

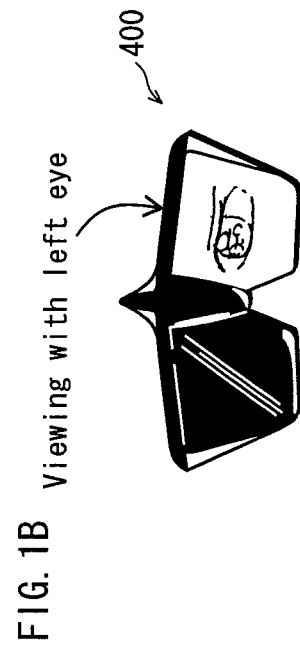
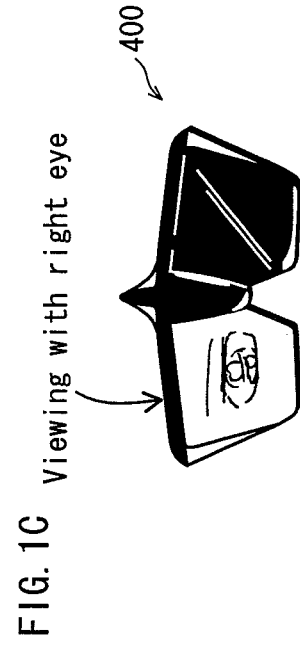
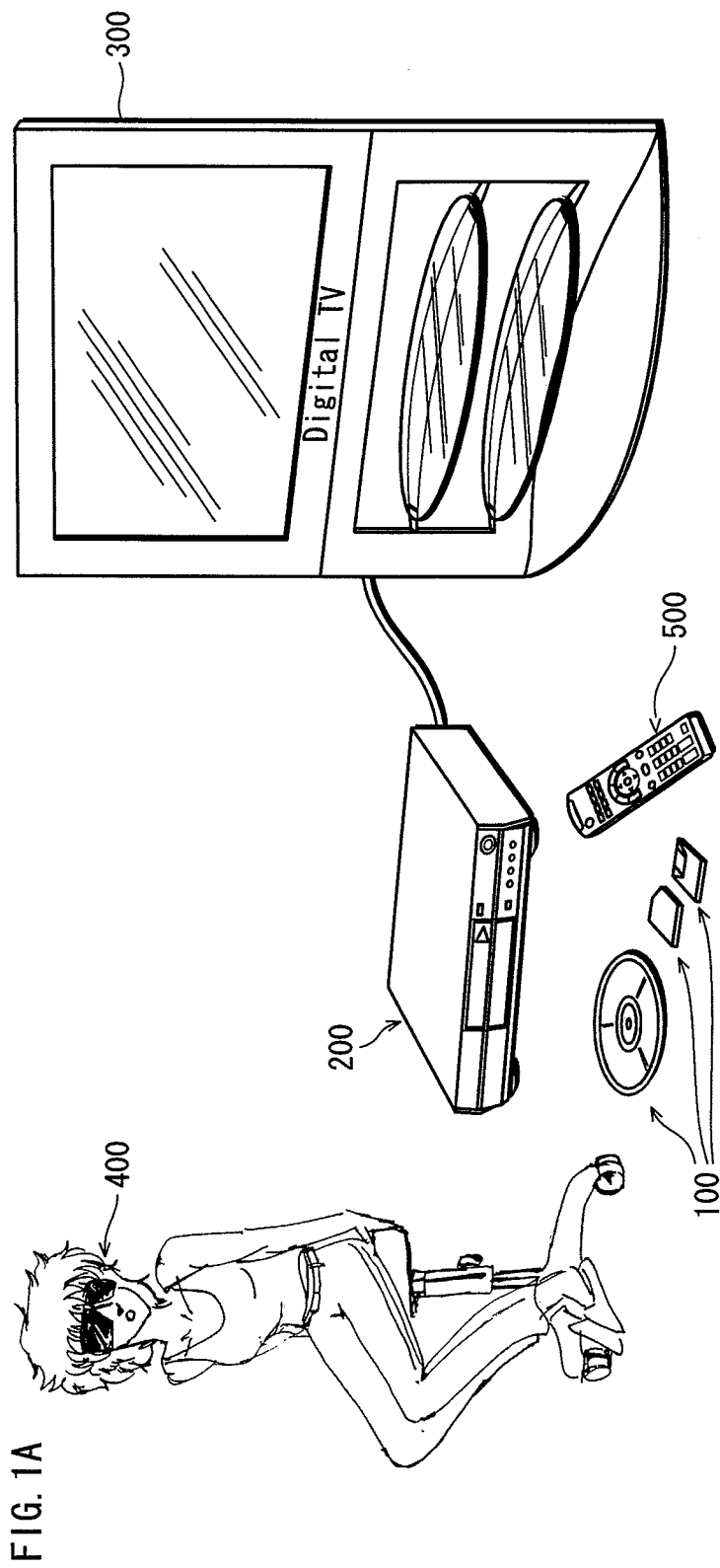


FIG. 2

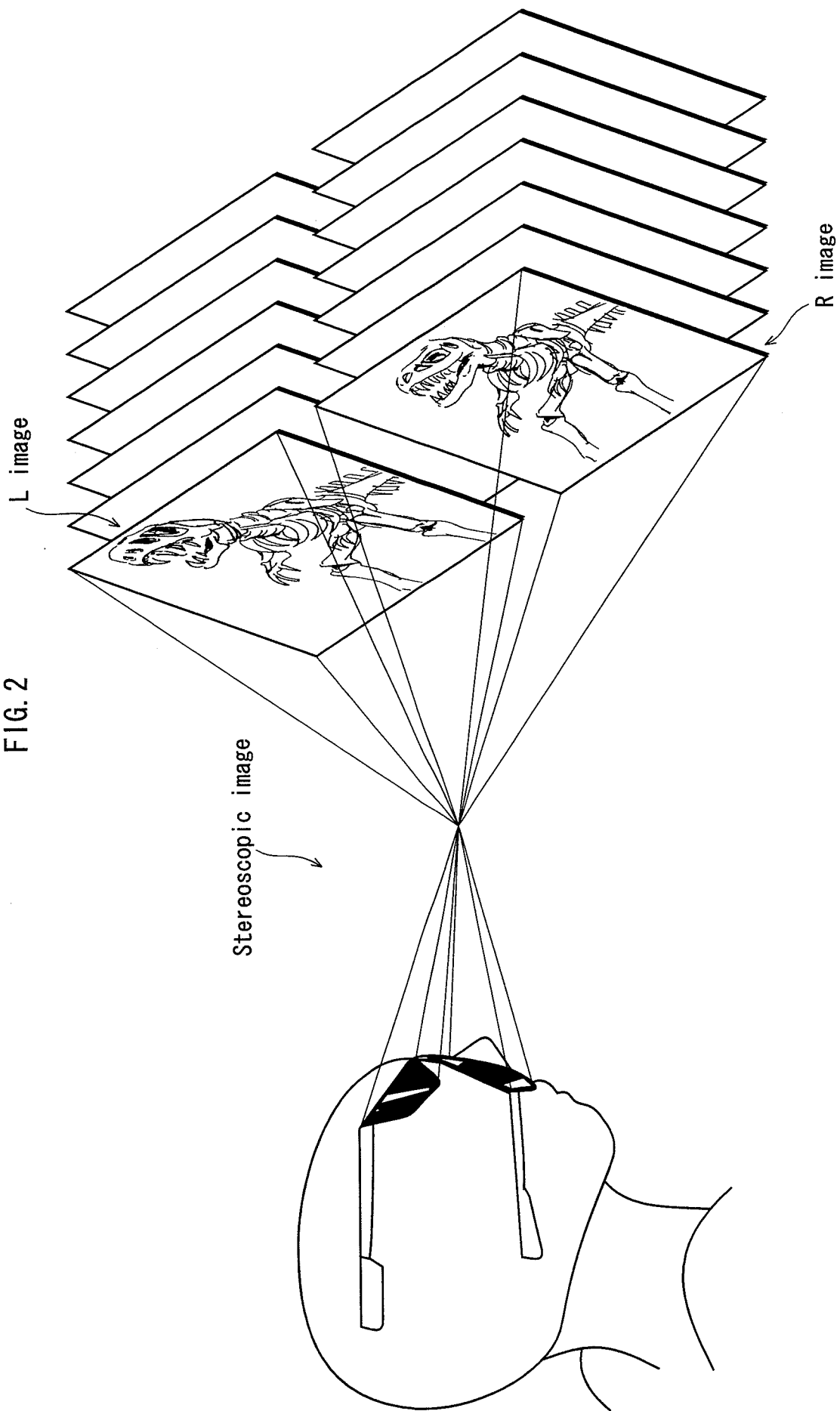


FIG. 3

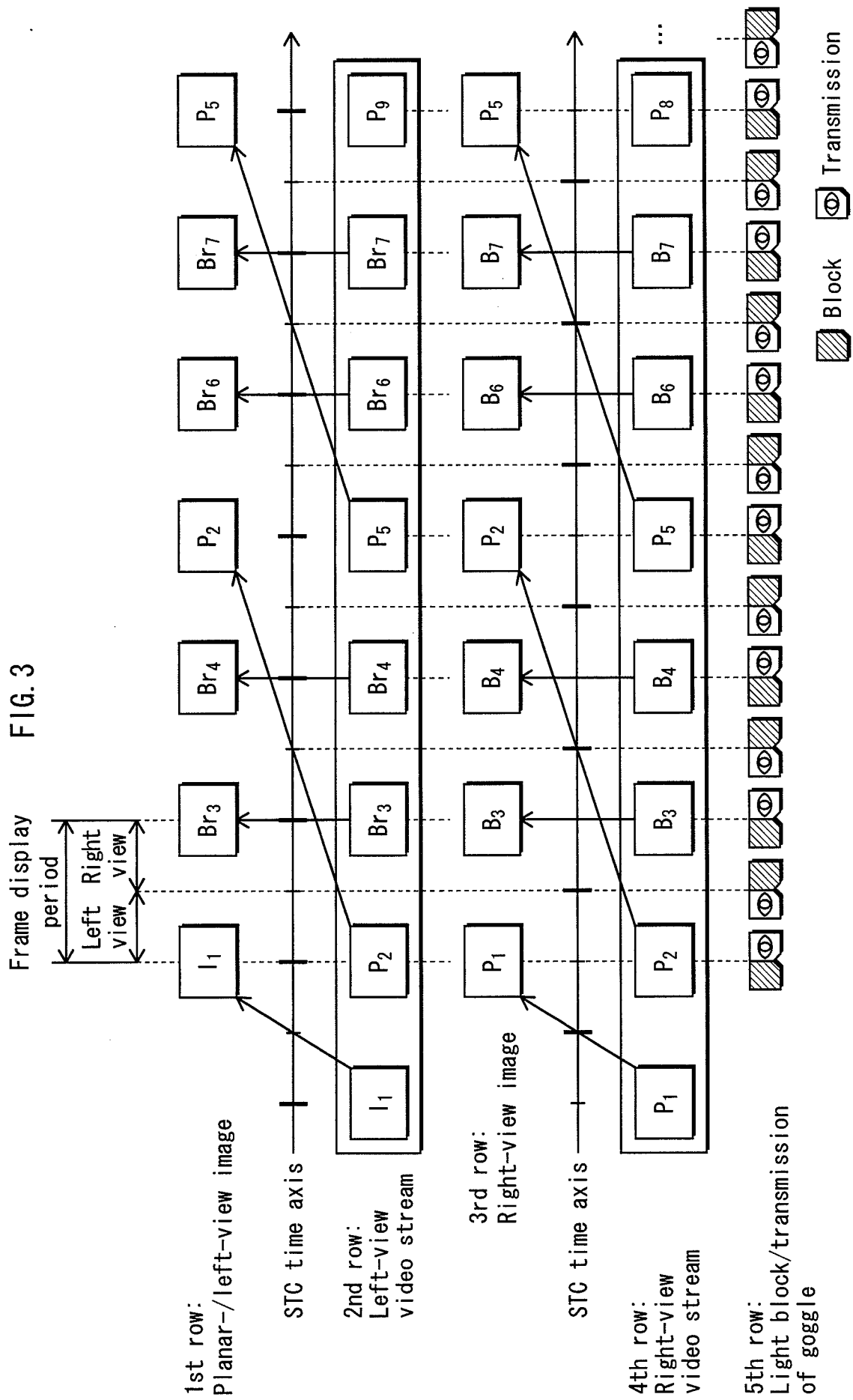


FIG.4

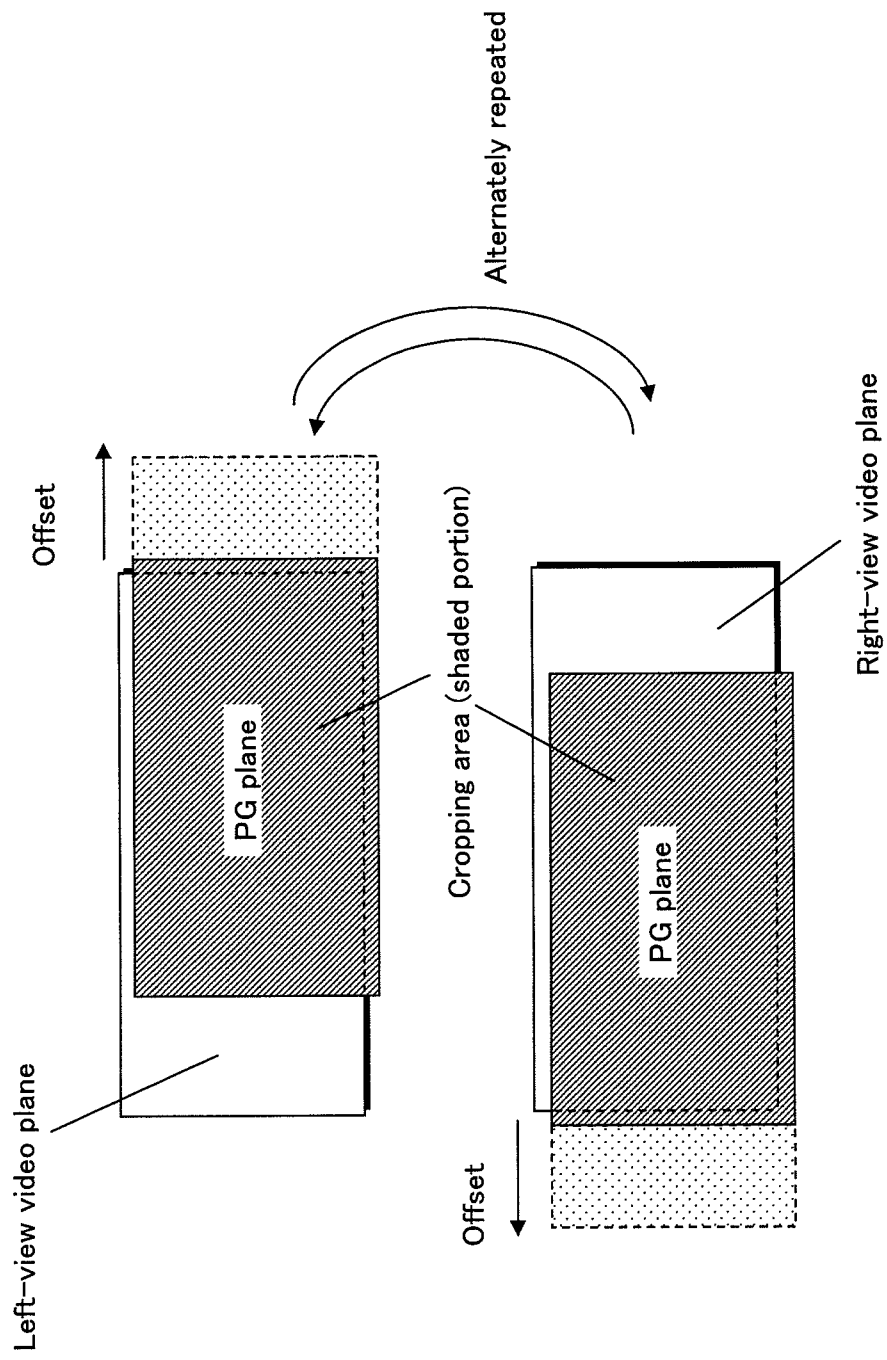


FIG.5

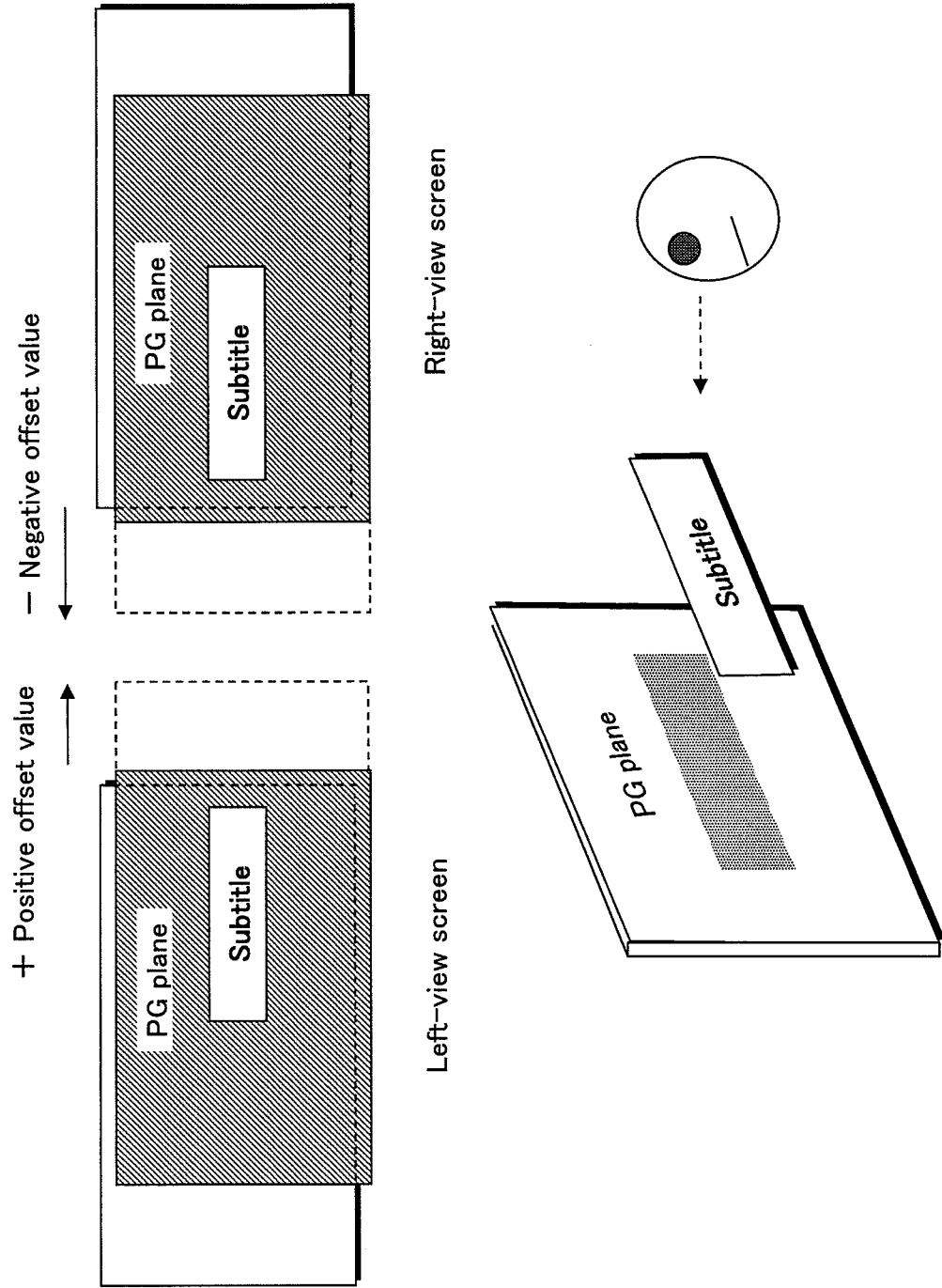


FIG. 6A

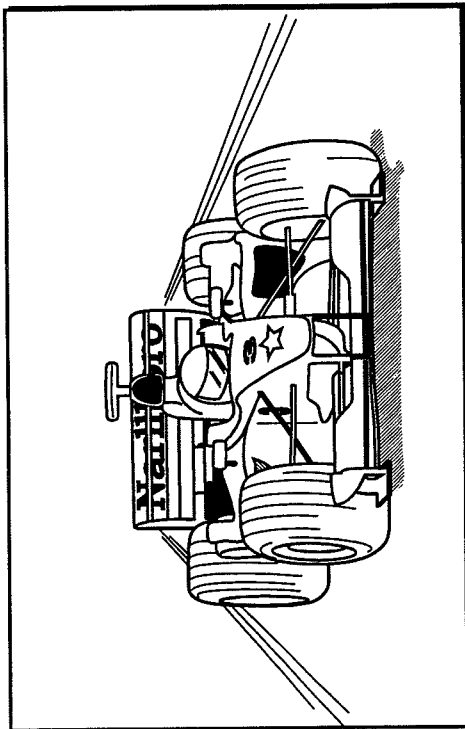
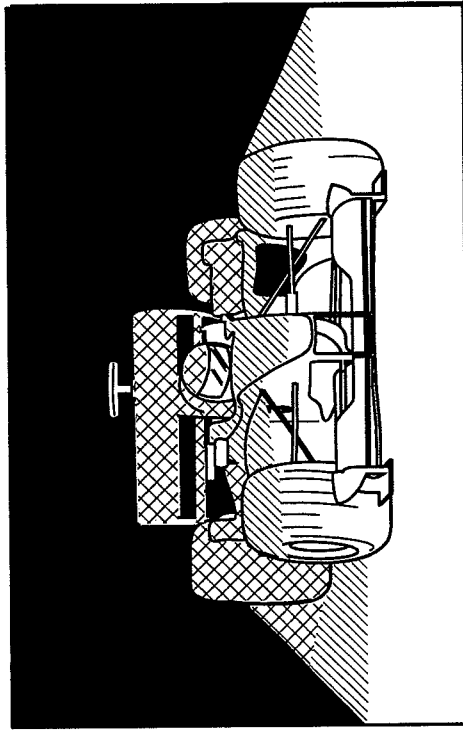


FIG. 6B Grayscale representing depth



Relationship between image and brightness

■	▨	...	□
Deeper			Shallower

FIG. 6C

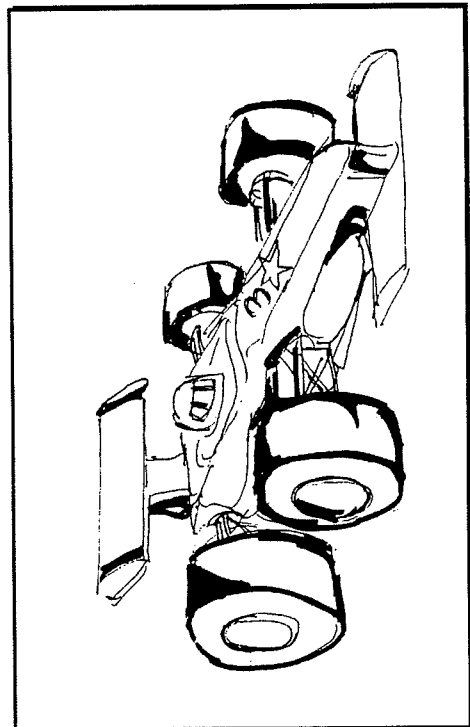


FIG. 6D

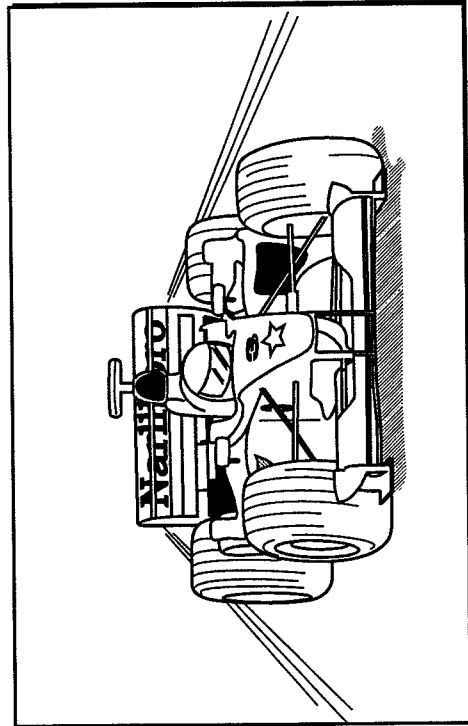


FIG. 7

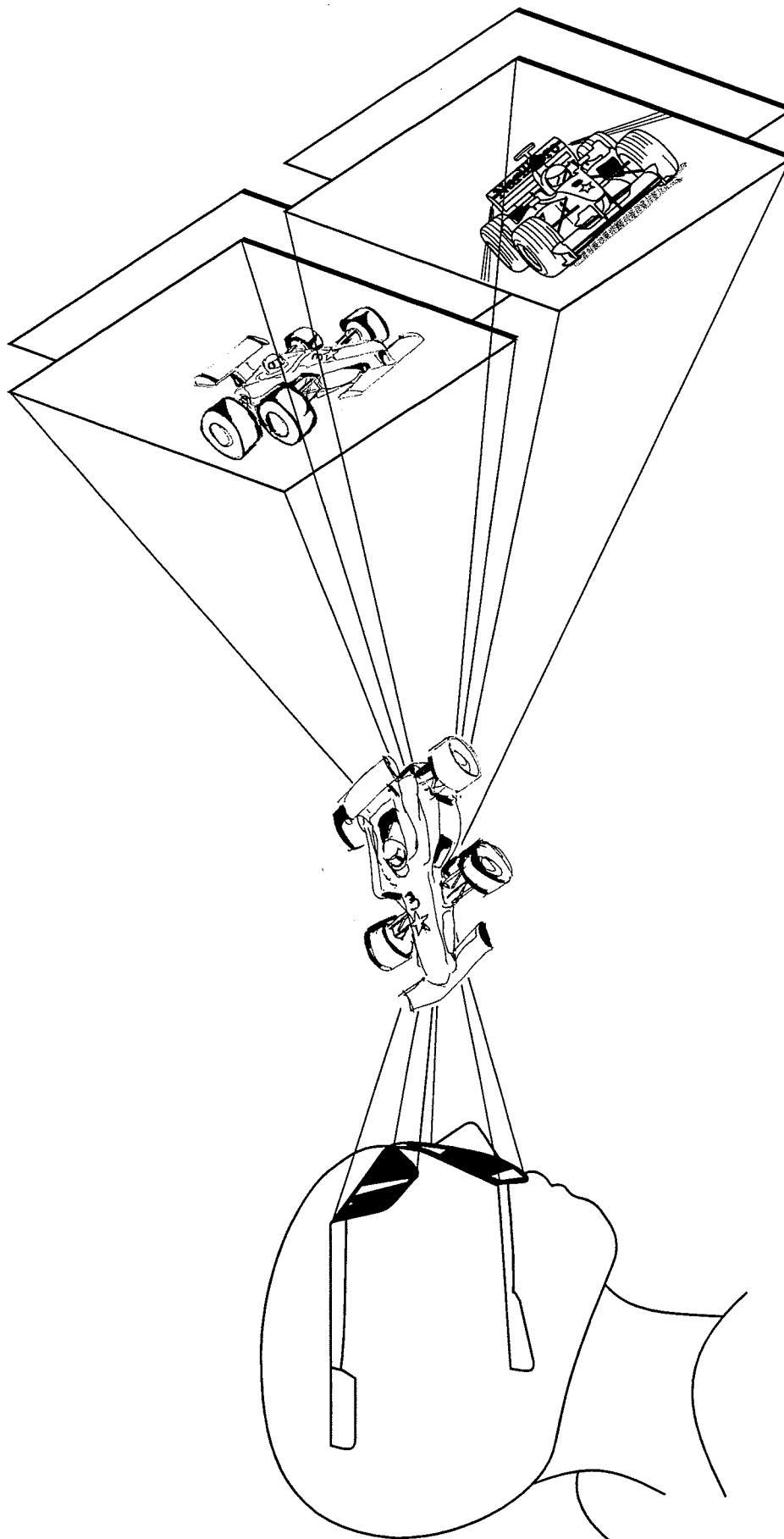


FIG. 8A

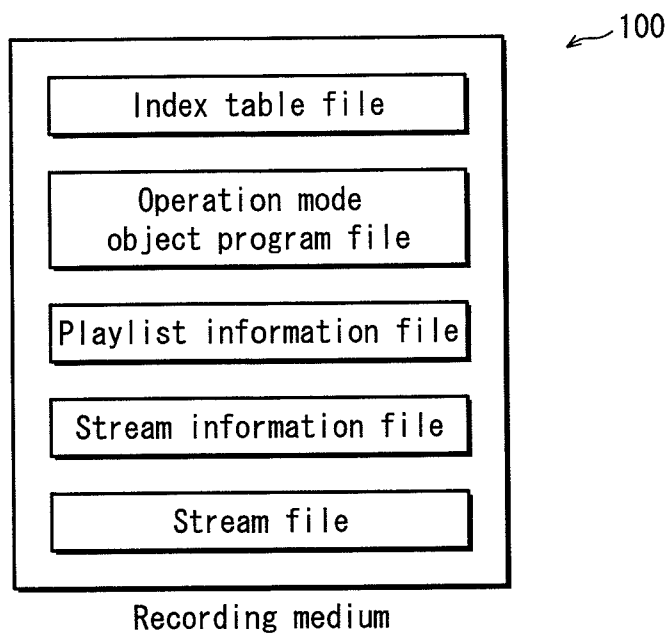


FIG. 8B

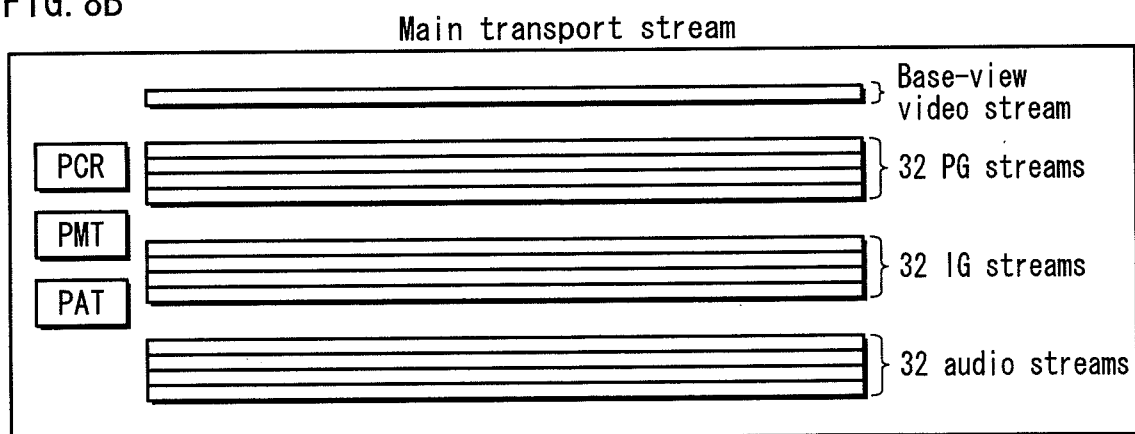
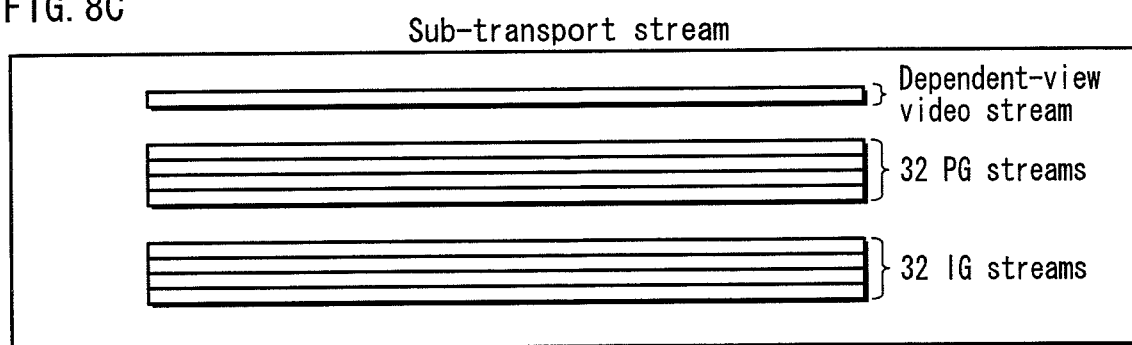


FIG. 8C



The diagram illustrates the mapping of video frames to PES packets and TS packets across three rows:

- 1st row: Video frame seq.** A sequence of video frames: I-picture, P-picture, B-picture, B-picture, P-picture, B-picture, and an ellipsis.
- 2nd row: PES packet seq.** A sequence of PES packets. The first packet contains an I-picture and is labeled 'yg1'. The second packet contains a P-picture and is labeled 'yg2'. The third packet contains a B-picture and is labeled 'yg3'. The fourth packet contains a B-picture and is labeled 'yg4'. The sequence continues with an ellipsis. A bracket indicates that the first packet is the 'PES header' and the rest is the 'PES payload'.
- 3rd row: TS packet seq.** A sequence of TS packets, each corresponding to a PES packet in the row above.

Arrows show the mapping from video frames to PES packets: the first I-picture frame maps to the first PES packet, the first P-picture frame maps to the second PES packet, the first B-picture frame maps to the third PES packet, and the second B-picture frame maps to the fourth PES packet.

1st row:
TS packet

TS header (4Byte)	TS payload (184Byte)
----------------------	-------------------------

