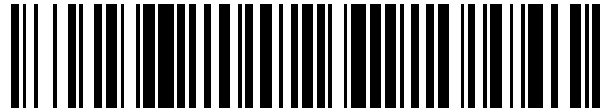


19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 526 578**

51 Int. Cl.:

H04N 19/00 (2014.01)

G11B 20/12 (2006.01)

H04N 5/92 (2006.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

96 Fecha de presentación y número de la solicitud europea: **25.03.2010 E 10761595 (7)**

97 Fecha y número de publicación de la concesión europea: **19.11.2014 EP 2285127**

54 Título: **Dispositivo de registro, método de registro, dispositivo de reproducción, método de reproducción, soporte de registro y programa**

30 Prioridad:

08.04.2009 JP 2009094254

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

13.01.2015

73 Titular/es:

**SONY CORPORATION (100.0%)
1-7-1 Konan Minato-ku
Tokyo 108-0075, JP**

72 Inventor/es:

HATTORI, SHINOBU

74 Agente/Representante:

LEHMANN NOVO, María Isabel

ES 2 526 578 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 registro, método de registro, dispositivo de reproducción, método de reproducción, soporte de registro y programa

5

CAMPO TÉCNICO

La presente invención se refiere a un dispositivo de registro, un método de registro, un dispositivo de reproducción, un método de reproducción, un soporte de registro y un programa y en particular, se refiere a un dispositivo de registro, un método de registro, un dispositivo de reproducción, un método de reproducción, un soporte de registro y un programa que permite a un soporte de registro, tal como un BD, registrar un flujo de imágenes de base y un flujo de imágenes extendidas obtenidas mediante la codificación de datos de vídeo de una pluralidad de puntos de vista utilizando un método de codificación predeterminado a reproducirse en un dispositivo incompatible con la reproducción de datos de vídeo de una pluralidad de puntos de vista.

10

15

ANTECEDENTES DE LA INVENCION

Un contenido de imágenes bidimensional es el flujo de contenidos, tales como imágenes en movimiento, pero recientemente, ha sido objeto de atención un contenido e imágenes estereoscópicas que permite la visión estereoscópica.

20

Un dispositivo dedicado es necesario para visualizar una imagen estereoscópica. Una realización, a modo de ejemplo, de dicho dispositivo para la visión estereoscópica incluye un sistema de imágenes estereoscópicas de IP (Fotografía Integral) desarrollado por NHK (Nippon Hoso Kyokai).

25

Los datos de imágenes de una imagen estereoscópica están constituidos por datos de imagen de una pluralidad de puntos de vista (datos de imágenes de una imagen capturada desde una pluralidad de puntos de vista). A medida que el número de puntos de vista es mayor y a medida que la gama cubierta por los puntos de vista es más amplia, puede verse un sujeto desde más diversas direcciones. Es decir, como una especie de "televisión en la que el sujeto puede examinarse" puede realizarse a este respecto.

30

Entre las imágenes estereoscópicas, una imagen con el más pequeño número de puntos de vista es una imagen estéreo (así denominada imagen en 3D) en donde el número de puntos de vista es dos. Los datos de imágenes de la imagen estéreo están constituidos por datos de una imagen izquierda, que es una imagen observada por un ojo izquierdo y los datos de una imagen derecha, que es una imagen observada por un ojo derecho.

35

Por otro lado, el contenido de una imagen de alta resolución, tal como imágenes en movimiento, tiene una gran cantidad de datos y por ello, se necesita un soporte de registro de gran capacidad para registrar un contenido que tenga dicha gran cantidad de datos.

40

Una realización, a modo de ejemplo, de dicho soporte de otro de gran capacidad incluye un disco Blu-Ray (marca registrada) (en adelante, también referido como BD), tal como un BD (Blu-Ray (marca registrada) -ROM (memoria de solamente lectura).

45

LISTA DE CITAS DE DOCUMENTACIÓN DE PATENTE

PTL 1: Solicitud de patente japonesa sin examinar con número de publicación JP-A-2005-348314

50

SUMARIO DE LA INVENCION**Problema técnico**

En la norma BD, se refiere a la forma de registrar datos de imagen de una imagen estereoscópica que incluye una imagen estéreo en un BD o cómo reproducir los datos de imagen no está definido.

55

Los datos de imagen de una imagen de estéreo están constituidos por dos flujos de datos: un flujo de datos de una imagen izquierda y un flujo de datos de una imagen derecha. Si los dos flujos de datos se registran en un BD tal como están, puede ser imposible reproducir los flujos de datos en un reproductor de BD ya de amplia dispersión.

60

El documento US 2008/0304766 A1 da a conocer un formato de memorización de ficheros de imágenes en 2D que comprende una denominada área Mdat que incluye datos de imagen y datos de voz, y un área Moov correspondiente a una zona de cabeceras. Un formato de memorización de ficheros de imagen en 3D comprende un área Mdat, un área Moov y un área Metadata. El área Mdat incluye una pista de voz, una primera pista de imagen y una segunda pista de imagen. El área Moov y el área Metadata comprenden, respectivamente, información respecto a la primera y la segunda vista de imagen, incluyendo dicha información de pista la información con respecto al número de todas las tramas de cada elemento de datos y las magnitudes de las tramas.

65

El documento EP 1 501 316 A1 se refiere a datos de imagen para un formato tridimensional. Un fichero de información multimedia incluye al menos tres elementos que son los datos de imagen en 3D, información de control de cabecera para reproducir los datos de imagen en 3D y la información de control de presentación visual en 3D para la conversión de una imagen en 3D obtenida en un formato en 3D deseado. La información de control de visualización en 3D incluye el número de puntos de vista y los respectivos flujos correspondientes a las respectivas posiciones de los puntos de vista. Un número de flujos 2 o 3 se asigna, respectivamente, a los datos de imagen para el ojo izquierdo y el ojo derecho y es objeto de escritura en cada cabecera de paquete.

La presente invención se ha realizado considerando dichas circunstancias y permite a un soporte de otro, tal como un BD, memorizar un flujo de imagen base y un flujo de imagen extendida obtenidos codificando datos de vídeo de una pluralidad de puntos de vista utilizando un método de codificación predeterminado a reproducirse en un dispositivo incompatible con la reproducción de datos de vídeo de una pluralidad de puntos de vista.

La invención se define en las reivindicaciones independientes.

Un dispositivo de registro puede incluir, a modo de ejemplo, un medio de codificación para codificar datos de vídeo de una pluralidad de puntos de vista utilizando un método de codificación predeterminado y proporcionando, a la salida, un flujo de imagen base que está constituido por datos sin una cabecera de datos que incluye información de información de un punto de vista y un flujo de imagen extendida que está constituida por datos con la cabecera de datos que incluye la información de identificación que muestra que los datos son datos de un punto de vista expandido.

Los medios de codificación pueden hacer que supriman la cabecera de datos desde el flujo de imagen base que se obtiene codificando datos de vídeo de una pluralidad de puntos de vista utilizando un método de codificación predeterminado y que está constituido por datos con la cabecera de datos y para proporcionar, a la salida, el flujo de imagen base que está constituida por datos sin la cabecera de datos.

Los medios de codificación pueden hacer que se establezca un valor de uno o más para la cabecera de datos, sirviendo dicho valor como la información de identificación que muestra que los datos son datos de un punto de vista expandido y para proporcionar, a la salida, el flujo de imagen extendida.

Un método de registro puede incluir, a modo de ejemplo, la etapa de codificar datos de vídeo de una pluralidad de puntos de vista utilizando un método de codificación predeterminado y proporcionando, a la salida, un flujo de imagen base que está constituido por datos sin una cabecera de datos que incluye información de identificación de un punto de vista y un flujo de imagen extendida que está constituido por datos con la cabecera de datos que incluye la información de identificación que muestra que los datos son datos de un punto de vista expandido.

Un programa puede hacer, a modo de ejemplo, que un ordenador ejecute un proceso que incluye la etapa codificar datos de vídeo de una pluralidad de puntos de vista utilizando un método de codificación predeterminado y proporcionando, a la salida, un flujo de imagen base que está constituido por datos sin una cabecera de datos que incluye información de identificación de un punto de vista y un flujo de imagen extendida que está constituido por datos con la cabecera de datos que incluye la información de identificación que muestra que los datos son datos de un punto de vista expandido.

Un soporte de registro puede memorizar, a modo de ejemplo, un flujo de imagen base que se obtiene codificando datos de vídeo de una pluralidad de puntos de vista utilizando un método de codificación predeterminado y que está constituido por datos sin una cabecera de datos que incluye la información de identificación de un punto de vista y un flujo de imagen extendida que está constituido por datos con la cabecera de datos que incluye la información de identificación que muestra que los datos son datos de un punto de vista expandido.

Un dispositivo de reproducción puede incluir, a modo de ejemplo, medios de lectura para leer, desde un soporte de registro, un flujo de imagen base que se obtiene codificando datos de vídeo de una pluralidad de puntos de vista utilizando un método de codificación predeterminado y que está constituido por datos sin una cabecera de datos que incluye información de identificación de un punto de vista y un flujo de imagen extendida que está constituido por datos con la cabecera de datos que incluye la información de identificación que tiene un valor de uno o más que muestra que los datos son datos de un punto de vista expandido y medios de decodificación para realizar un proceso, de forma secuencial, desde los datos de un punto de vista en donde un valor establecido como la información de identificación en la cabecera de datos es pequeño, considerando los datos del flujo de imagen base sin la cabecera de datos como datos en donde se establece un valor de cero como la información de identificación en la cabecera de datos y decodificando los datos del flujo de imagen base antes de decodificar los datos del flujo de imagen extendida.

Un método de reproducción puede incluir, a modo de ejemplo, las etapas de lectura, desde un soporte de registro, de un flujo de imagen base que se obtiene codificando datos de vídeo de una pluralidad de puntos de vista utilizando un método de codificación predeterminado y que está constituido por datos sin una cabecera de datos que incluye información de identificación de un punto de vista y un flujo de imagen extendida que está constituido por datos con

la cabecera de datos que incluye la información de identificación que tiene un valor de uno o más mostrando que los datos son datos de un punto de vista expandido y, en el caso de realizar un proceso, de forma secuencial, a partir de los datos de punto de vista en donde un valor establecido como la información de identificación en la cabecera de datos es pequeño, considerando los datos del flujo de imagen base, sin la cabecera de datos, como datos en donde un valor de cero se establece como la información de identificación en la cabecera de datos y decodificando los datos del flujo de imagen base antes de decodificar los datos del flujo de imagen extendida.

Un programa puede hacer, a modo de ejemplo, que un ordenador ejecute un proceso que incluye las etapas de lectura, desde un soporte de registro, de un flujo de imagen base que se obtiene codificando datos de vídeo de una pluralidad de puntos de vista utilizando un método de codificación predeterminado y que está constituido por datos sin una cabecera de datos que incluye información de identificación de un punto de vista y un flujo de imagen extendida que está constituido por datos con la cabecera de datos que incluye la información de identificación que tiene un valor de uno o más mostrando que los datos son datos de un punto de vista expandido y, en el caso de realizar un proceso, de forma secuencial, a partir de los datos de punto de vista en donde un valor establecido como la información de identificación en la cabecera de datos es pequeño, considerando los datos del flujo de imagen base sin la cabecera de datos como datos en donde un valor de cero se establece como la información de identificación en la cabecera de datos y decodificando los datos del flujo de imagen base antes de decodificar los datos de flujo de imagen extendida.

A modo de ejemplo, los datos de vídeo de una pluralidad de puntos de vista se codifican utilizando un método de codificación predeterminado y un flujo de imagen base que está constituido por datos sin una cabecera de datos que incluye información de identificación de un punto de vista y un flujo de imagen extendida que constituido por datos con la cabecera de datos que incluye la información de identificación mostrando que los datos son datos de un punto de vista expandido son objeto de salida.

A modo de ejemplo, un flujo de imagen base que se obtiene codificando datos de vídeo de una pluralidad de puntos de vista utilizando un método de codificación predeterminado y que está constituido por datos sin una cabecera de datos que incluye información de identificación de un punto de vista y un flujo de imagen extendida que está constituido por datos con la cabecera de datos, que incluye la información de identificación que tiene un valor de uno o más mostrando que los datos son datos de un punto de vista expandido, son objeto de lectura desde un soporte de registro y, en el caso de realizar un proceso, de forma secuencial, a partir de datos de un punto de vista en donde un valor establecido como la información de identificación en la cabecera de datos es pequeño, los datos del flujo de imagen base sin la cabecera de datos se consideran como datos en los que se establece un valor de cero como la información de identificación en la cabecera de datos y los datos del flujo de imagen base se decodifican antes de que se decodifiquen los datos del flujo de imagen extendida.

EFFECTOS VENTAJOSOS DE LA INVENCIÓN

Según la presente invención, un soporte de otro, tal como un BD, que memoriza un flujo de imagen base y un flujo de imagen extendida que se obtiene codificando datos de vídeo de una pluralidad de puntos de vista utilizando un método de codificación predeterminado puede reproducirse en un dispositivo incompatible con la reproducción de datos de vídeo de una pluralidad de puntos de vista.

BREVE DESCRIPCIÓN DE LOS DIBUJOS

La Figura 1 es un diagrama que ilustra una configuración, a modo de ejemplo, de un sistema de reproducción que incluye un dispositivo de reproducción al que se aplica la presente invención.

La Figura 2 es un diagrama que ilustra, a modo de ejemplo, la toma de imágenes.

La Figura 3 es un diagrama de bloques que ilustra una configuración, a modo de ejemplo, de un codificador MVC.

La Figura 4 es un diagrama que ilustra, a modo de ejemplo, imágenes de referencia.

La Figura 5 es un diagrama que ilustra una configuración, a modo de ejemplo, de TS.

La Figura 6 es un diagrama que ilustra otra configuración, a modo de ejemplo, de TS.

La Figura 7 es un diagrama que ilustra otra configuración, a modo de ejemplo, de TS.

La Figura 8 es un diagrama que ilustra la gestión, a modo de ejemplo, de flujos de AV.

La Figura 9 es un diagrama que ilustra estructuras de ruta principal y de ruta secundaria.

La Figura 10 es un diagrama que ilustra, a modo de ejemplo, una estructura de gestión de ficheros registrados en un disco óptico.

- La Figura 11 es un diagrama que ilustra la sintaxis de un fichero de lista de reproducción Playlist.
- La Figura 12 es un diagrama que ilustra, a modo de ejemplo, un método para la utilización reservada para uso futuro en la Figura 11.
- 5 La Figura 13 es un diagrama que ilustra los significados de los valores del parámetro 3D_PL_tipo.
- La Figura 14 es un diagrama que ilustra los significados de los valores del parámetro tipo_vista.
- 10 La Figura 15 es un diagrama que ilustra la sintaxis de la lista de reproducción Playlist () en la Figura 11.
- La Figura 16 es un diagrama que ilustra la sintaxis de ruta secundaria SubPath () en la Figura 15.
- La Figura 17 es un diagrama que ilustra la sintaxis del parámetro SubPlayItem (i) en la Figura 16.
- 15 La Figura 18 es un diagrama que ilustra la sintaxis del parámetro PlayItem () en la Figura 15.
- La Figura 19 es un diagrama que ilustra la sintaxis de STN_tabla () en la Figura 18.
- 20 La Figura 20 es un diagrama de bloques que ilustra una configuración, a modo de ejemplo, de un dispositivo de reproducción.
- La Figura 21 es un diagrama que ilustra una configuración, a modo de ejemplo, de una unidad de decodificador representada en la Figura 20.
- 25 La Figura 22 es un diagrama que ilustra una configuración para realizar en proceso en un flujo de datos de vídeo.
- La Figura 23 es un diagrama que ilustra una configuración para realizar un proceso en un flujo de datos de vídeo.
- 30 La Figura 24 es un diagrama que ilustra otra configuración para realizar un proceso en flujo de datos de vídeo.
- La Figura 25 es un diagrama que ilustra, a modo de ejemplo, Unidades de Acceso.
- La Figura 26 es un diagrama que ilustra otra configuración para realizar un proceso en un flujo de datos de vídeo.
- 35 La Figura 27 es un diagrama que ilustra una configuración de una unidad de combinación y su etapa anterior.
- La Figura 28 es otro diagrama que ilustra una configuración de una unidad de combinación y su etapa anterior.
- 40 La Figura 29 es un diagrama de bloques que ilustra una configuración, a modo de ejemplo, de una unidad de procesamiento de fabricación de programas informáticos.
- La Figura 30 es un diagrama que ilustra, a modo de ejemplo, configuraciones individuales que incluyen la unidad de procesamiento de fabricación de programas informáticos.
- 45 La Figura 31 es un diagrama que ilustra una configuración, a modo de ejemplo, de una unidad generadora de TS de vídeo en 3D proporcionada en un dispositivo de registro.
- La Figura 32 es un diagrama que ilustra otra configuración, a modo de ejemplo, de la unidad generadora de TS de vídeo en 3D proporcionada en el dispositivo de registro.
- 50 La Figura 33 es un diagrama que ilustra otra configuración, a modo de ejemplo, de la unidad generadora de TS de vídeo en 3D proporcionada en el dispositivo de registro.
- 55 La Figura 34 es un diagrama que ilustra una configuración de un lado del dispositivo de reproducción para decodificar Unidades de Acceso.
- La Figura 35 es un diagrama que ilustra un proceso de decodificación.
- 60 La Figura 36 es un diagrama que ilustra una estructura del GOP cerrada.
- La Figura 37 es un diagrama que ilustra una estructura del grupo GOP abierto.
- La Figura 38 es un diagrama que ilustra el número máximo de tramas y campos en un grupo GOP.
- 65 La Figura 39 es un diagrama que ilustra una estructura del GOP cerrada.

La Figura 40 es un diagrama que ilustra una estructura del GOP abierta.

La Figura 41 es un diagrama que ilustra, a modo de ejemplo, una posición de inicio de decodificación establecida para EP_map.

La Figura 42 es un diagrama que ilustra un problema que surge en un caso en donde no se define una estructura del GOP de datos de vídeo de vista dependiente.

La Figura 43 es un diagrama que ilustra el concepto de búsqueda de imagen.

La Figura 44 es un diagrama que ilustra una estructura de un flujo AV registrado en un disco óptico.

La Figura 45 es un diagrama que ilustra, a modo de ejemplo, un flujo de Clip AV

La Figura 46 es un diagrama que ilustra, desde el punto de vista conceptual, el parámetro EP_map correspondiente al flujo Clip AV en la Figura 45.

La Figura 47 es un diagrama que ilustra, a modo de ejemplo, una estructura de datos de un paquete origen indicado por el parámetro SPN_EP_inicio.

La Figura 48 es un diagrama de bloques que ilustra una configuración, a modo de ejemplo, de hardware de un ordenador.

DESCRIPCIÓN DETALLADA DE LAS FORMAS DE REALIZACIÓN DE LA INVENCION

Primera forma de realización

[Configuración, a modo de ejemplo, de un sistema de reproducción]

La Figura 1 es un diagrama que ilustra una configuración, a modo de ejemplo, de un sistema de reproducción incluyendo un dispositivo de reproducción 1 al que se aplica la presente invención.

Según se ilustra en la Figura 1, este sistema de reproducción está constituido conectando el dispositivo de reproducción 1 y un dispositivo de reproducción 3 utilizando un cable HDMI (Interfaz Multimedia de Alta Definición) o similar. Un disco óptico 2, tal como un BD, se carga en el dispositivo de reproducción 1.

Flujos que son necesarios para visualizar una imagen estéreo (una así denominada imagen en 3D) en donde el número de puntos de vista es dos que se registran en el disco óptico 2.

El dispositivo de reproducción 1 es un reproductor compatible con la reproducción de 3D de flujos registrados en el disco óptico 2. El dispositivo de reproducción 1 reproduce flujos registrados en el disco óptico 2 y visualiza una imagen en 3D obtenida mediante la reproducción en el dispositivo de presentación visual 3 constituido por un receptor de televisión o aparato similar. La señal de audio se reproduce también por el dispositivo de reproducción 1 y se proporciona, a la salida, desde un altavoz, o elemento similar, proporcionado en el dispositivo de presentación visual 3.

Varios métodos han sido propuestos como un método de presentación visual de imágenes en 3D. En este caso, un método de presentación visual de tipo 1 y un método de presentación visual de tipo 2, descritos a continuación, se utilizan como un método de presentación visual de imágenes en 3D.

El método de presentación visual de tipo 1 es un método para visualizar una imagen en 3D en donde los datos de la imagen en 3D están constituidos por datos de una imagen observada por un ojo izquierdo (imagen L) y los datos de una imagen observados por un ojo derecho (imagen R) y la imagen L y la imagen R se visualizan de forma alternada.

El método de presentación visual de tipo 2 es un método para visualizar una imagen en 3D mostrando una imagen L y una imagen R que se generan utilizando los datos de una imagen original, que es una imagen que sirve como el original para generar una imagen en 3D y los datos de fondo Depth. Los datos de una imagen en 3D, que se utilizan en el método de presentación visual de tipo 2, están constituidos por los datos de una imagen original y los de fondo Depth que se proporcionan a la imagen original para generar una imagen L y una imagen R.

El método de presentación visual de tipo 1 es un método de presentación visual en donde se necesitan gafas para la visión/escucha. El método de presentación visual de tipo 2 es un método de presentación visual en donde una imagen en 3D puede ser vista/oída sin necesidad de utilizar gafas.

El disco óptico 2 tiene flujos registrados de modo que una imagen en 3D puede visualizarse en los métodos de

presentación visual de tipo 1 y de tipo 2.

Como un método de codificación para registrar dichos flujos en el disco óptico 2, se utiliza la codificación H.264 AVC (Codificación de Vídeo Avanzada)/MVC (Codificación de Vídeo de Vistas Múltiples, a modo de ejemplo).

[Perfil de AVC/MVC H.264]

En AVC/MVC H.264, un flujo de imagen denominado vídeo de vista de base y un flujo de imagen denominado vídeo de vista dependiente se definen a este respecto. En adelante, H.264 AVC/MVC simplemente se referirá como MVC cuando sea necesario.

La Figura 2 es un diagrama que ilustra, a modo de ejemplo, la operación de toma de imágenes.

Según se ilustra en la Figura 2, la toma de imágenes se realiza sobre el mismo sujeto mediante una cámara para imagen L y una cámara para imagen R. Un flujo elemental de vídeo capturado por la cámara para imagen L y la cámara para imagen R es objeto de entrada a un codificador MVC.

La Figura 3 es un diagrama de bloques que ilustra una configuración, a modo de ejemplo, del codificador MVC.

Según se ilustra en la Figura 3, un identificador MVC 11 incluye un identificador H.264/AVC 21, un decodificador H.264/AVC 22, una unidad de cálculo de fondo Depth 23, un codificador de vídeo de vista dependiente 24 y un multiplexor 25.

Un flujo de vídeo nº 1 capturado por la cámara para la imagen L se aplica a la entrada del codificador H.264/AVC 21 y la unidad de cálculo de fondo Depth 23. Además, un flujo de vídeo nº 2 capturado por la cámara para la imagen R es objeto de entrada a la unidad de cálculo de fondo Depth 23 y al codificador de vídeo de vista dependiente 24. El flujo de vídeo nº 2 puede ser objeto de entrada al codificador H.264/AVC 21 y a la unidad de cálculo de fondo Depth 23 y el flujo de vídeo nº 1 puede ser objeto de entrada a la unidad de cálculo de fondo Depth 23 y al codificador de vídeo de vista dependiente 24.

El codificador H.264/AVC 21 codifica el flujo de vídeo nº 1 como un flujo de vídeo de H.264/AVC/alto perfil, a modo de ejemplo. El codificador H.264/AVC 21 proporciona, a la salida, un flujo de vídeo AVC obtenido mediante la codificación, que sirve como un flujo de vídeo de vista de base para el decodificador H.264/AVC 22 y el multiplexor 25.

El decodificador H.264/AVC 22 decodifica el flujo de vídeo AVC suministrado desde el codificador H.264/AVC 21 y proporciona, a la salida, el flujo de vídeo nº 1 obtenido mediante la decodificación al codificador de vídeo de vista dependiente 24.

La unidad de cálculo de fondo Depth 23 calcula el fondo sobre la base del flujo de vídeo nº 1 y el flujo de vídeo nº 2 y proporciona, a la salida, los datos del fondo Depth calculados al multiplexor 25.

El codificador de vídeo de vista dependiente 24 codifica el flujo de vídeo nº 1 suministrado desde el decodificador H.264/AVC 22 y el flujo de vídeo nº 2 que se aplica externamente a su entrada y proporciona, a la salida, un flujo de vídeo de vista dependiente.

La codificación predictiva, utilizando otro flujo como una imagen de referencia, no está permitida en el vídeo de vista de base. Sin embargo, según se ilustra en la Figura 4, la codificación predictiva utilizando vídeo de vista de base como una imagen de referencia está permitida para vídeo de vista dependiente. A modo de ejemplo, en un caso en donde la codificación se realiza con una imagen L siendo el elemento de vídeo de vista de base y con una imagen R siendo el elemento de vídeo de vista dependiente, la cantidad de datos de un flujo de vídeo de vista dependiente, así obtenido, es más pequeña que la cantidad de datos de un flujo de vídeo de vista de base.

Conviene señalar que, puesto que la codificación está basada en H.264/AVC, la predicción en la dirección temporal se realiza sobre vídeo de vista de base. Además, la predicción en la dirección del tiempo se realiza así como la predicción entre vistas de vídeo de vista dependiente. Con el fin de decodificar el vídeo de vista dependiente, es necesario que se finalice previamente la decodificación del vídeo de vista de base correspondiente, que es objeto de referencia durante la codificación.

El codificador de vídeo de vista dependiente 24 proporciona, a la salida, el flujo de vídeo de vista dependiente, que se obtiene mediante la codificación utilizando dicha predicción entre vistas, al multiplexor 25.

El multiplexor 25 realiza la multiplexación de flujo de vídeo de vista de base que se suministra desde el codificador H.264/AVC 21, el flujo de vídeo de vista dependiente (datos de fondo Depth) que se suministra desde la unidad de cálculo de fondo Depth 23 y el flujo de vídeo de vista dependiente que se suministra desde el codificador de vídeo de vista dependiente 24 en un MPEG2 TS, a modo de ejemplo. El flujo de vídeo de vista de base y el flujo de vídeo

de vista dependiente pueden multiplexarse en un MPEG2 TS único o pueden incluirse en MPEG2 TSs separados.

5 El multiplexor 25 proporciona, a la salida, el TS generado (MPEG2 TS). La salida de TS desde el multiplexor 25 se registra en el disco óptico 2, junto con otros datos de gestión, en un dispositivo de registro, y se proporciona al dispositivo de reproducción 1 mientras se registra en el disco óptico 2.

10 En un caso en donde el vídeo de vista dependiente que se utiliza junto con el vídeo de vista de base, en el método de presentación visual de tipo 1, necesita distinguirse del vídeo de vista dependiente (Depth) que se utiliza junto con el vídeo de vista de base en el método de presentación visual tipo 2, el primero se refiere como un vídeo de vista D1 y el último se refiere como un vídeo de vista D2.

15 Además, la reproducción en 3D en el método de presentación visual de tipo 1, que se realiza utilizando vídeo de vista de base y vídeo de vista D1, se refiere como una reproducción B-D1. La reproducción en 3D en el método de presentación visual de tipo 2, que se realiza utilizando vídeo de vista de base y vídeo de vista D2, se refiere como una reproducción B-D2.

20 En el caso de realización de una reproducción B-D1 en respuesta a una instrucción o elemento similar proporcionado desde un usuario, el dispositivo de reproducción 1 efectúa la lectura de un flujo de vídeo de vista de base y un flujo de vídeo de vista D1 desde el disco óptico 2 y los reproduce.

Además, en el caso de realización de una reproducción B-D2, el dispositivo de reproducción 1 efectúa la lectura de un flujo de vídeo de vista de base y un flujo de vídeo de vista D2 desde el disco óptico 2 y los reproduce.

25 Además, en el caso de realizar la reproducción de una imagen en 2D ordinaria, el dispositivo de reproducción 1 lee solamente un flujo de vídeo de vista de base desde el disco óptico 2 y los reproduce.

30 Puesto que el flujo de vídeo de vista de base es un flujo de vídeo AVC codificado con H.264/AVC y en consecuencia, cualquier reproductor compatible con el formato de BD puede reproducir el flujo de vídeo de vista de base para la presentación visual de una imagen en 2D.

35 En adelante, se proporcionará una descripción principalmente de un caso en donde el vídeo de vista dependiente es el vídeo de vista D1. Una mención simple como vídeo de vista dependiente que corresponde al vídeo de vista D1. De forma similar, el vídeo de D2 se registra en el disco óptico 2 y se reproduce de la misma manera que para el vídeo de vista D1.

[Configuración, a modo de ejemplo, de TS]

La Figura 5 es un diagrama que ilustra una configuración, a modo de ejemplo, de TS.

40 Los flujos de vídeo de vista de base, vídeo de vista dependiente, audio primario, PG base, PG dependiente, IG base e IG dependiente son objeto de multiplexación en un TS principal en la Figura 5. De este modo, el flujo de vídeo de vista dependiente puede incluirse en el TS principal junto con el flujo de vídeo de vista de base.

45 Un TS principal y un TS secundario se registran en el disco óptico 2. El TS principal es un TS que incluye al menos un flujo de vídeo de vista de base. El TS secundario es un TS que incluye flujos distintos del flujo de vídeo de vista de base y que se utilizan junto con el TS principal.

50 Los flujos de vista de base y de vista dependiente se preparan también para PG e IG según se describe a continuación, de modo que la presentación visual en 3D esté disponible como en vídeo.

55 El plano de vista de base de PG y de IG, obtenido mediante la decodificación de los flujos individuales, se visualiza combinándose con el plano de vídeo de vista de base que se obtiene decodificando el flujo de vídeo de vista de base. De forma similar, el plano de vista dependiente de PG y de IG se visualiza combinándose con el plano de vídeo de vista dependiente obtenido mediante la decodificación del flujo de vídeo de vista dependiente.

60 A modo de ejemplo, en un caso en donde el flujo de vídeo de vista de base es un flujo de una imagen L y el flujo de vídeo de vista dependiente es un flujo de una imagen R, los flujos de vista de base de PG y de IG son flujos de gráficos de la imagen L. Además, el flujo de PG y el flujo de IG de vista dependiente son flujos de gráficos de la imagen R.

65 Por otro lado, en un caso en donde el flujo de vídeo de vista de base es un flujo de una imagen R y el flujo de vídeo de vista dependiente es un flujo de una imagen L, los flujos de vista de base de PG y de IG son flujos de gráficos de la imagen R. Además, el flujo PG y el flujo IG de vista dependiente son flujos de gráficos de la imagen L.

La Figura 6 es un diagrama que ilustra otra configuración, a modo de ejemplo, de TS.

Los flujos de vídeo de vista de base y de vídeo de vista dependiente se multiplexan en el TS principal representado en la Figura 6.

5 Por otro lado, los flujos de audio primario, PG base, PG dependiente, IG base e IG dependiente se multiplexan en el Sub TS.

De este modo, los flujos de vídeo pueden multiplexarse en el TS principal y los flujos de PG y de IG pueden multiplexarse en el Sub TS.

10 La Figura 7 es un diagrama que ilustra otra configuración, a modo de ejemplo, de TS.

Los flujos de vídeo de vista de base, audio primario, PG base, PG dependiente, IG base e IG dependiente se multiplexan en el TS principal en la parte A ilustrada en la Figura 7.

15 Por otro lado, un flujo de vídeo de vista dependiente está incluido en el Sub TS.

De este modo, un flujo de vídeo de vista dependiente puede incluirse en un TS diferente de un TS que incluye un flujo de vídeo de vista de base.

20 Los flujos de vídeo de vista de base, audio primario, PG e IG se multiplexan en el TS principal en la parte B representada en la Figura 7. Por otro lado, los flujos de vídeo de vista dependiente, PG base, PG dependiente, IG base e IG dependiente se multiplexan en el Sub TS.

25 Los PG e IG incluidos en el TS principal son flujos para la reproducción en 2D. Los flujos incluidos en el Sub TS, son flujos para la reproducción en 3D.

De este modo, el flujo de PG y el flujo de IG no pueden compartirse por la reproducción en 2D y la reproducción en 3D.

30 Según se describió con anterioridad, el flujo de vídeo de vista de base y el flujo de vídeo de vista dependiente pueden incluirse en diferentes MPEG2 TSs. Se proporcionará una descripción de una ventaja del caso de registro del flujo de vídeo de vista de base y el flujo de vídeo de vista dependiente mientras se hace que los flujos sean incluidos en diferentes MPEG2 TSs.

35 A modo de ejemplo, se supone un caso en donde una tasa binaria permitida para la multiplexación en un MPEG2 TS único está limitada. En este caso, cuando el flujo de vídeo de vista de base y el flujo de vídeo de vista dependiente están incluidos en un TS MPEG2 único, las tasas binarias de los respectivos flujos necesitan reducirse con el fin de satisfacer la restricción. En consecuencia, se degrada la calidad de la imagen.

40 La necesidad para reducir la tasa binaria se elimina haciendo que los flujos sean incluidos en diferentes MPEG2 TSs, de modo que se pueda evitar la degradación de la calidad de la imagen.

[Formato de aplicación]

45 La Figura 8 es un diagrama que ilustra, a modo de ejemplo, la gestión de flujos AV realizada por el dispositivo de reproducción 1.

50 La gestión de los flujos AV se realiza utilizando dos capas de PlayList y Clip, según se ilustra en la Figura 8. Los flujos de AV pueden registrarse en una unidad de memorización local del dispositivo de reproducción 1 así como en el disco óptico 2.

55 En este caso, un par de un flujo AV y de Clip Information, que es información que acompaña al flujo AV, se considera como un solo objeto, que se refiere como Clip. En adelante, un fichero que memoriza un flujo AV se refiere como un fichero de flujo AV. además, un fichero que memoriza Clip Information se refiere como un fichero Clip Information.

60 Un flujo AV se representa en un eje de tiempos y un punto de acceso de cada Clip se especifica principalmente por una marca de tiempo en PlayList. Un fichero Clip Information se utiliza para encontrar una dirección en la que ha de iniciarse la decodificación en el flujo AV, a modo de ejemplo.

65 PlayList es un conjunto de secciones de reproducción de un flujo AV. Una sección de reproducción en un flujo AV se denomina PlayItem. PlayItem se expresa por un par de un punto de entrada IN y un punto de salida OUT de una sección de reproducción en el eje de tiempos. Según se ilustra en la Figura 8, PlayList está constituida por una o una pluralidad de PlayItems.

La primera PlayList desde la izquierda en la Figura 8 está constituida por dos PlayItems y la primera parte mitad y la

última parte mitad del flujo AV, incluidas en el Clip a la izquierda, se refieren mediante dichos dos PlayItems, respectivamente.

5 La segunda PlayList desde la izquierda está constituida por un PlayItem y el flujo AV completo incluido en el Clip a la derecha se refiere en consecuencia.

La tercera PlayList desde la izquierda está constituida por dos PlayItems y una parte determinada del flujo AV que se incluye en el Clip a la izquierda y una parte determinada del flujo AV que se incluye en el Clip a la derecha se refieren por dichos dos PlayItems, respectivamente.

