FIG. 1

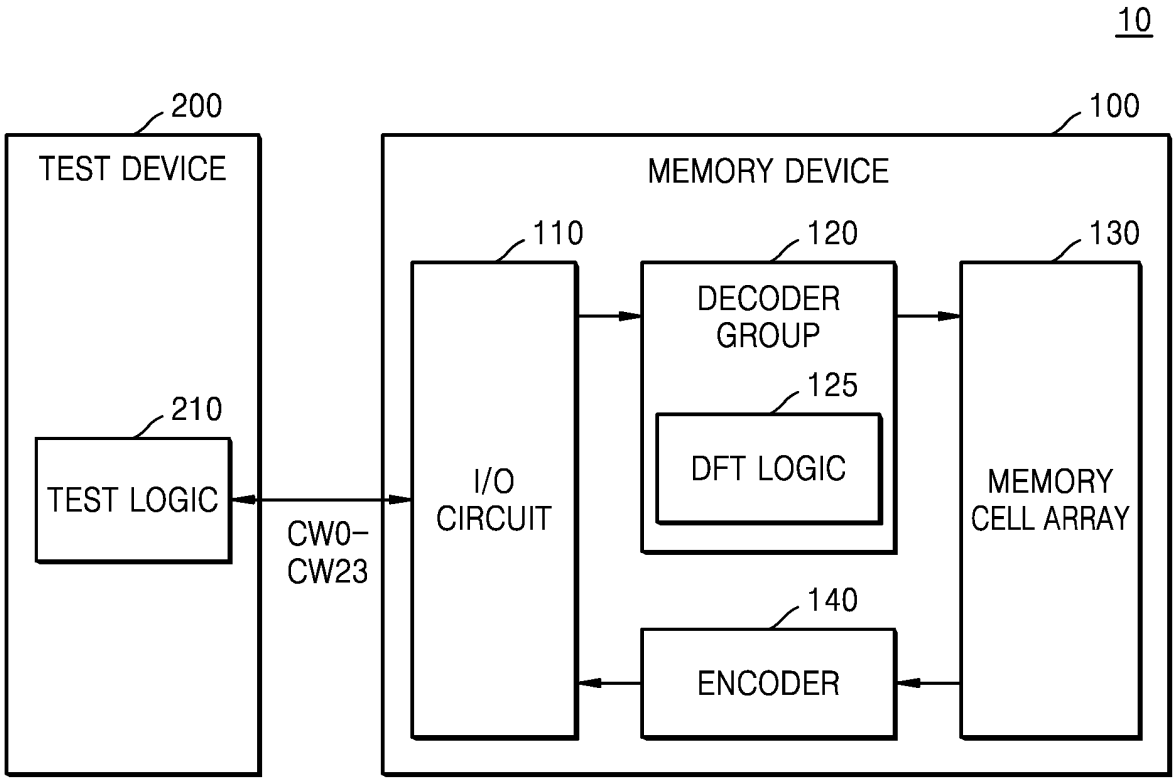

10
200
TEST DEVICE
210
TEST LOGIC
CW0~
CW23
100
MEMORY DEVICE
110
I/O
CIRCUIT
120
DECODER
GROUP
125
DFT LOGIC
130
MEMORY
CELL ARRAY
140
ENCODER

