en la etapa S5. Solamente el pip_metadata asociado con el PlayItem puede guardarse en una memoria después de una determinación en cuanto a si un flujo de vídeo secundario está contenido, o no, se realiza en la manera anteriormente descrita. Como alternativa, pip_metadata puede ser también objeto de lectura simultánea cuando el fichero de PlayList es leído en la etapa S1, puesto que pip_metadata es parte de la lista PlayList, según se describió anteriormente con referencia a las Figuras 7 a 9.

Más concretamente, sobre la base, a modo de ejemplo, del PlayItem_id actual y del secondary_video_stream_id determinado a partir de STN_table(), el controlador 34 identifica a pip_metadata de interés y efectúa la lectura de los pip_metadata desde el fichero de PlayList.

En la etapa S6, sobre la base de lo establecido en la lectura de pip_metadata, el controlador 34 determina si el flujo secundario está, o no, sincronizado con el flujo primario (esto es, está sincronizado con la cronología del PlayItem de la ruta principal).

Más concretamente, a modo de ejemplo, cuando pip_metadata tiene la estructura de datos anteriormente descrita con referencia a la Figura 9, el controlador 34 puede determinar si el flujo secundario está sincronizado, o no, con el flujo primario, sobre la base del valor de pip_metadata_type descrito con anterioridad con referencia a la Figura 10. Cuando pip_metadata tiene la estructura de datos descrita anteriormente con referencia a la Figura 11, el controlador 34 puede determinar si el flujo secundario está sincronizado, o no, con el flujo primario haciendo referencia al indicador Synchronous_PIP_metadata_flag.

A modo de ejemplo, cuando pip_metadata tenga la estructura de datos anteriormente descrita con referencia a la Figura 12, el controlador 34 puede determinar si flujo secundario está sincronizado, o no, con el flujo primario, sobre la base de Subpath_type.

Quando pip_metadata tiene la estructura de datos descrita anteriormente con referencia a la Figura 9 ó 11, una determinación en cuanto a si el flujo secundario está sincronizado, o no, con el flujo primario puede realizarse sobre la base del parámetro Subpath_type, distinto a pip_metadata, como en la estructura de datos pip_metadata ilustrada en la Figura 12. Es decir, el controlador 34 puede determinar si el flujo secundario está sincronizado, o no, con el flujo primario, sobre la base de Subpath_type.

Cundo se determina en la etapa S6 que el flujo secundario está sincronizado con el flujo primario, el proceso prosigue con la etapa S7. A continuación, según se describió anteriormente con referencia a la Figura 17, el controlador 37 determina si el tiempo de presentación visual del flujo primario es, o no, SubPlayItem_IN_time que indica el principio del segmento de reproducción del flujo secundario.

Quando se determina en la etapa S7, que el tiempo de presentación visual del flujo primario no es SubPlayItem_IN_time, se ejecuta el procesamiento de reproducción vídeo primario, que se describe a continuación con referencia a las Figuras 37 a 39, en la etapa S8.

Por el contrario, cuando se determina, en la etapa S7, que el tiempo de presentación visual del flujo primario es SubPlayItem_IN_time, el procesamiento de reproducción de vídeo secundario, que se describe a continuación haciendo referencia a las Figuras 40 a 42, se ejecuta en la etapa S9.

En la etapa S10, el controlador 34 determina si el tiempo de presentación visual del flujo primario es, o no, SubPlayItem_OUT_time que indica el final del segmento de reproducción del flujo secundario. Cuando se determina, en la etapa S10, que el tiempo de presentación visual no es SubPlayItem_Out_time, el proceso retorna a la etapa S9 y se repite el procesamiento subsiguiente. Cuando se determina que el tiempo de presentación visual es SubPlayItem_OUT_time, en la etapa S11, el controlador 34 determina si está acabado, o no, el flujo correspondiente al fichero PlayList objeto de lectura.

Quando se determina, en la etapa S11, que la reproducción de flujo no está acabada, el proceso retorna a la etapa S7 y se repite el procesamiento subsiguiente. Cuando se determina que la reproducción del flujo está terminada, finaliza el procesamiento.

Por el contrario, cuando se determina, en la etapa S6, que el flujo secundario no está sincronizado con el flujo primario (esto es, está sincronizado con la línea de tiempos de un SubPlayItem de la ruta secundaria), el procesamiento de reproducción de vídeo primario, que se describe a continuación haciendo referencia a las Figuras 37 a 39, se ejecuta en la etapa S12, puesto que solamente el flujo de vídeo primario se visualiza hasta que el usuario emita una instrucción para visualizar el flujo de vídeo secundario.

En la etapa S13, el controlador 34 determina si, o no, una instrucción para visualizar el flujo de vídeo secundario se ha recibido desde el usuario.

Quando se determina, en la etapa S13, que no se ha recibido una instrucción para visualizar el flujo secundario, en

la etapa S14, el controlador 34 determina si está acabada, o no, la reproducción del flujo correspondiente al fichero de Playlist objeto de lectura. Cuando se determina, en la etapa S14, que la reproducción del flujo no está acabada, el proceso retorna a la etapa S12 y se repite el procesamiento posterior. Cuando se determina que está acabada la reproducción del flujo, finaliza el procesamiento.

5 Por el contrario, cuando se determina, en la etapa S13, que se ha recibido una instrucción para visualizar el flujo de vídeo secundario, el procesamiento de reproducción de vídeo primario y secundario, que se describe a continuación haciendo referencia a las Figuras 40 a 42, se ejecuta en la etapa S15.

10 En la etapa S16, el controlador 34 determina si, o no, una instrucción para terminar la presentación visual del flujo de vídeo secundario ha sido recibida desde el usuario. Cuando se determina, en la etapa S16, que se ha recibido una instrucción para terminar la presentación visual del flujo secundario, el proceso retorna a la etapa S12 y se repite el procesamiento subsiguiente.

15 Cuando se determina, en la etapa S16, que no se recibe una instrucción para acabar la presentación visual del flujo de vídeo secundario, en la etapa S17, el controlador 34 determina si se termina, o no, la reproducción del flujo correspondiente al fichero Playlist objeto de lectura. Cuando se determina, en la etapa S17, que no está terminada la reproducción del flujo, el proceso retorna a la etapa S15 y se repite el procesamiento subsiguiente. Cuando se determina que la reproducción del flujo está terminada, finaliza el procesamiento.

20 En este procesamiento, se realiza una determinación en cuanto a si existe, o no, un flujo secundario para la reproducción de un flujo correspondiente a un fichero Playlist objeto de lectura y cuando existe un flujo secundario, se realiza una determinación en cuanto a si el flujo secundario ha de visualizarse, o no, mientras esté siempre superpuesto sobre un flujo primario (una ruta principal), sobre la base de si el flujo secundario está, o no, sincronizado con el flujo primario. Cuando el flujo secundario no está sincronizado con el flujo primario (la ruta principal), se realiza una determinación en cuanto a si el flujo secundario ha de superponerse, o no, sobre el flujo primario, sobre la base de si se emite, o no, una instrucción para visualizar el flujo secundario desde el usuario o una instrucción para terminar la presentación visual del flujo secundario se emite por el usuario.

25 30 A continuación, el procesamiento de reproducción de vídeo primario, ejecutado en la etapa S3, S8, o S12 ilustrada en la Figura 36, se describirá haciendo referencia a los diagramas de flujo representados en las Figuras 37 a 39.

35 En la etapa S41, el controlador 34 efectúa la lectura del Clip principal, del Clip secundario y de los datos de subtítulos de texto (Text_ST data). Más concretamente, el controlador 34 efectúa la lectura del Clip principal, sobre la base de un PlayItem, que se describió anteriormente con referencia a la Figura 22 y está contenido en la Playlist anteriormente descrita con referencia a la Figura 18. El controlador 34 efectúa también la lectura del Clip secundario y de datos de subtítulos de texto, sobre la base de SubPlayItems, que se describen con referencia a la Figura 21 y se refieren por las rutas secundarias contenidas en la lista Playlist.

40 En la etapa S42, el controlador 34 controla el conmutador 32 con el fin de suministrar los datos leídos (el Clip principal, el Clip secundario y los datos de subtítulos de texto) a las memorias intermedias correspondientes 51 a 54. Más concretamente, el controlador 34 conmuta el conmutador 32 con el fin de suministrar datos de imágenes de fondo a la memoria intermedia 51, suministrar los datos del Clip principal a la memoria intermedia 52, suministrar los datos del Clip secundario a la memoria intermedia 53 y suministrar los datos de Texto-ST a la memoria intermedia 54.

45 50 En la etapa S43, bajo el control del controlador 34, el conmutador 32 realiza la conmutación. Como resultado, los datos de imagen de fondo se suministran a la memoria intermedia 51, los datos del Clip principal se suministran a la memoria intermedia 52, los datos del Clip secundario se suministran a la memoria intermedia 53 y los datos de subtítulos de texto se suministran a la memoria intermedia 54.

55 En la etapa S44, las memorias intermedias 51 a 54 memorizan los datos suministrados correspondientes. Más concretamente, la memoria intermedia 51 memoriza los datos de imágenes de fondo, la memoria intermedia 52 memoriza los datos del Clip principal, la memoria intermedia 53 memoriza los datos del Clip secundario y la memoria intermedia 54 memoriza los datos de Texto-ST.

En la etapa S45, la memoria intermedia 51 proporciona, a la salida, los datos de imágenes de fondo al decodificador de fondo 71.

60 En la etapa S46, la memoria intermedia 52 proporciona, a la salida, los datos de flujo del Clip principal al filtro de PID 55.

65 En la etapa S47, el filtro de PID 55 clasifica los datos de flujo para los decodificadores de flujos elementales correspondientes, sobre la base de los PIDs datos a los paquetes de TS contenidos en el fichero de flujo AV de Clip principal. Más concretamente, el filtro de PID 55 suministra un flujo de vídeo al primer decodificador de vídeo 72-1 a través del filtro de PID 60, suministra un flujo de gráficos de presentación al conmutador 57 sirviendo como una

fuelle de suministro para el decodificador de gráfcos de presentación 73, suministra un flujo de gráfcos interactivos al conmutador 58 que sirve como una fuente de suministro para el decodificador de gráfcos interactivos 74 y suministra un flujo de audio al conmutador 59 que sirve como una fuente de suministro para el primer decodificador de audio 75-1. Es decir, el flujo de vídeo, el flujo de gráfcos de presentación, el flujo de gráfcos interactivos y el flujo de audio son PIDs diferentes entre sí.

En la etapa S48, la memoria intermedia 53 proporciona los datos de flujo del Clip secundario al filtro de PID 56.

En la etapa S49, el filtro de PID 56 clasifica los datos de flujo para los correspondientes decodificadores de flujos elementales, sobre la base de PIDs. Más concretamente, el filtro de PID 56 suministra el flujo de vídeo suministrado al primer decodificador de vídeo 72-1 a través del filtro de PID 60, suministra un flujo de gráfcos de presentación al conmutador 57 que sirve como una fuente de suministro para el decodificador de gráfcos de presentación 73, suministra un flujo de gráfcos interactivos al conmutador 58 que sirve como una fuente de suministro para el decodificador de gráfcos interactivos 74 y suministra el flujo de audio al conmutador 59 que sirve como una fuente de suministro para el primer decodificador de audio 75-1 y el segundo decodificador de audio 75-2.

En la etapa S50, bajo el control del controlador 34 a través de la interfaz de usuario, los conmutadores 57 a 59 y el filtro de PID 60 en las etapas posteriores al filtro de PID 55 y al filtro de PID 56 seleccionan uno de entre el Clip principal y el Clip secundario. Más concretamente, el conmutador 57 selecciona el flujo de gráfcos de presentación del Clip principal suministrado desde el filtro de PID 55 o el flujo de gráfcos de presentación de Clip secundario y suministra el flujo de gráfcos de presentación seleccionado al decodificador de gráfcos de presentación 73 en la etapa subsiguiente. El conmutador 58 selecciona el flujo de gráfcos interactivos de Clip principal que se suministra desde el filtro de PID 55 o el flujo de gráfcos interactivos del Clip secundario y suministra el flujo de gráfcos interactivos seleccionado al decodificador de gráfcos interactivos 74 en la etapa subsiguiente. Además, el conmutador 59 selecciona el flujo de audio de Clip principal suministrado desde el filtro de PID 55 o el flujo de audio de Clip secundario suministrado desde el filtro de PID 56 (flujo de audio nº 1 en este caso, puesto que no se ha conmutado la señal de audio) y suministra el flujo de audio seleccionado al primer decodificador de audio 75-1 en la etapa subsiguiente. Cuando el usuario emite una instrucción de conmutación de audio, el conmutador 59 suministra el flujo de audio del Clip principal al segundo decodificador de audio 75-2 o suministra el flujo de audio del Clip secundario al primer decodificador de audio 75-1 o al segundo decodificador de audio 75-2. En este caso, puesto que se ha descrito ya el procesamiento de reproducción antes de que se realice la conmutación de audio, se omitirá aquí su descripción.

En la etapa S51, la memoria intermedia 54 proporciona, a la salida, los datos de subtítulos de texto a la composición de subtítulos de texto 76.

En la etapa S52, el decodificador de fondo 71 decodifica los datos de imágenes de fondo y proporciona, a la salida, los datos de imágenes de fondo decodificados al generador de plano de fondo 91.

En la etapa S53, el primer decodificador de vídeo 72-1 decodifica el flujo de vídeo (esto es, el flujo de vídeo primario suministrado) y proporciona el flujo de vídeo decodificado al generador de plano de vídeo 92.

En la etapa S54, el decodificador de gráfcos de presentación 73 decodifica el flujo de gráfcos de presentación seleccionado por el conmutador 57 y suministra, a la salida, el flujo de gráfcos de presentación decodificados al conmutador 77 en la etapa subsiguiente.

En la etapa S55, el decodificador de gráfcos interactivos 74 decodifica el flujo de gráfcos interactivos seleccionado por el conmutador 58 y se suministra y proporciona, a la salida, el flujo de gráfcos interactivos decodificado al generador de plano de gráfcos interactivos 94.

En la etapa S56, el primer decodificador de audio 75-1 decodifica el flujo de audio (flujo de audio nº 1) seleccionado por el conmutador 59 y suministrado, a la salida, el flujo de audio decodificado al procesador de mezcla 101 en la etapa subsiguiente. En el procesamiento de reproducción, cuando el usuario no ha emitido una instrucción para la conmutación de audio, no existe ninguna salida de datos de audio desde el segundo decodificador de audio 75-2. En consecuencia, el procesador de mezcla 101 suministra directamente datos de audio, objeto de salida desde el primer decodificador de audio 75-1, al procesador de mezcla 97 en la etapa subsiguiente.

En la etapa S57, la composición de Texto-ST 76 decodifica los datos de subtítulos de texto y proporciona, a la salida, los datos de subtítulos de texto decodificados al conmutador 77 en la etapa subsiguiente.

En la etapa S58, el conmutador 77 selecciona los datos desde el decodificador de gráfcos de presentación 73 o los datos procedentes de la composición de Texto-ST 76. Más concretamente, el conmutador 77 selecciona uno de entre el flujo de gráfcos de presentación decodificado por el decodificador de gráfcos de presentación 73 y el Texto-ST (datos de subtítulos de texto) y suministra los datos seleccionados al generador de plano de gráfcos de presentación 93.

En la etapa S59, el generador de plano de fondo 91 genera un plano de fondo, sobre la base de los datos de imágenes de fondo suministrados desde el decodificador de fondo 71.

5 En la etapa S60, el generador de plano de vídeo 92 genera un plano de vídeo, sobre la base de los datos de vídeo suministrados desde el primer decodificador de vídeo 72-1.

10 En la etapa S61, el generador de plano de gráficos de presentación 93 genera un plano de gráficos de presentación sobre la base de los datos procedentes del decodificador de gráficos de presentación 73 o los datos procedentes de la composición de Texto-ST 76, siendo los datos seleccionados por el conmutador 77 y suministrados en el procesamiento en la etapa S58.

En la etapa S62, el generador de plano de gráficos interactivos 94 genera un plano de gráficos interactivos, sobre la base de los datos del flujo de gráficos interactivos suministrados desde el decodificador de gráficos interactivos 74.

15 En la etapa S63, la memoria intermedia 95 memoriza los datos de sonidos seleccionados y suministrados en el procesamiento en la etapa S43 y suministra los datos memorizados al procesador de mezcla 97 en una temporización predeterminada.

20 En la etapa S64, el procesador de datos de vídeo 96 combina los datos de los planos individuales y proporciona, a la salida, los datos combinados. Más concretamente, el procesador de datos de vídeo 96 combina los datos procedentes del generador de plano de fondo 91, los datos procedentes del generador de plano de vídeo 92, los datos procedentes del generador de plano de gráficos de presentación 93 y los datos procedentes del generador de plano de gráficos interactivos 94 y proporciona, a la salida, los datos combinados como datos de vídeo.

25 En la etapa S65, el procesador de mezcla 97 mezcla (combina) los datos de audio (la salida de datos audio desde el procesador de mezcla 101) y los datos de sonido y proporciona, a la salida, los datos mezclados. El proceso retorna, entonces, a la etapa S3 ilustrada en la Figura 36 y prosigue con la etapa S4, retorna a la etapa S8 ilustrada en la Figura 36 y prosigue con la etapa S7, o retorna a la etapa S12 ilustrada en la Figura 30 y prosigue con la etapa S13.

30 En el procesamiento descrito con referencia a las Figuras 37 a 39, el Clip principal, el Clip secundario y los datos de subtítulos de texto se refieren por medio de la ruta principal y las rutas secundarias contenidas en la Playlist y son objeto de reproducción. En este caso, la señal de vídeo visualizada es solamente la señal de vídeo del flujo de vídeo primario y la pantalla de presentación visual secundaria 2, descrita con referencia a la Figura 5, no se visualiza. En esta configuración, la ruta principal y las rutas secundarias se proporcionan en la Playlist y un Clip que es diferente del fichero de flujo AV de Clip especificado por la ruta principal puede especificarse por la ruta secundaria. De este modo, los datos del Clip secundario, que es un Clip que es distinto del Clip principal especificado por el PlayItem de la ruta principal y los datos del Clip principal pueden reproducirse juntos (al mismo tiempo).