10 A modo de ejemplo, en un caso en donde el PlayItem a la izquierda, que se incluye en la primera PlayList desde la izquierda, se especifica como un objetivo de reproducción por intermedio de un programa de navegación en disco, la reproducción de la primera parte mitad del flujo AV que se incluye en el Clip a la izquierda, que se refiere por PlayItem, es objeto de realización. De este modo, se utilizan PlayLists como información de gestión de reproducción para gestionar la reproducción de flujos AV.

En PlayList, una ruta de reproducción constituida por un conjunto matricial de uno o más PlayItems, se refiere como la Ruta Principal.

20 Además, en PlayList, una ruta de reproducción constituida por un conjunto matricial de uno o más SubPlayItems paralelos con la ruta principal se refiere como una ruta secundaria Sub Path.

La Figura 9 es un diagrama que ilustra estructuras de la ruta principal y de la ruta secundaria.

25 PlayList puede tener una ruta principal y una o más rutas secundarias.

El flujo de vídeo de vista de base, anteriormente descrito, se gestiona como un flujo al que se refiere el PlayItem que constituye una ruta principal. Además, el flujo de vídeo de vista dependiente se gestiona como un flujo que se refiere al SubPlayItem que constituye una ruta secundaria.

30 La PlayList en la Figura 9 tiene una sola ruta principal constituida por un conjunto matricial de tres PlayItems y tres Sub Paths como rutas secundarias.

35 Se establecen identificadores IDs para los PlayItems que constituyen la ruta principal en orden desde la cabecera. Los identificadores IDs se establecen también para las rutas secundarias, Subpath_id = 0, Subpath_id = 1 y Subpath_id = 2 en orden desde la cabecera.

40 En la realización, a modo de ejemplo, ilustrada en la Figura 9, un SubPlayItem se incluye en la ruta secundaria con Subpath_id = 0 y dos SubPlayItems se incluyen en la ruta secundaria con Subpath_id = 1. Además, un SubPlayItem se incluye en la ruta secundaria con Subpath_id = 2.

Un flujo AV se refiere por un PlayItem que incluye al menos un flujo de vídeo (datos de imagen principal).

45 Además, el flujo AV Clip puede incluir o puede no incluir, uno o más flujos de audio que se reproducen por la misma temporización que (en sincronización con) el flujo de vídeo incluido en el flujo AV Clip.

El flujo AV Clip puede incluir, o puede no incluir, uno o más flujos de datos de subtítulos de mapas de bits (PG (Gráfico de Presentación)) que se reproducen en sincronización con el flujo de vídeo incluido en el flujo AV Clip.

50 El flujo AV Clip puede incluir, o puede no incluir, uno o más flujos de IG (Gráfico interactivo) que se reproducen en sincronización con el flujo de vídeo incluido en un fichero de flujo AV Clip. El flujo de IG se utiliza para visualizar un gráfico, tal como una tecla pulsada por un usuario.

55 En el flujo AV Clip referido por un PlayItem, un flujo de vídeo, cero o más flujos de audio que se reproducen en sincronización con cero o más flujos PG y cero o más flujos IG son objeto de multiplexación.

Además, el SubPlayItem se refiere a un flujo de vídeo, un flujo de audio, un flujo PG o similar de un flujo diferente del flujo AV Clip referido por PlayItem (otro flujo).

60 La gestión de flujos AV utilizando dicha PlayList, PlayItem y SubPlayItem se describe en la publicación de solicitud de patente japonesa no examinada número 2008-252740 y la publicación de solicitud de patente japonesa no examinada número 2005-348314, a modo de ejemplo.

[Estructura de directorios]

65 La Figura 10 es un diagrama que ilustra, a modo de ejemplo, una estructura de gestión de ficheros registrado en el

disco óptico 2.

Según se ilustra en la Figura 10, los ficheros se gestionan, de forma jerárquica, por una estructura de directorios. Un directorio raíz se crea en el disco óptico 2. Por debajo del directorio raíz existe una gama que se gestiona por un sistema de registro/reproducción.

Un directorio de BDMV está situado bajo el directorio raíz.

Un fichero de Índice, que es un fichero con un nombre "Index.bdmv" que se establece, y un fichero MovieObject, que es un fichero con un nombre "MovieObject.bdmv" se memorizan inmediatamente debajo del directorio de BDMV.

Un directorio de seguridad BACKUP, un directorio de PLAYLIST, un directorio de CLIPINF, un directorio de STREAM, etc., se proporciona bajo el directorio de BDMV.

Los ficheros de PlayList, que describen PlayList, se memorizan en el directorio PLAYLIST. Un nombre constituido por un número de cinco dígitos y una extensión ".mpls" se establece para cada fichero de PlayList. Un nombre de fichero "00000.mpls" se establece para un fichero de PlayList ilustrado en la Figura 10.

Los ficheros de Clip Information se memorizan en el directorio Clip INF. Un nombre constituido por un número de cinco dígitos y una extensión ".clpi" se establecen para cada fichero Clip Information.

Los nombres de ficheros "00001.clpi", "00002.clpi" y "00003.clpi" se establecen para los tres ficheros Clip Information ilustrados en la Figura 10, respectivamente. En adelante, los ficheros Clip Information se refieren como ficheros clpi cuando sea necesario.

A modo de ejemplo, el fichero clpi "00001.clpi" es un fichero en donde se describe la información sobre el Clip del vídeo de vista de base.

El fichero clpi "00002.clpi" es un fichero en el que se describe la información sobre el Clip del vídeo de vista D2.

El fichero clpi "00003.clpi" es un fichero en el que se describe la información sobre Clip en el vídeo de vista D1.

Ficheros de flujos se memorizan en el directorio STREAM. Un nombre constituido por un número de cinco dígitos y una extensión ".m2ts" o un nombre constituido por un nombre de cinco dígitos y una extensión ".ilvt" se establece para cada fichero de flujos. En adelante, un fichero para el que se establece la extensión ".m2ts" se refiere como un fichero m2ts cuando sea necesario. Además, un fichero al que se establece la extensión ".ilvt" es referido como un fichero ilvt.

El fichero m2ts "00001.m2ts" es un fichero para reproducción en 2D. La lectura de un flujo de vídeo de vista de base se realiza especificando este fichero.

El fichero m2ts "00002.m2ts" es un fichero de un flujo de vídeo de vista D2 y el fichero m2ts "00003.m2ts" es un fichero de un flujo de vídeo de vista D1.

El fichero ilvt "10000.ilvt" es un fichero para reproducción de B-D1. La lectura de un flujo de vídeo de vista de base y un flujo de vídeo D1 se realiza especificando este fichero.

El fichero ilvt "20000.ilvt" es un fichero para reproducción de B-D2. La lectura de un flujo de vídeo de vista de base y un flujo de vídeo de vista D2 se realiza especificando este fichero.

Además de los directorios ilustrados en la Figura 10, un directorio que memoriza un fichero de un flujo de audio y similar se proporcionan bajo el directorio BDMV.

[Sintaxis de cada elemento de datos]

La Figura 11 es un diagrama que ilustra la sintaxis de un fichero de PlayList.

El fichero de PlayList es un fichero que se memoriza en el directorio PLAYLIST en la Figura 10 y que tiene una extensión establecida ".mpls".

El parámetro type_indicator en la Figura 11 representa el tipo de un fichero "xxxx.mpls".

version_number representa el número de versión de "xxxx.mpls". version_number está constituido por un número de cuatro dígitos. A modo de ejemplo, "0240" que representa a "versión 3D Spec" se establece para un fichero de PlayList para la reproducción en 3D.

PlayList_start_address representa la dirección de inicio de PlayList () con el número de bytes relativos desde el primer byte del fichero de PlayList siendo la unidad.

5 PlayListMark_start_address representa la dirección de inicio de PlayListMark () con el número de bytes relativos desde el primer byte del fichero PlayList siendo la unidad.

ExtensionData_start_address representa la dirección de inicio de ExtensionData () con el número de bytes relativos desde el primer byte de fichero PlayList siendo la unidad.

10 160 bits del tipo reservado para uso futuro se incluye después de la dirección ExtensionData_start_address.

Los parámetros relativos al control de la reproducción de una PlayList, tales como restricciones de la reproducción, se memorizan en ApplInfoPlayList ().

15 Los parámetros con respecto a una ruta principal, una ruta secundaria, etc., se memorizan en PlayList (). El contenido de PlayList () se describirá a continuación.

20 La información de Marca de una PlayList, es decir, información sobre una marca, que es un destino de salto (punto de salto) en una operación del usuario o una orden para dar instrucciones para el salto o similar, se memoriza en PlayListMark ():

Los datos privados pueden insertarse en ExtensionData ().

25 La Figura 12 es un diagrama que ilustra una descripción, a modo de ejemplo, de un fichero PlayList.

Según se ilustra en la Figura 12, los parámetros 3D_PL_type de 2 bits y view_type de 1 bit se describen en el fichero de PlayList.

30 3D_PL_type representa el tipo de PlayList.

View_type representa si el flujo de vídeo de vista de base del que se gestiona la reproducción por una PlayList es un flujo de una imagen L (vista L) o un flujo de una imagen R (vista R).

35 La Figura 13 es un diagrama que ilustra los significados de los valores de 3D_PL_type.

El valor 00 de 3D_PL_type representa una PlayList para la reproducción en 2D.

El valor 01 de 3D_PL_type representa una PlayList para la reproducción B-D1 de la reproducción en 3D.

40 El valor 10 de 3D_PL_type representa una PlayList para la reproducción B-D2 de la reproducción en 3D.

45 A modo de ejemplo, en un caso en donde el valor de 3D_PL_type es 01 o 10, la información de 3DPlayList se registra en ExtensionData () del fichero PlayList. A modo de ejemplo, como la información de 3DPlayList la información sobre la lectura de un flujo de vídeo de vista de base y un flujo de vídeo de vista dependiente, desde el disco óptico 2, es objeto de registro.

La Figura 14 es un diagrama que ilustra los significados de los valores de view_type.

50 En el caso de realización de una reproducción en 3D, el valor 0 de view_type representa que un flujo de vídeo de vista de base es un flujo de vídeo L. En el caso de realización de una reproducción en 2D, el valor 0 de view_type representa un flujo de vídeo de vista de base es un flujo de vídeo AVC.

El valor 1 de view_type representa que un flujo de vídeo de vista de base es un flujo de vista R.

55 La descripción de view_type en el fichero PlayList permite al dispositivo de reproducción 1 identificar si el flujo de vídeo de vista de base es un flujo de vista L o un flujo de vista R.

60 A modo de ejemplo, en un caso en donde una señal de vídeo es proporcionada, a la salida, al dispositivo de presentación visual 3 por intermedio de un cable HDMI, el dispositivo de reproducción 1 puede requerirse para proporcionar una señal de vista L y una señal de vista R mientras se distinguen entre sí.

Haciendo que el dispositivo de reproducción 1 sea capaz de identificar si un flujo de vídeo de vista de base es un flujo de vista L o un flujo de vista R, el dispositivo de reproducción 1 puede proporcionar, a la salida, una señal de vista L y una señal de vista R mientras se distinguen entre sí.

65 La Figura 15 es un diagrama que ilustra la sintaxis de PlayList () en la Figura 11.

La longitud Length es un número entero de 32 bit sin signo, que indica el número de bytes desde inmediatamente después de este campo de longitud al final de PlayList (). Es decir, la longitud representa el número de bytes desde los reservados para uso futuro a y de la lista de reproducción PlayList.

5 16-bit reserved_for_future_use se prepara después de la longitud.

number_of_PlayItems es un campo de 16 bits que muestra el número de PlayItem existente en la PlayList. En el caso, a modo de ejemplo, ilustrado en la Figura 9, el número de PlayItems es 3. El valor de PlayItem_id se asigna desde 0 en orden en donde PlayItem () aparece en la PlayList. A modo de ejemplo, PlayItem_Id = 0, 1 y 2 en la

10 Figura 9 se asignan a este respecto.

number_of_SuPaths es un campo de 16 bits que muestra el número de rutas secundarias existentes en la PlayList. En el caso, a modo de ejemplo, ilustrado en la Figura 9, el número de rutas secundarias es 3. El valor de SubPath_id se asigne desde 0 en orden en donde SubPath () aparece en la PlayList. A modo de ejemplo, Subpath_id = 0, 1 y 2 en la Figura 9 están asignados. En adelante, PlayItem () se refiere por el número de PlayItem y SubPath () se refiere por el número de rutas secundarias.

15

La Figura 16 es un diagrama que ilustra la sintaxis de SubPath () en la Figura 15.

20 La longitud es un número entero de 32 bits sin signo, que indica el número de bits desde inmediatamente después de este campo de longitud al final de SubPath (). Es decir, la longitud representa el número de bytes desde el reservado para uso futuro a y desde PlayList.

16-bit reserved_for_future_use se prepara después de la longitud.

25 SubPath_type es un campo de 8 bits que muestra el tipo de aplicación de SubPath. SubPath_type se utiliza para indicar el tipo, a modo de ejemplo, si la ruta secundaria es de audio, subtítulos de mapas de bits o subtítulos de texto.

30 15-bit reserved_for_future_use se prepara después de SubPath_type.

is_repeat_SubPath es un campo de 1 bit que especifica un método de reproducción de la ruta secundaria e indica si la reproducción de la ruta secundaria se realiza repetidamente durante la reproducción de la ruta principal o la reproducción de la ruta secundaria se realiza solamente una vez. A modo de ejemplo, este campo se utiliza en un caso en donde las temporizaciones de reproducción de Clip referidas por la ruta principal y Clip referidas por la ruta secundaria son diferentes (en un caso en donde la ruta principal se utiliza como una ruta de presentación de imágenes fijas y en donde la ruta secundaria se utiliza como una ruta de audio que sirve como BGM, a modo de ejemplo).

35

40 8-bit reserved_for_future_use se prepara después de is_repeat_SubPath

number_of_SubPlayItems es un campo de 8 bits que muestra el número de SubPlayItems (número de entradas) existente en una ruta secundaria. A modo de ejemplo, number_of_SubPlayItems de SubPlayItems de la SubPath_id = 0 en la Figura 9 es 1 y number_of_SubPlayItems de la SubPlayItems de la SubPath_id = 1 es 2. En adelante, SubPlayItems () se refiere por el número de SubPlayItems.

45

La Figura 17 es un diagrama que ilustra la sintaxis de SubPlayItems (i) en la Figura 16.

La longitud es un número entero de 16 bits sin signo, que indica el número de bytes desde inmediatamente después de este campo de longitud al final de SubPlayItems ().

50

SubPlayItems (i) en la Figura 17, se describe para ambos casos en donde SubPlayItems se refiere a un Clip y en donde SubPlayItems se refiere a una pluralidad de Clips.

55 Se proporcionará una descripción de un caso en donde SubPlayItems se refiere a un solo Clip.

Clip_Information_file_name [0] representa el Clip al que se refiere.

Clip_codec_identifier [0] representa un método de codificación-decodificación de Clip. Reserved_for_future_use se incluye después de Clip_codec_identifier [0].

60

is_multi_Clip_entries es un indicador que muestra si se registran, o no, múltiples Clips. Si el indicador de is_multi_Clip_entries se mantiene la sintaxis en un caso en donde SubPlayItems se refiere a una pluralidad de Clips se refiere en este caso.

65

ref_to_STC_id [0] es información sobre un punto discontinuo de STC (un punto discontinuo de la base de tiempos

del sistema).

SubPlayltems_IN_time representa la posición de partida de una sección de reproducción de la ruta secundaria y SubPlayltems_OUT_time representa la posición final.

5 sync_Playltem_id y sync_start_PTS_of_Playltem representa el tiempo en que la ruta secundaria inicia la reproducción en el eje de tiempos de la Ruta Principal.

10 SubPlayltems_IN_time, SubPlayltems_OUT_time, sync_Playltem_id y sync_start_PTS_of_Playltem se suelen utilizar en el Clip referido por el SubPlayltems.

Se proporcionará una descripción de un caso en donde "if (is_multi_Clip_entries == 1b" y en donde el SubPlayltems se refiere a una pluralidad de Clips.

15 num_of_Clip_entries representa el número de Clips a referirse. El número de Clip_Information_file_name [SubClip_entry_id] especifica el número de Clips excepto Clip_Information_file_name [0].

Clip_codec_identifier [SubClip_entry_id] representa un método de codificación-decodificación de Clip.

20 ref_to_STC_id [SubClip_entry_id] es información sobre un punto discontinuo de STC (un punto discontinuo de base de tiempos del sistema), reserved_for_future_use está incluido después de ref_to_STC_id [SubClip_entry_id].

La Figura 18 es un diagrama que ilustra la sintaxis de Playltem () en la Figura 15.

25 La longitud es un número entero de 16 bits sin signo, que indica el número de bytes desde inmediatamente después de este campo de longitud al final de Playltem ().

30 Clip_Information_file_name [0] representa el nombre de un fichero Clip Information de Clip referido por el Playltem. Conviene señalar que el mismo número de 5 dígitos se incluye en el nombre de fichero de un fichero mt2s que incluye el Clip y el nombre de fichero del fichero Clip Information correspondiente.

35 Clip_codec_identifier [0] representa un método de codificación-decodificación del Clip. Reserved_for_future_use se incluye después de Clip_codec_identifier [0]. is_multi_angle y connection_condition se incluyen después de reserved_for_future_use.

ref_to_STC_id [0] es información sobre un punto discontinuo de STC (un punto discontinuo de la base de tiempos del sistema).

40 IN_time representa la posición de partida de la sección de reproducción del Playltem y OUT_time representa la posición final.

UO_mask_table (), Playltem_random_access_mode y still_mode están incluidos después de OUT_time.

45 STN_table () incluye información sobre un flujo AV referido por un Playltem objetivo. Además, en un caso en donde existe una ruta secundaria Sub Path que ha de reproducirse mientras está asociada con el Playltem objetivo, se incluye también información sobre un flujo AV referido por el Playltem que constituye la ruta secundaria Sub Path.

La Figura 19 es un diagrama que ilustra la sintaxis de STN_table () en la Figura 18.

50 STN_table () se establece como el atributo de Playltem.

La longitud es un número entero de 16 bits sin signo, que indica el número de bytes desde inmediatamente después de este campo de longitud al final de STN_table (). 16-bit reserved_for_future_use se prepara después de la longitud.

55 number_of_video_stream_entries representa el número de flujos que se introducen (registran) en STN_table () y que se proporcionan con el identificador vídeo_stream_id.

60 vídeo_stream_id es información para identificar un flujo de vídeo. A modo de ejemplo, un flujo de vídeo de vista de base se especifica por este vídeo_stream_id.

El identificador ID de un flujo de vídeo de vista dependiente puede definirse en STN_table () o puede obtenerse mediante cálculo, p.e., sumando un valor predeterminado al identificador ID del flujo de vídeo de vista de base.

65 vídeo_stream_number es un número de flujo de vídeo que se utiliza para la conmutación de vídeo y que es objeto de vista desde un usuario.

number_of_audio_stream_entries representa el número de flujos del primer flujo de audio provisto de audio_stream_id, que se introduce en la STN_table (). audio_stream_id es información para identificar un flujo de audio y audio_stream_number es un número de flujo de audio que se utiliza para la conmutación de audio y que es objeto de vista desde el usuario.

5 number_of_audio_stream2_entries representa el número de flujos del segundo flujo de audio provisto con audio_stream_id2, que se introduce en STN_table (). audio_stream_id2 es información para identificar un flujo de audio y audio_stream_number es un número de flujo de audio que se utiliza para la conmutación de audio y que es objeto de vista desde el usuario. En esta realización, a modo de ejemplo, se puede conmutar el audio que se va a reproducir.

15 number_of_PG_txtST_stream_entries representa el número de flujos provistos del identificador PG_txtST_stream_id que se introduce en la STN_table (). Entre ellos, un flujo PG y un fichero de subtítulos de texto (txtST) obtenido realizando la codificación de longitud en los subtítulos de mapas de bits 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 flujo de subtítulos que se utiliza para la conmutación de subtítulos y que es objeto de vista desde el usuario.

20 number_of_IG_stream_entries representa el número de flujos provistos del identificador IG_stream_id, que se introducen en STN_table (). Entre ellos, se introduce un flujo IG. IG_stream_id es información para identificar el flujo de IG e IG_stream_number es un número de flujos de gráficos que se utiliza para conmutación de gráficos y que es objeto de vista desde el usuario.

25 Los identificadores IDs de un TS principal y un TS secundario se registran también en STN_table (). Se describe en stream_attribute () que su identificador ID no es el ID de un flujo elemental sino el ID de un TS.

[Configuración, a modo de ejemplo, del dispositivo de reproducción 1]

30 La Figura 20 es un diagrama de bloques que ilustra una configuración, a modo de ejemplo, del dispositivo de reproducción 1.

Un controlador 51 ejecuta un programa de control proporcionado por anticipado, con el fin de controlar la operación completa del dispositivo de reproducción 1.

35 A modo de ejemplo, el controlador 51 controla una unidad de disco 52 para la lectura de un fichero lista de reproducción para la reproducción en 3D. Además, el controlador 51 hace que el TS principal y un TS secundario a leerse sobre la base de identificadores IDs registrados en STN_table y a suministrarse a una unidad de decodificador 56.

40 La unidad de disco 52 lee datos desde el disco óptico 2 en conformidad con el control realizado por el controlador 51 y proporciona, a la salida, los datos leídos al controlador 51, una memoria 53 o la unidad de decodificador 56.

La memoria 53 memoriza datos que son necesarios para el controlador 51 para la ejecución de varios procesos cuando sea necesario.

45 Un dispositivo de memorización local 54 está constituido por, a modo de ejemplo, un HDD (Unidad de Disco Duro). Un flujo de vídeo de vista dependiente o similar descargado desde un servidor 72 se registra en el dispositivo de memorización local 54. El flujo registrado en el dispositivo de memorización local 54 se suministra también a la unidad de decodificador 56 cuando sea necesario.

50 Una interfaz de Internet 55 realiza la comunicación con el servidor 72 por intermedio de una red 71 en conformidad con el control realizado por el controlador 51 y suministra los datos descargados desde el servidor 72 al dispositivo de memorización local 54.

55 Los datos para actualizar los datos registrados en el disco óptico 2 se descargan desde el servidor 72. Activando el flujo de vídeo de vista dependiente descargado a utilizarse junto con un flujo de vídeo de vista de base registrado en el disco óptico 2, se puede realizar la reproducción en 3D de contenido diferente al contenido en el disco óptico 2. Cuando se descarga el flujo de vídeo de vista dependiente, se actualiza también el contenido de la Playlist cuando sea necesario.

60 La unidad de decodificador 56 decodifica el flujo suministrado desde la unidad de disco 52 o el dispositivo de memorización local 54 y proporciona, a la salida, una señal de vídeo obtenida, de este modo, al dispositivo de presentación visual 3. Una señal de audio se proporciona también al dispositivo de presentación visual 3 por intermedio de una ruta predeterminada.

65 Una unidad de entrada operativa 57 incluye dispositivos de entrada, tales como un botón, una tecla, panel táctil, mando giratorio *jog dial* y ratón y una unidad de recepción para recibir una señal tal como una radiación infrarroja

transmitida desde un dispositivo de mando a distancia predeterminado. La unidad de entrada operativa 57 detecta una operación del usuario y suministra una señal que representa el contenido de la operación detectada al controlador 51.

5 La Figura 21 es un diagrama que ilustra una configuración, a modo de ejemplo, de la unidad de decodificador 56.

La Figura 21 ilustra una configuración para procesar una señal de vídeo. En la unidad de decodificador 56, se realiza también un proceso de decodificación de una señal de audio. Un resultado del proceso de decodificación realizado sobre una señal de audio se proporciona al dispositivo de presentación visual 3 a través de una ruta que no está
10 ilustrada.

Un filtro de PID 101 identifica si un TS suministrado desde la unidad de disco 52 o el dispositivo de memorización local 54 es un TS principal o un TS secundario sobre la base de los identificadores PIDs de paquetes que constituyen el TS o el identificador ID del flujo. El filtro PID 101 proporciona un TS principal a una memoria intermedia 102 y proporciona, a la salida, un Sub TS a una memoria intermedia 103.
15

Un filtro PID 104 efectúa la lectura secuencial de los paquetes del TS principal memorizados en la memoria intermedia 102 y los clasifica sobre la base de los PIDs.

20 A modo de ejemplo, el filtro de PID 104 proporciona los paquetes que constituyen un flujo de vídeo de vista de base incluido en el TS principal a una memoria intermedia de vídeo B 106 y proporciona a la salida, los paquetes que constituyen un flujo de vídeo de vista dependiente a un conmutador 107.

Además, el filtro PID 104 proporciona los paquetes que constituyen un flujo IG de base incluidos en el TS principal o un conmutador 114 y proporciona, a la salida, los paquetes que constituyen un flujo de IG dependiente a un conmutador 118.
25

El filtro PID 104 proporciona, a la salida, los paquetes que constituyen un flujo PG base incluido en el TS principal o un conmutador 122 y proporciona, a la salida, los paquetes que constituyen un flujo de PG dependiente a un conmutador 126.
30

Según se describe con referencia a la Figura 5, los flujos de vídeo de vista de base, vídeo de vista dependiente, PG base, PG dependiente, IG base e IG dependiente pueden ser objeto de multiplexación en un TS principal.

35 Un filtro PID 105 efectúa la lectura secuencial del paquete del TS secundario memorizado en la memoria intermedia 103 y los clasifica sobre la base de los PIDs.

A modo de ejemplo, el filtro PID 105 proporciona, a la salida, los paquetes que constituyen un flujo de vídeo de vista dependiente incluido en el TS secundario al conmutador 107.
40

Además, el filtro PID 105 proporciona, a la salida, los paquetes que constituyen un flujo IG de base incluido en el sub TS al conmutador 114 y proporciona, a la salida, los paquetes que constituyen un flujo IG dependiente al conmutador 118.

45 El filtro PID 105 proporciona los paquetes que constituyen un flujo de PG de base incluido en el Sub TS al conmutador 122 y proporciona, a la salida, los paquetes que constituyen un flujo de PG dependiente al conmutador 126.

Según se describe con referencia a la Figura 7, un flujo de vídeo de vista dependiente puede incluirse en un Sub TS. Además, según se describe con referencia a la Figura 6, los flujos de PG de base, PG dependiente, IG base e IG dependiente pueden multiplexarse en un Sub TS.
50

El conmutador 107 proporciona los paquetes que constituyen el flujo de vídeo de vista dependiente suministrado desde el filtro PID 104 o el filtro PID 105 a una memoria intermedia de vídeo D 108.
55

Un conmutador 109 efectúa la lectura secuencial de los paquetes de vídeo de vista de base memorizado en la memoria intermedia de vídeo B 106 y los paquetes de vídeo de vista dependiente memorizados en la memoria intermedia de vídeo D 108 en conformidad con la información temporal que especifica la temporización de la decodificación. Una información temporal idéntica se establece para un paquete que memoriza los datos de una determinada imagen de vídeo de vista de base y un paquete que memoriza los datos de imagen de vídeo de vista dependiente correspondiente.
60

El conmutador 109 proporciona los paquetes objeto de lectura desde la memoria intermedia de vídeo B 106 con la memoria intermedia de vídeo D 108 a un decodificador de vídeo 110.
65

El decodificador de vídeo 110 decodifica los paquetes suministrados desde el conmutador 109 y proporciona, a la

salida, los datos de vídeo de vista de base o vídeo de vista dependiente que se obtienen mediante la decodificación a un conmutador 111.

5 El conmutador 111 proporciona los datos obtenidos mediante la decodificación de los paquetes de vídeo de vista de base a una unidad generadora de plano de vídeo B 112 y proporciona, a la salida, los datos obtenidos mediante la decodificación de los paquetes de vídeo de vista dependiente a una unidad generadora de planos de vídeo D 113.

10 La unidad generadora de planos de vídeo B 112 genera un plano de vídeo de vista de base en función de los datos suministrados desde el conmutador 111 y los proporciona, a la salida, a una unidad de combinación 130.

15 La unidad generadora de planos de vídeo D 113 genera un plano de vídeo de vista dependiente en función de los datos suministrados desde el conmutador 111 y los proporciona, a la salida, a la unidad de combinación 130.

20 El conmutador 114 proporciona los paquetes que constituyen el flujo IG de base que se suministra desde el filtro PID 104 o el filtro PID 105 a una memoria intermedia de IG B 115.

25 Un decodificador de IG B 116 decodifica los paquetes que constituyen el flujo IG de base memorizado en la memoria intermedia IG B 115 y proporciona, a la salida, los datos obtenidos mediante la decodificación a una unidad generadora de planos de IG B 117.

30 La unidad generadora de planos de IG B 117 genera un plano de IG de base en función de los datos suministrados desde el decodificador IG B 116 y los proporciona, a la salida, a la unidad de combinación 130.

35 El conmutador 118 proporciona los paquetes que constituyen el flujo IG dependiente que se suministra desde el filtro PID 104 o el filtro PID 105 a una memoria intermedia IG D 119.

40 Un decodificador IG D 120 decodifica los paquetes que constituyen el flujo de IG dependiente memorizado en la memoria intermedia IG D 119 y proporciona los datos obtenidos por intermedio de la decodificación a una unidad generadora de plano de IG D 121.

45 La unidad generadora de plano IG D 121 genera un plano de IG dependiente sobre la base de los datos suministrados desde el decodificador IG D 120 y los proporciona, a la salida, a la unidad de combinación 130.

50 El conmutador 122 proporciona los paquetes que constituyen el flujo de PG base suministrados desde el filtro PID 104 o el filtro PID 105 a una memoria intermedia PG B 123.

55 Un decodificador PG B 124 decodifica los paquetes que constituyen el flujo PG base memorizados en la memoria intermedia B PG 123 y proporciona, a la salida, los datos obtenidos por intermedio de la decodificación a una unidad generadora de planos PG B 125.

60 La unidad generadora de planos PG B 125 genera un plano de PG base en función de los datos suministrados desde el decodificador PG B 124 y los proporciona, a la salida, a la unidad de combinación 130.

65 El conmutador 126 proporciona, a la salida, los paquetes que constituyen el flujo de PG dependiente suministrado desde el filtro PID 104 o el filtro PID 105 a una memoria intermedia PG D 127.

70 Un decodificador PG D 128 decodifica los paquetes que constituyen el flujo de PG dependiente memorizado en la memoria intermedia PG D 127 y proporciona los datos obtenidos mediante la decodificación a una unidad generadora de planos PG D 129.

75 La unidad generadora de planos PG D 129 genera un plano de PG dependiente sobre la base de los datos suministrados desde el decodificador PG D 128 y los proporciona, a la salida, a la unidad de combinación 130.

80 La unidad de combinación 130 combina el plano de vídeo de vista de base suministrado desde la unidad generadora de planos de vídeo B 112, el plano de IG base suministrado desde la unidad generadora de planos IG B 117 y el plano de PG base suministrado desde la unidad generadora de planos PG B 125 apilándolos en un orden predeterminado con lo que se genera un plano de vista de base.

85 Además, la unidad de combinación 130 combina el plano de vídeo de vista dependiente suministrado desde la unidad generadora de planos de vídeo D 113, el plano de IG dependiente suministrado desde la unidad generadora de planos IG D 121 y el plano de PG dependiente suministrado desde la unidad generadora de planos PG D 129 apilándolos en un orden predeterminado, con lo que se genera un plano de vista dependiente.

90 La unidad de combinación 130 proporciona los datos del plano de vista de base y el plano de vista de base. La salida de datos de vídeo desde la unidad de combinación 130 se proporciona al dispositivo de presentación visual 3 y el plano de vista de base y el plano de vista dependiente se visualizan de forma alternada, con lo que se realiza la

presentación visual en 3D.

[Primera realización, a modo de ejemplo, de T-STD (Sistema de flujos de transporte, decodificador objetivo)]

5 A continuación se proporcionará una descripción de la configuración del decodificador y su entorno circundante en la configuración ilustrada en la Figura 21.

La Figura 22 es un diagrama que ilustra una configuración para realizar un proceso sobre un flujo de vídeo.

10 En la Figura 22, las mismas configuraciones que las ilustradas en la Figura 21 se indican por las mismas referencias numéricas. La Figura 22 ilustra el filtro PID 104, la memoria intermedia de vídeo B 106, el conmutador 107, la memoria intermedia de vídeo D 108, el conmutador 109, el decodificador de vídeo 110 y una DPB (Memoria Intermedia de Imágenes Decodificadas) 151. Aunque no se ilustra en la Figura 21, la memoria DPB 151 que memoriza los datos de imagen decodificada se proporciona en la etapa posterior del decodificador de vídeo 110.

15 El filtro PID 104 proporciona los paquetes que constituyen un flujo de vídeo de vista de base incluido en un TS principal a la memoria intermedia de vídeo B 106 y proporciona los paquetes que constituyen un flujo de vídeo de vista dependiente al conmutador 107.

20 A modo de ejemplo, PID = 0 se asigna como un valor fijo de PID a los paquetes que constituyen el flujo de vídeo de vista de base. Además, un valor fijo distinto de 0 se asigna como PID a los paquetes que constituyen el flujo de vídeo de vista dependiente.

25 El filtro PID 104 proporciona los paquetes en los que PID = 0 que se describe en la cabecera para la memoria intermedia de vídeo B 106 y proporciona, a la salida, los paquetes en donde un PID distinto de 0 se describe en la cabecera al conmutador 107.

30 Los paquetes proporcionados a la memoria intermedia de vídeo B 106 se memorizan en un VSB₁ por intermedio de una TB (Memoria Intermedia de Transporte)₁ y una MB (Memoria Intermedia de Multiplexación)₁. Los datos del flujo elemental de vídeo de vista de base se memorizan en VSB₁.

No solamente los paquetes proporcionados desde el filtro PID 104 sino también los paquetes que constituyen un flujo de vídeo de vista dependiente, que se extrae desde un Sub TS en el filtro PID 105, ilustrado en la Figura 21, se suministran al conmutador 107.

35 Cuando se suministran los paquetes que constituyen el flujo de vídeo de vista dependiente desde el filtro PID 104, el conmutador 107 los proporciona, a la salida, a la memoria intermedia de vídeo D 108.

40 Además, cuando se suministran con los paquetes que constituyen el flujo de vídeo de vista dependiente desde el filtro PID 105, el conmutador 107 los proporciona, a la salida, a la memoria intermedia de vídeo D 108.

Los paquetes proporcionados a la memoria intermedia de vídeo D 108 se memorizan en un VSB₂ por intermedio de un TB₂ y un MB₂. Los datos del flujo elemental del vídeo de vista dependiente se memorizan en VSB₂.

45 El conmutador 109 efectúa la lectura secuencial de los paquetes de vídeo de vista de base que se memorizan en VSB₁ de la memoria intermedia de vídeo B 106 y los paquetes de vídeo de vista dependiente memorizados en VSB₂ de la memoria intermedia de vídeo D 108 y los proporciona al decodificador de vídeo 110.

50 A modo de ejemplo, el conmutador 109 proporciona un paquete de vídeo de vista de base de un tiempo determinado e inmediatamente después de que se proporcione, a la salida, un paquete de vídeo dependiente del mismo tiempo. De este modo, el conmutador 109 proporciona secuencialmente el paquete de vídeo de vista de base y el paquete de vídeo de vista dependiente del mismo tiempo al decodificador de vídeo 110.