FIG. 2

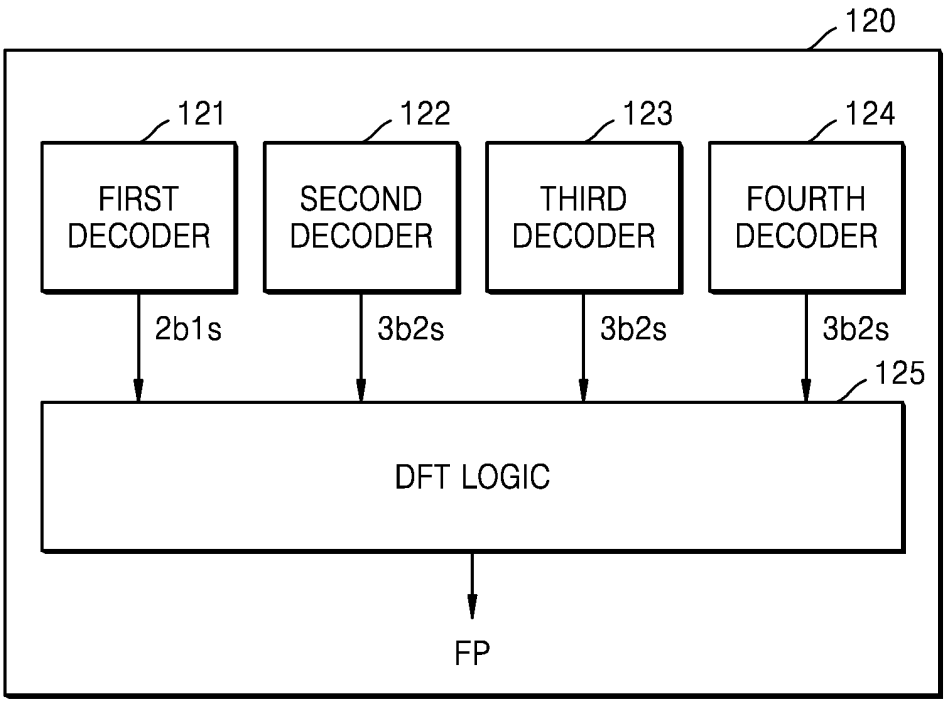

120
121
122
123
124
FIRST DECODER
SECOND DECODER
THIRD DECODER
FOURTH DECODER
2b1s
3b2s
3b2s
3b2s
125
DFT LOGIC
FP

FIG. 3

(a)

2bits		1symbol		
		lsb	msb	
x0	x1	y0	y1	
0	0	0	0	
0	1	0	1	invalid
1	0	1	0	
1	1	1	1	

(b)

3bits			2symbols				
			y0	y1	y2	y3	
x0	x1	x2	lsb	msb	lsb	msb	
0	0	0	0	0	1	0	
0	0	1	0	0	0	0	
0	1	0	0	0	1	1	
0	1	1	1	0	1	1	
			1	**0**	**1**	**0**	unused
1	0	0	1	1	1	0	
1	0	1	1	0	0	0	
1	1	0	1	1	1	0	
1	1	1	1	1	1	1	

