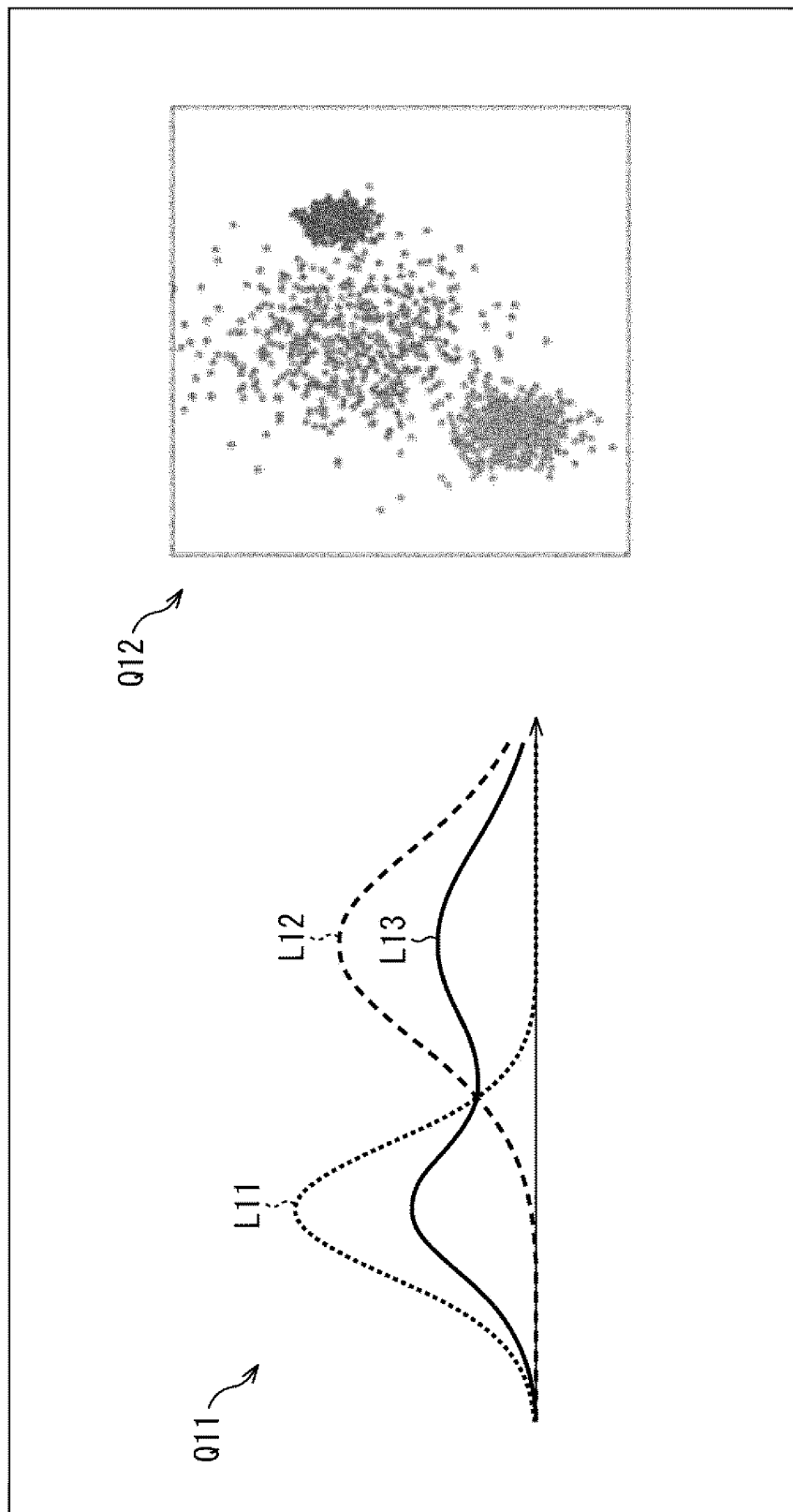
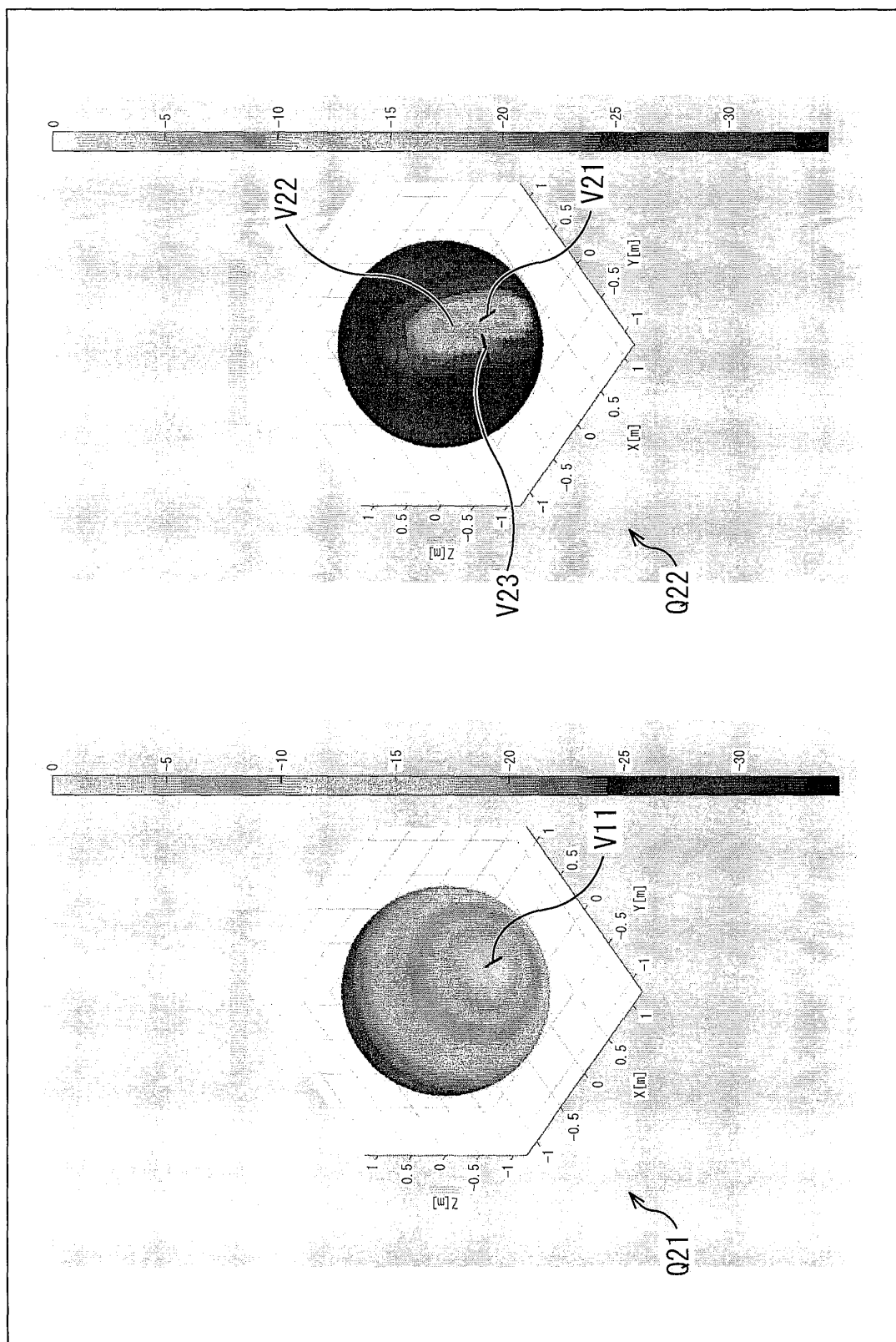


【図1】
FIG. 1



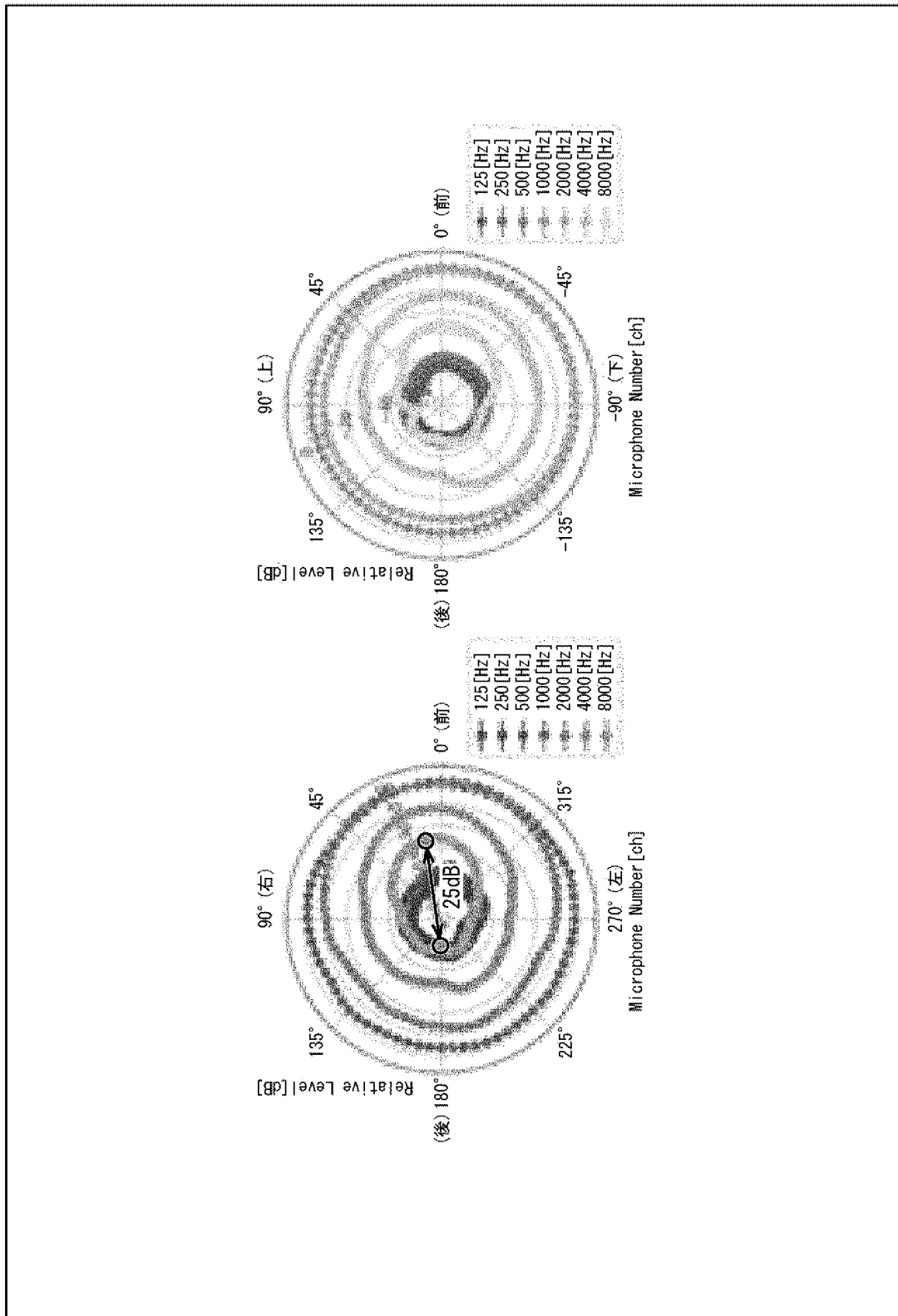
[図 2]