55 En un paquete que memoriza los datos de una determinada imagen de vídeo de vista de base y un paquete que memoriza los datos de una imagen de vídeo de vista dependiente correspondiente, la misma información temporal con la sincronización de PCR (Referencia de Reloj de Programa) que se asegura, se establece en el momento de la codificación. Aún cuando un flujo de vídeo de vista de base y un flujo de vídeo de vista dependiente estén incluidos en diferentes TSs, la misma información temporal se establece para los paquetes que memorizan los datos de imágenes correspondientes entre sí.

60 La información temporal puede ser una DTS (Marca Temporal de Decodificación) y una PTS (Marca Temporal de Presentación) y se establece para cada paquete PES (Flujo Elemental en Paquetes).

65 Más concretamente la imagen de vídeo de vista de base y la imagen de vídeo de vista dependiente que se posicionan al mismo tiempo cuando las imágenes de los flujos respectivos están dispuestas en orden de codificación/orden de decodificación se consideran como imágenes correspondientes entre sí. La misma DTS se

establece para un paquete de PES que memoriza los datos para una determinada imagen de vídeo de vista de base y un paquete PES que memoriza los datos de la imagen de vídeo de vista dependiente correspondiente a la imagen determinada en la orden de decodificación.

5 Además, la imagen de vídeo de vista de base y la imagen de vídeo de vista dependiente que se posicionan al mismo tiempo cuando las imágenes de los flujos respectivos están dispuestas en orden de presentación visual se consideran como imágenes correspondientes entre sí. La misma PTS se establece para un paquete de PES que memoriza los datos de una determinada imagen de vídeo de vista de base y un paquete PES que memoriza los datos de la imagen de vídeo de vista dependiente correspondiente a la imagen determinada en orden de
10 presentación visual.

En un caso en donde la estructura GOP de un flujo de vídeo de vista de base y la estructura GOP de un flujo de vídeo de vista dependiente son las mismas según se describe a continuación, las imágenes correspondientes entre sí, en el orden de decodificación, se corresponden también entre sí en el orden de presentación visual.
15

En un caso en donde la transferencia de paquetes se realiza en serie, la DTS_1 de un paquete objeto de lectura desde VS_{B1} de la memoria intermedia de vídeo B 106 en una determinada temporización y la DTS_2 de un paquete objeto de lectura desde VS_{B2} o la memoria intermedia de vídeo D 108 en la temporización inmediatamente posterior representan el mismo tiempo, según se ilustra en la Figura 22.
20

El conmutador 109 proporciona, a la salida, los paquetes de vídeo de vista de base leídos de VS_{B1} de la memoria intermedia de vídeo B 106 o los paquetes de vídeo de vista dependiente leídos desde VS_{B2} de la memoria intermedia de vídeo D 108 al decodificador de vídeo 110.

25 El decodificador de vídeo 110 decodifica secuencialmente los paquetes suministrados desde el conmutador 109 y hace que DPB 151 memorice los datos de una imagen de vídeo de vista de base o los datos de una imagen de vídeo de vista dependiente que se obtienen por intermedio de la decodificación.

Los datos de la imagen decodificada que se memorizan en DPB 151 son objeto de lectura por el conmutador 111 en una temporización predeterminada. Además, los datos de la imagen decodificada memorizada en DPB 151 se utilizan para la predicción de otra imagen por el decodificador de vídeo 110.
30

En un caso en donde la transferencia de datos se realice en serie, la PTS de los datos de una imagen de vídeo de vista de base objeto de lectura en una determinada temporización y el PTS de los datos de una imagen de vídeo de vista dependiente objeto de lectura en la temporización inmediatamente posterior representan el mismo tiempo.
35

Un flujo de vídeo de vista de base y un flujo de vídeo de vista dependiente pueden multiplexarse en un TS único, según se describe haciendo referencia a la Figura 5 y así sucesivamente, y pueden incluirse en diferentes TSs, según se describe haciendo referencia a la Figura 7.
40

Incluso en un caso en donde un flujo de vídeo de vista de base y un flujo de vídeo de vista dependiente se multiplexen en un TS único o se incluyan en diferentes TSs, el dispositivo de reproducción 1 puede gestionar el caso teniendo el modelo de decodificador en la Figura 22 montado en dicho dispositivo.

45 A modo de ejemplo, en un caso en donde solamente se supone una situación en la que se suministra un TS único, según se ilustra en la Figura 23, el dispositivo de reproducción 1 es incapaz de gestionar un caso en donde un flujo de vídeo de vista de base y un flujo de vídeo de vista dependiente están incluidos en diferentes TSs.

Además, según el modelo de decodificador ilustrado en la Figura 22, incluso en un caso en donde un flujo de vídeo de vista de base y un flujo de vídeo de vista dependiente estén incluidos en diferentes TSs, se pueden suministrar paquetes al decodificador de vídeo 110 en una temporización correcta debido a que tienen el mismo DTS.
50

Un decodificador para vídeo de vista de base y un decodificador para vídeo de vista dependiente pueden proporcionarse en paralelo. En este caso, los paquetes del mismo tiempo se suministran al decodificador para vídeo de vista de base y al decodificador para vídeo de vista dependiente en la misma temporización.
55

[Segunda realización, a modo de ejemplo]

60 La Figura 24 es un diagrama que ilustra otra configuración para realizar un proceso sobre un flujo de vídeo.

La Figura 24 ilustra el conmutador 111, una unidad generadora de planos de vídeo L 161 y una unidad generadora de planos de vídeo R 162, además de la configuración ilustrada en la Figura 22. Además, el filtro PID 105 se ilustra en la etapa anterior del conmutador 107. Se omitirá una descripción redundante cuando sea necesario.

65 La unidad generadora de planos de vídeo L 161 genera un plano de vídeo de vista L, que se proporciona en lugar de la unidad generadora de planos de vídeo B 112 que se ilustra en la Figura 21.

La unidad generadora de planos de vídeo R 162 genera un plano de vídeo de vista R, que se proporciona en lugar de la unidad generadora de planos de vídeo D 113 que se ilustra en la Figura 21.

5 En esta realización, a modo de ejemplo, el conmutador 111 necesita proporcionar, a la salida, datos de vídeo de vista L y datos de vídeo de vista R procediendo a su identificación.

Es decir, el conmutador 111 necesita identificar si los datos obtenidos decodificando un paquete de vídeo de vista de base son datos de vídeo de vista L o de vista R.

10 Además, el conmutador 111 necesita identificar si los datos obtenidos decodificando un paquete de vídeo de vista dependiente son datos de vídeo de vista L o de vista R.

15 Con el fin de identificar la vista L o la vista R, el parámetro `view_type` descrito con referencia a las Figuras 12 y 14 se utiliza a este respecto. A modo de ejemplo, el controlador 51 proporciona `view_type` descrito en un fichero de Playlist al conmutador 111.

20 En un caso en donde el valor de `view_type` es 0, el conmutador 111 proporciona, a la salida, a la unidad generadora de planos de vídeo L 161, los datos obtenidos decodificando el paquete de vídeo de vista de base identificado por `PID = 0` en los datos memorizados en DPB 151. Según se describió con anterioridad, el valor 0 de `view_type` representa que el flujo de vídeo de vista de base es un flujo de vista L.

En este caso, el conmutador 111 proporciona, a la salida, los datos obtenidos decodificando el paquete de vídeo de vista dependiente identificado por un `PID` distinto de 0 a la unidad generadora de planos de vídeo R 162.

25 Por otro lado, en un caso en donde el valor de `view_type` es 1, el conmutador 111 proporciona, a la salida, a la unidad generadora de planos de vídeo R 162, los datos obtenidos decodificando el paquete de vídeo de vista de base identificado por `PID = 0` en los datos memorizados en la DPB 151. El valor 1 de `view_type` representa que el flujo de vídeo de vista de base es un flujo de vista R.

30 En este caso, el conmutador 111 proporciona, a la salida, los datos obtenidos decodificando el paquete de vídeo de vista dependiente identificado por un `PID` distinto de 0 a la unidad generadora de planos de vídeo L 161.

La unidad generadora de planos de vídeo L 161 genera un plano de vídeo de vista L sobre la base de los datos suministrados desde el conmutador 111 y los proporciona a la unidad de combinación 130.

35 La unidad generadora de planos de vídeo R 162 genera un plano de vídeo de vista R sobre la base de los datos suministrados desde el conmutador 111 y los proporciona, a la salida, a la unidad de combinación 130.

40 En los flujos elementales de vídeo de vista de base y de vídeo de vista dependiente codificados por H.264 AVC/MVC, no existe ninguna información (campo) que indique si el flujo es de vista L o de vista R.

45 Por lo tanto, estableciendo `view_type` para un fichero de Playlist, el dispositivo de registro puede hacer que el dispositivo de reproducción 1 identifique si cada uno de flujo de vídeo de vista de base y un flujo de vídeo de vista dependiente es un flujo de vista L o de vista R.

El dispositivo de reproducción 1 puede identificar si cada uno del flujo de vídeo de vista de base y del flujo de vídeo de vista dependiente es un flujo de vídeo de vista L o de vista R y puede conmutar un destino de salida en conformidad con el resultado de la identificación.

50 En un caso en donde la vista L y la vista R se proporcionan para planos de IG y de PG, los flujos de vídeo de vista L y de vista R pueden distinguirse entre sí, de modo que el dispositivo de reproducción 1 puede combinar fácilmente planos de vista L o planos de vista R.

55 Según se describió con anterioridad, en el caso de proporcionar una señal de vídeo por intermedio de un cable HDMI, se requiere que la señal de vídeo se proporcione con una señal de vista L y una señal de vista R que se distinguen entre sí. El dispositivo de reproducción 1 puede responder a este requisito.

60 Los datos obtenidos mediante la decodificación de un paquete de vídeo de vista de base memorizado en la DPB 151 y los datos obtenidos decodificando un paquete de vídeo de vista dependiente pueden identificarse sobre la base de `view_id` en lugar del `PID`.

65 En el momento de la codificación en H.264 AVC/MVC, `view_id` se establece para las Unidades de Acceso que constituyen un flujo de un resultado de codificación. Con `view_id`, la componente de vista correspondiente a cada Unidad de Acceso puede identificarse de este modo.

La Figura 25 es un diagrama que ilustra, a modo de ejemplo, Unidades de Acceso.

La Unidad de Acceso nº 1 en la Figura 25, es una unidad que incluye los datos de vídeo de vista de base. La Unidad de Acceso nº 2 es una unidad que incluye los datos de vídeo de vista dependiente. Una Unidad de Acceso es una unidad que incluye los datos de una sola imagen, a modo de ejemplo, de modo que se pueda realizar el acceso en unidades de imágenes.

5 Con la codificación en H.264 AVC/MVC, los datos de cada imagen del vídeo de vista de base y del vídeo de vista dependiente se memorizan en dichas Unidades de Acceso. En el momento de la codificación en H.264 AVC/MVC, se añade una cabecera de MVC a cada componente de vista, según se ilustra en la Unidad de Acceso nº 2. La cabecera de MVC incluye view_id.

10 En el caso, a modo de ejemplo, ilustrado en la Figura 25, como para la Unidad de Acceso nº 2, puede identificarse a partir de view_id que la componente de vista memorizada en la Unidad de Acceso es de vídeo de vista dependiente.

15 Por el contrario, según se ilustra en la Figura 25, ningún acera MVC se añade al vídeo de vista de base, que es un componente de vista memorizada en la Unidad de Acceso nº 1.

Según se describió con anterioridad, un flujo de vídeo de vista de base son datos que se utilizan también para la reproducción en 2D. De este modo, con el fin de garantizar su compatibilidad, ninguna cabecera MVC se añade al vídeo de vista de base en el momento de la codificación. Como alternativa, se suprime una cabecera de MVC una vez añadida. La codificación por el dispositivo de registro se describirá a continuación.

20 El dispositivo de reproducción 1 se define (establece) para reconocer que el identificador view_id de la componente de vista sin una cabecera MVC es 0 y para reconocer la componente de vista como vídeo de vista de base. Un valor distinto de 0 se establece como view_id para el vídeo de vista dependiente en el momento de la codificación.

25 En consecuencia, el dispositivo de reproducción 1 puede identificar el vídeo de vista de base en función del view_id reconocido como 0 y puede identificar el vídeo de vista dependiente sobre la base de view_id distinto de 0 que se establece realmente.

30 En el conmutador 111 en la Figura 24, la identificación de los datos obtenidos mediante la decodificación de un paquete de vídeo de vista de base y los datos obtenidos decodificando un paquete de vídeo de vista dependiente pueden realizarse sobre la base de dicho view_id.

35 [Tercera realización, a modo de ejemplo]

La Figura 26 es un diagrama que ilustra otra configuración para realizar un proceso en un flujo de vídeo.

40 En la realización, a modo de ejemplo, ilustrada en la Figura 26, la unidad generadora de planos de vídeo B 112 está provista en lugar de la unidad generadora de planos de vídeo L 161 que se ilustra en la Figura 24 y la unidad generadora de planos de vídeo D 113 se proporciona en lugar de la unidad generadora de planos de vídeo R 162. Un conmutador 171 está provisto en la etapa posterior de la unidad generadora de planos de vídeo B 112 y la unidad generadora de planos de vídeo D 113. En la configuración ilustrada en la Figura 26, además, un destino de salida de datos se conmuta sobre la base del valor de view_type.

45 El conmutador 111 proporciona, a la salida, a la unidad generadora de planos de vídeo B 112, datos obtenidos decodificando un paquete de vídeo de vista de base en los datos memorizados en la DPB 151. Además, el conmutador 111 proporciona, a la salida, datos obtenidos decodificando un paquete de vídeo de vista dependiente a la unidad generadora de planos de vídeo D 113.

50 Los datos obtenidos decodificando un paquete de vídeo de vista de base y los datos obtenidos decodificando un paquete de vídeo de vista dependiente se identifican sobre la base del PID o de view_id, según se describió con anterioridad.

55 La unidad generadora de planos de vídeo B 112 genera un plano de vídeo de vista de base en función de los datos suministrados desde el conmutador 111 y los proporciona a la salida.

La unidad generadora de planos de vídeo D 113 genera un plano de vídeo de vista dependiente sobre la base de los datos suministrados desde el conmutador 111 y los proporciona a la salida.

60 View_type descrito en el fichero de Playlist se suministra desde el controlador 51 al conmutador 171.

65 En un caso en donde el valor de view_type es 0, el conmutador 171 proporciona, a la salida, el plano de vídeo de vista de base suministrado desde la unidad generadora de planos de vídeo B 112 a la unidad de combinación 130 como un plano de vídeo de vista L. El valor 0 de view_type representa que el flujo de vídeo de vista de base es un flujo de vista L.

Además, en este caso, el conmutador 171 proporciona el plano de vídeo de vista dependiente suministrado desde la unidad generadora de planos de vídeo D 113 a la unidad de combinación 130 como un plano de vídeo de vista R.

5 Por otro lado, en un caso en donde el valor de view_type es 1, el conmutador 171 proporciona el plano de vídeo de vista dependiente suministrado desde la unidad generadora de planos de vídeo D 113 a la unidad de combinación 130 como un plano de vídeo de vista L. El valor 1 de view_type representa que el flujo de vídeo de vista de base es un flujo de vista R.

10 Además, en este caso, el conmutador 171 proporciona, a la salida, el plano de vídeo de vista de base suministrado desde la unidad generadora de planos de vídeo B 112 a la unidad de combinación 130 como un plano de vídeo de vista R.

15 Con la configuración ilustrada en la Figura 26, el dispositivo de reproducción 1 puede identificar la vista L o la vista R y puede conmutar un destino de salida en función del resultado de la identificación.

[Primera configuración, a modo de ejemplo, del modelo de combinación de planos]

20 La Figura 27 es un diagrama que ilustra una configuración de la unidad de combinación 130 y su etapa anterior en la configuración ilustrada en la Figura 21.

En la Figura 27, además, las mismas configuraciones que las ilustradas en la Figura 21 se indican por las mismas referencias numéricas.

25 Los paquetes que constituyen un flujo de IG incluido en un TS principal o un TS secundario son objeto de entrada a un conmutador 181. Los paquetes que constituyen el flujo IG se aplican a la entrada del conmutador 181 incluyendo un paquete de vista de base y un paquete de vista dependiente.

30 Los paquetes que constituyen un flujo PG incluidos una TS principal o TS secundaria se aplican a la entrada de un conmutador 182. Los paquetes que constituyen el flujo PG a la entrada de conmutador 182 incluyen un paquete de vista de base y un paquete de vista dependiente.

Según se describe con referencia a la Figura 5 y así sucesivamente, un flujo de vista de base y un flujo de vista dependiente para realizar la presentación visual en 3D se proporcionan también para IG y PG.

35 El IG de vista de base se visualiza combinando con el vídeo de vista de base y el IG de vista dependiente se visualiza combinándose con el vídeo de vista dependiente, con lo que el usuario puede ver un botón y un icono en un modo en 3D, así como vídeo.

40 Además, el PG de vista de base se visualiza combinándose con el vídeo de vista de base y el PG de vista dependiente se visualiza combinándose con el vídeo de vista dependiente, con lo que el usuario puede ver el texto de subtítulos o similar en un modo en 3D así como vídeo.

45 El conmutador 181 proporciona, a la salida, los paquetes que constituyen un flujo IG de base al decodificador IG B 116 y proporciona los paquetes que constituyen un flujo IG dependiente al decodificador IG D 120. El conmutador 181 tienen las funciones del conmutador 114 y el conmutador 118 en la Figura 21. En la Figura 27, se omite la ilustración de las memorias intermedias individuales.

50 El decodificador IG B 116 decodifica los paquetes que constituyen el flujo de IG de base suministrado desde el conmutador 181 y proporciona, a la salida, los datos obtenidos por intermedio de la decodificación a la unidad generadora de planos IG B 117.

La unidad generadora de planos IG B 117 genera un plano de IG de base en función de los datos suministrados desde el decodificador IG B 116 y los proporciona, a la salida, a la unidad de combinación 130.

55 El decodificador IG D 120 decodifica los paquetes que constituyen el flujo de IG dependiente que se suministran desde el conmutador 181 y proporciona los datos obtenidos a través de la decodificación a la unidad generadora de planos IG D 121. El flujo de IG de base y el flujo de IG dependiente pueden decodificarse por un solo decodificador.

60 La unidad generadora de planos de IG D 121 genera un plano de IG dependiente sobre la base de los datos suministrados desde el decodificador IG D 120 y los proporciona, a la salida, a la unidad de combinación 130.

El conmutador 182 proporciona, a la salida, los paquetes que constituyen un flujo de PG de base al decodificador PG B 124 y proporciona los paquetes que constituyen un flujo de PG dependiente al decodificador PG D 128. El conmutador 182 tiene las funciones del conmutador 122 y el conmutador 126 que se ilustran en la Figura 21.

65 El decodificador PG B 124 decodifica los paquetes que constituyen el flujo de PG de base suministrados desde el

conmutador 182 y proporciona, a la salida, los datos obtenidos mediante la decodificación a la unidad generadora de planos PG B 125.

5 La unidad generadora de planos PG B 125 genera un plano de PG base en función de los datos suministrados desde el decodificador PG B 124 y los proporciona, a la salida, a la unidad de combinación 130.

10 El decodificador PG D 128 decodifica los paquetes que constituyen el flujo de PG dependiente suministrados desde el conmutador 182 y proporciona los datos obtenidos, mediante la decodificación, a la unidad generadora de planos de PG D 129. El flujo PG de base y el flujo de PG dependiente pueden decodificarse por un solo decodificador.

10 La unidad generadora de planos PG D 129 genera un plano de PG dependiente sobre la base de los datos suministrados desde el decodificador PG D 128 y los proporciona, a la salida, la unidad de combinación 130.

15 El decodificador de vídeo 110 decodifica secuencialmente los paquetes suministrados desde el conmutador 109 (Figura 22 y sucesivas) y proporciona los datos de vídeo de vista de base o los datos de vídeo de vista dependiente, obtenidos mediante la decodificación, al conmutador 111.

20 El conmutador 111 proporciona los datos obtenidos decodificando los paquetes de vídeo de vista de base a la unidad generadora de planos de vídeo B 112 y proporciona los datos obtenidos, mediante decodificación de los paquetes de vídeo de vista dependiente, a la unidad generadora de planos de vídeo D 113,

20 La unidad generadora de planos de vídeo B 112 genera un plano de vídeo de vista de base en función de los datos suministrados desde el conmutador 111 y los proporciona a la salida.

25 La unidad generadora de planos de vídeo D 113 genera un plano de vídeo de vista dependiente sobre la base de los datos suministrados desde el conmutador 111 y los proporciona a la salida.

La unidad de combinación 130 incluye unidades de adición 191 a 194 y un conmutador 195.

30 La unidad de adición 191 superpone el plano de PG dependiente suministrado desde la unidad generadora de planos PG D 129 en el plano de vídeo vista dependiente que se suministra desde la unidad generadora de planos de vídeo D 113 con el fin de combinar los planos y proporciona, a la salida, el resultado de la combinación a la unidad de adición 193. Un proceso de conversión de información de color (CLUT (Color Look Up Table process) se realiza en el plano de PG dependiente que se suministra desde la unidad generadora de planos PG D 129 a la unidad de adición 191.

40 La unidad de adición 192 superpone el plano de PG de base suministrado desde la unidad generadora de planos PG B 125 en el plano de vídeo de vista de base suministrado desde la unidad generadora de planos de vídeo B 112 con el fin de combinar los planos y proporciona, a la salida, el resultado de la combinación a la unidad de adición 194. Un proceso de conversión de información de color y un proceso de corrección utilizando un valor de compensación se realizan en el plano de PG base suministrado desde la unidad generadora de planos PG B 125 a la unidad de adición 192.

45 La unidad de adición 193 superpone el plano de IG dependiente, que se suministra desde la unidad generadora de planos IG D 121 en el resultado de combinación obtenido en la unidad de adición 191 con el fin de su combinación y proporciona el resultado de la combinación como un plano de vista dependiente. Un proceso de conversión de información de color se realiza en el plano de IG dependiente suministrado desde la unidad generadora de planos IG D 121 a la unidad de adición 193.

50 La unidad de adición 194 superpone el plano de IG base, que se suministra desde la unidad generadora de planos IG B 117, en el resultado de combinación obtenido en la unidad de adición 192 con el fin de su combinación y proporciona el resultado de la combinación como un plano de vista de base. Un proceso de conversión de información de color y un proceso de corrección utilizando un valor de compensación se realizan en el plano de IG base que se suministra desde la unidad generadora de planos IG D 121 a la unidad de adición 194.

55 Una imagen visualizada sobre la base del plano de la vista de base y el plano de vista dependiente, que se generan de esta manera, es una imagen en la que se ven un botón y un icono en una superficie frontal, un texto de subtítulos se ve por debajo (en la dirección de fondo) y se ve vídeo en dichas posiciones.

60 En un caso en donde el valor de view_type es 0, el conmutador 195 proporciona el plano de vista de base como un plano de vista L y proporciona el plano de vista dependiente como un plano de vista R. view_type se suministra desde el controlador 51 al conmutador 195.

65 Además, en un caso en donde el valor de view_type es 1, el conmutador 195 proporciona el plano de vista de base como un plano de vista R y proporciona el plano de vista dependiente como un plano de vista L. La determinación del plano entre los planos suministrados que el plano de vista de base o el plano de vista dependiente se identifica

sobre la base de PID y de view_id.

De este modo, en el dispositivo de reproducción 1, se realiza la combinación de planos de vista de base, planos de vista dependiente y planos de vídeo, IG y PG.

5 En una etapa en donde ha finalizado la combinación de todos los planos de vídeo, IG y PG, si un resultado de la combinación de los planos de vista de base es la vista L o la vista R se determina sobre la base de view_type y un plano de vista R y un plano de vista L son proporcionados a la salida.

10 Además, en una etapa en donde haya finalizado la combinación de todos los planos de vídeo, IG y PG, si un resultado de la combinación de los planos de vista dependiente es vista L o vista R se determina sobre la base de view_type y un plano de vista L y un plano de vista R son proporcionados a la salida.

[Segunda realización, a modo de ejemplo]

15 La Figura 28 es un diagrama que ilustra una configuración de la unidad de combinación 130 y su etapa anterior.

En la configuración ilustrada en la Figura 28, las mismas configuraciones que se ilustran en la Figura 27 se indican por las mismas referencias numéricas. En la Figura 28, la configuración de la unidad de combinación 130 es diferente de la configuración en la Figura 27. Además, la operación del conmutador 111 es diferente de la operación del conmutador 111 en la Figura 27. La unidad generadora de planos de vídeo L 161 se proporciona en lugar de la unidad generadora de planos de vídeo B 112 y la unidad generadora de planos de vídeo R 162 se proporciona en lugar de la unidad generadora de planos de vídeo D 113. Se omitirá una descripción redundante.

25 El mismo valor de view_type se suministra desde el controlador 51 al conmutador 111 y un conmutador 201 y un conmutador 202 de la unidad de combinación 130.

El conmutador 111 conmuta, como en el caso del conmutador 111 en la Figura 24, los destinos de salida de los datos obtenidos decodificando un paquete de vídeo de vista de base y los datos obtenidos decodificando un paquete de vídeo de vista dependiente sobre la base de view_type.

35 A modo de ejemplo, en un caso en donde el valor de view_type es 0, el conmutador 111 proporciona los datos obtenidos decodificando un paquete de vídeo de vista de base para la unidad generadora de planos de vídeo L 161. En este caso, el conmutador 111 proporciona los datos obtenidos decodificando un paquete de vídeo de vista dependiente a la unidad generadora de planos de vídeo R 162.

Por otro lado, en un caso en donde el valor de view_type es 1, el conmutador 111 proporciona los datos obtenidos decodificando un paquete de vídeo de vista de base para la unidad generadora de planos de vídeo R 162. En este caso, el conmutador 111 proporciona los datos obtenidos decodificando un paquete de vídeo de vista dependiente a la unidad generadora de planos de vídeo L 161.

La unidad generadora de planos de vídeo L 161 genera un plano de vídeo de vista L sobre la base de los datos suministrados desde el conmutador 111 y los proporciona a la unidad de combinación 130.

45 La unidad generadora de planos de vídeo R 162 genera un plano de vídeo de vista R sobre la base de los datos suministrados desde el conmutador 111 y los proporciona a la unidad de combinación 130.

La unidad de combinación 130 incluye el conmutador 201, el conmutador 202 y las unidades de adición 203 a 206.

50 El conmutador 201 conmuta los destinos de salida del plano de IG base suministrados desde la unidad generadora de planos de IG B 117 y el plano de IG dependiente suministrados desde la unidad generadora de planos IG D 121 sobre la base de view_type.

55 A modo de ejemplo, en un caso en donde el valor de view_type es 0, el conmutador 201 proporciona el plano de IG base suministrado desde la unidad generadora de planos IG B 117 a la unidad de adición 206 como un plano de vista L. En este caso, el conmutador 201 proporciona el plano de IG dependiente suministrado desde la unidad generadora de planos IG D 121 a la unidad de adición 205 como un plano de vista R.

60 Por otro lado, en un caso en donde el valor de view_type es 1, el conmutador 201 proporciona el plano de IG dependiente suministrado desde la unidad generadora de planos IG D 121 a la unidad de adición 206 como un plano de vista L. En este caso, el conmutador 201 proporciona el plano de IG base suministrado desde la unidad generadora de planos IG B 117 a la unidad de adición 205 como un plano de vista R.

65 El conmutador 202 conmuta los destinos de salida del plano de PG base que se suministran desde la unidad generadora de planos PG B 125 y el plano de PG dependiente que se suministra desde la unidad generadora de planos PG D 129 sobre la base de view_type.

A modo de ejemplo, en un caso en donde el valor de `view_type` es 0, el conmutador 202 proporciona el plano de PG base que se suministra desde la unidad generadora de planos de PG B 125 a la unidad de adición 204 como un plano de vista L. En este caso, el conmutador 202 proporciona el plano de PG dependiente suministrado desde la unidad generadora de planos PG D 129 a la unidad de adición 203 como el plano de vista R.

5 Por otro lado, en un caso en donde el valor de `view_type` es 1, el conmutador 202 proporciona el plano de PG dependiente suministrado desde la unidad generadora de planos PG D 129 a la unidad de adición 204 como un plano de vista L. En este caso, el conmutador 202 proporciona el plano de PG base suministrado desde la unidad generadora de planos PG B 125 a la unidad de adición 203 como un plano de vista R.

10 La unidad de adición 203 superpone el plano de PG de vista R suministrado desde el conmutador 202 sobre el plano de vídeo de vista R suministrado desde la unidad generadora de planos de vídeo R 162 con el fin de combinar los planos y proporciona, a la salida, el resultado de la combinación a la unidad de adición 205.

15 La unidad de adición 204 superpone el plano de PG de vista L suministrado desde el conmutador 202 sobre el plano de vídeo de vista L suministrado desde la unidad generadora de planos de vídeo L 161 con el fin de combinar los planos y proporciona, a la salida, el resultado de la combinación a la unidad de adición 206.

20 La unidad de adición 205 superpone el plano de IG de vista R suministrado desde el conmutador 201 sobre el plano como el resultado de la combinación obtenido en la unidad de adición 203 con el fin de su combinación y proporciona el resultado de la combinación como un plano de vista R.

25 La unidad de adición 206 superpone el plano de IG de vista L suministrado desde el conmutador 201 sobre el plano como el resultado de la combinación obtenido en la unidad de adición 204 con el fin de su combinación y proporciona el resultado de la combinación como un plano de vista L.

De este modo, en el dispositivo de reproducción 1, se determina si cada uno del plano de vista de base y el plano de vista dependiente de vídeo, IG y PG es un plano de vista L o de vista R antes de combinarse con otro plano.

30 Después de que se haya realizado dicha determinación, se realiza la combinación de los planos de vídeo, IG y PG para combinar planos de vista L entre sí y planos de vista R entre sí.

[Configuración del dispositivo de registro a modo de ejemplo]

35 La Figura 29 es un diagrama de bloques que ilustra una configuración, a modo de ejemplo, de una unidad de procesamiento de fabricación de programas informáticos 301.

40 Un codificador de vídeo 311 tiene la misma configuración que la del identificador MVC 11 en la Figura 3. El codificador de vídeo 311 codifica una pluralidad de elementos de datos de vídeo en conformidad con H.264 AVC/MVC, por lo que se genera un flujo de vídeo de vista de base y un flujo de vídeo de vista dependiente y los proporciona, a la salida, a una memoria intermedia 312.

45 A modo de ejemplo, el codificador de vídeo 311 establece DTS y PTS con el mismo PCR como una referencia en el momento de la codificación. Es decir, el codificador de vídeo 311 establece el mismo DTS para un paquete PES que memoriza los datos de una determina imagen de vídeo de vista de base y un paquete PES que memoriza los datos de una imagen de vídeo de vista dependiente correspondiente a la imagen en el orden de decodificación.

50 Además, el codificador de vídeo 311 establece el mismo PTS para un paquete PES que memoriza los datos de una determina imagen de vídeo de vista de base y un paquete PES que memoriza los datos de una imagen de vídeo de vista dependiente correspondiente a la imagen en orden de presentación visual.

55 Según se describe a continuación, el codificador de vídeo 311 establece la misma información que la información adicional, que es información auxiliar sobre la decodificación, para una imagen de vídeo de vista de base y una imagen de vista de base correspondientes entre sí, en el orden de decodificación.

60 Además, según se describió con anterioridad, el codificador de vídeo 311 establece el mismo valor, que el valor de establece el mismo valor, que el valor de POC que representa el orden de salida de las imágenes, para una imagen de vídeo de vista de base y una imagen de vídeo de vista de base correspondientes entre sí en el orden de presentación visual.

65 Además, según se describe a continuación, el codificador de vídeo 311 realiza la codificación de modo que la estructura GOP de un flujo de vídeo de vista de base esté en correspondencia con la estructura GOP de una flujo de vídeo de vista dependiente.

Un codificador de audio 313 codifica una entrada de flujo de audio y proporciona, a la salida, los datos obtenidos de esta operación a una memoria intermedia 314. Un flujo de audio que ha de registrarse en un disco junto con un flujo

de vídeo de vista de base y un flujo de vídeo de vista dependiente se aplica a la entrada del codificador de audio 313.

5 Un codificador de datos 315 codifica los diversos tipos anteriormente descritos de datos distintos a los de vídeo y audio, tal como un fichero de lista de reproducción Playlist, y proporciona, a la salida, los datos obtenidos mediante la codificación a una memoria intermedia 316.

10 El codificador de datos 315 establece el parámetro del tipo de vista view_type que representa si un flujo de vídeo de vista de base es un flujo de vista L o un flujo de vista R, al fichero de Playlist en conformidad con la codificación realizada por el codificador de vídeo 311. La información que muestra si un flujo de vídeo de vista dependiente es un flujo de vista L o un flujo de vista R puede establecerse en lugar del tipo de flujo de vídeo de vista de base.

15 Además, el codificador de datos 315 establece EP_map, que se describirá a continuación, a cada uno de un fichero de Clip Information de un flujo de vídeo de vista de base y un fichero Clip Information de un flujo de vídeo de vista dependiente. La imagen del flujo de vídeo de vista de base y la imagen del flujo de vídeo de vista dependiente, que se establecen para EP_map como una posición de partida de la decodificación, son imágenes que están en mutua correspondencia.

20 Una unidad de multiplexación 317 multiplexa los datos de vídeo y los datos de audio memorizados en las memorias intermedias individuales y los datos distintos a los flujos junto con una señal de sincronización y los proporciona, a la salida, a una unidad de codificación de corrección de errores 318.

25 La unidad de codificación de corrección de errores 318 añade un código para la corrección de errores a los datos multiplexados por la unidad de multiplexación 317.

Una unidad de modulación 319 modula los datos suministrados desde la unidad de codificación de corrección de errores 318 y los proporciona a la salida. La salida de la unidad de modulación 319 sirve como software que ha de registrarse en el disco óptico 2 que puede reproducirse en el dispositivo de reproducción 1.

30 La unidad de procesamiento de fabricación de programas informáticos 301, que tiene dicha configuración, se proporciona en el dispositivo de registro.

35 La Figura 30 es un diagrama que ilustra, a modo de ejemplo, una configuración que incluye la unidad de procesamiento de fabricación de programas informáticos 301.

Parte de la configuración ilustrada en la Figura 30 puede proporcionarse en el dispositivo de registro.

40 Un proceso de dominio operativo se realiza sobre una señal de registro generada por la unidad de procesamiento de fabricación de programas informáticos 301 por intermedio de una unidad de procesamiento pre-maestra 331, de modo que se genere una señal que tenga un formato a registrarse en el disco óptico 2. La señal generada se suministra a una unidad de registro maestra 333.

45 En una unidad de fabricación maestra para registro 332, se prepara un registro maestro hecho de vidrio o material similar, en donde se aplica un material de registro que incluye el denominado "photoresist" o similar. En consecuencia, se fabrica una unidad maestra para registro.

50 En la unidad de registro maestra 333, se modula un haz de láser en conformidad con la señal de registro desde la unidad de procesamiento pre-maestra 331 y se irradia en ella el material de 'photoresist' en la unidad maestra. En consecuencia, el material de photoresist en la unidad maestra queda expuesto en función de la señal de registro. Después de lo que antecede, se desarrolla esta unidad maestra, de modo que las depresiones surjan en la unidad maestra.