40 En las Figuras 37 a 39, el procesamiento en las etapas S45 y S46 puede ejecutarse en orden inverso o pueden ejecutarse en paralelo. El procesamiento en las etapas S47 y S49 puede ejecutarse también en orden inverso o pueden ejecutarse en paralelo. Además, el procesamiento en las etapas S52 a S57 puede ejecutarse también en orden inverso o pueden ejecutarse en paralelo. El procesamiento en las etapas S59 a S62 puede ejecutarse también en orden inverso o pueden ejecutarse en paralelo. El procesamiento en las etapas S64 y S65 puede ejecutarse también en orden inverso o puede ejecutarse en paralelo. Es decir, en la Figura 35, el procesamiento de las memorias intermedias 51 a 54, que están verticalmente en la misma jerarquía, puede ejecutarse en paralelo y puede ejecutarse en cualquier orden, mientras que el procesamiento de los conmutadores 57 a 59 puede ejecutarse en paralelo y puede ejecutarse en cualquier orden, el procesamiento de los decodificadores 71 a 76 puede ejecutarse en paralelo o puede ejecutarse en cualquier orden. El procesamiento de los generadores de planos 91 a 94 puede ejecutarse en paralelo y puede ejecutarse en cualquier orden y el procesamiento del procesador de vídeo 96 y del procesador de mezcla 97 puede ejecutarse en paralelo y puede ejecutarse en cualquier orden.

55 A continuación, el procesamiento de reproducción de vídeo primario y secundario, ejecutado en la etapa S9 o S15 que se ilustra en la Figura 36 se describirá haciendo referencia a los diagramas de flujo ilustrados en las Figuras 40 a 42.

El procesamiento que es esencialmente el mismo que el de las etapas S41 a S46 anteriormente descritas con referencia a las Figuras 37 y 38 se ejecuta en las etapas S101 a S106.

60 Es decir, el controlador 34 efectúa la lectura del Clip principal, del Clip secundario y de los datos de subtítulos de texto (Texto-ST) y controla el conmutador 32 con el fin de suministrar los datos leídos (el Clip principal, el Clip secundario y los datos de subtítulos de texto) a las memorias intermedias correspondientes 51 a 54. Bajo el control del controlador 34, el conmutador 32 suministra los datos de imágenes de fondo a la memoria intermedia 51, suministra los datos del Clip principal a la memoria intermedia 52, suministra los datos del Clip secundario a la memoria intermedia 53 y suministra los datos de subtítulos de texto a la memoria intermedia 54, de modo que los datos suministrados sean memorizados en las memorias intermedias correspondientes 51 a 54. La memoria intermedia 51 proporciona los datos de imágenes de fondo al decodificador de fondo 71. La memoria intermedia 52

proporciona, a la salida, los datos de flujo del Clip principal al filtro de PID 55.

En la etapa S107, el filtro de PID 55 clasifica los datos de flujo para los decodificadores de flujos elementales correspondientes, sobre la base de los PIDs datos a paquetes de TS contenido en el fichero de flujo AV de Clip principal. Más concretamente, el filtro de PID 55 suministra un flujo de vídeo al filtro de PID 60, suministra un flujo de gráficos de presentación al conmutador 57 que sirve como una fuente de suministro para el decodificador de gráficos de presentación 73, suministra un flujo de gráficos interactivos al conmutador 58 que sirve como una fuente de suministro para el decodificador de gráficos interactivos 74 y suministra el flujo de audio al conmutador 59 que sirve como una fuente de suministro para el primer decodificador de audio 75-1. Es decir, el flujo de vídeo, el flujo de gráficos de presentación, el flujo de gráficos interactivos y el flujo de audio son objeto de asignación de PIDs diferentes entre sí. Bajo el control del controlador 34, el filtro de PID 60 suministra el flujo de vídeo primario al primer decodificador de vídeo 72-1 y suministra el flujo de vídeo secundario al segundo decodificador de vídeo 72-2.

En la etapa S108, la memoria intermedia 53 proporciona los datos de flujo del Clip secundario al filtro de PID 56.

En la etapa S109, el filtro de PID 56 clasifica los datos de flujo para los decodificadores de flujos elementales correspondientes, sobre la base de PIDs. Más concretamente, el filtro de PID 56 suministra el flujo de vídeo predeterminado al filtro de PID 60, suministra un flujo de gráficos de presentación al conmutador 57 que sirve como una fuente de suministro para el decodificador de gráficos de presentación 73, suministra un flujo de gráficos interactivos al conmutador 58 que sirve como una fuente de suministro para el decodificador de gráficos interactivos 74 y suministra un flujo de audio al conmutador 59 que sirve como una fuente de suministro para el primer decodificador de audio 75-1 y el segundo decodificador de audio 75-2. Bajo el control del controlador 34, el filtro de PID 60 suministra el flujo de vídeo primario al primer decodificador de vídeo 72-1 y suministra el flujo de vídeo secundario al segundo decodificador de vídeo 72-2.

El procesamiento que es esencialmente el mismo que el realizado en las etapas S50 a S52 anteriormente descritas con referencia a las Figuras 37 y 38 se ejecuta en las etapas S110 a S112.

Es decir, bajo el control del controlador 34, los conmutadores 57 a 59 y el filtro de PID 60 seleccionan uno de entre el Clip principal y el Clip secundario. La memoria intermedia 54 proporciona los datos de subtítulos de texto a la composición de subtítulos de texto 76. El decodificador de fondo 71 decodifica los datos de imágenes de fondo y proporciona los datos de imágenes de fondo decodificados al generador de plano de fondo 91.

En la etapa S113, el primer decodificador de vídeo 72-1 decodifica el flujo de vídeo primario suministrado y proporciona, a la salida, el flujo de vídeo decodificado al generador de plano de vídeo 92.

En la etapa S114, el segundo decodificador de vídeo 72-2 decodifica el flujo de vídeo secundario suministrado y proporciona, a la salida, el flujo de vídeo secundario decodificado al generador de plano de vídeo 92.

En la etapa S115, el decodificador de gráficos de presentación 73 decodifica el flujo de gráficos de presentación seleccionado por el conmutador 57 y así suministrado y proporciona el flujo de gráficos de presentación decodificado al conmutador 77 en la etapa posterior.

En la etapa S116, el decodificador de gráficos interactivos 74 decodifica el flujo de gráficos interactivos seleccionado por el conmutador 58 y se suministra y proporciona, a la salida, el flujo de gráficos interactivos decodificado al generador de plano de gráficos interactivos 94 en la etapa posterior.

En la etapa S117, el primer decodificador de audio 75-1 decodifica el flujo de audio secundario seleccionado por el conmutador 59 y objeto de suministro y proporciona, a la salida, el flujo de audio primario decodificado al procesador de mezcla 101 en la etapa subsiguiente.

En la etapa S118, el segundo decodificador de audio 75-2 decodifica el flujo de audio secundario seleccionado por el conmutador 59 y suministrado y proporciona, a la salida, el flujo de audio secundario decodificado al procesador de mezcla 101 en la etapa posterior.

En la etapa S119 de los datos primarios o secundarios, la composición de Texto-ST 76 decodifica los datos de subtítulos de texto a visualizarse y proporciona, a la salida, los datos de subtítulos de texto decodificados al conmutador 77 en la etapa posterior.

En la etapa S120, el conmutador 77 selecciona los datos procedentes del decodificador de gráficos de presentación 73 o los datos procedentes de la composición de Texto-ST 76. Más concretamente, el conmutador 77 selecciona uno de entre el flujo de gráficos de presentación decodificado por el decodificador de gráficos de presentación 73 y el Texto-ST (datos de subtítulos de texto) y suministra los datos seleccionados al generador de plano de gráficos de presentación 93.

En la etapa S121, el generador de plano de fondo 91 genera un plano de fondo, sobre la base de los datos de

imágenes de fondo suministrados desde el decodificador de fondo 71.

En la etapa S122, bajo el control del controlador 34, que se refiere a pip_metadata, el generador de plano de vídeo 92 combina los datos de vídeo suministrados desde el primer decodificador de vídeo 72-1 y los datos de vídeo suministrados desde el segundo decodificador de vídeo 72-2 para generar un plano de vídeo que incluye una pantalla de presentación visual principal 1 y una pantalla de presentación visual secundaria 2, según se describe con referencia a la Figura 5 y suministra el plano de vídeo generado al procesador de datos de vídeo 96.

Más concretamente, bajo el control del controlador 34 que se refiere a pip_metadata descrito utilizando, a modo de ejemplo, la Figura 9, 11, o 12, el generador de plano de vídeo 92 combina el flujo de vídeo secundario posicionado y escalado con el flujo de vídeo primario para generar un plano de vídeo que incluye una pantalla de presentación visual principal 1 y una pantalla de presentación visual secundaria 2, según se describe con referencia a la Figura 5 y suministra el plano de vídeo generado al procesador de datos de vídeo 96.

Los detalles del posicionamiento y de la puesta en escala se describirán a continuación haciendo referencia a la Figura 43.

Según se ilustra en los pip_metadata anteriormente descritos con referencia a la Figura 9, 11, o 12, el parámetro pip_horizontal_position indica, a modo de ejemplo, en la pantalla de presentación visual principal 1, ilustrada en la Figura 5, una coordenada X en la esquina superior izquierda de la pantalla de presentación visual secundaria 2 en donde se visualiza la señal de vídeo secundaria y el parámetro pip_vertical_position indica, a modo de ejemplo, en la pantalla de presentación visual principal 1, ilustrada en la Figura 5, una coordenada Y en la esquina superior izquierda de la pantalla de presentación visual secundaria 2 en donde se visualiza la señal de vídeo secundaria. Además, pip_scale indica información con respecto a la magnitud de la pantalla de presentación visual secundaria 2 en donde se visualiza la señal de vídeo secundaria.

Es decir, según se ilustra en la Figura 43, una señal de vídeo secundaria en escala (scaled secondary video) se ajusta en un plano de vídeo primario (primary video plane) para tener un tamaño predeterminado, sobre la base del parámetro pip_scale y se posiciona de modo que la esquina superior izquierda del vídeo secundario en escala llegue a una posición indica por pip_horizontal_position en la dirección del eje X y pip_vertical_position en la dirección del eje Y en relación con la esquina superior izquierda del plano de vídeo primario.

En este caso, para "is_luma_key=1" en pip_metadata según se describe con referencia a la Figura 12, el generador de plano de vídeo 92 realiza la combinación de luma_keying en el flujo de vídeo primario y en el flujo de vídeo secundario para generar un plano de vídeo y suministra el plano de vídeo generado al procesador de datos de vídeo 96.

La "luma keying combination", en este caso es un sistema para superponer y combinar una imagen, desde la que se fijan partes no deseadas utilizando diferencias de componentes en brillo (valor de luminancia), con vídeo, según se describe en pip_metadata ilustrado en la Figura 12. A continuación, detalles de la combinación luma_keying se describirán haciendo referencia a las Figuras 44 y 45.

La Figura 44 es una vista que ilustra una señal de vídeo primaria (primary video) y una señal de vídeo secundaria (secondary video) antes de la combinación de luminancia denominada *luma_keying*.

Cuando la combinación de luma_keying se realiza en la señal de vídeo primaria en el lado derecho en la Figura y el vídeo secundario en el lado izquierdo, un valor de luminancia de vídeo secundario incluido en un margen de un límite inferior del valor de luminancia indicado por lower_limit_luma_key a un límite superior del valor de luminancia indicado por upper_limit_luma_key, según se describe utilizando pip_metadata ilustrado en la Figura 12, se realiza para tener un valor para la transferencia y la señal de vídeo secundaria resultante se combina con la señal de vídeo primaria.

Es decir, una señal de vídeo secundaria de la que se elimina una gama del límite inferior del valor de luminancia definido por lower_limit_luma_key y el límite superior del valor de luminancia definido por upper_limit_luma_key se superpone y combina con la señal de vídeo primaria. Más concretamente, según se ilustra en la Figura 44, una zona marcada distinta a un paralelogramo y un círculo en la señal de vídeo secundaria del lado izquierdo tiene un valor de luminancia en el margen del valor inferior de luminancia y su valor superior y en consecuencia, es transparente. La señal de vídeo secundaria del lado izquierdo, procesada de dicha manera, se superpone y se combina con la señal de vídeo primaria del lado derecho.

La Figura 45 es una vista que ilustra la señal de vídeo primaria y la señal de vídeo secundaria después de la combinación de luminancia *luma_keying*.

En la señal de vídeo primaria y en la señal de vídeo secundaria, después de la combinación de *luma_keying*, el área distinta al paralelogramo y al círculo en la señal de vídeo secundaria es transparente, según se ilustra en la Figura 45. De este modo, del área de la señal de vídeo secundaria, solamente las áreas del paralelogramo y del círculo se

combinan con la señal de vídeo primaria.

Aunque el área de la señal de vídeo secundaria se expresa por una línea de puntos para facilidad de ilustración en la Figura 45, la línea de puntos no se visualiza en la práctica.

Haciendo referencia de nuevo al diagrama de flujo ilustrado en la Figura 42, en la etapa S123, el generador de plano de gráficos de presentación 93 genera un plano de gráficos de presentación, sobre la base de los datos procedentes del decodificador de gráficos de presentación 73 o los datos procedentes de la composición de Texto-ST 76, seleccionándose los datos por el conmutador 77 y suministrados en el procesamiento en la etapa S58.

En la etapa S124, el generador de plano de gráficos interactivos 94 genera un plano de gráficos interactivos, sobre la base de los datos del flujo de gráficos interactivos suministrado desde el decodificador de gráficos interactivos 74.

En la etapa S125, la memoria intermedia 95 memoriza los datos de sonido seleccionados y suministrados en el procesamiento en la etapa S43 y suministra los datos memorizados al procesador de mezcla 97 en una temporización predeterminada.

En la etapa S126, el procesador de datos de vídeo 96 combina los datos de los planos individuales y proporciona, a la salida, los datos combinados. Más concretamente, el procesador de datos de vídeo 96 combina los datos procedentes del generador de plano de fondo 91, los datos procedentes del generador de plano de vídeo 92, los datos procedentes del generador de plano de gráficos de presentación 93 y los datos procedentes del generador de plano de gráficos interactivos 94 y proporciona, a la salida, los datos combinados como datos de vídeo.

En la etapa S127, el procesador de mezcla 101 combina la salida de datos de audio primario desde el primer decodificador de audio 75-1 y la salida de datos de audio secundarios desde el segundo decodificador de audio 75-2 y suministra los datos combinados al procesador de mezcla 97.

En la etapa S128, el procesador de mezcla 97 mezcla (combina) los datos de audio, combinados por el procesador de mezcla 101 y a su salida y los datos de sonido, proporcionado, a la salida, los datos resultantes. El proceso retorna luego a la etapa S9 ilustrada en la Figura 36 y prosigue con la etapa S10 o retorna a la etapa S15 ilustrada en la Figura 36 y prosigue con la etapa S16.

En el procesamiento descrito con referencia a las Figuras 40 a 42, el Clip principal, el Clip secundario y los datos de subtítulos de texto se refieren por la ruta principal y las rutas secundarias contenidas en la Playlist y son objeto de reproducción. En esta configuración, la ruta principal y las rutas secundarias se proporcionan en la Playlist y un Clip que es diferente de un fichero de flujo AV de Clip especificado por la ruta principal puede especificarse por la ruta secundaria. De este modo, la imagen visualizada de un flujo de vídeo secundario puede superponerse sobre la imagen visualizada de un flujo de vídeo primario.

En este caso, el tamaño y la posición en pantalla de la imagen visualizada del flujo de vídeo secundario pueden establecerse. De este modo, en comparación con un caso en el que la posición en pantalla y el tamaño de la presentación visual de la pantalla de presentación visual secundaria son preestablecidas con el fin de depender de un aparato de reproducción, el flujo de vídeo secundario puede visualizarse en una posición y en un tamaño que no constituye una obstrucción para la visualización del flujo de vídeo primario, dependiendo del contenido o de la temporización de la presentación visual. Como alternativa, en conformidad con los contenidos del flujo de vídeo secundario, a modo de ejemplo, para contenidos importantes, el flujo de vídeo secundario puede visualizarse en una posición distinta y en un mayor tamaño, o para contenidos no importantes, el flujo de vídeo secundario puede visualizarse en un tamaño pequeño en un borde de la pantalla de presentación visual principal 1.

El tamaño y la posición en pantalla de la imagen visualizada del flujo de vídeo secundario pueden definirse adecuadamente por el productor de contenidos o el distribuidor de contenidos.

En las Figuras 40 a 42, el procesamiento en las etapas S105 y S106 puede realizarse en orden inverso o puede realizarse en paralelo. El procesamiento en las etapas S107 y S109 puede realizarse también en orden inverso o puede realizarse en paralelo. Además, el procesamiento en las etapas S112 a S119 puede realizarse en orden inverso o puede realizarse en paralelo. El procesamiento en las etapas S121 a S124 puede realizarse en orden inverso o puede realizarse en paralelo. Además, el procesamiento en la etapa S126 y el procesamiento en las etapas S127 y S128 pueden realizarse en orden inverso o puede realizarse en paralelo. Es decir, en la Figura 35, el procesamiento de las memorias intermedias 51 a 54, que están verticalmente en la misma jerarquía, puede realizarse en paralelo y puede realizarse en cualquier orden, el procesamiento de los conmutadores 57 a 59 puede realizarse en paralelo y puede realizarse en cualquier orden. El procesamiento de los decodificadores 71 a 76 puede realizarse en paralelo y puede realizarse en cualquier orden. El procesamiento de los generadores de planos 91 a 94 puede realizarse en paralelo y puede realizarse en cualquier orden y el procesamiento del procesador de vídeo 96 y el procesador de mezcla 97 puede realizarse en paralelo y pueden ejecutarse en cualquier orden.

El procesamiento anteriormente descrito puede resumirse como sigue.

El aparato de reproducción 20 obtiene una lista PlayList, que sirve como información de control de la reproducción que incluye la información de una ruta principal, que es una ruta de reproducción principal que indica la posición de un fichero de flujo AV de Clip principal que contiene al menos un flujo y la información de una ruta secundaria, que es una ruta de reproducción de un fichero de flujo AV de Clip secundario que ha de reproducirse mediante una ruta de reproducción que es distinta de la ruta de reproducción del fichero de flujo AV de Clip principal. El aparato de reproducción 20 recibe flujos seleccionados que han de reproducirse, sobre la base de STN_table() para seleccionar los flujos que están contenidos en la PlayList y que han de reproducirse. STN_table() contiene información para seleccionar un tipo de flujo predeterminado (p.e., un flujo de vídeo primario) de un fichero de flujo AV de Clip principal y para seleccionar otro fichero de flujo a reproducirse en conjunción con o sin sincronización con al temporización de reproducción del tipo predeterminado de flujo. De este modo, sobre la base de STN_table(), es posible recibir flujos seleccionados para su reproducción.

Cuando Subpath_type de PlayList es In-Mux, una ruta principal y una ruta secundaria pueden referirse al mismo Clip y una ruta secundaria puede añadirse como elemento adicional. Lo que antecede puede proporcionar un flujo con posibilidad de expansión. Puesto que se utiliza la configuración en la que múltiples ficheros pueden referirse mediante una sola ruta secundaria, (p.e., según se ilustra en la Figura 4), el usuario puede realizar la selección desde múltiples flujos diferentes.