FIG. 2

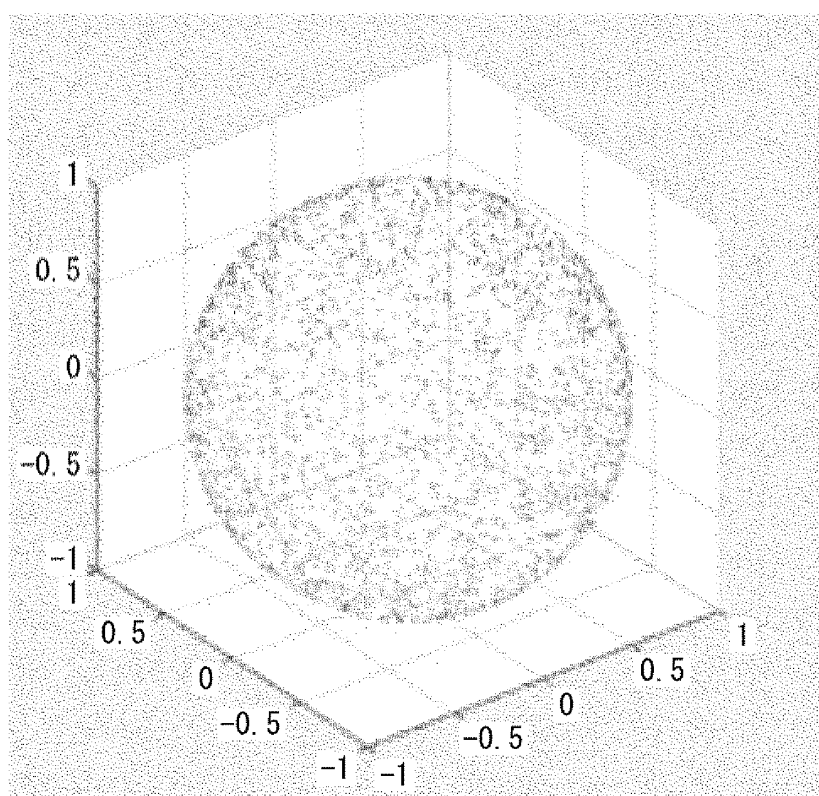


差替え用紙(規則26)

[図3]
FIG. 3



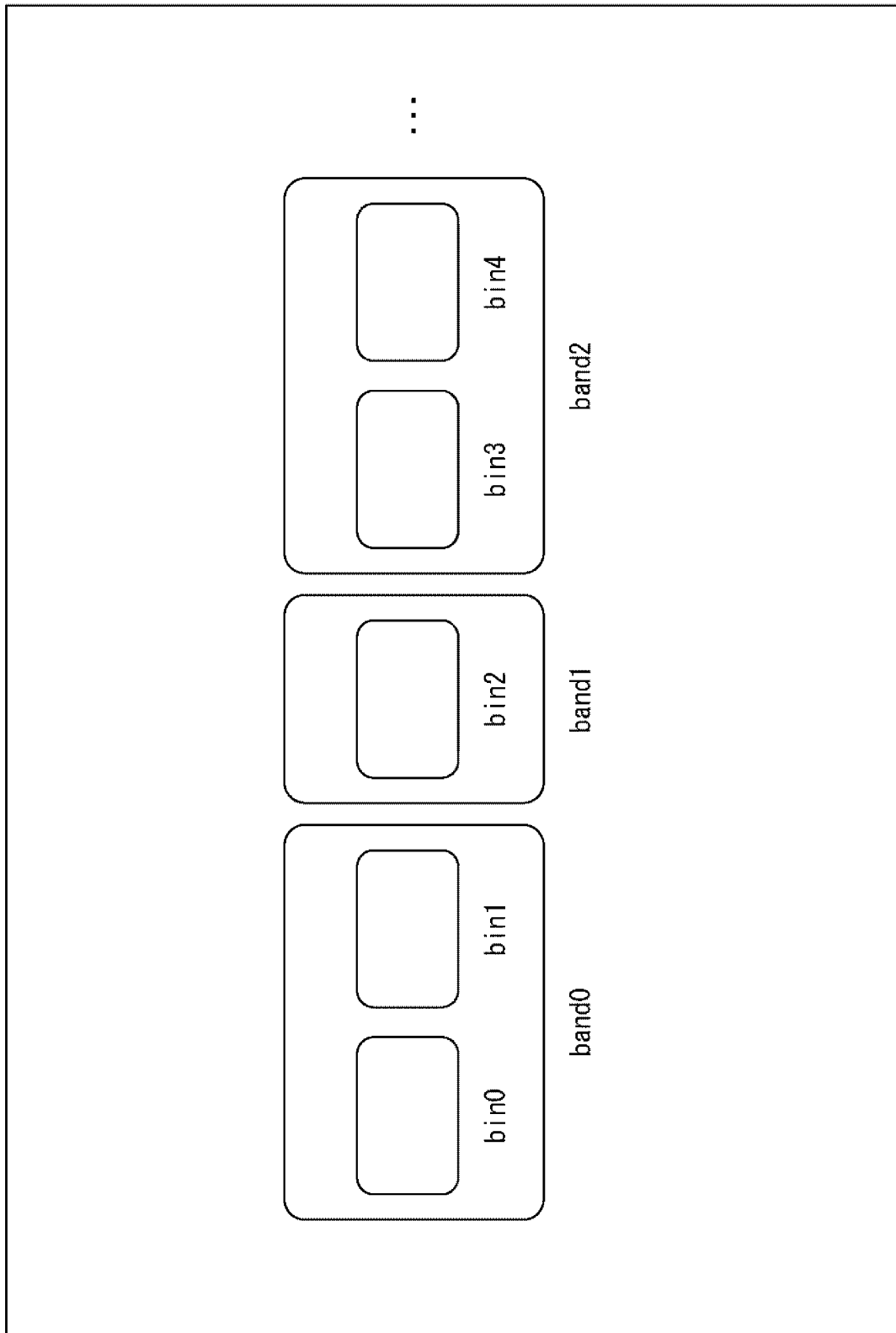
[図4]
FIG. 4



【図5】
FIG. 5

Syntax	No. of bits	Data format
directivityConfig(num_sound_types_id)		
{		
fs	32	unsigned int
num_point_indices: //データポイント数	32	unsigned int
for (i=0; i<num_point_indices; i++) {		
azimuth_table[i] //方位角	32	float
elevation_table[i] //仰角	32	float
distance[i] (radius) //距離 [meter]	32	float
}		
bin_count //周波数ポイント数	16	unsigned short
for (i_bin=0; i_bin<bin_count; i_bin++){		
freq[i_bin] //周波数[Hz]	32	unsigned int
}		
// VMF/kent distribution parameter		
band_count //モデル化するband数	16	unsigned short
for (i_band = 0; i_band<band_count; i_band++){		
mix_count[i_band] //各bandでの混合数	16	unsigned short
bin_range_per_band[i_band] //各bandに含まれる元データのbin情報	16	unsigned short
for (i_mix=0; i_mix<mix_count[i_band]; i_mix++){		
weight[i_band][i_mix]	32	float
kappa[i_band][i_mix]	32	float
gamma1[i_band][i_mix][x]	32	float
gamma1[i_band][i_mix][y]	32	float
dist_flag //Kent分布/VMF分布選択flag	8	unsigned char
if (dist_flag == 1) {		
beta[i_band][i_mix]	32	float
gamma2[i_band][i_mix][x]	32	float
gamma2[i_band][i_mix][y]	32	float
gamma3[i_band][i_mix][x]	32	float
gamma3[i_band][i_mix][y]	32	float
} //flag		
} //mix		
} //band		
for (i_bin=0; i_bin < bin_count; i_bin++){		
scale_factor[i_bin] //各binにおける指向性データのダイナミックレンジ	32	float
offset[i_bin] //各binにおける指向性データ最小値	32	float
} //bin		
for (i_point=0; i_point<num_point){		
diff_data[i_point] //差分情報	TBD	
}		
}		

[図6]
FIG. 6

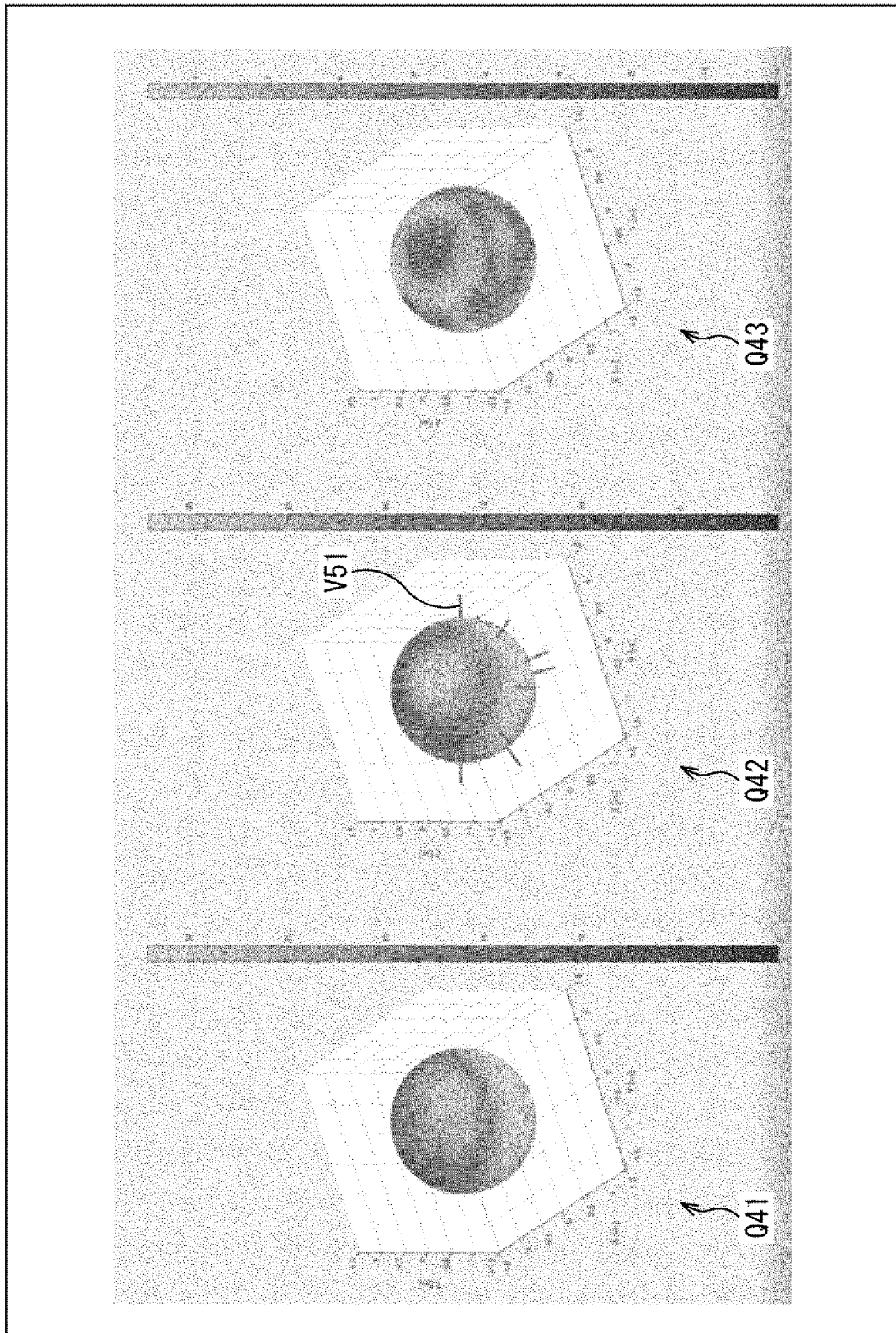


[図7]
FIG. 7

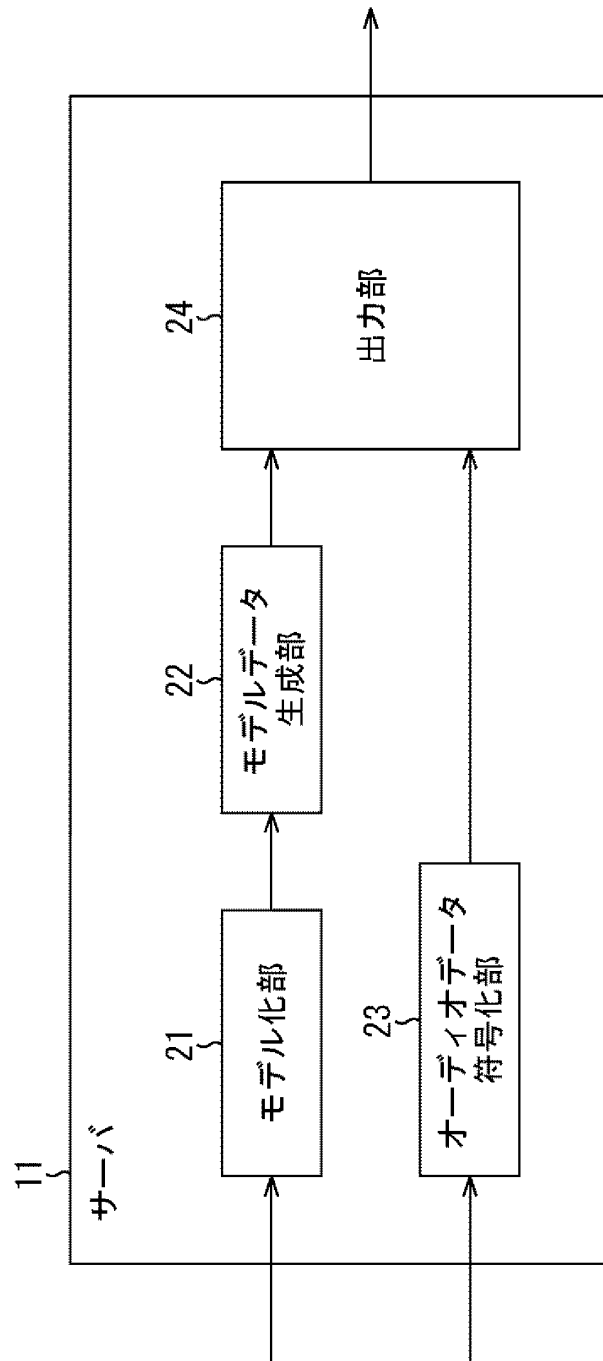
mixture		1	12	19
	band1	band2	band3	
weight	32	384		608
kappa	32	384		608
mean	96	1152		1824
offset	320	384		224
scale factor	320	384		224
Total (KB)	0. 097656	0. 328125		0. 42578125
				0. 8515625

