

FIG. 1

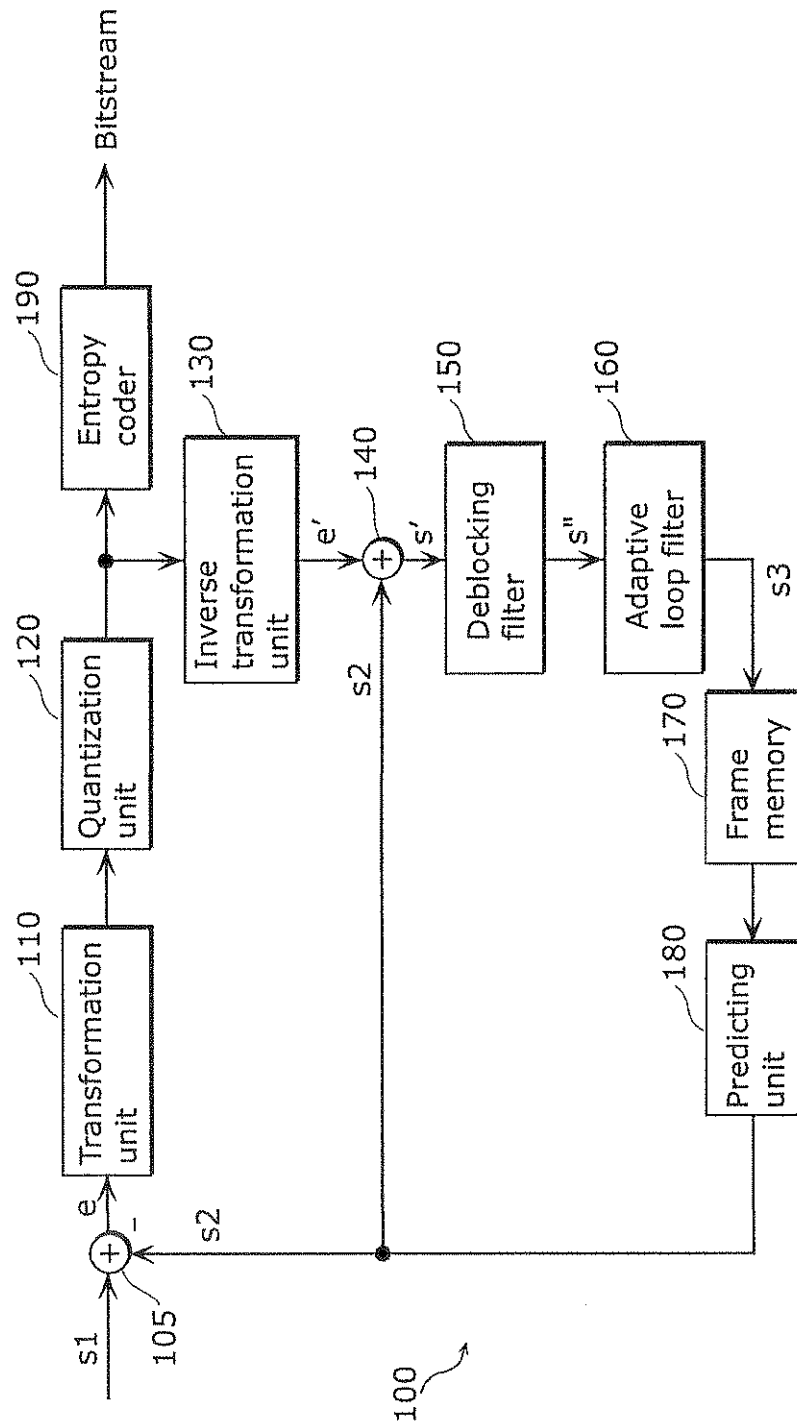


FIG. 2

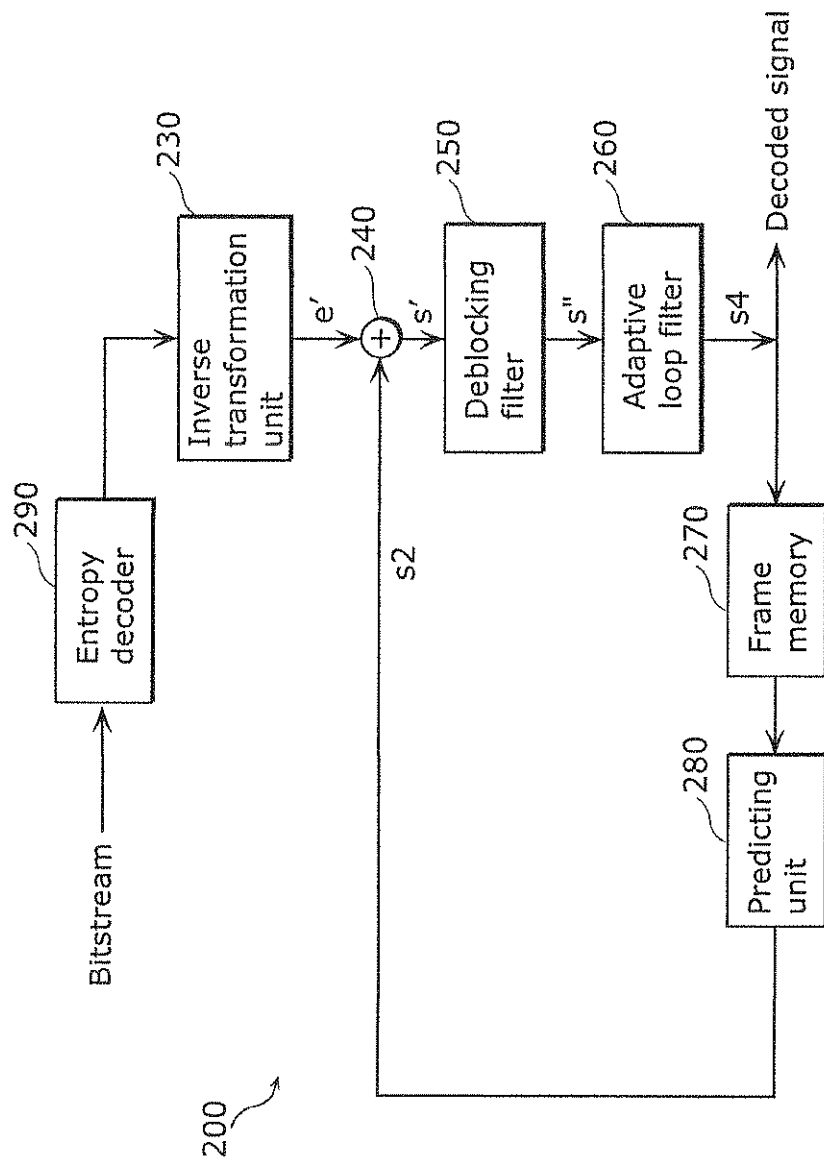


FIG. 3

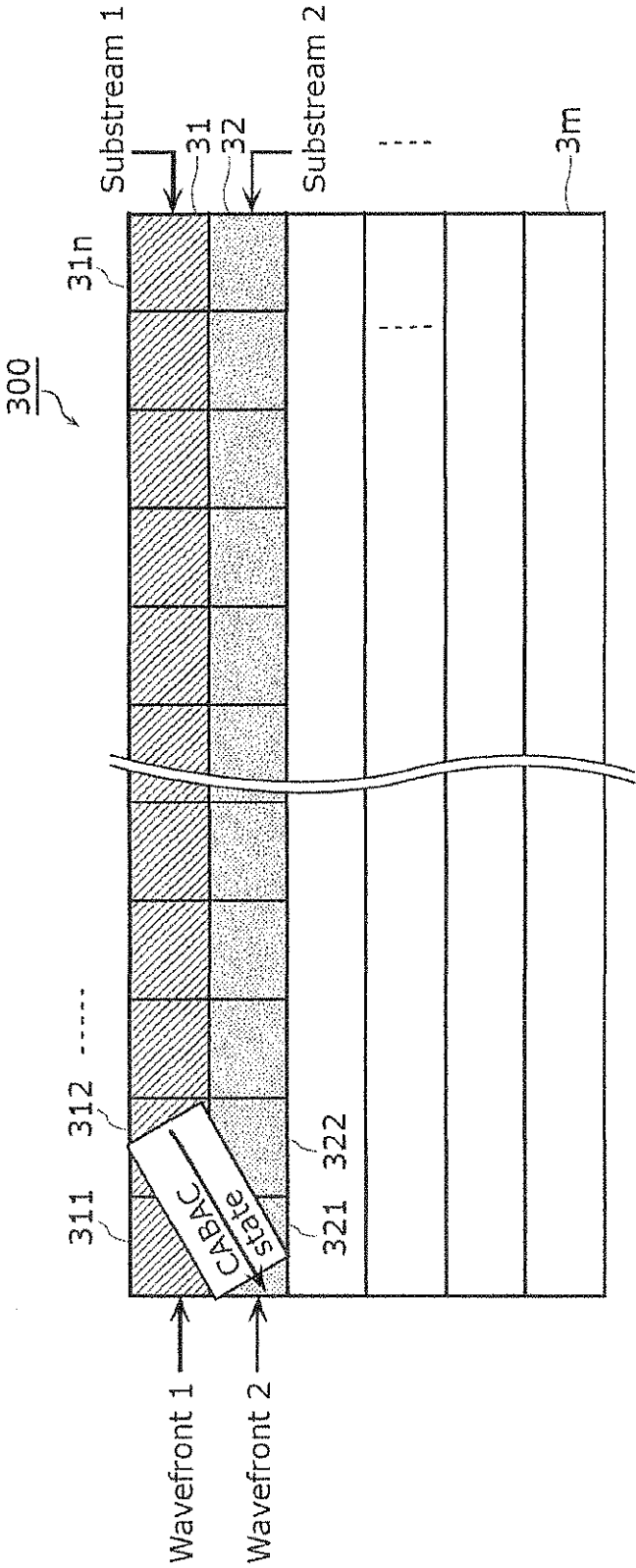


FIG. 4

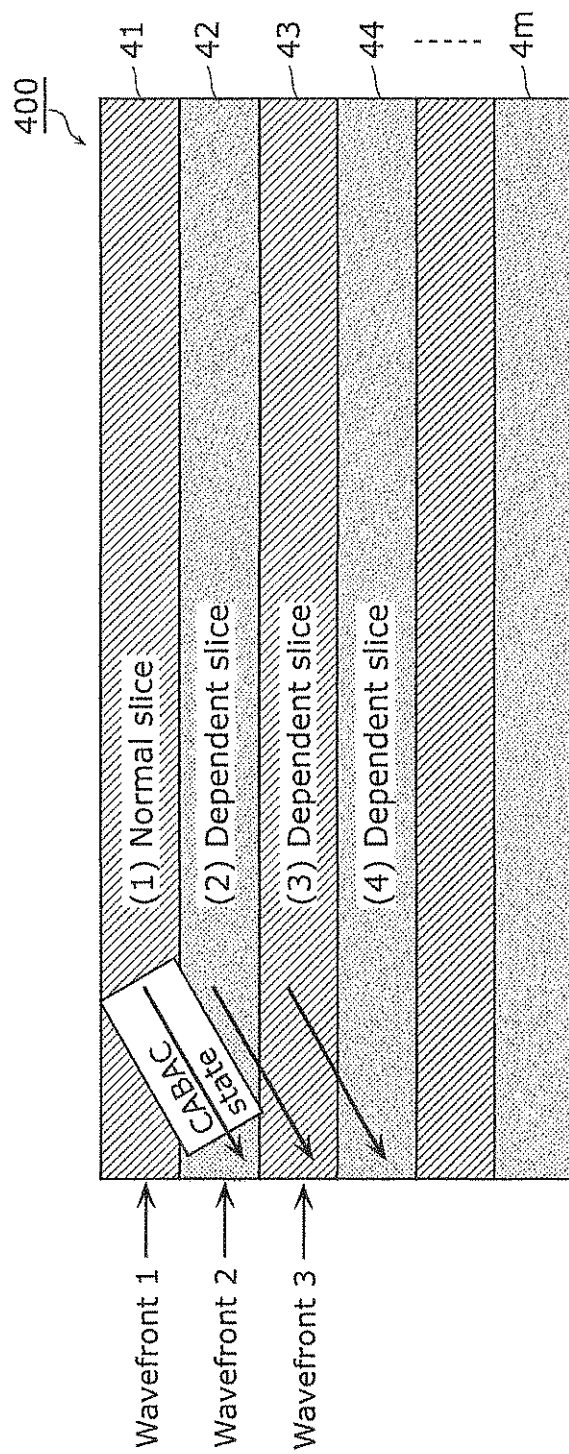


FIG. 5

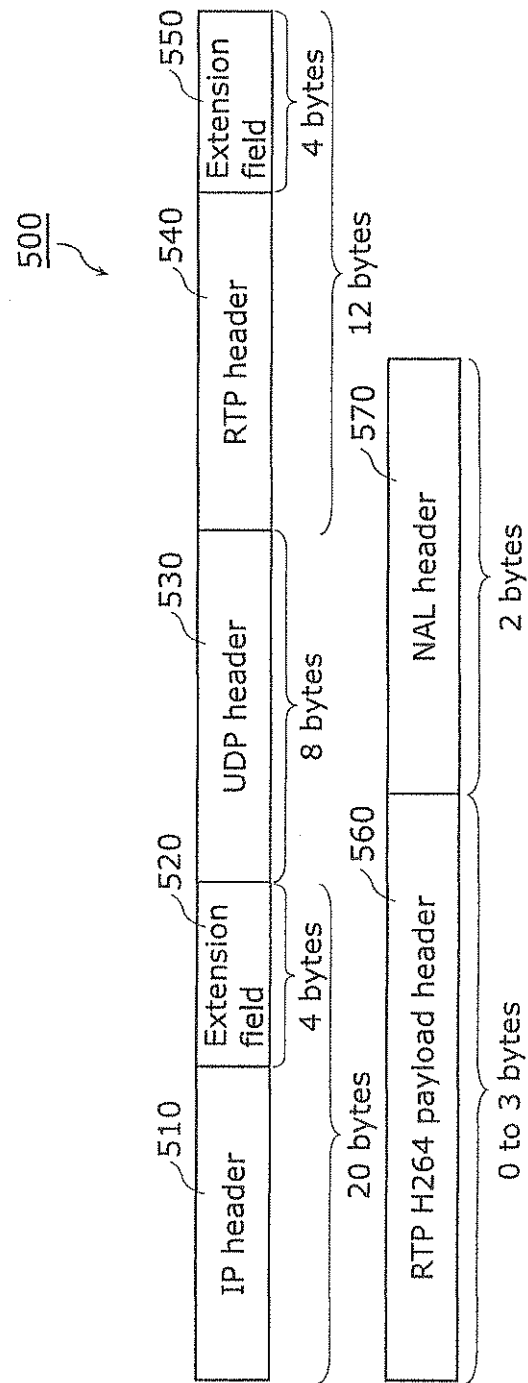


FIG. 6

600

	Descriptor
slice_header() {	
first_slice_in_pic_flag	u(1)
if(RapPicFlag)	
no_output_of_prior_pics_flag	u(1)
pic_parameter_set_id	ue(v)
if(!first_slice_in_pic_flag)	
slice_address	u(v)
if(dependent_slice_enabled_flag && !first_slice_in_pic_flag)	
dependent_slice_flag	u(1)
if(dependent_slice_flag == 0) slice_header_info	
if(tiles_or_entropy_coding_sync_idc == 1	
tiles_or_entropy_coding_sync_idc == 2) {	
num_entry_point_offsets	ue(v)
if(num_entry_point_offsets > 0) {	
offset_len_minus1	ue(v)
for(i = 0; i < num_entry_point_offsets; i++)	
entry_point_offset[i]	u(v)
}	
}	

Dependent slice flag

601

Number of
CABAC streams
in slice

602

Number of bytes
of substream
(navigation is easier
within bitstream)