55 En una unidad de fabricación maestra metálica 334, se realiza un proceso tal como electro-conformación sobre la unidad maestra, de modo que se obtenga una unidad maestra metálica a la que se transfieren las depresiones en la unidad maestra de vidrio. Además, un estampador metálico se fabrica a partir de esta unidad maestra metálica, que se utiliza como una matriz de moldeo.

60 En una unidad de procesamiento de moldeo 335, un material tal como PMMA (acrílico) o PC (policarbonato) se inyecta en la matriz de moldeo mediante inyección o de modo similar y se realiza la fijación en ella. Como alternativa, se aplica 2P (resina de curado ultravioleta) o material similar sobre el estampador metálico, que es irradiado con una radiación ultravioleta para su curado. En consecuencia, las depresiones en el estampador metálico pueden transferirse en una réplica hecha de resina.

65 En una unidad de procesamiento de formación de película 336, se forma una película reflectante sobre la réplica mediante deposición por vapor o dispersión. Como alternativa, se forma una película reflectante sobre la réplica mediante un revestimiento giratorio.

En una unidad de pos-procesamiento 337, se realizan los procesos necesarios, es decir, un proceso de diámetros interiores y exteriores se realiza en este disco y dos discos se adhieren juntos. Además, se adhiere una etiqueta y se conecta un concentrador y luego, el disco se inserta en un cartucho. De este modo, el disco óptico 2 que tiene datos que pueden reproducirse por el dispositivo de reproducción 1 allí registrado se completa.

5 Segunda forma de realización

[Operación 1 de flujo de vídeo de perfil H.264 AVC/MVC]

10 En la norma de BD-ROM, que es una norma del disco óptico 2 la codificación de vídeo en 3D se realiza utilizando el perfil de H.264 AVC/MVC, según se describió con anterioridad.

Además, en la norma BD-ROM, un flujo de vídeo de vista de base se considera como un flujo de vídeo de vista L y un flujo de vídeo de vista dependiente se considera como un flujo de vídeo de vista R.

15 El vídeo de vista de base se codifica como un flujo de vídeo del perfil H.264 AVC/alto perfil, en donde el disco óptico 2 es un disco compatible con 3D, puede reproducirse incluso en un reproductor anterior o un reproductor compatible con solamente la reproducción en 2D. Es decir, se puede garantizar una compatibilidad descendente.

20 Más concretamente, solamente se puede decodificar (reproducir) un flujo de vídeo de vista de base incluso en un decodificador incompatible con H.264 AVC/MVC. Es decir, el flujo de vídeo de vista de base es un flujo que puede reproducirse de forma fiable incluso en una capa de 2D BD ya existente.

25 Además, el flujo de vídeo de vista de base se utiliza en común en la reproducción en 2D y en la reproducción en 3D, con lo que se puede reducir una carga en el momento de dicha operación. En el lado de origen, como para un flujo AV, se puede obtener un disco compatible con 3D preparando un flujo de vídeo de vista dependiente, además del trabajo que se realiza de forma convencional.

30 La Figura 31 es un diagrama que ilustra una configuración, a modo de ejemplo, de una unidad generadora de TS de vídeo en 3D proporcionada en el dispositivo de registro.

35 La unidad generadora de TS de vídeo en 3D, representada en la Figura 31, incluye un codificador de MVC 401, una unidad de supresión de cabecera de MVC 402 y un multiplexor 403. Los datos de vídeo nº 1 de vista L y los datos de vídeo nº 2 de vista R, que son captados en la manera descrita haciendo referencia a la Figura 2, se aplican a la entrada del codificador MVC 401.

40 De forma similar a la del codificador MVC 11 en la Figura 3, el codificador MVC 401 codifica los datos de vídeo nº 1 de valor L utilizando H.264/AVC y proporciona, a la salida, datos de vídeo AVC obtenidos mediante la codificación como un flujo de vídeo de vista de base. Además, el codificador MVC 401 genera un flujo de vídeo de vista dependiente sobre la base de los datos de vídeo nº 1 de la vista L y los datos de vídeo nº 2 de vista R y los proporciona a la salida.

45 La salida de flujo de vídeo de vista de base, procedente del codificador MVC 401, está constituida por Unidades de Acceso que memorizan, cada una de ellas los datos de una imagen de vídeo de vista de base. Además, la salida de flujo de vídeo de vista dependiente desde el codificador MVC 401 está constituida por Unidades de Acceso que memoriza cada una de ellas, los datos de una imagen de vídeo de vista dependiente.

50 Cada una de las Unidades de Acceso que constituyen el flujo de vídeo de vista de base y cada una de las Unidades de Acceso que constituyen el flujo de vídeo de vista dependiente incluyen una cabecera de MVC que describe view_id para identificar la componente de vista allí memorizada.

55 Un valor fijo de 1 o más se utiliza como el valor de view_id descrito en la cabecera MVC de vídeo de vista dependiente. Lo que antecede es lo mismo para las configuraciones, a modo de ejemplo, ilustradas en las Figuras 32 y 33.

60 Es decir, a diferencia del codificador MVC 11 en la Figura 3, el codificador MVC 401 es un codificador que genera flujos individuales de vídeo de vista de base y de vídeo de vista dependiente en la forma de adición de cabeceras MVC y que proporciona, a la salida, los flujos. En el codificador MVC 11 en la Figura 3, las cabeceras MVC se añaden solamente en el vídeo de vista dependiente que se codifica utilizando H.264 AVC/MVC.

La salida de flujo de vídeo de vista de base, procedente del codificador MVC 401, se suministra a la unidad de supresión de cabecera de MVC 402 y el flujo de vídeo de vista dependiente se suministra al multiplexor 403.

65 La unidad de supresión de cabecera MVC 402 suprime las cabeceras MVC incluidas en las Unidades de Acceso individuales que constituyen el flujo de vídeo de vista de base. La unidad de supresión de cabecera MVC 402 proporciona, al multiplexor 403, el flujo de vídeo de vista de base constituido por Unidades de Acceso de las cuales

se han suprimido las cabeceras MVC.

El multiplexor 403 genera un TS que incluye el flujo de vídeo de vista de base suministrado desde la unidad de supresión de cabecera MVC 402 y el flujo de vídeo de vista dependiente suministrado desde el codificador MVC 401 y los proporciona a la salida. En la realización, a modo de ejemplo, ilustrada en la Figura 31, un TS que incluye el flujo de vídeo de vista de base y un TS que incluye el flujo de vídeo de vista dependiente son proporcionados individualmente, pero estos flujos pueden ser objeto de salida mediante la multiplexación en el mismo TS según se describió con anterioridad.

De este modo, dependiendo de la manera de montaje, un codificador MVC que recibe vídeo de vista L y vídeo de vista R y que proporciona flujos individuales de vídeo de vista de base y vídeo de vista dependiente, con cabeceras MVC, puede proporcionarse.

Como alternativa, la configuración completa ilustrada en la Figura 31 puede incluirse en un codificador MVC, según se ilustra en la Figura 3. Lo que antecede es lo mismo para las configuraciones ilustradas en las Figuras 32 y 33.

La Figura 32 es un diagrama que ilustra otra configuración, a modo de ejemplo, de la unidad generadora de TS de vídeo en 3D proporcionada en el dispositivo de registro.

La unidad generadora de TS de vídeo en 3D, ilustrada en la Figura 32, incluye una unidad de procesamiento de mezcla 411, un codificador MVC 412, una unidad de separación 413, una unidad de supresión de cabecera MVC 414 y un multiplexor 415. Los datos de vídeo nº 1 de vista L y los datos de vídeo nº 2 de vista R son objeto de entrada a la unidad de procesamiento de mezcla 411.

La unidad de procesamiento de mezcla 411 dispone las imágenes individuales de vista L y las imágenes individuales de vista R en orden de codificación. Las imágenes individuales de vídeo de vista dependiente se codifican haciendo referencia a las imágenes correspondientes de vídeo de vista de base. De este modo, en el resultado de la disposición en orden de codificación, las imágenes de vista L y las imágenes de vista R están dispuestas de forma alternada.

La unidad de procesamiento de mezcla 411 proporciona las imágenes de la vista L y las imágenes de vista R dispuestas en orden de codificación al codificador MVC 412.

El codificador MVC 412 codifica las imágenes individuales suministradas desde la unidad de procesamiento de mezcla 411 utilizando H.264 AVC/MVC y proporciona, a la salida, el flujo obtenido mediante la codificación a la unidad de separación 413. Un flujo de vídeo de vista de base y un flujo de vídeo de vista dependiente son objeto de multiplexación en la salida de flujos desde el codificador MVC 412.

El flujo de vídeo de vista de base incluido en la salida de flujo desde el codificador MVC 412 está constituido por Unidades de Acceso que memoriza, cada una de ellas, los datos de una imagen de vídeo de vista de base. Además, el flujo de vídeo de vista dependiente, incluido en la salida de flujos desde el codificador MVC 412, está constituido por Unidades de Acceso que memoriza, cada una de ellas, los datos de una imagen de vídeo de vista dependiente.

Cada una de las Unidades de Acceso que constituyen el flujo de vídeo de vista de base y cada una de las Unidades de Acceso que constituyen el flujo de vídeo de vista dependiente incluyen una cabecera MVC que describe view_id para identificar la componente de vista allí memorizada.

La unidad de separación 413 separa el flujo de vídeo de vista de base y el flujo de vídeo de vista dependiente multiplexados en el flujo suministrado desde el codificador MVC 412 entre sí y los proporciona a la salida. La salida de flujo de vídeo de vista de base desde la unidad de separación 413 se suministra a la unidad de supresión de cabecera MVC 414 y el flujo de vídeo de vista dependiente se suministra al multiplexor 415.

La unidad de supresión de cabecera MVC 414 suprime las cabeceras MVC incluidas en las Unidades de Acceso individuales que constituyen el flujo de vídeo de vista de base suministrado desde la unidad de separación 413. La unidad de supresión de cabecera MVC 414 proporciona al multiplexor 415, el flujo de vídeo de vista de base constituido por Unidades de Acceso de las que se han suprimido las cabeceras MVC.

El multiplexor 415 genera un TS que incluye el flujo de vídeo de vista de base suministrado desde la unidad de supresión de cabecera MVC 414 y el flujo de vídeo de vista dependiente suministrado desde la unidad de separación 413 y lo proporciona a la salida.

La Figura 33 es un diagrama que ilustra otra configuración, a modo de ejemplo, de la unidad generadora de TS de vídeo en 3D proporcionada en el dispositivo de registro.

La unidad generadora de TS de vídeo en 3D, representada en la Figura 33, incluye un codificador AVC 421, un codificador MVC 422 y un multiplexor 423. Los datos de vídeo nº 1 de vista L se aplican a la entrada de codificador

AVC 421 y los datos de vídeo nº 2 de vista R se aplican a la entrada del codificador MVC 422.

El codificador AVC 421 codifica los datos de vídeo nº 1 de la vista L utilizando H.264/AVC y proporciona, a la salida, el flujo de vídeo AVC obtenido mediante la codificación, que sirve como un flujo de vídeo de vista de base, al codificador MVC 422 y al multiplexor 423. Las Unidades de Acceso individuales que constituyen la salida de flujo de vídeo de vista de base procedente del codificador AVC 421 no incluyen las cabeceras MVC.

El codificador MVC 422 decodifica el flujo de vídeo de vista de base (flujo de vídeo AVC) suministrado desde el codificador AVC 421 para generar los datos de vídeo nº 1 de vista L.

Además, el codificador MVC 422 genera un flujo de vídeo de vista dependiente sobre la base de los datos de vídeo nº 1 de la vista L obtenidos mediante la decodificación y los datos de vídeo nº 2 de vista R que se le introducen externamente y los proporciona, a la salida, al multiplexor 423. Las Unidades de Acceso individuales que constituyen la salida de flujo de vídeo de vista dependiente desde el codificador MVC 422 incluyen cabeceras MVC.

El multiplexor 423 genera un TS que incluye el flujo de vídeo de vista de base suministrado desde el codificador AVC 421 y el flujo de vídeo de vista dependiente suministrado desde el codificador MVC 422 y los proporciona a la salida.

El codificador AVC 421 en la Figura 33, tiene la función del codificador H.264/AVC 21 en la Figura 3 y el codificador MVC 422 tiene las funciones del codificador H.264/AVC 22 y el codificador de vídeo de vista dependiente 24 en la Figura 3. Además, el multiplexor 423 tiene la función del multiplexor 25 ilustrado en la Figura 3.

La unidad generadora de TS de vídeo en 3D, que tiene dicha configuración, se proporciona en el dispositivo de registro, con lo que se puede prohibirse la codificación de la cabecera MVC en cada Unidad de Acceso que memoriza los datos de vídeo de vista de base. Además, una cabecera MVC en la que se establece view_id con un valor de 1 o más, puede incluirse en cada Unidad de Acceso que memoriza los datos de vídeo de vista dependiente.

La Figura 34 es un diagrama que ilustra una configuración del lado del dispositivo de reproducción 1 para decodificar Unidades de Acceso.

La Figura 34 ilustra el conmutador 109 y el decodificador de vídeo 110 descritos con referencia a la Figura 22 y así sucesivamente. La Unidad de Acceso nº 1 que incluye los datos de vídeo de vista de base y la Unidad de Acceso nº 2 que incluye los datos de vídeo de vista dependiente son objeto de lectura desde una memoria intermedia y se suministran al conmutador 109.

La codificación se realiza con referencia al vídeo de vista de base y de este modo, resulta necesario decodificar el vídeo de vista de base correspondiente con el fin de decodificar correctamente el vídeo de vista dependiente.

En la norma H.264/MVC, un lado de decodificador calcula el orden de decodificación de Unidades de Acceso individuales que utilizan view_id incluido en las cabeceras MVC. Además, en el vídeo de vista de base se define que un valor mínimo se establece constantemente como el valor de view_id en el instante de la codificación. El decodificador inicia la decodificación desde la Unidad de Acceso que incluye la cabecera MVC en donde se establece el view_id mínimo, con lo que es capaz de decodificar el vídeo de vista de base y el vídeo de vista dependiente en orden correcto.

A tal respecto, la codificación de una cabecera MVC está prohibida en una Unidad de Acceso que memoriza el vídeo de vista de base suministrado al decodificador de vídeo 110 del dispositivo de reproducción 1.

Por lo tanto, en el dispositivo de reproducción 1, la componente de vista memorizada en una Unidad de Acceso sin la cabecera MVC se define para reconocerse que su view_id es 0.

En consecuencia, el dispositivo de reproducción 1 puede identificar el vídeo de vista de base en función de view_id que se reconoce como 0 y puede identificar el vídeo de vista dependiente sobre la base del valor de view_id, realmente establecido, distinto de 0.

El conmutador 109 en la Figura 34 proporciona primero la Unidad de Acceso nº 1, en donde se reconoce que el valor mínimo 0 se establece como view_id, al decodificador de vídeo 110 y hace que se realice la decodificación correspondiente.

Además, después de que haya finalizado la decodificación de la Unidad de Acceso nº 1, el conmutador 109 proporciona, a la salida, la Unidad de Acceso nº 2 en donde Y como un valor fijo mayor que 0 se establece como view_id para el decodificador de vídeo 110 y hace que se realice la decodificación. La imagen de vídeo de vista dependiente memorizada en la Unidad de Acceso nº 2 es una imagen correspondiente a la imagen de vídeo de vista de base memorizada en la Unidad de Acceso nº 1.

De este modo, está prohibida la codificación de una cabecera MVC en una Unidad de Acceso que memoriza el

vídeo de vista de base, con lo que el flujo de vídeo de vista de base suministrado en el disco óptico 2 puede considerarse como un flujo que puede reproducirse incluso en un reproductor convencional.

5 Incluso en un caso en donde una condición de que un flujo puede reproducirse también en un reproductor convencional se establece como una condición de un flujo de vídeo de vista de base de la norma BD-ROM 3D expandida desde la norma BD-ROM, con lo que puede satisfacerse la condición.

10 A modo de ejemplo, según se ilustra en la Figura 35, en un caso en donde cabeceras MVC se añaden a un vídeo de vista de base y al vídeo de vista dependiente, respectivamente, y en donde se realiza primero la decodificación de vídeo de vista de base, el vídeo de vista de base no se puede reproducir en un reproductor convencional. La cabecera MVC es de datos indefinidos para un decodificador H.264/AVC montado en un reproductor convencional. En un caso en donde se introducen dichos datos indefinidos, algunos decodificadores no pueden ignorar los datos y puede fallar el procesamiento.

15 Conviene señalar que, en la Figura 35, el valor de view_id de vídeo de vista de base es X y el view_id del vídeo de vista dependiente es Y, que es mayor que X.

20 Además, incluso en un caso en donde esté prohibida la codificación de cabeceras MVC, el dispositivo de reproducción 1 puede hacerse que realice primero la decodificación de vídeo de vista de base y luego, realice la decodificación de vídeo de vista dependiente correspondiente haciendo la definición de modo que el view_id de vídeo de vista de base se considere como 0. Es decir, la decodificación puede realizarse en el orden correcto.

Operación 2

25 Sobre la estructura del GOP

En la norma H.264 AVC, la estructura del GOP (Grupo de Imágenes) en la norma de vídeo MPEG-2 no está definida.

30 Por lo tanto, en la norma BD-ROM para gestionar un flujo de vídeo de H.264/AVC, la estructura del grupo GOP de un flujo de vídeo de H.264/AVC se define y se realizan varios tipos de funciones que utilizan una estructura del GOP, tal como un acceso aleatorio.

35 En un flujo de vídeo de vista de base y en un flujo de vídeo de vista dependiente, que son flujos de vídeo obtenidos mediante codificación utilizando H.264 AVC/MVC, la definición de una estructura del GOP no existe como en el flujo de vídeo de H.264/AVC.

40 El flujo de vídeo de vista de base es un flujo de vídeo de H.264/AVC. De este modo, la estructura del GOP del flujo de vídeo de vista de base es la misma que la estructura del GOP del flujo de vídeo H.264/AVC definido en la norma BD-ROM.

45 La estructura del GOP del flujo de vídeo de vista dependiente se define también como la misma estructura que la estructura del GOP del flujo de vídeo de vista de base, es decir, la estructura del GOP del flujo de vídeo de H.264/AVC definido en la norma BD-ROM.

La estructura del GOP del flujo de vídeo de H.264/AVC que se define en la norma BD-ROM tiene las características operativas siguientes.

50 1. Características sobre la estructura de flujo

(1) Estructura del GOP abierto/cerrado

La Figura 36 es un diagrama que ilustra una estructura del GOP cerrado.

55 Las imágenes individuales, ilustradas en la Figura 36, son imágenes que constituyen un flujo de vídeo de H.264/AVC. Un grupo GOP cerrado incluye una imagen de IDR (Recuperación Instantánea de Decodificación).

60 La imagen de IDR es una imagen I, que se decodifica primero en el grupo GOP que incluye la imagen IDR. En el instante de la decodificación de la imagen IDR, todos los elementos de información sobre la decodificación, tal como el estado operativo de la memoria intermedia de imágenes de referencia (DPB 151 en la Figura 22) y los números de tramas y POC (Conteo del Orden de Imágenes) gestionados hasta ahora son objeto de reposición.

65 Según se ilustra en la Figura 36, en el GOP actual, que es un grupo GOP cerrado, las imágenes anteriores (pasadas) con respecto a la imagen IDR en el orden de presentación visual entre las imágenes del grupo GOP actual están prohibidas en cuanto a referirse a las imágenes del GOP anterior.

Además, entre las imágenes del GOP actual, las imágenes posteriores (futuras) con respecto a la imagen IDR, en el orden de presentación visual, están prohibidas de referirse a las imágenes del GOP anterior más allá de la imagen IDR. En H.264/AVC, está permitido que una imagen P después de una imagen I, en el orden de presentación visual se refiera a una imagen antes de la imagen I.

5

La Figura 37 es un diagrama que ilustra una estructura del GOP abierto.

Según se ilustra en la Figura 37, en el GOP actual, que es un GOP abierto, las imágenes antes de una imagen I no de IDR (imagen I que no es una imagen IDR), en el orden de presentación visual, entre las imágenes del GOP actual están permitidas para referirse a las imágenes del GOP anterior.

10

Además, entre las imágenes del GOP actual, las imágenes después de la imagen I, no de IDR, en el orden de presentación visual, están prohibidas de referirse a las imágenes del GOP anterior más allá de la imagen I no de IDR.

15

(2) SPS y PPS están fiablemente codificados en la primera Unidad de Acceso de un grupo GOP.

El SPS (Parámetro de Secuencia Establecido) es información de cabecera de una secuencia, que incluye información sobre la codificación de la secuencia completa. Al inicio de la decodificación de una determinada secuencia, un SPS que incluya la información de identificación de la secuencia es necesario. El PPS (Parámetro de Imagen Establecido) es información de cabecera de una imagen, que incluye información sobre la codificación de la imagen completa.

20

(3) Un máximo de 30 PPSs puede codificarse en la primera Unidad de Acceso de un grupo GOP. En un caso en donde una pluralidad de PPSs se codifica en la primera Unidad de Acceso, el identificador (pic_parameter_set_id) de cada PPS no debe ser el mismo.

25

(4) Un máximo de 1 PPS puede codificarse en una Unidad de Acceso distinta a la primera Unidad de Acceso de un grupo GOP.

30

2. Características sobre la estructura de referencia

(1) Las imágenes I, P y B son requeridas para ser imágenes constituidas por solamente segmentos I, P y B, respectivamente.

35

(2) La imagen B inmediatamente antes de una imagen de referencia (imagen I o P) en el orden de presentación visual, se requiere que sea fiablemente codificada inmediatamente después de la imagen de referencia en el orden de codificación.

40

(3) El orden de codificación y el orden de presentación visual de una imagen de referencia (imagen I o P) se requieren que se mantengan (siendo la misma).

(4) La referencia a una imagen B desde una imagen P está prohibida.

45

(5) En un caso en donde una imagen B no de referencia (B1) está antes de una imagen de no referencia (B2) en el orden de codificación, la imagen B1 se requiere que esté antes de la B2 también en el orden de presentación visual.

Una imagen B, no de referencia, es una imagen B que no es objeto de referencia por otra imagen posterior en el orden de la codificación.

50

(6) La imagen B de referencia puede referirse a la imagen de referencia anterior o posterior (imagen I o P) en el orden de presentación visual.

(7) La imagen B no de referencia puede referirse a la imagen de referencia anterior o posterior (imagen I o P) en el orden de presentación visual o la imagen B de referencia.

55

(8) Se requiere que el número máximo de imágenes B secuenciales sea 3.

3. Características sobre el número máximo de tramas y campos en el grupo GOP.

60

El número máximo de tramas y campos en un grupo GOP se especifica en conformidad con la tasa de tramas de vídeo, según se ilustra en la Figura 38.

Según se ilustra en la Figura 38, en un caso en donde se realiza una presentación visual entrelazada con una tasa de tramas de 29.97 tramas por segundo, a modo de ejemplo, el número máximo de campos que pueden visualizarse con imágenes de 1 GOP es 60. Además, en un caso en donde se realiza una presentación visual progresiva con una

65

tasa de tramas de 59.94 tramas por segundo, el número máximo de tramas que puede visualizarse con imágenes de 1 GOP es 60.

5 La estructura del GOP que tiene las características anteriores se define también como la estructura del GOP de un flujo de vídeo de vista dependiente.

Además, la puesta en correspondencia entre la estructura de un determinado GOP de un flujo de vídeo de vista de base y la estructura del GOP correspondiente de un flujo de vídeo de vista dependiente se define como una restricción operativa.

10 La Figura 39 ilustra una estructura del GOP cerrado de un flujo de vídeo de vista de base o de un flujo de vídeo de vista dependiente definido en la manera anteriormente descrita.

15 Según se ilustra en la Figura 39, en el GOP actual, que es un GOP cerrado, las imágenes anteriores (pasadas) con respecto a una imagen IDR o imagen de anclaje entre las imágenes del GOP actual están prohibidas de referirse a las imágenes del GOP anterior. La imagen de anclaje se describirá a continuación.

20 Además, entre las imágenes del GOP actual, las imágenes posteriores (futuras) con respecto a la imagen IDR o imagen de anclaje, en el orden de presentación visual, están prohibidas de referirse a las imágenes del GOP anterior más allá de la imagen IDR o de la imagen de anclaje.

La Figura 40 es un diagrama que ilustra una estructura del GOP abierto de un flujo de vídeo de vista de base o de un flujo de vídeo de vista dependiente.

25 Según se ilustra en la Figura 40, en el GOP actual, que es un GOP abierto, las imágenes antes de una imagen de anclaje no de IDR (imagen de anclaje que no es una imagen IDR), en el orden de presentación visual, entre las imágenes del GOP actual, se permite para referirse a las imágenes del GOP anterior.

30 Además, entre las imágenes del GOP actual, las imágenes después de la imagen de anclaje, no de IDR, en el orden de presentación visual, están prohibidas de referirse a las imágenes del GOP anterior más allá de la imagen de anclaje no de IDR.

35 La estructura del GOP se define esta manera, con lo que las características operativas de estructuras de flujos, un GOP abierto o un GOP cerrado, están en correspondencia entre un determinado GOP de un flujo de vídeo de vista de base y el GOP correspondiente de un flujo de vídeo de vista dependiente.

40 Además, las características operativas de estructuras de referencia de imágenes en correspondencia, es decir, la imagen del vídeo de vista dependiente correspondiente a la imagen B de no referencia del vídeo de vista de base es fiablemente una imagen B no de referencia.

Además, el número de tramas y el número de campos están en correspondencia entre un determinado GOP de un flujo de vídeo de vista de base y el GOP correspondiente de un flujo de vídeo de vista dependiente.

45 De este modo, la estructura del GOP de un flujo de vídeo de vista dependiente se define como la misma estructura que la estructura del GOP de un flujo de vídeo de vista de base, con lo que las mismas características operativas pueden proporcionarse a los grupos GOPs correspondientes a cada otro de los flujos.

50 Además, incluso en el caso de realizar la decodificación desde una parte media de un flujo, la decodificación puede realizarse sin problemas. La decodificación desde una parte media de un flujo se realiza de forma no estándar, o acceso aleatorio, a modo de ejemplo.

55 En un caso en donde la estructura de los grupos GOPs correspondientes a cada otro de los flujos son diferentes, a modo de ejemplo, en un caso en donde los números de tramas son distintos, puede producirse la situación siguiente: uno de los flujos puede reproducirse con normalidad pero el otro flujo no se puede reproducir. Sin embargo, puede evitarse dicha situación.

60 En un caso en donde la decodificación se inicia desde una parte media de un flujo mientras se supone que las estructura de los grupos GOPs correspondientes entre sí entre los flujos son distintos, puede producirse la situación siguiente: una imagen de vídeo de vista de base que es necesaria para decodificar el vídeo de vista dependiente no se decodifica. En este caso, en consecuencia, no se puede decodificar una imagen de vídeo de vista dependiente, de modo que no se puede realizar la presentación visual en 3D. Además, es posible que una imagen de vídeo de vista de base no pueda proporcionarse, a la salida, dependiendo de un método de montaje, pero se puede evitar dicha inconveniencia operativa.

65 Información sobre EP_map

Con el uso de las estructuras de grupos GOP de un flujo de vídeo de vista de base y un flujo de vídeo de vista dependiente, la posición de partida de la decodificación en el instante de acceso aleatorio o no estándar, a modo de truco operativo, puede establecerse para EP_map. EP_map se incluye en un fichero de Clip Information.

5 Las dos restricciones operativas siguientes se proporcionan como las restricciones de una imagen que puede establecerse para EP_map como una posición de partida de la decodificación.

10 1. La posición de una imagen de anclaje dispuesta después de los subconjuntos SubsetSPS o la posición de una imagen IDR dispuesta después de los subconjuntos SubsetSPS se considera como la posición que puede establecerse para un flujo de vídeo de vista dependiente.

La imagen de anclaje es una imagen definida en H.264 AVC/MVC y es una imagen de un flujo de vídeo de vista dependiente codificado realizando la referencia entre vistas sin realizar la referencia a la dirección temporal.

15 2. En un caso en donde una determinada imagen de un flujo de vídeo de vista dependiente se establece para EP_map como una posición de partida de la decodificación, la imagen correspondiente de un flujo de vídeo de vista de base se establece también para EP_map como la posición de partida de la decodificación.

20 La Figura 41 es un diagrama que ilustra, a modo de ejemplo, una posición de partida de la decodificación que se establece para EP_map que satisface las dos restricciones operativas anteriormente descritas.

En la Figura 41, las imágenes que constituyen un flujo de vídeo de vista dependiente y las imágenes que constituyen un flujo de vídeo de vista dependiente se ilustran en el orden de decodificación.

25 Entre las imágenes del flujo de vídeo de vista dependiente, una imagen P_1 mostrada con un color es una imagen de anclaje o una imagen IDR. SubsetSPS se incluye en la Unidad de Acceso inmediatamente antes de la Unidad de Acceso que incluye los datos de la imagen P_1 .

30 En la realización, a modo de ejemplo, ilustrada en la Figura 41, según se indica por una flecha blanca nº 11, la imagen P_1 se establece como posición de partida de la decodificación para EP_map del flujo de vídeo de vista dependiente.

35 Una imagen P_{11} , que es la imagen del flujo de vídeo de vista de base correspondiente a la imagen P_1 es una imagen IDR. Según se indica por una flecha blanca nº 12, la imagen P_{11} que sirve como una imagen IDR se establece también como una posición de partida de la decodificación para EP_map del flujo de vídeo de vista de base.

40 En el caso de que se inicie la decodificación desde la imagen P_1 y la imagen P_{11} en respuesta a una instrucción de acceso aleatorio y no estándar, se realiza primero la decodificación de la imagen P_{11} . La imagen P_{11} que es una imagen IDR, puede decodificarse sin referencia a otra imagen.

45 Después de que haya finalizado la decodificación de la imagen P_{11} , se decodifica a continuación la imagen P_1 . La imagen P_{11} decodificada se refiere, durante la decodificación, con respecto a la imagen P_1 . La imagen P_1 , que es una imagen de anclaje o una imagen IDR, puede decodificarse si está concluida la decodificación de la imagen P_{11} .

Después de lo que antecede, se realiza la decodificación en el orden de la siguiente imagen de la imagen P_1 del vídeo de vista de base, la siguiente imagen de la imagen P_{11} del vídeo de vista dependiente y así sucesivamente.

50 Puesto que las estructuras de los grupos GOPs correspondientes son las mismas y la decodificación se inicia a partir de las posiciones correspondientes, las imágenes establecidas para EP_map y las imágenes posteriores pueden decodificarse sin problemas para el vídeo de vista de base y el vídeo de vista dependiente, a la vez. En consecuencia, se puede realizar un acceso aleatorio.

55 Las imágenes dispuestas a la izquierda de la línea de puntos que se extiende en la dirección vertical en la Figura 41 son imágenes que no están decodificadas.

La Figura 42 es un diagrama que ilustra un problema que surge en un caso en donde la estructura del GOP de vídeo de vista dependiente no está definida.

60 En la realización, a modo de ejemplo, ilustrada en la Figura 42, una imagen P_{21} mostrada con un color, que es una imagen IDR de vídeo de vista de base, se establece para EP_map como una posición de partida de la decodificación.

65 Se supone un caso en donde una imagen P_{31} , que es la imagen de vídeo de vista dependiente correspondiente a la imagen P_{21} no es una imagen de anclaje en el caso de que se inicie la decodificación desde la imagen P_{21} de vídeo de vista de base. En un caso en donde no está definida una estructura del GOP, no está asegurado que la imagen de vídeo de vista dependiente correspondiente a la imagen IDR de vídeo de vista de base sea una imagen IDR o

una imagen de anclaje.

En este caso, incluso después de que haya finalizado la decodificación de la imagen P_{21} del vídeo de vista de base, no se puede decodificar la imagen P_{31} . La diferencia en la dirección temporal es también necesaria para la decodificación de la imagen P_{31} , pero las imágenes a la izquierda de la línea de puntos que se extiende en la dirección vertical (imágenes anteriores en el orden de la decodificación) no están decodificadas.

La imagen P_{31} no se puede decodificar y en consecuencia, las otras imágenes del vídeo de vista dependiente, que se refieren a la imagen P_{31} no se puede decodificar.

Dicha situación puede evitarse definiendo la estructura del GOP del flujo de vídeo de vista dependiente.

La posición de partida de la decodificación se establece con EP_map en no solamente el vídeo de vista de base sino también en el vídeo de vista dependiente, con lo que el dispositivo de reproducción 1 puede especificar fácilmente la posición de partida de la decodificación.

En un caso en donde solamente una determinada imagen de vídeo de vista de base se establece para EP_map como una posición de partida de la decodificación, el dispositivo de reproducción 1 necesita especificar la imagen de vídeo de vista dependiente correspondiente a la imagen en la posición de partida de la decodificación utilizando el cálculo, lo que complica el proceso.

Incluso si las imágenes correspondientes a cada otro del vídeo de vista de base y del vídeo de vista dependiente tienen los mismos DTS/PTS las estructuras matriciales de bytes en TSs no se pueden poner en correspondencia si las tasas binarias de los vídeos son distintas entre sí, lo que complica también el proceso.

La Figura 43 es un diagrama que ilustra el concepto de búsqueda de imagen que se necesita para realizar el acceso aleatorio o no estándar, a modo de truco operativo, en un flujo MVC constituido por un flujo de vídeo de vista de base y un flujo de vídeo de vista dependiente.

Según se ilustra en la Figura 43, cuando se realiza un acceso aleatorio o no estándar, se busca una imagen de anclaje no IDR o una imagen IDR y se determina una posición de partida de la decodificación.

A continuación, se describirá EP_map. Se proporcionará una descripción de un caso en donde la posición de partida de la decodificación del vídeo de vista de base se establece a EP_map. De modo similar, la posición de partida de la decodificación del vídeo de vista dependiente se establece a EP_map del vídeo de vista dependiente.

La Figura 44 es un diagrama que ilustra la estructura de un flujo AV registrado en el disco óptico 2.

Un TS que incluye un flujo de vídeo de vista de base está constituido por un número entero de unidades alineadas (Aligned Units) que tienen una magnitud de 6144 bytes.

Cada unidad alineada está constituida por 32 paquetes origen (Source Packets). Cada paquete origen tiene 192 bytes. Un paquete origen está constituido por una cabecera extra de paquete de transporte de 4 bytes (cabecera TP_extra) y un paquete de transporte de 188 bytes (Transport Packet).

Los datos del vídeo de vista de base son empaquetados en paquetes de MPEG2 PES. Un paquete PES se forma añadiendo una cabecera de paquete PES a la parte de datos del paquete PES. La cabecera del paquete PES incluye un identificador ID de flujo que especifica el tipo del flujo elemental transmitido por el paquete PES.

El paquete PES se incorpora, además, en los paquetes de transporte. Es decir, el paquete PES se divide en la magnitud de la carga útil de un paquete de transporte, se añade una cabecera de paquete de transporte a la carga útil, con lo que se forma un paquete de transporte. La cabecera del paquete de transporte incluye un PID, que es la información de identificación de datos memorizada en la carga útil.