2nd row:
Source packet

TP_extra_header (4Byte)	TS packet (188Byte)
SPN=0	SPN=1
SPN=2	SPN=3

FIG.10

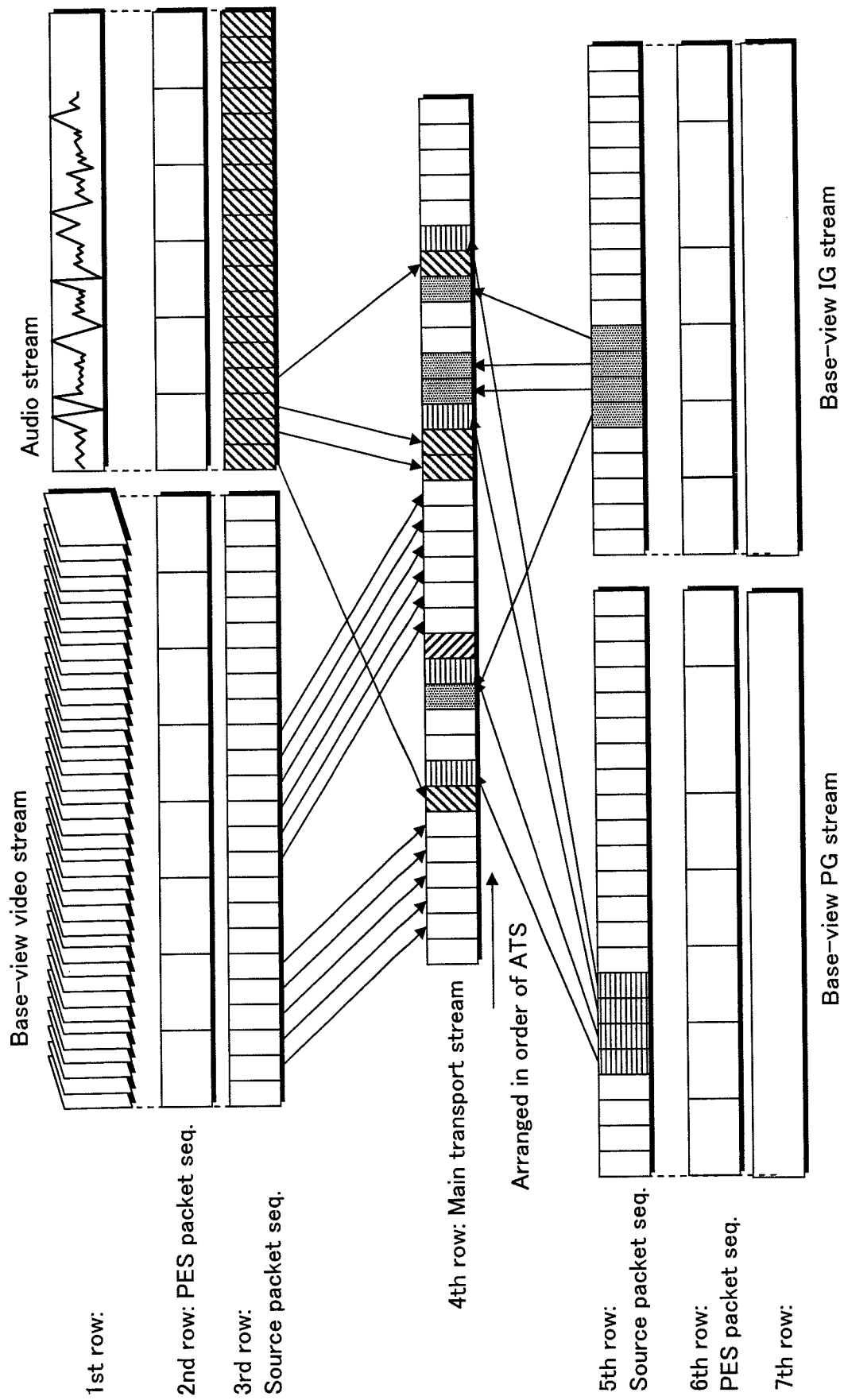


FIG. 11A

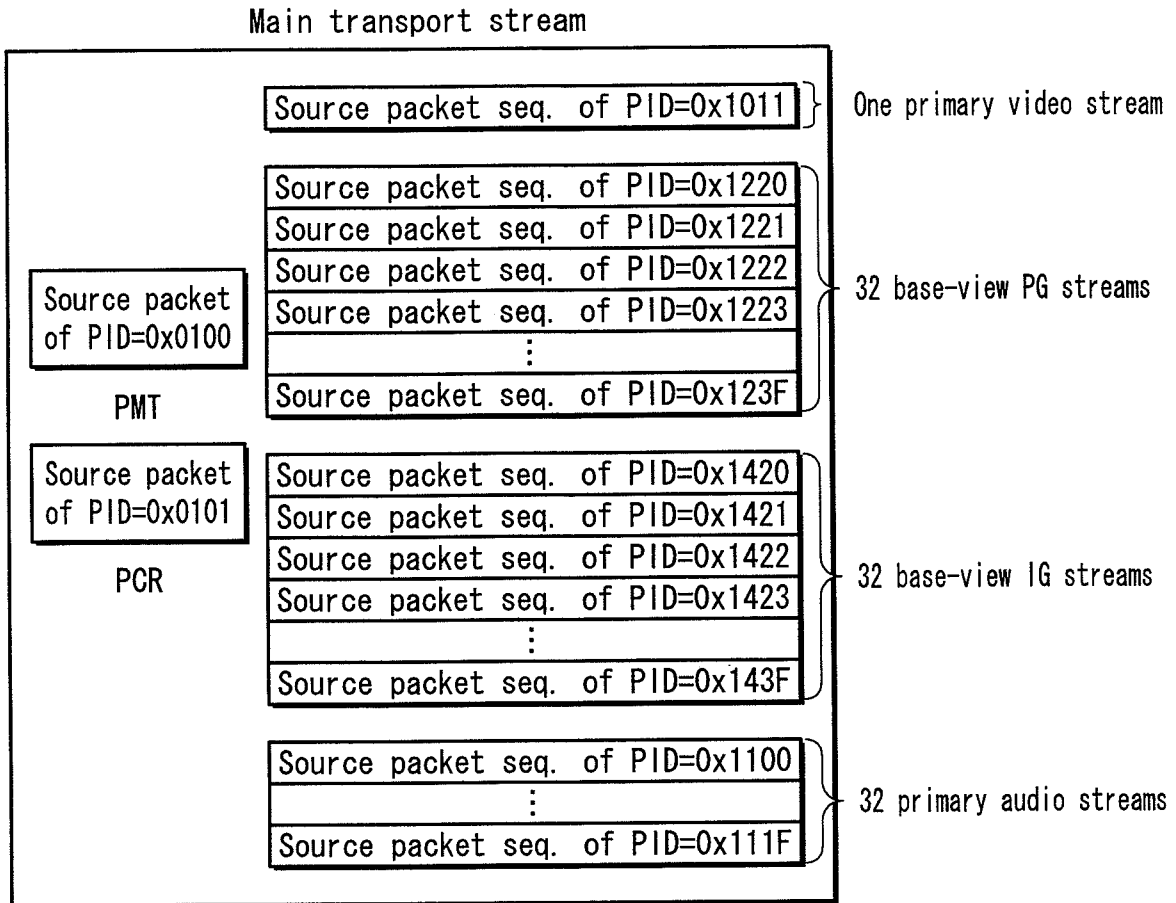


FIG. 11B

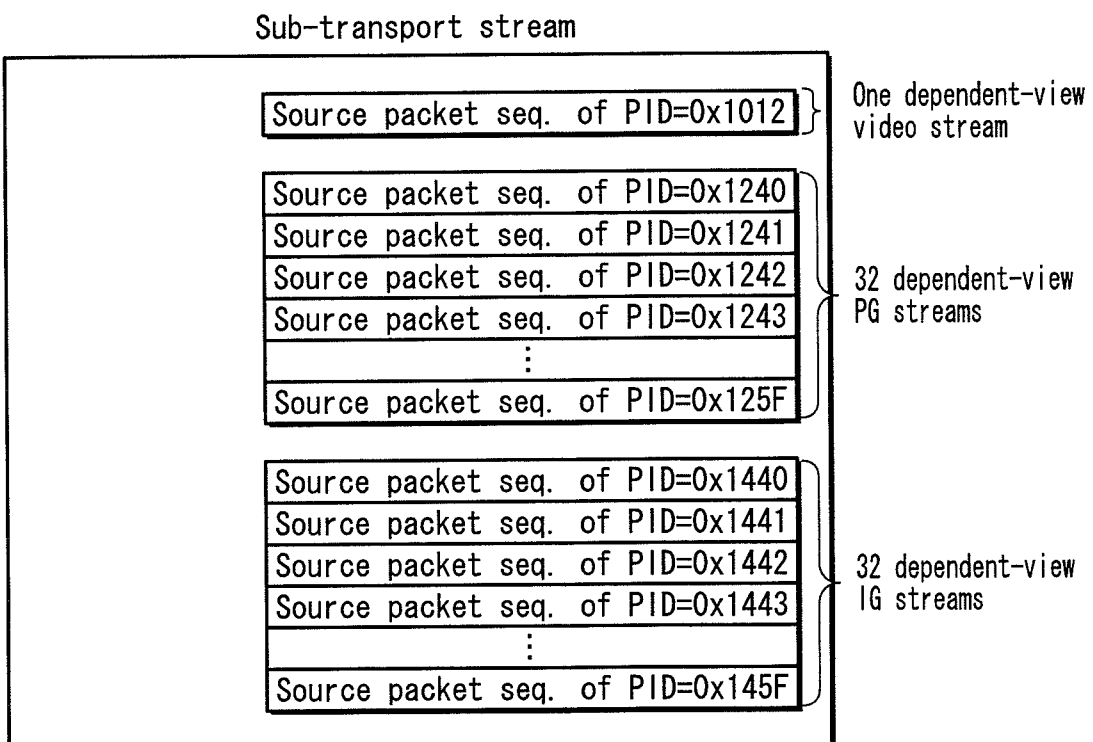


FIG. 12A

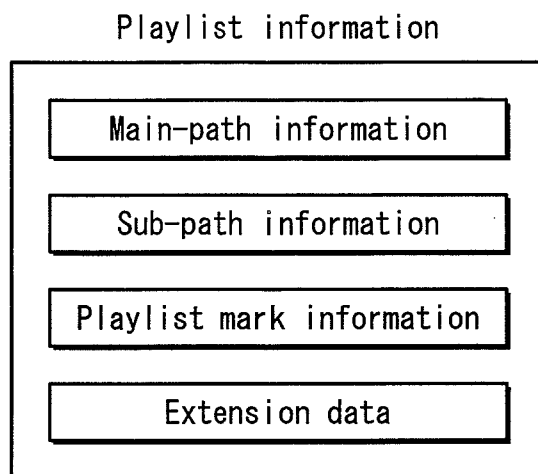


FIG. 12B

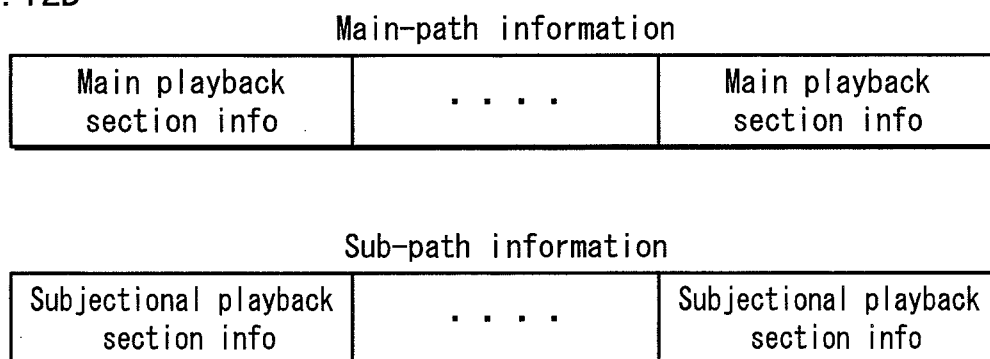


FIG. 12C

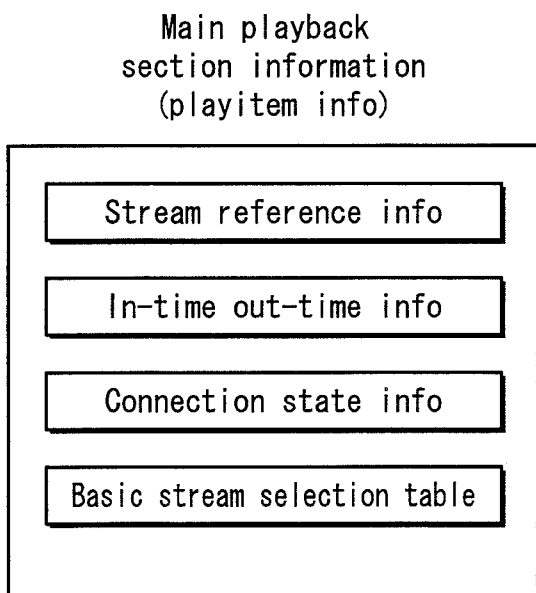


FIG. 12D

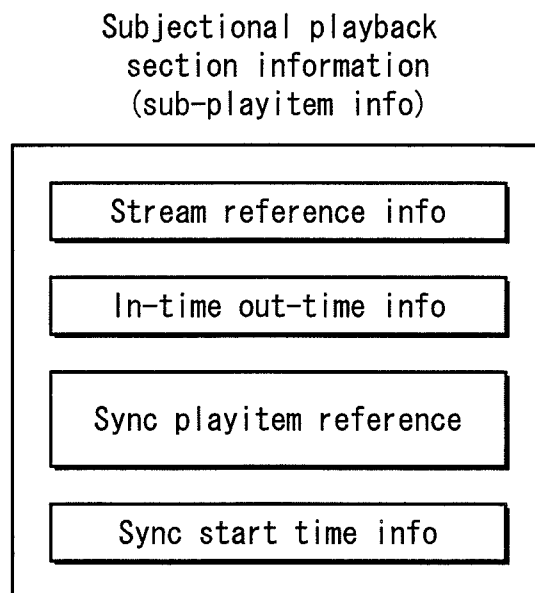


FIG. 13A

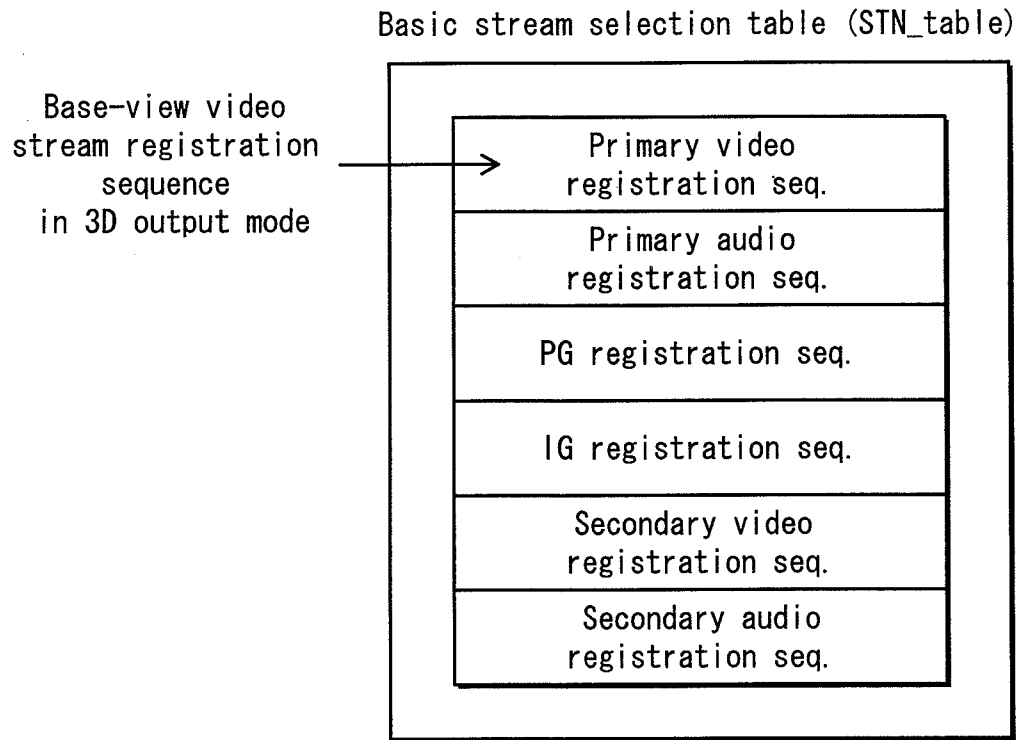


FIG. 13B

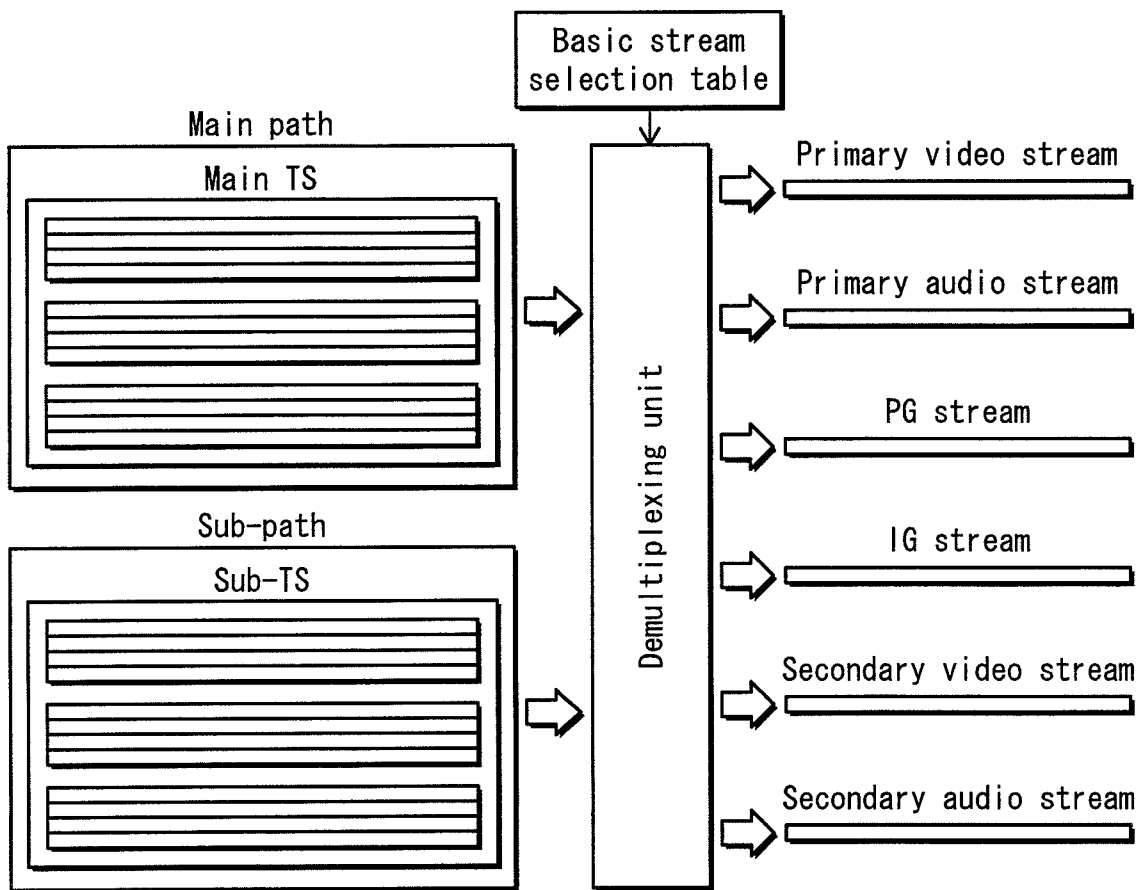


FIG. 14

Extension data

Extension stream selection table(STN_table_SS)	
Length	
Fixed_offset_during_Popup	
Dependent-view video stream registration seq. corresponding to playitem #1	Registration seq. corresponding to playitem #1
PG stream registration seq. corresponding to playitem #1	
IG stream registration seq. corresponding to playitem #1	
Dependent-view video stream registration seq. corresponding to playitem #2	Registration seq. corresponding to playitem #2
PG stream registration seq. corresponding to playitem #2	
IG stream registration seq. corresponding to playitem #2	
.	.
.	.
.	.
Dependent-view video stream registration seq. corresponding to playitem #N	Registration seq. corresponding to playitem #N
PG stream registration seq. corresponding to playitem #N	
IG stream registration seq. corresponding to playitem #N	

N represents the number of playitems in main path

FIG. 15A

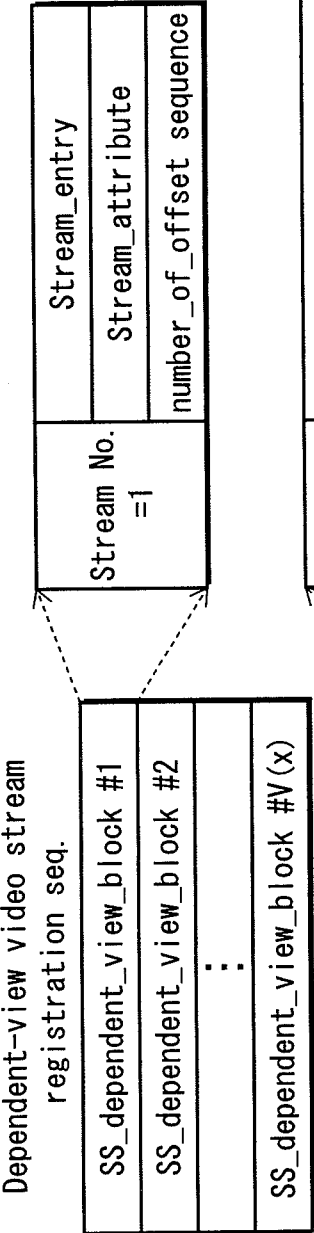


FIG. 15B

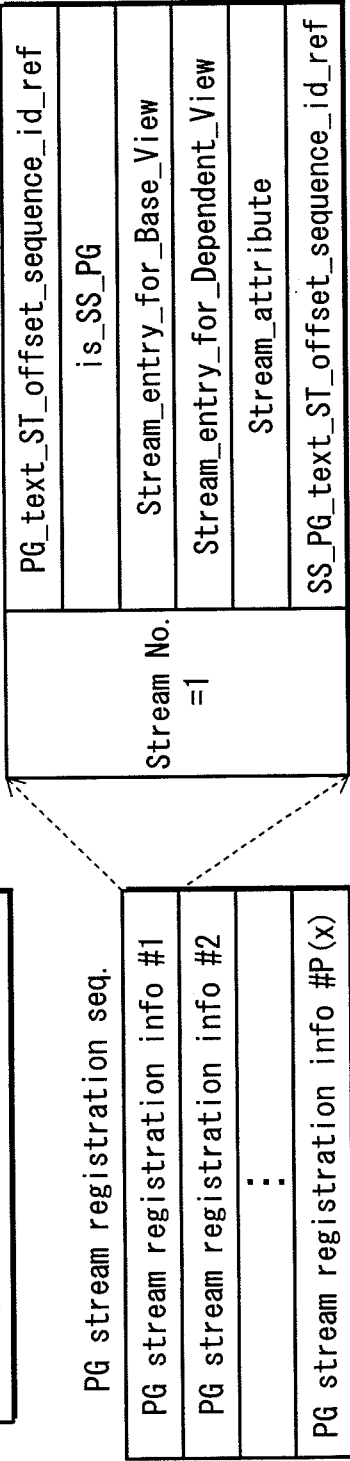


FIG. 15C

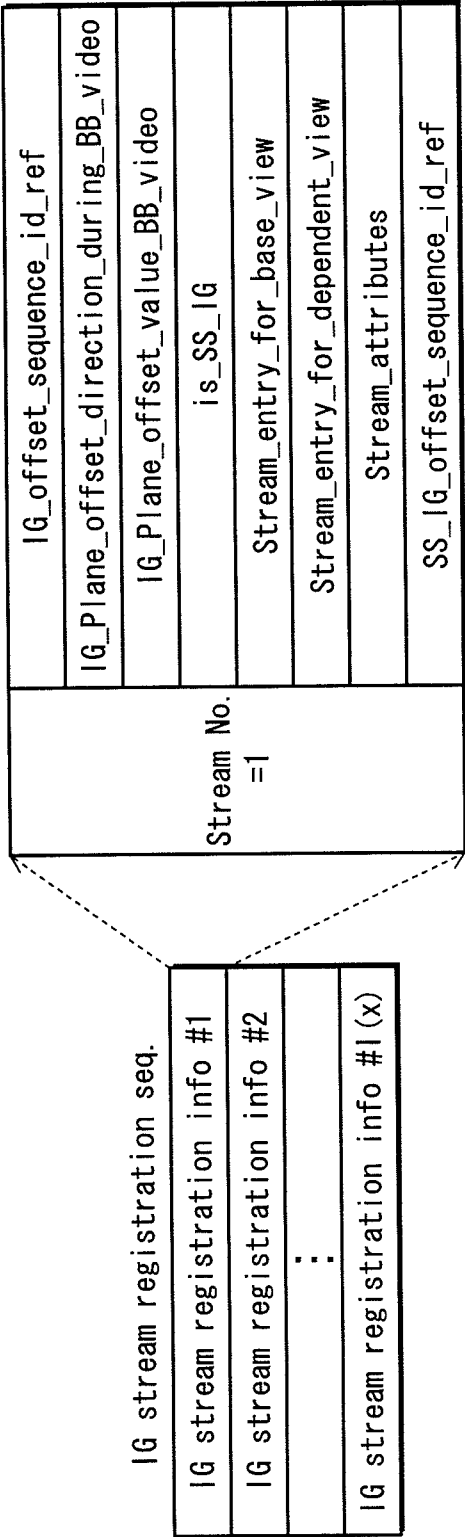


FIG. 16

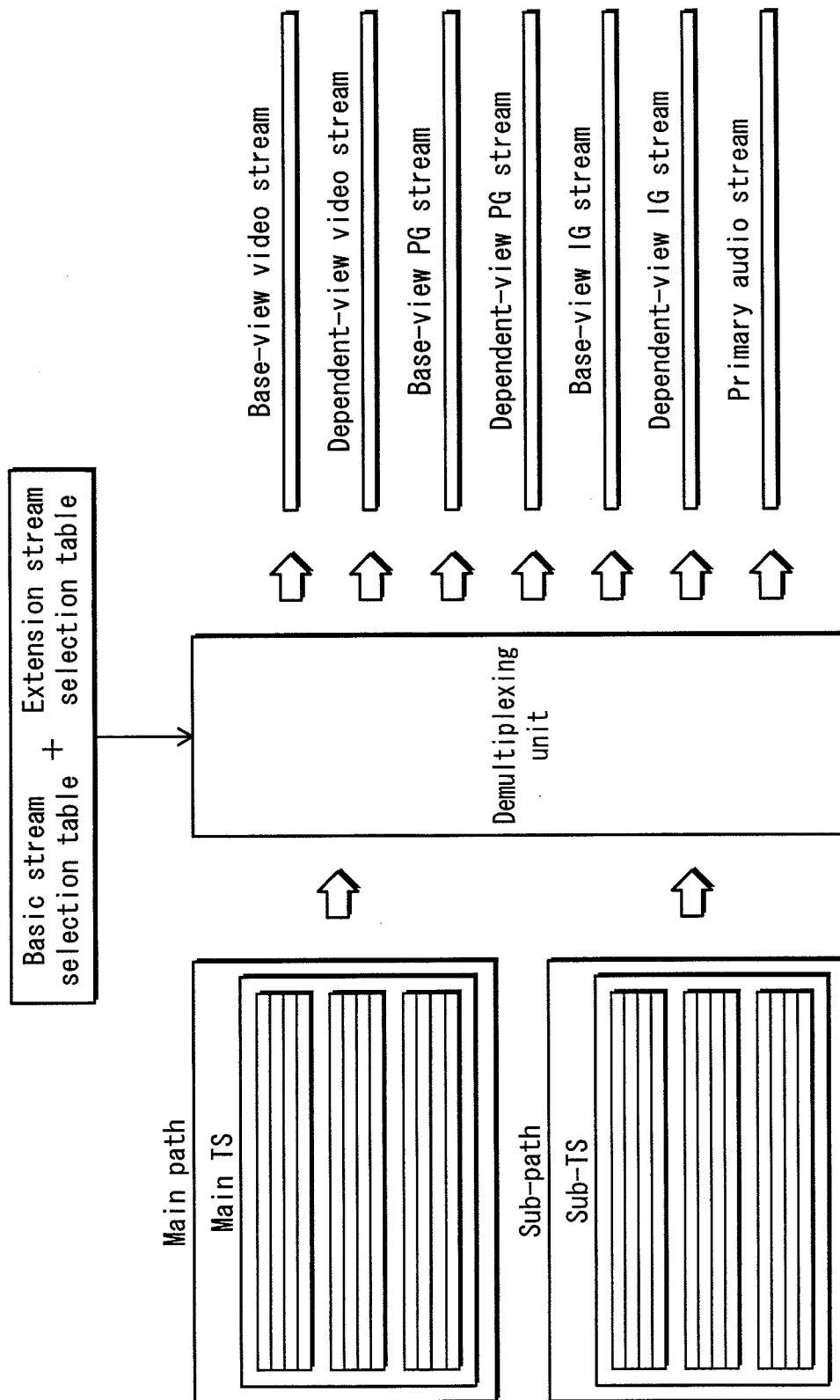


FIG. 17

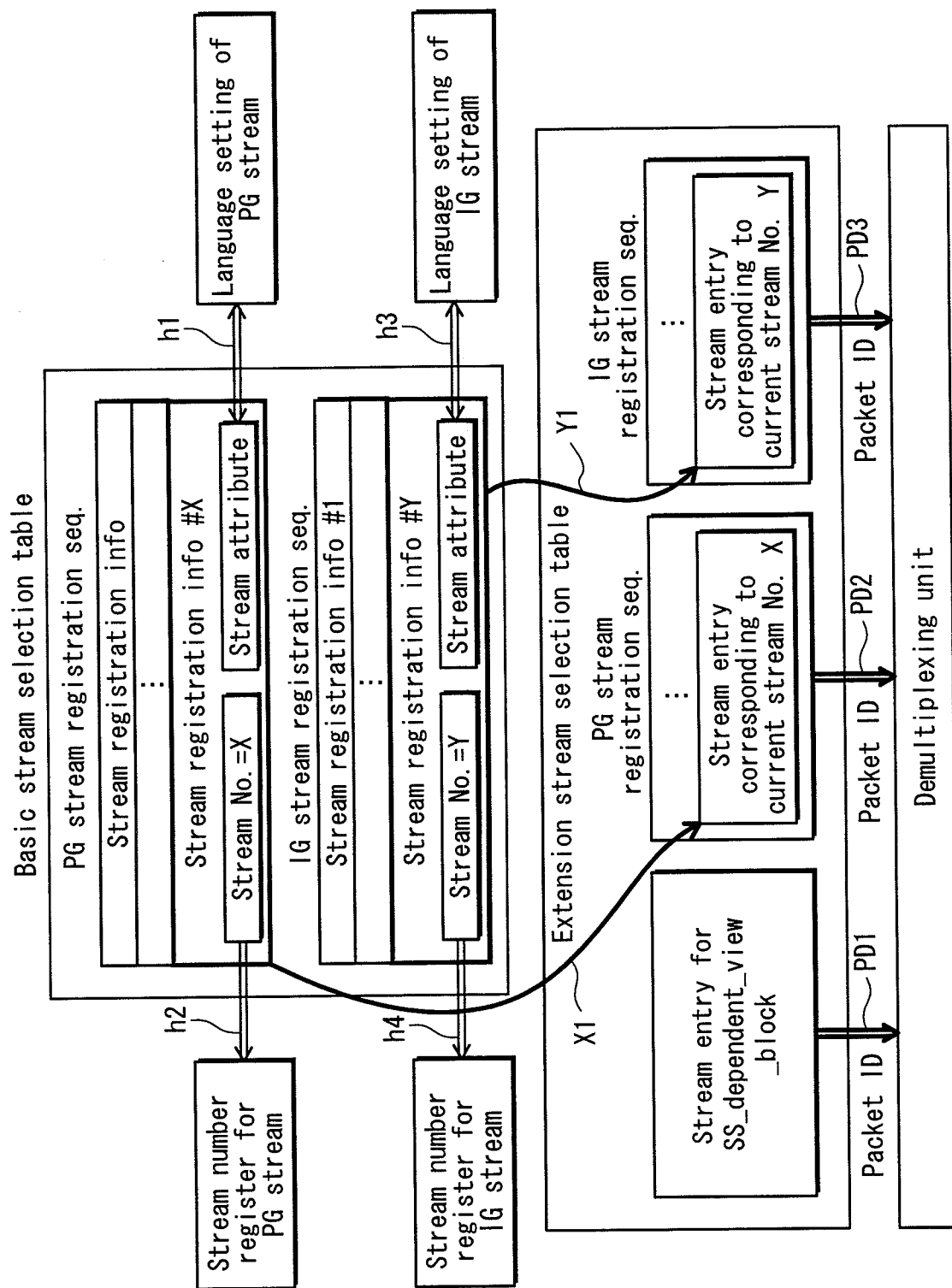


FIG.18

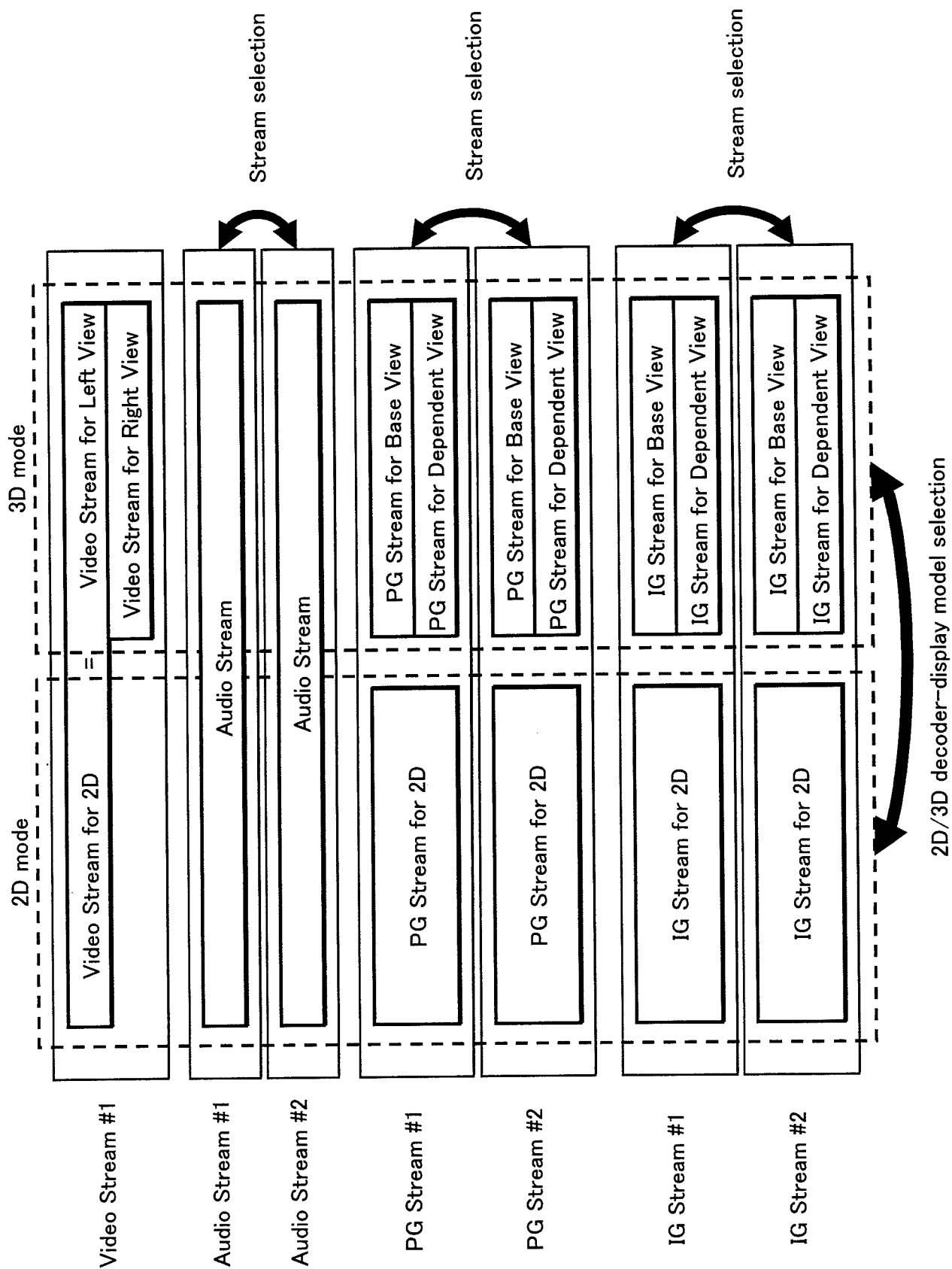


FIG. 19

STN_table_SS() {
for(PlayItem_id=0;PlayItem_id<number_of_PlayItems;PlayItem_id++) {
length
Fixed_offset_during_PopUp
for(primary_video_stream_id=0;
primary_video_stream_id
<number_of_primary_video_stream_entries;
primary_video_stream_id++) {
SS_dependent_view_block() {
stream_entry()
stream_attributes()
number_of_offset_sequence
}
}
for(PG_textST_stream_id=0;
PG_textST_stream_id
<number_of_PG_textST_stream_entries;
PG_textST_stream_id++) {
PG_textST_offset_sequence_id_ref
is_SS_PG
if(is_SS_PG==1b) {
stream_entry_for_base_view()
stream_entry_for_dependent_view()
stream_attributes()
SS_PG_textST_offset_sequence_id_ref
}
}
for(IG_stream_id=0;
IG_stream_id<number_of_IG_stream_entries;
IG_stream_id++) {
IG_offset_sequence_id_ref
IG_Plane_offset_direction_during_BB_video
IG_Plane_offset_value_during_BB_video
is_SS_IG
if(is_SS_IG==1b) {
stream_entry_for_base_view()
stream_entry_for_dependent_view()
stream_attributes()
if(Fixed_offset_during_PopUp==1) {
}else{
SS_IG_offset_sequence_id_ref
}
}
}
}
}

FIG. 20

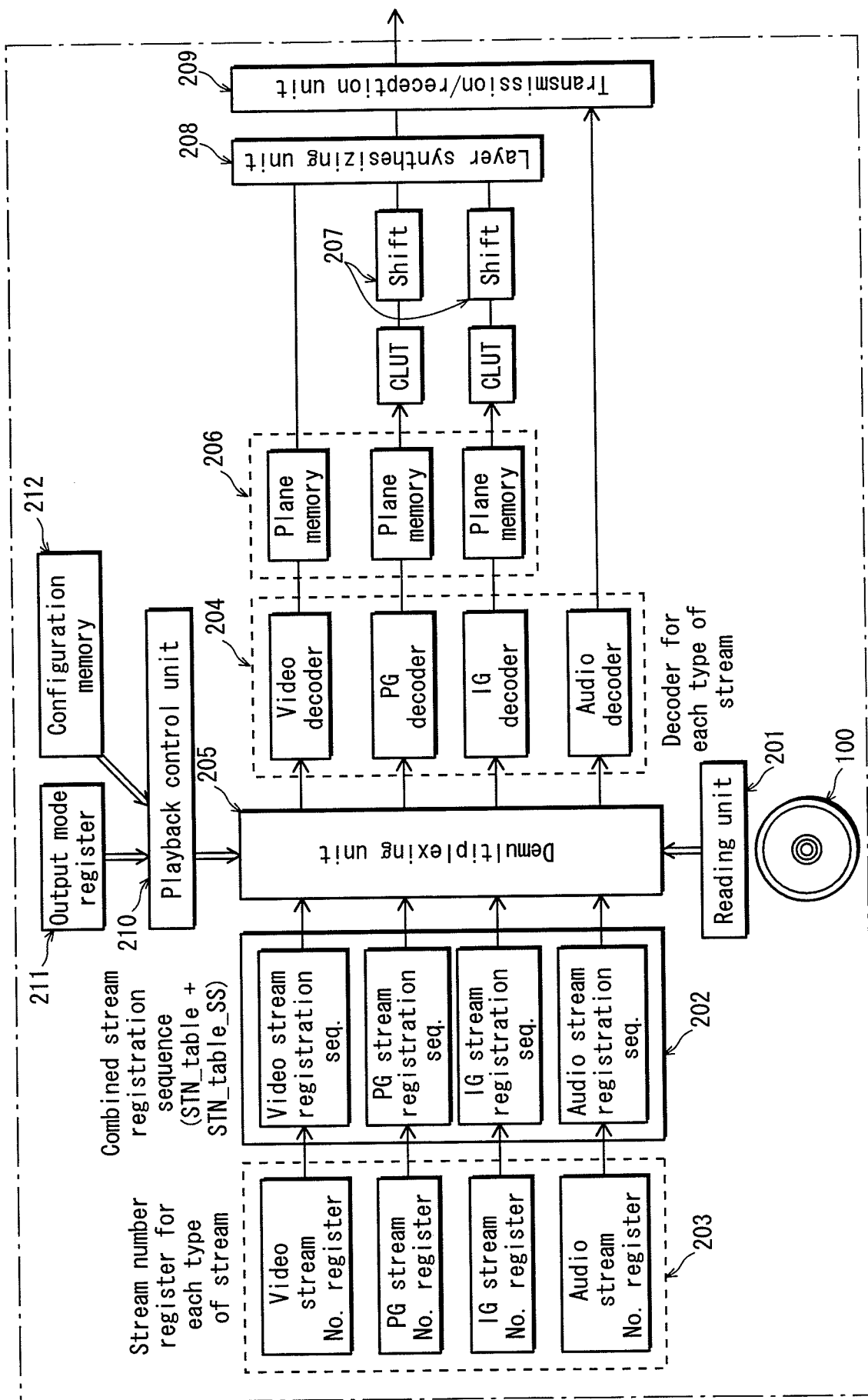
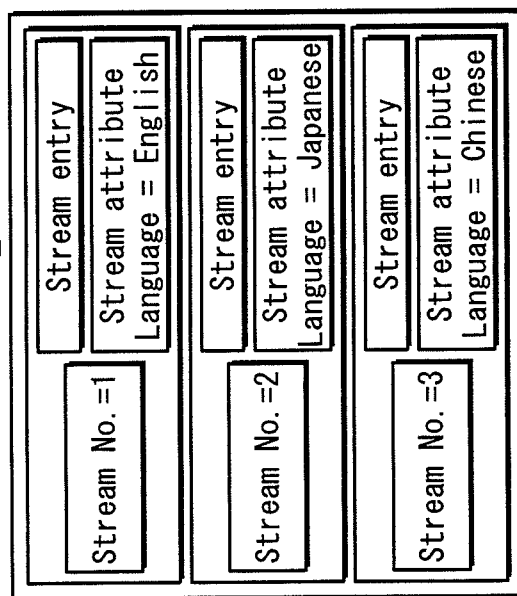


FIG. 21A Stream registration seq. of STN_table



Stream registration seq. of STN_table SS

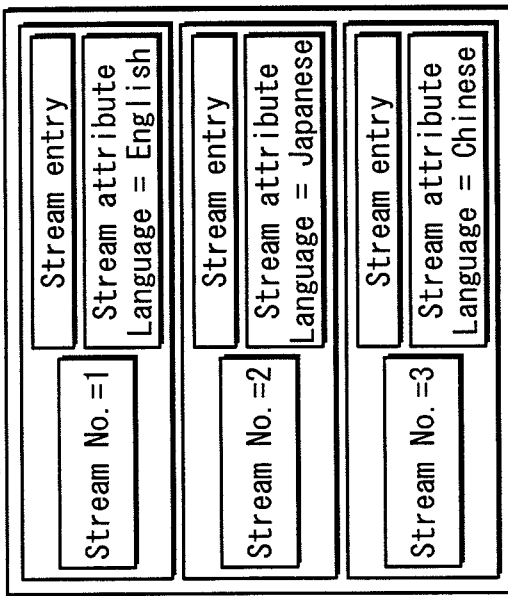


FIG. 21B

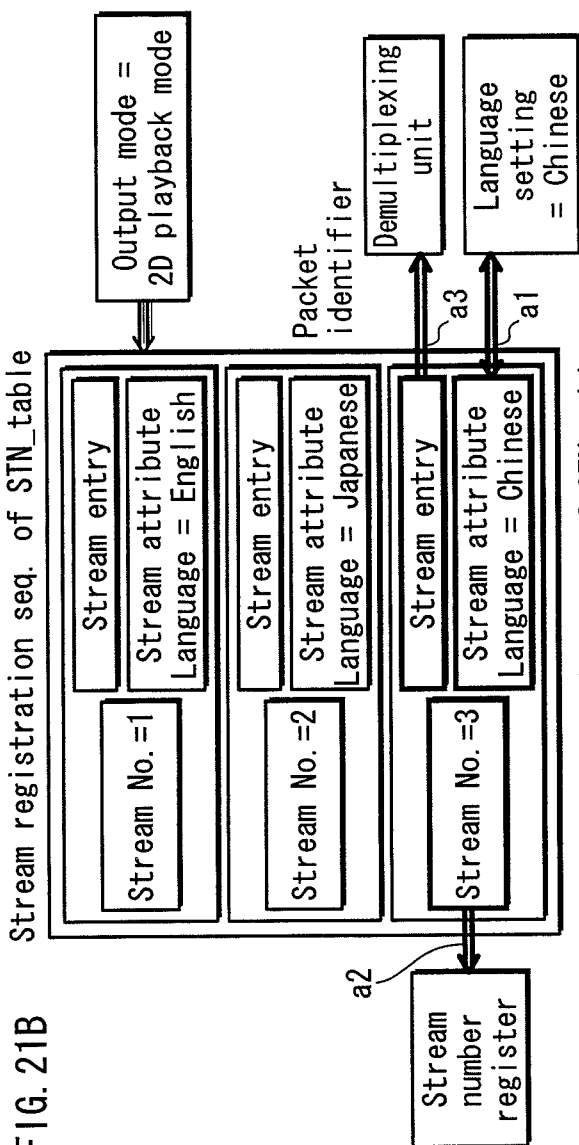


FIG. 21C

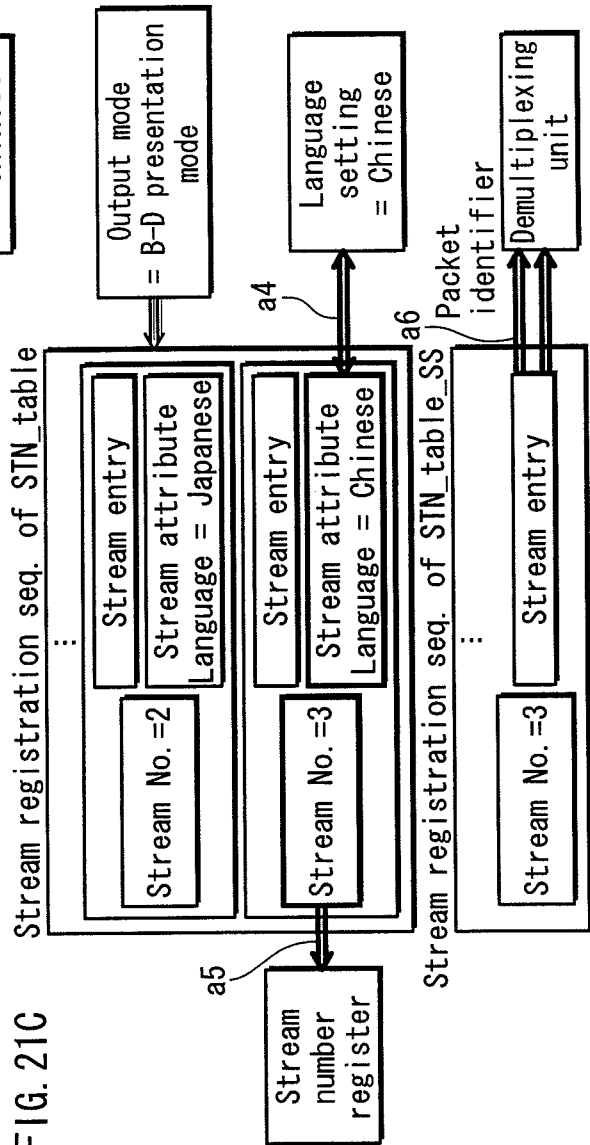


FIG. 22A Stream registration seq. of STN_table

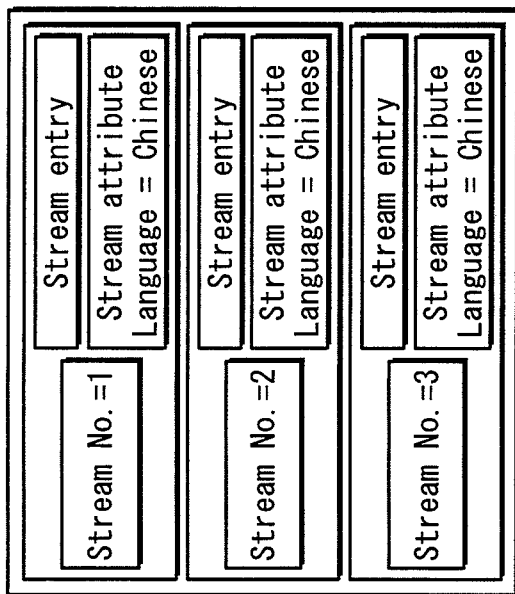


FIG. 22B

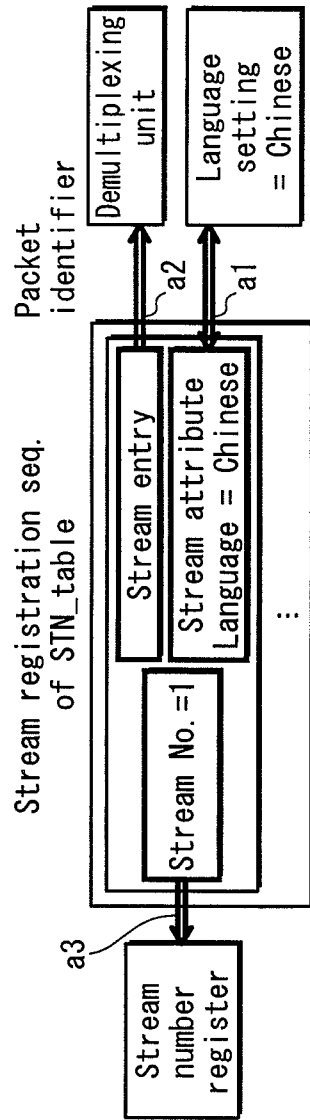


FIG. 22C

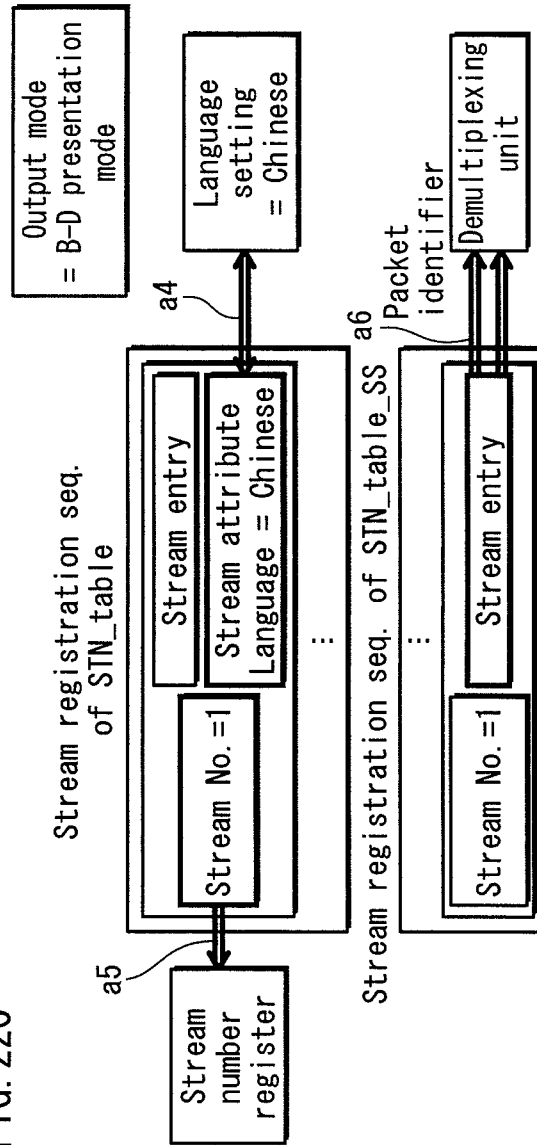
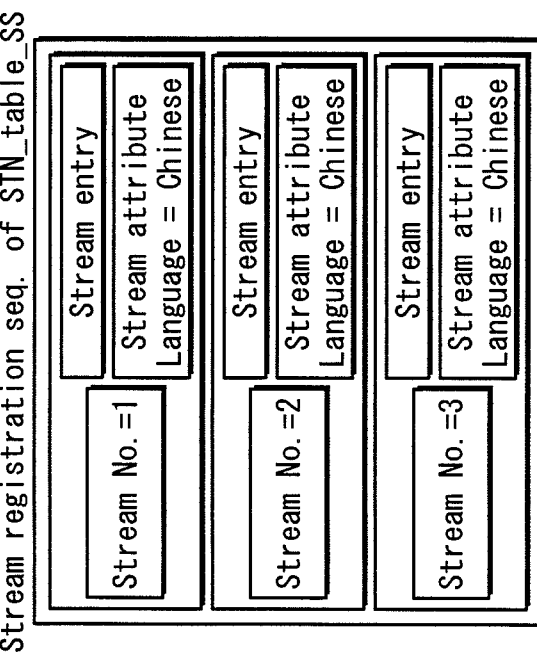


FIG. 23

Combined stream registration sequence
(STN_table + STN_table_SS)

