

FIG. 1A

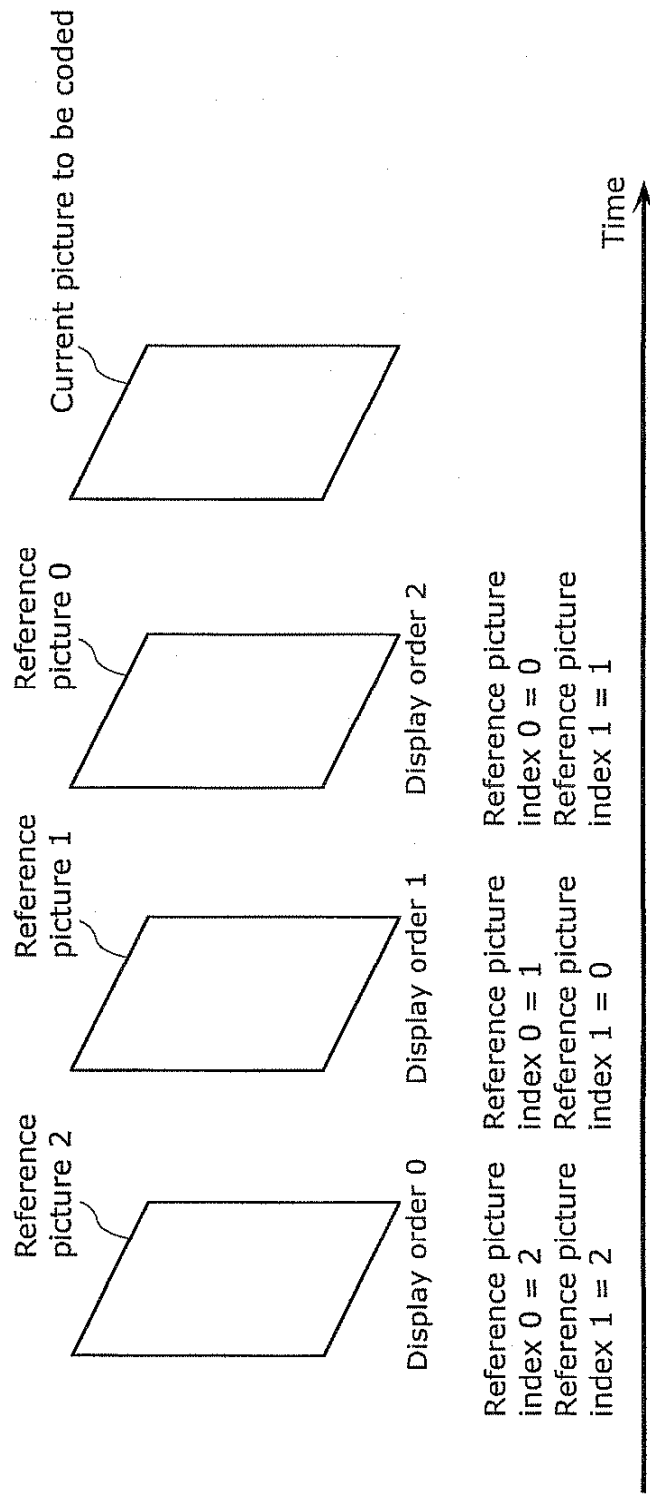


FIG. 1B

Reference picture list 0

Reference picture index 0	Display order
0	2
1	1
2	0

FIG. 1C

Reference picture list 1

Reference picture index 1	Display order
0	1
1	2
2	0

FIG. 2

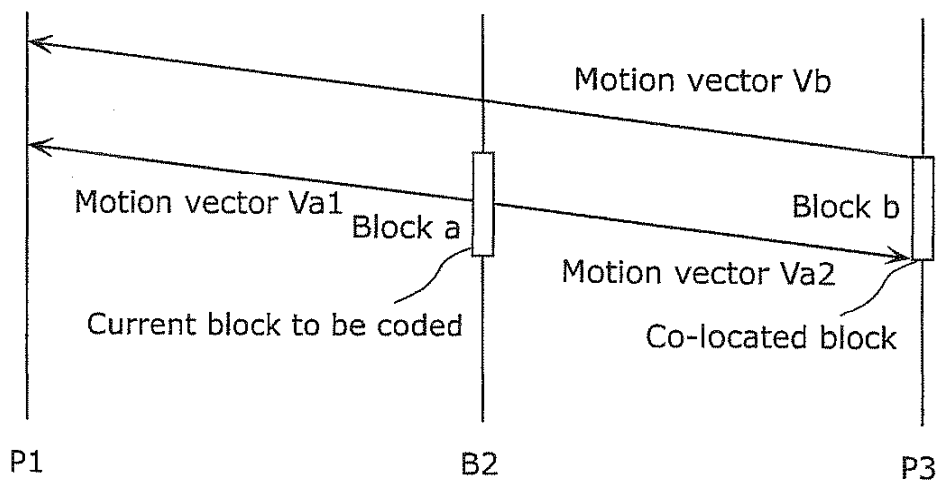


FIG. 3

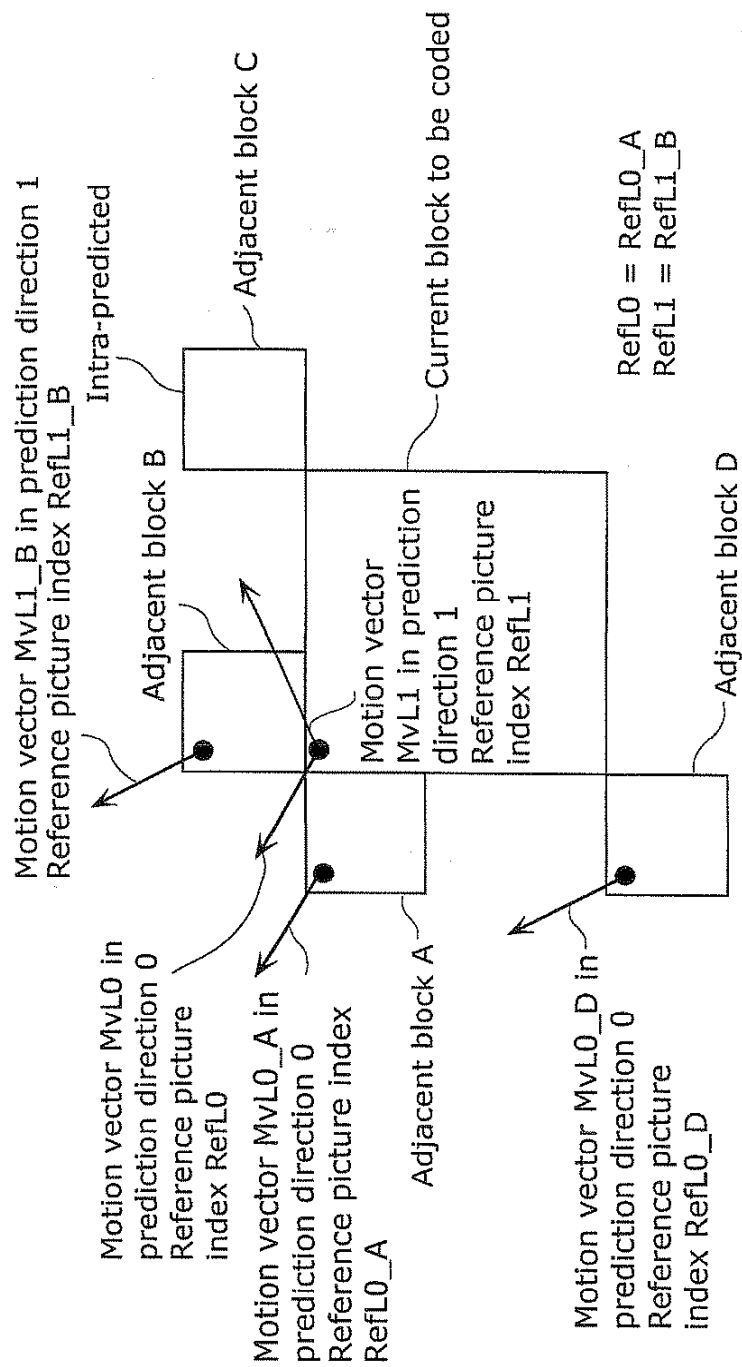


FIG. 4

(a)

Motion vector predictor index for prediction direction 0	Motion vector predictor candidate in prediction direction 0
0	Adjacent block A (MvL0_A, RefL0)
1	Adjacent block B (sMvL0_B, RefL0)
2	Co-located block (MvL0_Col, RefL0)
3	Non-available prediction block (since adjacent block C is intra-predicted)
4	Adjacent block D (sMvL0_D, RefL0)

When MvL0_A = sMvL0_D



Delete redundant candidate and non-available predictor candidate

(b)

Motion vector predictor index for prediction direction 0	Motion vector predictor candidate in prediction direction 0
0	Adjacent block A (MvL0_A, RefL0)
1	Adjacent block B (sMvL0_B, RefL0)
2	Co-located block (MvL0_Col, RefL0)

The number of motion vector predictor candidates in prediction direction 0 = 3

FIG. 5

(a)

Motion vector predictor index for prediction direction 1	Motion vector predictor candidate in prediction direction 1
0	Adjacent block A (sMvL1_A, RefL1)
1	Adjacent block B (MvL1_B, RefL1)
2	Co-located block (MvL1_Col, RefL1)
3	Non-available prediction block (since adjacent block C is intra-predicted)
4	Adjacent block D (sMvL1_D, RefL1)

When $MvL1_B = MvL1_Col = sMvL1_D$

Delete redundant candidate and non-available predictor candidate

(b)

Motion vector predictor index for prediction direction 1	Motion vector predictor candidate in prediction direction 1
0	Adjacent block A (sMvL1_A, RefL1)
1	Adjacent block B (MvL1_B, RefL1)