Además, el parámetro STN_table() ilustrado en la Figura 23 y en las Figuras 33 y 34 se proporciona en un PlayItem de una ruta principal como una tabla que define datos auxiliares (p.e., un flujo de audio) multiplexado (contenido) en un fichero de flujo AV referido por la ruta principal y datos auxiliares referidos por una ruta secundaria. Lo que antecede hace posible conseguir un flujo con alta capacidad de expansión. Además, la introducción de información en STN_table() permite la fácil expansión de una ruta secundaria.

Además, el atributo stream_attribute(), que es información de atributo de flujo y se ilustra en la Figura 25, se proporciona en STM_table(); por lo tanto, el lado del aparato de reproducción 20 puede determinar si, o no, puede reproducirse un flujo seleccionado. Además, haciendo referencia a stream_attribute() se hace posible seleccionar y reproducir solamente un flujo que pueda reproducirse.

Además, STN_table() (Figura 23 y Figuras 33 y 34) se adapta para definir dos ficheros de flujo que tienen el mismo tipo (en este caso, un fichero de flujo de vídeo primario y un fichero de flujo de vídeo secundario o un fichero de flujo de audio primario y un fichero de flujo de audio secundario). Además, el aparato de reproducción 20 ilustrado en la Figura 35 está provisto de dos decodificadores de vídeo (el primer decodificador de vídeo 72-1 y el segundo decodificador de vídeo 72-2) y dos decodificadores de audio (el primer decodificador de audio 75-1 y el segundo decodificador de audio 75-2). Además, el generador de plano de vídeo 92 se proporciona para combinar (mezclar) datos de vídeos primarios y datos de vídeos secundarios que se decodifican por dos decodificadores de vídeo y el procesador de mezcla 101 se proporciona para combinar los datos de audio decodificados por dos decodificadores de audio. Con esta disposición operativa, dos flujos que tienen el mismo tipo se pueden combinar y reproducir al mismo tiempo.

Es decir, según se ilustra en la Figura 5, el aparato de reproducción según la presente invención puede realizar la así denominada "visualización PinP (*picture-in-picture*)", en donde un flujo de vídeo primario se visualiza en una pantalla principal y un flujo secundario se combina en ella y se visualiza como una pantalla auxiliar dentro de la pantalla principal.

En este caso, el tamaño y la posición en pantalla de la imagen de visualización del flujo de vídeo secundario, durante la presentación visual *picture-in-picture*, puede establecerse utilizando pip_metadata. Con esta disposición operativa, en comparación con un caso en el que la posición en pantalla y el tamaño de la visualización de la pantalla de presentación visual secundaria se preestablecen con el fin de depender de un aparato de reproducción, el flujo de vídeo secundario puede visualizarse en una posición y en unas dimensiones que no representen una obstrucción para la presentación visual del flujo de vídeo primario, dependiendo del contenido o de la temporización de la presentación visual. Como alternativa, en conformidad con el contenido del flujo de vídeo secundario, a modo de ejemplo, para contenidos importantes. El flujo de vídeo secundario puede visualizarse en una posición distinta y en un mayor tamaño o para contenidos no importantes, el flujo de vídeo secundario puede visualizarse en un pequeño tamaño en un borde de la pantalla de presentación visual principal 1.

El tamaño y la posición de visualización de la imagen de presentación visual del flujo de vídeo secundario pueden definirse adecuadamente por el productor de contenidos o el distribuidor de contenidos.

Además, pip_metadata puede proporcionar información que indique el establecimiento de la presentación visual de *picture-in-picture*, que no sea el tamaño y la posición en pantalla de la imagen de visualización del flujo de vídeo secundario. Pip_metadata puede proporcionar no solamente información que indica la posición y el tamaño de la pantalla de presentación visual secundaria 2 en donde se visualiza el flujo de vídeo secundario, sino también información con respecto, a modo de ejemplo, a la designación de color y la rotación de la señal de vídeo visualizada en relación con la señal de vídeo original del flujo de vídeo secundario.

En este caso, la información relacionada con la presentación visual, tal como el tamaño y la posición en pantalla de la imagen visualizada del flujo de vídeo secundario no están contenidas en el flujo de vídeo secundario y se describe en una PlayList para controlar la reproducción del flujo de vídeo secundario. Es decir, cuando se desea cambiar solamente el formato de presentación visual del flujo secundario sin realizar cambios en el flujo de vídeo secundario (p.e., cuando se desea cambiar solamente la posición en pantalla y/o el tamaño de la presentación visual del flujo de vídeo secundario), es suficiente actualizar solamente la declaración de pip_metadata en la lista de reproducción sin realizar cambios en el establecimiento de configuración del aparato de reproducción o del flujo de vídeo secundario.

Además, para el primer contenido y el segundo contenido, el mismo flujo de vídeo secundario puede visualizarse fácilmente por diferentes métodos de presentación visual (p.e., el mismo flujo de vídeo secundario puede visualizarse en un gran tamaño para el primer contenido y en un tamaño pequeño para el segundo contenido). Es decir, para cada uno del primer contenido y del segundo contenido, información que indica un formato de presentación visual deseado puede establecerse en pip_metadata en la lista de reproducción sin modificar los datos del flujo de vídeo secundario. Con esta disposición operativa, para el primer contenido y para el segundo contenido, el mismo flujo de vídeo secundario puede visualizarse en diferentes métodos de presentación visual deseados por el productor de contenidos o el distribuidor de contenidos.

Además, puesto que la información relacionada con la presentación visual, tal como el tamaño y la posición en pantalla de la imagen visualizada del flujo de vídeo secundaria, se establece en la lista de reproducción que contiene información de control de reproducción detallada, tal como información temporal, la presentación visual de la señal de vídeo secundaria puede establecerse en detalle.

Según se ilustra en la Figura 20, una ruta secundaria que contiene SubPath_type que indica los tipos de rutas secundarias (una calase tal como subtítulos de texto o audio y un tipo que indique si la ruta secundaria está, o no, sincronizada con una ruta principal), Clip_Information_file_name que se ilustra en la Figura 21 y que indica un nombre de un Clip secundario referido por la ruta secundaria, SubPlayItem_IN_time que se ilustra en la Figura 21 y que indica el punto de entrada IN-point de un Clip referido por la ruta secundaria y SubPlayItem_OUT_time que se ilustra en la Figura 21 y que indica el punto de salida OUT-point del Clip. De este modo, es posible identificar correctamente los datos referidos por medio de la ruta secundaria.

Los datos leídos por la unidad de memorización 31, ilustrada en la Figura 35, pueden ser datos registrados en un soporte de registro, tal como un DVD (disco versátil digital), pueden ser datos registrados en un disco duro, pueden ser datos descargados a través de una red, que no se ilustra, o pueden ser datos resultantes de una de sus combinaciones. A modo de ejemplo, la reproducción puede realizarse sobre la base de una PlayList y de un Clip secundario descargado y registrado en un disco duro y un fichero de flujo AV de Clip principal registrado en un DVD. Además, a modo de ejemplo, cuando una PlayList y un Clip principal que utilizan, como un Clip secundario, un fichero de flujo AV de Clip registrado en un DVD se registran en un disco duro, el Clip principal y el Clip secundario pueden ser objeto de lectura desde el disco duro y desde el DVD, respectivamente, para su reproducción, sobre la base de la PlayList registrada en el disco duro.

La configuración puede diseñarse de modo que la posición y/o el tamaño de la señal de vídeo secundaria puedan cambiarse por la función del aparato de reproducción 20 (reproductor). En tal caso, el tamaño, etc., previsto por un autor no puede obtenerse en algunos casos. Por el contrario, como en la presente invención, el tamaño, la posición, etc., puede controlarse mediante el uso de la PlayList, que es información de control. Con esta disposición operativa, a modo de ejemplo, incluso cuando se desee cambiar la posición, etc., de la señal de vídeo secundaria después de obtener un disco ROM es posible cambiar la posición, etc., prevista por el autor obteniendo solamente la lista de reproducción que tenga un tamaño relativamente pequeño sin necesidad de realizar cambios en el Clip actual.

A continuación, haciendo referencia a las Figuras 46 y 47, un método para la fabricación de un soporte de registro 21 en el que se registren datos que puedan reproducir datos por el aparato de reproducción 20 se describirá en conjunción con una realización, a modo de ejemplo, en donde el soporte de registro 21 sea un soporte de registro en disco.

Es decir, según se ilustra en la Figura 46, una placa base fabricada de vidrio o de material similar es objeto de preparación y se aplica material de registro, tal como la resina protectora denominada *fotoresist* a la placa base, con lo que se fabrica una unidad maestra de registro.

En adelante, según se ilustra en la Figura 47, en una sección de procesamiento de producción de software, datos de vídeo que fueron codificados por un dispositivo de codificación (un codificador de vídeo) y que tienen un formato que puede reproducirse por el aparato de reproducción 20 se memorizan temporalmente en una memoria intermedia y los datos de audio codificados por un codificador de audio se guardan temporalmente en una memoria intermedia. Además, datos no en flujo (p.e., índices, una PlayList o PlayLists y PlayItem) decodificados por un decodificador de datos, se guardan temporalmente en una memoria intermedia. Los datos de vídeo, los datos de audio y los datos no en flujo, que se guardan en las memorias intermedias respectivas, son objeto de multiplexación con señales de sincronización por un multiplexor (MPX) y un código de corrección de errores se añade a los datos multiplexados

mediante un circuito de control y corrección de errores (ECC). Los datos resultantes se someten a una modulación dada por un circuito de modulación (MOD) y se registran temporalmente en, a modo de ejemplo, una unidad de cinta magnética en conformidad con un formato predeterminado. De este modo, se produce programa informático que puede reproducirse por el aparato de reproducción 20 y que ha de registrarse sobre el soporte de registro 21.

El programa informático se edita (pre-masteriza) cuando sea necesario, y se generan señales que tienen un formato a registrarse en el disco óptico. Según se ilustra en la Figura 46, un haz de láser se modula en conformidad con las señales de registro y el haz de láser modulado se hace incidir sobre la capa fotorresistente de *photoresist* en la unidad maestra. En consecuencia, la capa de *photoresist*, en la unidad maestra, se expone en respuesta a las señales de registro.

En adelante, esta unidad maestra se desarrolla, de modo que aparezcan pequeños receptáculos en dicha unidad maestra. La unidad maestra preparada de esta manera se somete a procesamiento, tal como electroconformación, para fabricar una unidad maestra metálica a la que se transfieren dichos pequeños receptáculos sobre el soporte maestro de vidrio. Sobre la base de la unidad maestra metálica, se fabrica además un estampador metálico, que sirve como una matriz de moldeo.

A modo de ejemplo, un material, tal como MMA (acrílico) o PC (policarbonato), se inyecta en la matriz de moldeo y se solidifica. Como alternativa, se aplica una resina 2P (resina curable por radiación ultravioleta) al estampador metálico y luego se ilumina con luz ultravioleta y se solidifica. Esto permite a los pocillos, en el dispositivo estampador metálico, transferirse a una réplica de resina.

Una película reflectante se forma sobre la réplica, creada según se describió con anterioridad, por deposición por vapor, pulverización o un proceso similar. Como alternativa, una película reflectante se forma sobre la réplica fabricada mediante recubrimiento giratorio.

Más adelante se procesa la circunferencia exterior y la circunferencia interior del disco y el disco resultante se somete a un proceso necesario, tal como laminación de dos discos juntos. Además, una etiqueta y un concentrador se incorporan al disco resultante y el disco se inserta en un cartucho. Con esta operación se completa el soporte de registro 21 sobre el que pueden reproducirse datos por el aparato de reproducción 20, que se registran.

La serie anteriormente descrita del procesamiento puede realizarse mediante equipos físicos y puede realizarse también por programas informáticos. En este caso, el procesamiento anteriormente descrito se ejecuta por un ordenador personal 500 según se ilustra en la Figura 48.

En la Figura 48, una CPU (unidad central de procesamiento) 501 realiza varios tipos de procesamiento en conformidad con un programa almacenado en una memoria ROM (memoria de solamente lectura) 502 o un programa cargado desde una unidad de memorización 508 en una memoria RAM (memoria de acceso aleatorio) 503. Cuando sea necesario, la memoria RAM 503 memoriza también, a modo de ejemplo, datos requeridos para la unidad CPU 501 para ejecutar varios tipos de procesamiento.

La unidad CPU 501, la memoria ROM 502 y la memoria RAM 503 están interconectadas a través de un bus interno 504. Una interfaz de entrada/salida 505 se conecta también al bus interno 504.

Una unidad de entrada 506 que incluye un teclado y un ratón, una unidad de salida 507 que incluye una pantalla y un altavoz, la unidad de memorización 508, que incluye un disco duro y una unidad de comunicación 509 que incluye un módem o adaptador de terminales están en conexión con la interfaz de entrada/salida 505. Realizaciones, a modo de ejemplo, de la presentación visual incluyen una pantalla de CRT (tubo de rayos catódicos) y una pantalla LCD (pantalla de cristal líquido). La unidad de comunicación 509 realiza el procesamiento de comunicación a través de varias redes que incluyen una línea telefónica y una red de televisión CATV.

Una unidad 510 está también conectada a la interfaz de entrada/salida 505, cuando sea necesario, y, a modo de ejemplo, un soporte extraíble 521, tal como un disco magnético, un disco óptico, un disco magneto-óptico o una memoria de semiconductores, está en conexión con la unidad 510, cuando sea adecuado. Un programa informático objeto de lectura desde el soporte extraíble 521 se instala en la unidad de memorización 508, cuando sea necesario.

Cuando la serie de procesamiento ha de ejecutarse por software, un programa incluido en el software se instala a través de la red o de un medio de memorización de programas.

Este medio de memorización de programas puede no solamente ser un soporte de paquetes que se distribuye, por separado, desde el ordenador para suministrar un programa a un usuario y que incluye el soporte extraíble 521 sobre el que se registra el programa, sino también puede ser un disco duro, o similar, que se suministre al usuario mientras se incorpora en el cuerpo principal del aparato y que incluye la memoria ROM 502 o la unidad de memorización 508 en la que se registra el programa.

Las formas de realización de la presente invención no están limitadas a las anteriormente descritas y se pueden

realizar varios cambios de las mismas sin desviarse por ello del alcance de protección de la presente invención.

REIVINDICACIONES

1. Un aparato de reproducción que comprende:

5 un medio de obtención (32, 34) para obtener información de control de reproducción que comprende una primera información que contiene una ruta de reproducción principal que indica una posición en una cronología de un fichero de flujo principal que contiene al menos un flujo de vídeo y una segunda información que incluye una ruta de reproducción secundaria de un fichero de flujo secundario que es diferente del fichero de flujo principal;

10 un medio de lectura (32, 55, 56) para la lectura del fichero de flujo principal y del fichero de flujo secundario que han de reproducirse sobre la base de la información de control de reproducción obtenida por el medio de obtención (32, 34) y

15 un medio de combinación de vídeo (92) para combinar la señal de vídeo del fichero de flujo principal y la señal de vídeo del fichero de flujo secundario, siendo el fichero de flujo principal y el fichero de flujo secundario objeto de lectura por el medio de lectura (32, 55, 56), sobre la base de la información de control de reproducción obtenida por el medio de obtención (32, 34), en donde la información de control de reproducción incluye información relacionada con la presentación de la señal de vídeo del fichero de flujo secundario en una unidad de presentación visual, siendo la señal de vídeo del fichero de flujo secundario combinada con la señal de vídeo del fichero de flujo principal y

20 el medio de combinación de vídeo (92) está adaptado para combinar la señal de vídeo del fichero de flujo principal y la señal de vídeo del fichero de flujo secundario, sobre la base de la información de presentación visual incluida en la información de control de reproducción, en donde la información de presentación visual incluye información relacionada con el procesamiento para combinar la señal de vídeo del fichero de flujo principal y la señal de vídeo del fichero de flujo secundario, realizándose el procesamiento sobre la base de la información que indica un brillo de la señal de vídeo del fichero de flujo secundario, siendo la señal de vídeo del fichero de flujo secundario objeto de combinación con la señal de vídeo del fichero de flujo principal, en donde la información que indica el brillo de la señal de vídeo del fichero de flujo secundario es al menos un valor umbral para la información que indica un brillo para el procesamiento de transparencia y en una primera alternativa operativa, el procesamiento de transparencia se realiza en una parte de vídeo en donde el brillo de la señal de vídeo del fichero de flujo secundario es mayor o igual que el valor umbral y el procesamiento de transparencia no se realiza sobre una parte de vídeo en donde el brillo de la señal de vídeo del fichero de flujo secundario es menor que el umbral o en una segunda alternativa operativa, el procesamiento de transparencia se realiza sobre una parte de vídeo en donde el brillo de la señal de vídeo del fichero de flujo secundario es menor o igual al valor umbral y el procesamiento de transparencia no se realiza sobre una parte de vídeo en donde el brillo de la señal de vídeo del fichero de flujo secundario es mayor que el valor umbral.

2. El aparato de reproducción según la reivindicación 1, en donde la información de presentación visual incluye información relacionada con una posición de presentación visual de la señal de vídeo del fichero de flujo secundario y el medio de combinación de vídeo (92) está adaptado para combinar la señal de vídeo del fichero de flujo secundario con la señal de vídeo del fichero de flujo principal.

3. El aparato de reproducción según la reivindicación 1, en donde la información de presentación visual incluye información relacionada con una magnitud de presentación visual de la señal de vídeo del fichero de flujo secundario, siendo la señal de vídeo del fichero de flujo secundario objeto de combinación con la señal de vídeo del fichero de flujo principal.

4. El aparato de reproducción según la reivindicación 1, en donde la información de presentación visual incluye información relacionada con la designación de color de la señal de vídeo del fichero de flujo secundario, siendo la señal de vídeo del fichero de flujo secundario objeto de combinación con la señal de vídeo del fichero de flujo principal.

5. El aparato de reproducción según la reivindicación 1, en donde la información de presentación visual incluye información relacionada, con un ángulo de rotación de post-combinación de la señal de vídeo del fichero de flujo secundario en relación con la señal de vídeo original, siendo la señal de vídeo del fichero de flujo secundario objeto de combinación con la señal de vídeo del fichero de flujo principal.

6. El aparato de reproducción según la reivindicación 1, que comprende, además, un medio de entrada de operación para recibir una entrada de operación de un usuario,

60 en donde la información de control de reproducción obtenida por el medio de obtención (32, 34) incluye información de sincronización que indica si el fichero de flujo secundario se reproduce siempre simultáneamente con el fichero de flujo principal o el fichero de flujo secundario se reproduce simultáneamente con el fichero de flujo principal solamente cuando se emite una instrucción desde el usuario y

65 el medio de combinación de vídeo (92) está adaptado para combinar, cuando la información de sincronización indica

que el fichero de flujo secundario se reproduce simultáneamente con el fichero de flujo principal solamente cuando se emite una instrucción desde el usuario, solamente a la recepción de una instrucción para reproducir el fichero de flujo secundario desde el usuario, la señal de vídeo del fichero de flujo principal y la señal de vídeo del fichero de flujo secundario, sobre la base de la entrada de operación del usuario recibida por el medio de entrada de operación.