Conviene señalar que un número de paquetes origen, que se incrementa en un por cada paquete origen con la cabecera de un flujo AV Clip siendo 0, a modo de ejemplo, se proporciona a cada paquete origen. Además, una unidad alineada se inicia desde el primer byte de un paquete origen.

EP_map se utiliza para buscar una dirección de datos en donde debe iniciarse la lectura de datos en un fichero de flujo AV Clip cuando se proporciona una marca temporal de un punto de acceso de Clip. EP_map es una lista de puntos de entrada que se extrae desde un flujo elemental y un flujo de transporte.

EP_map tiene información de dirección para buscar un punto de entrada en el que debe iniciarse la decodificación en un flujo AV. Un elemento de datos EP en EP_map está constituido por un par de un PTS y una dirección en el flujo AV de la Unidad de Acceso correspondiente al PTS. En AVC/H.264, los datos de una imagen se memorizan en una sola Unidad de Acceso.

La Figura 45 es un diagrama que ilustra, a modo de ejemplo, un flujo AV Clip.

El flujo AV Clip en la Figura 45 es un flujo de vídeo (flujo de vídeo de vista de base) constituido por paquetes origen identificados por $PID = x$. En el flujo de vídeo, cada paquete origen se distingue por un PID incluido en la cabecera del paquete de transporte en el paquete origen.

En la Figura 45, entre los paquetes origen del flujo de vídeo, un paquete origen que incluye el primer byte de una imagen IDR está coloreado. Los rectángulos sin color representan un paquete origen que incluye datos que no son un punto de acceso aleatorio y un paquete origen que incluye los datos de otro flujo.

A modo de ejemplo, un paquete origen que tiene un número de paquete $X1$ y que incluye el primer byte de una imagen IDR, aleatoriamente accesible, del flujo de vídeo distinguido por $PID = x$ está dispuesta en la posición $PTS = pts(x1)$ en el eje de tiempos del flujo AV Clip.

De modo similar, un paquete origen que incluye el primer byte de una imagen IDR aleatoriamente accesible siguiente se considera como un paquete origen que tiene un número de paquete origen $X2$ y está dispuesto en la posición $PTS = pts(x2)$.

La Figura 46 es un diagrama que ilustra, de forma conceptual, una realización, a modo de ejemplo, de `EP_map` correspondiente al flujo AV Clip en la Figura 45.

Según se ilustra en la Figura 46, `EP_map` está constituido por `stream_PID`, `PTS_EP_start` y `SPN_EP_start`.

`stream_PID` representa el PID de un paquete de protección para transmitir un flujo de vídeo.

`PTS_EP_start` representa el PTS de una Unidad de Acceso que se inicia desde una imagen IDR aleatoriamente accesible.

`SPN_EP_start` representa la dirección de un paquete origen que incluye el primer byte de una Unidad de Acceso que se refiere por el valor de `PTS_EP_start`.

El PID de un flujo de vídeo se memoriza en `stream_PID` y `EP_map_for_one_stream_PID()`, que es información de tabla que muestra la correspondencia entre `PTS_EP_start` y `SPN_EP_start`, se genera en este instante operativo.

A modo de ejemplo, en `EP_map_for_one_stream_PID[0]` del flujo de vídeo de $PID = x$, $PTS = pts(x1)$ y el número de paquetes de origen $X1$, $PTS = pts(x2)$ y el número de paquete origen $X2$, ... y $PTS = pts(xk)$ y el número de paquete origen Xk se describen en una manera correspondiente.

Dicha tabla se genera también para los flujos de vídeo individuales multiplexados en el mismo flujo AV Clip. `EP_map` que incluye las tablas generadas se memorizan el fichero de Clip Information correspondiente al flujo AV Clip.

La Figura 47 es un diagrama que ilustra, a modo de ejemplo, una estructura de datos del paquete origen indicado por `SPN_EP_start`.

Según se describió con anterioridad, un paquete origen está constituido en una forma en donde se añade una cabecera de 4 bytes a un paquete de transporte de 188 bytes. La parte de paquete de transporte está constituida por una parte de cabecera (cabecera TP) y una parte de carga útil. `SPN_EP_start` representa el número de paquete origen de un paquete origen que incluye el primer byte de una Unidad de Acceso comenzando desde una imagen IDR.

En AVC/H.264, una Unidad de Acceso, es decir, una imagen que se inicia desde un delimitador de AU (delimitador de Unidad de Acceso). El delimitador AU va seguido por SPS y PPS. A continuación, la parte de cabecera de la parte completa de los datos de un segmento de una imagen IDR es objeto de memorización.

El valor del parámetro `payload_unit_start_indicator` en la cabecera de TP del paquete de transporte si es 1 representa que un nuevo paquete PES se inicia a partir de la carga útil de este paquete de transporte. Una Unidad de Acceso se inicia a partir de este paquete origen.

Dicho `EP_map` se prepara para cada uno de un flujo de vídeo de vista de base y un flujo de vídeo de vista dependiente.

[Operación 3]

Un POC (Conteo de Orden de Imagen) se establece para las imágenes individuales que constituyen un flujo de vídeo de vista de base y un flujo de vídeo de vista dependiente en el instante de la codificación. El POC es un valor que representa el orden de presentación visual de la imagen.

En AVC/H.264, el POC se define como sigue: “Una variable que tiene un valor que es no decreciente con el incremento de la posición de imagen en el orden de salida en relación con la imagen IDR anterior en el orden de la decodificación o en relación con la imagen anterior que contiene la operación de control de gestión de memoria que marca todas las imágenes de referencia como “no utilizadas para referencia”.

5 En el instante de la codificación, el POC establecido para una imagen de un flujo de vídeo de vista de base y el POC establecido para una imagen de un flujo de vídeo de vista dependiente se utilizan de una manera uniforme.

10 A modo de ejemplo, POC = 1 se establece para la primera imagen en el orden de presentación visual de un flujo de vídeo de vista de base. En adelante, POCs se establecen para las imágenes individuales con incremento de 1.

15 Además, POC = 1, que es el mismo que se establece para la primera imagen del flujo de vídeo de vista de base, se establece para la primera imagen, en el orden de presentación visual, del flujo de vídeo de vista dependiente. En adelante, POCs se establecen para las imágenes individuales con incremento de 1.

20 Según se describió con anterioridad, puesto que la estructura del GOP del flujo de vídeo de vista de base es la misma que la estructura del GOP del flujo de vídeo de vista dependiente, los mismos POCs se establecen para las imágenes correspondientes entre sí, en el orden de presentación visual, en las imágenes individuales del flujo de vídeo de vista de base y el flujo de vídeo de vista dependiente.

25 En consecuencia, el dispositivo de reproducción 1 puede procesar los componentes de vista en donde los mismos POCs se establecen considerándolos como componentes de vista correspondientes entre sí en el orden de presentación visual.

30 A modo de ejemplo, el dispositivo de reproducción 1 es capaz de procesar la imagen en la que POC = 1 se establece entre las imágenes de un flujo de vídeo de vista de base y la imagen en el que POC = 1 se establece entre las imágenes de un flujo de vídeo de vista dependiente como imágenes correspondientes entre sí.

35 Además, la SEI (Información de Mejora Suplementaria) de la temporización de imágenes se establece en las imágenes individuales que constituyen el flujo de vídeo de vista de base y el flujo de vídeo de vista dependiente. SEI es información adicional que incluye información auxiliar sobre la decodificación, que se define por H.264/AVC.

40 La SEI de temporización de imágenes, que es una de SEIs, incluye información temporal, tal como el tiempo de lectura desde una memoria CPB (Memoria de Imágenes Codificadas) en el instante de la codificación y un tiempo de lectura desde una memoria DPB (DPB 151 en la Figura 22) en el instante de la decodificación. Además, la SEI de temporización de imágenes incluye información sobre un tiempo de presentación visual e información sobre una estructura de imagen.

45 En el instante de la codificación, la SEI de temporización de imágenes establecido para las imágenes de un flujo de vídeo de vista de base y la SEI de temporización de imágenes establecido para las imágenes de un flujo de vídeo de vista dependiente se utilizan en una manera uniforme.

50 A modo de ejemplo, en un caso en donde T1 se establece como un tiempo de lectura desde CPB a la primera imagen en el orden de la codificación del flujo de vídeo de vista de base, T1 se establece también como un tiempo de lectura desde la CPB a la primera imagen en el orden de codificación del flujo de vídeo de vista dependiente.

55 Es decir, la SEI de temporización de imágenes que tiene el mismo contenido se establece para las imágenes correspondientes entre sí en el orden de la codificación u orden de decodificación entre las imágenes individuales del flujo de vídeo de vista de base y del flujo de vídeo de vista dependiente.

60 En consecuencia, el dispositivo de reproducción 1 es capaz de procesar las componentes de vista en donde la misma SEI de temporización de imágenes se establece como componentes de vista correspondientes entre sí en el orden de la decodificación.

65 El POC y SEI de temporización de imágenes se incluyen en un flujo elemental de vídeo de vista de base y vídeo de vista dependiente y se refieren por el decodificador de vídeo 110 en el dispositivo de reproducción 1.

El decodificador de vídeo 110 es capaz de identificar las componentes de vista correspondientes entre sí sobre la base de la información incluida en el flujo elemental. Además, el decodificador de vídeo 110 es capaz de realizar un proceso de decodificación en el orden de decodificación correcto sobre la base de SEI de temporización de imágenes y en el orden de presentación visual correcto sobre la base de POC.

Puesto que no es necesario referirse a una lista Playlist o similar con el fin de identificar las componentes de vista correspondientes entre sí, se pueden tomar medidas cuando surge un problema en una capa del sistema o una capa superior. Además, puede realizarse el montaje del decodificador independiente de que una capa tenga un problema.

Las series anteriormente descritas de procesos pueden ejecutarse por hardware o software. En un caso en donde las series de procesos se ejecutan por software, un programa que constituye el software se instala desde un soporte de registro de programas en un ordenador incorporado en un hardware dedicado o un ordenador personal de uso general.

5 La Figura 48 es un diagrama de bloques que ilustra una configuración, a modo de ejemplo, de hardware de un ordenador que ejecuta las series anteriormente descritas de procesos en conformidad con un programa.

10 Una CPU (Unidad Central de Procesamiento) 501, una memoria ROM (Memoria de Solamente Lectura) 502 y una memoria RAM (Memoria de Acceso Aleatorio) 503 están mutuamente conectadas por intermedio de un bus 504.

15 Una interfaz de entrada/salida 505 está conectada, además, al bus 504. Una unidad de entrada 506 que incluye un teclado, un ratón, etc., y una unidad de salida 507 que incluye una pantalla, un altavoz, etc., están conectadas a la interfaz de entrada/salida 505. Además, una unidad de memorización 508 que incluye un disco duro, una memoria no volátil, etc., una unidad de comunicaciones 509 que incluye una interfaz de red o componente similar y una unidad de disco 510 que activa un soporte extraíble 511 están conectadas al bus 504.

20 En el ordenador que tiene la configuración anterior, la CPU 501 carga un programa memorizado en la unidad de memorización 508 para la memoria RAM 503 por intermedio de la interfaz de entrada/salida 505 y el bus 504 y lo ejecuta, a modo de ejemplo, de modo que se realicen las series de procesos anteriormente descritas.

25 El programa ejecutado por la unidad CPU 501 se proporciona registrándose en el soporte extraíble 511, a modo de ejemplo, o por intermedio de un soporte de transmisión cableado o inalámbrico, tal como una red de área local, la red Internet o una difusión digital y está instalado en la unidad de memorización 508.

El programa ejecutado por el ordenador puede ser un programa en el que se realicen los procesos en series temporales a lo largo del orden descrito en esta especificación o puede ser un programa en el que se realizan los procesos en paralelo o en una temporización necesaria, tal como cuando se llaman los procesos.

30 Una forma de realización de la presente invención no está limitada a la forma de realización antes descrita y se pueden realizar varios cambios sin desviarse por ello del alcance de protección de la presente invención.

Lista de referencias numéricas

35 1 dispositivo de reproducción, 2 disco óptico, 3 dispositivo de presentación visual, 11 codificador MVC, 21 codificador H.264/AVC, 22 decodificador H.264/AVC, 23 unidad de cálculo de fondo, 24 identificador de vídeo de vista dependiente, 25 multiplexor, 51 controlador, 52 unidad de disco, 53 memoria, 54 memoria local, 55 interfaz de Internet, 56 unidad de decodificador, 57 unidad de entrada de operación.

40

REIVINDICACIONES

1. Un dispositivo de registro, que comprende:

5 un medio de codificación (11) para codificar datos de vídeo de una pluralidad de puntos de vista utilizando un método de codificación predeterminado y proporcionando, a la salida, un flujo de datos de vídeo de vista de base constituido por primeras Unidades de Acceso (Unidad de Acceso nº 1) constituidas cada una de ellas por datos sin una cabecera de datos que incluyen información de identificación que indica que la componente de vista memorizada en la primera Unidad de Acceso (Unidad de Acceso nº 1) es un flujo de vídeo de vista de base y un flujo
 10 de datos de vídeo de vista dependiente constituido por segundas Unidades de Acceso (Unidad de Acceso nº 2), constituidas cada una de ellas por datos con una cabecera de datos que incluye información de identificación (view_id) que muestra que la componente de vista memorizada en la segunda Unidad de Acceso (Unidad de Acceso nº 2) es de vídeo de vista dependiente,
 15 en donde los datos de cada imagen de flujo de vídeo de vista de base y de flujo de vídeo de vista dependiente se memorizan, respectivamente, en las primeras Unidades de Acceso y en las segundas Unidades de Acceso.

2. Un dispositivo de registro que comprende:

20 un medio de codificación (401, 402) para codificar datos de vídeo de una pluralidad de puntos de vista utilizando un método de codificación predeterminado, la generación de un flujo de datos de vídeo de vista de base, constituido por primeras Unidades de Acceso (Unidad de Acceso nº 1) constituida cada una por datos con una cabecera de datos que incluye información de identificación (view_id) para identificar la componente de vistas allí memorizada como datos de vídeo de vista de base y un flujo de datos de vídeo de vista dependiente constituido por segundas
 25 Unidades de Acceso (Unidad de Acceso nº 2) constituidas, cada una de ellas, por datos con una cabecera de datos que incluye información de identificación (view_id) para identificar la componente de vista allí memorizada como datos de vídeo de vista dependiente,
 en donde el medio de codificación (401, 402) está adaptado para suprimir la cabecera de datos desde las primeras
 30 Unidades de Acceso (Unidad de Acceso nº 1) que constituye el flujo de datos de vídeo de vista de base y para proporcionar, a la salida, el flujo de datos de vídeo de vista de base que está constituido por datos sin la cabecera de datos y proporcionar el flujo de datos de vídeo de vista dependiente que está constituido por datos con la cabecera de datos,
 35 en donde los datos de cada imagen del flujo de vídeo de vista de base y el flujo de vídeo de vista dependiente se memorizan, respectivamente, en las primeras Unidades de Acceso y en las segundas Unidades de Acceso.

3. El dispositivo de registro según la reivindicación 1 o 2, en donde el medio de codificación establece un valor de uno o más para la cabecera de datos, sirviendo dicho valor como la información de identificación que muestra que
 40 los datos son datos de un flujo de vídeo de vista dependiente y proporciona, a la salida, el flujo de datos de vídeo de vista dependiente.

4. Un método de registro que comprende las etapas de:

45 codificar (11) datos de vídeo de una pluralidad de puntos de vista utilizando un método de codificación predeterminado y proporcionando, a la salida, un flujo de datos de vídeo de vista de base constituido por primeras Unidades de Acceso (Unidad de Acceso nº 1) que está constituida, cada una de ellas, por datos sin una cabecera de datos que incluye información de identificación de que la componente de vista memorizada en la primera Unidad de Acceso (Unidad de Acceso nº 1) es de vídeo de vista de base y un flujo de datos de vídeo de vista dependiente
 50 constituido por segundas Unidades de Acceso (Unidad de Acceso nº 2) estando constituida, cada una de ellas, por datos con una cabecera de datos que incluye información de identificación (view_id) que muestra que la componente de vista memorizada en la segunda Unidad de Acceso (Unidad de Acceso nº 2) es de vídeo de vista dependiente,
 en donde los datos de cada imagen del flujo de vídeo de vista de base y el flujo de vídeo de vista dependiente se
 55 memorizan, respectivamente, en las primeras Unidades de Acceso y en las segundas Unidades de Acceso.

5. Un método de registro que comprende las etapas de:

60 codificar (401) datos de vídeo de una pluralidad de puntos de vista utilizando un método de codificación predeterminado, la generación de un flujo de datos de vídeo de vista de base constituido por primeras Unidades de Acceso (Unidad de Acceso nº 1) estando, cada una de ellas, constituida por datos con una cabecera de datos que incluye información de identificación (view_id) para identificar la componente de vídeo allí memorizada como vídeo de vista de base y un flujo de datos de vídeo de vista dependiente constituido por segundas Unidades de Acceso (Unidad de Acceso nº 2) estando constituida, cada una de ellas, por datos con una cabecera de datos que incluye
 65 información de identificación (view_id) para identificar la componente de vista allí memorizada como de vídeo de vista dependiente,

suprimir (402) la cabecera de datos desde las primeras Unidades de Acceso (Unidad de Acceso nº 1) que constituye el flujo de datos de vídeo de vista de base y proporcionar, a la salida, el flujo de datos de vídeo de vista de base que está constituido por datos sin la cabecera de datos y proporcionando el flujo de datos de vídeo de vista dependiente que está constituido por datos con la cabecera de datos,

5 en donde los datos de cada imagen del flujo de vídeo de vista de base y el flujo de vídeo de vista dependiente se memorizan, respectivamente, en las primeras Unidades de Acceso y en la segundas Unidades de Acceso.

6. Un programa que hace que un ordenador ejecute un proceso que comprende la etapa de:

10 codificar (11) datos de vídeo de una pluralidad de puntos de vista utilizando un método de codificación predeterminado y proporcionando a la salida, un flujo de datos de vídeo de vista de base constituido por primeras Unidades de Acceso (Unidad de Acceso nº 1) cada una de ellas constituida por datos sin una cabecera de datos que incluye información de identificación de que la componente de vista memorizada en la primera Unidad de Acceso (Unidad de Acceso nº 1) es de vídeo de vista de base y un flujo de datos de vídeo de vista dependiente constituido por segundas Unidades de Acceso (Unidad de Acceso nº 2) que está constituida cada una de ellas, por datos con una cabecera de datos que incluye información de identificación que muestra que la componente de vista memorizada en la segunda Unidad de Acceso (Unidad de Acceso nº 2) es de vídeo de vista dependiente,

15 en donde los datos de cada imagen de los datos de vídeo de vista de base y los datos de vídeo de vista dependiente se memorizan, respectivamente, en las primeras Unidades de Acceso y en las segundas Unidades de Acceso.

7. Un programa que hace que un ordenador ejecute un proceso que comprende la etapa de:

25 codificar (401) datos de vídeo de una pluralidad de puntos de vista utilizando un método de codificación predeterminado, la generación de un flujo de datos de vídeo de vista de base constituido por primeras Unidades de Acceso (Unidad de Acceso nº 1) constituida, cada una de ellas, por datos con una cabecera de datos que incluye información de identificación (view_id) para identificar la componente de vista allí memorizada como datos de vídeo de vista de base y un flujo de datos de vídeo de vista dependiente constituido por segundas Unidades de Acceso (Unidad de Acceso nº 2) que está constituida, cada una de ellas, por datos, con una cabecera de datos que incluye información de identificación (view_id) para identificar la componente de vista allí memorizada como de vídeo de vista dependiente,

35 suprimir (402) la cabecera de datos desde las primeras Unidades de Acceso (Unidad de Acceso nº 1) que constituye el flujo de datos de vídeo de vista de base y proporcionando, a la salida, el flujo de datos de vídeo de vista de base que está constituido por datos sin la cabecera de datos y proporcionando a la salida, el flujo de datos de vídeo de vista dependiente que está constituido por datos con la cabecera de datos,

40 en donde los datos de cada imagen de los datos de vídeo de vista de base y los datos de vídeo de vista dependiente se memorizan, respectivamente, en las primeras Unidades de Acceso y en las segundas Unidades de Acceso.

8. Un medio de registro que memoriza un flujo de datos de vídeo de vista de base que se obtiene codificando (11) datos de vídeo de una pluralidad de puntos de vista utilizando un método de codificación predeterminado y constituido por las primeras Unidades de Acceso (Unidad de Acceso nº 1) que está constituida, cada una de ellas, por datos sin una cabecera de datos que incluye información de identificación de que la componente de vista memorizada en la primera Unidad de Acceso (Unidad de Acceso nº 1) es datos de vídeo de vista de base y un flujo de datos de vídeo de vista dependiente constituido por segundas Unidades de Acceso (Unidad de Acceso nº 2) que está constituida, cada una, por datos con una cabecera de datos que incluye información de identificación que muestra que la componente de vista memorizada en la segunda Unidad de Acceso (Unidad de Acceso nº 2) es de vídeo de vista dependiente,

45 en donde los datos de cada imagen de los datos de vídeo de vista de base y los datos de vídeo de vista dependiente se memorizan, respectivamente, en las primeras Unidades de Acceso y en las segundas Unidades de Acceso.

9. Un dispositivo de reproducción (1) que comprende:

55 un medio de lectura (52) para la lectura, desde un soporte de registro (2) de un flujo de datos de vídeo de vista de base que se obtiene codificando datos de vídeo de una pluralidad de puntos de vista utilizando un método de codificación predeterminado y constituido por primeras Unidades de Acceso (Unidad de Acceso nº 1) que está constituida, cada una de ellas, por datos sin una cabecera de datos que incluye información de identificación de que la componente de vista memorizada en la primera Unidad de Acceso (Unidad de Acceso nº 1) es de vídeo de vista de base y un flujo de datos de vídeo de vista dependiente constituido por segundas Unidades de Acceso (Unidad de Acceso nº 2) que está constituida, cada una de ellas, por datos con una cabecera de datos que incluye información de identificación que tiene un valor de uno o más que indica que la componente de vista memorizada en la segunda Unidad de Acceso (Unidad de Acceso nº 2) es de vídeo de vista dependiente,

en donde los datos de cada imagen de los datos de vídeo de vista de base y los datos de vídeo de vista dependiente se memorizan, respectivamente, en las primeras Unidades de Acceso y en las segundas Unidades de Acceso y

5 un medio de decodificación (56) para considerar los datos del flujo de datos de vídeo de vista de base sin la cabecera de datos como datos con una cabecera de datos en donde un valor de cero se establece como la información de identificación y decodificar los datos de flujos de datos de vídeo de vista de base antes de decodificar los datos del flujo de datos de vídeo de vista dependiente.

10 **10.** Un método de reproducción que comprende las etapas de:

10 leer (52), desde un soporte de registro (2), un flujo de datos de vídeo de vista de base que se obtiene mediante codificación de datos de vídeo de una pluralidad de puntos de vista utilizando un método de codificación predeterminado y constituido por primeras Unidades de Acceso (Unidad de Acceso nº 1) estando cada una de ellas constituida por datos sin una cabecera de datos que incluye información de identificación que indica que la
15 componente de vista memorizada en la primera Unidad de Acceso (Unidad de Acceso nº 1) es de vídeo de vista de base y un flujo de datos de vídeo de vista dependiente constituido por segundas Unidades de Acceso (Unidad de Acceso nº 2) constituidas, cada una de ellas, por datos, con una cabecera de datos que incluye información de identificación que tiene un valor de uno o más mostrando que la componente de vista memorizada en la segunda
20 Unidad de Acceso (Unidad de Acceso nº 2) es de vídeo de vista dependiente,

20 en donde los datos de cada imagen de los datos de vídeo de vista de base y de los datos de vídeo de vista dependiente se memorizan, respectivamente, en las primeras Unidades de Acceso y en las segundas Unidades de Acceso y

25 considerar los datos de flujos de datos de vídeo de vista de base sin la cabecera como datos con una cabecera de datos en donde un valor de cero se establece como la información de identificación y decodificar los datos del flujo de datos de vídeo de vista de base antes de decodificar los datos de flujos de datos de vídeo de vista dependiente.

30 **11.** Un programa que hace que un ordenador ejecute un proceso que comprende las etapas de:

30 leer (52), desde un soporte de registro (2) un flujo de datos de vídeo de vista de base que se obtiene codificando datos de vídeo de una pluralidad de puntos de vista utilizando un método de codificación predeterminado y constituido por primeras Unidades de Acceso (Unidad de Acceso nº 1) estando cada una de ellas constituida por
35 datos sin una cabecera de datos que incluye información de identificación de que la componente de vista memorizada en la primera Unidad de Acceso (Unidad de Acceso nº 1) es de vídeo de vista de base y un flujo de vídeo de vista dependiente constituido por segundas Unidades de Acceso (Unidad de Acceso nº 2) que está constituida, cada una, por datos con una cabecera de datos que incluye información de identificación que tiene un valor de uno o más mostrando que la componente de vista memorizada en la segunda Unidad de Acceso (Unidad de Acceso nº 2) es de vídeo de vista dependiente,

40 en donde los datos de cada imagen de los datos de vídeo de vista de base y de los datos de vídeo de vista dependiente se memorizan, respectivamente, en las primeras Unidades de Acceso y en las segundas Unidades de Acceso y

45 considerar datos del flujo de datos de vídeo de vista de base sin la cabecera de datos como datos con una cabecera de datos en donde un valor de cero se establece como la información de identificación y la decodificación de los datos del flujo de vídeo de vista de base antes de decodificar los datos del flujo de vídeo de vista dependiente.