The number of motion vector predictor candidates in prediction direction 1 = 2

FIG. 6

Motion vector predictor candidate list size = 2

Motion vector predictor index	Assigned bit string
0	0
1	1

Motion vector predictor candidate list size = 3

Motion vector predictor index	Assigned bit string
0	0
1	10
2	11

Motion vector predictor candidate list size = 4

Motion vector predictor index	Assigned bit string
0	0
1	10
2	110
3	111

Motion vector predictor candidate list size = 5

Motion vector predictor index	Assigned bit string
0	0
1	10
2	110
3	1110
4	1111

FIG. 7

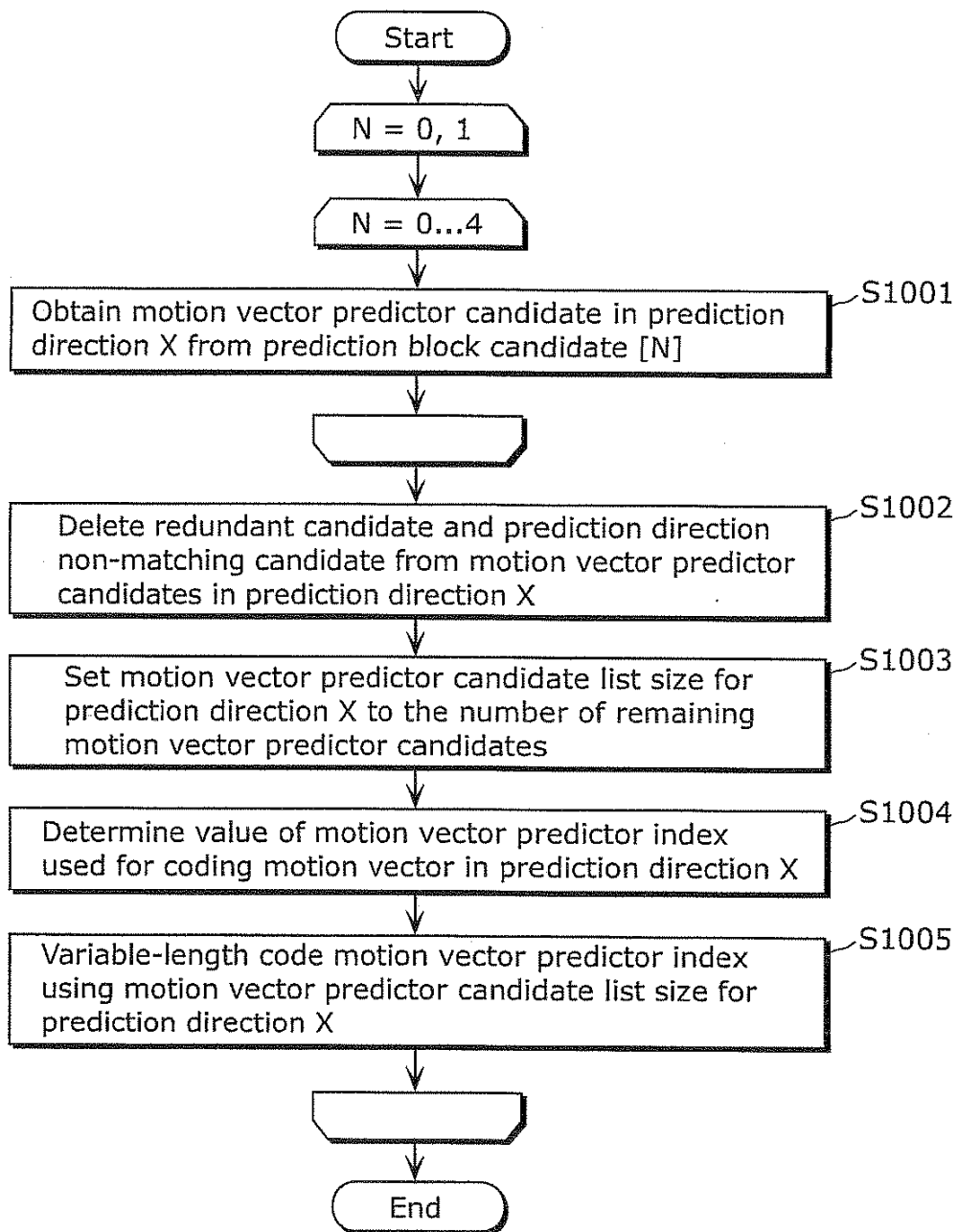


FIG. 8A

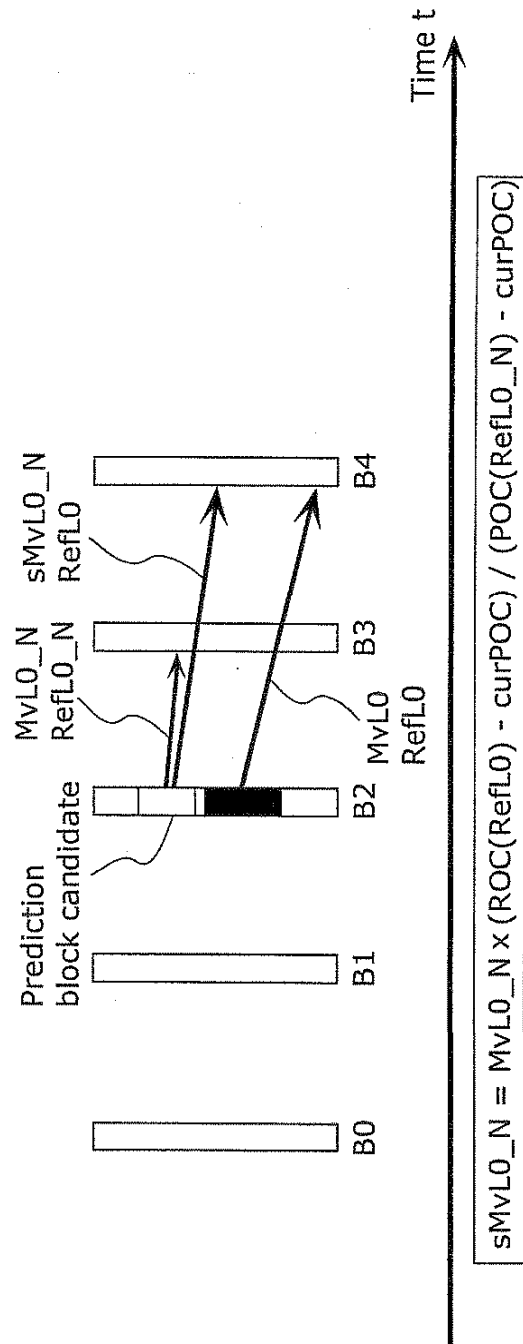


FIG. 8B

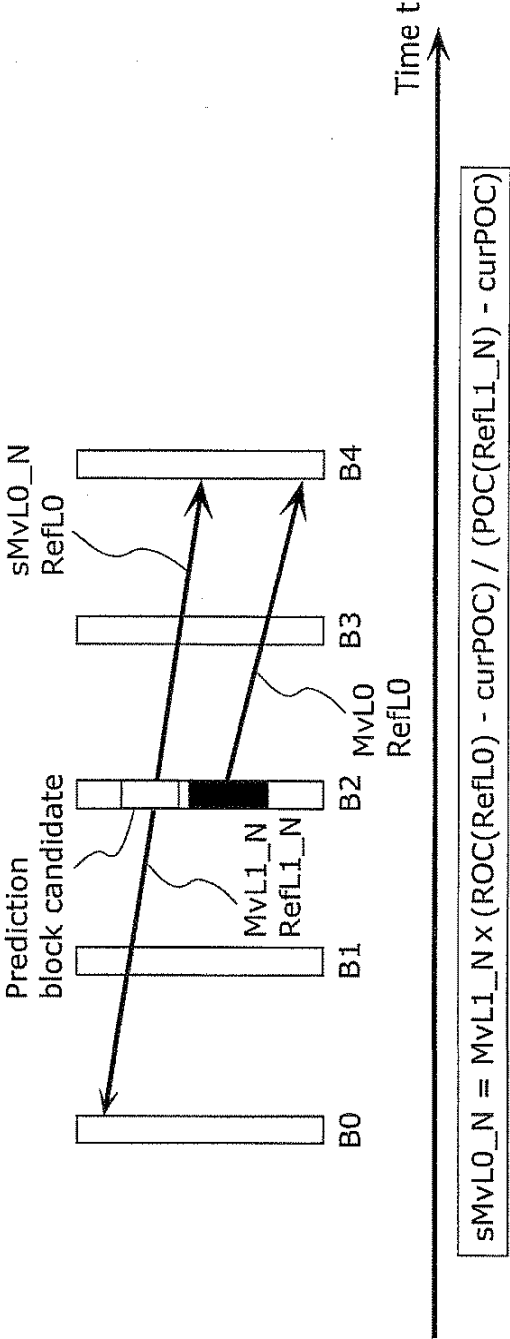


FIG. 9

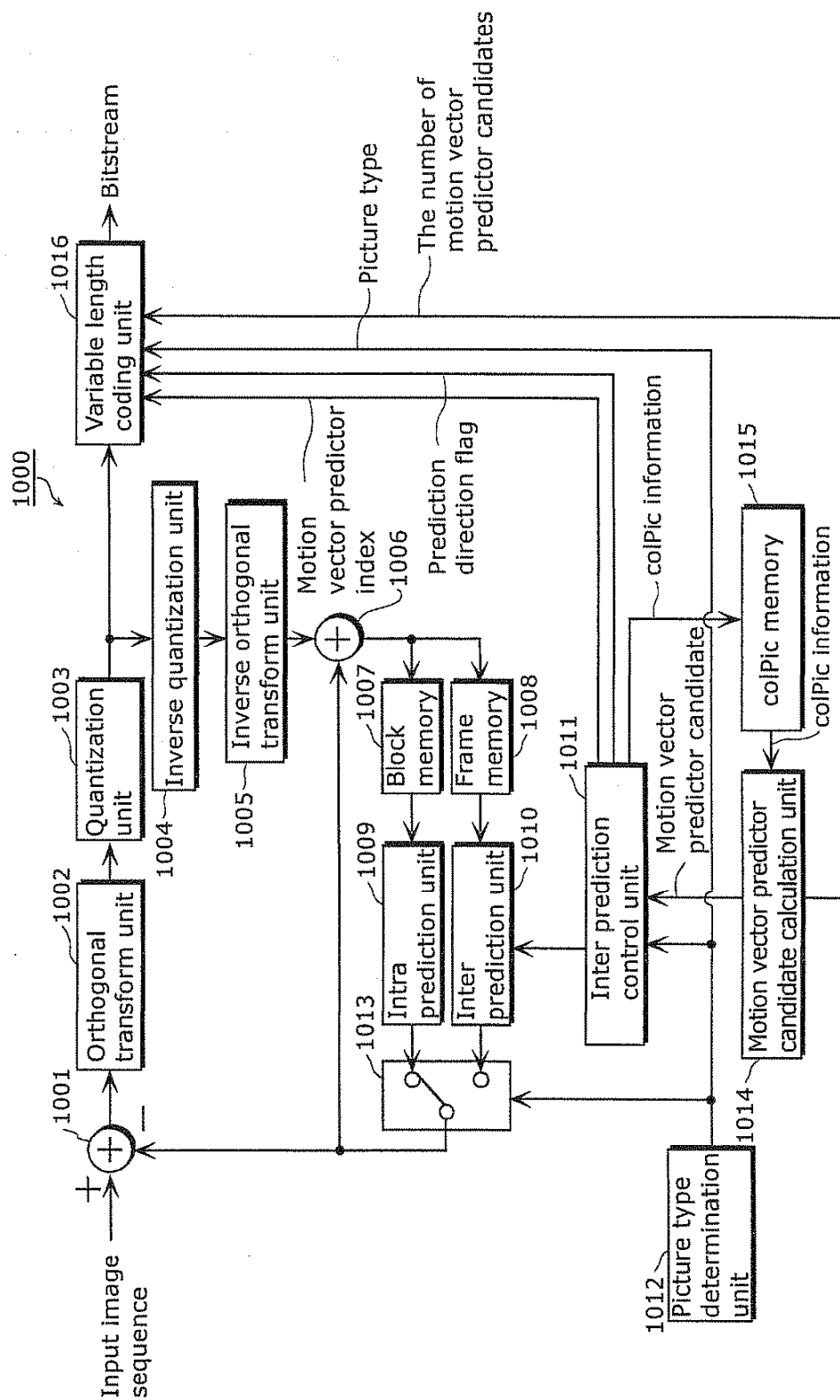


FIG. 10

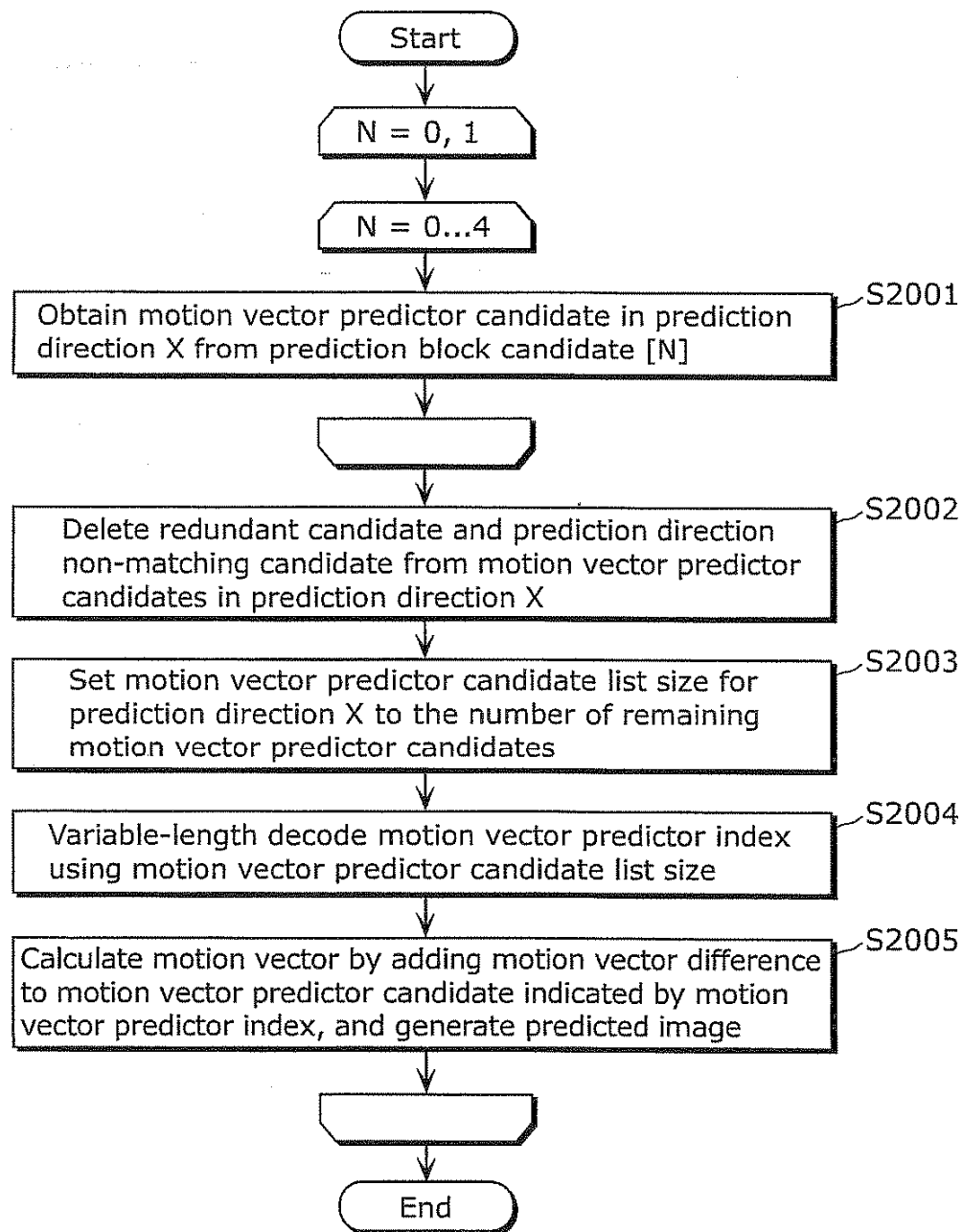


FIG. 11

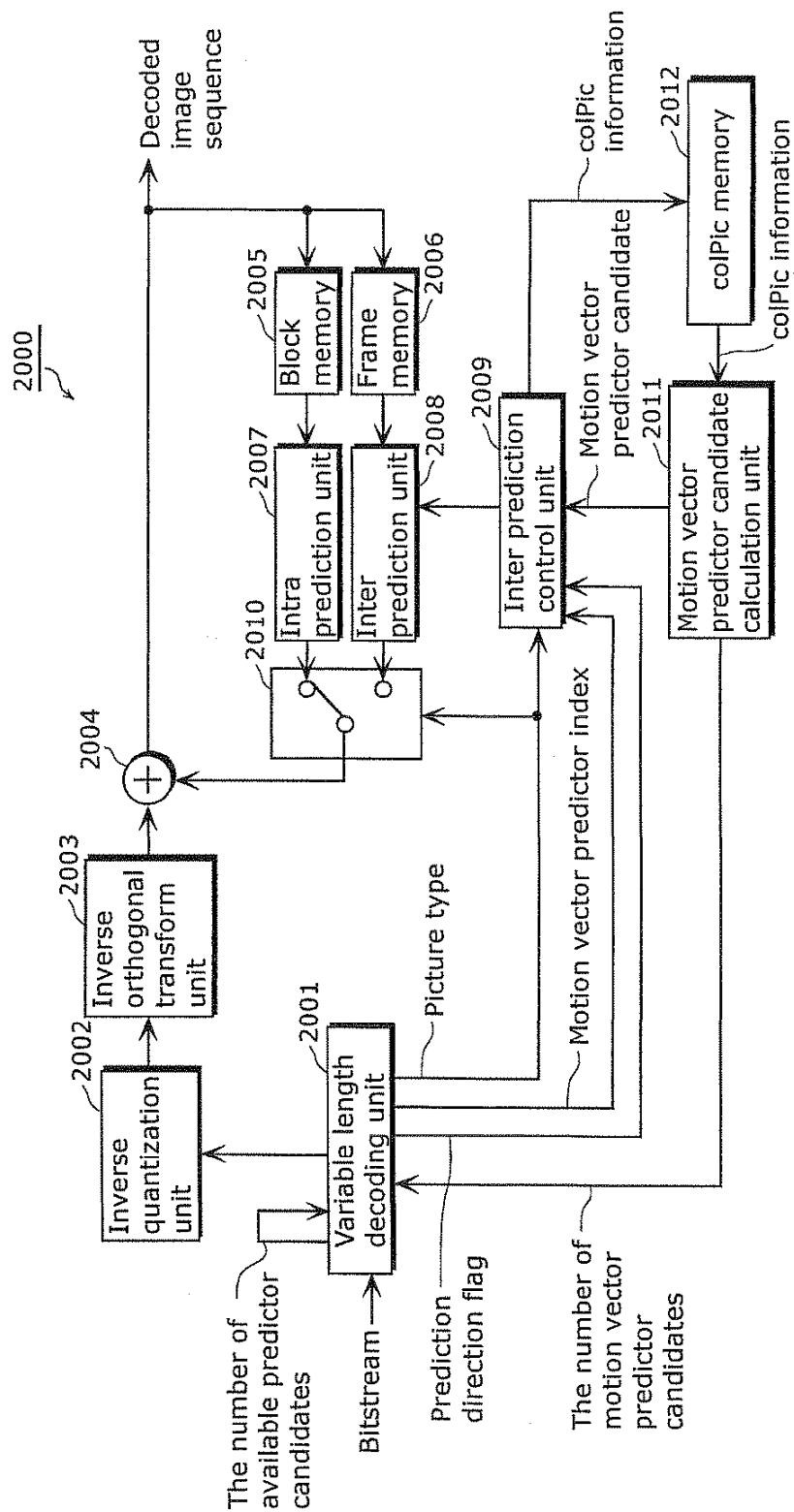


FIG. 12

	prediction_unit(x0, y0, log2PUWidth, log2PUHeight, PartIdx, InferredMergeFlag){	Descriptor
Prediction direction flag	if(skip_flag[x0][y0]){	
	...	
	} else if(PredMode == MODE_INTRA){	
	...	
Motion vector predictor candidate list size = the number of motion vector predictor candidates	} else { /* MODE_INTER */	
	if(!InferredMergeFlag)	
	merge_flag[x0][y0]	u(l) ae(v)
	if(merge_flag[x0][y0] && NumMergeCand > 1){	
	...	
	} else {	
	if(slice_type == B)	
	inter_pred_flag[x0][y0]	ue(v) ae(v)
Motion vector predictor index	if(inter_pred_flag[x0][y0] == Pred_LC){	
	...	
	if(NumMVPCand(LcToLx) > 1)	
Motion vector predictor candidate list size = the number of motion vector predictor candidates	mvp_idx_lc[x0][y0]	ue(v) ae(v)
	}	
	else { /* Pred_L0 or Pred_BI */	
	if(num_ref_idx_l0_active_minus1 > 0)	
	...	
	if(NumMVPCand(L0) > 1)	
	mvp_idx_l0[x0][y0]	ue(v) ae(v)
	}	
Motion vector predictor index	if(inter_pred_flag[x0][y0] == Pred_BI){	
Motion vector predictor candidate list size = the number of motion vector predictor candidates	...	
	if(NumMVPCand(L1) > 1)	
	mvp_idx_l1[x0][y0]	ue(v) ae(v)
	}	
	}	
	}	
	}	
Motion vector predictor index	}	

