FIG. 1A

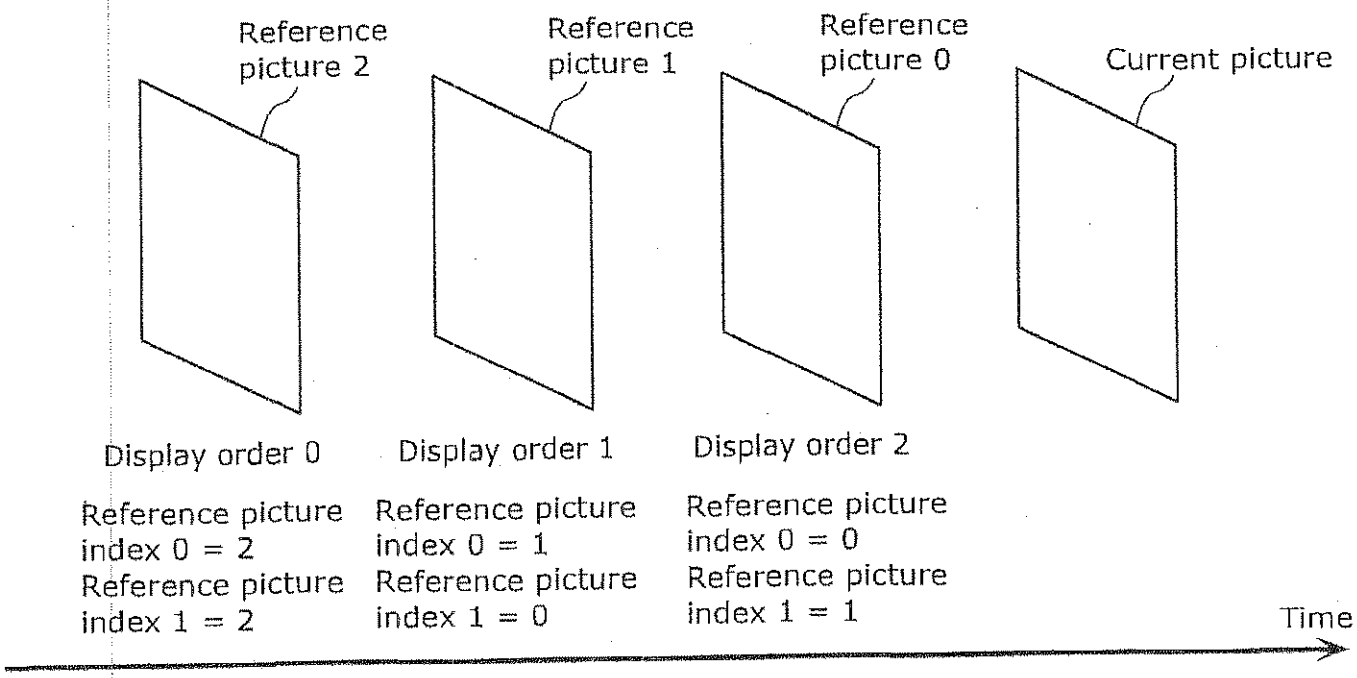

Reference picture 2
Reference picture 1
Reference picture 0
Current picture
Display order 0
Display order 1
Display order 2
Reference picture index 0 = 2
Reference picture index 1 = 2
Reference picture index 0 = 1
Reference picture index 1 = 0
Reference picture index 0 = 0
Reference picture index 1 = 1
Time

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

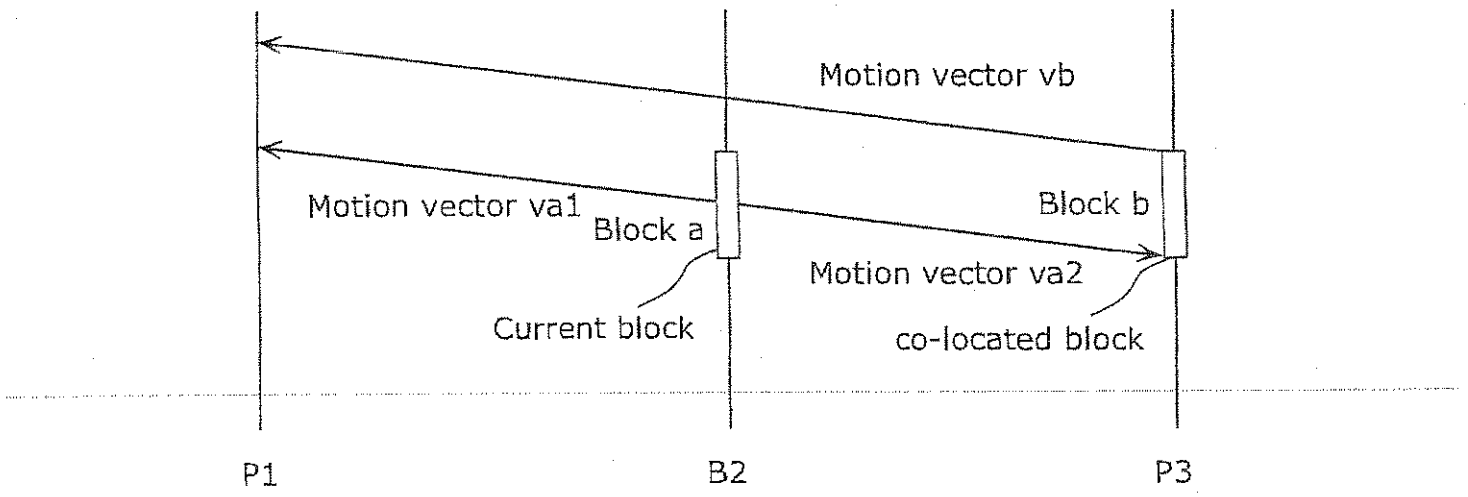


FIG. 3

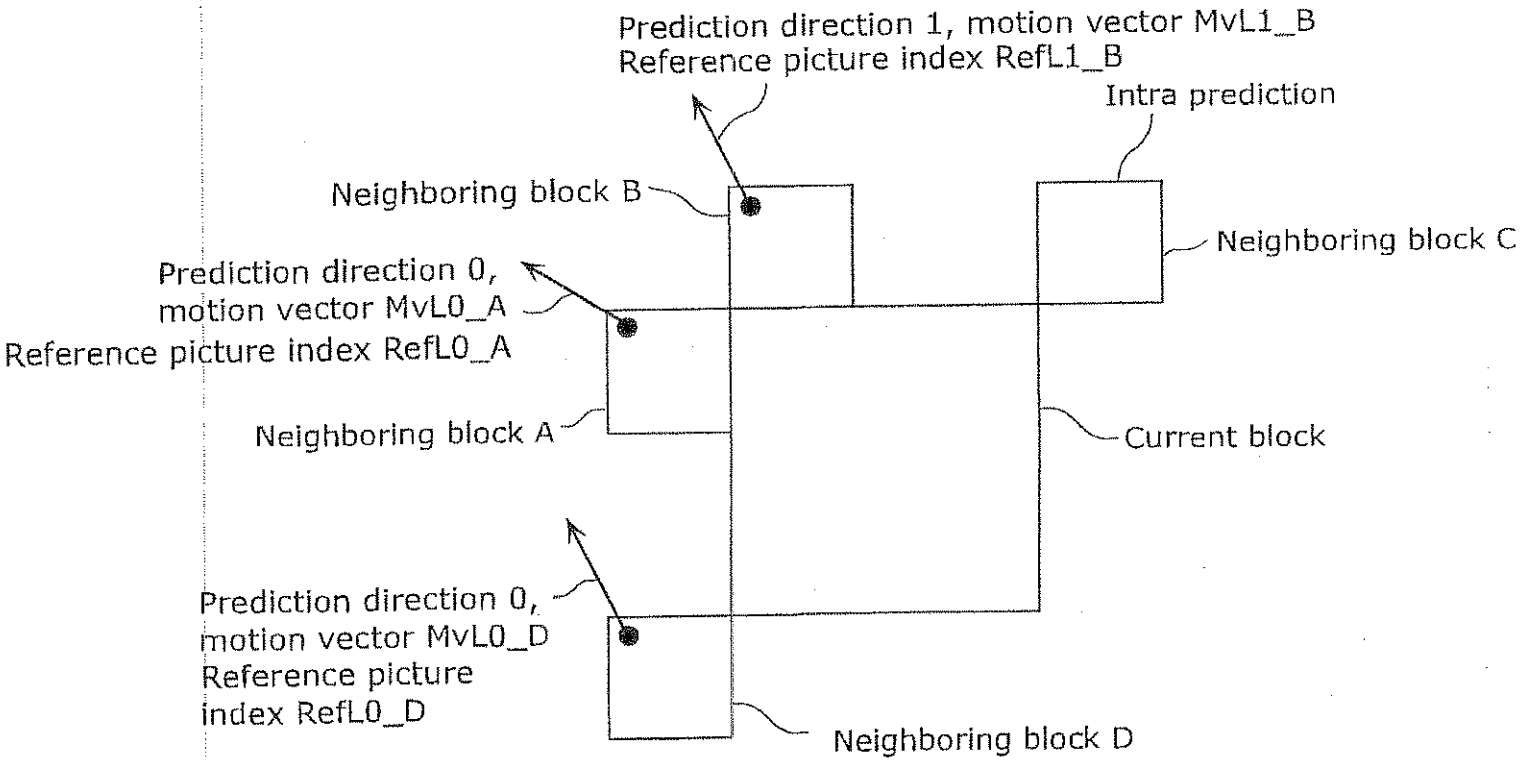

Prediction direction 1, motion vector MvL1_B
Reference picture index RefL1_B
Intra prediction
Neighboring block B
Neighboring block C
Prediction direction 0,
motion vector MvL0_A
Reference picture index RefL0_A
Neighboring block A
Current block
Prediction direction 0,
motion vector MvL0_D
Reference picture
index RefL0_D
Neighboring block D

FIG. 4

Merging block candidate index	Merging block candidate
0	Neighboring block A (Prediction direction 0, (MvL0_A, RefL0_A))
1	Neighboring block B (Prediction direction 1, (MvL1_B, RefL1_B))
2	co-located merging block (Bi-directional prediction, (MvL0_Col, RefL0_Col), (MvL1_Col, RefL1_Col))
3	Unusable-for-merging candidate (Neighboring block C, intra prediction)
4	Neighboring block D (Prediction direction 0, (MvL0_D, RefL0_V))

When MvL0_A = MvL0_D and RefL0_A = RefL0_D, remove one of neighboring blocks A and D

Remove identical candidate and unusable-for-merging candidate

Merging block candidate index	Merging block candidate
0	Neighboring block A (Prediction direction 0, (MvL0_A, RefL0_A))
1	Neighboring block B (Prediction direction 1, (MvL1_B, RefL1_B))
2	co-located merge block (Bi-directional prediction, (MvL0_Col, RefL0_Col), (MvL1_Col, RefL1_Col))

Total number of merging block candidates = 3

FIG. 5

Size of merging block candidate list = 2

Merging block candidate index	Assigned bit sequence
0	0
1	1

Size of merging block candidate list = 3

Merging block candidate index	Assigned bit sequence
0	0
1	10
2	11

Size of merging block candidate list = 4

Merging block candidate index	Assigned bit sequence
0	0
1	10
2	110
3	111

Size of merging block candidate list = 5

Merging block candidate index	Assigned bit sequence
0	0
1	10
2	110
3	1110
4	1111

FIG. 6

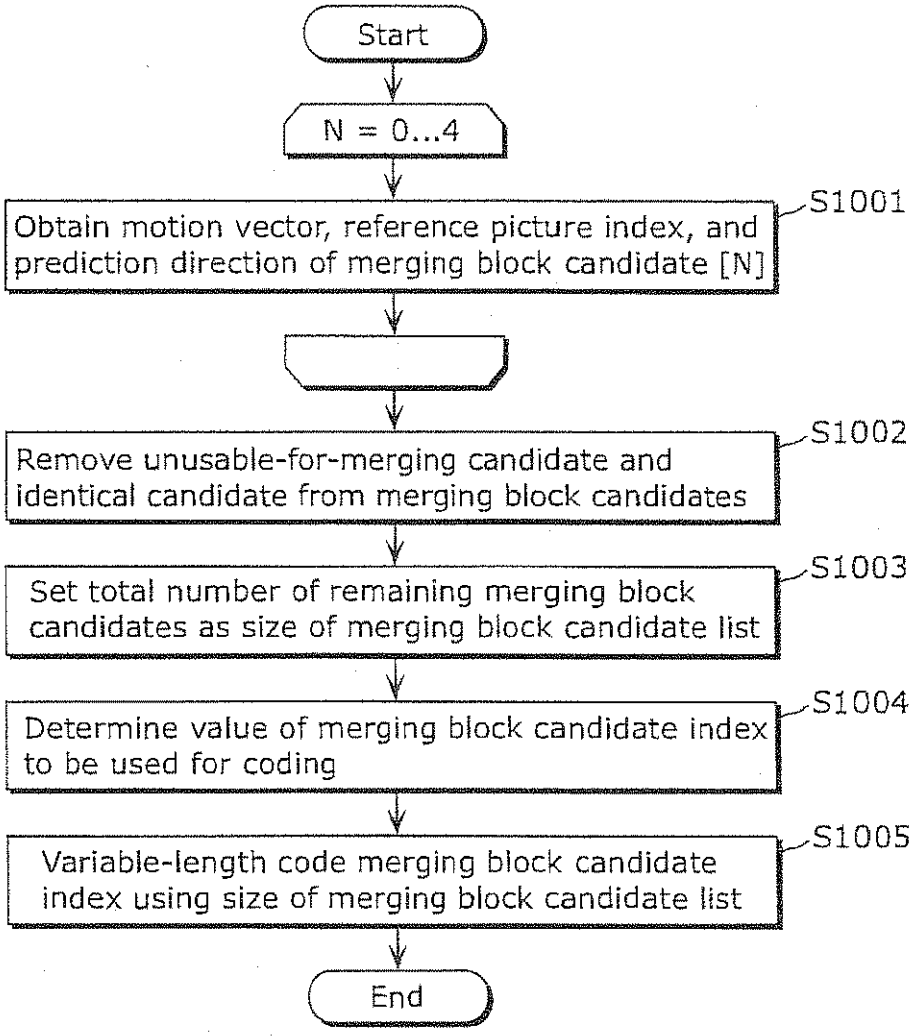

Start
N = 0...4
Obtain motion vector, reference picture index, and prediction direction of merging block candidate [N]
S1001
Remove unusable-for-merging candidate and identical candidate from merging block candidates
S1002
Set total number of remaining merging block candidates as size of merging block candidate list
S1003
Determine value of merging block candidate index to be used for coding
S1004
Variable-length code merging block candidate index using size of merging block candidate list
S1005
End

FIG. 7

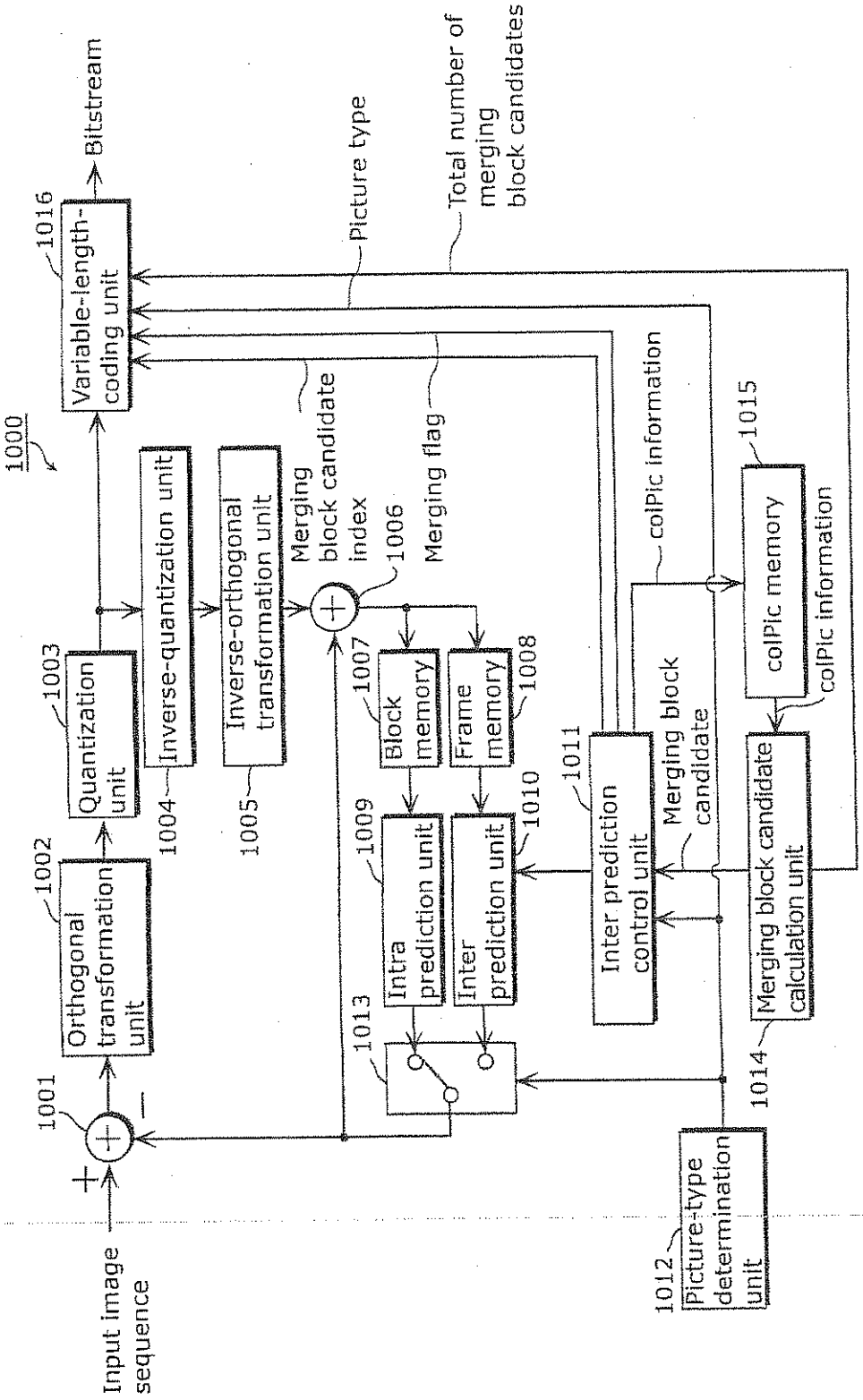

1000
Input image sequence
1001
1002
Orthogonal transformation unit
1003
Quantization unit
1004
Inverse-quantization unit
1005
Inverse-orthogonal transformation unit
1006
Merging block candidate index
Merging flag
1007
Block memory
1008
Frame memory
1009
Intra prediction unit
1010
Inter prediction unit
1011
Inter prediction control unit
1012
Picture-type determination unit
1013
1014
Merging block candidate calculation unit
1015
colPic memory
1016
Variable-length-coding unit
Bitstream
Picture type
Total number of merging block candidates
colPic information
Merging block candidate
colPic information

