

(19) **DANMARK**



Patent- og
Varemærkestyrelsen

(12)

Oversættelse af europæisk patentskrift

(10) **DK/EP 2445224 T3**

-
- (51) Int.Cl.: *H 04 N 13/04 (2006.01)* *G 11 B 27/30 (2006.01)* *H 04 N 5/445 (2011.01)*
H 04 N 5/775 (2006.01) *H 04 N 5/85 (2006.01)* *H 04 N 5/92 (2006.01)*
H 04 N 7/025 (2006.01) *H 04 N 7/03 (2006.01)* *H 04 N 7/035 (2006.01)*
H 04 N 9/79 (2006.01) *H 04 N 9/804 (2006.01)* *H 04 N 9/806 (2006.01)*
H 04 N 9/82 (2006.01) *H 04 N 13/00 (2006.01)* *H 04 N 19/597 (2014.01)*
- (45) Oversættelsen bekendtgjort den: **2017-08-21**
- (80) Dato for Den Europæiske Patentmyndigheds bekendtgørelse om meddelelse af patentet: **2017-05-03**
- (86) Europæisk ansøgning nr.: **10789226.7**
- (86) Europæisk indleveringsdag: **2010-06-16**
- (87) Den europæiske ansøgnings publiceringsdag: **2012-04-25**
- (86) International ansøgning nr.: **JP2010003998**
- (87) Internationalt publikationsnr.: **WO2010146847**
- (30) Prioritet: **2009-06-17 JP 2009144568**
- (84) Designerede stater: **AL AT BE BG CH CY CZ DE DK EE ES FI FR GB GR HR HU IE IS IT LI LT LU LV MC MK MT NL NO PL PT RO SE SI SK SM TR**
- (73) Patenthaver: **Panasonic Intellectual Property Management Co., Ltd., 7 OBP Panasonic Tower , 1-61, Shiromi 2-chome , Chuo-ku, Osaka-shi, Osaka 540-6207, Japan**
- (72) Opfinder: **SASAKI, Taiji, c/o Panasonic Corp., IPROC, 7F Twin 21 OBP Panasonic Tower, 2-1-61, Shiromi, Chuo-ku, Osaka-shi, Osaka 540-6207, Japan**
YAHATA, Hiroshi, c/o Panasonic Corp., IPROC, 7F Twin 21 OBP Panasonic Tower, 2-1-61, Shiromi, Chuo-ku, Osaka-shi, Osaka 540-6207, Japan
OGAWA, Tomoki, c/o Panasonic Corp., IPROC, 7F Twin 21 OBP Panasonic Tower, 2-1- 61, Shiromi, Chuo-ku, Osaka-shi, Osaka 540-6207, Japan
- (74) Fuldmægtig i Danmark: **Dennemeyer & Associates S.A, P.O. Box 700425, DE-81304 Munich, Tyskland**
- (54) Benævnelse: **INFORMATIONSDATAOVERFØRINGSMEDIUM TIL GENGIVELSE AF 3D-VIDEO OG GENGIVELSESIKRETTELSE**
- (56) Fremdragne publikationer:
WO-A1-2004/095837
WO-A2-96/38981
WO-A2-2008/044191
JP-A- 2002 016 884
US-A1- 2005 259 147
US-A1- 2007 253 680

DESCRIPTION

[Technical Field]

[0001] The present invention relates to a technology of playing back 3D and 2D images.

[Background Art]

[0002] The 2D images, also called monoscopic images, are represented by pixels on an X-Y plane that is applied to the display screen of the display device. In contrast, the 3D images have a depth in the Z-axis direction in addition to the pixels on the X-Y plane applied to the screen of the display device.

[0003] The 3D images are presented to the viewers (users) by simultaneously playing back the left-view and right-view images to be viewed respectively by the left and right eyes so that a stereoscopic effect can be produced. The users would see, among the pixels constituting the 3D image, pixels having positive Z-axis coordinates in front of the display screen, and pixels having negative Z-axis coordinates behind the display screen.

[0004] It is preferable that an optical disc storing a 3D image has compatibility with a playback device that can play back only 2D images (hereinafter, such a playback device is referred to as "2D playback device"). This is because, otherwise, two types of discs for 3D and 2D images need to be produced so that the 2D playback device can play back the same content as that stored in a disc for 3D image. Such an arrangement will take a higher cost. It is accordingly necessary to provide an optical disc storing a 3D image that is played back as a 2D image by the 2D playback device, and as a 2D or 3D image by a playback device supporting both the 3D and 2D images (hereinafter, such a playback device is referred to as "2D/3D playback device").

[0005] Japanese Patent No. 3935507 is one example of prior art documents describing technologies for ensuring the compatibility in playback between 2D and 3D images, with respect to optical discs storing 3D images.

[0006] Document US 2005/259147 A1 describes an apparatus for playing back video content obtained from a recording medium. A display mode of a video stream is changed based on user characteristic information. In particular, a monoscopic image is converted to a stereoscopic image by shifting an image in the video played in the leftward and rightward directions in order to generate a parallax difference between a left-view-point image and a right-view-point image.

[0007] Document WO 96/38918 A2 describes a device that is capable of displaying an image

including a caption with an aspect ratio of 4:3 on a screen with an aspect ratio of 16:9. When an image with an aspect ratio of 4:3 is expanded in accordance with the horizontal width of a screen with an aspect ratio of 16:9, the vertical width of the image after expansion is longer than the vertical width of the screen, and as a result a caption at a lower end of the image is not displayed within the screen. Therefore, the caption is shifted in the upward direction so as to be displayed within the screen.

[0008] Document WO 2008/044191 A2 describes a system for generating and playing back an AV content including a video stream and a graphics stream. The graphics stream may, in particular, correspond to subtitles. A subtitle for stereoscopic viewing is generated by splitting the graphics stream into a two dimensional image and a depth map indicating window information for copying the right-view image and the left-view image, respectively. The two pieces of window information are shifted in the video plane in leftward and rightward directions, which are equal to the parallax direction.

[Summary of Invention]

[Technical Problem]

[0009] By the way, in the case of a movie work or the like, subtitle data is stored in an optical disc subtitle. Generally, when such a movie work is played back, subtitles are overlaid with videos for display. Here, if long scripts or narrations are included in the movie work, most of a screen is occupied with display regions for subtitle characters. If playback of a video with a high level of jump-out is performed in the state where most of the screen is occupied with the display regions for subtitle characters, subtitles overlap stereoscopic display of the video. This results in playback of a stereoscopic image that is extremely difficult to view. There is a method of moving the arrangement position of the subtitle characters an end of the screen such that the subtitles does not overlap stereoscopic display of the video. However, a stereoscopic effect of a video greatly differs depending on a playback position in a plurality of playback sections in the time axis of a video stream. Also, subtitle characters often differ in character amount depending on the type of language. Accordingly, if any one of the ends of the screen is uniformly fixed as a subtitle display region, the use efficiency of the screen is deteriorated. As a result, even if a user spends his money to purchase an expensive widescreen 3D TV, there might occur a case where the user cannot enjoy the stereoscopic effect to the full.

[0010] The present invention provides a recording medium capable of avoiding degradation of a stereoscopic effect due to decrease of the use efficiency of a screen.

[Solution to Problem]

[0011] This is achieved by the features of claim 1.

[Advantageous Effects of Invention]

[0012] The additional information including a region-saving flag defining a display region of a subtitle is included in a stream selection table for each playback section in correspondence with a stream number. When the playback section changes, or when a request for changing the stream is received, a stream selection procedure is executed. A stream number in accordance with the language settings of a playback device is set in a stream number register. As a result, a region-saving flag indicated by a piece of additional information corresponding to the set stream number is provided with the playback device. With this structure, it is possible to realize control in which a display region of a subtitle is saved in the upper end of the screen in a playback section and a display region of a subtitle is saved in the lower end of the screen in another playback section.

[0013] The cinema scope size (1:2.35) is generally used for the aspect ratio of video of movies. In the case where a video is stored in an optical disc such as a BD-ROM, a main feature video is disposed in the center of an HD video having the aspect ratio of 16:9 without changing the aspect ratio, and a black frame is inserted into each of the upper side and the lower side of the HD video. Accordingly, with the above structure, it is possible to display subtitles in a large subtitle display region generated by collecting black frames located above and below the main feature video to one of the upper end and the lower end of the video plane. This can improve the use efficiency of the screen, thereby improving the stereoscopic effect.

(Addittional Technical Problem)

[0014] According to 3D videos in which the stereoscopic effect is realized using the parallax difference between main-view image and a sub-view image, the parallax difference differs depending on the screen size of display device. This causes difference in the depth of images depending on the screen size. As a result, if a 3D video created for viewing in a display device with a large screen is viewed in a display device with a small screen, the 3D video is not powerful and less width is displayed in such a display device than a creator of the 3D video has expected. On the other hand, a 3D video created for viewing in a display device with a small screen is viewed in a display device with a large screen, an excessive sense is given to the 3D video and this causes a viewer to suffer from eye strain.

[0015] The present invention aims to provide a recording medium that is capable of preventing occurrence of negative influence exerted by viewing of a 3D video in a display device with a screen whose size is different from that assumed in the creation of the 3D video.

[0016] A recording medium that can solve the above problem is a recording medium having recorded thereon a main-view video stream, a sub-view video stream, and meta data, wherein the main-view video stream includes main-view picture data constituting a main-view of a stereoscopic video, the sub-view video stream includes sub-view picture data constituting a sub-view of the stereoscopic video, the meta data includes offset correction values each corresponding to screen size information of each of a plurality of display devices, and the offset correction value defines an offset for shifting, in a leftward direction or a rightward direction of a horizontal coordinate, at least one of a main-view video plane in which the main-view picture data is to be rendered and a sub-view video plane in which the sub-view picture data is to be rendered.

[0017] By giving, to picture data, an offset defined in screen size information for each display screen size so as to shift the video plane, it is possible to give an appropriate parallax difference to a stereoscopic image depending on each screen size. As a result, with the above structure, it is possible to prevent occurrence of negative influence exerted by viewing of a 3D video in a display device with a screen whose size is different from that assumed in the creation of the 3D video.

[Brief Description of Drawings]

[0018]

FIGs. 1A through 1C show a home theater system that is composed of a recording medium being a package medium, a playback device being a player device, a display device, and glasses.

FIG 2 shows the user's head on the left side of the drawing and the images of a dinosaur skeleton seen respectively by the left eye and the right eye of the user on the right side of the drawing.

FIG. 3 shows one example of the internal structures of the base-view and dependent-view video streams for the stereoscopic viewing.

FIG. 4 shows the concept of collecting black frames that are not used for a main feature video and displaying subtitle data on the black frames;

FIGs. 5A through 5C show the internal structure of the recording medium in Embodiment 1.

FIGs. 6A and 6B show the internal structures of the main TS and sub-TS.

FIGs. 7A through 7D show the internal structures of the playlist information.

FIGs. 8A and 8B show one example of the basic stream selection table.

FIG. 9 shows the internal structure of the extension stream selection table.

FIGs. 10A through 10C show stream registration sequences in the extension stream selection table.

FIG. 11 shows plane overlaying in the case where video_shift_mode is set as "Keep".

FIG. 12A shows plane overlaying in the case where video_shift_mode is set as "Up", and FIG. 12B shows plane overlaying in the case where video_shift_mode is set as "Down".

FIG. 13 shows a constraint of the order of registering graphics streams in a stream selection table in the case where the video shift mode is added to the stream additional information of the stream selection information;

FIG. 14 shows what elementary streams are demultiplexed from the main TS and the sub-TSs with use of the basic stream selection table and the extension stream selection table.

FIG. 15 shows stream numbers to be assigned in the 2D output mode and the 3D output mode.

FIG 16 shows the internal structure of the playback device.

FIGs. 17A and 17B show the internal structure of the PG decoder.

FIGs. 18A and 18B show the internal structure of the text subtitle decoder.

FIGs. 19A and 19B show decoder models of the IG decoder.

FIG. 20 shows a circuit structure for overlaying the outputs of the decoder models and outputting the result in the 3D-LR mode.

FIG. 21 shows a circuit structure for overlaying the outputs of the decoder models and outputting the result in the 1 plane + offset mode.

FIG. 22 shows the circuit structure for overlaying data output from the decoder model and outputting the overlaid data in the upper end 2D subtitle playback mode and the lower end 2D subtitle playback mode.

FIG. 23 shows the internal structures of the register set 203 and the playback control unit.

FIG. 24 shows the bit assignment in PSR24.

FIGs. 25A and 25B show the bit assignment in PSR32.

FIG. 26 shows the playlist playback procedure.

FIG. 27 is a flow chart showing the procedure for determining the current PG_text subtitle stream when playback condition is changed.

FIG. 28 is a flow chart showing a procedure of determination processing of upper or lower end playback type.

FIG. 29 is a flow chart showing the procedure for selecting a PG_text subtitle stream that is optimum for the current playitem.

FIG 30 is a flow chart showing the procedure which is to be executed when a stream change is requested by the set stream stereoscopic command (set stream SS command).

FIG. 31 is a flow chart showing the procedure which is to be executed when a stream change is requested by the set stream command or by a user operation requesting a stream number change.

FIGs. 32A and 32B are flow charts showing the procedures for determining the current IG stream and the playback type thereof.

FIGs. 33A through 33C show what packet identifiers are output to the demultiplexing unit by the combined stream registration sequence.

FIGs. 34A through 34C show what packet identifiers are output to the demultiplexing unit by the combined stream registration sequence.

FIGs. 35A through 35C show the stream registration sequences in an extension stream selection table according to a modification example of Embodiment 1.

FIG. 36 shows the circuit structure for overlaying data output from the decoder model and outputting the overlaid data in the upper end 2D subtitle playback mode and the lower end 2D subtitle playback mode.

FIG. 37 shows a system parameter showing a shift amount of each plane in the longitudinal axis direction;

FIG. 38 shows a method of shifting and cropping a PG plane in accordance with a video shift mode;

FIG. 39 shows a constraint condition for disposing subtitle data in a region that is not cropped in plane overlaying in the upper end 2D subtitle playback mode and the lower end 2D subtitle playback mode;

FIG. 40 shows functions of an output offset correction value for the sense of depth influenced by each screen size of TV.

FIG. 41 shows a table in which inch types of TV to be stored in a playlist file and output offset correction values are recorded;

FIG. 42 shows an example where images are displayed on a TV having the size larger than the optimal inch size.

FIG. 43 shows the structure of a 2D/3D playback device for applying an output offset correction value;

FIG. 44 shows the structure in which an output offset correction value and an output offset

correction value α are applied;

FIG. 45 shows the correspondence between the file 2D/file base and the file dependent.

FIGs. 46A through 46C show the correspondence between the interleaved stream file and file 2D/file base.

FIG. 47 shows correspondence among the stereoscopic interleaved stream file, file 2D, file base, and file dependent.

FIG. 48 shows the 2D playlist and 3D playlist.

FIGs. 49A through 49D show the internal structure of the clip information file.

FIG. 50 shows the correspondence among the clip information file, playlist, and stereoscopic interleaved stream file.

FIGs. 51A and 51B show the internal structure of the clip base information and the clip dependent information.

FIG. 52 shows the basic entry map and the extension entry map.

FIG. 53 shows entries that are not permitted in the extension entry map.

FIG. 54 is a flow chart showing the playitem playback procedure.

FIG. 55 shows how the ATC sequence is restored from the data blocks constituting the stereoscopic interleaved stream file.

FIGs. 56A and 56B show how the ATC sequence is restored.

FIGs. 57A through 57D show one example of the extent start point information table in the base-view clip information and one example of the extent start point information table in the dependent-view clip information.

FIGs. 58A through 58C are illustrations provided for explanation of the source packet numbers of arbitrary data blocks in ATC sequences 1 and 2.

FIG. 59 shows the procedure for restoring the ATC sequence.

FIG. 60 shows a playback environment for a 2D/3D playback device;

FIG. 61 shows a case where only one of a right-eye video and a left-eye video is output during switching from playback 3D videos to 2D videos without switching a frame rate;

FIG. 62 shows the correlation between subtitles and streams for menu that are used in BD or the like;

FIG. 63 shows processing for realizing more smooth 2D/3D video display;

FIGs. 64A and 64B show a manufacturing method of an optical disc.

FIG. 65 is a flow chart showing the procedure of the authoring step.

FIG. 66 is a flow chart showing the procedure for writing the AV file.

FIG. 67 is a flow chart showing the procedure for generating the basic entry map and the extension entry map.

FIG. 68 is a flow chart showing the procedure for generating the BD-J application, BD-J object, movie object, and index table.

FIG. 69 shows an internal structure of a multi-layered optical disc.

FIG. 70 shows the application format of the optical disc based on the file system.

FIG. 71 shows the structure of a 2D/3D playback device.

FIGs. 72A through 72C show the embodiment of a usage act of a recording medium relating to the present invention, the structure of the BD-ROM, and the structure of the index file.

FIGs. 73A and 73B show the structure of an AV clip and how each stream is multiplexed in the AV clip.

FIGs. 74A and 74B illustrate in detail how the video stream is stored in the PES packet series, and show the TS packets and source packets in the AV clip.

FIGs. 75A and 75B show the data structure of the PMT and the internal structure of the clip information file.

FIGs. 76A and 76B show the internal structure of the stream attribute information and the internal structure of the entry map.

FIGs. 77A through 77C show the internal structure of the playlist and the internal structure of the playitem.

FIGs. 78A and 78B show the structure of a 2D playback device and explain the player variable.

FIG. 79 shows the internal structure of the system target decoder.

FIG. 80 illustrates the stereoscopic viewing.

FIG. 81 shows the data structure of a presentation graphics stream;

FIG. 82 shows decode processing of the presentation graphics stream;

FIG. 83 shows a method of storing a playlist of a shift value in video shift upward and a shift value in video shift downward;

FIG. 84 shows the structure of plane overlaying performed by the 2D/3D playback device for performing video shift of collecting black frames in one of an upper side and a lower side;

FIG. 85 shows the structure of a playlist in which the video shift mode is added to stream additional information of stream selection information;

FIG. 86 shows a plane overlaying method in the case where the video shift mode has been added to the stream additional information of the stream selection information;

FIG. 87 shows, in the upper level, a method of creating a video stream by disposing a main feature video not on the center but on a slightly upper side and, in the lower level, a method of creating a black frame by dynamically changing a transparent color of a PG stream;

FIGs. 88A and 88B show the structure in which each extent includes at least one entry point.

FIGs. 89A and 89B show a method of storing offset metadata in an AV stream information file.

FIGs. 90A and 90B show a method of storing offset metadata for each entry point.

FIGs. 91A and 91B show a method of storing offset metadata in a playlist.

FIGs. 92A and 92B show, in the case where offset metadata is stored in a playlist, a method of not storing offset metadata when a current playitem is the same as a previous playitem.

FIG. 93 shows, in the case where offset metadata is stored in a playlist, a method of storing only the same one piece of offset metadata with respect to a plurality of playitems having the same one piece of offset metadata.

FIG. 94 shows a playlist in which a header in units of playitems and offset metadata are separately stored.

FIG. 95 shows a case where a left-eye graphics subtitle in the 2 plane L/R method is displayed as a 2D display subtitle.

FIG. 96 shows the 2D display subtitle and the 1 plane + offset method subtitle, and an offset value of PG for the 2 plane LR method for sharing the left-eye PG in the 2 plane LR method.

FIG. 97 shows the structure of separating a 2D/3D playback path in order to increase in speed of jump playback;

FIG. 98 shows an example of an index file (Index.bdmv) stored in a BD-ROM for playing back stereoscopic images.

FIG. 99 is a flow chart showing switching between the playback of the 2D PlayList and the 3D PlayList of a program of a BD program file;

FIG. 100 shows an example structure of a 2D/3D playback device which is realized by using an integrated circuit.

FIG. 101 is a functional block diagram showing a typical structure of the stream processing unit.

FIG. 102 is a conceptual diagram showing the switching unit and the peripheral when the

switching unit is DMAC.

FIG. 103 is a functional block diagram showing a typical structure of the AV output unit.

FIG. 104 is an example structure showing the AV output unit, or the data output part of the playback device in more detail.

FIG. 105 shows arrangement of control buses and data buses in the integrated circuit.

FIG. 106 shows arrangement of control buses and data buses in the integrated circuit.

FIG. 107 shows an example structure of a display device which is realized by using an integrated circuit.

FIG. 108 is a functional block diagram showing a typical structure of an AV output unit of the display device.

FIG. 109 is a flow chart showing an operation procedure in the playback device.

FIG. 110 is a flow chart showing a detailed operation procedure in the playback device.

[Description of Embodiments]

[0019] The recording media provided with the solution to problem described above may be implemented as package media containing contents for sale on a store. Also, playback devices supporting the recording media may be implemented as player devices for playing back the package media, and integrated circuits supporting the recording media may be implemented as system LSIs to be embedded in the player devices.

[0020] FIGs. 1A through 1C show a home theater system that is composed of a recording medium being a package medium, a playback device being a player device, a display device, and glasses. As shown in FIG 1A, a recording medium 100 being a package medium as described above and a playback device 200 being a player device constitute the home theater system together with a display device 300, 3D glasses 400, and a remote control 500. The home theater system structured as such is subject to use by the user.

[0021] The recording medium 100 provides the home theater system with, for example, a movie work. The movie work may provide a stereoscopic image. Here, the stereoscopic image is composed of at least two view-point images. The view-point image is an image that is deflected to some extent, and the at least two view-point images include a main-view image and a sub-view image. As shown in FIG. 1A, the recording medium 100 may be, for example, a disc or a memory card among many types of recording media. In the following, a "recording medium" is presumed to be a disc unless otherwise noted.

[0022] The playback device 200 is connected with the display device 300 and plays back the recording medium 100. The playback device described in the present application is a 2D/3D playback device (player) which, provided with the 2D output mode and the 3D output mode, can switch between these output modes to play back a main-view video stream representing a main-view image and a sub-view video stream representing a sub-view image.

[0023] The display device 300 is a television and provides the user with an interactive operation environment by displaying a menu and the like as well as images of movie works. In the present embodiment, the user needs to wear the 3D glasses 400 for the display device 300 to realize the stereoscopic viewing. Here, the 3D glasses 400 are not necessary when the display device 300 displays images by the lenticular method.

[0024] The 3D glasses 400 are equipped with liquid-crystal shutters that enable the user to view a parallax image by the sequential segregation method or the polarization glasses method. Here, the parallax image is an image which is composed of a pair of (i) an image that enters only into the right eye and (ii) an image that enters only into the left eye, such that pictures respectively associated with the right and left eyes respectively enter the eyes of the user, thereby realizing the stereoscopic viewing. FIG. 1B shows the state of the 3D glasses 400 when the left-view image is displayed. At the instant when the left-view image is displayed on the screen, the liquid-crystal shutter for the left eye is in the light transmission state, and the liquid-crystal shutter for the right eye is in the light block state. FIG. 1C shows the state of the 3D glasses 400 when the right-view image is displayed. At the instant when the right-view image is displayed on the screen, the liquid-crystal shutter for the right eye is in the light transmission state, and the liquid-crystal shutter for the left eye is in the light block state.

[0025] The remote control 500 is a machine for receiving operations for playing back AV from the user. The remote control 500 is also a machine for receiving operations onto the layered GUI from the user. To receive the operations, the remote control 500 is equipped with a menu key, arrow keys, an enter key, a return key, and numeral keys, where the menu key is used to call a menu constituting the GUI, the arrow keys are used to move a focus among GUI components constituting the menu, the enter key is used to perform ENTER (determination) operation onto a GUI component constituting the menu, the return key or numeric keys are used to return to a higher layer in the layered menu.

[0026] In the home theater system shown in FIGs. 1A through 1C, an output mode of the playback device for causing the display device 300 to display images in the 3D output mode is called a "3D output mode", and an output mode of the playback device for causing the display device 300 to display images in the 2D output mode is called a "2D output mode".

[0027] This completes the description of the usage act of the recording medium and the playback device.

(Embodiment 1)

[0028] Embodiment 1 is characterized in that a register in the playback device stores information that indicates whether or not the playback device has a capability to realize a stereoscopic viewing using a right-eye graphics stream and a left-eye graphics stream.

[0029] In the following description, the main-view and the sub-view are used to realize the parallax image method. The parallax image method (also called 3D-LR mode) is a method for realizing the stereoscopic viewing by preparing separately an image for the right eye and an image for the left eye, and causing the image for the right eye to enter only into the right eye and the image for the left eye enter only into the left eye. FIG. 2 shows the user's head on the left side of the drawing and the images of a dinosaur skeleton seen respectively by the left eye and the right eye of the user on the right side of the drawing. When the light transmission and block are repeated alternately for the left and right eyes, the left and right scenes are overlaid in the brain of the user by the effect of residual images of eyes, and the overlaid image is recognized as a stereoscopic image appearing in front of the user.

[0030] The MPEG4-MVC method is used as the method for encoding the video streams for realizing such a stereoscopic viewing. In the description hereinafter it is presumed that the main-view video stream is "base-view video stream" in the MPEG4-MVC method, and the sub-view video stream is "dependent-view video stream" in the MPEG4-MVC method.

[0031] The MPEG4-MVC base-view video stream is a sub-bit stream with view_id being set to "0", and is a sequence of view components with view_id being set to "0". The MPEG4-MVC base-view video stream conforms to the restrictions imposed on the MPEG4-AVC video stream.

[0032] The MPEG4-MVC dependent-view video stream is a sub-bit stream with view_id being set to "1", and is a sequence of view components with view_id being set to "1".

[0033] A view component is one of a plurality of pieces of picture data that are played back simultaneously for the stereoscopic viewing in one frame period. A compress-encoding that makes use of the correlation between view points is realized by using, as picture data, view components of the base-view and dependent-view video streams to realize a compress-encoding that makes use of the correlation between pictures. View components of the base-view and dependent-view video streams assigned to one frame period constitute one access unit. This makes it possible for the random access to be performed in a unit of the access unit.

[0034] Each of the base-view video stream and the dependent-view video stream has a GOP structure in which each view component is a "picture", and is composed of closed GOPs and open GOPs. The closed GOP is composed of an IDR picture, and B-pictures and P-pictures that follow the IDR picture. The open GOP is composed of a non-IDR I-picture, and B-pictures and P-pictures that follow the non-IDR I-picture.

[0035] The non-IDR I-pictures, B-pictures, and P-pictures are compress-encoded based on

the frame correlation with other pictures. The B-picture is a picture composed of slice data in the bidirectionally predictive (B) format, and the P-picture is a picture composed of slice data in the predictive (P) format. The B-picture is classified into reference B (Br) picture and non-reference B (B) picture.

[0036] In the closed GOP, the IDR picture is disposed at the top. In the display order, the IDR picture is not the top, but pictures (B-pictures and P-pictures) other than the IDR picture cannot have dependency relationship with pictures existing in a GOP that precedes the closed GOP. As understood from this, the closed GOP has a role to complete the dependency relationship.

[0037] FIG. 3 shows one example of the internal structures of the base-view and dependent-view video streams for the stereoscopic viewing.

[0038] The second row of FIG 3 shows the internal structure of the base-view video stream. This stream includes view components with picture types I1, P2, Br3, Br4, P5, Br6, Br7, and P9. These view components are decoded according to the Decode Time Stamps (DTS). The first row shows the left-eye image. The left-eye image is played back by playing back the decoded view components I1, P2, Br3, Br4, P5, Br6, Br7, and P9 according to the PTS, in the order of I1, Br3, Br4, P2, Br6, Br7, and P5.

[0039] The fourth row of FIG. 3 shows the internal structure of the dependent-view video stream. This stream includes view components with picture types P1, P2, B3, B4, P5, B6, B7, and P8. These view components are decoded according to the DTS. The third row shows the right-eye image. The right-eye image is played back by playing back the decoded view components P1, P2, B3, B4, P5, B6, B7, and P8 according to the PTS, in the order of P1, B3, B4, P2, B6, B7, and P5.

[0040] The fifth row of FIG 3 shows how the state of the 3D glasses 400 is changed. As shown in the fifth row, when the left-eye image is viewed, the shutter for the right eye is closed, and when the right-eye image is viewed, the shutter for the left eye is closed.

[0041] Here, a mode, in which video frames of the base-view video stream (B) and video frames of the dependent-view video stream (D) are alternately output at a display cycle of 1/48 seconds like "B"- "D"- "B"- "D", is called a "B-D presentation mode".

[0042] The B-D presentation mode includes a 3D-depth mode in which the stereoscopic viewing is realized by using the 2D images and depth information, as well as a 3D-LR mode in which the stereoscopic viewing is realized by using L (Left) images and R (Right) images.

[0043] Also, a mode, in which a same type of video frame is repeatedly output twice or more while the 3D mode is maintained as the output mode, is called a "B-B presentation mode". In the B-B presentation mode, video frames of an independently playable base-view video stream are repeatedly output like "B"- "B"- "B"- "B".

[0044] The B-D presentation mode and the B-B presentation mode described above are basic presentation modes in the playback device. Other than these, output modes such as a 1 plane + offset mode, an upper end 2D subtitle playback mode, and a lower end 2D subtitle playback mode are available in the playback device.

[0045] The 1 plane + offset mode (also referred to as "3D-offset mode") is an output mode in which the stereoscopic viewing is realized by incorporating a shift unit at a location subsequent to the plane memory and functioning the shift unit. In each of the left-view period and the right-view period, the plane offset unit shifts the coordinates of the pixels in the plane memory in units of lines leftward or rightward to displace the image formation point of the right-eye and left-eye view lines frontward or backward so that the viewer can feel a change in the sense of depth. More specifically, when the pixels coordinates are shifted leftward in the left-view period, and rightward in the right-view period, the image formation point is displaced frontward; and when the pixels coordinates are shifted rightward in the left-view period, and leftward in the right-view period, the image formation point is displaced backward.

[0046] In such a plane shift, the plane memory for the stereoscopic viewing only needs to have one plane. It is thus the best method for generating the stereoscopic images with ease. However, the plane shift merely produces stereoscopic images in which monoscopic images come frontward or go backward. Therefore, it is suited for generating a stereoscopic effect for the menu or subtitle, but leaves something to be desired in realizing a stereoscopic effect for the characters or physical objects. This is because it cannot reproduce dimples or unevenness of the faces of characters.

[0047] To support the 1 plane + offset mode, the playback device is structured as follows. For the playback of graphics, the playback device includes a plane memory, a CLUT unit, and an overlay unit. The plane shift unit is incorporated between the CLUT unit and the overlay unit. The plane shift unit realizes the above-described change of pixel coordinates by using the offset in the offset sequence incorporated in the access unit structure of the dependent-view video stream. With this arrangement, the level of jump-out of pixels in the 1 plane + offset mode changes in synchronization with the MVC video stream. The 1 plane + offset mode includes "1 plane + zero offset mode". The 1 plane + zero offset mode is a display mode which, when the pop-up menu is ON, gives the stereoscopic effect only to the pop-up menu by making the offset value zero.

[0048] The target of the shift control by the offset sequence is a plurality of plane memories which constitute a predetermined layer model. The plane memory is a memory for storing one screen of pixel data, which has been obtained by decoding the elementary streams, in units of lines so that the pixel data can be output in accordance with the horizontal and vertical sync signals. Each of the plurality of plane memories stores one screen of pixel data that is obtained as a result of decoding by the video decoder, PG decoder, or IG decoder.

[0049] The predetermined layer model is composed of a layer of the left-eye video plane and the right-eye video plane, a layer of the PG plane, and a layer of the IG/BD-J plane, and is

structured so that these layers (and the contents of the plane memories in these layers) can be overlaid in the order of the base-view video plane, PG plane, and IG/BD-J plane from the bottom.

[0050] The layer overlay is achieved by executing a superimposing process onto all combinations of the two layers in the layer model. In the superimposing process, pixel values of pixel data stored in the plane memories of the two layers are superimposed. The following describes the plane memories in each layer.

[0051] The left-eye video plane is a plane memory for storing pixel data constituting the left-eye picture data among one screen of pixel data that is obtained by decoding the view components. The right-eye video plane is a plane memory for storing pixel data constituting the right-eye picture data among one screen of pixel data that is obtained by decoding the view components.

[0052] The presentation graphics (PG) plane is a plane memory for storing graphics that are obtained when a graphics decoder, which operates by the pipeline method, performs the decoding process. The IG/BD-J plane is a plane memory that functions as an IG plane in some operation mode and functions as a BD-J plane in other operation mode. The interactive graphics (IG) plane is a plane memory for storing graphics that are obtained when a graphics decoder, which operates based on the interactive process, performs the decoding process. The BD-J plane is a plane memory for storing the drawing image graphics that are obtained when an application of an object-oriented programming language performs the drawing process. The IG plane and the BD-J plane are exclusive to each other, and when one of them is used, the other cannot be used. Therefore the IG plane and the BD-J plane share one plane memory.

[0053] In the above-mentioned layer model, with regard to the video plane, there are a base-view plane and a dependent-view plane. On the other hand, with regard to the IG/BD-J plane and the PG plane, there is neither a base-view plane nor a dependent-view plane. For this reason, the IG/BD-J plane and the PG plane are the target of the shift control.

[0054] The upper end 2D subtitle playback mode is an output mode in which a display region of a 2D subtitle is saved in the upper end of a video frame by incorporating a shift unit at a location subsequent to the video plane memory and causing the shift unit to function. The lower end 2D subtitle playback mode is an output mode in which a display region of a 2D subtitle is saved in the lower end of a video frame by causing the shift unit to function. In the upper end 2D subtitle playback mode, the plane offset unit shifts downward pixel coordinates of picture data stored in the video plane memory during each of the left-view period and the right-view period. In the lower end 2D subtitle playback mode, the plane offset unit shifts upward the pixel coordinates of the picture data stored in the video plane memory during each of the left-view period and the right-view period.

[0055] In order to support the upper end 2D subtitle playback mode and the lower end 2D

subtitle playback mode, the playback device needs to be structured as follows. The playback device includes a video plane memory and an overlay unit for playback of video frames and graphics, and further includes a shift unit incorporated between the video plane memory and the overlay unit. The shift unit realizes the change of pixel coordinates as described above, using an offset incorporated into a stream registration sequence of a graphics stream.

[0056] FIG. 4A shows a video frame suitable for use in the upper end 2D subtitle playback mode and the lower end 2D subtitle playback mode. In the figure, an image of a cinema scope size having an aspect ratio of 2.35:1 and a resolution of 1920×818 pixels is disposed in the center of a screen having an aspect ratio of 16:9 and a resolution of 1920×1080 pixels. A black frame having 1920×131 pixels is disposed in each of the upper end and the lower end of the image of the cinema scope size. In the video plane memory in which such a video frame is stored, pixel coordinates are shifted upward or downward, and black color data is stored in a blank region obtained by the shifting. As a result, the black frames that have been originally disposed in the upper and lower ends are collected in either the upper end or the lower end, as shown in FIGs. 4B and 4C. As a result, it is possible to prepare a black frame enough large to display subtitles.

[0057] This completes the explanation of the 3D output mode. The following explains the internal structure of the recording medium pertaining to the present embodiment.

[0058] FIGs. 5A through 5C show the internal structure of the recording medium in Embodiment 1. As shown in FIG. 5A, the recording medium in Embodiment 1 stores an "index table", an "operation mode object program file", a "playlist information file", a "stream information file", and a "stream file".

<Index table>

[0059] The index table is management information of the entire recording medium. The index table is read first by a playback device after the recording medium is loaded into the playback device, thereby the recording medium is uniquely identified by the playback device.

<Program file>

[0060] The program file of the operation mode object stores control programs for operating the playback device. The control program may be written as a set of commands or written in an object-oriented compiler language. The former program supplies a plurality of navigation commands as a batch job to the playback device in the command-based operation mode to operate the playback device based on the navigation commands. The command-based operation mode is called an "HDMV mode".

[0061] The latter program supplies bytecode applications, which are instances of class structure, to the playback device in the operation mode which is based on the object-oriented compiler language, in order to operate the playback device based on the instances. Java™ applications, which are one of the bytecode applications, can be used as the instances of class structure. The operation mode based on the object-oriented compiler language is called a "BD-J mode".

<Stream file>

[0062] A stream file stores a transport stream that is obtained by multiplexing a video stream, one or more audio streams, and a graphics stream. The stream file has two types: 2D-only; and 2D/3D shared. The 2D-only stream file is in a normal transport stream format. The 2D/3D shared stream file is in a stereoscopic interleaved stream file format.

[0063] The stereoscopic interleaved stream file format is a file format in which Extents of a main transport stream (main TS) including a base-view stream and Extents of a sub transport stream (sub TS) including a dependent-view stream are arranged in an interleaved manner.

[0064] The main TS stored in the stream file contains packet management information (PCR, PMT, PAT) defined in the European digital broadcast standard, as information for managing and controlling a plurality of types of PES streams.

[0065] The PCR (Program Clock Reference) stores STC time information corresponding to an ATS that indicates the time when the PCR packet is transferred to a decoder, in order to achieve synchronization between an ATC (Arrival Time Clock) that is a time axis of ATs, and an STC (System Time Clock) that is a time axis of PTs and DTs.

[0066] The PMT (Program Map Table) stores PIDs in the streams of video, audio, graphics and the like contained in the transport stream file, and attribute information of the streams corresponding to the PIDs. The PMT also has various descriptors relating to the TS. The descriptors have information such as copy control information showing whether copying of the AV clip is permitted or not.

[0067] The PAT (Program Association Table) shows a PID of a PMT used in the TS, and is registered by the PID arrangement of the PAT itself.

[0068] These PCR, PMT, and PAT, in the European digital broadcast standard, have a role of defining partial transport streams constituting one broadcast program (one program). This enables the playback device to cause the decoder to decode TSs as if it deals with the partial TSs constituting one broadcast program, conforming to the European digital broadcast standard. This structure is aimed to support compatibility between the recording medium playback devices and the terminal devices conforming to the European digital broadcast standard.

[0069] Pair of extents in the main TS and the sub-TS are each set to have a data size such that underflow of double buffer is not occurred during playback. This enables the playback device to load these pairs of extents without interruption.

[0070] This completes the description of the stream file.

<Stream information file>

[0071] The stream information file is a file for ensuring a random access to any source packet in a transport stream stored in a stream file, and ensuring a seamless playback with other transport streams. Via the stream information files, the stream files are managed as "AV clips". The stream information file includes information of the AV clip such as the stream encoding format, frame rate, bit rate, and resolution, and includes a basic entry map that shows correspondence between source packet numbers at the starts of GOPs and the presentation time stamps in the frame periods. Thus, by preloading the stream information file prior to an access to the stream file, the property of the transport stream in the stream file to be accessed is recognized, thereby the execution of the random access is ensured. The stream information file has two types: 2D stream information file; and 3D stream information file. The 3D stream information file includes clip information for the base view (clip base information), clip information for the dependent view (clip dependent information), and an entry map extended for the stereoscopic viewing.

[0072] The clip base information includes base-view extent start point information, and the clip dependent information includes dependent-view extent start point information. The base-view extent start point information includes a plurality of source packet numbers. Each source packet number indicates a packet number of a packet including a boundary between Extents in the main TS. The dependent-view extent start point information also includes a plurality of source packet numbers. Each source packet number indicates a packet number of a packet including a boundary between Extents in the sub-TS. By using these extent start point information, the stereoscopic interleaved stream file is divided into an ATC sequence 1 constituting the main TS and an ATC sequence 2 constituting the sub-TS. The ATC sequence is a sequence of source packets, wherein Arrival_Time_Clocks referred to by the Arrival_Time_Stamps included in the ATC sequence includes "no arrival time-base discontinuity". Since the ATC sequence is a sequence of source packets in which the ATC time stamps are continuous, each source packet constituting the ATC sequence is subjected to continuous source packet depacketizing processes and continuous packet filtering processes while the clock counter is counting the arrival time clocks of the playback device.

[0073] While the ATC sequence is a sequence of source packets, a sequence of TS packets whose time stamps are continuous in the STC time axis is called an "STC sequences". The STC sequence is a sequence of TS packets which do not include "system time-base discontinuity", which is based on the STC (System Time Clock) that is a system standard time

for TSs. The presence of the system time-base discontinuity is indicated by a "discontinuity_indicator" being ON, where the discontinuity_indicator is contained in a PCR packet carrying a PCR (Program Clock Reference) that is referred to by the decoder to obtain an STC. The STC sequence is a sequence of TS packets whose time stamps are continuous in the STC time axis. Therefore, each TS packet constituting the STC sequence is subjected to continuous decoding processes performed by the decoder provided in the playback device, while the clock counter is counting the system time clocks of the playback device. The extension entry map indicates, in correspondence with the presentation time stamps representing the frame periods at the starts of GOPs, source packet numbers of access unit delimiters which indicate starting positions of view components at the starts of GOPs in the dependent-view video stream.

[0074] On the other hand, the basic entry map in the 3D stream information file indicates, while maintaining the compatibility with the 2D stream information file, in correspondence with the presentation time stamps representing the frame periods at the starts of GOPs, source packet numbers of access unit delimiters which indicate starting positions of view components at the starts of GOPs in the base-view video stream.

<Playlist information file>

[0075] The playlist information file is a file storing information that is used to cause the playback device to play back a playlist. The "playlist" indicates a playback path defined by logically specifying a playback order of playback sections, where the playback sections are defined on a time axis of transport streams (TS). The playlist has a role of defining a sequence of scenes to be displayed in order, by indicating which parts of which transport streams among a plurality of transport streams should be played back. The playlist 1 information defines "patterns" of the playlists. The playback path defined by the playlist information is what is called a "multi-path". The multi-path is composed of a "main path" and one or more "sub-paths". The main path is defined for the main transport streams. The sub-paths are defined for sub streams. A plurality of sub-paths can be defined while one main path is defined. By defining a playback path of the base-view video stream in the main path and defining a playback path of the dependent-view video stream in the sub-path, it is possible to suitably define a set of video streams for performing a stereoscopic playback.

[0076] AV playback by the multi-path can be started when the application of an object-oriented programming language instructs to generate a frame work player instance that plays back the playlist information. The frame work player instance is actual data that is generated on the heap memory of the virtual machine based on the media frame work player class. Also, arrangement may be made so that playback by the multi-path can be started when a command-based program issues a playback command with an argument specifying the playlist information.

[0077] The playlist information includes one or more pieces of playitem information. The

playitem information is playback section information that defines one or more pairs of an "in_time" time point and an "out_time" time point on the video stream playback time axis.

[0078] The playlist information has a hierarchical structure composed of playitem information, clip information, and a transport stream. It is possible to set a one-to-many relationship between (i) a pair of transport stream and clip information and (ii) playitem information so that one transport stream can be referenced by a plurality of pieces of playitem information. This makes it possible to adopt, as a bank film, a transport stream created for a title so that the bank film can be referenced by a plurality of pieces of playitem information in a plurality of playlist information files, making it possible to create a plurality of variations of a movie effectively. Note that the "bank film" is a term used in the movie industry and means an image that is used in a plurality of scenes.

[0079] In general, the users do not recognize the unit called playlist, and recognize a plurality of variations (for example, a theatrical version and a TV broadcast version) branched from the stream files as the playlists.

[0080] The playlist information falls into two types: 2D playlist information; and 3D playlist information. A difference between them is that the 3D playlist information includes a base-view indicator and a stereoscopic stream selection table.

[0081] The "stereoscopic stream selection table" is a table that shows, in correspondence with stream numbers, stream attributes and stream entries of elementary streams that are to be played back only in the 3D output mode.

[0082] The "base-view indicator" is information indicating either the left eye or the right eye for which the base-view video stream is to be indicated, wherein the base-view video stream is the base of the compress-encoding using the correlation between view points. By changing the base-view indicator of the playlist information, it is possible to change the assignment of the left eye and right eye at the level of the playlist.

[0083] Since the assignment of the left eye and right eye can be changed at the level of the playlist that does not depend on the structure of the stream, when, for example, there is a playlist in which the position and angle of an object in the image is set as "base view = left eye" and "dependent view = right eye", it is possible to generate a playlist in which the position and angle of an object in the image is set as "base view = right eye" and "dependent view = left eye", as another version.

[0084] By reversing the assignment of the left eye and right eye to the base-view and dependent-view video streams at the level of the playlist, it is possible to reverse the stereoscopic effect. For example, when there has already been generated a playlist intending a stereoscopic effect that the object appears in front of the screen, it is possible to generate another playlist intending a stereoscopic effect that the object appears behind the screen. This produces an advantageous effect that variations of 3D playlists with different stereoscopic

effects can be generated easily.

[0085] FIG. 5B shows the internal structure of the main TS. FIG. 5C shows the internal structure of the sub-TS. As shown in FIG 5B, the main TS includes one base-view video stream, 32 base-view PG streams, 32 base-view IG streams, and 32 audio streams. As shown in FIG. 5C, the sub-TS includes one dependent-view video stream, 32 dependent-view PG streams, and 32 dependent-view IG streams.

[0086] Next, the internal structure of TS will be described.

[0087] The elementary streams (ES) to be multiplexed in the TSs include the video stream, audio stream, presentation graphics stream, and interactive graphics stream.

(Video stream)

[0088] The base-view video stream constitutes a primary video stream in a picture-in-picture application. The picture-in-picture application is composed of the primary video stream and a secondary video stream. The primary video stream is a video stream composed of picture data of the picture-in-picture application that represents a parent picture in the screen; and the secondary video stream is a video stream composed of picture data of the picture-in-picture application that represents a child picture that is fit in the parent picture.

[0089] The picture data constituting the primary video stream and the picture data constituting the secondary video stream are stored in different plane memories after being decoded. The plane memory that stores the picture data constituting the secondary video stream has, in the first half thereof, a structural element (Scaling & Positioning) that performs changing scaling of the picture data constituting the secondary video stream, and positioning display coordinates of the picture data constituting the secondary video stream.

(Audio stream)

[0090] The audio stream is classified into two types of a primary audio stream and a secondary audio stream.

[0091] The primary audio stream is an audio stream that is to be a main audio when the mixing playback is performed; and the secondary audio stream is an audio stream that is to be a sub-audio when the mixing playback is performed. The secondary audio stream includes information for downsampling for the mixing, and information for the gain control.

(Presentation graphics (PG) stream)

[0092] The PG stream is a graphics stream that can be synchronized closely with the video, with the adoption of the pipeline in the decoder, and is suited for representing subtitles. The PG stream falls into two types: a 2D PG stream; and a stereoscopic PG stream. The stereoscopic PG stream further falls into two types: a left-eye PG stream; and a right-eye PG stream.

[0093] It is possible to define up to 32 2D PG streams, up to 32 left-eye PG streams, and up to 32 right-eye PG streams. These PG streams are attached with different packet identifiers. Thus, it is possible to cause a desired PG stream among these PG streams to be subjected to the playback, by specifying a packet identifier of the one to be played back to the demultiplexing unit.

[0094] Close synchronization with video is achieved due to the decoding with the pipeline adopted therein. Thus the use of the PG stream is not limited to the playback of characters such as the subtitle characters. For example, it is possible to display a mascot character of the movie that is moving in synchronization with the video. In this way, any graphics playback that requires close synchronization with the video can be adopted as a target of the playback by the PG stream.

[0095] The PG stream is a stream that is not multiplexed into the transport stream but represents a subtitle. The text subtitle stream (also referred to as a "textST stream") is a stream of this kind, as well. The textST stream is a stream that represents the contents of subtitle by the character codes.

[0096] The PG stream and the text subtitle stream are registered as the same stream type in the same stream registration sequence, without distinction between them in type. And then during execution of a procedure for selecting a stream, a PG stream or a text subtitle stream to be played back is determined according to the order of streams registered in the stream registration sequence. In this way, the PG streams and text subtitle streams are subjected to the stream selection procedure without distinction between them in type. Therefore, they are treated as belonging to a same stream type called "PG_text subtitle stream".

[0097] The PG_text subtitle stream for 2D is played back in the 1 plane + offset mode, the upper end 2D subtitle playback mode, and the lower end 2D subtitle playback mode.

(Interactive Graphics (IG) stream)

[0098] The IG stream is a graphics stream which, having information for interactive operation, can display menus with the progress of playback of the video stream and display pop-up menus in accordance with user operations.

[0099] As is the case with the PG stream, the IG stream is classified into two types of a 2D IG

stream and a stereoscopic IG stream. The IG stream control information (called "interactive control segment") includes information (user_interface_model) that defines the user interface model. The person in charge of authoring can specify either "always on" or "pop-up menu on" by setting the user interface model information, where with the "always on", menus are displayed with the progress of playback of the video stream, and with the "pop-up menu on", the pop-up menus are displayed in accordance with user operations.

[0100] The interactive operation information in the IG stream has the following meaning. When the Java™ virtual machine instructs the playback control engine, which is proactive in the playback control, to start playing back a playlist in accordance with a request from an application, the Java™ virtual machine, after instructing the playback control engine to start the playback, returns a response to the application to notify that the playback of the playlist has started. That is to say, while the playback of the playlist by the playback control engine continues, the Java™ virtual machine does not enter the state waiting for end of execution. This is because the Java™ virtual machine is what is called an "event-driven-type" performer, and can perform operation while the playback control engine is playing back the playlist.

[0101] On the other hand, when, in the HDMV mode, the command interpreter instructs the playback control engine to play back a playlist, it enters the wait state until the execution of playback of the playlist ends. Accordingly, the command execution unit cannot execute an interactive process while the playback of the playlist by the playback control engine continues. The graphics decoder performs an interactive operation in place of the command interpreter. Thus, to cause the graphics decoder to perform the interactive operation, the IG stream is embedded with control information defining interactive operations for which buttons are used.

(Display modes allowed for each stream type)

[0102] Different 3D display modes are allowed for each stream type. In the primary video stream 3D display mode, two output modes, namely the B-D presentation mode and the B-B presentation mode are allowed. The B-B presentation mode is allowed for the primary video stream only when the pop-up menu is on. The type of primary video stream when the playback is performed in the B-D presentation mode is called a "stereoscopic B-D playback type". The type of primary video stream when the playback is performed in the B-B presentation mode is called a "stereoscopic B-B playback type".

[0103] In the PG stream 3D display mode, five output modes, namely the B-D presentation mode, 1 plane + offset mode, "1 plane + Zero Offset" mode, upper end 2D subtitle playback mode, and lower end 2D subtitle playback mode are allowed. The "1 plane + zero offset" mode is allowed for the PG stream only when the pop-up menu is on. The type of PG stream when the playback is performed in the B-D presentation mode is called a "stereoscopic playback type". The type of PG stream and PG_text subtitle stream when the playback is performed in the 1 plane + offset mode is called a "1 plane + offset type". The type of PG stream and PG_text subtitle stream when the playback is performed in the "1 plane + zero offset" mode is

called a "1 plane + zero offset type". A type of a PG stream or a text subtitle stream that is played back in the upper end 2D subtitle playback mode is referred to as an "upper end 2D subtitle playback type". A type of a PG stream or a text subtitle stream that is played back in the lower end 2D subtitle playback mode is referred to as a "lower end 2D subtitle playback type".

[0104] In the text subtitle stream 3D display mode, four output modes, namely the 1 plane + offset mode, the "1 plane + zero offset", upper end 2D subtitle playback mode, and lower end 2D subtitle playback mode are allowed. The "1 plane + zero offset" mode is allowed for the text subtitle stream only when the pop-up menu is on.

[0105] In the IG stream 3D display mode, three output modes, namely the B-D presentation mode, 1 plane + offset mode, and "1 plane + zero offset" mode are allowed. The "1 plane + zero offset" mode is allowed for the IG stream only when the pop-up menu is on. It is supposed in the following description, except where otherwise mentioned, that the picture-in-picture cannot be used during playback in the 3D output mode. This is because each of the picture-in-picture and the 3D output mode requires two video planes for storing non-compressed picture data. It is also supposed in the following description, except where otherwise mentioned, that the sound mixing cannot be used in the 3D output mode.

[0106] Next, the internal structures of the main TS and sub-TS will be described. FIGs. 6A and 6B show the internal structures of the main TS and sub-TS.

[0107] FIG. 6A shows the internal structure of the main TS. The main TS is composed of the following source packets.

[0108] A source packet having packet ID "0x0100" constitutes a Program_Map_Table (PMT). A source packet having packet ID "0x0101" constitutes a PCR.

[0109] A source packet sequence having packet ID "0x1011" constitutes the primary video stream.

[0110] Source packet sequences having packet IDs "0x1200" through "0x121F" constitute 32 2D PG streams.

[0111] Source packet sequences having packet IDs "0x1400" through "0x141F" constitute 32 2D IG streams.

[0112] Source packet sequences having packet IDs "0x1100" through "0x111F" constitute primary audio streams.

[0113] By specifying a packet identifiers of one of these source packets to the demultiplexing unit, it is possible to cause a desired elementary stream among a plurality of elementary streams multiplexed in the main transport streams to be demultiplexed and subjected to the

decoder.

[0114] FIG. 6B shows the internal structure of the sub-TS. The sub-TS is composed of the following source packets.

[0115] A source packet sequence having packet ID "0x1012" constitutes the dependent-view video stream.

[0116] Source packet sequences having packet IDs "0x1220" through "0x123F" constitute 32 left-eye PG streams.

[0117] Source packet sequences having packet IDs "0x1240" through "0x125F" constitute 32 right-eye PG streams.

[0118] Source packet sequences having packet IDs "0x1420" through "0x143F" constitute 32 left-eye IG streams.

[0119] Source packet sequences having packet IDs "0x1440" through "0x145F" constitute 32 right-eye IG streams.

[0120] This completes the description of the stream file. Next is a detailed explanation of the playlist information.

[0121] To define the above-described multi-path, the internal structures shown in FIGs. 7A through 7D are provided. FIG. 7A shows the internal structure of the playlist information. As shown in FIG. 7A, the playlist information includes main-path information, sub-path information, playlist mark information, and extension data. These constitutional elements will be described in the following.

1. 1) The main-path information is composed of one or more pieces of main playback section information. FIG. 7B shows the internal structures of the main-path information and the sub-path information. As shown in FIG. 7B, the main-path information is composed of one or more pieces of main playback section information, and the sub-path information is composed of one or more pieces of sub playback section information.

The main playback section information, called playitem information, is information that defines one or more logical playback sections by defining one or more pairs of an "in_time" time point and an "out_time" time point on the TS playback time axis. The playback device is provided with a playitem number register storing the playitem number of the current playitem. The playitem being played back currently is one of the plurality of playitems whose playitem number is currently stored in the playitem number register.

FIG. 7C shows the internal structure of the playitem information. As shown in FIG. 7C, the playitem information includes stream reference information, in-time out-time information, connection state information, and a basic stream selection table.

The stream reference information includes: "stream Information file name information

(clip_information_file_name)" that indicates the file name of the stream information file that manages, as "AV clips", the transport streams constituting the playitem; "clip encoding method identifier (clip_codec_identifier)" that indicates the encoding method of the transport stream; and "STC identifier reference (STC_ID_reference)" that indicates STC sequences in which in-time and out-time are set, among the STC sequences of the transport stream.

This completes description of the playitem information.

2. 2) The sub playback section information, called sub-path information, is composed of a plurality of pieces of sub-playitem information. FIG. 7D shows the internal structure of the sub-playitem information. As shown in FIG. 7D, the sub-playitem information is information that defines playback sections by defining pairs of an "in_time" and an "out_time" on the STC sequence time axis, and includes stream reference information, in-time out-time information, sync playitem reference, and sync start time information.

The stream reference information, as in the playitem information, includes: "stream Information file name information", "clip encoding method identifier", and "STC identifier reference".

The "in-time out-time information (SubPlayItem_In_Time, SubPlayItem_Out_Time)" indicates the start point and end point of the sub-playitem on the STC sequence time axis.

The "sync start time information (Sync_Start_PTS_of_Playitem)" indicates a time point on the STC sequence time axis of the playitem specified by the sync playitem identifier, that corresponds to the start point of the sub-playitem specified by the sub-playitem In_Time. The sub-playitem In_Time exists on playback time axis of the playitem specified by this sync playitem identifier.

The "sync start time information (Sync_Start_PTS_of_Playitem)" indicates a time point on the STC sequence time axis of the playitem specified by the sync playitem identifier, that corresponds to the start point of the sub-playitem specified by the sub-playitem In_Time.

3. 3) The playlist mark information is information that defines the mark point unique to the playback section. The playlist mark information includes an indicator indicating a playback section, a time stamp indicating the position of a mark point on the time axis of the digital stream, and attribute information indicating the attribute of the mark point.

[0122] The attribute information indicates whether the mark point defined by the playlist mark information is a link point or an entry mark.

[0123] The link point is a mark point that can be linked by the link command, but cannot be selected when the chapter skip operation is instructed by the user.

[0124] The entry mark is a mark point that can be linked by the link command, and can be selected even if the chapter skip operation is instructed by the user.

[0125] The link command embedded in the button information of the IG stream specifies a position for a random-access playback, in the form of an indirect reference via the playlist mark information.

<Basic stream selection table (StreamNumber_table)>

[0126] The basic stream selection table shows a list of elementary streams that are to be played back in a monoscopic output mode, and the table, when a playitem containing the basic stream selection table itself becomes the current playitem among a plurality of playitems constituting the playlist, specifies, for each of the plurality of stream types, an ES which is permitted to be played back, among ESs multiplexed in AV clips referenced by the main path and the sub-path of the multi-path. Here, the stream types include: the primary video stream in the picture-in-picture; the secondary video stream in the picture-in-picture; the primary audio stream in the sound mixing; the secondary audio stream in the sound mixing; the PG_text subtitle stream; and the IG stream. It is possible to register an ES which is permitted to be played back, for each of these stream types. More specifically, the basic stream selection table is composed of sequences of stream registrations. Here, the stream registration is information that, when a playitem containing the basic stream selection table itself becomes the current playitem, indicates what kind of stream is the ES permitted to be played back. Each stream registration is associated with the stream number of the stream. Each stream registration has a data structure in which a pair of a stream entry and a stream attribute is associated with a logical stream number.

[0127] The stream number in the stream registration is represented by an integer such as "1", "2", or "3". The largest stream number for a stream type is identical with the number of streams for the stream type.

[0128] The playback device is provided with a stream number register for each stream type, and the current stream, namely the ES being played back currently, is indicated by the stream number stored in the stream number register.

[0129] A packet identifier of the ES to be played back is written in the stream entry. By making use of this structure in which a packet identifier of the ES to be played back can be written in the stream entry, the stream numbers included in the stream registrations are stored in the stream number registers of the playback device, and the playback device causes the PID filter thereof to perform a packet filtering based on the packet identifiers stored in the stream entries of the stream registrations. With this structure, TS packets of the ESs that are permitted to be played back according to the basic stream selection table are output to the decoder, so that the ESs are played back.

[0130] In the basic stream selection table, the stream registrations are arranged in an order of stream numbers. When there are a plurality of streams that satisfy the conditions: "playable by playback device"; and "the language attribute of the stream matches the language setting in

the device", a stream corresponding to the highest stream number in the stream registration sequences is selected.

[0131] With this structure, when there is found a stream that cannot be played back by the playback device, among the stream registrations in the basic stream selection table, the stream is excluded from the playback. Also, when there are a plurality of streams that satisfy the conditions: "playable by playback device"; and "the language attribute of the stream matches the language setting in the device", the person in charge of authoring can convey the playback device how to select one with priority from among the plurality of streams.

[0132] It is judged whether there is a stream that satisfies the conditions: "playable by playback device"; and "the language attribute of the stream matches the language setting in the device". Also, a stream is selected from among a plurality of streams that satisfy the conditions. The procedure for the judgment and selection is called "stream selection procedure". The stream selection procedure is executed when the current playitem is switched, or when a request to switch the stream is input by the user.

[0133] A sequential procedure for performing the above-described judgment and selection and setting a stream number in the stream number register of the playback device when a state change occurs in the playback device, such as when the current playitem is switched, is called "procedure to be executed at state change". Since the stream number registers are provided respectively in correspondence with the stream types, the above-described procedure is executed for each stream type.

[0134] A sequential procedure for performing the above-described judgment and selection and setting a stream number in the stream number register of the playback device when a request to switch the stream is input by the user is called "procedure at state change request".

[0135] A procedure for setting the stream number registers to the initial values of the stream registration sequences when a BD-ROM is loaded, is called "initialization".

[0136] Priorities are assigned evenly to the streams specified in the sub-playitem information and the streams specified in the playitem information, as indicated by the stream registration sequences in the basic stream selection table. As a result, even a stream not multiplexed with a video stream is targeted for selection as a stream to be played back in sync with the video stream, if the stream is specified by the sub-playitem information.

[0137] Furthermore, when playback device can play back a stream specified by the sub-playitem information, and when the priority of the stream specified by the sub-playitem information is higher than the priority of the graphics stream multiplexed with the video stream, the stream specified by the sub-playitem information is played back in place of the stream multiplexed with the video stream.

[0138] The following explains the use of the stream numbers recited in the basic stream

selection table. The stream numbers recited in the basic stream selection table can be used as operands of the set stream command.

[0139] The set stream command is a command that instructs the playback device to change the current stream by setting the stream number specified by the operand into the stream number register as the current stream number. The set stream command is used by a command-based program when it causes the playback device to change the stream.

[0140] The set stream command can be used as an argument of the stream change UO or an argument of the set stream API, as well. The stream change UO is a user operation event that instructs the playback device to change the current stream by setting the stream number specified by the argument into the stream number register as the current stream number.

[0141] The set stream API is an API that instructs the playback device to change the current stream by setting the stream number specified by the argument into the stream number register as the current stream number. The set stream API is used by a program based on an object-oriented programming language when it causes the playback device to change the stream.

[0142] FIGs. 8A and 8B show one example of the basic stream selection table. FIG. 8A shows a plurality of stream registration sequences that are provided in the basic stream selection table when there are following stream types: primary video stream; primary audio stream; PG stream; IG stream; secondary video stream; and secondary audio stream. FIG. 8B shows the elementary streams that are demultiplexed from the main TS and the sub-TSs with use of the basic stream selection table. The left side of FIG. 8B shows the main TS and the sub-TSs, the middle part of FIG. 8B shows the basic stream selection table and the demultiplexing unit, and the right side of FIG. 8B shows the primary video stream, primary audio stream, PG stream, IG stream, secondary video stream, and secondary audio stream that are demultiplexed based on the basic stream selection table.

[0143] Next, the extension data will be described in detail.

[0144] When the playlist information refers to the MVC video stream, an extension stream selection table needs to be stored in a data block of extension data in the playlist information file.

[0145] When the playlist information refers to the MVC video stream on the disc, or the MVC video stream in the stereoscopic IG stream playback menu, extension information of the sub-path information (sub-path block extension) needs to be stored in a data block of extension data in the playlist information file.

[0146] When a 2D playback device finds unknown extension data in the playlist file, the 2D playback device should disregard the extension data.

<Extension stream selection table (StreamNumber_table_ StereoScopic(SS))>

[0147] The extension stream selection table shows a list of elementary streams that are to be played back in a stereoscopic output mode, and is used together with the basic stream selection table only in the stereoscopic output mode. The extension stream selection table defines the elementary streams that can be selected when a playitem is played back or when a sub-path related to the playitem is played back.

[0148] The extension stream selection table indicates the elementary streams that are permitted to be played back only in the stereoscopic output mode, and includes stream registration sequences. Each piece of stream registration information in the stream registration sequences includes a stream number, and a stream entry and a stream attribute corresponding to the stream number. The extension stream selection table means an extension that is unique to the stereoscopic output mode. Therefore, a playlist for which each piece of playitem information is associated with the extension stream selection table (STN_table_SS) is called "3D playlist".

[0149] Each stream entry in the extension stream selection table indicates a packet identifier that is to be used in the demultiplexing by the playback device, when the playback device is in the stereoscopic output mode, and the corresponding stream number is set in the stream number register of the playback device. A difference from the basic stream selection table is that the stream registration sequences in the extension stream selection table are not targeted by the stream selection procedure. That is to say, the stream registration information in the stream registration sequences of the basic stream selection table is interpreted as the priorities of the elementary streams, and a stream number in any piece of stream registration information is written into the stream number register. In contrast, the stream registration sequences of the extension stream selection table are not targeted by the stream selection procedure, and the stream registration information of the extension stream selection table is used only for the purpose of extracting a stream entry and a stream attribute that correspond to a certain stream number when the certain stream number is stored in the stream number register.

[0150] Suppose that, when the output mode switches from the 2D output mode to the 3D output mode, the target stream selection table also switches from the basic stream selection table to the extension stream selection table. Then, the identity of the stream numbers may not be maintained, and the identity of the language attribute may be lost, as well.

[0151] Accordingly, the use of the extension stream selection table is restricted to the above-described one to maintain the identity of the stream attribute such as the language attribute.

[0152] The following explains the use of the stream numbers recited in the extension stream selection table. The stream numbers recited in the extension stream selection table can be

used as operands of the set stream command and the set stereoscopic stream command.

[0153] The set stereoscopic stream command is a command that instructs the playback device to change the current stream by setting the stream number for stereoscopic viewing specified by the operand into the stream number register as the current stream number. The set stereoscopic stream command is used by a command-based program when it causes the playback device to change the stereoscopic stream.

[0154] The set stereoscopic stream command can be used as an argument of the stream change UO or an argument of the set stream API, as well.

[0155] The extension stream selection table is composed of stream registration sequences of the dependent-view streams, stream registration sequences of the PG streams, and stream registration sequences of the IG streams.

[0156] The stream registration sequences in the extension stream selection table are combined with the stream registration sequences of the same stream types in the basic stream selection table. More specifically, the dependent-view video stream registration sequences in the extension stream selection table are combined with the primary video stream registration sequences in the basic stream selection table; the PG stream registration sequences in the extension stream selection table are combined with the PG stream registration sequences in the basic stream selection table; and the IG stream registration sequences in the extension stream selection table are combined with the IG stream registration sequences in the basic stream selection table.

[0157] After this combination, the above-described procedure is executed onto the stream registration sequences in the basic stream selection table among the two tables after the combination.

[0158] FIG. 9 shows the internal structure of the extension stream selection table. The extension stream selection table is composed of: "length" which indicates the entire length of the extension stream selection table; "fixed offset during pop-up (Fixed_offset_during_Popup)"; and the stream registration sequences of each stream type corresponding to each playitem.

[0159] When there are N pieces of playitems identified as playitems #1-#N, stream registration sequences respectively corresponding to the playitems #1-#N are provided in the extension stream selection table. The stream registration sequences corresponding to each playitem are dependent-view stream registration sequence, PG stream registration sequence, and IG stream registration sequence.

[0160] The "Fixed_offset_during_Popup" is a fixed offset during pop-up, and controls the playback type of the video or PG_text subtitle stream when the pop-up menu is set to "on" in the IG stream. The "Fixed_offset_during_Popup" field is set to "on" when the "user_interface_model" field in the IG stream is on, namely, when the user interface of the pop-

up menu is set to "on". Also, the "Fixed_offset_during_Popup" field is set to "off" when the "user_interface_model" field in the IG stream is off, namely, when the "AlwaysON" user interface is set.

[0161] When the fixed offset during pop-up is set to "0", namely, when the pop-up menu is set to "off" in the user interface of the IG stream, the video stream is in the B-D presentation mode, the stereoscopic PG stream becomes the stereoscopic playback type, and during playback in the 1 plane + offset mode, the PG_text subtitle stream is in the 1 plane + offset mode.

[0162] When the fixed offset during pop-up is set to "1", namely, when the pop-up menu is set to "on" in the IG stream, the video stream is in the B-B presentation mode. The stereoscopic PG stream is in the 1 plane + offset mode, and the PG stream for "1 plane + offset" is played back as the "1 plane + zero offset" playback type.

[0163] In the 1 plane + offset mode, the PG-text subtitle stream becomes "1 plane + zero offset".

[0164] FIGs. 10A through 10C show the stream registration sequences in the extension stream selection table.

[0165] FIG. 10A shows the internal structure of the dependent-view video stream registration sequence. The dependent-view video stream registration sequence is composed of $v(x)$ pieces of SS_dependent_view_blocks. Here, " $v(x)$ " represents the number of primary video streams that are permitted to be played back in the basic stream selection table of the playitem information #x. The lead lines in the drawing indicate the close-up of the internal structure of the dependent-view video stream registration sequence. As indicated by the lead lines, the "SS_dependent_view_block" is composed of the stream number, stream entry, stream attribute, and the number of offset sequences (number_of_offset_sequence).

[0166] The stream entry includes: a sub-path identifier reference (ref_to_Subpath_id) specifying a sub-path to which the playback path of the dependent-view video stream belongs; a stream file reference (ref_to_subClip_entry_id) specifying a stream file in which the dependent-view video stream is stored; and a packet identifier (ref_to_stream_PID_subclip) of the dependent-view video stream in this stream file.

[0167] The "stream attribute" includes the language attribute of the dependent-view video stream.

[0168] The "the number of offset sequences (number_of_offset_sequence)" indicates the number of offsets provided in the dependent-view video stream.

[0169] The "offset sequence number information" ("number_of_offset_sequence" in the drawing) indicates the number of offset sequences in the dependent-view stream.

[0170] The value of the "offset sequence number information in the extension stream selection table is identical with the number of offset sequences that is included in the dependent-view stream.

[0171] The dependent-view video stream registration sequences shown in FIG 10A indicate that a plurality of pieces of stream registration information are provided in correspondence with a plurality of dependent-view video streams. However, FIG. 10A merely illustrates the data structure thereof. In the actuality, since there is only one base-view video stream normally, the number of pieces of stream registration information for the dependent-view video stream is one.

[0172] FIG. 10B shows the internal structure of the PG stream registration sequence. The PG stream registration sequence is composed of P(x) pieces of stream registration information. Here, "P(x)" represents the number of PG streams that are permitted to be played back in the basic stream selection table of the playitem information #x.

[0173] The lead lines in the drawing indicates the close-up of the common internal structure of the PG stream registration sequences.

[0174] The "PG_text subtitle offset sequence ID reference information (PGtextST_offset_sequence_id_ref)" is PG_text subtitle stream offset sequence reference information, and indicates an offset sequence with respect to the PG_text subtitle stream in the 1 plane + offset mode.

[0175] The offset metadata is supplied by the access unit of the dependent-view video stream. The playback device should apply the offset, which is supplied by this field, to the presentation graphics (PG) plane of the 1 plane + offset mode type.

[0176] When the field is an undefined value (FF), the playback device does not apply this offset to the PG stream plane memory.

[0177] The "stereoscopic PG presence/absence flag (is_SS_PG)" indicates the validity and presence of the following in the PG stream: the left-eye IG stream entry; the right-eye IG stream entry; and the stream attributes. When the structure is absent in the stereoscopic PG stream, this field should be set to "0"; and when the structure is present in the stereoscopic PG stream, this field should be set to "1".

[0178] The "left-eye stream entry" includes: a sub-path identifier reference (ref_to_Subpath_id) specifying a sub-path to which the playback path of the left-eye PG stream belongs; a stream file reference (ref_to_subClip_entry_id) specifying a stream file in which the left-eye PG stream is stored; and a packet identifier (ref_to_stream_PID_subclip) of the left-eye PG stream in this stream file.

[0179] The "right-eye stream entry" includes: a sub-path identifier reference

(ref_to_Subpath_id) specifying a sub-path to which the playback path of the right-eye PG stream belongs; a stream file reference (ref_to_subClip_entry_id) specifying a stream file in which the right-eye PG stream is stored; and a packet identifier (ref_to_stream_PID_subclip) of the right-eye PG stream in this stream file. When the stream file referenced by the "stream_entry_for_dependent_view" in the stream registration information in the extension stream selection table is different from the stream file referenced by the stream entry in the basic stream selection table, a stream file storing the right-eye PG stream needs to be read again.

[0180] The "common stream attribute" includes language attributes of the left-eye PG stream and the right-eye PG stream.

[0181] The "stereoscopic PG_text subtitle offset sequence ID reference information (SS_PG_textST_offset_sequence_id_ret)" is reference information for referencing an offset sequence for the PG_text subtitle stream, and indicates the offset sequence for the PG_text subtitle stream. The playback device should apply the offset, which is supplied by this field, to the PG plane.

[0182] When the field is an undefined value (FF), the playback device does not apply this offset to the PG stream plane memory.

[0183] The "Video shift mode (video_shift_mode)" is a region-saving flag that defines processing of saving a display region of a subtitle. The region-saving flag indicates whether the display region of the subtitle is to be saved in the upper end or the lower end in the video plane. When the display region of the subtitle is neither saved in the upper end nor the lower end in the video plane, the video shift mode is set to "Keep". When the video_shift_mode is set to "Keep", picture data stored in the video plane memory is neither shifted upward nor downward, and the picture data are overlaid with a subtitle stored in the PG stream plane memory, as shown in FIG. 11.

[0184] When the subtitle display region of the PG_text subtitle stream is located in the lower end of the video plane, the video shift mode is set to "Up". When the subtitle display region of the PG_text subtitle stream is located in the upper end of the video plane, the video shift mode is set to "Down".

[0185] When subtitles obtained by decoding a PG stream whose video_shift_mode is set to "Up" are located in the lower end of the screen. Accordingly, as shown in FIG. 12A, picture data stored in the video plane memory is shifted upward, and the picture data is overlaid with a subtitle stored in the PG stream plane memory. This can prevent the subtitle from being displayed as if the subtitle dents in stereoscopic images. When subtitles obtained by decoding a PG stream whose video_shift_mode is set to "Down" are located in the upper end of the screen. Accordingly, as shown in FIG. 12B, picture data stored in the video plane memory is shifted downward, and the picture data are overlaid with subtitles stored in the PG stream plane memory. This can prevent the subtitles from being displayed as if the subtitles dent in stereoscopic images.

[0186] FIG. 10C shows the internal structure of the IG stream registration sequence. The IG stream registration sequence is composed of $I(x)$ pieces of stream registration information. Here, " $I(x)$ " represents the number of IG streams that are permitted to be played back in the basic stream selection table of the playitem information # x . The lead lines in the drawing indicate the close-up of the common internal structure of the PG stream registration sequences.

[0187] The "IG offset sequence ID reference information (IG_offset_sequence_id_ref)" is an interactive graphics offset sequence reference, and is a reference to the sequence ID of the IG stream in the 1 plane + offset mode. This value indicates an offset sequence ID defined for the offset sequence. As described above, the offset metadata is supplied by the dependent-view video stream. The playback device should apply the offset, which is supplied by this field, to the IG stream of the 1 plane + offset mode type.

[0188] When the field is an undefined value (FF), the playback device does not apply this offset to the interactive graphics (IG) stream plane.

[0189] The "B-B mode offset direction information (IG_Plane_offset_direction_during_BB_video)" is the user interface of the pop-up menu in the B-B presentation mode, and indicates the offset direction in the IG plane in the 1 plane + offset mode while the IG stream is played back.

[0190] When this field is set to "0", it is the front setting. That is to say, the plane memory exists between the television and the viewer, and the plane is shifted rightward during the left-view period, and the plane is shifted leftward during the right-view period.

[0191] When this field is set to a value "1", it is the behind setting. That is to say, the plane memory exists behind the television or the screen, and the left plane is shifted rightward, and the right plane is shifted leftward.

[0192] The "B-B mode offset value information (IG_Plane_offset_value_during_BB_video)" indicates, in units of pixels, the offset value of the IG plane in the 1 plane + offset mode while the IG stream is played back by the user interface of the pop-up menu in the B-B presentation mode.

[0193] The "stereoscopic IG presence/absence flag (is_SS_IG)" indicates the validity and presence of the following in the IG stream: the left-eye IG stream entry; the right-eye IG stream entry; and the stream attributes. When the structure is absent in the stereoscopic PG stream, this field should be set to "0". When the structure is present in the stereoscopic PG stream, this field should be set to "1".

[0194] The "left-eye stream entry" includes: a sub-path identifier reference (ref_to_Subpath_id) specifying a sub-path to which the playback path of the left-eye IG stream

belongs; a stream file reference (ref_to_subClip_entry_id) specifying a stream file in which the left-eye IG stream is stored; and a packet identifier (ref_to_stream_PID_subclip) of the left-eye IG stream in this stream file.

[0195] The "right-eye stream entry" includes: a sub-path identifier reference (ref_to_Subpath_id) specifying a sub-path to which the playback path of the right-eye IG stream belongs; a stream file reference (ref_to_subClip_entry_id) specifying a stream file in which the right-eye IG stream is stored; and a packet identifier (ref_to_stream_PID_subclip) of the right-eye IG stream in this stream file. When the stream file referenced by the "stream_entry_for_dependent_view" in the stream registration information in the extension stream selection table is different from the stream file referenced by the stream entry in the basic stream selection table, a stream file storing the right-eye IG stream needs to be read.

[0196] The "common stream attribute" includes language attributes of the left-eye IG stream and the right-eye IG stream.

[0197] The "stereoscopic IG offset sequence ID reference information" is a reference to the offset sequence ID for the stereoscopic-type IG stream, and indicates the offset sequence for the offset metadata of the dependent-view video stream. The playback device should apply the offset, which is supplied by this field, to the stereoscopic-type IG plane.

[0198] When the field is an undefined value (FF), the playback device does not apply this offset to the IG plane.

[0199] The PG_text subtitle stream offset sequence reference information and the IG stream offset sequence reference information are written in the stream registration information in correspondence with stream numbers. Therefore, when the stream selection procedure is executed due to a change of the device state or occurrence of a request for stream change and a stream number corresponding to the language setting on the device side is set in the stream number register, an offset sequence indicated by a reference corresponding to the new stream number is supplied from the video decoder to the shift unit. With this structure, an optimum offset sequence corresponding to the language setting in the playback device is supplied to the shift unit, thus it is possible to set the depth of the graphics in 1 plane + offset mode to an optimum value corresponding to the language setting in the playback device.

[0200] The following describes restrictions for the extension stream selection table.

[0201] The stream entry in the stereoscopic dependent-view block should not change in the playlist.

[0202] When the type of the stream entry in the stereoscopic dependent-view block is the ES type (stream type = 2) that is used by the sub-path, the sub-path ID reference and the subclip entry ID reference (ref_to_subclip_entry_id) do not change in the playlist.

[0203] Only two types of elementary streams are permitted to be the types of the stream entry, stream entry for the base view, and stream entry for the dependent view. The two types are: ES (stream type = 1) in the AV clip used by the playitem; and ES (stream type = 2) in the AV clip used by the sub-path.

[0204] In the stereoscopic dependent-view block, the stream encoding method in the stream attribute is set to "0x20".

[0205] FIG. 14 shows what elementary streams are demultiplexed from the main TS and the sub-TSs with use of the basic stream selection table and the extension stream selection table.

[0206] The middle part of FIG. 14 shows the demultiplexing unit. The upper part of FIG. 14 shows the combination of the basic stream selection table and the extension stream selection table. The left side of FIG. 14 shows the main TS and the sub-TSs, and the right side of FIG. 14 shows the demultiplexed base-view video stream, dependent-view video stream, left-eye PG stream, right-eye PG stream, left-eye IG stream, right-eye IG stream, and primary audio stream.

[0207] FIG. 15 shows stream numbers to be assigned in the 2D output mode and the 3D output mode.

[0208] The vertical column on the left side of FIG. 15 shows the following stream numbers: primary video stream #1; primary audio streams #1 and #2; PG_text subtitle streams #1, #2 and #3; and IG streams #1 and #2.

[0209] The element streams arranged on the left side of FIG. 15, enclosed by a dotted line, are element streams that are targeted for demultiplexing only in the 2D output mode, and that are permitted by the stream selection table (STN_table) to be played back.

[0210] The element streams arranged on the right side of FIG. 15, enclosed by a dotted line, are element streams that are targeted for demultiplexing only in the 3D output mode, and that are permitted by the extension stream selection table (STN_table_SS) to be played back.

[0211] The element streams enclosed by the combined dotted lines of the left side and the right side are element streams that are targeted for demultiplexing in the 3D output modes.

[0212] With regard to the video stream #1, the MPEG4-MVC base-view video stream is enclosed by the combined dotted lines of the left side and the right side. This indicates that the MPEG4-MVC base-view video stream is targeted to be played back in both the 2D and the 3D output modes. On the other hand, the MPEG4-MVC dependent-view video stream is enclosed by only the dotted line of the right side. This indicates that the MPEG4-MVC dependent-view video stream is to be played back only in the 3D output mode.

[0213] With regard to the primary audio streams #1 and #2, they are both enclosed by the

combined dotted lines of the left side and the right side. This indicates that the audio streams #1 and #2 are targeted to be played back in both the 2D and the 3D output modes.

[0214] With regard to the PG_text subtitle streams, the PG_text subtitle streams #1 and #2 are 2D PG streams, and are enclosed by the combined dotted lines of the left side and the right side, indicating that they are targeted to be played back in both the 2D and the 3D output modes. On the other hand, the left-eye PG stream and the right-eye PG stream are enclosed by only the dotted line of the right side. This indicates that the left-eye PG stream and the right-eye PG stream are to be played back only in the 3D output mode.

[0215] With regard to the IG streams, the IG streams #1 and #2 are 2D IG streams, and are enclosed by the combined dotted lines of the left side and the right side. This indicates that IG streams #1 and #2 are targeted to be played back only in the 2D output mode. On the other hand, the left-eye IG stream and the right-eye IG stream are enclosed by only the dotted line of the right side. This indicates that the left-eye IG stream and the right-eye IG stream are to be played back in the 2D output mode and the 3D output mode.

[0216] As understood from the above description, in the 3D output mode, the dependent-view video stream is added to the target for playback regarding the stream type "video stream".

[0217] It is also understood that, in the 3D output mode, the left-eye PG stream and the right-eye PG stream are added to the target for playback regarding the stream type "PG stream", and the left-eye IG stream and the right-eye IG stream are added to the target for playback regarding the stream type "IG stream". The reason for adding the left-eye PG stream and the right-eye PG stream to the target for playback is that the left-eye PG stream and the right-eye PG stream are used to realize the stereoscopic playback in the 3D output mode. The reason for adding the left-eye IG stream and the right-eye IG stream to the target for playback is that the left-eye IG stream and the right-eye IG stream are used to realize the stereoscopic playback in the 3D output mode.

[0218] This completes the description of the recording medium. In the following, the playback device will be described in detail.

[0219] FIG. 16 shows the internal structure of the playback device. As shown in FIG 16, the playback device includes a reading unit 201, a memory 202, a register set 203, a decoder 204, a demultiplexing unit 205, a plane memory set 206, a shift unit 207, a layer overlay unit 208, a transmission/reception unit 209, and a playback control unit 210. The internal structure of FIG. 16 is composed of the minimum structural elements that are required to realize the playback device provided with a problem solving means. A more detailed internal structure will be described in a later embodiment.

[0220] The reading unit 201 reads out, from the recording medium, the index table, program file, playlist information file, stream information file, and stream file. When reading the stereoscopic interleaved stream file, the reading unit 201 performs a process in which it divides

the stereoscopic interleaved stream file into (i) an ATC sequence 1 corresponding to the main TS and (ii) an ATC sequence 2 corresponding to the sub-TS, by using (a) the extent start point information of the clip base information in the 3D clip information file and (b) the extent start point information in the clip dependent information, and stores the ATC sequences 1 and 2 into different read buffers. This division is realized by repeating two processes: the first process of extracting, from the stereoscopic interleaved stream file, as many source packets as the number of packets corresponding to the source packet number indicated by the extent start point information in the clip dependent information, and adding the extracted source packets into the ATC sequence 1; and the second process of extracting, from the stereoscopic interleaved stream file, as many source packets as the number of packets corresponding to the source packet number indicated by the extent start point information in the clip base information, and adding the extracted source packets into the ATC sequence 2.

[0221] The memory 202 stores a combined stream registration sequence that is obtained by combining the extension stream selection table and the basic stream selection table included in the playlist information.

[0222] The player number register 203 includes a plurality of registers that are required for the playback device to operate.

[0223] The decoder 204 is composed of a video decoder 211, a PG decoder 212, an IG decoder 214, and an audio decoder which correspond to respective stream types.

[0224] The demultiplexing unit 205 is provided with: a source depacketizer for converting the source packets into TS packets; and a PID filter for performing the packet filtering. The demultiplexing unit 205 converts source packets having packet identifiers written in stream entries of the basic stream selection table in the 3D playlist information into TS packets, and outputs the TS packets to the decoder. Also, the demultiplexing unit 207 converts source packets having packet identifiers written in stream entries of the stereoscopic stream selection table in the 3D playlist information into TS packets, and outputs the TS packets to the decoder. Which packet identifiers, among a plurality of packet identifiers written in a plurality of stream entries of the basic and stereoscopic stream selection tables, are to be used is determined in accordance with the setting in the stream number register among the player status registers. The stream number register is a register for storing the current stream number.

[0225] The plane memory set 206 is composed of a plurality of plane memories.

[0226] These plane memories constitute a layer model, and the data stored in each plane memory are used to overlay the layers with each other. The plane memory set includes a left-eye plane memory and a right-eye plane memory. Respective non-compressed picture data obtained by decoding the base-view and dependent-view components of each access unit are written into the left-eye and right-eye plane memories. The plane memory set includes a left-eye plane memory and a right-eye plane memory. Respective non-compressed picture data obtained by decoding the base-view and dependent-view components of each access unit are

written into the left-eye and right-eye plane memories. The writing is performed each time the playback start time indicated by the presentation time stamp of each access unit is reached.

