

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第4679431号
(P4679431)

(45) 発行日 平成23年4月27日 (2011. 4. 27)

(24) 登録日 平成23年2月10日 (2011. 2. 10)

(51) Int. Cl.		F I			
G 1 O H	1/00	(2006. 01)	G 1 O H	1/00	Z
G 1 O H	1/34	(2006. 01)	G 1 O H	1/34	
G 1 O H	1/053	(2006. 01)	G 1 O H	1/053	C

請求項の数 5 (全 18 頁)

(21) 出願番号	特願2006-124830 (P2006-124830)	(73) 特許権者	000233778
(22) 出願日	平成18年4月28日 (2006. 4. 28)		任天堂株式会社
(65) 公開番号	特開2007-298598 (P2007-298598A)		京都府京都市南区上鳥羽鉾立町 1 1 番地 1
(43) 公開日	平成19年11月15日 (2007. 11. 15)	(74) 代理人	100090181
審査請求日	平成21年3月24日 (2009. 3. 24)		弁理士 山田 義人
		(72) 発明者	稲垣 陽司
			京都府京都市南区上鳥羽鉾立町 1 1 番地 1
			任天堂株式会社内
		審査官	鈴木 聡一郎

最終頁に続く

(54) 【発明の名称】 音出力制御プログラムおよび音出力制御装置

(57) 【特許請求の範囲】

【請求項 1】

操作手段の操作に応じて音を出力手段から出力する音出力制御装置の音出力制御プログラムであって、

前記操作手段は少なくとも2つの互いに直交する軸方向の加速度を検出する加速度センサを備えていて、

前記音出力制御装置のプロセッサに、

前記加速度センサによって検出された加速度を取得する取得ステップ、

前記取得ステップによって取得された2軸の加速度の和を算出する算出ステップ、

前記算出ステップによって算出された加速度の和の変化に応じて、前記操作手段を振る操作における操作位置を特定する位置特定ステップ、

前記位置特定ステップによって特定された操作位置に応じて、少なくとも出力する音を決定する音決定ステップ、

前記音決定ステップによって決定された音を出力するための制御信号を生成する音制御ステップ、および

前記音制御ステップによって生成された制御信号に基づいて前記出力手段から音を出力させる音出力ステップを実行させる、音出力制御プログラム。

【請求項 2】

前記位置特定ステップは、前記算出ステップによって算出された加速度の和の変化が、前記操作位置に対応して段階的に複数設定されている閾値のいずれかと所定の関係になっ

10

20

たか否かを判別し、前記加速度の和の変化が前記複数設定されている閾値のいずれかと所定の関係になったと判別されるとき、当該閾値に対応して操作位置を特定する、請求項 1 記載の音出力制御プログラム。

【請求項 3】

前記取得ステップ、前記算出ステップ、前記位置特定ステップ、前記音決定ステップ、前記音制御ステップおよび前記音出力ステップを繰り返す繰り返し制御ステップを、前記プロセッサにさらに実行させる、請求項 2 記載の音出力制御プログラム。

【請求項 4】

前記操作手段は、音を出力するか否かを指示するための発音指示手段をさらに備えていて、

前記音制御ステップは、前記発音指示手段からの指示に応じて音の出力の有無を制御する、請求項 1 ないし 3 のいずれかに記載の音出力制御プログラム。

【請求項 5】

少なくとも 2 つの互いに直交する軸方向の加速度を検出する加速度センサを有する操作手段の操作に応じて音を出力手段から出力する音出力制御装置であって、

前記加速度センサによって検出された加速度を取得する取得手段、

前記取得手段によって取得された 2 軸の加速度の和を算出する算出手段、

前記算出手段によって算出された加速度の和の変化に応じて、前記操作手段を振る操作における操作位置を特定する位置特定手段、

前記位置特定手段によって特定された操作位置に応じて、少なくとも出力する音を決定する音決定手段、

前記音決定手段によって決定された音を出力するための制御信号を生成する音制御手段、および

前記音制御手段によって生成された制御信号に基づいて前記出力手段から音を出力させる音出力手段を備える、音出力制御装置。

【発明の詳細な説明】

【技術分野】

【0001】

この発明は、音出力制御プログラムおよび音出力制御装置に関し、特にたとえば、操作装置を振る操作に応じてサウンドを出力することによって模擬的な演奏を行うための音出力制御プログラムおよび音出力制御装置に関する。

【背景技術】

【0002】

操作装置を振る動作に応じて楽器音出力される模擬的な演奏装置の一例が、たとえば特許文献 1 および 2 に開示されている。特許文献 1 の技術では、手のひらに衝撃センサが取り付けられ、スティックを振った際に、衝撃センサとスティックとが衝突することによって生ずる衝撃が検出されて、その検出に応じてドラムなどの楽器音出力される。

【0003】

また、特許文献 2 の技術では、操作部材を振る動作の角度に応じて異なる制御信号が発生される。具体的には、水銀と複数の接点とが利用され、操作部材の傾きに応じて水銀が異なる接点に達することに基づいて振り上げ角度が検出され、その検出角度に応じた音高の楽器音出力される。

【特許文献 1】特開 2000-330567 号公報

【特許文献 2】特開昭 63-192096 号公報

【発明の開示】

【発明が解決しようとする課題】

【0004】

特許文献 1 の技術では、スティックの振り操作の衝撃が検出されるので、打楽器以外の模擬的な楽器演奏を実現するのは困難であった。たとえばギターストローク演奏のような手首のスナップの効かされる演奏にこの技術を適用しようとしても、演奏の動作がギター

10

20

30

40

50

ストローク演奏とは全く異なったものになってしまう。また、特許文献2の技術では、操作部材の振り上げ角度が水銀スイッチで検出されるため、コストと使用部材の危険性を考慮すると、あまり現実的な技術とは言えない。さらに、ギターストローク演奏のような動作の状態を適切に検出することは困難であると思われる。

【0005】

それゆえに、この発明の主たる目的は、コントローラを振るといった操作によって模擬的なストローク演奏を行うことを可能にする、音出力制御プログラムおよび音出力制御装置を提供することである。

