

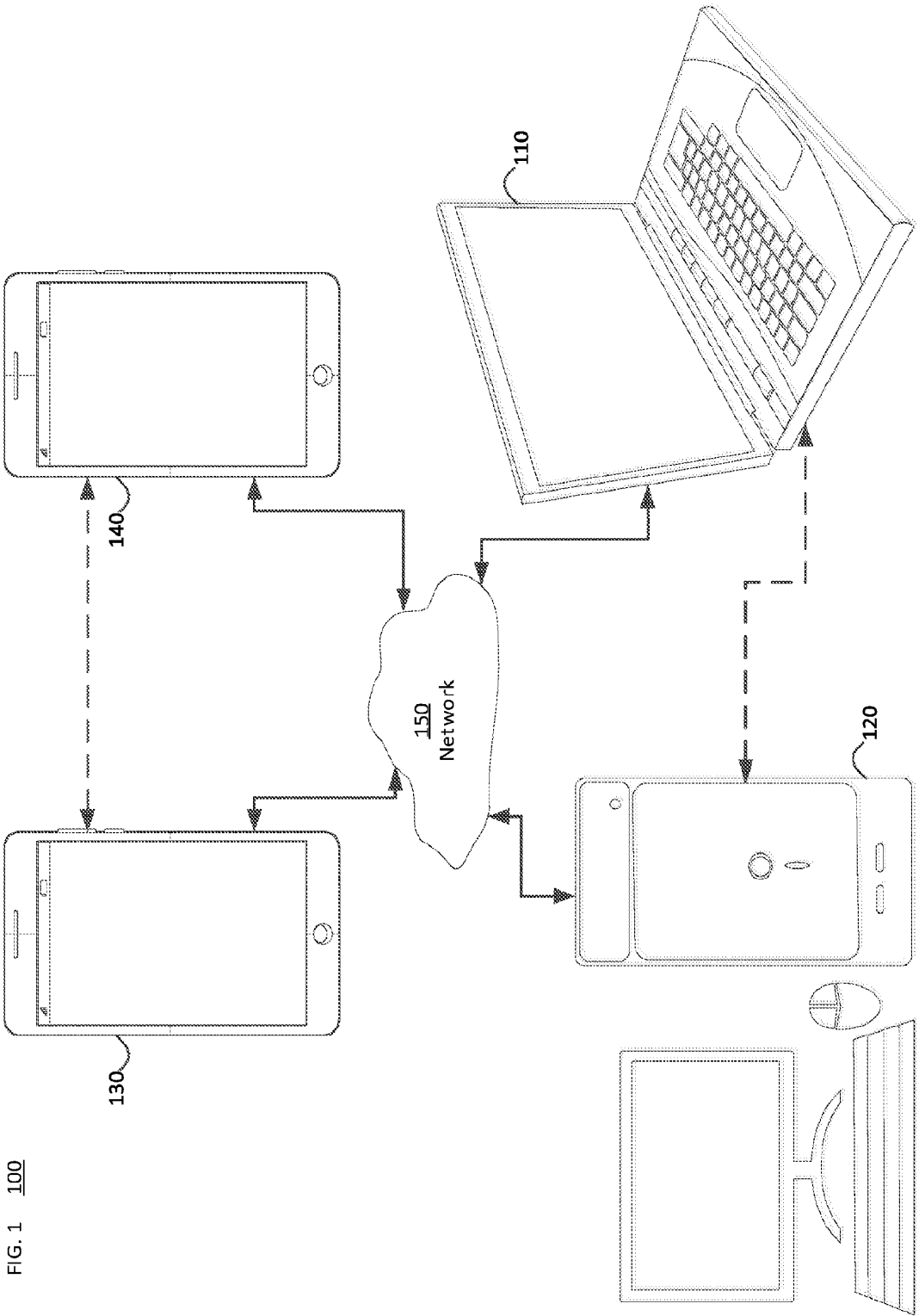


FIG. 1 100

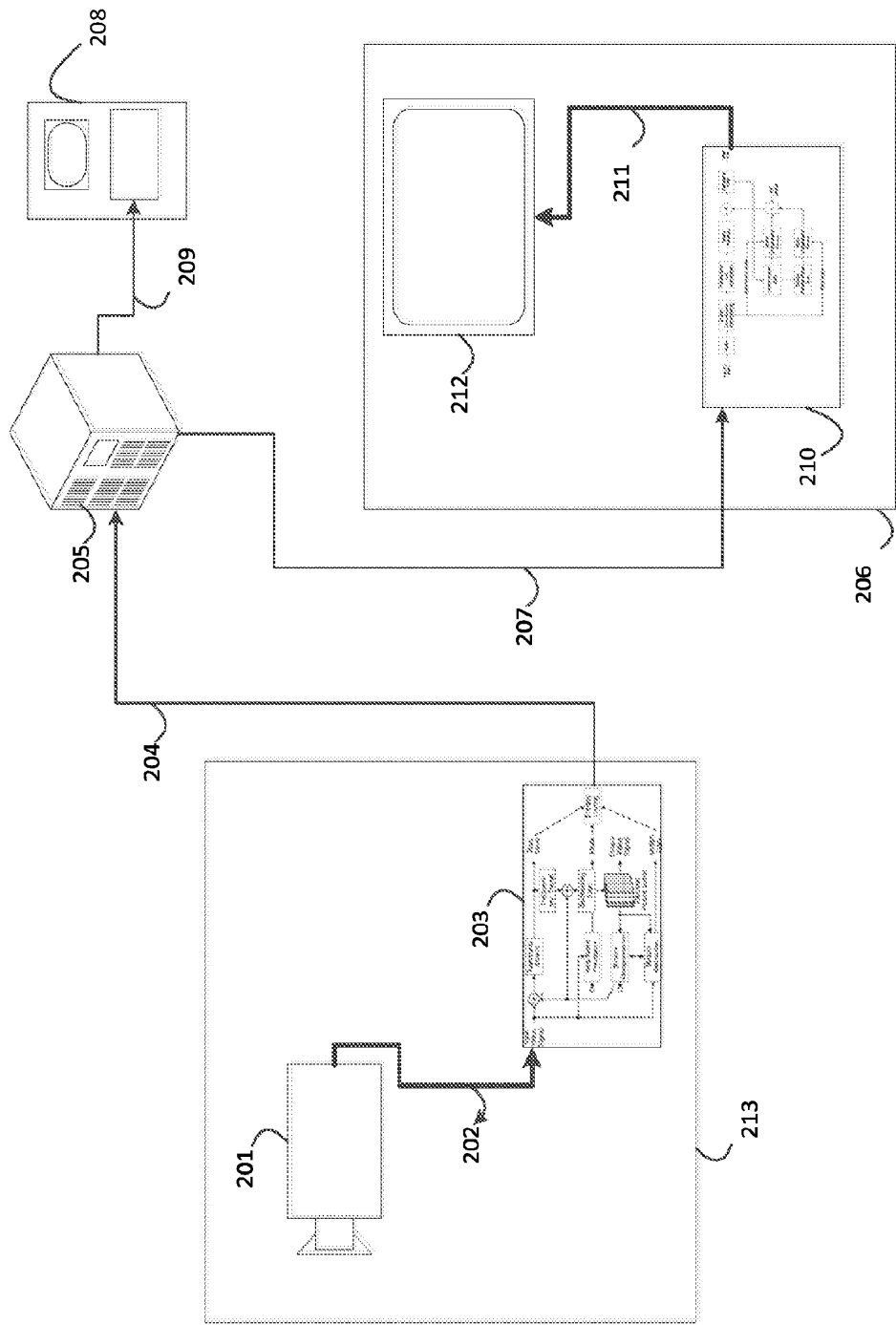


FIG. 2 Streaming System 200