[0227] To which of the left-eye plane memory and the right-eye plane memory the picture data after decoding is to be written is determined in accordance with the base-view indicator in the playlist information. When the base-view indicator specifies the base-view video stream as "for the left eye", the picture data of the base-view video stream is written to the left-eye plane memory, and the picture data of the dependent-view video stream is written to the right-eye plane memory.

[0228] When the base-view indicator specifies the base-view video stream as "for the right eye", the picture data of the base-view video stream is written to the right-eye plane memory, and the picture data of the dependent-view video stream is written to the left-eye plane memory. These view components are output to the display device in sequence. More specifically, in one frame period, the picture data stored in the left-eye plane memory and the picture data stored in the right-eye plane memory are output simultaneously.

[0229] The shift unit 207 shifts the pixel coordinates.

[0230] The layer overlay unit 208 overlays the layers in the plurality of plane memories.

[0231] The transmission/reception unit 209 transits to a data transfer phase via a mutual authentication phase and a negotiation phase, when playback device is connected with another device in the home theater system via an interface. The transmission/reception unit 209 performs data transfer in the transfer phase.

[0232] In the negotiation phase, the capabilities of the partner device (including the decode capability, playback capability, and display frequency) are grasped, and the capabilities are set in the player setting register, so that the transfer method for the succeeding data transfers is determined. The negotiation phase includes a mutual authentication phase in which each of the devices confirms the authenticity of the other device. After the negotiation phase, one line of the pixel data in the non-compression/plaintext format in the picture data after the layer overlaying is transferred to the display device at a high transfer rate in accordance with the horizontal sync period of the display device. On the other hand, in the horizontal and vertical blanking intervals, audio data in the non-compression/plaintext format is transferred to other devices (including an amplifier and a speaker as well as the display device) connected with the playback device. With this structure, the devices such as the display device, amplifier and speaker can receive the picture data and audio data both in the non-compression/plaintext format, and a reproduced output is achieved. Further, when the partner device has the decode capability, a pass-through transfer of the video and audio streams is possible. In the pass-through transfer, it is possible to transfer the video stream and audio stream in the compressed/encrypted format, as they are.

[0233] The playback control unit 210 executes a random access from an arbitrary time point

on the time axis of the video stream. More specifically, when it is instructed to play back from an arbitrary time point on the time axis of the video stream, the playback control unit 210 searches for a source packet number of an access unit corresponding to the arbitrary time point, by using a base entry map in the 3D stream information file and an extension entry map. The access unit includes a pair of a view component of the base-view video stream and a view component of the dependent-view video stream, and this searching identifies a source packet number of a source packet storing an access unit delimiter for the access unit. Reading from the source packet number and decoding enable a random access to be performed. When a 3D playlist is to be played back, random accesses to the main TS and the sub-TS are executed by using the in-time and the out-time defined in the main-path information and the in-time and the out-time defined in the sub-path information of the 3D playlist information, to start the playback of the playlist.

[0234] The video decoder 211 is a representative decoder among the decoders constituting the decoder set 204. The video decoder 211 preloads view components that constitute the dependent-view video stream, and decodes view components of a picture type for which the Instantaneous Decoder Refresh (IDR) at the start of the closed GOP in the base-view video stream is intended (IDR type). In this decoding, all the coded data buffers and decode data buffers are cleared. After decoding the view components of the IDR type in this way, (i) view components following the base-view video stream compress-encoded based on the correlation with these view components and (ii) view components of the dependent-view video stream, are decoded. Non-compressed picture data is obtained by this decoding of the view components. The obtained non-compressed picture data is stored in the decode data buffer to be used as the reference picture.

[0235] By using the reference picture, the motion compensation is performed onto (i) view components following the base-view video stream and (ii) view components of the dependent-view video stream. Non-compressed picture data with regard to (i) view components following the base-view video stream and non-compressed picture data with regard to (ii) view components of the dependent-view video stream are obtained by the motion compensation. The obtained non-compressed picture data are stored in the decode data buffer to be used as reference pictures. The above-described decoding is performed each time the decode start time indicated in the decode time stamp of each access unit is reached.

[0236] The following describes the PG decoder 212, text subtitle decoder 213, and IG decoder 214, and the internal structures of the streams that are to be decoded by these decoders.

[0237] For the PG stream: the decoder structure is "1 decoder + 1 plane" when the "1 plane + offset" method is adopted; and the decoder structure is "2 decoders + 2 planes" when the 3D-LR method is adopted.

[0238] Similarly, for the IG stream: the decoder structure is "1 decoder + 1 plane" when the "1 plane + offset" method is adopted; and the decoder structure is "2 decoders + 2 planes" when

the 3D-LR method is adopted.

[0239] For the text subtitle stream for which the 3D-LR method cannot be adopted: the decoder structure is "1 decoder + 1 plane" when the "1 plane + offset" method is adopted.

[0240] First, the internal structure of the PG stream, and the internal structure of the PG decoder for decoding the PG stream will be described.

[0241] Each of the left-eye PG stream and the right-eye PG stream includes a plurality of display sets. The display set is a set of functional segments that constitute one screen display. The functional segments are processing units that are supplied to the decoder while they are stored in the payloads of the PES packets which each have the size of approximately 2 KB, and are subjected to the playback control with use of the DTSs and PTSs.

[0242] The display set falls into the following types.

A. Epoch-start display set

[0243] The epoch-start display set is a set of functional segments that start the memory management by resetting the composition buffer, code data buffer, and graphics plane in the graphics decoder. The epoch-start display set includes all functional segments required for composition of the screen.

B. Normal-case display set

[0244] The normal-case display set is a display set that performs the composition of the screen while continuing the memory management of the composition buffer, code data buffer, and graphics plane in the graphics decoder. The normal-case display set includes functional segments that are differentials from the preceding display set.

C. Acquisition-point display set

[0245] The acquisition-point display set is a display set that includes all functional segments required for composition of the screen, but does not reset the memory management of the composition buffer, code data buffer, and graphics plane in the graphics decoder. The acquisition-point display set may include functional segments that are different from those in the previous display set.

D. Epoch-continue display set

[0246] The epoch-continue display set is a display set that continues the memory management of the composition buffer, code data buffer, and graphics plane in the playback device as it is when the connection between a playitem permitting the playback of the PG stream and a playitem immediately before the playitem is the "seamless connection" (CC = 5) that evolves a clean break. In this case, the graphics objects obtained in the object buffer and the graphics plane are kept to be present in the object buffer and the graphics plane, without being discarded.

[0247] Certain time points on the playback time axis of the STC sequence are assigned to the start point and end point of these display sets, and the same times are assigned to the left-eye view and to the right-eye view. Also, for the left-eye PG stream and the right-eye PG stream, the types of the display sets that are present on the same time point on the time axis are the same. That is to say, when the display set on the left-eye side is the epoch-start display set, the display set on the right-eye side that is at the same time point on the time axis of the STC sequence is the epoch-start display set.

[0248] Further, when the display set on the left-eye side is the acquisition-point display set, the display set on the right-eye side that is at the same time point on the time axis of the STC sequence is the acquisition-point display set.

[0249] Each display set includes a plurality of functional segments. The plurality of functional segments include the following.

(1) Object definition segment

[0250] The object definition segment is a functional segment for defining the graphics object. The graphics definition segment defines the graphics object by using a code value and a run length of the code value.

(2) Pallet definition segment

[0251] The pallet definition segment includes pallet data that indicates correspondence among each code value, brightness, and red color difference/blue color difference. The same correspondence among the code value, brightness, and color difference is set in both the pallet definition segment of the left-eye graphics stream and the pallet definition segment of the right-eye graphics stream.

(3) Window definition segment

[0252] The window definition segment is a functional segment for defining a rectangular frame called "window" in the plane memory that is used to extend the non-compressed graphics object onto the screen. The drawing of the graphics object is restricted to the inside of the plane memory, and the drawing of the graphics object is not performed outside the window.

[0253] Since a part of the plane memory is specified as the window for displaying the graphics, the playback device does not need to perform the drawing of the graphics for the entire plane. That is to say, the playback device only needs to perform the graphics drawing onto the window that has a limited size. The drawing of the part of the plane for display other than the window can be omitted. This reduces the load of the software on the playback device side.

(4) Screen composition segment

[0254] The screen composition segment is a functional segment for defining the screen composition using the graphics object, and includes a plurality of control items for the composition controller in the graphics decoder. The screen composition segment is a functional segment that defines in detail the display set of the graphics stream, and defines the screen composition using the graphics object. The screen composition falls into the types such as Cut-In/-Out, Fade-In/-Out, Color Change, Scroll, and Wipe-In/-Out. With use of the screen composition defined by the screen composition segment, it is possible to realize display effects such as deleting a subtitle gradually, while displaying the next subtitle.

(5) End segment

[0255] The end segment is a functional segment that is located at the end of a plurality of functional segments belonging to one display set. The playback device recognizes a series of segments from the screen composition segment to the end segment as the functional segments that constitute one display set.

[0256] In the PG stream, the start time point of the display set is identified by the DTS of the PES packet storing the screen composition segment, and the end time point of the display set is identified by the PTS of the PES packet storing the screen composition segment.

[0257] The left-eye graphics stream and the right-eye graphics stream are packetized elementary streams (PES). The screen composition segment is stored in the PES packet. The PTS of the PES packet storing the screen composition segment indicates the time when the display by the display set to which the screen composition segment belongs should be executed.

[0258] The value of the PTS of the PES packet storing the screen composition segment is the

same for both the left-eye video stream and the right-eye video stream.

(Decoder models of PG decoder)

[0259] The PG decoder includes: a "coded data buffer" for storing functional segments read from the PG stream; a "stream graphics processor" for obtaining a graphics object by decoding the screen composition segment; an "object buffer" for storing the graphics object obtained by the decoding; a "composition buffer" for storing the screen composition segment; and a "composition controller" for decoding the screen composition segment stored in the composition buffer, and performing a screen composition on the graphics plane by using the graphics object stored in the object buffer, based on the control items included in the screen composition segment.

[0260] A "transport buffer" for adjusting the input speed of the TS packets constituting the functional segments is provided at a location before the graphics plane.

[0261] Also, at locations subsequent to the graphics decoder, a "graphics plane", a "CLUT unit" for converting the pixel codes constituting the graphics object stored in the graphics plane into values of brightness/color difference based on the pallet definition segment, and a "shift unit" for the plane shift are provided.

[0262] The pipeline in the PG stream makes it possible to simultaneously executes the following processes: the process in which the graphics decoder decodes an object definition segment belonging to a certain display set and writes the graphics object into the graphics buffer; and the process in which a graphics object obtained by decoding an object definition segment belonging to a preceding display set is written from the object buffer to the plane memory.

[0263] FIGs. 17A and 17B show the internal structure of the PG decoder. FIG. 17A shows a decoder model for displaying data in the 1 plane + offset mode. FIG. 17B shows a decoder model for displaying data in the LR mode.

[0264] In FIGs. 17A and 17B, the PG decoder itself is represented by a frame drawn by the solid line, and a portion that follows the graphics decoder is represented by a frame drawn by the chain line.

[0265] FIG. 17A shows that the PG decoder has "1 decoder" structure, and the graphics plane has "1 plane" structure. However, the output of the graphics plane branches to the left-eye output and the right-eye output. Thus the left-eye output and the right-eye output are each provided with a shift unit.

[0266] FIG. 17B shows that two series of "transport buffer" - "PG decoder" - "graphics plane" - "CLUT unit" are provided so that the left-eye stream and the right-eye stream can be

processed independently.

[0267] The offset sequence is contained in the right-eye video stream. Thus, in the plane offset format, the PG decoder has "1 decoder" structure, and the output from the PG decoder is supplied to the left-eye view and the right-eye view by switching therebetween.

[0268] The PG decoder performs the following to switch between 2D and 3D.

1. 1. The mutual switching between the 1 plane + offset mode and the 2D mode is performed seamlessly. This is realized by invalidating the "Offset".
2. 2. When switching between the 3D-LR mode and the 2D mode is performed, the display of the subtitle temporarily disappears because the switching between the modes requires switching between PIDs. This is the same as the switching between streams.

[0269] This completes the explanation of the PG decoder. In the following, the text subtitle decoder will be described in detail.

(Decoder models of text subtitle decoder)

[0270] The text subtitle decoder is composed of a plurality of pieces of subtitle description data.

[0271] The text subtitle decoder includes: a "subtitle processor" for separating the text code and the control information from the subtitle description data; a "management information buffer" for storing the text code separated from the subtitle description data; a "text render" for extending the text code in the management information buffer to the bit map by using the font data; an "object buffer" for storing the bit map obtained by the extension; and a "drawing control unit" for controlling the text subtitle playback along the time axis by using the control information separated from the subtitle description data.

[0272] The text subtitle decoder is preceded by: a "font preload buffer" for preloading the font data; a "TS buffer" for adjusting the input speed of the TS packets constituting the text subtitle stream; and a "subtitle preload buffer" for preloading the text subtitle stream before the playback of the playitem.

[0273] The graphics decoder is followed by a "graphics plane"; a "CLUT unit" for converting the pixel codes constituting the graphics object stored in the graphics plane into values of brightness and color difference based on the pallet definition segment; and a "shift unit" for the plane shift.

[0274] FIGs. 18A and 18B show the internal structure of the text subtitle decoder. FIG 18A

shows a decoder model of the text subtitle decoder in the 1 plane + offset mode. FIG 18B shows a decoder model of the text subtitle decoder in the 3D-LR method. In FIGs. 18A and 18B, the text subtitle decoder itself is represented by a frame drawn by the solid line, a portion that follows the text subtitle decoder is represented by a frame drawn by the chain line, and a portion that precedes the text subtitle decoder is represented by a frame drawn by the dotted line.

[0275] FIG. 18A shows that the output of the graphics plane branches to the left-eye output and the right-eye output, and that the left-eye output and the right-eye output are each provided with a shift unit.

[0276] FIG. 18B shows that the left-eye graphics plane and the right-eye graphics plane are provided, and that the bit map extended by the text subtitle decoder is written into the graphics planes.

[0277] The text subtitle stream differs from the PG stream as follows. That is to say the font data and the character code are sent, not the graphics data is sent as the bit map, so that the rendering engine generates the subtitle. Thus the stereoscopic viewing of the subtitle is realized in the 1 plane + offset mode.

[0278] This completes the description of the text subtitle stream and the text subtitle decoder. Next, the internal structure of the IG stream and the structure of the IG decoder will be described.

(IG stream)

[0279] Each of the left-eye IG stream and the right-eye IG stream includes a plurality of display sets. Each display set includes a plurality of functional segments. As is the case with the PG stream, the display set falls into the following types. epoch-start display set, normal-case display set, acquisition-point display set, and epoch-continue display set.

[0280] The plurality of functional segments belonging to these display sets include the following types.

(1) Object definition segment

[0281] The object definition segment of the IG stream is the same as that of the PG stream. However, the graphics object of the IG stream defines the in-effect and out-effect of pages, the normal, selected, and active states of the button members. The object definition segments are grouped into those that define the same state of the button members, and those that constitute the same effect image. The group of object definition segments defining the same state is

called a "graphics data set".

(2) Pallet definition segment

[0282] The pallet definition segment of the IG stream is the same as that of the PG stream.

(3) Interactive control segment

[0283] The interactive control segment includes a plurality of pieces of page information. The page information is information that defines a screen composition of the multi-page menu. Each piece of page information includes an effect sequence, a plurality of pieces of button information, and a reference value of a pallet identifier.

[0284] The button information is information that realizes an interactive screen composition on each page constituting the multi-page menu by displaying the graphics object as one state of a button member.

[0285] The effect sequence constitutes the in-effect or the out-effect with use of the graphics object, and includes effect information, where the in-effect is played back before a page corresponding to the page information is displayed, and the out-effect is played back after the page is displayed.

[0286] The effect information is information that defines each screen composition for playing back the in-effect or the out-effect. The effect information includes: a screen composition object that defines a screen composition to be executed in the window (partial region) defined by the window definition segment on the graphics plane; and effect period information that indicates a time interval between the current screen and the next screen in the same region.

[0287] The screen composition object in the effect sequence defines a control that is similar to the control defined by the screen composition segment of the PG stream. Among the plurality of object definition segments, an object definition segment that defines the graphics object used for the in-effect is disposed at a location that precedes an object definition segment that defines the graphics object used for the button member.

[0288] Each piece of button information in the page information is information that an interactive screen composition on each page constituting the multi-page menu by displaying the graphics object as one state of a button member. The button information includes a set button page command that, when a corresponding button member becomes active, causes the playback device to perform the process of setting a page other than the first page as the current page.

[0289] To make it possible for the offset in the plane shift to be changed for each page during playback of the IG stream, a navigation command for changing the offset is incorporated into the button information, and the "auto-activate" of the navigation command is defined in the corresponding piece of button information, in advance. This makes it possible to change automatically the value or direction of the offset defined in the stream registration information of the IG stream.

(4) End segment

[0290] The end segment is a functional segment that is located at the end of a plurality of functional segments belonging to one display set. A series of segments from the interactive control segment to the end segment are recognized as the functional segments that constitute one display set.

[0291] The following are the control items of the interactive control segment that are the same for both the left-eye graphics stream and the right-eye graphics stream: button adjacency information; selection time-out time stamp; user time-out duration; and composition time-out information.

1. Button adjacency information

[0292] The button adjacency information is information that specifies a button to be changed to the selected state when a key operation specifying any of upward, downward, leftward, and rightward is performed while a certain button adjacent to the specified button is in the selected state.

2. Selection time-out time stamp

[0293] The selection time-out time stamp indicates a time-out time that is required to automatically activate a button member in the current page and cause the playback device to execute the button member.

3. User time-out duration

[0294] The user time-out duration indicates a time-out time that is required to return the current page to the first page so that only the first page is displayed.

4. Composition time-out information

[0295] The composition time-out information indicates a time period that is required to end an interactive screen display by the interactive control segment. With respect to the IG stream, the start time point of a display set is identified by the DTS of the PES packet storing the interactive control segment, and the end time point of the display set is identified by the composition time-out time of the interactive control segment. The same DTS and the same composition time-out time are set for both the left eye and the right eye.

(Decoder models of IG decoder)

[0296] The IG decoder includes: a "coded data buffer" for storing functional segments read from the IG stream; a "stream graphics processor" for obtaining a graphics object by decoding the screen composition segment; an "object buffer" for storing the graphics object obtained by the decoding; a "composition buffer" for storing the screen composition segment; and a "composition controller" for decoding the screen composition segment stored in the composition buffer, and performing a screen composition on the graphics plane by using the graphics object stored in the object buffer, based on the control items included in the screen composition segment.

[0297] A "transport buffer" for adjusting the input speed of the TS packets constituting the functional segments is provided at a location before the graphics plane.

[0298] Also, at locations after the graphics decoder, a "graphics plane", a "CLUT unit" for converting the pixel codes constituting the graphics object stored in the graphics plane into values of brightness/color difference based on the pallet definition segment, and a "shift unit" for the plane shift are provided.

[0299] FIGs. 19A and 19B show decoder models of the IG decoder. In FIGs. 19A and 19B, the IG decoder itself is represented by a frame drawn by the solid line, a portion that follows the graphics decoder is represented by a frame drawn by the chain line, and a portion that precedes the IG decoder is represented by a frame drawn by the dotted line. FIG. 19A shows a decoder model for displaying the 2D-format IG stream in the LR format in the 1 plane + offset mode. FIG. 19B shows a decoder model of the IG stream for displaying LR-format data.

[0300] These decoders include a circuit for reflecting values of system parameters onto the offsets so that the program can control the depth information of the menu graphics.

[0301] FIG. 19B shows a two-decoder model that enables the offset values to be changed with use of a command. Accordingly, in this decoder model, the depth information of the menu can be changed by the command. Note that different offset values may be set for the left view and the right view. On the other hand, in the depth method, the offset is invalid.

[0302] The composition controller in the graphics decoder realizes the initial display of the interactive screen by displaying the current button, among a plurality of button members in the interactive screen, by using the graphics data of the graphics data set corresponding to the selected state, and displaying the remaining buttons by using the graphics data set corresponding to the normal state.

[0303] When a user operation specifying any of upward, downward, leftward, and rightward is performed, it writes, into the button number register, a number of a button member that is present in the direction specified by the user operation among a plurality of button members in the normal state and adjacent to the current button, the writing causing the button member having become newly the current button to change from the normal state to the selected state.

[0304] In the interactive screen, when a user operation for changing the button member from the selected state to the active state is performed, the interactive screen is updated by extracting the graphics data constituting the active state from the graphics data set and displaying the extracted graphics data.

[0305] The update of the interactive screen should be executed in common to the left-eye view and the right-eye view. Thus it is preferable that the left-eye graphics decoder and the right-eye graphics decoder have in common a composition controller for the two-decoder model.

[0306] In the above-described case, the inter-changing is realized by using the same navigation command for both the left-eye view and the right-eye view of the stereoscopic IG stream, and setting the same button structure for both the 3D graphics object and the 2D graphics object.

[0307] When switching between the 2D IG stream and the stereoscopic IG stream, it is possible to change only the displayed graphics object when the attribute and number and the like of the navigation command and button information are the same for both. Switching from the 3D-LR mode to the display of only the L image can be made without reloading, but there is a possibility that the display position may be shifted. It is preferable that the playback device performs the switching based on a flag set to indicate which is adopted by the title producer.

[0308] The following are notes on switching between modes.

- Reloading does not occur when switching between the 1 plane + offset mode and the 2D mode is performed. This is because the IG stream does not need to be reloaded, and only invalidation of the offset is required.
- Reloading occurs when switching between the 3D-LR mode and the 2D mode is performed. This is because the streams are different.

[0309] This completes the description of the IG stream and the IG decoder. Next, the plane

memory will be described in detail.

[0310] The following describes the plane memory structure in the 1 plane + offset mode.

[0311] The layer overlaying in the plane memory is achieved by executing a superimposing process onto all combinations of the layers in the layer model. In the superimposing process, pixel values of pixel data stored in the plane memories of the two layers are superimposed. The layer overlaying by the layer overlay unit 208 is achieved by executing a superimposing process onto all combinations of two layers among the layers in the layer model. In the superimposing process, pixel values of pixel data stored in the plane memories of the two layers are superimposed in the layer model of the plane memory.

[0312] The superimposing between layers is performed as follows. A transmittance α as a weight is multiplied by a pixel value in unit of a line in the plane memory of a certain layer, and a weight of $(1 - \text{transmittance } \alpha)$ is multiplied by a pixel value in unit of a line in the plane memory of a layer below the certain layer. The pixel values with these brightness weights are added together. The resultant pixel value is set as a pixel value in unit of a line in the layer. The layer overlaying is realized by repeating this superimposing between layers for each pair of corresponding pixels in a unit of a line in adjacent layers in the layer model.

[0313] A multiplication unit for multiplying each pixel value by the transmittance to realize the layer overlaying and an addition unit for adding up the pixels are provided at locations subsequent to the plane memory, as well as the above-described CLUT unit, shift unit and the like.

[0314] FIG. 20 shows a circuit structure for overlaying the outputs of the decoder models and outputting the result in the 3D-LR mode. In FIG. 20, the layer models each composed of video plane, PG plane, and IG plane are enclosed by solid lines, and portions that follow the plane memories are enclosed by chain lines. As shown in FIG 20, there are two above-described layer models. Also, there are two portions following the plane memories.

[0315] With the plane memory structure for the 3D-LR method which is provided with two pairs of a layer model and a portion following the plane memory, two pairs of the video plane, PG plane, and IG plane are provided for the left-eye view and the right-eye view, and the outputs from each plane memory are overlaid, as the layer overlaying, separately for the left-eye view and the right-eye view.

[0316] FIG. 21 shows a circuit structure for overlaying the outputs of the decoder models and outputting the result in the 1 plane + offset mode.

[0317] In FIG. 21, the layer model composed of the left-eye and right-eye video planes, PG plane, and IG plane is encircled by the solid line, and a portion that follows the plane memory is encircled by the chain line. As shown in FIG 21, there is only one above-described layer model. Also, there are two portions following the plane memory.

[0318] In the 1 plane + offset mode: the video plane is provided, one for each of the left-eye view and right-eye view; and each of the PG plane and the IG plane is provided, one for both the left view and the right view. A PG plane and an IG plane each are not separately prepared for each of the left-eye view and the right-eye view. There is only one plane memory for both the left-eye view and the right-eye view. With this structure, the above-described layer overlaying is performed onto the left-eye and right-eye outputs.

[0319] FIG. 22 shows the circuit structure for overlaying data output from the decoder model and outputting the overlaid data in the upper end 2D subtitle playback mode and the lower end 2D subtitle playback mode.

[0320] In the upper end 2D subtitle playback mode method and the lower end 2D subtitle playback mode, a video plane is prepared for each of the left-eye view and the right-eye view. A PG plane and an IG plane each are not separately prepared for each of the left-eye view and the right-eye view. There is only one plane memory for both the left-eye view and the right-eye view. In accordance with the setting of the register (PSR32), which is described later, indicating the video shift mode of the playback device, pixel shift of the video plane is performed upward or downward by 131 pixels for each of the left-eye view and the right-eye view. Then, layer overlaying is performed on the left-eye output and the right-eye output.

[0321] The playback device needs to support all of the 3D-LR mode, the 1 plane + offset mode, the upper end 2D subtitle playback mode, and the lower end 2D subtitle playback mode. Thus the hardware structure of the playback device is basically "2 decoders + 2 planes". When the mode switches to either of the 1 plane + offset mode, the 2D output mode, the upper end 2D subtitle playback mode, and the lower end 2D subtitle playback mode, the playback device becomes to have the "1 decoder + 1 plane" structure, invalidating one of the two pairs of "1 decoder + 1 plane".

[0322] It is at the discretion of the manufacturer of the playback device which of 1-decoder structure and 2-decoder structure is adopted as the decoder model and which of 1-plane structure and 2-planes structure is adopted as the plane model. Of course, the playback device may be designed to have the 2-decoder and 2-plane structure, then it may be set to be able to play back the stereoscopic PG and IG as the top-of-the-line product, and may be set not to be able to play back the stereoscopic PG and IG as the lower-cost product. This expands the lineup. Such a configuration having the capability to play back the stereoscopic PG or a configuration having the capability to play back the stereoscopic IG exists in the register set.

[0323] The following explains the register set.

[0324] The register set is composed of a plurality of player status registers and a plurality of player setting registers. Each of the player status registers and player setting registers is a 32-bit register and is assigned with a register number so that a register to be accessed is identified by the register number.

[0325] The bit positions of the bits (32 bits) that constitute each register are represented as "b0" through "b31". Among these, bit "b31" represents the highest-order bit, and bit "b0" represents the lowest-order bit. Among the 32 bits, a bit sequence from bit "bx" to bit "by" is represented by [bx:by].

[0326] The value of an arbitrary bit range [bx:by] in a 32-bit sequence stored in the player setting register/player status register of a certain register number is treated as an environment variable (also called "system parameter" or "player variable") that is a variable of an operation system in which the program runs. The program that controls the playback can obtain a system parameter via the system property or the application programming interface (API). Also, unless otherwise specified, the program can rewrite the values of the player setting register and the player status register. For the program based on an object-oriented programming language to do this, the program needs to have the authority to obtain or rewrite system parameters.

[0327] The player status register is a hardware resource for storing values that are to be used as operands when the MPU of the playback device performs an arithmetic operation or a bit operation. The player status register is also reset to initial values when an optical disc is loaded, and the validity of the stored values is checked. The values that can be stored in the player status register are the current title number, current playlist number, current playitem number, current stream number, current chapter number, and so on. The values stored in the player status register are temporary values because the player status register is reset to initial values each time an optical disc is loaded. The values stored in the player status register become invalid when the optical disc is ejected, or when the playback device is powered off.

[0328] The player setting register differs from the player status register in that it is provided with power handling measures. With the power handling measures, the values stored in the player setting register are saved into a non-volatile memory when the playback device is powered off, and the values are restored when the playback device is powered on. The values that can be set in the player setting register include: various configurations of the playback device that are determined by the manufacturer of the playback device when the playback device is shipped; various configurations that are set by the user in accordance with the set-up procedure; and capabilities of a partner device that are detected through negotiation with the partner device when the device is connected with the partner device.

[0329] FIG 23 shows the internal structures of the register set 203 and the playback control unit.

[0330] The left side of FIG. 23 shows the internal structures of the register set 203, and the right side shows the internal structures of the playback control unit.

[0331] The following describes the player status registers and the player setting registers assigned with respective register numbers.

[0332] PSR0 is a stream number register for the IG stream, and stores a current IG stream number.

[0333] PSR2 is a stream number register for the PG stream, and stores a current PG stream number.

[0334] PSR24 is used for the setting of "Player Capability for 3D". This indicates whether or not the playback device has a capability to perform the stereoscopic playback.

[0335] On the other hand, the playback control unit includes a stream selection procedure for determining a unique current PG stream number and a unique current IG stream number in the current playlist, by referring to the PSR24 in the register set 203 and the stream selection table of the current playlist information in the memory. The stream selection procedure includes "Initialization" and "Procedure when playback condition changed".

[0336] FIG. 25 shows the bit assignment in PSR32. The PSR32 indicates the video shift mode of the playback device. The value of the PSR 32 is set via an API of a BD program, a command, or the like. Also, video_shift_mode relating to a current PG stream selected by stream switching is acquired from stream registration information included in the stream selection table stream, and is set.

[0337] FIG. 24 shows the bit assignment in PSR24. PSR24 indicates the 3D capability of the playback device. The program recorded on the recording medium cannot change the value of PSR24.

[0338] The bit "b0" in PSR24 represents the video display capability of stereoscopic 1280×720 50p. More specifically, when bit "b0" is set to "0", it indicates that the playback device does not have the processing capability to display the 1280×720/50Hz progressive video; and when bit "b0" is set to "1", it indicates that the playback device has the processing capability to display the 1280×720/50Hz progressive video.

[0339] The bit "b2" in PSR24 represents the stereoscopic PG capability. More specifically, when bit "b2" is set to "0", it indicates that the playback device does not have the capability to play back the stereoscopic PG; and when bit "b2" is set to "1", it indicates that the playback device has the capability to play back the stereoscopic PG.

[0340] The bit "b3" in PSR24 represents the stereoscopic IG capability. More specifically, when bit "b3" is set to "0", it indicates that the playback device does not have the capability to play back the stereoscopic IG; and when bit "b3" is set to "1", it indicates that the playback device has the capability to play back the stereoscopic IG.

[0341] The bit "b5" in PSR24 represents the BD-J capability in the 3D output mode. More specifically, when bit "b5" is set to "1", it indicates that the playback device can process the BD-J mode in the 3D output mode; and when bit "b5" is set to "0", it indicates that the playback

device cannot process the BD-J mode in the 3D output mode. The use of the bit "b5" in PSR24 is not related to the subject of the present embodiment, and thus will be described in some subsequent embodiment.

[0342] As described above, PSR24 can be set to indicate whether or not the stereoscopic playback is available for each of the IG and PG. This makes it possible to provide: a configuration in which each of the IG and PG decoders is composed of two decoders and the playback device supports the stereoscopic playback for both IG and PG, or a configuration in which each of the IG and PG decoders is composed of two decoders and the playback device supports the stereoscopic playback for only PG and the 1 plane + offset mode for IG, or a converse configuration in which each of the IG and PG decoders is composed of two decoders and the playback device supports the stereoscopic playback for only IG and the 1 plane + offset mode for PG.

[0343] Furthermore, to sell the playback device as a lower-cost product, it is possible to provide a configuration in which although each of the IG and PG decoders is composed of two decoders, the playback device supports merely the 1 plane + offset mode for each of IG and PG. In this way, while having a common configuration in which each of the IG and PG decoders is composed of two decoders, the present embodiment makes it possible to determine whether to support the stereoscopic playback for each of IG and PG separately, depending on the grade of the product. This expands lineup of products of the playback device that the manufacturer can provide.

[0344] Also, when each of or both of the IG and PG decoders is composed of one decoder, it clearly indicates the availability of the stereoscopic playback. Accordingly, even if the playlist can be played back in a stereoscopic mode, it is possible to prevent the playback type from being set to the stereoscopic PG or stereoscopic IG erroneously.

[0345] The playback control having been described up to now can be realized by causing a computer to execute a program which is generated by writing the processing procedure represented by the flow charts of FIGs. 26 through 32 in an object-oriented compiler language.

[0346] FIG. 26 shows the playlist playback procedure. In this flow chart, the current playitem number is set to "1" in step S1, and then the control enters a loop in which the steps S2 to S6 are repeated. In this loop, the steps are performed as follows. The stream number is determined by the "procedure when playback condition is changed" (step S2). A stream file storing an elementary stream corresponding to the stream number is opened, and the source packet sequence is read therefrom (step S3). It is instructed that a source packet, among those constituting the source packet sequence, that corresponds to the stream number should be demultiplexed (step S4). The decoder is instructed to play back the read source packet for the period from the in-time to the out-time of the playitem, and for the period from the in-time to the out-time of the sub-playitem (step S5). These steps constituting the loop are repeated until the current playitem number becomes the last number. When it is judged that the current playitem number is not the last number (NO in step S6), the current playitem number is

incremented, and the control moves to step S2.

[0347] At this timing, step S7 is performed to judge whether or not there has been a stream selection request. When it is judged that there has been a stream selection request, the "procedure when playback condition is changed" is executed, with the requested stream number being regarded as "x" (step S8). When it is judged that the current playitem number is the last number (YES in step S6), the process ends.

<Determination of current PG stream and playback type thereof>

[0348] A current PG_text subtitle stream whose stream number is to be stored in PSR2 is selected based on the output mode (PSR22), stereoscopic PG capability in PSR24, and "is_SS_PG".

[0349] FIG. 27 is a flow chart showing the procedure of the "procedure when playback condition is changed" for the PG_text subtitle stream. Among the steps shown in this flow chart, the process of steps S11 to S22 is common to the 3D output mode and the 2D output mode, and the process of steps S23 to S28 is unique to the 3D output mode.

[0350] In step S11, the current PG_text subtitle stream number is obtained from PSR2. In step S12, it is judged whether the current PG_text subtitle stream number is of PG (YES) or of text subtitle stream number (NO). In step S13, it is checked whether or not the PG stream corresponding to the current PG_text subtitle stream number satisfies conditions (A) and (B).

[0351] Here, the conditions (A) and (B) are defined as follows.

[0352] Condition (A): The playback device has a capability to decode a PG stream that is identified by the current PG_text subtitle stream number.

[0353] Condition (B): The playback device has a capability to play back the specified language.

[0354] On the other hand, in step S14, it is checked whether or not the text subtitle stream corresponding to the current PG_text subtitle stream number satisfies conditions (A) and (B).

[0355] Condition (A): The playback device has a capability to extend the character code of the text subtitle stream, which is identified by the current PG_text subtitle stream number, into a bit map. This playback capability is indicated in PSR30 in the PSR set 203.

[0356] Condition (B): the playback device has a capability to support characteristics of the language of the text subtitle stream identified by the current PG_text subtitle stream number.

[0357] It should be noted here that, for a playback device to "be able to decode" a text subtitle

stream which represents the subtitle of a language, the playback device should have the capability to extend the text subtitle stream of the language into the bit map, and the capability to support characteristic of the language.

[0358] Here, this will be considered by taking examples of English, Japanese, and Arabic. As for subtitle displays in English, the language characteristics of English are judged to be supported only when the functions of "horizontal writing", "kerning", "double letter /logotype" are all supported.

[0359] As for subtitle displays in Japanese, the language characteristics of Japanese are judged to be supported only when the functions of "horizontal writing" "vertical writing" "prohibit line breaks after certain characters", "characters in smaller size" are all supported.

[0360] As for subtitle displays in Arabic, the language characteristics of Arabic are judged to be supported only when the functions of "rendering from the right to the left" and "double letter/logotype" are all supported.

[0361] When the playback device has the capability to extend the text subtitle stream of a language into the bit map and has the capability to support characteristics of the language, it can be said that the above-described conditions (A) and (B) are satisfied. When the playback device has the capability to extend the text subtitle stream of a language into the bit map, but not the capability to support characteristic of the language, it can be said that the condition (B) is not satisfied, but only the condition (A) is satisfied.

[0362] The capability to support characteristics of language is set for each language in bits constituting PSR48 through PSR61 in the register set. More specifically, PSR48 through PSR61 have flags that correspond to respective 3-byte language codes defined in ISO 639-2/T. Each of the flags is set to indicate whether or not the playback device has a capability to display a text subtitle of a language code that corresponds to the flag.

[0363] Among the 3-byte language codes defined in ISO 639-2/T, a 3-byte language code called "ita" indicates Italian, and a 3-byte language code called "jpn" indicates Japanese. Also, a 3-byte language code called "jav" indicates Japanese. Approximately 430 languages are covered by the 3-byte language codes defined in ISO 639-2/T. The flags in PSR48 through PSR61 are referred to when, to determine the current PG_text subtitle stream, it is judged whether or not the text subtitle stream written in the stream number table can be decoded. With this structure, it is possible to perform appropriately the judgment on whether a text subtitle stream can be decoded, even if the text subtitle stream is of a minor language.

[0364] After the above-described judgments, the control proceeds to step S15 in which it is judged whether or not the playback device satisfies a condition (Z).

[0365] Here, the condition (Z) is that the user is intending to play back a subtitle of an unsupported language, wherein the "unsupported language" is a language whose

characteristics are not supported. The intention is indicated in PSR30 in the register set.

[0366] The control then proceeds to step S16 in which it is judged whether or not the number of PG_text subtitle streams in the stream selection table of the current playitem is "0". When the stream selection table indicates that no PG_text subtitle stream has been permitted to be played back, the PG_text subtitle stream number stored in PSR2 is maintained (step S17).

[0367] When the stream selection table indicates at least one PG_text subtitle stream that is permitted to be played back, the control proceeds to step S18 to check for the validity of the current PG_text subtitle stream. In step S 18, it is judged whether or not the current PG_text subtitle stream number is equal to or greater than the total number of stream entries in the stream selection table and conditions (A) and (B) are satisfied.

[0368] When the result of judgment in step S18 is negative, the control proceeds to step S20 in which it is judged whether or not the current PG_text subtitle stream number is equal to or greater than the total number of stream entries in the stream selection table and conditions (A) and (Z) are satisfied. When the result of judgment in step S20 is affirmative, the value in PSR2 is maintained since it is determined that, although a PG_text subtitle stream number of a text subtitle of an unsupported language is set in PSR2, the user is intending to play back a subtitle of the unsupported language (step S21). When the result of judgment in step S20 is negative, an optimum stream for the current playitem is selected (step S22).

[0369] The steps S23 through S28 following this are unique to the 3D output mode. Specifically, in the 3D output mode, determination processing of upper or lower end playback type is firstly performed (step S23). In the determination processing of upper or lower end playback type, when the playback type is neither set as the upper end 2D subtitle playback type nor the lower end 2D subtitle playback type (step S24: No), is SS PG of a PG stream identified by a PG stream number of a PSR2 is acquired from stream registration information included in the stream selection table (step 25). Then, judgment is performed on whether a flag of the acquired is_SS_PG indicates "1" and whether stereoscopic PG capability of b2 in PSR24 indicates "1" (step 26). If a result of the judgment in step S26 is Yes, the playback type is set as the stereoscopic PG in which a left-eye PG stream and a right-eye PG stream (step S27).

[0370] When the playback type is set as the stereoscopic PG, the stereoscopic playback is performed by using packet identifier references that are included in the left-eye and right-eye stream entries of a piece of stream registration information corresponding to the current stream number stored in PSR2, among a plurality of pieces of stream registration information in the extension stream selection table. More specifically, the demultiplexing unit is caused to demultiplex TS packets whose packet identifiers are indicated by the packet identifier references that are included in the left-eye and right-eye stream entries of a piece of stream registration information corresponding to the current stream number stored in PSR2.

[0371] When the judgment result in step S26 is NO, the playback type is set as "1 plane +

offsetPG" (step S28). When the playback type is set as "1 plane + offsetPG", the PG playback in the 1 plane + offset mode is executed by using an offset sequence indicated by the PG_text subtitle stream offset sequence ID reference information in a piece of stream registration information corresponding to the current stream number stored in PSR2, among a plurality of pieces of stream registration information in the extension stream selection table.

[0372] Here the offset sequence is explained. A plurality of offset sequences to be used in the 1 plane + offset mode exist in the video access unit of the dependent-view video stream.

[0373] The video access unit of the dependent-view video stream is structured as a sequence of a video access unit delimiter, a sequence parameter set, a picture parameter set, an MVC scalable nesting SEI message, a first view component, a sequence end code, and a stream end code. The MVC scalable nesting SEI message includes a user data container. The user data container is unregistered user data, and falls into three types: closed caption information; GOP structure map; and offset metadata. One of these types is indicated by the "type_indicator" in the user data container.

[0374] The offset metadata is a sequence list for the PG plane, IG plane, and BD-J plane, and is used for the offset setting while the presentation graphics, text subtitle, and IG/BD-J plane are played back in the 1 plane + offset mode. More specifically, the offset metadata indicates the offset control on the PG plane, IG plane, and BD-J plane when the graphics to be overlaid with the picture data is played back in the 1 plane + offset mode.

[0375] The offset metadata should be stored in the MVC scalable nesting SEI message in the starting video component of each GOP in the encoding order of the dependent-view access unit. The offset metadata contains the above-described plurality of offset sequences. The offset sequence is a parameter sequence that indicates control parameters for each frame period in a group of pictures, where the control parameters are used when the graphics are overlaid with each piece of picture data belonging to the group of pictures. The offset sequence is composed of as many control parameters as the number indicated by the "number_of_displayed_frames_in_GOP". The control parameter is composed of plane offset direction information and a plane offset value.

[0376] The plane offset direction information ("Plane_offset_direction") indicates the direction of offset in the plane. When the plane offset direction information is set to a value "0", it indicates the front setting in which the plane memory exists between the TV and the viewer, and in the left-view period, the plane is shifted rightward, and in the right-view period, the plane is shifted leftward. When the plane offset direction information is set to a value "1", it indicates the behind setting in which the plane memory exists behind the TV or the screen, and in the left-view period, the plane is shifted leftward, and in the right-view period, the plane is shifted rightward. When the plane offset direction information indicates the front setting, the Z-axis coordinate of the control parameter in the three-dimensional coordinate system is a positive coordinate. When the plane offset direction information indicates the behind setting, the Z-axis coordinate of the control parameter in the three-dimensional coordinate system is a negative

coordinate.

[0377] The plane offset value ("plane_offset_value") indicates the amount of deviation in the horizontal direction, of the pixels constituting the graphics, and indicates the offset value of the plane in units of pixels.

[0378] When the playback type of PG is set as "1 plane + offsetPG", an offset sequence is extracted from the video decoder and the extracted offset sequence is supplied to the shift unit, wherein the offset sequence to be extracted is indicated by the PG_text subtitle stream offset sequence ID reference information in a piece of stream registration information corresponding to the current stream number, among a plurality of pieces of stream registration information stored in the SEI message of the dependent-view video stream.

[0379] This completes the explanation of the "procedure when playback condition is changed" for the PG_text subtitle stream.

[0380] FIG. 28 is a flow chart showing a procedure of determination processing of upper or lower end playback type.

[0381] In the determination processing of upper and lower edges playback type, video shift mode of a PG stream identified by a PG stream number in the PSR2 is acquired from stream registration information included in the extension stream selection table (step S101). It is judged on whether the acquired video_shift_mode indicates "Down" and the video shift mode of the playback device in the PSR 32 indicates "Down" (step S102).

[0382] When a result of judgment in step S102 is YES, the playback type is set as the upper end 2D subtitle playback type (step S103). In this case, PG is played back in the upper end 2D subtitle playback mode. Specifically, the demultiplexing unit performs demultiplexing on a TS packet having a packet identifier indicated by a packet identifier reference included in a stream entry corresponding to a stream number of a current stream stored in the PSR2. Also, the shift unit shifts picture data, which is stored in each of the right-eye video plane and the left-eye video plane, downward by 131 pixels.

[0383] When a result of the judgment in step S102 is No, it is judged on whether the video_shift_mode acquired in step S101 indicates "Up" and the video shift mode of the playback device in the PSR32 indicates "Up" (step S104). When a result of judgment in step S104 is YES, the playback type is set as the lower end 2D subtitle playback type (step S105). In this case, PG is played back in the lower end 2D subtitle playback mode. In the lower end 2D subtitle playback mode, the shift unit shifts picture data, which is stored in each of the right-eye video plane and the left-eye video plane, upward by 131 pixels.

[0384] This completes the description of the determination processing of upper or lower end playback type.

[0385] FIG. 29 is a flow chart showing the procedure for selecting a PG_text subtitle stream that is optimum for the current playitem.

[0386] In step S30, it is checked for all PG_text subtitle streams whether the following conditions (a), (b), and (c) are satisfied.

[0387] The conditions (a), (b), and (c) are defined as follows when the check-target PG stream is a PG stream i.

[0388] Condition (a): the playback device has the capability to decode the PG stream i.

[0389] Condition (b): PG_language_code of the PG stream i matches the language setting in the playback device. Here, the language setting in the playback device is indicated by PSR17 in the register set.

[0390] The conditions (a), (b), and (c) are defined as follows when the check-target text subtitle stream is a text subtitle stream i.

[0391] Condition (a): the playback device has the capability to extend the character code of the text subtitle stream i into a bit map.

[0392] Condition (b): the playback device has the capability to support the language attribute of the text subtitle stream i.

[0393] Condition (c): the "textST_language_code" of the text subtitle stream i matches the language setting in the playback device.

[0394] After the checking, it is judged in step S31 whether or not the playback device satisfies the condition (Z) described in the previous flow chart (playback of unsupported language). When the playback device does not satisfy the condition (Z), the control goes to step S32 in which it is judged whether or not there is a PG_text subtitle stream that satisfies the conditions (a), (b), and (c). When there are PG_text subtitle streams that satisfy the conditions (a), (b), and (c), a PG_text subtitle stream whose corresponding stream entry is placed first in the stream selection table is selected from among the PG_text subtitle streams that satisfy the conditions (a) through (c), and the PG_text subtitle stream number of the selected PG_text subtitle stream is set in PSR2 (step S33).

[0395] When there is no PG_text subtitle stream that satisfies the conditions (a), (b), and (c), the control goes to step S34 in which it is judged whether or not there is a PG_text subtitle stream that satisfies less conditions. Here the less conditions in this context mean the conditions (a) and (b). Namely, in step S34, it is judged whether or not there is a PG_text subtitle stream that satisfies the conditions (a) and (b). When there are PG_text subtitle streams that satisfy the conditions (a) and (b), a PG_text subtitle stream whose corresponding stream entry is placed first in the Stream selection table is selected among the PG_text subtitle

streams that satisfy conditions (a) and (b), and the PG_text subtitle stream number of the selected PG_text subtitle stream is set in PSR2 (step S36).

[0396] When there is no PG_text subtitle stream that satisfies the conditions (a) and (b), a value 0xFFFF as a PG_text subtitle stream number is set in PSR2 (step S35). When it is judged in step S31 that the playback device satisfies the condition (Z), the control goes to step S37 in which it is judged whether or not there is a PG_text subtitle stream that satisfies another less conditions. Here the "another less conditions" in this context mean the conditions (a) and (c). Namely, in step S37, it is judged whether or not there is a PG_text subtitle stream that satisfies the conditions (a) and (c).

[0397] When there are PG_text subtitle streams that satisfy the conditions (a) and (c), a PG_text subtitle stream whose corresponding stream entry is placed first in the stream selection table is selected among the PG_text subtitle streams that satisfy conditions (a) and (c), and the PG_text subtitle stream number of the selected PG_text subtitle stream is set in PSR2 (step S38).

[0398] When there is no PG_text subtitle stream that satisfies the conditions (a) and (c), the control goes to step S39 in which it is judged whether or not there is a PG_text subtitle stream that satisfies the condition (a). When there are PG_text subtitle streams that satisfy the condition (a), a PG_text subtitle stream whose corresponding stream entry is placed first in the stream selection table is selected among the PG_text subtitle streams that satisfy the condition (a), and the PG_text subtitle stream number of the selected PG_text subtitle stream is set in PSR2 (step S40). When there is no PG_text subtitle stream that satisfies the condition (a), a value 0xFFFF is set in PSR2 (step S35).

[0399] This completes the explanation of the procedure for selecting an optimum PG_text subtitle stream.

[0400] FIG 30 is a flow chart showing the procedure which is to be executed when a stream change is requested by the set stream stereoscopic command (set stream SS command).

[0401] In step S41, it is judged whether the number "x" specified by an operand of the set stream stereoscopic command indicates a stream number of the PG stream (YES) or the text subtitle stream (NO). In step S42, it is checked whether or not the PG stream corresponding to the number "x" (PGx) satisfies the following conditions (A) and (B).

Condition (A): The playback device has a capability to decode a PG stream that is identified by the number x.

Condition (B): The language attribute of the identified PG stream matches the language attribute of the playback device.

[0402] In step S43, it is checked whether or not the text subtitle stream corresponding to the number "x" (textSTx) satisfies the following conditions (A) and (B).

Condition (A): The playback device has a capability to extend the character code of the text subtitle stream x into a bit map.

Condition (B): the playback device has the capability to support the language attribute of the text subtitle stream x.

[0403] In step S44, it is checked whether or not the playback device satisfies the condition (Z), and then in step S45, it is judged whether or not the number is equal to or lower than the total number of stream entries in the stream selection table and conditions (A) and (B) are satisfied. When the result of judgment in step S45 is affirmative, a PG_text subtitle stream with a PG_text subtitle stream number corresponding to the number x is selected, and the number x is set in PSR2 (step S46).

[0404] When the result of judgment in step S45 is negative, the control proceeds to step S47 in which it is judged whether or not the number is equal to or lower than the total number of stream entries in the stream selection table and conditions (A) and (Z) are satisfied. When the result of judgment in step S47 is affirmative, a PG_text subtitle stream with a PG_text subtitle stream number corresponding to the number x is selected, and the number x is set in PSR2 (step S48).

[0405] When the result of judgment in step S47 is negative, the control proceeds to step S49 in which it is judged whether or not the number x is 0xFFFF. When it is judged that the number x is not 0xFFFF, the value in PSR2 is maintained since it is determined that the stream selection table indicates that no PG_text subtitle stream has been permitted to be played back (step S50).

[0406] When it is judged that the number x is 0xFFFF, a PG_text subtitle stream that is optimum for the current playitem is selected (step S51). This selection of an optimum PG_text subtitle stream is performed in a similar manner to the procedure shown in FIG. 29.

[0407] The process of subsequent steps S52 to S57 is unique to the 3D output mode. Specifically, determination processing of upper or lower end playback type is performed (step S52). In the determination processing of upper or lower end playback type, if the playback type is set as neither the upper end 2D subtitle playback type nor the lower end 2D subtitle playback type (step S53: No), is SS PG of a PG stream X identified by a PG stream number X is acquired from stream registration information included in the extension stream selection table (step S54). Then, it is judged on whether a flag of the acquired is_SS_PG indicates "1" and whether stereoscopic PG capability of PSR24 indicates "1" (step 55). If a result of the judgment in step 55 is Yes, the playback type is determined as a stereoscopic PG playback type (step 56). When the result of the judgment in step S55 is NO, the playback type is set as "1 plane +

offset" (step S57).

[0408] FIG 31 is a flow chart showing the procedure which is to be executed when a stream change is requested by the set stream command or by a user operation requesting a stream number change. In this flow chart, in step S58, the stream number specified by an operand of the set stream command, or the stream number specified by a user operation requesting a stream number change, is set as the number x, and then the process of steps S41 through S57 is executed. The contents of steps S41 through S57 are the same as those shown in FIG. 30, and thus the same reference numbers are assigned thereto, and description thereof is omitted here.

<Determination of current IG stream and playback type thereof>

[0409] A current IG stream whose stream number is to be stored in PSR0 is selected based on the output mode in PSR22, stereoscopic PG capability in PSR24, and "is_SS_IG".

[0410] FIGs. 32A and 32B are flow charts showing the procedures for determining the current IG stream and the playback type thereof.

[0411] FIG. 32A is a flow chart showing the procedure for determining the current IG stream when the playitem is changed and the playback condition of the playback device is changed. Among the steps shown in this flow chart, the process of steps S61 to S65 is common to the 3D output mode and the 2D output mode, and the process of steps S64 to S67 is unique to the 3D output mode.

[0412] In step S61, it is judged whether or not the number of entries in the stream selection table is "0". When the number is "0", the value in PSR0 is maintained (step S64).

[0413] When it is judged in step S61 that the number of entries in the stream selection table is not "0", the control proceeds to step S62 in which it is judged whether or not the number of entries in the stream selection table is equal to or greater than the value in PSR0. When the result of judgment in step S62 is affirmative, the value in PSR0 is maintained (step S65). When it is judged that the value in PSR1 is greater than the number of entries in the stream selection table, value "1" is set in PSR0 (step S63). Steps S64 through S67 that follow step S63 are unique to the 3D output mode. More specifically, steps S64 through S67 in the 3D output mode are performed as follows. An "is_SS_IG" of an IG stream identified by the IG stream number stored in PSR0 is obtained from the stream registration information in the extension stream selection table (step S64). It is judged whether or not the obtained "is_SS_IG" flag is "1" and the stereoscopic IG capability indicated by "b3" in PSR24 is "1" (step S65). When the result of judgment in step S65 is YES, the playback type is set as the stereoscopic IG (step S66). When the playback type is set as the stereoscopic IG, the stereoscopic playback is performed by using packet identifier references that are included in the left-eye and right-eye stream entries of a piece of stream registration information corresponding to the current stream number

stored in PSR0, among a plurality of pieces of stream registration information in the extension stream selection table. More specifically, the demultiplexing unit is caused to demultiplex TS packets whose packet identifiers are indicated by the packet identifier references that are included in the left-eye and right-eye stream entries of a piece of stream registration information corresponding to the current stream number stored in PSR0.

[0414] When the judgment result in step S65 is NO, the playback type is set as "1 plane + offset" (step S67).

[0415] When the playback type is set as "1 plane + offsetIG", the IG playback in the 1 plane + offset mode is executed by using an offset sequence indicated by the stereoscopic IG offset sequence ID reference information in a piece of stream registration information corresponding to the current stream number stored in PSR0, among a plurality of pieces of stream registration information in the extension stream selection table. More specifically, an offset sequence is extracted from the video decoder and the extracted offset sequence is supplied to the shift unit, wherein the offset sequence to be extracted is indicated by the stereoscopic IG offset sequence ID reference information in a piece of stream registration information corresponding to the current stream number, among a plurality of pieces of stream registration information stored in the SEI message of the dependent-view video stream.

[0416] FIG. 32B is a flow chart showing the procedure for setting PSR0 which is to be executed when a stream change is requested by the set stream stereoscopic command (set stream SS command), by the set stream command, or by a user operation requesting a stream number change.

[0417] When a stream change is requested by the set stream stereoscopic command (set stream SS command), by the set stream command, or by a user operation requesting a stream number change, the stream number specified by an operand of the command, or the stream number specified by a user operation, is set as the number x and the procedure is executed as follows.

[0418] In step S71, it is judged whether or not the number of entries in the stream selection table is equal to or greater than the number x. When the result of judgment in step S71 is affirmative, the value is set in PSR0 (step S74). When it is judged that the value x is greater than the number of entries in the stream selection table, value "1" is set in PSR0 (step S72). In the 3D output mode, the procedure is executed as follows. An "is_SS_IG" of an IG stream identified by the IG stream number stored in PSR0 is obtained from the stream registration information in the extension stream selection table (step S73). It is judged whether or not the obtained "is_SS_IG" flag is "1" and the stereoscopic IG capability indicated by PSR24 is "1" (step S74). When the result of judgment in step S74 is YES, the playback type is set as the stereoscopic IG (step S75). When the judgment result in step S74 is NO, the playback type is set as "1 plane + offset" (step S76).

[0419] FIGs. 33A through 33C show what packet identifiers are output to the demultiplexing

unit by the combined stream registration sequence.

[0420] FIG. 33A shows the combined stream registration sequence used in the operation as an example. The combined stream registration sequence is composed of three pieces of stream registration information provided in the basic stream selection table and three pieces of stream registration information provided in the extension stream selection table.

[0421] The three pieces of stream registration information provided in the extension stream selection table have stream numbers "1", "2", and "3", respectively, and the stream attributes in the three pieces of stream registration information have "English", "Japanese", and "Chinese" as the language attributes, respectively. The stream registration information provided in the basic stream selection table differs in the packet identifier stored in the stream entry, from the stream registration information provided in the extension stream selection table. Also, the stream registration information provided in the extension stream selection table contains (i) a packet identifier for a base-view PG stream for the B-D presentation mode, and (ii) a packet identifier for a dependent-view PG stream.

[0422] FIG. 33B shows the setting of a stream number and the outputting of a packet identifier when such a combined stream registration sequence is supplied to playback device in which the language has been set to "Chinese" and the output mode has been set to the 2D output mode.

[0423] The arrows identified by signs "a1", "a2", and "a3" schematically indicate (i) the judgment on whether language settings match each other, (ii) the setting of a stream number in the stream number register, and (iii) the output of a packet identifier to the demultiplexing unit, respectively.

[0424] In the operation procedure of this example, it is judged whether the language setting of the playback device matches the stream attribute contained in the stream registration information whose stream number is "3", and it is judged that they match. As a result of this, the stream number "3" of this stream registration information is written into the stream number register. Also, the packet identifier written in the stream entry of the basic stream selection table is output to the demultiplexing unit. Following this, a TS packet identified by the packet identifier written in the stream entry of the stream registration information whose stream number is "3" in the basic stream selection table is output to the decoder.

[0425] FIG. 33C shows the setting of a stream number and the outputting of a packet identifier when such a combined stream registration sequence is supplied to playback device in which the language has been set to "Chinese" and the output mode has been set to the B-D presentation mode.

[0426] The arrows identified by signs "a4", "a5", and "a6" schematically indicate (i) the judgment on whether language settings match each other, (ii) the setting of a stream number in the stream number register, and (iii) the output of a packet identifier to the demultiplexing

unit, respectively.

[0427] In the operation procedure of this example, it is judged whether the language setting of the playback device matches the stream attribute contained in the stream registration information whose stream number is "3", and it is judged that they match. As a result of this, the stream number "3" of this stream registration information is written into the stream number register. Also, the packet identifier written in the stream entry of the basic stream selection table is output to the demultiplexing unit. Following this, a pair of TS packets identified by a pair of packet identifiers written in the stream entry of the stream registration information whose stream number is "1" in the extension stream selection table are output to the decoder.

[0428] FIGs. 34A through 34C show what packet identifiers are output to the demultiplexing unit by the combined stream registration sequence.

[0429] FIG 34A shows the combined stream registration sequence used in the operation as an example. The combined stream registration sequence is composed of three pieces of stream registration information provided in the basic stream selection table and three pieces of stream registration information provided in the extension stream selection table. The three pieces of stream registration information provided in the basic stream selection table have stream numbers "1", "2", and "3", respectively, and all of the stream attributes in the three pieces of stream registration information have "Chinese" as the language attributes.

[0430] The three pieces of stream registration information provided in the extension stream selection table have stream numbers "1", "2", and "3", respectively, and all of the stream attributes in the three pieces of stream registration information have "Chinese" as the language attributes. The stream registration information provided in the basic stream selection table differs in the packet identifier stored in the stream entry, from the stream registration information provided in the extension stream selection table. Also, the stream registration information provided in the extension stream selection table contains (i) a packet identifier for a left-eye PG stream for the B-D presentation mode, and (ii) a packet identifier for a right-eye PG stream for the B-D presentation mode.

[0431] FIG. 34B shows the setting of a stream number and the outputting of a packet identifier when such a combined stream registration sequence is supplied to playback device in which the language has been set to "Chinese" and the output mode has been set to the 2D output mode.

[0432] The arrows identified by signs "a1", "a2", and "a3" schematically indicate (i) the judgment on whether language settings match each other, (ii) the setting of a stream number, and (iii) the output of a packet identifier to the demultiplexing unit, respectively.

[0433] In the stream selection procedure of this example, it is judged whether the language setting of the playback device matches the stream attribute contained in the stream registration information whose stream number is "1", and it is judged that they match. As a result of this,

the stream number "1" of this stream registration information is written into the stream number register. Also, the packet identifier written in the stream entry of the basic stream selection table is output to the demultiplexing unit. Following this, a TS packet identified by the packet identifier written in the stream entry of the stream registration information whose stream number is "1" in the basic stream selection table is output to the decoder.

[0434] FIG. 34C shows the setting of a stream number and the outputting of a packet identifier when such a combined stream registration sequence is supplied to playback device in which the language has been set to "Chinese" and the playback type has been set to the 1 plane + Offset type.

[0435] The arrows identified by signs "a4", "a5", and "a6" schematically indicate (i) the judgment on whether language settings match each other, (ii) the setting of a stream number in the stream number register, and (iii) the output of a packet identifier to the demultiplexing unit, respectively.

[0436] In the operation procedure of this example, it is judged whether the language setting of the playback device matches the stream attribute contained in the stream registration information whose stream number is "1", and it is judged that they match. As a result of this, the stream number "1" of this stream registration information is written into the stream number register. Also, the packet identifier written in the stream entry of the basic stream selection table is output to the demultiplexing unit. Following this, a pair of TS packets identified by a pair of packet identifiers written in the stream entry of the stream registration information whose stream number is "1" in the extension stream selection table are output to the decoder.

[0437] According to the present embodiment as described above, the extension stream selection table includes a video shift mode that defines saving of subtitle display region in correspondence with a stream number. Accordingly, when the playback section changes, or when a request for changing the stream is received, a stream selection procedure is executed. When a new stream number is set in a stream number register, a video shift mode corresponding to the set new stream number is provided with the playback device. With this structure, it is possible to realize control in which a display region of a subtitle is saved in the upper end of the screen in a playback section and a display region of a subtitle is saved in the lower end of the screen in another playback section.

[0438] The cinema scope size (1:2.35) is generally used for the aspect ratio of video of movies. In the case where a video is stored in an optical disc such as a BD-ROM, a main feature video is disposed in the center of an HD video having the aspect ratio of 16:9 without changing the aspect ratio, and a black frame is inserted into each of the upper side and the lower side of the HD video. Accordingly, with the above structure, it is possible to display subtitles in a large subtitle display region generated by collecting black frames located above and below the main feature video to one of the upper end and the lower end of the video plane. This can improve the use efficiency of the screen, thereby improving the stereoscopic effect.

(Modification Example)

[0439] As a modification example of the present embodiment, the following describes a method of shifting upward or downward not only picture data stored in the video plane memory but also subtitles stored in the PG plane memory so as to overlay the picture data with the subtitles.

[0440] FIGs. 35A through 35C show the stream registration sequences in the extension stream selection table according to the present modification example. FIG 35B shows the internal structure of the PG stream registration sequence.

[0441] In the present modification example, the stream registration information of the PG stream additionally includes a "PG shift value video shift upward (PG_v_shift_value_for_Up)" and a "PG shift value video shift downward (PG_v_shift_value_for_Down)".

[0442] The "PG shift value in video shift upward (PG_v_shift_value_for_Up)" represents an amount of downward shift of subtitle data stored in the PG plane memory in the case where the video shift mode is set as "Up" and a display region of subtitles of a PG_text subtitle stream is saved in the lower ends of the video plane.

[0443] The "PG shift value in video shift downward (PG-v-shift-value-for-Down)" represents an amount of upward shift of subtitle data stored in the PG plane memory in the case where the video shift mode is set as "Down" and a display region of subtitles of a PG_text subtitle stream is saved in the upper ends of the video plane.

[0444] These values are set in PSR33 shown in FIG 37. The shift amount shown by the PSR33 includes a plane shift amount in video shift upward and a plane shift amount in video shift downward for each plane. For example, the PSR33 includes "PG_shift_value_for_UP" and "PG_shift_value_for_Down" for a PG plane. These values are set by acquiring PG_v_shift_value_for_Up and PG_v_shift_value_for_Down of a current PG stream selected by stream switching from stream registration information in the extension stream selection table.

[0445] FIG. 36 shows the circuit structure for overlaying data output from the decoder model and outputting the overlaid data in the upper end 2D subtitle playback mode and the lower end 2D subtitle playback mode. In the present modification example, in the upper end 2D subtitle playback mode and the lower end 2D subtitle playback mode, in accordance with the setting of the PSR32, pixel shift of the video plane is performed upward or downward by 131 pixels for each of the left-eye view and the right-eye view. Also, in accordance with the value of PG_shift_value_for_Up or the value of PG_shift_value_for_Down set in the PSR33, pixel shift of the PG plane is performed upward or downward for each of the left-eye view and the right-eye view. Then, layer overlaying is performed on these pixels.

[0446] Specifically, when the video_shift_mode in the PSR32 is set as "Up", a picture output from the video plane memory is shifted upward by 131 pixels, and a subtitle output from the PG plane is shifted downward by the number of pixels set in the PG_shift_value_for_Up in the PSR33, and layer overlaying is performed on the picture and subtitle, as shown in FIG. 38A. On the other hand, when the video_shift_mode in the PSR32 is set as "Down", a picture output from the video plane memory is shifted downward by 131 pixels, and a subtitle output from the PG plane is shifted upward by the number of pixels set in the PG_shift_value_for_Down in the PSR33, and layer overlaying is performed on the picture and subtitle, as shown in FIG. 38B.

[0447] Here, in the present modification example, as shown in FIG. 39, in the case where the video_shift_mode indicates "Up" or "Down", plane shift results in a cropped region. Accordingly, there only needs to make a restriction such that no subtitle data is in the cropped region. In other words, as shown in the left side of FIG. 39, since a region other than a region surrounded by a dashed line has a possibility to be cropped, a display position of the PG is restricted such that no subtitle data is displayed on the region other than the region surrounded by the dashed line. The coordinate of the region is represented by $(0, PG_v_shfit_value_for_Down)$, $(0, height + PG_v_sfhit_value_for_Up)$, $(width, PG_v_shfit_value_for_Down)$, and $(width, height + PG_v_sfhit_value_for_Up)$. For example, if PG_v_ssthit_value_for_Up indicates -a and PG_v_sfhit_value_for_Down indicates +b, the region is represented by $(0, b)$, $(0, height - a)$, $(width, b)$, and $(width, height - a)$. As the constraint conditions for PG, the display position is restricted so as not to go beyond the above region, the display position to which the size of an object to be displayed is added is restricted so as not to go beyond the above region, the display position of the window is restricted so as not to go beyond the above region, and the display position of the window to which the window size is added is restricted so as not to go beyond the above region, for example. Such constraint conditions can prevent display of a partially lacking.

(Embodiment 2)

[0448] The following describes Embodiment 2 of the present invention.

[0449] In the present embodiment, a method is described for realizing 3D video having an appropriate depth depending on the screen size of a TV connected to the 2D/3D playback device.

[0450] In the case of 3D video with use of the parallax images, the screen size affects the sense of depth of 3D video, as shown in the left side of FIG. 40. This is due to the difference value between the left-eye video and the right eye video that varies depending on the screen size of the TV. Suppose that, for example, in the case where a left video and a right video are created so as to realize the most appropriate width for a 50-inch TV as shown in the left side of FIG 40. In such a case, it is possible to realize the most appropriate viewing for a 50-inch TV. However, the difference value between the left-eye video and the right eye video is small for a TV smaller than the 50-inch TV, and as a result a video that is not powerful and does not have

much width is displayed in such a TV. On the other hand, the difference is too large for a TV larger than the 50-inch TV, and this causes the user to have an eyestrain. In view of this, it is preferable to apply an offset value for correcting the screen size to each of the left-eye plane and the right eye plane for output to the TV, as shown in the right side of FIG. 40. For example, in the case where the left-eye video and the right eye video are optimized for the 50-inch TV as shown in FIG. 40, an offset value is set for a 32-inch TV so as to increase the sense of depth for output to the TV. An offset value is set for a 100-inch TV so as to decrease the sense of depth for output to the TV. Setting of an offset value indicates, like the 1 plane + offset method, the last plane output from the player is shifted based on an offset value, and is cropped. An offset value to be applied to this last plane of the player is referred to as an "output offset correction value". The following describes a specific method.

[0451] Firstly, the data structure is described. The basic parts of the data structure are the same as those for storing 3D videos described in the above embodiments, and accordingly additional parts or different parts from the above embodiments are mainly described here.

[0452] In a file such as an index file, a playlist file, and an AV stream information file, a table as shown in FIG. 41A is stored. In this table, a plurality of pieces of screen size information are registered, each of which includes a TV inch and an output offset correction value that are grouped into pairs. In FIG. 41A, the inch number is defined for each 10 inches. Alternatively, the inch number may be defined for each arbitrary inch in accordance with a predetermined standard. Further alternatively, the user may define the inch number. It may be employed to prepare some tables as shown in FIG. 41A, and only reference IDs of these tables are registered in the file such as the index file, the playlist file, and the AV stream information file. It may be employed to prepare a function for determining an offset value depending on the inch number as shown in the right side of FIG. 41B.

[0453] The table may include, in addition to the pairs of TV inch and output offset correction values, a value of the optimal TV size (assumed_TV_size_when authoring) indicating an inch a created content targets. The use of the values of the optimal TV size makes it possible to perform various types of correction processing. For example, in the case where display is performed by a TV having an optimal inch size or greater, it is possible to perform processing such as displaying images having the optimal inch size on the center of the screen of the TV and displaying a black frame around a video, as shown in FIG. 42.

[0454] Next, the playback device relating to the preset embodiment is described. The playback device includes, as shown in FIG 43, a PSR35 that is a system parameter for storing an output offset correction value and an output offset correction value application unit. The playback control unit acquires the screen size (inch number) of a TV to be connected with the playback device via an HDMI cable or the like, identifies an output offset correction value corresponding to the screen size based on the table shown in FIG. 41, and stores the identified output offset correction value in the PSR35. The output offset correction value application unit refers to the value stored in the PSR35, and set an offset value for a plane the left-eye video and the right-eye video that are overlaid by the plane addition unit, using a value of the PSR35.

[0455] Instead of storing an output offset correction value in the PSR35, the structure may be employed in which the screen size is stored in the PSR35 and the output offset correction value application unit identifies an output offset correction value with reference to the table shown in FIG. 41.

[0456] Note that the output offset correction value may be adjusted depending on a user who is watching the video. For example, since an infant has a narrow distance between left and right eyes, the smaller difference between a left-eye video and a right eye video is preferable. In view of this, an "output offset correction value α " for correcting the output offset correction value is prepared. The output offset correction value application unit performs offset correction processing with use of a value resulting from multiplying the offset correction value by the "output offset correction value α ". Specifically, the processing is realized with use of the structure shown in FIG. 44. FIG 44 shows a PSR36 in which the output offset correction value α is stored. The playback control unit or the program execution unit sets the value on the PSR36 via the menu screen, the OSD screen of the player, or the like. For example, in order to decrease the depth for a child user, it is possible to decrease the depth by setting a value greater than 1. The output offset correction value application unit refers to the PSR35 and the PSR36, and applies offset on the plane with use of a value resulting from multiplying the output offset correction value by the output offset correction value α . As a result, it is possible to adjust the sense of depth in accordance with a user's preference.

[0457] It may be possible to employ the structure in which the "output offset correction value α " is set on the menu screen of the BD program by selecting one of three modes "weak", "normal", and "strong" for the depth of 3D.

[0458] The "output offset correction value α " may be stored for each of the SEI message of a video stream, the descriptor of the PMT packet, the playitem, or the like, and may be changed depending on the scene. With such a structure, it is possible to set an "output offset correction value α " having a greater value for a scene having a large depth, for example.

[0459] In the present embodiment, the output offset amount correction value is changed depending on the screen size of TV. Alternatively, the output offset amount correction value or the output offset correction value α may be changed depending on the distance from the TV to the user. In this case, the following structure may be employed. Glasses for 3D viewing measures the distance from the screen of the TV to the glasses, the TV acquires the distance, and then the TV notifies the playback device of the distance via an HDMI cable.

[0460] In the present embodiment, the output offset amount correction value is changed depending on the screen size of TV. Alternatively, the projector may measure the screen size in the following methods because the projector cannot recognize the size of the screen. According to one of the methods, the projector outputs a laser to the screen such as infrared light, the distance is measured with use of the flashback of the infrared light from the screen, and the screen size is calculated with use of the optical parameter of the lens. According to

another one of the methods, the length of a "line segment" is displayed on the projector, and the user measures the length of the line segment on the screen inputs the length via the OSD of the projector. The projector can calculate the length of the screen depending on the length of the line segment on the screen.

[0461] According to the present embodiment as described above, it is possible to achieve an optimal stereoscopic effect suitable for each screen size by performing offset processing for changing the difference value between the left-eye video and the right-eye video depending on the screen size for video display.

(Embodiment 3)

[0462] Embodiment 3 relates to an improvement of the internal structure of the stereoscopic interleaved stream file.

[0463] Here, as a premise of the present embodiment, files in the UDF file system will be explained briefly. The UDF file is composed of a plurality of Extents managed by the file entry. The "file entry" includes a "descriptor tag", an "ICB tag", and an "allocation descriptor".

[0464] The "descriptor tag" is a tag identifying, as a "file entry", the file entry which includes the descriptor tag itself. The descriptor tag is classified into a file entry descriptor tag, a space bit map descriptor tag, and so on. In the case of a file entry descriptor tag, "261", which indicates "file entry" is written therein.

[0465] The "ICB tag" indicates attribute information concerning the file entry itself.

[0466] The "allocation descriptor" includes a Logical Block Number (LBN) indicating a recording position of an Extent constituting a low-order file under a directory. The allocation descriptor also includes data that indicates the length of the Extent. The high-order two bits of the data that indicates the length of the Extent are set as follows: "00" to indicate an allocated and recorded Extent; "01" to indicate an allocated and not-recorded Extent; and: "11" to indicate an Extent that follows the allocation descriptor. When a low-order file under a directory is divided into a plurality of Extents, the file entry should include a plurality of allocation descriptors in correspondence with the Extents.

[0467] It is possible to detect an address of an Extent constituting a stream file by referring to the allocation descriptor in the file entry described above.

[0468] The following describes the files in various types that are used in the present embodiment.

<Stereoscopic interleaved stream file (FileSS)>

[0469] The stereoscopic interleaved stream file (FileSS) is a stream file (2TS-interleaved file) in which two TSs are interleaved, and is identified by a five-digit integer value and an extension (ssif) indicating an interleave-format file for stereoscopic playback. The stereoscopic interleaved stream file is composed of Extent SS[n]. The Extent SS[n] (also referred to as "EXTSS[n]") is identified by the index number "n". The index number "n" increments in order starting from the top of the stereoscopic interleaved stream file.

[0470] Each Extent SS[n] is structured as a pair of a dependent-view data block and a base-view data block.

[0471] The dependent-view data block and base-view data block constituting the Extent SS[n] are a target of cross reference by the file 2D, file base, and file dependent. Note that the cross reference means that a piece of data recorded on a recording medium is registered as an Extent of a plurality of files in the file entries thereof. In the present embodiment, the starting addresses and continuation lengths of the dependent-view data block and base-view data block are registered in the file entries of the file 2D, file base, and file dependent.

<File base (FileBase)>

[0472] The file base (FileBase) is a virtual stream file that is presumed to "store" a main TS specified by the extent start point information in the clip information corresponding to the file 2D. The file base (FileBase) is composed of at least one Extent 1[i] (also referred to as "EXT1[i]"). The Extent 1[i] is the i^{th} Extent in the file base, where "i" is an index number of the Extent and is incremented starting from "0" at the top of the file base. The file base is a virtual stream file used to treat the stereoscopic interleaved stream file, which is a 2TS-file, as a ITS-file. The file base is generated in a virtual manner by building its file entry in the memory of the playback device.

[0473] In the actual reading, the file base is identified by performing a file open using a file name of the stereoscopic interleaved stream file. More specifically, when the file open using a file name of the stereoscopic interleaved stream file is called, the middleware of the playback device generates, in the memory, a file entry identifying an Extent in the file base, and opens the file base in a virtual manner. The stereoscopic interleaved stream file can be interpreted as "including only one TS", and thus it is possible to read a 2TS stereoscopic interleaved stream file from the recording medium as a 1TS file base.

[0474] When only a base-view data block is to be read in the B-B presentation mode, only the Extents constituting the file base become the target of the reading. Even if the mode is switched from the B-B presentation mode to the B-D presentation mode, both the dependent-view data block and the base-view data block can be read by extending the reading range from the Extents constituting the file base to the Extents constituting the stereoscopic interleaved

stream file. Thus, with this arrangement, the efficiency of the file reading is not decreased.

<File dependent (FileDependent)>

[0475] The file dependent (FileDependent) is a stream file that is presumed to "store" a sub-TS, and is composed of Extent 2[i] (also referred to as "EXT2[i]"). The Extent 2[i] is the i^{th} Extent in the file dependent, where "i" is an index number of the Extent and is incremented starting from "0" at the top of the file dependent. The file dependent is a virtual stream file used to treat the stereoscopic interleaved stream file, which is a 2TS-file, as a 1TS-file storing the sub-TS. The file dependent is generated in a virtual manner by building its file entry in the memory of the playback device.

[0476] The dependent-view video stream is attached with and accessed with use of a file name that is represented by a number generated by adding "1" to the five-digit integer representing the file name of the stereoscopic interleaved stream file. The recording medium stores a dummy file, and the "number generated by adding 1", namely, the identification number of the dependent-view video stream, is attached to the dummy file. Note that the dummy file is a file that stores no Extent, namely, substantial information, but is attached with only a file name. The dependent-view video stream is treated as being stored in the dummy file.

<File 2D (File2D)>

[0477] The file 2D (File2D) is a 1TS stream file storing a main TS that is played back in the 2D output mode, and is composed of the Extent 2D. The file 2D is identified by a five-digit integer value and an extension (ssif) indicating an interleave-format file for stereoscopic playback.

[0478] FIG. 45 shows the correspondence between the file 2D/file base and the file dependent.

[0479] In FIG. 45, the first row shows a file 2D/file base 00001.m2ts and a file dependent 00002.m2ts. The second row shows Extents that store dependent-view data blocks and base-view data blocks. The third row shows a stereoscopic interleaved stream file 00001.ssif.

[0480] The dotted arrows h1, h2, h3, and h4 show the files to which Extents EXT1[i] and EXT2[i] belong, the belongingness being indicated by the allocation identifiers. According to the belongingness guided by the arrows h1 and h2, Extents EXT1[i] and EXT1[i+1] are registered as Extents of the file base 00001.m2ts.

[0481] According to the belongingness guided by the arrows h3 and h4, Extents EXT2[i] and EXT2[i+1] are registered as Extents of the file dependent 00002.m2ts.

[0482] According to the belongingness guided by the arrows h5, h6, h7, and h8, Extents EXT1[i], EXT2[i], EXT1[i+1], and EXT2[i+1] are registered as Extents of 00001.ssif. As understood from this, Extents EXT1[i] and EXT1[i+1] have the duality of belonging to 00001.ssif and 00001.m2ts. The extension "ssif" is made of capital letters of StereoScopic Interleave File, indicating that the file is in the interleave format for stereoscopic playback.

[0483] Here, a pair of an Extent constituting the file base and an Extent constituting the file dependent that are both identified by the same Extent identifier is called an "interleave Extent unit". In the example shown in FIG. 45, a pair of EXT1[i] and EXT2[i] that are both identified by an Extent identifier "i" is an interleave Extent unit [i]. Also, a pair of EXT1[i+1] and EXT2[i+1] that are both identified by an Extent identifier "i+1" is an interleave Extent unit [i+1]. In a random access to a stereoscopic interleaved stream file, it is necessary to ensure that an interleave Extent unit identified by the Extent identifier is read from the recording medium completely at once.

[0484] FIGs. 46A through 46C show the correspondence between the interleaved stream file and file 2D/file base.

[0485] The third row in FIG 46A shows the internal structure of the interleaved stream file. The stereoscopic interleaved stream file is composed of Extents EXT1[1] and EXT1[2] storing base-view data blocks and EXT2[1] and EXT2[2] storing dependent-view data blocks, wherein they are arranged alternately in the interleave format.

[0486] The first row in FIG. 46A shows the internal structure of the file 2D/file base. The file 2D/file base is composed of only Extents EXT1[1] and EXT1[2] storing base-view data blocks, among the Extents constituting the interleaved stream file shown in the third row. The file 2D/file base and the interleaved stream file have the same name, but different extensions.

[0487] The second row in FIG 46A shows the internal structure of the file dependent. The file dependent is composed of only Extents EXT2[1], EXT2[2], and EXT2[3] storing dependent-view data blocks, among the Extents constituting the interleaved stream file shown in the third row. The file name of the file dependent is a value higher by "1" than the file name of the interleaved stream file, and they have different extensions.

[0488] Not all playback devices necessarily support the 3D playback system. Therefore, it is preferable that even an optical disc including a 3D image supports a 2D playback. It should be noted here that the playback devices supporting only the 2D playback do not identify the data structure extended for the 3D. The 2D playback devices need to access only the 2D playlists and 2D streams by using a conventional identification method provided to the 2D playback devices. In view of this, the base-view video streams are stored in a file format that can be recognized by the 2D playback devices.

[0489] According to the first method, the main TS is assigned with the same file name as that

in the 2D playback system so that the above-described referencing of playlist information can be realized, that is to say, so that the main TS can be used in the 2D playback as well, and stream files in the interleave format have a different extension. FIG 46B shows that files "00001.m2ts" and "00001.ssif" are coupled with each other by the same file name "00001", although the former is in the 2D format and the latter is in the 3D format.

[0490] In a conventional 2D playback device, the playlist refers to only the AV clips the main TS, and therefore the 2D playback device plays back only the file 2D. On the other hand, in a 3D playback device, although the playlist refers to only the file 2D storing the main TS, when it finds a file that has the same identification number and a different extension, it judges that the file is a stream file in the interleave format for the 3D image, and outputs the main TS and sub-TS.

[0491] The second method is to use different folders. The main TSs are stored in folders with conventional folder names (for example, "STREAM"), but sub-TSs are stored in folders with folder names unique to 3D (for example, "SSIF"), with the same file name "00001". In the 2D playback device, the playlist refers to only files in the "STREAM" folder, but in the 3D playback device, the playlist refers to files having the same file name in the "STREAM" and "SSIF" folders simultaneously, making it possible to associate the main TS and the sub-TS.

[0492] The third method uses the identification numbers. That is to say, this method associates the files based on a predetermined rule regarding the identification numbers. For example, when the identification number of the file 2D/file base is "00001", the file dependent is assigned with identification number "00002" that is made by adding "1" to the identification number of the file 2D, as shown in FIG 46C. However, the file system of the recording medium treats the file dependent, which is assigned with a file name according to the rule, as a non-substantial dummy file. This is because the file dependent is, in the actuality, the stereoscopic interleaved stream file. The file names having been associated with each other in this way are written into (i) the stream registration information in the basic stream selection table and (ii) the sub-clip entry ID reference (ref_to_subclip_entry_id) in the stream registration information in the extension stream selection table. On the other hand, the playback device recognizes a file name, which is a value higher by "1" than the file name written in the sub-clip entry ID reference, as the file name of the dummy file, and performs the process of opening the file dependent in a virtual manner. This ensures that the stream selection procedure reads, from the recording medium, the file dependent that is associated with other files in the above-described manner.

[0493] The clip information files are identified by the same rule as above.

[0494] This completes the description of the file 2D, file base, and file dependent.

[0495] The following explains the data blocks in detail.

<Base-view data block>

[0496] The base-view data block (B[i]) is the i^{th} data in the main TS. Note that the main TS is a TS specified as the main element of the main path by the clip information file name information of the current playitem information. The "i" in B[i] is an index number that is incremented starting from "0" corresponding to the data block at the top of the file base.

[0497] The base-view data blocks fall into those shared by the file base and the file 2D, and those not shared by the file base and the file 2D.

[0498] The base-view data blocks shared by the file base and the file 2D and the base-view data blocks unique to the file 2D become the Extents of the file 2D, and they are set to have a length that does not cause a buffer underflow in the playback device. The starting sector address of the base-view data blocks is written in the allocation descriptor in the file entry of the file 2D.

[0499] The base-view data blocks unique to the file base, which are not shared by the file 2D, do not become the Extents of the file 2D, and thus they are not set to have a length that does not cause an underflow in a single buffer in the playback device. The base-view data blocks are set to have a smaller size, namely, a length that does not cause an underflow in a double buffer in the playback device.

[0500] The starting sector addresses of the base-view data block unique to the file base are not written in the allocation descriptor in the file entry. Instead of this, the starting source pocket in the base-view data block is pointed to by the extent start point information in the clip information of the clip information file corresponding to the main TS. Therefore, the starting sector address of a base-view data block unique to the file base needs to be obtained by using (i) the allocation descriptor in the file entry of the stereoscopic interleaved stream file and (ii) the extent start point information in the clip information.

