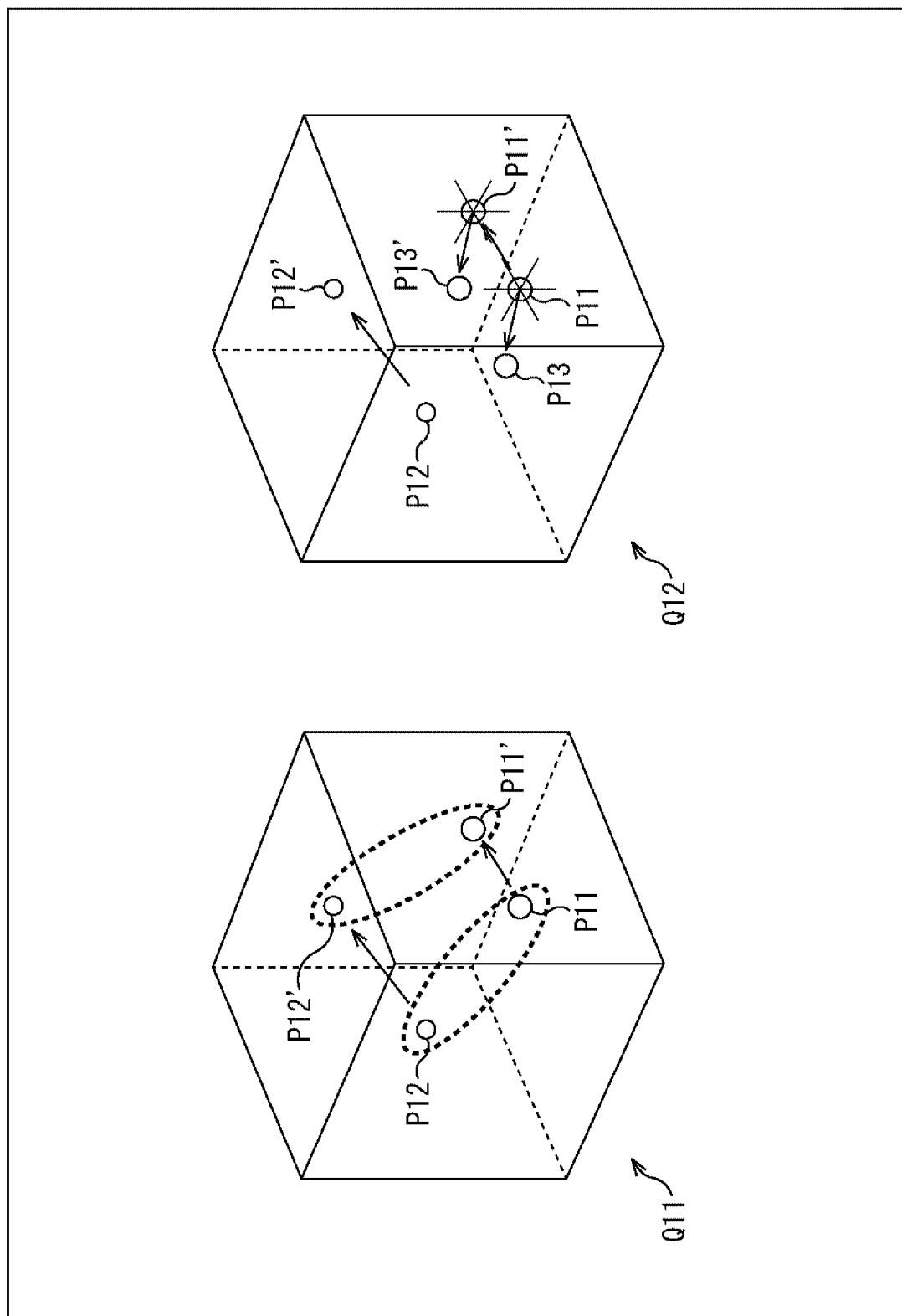
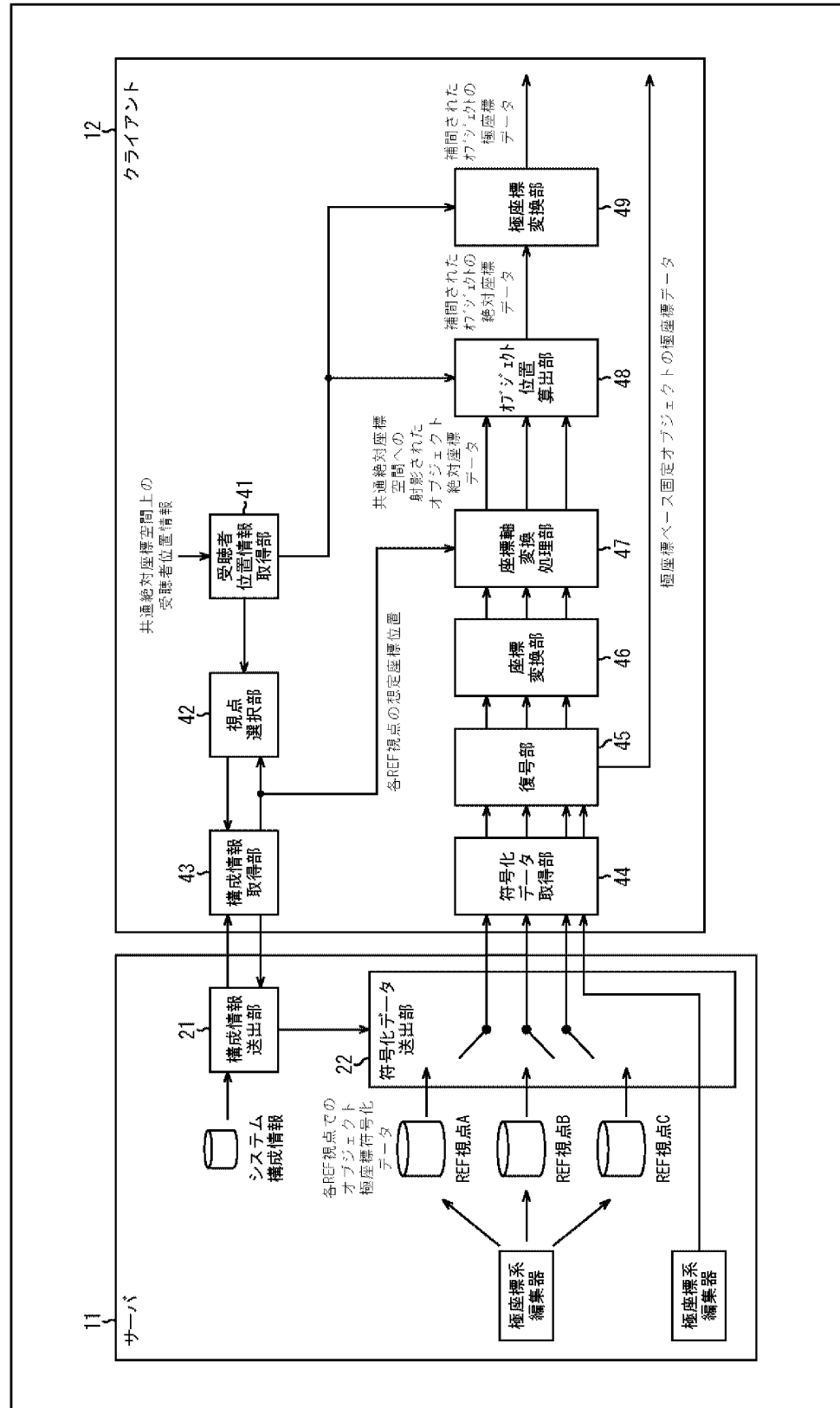