FIG. 1

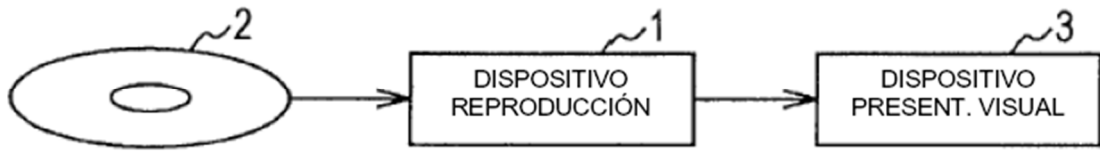


FIG. 2

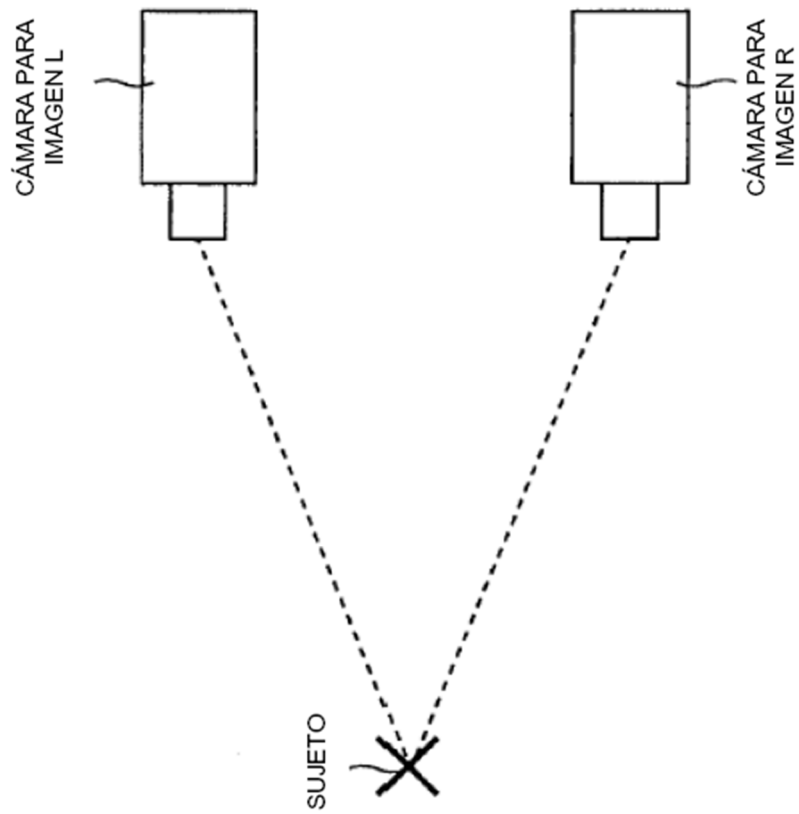


FIG. 3

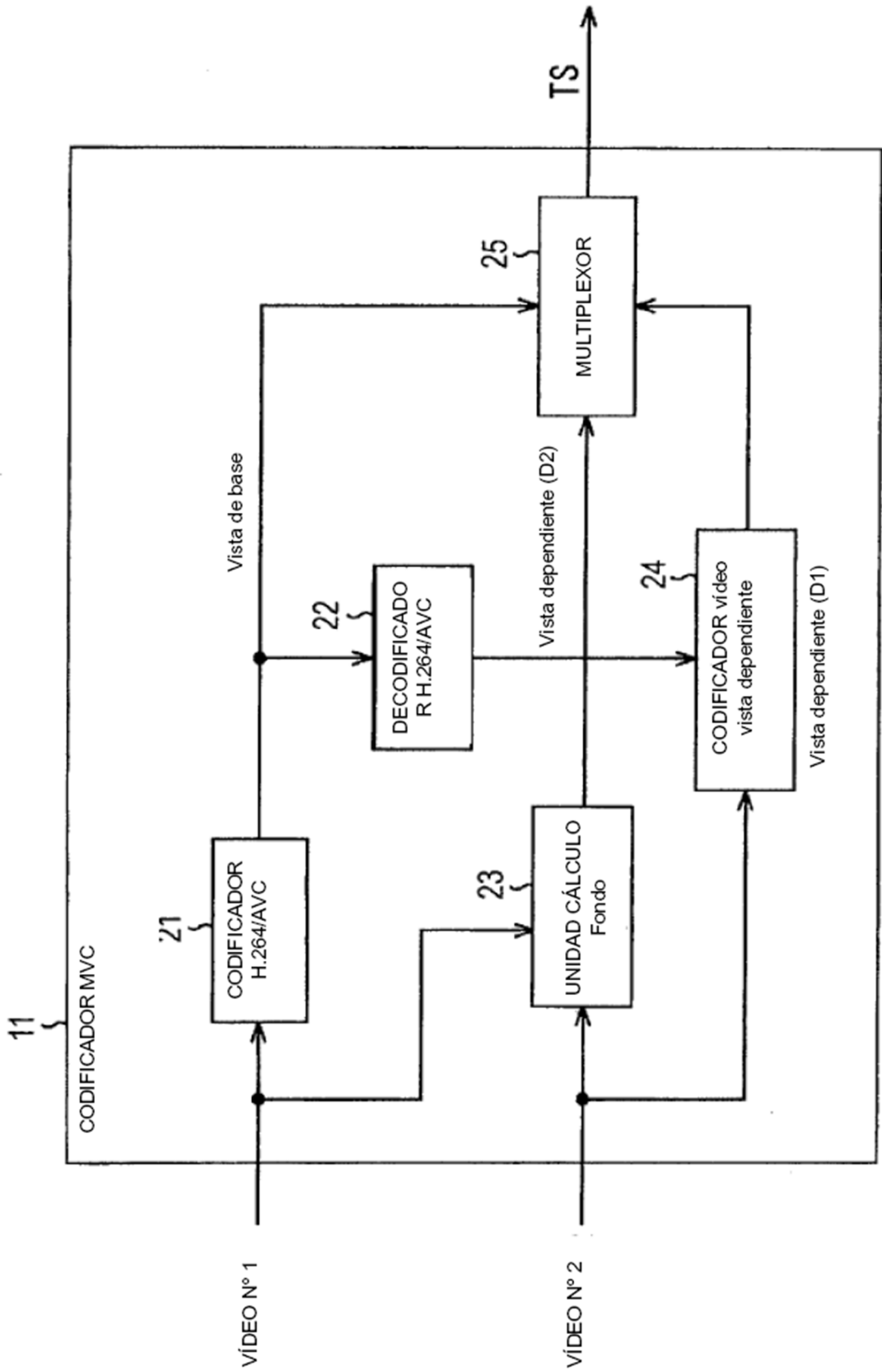


FIG. 4

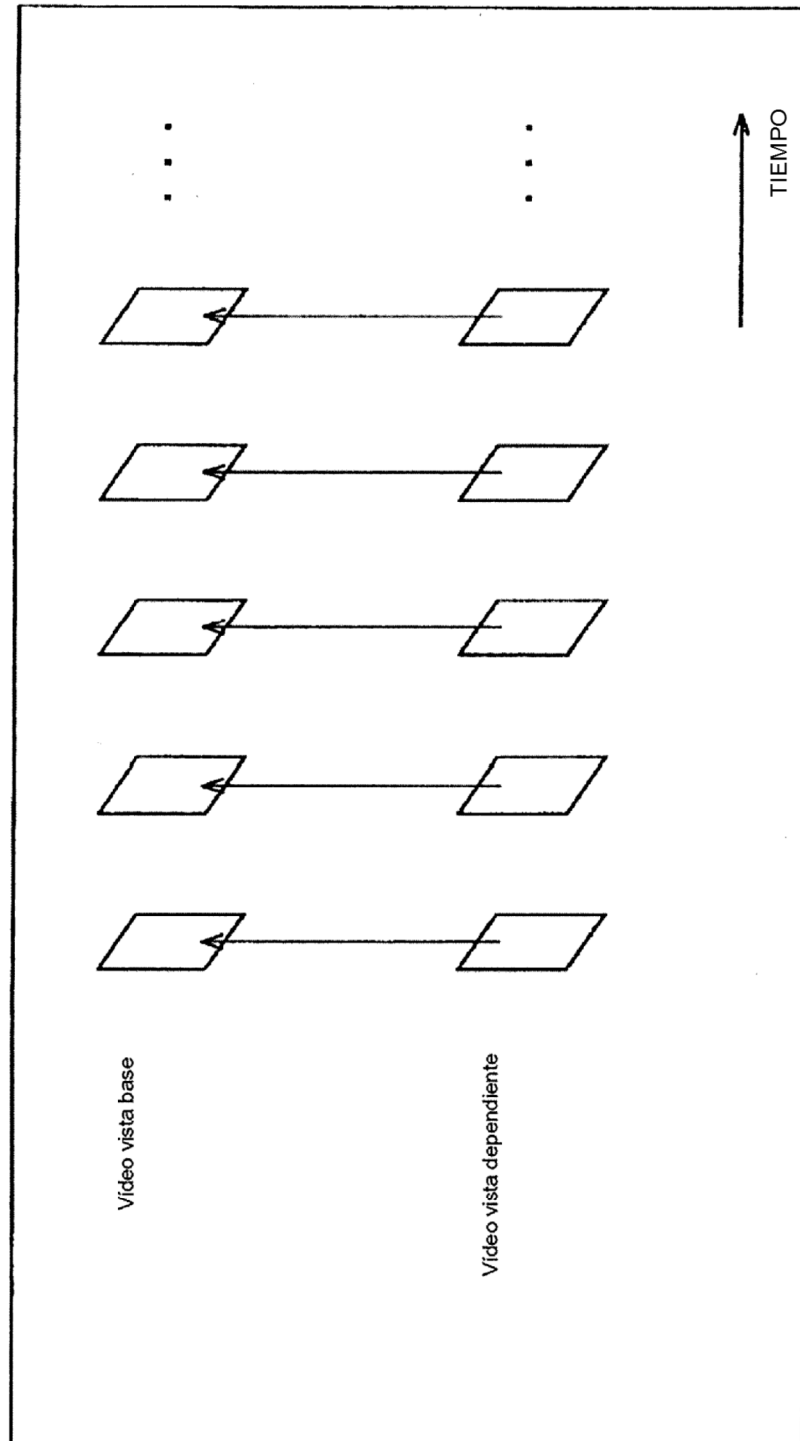


FIG. 5

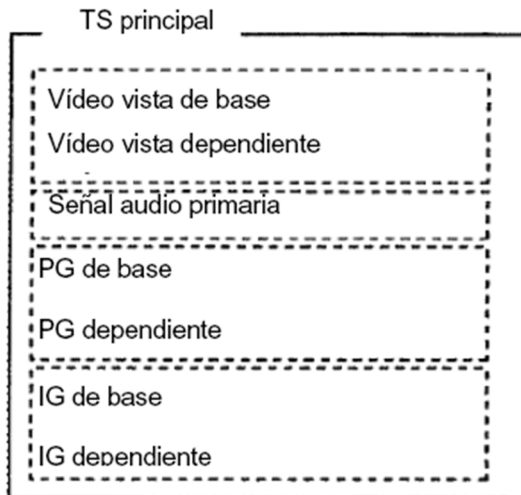


FIG. 6

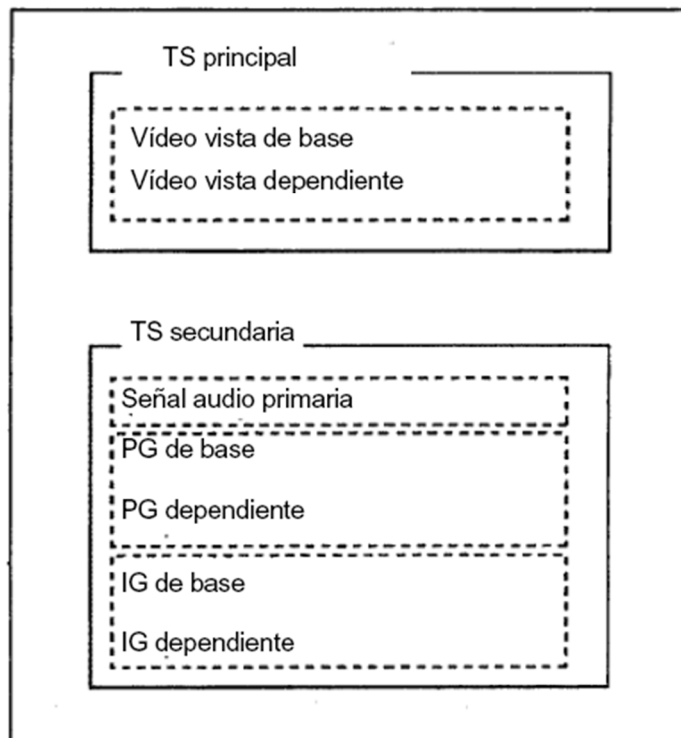


FIG. 7

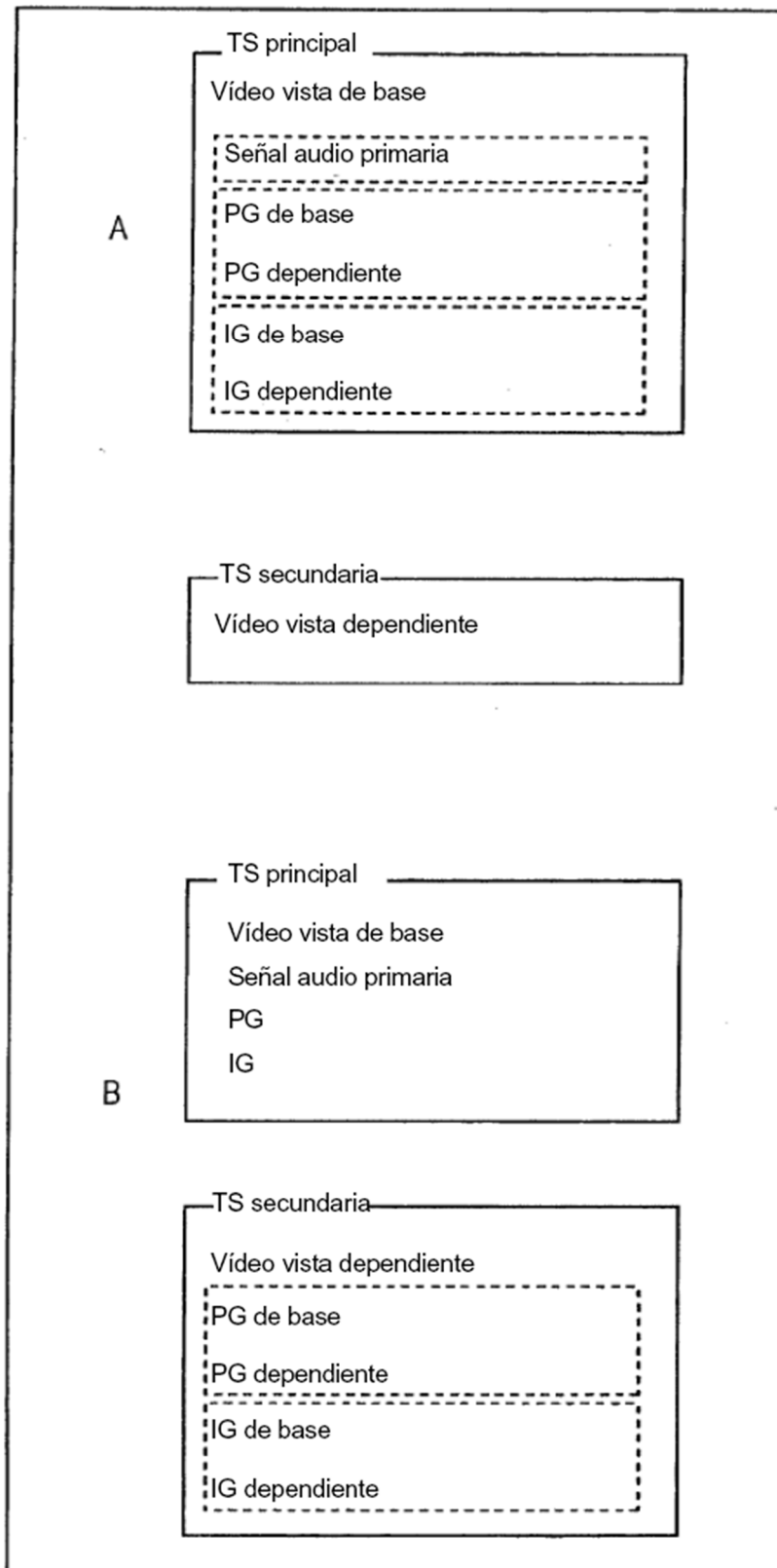


FIG. 8

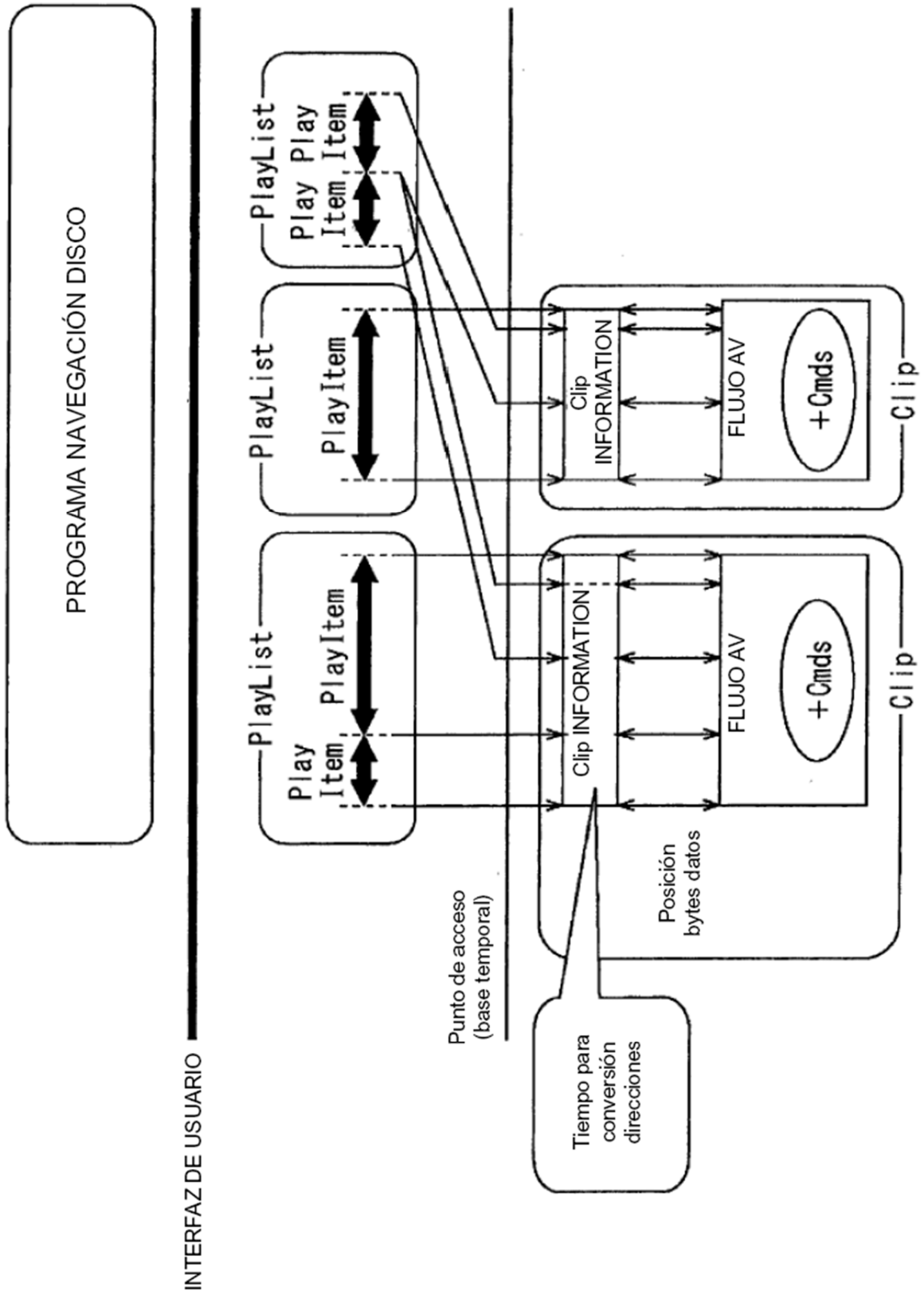


FIG. 9

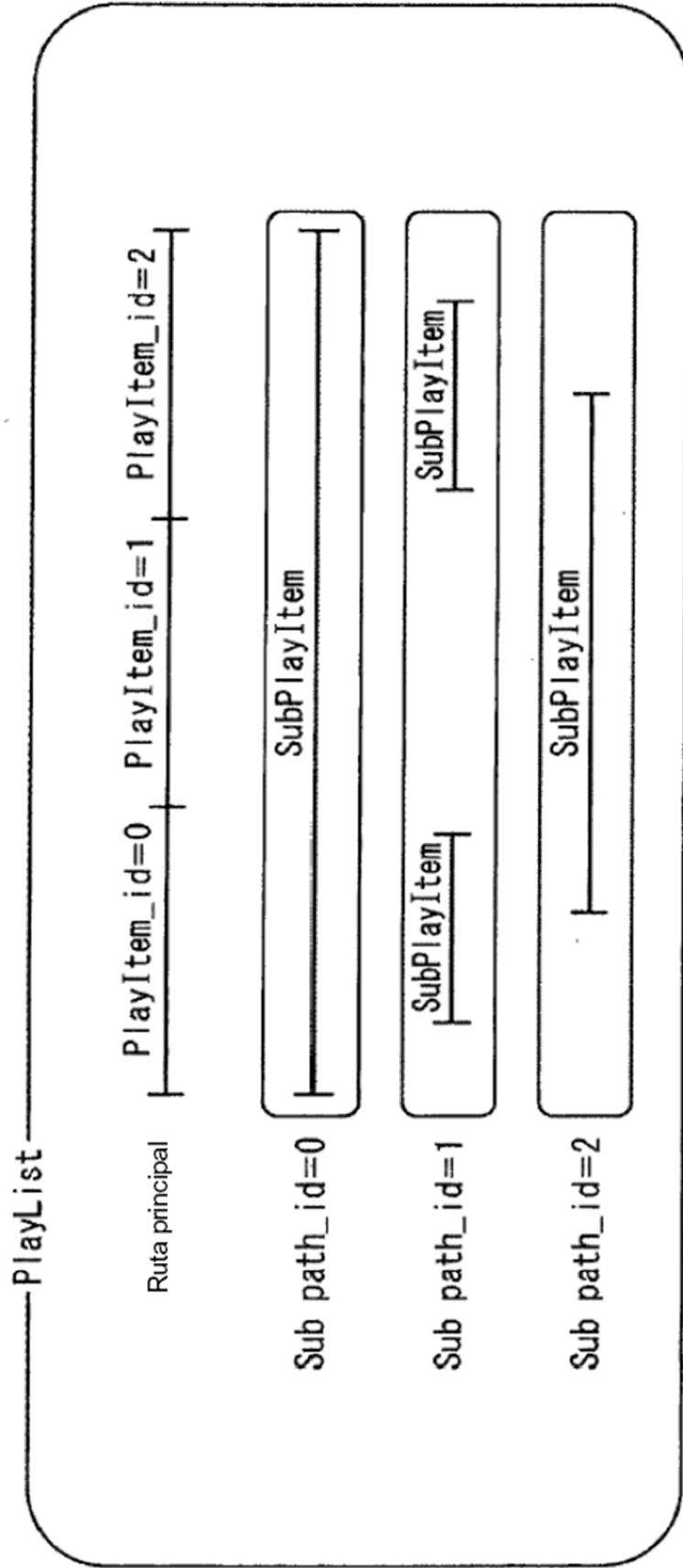


FIG. 10

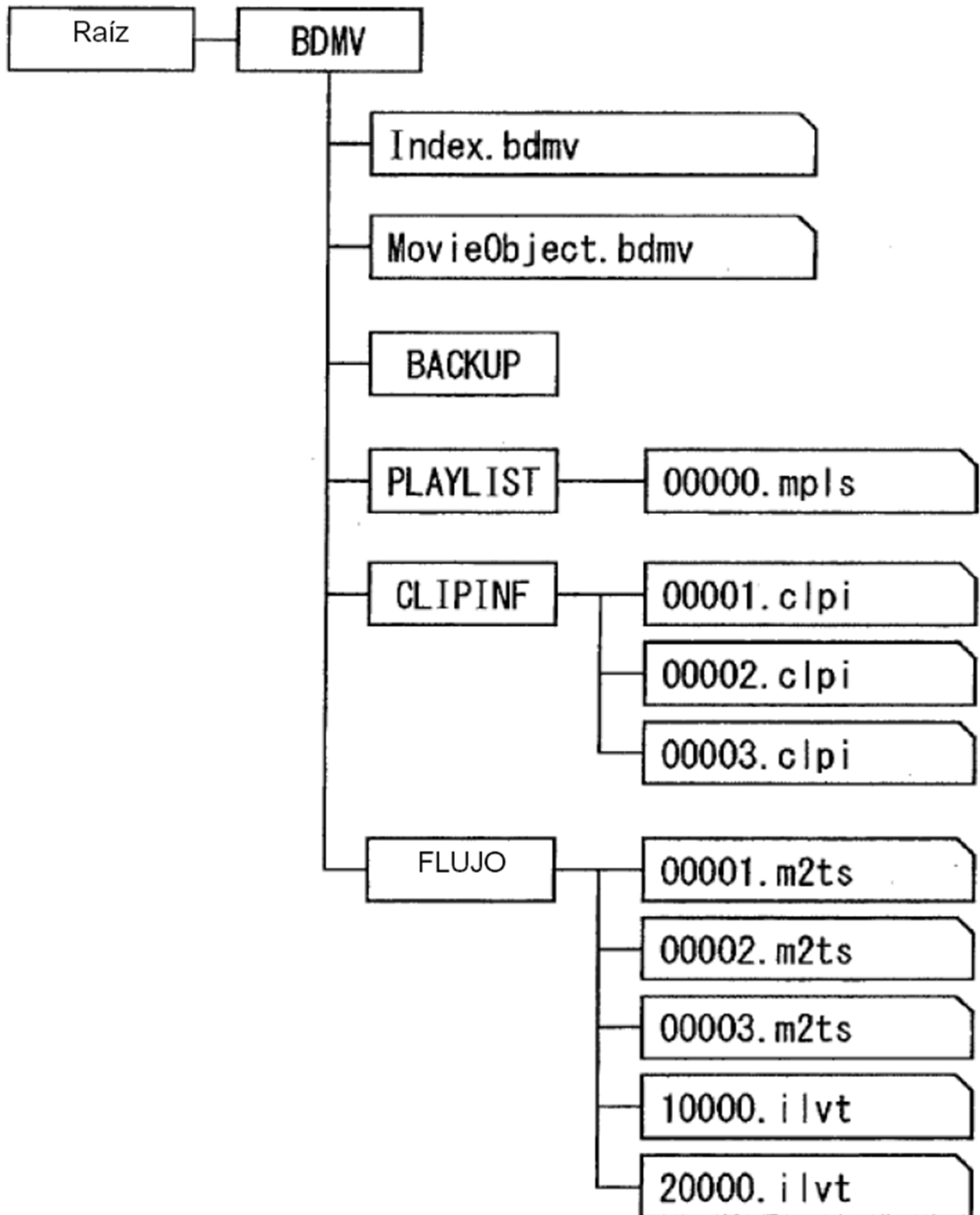


FIG. 11

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
ExtensionData_start_address	32	uimsbf
reserved_for_future_use	160	bslbf
AppInfoPlayList ()		
para (i=0; i<N1; i++) {		
padding_word	16	bslbf
}		
PlayList ()		
para (i=0; i<N2; i++) {		
padding_word	16	bslbf
}		
PlayListMark ()		
para (i=0; i<N3; i++) {		
padding_word	16	bslbf
}		
ExtensionData ()		
para (i=0; i<N4; i++) {		
padding_word	16	bslbf
}		
}		

FIG. 12

xxxxx.mp s {	
... corte ...	
3D_PL_type	2
if (3D_PL_type==01b 3D_PL_type==10) {	
view_type	1
reserved_for_future_use	157
}else {	
reserved_for_future_use	158
}	
... corte ...	
}	

FIG. 13

3D_PL_type	Significado
00	2D PlayList
01	3D B-D1 PlayList
10	3D B-D2 PlayList

FIG. 14

view_type (is_L_Base_view_flag)	Significado
0	Vista L o flujo vídeo AVC en 2D
1	Vista R

FIG. 15

Sintaxis	Nº de bits	Nemónico
Playlist-Sintaxis		
Playlist() {		
longitud	32	uimsbf
reserved_for_future_use	16	bslbf
number_of_PlayItems	16	uimsbf
number_of_SubPaths	16	uimsbf
para (PlayItem_id=0;		
PlayItem_id<number_of_PlayItems;		
PlayItem_id++) {		
PlayItem()		
}		
para (SubPath_id= 0;		
SubPath_id<number_of_SubPaths;		
SubPath_id++) {		
SubPath()		
}		
}		

FIG. 16

SubPath-Sintaxis

Sintaxis	Nº de bits	Nemónico
SubPath() {		
longitud	32	uimbsf
reserved_for_future_use	8	bs bf
SubPath_type	8	uimbsf
reserved_for_future_use	15	uimbsf
is_repeat_SubPath	1	bs bf
reserved_for_future_use	8	bs bf
number_of_SubPlayItems	8	uimbsf
para (i=0; i < number_of_SubPlayItems; i++) {		
SubPlayItem(i)		
}		
}		

FIG. 17

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
si (is_multi_Clip_entries==1b) {		
reserved_for_future_use	8	bslbf
num_of_Clip_entries	8	uimsbf
para (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. 18

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
si (still_mode==0x1) {		
still_time	16	uimsbf
]entonces[
reservado	16	bslbf
}		
si (is_multi_angle==1b) {		
number_of_angles	8	uimsbf
reserved_for_future_use	7	bslbf
is_seamless_angle_change	1	uimsbf
para (angle_id = 1; //Nota: Ángulo después de 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. 19

SIN_tabla ()

Sintaxis	Nº de bits	Nemónico
SIN_tabla ()		
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
para (video_stream_id=0; video_stream_id < number_of_video_stream_entries; video_stream_id++) {		
stream_entry()		
stream_attribute()		
}		
para (audio_stream_id=0; audio_stream_id < number_of_audio_stream_entries; audio_stream_id++) {		
stream_entry()		
stream_attributes()		
}		
para (audio_stream_id2=0; audio_stream_id2 < number_of_audio_stream2_entries; audio_stream_id2++) {		
stream_entry()		
stream_attributes()		
}		
para (PG_textST_stream_id=0; PG_textST_stream_id < number_of_PG_textST_stream_entries; PG_textST_stream_id++) {		
stream_entry()		
stream_attribute()		
}		
para (IG_stream_id=0; IG_stream_id < number_of_IG_stream_entries; IG_stream_id++) {		
stream_entry()		
stream_attribute()		
}		