【課題を解決するための手段】

【0006】

第1の発明（請求項1の発明）は、操作手段の操作に応じて音を出力手段から出力する音出力制御装置の音出力制御プログラムである。操作手段は少なくとも2つの互いに直交する軸方向の加速度を検出する加速度センサを備えている。この音出力制御プログラムは、音出力制御装置のプロセッサに、取得ステップ、算出ステップ、位置特定ステップ、音決定ステップ、音制御ステップ、および音出力ステップを実行させる。取得ステップは、加速度センサによって検出された加速度を取得する。算出ステップは、取得ステップによって取得された2軸の加速度の和を算出する。位置特定ステップは、算出ステップによって算出された加速度の和に応じて、操作手段を振る操作における操作位置を特定する。音決定ステップは、位置特定ステップによって特定された操作位置に応じて、少なくとも出力する音を決定する。音制御ステップは、音決定ステップによって決定された音を出力するための制御信号を生成する。音出力ステップは、音制御ステップによって生成された制御信号に基づいて出力手段から音を出力させる。

【0007】

請求項1の発明では、音出力制御プログラムは、ユーザによる操作手段（14：実施例で相当する参照符号。以下同じ。）の操作に応じて音を出力手段（24）から出力するためのものであり、音出力制御装置（10）のプロセッサ（26）に実行させる以下のステップを含む。ただし、操作手段には、少なくとも2つの互いに直交する軸方向の加速度を検出する加速度センサ（60）が設けられる。ユーザは、この操作手段を把持して、振るような操作、たとえばギターのストローク奏法のような操作を行う。取得ステップ（S3）では、加速度センサによって検出された加速度が取得され、算出ステップ（S25）では、2軸の加速度の和が算出される。この加速度の和は、ユーザによる操作手段に対するストローク操作の状態を示している。位置特定ステップ（S27 - S37）では、加速度の和の変化に応じて、操作手段を振る操作における操作位置を特定する。音決定ステップ（S41）では、操作位置に応じて、少なくとも出力する音を決定する。音制御ステップ（S39、S41）では、決定された音を出力するための制御信号が生成される。音出力ステップ（S43）では、制御信号に基づいて出力手段から音が出力される。このようにして、ユーザのストローク操作の状態に応じた音出力される。

【0008】

請求項1の発明によれば、たとえばギターのストローク演奏などのように単純なポイントングでは再現できないような演奏に関して、2軸の加速度の和によって示される操作の状態に応じて音出力を行うことができる。したがって、模擬的なストローク演奏を実現できる。

【0009】

請求項2の発明は、請求項1の発明に従属し、位置特定ステップは、算出ステップによって算出された加速度の和の変化が、操作位置に対応して段階的に複数設定されている閾値のいずれかと所定の関係になったか否かを判別し、加速度の和の変化が複数設定されている閾値のいずれかと所定の関係になったと判別されるとき、当該閾値に対応して操作位置を特定する。

【0010】

請求項2の発明では、位置特定ステップは、算出された加速度の和の変化が、操作位置

10

20

30

40

50

に対応して段階的に複数設定されている閾値のいずれかと所定の関係になったか否かを判別し（S33、S37）、加速度の和の変化が複数設定されている閾値のいずれかと所定の関係になったと判別されるとき（S33またはS37で“YES”）、当該閾値に対応して操作位置を特定する。後述の実施例では複数の閾値は、ギターのような弦楽器の各弦に対応付けられて弦閾値テーブルデータとして記憶されている。所定の関係とは、加速度の和が閾値を越えて変化したことであり、具体的には、当該閾値に対応付けられた弦が弾かれたことを意味する。

【0011】

請求項2の発明によれば、更に、加速度の和が予め定められた複数の閾値のそれぞれを越えたか否かに応じて音出力が制御されるようにしたので、たとえばギターの6つの弦のそれぞれに対応する閾値を設定しておくことによって、操作に応じて複数の音が実際のギターのように時間差を伴って鳴るような状況を模擬的に作り出すことができる。

10

【0012】

請求項3の発明は、請求項2の発明に従属し、取得ステップ、算出ステップ、位置特定ステップ、音決定ステップ、音制御ステップおよび音出力ステップを繰り返す繰り返し制御ステップを、プロセッサにさらに実行させる。

【0013】

請求項3の発明では、繰り返し制御ステップ（S3、S5（S25-S43）、S9）では、取得ステップ、算出ステップ、位置特定ステップ、音決定ステップ音制御ステップおよび音出力ステップを繰り返す。つまり、1回の操作手段を振る操作で様々な音が段階的に出力される。したがって、たとえばギターの6つの弦にそれぞれ異なる音色を設定しておけば、たとえば閾値ごとに異なる音色がなるようにできるので、更に実際のコードに応じた模擬的なギター演奏を実現することができる。

20

【0014】

請求項4の発明は、請求項1ないし3のいずれかの発明に従属し、操作手段は、音を出力するか否かを指示するための発音指示手段をさらに備えている。音制御ステップは、発音指示手段からの指示に応じて音の出力の有無を制御する。

【0015】

請求項4の発明では、操作手段には発音指示手段（52d）が設けられる。ユーザの発音指示手段に対する操作に応じて音を出力するか否かが指示される。音制御ステップでは、当該指示に応じて音の出力の有無が制御される。したがって、更に、実際にスピーカなどから音を鳴らすのか鳴らさないのかを制御するようにしたので、加速度の和の変化に応じた音出力に対して更に発音の有無を調整することが可能になり、たとえば、模擬的なギター演奏において単独の弦についてのみ当該サウンドを出力することができる。

30

【0016】

第2の発明（請求項5の発明）は、少なくとも2つの互いに直交する軸方向の加速度を検出する加速度センサを有する操作手段の操作に応じて音を出力手段から出力する音出力制御装置である。この音出力制御装置は、取得手段、算出手段、位置特定手段、音決定手段、音制御手段、および音出力手段を備える。取得手段は、加速度センサによって検出された加速度を取得する。算出手段は、取得手段によって取得された2軸の加速度の和を算出する。位置特定手段は、算出手段によって算出された加速度の和に応じて、操作手段を振る操作における操作位置を特定する。音決定手段は、位置特定手段によって特定された操作位置に応じて、少なくとも出力する音を決定する。音制御手段は、音決定手段によって決定された音を出力するための制御信号を生成する。音出力手段は、音制御手段によって生成された制御信号に基づいて出力手段から音を出力させる。