FIG. 13

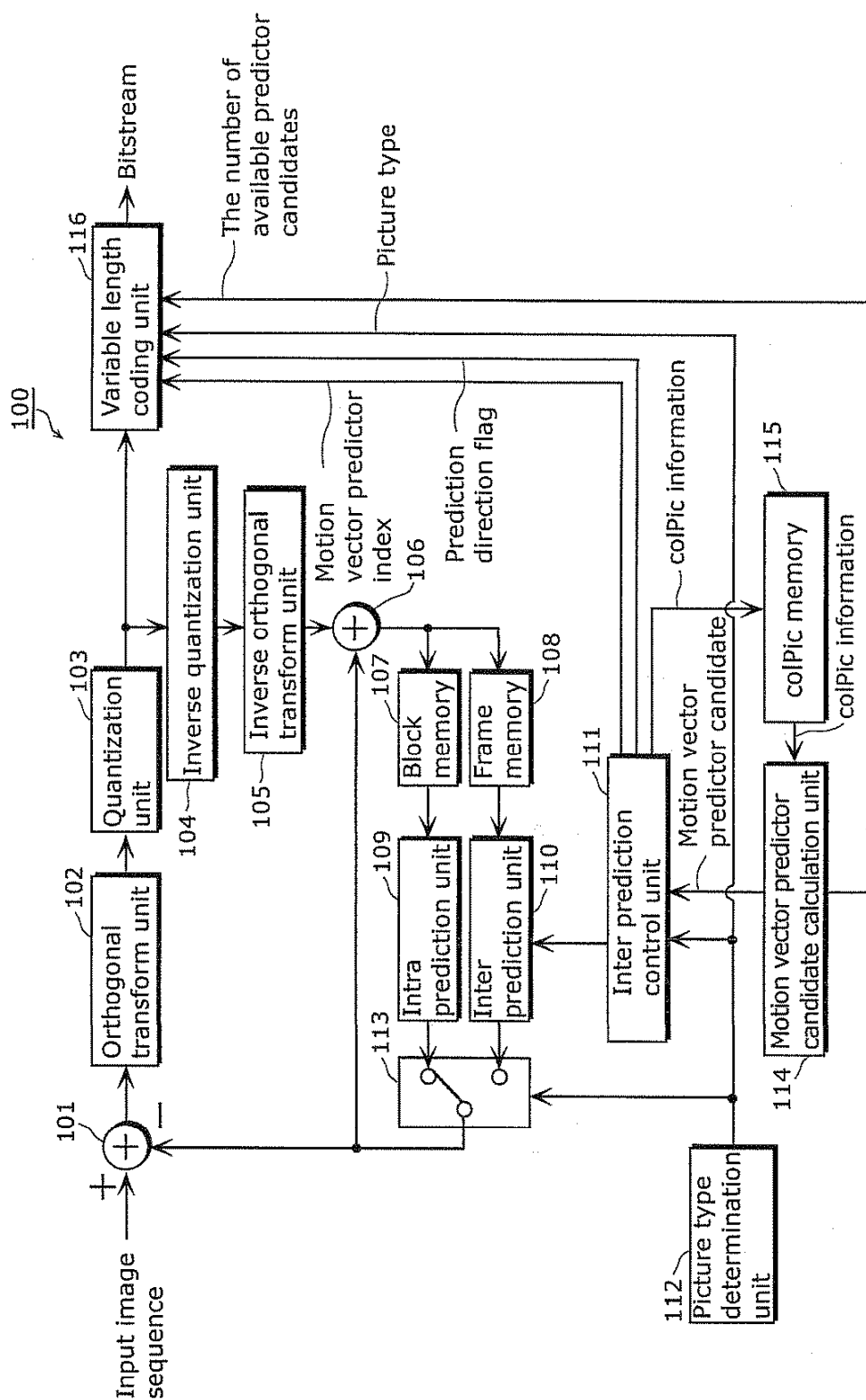


FIG. 14

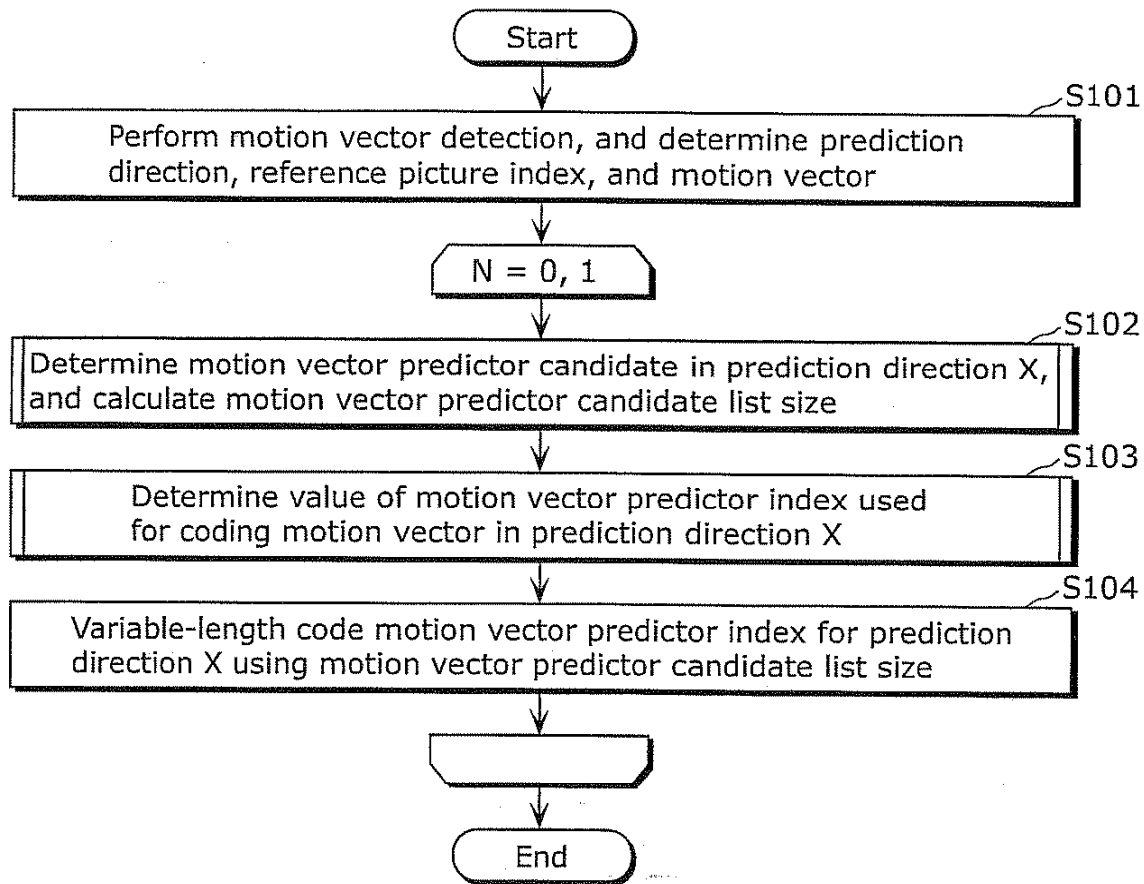


FIG. 15A

(a)

Motion vector predictor index for prediction direction 0	Motion vector predictor candidate in prediction direction 0
0	Adjacent block A (MvL0_A, RefL0)
1	Adjacent block B (sMvL0_B, RefL0)
2	Co-located block (MvL0_Col, RefL0)
3	Non-available prediction block (since adjacent block C is intra-predicted)
4	Adjacent block D (sMvL0_D, RefL0)

When $MvL0_A = sMvL0_D$

-Delete non-available predictor candidate and redundant candidate
 -Add new candidate and add 1 to the number of motion vector predictor candidates if (the number of available predictor candidates - the number of motion vector predictor candidates) > 0

(b)

Motion vector predictor index for prediction direction 0	Motion vector predictor candidate in prediction direction 0
0	Adjacent block A (MvL0_A, RefL0)
1	Adjacent block B (sMvL0_B, RefL0)
2	Co-located block (MvL0_Col, RefL0)
3	New candidate

The number of available predictor candidates in prediction direction 0 = 4

FIG. 15B

(a)

Motion vector predictor index for prediction direction 0	Motion vector predictor candidate in prediction direction 0
0	Adjacent block A (MvL0_A, RefL0)
1	Adjacent block B (sMvL0_B, RefL0)
2	Co-located block (MvL0_Col, RefL0)
3	Non-available prediction block (since adjacent block C is intra-predicted)
4	Adjacent block D (sMvL0_D, RefL0)

When MvL0_A = sMvL0_D

-Delete non-available predictor candidate and redundant candidate
 -Add new candidate and add 1 to the number of motion vector predictor candidates if (the number of available predictor candidates - the number of motion vector predictor candidates) > 0

(b)

Motion vector predictor index for prediction direction 0	Motion vector predictor candidate in prediction direction 0
0	Adjacent block A (MvL0_A, RefL0)
1	Adjacent block B (sMvL0_B, RefL0)
2	Co-located block (MvL0_Col, RefL0)
3	New candidate
4	Second candidate

The number of available predictor candidates in prediction direction 0 = 4

FIG. 15C

(a)

Motion vector predictor index for prediction direction 0	Motion vector predictor candidate in prediction direction 0
0	Second candidate
1	Second candidate
2	Second candidate
3	Second candidate
4	Second candidate

When $MvL0_A = sMvL0_D$ ↓ -Add candidate other than non-available predictor candidate and
redundant candidate to list

(b)

Motion vector predictor index for prediction direction 0	Motion vector predictor candidate in prediction direction 0
0	Adjacent block A ($MvL0_A$, RefL0)
1	Adjacent block B ($sMvL0_B$, RefL0)
2	Co-located block ($MvL0_Col$, RefL0)
3	Second candidate
4	Second candidate

↓ -Add new candidate and add 1 to the number of motion vector
predictor candidates if (candidate list size - the number of motion
vector predictor candidates) > 0

(c)

Motion vector predictor index for prediction direction 0	Motion vector predictor candidate in prediction direction 0
0	Adjacent block A ($MvL0_A$, RefL0)
1	Adjacent block B ($sMvL0_B$, RefL0)
2	Co-located block ($MvL0_Col$, RefL0)
3	New candidate
4	Second candidate

FIG. 16A

(a)

Motion vector predictor index for prediction direction 1	Motion vector predictor candidate in prediction direction 1
0	Adjacent block A (MvL1_A, RefL1)
1	Adjacent block B (sMvL1_B, RefL1)
2	Co-located block (MvL1_Col, RefL1)
3	Non-available prediction block (since adjacent block C is intra-predicted)
4	Adjacent block D (sMvL1_D, RefL1)

When $MvL1_B = MvL1_Col = sMvL1_D$

- Delete non-available predictor candidate and redundant candidate
- Add new candidate and add 1 to the number of motion vector predictor candidates if (the number of available predictor candidates - the number of motion vector predictor candidates) > 0

(b)

Motion vector predictor index for prediction direction 1	Motion vector predictor candidate in prediction direction 1
0	Adjacent block A (sMvL1_A, RefL1)
1	Adjacent block B (MvL1_B, RefL1)
2	New candidate 1
3	New candidate 2

The number of available predictor candidates in prediction direction 1 = 4

FIG. 16B

(a)

Motion vector predictor index for prediction direction 1	Motion vector predictor candidate in prediction direction 1
0	Adjacent block A (MvL1_A, RefL1)
1	Adjacent block B (sMvL1_B, RefL1)
2	Co-located block (MvL1_Col, RefL1)
3	Non-available prediction block (since adjacent block C is intra-predicted)
4	Adjacent block D (sMvL1_D, RefL1)

When $MvL1_B = MvL1_Col = sMvL1_D$

- Delete non-available predictor candidate and redundant candidate
- Add new candidate and add 1 to the number of motion vector predictor candidates if (the number of available predictor candidates - the number of motion vector predictor candidates) > 0

(b)

Motion vector predictor index for prediction direction 1	Motion vector predictor candidate in prediction direction 1
0	Adjacent block A (sMvL1_A, RefL1)
1	Adjacent block B (MvL1_B, RefL1)
2	New candidate 1
3	New candidate 2
3	Second candidate

The number of available predictor candidates in prediction direction 1 = 4

FIG. 16C

(a)

Motion vector predictor index for prediction direction 1	Motion vector predictor candidate in prediction direction 1
0	Second candidate
1	Second candidate
2	Second candidate
3	Second candidate
4	Second candidate

When $MvL1_B = MvL1_Col = sMvL1_D$ ↓ -Add candidate other than non-available predictor candidate and redundant candidate to list

(b)

Motion vector predictor index for prediction direction 1	Motion vector predictor candidate in prediction direction 1
0	Adjacent block A ($sMvL1_A$, $RefL1$)
1	Adjacent block B ($MvL1_B$, $RefL1$)
2	Second candidate
3	Second candidate
4	Second candidate

↓ -Add new candidate and add 1 to the number of motion vector predictor candidates if (candidate list size - the number of motion vector predictor candidates) > 0

(c)

Motion vector predictor index for prediction direction 1	Motion vector predictor candidate in prediction direction 1
0	Adjacent block A ($sMvL1_A$, $RefL1$)
1	Adjacent block B ($MvL1_B$, $RefL1$)
2	New candidate 1
3	New candidate 2
4	Second candidate