FIG. 3 Decoder 210

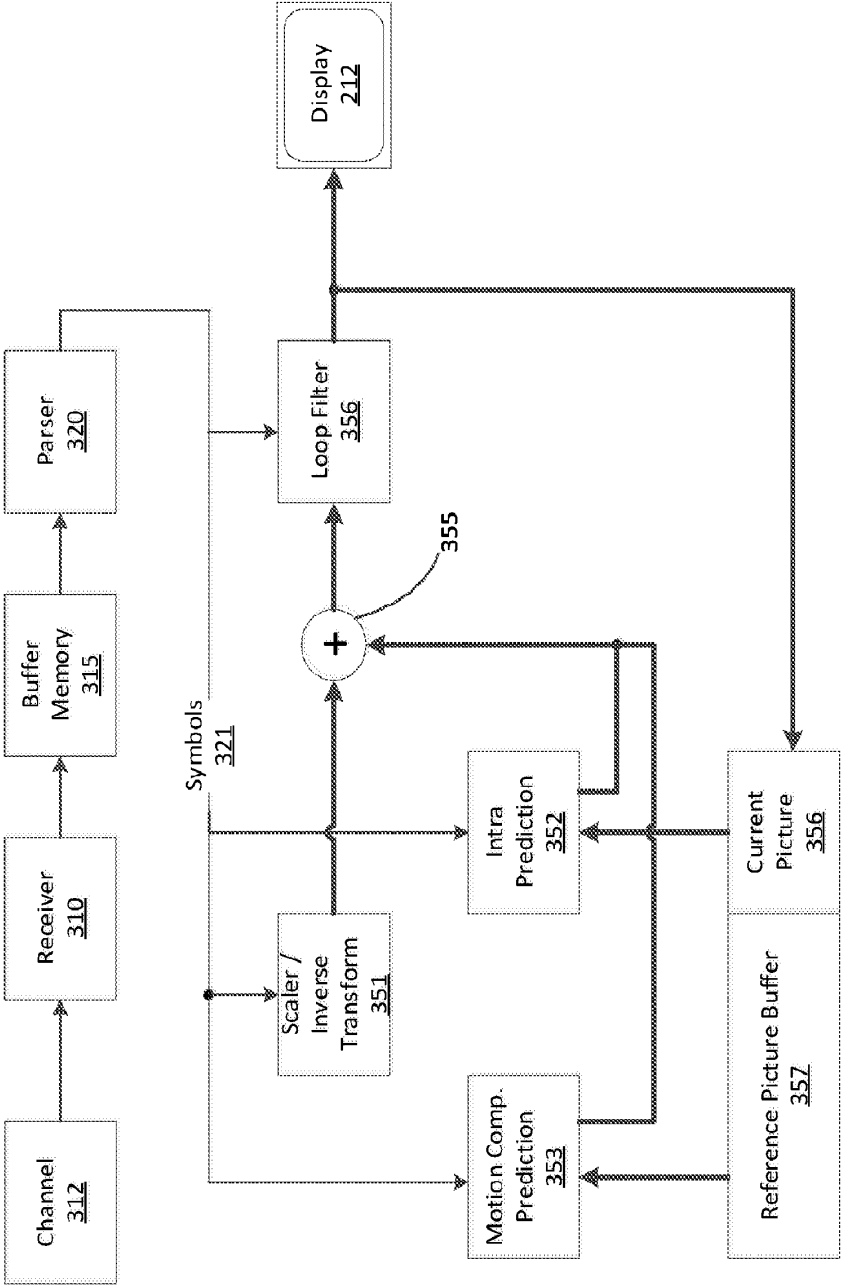
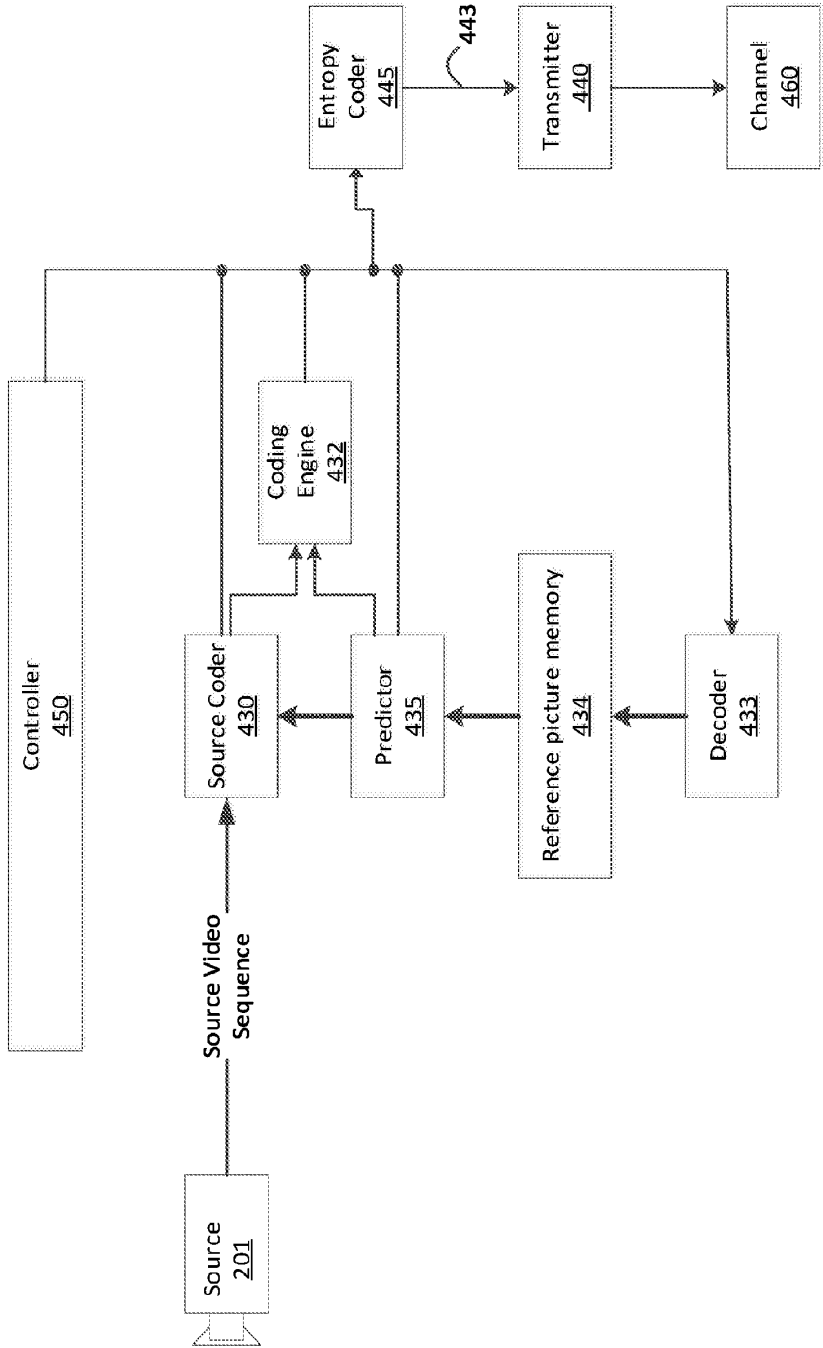
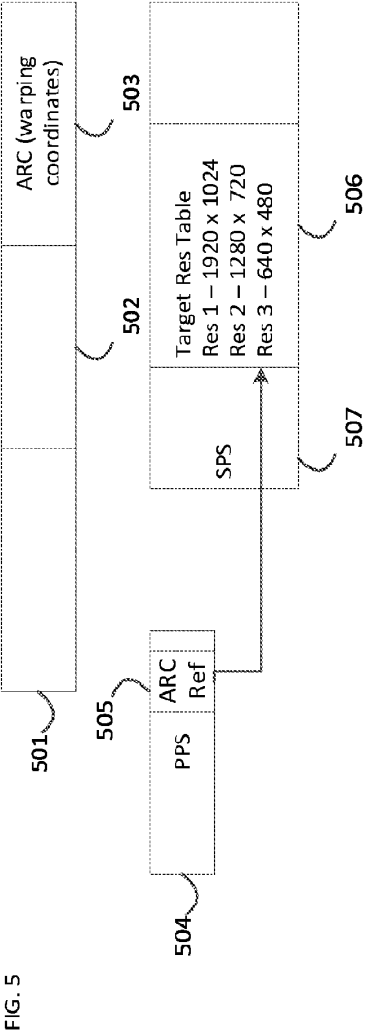