FIG. 8

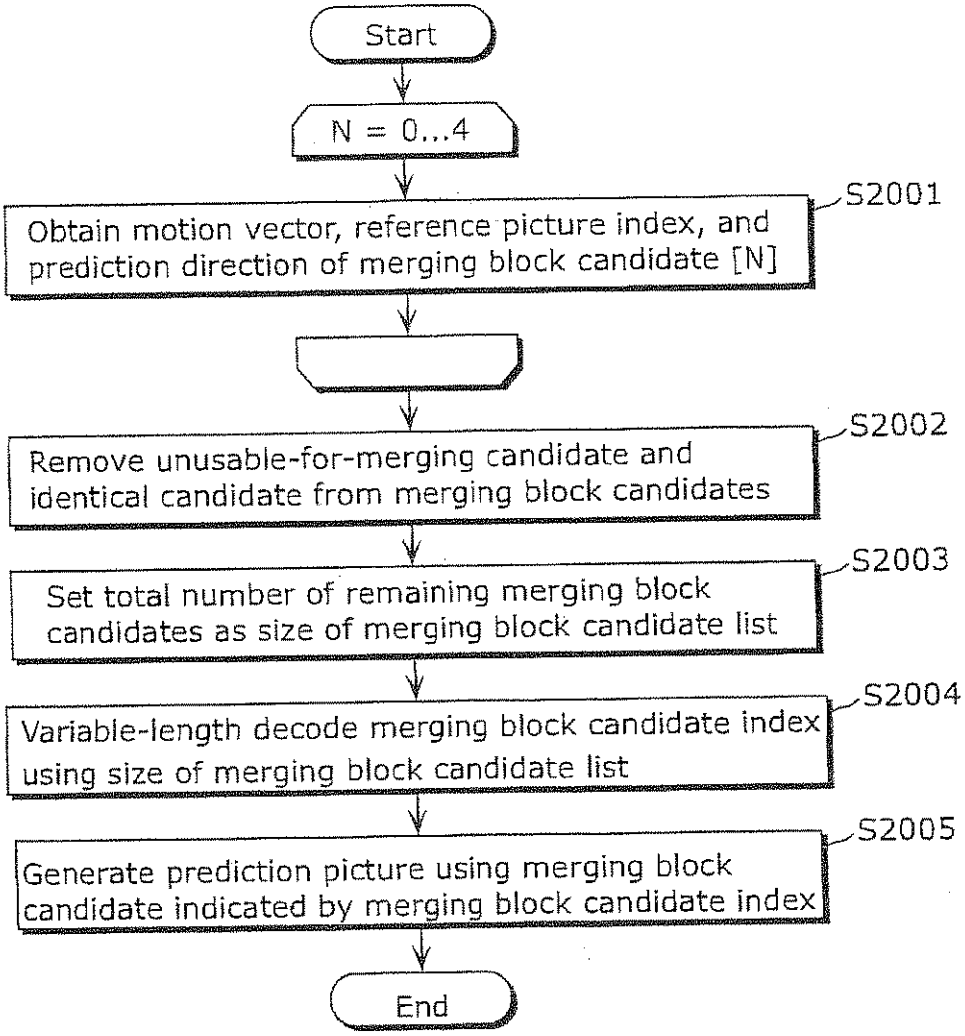

Start
N = 0...4
Obtain motion vector, reference picture index, and prediction direction of merging block candidate [N]
S2001
Remove unusable-for-merging candidate and identical candidate from merging block candidates
S2002
Set total number of remaining merging block candidates as size of merging block candidate list
S2003
Variable-length decode merging block candidate index using size of merging block candidate list
S2004
Generate prediction picture using merging block candidate indicated by merging block candidate index
S2005
End

FIG. 9

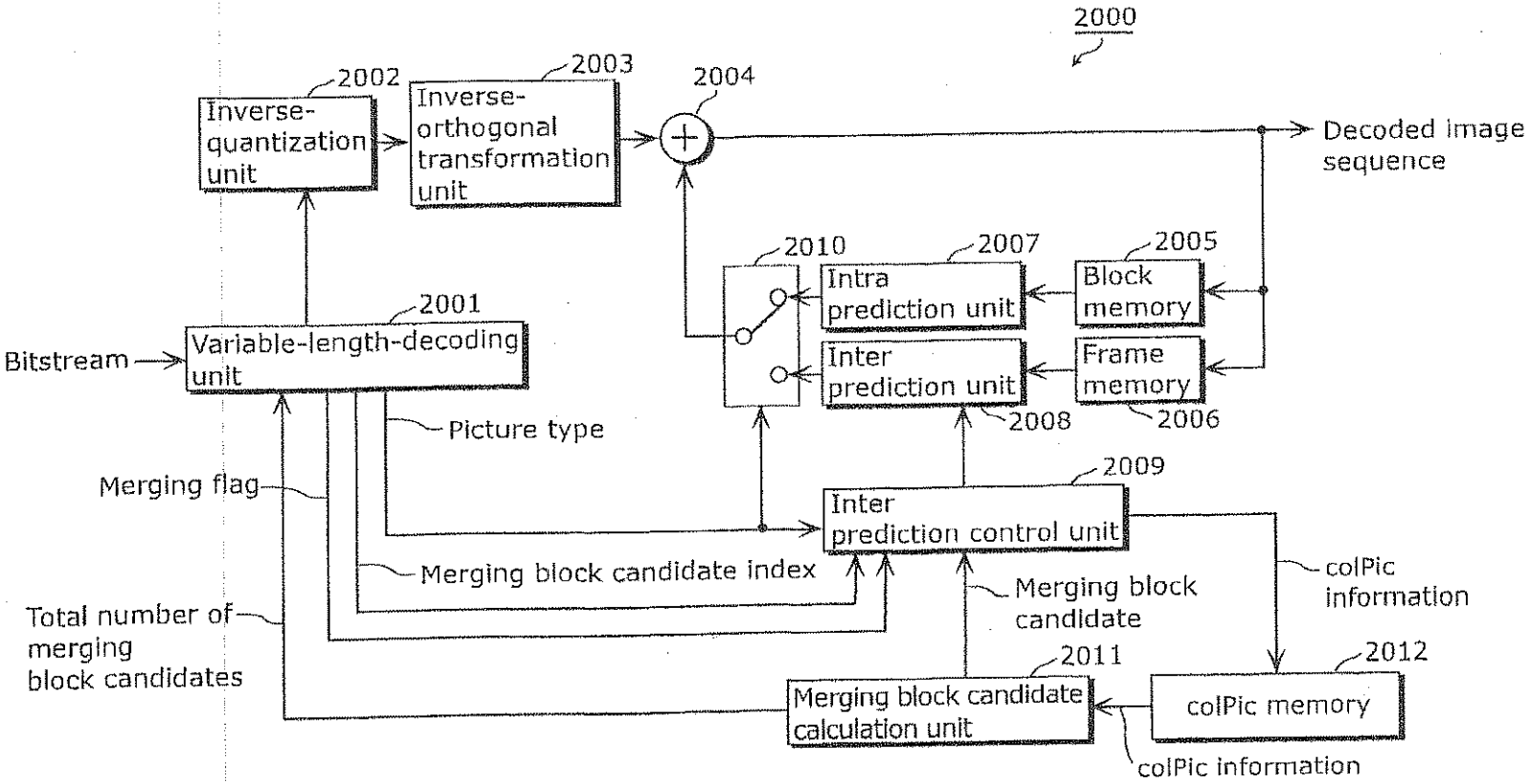
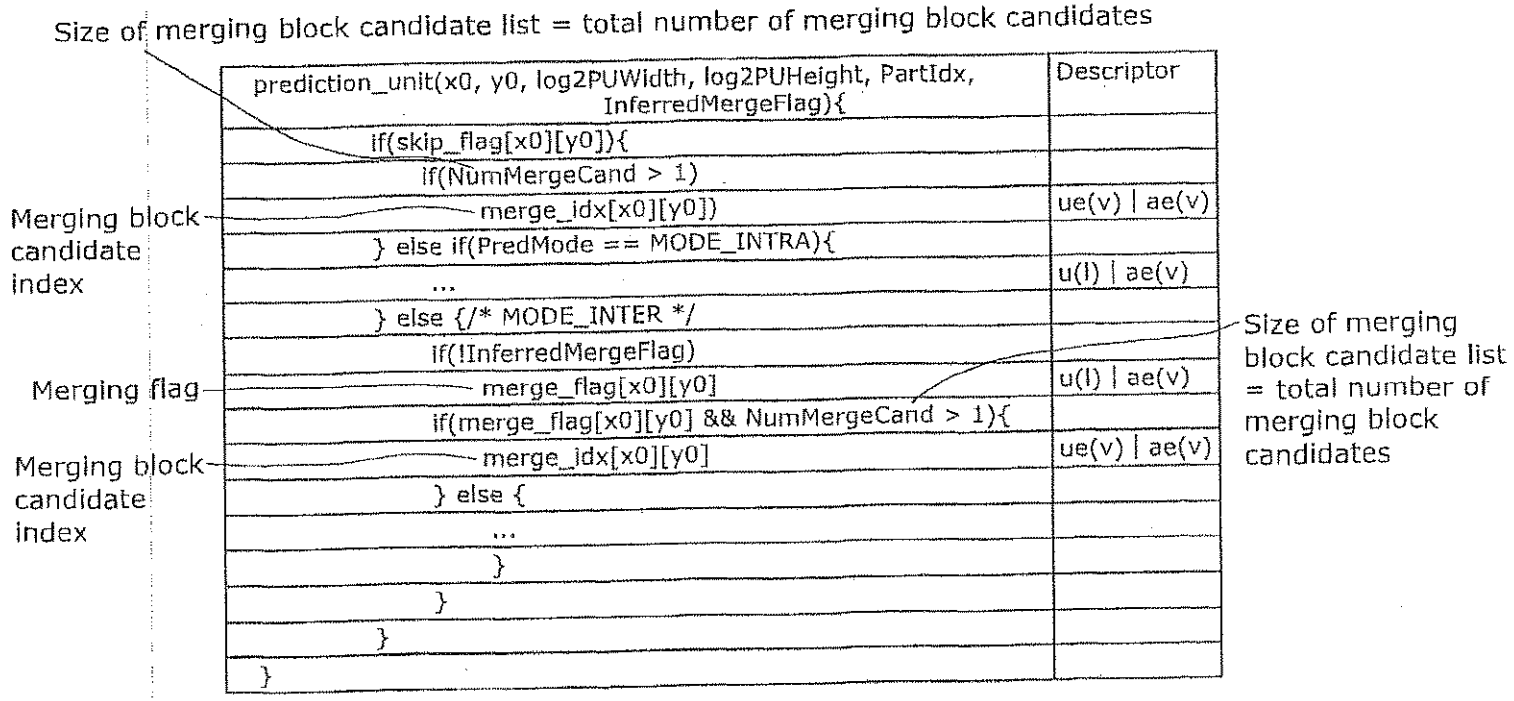

2000
2002
Inverse-quantization unit
2003
Inverse-orthogonal transformation unit
2004
Decoded image sequence
2010
2007
Intra prediction unit
2005
Block memory
2001
Variable-length-decoding unit
Bitstream
Inter prediction unit
Frame memory
2008
2006
Picture type
2009
Merging flag
Inter prediction control unit
Merging block candidate index
colPic information
Merging block candidate
Total number of merging block candidates
2011
2012
Merging block candidate calculation unit
colPic memory
colPic information

FIG. 10



prediction_unit(x0, y0, log2PUWidth, log2PUHeight, PartIdx, InferredMergeFlag){	Descriptor
if(skip_flag[x0][y0]){	
if(NumMergeCand > 1)	
merge_idx[x0][y0])	ue(v) | ae(v)
} else if(PredMode == MODE_INTRA){	
...	u(l) | ae(v)
} else {/* MODE_INTER */	
if(!InferredMergeFlag)	
merge_flag[x0][y0]	u(l) | ae(v)
if(merge_flag[x0][y0] && NumMergeCand > 1){	
merge_idx[x0][y0]	ue(v) | ae(v)
} else {	
...	
}	
}	
}	
}	

FIG. 11

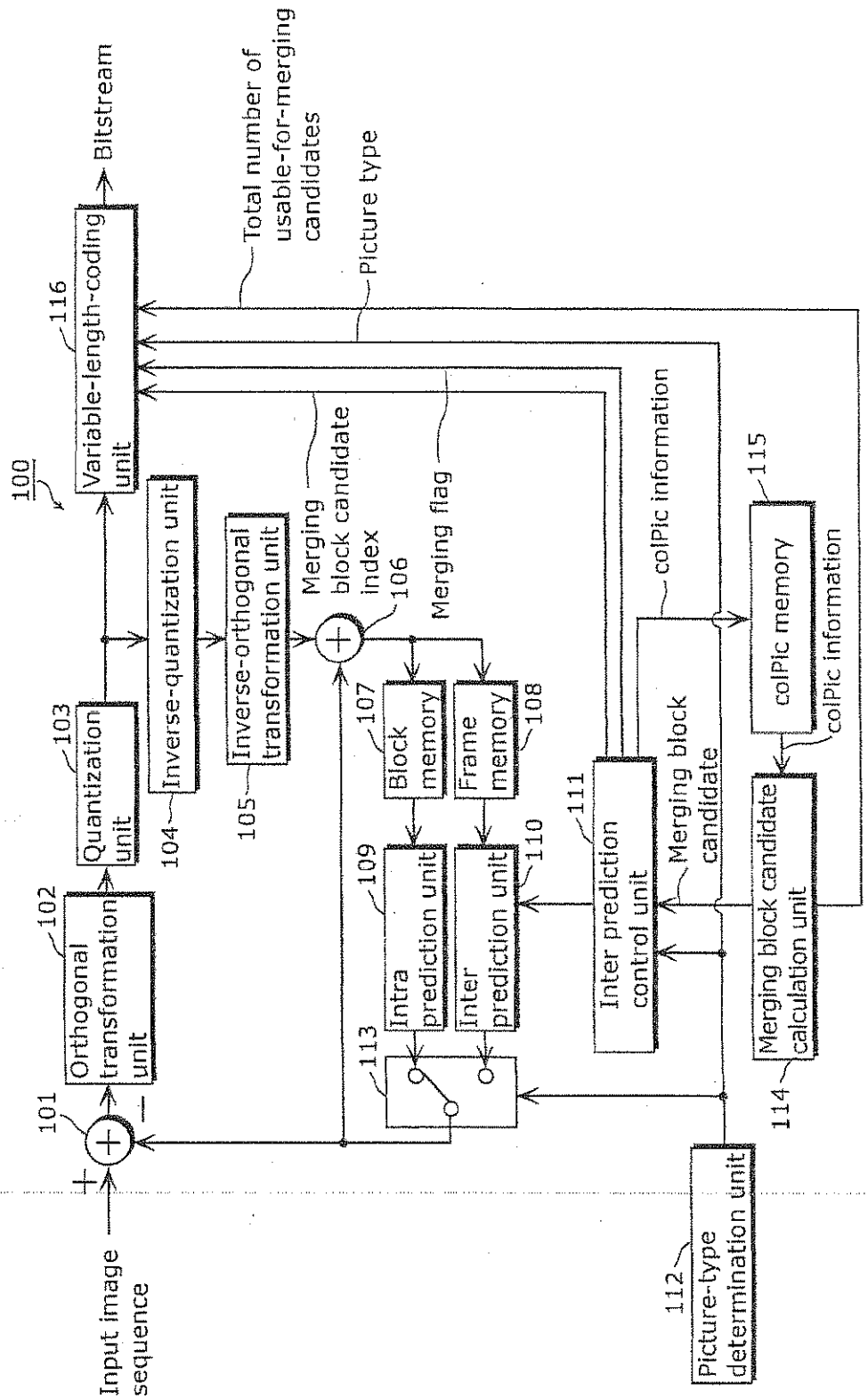

100
Input image sequence
101
Orthogonal transformation unit
102
Quantization unit
103
Inverse-quantization unit
104
Inverse-orthogonal transformation unit
105
Merging block candidate index
106
Block memory
107
Frame memory
108
Intra prediction unit
109
Inter prediction unit
110
Inter prediction control unit
111
Picture-type determination unit
112
113
Merging block candidate calculation unit
114
colPic memory
115
Variable-length-coding unit
116
Bitstream
Total number of usable-for-merging candidates
Picture type
Merging flag
colPic information
Merging block candidate
colPic information

