

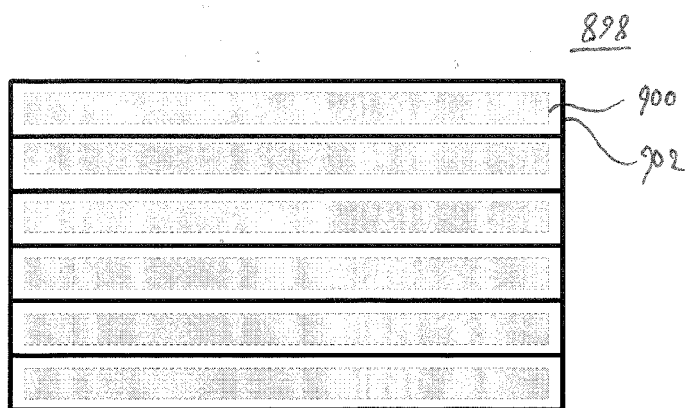


Fig. 1

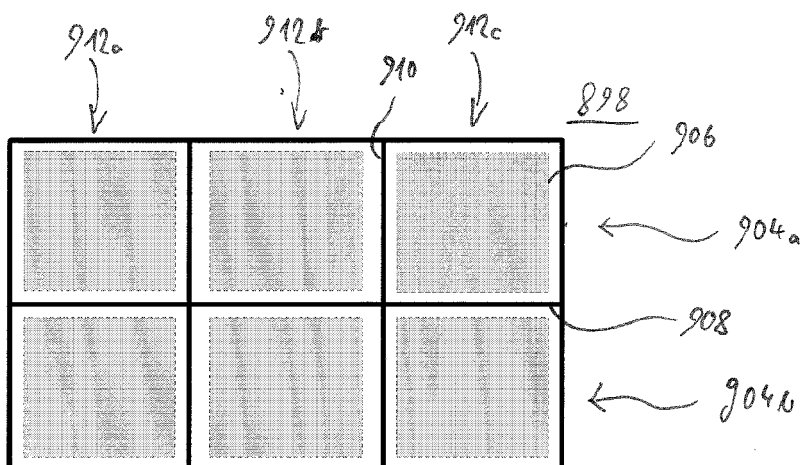


Fig. 2

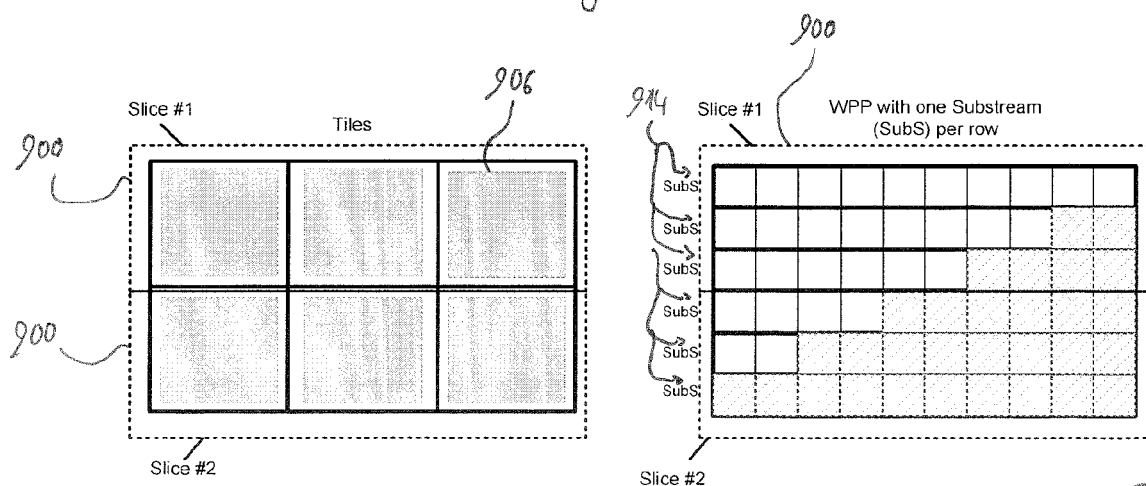


Fig. 3

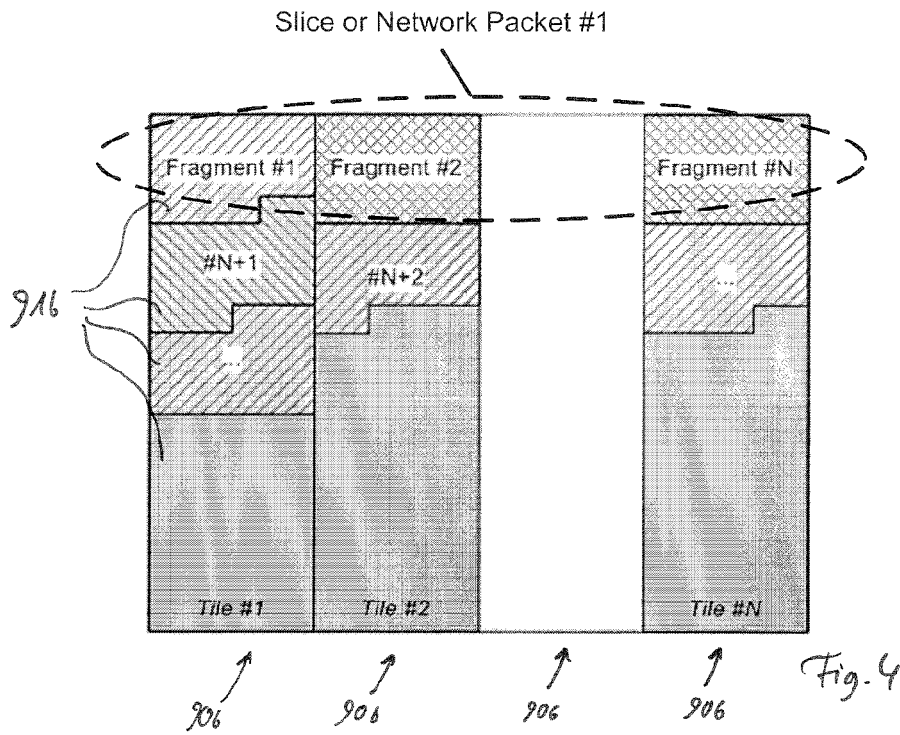
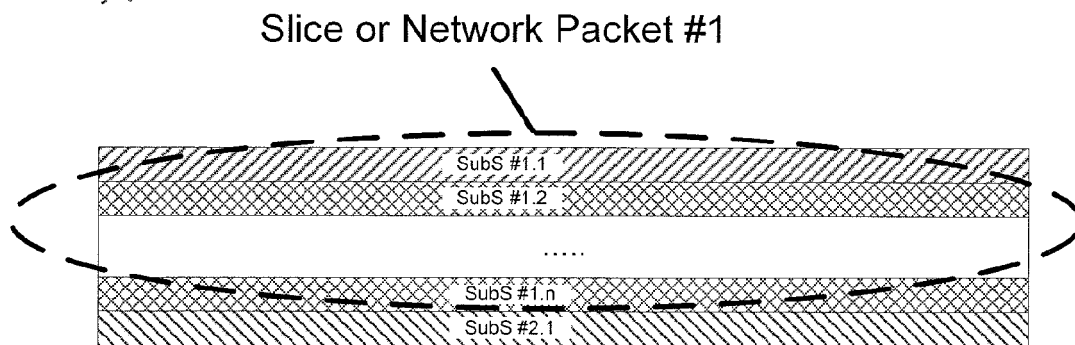