Original data	Modeled by mixture of vMFdistribution
306KB	0. 85KB

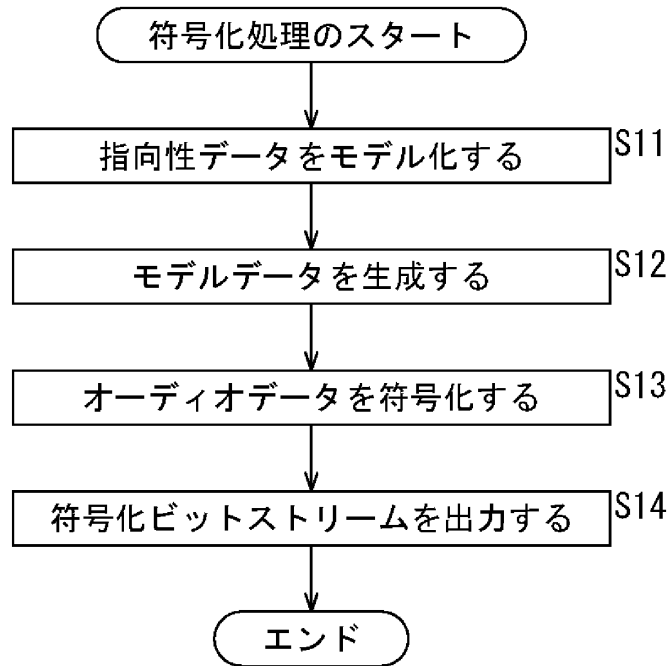
[図8]
FIG. 8



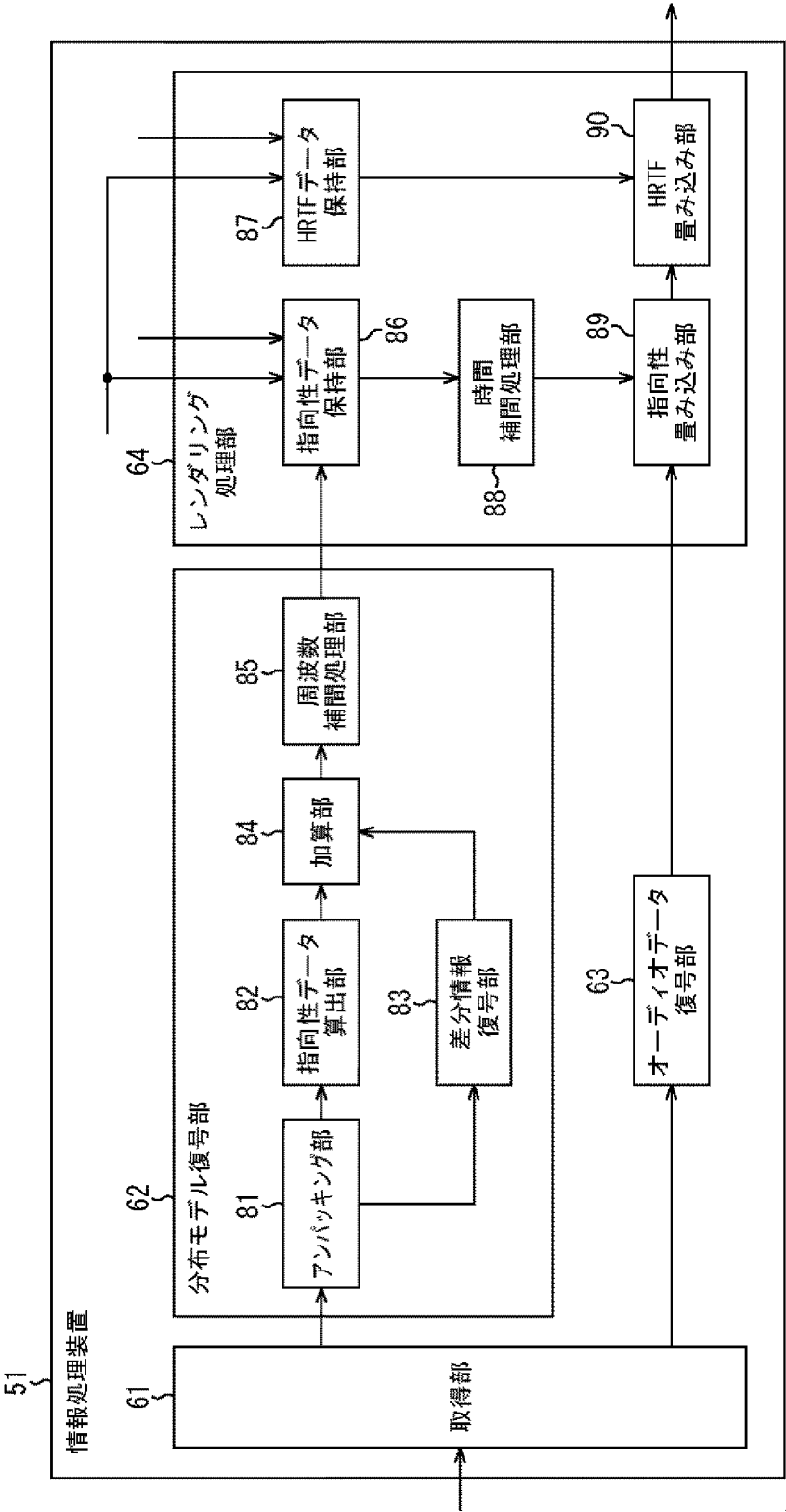
【図9】
FIG. 9



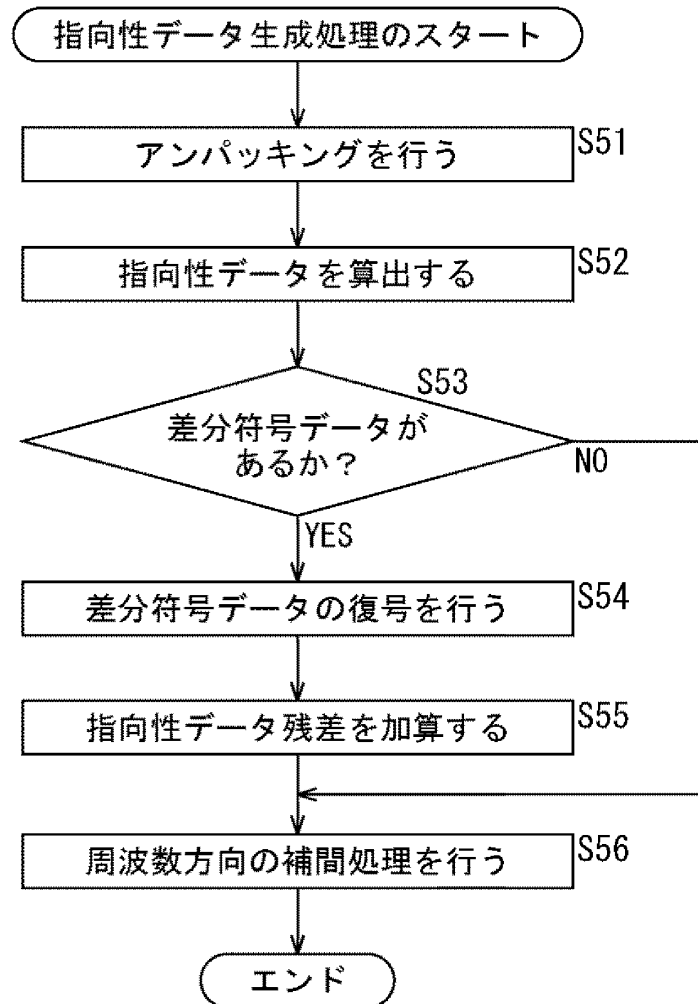
【図10】
FIG. 10



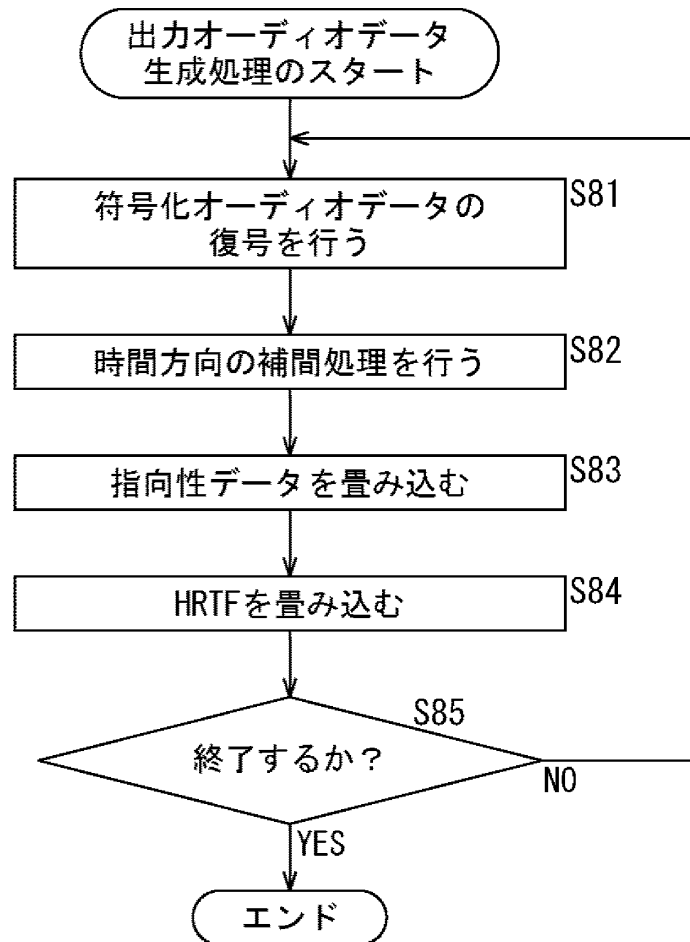
[図11]
FIG. 11



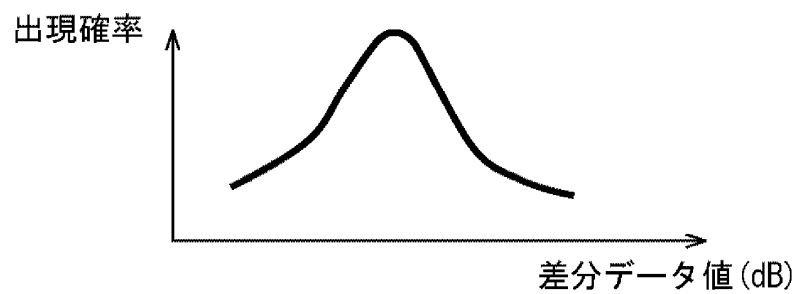
【図12】
FIG. 12



【図13】
FIG. 13



【図14】
FIG. 14



Syntax	No. of bits	Data format
directivityConfig(num_sound_types_id)		
{		
fs	32	unsigned int
num_point_indices; //データポイント数	32	unsigned int
for (i = 0; i < num_point_indices; i++) {		
azimuth_table[i] //仰角	16	unsigned short
elevation_table[i] //方位角	16	unsigned short
distance[i] (radius) //距離 [meter]	32	float
}		
bin_count //周波数ポイント数	16	unsigned short
for (i_bin=0; i_bin<bin_count; i_bin++){		
freq[i_bin] //周波数[Hz]	32	unsigned int
}		
// vMF/kent distribution parameter		
band_count //モデル化するband数	8	unsigned char
for (i_band = 0; i_band < band_count; i_band++){		
mix_count[i_band] //各バンドでの混合数	8	unsigned char
bin_range_per_band[i_band] //各bandに含まれる元データの bin情報	16	unsigned short
for (i_mix = 0; i_mix < mix_count[i_band]; i_mix++){		
weight[i_band][i_mix]	32	float
kappa[i_band][i_mix]	32	float
gamma1[i_band][i_mix][x]	32	float
gamma1[i_band][i_mix][y]	32	float
dist_flag //Kent分布/vMF分布選択flag	1	bool
if(dist_flag == 1){		
beta[i_band][i_mix]	32	float
gamma2[i_band][i_mix][x]	32	float
gamma2[i_band][i_mix][y]	32	float
gamma3[i_band][i_mix][x]	32	float
gamma3[i_band][i_mix][y]	32	float
} //flag		
} //i_mix		
} //i_band		