5 7. El aparato de reproducción según la reivindicación 1, que comprende, además, un medio de combinación de audio (101) para combinar la señal de audio a reproducirse en asociación con el fichero de flujo principal y la señal de audio a reproducirse en asociación con el fichero de flujo secundario,

10 en donde el medio de combinación de audio (101) combina la señal de audio a reproducirse en asociación con el fichero de flujo principal y la señal de audio a reproducirse en asociación con el fichero de flujo secundario, cuando el medio de combinación de vídeo (92) combina la señal de vídeo del fichero de flujo principal y la señal de vídeo del fichero de flujo secundario.

15 8. El aparato de reproducción según la reivindicación 1, en donde la información de control de reproducción incluye información que indica que está incluida la información de presentación visual.

20 9. Un método de reproducción para un aparato de reproducción para reproducir un fichero de flujo principal y un fichero de flujo secundario, comprendiendo el método de reproducción las etapas de:

la lectura (32, 34) de información de control de reproducción que comprende una primera información que incluye una ruta de reproducción principal que indica una posición en una cronología del fichero de flujo principal que contiene al menos un flujo de vídeo y una segunda información que incluye una ruta de reproducción secundaria del fichero de flujo secundario que es diferente del fichero de flujo principal;

25 la lectura (32, 55, 56) del fichero de flujo principal y del fichero de flujo secundario que han de reproducirse, sobre la base de la información de control de reproducción objeto de lectura y

30 la combinación (92) de la señal de vídeo del fichero de flujo principal leído y de la señal de vídeo del fichero de flujo secundario leído, sobre la base de la información que está incluida en la información de control de reproducción y que está relacionada con la presentación de la señal de vídeo del fichero de flujo secundario en una pantalla, siendo la señal de vídeo del fichero de flujo secundario objeto de combinación con la señal de vídeo del fichero de flujo principal, en donde la información de presentación visual incluye información relacionada con el procesamiento para combinar la señal de vídeo del fichero de flujo principal y la señal de vídeo del fichero de flujo secundario, realizándose el procesamiento sobre la base de la información que indica un brillo de la señal de vídeo del fichero de flujo secundario, siendo la señal de vídeo del fichero de flujo secundario objeto de combinación por la señal de vídeo del fichero de flujo principal, en donde la información que indica el brillo de la señal de vídeo del fichero de flujo secundario es al menos un valor umbral para la información que indica un brillo para procesamiento de transparencia y en una primera alternativa operativa, el procesamiento de transparencia se realiza sobre una parte de vídeo en donde el brillo de la señal de vídeo del fichero de flujo secundario es mayor o igual al valor umbral y el procesamiento de transparencia no se realiza sobre una parte de vídeo en donde el brillo de la señal de vídeo del fichero de flujo secundario es menor que el valor umbral o en una segunda alternativa operativa, el procesamiento de transparencia se realiza sobre una parte de vídeo en donde el brillo de la señal de vídeo del fichero de flujo secundario es menor o igual al valor umbral y el procesamiento de transparencia no se realiza sobre una parte de vídeo en donde el brillo de la señal de vídeo del fichero de flujo secundario es mayor que el valor umbral.

40 10. Un programa que, cuando se ejecuta en un ordenador o se carga en un ordenador, hace que el ordenador realice las etapas del método de reproducción según la reivindicación 9.

50 11. Un medio de memorización de programas en donde se registra el programa según la reivindicación 10.