Fig. 5

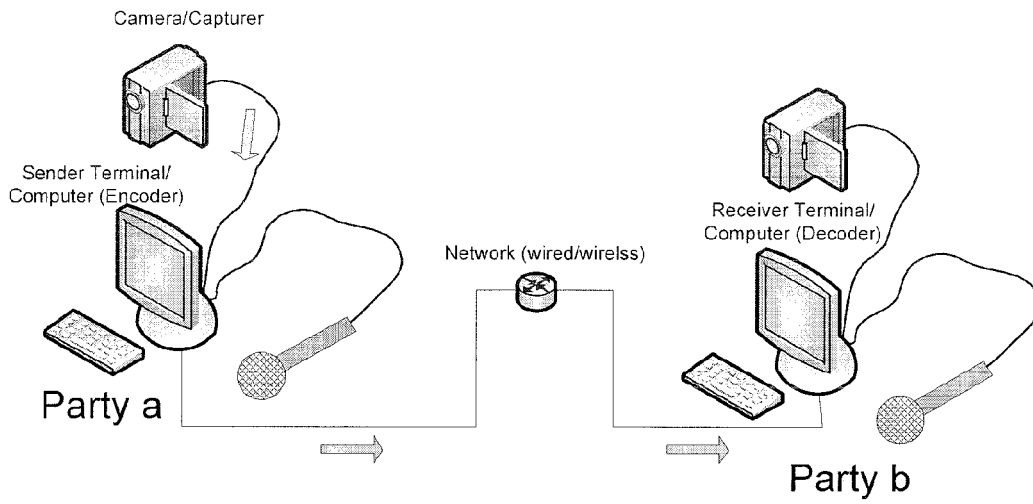


Fig.6

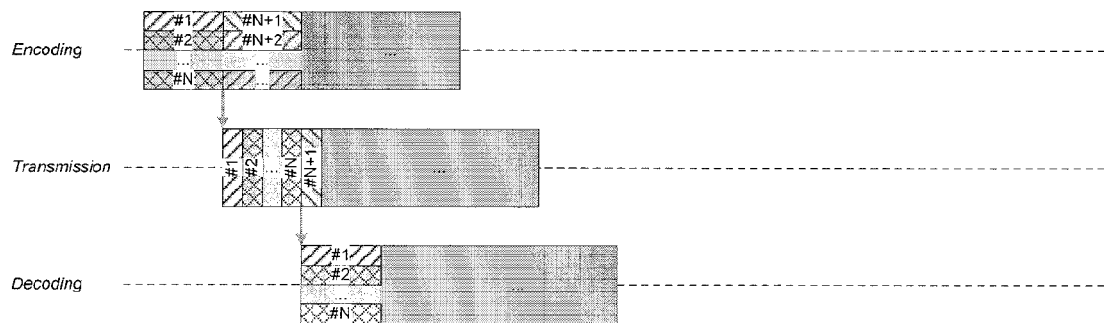


Fig.7

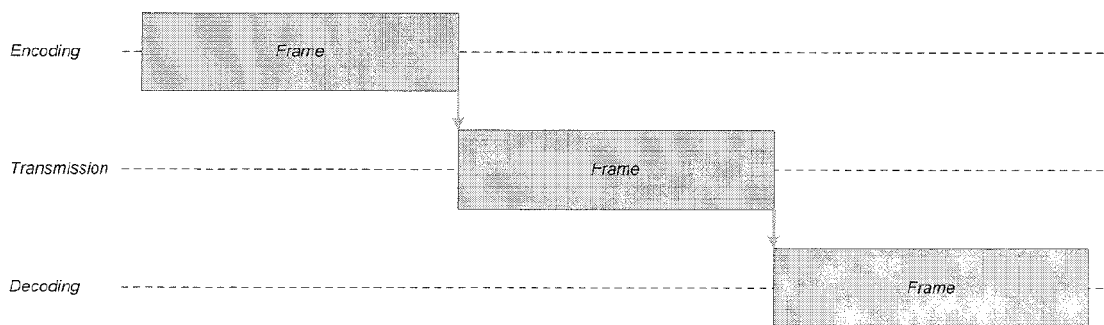


Fig.8

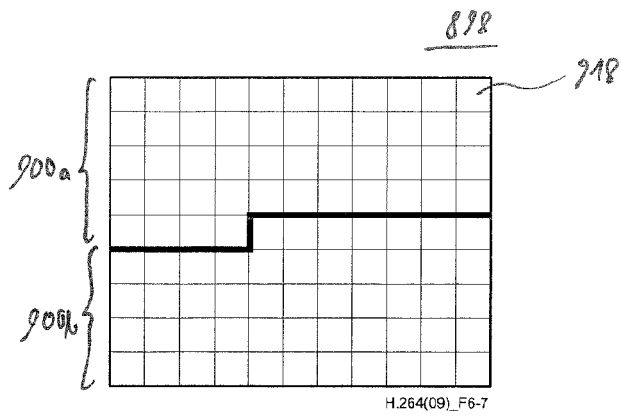


Fig. 9

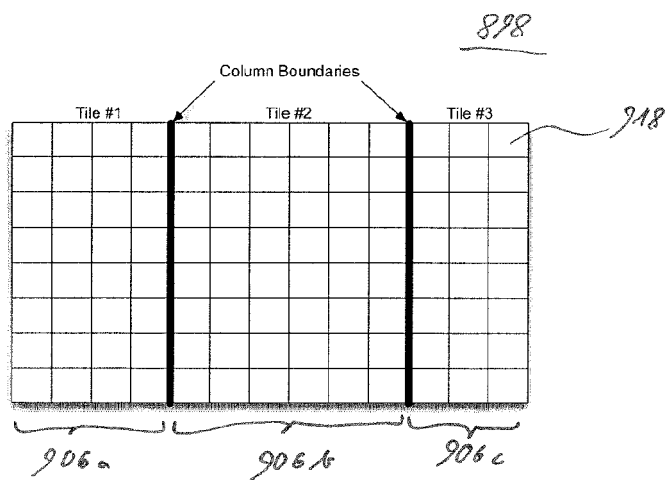


Fig. 10

scaling_list_enable_flag	u(1)
chroma_pred_from_luma_enabled_flag	u(1)
deblocking_filter_in_aps_enabled_flag	u(1)
seq_loop_filter_across_slices_enabled_flag	u(1)
asymmetric_motion_partitions_enabled_flag	u(1)
non_square_quadtree_enabled_flag	u(1)
sample_adaptive_offset_enabled_flag	u(1)
adaptive_loop_filter_enabled_flag	u(1)
if adaptive_loop_filter_enabled_flag	
alf_coef_in_slice_flag	u(1)
if pcm_enabled_flag	
pcm_loop_filter_disable_flag	u(1)
temporal_id_nesting_flag	u(1)
if log2_min_coding_block_size_minus3 == 0	
inter_4x4_enabled_flag	u(1)
num_short_term_ref_pic_sets	ue(v)
for(i = 0; i < num_short_term_ref_pic_sets; i++)	
short_term_ref_pic_set(i)	
long_term_ref_pics_present_flag	u(1)
tiles_or_entropy_coding_sync_idc	u(2)
if(tiles_or_entropy_coding_sync_idc == 1) {	
num_tile_columns_minus1	ue(v)
num_tile_rows_minus1	ue(v)
uniform_spacing_flag	u(1)
if(uniform_spacing_flag) {	
for(i = 0; i < num_tile_columns_minus1; i++)	
column_width[i]	ue(v)
for(i = 0; i < num_tile_rows_minus1; i++)	
row_height[i]	ue(v)
}	
loop_filter_across_tiles_enabled_flag	u(1)
}	
vui_parameters_present_flag	u(1)
if(vui_parameters_present_flag)	
vui_parameters()	
sps_extension_flag	u(1)
if(sps_extension_flag)	
while(more_rbsp_data())	
sps_extension_data_flag	u(1)
rbsp_trailing_bits()	
}	

