REPLACED SHEETS SG
☐ Description : paged ..... to paged .....
☑ Claim(s) : paged 257 to paged 265
☐ Drawing(s) : paged ..... to paged .....
☐ Abstract : paged ..... to paged .....

PCT/JP96/02803

DESCRIPTION

A METHOD AND AN APPARATUS FOR SYSTEM ENCODING OF BITSTREAMS TO CONNECT SEAMLESSLY THEREOF

Technical Field

The present invention relates to a method and apparatus for system encoding bitstreams to connect seamlessly thereof and, more particularly, bitstreams for use in an authoring system for variously processing a data bitstream comprising the video data, audio data, and sub-picture data constituting each of plural program titles containing related video data, audio data, and sub-picture data content to generate a bitstream from which a new title containing the content desired by the user can be reproduced, and efficiently recording and reproducing said generated bitstream using a particular recording medium.

Background Art

Authoring systems used to produce program titles comprising related video data, audio data, and sub-picture data by digitally processing, for example, multimedia data comprising video, audio,

and sub-picture data recorded to laser disk or video CD formats are currently available.

Systems using Video-CDs in particular are able to record video data to a CD format disk, which was originally designed with an approximately 600 MB recording capacity for storing digital audio data only, by using such high efficiency video compression techniques as MPEG. As a result of the increased effective recording capacity achieved using data compression techniques, karaoke titles and other conventional laser disk applications are gradually being transferred to the video CD format.

Users today expect both sophisticated title content and high reproduction quality. To meet these expectations, each title must be composed from bitstreams with an increasingly deep hierarchical structure. The data size of multimedia titles written with bitstreams having such deep hierarchical structures, however, is ten or more times greater than the data size of less complex titles. The need to edit small image (title) details also makes it necessary to process and control the bitstream using low order hierarchical data units.

It is therefore necessary to develop and prove a bitstream structure and an advanced digital processing method including both recording and reproduction capabilities whereby a large volume, multiple level hierarchical digital bitstream can be efficiently controlled at each level of the hierarchy. Also needed are an apparatus for executing this digital processing method, and a recording media to which the bitstream digitally processed by said apparatus can be efficiently recorded for storage and from which said recorded information can be quickly reproduced.

Means of increasing the storage capacity of conventional optical disks have been widely researched to address the recording medium aspect of this problem. One way to increase the storage capacity of the optical disk is to reduce the spot diameter D of the optical (laser) beam. If the wavelength of the laser beam is 1 and the aperture of the objective lens is NA, then the spot diameter D is proportional to 1/NA, and the storage capacity can be efficiently improved by decreasing 1 and increasing NA.

As described, for example, in United States Patent 5,235,581, however, coma caused by a

relative tilt between the disk surface and the optical axis of the laser beam (hereafter "tilt") increases when a large aperture (high NA) lens is used. To prevent tilt-induced coma, the transparent substrate must be made very thin. The problem is that the mechanical strength of the disk is low when the transparent substrate is very thin.

MPEG1, the conventional method of recording and reproducing video, audio, and graphic signal data, has also been replaced by the more robust MPEG2 method, which can transfer large data volumes at a higher rate. It should be noted that the compression method and data format of the MPEG2 standard differ somewhat from those of MPEG1. The specific content of and differences between MPEG1 and MPEG2 are described in detail in the ISO-11172 and ISO-13818 MPEG standards, and further description thereof is omitted below.

Note, however, that while the structure of the encoded video stream is defined in the MPEG2 specification, the hierarchical structure of the system stream and the method of processing lower hierarchical levels are not defined.

As described above, it is therefore not possible in a conventional authoring system to

process a large data stream containing sufficient information to satisfy many different user requirements. Moreover, even if such a processing method were available, the processed data recorded thereto cannot be repeatedly used to reduce data redundancy because there is no large capacity recording medium currently available that can efficiently record and reproduce high volume bitstreams such as described above.

More specifically, particular significant hardware and software requirements must be satisfied in order to process a bitstream using a data unit smaller than the title. These specific hardware requirements include significantly increasing the storage capacity of the recording medium and increasing the speed of digital processing; software requirements include inventing an advanced digital processing method including a sophisticated data structure.

Therefore, the object of the present invention is to provide an effective authoring system for controlling a multimedia data bitstream with advanced hardware and software requirements using a data unit smaller than the title to better address advanced user requirements.

To share data between plural titles and thereby efficiently utilize optical disk capacity, multi-scene control whereby scene data common to plural titles and the desired scenes on the same time-base from within multi-scene periods containing plural scenes unique to particular reproduction paths can be freely selected and reproduced is desirable.

However, when plural scenes unique to a reproduction path within the multi-scene period are arranged on the same time-base, the scene data must be contiguous. Unselected multi-scene data is therefore unavoidably inserted between the selected common scene data and the selected multi-scene data. The problem this creates when reproducing multi-scene data is that reproduction is interrupted by this unselected scene data.

When one of the multiple scenes is connected to common scene data, the difference between the video reproduction time and the audio reproduction time differs on each of the reproduction paths because of the offset between the audio and video frame reproduction times. As a result, the audio or video buffer underflows at the scene connection, causing video reproduction to

stop ("freeze") or audio reproduction to stop ("mute"), and thus preventing seamless reproduction. It will also be obvious that the difference between the audio and video reproduction times can cause a buffer underflow state even when common scene data is connected 1:1.

Therefore, the object of the present invention is to provide a data structure whereby multi-scene data can be naturally reproduced as a single title without the video presentation stopping ("freezing") at one-to-one, one-to-many, or many-to-many scene connections in the system stream; a method for generating a system stream having said data structure; a recording apparatus and a reproduction apparatus for recording and reproducing said system stream; and a medium to which said system stream can be recorded and from which said system stream can be reproduced by said recording apparatus and reproduction apparatus.

The present application is based upon Japanese Patent Application No. 7-252735 and 8-041581, which were filed on September 29, 1995 and February 28, 1996, respectively, the entire contents of which are expressly incorporated by reference herein.

Disclosure of Invention

The present invention has been developed with a view to substantially solving the above described disadvantages and has for its essential object to provide an optical disk for recording more than one system stream containing audio data and video data, wherein the audio data and video data of the plural system streams recorded to the optical disk are interleaved such that the difference between the input start times of the video data and audio data to the video buffer in the video decoder and the audio buffer in the audio decoder is less than the reproduction time of the number of audio frames that can be stored in the audio buffer plus one audio frame.

Brief Description of Drawings

Fig. 1 is a graph schematically showing a structure of multi media bit stream according to the present invention,

Fig. 2 is a block diagram showing an authoring encoder according to the present invention,

Fig. 3 is a block diagram showing an authoring decoder according to the present invention,

Fig. 4 is a side view of an optical disk storing the multi media bit stream of Fig. 1,

Fig. 5 is an enlarged view showing a portion confined by a circle of Fig. 4,

Fig. 6 is an enlarged view showing a portion confined by a circle of Fig. 5,

Fig. 7 is a side view showing a variation of the optical disk of Fig. 4,

Fig. 8 is a side view showing another variation of the optical disk of Fig. 4,

Fig. 9 is a plan view showing one example of track path formed on the recording surface of the optical disk of Fig. 4,

Fig. 10 is a plan view showing another example of track path formed on the recording surface of the optical disk of Fig. 4,

Fig. 11 is a diagonal view schematically showing one example of a track path pattern formed on the optical disk of Fig. 7,

Fig. 12 is a plan view showing another example of track path formed on the recording surface of the optical disk of Fig. 7,

Fig. 13 is a diagonal view schematically showing one example of a track path pattern formed on the optical disk of Fig. 8,

Fig. 14 is a plan view showing another example of track path formed on the recording surface of the optical disk of Fig. 8,

Fig. 15 is a flow chart showing details of the decoder synchronization process of Fig. 66,

Fig. 16 is a graph schematically showing the structure of multimedia bit stream for use in Digital Video Disk system according to the present invention,

Fig. 17 is a graph schematically showing the encoded video stream according to the present invention,

Fig. 18 is a graph schematically showing an internal structure of a video zone of Fig. 16.

Fig. 19 is a graph schematically showing the stream management information according to the present invention,

Fig. 20 is a graph schematically showing the structure the navigation pack NV of Fig. 17,

Fig. 21 is a graph in assistance of explaining a concept of parental lock playback control according to the present invention,

Fig. 22 is a graph schematically showing the data structure used in a digital video disk system according to the present invention,

Fig. 23 is a graph in assistance of explaining a concept of Multi-angle scene control according to the present invention,

Fig. 24 is a graph in assistance of explaining a concept of multi scene data connection,

Fig. 25 is a block diagram showing a DVD encoder according to the present invention,

Fig. 26 is a block diagram showing a DVD decoder according to the present invention,

Fig. 27 is a graph schematically showing an encoding information table generated by the encoding system controller of Fig. 25,

Fig. 28 is a graph schematically showing an encoding information tables,

Fig. 29 is a graph schematically showing an encoding parameters used by the video encoder of Fig. 25,

Fig. 30 is a graph schematically showing an example of the contents of the program chain information according to the present invention,

Fig. 31 is a graph schematically showing another example of the contents of the program chain information according to the present invention,

Fig. 32 is a flow chart showing the encode parameters generating operation for a system stream containing a single scene,

Fig. 33 is a graph in assistance of explaining a concept of multi-angle scene control according to the present in invention,

Fig. 34 is a flow chart, formed by Figs. 34A and 34B, showing an operation of the DVD encoder of Fig. 25,

Fig. 35 is a flow chart showing detailed of the encode parameter production sub-routine of Fig. 34,

Fig. 36 is a flow chart showing the detailed of the VOB data setting routine of Fig. 35,

Fig. 37 is a flow chart showing the encode parameters generating operation for a seamless switching,

Fig. 38 is a flow chart showing the encode parameters generating operation for a system stream,

Fig. 39 is a graph showing simulated results of data input/output to the video buffer and audio buffer of the DVD decoder of Fig. 26,

Fig. 40 is a graph in assistance of explaining a concept of parental control according to the present invention,

Fig. 41 is a graph in assistance of explaining the data input/output to the video buffer of the DVD decoder DCD shown in Fig. 26 during contiguous reproduction,

Fig. 42 is a graph in assistance of explaining a possible problem under a parental lock control example shown in Fig. 40,

Fig. 43 is a graph in assistance of explaining a reproduction gap generated under parental lock control,

Fig. 44 is a graph showing a system streams produced according to the present invention,

Fig. 45 is a graph in assistance of explaining an operation whereby these system streams are connected,

Fig. 46 is a graph in assistance of explaining a method of generating a system streams,

Fig. 47 is a graph in assistance of explaining another method of producing a system stream,

Fig. 48 is a graph showing a structure of the end of the second common system stream and the beginnings of the two parental lock control system streams,

Fig. 49 is a graph in assistance of explaing the difference in the video reproduction time and audio reproduction time of different reproduction paths,

Fig. 50 is a block diagram showing an internal structure of the system encoder in the DVD encoder of Fig. 25,

Fig. 51 is a graph showing a structure of the end of the two parental lock control system streams and the beginning of the following common system stream Sse,

Fig. 52 is a graph in assistance of explaining the difference in the video reproduction time and audio reproduction time of different reproduction paths,

Fig. 53 is a flow chart showing details of system stream producing routine of Fig. 34,

Fig. 54 is a graph in assistance of explaining an operation to calculate an audio data movement MFAp1,

Fig. 55 is a graph in assistance of explaining an operation to calculate an audio data movement MFAp2,

Fig. 56 is a block diagram showing an internal structure of the synchronizer of Fig. 26,

Fig. 57 is a flow chart showing an operation executed by the audio decoder controller of Fig. 26,

Figs. 58 and 59 are graphs showing decoding information table produced by the decoding system controller of Fig. 26,

Fig. 60 is a flow chart showing the operation of the DVD decoder DCD of Fig. 26,

Fig. 61 is a flow chart showing details of reproduction extracted PGC routing of Fig. 60,

Fig. 62 is a flow chat showing details of the stream buffer data transfer process according to the present invention,

Fig. 63 is a flow chart showing details of the non multi-angle decoding process of Fig. 62,

Fig. 64 is a flow chart showing details of the non-multi-angled interleave process of Fig. 63,

Fig. 65 is a flow chart showing details of the non-multi-angled contiguous block process,

Fig. 66 is a flow chart showing details of decoding data process of Fig. 64, performed by the stream buffer, is shown,

Fig. 67 is a graph schematically showing an actual arrangement of data blocks recorded to a data recording track on a recording medium according to the present invention,

Fig. 68 is a graph schematically showing contiguous block regions and interleaved block regions array,

Fig. 69 is a graph schematically showing a content of a VTS title VOBS according to the present invention, and

Fig. 70 is a graph schematically showing an internal data structure of the interleaved block regions according to the present invention.

Best Mode for Carrying Out the Invention

The prevent invention is detailedly described with reference to the accompanying drawings.

Data structure of the authoring system

The logic structure of the multimedia data bitstream processed using the recording apparatus, recording medium, reproduction apparatus, and authoring system according to the present invention is described first below with reference to Fig. 1.

In this structure, one title refers to the combination of video and audio data expressing program content recognized by a user for education, entertainment, or other purpose. Referenced to a motion picture (movie), one title may correspond to the content of an entire movie, or to just one scene within said movie.

A video title set (VTS) comprises the bitstream data containing the information for a specific number of titles. More specifically, each VTS comprises the video, audio, and other reproduction data representing the content of each title in the set, and control data for controlling the content data.

The video zone VZ is the video data unit processed by the authoring system, and comprises a specific number of video title sets. More specifically, each video zone is a linear sequence of K + 1 video title sets numbered VTS #0 - VTS #K where K is an integer value of zero or greater.

One video title set, preferably the first video title set VTS #0, is used as the video manager describing the content information of the titles contained in each video title set.

The multimedia bitstream MBS is the largest control unit of the multimedia data bitstream handled by the authoring system of the present invention, and comprises plural video zones VZ.

Authoring encoder EC

A preferred embodiment of the authoring encoder EC according to the present invention for generating a new multimedia bitstream MBS by re-encoding the original multimedia bitstream MBS according to the scenario desired by the user is shown in Fig. 2. Note that the original multimedia bitstream MBS comprises a video stream St1 containing the video information, a sub-picture stream St3 containing caption text and other auxiliary video information, and the audio stream St5 containing the audio information.

The video and audio streams are the bitstreams containing the video and audio information obtained from the source within a particular period of time. The sub-picture stream

is a bitstream containing momentary video information relevant to a particular scene. The sub-picture data encoded to a single scene may be captured to video memory and displayed continuously from the video memory for plural scenes as may be necessary.

When this multimedia source data St1, St3, and St5 is obtained from a live broadcast, the video and audio signals are supplied in real-time from a video camera or other imaging source; when the multimedia source data is reproduced from a video tape or other recording medium, the audio and video signals are not real-time signals.

While the multimedia source stream is shown in Fig. 2 as comprising these three source signals, this is for convenience only, and it should be noted that the multimedia source stream may contain more than three types of source signals, and may contain source data for different titles. Multimedia source data with audio, video, and sub-picture data for plural titles are referred to below as multi-title streams.

As shown in Fig. 2, the authoring encoder EC comprises a scenario editor 100, encoding system controller 200, video encoder 300, video stream

buffer 400, sub-picture encoder 500, sub-picture stream buffer 600, audio encoder 700, audio stream buffer 800, system encoder 900, video zone formatter 1300, recorder 1200, and recording medium M.

The video zone formatter 1300 comprises video object (VOB) buffer 1000, formatter 1100, and volume and file structure formatter 1400.

The bitstream encoded by the authoring encoder EC of the present embodiment is recorded by way of example only to an optical disk.

The scenario editor 100 of the authoring encoder EC outputs the scenario data, i.e., the user-defined editing instructions. The scenario data controls editing the corresponding parts of the multimedia bitstream MBS according to the user's manipulation of the video, sub-picture, and audio components of the original multimedia title. This scenario editor 100 preferably comprises a display, speaker(s), keyboard, CPU, and source stream buffer. The scenario editor 100 is connected to an external multimedia bitstream source from which the multimedia source data St1, St3, and St5 are supplied.

The user is thus able to reproduce the video and audio components of the multimedia source data using the display and speaker to confirm the content of the generated title. The user is then able to edit the title content according to the desired scenario using the keyboard, mouse, and other command input devices while confirming the content of the title on the display and speakers. The result of this multimedia data manipulation is the scenario data St7.

The scenario data St7 is basically a set of instructions describing what source data is selected from all or a subset of the source data containing plural titles within a defined time period, and how the selected source data is reassembled to reproduce the scenario (sequence) intended by the user. Based on the instructions received through the keyboard or other control device, the CPU codes the position, length, and the relative time-based positions of the edited parts of the respective multimedia source data streams St1, St3, and St5 to generate the scenario data St7.

The source stream buffer has a specific capacity, and is used to delay the multimedia

source data streams St1, St3, and St5 a known time Td and then output streams St1, St3, and St5.

This delay is required for synchronization with the editor encoding process. More specifically, when data encoding and user generation of scenario data St7 are executed simultaneously, i.e., when encoding immediately follows editing, time Td is required to determine the content of the multimedia source data editing process based on the scenario data St7 as will be described further below. As a result, the multimedia source data must be delayed by time Td to synchronize the editing process during the actual encoding operation. Because this delay time Td is limited to the time required to synchronize the operation of the various system components in the case of sequential editing as described above, the source stream buffer is normally achieved by means of a high speed storage medium such as semiconductor memory.

During batch editing in which all multimedia source data is encoded at once ("batch encoded") after scenario data St7 is generated for the complete title, delay time Td must be long enough to process the complete title or longer. In

this case, the source stream buffer may be a low speed, high capacity storage medium such as video tape, magnetic disk, or optical disk.

The structure (type) of media used for the source stream buffer may therefore be determined according to the delay time Td required and the allowable manufacturing cost.

The encoding system controller 200 is connected to the scenario editor 100 and receives the scenario data St7 therefrom. Based on the time-base position and length information of the edit segment contained in the scenario data St7, the encoding system controller 200 generates the encoding parameter signals St9, St11, and St13 for encoding the edit segment of the multimedia source data. The encoding signals St9, St11, and St13 supply the parameters used for video, sub-picture, and audio encoding, including the encoding start and end timing. Note that multimedia source data St1, St3, and St5 are output after delay time Td by the source stream buffer, and are therefore synchronized to encoding parameter signals St9, St11, and St13.

More specifically, encoding parameter signal St9 is the video encoding signal specifying

the encoding timing of video stream St1 to extract the encoding segment from the video stream St1 and generate the video encoding unit. Encoding parameter signal St11 is likewise the sub-picture stream encoding signal used to generate the sub-picture encoding unit by specifying the encoding timing for sub-picture stream St3. Encoding parameter signal St13 is the audio encoding signal used to generate the audio encoding unit by specifying the encoding timing for audio stream St5.

Based on the time-base relationship between the encoding segments of streams St1, St3, and St5 in the multimedia source data contained in scenario data St7, the encoding system controller 200 generates the timing signals St21, St23, and St25 arranging the encoded multimedia-encoded stream in the specified time-base relationship.

The encoding system controller 200 also generates the reproduction time information IT defining the reproduction time of the title editing unit (video object, VOB), and the stream encoding data St33 defining the system encode parameters for multiplexing the encoded multimedia stream containing video, audio, and sub-picture data.

Note that the reproduction time information IT and stream encoding data St33 are generated for the video object VOB of each title in one video zone VZ.

The encoding system controller 200 also generates the title sequence control signal St39, which declares the formatting parameters for formatting the title editing units VOB of each of the streams in a particular time-base relationship as a multimedia bitstream. More specifically, the title sequence control signal St39 is used to control the connections between the title editing units (VOB) of each title in the multimedia bitstream MBS, or to control the sequence of the interleaved title editing unit (VOBs) interleaving the title editing units VOB of plural reproduction paths.

The video encoder 300 is connected to the source stream buffer of the scenario editor 100 and to the encoding system controller 200, and receives therefrom the video stream St1 and video encoding parameter signal St9, respectively. Encoding parameters supplied by the video encoding signal St9 include the encoding start and end timing, bit rate, the encoding conditions for the encoding

start and end, and the material type. Possible material types include NTSC or PAL video signal, and telecine converted material. Based on the video encoding parameter signal St9, the video encoder 300 encodes a specific part of the video stream St1 to generate the encoded video stream St15.

The sub-picture encoder 500 is similarly connected to the source stream buffer of the scenario editor 100 and to the encoding system controller 200, and receives therefrom the sub-picture stream St3 and sub-picture encoding parameter signal St11, respectively. Based on the sub-picture encoding parameter signal St11, the sub-picture encoder 500 encodes a specific part of the sub-picture stream St3 to generate the encoded sub-picture stream St17.

The audio encoder 700 is also connected to the source stream buffer of the scenario editor 100 and to the encoding system controller 200, and receives therefrom the audio stream St5 and audio encoding parameter signal St13, which supplies the encoding start and end timing. Based on the audio encoding parameter signal St13, the audio encoder

700 encodes a specific part of the audio stream St5 to generate the encoded audio stream St19.

The video stream buffer 400 is connected to the video encoder 300 and to the encoding system controller 200. The video stream buffer 400 stores the encoded video stream St15 input from the video encoder 300, and outputs the stored encoded video stream St15 as the time-delayed encoded video stream St27 based on the timing signal St21 supplied from the encoding system controller 200.

The sub-picture stream buffer 600 is similarly connected to the sub-picture encoder 500 and to the encoding system controller 200. The sub-picture stream buffer 600 stores the encoded sub-picture stream St17 output from the sub-picture encoder 500, and then outputs the stored encoded sub-picture stream St17 as time-delayed encoded sub-picture stream St29 based on the timing signal St23 supplied from the encoding system controller 200.

The audio stream buffer 800 is similarly connected to the audio encoder 700 and to the encoding system controller 200. The audio stream buffer 800 stores the encoded audio stream St19 input from the audio encoder 700, and then outputs

the encoded audio stream St19 as the time-delayed encoded audio stream St31 based on the timing signal St25 supplied from the encoding system controller 200.

The system encoder 900 is connected to the video stream buffer 400, sub-picture stream buffer 600, audio stream buffer 800, and the encoding system controller 200, and is respectively supplied thereby with the time-delayed encoded video stream St27, time-delayed encoded sub-picture stream St29, time-delayed encoded audio stream St31, and the stream encoding data St33. Note that the system encoder 900 is a multiplexer that multiplexes the time-delayed streams St27, St29, and St31 based on the stream encoding data St33 (timing signal) to generate title editing unit (VOB) St35. The stream encoding data St33 contains the system encoding parameters, including the encoding start and end timing.

The video zone formatter 1300 is connected to the system encoder 900 and the encoding system controller 200 from which the title editing unit (VOB) St35 and title sequence control signal St39 (timing signal) are respectively supplied. The title sequence control signal St39 contains the

formatting start and end timing, and the formatting parameters used to generate (format) a multimedia bitstream MBS. The video zone formatter 1300 rearranges the title editing units (VOB) St35 in one video zone VZ in the scenario sequence defined by the user based on the title sequence control signal St39 to generate the edited multimedia stream data St43.

The multimedia bitstream MBS St43 edited according to the user-defined scenario is then sent to the recorder 1200. The recorder 1200 processes the edited multimedia stream data St43 to the data stream St45 format of the recording medium M, and thus records the formatted data stream St45 to the recording medium M. Note that the multimedia bitstream MBS recorded to the recording medium M contains the volume file structure VFS, which includes the physical address of the data on the recording medium generated by the video zone formatter 1300.

Note that the encoded multimedia bitstream MBS St35 may be output directly to the decoder to immediately reproduce the edited title content. It will be obvious that the output multimedia

bitstream MBS will not in this case contain the volume file structure VFS.

Authoring decoder DC

A preferred embodiment of the authoring decoder DC used to decode the multimedia bitstream MBS edited by the authoring encoder EC of the present invention, and thereby reproduce the content of each title unit according to the user-defined scenario, is described next below with reference to Fig. 3. Note that in the preferred embodiment described below the multimedia bitstream St45 encoded by the authoring encoder EC is recorded to the recording medium M.

As shown in Fig. 3, the authoring decoder DC comprises a multimedia bitstream producer 2000, scenario selector 2100, decoding system controller 2300, stream buffer 2400, system decoder 2500, video buffer 2600, sub-picture buffer 2700, audio buffer 2800, synchronizer 2900, video decoder 3800, sub-picture decoder 3100, audio decoder 3200, synthesizer 3500, video data output terminal 3600, and audio data output terminal 3700.

The bitstream producer 2000 comprises a recording media drive unit 2004 for driving the recording medium M; a reading head 2006 for reading

the information recorded to the recording medium M and producing the binary read signal St57; a signal processor 2008 for variously processing the read signal St57 to generate the reproduced bitstream St61; and a reproduction controller 2002.

The reproduction controller 2002 is connected to the decoding system controller 2300 from which the multimedia bitstream reproduction control signal St53 is supplied, and in turn generates the reproduction control signals St55 and St59 respectively controlling the recording media drive unit (motor) 2004 and signal processor 2008.

So that the user-defined video, sub-picture, and audio portions of the multimedia title edited by the authoring encoder EC are reproduced, the authoring decoder DC comprises a scenario selector 2100 for selecting and reproducing the corresponding scenes (titles). The scenario selector 2100 then outputs the selected titles as scenario data to the authoring decoder DC.

The scenario selector 2100 preferably comprises a keyboard, CPU, and monitor. Using the keyboard, the user then inputs the desired scenario based on the content of the scenario input by the

authoring encoder EC. Based on the keyboard input, the CPU generates the scenario selection data St51 specifying the selected scenario. The scenario selector 2100 is connected by an infrared communications device, for example, to the decoding system controller 2300, to which it inputs the scenario selection data St51.

Based on the scenario selection data St51, the decoding system controller 2300 then generates the bitstream reproduction control signal St53 controlling the operation of the bitstream producer 2000.

The stream buffer 2400 has a specific buffer capacity used to temporarily store the reproduced bitstream St61 input from the bitstream producer 2000, extract the address information and initial synchronization data SCR (system clock reference) for each stream, and generate bitstream control data St63. The stream buffer 2400 is also connected to the decoding system controller 2300, to which it supplies the generated bitstream control data St63.

The synchronizer 2900 is connected to the decoding system controller 2300 from which it receives the system clock reference SCR contained

in the synchronization control data St81 to set the internal system clock STC and supply the reset system clock St79 to the decoding system controller 2300.

Based on this system clock St79, the decoding system controller 2300 also generates the stream read signal St65 at a specific interval and outputs the read signal St65 to the stream buffer 2400.

Based on the supplied read signal St65, the stream buffer 2400 outputs the reproduced bitstream St61 at a specific interval to the system decoder 2500 as bitstream St67.

Based on the scenario selection data St51, the decoding system controller 2300 generates the decoding signal St69 defining the stream Ids for the video, sub-picture, and audio bitstreams corresponding to the selected scenario, and outputs to the system decoder 2500.

Based on the instructions contained in the decoding signal St69, the system decoder 2500 respectively outputs the video, sub-picture, and audio bitstreams input from the stream buffer 2400 to the video buffer 2600, sub-picture buffer 2700, and audio buffer 2800 as the encoded video stream

St71, encoded sub-picture stream St73, and encoded audio stream St75.

The system decoder 2500 detects the presentation time stamp PTS and decoding time stamp DTS of the smallest control unit in each bitstream St67 to generate the time information signal St77. This time information signal St77 is supplied to the synchronizer 2900 through the decoding system controller 2300 as the synchronization control data St81.

Based on this synchronization control data St81, the synchronizer 2900 determines the decoding start timing whereby each of the bitstreams will be arranged in the correct sequence after decoding, and then generates and inputs the video stream decoding start signal St89 to the video decoder 3800 based on this decoding timing. The synchronizer 2900 also generates and supplies the sub-picture decoding start signal St91 and audio stream decoding start signal St93 to the sub-picture decoder 3100 and audio decoder 3200, respectively.

The video decoder 3800 generates the video output request signal St84 based on the video stream decoding start signal St89, and outputs to

the video buffer 2600. In response to the video output request signal St84, the video buffer 2600 outputs the video stream St83 to the video decoder 3800. The video decoder 3800 thus detects the presentation time information contained in the video stream St83, and disables the video output request signal St84 when the length of the received video stream St83 is equivalent to the specified presentation time. A video stream equal in length to the specified presentation time is thus decoded by the video decoder 3800, which outputs the reproduced video signal St104 to the synthesizer 3500.

The sub-picture decoder 3100 similarly generates the sub-picture output request signal St86 based on the sub-picture decoding start signal St91, and outputs to the sub-picture buffer 2700. In response to the sub-picture output request signal St86, the sub-picture buffer 2700 outputs the sub-picture stream St85 to the sub-picture decoder 3100. Based on the presentation time information contained in the sub-picture stream St85, the sub-picture decoder 3100 decodes a length of the sub-picture stream St85 corresponding to the specified presentation time to reproduce and supply

to the synthesizer 3500 the sub-picture signal St99.