FIG. 4

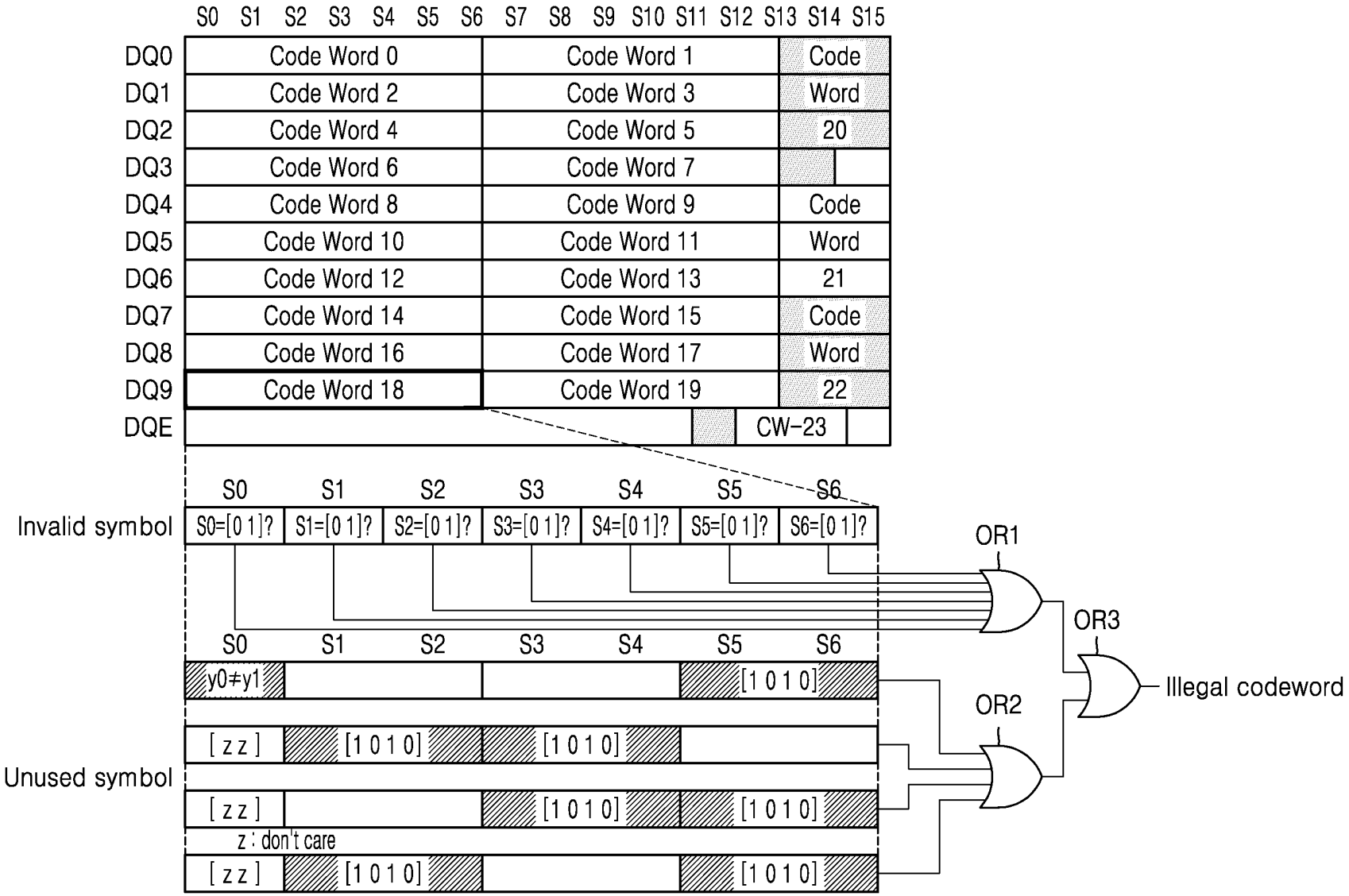

S0 S1 S2 S3 S4 S5 S6 S7 S8 S9 S10 S11 S12 S13 S14 S15
DQ0
DQ1
DQ2
DQ3
DQ4
DQ5
DQ6
DQ7
DQ8
DQ9
DQE
Code Word 0
Code Word 1
Code
Code Word 2
Code Word 3
Word
Code Word 4
Code Word 5
20
Code Word 6
Code Word 7
Code Word 8
Code Word 9
Code
Code Word 10
Code Word 11
Word
Code Word 12
Code Word 13
21
Code Word 14
Code Word 15
Code
Code Word 16
Code Word 17
Word
Code Word 18
Code Word 19
22
CW-23
S0 S1 S2 S3 S4 S5 S6
Invalid symbol
S0=[0 1]?
S1=[0 1]?
S2=[0 1]?
S3=[0 1]?
S4=[0 1]?
S5=[0 1]?
S6=[0 1]?
OR1
OR3
Illegal codeword
OR2
S0 S1 S2 S3 S4 S5 S6
y0≠y1
[1 0 1 0]
Unused symbol
[z z]
[1 0 1 0]
[1 0 1 0]
[z z]
[1 0 1 0]
[1 0 1 0]
z : don't care
[z z]
[1 0 1 0]
[1 0 1 0]