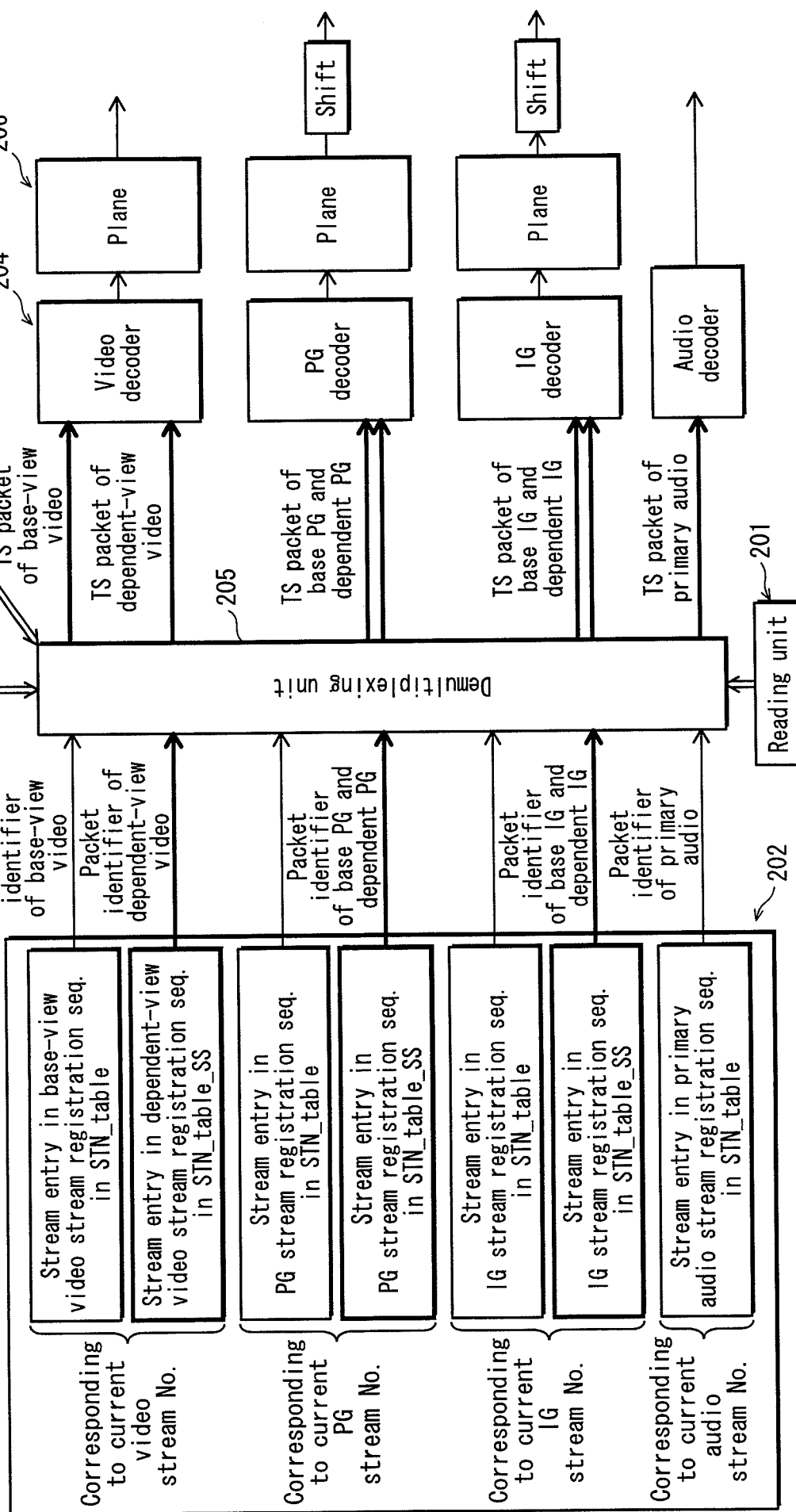


FIG. 24

Combined stream registration sequence
(STN_table + STN_table_SS)

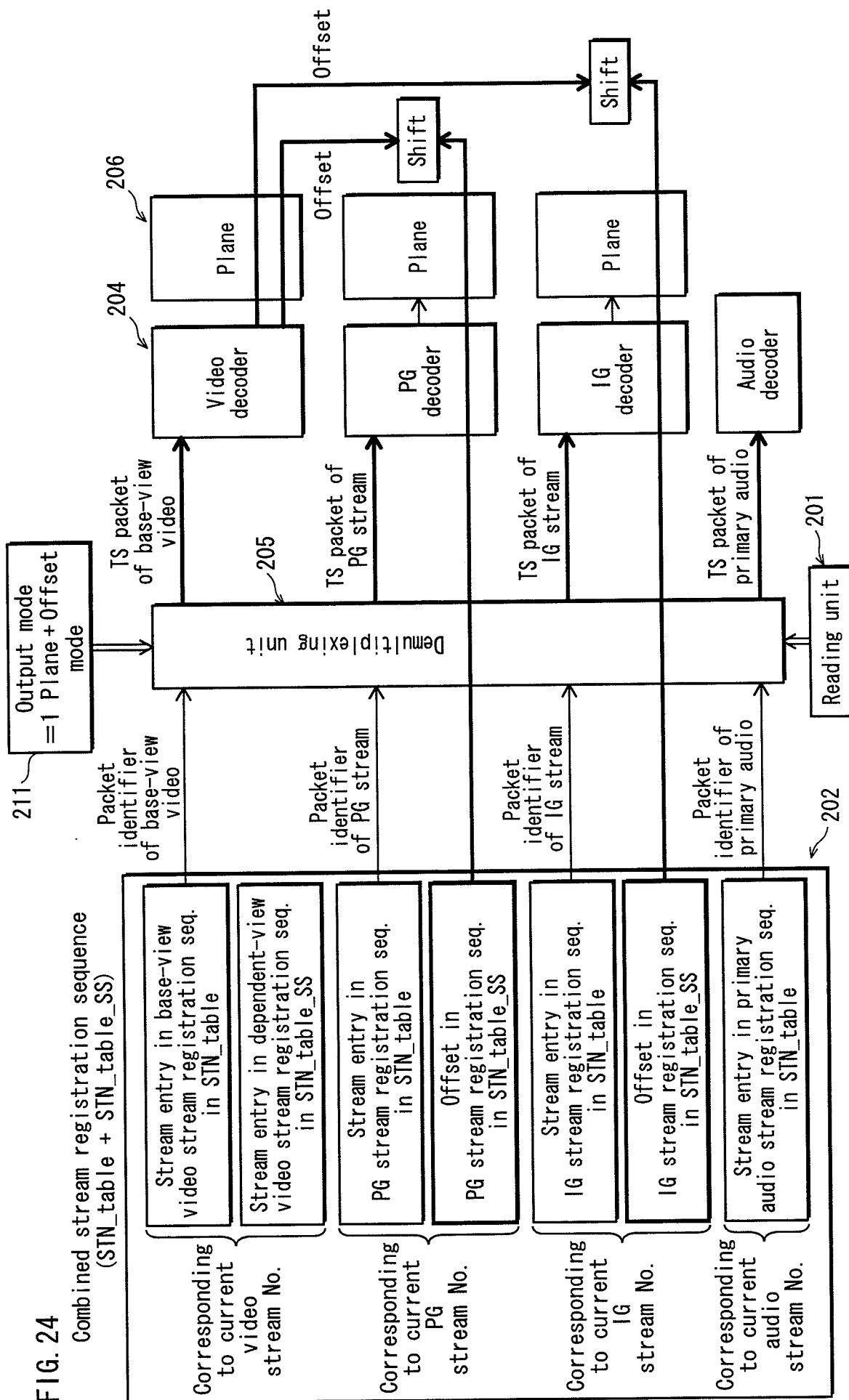


FIG. 25

Combined stream registration sequence
(STN_table + STN_table_SS)

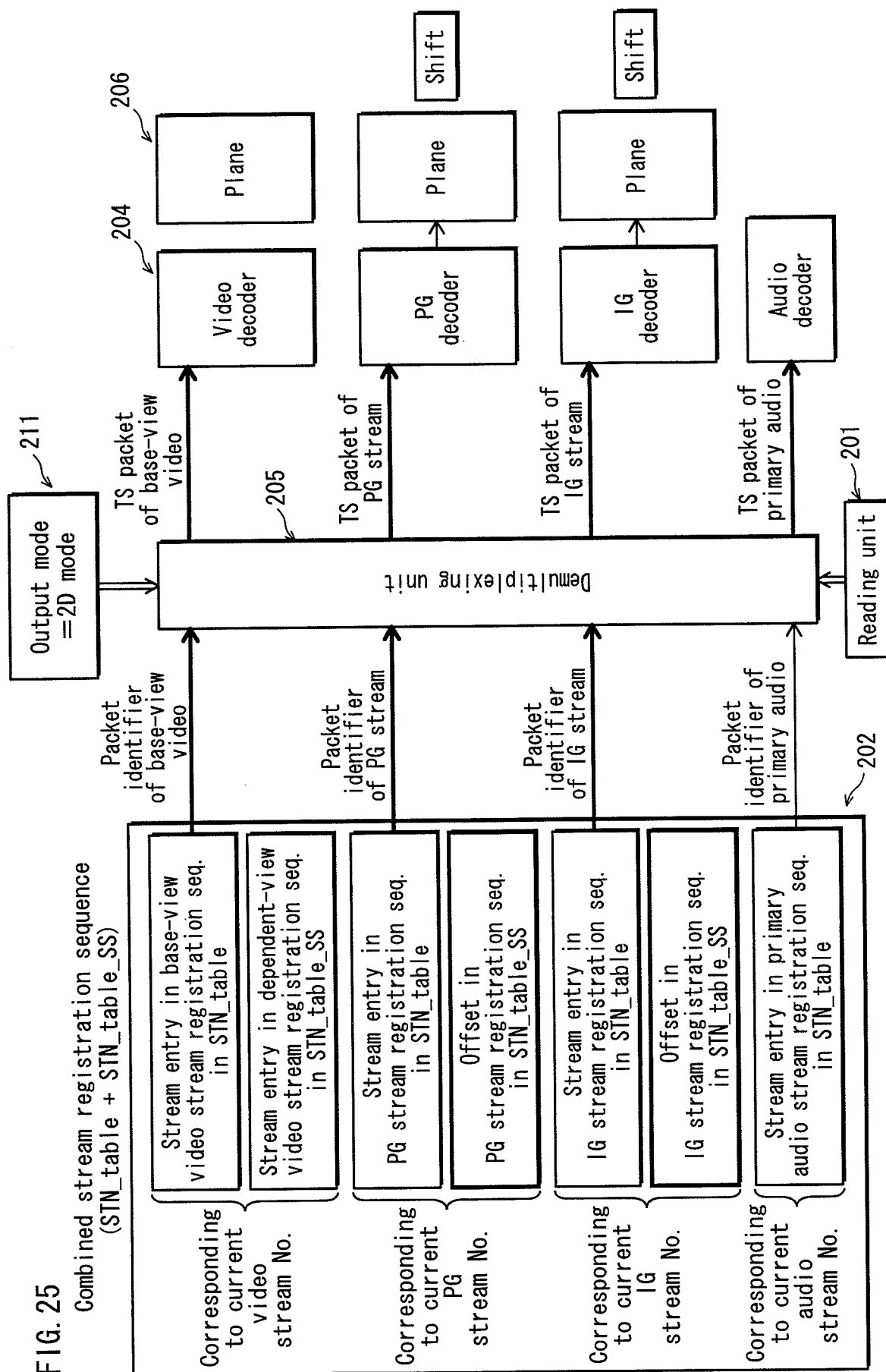


FIG. 26

Combined stream registration sequence
(STN_table + STN_table_SS)

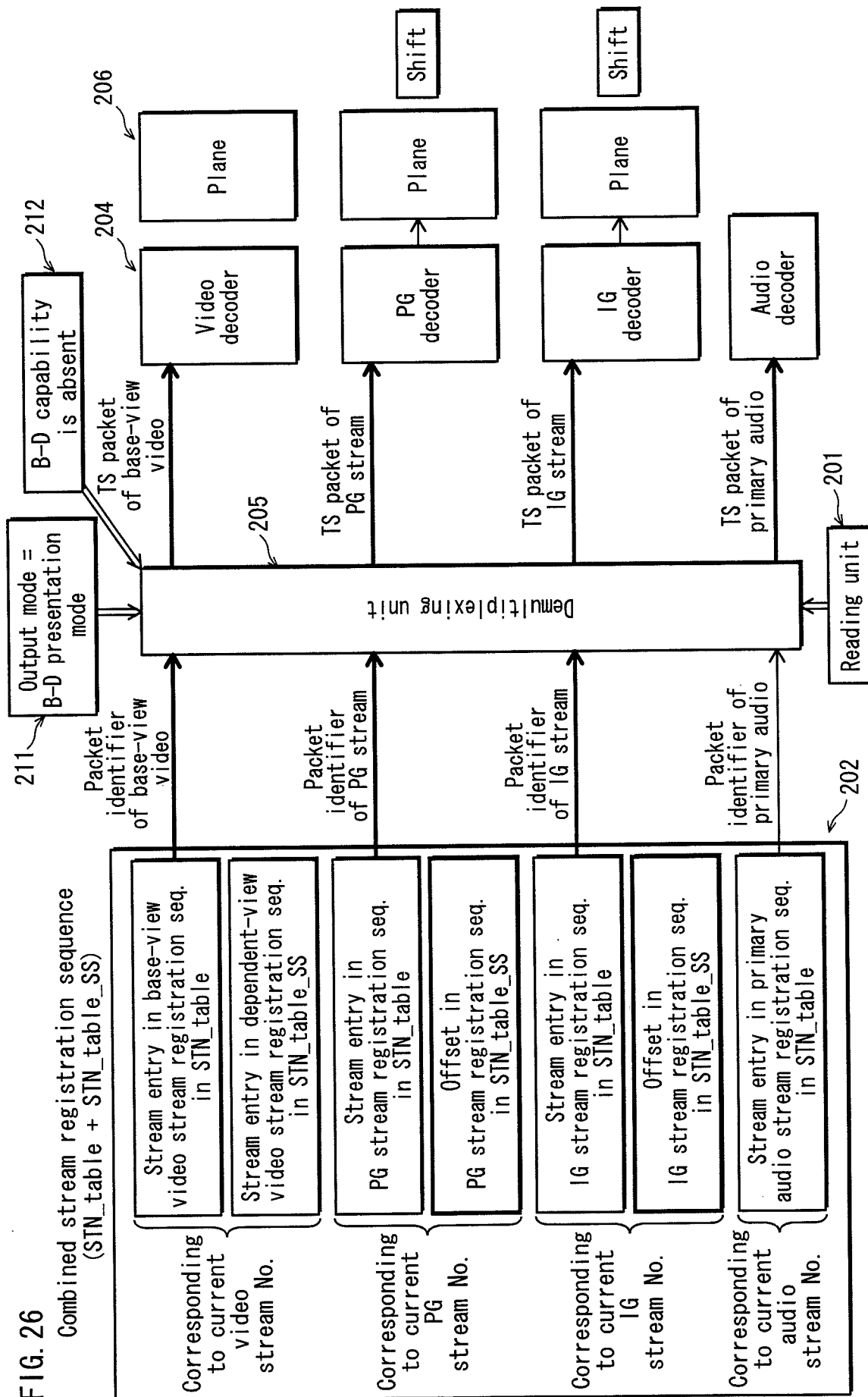


FIG. 27

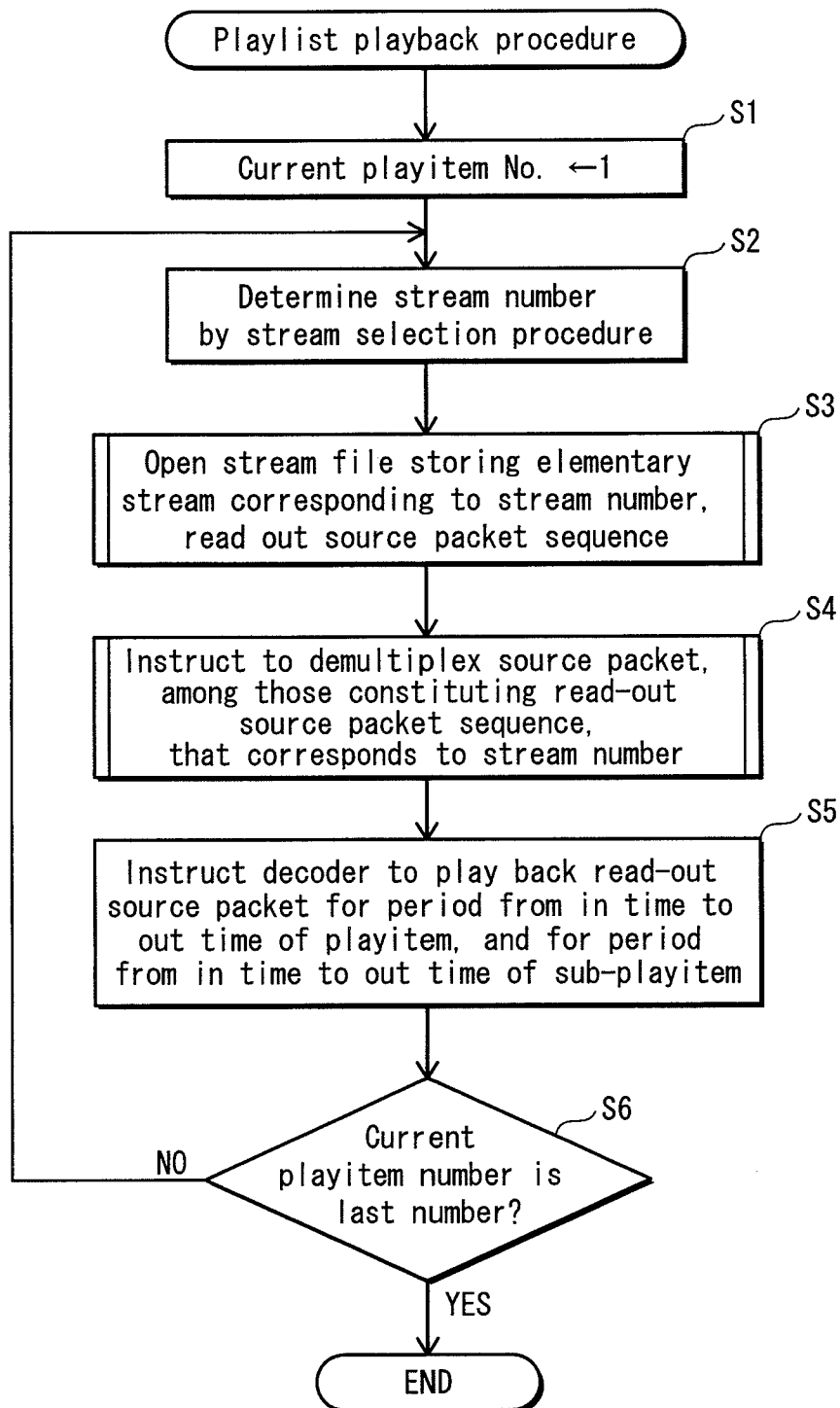


FIG. 28

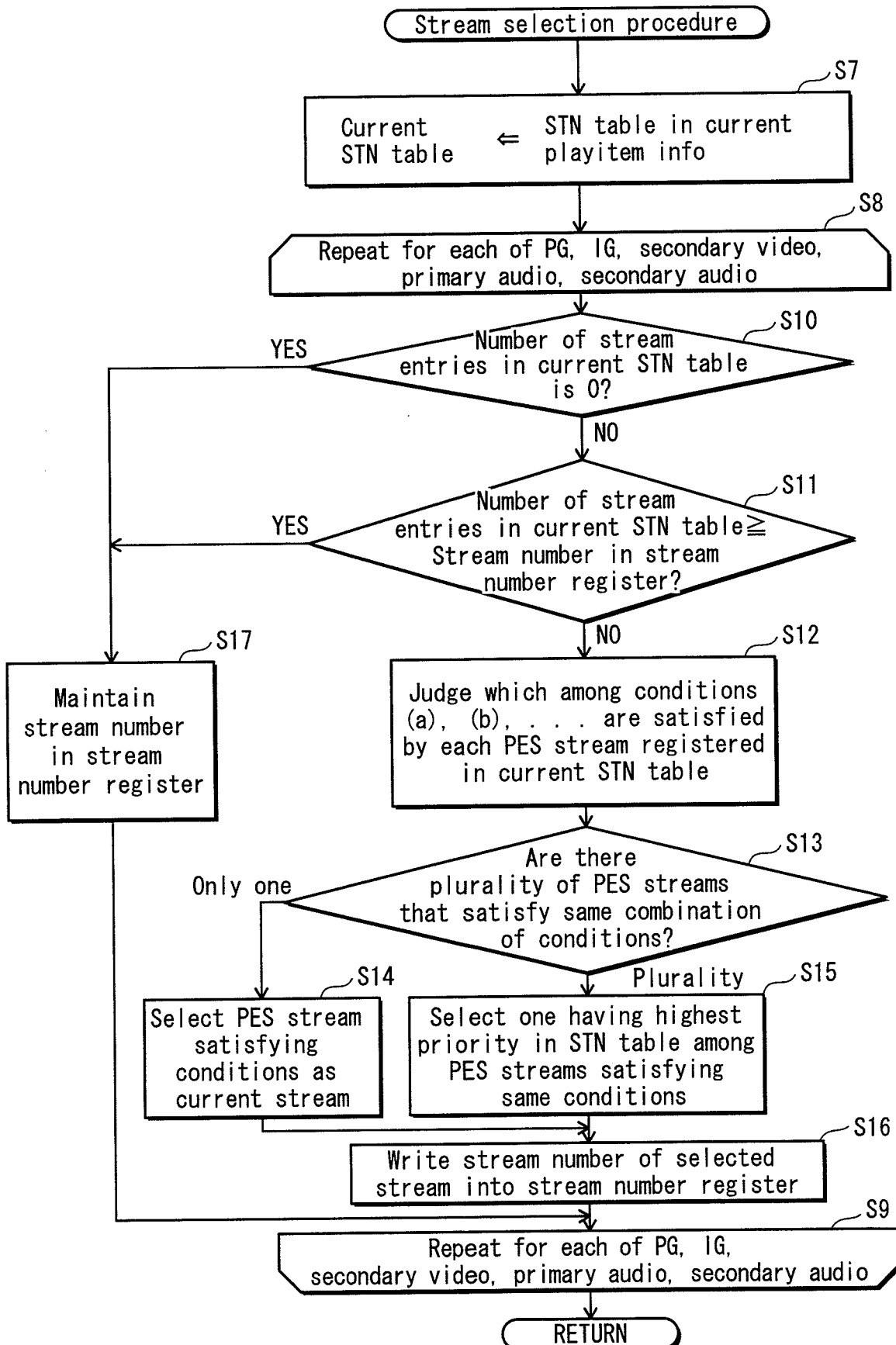


FIG. 29

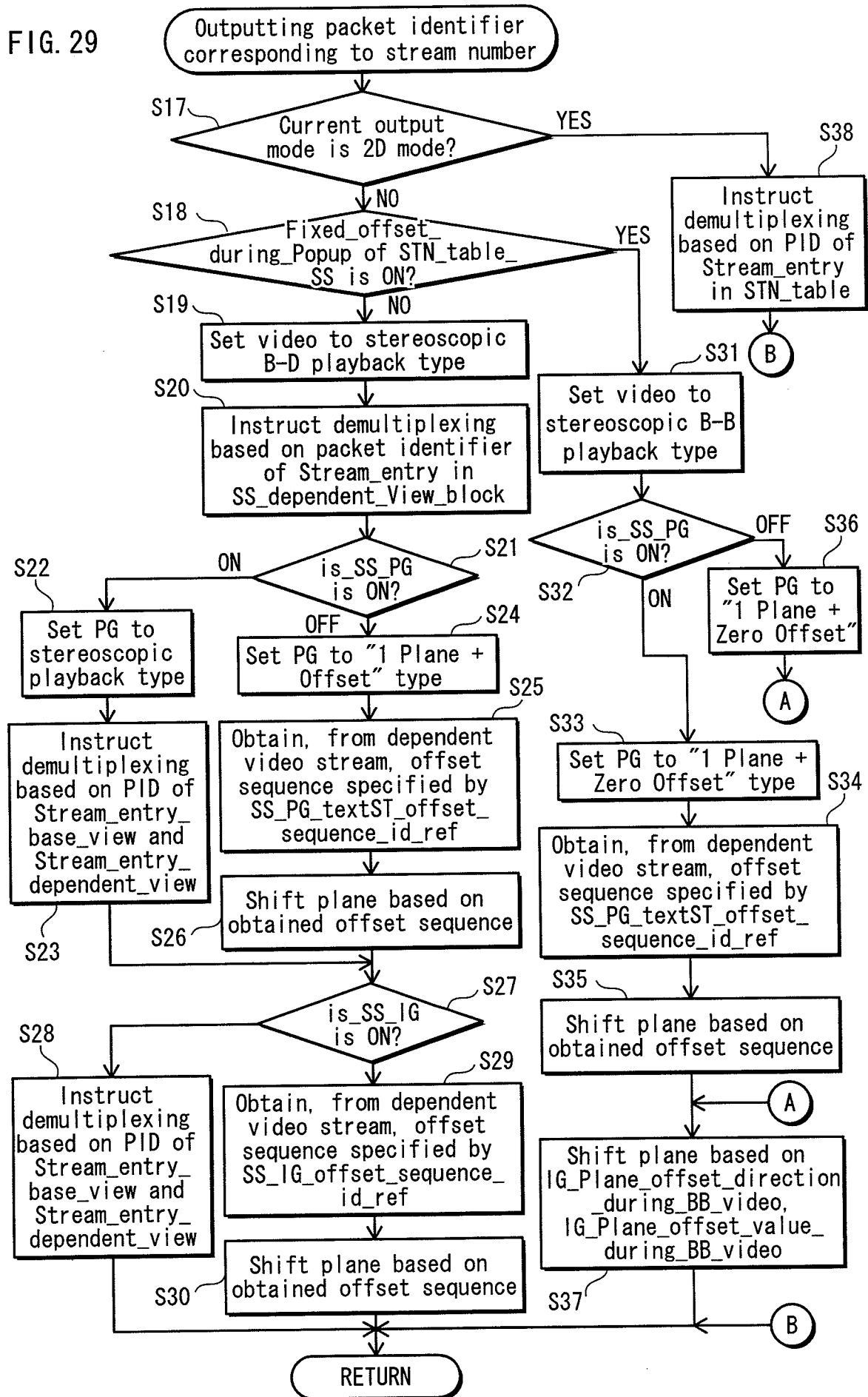


FIG. 30

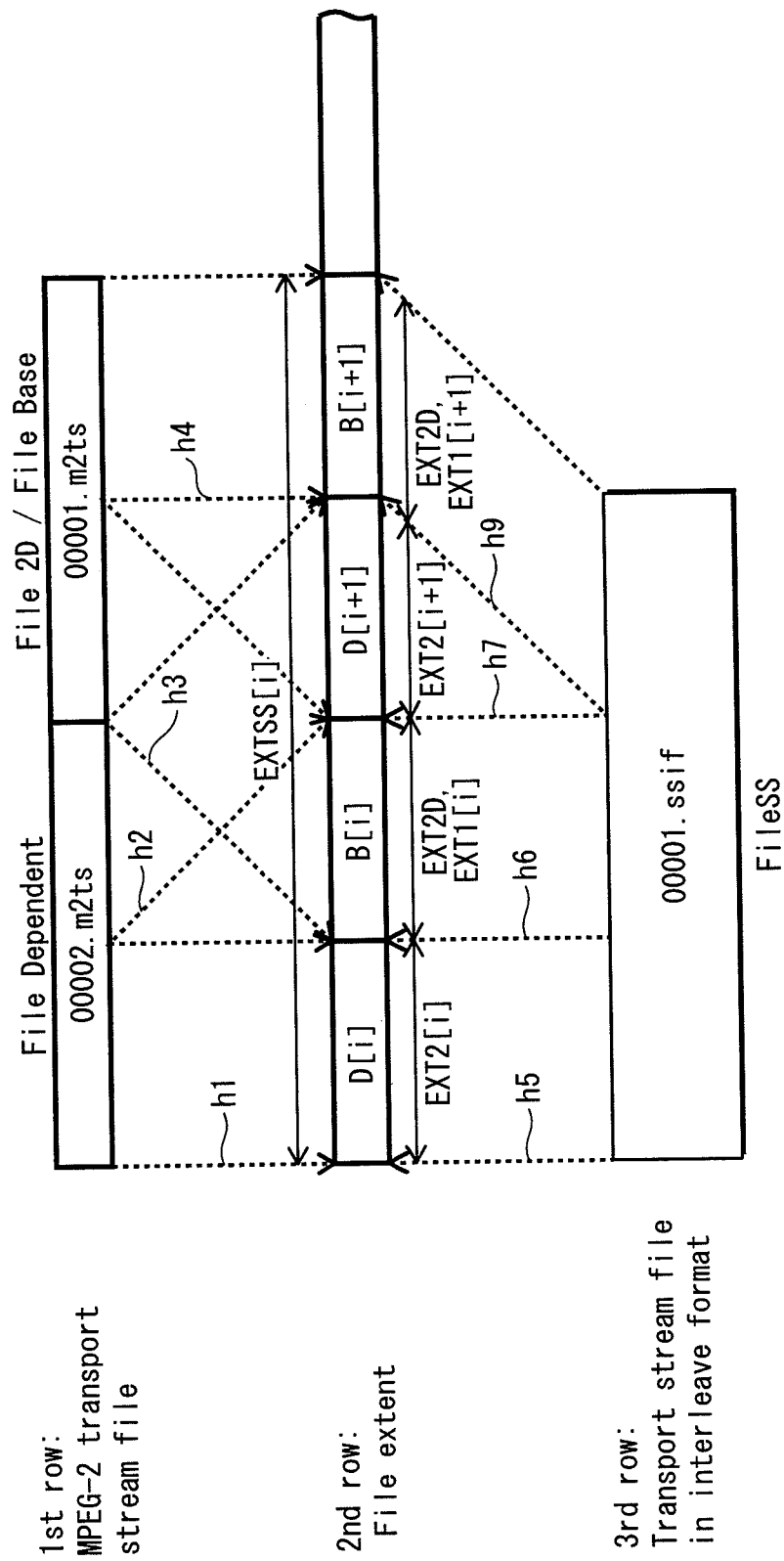


FIG. 31A

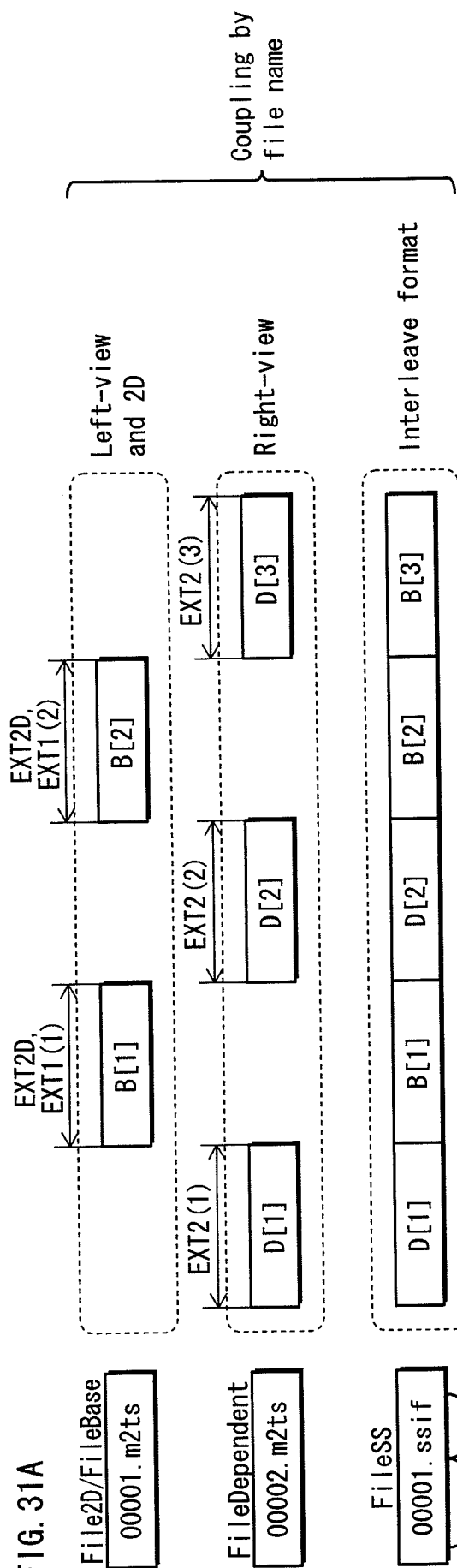


FIG. 31C

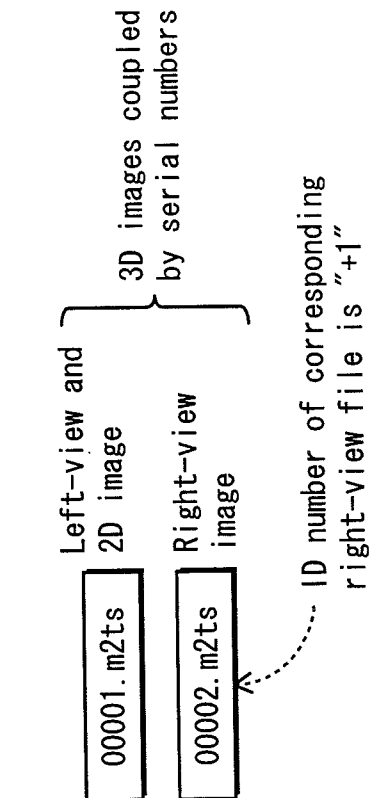


FIG. 31B

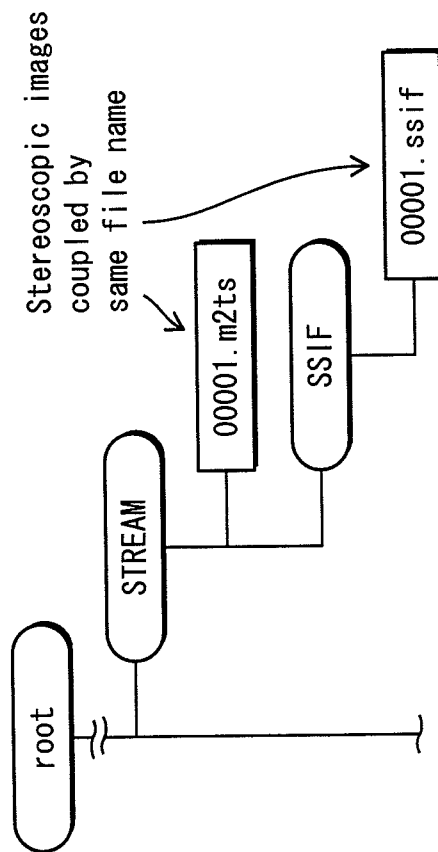


FIG. 32

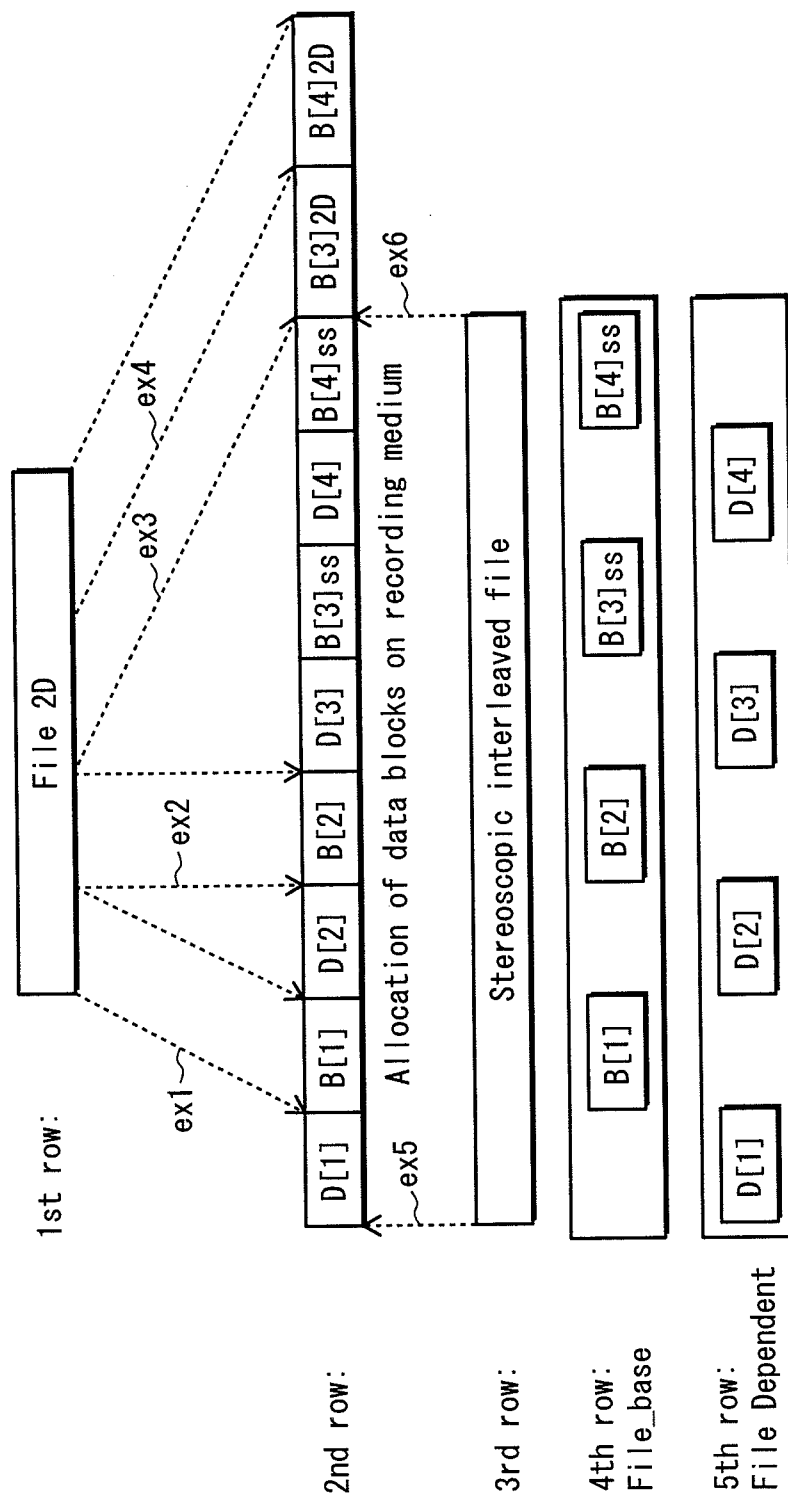


FIG.33

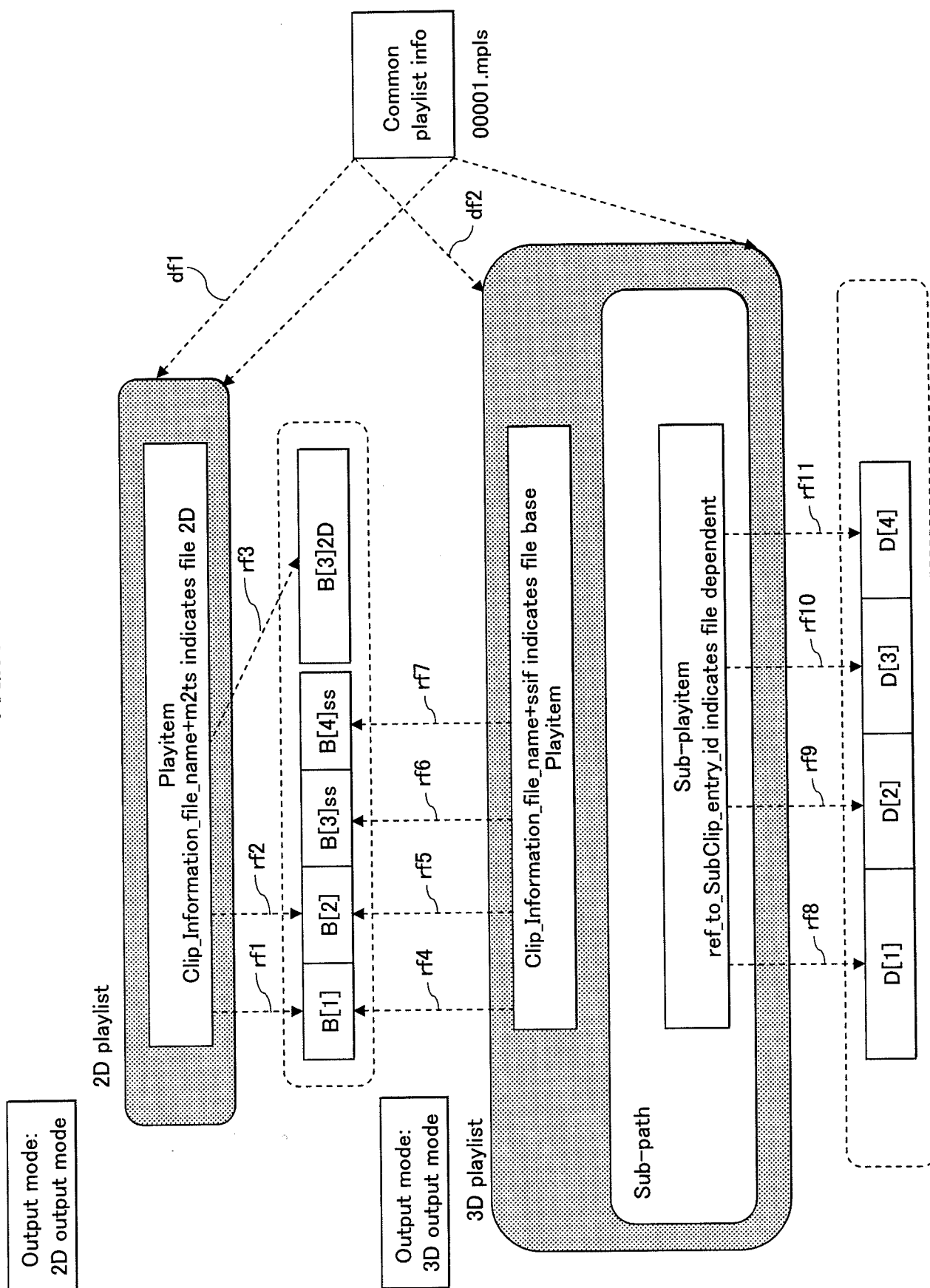
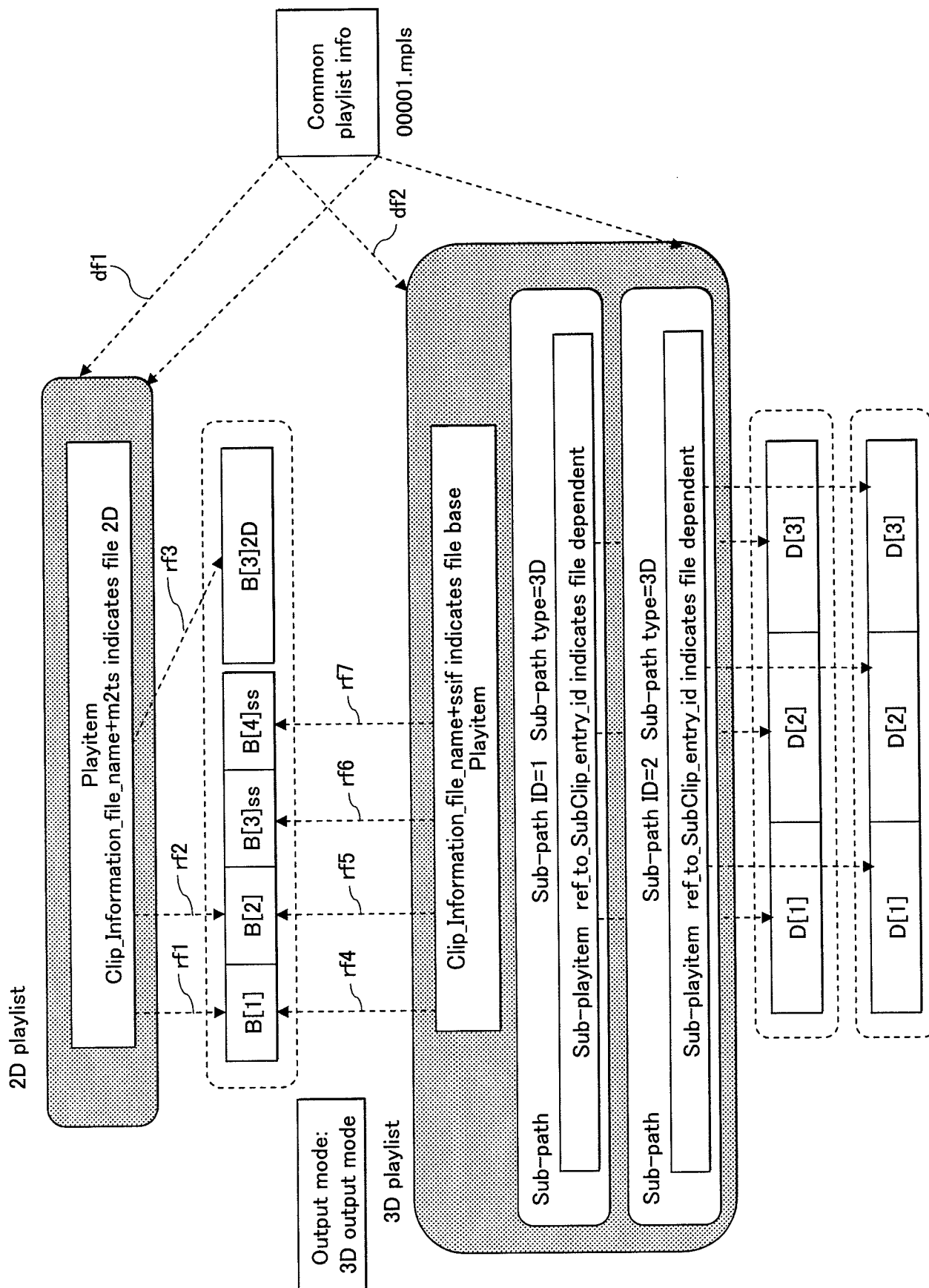


FIG.34



Set to 0 when stream referenced by main (playitem) is left view; and set to 1 when it is right view

FIG.35A

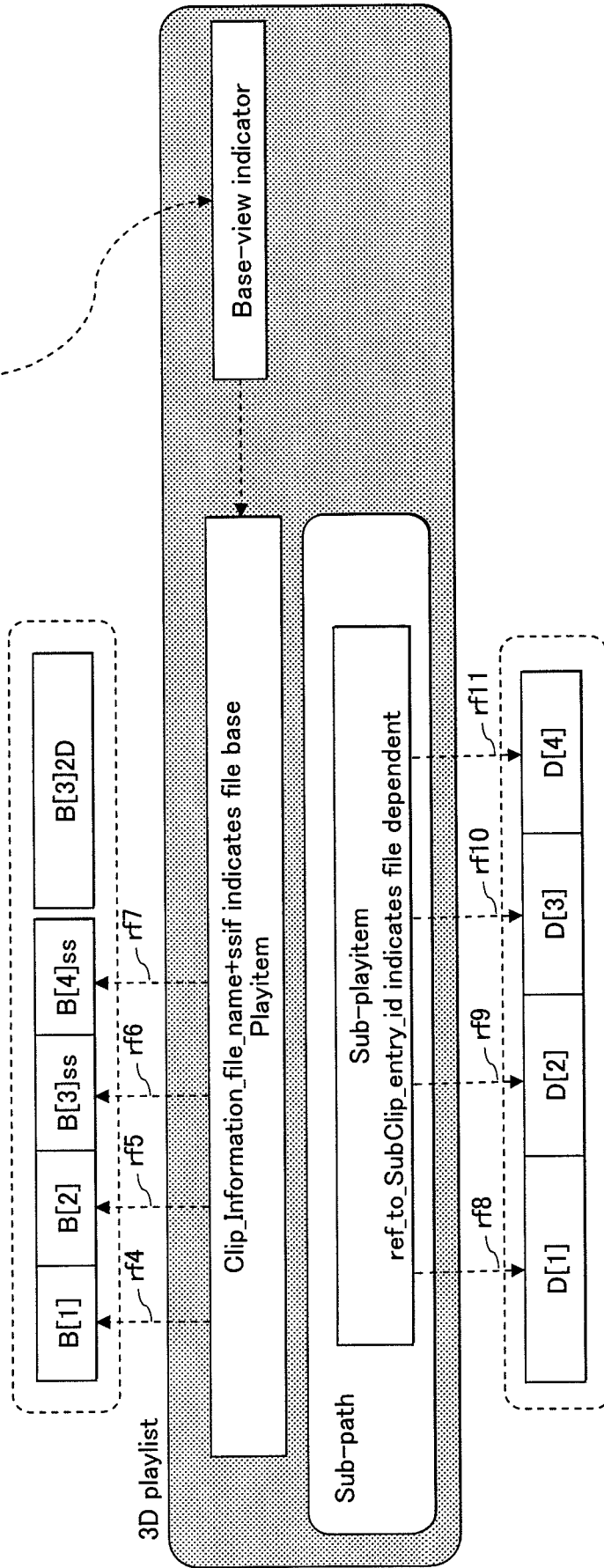


FIG.35B

Play Item {
Clip_Information_file_name
.....
Base_View_Indicator
.....