FIG. 4 Encoder 203





Disclosed Subject
Matter

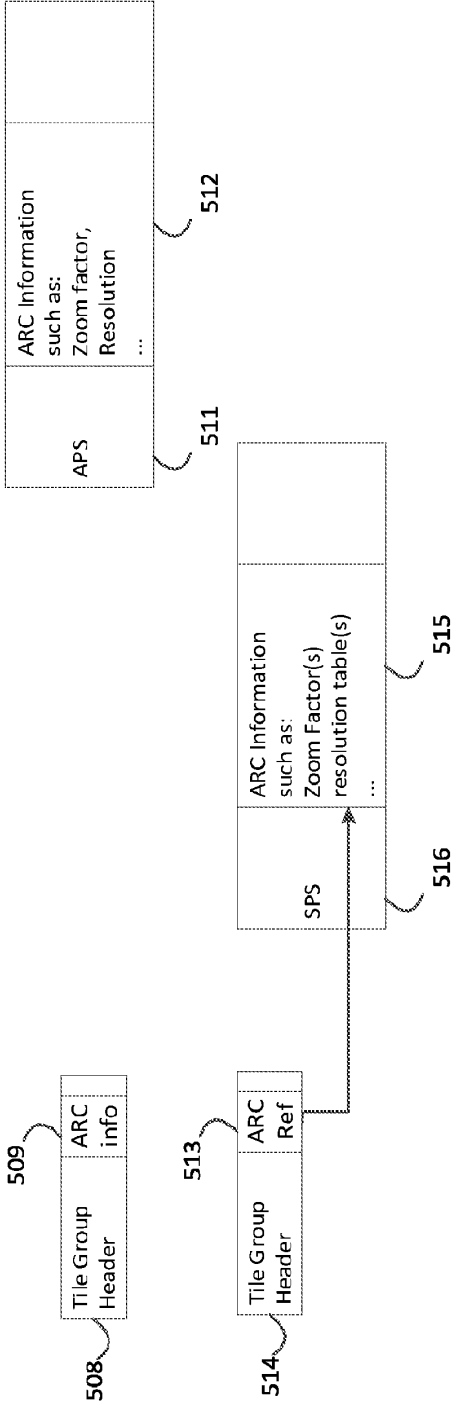


Fig. 6

601	tile_group_header() {		
	...		
603	if(adaptive_pic_resolution_change_flag) {		
602	dec_pic_size_idx		u(1)
	}		
	...		
	}		

610	seq_parameter_set_rbsp() {		Descriptor
	...		
611	adaptive_pic_resolution_change_flag		u(1)
612	if(adaptive_pic_resolution_change_flag) {		
	output_pic_width_in_luma_samples		ue(v)
613	output_pic_height_in_luma_samples		ue(v)
	reference_pic_size_present_flag		u(1)
614	if(reference_pic_size_present_flag)		
	{		
	reference_pic_width_in_luma_samples		ue(v)
	reference_pic_height_in_luma_samples		ue(v)
	}		
	num_dec_pic_size_in_luma_samples_minus1		ue(v)
616	for(i = 0; i <= num_dec_pic_size_in_luma_samples_minus1; i++) {		
	dec_pic_width_in_luma_samples[i]		ue(v)
617	dec_pic_height_in_luma_samples[i]		ue(v)
	}		
	}		
	...		
	}		