FIG. 5

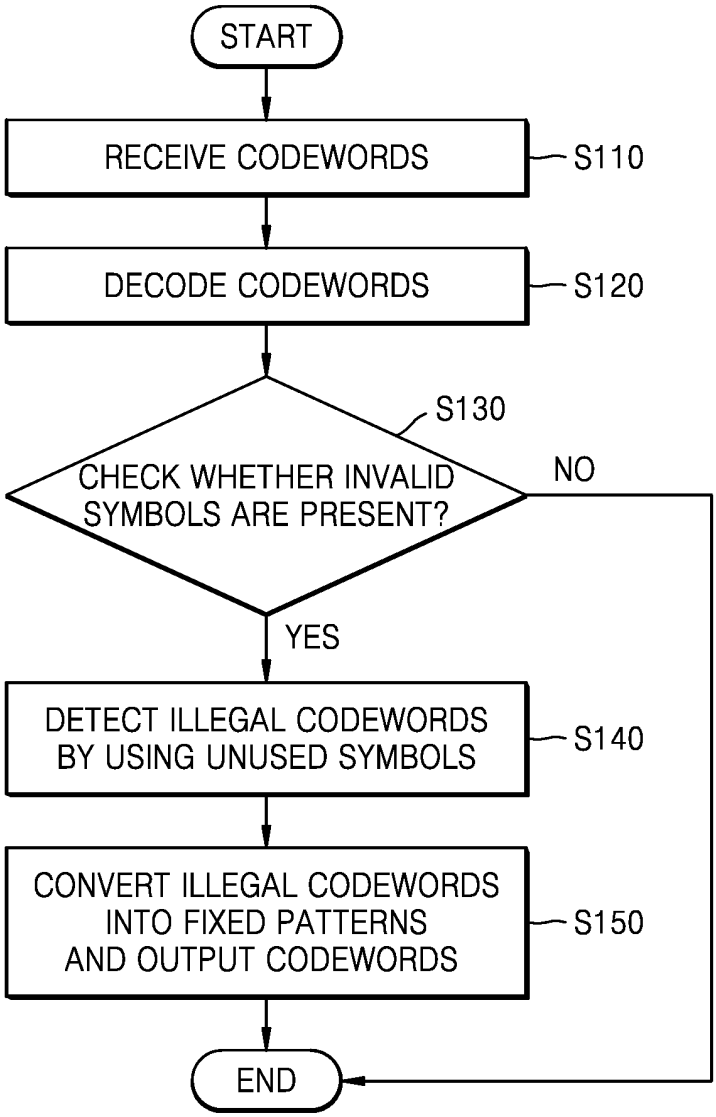

START
RECEIVE CODEWORDS
S110
DECODE CODEWORDS
S120
S130
CHECK WHETHER INVALID SYMBOLS ARE PRESENT?
NO
YES
DETECT ILLEGAL CODEWORDS BY USING UNUSED SYMBOLS
S140
CONVERT ILLEGAL CODEWORDS INTO FIXED PATTERNS AND OUTPUT CODEWORDS
S150
END

FIG. 6

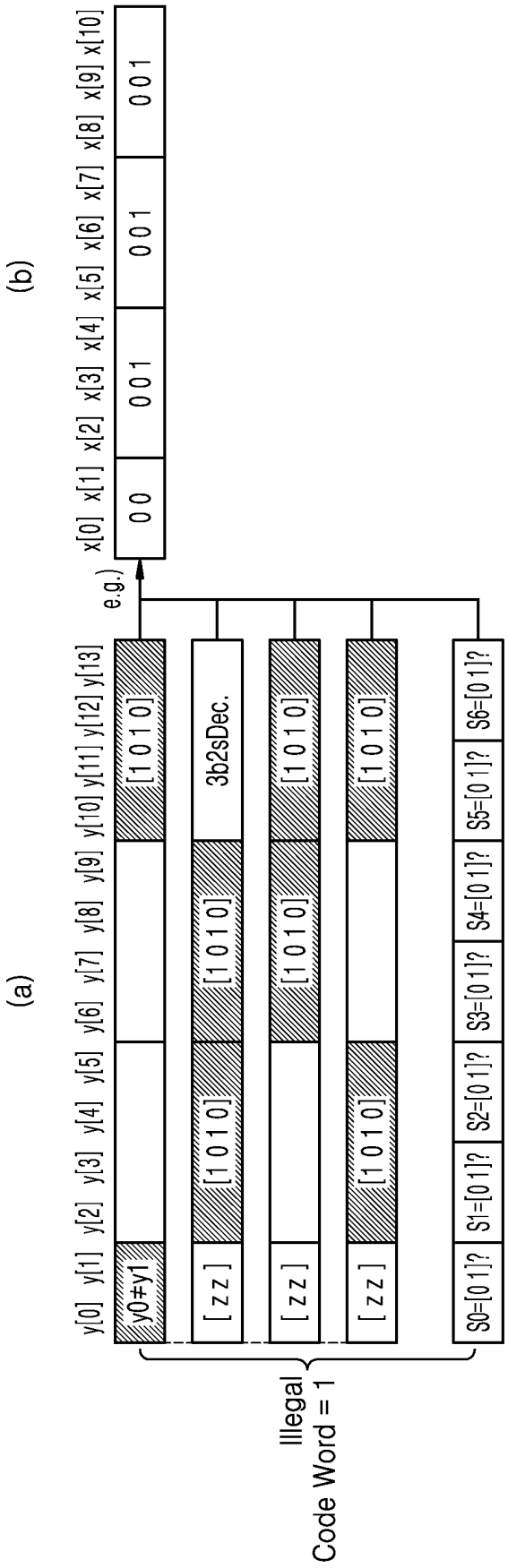

(a)
(b)
y[0] y[1] y[2] y[3] y[4] y[5] y[6] y[7] y[8] y[9] y[10] y[11] y[12] y[13]
y0≠y1
[z z]
[1 0 1 0]
3b2sDec.
S0=[0 1]?
S1=[0 1]?
S2=[0 1]?
S3=[0 1]?
S4=[0 1]?
S5=[0 1]?
S6=[0 1]?
Illegal
Code Word = 1
e.g.)
x[0] x[1] x[2] x[3] x[4] x[5] x[6] x[7] x[8] x[9] x[10]
0 0
0 0 1
0 0 1
0 0 1