<Dependent-view data block>

[0501] The dependent-view data block (D[i]) is the i^{th} data in the sub-TS. Note that the sub-TS is a TS specified as the main element of the sub-path by the stream entry in the stream registration sequence in the extension stream selection table corresponding to the current playitem information. The "i" in D[i] is an index number that is incremented starting from "0" corresponding to the data block at the top of the file dependent.

[0502] The dependent-view data blocks become the Extents of the file dependent, and are set to have a length that does not cause an underflow in a double buffer in the playback device.

[0503] Also, in the continuous regions in the recording medium, a dependent-view data block

is disposed before a base-view data block that is played back in the same playback time together the dependent-view data block. For this reason, when the stereoscopic interleaved stream file is read, the dependent-view data block is read before the corresponding base-view data block, without fail.

[0504] The starting sector addresses of the dependent-view data blocks are not written in the allocation descriptor in the file entry of the file 2D since the dependent-view data blocks are not shared by the file 2D. Instead of this, the starting source pocket in the dependent-view data block is pointed to by the extent start point information in the clip information. Therefore, the starting sector address of a dependent-view data block needs to be obtained by using (i) the allocation descriptor in the file entry of the file 2D and (ii) the extent start point information in the clip information.

<Classification of Extent>

[0505] As described above, the Extents of the file 2D fall into those shared by the file base, and those not shared by the file base.

[0506] Suppose here that the Extents of the file 2D are B[0], B[1], B[2], B[3]2D, and B[4]2D, and that the Extents of the file base are B[0], B[1], B[2], B[3]ss, and B[4]ss. Of these, B[0], B[1], and B[2] are base-view data blocks shared by the file base. B[3]2D and B[4]2D are base-view data blocks unique to the file 2D, not shared by the file base.

[0507] Also, B[3]ss and B[4]ss are base-view data blocks unique to the file base, not shared by the file 2D.

[0508] The data of B[3]2D is bit-for-bit same as data of B[3]ss. The data of B[4]2D is bit-for-bit same as data of B[4]ss.

[0509] The data blocks B[2], B[3]2D, and B[4]2D in the file 2D constitute Extents (big Extents) having a large continuation length immediately before a position at which a long jump is caused. In this way, big Extents can be formed immediately before a long jump in the file 2D. Accordingly, even when a stereoscopic interleaved stream file is played back in the 2D output mode, there is no need to worry an occurrence of an underflow in the read buffer.

[0510] The file 2D and the file base have sameness, although being partially different in Extents. Therefore, the file 2D and the file base are generically called "file 2D/file base".

[0511] FIG. 47 shows correspondence among the stereoscopic interleaved stream file, file 2D, file base, and file dependent. The first row in FIG. 47 shows the file 2D, the second row shows data blocks recorded on the recording medium, the third row shows the stereoscopic interleaved stream file, the fourth row shows the file base, and the shows the file dependent.

[0512] The data blocks shown in the second row are D[1], B[1], D[2], B[2], D[3], B[3]ss, D[4], B[4]ss, B[3]2D, and B[4]2D. The arrows ex1, ex2, ex3, and ex4 show the belongingness in which, among these data blocks, data blocks B[1], B[2], B[3]2D, and B[4]2D constitute the Extents of the file 2D.

[0513] The arrows ex5 and ex6 show the belongingness in which D[1], B[1], D[2], B[2], D[3], B[3]ss, D[4], and B[4]ss constitute the Extents of the stereoscopic interleaved stream file.

[0514] The fourth row shows that, among these data blocks constituting the stereoscopic interleaved stream file, B[1], B[2], B[3]ss, and B[4]ss constitute the Extents of the file base. The fifth row shows that, among the data blocks constituting the stereoscopic interleaved stream file, D[1], D[2], D[3], and D[4] constitute the Extents of the file dependent.

[0515] FIG. 48 shows the 2D playlist and 3D playlist. The first row shows the 2D playlist information. The second row shows the base-view data blocks. The third row shows the 3D playlist. The fourth row shows the dependent-view data blocks.

[0516] The arrows rf1, rf2, and rf3 show a playback path generated by combining the extension "m2ts" and a file name "00001" described in "clip_information_file_name" in the playitem information of the 2D playlist information. In this case, the playback path on the base-view side is constituted from data blocks B[1], B[2], and B[3]2D.

[0517] The arrows rf4, rf5, rf6, and rf7 show a playback path specified by the playitem information of the 3D playlist information. In this example, the playback path on the base-view side is constituted from data blocks B[1], B[2], B[3]ss, and B[4]ss.

[0518] The arrows rf8, rf9, rf10, and rf11 show a playback path specified by the sub-playitem information of the 3D playlist information. In this example, the playback path on the dependent-view side is constituted from data blocks D[1], D[2], D[3], and D[4]. These data blocks constituting the playback paths specified by the playitem information and the sub-playitem information can be read by opening files that are generated by combining the extension "ssif" and file names written in "clip_information_file_name" in the playitem information.

[0519] As shown in FIG. 48, the clip information file name information in the 3D playlist and the clip information file name information in the 2D playlist have file names in common. Accordingly, the playlist information can be written to include description that is common to the 3D playlist and the 2D playlist (see as signs df1 and df2 indicate) so as to define the 3D playlist and the 2D playlist. Accordingly, once playlist information for realizing the 3D playlist is written: the playlist information functions as the 3D playlist when the output mode of the playback device is the stereoscopic output mode; and the playlist information functions as the 2D playlist when the output mode of the playback device is the 2D output mode. The 2D playlist and the 3D playlist shown in FIG 48 have in common a piece of playlist information, which is interpreted as the 2D playlist or the 3D playlist depending on the output mode of the playback device that interprets the piece of playlist information. This reduces the amount of time and effort made by

a person in charge of authoring.

[0520] When main TSs and sub-TSs are stored in the stereoscopic interleaved stream file, a file name of the file 2D is written in "clip information file name" in the playitem information of the 2D playlist, and a file name of the file base is written in "clip information file name" in the playitem information of the 3D playlist. Since the file base is a virtual file and its file name is the same as that of the stereoscopic interleaved stream file, the file name of the stereoscopic interleaved stream file can be written in "clip_information_file_name" in the playitem information. A file name of the file dependent is written in "ref_to_subclip_entry_id" in the stream registration information in the extension stream selection table. The file name of the file dependent is created by adding "1" to the identification number of the stereoscopic interleaved stream file.

[0521] As described above, base-view and dependent-view data blocks are stored in one stereoscopic interleaved stream file, and the stereoscopic interleaved stream file can be opened as a file of any of the file 2D, file base, and file dependent. With this structure, the decoder can treat the stereoscopic interleaved stream file in the same manner as a regular stream file. Thus the storage method of the base-view and dependent-view video streams can be positively used for the storage of the stereoscopic interleaved stream file.

[0522] Next, the internal structure of the clip information file will be described in detail.

[0523] FIGs. 49A through 49D show the internal structure of the clip information file.

[0524] FIG. 49A shows the clip information file for 2D. FIG. 49B shows the clip information file for 3D. These clip information files include "clip information", "sequence information", "program information", and "characteristic point information".

[0525] The "clip information" is information indicating, for each ATC sequence, what type of AV clip each source packet sequence stored in the stream file is.

[0526] The "sequence information" indicates, for each ATC sequence, information (ATC sequence information) that indicates what type of ATC sequence one or more source packet sequences stored in the stream file are. The ATC sequence information includes: information indicating, by the source packet number, where the source packet being the start point of the ATC exists; offsets between the STC sequence identifiers and the ATC sequence identifiers; and STC sequence information corresponding to each of a plurality of STC sequences. Each piece of STC sequence information includes: a packet number of a source packet storing the PCR of the STC sequence in concern; information indicating where in the STC sequence the source packet being the start point of the STC sequence exists; and the playback start time and the playback end time of the STC sequence.

[0527] The "program information" indicates the program structures of the main TS and sub-TSs managed as "AV clips" by the clip information file. The program information indicates what

types of ESs are multiplexed in the AV clip. More specifically, the program information indicates what types of packet identifiers the ESs multiplexed in the AV clip have, and indicates the encoding method. Thus the program information indicates the encoding method, such as MPEG2-video or MPEG4-AVC, that is used to compress-encode the video stream.

[0528] The "characteristic point information" is information indicating, for each ES, where the characteristic points of a plurality of ESs multiplexed in the AV clip exist. The information indicating the characteristic point for each ES is called a "basic entry map".

[0529] What becomes the characteristic point is different for each type of stream. In the case of the base-view and dependent-view video streams, the characteristic point is the access unit delimiter that indicates the I-picture-type view component that is located at the start of the open GOP and closed GOP. In the case of the audio stream, the characteristic point is the access unit delimiter indicating the start positions of the audio frames that exist at regular intervals, for example, every one second. In the case of the PG and IG streams, the characteristic point is the access unit delimiter indicating the start positions of the display sets (display set of epoch start, display set of acquisition point) that are provided with all the functional segments necessary for the display, among the display sets of the graphics streams.

[0530] The ATC sequence and the STC sequence differ in how they represent the characteristic point. The ATC sequence represents the characteristic point by the source packet number. The STC sequence represents the characteristic point by using the PTS that indicates the time point on the STC time axis.

[0531] In view of the above-described differences, the basic entry map for each ES is composed of a plurality of entry points. More specifically, in each entry point constituting the entry map, a source packet number that indicates the location of the characteristic point in the ATC sequence is associated with a PTS that indicates the location of the characteristic point in the STC sequence. Further, each entry point includes a flag ("is_angle_change" flag) that indicates whether an angle change to the characteristic point is available. Since an angle change is available at the source packet located at the start of the interleave unit constituting the multi-angle section, the "is_angle_change" flag in the entry point indicating the starting source packet of the interleave unit is always set ON. Also, the entry point indicating the starting source packet of the interleave unit is associated with In_Time in the playitem information by the entry point.

[0532] The entry map for each ES indicates the source packet numbers of the characteristic points for respective stream types in correspondence with the PTSs. Accordingly, by referencing this entry map, it is possible to obtain, from an arbitrary time point in the ATC sequence, source packet numbers that indicate locations of the characteristic points for the ESs that are closest to the arbitrary time point.

[0533] This completes the explanation of the clip information file for 2D. Next is a detailed explanation of the clip information file for 3D. FIG 49B shows the internal structure of clip

information file for 3D. The clip information file for 3D includes: "clip dependent information (dependent-view management information)" which is clip information for the file dependent; and "clip base information (base-view management information)" which is clip information for the file base, as well as the "clip information for file 2D" that is regular clip information (management information). The reason is as follows. As described, the stereoscopic interleaved stream file is stored in a directory that is different from the directory in which the regular stream files are stored, to prevent them from mixing each other. Accordingly, the clip information files cannot be associated with the stereoscopic interleaved stream file. Thus the clip dependent information and the clip base information are stored in the clip information file for 2D.

[0534] The clip dependent information and the clip base information differ from the clip information file for 2D in that the clip dependent information and the clip base information include metadata that has the extent start point sequence.

[0535] As shown in FIG. 49B, the clip dependent information includes the extent start point sequence, and the clip base information also includes the extent start point sequence. The characteristic point information includes an entry map, and the extension data includes an extension entry map.

[0536] In the 3D output mode, the clip information file is divided into a clip base information file and a clip dependent information file.

[0537] FIG. 49C shows the clip base information file. The clip base information file includes clip base information and a basic entry map. The clip base information includes extent start point information.

[0538] FIG. 49D shows the clip dependent information file. The clip dependent information file includes clip dependent information and an extension entry map. The clip dependent information includes extent start point information.

[0539] A clip information file for the 2D output mode is stored under the directory for the clip information file (CLPI directory). The clip base information file is generated from the clip information file in the 3D output mode, and is treated to be stored in the clip information file for the 2D output mode.

[0540] A dummy clip information file is stored under the directory for the clip information file (CLPI directory). The dummy clip information file is assigned with a file name that is represented by a number corresponding to the file dependent, namely, a number generated by adding "1" to the identification number of the file 2D/file base. The clip dependent information file is generated in the 3D output mode from the clip information file corresponding to the file 2D, and is treated to be stored in the dummy clip information file. Suppose here that the clip information file in the 2D output mode is 00001.clpi, then the clip base information file in the 3D output mode is treated to be stored in 00001.clpi, and the clip dependent information file in the

3D output mode is treated to be stored in 00002.clpi.

<Extent start point>

[0541] The following explains the extent start point.

[0542] As described above, the stereoscopic interleaved stream file is composed of two clip AV streams (BDAV MPEG2 transport stream). The pair of extent start point information enables the stereoscopic interleaved stream file to be divided into two AV streams. The extent start point information is supplied as follows.

1. (1) An extent start point information table is supplied, to the playback device, in a piece of clip information that is referenced by a playitem of a playlist which includes a sub-path of "sub-path type = 8". It should be noted here that the sub-path of "sub-path type = 8" is an out-of-MUX dependent-view video stream playback path of an on-disc type.
2. (2) Another extent start point information table is supplied, to the playback device, in a piece of clip information that is referenced by a sub-playitem of a playlist which includes a sub-path of "sub-path type = 8".

[0543] When a flag in the playitem information (flug_is_multiangle_flag), which indicates whether a multi-angle section exists, is set ON, the extent start point information tables in a pair are supplied to the playback device, one in a piece of clip information that is referenced by an angle ID value, and the other in a piece of clip information that is referenced by a sub-clip entry ID value.

[0544] The extent start point information in the clip information file has the following data structure. The ID1 value and ID2 value in the extension data in ext_data_entry() should be set to 0x0002 and 0x0004, respectively.

[0545] The clip information file including the extent start point information tables needs to satisfy the following two conditions.

1. (a) The clip information file needs to be referenced by a playitem of a playlist which includes a sub-path of "sub-path type = 8".
2. (b) The clip information file needs to be referenced by a sub-playitem in a sub-path of "sub-path type = 8". Note that the sub-path of "sub-path type = 8" is an out-of-MUX dependent-view video stream playback path of an on-disc type.

[0546] FIG. 50 shows the correspondence among the clip information file, playlist, and stereoscopic interleaved stream file. On the right side of FIG 50, the stereoscopic interleaved

stream file is shown, and on the left side of FIG. 50, the clip information file is shown. In the middle of FIG 50, the first row shows the file base, the second row shows the clip base information file, the third row shows the 3D playlist, the fourth row shows the clip dependent information file, and the fifth row shows the file dependent.

[0547] The arrows bk1 and bk2 indicate that the file base and the file dependent are obtained respectively by dividing the stream file shown on the right side of the drawing.

[0548] The clip information file shown on the left side of FIG50 includes characteristic point information, extension data, clip base information, and clip dependent information. The arrows bk3 and bk4 indicate that the extent start point information tables in the clip base information and the clip dependent information enable the stereoscopic interleaved stream file to be divided.

[0549] The following explains the extent start point.

[0550] In the extent start point information of the clip information file, an ID1 value and an ID2 value in the extension data in `ext_data_entry()` should be set to 0x0002 and 0x0004, respectively.

[0551] The clip information file including the extent start point information needs to satisfy the following two conditions.

1. (i) The clip information file needs to be referenced by a playitem of a playlist which includes a sub-path having a sub-path type indicating 8.
2. (ii) The clip information file needs to be referenced by a sub-playitem in a sub-path having a sub-path type indicating 8. Note that the sub-path having a sub-path type indicating 8 is an out-of-MUX dependent-view video stream playback path of an on-disc type.

[0552] The stereoscopic interleaved stream file is composed of two clip AV streams (BD-AV MPEG2 transport streams). The pair of extent start point information enables the stereoscopic interleaved stream file to be divided into two AV streams. The extent start point information is supplied as follows.

[0553]

1. (1) An extent start point information table is stored in clip information that is referenced by a playitem of a playlist which includes a sub-path having a sub-path type indicating 8, so as to be supplied to the playback device.
2. (2) Another extent start point information table is stored in clip information that is referenced by a sub-playitem of a playlist which includes a sub-path having a sub-path type indicating 8, so as to be supplied to the playback device.

[0554] If the "is_multiangle" flag in the playitem is set to 1, the pair of the extent start point information tables is clip information referred by an angle ID value and clip information referred by a subclip entry ID value, respectively, so as to be supplied to the playback device.

[0555] FIGs. 51A and 51B show the internal structure of the clip base information and the clip dependent information. As shown in FIG. 51A, the clip base information and the clip dependent information include: "clip stream type information" indicating the stream type to which the corresponding AV clip belongs; "application type information" indicating the type to which the application composed of the corresponding AV clip belongs, such as a movie application, a time-base slide-show application, or a browsable slide-show application; "TS recording rate" indicating a transfer rate at which the TS packets in the AV clip are transferred in the playback device after the source packets pass through the source packet depacketizer; "number of source packets" indicating the number of source packets constituting the corresponding AV clip; "ATC delta" indicating a difference in ATC from the ATC sequence constituting the preceding AV clip; "extent start point information table"; and "extent start point information".

[0556] FIG 51B shows the internal structure of the extent start point information table. As shown in FIG. 51B, the extent start point information table includes "number_of_extent_start_point", and as many "SPN_extent_start_point" as the number indicated by the "number_of_extent_start_point".

[0557] The "number_of_extent_start_point" indicates the number of Extents that belong to the related AV stream file. The extent start point information tables in the clip base information and the clip dependent information in the same pair have the same value in the "number_of_extent_start_point".

[0558] The number of "SPN_extent_start"s (SPN_extent_start[0] through SPN_extent_start[number_of_extent_start_point]) is "number_of_extent_start_point + 1". Each SPN_extent_start is specified by the Extent identifier [extent_id], and is a 32-bit value that indicates a source packet number of the source packet that corresponds to the extent_idth Extent in the AV stream file.

[0559] The following explains the extension data of the clip information file. The extension data includes an extension entry map. The extension entry map, as is the case with the basic entry map, is composed of a plurality of entry points. More specifically, in each entry point constituting the extension entry map, a source packet number that indicates the location of the characteristic point in the ATC sequence is associated with a PTS that indicates the location of the characteristic point in the STC sequence. Each entry point further includes: a flag ("is_angle_change" flag) that indicates whether an angle change to the characteristic point is available; and information (I_size) that indicates the size of the intra picture located at the start of GOP. The extension entry map differs from the basic entry map in that the following

restrictions are imposed thereon.

[0560] When the extension entry map includes entries for the MPEG4-MVC view components, the extension entry map should also include entries for view components in correspondence with the PTSs in the extension entry map.

[0561] When there are two clip information files whose respective application types are "1" and "8" and which correspond to a stereoscopic interleaved stream file, the following conditions should be satisfied. That is to say, when an Extent identified by an Extent ID value of clip information with "application type = 1" (clip information of an application type for the primary video stream) includes a source packet that is to be referenced by PTS_EP_Start of the base-view video stream, an Extent identified by the same Extent ID value of clip information with "application type = 8" should include a source packet that is to be referenced by the same PTS_EP_Start value of the dependent-view video stream.

[0562] FIG. 52 shows the basic entry map and the extension entry map. In FIG. 52, the fifth row shows a plurality of pairs of a dependent-view data block and a base-view data block. The fourth row shows a sequence of source packets that constitute the dependent-view data blocks and the base-view data blocks. The first row shows the view components that are identified by the PTSs. The second row shows the basic entry map. The third row shows the extension entry map.

[0563] When Extent[1] specified by the extent start point with "Extent ID = 1" has a source packet [n1] with "SPN = n1" that is referenced by an entry with "PTS_EP_Start = t1" of the base-view video stream, Extent[1] specified by the extent start point with "Extent ID = 1", which is the same Extent ID of the clip information with "application type = 8", includes a source packet [n11] with "SPN = n11" that is referenced by an entry with "PTS_EP_Start = t1", which is an entry having the same value in the dependent-view video stream.

[0564] As apparent from this, when a source packet located at the start of GOP(i) of the base-view video stream and a source packet located at the start of GOP(i) of the dependent-view video stream belong to the same interleave Extent unit, entries pointing to the source packet located at the start of GOP(i) of the base-view video stream and the source packet located at the start of GOP(i) of the dependent-view video stream are added into each of the basic entry map and the extension entry map. Accordingly, by using both the basic entry map and the extension entry map, it is possible to ensure the continuous reading of the GOP(i) of the base-view video stream and the GOP(i) of the dependent-view video stream.

[0565] FIG. 53 shows entries that are not permitted in the extension entry map.

[0566] It is presumed here that a source packet [x] with "SPN = x" that is referenced by an entry with "PTS_EP_Start = x" of the base-view video stream exists at the start of a file base Extent that is referenced by an Extent ID = x, and that a source packet [y] with "SPN = y" that is referenced by an entry with "PTS_EP_Start = x" exists at the start of a file dependent Extent

that is referenced by an Extent ID = j, wherein "i" and "j" are different from each other.

[0567] It cannot be said that Extent [i] specified by the extent start point of the clip dependent with "Extent ID = i" includes a source packet with "SPN = x" that is referenced by an entry with "PTS_EP_Start = x", which is an entry of the base-view video stream having the same value. Therefore an entry with "PTS_EP_Start = x" cannot be added into the extension entry map.

[0568] When a source packet located at the start of GOP(i) of the base-view video stream and a source packet located at the start of GOP(i) of the dependent-view video stream belong to different interleave Extent units, an entry pointing to the source packet located at the start of GOP(i) is not added into any of the basic entry map and the extension entry map. In this case, GOP(i) of the base-view video stream and GOP(i) of the dependent-view video stream are excluded from the access destination of the random access. This prevents the access performance from being degraded.

[0569] FIG. 54 is a flow chart showing the playitem playback procedure.

[0570] In step S201, it is judged whether or not the current output mode is the 3D output mode. When the current output mode is the 2D output mode, a loop constituted from steps S203 through S206 is performed.

[0571] In step S203, the stream file, which is identified by: "xxxxx" described in Clip_information_file_name of the current playitem; and extension "m2ts", is opened. In step S204, the "In_time" and "Out_time" of the current playitem are converted into "Start_SPN[i]" and "End_SPN[i]" by using the entry map corresponding to the packet ID of the video stream.

[0572] In step S205, the Extents belonging to the reading range [i] are identified to read the TS packet with PID [i] from the Start_SPN[i] to the End_SPN[i]. In step S206, the drive of the recording medium is instructed to continuously read the Extents belonging to the reading range [i].

[0573] When the current output mode is the stereoscopic output mode, a loop constituted from steps S300 through S308 is performed.

[0574] In step S300, the stream file, which is identified by: "xxxxx" described in the Clip_information_file_name of the current playitem; and extension "ssif", is opened. In step S301, the base-view video stream is assigned to either the left-view or right-view video plane in accordance with the base-view indicator of the current playitem information, and the dependent-view video stream is assigned to the other, namely the left-view or right-view video plane that has not been assigned to the base-view video stream.

[0575] In step S302, the "In_time" and "Out_time" of the current playitem are converted to "Start_SPN[i]" and "End_SPN[i]" by using the basic entry map corresponding to the base-view video stream.

[0576] In step S303, the sub-playitem corresponding to the dependent-view stream is identified. In step S304, the "In_time" and "Out_time" of the identified sub-playitem are converted into "Start_SPN[j]" and "End_SPN[j]" by using the extension entry map corresponding to the dependent-view video stream.

[0577] The Extents belonging to the reading range [i] are identified to read the TS packet having the packet ID [i] constituting the base-view video stream from "Start_SPN[i]" to "End_SPN[i]" (step S305). The Extents belonging to the reading range [j] are identified to read the TS packet having the packet ID [j] from "Start_SPN[j]" to "End_SPN[j]" (step S306). Following this, in step S307, the Extents belonging to the reading ranges [i] and [j] are sorted in the ascending order. In step S308, the drive is instructed to continuously read the Extents belonging to the reading ranges [i] and [j] using the sorted addresses. After this, when the source packet sequence is read, in step S309, the base-view and dependent-view ATC sequences are restored and supplied to the PID filters for the base view and dependent view.

[0578] As described above, according to the present embodiment, when GOPs of the main TS and sub-TS are to be recorded onto the above-described recording medium, entries of the extension entry map point to only dependent-view picture data pieces that correspond to base-view picture data pieces pointed to by entries of the basic entry map as those that are to be played back at the same playback times as the dependent-view picture data pieces.

[0579] The picture data pieces pointed to by entries of the basic entry map and the picture data pieces pointed to by entries of the extension entry map make pairs in Extents. Accordingly, when an Extent is accessed via the basic entry map and the extension entry map, it is possible to play back each set of GOPs of the base view and dependent view corresponding to each other as one unit. This makes it possible to solve the problem of playback start delay.

[0580] Note that it may be possible to define that each extent include at least one entry point, as shown in FIG. 88A. With this definition, it is possible to prevent increase in length of an interval between entry points, thereby suppressing a delay amount for jump playback or the like as shown in FIG. 88B.

(Embodiment 4)

[0581] The present embodiment relates to an improvement for restoring the ATC sequence from the data blocks that constitute the stereoscopic interleaved stream file. FIG. 55 shows how the ATC sequence is restored from the data blocks constituting the stereoscopic interleaved stream file.

[0582] The fourth row of FIG. 55 shows a plurality of data blocks that constitute the stereoscopic interleaved stream file. The third row shows the source packet sequence

multiplexed in the main TS and the sub-TS.

[0583] The second row shows a set of STC sequence 2 constituting the dependent view, an entry map, and ATC sequence 2 constituting the dependent view. The first row shows a set of STC sequence 1 constituting the dependent view, an entry map, and ATC sequence 1 constituting the dependent view. The arrows extending from the third row to the first and the second rows schematically show that the ATC sequences 1 and 2 are restored from the data blocks of the two TSs (main TS and sub-TS) interleaved in the stereoscopic interleaved stream file. These ATC sequences are associated with the STC sequences by the entry map in the clip information.

[0584] This completes the description of the recording medium in the present embodiment. In the following, the playback device will be described in detail.

[0585] The playback device in the present embodiment has a structure in which the reading unit receives inputs of source packets from two recording mediums. For this purpose, the reading unit includes two drives and two read buffers. The two drives are used to access the two recording mediums, respectively. The two read buffers are used to temporarily store the source packets input from the two drives and output them to the decoder. An ATC sequence restoring unit is provided between the two drives and the two read buffers. The ATC sequence restoring unit separates the ATC sequence constituting the base-view stream and the ATC sequence constituting the dependent-view stream, from the source packets in the interleaved stream file read from one recording medium, and writes the two ATC sequences into the two read buffers, respectively. With this structure, the playback device can process the ATC sequence constituting the base-view video stream and the ATC sequence constituting the dependent-view video stream as if they have been read from different recording mediums, respectively.

[0586] FIGs. 56A and 56B show how the ATC sequence is restored. FIG. 56A shows the internal structure of the reading unit provided with the ATC sequence restoring unit. As described above, the ATC sequence restoring unit is provided between the two drives and the two read buffers. The arrow B0 symbolically indicates the input of the source packet from one drive. The arrow B1 schematically indicates the writing of the ATC sequence 1 constituting the base-view video stream. The arrow D1 schematically indicates the writing of the ATC sequence 2 constituting the dependent-view video stream.

[0587] FIG. 56B shows how the two ATC sequences obtained by the ATC sequence restoring unit are treated. In the middle part of FIG. 56B, the PID filters provided in the demultiplexing unit are shown. On the left side in the figure, the two ATC sequences obtained by the ATC sequence restoring unit are shown. The right side of the figure shows the base-view video stream, dependent-view video stream, left-eye PG stream, right-eye PG stream, base-view IG stream, and dependent-view IG stream, which are obtained by demultiplexing the two ATC sequences.

[0588] FIGs. 57A through 57D show one example of the extent start point information table in the base-view clip information and one example of the extent start point information table in the dependent-view clip information. FIG 57A shows the extent start point information table in the base-view clip information and the extent start point information table in the dependent-view clip information.

[0589] FIG. 57B shows base-view data blocks $B[0]$, $B[1]$, $B[2]$, ... $B[n]$ constituting the ATC sequence 1, and dependent-view data blocks $D[0]$, $D[1]$, $D[2]$, ... $D[n]$ constituting the ATC sequence 2. FIG 57C shows the number of source packets in the dependent-view data block and the number of source packets in the base-view data block.

[0590] FIG 57D shows a plurality of data blocks included in the stereoscopic interleaved stream file.

[0591] As shown in FIG. 57B, when the ATC sequence 2 is composed of the dependent-view data blocks $D[0]$, $D[1]$, $D[2]$, ... $D[n]$, the source packet numbers 0, b_1 , b_2 , b_3 , b_4 , ... b_n , which are relative to the dependent-view data blocks $D[0]$, $D[1]$, $D[2]$, ... $D[n]$ constituting the ATC sequence 2, are written in the `SPN_extent_start` in the extent start point information table of the file dependent.

[0592] When the ATC sequence 1 is composed of the base-view data blocks $B[0]$, $B[1]$, $B[2]$, ... $B[n]$, the number of source packets 0, a_1 , a_2 , a_3 , a_4 , ... a_n , which are relative to the base-view data blocks $B[0]$, $B[1]$, $B[2]$, .. $B[n]$ constituting the ATC sequence 1, are written in the `SPN_extent_start` in the extent start point information table of the file base.

[0593] FIG. 57C shows the number of source packets in an arbitrary dependent-view data block $D[x]$ and the number of source packets in an arbitrary base-view data block $B[x]$. When the starting source packet number of the dependent-view data block $D[x]$ is bx and the starting source packet number of the dependent-view data block $D[x + 1]$ is $bx + 1$, the number of source packets constituting the dependent-view data block $D[x]$ is " $(bx + 1) - bx$ ".

[0594] Similarly, when the starting source packet number of the base-view data block $B[x]$ is ax and the starting source packet number of the base-view data block $B[x + 1]$ is $ax + 1$, the number of source packets constituting the base-view data block $B[x]$ is " $(ax + 1) - ax$ ".

[0595] When the starting source packet number of the last base-view data block $B[n]$ in the stereoscopic interleaved stream file is " an " and the number of source packets constituting the ATC sequence 1 is "`number_of_source_Packet1`", the number of source packets constituting the base-view data block $B[n]$ is "`number_of_source_packet1 - an`".

[0596] When the starting source packet number of the last dependent-view data block $D[n]$ in the stereoscopic interleaved stream file is " bn " and the number of source packets constituting the ATC sequence 2 is "`number_of_source_packet2`", the number of source packets constituting the dependent-view data block $D[n]$ is "`number_of_source_packet2 - bn`".

[0597] FIG. 57D shows the starting source packet numbers of the dependent-view data blocks and the base-view data blocks in the present example.

[0598] In the stereoscopic interleaved stream file, the starting SPN of D[0] is "0" and the starting SPN of B[0] is "b1".

[0599] The starting SPN of D[1] is " $b1 + a1$ ", representing the sum of b1 (the number of source packets in the preceding dependent-view data block D[0]) and a1 (the number of source packets in the preceding base-view data block B[0]).

[0600] The starting SPN of B[1] is " $b2 + a1$ " ($= b1 + a1 + b2 - b1$), representing the sum of b1 (the number of source packets in the preceding dependent-view data block D[0]) and a1 (the number of source packets in the preceding base-view data block B[0]) and " $b2 - b1$ " (the number of source packets in the preceding dependent-view data block D[1]).

[0601] The starting SPN of D[2] is " $b2 + a2$ " ($= b1 + a1 + b2 - b1 + a2 - a1$), representing the sum of b1 (the number of source packets in the preceding dependent-view data block D[0]) and a1 (the number of source packets in the preceding base-view data block B[0]) and " $b2 - b1$ " (the number of source packets in the preceding dependent-view data block D[1]) and " $a2 - a1$ " (the number of source packets in the preceding base-view data block B[1]).

[0602] The starting SPN of B[2] is " $b3 + a2$ " ($= b1 + a1 + b2 - b1 + a2 - a1 + b3 - b2$), representing the sum of b1 (the number of source packets in the preceding dependent-view data block D[0]) and a1 (the number of source packets in the preceding base-view data block B[0]) and " $b2 - b1$ " (the number of source packets in the preceding dependent-view data block D[1]) and " $a2 - a1$ " (the number of source packets in the preceding base-view data block B[1]) and " $b3 - b2$ " (the number of source packets in the preceding dependent-view data block D[2]).

[0603] FIGs. 58A through 58C are illustrations provided for explanation of the source packet numbers of arbitrary data blocks in ATC sequences 1 and 2.

[0604] Suppose that an attempt is made to obtain a source packet number in a stereoscopic interleaved stream file in D[x] with a source packet number "bx", in the ATC sequence 2 shown in FIG. 58A. In this case, the starting source packet number of D[x] is " $bx + ax$ ", representing the sum of source packet numbers which are relative to data blocks D[0], B[0], D[1], B[1], D[2], B[2], ... D[x - 1], B[x-1], as shown in FIG 58B.

[0605] Suppose that an attempt is made to obtain a source packet number in a stereoscopic interleaved stream file in B[x] with a source packet number "ax", in the ATC sequence 1 shown in FIG. 58A. In this case, the starting source packet number of B[x] is " $bx + 1 + ax$ ", representing the sum of source packet numbers which are relative to data blocks D[0], B[0], D[1], B[1], D[2], B[2], ... D[x - 1], B[x - 1], D[x], as shown in FIG. 58B.

[0606] FIG. 58C shows a file base and a file dependent, wherein the Extents constituting the file base are the above-described base-view data blocks and the Extents constituting the file dependent are the above-described dependent-view data blocks.

[0607] The starting LBN and continuous length of EXT1[x] and EXT2[x] are obtained as follows, wherein EXT1[x] is an Extent of a file base corresponding to B[x], and EXT2[x] is an Extent of a file dependent corresponding to D[x].

[0608] The LBN can be obtained from the starting source packet number of D[x] by converting the source packet into the LBN by performing a calculation $((bx + ax) * 192/2048)$. Similarly, the LBN can be obtained from the starting source packet number of B[x] by converting the source packet into the LBN by performing a calculation $((bx + 1 + ax) * 192/2048)$. Here, the number "192" indicates the number of bytes representing the source packet size, and the number "2048" indicates the number of bytes representing the sector size (logical block size). The LBN of an Extent in the stereoscopic interleaved stream file that is closest to these LBNs can be obtained by using these converted LBNs as "file_offset" that is an argument of a function "SSIF_LBN(file_offset)". The function SSIF_LBN is a function that returns an LBN corresponding to the file_offset after tracing the allocation descriptors of the SSIF starting with the file_offset.

[0609] Accordingly, the starting LBN of EXT2[x] is represented as "SSIF_LBN $((bx + ax) * 192/2048)$ ". Also, the starting LBN of EXT1[x] is represented as "SSIF_LBN $((bx + 1 + ax) * 192/2048)$ ".

[0610] On the other hand, the continuous length of EXT2[x] is represented as "SSIF_LBN $((bx + 1 + ax) * 192/2048) - SSIF_LBN ((bx + ax) * 192/2048)$ ". Also, the continuous length of EXT1[x] is represented as "SSIF_LBN $((bx + 1 + ax + 1) * 192/2048) - SSIF_LBN ((bx + 1 + ax) * 192/2048)$ ". When file entries indicating these starting LBNs and continuous lengths are generated on a memory, it is possible to obtain file bases and file dependents virtually.

[0611] The demultiplexing performed by the two ATC sequences is based on the basic stream selection table and the extension stream selection table described in Embodiment 1. The ATC sequence restoring unit is realized by creating a program that causes the hardware resource to perform the process shown in FIG 59. FIG. 59 shows the procedure for restoring the ATC sequence.

[0612] In step S91, the ATC sequence for base-view is set as the ATC sequence 1, and the ATC sequence for dependent-view is set as the ATC sequence 2. In step S92, the variable "x" is initialized to "1". The variable "x" specifies a base-view data block and a dependent-view data block. After this, the control enters a loop in which steps S94 through S96 are repeatedly performed as follows.

[0613] It is judged whether or not a source packet number bx specified by the variable "x" is equal to a source packet number bn specified by the last numeral "n" of the base-view data

block (step S93). When the result of the judgment is in the negative (No in step S93), the source packets from the source packet ($bx + ax$), which is specified by the source packet number " $bx + ax$ ", to the source packet immediately before the source packet ($b_{x+1} + ax$) specified by the source packet number " $b_{x+1} + ax$ " are added into the ATC sequence 2 (step S94). Then, the source packets from the source packet ($bx + 1 + ax$) to the source packet immediately before the source packet ($bx + 1 + ax + 1$) are added into the ATC sequence 1 (step S95). And then the variable " x " is incremented (step S96). These steps are repeated until it is judged Yes in step S93.

[0614] When it is judged Yes in step S93, as many source packets as the number specified by "number_of_source_packet2-bn" starting from the source packet number "bn" are added into the ATC sequence 2 (step S97). And as many source packets as the number specified by "number_of_source_packet1-bn" starting from the source packet number "an" are added into the ATC sequence 1 (step S98).

[0615] After the ATC sequences 1 and 2 are restored through the above-described steps, the file base is virtually opened by generating, in the memory, the file entry that indicates the start LBN of the base-view data block and the continuation length (step S99). Similarly, the file dependent is virtually opened by generating, in the memory, the file entry that indicates the start LBN of the dependent-view data block and the continuation length (step S100).

<Technical meaning of opening file base>

[0616] When a random access from an arbitrary time point is to be performed, a sector search within a stream file needs to be performed. The sector search is a process for identifying a source packet number of a source packet corresponding to the arbitrary time point, and reading a file from a sector that contains a source packet of the source packet number.

[0617] Since the size of one Extent constituting the stereoscopic interleaved stream file is large, the sector search requires a wide range of searching. In that case, when a random access from an arbitrary time point is performed, it may take a long time to identify the reading-target sector.

[0618] This is because, in the interleaved stream file, data blocks constituting the base-view video stream and the dependent-view video stream are disposed in the interleaved manner to constitute one long Extent, and the allocation descriptor of the file entry of the interleaved stream file merely indicates the start address of the long Extent.

[0619] In contrast, the file base is composed of a plurality of short Extents, and the start address of each Extent is written in the allocation descriptor. As a result, the sector search requires a narrow range of searching. Thus, when a random access from an arbitrary time

point is performed, the reading-target sector can be identified in a short time.

[0620] That is to say, since the data blocks constituting the base-view video stream are managed as Extents of the file base, and the start address of the data block is written in the allocation descriptor in the file entry corresponding to the file base, it is possible to quickly reach the sector including the source packet at the target random access position, by starting the sector search from the start address of the Extent that contains the target random access position.

[0621] With the above-described structure in which the data blocks constituting the base-view video stream are managed as Extents of the file base, and the start address of each Extent and the continuation length are written in the allocation descriptor in the file entry corresponding to the file base, it is possible to perform a random access from an arbitrary time point in the base-view video stream at a high speed.

[0622] More specifically, the sector search is performed as follows. First, the entry map corresponding to the base-view video stream is used to detect a source packet number that is the random access position corresponding to the arbitrary time point.

[0623] Next, the extent start point information in the clip information corresponding to the base-view video stream is used to detect an Extent that contains the source packet number that is the random access position.

[0624] Further, the allocation descriptor in the file entry corresponding to the file base is referenced to identify the start sector address of the Extent that contains the source packet number that is the random access position. Then a file read is performed by setting a file pointer to the start sector address, and a packet analysis is executed onto the read source packet to identify the source packet with the source packet number that is the random access position. Then the identified source packet is read. With this procedure, the random access to the main TS is executed efficiently. This also applies to the sub-TS.

[0625] As described above, according to the present embodiment, Extents of the base-view video stream and the dependent-view video stream in the interleaved stream file are supplied to the demultiplexing unit and the decoder after they are rearranged based on the extent start point information. Thus the decoder and program can treat, as the files virtually existing on the recording medium, the file base storing the base-view video stream and the file dependent storing the dependent-view video stream.

[0626] In this structure, the base-view video stream and the dependent-view video stream for the stereoscopic viewing are recorded on the recording medium, while the base-view video stream and the dependent-view video stream can be accessed separately. With this structure, the processing efficiency of the playback device is improved.

(Embodiment 5)

[0627] In the present embodiment, the following describes a problem of HDMI output, a super-resolution method, and a method of increasing frame rate in order to play back 3D contents stored in a BD-ROM.

(HDMI Output)

[0628] Firstly, identification of 3D display/glasses method is described with reference to FIG. 60. In the case where a plurality of TVs are connected with a 3D displayable player, it is desirable to notify the player, via an I/F such as HDMI, of whether glasses are necessary for 3D viewing with use of each TV, and if necessary, which type of glasses are necessary. For example, if it is recognizable in a BD player that active shutter glasses are necessary for 3D viewing with use of a TV connected with the BD player, it is possible to program such that before 3D playback is started, a viewer is informed of a message that indicates active shutter glasses are necessary for 3D viewing with use of the connected TV. Accordingly, it is desirable that apparatuses connected with the BD player via E-EDID, InfoFrame, or the like share information for identifying whether glasses are necessary for 3D viewing with use of each TV, and if necessary, which type of glasses are necessary (anaglyph, circular deflection, or active shutter, for example), such that the BD player is informed of the information. If a communication function between each TV and the glasses is prepared, it is possible to appropriately change 3D videos for each user by providing the player with position information of the glasses (specifically, the vertical line extending from the center of the TV screen and the horizontal/vertical angle of the glasses). In the case where a TV-1 is a 2D dedicated TV and a TV-2 is a 3D dedicated TV, it is desirable, as output for the TV-1, to extract and output only a right eye video or a left-eye video to be output to the TV-2, or display a message indicating that 3D viewing is impossible in the TV-1 such as a message "3D is being played back in TV-2" or a message "3D viewing is impossible in TV-1 ". As shown in FIG. 61, in the case where a playback video is switched to a 2D video to a 3D video, it is desirable to doubly output only one of a right-eye video and a left-eye video at a 3D video frame rate with no change. This is because change of the frame rate causes delay due to the necessity of authentication of HDMI, for example. However, in the case where a 3D video is displayed, in consideration of that glasses darken the user's eyesight, a video to be displayed in a TV has a high brightness level, and appropriate processing for performing 2D video display might not be performed on the video. In view of this, in the case where only one of a right-eye video and a left-eye video is doubly output like a section "2D dubbing playback" shown in FIG. 61, when a video is output via HDMI, a flag indicating that playback. Accordingly, it is possible to judge, in the TV, a video to be transferred is a 2D video. This enables control such as image processing appropriate for 2D video playback. Next, the correlation between subtitles and menu streams used for BD is described with reference to FIG. 62. Suppose that a player needs to simultaneously perform 2D output and 3D output. Since presentation Graphics (a stream for subtitle, and is abbreviated as "PG") and Interactive Graphics stream (a stream for menu, and is abbreviated as "IG") that are used in BD-ROMs are displayed as different patterns, PG and IG each have a

different PID and accordingly need to be decoded separately. However, in order to prevent occurrence of the difference between the user's operation while viewing 2D display on a TV and the viewer's operation while viewing 3D viewing on the TV, it is desirable that all pieces of information of 2D stream (C) correspond to all pieces of information of 3D left-eye/right-eye streams (L,R), except all patterns of 2D stream (C) and display positions thereof and all patterns of 3D left-eye/right-eye streams (L,R) and display positions thereof. For example, with respect to PG, it is desirable that streams are recorded such that the same subtitle is displayed at the same display time among C, L, and R (that is, only patterns of subtitle and display information are different among C, L, and R). Also, with respect to IG, it is desirable that streams are recorded such that the page structure of a menu, the shift among buttons, button commands in execution, and the like are the same among corresponding C, L, and R (that is, only patterns of subtitle and display information are different among C, L, and R), thereby to realize the same menu operation among C, L, and R. This is necessary to realize that a 2D viewer and a 3D viewer can the same subtitle and menu in the TV-1 and the TV-2, respectively, and menu can be controlled by both the TV-1 and the TV-2. In order to perform simultaneous output, the player separately overlays L, R, and C to generate 2D video and 3D video, as shown in FIG. 62. Although only an IG decoder is shown in FIG. 62, the same applies to a PG decoder. Also, instead of using an L video for 3D for 2D video output, R video may be used.

(Super-Resolution)

[0629] Next, processing for realizing more smooth 2D/3D video display is described, with reference to FIG. 63. In a TV that performs 2D display based on the input in which a left-eye (L_n) video and a right eye (R_n) video alternately succeed, such as $L_1, R_1, L_2, R_2, \dots$, either left-eye videos or right-eye videos are displayed, such as $L_1, L_2, L_3, \dots$ or $R_1, R_2, R_3, \dots$. Accordingly, in the case where a right-eye video and a left-eye video are simultaneously filmed according to a conventional art, only the image quality of the video L_2 is increased using a video (L_1) temporally previous to the video L_2 and a video (L_3) temporally subsequent to the video L_2 . However, there is a case that a L_2 video is more highly associated with R_1, R_2 , and R_3 videos for the eye on the other side than L_1 and L_3 videos. Accordingly, even in the case where 2D display is performed, it is possible to further increase the image quality by referring to a video that is not displayed. The method of increasing the image quality is no object here. It is greatly important to use either a right-eye video or a left-eye video that is not displayed (both of the right-eye video and the left-eye video after all) in processing of increasing the image quality while performing 2D display, in order to increase the image quality with a high precision. In this case, although 2D display is performed for the viewer, the player and the TV need to be connected with each other such that 3D display is performed. It is necessary to perform connection authentication as 3D with respect to 1/F such as HDMI to perform switch control to cause the player to output both of L/R videos. In the case of a TV performing 3D display based on left-eye videos (L_n) and right-eye videos (R_n) such as $L_1, R_1, L_2, R_2, \dots$ that are sequentially input, it is considered to be effective to use both of L/R videos is effective in order to increase the image quality of each of the videos. Also, it is possible to estimate, with a high accuracy, a frame to be used for performing increasing the image quality by recording the

optical parameter of the L/R camera (angle between cameras, the focal distance, or the like) in a stream.

(Increase in Frame Rate)

[0630] Next, the following describes processing for realizing more smooth 2D/3D video display, with reference to FIG. 63 again. Especially, when 3D display is realized, right-eye videos and left-eye videos are often alternately displayed in accordance with the time division display method. This tends to cause the user to suffer from eye strain due to the low frame rate for display. Furthermore, in the case where there is a great change amount of video materials among frames due to the recent increase in the screen size of TV, the user tends to feel tired due to the low frame rate. Accordingly, in the case where 3D playback is performed, playback is normally performed at a frame rate such as a frame rate twice or three times the normal frame rate of a video material. However, even if display is performed at a two or three times faster than normal, a video being displayed is the same as the video being displayed at the normal speed. This still remains a problem caused by a change amount of videos among frames in viewing by a large screen. For example, in the case where display is performed at twice faster than normal, left-eye videos and right-eye videos are displayed two sets for each time such as L1, R1, L1, R1, L2, R2, L2, R2, At the second display of L1, R1, it is possible to reduce the user's tired feeling by displaying videos (an intermediate video of L1 and L2 and an intermediate video of R1 and R2) whose time resolutions have been increased using a high image quality circuit. In other words, it is possible to reduce the user's eye strain in 3D viewing by a large screen by generating an intermediate video at a sampling frequency higher than a sampling frequency (frame rate) of a video material and performing 3D display.

[0631] Note that in the case of a TV that performs stereoscopic display by displaying parallax images for the left eye and right eye in which a viewer needs to wear glasses, when parallax images for the left eye and right eye become larger than a distance between the right eye and the left eye of the viewer, a picture as 3D is not composed. This causes a problem that the user suffers from eye strain and 3D sickness. Accordingly, as a TV display processing, it is desirable to shift the right videos and the left videos as a whole to the right or the left for display, such that parallax images for the left eye and right eye are not misaligned beyond a distance between a right eye and a left eye corresponding to the smallest glasses among commercially available ones. The player may shift the right videos and the left videos as a whole to the right or the left to perform output processing, such that an appropriate parallax difference is displayed by causing the viewer to input or select his age or desired 3D strength on an interactive menu screen of the BD.

(Embodiment 6)

[0632] The present embodiment describes the production of the recording mediums

described in the embodiments so far, namely, the production act of the recording medium.

[0633] The recording method of the present embodiment can be realized as a real-time recording in which AV files (stream files) and non-AV files (files other than the stream files) are generated in real time, and are written directly into the AV data recording region and the non-AV data recording region provided in the recording medium. However, not limited to this, the recording method of the present embodiment can be realized as a pre-format recording in which bit streams to be recorded into the volume region are generated in advance, a master disc is generated based on the bit streams, and the master disc is pressed, thereby making possible a mass production of the optical disc. The recording method of the present embodiment is applicable to either the real-time recording or the pre-format recording.

[0634] When the recording method is to be realized by the real-time recording technology, the recording device for performing the recording method creates an AV clip in real time, and stores the AV clip into the BD-RE, BD-R, hard disk, or semiconductor memory card.

[0635] In this case, the AV clip may be a transport stream that is obtained as the recording device encodes an analog input signal in real time, or a transport stream that is obtained as the recording device partializes a digital input transport stream. The recording device for performing the real-time recording includes: a video encoder for obtaining a video stream by encoding a video signal; an audio encoder for obtaining an audio stream by encoding an audio signal; a multiplexor for obtaining a digital stream in the MPEG2-TS format by multiplexing the video stream, audio stream and the like; and a source packetizer for converting TS packets constituting the digital stream in the MPEG2-TS format into source packets. The recording device stores an MPEG2 digital stream having been converted into the source packet format, into an AV clip file, and writes the AV clip file into the BD-RE, BD-R, or the like. When the digital stream is written, the control unit of the recording device performs a process of generating the clip information and the playlist information in the memory. More specifically, when the user requests a recording process, the control unit creates an AV clip file and an AV clip information file in the BD-RE or the BD-R.

[0636] After this, when the starting position of GOP in the video stream is detected from the transport stream which is input from outside the device, or when the GOP of the video stream is created by the encoder, the control unit of the recording device obtains (i) the PTS of the intra picture that is positioned at the start of the GOP and (ii) the packet number of the source packet that stores the starting portion of the GOP, and additionally writes the pair of the PTS and the packet number into the entry map of the clip information file, as a pair of EP_PTS entry and EP_SPN entry. After this, each time a GOP is generated, a pair of EP_PTS entry and EP_SPN entry is written additionally into the entry map of the clip information file. In so doing, when the starting portion of a GOP is an IDR picture, an "is_angle_change" flag having been set to "ON" is added to a pair of EP_PTS entry and EP_SPN entry. Also, when the starting portion of a GOP is not an IDR picture, an "is_angle_change" flag having been set to "OFF" is added to a pair of EP_PTS entry and EP_SPN entry.

[0637] Further, the attribute information of a stream in the clip information file is set in accordance with the attribute of the stream to be recorded. After the clip and the clip information are generated and written into the BD-RE or the BD-R, the playlist information defining the playback path via the basic entry map in the clip information is generated and written into the BD-RE or the BD-R. When this process is executed with the real-time recording technology, a hierarchical structure composed of the AV clip, clip information, and playlist information is obtained in the BD-RE or the BD-R.

[0638] This completes the description of the recording device for performing the recording method by the real-time recording. Next is a description of the recording device for performing the recording method by the pre-format recording.

[0639] The recording method by the pre-format recording is realized as a manufacturing method of an optical disc including an authoring procedure.

[0640] FIGs. 64A and 64B show a recording method of an optical disc. FIG. 64A is a flow chart of the recording method by the pre-format recording and shows the procedure of the optical disc manufacturing method. The optical disc manufacturing method includes the authoring step, signing step, medium key obtaining step, medium key encrypting step, physical format step, identifier embedding step, a mastering step, and replication step.

[0641] In the authoring step S201, a bit stream representing the whole volume region of the optical disc is generated.

[0642] In the signing step S202, a request for signature is made to the AACS LA to manufacture the optical disc. More specifically, a portion is extracted from the bit stream is sent to the AACS LA. Note that the AACS LA is an organization for managing the license of the copyrighted work protection technologies for the next-generation digital household electric appliances. The authoring sites and mastering sites are licensed by the AACS LA, where the authoring sites perform authoring of optical discs by using authoring devices, and the mastering sites execute mastering by using mastering devices. The AACS LA also manages the medium keys and invalidation information. The AACS LA signs and returns the portion of the bit stream.

[0643] In the medium key obtaining step S203, a medium key is obtained from the AACS LA. The medium key provided from the AACS LA is not fixed. The medium key is updated to a new one when the number of manufactured optical discs reaches a certain number. The update of the medium key makes it possible to exclude certain makers or devices, and to invalidate an encryption key by using the invalidation information even if the encryption key is cracked.

[0644] In the medium key encrypting step S204, a key used for encrypting a bit stream is encrypted by using the medium key obtained in the medium key obtaining step.

[0645] In the physical format step S205, the physical formatting of the bit stream is performed.

[0646] In the identifier embedding step S206, an identifier, which is unique and cannot be detected by ordinary devices, is embedded, as electronic watermark, into the bit stream to be recorded on the optical disc. This prevents mass production of pirated copies by unauthorized mastering.

[0647] In the mastering step S207, a master disc of the optical disc is generated. First, a photoresist layer is formed on the glass substrate, a laser beam is radiated onto the photoresist layer in correspondence with desired grooves or pits, and then the photoresist layer is subjected to the exposure process and the developing process. The grooves or pits represent values of the bits constituting the bit stream that has been subjected to the eight-to-sixteen modulation. After this, the master disc of the optical disc is generated based on the photoresist whose surface has been made uneven by the laser cutting in correspondence with the grooves or pits.

[0648] In the replication step S208, copies of the optical disc are produced by a mass production by using the master disc of the optical disc.

[0649] FIG. 64B shows the procedure of the recording method by the pre-format recording when a general user records any of the various files described in the embodiment so far onto a recording medium such as BD-R or BD-RE by using a personal computer, not when the optical disc is mass-produced. Compared with FIG. 64A, in the recording method shown in FIG. 64B, the physical format step S205 and the mastering step S207 have been omitted, and each file writing step S209 has been added.

[0650] Next, the authoring step is explained.

[0651] FIG 65 is a flow chart showing the procedure of the authoring step.

[0652] In step S101, the reel sets of the main TS and sub-TS are defined. A "reel" is a file which stores the material data of an elementary stream. In the authoring system, the reels exist on a drive on a local network. The reels are data representing, for example, L and R images shot by a 3D camera, audio recorded at the shooting, audio recorded after the shooting, subtitles for each language, and menus. A "reel set" is a group of links to the material files, representing a set of elementary streams to be multiplexed into one transport stream. In this example, a reel set is defined for each of the main TS and the sub-TS.

[0653] In step S102, the prototypes of playitem and sub-playitem are defined, and the prototypes of the main path and sub-path are defined by defining a playback order of playitem and sub-playitem. The prototype of the playitem can be defined by receiving, via a GUI, a specification of a reel that is permitted to be played back by a targeted playitem in the monoscopic output mode, and a specification of In_Time and Out_Time. The prototype of the sub-playitem can be defined by receiving, via a GUI, a specification of a reel that is permitted to be played back by a playitem corresponding to a targeted sub-playitem in the stereoscopic

output mode, and a specification of In_Time and Out_Time.

[0654] For the specification of a reel to be permitted to be played back, a GUI is provided to make it possible to check a check box corresponding to, among the links to the material files in the reel set, a link to a material file permitted to be played back. With this GUI, numeral input columns are displayed in correspondence with the reels. With use of the numeral input columns, the priority of each reel is received, and based on this, the priorities of the reels are determined. With the setting of the reels permitted to be played back and the setting of the priorities, the stream selection table and the extension stream selection table are generated.

[0655] The specification of In_Time and Out_Time is performed when the recording device executes the process in which the time axis of the base-view video stream or the dependent-view video stream is displayed as a graphic on the GUI, a slide bar is moved on the graphic of the time axis, and specification of a positional setting of the slide bar is received from the user.

[0656] The definition of the playback order of the playitem and the sub-playitem is realized by the following process: a picture at In_Time of the playitem is displayed as a thumbnail on the GUI, and the recording device receives from the user an operation made onto the thumbnail to set the playback order.

[0657] In step S103, a plurality of elementary streams are obtained by encoding the material files specified by the reel sets. The plurality of elementary streams include the base-view video stream and the dependent-view video stream, and the audio stream, PG stream, and IG stream that are to be multiplexed with the base-view video stream and the dependent-view video stream.

[0658] In step S104, one main TS is obtained by multiplexing therein the base-view video stream and an elementary stream which, among the elementary streams obtained by the encoding, belongs to same reel set as the base-view video stream.

[0659] In step S105, one sub-TS is obtained by multiplexing therein the dependent-view video stream and an elementary stream which, among the elementary streams obtained by the encoding, belongs to the same reel set as the dependent-view video stream.

[0660] In step S106, the prototype of the clip information file is created based on the parameters having been set during the encoding and multiplexing.

[0661] In step S107, the playlist information is defined by generating the playitem information and the sub-playitem information based on the prototype of the playitem, and then generating the main path information and the sub-path information by defining the playback order based on the playitem information and the sub-playitem information.

[0662] In the generation of the playitem information, the stream selection table is generated in the playitem information so that, among the elementary streams multiplexed in the main TS,

elementary streams that are defined, in the basic structure of the playitem, to be played back in the monoscopic output mode are set to "playable". Also, to define the playback section in the base-view video stream, the In_Tlme and Out_Tlme having been defined by the above-described editing are written in the playitem information.

[0663] In the generation of the sub-playitem information, the extension stream selection table is generated in the extension data of the playlist information so that, among the elementary streams multiplexed in the sub-main TS, elementary streams that are defined, in the basic structure of the playitem, to be played back in the stereoscopic output mode are set to "playable". The playitem information and the sub-playitem information are defined based on information in the clip information file, and thus are set based on the prototype of the prototype of the clip information file.

[0664] In step S108, the main TS, sub-TS, prototype of the clip information file, and prototype of the playlist information are converted into a directory file group in a predetermined application format.

[0665] Through the above-described processes, the main TS, sub-TS, clip information, playitem information, and sub-playitem information are generated. Then the main TS and the sub-TS are converted into respective independent stream files, the clip information is converted into the clip information file, and the playitem information and the sub-playitem information are converted into the playlist information file. In this way, a set of files to be recorded onto the recording medium are obtained.

[0666] After this, when the video stream encoding step is executed, the plane offset value and the offset direction information obtained the above-described conversion are written into the metadata of each GOP. In this way, the offset sequence can be generated in the encoding process.

[0667] FIG. 66 is a flow chart showing the procedure for writing the AV file. The AV files are written according to this flow chart when the recording method by the real-time recording or the recording method including the mastering or replication is implemented.

[0668] In step S401, the recording device generates the file entry in the memory of the recording device by creating "xxxxx.ssif". In step S402, it is judged whether the continuous free sector regions have been ensured. When the continuous free sector regions have been ensured, the control proceeds to step S403 in which the recording device writes the source packet sequence constituting the dependent-view data block into the continuous free sector regions as much as EXT2[i]. After this, steps S404 through S408 are executed. When it is judged in step S402 that the continuous free sector regions have not been ensured, the control proceeds to step S409 in which the exceptional process is performed, and then the process ends.

[0669] The steps S404 through S408 constitute a loop in which the process of steps S404-

S406 and S408 is repeated until it is judged "NO" in step S407.

[0670] In step S405, the recording device writes the source packet sequence constituting the base-view data block into the continuous free sector regions as much as EXT1[i]. In step S406, it adds, into the file entry, the allocation identifier that indicates the start address of the source packet sequence and continuation length, and registers it as an Extent. In connection with this, it writes, into the metadata in the clip base information and the clip dependent information, the Extent start point information that indicates the start source packet number thereof.

[0671] The step S407 defines the condition for ending the loop. In step S407, it is judged whether or not there is a non-written source packet in the base-view and dependent-view data blocks. When it is judged that there is a non-written source packet, the control proceeds to step S408 to continue the loop. When it is judged that there is no non-written source packet, the control proceeds to step S410.

[0672] In step S408, it is judged whether or not there are continuous sector regions. When it is judged that there are continuous sector regions, the control proceeds to step S403. When it is judged that there are no continuous sector regions, the control returns to step S402.

[0673] In step S410, "xxxxx.ssif" is closed and the file entry is written onto the recording medium. In step S411, "xxxxx.m2ts" is created and the file entry of "xxxxx.m2ts" is generated in the memory. In step S412, the allocation descriptor that indicates the continuation length and the start address of Extent of the base-view data block unique to the file 2D is added into the file entry of "xxxxx.m2ts". In step S413, "xxxxx.m2ts" is closed and the file entry is written.

[0674] In step S404, it is judged whether or not there is a long jump occurrence point in the range of "EXTss + EXT2D". In the present example, it is presumed that the long jump occurrence point is a boundary between layers. When it is judged that there is a long jump occurrence point in the range of "EXTss + EXT2D", the control proceeds to step S420 in which a copy of the base-view data block is created, and base-view data blocks B[i]ss and B[i]2D are written into the region immediately before the long jump occurrence point, and then the control proceeds to step S406. These become Extents of the file 2D and Extents of the file base.

[0675] The following explains specific values of EXT2D, EXT1[n], EXT2[n], and EXTss[n].

[0676] The lowermost value of EXT2D is determined so that, when a playback in the 2D output mode is performed, a buffer under flow does not occur in the read buffer of the playback device during a jump period from each base-view data block to the next base-view data block.

[0677] The lowermost value of EXT2D is represented by the following expression for Condition 1, when it takes Tjump2D(n) of time when a jump from the nth base-view data block to the (n+1)th base-view data block is made, each base-view data block is read into the read buffer at a speed of Rud2D, and the base-view data block is transferred from the read buffer to

the video decoder at an average speed of R_{bext2D} .

<Condition 1>

[0678]

$$\{\text{Lowermost value of EXT2D}\} \geq (R_{ud2D} + R_{bext2D}) / (R_{ud2D} - R_{bext2D}) \times T_{jump2D}$$

(n)

[0679] It is presumed here that an Extent corresponding to a base-view data block $B[n]_{ss}$ is represented as $EXT1[n]$. In this case, the lowermost value of $EXT1[n]$ is determined so that, when a playback in the B-D presentation mode is performed, a buffer under flow does not occur in the double buffer during a jump period from each base-view data block to the next dependent-view data block, and during a jump period from the dependent-view data block to the next base-view data block.

[0680] In the present example, the double buffer is composed of a read buffer 1 and a read buffer 2. The read buffer 1 is the same as the read buffer provided in the 2D playback device.

[0681] It is presumed here that, when a playback in the B-D presentation mode is performed, it takes $T_{fjump3D}(n)$ of time when a jump from the n^{th} base-view data block to the p^{th} dependent-view data block is made, and it takes $T_{Bjump3D}(n)$ of time when a jump from the p^{th} dependent-view data block to the $(n+1)^{th}$ base-view data block is made.

[0682] It is further presumed that, each base-view data block is read into the read buffer 1 at a speed of R_{ud3D} , each dependent-view data block is read into the read buffer 2 at the speed of R_{ud3D} , and the base-view data block is transferred from the read buffer 1 to the video decoder at an average speed of R_{bext3D} . Then the lowermost value of $EXT1[n]$ is represented by the following expression for Condition 2. The continuation length of the big Extents is set to a value that is equal to or higher than the lowermost value.

<Condition 2>

[0683]

$$\{\text{Lowermost value of EXT1}[n]\} \geq (R_{ud3D} \times R_{bext3D}) / (R_{ud3D} - R_{bext3D}) \times (T_{fjump3D}(n) + EXT2[n] / (R_{ud3D} + T_{Bjump3D}(n)))$$

The lowermost value of $EXT2$ is determined so that, when a playback in the B-D presentation mode is performed, a buffer under flow does not occur in the double buffer of the playback device during a jump period from each dependent-view Extent to the next base-view data Extent, and during a jump period from the base-view Extent to the next dependent-view Extent.

[0684] The lowermost value of EXT2[n] is represented by the following expression for Condition 3, when it takes Tjump3D(n+1) of time when a jump from the (n+1)th base-view data block to the (p+1)th dependent-view data block is made, and the dependent-view data block is transferred from the read buffer 2 to the decoder at an average speed of Rdext3D.

<Condition 3>

[0685]

[Lowermost value of EXT2[n]] $\geq$ (Rud3D + Rbext3D) / (Rud3D - Rdext3D) $\times$ (Tjump3D(n) + EXT2[n+1] / (Rud3D + TFjump3D(n+1)))

<Specific values of EXTSS>

[0686] When a jump from a reading of an Extent to the next Extent is to be made, the buffer should be occupied by a sufficient amount of data immediately before the jump. Accordingly, when a stereoscopic interleaved stream file is to be read, the read buffer needs to store one Extent, and occurrence of a buffer under flow should be avoided.

[0687] However, the "EXTSS" needs to be determined based not only on "Tjump", a time period taken when a jump from an Extent to another Extent, but on "Tdiff". It should be noted here that the "Tdiff" represents a delay time that occurs in connection with a preloading of dependent-view data blocks in EXTss and a preloading of dependent-view data blocks in EXTssnext. The following further explains the meaning of Tdiff. When a stereoscopic interleaved stream file is read while the starting dependent-view data block is being preloaded.

[0688] In EXTss, the playback is delayed as much as the time period required for preloading the dependent-view data block. Here, the time period required for preloading the starting dependent-view data block in EXTss is referred to as "delay period" because the playback is delayed as much as the period.

[0689] On the other hand, in EXTssnext, immediately after a jump from EXTss to EXTssnext is made, the starting dependent-view data block is preloaded. Thus the playback by the video decoder is allowed to be delayed for the period of the preloading. Therefore the time period in which the starting dependent-view data block is preloaded in the playback of EXTssnext is referred to as "grace period" because the start of playback by the video decoder is allowed to be delayed for the period.

[0690] In view of this, a value of Tdiff is obtained by subtracting the delay period from the grace period of the dependent-view data block. More specifically, the value Tdiff is calculated using the following expression.

$$Tdiff = \max((S1+EXT1)EXTSSnext) - (S1+EXT1)EXTSS \times 1000 \times 01/D...1771$$

$$T_{diff} = \text{ceil}[(S1stEXT2[i]EXTss - S1stEXT2[i]EXTssnext) \times 1000 \times 8 / Rud72]$$

[0691] In the above expression, Tdiff means a difference between the time period for reading S1stEXT2[i]EXTss and the time period for reading S1stEXT2[i]EXTssnext; S1stEXT2[i]EXTss represents the size of EXT2[i] which is located at the start of EXTss; S1stEXT2[i]EXTssnext represents the size of EXT2[i] which is located at the start of EXTssnext. EXTssnext is an Extent in the stereoscopic interleaved stream file, is located immediately after EXTss, and is played back seamlessly with EXTss.

[0692] With use of Tdiff and Tjump, which is a time period required for jump to EXTssnext, Sextss, which is the minimum Extent size based on the average bit rate in each Extent, is calculated as a value satisfying the following Condition 4.

<Condition 4>

[0693]

$$\text{SextSS[Byte]} \geq \text{ceil}[(T_{\text{jump}} + T_{\text{diff}} \times Rud72) / (1000 \times 8)] \times (\text{Rextss} \times 192) / (Rud72 \times 188 - \text{Rextss} \times 192)$$

[0694] In the above Condition 4, Rud72 represents a data rate in transfer from the BD-ROM drive in the stereoscopic output mode.

[0695] Rextss represents an average bit rate in EXTss and is obtained using the following expressions.

$$\text{Rextss} = \text{ceil}[\text{Nsp} \times 188 \times 8 / (\text{ATCDextss} / 27000000)]$$

$$\text{ATCDextss} = \text{ATCstart_EXTssnext} - \text{ATCstart_EXTss}$$

$$\text{ATCDextss} = \text{ATClast_EXTss} - \text{ATCstart_EXTss} + \text{ceil}(27000000 \times 188 \times 8 / \min(\text{Rts1}, \text{Rts2}))$$

[0696] In the above expressions, ATCDextss represents the ATC period of EXTss.

[0697] ATCstart_EXTss represents the minimum ATC value specified by the ATC field of the source packet sequence in EXTss.

[0698] ATCstart_EXTssnext represents the minimum ATC value specified by the ATC field of the source packet sequence in EXTssnext.

[0699] ATClast_EXTss represents the maximum ATC value specified by the ATC field of the source packet sequence in EXTss.

[0700] Nsp represents the number of source packets which are included in the main TS and sub-TS and have ATC values corresponding to ATCs in the range of ATCDexss.

[0701] Rts1 represents a value of the TS recording rate in the main TS, and its maximum value is 48 Mbps.

[0702] Rts2 represents a value of the TS recording rate in the sub-TS, and its maximum value is 48 Mbps.

[0703] When two playitems are to be played back continuously, EXTss includes the first byte of data in the ATC sequence that is used by the previous playitem (Playitem 1).

- EXTss has a size equal to or more than the minimum Extent size defined in Condition 4.
- When EXTss is the first byte of data in the ATC sequence that is used by the previous playitem, the connection condition information of the previous playitem is not set to "5" or "6". In this case, it is not necessary to satisfy the size of EXTss.

[0704] EXTss includes byte of data in the ATC sequence that is used by the current playitem (Playitem 2).

- EXTss has a size equal to or more than the minimum Extent size defined in Condition 4.
- When EXTss is the last byte of data in the ATC sequence that is used by the Playitem 2, the connection condition information of Playitem 2 is not set to "5" or "6". In this case, it is not necessary to satisfy the size of EXTss.

<Detailed recording of base-view data blocks and dependent-view data blocks>

[0705] When GOPs of the main TS and sub-TS are to be recorded onto a recording medium, entries of the extension entry map point to only dependent-view picture data pieces that correspond to base-view picture data pieces pointed to by entries of the basic entry map as those that are to be played back at the same playback times as the dependent-view picture data pieces.

[0706] To realize such pointing, the recording process is performed as follows.

[0707] In the recording process, an attempt is made so that a boundary between a dependent-view data block and a base-view data block matches a boundary between a dependent-view GOP and a base-view GOP. More specifically, in this attempt, the access unit delimiter of the starting video access unit of GOP(i) in the sub-TS is divided as a boundary

between dependent-view data blocks, and the access unit delimiter of the starting video access unit of GOP(i) in the main TS is divided as a boundary between base-view data blocks. In this division, the restriction on the Extent length described earlier should be satisfied.

[0708] In this division, when either a base-view data block or a dependent-view data block does not satisfy the restriction that the Extent should have a length that does not cause an underflow in a double buffer in the playback device, a padding packet is inserted either into immediately before the access unit delimiter of the starting video access unit of GOP(i) in the sub-TS, or into immediately before the access unit delimiter of the starting video access unit of GOP(i) in the main TS, and then the above-described attempt is made again so that the boundaries match.

[0709] When the boundaries match successfully by the above-described method, an entry pointing to a source packet number of a source packet storing the access unit delimiter of the starting access unit of the dependent-view GOP is added into the extension entry map. Also, an entry pointing to a source packet number of a source packet storing the access unit delimiter of the starting access unit of the base-view GOP is added into the base entry map, as well.

[0710] When the boundaries do not match even if the padding packet is inserted, and the source packet storing the access unit delimiter of the starting access unit of the dependent-view GOP is in the middle of the dependent-view data block, an entry pointing to the source packet is not added into the extension entry map. Similarly, when the source packet storing the access unit delimiter of the starting access unit of the base-view GOP is in the middle of the base-view data block, an entry pointing to the source packet is not added into the extension entry map.

[0711] When such entries are excluded from the extension entry map in this way, it is ensured that pairs of a base view and a dependent view are pointed to by the entries of the basic entry map and the extension entry map.

[0712] The process of recording base-view data blocks and dependent-view data blocks and then generating the entry maps is realized by a process in which the starts of GOPs are detected from the recorded stereoscopic interleaved stream file, and entries pointing to the detected starts of GOPs are added into the entry maps. The following describes the procedure for generating the basic and extension entry maps by detecting the starts of GOPs and adding the entries, with reference to FIG. 67.

[0713] FIG. 67 is a flow chart showing the procedure for generating the basic entry map and the extension entry map.

[0714] In step S601, forms of the basic entry map and the extension entry map are generated in the memory, and the control proceeds to a loop composed of steps S602 through S610. In this loop, the variable x identifies a GOP. The loop is executed as follows. The variable x is

initialized to 1 (step S602). The start of GOP(x) is identified (step S603). An SPN(x) corresponding to the starting PTS(x) of the GOP is identified (step S604). After this, judgments are performed in steps S605 and S607. In step S605, it is judged whether or not SPN(x) is the start of EXT1[i]. When it is judged that SPN(x) is not the start of EXT1[i], steps S606-609 are skipped. When it is judged that SPN(x) is the start of EXT1[i], the control proceeds to step S606 in which EXT2[j], whose start SPN(y) corresponds to PTS(x), is identified.

[0715] In step S607, it is judged whether or not variable "i" that identifies EXT1[i] matches variable "j" that identifies EXT2[j]. When it is judged that variable "i" does not match variable "j", the steps after this are skipped. When it is judged that variable "i" matches variable "j", EP_entry(x) pointing to a pair of PTS(x) and SPN(x) is added into the basic entry map (step S608), and EP_entry(x) pointing to a pair of PTS(x) and SPN(y) is added into the extension entry map (step S609).

[0716] In step S610, it is judged whether or not variable x specifies the last GOP. When it is judged that variable x does not specify the last GOP, variable x is incremented, and the control moves to step S603.

<Creation of index table>

[0717] The index table described in Embodiment 3 can be created in the following manner. When the base-view video stream, dependent-view video stream, clip information file, and playlist information file are generated in accordance with the flow chart shown in FIG 59, the display frequencies of playlists to be recorded on the recording medium are identified. Of these display frequencies, the resolution/display frequency of the playlist to be used in the first play title, or the resolution/display frequency of the playlist of the title specified by the title number in the range from 0 to 999 is set in the video format information and the frame rate information in the BDMV application information in the index table. With this structure, the resolution/display frequency to be applied to the display of the playlist is set in the index table.

[0718] FIG. 68 is a flow chart showing the procedure for generating the BD-J application, BD-J object, movie object, and index table. In step S701, a source program, which instructs the playback device to generate a player instance for a playlist, is generated by the object-oriented programming. In step S702, a BD-J application is generated by compiling and archiving the generated source program.

[0719] In step S703, a BD-J object is generated. In step S704, a movie object is described with use of a command that instructs playback of a playlist. In step S705, an index table is generated by describing correspondence between title numbers and BD-J object or movie object. In step S706, a playlist to be the first play title is selected. In step S707, BDMV application information, which indicates the video format and video rate of the playlist in the first play title, is generated. In step S708, an index table that includes the title index and the BDMV application information is generated. In step S709, BD-J object, the BD-J application,

movie object, and index table are written onto the recording medium.

[0720] The following explains the recording medium that is generated by the above-described recording.

[0721] FIG 69 shows an internal structure of a multi-layered optical disc.

[0722] The first row of FIG. 69 shows one example of a multi-layered optical disc. The second row shows tracks in the horizontally extended format though they are in reality formed spirally in the recording layers. These spiral tracks in the recording layers are treated as one continuous volume region. The volume region is composed of a lead-in region, recording layers of recording layers 1 through 3, and a lead-out region, where the lead-in region is located at the inner circumference, the lead-out region is located at the outer circumference, and the recording layers of recording layers 1 through 3 are located between the lead-in region and the lead-out region. The recording layers of recording layers 1 through 3 constitute one consecutive logical address space.

[0723] The volume region is sectioned into units in which the optical disc can be accessed, and serial numbers are assigned to the access units. The serial numbers are called logical addresses. Data is read from the optical disc by specifying a logical address. Here, in the case of a read-only disc such as the BD-ROM, basically, sectors with consecutive logical addresses are also consecutive in the physical disposition on the optical disc. That is to say, data stored in the sectors with consecutive logical addresses can be read without performing a seek operation. However, at the boundaries between recording layers, consecutive data reading is not possible even if the logical addresses are consecutive. It is thus presumed that the logical addresses of the boundaries between recording layers are registered in the recording device in advance.

[0724] In the volume region, file system management information is recorded immediately after the lead-in region. Following this, a partition region managed by the file system management information exists. The file system is a system that expresses data on the disc in units called directories or files. In the case of the BD-ROM, the file system is a UDF (Universal Disc Format). Even in the case of an everyday PC (personal computer), when data is recorded with a file system called FAT or NTFS, the data recorded on the hard disk under directories and files can be used on the computer, thus improving usability. The file system makes it possible to read logical data in the same manner as in an ordinary PC, using a directory and file structure.

[0725] The fourth row shows how the regions in the file system region managed by the file system are assigned. As shown in the fourth row, a non-AV data recording region exists on the innermost circumference side in the file system region; and an AV data recording region exists immediately following the non-AV data recording region. The fifth row shows the contents recorded in the non-AV data recording region and the AV data recording region. As shown in the fifth row, Extents constituting the AV files are recorded in the AV data recording region; and

Extents constituting non-AV files, which are files other than the AV files, are recorded in the non-AV data recording region.

[0726] FIG. 70 shows the application format of the optical disc based on the file system.

[0727] The BDMV directory is a directory in which data such as AV content and management information used in the BD-ROM are recorded. Five sub-directories called "PLAYLIST directory," "CLIPINF directory", "STREAM directory", "BDJO directory", "JAR directory", and "META directory" exist below the BDMV directory. Also, two types of files (i.e. index.bdmv and MovieObject.bdmv) are arranged under the BDMV directory.

[0728] A file "index.bdmv" (the file name "index.bdmv" is fixed) stores an index table.

[0729] A file "MovieObject.bdmv" (the file name "MovieObject. bdmv" is fixed) stores one or more movie objects. The movie object is a program file that defines a control procedure to be performed by the playback device in the operation mode (HDMV mode) in which the control subject is a command interpreter. The movie object includes one or more commands and a mask flag, where the mask flag defines whether or not to mask a menu call or a title call when the call is performed by the user onto the GUI.

[0730] A program file (XXXXX.bdjo---"XXXXX" is variable, and the extension "bdjo" is fixed) to which an extension "bdjo" is given exists in the BDJO directory. The program file stores a BD-J object that defines a control procedure to be performed by the playback device in the BD-J mode.

[0731] A substance of such a Java™ application is a Java™ archive file (YYYYY.jar) stored in the JAR directory under the BDMV directory.

[0732] An application may be, for example, a Java™ application that is composed of one or more xlet programs having been loaded into a heap memory (also called work memory) of a virtual machine. The application is constituted from the xlet programs having been loaded into the work memory, and data.

[0733] In the "PLAYLIST directory", a playlist information file ("xxxxx.mpls"---"XXXXX" is variable, and the extension "mpls" is fixed) to which an extension "mpls" is given exists.

[0734] In the "CLIPINF directory", a clip information file ("xxxxx.clpi"---"XXXXX" is variable, and the extension "clpi" is fixed) to which an extension "clpi" is given exists.

[0735] The Extents constituting the files existing in the directories explained up to now are recorded in the non-AV data region.

[0736] The "STREAM directory" is a directory storing a transport stream file. In the "STREAM directory", a transport stream file ("xxxxx.m2ts"---"XXXXX" is variable, and the extension "m2ts"

is fixed) to which an extension "m2ts" is given exists.

[0737] The above-described files are formed on a plurality of sectors that are physically continuous in the partition region. The partition region is a region accessed by the file system and includes an "region in which file set descriptor is recorded", "region in which end descriptor is recorded", "ROOT directory region", "BDMV directory region", "JAR directory region", "BDJO directory region", "PLAYLIST directory region", "CLIPINF directory region", and "STREAM directory region". The following explains these regions.

[0738] The "file set descriptor" includes a logical block number (LBN) that indicates a sector in which the file entry of the ROOT directory is recorded, among directory regions. The "end descriptor" indicates an end of the file set descriptor.

[0739] Next is a detailed description of the directory regions. The above-described directory regions have an internal structure in common. That is to say, each of the "directory regions" is composed of a "file entry", "directory file", and "file recording region of lower file".

[0740] The "file entry" includes a "descriptor tag", an "ICB tag", and an "allocation descriptor".

[0741] The "descriptor tag" is a tag identifying, as a "file entry", the file entry which includes the descriptor tag itself.

[0742] The "ICB tag" indicates attribute information concerning the file entry itself.

[0743] The "allocation descriptor" includes a logical block number (LBN) that indicates a recording position of the directory file. Up to now, the file entry has been described. Next is a detailed description of the directory file.

[0744] The "directory file" includes a "file identification descriptor of lower directory" and "file identification descriptor of lower file".

[0745] The "file identification descriptor of lower directory" is information that is referenced to access a lower directory that belongs to the directory file itself, and is composed of identification information of the lower directory, the length of the directory name of the lower directory, a file entry address that indicates the logical block number of the block in which the file entry of the lower directory is recorded, and the directory name of the lower directory.

[0746] The "file identification descriptor of lower file" is information that is referenced to access a file that belongs to the directory file itself, and is composed of identification information of the lower file, the length of the lower file name, a file entry address that indicates the logical block number of the block in which the file entry of the lower file is recorded, and the file name of the lower file.

[0747] The file identification descriptors of the directory files of the directories indicate the

logical blocks in which the file entries of the lower directory and the lower file are recorded. By tracing the file identification descriptors, it is therefore possible to reach from the file entry of the ROOT directory to the file entry of the BDMV directory, and reach from the file entry of the BDMV directory to the file entry of the PLAYLIST directory. Similarly, it is possible to reach the file entries of the JAR directory, BDJO directory, CLIPINF directory, and STREAM directory.

[0748] The "file recording region of lower file" is a region in which the substance of the lower file that belongs to a directory. A "file entry" of the lower entry and one or more "Extents" are recorded in the "file recording region of lower file".

[0749] The stream file that constitutes the main feature of the present application is a file recording region that exists in the directory region of the directory to which the file belongs. It is possible to access the transport stream file by tracing the file identification descriptors of the directory files, and the allocation descriptors of the file entries.

(Embodiment 7)

[0750] The present embodiment describes the internal structure of a 2D/3D playback device that has integrated functions of the playback devices having been described in the embodiments so far.

[0751] FIG. 71 shows the structure of a 2D/3D playback device. The 2D/3D playback device includes a BD-ROM drive 1, a read buffer 2a, a read buffer 2b, a switch 3, a system target decoder 4, a plane memory set 5a, a plane overlay unit 5b, an HDMI transmission/reception unit 6, a playback control unit 7, a memory, a register set 203, a program executing unit 11, a program memory 12, an HDMV module 13, a BD-J platform 14, a middleware 15, a mode management module 16, a user event processing unit 17, a local storage 18, and a nonvolatile memory 19.

[0752] The BD-ROM drive 1, like a 2D playback device, reads out data from a BD-ROM disc based on a request from the playback control unit 7. AV clips read from the BD-ROM disc are transferred to the read buffer 2a or 2b.

[0753] When a 3D image is to be played back, the playback control unit 7 issues a read request that instructs to read the base-view data block and the dependent-view data block alternately in units of Extents. The BD-ROM drive 1 reads out Extents constituting the base-view data block into the read buffer 2a, and reads out Extents constituting the dependent-view data block into the read buffer 2b. When a 3D image is to be played back, the BD-ROM drive 1 should have a higher reading speed than the BD-ROM drive for a 2D playback device, since it is necessary to read both the base-view data block and the dependent-view data block simultaneously.

[0754] The read buffer 2a is a buffer that may be realized by, for example, a dual-port

memory, and stores the data of the base-view data blocks read by the BD-ROM drive 1.

[0755] The read buffer 2b is a buffer that may be realized by, for example, a dual-port memory, and stores the data of the dependent-view data blocks read by the BD-ROM drive 1.

[0756] The switch 3 is used to switch the source of data to be input into the read buffers, between the BD-ROM drive 1 and the local storage 18.

[0757] The system target decoder 4 decodes the streams by performing the demultiplexing process onto the source packets read into the read buffer 2a and the read buffer 2b.

[0758] The plane memory set 5a is composed of a plurality of plane memories. The plane memories include those for storing a left-view video plane, a right-view video plane, a secondary video plane, an interactive graphics plane (IG plane), and a presentation graphics plane (PG plane).

[0759] The plane overlay unit 5b performs the plane overlaying explained the embodiments so far. When the image is to be output to the television or the like, the output is conformed to the 3D system. When it is necessary to play back the left-view image and the right-view image alternately by using the shutter glasses, the image is output as it is. When the image is to be output to, for example, the lenticular television, a temporary buffer is prepared, the left-view image is first transferred into the temporary buffer, and the left-view image and the right-view image are output simultaneously after the right-view image is transferred.

[0760] The HDMI transmission/reception unit 6 executes the negotiation phase described in Embodiment 1 in conformance with, for example, the HDMI standard, where HDMI stands for High Definition Multimedia Interface. In the negotiation phase, the HDMI transmission/reception unit 6 can receive, from the television, (i) information indicating whether or not it supports a stereoscopic display, (ii) information regarding resolution for a monoscopic display, and (iii) information regarding resolution for a stereoscopic display.