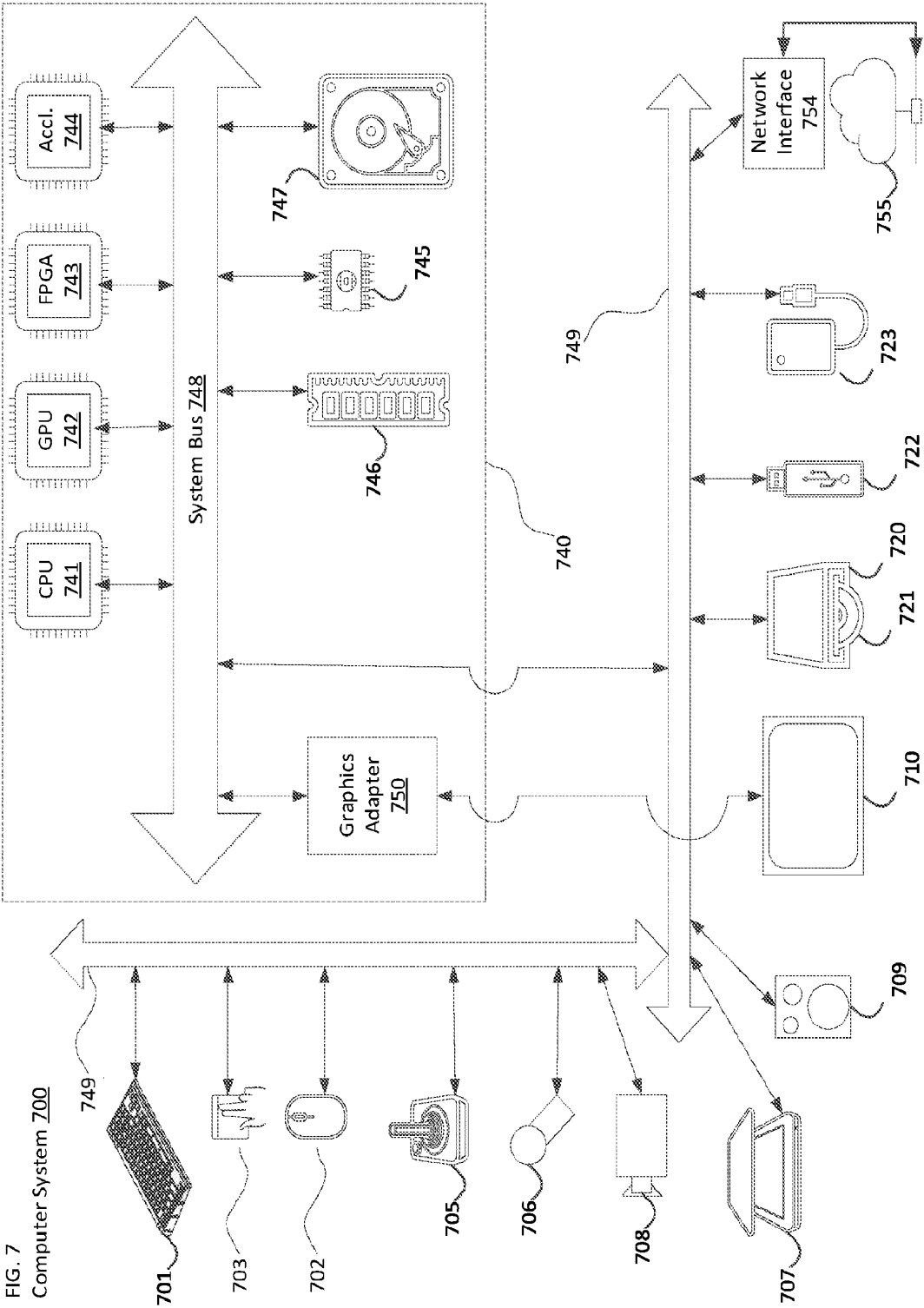


FIG. 7
Computer System 700

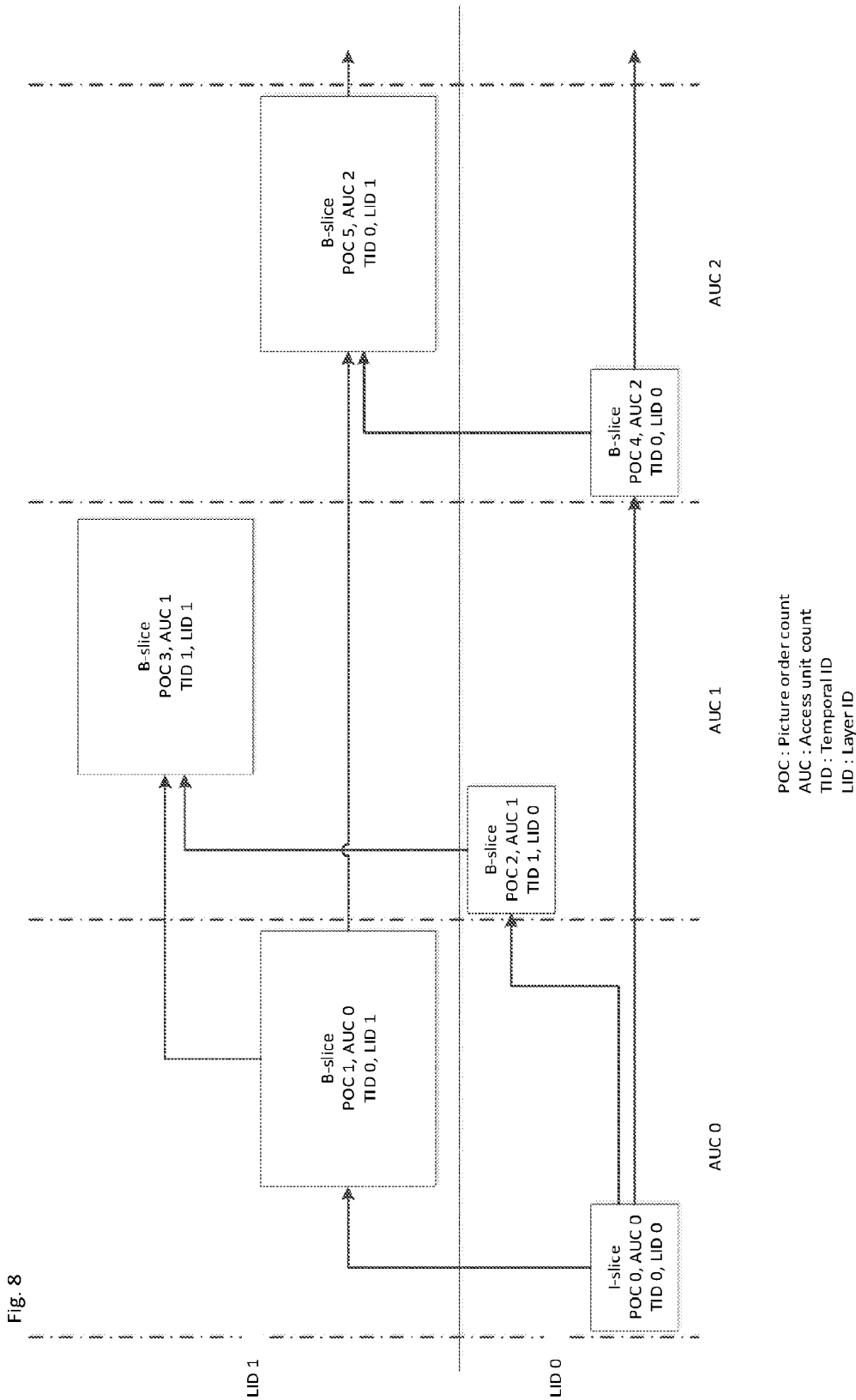


Fig. 9

	Descriptor
video_parameter_set_id { xps_video_parameter_set_id xps_max_layers_minus1 for(i = 0; i <= xps_max_layers_minus1; i++) { xps_included_layer_id[i] xps_reserved_zero_bit } xps_constraint_info_present_flag xps_constant_poc_cycle_per_an if(xps_constant_poc_cycle_per_an) xps_poc_cycle_an ... }	u(4) u(8) s u(7) u(1) s u(1) u(1) s u(8) s s
slice_header { slice_pic_parameter_set_id if(rect_slice_flag == NumBrickAtPic > 1) slice_address if(rect_slice_flag && (angle_brick_per_slice_flag) num_brick_in_slice_minus1 slice_type if(NatUIntType == GRA_NUT) recovery_poc_cnr slice_pic_order_cat_lsb ... if(xps_constant_poc_cycle_per_an) slice_poc_cycle_an }	ue(v) s u(v) s ue(v) ue(v) s ue(v) u(v) s s u(8) s s s