FIG. 7

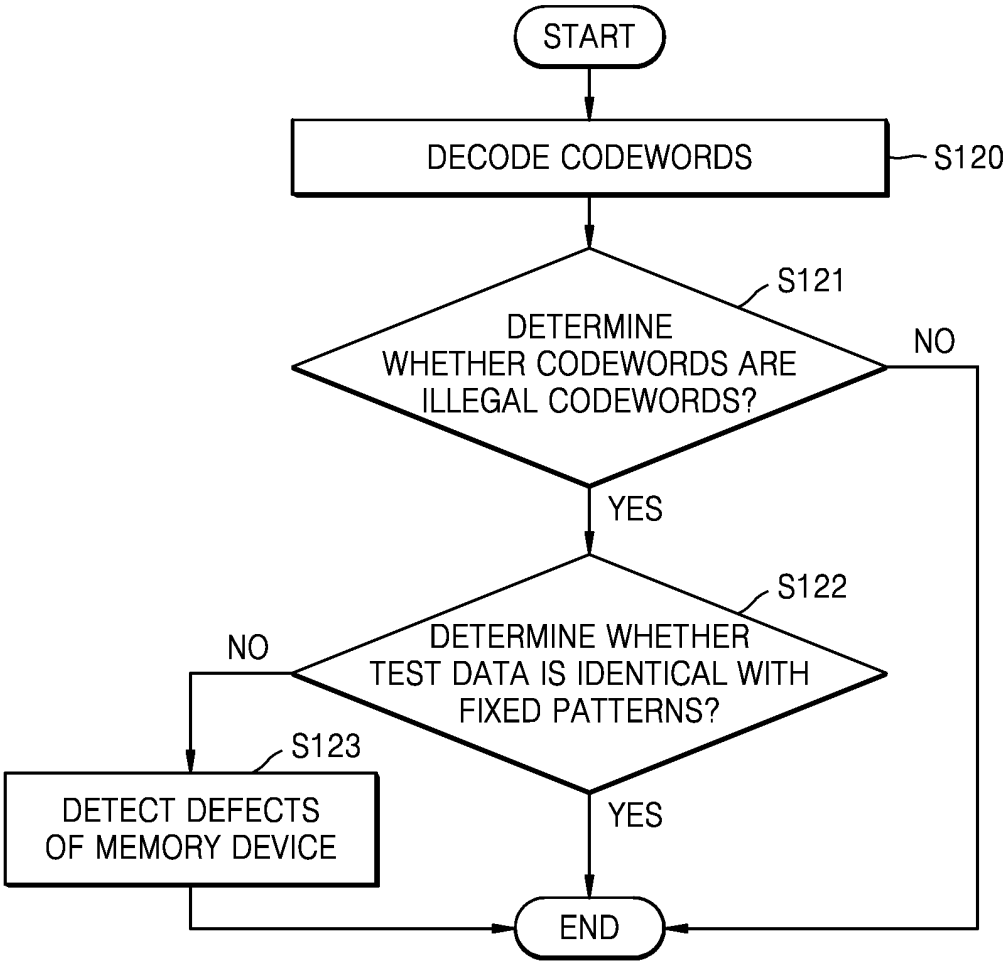

START
DECODE CODEWORDS
S120
S121
DETERMINE WHETHER CODEWORDS ARE ILLEGAL CODEWORDS?
NO
YES
S122
DETERMINE WHETHER TEST DATA IS IDENTICAL WITH FIXED PATTERNS?
NO
S123
DETECT DEFECTS OF MEMORY DEVICE
YES
END