The synthesizer 3500 superimposes the video signal St104 and sub-picture signal St99 to generate and output the multi-picture video signal St105 to the video data output terminal 3600.

The audio decoder 3200 generates and supplies to the audio buffer 2800 the audio output request signal St88 based on the audio stream decoding start signal St93. The audio buffer 2800 thus outputs the audio stream St87 to the audio decoder 3200. The audio decoder 3200 decodes a length of the audio stream St87 corresponding to the specified presentation time based on the presentation time information contained in the audio stream St87, and outputs the decoded audio stream St101 to the audio data output terminal 3700.

It is thus possible to reproduce a user-defined multimedia bitstream MBS in real-time according to a user-defined scenario. More specifically, each time the user selects a different scenario, the authoring decoder DC is able to reproduce the title content desired by the user in the desired sequence by reproducing the

multimedia bitstream MBS corresponding to the selected scenario.

It is therefore possible by means of the authoring system of the present invention to generate a multimedia bitstream according to plural user-defined scenarios by real-time or batch encoding multimedia source data in a manner whereby the substreams of the smallest editing units (scenes), which can be divided into plural substreams, expressing the basic title content are arranged in a specific time-base relationship.

The multimedia bitstream thus encoded can then be reproduced according to the one scenario selected from among plural possible scenarios. It is also possible to change scenarios while playback is in progress, i.e., to select a different scenario and dynamically generate a new multimedia bitstream according to the most recently selected scenario. It is also possible to dynamically select and reproduce any of plural scenes while reproducing the title content according to a desired scenario.

It is therefore possible by means of the authoring system of the present invention to encode

and not only reproduce but to repeatedly reproduce a multimedia bitstream MBS in real-time.

A detail of the authoring system is disclosed Japanese Patent Application filed September 27, 1996, and entitled and assigned to the same assignee as the present application.

DVD

An example of a digital video disk (DVD) with only one recording surface (a single-sided DVD) is shown in Fig. 4.

The DVD recording medium RC1 in the preferred embodiment of the invention comprises a data recording surface RS1 to and from which data is written and read by emitting laser beam LS, and a protective layer PL1 covering the data recording surface RS1. A backing layer BL1 is also provided on the back of data recording surface RS1. The side of the disk on which protective layer PL1 is provided is therefore referred to below as side SA (commonly "side A"), and the opposite side (on which the backing layer BL1 is provided) is referred to as side SB ("side B"). Note that digital video disk recording media having a single data recording surface RS1 on only one side such as

this DVD recording medium RC1 is commonly called a single-sided single layer disk.

A detailed illustration of area C1 in Fig. 4 is shown in Fig. 5. Note that the data recording surface RS1 is formed by applying a metallic thin film or other reflective coating as a data layer 4109 on a first transparent layer 4108 having a particular thickness T1. This first transparent layer 4108 also functions as the protective layer PL1. A second transparent substrate 4111 of a thickness T2 functions as the backing layer BL1, and is bonded to the first transparent layer 4108 by means of an adhesive layer 4110 disposed therebetween.

A printing layer 4112 for printing a disk label may also be disposed on the second transparent substrate 4111 as necessary. The printing layer 4112 does not usually cover the entire surface area of the second transparent substrate 4111 (backing layer BL1), but only the area needed to print the text and graphics of the disk label. The area of second transparent substrate 4111 to which the printing layer 4112 is not formed may be left exposed. Light reflected from the data layer 4109 (metallic thin film)

forming the data recording surface RS1 can therefore be directly observed where the label is not printed when the digital video disk is viewed from side SB. As a result, the background looks like a silver-white over which the printed text and graphics float when the metallic thin film is an aluminum thin film, for example.

Note that it is only necessary to provide the printing layer 4112 where needed for printing, and it is not necessary to provide the printing layer 4112 over the entire surface of the backing layer BL1.

A detailed illustration of area C2 in Fig. 5 is shown in Fig. 6. Pits and lands are molded to the common contact surface between the first transparent layer 4108 and the data layer 4109 on side SA from which data is read by emitting a laser beam LS, and data is recorded by varying the lengths of the pits and lands (i.e., the length of the intervals between the pits). More specifically, the pit and land configuration formed on the first transparent layer 4108 is transferred to the data layer 4109. The lengths of the pits and lands is shorter, and the pitch of the data tracks formed by the pit sequences is narrower,

than with a conventional Compact Disc (CD). The surface recording density is therefore greatly improved.

Side SA of the first transparent layer 4108 on which data pits are not formed is a flat surface. The second transparent substrate 4111 is for reinforcement, and is a transparent panel made from the same material as the first transparent layer 4108 with both sides flat. Thicknesses T1 and T2 are preferably equal and commonly approximately 0.6 mm, but the invention shall not be so limited.

As with a CD, information is read by irradiating the surface with a laser beam LS and detecting the change in the reflectivity of the light spot. Because the objective lens aperture NA can be large and the wavelength 1 of the light beam small in a digital video disk system, the diameter of the light spot Ls used can be reduced to approximately 1/1.6 the light spot needed to read a CD. Note that this means the resolution of the laser beam LS in the DVD system is approximately 1.6 times the resolution of a conventional CD system.

The optical system used to read data from the digital video disk uses a short 650 nm wavelength red semiconductor laser and an objective lens with a 0.6 mm aperture NA. By thus also reducing the thickness T of the transparent panels to 0.6 mm, more than 5 GB of data can be stored to one side of a 120 mm diameter optical disk.

It is therefore possible to store motion picture (video) images having an extremely large per unit data size to a digital video disk system disk without losing image quality because the storage capacity of a single-sided, single-layer recording medium RC1 with one data recording surface RS1 as thus described is nearly ten times the storage capacity of a conventional CD. As a result, while the video presentation time of a conventional CD system is approximately 74 minutes if image quality is sacrificed, high quality video images with a video presentation time exceeding two hours can be recorded to a DVD.

The digital video disk is therefore well-suited as a recording medium for video images.

A digital video disk recording medium with plural recording surfaces RS as described above is shown in Figs. 7 and 8. The DVD recording medium

RC2 shown in Fig. 7 comprises two recording surfaces, i.e., first recording surface RS1 and semi-transparent second recording surface RS2, on the same side, i.e. side SA, of the disk. Data can be simultaneously recorded or reproduced from these two recording surfaces by using different laser beams LS1 and LS2 for the first recording surface RS1 and the second recording surface RS2. It is also possible to read/write both recording surfaces RS1 and RS2 using only one of the laser beams LS1 or LS2. Note that recording media thus comprised are called "single-side, dual-layer disks."

It should also be noted that while two recording surfaces RS1 and RS2 are provided in this example, it is also possible to produce digital video disk recording media having more than two recording surfaces RS. Disks thus comprised are known as "single-sided, multi-layer disks."

Though comprising two recording surfaces similarly to the recording media shown in Fig. 7, the DVD recording medium RC3 shown in Fig. 8 has the recording surfaces on opposite sides of the disk, i. e., has the first data recording surface RS1 on side SA and the second data recording surface RS2 on side SB. It will also be obvious

that while only two recording surfaces are shown on one digital video disk in this example, more than two recording surfaces may also be formed on a double-sided digital video disk. As with the recording medium shown in Fig. 7, it is also possible to provide two separate laser beams LS1 and LS2 for recording surfaces RS1 and RS2, or to read/write both recording surfaces RS1 and RS2 using a single laser beam. Note that this type of digital video disk is called a "double-sided, dual-layer disk." It will also be obvious that a double-sided digital video disk can be comprised with two or more recording surfaces per side. This type of disk is called a "double-sided, multi-layer disk."

A plan view from the laser beam LS irradiation side of the recording surface RS of the DVD recording medium RC is shown in Fig. 9 and Fig. 10. Note that a continuous spiral data recording track TR is provided from the inside circumference to the outside circumference of the DVD. The data recording track TR is divided into plural sectors each having the same known storage capacity. Note that for simplicity only the data recording track

TR is shown in Fig. 9 with more than three sectors per revolution.

As shown in Fig. 9, the data recording track TR is normally formed clockwise inside to outside (see arrow DrA) from the inside end point IA at the inside circumference of disk RCA to the outside end point OA at the outside circumference of the disk with the disk RCA rotating counterclockwise RdA. This type of disk RCA is called a clockwise disk, and the recording track formed thereon is called a clockwise track TRA.

Depending upon the application, the recording track TRB may be formed clockwise from outside to inside circumference (see arrow DrB in Fig. 10) from the outside end point OB at the outside circumference of disk RCB to the inside end point IB at the inside circumference of the disk with the disk RCB rotating clockwise RdB. Because the recording track appears to wind counterclockwise when viewed from the inside circumference to the outside circumference on disks with the recording track formed in the direction of arrow DrB, these disks are referred to as counterclockwise disk RCB with counterclockwise track TRB to distinguish them from disk RCA in Fig.

9. Note that track directions DrA and DrB are the track paths along which the laser beam travels when scanning the tracks for recording and playback. Direction of disk rotation RdA in which disk RCA turns is thus opposite the direction of track path DrA, and direction of disk rotation RdB in which disk RCB turns is thus opposite the direction of track path DrB.

An exploded view of the single-sided, dual-layer disk RC2 shown in Fig. 7 is shown as disk RC2o in Fig. 11. Note that the recording tracks formed on the two recording surfaces run in opposite directions. Specifically, a clockwise recording track TRA as shown in Fig. 9 is formed in clockwise direction DrA on the (lower) first data recording surface RS1, and a counterclockwise recording track TRB formed in counterclockwise direction DrB as shown in Fig. 10 is provided on the (upper) second data recording surface RS2. As a result, the outside end points OA and OB of the first and second (top and bottom) tracks are at the same radial position relative to the center axis of the disk RC2o. Note that track paths DrA and DrB of tracks TR are also the data read/write directions to disk RC. The first and second (top

and bottom) recording tracks thus wind opposite each other with this disk RC, i.e., the track paths DrA and DrB of the top and bottom recording layers are opposite track paths.

Opposite track path type, single-sided, dual-layer disks RC2o rotate in direction RdA corresponding to the first recording surface RS1 with the laser beam LS traveling along track path DrA to trace the recording track on the first recording surface RS1. When the laser beam LS reaches the outside end point OA, the laser beam LS can be refocused to end point OB on the second recording surface RS2 to continue tracing the recording track from the first to the second recording surface uninterrupted. The physical distance between the recording tracks TRA and TRB on the first and second recording surfaces RS1 and RS2 can thus be instantaneously eliminated by simply adjusting the focus of the laser beam LS.

It is therefore possible with an opposite track path type, single-sided, dual-layer disk RC2o to easily process the recording tracks disposed to physically discrete top and bottom recording surfaces as a single continuous recording track. It is therefore also possible in an authoring

system as described above with reference to Fig. 1 to continuously record the multimedia bitstream MBS that is the largest multimedia data management unit to two discrete recording surfaces RS1 and RS2 on a single recording medium RC2o.

It should be noted that the tracks on recording surfaces RS1 and RS2 can be wound in the directions opposite those described above, i.e., the counterclockwise track TRB may be provided on the first recording surface RS1 and the clockwise track TRA on the second recording surface RS2. In this case the direction of disk rotation is also changed to a clockwise rotation RdB, thereby enabling the two recording surfaces to be used as comprising a single continuous recording track as described above. For simplification, a further example of this type of disk is therefore neither shown nor described below.

It is therefore possible by thus constructing the digital video disk to record the multimedia bitstream MBS for a feature-length title to a single opposite track path type, single-sided, dual-layer disk RC2o. Note that this type of digital video disk medium is called a single-sided dual-layer disk with opposite track paths.

Another example of the single-sided, dual-layer DVD recording medium RC2 shown in Fig. 7 is shown as disk RC2p in Fig. 12. The recording tracks formed on both first and second recording surfaces RS1 and RS2 are clockwise tracks TRA as shown in Fig. 9. In this case, the single-sided, dual-layer disk RC2p rotates counterclockwise in the direction of arrow RdA, and the direction of laser beam LS travel is the same as the direction of the track spiral, i.e., the track paths of the top and bottom recording surfaces are mutually parallel (parallel track paths). The outside end points OA of both top and bottom tracks are again preferably positioned at the same radial position relative to the center axis of the disk RC2p as described above. As also described above with disk RC2o shown in Fig. 11, the access point can be instantaneously shifted from outside end point OA of track TRA on the first recording surface RS1 to the outside end point OA of track TRA on the second recording surface RS2 by appropriately adjusting the focus of the laser beam LS at outside end point OA.

However, for the laser beam LS to continuously access the clockwise recording track

TRA on the second recording surface RS2, the recording medium RC2p must be driven in the opposite direction (clockwise, opposite direction RdA). Depending on the radial position of the laser beam LS, however, it is inefficient to change the rotational direction of the recording medium. As shown by the diagonal arrow in Fig. 12, the laser beam LS is therefore moved from the outside end point OA of the track on the first recording surface RS1 to the inside end point IA of the track on the second recording surface RS2 to use these physically discrete recording tracks as one logically continuous recording track.

Rather than using the recording tracks on top and bottom recording surfaces as one continuous recording track, it is also possible to use the recording tracks to record the multimedia bitstreams MBS for different titles. This type of digital video disk recording medium is called a "single-sided, dual-layer disk with parallel track paths."

Note that if the direction of the tracks formed on the recording surfaces RS1 and RS2 is opposite that described above, i.e., counterclockwise recording tracks TRB are formed,

disk operation remains the same as that described above except for the direction of disk rotation, which is clockwise as shown by arrow RdB.

Whether using clockwise or counterclockwise recording tracks, the single-sided, dual-layer disk RC2p with parallel track paths thus described is well-suited to storing on a single disk encyclopedia and similar multimedia bitstreams comprising multiple titles that are frequently and randomly accessed.

An exploded view of the dual-sided single-layer DVD recording medium RC3 comprising one recording surface layer RS1 and RS2 on each side as shown in Fig. 8 is shown as DVD recording medium RC3s in Fig. 13. Clockwise recording track TRA is provided on the one recording surface RS1, and a counterclockwise recording track TRB is provided on the other recording surface RS2. As in the preceding recording media, the outside end points OA and OB of the recording tracks on each recording surface are preferably positioned at the same radial position relative to the center axis of the DVD recording medium RC3s.

Note that while the recording tracks on these recording surfaces RS1 and RS2 rotate in

opposite directions, the track paths are symmetrical. This type of recording medium is therefore known as a double-sided dual layer disk with symmetrical track paths. This double-sided dual layer disk with symmetrical track paths RC3s rotates in direction RdA when reading/writing the first recording surface RS1. As a result, the track path on the second recording surface RS2 on the opposite side is opposite the direction DrB in which the track winds, i.e., direction DrA. Accessing both recording surfaces RS1 and RS2 using a single laser beam LS is therefore not realistic irrespective of whether access is continuous or non-continuous. In addition, a multimedia bitstream MBS is separately recorded to the recording surfaces on the first and second sides of the disk.

A different example of the double-sided single layer disk RC3 shown in Fig. 8 is shown in Fig. 14 as disk RC3a. Note that this disk comprises clockwise recording tracks TRA as shown in Fig. 9 on both recording surfaces RS1 and RS2. As with the preceding recording media, the outside end points OA and OA of the recording tracks on each recording surface are preferably positioned at

the same radial position relative to the center axis of the DVD recording medium RC3a. Unlike the double-sided dual layer disk with symmetrical track paths RC3s described above, the tracks on these recording surfaces RS1 and RS2 are asymmetrical. This type of disk is therefore known as a double-sided dual layer disk with asymmetrical track paths. This double-sided dual layer disk with asymmetrical track paths RC3a rotates in direction RdA when reading/writing the first recording surface RS1. As a result, the track path on the second recording surface RS2 on the opposite side is opposite the direction DrA in which the track winds, i.e., direction DrB.

This means that if a laser beam LS is driven continuously from the inside circumference to the outside circumference on the first recording surface RS1, and then from the outside circumference to the inside circumference on the second recording surface RS2, both sides of the recording medium RC3a can be read/written without turning the disk over and without providing different laser beams for the two sides.

The track paths for recording surfaces RS1 and RS2 are also the same with this double-sided

dual layer disk with asymmetrical track paths RC3a. As a result, it is also possible to read/write both sides of the disk without providing separate laser beams for each side if the recording medium RC3a is turned over between sides, and the read/write apparatus can therefore be constructed economically.

It should be noted that this recording medium remains functionally identical even if counterclockwise recording track TRB is provided in place of clockwise recording track TRA on both recording surfaces RS1 and RS2.

As described above, the true value of a DVD system whereby the storage capacity of the recording medium can be easily increased by using a multiple layer recording surface is realized in multimedia applications whereby plural video data units, plural audio data units, and plural graphics data units recorded to a single disk are reproduced through interactive operation by the user.

It is therefore possible to achieve one long-standing desire of software (programming) providers, specifically, to provide programming content such as a commercial movie on a single recording medium in plural versions for different

language and demographic groups while retaining the image quality of the original.

Parental control

Content providers of movie and video titles have conventionally had to produce, supply, and manage the inventory of individual titles in multiple languages, typically the language of each distribution market, and multi-rated title packages conforming to the parental control (censorship) regulations of individual countries in Europe and North America. The time and resources required for this are significant. While high image quality is obviously important, the programming content must also be consistently reproducible.

The digital video disk recording medium is close to solving these problems.

Multiple angles

One interactive operation widely sought in multimedia applications today is for the user to be able to change the position from which a scene is viewed during reproduction of that scene. This capability is achieved by means of the multiple angle function.

This multiple angle function makes possible applications whereby, for example, a user

can watch a baseball game from different angles (or virtual positions in the stadium), and can freely switch between the views while viewing is in progress. In this example of a baseball game, the available angles may include a position behind the backstop centered on the catcher, batter, and pitcher; one from behind the backstop centered on a fielder, the pitcher, and the catcher; and one from center field showing the view to the pitcher and catcher.

To meet these requirements, the digital video disk system uses MPEG, the same basic standard format used with Video-Cds to record the video, audio, graphics, and other signal data. Because of the differences in storage capacity, transfer rates, and signal processing performance within the reproduction apparatus, DVD uses MPEG2, the compression method and data format of which differ slightly from the MPEG1 format used with Video-Cds.

It should be noted that the content of and differences between the MPEG1 and MPEG2 standards have no direct relationship to the intent of the present invention, and further description is

therefore omitted below (for more information, see MPEG specifications ISO-11172 and ISO-13818).

The data structure of the DVD system according to the present invention is described in detail below with reference to Figs. 16, 17, 18, 19, 20, and 21.

Multi-scene control

A fully functional and practical parental lock playback function and multi-angle scene playback function must enable the user to modify the system output in minor, subtle ways while still presenting substantially the same video and audio output. If these functions are achieved by preparing and recording separate titles satisfying each of the many possible parental lock and multi-angle scene playback requests, titles that are substantially identical and differ in only minor ways must be recorded to the recording medium. This results in identical data being repeatedly recorded to the larger part of the recording medium, and significantly reduces the utilization efficiency of the available storage capacity. More particularly, it is virtually impossible to record discrete titles satisfying every possible request even using the massive

capacity of the digital video disk medium. While it may be concluded that this problem can be easily solved by increasing the capacity of the recording medium, this is an obviously undesirable solution when the effective use of available system resources is considered.

Using multi-scene control, the concept of which is described in another section below, in a DVD system, it is possible to dynamically construct titles for numerous variations of the same basic content using the smallest possible amount of data, and thereby effectively utilize the available system resources (recording medium). More specifically, titles that can be played back with numerous variations are constructed from basic (common) scene periods containing data common to each title, and multi-scene periods comprising groups of different scenes corresponding to the various requests. During reproduction, the user is able to freely and at any time select particular scenes from the multi-scene periods to dynamically construct a title conforming to the desired content, e.g., a title omitting certain scenes using the parental lock control function.

Note that multi-scene control enabling a parental lock playback control function and multi-angle scene playback is described in another section below with reference to Fig. 21.

Data structure of the DVD system

The data structure used in the authoring system of a digital video disk system according to the present invention is shown in Fig. 22. To record a multimedia bitstream MBS, this digital video disk system divides the recording medium into three major recording areas, the lead-in area LI, the volume space VS, and the lead-out area LO.

The lead-in area LI is provided at the inside circumference area of the optical disk. In the disks described with reference to Figs. 9 and 10, the lead-in area LI is positioned at the inside end points IA and IB of each track. Data for stabilizing the operation of the reproducing apparatus when reading starts is written to the lead-in area LI.

The lead-out area LO is correspondingly located at the outside circumference of the optical disk, i.e., at outside end points OA and OB of each track in the disks described with reference to Figs. 9 and 10. Data identifying the end of the

volume space VS is recorded in this lead-out area LO.

The volume space VS is located between the lead-in area LI and lead-out area LO, and is recorded as a one-dimensional array of n+1 (where n is an integer greater than or equal to zero) 2048-byte logic sectors LS. The logic sectors LS are sequentially number #0, #1, #2, ... #n. The volume space VS is also divided into a volume and file structure management area VFS and a file data structure area FDS.

The volume and file structure management area VFS comprises m+1 logic sectors LS#0 to LS#m (where m is an integer greater than or equal to zero and less than n. The file data structure FDS comprises n-m logic sectors LS #m+1 to LS #n.

Note that this file data structure area FDS corresponds to the multimedia bitstream MBS shown in Fig. 1 and described above.

The volume file structure VFS is the file system for managing the data stored to the volume space VS as files, and is divided into logic sectors LS#0 - LS#m where m is the number of sectors required to store all data needed to manage the entire disk, and is a natural number less than

n. Information for the files stored to the file data structure area FDS is written to the volume file structure VFS according to a known specification such as ISO-9660 or ISO-13346.

The file data structure area FDS comprises n-m logic sectors LS#m - LS#n, each comprising a video manager VMG sized to an integer multiple of the logic sector (2048 x I, where I is a known integer), and k video title sets VTS #1 - VTS#k (where k is a natural number less than 100).

The video manager VMG stores the title management information for the entire disk, and information for building a volume menu used to set and change reproduction control of the entire volume.

Any video title set VTS #k is also called a "video file" representing a title comprising video, audio, and/or still image data.

The internal structure of each video title set VTS shown in Fig. 22 is shown in Fig. 16. Each video title set VTS comprises VTS information VTSI describing the management information for the entire disk, and the VTS title video objects VOB (VTSTT_VOBS), i.e., the system stream of the multimedia bitstream. The VTS information VTSI is

described first below, followed by the VTS title VOBS.

The VTS information primarily includes the VTSI management table VTSI_MAT and VTSPGC information table VTS_PGCIT.

The VTSI management table VTSI_MAT stores such information as the internal structure of the video title set VTS, the number of selectable audio streams contained in the video title set VTS, the number of sub-pictures, and the video title set VTS location (storage address).

The VTSPGC information table VTS_PGCIT records i (where i is a natural number) program chain (PGC) data blocks VTS_PGCI #1 - VTS_PGCI #i for controlling the playback sequence. Each of the table entries VTS_PGCI #i is a data entry expressing the program chain, and comprises j (where j is a natural number) cell playback information blocks C_PBI #1 - C_PBI #j. Each cell playback information block C_PBI #j contains the playback sequence of the cell and playback control information.

The program chain PGC is a conceptual structure describing the story of the title content, and therefore defines the structure of

each title by describing the cell playback sequence. Note that these cells are described in detail below.

If, for example, the video title set information relates to the menus, the video title set information VTSI is stored to a buffer in the playback device when playback starts. If the user then presses a MENU button on a remote control device, for example, during playback, the playback device references the buffer to fetch the menu information and display the top menu #1. If the menus are hierarchical, the main menu stored as program chain information VTS_PGCI #1 may be displayed, for example, by pressing the MENU button, VTS_PGCI #2 - #9 may correspond to submenus accessed using the numeric keypad on the remote control, and VTS_PGCI #10 and higher may correspond to additional submenus further down the hierarchy. Alternatively, VTS_PGCI #1 may be the top menu displayed by pressing the MENU button, while VTS_PGCI #2 and higher may be voice guidance reproduced by pressing the corresponding numeric key.

The menus themselves are expressed by the plural program chains defined in this table. As a

result, the menus may be freely constructed in various ways, and shall not be limited to hierarchical or non-hierarchical menus or menus containing voice guidance.

In the case of a movie, for example, the video title set information VTSI is stored to a buffer in the playback device when playback starts, the playback device references the cell playback sequence described by the program chain PGC, and reproduces the system stream.

The "cells" referenced here may be all or part of the system stream, and are used as access points during playback. Cells can therefore be used, for example, as the "chapters" into which a title may be divided.

Note that each of the PGC information entries C_PBI #j contain both cell playback processing information and a cell information table. The cell playback processing information comprises the processing information needed to reproduce the cell, such as the presentation time and number of repetitions. More specifically, this information includes the cell block mode CBM, cell block type CBT, seamless playback flag SPF, interleaved allocation flag IAF, STC resetting flag

STCDF, cell presentation time C_PBTM, seamless angle change flag SACF, first cell VOBU start address C_FVOBU_SA, and the last cell VOBU start address C_LVOBU_SA.

Note that seamless playback refers to the reproduction in a digital video disk system of multimedia data including video, audio, and sub-picture data without intermittent breaks in the data or information. Seamless playback is described in detail in another section below with reference to Fig. 23 and Fig. 24.

The cell block mode CBM indicates whether plural cells constitute one functional block. The cell playback information of each cell in a functional block is arranged consecutively in the PGC information. The cell block mode CBM of the first cell playback information in this sequence contains the value of the first cell in the block, and the cell block mode CBM of the last cell playback information in this sequence contains the value of the last cell in the block. The cell block mode CBM of each cell arrayed between these first and last cells contains a value indicating that the cell is a cell between these first and last cells in that block.

The cell block type CBT identifies the type of the block indicated by the cell block mode CBM. For example, when a multiple angle function is enabled, the cell information corresponding to each of the reproducible angles is programmed as one of the functional blocks mentioned above, and the type of these functional blocks is defined by a value identifying "angle" in the cell block type CBT for each cell in that block.

The seamless playback flag SPF simply indicates whether the corresponding cell is to be linked and played back seamlessly with the cell or cell block reproduced immediately therebefore. To seamlessly reproduce a given cell with the preceding cell or cell block, the seamless playback flag SPF is set to 1 in the cell playback information for that cell; otherwise SPF is set to 0.

The interleaved allocation flag IAF stores a value identifying whether the cell exists in a contiguous or interleaved block. If the cell is part of an interleaved block, the flag IAF is set to 1; otherwise it is set to 0.

The STC resetting flag STCDF identifies whether the system time clock STC used for

synchronization must be reset when the cell is played back; when resetting the system time clock STC is necessary, the STC resetting flag STCDF is set to 1.

The seamless angle change flag SACF stores a value indicating whether a cell in a multi-angle period should be connected seamlessly at an angle change. If the angle change is seamless, the seamless angle change flag SACF is set to 1; otherwise it is set to 0.

The cell presentation time C_PBTM expresses the cell presentation time with video frame precision.

The first cell VOBU start address C_FVOBU_SA is the VOBU start address of the first cell in a block, and is also expressed as the distance from the logic sector of the first cell in the VTS title VOBS (VTSTT_VOBS) as measured by the number of sectors.

The last cell VOBU start address C_LVOBU_SA is the VOBU start address of the last cell in the block. The value of this address is expressed as the distance from the logic sector of the first cell in the VTS title VOBS (VTSTT_VOBS) as measured by the number of sectors.

The VTS title VOBS (VTSTT_VOBS), i.e., the multimedia system stream data, is described next. The system stream data VTSTT_VOBS comprises i (where i is a natural number) system streams SS, each of which is referred to as a "video object" (VOB). Each video object VOB #1 - VOB #i comprises at least one video data block interleaved with up to a maximum eight audio data blocks and up to a maximum 32 sub-picture data blocks.

Each video object VOB comprises q (where q is a natural number) cells C#1 - C#q. Each cell C comprises r (where r is a natural number) video object units VOBU #1 - VOBU #r.

Each video object unit VOBU comprises plural groups_of_pictures GOP, and the audio and sub-pictures corresponding to the playback of said plural groups_of_pictures GOP. Note that the group_of_pictures GOP corresponds to the video encoding refresh cycle. Each video object unit VOBU also starts with an NV pack, i.e., the control data for that VOBU.

The structure of the navigation packs NV is described with reference to Fig. 18.

Before describing the navigation pack NV, the internal structure of the video zone VZ (see

Fig. 22), i.e., the system stream St35 encoded by the authoring encoder EC described with reference to Fig. 25, is described with reference to Fig. 17. Note that the encoded video stream St15 shown in Fig. 17 is the compressed one-dimensional video data stream encoded by the video encoder 300. The encoded audio stream St19 is likewise the compressed one-dimensional audio data stream multiplexing the right and left stereo audio channels encoded by the audio encoder 700. Note that the audio signal shall not be limited to a stereo signal, and may also be a multichannel surround-sound signal.