seq_parameter_set_rbsp() {		Descriptor
profile_idc		u(8)
reserved_zero_8bits /* equal to 0 */		u(8)
level_idc		u(8)
seq_parameter_set_id		ue(v)
chroma_format_idc		ue(v)
if(chroma_format_idc == 3)		
separate_colour_plane_flag		u(1)
max_temporal_layers_minus1		u(3)
pic_width_in_luma_samples		ue(v)
pic_height_in_luma_samples		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_bit_depth_luma_minus1		u(4)
pcm_bit_depth_chroma_minus1		u(4)
}		
qp_prime_y_zero_transquant_bypass_flag		u(1)
log2_max_pic_order_cnt_lsb_minus4		ue(v)
for(i = 0; i <= max_temporal_layers_minus1; i++) {		
max_dec_pic_buffering[i]		ue(v)
num_reorder_pics[i]		ue(v)
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		ue(v)
log2_diff_max_min_coding_block_size		ue(v)
log2_min_transform_block_size_minus2		ue(v)
log2_diff_max_min_transform_block_size		ue(v)
if(pcm_enabled_flag) {		
log2_min_pcm_coding_block_size_minus3		ue(v)
log2_diff_max_min_pcm_coding_block_size		ue(v)
}		
max_transform_hierarchy_depth_inter		ue(v)
max_transform_hierarchy_depth_intra		ue(v)

Fig. 1A

pic_parameter_set_rbsp() {	Descriptor
pic_parameter_set_id	ue(v)
seq_parameter_set_id	ue(v)
sign_data_hiding_flag	u(1)
if(sign_data_hiding_flag)	
sign_hiding_threshold	u(4)
cabac_init_present_flag	u(1)
num_ref_idx_l0_default_active_minus1	ue(v)
num_ref_idx_l1_default_active_minus1	ue(v)
pic_init_qp_minus26	se(v)
constrained_intra_pred_flag	u(1)
enable_temporal_mvp_flag	u(1)
slice_granularity	u(2)
max_cu_qp_delta_depth	ue(v)
cb_qp_offset	se(v)
cr_qp_offset	se(v)
weighted_pred_flag	u(1)
weighted_bipred_idc	u(2)
output_flag_present_flag	u(1)
if(tiles_or_entropy_coding_sync_idc == 1) {	
tile_info_present_flag	u(1)
tile_control_present_flag	u(1)
if(tile_info_present_flag) {	
num_tile_columns_minus1	ue(v)
num_tile_rows_minus1	ue(v)
uniform_spacing_flag	u(1)
if(!uniform_spacing_flag) {	
for(i = 0; i < num_tile_columns_minus1; i++)	
column_width[i]	ue(v)
for(i = 0; i < num_tile_rows_minus1; i++)	
row_height[i]	ue(v)
}	
}	
if(tile_control_present_flag)	
loop_filter_across_tiles_enabled_flag	u(1)
} else if(tiles_or_entropy_coding_sync_idc == 2)	
num_substreams_minus1	ue(v)
deblocking_filter_control_present_flag	u(1)
if(slice_type == P slice_type == B)	
log2_parallel_merge_level_minus2	ue(v)
pps_extension_flag	u(1)
if(pps_extension_flag)	
while(more_rbsp_data())	
pps_extension_data_flag	u(1)
rbsp_trailing_bits()	
}	

Fig. 12

Fig. 13

if (slice_type == P slice_type == B) {	
num_ref_idx_active_override_flag	u(1)
if (num_ref_idx_active_override_flag) {	
num_ref_idx_l0_active_minus1	ue(v)
if (slice_type == B)	
num_ref_idx_l1_active_minus1	ue(v)
}	
if (lists_modification_present_flag) {	
ref_pic_list_modification()	
ref_pic_list_combination()	
}	
if (slice_type == B)	
mvd_l1_zero_flag	u(1)
}	
if (cabac_init_present_flag && slice_type != I)	
cabac_init_flag	u(1)
if (entropy_slice_flag) {	
slice_qp_delta	se(v)
if (deblocking_filter_control_present_flag) {	
if (deblocking_filter_in_aps_enabled_flag)	
inherit_dbl_params_from_aps_flag	u(1)
if (inherit_dbl_params_from_aps_flag) {	
disable_deblocking_filter_flag	u(1)
if (disable_deblocking_filter_flag) {	
beta_offset_div2	se(v)
te_offset_div2	se(v)
}	
}	
if (slice_type == B)	
collocated_from_l0_flag	u(1)
if (slice_type != I &&	
((collocated_from_l0_flag && num_ref_idx_l0_active_minus1 > 0)	
(collocated_from_l0_flag && num_ref_idx_l1_active_minus1 > 0))	
collocated_ref_idx	ue(v)
if ((weighted_pred_flag && slice_type == P)	
(weighted_bipred_idc == 1 && slice_type == B))	
pred_weight_table()	
}	
if (slice_type == P slice_type == B)	
five_minus_max_num_merge_cand	ue(v)
if (adaptive_loop_filter_enabled_flag) {	
slice_adaptive_loop_filter_flag	u(1)
if (slice_adaptive_loop_filter_flag && alf_coef_in_slice_flag)	
alf_param()	
if (slice_adaptive_loop_filter_flag && !alf_coef_in_slice_flag)	
alf_cu_control_param()	