40

【0017】

請求項5の発明は、上述の第1の発明に対応する音出力制御装置であり、上述の第1の発明と同様の効果を奏する。

【発明の効果】

【0018】

50

この発明によれば、操作手段に設けた加速度センサによって検出される2軸の加速度の和の変化に応じて音が制御されるようにしたので、ギターのストローク演奏のような手首のスナップを効かせた操作による演奏を模擬的に行うことができる。単純なポインティングでは再現できないような演奏に対して、操作手段の振りなど操作の状態に応じて音の出力を制御できるので、今までに無い娯楽的な音出力が可能な装置を提供することができる。

【0019】

この発明の上述の目的、その他の目的、特徴および利点は、図面を参照して行う以下の実施例の詳細な説明から一層明らかとなる。

【発明を実施するための最良の形態】

【0020】

図1を参照して、この実施例の音出力制御装置10は、一例としてゲームシステムの形態で実現される。ゲームシステム10はゲーム装置12およびコントローラ14を含む。ゲーム装置12は、据置型ゲーム装置であり、家庭用テレビジョン受像機のようなディスプレイないしモニタ16にケーブルを介して接続される。コントローラ14は、プレイヤーないしユーザによって操作される操作装置であり、ゲーム装置12に操作データを与える。

【0021】

ゲーム装置12には、接続端子を介して受信ユニット18が接続される。受信ユニット18は、コントローラ14から無線送信される操作データを受信する。具体的には、コントローラ14は、たとえばBluetooth(ブルートゥース)(登録商標)のような無線通信技術を用いて、受信ユニット18が接続されたゲーム装置12へ操作データを送信する。

【0022】

また、ゲーム装置12には、当該ゲーム装置12に対して交換可能に用いられる情報記憶媒体の一例である光ディスク20が脱着される。ゲーム装置12の上部主面には、当該ゲーム装置12の電源ON/OFFスイッチ、ゲーム処理のリセットスイッチ、およびゲーム装置12上部の蓋を開くOPENスイッチが設けられている。ここで、プレイヤーがOPENスイッチを押下することによって上記蓋が開き、光ディスク20の脱着が可能となる。また、ゲーム装置12には、外部メモリカード22が必要に応じて着脱自在に装着される。当該メモリカード22に搭載されるフラッシュメモリ等にはセーブデータ等が記憶される。

【0023】

ゲーム装置12は、光ディスク20に記憶されたゲームプログラムを実行することによって、その結果をゲーム画像としてモニタ16に表示する。なお、ゲーム装置12は、外部メモリカード22に記憶されたセーブデータを用いて、過去に実行されたゲーム状態を再現して、当該ゲーム画像をモニタ16に表示することもできる。また、モニタ16に備えられるスピーカ24(図2参照)からはゲームサウンド(この実施例ではユーザの演奏に応じたサウンド)が出力される。そして、プレイヤーは、コントローラ14を操作することによって、仮想ゲームをプレイする(この実施例では模擬演奏を行う)。

【0024】

図2にはゲーム装置12の電気的構成の一例が示される。ゲーム装置12は、各種プログラムを実行するたとえばリスク(RISC)CPU(セントラルプロセッシングユニット)26を含む。CPU26は、図示しないブートROMに記憶された起動プログラムを実行し、メインメモリ28等のメモリの初期化等を行った後、光ディスク20に記憶されているゲームプログラム(この実施例では音出力制御プログラム)およびデータをロードし、当該ゲームプログラムに従ってゲーム処理を行う。

【0025】

CPU26には、メモリコントローラ30を介して、GPU(Graphics Processing Unit)32、メインメモリ28、DSP(Digital Signal Processor)34、およびARA

10

20

30

40

50

M (Audio RAM) 36 が接続される。また、メモリコントローラ 30 には、所定のバスを介して、コントローラ I / F (インターフェース) 38、ビデオ I / F 40、外部メモリ I / F 42、オーディオ I / F 44、およびディスク I / F 46 が接続され、それぞれ受信ユニット 18、モニタ 16、外部メモリカード 22、スピーカ 24、およびディスクドライブ 48 が接続されている。

【0026】

GPU 32 は、CPU 26 の命令に従って画像処理を行う。GPU 32 は、たとえば、3D グラフィックスの表示に必要な計算処理を行う半導体チップで構成される。GPU 32 は、図示しない画像処理専用のメモリやメインメモリ 28 の一部の記憶領域を用いて画像処理を行う。GPU 32 は、表示すべきゲーム画像データやムービー映像を生成し、適宜メモリコントローラ 30 およびビデオ I / F 40 を介してモニタ 16 に出力する。

10

【0027】

メインメモリ 28 は、CPU 26 によって使用される記憶領域であり、CPU 26 の処理に必要なゲームプログラムおよびデータを適宜記憶する。たとえば、メインメモリ 28 は、CPU 26 によって光ディスク 20 から読み出されたゲームプログラムや各種データ等を記憶する。

【0028】

DSP 34 はサウンドプロセッサとして機能し、DSP 34 にはサウンドデータ等を記憶するための RAM 36 が接続される。RAM 36 は、DSP 34 が所定の処理 (たとえば、先読みしておいたゲームプログラムやサウンドデータの記憶) を行う際に用いられる。DSP 34 は、RAM 36 に記憶されたサウンドデータ (音波形データ) を読み出し、CPU 26 からの音制御データと当該音波形データ等に基づいて音出力のためのデータを生成し、メモリコントローラ 30 およびオーディオ I / F 44 を介してモニタ 16 に備えられるスピーカ 24 から音を出力させる。

20

【0029】

メモリコントローラ 30 は、データ転送を統括的に制御し、メモリコントローラ 30 には上述した各種 I / F が接続される。コントローラ I / F 38 は、たとえば4つのコントローラ I / F で構成され、それらが有するコネクタを介して外部機器とゲーム装置 12 とを通信可能に接続する。たとえば、受信ユニット 18 は、上記コネクタと嵌合し、コントローラ I / F 38 を介してゲーム装置 12 と接続される。上述したように受信ユニット 18 は、コントローラ 14 からの操作データを受信し、コントローラ I / F 38 を介して当該操作データを CPU 26 へ出力する。なお、他の実施例では、ゲーム装置 12 には、受信ユニット 18 に代えて、コントローラ 14 から送信されてくる操作データを受信する受信モジュールがその内部に設けられてよい。この場合、受信モジュールが受信した送信データは、所定のバスを介して CPU 26 に出力される。