FIG. 12

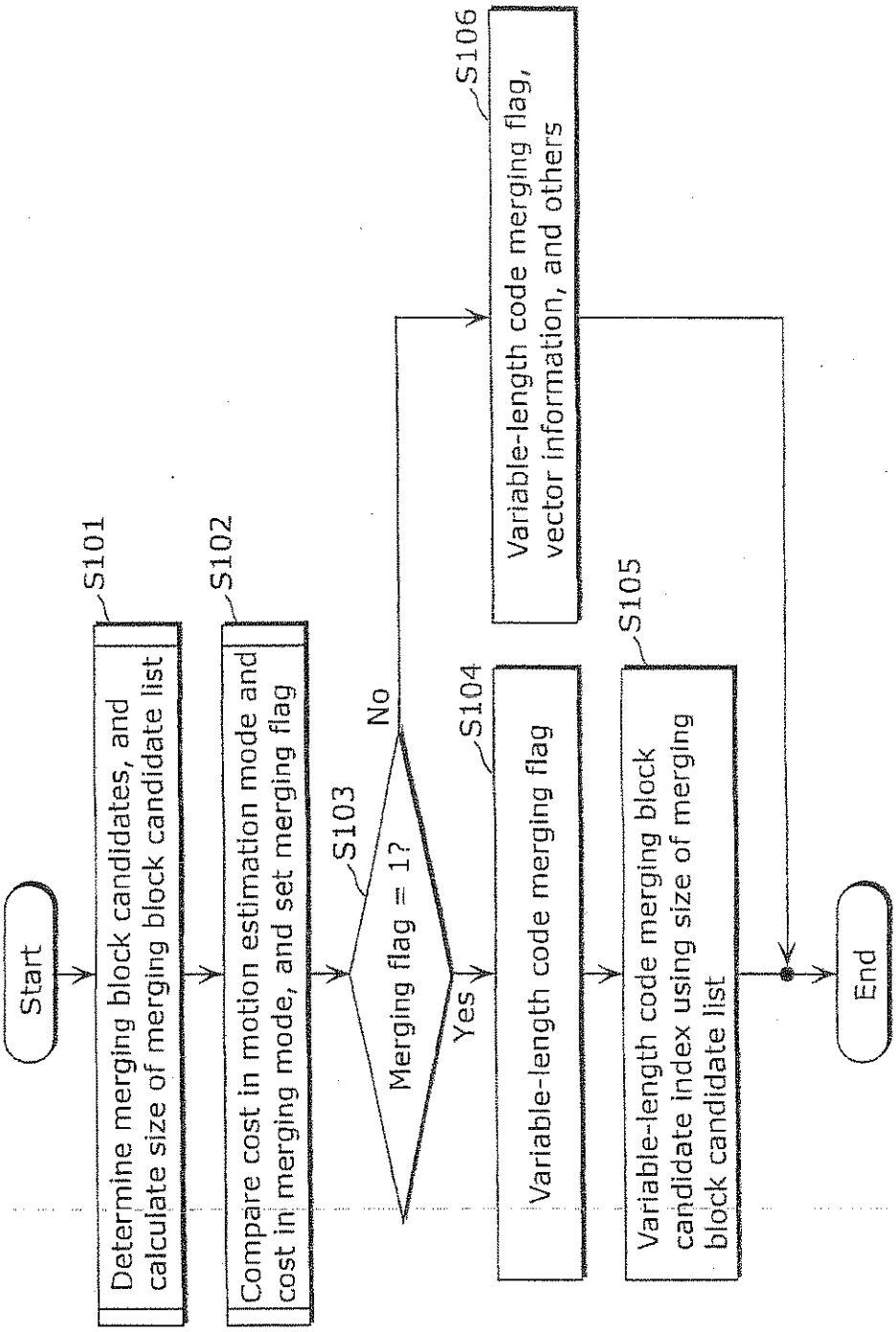

Start
S101
Determine merging block candidates, and calculate size of merging block candidate list
S102
Compare cost in motion estimation mode and cost in merging mode, and set merging flag
S103
Merging flag = 1?
Yes
No
S104
Variable-length code merging flag
S105
Variable-length code merging block candidate index using size of merging block candidate list
S106
Variable-length code merging flag, vector information, and others
End

FIG. 13

(a)

Merging block candidate index	Merging block candidate
0	Neighboring block A (Prediction direction 0, (MvL0_A, RefL0_A))
1	Neighboring block B (Prediction direction 1, (MvL1_B, RefL1_B))
2	co-located merge block (Bi-directional prediction, (MvL0_Col, RefL0_Col), (MvL1_Col, RefL1_Col))
3	Unusable-for-merging candidate (Neighboring block C, intra prediction)
4	Neighboring block D (Prediction direction 0, (MvL0_D, RefL0_V))

When MvL0_A = MvL0_D and RefL0_A = RefL0_D, remove one of neighboring blocks A and D

- Remove identical candidate and unusable-for-merging candidate
- Add new candidate and increment total number of merging block candidate by 1 when total number of usable-for-merging candidates - total number of merging block candidates > 0

(b)

Merging block candidate index	Merging block candidate
0	Neighboring block A (Prediction direction 0 (MvL0_A, RefL0_A))
1	Neighboring block B (Prediction direction 1 (MvL1_B, RefL1_B))
2	co-located merging block (Bi-directional prediction, (MvL0_Col, RefL0_Col), (MvL1_Col, RefL1_Col))
3	New candidate

Total number of merging block candidates = 4

FIG. 14

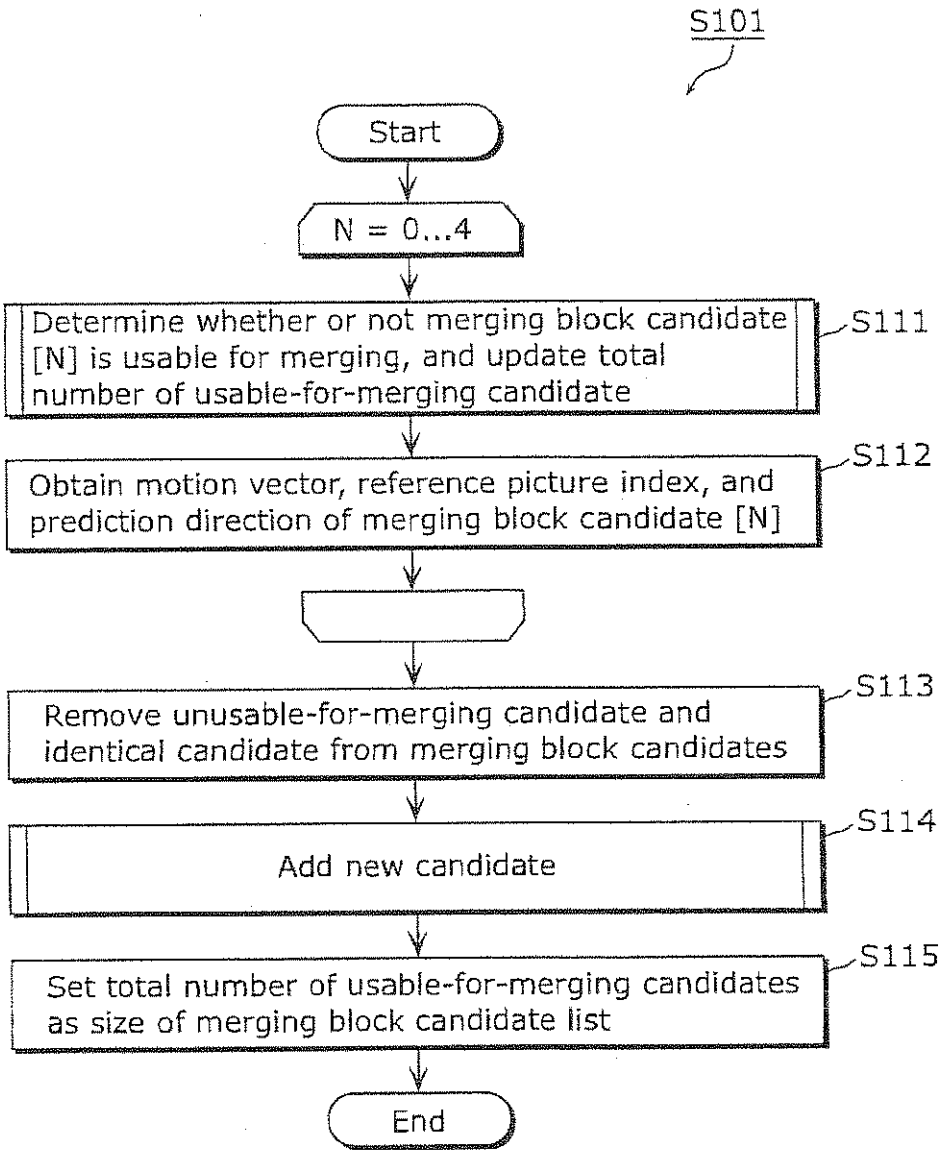

S101
Start
N = 0...4
Determine whether or not merging block candidate [N] is usable for merging, and update total number of usable-for-merging candidate
S111
Obtain motion vector, reference picture index, and prediction direction of merging block candidate [N]
S112
Remove unusable-for-merging candidate and identical candidate from merging block candidates
S113
Add new candidate
S114
Set total number of usable-for-merging candidates as size of merging block candidate list
S115
End

FIG. 15

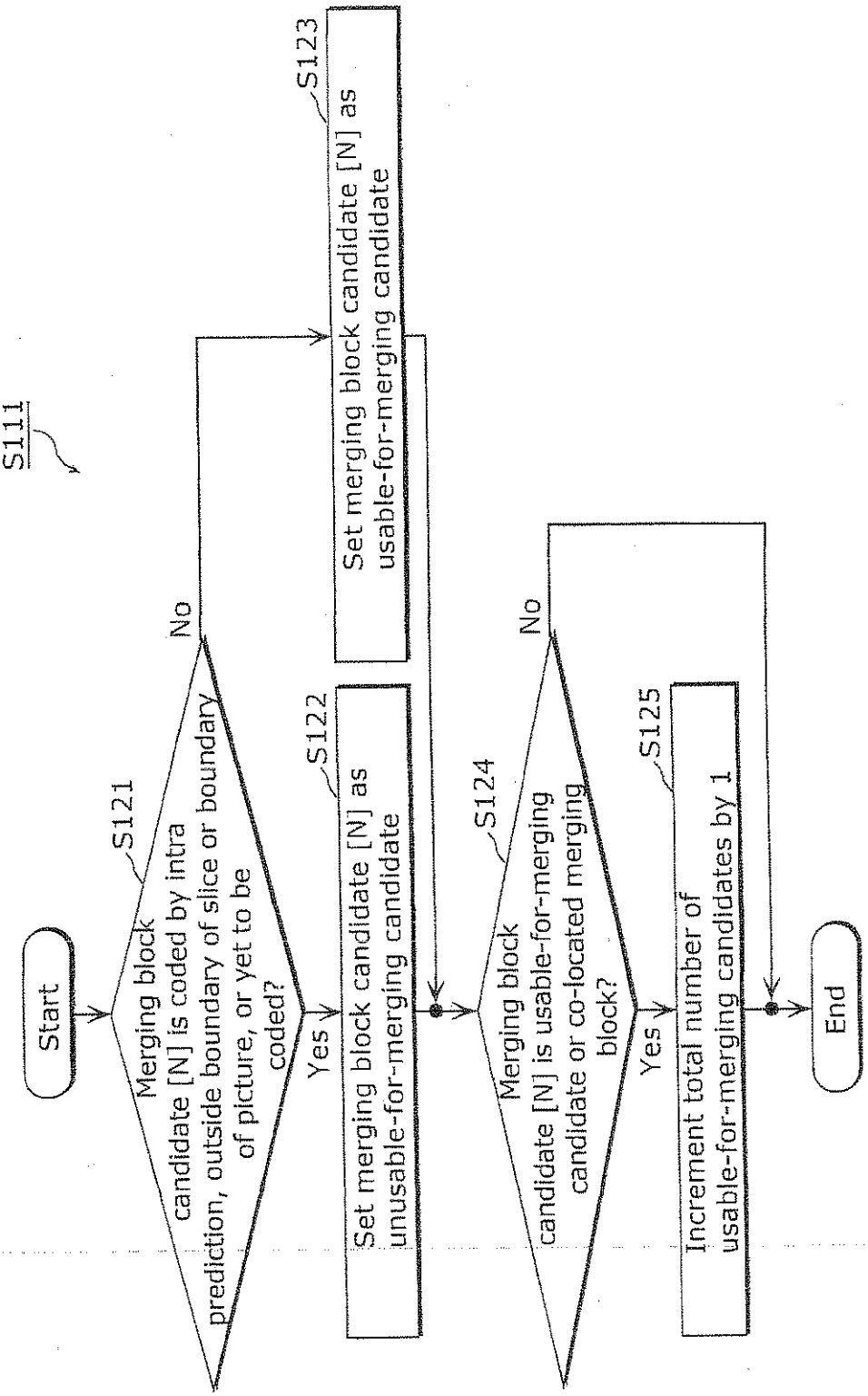

S111
Start
S121
Merging block candidate [N] is coded by intra prediction, outside boundary of slice or boundary of picture, or yet to be coded?
No
Yes
S122
Set merging block candidate [N] as unusable-for-merging candidate
S123
Set merging block candidate [N] as usable-for-merging candidate
S124
Merging block candidate [N] is usable-for-merging candidate or co-located merging block?
No
Yes
S125
Increment total number of usable-for-merging candidates by 1
End

FIG. 16

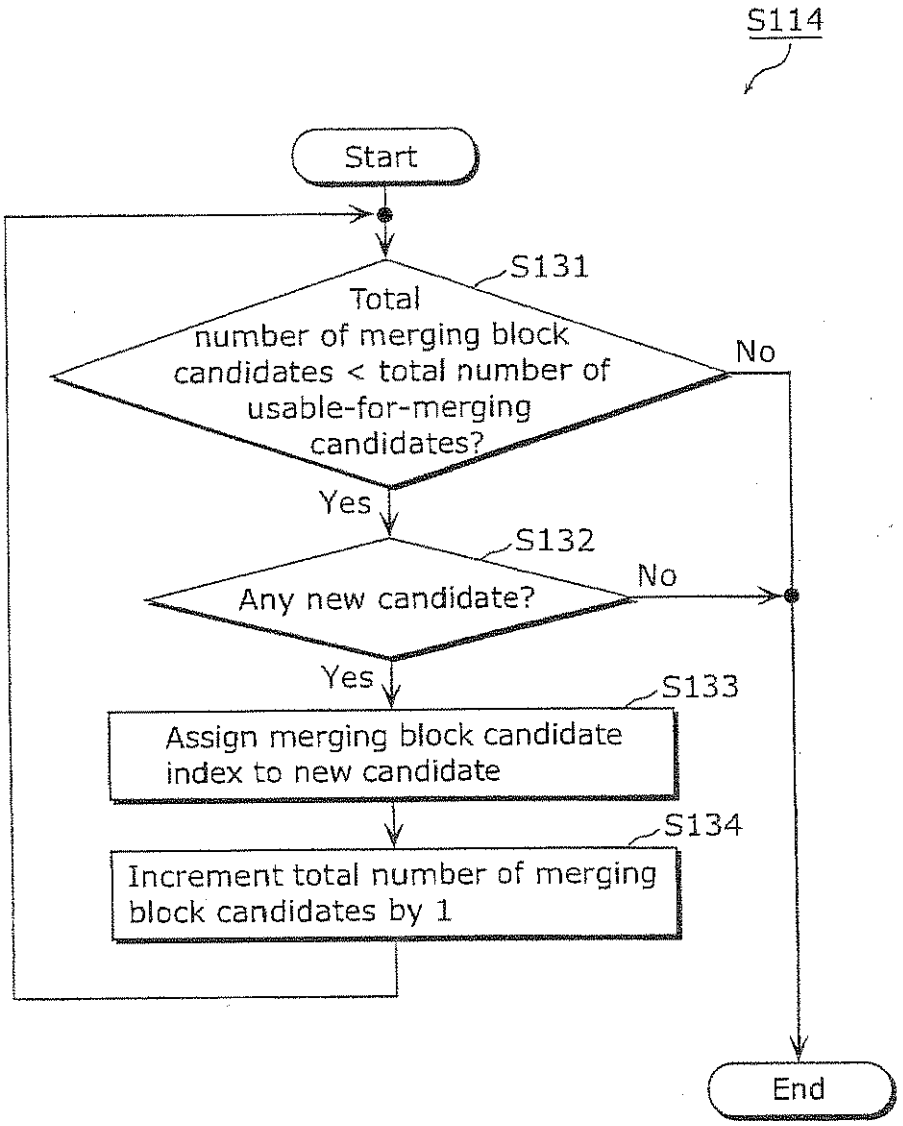

S114
Start
S131
Total number of merging block candidates < total number of usable-for-merging candidates?
No
Yes
S132
Any new candidate?
No
Yes
S133
Assign merging block candidate index to new candidate
S134
Increment total number of merging block candidates by 1
End

FIG. 17

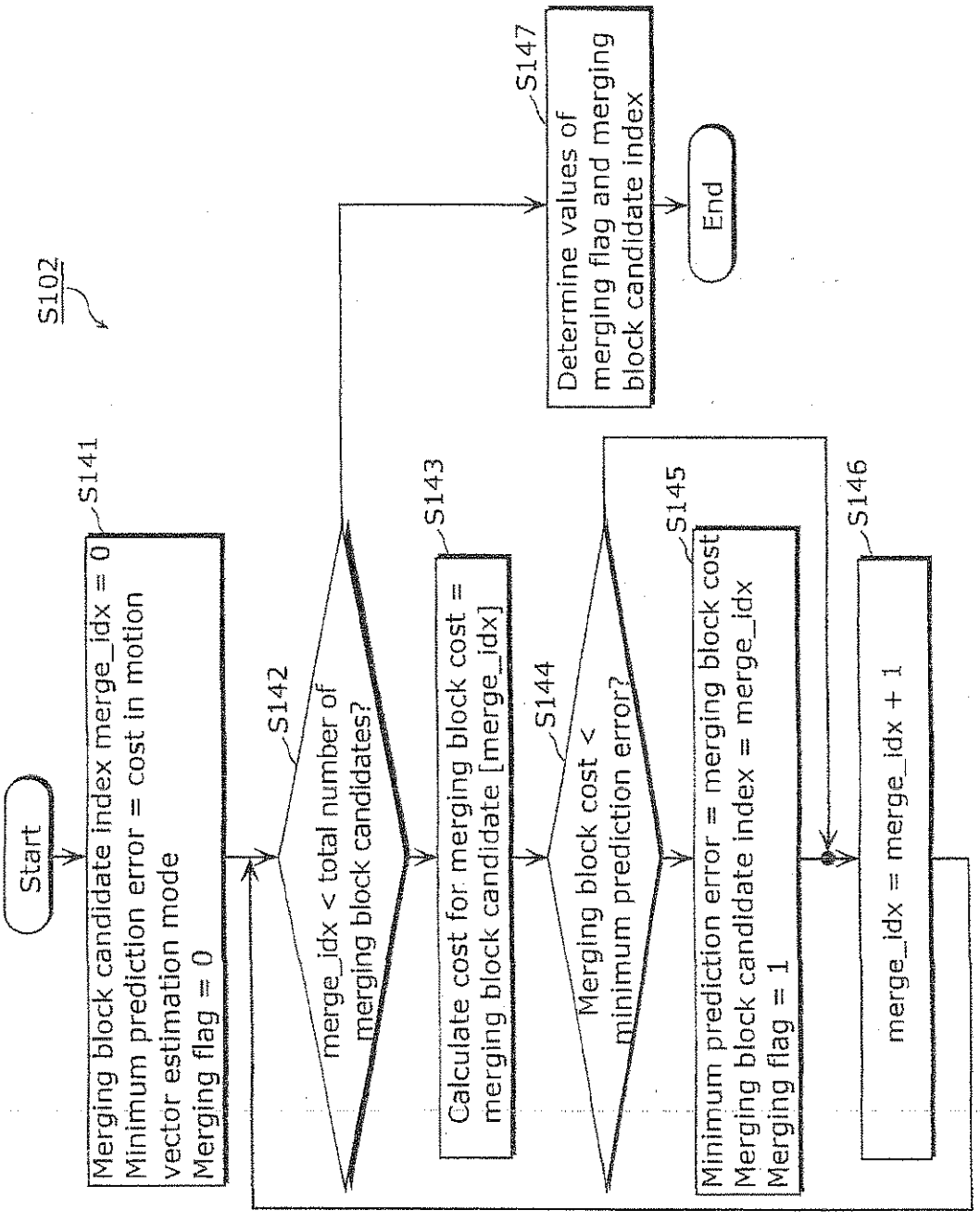

S102
Start
S141
Merging block candidate index merge_idx = 0
Minimum prediction error = cost in motion vector estimation mode
Merging flag = 0
S142
merge_idx < total number of merging block candidates?
S143
Calculate cost for merging block cost = merging block candidate [merge_idx]
S144
Merging block cost < minimum prediction error?
S145
Minimum prediction error = merging block cost
Merging block candidate index = merge_idx
Merging flag = 1
S146
merge_idx = merge_idx + 1
S147
Determine values of merging flag and merging block candidate index
End