603

FIG. 7

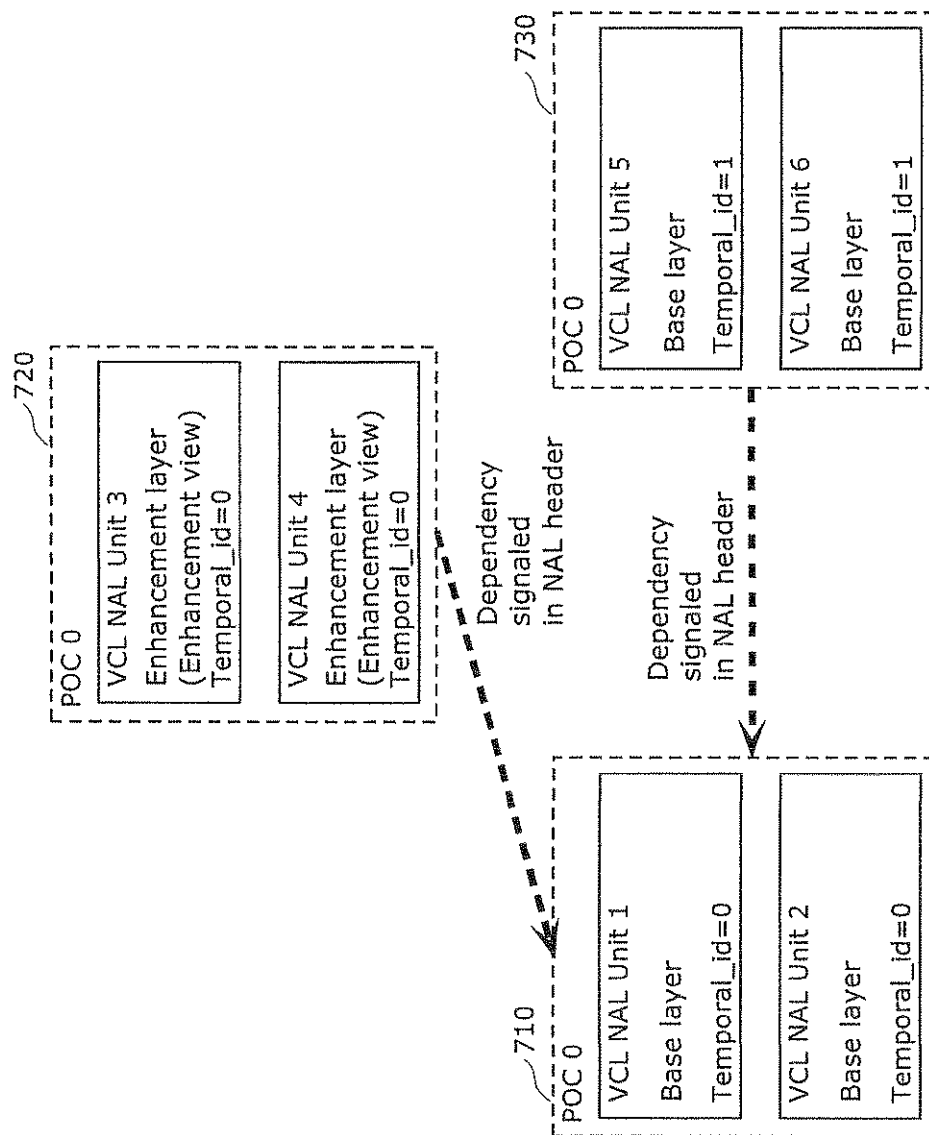


FIG. 8

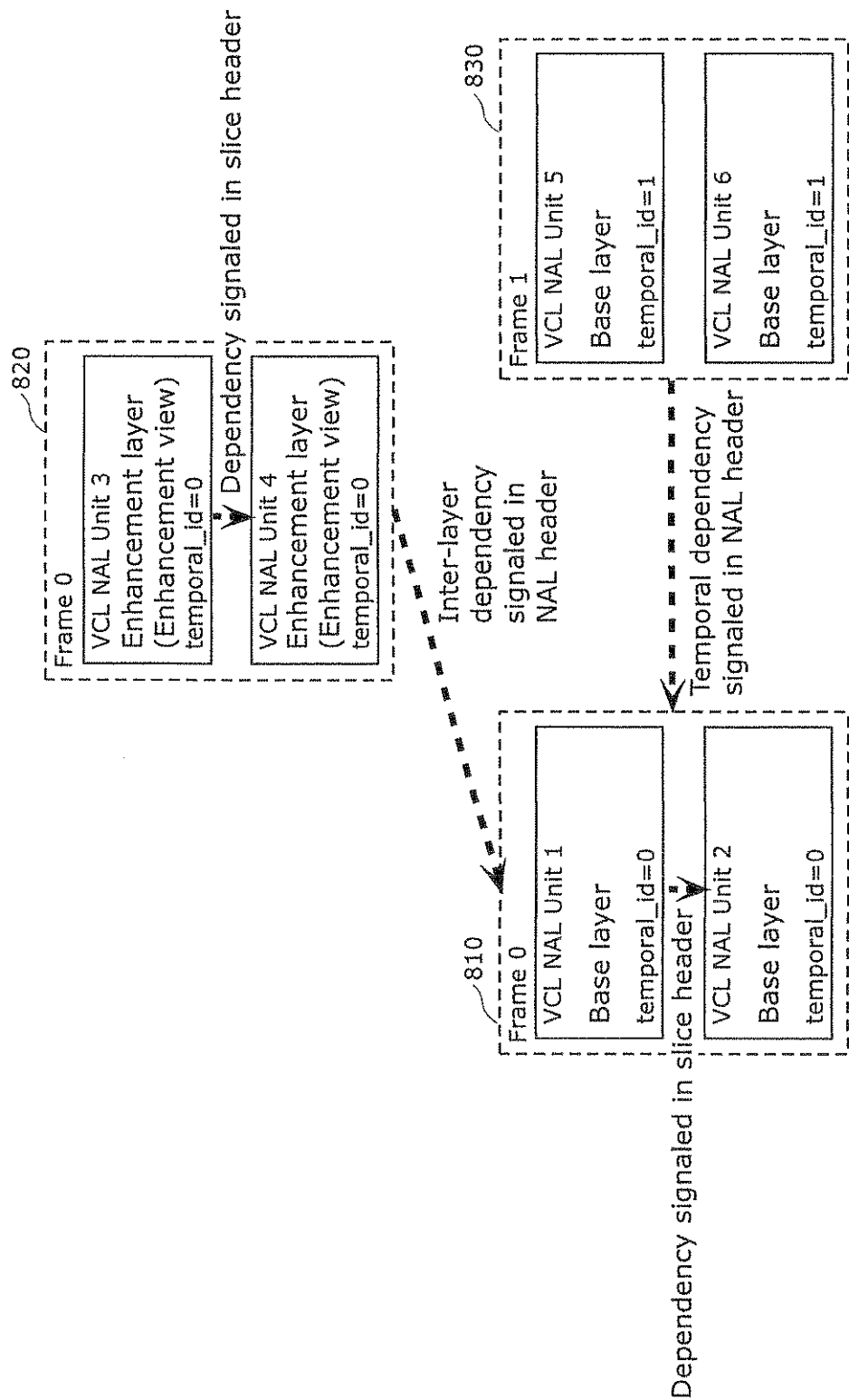


FIG. 9A

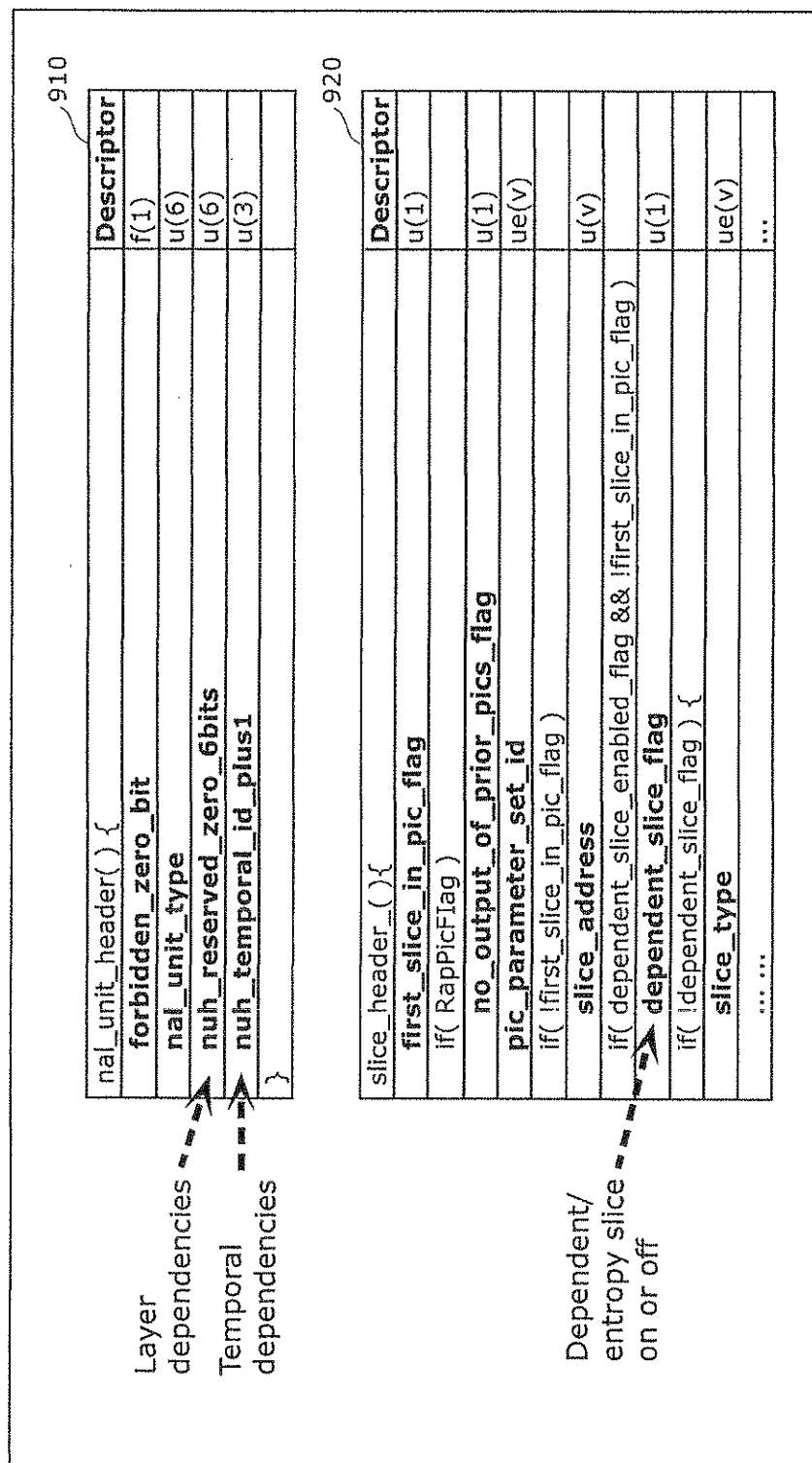


FIG. 9B

	Descriptor
seq_parameter_set_rbsp() {	
video_parameter_set_id	u(4)
sps_max_sub_layers_minus1	u(3)
sps_reserved_zero_bit	u(1)
profile_and_level(i,sps_max_sub_layers_minus1)	
seq_parameter_set_id	ue(v)
chroma_format_idc	ue(v)
if(chroma_format_idc == 3)	
separate_colour_plane_flag	u(1)
pic_width_in_luma_samples ~931	ue(v)
pic_height_in_luma_samples ~932	ue(v)
pic_cropping_flag	u(1)
if(pic_cropping_flag) {	
pic_crop_left_offset	ue(v)
pic_crop_right_offset	ue(v)
pic_crop_top_offset	ue(v)
pic_crop_bottom_offset	ue(v)
}	
bit_depth_luma_minus8	ue(v)
bit_depth_chroma_minus8	ue(v)
pcm_enabled_flag	u(1)
if(pcm_enabled_flag) {	
pcm_sample_bit_depth_luma_minus1	u(4)
pcm_sample_bit_depth_chroma_minus1	u(4)
}	
log2_max_pic_order_cnt_lsb_minus4	ue(v)
for(i=0; i <= sps_max_sub_layers_minus1; i++) {	
sps_max_dec_pic_buffering[i]	ue(v)
sps_max_num_reorder_pics[i]	ue(v)
sps_max_latency_increase[i]	ue(v)
}	
restricted_ref_pic_lists_flag	u(1)
if(restricted_ref_pic_lists_flag)	
lists_modification_present_flag	u(1)
log2_min_coding_block_size_minus3 ~933	ue(v)
log2_diff_max_min_coding_block_size ~934	ue(v)
...	
...	
...	
}	

920

slice_header() {
first_slice_in_pic_flag
if(RapPicFlag)
no_output_of_prior_pics_flag
pic_parameter_set_id
if(!first_slice_in_pic_flag)
slice_address
if(dependent_slice_enabled_flag && !first_slice_in_pic_flag)
dependent_slice_flag
if(!dependent_slice_flag) {
slice_type
... ..

930

935

In order to parse slice_address, syntax elements from SPS are needed. Therefore, SPS needs to be parsed up to this point.