FIG. 1

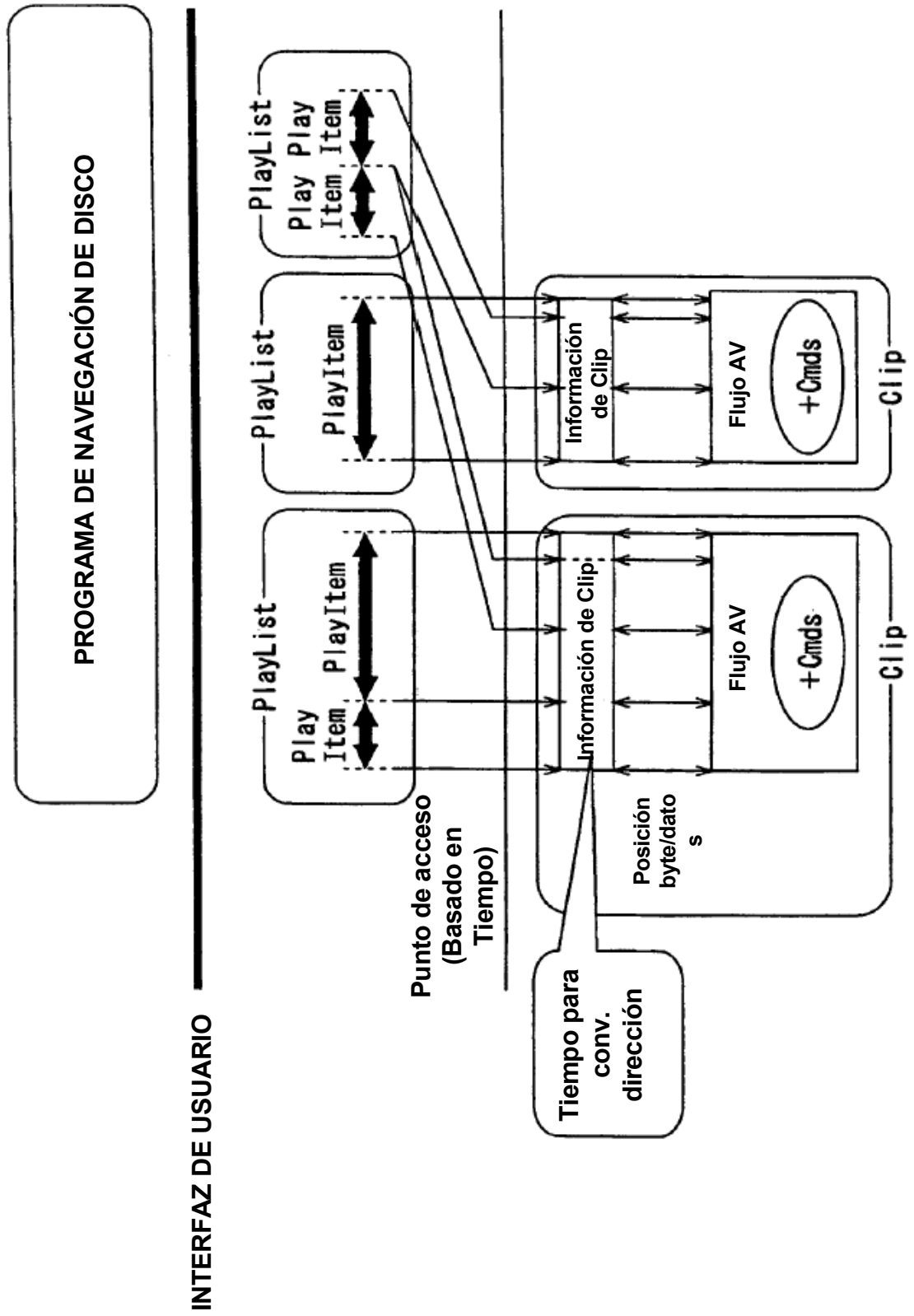


FIG. 2

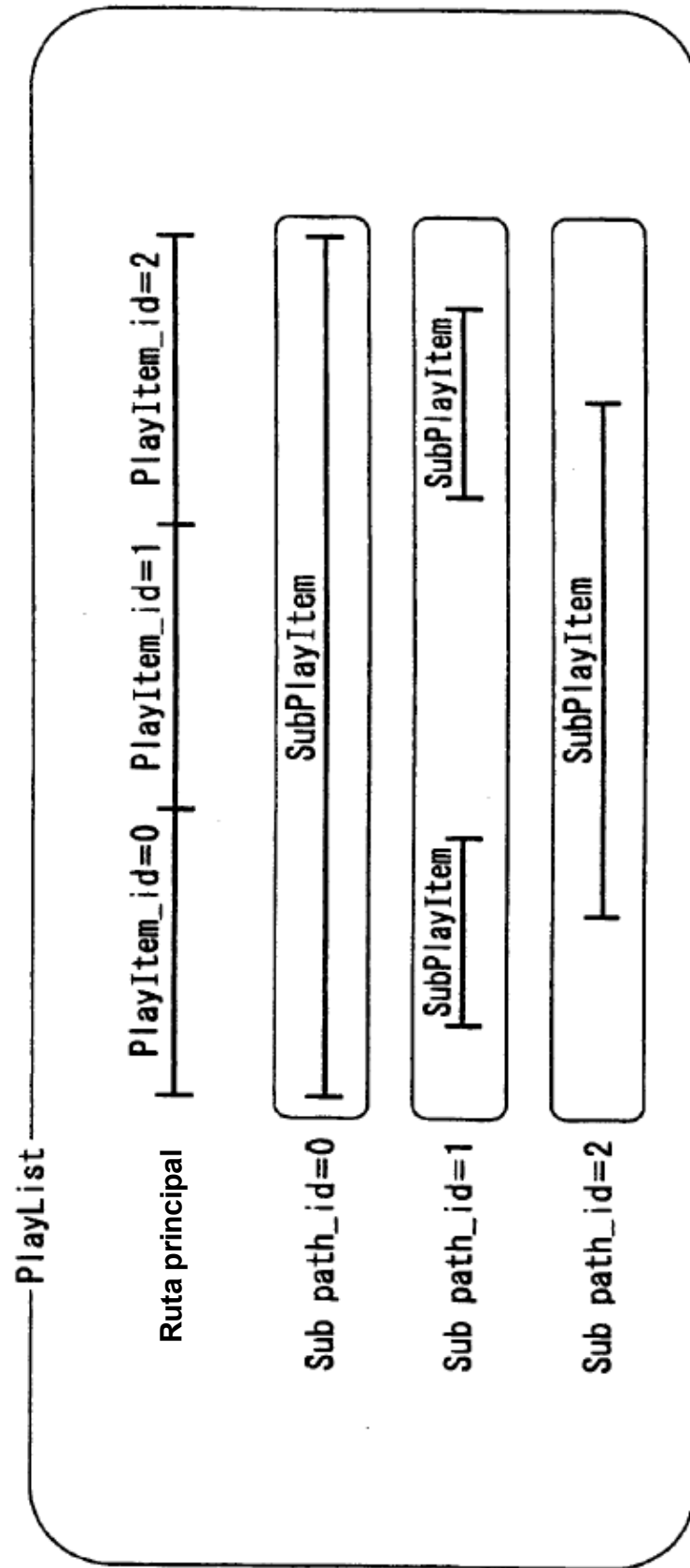


FIG. 3

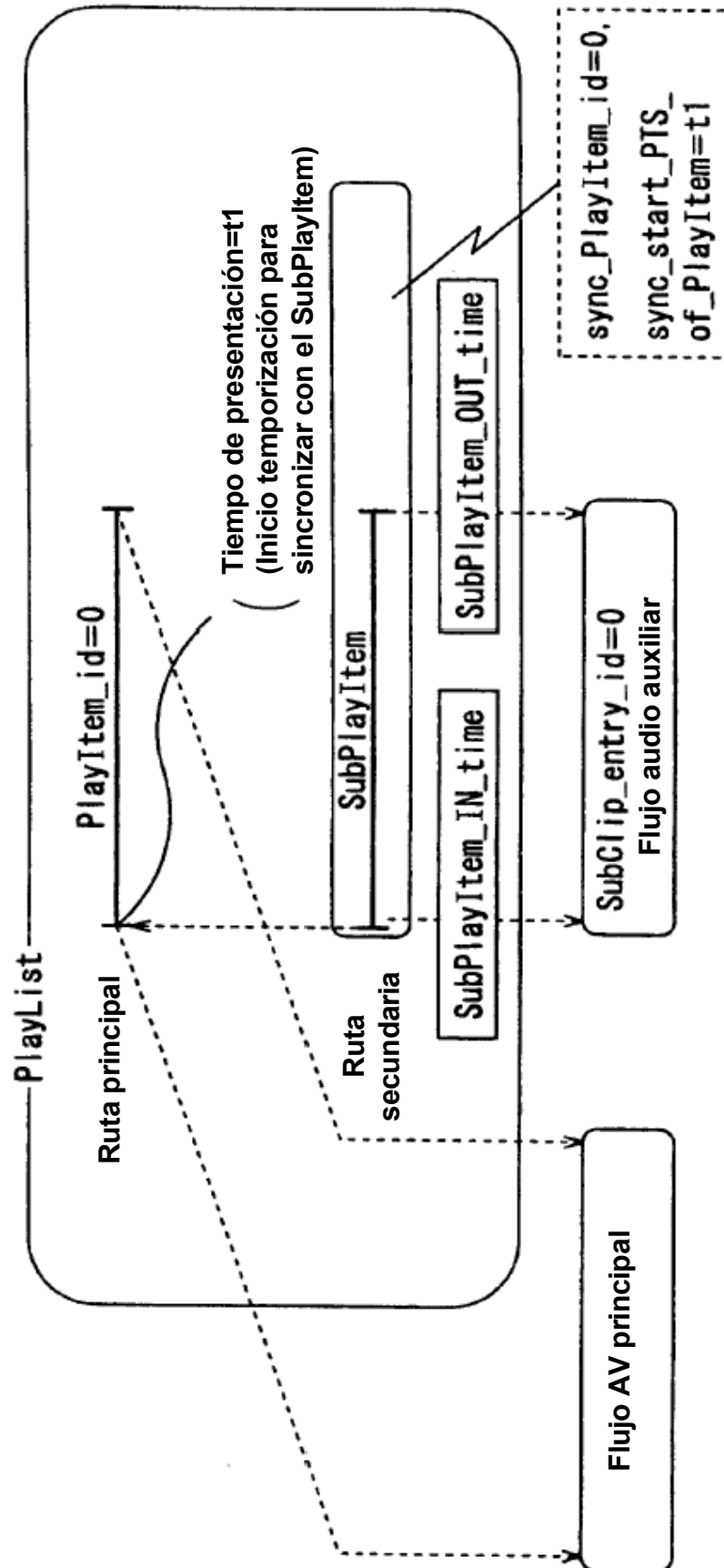


FIG. 4

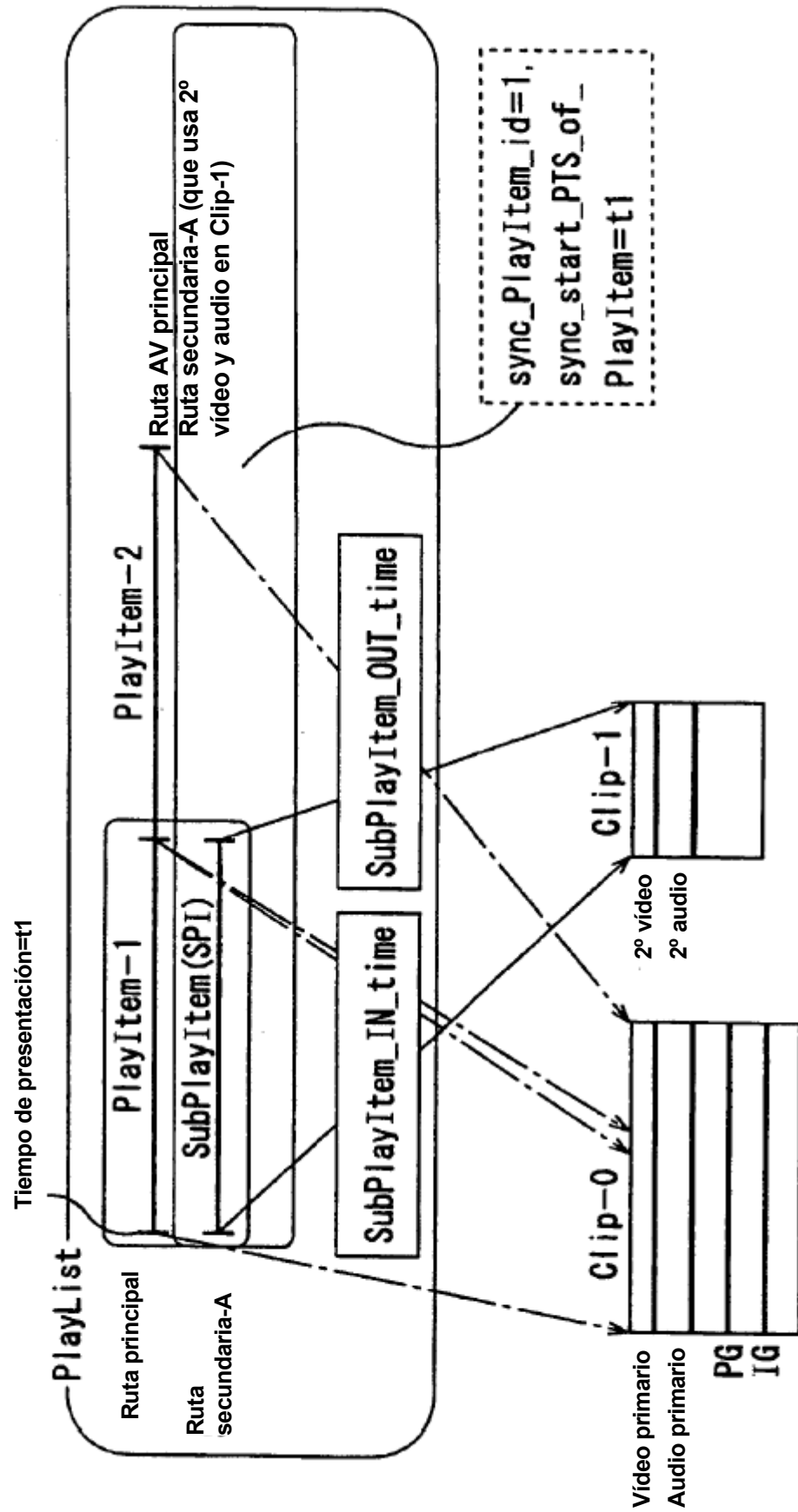


FIG. 5

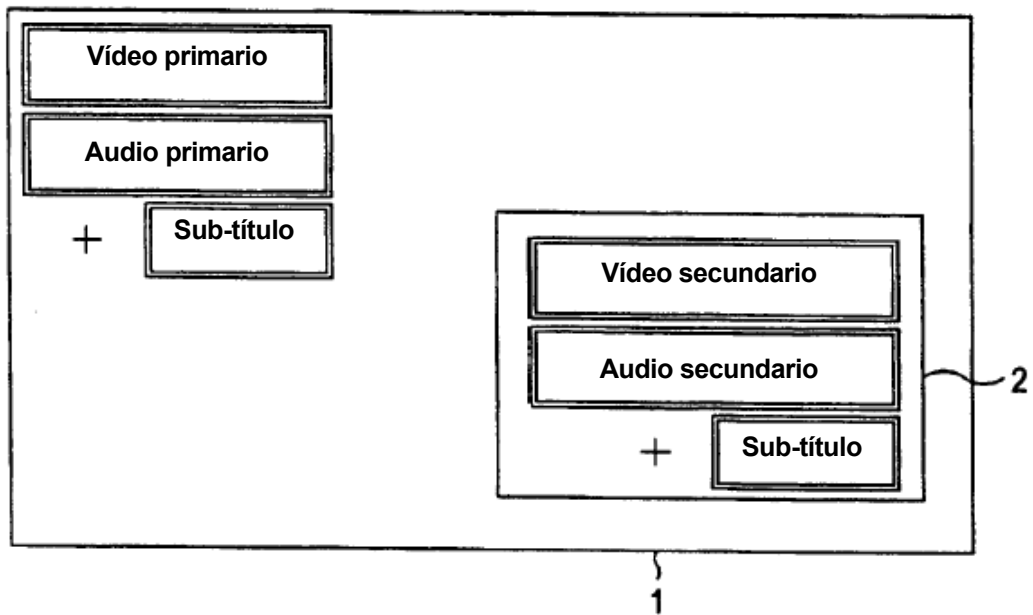


FIG. 6

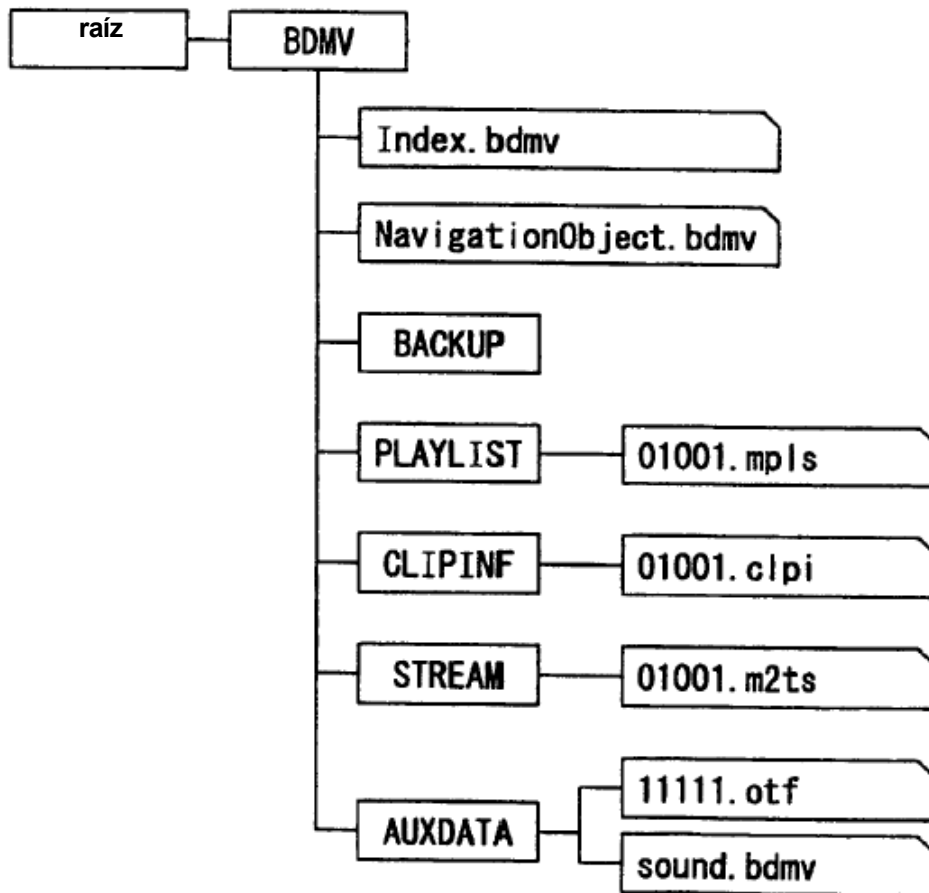


FIG. 7

Sintaxis	Nº de bits	Nemónico
xxxxx.mpls {		
type_indicator	8*4	bslbf
version_number	8*4	bslbf
PlayList_start_address	32	uimsbf
PlayListMark_start_address	32	uimsbf
PlayListExtensionData_start_address	32	uimsbf
reserved_for_future_use	160	bslbf
AppInfoPlayList()		
for (i=0; i<N1; i++) {		
padding_word	16	bslbf
}		
PlayList()		
for (i=0; i<N2; i++) {		
padding_word	16	bslbf
}		
PlayListMark()		
for (i=0; i<N3; i++) {		
padding_word	16	bslbf
}		
PlayListExtensionData()		
for (i=0; i<N4; i++) {		
padding_word	16	bslbf
}		
}		

FIG. 8

Sintaxis	Nº de bits	Nemónico
PlayListExtensionData() {		
Longitud	32	uimsbf
if (longitud != 0) {		
data_block_start_address	32	uimsbf
reserved_for_word_align	24	
number_of_PL_ext_data_entries	8	uimsbf
for (i=0; i<number_of_PL_ext_data_entries; i++) {		
ID1	16	uimsbf
ID2	16	uimsbf
PL_ext_data_start_address	32	uimsbf
PL_ext_data_length	32	uimsbf
}		
for (i=0; i<L1; i++) {		
padding_word	16	bslbf
padding_word	16	bslbf
}		
data_block()	32+	
	8*(length- data_block_start	
}		
}		

FIG. 9

Sintaxis	Nº de bits	Nemónico
<code>pip_metadata() {</code>		
Longitud	32	uimbsf
number_of_pip_metadata_entries		
for (i=0; i<number_of_pip_metadata_entries; i++) {		
pip_metadata_type	8	bslbf
reserved_for_future_use	8	bslbf
ref_to_PlayItem/SubPath_id	16	uimbsf
pip_metadata_time_stamp	32	uimbsf
ref_to_secondary_video_stream_id	8	uimbsf
reserved_for_future_use	8	bslbf
pip_composition_metadata() {		
pip_horizontal_position	16	
pip_vertical_position	16	
pip_scale	8	
reserved_for_future_use	8	bslbf
}		
}		
}		

FIG. 10

Valor	Significado	Nota
0x00	Reservado	
0x01	PIP metadata síncrono	● Vídeo secundario referenciado por ref_to_secondary_video_stream_id se presenta en el momento indicado por pip_metadata_time_stamp con PlayItem referenciado por ref_to_PlayItem_id utilizando información en pip_composition_metadata().
0x02	PIP metadata asíncrono	● Vídeo secundario referenciado por ref_to_secondary_video_stream_id se presenta en el momento indicado por pip_metadata_time_stamp con SubPlayItem referenciado por ref_to_SubPlayItem_id utilizando información en pip_composition_metadata().
Otros	Reservado	

FIG. 11

Sintaxis	Nº de bits	Nemónico
pip_metadata() {		
Longitud	32	uimsbf
number_of_pip_entries		
for (i=0; i<number_of_pip_entries; i++) {		
Synchronous_PIP_metadata_flag		
ref_to_PlayItem/SubPath_id	16	uimsbf
ref_to_secondary_video_stream_id	8	uimsbf
reserved_for_future_use	8	bslbf
number_of_pip_metadata_entries		
for (j=0; j<number_of_pip_metadata_entries; j++) {		
pip_metadata_type	8	bslbf
reserved_for_future_use	8	bslbf
pip_metadata_time_stamp	32	uimsbf
pip_composition_metadata() {		
pip_horizontal_position	16	
pip_vertical_position	16	
pip_scale	8	
reserved_for_future_use	8	bslbf
}		
}		
}		
}		

FIG. 12

Sintaxis	Nº de bits	Nemónico
<code>pip_metadata() {</code>		
Longitud	32	uimsbf
number_of_metadata_block_entries	16	uimsbf
for (k=0; k<number_of_metadata_block_entries; k++) {		
metadata_block_header[k] {		
ref_to_PlayItem_id[k]	16	uimsbf
ref_to_secondary_video_stream_id[k]	8	uimsbf
reserved_for_future_use	8	bslbf
pip_timeline_type[k]	4	uimsbf
flags {		
is_luma_key	1	bslbf
Reservado	1	bslbf
}		
reserved_for_word_align	10	bslbf
if (is_luma_key==1) {		
lower_limit_luma_key	8	uimsbf
upper_limit_luma_key	8	uimsbf
} else {		
reserved_for_future_use	16	bslbf
}		
reserved_for_future_use	16	
metadata_block_data_start_address[k]	32	uimsbf
}		
}		
for (i=0; i<X; i++) {		
padding_word		
}		
for (k=0; k<number_of_metadata_block_entries; k++) {		
metadata_block_data[k] {		
number_of_pip_metadata_entries[k]	16	bslbf
for (i=0; i<number_of_pip_metadata_entries; i++) {		
pip_metadata_time_stamp[i]	32	uimsbf
pip_composition_metadata {		
pip_horizontal_position[i]	12	uimsbf
pip_vertical_position[i]	12	uimsbf
pip_scale[i]	4	uimsbf
reserved_for_future_use	4	bslbf
}		
}		
}		
}		
for (i=0; i<Y[k]; i++) {		
padding_word	16	bslbf
}		
}		

FIG. 13

pip_scale	Significado
0	Reservado
1	Sin puesta en escala (x1)
2	A mitad de escala (x1/2)
3	A cuarta parte de escala (x1/4)
4	Escala 1.5 (x1.5)
5	Escala a pantalla completa
Otros	Reservado

FIG. 14

pip_timeline_type[k]	Significado
0	Reservado
1	Tipo sincronizado de ruta de presentación Picture-in-Picture y pip_metadata_time_stamp _entries se refiere a la cronología de PlayItem referida por la ref_to_PlayItem_id[k]. La ruta secundaria indicada por ref_to_secondary_video_stream_id[k] tendrá el tipo SubPath_type establecido a 5 o 7.
2	Tipo asíncrono de ruta de presentación Picture-in-Picture y pip_metadata_time_stamp _entries se refiere a una cronología de ruta secundaria. La ruta secundaria indicada por ref_to_secondary_video_stream_id[k] tendrá el tipo SubPath_type establecido a 6.
3	Tipo asíncrono de ruta de presentación Picture-in-Picture y pip_metadata_time_stamp _entries se refiere a la cronología de PlayItem referida por la ref_to_PlayItem_id[k]. La ruta secundaria indicada por ref_to_secondary_video_stream_id[k] tendrá el tipo SubPath_type establecido a 6.
Otros	Reservado

FIG. 15

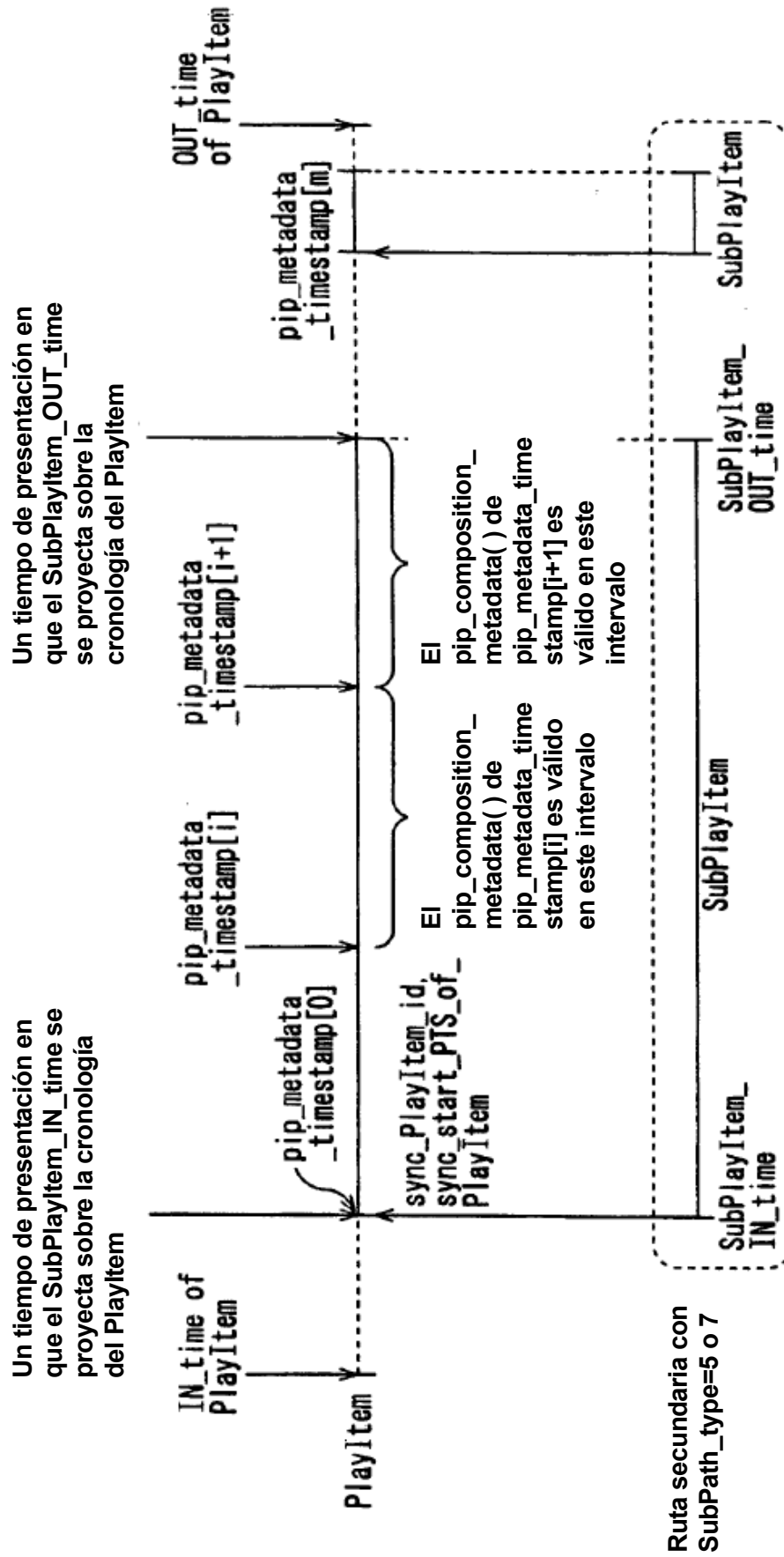


FIG. 16

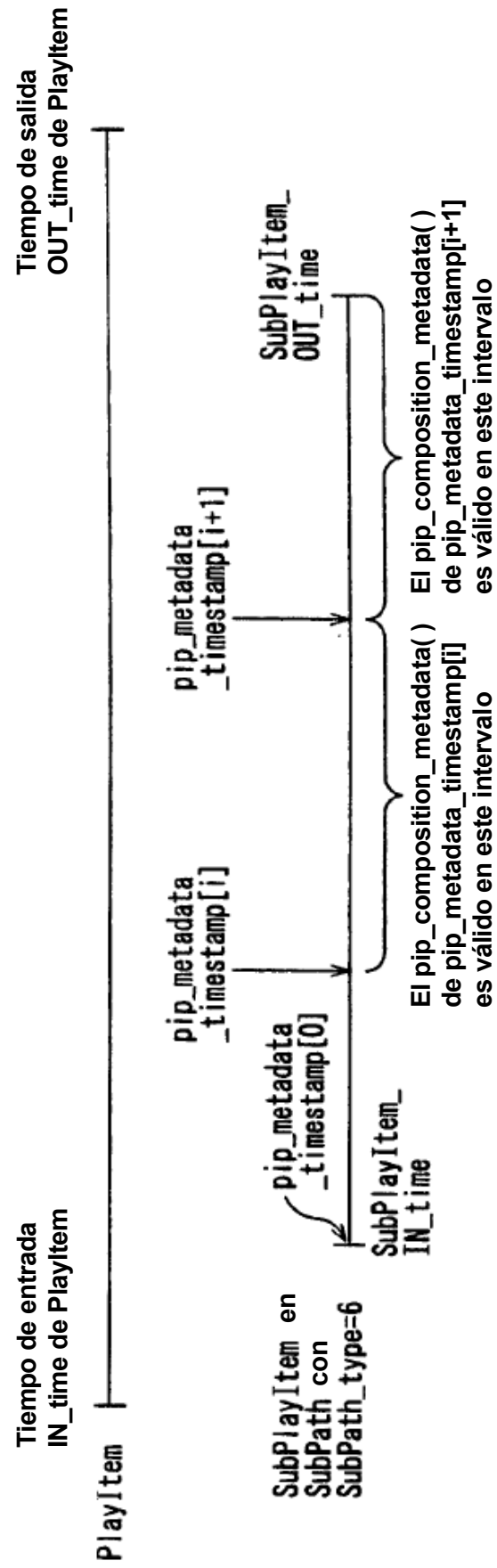


FIG. 17

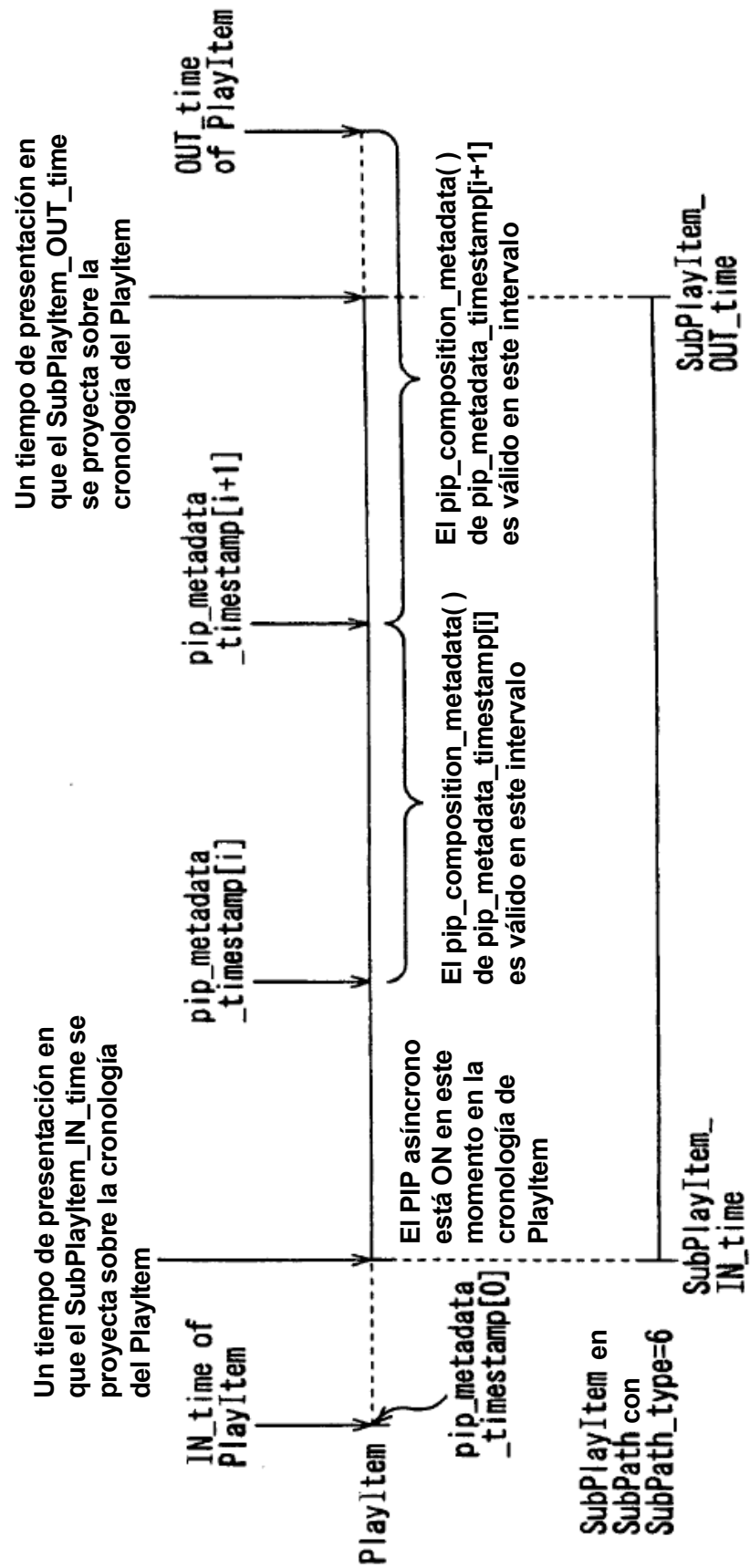


FIG. 18

PlayList-Sintaxis

Sintaxis	Nº de bits	Nemónico
PlayList() {		
Longitud	32	uimsbf
reserved_for_future_use	16	bslbf
number_of_PlayItems	16	uimsbf
number_of_SubPaths	16	uimsbf
for (PlayItem_id=0;		
PlayItem_id<number_of_PlayItems;		
PlayItem_id++) {		
PlayItem()		
}		
for (SubPath_id= 0;		
SubPath_id<number_of_SubPaths;		
SubPath_id++) {		
SubPath()		
}		
}		