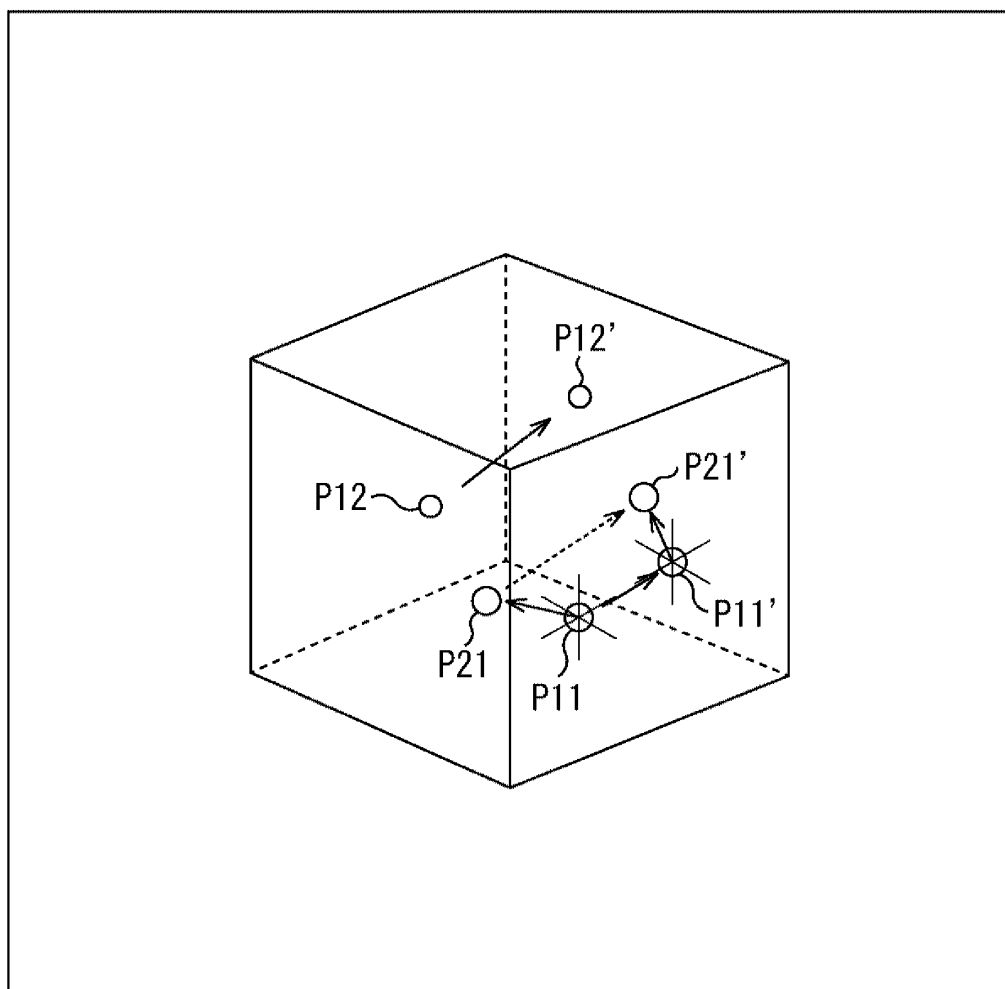
[図1]
FIG. 1



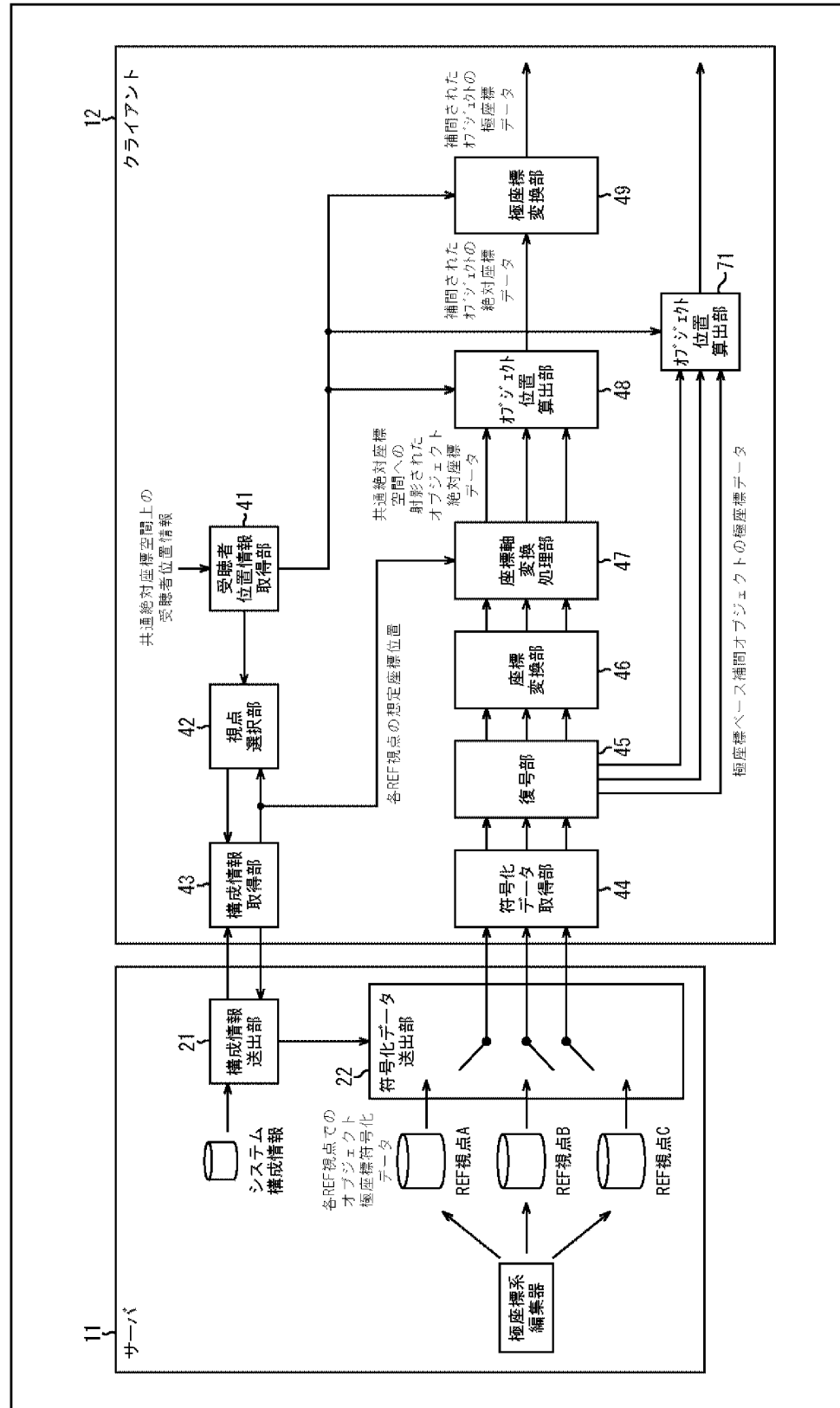
【図2】
FIG. 2



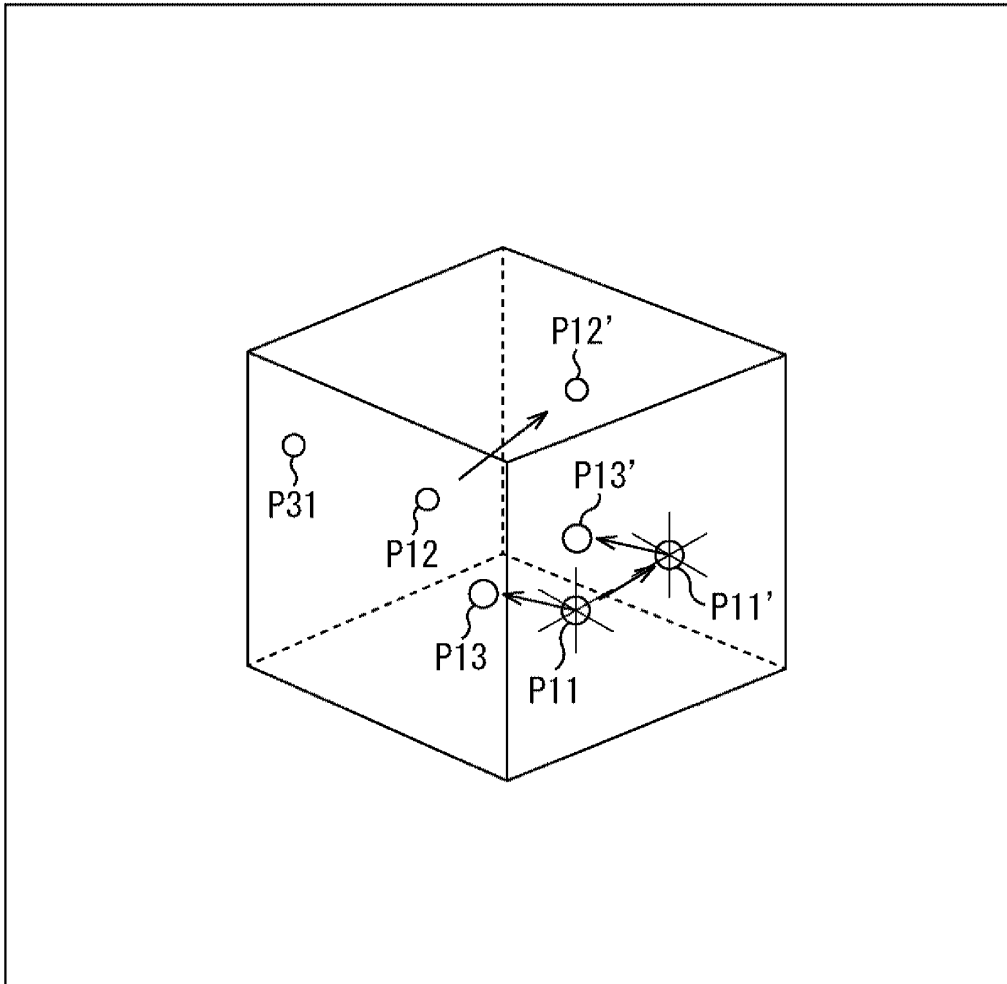
[図3]
FIG. 3



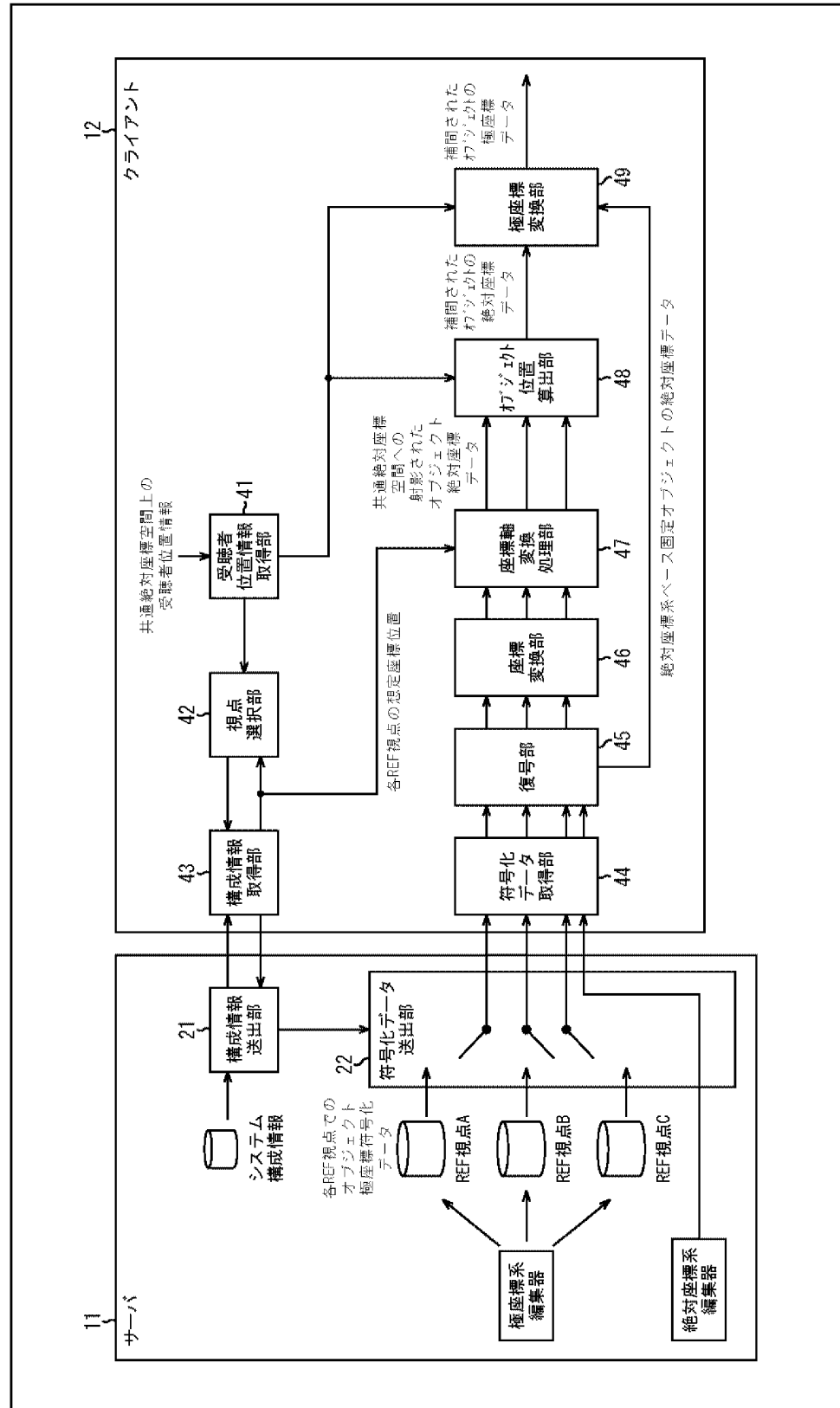
【図4】
FIG. 4

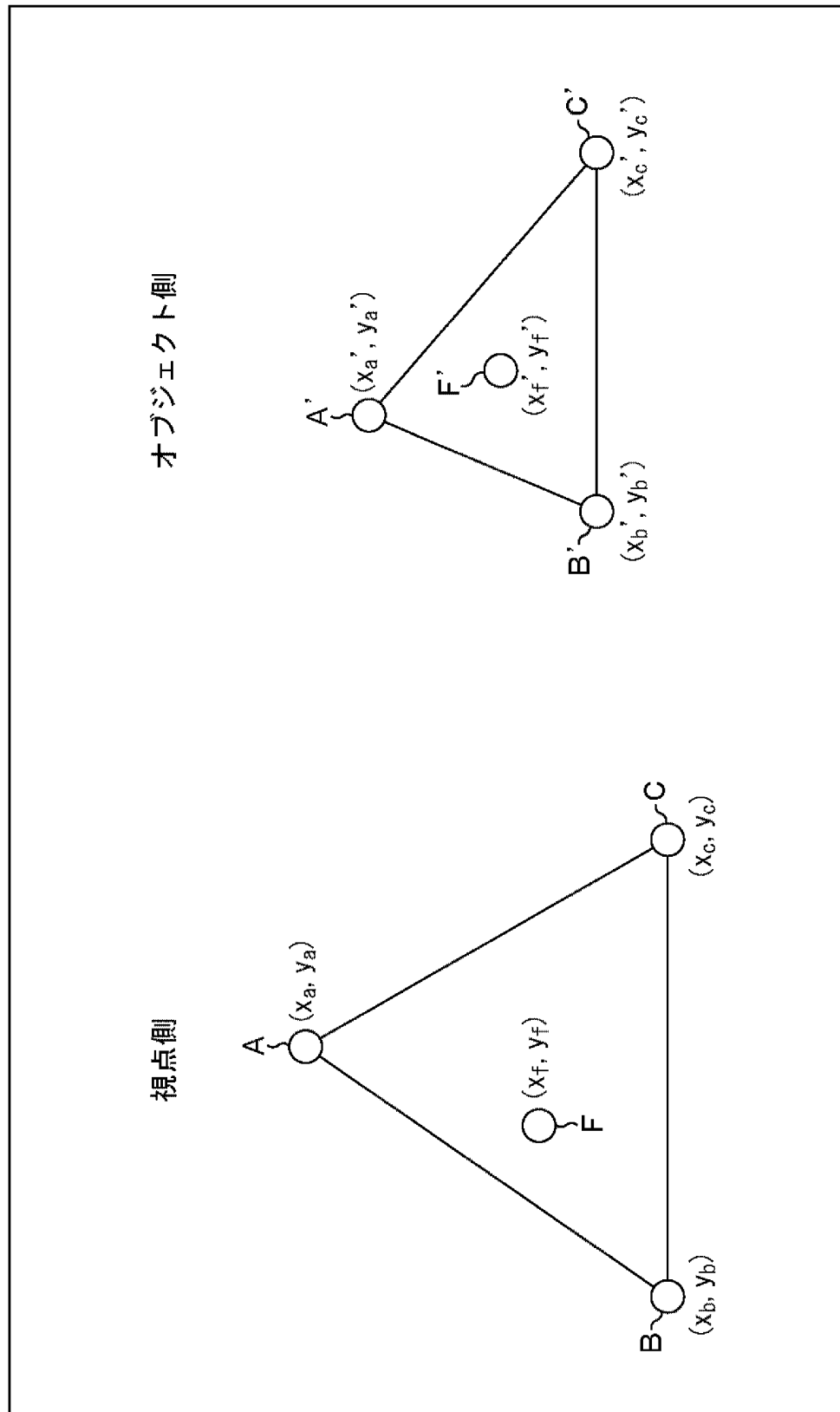


[図5]
FIG. 5

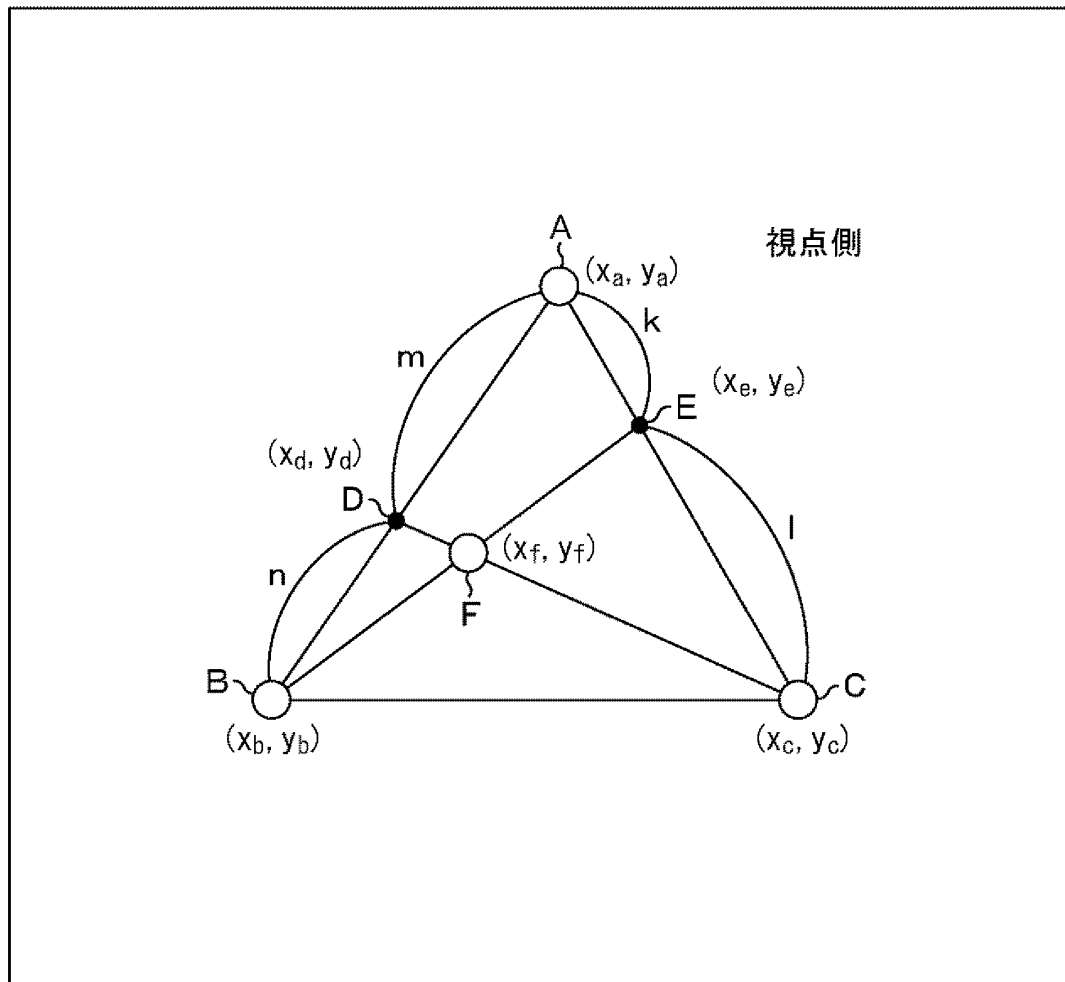


【図6】
FIG. 6

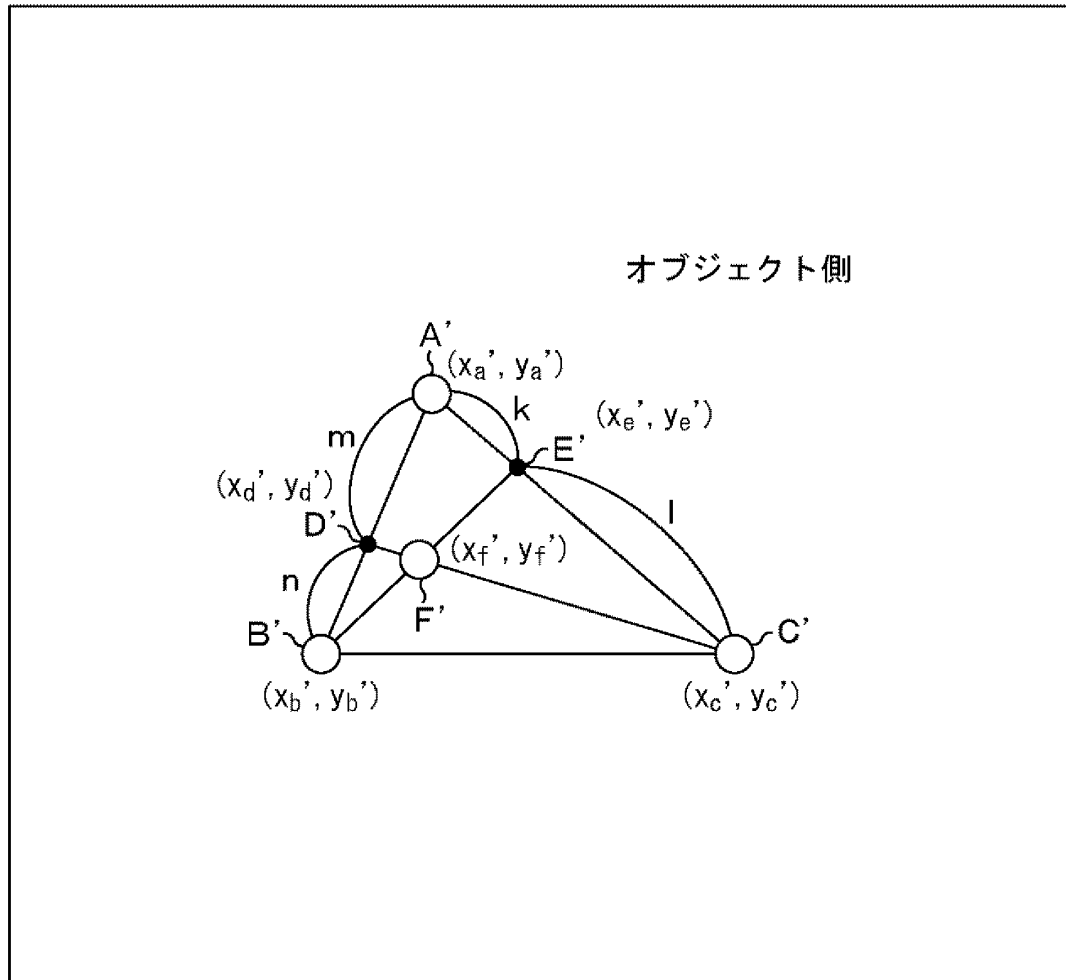


[図7]
FIG. 7

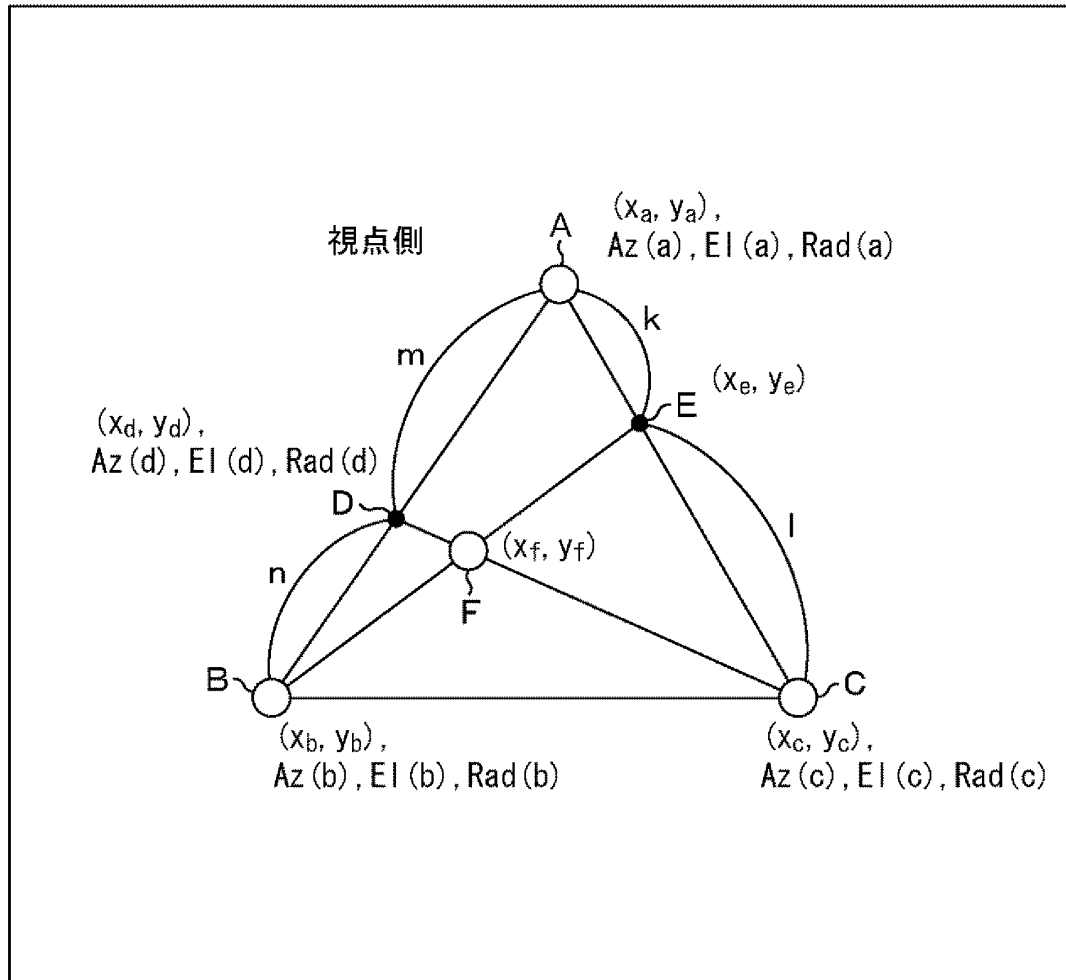
[図8]
FIG. 8



[図9]
FIG. 9



[図10]
FIG. 10



[図11]
FIG. 11

syntax	bit	Comment
fva_structure_info_polar(){		
NumOfObjs;	UINT32	オブジェクト数
NumOfRefViewPoint;	UINT32	リファレンス視点の数
for(i=0; i<NumOfRefViewPoint; i++){		