FIG. 9C

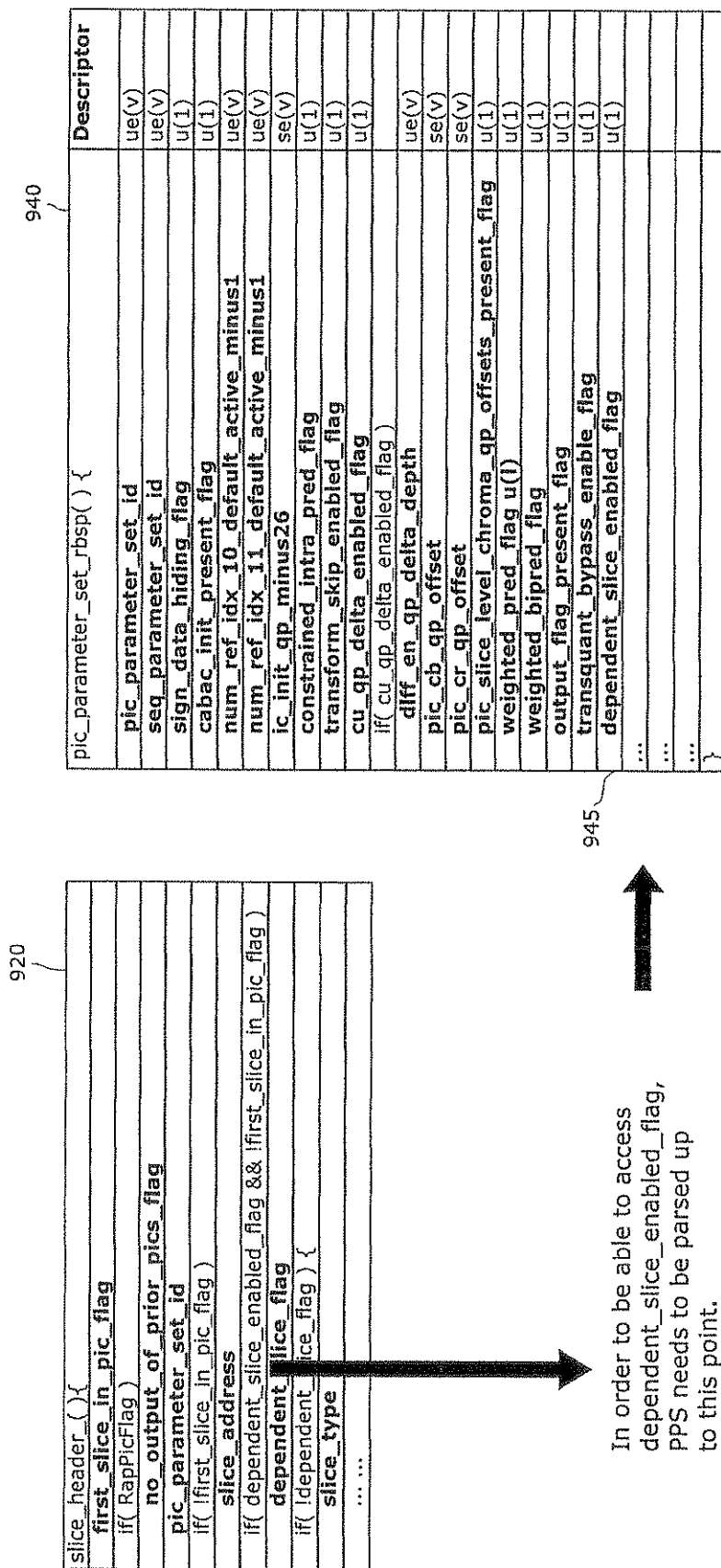


FIG. 10

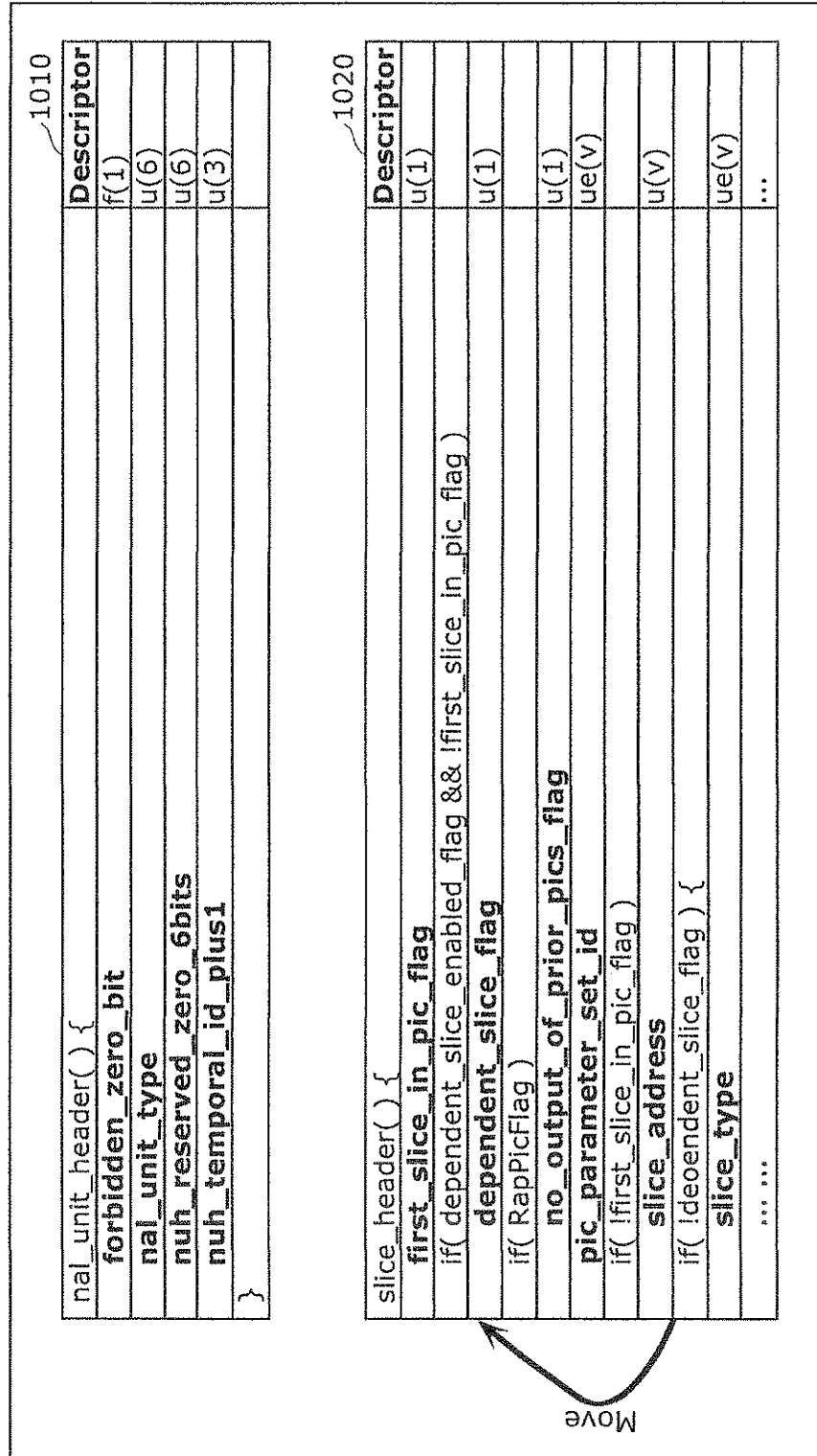


FIG. 11

slice_header() {	Descriptor
first_slice_in_pic_flag	u(1)
if(dependent_slice_enabled_flag && !first_slice_in_pic_flag)	
dependent_slice_flag	u(1)
if(RapPicFlag)	
no_output_of_prior_pics_flag	u(1)
pic_parameter_set_id	ue(v)
if(!first_slice_in_pic_flag)	
slice_address	u(v)
if(!dependent_slice_flag) {	
slice_type	ue(v)
... ..	...

1110

FIG. 12

1210	
nal_unit_header() {	Descriptor
forbidden_zero_bit	f(1)
nal_unit_type	u(6)
nuh_reserved_zero_6bits	u(6)
nuh_temporal_id_plus1	u(3)
}	
1220	
slice_header() {	Descriptor
dependent_slice_flag	u(1)
if(!dependent_slice_flag)	
first_slice_in_pic_flag	u(1)
if(RapPicFlag)	
no_output_of_prior_pics_flag	u(1)
pic_parameter_set_id	ue(v)
if(!first_slice_in_pic_flag)	
slice_address	u(v)
if(!dependent_slice_flag) {	
slice_type	ue(v)
... ..	...

FIG. 13

1310

slice_header () {	Descriptor
first_slice_in_pic_flag	u(1)
if(RapPicFlag)	
no_output_of_prior_pics_flag	u(1)
pic_parameter_set_id	ue(v)
if(dependent_slice_enabled_flag && !first_slice_in_pic_flag)	
dependent_slice_flag	u(1)
if(!first_slice_in_pic_flag)	
slice_address	u(v)
if(!dependent_slice_flag) {	
slice_type	ue(v)
... ..	...

FIG. 14

nal_unit_header() {		1410
forbidden_zero_bit	Descriptor	f(1)
nal_unit_type		u(6)
nuh_reserved_zero_6bits		u(5)
nuh_temporal_id_plus1		u(3)
dependent_slice_flag		u(1)
}		

FIG. 15

nal_unit_header() {		1510
forbidden_zero_bit	Descriptor	f(1)
nal_unit_type		u(6)
nuh_reserved_zero_6bits		u(6)
nuh_temporal_id_plus1		u(3)
}		
slice_header() {		1520
first_slice_in_pic_flag	Descriptor	u(1)
if(RapPicFlag)		
no_output_of_prior_pics_flag		u(1)
pic_parameter_set_id		ue(v)
if(!first_slice_in_pic_flag)		
slice_address		u(v)
if(nal_unit_type == 15 nal_unit_type == 16)		
dependent_slice_flag == 1		
if(!dependent_slice_flag) {		
slice_type		ue(v)
... ..		...

FIG. 16

1610	
nal_unit_header() {	Descriptor
forbidden_zero_bit	f(1)
nal_unit_type	u(6)
nuh_reserved_zero_6bits	u(6)
nuh_temporal_id_plus1	u(3)
}	
1620	
slice_header() {	Descriptor
first_slice_in_pic_flag	u(1)
if(RapPicFlag)	
no_output_of_prior_pics_flag	u(1)
pic_parameter_set_id	ue(v)
if(!first_slice_in_pic_flag)	
slice_address	u(v)
if(dependent_slice_enabled_flag && !first_slice_in_pic_flag)	
dependent_slice_flag	u(1)
if(!dependent_slice_flag) {	
slice_type	ue(v)
... ..	...