The system stream (title editing unit VOB) St35 is a one dimensional array of packs with a byte size corresponding to the logic sectors LS #n having a 2048-byte capacity as described using Fig. 21. A stream control pack is placed at the beginning of the title editing unit (VOB) St35, i.e., at the beginning of the video object unit VOBU. This stream control pack is called the "navigation pack NV", and records the data arrangement in the system stream and other control information.

The encoded video stream St15 and the encoded audio stream St19 are packetized in byte units corresponding to the system stream packs. These packets are shown in Fig. 17 as packets V1, V2, V3, V4... and A1, A2, A3.... As shown in Fig. 17, these packets are interleaved in the appropriate sequence as system stream St35, thus forming a packet stream, with consideration given to the decoder buffer size and the time required by the decoder to expand the video and audio data packets. In the example shown in Fig. 17, the packet stream is interleaved in the sequence V1, V2, A1, V3, V4, A2....

Note that the sequence shown in Fig. 17 interleaves one video data unit with one audio data unit. Significantly increased recording/playback capacity, high speed recording/playback, and performance improvements in the signal processing LSI enable the DVD system to record plural audio data and plural sub-picture data (graphics data) to one video data unit in a single interleaved MPEG system stream, and thereby enable the user to select the specific audio data and sub-picture data to be reproduced during playback. The structure of

the system stream used in this type of DVD system is shown in Fig. 18 and described below.

As in Fig. 17, the packetized encoded video stream St15 is shown in Fig. 18 as V1, V2, V3, V4, ... In this example, however, there is not just one encoded audio stream St19, but three encoded audio streams St19A, St19B, and St19C input as the source data. There are also two encoded sub-picture streams St17A and St17B input as the source data sub-picture streams. These six compressed data streams, St15, St19A, St19B, St19C, St17A and St17B, are interleaved to a single system stream St35.

The video data is encoded according to the MPEG specification with the group_of_pictures GOP being the unit of compression. In general, each group_of_pictures GOP contains 15 frames in the case of an NTSC signal, but the specific number of frames compressed to one GOP is variable. The stream management pack, which describes the management data containing, for example, the relationship between interleaved data, is also interleaved at the GOP unit interval. Because the group_of_pictures GOP unit is based on the video data, changing the number of video frames per GOP

unit changes the interval of the stream management packs. This interval is expressed in terms of the presentation time on the digital video disk within a range from 0.4 sec. to 1.0 sec. referenced to the GOP unit. If the presentation time of contiguous plural GOP units is less than 1 sec., the management data packs for the video data of the plural GOP units is interleaved to a single stream.

These management data packs are referred to as navigation packs NV in the digital video disk system. The data from one navigation pack NV to the packet immediately preceding the next navigation pack NV forms one video object unit VOBU. In general, one contiguous playback unit that can be defined as one scene is called a video object VOB, and each video object VOB contains plural video object units VOBU. Data sets of plural video objects VOB form a VOB set (VOBS). Note that these data units were first used in the digital video disk.

When plural of these data streams are interleaved, the navigation packs NV defining the relationship between the interleaved packs must also be interleaved at a defined unit known as the pack number unit. Each group_of_pictures GOP is

normally a unit containing approximately 0.5 sec. of video data, which is equivalent to the presentation time required for 12 - 15 frames, and one navigation pack NV is generally interleaved with the number of data packets required for this presentation time.

The stream management information contained in the interleaved video, audio, and sub-picture data packets constituting the system stream is described below with reference to Fig. 19 As shown in Fig. 19, the data contained in the system stream is recorded in a format packed or packetized according to the MPEG2 standard. The packet structure is essentially the same for video, audio, and sub-picture data. One pack in the digital video disk system has a 2048 byte capacity as described above, and contains a pack header PKH and one packet PES; each packet PES contains a packet header PTH and data block.

The pack header PKH records the time at which that pack is to be sent from stream buffer 2400 to system decoder 2500 (see Fig. 26), i.e., the system clock reference SCR defining the reference time for synchronized audio-visual data playback. The MPEG standard assumes that the

system clock reference SCR is the reference clock for the entire decoder operation. With such disk media as the digital video disk, however, time management specific to individual disk players can be used, and a reference clock for the decoder system is therefore separately provided.

The packet header PTH similarly contains a presentation time stamp PTS and a decoding time stamp DTS, both of which are placed in the packet before the access unit (the decoding unit). The presentation time stamp PTS defines the time at which the video data or audio data contained in the packet should be output as the playback output after being decoded, and the decoding time stamp DTS defines the time at which the video stream should be decoded. Note that the presentation time stamp PTS effectively defines the display start timing of the access unit, and the decoding time stamp DTS effectively defines the decoding start timing of the access unit. If the PTS and DTS are the same time, the DTS is omitted.

The packet header PTH also contains an 8-bit field called the stream ID identifying the packet type, i.e., whether the packet is a video

packet containing a video data stream, a private packet, or an MPEG audio packet.

Private packets under the MPEG2 standard are data packets of which the content can be freely defined. Private packet 1 in this embodiment of the invention is used to carry audio data other than the MPEG audio data, and sub-picture data; private packet 2 carries the PCI packet and DSI packet.

Private packets 1 and 2 each comprise a packet header, private data area, and data area. The private data area contains an 8-bit sub-stream ID indicating whether the recorded data is audio data or sub-picture data. The audio data defined by private packet 2 may be defined as any of eight types #0 - #7 of linear PCM or AC-3 encoded data. Sub-picture data may be defined as one of up to 32 types #0 - #31.

The data area is the field to which data compressed according to the MPEG2 specification is written if the stored data is video data; linear PCM, AC-3, or MPEG encoded data is written if audio data is stored; or graphics data compressed by runlength coding is written if sub-picture data is stored.

MPEG2-compressed video data may be compressed by constant bit rate (CBR) or variable bit rate (VBR) coding. With constant bit rate coding, the video stream is input continuously to the video buffer at a constant rate. This contrasts with variable bit rate coding in which the video stream is input intermittently to the video buffer, thereby making it possible to suppress the generation of unnecessary code. Both constant bit rate and variable bit rate coding can be used in the digital video disk system.

Because MPEG video data is compressed with variable length coding, the data quantity in each group_of_pictures GOP is not constant. The video and audio decoding times also differ, and the time-base relationship between the video and audio data read from an optical disk, and the time-base relationship between the video and audio data output from the decoder, do not match. The method of time-base synchronizing the video and audio data is therefore described in detail below with reference to Fig. 26, but is described briefly below based on constant bit rate coding.

The navigation pack NV structure is shown in Fig. 20. Each navigation pack NV starts with a

pack header PKH, and contains a PCI packet and DSI packet.

As described above, the pack header PKH records the time at which that pack is to be sent from stream buffer 2400 to system decoder 2500 (see Fig. 26), i.e., the system clock reference SCR defining the reference time for synchronized audio-visual data playback.

Each PCI packet contains PCI General Information (PCI_GI) and Angle Information for Non-seamless playback (NMSL_AGLI).

The PCI General Information (PCI_GI) declares the display time of the first video frame (the Start PTM of VOBU (VOBU_S_PTM)), and the display time of the last video frame (End PTM of VOBU (VOBU_E_PTM)), in the corresponding video object unit VOBU with system clock precision (90 Khz).

The Angle Information for Non-seamless playback (NMSL_AGLI) states the read start address of the corresponding video object unit VOBU when the angle is changed expressed as the number of sectors from the beginning of the video object VOB. Because there are nine or fewer angles in this example, there are nine angle address declaration

cells: Destination Address of Angle Cell #1 for Non-seamless playback (NMSL_AGL_C1_DSTA) to Destination Address of Angle Cell #9 for Non-seamless playback (NMSL_AGL_C9_DSTA).

Each DSI packet contains DSI General Information (DSI_GI), Seamless Playback Information (SML_PBI), and Angle Information for Seamless playback (SML_AGLI).

The DSI General Information (DSI_GI) declares the address of the last pack in the video object unit VOBU, i. e., the End Address for VOB (VOBU_EA), expressed as the number of sectors from the beginning of the video object unit VOBU.

While seamless playback is described in detail later, it should be noted that the continuously read data units must be interleaved (multiplexed) at the system stream level as an interleaved unit ILVU in order to seamlessly reproduce split or combined titles. Plural system streams interleaved with the interleaved unit ILVU as the smallest unit are defined as an interleaved block.

The Seamless Playback Information (SML_PBI) is declared to seamlessly reproduce the stream interleaved with the interleaved unit ILVU

as the smallest data unit, and contains an Interleaved Unit Flag (ILVU flag) identifying whether the corresponding video object unit VOBU is an interleaved block. The ILVU flag indicates whether the video object unit VOBU is in an interleaved block, and is set to 1 when it is. Otherwise the ILVU flag is set to 0.

When a video object unit VOBU is in an interleaved block, a Unit END flag is declared to indicate whether the video object unit VOBU is the last VOBU in the interleaved unit ILVU. Because the interleaved unit ILVU is the data unit for continuous reading, the Unit END flag is set to 1 if the VOBU currently being read is the last VOBU in the interleaved unit ILVU. Otherwise the Unit END flag is set to 0.

An Interleaved Unit End Address (ILVU_EA) identifying the address of the last pack in the ILVU to which the VOBU belongs, and the starting address of the next interleaved unit ILVU, Next Interleaved Unit Start Address (NT_ILVU_SA), are also declared when a video object unit VOBU is in an interleaved block. Both the Interleaved Unit End Address (ILVU_EA) and Next Interleaved Unit Start Address (NT_ILVU_SA) are expressed as the

number of sectors from the navigation pack NV of that VOBU.

When two system streams are seamlessly connected but the audio components of the two system streams are not contiguous, particularly immediately before and after the seam, it is necessary to pause the audio output to synchronize the audio and video components of the system stream following the seam. Note that non-contiguous audio may result from different audio signals being recording with the corresponding video blocks. With an NTSC signal, for example, the video frame cycle is approximately 33. 33 msec while the AC-3 audio frame cycle is 32 msec.

To enable this resynchronization, audio reproduction stopping times 1 and 2, i.e., Audio Stop PTM 1 in VOB (VOB_A_STP_PTM1), and Audio Stop PTM2 in VOB (VOB_A_STP_PTM2), indicating the time at which the audio is to be paused; and audio reproduction stopping periods 1 and 2, i.e., Audio Gap Length 1 in VOB (VOB_A_GAP_LEN1) and Audio Gap Length 2 in VOB (VOB_A_GAP_LEN2), indicating for how long the audio is to be paused, are also declared in the DSI packet. Note that these times

are specified at the system clock precision (90 Khz).

The Angle Information for Seamless playback (SML_AGLI) declares the read start address when the angle is changed. Note that this field is valid when seamless, multi-angle control is enabled. This address is also expressed as the number of sectors from the navigation pack NV of that VOBU. Because there are nine or fewer angles, there are nine angle address declaration cells: Destination Address of Angle Cell #1 for Seamless playback (SML_AGL_C1_DSTA) to Destination Address of Angle Cell #9 for Seamless playback (SML_AGL_C9_DSTA).

Note also that each title is edited in video object (VOB) units. Interleaved video objects (interleaved title editing units) are referenced as "VOBS"; and the encoded range of the source data is the encoding unit.

DVD encoder

A preferred embodiment of a digital video disk system authoring encoder ECD in which the multimedia bitstream authoring system according to the present invention is applied to a digital video disk system is described below and shown in Fig.

25. It will be obvious that the authoring encoder ECD applied to the digital video disk system, referred to below as a DVD encoder, is substantially identical to the authoring encoder EC shown in Fig. 2. The basic difference between these encoders is the replacement in the DVD encoder ECD of the video zone formatter 1300 of the authoring encoder EC above with a VOB buffer 1000 and formatter 1100. It will also be obvious that the bitstream encoded by this DVD encoder ECD is recorded to a digital video disk medium M. The operation of this DVD encoder ECD is therefore described below in comparison with the authoring encoder EC described above.

As in the above authoring encoder EC, the encoding system controller 200 generates control signals St9, St11, St13, St21, St23, St25, St33, and St39 based on the scenario data St7 describing the user-defined editing instructions input from the scenario editor 100, and controls the video encoder 300, sub-picture encoder 500, and audio encoder 700 in the DVD encoder ECD. Note that the user-defined editing instructions in the DVD encoder ECD are a superset of the editing

instructions of the authoring encoder EC described above.

Specifically, the user-defined editing instructions (scenario data St7) in the DVD encoder ECD similarly describe what source data is selected from all or a subset of the source data containing plural titles within a defined time period, and how the selected source data is reassembled to reproduce the scenario (sequence) intended by the user. The scenario data St7 of the DVD encoder ECD, however, further contains such information as: the number of streams contained in the editing units, which are obtained by splitting a multi-title source stream into blocks at a constant time interval; the number of audio and sub-picture data cells contained in each stream, and the sub-picture display time and period; whether the title is a multi-rated title enabling parental lock control; whether the user content is selected from plural streams including, for example, multiple viewing angles; and the method of connecting scenes when the angle is switched among the multiple viewing angles.

The scenario data St7 of the DVD encoder ECD also contains control information on a video

object VOB unit basis. This information is required to encode the media source stream, and specifically includes such information as whether there are multiple angles or parental control features. When multiple angle viewing is enabled, the scenario data St7 also contains the encoding bit rate of each stream considering data interleaving and the disk capacity, the start and end times of each control, and whether a seamless connection should be made between the preceding and following streams.

The encoding system controller 200 extracts this information from the scenario data St7, and generates the encoding information table and encoding parameters required for encoding control. The encoding information table and encoding parameters are described with reference to Figs. 27, 28, and 29 below.

The stream encoding data St33 contains the system stream encoding parameters and system encoding start and end timing values required by the DVD system to generate the VOBs. These system stream encoding parameters include the conditions for connecting one video object VOB with those before and after, the number of audio streams, the

audio encoding information and audio Ids, the number of sub-pictures and the sub-picture Ids, the video playback starting time information VPTS, and the audio playback starting time information APTS.

The title sequence control signal St39 supplies the multimedia bitstream MBS formatting start and end timing information and formatting parameters declaring the reproduction control information and interleave information.

Based on the video encoding parameter and encoding start/end timing signal St9, the video encoder 300 encodes a specific part of the video stream St1 to generate an elementary stream conforming to the MPEG2 Video standard defined in ISO-13818. This elementary stream is output to the video stream buffer 400 as encoded video stream St15.

Note that while the video encoder 300 generates an elementary stream conforming to the MPEG2 Video standard defined in ISO-13818, specific encoding parameters are input via the video encoding parameter signal St9, including the encoding start and end timing, bit rate, the encoding conditions for the encoding start and end, the material type, including whether the material

is an NTSC or PAL video signal or telecine converted material, and whether the encoding mode is set for either open GOP or closed GOP encoding.

The MPEG2 coding method is basically an interframe coding method using the correlation between frames for maximum signal compression, i.e., the frame being coded (the target frame) is coded by referencing frames before and/or after the target frame. However, intra-coded frames, i. e. , frames that are coded based solely on the content of the target frame, are also inserted to avoid error propagation and enable accessibility from mid-stream (random access). The coding unit containing at least one intra-coded frame ("intra-frame") is called a group_of_pictures GOP.

A group_of_pictures GOP in which coding is closed completely within that GOP is known as a "closed GOP." A group_of_pictures GOP containing a frame coded with reference to a frame in a preceding or following (ISO-13818 DOES NOT LIMIT P- and B-picture CODING to referencing PAST frames) group_of_pictures GOP is an "open GOP." It is therefore possible to playback a closed GOP using only that GOP. Reproducing an open GOP, however,

also requires the presence of the referenced GOP, generally the GOP preceding the open GOP.

The GOP is often used as the access unit. For example, the GOP may be used as the playback start point for reproducing a title from the middle, as a transition point in a movie, or for fast-forward play and other special reproduction modes. High speed reproduction can be achieved in such cases by reproducing only the intra-frame coded frames in a GOP or by reproducing only frames in GOP units.

Based on the sub-picture stream encoding parameter signal St11, the sub-picture encoder 500 encodes a specific part of the sub-picture stream St3 to generate a variable length coded bitstream of bitmapped data. This variable length coded bitstream data is output as the encoded sub-picture stream St17 to the sub-picture stream buffer 600.

Based on the audio encoding parameter signal St13, the audio encoder 700 encodes a specific part of the audio stream St5 to generate the encoded audio data. This encoded audio data may be data based on the MPEG1 audio standard defined in ISO-11172 and the MPEG2 audio standard defined in ISO-13818, AC-3 audio data, or PCM

(LPCM) data. Note that the methods and means of encoding audio data according to these standards are known and commonly available.

The video stream buffer 400 is connected to the video encoder 300 and to the encoding system controller 200. The video stream buffer 400 stores the encoded video stream St15 input from the video encoder 300, and outputs the stored encoded video stream St15 as the time-delayed encoded video stream St27 based on the timing signal St21 supplied from the encoding system controller 200.

The sub-picture stream buffer 600 is similarly connected to the sub-picture encoder 500 and to the encoding system controller 200. The sub-picture stream buffer 600 stores the encoded sub-picture stream St17 input from the sub-picture encoder 500, and then outputs the stored encoded sub-picture stream St17 as time-delayed encoded sub-picture stream St29 based on the timing signal St23 supplied from the encoding system controller 200.

The audio stream buffer 800 is similarly connected to the audio encoder 700 and to the encoding system controller 200. The audio stream buffer 800 stores the encoded audio stream St19

input from the audio encoder 700, and then outputs the encoded audio stream St19 as the time-delayed encoded audio stream St31 based on the timing signal St25 supplied from the encoding system controller 200.

The system encoder 900 is connected to the video stream buffer 400, sub-picture stream buffer 600, audio stream buffer 800, and the encoding system controller 200, and is respectively supplied thereby with the time-delayed encoded video stream St27, time-delayed encoded sub-picture stream St29, time-delayed encoded audio stream St31, and the system stream encoding parameter data St33. Note that the system encoder 900 is a multiplexer that multiplexes the time-delayed streams St27, St29, and St31 based on the stream encoding data St33 (timing signal) to generate title editing units (VOBs) St35.

The VOB buffer 1000 temporarily stores the video objects VOBs produced by the system encoder 900. The formatter 1100 reads the delayed video objects VOB from the VOB buffer 1000 based on the title sequence control signal St39 to generate one video zone VZ, and adds the volume file structure

VFS to generate the edited multimedia stream data St43.

The multimedia bitstream MBS St43 edited according to the user-defined scenario is then sent to the recorder 1200. The recorder 1200 processes the edited multimedia stream data St43 to the data stream St45 format of the recording medium M, and thus records the formatted data stream St45 to the recording medium M.

DVD decoder

A preferred embodiment of a digital video disk system authoring decoder DCD in which the multimedia bitstream authoring system of the present invention is applied to a digital video disk system is described below and shown in Fig. 26. The authoring decoder DCD applied to the digital video disk system, referred to below as a DVD decoder DCD, decodes the multimedia bitstream MBS edited using the DVD encoder ECD of the present invention, and recreates the content of each title according to the user-defined scenario. It will also be obvious that the multimedia bitstream St45 encoded by this DVD encoder ECD is recorded to a digital video disk medium M.

The basic configuration of the DVD decoder DCD according to this embodiment is the same as that of the authoring decoder DC shown in Fig. 3. The differences are that a different video decoder 3801 (shown as 3800 in Fig. 26) is used in place of the video decoder 3800, and a reordering buffer 3300 and selector 3400 are disposed between the video decoder 3801 and synthesizer 3500.

Note that the selector 3400 is connected to the synchronizer 2900, and is controlled by a switching signal St103.

The operation of this DVD decoder DCD is therefore described below in comparison with the authoring decoder DC described above.

As shown in Fig. 26, the DVD decoder DCD comprises a multimedia bitstream producer 2000, scenario selector 2100, decoding system controller 2300, stream buffer 2400, system decoder 2500, video buffer 2600, sub-picture buffer 2700, audio buffer 2800, synchronizer 2900, video decoder 3801, reordering buffer 3300, sub-picture decoder 3100, audio decoder 3200, selector 3400, synthesizer 3500, video data output terminal 3600, and audio data output terminal 3700.

The bitstream producer 2000 comprises a recording media drive unit 2004 for driving the recording medium M; a reading head 2006 for reading the information recorded to the recording medium M and producing the binary read signal St57; a signal processor 2008 for variously processing the read signal St57 to generate the reproduced bitstream St61; and a reproduction controller 2002.

The reproduction controller 2002 is connected to the decoding system controller 2300 from which the multimedia bitstream reproduction control signal St53 is supplied, and in turn generates the reproduction control signals St55 and St59 respectively controlling the recording media drive unit (motor) 2004 and signal processor 2008.

So that the user-defined video, sub-picture, and audio portions of the multimedia title edited by the authoring encoder EC are reproduced, the authoring decoder DC comprises a scenario selector 2100 for selecting and reproducing the corresponding scenes (titles). The scenario selector 2100 then outputs the selected titles as scenario data to the DVD decoder DCD.

The scenario selector 2100 preferably comprises a keyboard, CPU, and monitor. Using the

keyboard, the user then inputs the desired scenario based on the content of the scenario input by the DVD encoder ECD. Based on the keyboard input, the CPU generates the scenario selection data St51 specifying the selected scenario. The scenario selector 2100 is connected to the decoding system controller 2300 by an infrared communications device, for example, and inputs the generated scenario selection data St51 to the decoding system controller 2300.

The stream buffer 2400 has a specific buffer capacity used to temporarily store the reproduced bitstream St61 input from the bitstream producer 2000, extract the volume file structure VFS, the initial synchronization data SCR (system clock reference) in each pack, and the VOBU control information (DSI) in the navigation pack NV, to generate the bitstream control data St63. The stream buffer 2400 is also connected to the decoding system controller 2300, to which it supplies the generated bitstream control data St63.

Based on the scenario selection data St51 supplied by the scenario selector 2100, the decoding system controller 2300 then generates the bitstream reproduction control signal St53

controlling the operation of the bitstream producer 2000. The decoding system controller 2300 also extracts the user-defined playback instruction data from the bitstream reproduction control signal St53, and generates the decoding information table required for decoding control. This decoding information table is described further below with reference to Figs. 58 and 59. The decoding system controller 2300 also extracts the title information recorded to the optical disk M from the file data structure area FDS of the bitstream control data St63 to generate the title information signal St200. Note that the extracted title information includes the video manager VMG, VTS information VTSI, the PGC information entries C_PBI #j, and the cell presentation time C_PBTM.

Note that the bitstream control data St63 is generated in pack units as shown in Fig. 19, and is supplied from the stream buffer 2400 to the decoding system controller 2300, to which the stream buffer 2400 is connected.

The synchronizer 2900 is connected to the decoding system controller 2300 from which it receives the system clock reference SCR contained in the synchronization control data St81 to set the

internal system clock STC and supply the reset system clock St79 to the decoding system controller 2300.

Based on this system clock St79, the decoding system controller 2300 also generates the stream read signal St65 at a specific interval and outputs the read signal St65 to the stream buffer 2400. Note that the read unit in this case is the pack.

The method of generating the stream read signal St65 is described next.

The decoding system controller 2300 compares the system clock reference SCR contained in the stream control data extracted from the stream buffer 2400 with the system clock St79 supplied from the synchronizer 2900, and generates the read request signal St65 when the system clock St79 is greater than the system clock reference SCR of the bitstream control data St63. Pack transfers are controlled by executing this control process on a pack unit.

Based on the scenario selection data St51, the decoding system controller 2300 generates the decoding signal St69 defining the stream Ids for the video, sub-picture, and audio bitstreams

corresponding to the selected scenario, and outputs to the system decoder 2500.

When a title contains plural audio tracks, e.g. audio tracks in Japanese, English, French, and/or other languages, and plural sub-picture tracks for subtitles in Japanese, English, French, and/or other languages, for example, a discrete ID is assigned to each of the language tracks. As described above with reference to Fig. 19, a stream ID is assigned to the video data and MPEG audio data, and a substream ID is assigned to the sub-picture data, AC-3 audio data, linear PCM data, and navigation pack NV information. While the user need never be aware of these ID numbers, the user can select the language of the audio and/or subtitles using the scenario selector 2100. If English language audio is selected, for example, the ID corresponding to the English audio track is sent to the decoding system controller 2300 as scenario selection data St51. The decoding system controller 2300 then adds this ID to the decoding signal St69 output to the system decoder 2500.

Based on the instructions contained in the decoding signal St69, the system decoder 2500 respectively outputs the video, sub-picture, and

audio bitstreams input from the stream buffer 2400 to the video buffer 2600, sub-picture buffer 2700, and audio buffer 2800 as the encoded video stream St71, encoded sub-picture stream St73, and encoded audio stream St75. Thus, when the stream ID input from the scenario selector 2100 and the pack ID input from the stream buffer 2400 match, the system decoder 2500 outputs the corresponding packs to the respective buffers (i.e., the video buffer 2600, sub-picture buffer 2700, and audio buffer 2800).

The system decoder 2500 detects the presentation time stamp PTS and decoding time stamp DTS of the smallest control unit in each bitstream St67 to generate the time information signal St77. This time information signal St77 is supplied to the synchronizer 2900 through the decoding system controller 2300 as the synchronization control data St81.

Based on this synchronization control data St81, the synchronizer 2900 determines the decoding start timing whereby each of the bitstreams will be arranged in the correct sequence after decoding, and then generates and inputs the video stream decoding start signal St89 to the video decoder 3801 based on this decoding timing. The

synchronizer 2900 also generates and supplies the sub-picture decoding start signal St91 and audio stream decoding start signal St93 to the sub-picture decoder 3100 and audio decoder 3200, respectively.

The video decoder 3801 generates the video output request signal St84 based on the video stream decoding start signal St89, and outputs to the video buffer 2600. In response to the video output request signal St84, the video buffer 2600 outputs the video stream St83 to the video decoder 3801. The video decoder 3801 thus detects the presentation time information contained in the video stream St83, and disables the video output request signal St84 when the length of the received video stream St83 is equivalent to the specified presentation time. A video stream equal in length to the specified presentation time is thus decoded by the video decoder 3801, which outputs the reproduced video signal St95 to the reordering buffer 3300 and selector 3400.

Because the encoded video stream is coded using the interframe correlations between pictures, the coded order and display order do not necessarily match on a frame unit basis. The video

cannot, therefore, be displayed in the decoded order. The decoded frames are therefore temporarily stored to the reordering buffer 3300. The synchronizer 2900 therefore controls the switching signal St103 so that the reproduced video signal St95 output from the video decoder 3800 and the reordering buffer output St97 are appropriately selected and output in the display order to the synthesizer 3500.

The sub-picture decoder 3100 similarly generates the sub-picture output request signal St86 based on the sub-picture decoding start signal St91, and outputs to the sub-picture buffer 2700. In response to the sub-picture output request signal St86, the sub-picture buffer 2700 outputs the sub-picture stream St85 to the sub-picture decoder 3100. Based on the presentation time information contained in the sub-picture stream St85, the sub-picture decoder 3100 decodes a length of the sub-picture stream St85 corresponding to the specified presentation time to reproduce and supply to the synthesizer 3500 the sub-picture signal St99.

The synthesizer 3500 superimposes the selector 3400 output with the sub-picture signal

St99 to generate and output the video signal St105 to the video data output terminal 3600.

The audio decoder 3200 generates and supplies to the audio buffer 2800 the audio output request signal St88 based on the audio stream decoding start signal St93. The audio buffer 2800 thus outputs the audio stream St87 to the audio decoder 3200. The audio decoder 3200 decodes a length of the audio stream St87 corresponding to the specified presentation time based on the presentation time information contained in the audio stream St87, and outputs the decoded audio stream St101 to the audio data output terminal 3700.

It is thus possible to reproduce a user-defined multimedia bitstream MBS in real-time according to a user-defined scenario. More specifically, each time the user selects a different scenario, the DVD decoder DCD is able to reproduce the title content desired by the user in the desired sequence by reproducing the multimedia bitstream MBS corresponding to the selected scenario.

It should be noted that the decoding system controller 2300 may supply the title

information signal St200 to the scenario selector 2100 by means of the infrared communications device mentioned above or another means. Interactive scenario selection controlled by the user can also be made possible by the scenario selector 2100 extracting the title information recorded to the optical disk M from the file data structure area FDS of the bitstream control data St63 contained in the title information signal St200, and displaying this title information on a display for user selection.

Note, further, that the stream buffer 2400, video buffer 2600, sub-picture buffer 2700, audio buffer 2800, and reordering buffer 3300 are expressed above and in the figures as separate entities because they are functionally different. It will be obvious, however, that a single buffer memory can be controlled to provide the same discrete functionality by time-share controlled use of a buffer memory with an operating speed plural times faster than the read and write rates of these separate buffers.