RefView[i];	FLOAT32	リファレンス視点の想定絶対座標位置(X軸) Float32値
RefView[i];	FLOAT32	リファレンス視点の想定絶対座標位置(Y軸) Float32値
RefViewZ[i];	FLOAT32	リファレンス視点の想定絶対座標位置(Z軸) Float32値
RefView[i];	FLOAT32	各リファレンス視点位置で想定される受聴者の顔の向き(AU方向)
}		
for(i=0; i<NumOfObjs; i++){		
ObjectOverlapMode[i];	UINT32	受聴者とオブジェクトが重なった場合の再生モード
if (ObjectOverlapMode[i] == 2){		
ProhibitRadius	FLOAT32	空間を1.0で正規化した場合のオブジェクトからのリスナーまで距離の正規化値
}		
}		
InterpolationMode	UINT32 (CHAR)	クワイアソフトに許可する補間モード (8bitのビットマスク)
NonInterpolatePolarObjFlag	UINT32	極座標ベース固定オブジェクトが存在するかフラグ 0:存在しない, 1:存在する
if (NonInterpolatePolarObjFlag == 1){		
NumOfObj_MIP0;	UINT32	極座標ベース固定オブジェクト数
}		
NonInterpolateCartesianObjFlag	UINT32	絶対座標ベース固定オブジェクトが存在するかフラグ 0:存在しない, 1:存在する
if (NonInterpolateCartesianObjFlag == 1){		
NumOfObj_MIP0;		絶対座標ベース固定オブジェクト数
}		
InterpolatePolarObjFlag	UINT32	極座標ベース補間オブジェクトが存在するかフラグ 0:存在しない, 1:存在する
if (InterpolatePolarObjFlag == 1){		
NumOfObj_IP0;		
}		
NumOfAnchBytes;	UINT32	拡張情報領域のサイズ(最大255byte)
for(i=0; i<NumOfAnchBytes; i++){		
AnchByteData[i];	UINT32 (CHAR)	拡張情報域/バイトデータ
}		
}		

[図12]
FIG. 12

syntax	bit	Comment
object_metadata_fva_polar {		
fva_structure_info_polar {		構成情報
for (i = 0; i < NumOfRefViewPoint; i++) {		
object_metadata {		絶対座標ベース補間オブジェクト (AAI対応オブジェクト) object_metadata () をREF視点の数だけ伝送する
}		