FIG. 17

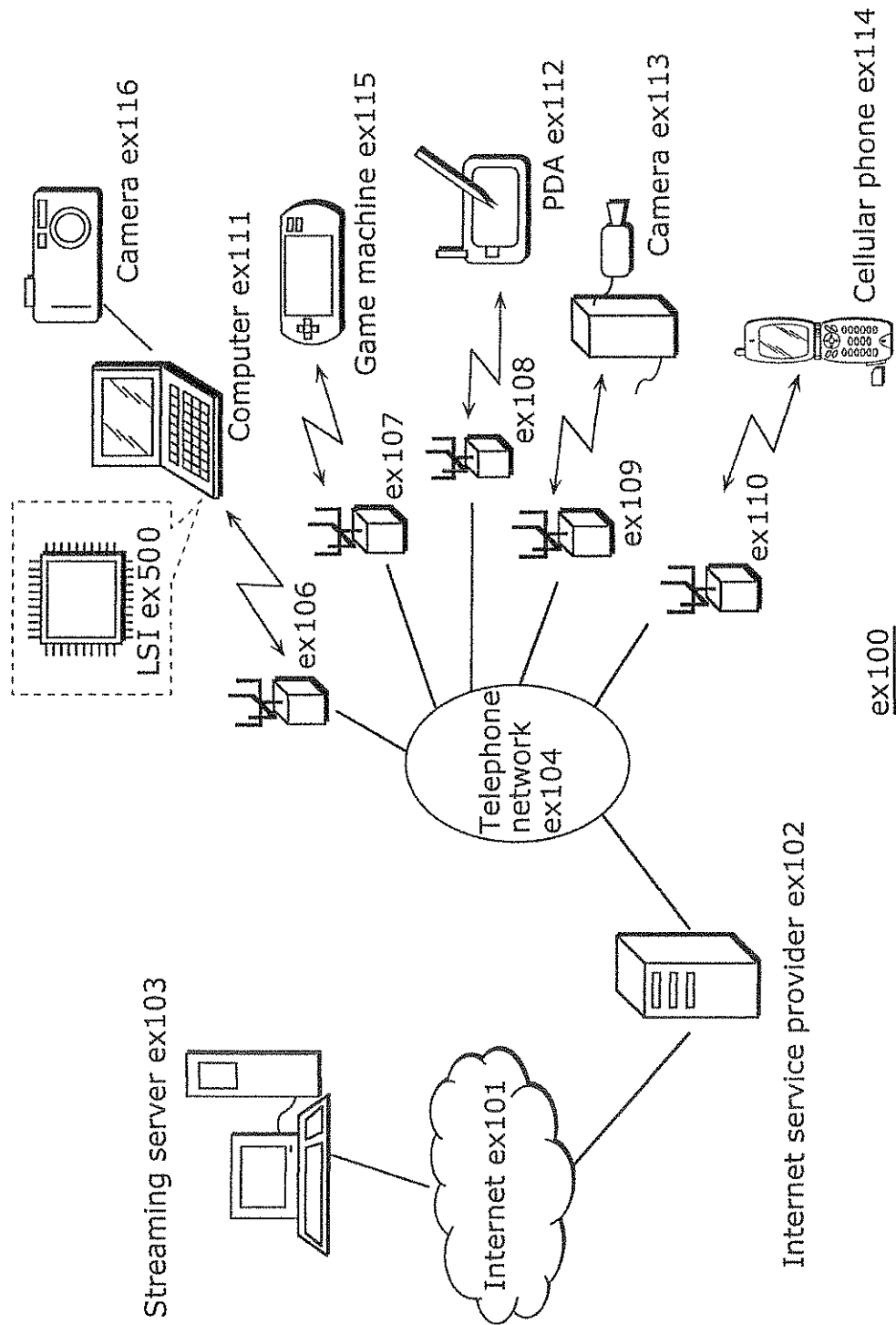


FIG. 18

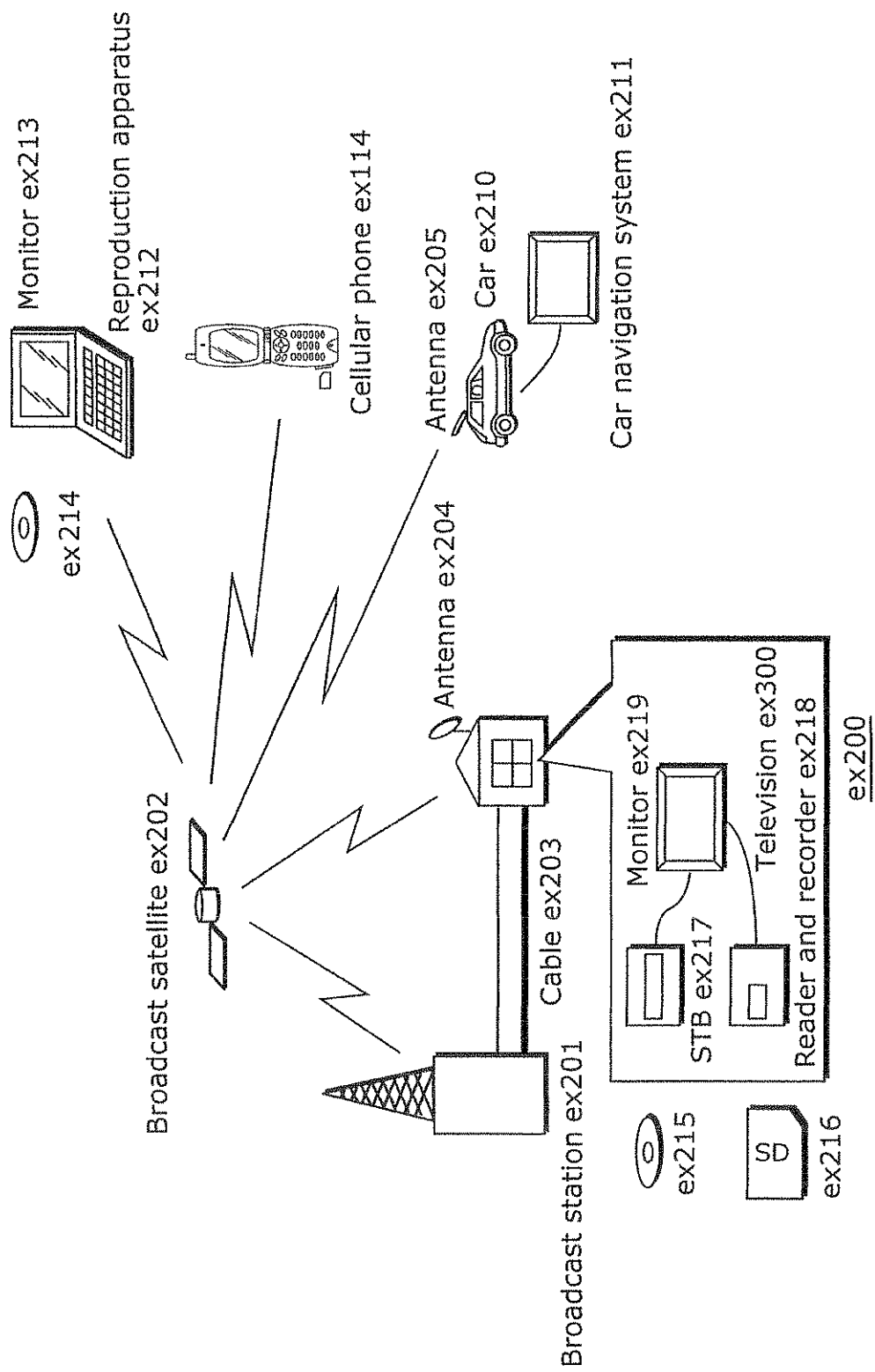


FIG. 19

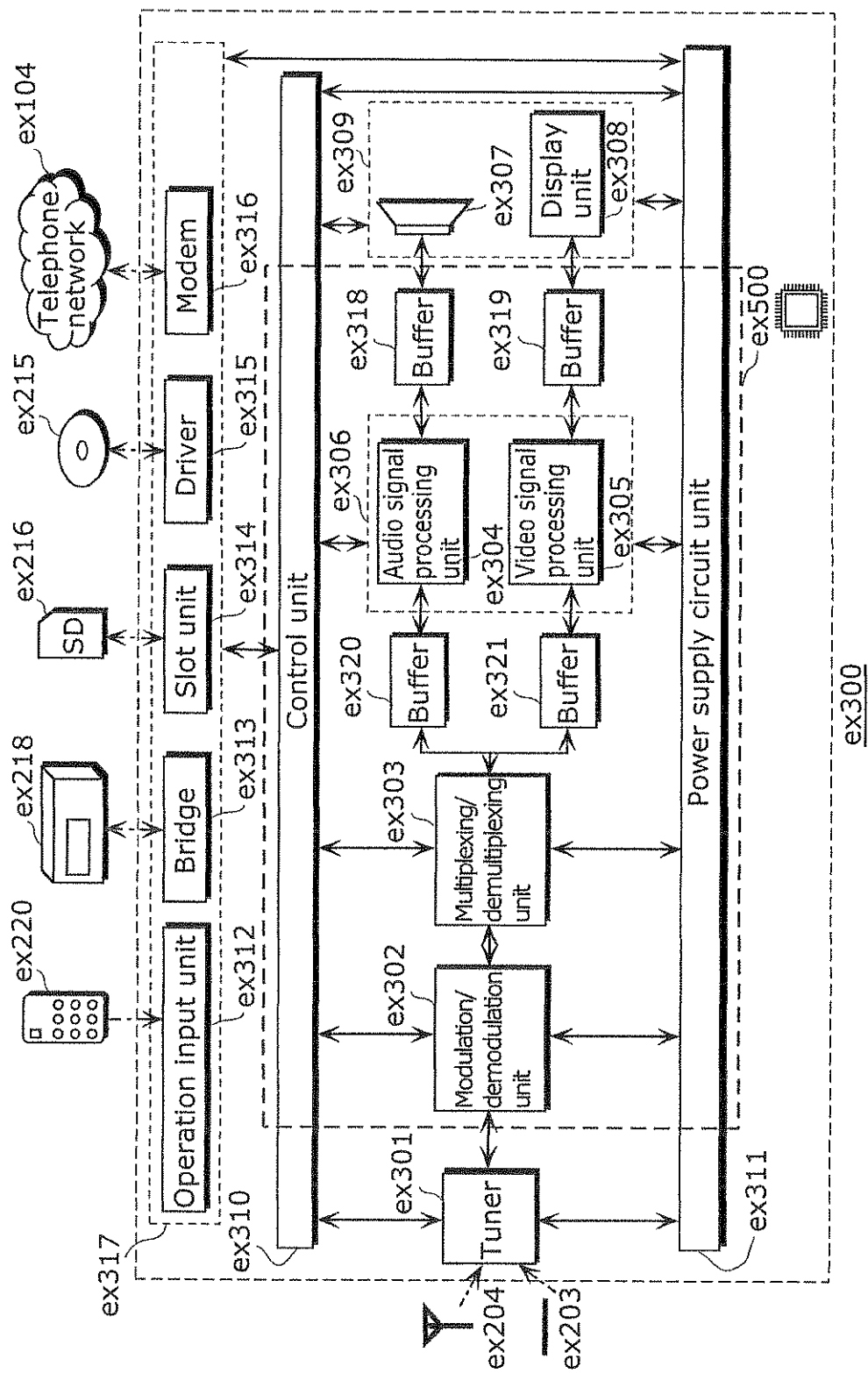
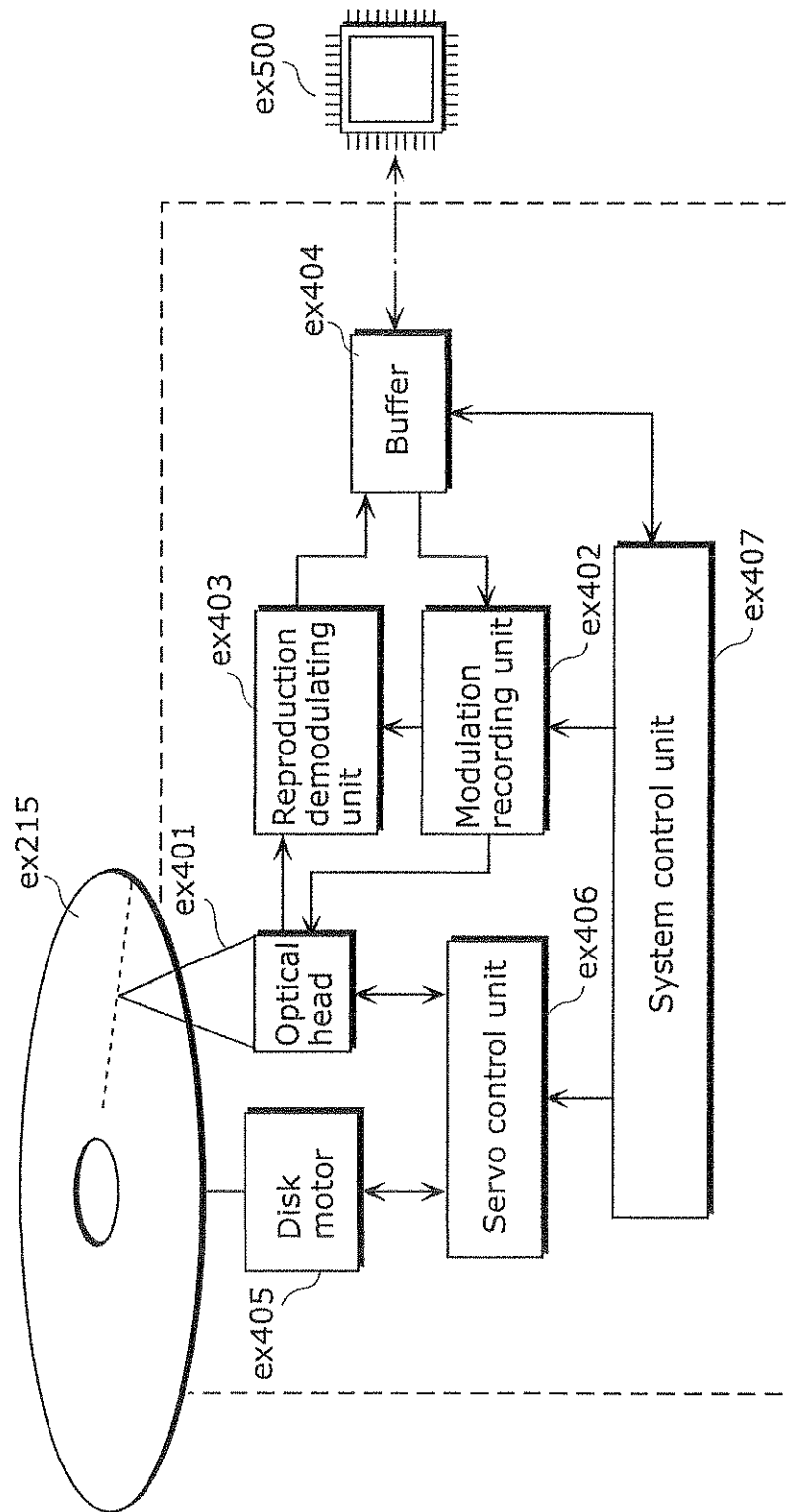