FIG. 18

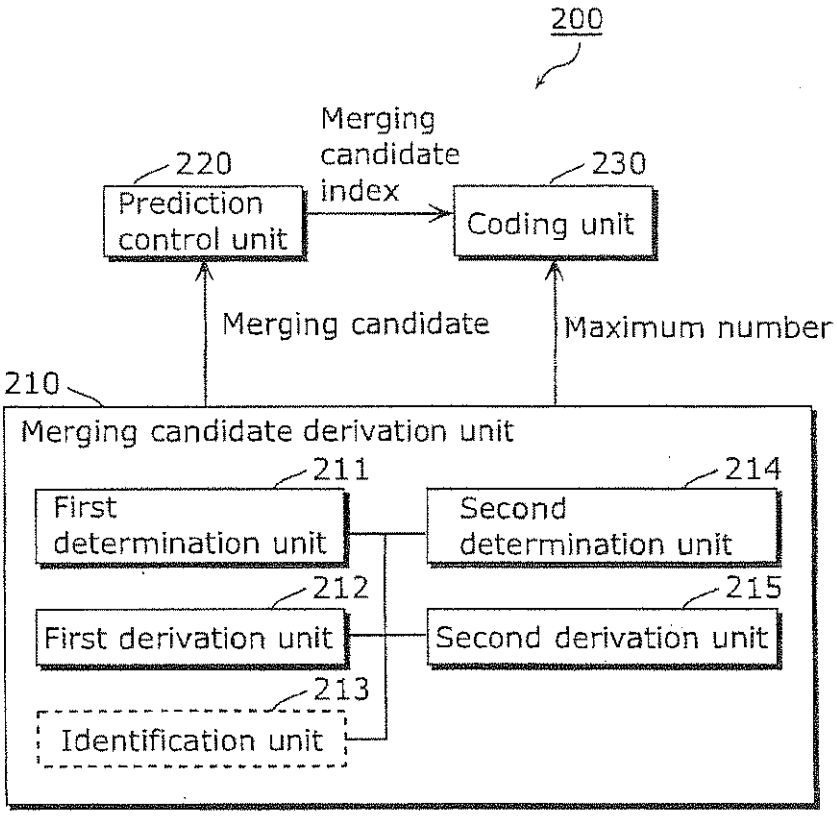

200
Merging candidate index
220
Prediction control unit
230
Coding unit
Merging candidate
Maximum number
210
Merging candidate derivation unit
211
First determination unit
214
Second determination unit
212
First derivation unit
215
Second derivation unit
213
Identification unit

FIG. 19

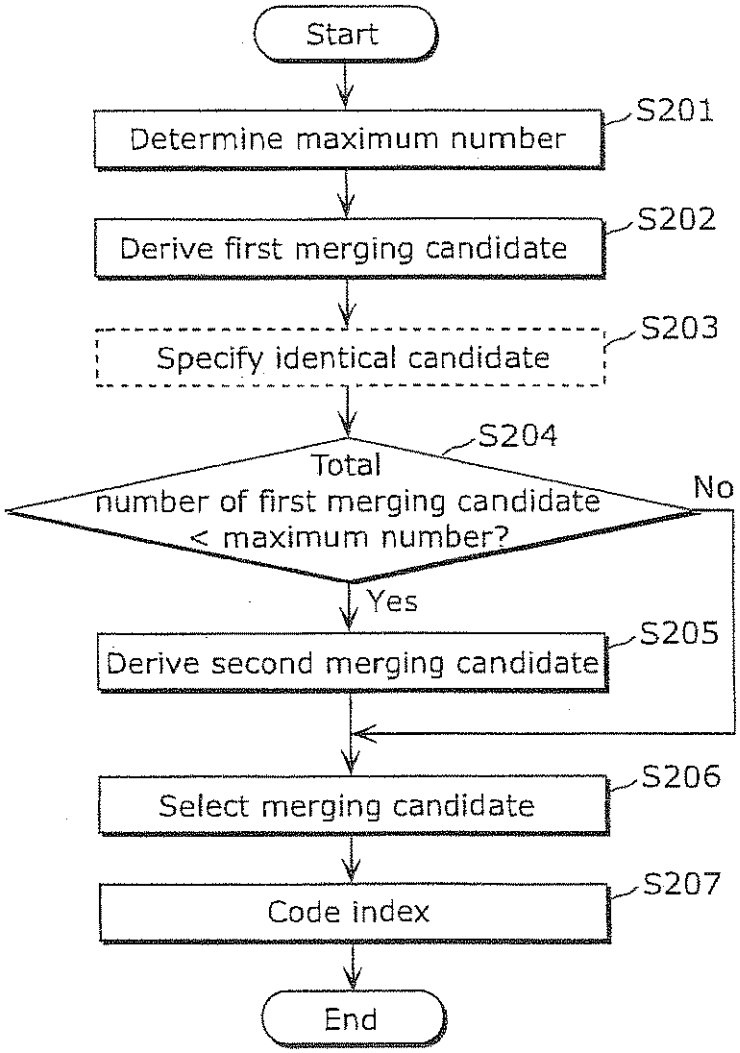

Start
Determine maximum number
S201
Derive first merging candidate
S202
Specify identical candidate
S203
S204
Total number of first merging candidate < maximum number?
No
Yes
Derive second merging candidate
S205
Select merging candidate
S206
Code index
S207
End

FIG. 20

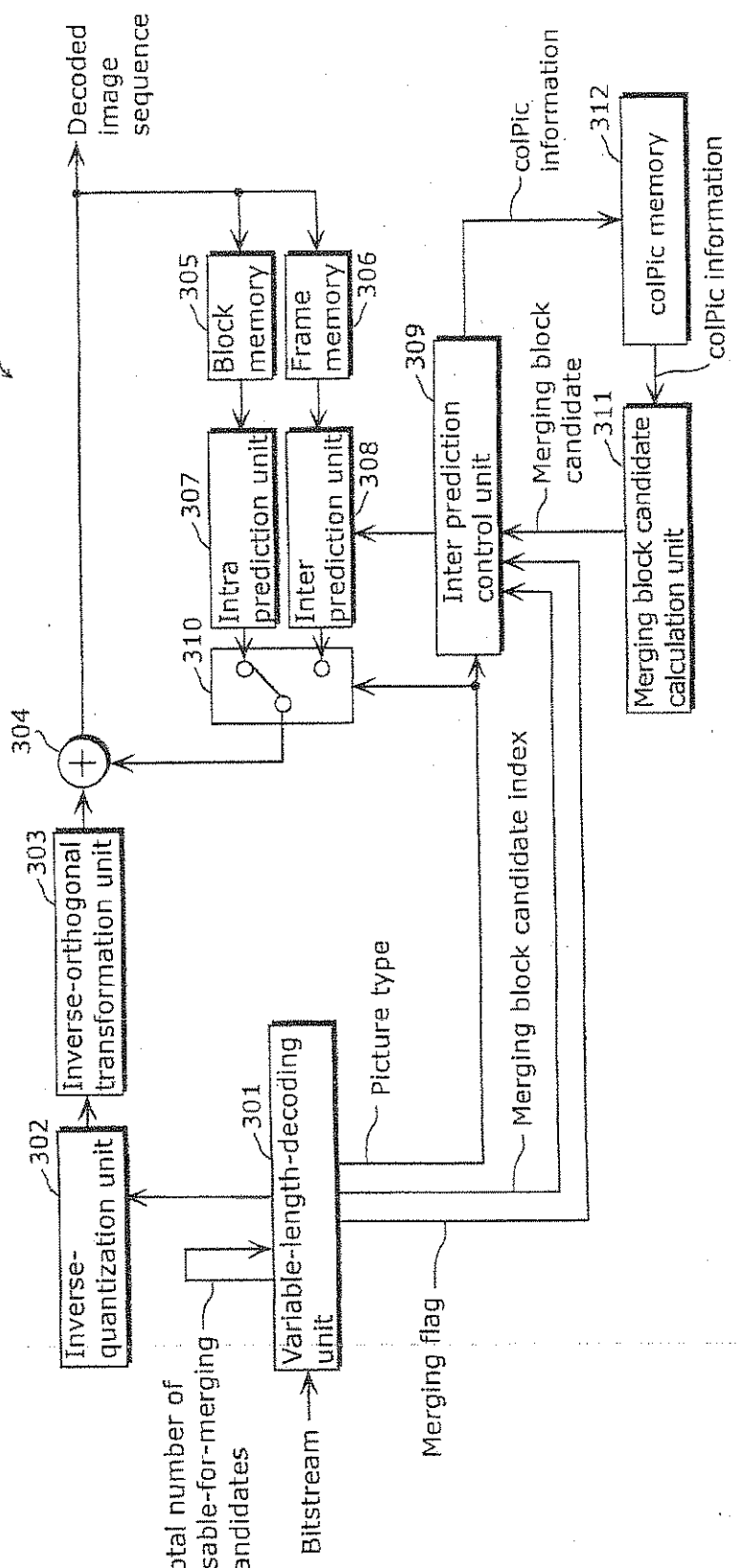

300
Decoded image sequence
305
Block memory
306
Frame memory
307
Intra prediction unit
308
Inter prediction unit
310
304
303
Inverse-orthogonal transformation unit
302
Inverse-quantization unit
301
Variable-length-decoding unit
Total number of usable-for-merging candidates
Bitstream
Picture type
Merging flag
Merging block candidate index
309
Inter prediction control unit
Merging block candidate
311
Merging block candidate calculation unit
colPic information
312
colPic memory
colPic information

FIG. 21

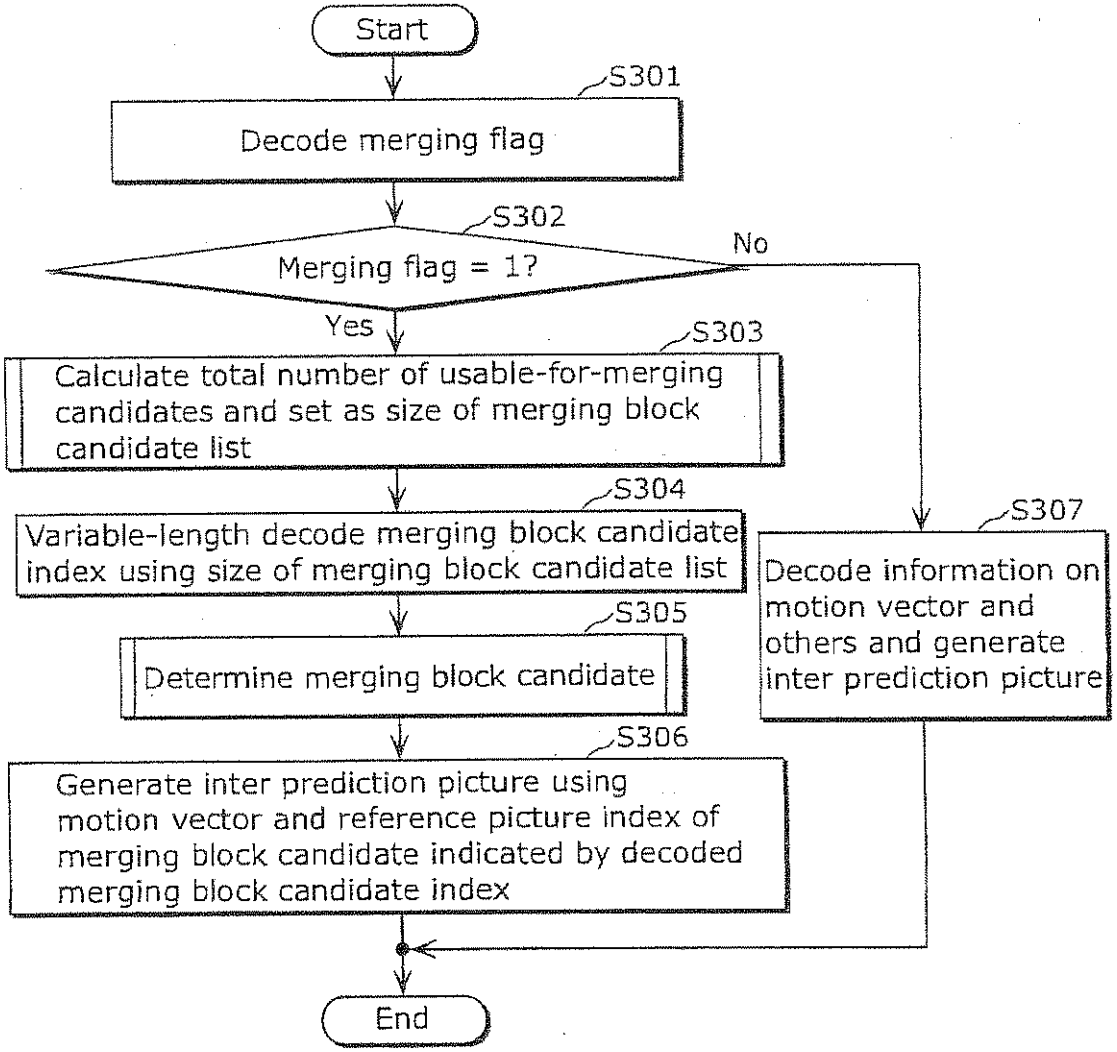

Start
S301
Decode merging flag
S302
Merging flag = 1?
No
Yes
S303
Calculate total number of usable-for-merging candidates and set as size of merging block candidate list
S304
Variable-length decode merging block candidate index using size of merging block candidate list
S305
Determine merging block candidate
S306
Generate inter prediction picture using motion vector and reference picture index of merging block candidate indicated by decoded merging block candidate index
S307
Decode information on motion vector and others and generate inter prediction picture
End

FIG. 22

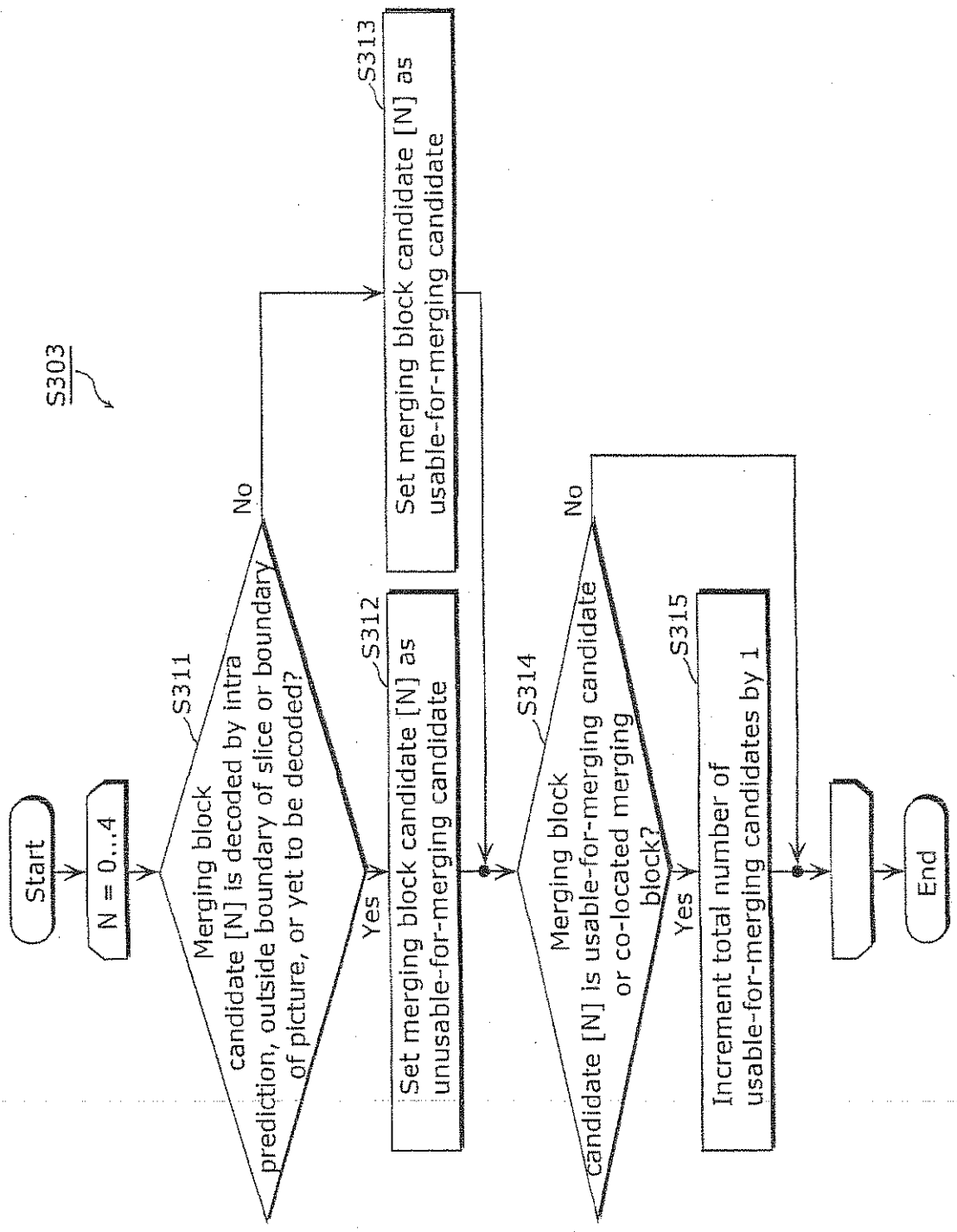

S303
Start
N = 0...4
S311
Merging block candidate [N] is decoded by intra prediction, outside boundary of slice or boundary of picture, or yet to be decoded?
No
Yes
S312
Set merging block candidate [N] as unusable-for-merging candidate
S313
Set merging block candidate [N] as usable-for-merging candidate
S314
Merging block candidate [N] is usable-for-merging candidate or co-located merging block?
No
Yes
S315
Increment total number of usable-for-merging candidates by 1
End