FIG. 8

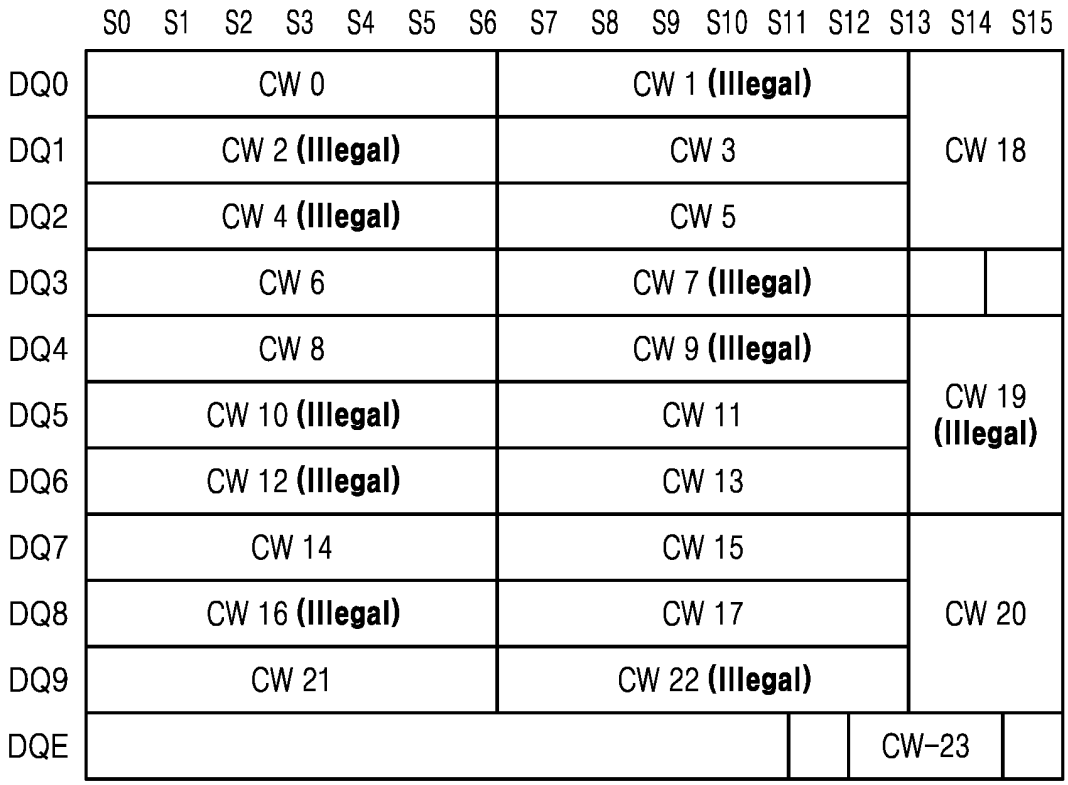

S0 S1 S2 S3 S4 S5 S6 S7 S8 S9 S10 S11 S12 S13 S14 S15
DQ0
DQ1
DQ2
DQ3
DQ4
DQ5
DQ6
DQ7
DQ8
DQ9
DQE
CW 0
CW 1 (Illegal)
CW 2 (Illegal)
CW 3
CW 4 (Illegal)
CW 5
CW 18
CW 6
CW 7 (Illegal)
CW 8
CW 9 (Illegal)
CW 10 (Illegal)
CW 11
CW 12 (Illegal)
CW 13
CW 19 (Illegal)
CW 14
CW 15
CW 16 (Illegal)
CW 17
CW 20
CW 21
CW 22 (Illegal)
CW-23