FIG. 20



ex400

FIG. 21

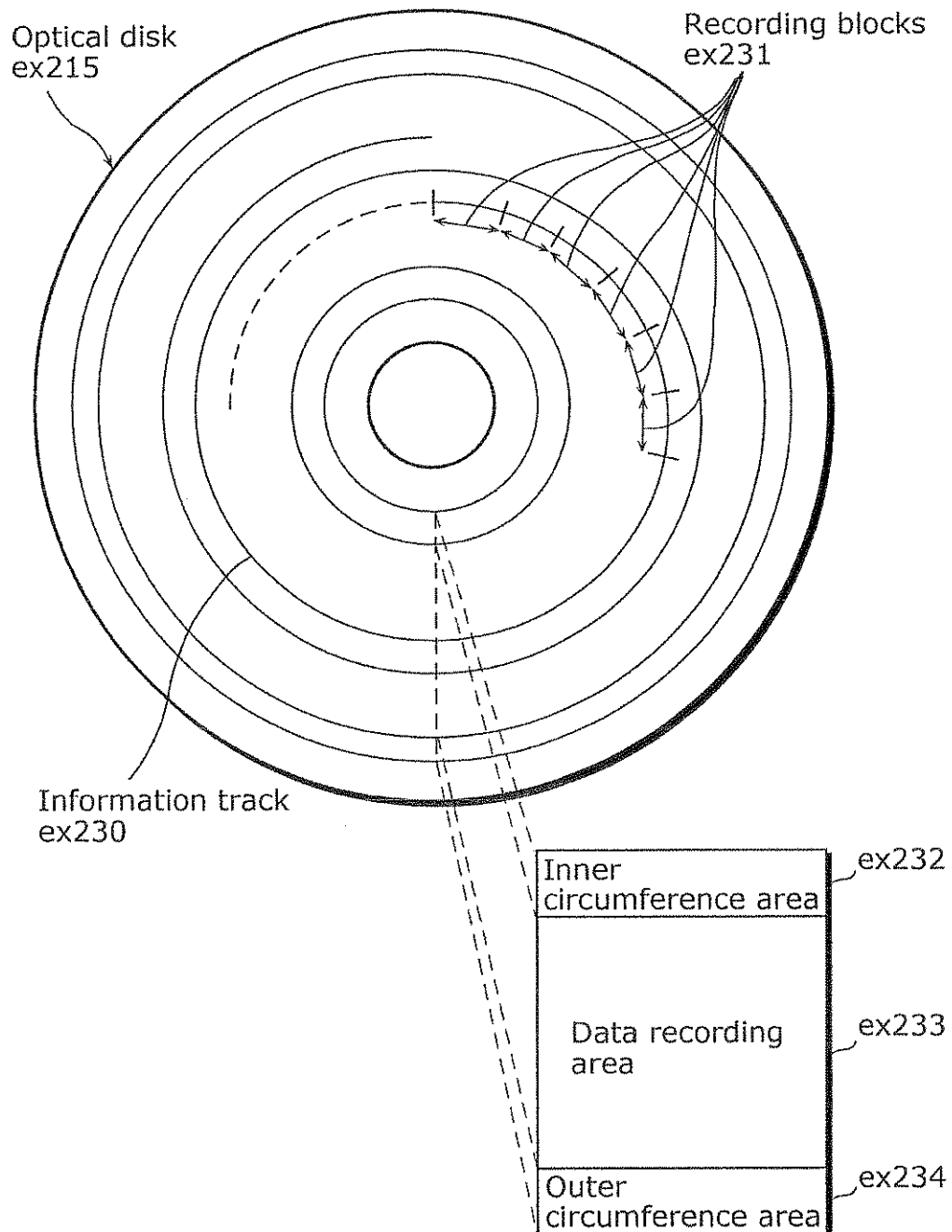


FIG. 22A

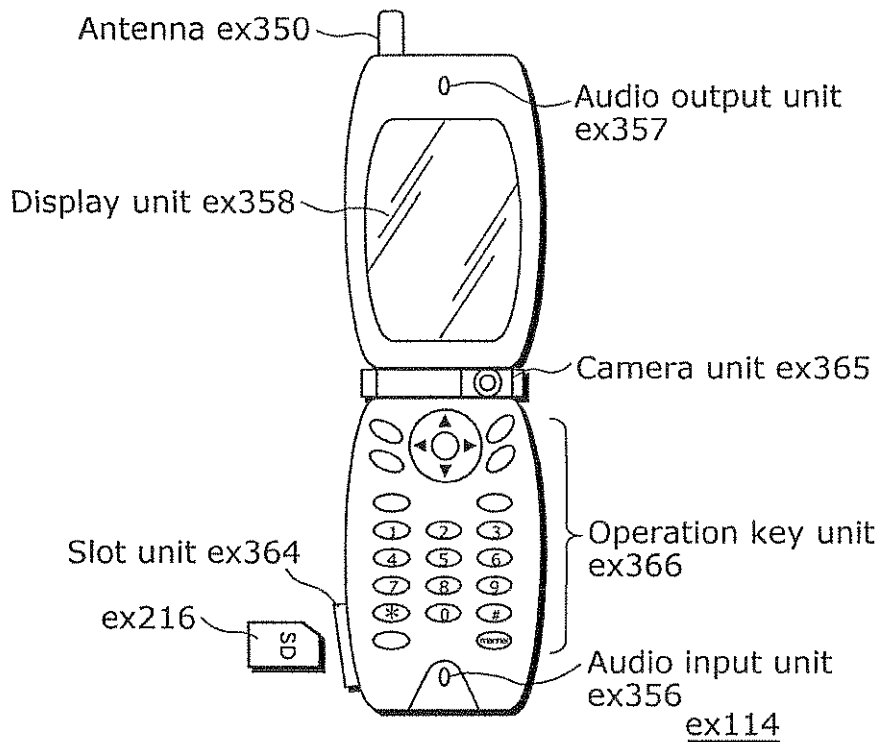


FIG. 22B

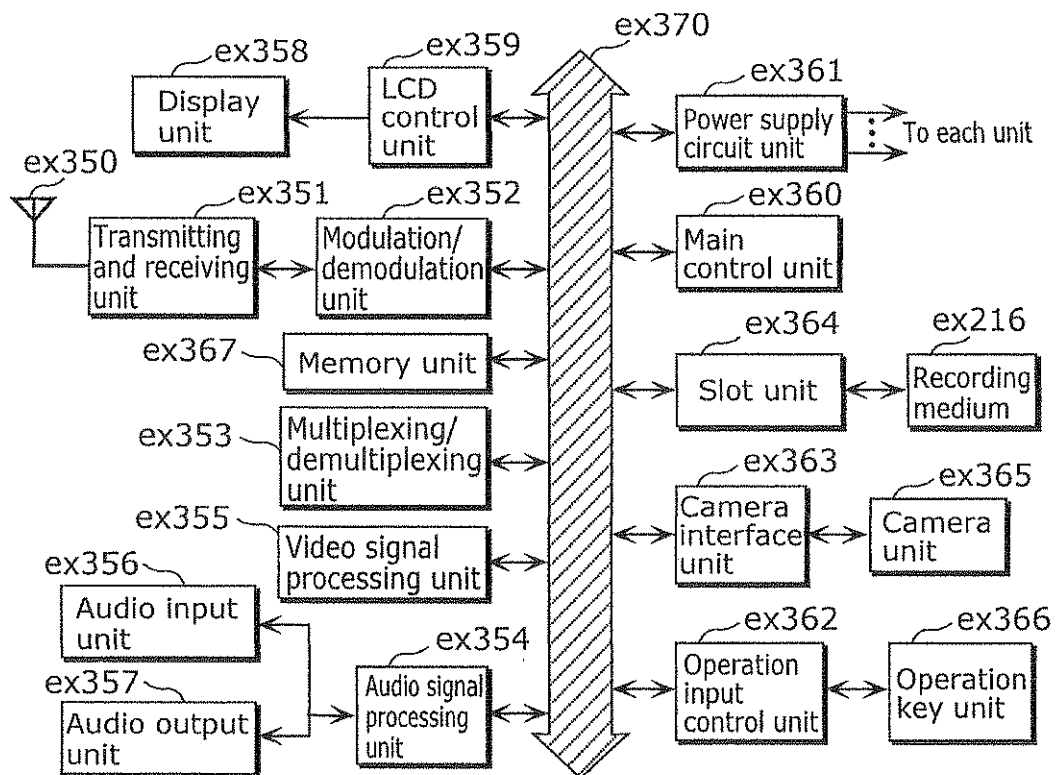


FIG. 23

Video stream (PID=0x1011, Primary video)
Audio stream (PID=0x1100)
Audio stream (PID=0x1101)
Presentation graphics stream (PID=0x1200)
Presentation graphics stream (PID=0x1201)
Interactive graphics stream (PID=0x1400)
Video stream (PID=0x1B00, Secondary video)
Video stream (PID=0x1B01, Secondary video)