Descriptor	
slice_header() {	
first_slice_in_pic_flag	u(1)
if (first_slice_in_pic_flag == 0)	
slice_address	
slice_type	u(v)
entropy_slice_flag	ue(v)
u(1)	
pic_parameter_set_id	ue(v)
if (entropy_slice_flag) {	
if (output_flag_present_flag)	
pic_output_flag	u(1)
if (separate_colour_plane_flag == 1)	
colour_plane_id	u(2)
if (HrdPicFlag) {	
idr_pic_id	ue(v)
no_output_of_prior_pics_flag	u(1)
} else {	
pic_order_cnt_lsb	u(v)
short_term_ref_pic_set_sps_flag	u(1)
if (short_term_ref_pic_set_sps_flag)	
short_term_ref_pic_set(num_short_term_ref_pic_sets)	
} else	
short_term_ref_pic_set_idx	u(v)
if (long_term_ref_pics_present_flag) {	
num_long_term_pics	ue(v)
for (i = 0; i < num_long_term_pics; i++) {	
delta_poc_lsb_l[i]	ue(v)
delta_poc_msb_present_flag[i]	u(1)
if (delta_poc_msb_present_flag[i])	
delta_poc_msb_cycle_l_minus1[i]	ue(v)
used_by_curr_pic_lt_flag[i]	u(1)
}	
}	
if (sample_adaptive_offset_enabled_flag) {	
slice_sao_interleaving_flag	u(1)
slice_sample_adaptive_offset_flag	u(1)
if (slice_sao_interleaving_flag &&	
slice_sample_adaptive_offset_flag) {	
sao_cb_enable_flag	u(1)
sao_or_enable_flag	u(1)
}	
if (scaling_list_enable_flag	
deblocking_filter_in_aps_enabled_flag	
(sample_adaptive_offset_enabled_flag && !slice_sao_interleaving_flag)	
adaptive_loop_filter_enabled_flag)	
aps_id	ue(v)

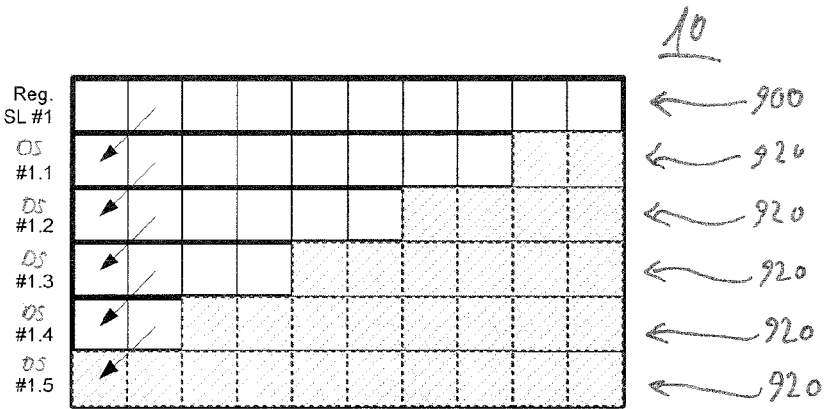


Fig. 14

pic_parameter_set_rbsp() {	Descriptor
.....	
dependent_slices_present_flag	u(1)
last_ctb_cabac_init_flg	u(1)
}	

Fig. 15

10/21

slice_header() {	Descriptor
first_slice_in_pic_flag	u(1)
if(first_slice_in_pic_flag == 0)	
slice_address	u(v)
slice_type	ue(v)
entropy_slice_flag	u(1)
dependent_slice_flag	u(1)
if(!entropy_slice_flag) {	
if(!dependent_slice_flag) {	
.....	
} else {	
no_cabac_reset_flag	u(1)
}	
.....	
if(cabac_init_present_flag && slice_type != I && !no_cabac_reset_flag)	
cabac_init_flag	u(1)
.....	
if(slice_type == P slice_type == B)	
five_minus_max_num_merge_cand	ue(v)
if(adaptive_loop_filter_enabled_flag) {	
slice_adaptive_loop_filter_flag	u(1)
if(slice_adaptive_loop_filter_flag && alf_coef_in_slice_flag)	
alf_param()	
if(slice_adaptive_loop_filter_flag && !alf_coef_in_slice_flag)	
alf_cu_control_param()	
}	
if(seq_loop_filter_across_slices_enabled_flag && (slice_adaptive_loop_filter_flag slice_sample_adaptive_offset_flag !disable_deblocking_filter_flag))	
slice_loop_filter_across_slices_enabled_flag	u(1)
if(tiles_or_entropy_coding_sync_idc > 0) {	
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)
}	
}	
}	
}	

Fig. 16

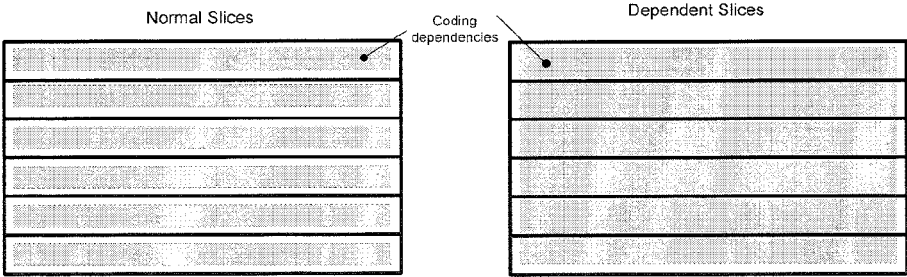
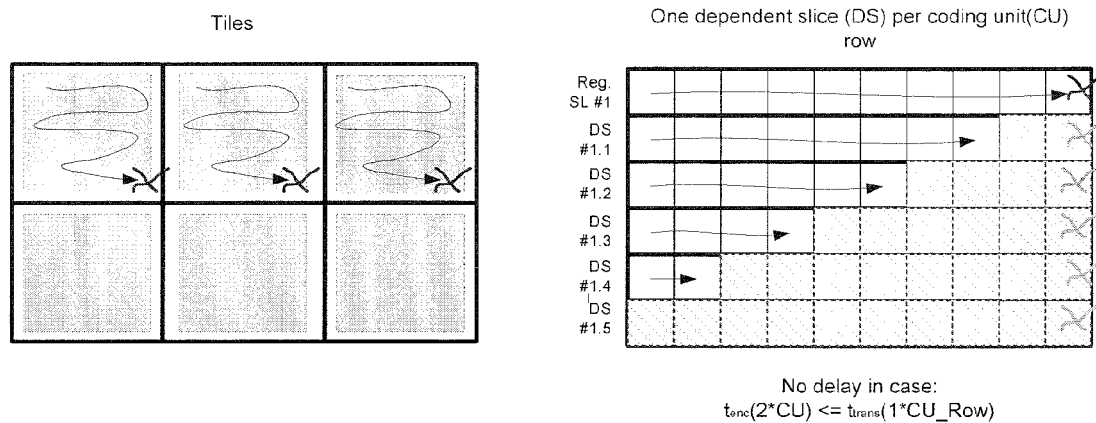
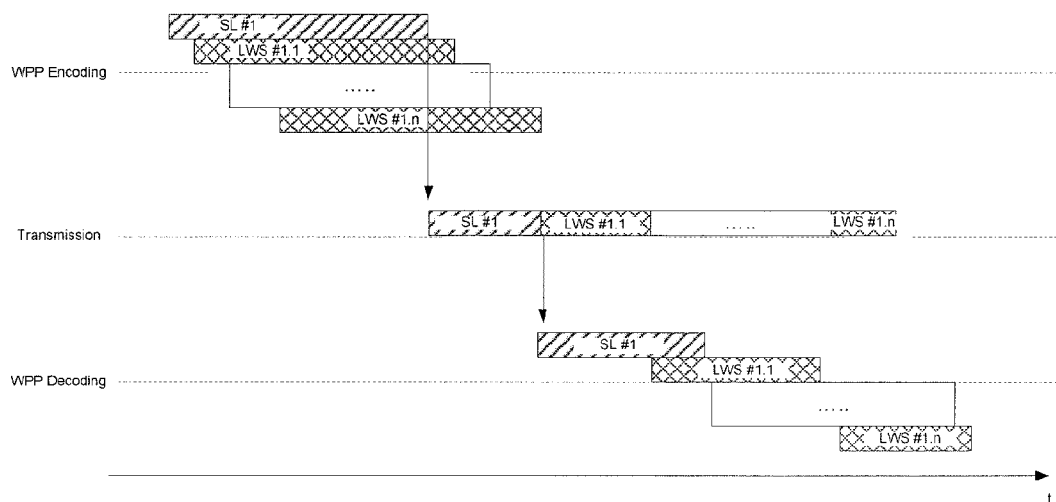
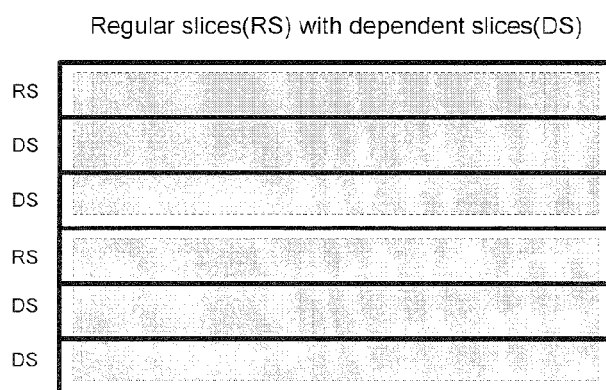
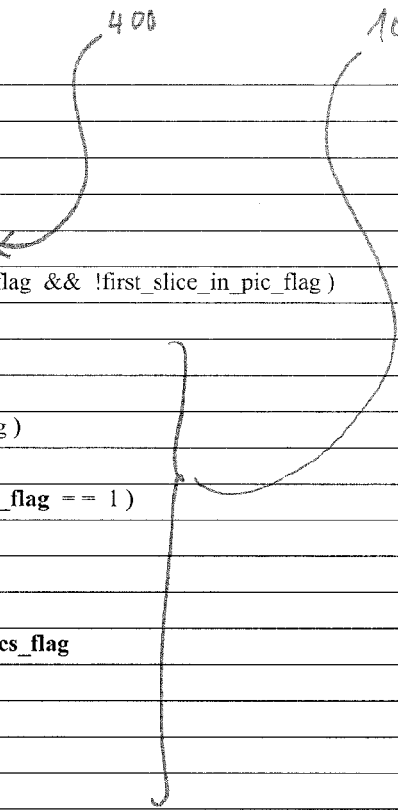