FIG. 19

SubPath-Sintaxis

Sintaxis	Nº de bits	Nemónico
SubPath() {		
Longitud	32	uimsbf
reserved_for_future_use	8	bslbf
SubPath_type	8	uimsbf
reserved_for_future_use	15	uimsbf
is_repeat_SubPath	1	bslbf
reserved_for_future_use	8	bslbf
number_of_SubPlayItems	8	uimsbf
for (i=0; i < number_of_SubPlayItems; i++) {		
SubPlayItem(i)		
}		
}		

FIG. 20**SubPath_type**

SubPath_type	Significado
0	Reservado
1	Reservado
2	Out-of-mux y ruta de presentación de audio primario de la diapositiva explorable (La ruta de presentación de audio que utiliza la ruta secundaria SubPath no está sincronizada con la ruta principal que utiliza PlayItems en la PlayList).
3	Out-of-mux y el menú de gráficos interactivos (El menú de gráficos interactivos que utiliza la ruta secundaria SubPath no está sincronizado con la ruta principal que utiliza PlayItems en la PlayList).
4	Out-of-mux y ruta de presentación de subtítulos de texto (La ruta de presentación de subtítulos de texto que utiliza la ruta secundaria SubPath está sincronizada con la ruta principal que utiliza PlayItems en la PlayList).
5	Out-of-mux y tipo sincronizado de AV de una o más rutas de flujos elementales (Ruta de audio primario/PG/IG/audio secundario). Out-of-mux y tipo sincronizado de AV de ruta de presentación Picture-in-Picture que contiene una o más rutas de flujos elementales. (Los flujos elementales utilizados por la ruta se multiplexan en un Clip separado del otro Clip usado por PlayItem. La ruta que utiliza la SubPath está sincronizada con la ruta principal que utiliza PlayItems en la PlayList).
6	Out-of-mux y tipo no sincronizado de AV de ruta de presentación Picture-in-Picture que contiene una o más rutas de flujos elementales. (Los flujos elementales utilizados por la ruta se multiplexan en un Clip separado del otro Clip usado por PlayItem. La ruta que utiliza la SubPath no está sincronizada con la ruta principal que utiliza PlayItems en la PlayList).
7	El tipo In-mux y el tipo sincronizado AV de ruta de presentación Picture-in-Picture que contiene una o más rutas de flujos elementales. (Los flujos elementales utilizados por la ruta se multiplexan en el mismo Clip usado por PlayItem. La ruta que utiliza la SubPath está sincronizada con la ruta principal que utiliza PlayItems en la PlayList).
8-255	Reservado

FIG. 21

SubPlayItem(i)-Sintaxis

Sintaxis	Nº de bits	Nemónico
SubPlayItem(i) {		
Longitud	16	uimsbf
Clip_Information_file_name[0] //subclip_entry_id=0	8*5	bslbf
Clip_codec_identifier[0]	8*4	bslbf
reserved_for_future_use	31	bslbf
is_multi_Clip_entries	1	bslbf
ref_to_STC_id[0]	8	uimsbf
SubPlayItem_IN_time	32	uimsbf
SubPlayItem_OUT_time	32	uimsbf
sync_PlayItem_id	16	uimsbf
sync_start_PTS_of_PlayItem	32	uimsbf
if(is_multi_Clip_entries==1b) {		
reserved_for_future_use	8	bslbf
num_of_Clip_entries	8	uimsbf
for(subclip_entry_id=1://Note:Entries after subclip_entry_id=0		
subclip_entry_id<num_of_Clip_entries:subclip_entry_id ++){		
Clip_Information_file_name[subclip_entry_id]	8*5	bslbf
Clip_codec_identifier[subclip_entry_id]	8*4	bslbf
ref_to_STC_id[subclip_entry_id]	8	uimsbf
reserved_for_future_use	8	bslbf
}		
}		
}		

FIG. 22

PlayItem-Sintaxis

Sintaxis	Nº de bits	Nemónico
PlayItem() {		
Longitud	16	uimsbf
Clip_Information_file_name[0]	8*5	bslbf
Clip_codec_identifier[0]	8*4	bslbf
reserved_for_future_use	11	bslbf
is_multi_angle	1	bslbf
connection_condition	4	uimsbf
ref_to_STC_id[0]	8	uimsbf
IN_time	32	uimsbf
OUT_time	32	uimsbf
UO_mask_table()		
PlayItem_random_access_mode	8	uimsbf
still_mode	8	uimsbf
if(still_mode==0x1) {		
still_time	16	uimsbf
} else {		
Reservado	16	bslbf
}		
if(is_multi_angle==1b) {		
number_of_angles	8	uimsbf
reserved_for_future_use	7	bslbf
is_seamless_angle_change	1	uimsbf
for(angle_id = 1; //Note: angles after angle_id=1		
angle_id<number_of_angles; angle_id++) {		
Clip_Information_file_name[angle_id]	8*5	bslbf
Clip_codec_identifier[angle_id]	8*4	bslbf
ref_to_STC_id[angle_id]	8	uimsbf
reserved_for_future_use	16	bslbf
}		
}		
STN_table()		
}		

FIG. 23

STN_table()

Sintaxis	Nº de bits	Nemónico
STN_table() {		
Longitud	16	uimsbf
reserved_for_future_use	16	bslbf
number_of_video_stream_entries	8	uimsbf
number_of_audio_stream_entries	8	uimsbf
number_of_audio_stream2_entries	8	uimsbf
number_of_PG_textST_stream_entries	8	uimsbf
number_of_IG_stream_entries	8	uimsbf
reserved_for_future_use	64	bslbf
for (video_stream_id=0;		
video_stream_id < number_of_video_stream_entries;		
video_stream_id++) {		
stream_entry()		
stream_attribute()		
}		
for (audio_stream_id=0;		
audio_stream_id < number_of_audio_stream_entries;		
audio_stream_id++) {		
stream_entry()		
stream_attributes()		
}		
for (audio_stream_id2=0;		
audio_stream_id2 < number_of_audio_stream2_entries;		
audio_stream_id2++) {		
number_of_combinations		
for (i=0; i < number_of_combinations; i++) {		
audio_stream_id		
}		
stream_entry()		
stream_attributes()		
}		
for (PG_textST_stream_id=0;		
PG_textST_stream_id < number_of_PG_textST_stream_entries;		
PG_textST_stream_id++) {		
stream_entry()		
stream_attribute()		
}		
for (IG_stream_id=0;		
IG_stream_id < number_of_IG_stream_entries;		
IG_stream_id++) {		
stream_entry()		
stream_attribute()		
}		
}		

FIG. 24

Sintaxis	Nº de bits	Nemónico
stream_entry(id) {		
Longitud	8	uimsbf
Tipo	8	uimsbf
if (type==1) {		
ref_to_stream_PID_of_mainClip	16	uimsbf
reserved_for_future_use	48	Bslbf
}else if (type==2) {		
ref_to_SubPath_id	8	Uimsbf
ref_to_subClip_entry_id	8	Uimsbf
ref_to_stream_PID_of_subClip	16	
reserved_for_future_use	32	bslbf
}		
}		

FIG. 25

stream_attribute()

Sintaxis	Nº de bits	Nemónico
stream_attribute() {		
Longitud	8	uimsbf
stream_coding_type	8	bslbf
if (stream_coding_type==0x02) {		
video_format	4	bslbf
frame_rate	4	bslbf
aspect_ratio	4	bslbf
reserved_for_future_use	4	bslbf
} else if (stream_coding_type==0x80 stream_coding_type==0x81 stream_coding_type==0x82) {		
audio_presentation_type	4	bslbf
sampling_frequency	4	bslbf
audio_language_code	8*2	bslbf
reserved_for_future_use	8	bslbf
} else if (stream_coding_type==0x90) { // Flujo gráficos presentación		
PG_language_code	8*2	bslbf
} else if (stream_coding_type==0x91) { // Flujo gráficos interactivos		
IG_language_code	8*2	bslbf
} else if (stream_coding_type==0x92) { // Flujo subtítulos de texto		
character_code	8	bslbf
textST_language_code	8*2	bslbf
}		
}		

FIG. 26**stream_coding_type**

stream_coding_type	Significado
0x02	Flujo vídeo MPEG-2
0x80	Audio HDMV LPCM
0x81	Audio Dolby AC-3
0x82	Audio dts
0x90	Flujo gráficos presentación
0x91	Flujo gráficos interactivos
0x92	Flujo subtítulos de texto
Otros valores	Reservado

FIG. 27**video_format**

video_format	Significado	Norma de vídeo
0	Reservado	
1	480i	ITU-R BT. 601-4
2	576i	ITU-R BT. 601-4
3	480p	SMPTE 293M
4	1080i	SMPTE 274M
5	720p	SMPTE 296M
6	1080p	SMPTE 274M
7 - 14	Reservado	

FIG. 28**frame_rate**

frame_rate	Significado [Hz]
0	Reservado
1	24 000/1001 (23.976...)
2	24
3	25
4	30 000/1001 (29.97...)
5	Reservado
6	50
7	60 000/1001 (59.94...)
8-15	Reservado

FIG. 29**aspect_ratio**

aspect_ratio	Significado
0	Reservado
1	Reservado
2	Ratio de aspecto de pantalla 4:3
3	Ratio de aspecto de pantalla 16:9
4-15	Reservado

FIG. 30**audio_presentation_type**

audio_presentation_type	Significado
0	Reservado
1	Canal mono único
2	Canal mono dual
3	Estéreo (2 canales)
4	Reservado
5	Reservado
6	Multi-canal
7-15	Reservado

FIG. 31**sampling_frequency**

sampling_frequency	Significado
0	Reservado
1	48 kHz
2	Reservado
3	Reservado
4	96 kHz
5 -15	Reservado

FIG. 32

Código de carácter

Valor de código de carácter	Conjunto de caracteres	Sistema de codificación caracteres
0x00	Reservado	
0x01	Unicode V1.1 (ISO 10646-1)	UTF8
0x02	Unicode V1.1 (ISO 10646-1)	UTF16 big endian
0x03	Shift JIS (Japonés)	
0x04	KSC 5601-1987 incluyendo KSC 5653 para carácter romano (Coreano)	
0x05	GB18030-2000 (Chino)	
0x06	GB2312 (Chino)	
0x07	BIG5 (Chino)	
Otros	Reservado	

FIG. 33

STN_table()

Sintaxis	Nº de bits	Nemónico
STN_table() {		
Longitud	16	uimsbf
reserved_for_future_use	16	bslbf
number_of_video_stream_entries	8	uimsbf
number_of_audio_stream_entries	8	uimsbf
number_of_audio_stream2_entries	8	uimsbf
number_of_PG_textST_stream_entries	8	uimsbf
number_of_IG_stream_entries	8	uimsbf
number_of_video_stream2_entries	8	uimsbf
reserved_for_future_use	48	bslbf
for (video_stream_id=0;		
video_stream_id < number_of_video_stream_entries;		
video_stream_id++) {		
stream_entry()		
stream_attribute()		
}		
for (audio_stream_id=0;		
audio_stream_id < number_of_audio_stream_entries;		
audio_stream_id++) {		
stream_entry()		
stream_attributes()		
}		
for (audio_stream_id2=0;		
audio_stream_id2 < number_of_audio_stream2_entries;		
audio_stream_id2++) {		
number_of_combinations		
for (i=0; i < number_of_combinations; i++) {		
audio_stream_id		
}		
stream_entry()		
stream_attributes()		

FIG. 34

Sintaxis	Nº de bits	Nemónico
}		
for (PG_textST_stream_id=0:		
PG_textST_stream_id < number_of_PG_textST_stream_entries;		
PG_txtST_stream_id++) {		
stream_entry()		
stream_attribute()		
}		
for (IG_stream_id=0:		
IG_stream_id < number_of_IG_stream_entries;		
IG_stream_id++) {		
stream_entry()		
stream_attribute()		
}		
for (video_stream_id2=0:		
video_stream_id2 < num_of_video_stream2_entries;		
video_stream_id2++) {		
number_of_Audio_combinations_for_video2	8	
for(i=0; i<number_of_Audio_combinations_for_video2; i++){		
audio_stream_id	8	
audio_stream_id2	8	
}		
number_of_Subtitle_combinations_for_video2	8	
for(i=0; i<number_of_Subtitle_combinations_for_video2; i++){		
PG_textST_stream_id	8	
}		
stream_entry()		
stream_attributes()		
}		
)		