Multi-scene control

The concept of multiple angle scene control according to the present invention is

described below with reference to Fig. 21. As described above, titles that can be played back with numerous variations are constructed from basic scene periods containing data common to each title, and multi-scene periods comprising groups of different scenes corresponding to the various scenario requests. In Fig. 21, scenes 1, 5, and 8 are the common scenes of the basic scene periods. The multi-angle scenes (angles 1, 2, and 3) between scenes 1 and 5, and the parental locked scenes (scenes 6 and 7) between scenes 5 and 8, are the multi-scene periods.

Scenes taken from different angles, i.e., angles 1, 2, and 3 in this example, can be dynamically selected and reproduced during playback in the multi-angle scene period. In the parental locked scene period, however, only one of the available scenes, scenes 6 and 7, having different content can be selected, and must be selected statically before playback begins.

Which of these scenes from the multi-scene periods is to be selected and reproduced is defined by the user operating the scenario selector 2100 and thereby generating the scenario selection data St51. In scenario 1 in Fig. 21 the user can freely

select any of the multi-angle scenes, and scene 6 has been preselected for output in the parental locked scene period. Similarly in scenario 2, the user can freely select any of the multi-angle scenes, and scene 7 has been preselected for output in the parental locked scene period.

With reference to Figs. 30 and 31, furthermore, the contents of the program chain information VTS_PGCI is described. In Fig. 30, the case that a scenario requested by the user is shown with respect to a VTSI data construction. The scenario 1 and scenario 2 shown in Fig. 21 are described as program chain information VTS_PGC#1 and VTS_PGC#2. VTS_PGC#1 describing the scenario 1 consists of cell playback information C_PBI#1 corresponding to scene 1, C_PBI#2, C_PBI#3, and C_PBI#4 within a multi-angle cell block, C_PBI#5 corresponding to scene 5, C_PBI#6 corresponding to scene 6, and C_PBI#7 corresponding to scene 8.

VTS_PGCI#2 describing the scenario 2 consists of cell playback information C_PBI#1 corresponding to scene 1, C_PBI#2, C_PBI#3, and C_PBI#4 within a multi-angle cell block corresponding to a multi-angle scene, C_PBI#5 corresponding to scene 5, C_PBI#6 corresponding to

scene 7, and C_PBI#7 corresponding to scene 8. According to the digital video system data structure, a scene which is a control unit of a scenario is described as a cell which is a unit thereunder, thus a scenario requested by a user can be obtained.

In Fig. 31, the case that a scenario requested by the user shown in Fig. 21 is shown with respect to a VOB data construction VTSTT_VOBS. As specifically shown in Fig. 31, the two scenarios 1 and 2 use the same VOB data in common. With respect to a single scene commonly owned by each scenario, VOB#1 corresponding to scene 1, VOB#5 corresponding to scene 5, and VOB#8 corresponding to scene 8 are arranged in non-interleaved block which is the contiguous block.

With respect to the multi-angle data commonly owned by scenarios 1 and 2, one angle scene data is constructed by a single VOB. Specifically speaking, angle 1 is constructed by VOB#2, and angle 2 is constructed by VOB#3, angle 3 is constructed by VOB#4. Thus constructed multi-angle data is formed as the interleaved block for the sake of switching between each angle and seamless reproduction of each angle data. Scenes 6

and 7 peculiar to scenarios 1 and 2, respectively, are formed as the interleaved block for the sake of seamless reproduction between common scenes before and behind thereof as well as seamless reproduction between each scene.

As described in the above, the user's requesting scenario shown in Fig. 21 can be realized by utilizing the video title playback control information shown in Fig. 30 and the title playback VOB data structure shown in Fig. 31.

Seamless playback

The seamless playback capability briefly mentioned above with regard to the digital video disk system data structure is described below. Note that seamless playback refers to the reproduction in a digital video disk system of multimedia data including video, audio, and sub-picture data without intermittent breaks in the data or information between basic scene periods, between basic scene periods and multi-scene periods, and between multi-scene periods.

Hardware factors contributing to intermittent playback of this data and title content include decoder underflow, i.e., an

imbalance between the source data input speed and the decoding speed of the input source data.

Other factors relate to the properties of the playback data. When the playback data is data that must be continuously reproduced for a constant time unit in order for the user to understand the content or information, e.g., audio data, data continuity is lost when the required continuous presentation time cannot be assured. Reproduction of such information whereby the required continuity is assured is referred to as "contiguous information reproduction," or "seamless information reproduction." Reproduction of this information when the required continuity cannot be assured is referred to as "non-continuous information reproduction," or "non-seamless information reproduction." It is obvious that continuous information reproduction and non-continuous information reproduction are, respectively, seamless and non-seamless reproduction.

Note that seamless reproduction can be further categorized as seamless data reproduction and seamless information reproduction. Seamless data reproduction is defined as preventing physical blanks or interruptions in the data playback

(intermittent reproduction) as a result of a buffer underflow state, for example. Seamless information reproduction is defined as preventing apparent interruptions in the information when perceived by the user (intermittent presentation) when recognizing information from the playback data where there are no actual physical breaks in the data reproduction.

Details of Seamless playback

The specific method enabling seamless reproduction as thus described is described later below with reference to Figs. 23 and 24.

Interleaving

The DVD data system streams described above are recorded using an appropriate authoring encoder EC as a movie or other multimedia title on a DVD recording medium. Note that the following description refers to a movie as the multimedia title being processed, but it will be obvious that the invention shall not be so limited.

Supplying a single movie in a format enabling the movie to be used in plural different cultural regions or countries requires the script to be recorded in the various languages used in those regions or countries. It may even

necessitate editing the content to conform to the mores and moral expectations of different cultures. Even using such a large-capacity storage system as the DVD system, however, it is necessary to reduce the bit rate, and therefore the image quality, if plural full-length titles edited from a single common source title are recorded to a single disk. This problem can be solved by recording the common parts of plural titles only once, and recording the segments different in each title for each different title only. This method makes it possible to record plural titles for different countries or cultures to a single optical disk without reducing the bit rate, and, therefore, retaining high image quality.

As shown in Fig. 21, the titles recorded to a single optical disk contain basic scene periods of scenes common to all scenarios, and multi-scene periods containing scenes specific to certain scenarios, to provide parental lock control and multi-angle scene control functions.

In the case of the parental lock control function, titles containing sex scenes, violent scenes, or other scenes deemed unsuitable for children, i.e., so-called "adult scenes," are

recorded with a combination of common scenes, adult scenes, and children's scenes. These title streams are achieved by arraying the adult and children's scenes to multi-scene periods between the common basic scene periods.

Multi-angle control can be achieved in a conventional single-angle title by recording plural multimedia scenes obtained by recording the subjects from the desired plural camera angles to the multi-scene periods arrayed between the common basic scene periods. Note, however, that while these plural scenes are described here as scenes recorded from different camera angles (positions), it will be obvious that the scenes may be recorded from the same camera angle but at different times, data generated by computer graphics, or other video data.

When data is shared between different scenarios of a single title, it is obviously necessary to move the laser beam LS from the common scene data to the non-common scene data during reproduction, i.e., to move the optical pickup to a different position on the DVD recording medium RC1. The problem here is that the time required to move the optical pickup makes it difficult to continue

reproduction without creating breaks in the audio or video, i.e., to sustain seamless reproduction. This problem can be theoretically solved by providing a track buffer (stream buffer 2400) to delay data output an amount equivalent to the worst access time. In general, data recorded to an optical disk is read by the optical pickup, appropriately processed, and temporarily stored to the track buffer. The stored data is subsequently decoded and reproduced as video or audio data.

Definition of Interleaving

To thus enable the user to selectively excise scenes and choose from among plural scenes, a state wherein non-selected scene data is recorded inserted between common scene data and selective scene data necessarily occurs because the data units associated with individual scenes are contiguously recorded to the recording tracks of the recording medium. If data is then read in the recorded sequence, non-selected scene data must be accessed before accessing and decoding the selected scene data, and seamless connections with the selected scene is difficult. The excellent random access characteristics of the digital video disk

system, however, make seamless connections with the selected scenes possible.

In other words, by splitting scene-specific data into plural units of a specified data size, and interleaving plural split data units for different scenes in a predefined sequence that is recorded to disk within the jumping range whereby an data underflow state does not occur, it is possible to reproduce the selected scenes without data interruptions by intermittently accessing and decoding the data specific to the selected scenes using these split data units. Seamless data reproduction is thereby assured.

Interleaved block and Interleave unit

The interleaving method enabling seamless data reproduction according to the present invention is described below with reference to Fig. 24 and Fig. 67. Shown in Fig. 24 is a case from which three scenarios may be derived, i.e., branching from one video object VOB-A to one of plural video objects VOB-B, VOB-C, and VOB-D, and then merging back again to a single video object VOB-E. The actual arrangement of these blocks recorded to a data recording track TR on disk is shown in Fig. 67.

Referring to Fig. 67, VOB-A and VOB-E are video objects with independent playback start and end times, and are in principle arrayed to contiguous block regions. As shown in Fig. 24, the playback start and end times of VOB-B, VOB-C, and VOB-D are aligned during interleaving. The interleaved data blocks are then recorded to disk to a contiguous interleaved block region. The contiguous block regions and interleaved block regions are then written to disk in the track path Dr direction in the playback sequence. Plural video objects VOB, i.e., interleaved video objects VOBS, arrayed to the data recording track TR are shown in Fig. 67.

Referring to Fig. 67, data regions to which data is continuously arrayed are called "blocks," of which there are two types: "contiguous block regions" in which VOB with discrete starting and end points are contiguously arrayed, and "interleaved block regions" in which plural VOB with aligned starting and end points are interleaved. The respective blocks are arrayed as shown in Fig. 68 in the playback sequence, i.e., block 1, block 2, block 3, . . . block 7.

As shown in Fig. 68, the VTS title VOBS (VTSTT_VOBS) consist of blocks 1 - 7, inclusive. Block 1 contains VOB 1 alone. Blocks 2, 3, 5, and 7 similarly discretely contain VOBS 2, 3, 6, and 10. Blocks 2, 3, 5, and 7 are thus contiguous block regions.

Block 4, however, contains VOB 4 and VOB 5 interleaved together, while block 6 contains VOB 7, VOB 8, and VOB 9 interleaved together. Blocks 4 and 6 are thus interleaved block regions.

The internal data structure of the contiguous block regions is shown in Fig. 69 with VOB-i and VOB-j arrayed as the contiguous blocks in the VOBs. As described with reference to Fig. 16, VOB-i and VOB-j inside the contiguous block regions are further logically divided into cells as the playback unit. Both VOB-i and VOB-j in this figure are shown comprising three cells CELL #1, CELL #2, and CELL #3.

Each cell comprises one or more video object unit VOBU with the video object unit VOBU defining the boundaries of the cell. Each cell also contains information identifying the position of the cell in the program chain PGC (the playback control information of the digital video disk

system). More specifically, this position information is the address of the first and last VOBU in the cell. As also shown in Fig. 69, these VOB and the cells defined therein are also recorded to a contiguous block region so that contiguous blocks are contiguously reproduced. Reproducing these contiguous blocks is therefore no problem.

The internal data structure of the interleaved block regions is shown in Fig. 70. In the interleaved block regions each video object VOB is divided into interleaved units ILVU, and the interleaved units ILVU associated with each VOB are alternately arrayed. Cell boundaries are defined independently of the interleaved units ILVU. For example, VOB-k is divided into four interleaved units ILVUk1, ILVUk2, ILVUk3, and ILVUk4, and are confined by a single cell CELL#k. VOB-k is likewise divided into four interleaved units ILVUm1, ILVUm2, ILVUm3, and ILVUm4, and is confined by a sincle cell CELL#m. Note that instead of a single cell CELL#k or CELL#m, each of VOB-k and VOB-m can be divided into more than two cells. The interleaved units ILVU thus contains both audio and video data.

In the example shown in Fig. 70, the interleaved units ILVUk1, ILVUk2, ILVUk3, and

ILVUk4, and ILVUm1, ILVUm2, ILVUm3, and ILVUm4, from two different video objects VOB-k and VOB-m are alternately arrayed within a single interleaved block. By interleaving the interleaved units ILVU of two video objects VOB in this sequence, it is possible to achieve seamless reproduction branching from one scene to one of plural scenes, and from one of plural scenes to one scene.

Multi-scene control

The multi-scene period is described together with the concept of multi-scene control according to the present invention using by way of example a title comprising scenes recorded from different angles.

Each scene in multi-scene control is recorded from the same angle, but may be recorded at different times or may even be computer graphics data. The multi-angle scene periods may therefore also be called multi-scene periods.

Parental control

The concept of recording plural titles comprising alternative scenes for such functions as parental lock control and recording director's cuts is described below using Fig. 40.

An example of a multi-rated title stream providing for parental lock control is shown in Fig. 40. When so-called "adult scenes" containing sex, violence, or other scenes deemed unsuitable for children are contained in a title implementing parental lock control, the title stream is recorded with a combination of common system streams SSa, SSb, and Sse, an adult-oriented system stream SSc containing the adult scenes, and a child-oriented system stream SSd containing only the scenes suitable for children. Title streams such as this are recorded as a multi-scene system stream containing the adult-oriented system stream Ssc and the child-oriented system stream Ssd arrayed to the multi-scene period between common system streams Ssb and Sse.

The relationship between each of the component titles and the system stream recorded to the program chain PGC of a title stream thus comprised is described below.

The adult-oriented title program chain PGC1 comprises in sequence the common system streams Ssa and Ssb, the adult-oriented system stream Ssc, and the common system stream Sse. The child-oriented title program chain PGC2 comprises

in sequence the common system streams Ssa and Ssb, the child-oriented system stream Ssd, and the common system stream Sse.

By thus arraying the adult-oriented system stream Ssc and child-oriented system stream Ssd to a multi-scene period, the decoding method previously described can reproduce the title containing adult-oriented content by reproducing the common system streams Ssa and Ssb, then selecting and reproducing the adult-oriented system stream Ssc, and then reproducing the common system stream Sse as instructed by the adult-oriented title program chain PGC1. By alternatively following the child-oriented title program chain PGC2 and selecting the child-oriented system stream Ssd in the multi-scene period, a child-oriented title from which the adult-oriented scenes have been expurgated can be reproduced.

This method of providing in the title stream a multi-scene period containing plural alternative scenes, selecting which of the scenes in the multi-scene period are to be reproduced before playback begins, and generating plural titles containing essentially the same title

content but different scenes in part, is called parental lock control.

Note that parental lock control is so named because of the perceived need to protect children from undesirable content. From the perspective of system stream processing, however, parental lock control is a technology for statically generating different title streams by means of the user pre-selecting specific scenes from a multi-scene period. Note, further, that this contrasts with multi-angle scene control, which is a technology for dynamically changing the content of a single title by means of the user selecting scenes from the multi-scene period freely and in real-time during title playback.

This parental lock control technology can also be used to enable title stream editing such as when making the director's cut. The director's cut refers to the process of editing certain scenes from a movie to, for example, shorten the total presentation time. This may be necessary, for example, to edit a feature-length movie for viewing on an airplane where the presentation time is too long for viewing within the flight time or certain content may not be acceptable. The movie director

thus determines which scenes may be cut to shorten the movie. The title can then be recorded with both a full-length, unedited system stream and an edited system stream in which the edited scenes are recorded to multi-scene periods. At the transition from one system stream to another system stream in such applications, parental lock control must be able to maintain smooth playback image output. More specifically, seamless data reproduction whereby a data underflow state does not occur in the audio, video, or other buffers, and seamless information reproduction whereby no unnatural interruptions are audibly or visibly perceived in the audio and video playback, are necessary.

Multi-angle control

The concept of multi-angle scene control in the present invention is described next with reference to Fig. 33. In general, multimedia titles are obtained by recording both the audio and video information (collectively "recording" below) of the subject over time T. The angled scene blocks #SC1, #SM1, #SM2, #SM3, and #SC3 represent the multimedia scenes obtained at recording unit times T1, T2, and T3 by recording the subject at respective camera angles. Scenes #SM1, #SM2, and

#SM3 are recorded at mutually different (first, second, and third) camera angles during recording unit time T2, and are referenced below as the first, second, and third angled scenes.

Note that the multi-scene periods referenced herein are basically assumed to comprise scenes recorded from different angles. The scenes may, however, be recorded from the same angle but at different times, or they may be computer graphics data. The multi-angle scene periods are thus the multi-scene periods from which plural scenes can be selected for presentation in the same time period, whether or not the scenes are actually recorded at different camera angles.

Scenes #SC1 and #SC3 are scenes recorded at the same common camera angle during recording unit times T1 and T3, i.e., before and after the multi-angle scenes. These scenes are therefore called "common angle scenes." Note that one of the multiple camera angles used in the multi-angle scenes is usually the same as the common camera angle.

To understand the relationship between these various angled scenes, multi-angle scene

control is described below using a live broadcast of a baseball game for example only.

The common angle scenes #SC1 and #SC3 are recorded at the common camera angle, which is here defined as the view from center field on the axis through the pitcher, batter, and catcher.

The first angled scene #SM1 is recorded at the first multi-camera angle, i.e., the camera angle from the backstop on the axis through the catcher, pitcher, and batter. The second angled scene #SM2 is recorded at the second multi-camera angle, i.e., the view from center field on the axis through the pitcher, batter, and catcher. Note that the second angled scene #SM2 is thus the same as the common camera angle in this example. It therefore follows that the second angled scene #SM2 is the same as the common angle scene #SC2 recorded during recording unit time T2. The third angled scene #SM3 is recorded at the third multi-camera angle, i.e., the camera angle from the backstop focusing on the infield.

The presentation times of the multiple angle scenes #SM1, #SM2, and #SM3 overlap in recording unit time T2; this period is called the "multi-angle scene period." By freely selecting

one of the multiple angle scenes #SM1, #SM2, and #SM3 in this multi-angle scene period, the viewer is able to change his or her virtual viewing position to enjoy a different view of the game as though the actual camera angle is changed. Note that while there appears to be a time gap between common angle scenes #SC1 and #SC3 and the multiple angle scenes #SM1, #SM2, and #SM3 in Fig. 33, this is simply to facilitate the use of arrows in the figure for easier description of the data reproduction paths reproduced by selecting different angled scenes. There is no actual time gap during playback.

Multi-angle scene control of the system stream based on the present invention is described next with reference to Fig. 23 from the perspective of connecting data blocks. The multimedia data corresponding to common angle scene #SC is referenced as common angle data BA, and the common angle data BA in recording unit times T1 and T3 are referenced as BA1 and BA3, respectively. The multimedia data corresponding to the multiple angle scenes #SM1, #SM2, and #SM3 are referenced as first, second, and third angle scene data MA1, MA2, and MA3. As previously described with reference to

Fig. 33, scenes from the desired angled can be viewed by selecting one of the multiple angle data units MA1, MA2, and MA3. There is also no time gap between the common angle data BA1 and BA3 and the multiple angle data units MA1, MA2, and MA3.

In the case of an MPEG system stream, however, intermittent breaks in the playback information can result between the reproduced common and multiple angle data units depending upon the content of the data at the connection between the selected multiple angle data unit MA1, MA2, and MA3 and the common angle data BA (either the first common angle data BA1 before the angle selected in the multi-angle scene period or the common angle data BA3 following the angle selected in the multi-angle scene period). The result in this case is that the title stream is not naturally reproduced as a single contiguous title, i.e., seamless data reproduction is achieved but non-seamless information reproduction results.

The multi-angle selection process whereby one of plural scenes is selectively reproduced from the multi-angle scene period with seamless information presentation to the scenes before and

after is described below with application in a digital video disk system using Fig. 23.

Changing the scene angle, i.e., selecting one of the multiple angle data units MA1, MA2, and MA3, must be completed before reproduction of the preceding common angle data BA1 is completed. It is extremely difficult, for example, to change to a different angle data unit MA2 during reproduction of common angle data BA1. This is because the multimedia data has a variable length coded MPEG data structure, which makes it difficult to find the data break points (boundaries) in the selected data blocks. The video may also be disrupted when the angle is changed because inter-frame correlations are used in the coding process. The group_of_pictures GOP processing unit of the MPEG standard contains at least one refresh frame, and closed processing not referencing frames belonging to another GOP is possible within this GOP processing unit.

In other words, if the desired angle data, e. g., MA3, is selected before reproduction reaches the multi-angle scene period, and at the latest by the time reproduction of the preceding common angle data BA1 is completed, the angle data selected from

within the multi-angle scene period can be seamlessly reproduced. However, it is extremely difficult while reproducing one angle to select and seamlessly reproduce another angle within the same multi-angle scene period. It is therefore difficult when in a multi-angle scene period to dynamically select a different angle unit presenting, for example, a view from a different camera angle.

Flow chart: encoder

The encoding information table generated by the encoding system controller 200 from information extracted from the scenario data St7 is described below referring to Fig. 27.

The encoding information table contains VOB set data streams containing plural VOB corresponding to the scene periods beginning and ending at the scene branching and connecting points, and VOB data streams corresponding to each scene. These VOB set data streams shown in Fig. 27 are the encoding information tables generated at step #100 in Fig. 34 by the encoding system controller 200 for creating the DVD multimedia stream based on the user-defined title content.

The user-defined scenario contains branching points from common scenes to plural scenes, or connection points to other common scenes. The VOB corresponding to the scene period delimited by these branching and connecting points is a VOB set, and the data generated to encode a VOB set is the VOB set data stream. The title number specified by the VOB set data stream is the title number TITLE_NO of the VOB set data stream.

The VOB Set data structure in Fig. 27 shows the data content for encoding one VOB set in the VOB set data stream, and comprises: the VOB set number VOBS_NO, the VOB number VOB_NO in the VOB set, the preceding VOB seamless connection flag VOB_Fsb, the following VOB seamless connection flag VOB_Fsf, the multi-scene flag VOB_Fp, the interleave flag VOB_Fi, the multi-angle flag VOB_Fm, the multi-angle seamless switching flag VOB_FsV, the maximum bit rate of the interleaved VOB ILV_BR, the number of interleaved VOB divisions ILV_DIV, and the minimum interleaved unit presentation time ILVU_MT.

The VOB set number VOBS_NO is a sequential number identifying the VOB set and the position of

the VOB set in the reproduction sequence of the title scenario.

The VOB number VOB_NO is a sequential number identifying the VOB and the position of the VOB in the reproduction sequence of the title scenario.

The preceding VOB seamless connection flag VOB_Fsb indicates whether a seamless connection with the preceding VOB is required for scenario reproduction.

The following VOB seamless connection flag VOB_Fsf indicates whether there is a seamless connection with the following VOB during scenario reproduction.

The multi-scene flag VOB_Fp identifies whether the VOB set comprises plural video objects VOB.

The interleave flag VOB_Fi identifies whether the VOB in the VOB set are interleaved.

The multi-angle flag VOB_Fm identifies whether the VOB set is a multi-angle set.

The multi-angle seamless switching flag VOB_FsV identifies whether angle changes within the multi-angle scene period are seamless or not.

The maximum bit rate of the interleaved VOB ILV_BR defines the maximum bit rate of the interleaved VOBs.

The number of interleaved VOB divisions ILV_DIV identifies the number of interleave units in the interleaved VOB.

The minimum interleave unit presentation time ILVU_MT defines the time that can be reproduced when the bit rate of the smallest interleave unit at which a track buffer data underflow state does not occur is the maximum bit rate of the interleaved VOB ILV_BR during interleaved block reproduction.

The encoding information table for each VOB generated by the encoding system controller 200 based on the scenario data St7 is described below referring to Fig. 28. The VOB encoding parameters described below and supplied to the video encoder 300, audio encoder 700, and system encoder 900 for stream encoding are produced based on this encoding information table.

The VOB data streams shown in Fig. 28 are the encoding information tables generated at step #100 in Fig. 34 by the encoding system controller

200 for creating the DVD multimedia stream based on the user-defined title content.

The encoding unit is the video object VOB, and the data generated to encode each video object VOB is the VOB data stream. For example, a VOB set comprising three angle scenes comprises three video objects VOB. The data structure shown in Fig. 28 shows the content of the data for encoding one VOB in the VOB data stream.

The VOB data structure contains the video material start time VOB_VST, the video material end time VOB_VEND, the video signal type VOB_V_KIND, the video encoding bit rate V_BR, the audio material start time VOB_AST, the audio material end time VOB_AEND, the audio coding method VOB_A_KIND, and the audio encoding bit rate A_BR.

The video material start time VOB_VST is the video encoding start time corresponding to the time of the video signal.

The video material end time VOB_VEND is the video encoding end time corresponding to the time of the video signal.

The video material type VOB_V_KIND identifies whether the encoded material is in the NTSC or PAL format, for example, or is photographic

material (a movie, for example) converted to a television broadcast format (so-called telecine conversion).

The video encoding bit rate V_BR is the bit rate at which the video signal is encoded.

The audio material start time VOB_AST is the audio encoding start time corresponding to the time of the audio signal.

The audio material end time VOB_AEND is the audio encoding end time corresponding to the time of the audio signal.

The audio coding method VOB_A_KIND identifies the audio encoding method as AC-3, MPEG, or linear PCM, for example.

The audio encoding bit rate A_BR is the bit rate at which the audio signal is encoded.

The encoding parameters used by the video encoder 300, sub-picture encoder 500, and audio encoder 700, and system encoder 900 for VOB encoding are shown in Fig. 29. The encoding parameters include: the VOB number VOB_NO, video encode start time V_STTM, video encode end time V_ENDTM, the video encode mode V_ENCMD, the video encode bit rate V_RATE, the maximum video encode bit rate V_MRATE, the GOP structure fixing flag

GOP_Fxflag, the video encode GOP structure GOPST, the initial video encode data V_INTST, the last video encode data V_ENDST, the audio encode start time A_STTM, the audio encode end time A_ENDTM, the audio encode bit rate A_RATE, the audio encode method A_ENCMD, the audio start gap A_STGAP, the audio end gap A_ENDGAP, the preceding VOB number B_VOB_NO, and the following VOB number F_VOB_NO.

The VOB number VOB_NO is a sequential number identifying the VOB and the position of the VOB in the reproduction sequence of the title scenario.

The video encode start time V_STTM is the start time of video material encoding.

The video encode end time V_ENDTM is the end time of video material encoding.

The video encode mode V_ENCMD is an encoding mode for declaring whether reverse telecine conversion shall be accomplished during video encoding to enable efficient coding when the video material is telecine converted material.

The video encode bit rate V_RATE is the average bit rate of video encoding.

The maximum video encode bit rate V_MRATE is the maximum bit rate of video encoding.

The GOP structure fixing flag GOP_Fxflag specifies whether encoding is accomplished without changing the GOP structure in the middle of the video encoding process. This is a useful parameter for declaring whether seamless switch is enabled in a multi-angle scene period.

The video encode GOP structure GOPST is the GOP structure data from encoding.

The initial video encode data V_INTST sets the initial value of the VBV buffer (decoder buffer) at the start of video encoding, and is referenced during video decoding to initialize the decoding buffer. This is a useful parameter for declaring seamless reproduction with the preceding encoded video stream.

The last video encode data V_ENDST sets the end value of the VBV buffer (decoder buffer) at the end of video encoding, and is referenced during video decoding to initialize the decoding buffer. This is a useful parameter for declaring seamless reproduction with the preceding encoded video stream.

The audio encode start time A_STTM is the start time of audio material encoding.

The audio encode end time A_ENDTM is the end time of audio material encoding.

The audio encode bit rate A_RATE is the bit rate used for audio encoding.

The audio encode method A_ENCMD identifies the audio encoding method as AC-3, MPEG, or linear PCM, for example.

The audio start gap A_STGAP is the time offset between the start of the audio and video presentation at the beginning of a VOB. This is a useful parameter for declaring seamless reproduction with the preceding encoded system stream.

The audio end gap A_ENDGAP is the time offset between the end of the audio and video presentation at the end of a VOB. This is a useful parameter for declaring seamless reproduction with the preceding encoded system stream.

The preceding VOB number B_VOB_NO is the VOB_NO of the preceding VOB when there is a seamlessly connected preceding VOB.

The following VOB number F_VOB_NO is the VOB_NO of the following VOB when there is a seamlessly connected following VOB.

The operation of a DVD encoder ECD according to the present invention is described below with reference to the flow chart in Fig. 34. Note that the steps shown with a double line are subroutines. It should be obvious that while the operation described below relates specifically in this case to the DVD encoder ECD of the present invention, the operation described also applies to an authoring encoder EC.

At step #100, the user inputs the editing commands according to the user-defined scenario while confirming the content of the multimedia source data streams St1, St2, and St3.

At step #200, the scenario editor 100 generates the scenario data St7 containing the above edit command information according to the user's editing instructions.

When generating the scenario data St7 in step #200, the user editing commands related to multi-angle and parental lock multi-scene periods in which interleaving is presumed must be input to satisfy the following conditions.