続<

[図16]
FIG. 16

続き

<pre>for (i_bin=0; i_bin < bin_count; i_bin++){ scale_factor[i_bin] offset[i_bin] } //bin mode table_index int db_resolution for (i_bin = 0; i_bin < bin_count; i_bin++){ for (i_point=0; i_point < num_point){ diff_data[i_bin][i_point] } }</pre>	<pre>//各binにおける指向性データの ダイナミックレンジ //各binにおける指向性データ 最小値 //符号化に使われたハイブリッド のモードID //符号化に使われたtable index // 1.0dBごとの量子化など、量子 化stepサイズを指定 ※例 0: 量子化無し, 1: 0.01dB, 2: 0.2dB, 3: 0.4dB, ... 2^8=256: 25.6dB //差分情報を符号化した huffman符号</pre>	<pre>32 32 8 8 8 1~nbits_res_data</pre>	<pre>float float Unsigned char Unsigned char Unsigned char</pre>
--	---	--	--

[図17]
FIG. 17

diff_mode_count	8	Unsigned char
// 差分符号化の方式総数(差分符号化無し、空間差分、bin間差分、etc.)		
for (i_mode = 0; i_mode < diff_mode_count; i_mode++)		
int nbits_res_data	32	Int
// Huffman codeの最大bit数、1.0dBの場合7bit, 0-128dB Huffmanコード最大語長		
element_count //Huffman tableの要素数	32	int
for (i_element = 0; i_element < element_count; i_element++)		
Huff_dec_table[i_element]	32	int
Huffman code ->データ		
}		
}		

[図18]

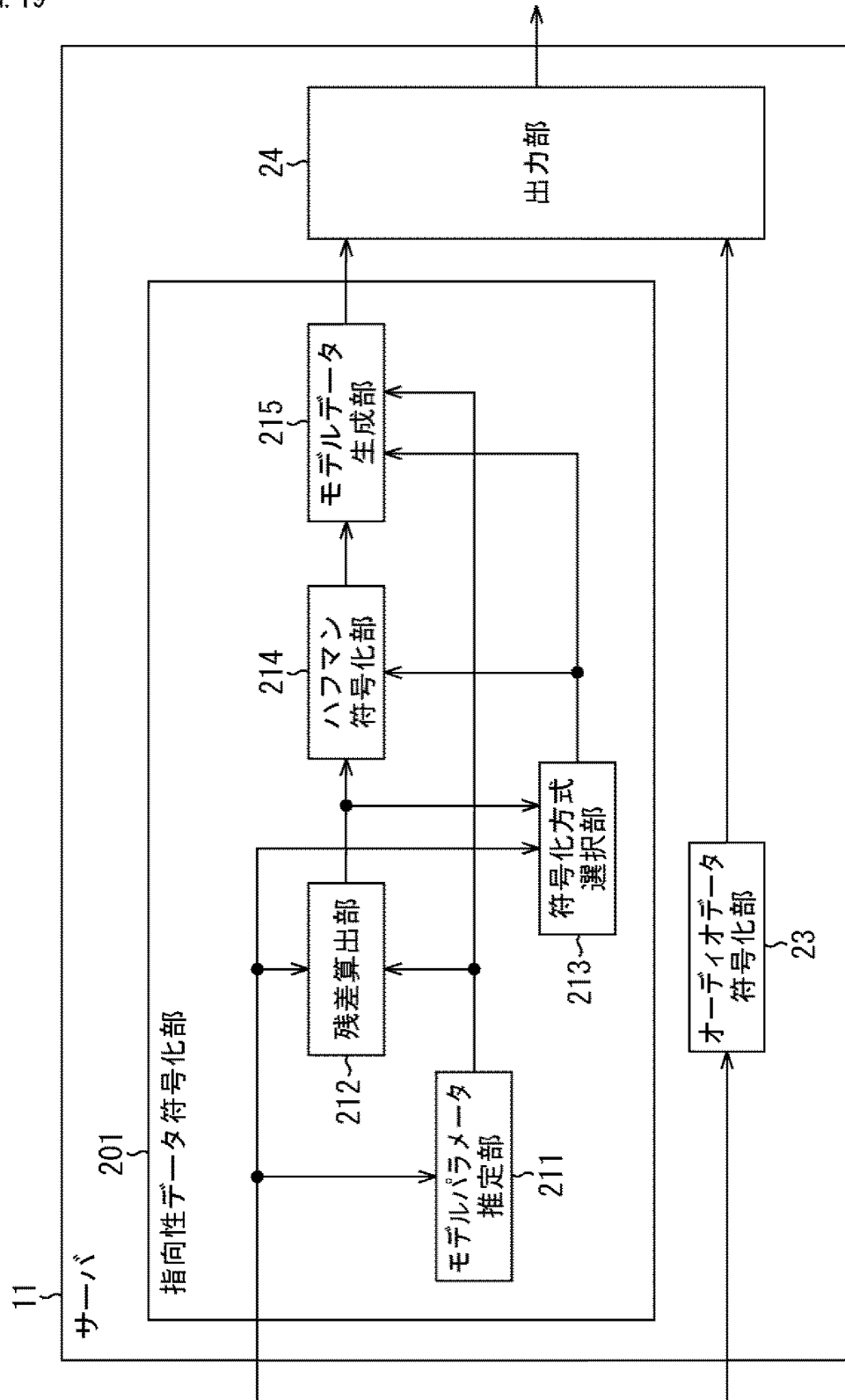
FIG. 18

```

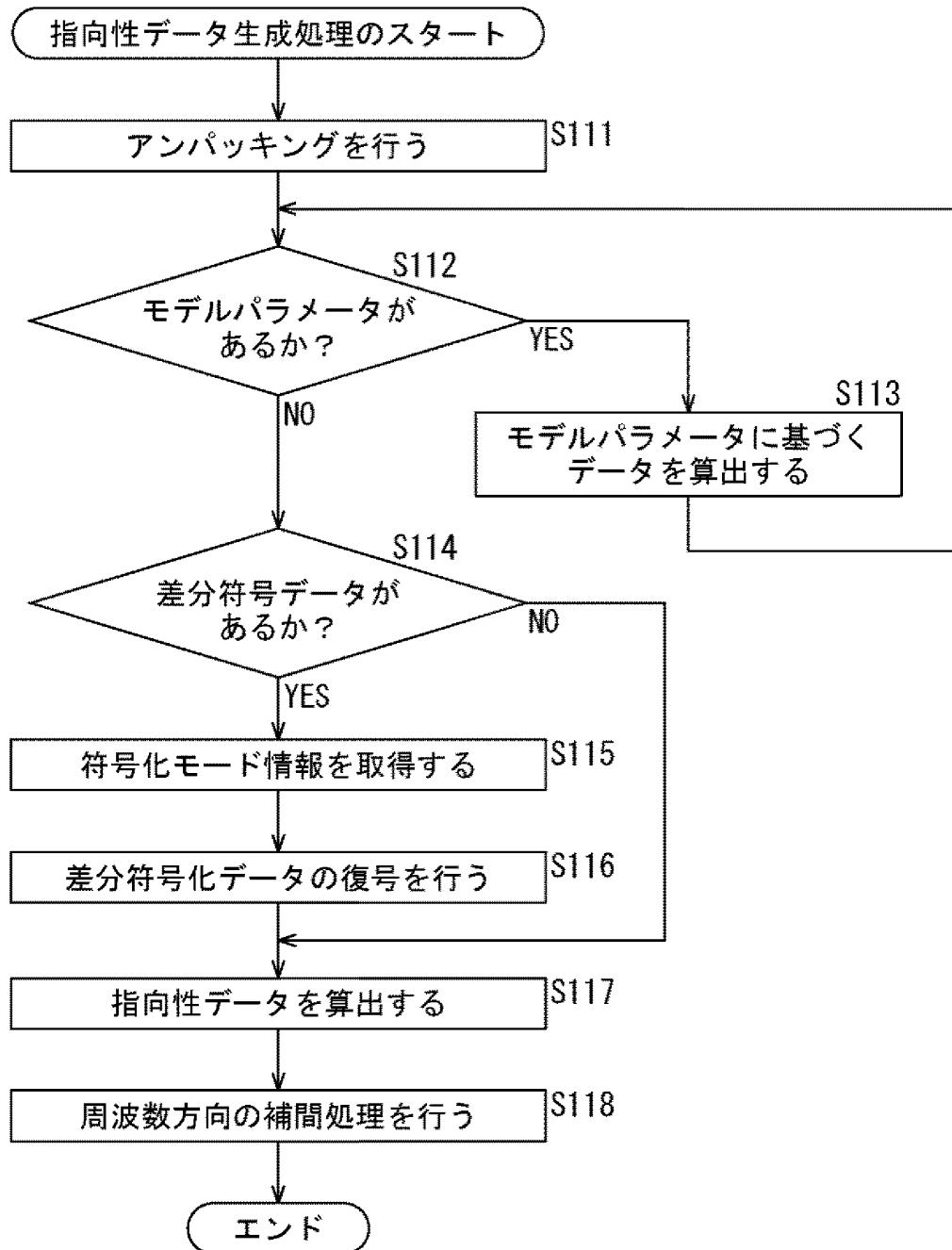
const unsigned char g_a_res_model_dectbl[NCODES_RES_MODEL_DECTBL] = {
100, 100, 100, 100, 100, 100, 100, 100, 2, 2, 4, 4, 12, 12, 12, 12,
36, 36, 36, 36, 36, 36, 36, 36, 16, 16, 16, 16, 108, 108, 108, 108,
50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50,
81, 81, 81, 81, 81, 81, 81, 81, 81, 81, 81, 81, 81, 81, 81, 81,
21, 21, 21, 21, 22, 22, 22, 22, 20, 20, 20, 20, 110, 110, 110, 110,
99, 99, 99, 99, 99, 99, 99, 99, 15, 15, 15, 15, 112, 112, 112, 112,
13, 13, 13, 13, 111, 111, 111, 111, 37, 37, 37, 37, 37, 37, 37, 37,
78, 78, 78, 78, 78, 78, 78, 78, 78, 78, 78, 78, 78, 78, 78, 78,
52, 52, 52, 52, 52, 52, 52, 52, 52, 52, 52, 52, 52, 52, 52, 52,
53, 53, 53, 53, 53, 53, 53, 53, 53, 53, 53, 53, 53, 53, 53, 53,
51, 51, 51, 51, 51, 51, 51, 51, 51, 51, 51, 51, 51, 51, 51, 51,
38, 38, 38, 38, 38, 38, 38, 38, 109, 109, 109, 109, 6, 6, 8, 8,
80, 80, 80, 80, 80, 80, 80, 80, 80, 80, 80, 80, 80, 80, 80, 80,
79, 79, 79, 79, 79, 79, 79, 79, 79, 79, 79, 79, 79, 79, 79, 79,
58, 58, 58, 58, 58, 58, 58, 58, 58, 58, 58, 58, 58, 58, 58, 58,
77, 77, 77, 77, 77, 77, 77, 77, 77, 77, 77, 77, 77, 77, 77, 77,
56, 56, 56, 56, 56, 56, 56, 56, 56, 56, 56, 56, 56, 56, 56, 56,
98, 98, 98, 98, 98, 98, 98, 98, 25, 25, 25, 25, 107, 107, 107, 107,
54, 54, 54, 54, 54, 54, 54, 54, 54, 54, 54, 54, 54, 54, 54, 54,
61, 61, 61, 61, 61, 61, 61, 61, 61, 61, 61, 61, 61, 61, 61, 61,
97, 97, 97, 97, 97, 97, 97, 97, 39, 39, 39, 39, 39, 39, 39, 39,
59, 59, 59, 59, 59, 59, 59, 59, 59, 59, 59, 59, 59, 59, 59, 59,
57, 57, 57, 57, 57, 57, 57, 57, 57, 57, 57, 57, 57, 57, 57, 57,
60, 60, 60, 60, 60, 60, 60, 60, 60, 60, 60, 60, 60, 60, 60, 60,
55, 55, 55, 55, 55, 55, 55, 55, 55, 55, 55, 55, 55, 55, 55, 55,
74, 74, 74, 74, 74, 74, 74, 74, 74, 74, 74, 74, 74, 74, 74, 74,
62, 62, 62, 62, 62, 62, 62, 62, 62, 62, 62, 62, 62, 62, 62, 62,
63, 63, 63, 63, 63, 63, 63, 63, 63, 63, 63, 63, 63, 63, 63, 63,
64, 64, 64, 64, 64, 64, 64, 64, 64, 64, 64, 64, 64, 64, 64, 64,
76, 76, 76, 76, 76, 76, 76, 76, 76, 76, 76, 76, 76, 76, 76, 76,
40, 40, 40, 40, 40, 40, 40, 40, 125, 125, 5, 5, 7, 7, 127, 127,
65, 65, 65, 65, 65, 65, 65, 65, 65, 65, 65, 65, 65, 65, 65, 65,
75, 75, 75, 75, 75, 75, 75, 75, 75, 75, 75, 75, 75, 75, 75, 75,
27, 27, 27, 27, 24, 24, 24, 24, 28, 28, 28, 28, 106, 106, 106, 106,
73, 73, 73, 73, 73, 73, 73, 73, 73, 73, 73, 73, 73, 73, 73, 73,
66, 66, 66, 66, 66, 66, 66, 66, 66, 66, 66, 66, 66, 66, 66, 66,
69, 69, 69, 69, 69, 69, 69, 69, 69, 69, 69, 69, 69, 69, 69, 69,
70, 70, 70, 70, 70, 70, 70, 70, 70, 70, 70, 70, 70, 70, 70, 70,
68, 68, 68, 68, 68, 68, 68, 68, 68, 68, 68, 68, 68, 68, 68, 68,
95, 95, 95, 95, 95, 95, 95, 95, 96, 96, 96, 96, 96, 96, 96, 96,
67, 67, 67, 67, 67, 67, 67, 67, 67, 67, 67, 67, 67, 67, 67, 67,
72, 72, 72, 72, 72, 72, 72, 72, 72, 72, 72, 72, 72, 72, 72, 72,
71, 71, 71, 71, 71, 71, 71, 71, 71, 71, 71, 71, 71, 71, 71, 71,
26, 26, 26, 26, 29, 29, 29, 29, 41, 41, 41, 41, 41, 41, 41, 41,
93, 93, 93, 93, 93, 93, 93, 93, 124, 124, 126, 126, 123, 123, 122, 122,
42, 42, 42, 42, 42, 42, 42, 42, 92, 92, 92, 92, 92, 92, 92, 92,
91, 91, 91, 91, 91, 91, 91, 91, 43, 43, 43, 43, 43, 43, 43, 43,
105, 105, 105, 105, 9, 9, 121, 121, 30, 30, 30, 30, 1, 3, 120, 120,
94, 94, 94, 94, 94, 94, 94, 94, 32, 32, 32, 32, 104, 104, 104, 104,
45, 45, 45, 45, 45, 45, 45, 45, 31, 31, 31, 31, 10, 10, 11, 11,
90, 90, 90, 90, 90, 90, 90, 90, 102, 102, 102, 102, 116, 116, 119, 119,
44, 44, 44, 44, 44, 44, 44, 44, 89, 89, 89, 89, 89, 89, 89, 89,
46, 46, 46, 46, 46, 46, 46, 46, 34, 34, 34, 34, 103, 103, 103, 103,
47, 47, 47, 47, 47, 47, 47, 47, 33, 33, 33, 33, 118, 118, 18, 18,
88, 88, 88, 88, 88, 88, 88, 88, 114, 114, 115, 115, 35, 35, 35, 35,
85, 85, 85, 85, 85, 85, 85, 85, 86, 86, 86, 86, 86, 86, 86, 86,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
48, 48, 48, 48, 48, 48, 48, 48, 49, 49, 49, 49, 49, 49, 49, 49,
101, 101, 101, 101, 23, 23, 113, 113, 84, 84, 84, 84, 84, 84, 84, 84,
87, 87, 87, 87, 87, 87, 87, 87, 117, 117, 14, 14, 17, 17, 19, 19,
83, 83, 83, 83, 83, 83, 83, 83, 82, 82, 82, 82, 82, 82, 82, 82
};