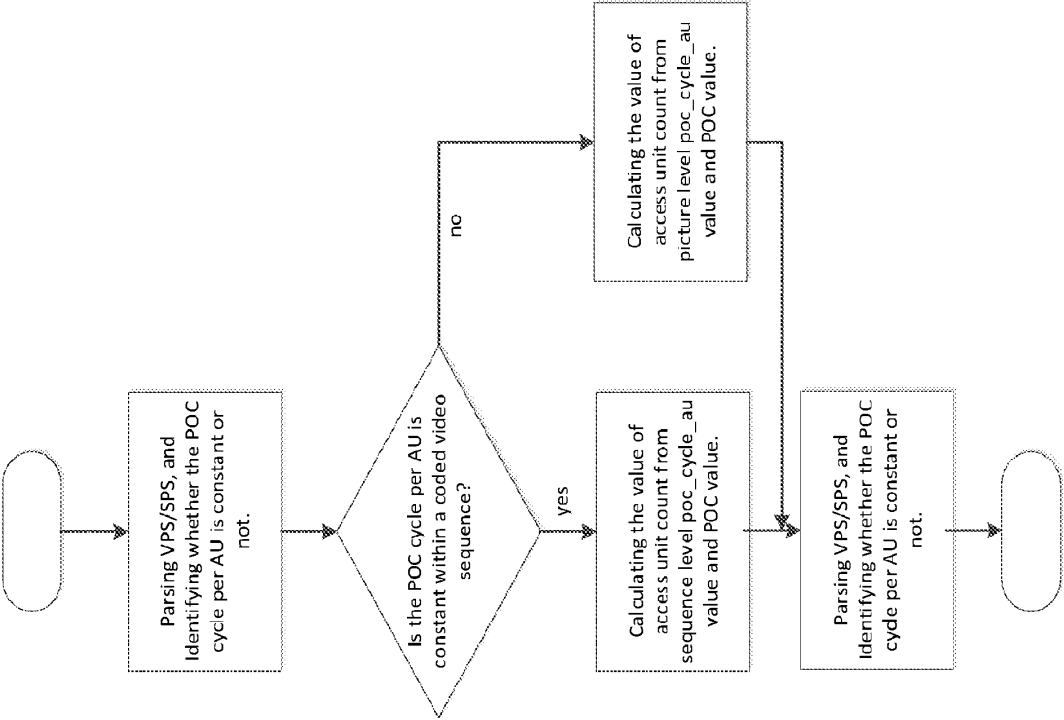
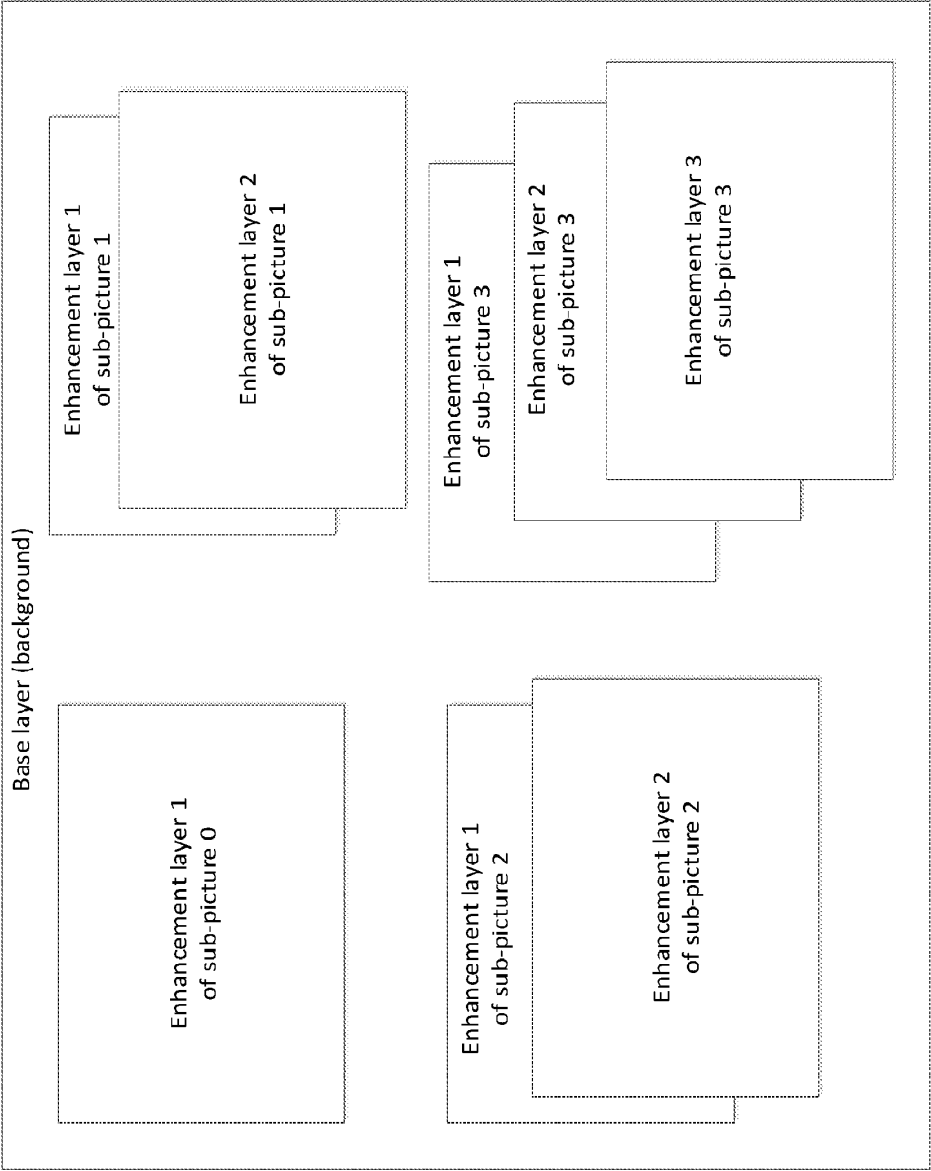