FIG. 35

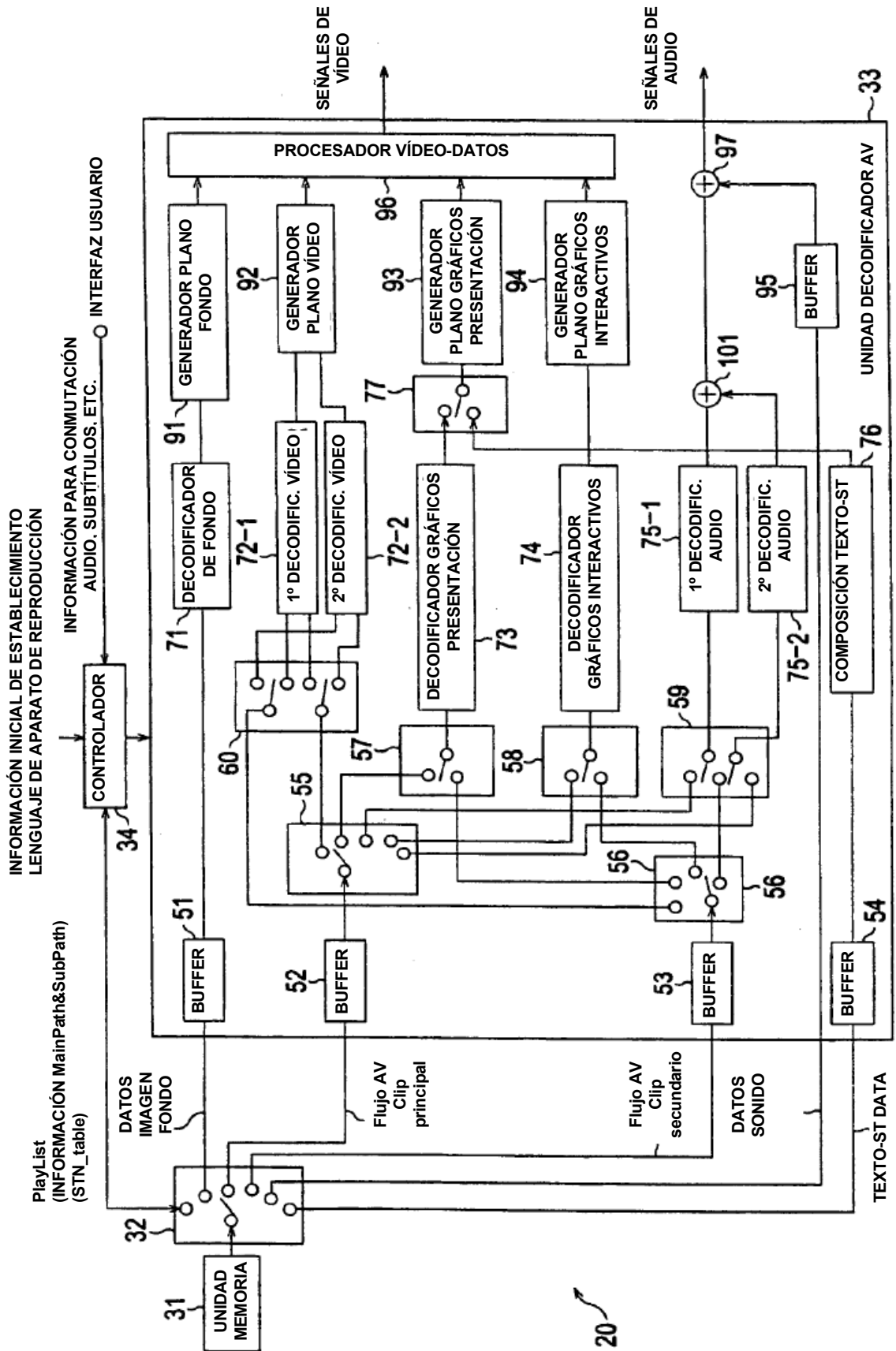


FIG. 36

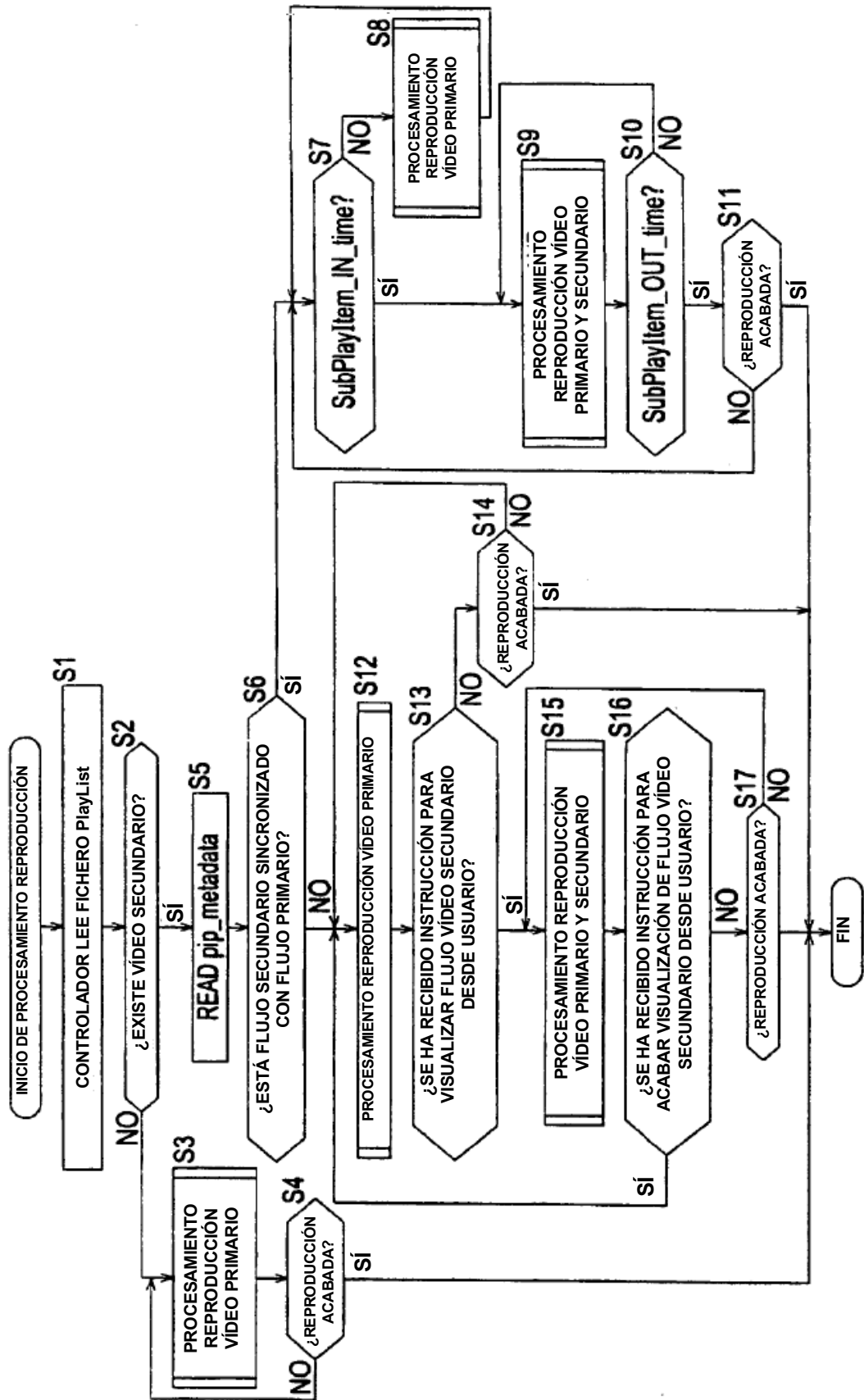


FIG. 37

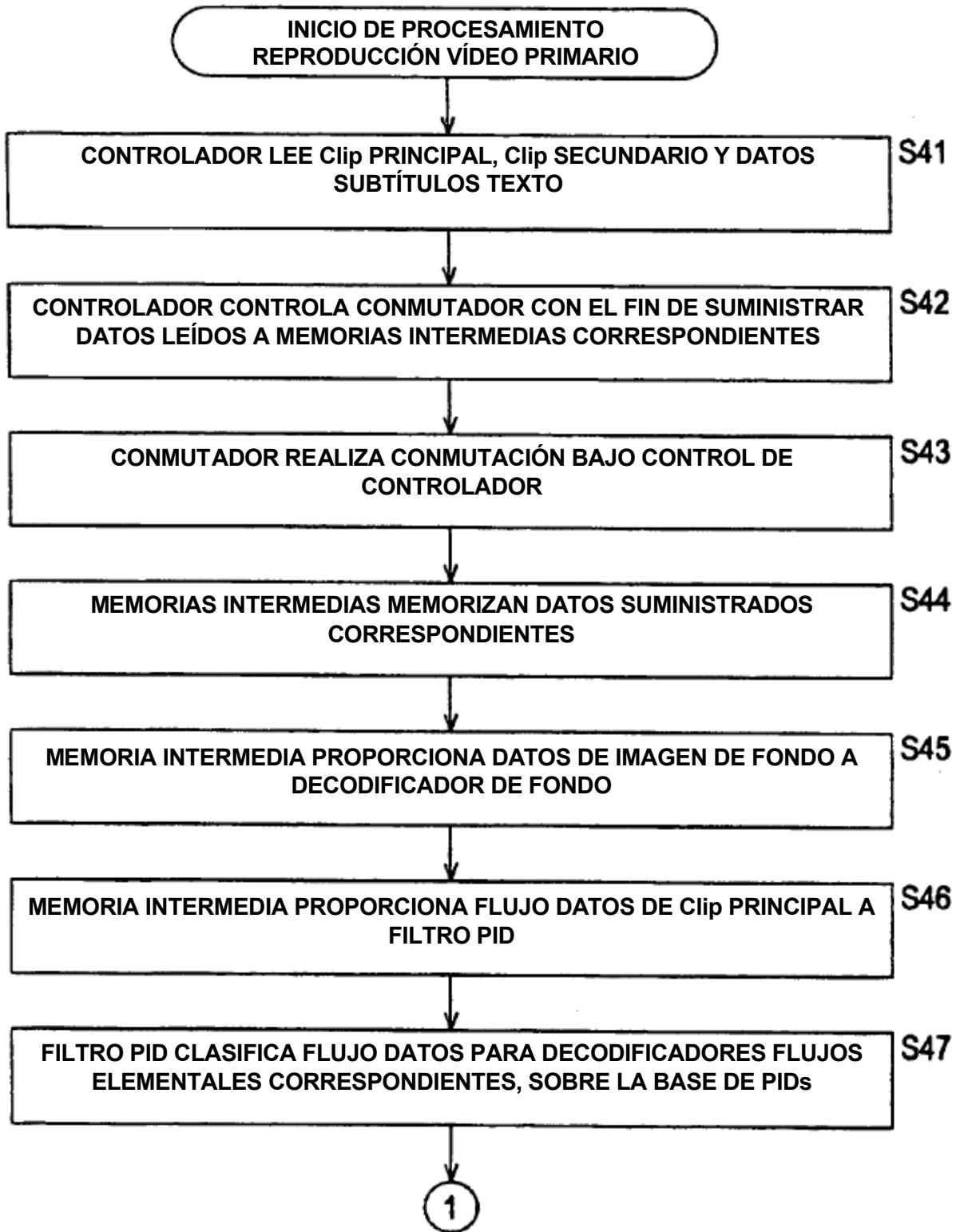


FIG. 38

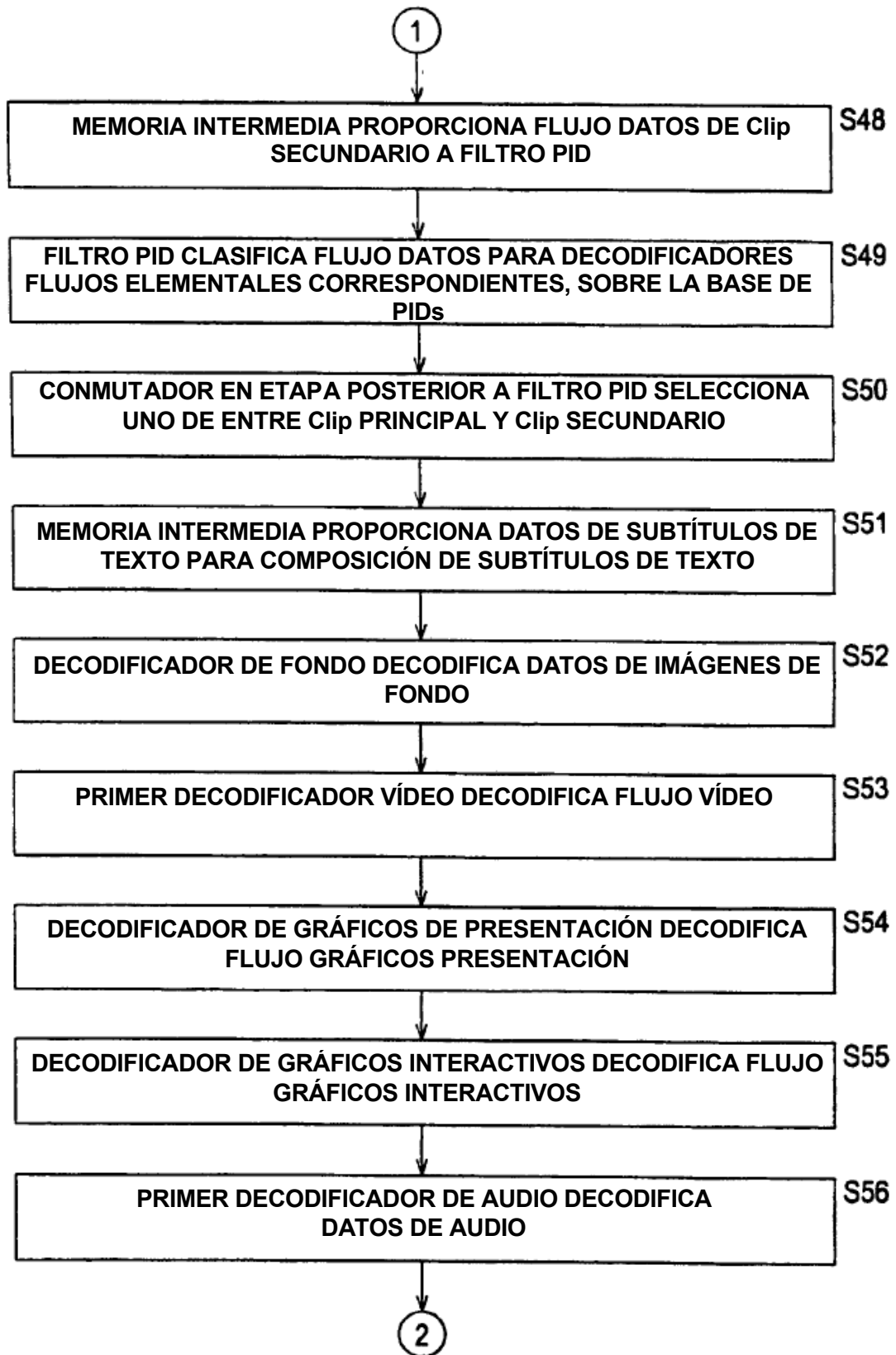


FIG. 39

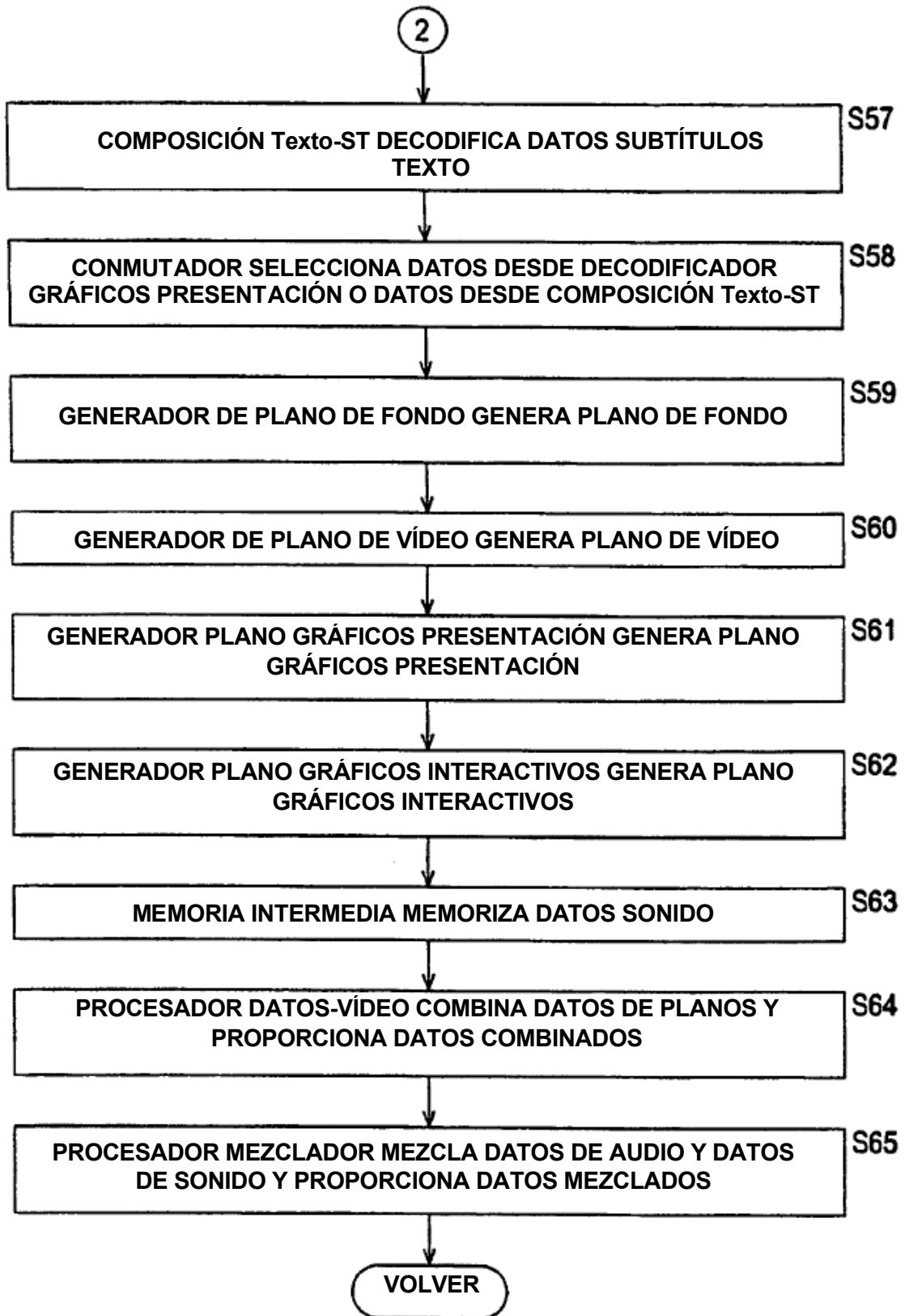


FIG. 40



FIG. 41

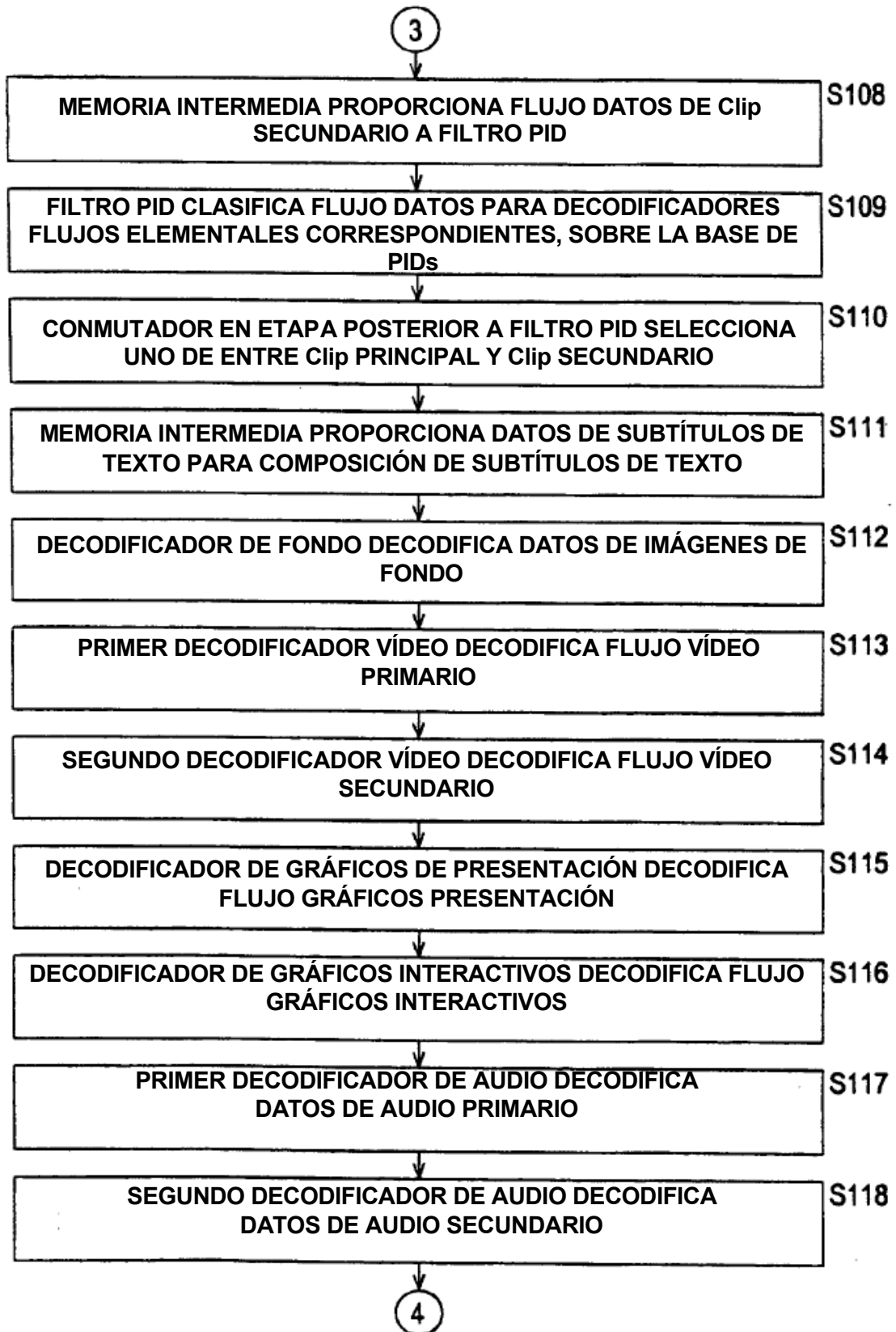


FIG. 42

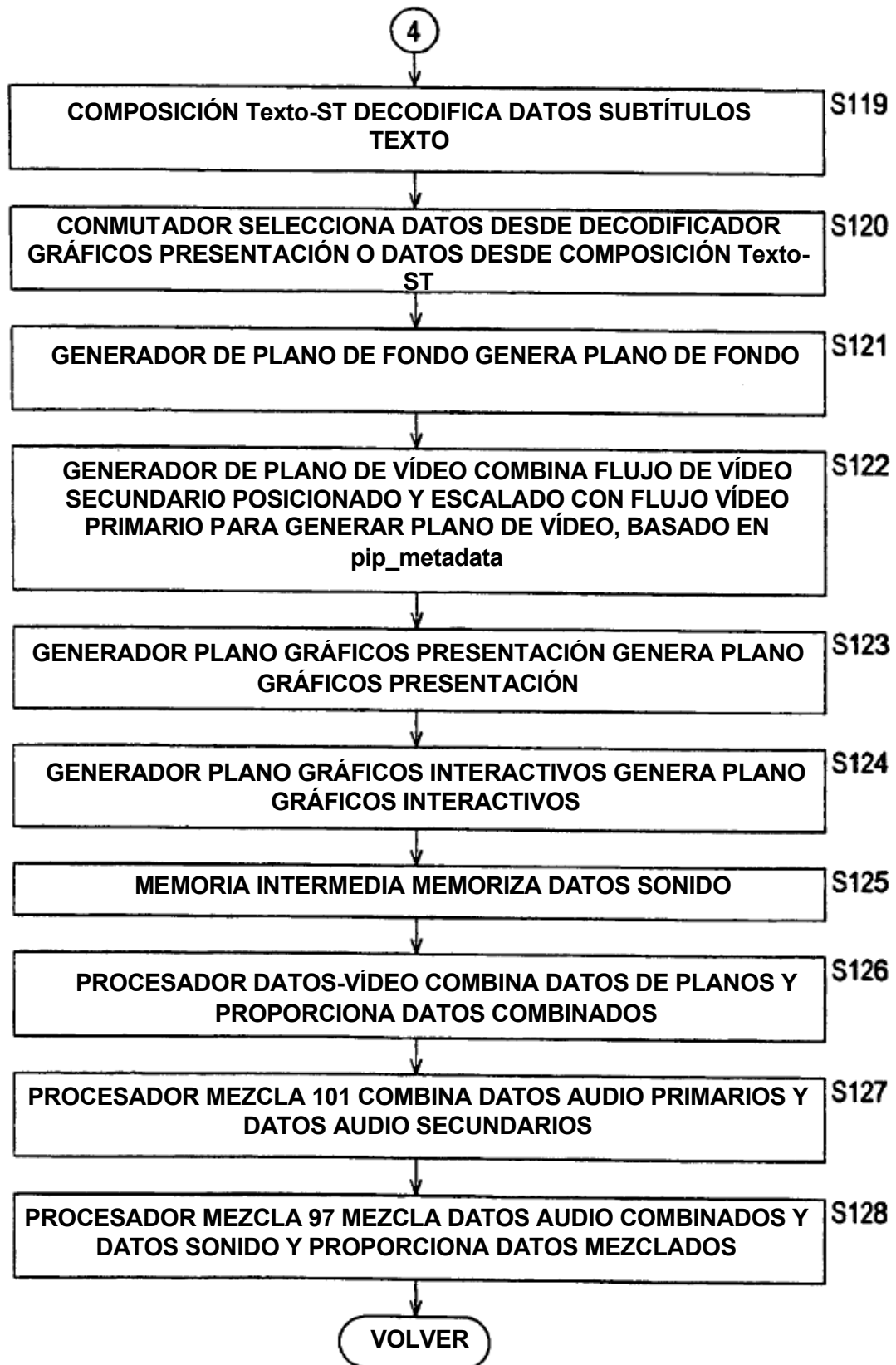