FIG. 23

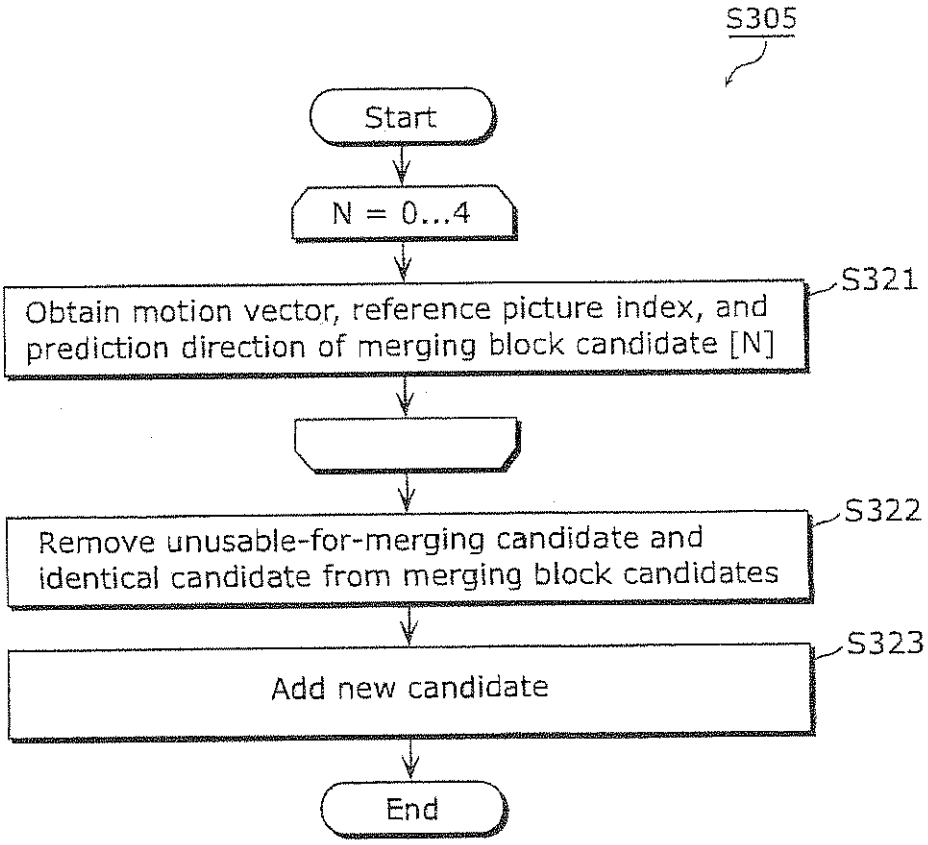

S305
Start
N = 0...4
Obtain motion vector, reference picture index, and prediction direction of merging block candidate [N]
S321
Remove unusable-for-merging candidate and identical candidate from merging block candidates
S322
Add new candidate
S323
End

FIG. 24

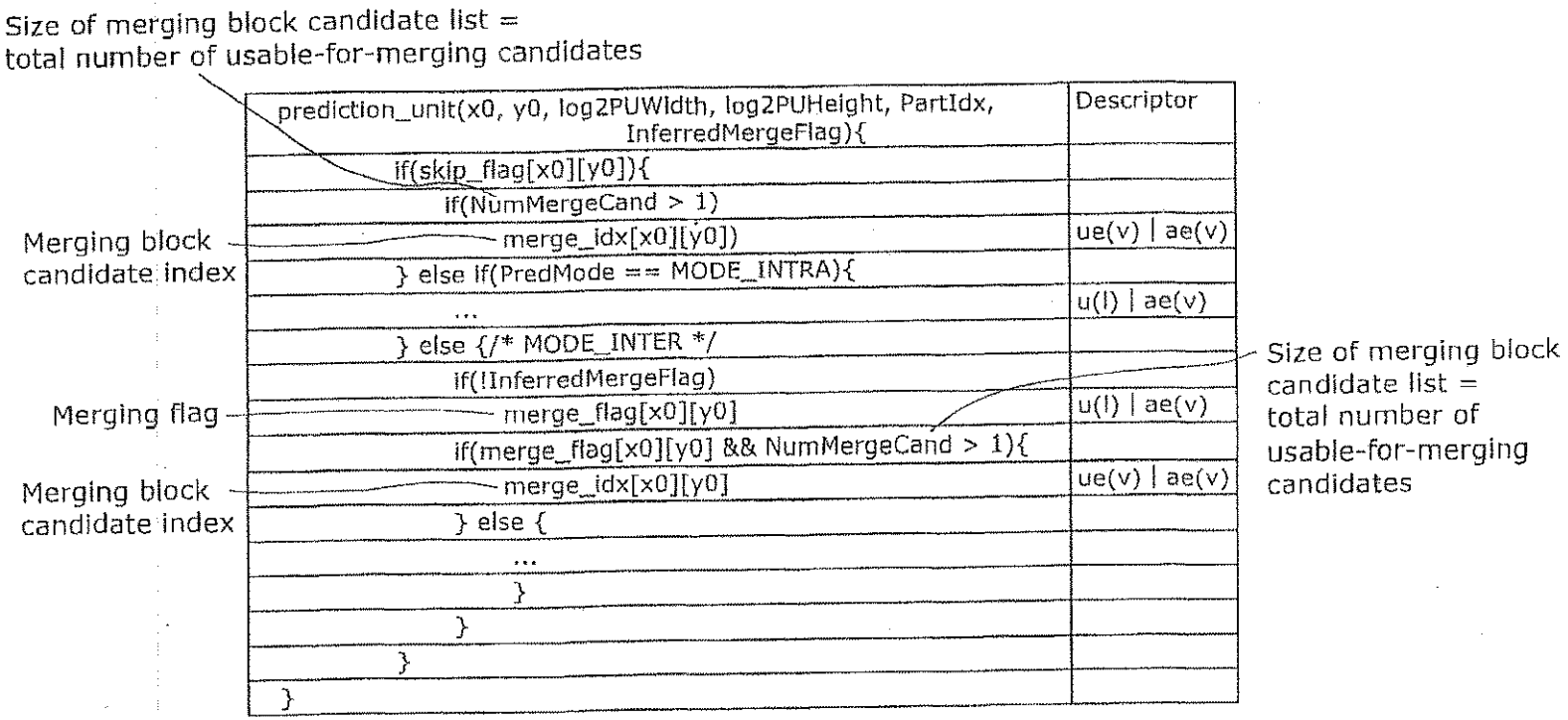

Size of merging block candidate list = total number of usable-for-merging candidates
Merging block candidate index
Merging flag
Merging block candidate index
Size of merging block candidate list = total number of usable-for-merging candidates
prediction_unit(x0, y0, log2PUWidth, log2PUHeight, PartIdx, InferredMergeFlag){
if(skip_flag[x0][y0]){
if(NumMergeCand > 1)
merge_idx[x0][y0])
} else if(PredMode == MODE_INTRA){
...
} else {/* MODE_INTER */
if(!InferredMergeFlag)
merge_flag[x0][y0]
if(merge_flag[x0][y0] && NumMergeCand > 1){
merge_idx[x0][y0]
} else {
...
}
}
}
}
Descriptor
ue(v) | ae(v)
u(l) | ae(v)
u(l) | ae(v)
ue(v) | ae(v)

FIG. 25

	prediction_unit(x0, y0, log2PUWidth, log2PUHeight, PartIdx, InferredMergeFlag){	Descriptor
	if(skip_flag[x0][y0]){	
Merging block candidate index	merge_idx[x0][y0]	ue(v) | ae(v)
	} else if(PredMode == MODE_INTRA){	
	...	u(l) | ae(v)
	} else {/* MODE_INTER */	
	if(!InferredMergeFlag)	
Merging flag	merge_flag[x0][y0]	u(l) | ae(v)
	if(merge_flag[x0][y0]){	
Merging block candidate index	merge_idx[x0][y0]	ue(v) | ae(v)
	} else {	
	...	
	}	
	}	
	}	
	}	

FIG. 26

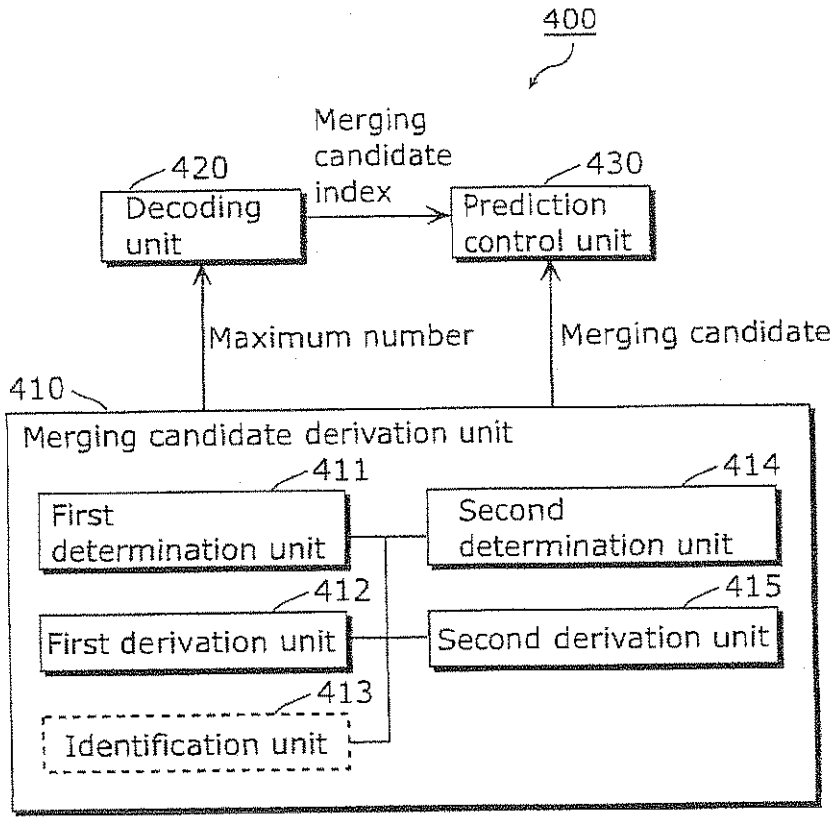

400
Merging candidate index
420
Decoding unit
430
Prediction control unit
Maximum number
Merging candidate
410
Merging candidate derivation unit
411
First determination unit
414
Second determination unit
412
First derivation unit
415
Second derivation unit
413
Identification unit

FIG. 27

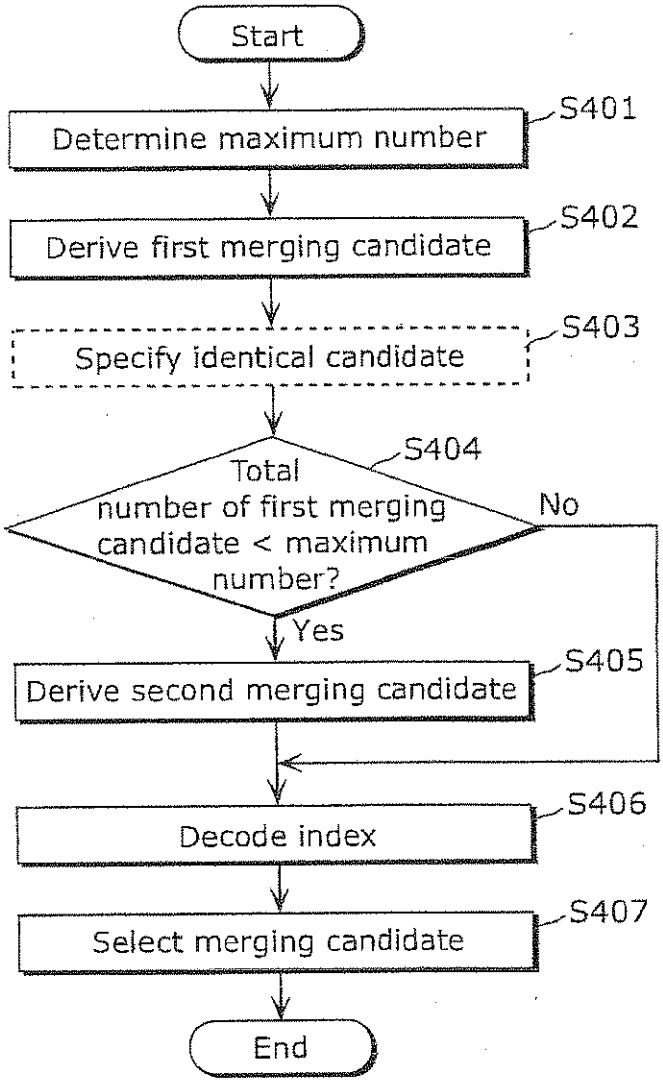

Start
Determine maximum number
S401
Derive first merging candidate
S402
Specify identical candidate
S403
Total number of first merging candidate < maximum number?
S404
No
Yes
Derive second merging candidate
S405
Decode index
S406
Select merging candidate
S407
End

FIG. 28

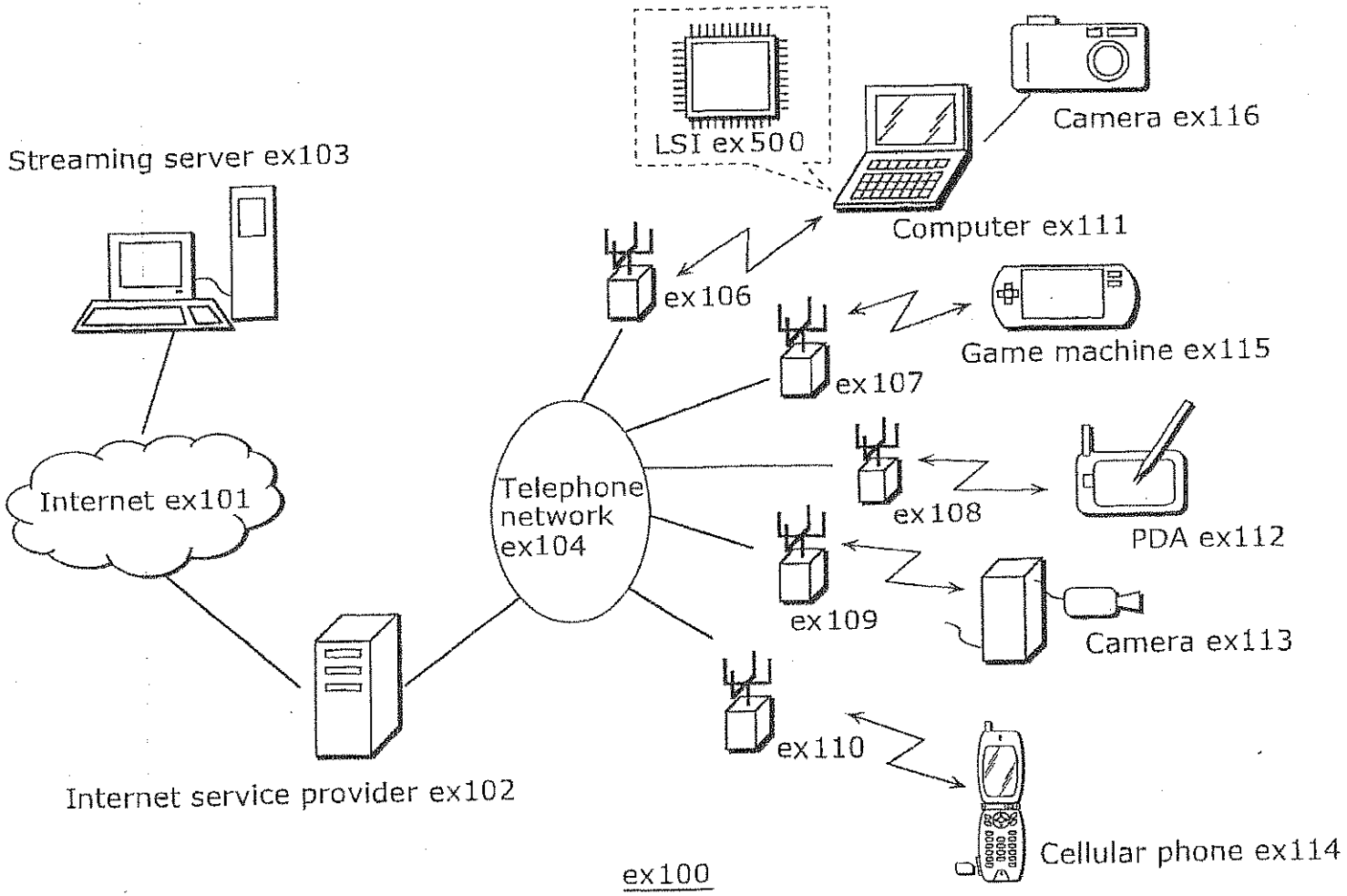

LSI ex500
Camera ex116
Streaming server ex103
Computer ex111
ex106
ex107
Game machine ex115
Internet ex101
Telephone network ex104
ex108
PDA ex112
ex109
Camera ex113
ex110
Internet service provider ex102
Cellular phone ex114
ex100