if (NonInterpolatePolar0bjFlag == 1) {		
object_metadata_nonintpolar {		極座標ベース固定オブジェクト
}		
if (NonInterpolateCartesian0bjFlag == 1) {		
object_metadata_nonintpCarte {		絶対座標ベース固定オブジェクト
}		
if (InterpolatePolar0bjFlag == 1) {		
for (i = 0; i < NumOfRefViewPoint; i++) {		
object_metadata_intpolar {		極座標ベース補間オブジェクト object_metadata () をREF視点の数だけ伝送する
}		
}		
}		

[図13]
FIG. 13

syntax	bit	Comment
object_metadata_fva_polar () {		
fva_structure_info_polar_present	INT32	構成情報存在フラグ
if (fva_structure_info_polar_present == 1) {		
fva_structure_info_polar ();		構成情報
}		
for (i = 0; i < 3; i++) {		
object_metadata ();		絶対座標ベース補間オブジェクト (AAI 対応オブジェクト) object_metadata () を REF 視点の数だけ伝送する
}		
if (NonInterpolatePolarObjFlag == 1) {		
object_metadata_nonintpolar ();		極座標ベース固定オブジェクト
}		
if (NonInterpolateCartesianObjFlag == 1) {		
object_metadata_nonintpCarte ();		絶対座標ベース固定オブジェクト
}		
if (InterpolatePolarObjFlag == 1) {		
for (i = 0; i < 3; i++) {		
object_metadata_intpPolar ();		極座標ベース補間オブジェクト object_metadata () を REF 視点の数だけ伝送する
}		
}		
}		

[図14]
FIG. 14

syntax	bit	Comment
object_metadata_nonintpPolar() {		
for (i = 0; i < NumOf0bjs_NIP0; i++) {		
PosAzi[i];	Float32	Azimuth: 水平方向の角度
PosEle[i];	Float32	Elevation: 垂直方向の角度
PosRad[i];	Float32	Radius : 半径
Gain[i]	Float32	Gain: ゲイン量
}		
}		