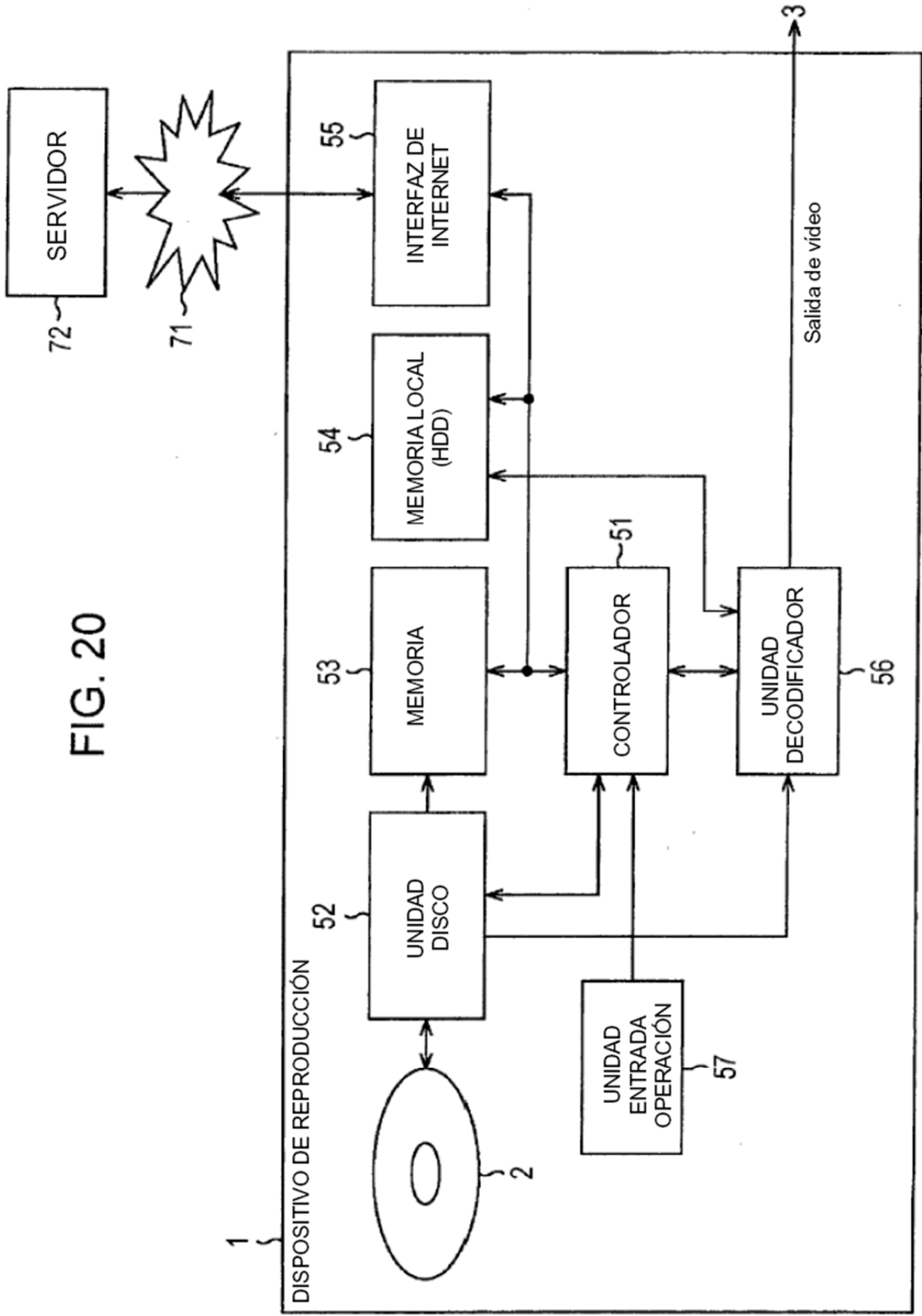


FIG. 20

FIG. 21

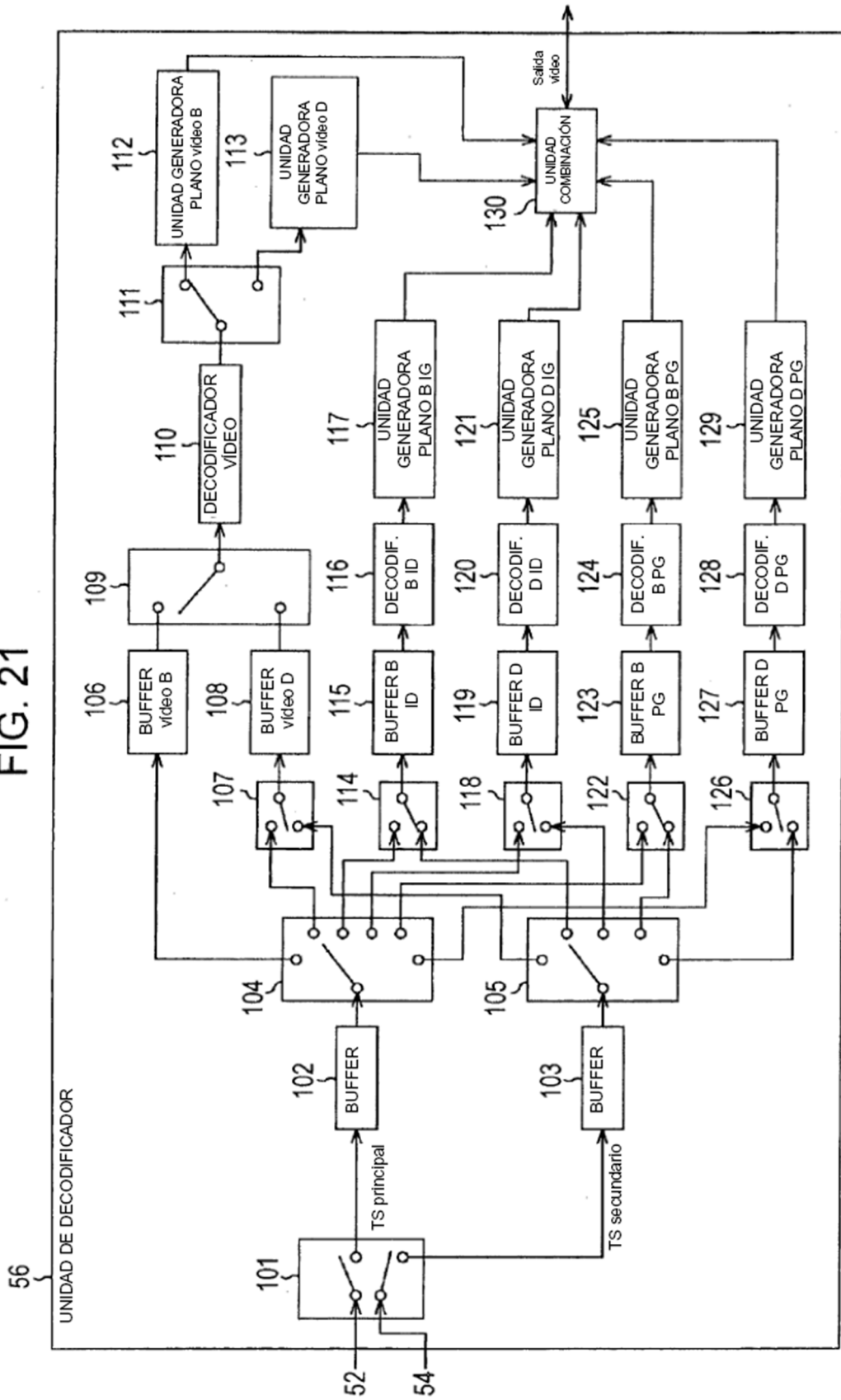


FIG. 22

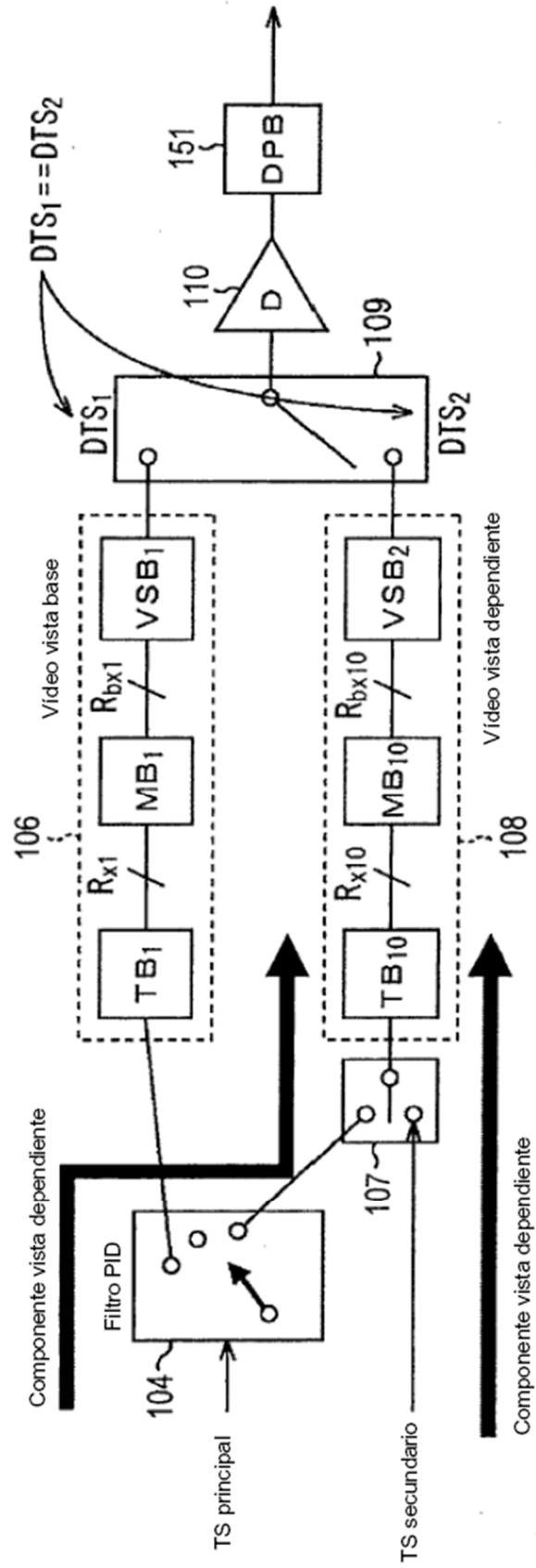


FIG. 24

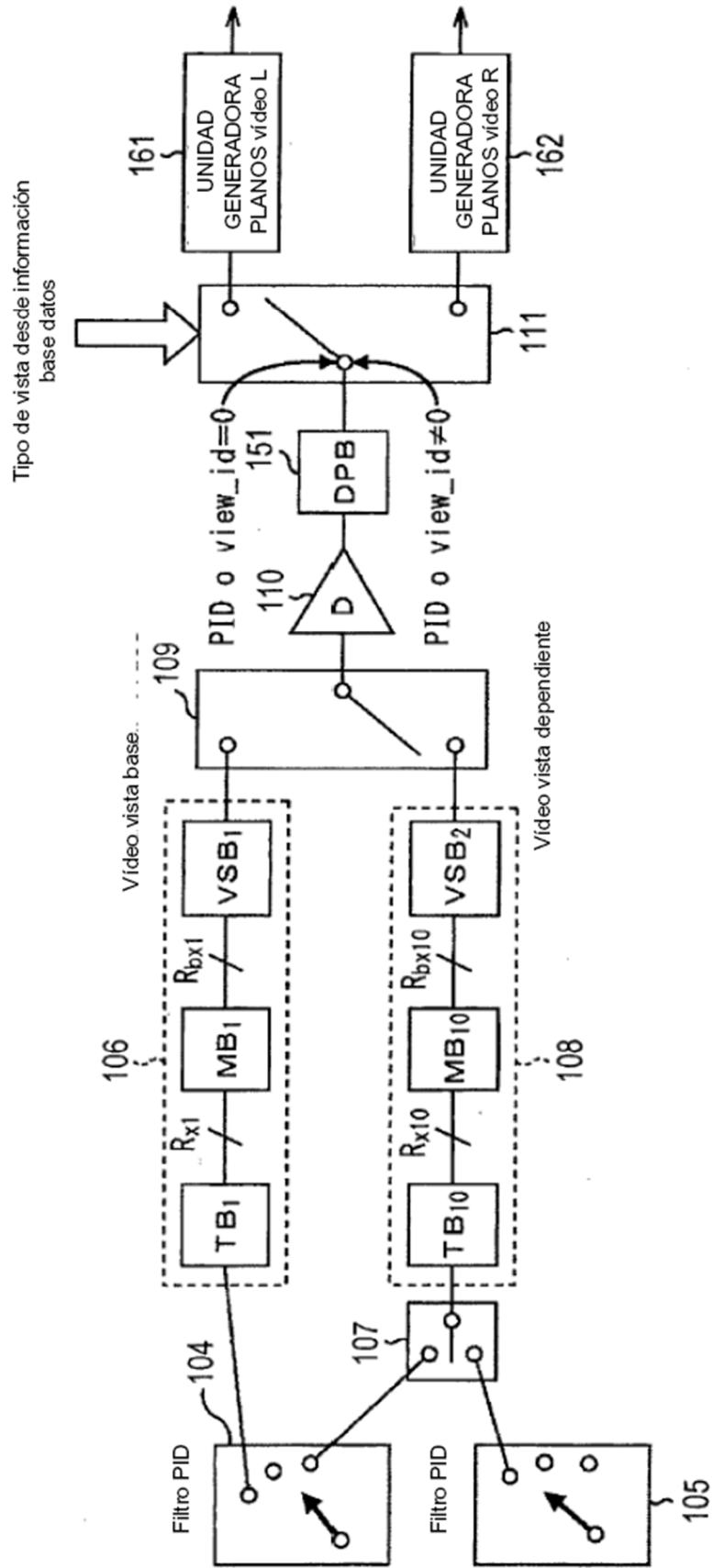


FIG. 25

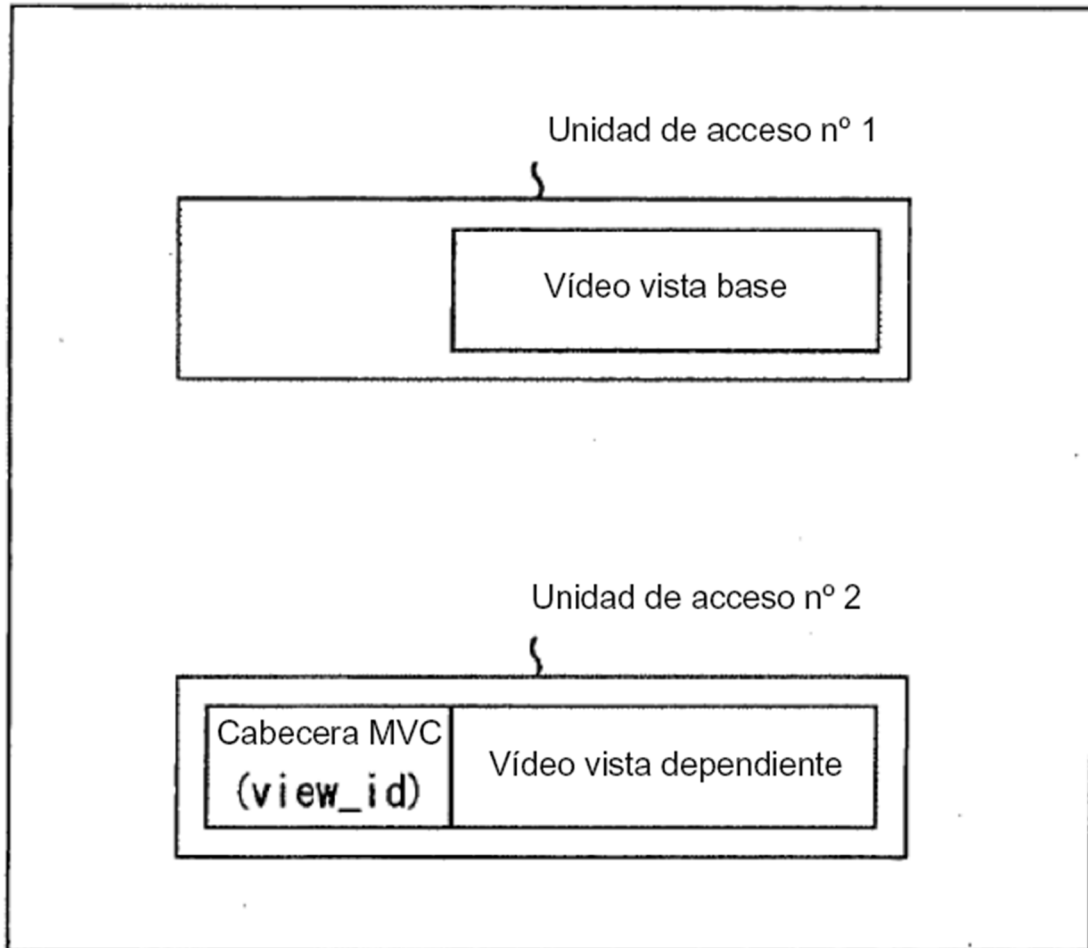


FIG. 26

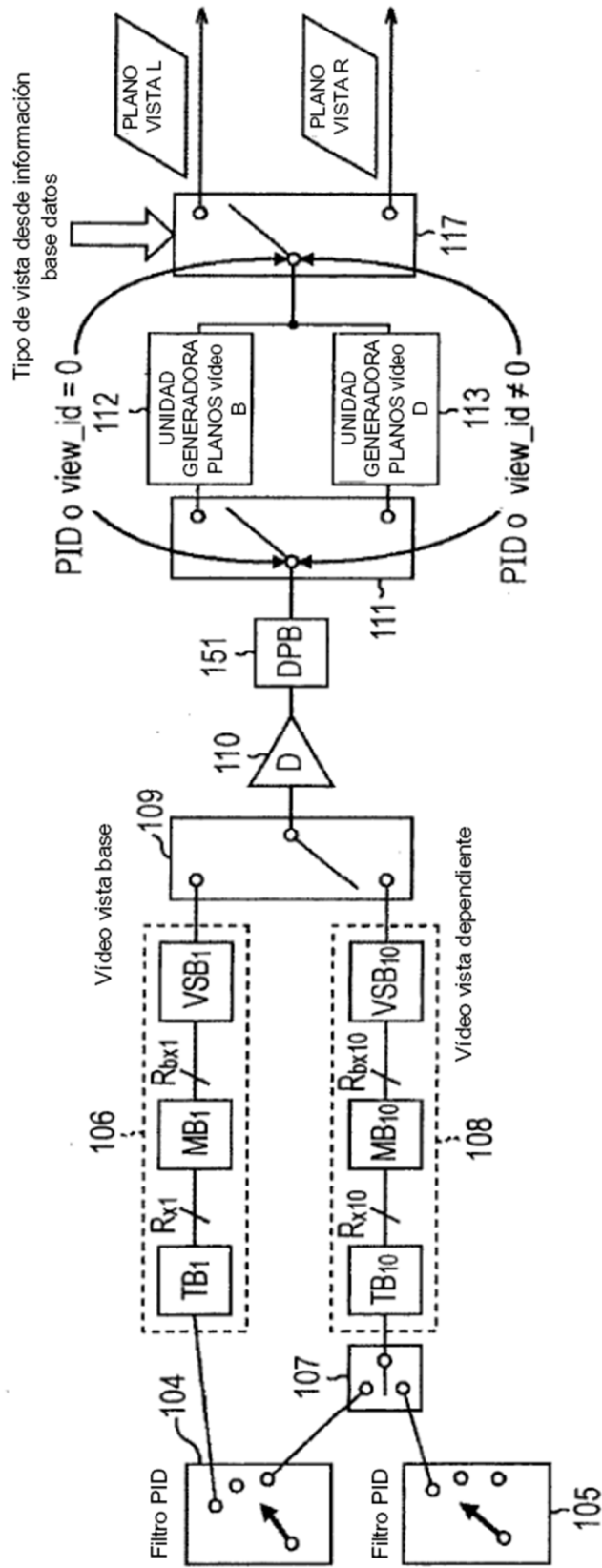


FIG. 27

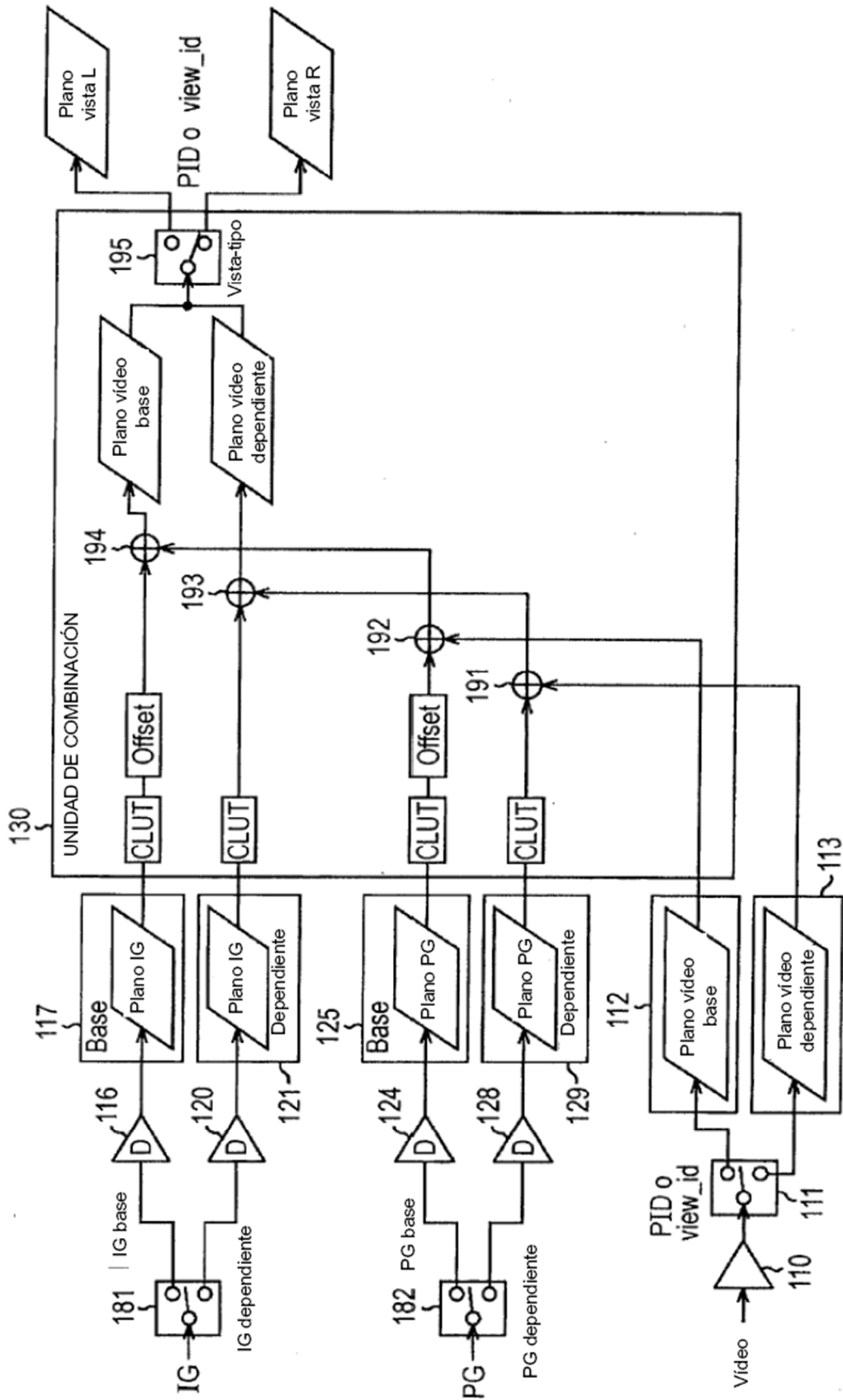


FIG. 28

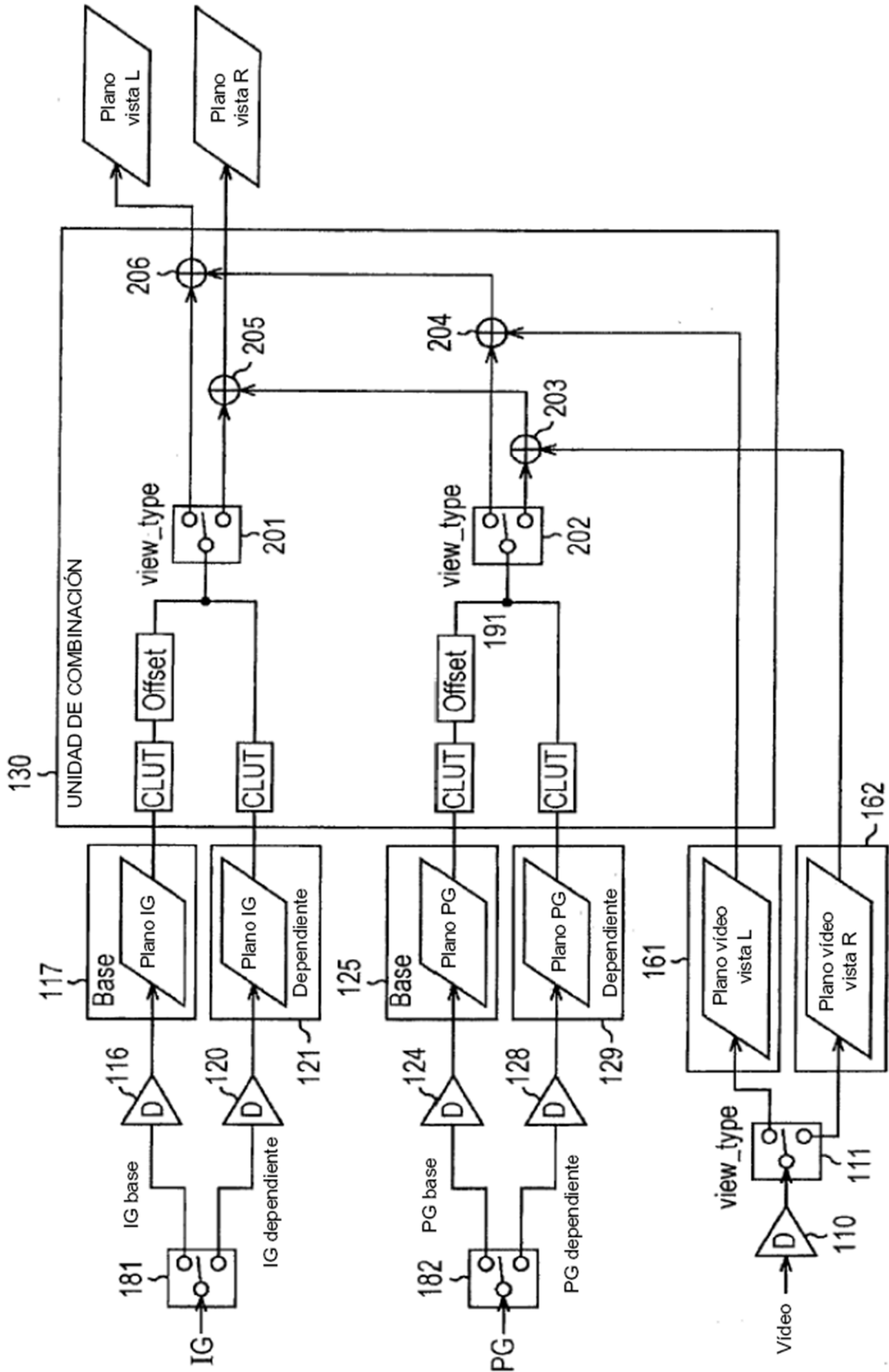


FIG. 29

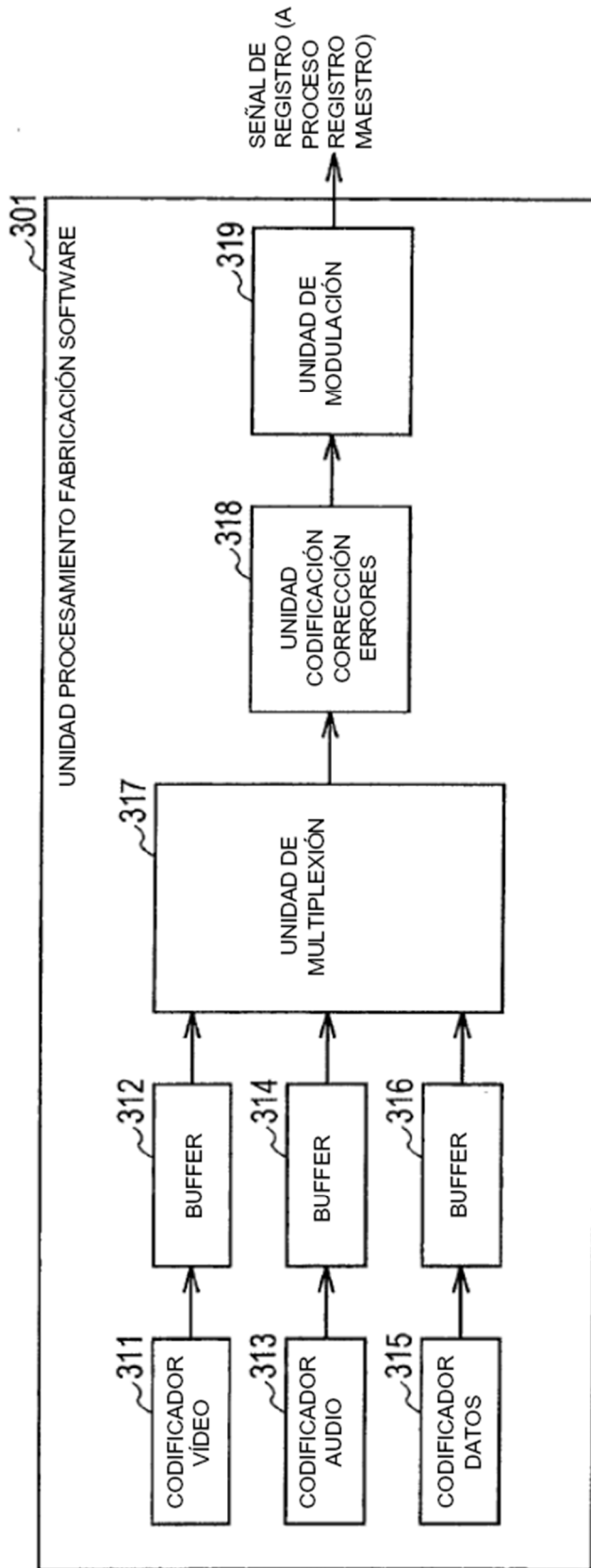


FIG. 30

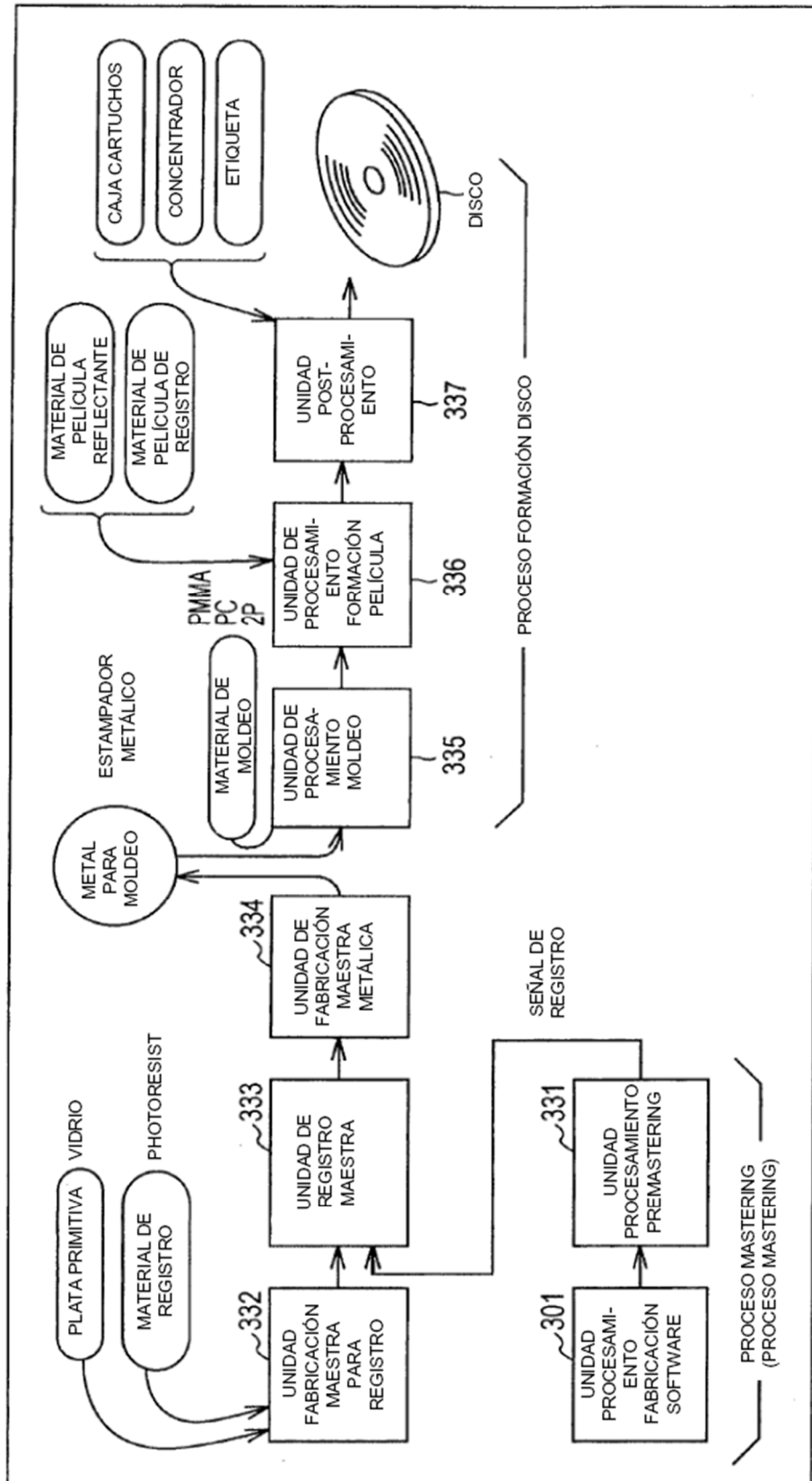


FIG. 31

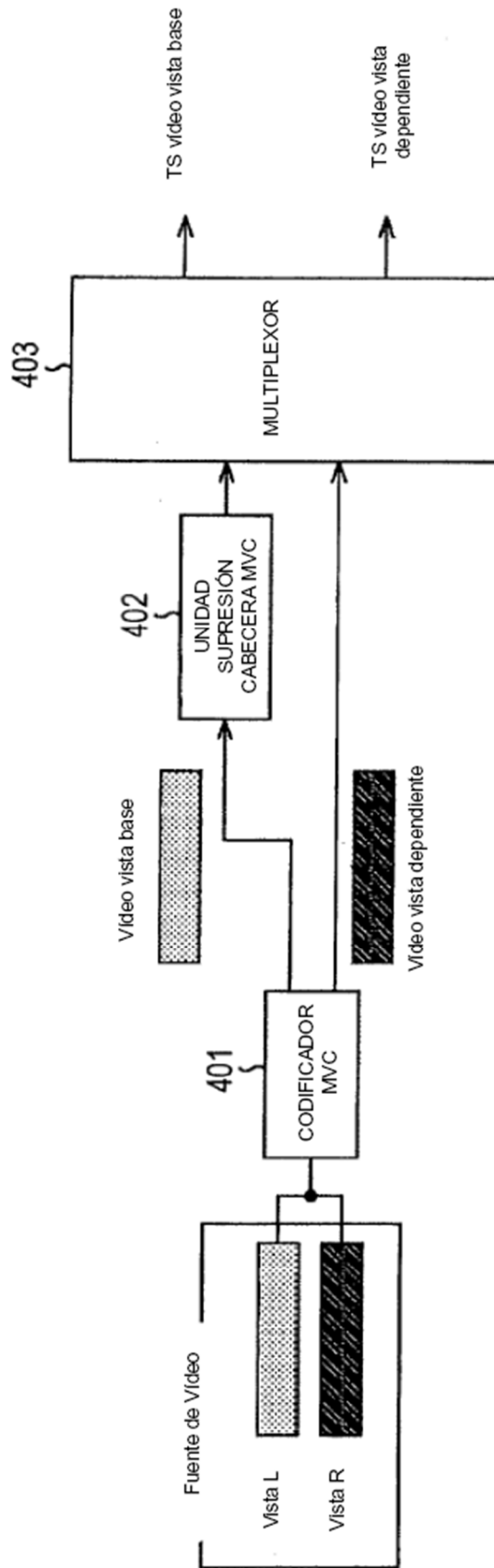


FIG. 32

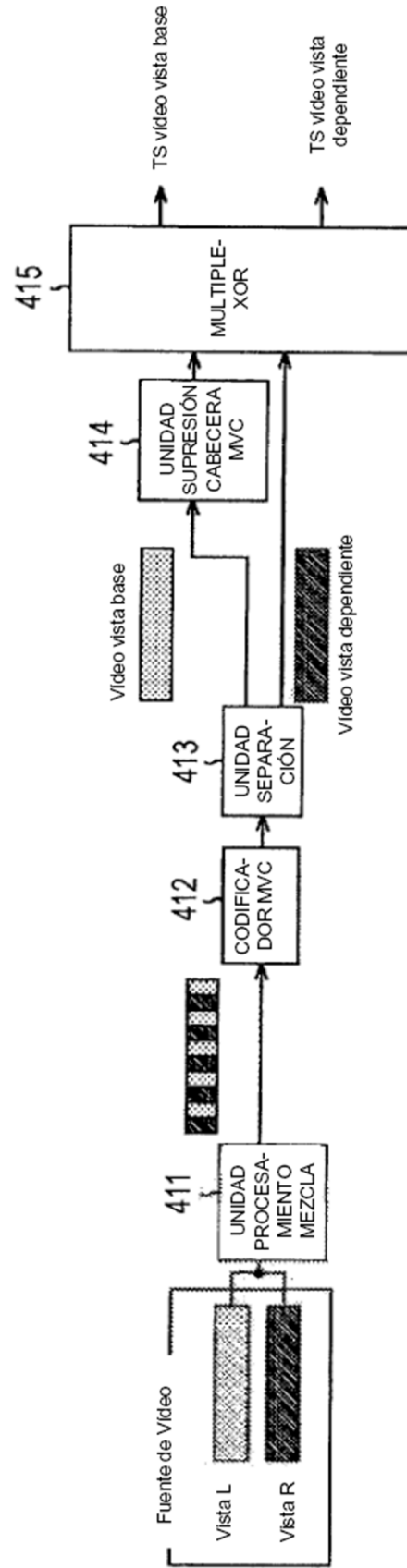


FIG. 33

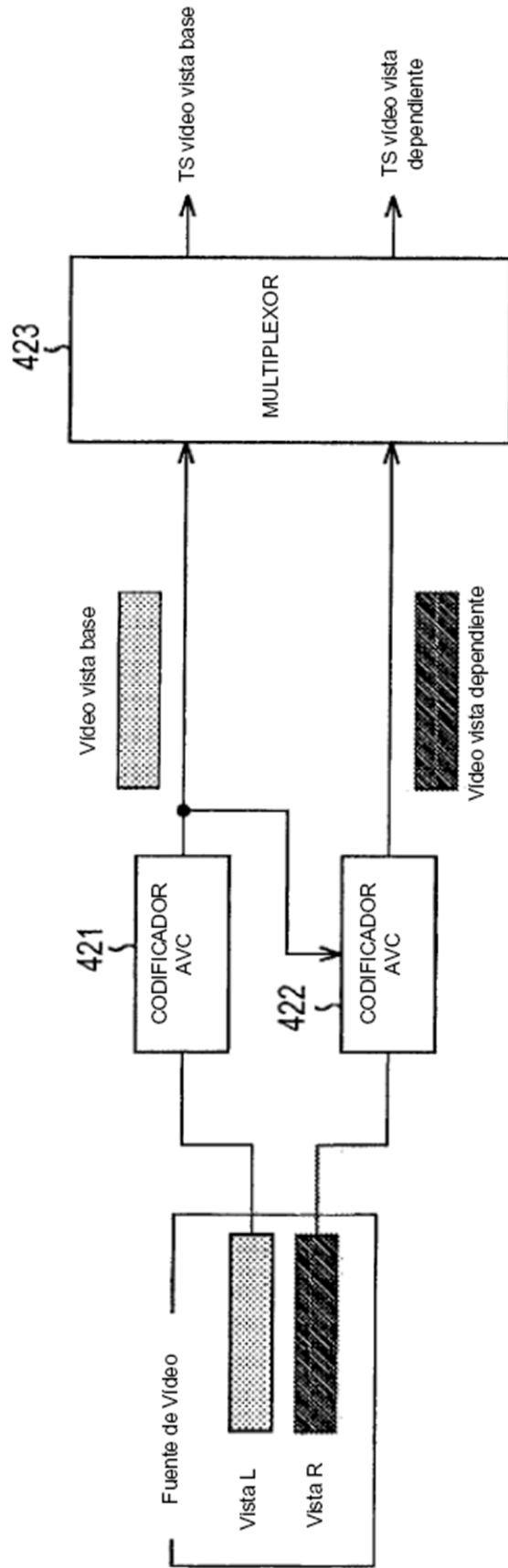


FIG. 34

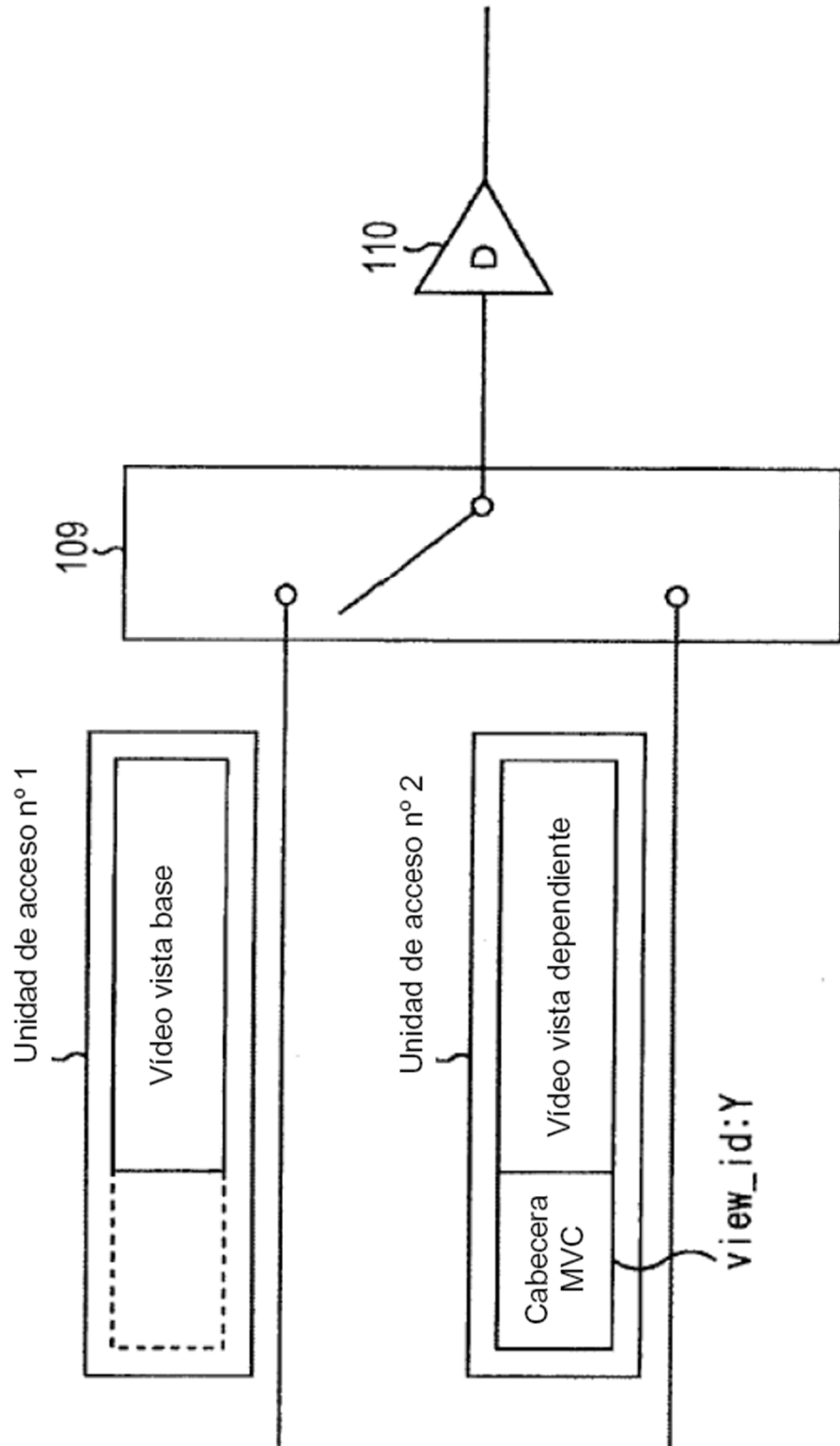


FIG. 35

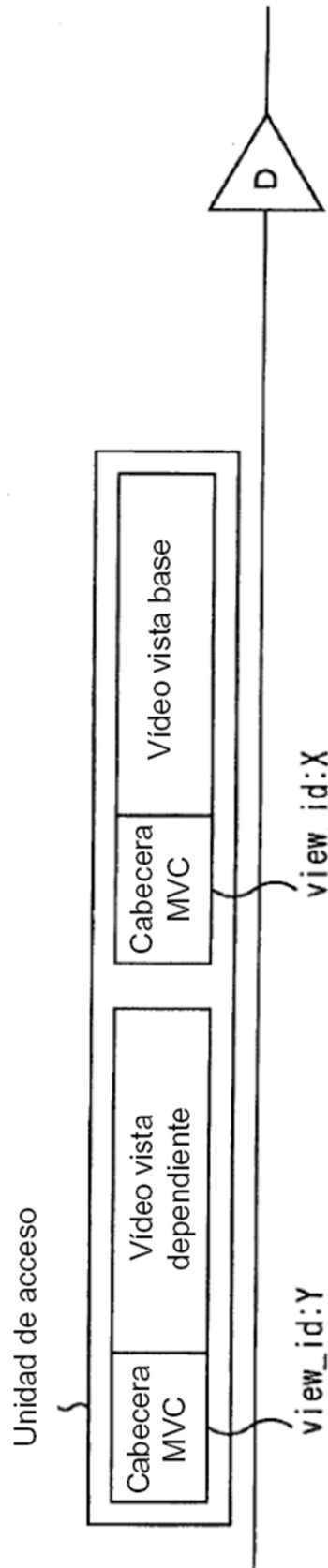


FIG. 36

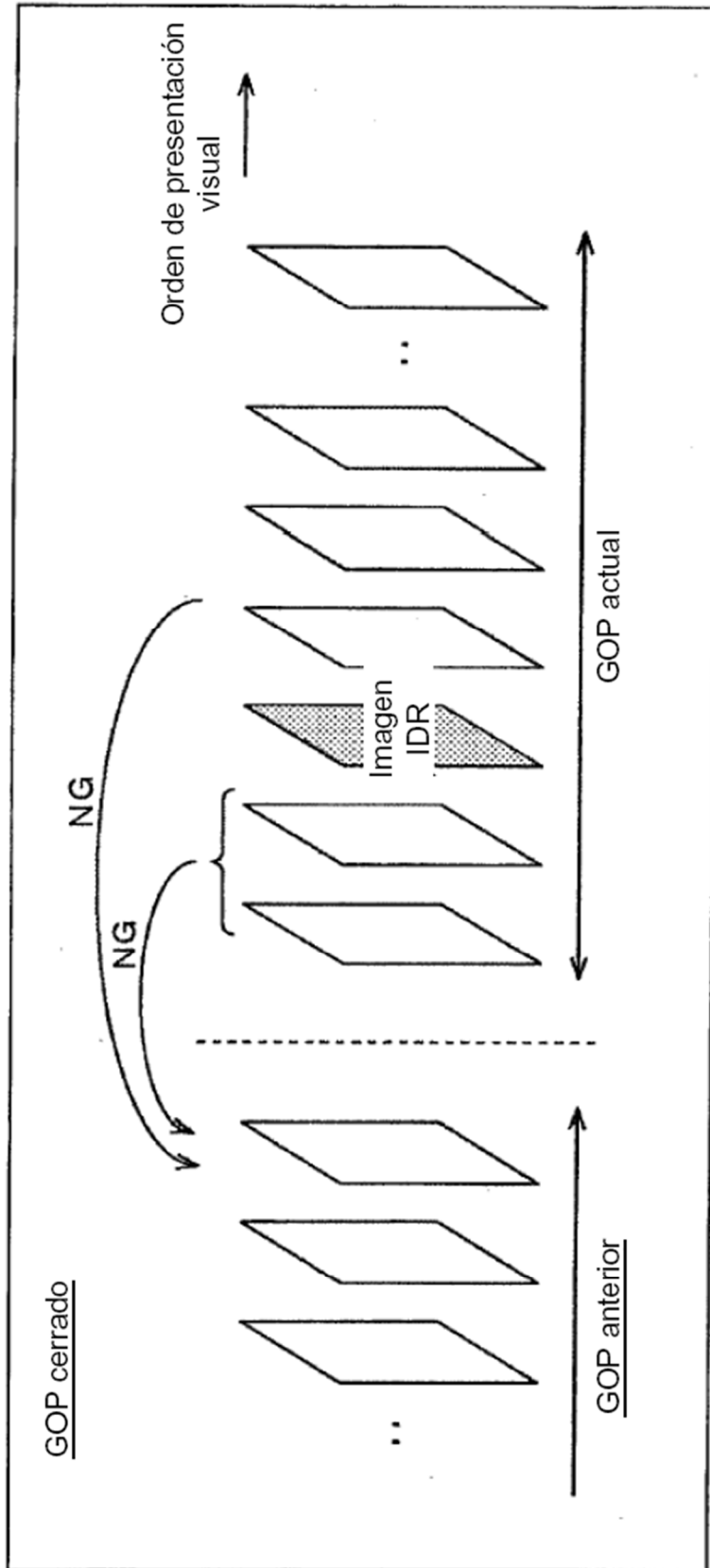