Fig. 10

FIG. 11



12/21

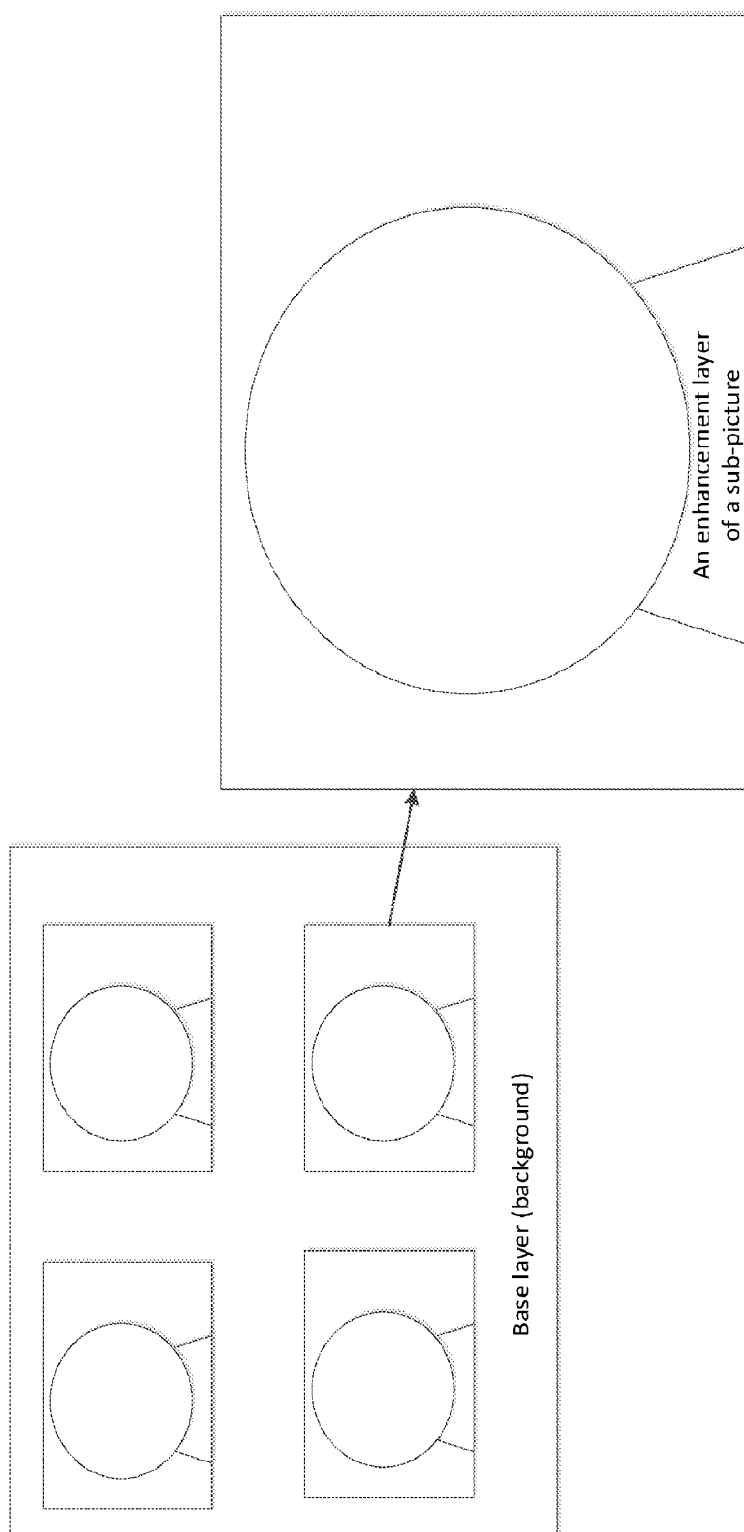
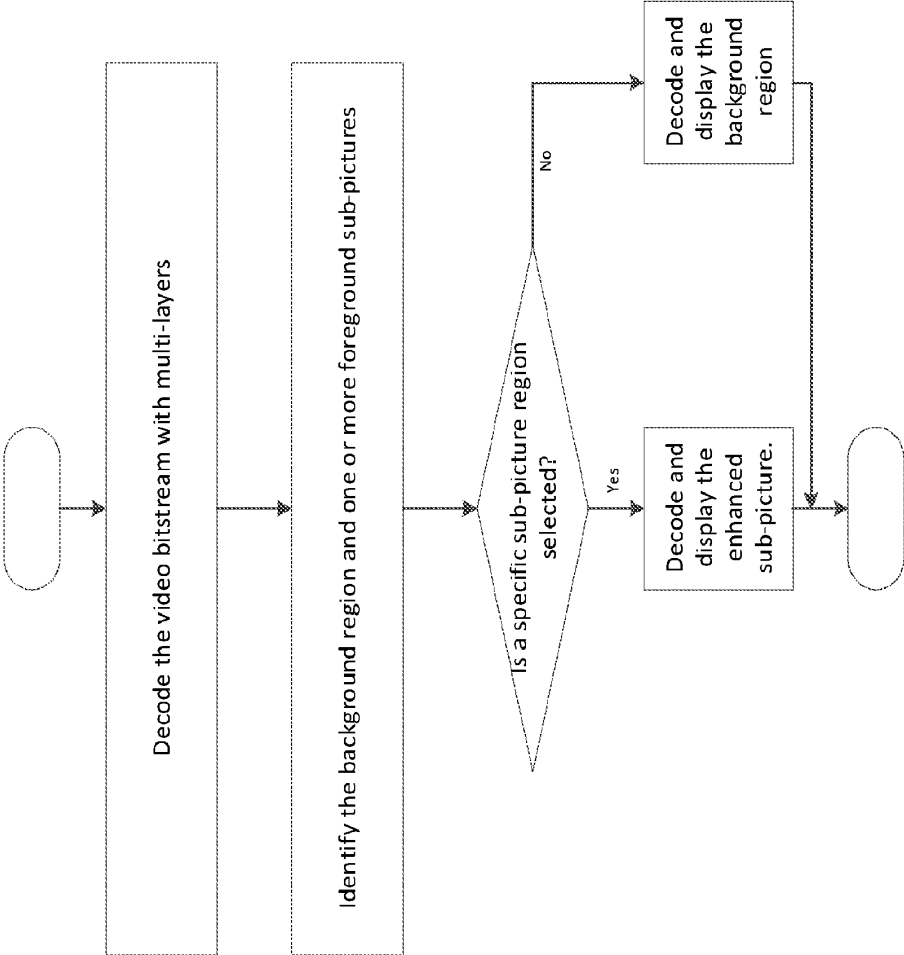