FIG. 24

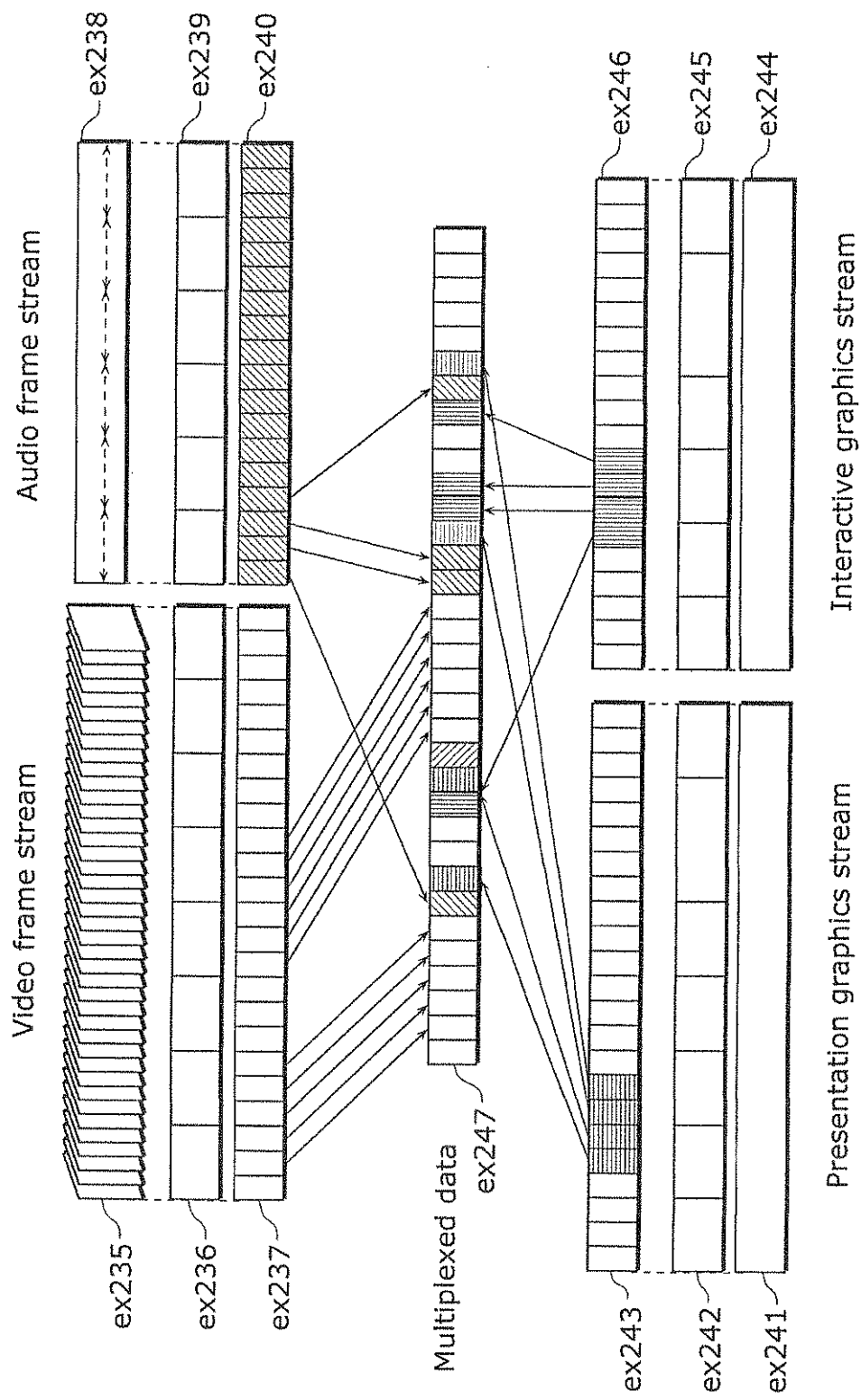


FIG. 25

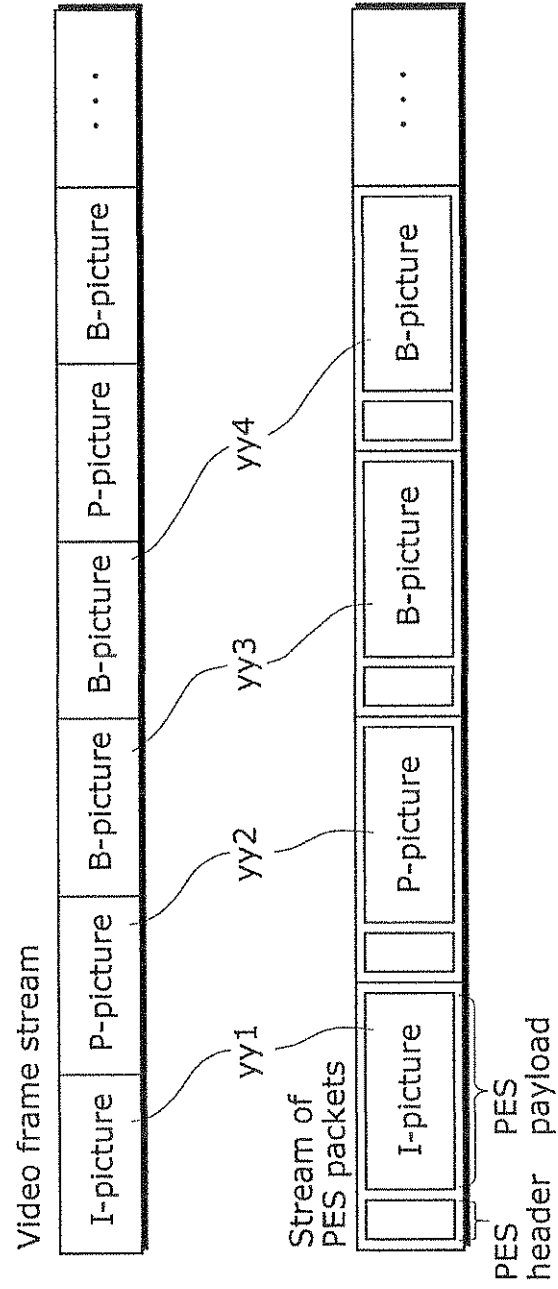


FIG. 26

Stream of TS packets

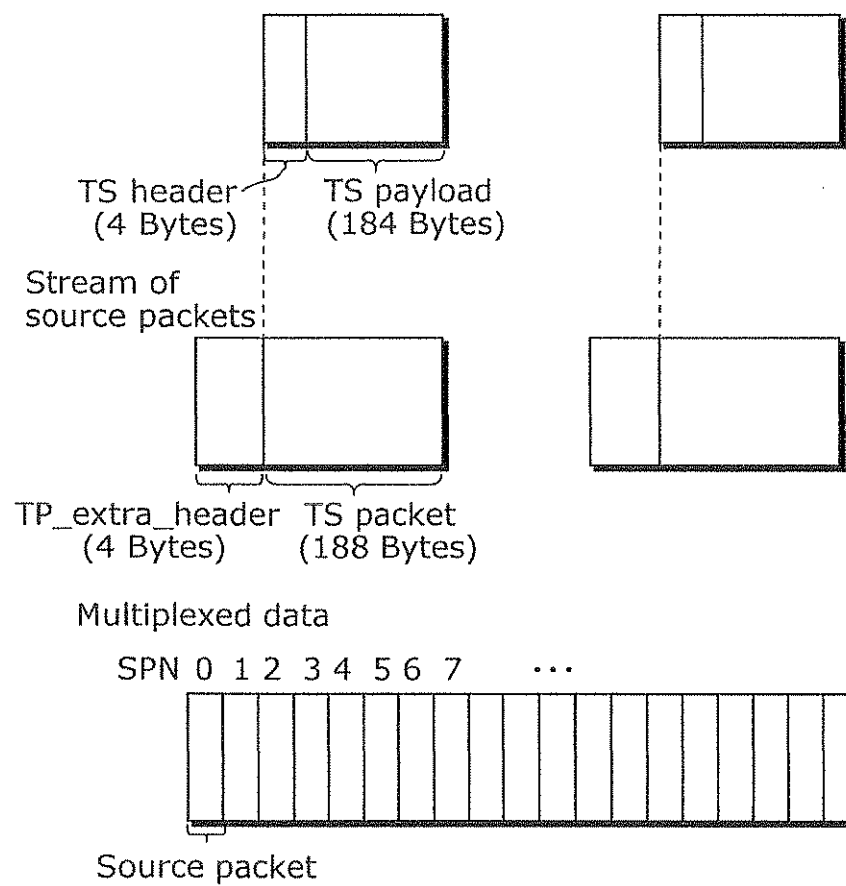


FIG. 27

Data structure of PMT

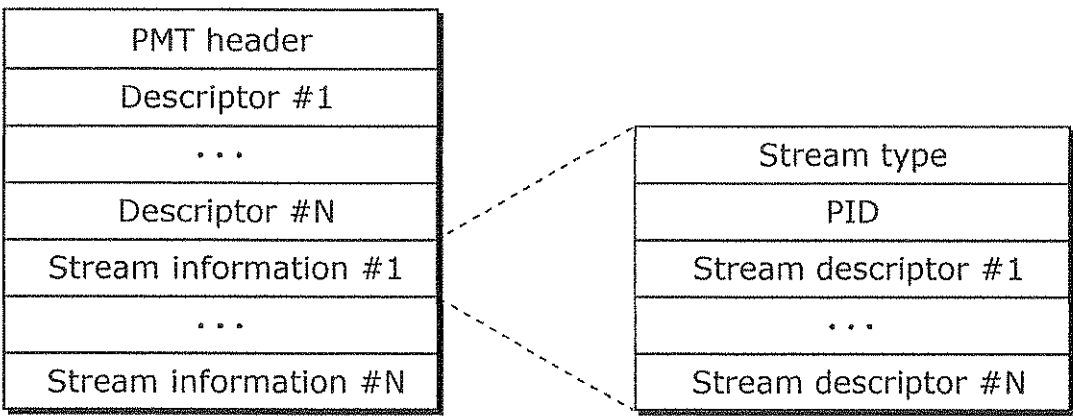