First, the VOB maximum bit rate must be set to assure sufficient image quality, and the track buffer capacity, jump performance, jump time,

and jump distance of the DVD decoder DCD used as the reproduction apparatus of the DVD encoded data must be determined. Based on these values, the reproduction time of the shortest interleaved unit is obtained from equations 3 and 4. Based on the reproduction time of each scene in the multi-scene period, it must then be determined whether equations 5 and 6 are satisfied. If equations 5 and 6 are not satisfied, the user must change the edit commands until equations 5 and 6 are satisfied by, for example, connecting part of the following scene to each scene in the multi-scene period.

When multi-angle edit commands are used, equation 7 must be satisfied for seamless switching, and edit commands matching the audio reproduction time with the reproduction time of each scene in each angle must be entered. If non-seamless switching is used, the user must enter commands to satisfy equation 8.

At step #300, the encoding system controller 200 first determines whether the target scene is to be seamlessly connected to the preceding scene based on the scenario data St7.

Note that when the preceding scene period is a multi-scene period comprising plural scenes

but the presently selected target scene is a common scene (not in a multi-scene period), a seamless connection refers to seamlessly connecting the target scene with any one of the scenes contained in the preceding multi-scene period. When the target scene is a multi-scene period, a seamless connection still refers to seamlessly connecting the target scene with any one of the scenes from the same multi-scene period.

If step #300 returns NO, i.e., a non-seamless connection is valid, the procedure moves to step #400.

At step #400, the encoding system controller 200 resets the preceding VOB seamless connection flag VOB_Fsb indicating whether there is a seamless connection between the target and preceding scenes. The procedure then moves to step #600.

On the other hand, if step #300 returns YES, i.e., there is a seamless connection to the preceding scene, the procedure moves to step #500.

At step #500 the encoding system controller 200 sets the preceding VOB seamless connection flag VOB_Fsb. The procedure then moves to step #600.

At step #600 the encoding system controller 200 determines whether there is a seamless connection between the target and following scenes based on scenario data St7. If step #600 returns NO, i.e., a non-seamless connection is valid, the procedure moves to step #700.

At step #700, the encoding system controller 200 resets the following VOB seamless connection flag VOB_Fsf indicating whether there is a seamless connection with the following scene. The procedure then moves to step #900.

However, if step #600 returns YES, i.e., there is a seamless connection to the following scene, the procedure moves to step #800.

At step #800 the encoding system controller 200 sets the following VOB seamless connection flag VOB_Fsf. The procedure then moves to step #900.

At step #900 the encoding system controller 200 determines whether there is more than connection target scene, i.e., whether a multi-scene period is selected, based on the scenario data St7. As previously described, there are two possible control methods in multi-scene

periods: parental lock control whereby only one of plural possible reproduction paths that can be constructed from the scenes in the multi-scene period is reproduced, and multi-angle control whereby the reproduction path can be switched within the multi-scene period to present different viewing angles.

If step #900 returns NO, i.e., there are not multiple scenes, the procedure moves to step #1000.

At step #1000 the multi-scene flag VOB_Fp identifying whether the VOB set comprises plural video objects VOB (a multi-scene period is selected) is reset, and the procedure moves to step #1800 for encode parameter production. This encode parameter production subroutine is described below.

However, if step #900 returns YES, there is a multi-scene connection, the procedure moves to step #1100.

At step #1100, the multi-scene flag VOB_Fp is set, and the procedure moves to step #1200 whereat it is judged whether a multi-angle connection is selected, or not.

At step #1200 it is determined whether a change is made between plural scenes in the

multi-scene period, i.e., whether a multi-angle scene period is selected. If step #1200 returns NO, i.e., no scene change is allowed in the multi-scene period as parental lock control reproducing only one reproduction path has been selected, the procedure moves to step #1300.

At step #1300 the multi-angle flag VOB_Fm identifying whether the target connection scene is a multi-angle scene is reset, and the procedure moves to step #1302.

At step #1302 it is determined whether either the preceding VOB seamless connection flag VOB_Fsb or following VOB seamless connection flag VOB_Fsf is set. If step #1302 returns YES, i.e., the target connection scene seamlessly connects to the preceding, the following, or both the preceding and following scenes, the procedure moves to step #1304.

At step #1304 the interleave flag VOB_Fi identifying whether the VOB, the encoded data of the target scene, is interleaved is set. The procedure then moves to step #1800.

However, if step #1302 returns NO, i.e., the target connection scene does not seamlessly

connect to the preceding or following scene, the procedure moves to step #1306.

At step #1306 the interleave flag VOB_Fi is reset, and the procedure moves to step #1800.

If step #1200 returns YES, however, i. e., there is a multi-angle connection, the procedure moves to step #1400.

At step #1400, the multi-angle flag VOB_Fm and interleave flag VOB_Fi are set, and the procedure moves to step #1500.

At step #1500 the encoding system controller 200 determines whether the audio and video can be seamlessly switched in a multi-angle scene period, i.e., at a reproduction unit smaller than the VOB, based on the scenario data St7. If step #1500 returns NO, i.e., non-seamless switching occurs, the procedure moves to step #1600.

At step #1600 the multi-angle seamless switching flag VOB_FsV indicating whether angle changes within the multi-angle scene period are seamless or not is reset, and the procedure moves to step #1800.

However, if step #1500 returns YES, i.e., seamless switching occurs, the procedure moves to step #1700.

At step #1700 the multi-angle seamless switching flag VOB_FsV is set, and the procedure moves to step #1800.

Therefore, as shown by the flow chart in Fig. 34, encode parameter production (step #1800) is only begun after the editing information is detected from the above flag settings in the scenario data St7 reflecting the user-defined editing instructions.

Based on the user-defined editing instructions detected from the above flag settings in the scenario data St7, information is added to the encoding information tables for the VOB Set units and VOB units as shown in Figs. 27 and 28 to encode the source streams, and the encoding parameters of the VOB data units shown in Fig. 29 are produced, in step #1800. The procedure then moves to step #1900 for audio and video encoding.

The encode parameter production steps (step #1800) are described in greater detail below referring to Figs. 35, 36, 37, and 38.

Based on the encode parameters produced in step #1800, the video data and audio data are encoded in step #1900, and the procedure moves to step #2000.

Note that the sub-picture data is normally inserted during video reproduction on an as-needed basis, and contiguity with the preceding and following scenes is therefore not usually necessary. Moreover, the sub-picture data is normally video information for one frame, and unlike audio and video data having an extended time-base, sub-picture data is usually static, and is not normally presented continuously. Because the present invention relates specifically to seamless and non-seamless contiguous reproduction as described above, description of sub-picture data encoding is omitted herein for simplicity.

Step #2000 is the last step in a loop comprising steps #300 to step #2000, and causes this loop to be repeated as many times as there are VOB Sets. This loop formats the program chain VTS_PGC#i to contain the reproduction sequence and other reproduction information for each VOB in the title (Fig. 16) in the program chain data structure, interleaves the VOB in the multi-scene periods, and completes the VOB Set data stream and VOB data stream needed for system stream encoding. The procedure then moves to step #2100.

At step #2100 the VOB Set data stream is completed as the encoding information table by adding the total number of VOB Sets VOBS_NUM obtained as a result of the loop through step #2000 to the VOB Set data stream, and setting the number of titles TITLE_NO defining the number of scenario reproduction paths in the scenario data St7. The procedure then moves to step #2200.

System stream encoding producing the VOB (VOB#i) data in the VTS title VOBS (VTSTT_VOBS) (Fig. 16) is accomplished in step #2200 based on the encoded video stream and encoded audio stream output from step #1900, and the encode parameters in Fig. 29. The procedure then moves to step #2300.

At step #2300 the VTS information VTSI, VTSI management table VTSI_MAT, VTSPGC information table VTS_PGCIT, and the program chain information VTS_PGCI#i controlling the VOB data reproduction sequence shown in Fig. 16 are produced, and formatting to, for example, interleave the VOB contained in the multi-scene periods, is accomplished.

The encode parameter production subroutine shown as step #1800 in Fig. 34B is described next

using Figs. 35, 36, and 37 using by way of example the operation generating the encode parameters for multi-angle control.

Starting from Fig. 35, the process for generating the encode parameters of a non-seamless switching stream with multi-angle control is described first. This stream is generated when step #1500 in Fig. 34 returns NO and the following flags are set as shown: VOB_Fsb = 1 or VOB_Fsf = 1, VOB_Fp = 1, VOB_Fi = 1, VOB_Fm = 1, and VOB_FsV = 0. The following operation produces the encoding information tables shown in Fig. 27 and Fig. 28, and the encode parameters shown in Fig. 29.

At step #1812, the scenario reproduction sequence (path) contained in the scenario data St7 is extracted, the VOB Set number VOBS_NO is set, and the VOB number VOB_NO is set for one or more VOB in the VOB Set.

At step #1814 the maximum bit rate ILV_BR of the interleaved VOB is extracted from the scenario data St7, and the maximum video encode bit rate V_MRATE from the encode parameters is set based on the interleave flag VOB_Fi setting (= 1).

At step #1816, the minimum interleaved unit presentation time ILVU_MT is extracted from the scenario data St7.

At step #1818, the video encode GOP structure GOPST values N = 15 and M = 3 are set, and the GOP structure fixing flag GOP_Fxflag is set (= 1), based on the multi-scene flag VOB_Fp setting (= 1).

Step #1820 is the common VOB data setting routine, which is described below referring to the flow chart in Fig. 36. This common VOB data setting routine produces the encoding information tables shown in Figs. 27 and 28, and the encode parameters shown in Fig. 29.

At step #1822 the video material start time VOB_VST and video material end time VOB_VEND are extracted for each VOB, and the video encode start time V_STTM and video encode end time V_ENDTM are used as video encoding parameters.

At step #1824 the audio material start time VOB_AST of each VOB is extracted from the scenario data St7, and the audio encode start time A_STTM is set as an audio encoding parameter.

At step #1826 the audio material end time VOB_AEND is extracted for each VOB from the

scenario data St7, and at a time not exceeding the VOB_AEND time. This time extracted at an audio access unit (AAU) is set as the audio encode end time A_ENDTM which is an audio encoding parameter. Note that the audio access unit AAU is determined by the audio encoding method.

At step #1828 the audio start gap A_STGAP obtained from the difference between the video encode start time V_STTM and the audio encode start time A_STTM is defined as a system encode parameter.

At step #1830 the audio end gap A_ENDGAP obtained from the difference between the video encode end time V_ENDTM and the audio encode end time A_ENDTM is defined as a system encode parameter.

At step #1832 the video encoding bit rate V_BR is extracted from the scenario data St7, and the video encode bit rate V_RATE, which is the average bit rate of video encoding, is set as a video encoding parameter.

At step #1834 the audio encoding bit rate A_BR is extracted from the scenario data St7, and the audio encode bit rate A_RATE is set as an audio encoding parameter.

At step #1836 the video material type VOB_V_KIND is extracted from the scenario data St7. If the material is a film type, i.e., a movie converted to television broadcast format (so-called telecine conversion), reverse telecine conversion is set for the video encode mode V_ENCMD, and defined as a video encoding parameter.

At step #1838 the audio coding method VOB_A_KIND is extracted from the scenario data St7, and the encoding method is set as the audio encode method A_ENCMD and set as an audio encoding parameter.

At step #1840 the initial video encode data V_INTST sets the initial value of the VBV buffer to a value less than the VBV buffer end value set by the last video encode data V_ENDST, and defined as a video encoding parameter.

At step #1842 the VOB number VOB_NO of the preceding connection is set to the preceding VOB number B_VOB_NO based on the setting (= 1) of the preceding VOB seamless connection flag VOB_Fsb, and set as a system encode parameter.

At step #1844 the VOB number VOB_NO of the following connection is set to the following VOB number F_VOB_NO based on the setting (= 1) of the

following VOB seamless connection flag VOB_Fsf, and set as a system encode parameter.

The encoding information table and encode parameters are thus generated for a multi-angle VOB Set with non-seamless multi-angle switching control enabled.

The process for generating the encode parameters of a seamless switching stream with multi-angle control is described below with reference to Fig. 37. This stream is generated when step #1500 in Fig. 34 returns YES and the following flags are set as shown: VOB_Fsb = 1 or VOB_Fsf = 1, VOB_Fp = 1, VOB_Fi = 1, VOB_Fm = 1, and VOB_FsV = 1. The following operation produces the encoding information tables shown in Fig. 27 and Fig. 28, and the encode parameters shown in Fig. 29.

The following operation produces the encoding information tables shown in Fig. 27 and Fig. 28, and the encode parameters shown in Fig. 29.

At step #1850, the scenario reproduction sequence (path) contained in the scenario data St7 is extracted, the VOB Set number VOBS_NO is set,

and the VOB number VOB_NO is set for one or more VOB in the VOB Set.

At step #1852 the maximum bit rate ILV_BR of the interleaved VOB is extracted from the scenario data St7, and the maximum video encode bit rate V_MRATE from the encode parameters is set based on the interleave flag VOB_Fi setting (= 1).

At step #1854, the minimum interleaved unit presentation time ILVU_MT is extracted from the scenario data St7.

At step #1856, the video encode GOP structure GOPST values N = 15 and M = 3 are set, and the GOP structure fixing flag GOP_Fxflag is set (= 1), based on the multi-scene flag VOB_Fp setting (= 1).

At step #1858, the video encode GOP GOPST is set to "closed GOP" based on the multi-angle seamless switching flag VOB_FsV setting (= 1), and the video encoding parameters are thus defined.

Step #1860 is the common VOB data setting routine, which is as described referring to the flow chart in Fig. 35. Further description thereof is thus omitted here.

The encode parameters of a seamless switching stream with multi-angle control are thus

defined for a VOB Set with multi-angle control as described above.

The process for generating the encode parameters for a system stream in which parental lock control is implemented is described below with reference to Fig. 38. This stream is generated when step #1200 in Fig. 34 returns NO and step #1304 returns YES, i.e., the following flags are set as shown: VOB_Fsb = 1 or VOB_Fsf = 1, VOB_Fp = 1, VOB_Fi = 1, VOB_Fm = 0. The following operation produces the encoding information tables shown in Fig. 27 and Fig. 28, and the encode parameters shown in Fig. 29.

At step #1870, the scenario reproduction sequence (path) contained in the scenario data St7 is extracted, the VOB Set number VOBS_NO is set, and the VOB number VOB_NO is set for one or more VOB in the VOB Set.

At step #1872 the maximum bit rate ILV_BR of the interleaved VOB is extracted from the scenario data St7, and the maximum video encode bit rate V_MRATE from the encode parameters is set based on the interleave flag VOB_Fi setting (= 1).

At step #1872 the number of interleaved VOB divisions ILV_DIV is extracted from the scenario data St7.

Step #1876 is the common VOB data setting routine, which is as described referring to the flow chart in Fig. 35. Further description thereof is thus omitted here.

The encode parameters of a system stream in which parental lock control is implemented are thus defined for a VOB Set with multi-scene selection control enabled as described above.

The process for generating the encode parameters for a system stream containing a single scene is described below with reference to Fig. 32. This stream is generated when step #900 in Fig. 34 returns NO, i.e., when VOB_Fp=0. The following operation produces the encoding information tables shown in Fig. 27 and Fig. 28, and the encode parameters shown in Fig. 29.

At step #1880, the scenario reproduction sequence (path) contained in the scenario data St7 is extracted, the VOB Set number VOBS_NO is set, and the VOB number VOB_NO is set for one or more VOB in the VOB Set.

At step #1882 the maximum bit rate ILV_BR of the interleaved VOB is extracted from the scenario data St7, and the maximum video encode bit rate V_MRATE from the encode parameters is set based on the interleave flag VOB_Fi setting (= 1).

Step #1884 is the common VOB data setting routine, which is as described referring to the flow chart in Fig. 35. Further description thereof is thus omitted here.

These flow charts for defining the encoding information table and encode parameters thus generate the parameters for DVD video, audio, and system stream encoding by the DVD formatter.

Decoder flow charts

Disk-to-stream buffer transfer flow

The decoding information table produced by the decoding system controller 2300 based on the scenario selection data St51 is described below referring to Figs. 58 and 59. The decoding information table comprises the decoding system table shown in Fig. 58, and the decoding table shown in Fig. 59.

As shown in Fig. 58, the decoding system table comprises a scenario information register and a cell information register. The scenario

information register records the title number and other scenario reproduction information selected by the user and extracted from the scenario selection data St51. The cell information register extracts and records the information required to reproduce the cells constituting the program chain PGC based on the user-defined scenario information extracted into the scenario information register.

More specifically, the scenario information register contains plural sub-registers, i.e., the angle number ANGLE_NO_reg, VTS number VTS_NO_reg, PGC number VTS_PGCI_NO_reg, audio ID AUDIO_ID_reg, sub-picture ID SP_ID_reg, and the system clock reference SCR buffer SCR_buffer.

The angle number ANGLE_NO_reg stores which angle is reproduced when there are multiple angles in the reproduction program chain PGC.

The VTS number VTS_NO_reg records the number of the next VTS reproduced from among the plural VTS on the disk.

The PGC number VTS_PGCI_NO_reg records which of the plural program chains PGC present in the video title set VTS is to be reproduced for parental lock control or other applications.

The audio ID AUDIO_ID_reg records which of the plural audio streams in the VTS are to be reproduced.

The sub-picture ID SP_ID_reg records which of the plural sub-picture streams is to be reproduced when there are plural sub-picture streams in the VTS.

The system clock reference SCR buffer SCR_buffer is the buffer for temporarily storing the system clock reference SCR recorded to the pack header as shown in Fig. 19. As described using Fig. 26, this temporarily stored system clock reference SCR is output to the decoding system controller 2300 as the bitstream control data St63.

The cell information register contains the following sub-registers: the cell block mode CBM_reg, cell block type CBT_reg, seamless reproduction flag SPF_reg, interleaved allocation flag IAF_reg, STC resetting flag STCDF, seamless angle change flag SACF_reg, first cell VOBU start address C_FVOBU_SA_reg, and last cell VOBU start address C_LVOBU_SA_reg.

The cell block mode CBM_reg stores a value indicating whether plural cells constitute one functional block. If there are not plural cells in

one functional block, CBM_reg stores N_BLOCK. If plural cells constitute one functional block, the value F_CELL is stored as the CBM_reg value of the first cell in the block, L_CELL is stored as the CBM_reg value of the last cell in the block, and BLOCK is stored as the CBM_reg of value all cells between the first and last cells in the block.

The cell block type CBT_reg stores a value defining the type of the block indicated by the cell block mode CBM_reg. If the cell block is a multi-angle block, A_BLOCK is stored; if not, N_BLOCK is stored.

The seamless reproduction flag SPF_reg stores a value defining whether that cell is seamless connected with the cell or cell block reproduced therebefore. If a seamless connection is specified, SML is stored; if a seamless connection is not specified, NSML is stored.

The interleaved allocation flag IAF_reg stores a value identifying whether the cell exists in a contiguous or interleaved block. If the cell is part of a an interleaved block, ILVB is stored; otherwise N_ILVB is stored.

The STC resetting flag STCDF defines whether the system time clock STC used for

synchronization must be reset when the cell is reproduced; when resetting the system time clock STC is necessary, STC_RESET is stored; if resetting is not necessary, STC_NRESET is stored.

The seamless angle change flag SACF_reg stores a value indicating whether a cell in a multi-angle period should be connected seamlessly at an angle change. If the angle change is seamless, the seamless angle change flag SACF is set to SML; otherwise it is set to NSML.

The first cell VOBU start address C_FVOBU_SA_reg stores the VOBU start address of the first cell in a block. The value of this address is expressed as the distance from the logic sector of the first cell in the VTS title VOBS (VTSTT_VOBS) as measured by and expressed (stored) as the number of sectors.

The last cell VOBU start address C_LVOBU_SA_reg stores the VOBU start address of the last cell in the block. The value of this address is also expressed as the distance from the logic sector of the first cell in the VTS title VOBS (VTSTT_VOBS) measured by and expressed (stored) as the number of sectors.

The decoding table shown in Fig. 59 is described below. As shown in Fig. 59, the decoding table comprises the following registers: information registers for non-seamless multi-angle control, information registers for seamless multi-angle control, a VOBU information register, and information registers for seamless reproduction.

The information registers for non-seamless multi-angle control comprise sub-registers NSML_AGL_C1_DSTA_reg - NSML_AGL_C9_DSTA_reg.

NSML_AGL_C1_DSTA_reg - NSML_AGL_C9_DSTA_reg record the NMSL_AGL_C1_DSTA - NMSL_AGL_C9_DSTA values in the PCI packet shown in Fig. 20.

The information registers for seamless multi-angle control comprise sub-registers SML_AGL_C1_DSTA_reg - SML_AGL_C9_DSTA_reg.

SML_AGL_C1_DSTA_reg - SML_AGL_C9_DSTA_reg record the SML_AGL_C1_DSTA - SML_AGL_C9_DSTA values in the DSI packet shown in Fig. 20.

The VOBU information register stores the end pack address VOBU_EA in the DSI packet shown in Fig. 20.

The information registers for seamless reproduction comprise the following sub-registers: an interleaved unit flag ILVU_flag_reg, Unit END flag UNIT_END_flag_reg, Interleaved Unit End Address ILVU_EA_reg, Next Interleaved Unit Start Address NT_ILVU_SA_reg, the presentation start time of the first video frame in the VOB (Initial Video Frame Presentation Start Time) VOB_V_SPTM_reg, the presentation end time of the last video frame in the VOB (Final Video Frame Presentation Termination Time) VOB_V_EPTM_reg, audio reproduction stopping time 1 VOB_A_STP_PTM1_reg, audio reproduction stopping time 2 VOB_A_STP_PTM2_reg, audio reproduction stopping period 1 VOB_A_GAP_LEN1_reg, and audio reproduction stopping period 2 VOB_A_GAP_LEN2_reg.

The interleaved unit flag ILVU_flag_reg stores the value indicating whether the video object unit VOBU is in an interleaved block, and stores ILVU if it is, and N_ILVU if not.

The Unit END flag UNIT_END_flag_reg stores the value indicating whether the video object unit VOBU is the last VOBU in the interleaved unit ILVU. Because the interleaved unit ILVU is the data unit for continuous reading, the UNIT_END_flag_reg

stores END if the VOBU currently being read is the last VOBU in the interleaved unit ILVU, and otherwise stores N_END.

. The Interleaved Unit End Address ILVU_EA_reg stores the address of the last pack in the ILVU to which the VOBU belongs if the VOBU is in an interleaved block. This address is expressed as the number of sectors from the navigation pack NV of that VOBU.

The Next Interleaved Unit Start Address NT_ILVU_SA_reg stores the start address of the next interleaved unit ILVU if the VOBU is in an interleaved block. This address is also expressed as the number of sectors from the navigation pack NV of that VOBU.

The Initial Video Frame Presentation Start Time register VOB_V_SPTM_reg stores the time at which presentation of the first video frame in the VOB starts.

The Final Video Frame Presentation Termination Time register VOB_V_EPTM_reg stores the time at which presentation of the last video frame in the VOB ends.

The audio reproduction stopping time 1 VOB_A_STP_PTM1_reg stores the time at which the

audio is to be paused to enable resynchronization, and the audio reproduction stopping period 1 VOB_A_GAP_LEN1_reg stores the length of this pause period.

The audio reproduction stopping time 2 VOB_A_STP_PTM2_reg and audio reproduction stopping period 2 VOB_A_GAP_LEN2_reg store the same values.

The operation of the DVD decoder DCD according to the present invention as shown in Fig. 26 is described next below with reference to the flow chart in Fig. 60.

At step #310202 it is first determined whether a disk has been inserted. If it has, the procedure moves to step #310204.

At step #310204, the volume file structure VFS (Fig. 21) is read, and the procedure moves to step #310206.

At step #310206, the video manager VMG (Fig. 21) is read and the video title set VTS to be reproduced is extracted. The procedure then moves to step #310208.

At step #310208, the video title set menu address information VTSM_C_ADT is extracted from the VTS information VTSI, and the procedure moves to step #310210.

At step #310210 the video title set menu VTSM_VOBS is read from the disk based on the video title set menu address information VTSM_C_ADT, and the title selection menu is presented.

The user is thus able to select the desired title from this menu in step #310212. If the titles include both contiguous titles with no user-selectable content, and titles containing audio numbers, sub-picture numbers, or multi-angle scene content, the user must also enter the desired angle number. Once the user selection is completed, the procedure moves to step #310214.

At step #310214, the VTS_PGCI #i program chain (PGC) data block corresponding to the title number selected by the user is extracted from the VTSPGC information table VTS_PGCIT, and the procedure moves to step #310216.

Reproduction of the program chain PGC then begins at step #310216. When program chain PGC reproduction is finished, the decoding process ends. If a separate title is thereafter to be reproduced as determined by monitoring key entry to the scenario selector, the title menu is presented again (step #310210).

Program chain reproduction in step #310216 above is described in further detail below referring to Fig. 61. The program chain PGC reproduction routine consists of steps #31030, #31032, #31034, and #31035 as shown.

At step #31030 the decoding system table shown in Fig. 58 is defined. The angle number ANGLE_NO_reg, VTS number VTS_NO_reg, PGC number VTS_PGCI_NO_reg, audio ID AUDIO_ID_reg, and sub-picture ID SP_ID_reg are set according to the selections made by the user using the scenario selector 2100.

Once the PGC to be reproduced is determined, the corresponding cell information (PGC information entries C_PBI #j) is extracted and the cell information register is defined. The sub-registers therein that are defined are the cell block mode CBM_reg, cell block type CBT_reg, seamless reproduction flag SPF_reg, interleaved allocation flag IAF_reg, STC resetting flag STCDF, seamless angle change flag SACF_reg, first cell VOBU start address C_FVOBU_SA_reg, and last cell VOBU start address C_LVOBU_SA_reg.

Once the decoding system table is defined, the process transferring data to the stream buffer

(step #31032) and the process decoding the data in the stream buffer (step #31034) are activated in parallel.

The process transferring data to the stream buffer (step #31032) is the process of transferring data from the recording medium M to the stream buffer 2400. This is, therefore, the processing of reading the required data from the recording medium M and inputting the data to the stream buffer 2400 according to the user-selected title information and the playback control information (navigation packs NV) written in the stream.

The routine shown as step #31034 is the process for decoding the data stored to the stream buffer 2400 (Fig. 26), and outputting the decoded data to the video data output terminal 3600 and audio data output terminal 3700. Thus, is the process for decoding and reproducing the data stored to the stream buffer 2400.

Note that step #31032 and step #31034 are executed in parallel.

The processing unit of step #31032 is the cell, and as processing one cell is completed, it is determined in step #31035 whether the complete

program chain PGC has been processed. If processing the complete program chain PGC is not completed, the decoding system table is defined for the next cell in step #31030. This loop from step #31030 through step #31035 is repeated until the entire program chain PGC is processed.

The stream buffer data transfer process of step #31032 is described in further detail below referring to Fig. 62. The stream buffer data transfer process (step #31032) comprises steps #31040, #31042, #31044, #31046, and #31048 shown in the figure.

At step #31040 it is determined whether the cell is a multi-angle cell. If not, the procedure moves to step #30144.

At step #31044 the non-multi-angle cell decoding process is executed.

However, if step #30140 returns YES because the cell is a multi-angle cell, the procedure moves to step #30142 where the seamless angle change flag SACF is evaluated to determine whether seamless angle reproduction is specified.

If seamless angle reproduction is specified, the seamless multi-angle decoding process is executed in step #30146. If seamless

angle reproduction is not specified, the non-seamless multi-angle decoding process is executed in step #30148.

The non-multi-angle cell decoding process (step #31044, Fig. 62) is described further below with reference to Fig. 63. Note that the non-multi-angle cell decoding process (step #31044) comprises the steps #31050, #31052, and #31054.

The first step #31050 evaluates the interleaved allocation flag IAF_reg to determine whether the cell is in an interleaved block. If it is, the non-multi-angle interleaved block process is executed in step #31052.

The non-multi-angle interleaved block process (step #31052) processes scene branching and connection where seamless connections are specified in, for example, a multi-scene period.

However, if the cell is not in an interleaved block, the non-multi-angle contiguous block process is executed in step #31054. Note that the step #31054 process is the process executed when there is no scene branching or connection.

The non-multi-angle interleaved block process (step #31052, Fig. 63) is described further below with reference to Fig. 64.

At step #31060 the reading head 2006 is jumped to the first cell VOBU start address C_FVOBU_SA read from the C_FVOBU_SA_reg register.

More specifically, the address data C_FVOBU_SA_reg stored in the decoding system controller 2300 (Fig. 26) is input as bitstream reproduction control signal St53 to the reproduction controller 2002. The reproduction controller 2002 thus controls the recording media drive unit 2004 and signal processor 2008 to move the reading head 2006 to the specified address, data is read, error correction code ECC and other signal processing is accomplished by the signal processor 2008, and the cell start VOBU data is output as the reproduced bitstream St61 to the stream buffer 2400. The procedure then moves to step #31062.

At step #31062 the DSI packet data in the navigation pack NV (Fig. 20) is extracted in the stream buffer 2400, the decoding table is defined, and the procedure moves to step #31064. The registers set in the decoding table are the

ILVU_EA_reg, NT_ILVU_SA_reg, VOB_V_SPTM_reg, VOB_V_EPTM_reg, VOB_A_STP_PTM1_reg, VOB_A_STP_PTM2_reg, VOB_A_GAP_LEN1_reg, and VOB_A_GAP_LEN2_reg.