Fig. 17





slice_header() {	Descriptor
first_slice_in_pic_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)
if(output_flag_present_flag)	
pic_output_flag	u(1)
if(separate_colour_plane_flag == 1)	
colour_plane_id	u(2)
if(RapPicFlag) {	
rap_pic_id	ue(v)
no_output_of_prior_pics_flag	u(1)
}	
if(!IdrPicFlag) {	
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)
}	
}	
if(slice_header_extension_present_flag) {	
slice_header_extension_length	ue(v)
for(i = 0; i < slice_header_extension_length; i++)	
slice_header_extension_data_byte	u(8)
}	
byte_alignment()	
}	

Fig. 21

pic_parameter_set_rbsp() {	Descriptor
pic_parameter_set_id	ue(v)
seq_parameter_set_id	ue(v)
sign_data_hiding_flag	u(1)
if(sign_data_hiding_flag)	
sign_hiding_threshold	u(4)
cabac_init_present_flag	u(1)
num_ref_idx_l0_default_active_minus1	ue(v)
num_ref_idx_l1_default_active_minus1	ue(v)
[Ed. (BB): not present in HM software]	
pic_init_qp_minus26	se(v)
constrained_intra_pred_flag	u(1)
slice_granularity	u(2)
diff_cu_qp_delta_depth	ue(v)
cb_qp_offset	se(v)
cr_qp_offset	se(v)
weighted_pred_flag	u(1)
weighted_bipred_idc	u(2)
output_flag_present_flag	u(1)
transquant_bypass_enable_flag	u(1)
dependent_slice_enabled_flag	u(1)
tiles_or_entropy_coding_sync_idc	u(2)
if(tiles_or_entropy_coding_sync_idc == 1) {	
num_tile_columns_minus1	ue(v)
num_tile_rows_minus1	ue(v)
uniform_spacing_flag	u(1)
if(uniform_spacing_flag) {	
for(i = 0; i < num_tile_columns_minus1; i++)	
column_width[i]	ue(v)
for(i = 0; i < num_tile_rows_minus1; i++)	
row_height[i]	ue(v)
}	
loop_filter_across_tiles_enabled_flag	u(1)
} else if(tiles_or_entropy_coding_sync_idc == 3)	
cabac_independent_flag	u(1)
...	
}	