30

【0030】

ビデオ I / F 40 にはモニタ 16 が接続され、モニタ 16 にはビデオ I / F 40 からの画像信号によってゲーム画像が表示される。外部メモリ I / F 42 には外部メモリカード 22 が接続され、CPU 26 は、メモリコントローラ 30 を介して外部メモリカード 22 に設けられたフラッシュメモリ等にアクセスする。

40

【0031】

オーディオ I / F 44 にはモニタ 16 に内蔵されるスピーカ 24 が接続される。オーディオ I / F 44 は、DSP 34 によって RAM 36 から読み出されたり生成されたりしたサウンドデータやディスクドライブ 48 から直接出力されるサウンドデータに対応するオーディオ信号をスピーカ 24 に与える。スピーカ 24 からは当該サウンドが出力される。

【0032】

ディスク I / F 46 には、ディスクドライブ 48 が接続される。ディスクドライブ 48 は、所定の読み出し位置に配置された光ディスク 20 に記憶されているデータを読み出す。読み出されたデータはディスク I / F 46 およびメモリコントローラ 30 等を介してメ

50

インメモリ 28 に書き込まれ、あるいは、オーディオ I / F 44 に出力される。

【0033】

図3にはコントローラ14の外観の一例が示される。図3(A)は、コントローラ14の上面後方から見た斜視図であり、図3(B)は、コントローラ14を下面後方から見た斜視図である。

【0034】

コントローラ14は、たとえばプラスチック成型によって形成されたハウジング50を有している。ハウジング50は、その前後方向(図3に示すZ軸方向)を長手方向とした略直方体形状を有しており、全体として大人や子供の片手で把持可能な大きさである。一例として、ハウジング50は人間の掌とほぼ同じ長さまたは幅を持つ大きさをしている。

プレイヤーは、コントローラ14を用いて、それに設けられたボタンを押下するゲーム操作を行うことと、コントローラ14自体の位置や向きを変えることとによって、ゲーム操作を行うことができる。たとえば、或るゲームでは、プレイヤーは、長手方向を軸としてコントローラ14を回転させることによって、操作対象に移動動作を行わせることができる。

10

【0035】

ハウジング50には、複数の操作ボタンが設けられる。ハウジング50の上面には、十字キー52a、Xボタン52b、Yボタン52c、Aボタン52d、セレクトスイッチ52e、メニュースイッチ52f、およびスタートスイッチ52gが設けられる。一方、ハウジング50の下面には凹部が形成されており、当該凹部の後面側傾斜面にはBボタン52iが設けられる。これらの各ボタン(スイッチ)52は、ゲーム装置12が実行するゲームプログラムに応じてそれぞれの機能が割り当てられる。また、ハウジング50の上面には、遠隔からゲーム装置12本体の電源をオン/オフするための電源スイッチ52hが設けられる。

20

【0036】

また、ハウジング50の後面にはコネクタ54が設けられている。コネクタ54は、たとえば32ピンのエッジコネクタであり、コントローラ14に他の機器を接続するために利用される。また、ハウジング50上面の後面側には複数のLED56が設けられる。ここで、コントローラ14には、他のコントローラ14と区別するためにコントローラ種別(番号)が付与される。コントローラ14からゲーム装置12へ操作データを送信する際、当該コントローラ14に現在設定されているコントローラ種別に対応する1つのLED56が点灯する。

30

【0037】

図4にはコントローラ14の電気的構成が示される。コントローラ14は、操作部52(各操作ボタン52a-52h)の他に、その内部に通信部58および加速度センサ60を備えている。

【0038】

加速度センサ60は、当該加速度センサの検出部に加わっている加速度のうち、センシング軸ごとの直線成分の加速度や重力加速度を検出する。加速度センサ60は、少なくとも2つの互いに直交する軸方向の加速度を検出する。たとえば、2軸または3軸の加速度センサの場合には、加速度センサの検出部に加わっている加速度を、各軸に沿った直線成分の加速度としてそれぞれ検出する。具体的には、この実施例では、3軸加速度センサが適用され、コントローラ14の上下方向(図3に示すY軸方向)、左右方向(図3に示すX軸方向)および前後方向(図3に示すZ軸方向)の3軸でそれぞれ加速度を検知する。また、この加速度センサ60の各軸ごとに検出される加速度に所定の演算処理を施すことによって、コントローラ14の傾きや回転を算出することが可能である。たとえば、加速度センサ60が静止した状態においては、常に重力加速度が加わっており、この重力加速度に対する各軸の傾きに応じた加速度が各軸ごとに検出される。具体的には、加速度センサ60が水平状態で静止しているとき、加速度センサのY軸に1Gの重力加速度が加わり、他の軸の重力加速度はほぼ0となる。次に、加速度センサ60の姿勢が水平状態より傾くと、加速度センサ60の各軸方向と重力方向との角度に応じて、重力加速度が加速度セ

40

50

ンサ60の各軸に分散され、このとき、加速度センサ60の各軸の加速度値が検出される。このような各軸ごとの加速度値に演算を加えることによって、重力方向に対する加速度センサ60の姿勢を算出することができる。

【0039】

なお、加速度センサ60としては、必要な操作信号の種類によっては、上下方向、左右方向および前後方向のうちいずれか2つの組合せの2軸でそれぞれ加速度を検出する2軸加速度センサが用いられてもかまわない。この実施例では、ユーザによるコントローラ14に対するストローク奏法のような操作状態を加速度センサ60で検出する。後述する図5に示すように、ユーザはハウジング50の長手方向をおおよそ水平方向に向けかつハウジング50の上面を自分の方に向けた状態でコントローラ14を把持して操作するので、当該ハウジング50の前後方向(Z軸方向)および左右方向(X軸方向)の2軸を検出する加速度センサ60が適用される。

10

【0040】

加速度センサ60が検出した加速度を示すデータは、通信部58へ出力される。なお、加速度センサ60は、典型的には静電容量式の加速度センサが用いられ得る。加速度センサ60は、たとえば最大200フレーム/秒程度のサンプリング周期を有する。

【0041】

通信部58は、マイクロコンピュータ(Micro Computer:マイコン)62、メモリ64、無線モジュール66およびアンテナ68を含む。マイコン62は、処理の際にメモリ64を記憶領域として用いながら、取得したデータを無線送信する無線モジュール66を制御する。