```

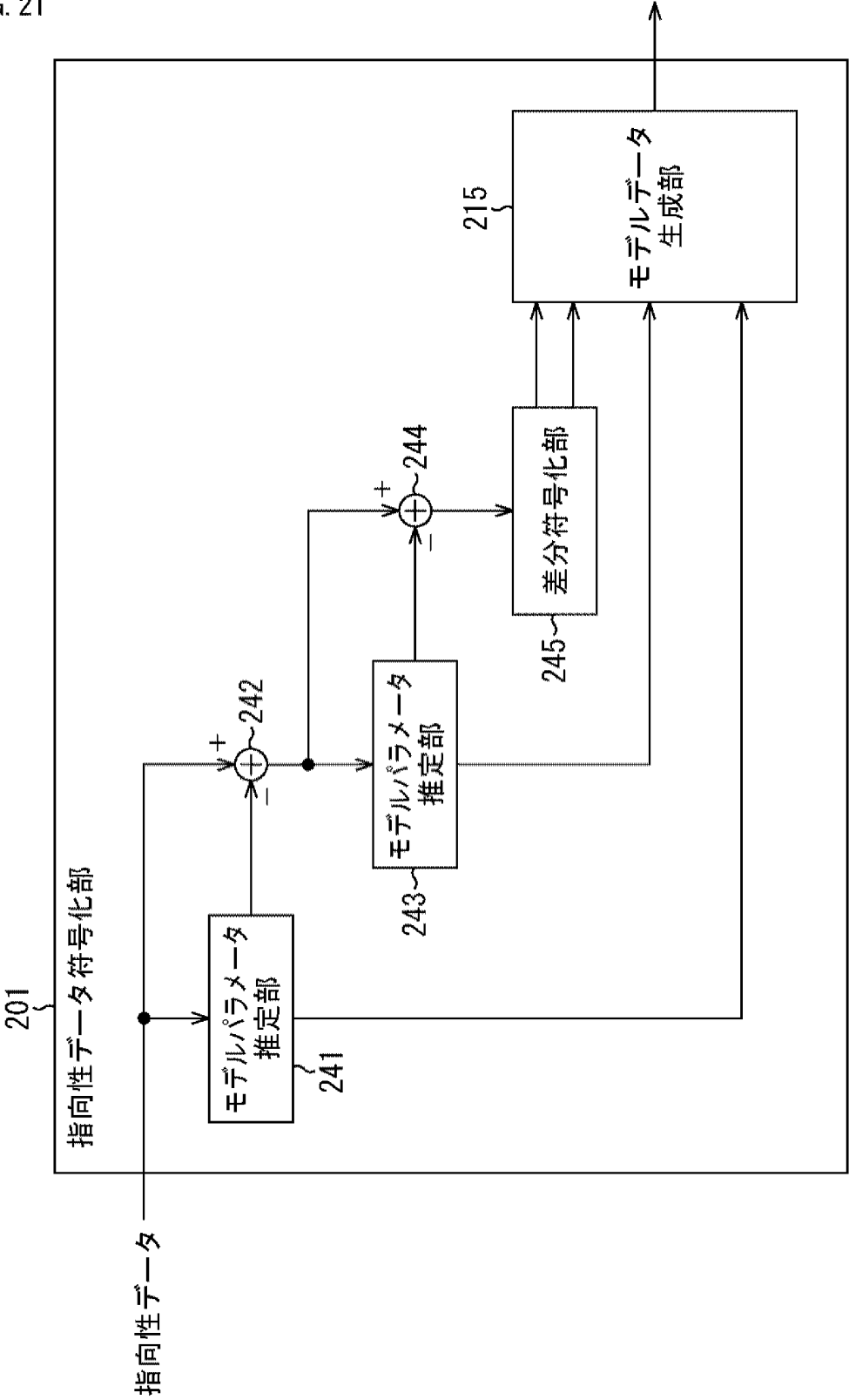
[図19]
FIG. 19



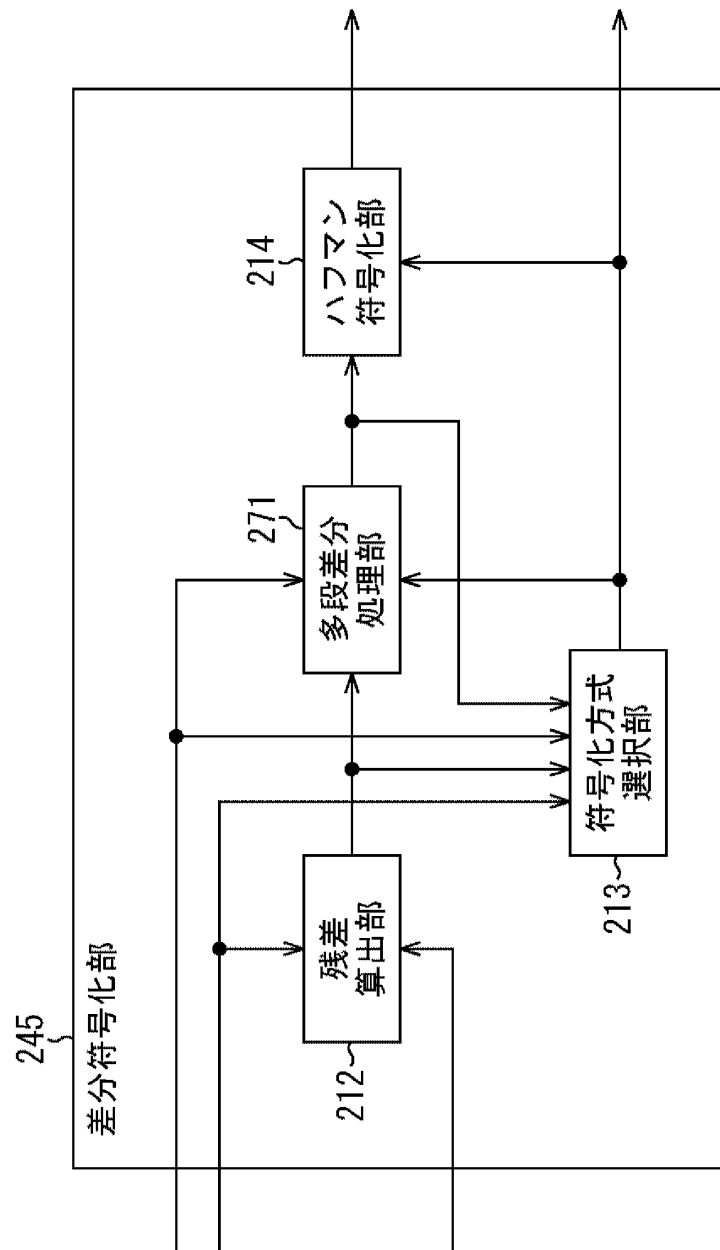
【図20】
FIG. 20



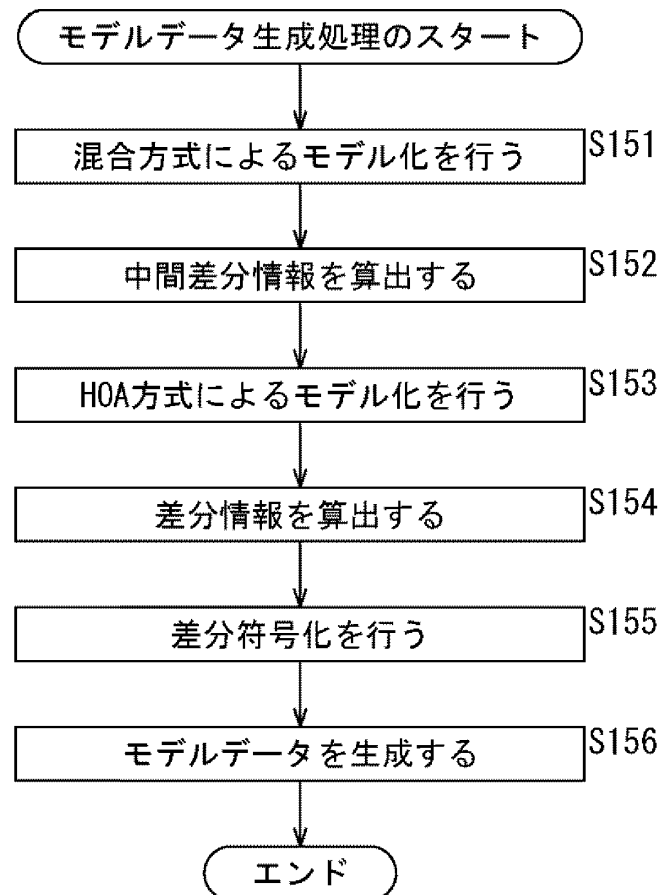
[図21]
FIG. 21



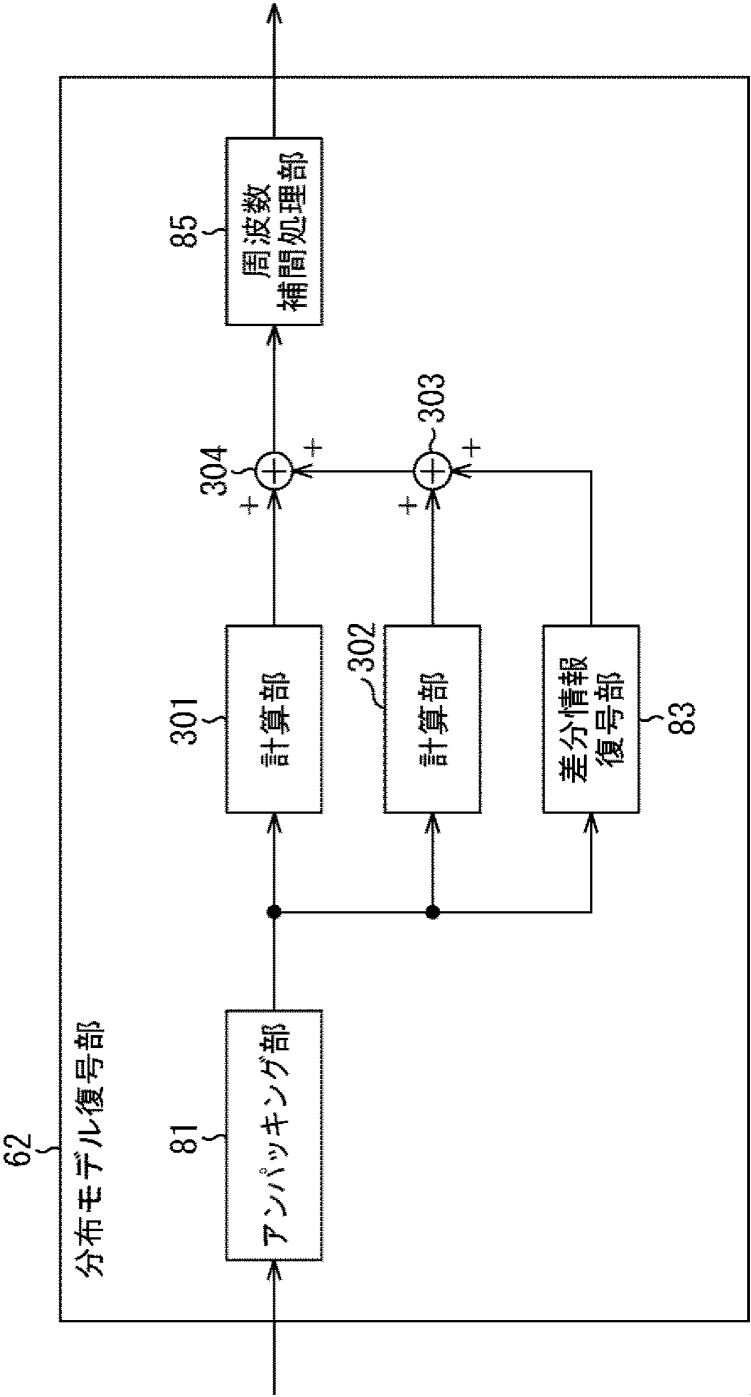
[図22]
FIG. 22



【図23】
FIG. 23



[図24]
FIG. 24

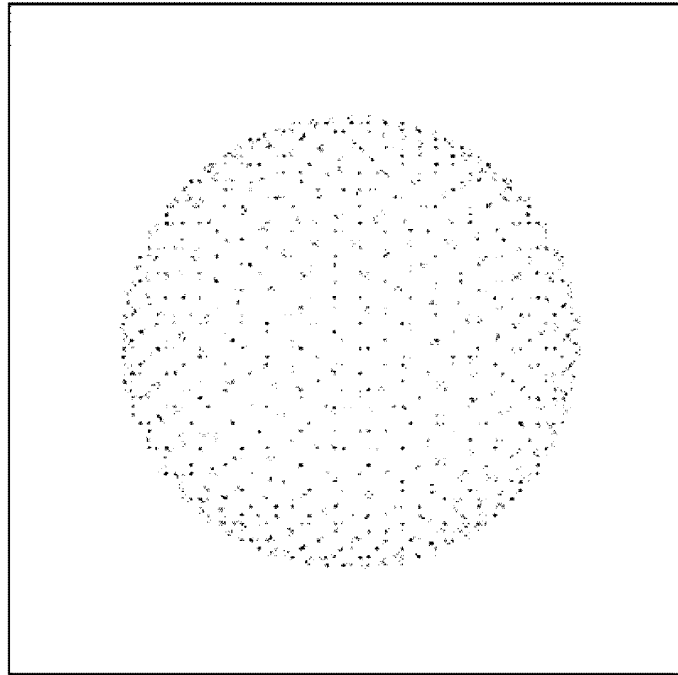


[図25]

FIG. 25

Syntax	Mnemonic	Modified No. of bits
DirectivityReproduction() {		
bin_count;	uimsbf	12
for (i=0; i < bin_count; i++) {		
bin_freq[i];	bslbf	18
}		
band_count;	uimsbf	12
for (j=0; j < band_count; j++) {		
mix_count[j];	uimsbf	8
bin_range_per_band[j];	uimsbf	12
for (k=0; k < mix_count[j]; k++) {		
kappa[j][k];	uimsbf	11
weight[j][k];	uimsbf	10
gamma_x[j][k];	uimsbf	9
gamma_y[j][k];	uimsbf	9
gamma_z[j][k];	uimsbf	9
dist_flag	bool	1
if (dist_flag == 1) {		
beta[j][k];	uimsbf	11
gamma2_x[j][k];	uimsbf	9
gamma2_y[j][k];	uimsbf	9
gamma2_z[j][k];	uimsbf	9
gamma3_x[j][k];	uimsbf	9
gamma3_y[j][k];	uimsbf	9
gamma3_z[j][k];	uimsbf	9
} //flag		
} //mix_count		
} //band_count		
for (i=0; i < bin_count; i++) {		
scale_factor[i];	uimsbf	7
offset[i];	uimsbf	7
}		
}		