FIG. 28

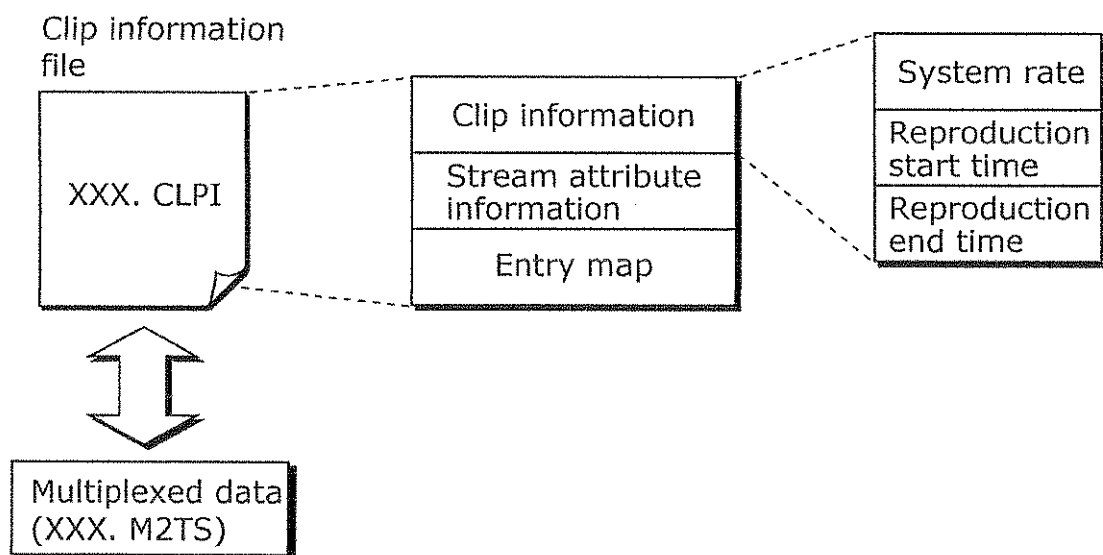


FIG. 29

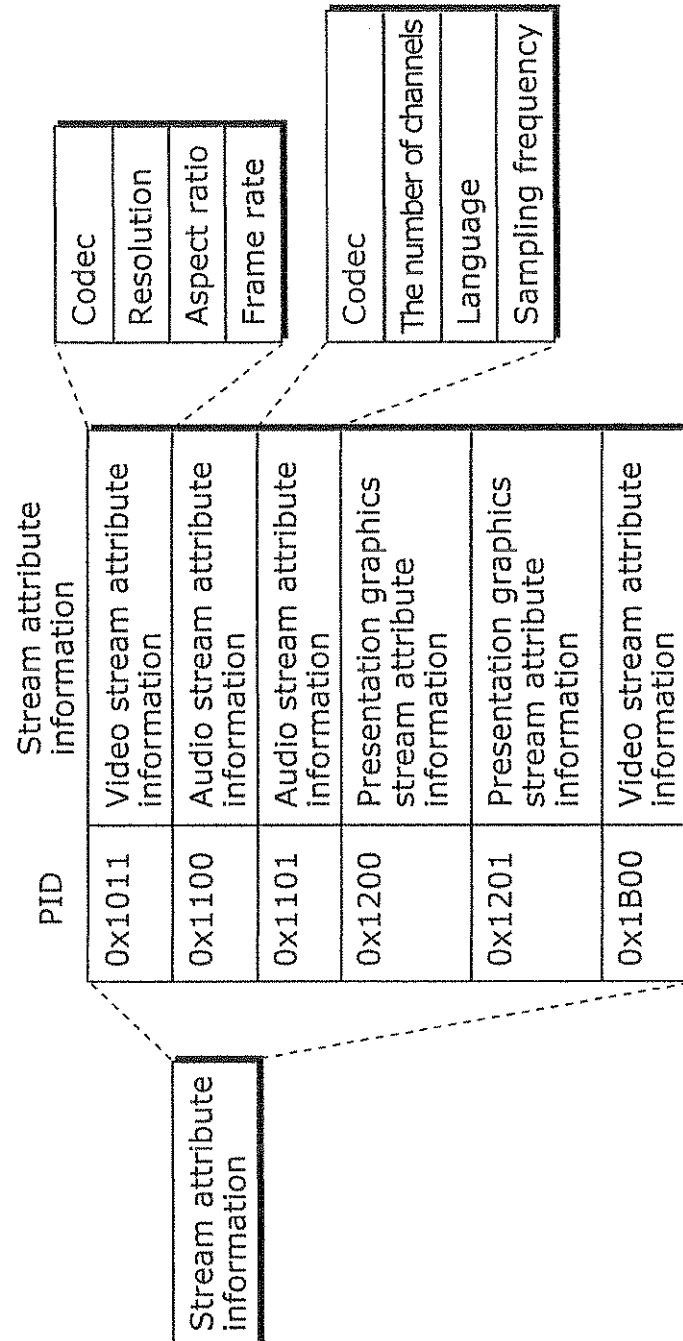


FIG. 30

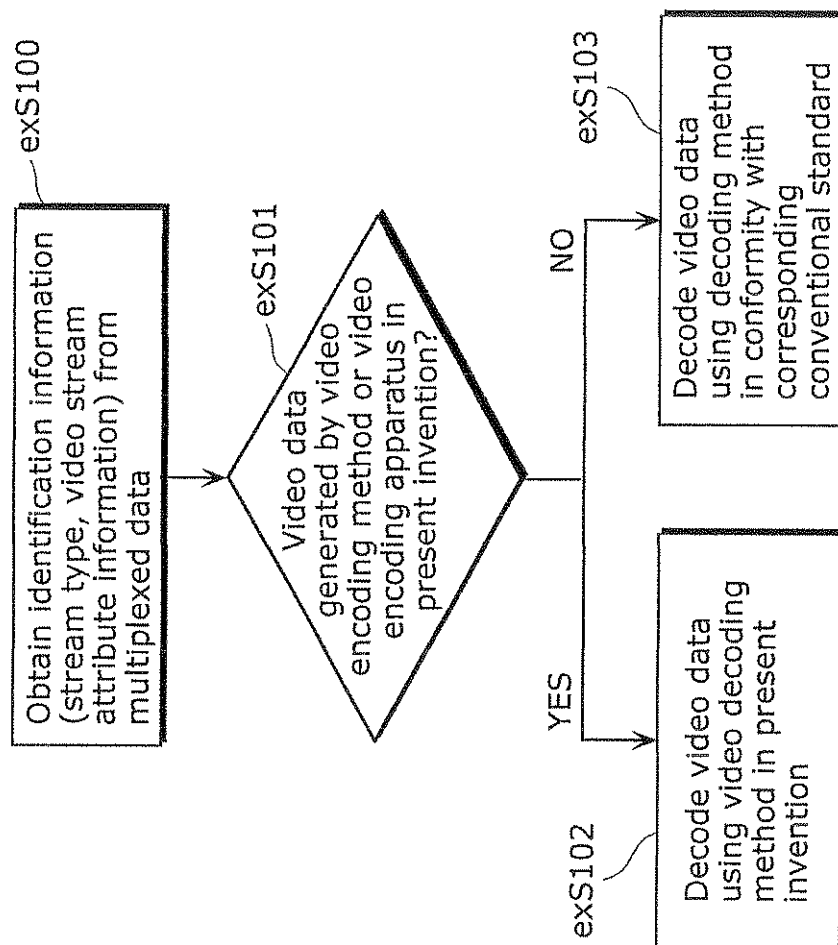


FIG. 31

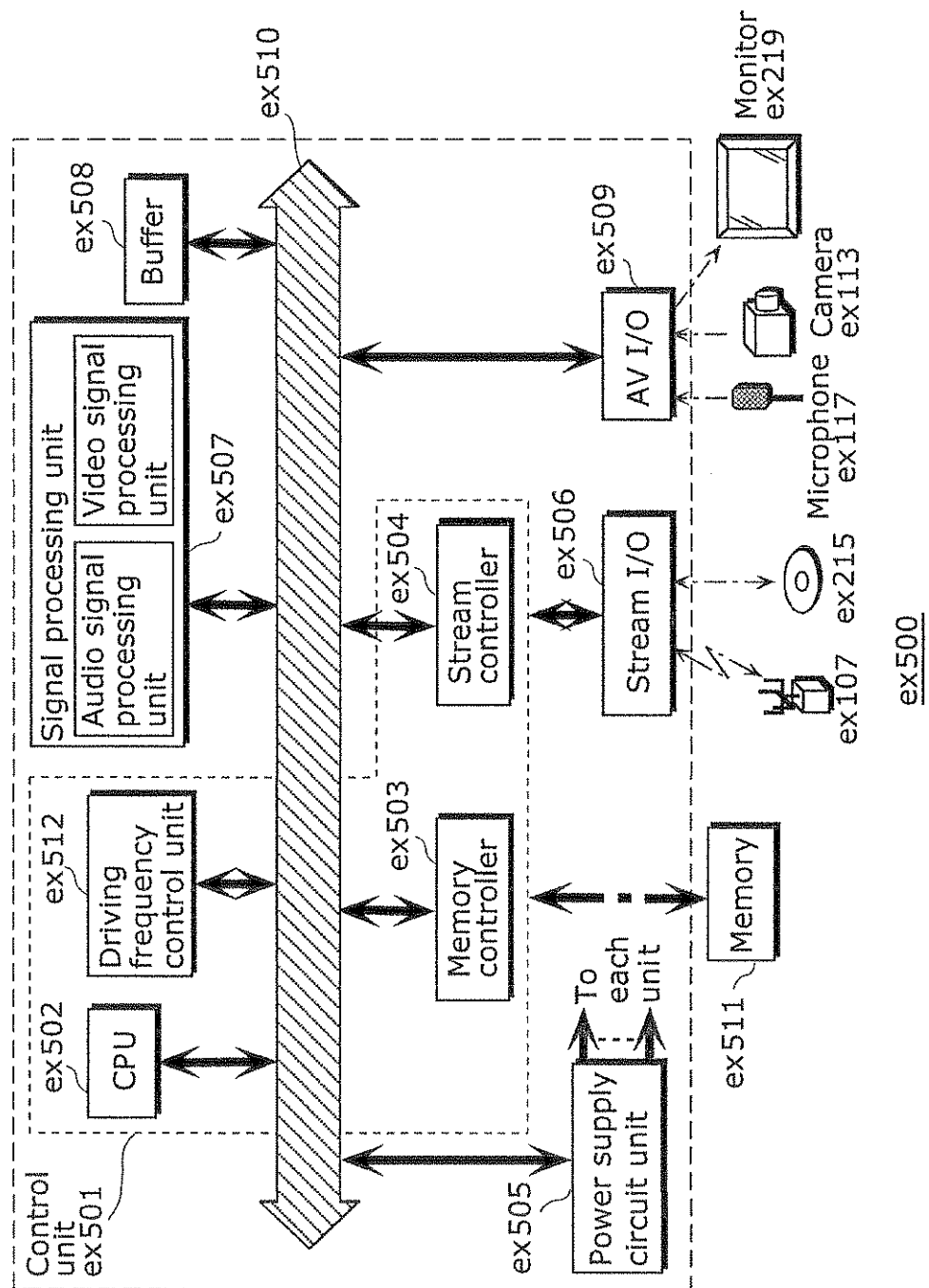


FIG. 32