FIG. 37

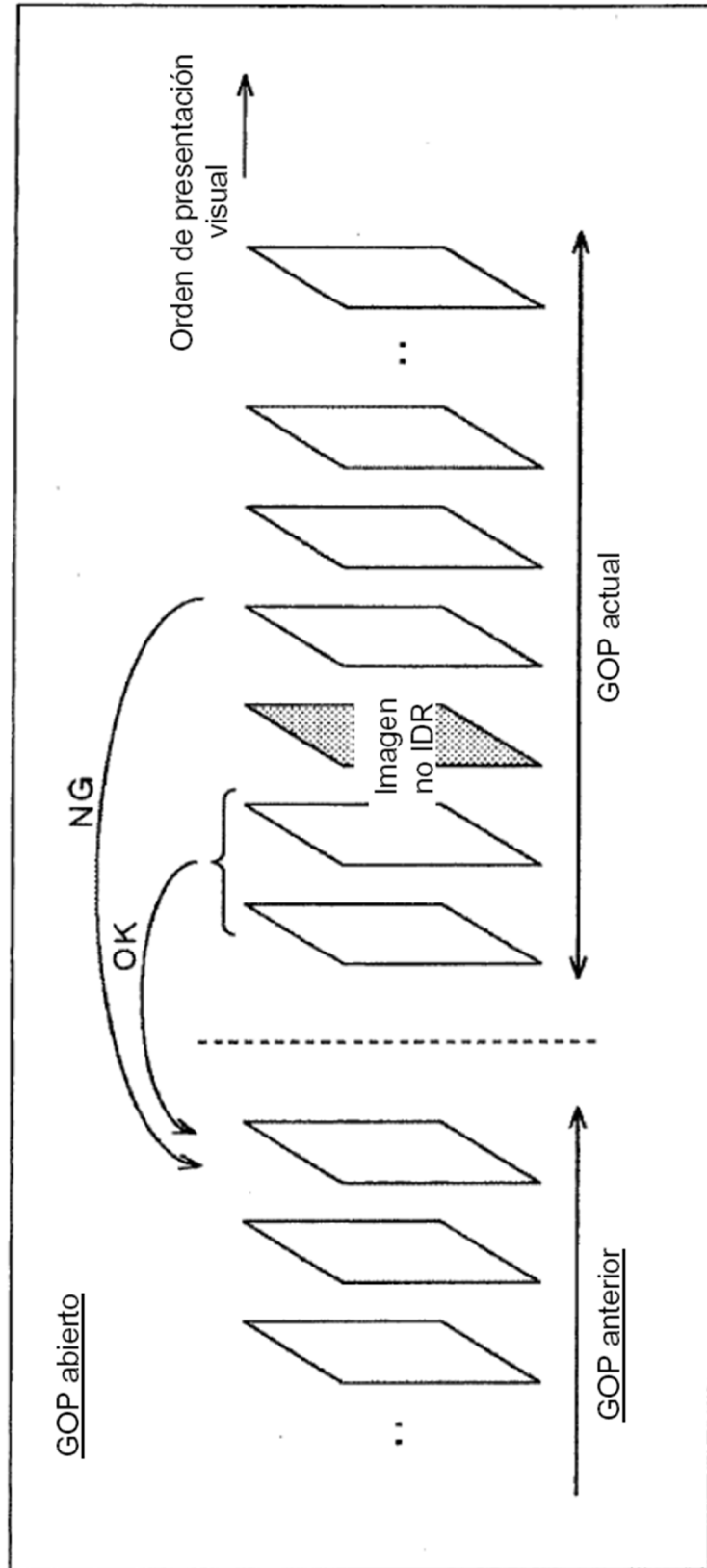


FIG. 38

Tasa de tramas progresiva / Entrelazado

NÚMERO MÁXIMO DE CAMPOS VISUALIZADOS EN 1 GOP		
29.97	Entrelazado	60
25	Entrelazado	50
NÚMERO MÁXIMO DE TRAMAS VISUALIZADAS EN 1 GOP		
59.94	Progresivo	60
50	Progresivo	50
23.976	Progresivo	24
24	Progresivo	24

FIG. 39

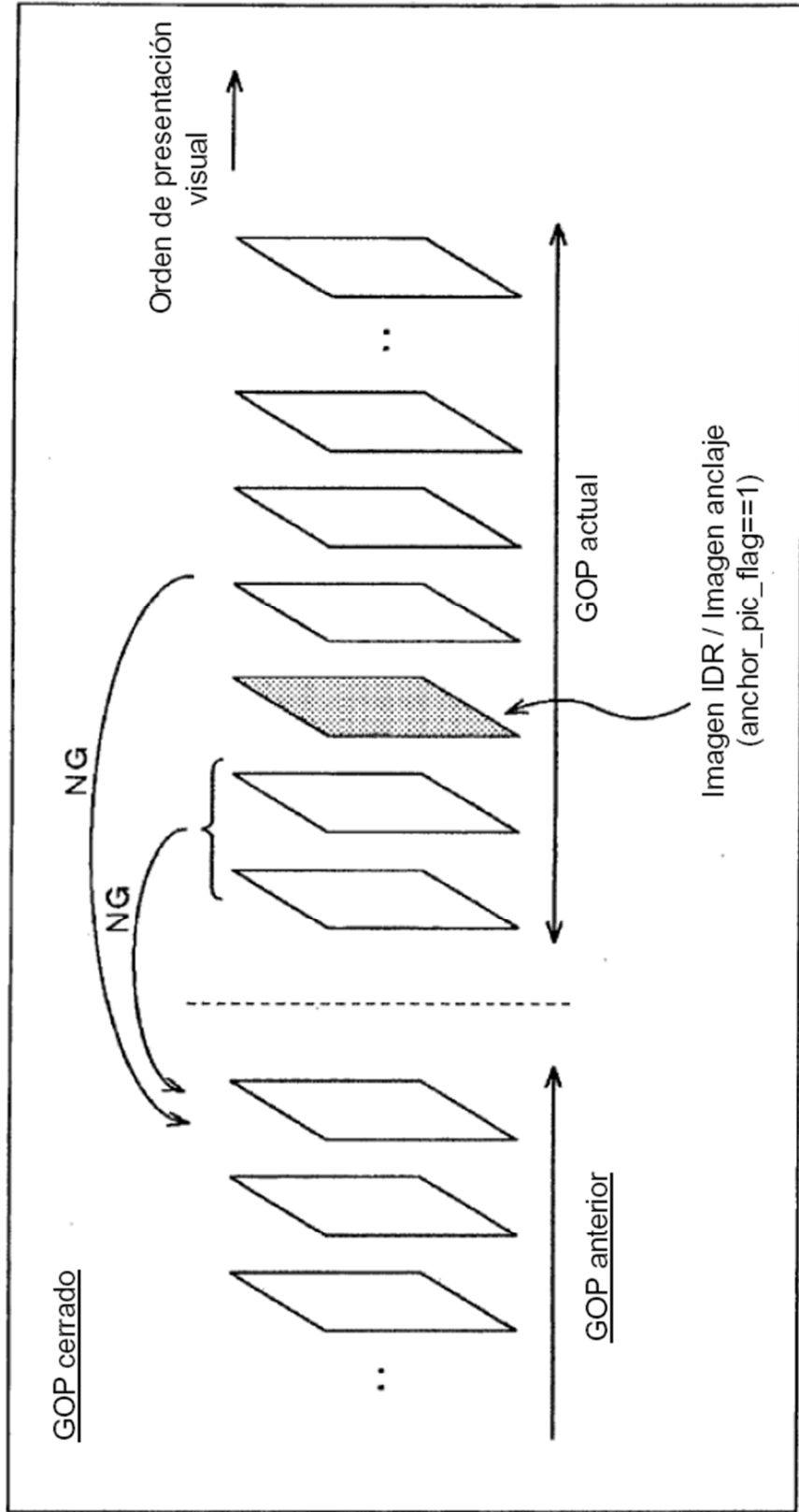


FIG. 40

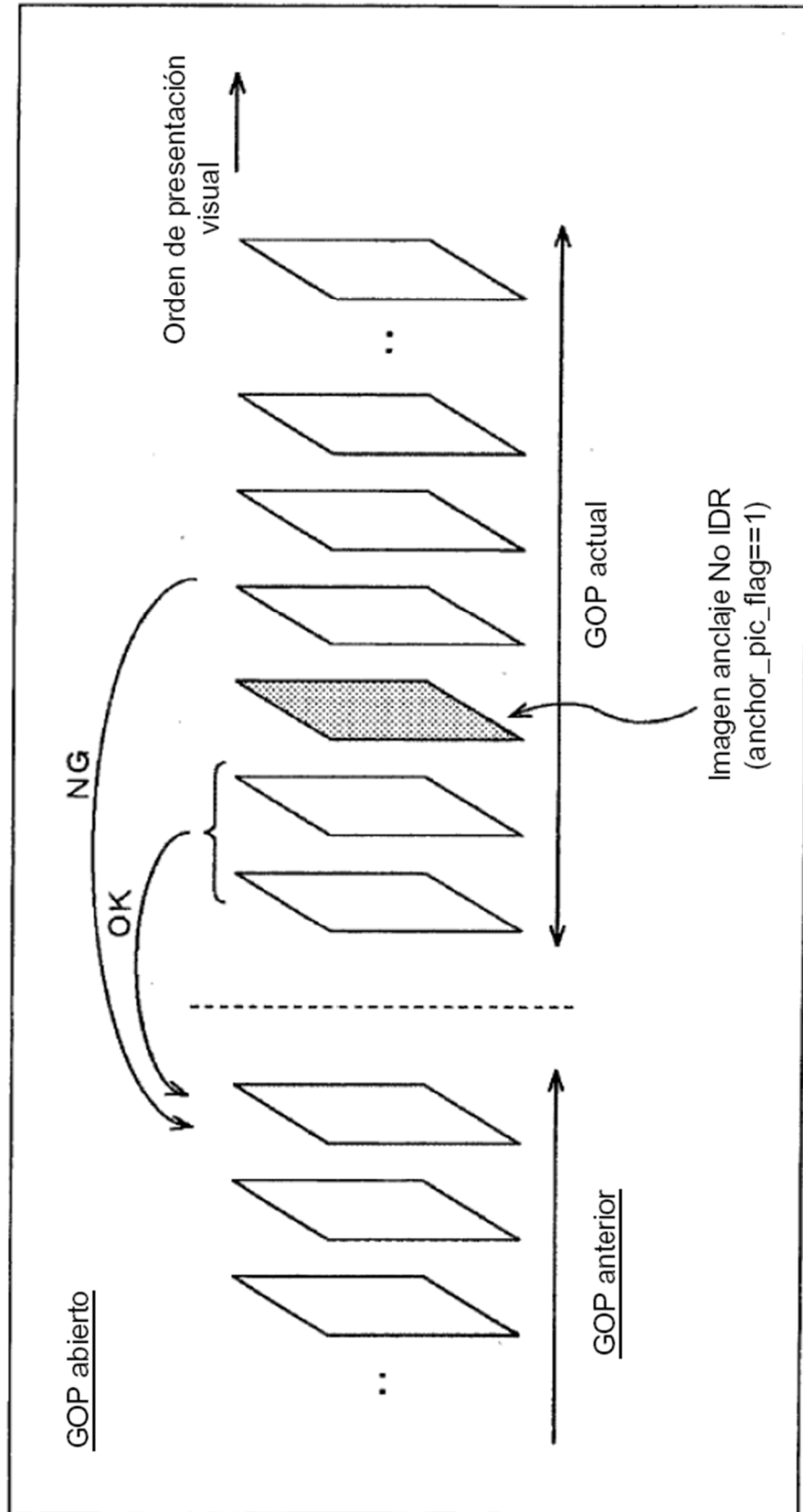


FIG. 41

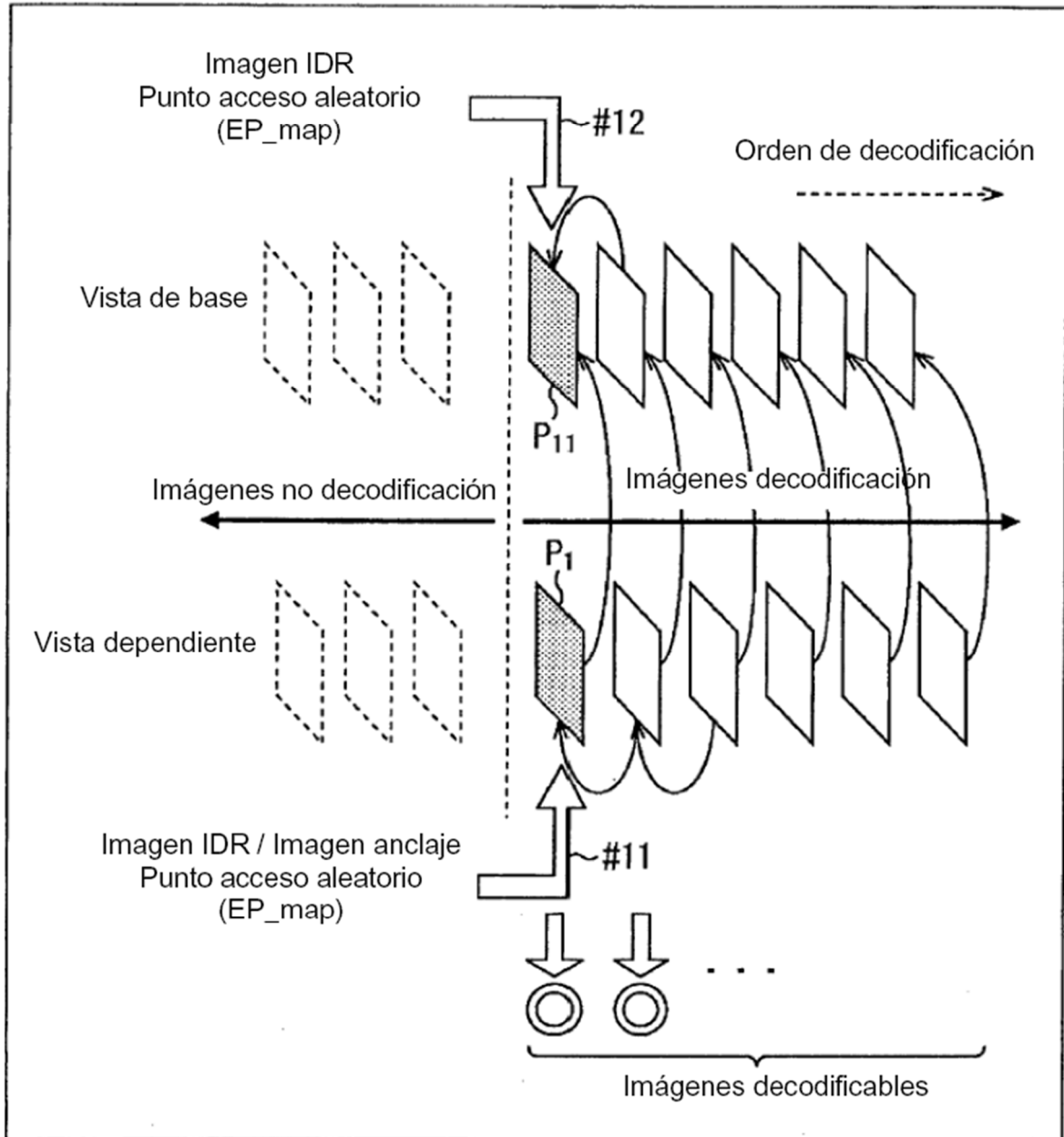


FIG. 42

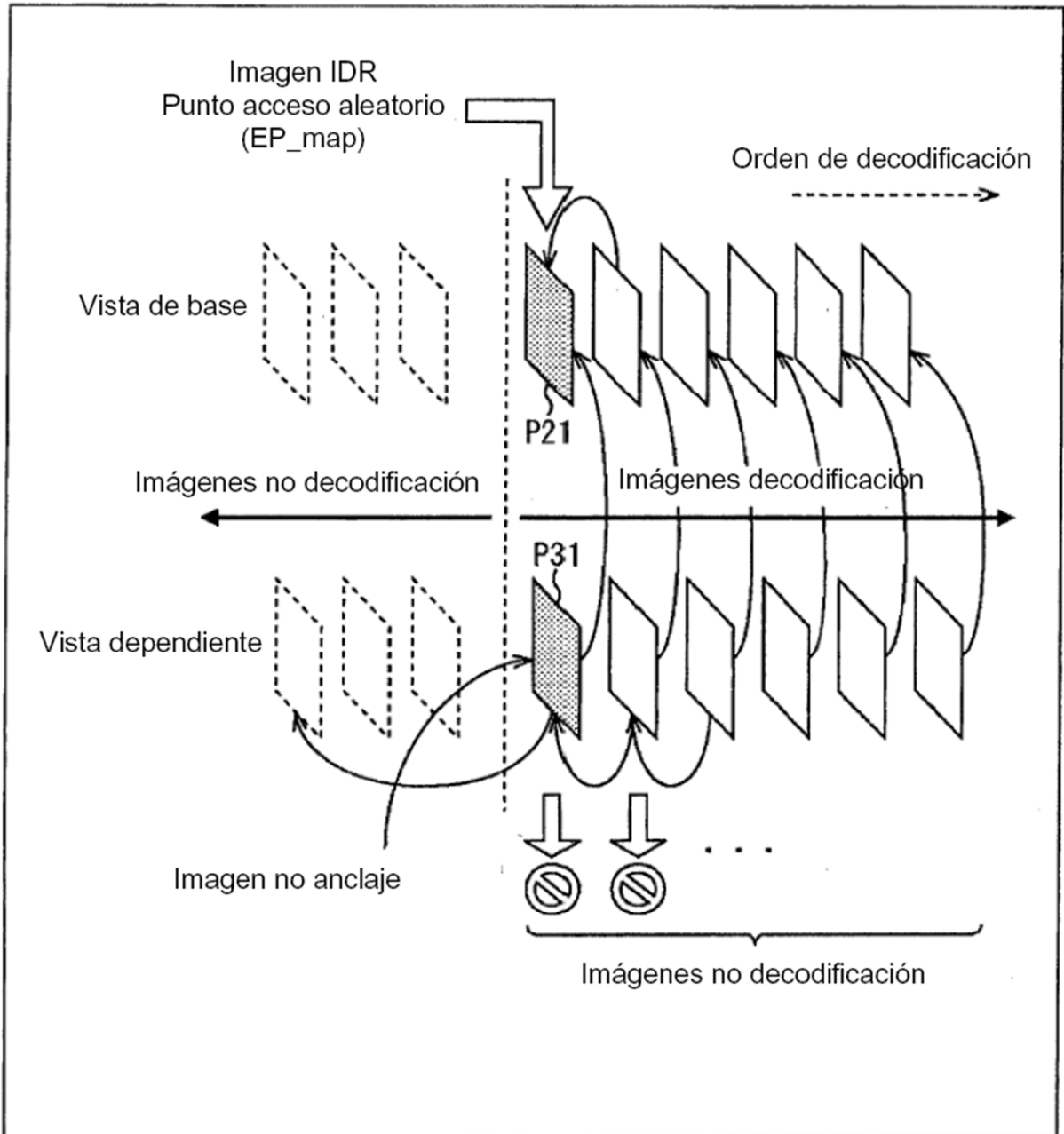


FIG. 43

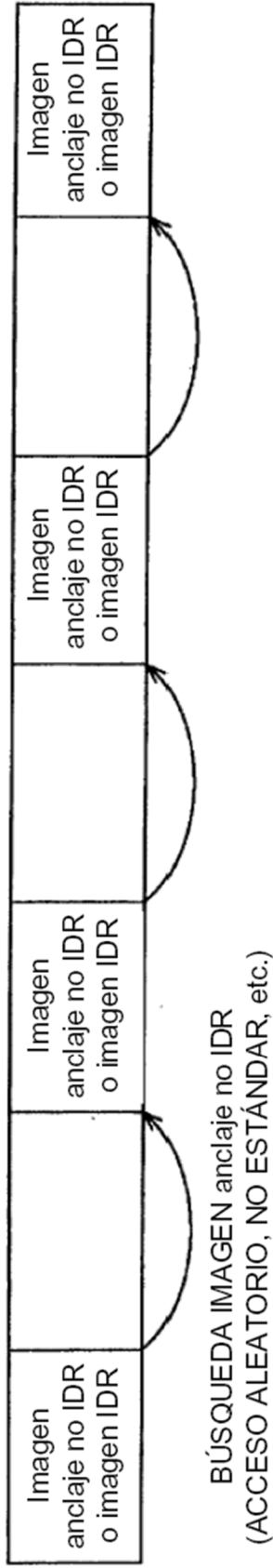


FIG. 44

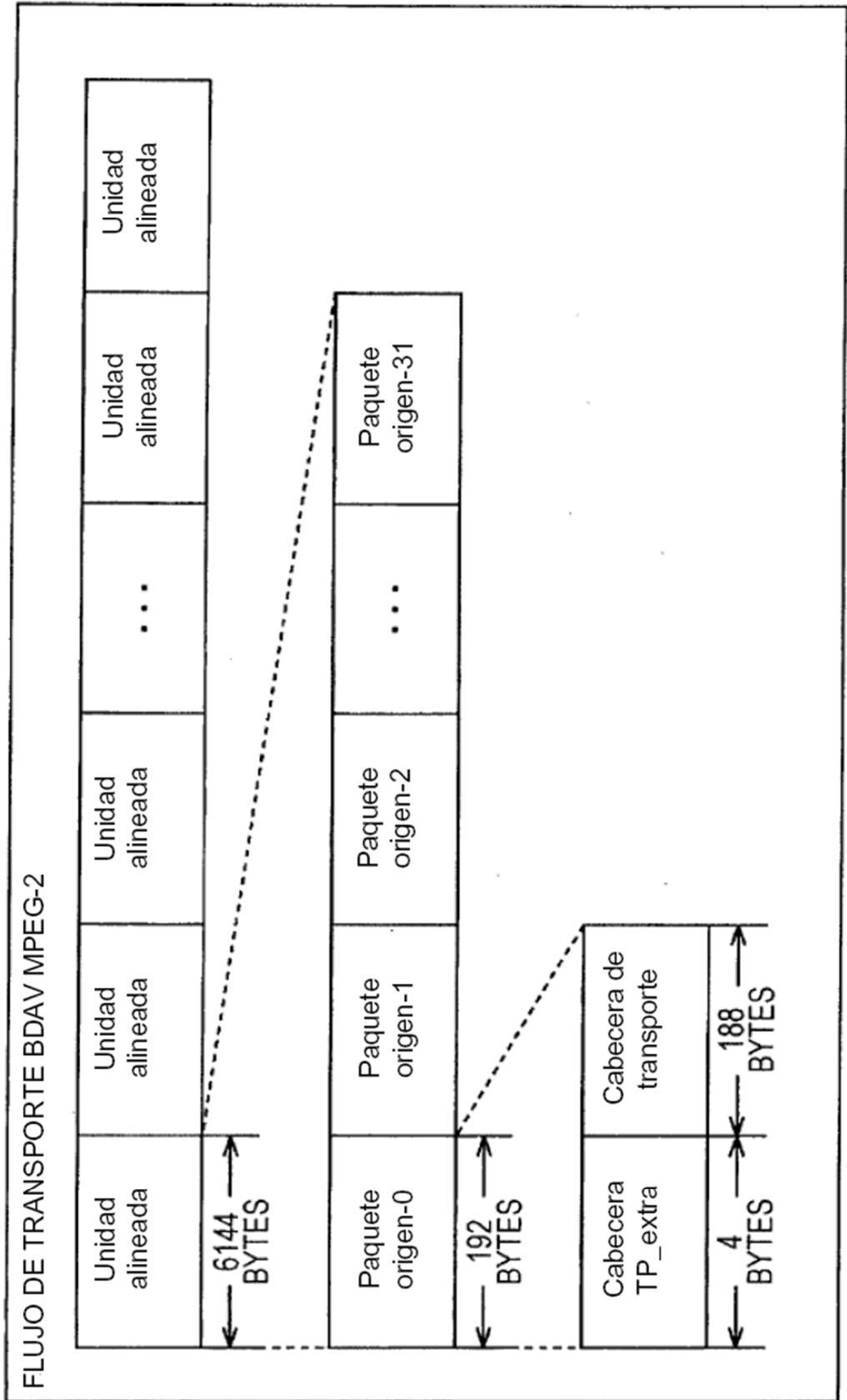


FIG. 45

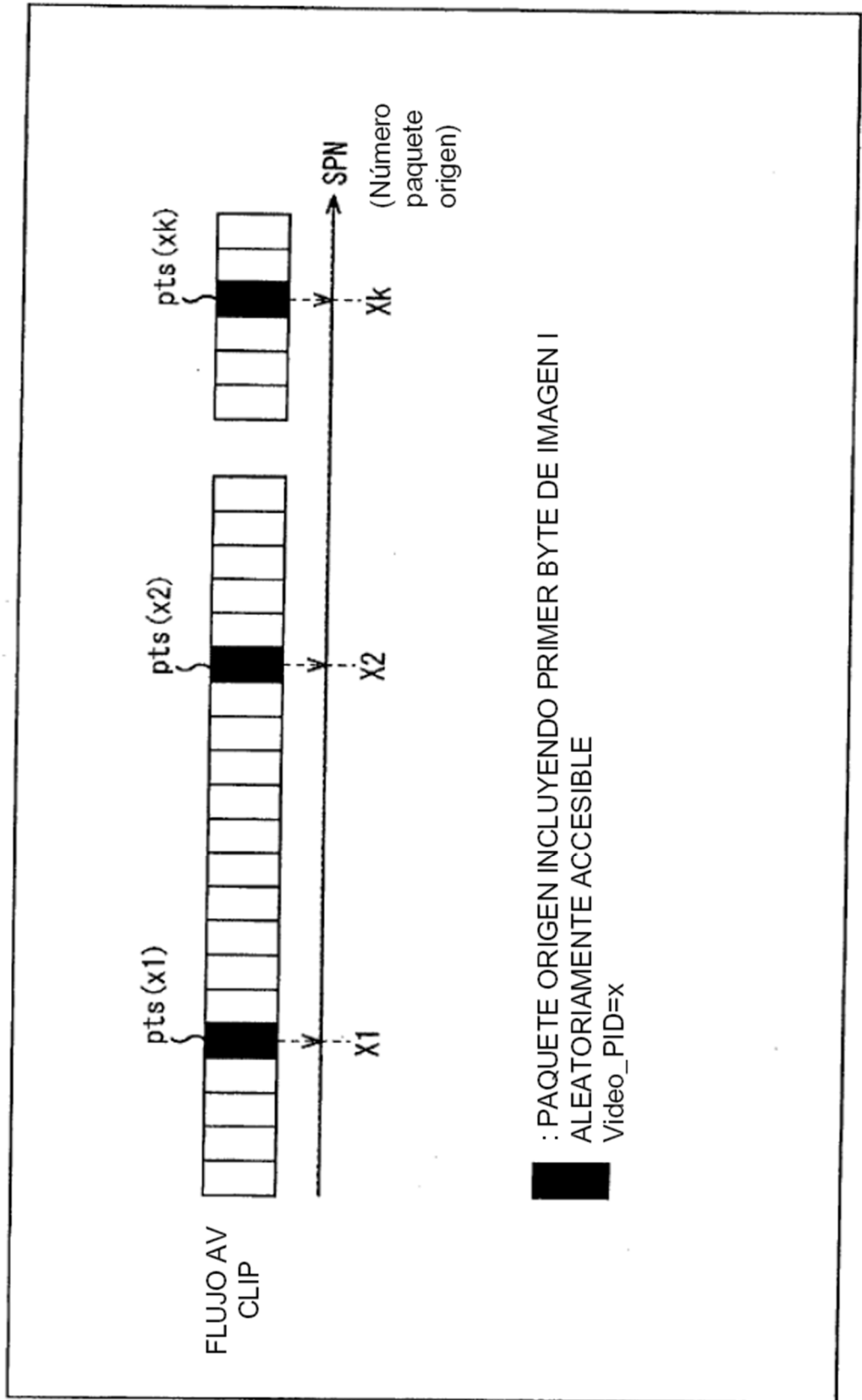


FIG. 46

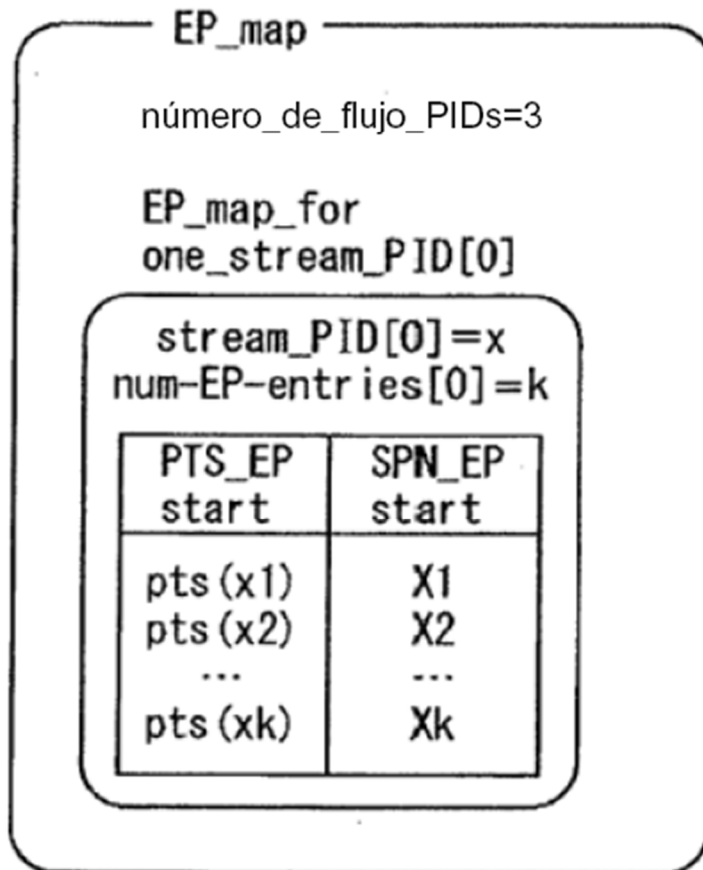


FIG. 47

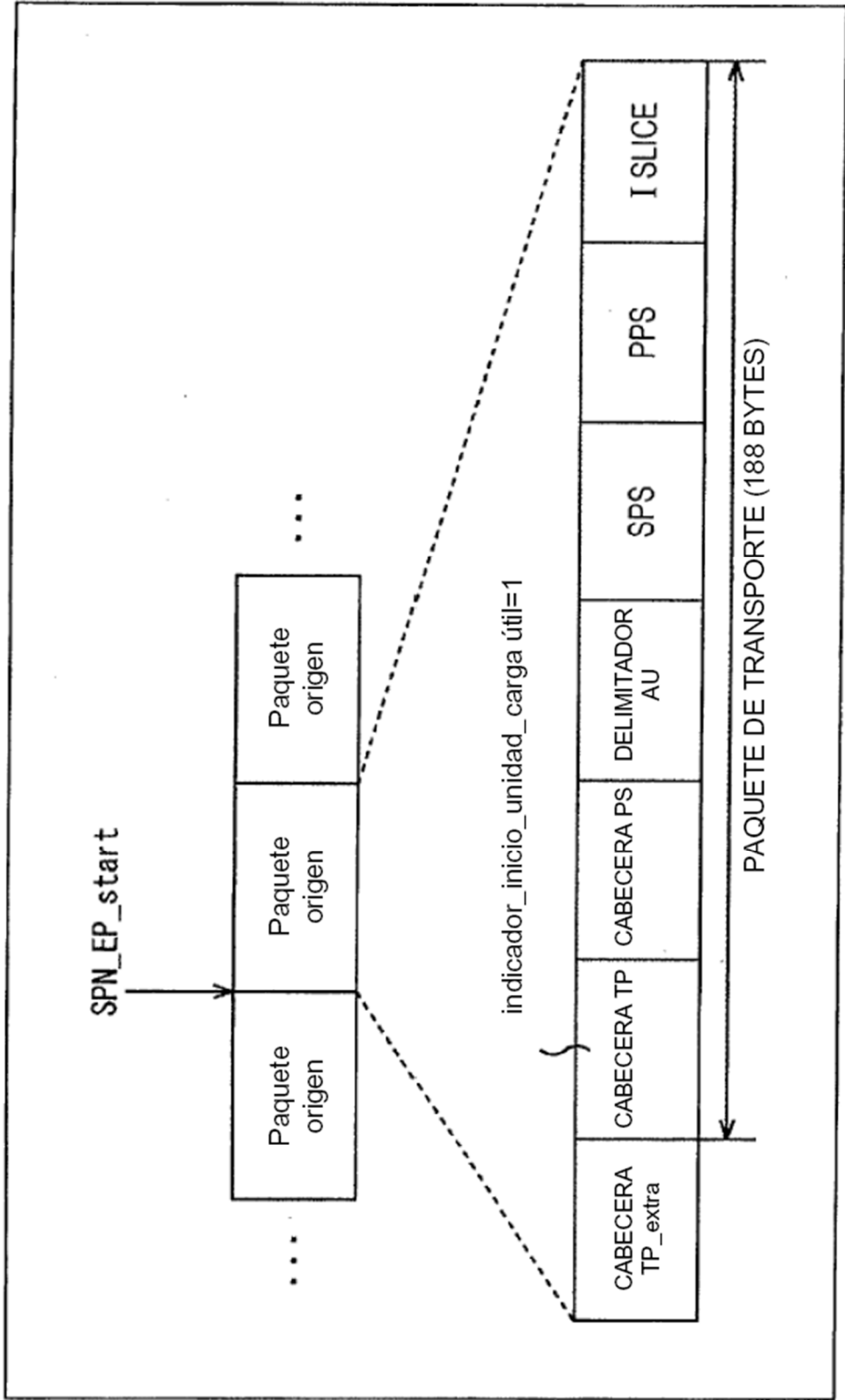


FIG. 48