At step #31064 the data from the first cell VOBU start address C_FVOBU_SA_reg to the ILVU end pack address ILVU_EA_reg, i.e., the data for one interleaved unit ILVU, is transferred to the stream buffer 2400. The procedure then moves to step #31066.

More specifically, the address data ILVU_EA_reg stored in the decoding system controller 2300 (Fig. 26) is supplied to the reproduction controller 2002. The reproduction controller 2002 thus controls the recording media drive unit 2004 and signal processor 2008 to read the data to the ILVU_EA_reg address, and after error correction code ECC and other signal processing is accomplished by the signal processor 2008, the data for the first ILVU in the cell is output as the reproduced bitstream St61 to the stream buffer 2400. It is thus possible to output the data for one contiguous interleaved unit ILVU on the recording medium M to the stream buffer 2400.

At step #31066 it is determined whether all interleaved units in the interleaved block have been read and transferred. If the interleaved unit ILVU processed is the last ILVU in the interleaved block, "0x7FFFFFFF" indicating termination is set to the next-ILVU start address NT_ILVU_SA_reg as the next read address. If all interleaved units in the interleaved block have thus been processed, the procedure moves to step #31068.

At step #31068 the reading head 2006 is again jumped to the address NT_ILVU_SA_reg of the next interleave unit to be reproduced, and the procedure loops back to step #31062. Note that this jump is also accomplished as described above, and the loop from step #31062 to step #31068 is repeated.

However, if step #31066 returns YES, i.e., all interleaved unit ILVU in the interleaved block have been transferred, step #31052 terminates.

The non-multi-angle interleaved block process (step #31052) thus transfers the data of one cell to the stream buffer 2400.

The non-multi-angle contiguous block process is executed in step #31054, Fig. 63, is described further below with reference to Fig. 65.

At step #31070 the reading head 2006 is jumped to the first cell VOBU start address C_FVOBU_SA read from the C_FVOBU_SA_reg register. This jump is also accomplished as described above, and the loop from step #31072 to step #31076 is initiated.

At step #31072 the DSI packet data in the navigation pack NV (Fig. 20) is extracted in the stream buffer 2400, the decoding table is defined, and the procedure moves to step #31074. The registers set in the decoding table are the VOBU_EA_reg, VOB_V_SPTM_reg, VOB_V_EPTM_reg, VOB_A_STP_PTM1_reg, VOB_A_STP_PTM2_reg, VOB_A_GAP_LEN1_reg, and VOB_A_GAP_LEN2_reg.

At step #31074 the data from the first cell VOBU start address C_FVOBU_SA_reg to the end pack address VOBU_EA_reg, i.e., the data for one video object unit VOBU, is transferred to the stream buffer 2400. The procedure then moves to step #31076. The data for one video object unit VOBU contiguously arrayed to the recording medium M can thus be transferred to the stream buffer 2400.

At step #31076 it is determined whether all cell data has been transferred. If all VOBU in the cell has not been transferred, the data for the

next VOBU is read continuously, and the process loops back to step #31070.

However, if all VOBU data in the cell has been transferred as determined by the C_LVOBU_SA_reg value in step #31076, the non-multi-angle contiguous block process (step #31054) terminates. This process thus transfers the data of one cell to the stream buffer 2400.

Decoding flows in the stream buffer

The process for decoding data in the stream buffer 2400 shown as step #31034 in Fig. 61 is described below referring to Fig. 66. This process (step #31034) comprises steps #31110, #31112, #31114, and #31116.

At step #31110 data is transferred in pack units from the stream buffer 2400 to the system decoder 2500 (Fig. 26). The procedure then moves to step #31112.

At step #31112 the pack data is from the stream buffer 2400 to each of the buffers, i.e., the video buffer 2600, sub-picture buffer 2700, and audio buffer 2800.

At step #31112 the Ids of the user-selected audio and sub-picture data, i.e., the audio ID AUDIO_ID_reg and the sub-picture ID

SP_ID_reg stored to the scenario information register shown in Fig. 58, are compared with the stream ID and sub-stream ID read from the packet header (Fig. 19), and the matching packets are output to the respective buffers. The procedure then moves to step #31114.

The decode timing of the respective decoders (video, sub-picture, and audio decoders) is controlled in step #31114, i.e., the decoding operations of the decoders are synchronized, and the procedure moves to step #31116.

Note that the decoder synchronization process of step #31114 is described below with reference to Fig. 15.

The respective elementary strings are then decoded at step #31116. The video decoder 3801 thus reads and decodes the data from the video buffer, the sub-picture decoder 3100 reads and decodes the data from the sub-picture buffer, and the audio decoder 3200 reads and decodes the data from the audio buffer.

This stream buffer data decoding process then terminates when these decoding processes are completed.

The decoder synchronization process of step #31114, Fig. 66, is described below with reference to Fig. 15. This processes comprises steps #31120, #31122, and #31124.

At step #31120 it is determined whether a seamless connection is specified between the current cell and the preceding cell. If a seamless connection, the procedure moves to step #31122, if not, the procedure moves to step #31124.

A process synchronizing operation for producing seamless connections is executed in step #31122, and a process synchronizing operation for non-seamless connections is executed in step #31124.

System encoder

In the embodiment described below, plural buffers, including a stream buffer 2400, video buffer 2600, audio buffer 2800, and reordering buffer 3300 as shown in Fig. 26, are used for the single time-share controlled buffer of the DVD decoder DCD in the present invention.

Note that in the following description the actual buffer means made from semiconductor memory devices or similar physical means are referred to as "physical buffers," and the buffer means to

which different data are stored by time-share controlled use of the physical buffers are referred to as "functional buffers." Note that sub-picture data decoding is completed instantaneously, and the load imposed on DVD decoder DCD operation can thus be ignored in comparison with load imposed by the encoded audio and video streams. The description of the present embodiment below is therefore limited to a single encoded video stream and a single encoded audio stream for simplicity.

Shown in Fig. 39 are the simulated results of data input/output to the video buffer 2600 and audio buffer 2800 of the DVD decoder DCD, and the sequence in which the DVD encoder ECD multiplexes the encoded video stream St27 and the encoded audio stream St31 to generate the corresponding bitstream. Note that the progression of time is shown on the horizontal axis T.

The frame G1 shown at the top row in Fig. 39 shows the packetizing of the encoded video stream St27 by the DVD encoder ECD. Each block V in frame G1 indicates a video packet V. The vertical axis indicates the input transfer rate to the video buffer 2600, and the horizontal axis, time-base T, indicates the transfer time. The area of each video

packet represents the data size of the packet. The audio packets A are similarly shown with the area of the audio packet also indicating the packet size. Note, however, that while the audio packets appear to be larger than the video packets V, i.e., contain more data, the audio packets and video packets are all the same size.

Data input/output to the video buffer 2600 of the DVD decoder DCD is shown on the second row of Fig. 39. The vertical axis Vdv here indicates the accumulated video data volume Vdv in the video buffer 2600.

More specifically, the first video packet V in the encoded video stream St71 input to the video buffer 2600 is input at time Tb1. The last video packet V in the encoded video stream St71 is input at time Tvf. Line SVi thus indicates the change in the video data volume Vdv accumulated in the video buffer 2600 at the front of the encoded video stream St71, and line SVf indicates the change in the video data volume Vdv accumulated in the video buffer 2600 at the end of the encoded video stream St71. Thus, the slopes of lines SVi and SVf indicate the input rate to the video buffer 2600. Line BCv indicates the maximum accumulation

capacity (storage capacity) of the video buffer 2600.

Note that lines BCv and BCa are determined based on data written to the system stream header according to the MPEG standard.

The accumulated video data volume Vdv in the video buffer 2600 increases linearly, and at time Td1 the first block d1 of video data is batch transferred in a first-in first-out (FIFO) fashion to the video decoder 3801 whereby it is consumed for decoding. As a result, the accumulated video data volume Vdv is reduced to (BCv - d1), and then continues to accumulate. Note that while this example shows the accumulated video data volume Vdv at time Td1 to have reached the maximum storage capacity BCv of the video buffer 2600, it is not necessary for the accumulated video data volume Vdv to have reached the maximum storage capacity BCv when decoding begins, and may obviously be less than the maximum storage capacity BCv.

Part of the data d1 transferred to the video buffer 2600, specifically the data at point B at the top end of the dotted line having the same slope as line SVi and intersecting the time-base at intersection tb, was data input at time Tb. Thus,

the data block d1 first decoded is the data input between time Tb1 to Tb2. Furthermore, when data input time Tb2 is later than decoding time Td1, a data underflow state occurs in the video buffer 2600 at time Td1.

The variation in the per-picture encoded data quantity is great in an MPEG-compressed encoded video stream, and temporary depletion of large amounts of encoded data may occur. To prevent a data underflow state from occurring in the video buffer in such cases, it is necessary to write as much data as possible to the video buffer 2600. The time required for data transfer is thus called the video buffer verifier delay vbv_delay.

The third row in Fig. 39 shows the audio data packetizing process. As with the video data packets in the first row, the frames A indicate the audio packets A, the size of which is equal to the size of the video packets V.

The fourth row simulates the results of data input/output to the audio buffer 2800 similar to the results of data input/output to the video buffer 2600 in the second row. The vertical axis here indicates the accumulated audio data volume Vda in the audio stream buffer 2800.

Note that in Fig. 39 time Tvp1 is the video presentation start time, Tap1 is the audio presentation start time, Fv is the video frame reproduction time, and Fa is the audio frame reproduction time Fa.

At time Tad1, the first audio packet A in the encoded audio stream St75 is input to the audio buffer 2800. Line SAi thus indicates the change in the audio data volume Vda accumulated in the audio buffer 2800 at the front of the encoded audio stream St75, and line SAf indicates the change in the audio data volume Vda accumulated in the audio buffer 2800 at the end of the encoded audio stream St75. Thus, the slopes of lines SAi and SAf indicate the input rate to the audio buffer 2800. Line BCa indicates the maximum accumulation capacity (storage capacity) of the 2800. Note that the maximum storage capacity BCa is obtained in the same manner as the maximum storage capacity BCv of the video buffer 2600.

The audio access unit, i.e., the audio frame (which is also the audio compression unit), is generally constant in the audio stream. A data overflow state occurs in the audio buffer 2800 if the encoded audio stream St75 is input to the audio

buffer 2800 in a short period at a rate exceeding the consumption rate, and the input volume thus exceeds the maximum storage capacity BCa of the audio buffer 2800. When this happens, the next audio packet A cannot be input until audio data stored in the audio buffer 2800 is consumed, i.e., decoded.

Furthermore, because the video packets V and audio packets A are contiguous in a single bitstream, the following video packet V cannot be input to the video buffer 2600 even through the video buffer 2600 itself is not in a data overflow state if a data overflow state occurs in the audio buffer 2800. Thus, a data overflow state in the audio buffer 2800 may create a data underflow state in the video buffer 2600 depending on the duration of the data overflow state.

Therefore, to prevent an audio buffer overflow, data input to the audio buffer 2800 is restricted when the sum of the data accumulated in the audio buffer and the data size of the packet exceeds the maximum audio buffer capacity. More specifically, the present embodiment transfers only the packet(s) containing the (frame) data required by the audio decode time, and does not permit

inputting more than the required amount of data to the audio buffer. However, because of the difference in the data size of the packets (approx. 2 KB) and the audio frame (1536 bytes at 384 Kbps with Dolby AC-3 coding), the data for the frame following the current frame is simultaneously input.

Thus, as shown by the audio data packet stream (row three, Fig. 39 and the audio buffer input/output timing (row four, Fig. 39), only approximately one audio frame of data is input to the audio buffer 2800 before the audio decode time.

Because of the characteristics of an MPEG-compressed video stream, decoding normally starts at video frame reproduction time Fv before the first video presentation start time Tvp1, and the audio data is input to the audio buffer 2800 at audio frame reproduction time Fa before decoding starts, i.e., before audio presentation start time Tap1. The video stream is thus input to the video buffer 2600 approximately (video buffer verifier delay vbv_delay + video frame reproduction time Fv - audio frame reproduction time Fa) before audio stream input begins.

The fifth row in Fig. 39 shows the results of interleaving the video packet stream G1 (row 1) with the audio packet stream G2 (row 3). The audio and video packets are interleaved by multiplexing referenced to the respective input times to the audio and video buffers.

For example, Tb1 is the index for the buffer input time of the first pack in the encoded video stream, and Ta1 is the index for the buffer input time of the first pack in the encoded audio stream. The packed data is then multiplexed referenced to the buffer input time of the data in the packs to the audio and video buffers. Because the encoded video stream is thus input to the video buffer 2600 at approximately the vbv_delay plus one video frame minus one audio frame, plural video frames are contiguous at the front of the system stream. There is a similar series of audio packets at the end of the system stream equivalent to approximately the lead time at which the video stream is buffered before the encoded audio stream.

Note again that a data overflow state occurs in the audio buffer 2800 if the encoded audio stream St75 is input to the audio buffer 2800

in a short period at a rate exceeding the consumption rate, and the input volume thus exceeds the maximum storage capacity BCa of the audio buffer 2800. When this happens, the next audio packet A cannot be input until audio data stored in the audio buffer 2800 is consumed, i.e., decoded. Gaps therefore occur at the end of the system stream when only the audio packets are being transferred.

For example, if the video bit rate is 8 Mbps, the video buffer capacity is 224 KB, and 224 KB of video data are buffered before video decoding starts in the DVD system, the video buffer verifier delay vbv_delay will be approximately 219 msec. If NTSC video and AC-3 audio coding are used, one NTSC video frame is approximately 33 msec, and one AC-3 audio frame is approximately 32 msec. At the head of the system stream in this example the video stream leads the audio stream by approximately 220 msec (= 219 msec + 33 msec - 32 msec), and video packets are arrayed contiguously for this period.

The audio packets continue in a similar series at the end of the system stream for the lead time of the encoded video stream to the encoded audio stream.

By thus producing and recording the system streams, audio and video reproduction can be accomplished without creating a data underflow state in the video buffer of the DVD decoder shown in Fig. 26.

Movies and other titles can be recorded to an optical disk by a DVD system using this type of MPEG system stream. However, if plural titles implementing parental lock control, director's cut selections, and other features are recorded to a single optical disk, it may be necessary to record ten or more titles to the disk. This may require the bit rate to be dropped with the incumbent loss of image quality.

However, by sharing the system streams common to plural titles, e.g., titles implementing parental lock control, director's cut selections, and other features, and discretely recording for each of the plural titles only those scenes that are unique to those titles, it is possible to record plural different titles to a single optical disk without reducing the bit rate, and thereby without loss of image quality. This method thus makes it possible, for example, to record plural titles for different countries, cultures, or

language groups to a single optical disk without reducing the bit rate and therefore without loss of image quality.

An example of a title stream providing for parental lock control is shown in Fig. 40. When so-called "adult scenes" containing sex, violence, or other scenes deemed unsuitable for children are contained in a title implementing parental lock control, the title stream is recorded with a combination of common system streams SSa, SSb, and SSe, an adult-oriented system stream SSc containing the adult scenes, and a child-oriented system stream SSd containing only the scenes suitable for children. Title streams such as this are recorded as a multi-scene system stream containing the adult-oriented system stream SSc and the child-oriented system stream SSd arrayed to the multi-scene period between common system streams SSb and SSe.

The relationship between each of the component titles and the system stream recorded to the program chain PGC of a title stream thus comprised is described below.

The adult-oriented title program chain PGC1 comprises in sequence the common system

streams SSa and SSb, the adult-oriented system stream SSc, and the common system stream SSe. The child-oriented title program chain PGC2 comprises in sequence the common system streams SSa and SSb, the child-oriented system stream SSd, and the common system stream SSe.

To share system streams within titles comprising multi-scene periods such as this, and to divide the system stream as needed for authoring, it is also necessary to be able to connect and contiguously reproduce these system streams. When system streams are connected and contiguously reproduced, however, pauses in the video presentation (freezes) occur at the system stream connections, and seamless reproduction presenting a natural flow of a single title can be difficult to achieve.

Data input/output to the video buffer 2600 of the DVD decoder DCD shown in Fig. 26 during contiguous reproduction is shown in Fig. 41. In Fig. 41, block Ga shows the data input/output to the video buffer 2600 when encoded video stream Sva and encoded video stream Svb are input to the DVD decoder DCD. Block Gb shows the video packet streams of encoded video stream Sva and encoded

video stream Svb. Block Gc shows the interleaved system streams Sra and Srb. Note that blocks Ga, Gb, and Gc are arranged referenced to the same time-base T as that shown in Fig. 39.

In block Ga the vertical axis shows the accumulated video data volume Vdv in the video buffer, and slope Sva indicates the input rate to the video buffer 2600. Where the video data volume Vdv accumulated in the video buffer 2600 is shown to decrease in block Ga therefore indicates data consumption, i.e., that data has been output for decoding.

Time T1 also indicates the input end time of the last video packet V1 in the system stream Sra (block Gc), time T3 indicates the input end time of the last audio packet A1 in system stream Srb, and time Td indicates the first decode time of encoded video stream Svb (block Ga).

Of the two system streams, the encoded video stream Sva and the encoded audio stream Saa, constituting system stream Sra, the encoded video stream Sva is input to the video buffer 2600 before the encoded audio stream Saa is input to the audio buffer 2800 as described above. A series of audio

packets A therefore remains at the end of the system stream Sra.

A data overflow state also occurs in the audio buffer 2800 if audio packets A exceeding the capacity of the audio buffer 2800 are input thereto. When this occurs, the next audio packet cannot be buffered until an equivalent amount of audio data is consumed, i.e., decoded.

The first video packet V2 in system stream Srb therefore cannot be input to the video buffer 2600 until input of the last audio packet A1 in the system stream Sra is completed. As a result, video stream input to the video buffer 2600 cannot be continued due to the interference from audio packet A1 during the period from T1, the input end time of the last video packet V1 in system stream Sra, to T3, the input end time of the last audio packet A1 in system stream Sra.

In the following example it is assumed that the video bit rate of the DVD system is 8 Mbps, the video buffer capacity is 224 KB, the audio buffer capacity is 4 KB, the audio data is encoded with Dolby AC-3 compression, and the compression bit rate is 384 Kbps. In AC-3 audio compression, the reproduction time of one audio

frame is 32 msec, corresponding to a data size of 1536 bytes/frame, and two audio frames can therefore be stored in the audio buffer.

Because the number of audio frames that can be stored in the audio buffer is two, the earliest possible time T3, which is the input end time of the last audio packet A1 in system stream Sra, is at the (reproduction start time of the last audio frame in system stream Sra) - (reproduction time of two audio frames). The reproduction start time of the last audio frame in system stream Sra is also approximately one audio frame earlier than the presentation start time of the first frame in the encoded video stream Svb of system stream Srb. The presentation start time of encoded video stream Svb is at the video buffer verifier delay vbv_delay plus one video frame after the input end time T1 of the last video packet V1 in system stream Sra.

Therefore, if 224 KB of video data is buffered by the time video decoding starts, the video buffer verifier delay vbv_delay is approximately 219 msec. If NTSC video and AC-3 audio coding are used, one NTSC video frame is approximately 33 msec, and one AC-3 audio frame is approximately 32 msec. Thus, there is approximately

156 msec (= 219 msec + 33 msec - 32 msec - 2 x 32 msec) from the input end time T1 of the last video packet V1 in system stream Sra to the input end time T3 of the last audio packet A1 in system stream Sra. The encoded video stream Svb cannot be input to the video buffer 2600 during this approximately 156 msec period.

Therefore, because all decode data d1 is not input to the video buffer 2600 at time Td, a data underflow state occurs in the video buffer 2600. In such cases the video presentation is intermitted, video freezing occurs, and correct video presentation is interrupted.

Thus, when plural system streams are connected and contiguously decoded to reproduce a single contiguous sequence of scenes from plural titles comprising a commonly shared system stream and plural system streams containing content discretely encoded for specific titles, video freezing apparent as pauses in the video presentation at system stream connections can occur, and it is not always possible to seamlessly reproduce such plural system streams as a single contiguous title.

When plural different system streams SSc and SSd are connected to one common system stream SSe as shown in Fig. 40, a time difference occurs between the video reproduction time and the audio reproduction time because of the offset between the audio and video frame reproduction times, and this time difference varies according to the reproduction path. As a result, buffer control fails at the connection, video reproduction freezes or the audio reproduction is muted, and seamless reproduction is not possible.

This problem is considered below with reference to Fig. 42 as it applies to the parental lock control example shown in Fig. 40. In Fig. 42 SScv and SSca represent the reproduction times of the video and audio frame unit streams in adult-oriented system stream SSc. SSdv and SSda similarly represent the reproduction times of the video and audio frame unit streams in the child-oriented system stream SSd.

As described above, if NTSC video and AC-3 audio coding are used, one NTSC video frame is approximately 33 msec, and one AC-3 audio frame is approximately 32 msec., and the audio and video reproduction times therefore do not match. As a

result, a difference occurs in the video reproduction time, which is an integer multiple of the video frame reproduction time, and the audio reproduction time, which is an integer multiple of the audio frame reproduction time. This reproduction time different is expressed as Tc in the adult-oriented system stream SSc, and time Td in the child-oriented system stream SSd. This difference also varies according to the change in the reproduction time of the reproduction paths, and Tc _ Td.

Therefore, when plural system streams are connected with a single system stream as described above with parental lock control and director's cut titles, there is a maximum reproduction gap of one frame in the audio and video reproduction times at the points where the system streams branch and connect.

This reproduction gap is described next with reference to Fig. 43. The top program chain PGC1 represents the reproduction path of the adult-oriented system stream. SScv and SSev represent the reproduction times of the video frame unit streams in adult-oriented system stream SSc and common system stream SSe, and SSca and SSea

represent the reproduction times of the audio frame unit streams in adult-oriented system stream SSc and common system stream SSe.

These frame unit reproduction times are expressed in the figure by the line segments ended with arrows on both ends.

The video stream SScv of the adult-oriented system stream SSc in this example ends after 3 frames, and is followed by the common system stream SSe starting at frame 4 with the first frame of the video stream SSev. The audio stream SSca likewise ends at frame 4, and the first frame of the common audio stream SSea starts from frame 5. The resulting difference in the frame reproduction times between the audio and video streams produces a reproduction gap of time Tc equivalent to a maximum one frame between the video stream and the audio stream when these two streams SSc and SSe are connected.

The bottom program chain PGC2 similarly represents the reproduction path of the child-oriented system stream. SSdv and SSev represent the reproduction times of the video frame unit streams in child-oriented system stream SSd and common system stream SSe, and SSda and SSea

represent the reproduction times of the audio frame unit streams in child-oriented system stream SSd and common system stream SSe.

As with the adult-oriented program chain PGC1 above, a reproduction gap of time Td equivalent to a maximum one frame between the video stream and the audio stream occurs when these two streams SSd and SSe are connected. When the reproduction paths to the common system streams differ before the connection point as shown in Fig. 43, it is possible to adjust the reproduction start times of the connected common audio and video streams to the reproduction start time difference of at least one reproduction path. As shown in this figure, the audio and video end times of the adult-oriented system stream SSc are the same as the audio and video start times of the common system stream SSe, i.e., a gap-less connection is achieved. Note that in this example the gap Td of the child-oriented system stream SSd is less than the gap Tc of the adult-oriented system stream SSc ($Td < Tc$).

The one program chain PGC1, i.e., adult-oriented system stream SSc and common system stream SSe, is thus reproduced without a reproduction gap,

but program chain PGC2, i.e., child-oriented system stream SSd and common system stream SSe, is reproduced with an audio reproduction gap of Tc - Td. Thus, even when connecting from plural reproduction paths (SSc and SSd) to one system stream (SSe), it is possible to eliminate any reproduction gap in the video or audio on at least one reproduction path.

The third row in Fig. 43 shows the change in audio buffer storage during continuous reproduction of program chain PGC2, i.e., child-oriented system stream SSd and common system stream SSe. The per-frame reproduction time of the audio stream in the audio buffer is indicated by the arrows. Note that system streams SSd and SSe are connected with an audio reproduction gap of Tc - Td, the difference between the reproduction time difference Tc of PGC1 and the reproduction time difference td of PGC2, at the connection.

However, because DVD players normally synchronize audio and video output referenced to the audio signal, the audio frames are output continuously. As a result, the audio reproduction gap Tc - Td is not reproduced as a gap during

reproduction, and audio reproduction is therefore contiguous.

The common system stream SSe is encoded so that the audio is reproduced, i.e., decoded, at a delay of Tc to the video. As a result, when the audio is reproduced, i.e., decoded, so that there is no audio reproduction gap Tc - Td, audio decoding is accomplished before the audio data is completely input to the audio buffer, and a data underflow state results in the audio buffer as shown by line Lu in Fig. 43.

When the audio reproduction is contiguous and a reproduction gap is inserted between video frames, a data underflow state occurs in the video buffer due to video stream reproduction, similarly to video stream reproduction being interrupted as shown in Fig. 41.

As thus described, when plural difference system streams and one common system stream are connected, a difference occurs between the video reproduction time and the audio reproduction time of the respective paths due to the offset in the audio and video frame reproduction times.

The present invention therefore provides a recording method and apparatus and a reproduction

method and apparatus whereby a video or audio buffer underflow state is prevented at the system stream connections, and seamless reproduction in which pauses in the video reproduction (freezes) or pauses in the audio reproduction (muting) do not occur.

A method of connecting a single common system stream to the plural system streams contained in the multi-scene period of a title stream as shown in Fig. 40 is described below according to the present invention. The physical structure of the optical disk M, the overall data structure of the optical disk M, and the structures of the DVD encoder ECD and DVD decoder DCD in this embodiment are as previously described with reference to Figs. 4 - 14, Figs. 1,16 - 20,22, Figs. 25,27 - 29, and Figs. 26 above, and further description thereof is thus omitted below.

There are two data transfer models under the MPEG standard: constant bit rate (CBR) whereby data is transferred continuously without interruptions, and variable bit rate (VBR) whereby data is transferred intermittently with interruptions in the transfer. For simplicity, the

present embodiment is described below using the CBR model only.

Referring first to Figs. 44, 45, and 46, a simple one-to-one system stream connection between first and second common system streams SSa and SSb is described first. For simplicity the following description is restricted to operation using one video stream SSav and one audio stream SSba.

The system streams produced according to the present invention are shown in Fig. 44, the operation whereby these system streams are connected is shown in Fig. 45, and the method of generating the system streams is shown in Fig. 46.

The structure of the tail of the leading common system stream SSa, and the head of the following common system stream SSb, recorded to the optical disk M are shown in Fig. 44.

In Fig. 44 are shown the structure of the end of the preceding common system stream SSa and the common system stream SSb following thereafter. Note that both system streams SSa and SSb are recorded to the optical disk M.

The fifth row block Ge shows the structure of both system streams SSa and SSb. The first common system stream SSa comprises video stream

SSav and audio stream SSaa; the second common system stream SSb similarly comprises video stream SSbv and audio stream SSba.

The fourth row Gd shows the audio packet streams A of audio stream SSaa and audio stream SSba extracted from system stream SSa and system stream SSb.

The third row Gc shows the data input/output state of the audio buffer 2800 when audio stream SSaa and audio stream SSba are input to the DVD decoder DCD shown in Fig. 26.

The second row Gb shows the video packet streams V of video stream SSav and video stream SSbv extracted from system stream SSa and system stream SSb.

The first row Ga shows the data input/output state of the video buffer 2600 when video stream SSav and video stream SSbv are input to the DVD decoder DCD shown in Fig. 26.

Note that Ga, Gb, Gc, Gd, and Ge are all referenced to the same time-base (direction T).

Tvae in Fig. 44 is the input end time of the video stream SSav to the video buffer 2600, and Taae is the input end time of the audio stream SSaa to the audio buffer 2800.

When system stream SSa is input to the DVD decoder DCD, the difference between the input end times Tvae and Taae of the video stream SSav and audio stream SSaa to the respective buffers 2600 and 2800 is small, and is less than the reproduction time of two audio frames. As a result, the last audio packet A can be accumulated in the audio buffer 2800 before input of the audio and video streams in the next system stream starts.

Likewise, when system stream SSb is input to the DVD decoder DCD, the difference between the input start times of the video stream SSbv and audio stream SSba to the respective buffers 2600 and 2800 is small, and is less than the reproduction time of two audio frames.

The data input/output state to the video buffer 2600 when system streams SSa and SSb (Fig. 44) stored to the optical disk M are connected and contiguously reproduced.

The top row in Fig. 45 shows the data input/output state of the video buffer 2600 when video stream SSav and video stream SSbv are input continuously to the DVD decoder DCD.