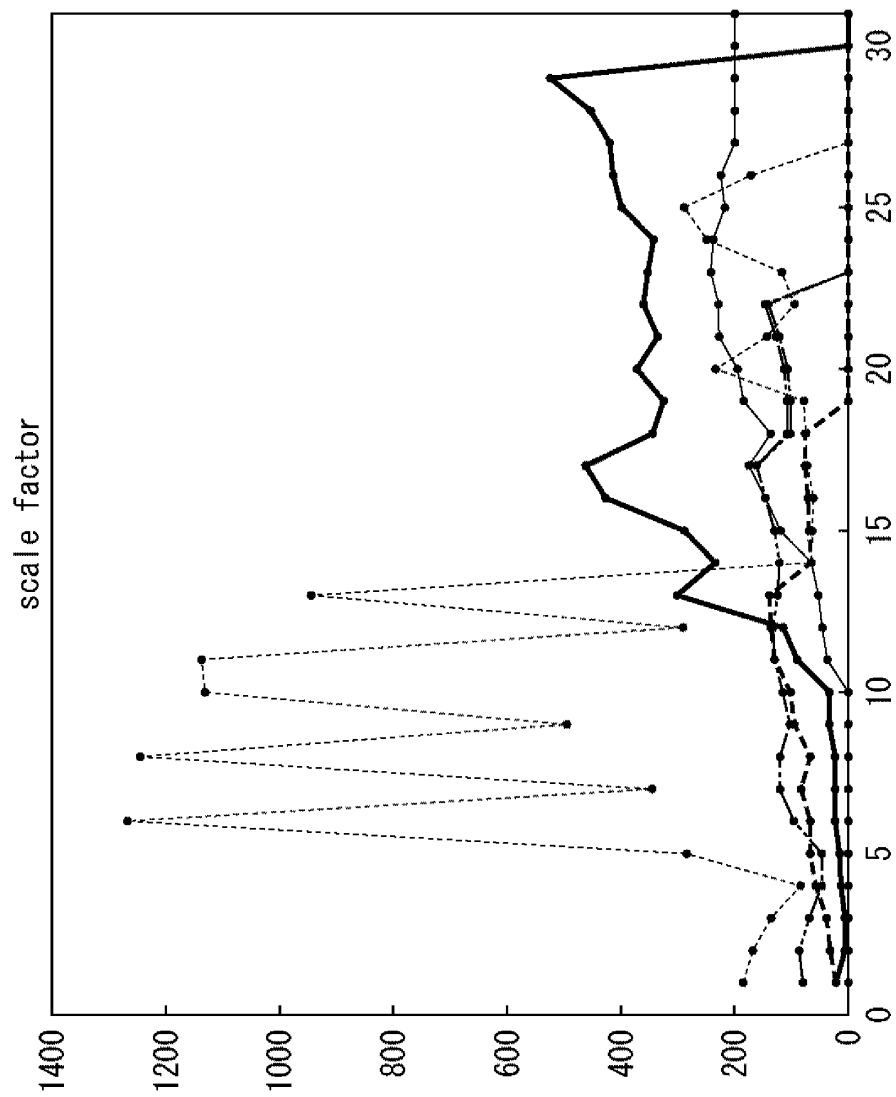
[図26]
FIG. 26



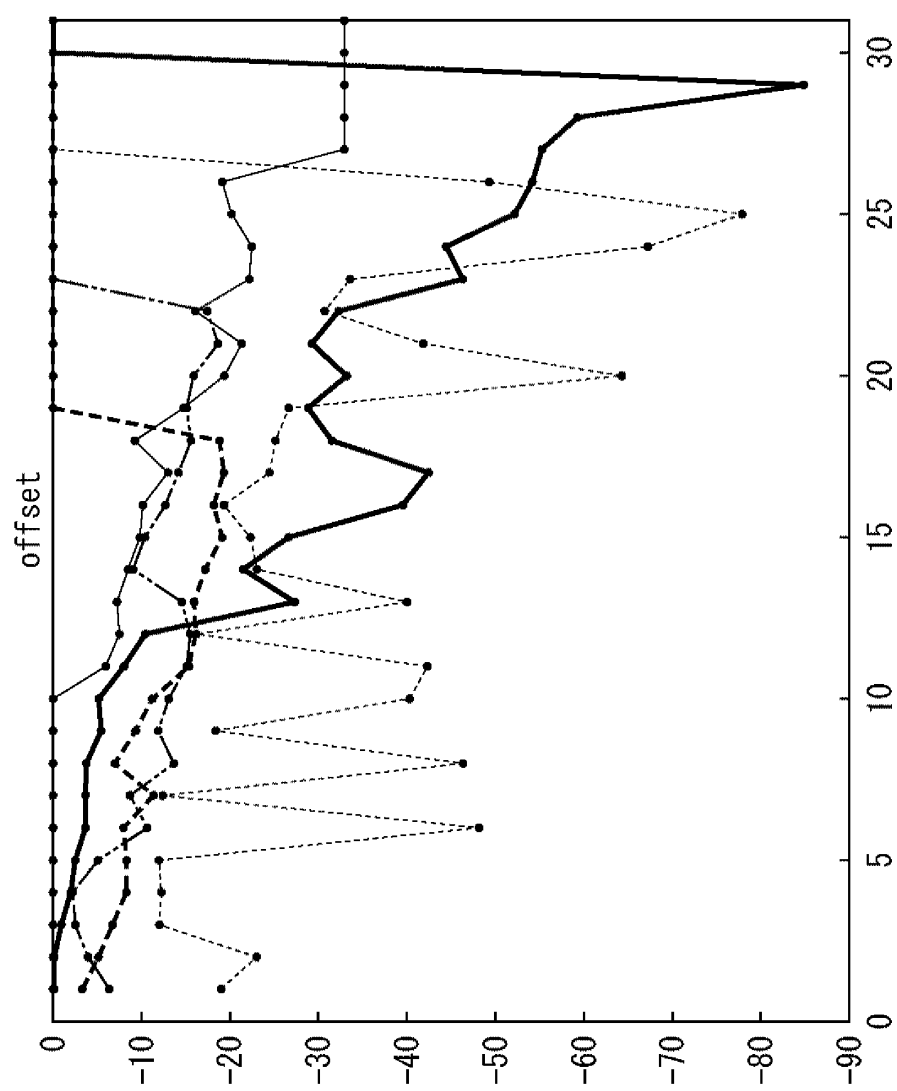
[図27]
FIG. 27

Syntax	Mnemonic	No. of bits	notes
DirectivityReproductionPosInfo() [position_type priority_index Switch (position_type) { Case 0x000 azimuth_interval elevation_interval Case 0x001 uniform_dist_point_count Case 0x010 num_mandatory_point for (i = 0; i < num_mandatory_point; i++) { azimuth_table[i] elevation_table[i] } gain_resolution } }]	 bslbf bslbf bslbf bslbf bslbf bslbf bslbf bslbf bslbf bslbf bslbf	 3 18 10 9 17 17 16 16 8	 座標記録方式 指向性データの優先度 隣接データの方位角の間隔 隣接データの仰角の間隔 一様分布する際のデータポイント数 方位角データ 仰角データ データポイント配置用解像度 (dB)

[図28]
FIG. 28



[図29]
FIG. 29



[図30]
FIG. 30

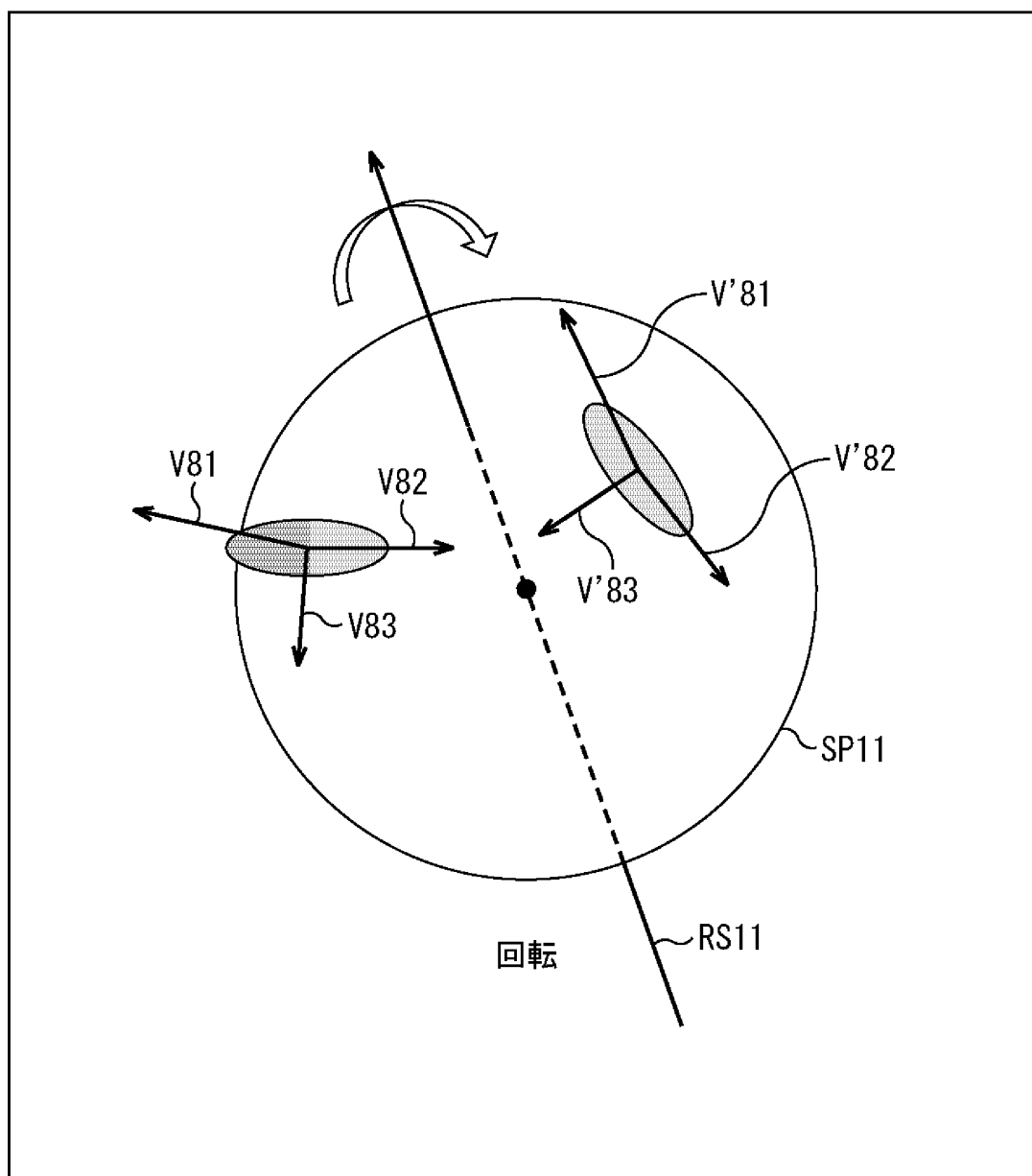
Syntax	No. of bits	Mnemonic
<pre>DirectivityReproduction() { bin_count; for (i=0; i<bin_count; i++) { bin_freq[i]; } band_count; for (j=0; j<band_count; j++) { use_symmetry switch(use_symmetry) case 4: LeftRightVerticalLineSymmetricDir()//上下左右対称 case 3: LeftRightLineSymmetricDir()//左右対称 case 2: VerticalLineSymmetricDir()//上下対称 case 1: SymmetricDir()//任意の対称、回転 case 0: NonSymmetricDir()//対称、回転無し(全vmf/Kent分布を記述) } }</pre>	12 32 8 3	uimsbf bslbf uimsbf uimsbf