[0761] The playback control unit 7 includes a playback engine 7a and a playback control engine 7b. When it is instructed from the program executing unit 11 or the like to play back a 3D playlist, the playback control unit 7 identifies a base-view data block of a playitem that is the playback target among the 3D playlist, and identifies a dependent-view data block of a sub-playitem in the 3D sub-path that should be played back in synchronization with the playitem. After this, the playback control unit 7 interprets the entry map of the corresponding clip information file, and requests the BD-ROM drive 1 to alternately read the Extent of the base-view data block and the Extent of the dependent-view data block, starting with the playback start point, based on the Extent start type that indicates which of an Extent constituting the base-view video stream and an Extent constituting the dependent-view video stream is disposed first. When the playback is started, the first Extent is read into the read buffer 2a or the read buffer 2b completely, and then the transfer from the read buffer 2a and the read buffer 2b to the system target decoder 4 is started.

[0762] The playback engine 7a executes AV playback functions. The AV playback functions in the playback device are a group of traditional functions succeeded from CD and DVD players. The AV playback functions include: Play, Stop, Pause On, Pause Off, Still Off, Forward Play (with specification of the playback speed by an immediate value), Backward Play (with specification of the playback speed by an immediate value), Audio Change, Picture Data Change for Secondary Video, and Angle Change.

[0763] The playback control engine 7b performs playlist playback functions in response to function calls from the command interpreter which is the main operating body in the HDMV mode, and from the Java platform which is the main operating body in the BD-J mode. The playlist playback functions mean that, among the above-described AV playback functions, the Play and Stop functions are performed in accordance with the current playlist information and the current clip information, where the current playlist information constitutes the current playlist.

[0764] The memory is a memory for storing the current playlist information and the current clip information. The current playlist information is a piece of playlist information that is currently a target of processing, among a plurality of pieces of playlist information that can be accessed from the BD-ROM, built-in medium drive, or removable medium drive. The current clip information is a piece of clip information that is currently a target of processing, among a plurality of pieces of clip information that can be accessed from the BD-ROM, built-in medium drive, or removable medium drive.

[0765] The register set 10 is a player status/setting register set that is a set of registers including a general-purpose register for storing arbitrary information that is to be used by contents, as well as the playback status register and the playback setting register having been described in the embodiments so far.

[0766] The program executing unit 11 is a processor for executing a program stored in a BD program file. Operating according to the stored program, the program executing unit 11 performs the following controls: (1) instructing the playback control unit 7 to play back a playlist; and (2) transferring, to the system target decoder, PNG/JPEG that represents a menu or graphics for a game so that it is displayed on the screen. These controls can be performed freely in accordance with construction of the program, and how the controls are performed is determined by the process of programming the BD-J application in the authoring process.

[0767] The program memory 12 stores a current dynamic scenario which is provided to the command interpreter that is an operator in the HDMV mode, and to the Java™ platform that is an operator in the BD-J mode. The current dynamic scenario is a current execution target that is one of Index.bdmv, BD-J object, and movie object recorded in the BD-ROM. The program memory 12 includes a heap memory.

[0768] The heap memory is a stack region for storing byte codes of the system application,

byte codes of the BD-J application, system parameters used by the system application, and application parameters used by the BD-J application.

[0769] The HDMV module 13 is provided with a command interpreter, and controls the HDMV mode by decoding and executing the navigation command which constitutes the movie object.

[0770] The BD-J platform 14 is a Java™ platform that is an operator in the BD-J mode, and is fully implemented with Java™ 2Micro_Edition (J2ME) Personal Basis Profile (PBP 1.0), and Globally Executable MHP specification (GEM 1.0.2) for package media targets. The BD-J platform 14 is composed of a class loader, a byte code interpreter, and an application manager.

[0771] The class loader is one of system applications, and loads a BD-J application by reading byte codes from the class file existing in the JAR archive file, and storing the byte codes into the heap memory.

[0772] The byte code interpreter is what is called a Java™ virtual machine. The byte code interpreter converts (i) the byte codes constituting the BD-J application stored in the heap memory and (ii) the byte codes constituting the system application, into native codes, and causes the MPU to execute the native codes.

[0773] The application manager is one of system applications, and performs application signaling for the BD-J application based on the application management table in the BD-J object, such as starting or ending a BD-J application. This completes the internal structure of the BD-J platform.

[0774] The middleware 15 is an operating system for the embedded software, and is composed of a kernel and a device driver. The kernel provides the BD-J application with a function unique to the playback device, in response to a call for the Application Programming Interface (API) from the BD-J application. The middleware 15 also realizes controlling the hardware, such as starting the interruption handler by sending an interruption signal.

[0775] The mode management module 16 holds Index.bdmv that was read from the BD-ROM, built-in medium drive, or removable medium drive, and performs a mode management and a branch control. The management by the mode management is a module assignment to cause either the BD-J platform or the HDMV module to execute the dynamic scenario.

[0776] The user event processing unit 17 receives a user operation via a remote control, and causes the program executing unit 11 or the playback control unit 7 to perform a process as instructed by the received user operation. For example, when the user presses a button on the remote control, the user event processing unit 17 instructs the program executing unit 11 to execute a command included in the button. For example, when the user presses a fast forward/rewind button on the remote control, the user event processing unit 17 instructs the playback control unit 7 to execute the fast forward/rewind process onto the AV clip of the

currently played-back playlist.

[0777] The local storage 18 includes the built-in medium drive for accessing a hard disc, and the removable medium drive for accessing a semiconductor memory card, and stores downloaded additional contents, data to be used by applications, and other data. A region for storing the additional contents is divided into as many small regions as BD-ROMs. Also, a region for storing data used by applications is divided into as many small regions as the applications.

[0778] The nonvolatile memory 19 is a recording medium that is, for example, a readable/writable memory, and is a medium such as a flash memory or FeRAM that can preserve the recorded data even if a power is not supplied thereto. The nonvolatile memory 19 is used to store a backup of the register set 203.

(Embodiment 8)

[0779] The present embodiment is an embodiment for implementing an invention that is the same as the invention (hereinafter referred to as "present invention") recited in the description and the drawings attached to a request for a patent application which is a basis of the priority declaration of the present application.

[0780] Firstly, of the implementation acts of the recording medium of the present invention, an embodiment of a usage act is described. FIG. 72A shows the embodiment of a usage act of a recording medium relating to the present invention. A BD-ROM 101 in FIG. 72A is a recording medium pertaining to the present invention. The BD-ROM 101 is used to supply movies to a home theater system composed of a playback device 102, a television 103, and a remote control 104.

[0781] This completes the description of the usage act of the recording medium relating to the present invention.

[0782] The following describes the data structure of a BD-ROM (i.e., a recording medium of the present invention) for recording 2D images.

[0783] FIG. 72B shows the structure of the BD-ROM.

[0784] The fourth row in FIG 72B shows the BD-ROM 101, and the third row shows a track on the BD-ROM. Although the track is usually formed to extend spirally from an inner circumference to an outer circumference, the track is drawn in a laterally expanded manner in the present figure. As with other optical discs such as DVDs and CDs, the BD-ROM 101 has a recording region that spirals from the inner circumference to the outer circumference of the BD-ROM 101. The BD-ROM 101 also has a volume region in which logical data can be recorded, between the lead-in on the inner circumference side and the lead-out on the outer

circumference side. The volume region is sectioned into units in which the optical disc can be accessed, and serial numbers are assigned to the access units. The serial numbers are called logical addresses. Data is read out from the optical disc by specifying logical addresses. It is defined here that the logical addresses also indicate physically consecutive regions on the optical disc. That is to say, data with consecutive logical addresses can be read without a seek operation. There is a special area called BCA (Burst Cutting Area) provided at a place more inner than the lead-in. Since it can be read only by a drive, not by an application, the BCA is often used by the copyright protection technology.

[0785] At the head of the volume region, volume information of a file system is recorded, followed by application data such as video data. The file system is a system that expresses data on the disc in units of directories and files. In the BD-ROM 101, the file system is recorded in a format called UDF (Universal Disc Format). Even in the case of an everyday PC (Personal Computer), when data is recorded with a file system called FAT or NTFS, the data recorded on the hard disk under directories and files can be used on the computer, thus improving usability. The file system makes it possible to read logical data in the same manner as in an ordinary PC, using a directory and file structure.

[0786] The directory and file structure on the BD-ROM 101 is as follows. A BDMV directory is provided directly below a root directory (ROOT). Data such as AV contents and management information on the BD-ROM 101 is recorded in the BDMV directory. Provided below the BDMV directory are an index file (index.bdmv) defining an index table constituting a title, a PLAYLIST directory, a CLIPINF directory, a STREAM directory, a BDJO directory, and a JAR directory. Provided below the STREAM directory, CLIPINF directory and PLAYLIST directory are: an AV clip (XXX.M2TS) storing AV contents such as video and audio that are multiplexed together; a clip information file (XXX.CLPI) storing AV clip management information; a playlist file (YYY.MPLS) defining logical playback paths of AV clips; and a BD program file (AAA.PROG) storing a program that defines a dynamic scenario.

[0787] The following describes the data structure of the files that are stored under the BDMV directory.

[0788] The index file (Index.bdmv) is described first. The index file has the index table shown in FIG. 72C. The index table is a table that is provided in the highest layer and defines the title structure of the top menu, FirstPlay, and all titles stored on the BD-ROM. The index table specifies program files to be executed first from each title, the top menu, and the FirstPlay. Each time a title or a menu is called, a BD-ROM player refers to the index table, to execute a predetermined BD program file. Here, FirstPlay is set by a content provider, and indicates a BD program file to be executed automatically when the disc is loaded into a BD-ROM player. The top menu specifies a movie object or a BD-J object which is to be called when a command "Return to the menu" is executed according to a user operation via a remote controller.

[0789] The BD program file (AAA.PRG) stores a plurality of programs to be specified and executed from each title. Different prefixes (e.g., AAA) are used to identify corresponding files.

Although interpreter-approach programs with unique specifications are used to generate programs for Blu-ray Disc, the programs to be used may be written in a general-purpose programming language such as Java™ or Java™ Script. The programming language is not essential to the present invention. The programs specify playlists to be played back.

[0790] A description is now given on the AV clip (XXX.M2TS) and the clip information file (XXX.CLPI).

[0791] The AV clip is a digital stream having an MPEG-2 transport stream format.

[0792] FIG. 73A shows the structure of an AV clip. As shown in FIG. 73A, an AV clip is obtained by multiplexing one or more of the video stream, audio stream, presentation graphics stream (PG), and interactive graphics stream (IG). The video stream represents the primary and secondary videos of a movie. The audio stream represents the primary audio of the movie and the secondary audio to be mixed with the primary audio. The presentation graphics stream represents subtitles for the movie. Note that the primary video is an ordinary video displayed on the screen, and the secondary video is displayed in a small screen provided within display of the primary video. The interactive graphics stream represents an interactive screen created by disposing GUI components on a screen. The video stream is encoded by an encoding method such as MPEG-2, MPEG-4 AVC, or SMPTE VC-1 before it is recorded. The audio stream is compress-encoded by a method such as Dolby AC-3, Dolby Digital Plus, MLP, DTS, DTS-HD, or linear PCM before it is recorded.

[0793] Described below is the structure of the video stream. When a video compression/encoding technique such as MPEG-2, MPEG-4 AVC and SMPTE VC-1 is used, data is compressed in size by taking advantage of spatial and temporal redundancy of the video. One method that takes advantage of temporal redundancy of the video is inter-picture predictive encoding. According to the inter-picture predictive encoding, when encoding a certain picture, another picture to be displayed before or after the certain picture along the display time axis is designated as a reference picture. After detecting a motion amount by which data of the certain picture differs from data of the reference picture, the data of the certain picture is compressed in size by removing the spatial redundancy, which is obtained by subtracting the certain picture (target of encoding) from the motion-compensated reference picture.

[0794] An I-picture is a picture that is encoded by inter-picture predictive encoding-i.e., by only using information present in itself without referring to a reference picture. It should be noted that a "picture" is a unit of encoding and denotes both of a frame and a field. A P-picture is a picture that is encoded by inter-picture predictive encoding-more specifically, by referring to another picture that has already been processed. A B-picture is a picture that is encoded by inter-picture predictive encoding-more specifically, by simultaneously referring to other two pictures that have already been processed. A B-picture that is referred to by another picture is called a "Br-picture". A frame (in the case of the frame structure) and a field (in the case of the field structure) are called video access units.

[0795] Each stream in the AV clip is identified by a PID. For example, an alignment 0x1011 is allocated to a video stream used as the video of the movie, alignments 0x1100 to 0x111F are allocated to the audio streams, alignments 0x1200 to 0x121F are allocated to the presentation graphics, alignments 0x1400 to 0x141F are allocated to the interactive graphics streams, alignments 0x1B00 to 0x1B1F are allocated to the video streams used as secondary video of the movie, and alignments 0x1A00 to 0x1A1F are allocated to the audio stream used as secondary audio mixed with the primary audio.

[0796] FIG 73B schematically shows how the AV clip is multiplexed. Firstly, a video stream 501 composed of a plurality of video frames and an audio stream 504 composed of a plurality of audio frames are converted into a PES packet series 502 and a PES packet series 505, respectively. The PES packet series 502 and 505 are converted into TS packets 503 and 506, respectively. Similarly, data pieces of a presentation graphics stream 507 and interactive graphics 510 are converted into a PES packet series 508 and a PES packet series 511, respectively, and the PES packet series 508 and 511 are converted into TS packets 509 and 512, respectively. An AV clip 513 is composed of the TS packets 503, 506, 509, and 512 multiplexed on one stream.

[0797] FIG. 74A illustrates in more detail how the video stream is stored in the PES packet series. The first row shows a video frame series of the video stream. The second row shows a PES packet series. As shown by arrows yy1, yy2, yy3 and yy4, the video stream is composed of a plurality of video presentation units (I-picture, B-picture, P-picture). The video stream is divided up into the individual pictures, and each picture is stored in the payload of a PES packet. Each PES packet has a PES header storing a PTS (Presentation Time-Stamp) that indicates a display time of the picture stored in the payload of the PES packet, and a DTS (Decoding Time-Stamp) that indicates a decoding time of the picture stored in the payload of the PES packet.

[0798] FIG. 74B shows the format of the TS packets ultimately written in the AV clip. Each TS packet is a fixed-length, 188-byte packet composed of a 4-byte TS header carrying information such as a PID identifying the stream, and a 184-byte TS payload storing data. The PES packets are stored in the divided form in the TS payloads. In the case of BD-ROM, each TS packet is attached a 4-byte TP_Extra_Header, thus constituting a 192-byte source packet. The source packets are written in the AV clip. The TP_Extra_Header stores information such as an ATS (Arrival_Time_Stamp). The ATS shows a transfer start time at which the TS packet is to be transferred to a PID filter of a system target decoder 1503, which will be described later. The source packets are arranged in the AV clip as shown on the lower row in FIG. 74B. The numbers incrementing from the head of the AV clip are called SPNs (Source Packet Numbers).

[0799] In addition to TS packets of audio, video, subtitles and the like, the AV clip also includes TS packets of a PAT (Program Association Table), a PMT (Program Map Table) and a PCR (Program Clock Reference). The PAT shows a PID of a PMT used in the AV clip. The PID of the PAT itself is registered as "0". The PMT stores the PIDs in the streams of video, audio, subtitles

and the like, and attribute information corresponding to the PIDs. The PMT also has various descriptors relating to the AV clip. The descriptors have information such as copy control information showing whether copying of the AV clip is permitted or not permitted. The PCR stores STC time information corresponding to an ATS showing when the PCR packet is transferred to a decoder, in order to achieve synchronization between an ATC (Arrival Time Clock) that is a time axis of ATSSs, and an STC (System Time Clock) that is a time axis of PTSs and DTSSs.

[0800] FIG. 75A explains the data structure of the PMT in detail. A PMT header is disposed at the top of the PMT. Information written in the PMT header includes the length of data included in the PMT to which the PMT header is attached. A plurality of descriptors relating to the AV clip is disposed after the PMT header. Information such as the described copy control information is listed in the descriptors. After the descriptors is a plurality of pieces of stream information relating to the streams included in the AV clip. Each piece of stream information is composed of stream descriptors, each listing information such as a stream type for identifying the compression codec of the stream, a stream PID, or stream attribute information (such as frame rate or aspect ratio). The number of stream descriptors is equal to that of streams in the AV clip.

[0801] As shown in FIG. 75B, each piece of clip information file is management information for an AV clip. The clip information files are in one to one correspondence with the AV clips, and are each composed of clip information, stream attribute information, and entry map.

[0802] As shown in FIG. 75B, clip information is composed of a system rate, a playback start time, and a playback end time. The system rate represents a maximum transfer rate at which the AV clip is transferred to the PID filter of the system target decoder, which will be described later. The interval between the ATSSs in the AV clip is equal to or lower than the system rate. The playback start time is the PTS of the first video frame in the AV clip. The playback end time is obtained by adding a per-frame playback interval to the PTS of the last video frame in the AV clip.

[0803] As shown in FIG. 76A, a piece of attribute information is registered for each PID of each stream in the AV clip. Each piece of attribute information has different information depending on whether the corresponding stream is a video stream, an audio stream, a presentation graphics stream, or an interactive graphics stream. Each piece of video stream attribute information carries information including what kind of compression codec the video stream was compressed with, and the resolution, aspect ratio and frame rate of the pieces of picture data that compose the video stream. Each piece of audio stream attribute information carries information including what kind of compression codec the audio stream was compressed with, how many channels are included in the audio stream, how many languages the audio stream supports, and the sampling frequency. The information in the video stream attribute information and the audio stream attribute information is used for purposes such as initialization of a decoder before the player performs playback.

[0804] As shown in FIG 76B, the entry map is table information that shows entry map header information 1101, PTSs, and SPNs. Each PTS shows a display time of each I-picture in the video stream in the AV clip. Each SPN is the SPN of the AV clip that is started with an I-picture. Here, a pair of a PTS and an SPN shown in a same row in the table is called an "entry point". Each entry point has an entry point ID (hereinafter also referred to as an "EP_ID"). Starting with the top entry point, which has an entry point ID 0, the entry points have successively incremented entry point IDs. Using the entry map, the player can specify the location of a file of an AV clip corresponding to an arbitrary point on the playback axis of the video stream. For instance, when performing special playback such as fast forward or rewind, the player can perform processing efficiently without analyzing the AV clip, by specifying, selecting and playing back an I-picture registered in the entry map. An entry map is created for each video stream multiplexed in the AV clip. The entry maps are managed according to PIDs. The entry map header information 1101 is stored at the head of each entry map. The entry map header information 1101 carries information such as the PID of the corresponding video stream and the number of entry points.

[0805] A description is now given of the playlist file (YYY.MPLS).

[0806] A playlist indicates the playback path of an AV clip. As shown in FIG 77A, a playlist is composed of one or more playitems 1201. Each playitem shows a playback segment with respect to an AV clip. The playitems 1201 are each identified by a respective playitem ID, and are written in the order in which they are to be played in the playlist. Furthermore, the playlist includes an entry mark 1202 showing a playback start point. The entry mark 1202 can be assigned in the playback segments defined in the playitem. As shown in FIG. 77A, entry marks 1202 are assigned to positions that are potential playback start positions in playitems, and used for cued playback. In the case of a Movie title, for instance, the entry marks 1202 may be assigned to the head of each chapter, thus making chapter playback possible. It should be noted that the playback path of a series of playitems is defined as a main path 1205 in the present example.

[0807] The content of the playitems is now described with reference to FIG. 77B. A playitem includes clip information 1301 of the clip to be played back, a playback start time 1302, a playback end time 1303, a connection condition 1310, and a stream selection table 1305. Since the playback start time and the playback end time are time information, the player refers to the entry map of the clip information file, acquires an SPN corresponding to the designated playback start time and playback end time, and designates a read start position, to perform playback processing.

[0808] The connection condition 1310 shows a previous playitem and a connection type. When the connection condition 1310 of a playitem is "1", it is not guaranteed that the AV clip indicated by this playitem is seamlessly connected with another AV clip indicated by a previous playitem that precedes this playitem. When the connection condition 1310 of a playitem is "5" or "6", it is guaranteed that the AV clip indicated by this playitem is seamlessly connected with another AV clip indicated by a previous playitem that precedes this playitem. When the

connection condition 1310 is "5", an STC of one playitem and an STC of another playitem does not need to be continuous with each other. That is to say, the video display start time of a start of an AV clip indicated by a post-connection playitem may not be continuous from the video display start time of an end of an AV clip indicated by a pre-connection playitem. However, in the case where the AV clip indicated by the pre-connection playitem and the AV clip indicated by the post-connection playitem are input to the PID filter of the system target decoder 1503 and sequentially played back, these AV clips should not crash the decoding ability of the system target decoder 1503. Also, there are several conditions that must be met. For example, the last frame of the audio in the AV clip indicated by the pre-connection playitem must overlap the first frame of the audio in the AV clip indicated by the post-connection playitem on the playback time axis. Also, in the case where the connection condition 1310 is "6", when the AV clips indicated by the pre-connection and post-connection playitems are combined together, they must be playable as a single AV clip. In other words, an STC and ATC of the AV clip indicated by the pre-connection playitem are continuous, and an STC and ATC of the AV clip indicated by the post-connection playitem are continuous.

[0809] The stream selection table 1305 is composed of a plurality of stream entries 1309. Each stream entry 1309 is composed of a stream selection number 1306, stream path information 1307 and stream identification information 1308. The stream selection numbers 1306 are numbers that increment in order from the first stream entry 1309 included in the stream selection table. The stream selection numbers 1306 are used for stream identification in the player. The stream path information 1307 is information showing which AV clip the stream shown by the stream identification information 1308 is multiplexed on. For example, if the stream path information 1307 shows "main path", this indicates the AV stream of the playitem. If the stream path information 1307 shows "sub-path ID = 1", this indicates an AV clip of a sub-playitem corresponding to a playback segment of the playitem. Specifics of the sub-path will be described in the next section. The stream identification information 1308 is information such as PIDs, and shows streams multiplexed on the AV clip being referred to. Furthermore, stream attribute information is also recorded in the stream entries 1309. Each stream attribute information is a piece of information showing a property of a stream, and for instance includes a language attribute in the case of audio, presentation graphics, or interactive graphics.

[0810] As shown in FIG. 77C, a playlist may have one or more sub-paths. The sub-paths are assigned IDs in the order they are registered in the playlist. These IDs are used as sub-path IDs for identifying the sub-paths. Sub-paths are a series of playback paths played back in synchronization with a main path. As with a playitem, a sub-playitem has the clip information 1301 of the clip to be played back, the playback start time 1302, and the playback end time 1303. The playback start time 1302 and the playback end time 1303 of the sub-playitem are expressed using the same time axis as the main path. For example, if a certain stream entry 1309 registered in the stream selection table 1305 of the playitem #2 shows sub-path ID = 0 and presentation graphics 1, the presentation graphics 1 multiplexed on the AV clip of the sub-playitem #2 played back in synchronization with the playback segment of the playitem #2, among the sub-paths of sub-path ID = 0, will be played back in the playitem #2 playback

segment. Furthermore, a sub-playitem includes a field called an SP connection condition, which has the same meaning as a connection condition of a playitem. An AV clip on a border between sub-playitems whose SP connection conditions are "5" or "6" needs to meet the conditions that the stated playitems whose connection conditions are "5" or "6" need to meet.

[0811] This concludes the description of the data structure of the BD-ROM (i.e., a recording medium relating to the present invention) for recording thereon 2D images.

[0812] A description is now given of a playback device (2D playback device) relating to the present invention, the playback device playing back a BD-ROM having 2D images recorded thereon.

[0813] FIG. 78A shows the structure of a 2D playback device 1500. The 2D playback device 1500 is composed of a BD-ROM drive 1501, a read buffer 1502, a system target decoder 1503, a program memory 1504, a management information memory 1505, a program execution unit 1506, a playback control unit 1507, a player variable 1508, a user event processing unit 1509, and a plane adder 1510.

[0814] The BD-ROM drive 1501 reads data from a BD-ROM disc based on a request from the playback control unit 1507. An AV clip read from the BD-ROM disc is transferred to the read buffer 1502. An index file, a playlist file, and a clip information file read from the BD-ROM disc are transferred to the management information memory 1505. A movie object file read from the BD-ROM disc is transferred to the program memory 1504.

[0815] The read buffer 1502 is a buffer constituted from a memory or the like that stores data read using a BD-ROM drive. The management information memory 1505 is a buffer constituted from a memory or the like that stores management information on the index file, playlist file and clip information file. The program memory 1504 is a buffer constituted from a memory or the like that stores the movie object file.

[0816] The system target decoder 1503 performs (i) demultiplexing processing on source packets read into the read buffer 1502 and (ii) processing to decode streams. Information necessary to decode streams included in an AV clip, such as codec types and stream attributes, is transferred from the playback control unit 1507. The system target decoder 1503 writes the decoded primary video stream, secondary video stream, interactive graphics stream, and presentation graphics stream in their plane memories, namely a primary video plane, a secondary video plane, an interactive graphics plane (IG plane), and a presentation graphics plane (PG plane), respectively. The system target decoder 1503 also mixes the decoded primary audio stream with the decoded secondary audio stream, and outputs the mixed streams to a speaker or the like. The system target decoder 1503 also performs processing to decode graphics data such as JPEG and PNG (transferred from the program execution unit 1506) for display of a menu or the like, and to write the decoded graphics data to an image plane. Details of the system target decoder 1503 are given later.

[0817] The user event processing unit 1509 requests processing by the program execution unit 1506 or the playback control unit 1507 in response to a user operation made through the remote control. For instance, when a button on the remote control is pressed, the user event processing unit 1509 makes a request to the program execution unit 1506 to execute a command included in the button. As another example, when a fast forward or rewind button in the remote control is pressed, the user event processing unit 1509 instructs the playback control unit 1507 to execute fast forward or rewind processing of the AV clip of the playlist currently being played back.

[0818] The playback control unit 1507 has the function of controlling playback of the AV clip by controlling the BD-ROM drive 1501 and the system target decoder 1503. The playback control unit 1507 also controls playback processing of an AV clip by interpreting playlist information based on a playback instruction from the program execution unit 1506 or notification by the user event processing unit 1509. Furthermore, the playback control unit 1507 also performs setting and referencing of the player variable 1508, and performs playback operations.

[0819] The player variable 1508 includes system parameters () indicating the status of the player, and general parameters (GPRM) for general use.

[0820] FIG. 78B is a list of the system parameters (PSR).

[0821]

PSR0 :Language Code

PSR1 :Primary audio stream number

PSR2 :Subtitle stream number

PSR3 :Angle number

PSR4 :Title number

PSR5 :Chapter number

PSR6 :Program number

PSR7 :Cell number

PSR8 :Selected key information

PSR9 :Navigation timer

PSR10 :Playback time information

PSR11 :Mixing mode for Karaoke

PSR12 :Country information for parental management

PSR13 :Parental level

PSR14 :Player configuration value (video)

PSR15 :Player configuration value (audio)

PSR16 :Language code for audio stream

PSR17 :Language code extension for audio stream

PSR18 :Language code for subtitle stream

PSR19 :Language code extension for subtitle stream

PSR20 :Player region code

PSR21 :User's preferential selection of 2D/3D output mode

PSR22 :Current 2D/3D output mode

PSR23 :3D video output capability of display

PSR24 :3D image playback capability

PSR25 :Reserved

PSR26 :Reserved

PSR27 :Reserved

PSR28 :Reserved

PSR29 :Reserved

PSR30 :Reserved

PSR31 :Reserved

[0822] The PSR10 is updated every time picture data belonging to an AV clip is displayed. In other words, if the playback device causes a new piece of picture data to be displayed, the PSR10 is updated to show the display time (PTS) of the new picture. The current playback point can be known by referring to the PSR10.

[0823] The language code for the audio stream of the PSR16 and the language code for the subtitle stream of the PSR18 are items that can be set in the OSD of the player or the like, and show default language codes of the player. For example, the BD program file may have the following function. Namely, if the language code for audio stream PSR16 is English, when a playlist is played back, a stream entry having the same language code is searched for in the

stream selection table of the playitem, and the corresponding audio stream is selected and played back.

[0824] Furthermore, the playback control unit 1507 checks the status of the system parameter while playback is performed. The PSR1, PSR2, PSR21 and PSR22 show the audio stream number, subtitle stream number, secondary video stream number and secondary audio stream number, respectively. These values correspond to the stream selection number 606. As one example, the audio stream number PSR1 may be changed by the program execution unit 1506. The playback control unit 1507 compares the stream section number 606 from among the stream selection table 605 of the playitem currently being played back, refers to the matching stream entry 609, and switches playback of the audio stream. In this way, switches can be made between which audio, subtitle or secondary video stream is played back or not.

[0825] The program execution unit 1506 is a processor for executing a program stored in the BD program file. The program execution unit 1506 performs operations in accordance with the stored program, and performs control as follows. (1) The program execution unit 1506 instructs the playback control unit 1507 to perform playlist playback. (2) The program execution unit 1506 transfers PNG/JPEG for graphics for a menu or a game to the system target decoder, for display on a screen. These operations can be performed flexibly in accordance with the makeup of the programs. What kind of control is performed is determined according to programming procedure of the BD program file in the authoring procedure.

[0826] The plane adder instantaneously superimposes data pieces written in the primary video plane, the secondary video plane, the interactive graphics plane, the presentation graphics plane and the image plane, and displays the resultant superimposed data on the screen of a television or the like.

[0827] A description of the system target decoder 1503 is now given with reference to FIG. 79.

[0828] The source depacketizer interprets a source packet transferred to the system target decoder 1503, extracts the TS packet, and sends the TS packet to the PID filter. In sending the TS packet, the source depacketizer adjusts the time of input into the decoder in accordance with the ATS of the source packet. More specifically, in accordance with the rate of storing an AV clip, the source depacketizer transfers the TS packet to the PID filter at the instant that the value of the ATC generated by the ATC counter and the value of the ATS of the source packet become identical.

[0829] The PID filters transfer TS packets output from the source depacketizers. More specifically, the PID filters transfer TS packets having a PID that matches a PID required for playback to the primary video decoder, the secondary video decoder, the IG decoder, the PG decoder, the audio decoder or the secondary audio decoder, depending on the PID of the TS packet. For instance, in the case of the BD-ROM, a TS packet having a PID 0x1011 is transferred to the primary video decoder, TS packets having PIDs 0x1B00 to 0x1B1F are

transferred to the secondary video decoder, TS packets having PIDs 0x1100 to 0x111F are transferred to the primary audio decoder, TS packets having PIDs 0x1A00 to 0x1A1F are transferred to the secondary audio decoder, TS packets having PIDs 0x1200 to 0x121F are transferred to the PG decoder, and TS packets having PIDs 0x1400 to 0x141F are transferred to the IG decoder.

[0830] The primary video decoder is composed of a TB (Transport Stream Buffer) 1701, an MB (Multiplexing Buffer) 1702, an EB (Elementary Stream Buffer) 1703, a compressed video decoder 1704, and a DPB (Decoded Picture Buffer) 1705.

[0831] The TB 1701 is a buffer that, when a TS packet including a video stream is output from the PID filter 1702, temporarily stores the TS packet as it is.

[0832] The MB 1702 is a buffer that, when a video stream is output from the TB 1701 to the EB 1703, temporarily stores PES packets. When data is transferred from the TB 1701 to the MB 1702, the TS header of each TS packet is removed.

[0833] The EB 1703 is a buffer that stores a picture in an encoded state (I-picture, B-picture and P-picture). When data is transferred from the MB 1702 to the EB 1703, the PES header is removed.

[0834] The compressed video decoder 1704 creates a frame/field image by decoding each video access unit in a video elementary stream at respective predetermined decode times (DTS). Possible compression encoding formats of the video stream multiplexed on the AV clip include MPEG2, MPEG4AVC, and VC1, and therefore the decoding scheme used by the compressed video decoder 1704 can be changed in accordance with stream attributes. The compressed video decoder 1704 transfers each of the decoded frame/field images to the DPB 1705, and writes each of the decoded frame/field images in the primary video plane at respective display times (PTS).

[0835] The DPB 1705 is a buffer that temporarily stores the decoded frame/field images. The compressed video decoder 1704 makes use of the DPB 1705 to, when decoding the video access units (e.g., a P-picture and a B-picture encoded by the inter-picture predictive encoding), refer to pictures that have already been decoded.

[0836] The secondary video decoder has the same structure as the primary video decoder. The secondary video decoder performs decoding of an input secondary video stream, and writes resultant pictures to the secondary video plane in accordance with respective display times (PTS).

[0837] The IG decoder extracts and decodes an interactive graphics stream from the TS packets input from source packetizers, and writes the resultant decompressed graphics data to the IG plane in accordance with respective display times (PTS).

[0838] The PG decoder extracts and decodes a presentation graphics stream from the TS packets input from the source packetizers, and writes the resultant decompressed graphics data to the PG plane in accordance with respective display times (PTS).

[0839] The primary audio decoder has a buffer. While accumulating data in the buffer, the primary audio decoder extracts information such as a TS header and a PES header, and performs audio stream decode processing to obtain decompressed LPCM-state audio data. The primary audio decoder outputs the obtained audio data to the audio mixer in accordance with the respective playback time (PTS). Possible compression encoding formats of the audio stream multiplexed on the AV clip include AC3 and DTS, and therefore the decoding scheme used to decode the compressed audio is changed in accordance with stream attributes.

[0840] The secondary audio decoder has the same structure as the primary audio decoder. The secondary audio decoder performs decoding of an input secondary audio stream, and outputs resultant decompressed LPCM-state audio data to the audio mixer in accordance with respective display times. Possible compression encoding formats of the audio stream multiplexed on the AV clip include Dolby Digital Plus and DTS-HD LBR, and therefore the decoding scheme used to decode the compressed audio is changed in accordance with stream attributes.

[0841] The audio mixer mixes (superimposes) the decompressed audio data output from the primary audio decoder and the decompressed audio data output from the secondary audio decoder with each other, and outputs the resultant audio to a speaker or the like.

[0842] The image processor decodes graphics data (PNG and JPEG) transferred from the program execution unit, and outputs the resultant decoded graphics data to the image plane in accordance with a display time designated by the program execution unit.

[0843] This concludes the description of the structure of the 2D playback device relating to the present invention.

(Principle of 3D Playback)

[0844] With reference to FIG. 80, the following describes the principle of enabling stereoscopic viewing on a home-use screen. There are two major methods to enable the stereoscopic viewing: a method that utilizes holography; and a method that utilizes parallax images.

[0845] The first method utilizing the holography is characterized in that it can create 3D images of an object in such a manner that a human viewer recognizes the three-dimensionality of the created 3D images in the same way as he/she recognizes the three-dimensionality of the actual object. However, although a technical theory has already been established in the field of holography, when it comes to playback of a video, it is extremely difficult to create

holograms of a video with the current holography technique, because doing so requires use of (i) a computer that can perform an enormous amount of operations to create holograms of the video in real time, and (ii) a display device whose resolution is high enough to be able to draw thousands of linear materials in a distance of 1 mm. For this reason, there are almost no practical examples of holography that are commercially used.

[0846] The second method utilizing the parallax images is characterized in that, after right-eye images and left-eye images are separately prepared, it enables stereoscopic viewing by making the right-eye images and the left-eye images only visible to the right eye and the left eye, respectively. FIG. 80 shows a user looking at a relatively small cube that is on a straight line connecting the center of the user's face and the center of the cube, as viewed from above. The top right view exemplarily shows the cube as seen by the left eye of the user. The bottom right view exemplarily shows the cube as seen by the right eye of the user.

[0847] The merit of the second method is that it can realize the stereoscopic viewing merely by preparing right-eye images and left-eye images separately. As there are several technical ways to make the right-eye and left-eye images only visible to the right eye and left-eye, respectively, the second method has already been practically implemented as different techniques.

[0848] One technique is called a "sequential segregation" method, with which the user views the left-eye and right-eye images, which are displayed alternately in the time axis direction on a screen, while wearing stereoscopic glasses (with liquid-crystal shutters). At this time, to the user's eyes, a left-eye image and a corresponding right-eye image look superimposed over each other due to the afterimage effect. Accordingly, the user's eyes recognize that the pair of the left-eye image and the corresponding right-eye image is a 3D image. To be more specific, while a left-eye image is being displayed on the screen, the stereoscopic glasses make the left-eye liquid-crystal shutter transparent and the right-eye liquid-crystal shutter dark. Conversely, while a right-eye image is being displayed on the screen, the stereoscopic glasses make the right-eye liquid-crystal shutter transparent and the left-eye liquid-crystal shutter dark. As stated earlier, this technique (alternate-frame sequencing) displays right-eye and left-eye images alternately in the time axis direction. Thus, unlike an ordinary 2D movie that is displayed at 24 frames-per-second, this technique needs to display a total of 48 left-eye and right-eye images per second. Therefore, the alternate-frame sequencing is suitable for use in a display device that can rewrite the screen at a relatively high speed. The alternate-frame sequencing can also be used in any display device that can rewrite the screen for a predetermined number of times per second.

[0849] As opposed to the aforementioned sequential segregation method that outputs the left-eye and right-eye pictures alternately in the time axis direction, there is another technique that simultaneously displays, on a single screen, a left-eye picture and a right-eye picture horizontally next to each other. Here, with the aid of a lenticular lens that is semicircular in shape and attached to the surface of the screen, pixels constituting the left-eye picture and pixels constituting the right-eye picture are only presented to the left eye and the right eye,

respectively. In the above manner, this technique can create the illusion of 3D images by presenting parallax pictures to the left eye and the right eye. Note, the lenticular lens may be replaced with another device (e.g., liquid crystal elements) that has the same function as the lenticular lens. Also, a vertical polarizing filter and a horizontal polarizing filter may be provided for left-eye pixels and right-eye pixels, respectively. Here, stereoscopic viewing can be realized by the viewer viewing the screen through polarizing glasses composed of a vertical polarizing filter (for the left eye) and a horizontal polarizing filter (for the right eye).

[0850] This stereoscopic viewing technique utilizing the parallax images has been commonly used for attractions of amusement parks and the like, and has already been established. Hence, this technique may be the closest form of technology that could be practically implemented for home use. It should be mentioned that many other methods/techniques have been suggested to realize such stereoscopic viewing utilizing the parallax images, such as a two-color separation method. Although the alternate-frame sequencing and the polarization glass technique are explained in the present embodiment as examples of methods/techniques to realize the stereoscopic viewing, the stereoscopic viewing may be realized using other methods/techniques other than the aforementioned two techniques, as long as it is realized using parallax images.

[0851] In the present embodiment, a description is given of a method for recording, on an information recording medium, parallax images used for stereoscopic viewing. Hereafter, an image for the left eye is referred to as a "left-eye image", an image for the right eye is referred to as a "right-eye image", and a pair of the left-eye image and the corresponding right-eye image is referred to as a "3D image". (Switching between 2D and 3D displays)

[0852] Described below is the data structure of the BD-ROM, which is a recording medium pertaining to the present invention, for storing 3D images.

[0853] Basic parts of the data structure are the same as those of the data structure for recording 2D video images. Therefore, the following description focuses on extended or different parts of such data structure. The following description will be given under the assumption that 3D images are recorded on a BD-ROM. Hereafter, a playback device that can only play back 2D images is referred to as a "2D playback device", and a playback device that can play back both of 2D images and 3D images is referred to as a "2D/3D playback device".

[0854] The following describes an index file (Index.bdmv) stored in a BD-ROM for playing back stereoscopic images. FIG 98 shows an example of an index file (Index.bdmv) stored in a BD-ROM for playing back stereoscopic images. In the example shown in FIG. 98, as a playlist, there are prepared a 2D PlayList 2601 showing a playback path of 2D images and a 3D PlayList 2602 showing a playback path of 3D images. A title is selected by a user, and the executed BD program file checks if the playback device is compliant with the 3D image playback according to a program stored therein. If the playback device is compliant with the 3D image playback, the executed BD program file checks if the user has selected the playback of the 3D images, and switches a PlayList to be played back accordingly.

[0855] Also, a "3D existence flag" and a "2D/3D preference flag" are prepared for the index file. The 3D existence flag is a flag that identifies whether or not a PlayList for playing back the 3D images exists in the title. Since the 2D/3D playback device does not have to prepare for the playback of the 3D images in a case where the flag shows "FALSE", the 2D/3D playback device can skip processing such as HDMI authentication, thereby performing processing at high speed. The 2D/3D preference flag is an identifier showing whether a content provider specifies the playback of the 2D images or 3D images when the TV and the playback device are capable of playing back both the 2D images and the 3D images. When the flag shows "3D", the playback device can promptly perform the HDMI authentication since switching to a 2D mode is not necessary. In general, a large delay occurs during the HDMI authentication between the playback device and the TV when the video stream attribute such as the frame rate is different. Therefore, when another switching is performed from the playback of the 2D images to the playback of the 3D images after switching to the 2D images, a large delay occurs. Therefore, it is possible to prevent a delay time of the HDMI authentication if the switching to the playback of the 2D images can be skipped with use of the 2D/3D preference flag.

[0856] Note that the "3D existence flag" and the "2D/3D preference flag" may be set for each title instead of the index file as a whole.

[0857] FIG. 99 shows a selection flow of the 2D PlayList and the 3D PlayList according to the program in the BD program file.

[0858] In S2701, a value in PSR24 is checked. When the value is "0", since the playback device is a 2D playback device, the 2D PlayList is played back. When the value is "1", the process advances to S2702.

[0859] In S2702, a menu screen is displayed to ask whether the user wishes for playback of 2D images or 3D images. In accordance with a result of the user's selection made with a remote control or the like, when the user wishes for the 2D image playback, the 2D PlayList is played back, and when the user wishes for the 3D image playback, the process advances to S2703.

[0860] In S2703, it is checked whether the display corresponds to the 3D image playback. For example, after the playback device is connected to the display using HDMI, the playback device makes an inquiry to the display as to whether the display corresponds to the 3D image playback. When the display does not correspond to the 3D image playback, the display device plays back the 2D PlayList. Alternatively, the playback device may display, on a menu screen or the like, a notification that informs the user that the television is not ready for the playback. When the display corresponds to the 3D image playback, the display device plays back the 3D PlayList.

[0861] Also, in the above is described that the parental level can be set in the PSR13 in the

2D playback device. With this setting, control can be performed such that only the user who is the appropriate age or over can play back the BD-ROM disc. In addition to this parental level, a 3D parental level in PSR30 is prepared for the 2D/3D playback device. In the 3D parental level is stored information on the age of the user who uses the 2D/3D playback device, as with the PSR13. The BD program file of the title of the BD-Rom disc judges whether or not the playback is permitted, with use of this PSR30 in addition to the PSR13. Since the PSR30 is the parental level regarding the playback of the 3D images, the parental level is controlled with use of the PSR13 in the 2D playback device. With these two kinds of parental levels, control can be performed, in view of physical effects on a small child in the process of growing, based on a demand that "small children cannot watch 3D image but 2D images", for example. For example, a playlist to be played back may be selected with reference to the PSR30 after it is checked that the display supports the playback of the 3D image (S2703: YES) in the flow chart shown in FIG. 99.

[0862] Note that although the age information is stored in the PSR30 as with the PSR13, in the PSR30 may be set whether the playback of the 3D images is prohibited or not.

[0863] Also, in the system parameter (in this example, the PSR31) is set the information showing "which of the 2D images and the 3D images the user prefers to be played back". In the PSR31 is set, by the user via the OSD of the 2D/3D playback device, which of the playback of the 2D images and the playback of the 3D images the user give a priority to. When the display supports the playback of the 3D images, and information on the PSR31 shows that the user gives the priority to the playback of the 3D images, it is not necessary to switch to the playback of the 2D images. Therefore, the HDMI authentication can be promptly performed, and the playback processing of the 3D images can be also promptly performed. Also, the BD program determines whether to playback 2D or 3D with reference to this PSR31, thereby enabling playback processing in accordance with a user's preference.

[0864] Note that a BD program may refer to the PSR31 to determine a default selection button of a menu to be displayed by a BD program. For example, suppose that a menu prompts the user to branch to "2D video playback" or "3D video playback". In this case, if the value of the PSR31 indicates "2D", the user puts his cursor on a button of "2D video playback". If the value of the PSR31 indicates "3D", the user puts his cursor on a button of "3D video playback".

[0865] The selection on "which of 2D playback and 3D playback user prefers" differs for each user who performs playback. In the case where the 2D/3D playback device includes a unit for identifying a person who is watching the 2D/3D playback device, the value of the PSR31 may be set depending on a user who is currently watching the 2D/3D playback device. For example, suppose that three family members (father, mother, and child) use a 2D/3D playback device. The 2D/3D playback device manages an account for each user. In this case, by updating the value of the PSR31 depending on a user who is currently logging in, it is possible to perform control in accordance with a preference of a person who is actually using the 2D/3D playback device.

[0866] The selection on "which of 2D playback and 3D playback user prefers" may be performed by setting levels, instead of selecting between 2D and 3D. For example, four levels of "always 2D", "rather 2D", "rather 3D", and "always 3D" may be set. With such a structure, it is possible to perform playback processing in a 2D/3D playback device much more in accordance with a user's preference. For example, suppose that a PSR25 is used, which is a system parameter showing the status of a playback device. In this case, if the value of the PSR31 indicates the level "always 2D", the value of the PSR25 is always set to the 2D mode. If the value of the PSR31 indicates the level "always 3D", the value of the PSR25 is always set to the 3D mode (L/R mode or DEPTH mode).

[0867] Next, the following describes the structure of a presentation graphics with reference to FIG. 81. A subtitle entry displayed as shown in FIG. 81 is composed of a plurality of subtitle data entries. Each of the subtitle data entries is composed of composition information, window information, palette information, and object information. The composition information is information for defining the screen structure of subtitle data. The composition information stores therein cropping information of an object, a display position of the cropped object, a window ID for identifying a window to be referred to, a palette ID for identifying a palette to be referred to, and an object ID for identifying the palette to be referred to. The window information stores therein a window region for defining a region in which the decoder will perform decoding together with a window ID. The object information stores therein a graphics image together with the object ID. The graphics image is image data composed of 256 index colors, and is compressed by a compression method such as the run-length compression method. The palette information stores therein table information (CLUT) on a color to be used for the object together with the palette ID. The table is storable therein 256 colors, and each color is referable using a corresponding color ID. The color ID has either value of 0-255. The color ID having the value 255 fixedly corresponds to a clear and colorless color.

[0868] FIG. 82 shows decoding processing of a presentation graphics. Firstly, in STEP 1, for each subtitle data entry, a compressed graphics image specified using a reference object ID of composition information is decoded. In STEP 2, only necessary data is cropped from the graphics image using cropping information included in the composition information. In STEP 3, in accordance with a display position included in the composition information, a display position of the cropped data on a graphic plane is determined. In STEP 4, object data corresponding to only a range of a window region included in the window information specified using a reference window ID included in the composition information is rendered on the graphics plane. In STEP 5, a color is given to the graphic plane for display using palette information specified by a reference palette ID included in the composition information. The display timing is in accordance with a PTS of a PES packet in which the composition information is stored.

[0869] The following describes improvement in subtitle display.

[0870] When the "1 plane + offset" method is applied to a PG plane, in order to create subtitle data, it is necessary to adjust offset metadata depending on the depth of a video image. This makes creation of subtitle data difficult.

[0871] In view of this problem, the following describes a method in which regions of black frames on the screen that are not used for a main feature video of a movie work, the regions of the black frames are collected on the upper side or the lower side on the screen, and subtitle data is displayed on the regions of the black frames.

[0872] Since the black frames inserted in the video stream are unnecessary regions, subtitle data may be displayed on the black frames. However, as shown in the right side on the upper level of FIG. 4A, the black frame provided on each of the upper side and the lower side has only 131 pixels. The black frame having this size is slightly small to insert subtitle data. In view of this, as shown in FIG. 4B and 4C, the main feature video is shifted upward or downward, and a black color is given to a region obtained after shifting the main feature video, and the black frames provided on the upper side and the lower side are collected in the upper side or the lower side. As a result, it is possible to prepare a black frame enough large to insert subtitle data.

[0873] The following describes the data structure for realizing this concept.

[0874] The basic parts of the data structure are the same as those for storing 3D videos described in the above embodiments, and accordingly additional parts or different parts from the above embodiments are mainly described here. Also, the following description of PG is applicable to IG or a sub video in the same way as PG by replacing the PG with the IG or the sub video.

[0875] FIG. 83 shows the structure of a playitem of 3D playlist. The stream additional information 1311 of a PG stream included in a stream selection table includes a "shift value in video shift upward (PG_v_shift_value_for_Up)" and a "shift value in video shift downward (PG_v_shift_value_for_Down)". The "shift value in video shift upward (PG_v_shift_value_for_Up)" represents a shift amount of the PG plane in the case where the main video plane is shifted upward (the black frames are collected in the lower side), and the "shift value in video shift downward (PG_v_shift_value_for_Down)" represents a shift amount of the PG plane in the case where the main video plane is shifted downward (the black frames are collected in the upper side). The 2D/3D playback device adjusts the shift amount of the PG plane based on the shift value. The method of plane overlaying is described later.

[0876] Next, the 2D/3D playback device relating to the preset embodiment is described. FIG. 84 shows the structure of plane overlaying performed by the 2D/3D playback device. Although the description is given here using a PG plane as a representative example, the description is applicable to any plane such as a sub video plane, an IG plane, and an image plane.

[0877] In addition to the compositional elements described in the above embodiments, the 2D/3D playback device shown in FIG 84 includes, a video plane cropping unit 9701 that performs cropping processing of 2D/left-eye video plane and right eye video plane, a PSR32 for writing a shift mode of a video, a PG plane cropping unit 9702 that performs cropping

processing of a PG plane, and a PSR33 in which a shift amount of a plane such as a PG plane.

[0878] The PSR32 shown in FIG. 25A is a system parameter of the 2D/3D playback device, and indicates a shift mode of a video (video_shift_mode). The video_shift_mode of the PSR32 includes three modes of "Keep", "Up", and "Down". A value 0 of the PSR32 indicates "Up", and a value of 2 of the PSR32 indicates "Down". The video plane cropping unit 9701 performs cropping processing of a video plane in accordance with a video shift mode written in the PSR32. The value of the PSR32 is set via an API of a BD program or a command.

[0879] In the case where the value of PSR32 indicates "Keep", the video plane cropping unit 9701 does not change the 2D/left-eye video plane and the right eye video plane, and proceeds to processing of superimposing with other plane, as shown in FIG 25B(1). In the case where the value of the PSR32 indicates "Up", the video plane cropping unit 9701 shifts upwards the 2D/left-eye video plane and the right eye video plane, respectively, crops a black frame from the upper region, and inserts the cropped black frame into the lower region, as shown in FIG 25B(2). Then, the video plane cropping unit 9701 proceeds to processing of superimposing with the plane. As a result, the black frame can be concentrated downward of the plane. Also, in the case where the value of the PSR32 indicates "Down", the video plane cropping unit 9701 shifts downward the 2D/left-eye video plane and the right eye video plane, respectively, and crops a black frame from the lower region, and inserts the cropped black frame into the upper region, as shown in FIG. 25B(3). Then, the video plane cropping unit 9701 proceeds to processing of superimposing with the plane. As a result, the black frame can be concentrated upward of the plane.

[0880] FIG. 37 shows a system parameter (PSR33 is used here) showing a shift amount of each plane in the longitudinal axis direction. The shift amount shown by the PSR33 includes a plane shift amount of video shift upward and a plane shift amount of video shift downward. For example, the SPRM(33) of a PG plane includes "PG_shift_value_for_UP" and "PG_shift_value_for_Down". The value of the PSR33 is updated with a value ("PG_v_shift_value_for_Up" or "PG_v_shift_value_for_Down") set in the playlist due to switching between streams. Also, the SPRM(33) may be set via API of command of a BD program.

[0881] The PG plane cropping unit 9702 shown in FIG. 84 performs plane shift depending on a shift amount of the PG plane shown by the PSR33. The shift processing and overlay processing of overlaying with a video plane performed by the PG plane cropping unit 9702 are shown in FIG. 11 and FIG. 38. As shown in FIG 11, the video_shift_mode of the PSR32 indicates "Keep", the PG plane cropping unit 9702 performs overlay processing of overlaying with the video plane without performing the shift processing. As shown in FIG 38A, if the video_shift_mode of the PSR32 indicates "Up", the PG plane cropping unit 9702 performs shift processing of the PG plane using the value of the PG_shift_value_for_Up stored in the PSR33 to crop a part protruding from the plane and superimpose the cropped protruding part with the video plane. By performing such processing, it is possible to display a subtitle on a lower side compared with a case of 2D playback, and display the subtitle in an appropriate position within

a region of a black frame in the lower side. As shown in FIG. 38B, the video_shift_mode of the PSR32 indicates "Down", the PG plane cropping unit 9702 performs shift processing of the PG plane using the value of the PG_shift_value_for_Down stored in the PSR33 to crop a part protruding from the plane and superimpose the cropped protruding part with the video plane. By performing such processing, it is possible to display a subtitle on an upper side compared with a case of 2D playback, and display the subtitle in an appropriate position within a region of a black frame in the upper side.

[0882] Note that in the structure shown in FIG 84, offset processing in the horizontal axis direction (1 plane + offset method) for preventing jumping is omitted. Alternatively, it may be employed to add the mechanism of cropping processing based on an offset value in the lateral axis direction. With such a structure, even in the case where a subtitle is displayed on a region of a black frame, it is possible to make the subtitle to look like jumping.

[0883] Note that in the case where a subtitle is displayed on a region of a black frame as shown in FIG. 84, an offset value in the lateral axis direction may be a fixed value. In such a case, it may be possible to define a shift amount in the X-axis direction to additional information shown in FIG. 83, store a value of the shift amount in a PSR in the similar way as in the PSR33, and perform offset processing in the lateral axis direction using the value. This enables easy data creation.

[0884] In the structure of the plane overlaying described with reference to FIG. 84, the shift amount in the Y-axis direction is stored in the PSR33. Alternatively, instead of setting a system parameter, it may be possible to employ the structure in which the PG plane cropping unit 9702 directly refers to the playlist.

[0885] In the case where the video_shift_mode indicates "Up" or "Down", the shift amount of the video plane may be fixed to the size of each of the black frames provided in the upper and lower sides of the plane (131 pixels in the example shown in FIG. 4). Alternatively, an author or a user may set the shift amount without limitation. Further alternatively, it may be employed to prepare a new system parameter, store the shift amount in the new system parameter, and set the shift amount via a BD program or a player OSD.

[0886] In the structure of the plane overlaying described with reference to FIG. 84, the description is given on processing of shifting the whole plane using a value stored in the PSR33. Alternatively, the value may be used as a value to be added to a display position of the PG in the composition information. For example, in the case where the display position of the PG in the composition information is (x,y) and the video_shift_mode indicates "Keep", the PG decoder displays a corresponding subtitle data entry in a position indicated by (x,y + PG_shift_value_for_UP). With such a structure, processing is reduced compared with plane shift. In such a use case, PG_shift_value_for_UP may be stored in the composition information.

[0887] As shown in FIG 39, in the case where the video_shift_mode indicates "Up" or "Down",

plane shift results in a cropped region. Accordingly, there only needs to make a restriction such that no subtitle data is in the cropped region. In other words, as shown in the right side of FIG. 39, since a region other than a region surrounded by a dashed line has a possibility to be cropped, a display position of the PG is restricted such that no subtitle data is displayed on the region other than the region surrounded by the dashed line. The coordinate of the region is represented by $(0, PG_v_shfit_value_for_Down)$, $(0, height + PG_v_shfit_value_for_Up)$, $(width, PG_v_shfit_value_for_Down)$, and $(width, height + PG_v_shfit_value_for_Up)$. For example, if $PG_v_shfit_value_for_Up$ indicates $-a$ and $PG_v_shfit_value_for_Down$ indicates $+b$, the region is represented by $(0, b)$, $(0, height - a)$, $(width, b)$, and $(width, height - a)$. As the constraint conditions for PG, the display position is restricted so as not to go beyond the above region, the display position to which the size of an object to be displayed is added is restricted so as not to go beyond the above region, the display position of the window is restricted so as not to go beyond the above region, and the display position of the window to which the window size is added is restricted so as not to go beyond the above region, for example. Such constraint conditions can prevent display of a partially lacking.

[0888] Note that "video_shift_mode" may be added to the stream additional information 1311 of the stream selection information, as shown in FIG. 85. In this case, the structure of plane overlay processing performed in the 2D/3D playback device is as shown in FIG. 86. The structure shown in FIG. 86 includes a PSR34 in addition. The PSR34 stores therein a flag of On/Off indicating whether to perform video shift. In other words, the PSR34 having the value 1 indicates to perform video shift. The PSR34 having the value 0 indicates not to perform video shift. The PSR34 that is a flag of On/Off indicating whether to perform video shift is controlled by a program execution unit or the like in accordance with a menu, for example. The PSR34 may be set in accordance with a user operation such as OSD of a player. A video shift mode is stored in the PSR32. A value of the video shift mode is set based on additional information of a subtitle stream selected by PG stream selection. If the PSR34 indicates On, the video plane cropping unit 9701 performs cropping processing of video plane based on video_shift_mode set in the PSR32. If the PSR34 indicates Off, the video plane cropping unit 9701 does not perform the cropping processing. With such a structure, it is possible to set an appropriate video_shift_mode for each subtitle.

[0889] As shown in FIG. 85, video_shift_mode is stored in the stream additional information 1311 of the stream selection information such that PG streams whose shift modes video_shift_mode has the same attribute are registered in a row in the stream selection table. A remote control of the 2D/3D playback device generally includes a subtitle switching button. A user operation is defined such that each time the user presses the subtitle switching button, PG streams sequentially switch in the order of subtitle streams registered in the stream selection table. In the case where the user switches a subtitle using the subtitle switching button of the remote control, the video plane frequently moves upward and downward. As a result, the video becomes difficult to watch, and the user has an uncomfortable feeling. Accordingly, PG streams whose shift modes video_shift_mode has the same attribute are registered in a row in the stream selection table, as shown in FIG. 13. For example, in the example shown in FIG. 13, subtitle entries 1-3 each have a video_shift_mode=Keep, subtitle

entries 4-5 each have a video_shift_mode=Up, and subtitle entries 6-9 each have a video_shift_mode=Down. By collectively arranging subtitles having the same video shift mode in this way, it is possible to prevent frequent shift of the video plane.

[0890] In the case where the video_shift_mode instantly switches among "Keep", "Up", and "Down", the user feels unnatural. Accordingly, the video_shift_mode preferably switches among "Keep", "Up", and "Down" with a smooth effect. In this case, shift processing of the PG plane is preferably performed after completion of shift of the video plane.

[0891] In the present embodiment, the method has been described in which black frames are dynamically collected in the upper region or the lower region on the screen. Alternatively, the following structure may be employed, as shown in the upper level of FIG. 87. Specifically, a main feature video is arranged not in the middle of the screen but a slightly upper side of the screen so as to create a video stream, more black frames are arranged in the lower side so as to use the lower side for displaying subtitles. With such a structure, the black frames do not need to be dynamically changed for displaying subtitles. As a result, the videos do not move upward and downward, and the user does not feel uncomfortable.

[0892] As described with reference to FIG 81, in the palette information of the PG stream, a clear and colorless color is fixedly assigned to the color whose ID is 255. The 2D/3D playback device may control the value of this color to create a black frame. Specifically, the value of the color whose ID is 255 is stored in a system parameter PSR37. The 2D/3D playback device changes the color whose ID is 255 of the PG plane in accordance with the PSR37. With such a structure, by setting a background color of a subtitle to the color whose ID is 255, the subtitle is displayed using a transparent color and the background can be seen through the subtitle in the normal state, as shown in the left side on the lower level of FIG 87. By changing the color whose ID is 255 to an untransparent color, it is possible to change the background color of the subtitle, as shown in the right side on the lower level of FIG. 87. The value of PSR37 can be set on the menu screen of the BD program or the like.

(Embodiment 9)

[0893] The present embodiment describes an example structure of a playback device (FIG. 100) for playing back the data of the structure described in an earlier embodiment, which is realized by using an integrated circuit 3.

[0894] A medium interface unit 1 receives (reads) data from the medium, and transfers the data to the integrated circuit 3. Note that the medium interface unit 1 receives the data of the structure described in the earlier embodiment. The medium interface unit 1 is, for example: a disc drive when the medium is the optical disc or hard disk; a card interface when the medium is the semiconductor memory such as the SD card or the USB memory; a CAN tuner or Si tuner when the medium is broadcast waves of broadcast including the CATV; or a network interface when the medium is the Ethernet™, wireless LAN, or wireless public line.

[0895] A memory 2 is a memory for temporarily storing the data received (read) from the medium, and the data that is being processed by the integrated circuit 3. For example, the SDRAM (Synchronous Dynamic Random Access Memory), DDRx SDRAM (Double-Data-Rate Synchronous Dynamic Random Access Memory; x=1,2,3...) or the like is used as the memory 2. Note that the number of the memories 2 is not fixed, but may be one or two or more, depending on the necessity.

[0896] The integrated circuit 3 is a system LSI for performing the video/audio processing onto the data transferred from the interface unit 1, and includes a main control unit 6, a stream processing unit 5, a signal processing unit 7, an AV output unit 8, and a memory control unit 9.

[0897] The main control unit 6 includes a processor core having the timer function and the interrupt function. The processor core controls the integrated circuit 3 as a whole according to the program stored in the program memory or the like. Note that the basic software such as the OS (operating software) is stored in the program memory or the like in advance.

[0898] The stream processing unit 5, under the control of the main control unit 6, receives the data transferred from the medium via the interface unit 1 and stores it into the memory 2 via the data bus in the integrated circuit 3. The stream processing unit 5, under the control of the main control unit 6, also separates the received data into the video-base data and the audio-base data. As described earlier, on the medium, AV clips for 2D/L including left-view video stream and AV clips for R including right-view video stream are arranged in an interleaved manner, in the state where each clip is divided into some Extents. Accordingly, the main control unit 6 performs the control so that, when the integrated circuit 3 receives the left-eye data including left-view video stream, the received data is stored in the first region in the memory 2; and when the integrated circuit 3 receives the right-eye data including right-view video stream, the received data is stored in the second region in the memory 2. Note that the left-eye data belongs to the left-eye Extent, and the right-eye data belongs to the right-eye Extent. Also note that the first and second regions in the memory 2 may be regions generated by dividing a memory logically, or may be physically different memories. Further note that although the present embodiment presumes that the left-eye data including the left-view video stream is the main-view data, and the right-eye data including the right-view video stream is the sub-view data, the right-eye data may be the main-view data and the left-eye data may be the sub-view data. Also, the graphics stream is multiplexed in either or both of the main-view data and the sub-view data.

[0899] The signal processing unit 7, under the control of the main control unit 6, decodes, by an appropriate method, the video-base data and the audio-base data separated by the stream processing unit 5. The video-base data has been recorded after being encoded by a method such as MPEG-2, MPEG-4 AVC, MPEG-4 MVC, or SMPTE VC-1. Also, the audio-base data has been recorded after being compress-encoded by a method such as Dolby AC-3, Dolby Digital Plus, MLP, DTS, DTS-HD, or Linear PCM. Thus, the signal processing unit 7 decodes the video-base data and the audio-base data by the methods corresponding thereto. Models of

the signal processing unit 7 are various decoders of Embodiment 1 shown in FIG. 16.

[0900] The memory control unit 9 mediates the access to the memory 2 from each functional block in the integrated circuit 3.

[0901] The AV output unit 8, under the control of the main control unit 6, performs the superimposing of the video-base data having been decoded by the signal processing unit 7, or format conversion of the video-base data and the like, and outputs the data subjected to such processes to the outside of the integrated circuit 3.

[0902] FIG 101 is a functional block diagram showing a typical structure of the stream processing unit 5. The stream processing unit 5 includes a device/stream interface unit 51, a demultiplexing unit 52, and a switching unit 53.

[0903] The device/stream interface unit 51 is an interface for transferring data between the interface unit 1 and the integrated circuit 3. The device/stream interface unit 51 may be: SATA (Serial Advanced Technology Attachment), ATAPI (Advanced Technology Attachment Packet Interface), or PATA (Parallel Advanced Technology Attachment) when the medium is the optical disc or the hard disk; a card interface when the medium is the semiconductor memory such as the SD card or the USB memory; a tuner interface when the medium is broadcast waves of broadcast including the CATV; or a network interface when the medium is the Ethernet, wireless LAN, or wireless public line. The device/stream interface unit 51 may have a part of the function of the interface unit 1, or the interface unit 1 may be embedded in the integrated circuit 3, depending on the type of the medium.

[0904] The demultiplexing unit 52 separates the playback data, transferred from the medium, including video and audio, into the video-base data and the audio-base data. Each Extent, having been described earlier, is composed of source packets of video, audio, PG (subtitle), IG (menu) and the like (dependent source packets may not include audio). The demultiplexing unit 52 separates the playback data into video-base TS packets and audio-base TS packets based on the PID (identifier) included in each source packet. The demultiplexing unit 52 transfers the data after the separation to the signal processing unit 7. The data on which the processing has been performed is directly transferred to the signal processing unit 7, or is stored in the memory 2 and then transferred to the signal processing unit 7. A model of the demultiplexing unit 52 is, for example, the source depacketizer and the PID filter of Embodiment 8 as shown in FIG. 79. Also, a graphics stream, as a single stream, which has not been multiplexed with main view data or sub view data is transmitted to the signal processing unit 7 without being processed by the demultiplexing unit 52.

[0905] The switching unit 53 switches the output destination (storage destination) so that, when the device/stream interface unit 51 receives the left-eye data, the received data is stored in the first region in the memory 2; and when the switching unit 53 receives the right-eye data, the received data is stored in the second region in the memory 2. Here, the switching unit 53 is, for example, DMAC (Direct Memory Access Controller). FIG. 102 is a conceptual diagram

showing the switching unit 53 and the peripheral when the switching unit 53 is DMAC. The DMAC, under the control of the main control unit 6, transmits the data received by the device stream interface and the data storage destination address to the memory control unit 9. More specifically, the DMAC switches the output destination (storage destination) depending on the received data, by transmitting Address 1 (the first storage region) to the memory control unit 9 when the device stream interface receives the left-eye data, and transmitting Address 2 (the second storage region) to the memory control unit 9 when the device stream interface receives the right-eye data. The memory control unit 9 stores data into the memory 2 in accordance with the storage destination address sent from the DMAC. Note that a dedicated circuit for controlling the switching unit 53 may be provided, instead of the main control unit 6.

[0906] In the above description, the device/stream interface unit 51, demultiplexing unit 52, and switching unit 53 are explained as a typical structure of the stream processing unit 5. However, the stream processing unit 5 may further include an encryption engine unit for decrypting received encrypted data, key data or the like, a secure management unit for controlling the execution of a device authentication protocol between the medium and the playback device and for holding a secret key, and a controller for the direct memory access. In the above, it has been explained that, when the data received from the medium is stored into the memory 2, the switching unit 53 switches the storage destination depending on whether the received data is left-eye data or right-eye data. However, not limited to this, the data received from the medium may be temporarily stored into the memory 2, and then, when the data is to be transferred to the demultiplexing unit 52, the data may be separated into the left-eye data and the right-eye data.

[0907] FIG. 103 is a functional block diagram showing a typical structure of the AV output unit 8. The AV output unit 8 includes an image superimposing unit 81, a video output format converting unit 82, and an audio/video output interface unit 83.

[0908] The image superimposing unit 81 superimposes the decoded video-base data. More specifically, the image superimposing unit 81 superimposes the PG (subtitle) and the IG (menu) onto the left-view video data or the right-view video data in units of pictures. A model of the image superimposing unit 81 is, for example, Embodiment 1 and FIGs. 20-22. More specifically, decoded video data and subtitle data are stored in a region of the memory 2 for storing data to be rendered in each plane. Here, the plane is a region included in the memory 2 or a virtual space. The image superimposing unit 81 superimposes a left-view plane with a subtitle plane corresponding thereto, and superimposes a right-view plane with a subtitle plane corresponding thereto. Then, based on a region-saving flag corresponding to subtitle data (stream) to be superimposed, the left-view plane and the right-view plane are each superimposed with the subtitle data such that the subtitle data is superimposed in a display region for the subtitle data indicated by the region-saving flag (for example, Embodiment 1 and FIG. 12). In other words, if the region-saving flag indicates the display region for the subtitle data as the upper end, the left-view plane and the right-view plane are each shifted downward in the vertical coordinate, and superimposed with subtitle data. If the region-saving flag indicates the display region for the subtitle data as the lower end, the left-view plane and the

right-view plane are each shifted upward in the vertical coordinate, and superimposed with subtitle data.

[0909] The video output format converting unit 82 performs the following processes and the like as necessary: the resize process for enlarging or reducing the decoded video-base data; the IP conversion process for converting the scanning method from the progressive method to the interlace method and vice versa; the noise reduction process for removing the noise; and the frame rate conversion process for converting the frame rate.

[0910] The audio/video output interface unit 83 encodes, in accordance with the data transmission format, the video-base data, which has been subjected to the image superimposing and the format conversion, and the decoded audio-base data. Note that, as will be described later, the audio/video output interface unit 83 may be provided outside the integrated circuit 3.

[0911] FIG. 104 is an example structure showing the AV output unit 8, or the data output part of the playback device in more detail. The integrated circuit 3 of the present embodiment and the playback device support a plurality of data transmission formats for the video-base data and the audio-base data. The audio/video output interface unit 83 shown in FIG 103 corresponds to an analog video output interface unit 83a, a digital video/audio output interface unit 83b, and an analog audio output interface unit 83c.

[0912] The analog video output interface unit 83a converts and encodes the video-base data, which has been subjected to the image superimposing process and the output format conversion process, into the analog video signal format, and outputs the conversion result. The analog video output interface unit 83a is, for example: a composite video encoder that supports any of the NTSC method, PAL method, and SECAM method; an encoder for the S image signal (Y/C separation); an encoder for the component image signal; or a DAC (D/A converter).

[0913] The digital video/audio output interface unit 83b overlays the decoded audio-base data with the video-base data having been subjected to the image superimposing and the output format conversion, encrypts the overlaid data, encodes in accordance with the data transmission standard, and outputs the encoded data. The digital video/audio output interface unit 83b is, for example, HDMI (High-Definition Multimedia Interface).

[0914] The analog audio output interface unit 83c, being an audio DAC or the like, performs the D/A conversion onto the decoded audio-base data, and outputs analog audio data.

[0915] The transmission format of the video-base data and audio-base data may be switched depending on the data receiving device (data input terminal) supported by the display device/speaker 4, or may be switched in accordance with the selection by the user. Furthermore, it is possible to transmit a plurality of pieces of data corresponding to the same content in parallel by a plurality of transmission formats, not limited to the transmission by a

single transmission format.

[0916] In the above description, the image superimposing unit 81, video output format converting unit 82, and audio/video output interface unit 83 are explained as a typical structure of the AV output unit 8. However, the AV output unit 8 may further include, for example, a graphics engine unit for performing the graphics processing such as the filter process, image overlaying, curvature drawing, and 3D display.

[0917] This completes the description of the structure of the playback device in the present embodiment. Note that all of the functional blocks included in the integrated circuit 3 may not be embedded, and that, conversely, the memory 2 shown in FIG. 100 may be embedded in the integrated circuit 3. Also, in the present embodiment, the main control unit 6 and the signal processing unit 7 have been described as different functional blocks. However, not limited to this, the main control unit 6 may perform a part of the process performed by the signal processing unit 7.

[0918] Also, as shown in FIG. 107, the process performed by the playback device in the present embodiment may be performed by the display device. In that case, the data received by the medium interface unit 1 is subjected to the signal processing performed by the integrated circuit 3, and the video data after this processing is output via the display drive unit 10 onto the display panel 11 and the audio data after this processing is output onto the speaker 12. Here, the AV output unit 8 has, for example, a structure shown in FIG. 108, and the data is transferred via the video output interface unit 84 and the audio output interface unit 85 that are provided inside or outside the integrated circuit 3. Note that the device may be provided with a plurality of video output interface units 84 and a plurality of audio output interface units 85, or may be provided with an interface unit that is common to the video and the audio.

[0919] The route of the control buses and the data buses in the integrated circuit 3 is designed in an arbitrary manner depending on the processing procedure of each processing block or the contents of the processing. However, the data buses may be arranged so that the processing blocks are connected directly as shown in FIG 105, or maybe arranged so that the processing blocks are connected via the memory 2 (the memory control unit 9) as shown in FIG. 106.

[0920] The integrated circuit 3 may be a multi-chip module that is generated by enclosing a plurality of chips into one package, and its outer appearance is one LSI.

[0921] It is also possible to realize the system LSI by using the FPGA (Field Programmable Gate Array) that can be re-programmed after the manufacturing of the LSI, or the reconfigurable processor in which the connection and setting of the circuit cells inside the LSI can be reconfigured.

[0922] Next, the operation of the playback device having the above-described structure will be

explained.

[0923] FIG. 109 is a flow chart showing a playback procedure in which data is received (read) from the medium, is decoded, and is output as a video signal and an audio signal.

S1: data is received (read) from the medium (the medium interface unit 1 and the stream processing unit 5).

S2: the data received (read) in S1 is separated into various data (the video-base data and the audio-base data) (the stream processing unit 5).

S3: the various data generated by the separation in S2 are decoded by the appropriate format (the signal processing unit 7).

S4: among the various data decoded in S3, the video-base data is subjected to the superimposing process (the AV output unit 8).

S6: the video-base data and the audio-base data having been subjected to the processes in S2 through S5 are output (the AV output unit 8).

[0924] FIG. 110 is a flow chart showing in more detail the playback procedure. Each of the operations and processes is performed under the control of the main control unit 6.

S101: the device/stream interface unit 51 of the stream processing unit 5 receives (reads out) data (playlist, clip information, etc.) which is other than the data stored in the medium to be played back and is necessary for playback of the data, via the interface unit 1, and stores the received data into the memory 2 (the interface unit 1, the device/stream interface unit 51, the memory control unit 9, and the memory 2).

S102: the main control unit 6 recognizes the compression method of the video and audio data stored in the medium by referring to the stream attribute included in the received clip information, and initializes the signal processing unit 7 so that the corresponding decode processing can be performed (the main control unit 6).

S103: the device/stream interface unit 51 of the stream processing unit 5 receives (reads out) the data of video/audio that is to be played back, from the medium via the interface unit 1, and stores the received data into the memory 2 via the stream processing unit 5 and the memory control unit 9. Note that the data is received (read) in units of Extents, and the main control unit 6 controls the switching unit 53 so that, when the left-eye data is received (read), the received data is stored in the first region; and when the right-eye data is received (read), the received data is stored in the second region, and the switching unit 53 switches the data output destination (storage destination) (the interface unit 1, the device/stream interface unit 51, the main control unit 6, the switching unit 53, the memory control unit 9, and the memory 2).

S104: the data stored in the memory 2 is transferred to the demultiplexing unit 52 of the

stream processing unit 5, and the demultiplexing unit 52 identifies the video-base data (main video, sub-video), PG (subtitle), IG (menu), and audio-base data (audio, sub-audio) based on the PIDs included in the source packets constituting the stream data, and transfers the data to each corresponding decoder in the signal processing unit 7 in units of TS packets (the demultiplexing unit 52).

S105: each decoder in the signal processing unit 7 performs the decode process onto the transferred TS packets by the appropriate method (the signal processing unit 7).

S106: among the video-base data decoded by the signal processing unit 7, the data corresponding to the left-view video stream and the right-view video stream is resized based on the display device 4 (the video output format converting unit 82).

S107: the PG (subtitle) and IG (menu) are superimposed onto the video stream resized in S106 (the image superimposing unit 81).

S108: the IP conversion, which is a conversion of the scanning method, is performed onto the video data after the superimposing in S107 (the video output format converting unit 82).

S109: the encoding, D/A conversion and the like are performed onto video-base data and the audio-base data having been subjected to the above-described processes, based on the data output format of the display device/speaker or the data transmission format for transmission to the display device/speaker 4. For example, processing is performed on the video-base data and the audio-base data so as to be outputted in analog or digital format. The composite video signal, the S image signal, the component image signal and the like are supported for the analog output of the video-base data. Also, HDMI is supported for the digital output of the video-base data and the audio-base data. (the audio/video output interface unit 83).

S110: the video-base data and the audio-base data having been subjected to the process in S109 is output and transmitted to the display device/speaker (the audio/video output interface unit 83, the display device/speaker 4).

[0925] This completes the description of the operation procedure of the playback device in the present embodiment. Note that the result of process may be temporarily stored into the memory 2 each time a process is completed. Note that when the playback process is performed by the display device shown in FIG. 107, the operation procedure is basically the same, and functional blocks corresponding to the functional blocks of the playback device shown in FIG. 100 perform the processes similarly. Also, in the above operation procedure, the video output format converting unit 82 performs the resize process and the IP conversion process. However, not limited to this, the processes may be omitted as necessary, or other processes (noise reduction process, frame rate conversion process, etc.) may be performed. Furthermore, the processing procedures may be changed if possible.

(Supplementary Notes)

[0926] Up to now, the present invention has been described through the best embodiments that the Applicant recognize as of now. However, further improvements or changes can be added regarding the following technical topics. Whether to select any of the embodiments or the improvements and changes to implement the invention is optional and may be determined by the subjectivity of the implementer.

(Offset Metadata)

[0927] Offset metadata described in the embodiments may be realized by not only the data formats described above but also other data formats. The following lists other data formats of offset metadata.

[0928] FIG. 89 shows a first data format of offset metadata.

[0929] In the first data format, offset metadata is stored in a clip information file. In this case, as shown in FIG. 89A, it may be possible to include table information in which PTS and offset amounts of a plurality of pieces of offset_id are included. A specific syntax is shown in FIG. 89B.

[0930] FIG. 90 shows a second data format of offset metadata. Offset metadata described in the embodiments is stored in a head access unit of each GOP, and is applied to a frame included in the GOP. In the second data format, when offset metadata is stored in a clip information file, offset metadata is stored for each entry point, as shown in FIG 90A. A specific syntax is structured so as to correspond to EP_ID that is an ID of an entry point, as shown in FIG. 90B. With such a structure, a PTS is identifiable by the EP_ID. Accordingly, since a value of the PTS does not need to be stored compared with the data format shown in FIG. 89. This can reduce the data amount. Also, with such a structure, when offset metadata is stored in both of an access unit of a video stream and a clip information file, verification is easily performed for checking whether the same offset metadata is stored in the access unit and the clip information file.

[0931] FIG. 91 shows a third data format of offset metadata. In the embodiments, an offset sequence is stored in offset metadata for each offset sequence ID, and an offset value is referenced using a reference offset sequence ID for each PG stream. In the third data format, such offset metadata is stored in a playlist information file. FIG. 91A shows a syntax of offset metadata to be stored in a playlist information file. A first loop 11201 is a loop for a playitem. number_of_offsets[playitem] represents the number of offset entries of the playitem. number_of_offset_id[playitem] represents the number of offset sequence IDs. A second loop 11202 is a loop for offset entries of the playitem. Information included in one loop is defined as an offset entry. offset_frame_number represents the number of video frames starting with the head in the playitem. The offset_frame_number may represent PTS. However, by setting the

offset_frame_number to represent the number of frames, a data amount can be reduced. offset_frame_duration represents an interval in which an offset value between each two offset entries is inserted. number_of_suboffsets represents the number of offset values to be inserted into an interval between offset_frame_number[i] and a subsequent offset entry. FIG. 91B shows a relationship among the offset_frame_number[i], the offset_frame_duration[i], and the number_of_suboffsets[i]. An offset value is stored for each offset ID, as shown in the loop 11203. offset_frame_number may represent the number of differential video frames showing difference from an immediately previous offset entry.

[0932] FIG 92 shows a fourth data format of offset metadata. The fourth data format is another data format for storing offset metadata in a playlist information file. As shown in FIG. 92A, a flag (is_same_as_previous_playitem) is additionally included which shows whether a current playitem is the same as a previous playitem. In order to create a video image menu for BD-ROM loop, a structure is employed in which many playitems are repeated in a playlist as if there occurs an infinite loop of a playitem that refers to the same clip as shown in FIG. 92B. In this case, if the same offset metadata equal in number to playitems is prepared, a data amount excessively increases. As a result, it is necessary to increase a memory amount of the 2D/3D playback device. Accordingly, when the is_same_as_previous_playitem indicates 1, the 2D/3D playback device refers to a piece of offset metadata information of an immediately previous playitem. As a result, it is possible to reduce a data amount.

[0933] FIG. 93 shows a fifth data format of offset metadata.

The fifth data format is a yet another data format for storing offset metadata in a playlist information file. As shown in FIG. 93, a reference ID (ref_playitem_id_of_same_offset_metadata) for a playitem using the same offset metadata is additionally included. When "ref_playitem_id_of_same_offset_metadata" does not indicate 0xFFFF representing invalid, the 2D/3D playback device applies the offset metadata that is the same as a playitem indicated by the "ref_playitem_id_of_same_offset_metadata". With this structure, it is only necessary to define one piece of offset metadata with respect to a plurality of playitems having the same offset metadata. This can reduce the data amount.

[0934] FIG. 94 shows a sixth data format of offset metadata.

[0935] The sixth data format is a yet another format for storing offset metadata in a playlist information file.

[0936] In this data format, a header in which loop is performed in units of playitems and a region in which offset metadata is stored are separately provided, as shown in FIG. 94. A playitem is associated with a piece of offset metadata by offset_block_id. With this structure, in the case where a plurality of playitems using the same piece of offset metadata are included, it is only necessary to define one piece of offset metadata. This can reduce a data amount. Also, the header may store therein an address value (start_address) of a file in which a corresponding piece of offset metadata is stored. This structure facilitates access in units of playitems.

[0937] In the syntax shown in FIG. 89 to FIG. 94, an entry of offset metadata is composed of 7-bit "offset_direction,offset_value". Alternatively, offset metadata may be prepared by using the difference from a certain sequence of offset metadata. This structure can decrease the size of "offset_direction,offset_value".

[0938] As another data format, it may be possible to employ a structure in which offset metadata is embedded on an audio stream using audio watermark technique. Alternatively, it may be possible to employ a structure in which offset metadata is embedded on a video stream using video watermark technique.

(PG stream)

[0939] In order to reduce the number of subtitles to suppress increase in the band of streams, it is effective to share one PG stream as one of a PG stream for use in the "1 plane + offset" method PG stream and either a left-eye or right-eye PG stream for use in the 2 plane L/R method.

[0940] However, if such a structure is employed, there is a case where shift occurs between a position in which a depth between the left-eye graphics and the right eye graphics is large (position in which the graphicses protrude toward the user) and a position in which the depth is small, as shown in FIG. 95. In such a case, each of the graphicses moves between right and left. In an example shown in FIG. 95, if subtitle data is shift from a scene having a small depth to a scene having a large depth, the left-eye graphics is shifted in the right direction, and the right eye graphics is shifted in the left direction. If the left-eye graphics is used for the 2D display and the 1 plane + offset method, the left-eye graphics is shifted in the left direction in the same way. This causes the user to feel uncomfortable.

[0941] In view of this problem, in order to display the 2D display subtitle and the 1 plane + offset method subtitle without causing the user to feel uncomfortable, the display position of the composition information is fixed, as shown in FIG. 96. Also, an offset (1_offset) for displaying as a 2 plane L/R method left-eye PG is separately prepared in the composition information. In the case where display is performed in accordance with the 2 plane L/R method, the 2D/3D playback device adds the offset value to the display position of the composition information for display. With such a structure, even in the case where the same stream is used for the 2D display subtitle, the 1 plane + offset method subtitle, and the 2 plane L/R method left-eye subtitle, it is possible to perform display in any display mode without causing the user to feel uncomfortable.

(Speed Increase in Jump Playback)

[0942] FIG. 97A shows the structure in which extents in a FileBase and a FileDependent are

interleaved with each other. In the figure, an inverted triangle attached to the head of a data region R[2] on the disc indicates a position of an entry point of the FileDependent on the disc. An inverted triangle attached to the head of a data region L[2] on the disc indicates a position of an entry point of the FileBase on the disc. Here, in the case where jump playback is performed from the entry point, the 2D/3D playback device loads data stored in R[2] that is a data region on the disc, and then starts decoding while reading L[2]. Until completion of loading the data stored in R[2], the 2D/3D playback device cannot read a subsequent L[2], and accordingly cannot start decoding.

[0943] In view of this, in order to reduce a time period from loading to the entry point to starting playback, the structure as shown in FIG. 97B is employed. In FIG. 97B, a File2D indicates data regions L[0], L[1], L[2] for 2D, and L[3] on the disc. A File3D indicates data regions L[0], L[1], L[2] for 3D, and L[3] on the disc. The L[2] for 2D and the L[2] for 3D are structured so as to have the same data. With such a structure, the same data can be read although different playback paths are used. FIG. 97B shows the structure in which data of a right eye AV clip corresponding to the data region L[2] for 3D is interleaved in small units (range indicated by an arrow 10701). With such a structure, in the case where the 2D/3D playback device starts playback from the entry point, it is possible to structure a head extent of the FileDependent smaller compared with the structure shown in FIG. 97A. This can reduce a time period from starting at the entry point to the start of decoding.

(Additional Information)

[0944] Additional information may be incorporated into an extension information field of playlist information, as an extension stream selection table that includes information elements shown below.

[0945] An "upper end region flag" is a flag indicating whether there is an upper end region during playback of a PG_text subtitle stream.