FIG. 17A

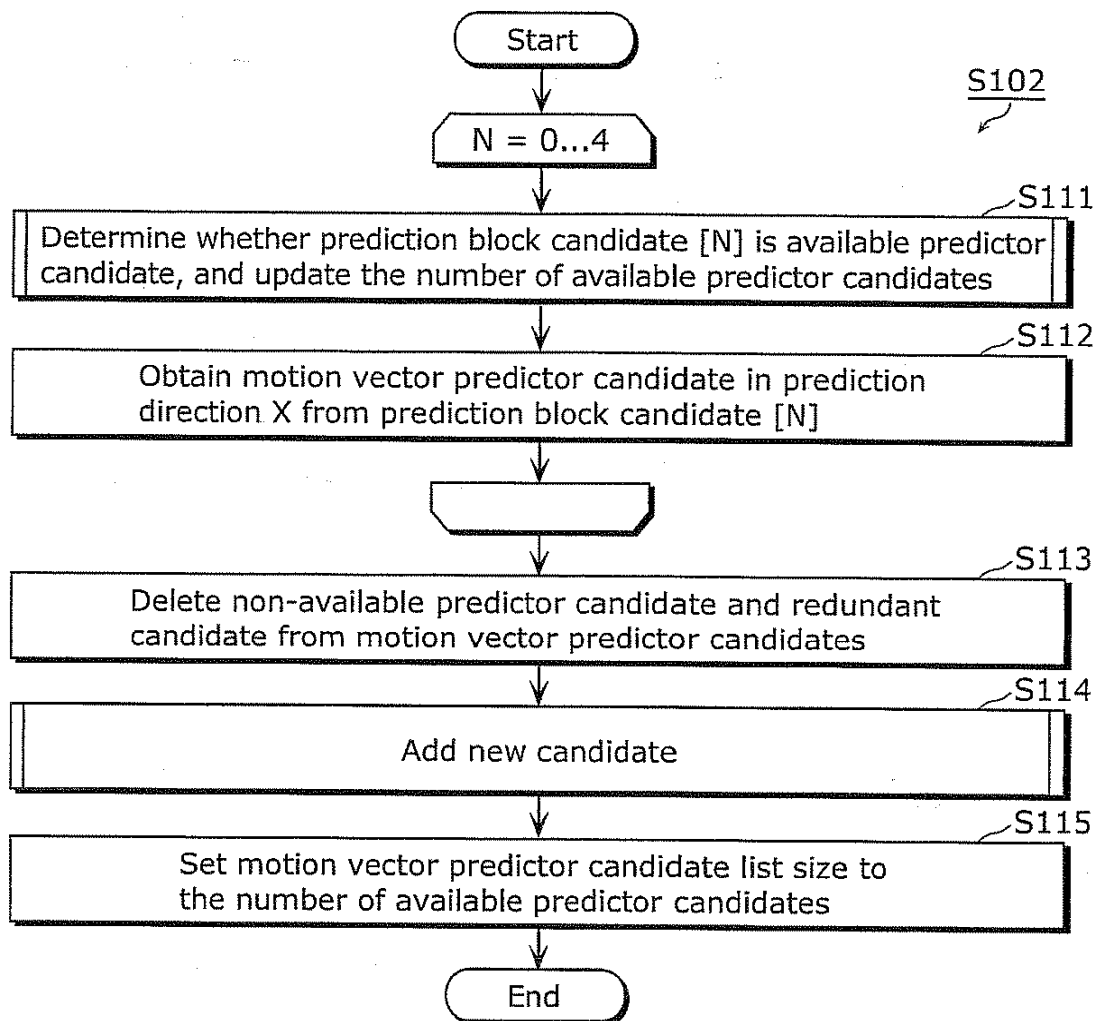


FIG. 17B

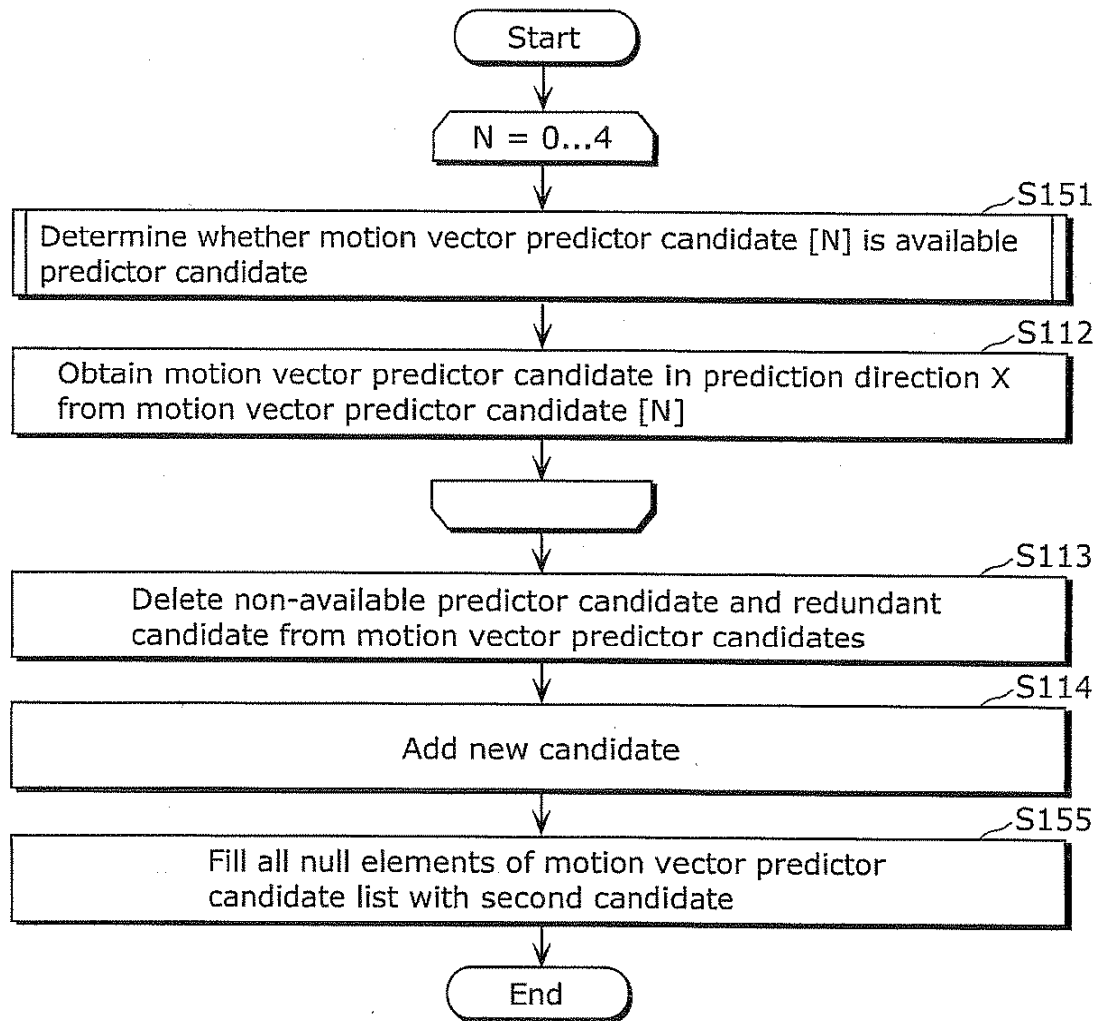


FIG. 17C

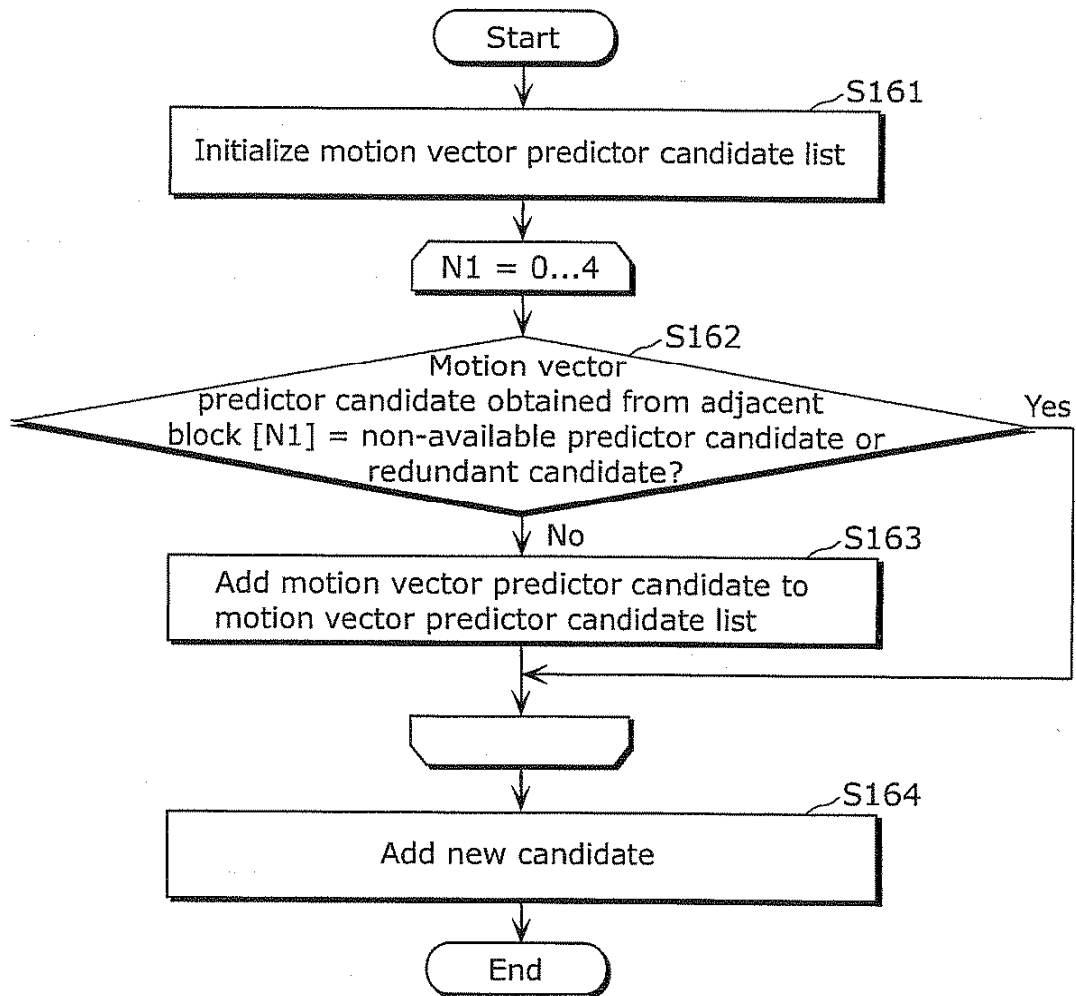


FIG. 18

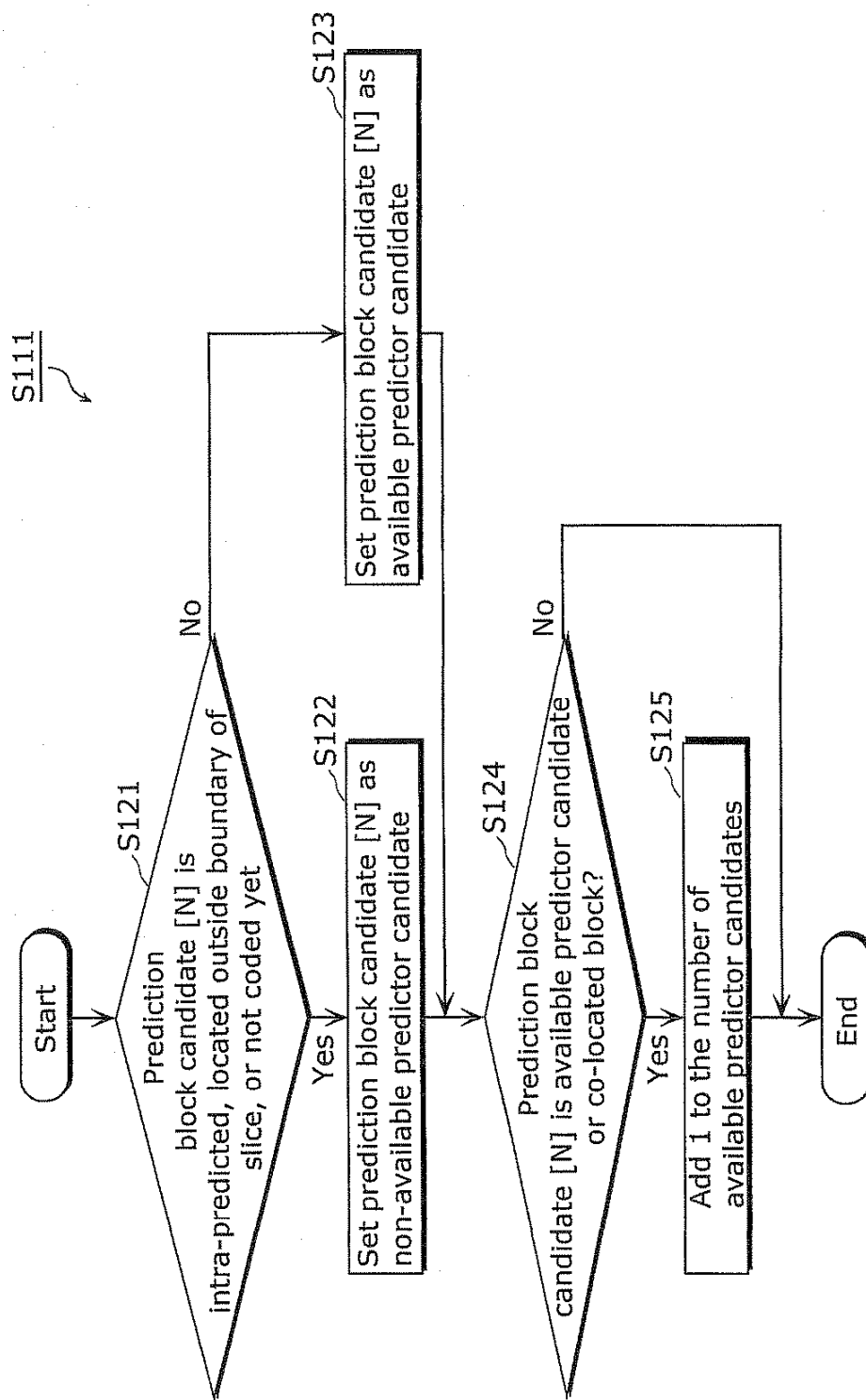


FIG. 19A

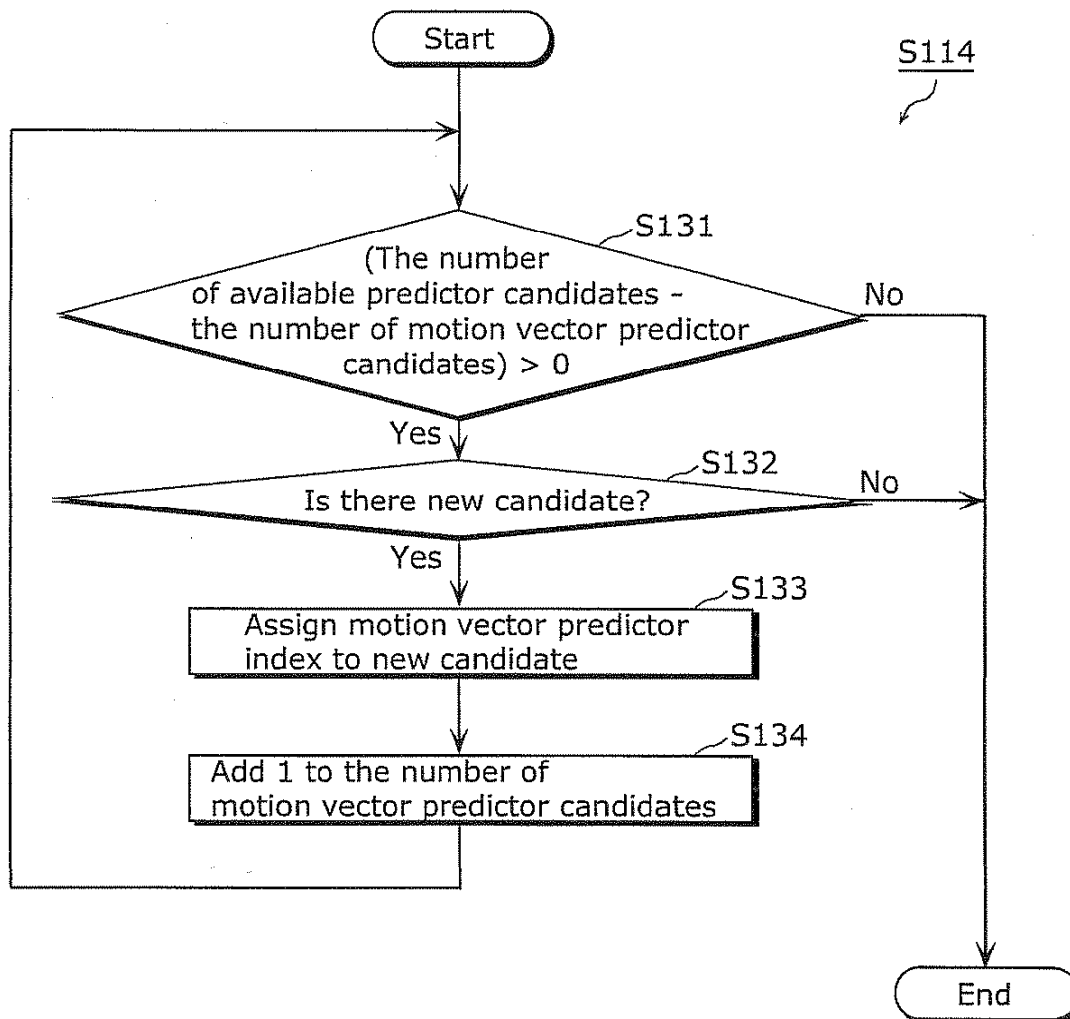


FIG. 19B

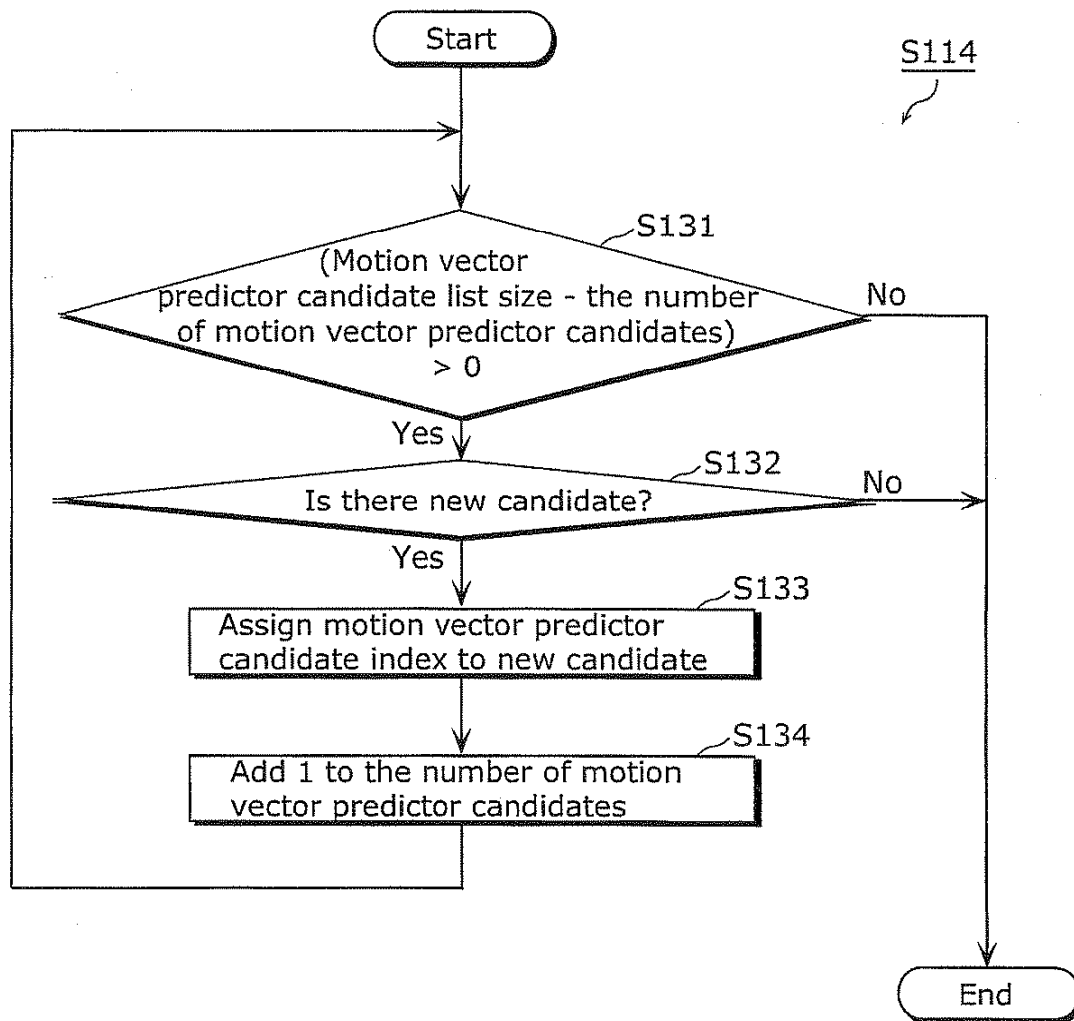


FIG. 20

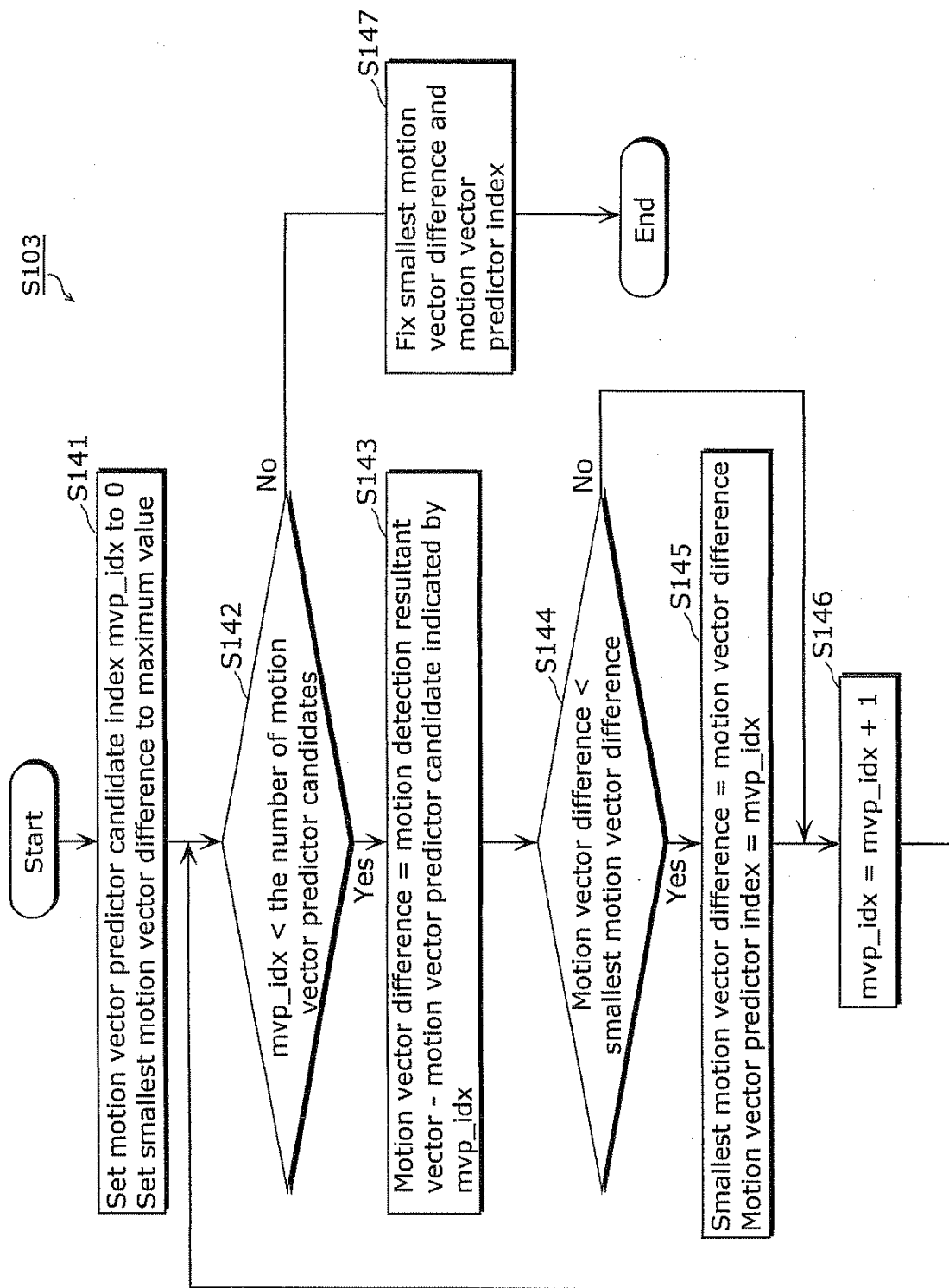


FIG. 21

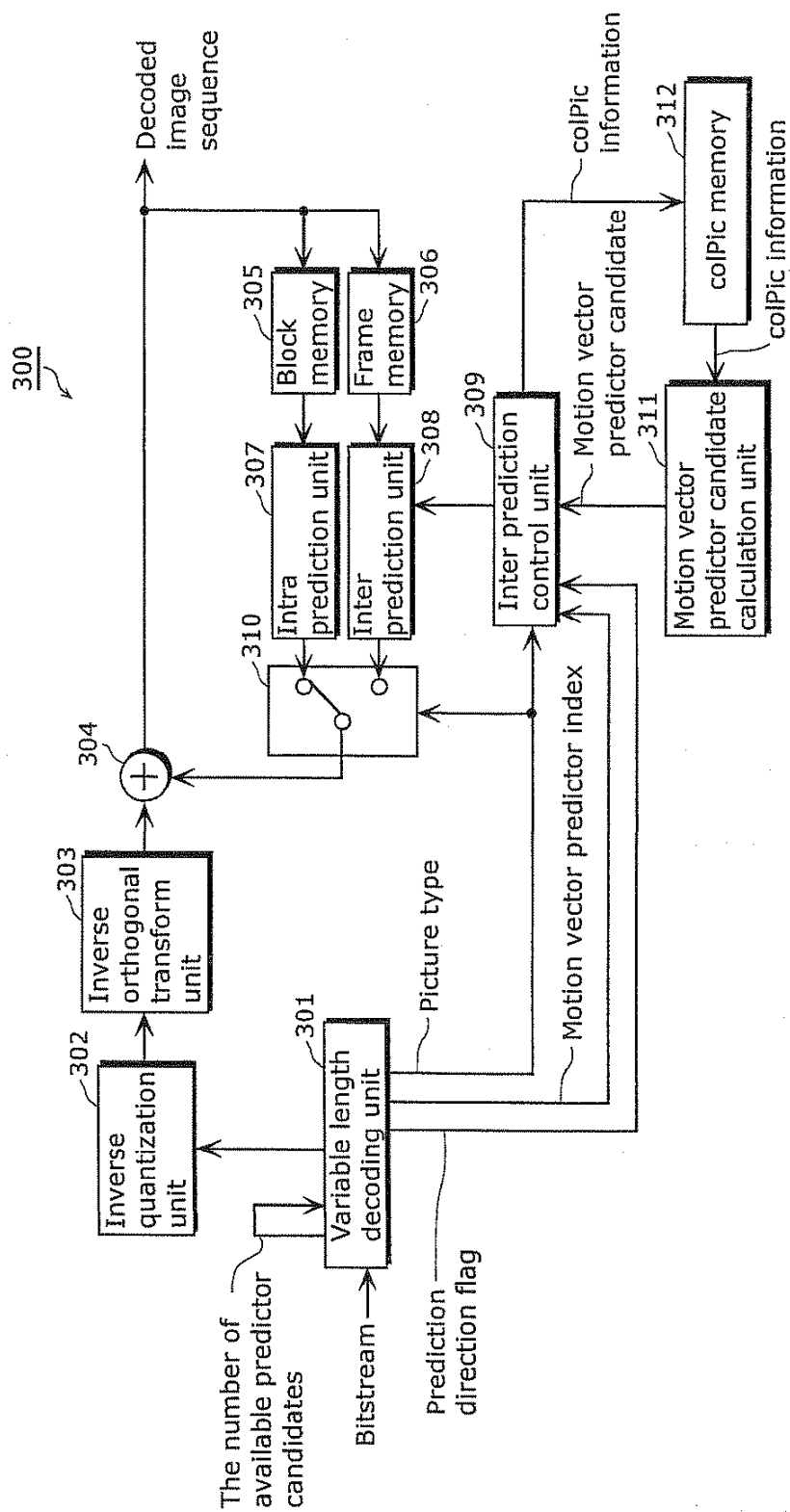


FIG. 22

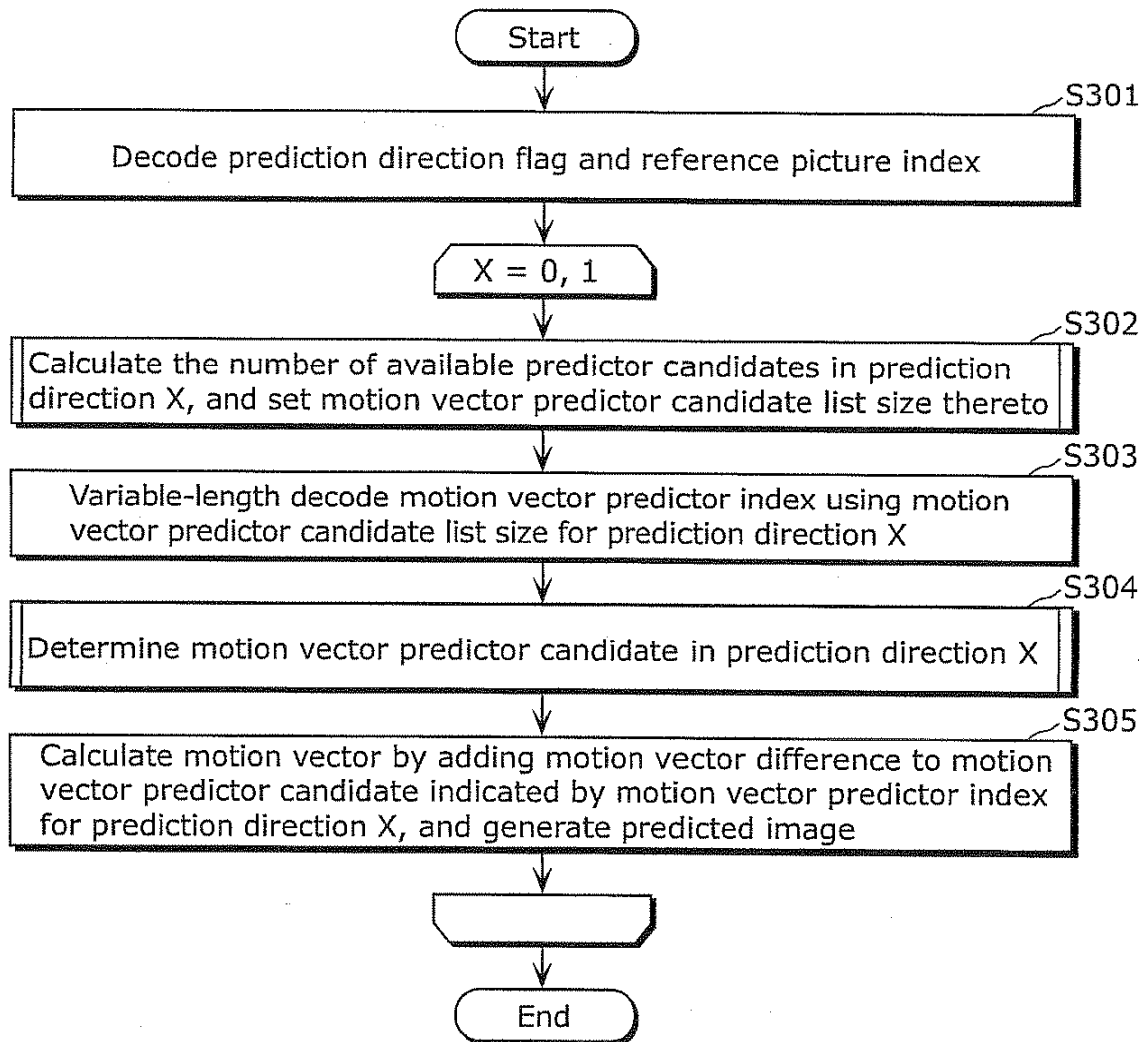


FIG. 23

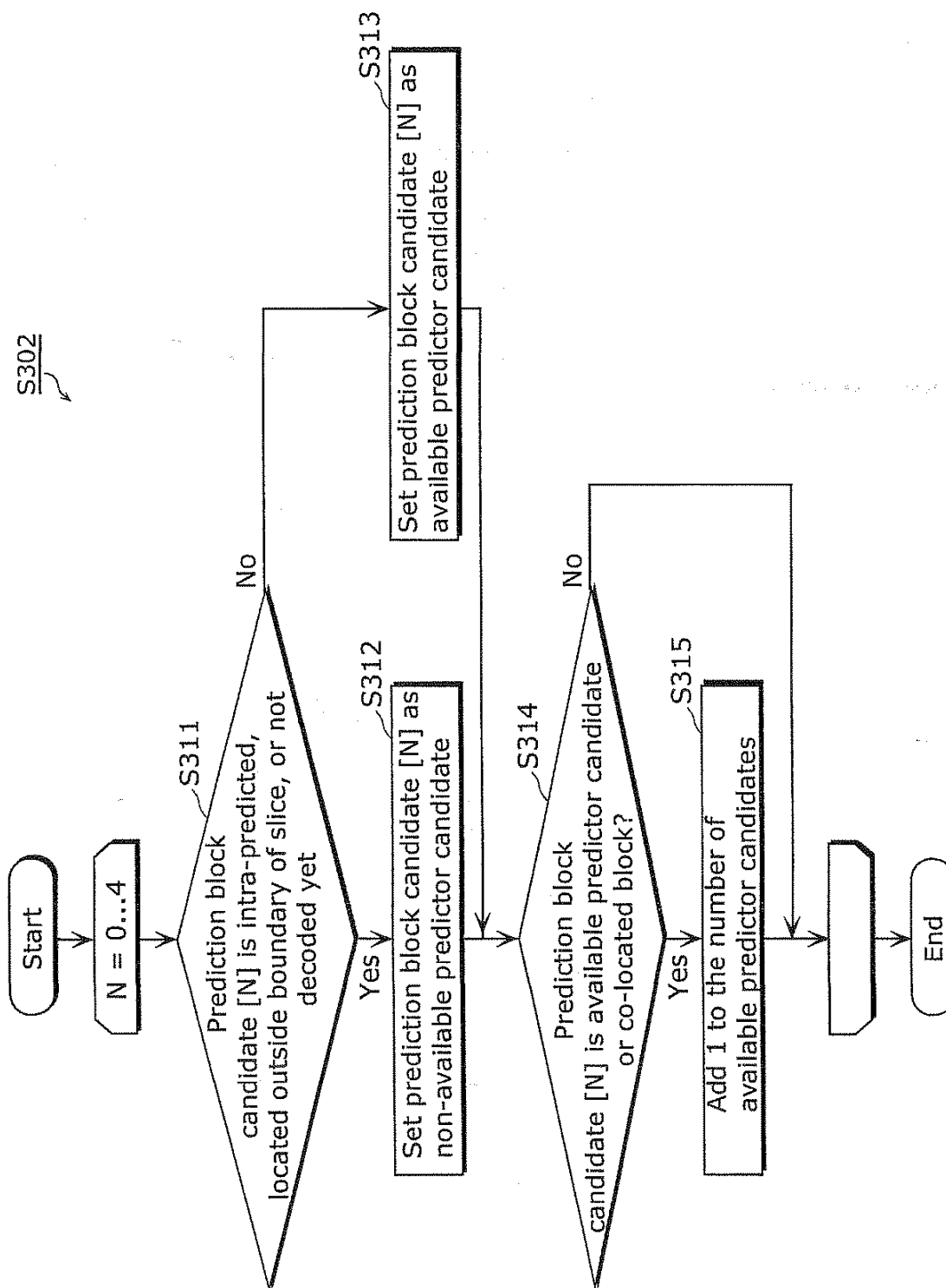


FIG. 24

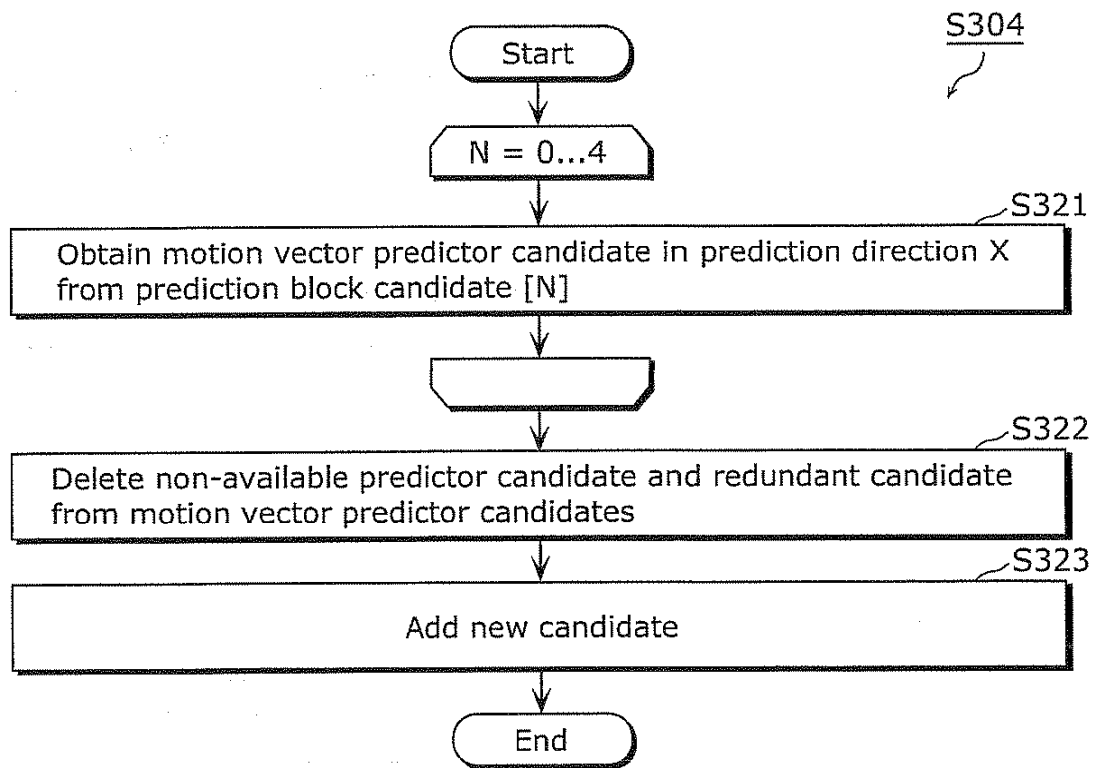


FIG. 25

	prediction_unit(x0, y0, log2PUWidth, log2PUHeight, PartIdx, InferredMergeFlag){	Descriptor
	if(skip_flag[x0][y0]){	
	...	
	} else if(PredMode == MODE_INTRA){	
	...	
Prediction direction flag	} else { /* MODE_INTER */	
	if(!InferredMergeFlag)	
Motion vector predictor candidate list size = the number of available predictor candidates	merge_flag[x0][y0]	u(l) ae(v)
	if(merge_flag[x0][y0] && NumMergeCand > 1){	
	...	
	} else {	
	if(slice_type == B)	
	inter_pred_flag[x0][y0]	ue(v) ae(v)
	if(inter_pred_flag[x0][y0] == Pred_LC){	
	...	
	if(NumMVPCand(LcToLx) > 1)	
Motion vector predictor index	mvp_idx_lc[x0][y0]	ue(v) ae(v)
	}	
	else { /* Pred_L0 or Pred_BI */	
Motion vector predictor candidate list size = the number of available predictor candidates	if(num_ref_idx_l0_active_minus1 > 0)	
	...	
	if(NumMVPCand(L0) > 1)	
	mvp_idx_l0[x0][y0]	ue(v) ae(v)
	}	
	if(inter_pred_flag[x0][y0] == Pred_BI){	
	...	
	if(NumMVPCand(L1) > 1)	
Motion vector predictor index	mvp_idx_l1[x0][y0]	ue(v) ae(v)
	}	
	}	
Motion vector predictor candidate list size = the number of available predictor candidates	}	
	}	
Motion vector predictor index	}	