FIG. 29

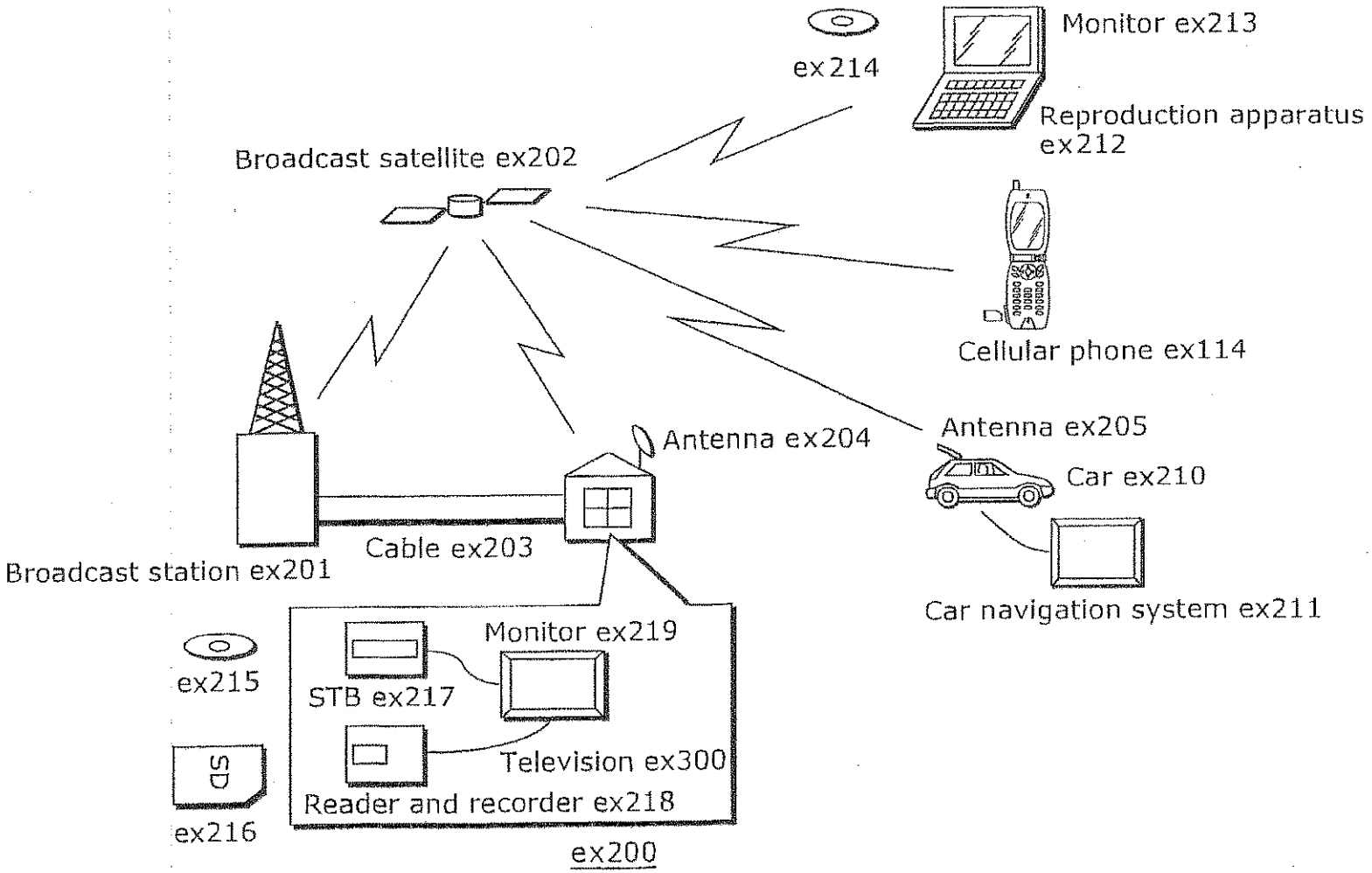

ex214
Monitor ex213
Reproduction apparatus ex212
Broadcast satellite ex202
Cellular phone ex114
Antenna ex204
Antenna ex205
Car ex210
Broadcast station ex201
Cable ex203
Car navigation system ex211
Monitor ex219
ex215
STB ex217
Television ex300
SD
Reader and recorder ex218
ex216
ex200

FIG. 30

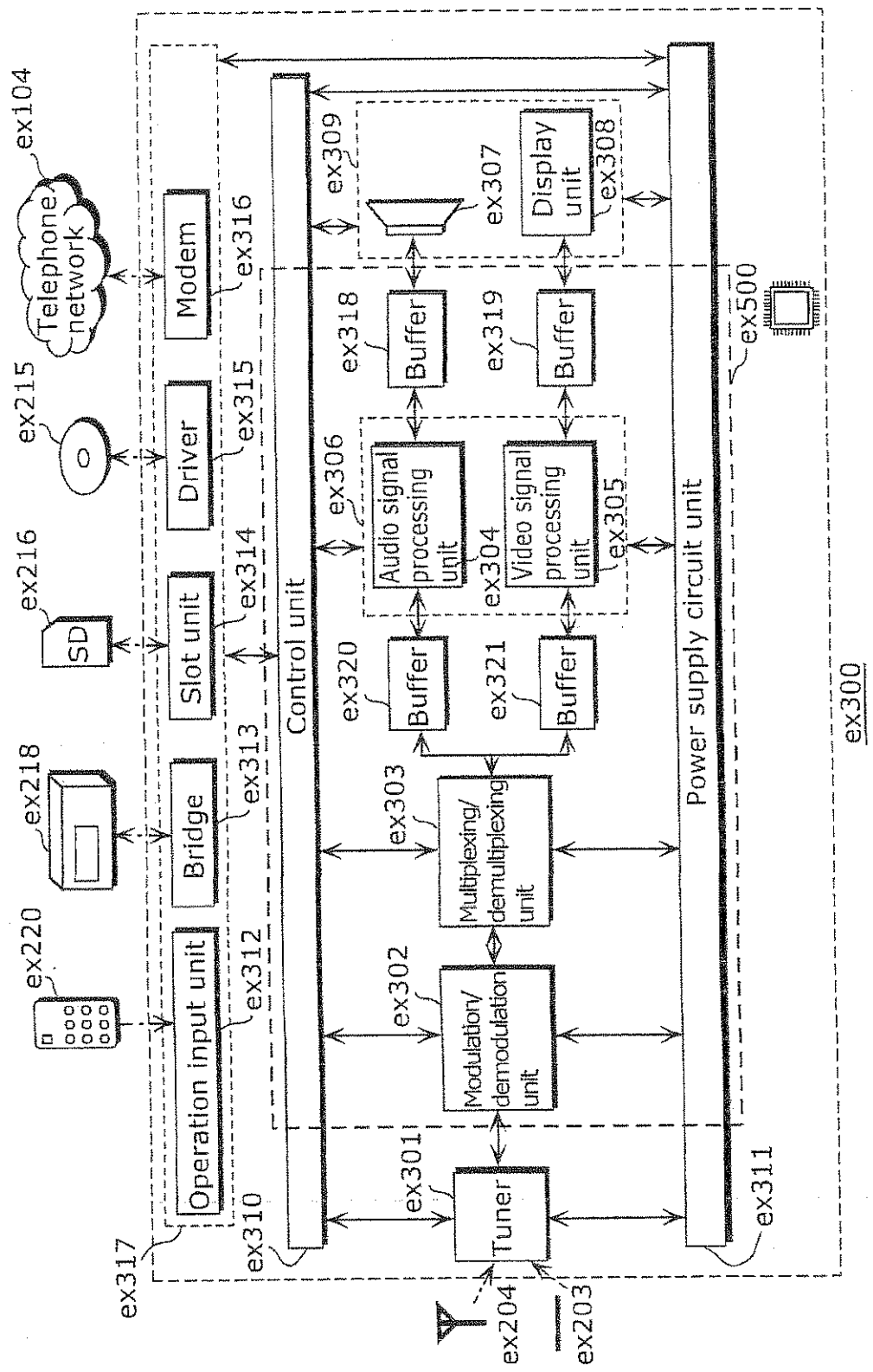

ex104
Telephone network
ex215
ex216
SD
ex218
ex220
ex317
Operation input unit
ex312
Bridge
ex313
Slot unit
ex314
Driver
ex315
Modem
ex316
Control unit
ex310
ex301
Tuner
ex302
Modulation/ demodulation unit
ex303
Multiplexing/ demultiplexing unit
ex320
Buffer
ex321
Buffer
ex304
ex306
Audio signal processing unit
Video signal processing unit
ex305
ex318
Buffer
ex319
Buffer
ex307
ex309
Display unit
ex308
Power supply circuit unit
ex311
ex500
ex204
ex203
ex300

FIG. 31

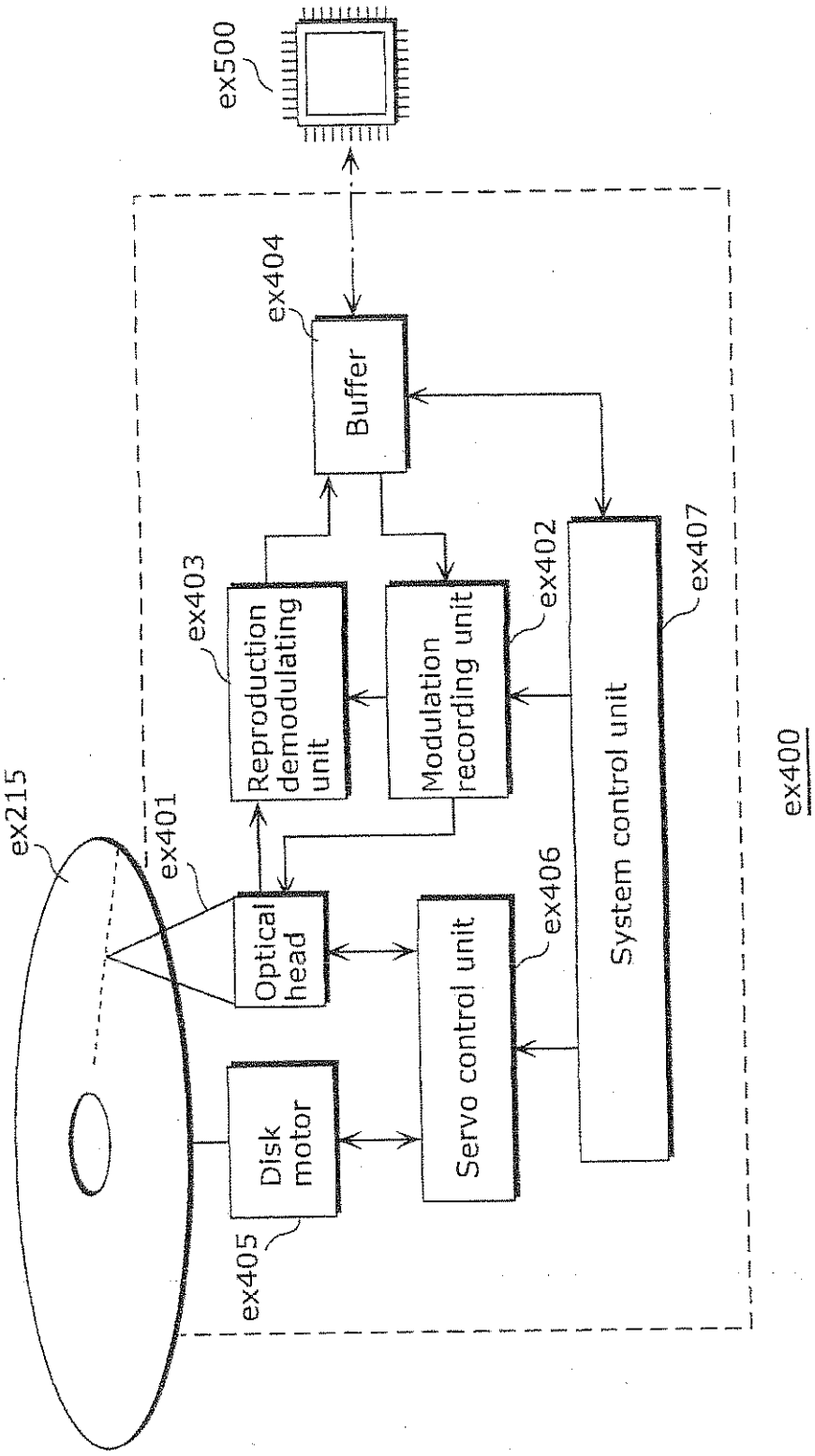

ex500
ex404
Buffer
ex403
Reproduction demodulating unit
ex402
Modulation recording unit
ex407
System control unit
ex400
ex215
ex401
Optical head
ex406
Servo control unit
Disk motor
ex405

FIG. 32

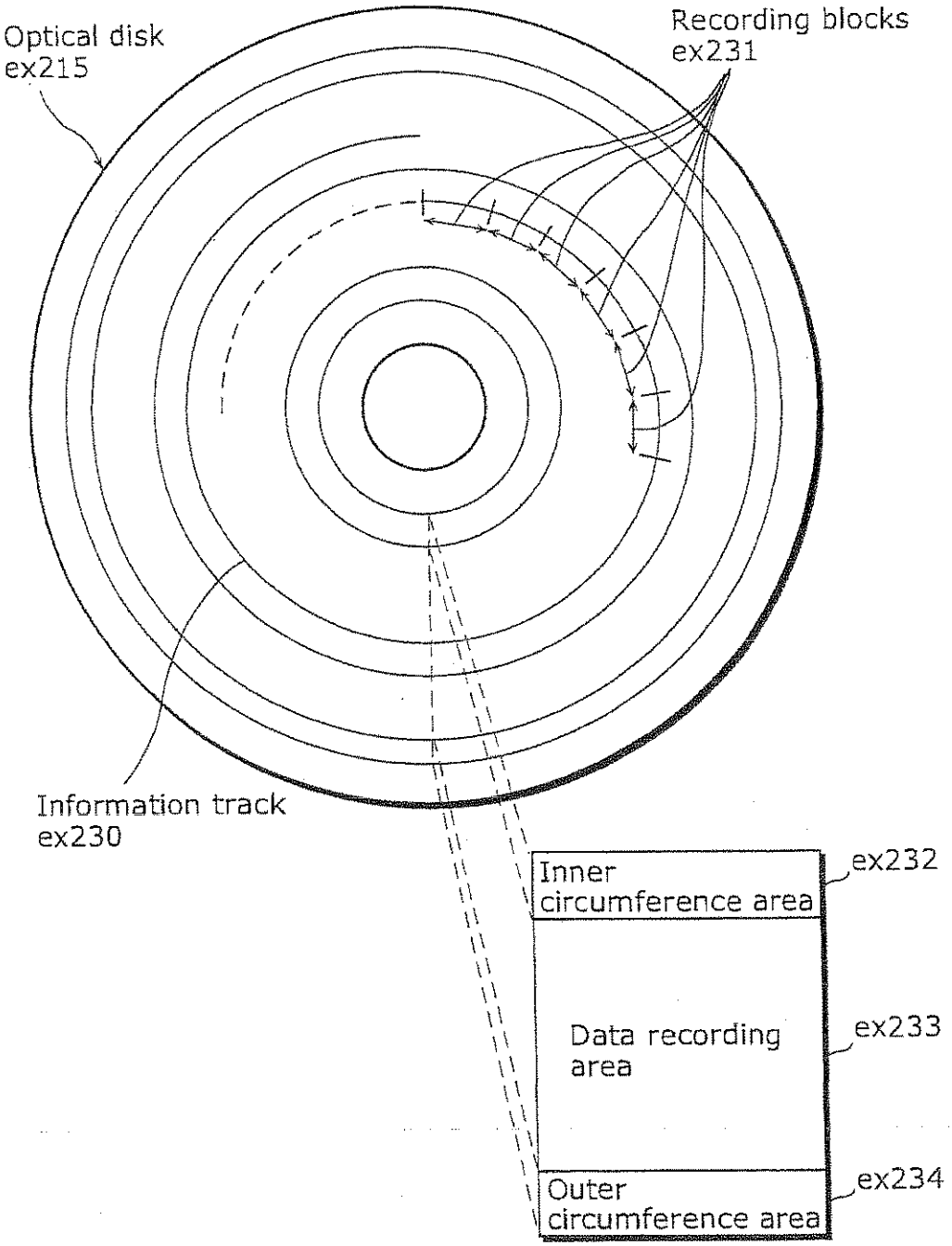

Optical disk
ex215
Recording blocks
ex231
Information track
ex230
Inner
circumference area
ex232
Data recording
area
ex233
Outer
circumference area
ex234