FIG. 43

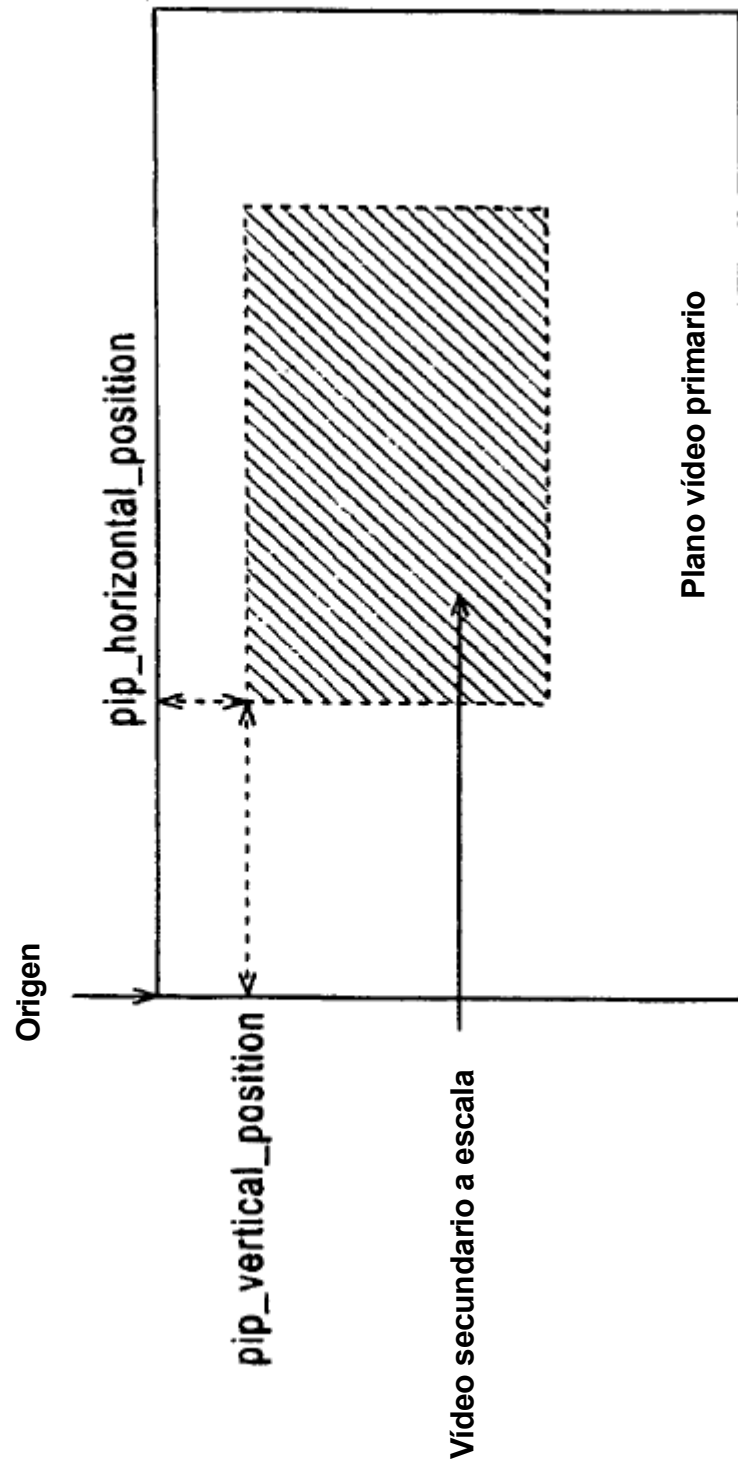
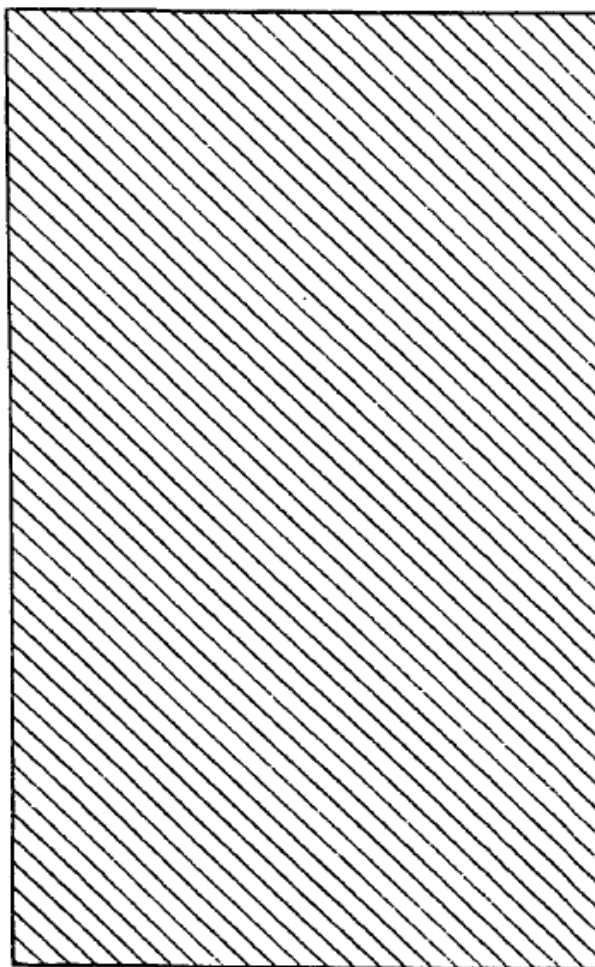


FIG. 44

Vídeo primario



Vídeo secundario

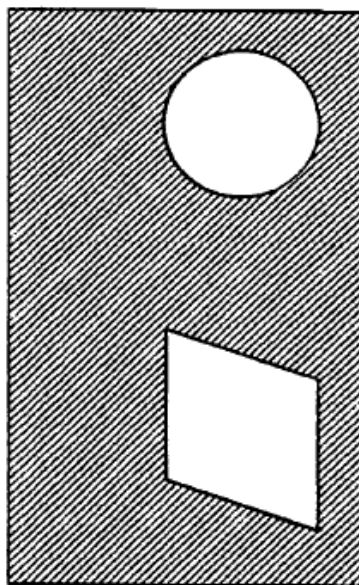


FIG. 45

Vídeo primario + vídeo secundario

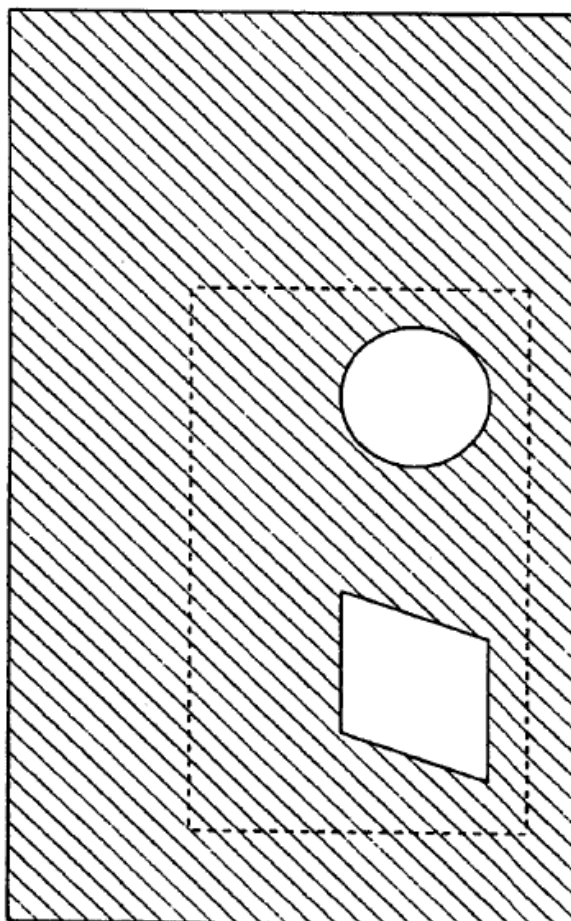


FIG. 46

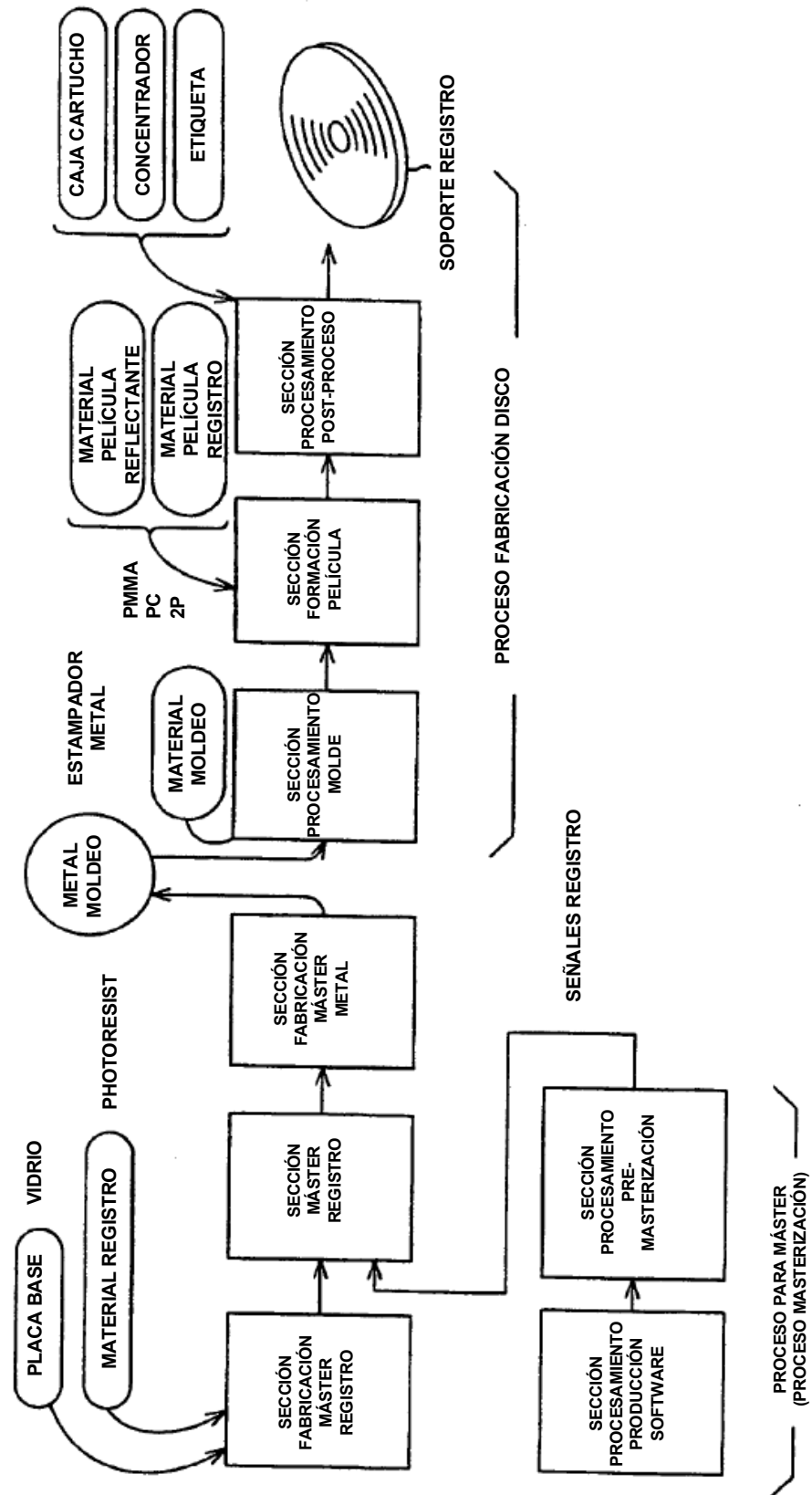


FIG. 47

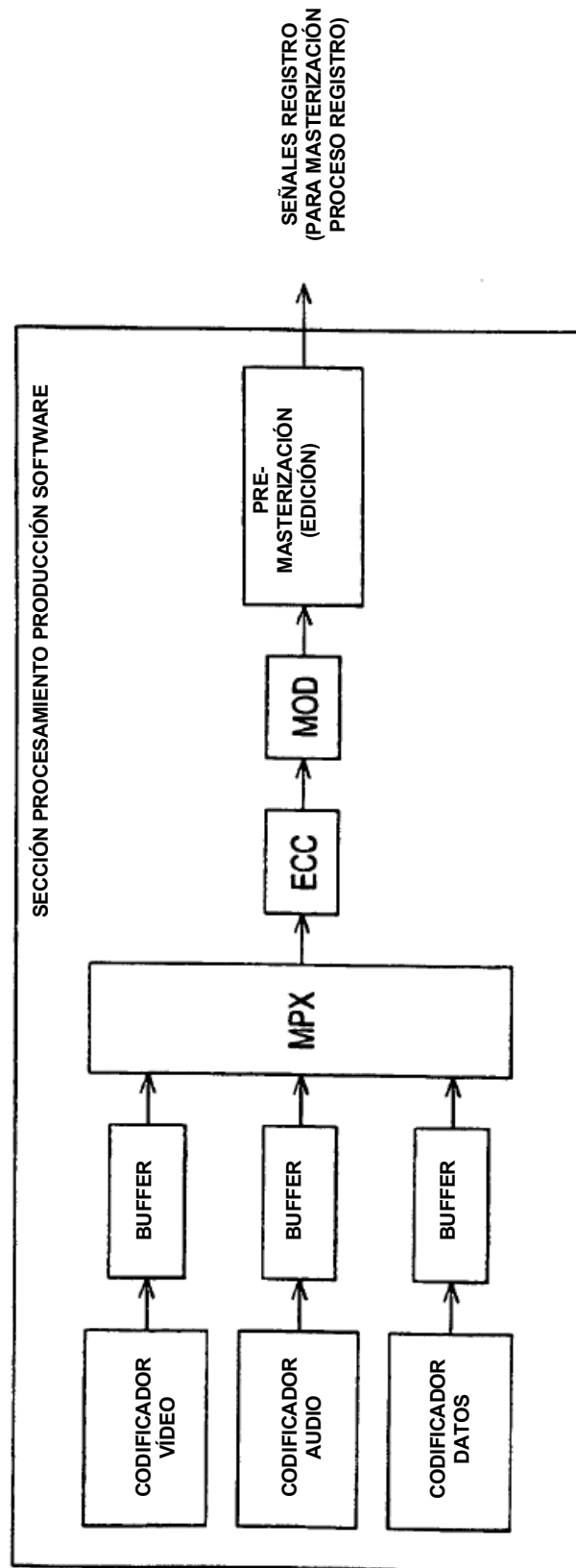


FIG. 48