FIG. 36

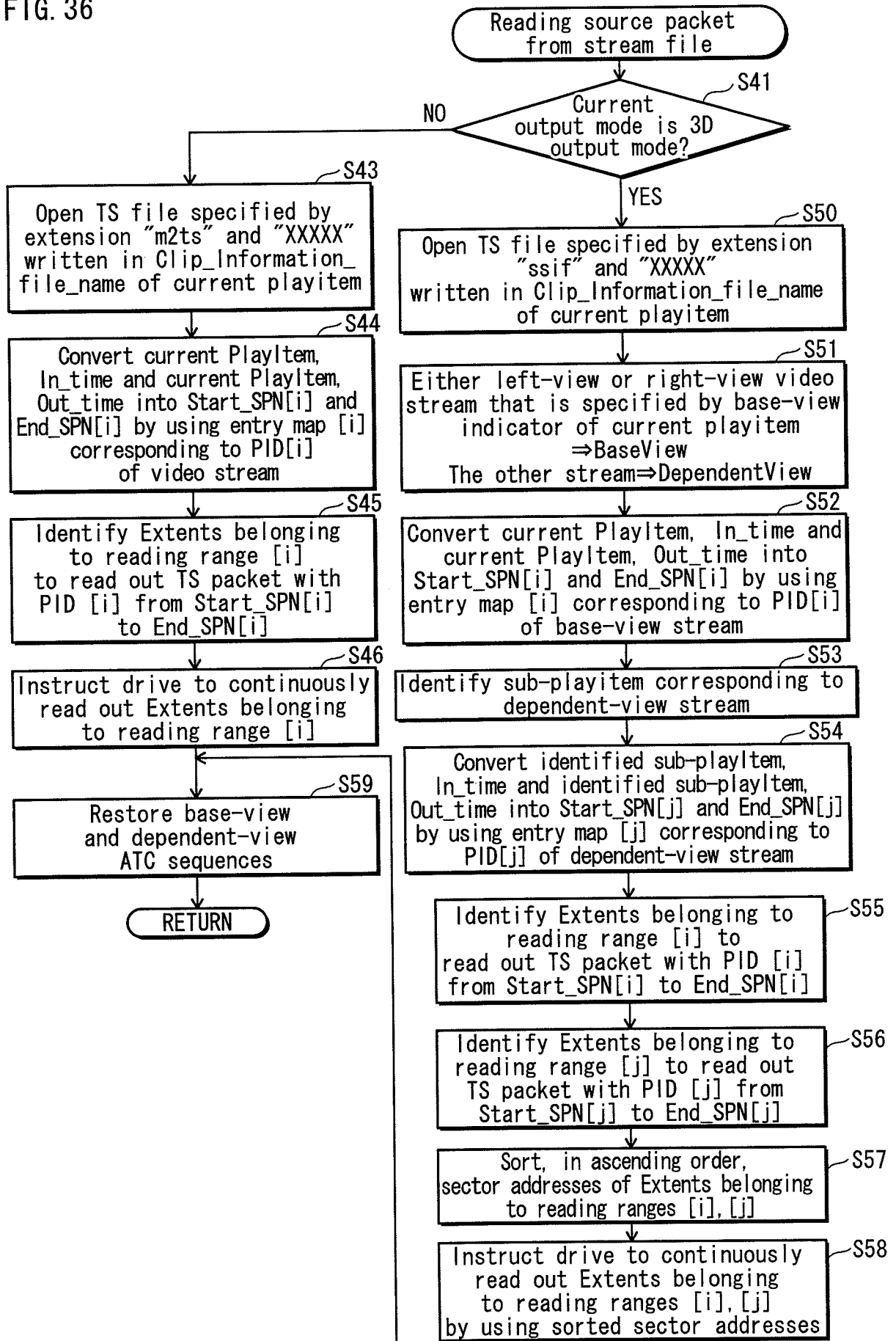


FIG. 37A

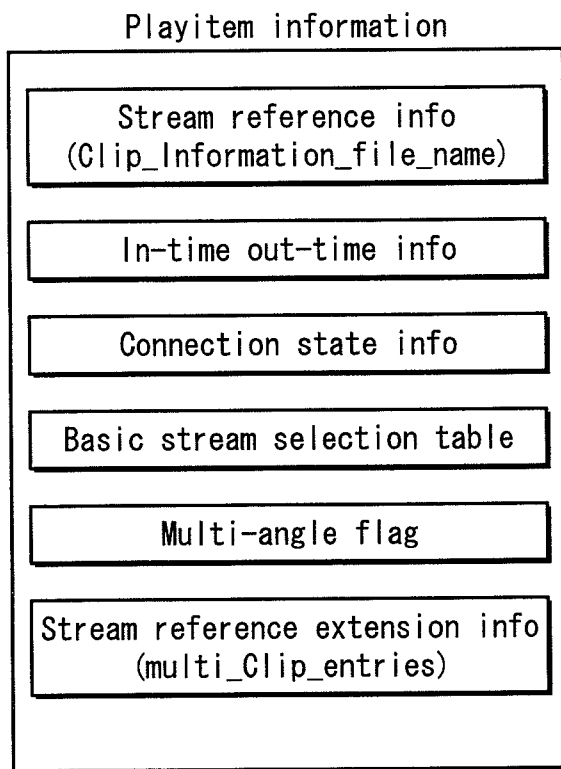


FIG. 37B

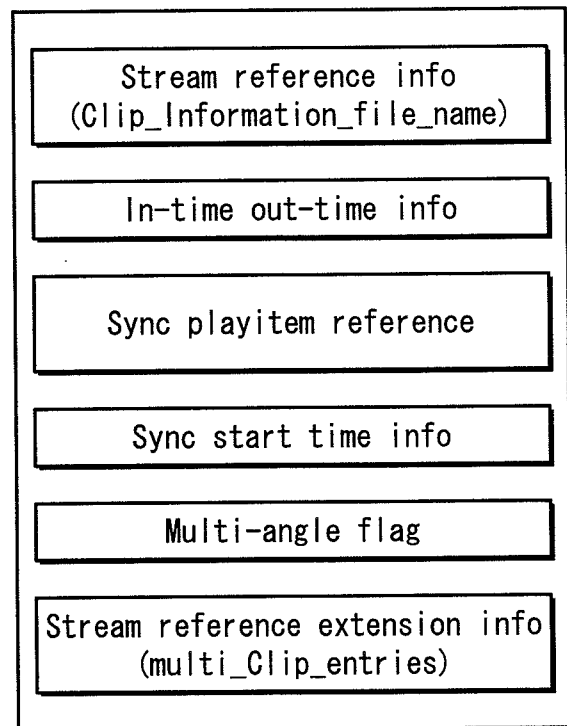


FIG. 37C

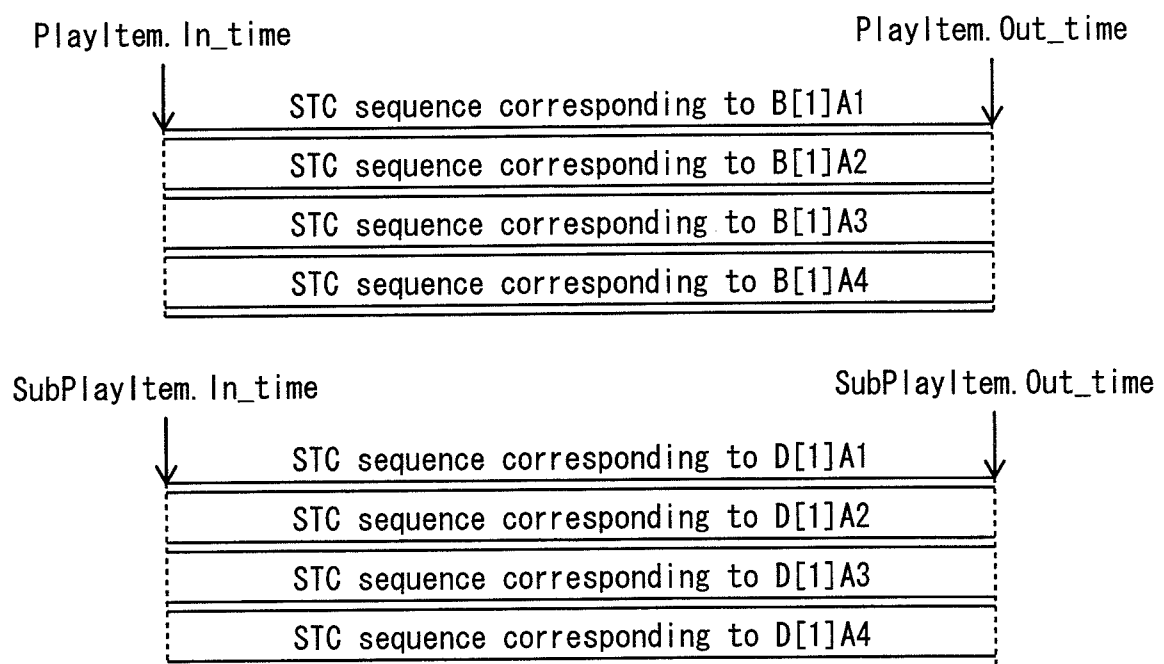


FIG. 38

Playitem information

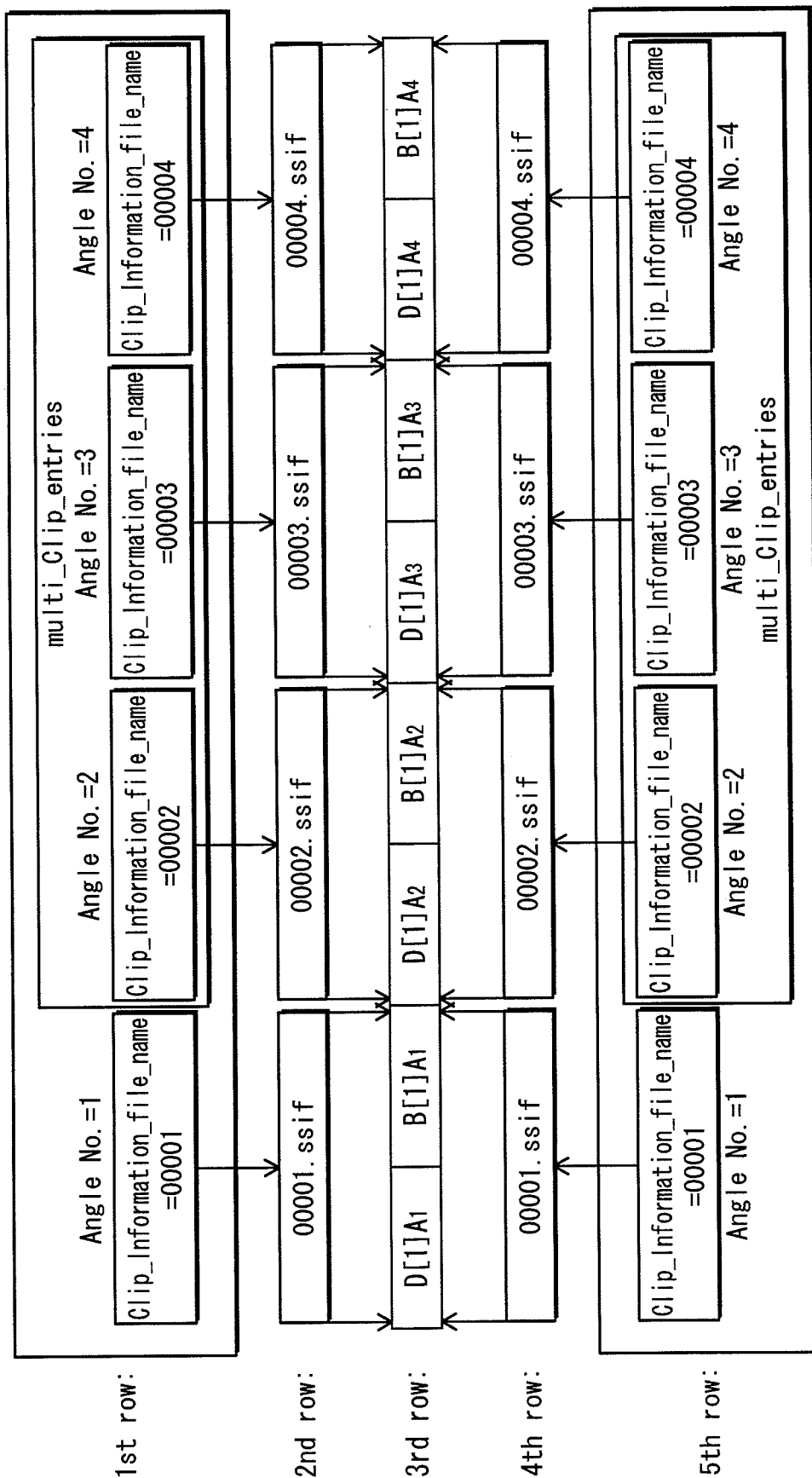


FIG. 39A Stereoscopic video
from front angle

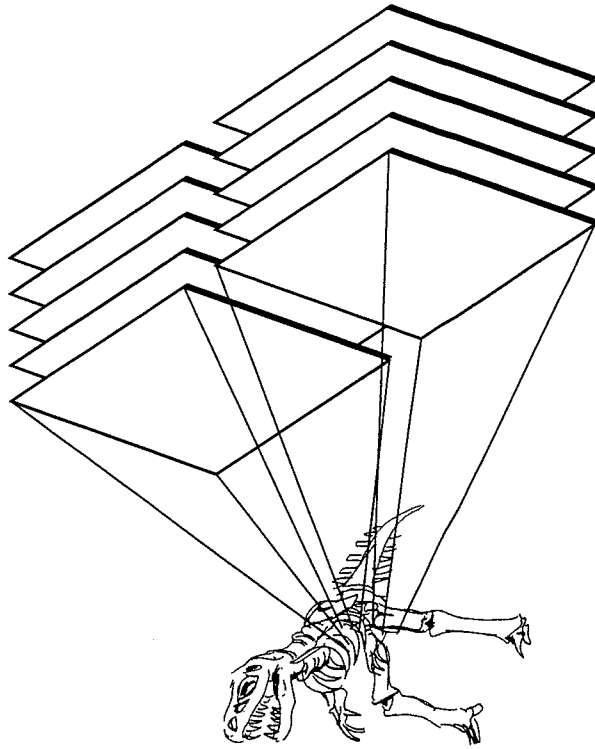


FIG. 39B Stereoscopic video
from back angle

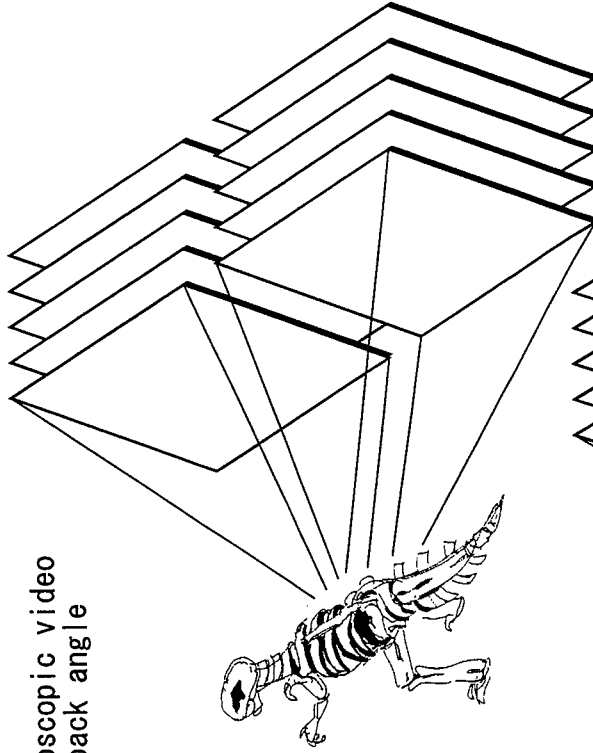


FIG. 39C Stereoscopic video
from upper-left angle

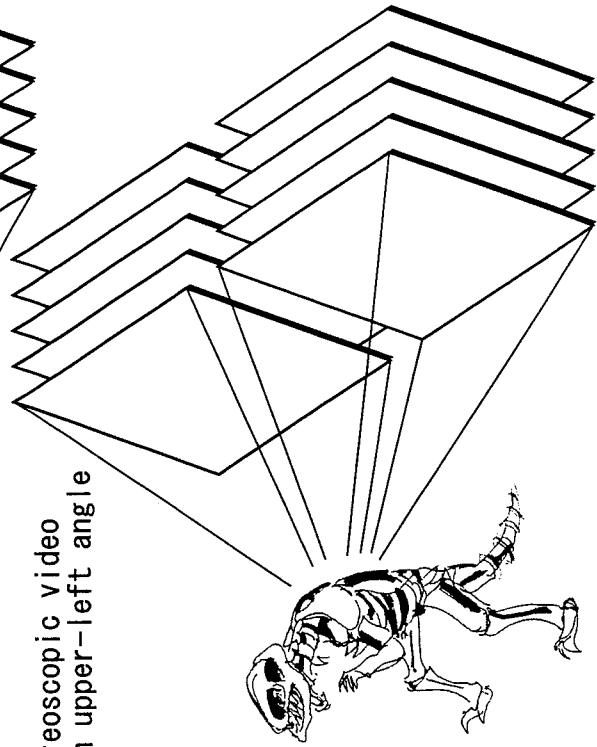


FIG. 40

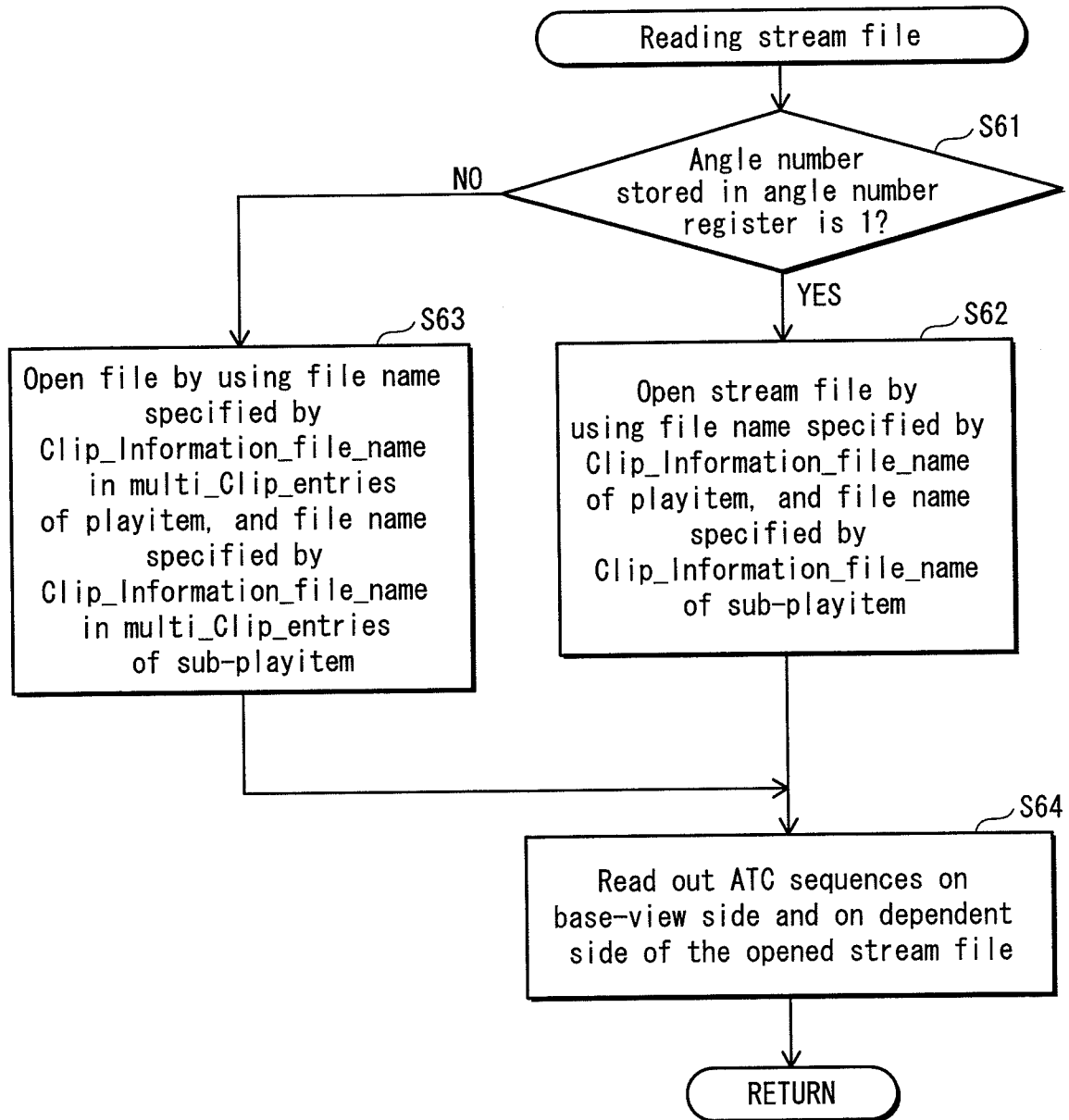


FIG. 41A

Clip information file

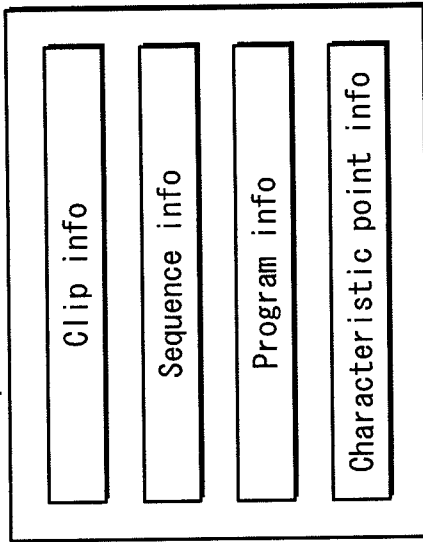


FIG. 41B

Clip information file

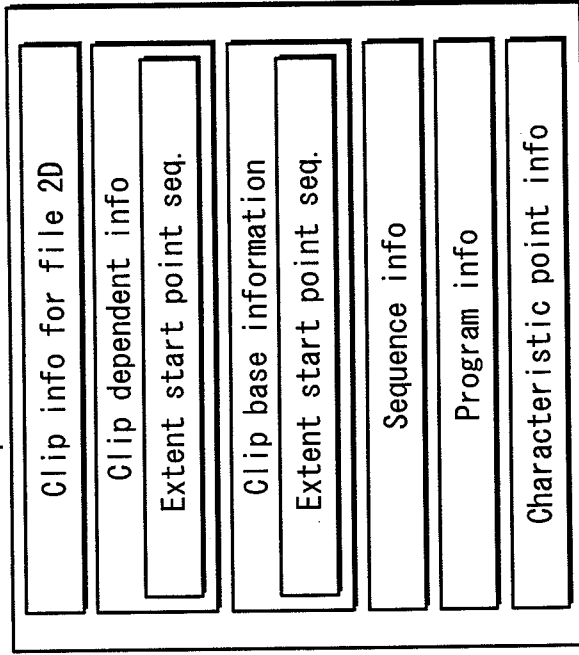


FIG. 41C

Extention_Start_Point

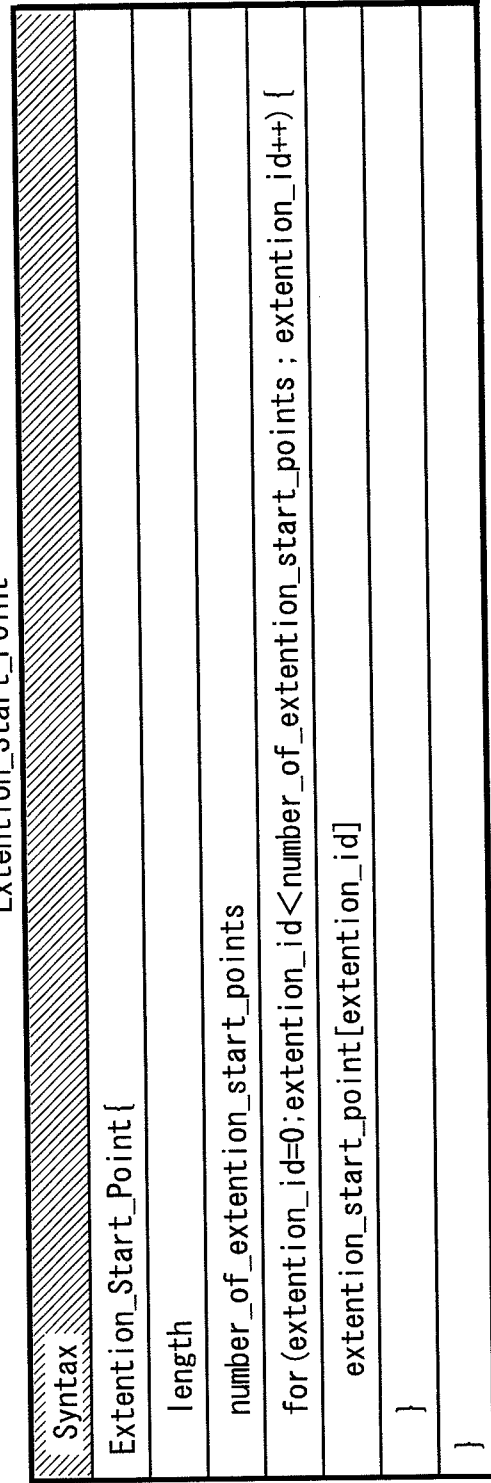


FIG. 42A

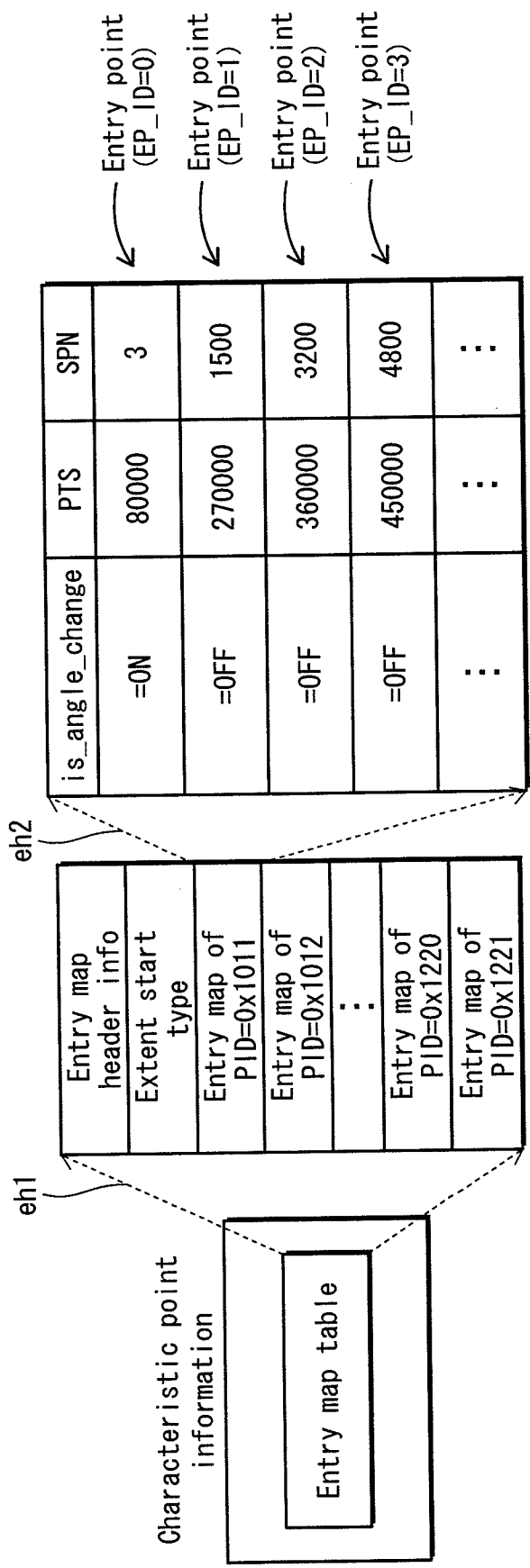


FIG. 42B

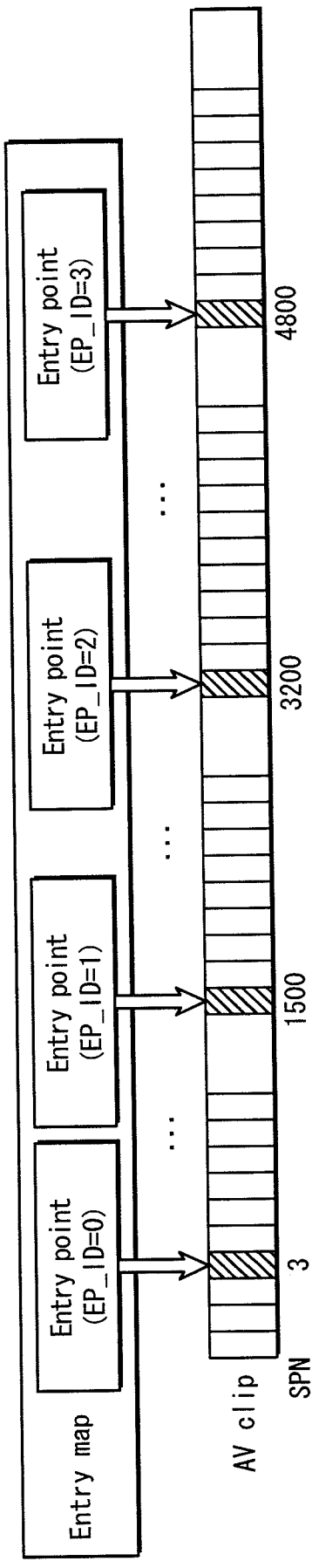


FIG. 43

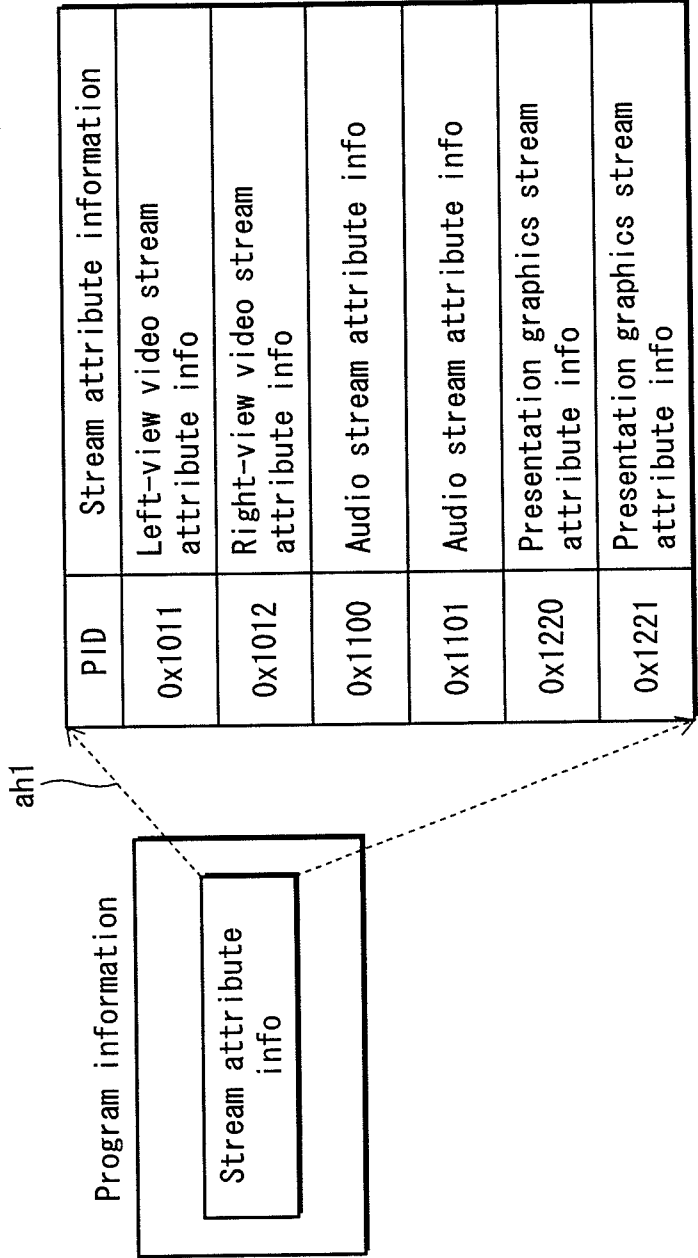
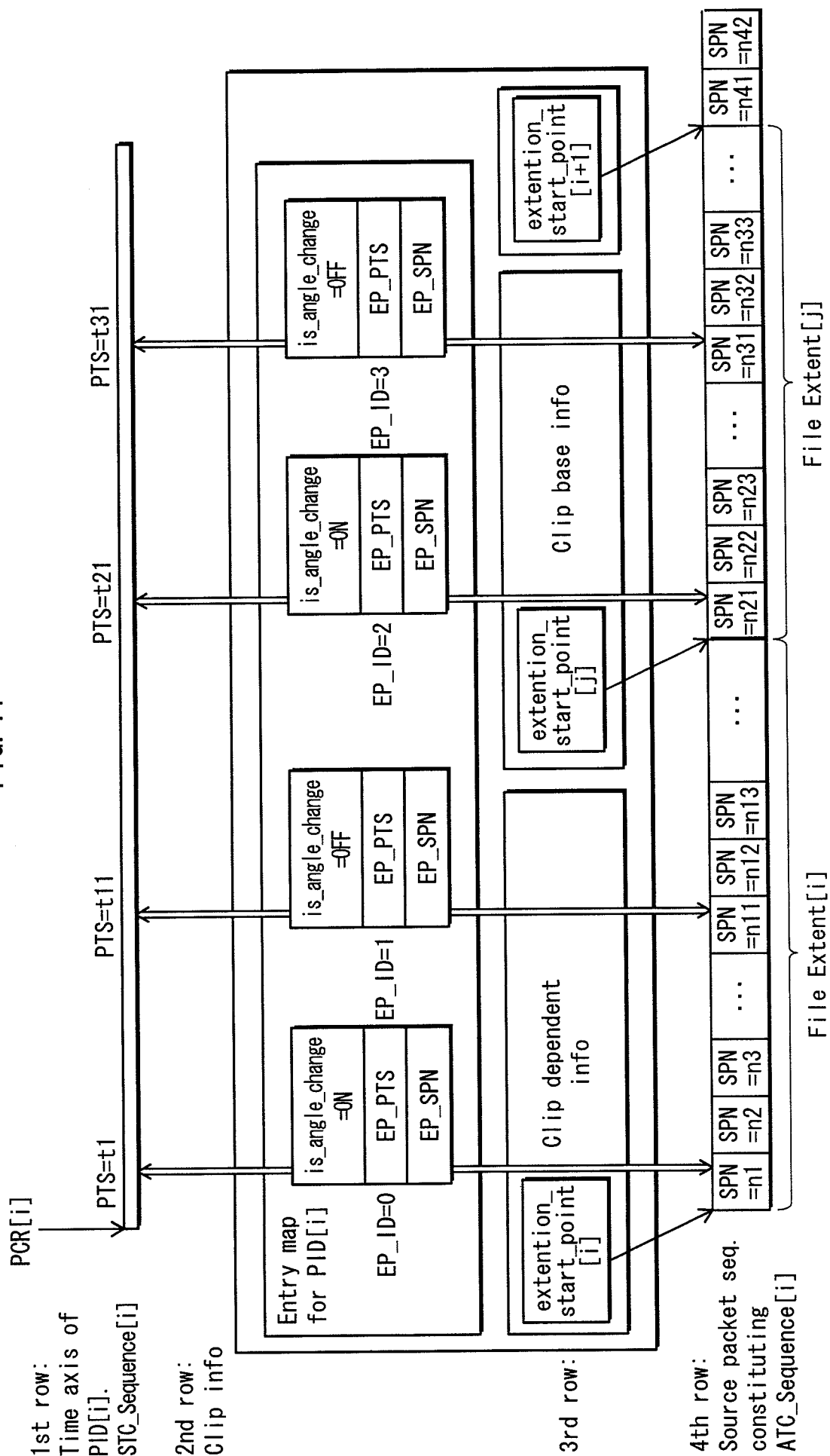


FIG. 44



The diagram illustrates the interleaved storage of source packet sequences in main TS and sub-TS. It shows four rows of data structures:

- 1st row:** Contains STC sequence 1, Entry map, PlayItem. In_time, and PlayItem. Out_time.
- 2nd row:** Contains SubPlayItem. In_time, STC sequence 2, Entry map, and SubPlayItem. Out_time.
- 3rd row:** Contains Start_SPN, ATC sequence 1, and End_SPN.
- 4th row:** Contains Start_SPN, ATC sequence 2, and End_SPN.

Arrows indicate the mapping from the 3rd and 4th rows to the 1st and 2nd rows respectively, showing the interleaved storage of source packet sequences in main TS and sub-TS.