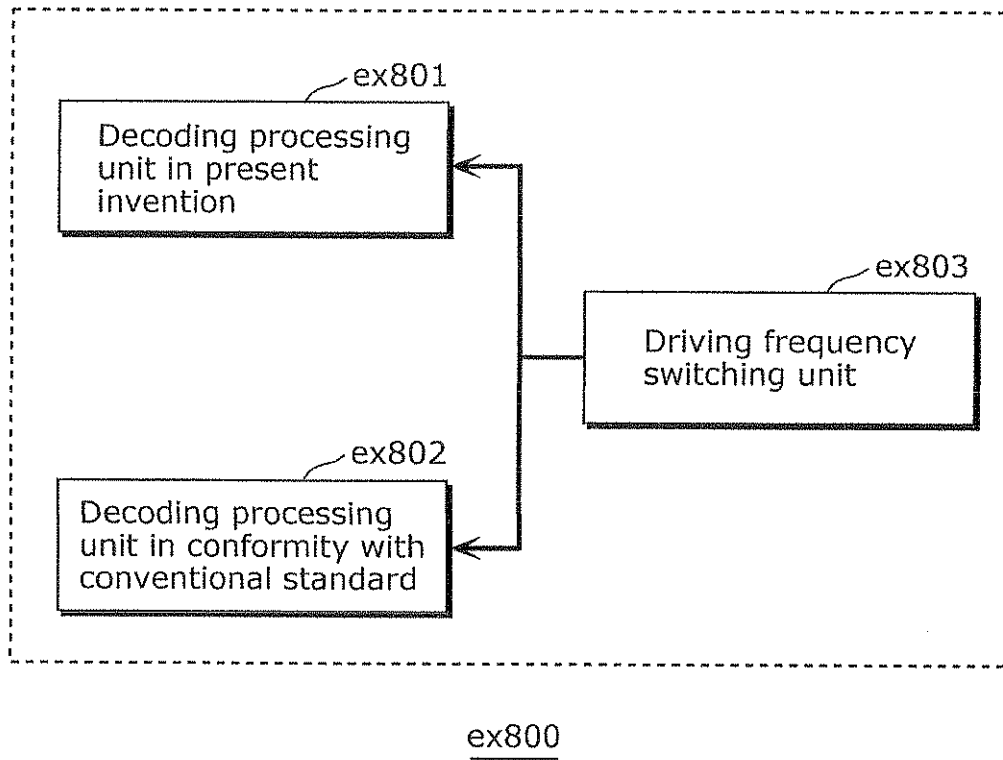


FIG. 33

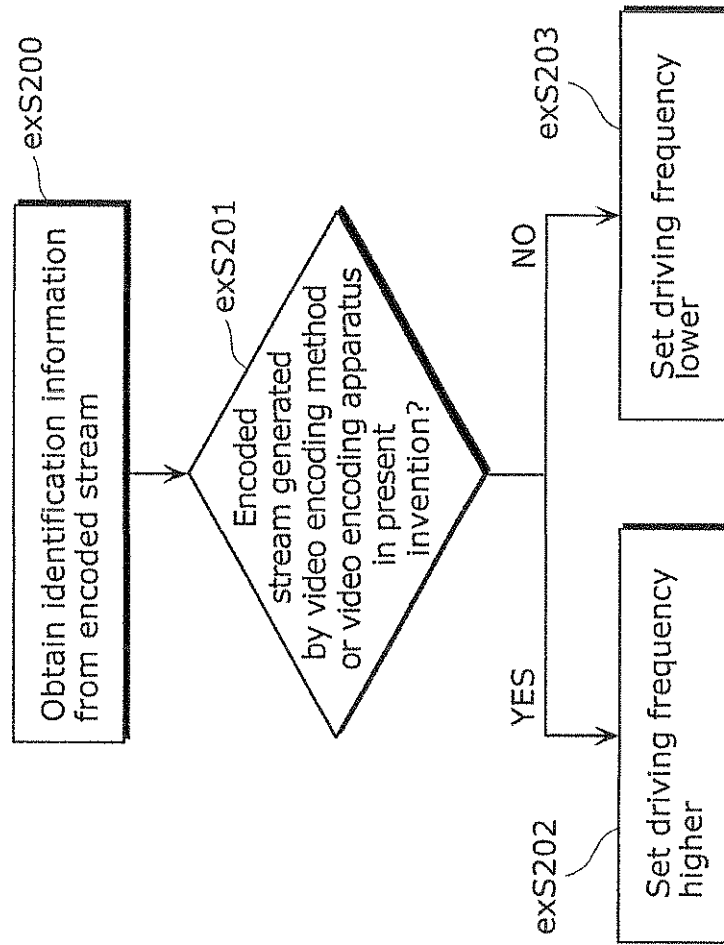


FIG. 34

Corresponding standard	Driving frequency
MPEG-4 AVC	500 MHz
MPEG-2	350 MHz
⋮	⋮

FIG. 35A

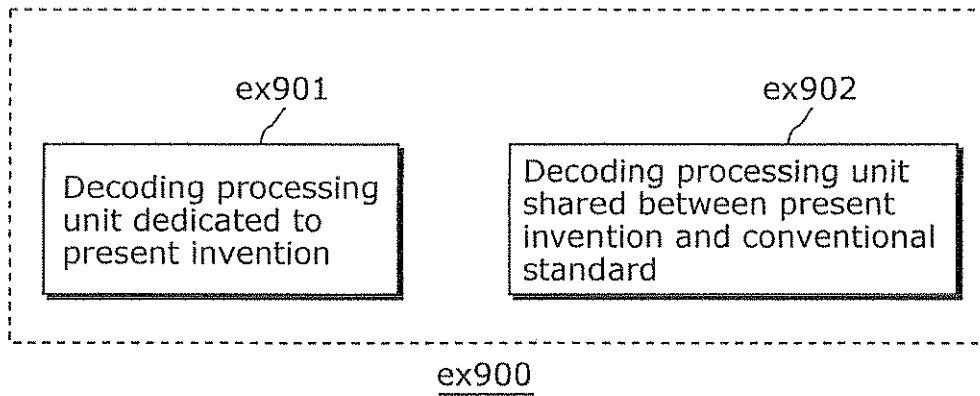


FIG. 35B