[0946] An "upper end region stream entry" includes: a sub-path identifier reference (ref_to_Subpath_id) specifying a sub-path to which a playback path of a PG_text subtitle stream belongs; a stream file reference (ref_to_subClip_entry_id) specifying a stream file in which the PG_text subtitle stream is stored; and a packet identifier (ref_to_stream_PID_subclip) of the PG_text subtitle stream in this stream file.

[0947] "Upper end region depth reference information" is reference information for referring an offset sequence for a PG_text subtitle stream in the case where subtitles are displayed in the upper end region, and indicates an offset sequence for a PG_text subtitle stream in the case where subtitles are displayed in the upper end region. The playback device should apply the offset, which is supplied by this field, to the PG plane.

[0948] A "lower end region flag" is a flag indicating whether there is a lower end region during

playback of a PG_text subtitle stream.

[0949] A "lower end region stream entry" includes: a sub-path identifier reference (ref_to_Subpath_id) specifying a sub-path to which a playback path of a PG_text subtitle stream belongs; a stream file reference (ref_to_subClip_entry_id) specifying a stream file in which the PG_text subtitle stream is stored; and a packet identifier (ref_to_stream_PID_subclip) of the PG_text subtitle stream in this stream file.

[0950] "Lower end region depth reference information" is reference information for referring an offset sequence for a PG_text subtitle stream in the case where subtitles are displayed in the lower end region, and indicates an offset sequence for a PG_text subtitle stream in the case where subtitles are displayed in the lower end region. The playback device should apply the offset, which is supplied by this field, to the PG plane.

(Playback of optical disc)

[0951] The BD-ROM drive is equipped with an optical head that includes a semiconductor laser, collimated lens, beam splitter, objective lens, collecting lens, and light detector. The light beams emitted from the semiconductor laser pass through the collimated lens, beam splitter, and objective lens, and are collected on the information surface of the optical disc.

[0952] The collected light beams are reflected/diffracted on the optical disc, pass through the objective lens, beam splitter, and collimated lens, and are collected in the light detector. A playback signal is generated depending on the amount of light collected in the light detector.

(Variations of recording medium)

[0953] The recording medium described in each Embodiment indicates a general package medium as a whole, including the optical disc and the semiconductor memory card. In each Embodiment, it is presumed, as one example, that the recording medium is an optical disc in which necessary data is recorded in advance (for example, an existing read-only optical disc such as the BD-ROM or DVD-ROM). However, the present invention is not limited to this. For example, the present invention may be implemented as follows: (i) obtain a 3D content that includes the data necessary for implementing the present invention and is distributed by a broadcast or via a network; (ii) record the 3D content into a writable optical disc (for example, an existing writable optical disc such as the BD-RE, DVD-RAM) by using a terminal device having the function of writing into an optical disc (the function may be embedded in a playback device, or the device may not necessarily be a playback device); and (iii) apply the optical disc recorded with the 3D content to the playback device of the present invention.

(Embodiments of semiconductor memory card recording device and playback device)

[0954] The following describes embodiments of the recording device for recording the data structure of each Embodiment into a semiconductor memory, and the playback device for playing back thereof.

[0955] First, the mechanism for protecting the copyright of the data recorded on the BD-ROM will be explained, as a presupposed technology.

[0956] Some of the data recorded on the BD-ROM may have been encrypted as necessitated in view of the confidentiality of the data.

[0957] For example, the BD-ROM may contain, as encrypted data, the data corresponding to a video stream, an audio stream, or a stream including these.

[0958] The following describes decryption of the encrypted data among the data recorded on the BD-ROM.

[0959] The semiconductor memory card playback device stores in advance data (for example, a device key) that corresponds to a key that is necessary for decrypting the encrypted data recorded on the BD-ROM.

[0960] On the other hand, the BD-ROM is recorded in advance with (i) data (for example, a medium key block (MKB) corresponding to the above-mentioned device key) that corresponds to a key that is necessary for decrypting the encrypted data, and (ii) encrypted data (for example, an encrypted title key corresponding to the above-mentioned device key and MKB) that is generated by encrypting the key itself that is necessary for decrypting the encrypted data. Note here that the device key, MKB, and encrypted title key are treated as a set, and are further associated with an identifier (for example, a volume ID) written in an area (called BCA) of the BD-ROM that cannot be copied in general. It is structured such that encrypted data cannot be decrypted if these elements are combined incorrectly. Only if the combination is correct, a key (for example, a title key that is obtained by decrypting the encrypted title key by using the above-mentioned device key, MKB, and volume ID) that is necessary for decrypting the encrypted data can be derived. The encrypted data can be decrypted by using the derived key.

[0961] When a playback device attempts to play back a BD-ROM loaded in the device, it cannot play back the encrypted data unless the device itself has a device key that makes a pair (or corresponds to) the encrypted title key and MKB recorded on the BD-ROM. This is because the key (title key) that is necessary for decrypting the encrypted data has been encrypted, and is recorded on the BD-ROM as the encrypted title key, and the key that is necessary for decrypting the encrypted data cannot be derived if the combination of the MKB and the device key is not correct.

[0962] Conversely, when the combination of the encrypted title key, MKB, device key, and volume ID is correct, the video stream and audio stream are decoded by the video decoder and the audio decoder with use of the above-mentioned key (for example, a title key that is obtained by decrypting the encrypted title key by using the device key, MKB, and volume ID) that is necessary for decrypting the encrypted data. The playback device is structured in this way.

[0963] This completes the description of the mechanism for protecting the copyright of the data recorded on the BD-ROM. It should be noted here that this mechanism is not limited to the BD-ROM, but may be applicable to, for example, a readable/writable semiconductor memory (such as a portable semiconductor memory such as the SD card) for the implementation.

[0964] Next, the playback procedure in the semiconductor memory card playback device will be described. In the case in which the playback device plays back an optical disc, it is structured to read data via an optical disc drive, for example. On the other hand, in the case in which the playback device plays back a semiconductor memory card, it is structured to read data via an interface for reading the data from the semiconductor memory card.

[0965] More specifically, the playback device may be structured such that, when a semiconductor memory card is inserted into a slot provided in the playback device, the playback device and the semiconductor memory card are electrically connected with each other via the semiconductor memory card interface, and the playback device reads out data from the semiconductor memory card via the semiconductor memory card interface.

(Embodiments of receiving device)

[0966] The playback device explained in each Embodiment may be realized as a terminal device that receives data (distribution data) that corresponds to the data explained in each Embodiment from a distribution server for an electronic distribution service, and records the received data into a semiconductor memory card.

[0967] Such a terminal device may be realized by structuring the playback device explained in each Embodiment so as to perform such operations, or may be realized as a dedicated terminal device that is different from the playback device explained in each Embodiment and stores the distribution data into a semiconductor memory card. Here, a case where the playback device is used will be explained. Also, in this explanation, an SD card is used as the recording-destination semiconductor memory.

[0968] When the playback device is to record distribution data into an SD memory card inserted in a slot provided therein, the playback device first send requests a distribution server that stores distribution data, to transmit the distribution data. In so doing, the playback device reads out identification information for uniquely identifying the inserted SD memory card (for

example, identification information uniquely assigned to each SD memory card, more specifically, the serial number or the like of the SD memory card), from the SD memory card, and transmits the read identification information to the distribution server together with the distribution request.

[0969] The identification information for uniquely identifying the SD memory card corresponds to, for example, the volume ID having been described earlier.

[0970] On the other hand, the distribution server stores necessary data (for example, video stream, audio stream and the like) in an encrypted state such that the necessary data can be decrypted by using a predetermined key (for example, a title key).

[0971] The distribution server, for example, holds a private key so that it can dynamically generate different pieces of public key information respectively in correspondence with identification numbers uniquely assigned to each semiconductor memory card.

[0972] Also, the distribution server is structured to be able to encrypt the key (title key) itself that is necessary for decrypting the encrypted data (that is to say, the distribution server is structured to be able to generate an encrypted title key).

[0973] The generated public key information includes, for example, information corresponding to the above-described MKB, volume ID, and encrypted title key. With this structure, when, for example, a combination of the identification number of the semiconductor memory card, the public key contained in the public key information which will be explained later, and the device key that is recorded in the playback device in advance, is correct, a key (for example, a title key that is obtained by decrypting the encrypted title key by using the device key, the MKB, and the identification number of the semiconductor memory) necessary for decrypting the encrypted data is obtained, and the encrypted data is decrypted by using the obtained necessary key (title key).

[0974] Following this, the playback device records the received piece of public key information and distribution data into a recording region of the semiconductor memory card being inserted in the slot thereof.

[0975] Next, a description is given of an example of the method for decrypting and playing back the encrypted data among the data contained in the public key information and distribution data recorded in the recording region of the semiconductor memory card.

[0976] The received public key information stores, for example, a public key (for example, the above-described MKB and encrypted title key), signature information, identification number of the semiconductor memory card, and device list being information regarding devices to be invalidated.

[0977] The signature information includes, for example, a hash value of the public key

information.

[0978] The device list is, for example, information for identifying the devices that might be played back in an unauthorized manner. The information, for example, is used to uniquely identify the devices, parts of the devices, and functions (programs) that might be played back in an unauthorized manner, and is composed of, for example, the device key and the identification number of the playback device that are recorded in the playback device in advance, and the identification number of the decoder provided in the playback device.

[0979] The following describes playing back the encrypted data among the distribution data recorded in the recording region of the semiconductor memory card.

[0980] First, it is checked whether or not the decryption key itself can be used, before the encrypted data is decrypted by using the decryption key.

[0981] More specifically, the following checks are conducted. (1) A check on whether the identification information of the semiconductor memory card contained in the public key information matches the identification number of the semiconductor memory card stored in advance in the semiconductor memory card. (2) A check on whether the hash value of the public key information calculated in the playback device matches the hash value included in the signature information. (3) A check, based on the information included in the device list, on whether the playback device to perform the playback is authentic (for example, the device key shown in the device list included in the public key information matches the device key stored in advance in the playback device). These checks may be performed in any order.

[0982] After the above described checks (1) through (3), the playback device performs a control not to decrypt the encrypted data when any of the following conditions is satisfied: (i) the identification information of the semiconductor memory card contained in the public key information does not match the identification number of the semiconductor memory card stored in advance in the semiconductor memory card; (ii) the hash value of the public key information calculated in the playback device does not match the hash value included in the signature information; and (iii) the playback device to perform the playback is not authentic.

[0983] On the other hand, when all of the conditions: (i) the identification information of the semiconductor memory card contained in the public key information matches the identification number of the semiconductor memory card stored in advance in the semiconductor memory card; (ii) the hash value of the public key information calculated in the playback device matches the hash value included in the signature information; and (iii) the playback device to perform the playback is authentic, are satisfied, it is judged that the combination of the identification number of the semiconductor memory, the public key contained in the public key information, and the device key that is recorded in the playback device in advance, is correct, and the encrypted data is decrypted by using the key necessary for the decryption (the title key that is obtained by decrypting the encrypted title key by using the device key, the MKB, and the identification number of the semiconductor memory).

[0984] When the encrypted data is, for example, a video stream and an audio stream, the video decoder decrypts (decodes) the video stream by using the above-described key necessary for the decryption (the title key that is obtained by decrypting the encrypted title key), and the audio decoder decrypts (decodes) the audio stream by using the above-described key necessary for the decryption.

[0985] With such a structure, when devices, parts of the devices, and functions (programs) that might be used in an unauthorized manner are known at the time of the electronic distribution, a device list showing such devices and the like may be distributed. This enables the playback device having received the list to inhibit the decryption with use of the public key information (public key itself) when the playback device includes anything shown in the list. Therefore, even if the combination of the identification number of the semiconductor memory, the public key itself contained in the public key information, and the device key that is recorded in the playback device in advance, is correct, a control is performed not to decrypt the encrypted data. This makes it possible to prevent the distribution data from being used by an unauthentic device.

[0986] It is preferable that the identifier of the semiconductor memory card that is recorded in advance in the semiconductor memory card is stored in a highly secure recording region. This is because, when the identification number (for example, the serial number of the SD memory card) that is recorded in advance in the semiconductor memory card is tampered with, unauthorized copying becomes easy. More specifically, unique, although different identification numbers are respectively assigned to semiconductor memory cards, if the identification numbers are tampered with to be the same, the above-described judgment in (1) does not make sense, and as many semiconductor memory cards as tampering may be copied in an unauthorized manner.

[0987] For this reason, it is preferable that information such as the identification number of the semiconductor memory card is stored in a highly secure recording region.

[0988] To realize this, the semiconductor memory card, for example, may have a structure in which a recording region for recording highly confidential data such as the identifier of the semiconductor memory card (hereinafter, the recording region is referred to as a "second recording region") is provided separately from a recording region for recording regular data (hereinafter, the recording region is referred to as a "first recording region"), a control circuit for controlling accesses to the second recording region is provided, and the second recording region is accessible only through the control circuit.

[0989] For example, data may be encrypted so that encrypted data is recorded in the second recording region, and the control circuit may be embedded with a circuit for decrypting the encrypted data. In this structure, when an access is made to the second recording region, the control circuit decrypts the encrypted data and returns decrypted data. As another example, the control circuit may hold information indicating the location where the data is stored in the

second recording region, and when an access is made to the second recording region, the control circuit identifies the corresponding storage location of the data, and returns data that is read from the identified storage location.

[0990] An application, which is running on the playback device and is to record data onto the semiconductor memory card with use of the electronic distribution, issues, to the control circuit via a memory card interface, an access request requesting to access the data (for example, the identification number of the semiconductor memory card) recorded in the second recording region. Upon receiving the request, the control circuit reads out the data from the second recording region and returns the data to the application running on the playback device. It sends the identification number of the semiconductor memory card and requests the distribution server to distribute the data such as the public key information, and corresponding distribution data. The public key information and corresponding distribution data that are sent from the distribution server are recorded into the first recording region.

[0991] Also, it is preferable that the application, which is running on the playback device and is to record data onto the semiconductor memory card with use of the electronic distribution, checks in advance whether or not the application is tampered with before it issues, to the control circuit via a memory card interface, an access request requesting to access the data (for example, the identification number of the semiconductor memory card) recorded in the second recording region. For checking this, an existing digital certificate conforming to the X.509 standard, for example, may be used.

[0992] Also, the distribution data recorded in the first recording region of the semiconductor memory card 1 may not necessarily be accessed via the control circuit provided in the semiconductor memory card.

[Industrial Applicability]

[0993] The information recording medium of the present invention stores a 3D image, but can be played back in both 2D-image playback devices and 3D-image playback devices. This makes it possible to distribute movie contents such as movie titles storing 3D images, without causing the consumers to be conscious about the compatibility. This activates the movie market and commercial device market. Accordingly, the recording medium and the playback device of the present invention have high usability in the movie industry and commercial device industry.

[Reference Signs List]

[0994]

100:

recording medium
200:
playback device
300:
display device
400:
3D glasses

REFERENCES CITED IN THE DESCRIPTION

This list of references cited by the applicant is for the reader's convenience only. It does not form part of the European patent document. Even though great care has been taken in compiling the references, errors or omissions cannot be excluded and the EPO disclaims all liability in this regard.

Patent documents cited in the description

- [JP3935507B \[0005\]](#)
- [US2005259147A1 \[0006\]](#)
- [WO9638918A2 \[0007\]](#)
- [WO2008044191A2 \[0008\]](#)

INFORMATIONSDISTRIBUERINGSMEDIUM TIL GENGIVELSE AF 3D-VIDEO OG GENGIVELSESDIREKTION

Patentkrav

1. Afspilningsindretning til afspilning af et registreringsmedium (100) med en video-stream, der danner et stereoskopisk billede, afspilningslisteinformationer og en undertitel-stream, hvor afspilningslisteinformationerne indbefatter en stream-udvalgstabel, der viser en stream-postering, en stream-attribut og yderligere informationer, for at den pågældende undertitel-stream kan afspilles i en monoskopisk afspilningsmodus, de yderligere informationer som et afbildningsområde af en undertitel i en stereoskopisk afspilningsmodus for afspilningsindretningen angiver en øvre ende eller en nedre ende i et videoplan, afspilningsindretningen omfatter:

et register (PSR32), som kan lagre informationer fra en videoforskydningsmodus for afspilningsindretningen, idet videoforskydningsmodussen indbefatter en opadgående forskydningsmodus og en nedadgående forskydningsmodus, og

en overlejringseenhed, som,

når afspilningsindretningen foretager den opadgående forskydningsmodus baseret på de informationer, som er lagret i registret (PSR32), hvis de yderligere informationer angiver den nedre ende i videoplanet, kan forskyde et billede som følge af afkodning af den pågældende video-stream i videoplanet i opadgående retning og forskyde undertiteldata som følge af afkodning af den pågældende undertitel-stream til et område i den nedre ende i videoplanet, hvor det forskudte billede ikke afbildes, således at billedet og undertiteldataene overlejres, og

når afspilningsindretningen foretager den nedadgående forskydningsmodus baseret på de informationer, som er lagret i registret (PSR32), hvis de yderligere informationer angiver den øvre ende i videoplanet, kan forskyde billedet i videoplanet i nedadgående retning og forskyde undertiteldataene til et område i den øvre ende i videoplanet, hvor det forskudte billede ikke afbildes, således at billedet og undertiteldataene overlejres.

2. Afspilningssystem til registreringsmedium, hvilket system omfatter:

afspilningsindretningen (200) ifølge krav 1; og

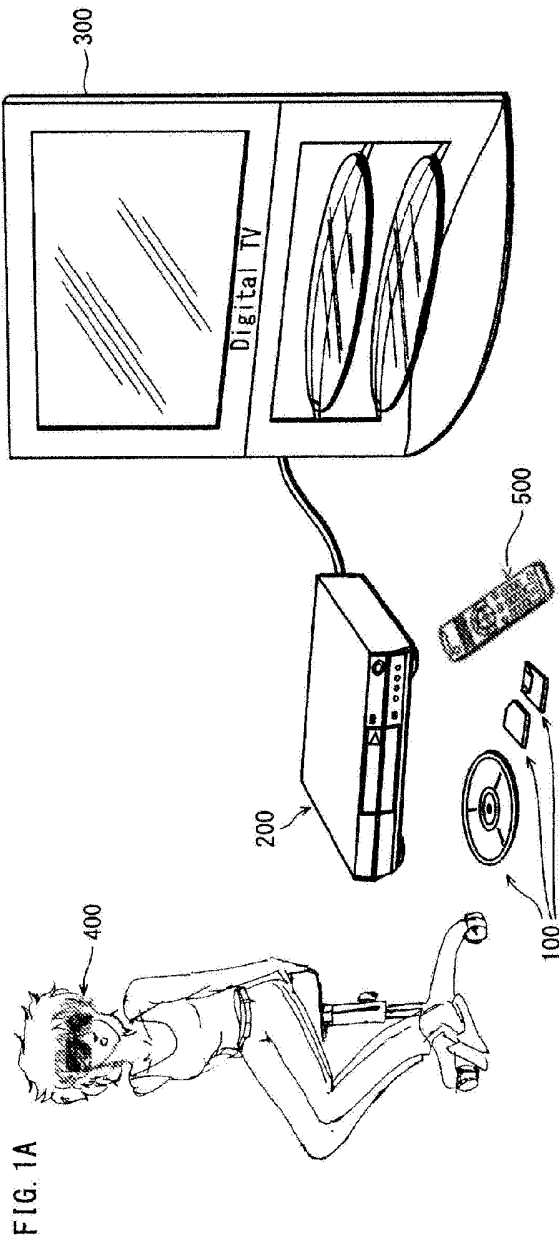
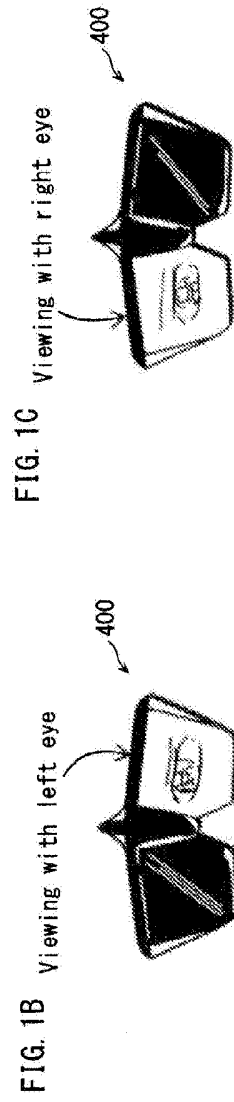
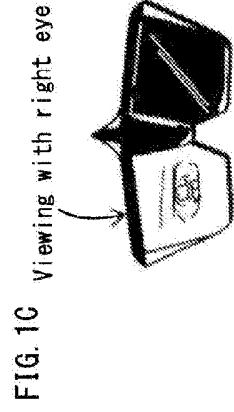
et registreringsmedium (100), hvor

registreringsmediet (100) har en video-stream, der danner et stereoskopisk billede, afspilningslisteinformationer og en undertitel-stream,

afspilningslisteinformationerne indbefatter en stream-udvalgstabel, der viser en stream-postering, en stream-attribut og yderligere informationer, for at den pågældende undertitel-stream kan afspilles i en monoskopisk afspilningsmodus, og

- 5 de yderligere informationer som et afbildningsområde af en undertitel i en stereoskopisk afspilningsmodus for afspilningsindretningen angiver en øvre ende eller en nedre ende i et videoplan.

DRAWINGS



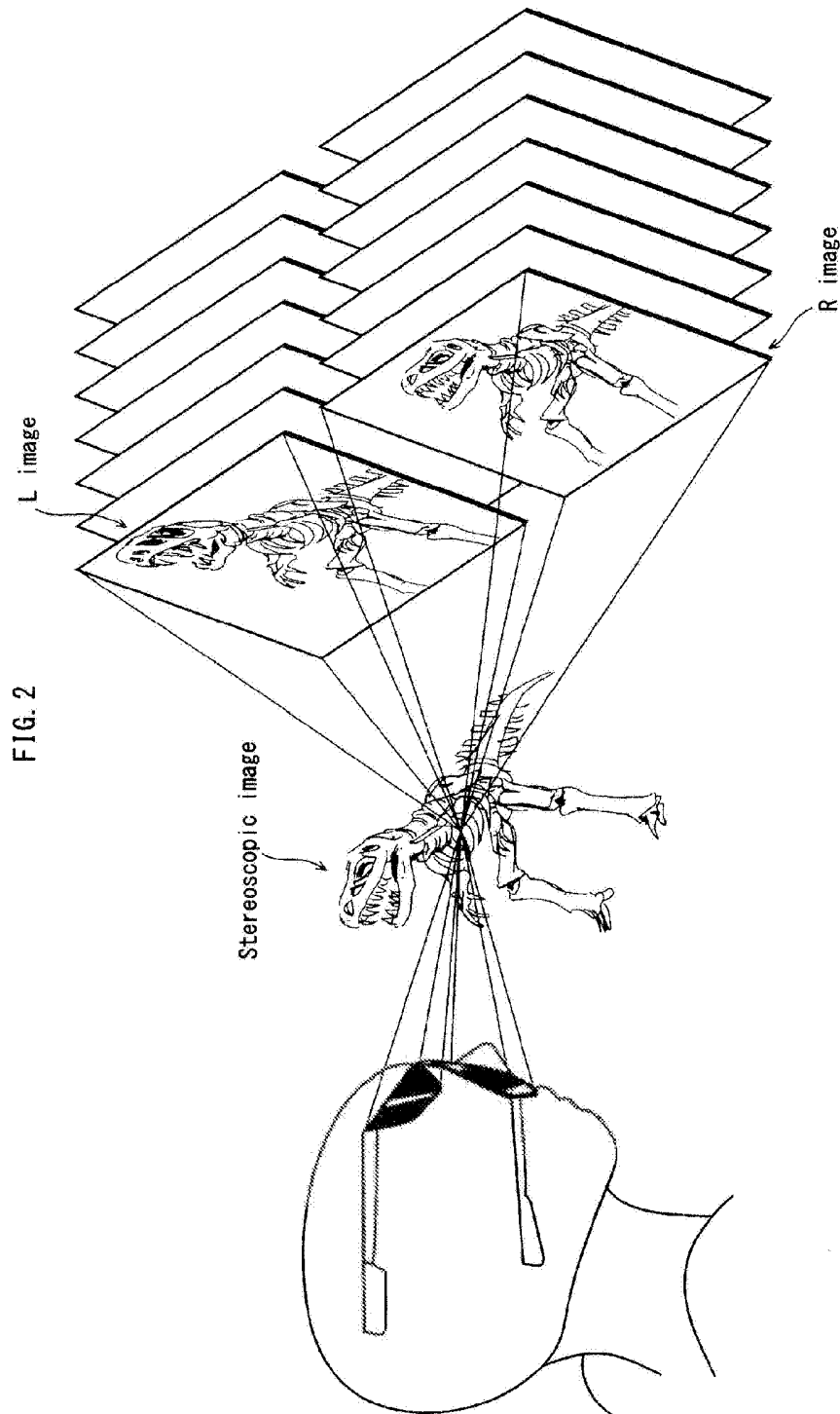
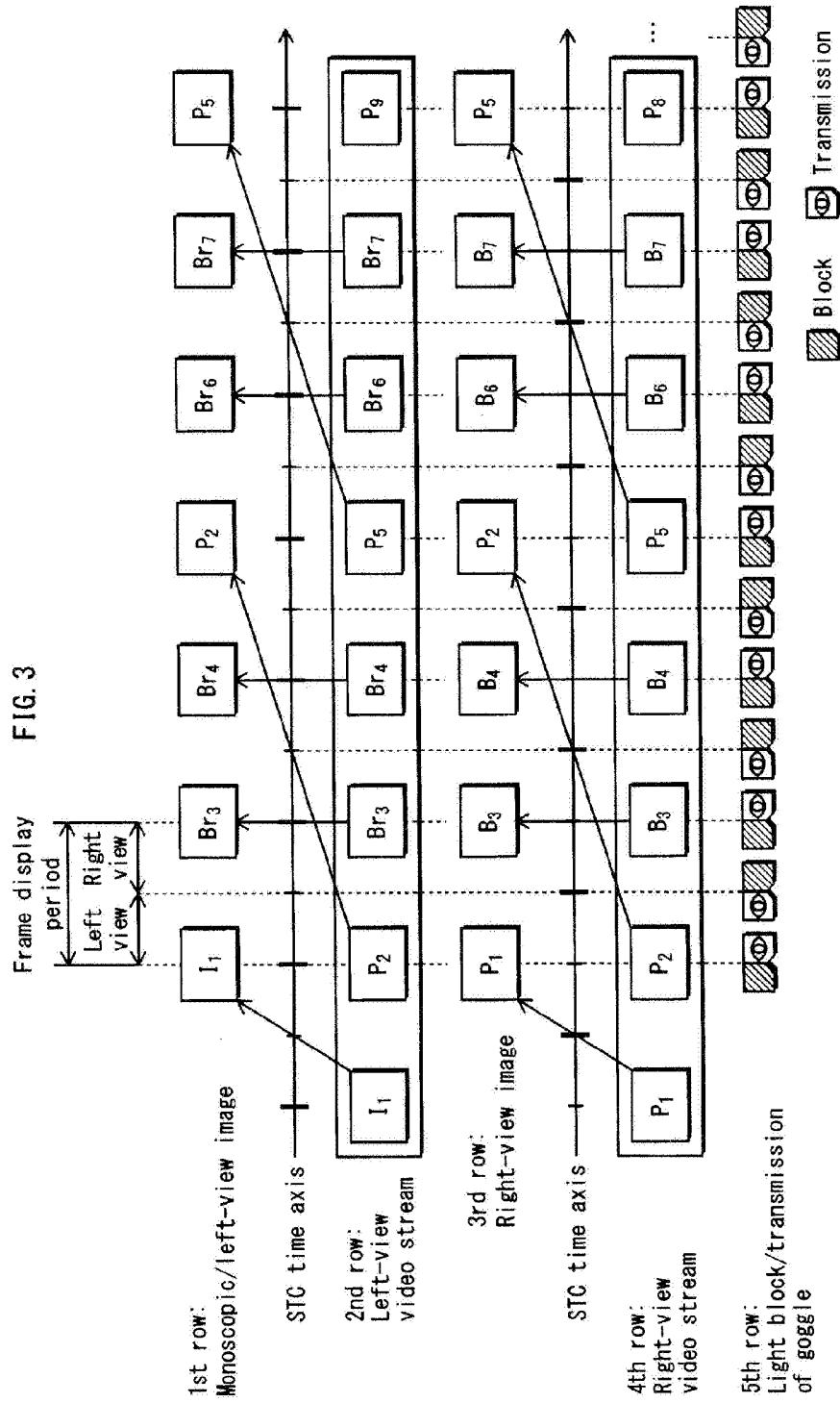


FIG. 3



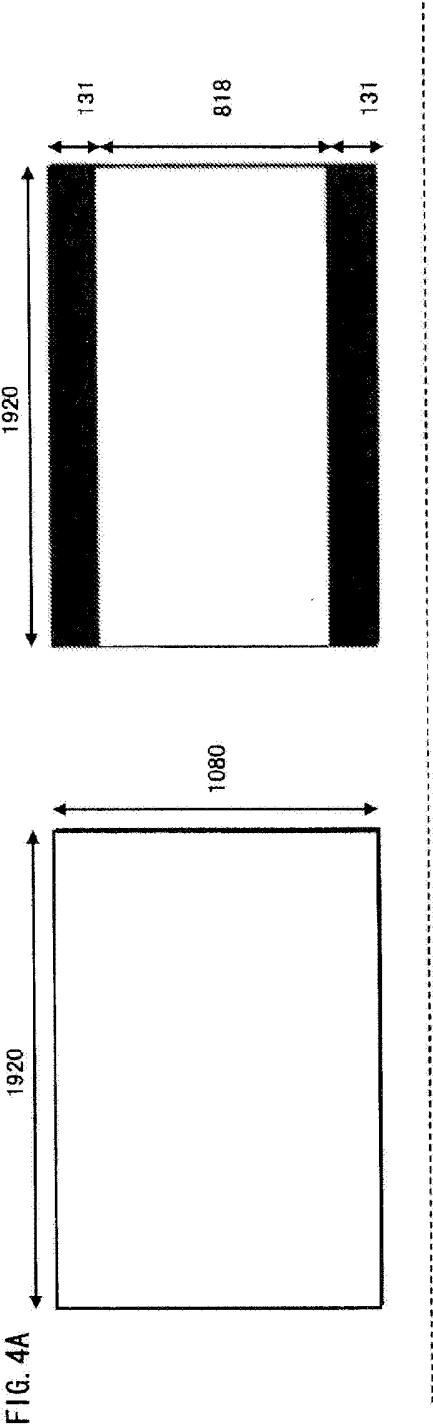
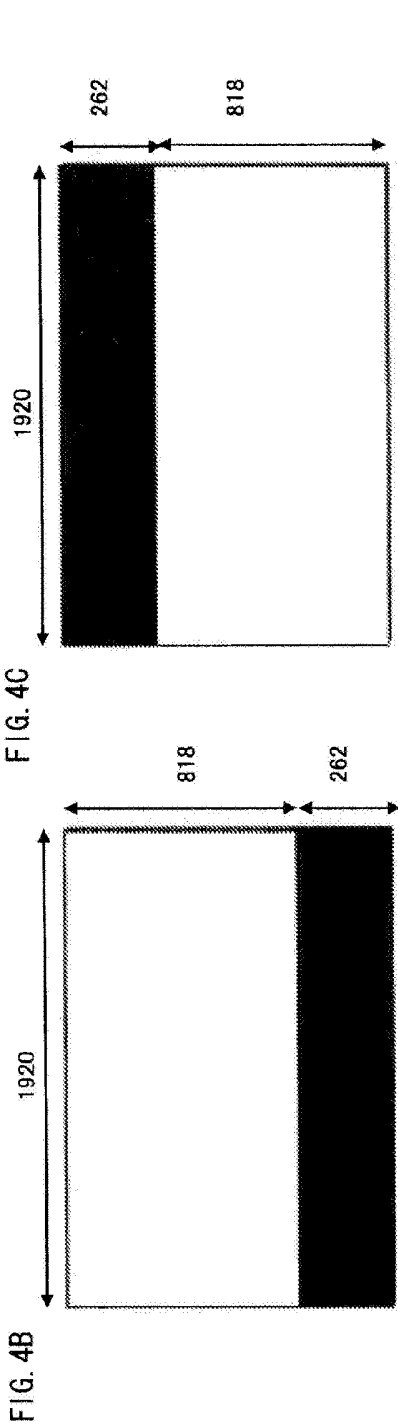


FIG. 5A

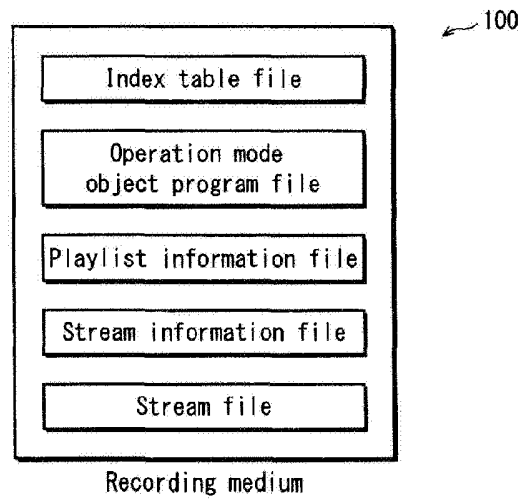


FIG. 5B

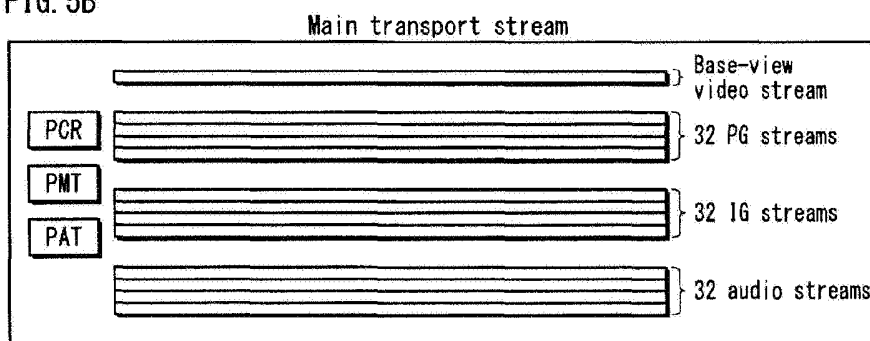


FIG. 5C

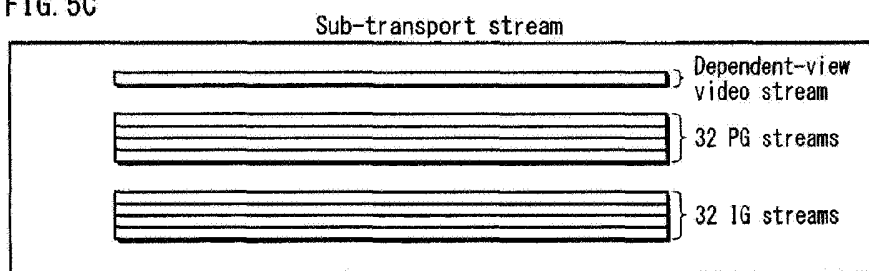


FIG. 6A Main transport stream

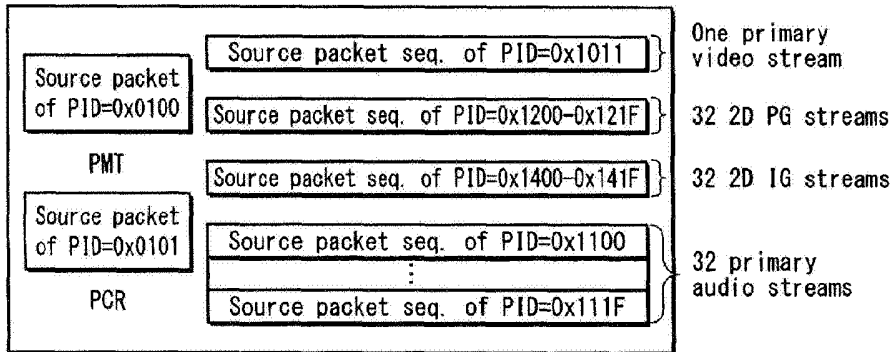


FIG. 6B Sub-transport stream

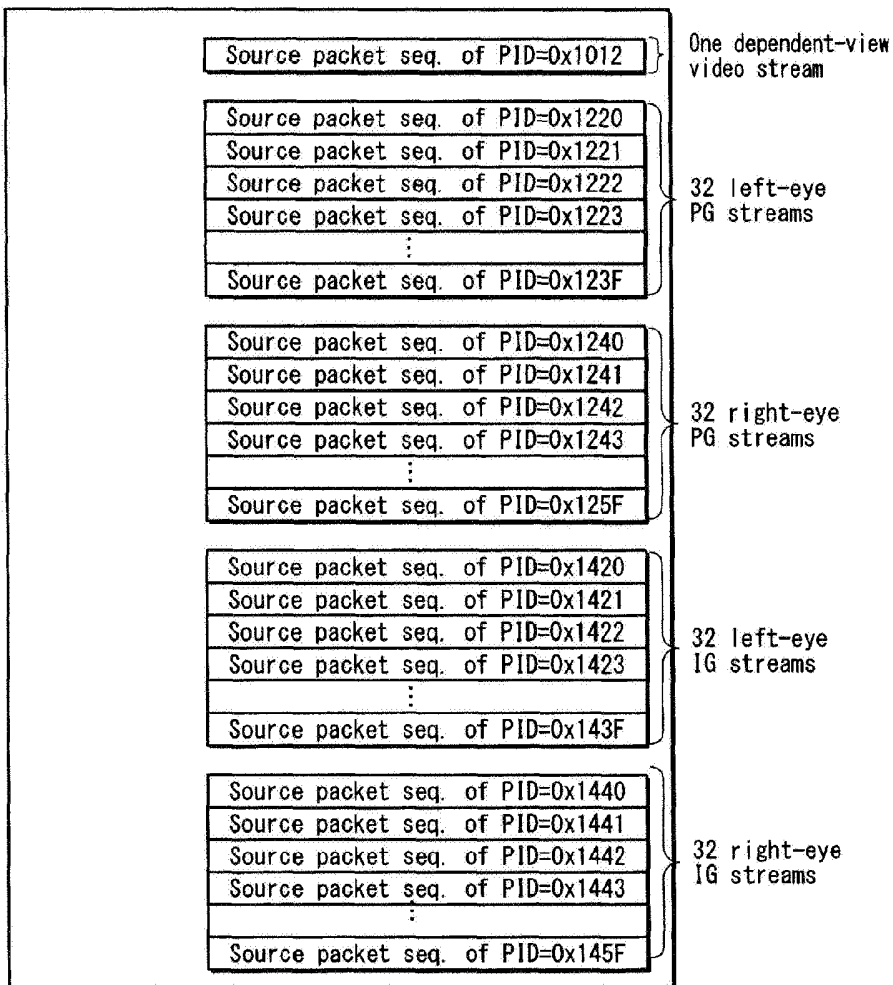


FIG. 7A

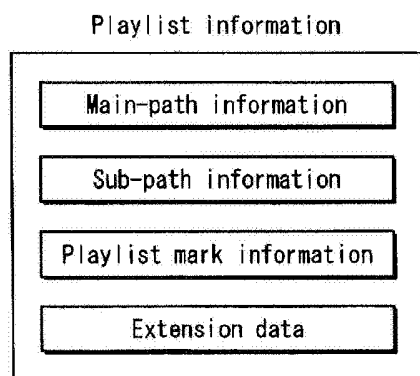


FIG. 7B

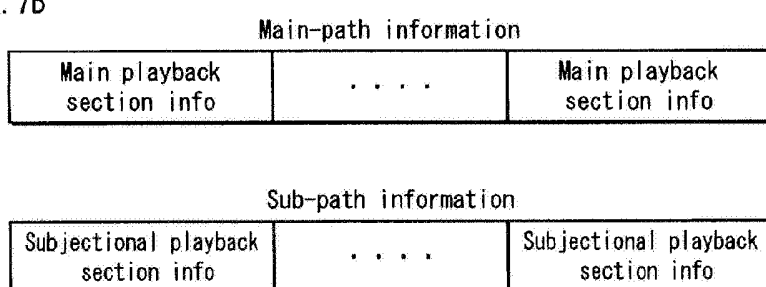


FIG. 7C

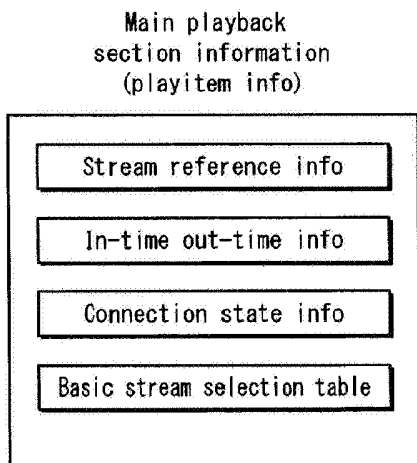


FIG. 7D

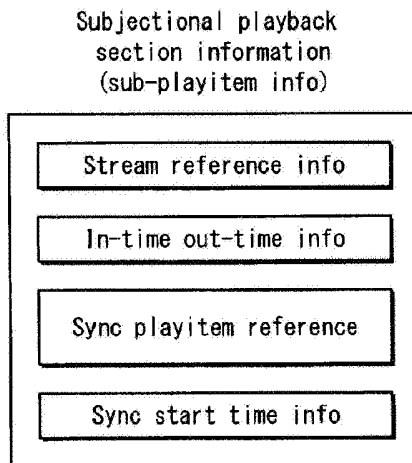


FIG. 8A

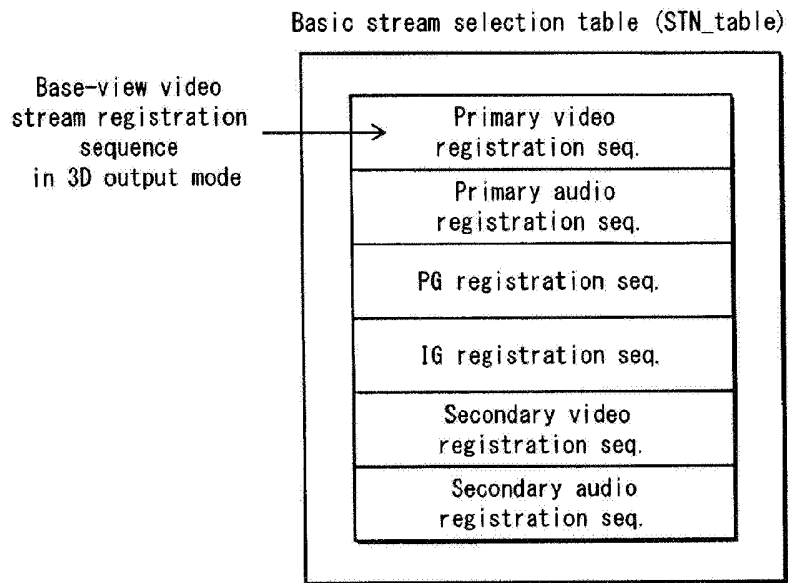


FIG. 8B

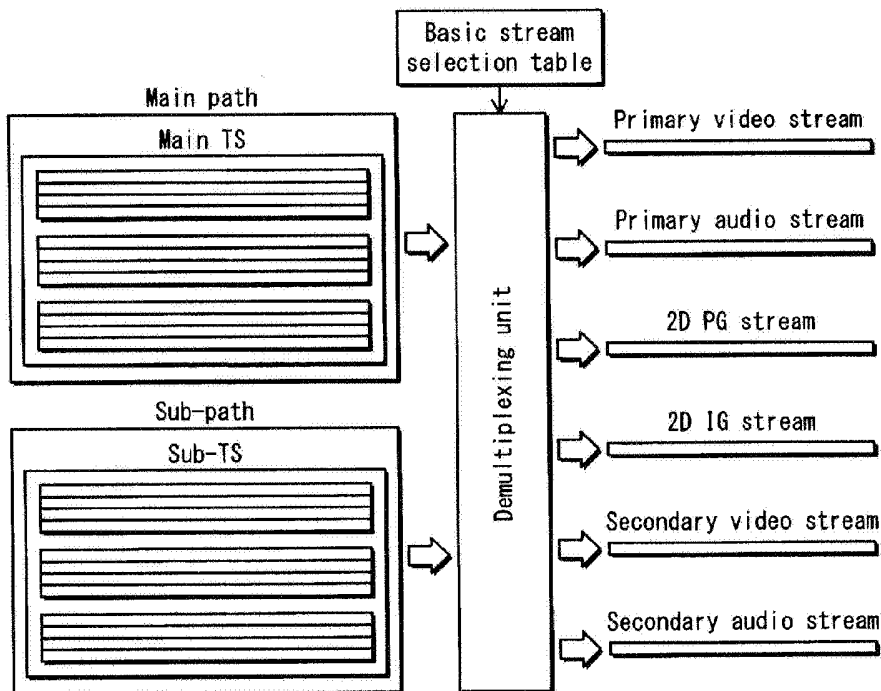


FIG. 9
Extension data

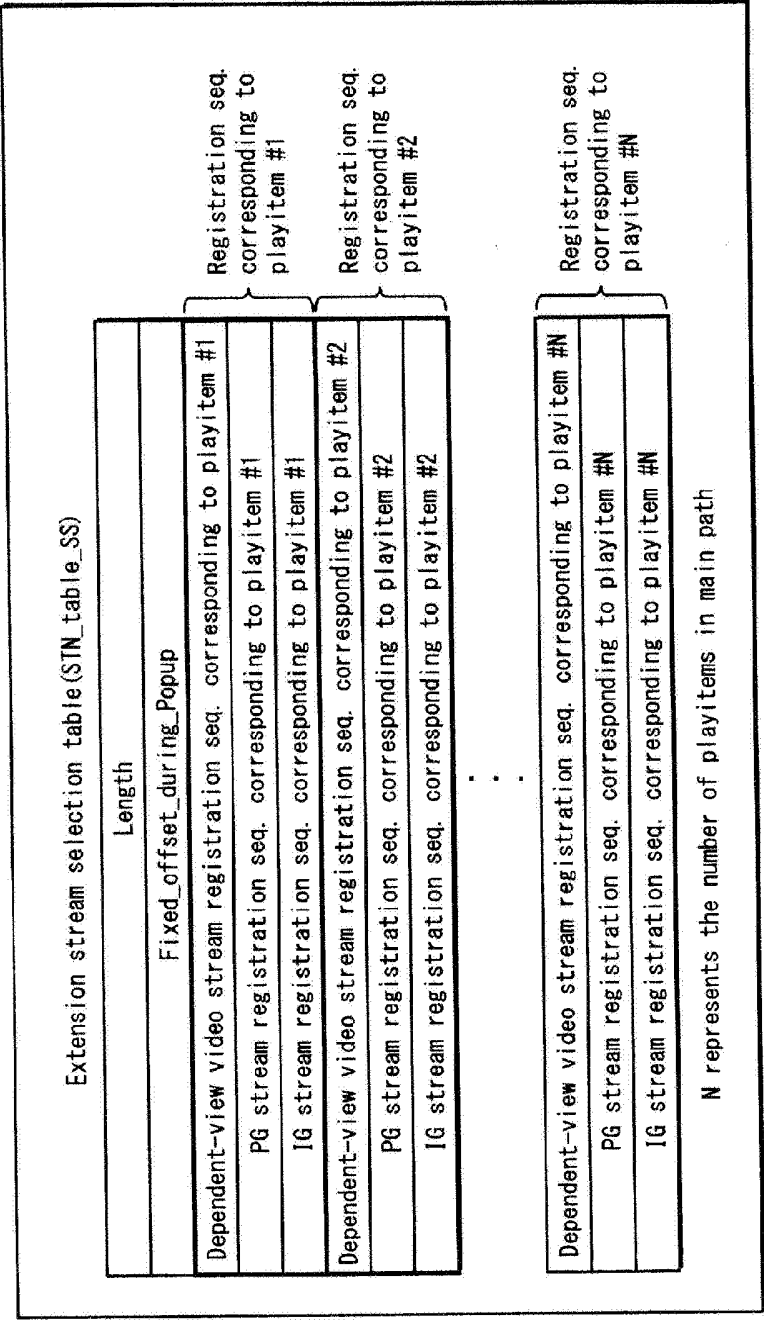


FIG. 10A

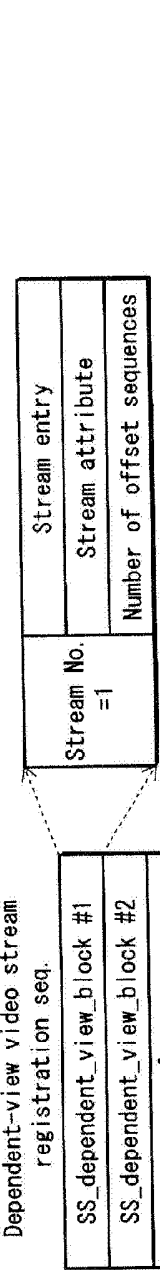


FIG. 10B

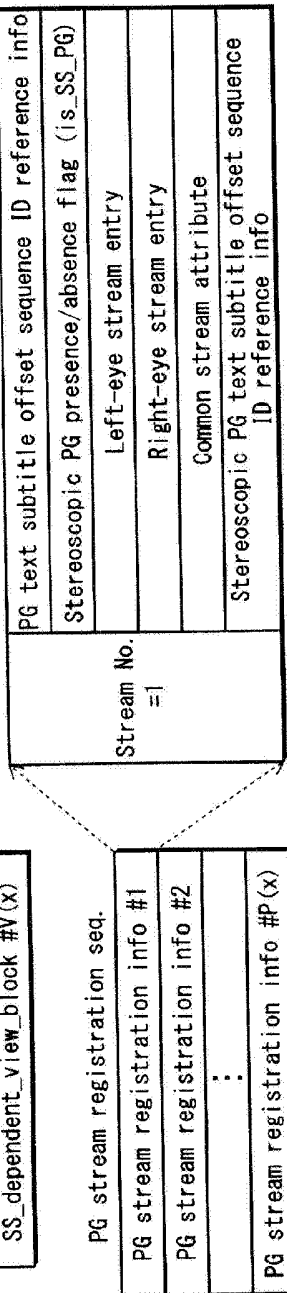


FIG. 10C

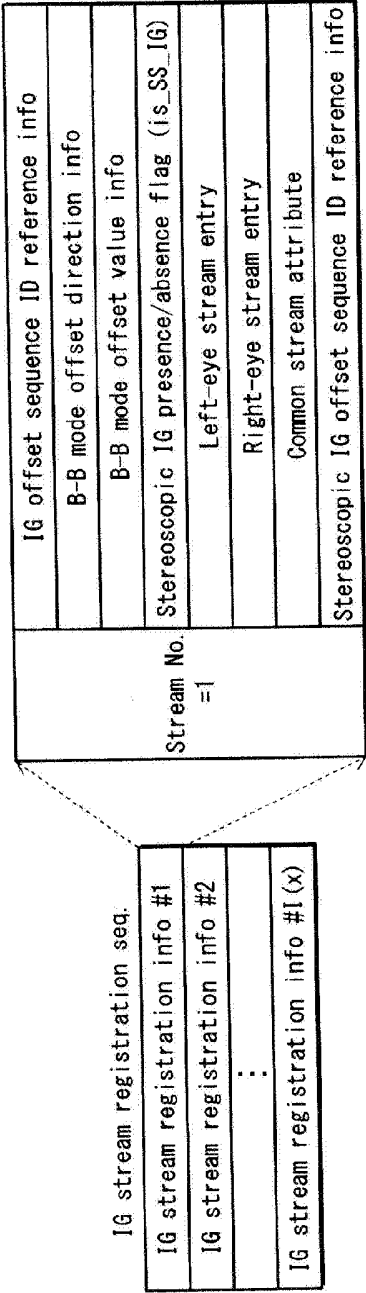


FIG. 11

Case where Video_shift_mode is "Keep":

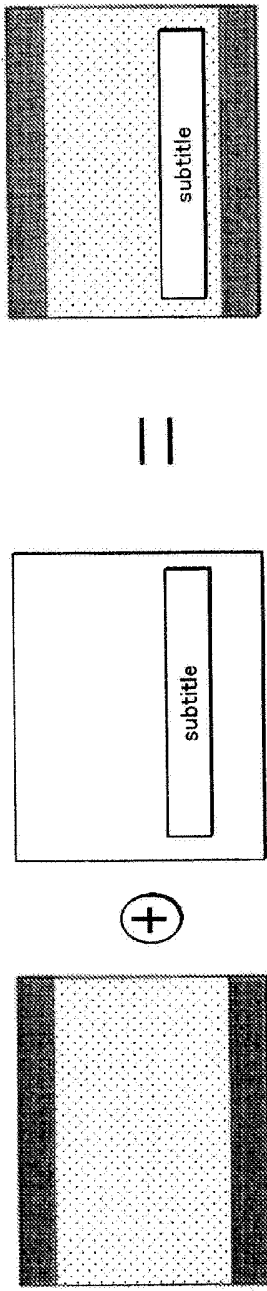


FIG. 12A

Case where Video_shift_mode is "Up":

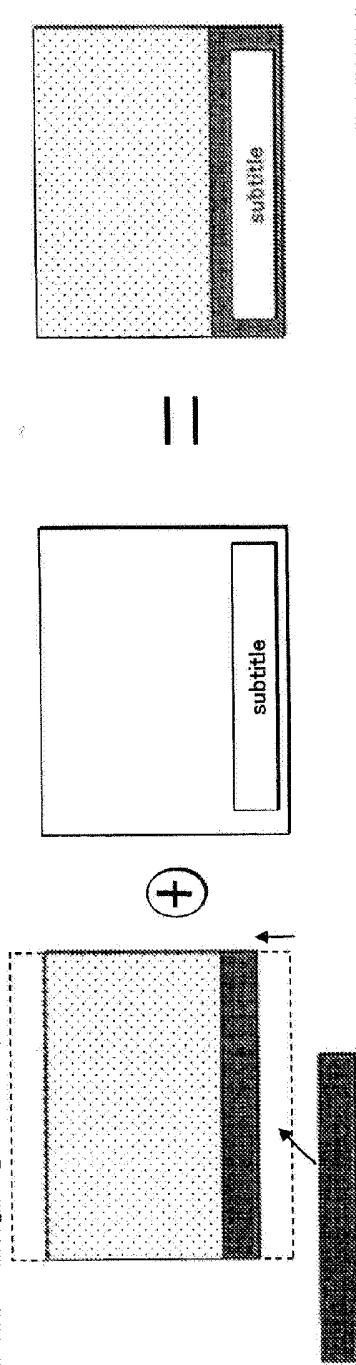


FIG. 12B

Case where Video_shift_mode is "Down":

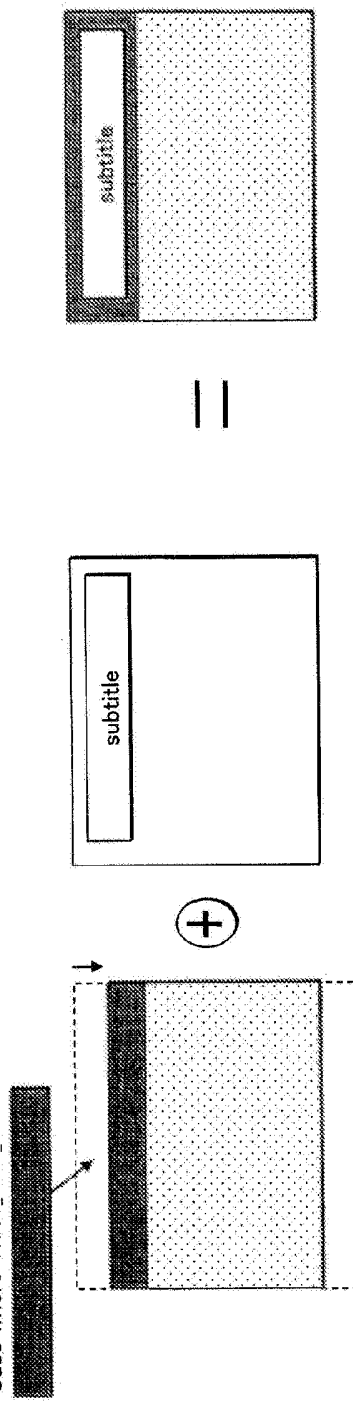


FIG. 13

Subtitle entry number	video_shift_mode
1	Keep
2	Keep
3	Keep
4	Up
5	Up
6	Down
7	Down
8	Down
9	Down

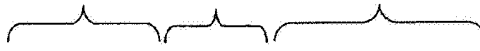


FIG. 14

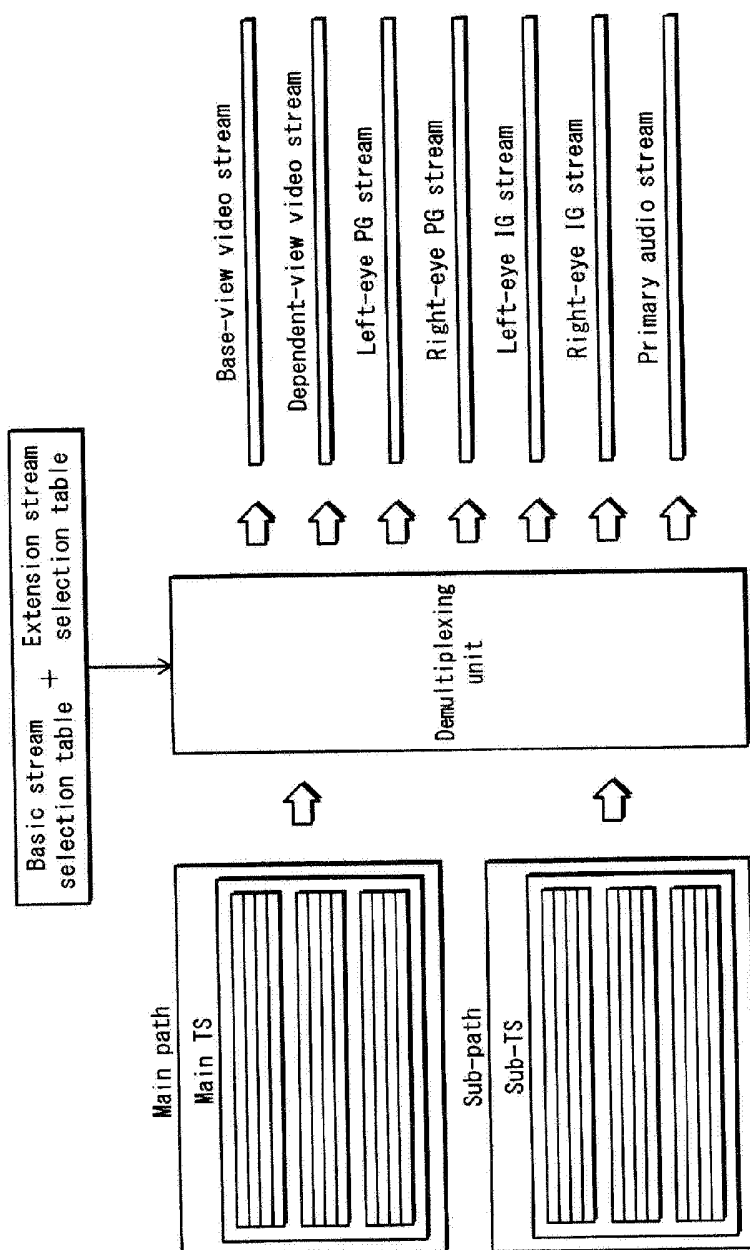


FIG. 15

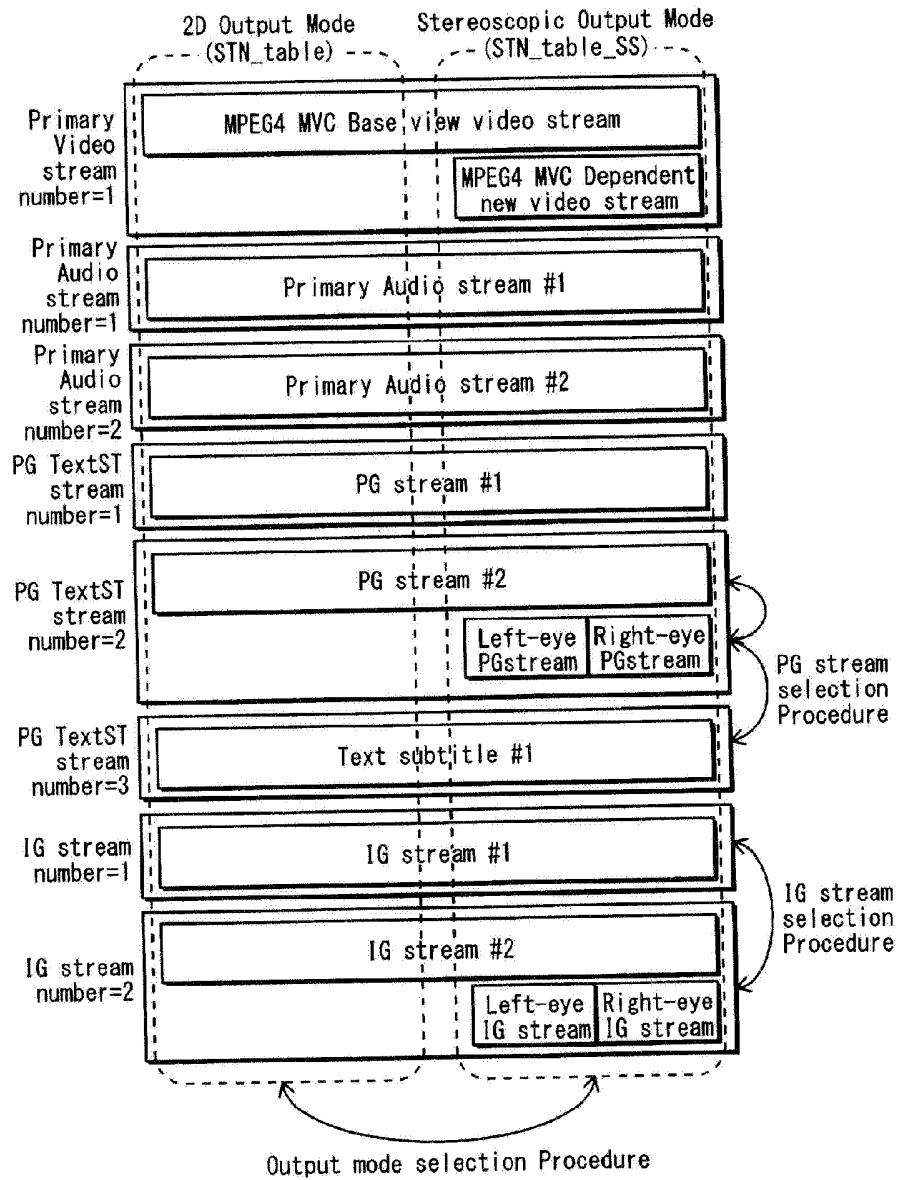


FIG. 16

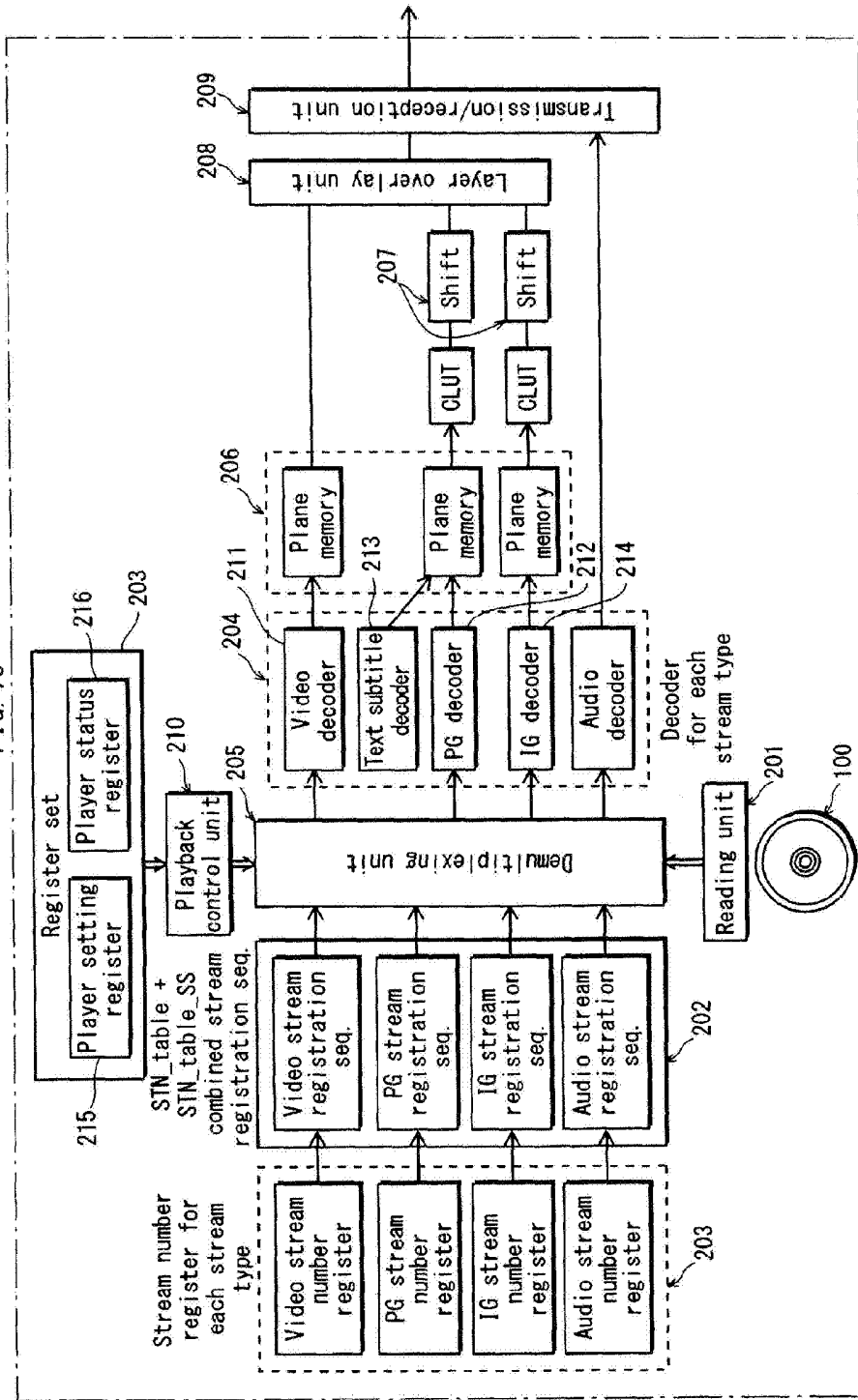


FIG. 17A

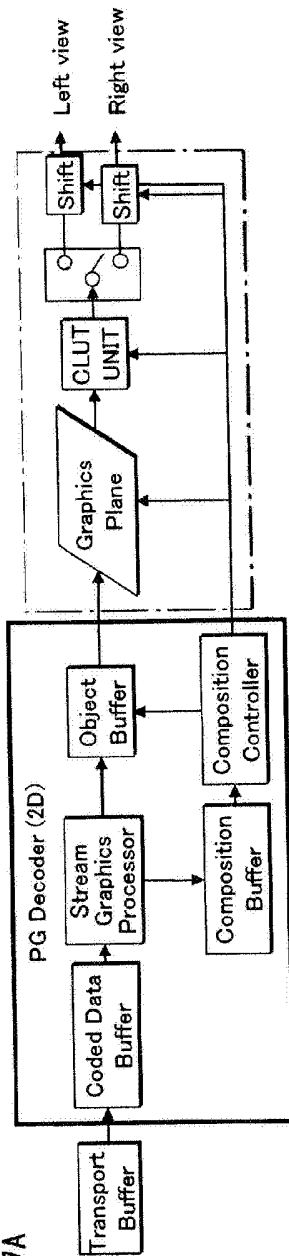


FIG. 17B

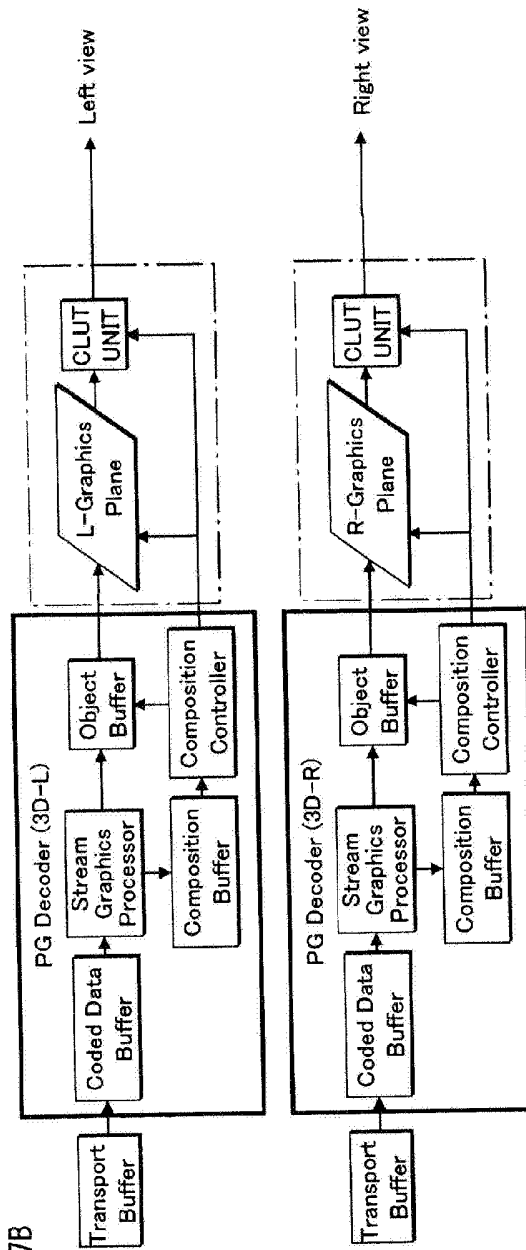


FIG. 18A

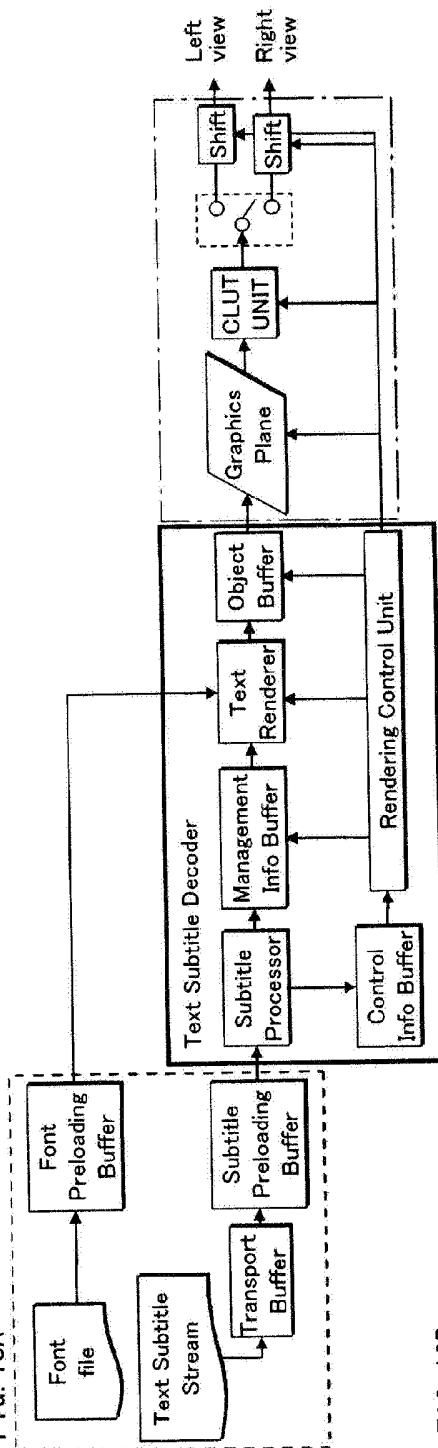
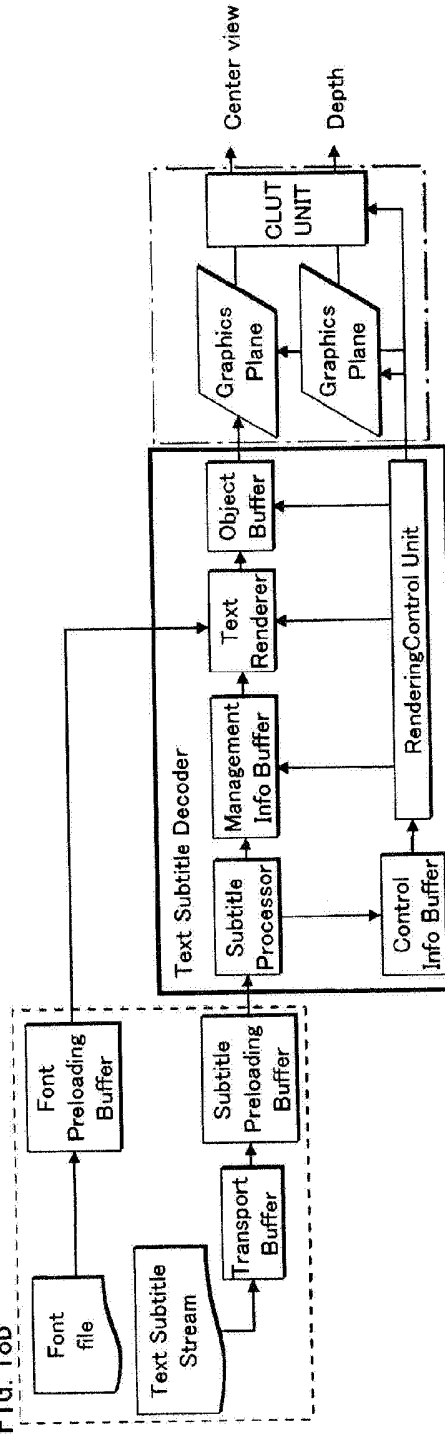


FIG. 18B



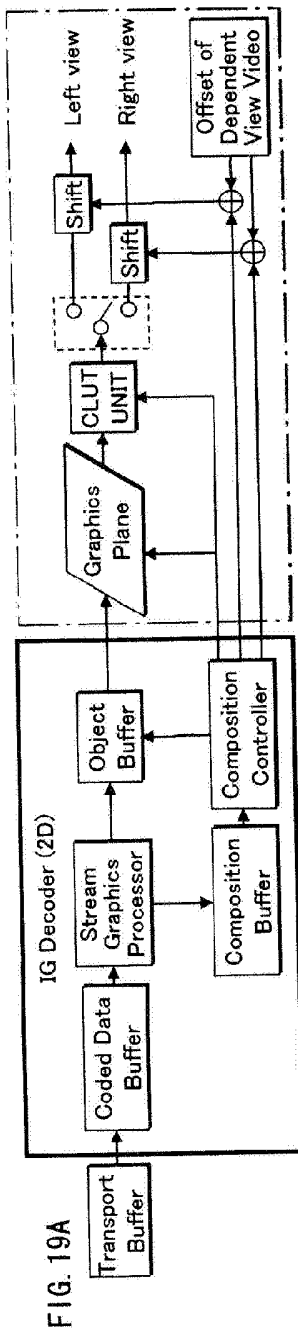
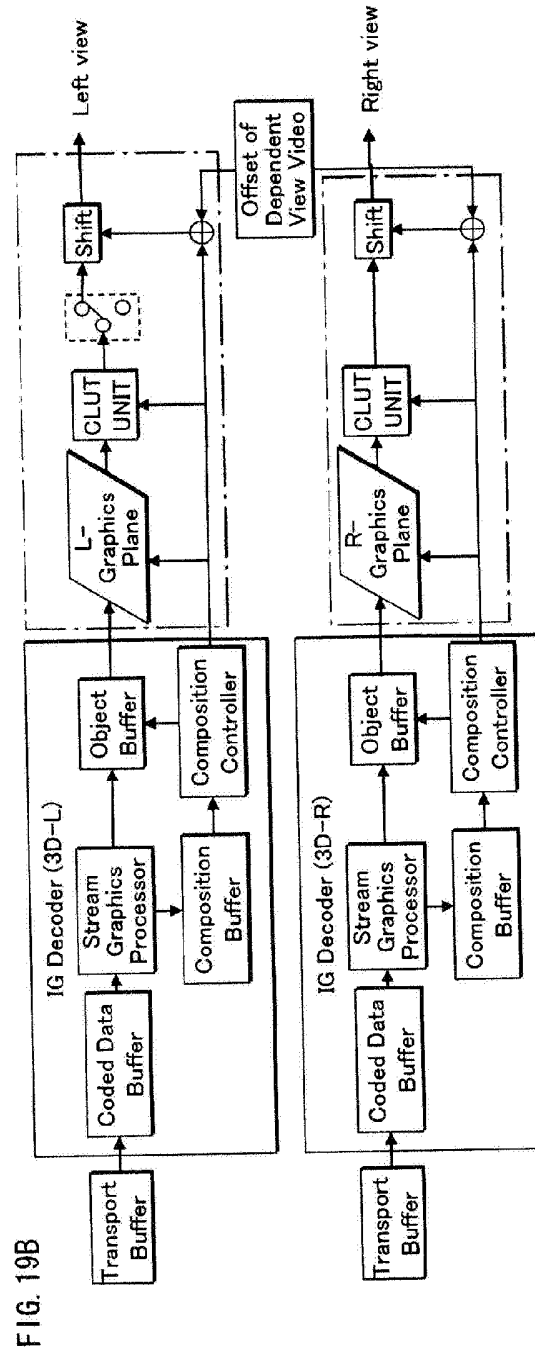


FIG. 20

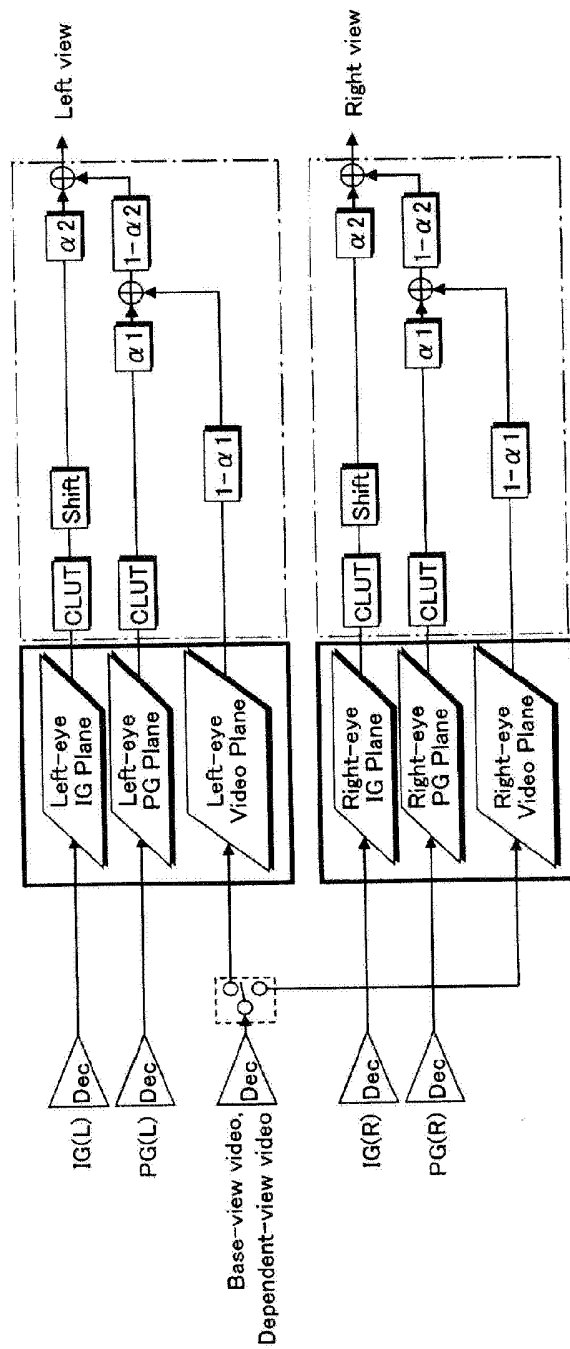


FIG. 22

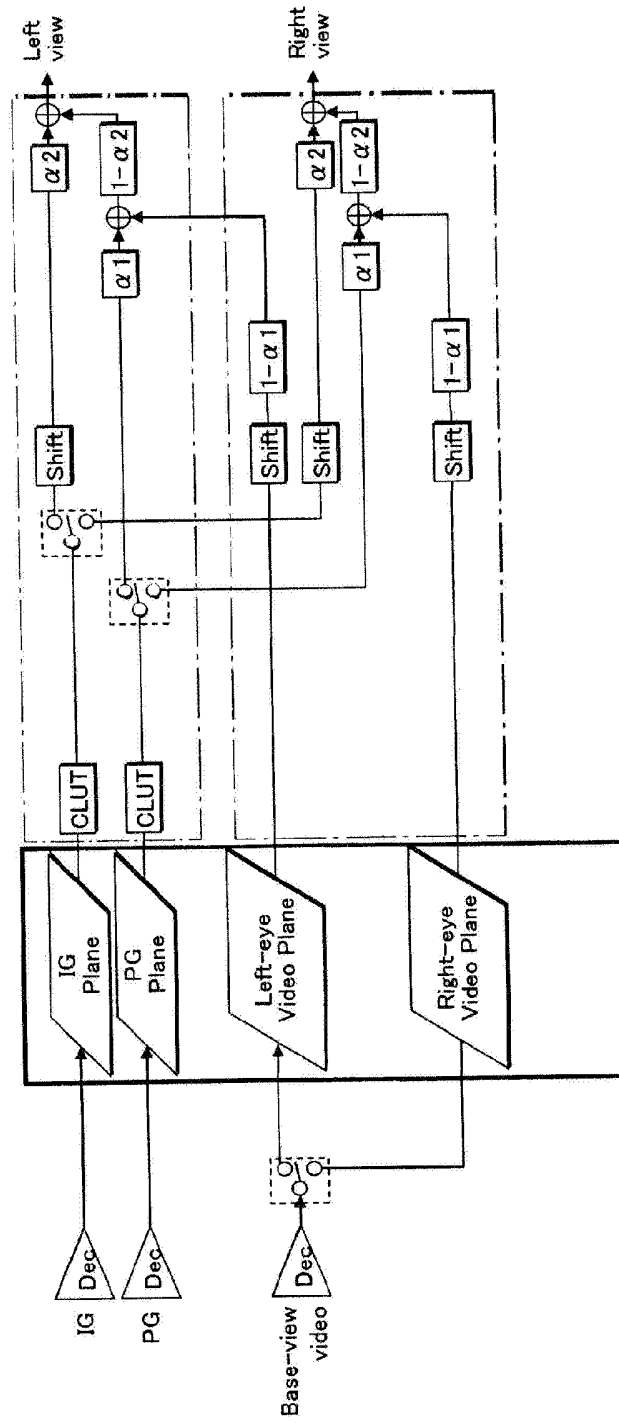


FIG. 23

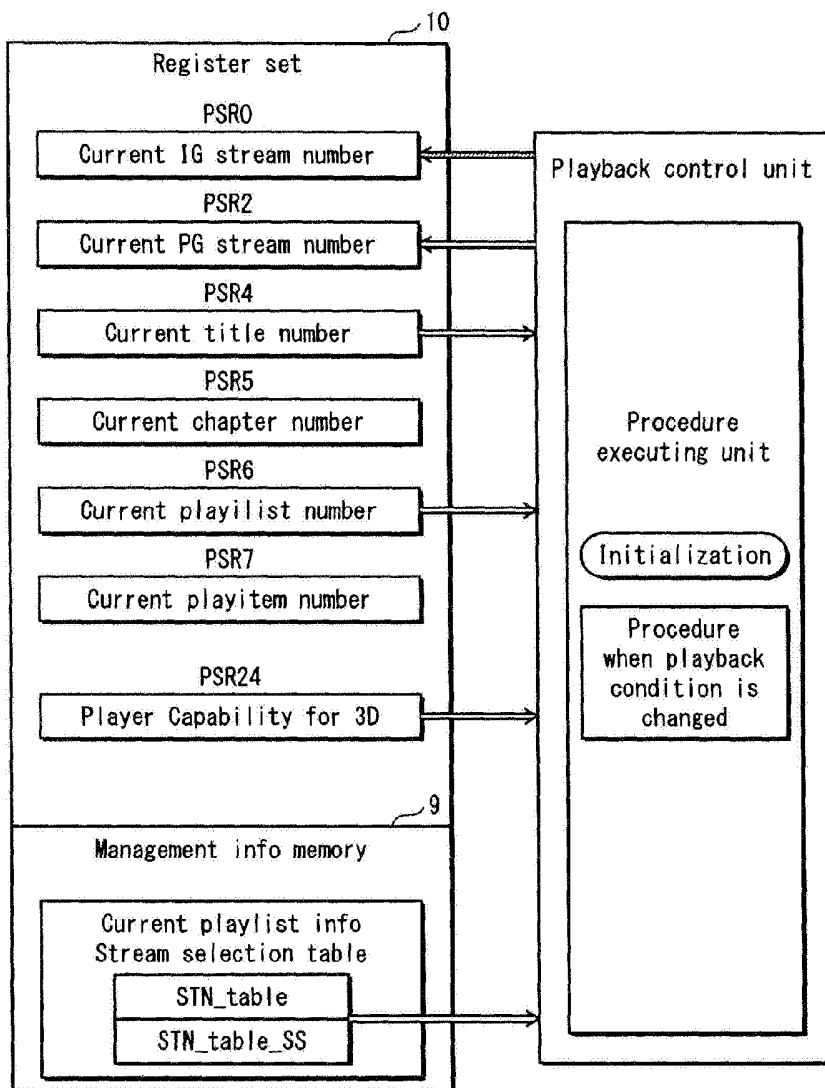


FIG. 24 PSR24

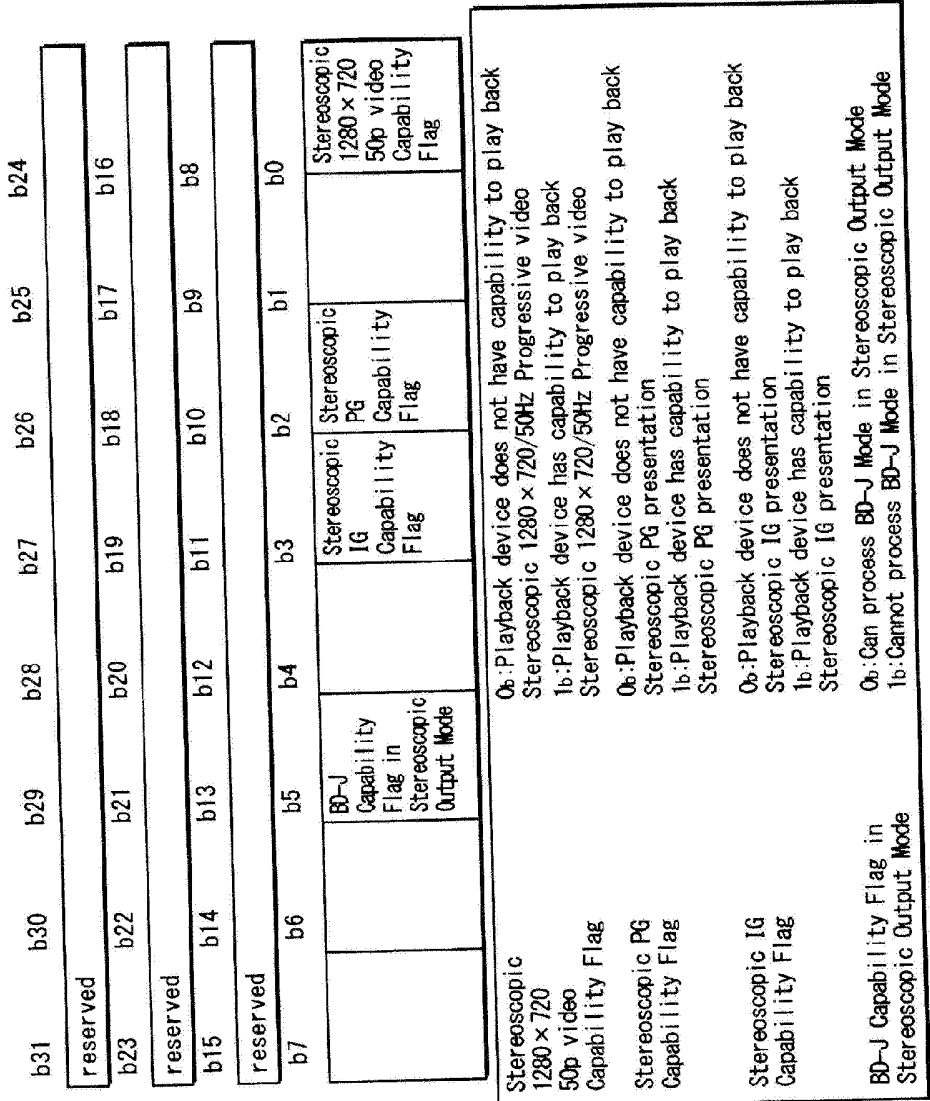
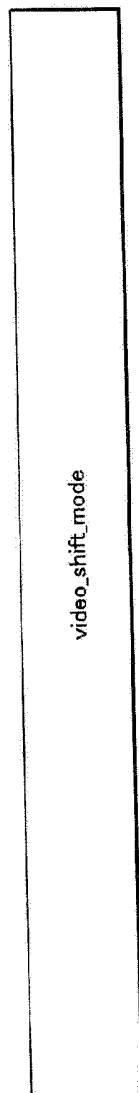