Fig. 22

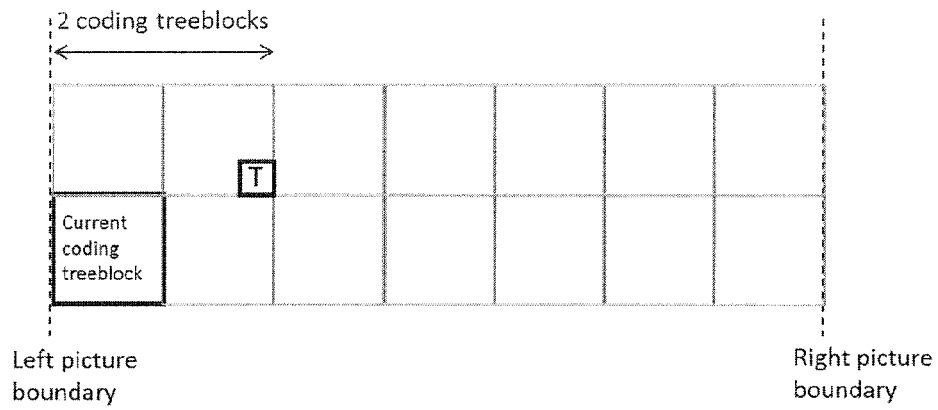
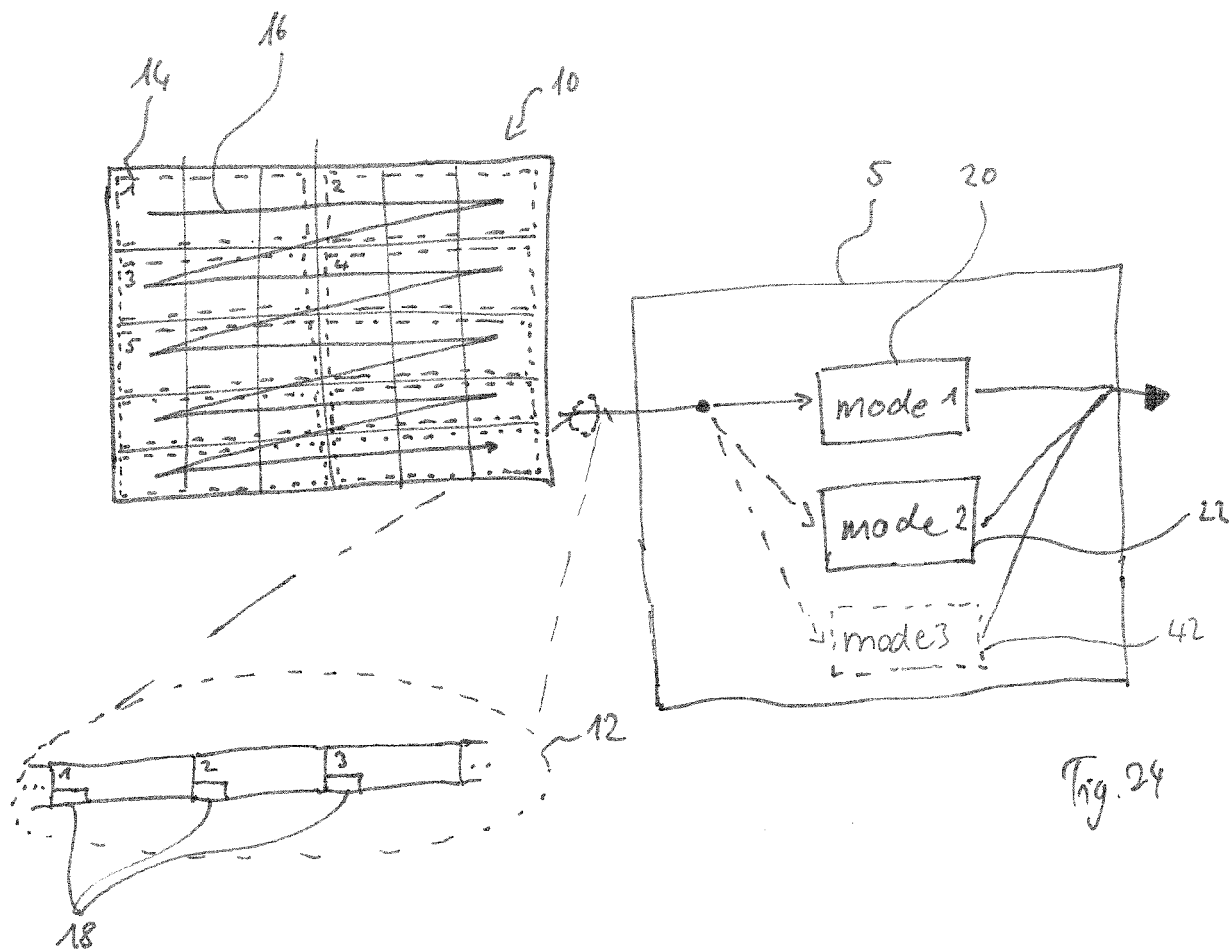
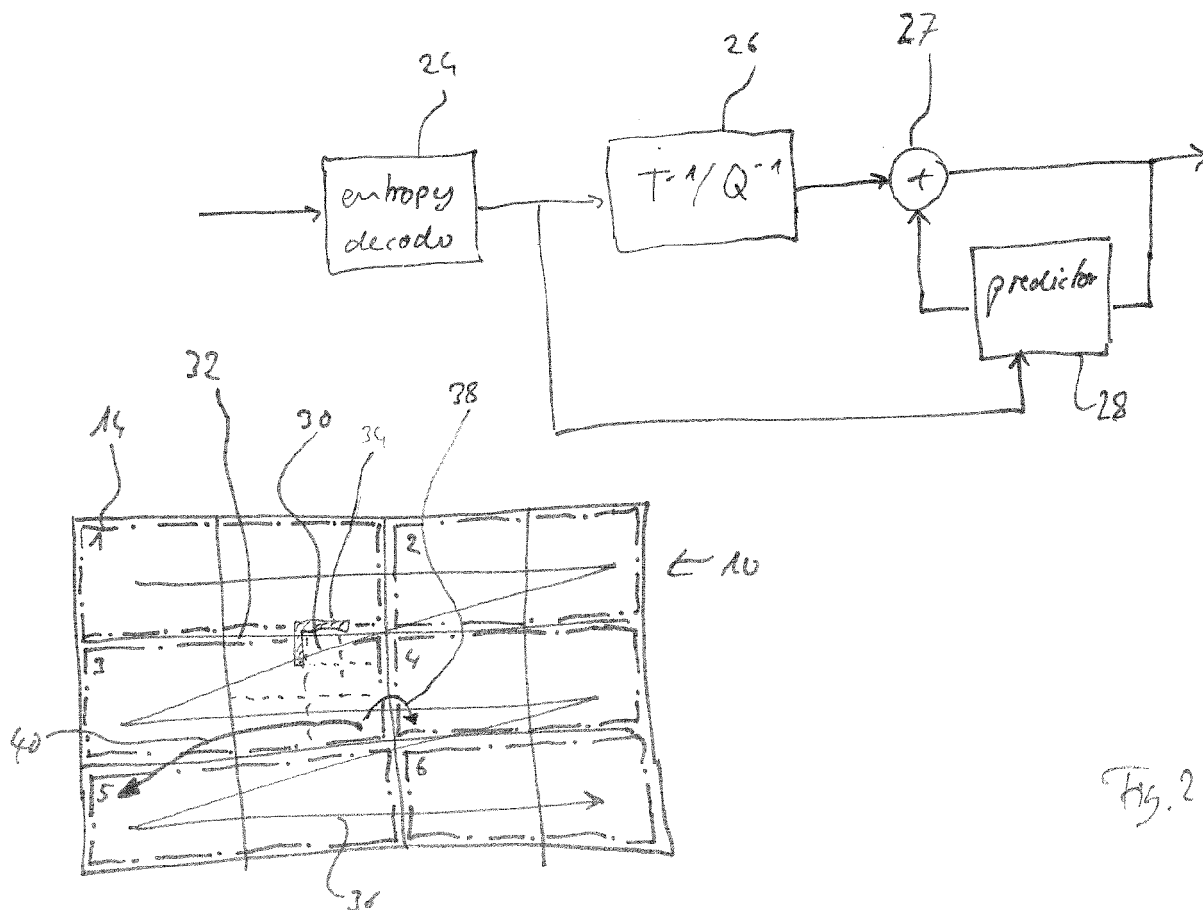