As in Fig. 39, Fig. 41, and Fig. 44, the vertical axis Vdv indicates the accumulated video

data volume Vdv in the video buffer 2600, and the horizontal axis indicates time T. Lines SSav and SSbv indicate the change in the video data volume Vdv accumulated in the video buffer 2600, and the slopes of the lines indicate the input rate to the video buffer 2600. Where the accumulated video data volume Vdv in the video buffer 2600 drops indicates data consumption, i.e., that decoding has occurred.

The second row shows the video packet streams in the video streams SSav and SSbv shown in Fig. 26.

The third row shows the system streams SSa and SSb according to the present embodiment. Time T1 is the input end time of the last video packet V1 in system stream SSa, time T2 indicates the input start time of the first v] V2 in system stream SSb, and time T indicates the decoding start time of system stream SSb.

The difference between the input end times to the video buffer 2600 and audio buffer 2800 of the video stream SSav and the audio stream SSaa forming the system stream SSa of the present embodiment is reduced by the system stream production method shown in Fig. 46. As a result,

interference with the input of system stream SSb resulting from a succession of remaining audio packets A at the end of system stream SSa does not occur. The difference between the input end time T1 of the first video packet V1 of system stream SSa and the input start time T2 of the first video packet V2 in system stream SSb is small, there is sufficient time from the input start time T2 of video packet V2 to the first decode time Td of the video stream SSbv, and the video buffer 2600 therefore does not underflow at time Td.

Unlike with the system stream shown in Fig. 41, the audio buffer 2800 therefore does not overflow at the end of the system stream, i.e., there is no interference with inputting the encoded video stream of the next system stream, when connecting and contiguously reproducing system streams SSa and SSb according to the present embodiment, and seamless reproduction can be achieved.

A first method of producing a first common system stream SSa and a second common system stream SSb connected thereafter is described below with reference to Fig. 46. Note that as in Fig. 44, the structure of the tail of the leading common system

stream SSa, and the head of the following common system stream SSb, recorded to the optical disk M are shown in Fig. 46.

The first row in Fig. 46 corresponds to block Ga in Fig. 44, and simulates the data input/output of video stream SSav and video stream SSbv to the video buffer 2600. Time T1 is the input end time of all data in the video stream SSav.

The second row similarly corresponds to block Gb in Fig. 44, and shows the video data packet stream.

The third row similarly corresponds to block Gc in Fig. 44, and simulates the data input/output of audio stream SSaa and audio stream SSba to the audio buffer 2800.

The fourth row similarly corresponds to block Gd in Fig. 44, and shows the audio data packet stream.

The fifth row similarly corresponds to block Ge in Fig. 44, and shows the system stream resulting from interleaving and packing the video packets V shown in the second row and the audio packets A shown in the fourth row. The video packets and audio packets are interleaved in a FIFO manner from the video and audio buffers referenced

to the packet input time to the respective buffer. In other words, the packed data is multiplexed referenced to the time the data contained in the pack is input to the video or audio buffer.

The method of generating the first common system stream and the following second common system stream is described next.

It is assumed below that the video bit rate is 8 Mbps, the video buffer capacity is 224 KB, the audio buffer capacity is 4 KB, the audio data is encoded with Dolby AC-3 compression, and the compression bit rate is 384 Kbps. In AC-3 audio compression, the reproduction time of one audio frame is 32 msec, corresponding to a data size of 1536 bytes/frame, and two audio frames can therefore be stored in the audio buffer.

Referenced to the input end time T1 of the video stream SSav to the video buffer 2600, the audio frame data following the current audio frame is moved to the audio stream SSba at time T1 to accumulate one audio frame in the audio buffer. This operation is described in detail below referring to the simulation results shown in row 3 of Fig. 46.

Specifically, two audio frames (= 1536 bytes) from the encoded audio stream SSaa are accumulated in the audio buffer (4 KB capacity) at time T1, and the third to sixth audio frames following thereafter, indicated by frame Ma in Fig. 46, are moved to the beginning of the following encoded audio stream SSba. Note that the encoded audio stream is moved in audio frame units because the audio frame is the unit of reproduction.

Following the above operation, the encoded video stream SSav is packetized as shown in row 2 in Fig. 46, and the encoded audio stream SSaa is packetized as shown in row 4. As shown in row 5, the video packets V and audio packets A are then interleaved (multiplexed) to maintain an average distribution of audio packets to video packets in the FIFO sequence described above referenced to the packet input times to the buffers 2600 and 2800. After packing and converting to a system stream, the data is then recorded to the optical disk.

In the same manner the encoded video stream SSbv is packetized as shown in row 2 in Fig. 46, and the encoded audio stream SSba is packetized as shown in row 4. As shown in row 5, the video packets V and audio packets A are then interleaved

(multiplexed) to maintain an average distribution of audio packets to video packets in the FIFO sequence described above referenced to the packet input times to the buffers 2600 and 2800. After packing and converting to a system stream, the data is then recorded to the optical disk.

The resulting system streams SSa and SSb are thus structured as shown in Fig. 44, enabling seamless reproduction by the DVD decoder DCD shown in Fig. 26.

Because two audio frames can be accumulated in the audio buffer, the last audio frame in the system stream SSa stored in the audio buffer at time T1 is transferred as the last audio packet in system stream SSa during the two-frame reproduction time before decoding said last audio frame begins. The maximum input end time difference between the video packets and audio packets at the end of the system stream SSa is therefore the reproduction time of two audio frames.

Furthermore, the audio buffer will not underflow if the next audio data is input to the audio buffer before the presentation end time of the audio frames accumulated in the audio buffer as of time T2. The input time of the first audio

packet in system stream SSb is therefore at latest within the reproduction time of two audio frames after time T2 (= the presentation time of the accumulated audio frames + the reproduction time of one audio frame). Therefore the maximum input start time difference between the video packets and audio packets at the beginning of system stream SSb is the reproduction time of two audio frames.

A second method of producing the system stream recorded to an optical disk according to the present embodiment is described next below with reference to Fig. 47. The first, second, third, fourth, and fifth rows in Fig. 47 simulate the video and audio data input/output states to the respective buffers referenced to the same time-base T as shown in Fig. 44.

The first row in Fig. 47 corresponds to block Ga in Fig. 44, and simulates the data input/output of video stream SSav and video stream SSbv to the video buffer 2600.

The second row similarly corresponds to block Gb in Fig. 44, and shows the video data packet stream.

The third row similarly corresponds to block Gc in Fig. 44, and simulates the data

input/output of audio stream SSaa and audio stream SSba to the audio buffer 2800.

The fourth row similarly corresponds to block Gd in Fig. 44, and shows the audio data packet stream.

The fifth row similarly corresponds to block Ge in Fig. 44, and shows the system stream resulting from interleaving and packing the video packets V shown in the second row and the audio packets A shown in the fourth row. The video packets and audio packets are interleaved in a FIFO manner from the video and audio buffers referenced to the packet input time to the respective buffer. In other words, the packed data is multiplexed referenced to the time the data contained in the pack is input to the video or audio buffer. The first common system stream SSa and the second common system stream SSb following thereafter can be produced using the first method described above with reference to Fig. 46.

A different method for generating the first common system stream SSa and the second common system stream SSb following thereafter, i.e., a method different from that described with

reference to Fig. 46, is described below with reference to Fig. 47.

In the first method described above, part of the encoded audio stream from the preceding system stream is moved to the following system stream. This second method, however, is characterized by moving part of the encoded video and audio streams from the following system stream. This second method is particularly effective when the preceding scene (system stream) is a scene from a multi-scene period, i.e., when moving from one of plural scenes (system streams) to the encoded system stream of a single scene is extremely difficult.

With this method the first GOP in video stream SSbv is moved to video stream SSav. The one GOP moved from video stream SSbv is connected to video stream SSav to assure time-base contiguity at the end of video stream SSav. At the second GOP from the beginning of video stream SSbv, i.e., the second GOP counted from the beginning of video stream SSbv including the first GOP already moved, referenced to the input start time T2 of the data decoded first, the audio data of one audio frame is

moved to the audio stream SSaa to accumulate one audio frame in the audio buffer.

The one audio frame of data moved from audio stream SSba is then connected to audio stream SSaa to assure time-base contiguity at the end of audio stream SSaa.

The video data is moved in GOP units because the GOP is the unit of video data reproduction. Audio data is likewise moved in audio frame units because the audio frame is the unit of audio frame reproduction.

Following the above operation, the encoded video stream SSav is packetized as shown in row 2 in Fig. 47, and the encoded audio stream SSaa is packetized as shown in row 4. As shown in row 5, the video packets V and audio packets A are then interleaved (multiplexed) to maintain an average distribution of audio packets to video packets in the FIFO sequence described above referenced to the packet input times to the buffers 2600 and 2800. After packing and converting to a system stream, the data is then recorded to the optical disk.

In the same manner the encoded video stream SSbv is packetized as shown in row 2 in Fig. 47, and the encoded audio stream SSba is packetized

as shown in row 4. As shown in row 5, the video packets V and audio packets A are then interleaved (multiplexed) to maintain an average distribution of audio packets to video packets in the FIFO sequence described above referenced to the packet input times to the buffers 2600 and 2800. After packing and converting to a system stream, the data is then recorded to the optical disk.

The resulting system streams SSa and SSb are thus structured as shown in Fig. 39, enabling seamless reproduction by the DVD decoder DCD shown in Fig. 26.

Because two audio frames can be accumulated in the audio buffer, the last audio frame in the system stream SSa stored in the audio buffer at time T1 is transferred as the last audio packet in system stream SSa during the two-frame reproduction time before decoding said last audio frame begins. The maximum input end time difference between the video packets and audio packets at the end of the system stream SSa is therefore the reproduction time of two audio frames.

Furthermore, the audio buffer will not underflow if the next audio data is input to the audio buffer before the presentation end time of

the audio frames accumulated in the audio buffer as of time T2. The input time of the first audio packet in system stream SSb is therefore at latest within the reproduction time of two audio frames after time T2 (= the presentation time of the accumulated audio frames + the reproduction time of one audio frame). Therefore the maximum input start time difference between the video packets and audio packets at the beginning of system stream SSb is the reproduction time of two audio frames.

The next embodiment relates to connecting the system stream branches obtained by means of the system encoder according to the preferred embodiment of the present invention.

The physical structure of the optical disk, the overall data structure of the optical disk, and the DVD decoder DCD in the present embodiment are as described above, and further description thereof is thus omitted below.

Note that the description of the present embodiment below is limited to a single encoded video stream and a single encoded audio stream for simplicity.

Fig. 48 shows the structure of the end of the second common system stream SSb, and the

beginnings of the two parental lock control system streams SSc and SSd that can be connected to the end of common system stream SSb. Note that the common system stream SSb and one of the two parental lock control system streams SSc and SSd are arrayed to the same time-base (horizontal time axis T) as shown in Fig. 46.

System streams SSb, SSc, and SSd shown as separate blocks in Fig. 48 represent the following content as in Fig. 46.

The fifth row in each block shows the structure of system streams SSb, SSc, and SSd. System stream SSb comprises video stream SSbv and audio stream SSba; system stream SSc similarly comprises video stream SScv and audio stream SSca; and system stream SSd similarly comprises video stream SSdv and audio stream SSda.

The fourth rows show the audio packet streams A of audio stream SSba, audio stream SSca and audio stream SSda extracted from system streams SSb, SSc, and SSd.

The third rows show the data input/output state of the audio buffer 2800 when audio stream SSba, audio stream SSca and audio stream SSda are input to a DVD decoder DCD shown in Fig. 26.

The second rows show the video packet streams V of video stream SSbv, video stream SScv, and video stream SSdv extracted from system streams SSb, SSc, and SSd.

The first rows show the data input/output state of the video buffer 2600 when video stream SSbv, video stream SScv, and video stream SSdv are input to a DVD decoder DCD.

The audio content of the first several audio frames in audio stream SSca and audio stream SSda at the beginning of system stream SSc and system stream SSd is the same.

When system stream SSb is input to the DVD decoder DCD, the difference between the input end times of the video stream SSbv and audio stream SSba to the respective buffers 2600 and 2800 is small, and at most is less than the reproduction time of two audio frames.

When system stream SSc is input to the DVD decoder DCD, the difference between the input end times of the video stream SScv and audio stream SSca to the respective buffers 2600 and 2800 is small, and at most is less than the reproduction time of two audio frames.

When system stream SSd is input to the DVD decoder DCD, the difference between the input end times of the video stream SSdv and audio stream SSda to the respective buffers 2600 and 2800 is small, and at most is less than the reproduction time of two audio frames.

The data input/output state of the video buffer 2600 when system stream SSb is connected to and contiguously reproduced with system stream SSc or system stream SSd is the same as shown in Fig. 44. Specifically, system stream SSa in Fig. 44 corresponds to system stream SSb in Fig. 48, and system stream SSb in Fig. 44 corresponds to either system stream SSc or system stream SSd in Fig. 48.

When system stream SSb and system stream SSd or system stream SSc in Fig. 48 are contiguously reproduced using the DVD decoder DCD shown in Fig. 26, the video buffer also does not overflow as described above with reference to Fig. 44. As a result, seamless reproduction can be achieved when system stream SSb is connected and contiguously reproduced with system stream SSc or system stream SSd.

Note also that system streams SSb, SSc, and SSd are produced using the method described with reference to Fig. 46.

The data structure of system streams SSb, SSc, and SSd produced according to the method shown in Fig. 46 is shown in Fig. 48, and seamless reproduction can therefore be achieved using the DVD decoder DCD shown in Fig. 26.

As described with reference to audio frame movement Fig. 46, the maximum input end time difference between the video packets and audio packets at the end of the system stream SSb is at most the reproduction time of two audio frames, and the maximum input start time difference between the video packets and audio packets at the beginning of system stream SSc or SSd is at most the reproduction time of two audio frames.

When the audio frame moved from audio stream SSba is connected to destination audio streams SSca and SSda, an audio reproduction stop, i.e., an audio reproduction gap, is provided when making the connection. As a result, the differences in the video reproduction time and the audio reproduction time of each reproduction path can be corrected based on the reproduction gap information

in the system streams not shared between different program chains PGC. As a result, this video and audio reproduction time difference can be prevented from affecting the process connecting preceding and following system streams.

Fig. 49 is used to describe the difference in the video reproduction time and audio reproduction time of different reproduction paths according to the present embodiment. In Fig. 49, time Tb represents the time difference between the audio and video reproduction end times at the end of the system stream common to the adult-oriented title and the child-oriented title before moving the audio data; time Tc is the time difference between the audio and video reproduction start times at the beginning of the adult-oriented title before audio data movement; and time Td is the time difference between the audio and video reproduction start times at the beginning of the child-oriented title before audio data movement.

It is possible to match the time difference between the audio and video reproduction start times of at least one of the plural different reproduction paths following the branch to the time difference of the audio and video reproduction end

times before the branch. Note that it is assumed in the following description that Tb = Tc, and Tb < Td.

Because Tb = Tc in the adult-oriented title after the branch, the audio frame moved from the common part of the adult-oriented and child-oriented title streams can be connected to the beginning of the adult-oriented title without an audio reproduction gap.

To enable seamless reproduction between system stream SSb and system stream SSc at the connection, the system streams are generated using the first system stream encoding method described above with reference to moving audio data from one system stream SSb to another system stream SSc.

The method of producing the system streams is the same as that described above with reference to Fig. 46 except that system streams SSa and SSb are replaced by system streams SSb and SSc in Fig. 49, and further description thereof is thus omitted below.

Because Tb < Td in the child-oriented title after the branch, the audio frame moved from the common part of the adult-oriented and child-oriented title streams can be connected to the

beginning of the child-oriented title stream with an audio reproduction gap of only Td - Tb.

To enable seamless reproduction between system stream SSb and system stream SSd at the connection, the system streams are generated using the first system stream encoding method described above with reference to moving audio data from one system stream SSb to another system stream SSd.

The method of producing the system streams is the same as that described above with reference to Fig. 46 except that system streams SSa and SSb are replaced by system streams SSb and SSd in Fig. 49, and further description thereof is thus omitted below.

Note that packetizing in this case is controlled so that the audio frames before and after the audio reproduction gap are not included in the same packet. As a result, it is possible to write the audio playback starting time information APTS (the audio frame reproduction start time including the audio reproduction pause time) of the audio frames before and after the audio reproduction gap into the system stream.

The packet containing the audio frame immediately preceding the audio reproduction gap is

of necessity small. During the packing process a padding packet is therefore used to produce a fixed-length pack of 2048 bytes/pack.

The audio reproduction gap information for the audio reproduction gap of system stream SSd in this embodiment is inserted to the system stream by writing the audio frame reproduction end time immediately before the audio reproduction gap of the child-oriented title to the audio reproduction stopping time 1 (VOB_A_STP_PTM1) in the navigation pack NV (Fig. 20), and writing the audio reproduction gap time Td - Tb to the audio reproduction stopping period 1 (VOB_A_GAP_LEN1) in the DSI packet.

When there is no audio reproduction gap, it is possible to determine that there is no audio reproduction gap by writing a 0 value to the audio reproduction stopping time 1 (VOB_A_STP_PTM1).

By means of the above process, it is possible to set the time difference between the different audio and video reproduction times of different reproduction paths to the audio reproduction gap of the system streams not shared by different program chains PGC.

In addition, by writing information relating to the audio reproduction gap to the reproduction control information contained in the navigation packs NV, the audio reproduction gap and the information relating to the audio reproduction gap can all be contained within a single system stream.

Furthermore, by containing the audio reproduction gap and the information relating to the audio reproduction gap within a single system stream, it is possible to move the audio reproduction gap within the system stream. It is therefore possible to move the audio reproduction gap to a silent space or other place where least audibly perceptible, and thereby achieve a more seamless reproduction.

The internal structure of the system encoder 900 in the DVD encoder ECD shown in Fig. 25 is shown in detail in the block diagram in Fig. 50. Note that the system encoder 900 generates the system streams.

As shown in Fig. 50, the system encoder 900 comprises an elementary stream buffer 3301 for temporarily storing the video, sub-picture, and audio data; a video analyzer 3302 for simulating

the video buffer state; a sub-picture analyzer 3308 for simulating the sub-picture buffer state; an audio analyzer 3303 for simulating the audio buffer state; a movement calculator 3304 for calculating the number of audio frames to move; a packet producer 3305 for packetizing the video data, audio data, and sub-picture data; a multiplexer 3306 for determining the packet arrangement; and a pack producer 3307 for packing the packets to produce the system stream.

The elementary stream buffer 3301 is connected to the video stream buffer 400, sub-picture stream buffer 600, and audio stream buffer 800 shown in Fig. 26, and temporarily stores the elementary streams. The elementary stream buffer 3301 is also connected to the packet producer 3305.

The video analyzer 3302 is connected to the video stream buffer 400, thus receives the encoded video stream St27, simulates the video buffer state, and supplies the simulation result to the movement calculator 3304 and multiplexer 3306.

The audio analyzer 3303 is likewise connected to the audio stream buffer 800, thus receives the encoded audio stream St31, simulates

the audio buffer state, and supplies the simulation result to the movement calculator 3304 and multiplexer 3306.

The sub-picture analyzer 3308 is likewise connected to the sub-picture stream buffer 600, thus receives the encoded sub-picture stream St29, simulates the sub-picture buffer state, and supplies the simulation result to the movement calculator 3304 and multiplexer 3306.

Based on the simulated buffer states, the movement calculator 3304 calculates the audio movement (number of audio frames) and the audio reproduction gap information, and supplies the calculation results to the packet producer 3305 and multiplexer 3306. More specifically, the movement calculator 3304 calculates the audio data movement MFAp1 from the preceding scene, the audio data movement MFAp2 to the preceding scene, the movement MGVp of 1 GOP of video data to the preceding scene, the movement MGVf of 1 GOP of video data from the following scene, the movement MFAf1 of audio data to the following scene, and the movement MFAf2 of audio data from the following scene.

The packet producer 3305 produces the video, sub-picture, and audio packets from the

video data, sub-picture data, and audio data stored in the elementary stream buffer 3301 according to the audio movement calculated by the movement calculator 3304. The packet producer 3305 also produces the reproduction control information, i.e., the navigation packs NV. The audio reproduction gap information is also written to the navigation packs NV at this time.

Based on the audio reproduction gap information and the video and audio buffer state information simulated by the video analyzer 3302 and audio analyzer 3303, the multiplexer 3306 rearranges, i.e., multiplexes, the video packets, audio packets, and navigation packs NV. The movement calculator 3304 also performs based on the audio reproduction gap information.

The pack producer 3307 then packs the packets, adds the system header, and produces the system stream.

Note that the operation of the system encoder 900 is described in detail below with reference to Fig. 53.

The present embodiment relates to connecting system streams by coupling. The next embodiment relates to connecting system streams at

the trailing end of a multi-scene period, i.e., connecting one of plural preceding system streams to the common system stream following thereafter.

The physical structure of the optical disk, the overall data structure of the optical disk, and the DVD decoder DCD in the present embodiment are as described above, and further description thereof is thus omitted below.

Note that the description of the present embodiment below is limited to a single encoded video stream and a single encoded audio stream for simplicity.

Fig. 51 shows the structure of the end of the two parental lock control system streams SSc and SSd, and the beginning of the following common system stream SSe that can be connected to either of the preceding parental lock control system streams SSc and SSd. Note that this figure is basically the same as Fig. 48 in which the parental lock control system streams are the following system streams.

Note that the one of the two parental lock control system streams SSc and SSd and the common system stream SSe are arrayed to the same time-base (horizontal time axis T) as shown in Fig. 51.

System streams SSc, SSd, and SSe shown as separate blocks in Fig. 51 represent the following content as in Fig. 46.

The fifth row in each block shows the structure of system streams SSc, SSd, and SSe. System stream SSc comprises video stream SScv and audio stream SSca; system stream SSd similarly comprises video stream SSdv and audio stream SSda; and system stream SSe comprises video stream SSev and audio stream SSea.

The fourth rows show the audio packet streams A of audio stream SSca, audio stream SSda, and audio stream SSea, extracted from system streams SSc, SSd, and SSe.

The third rows show the data input/output state of the audio buffer 2800 when audio stream SSca, audio stream SSda, and audio stream SSea, are input to the DVD decoder DCD.

The second rows show the video packet streams V of video stream SScv, video stream SSdv, and video stream SSev extracted from system streams SSc, SSd, and SSe.

The first rows show the data input/output state of the video buffer 2600 when video stream

SScv, video stream SSdv, and video stream SSev are input to the DVD decoder DCD.

At the end of system streams SSc and SSd, the video content of at least the last GOP in each video stream SSdv and SSev is the same.

Likewise, the audio content of the last several audio frames in audio streams SSca and SSda at the end of system streams SSc and SSd is the same.

When system stream SSc is input to the DVD decoder DCD, the difference between the input end times of the video stream SScv and audio stream SSca to the respective buffers 2600 and 2800 is small, and at most is less than the reproduction time of two audio frames.

When system stream SSd is input to the DVD decoder DCD, the difference between the input end times of the video stream SSdv and audio stream SSda to the respective buffers 2600 and 2800 is small, and at most is less than the reproduction time of two audio frames.

When system stream SSe is input to the DVD decoder DCD, the difference between the input end times of the video stream SSev and audio stream SSea to the respective buffers 2600 and 2800 is

small, and at most is less than the reproduction time of two audio frames.

The data input/output state of the video buffer 2600 when system stream SSc or system stream SSd is connected to and contiguously reproduced with system stream SSe is the same as shown in Fig. 44. Specifically, system stream SSa in Fig. 44 corresponds to either system stream SSc or system stream SSd in Fig. 51, and system stream SSb in Fig. 44 corresponds to system stream SSe in Fig. 51.

Seamless reproduction can thus be achieved when system stream SSc or system stream SSd is connected and contiguously reproduced with system stream SSe.

Note also that system streams SSc, SSd, and SSe are produced using the second method described above with reference to Fig. 47. More specifically, the system streams can be similarly created by substituting system streams SSc and SSd in Fig. 51 for system stream SSa in Fig. 47, and substituting system stream SSe in Fig. 51 for system stream SSb in Fig. 47. The method of generating the system streams is as described above with reference to Fig. 47.

The data structure of system streams SSc, SSd, and SSe produced according to the method shown in Fig. 47 is shown in Fig. 51, and seamless reproduction can therefore be achieved using the DVD decoder DCD shown in Fig. 26.

As described with reference to audio frame movement Fig. 46, the maximum input end time difference between the video packets and audio packets at the end of system streams SSc and SSd is at most the reproduction time of two audio frames, and the maximum input start time difference between the video packets and audio packets at the beginning of system stream SSe is at most the reproduction time of two audio frames.

By providing an audio reproduction stop, i.e., an audio reproduction gap, when moving and connecting audio frames from audio stream SSea to destination audio streams SSca and SSda, the differences in the video reproduction time and the audio reproduction time of each reproduction path can be contained within the system streams not shared between different program chains PGC.

Fig. 52 is used to describe the difference in the video reproduction time and audio reproduction time of different reproduction paths

according to the present embodiment. In Fig. 52, time Te represents the time difference between the audio and video reproduction start times at the beginning of the adult-oriented title before audio data movement; time Tc' is the time difference between the audio and video reproduction end time at the end of the adult-oriented title stream; and time Td' is the time difference between the audio and video reproduction end times at the end of the child-oriented title stream before moving the audio data.

It is possible to match the time difference between the audio and video reproduction end times of at least one of the plural different reproduction paths before the connection with the time difference of the audio and video reproduction start times following the connection. Note that it is assumed in the following description that Te = Tc', and Te < Td'.

Because Te = Tc' in the adult-oriented title before the connection, the audio frame moved from the common part of the adult-oriented and child-oriented title streams can be connected to the end of the child-oriented title stream without an audio reproduction gap. A seamless stream is

then produced after the connection as shown in the figure.

Because Td' < Te in the child-oriented title stream before the connection, the audio frame moved from the common part of the adult-oriented and child-oriented title streams can be connected to the end of the child-oriented title stream with an audio reproduction gap of only Te - Td'.

To enable seamless reproduction between system stream SSc and system stream SSd at the connection with system stream SSe, the system streams are generated using the second system stream encoding method described above with reference to moving the encoded video stream and audio data from one system stream SSe to another system stream SSc and SSd.

The method of producing the system streams is the same as that described above with reference to Fig. 47 except that system streams SSc and SSd in Fig. 51 are substituted for system stream SSa in Fig. 47, and system stream SSe in Fig. 51 is substituted for system stream SSb in Fig. 47, and further description thereof is thus omitted below.

When producing these system streams, the packets are generated so that the audio frames

before and after the audio reproduction gap are not contained in the same packet. As a result, it is possible to write the audio playback starting time information APTS (the audio frame reproduction start time including the audio reproduction pause time) of the audio frames before and after the audio reproduction gap into the system stream.

The packet containing the audio frame immediately preceding the audio reproduction gap is of necessity small. During the packing process a padding packet is therefore used to produce a fixed-length pack of 2048 bytes/pack.

The audio reproduction gap information for the audio reproduction gap of system stream SSd in this embodiment is inserted to the system stream by writing the audio frame reproduction end time immediately before the audio reproduction gap of the child-oriented title to the audio reproduction stopping time 2 (VOB_A_STP_PTM2) in the navigation pack NV (Fig. 20), and writing the audio reproduction gap time Te - Td' to the audio reproduction stopping period 2 (VOB_A_GAP_LEN2) in the DSI packet.

When there is no audio reproduction gap, it is possible to determine that there is no audio

reproduction gap by writing a 0 value to the audio reproduction stopping time 2 (VOB_A_STP_PTM2).

By means of the above process, it is possible to set the time difference between the different audio and video reproduction times of different reproduction paths to the audio reproduction gap of the system streams not shared by different program chains PGC.

In addition, by writing information relating to the audio reproduction gap to the reproduction control information contained in the navigation packs NV, the audio reproduction gap and the information relating to the audio reproduction gap can all be contained within a single system stream.

Furthermore, by containing the audio reproduction gap and the information relating to the audio reproduction gap within a single system stream, it is possible to move the audio reproduction gap within the system stream. It is therefore possible to move the audio reproduction gap to a silent space or other place where least audibly perceptible, achieve seamless data reproduction not permitting the audio buffer to underflow, and thereby achieve seamless

reproduction of the audio information that is important for human perception of data contiguity.

The system streams described above can be produced using the system encoder 900 of the DVD encoder ECD shown in Fig. 25. The structure of the system encoder 900 is as described above with reference to Fig. 50, and further description thereof is thus omitted below.

The process of producing the above described system streams is described below with reference to Fig. 53. Note that this process is the system encoding subroutine shown as step #2200 of the system encoder flow chart shown in Fig. 34.

System encoder flow chart

The system encoding process is described below with reference to Fig. 53.

At step #307002 the conditions for connecting with the preceding scene are evaluated based on the state of the preceding VOB seamless connection flag VOB_Fsb. If a non-seamless connection with the preceding scene is specified, i.e., VOB_Fsb _ 1, the procedure moves to step #307010.