FIG. 12

Fig. 13



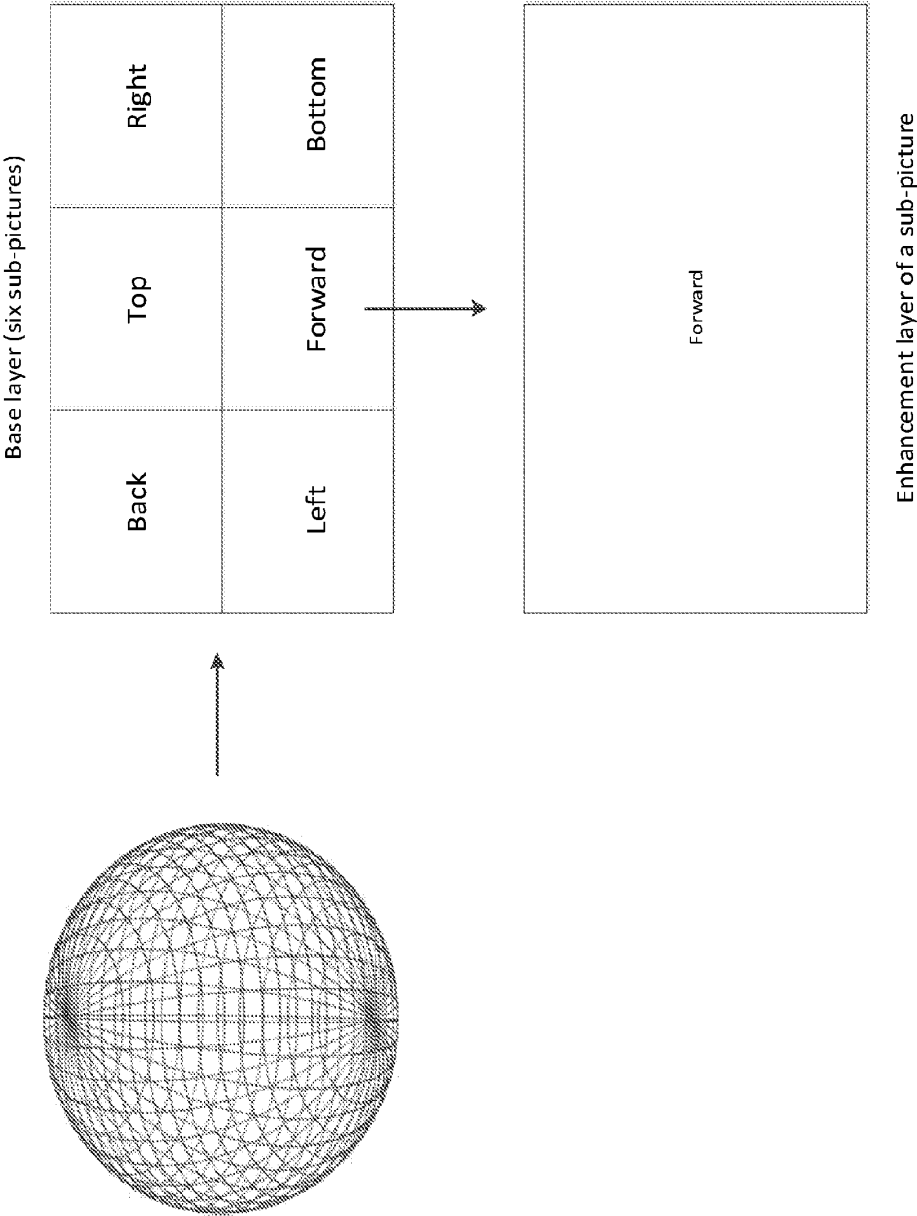


Fig. 14

Fig. 15

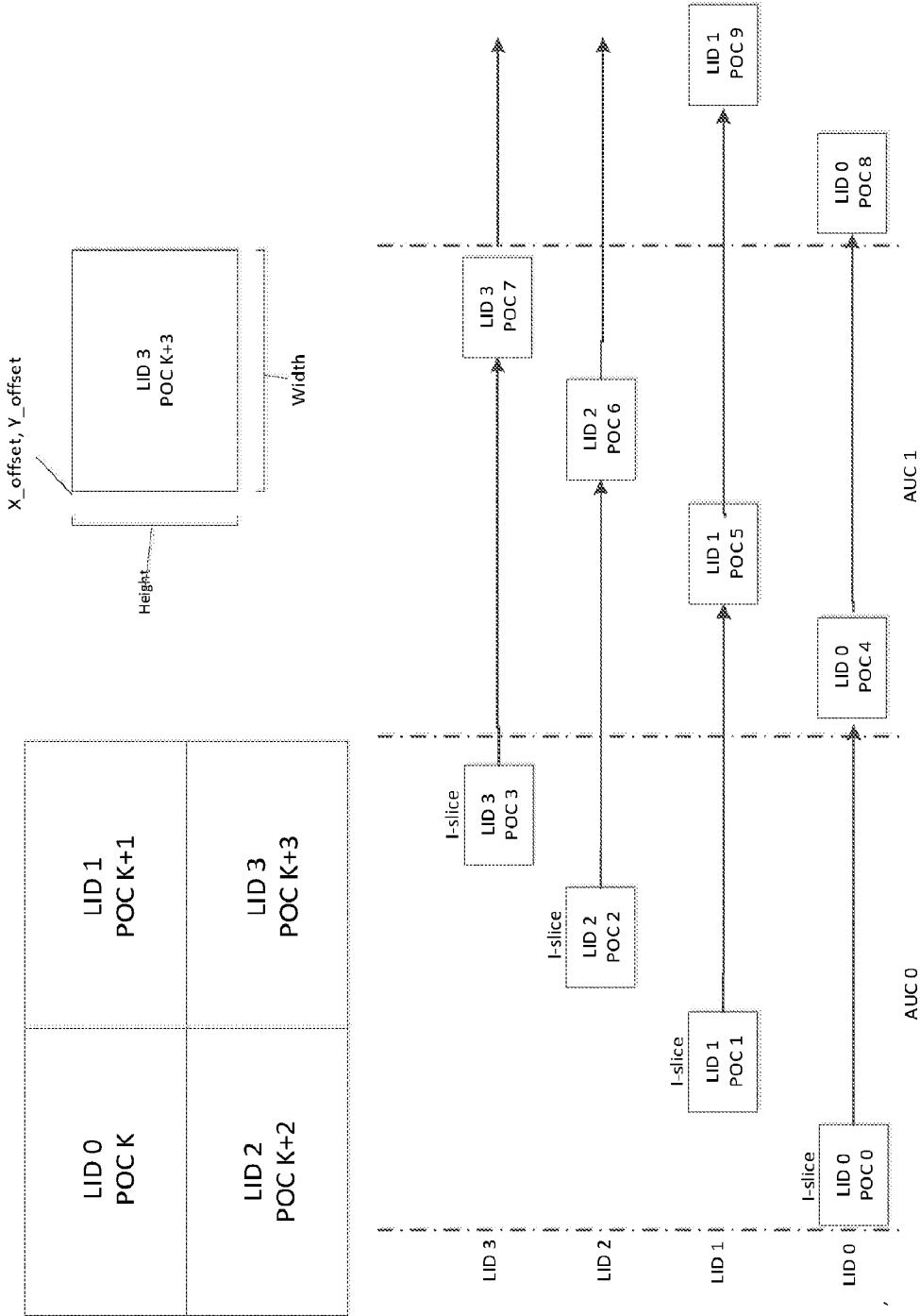


Fig. 16

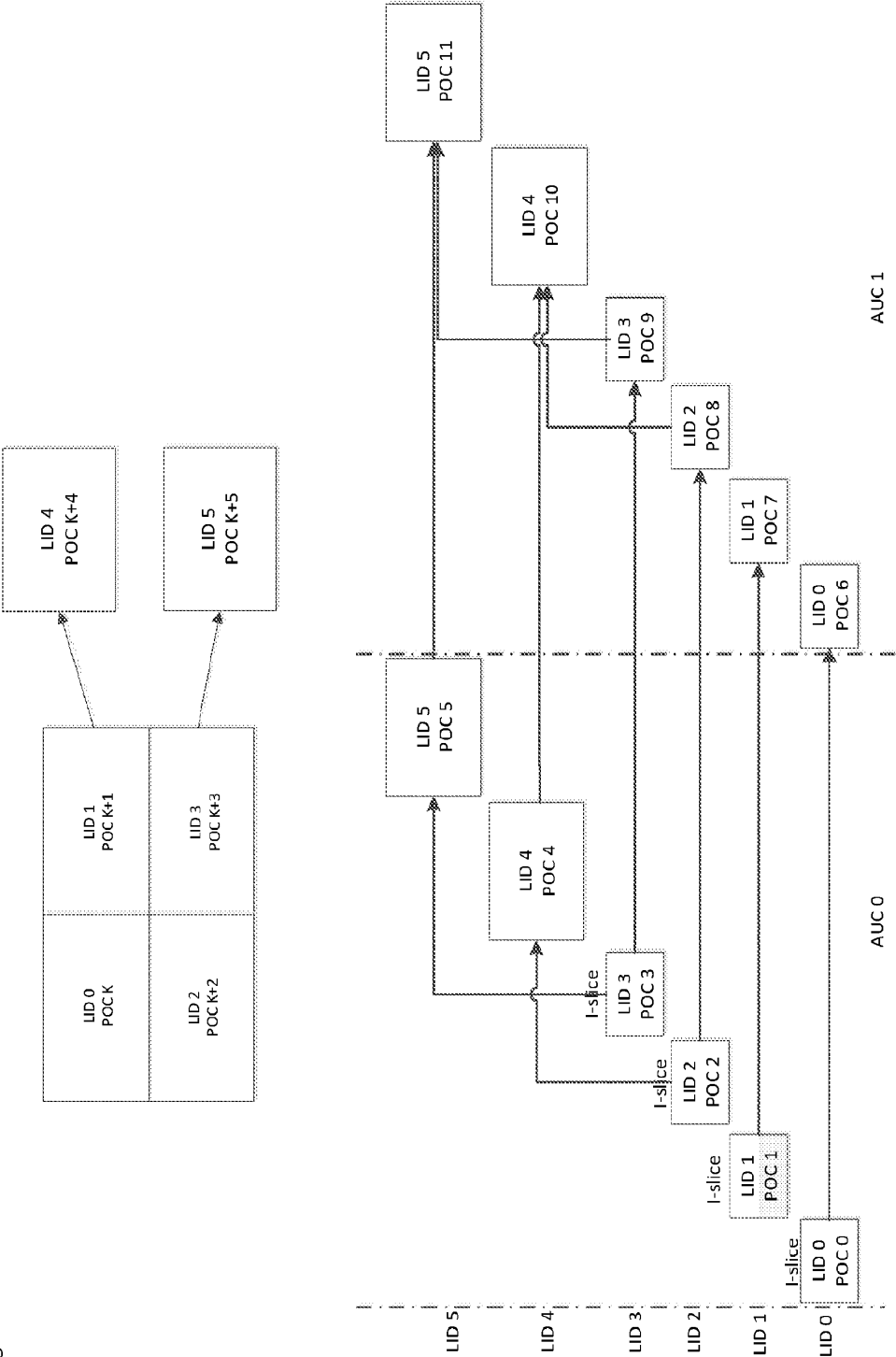


Fig. 17

	Descriptor
video_parameter_set_rbsp() {	u(4)
vps_video_parameter_set_id	u(3)
vps_max_layers_minus1	s
for(i = 0; i <= vps_max_layers_minus1; i++) {	
vps_included_layer_id[i]	u(7)
vps_reserved_zero_bit	u(1)
}	s
...	s
vps_sub_picture_dividing_flag	u(1)
if(vps_sub_picture_dividing_flag) {	
vps_full_pic_width_in_luma_samples	u(1)
vps_full_pic_height_in_luma_samples	u(1)
}	s
...	s
}	s
seq_parameter_set_rbsp() {	Descriptor
sps_decoding_parameter_set_id	u(4)
sps_video_parameter_set_id	u(4)
vps_max_sub_layers_minus1	u(3)
...	s
pic_width_in_luma_samples	u(1)
pic_height_in_luma_samples	u(1)
if(vps_sub_picture_dividing_flag) {	
pic_offset_x	s
pic_offset_y	u(1)
}	s
...	s
}	s

Fig. 18

sub_region_partitioning_info(payloadSize) { num_sub_region num_layers for(i = 0; i <= num_layers; i++) layer_id [i] for(i = 1; i < num_layers; i++) for(j = 0; j < i; j++) direct_dependency_flag [i][j] for(i = 0; i < num_sub_region; i++) { num_layers_for_region [i] for(j = 0; j < num_layers_for_region [i]; j++) sub_region_layer_id [i][j] sub_region_offset_x [i] sub_region_offset_y [i] sub_region_width [i] sub_region_height [i] } ... }	Descriptor u2(v) u2(v) v u2(v) v v u2(v) v u2(v) u2(v) u2(v) u2(v) u2(v) v v v
---	--

Fig. 19

	Descriptor
video_parameter_set_rbsp() {	
...	
vps_max_layers_minus1	u(6)
num_output_layer_sets	ue(v)
num_profile_tier_level	ue(v)
for(i = 0; i < num_profile_tier_level; i++)	