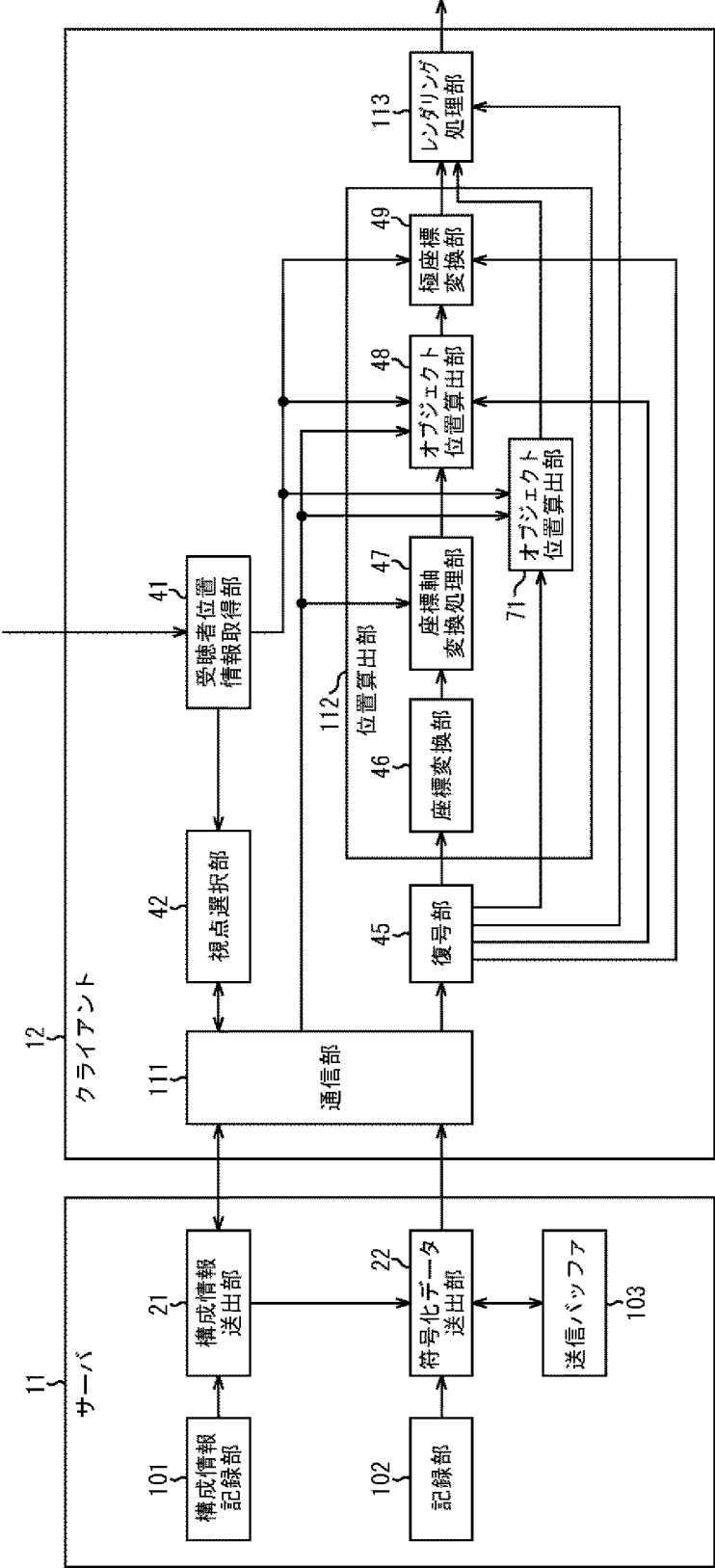
[図15]
FIG. 15

syntax	bit	Comment
object_metadata_intpPolar () {		
for (i = 0; i < NumOf0bjs_IP0; i++) {		
PosAzi[i];	Float32	Azimuth: 水平方向の角度
PosEle[i];	Float32	Elevation: 垂直方向の角度
PosRad[i];	Float32	Radius : 半径
Gain[i]	Float32	Gain: ゲイン量
}		
}		

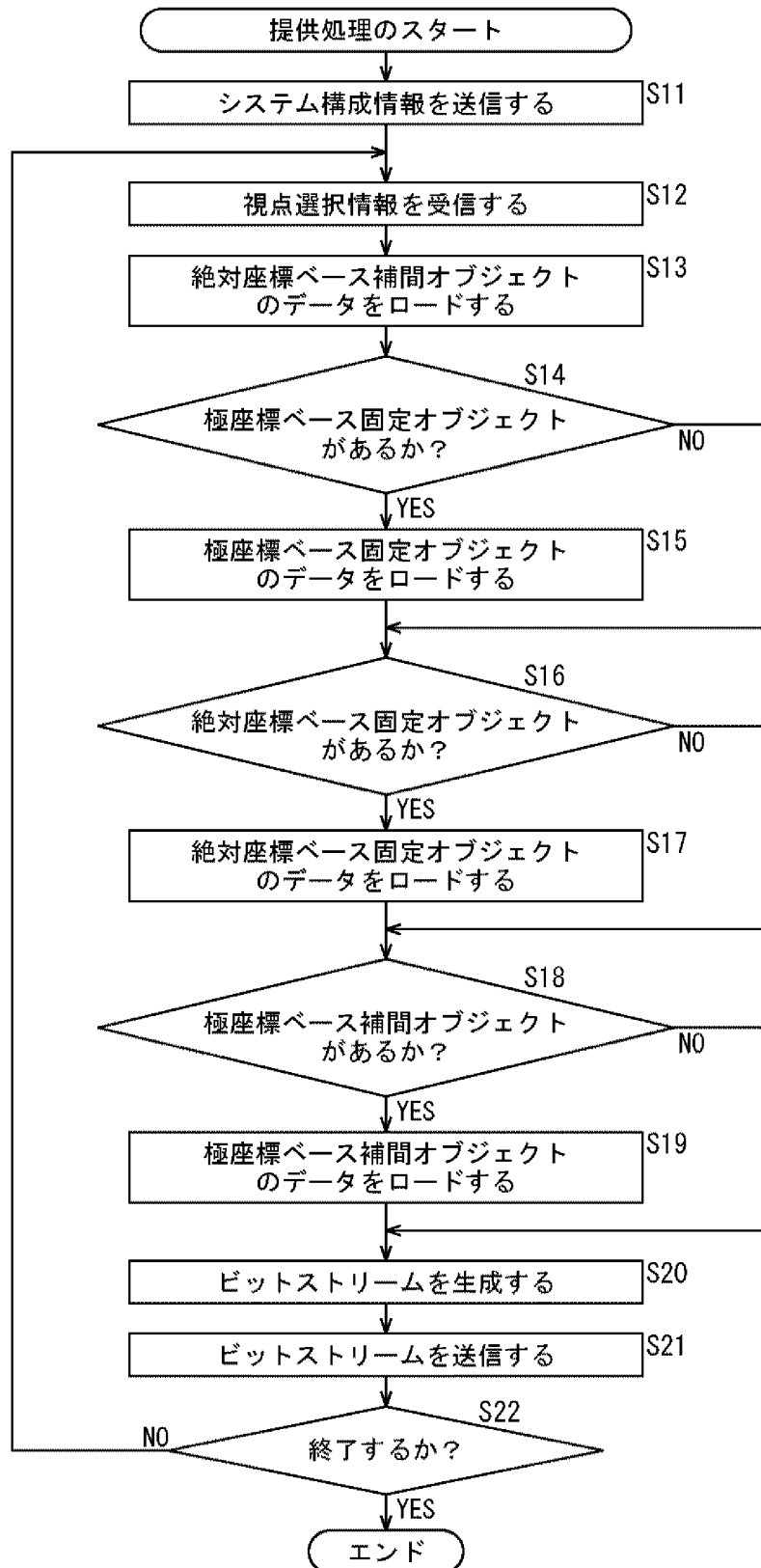
[図16]
FIG. 16

syntax	bit	Comment
object_metadata_nonintpCarte() {		
for (i = 0; i < NumOf0bjs_NICO; i++) {		
PosX[i];	Float32	X座標の位置情報
PosY[i];	Float32	Y座標の位置情報
PosZ[i];	Float32	Z座標の位置情報
Gain[i]	Float32	Gain:ゲイン量
}		
}		