Fig. 23



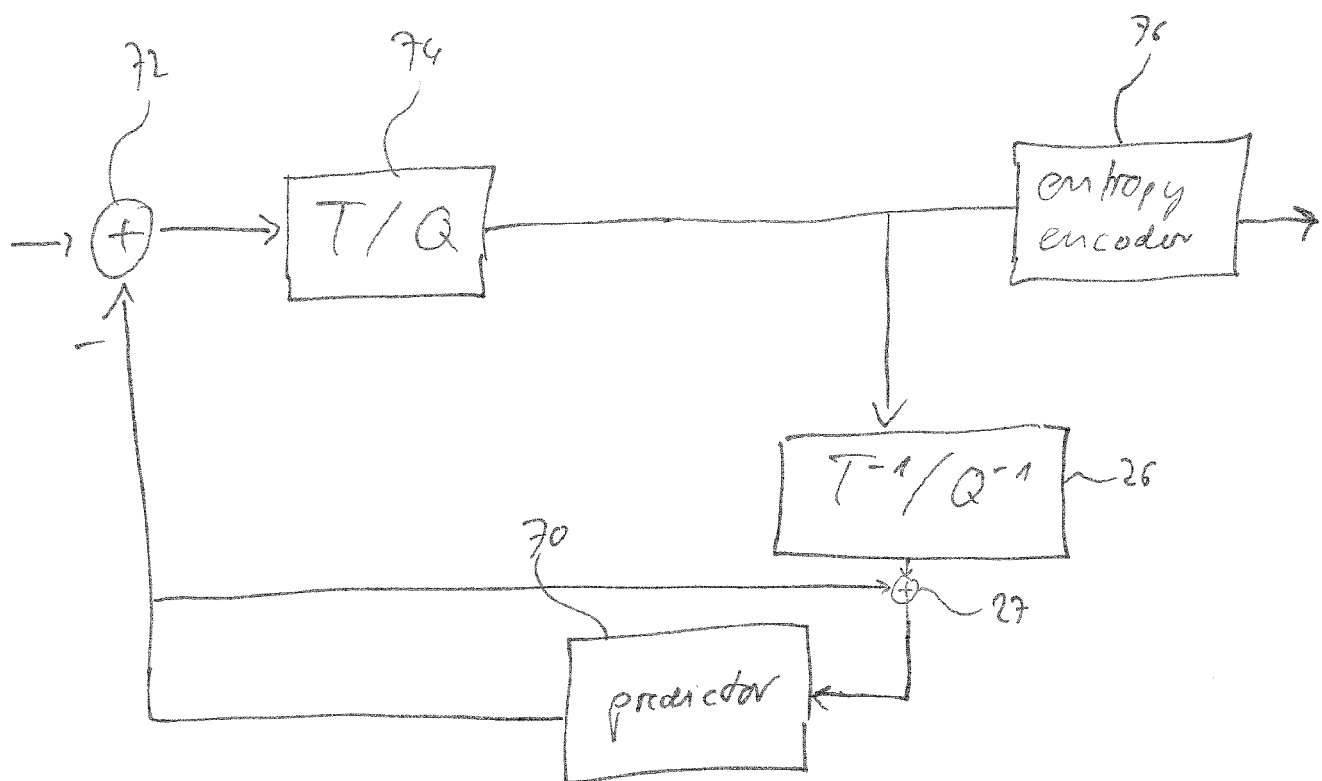


Fig. 26

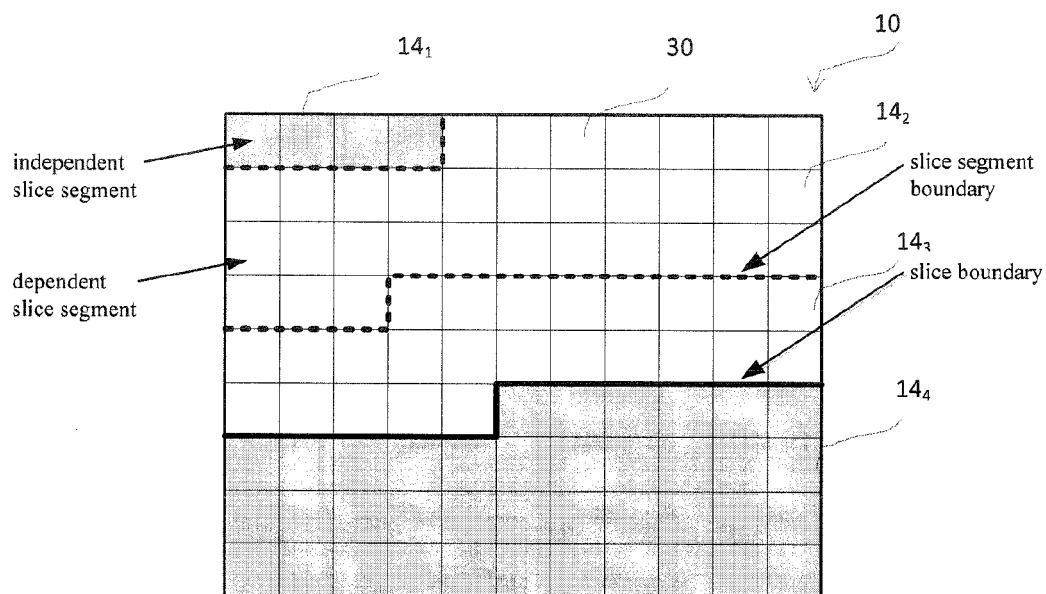


Fig. 27

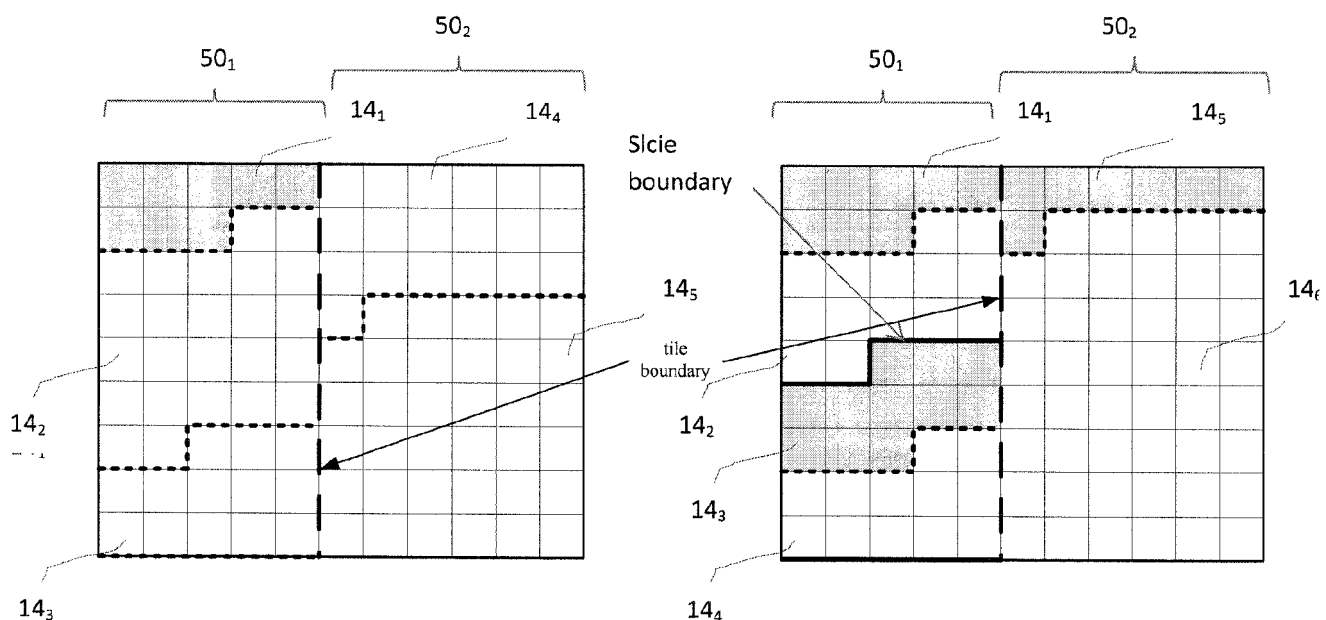


Fig. 28a