[図31]

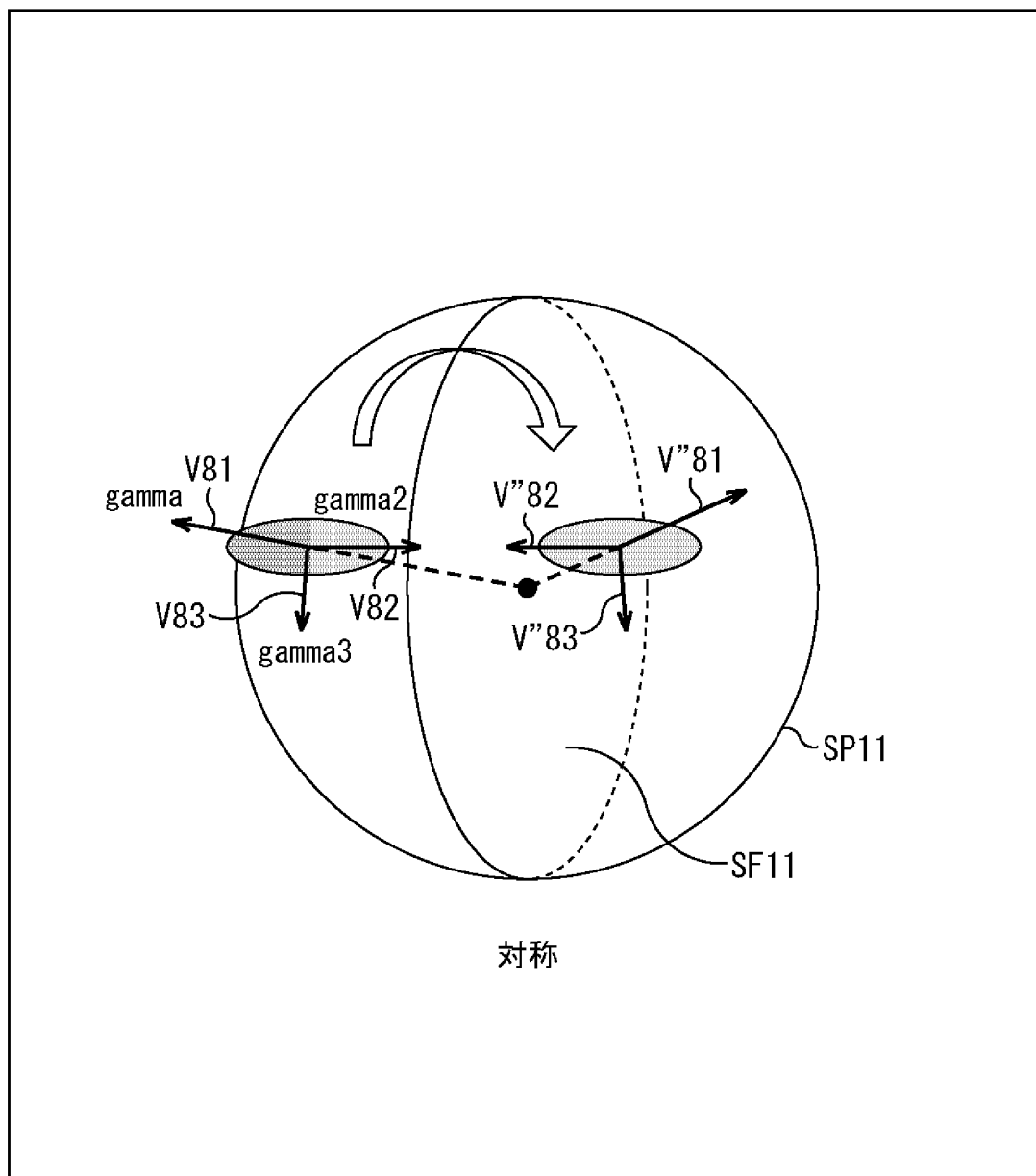
FIG. 31

Syntax	No. of bits	Mnemonic
SymmetricDir() {		
mix_count[j];	12	uimsbf
bin_range_per_band[j];	12	uimsbf
for(k=0; k<mix_count[k]; k++) {		
kappa[j][k];	11	uimsbf
weight[j][k];	10	uimsbf
gamma_x[j][k];	9	uimsbf
gamma_y[j][k];	9	uimsbf
gamma_z[j][k];	9	uimsbf
dist_flag[j][k];	1	uimsbf
if(dist_flag == 1) {		
beta[j][k];	11	uimsbf
gamma2_x[j][k];	9	uimsbf
gamma2_y[j][k];	9	uimsbf
gamma2_z[j][k];	9	uimsbf
gamma3_x[j][k];	9	uimsbf
gamma3_y[j][k];	9	uimsbf
gamma3_z[j][k];	9	uimsbf
} //flag		
sym_operation_count;	2	uimsbf
for(j=0; j<sym_operation_count; j++) {		
sym_operation_flag;	1	uimsbf
Switch(sym_operation_flag)		
Case 1://rotation		
sym_azi;	9	uimsbf
sym_elev;	9	uimsbf
sym_rotation;	9	uimsbf
Default://line symmetry		
sym_yaw;	9	uimsbf
sym_pitch;	9	uimsbf
sym_roll;	9	uimsbf
}		
}		
for(i=0; i<bin_count; i++) {		
scale_factor[i];	7	uimsbf
offset[i];	7	uimsbf
}		
}		