20

【0042】

操作部52および加速度センサ60からマイコン62へ出力されたデータは、一時的にメモリ64に格納される。ここで、通信部58から受信ユニット18への無線送信は所定の周期毎に行われるが、ゲームの処理は1/60秒を単位として行われることが一般的であるので、それよりも短い周期で送信を行うことが必要となる。マイコン62は、受信ユニット18への送信タイミングが到来すると、メモリ64に格納されているデータを操作データとして無線モジュール66へ出力する。無線モジュール66は、たとえばBluetooth(ブルートゥース)(登録商標)の技術を用いて、所定周波数の搬送波を操作データで変調し、その微弱電波信号をアンテナ68から放射する。つまり、操作データは、無線モジュール66で微弱電波信号に変調されてコントローラ14から送信される。微弱電波信号はゲーム装置12側の受信ユニット18で受信される。受信された微弱電波信号について復調や復号を行うことによって、ゲーム装置12は操作データを取得することができる。ゲーム装置12のCPU26は、コントローラ14から取得した操作データに基づいてゲーム処理を行う。

30

【0043】

なお、図3に示したコントローラ14の形状や、各操作スイッチ52の形状、数および設置位置等は単なる一例に過ぎず、他の形状、数および設置位置に適宜変更され得る。

【0044】

上記コントローラ14を用いることによって、プレイヤは、各操作スイッチを押下する従来の一般的なゲーム操作に加えて、コントローラ14自身の位置を動かしたり、コントローラ14を回転させたりするというゲーム操作を行うことができる。

40

【0045】

この音出力制御装置10では、ギターなどのストローク奏法のようにコントローラ14を振る操作が行われることによって、音出力される。つまり、このゲームシステム10では、ユーザは、コントローラ14を把持して手首にスナップを効かせたようなストローク動作を行うことによって、ギターなどの模擬的なストローク演奏を行うことができる。

【0046】

図5に、コントローラ14の把持の仕方と操作方法とが示される。なお、図5では右手でコントローラ14を扱う場合が示される。図5(A)はユーザ側から見た図を示し、図

50

5 (B) はユーザの左側から見た図を示す。ユーザは、ハウジング 5 0 の上面がユーザ側に向けられ先端側がユーザの左手側に向けられているコントローラ 1 4 をほぼ水平にして、親指をハウジング 5 0 の上面で長手方向に添わせて、コントローラ 1 4 を把持する。言い換えれば、ユーザは、Z 軸正方向がユーザの左側を向き、かつ、X 軸正方向が上側を向くようにしてコントローラ 1 4 を把持する。そして、図 5 に示されるように、ユーザは、ストローク奏法のように手首にスナップを効かせるようにしてコントローラ 1 4 を上下方向に振る。

【 0 0 4 7 】

コントローラ 1 4 に対して上述のようなストローク操作が行われるときの加速度が加速度センサ 6 0 によって検出される。発明者等は、ストローク奏法のようにコントローラ 1 4 を操作するときに検出される Z 軸方向の加速度と X 軸方向の加速度の和の値が、操作されているコントローラ 1 4 の位置に応じて変化すること、そして、特定の値を特定の弦に割り当てることによってストローク演奏に似た操作感で模擬的な演奏を行えることを見出した。

10

【 0 0 4 8 】

具体的には、図 6 に示すように、コントローラ 1 4 がストローク奏法のように振られるとき、加速度センサ 6 0 の X 軸方向の加速度と Z 軸方向の加速度の和の値は、所定の範囲内で変化する。X 軸方向の加速度と Z 軸方向の加速度の和の値は、ストローク操作の状態を示すことから、ストローク値と呼ぶものとする。たとえば、重力加速度を 1 . 0 としたとき、ストローク値は、おおよそ - 1 . 3 (手が最上にあるとき) から 1 . 3 (手が最下にあるとき) の範囲の値を示すことが実験によって確認されている。そして、実際のギターのストローク演奏時と同様な手の振りで音が出力されるように、特定の位置 (特定のストローク値) と各弦の位置とが対応付けられる。この実施例では、0 . 2 5 から 0 . 7 5 までの範囲の特定の値のそれぞれが、ギターの場合の第 6 弦から第 1 弦の各弦の位置に対応付けられる。

20

【 0 0 4 9 】

ゲーム装置 1 2 では、ストローク値の変化が各弦の閾値と所定の関係になったか否かが判別され、所定の関係になったとき、つまり、ストローク値が各弦に対応付けられた値を通過したとき、当該弦が弾かれたものと見なして、当該弦に対応付けられた音が出力されることとなる。

30

【 0 0 5 0 】

このように、ストローク値の変化が予め定められた複数の閾値と所定の関係になったことに応じて、音出力が制御されるようにしている。したがって、この実施例のように、たとえばギターの 6 つの弦のそれぞれに対応する各閾値を予め適切に設定しておくことによって、複数の弦が実際のギターのように時間差で鳴るような模擬的な演奏の状況を、簡単な処理で実現することができる。

【 0 0 5 1 】

また、ストローク値の変化と比較される閾値ごとに異なる音色を対応付けることができるので、1 回のストローク操作によって異なる音を順次出力することができる。この実施例のように、たとえばギターの 6 つの弦にそれぞれ異なる所定の音色を設定しておけば、本物のギターと同じ和音を鳴らすような模擬演奏を、簡単な処理で実現することができる。

40

【 0 0 5 2 】

図 7 にはメモリマップの一例が示される。メインメモリ 2 8 はプログラム記憶領域 8 0 およびデータ記憶領域 8 2 を含む。なお、図 7 にはメモリマップの一部が示されており、音出力制御処理に必要な他のプログラムおよびデータも、光ディスク 2 0 等から読み出されたり、CPU 2 6 によって生成されたり取得されたりして、メインメモリ 2 8 に記憶されている。