Fig. 28b

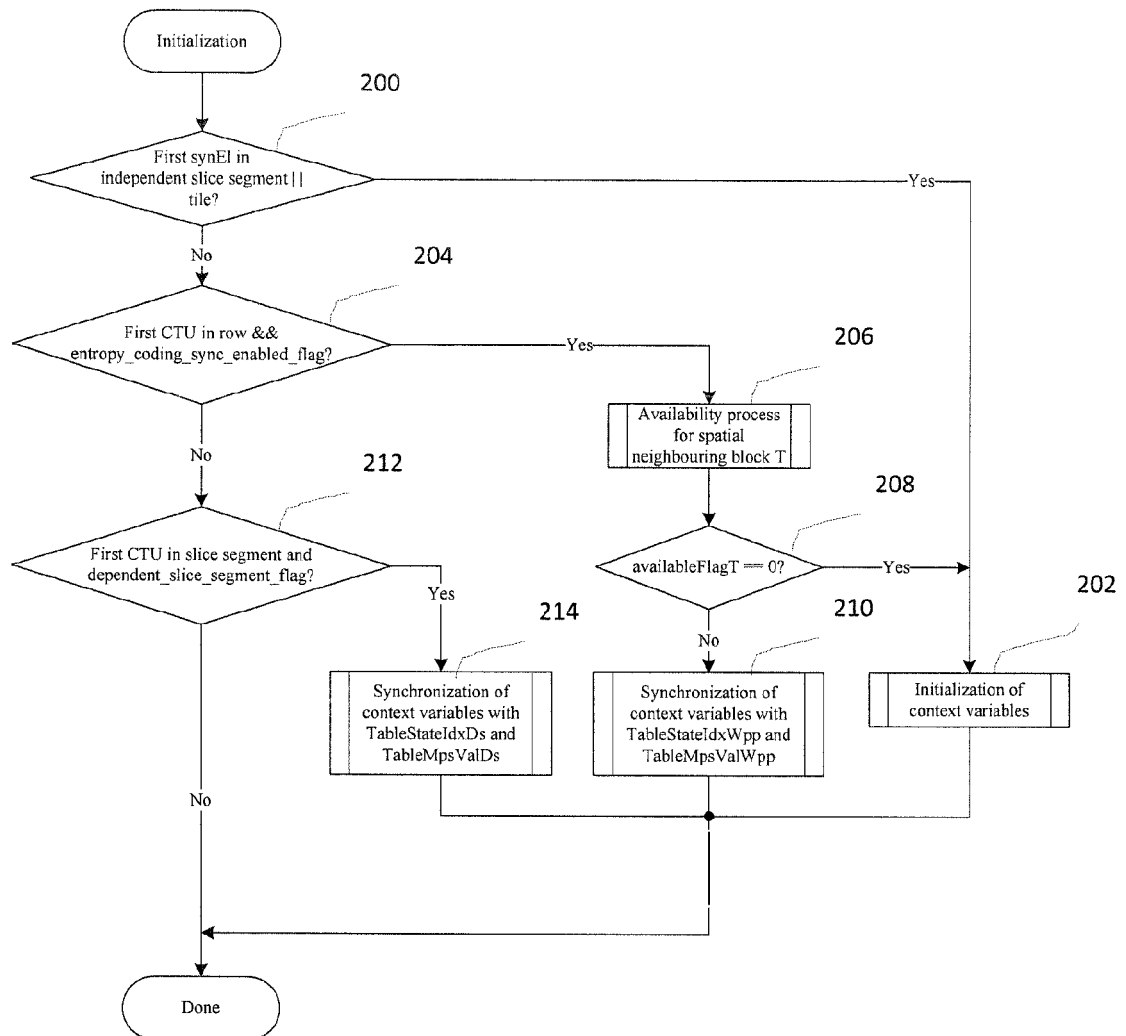


Fig. 29

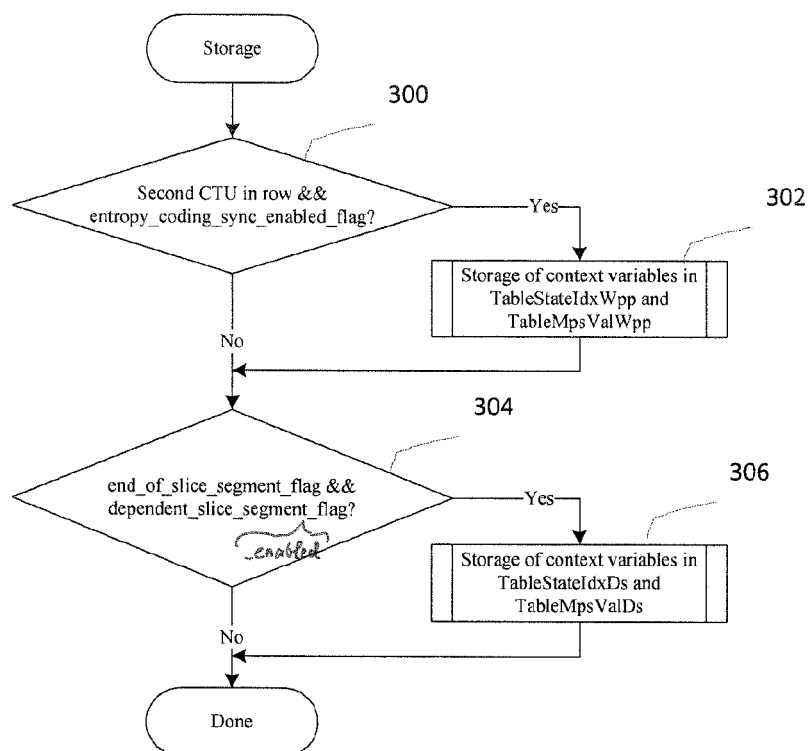


Fig. 30

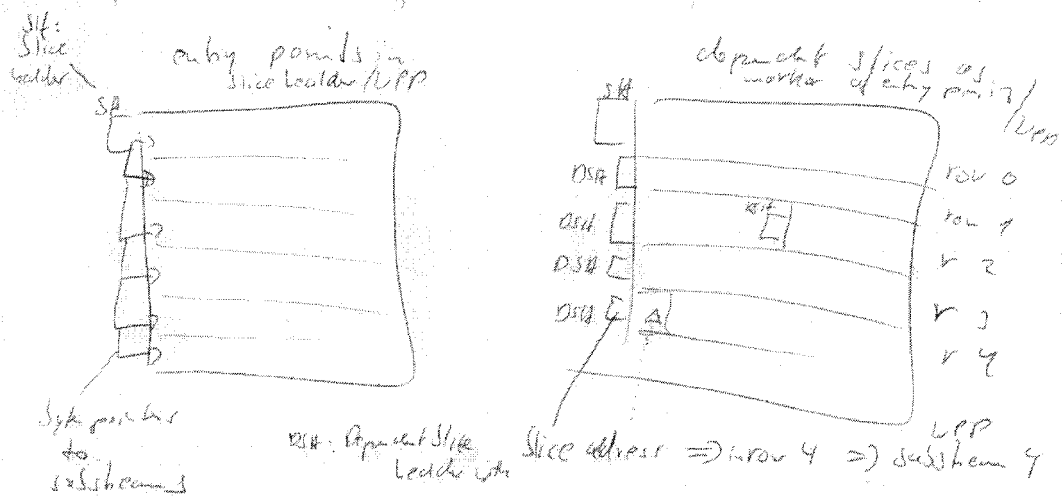


Fig. 31