FIG. 46A

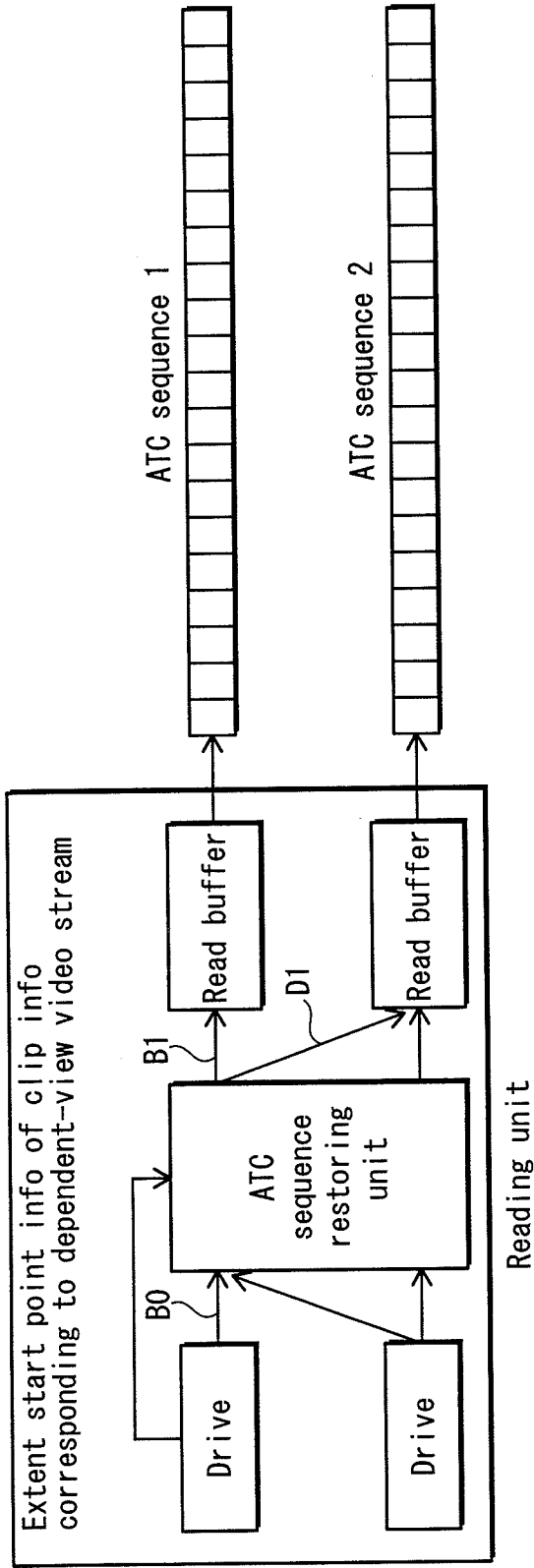


FIG. 46B

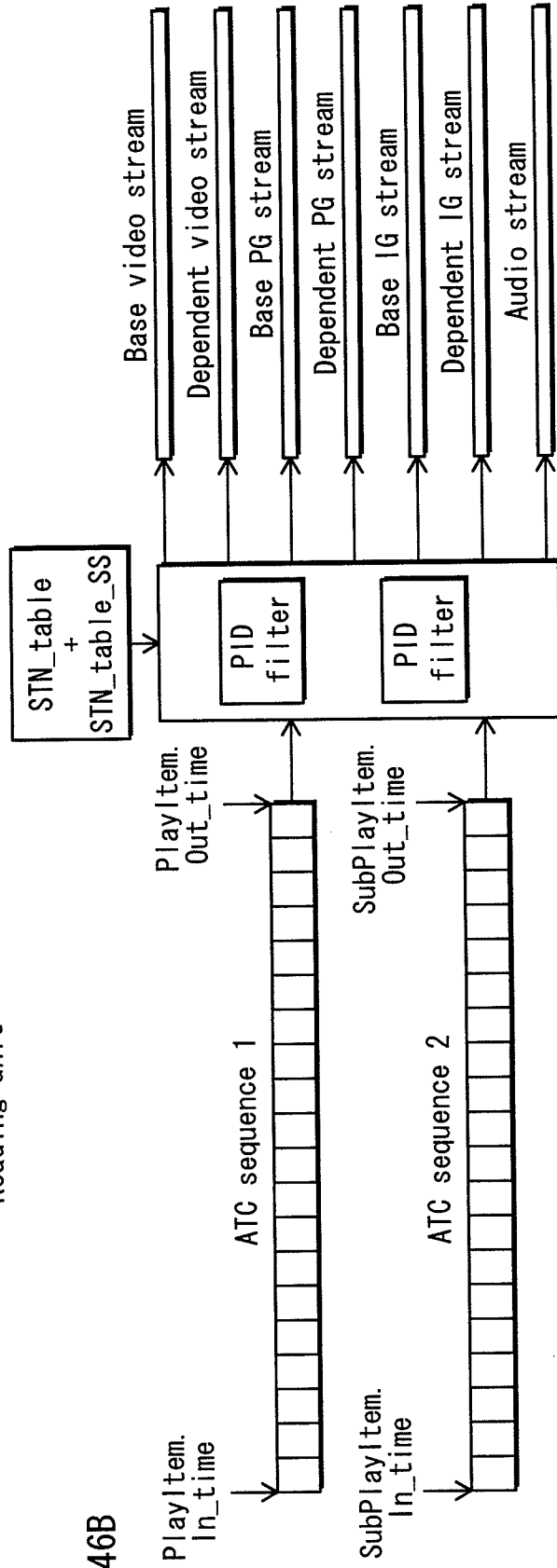


FIG. 47

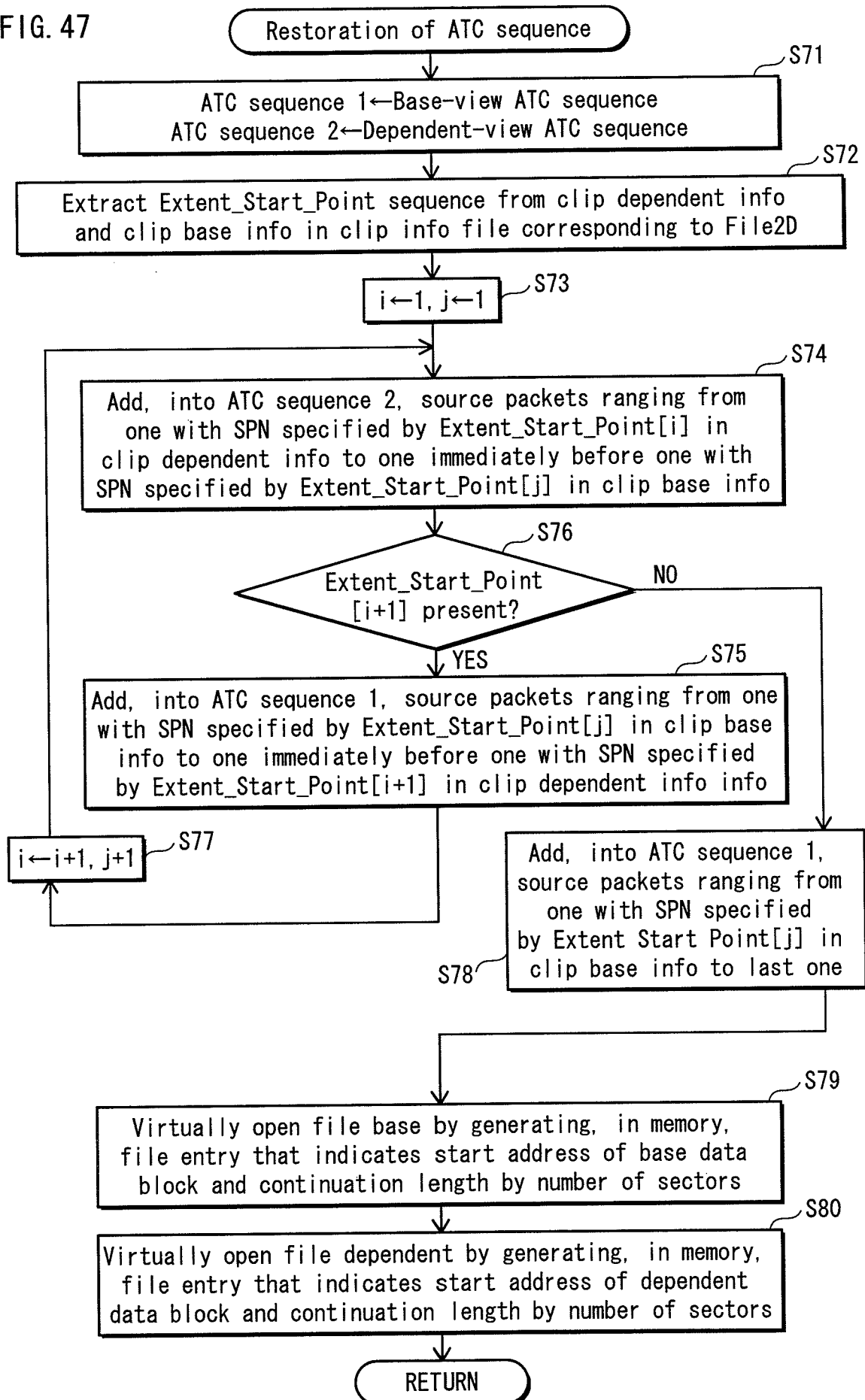


FIG. 48A

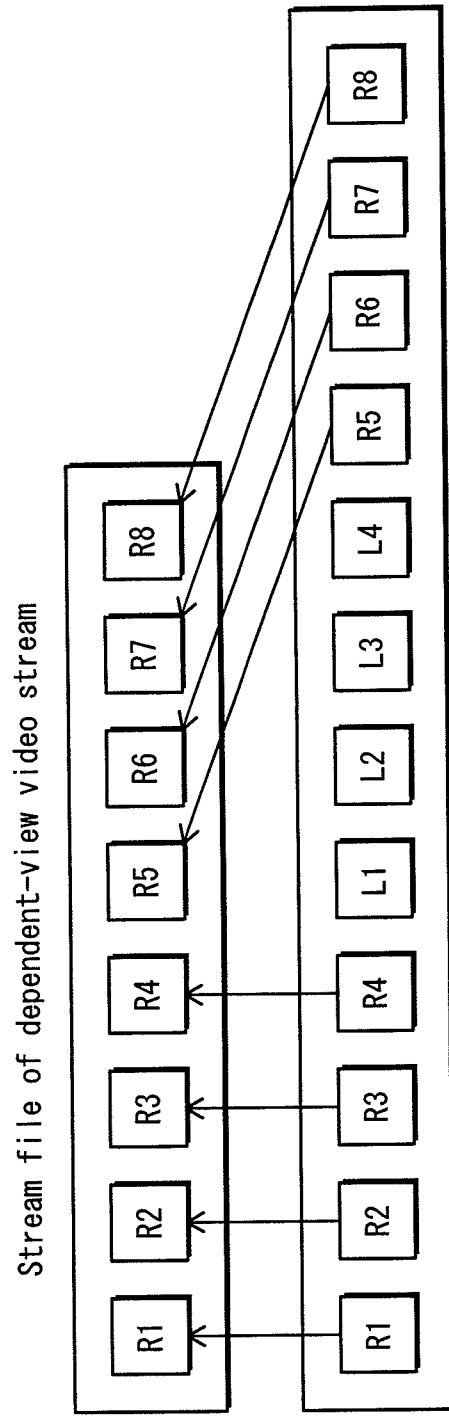


FIG. 48B

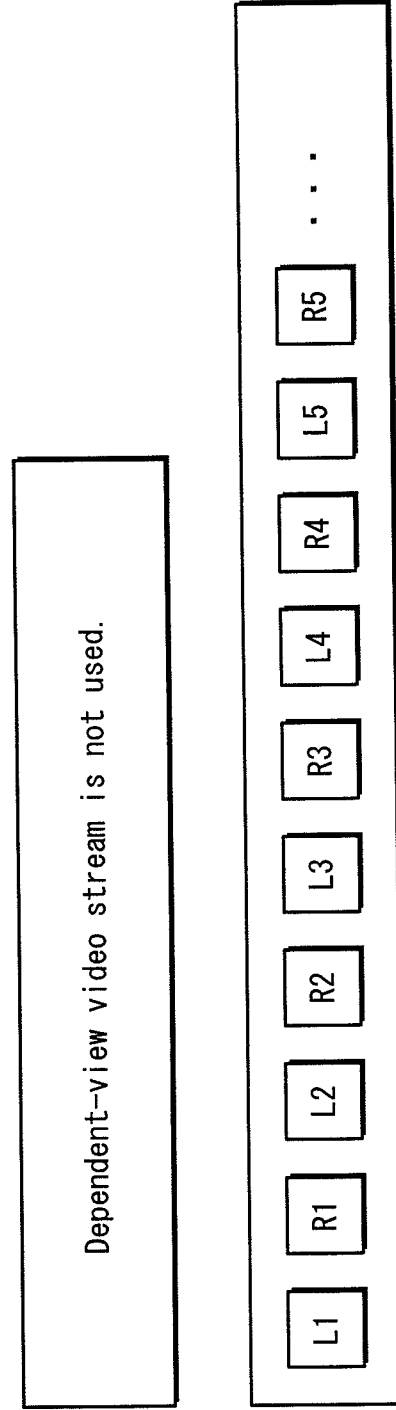
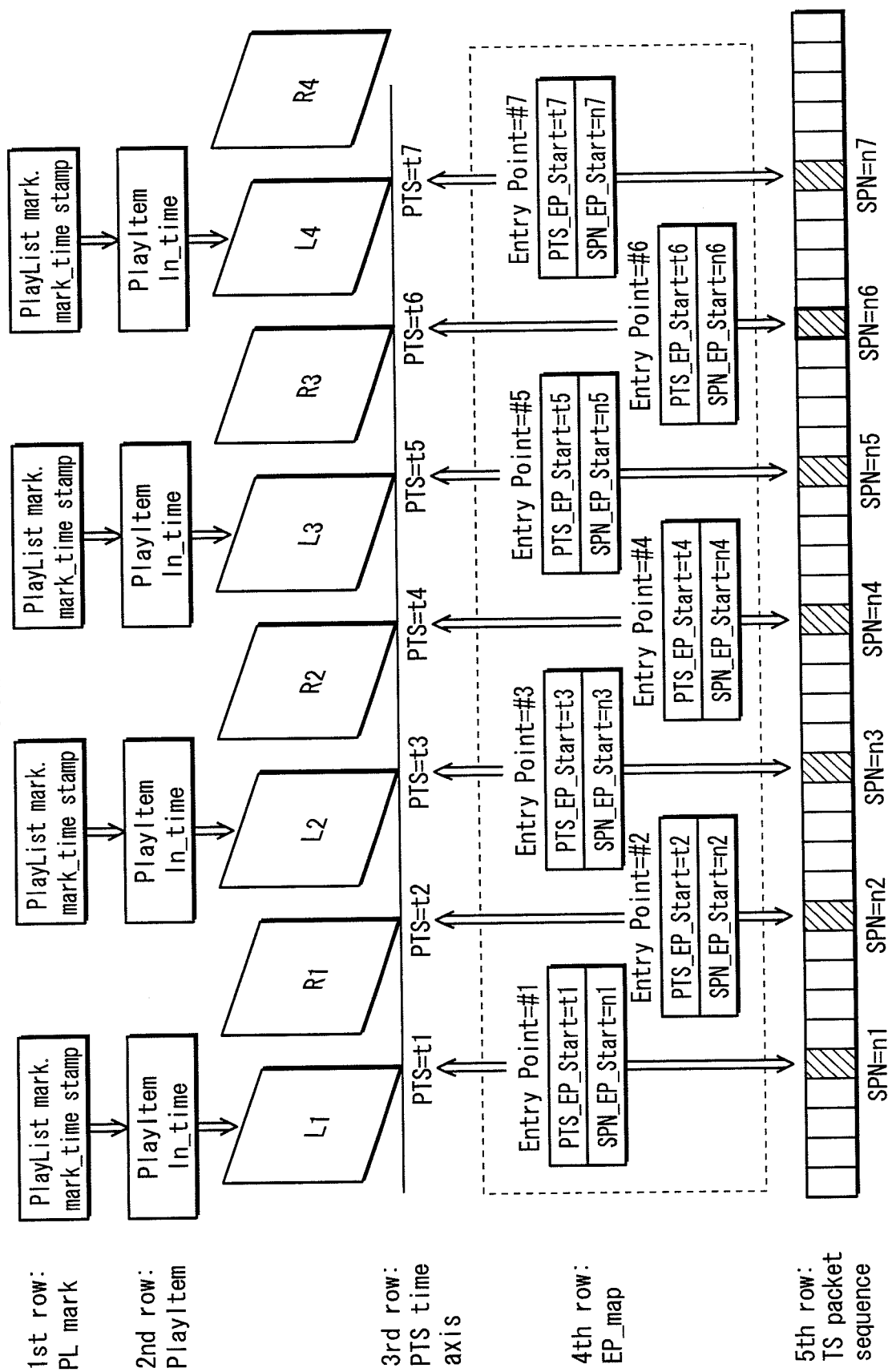


FIG. 49



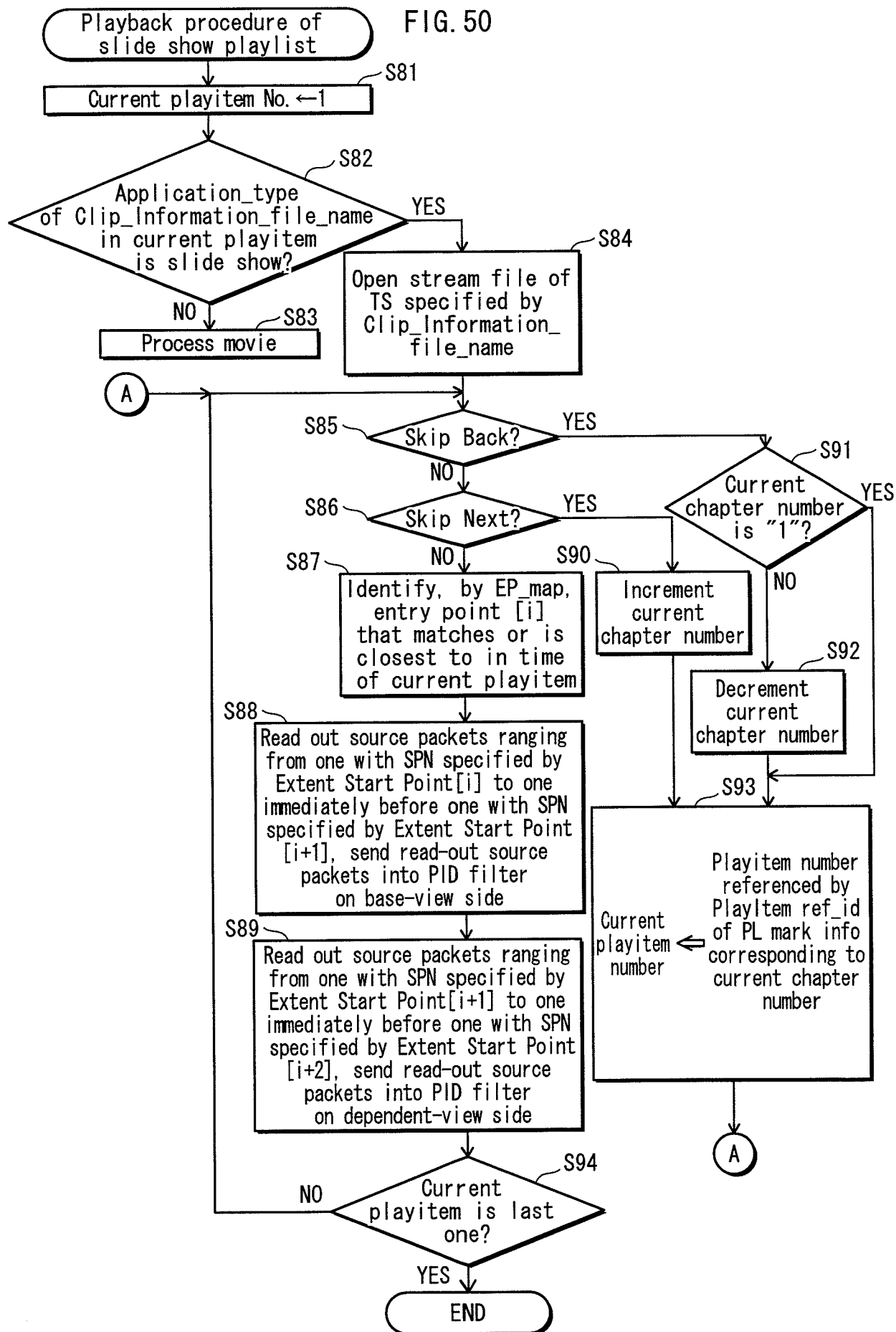


Figure 1 is a block diagram of a demultiplexing unit. The unit is composed of two parallel processing paths, labeled 2a and 2b. Path 2a includes a Read buffer (2a) connected to a Source depacketizer (22), which is then connected to a Decoder (23). Path 2b includes a Read buffer (2b) connected to a Source depacketizer (26), which is then connected to a Decoder (27). A dashed box labeled 21 encloses the Source depacketizers (22 and 26) and the Decoders (23 and 27). The entire unit is labeled 'Demultiplexing unit' on the right side.

FIG.52A

L/R mode

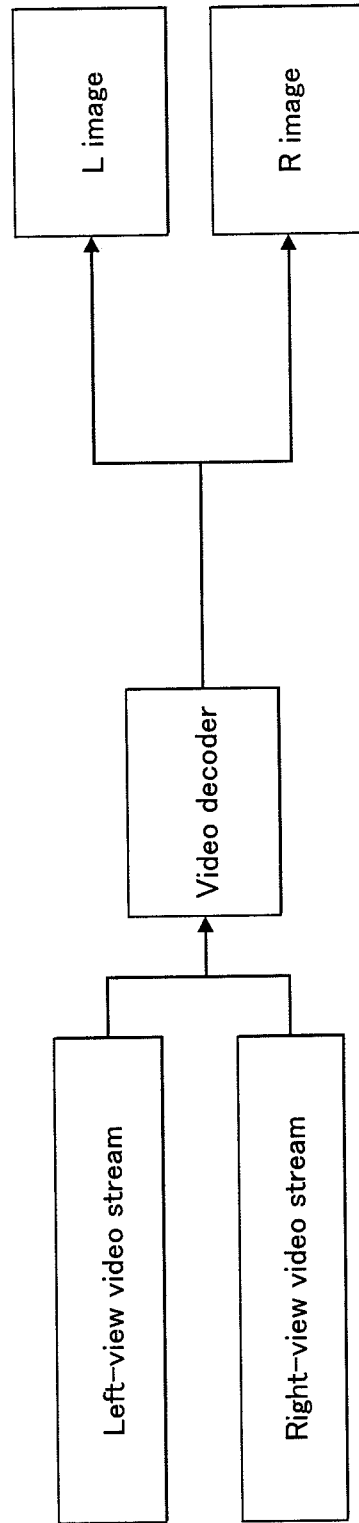


FIG.52B

DEPTH mode

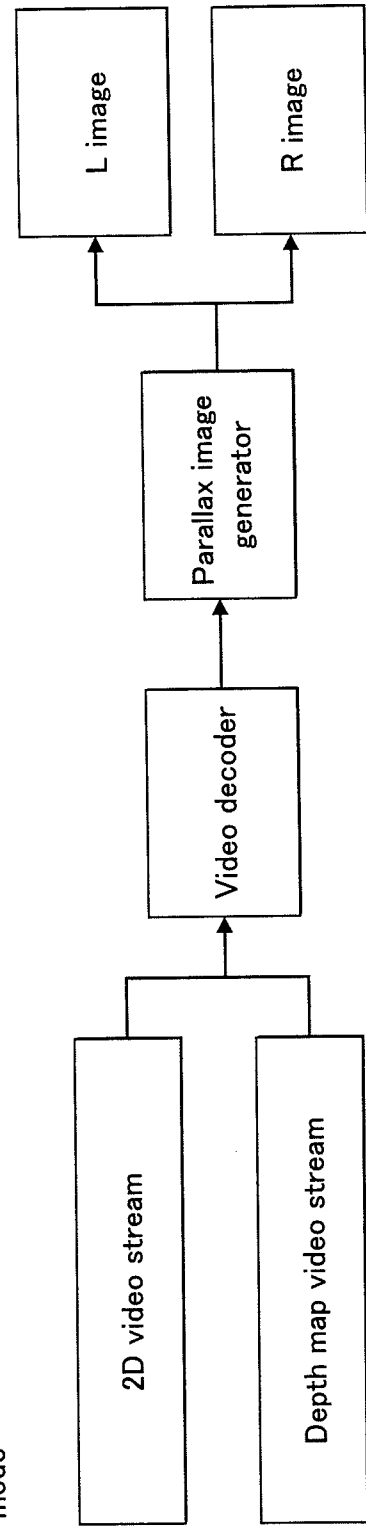


FIG.53A

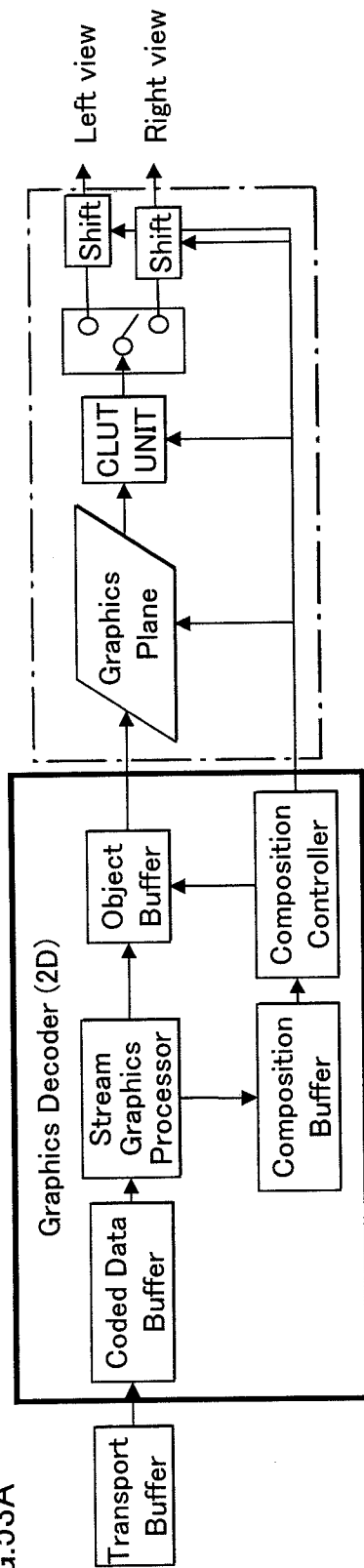


FIG.53B

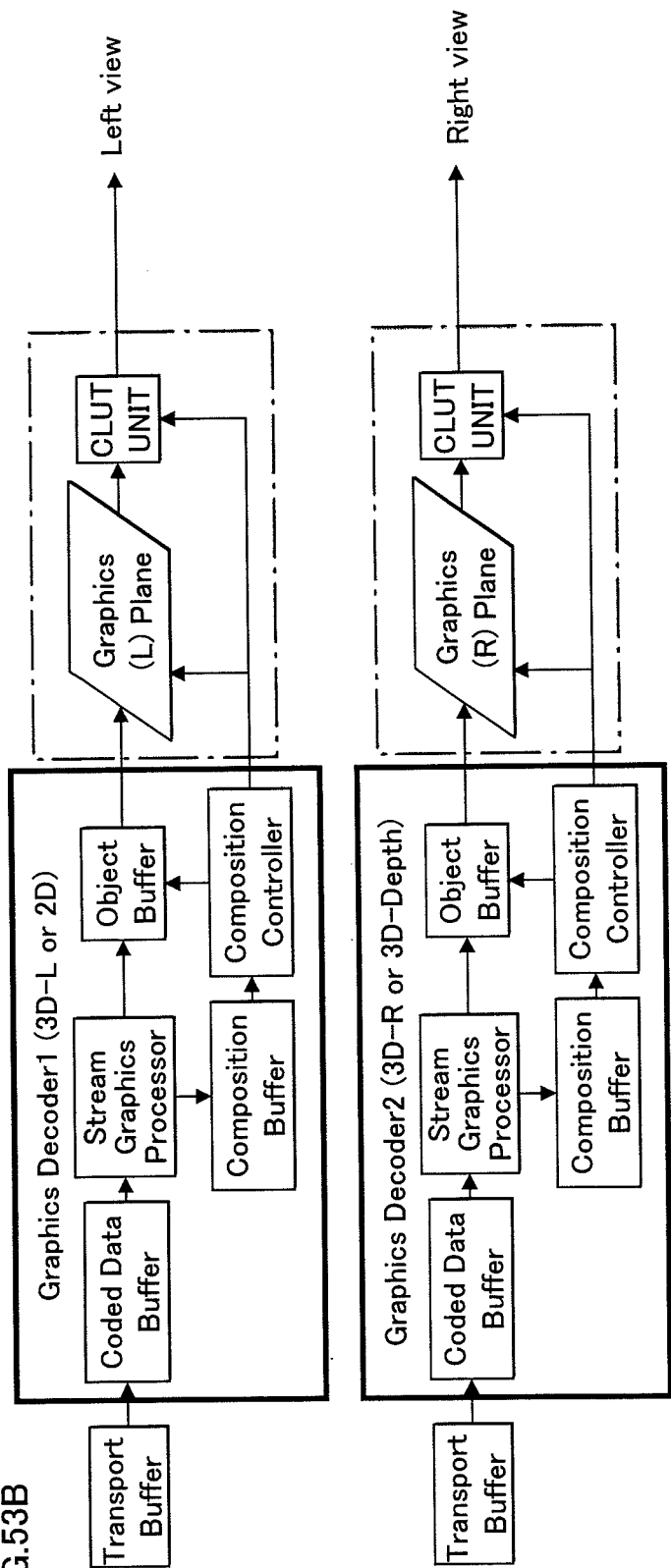


FIG.54A

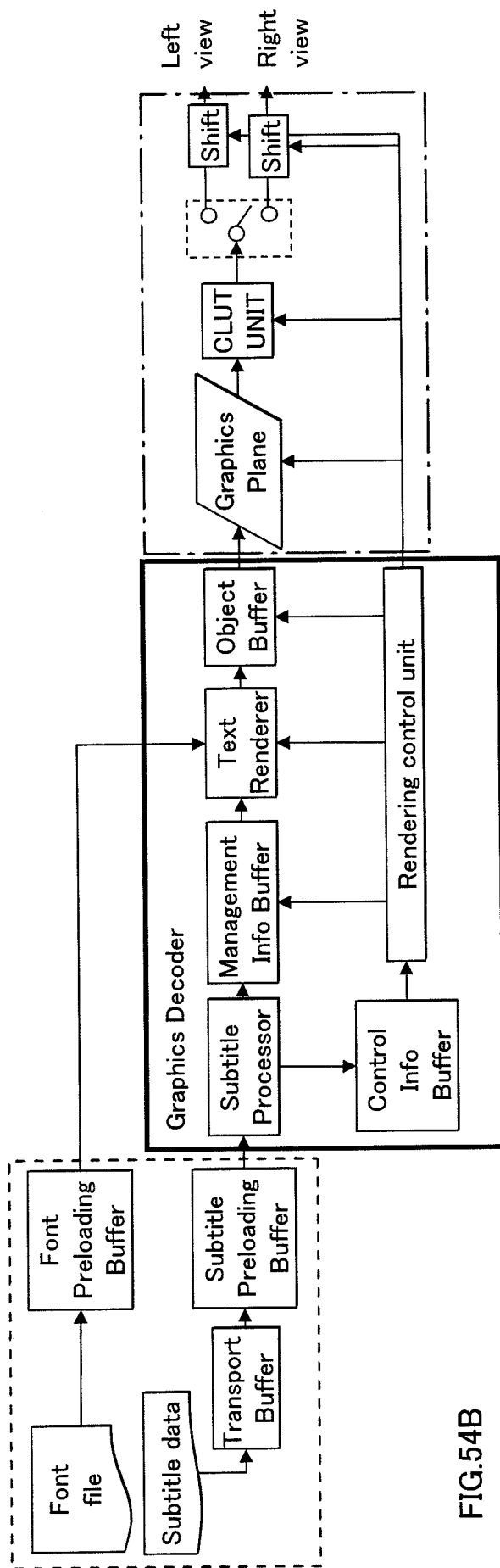
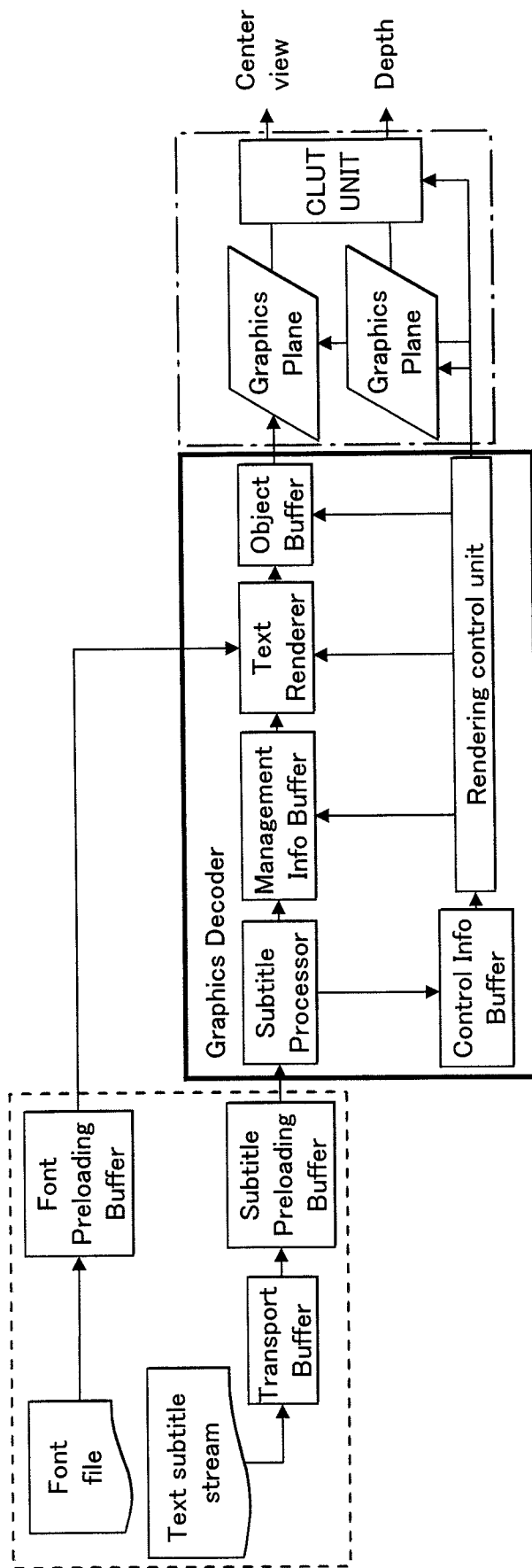


FIG.54B



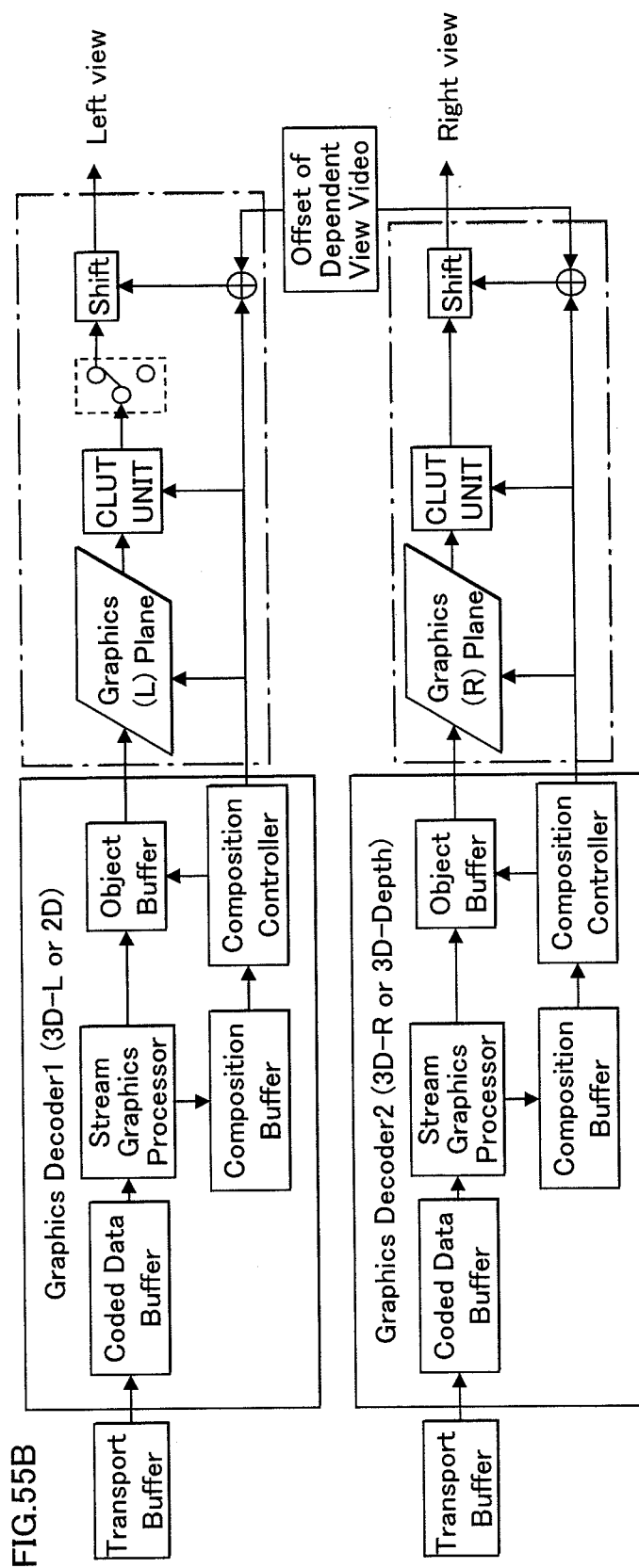
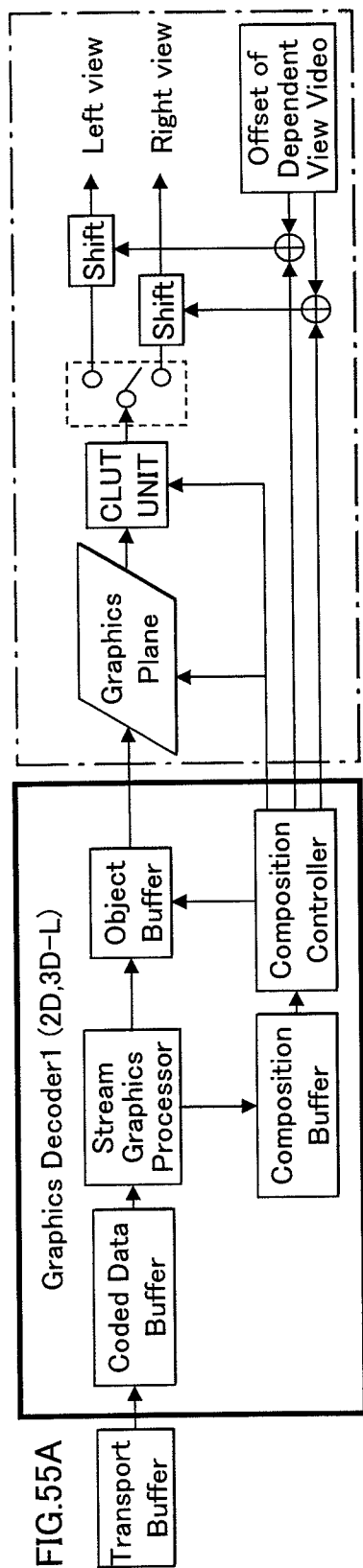


FIG.56

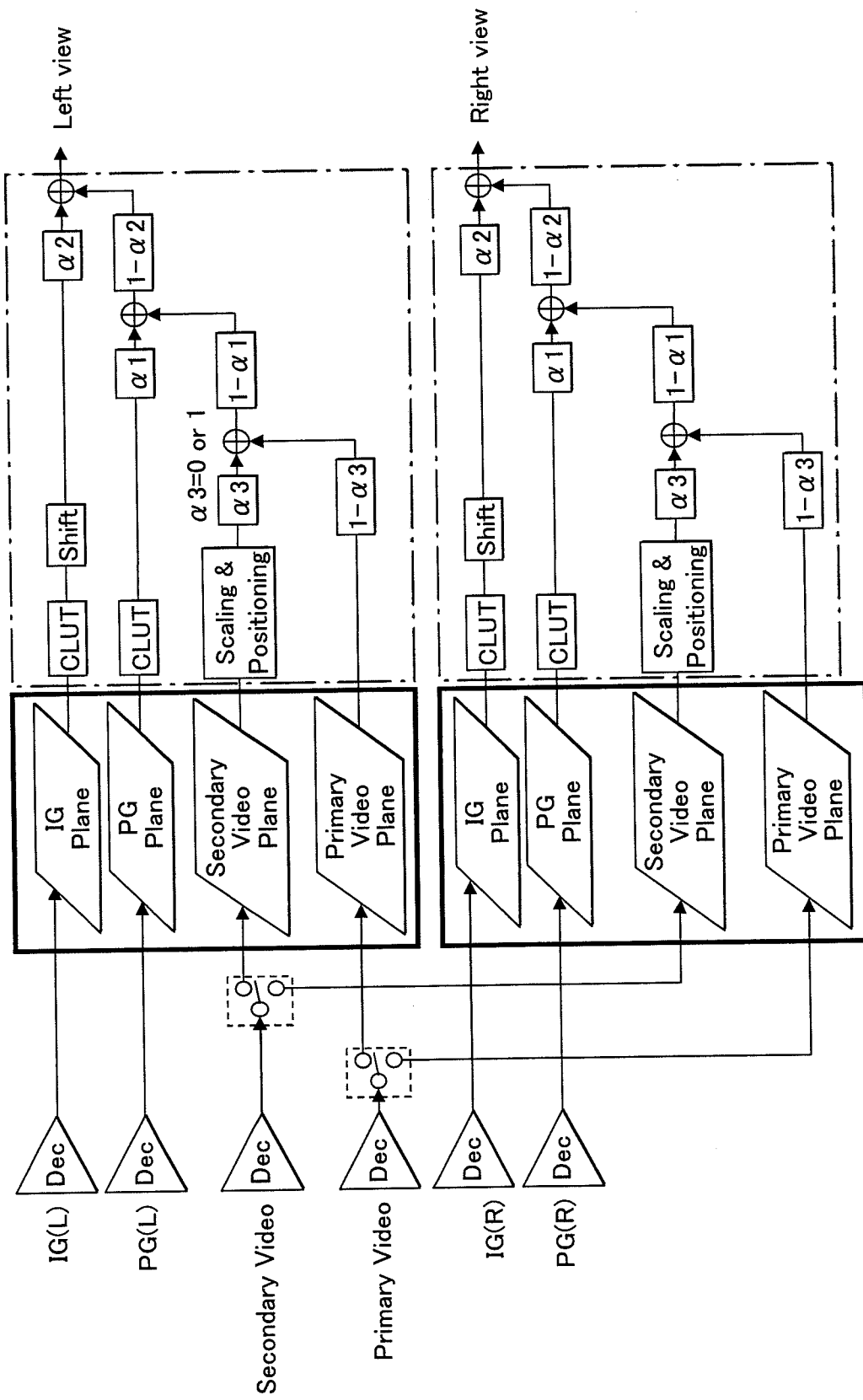


FIG.57

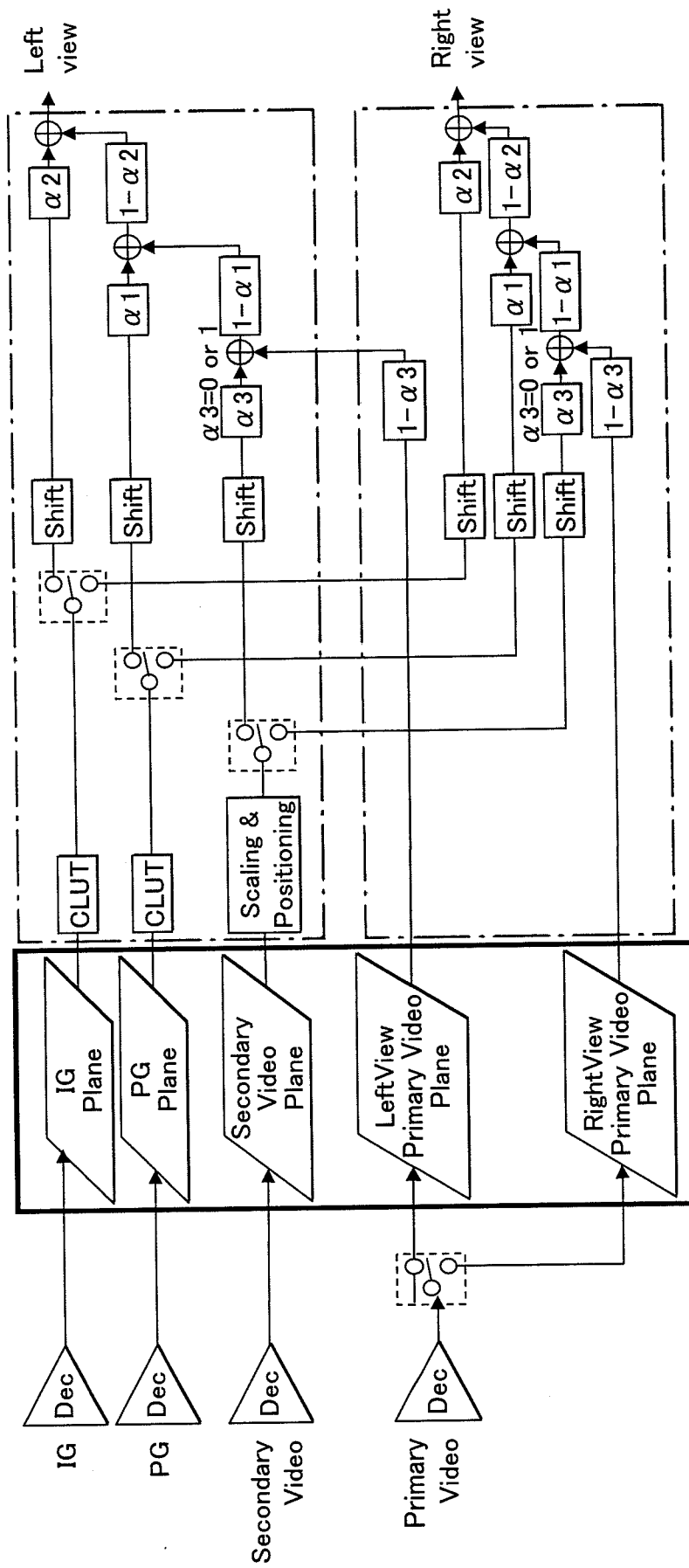
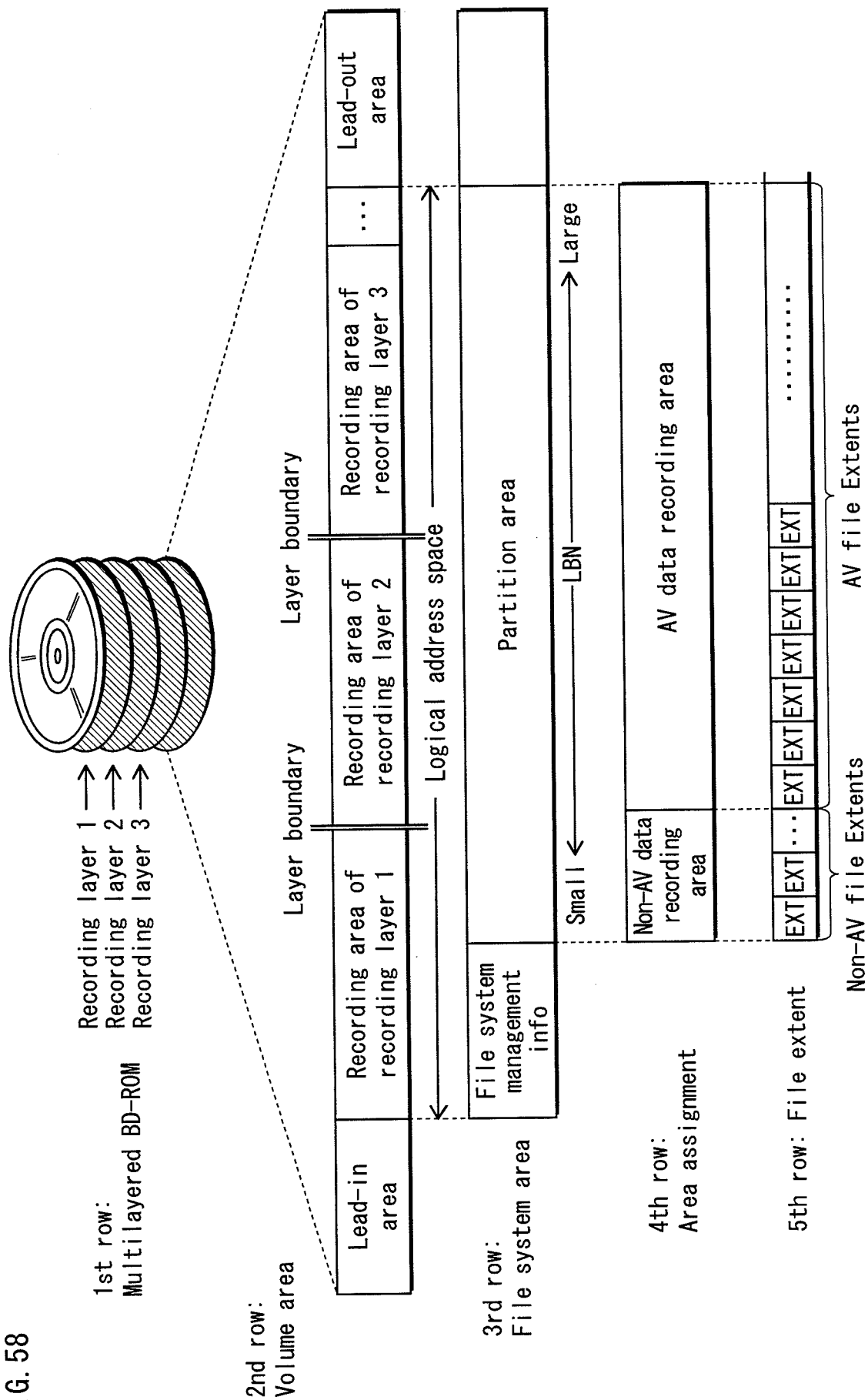