FIG. 26

	prediction_unit(x0, y0, log2PUWidth, log2PUHeight, PartIdx, InferredMergeFlag){	Descriptor
	if(skip_flag[x0][y0]){	
	...	
	} else if(PredMode == MODE_INTRA){	
	...	
	} else { /* MODE_INTER */	
	if(!InferredMergeFlag)	
Prediction direction flag	merge_flag[x0][y0]	u(l) ae(v)
	if(merge_flag[x0][y0] && NumMergeCand > 1){	
	...	
	} else {	
	if(slice_type == B)	
	inter_pred_flag[x0][y0]	ue(v) ae(v)
	if(inter_pred_flag[x0][y0] == Pred_LC){	
	...	
Motion vector predictor index	mvp_idx_lc[x0][y0]	ue(v) ae(v)
	}	
	else { /* Pred_L0 or Pred_BI */	
	if(num_ref_idx_l0_active_minus1 > 0)	
	...	
	mvp_idx_l0[x0][y0]	ue(v) ae(v)
	}	
Motion vector predictor index	if(inter_pred_flag[x0][y0] == Pred_BI){	
	...	
	mvp_idx_l1[x0][y0]	ue(v) ae(v)
	}	
Motion vector predictor index	}	
	}	
	}	

FIG. 27

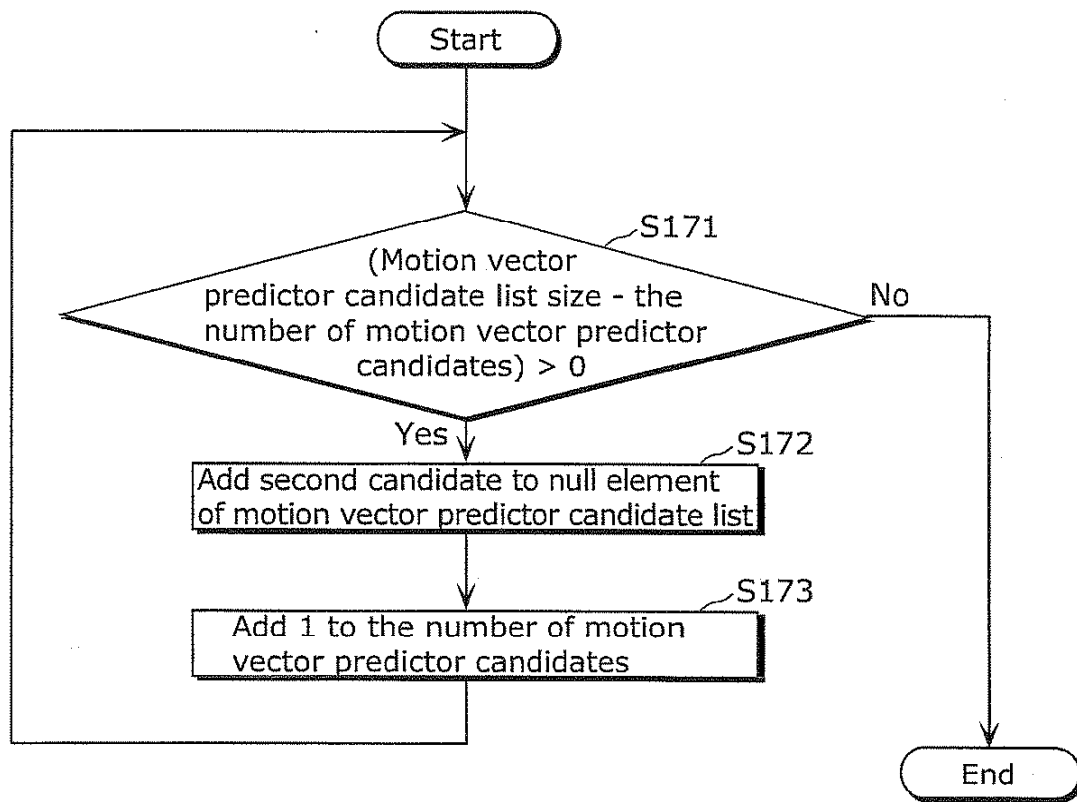


FIG. 28

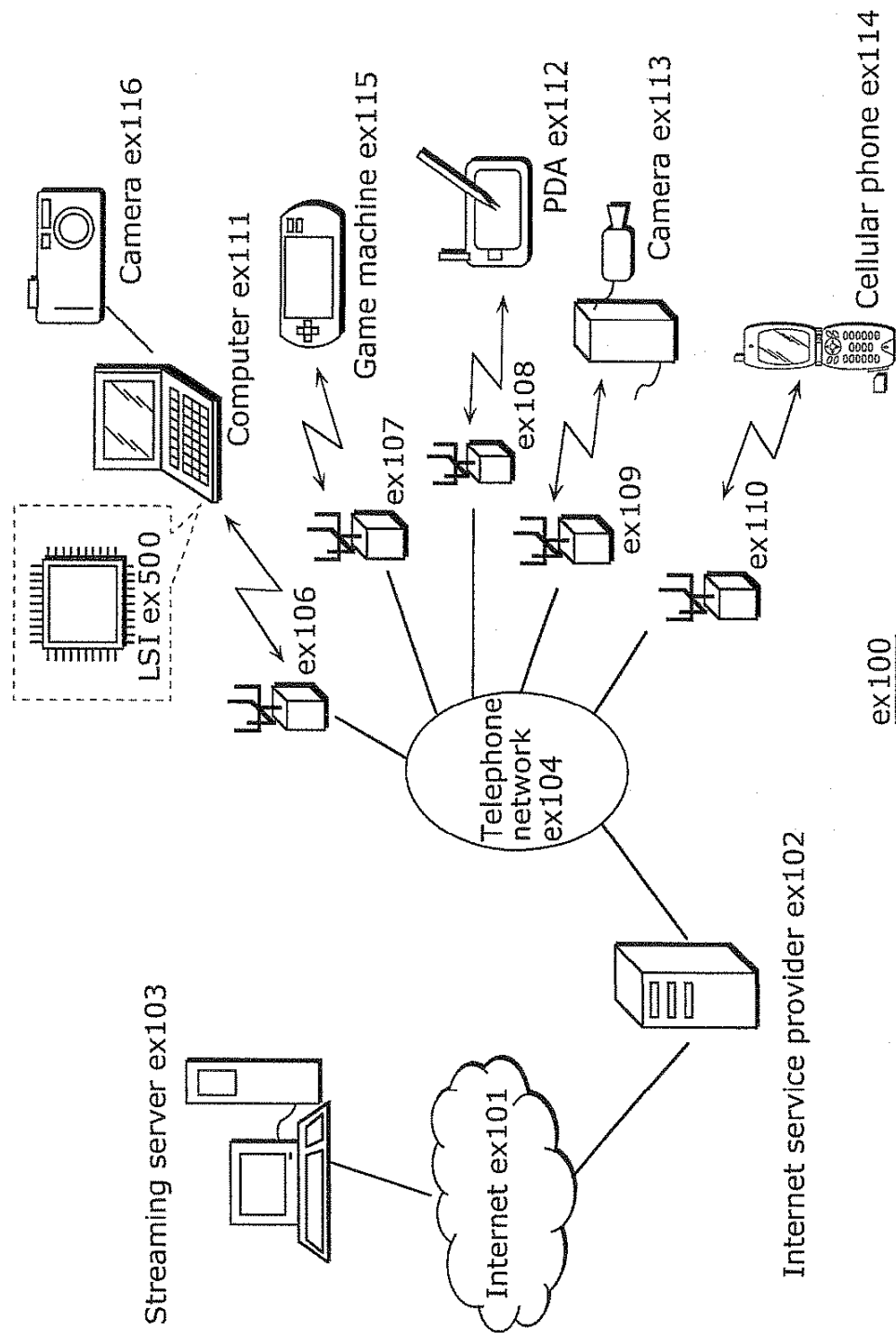


FIG. 30

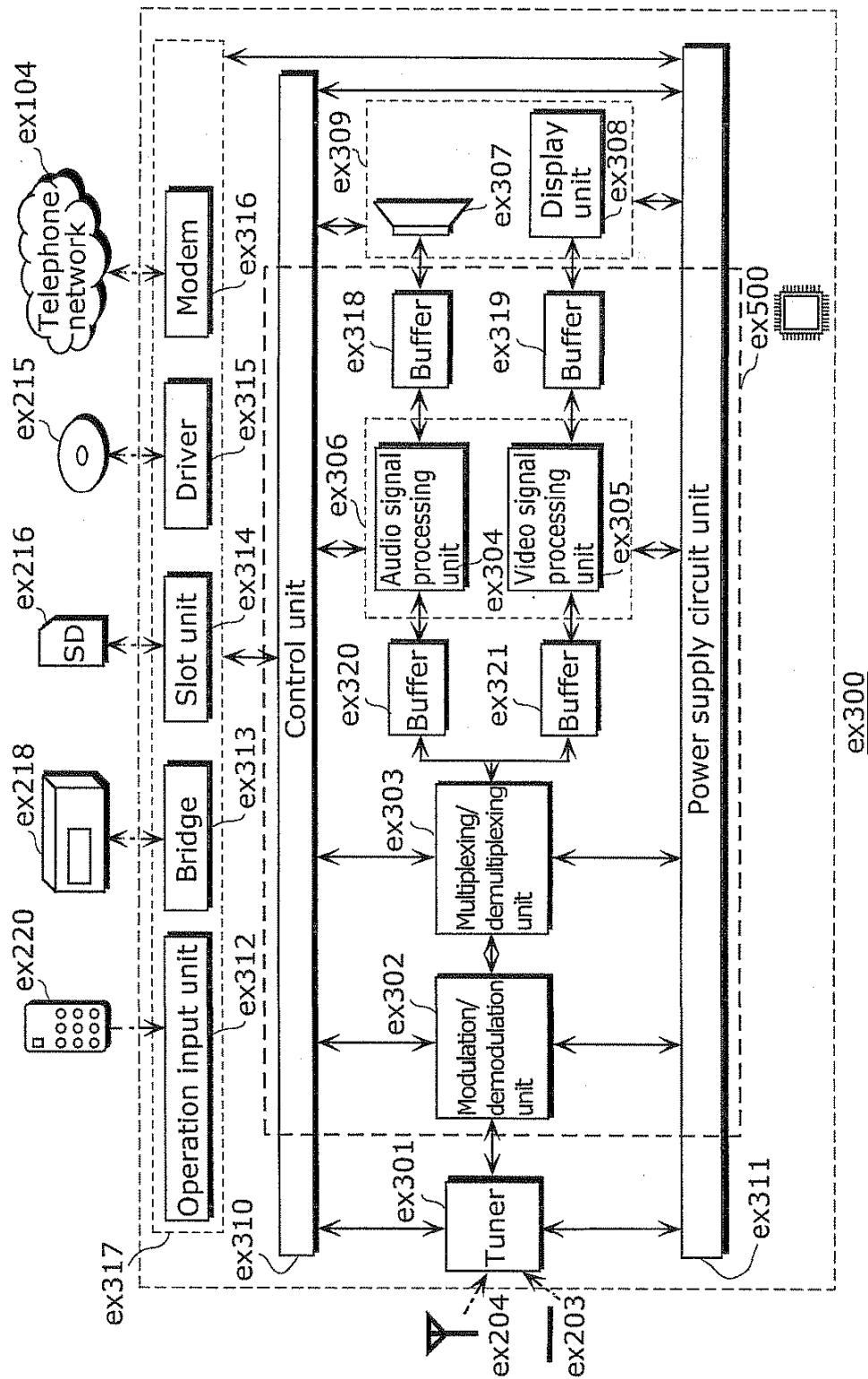


FIG. 31

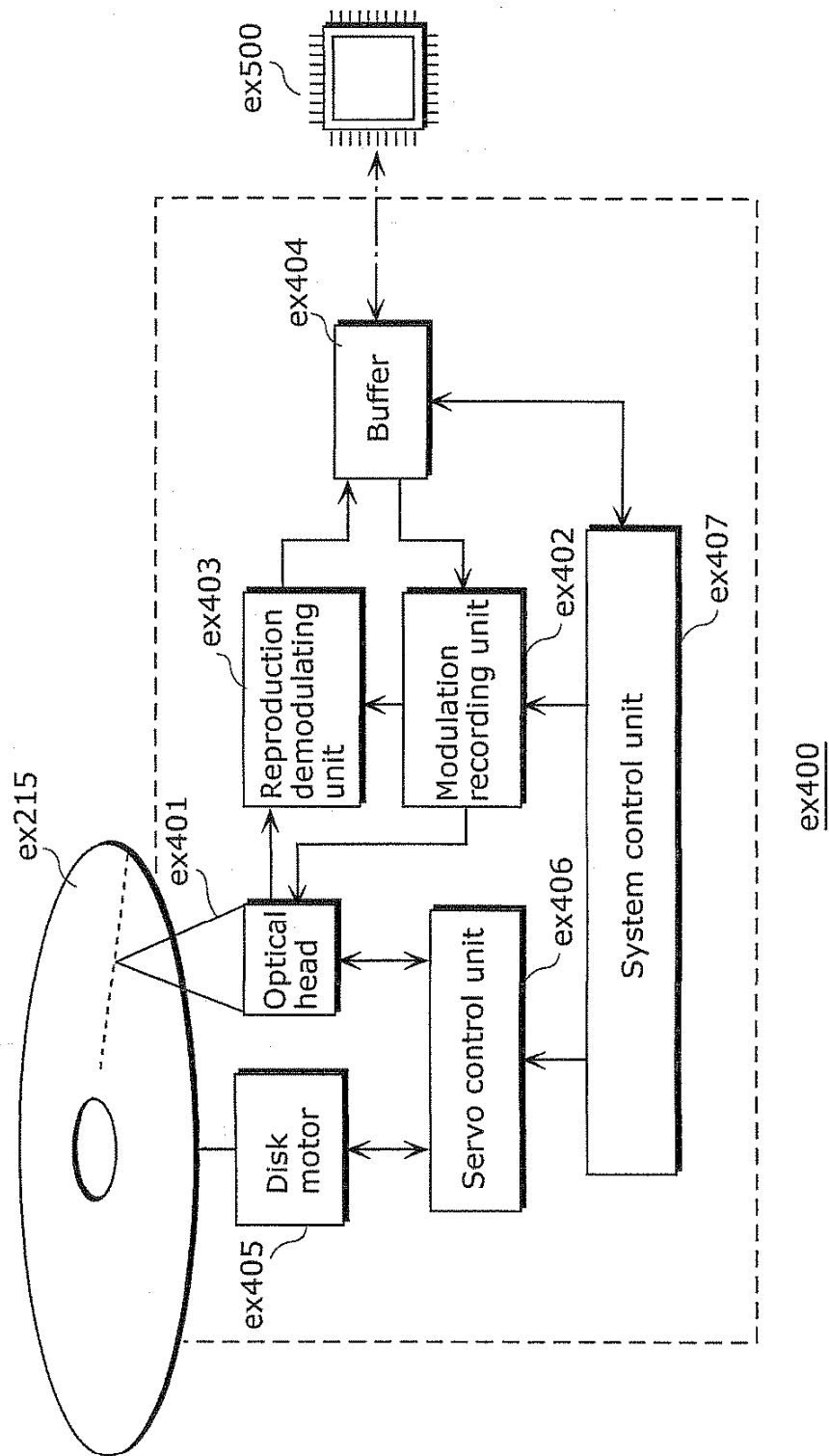


FIG. 32

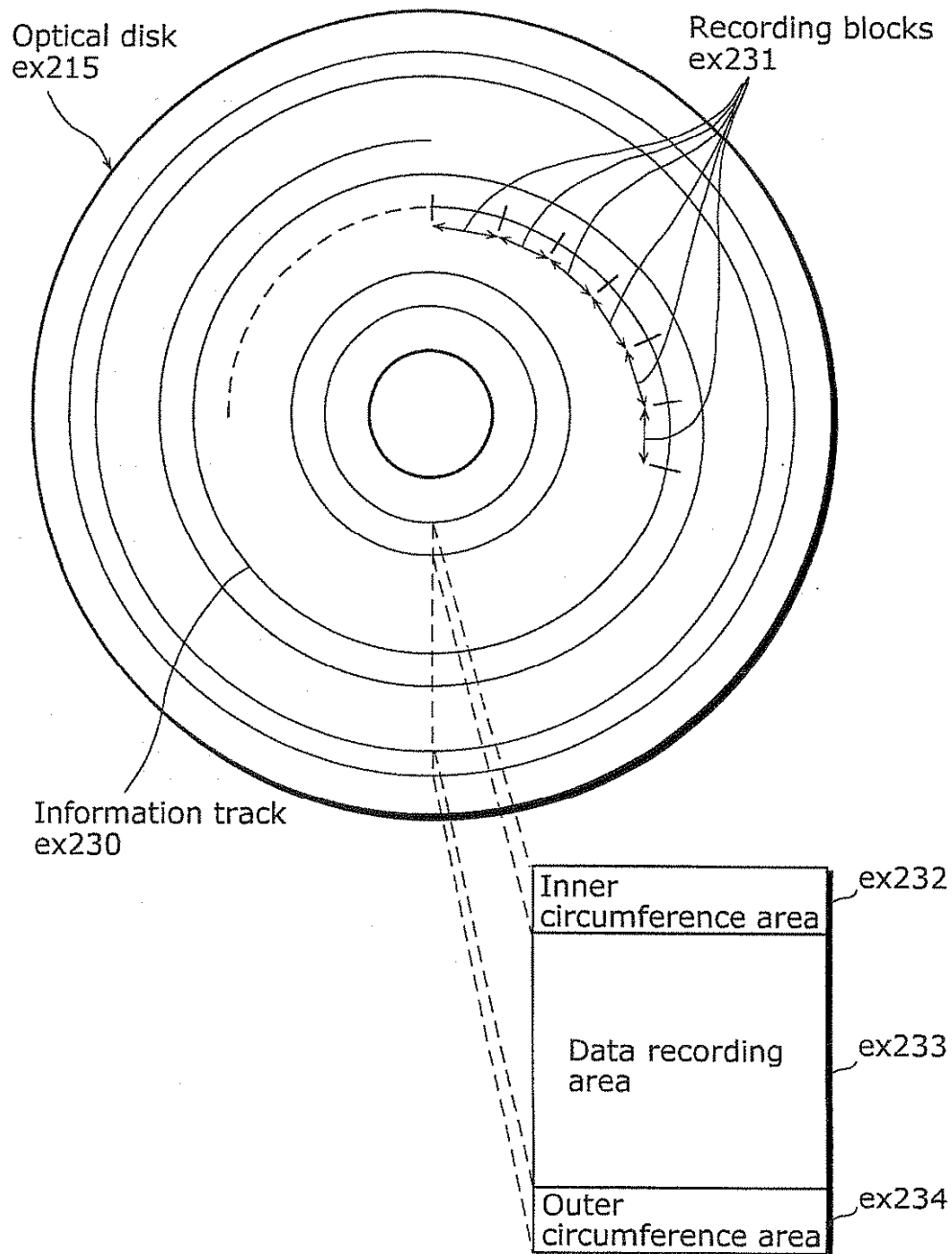


FIG. 33A

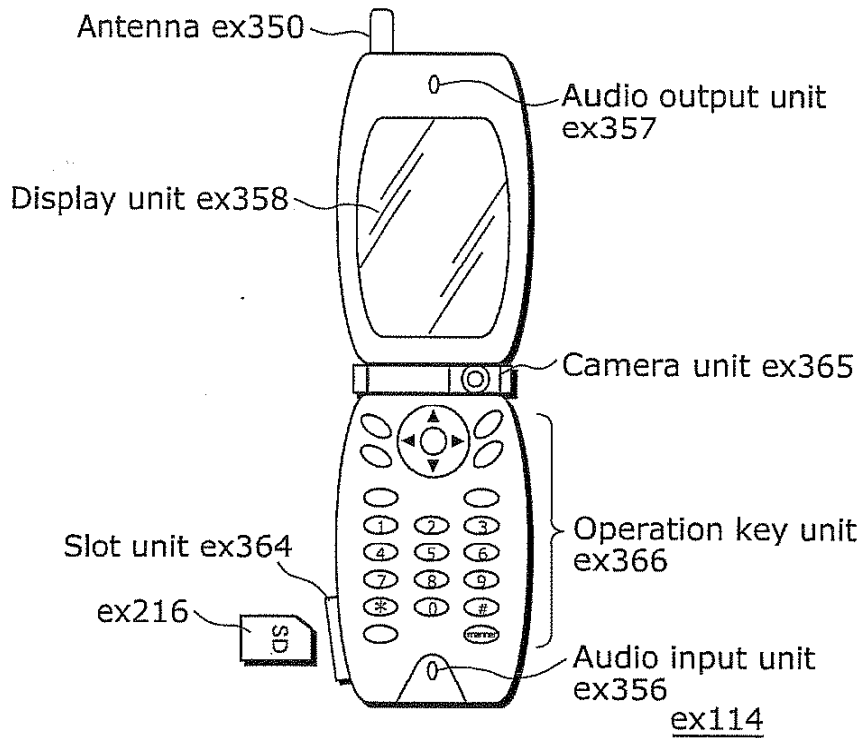


FIG. 33B

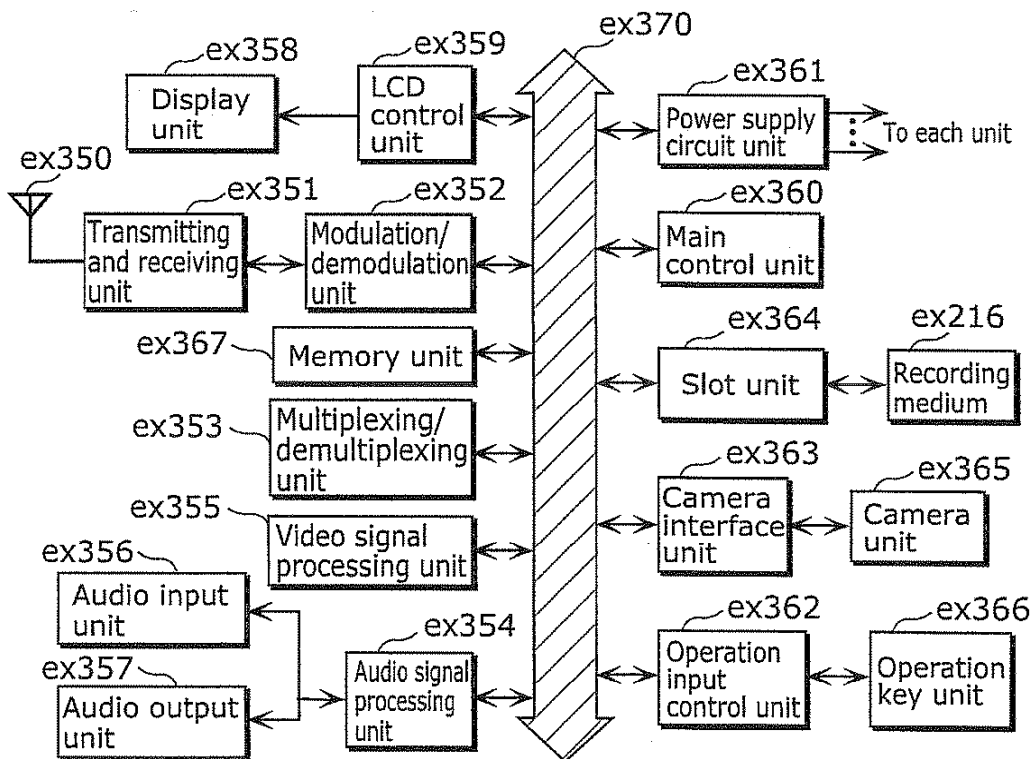


FIG. 34

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. 35

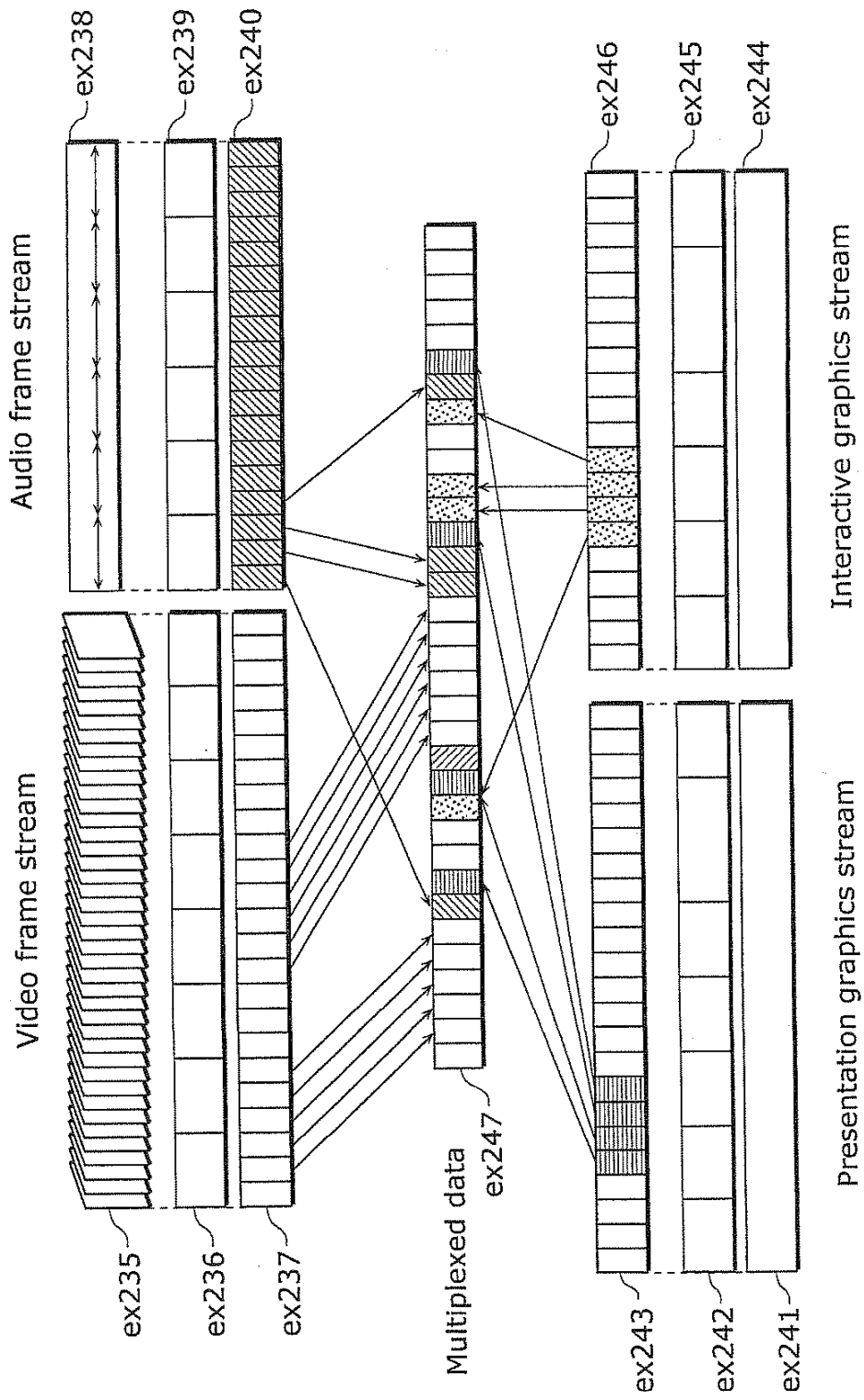


FIG. 36