FIG. 33A

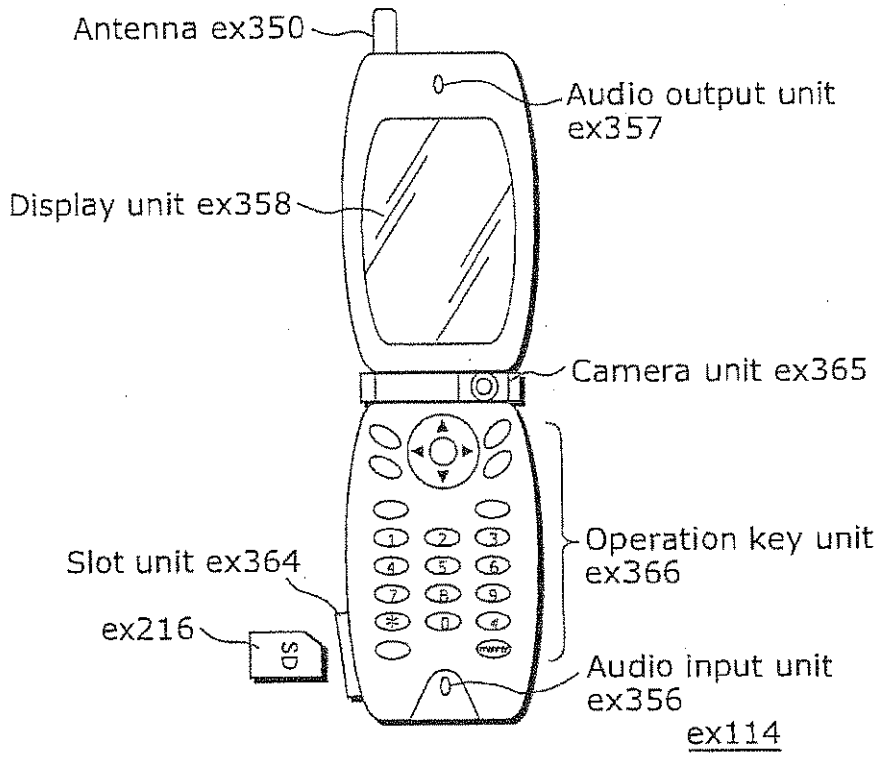


FIG. 33B

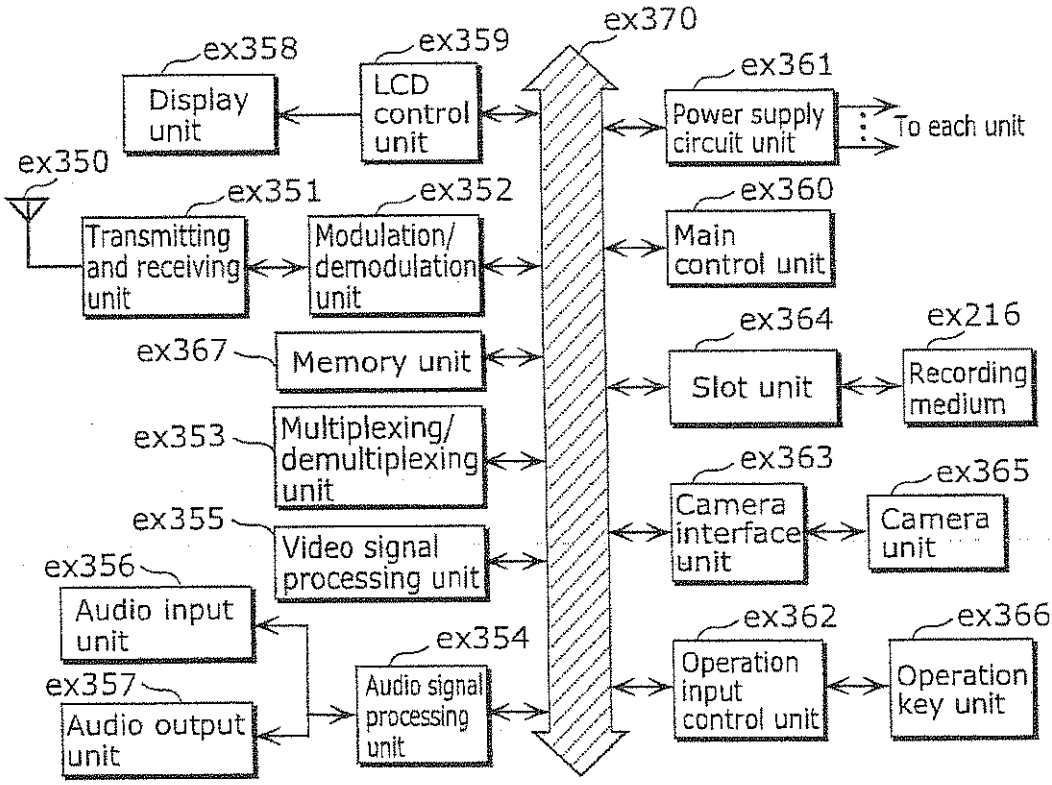


FIG. 34

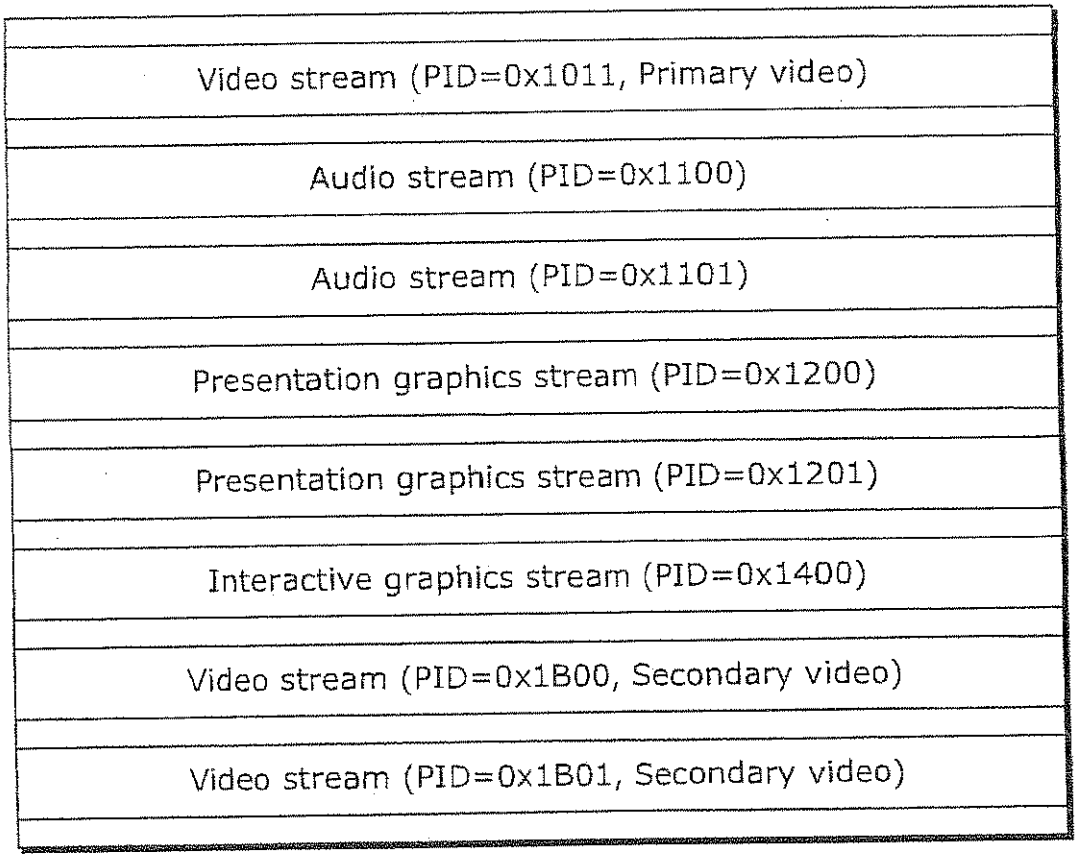

Video stream (PID=0x1011, Primary video)
Audio stream (PID=0x1100)
Audio stream (PID=0x1101)
Presentation graphics stream (PID=0x1200)
Presentation graphics stream (PID=0x1201)
Interactive graphics stream (PID=0x1400)
Video stream (PID=0x1B00, Secondary video)
Video stream (PID=0x1B01, Secondary video)

FIG. 35

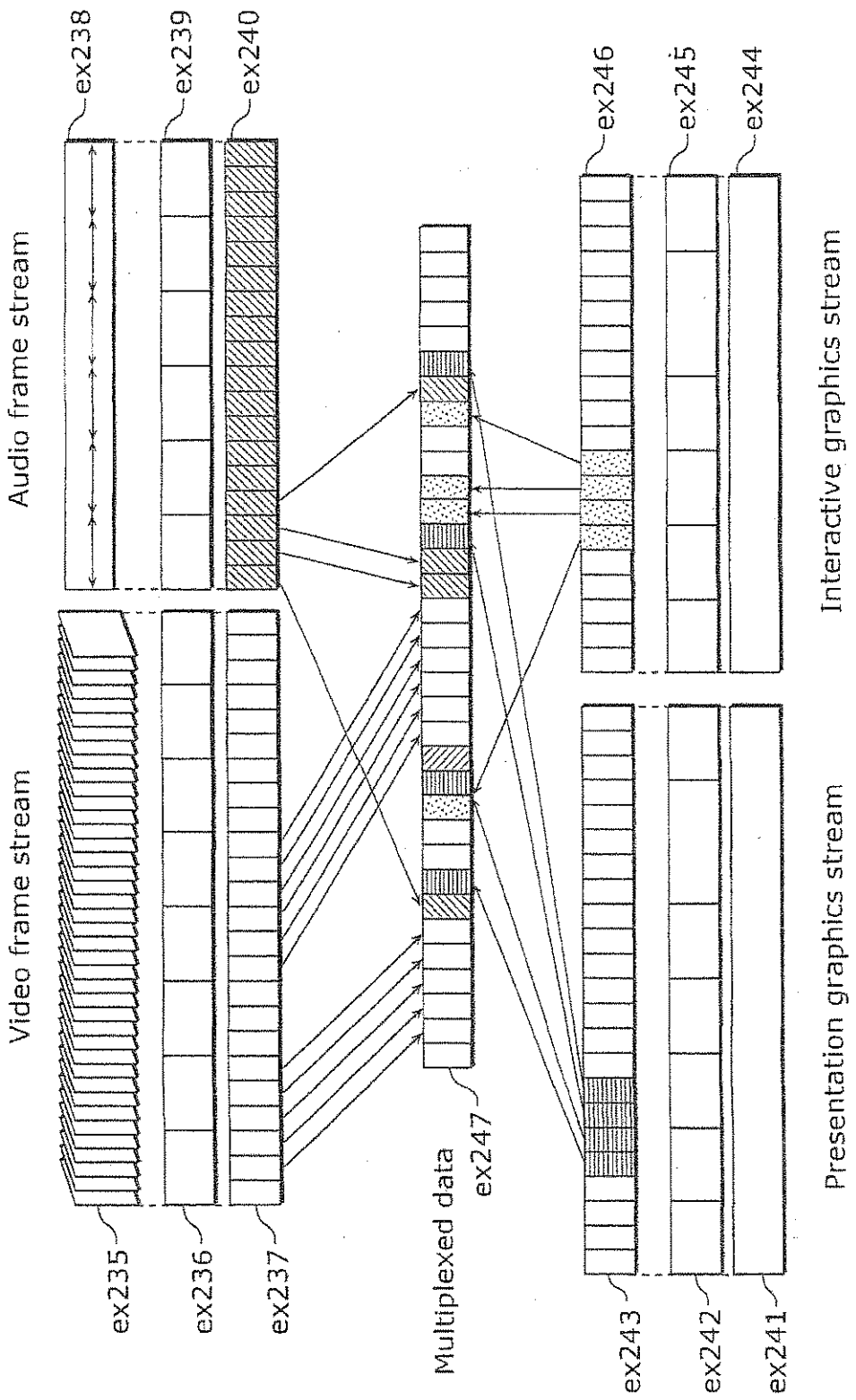

Video frame stream
Audio frame stream
Multiplexed data
Presentation graphics stream
Interactive graphics stream
ex235
ex236
ex237
ex238
ex239
ex240
ex241
ex242
ex243
ex244
ex245
ex246
ex247

FIG. 36

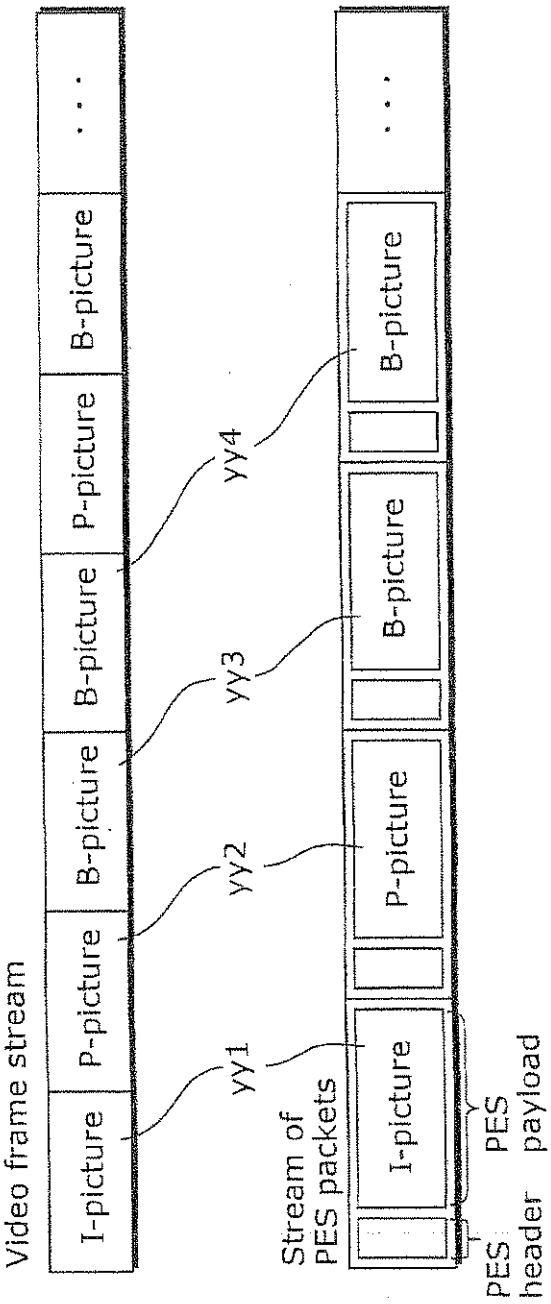

Video frame stream
I-picture
P-picture
B-picture
B-picture
P-picture
B-picture
. . .
yy1
yy2
yy3
yy4
Stream of PES packets
I-picture
P-picture
B-picture
B-picture
. . .
PES header
PES payload

FIG. 37

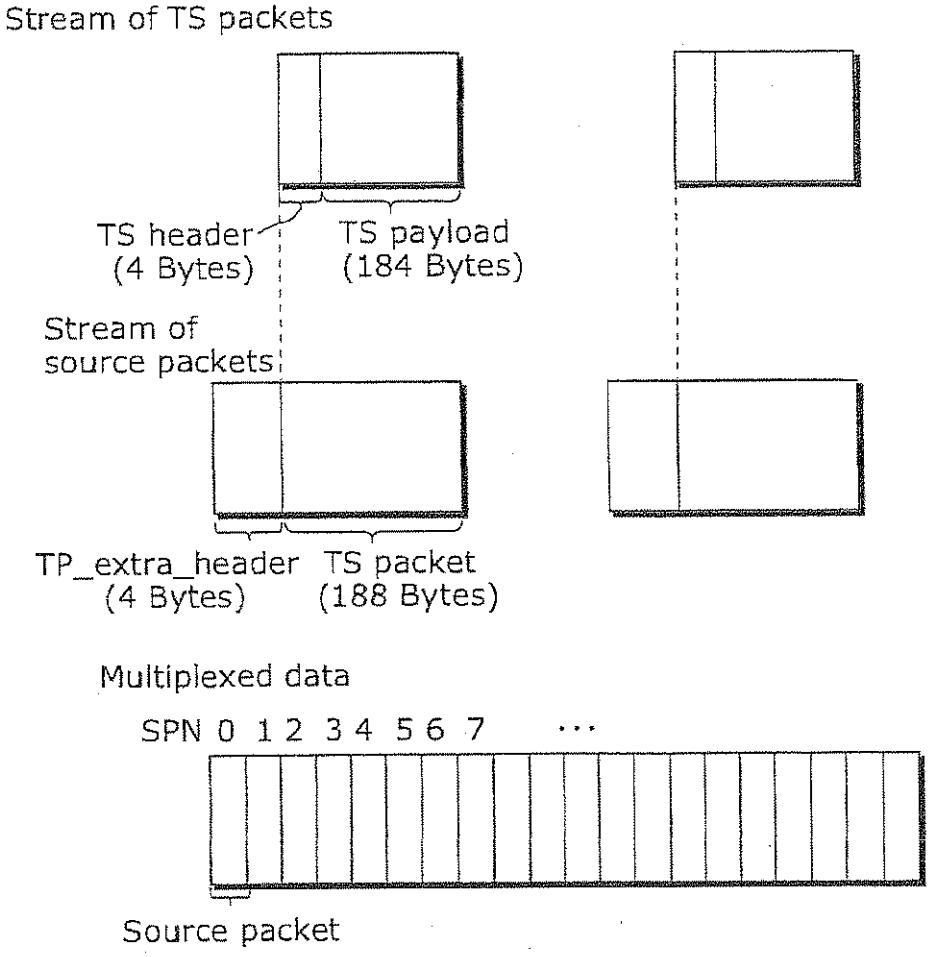

Stream of TS packets
TS header
(4 Bytes)
TS payload
(184 Bytes)
Stream of
source packets
TP_extra_header
(4 Bytes)
TS packet
(188 Bytes)
Multiplexed data
SPN 0 1 2 3 4 5 6 7 ...
Source packet

FIG. 38

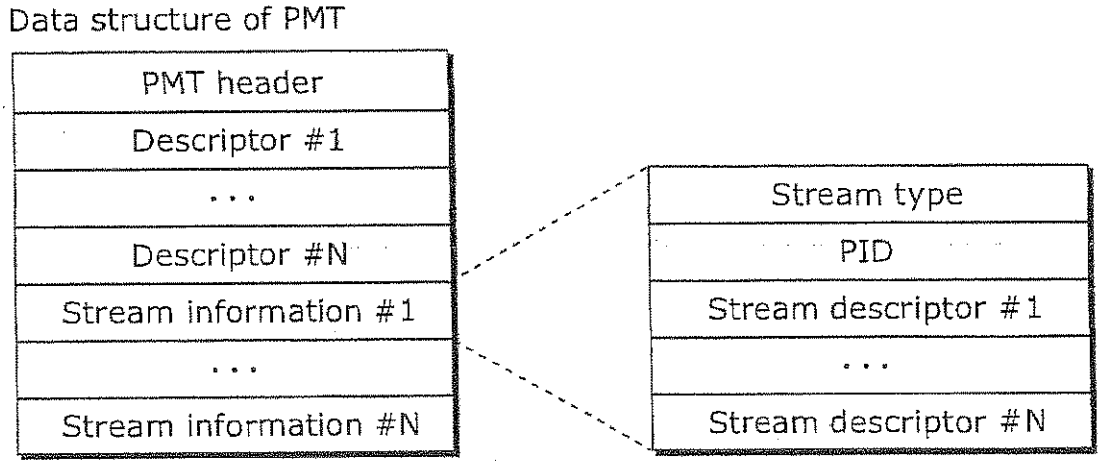

Data structure of PMT
PMT header
Descriptor #1
...
Descriptor #N
Stream information #1
...
Stream information #N
Stream type
PID
Stream descriptor #1
...
Stream descriptor #N