FIG. 9

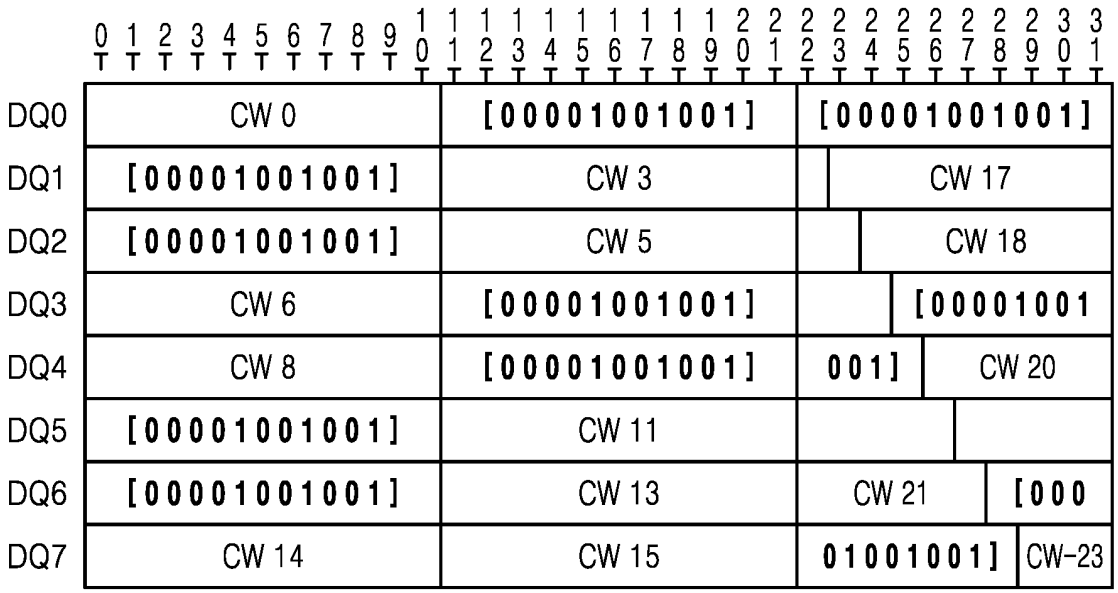

0T 1T 2T 3T 4T 5T 6T 7T 8T 9T 10T 11T 12T 13T 14T 15T 16T 17T 18T 19T 20T 21T 22T 23T 24T 25T 26T 27T 28T 29T 30T 31T
DQ0
CW 0
[00001001001]
[00001001001]
DQ1
[00001001001]
CW 3
CW 17
DQ2
[00001001001]
CW 5
CW 18
DQ3
CW 6
[00001001001]
[00001001
DQ4
CW 8
[00001001001]
001]
CW 20
DQ5
[00001001001]
CW 11
DQ6
[00001001001]
CW 13
CW 21
[000
DQ7
CW 14
CW 15
01001001]
CW-23