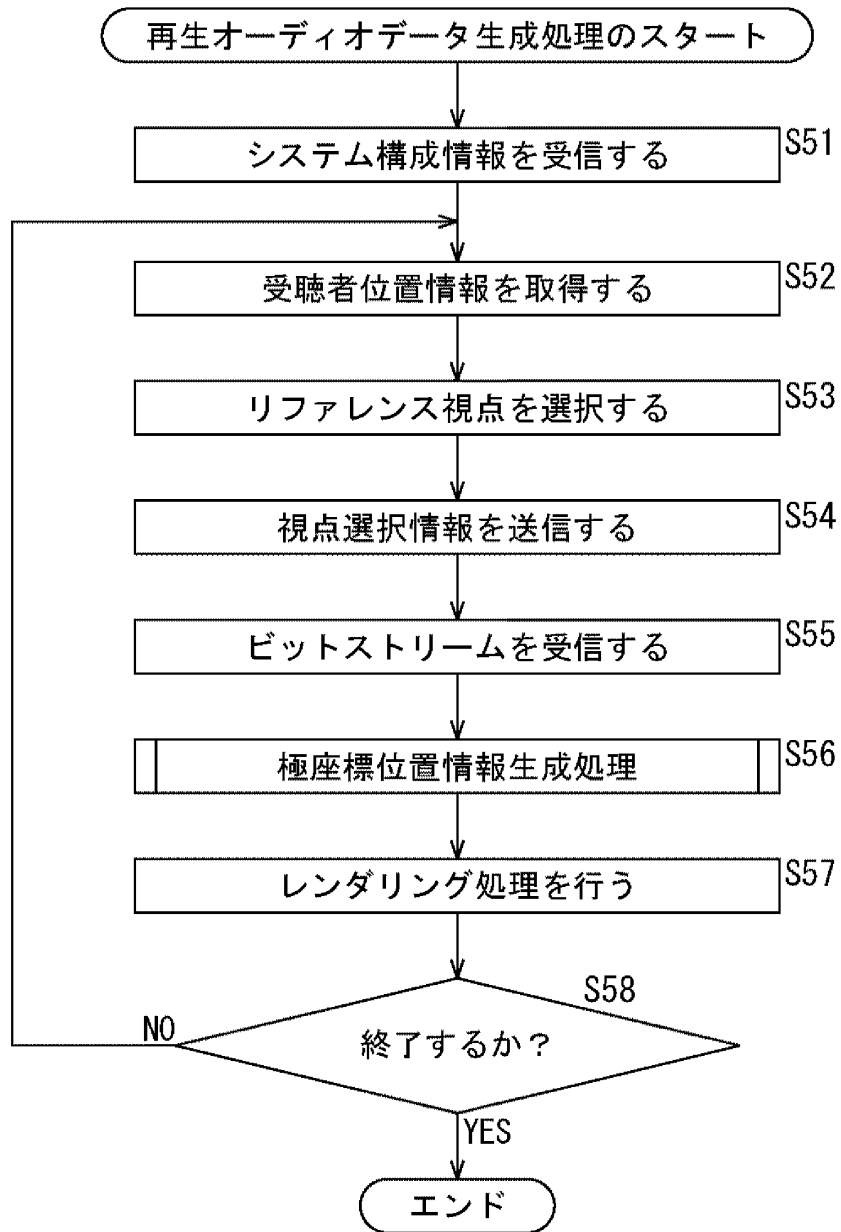
[図17]
FIG. 17



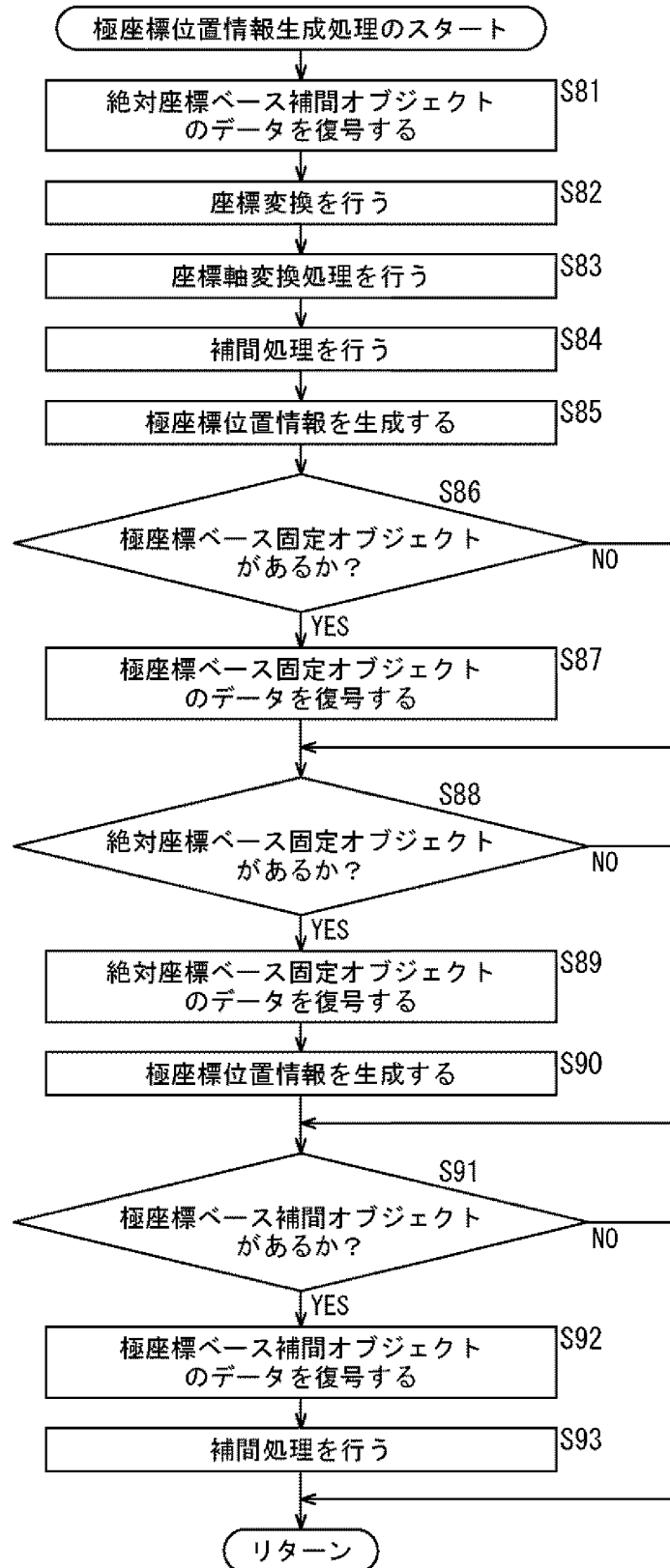
[図18]
FIG. 18



【図19】
FIG. 19



[図20]
FIG. 20



【図21】
FIG. 21