【 0 0 5 3 】

プログラム記憶領域 8 0 の記憶領域 8 4 には操作データ取得プログラムが記憶されてい

50

る。このプログラムによって、コントローラ 14 からの操作データが、受信ユニット 18 およびコントローラ I/F 38 を介してメインメモリ 28 に取得される。上述のように、コントローラ 14 は、ゲーム装置 12 における 1 フレーム（たとえば 1 / 60 秒）よりも短い周期で操作データを送信してくる。また、コントローラ 14 の加速度センサ 60 のサンプリング周期は、ゲーム装置 12 における 1 フレームよりも短い周期（たとえば 1 / 200 秒）に設定されており、コントローラ 14 の 1 回の送信データには複数の検出タイミングにおける加速度値が含まれる。したがって、この実施例では、ゲーム装置 12 では 1 フレームに複数の操作情報（加速度値等）を含む操作データを取得することができる。CPU 26 は、複数の操作情報を必要に応じて利用して音出力制御処理を実行することができる。

10

【0054】

記憶領域 86 には、コード音設定プログラムが記憶されている。このプログラムによってコード音が設定される。この実施例で設定され得るギターコードは C、G7、Am、F、Dm 等を含む。この実施例では、操作スイッチ 52 のうちたとえば十字キー 52a の操作によってコードが選択される。また、設定されたコードに応じて、各弦の音色が設定される。つまり、各弦の音がコードの構成音にされる。したがって、本物のギターと同様に和音を鳴らすことができる。

【0055】

記憶領域 88 には発音制御プログラムが記憶されている。このプログラムによって音の出力の有無が制御される。具体的には、この実施例では、操作スイッチ 52 のうち A ボタン 52d の押下げの有無によって、音の出力のオンオフが切替えられる。つまり、A ボタン 52d が押下げられているときは音出力されるが、A ボタン 52d が押されていないときには音出力されない。したがって、ユーザは、操作中に A ボタン 52d を押したり離したりすることによって、特定の弦の音を弾いたり弾かなかったりすることが可能である。なお、他の実施例では、逆に、A ボタン 52d が押されたときに消音されるようにしてもよい。

20

【0056】

記憶領域 90 には音色選択プログラムが記憶されている。このプログラムによって、加速度の変化に応じて出力すべき音色が選択される。つまり、ストローク値の変化が各弦に対応付けられた閾値と所定の関係になったか否かが判別され、所定の関係が成立したと判別された弦に設定されている音が選択される。具体的には、いずれかの弦の閾値が、現フレームのストローク値と前フレームのストローク値の間に存在するか否かが判別される。これは、前フレームから現フレームの間にどの弦が弾かれたかを調査している。詳しくは、通常のダウンストロークの場合、各弦の閾値が前フレームのストローク値以上かつ現フレームのストローク値未満であるか否かが判定される。また、アップストロークの場合、各弦の閾値が現フレームのストローク値以上かつ前フレームのストローク値未満であるか否かが判定される。

30

【0057】

記憶領域 92 にはサウンド出力プログラムが記憶されている。このプログラムによってサウンド出力のための制御データが生成され、当該制御データに基づいて音出力される。

40

【0058】

データ記憶領域 82 の記憶領域 94 は操作データバッファであり、コントローラ 14 から送信される操作データが記憶される。上述のように、ゲーム装置 12 における 1 フレームの間に、コントローラ 14 から複数の操作情報を含む操作データを少なくとも 1 回受信するので、受信された操作データが順次この記憶領域 94 に蓄積される。操作データは、加速度センサ 60 で検出された X、Y、Z 軸の加速度値を示す加速度データ、および操作部 52 の各ボタンの操作の有無を示すボタン操作データを含む。

【0059】

記憶領域 96 には加速度履歴が記憶される。所定数のフレーム分の加速度値が記憶され

50

る。この実施例では、XおよびZ軸の加速度値が使用されるので、XおよびZ軸の加速度が操作データバッファから取得されて、この記憶領域96に記憶される。なお、上述のように、1フレームで複数個の加速度値が取得されるので、複数個の値の平均値をとるようにしてよい。あるいは、最大値もしくは最小値を採用してもよい。

【0060】

記憶領域98には、操作データバッファ94から取得されたボタン操作データに基づいて、現フレームにおいて操作されているボタンを示すボタン操作情報が記憶される。

【0061】

記憶領域100にはストローク値履歴が記憶される。ストローク値は上述のようにX軸方向の加速度値とZ軸方向の加速度値の和である。少なくとも現フレームと前フレームにおけるストローク値が記憶される。

10

【0062】

記憶領域102には、光ディスク20から読み出された弦閾値テーブルが記憶されている。弦閾値テーブルは、ストローク値の変化との比較のために、各弦の位置を示す閾値が記憶される。図8に示すように、各弦の識別情報を示す弦番号に対応付けて閾値が登録されている。この実施例では、第6弦の閾値は0.25であり、第1弦までの各閾値が0.1刻みの値で設定されている。ストローク値の変化がこれら閾値のいずれかを越えたか否かが判定される。なお、この図8に示された閾値テーブルの値は、図5で示したように右手で操作される場合のデータであり、弦閾値テーブル記憶領域102には左手用のテーブルも記憶される。操作する手は図示しない設定画面などでユーザによって選択され、当該設定に応じて使用されるテーブルが決められる。

20

【0063】

記憶領域104には設定コードが記憶される。たとえば、コードのデフォルトはCであり、十字キー52aの操作に応じてコードが変更される。一例として、十字キー52aの上方向を指示する部分にG7、下方向を指示する部分にF、右方向を指示する部分にAm、左方向を指示する部分にDmがそれぞれ対応付けられており、各方向指示部分が押されると、当該部分に対応付けられたコードが選択される。また、選択されたコードと反対方向を指示する部分等が押されることによって当該選択が解除され、デフォルトのCが選択される。

【0064】

30

記憶領域106には音色データが記憶される。音色データは、各弦の音色を示す情報である。コードの設定に応じて、当該コードを構成する各弦の音色のサウンドデータを指定するデータが各弦について選択されて記憶される。

【0065】

図9にはゲーム装置12の音出力制御処理における動作の一例が示される。まず、ステップS1でCPU26は初期設定を実行する。これによってメインメモリ28がクリアされ、必要なプログラムおよびデータが光ディスク20からメインメモリ28に読み出される。また、各種変数やフラグに初期値が設定される。

【0066】

続くステップS3からS9の処理は1フレームごとに実行される。CPU26は、ステップS3で加速度情報とボタン操作情報を取得する。具体的には、CPU26は、操作データバッファ94から操作データを読み出して、X軸方向およびZ軸方向の加速度値を取得して加速度履歴記憶領域96に記憶するとともに、各ボタン52の操作情報をボタン操作情報記憶領域98に記憶する。