FIG. 10

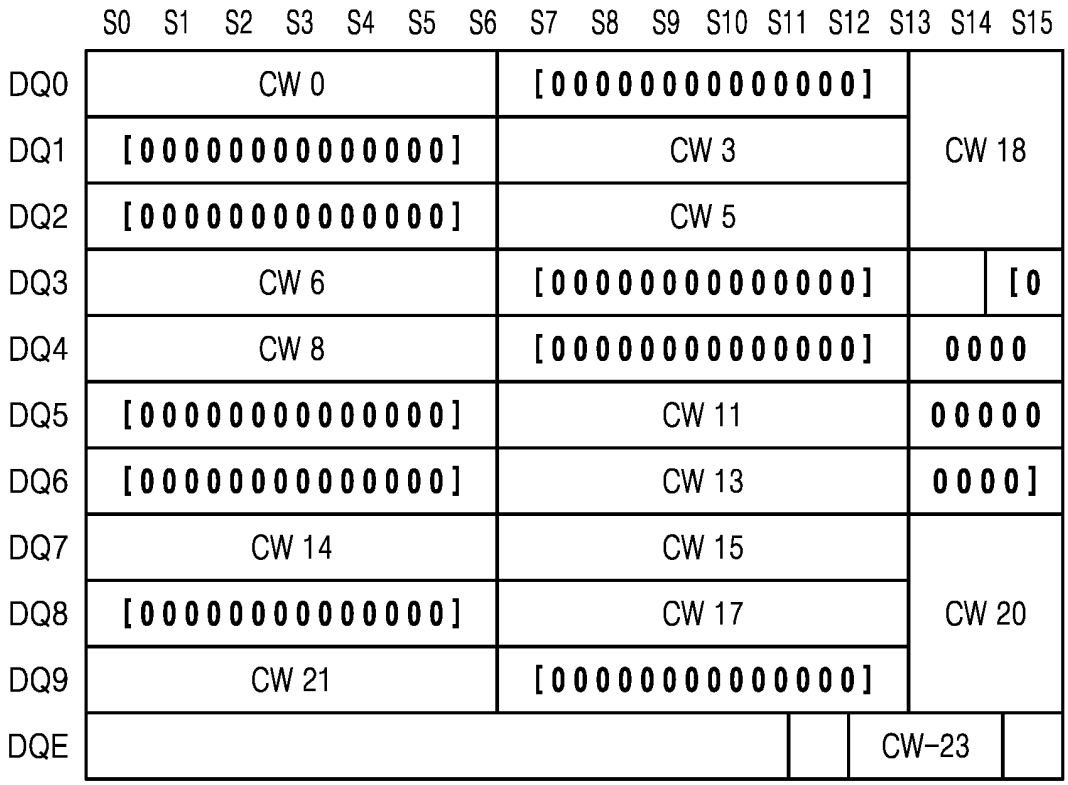
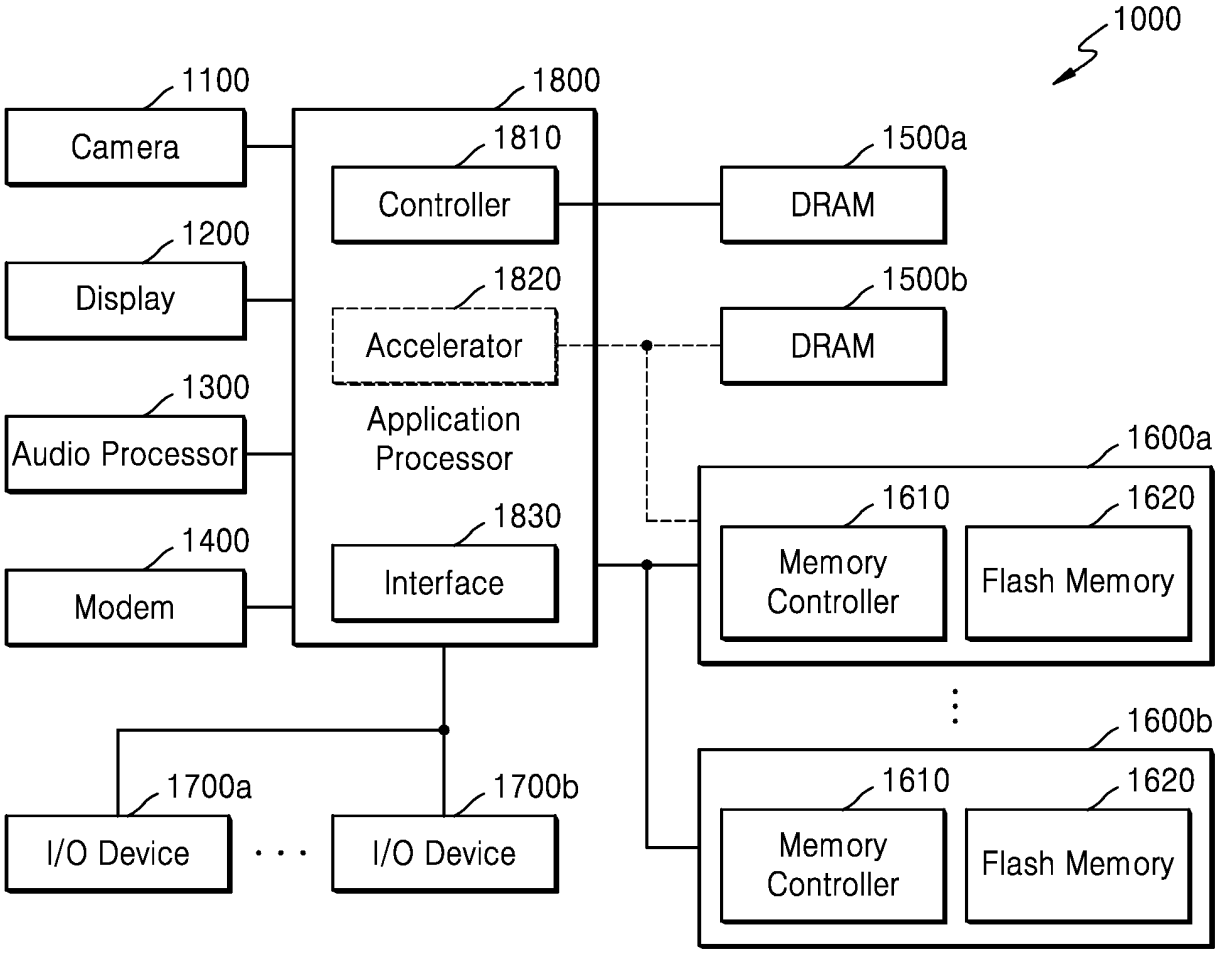

S0 S1 S2 S3 S4 S5 S6 S7 S8 S9 S10 S11 S12 S13 S14 S15
DQ0
CW 0
[0000000000000000]
CW 18
DQ1
[0000000000000000]
CW 3
DQ2
[0000000000000000]
CW 5
DQ3
CW 6
[0000000000000000]
[0
DQ4
CW 8
[0000000000000000]
0000
DQ5
[0000000000000000]
CW 11
00000
DQ6
[0000000000000000]
CW 13
0000]
DQ7
CW 14
CW 15
DQ8
[0000000000000000]
CW 17
CW 20
DQ9
CW 21
[0000000000000000]
DQE
CW-23

FIG. 11


1000
1100
Camera
1200
Display
1300
Audio Processor
1400
Modem
1800
1810
Controller
1820
Accelerator
Application Processor
1830
Interface
1500a
DRAM
1500b
DRAM
1600a
1610
Memory Controller
1620
Flash Memory
1600b
1610
Memory Controller
1620
Flash Memory
1700a
I/O Device
1700b
I/O Device