[図32]
FIG. 32



[図33]
FIG. 33



[図34]

FIG. 34

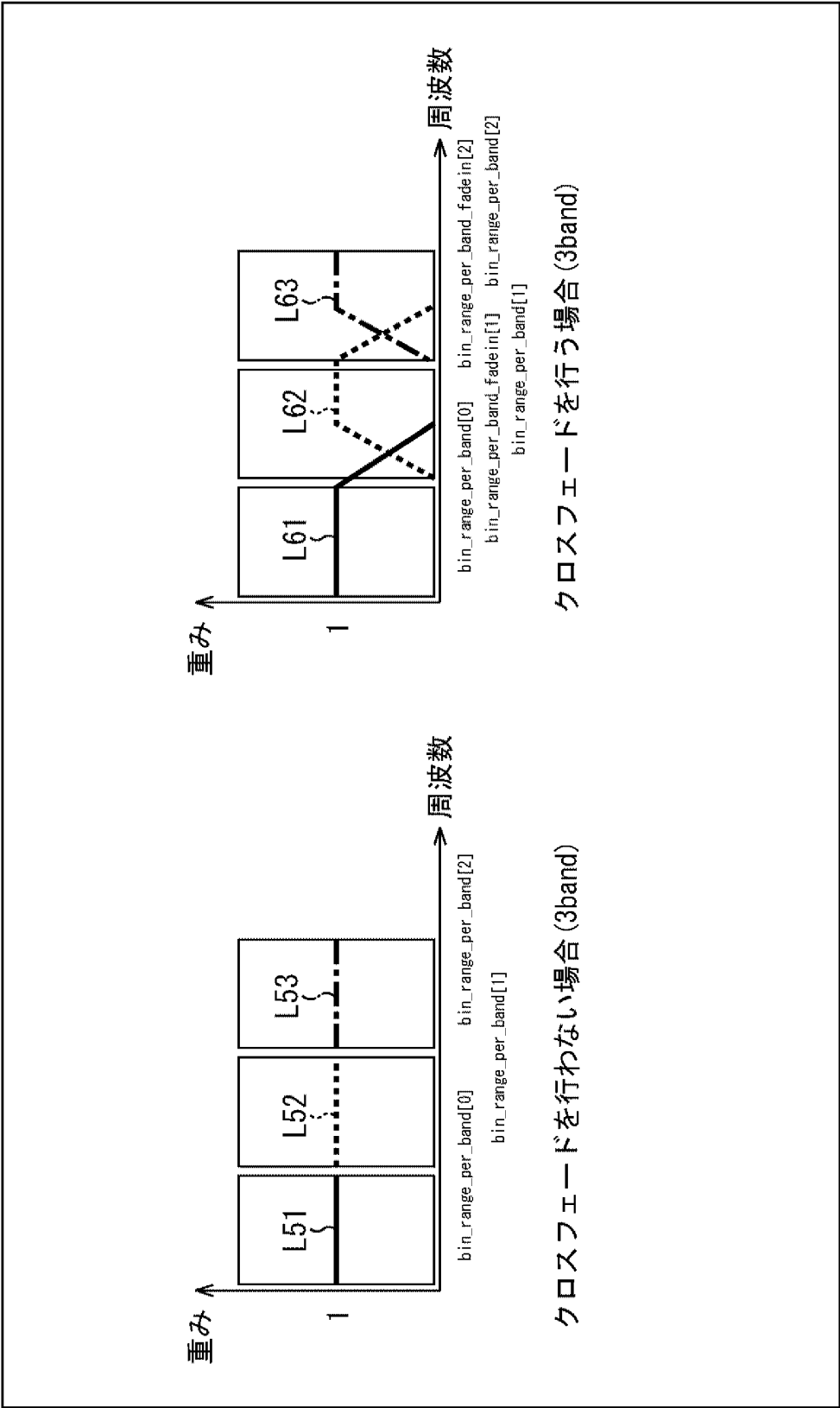
Syntax	No. of bits	Mnemonic
NonSymmetricDir () {		
mix_count[j];	12	uimsbf
bin_range_per_band[j];	12	uimsbf
for (k=0; k < mix_count[k]; k++) {		
kappa[j][k];	11	uimsbf
weight[j][k];	10	uimsbf
gamma_x[j][k];	9	uimsbf
gamma_y[j][k];	9	uimsbf
gamma_z[j][k];	9	uimsbf
dist_flag[j][k];	1	uimsbf
if (dist_flag == 1) {		
beta[j][k];	11	uimsbf
gamma2_x[j][k];	9	uimsbf
gamma2_y[j][k];	9	uimsbf
gamma2_z[j][k];	9	uimsbf
gamma3_x[j][k];	9	uimsbf
gamma3_y[j][k];	9	uimsbf
gamma3_z[j][k];	9	uimsbf
} // flag		
}		
for (i=0; i < bin_count; i++) {		
scale_factor[i];	7	uimsbf
offset[i];	7	uimsbf
}		
}		

[図35]

FIG. 35

Syntax	Mnemonic	Modified No. of bits
DirectivityReproduction() {		
bin_count;	uimsbf	12
for (i=0; i<bin_count; i++) {		
bin_freq[i];	bslbf	18
}		
band_count;	uimsbf	12
for (j=0; j<band_count; j++) {		
mix_count[j];	uimsbf	8
bin_range_per_band[j];	uimsbf	12
fade_flag;	uimsbf	1
if (fade_flag & j>0) {		
bin_range_per_band_fadein[j]	uimsbf	12
}		
for (k=0; k<mix_count[j]; k++) {		
kappa[j][k];	uimsbf	11
weight[j][k];	uimsbf	10
gamma_x[j][k];	uimsbf	9
gamma_y[j][k];	uimsbf	9
gamma_z[j][k];	uimsbf	9
dist_flag	bool	1
if (dist_flag == 1) {		
beta[j][k];	uimsbf	9
gamma2_x[j][k];	uimsbf	9
gamma2_y[j][k];	uimsbf	9
gamma2_z[j][k];	uimsbf	9
gamma3_x[j][k];	uimsbf	9
gamma3_y[j][k];	uimsbf	9
gamma3_z[j][k];	uimsbf	9
} //flag		
} //mix_count		
} //band_count		
for (i=0; i<bin_count; i++) {		
scale_factor[i];	uimsbf	7
offset[i];	uimsbf	7
}		
}		

[図36]
FIG. 36

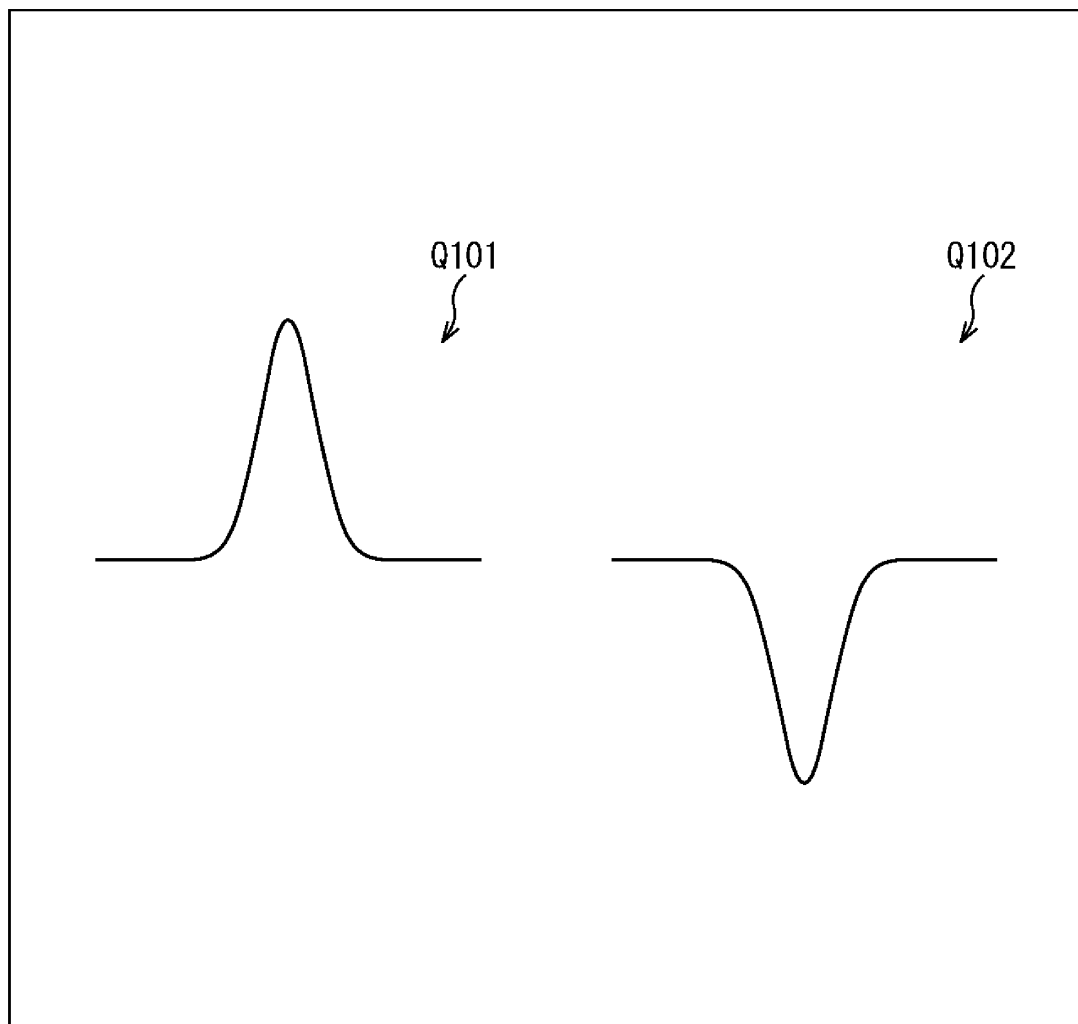


[図37]

FIG. 37

Syntax	No. of bits	Mnemonic
DirectivityReproduction() { bin_count; for (i=0; i<bin_count; i++) { bin_freq[i]; } band_count; for (j=0; j<band_count; j++) { use_symmetry[j] mix_count[j]; bin_range_per_band[j]; if (j>0) { fade_flag; if (fade_flag) { bin_range_per_band_fadein[j] } } start_bin } switch (use_symmetry[j]) case 7: FrontBackVerticalSymmetricDir() //上下前後対称 case 6: FrontBackLeftRightSymmetricDir() //前後左右対称 case 5: FrontBackSymmetricDir() //前後対称 case 4: LeftRightVerticalLineSymmetricDir() //上下左右対称 case 3: LeftRightLineSymmetricDir() //左右対称 case 2: VerticalLineSymmetricDir() //上下対称 case 1: SymmetricDir() //任意の対称、回転 case 0: NonSymmetricDir() //対称、回転無し (全vmf/Kent分布を記述) } //use_symmetry[j] } //j: band_count DynamicRangeForDir() }	12 32 12 3 8 12 1 12 4	uimbsf bslbf uimbsf uimbsf uimbsf uimbsf uimbsf

[図40]
FIG. 40



[図41]