FIG. 25A

PSR 32



0b : Keep, 01b:Up, 11b:Down

FIG. 25B ①

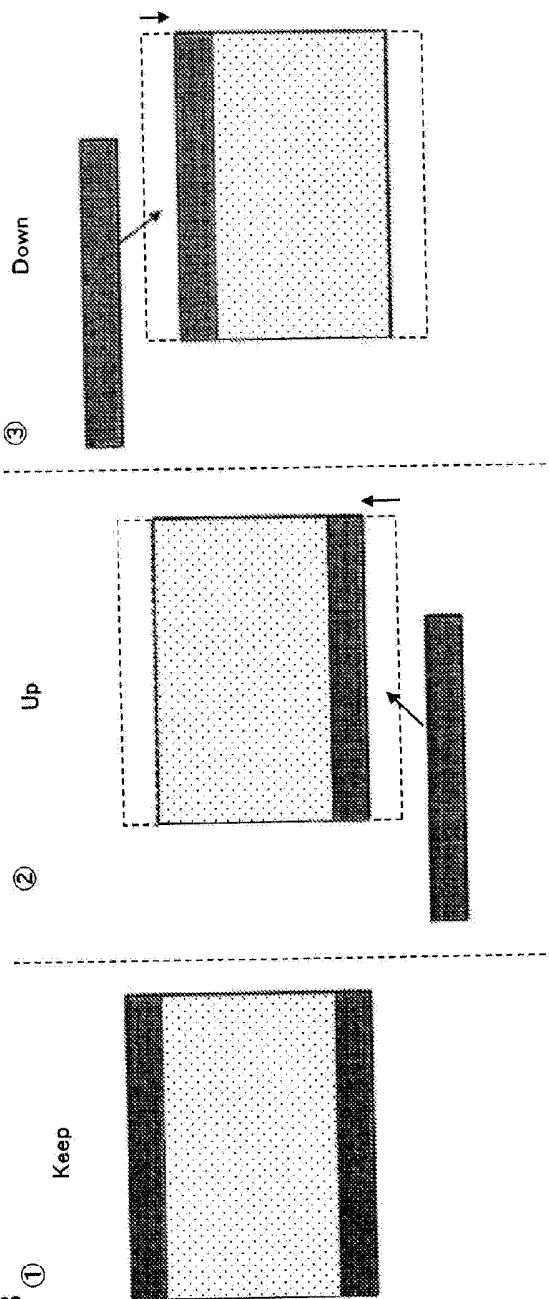


FIG. 26

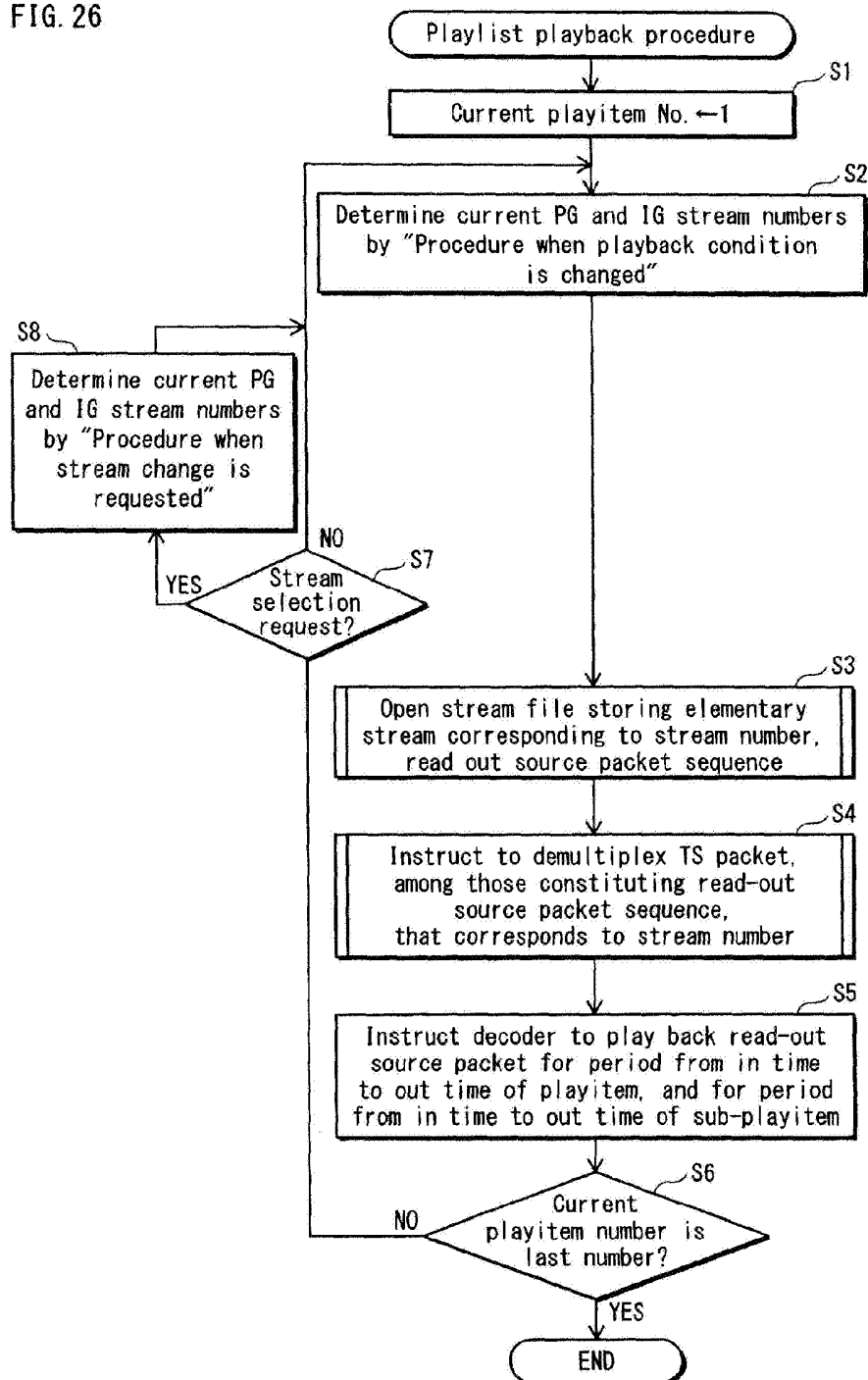


FIG. 27

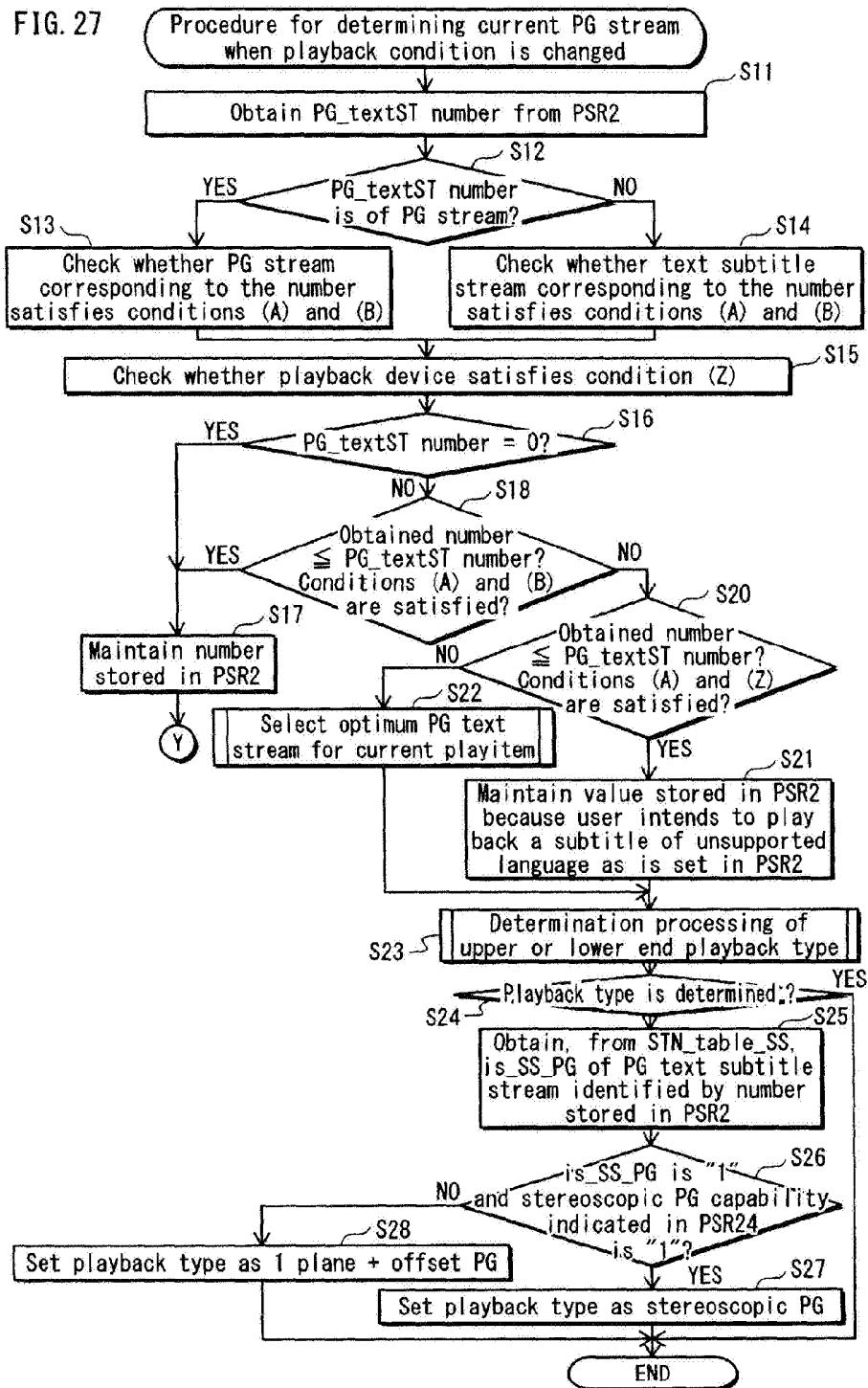


FIG. 28

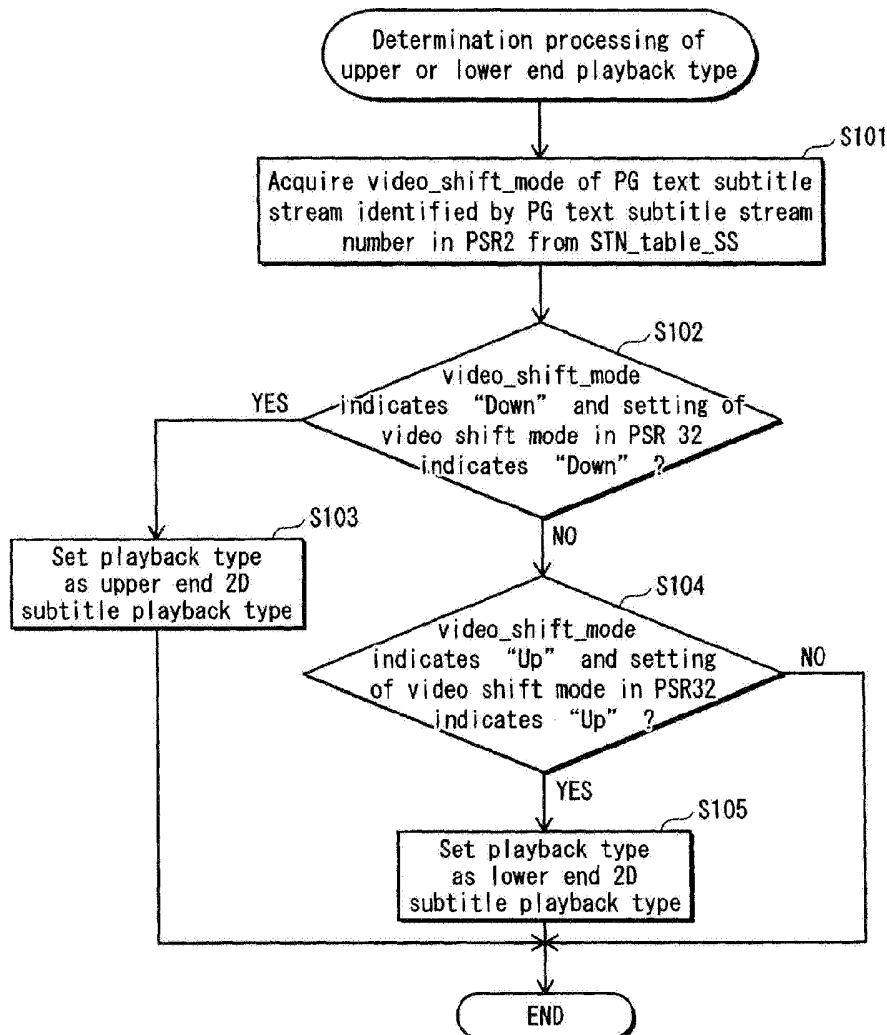


FIG. 29

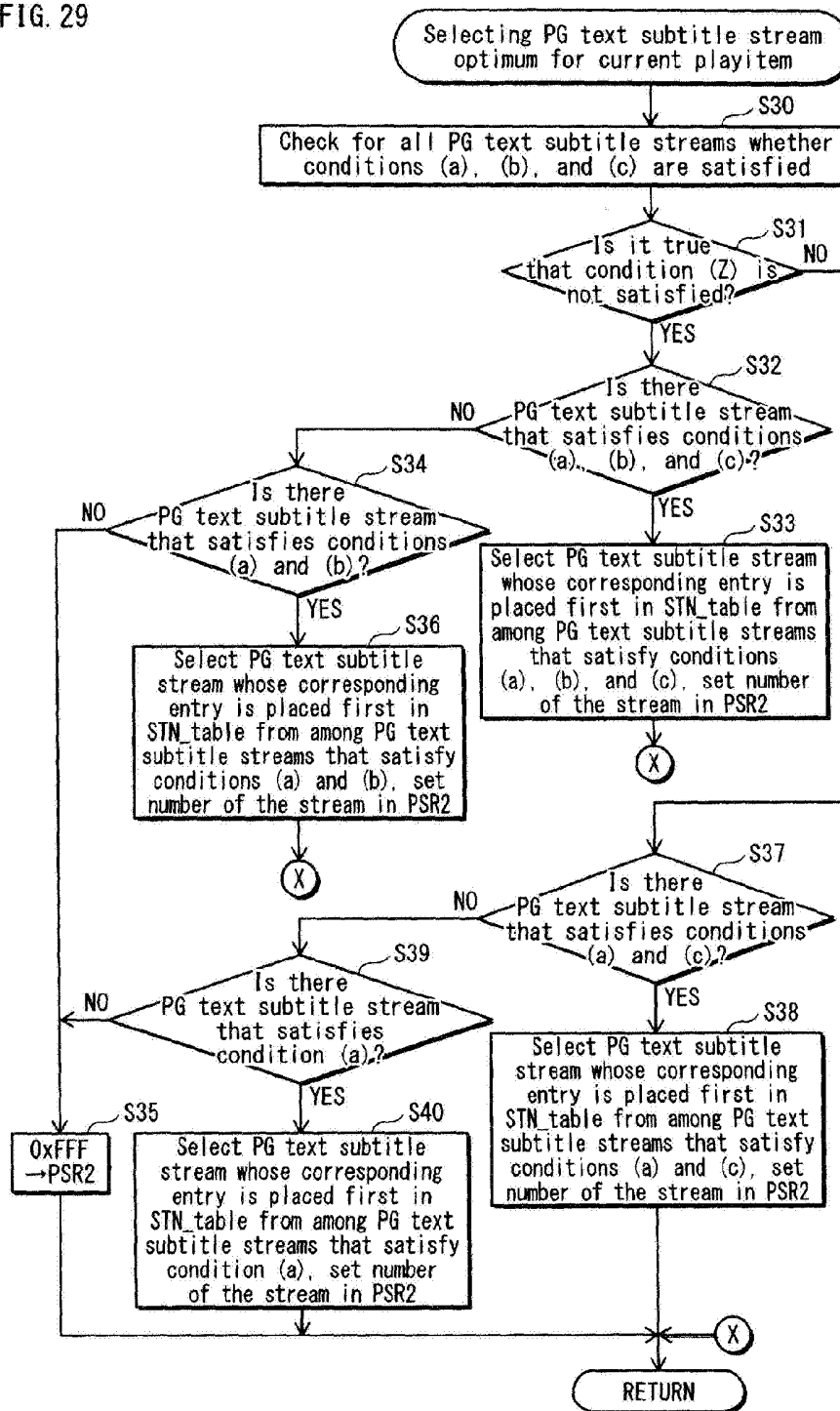


FIG. 30

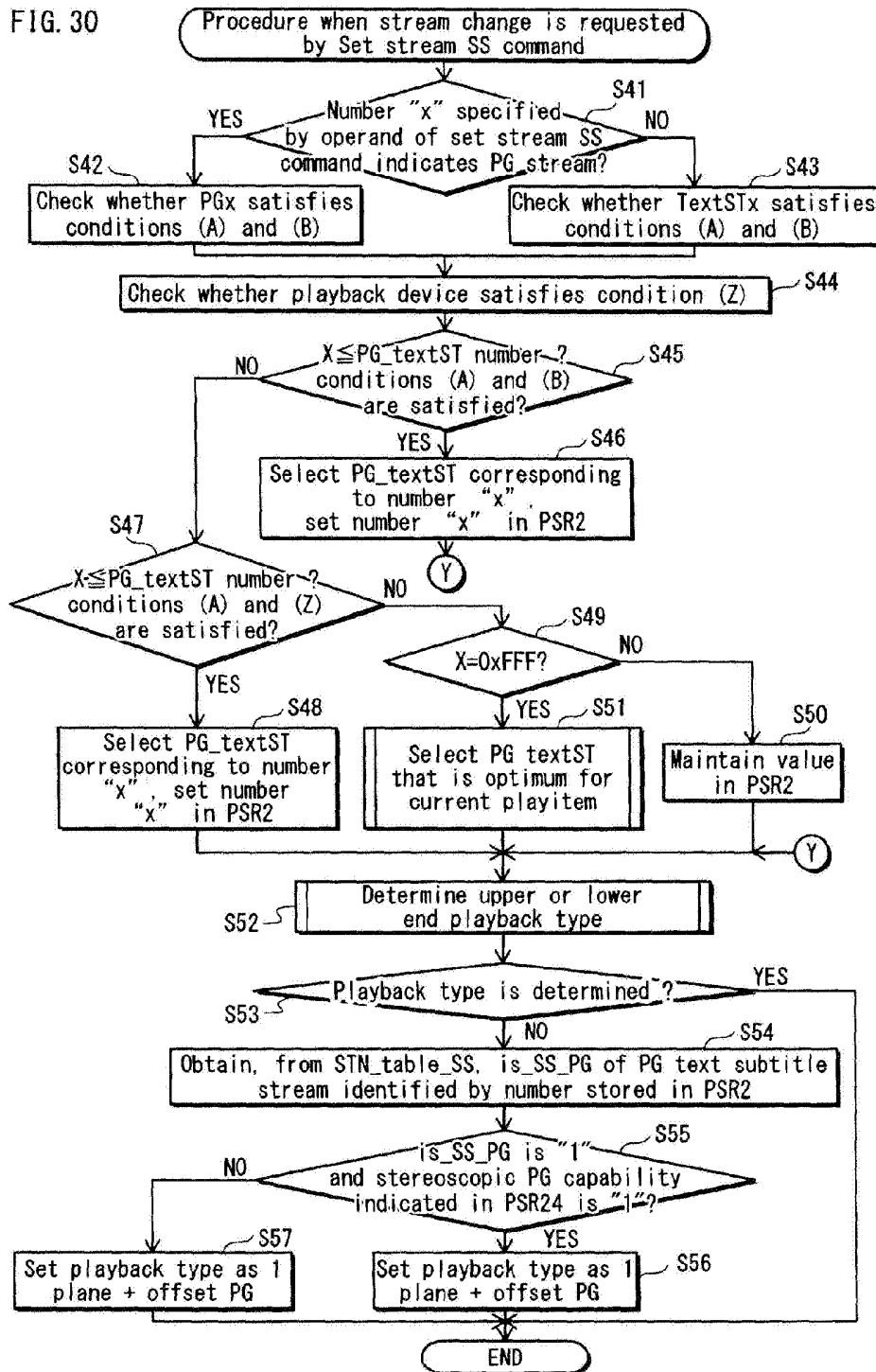


FIG. 31

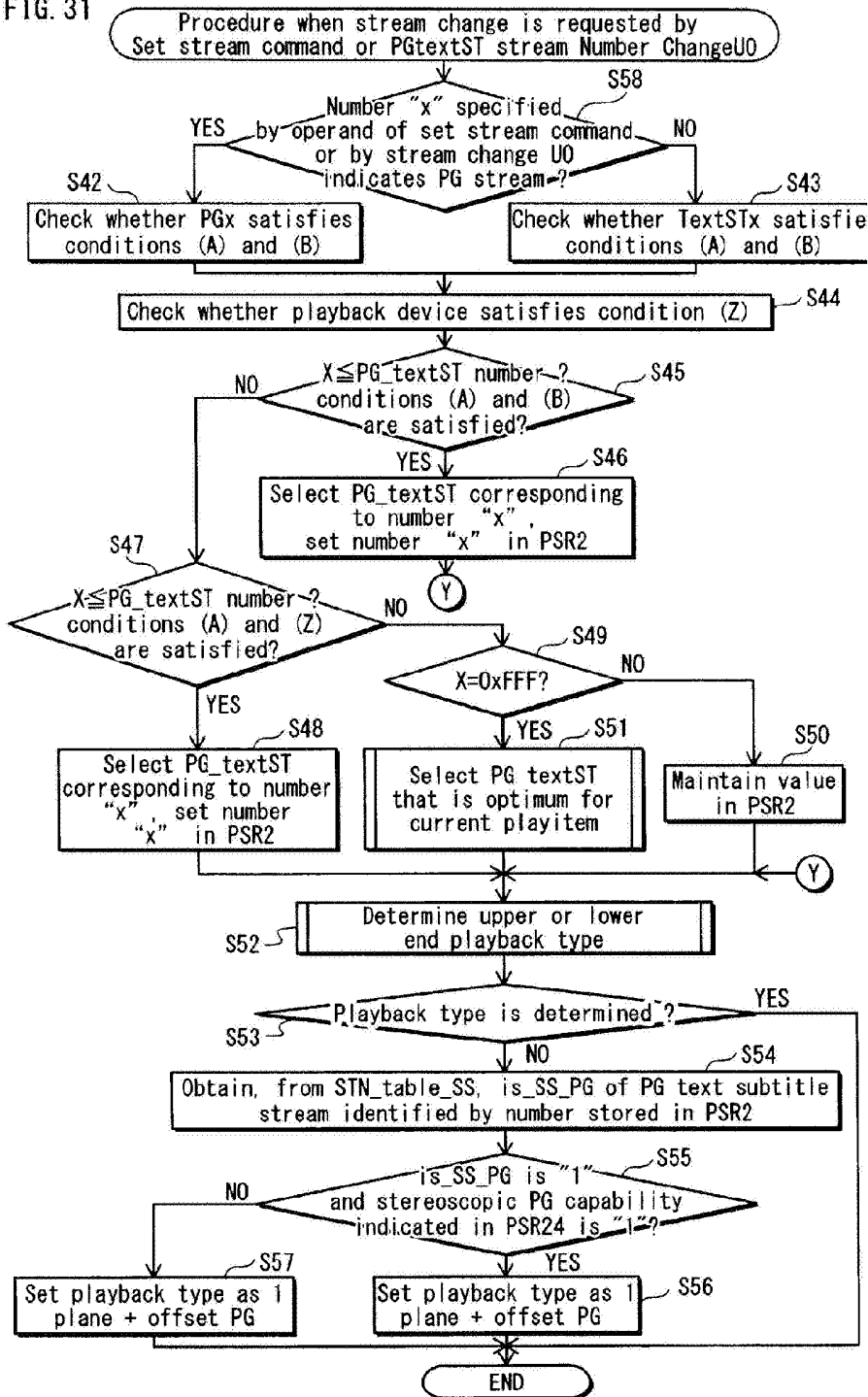


FIG. 32A

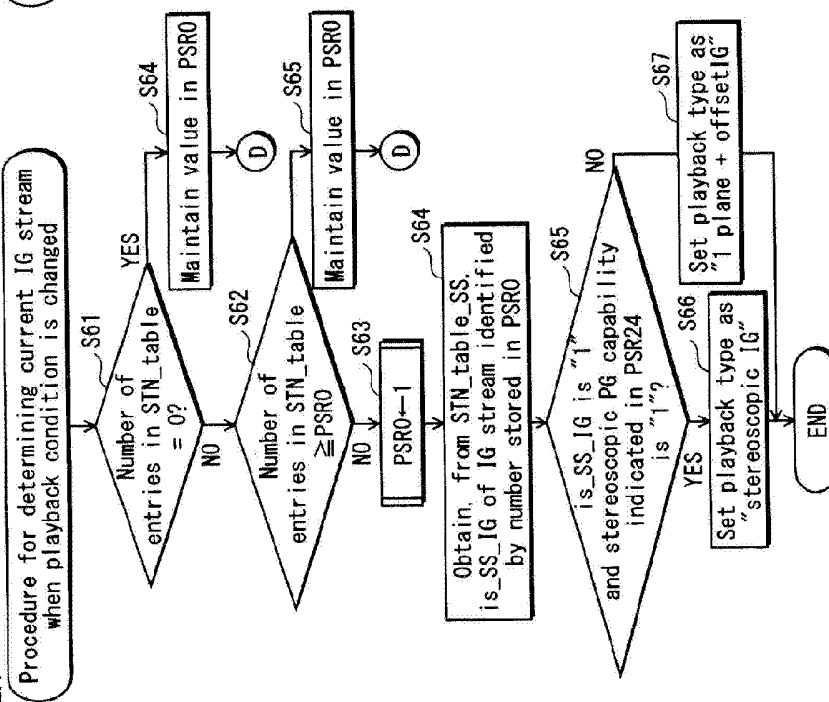


FIG. 32B

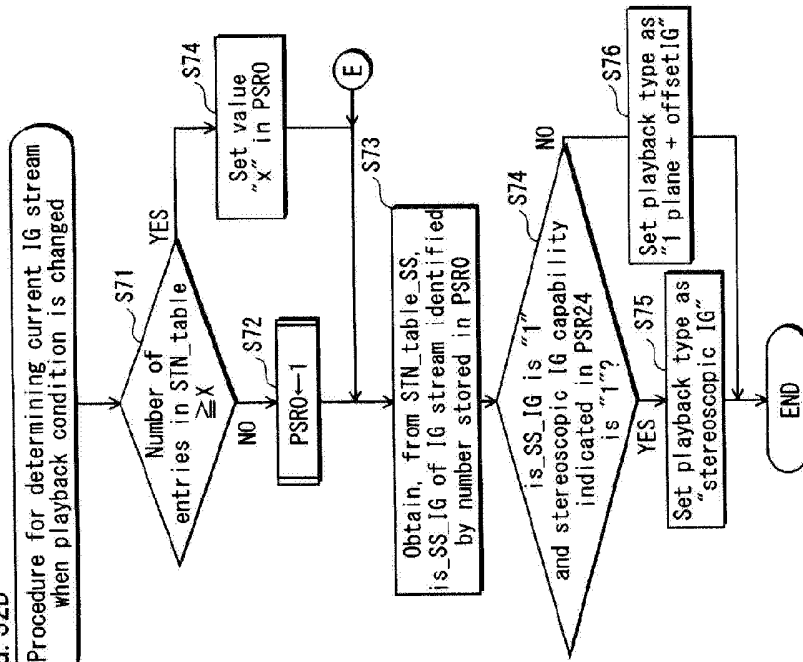
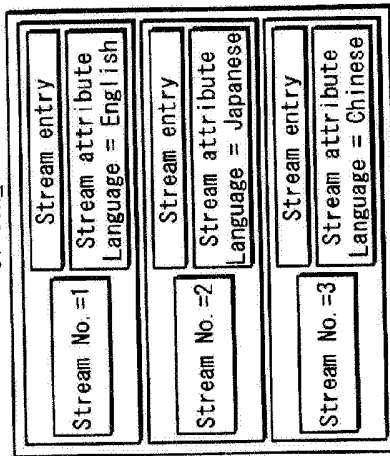


FIG. 33A Stream registration seq. of STN_table



Stream registration seq. of STN_table SS

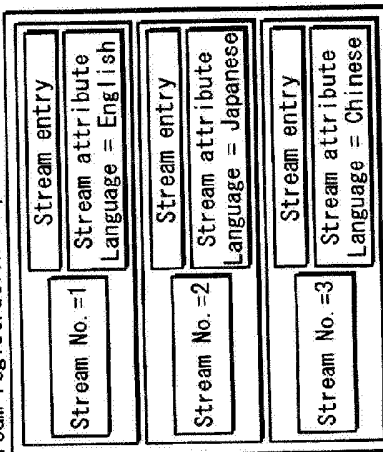


FIG. 33B

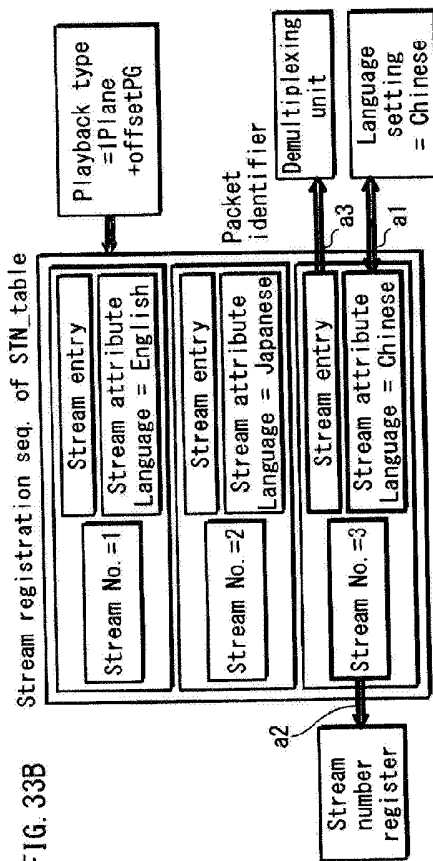


FIG. 33C

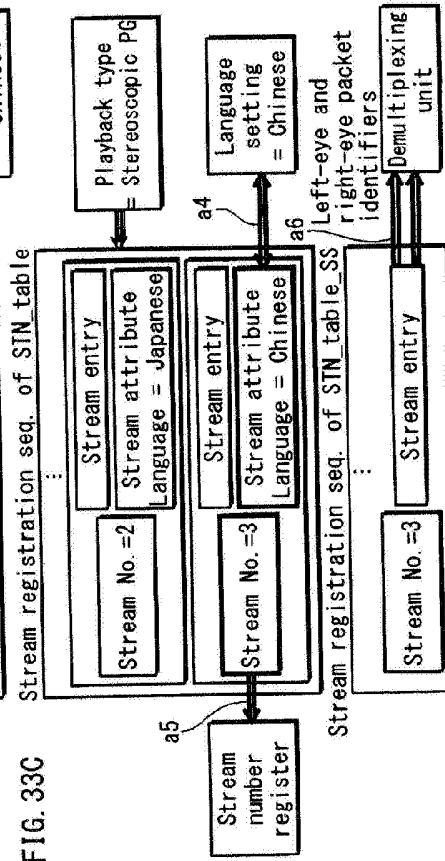
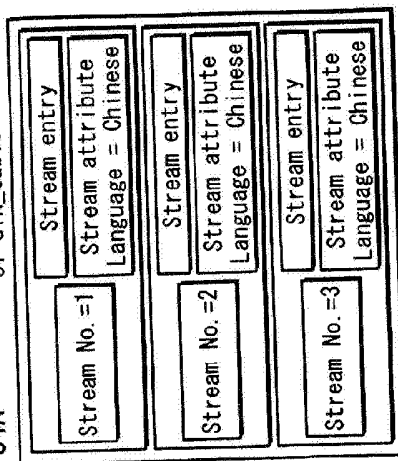


FIG. 34A
Stream registration seq.
of STN_table

Stream registration seq. of STN_table SS

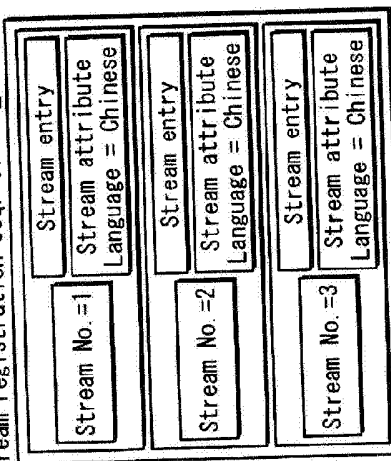


FIG. 34B

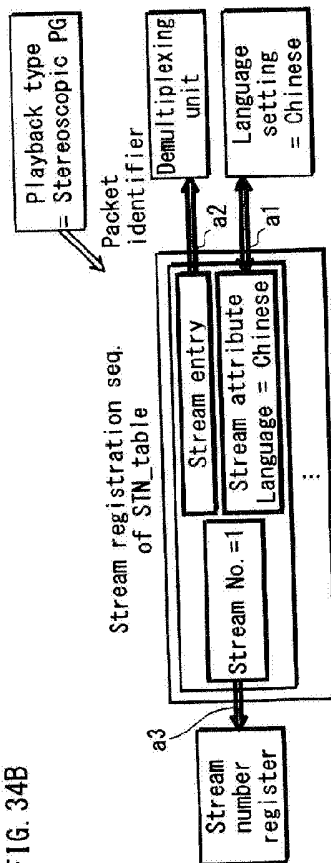


FIG. 34C

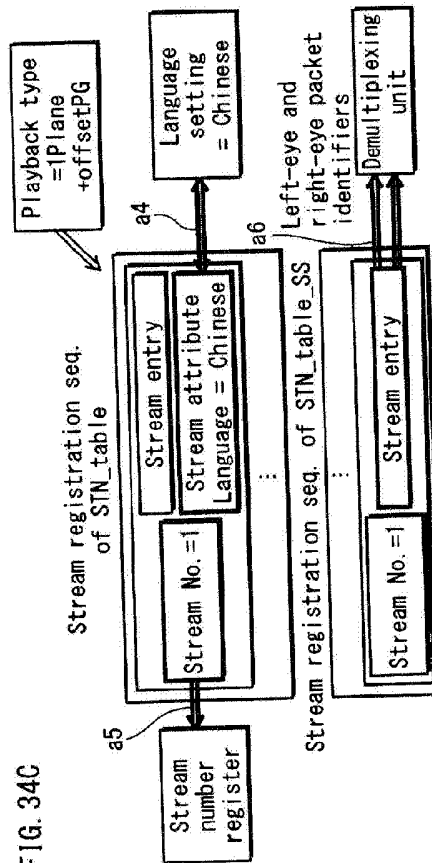


FIG. 35A

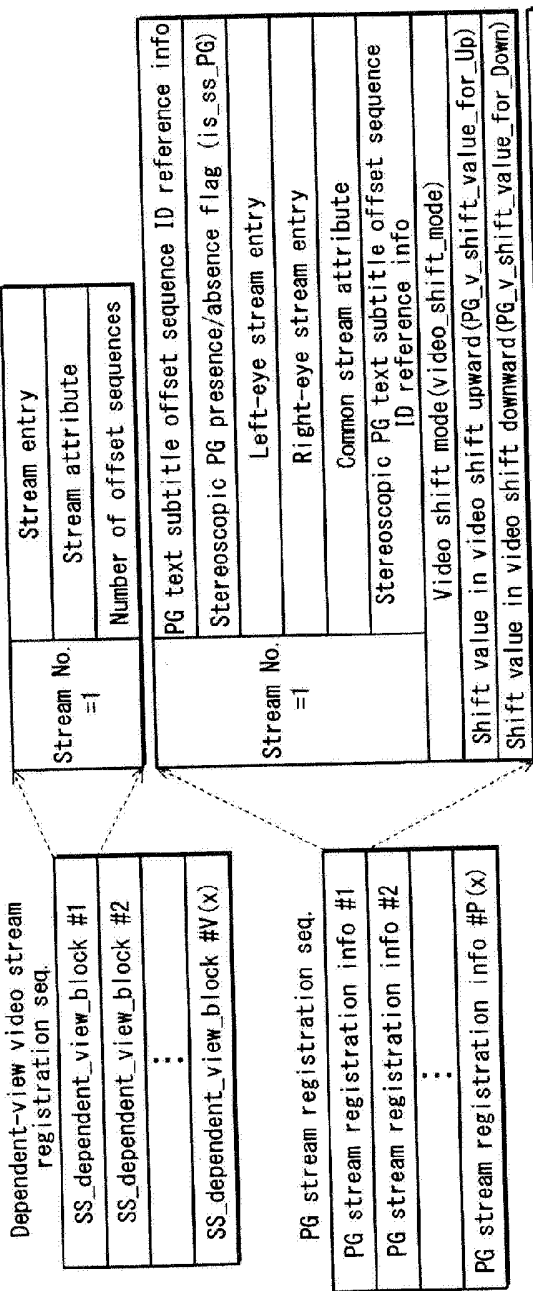


FIG. 35B

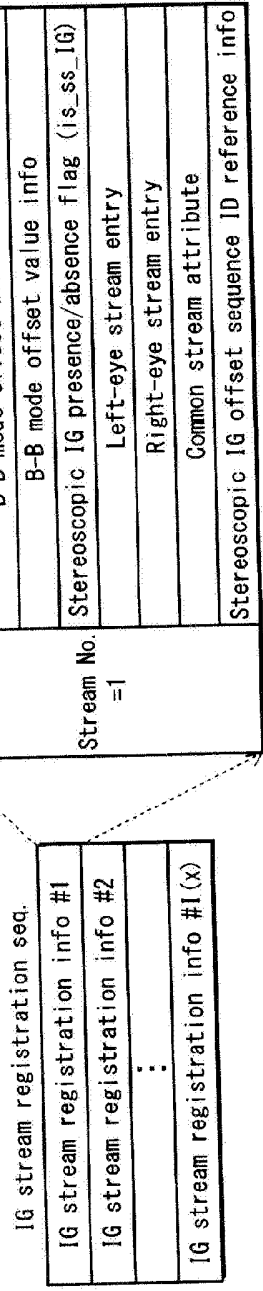


FIG. 35C

FIG. 36

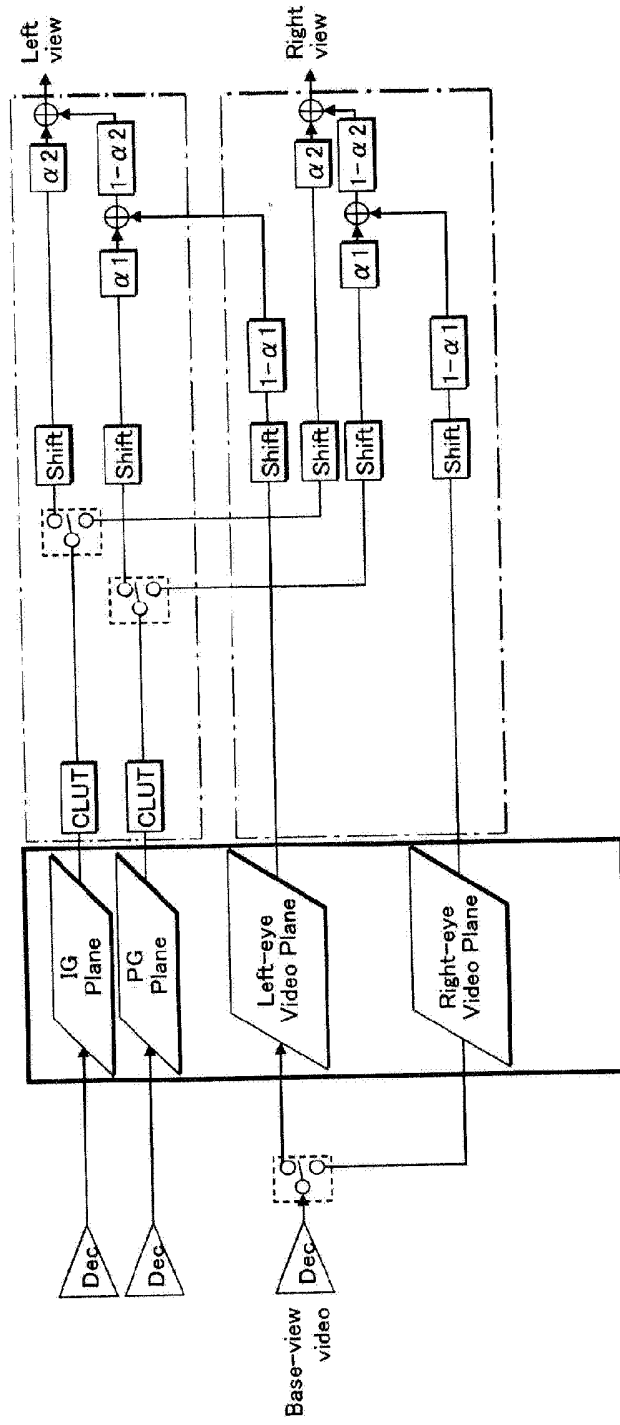


FIG. 37

PSR33

PG_shift_value_for_Up	IG_shift_value_for_Up	SV_shift_value_for_Up	IM_shift_value_for_Up
PG_shift_value_for_Down	IG_shift_value_for_Down	SV_shift_value_for_Down	IM_shift_value_for_Down

FIG. 38A

Case where Video_shift_mode is "Keep":

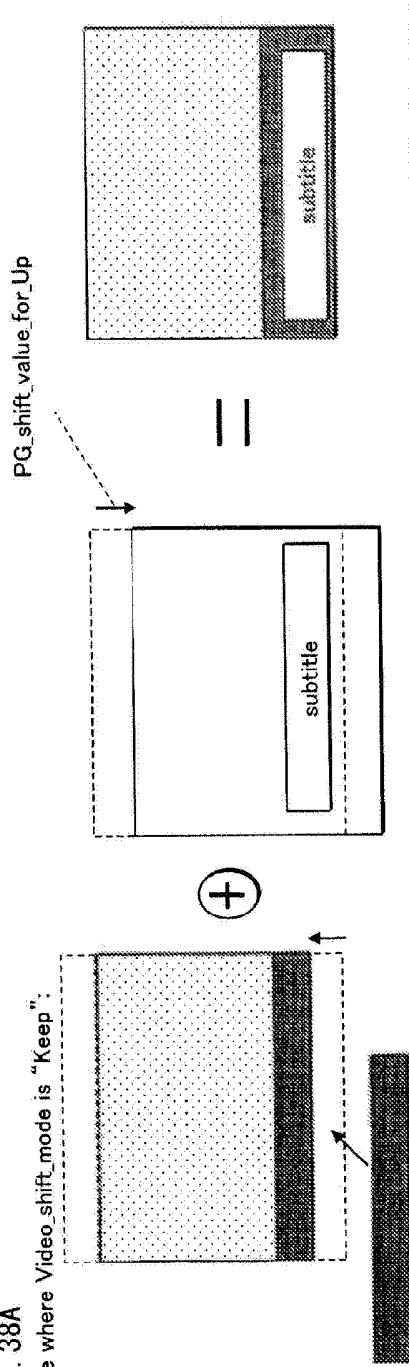
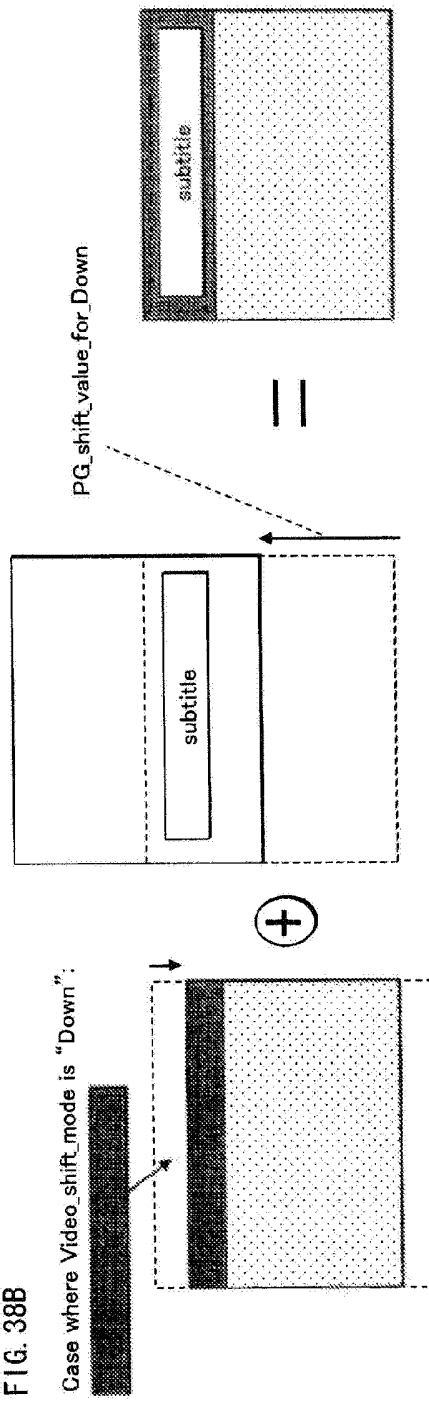


FIG. 38B

Case where Video_shift_mode is "Down":



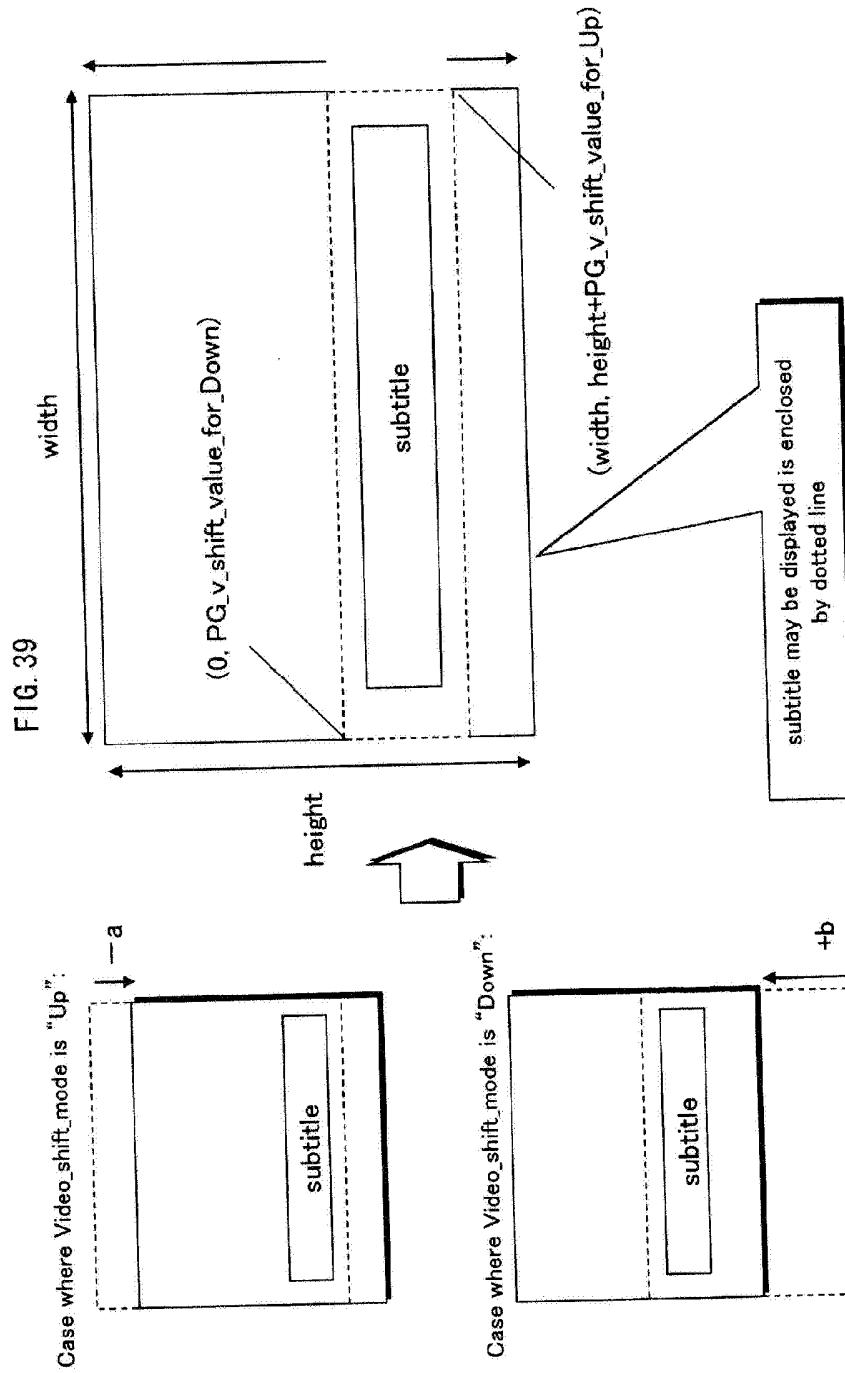


FIG. 40

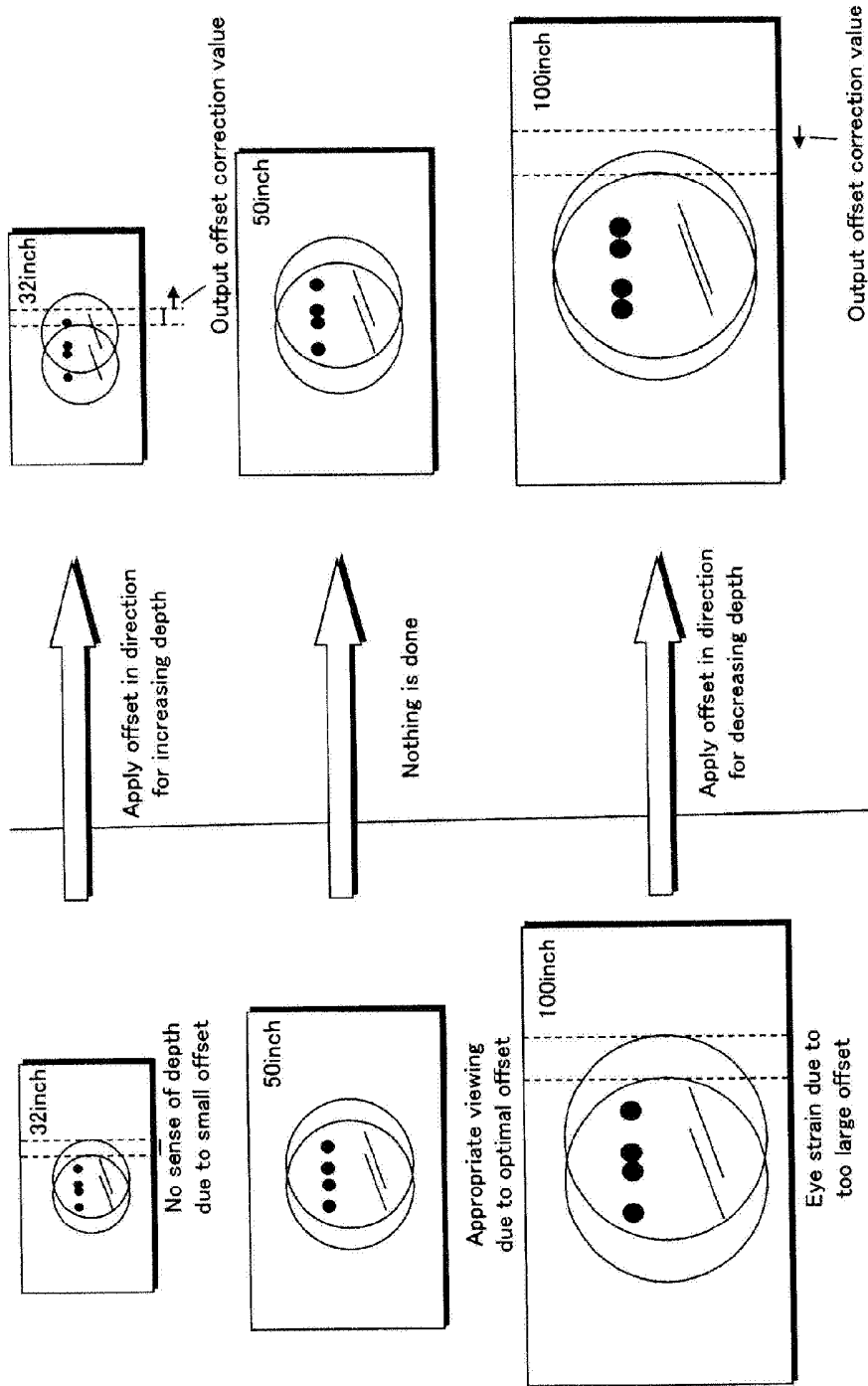


FIG. 41A

Inch	Output offset correction value
Below 10	+3
11-20	+2
21-30	+2
31-40	+1
41-50	+0
51-60	-1
61-70	-1
71-80	-2
81-90	-3
Over 91	-3

FIG. 41B

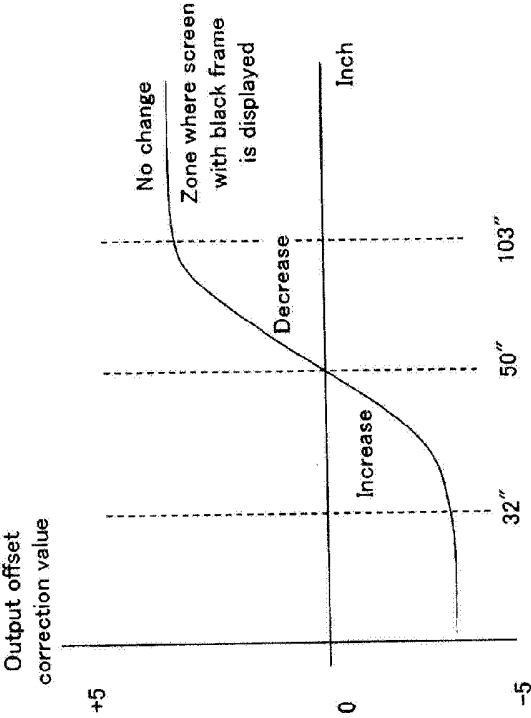


FIG. 42

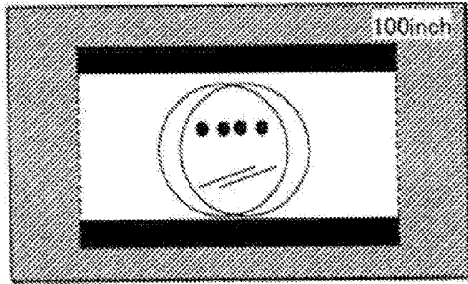


FIG. 43

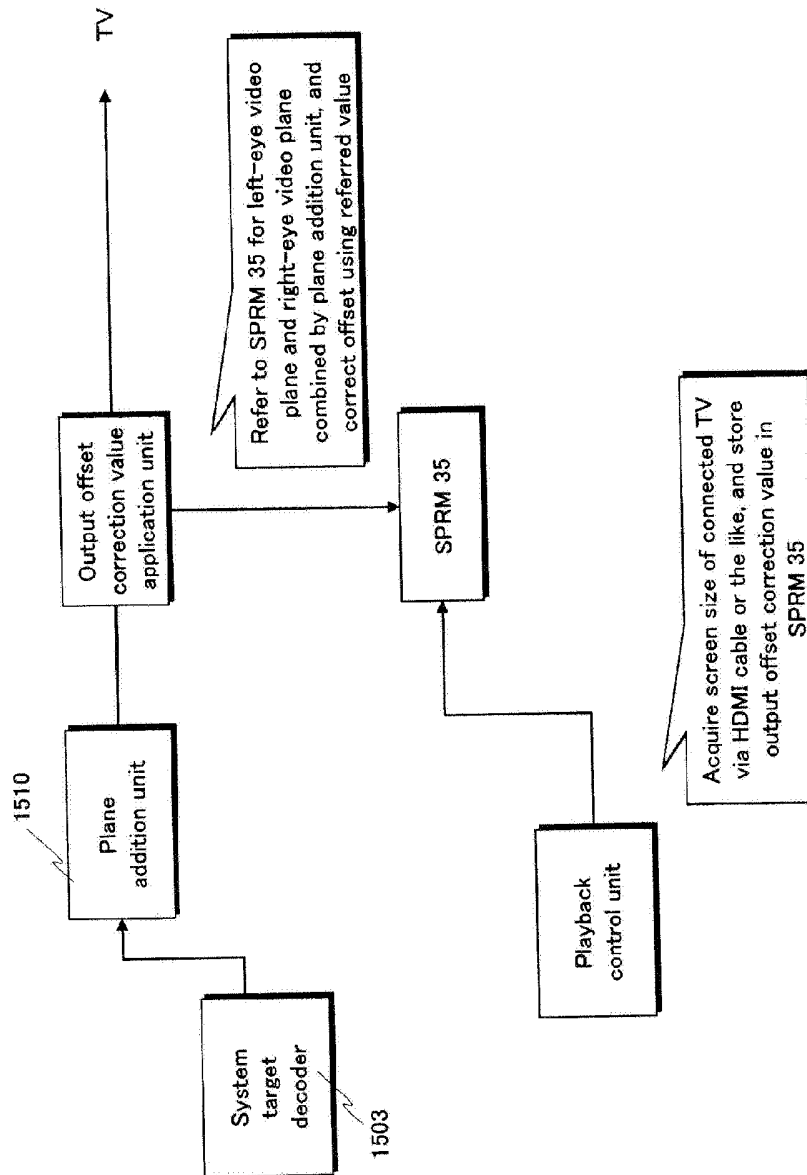


FIG. 44

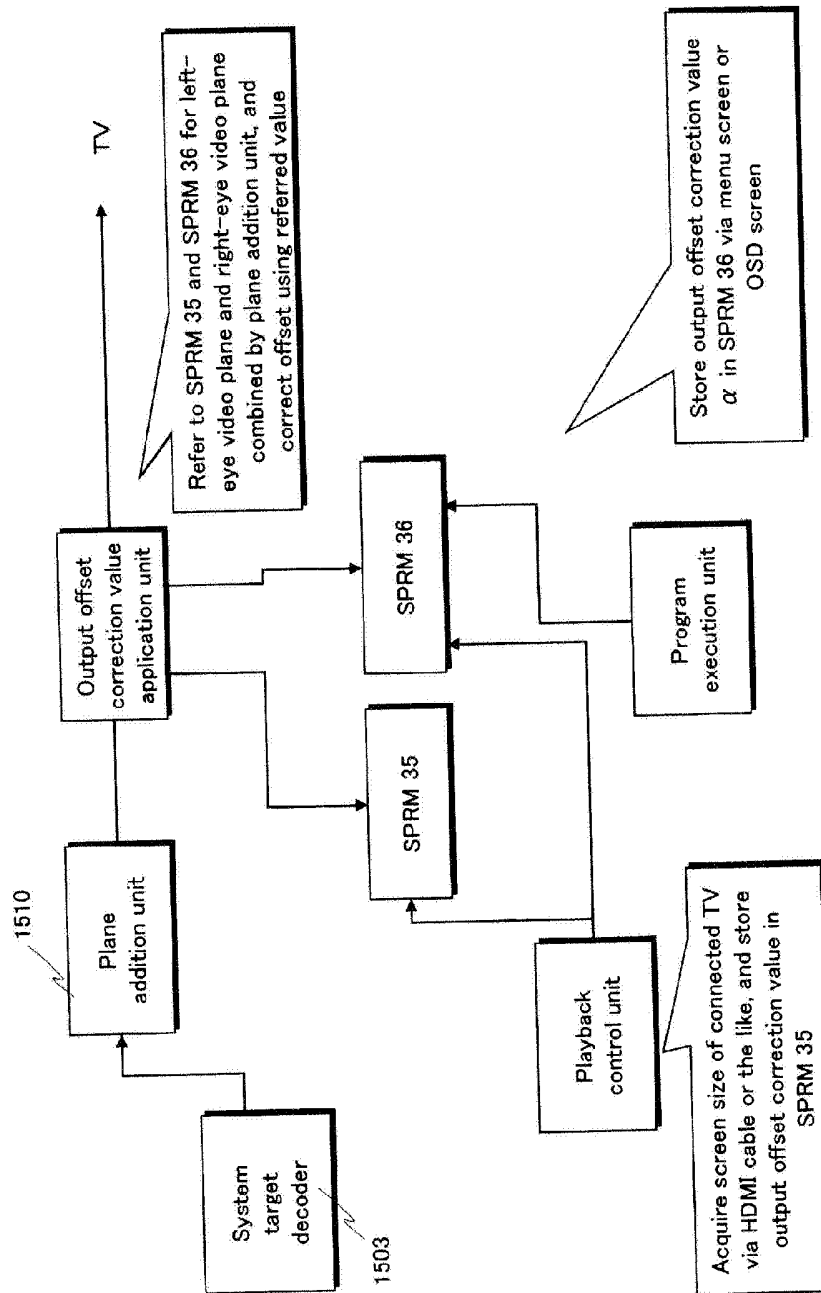


FIG. 45

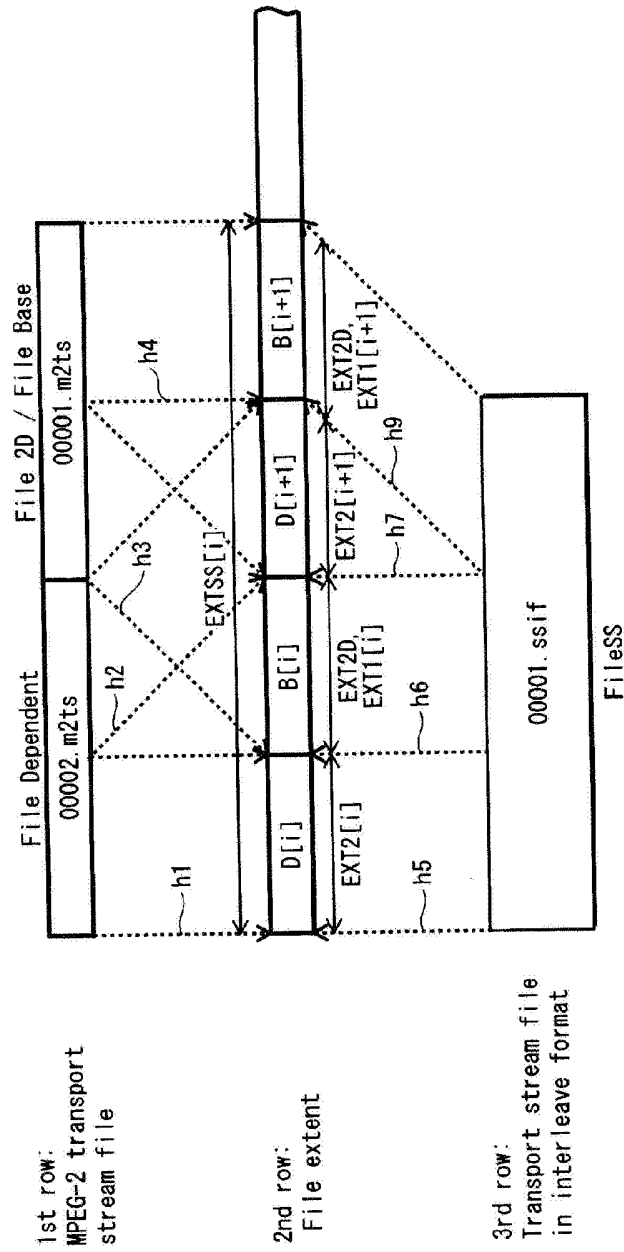


FIG. 46A

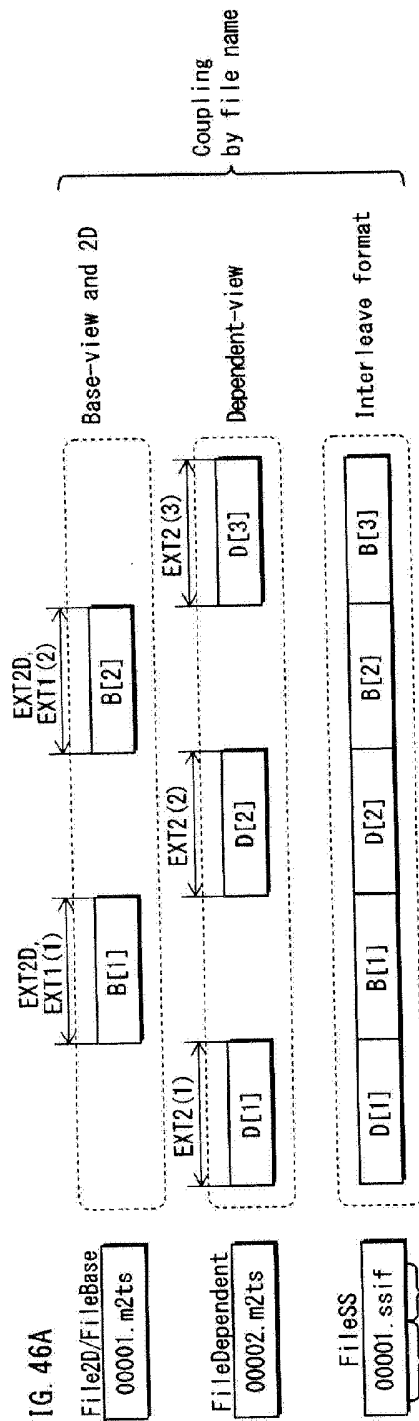


FIG. 46C

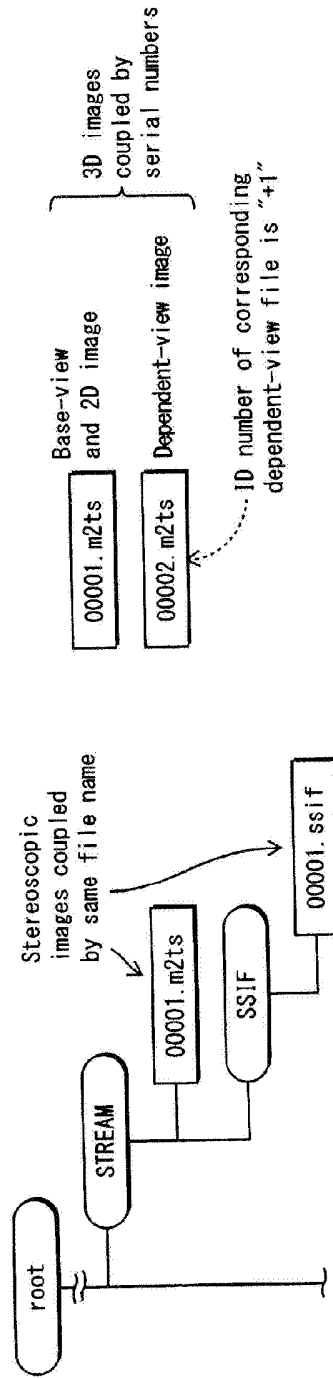


FIG. 46B

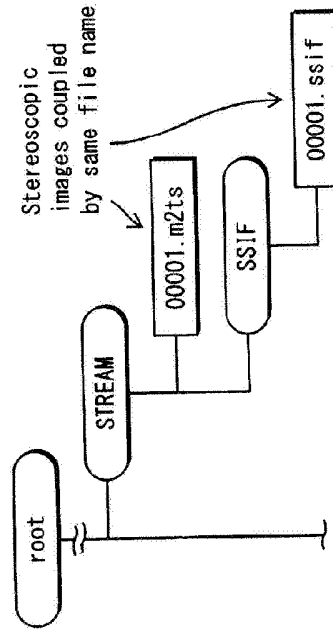


FIG. 47

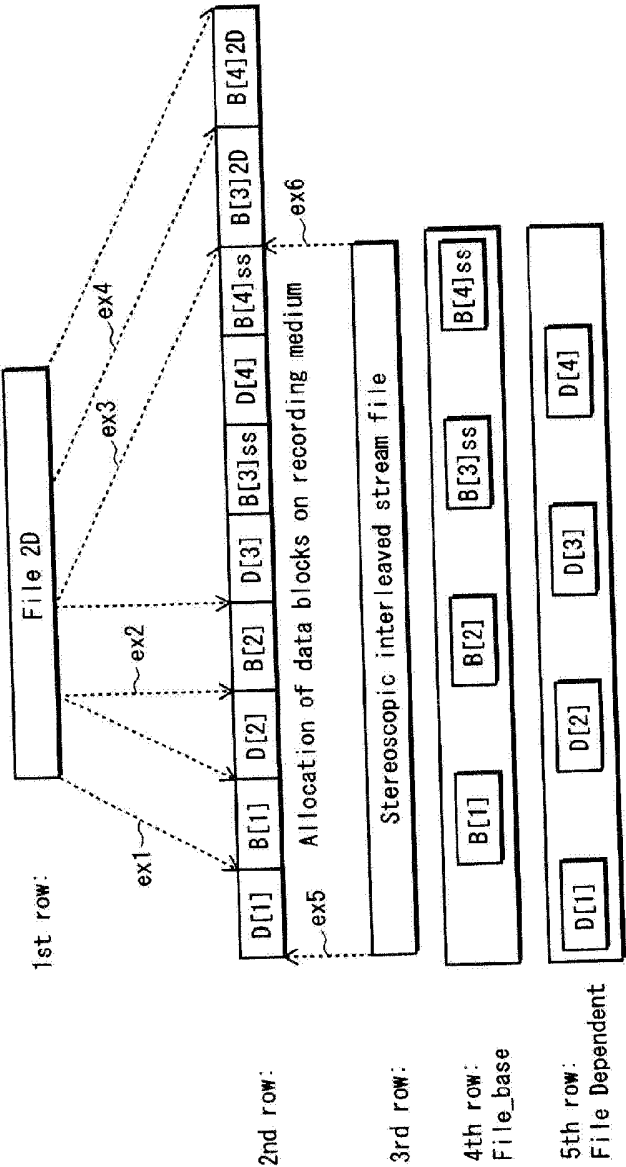


FIG. 48

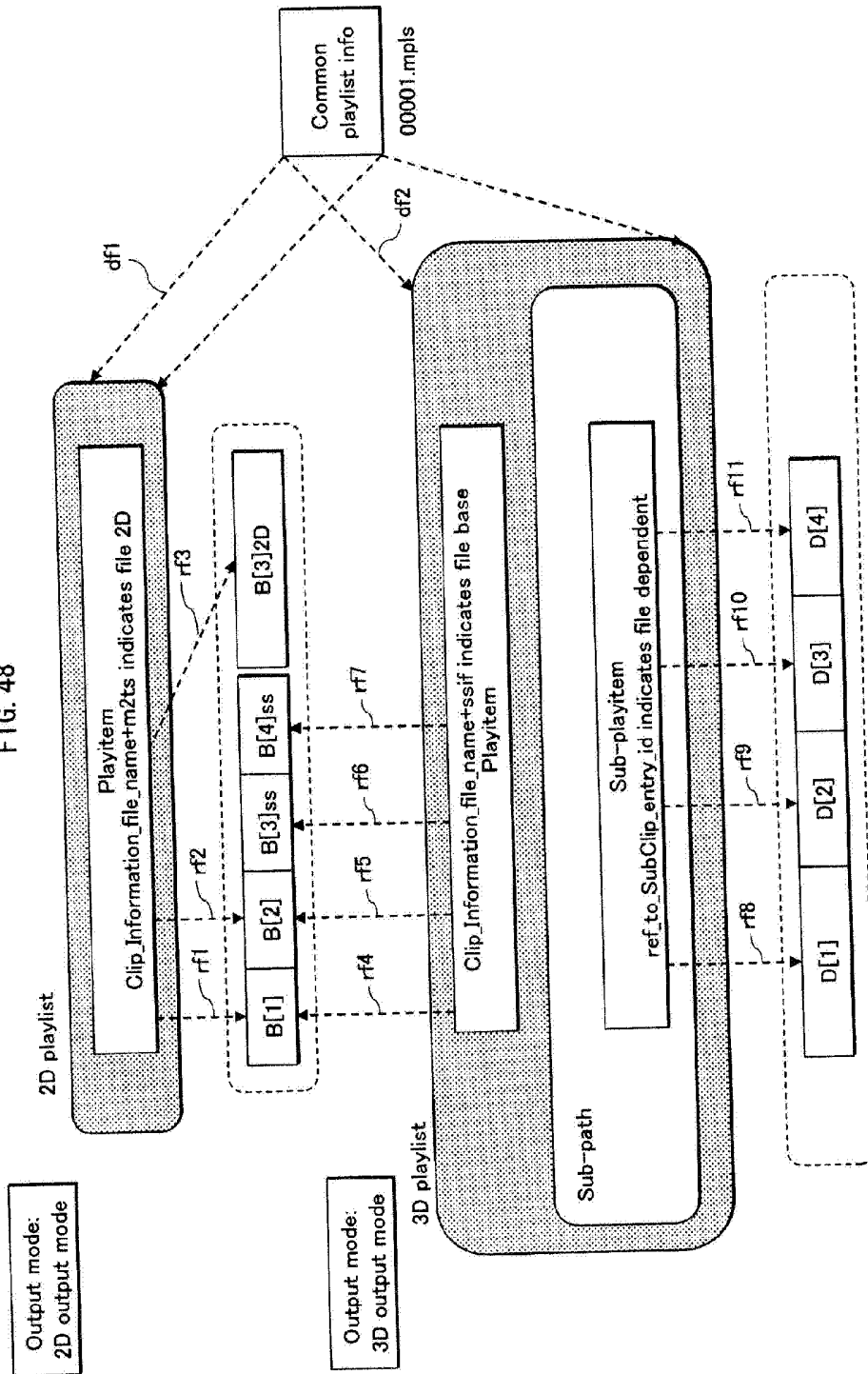


FIG. 49A Clip info file

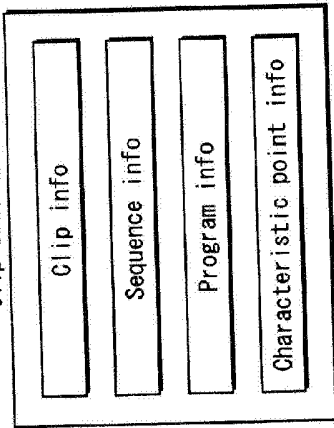


FIG. 49B Clip info file

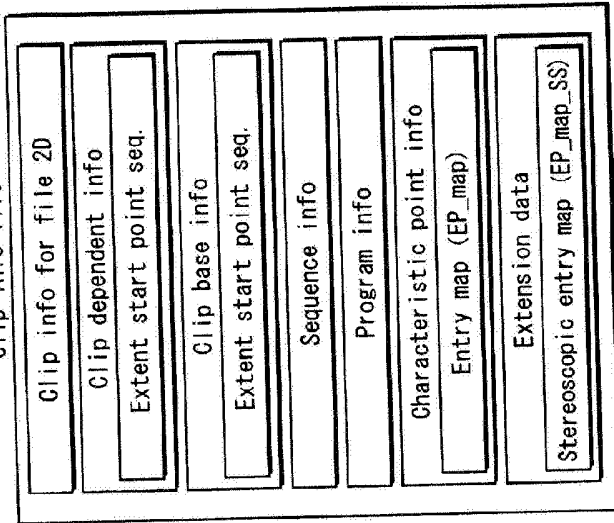


FIG. 49D Clip dependent info file (00002.clpi)

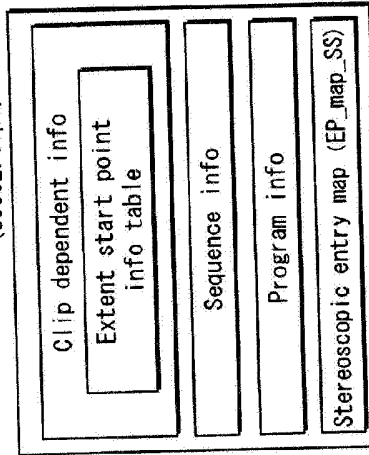


FIG. 49C Clip base info file (00001.clpi)

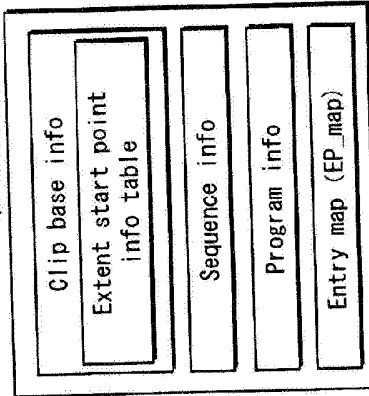


FIG. 50

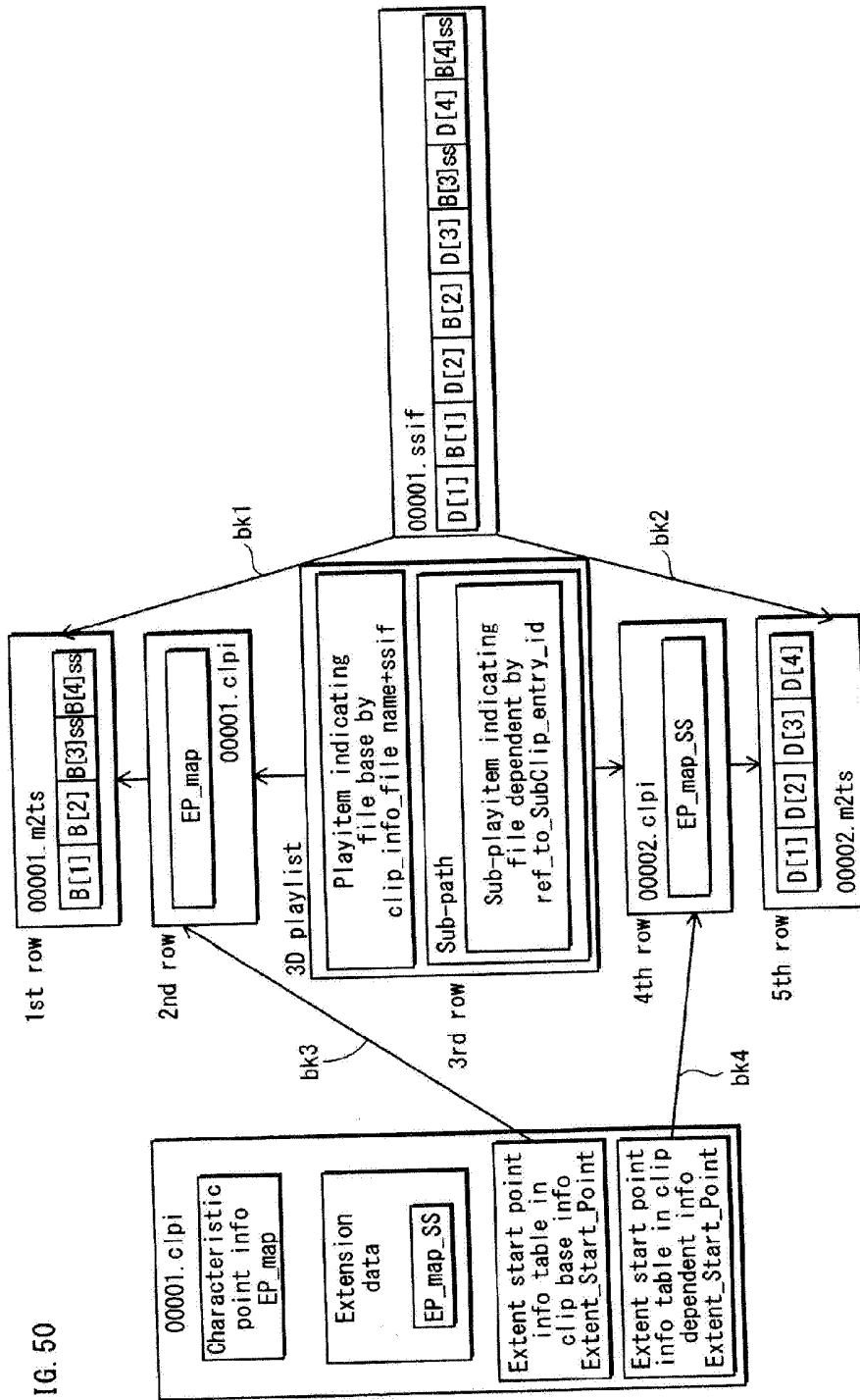


FIG. 51A

Clip base info, clip dependent info

Clip stream type info
Application type info
TS recording rate
Number of source packets
ATC delta
Extent start point info table