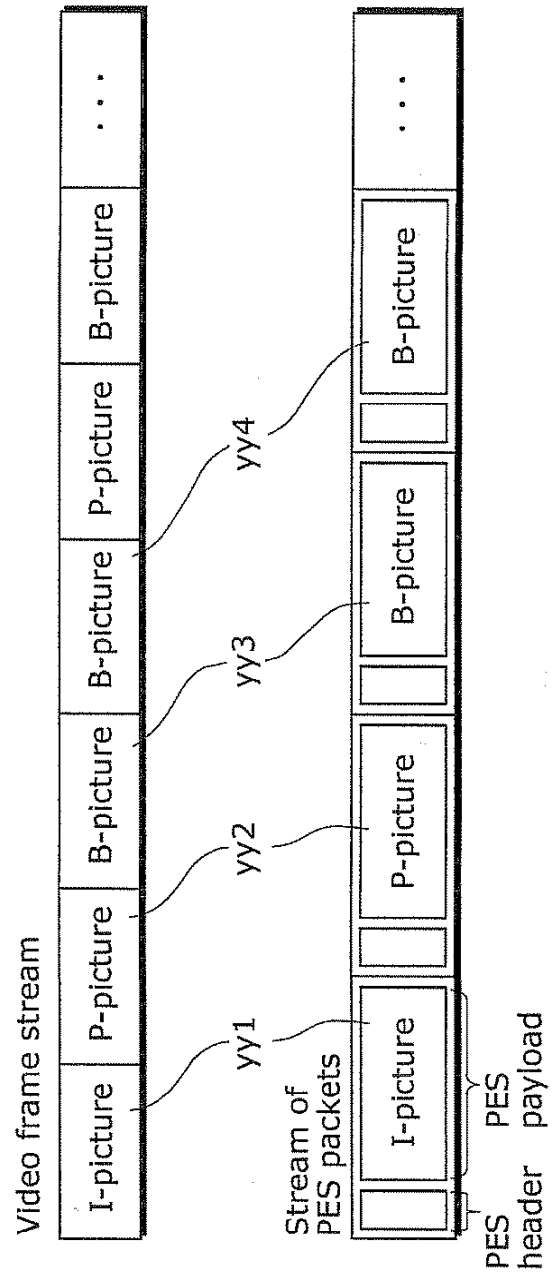


FIG. 37

Stream of TS packets

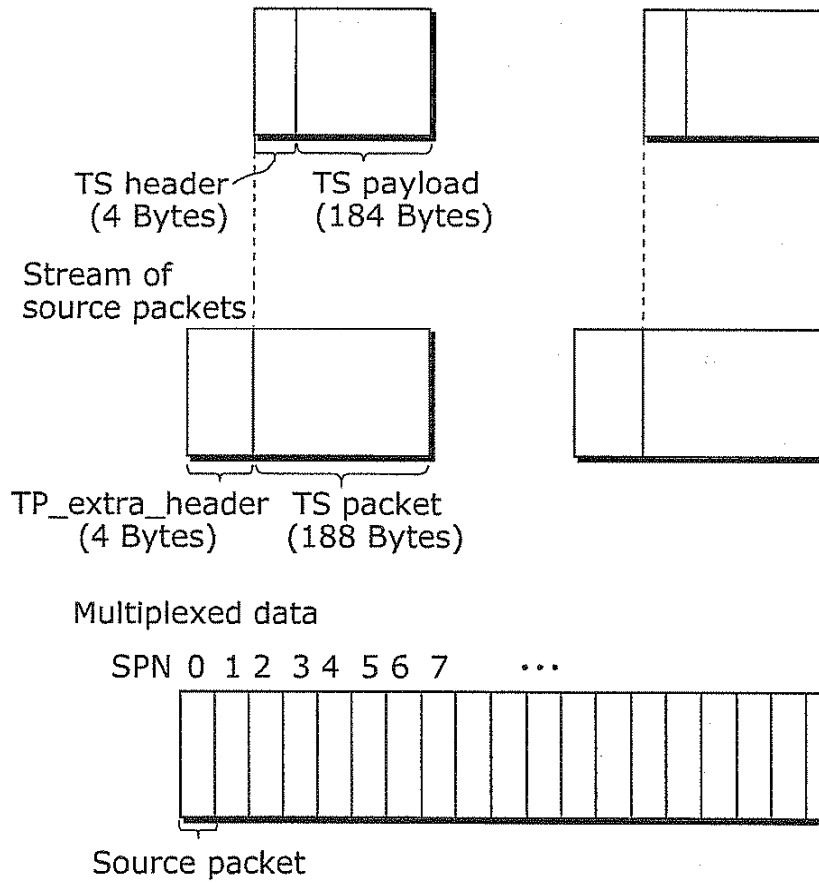


FIG. 38

Data structure of PMT

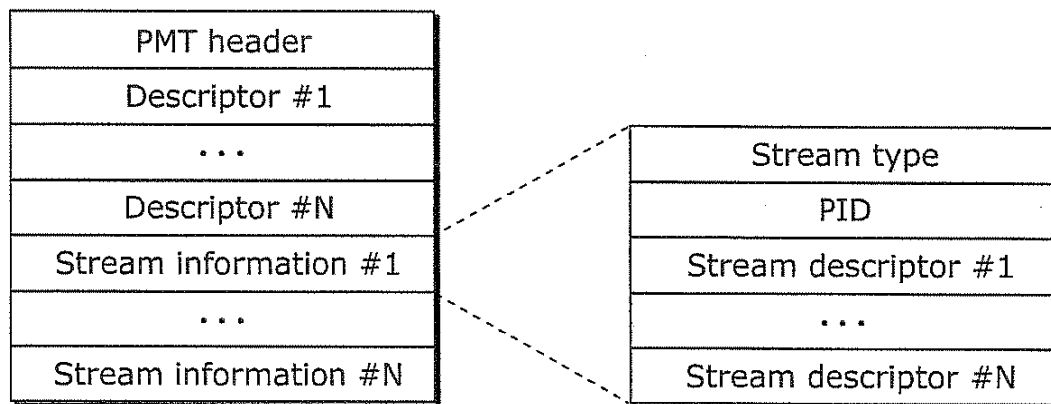


FIG. 39

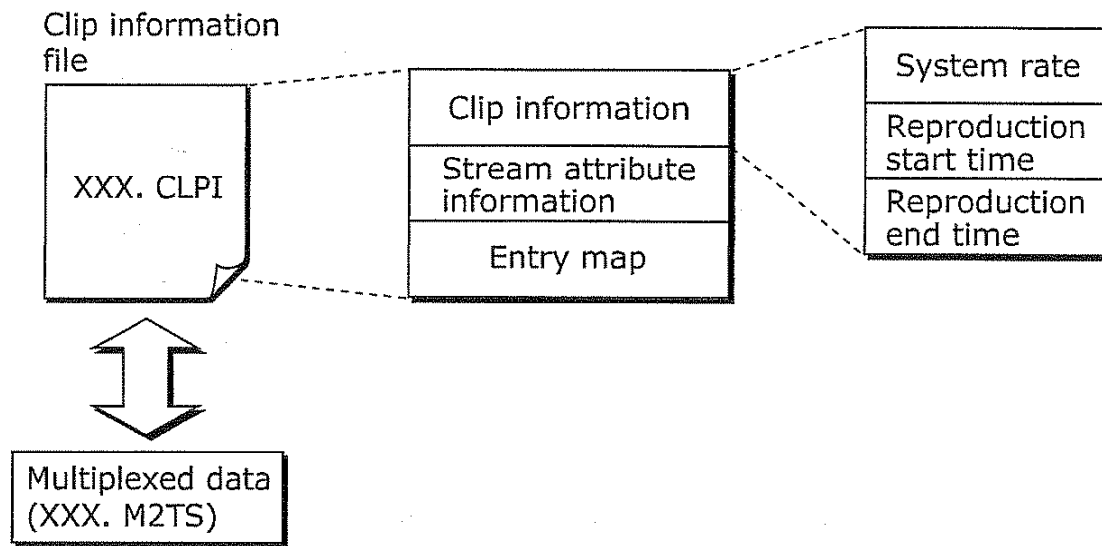


FIG. 40

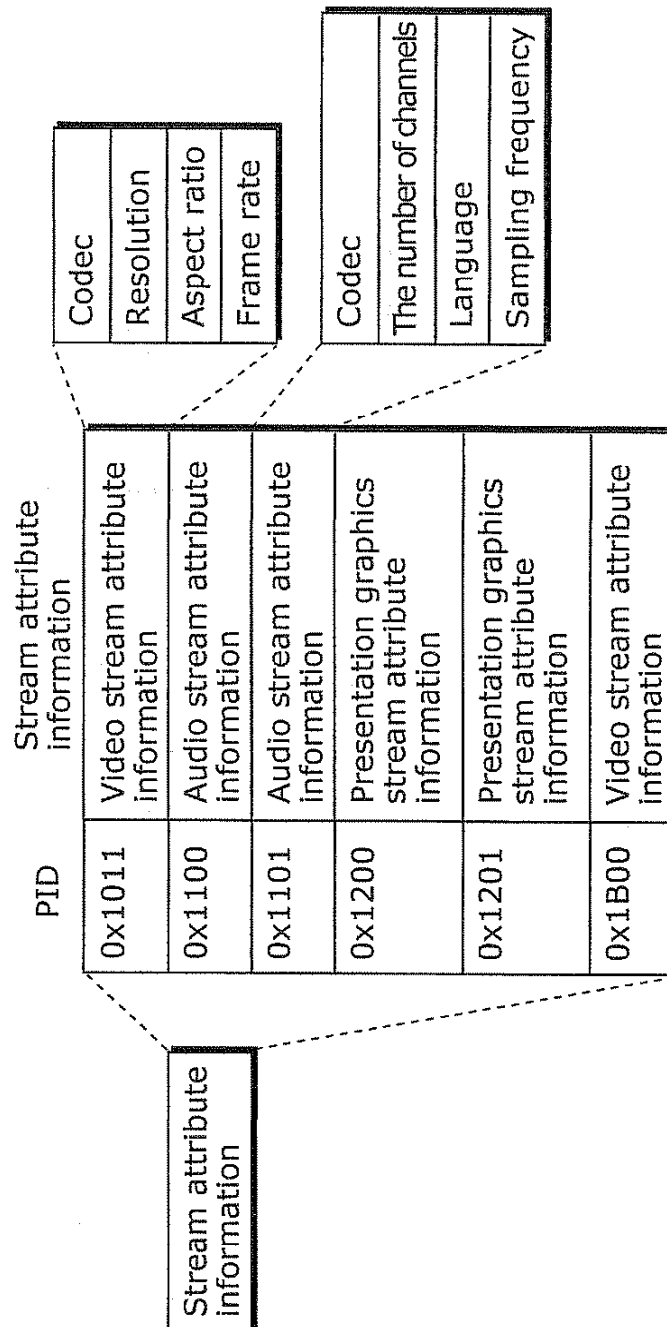


FIG. 41

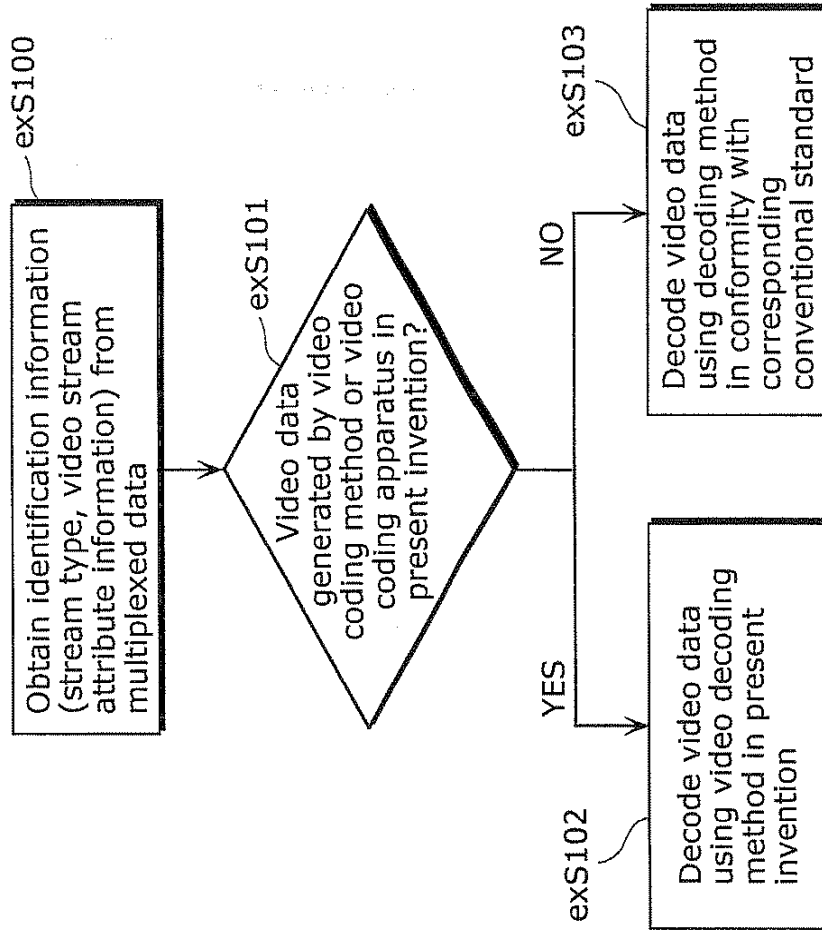


FIG. 42

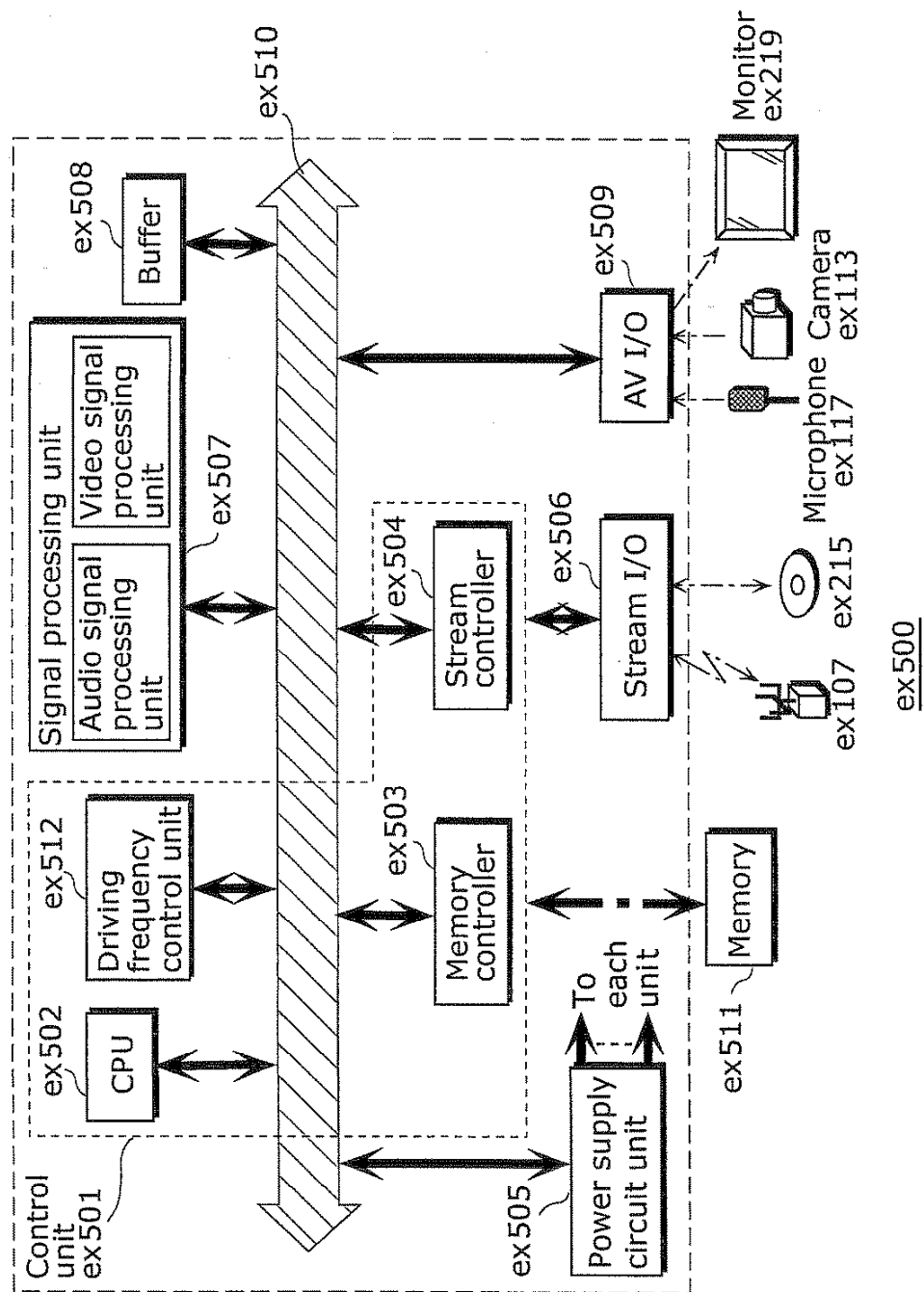


FIG. 43

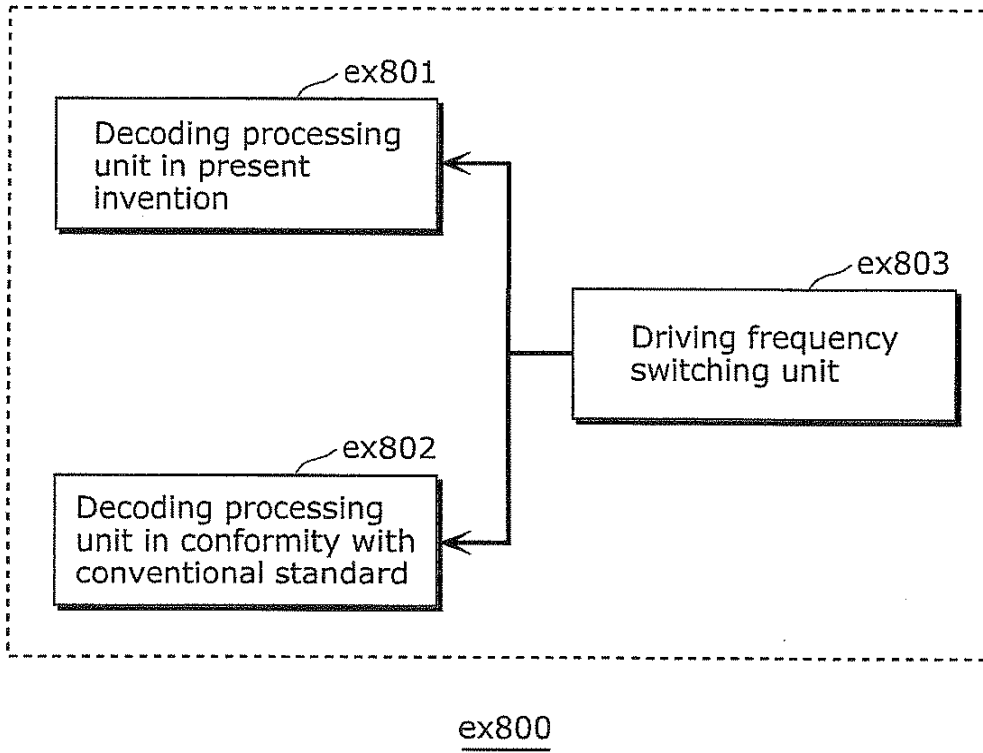


FIG. 44

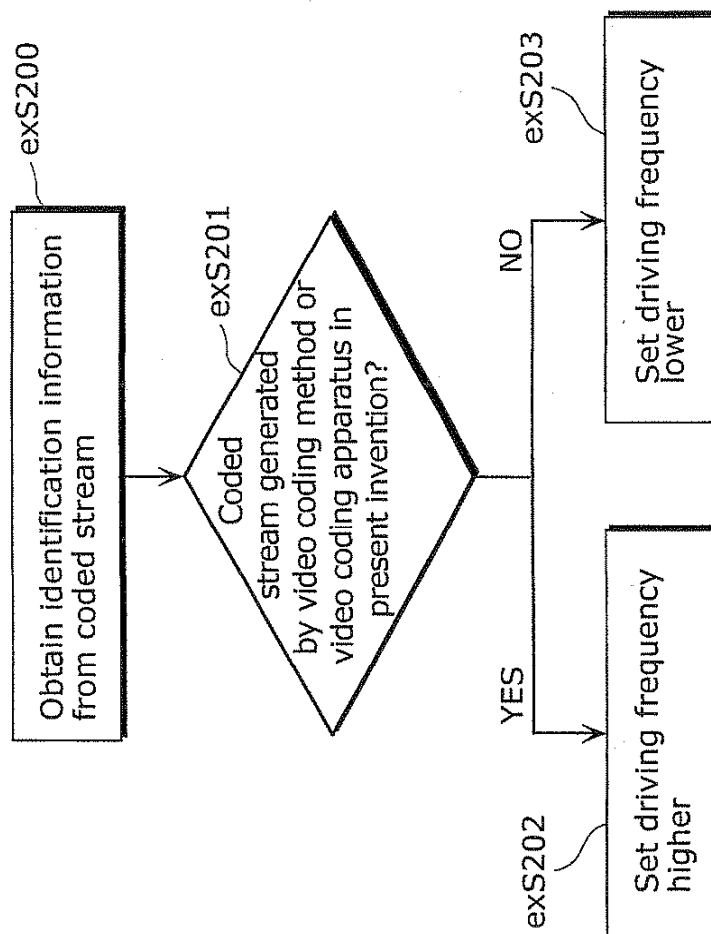


FIG. 45

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

FIG. 46A

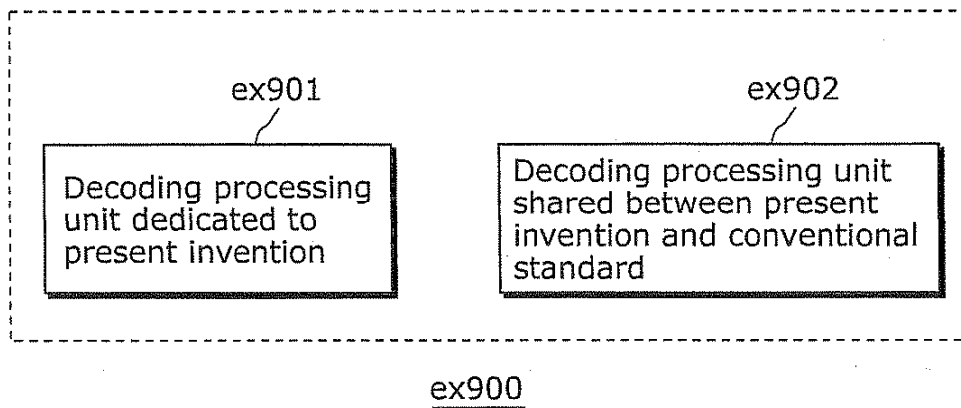


FIG. 46B