FIG. 39

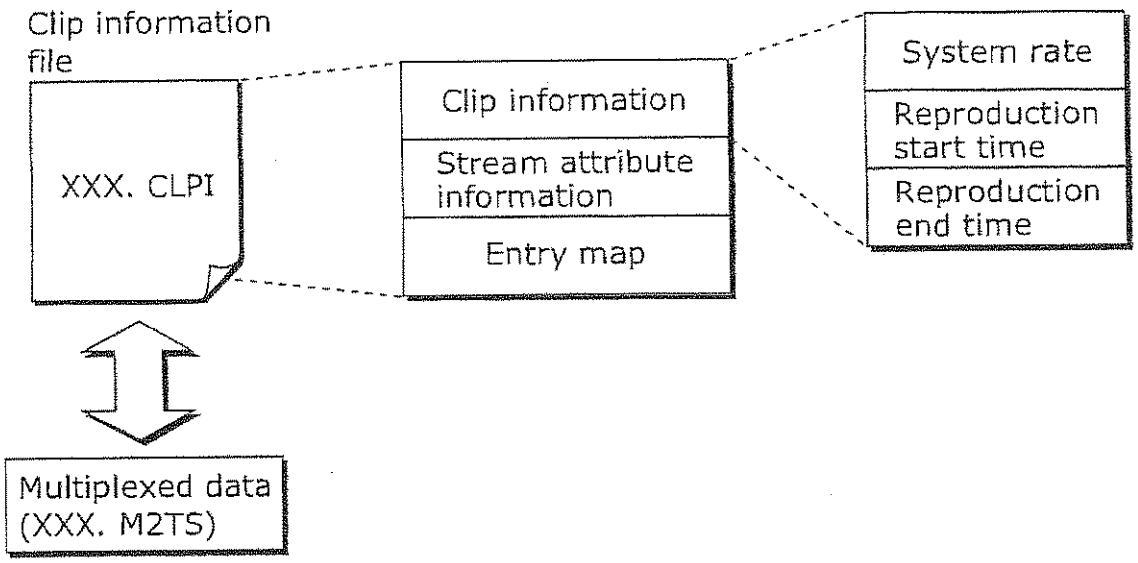

Clip information file
XXX. CLPI
Clip information
Stream attribute information
Entry map
System rate
Reproduction start time
Reproduction end time
Multiplexed data (XXX. M2TS)

FIG. 40

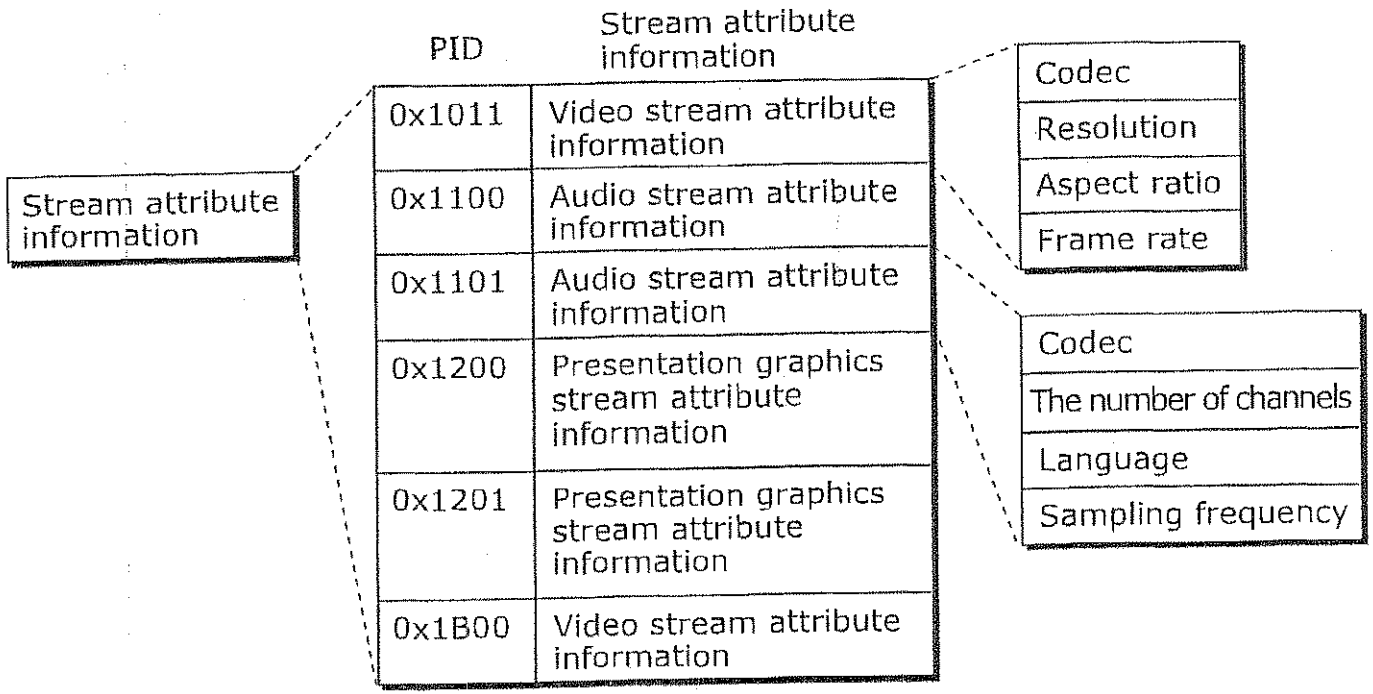

Stream attribute information
PID
Stream attribute information
0x1011
Video stream attribute information
0x1100
Audio stream attribute information
0x1101
Audio stream attribute information
0x1200
Presentation graphics stream attribute information
0x1201
Presentation graphics stream attribute information
0x1B00
Video stream attribute information
Codec
Resolution
Aspect ratio
Frame rate
Codec
The number of channels
Language
Sampling frequency

FIG. 41

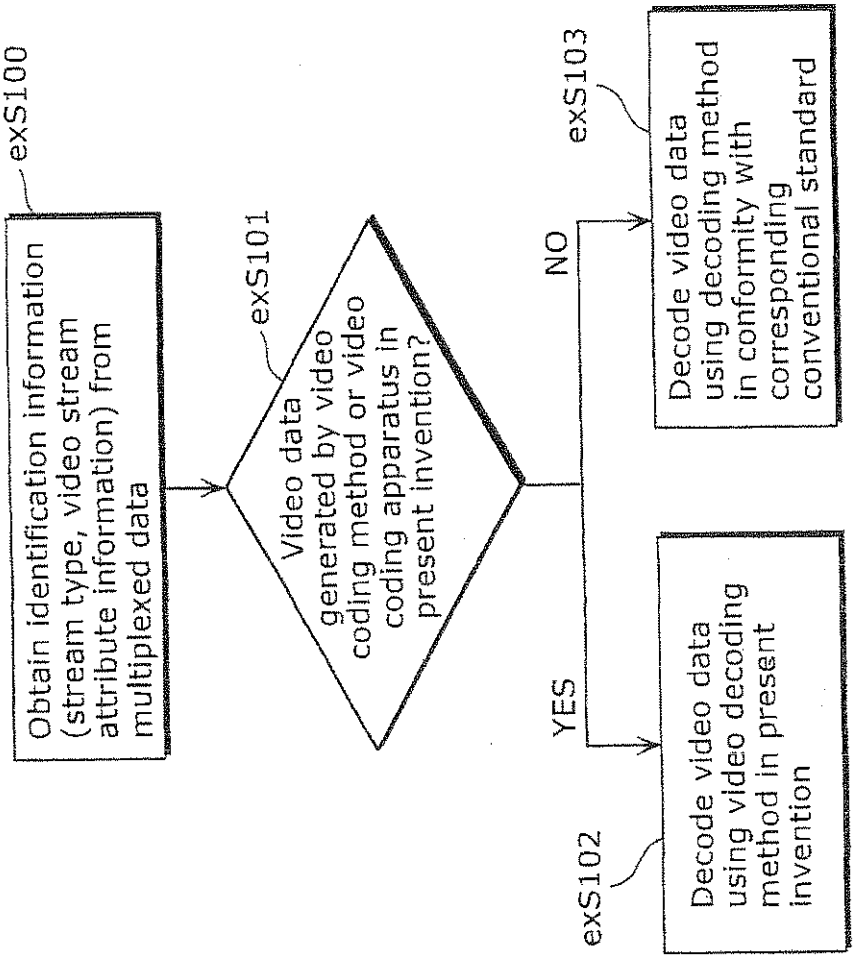

exS100
Obtain identification information (stream type, video stream attribute information) from multiplexed data
exS101
Video data generated by video coding method or video coding apparatus in present invention?
YES
NO
exS102
Decode video data using video decoding method in present invention
exS103
Decode video data using decoding method in conformity with corresponding conventional standard

FIG. 42

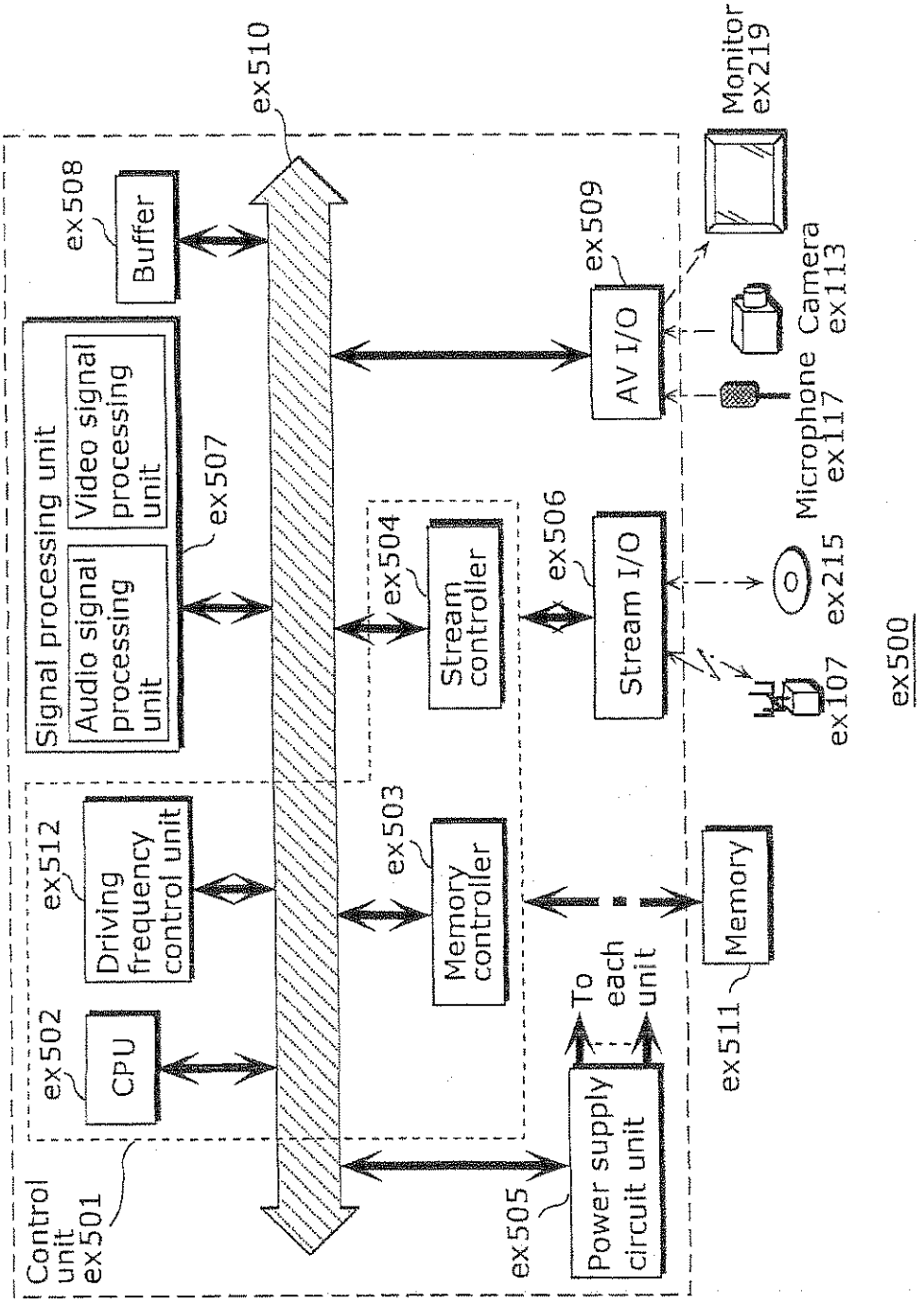
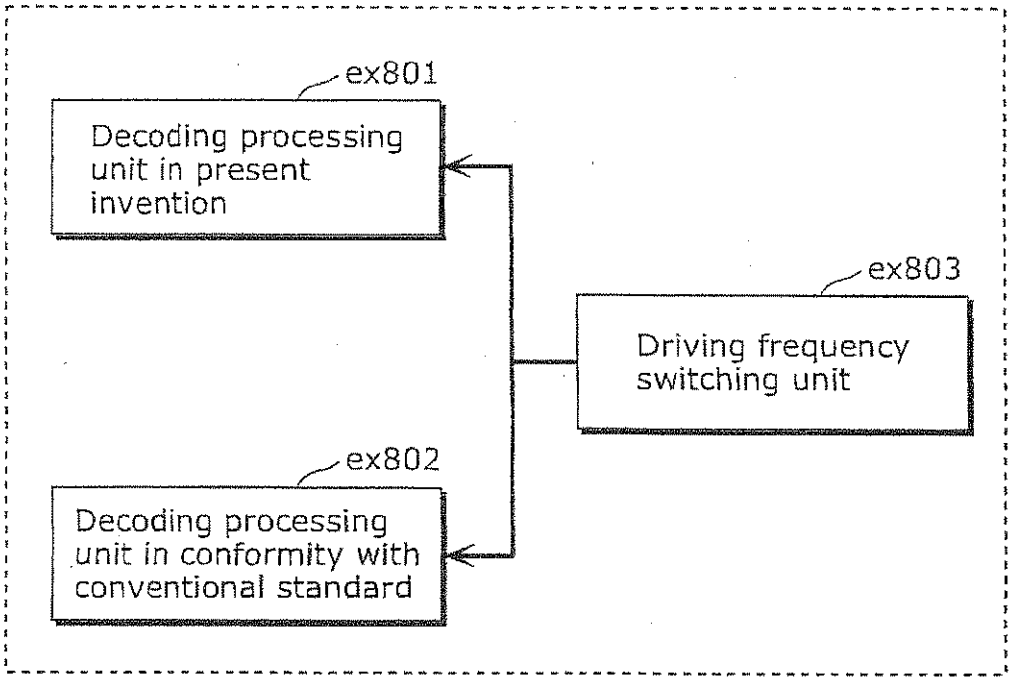

Control unit ex501
CPU ex502
Driving frequency control unit ex512
Signal processing unit
Audio signal processing unit
Video signal processing unit ex507
Buffer ex508
Power supply circuit unit ex505
To each unit
Memory controller ex503
Stream controller ex504
Stream I/O ex506
AV I/O ex509
Memory ex511
ex510
Monitor ex219
Camera ex113
Microphone ex117
ex215
ex107
ex500

FIG. 43


ex801
Decoding processing unit in present invention
ex803
Driving frequency switching unit
ex802
Decoding processing unit in conformity with conventional standard

ex800

FIG. 44

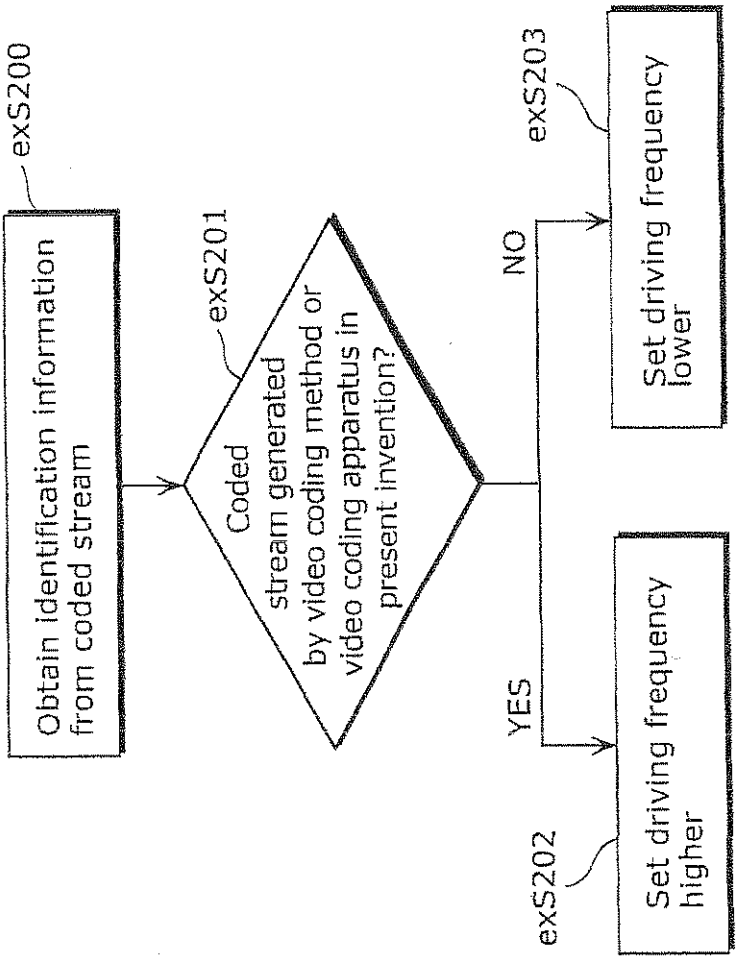

exS200
Obtain identification information from coded stream
exS201
Coded stream generated by video coding method or video coding apparatus in present invention?
YES
NO
exS202
Set driving frequency higher
exS203
Set driving frequency lower

FIG. 45

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

FIG. 46A

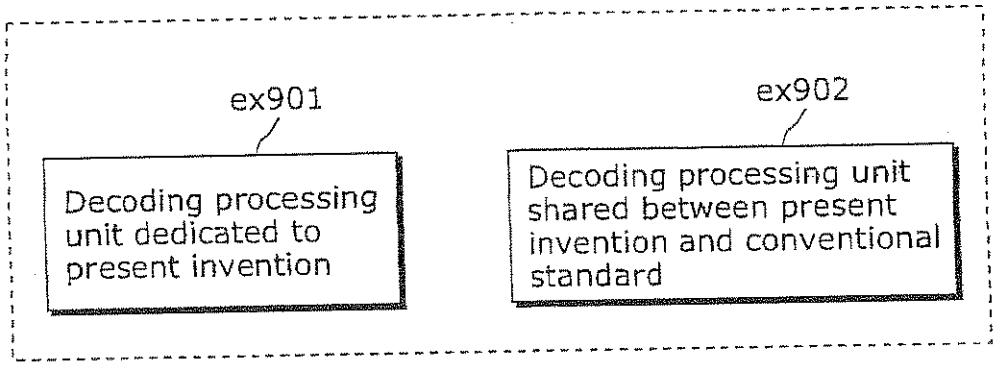


FIG. 46B