At step #307010 the movement calculator 3304 (Fig. 50) sets the audio data movement MFAp1

from the preceding scene, i.e., the number of audio frames moved, to 0 based on the VOB_Fsb _ 1 declaration. The procedure then moves to step #307014.

If a seamless connection with the preceding scene is specified, i.e., VOB_Fsb = 1, at step #307002, the procedure moves to step #307004.

At step #307004 it is determined whether the preceding scene is in a multi-scene period. If it is not, the procedure moves to step #307012; if it is, the procedure moves to step #307006.

At step #307012 the audio data movement MFAp1 from the preceding scene is calculated, and the procedure moves to step #307014. Note that the method of calculating the audio data movement MFAp1 is described after this process with reference to Fig. 54 below.

At step #307006 the movement MGVp of 1 GOP of video data to the preceding scene is calculated, and the procedure moves to step #307008. If the preceding scene is in a multi-scene period, it is not possible to uniformly calculate the audio data movement MFAp1 as in step #307012. As a result, the movement of one GOP of video data from the

beginning of the present scene to the preceding scene is calculated.

At step #307008 the audio data movement MFAp2 to the following scene is calculated, and the procedure moves to step #307014. Note that the method of calculating the audio data movement MFAp2 is described after this process with reference to Fig. 55 below.

At step #307014 the conditions for connecting with the following scene are evaluated based on the state of the following VOB seamless connection flag VOB_Fsf. If a non-seamless connection with the following scene is specified, i.e., VOB_Fsf _ 1, the procedure moves to step #307022. If a seamless connection with the following scene is specified, i.e., VOB_Fsf = 1, the procedure moves to step #307016.

At step #307022 the movement calculator 3304 (Fig. 50) sets the audio data movement MFAp1 to the following scene to 0 based on the VOB_Fsb _ 1 declaration. The procedure then moves to step #307026.

At step #307016 it is determined whether the following scene is in a multi-scene period based on the multi-scene flag VOB_Fp. If it is not,

i.e., VOB_Fp _ 1, the procedure moves to step #307024; if it is, i.e., VOB_Fp = 1, the procedure moves to step #307018.

At step #307024 the audio data movement MFAp1, MFAp2 to the following scene is calculated, and the procedure moves to step #307026. Note that the method of calculating the audio data movement MFAp2 is the same as that used in step #307012.

At step #307018 the movement MGVf of 1 GOP of video data from the following scene is calculated, and the procedure moves to step #307020.

At step #307020 the audio data movement MFAp2 from the following scene is calculated, and the procedure moves to step #307026. Note that the method of calculating the audio data movement MFAf2 is the same as that used in step #307008.

At step #307026 the audio reproduction stopping time 1 (VOB_A_STP_PTM1) and the audio reproduction stopping period 1 (VOB_A_GAP_LEN1) are calculated from the audio and video end times of the preceding scene. The procedure then moves to step #307028.

At step #307028 the audio reproduction stopping time 2 (VOB_A_STP_PTM2) and the audio

reproduction stopping period 2 (VOB_A_GAP_LEN2) are calculated from the audio and video start times in the following scene. The procedure then moves to step #307030.

At step #307030 the audio data, including the audio movement, is packetized, and the procedure moves to step #307032.

At step #307032 the video data, including the video movement, is packetized, and the procedure moves to step #307034.

At step #307034 the navigation pack NV is generated, the audio reproduction stopping time 1 (VOB_A_STP_PTM1) and the audio reproduction stopping period 1 (VOB_A_GAP_LEN1), and the audio reproduction stopping time 2 (VOB_A_STP_PTM2) and the audio reproduction stopping period 2 (VOB_A_GAP_LEN2) are recorded, and the procedure moves to step #307036.

At step #307036 the video packets V, audio packets A, and navigation pack NV are multiplexed.

As described above, it is thus possible to move audio and video data between scenes according to the conditions for connections with the preceding and following scenes, and generate the system stream accordingly.

The method of calculating the audio data movement MFAp1 in step #307012 above is described below with reference to Fig. 54.

In Fig. 54 video 1 is the video data at the end of the preceding scene, with the video 1 line representing the change in video data accumulation at the end of the preceding scene in the video buffer 2600 of the DVD decoder DCD; video 2 is similarly the video data at the beginning of said scene with the video 2 line representing the change in said video data accumulation in the video buffer 2600 at the beginning of said scene.

Note that both video 1 and video 2 represent the state of the video buffer before system stream connection. VDTS is the time video 2 is first decoded; tv is the video 2 transfer start time, and is calculated from equation 30631 below where video buffer verifier delay vbv_delay is defined as the time from the start of data input to the video buffer to the start of decoding. If decoding starts at vbv_delay after the start of data input to the video buffer, a video buffer data underflow state can be reliably prevented during the following decoding process.

$$tv = VDTS - vbv_delay \quad [1]$$

Audio 1 shows the transfer of audio frames at the end of the preceding scene to the audio buffer where af1, af2, af3, and af4 are the audio frames contained in audio 1. Note that the audio frame is the encoding process unit, and contains the audio data for a defined period of time (Af).

Audio 2 shows the transfer of audio frames at the beginning of the scene to the audio buffer where af5 and af6 are the audio frames contained in audio 2.

APTS is the time the audio in audio 2 is first reproduced.

The audio frames (af3, af4) transferred during period APTS from time tv, i.e., the number of audio frames (Amove) MFAp1 attached to audio 1 transferred after the start of video 2 transfer, is calculated according to equation 2.

$$Amove = (APTS - tv - Af) / Af \quad [2]$$

The audio data movement (number of audio frames) from the preceding scene is thus calculated.

The method of calculating the audio data movement MFAp2 to the following scene in step #307008 above is described below with reference to Fig. 55.

As in Fig. 54, video 1 is the video data at the end of the preceding scene, and video 2 is similarly the video data at the beginning of said scene. Note that both video 1 and video 2 show the video buffer state before scene connection. VDTS is the time video 2 is first decoded; GOP_move is the one GOP video data GMVp moved in step #307006; tv is the time video 2 transfer starts after moving the GOP_move quantity of GOP, and can be uniformly calculated.

Audio 1 shows the transfer of audio frames at the end of the preceding scene to the audio buffer where af1, af2, af3, and af4 are the audio frames contained in audio 1. Note that the audio frame is the encoding process unit, and contains the audio data for a defined period of time (Af).

Audio 2 shows the transfer of audio frames at the beginning of the scene to the audio buffer where af5, af6, and af7 are the audio frames contained in audio 2.

APTS is the time the audio in audio 2 is first reproduced.

The audio frames (af5, af6, af7) transferred during period APTS from time tv, i.e., the number of audio frames (Amove) MFAp1 attached

to audio 2 transferred before the start of video 2 transfer after moving GOP_move quantity of GOP, is calculated according to equation 3.

$$Amove = (APTS - tv + 2Af) / Af \quad [3]$$

The audio data movement (number of audio frames) to the preceding scene is thus calculated.

Audio gap reproduction processing

While the basic structure of the DVD decoder DCD used in the present embodiment is as shown in Fig. 26, the synchronizer 2900 is structured as shown in Fig. 56 to process the audio reproduction gap.

As shown in Fig. 56, a block diagram of the synchronizer 2900 shown in Fig. 26, the synchronizer 2900 comprises an STC generator 2950, audio decoder controller 2952, and audio decoder control data buffer 2954.

The STC generator 2950 generates the system clock STC used as the reference clock for decoding control based on the system clock reference SCR value set by the decoding system controller 2300.

The audio decoder controller 2952 controls the decoding start and stop of the audio decoder 3200 based on the STC value from the STC generator

2950 and the control information from the audio decoder control data buffer 2954.

The audio decoder control data buffer 2954 stores the values of the audio decoding control information (such as VOB_A_STP_PTM and VOB_A_GAP_LEN) set by the decoding system controller 2300.

The operation of the synchronizer 2900 thus comprised according to the present embodiment is described below with reference to Fig. 26 and Fig. 56.

The overall operation of the DVD decoder DCD in Fig. 26 is as previously described, and further description thereof is thus omitted below. The operation related to the specific processes of the present embodiment is described below.

Referring to Fig. 26, the decoding system controller 2300 reads the audio reproduction stopping time 1 (VOB_A_STP_PTM1), the audio reproduction stopping period 1 (VOB_A_GAP_LEN1), the audio reproduction stopping time 2 (VOB_A_STP_PTM2), and the audio reproduction stopping period 2 (VOB_A_GAP_LEN2) from the DSI packet in the navigation pack NV, and stores these four values as the audio decode reproduction

stopping information to the audio decoder control data buffer 2954 of the synchronizer 2900.

When the time supplied from the STC generator 2950 matches the audio reproduction stopping time 1 (VOB_A_STP_PTM1) stored in the audio decoder control data buffer 2954, the audio decoder controller 2952 stops the audio decoder 3200 for the audio reproduction stopping period 1 (VOB_A_GAP_LEN1) stored in the audio decoder control data buffer 2954. Likewise when the time supplied from the STC generator 2950 matches the audio reproduction stopping time 2 (VOB_A_STP_PTM2) stored in the audio decoder control data buffer 2954, the audio decoder controller 2952 stops the audio decoder 3200 for the audio reproduction stopping period 2 (VOB_A_GAP_LEN2) stored in the audio decoder control data buffer 2954.

By thus comprising a STC generator 2950 and audio decoder controller 2952, the synchronizer 2900 is able to process audio reproduction gaps contained in the system stream of a multi-scene period when connecting a stream from a multi-scene period with a common scene stream.

Note that an audio reproduction gap may occur in the present invention in one or both of

VOB 6 and VOB 7 corresponding to scenes 6 and 7 in a parental lock control scene period as shown in Fig. 21.

The decoding process executed by the decoding system controller 2300 of the present invention is described briefly below with reference to Fig. 60, Fig. 61, Fig. 62, Fig. 63, and Fig. 64. The process executed by the audio decoder controller 2952 of the present invention is then described with reference to Fig. 57.

In Fig. 60 the title selected by the user is extracted from the multimedia bitstream MBS stored to the digital video disk, and the VTS_PGCI #i program chain (PGC) data for reproducing the selected title is extracted by the decoding system controller 2300 of the DVD decoder DCD at step #310214. Then at step #310216 the selected title is reproduced based on the extracted VTS_PGCI #i program chain (PGC) information. The process shown in Fig. 60 has already been described in detail above, and further description thereof is thus omitted below.

The process of reproducing the VTS_PGCI #i program chain in step #310216, Fig. 60, is shown in Fig. 61 and described below.

At step #31030 the decoding system table shown in Fig. 58 is set. The transfer process to the stream buffer 2400 (step #31032), and the data decoding process in the stream buffer 2400 (step #31034) are executed in parallel. Note that the process of step #31032 is based on the cell reproduction information in the PGC information entries C_PBI #j. The process shown in Fig. 61 has already been described in detail above, and further description thereof is thus omitted below.

The stream buffer data transfer executed for each cell reproduction information entry (PGC information entries C_PBI #j) by the process of step #31032 is described in further detail below referring to Fig. 62. Because a parental lock control scene is being processed in the present embodiment, step #31040 of Fig. 62 returns NO, and the procedure moves to step #30144. The process shown in Fig. 62 has already been described in detail above, and further description thereof is thus omitted below.

The non-multi-angle cell decoding process, i.e., the parental lock control cell decoding process executed as step #31044, Fig. 62, is described further below with reference to Fig. 63.

Step #31050 evaluates the interleaved allocation flag IAF_reg to determine whether the cell is in an interleaved block. Because the seamless connection, parental lock control title processed by the present embodiment is arrayed to an interleaved block, step #31050 routes control to step #31052. The process shown in Fig. 63 has already been described in detail above, and further description thereof is thus omitted below.

The non-multi-angle interleaved block process (step #31052, Fig. 63) is described further below with reference to Fig. 64. At step #31062 the audio reproduction stopping time 1 (VOB_A_STP_PTM1), the audio reproduction stopping period 1 (VOB_A_GAP_LEN1), the audio reproduction stopping time 2 (VOB_A_STP_PTM2), and the audio reproduction stopping period 2 (VOB_A_GAP_LEN2) are extracted as the table data from the DSI packet in the navigation pack NV (Fig. 20) and stored to the audio decoder control data buffer 2954 (Fig. 56). The procedure then moves to step #31064 whereby VOB data transfer is continued until it is determined at step #31066 that all interleave units in the interleaved block have been transferred.

The process executed by the audio decoder controller 2952 in Fig. 56 is described next with reference to Fig. 57.

At step #202301 the audio decoder controller 2952 reads the audio reproduction stopping time 1 (VOB_A_STP_PTM1) from the audio decoder control data buffer 2954, and compares VOB_A_STP_PTM1 with the system clock STC from the STC generator 2950. If the values match, i.e., a YES is returned, the procedure moves to step #202302; if the values do not match, i.e., a NO is returned, the procedure moves to step #202303.

At step #202302 the audio reproduction stopping period 1 (VOB_A_GAP_LEN1) is read from the audio decoder control data buffer 2954, and the audio decoder 3200 is stopped for this period.

At step #202303, the audio decoder controller 2952 reads the audio reproduction stopping time 2 (VOB_A_STP_PTM2) from the audio decoder control data buffer 2954, and compares VOB_A_STP_PTM2 with the system clock STC from the STC generator 2950. If the values match, i.e., a YES is returned, the procedure moves to step #202304; if the values do not match, i.e., a NO is returned, the procedure returns to step #202301.

At step #202304 the audio reproduction stopping period 2 (VOB_A_GAP_LEN2) is read from the audio decoder control data buffer 2954, and the audio decoder 3200 is stopped for this period.

The audio reproduction stopping time information (VOB_A_STP_PTM and VOB_A_GAP_LEN) is thus written to the DSI packet of the navigation pack NV in the system stream. Based on this audio reproduction stopping time information, the DVD decoder DCD comprising an audio decoder control data buffer 2954 and an audio decoder controller 2952 for controlling the audio stream decoding operation is able to process audio reproduction gaps found in parental lock control scenes, i.e., in system streams shared by plural different program chains as shown in Fig. 30. It is therefore able to prevent intermittent video reproduction (video freezing) and intermittent audio reproduction (muting) caused by a data underflow state in the video buffer or audio buffer resulting when one common system stream is connected to one of plural system streams branching from (following) or to (preceding) the one system stream.

Note that while audio data is moved in audio frame units in the above embodiment, the same

effect can be achieved if the audio frames are broken into smaller units used as the movement unit to connect and contiguously reproduce system streams.

Furthermore, while video data is moved in GOP units according to the second system stream production method in the above embodiment, the same effect can be achieved if the GOP units are broken into smaller units used as the movement unit to connect and contiguously reproduce system streams.

Furthermore, while only audio data is moved according to the first system stream production method in the above embodiment, the same effect can be achieved if video data is also moved from the system stream preceding the connection to the system stream following the connection.

The present embodiment has also been described with reference to only one video stream and one audio stream, but the invention shall not be so limited.

While the present embodiment has been described with particular reference to branching and connecting streams as used to implement a parental lock control feature, seamless contiguous reproduction can also be achieved in multi-angle

scene periods in which the plural video streams provide different perspectives (views) of the same title content, and using multimedia optical disks to which system streams configured as described above are recorded.

The second system stream production method described above is described as being used at connections from one of plural system streams to a single common system stream in the present embodiment. However, the same effect can be achieved using the first system stream production method described above when the same audio information is recorded to system streams not shared between different program chains.

The present embodiment was also described using a digital video disk DVD, but the same effect can be achieved using other optical disks recording system streams having the same data structure as that of the present embodiment described above.

With the audio and video data interleaving method of the present embodiment the audio data input by the decoding time includes only the data used in the next audio decode operation and any remainder from the packet transfer operation (approximately 2 KB). However, insofar as an audio

buffer underflow state does not occur, i.e., insofar as the interleaving method interleaves the audio and video data to transfer audio data in a quantity and frequency preventing an audio buffer underflow state, the same effect can be achieved.

Information relating to the audio reproduction gap at a system stream branch is written to the audio reproduction stopping time 1 (VOB_A_STP_PTM1) and the audio reproduction stopping period 1 (VOB_A_GAP_LEN1) fields of the navigation pack NV in the present embodiment, but this audio reproduction gap information may be written to the audio reproduction stopping time 2 (VOB_A_STP_PTM2) and the audio reproduction stopping period 2 (VOB_A_GAP_LEN2) fields.

Information relating to the audio reproduction gap at a system stream connection is written to the audio reproduction stopping time 2 (VOB_A_STP_PTM2) and the audio reproduction stopping period 2 (VOB_A_GAP_LEN2) fields of the navigation pack NV in the present embodiment, but this audio reproduction gap information may be written to the audio reproduction stopping time 1 (VOB_A_STP_PTM1) and the audio reproduction stopping period 1 (VOB_A_GAP_LEN1) fields.

The difference between the input end times to the respective buffers of the audio and video data in the system stream is defined as at most the reproduction time of two audio frames in this embodiment. However, if the video is encoded with variable bit rate (VBR) coding and the video bit rate before the connection is reduced, the same effect can be achieved even when the input start time of the video data to the video buffer is advanced.

The difference between the input start times to the respective buffers of the audio and video data in the system stream is defined as at most the reproduction time of two audio frames in this embodiment. However, if the video is encoded with variable bit rate (VBR) coding and the video bit rate before the connection is reduced, the same effect can be achieved even when the input end time of the video data to the video buffer is delayed.

The present embodiment is also described as accumulating one audio frame in the audio buffer when system streams are connected, but the present invention shall not be so limited and the same effects can be achieved if a different audio buffer

accumulation level is used insofar as an audio buffer overflow state is not induced.

Furthermore, while video data is moved in GOP units in the above embodiment, if the video data input bit rates differ in the connected system streams, the same effect can be achieved by encoding the GOP to be moved at the input bit rate of the video data in the system stream to which the GOP is moved.

The compressed audio and video streams are also used for data movement in the above embodiment, but the same effect can be achieved by first moving the data at the pre-encoded material level.

Only one GOP is also moved in the above embodiment, but the same effect can be achieved by moving two or more, i.e., plural, GOP.

It is therefore possible by means of the present invention thus described to reproduce system streams from different program chains as a single contiguous title without intermittent video presentation (freezing) or intermittent audio presentation (muting) when connecting and contiguously reproducing plural system streams from a multimedia optical disk recorded with the video

packets and audio packets interleaved to a single system stream meeting the following conditions:

(a) the difference between the input start time of the first video packet and the input start time of the first audio packet at the beginning of the system stream is less than the reproduction time of the number of audio frames that can be stored in the audio buffer plus one audio frame, and

(b) the difference between the input end time of the last video packet and the input end time of the last audio packet at the end of the system stream is less than the reproduction time of the number of audio frames that can be stored in the audio buffer plus one audio frame.

Using a multimedia optical disk recorded with a system stream containing plural scenario branches, i.e., plural system streams branching from a single system stream to which said plural system streams may connect, where at least the same audio content is recorded to one or more audio frames at the beginning of each of the plural system streams connecting to said single system stream, it is particularly possible to reproduce plural scenario titles as single natural titles

without stopping the video presentation (video freezing) at the system stream connection when connecting and contiguously reproducing the system streams.

Using a multimedia optical disk recorded with a system stream containing plural scenario connections, i.e., plural system streams connecting to a single system stream following thereafter, where at least the same video content is recorded to one or more video frames at the beginning of each of the plural system streams connecting to said single system stream end of each of the plural system streams connecting to said single system stream or beginning of the single system stream connecting to the plural system streams, it is particularly possible to reproduce plural scenario titles as single natural titles without stopping the video presentation (video freezing) at the system stream connection when connecting and contiguously reproducing the system streams.

Video or audio buffer underflow states at system stream connections, i.e., intermittent video presentation (video freezing) or intermittent audio presentation (audio muting), resulting from the time difference in the video and audio reproduction

times of different reproduction paths can also be prevented by means of a DVD reproduction apparatus wherewith audio reproduction gap information is recorded to the reproduction control information, and said audio reproduction gap information is used by an audio decoder controller to start and stop audio decoder operation appropriately.

By inserting a time difference in the video and audio reproduction times of different reproduction paths as an audio reproduction gap in one system stream not shared by different program chains, problems created by system stream connections, i.e., across system streams, can be converted to a problem contained within a single system stream. It is therefore possible to contain the audio reproduction gap information within the DSI packet of the system stream, thus writing both the audio reproduction gap and the audio reproduction gap information to a single system stream, and thereby simplifying the data structure.

As a result, the present invention makes it simple to reuse, i.e., share, system streams.

Furthermore, because the audio reproduction gap is contained within a single

system stream, the audio reproduction gap can be moved to any desirable position in the system stream. As a result, it is possible to move the audio reproduction gap to a silent or other audibly innocuous location.

Industrial Applicablity

As is apparent from a method and an apparatus according to the present invention for interleaving a bitstream to record the interleaved bitstream to a recording medium and reproduce the recorded bitstream therefrom is suitable for the application of an authoring system which can generate a new title by editing a title constructed by bitstreams carrying various information in accordance with the user's request, and is also suitable for a Digital Video Disk System, or DVD System being developed recently.

This Sheet Has Been Replaced

CLAIMS

1. An optical disc (M) for recording one or a plurality of system streams (VOB) containing audio data and video data, wherein audio data and video data of a plurality of system streams (VOB) recorded to the optical disc (M) are interleaved (multiplexed) such that a difference (Tb1 - Tad1) of the input start times of video data and audio data to a buffer (2600) in a video decoder and a buffer (2800) in an audio decoder is less than the reproduction time of a number of audio frames (Af) that can be stored in the audio buffer plus one frame.

2. An optical disc (M) for recording one or a plurality of system streams (VOB) containing audio data and video data,

wherein audio data and video data of a plurality of system streams (VOB) recorded to the optical disc (M) are interleaved (multiplexed) such that a difference between the input start times of the video data and audio data to a buffer in the video decoder and a buffer in the audio decoder is less than the reproduction time of two audio frames (2 x Af).

3. The optical disc (M) according to claim 1, wherein audio data and video data of a plurality of system streams (VOB) recorded to the optical disc (M) are interleaved such that a difference (Tvae - Taae) of the input end times of video data and audio data to a buffer (2600) in a video decoder and a buffer (2800) in an audio decoder is less than the reproduction time of a number of audio frames (Af) that can be stored in the audio buffer plus one frame (Af).

4. The optical disc (M) according to claim 3, wherein audio data and video data of a plurality of system streams (VOB) recorded to the optical disc (M) are interleaved such that a difference of the input end times of video data and audio data to a buffer (2600) in a video decoder and a buffer (2800) in an audio decoder is less than the reproduction time of two audio frames (2 x Af).

5. An optical disc (M) for recording one or a plurality of system streams (VOB) containing audio data and video data,

wherein when one or a plurality of system streams (VOB) is shared by a plurality of program chains (VTS_PGC), and there are at least two different system streams (VOB) reproduced immediately

* C L M S *

This Sheet Has Been Replaced

after a system stream (VOB) shared by a plurality of program chains (VTS_PGC) that are not the same in each of the plurality of system streams (VOB),

a same audio content is recorded to at least a first audio frame (Af) in at least two system streams (VOB) reproduced immediately after a system stream (VOB) shared by a plurality of program chains (VTS_PGC).

6. The optical disc (M) according to claim 5, wherein when one or a plurality of system streams (VOB) is shared by a plurality of program chains (VTS_PGC), and there are at least two different system streams (VOB) reproduced immediately after a system stream (VOB) shared by a plurality of program chains (VTS_PGC) that are not the same in each of the plurality of system streams (VOB),

a same video content is recorded to at least a first video frame (Vf) in at least two system streams (VOB) reproduced immediately after a system stream (VOB) shared by a plurality of program chains (VTS_PGC).

7. The optical disc (M) according to claim 6 and claim 7 [sic], wherein when one or a plurality of system streams (VOB) is shared by a plurality of program chains (VTS_PGC), and there are at least two

* C L M S *

This Sheet Has Been Replaced

different system streams (VOB) reproduced immediately before a system stream (VOB) shared by a plurality of program chains (VTS_PGC) that are not the same in each of the plurality of system streams (VOB),

a same video content is recorded to at least a last video frame (Vf) in at least two system streams (VOB) reproduced immediately before a system stream (VOB) shared by a plurality of program chains (VTS_PGC).

8. The optical disc (M) according to claim 5, claim 6, or claim 7, wherein when one or a plurality of system streams (VOB) is shared by a plurality of program chains (VTS_PGC), and there are at least two different system streams (VOB) reproduced immediately before a system stream (VOB) shared by a plurality of program chains (VTS_PGC) that are not the same in each of the plurality of system streams (VOB),

a same audio content is recorded to at least a last audio frame (Af) in at least two system streams (VOB) reproduced immediately before a system stream (VOB) shared by a plurality of program chains (VTS_PGC).

9. The optical disc (M) according to claim 5, claim 6, claim 7, or claim 8, wherein reproduction control information (NV) is provided in a system

* C L M S *

This Sheet Has Been Replaced

stream (VOB), and audio reproduction stop information (VOB_A_STP_PTM1 and VOB_A_GAP_LEN) are written to said reproduction control information.

10. A recording method for recording one or a plurality of system streams (VOB) containing audio data and video data to an optical disc (M), whereby one or a plurality of system streams (VOB) are recorded with the audio data and video data interleaved such that a difference of the input start times of video data and audio data to a buffer (2600) in a video decoder and a buffer (2800) in an audio decoder is less than the reproduction time of a number of audio frames (Af) that can be stored in the audio buffer plus one frame.

11. The recording method according to claim 10, wherein one or a plurality of system streams (VOB) are recorded with the audio data and video data interleaved such that a difference of the input end times of video data and audio data to a buffer (2600) in a video decoder and a buffer (2800) in an audio decoder is less than the reproduction time of a number of audio frames (Af) that can be stored in the audio buffer (2800) plus one frame.

12. A recording method for recording one or a plurality of system streams (VOB) containing audio

data and video data to an optical disc (M), whereby when one or a plurality of system streams (VOB) is shared by a plurality of program chains (VTS_PGC), and there are at least two different system streams (VOB) reproduced immediately after a system stream (VOB) shared by a plurality of program chains (VTS_PGC) that are not the same in each of the plurality of system streams (VOB),

a same audio content is recorded to at least a first audio frame (Af) in at least two system streams (VOB) reproduced immediately after a system stream (VOB) shared by a plurality of program chains (VTS_PGC).

13. The recording method according to claim 12, wherein when one or a plurality of system streams (VOB) is shared by a plurality of program chains (VTS_PGC), and there are at least two different system streams (VOB) reproduced immediately after a system stream (VOB) shared by a plurality of program chains (VTS_PGC) that are not the same in each of the plurality of system streams (VOB),

a same video content is recorded to at least a first video frame (Vf) in at least two system streams (VOB) reproduced immediately after a system

stream (VOB) shared by a plurality of program chains (VTS_PGC).

14. The recording method according to claim 12 and claim 13, wherein when one or a plurality of system streams (VOB) is shared by a plurality of program chains (VTS_PGC), and there are at least two different system streams (VOB) reproduced immediately before a system stream (VOB) shared by a plurality of program chains (VTS_PGC) that are not the same in each of the plurality of system streams (VOB),

a same video content is recorded to at least a last video frame (Vf) in at least two system streams (VOB) reproduced immediately before a system stream (VOB) shared by a plurality of program chains (VTS_PGC).

15. The recording method according to claim 12, claim 13, or claim 14, wherein when one or a plurality of system streams (VOB) is shared by a plurality of program chains (VTS_PGC), and there are at least two different system streams (VOB) reproduced immediately before a system stream (VOB) shared by a plurality of program chains (VTS_PGC) that are not the same in each of the plurality of system streams (VOB),

a same audio content is recorded to at least a last audio frame (Af) in at least two system streams (VOB) reproduced immediately before a system stream (VOB) shared by a plurality of program chains (VTS_PGC).

16. The recording method according to claim 12, claim 13, claim 14, or claim 15, wherein reproduction control information (NV) is provided in a system stream (VOB), and audio reproduction stop information (VOB_A_STP_PTM1 and VOB_A_GAP_LEN) are written to said reproduction control information.

17. An optical disc reproduction apparatus (DCD) comprising a data read means for reading reproduction control information from an optical disc (M) to which is recorded a system stream (VOB) containing reproduction control information declaring audio reproduction stop information (VOB_A_STP_PTM1 and VOB_A_GAP_LEN), and

an audio reproduction stop control (VOB_A_STP_PTM1 and VOB_A_GAP_LEN) means for stopping audio reproduction based on the read reproduction control information.

18. A method for reproducing data from an optical disc (M), whereby reproduction control information (NV) is read from an optical disc (M) to

* C L M S *

This Sheet Has Been Replaced

which is recorded a system stream (VOB) containing reproduction control information declaring audio reproduction stop information (VOB_A_STP_PTM1 and VOB_A_GAP_LEN), and

audio reproduction stop (VOB_A_STP_PTM1 and VOB_A_GAP_LEN) control stops audio reproduction based on the read reproduction control information (NV).