FIG. 58



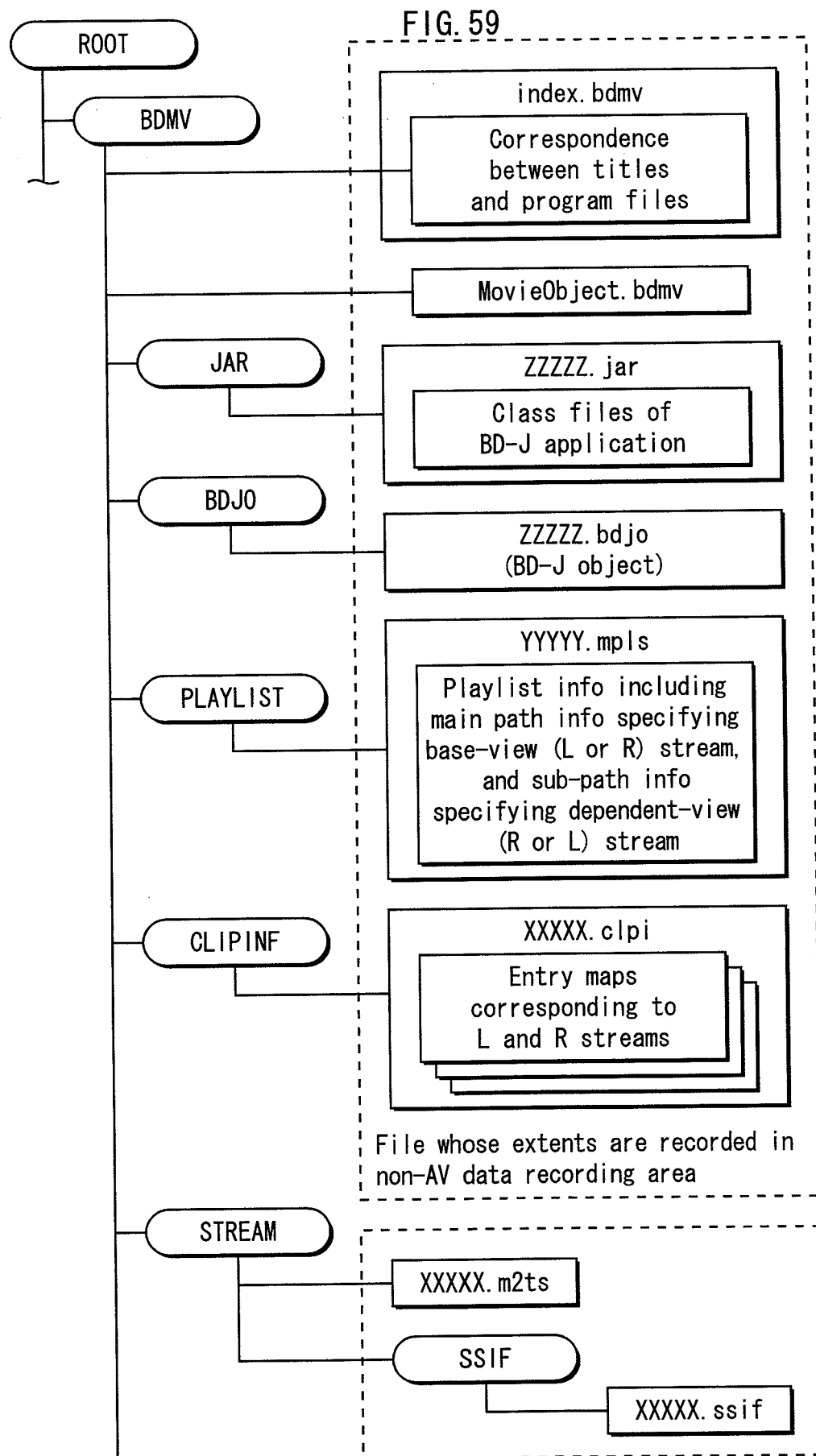


FIG. 60

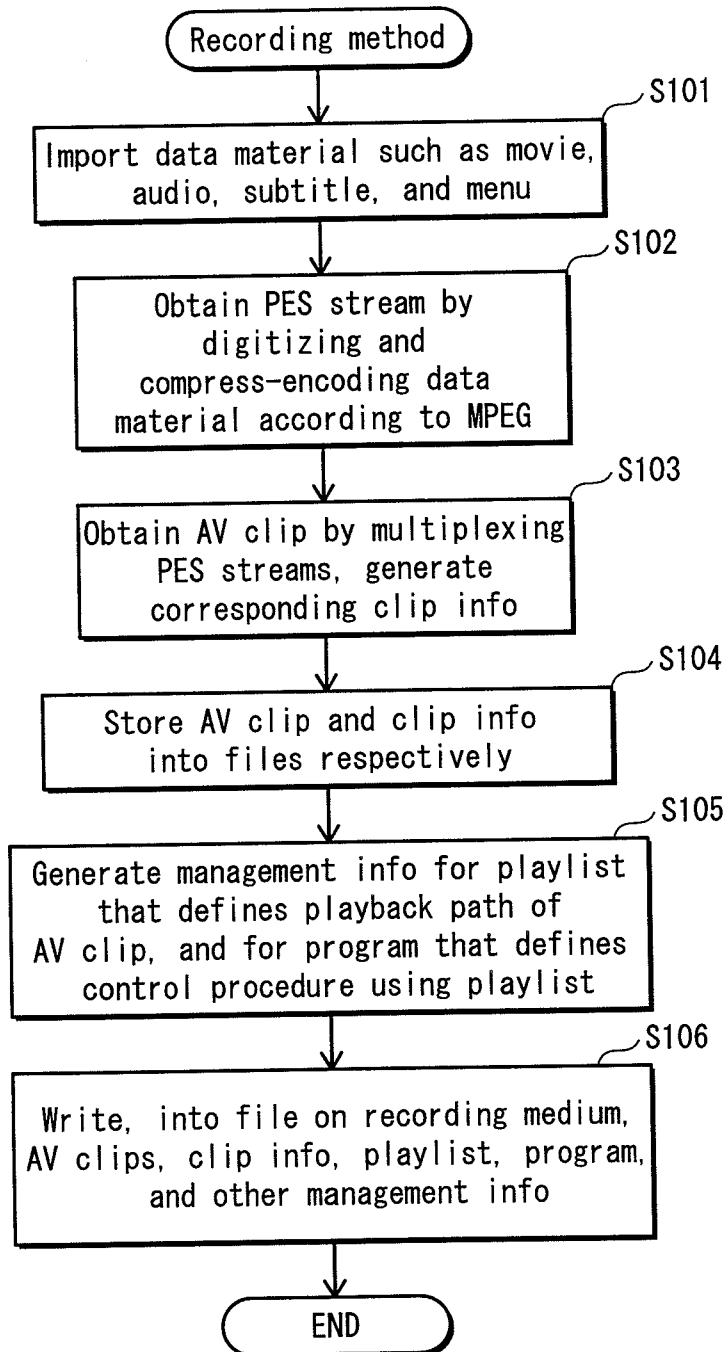


FIG. 61

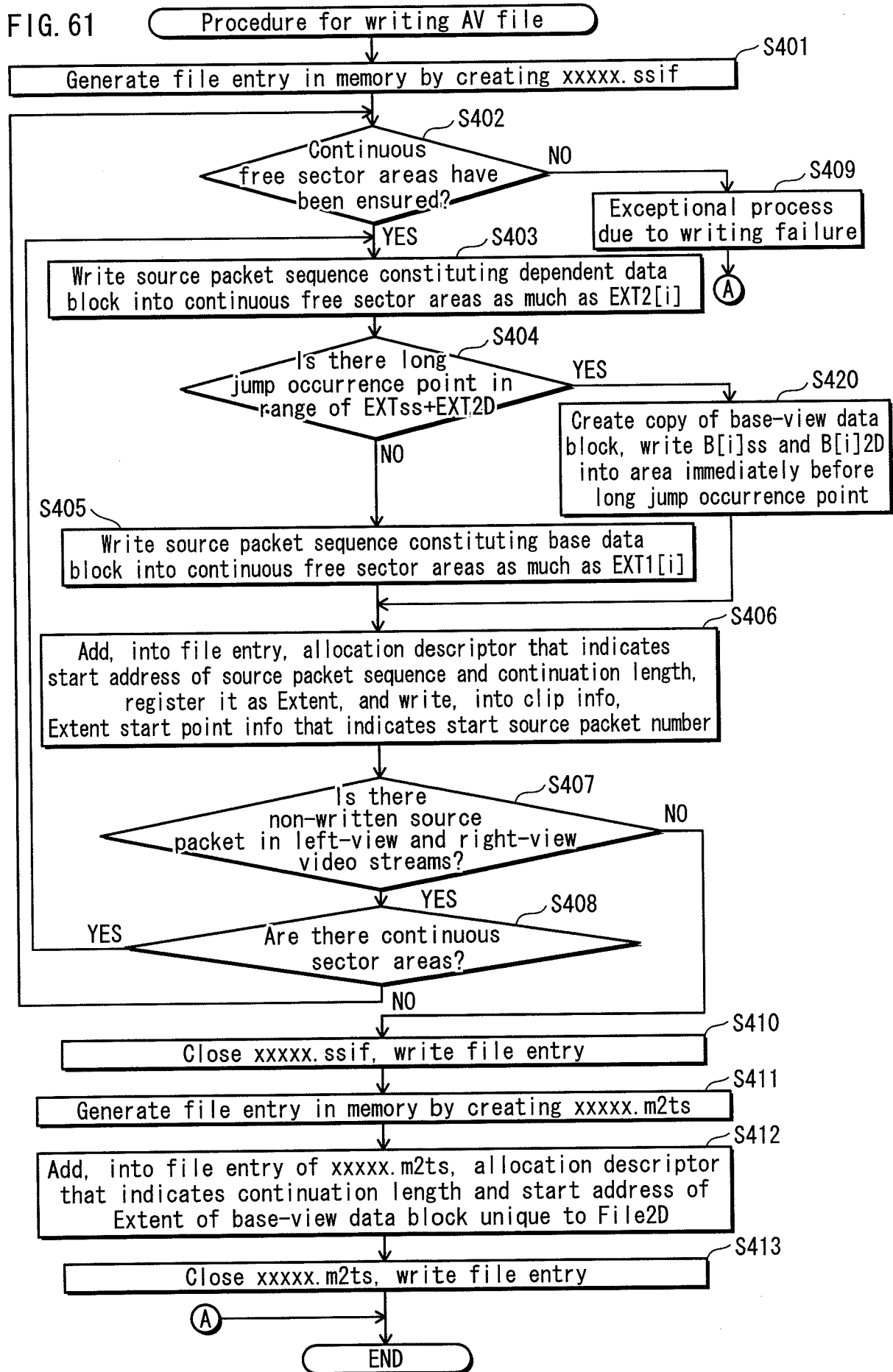


FIG. 62A

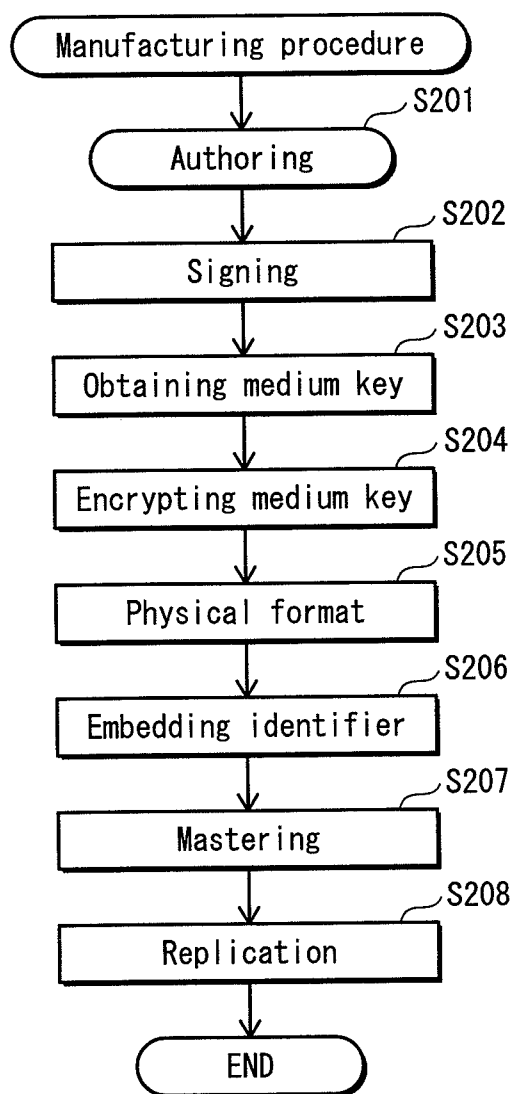


FIG. 62B

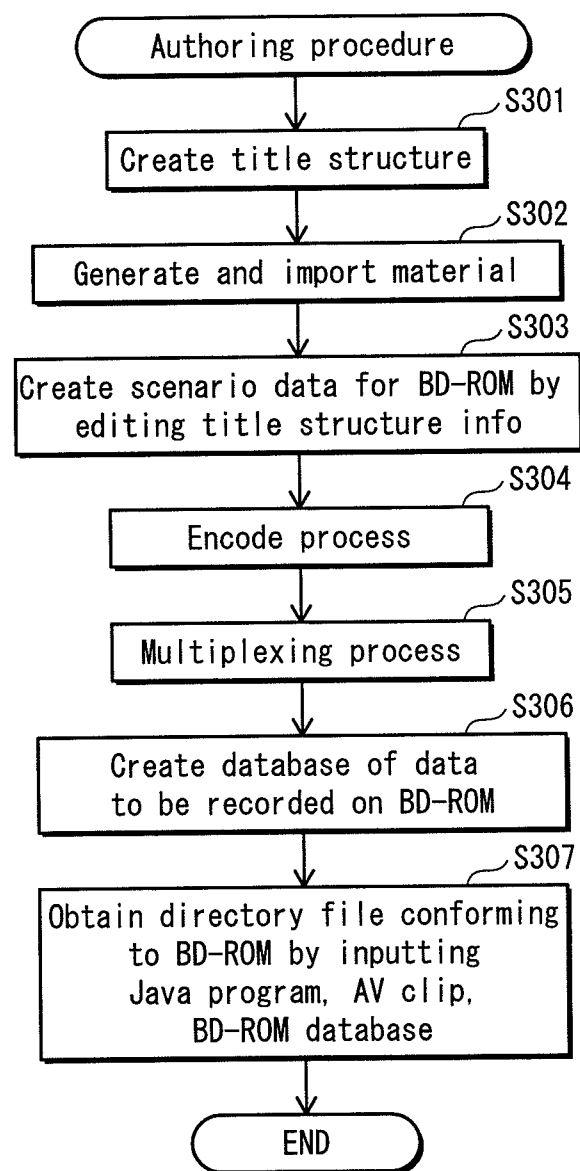


FIG.63

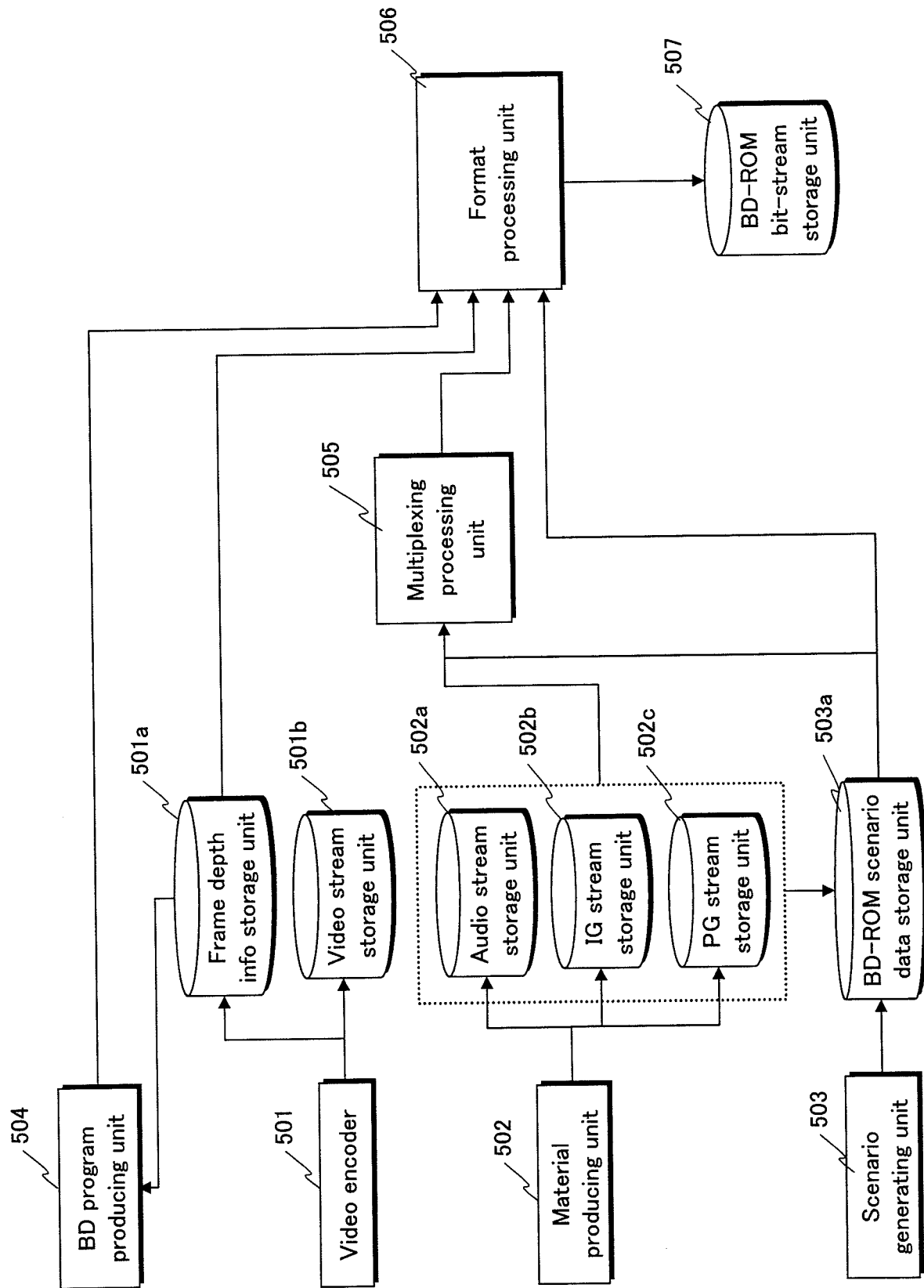


FIG. 64

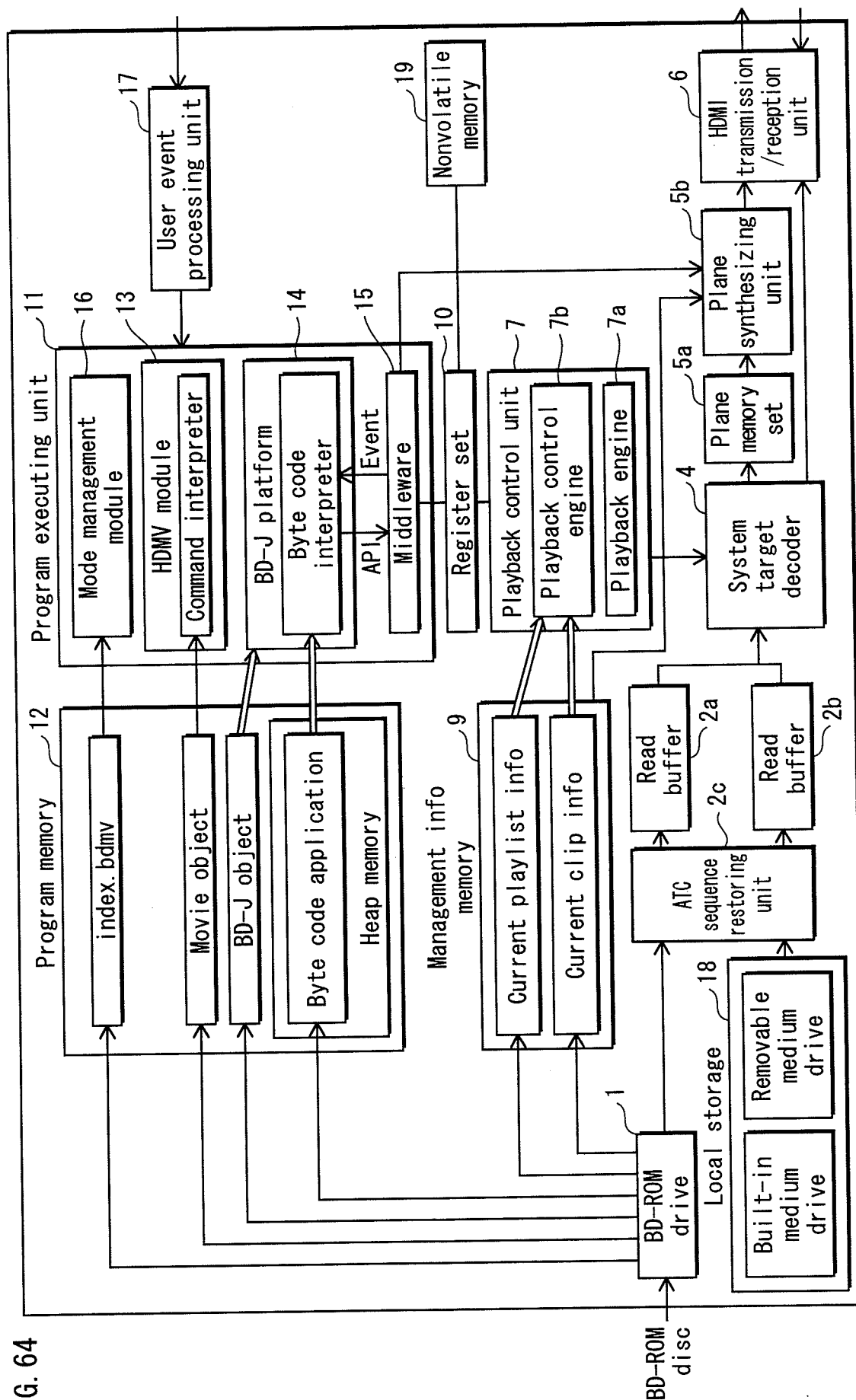


FIG.65

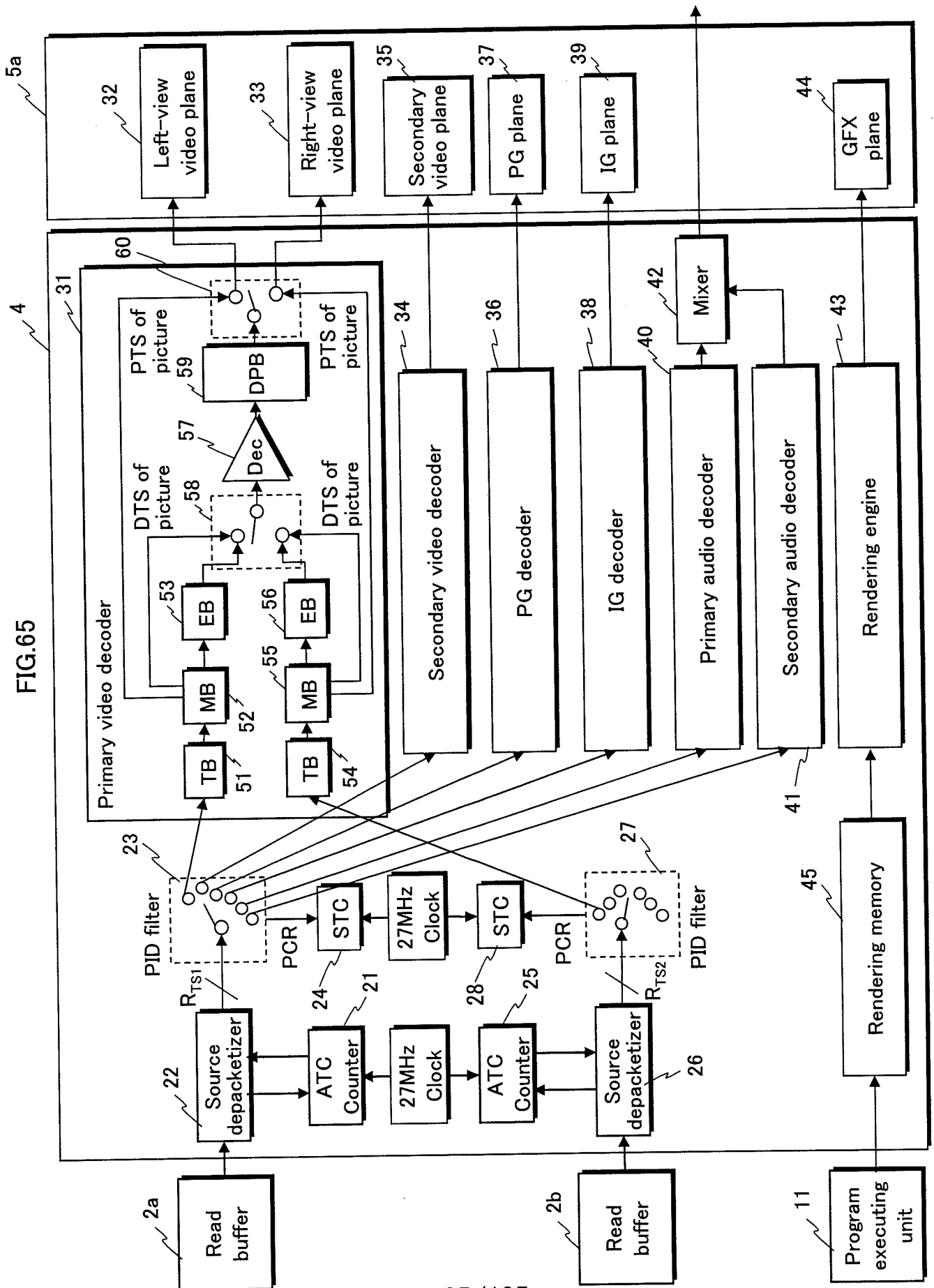


FIG. 66

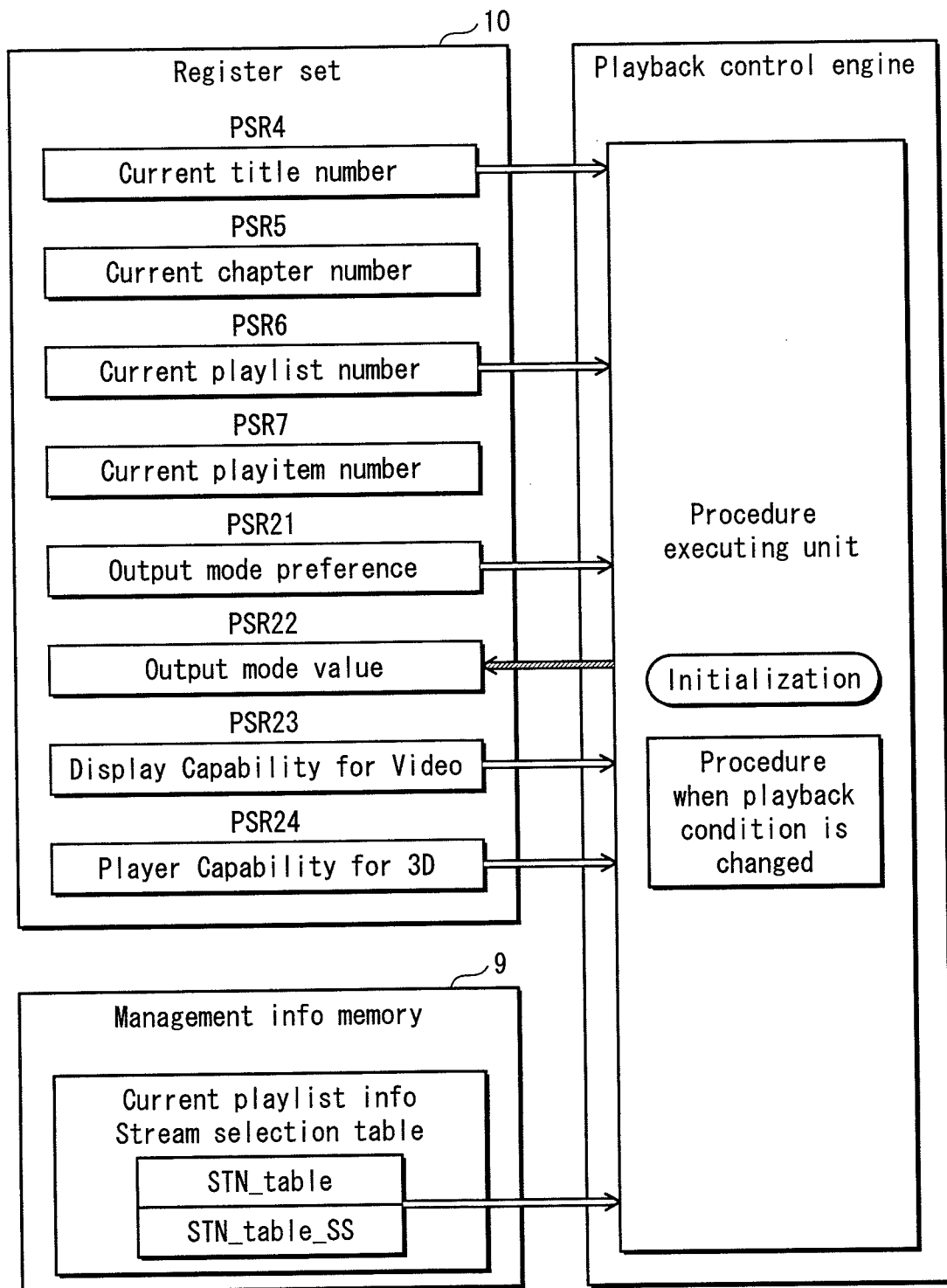


FIG. 67

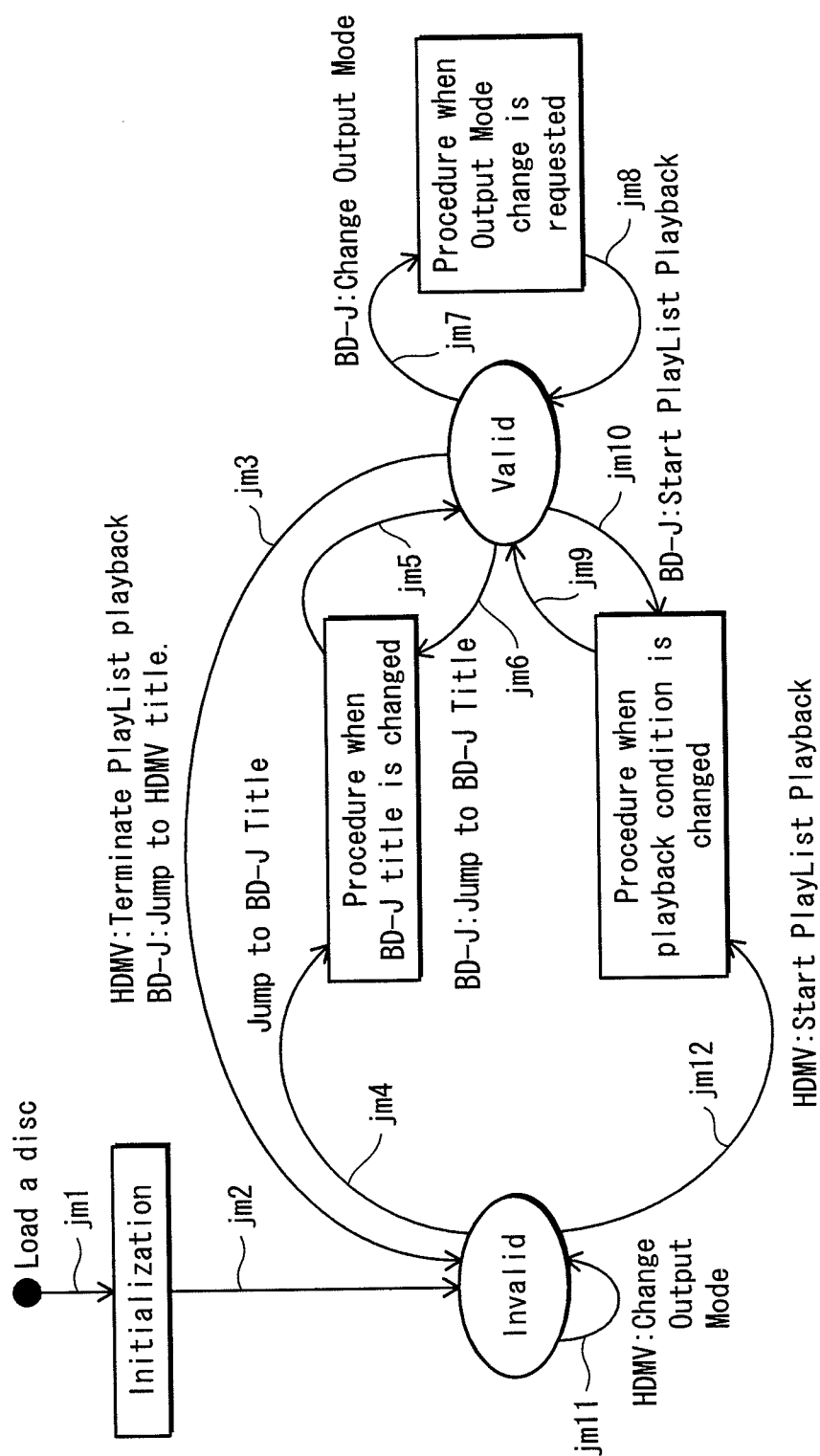


FIG. 68

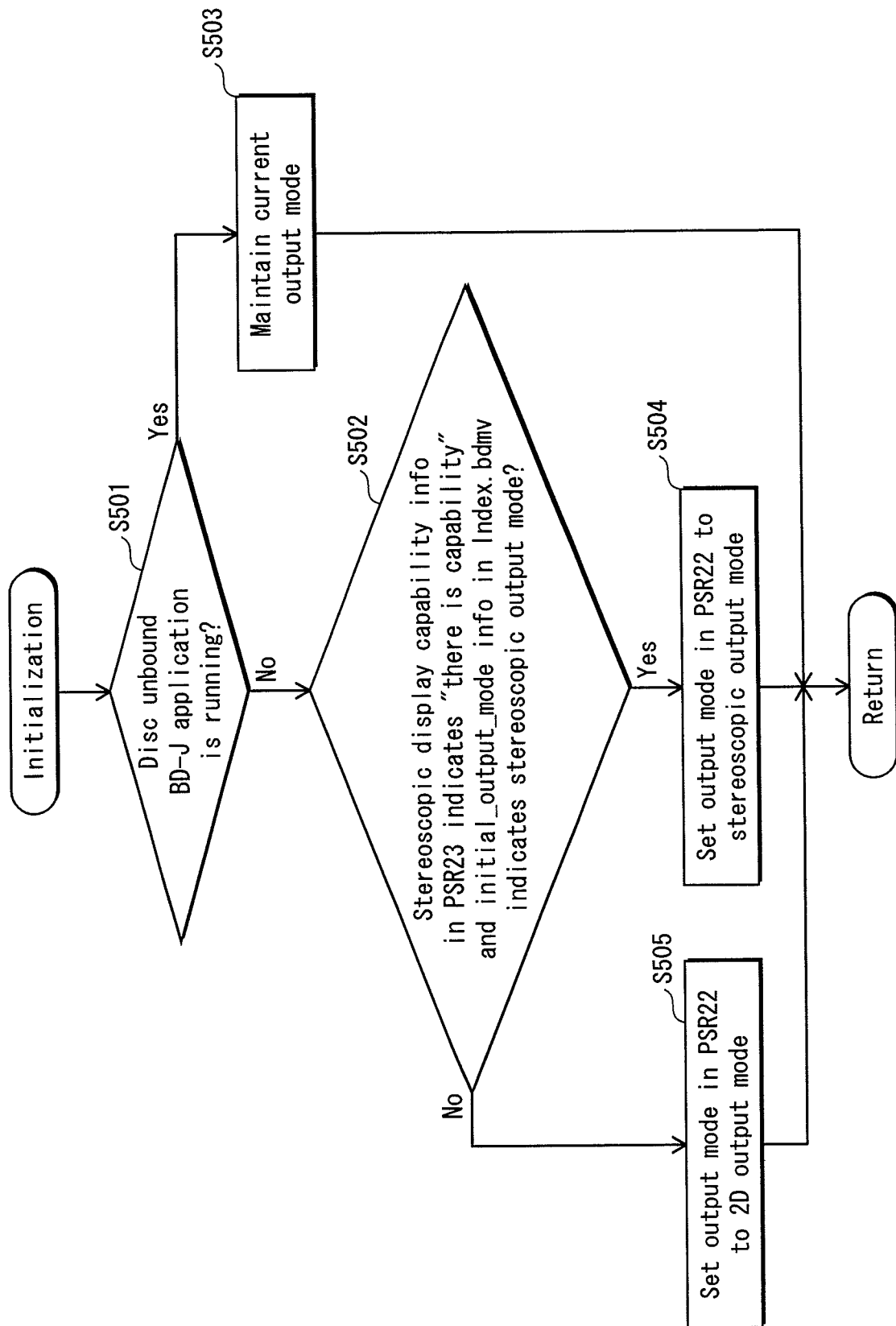


FIG. 69

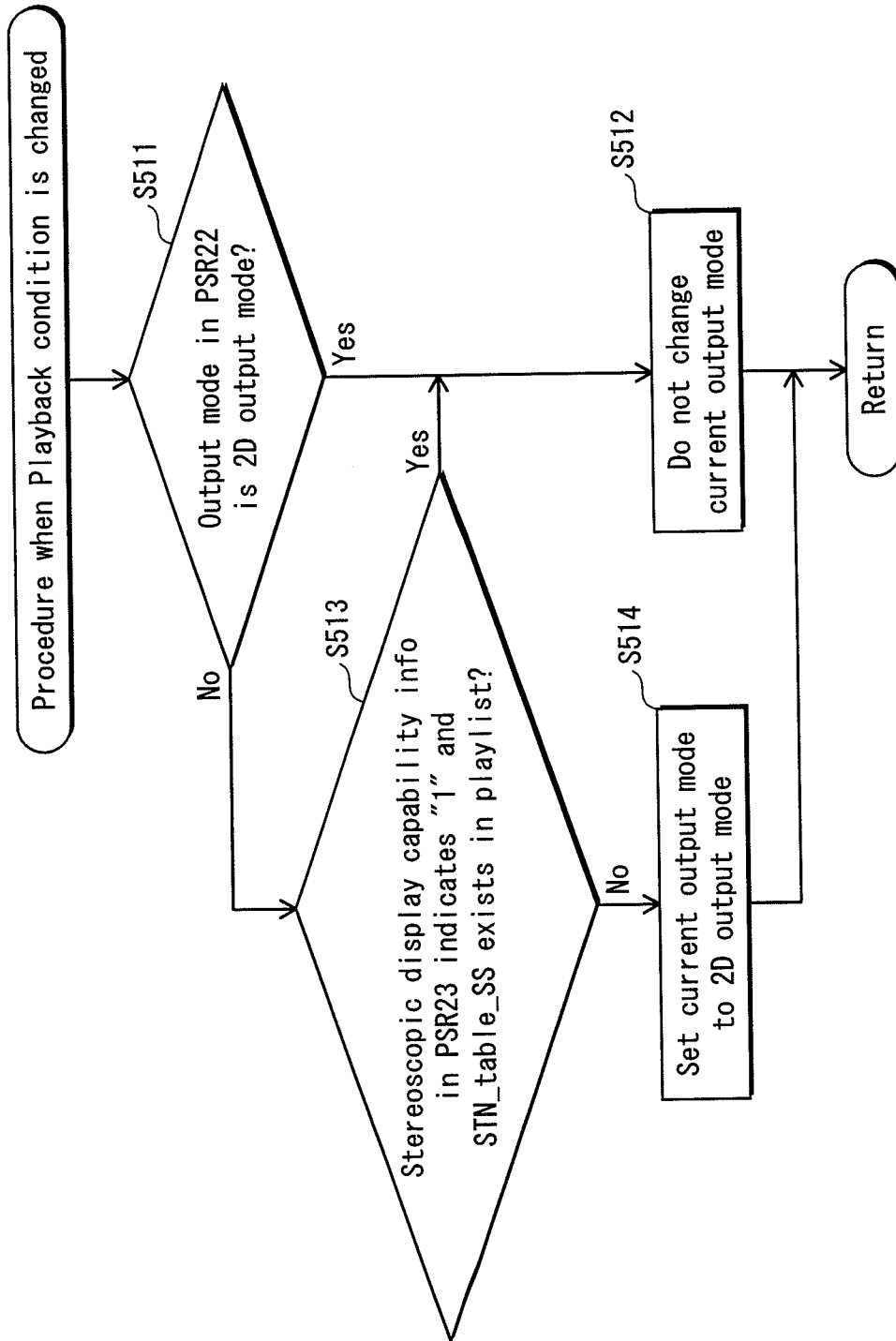


FIG. 70A PSR21:Output Mode Preference

b31	b30	b29	b28	b27	b26	b25	b24
reserved							
b23	b22	b21	b20	b19	b18	b17	b16
reserved							
b15	b14	b13	b12	b11	b10	b9	b8
reserved							
b7	b6	b5	b4	b3	b2	b1	b0
reserved							Output Mode Preference

Output Mode Preference ... 0b:2D Output Mode
1b:Stereoscopic Output Mode

FIG. 70C PSR23:Display Capability

b31	b30	b29	b28	b27	b26	b25	b24
reserved							
b23	b22	b21	b20	b19	b18	b17	b16
reserved							
b15	b14	b13	b12	b11	b10	b9	b8
reserved							
b7	b6	b5	b4	b3	b2	b1	b0
							Stereoscopic Display Capability