FIG. 51B

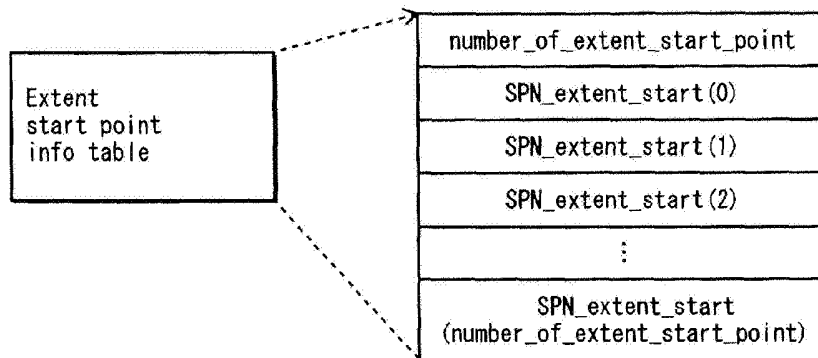


FIG. 52

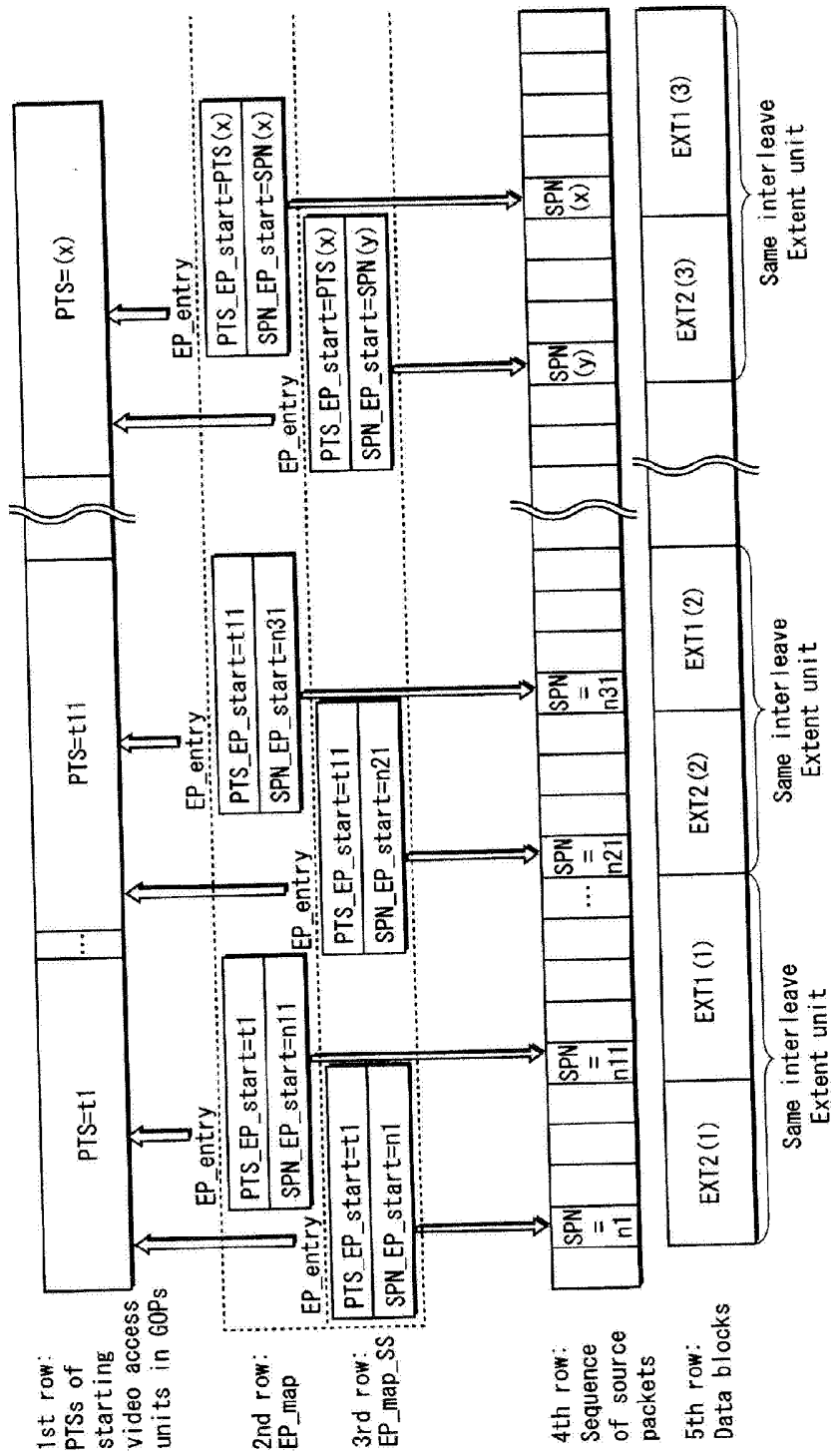


FIG. 54

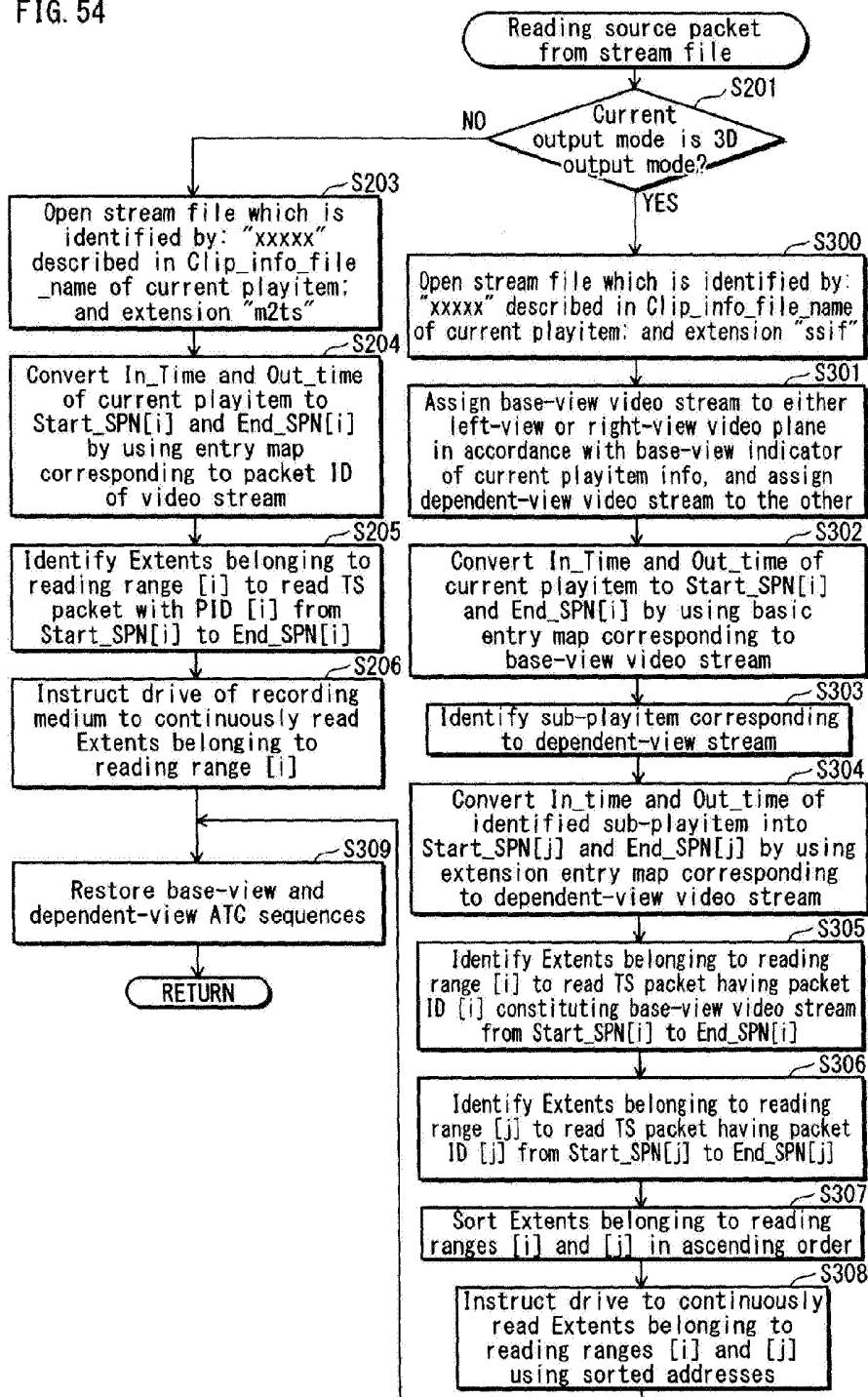
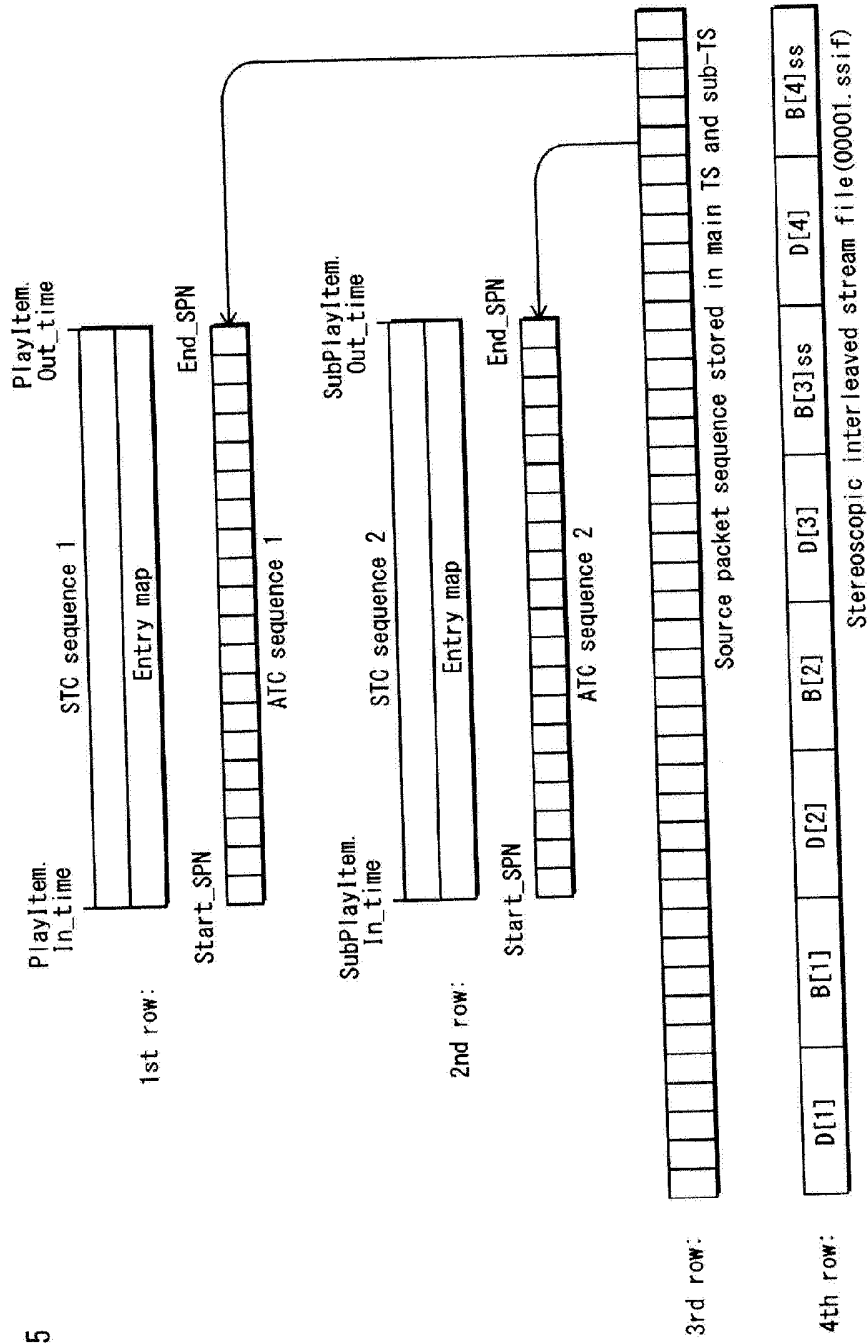


FIG. 55



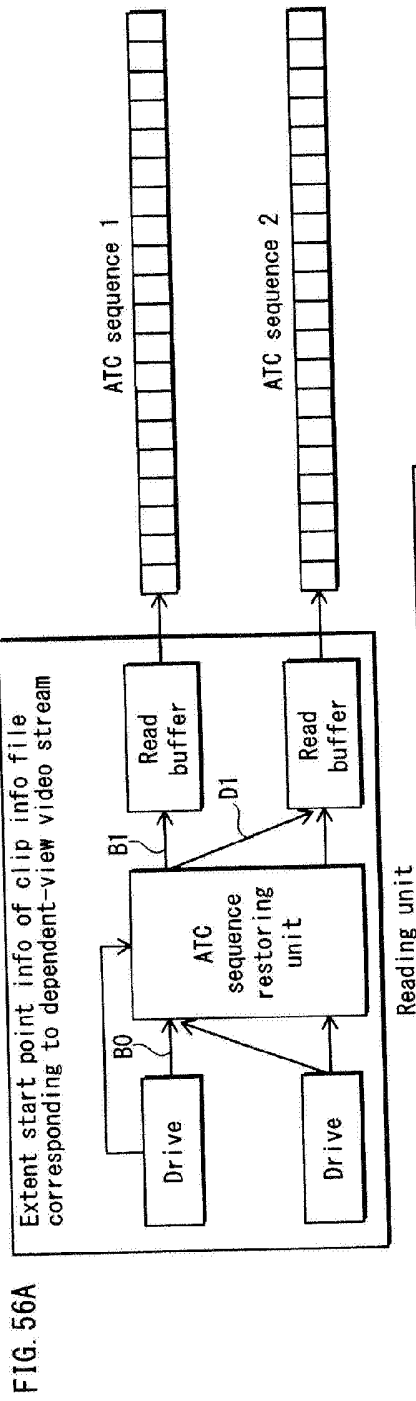
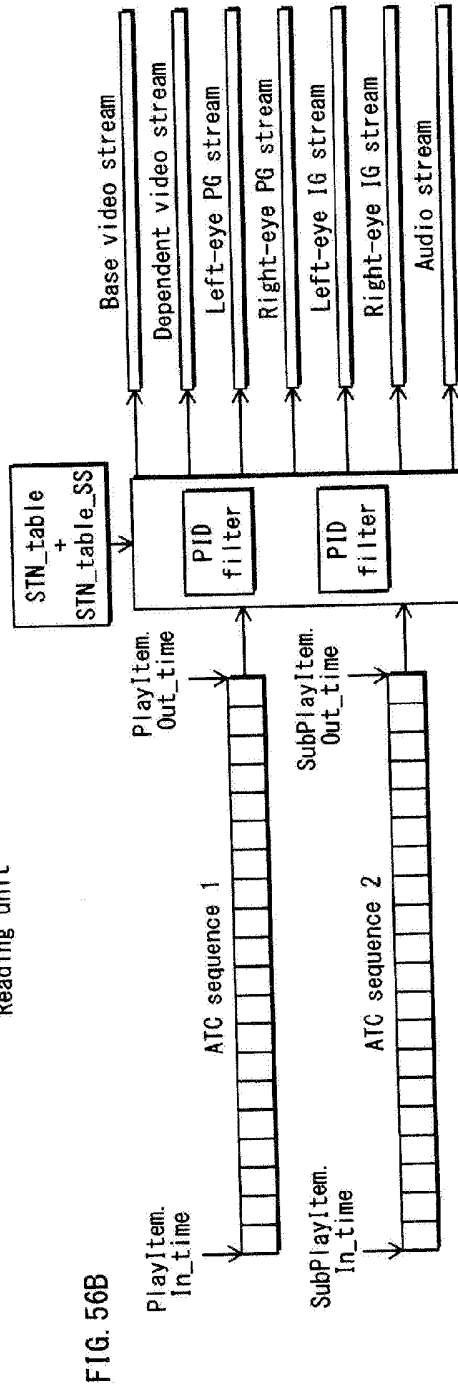


FIG. 57A

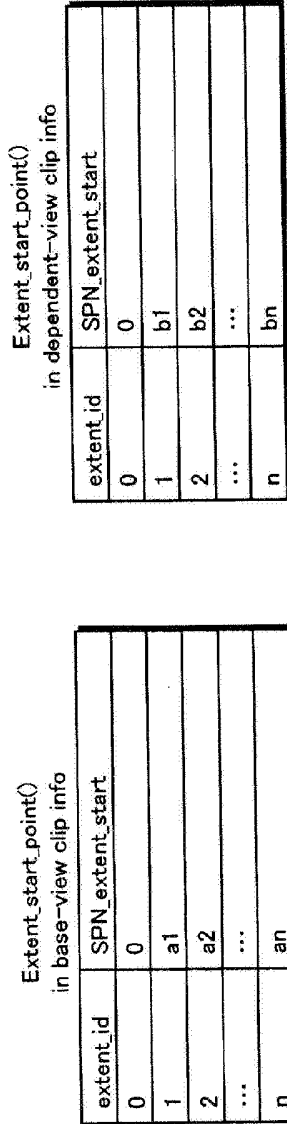


FIG. 57B

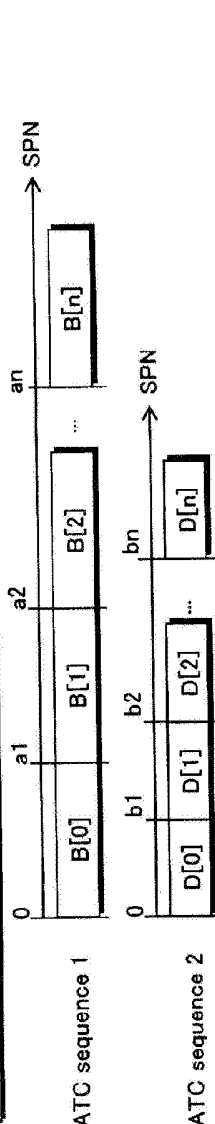


FIG. 57C

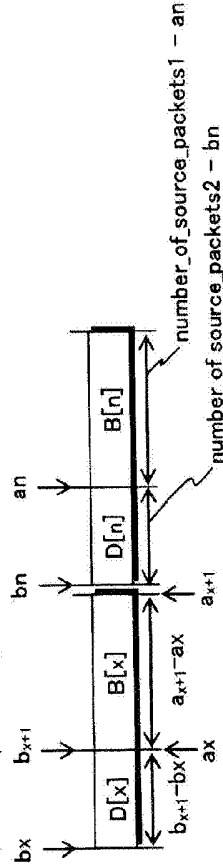


FIG. 57D

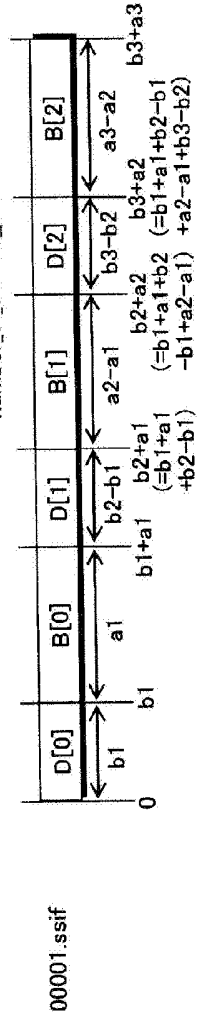


FIG. 59

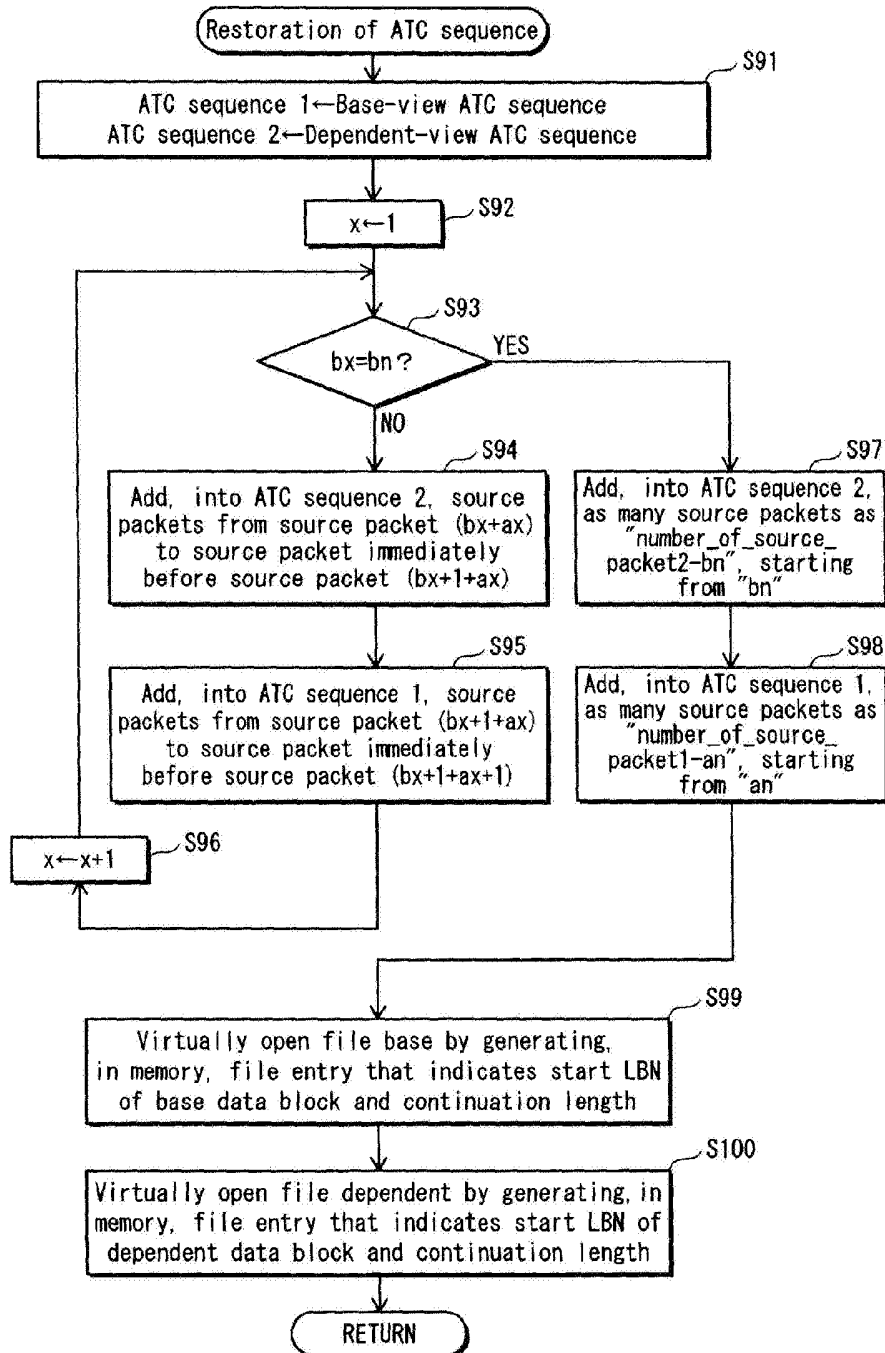


FIG. 60

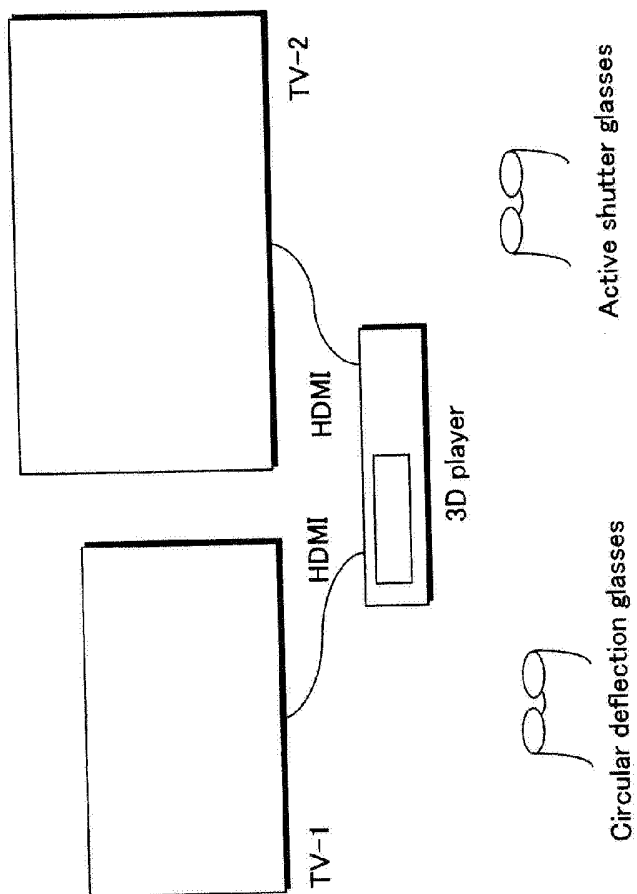


FIG. 61

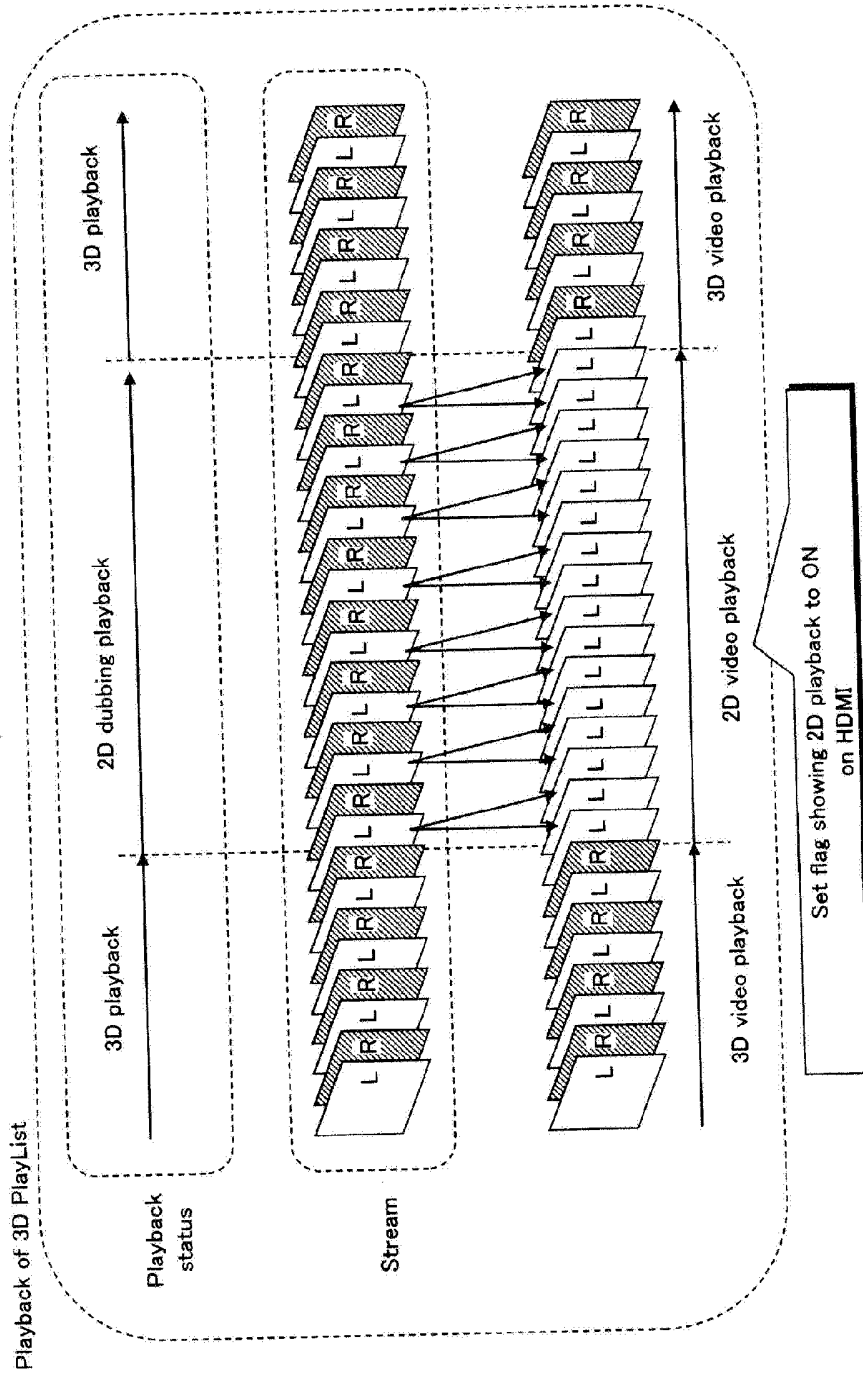
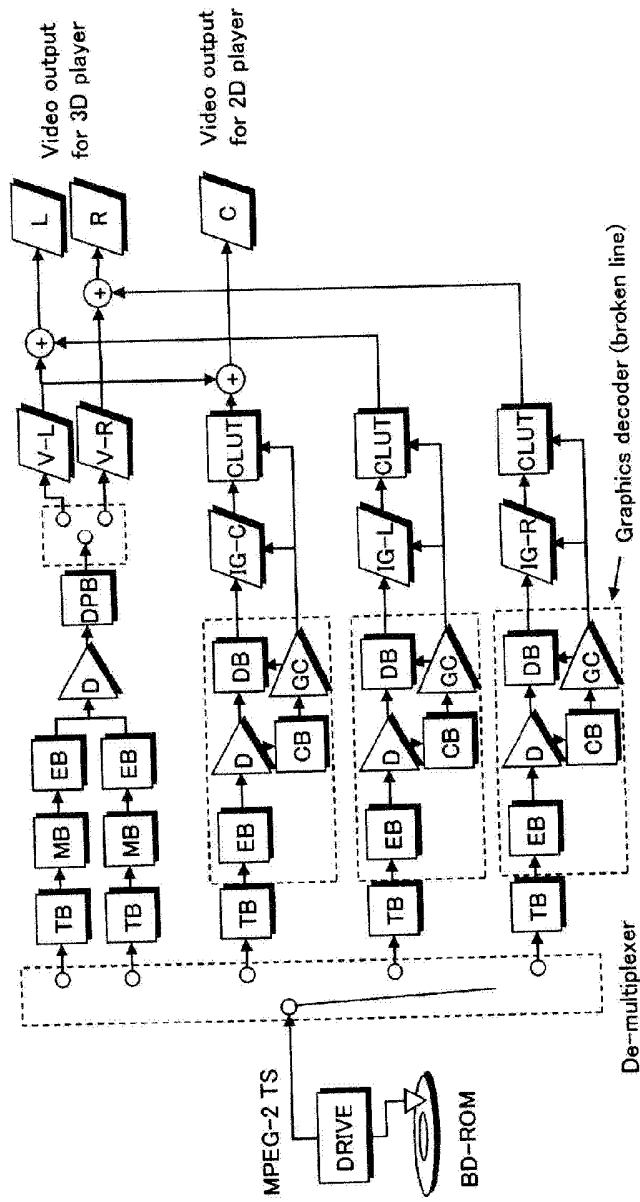
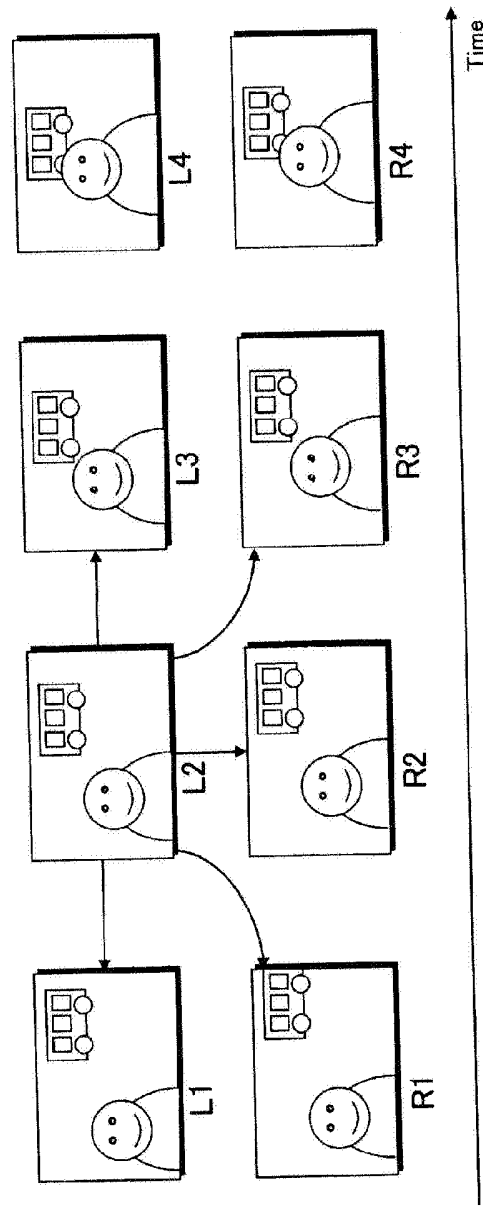


FIG. 62



Display time of equivalent object, menu structure, and command for execution are the same among C, L, and R such that equivalent subtitle and menu are realized between 2D and 3D

FIG. 63



Although only L picture is displayed in 2D display mode, R picture that is not used for increasing image quality is also used. Accordingly, connection mode of HDMI is for 3D, and L video and R video are input.

FIG. 64A

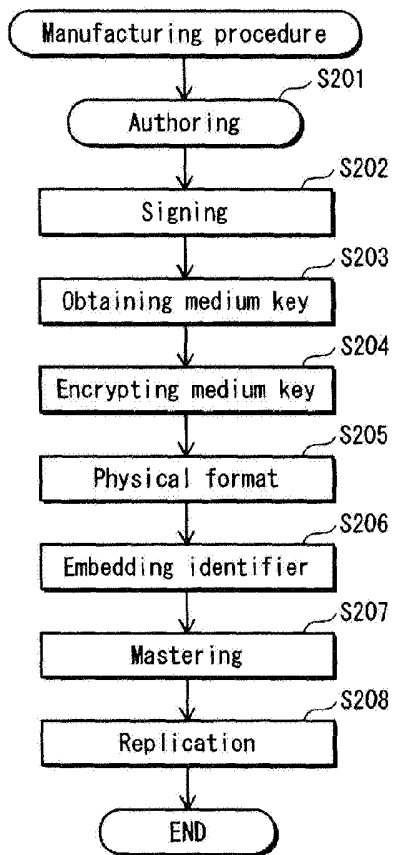


FIG. 64B

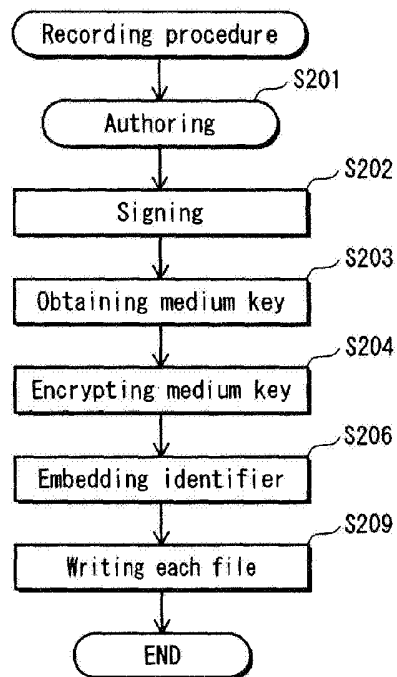


FIG. 65

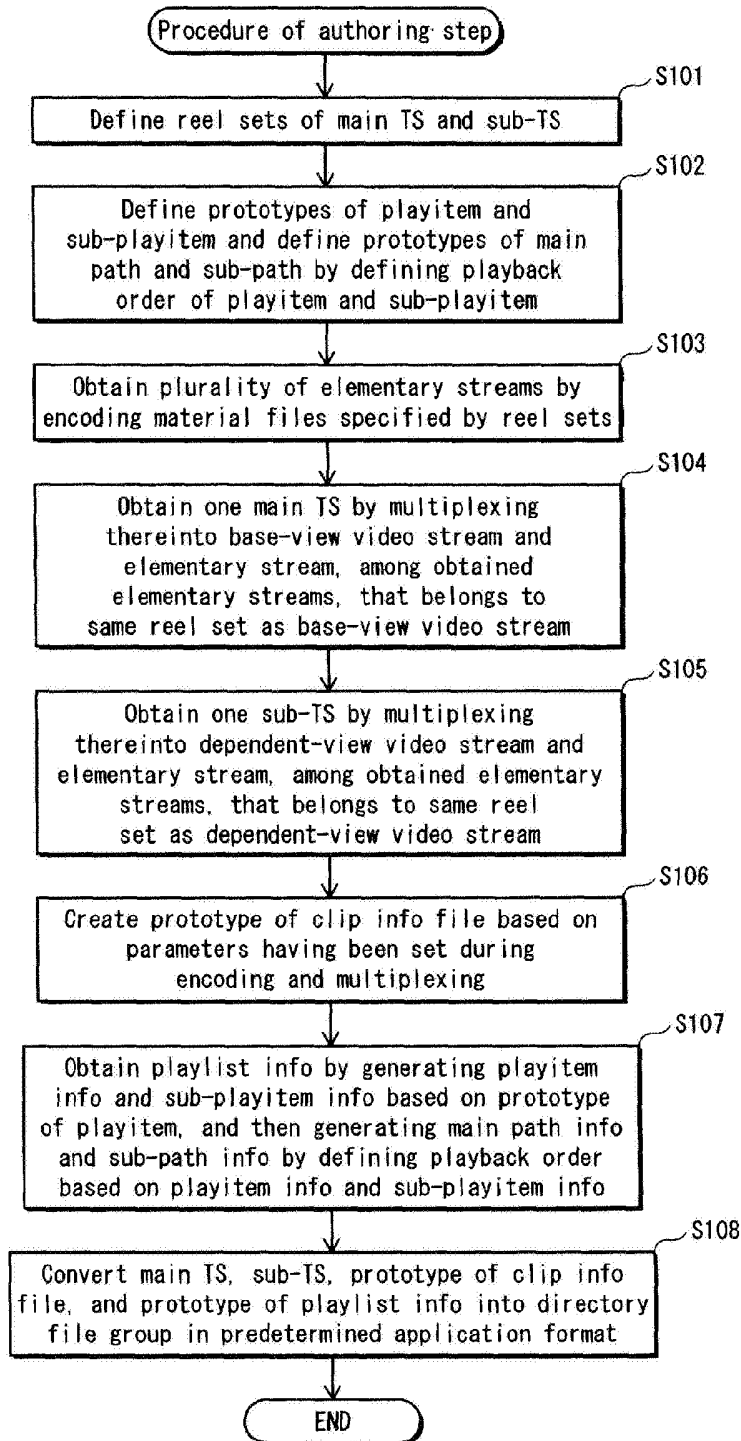


FIG. 66

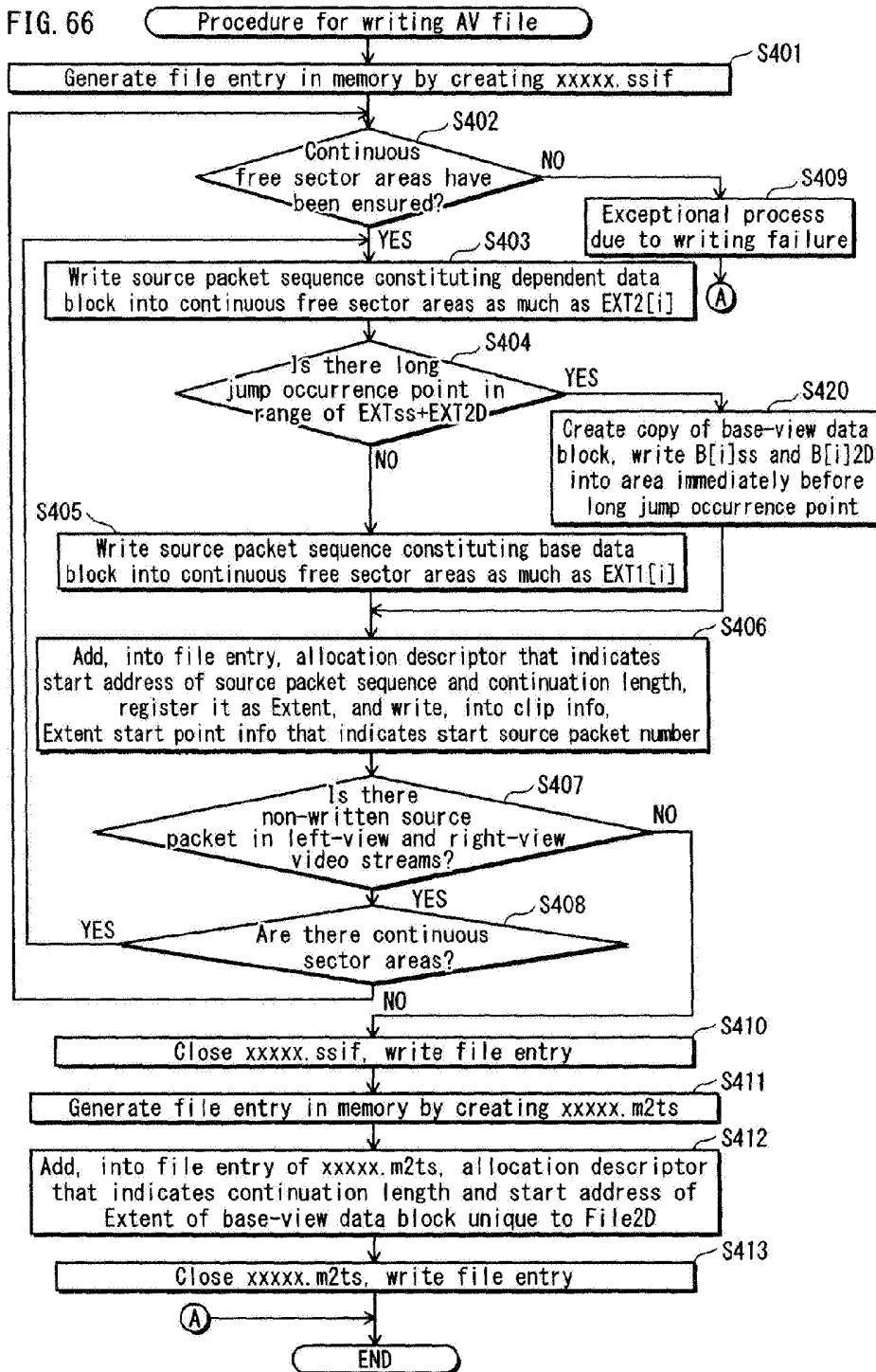


FIG. 67

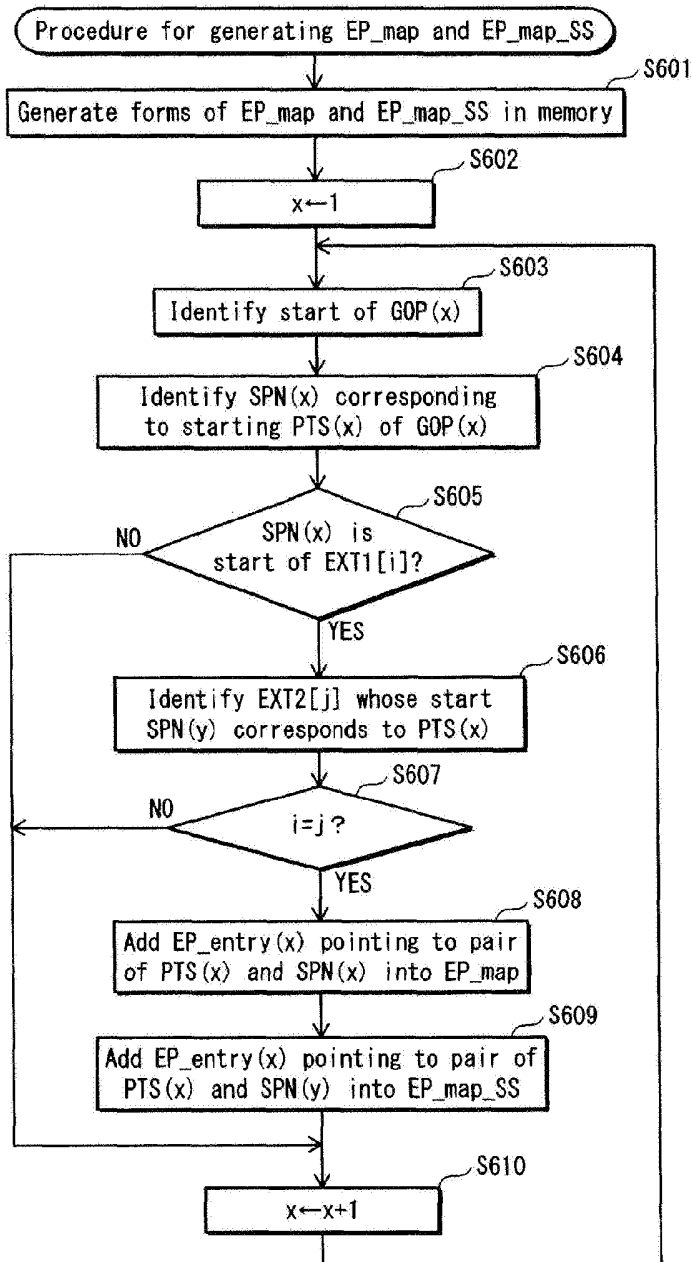


FIG. 68

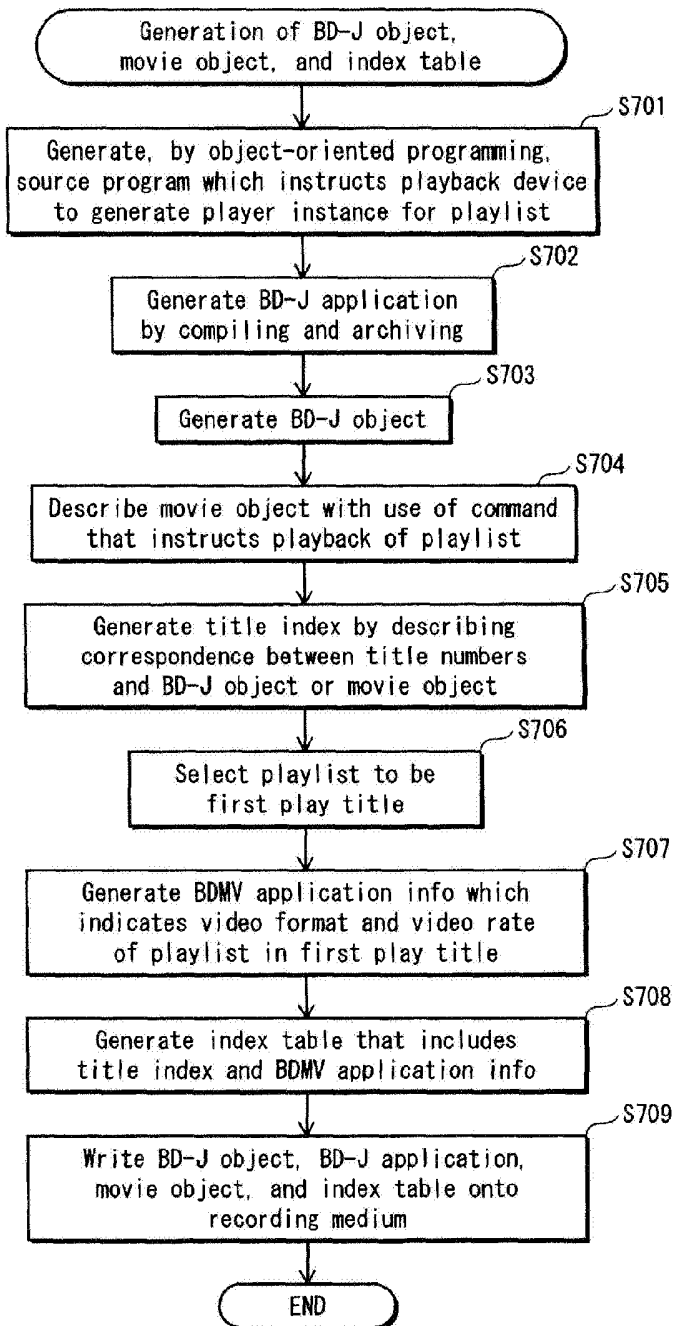
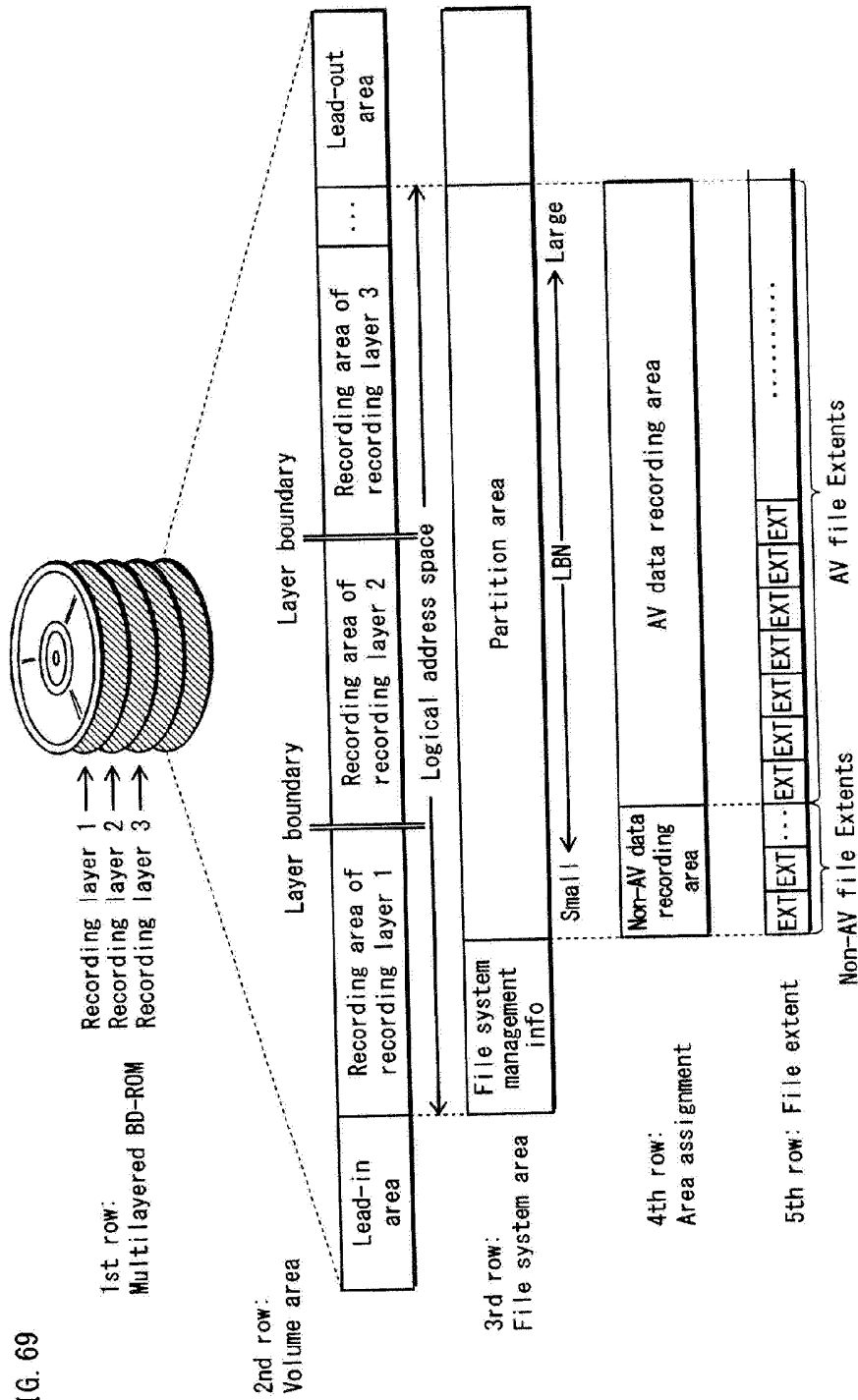
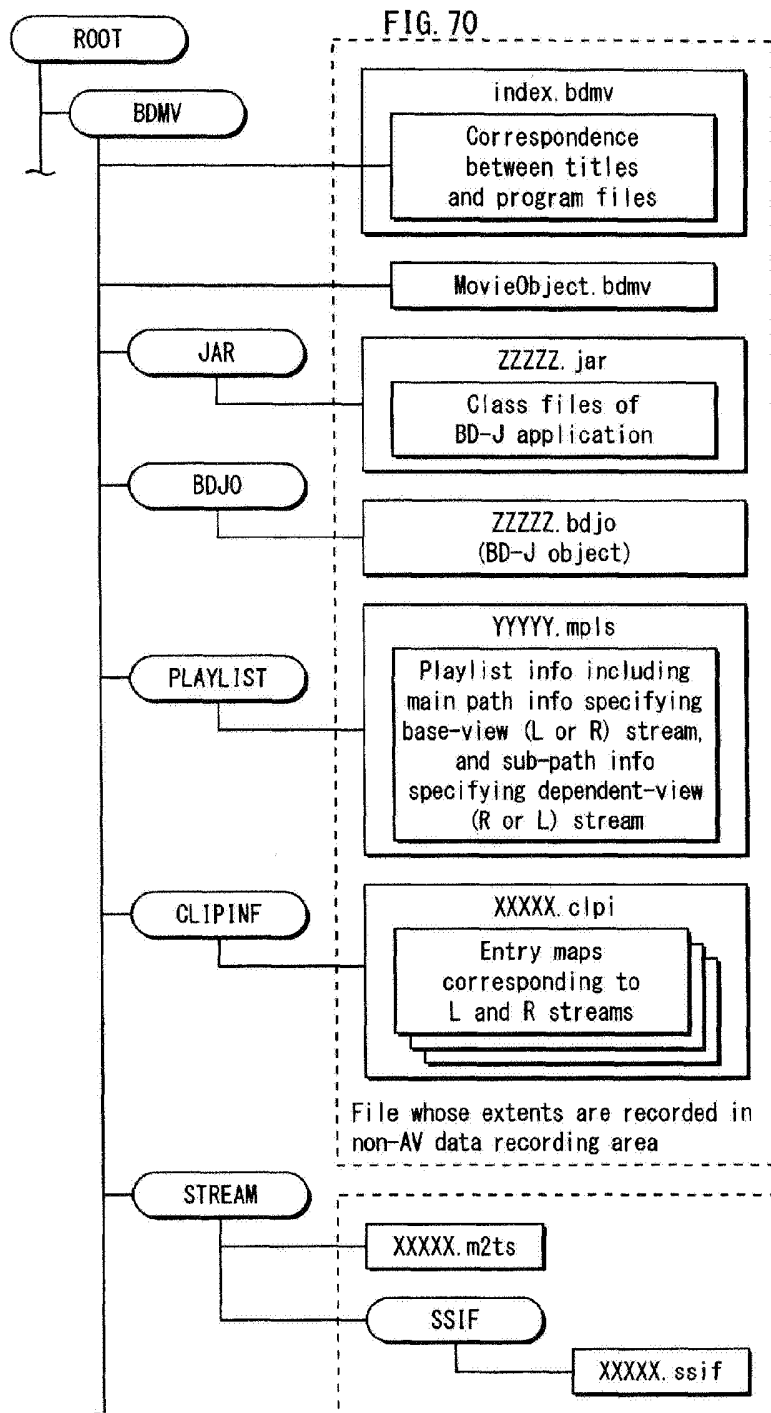


FIG. 69





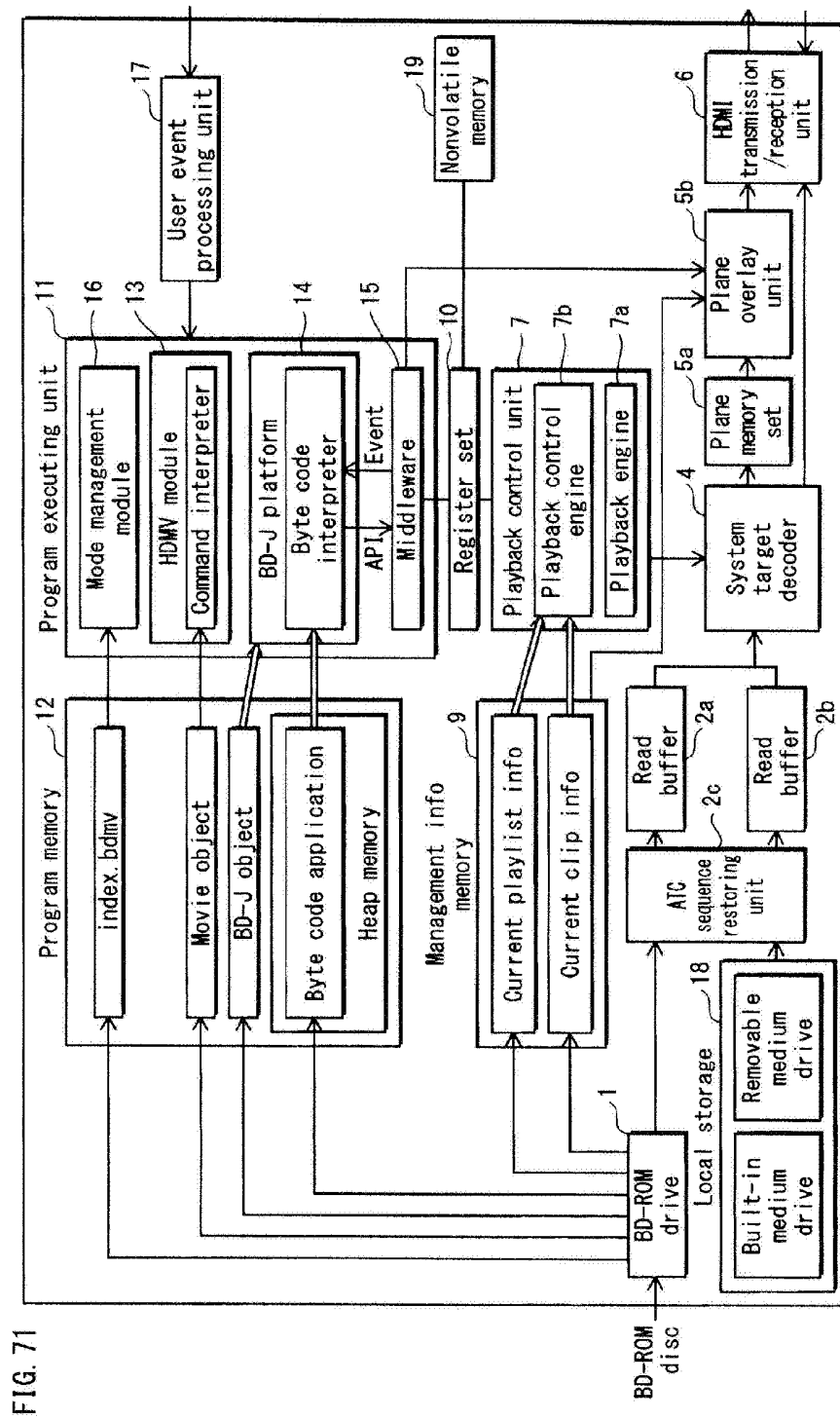


FIG. 72A

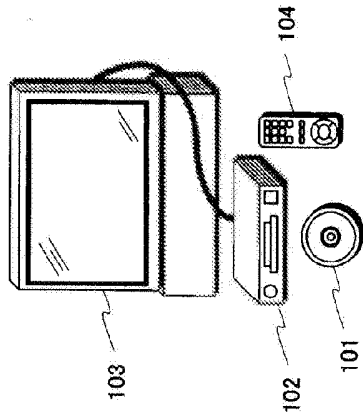


FIG. 72B

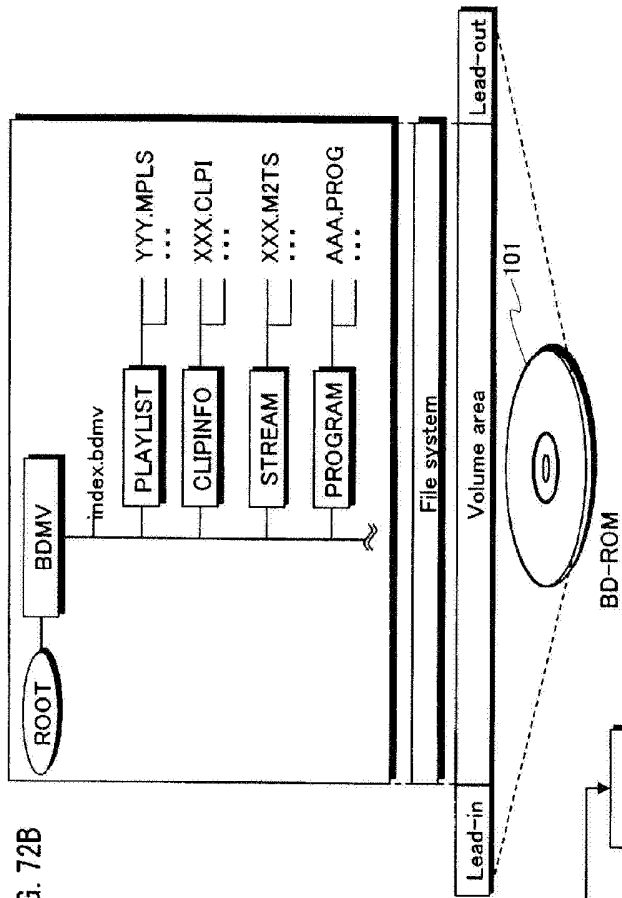


FIG. 72C

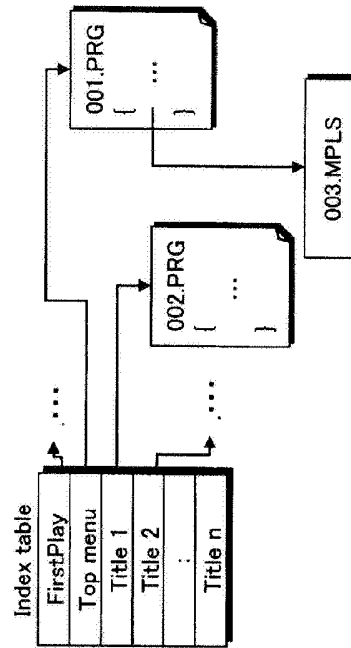


FIG. 73A

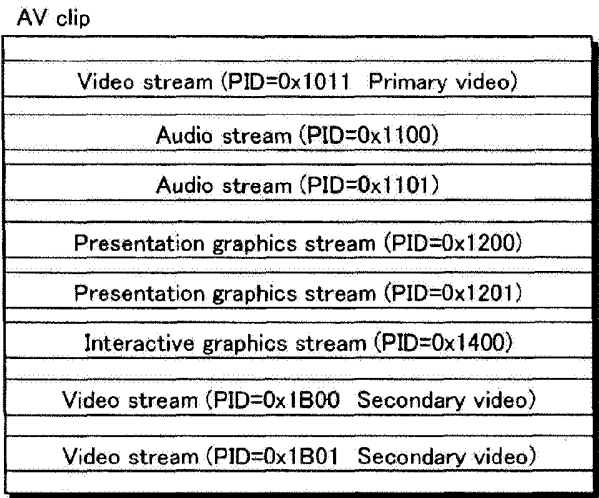


FIG. 73B

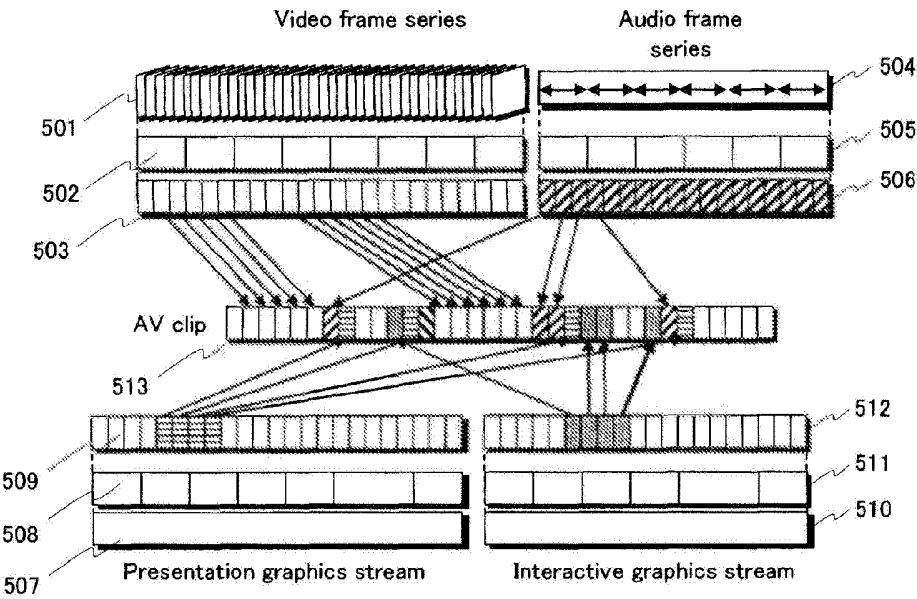


FIG. 74A

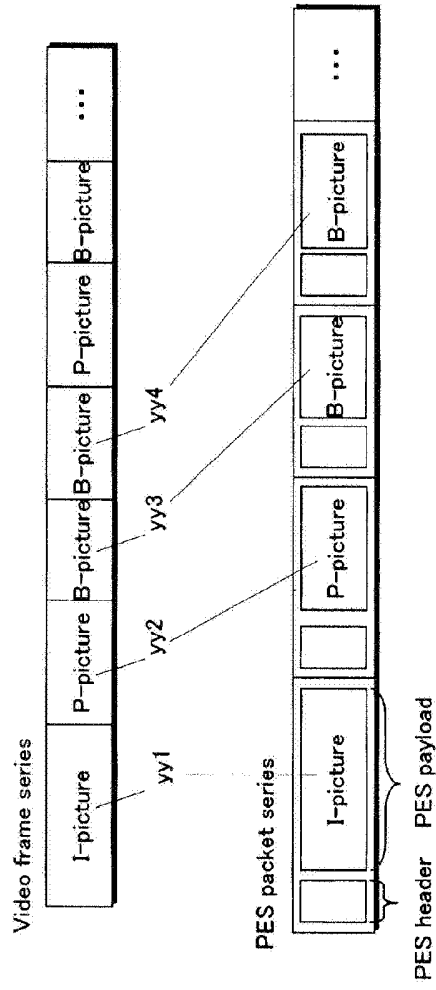


FIG. 74B

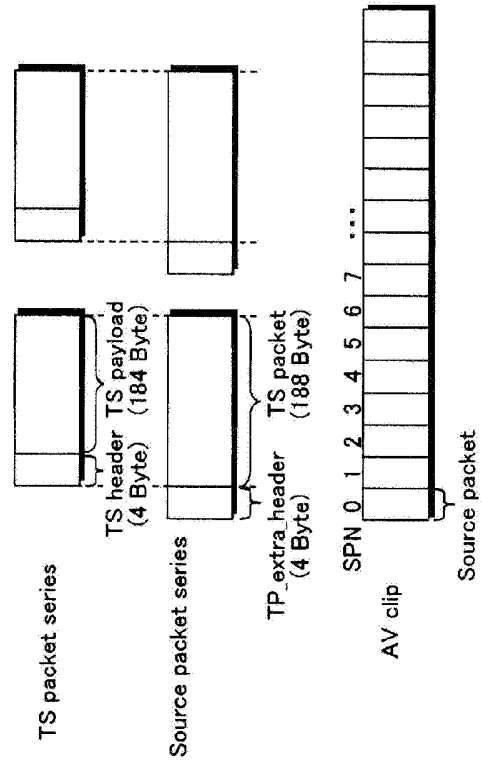


FIG. 75A

Data structure of PMT

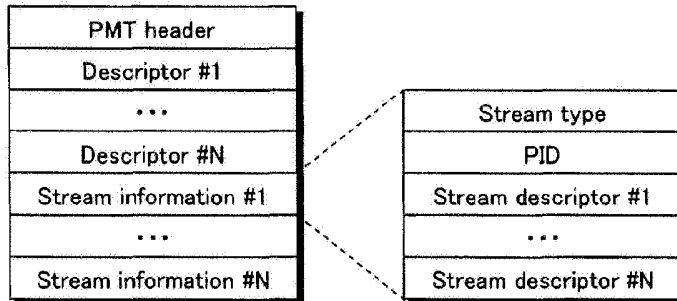


FIG. 75B

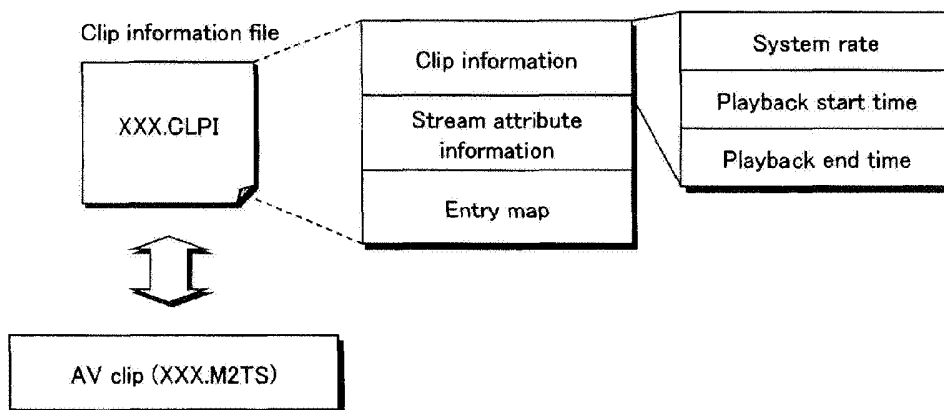


FIG. 76A

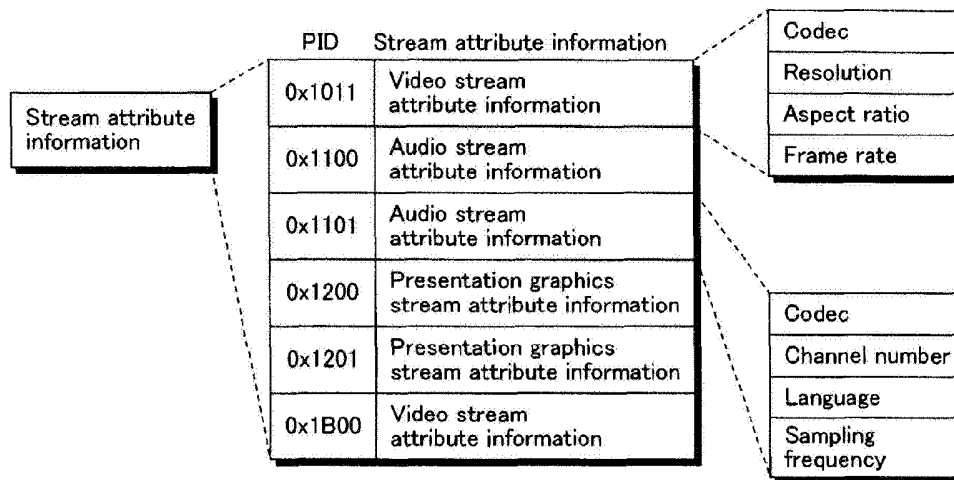


FIG. 76B

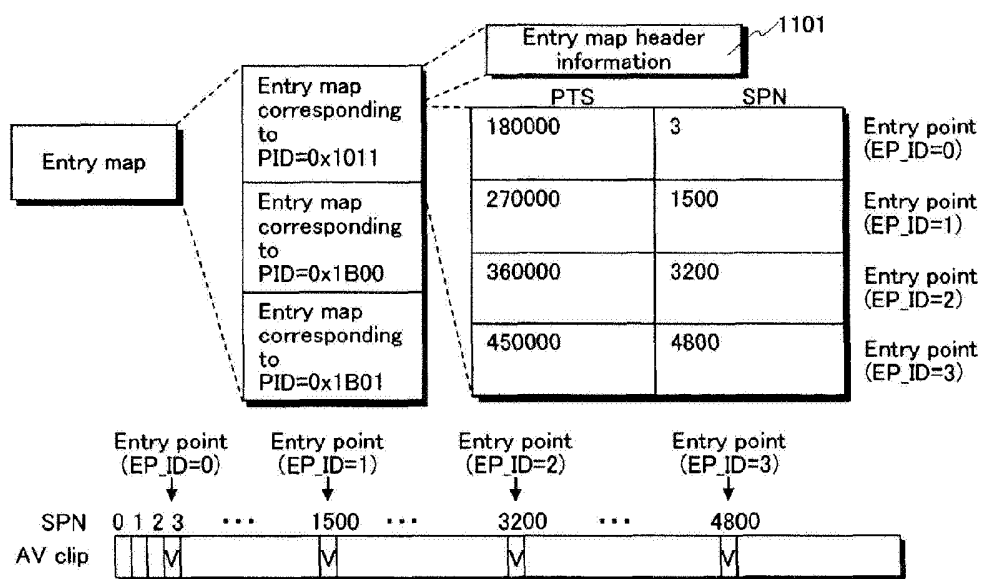


FIG. 77A

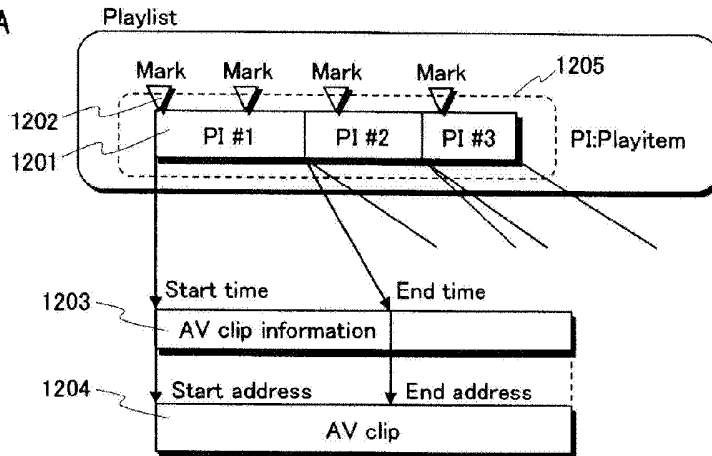


FIG. 77B

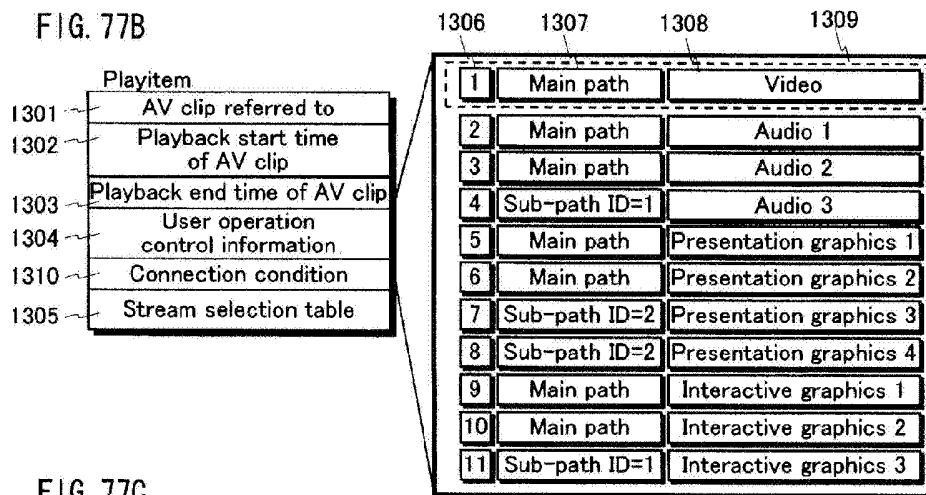


FIG. 77C

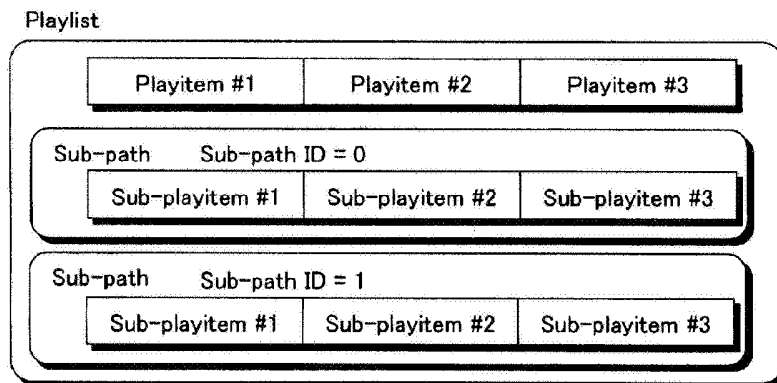


FIG. 78A

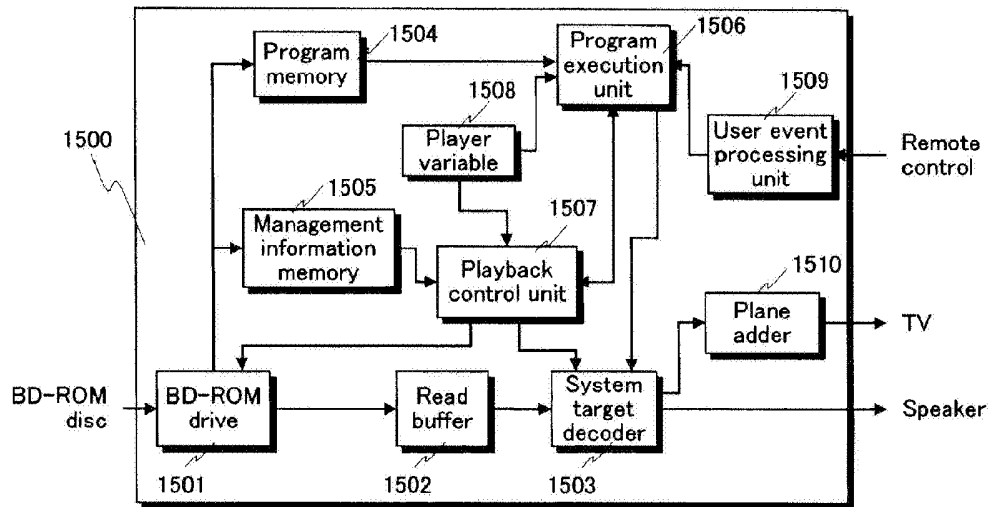
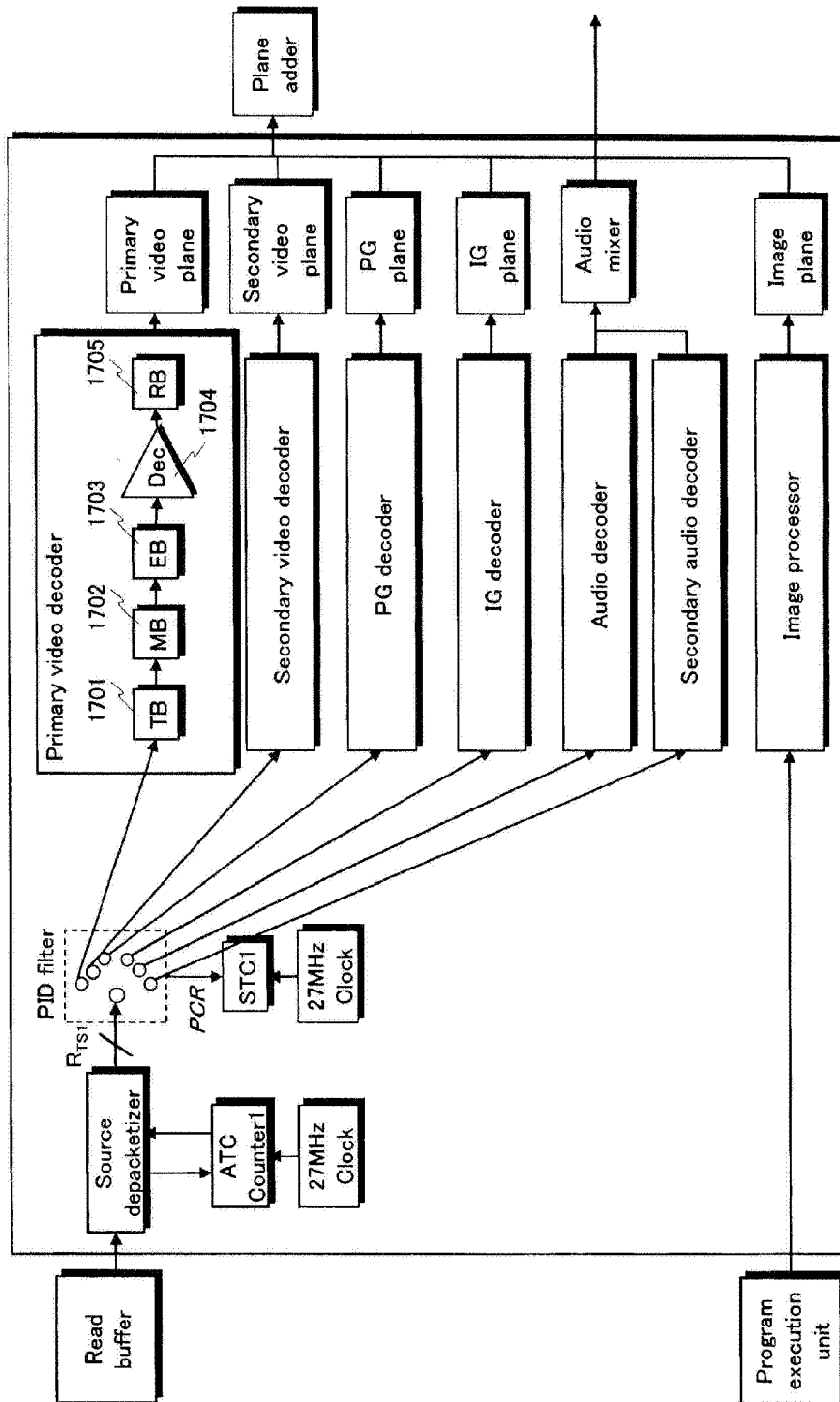


FIG. 78B

Player variable (system parameter)

0	Language Code	11	Player audio mixing mode for Karaoke	22	2D/3D Output Mode
1	Audio stream number	12	Country code for parental management	23	Display Capability
2	Subtitle stream number	13	Parental level	24	3D-Capability
3	Angle number	14	Player configuration for Video	25	reserved
4	Title number	15	Player configuration for Audio	26	reserved
5	Chapter number	16	Language code for AST	27	reserved
6	Program number	17	Language code ext. for AST	28	reserved
7	Cell number	18	Language code for STST	29	reserved
8	Key name	19	Language coded ext. for STST	30	reserved
9	Navigation timer	20	Player region code	31	reserved
10	Current playback time	21	2D/3D Output User Preference	32	reserved