40

【0067】

続いて、ステップS5で、CPU26は演奏処理を実行する。この演奏処理によって、ユーザによるコントローラ14のストローク操作に応じて音が出力される。演奏処理の動作の一例は図10に詳細に示される。

【0068】

図10のステップS21で、CPU26は、十字キー操作に応じてコードを設定する。

50

具体的には、ボタン操作情報記憶領域 98 から十字キー 52 a の操作情報が取得され、十字キー 52 a の操作された方向に基づいてコードが選択されて、選択されたコードを示す情報が記憶領域 104 に記憶される。なお、設定コード記憶領域 104 には初期設定で C コードを示す情報が記憶されている。十字キー 52 a が操作されていない場合には設定コードは変更されない。

【 0069 】

次に、ステップ S23 で、CPU 26 は、設定されたコードに基づいて各弦の音色を設定し、各弦の音色を示す情報を音色データ記憶領域 106 に記憶する。

【 0070 】

ステップ S25 では、CPU 26 は、現フレームにおける X 軸方向および Z 軸方向の加速度値を記憶領域 96 から読み出して、両値を加算してストローク値を算出する。算出された現フレームのストローク値はストローク値履歴記憶領域 100 に記憶される。

10

【 0071 】

続いて、ステップ S27 で、CPU 26 は、A ボタンが押されているか否かを、ボタン操作情報記憶領域 98 のデータに基づいて判断する。つまり、ユーザによって音の出力が指示されているか否かが判定される。ステップ S27 で“NO”であれば、つまり、ユーザが音の出力をしないことを意図している場合には、この演奏処理を終了し、処理は図 9 のステップ S7 へ戻る。

【 0072 】

一方、ステップ S27 で“YES”であれば、各弦が弾かれたか否かの判定をストローク値の変化に基づいて行う。すなわち、CPU 26 は、ステップ S29 で、処理対象の弦を指定するための変数 N に初期値（この実施例では第 6 弦を示す「6」）を設定する。

20

【 0073 】

続いて、ステップ S33 で、CPU 26 は、N 値に対応する弦の閾値を閾値テーブル記憶領域 102 から読み出す。そして、ステップ S33 で、CPU 26 は、第 N 弦の閾値が前フレームのストローク値以上でありかつ現フレームのストローク値未満であるか否かを判断する。ここでは、ストローク値が第 N 弦の閾値を下から上に通過したか否か、つまり、ダウンストロークによって第 N 弦が上から下に弾かれたか否かを判定している。ステップ S33 で“YES”であれば、当該第 N 弦の音の出力を行うために処理はステップ S39 へ進む。

30

【 0074 】

一方、ステップ S33 で“NO”であれば、CPU 26 は、ステップ S35 でアップストロークを行うことが可能であるか否かを判断する。なお、データ記憶領域 82 に記憶されるアップストロークの可否を示すフラグに基づいて当該判定を行う。ダウンストロークだけでなくアップストロークの場合にも音の出力を許可する場合には、たとえば初期設定で上記フラグに許可を示す情報が設定される。アップストロークの可否の設定は、ユーザによる操作ボタン 52 の操作によって、図示しない設定画面等で変更可能にされる。

【 0075 】

ステップ S35 で“YES”の場合には、CPU 26 は、ステップ S37 で、第 N 弦の閾値が現フレームのストローク値以上でありかつ前フレームのストローク値未満であるか否かを判断する。ここでは、ストローク値が第 N 弦の閾値を上から下に通過したか否か、つまり、アップストロークによって第 N 弦が下から上に弾かれたか否かを判定している。ステップ S37 で“YES”であれば、当該第 N 弦の音の出力を行うために処理はステップ S39 へ進む。

40

【 0076 】

一方、ステップ S37 で“NO”の場合、つまり、処理対象として設定した第 N 弦が弾かれていない場合には、次の弦のための制御を行うべく処理はステップ S45 に進む。また、ステップ S35 で“NO”の場合にも、同様に処理はステップ S45 に進む。

【 0077 】

ステップ S39 では、CPU 26 は X 軸および Z 軸方向の加速度値の和から音量を設定

50

する。具体的には、X軸加速度とZ軸加速度の和であるストローク値の現在と前回の値を使用し、現フレームのストローク値と前フレームのストローク値との差分の絶対値に基づいて、音量を算出する。ストローク値がストローク操作の「位置」を示し、「位置」の差分がストローク操作の「速度」を示すため、この「速度」を音量のスケールに変換する。早く動いたときほど音量は大きくされる。

【0078】

ステップS41では、CPU26は第N弦の音を出力するための制御データを生成する。つまり、CPU26は、第N弦に設定されている音色データを記憶領域106から読み出し、当該音色を設定音量で出力することを指示する情報を含む制御データを生成する。

【0079】

そして、ステップS43で、CPU26は、第N弦の音を出力する。具体的には、CPU26は、第N弦の音を出力するための制御データをメモリコントローラ30を介してDSP34に与える。この制御データに応じて、DSP34は、ARAM36に記憶されている音波形データを用いて、当該音を出力するためのデータを生成し、当該データをメモリコントローラ30を介してオーディオI/F44に与える。オーディオI/F44は、当該音を出力するためのデータに基づいて当該音を出力するためのオーディオ信号をスピーカ24に与える。これによって当該第N弦の音がスピーカ24から出力される。

【0080】

ステップS45では、CPU26は、変数Nの値から1を減算し、処理対象を次の弦に設定する。そして、ステップS47で、CPU26は変数Nの値が0になったか否かを判定する。つまり、すべての弦の閾値に関してストローク値の変化との比較を完了したか否かを判定している。ステップS47で“NO”であれば、次の弦のための処理を行うために、処理はステップS31へ戻る。このようにして、ストローク値の変化に応じて弾かれた弦が特定され、当該弦の音出力される。一方、ステップS47で“YES”であれば、この演奏処理を終了して処理は図9のステップS7へ戻る。

【0081】

図9のステップS7では、CPU26は表示処理を実行する。これによって、CPU26は、GPU32を用いて画面を表示するためのデータを生成し、当該生成した画面をモニタ16に表示する。画面には、たとえば、コントローラ14の持ち方やストローク操作の方法などの説明が表示されてよい。