profile_tier_level(vps_max_sub_layers_minus1)	
for(i = 0; i < num_output_layer_sets; i++)	
for(j = 0; j < NumLayersInIdList[i]; j++) {	
output_layer_flag[i][j]	u(1)
profile_tier_level_id[i][j]	u(v)
}	
...	
}	

Fig. 20

	Descriptor
video_parameter_set_rbsp() {	
...	
vps_max_layers_minus1	u(6)
if(vps_max_layers_minus1 > 0) {	
num_output_layer_sets	ue(v)
num_profile_tier_level	ue(v)
}	
for(i = 0; i < num_profile_tier_level; i++)	
profile_tier_level(vps_max_sub_layers_minus1)	
for(i = 0; i < num_output_layer_sets; i++) {	
vps_output_layers_mode[i]	u(2)
vps_ptl_signal_flag[i]	u(1)
for(j = 0; j < NumLayersInIdList[i]; j++) {	
if(vps_output_layers_mode[i] == 2)	
output_layer_flag[i][j]	u(1)
if(vps_ptl_signal_flag[i])	
profile_tier_level_idx[i][j]	u(v)
}	
}	
...	
}	

Fig. 21

video parameter set rsp() {	Descriptor
...	
vps_max_layers_minus1	u(6)
if(vps_max_layers_minus1 > 0) {	
num_output_layer_sets	ue(v)
num_profile_tier_level	ue(v)
}	
max_subpics_minus1	u(8)
for(i = 0; i < max_subpics_minus1; i++) {	
sub_pic_id[i]	u(8)
}	
for(i = 0; i < num_profile_tier_level; i++)	
profile_tier_level(vps_max_sub_layers_minus1)	
for(i = 0; i < num_output_layer_sets; i++) {	
vps_output_layers_mode[i]	u(2)
vps_ptl_signal_flag[i]	u(1)
for(j = 0; j < NumLayersInList[i]; j++) {	
num_output_subpic_layer_fill[j]	ue(v)
for(k = 0; k < num_output_subpic_layer_fill[j]; k++)	
sub_pic_id_layer_fill[j][k]	u(8)
if(vps_output_layers_mode[j] == 2)	
output_layer_flag[i][j]	u(1)
if(vps_ptl_signal_flag[j])	
profile_tier_level_idx[i][j]	u(v)
}	
}	
...	
}	