FIG. 79



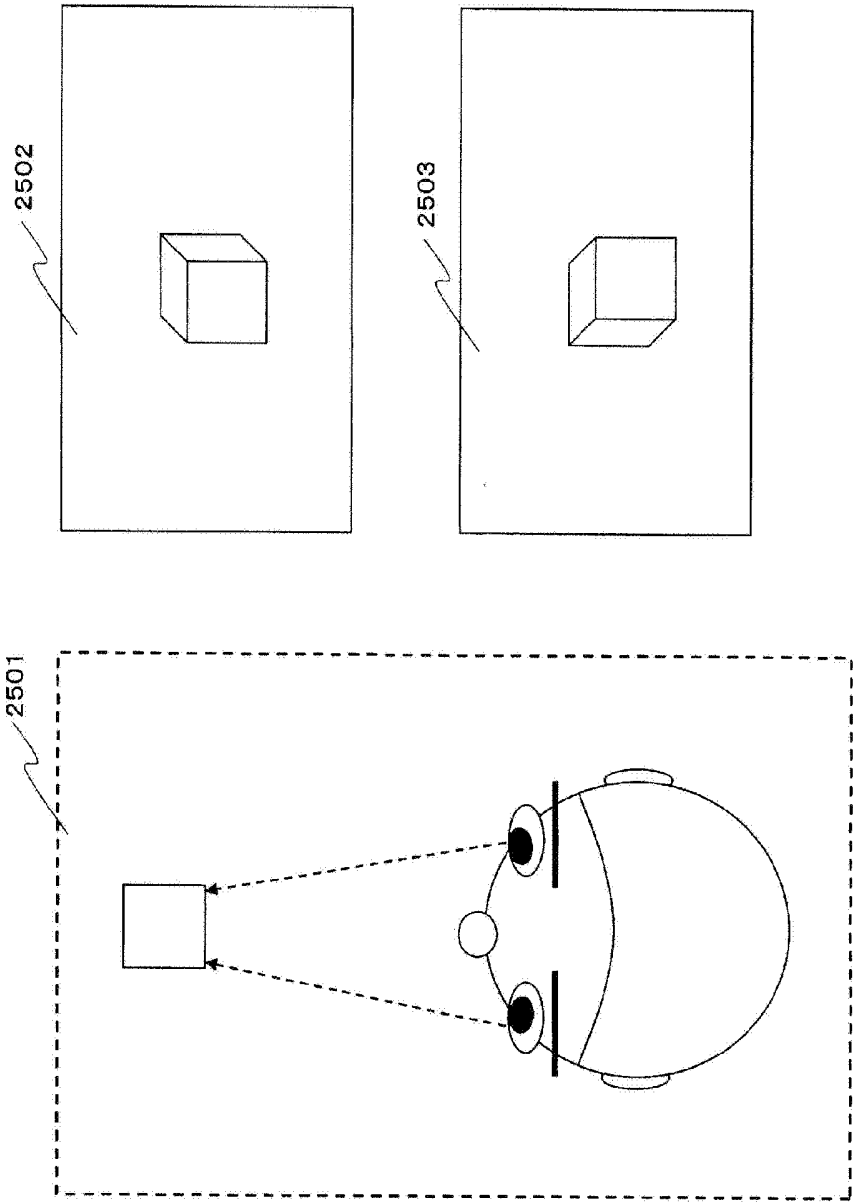


FIG. 80

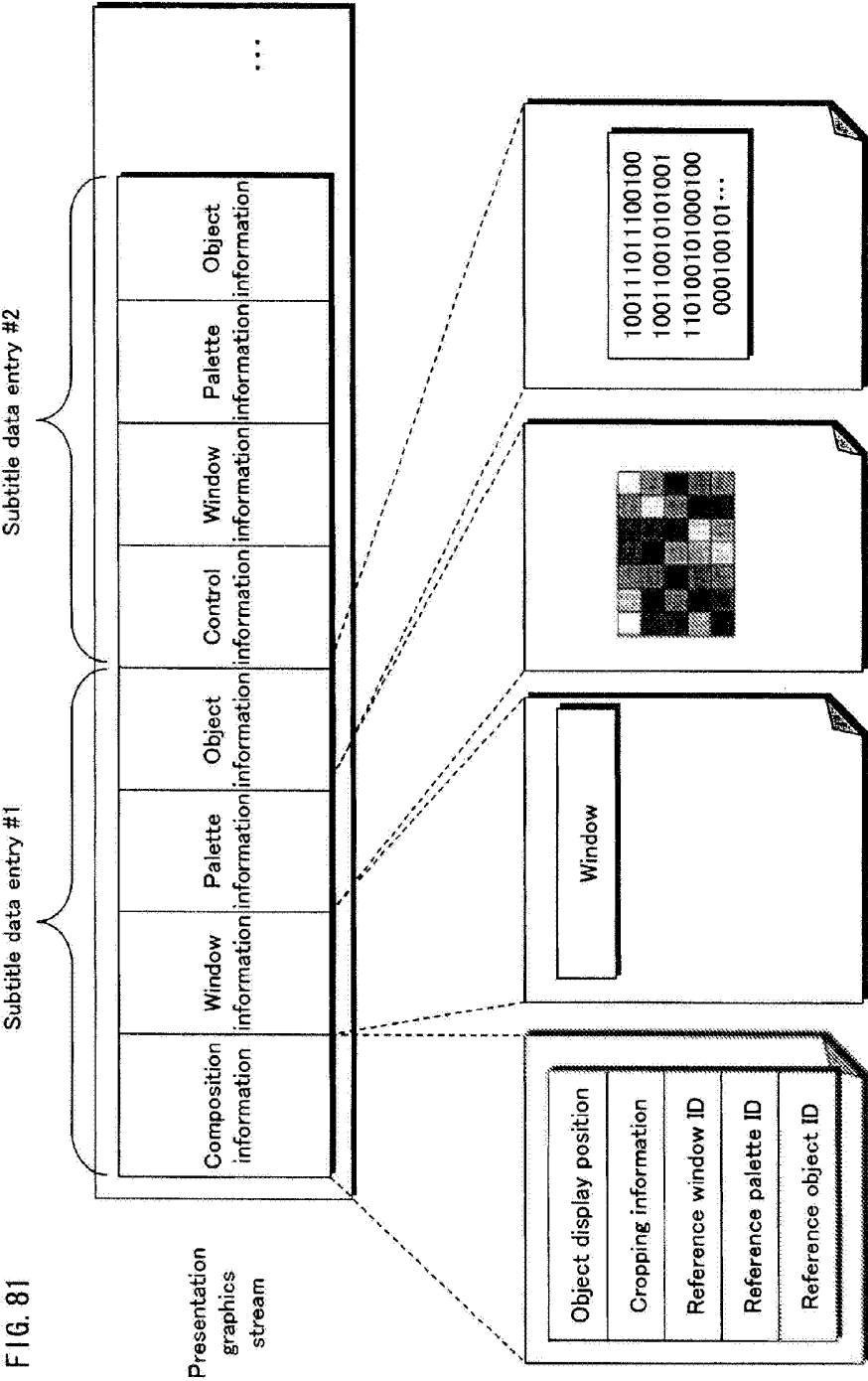


FIG. 82

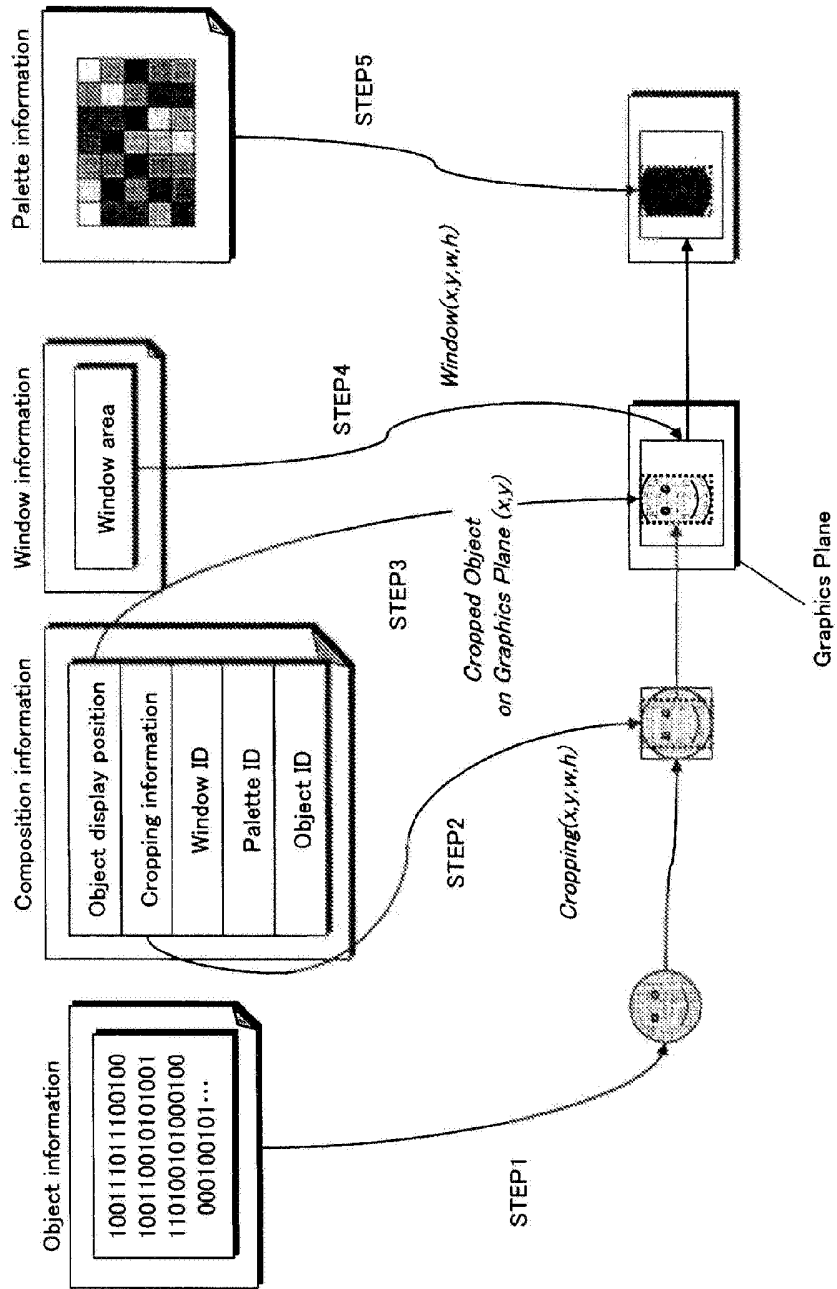


FIG. 83

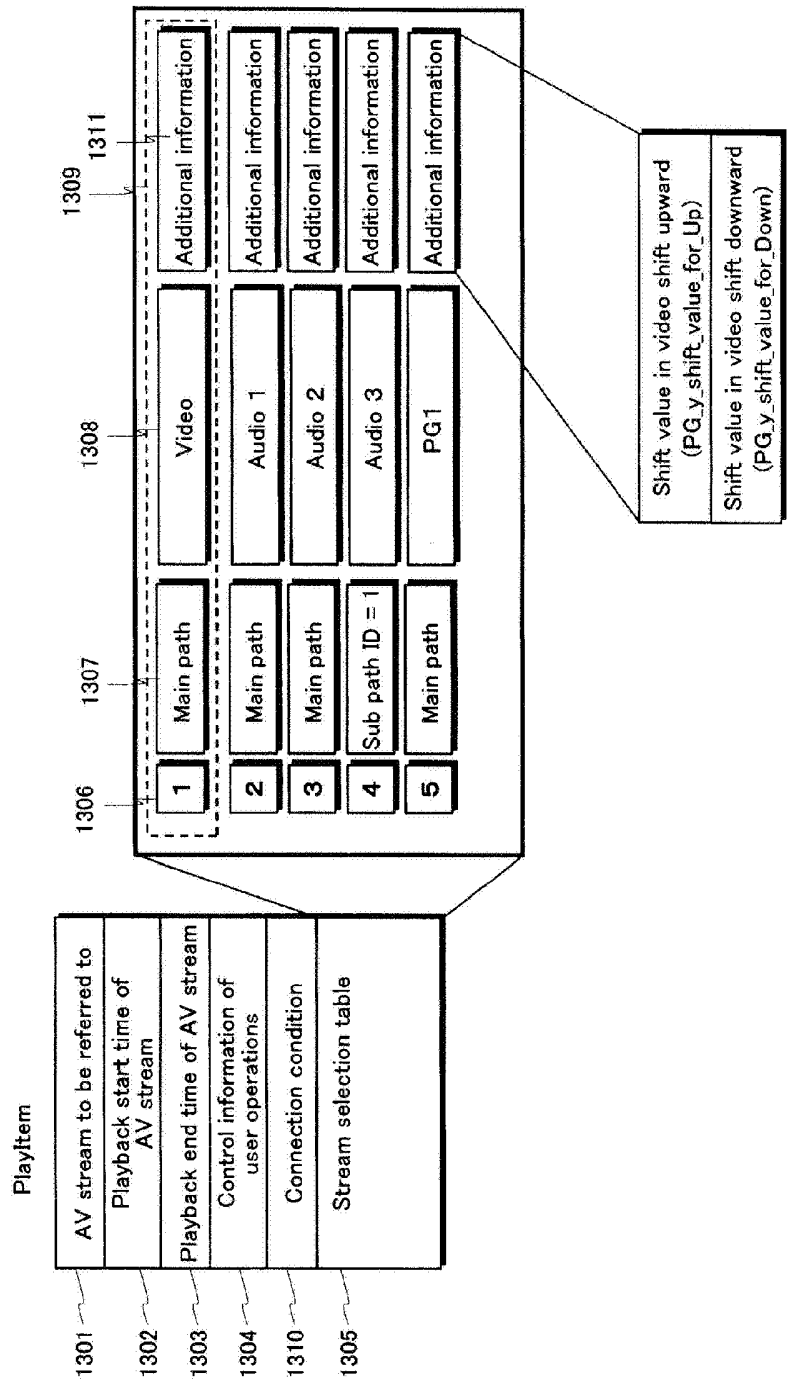


FIG. 84

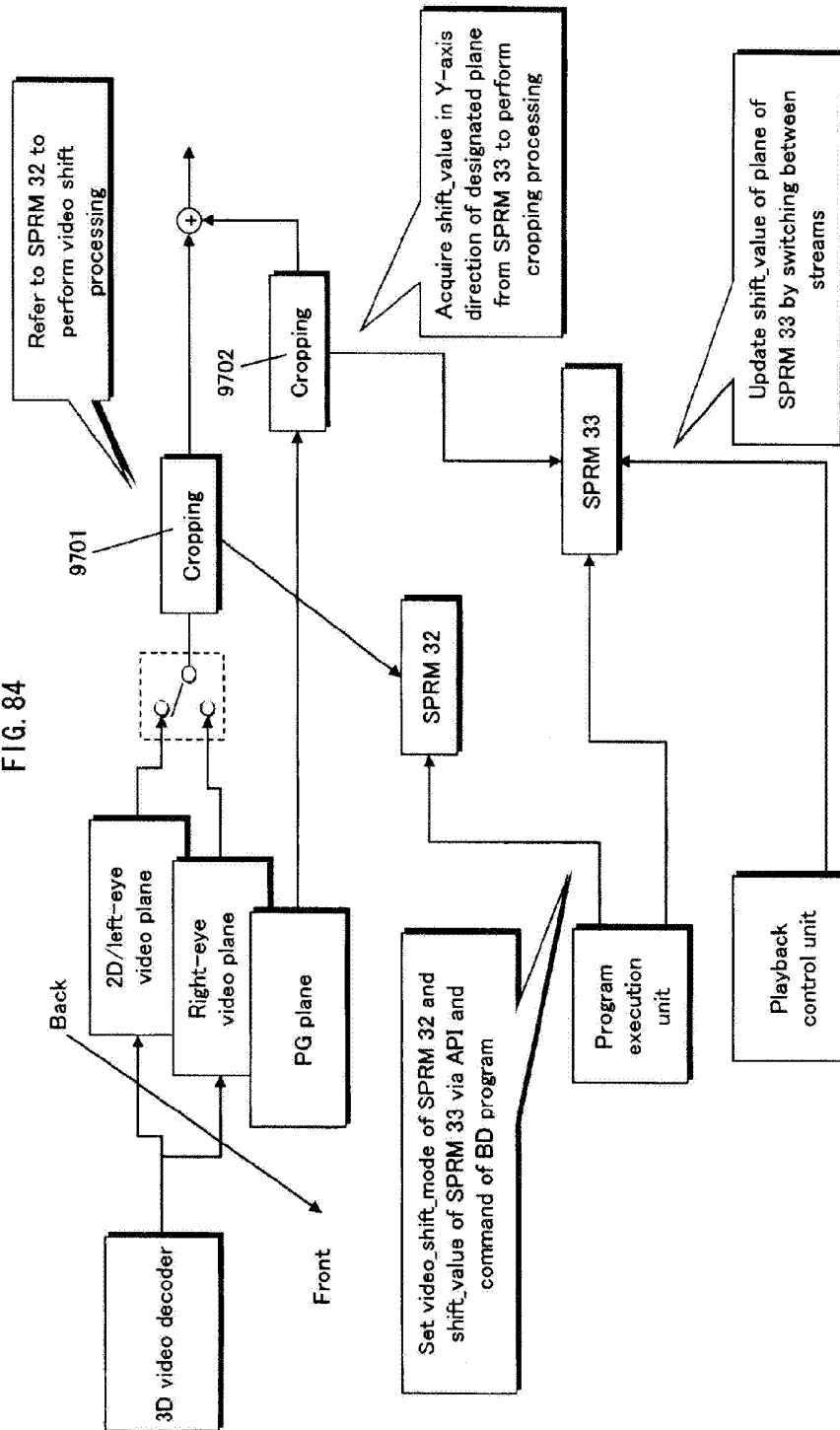


FIG. 85

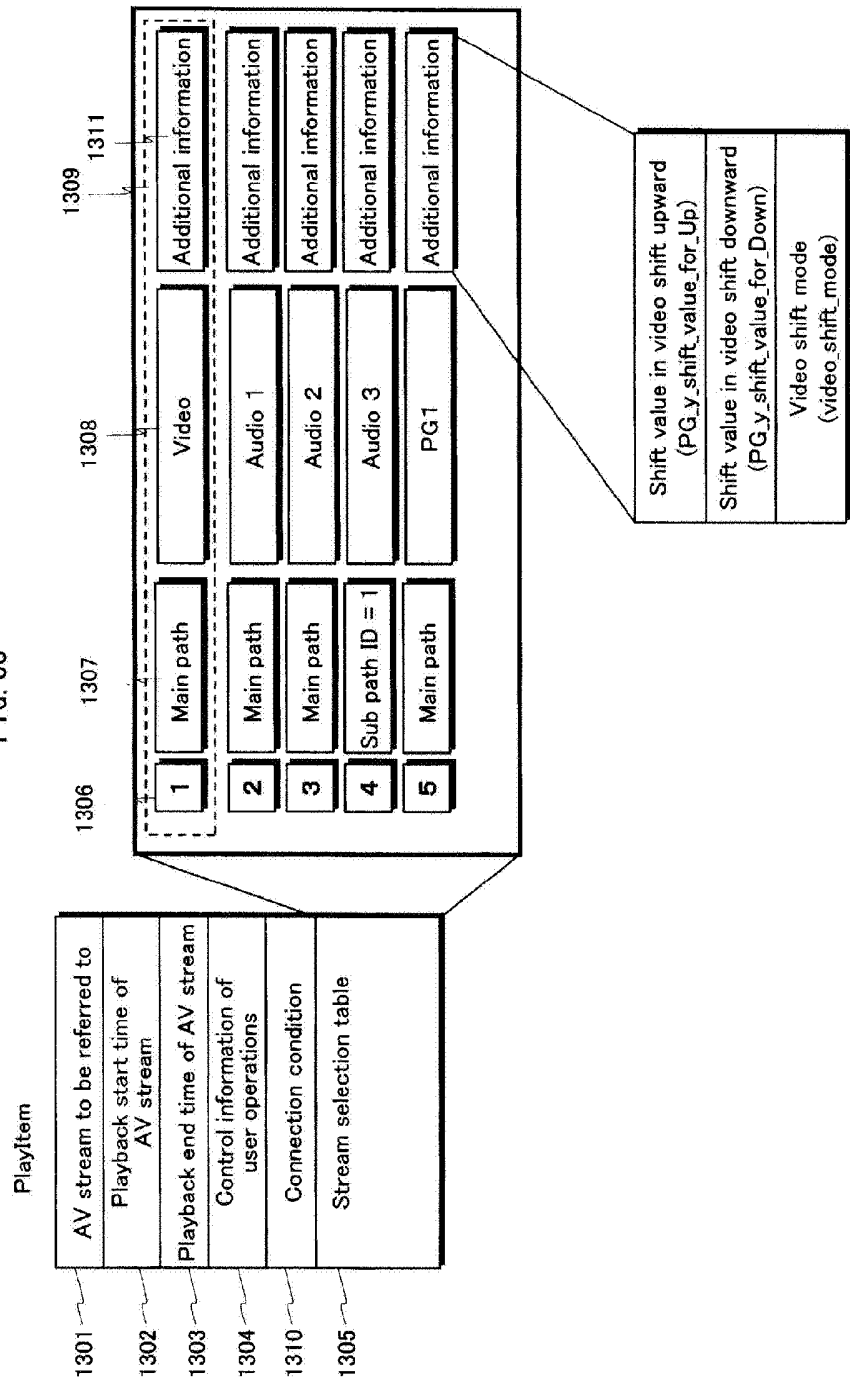


FIG. 86

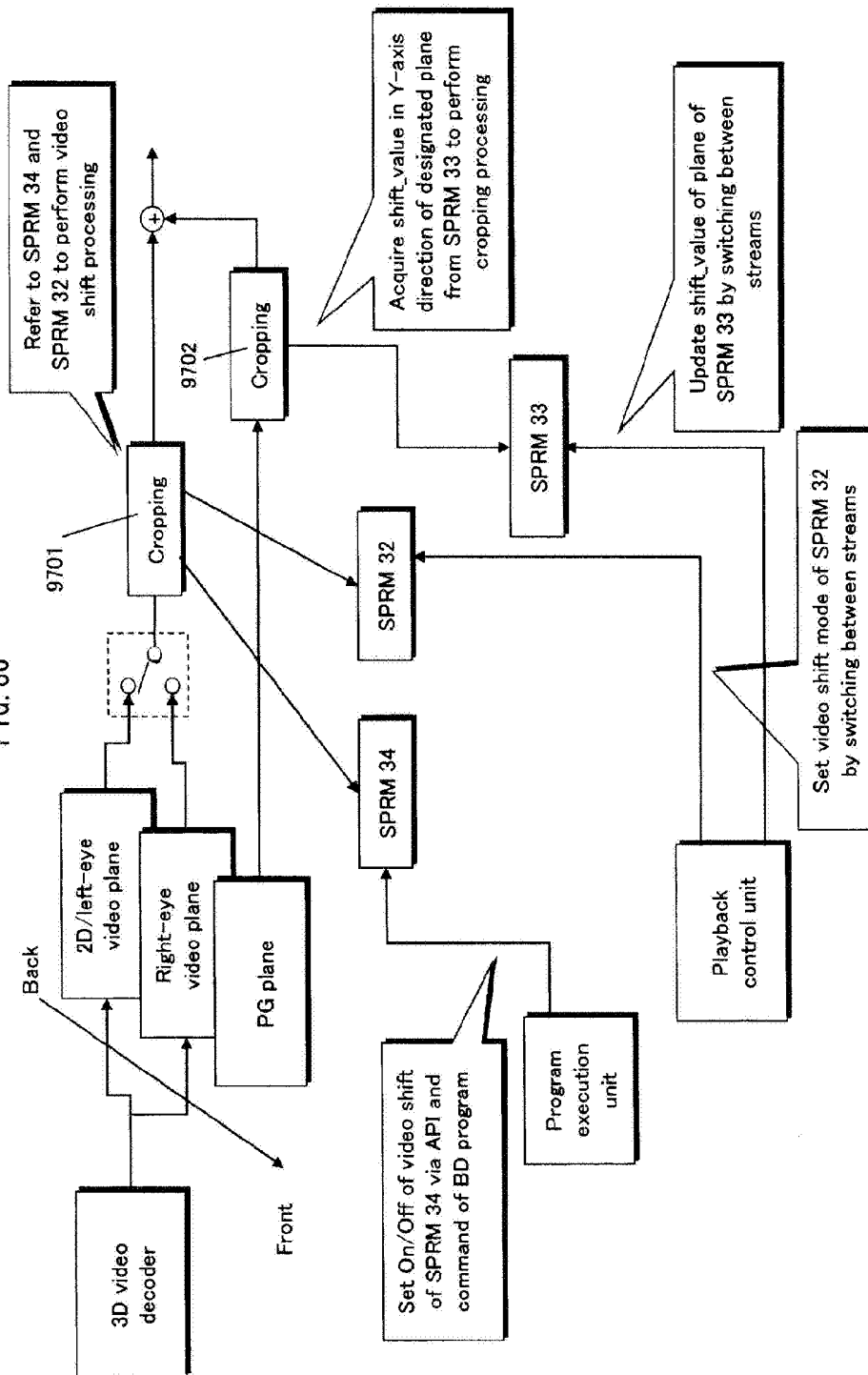


FIG. 87

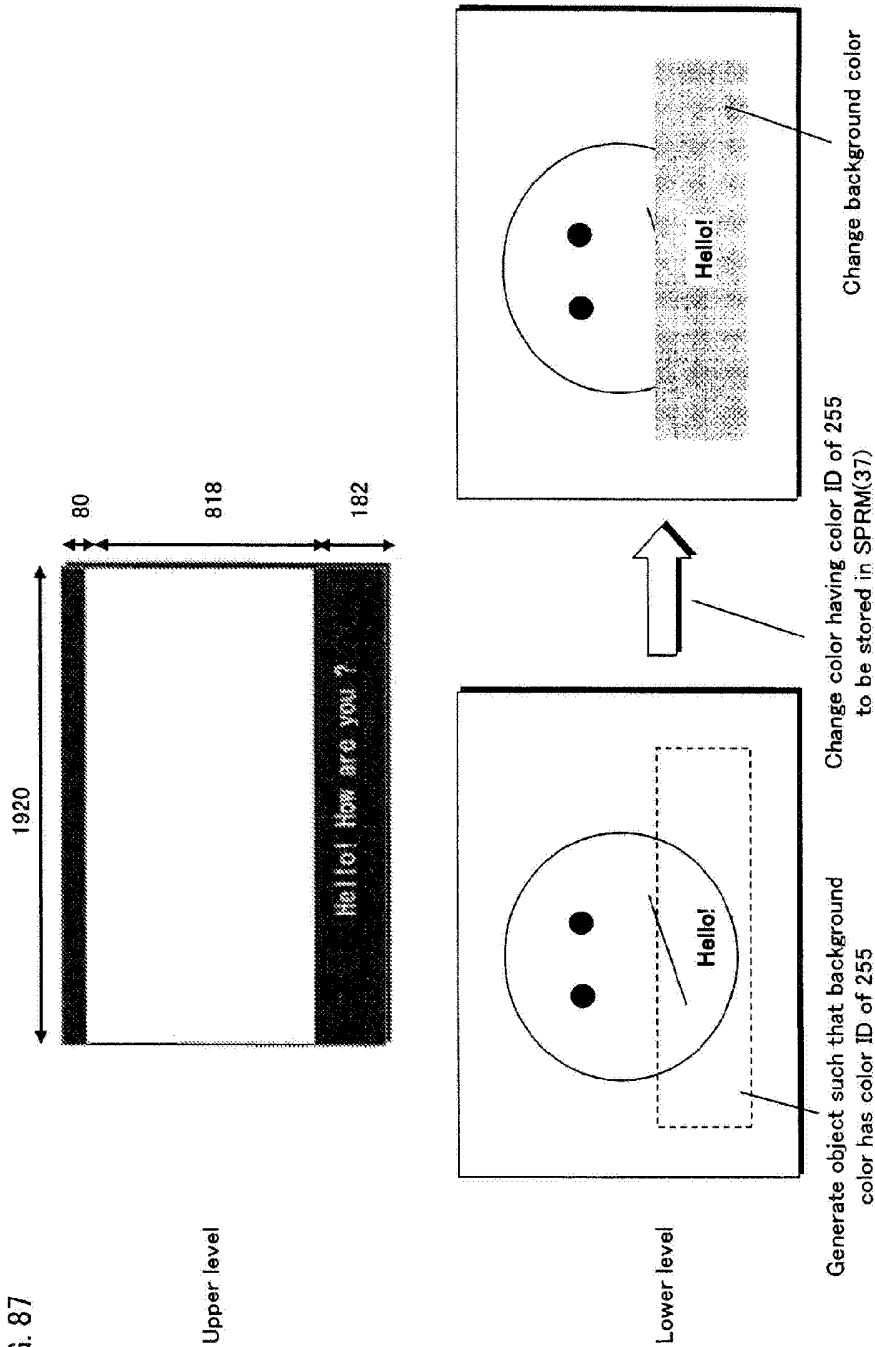


FIG. 88A

OK: Each extent includes at least one entry

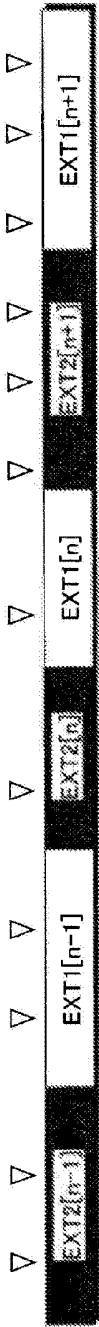


FIG. 88B

NG: Not all of extents include at least one entry

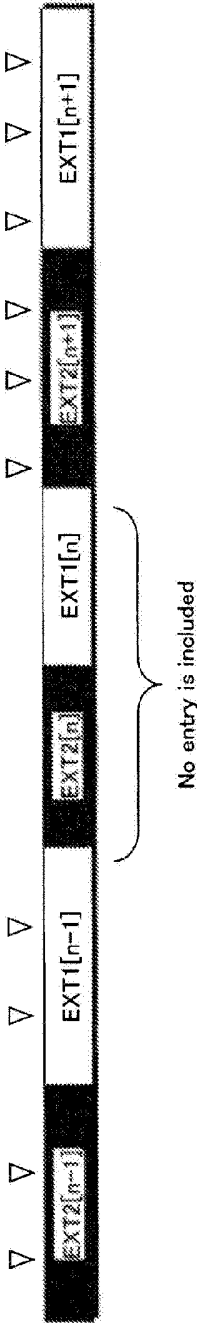


FIG.89A

Offset metadata		
PTS	Offset information of OffsetID = 0	Offset information of offset_id = 1
180000	+1	+2
270000	+3	+3
360000	-1	-4
450000	+5	+2

FIG.89B

No. of bits	
Offset_metadata{}	
length	32
number_of_offset_entries	32
for(i=0; i<number_of_offset_entries; i++) {	
PTS	32
for(offset_id=0; offset_id<number_of_offset_items; offset_id++) {	
offset_direction[offset_id]	1
offset_value[offset_id]	8
}	
}	

FIG.90A

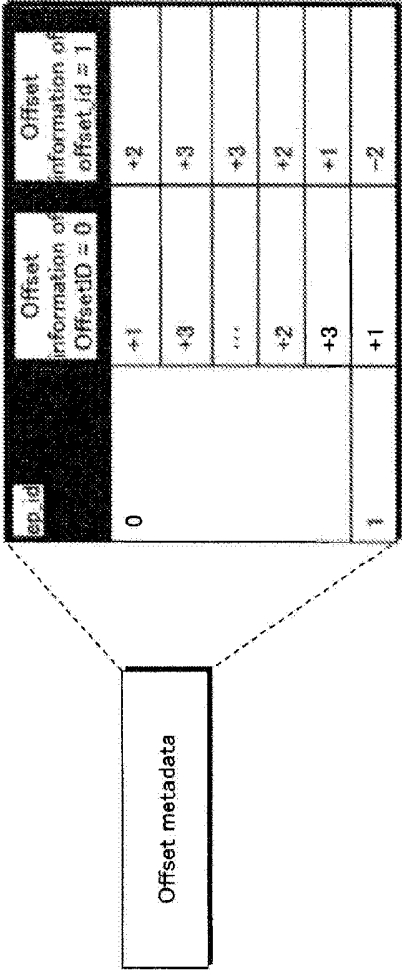


FIG.90B

No. of bits	
Offset_metadata[i]	
length	32
for(ep_id=0; ep_id < number_of_entry_points; ep_id++){	
number_of_frames;	8
number_of_offset_items;	6
for(i=0; i<number_of_frames; i++){	
for(offset_id=0; offset_id<number_of_offset_items; offset_id++){	
offset_direction[i][offset_id]	1
offset_value[i][offset_id]	6
}	
}	
}	

FIG. 91A

Offset_metadata{}		No. of bits
length		32
for(playitem_id=0; playitem_id<number_of_playitem; playitem_id++)		
number_of_offsets[playitem_id]		16
number_of_offset_id[playitem_id]		4
for(j=0; j<number_of_offsets[playitem_id]; j++)		6
offset_frame_number[j]		16
offset_frame_duration[j]		5
number_of_suboffsets[j]		5
for(j=0; j<number_of_suboffsets[j]; j++)		
for(offset_id=0; offset_id<number_of_offset_id; offset_id++)		
offset_direction[j][offset_id]		1
offset_value[j][offset_id]		6
}		
}		
}		
}		

FIG. 91B

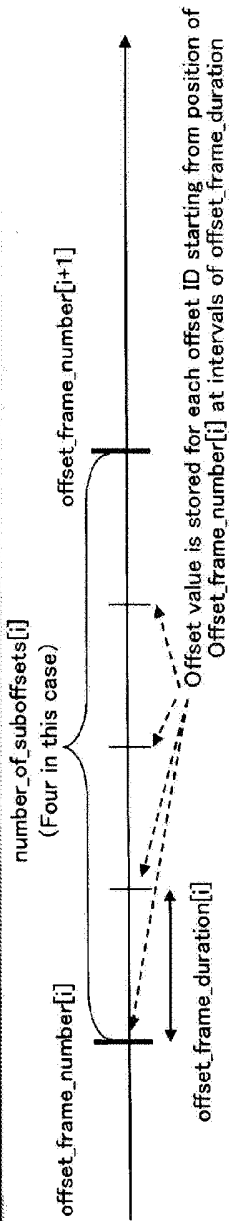


FIG. 92A

	No. of bits
Offset_metadata{	
length	32
for(playitem_id=0; playitem_id<number_of_playitem; playitem_id++){	
is_same_as_previous_playitem;	
if(is_same_as_previous_playitem==0){	
number_of_offsets[playitem_id]	16
number_of_offset_id[playitem_id];	4
for(i=0; i<number_of_offsets[playitem_id]; i++){	6
offset_frame_number[i]	16
offset_frame_duration[i]	5
number_of_suboffsets[i]	5
for(j=0; j<number_of_suboffsets[i]; j++){	
for(offset_id=0; offset_id<number_of_offset_id; offset_id++){	
offset_direction[i][offset_id]	1
offset_value[i][offset_id]	6
}	
}	
}	
}	
}	

playlist

FIG. 92B

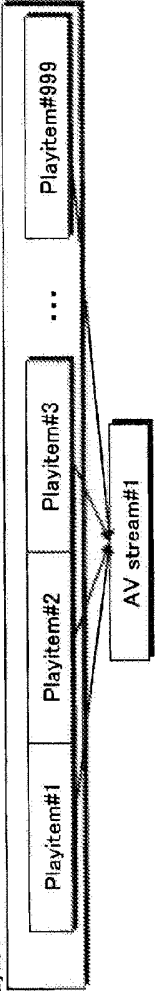
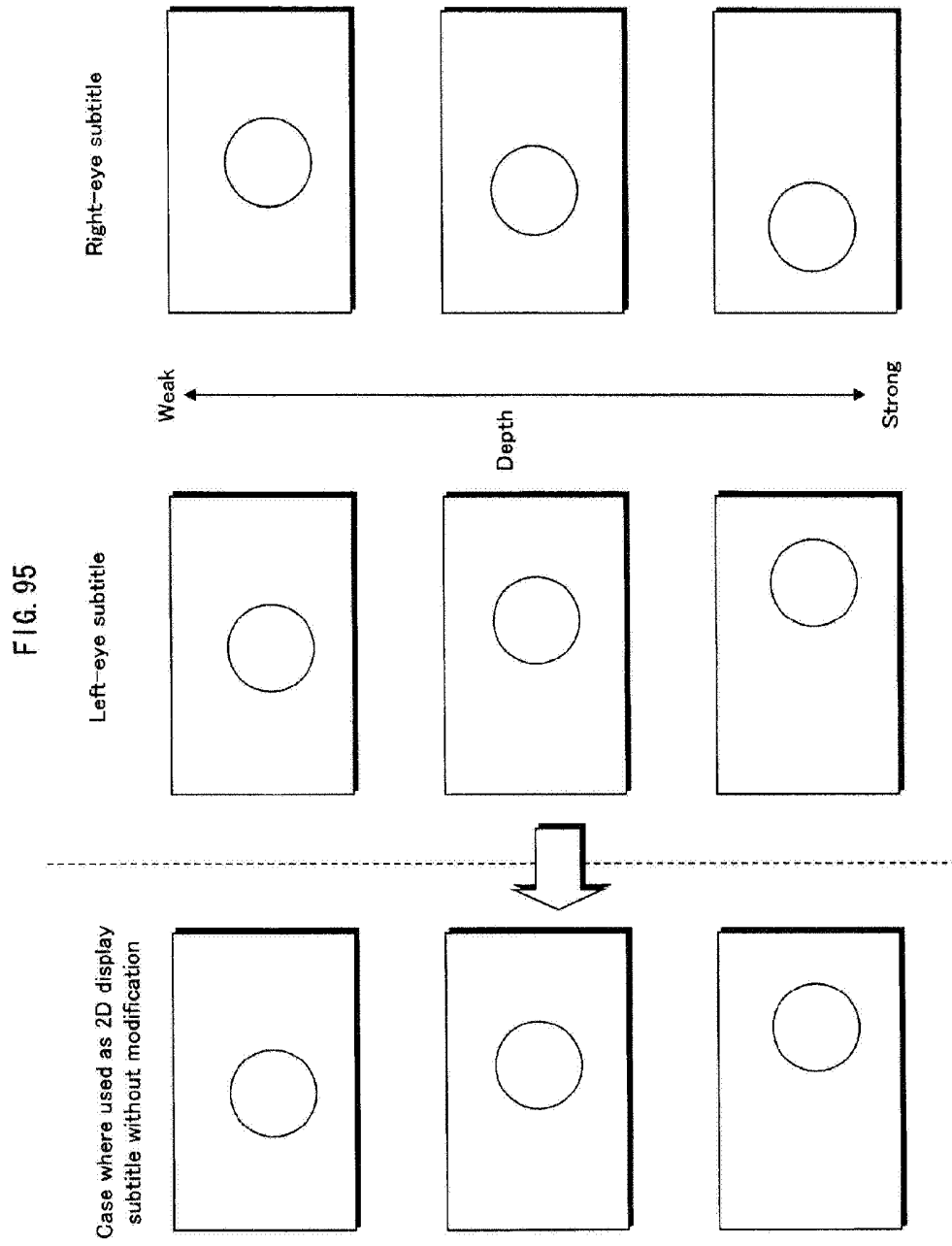


FIG. 93

No. of bits	
Offset_metadata{X	
length	32
for(playitem_id=0; playitem_id<number_of_playitem; playitem_id++)	
ref_playitem_id_of_same_offset_metadata;	16
if(ref_playitem_id_of_same_offset_metadata == 0xFFFF)	
number_of_offsets[playitem_id]	16
number_of_offset_id[playitem_id];	4
for(i=0; i<number_of_offsets[playitem_id]; i++)	6
offset_frame_number[i]	16
offset_frame_duration[i]	5
number_of_suboffsets[i]	5
for(j=0; j<number_of_suboffsets[i]; j++)	
for(offset_id=0; offset_id<number_of_offset_id; offset_id++)	
offset_direction[i][offset_id]	1
offset_value[i][offset_id]	6
}	
}	
}	
}	
}	

FIG. 94

Offset_metadata{X}		No. of bits
length		32
number_of_offset_entries;		16
for(playitem_id=0; playitem_id<number_of_playitem; playitem_id++) {		
ref_offset_block_id;		16
start_address		32
}		
for(offset_block_id =0; offset_block_id <number_of_offset_blocks; offset_block_id++) {		
number_of_offsets[offset_block_id]		16
number_of_offset_id[offset_block_id]		4
for(i=0; i<number_of_offsets[offset_block_id]; i++) {		6
offset_frame_number[i]		16
offset_frame_duration[i]		5
number_of_suboffsets[i]		5
for(j=0; j<number_of_suboffsets[i]; j++) {		
for(offset_id=0; offset_id<number_of_offset_id; offset_id++) {		
offset_direction[i][offset_id]		1
offset_value[i][offset_id]		6
}		
}		
}		
}		



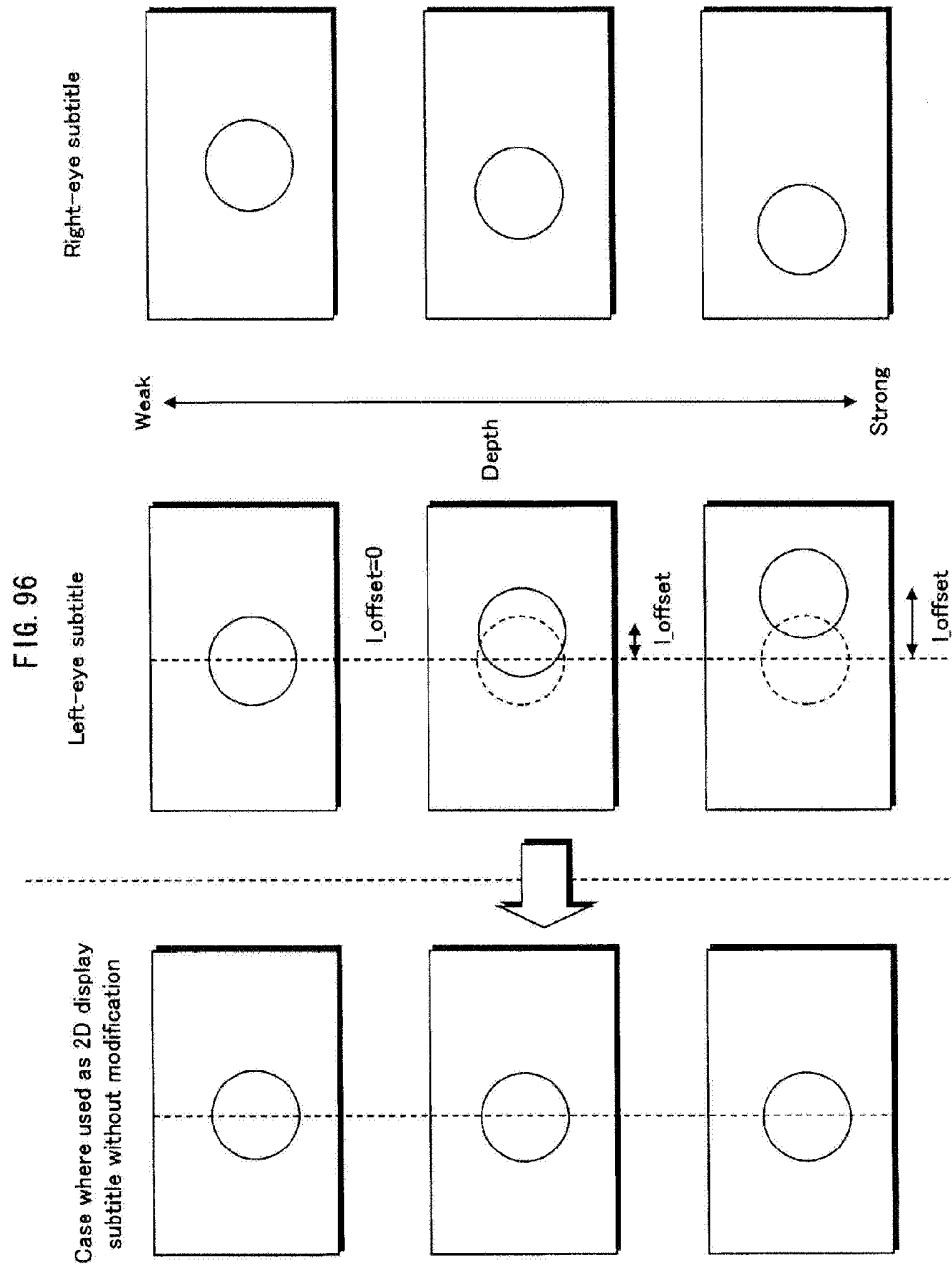


FIG. 97A

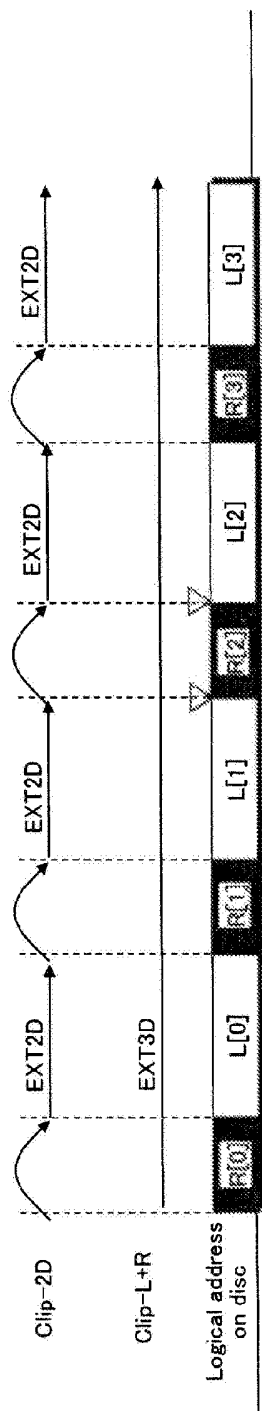


FIG. 97B

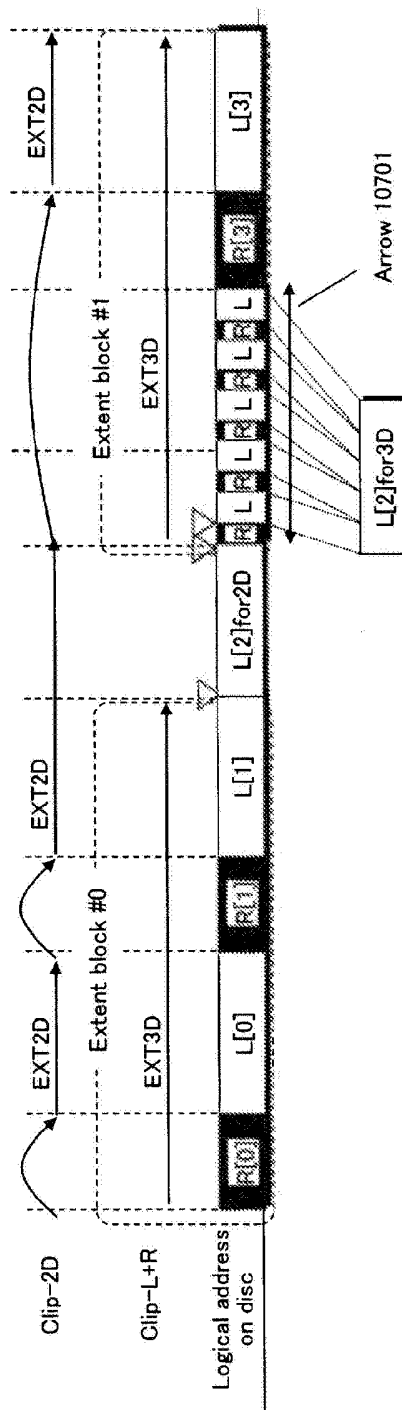


FIG. 98

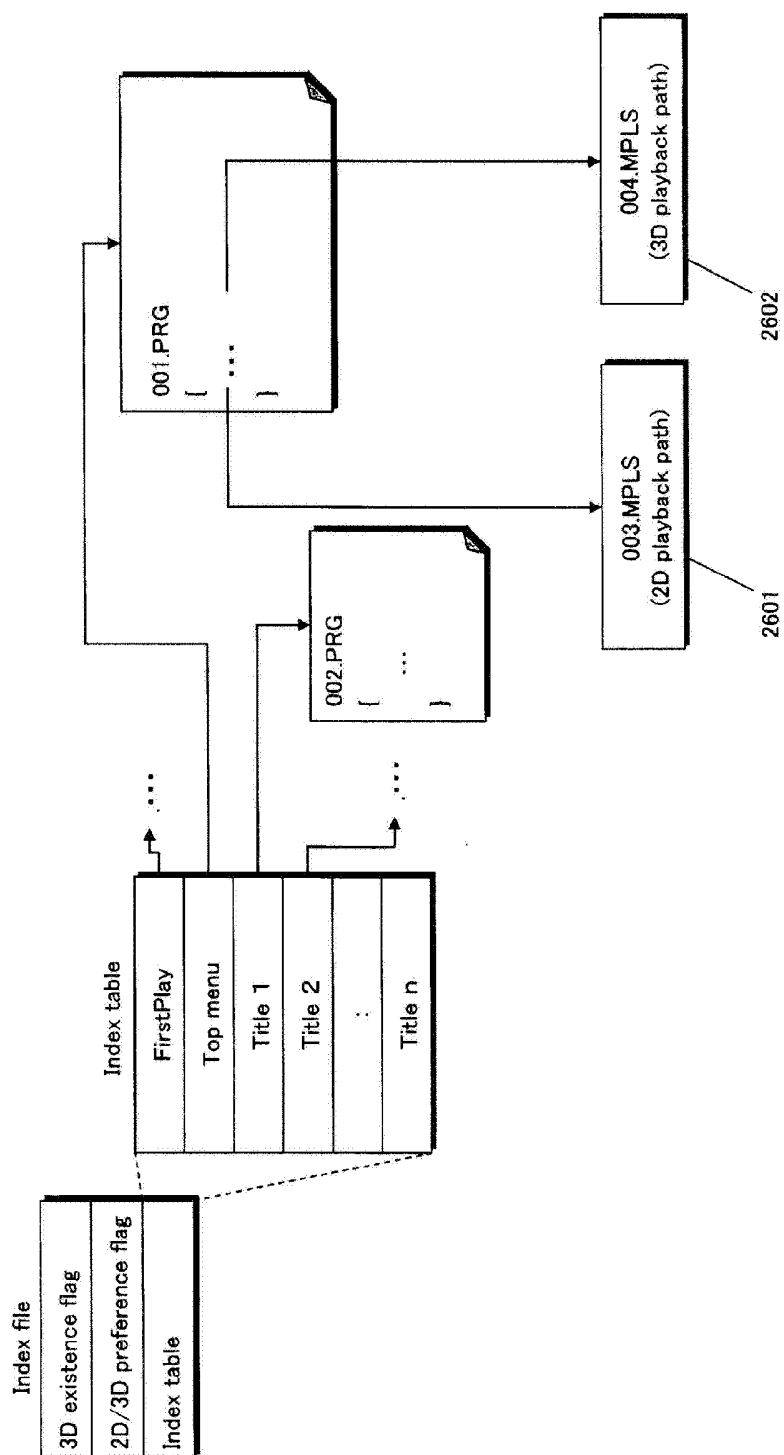


FIG. 99

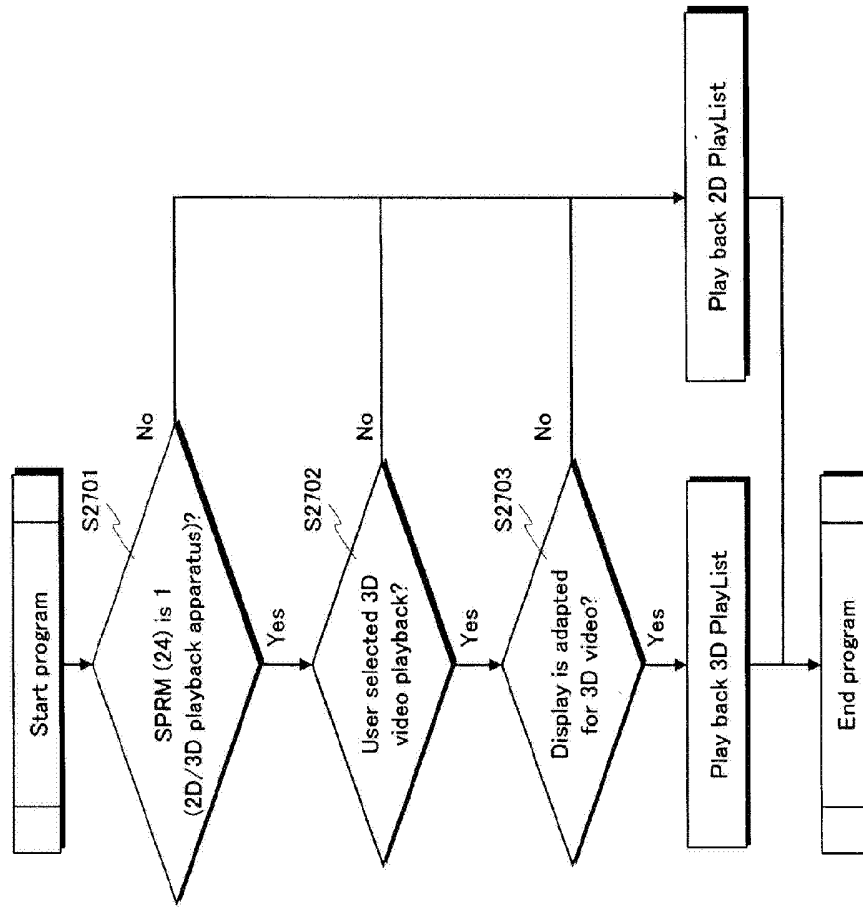


FIG. 100

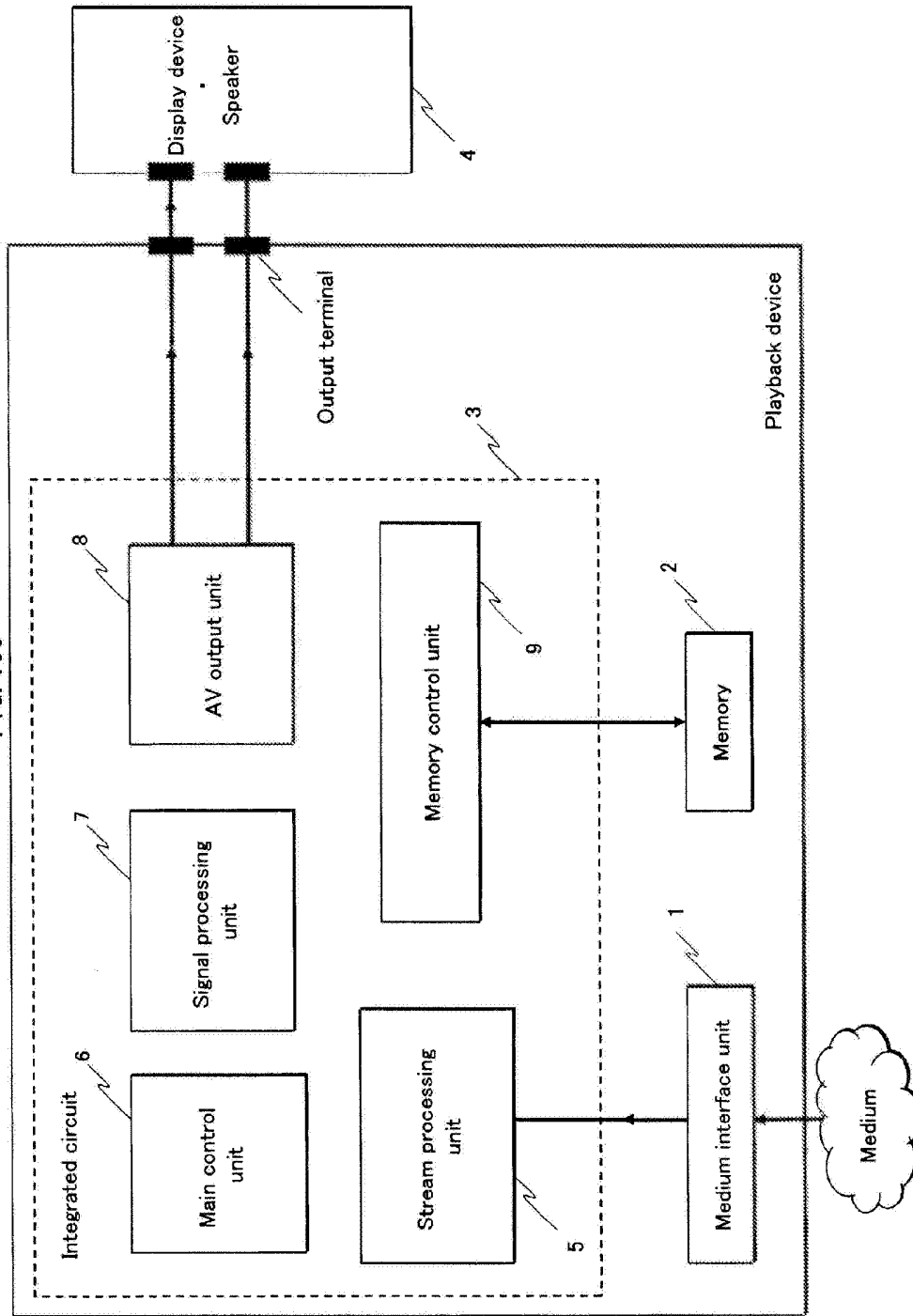


FIG. 101

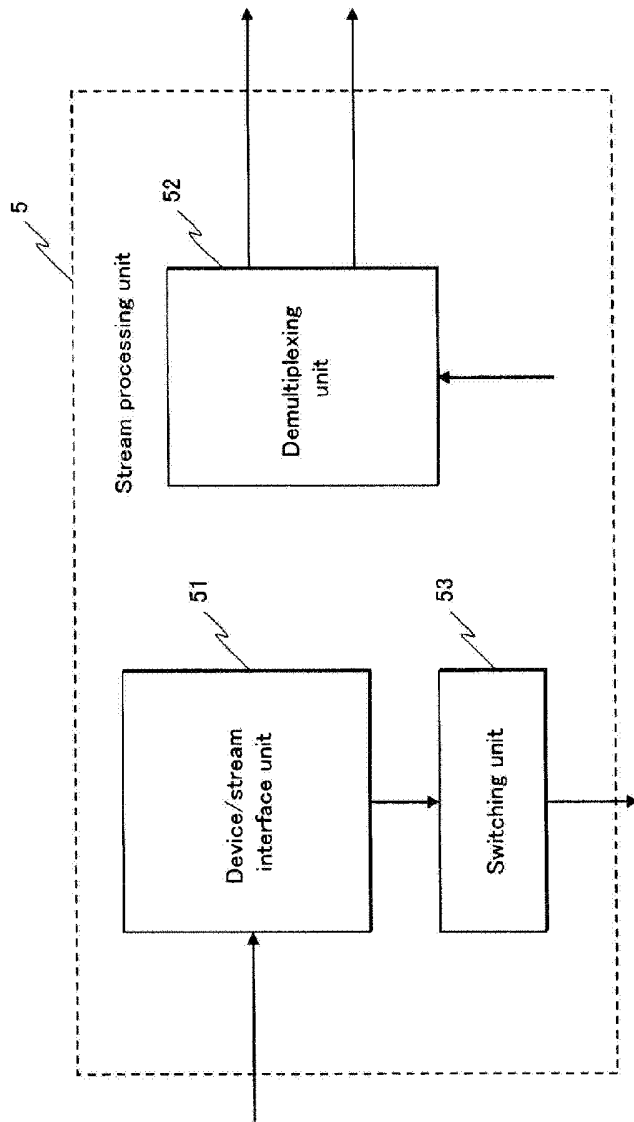
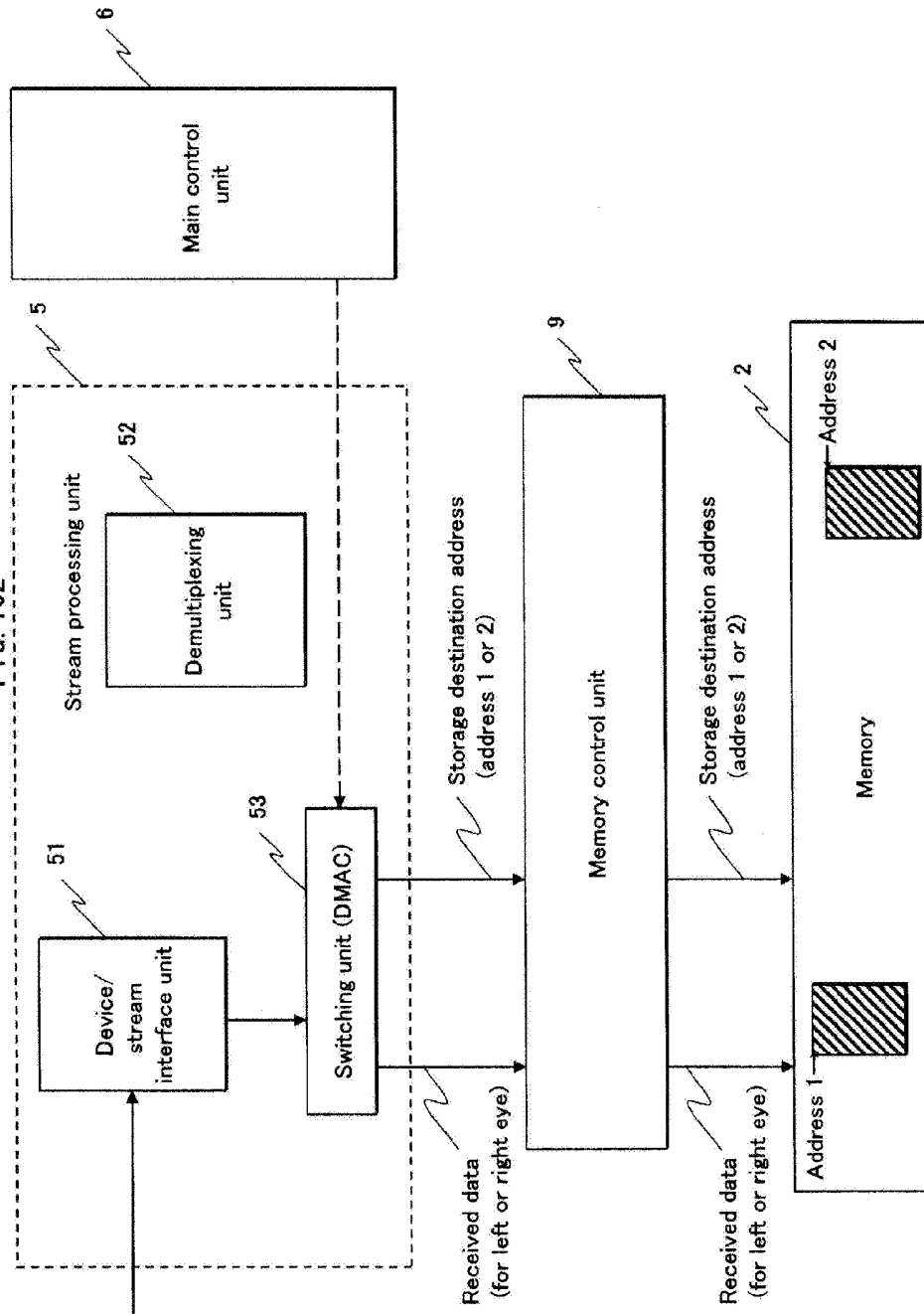


FIG. 102



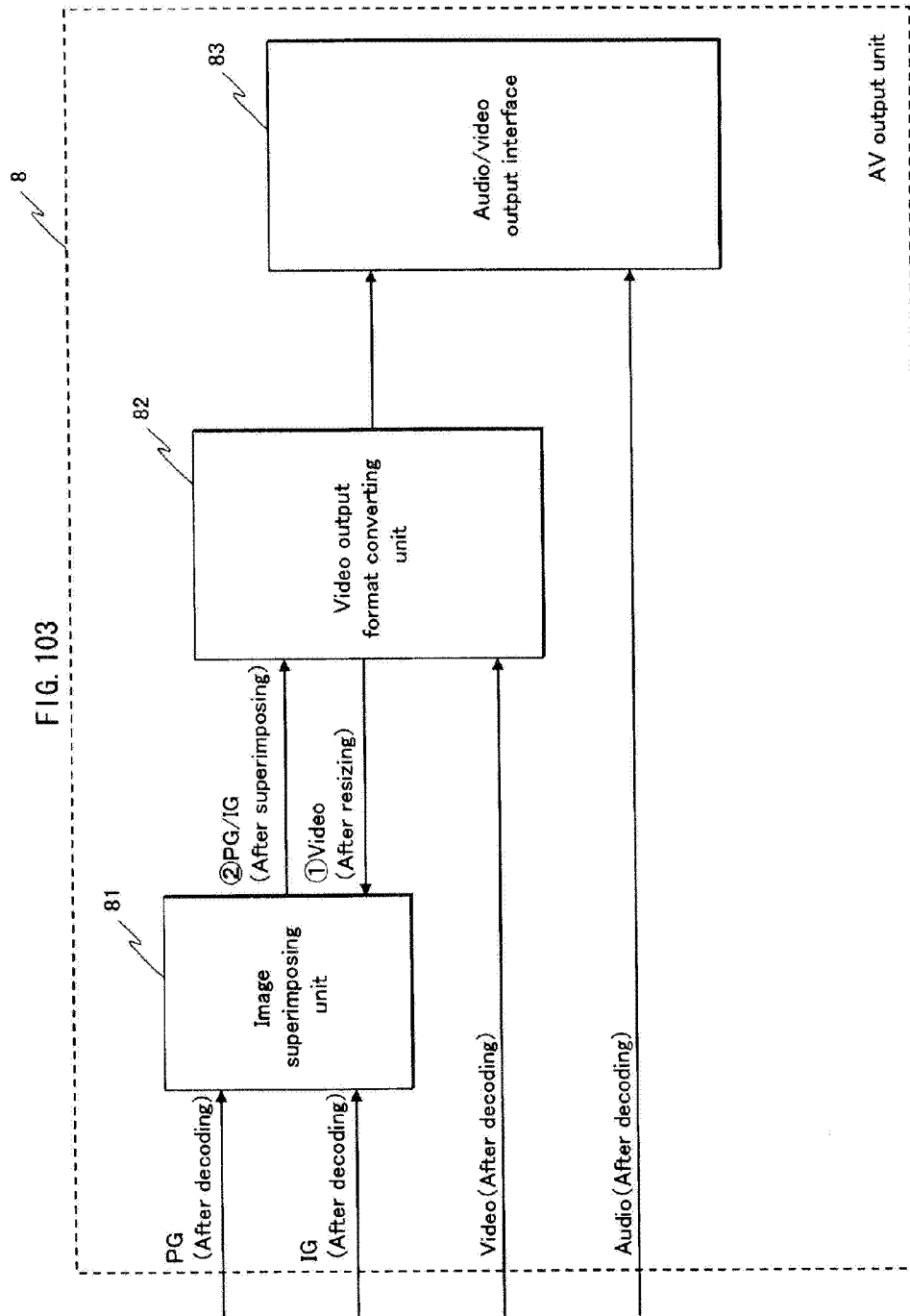


FIG. 104

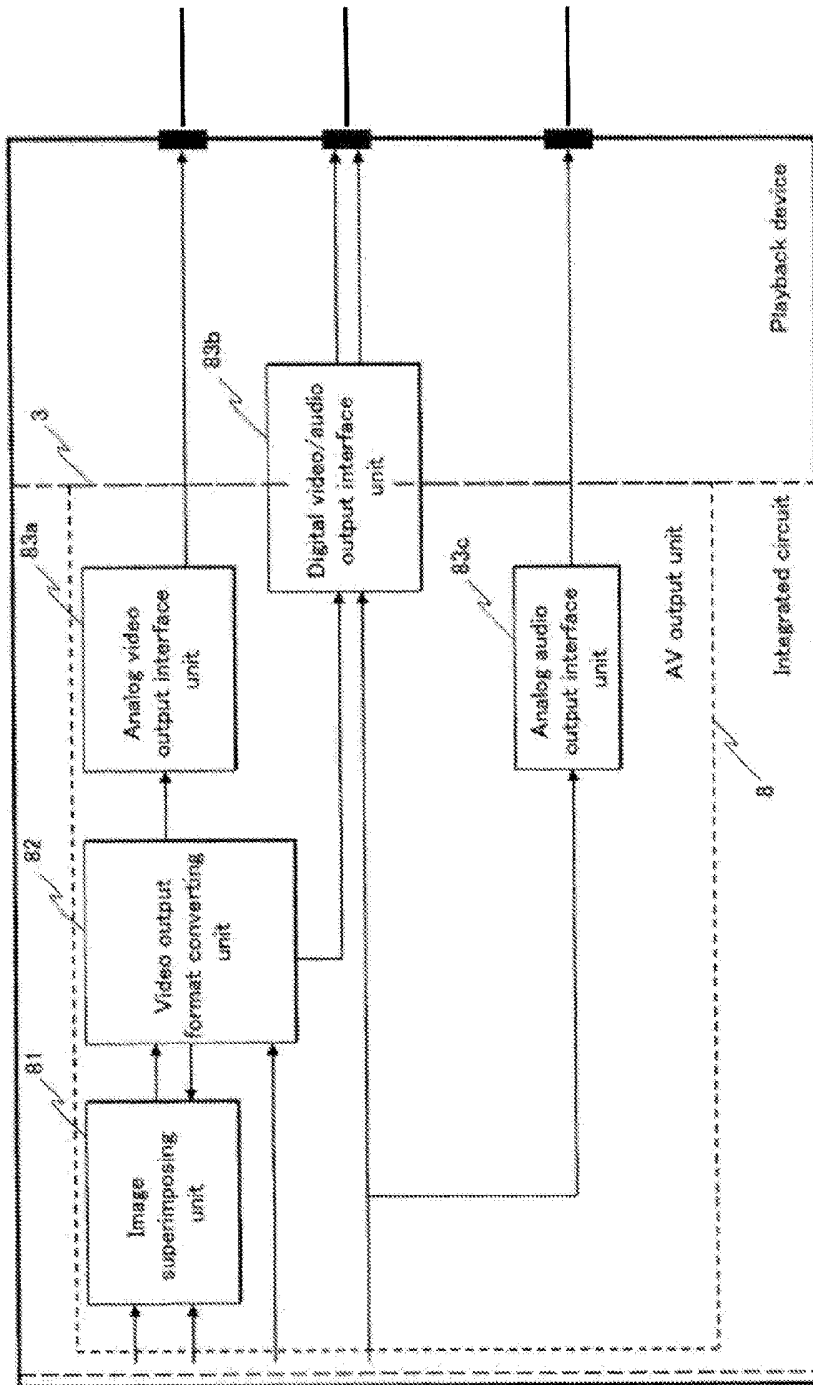


FIG. 105

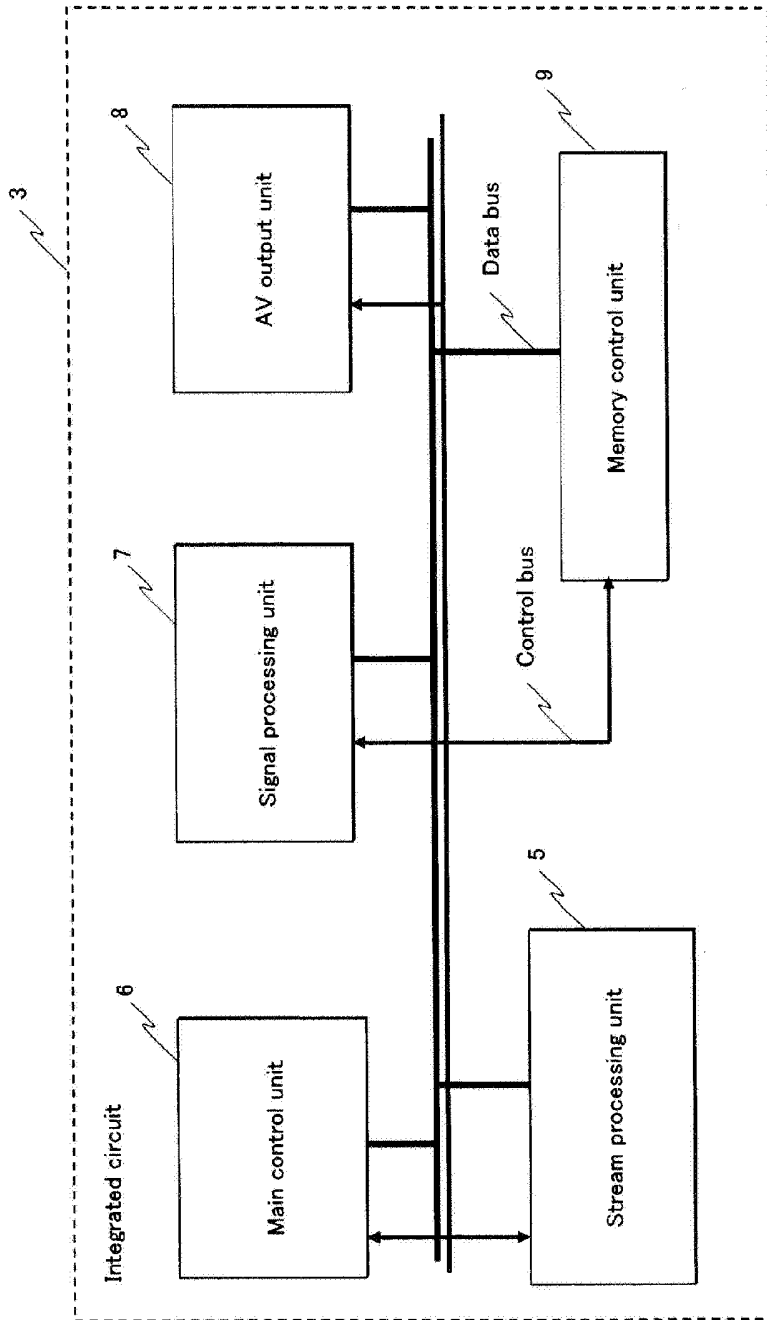
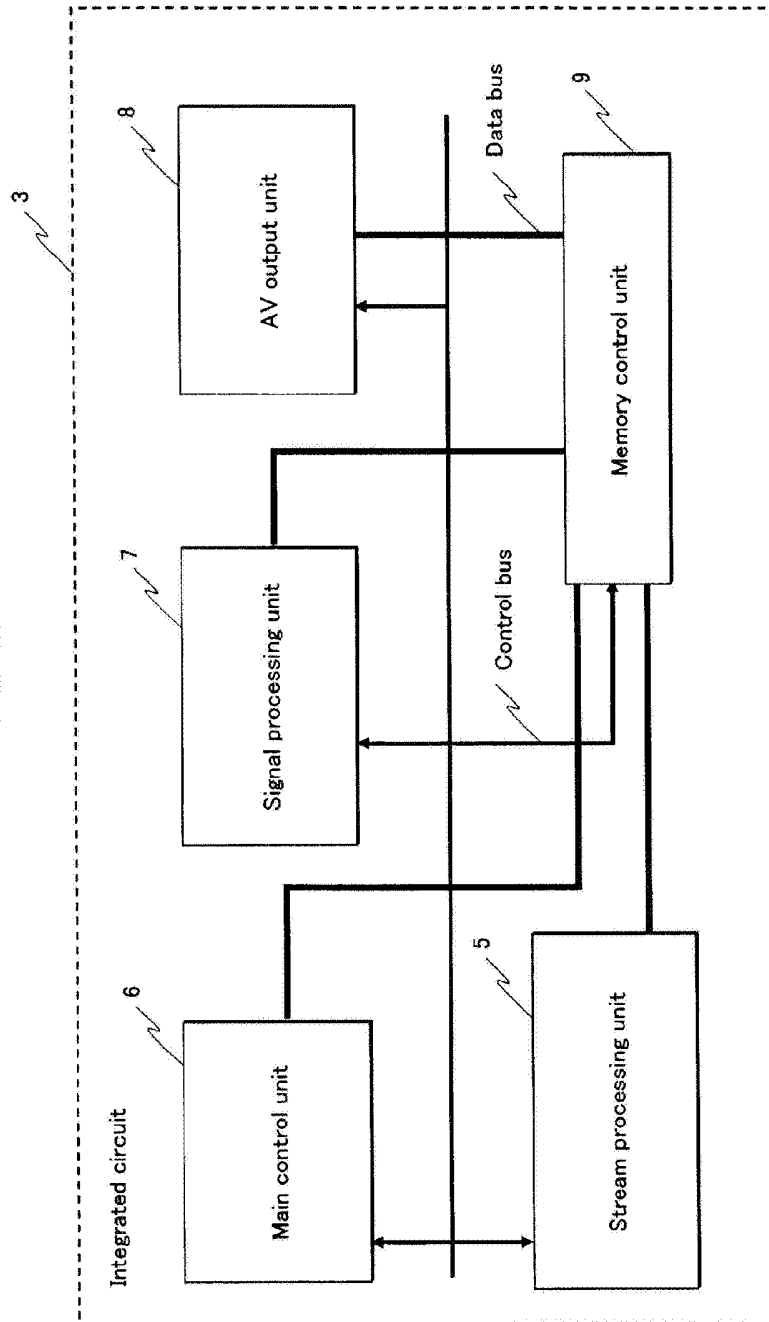


FIG. 106



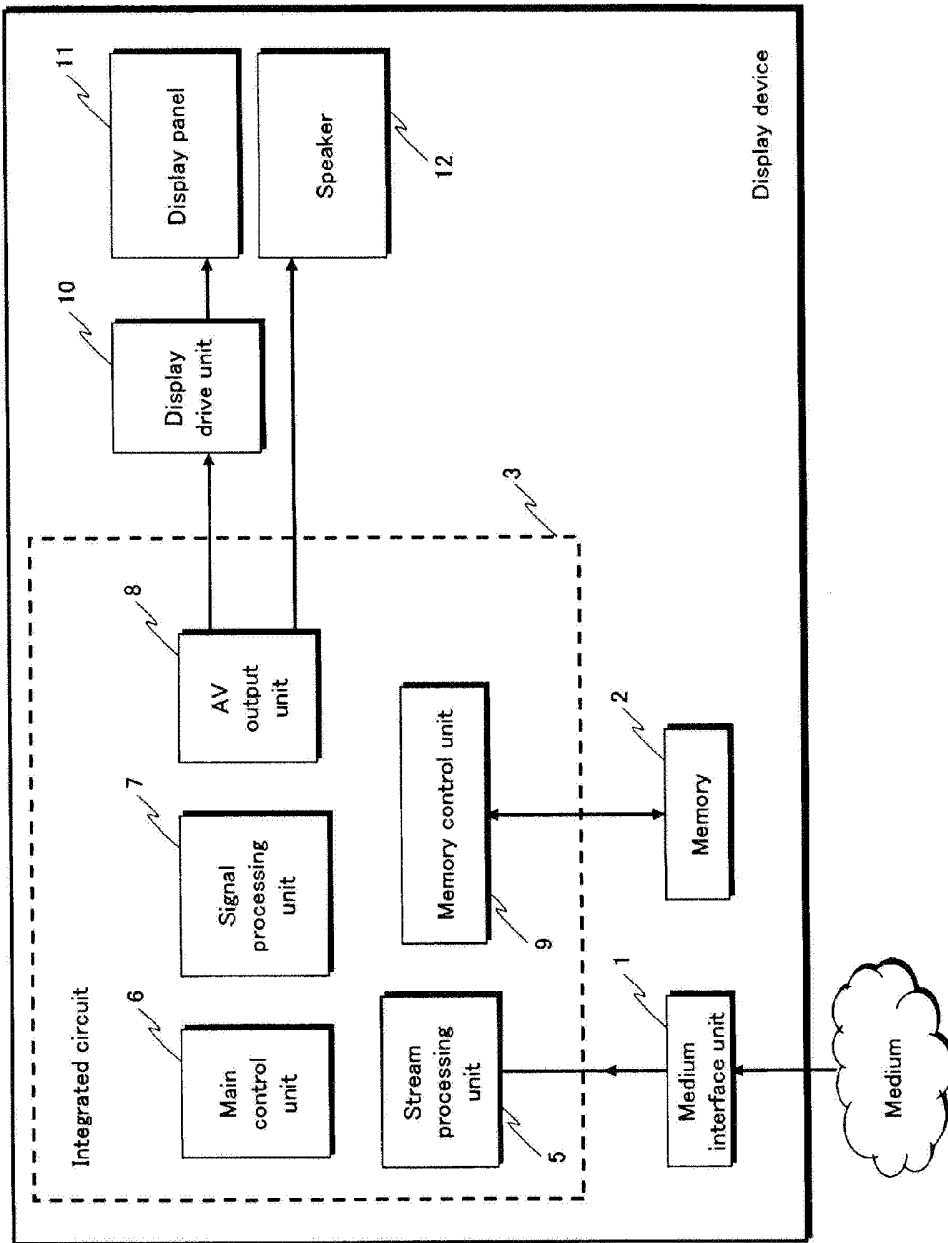


FIG. 107

FIG. 108

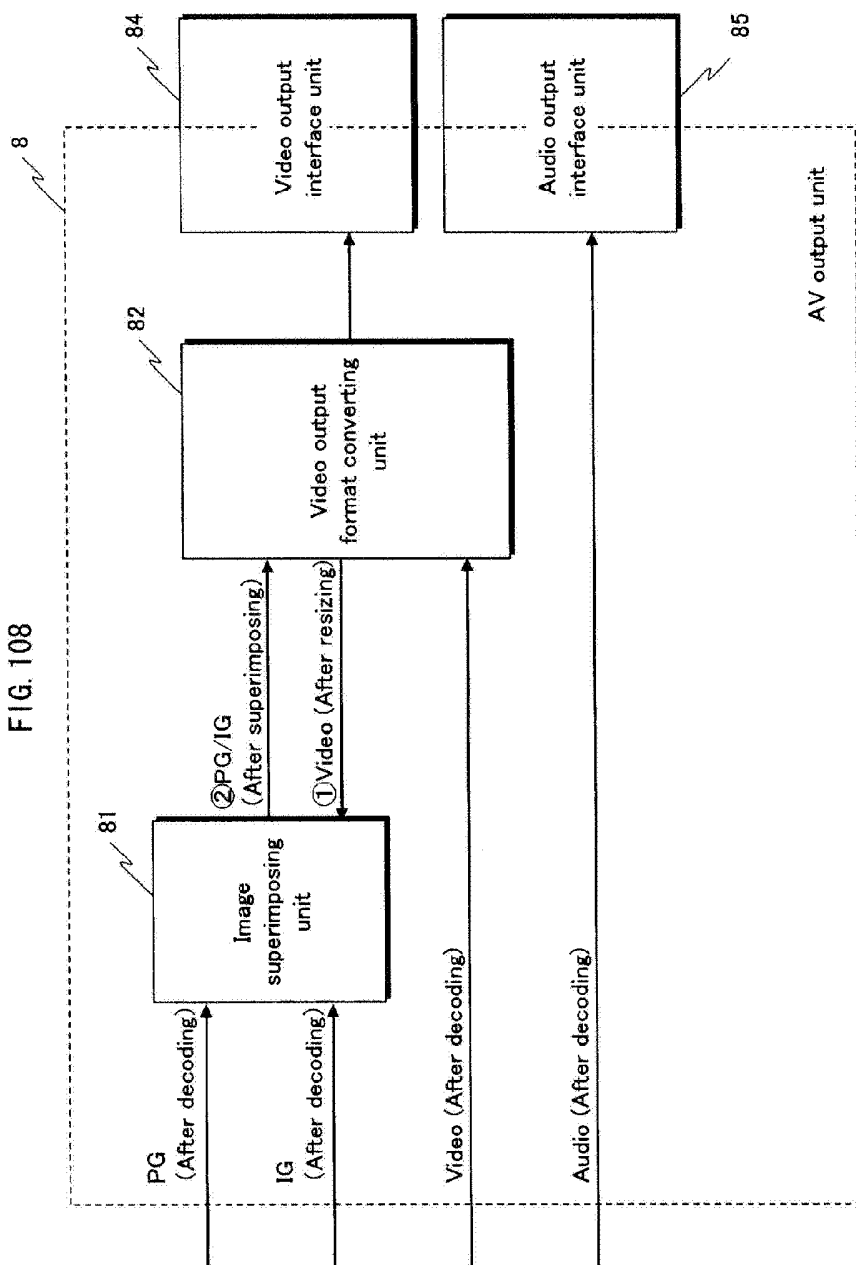


FIG. 109

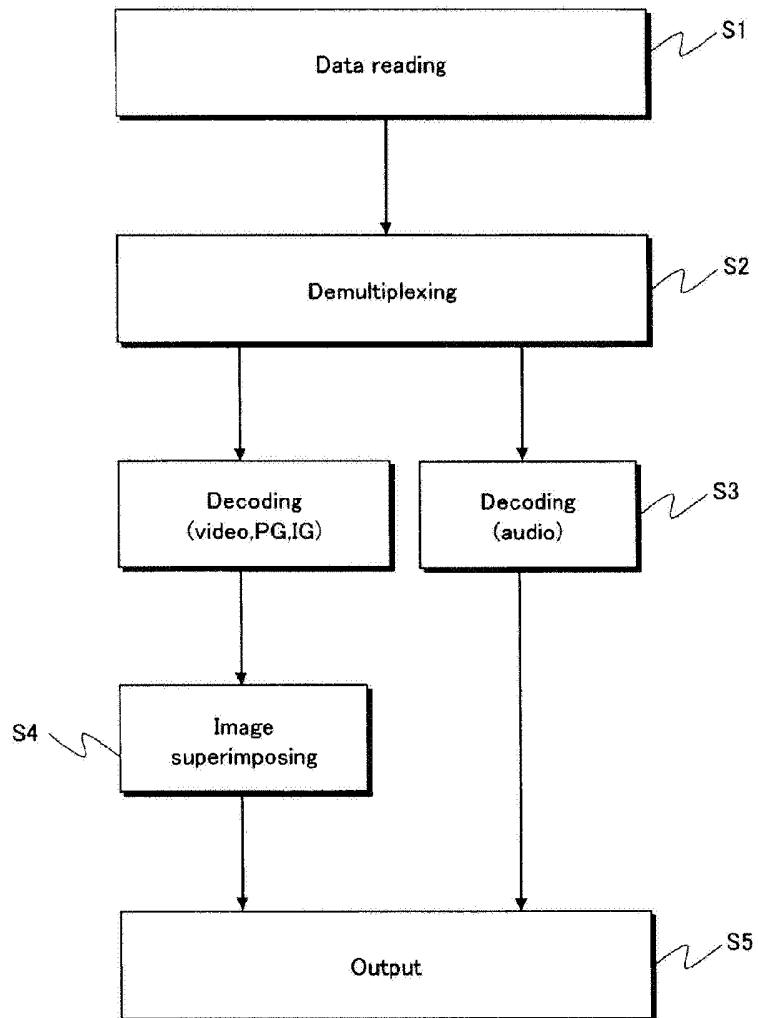


FIG. 110