Stereoscopic Display Capability ... Connected TV system is
0b:Stereoscopic presentation incapable
1b:Stereoscopic presentation capable

FIG. 70B PSR22:Stereoscopic status

b31	b30	b29	b28	b27	b26	b25	b24
reserved							
b23	b22	b21	b20	b19	b18	b17	b16
reserved							
b15	b14	b13	b12	b11	b10	b9	b8
reserved							
b7	b6	b5	b4	b3	b2	b1	b0
reserved							Output Mode

Output Mode ... 0b:2D Output Mode
1b:Stereoscopic Output Mode

FIG. 70D PSR24:3D Capability

b31	b30	b29	b28	b27	b26	b25	b24
reserved							
b23	b22	b21	b20	b19	b18	b17	b16
reserved							
b15	b14	b13	b12	b11	b10	b9	b8
reserved							
b7	b6	b5	b4	b3	b2	b1	b0
reserved							Stereoscopic Display Capability

Stereoscopic Display Capability ... 0b:Stereoscopic presentation incapable
1b:Stereoscopic presentation capable

FIG.71

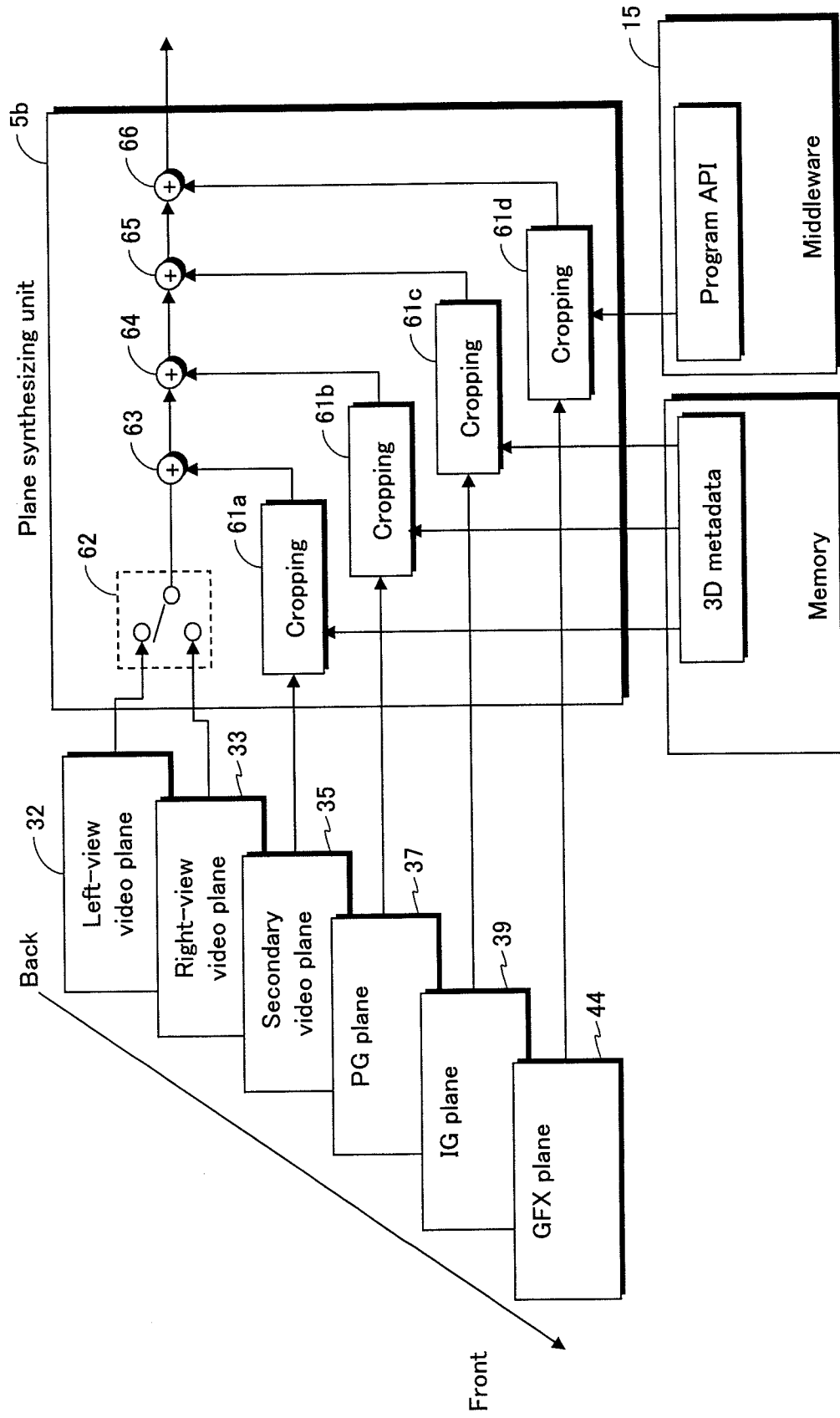


FIG.72A

	2D disc	3D disc (2D compatible)	3D disc (3D-only)
2D player	2D playback	2D playback	x
3D-support player	2D playback	3D playback (switchable to 2D playback)	3D playback (2D displayable)

FIG.72B

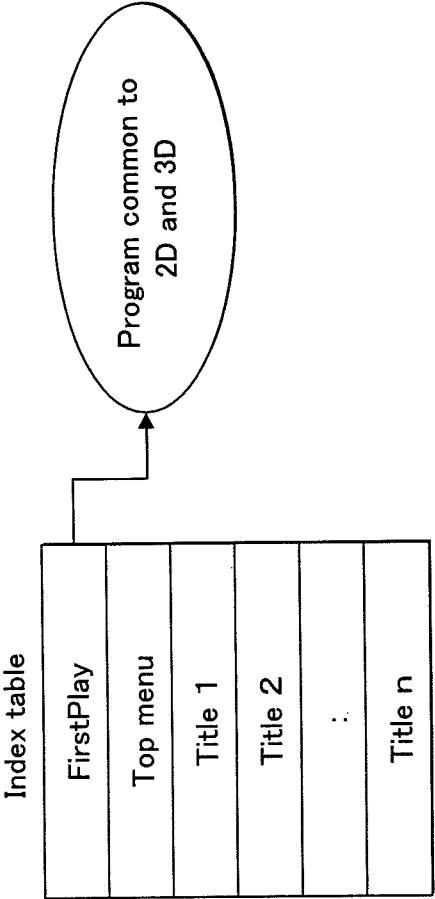
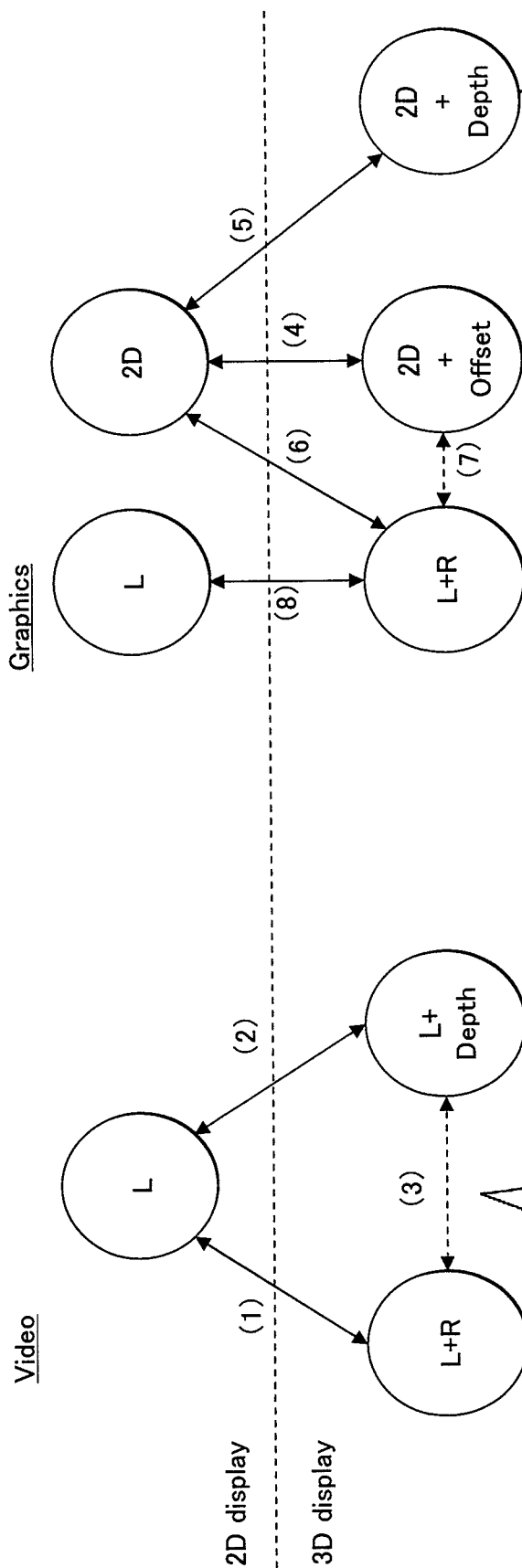


FIG.73

Switch between 2D and 3D



Video: When primary and secondary videos and background image are to be played back:

- (1) Switching is made at the output.
- (2) Switching is made at the output.
- (3) Decoders are switched because decode-target streams are switched. Seamless switch is not guaranteed.

Graphics: When PG, IG, and text subtitle are to be played back:

- (4) Plane offset switch
- (5) Switching is made at the output.
- (6) Decode-target streams are switched. Seamless switch is not guaranteed. Playback of AV is interrupted when preloaded streams are switched.
- (7) Decode-target streams are switched. Seamless switch is not guaranteed.
- (8) When switch between 2D and 3D is made during display of LR, switch between L and LR is made. Decode-target streams are not switched.

FIG.74

System parameter (player variable)
Player Capability for Video

b31	b30	b29	b28	b27	b26	b25	b24
reserved							
b23	b22	b21	b20	b19	b18	b17	b16
reserved							
b15	b14	b13	b12	b11	b10	b9	b8
reserved							
b7	b6	b5	b4	b3	b2	b1	b0
reserved						Dependent View Capability	Depth Capability

FIG.75

System parameter (player variable)
3D Capability

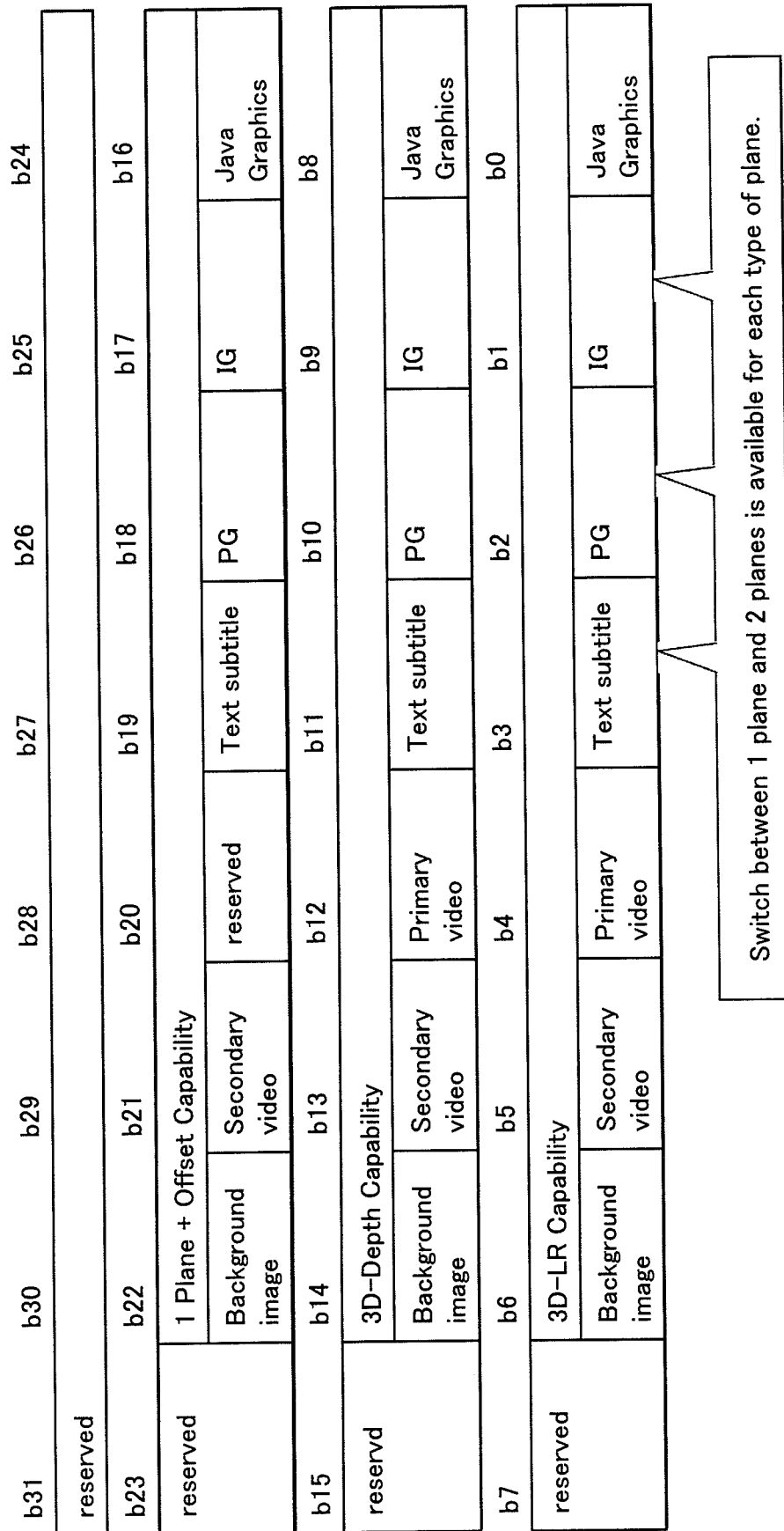


FIG.76

System parameter (player variable)
Corresponding database identification information

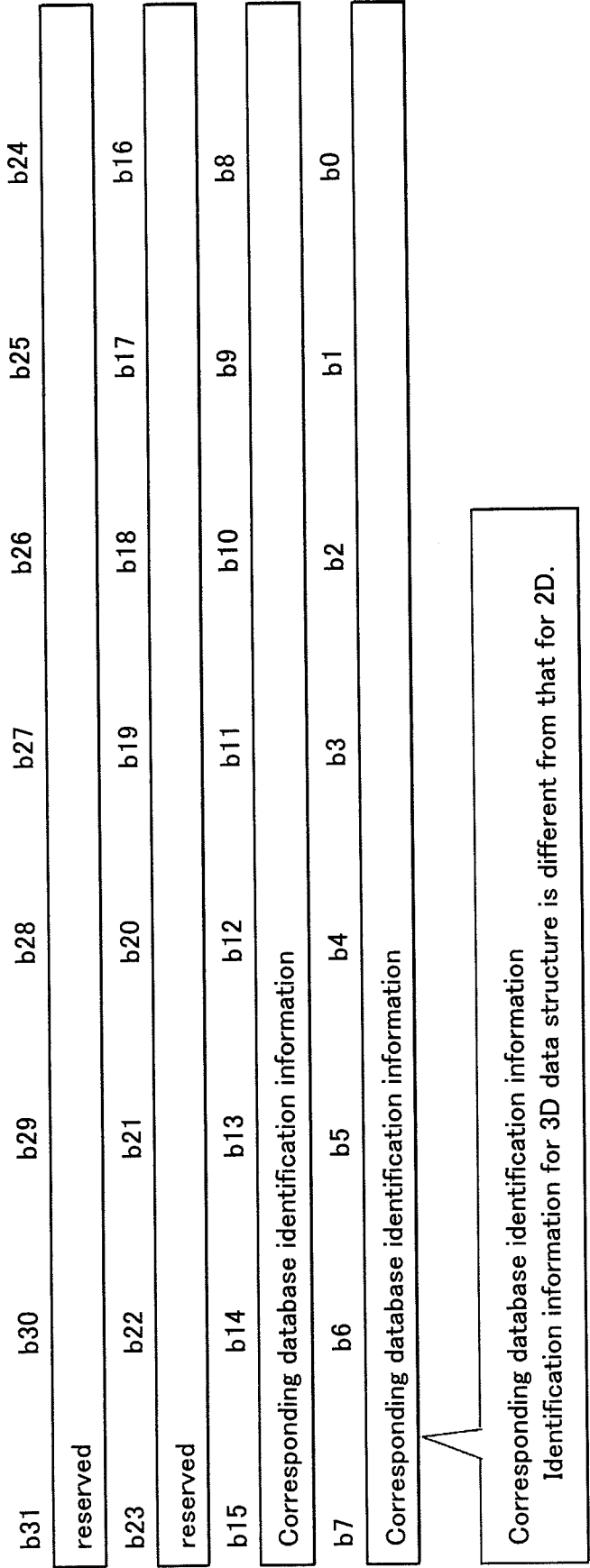


FIG.77

System parameter (player variable)
3D Presentation Preference

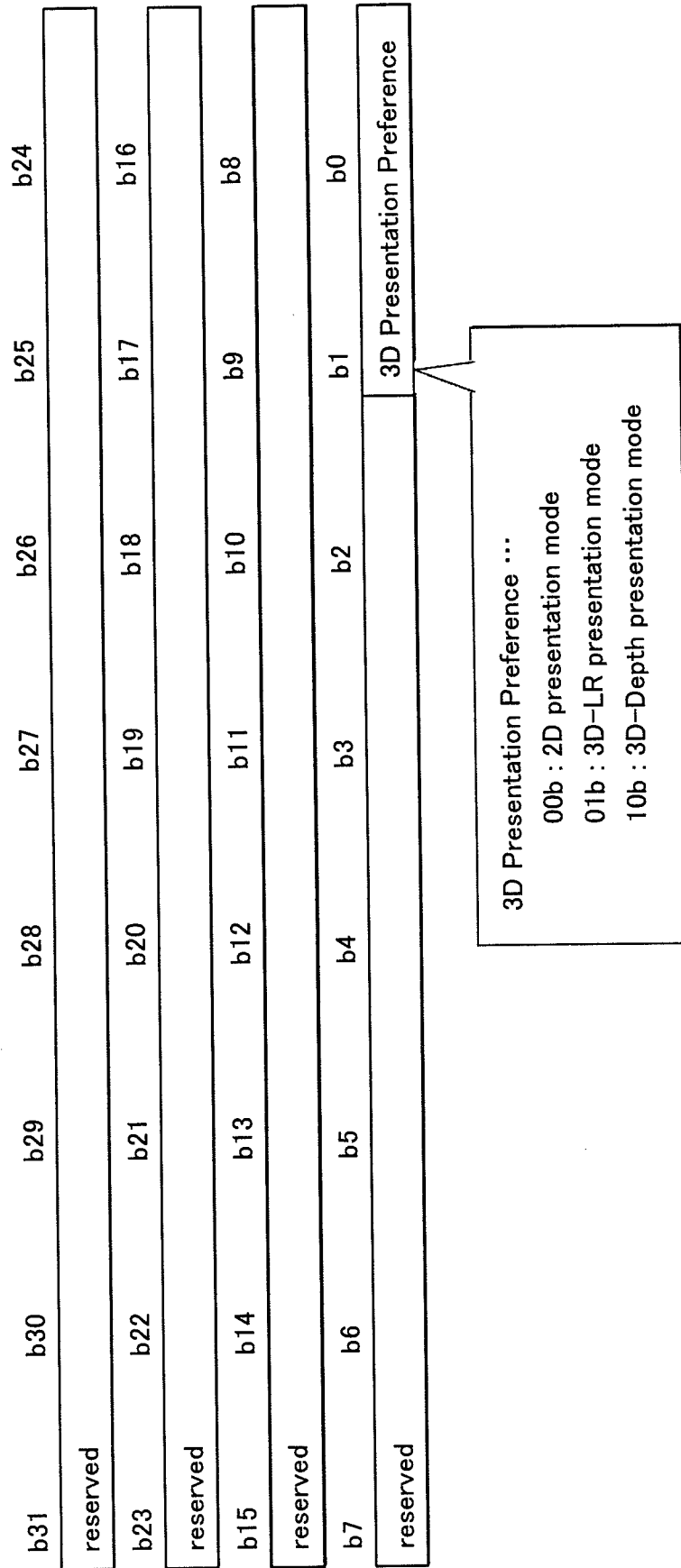


FIG.78

System parameter (player variable)
3D Presentation Type

b31	b30	b29	b28	b27	b26	b25	b24
reserved	Background image		Primary video		Secondary video		
b23	b22	b21	b20	b19	b18	b17	b16
Text subtitle		PG	IG		Java Graphics		
b15	b14	b13	b12	b11	b10	b9	b8
reserved							
b7	b6	b5	b4	b3	b2	b1	b0
reserved							3D Presentation Type

3D Presentation Type ...

00b : 2D presentation mode

01b : 3D-LR presentation mode

10b : 3D-Depth presentation mode

It can represent 3D playback method for each stream or plane (b31-b16).

FIG.79

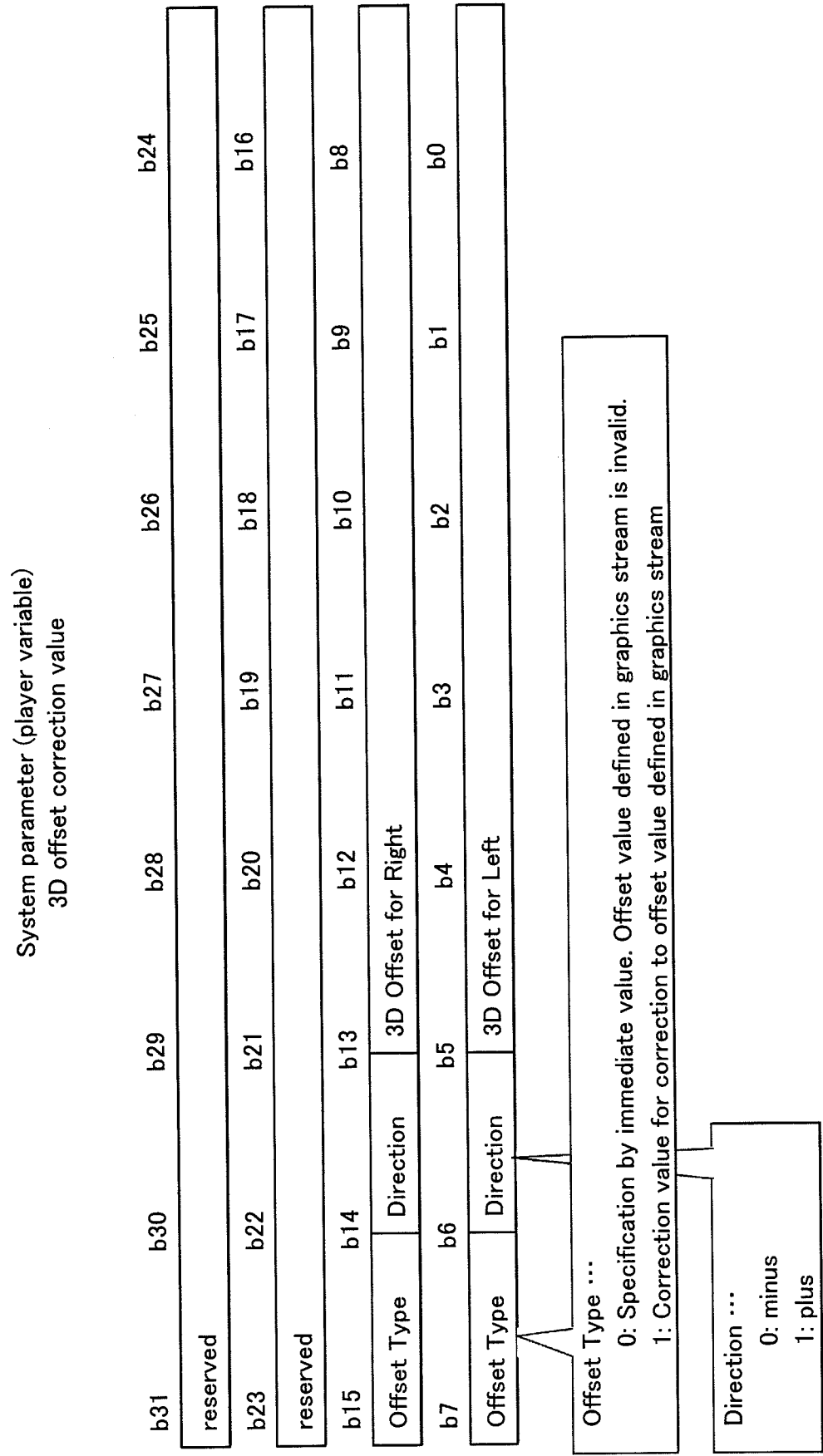
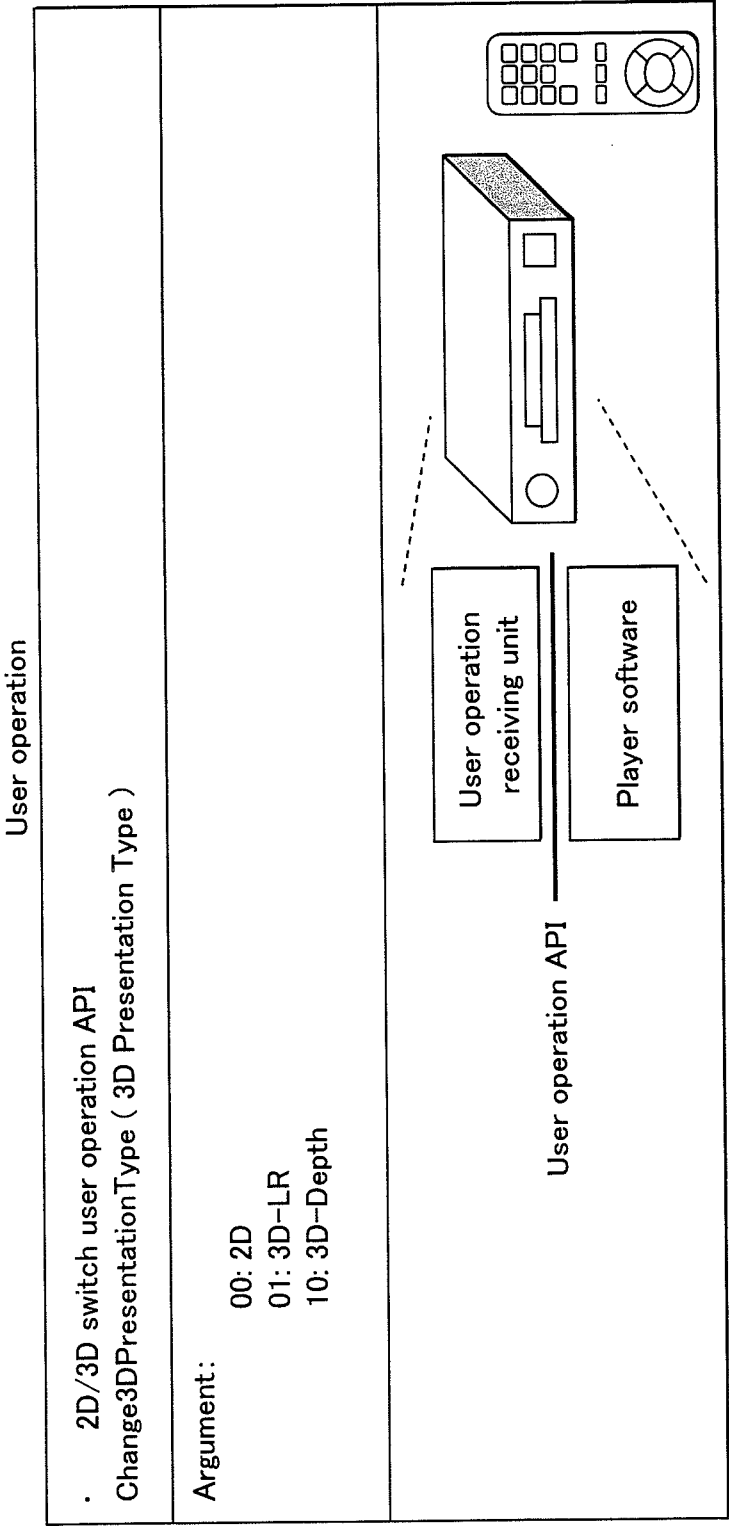


FIG.80A

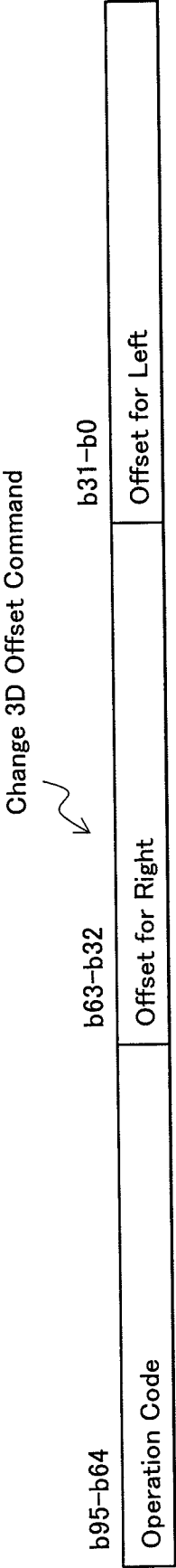


UO_mask_table

FIG.80B

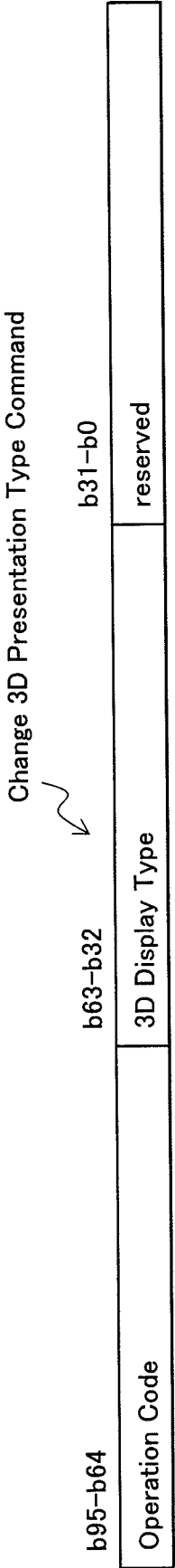
Syntax
UO_mask_table() {
...
Change3DPresentationType_mask
reserved
...

FIG.81



- 2D/3D switch command
- 2D, 1 plane + Offset, 3D-LR, 3D-Depth
- [Functionality]
 - Set 3D offset to PSRxx
- [Operands]
 - Two Operands

FIG.82



- Graphics offset value setting command
- [Functionality]
 - Set 3D Presentation Type to PSRxx
- [Operands]
 - One Operand

Data arrangement

FIG.83A

Segmenting Clip for each of L, R, D

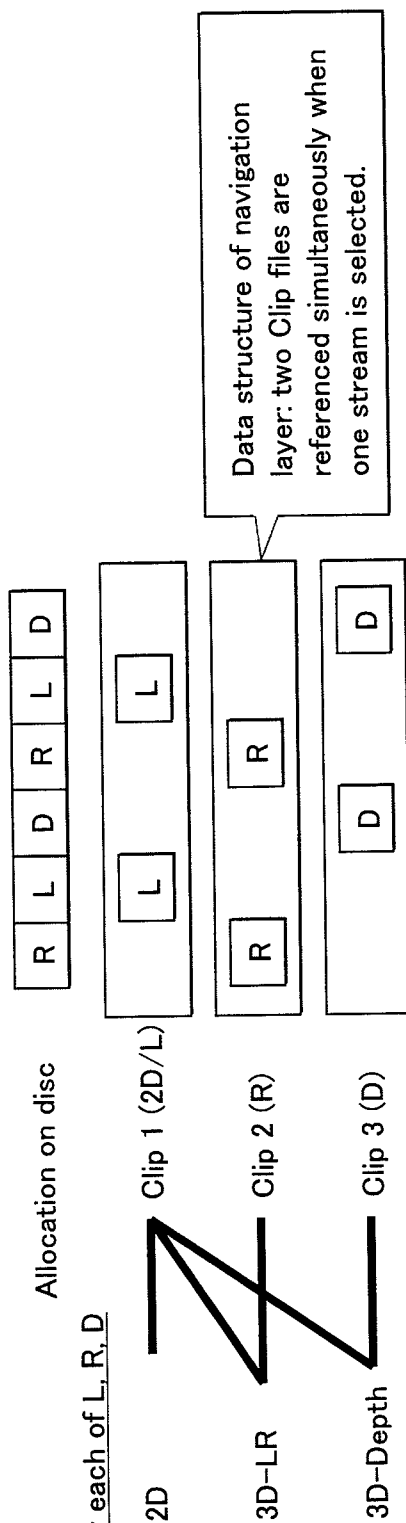


FIG.83B

Structuring Clip for each of 2D, 3D(LR), 3D(LD)

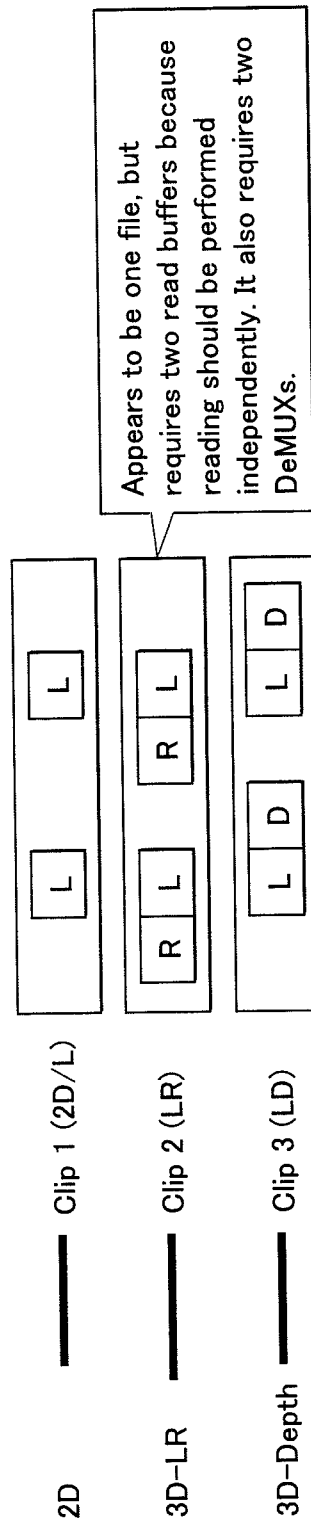


FIG.83C

Multiplexing into one Clip

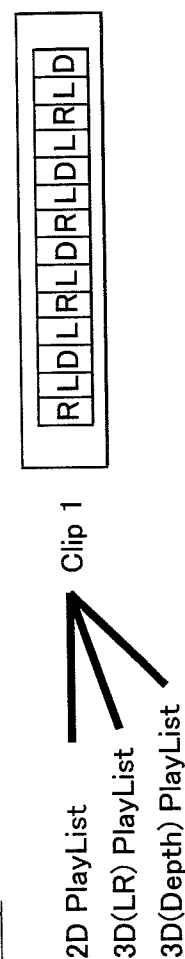


FIG.84

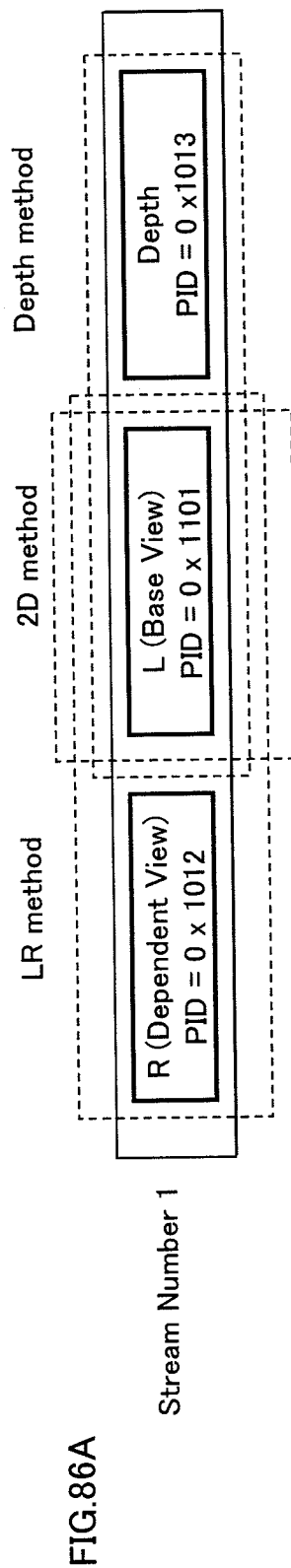
Multiplexing (TS level)

	Clip1 (2D/L)	Clip2 (R)	Clip3 (D)	1 Clip
Primary video	L (0x1011)	R (0x1012)	D (0x1013)	LRD
Primary audio	all	x	x	all
PG	2D	L,R (when necessary)	D	CLRD
IG	2D	L,R (when necessary)	D	CLRD
Secondary video	2D	x	x	2D
Secondary audio	all	?	?	all

- Clips 1, 2, 3 are interleaved.
- 2D players play back only Clip 1.
- 3D-LR players play back Clips 1, 2.
- 3D-Depth players play back Clips 1, 3.

FIG.85

PID_value	meaning	PID_value	meaning
0x1012	Dependent View	0x1420	IG for Left of 3D-LR
0x1013	Dependent View, Depth	0x1440	IG for Right of 3D-LR
0x1220	PG for Left of 3D-LR	0x1460	IG for Depth of 3D-Depth
0x1240	PG for Right of 3D-LR	0x1801	3D Text Subtitle
0x1260	PG for Depth of 3D-Depth	0x1B20	Secondary Video Dependent View
		0x1B40	Secondary Video Dependent View, Depth

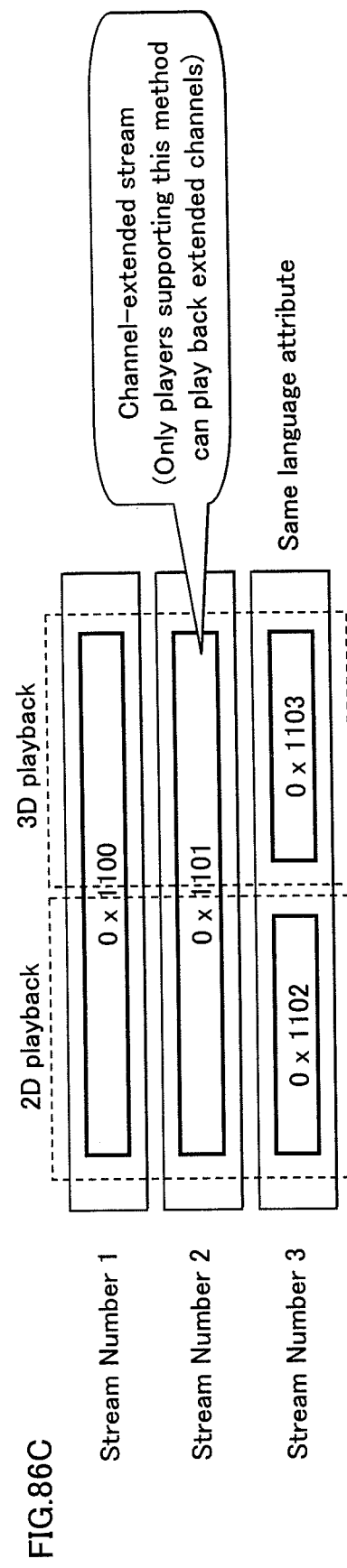


These combinations are treated as one set

Combinations for L/R/Depth

FIG.86B

Base View	Dependent View, Depth	Restriction
MPEG-4 AVC Base View	MPEG-4 AVC Dependent View	Cannot co-exist with secondary video
MPEG-2 Video	MPEG-2 Video	Cannot co-exist with secondary video
MPEG-4 AVC	MPEG-4 AVC	Cannot co-exist with secondary video
VC-1	VC-1	Cannot co-exist with secondary video



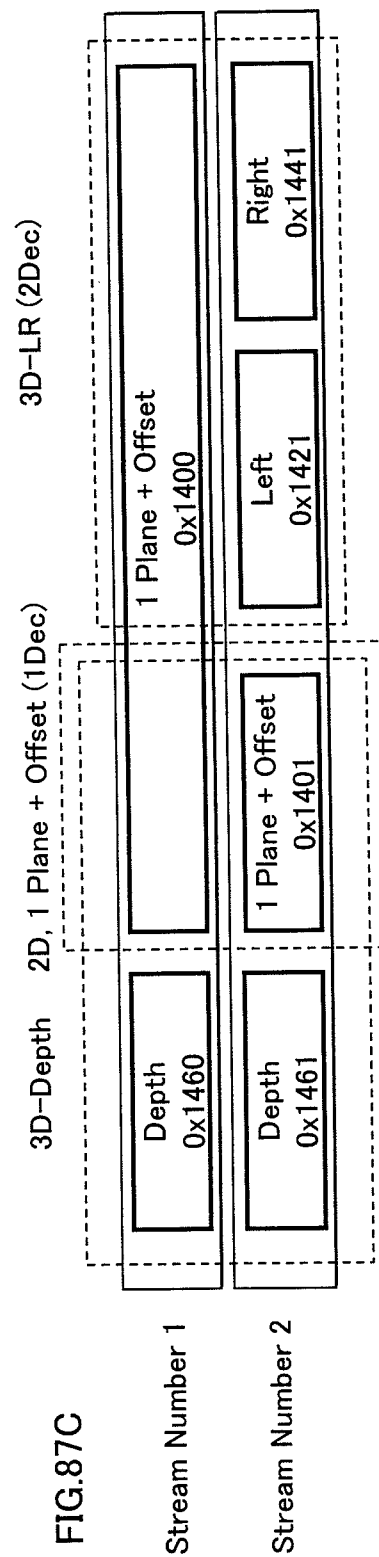
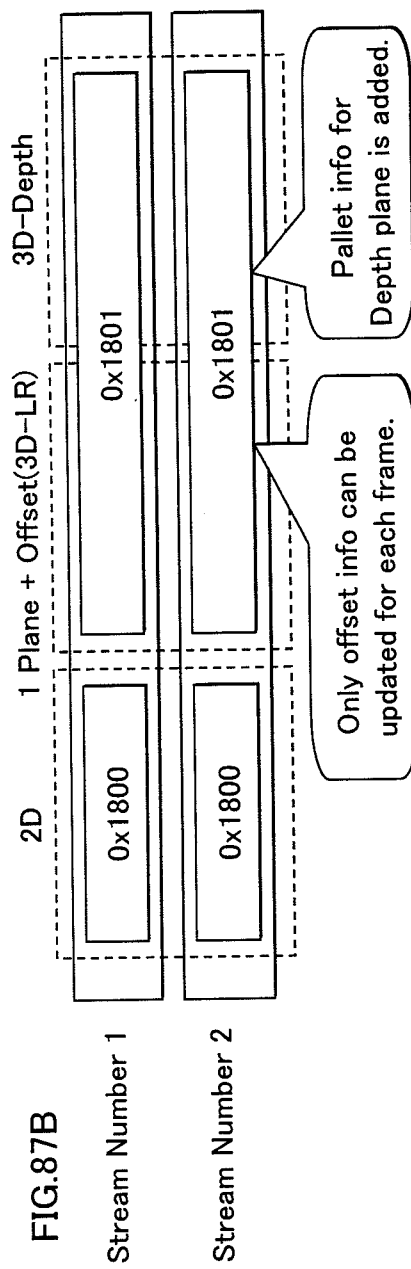
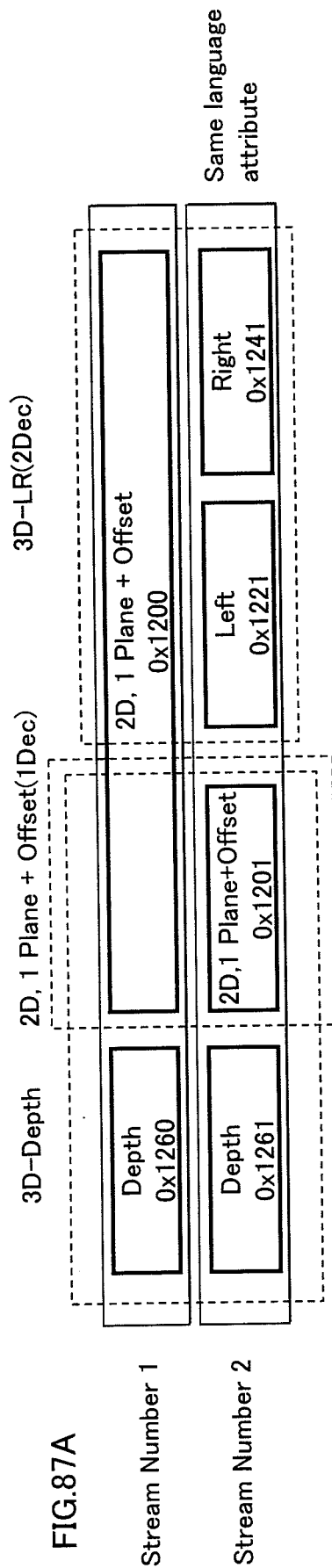


FIG.88A

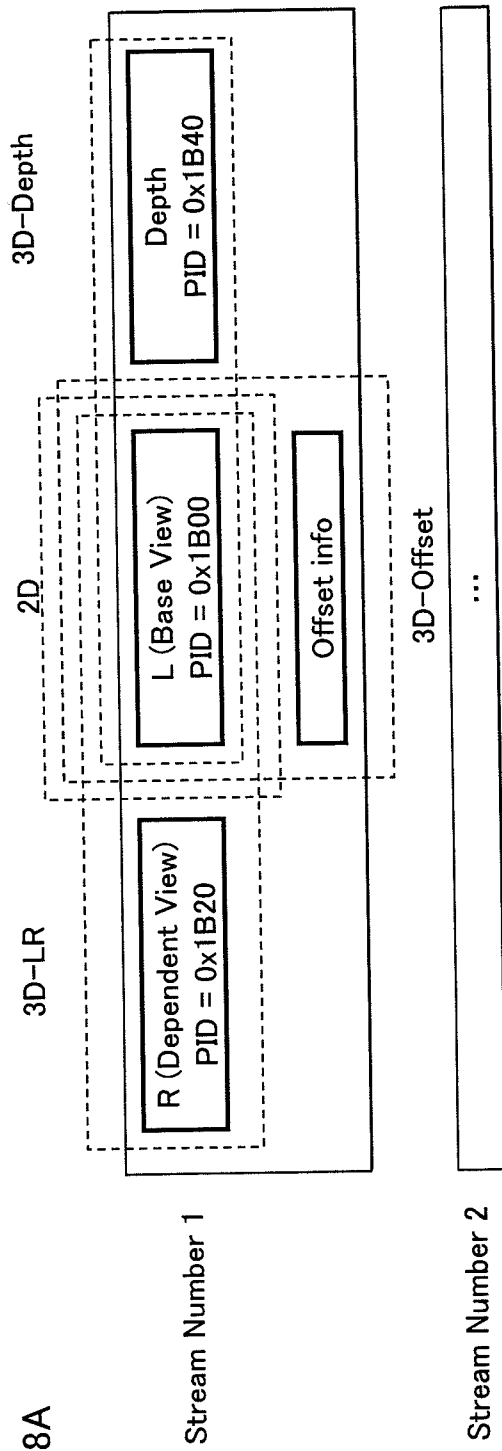
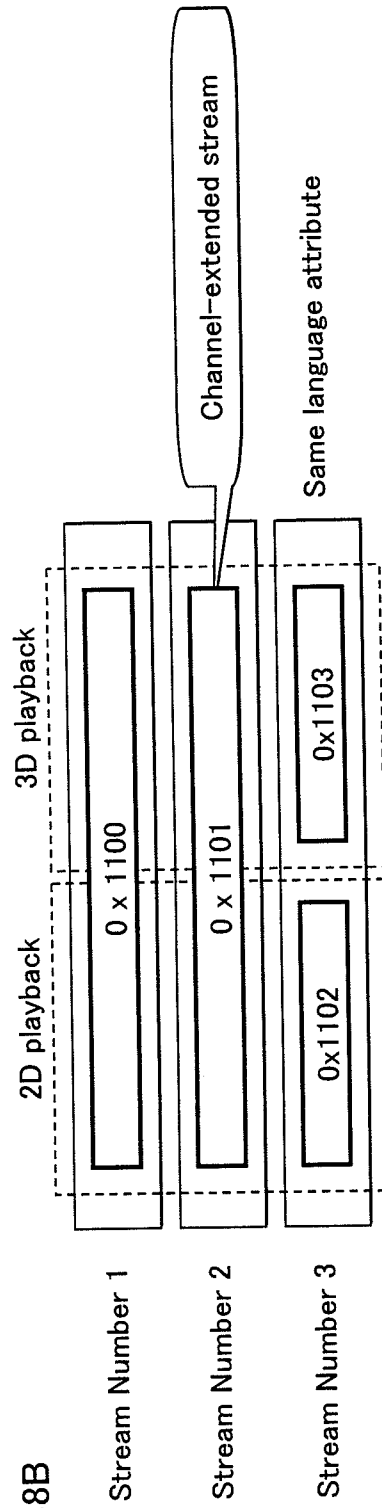


FIG.88B



Conditions for seamless connection

FIG.89A

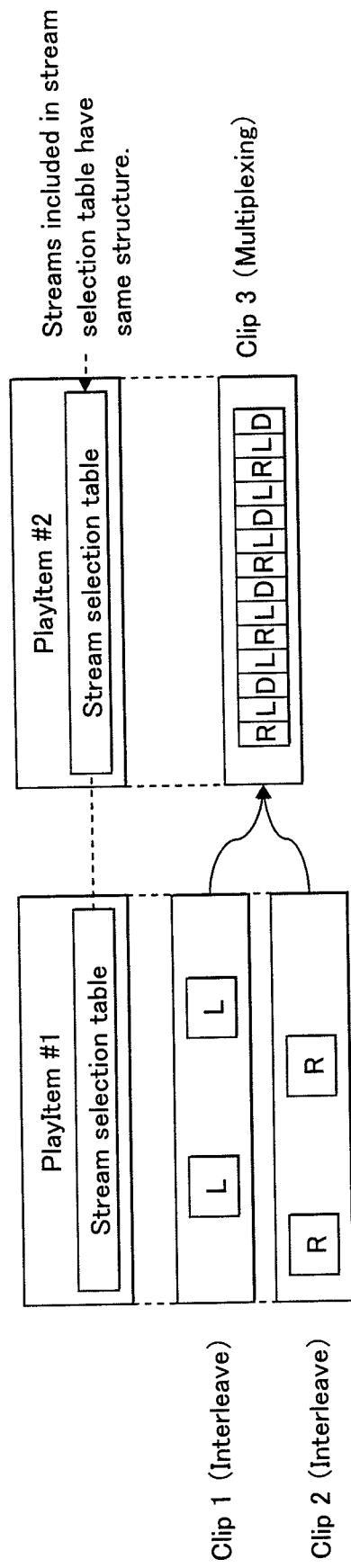


FIG.89B

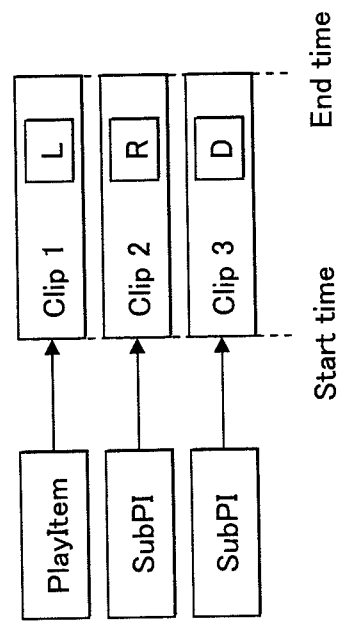


FIG.90A

SubPath

SubPath Type	Meaning
...	
8	Dependent View for LR
9	Dependent View for Depth
10	3D file to be referenced in layer-boundary jump instead of 2D file

FIG.90B

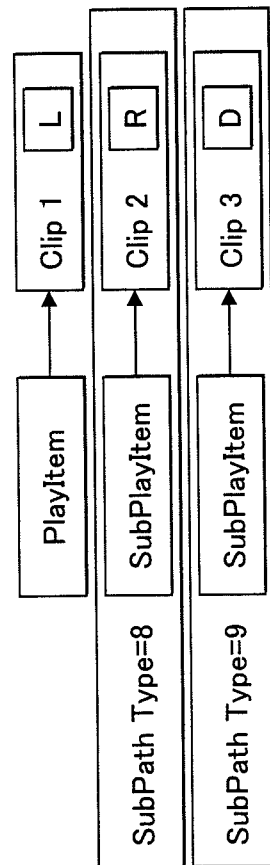


FIG.90C

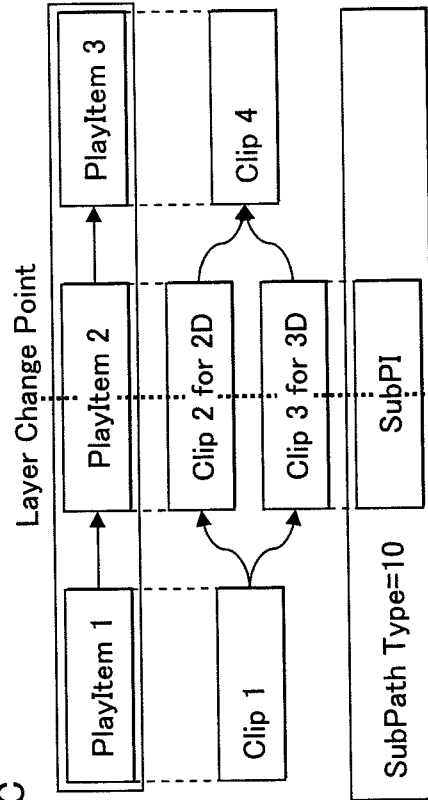


FIG.91

Syntax
STN_table_extention(){
length
for (Loop for repeating as many as primary video streams) {
dependent_view_is_available
depth_is_available
reserved
if (dependent_view_is_available) {
stream_entry()
stream_attribute()
}
if (depth_is_available) {
stream_entry()
stream_attribute()
}
}
for (Loop for repeating as many as primary audio streams) {
replace_3D_audio_stream
reserved
stream_entry()
stream_attribute()
}
...

FIG.92

Syntax
...
for (Loop for repeating as many as PG streams) {
offset_is_available
LR_streams_are_available
Depth_stream_is_available
reseved
if (offset_is_available) {
stream_entry()
stream_attribute()
}
if (LR_streams_are_available) {
stream_entry()for left
stream_entry()for right
stream_attribute()
}
if (Depth_stream_is_available) {
stream_entry()
stream_attribute()
}
}
...

FIG.93

Syntax
...
for (Loop for repeating as many as IG streams) {
offset_is_available
LR_streams_are_available
Depth_stream_is_available
reseved
if (offset_is_available) {
stream_entry()
stream_attribute()
}
if (LR_streams_are_available) {
stream_entry()for left
stream_entry()for right
stream_attribute()
}
if (Depth_stream_is_available) {
stream_entry()
stream_attribute()
}
}
...}

FIG.94

Syntax
for (Loop for repeating as many as secondary audio streams) {
replace_3D_audio_stream
reserved
stream_entry()
stream_attribute()
}
for (Loop for repeating as many as secondary video streams) {
dependent_view_is_available
Depth_is_available
reserved
if (dependent_view_is_available) {
stream_entry()
stream_attribute()
}
if (Depth_is_available) {
stream_entry()
stream_attribute()
}
}

FIG.95

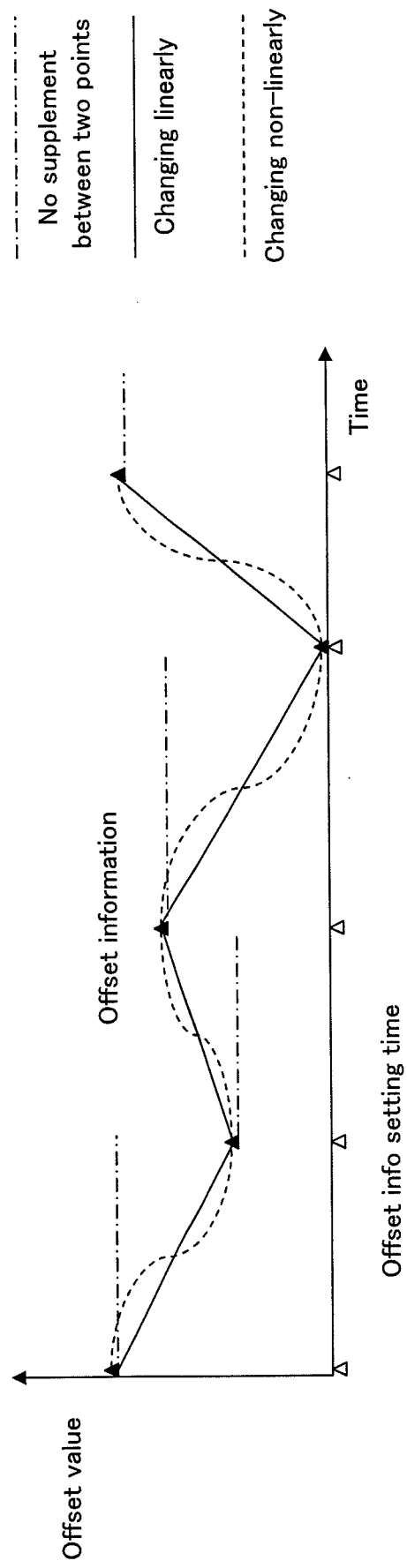
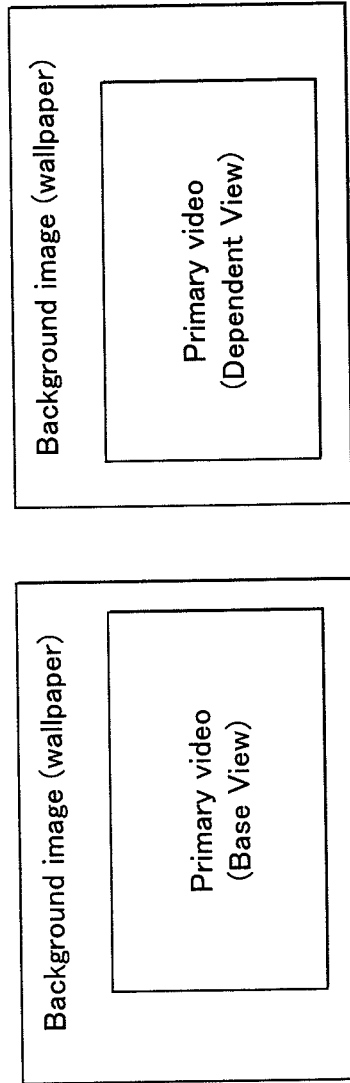


FIG.96



Only this one is displayed during 2D playback

FIG.97

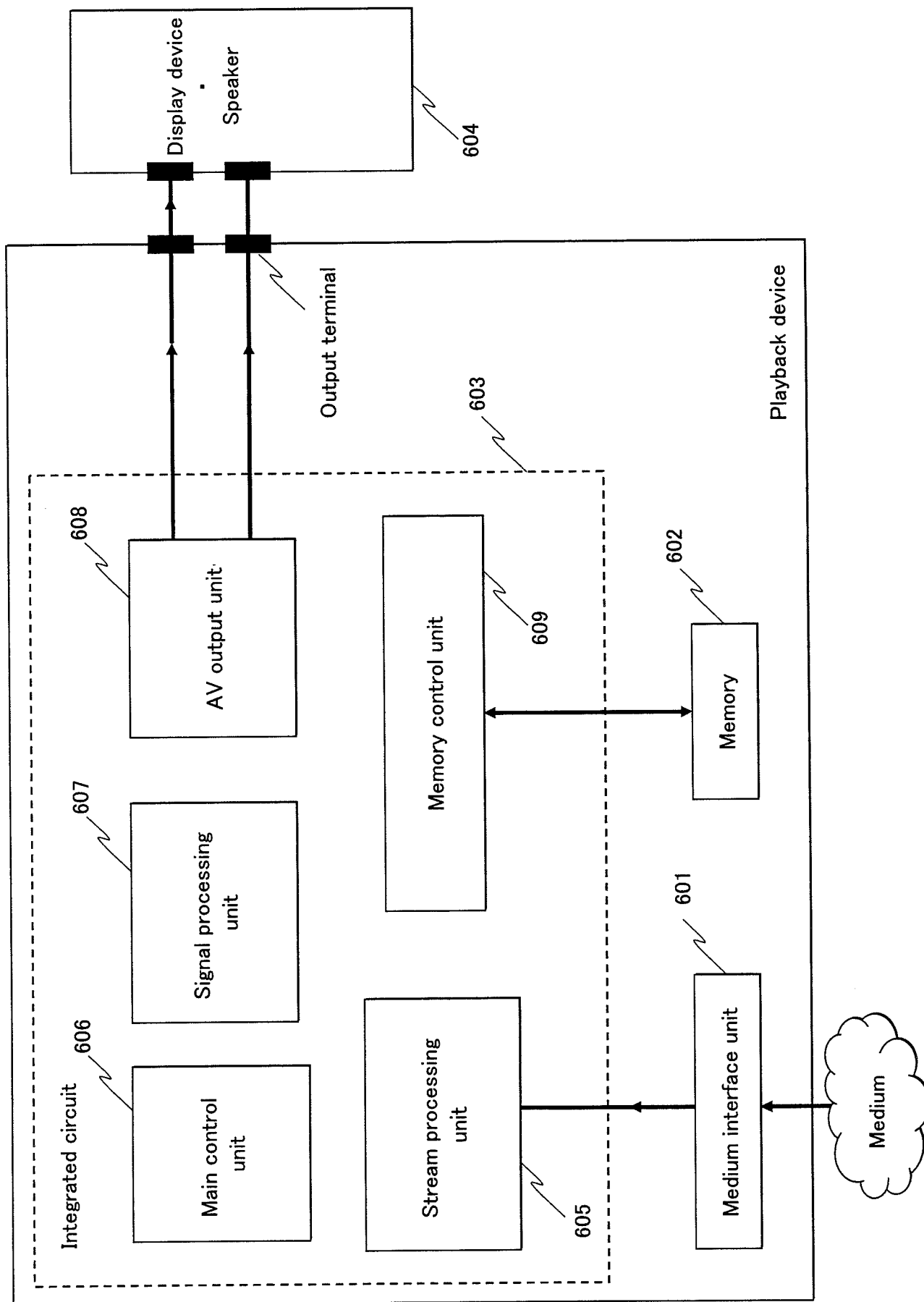


FIG.98

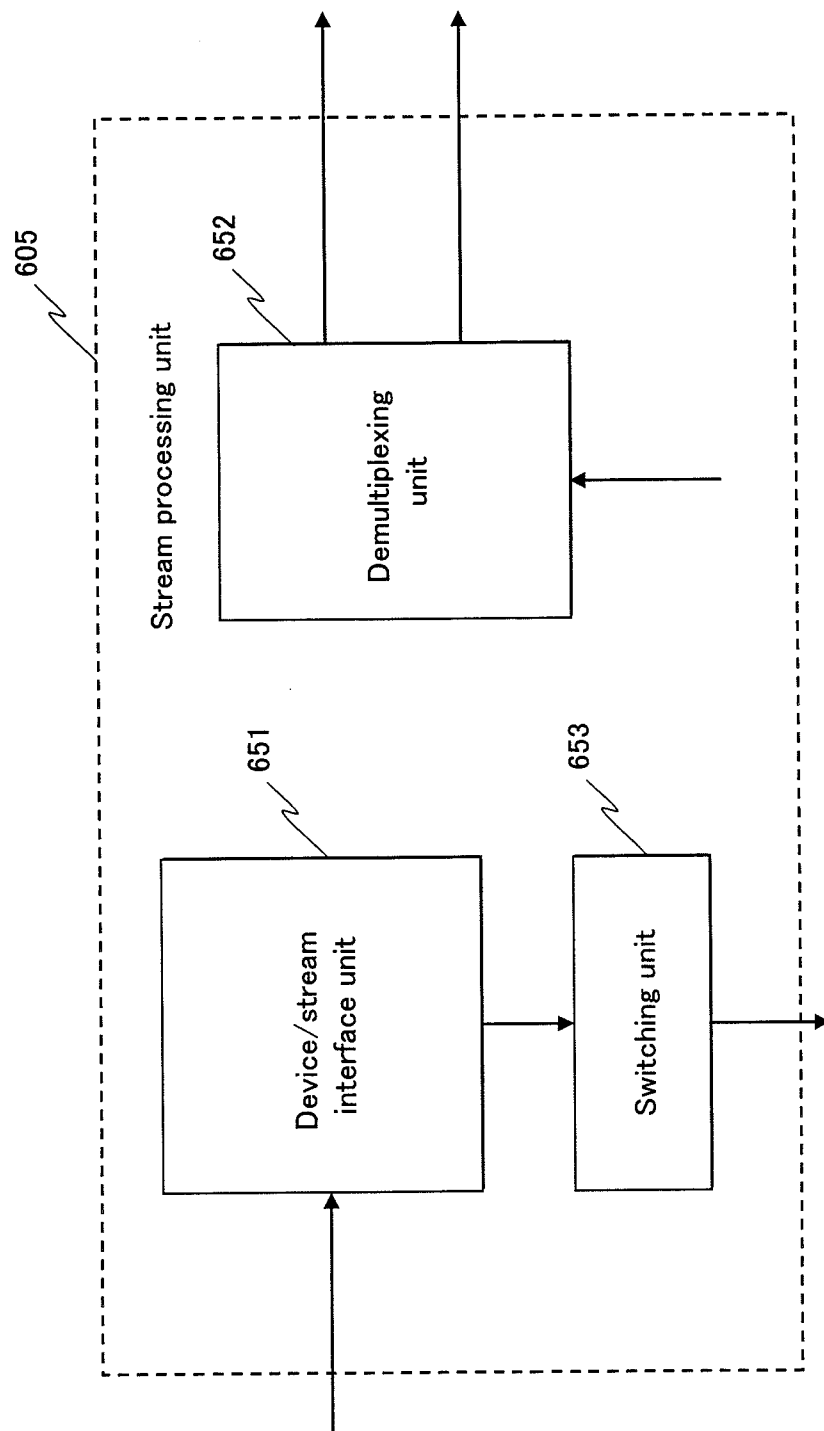


FIG.99

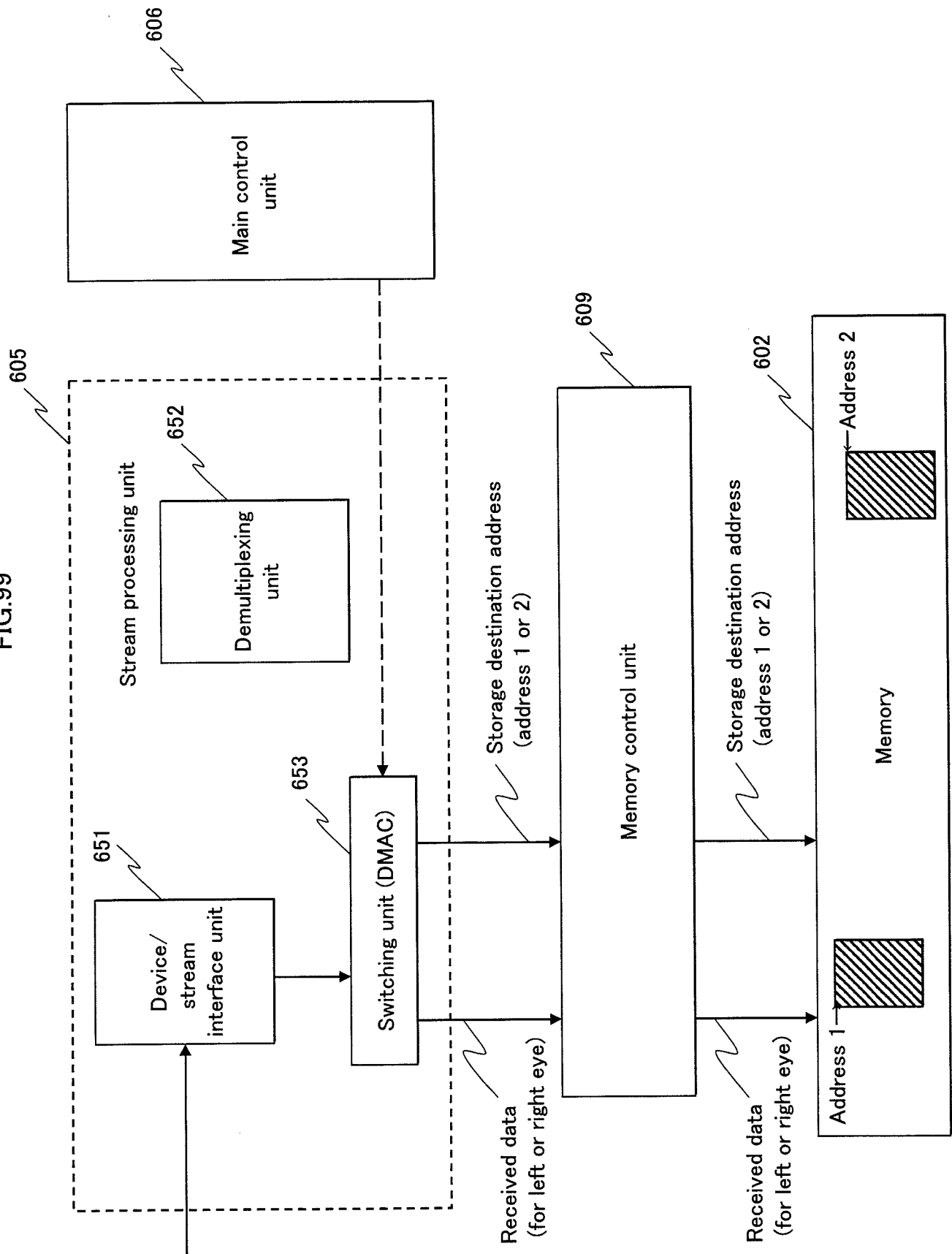
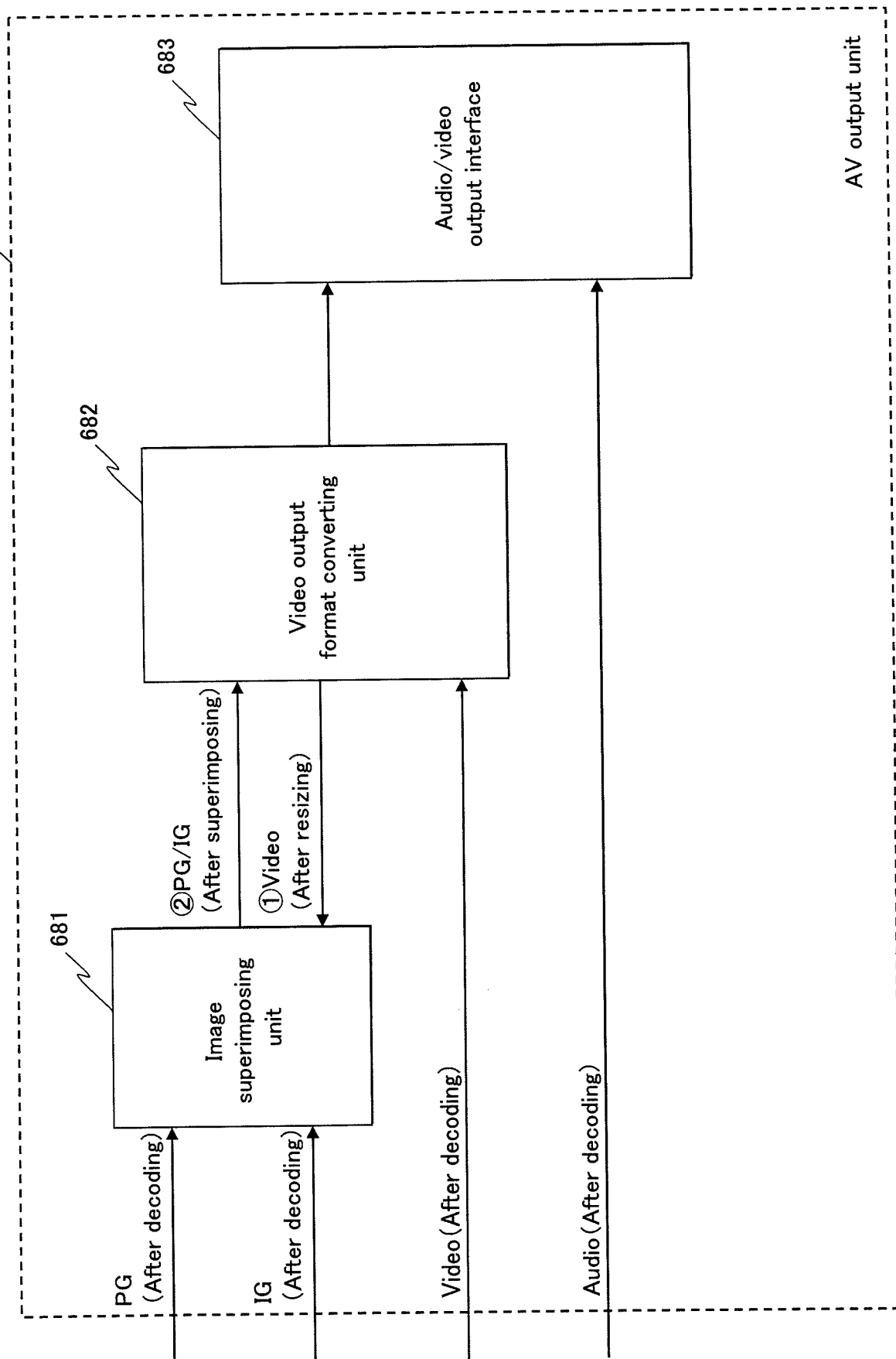


FIG.100

608



AV output unit

FIG.101

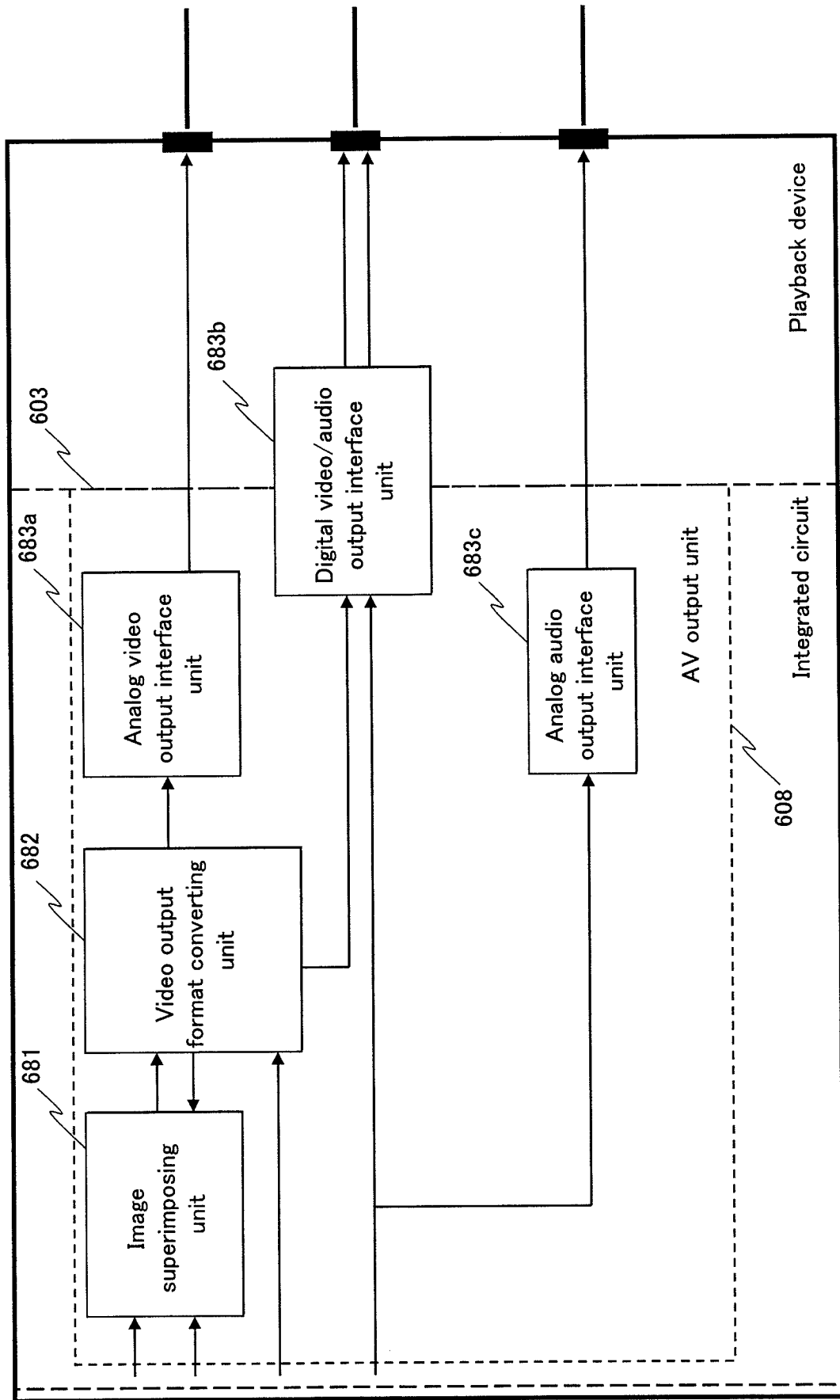


FIG.102

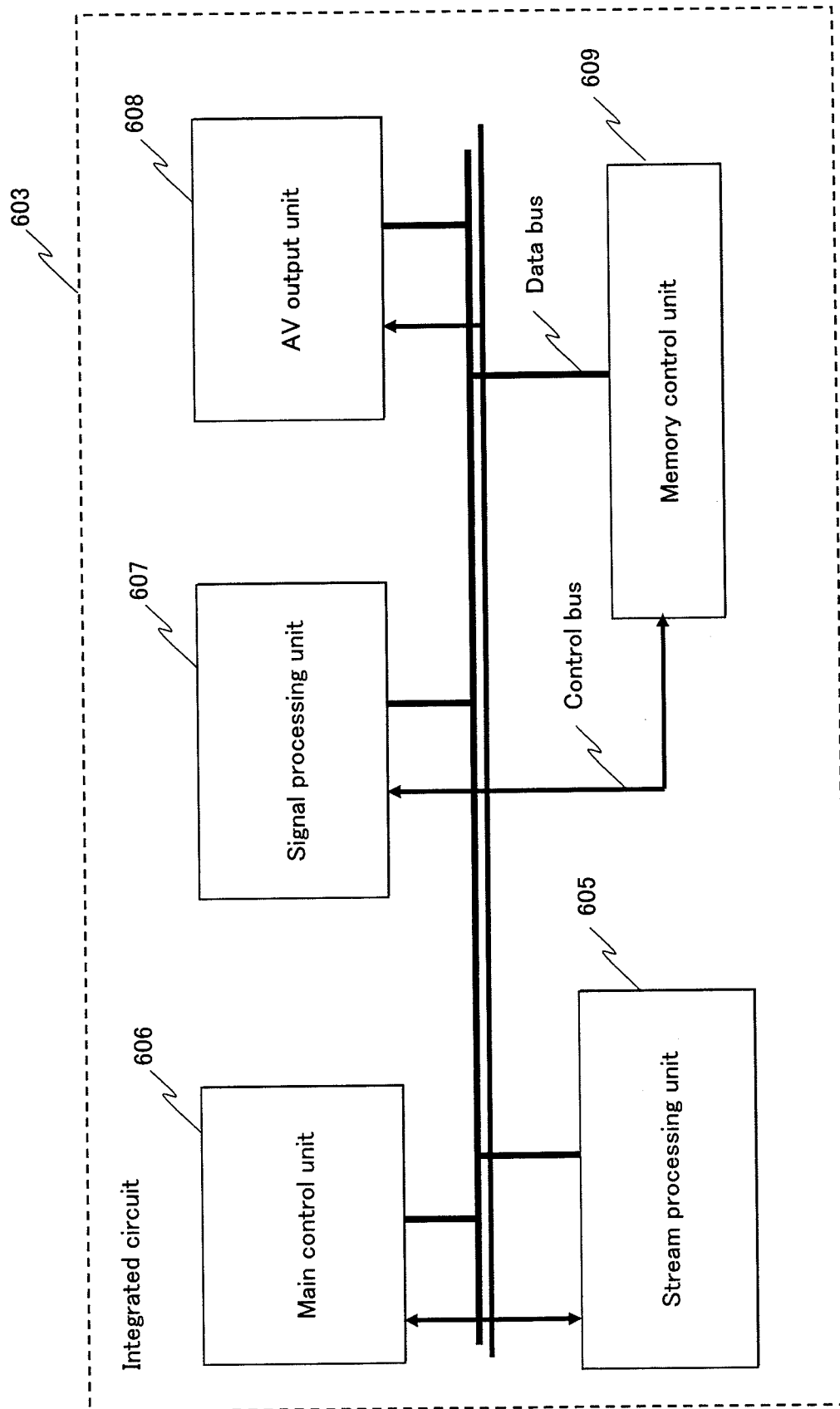


FIG.103

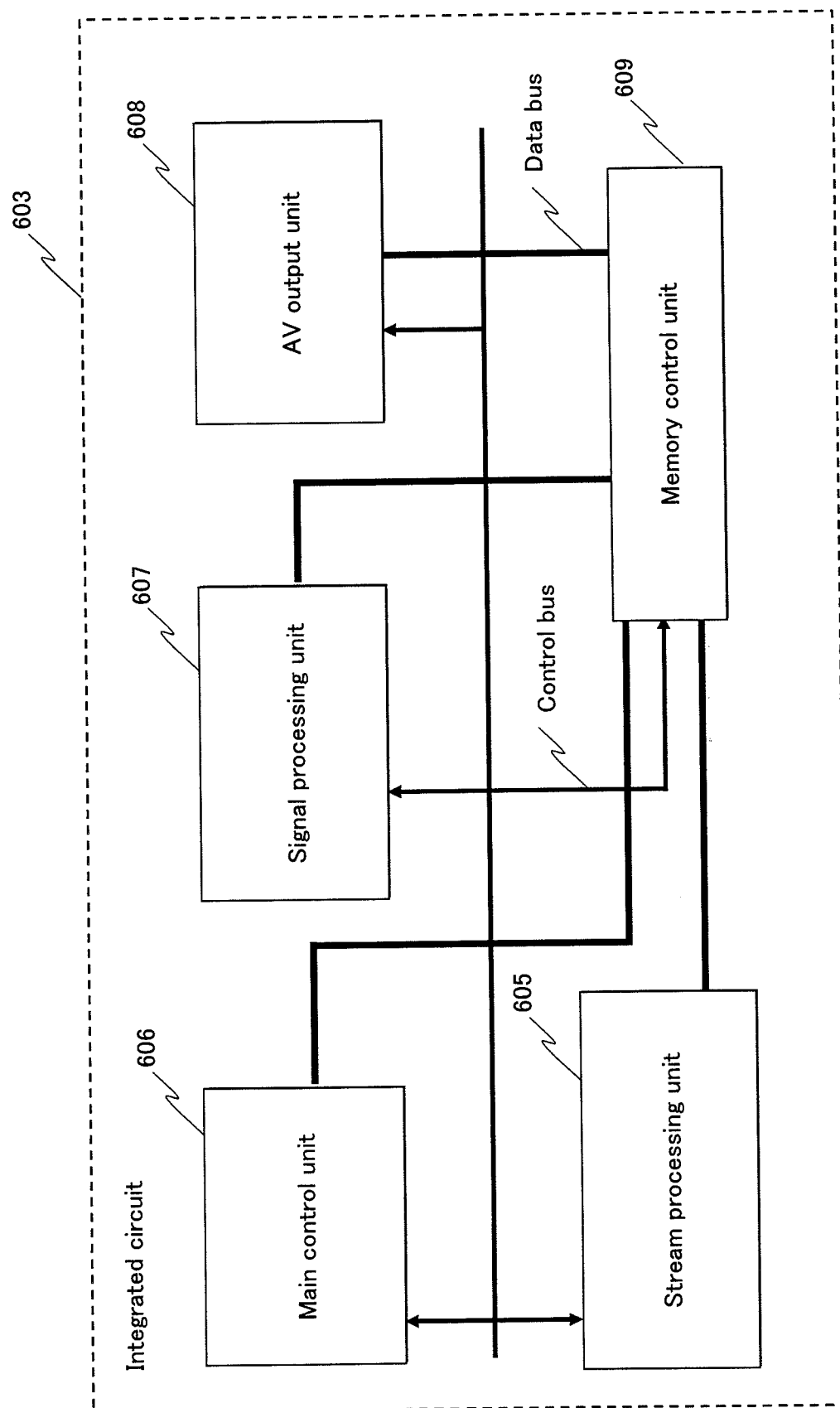


FIG.104

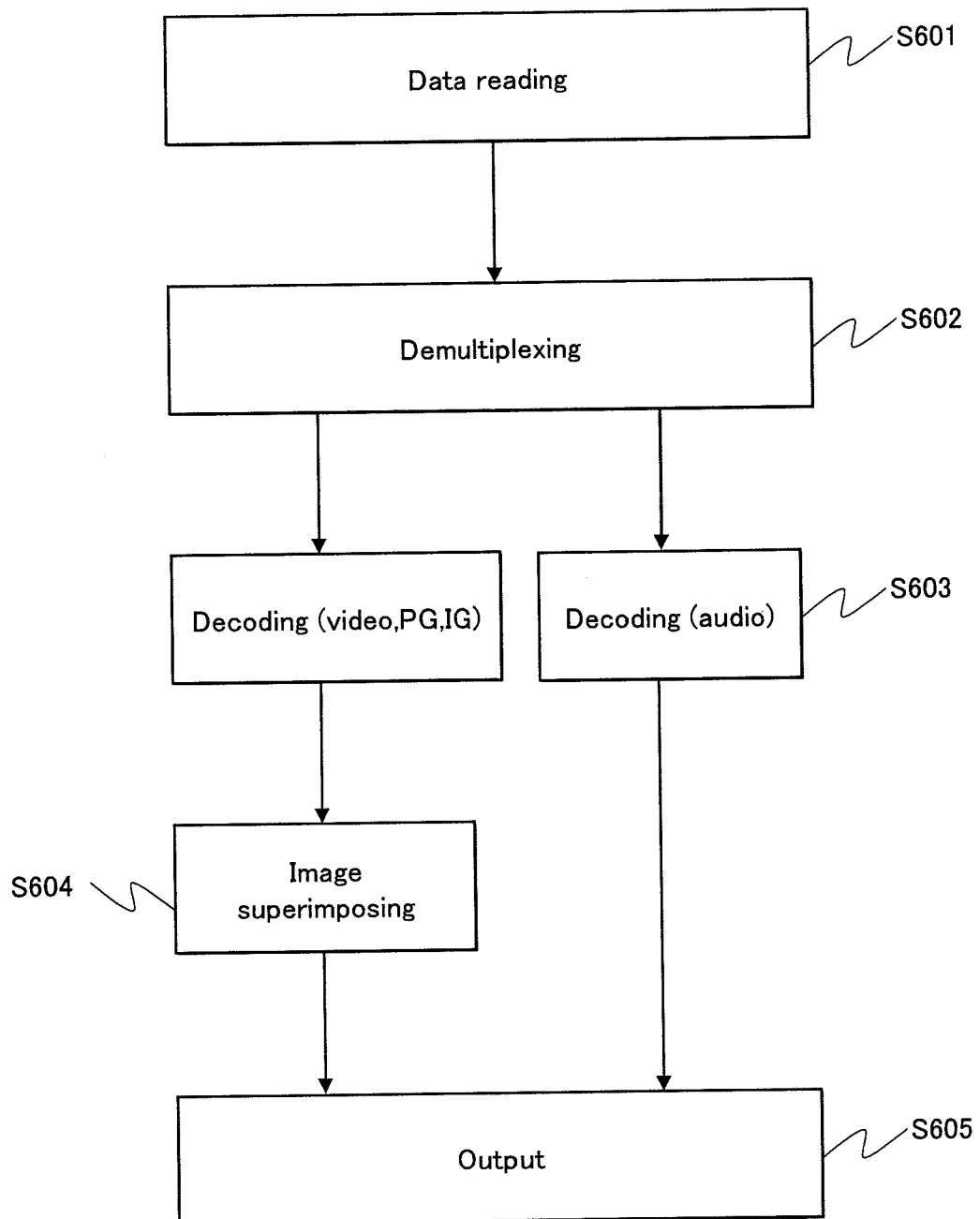


FIG.105