【0082】

ステップS9では、CPU26は模擬演奏を終了するか否かを判定する。たとえば、模擬演奏の終了を指示するためのいずれかの操作スイッチ52が押されたか否かをボタン操作情報に基づいて判断する。ステップS9で“NO”であれば、処理はステップS3に戻り、ユーザのストローク操作に応じた音出力制御を続け、“YES”であればこの音出力制御処理を終了する。

【0083】

この実施例によれば、ユーザによって操作されるコントローラ14に少なくとも2軸の加速度を検出する加速度センサ60を設けて、当該加速度センサ60の2軸の加速度の和の変化に応じてすなわちストローク操作の状態に応じて、音が制御されるようにしたので、ギターのストローク演奏のような手首のスナップを効かせた操作による演奏を模擬的に行うことができる。単純なポイントングでは再現できないような演奏に対して、コントローラ14の振りなど操作の状態に応じて音の出力を制御できるので、今までに無い娯楽的な音出力が可能である。

【0084】

なお、他の実施例では、図9のステップS7の表示処理によって、たとえば図11に示すように、ユーザのストローク操作の状態（弾いている位置）を示す画面を表示するようにしてもよい。図11に示すように、画面の右側部分において、ギターの6つの弦とともに、一例としてピックを模したアイコン110が表示される。この画面において、第6弦から第1弦のそれぞれは各閾値に対応する位置に表示される。また、ピックアイコン11

10

20

30

40

50

0はストローク値に対応する位置に表示され、ストローク操作に応じて上下方向に移動する。このような画面表示によって、ユーザのストローク操作の状態と弦との位置関係を、ユーザに明示することができる。したがって、ユーザは、自分のストローク操作によってどの弦を弾いているかを逐一確認でき、どのようなストローク操作が適切かを把握できるので、より容易に模擬的なストローク演奏を行うことができるようになる。

【0085】

ただし、コントローラ14をストローク奏法のように動かせば、ストローク値の変化に応じて各弦の音が出力されるので、初めてプレイするユーザであっても、何度かコントローラ14をストローク操作すれば、上手く音出力される操作の感覚を容易に掴むことができる。したがって、図11のような画面表示は全く行われなくてもよいし、あるいは、

10

【0086】

また、図11の画面では、ピックアイコン110の位置がストローク操作の位置を示しているので、Aボタン52dによる発音の有無の切り替えを容易に行うことができる。つまり、ユーザは、ピックアイコン110が所望の弦を通過しようとしているときにAボタン52dを押すことによって、所望の弦のみを簡単に弾くことができる。なお、Aボタン52dの操作の有無に応じてピックアイコン110の色などの表示態様を切り替えることによって、Aボタン52dの操作状態を視覚によって認識させて、さらに容易に発音制御が行えるようにしてもよい。

20

【0087】

さらに、図11に示すように、コード選択のためにコード表を表示するようにしてもよい。上述の実施例では、一例として、デフォルトのコードがCで、他にG7、F、Am、Dmを十字キー52aの各方向指示部分の操作によって選択可能にしていたので、図11のコード表では、中央部分にCが表示され、上下左右にそれぞれG7、F、DmおよびAmが表示されている。また、現在設定(選択)されているコードを示す部分(図11ではCの部分)にカーソルが表示されている。カーソルはユーザの十字キー52aの操作に応じて移動される。たとえば、図11の状態では十字キー52aの上方向が操作されると、カーソルはG7の部分に移動し、その後下方向が操作されるとカーソルは再びCの部分に戻る。このように、設定されているコードと選択可能なコードを表示することによって、ユーザにコード選択を容易に行わせることができる。

30

【0088】

なお、上述の各実施例では、ギターの模擬演奏システムとして構成された音出力制御装置10を説明した。しかし、この音出力制御装置10では、指先などで弦を弾ずる弦楽器であれば、たとえばウクレレやシタールといったギター以外の弦楽器についても模擬的なストローク演奏を実現可能であるのは勿論である。

【図面の簡単な説明】

【0089】

【図1】この発明の一実施例の音出力制御装置の一例を示す外観図である。

【図2】図1のゲーム装置の電気的構成の一例を示すブロック図である。

40

【図3】図1のコントローラの一例を示す図解図であり、(A)は後方上側から見た斜視図であり、(B)は後方下側から見た斜視図である。

【図4】図1のコントローラの電気的構成の一例を示すブロック図である。

【図5】ストローク演奏を行う際のコントローラの操作方法を示す図解図であり、(A)はユーザ側から見た動きを示し、(B)はユーザの左側から見た動きを示す。

【図6】コントローラの操作状態に応じて変化するストローク値と各弦の閾値との関係の一例を示す図解図である。

【図7】メモリマップの一例を示す図解図である。

【図8】弦閾値テーブルデータの一例を示す図解図である。

【図9】ゲーム装置の動作の一例を示すフロー図である。

50

【図10】図9の演奏処理の動作の一例を示すフロー図である。

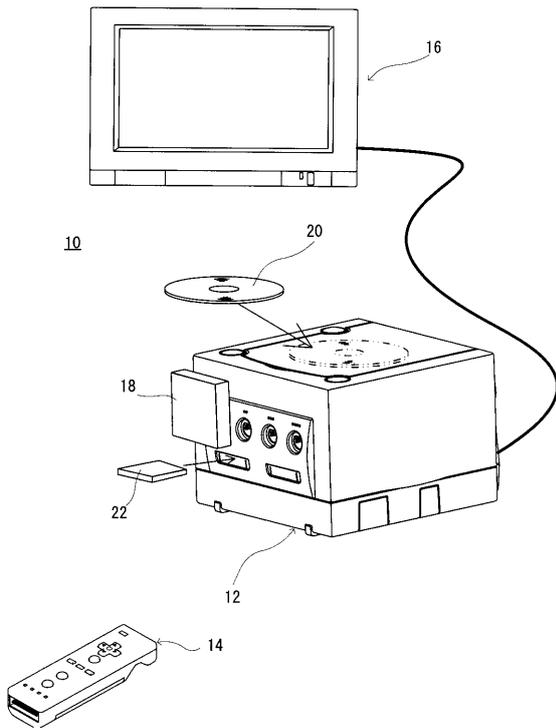
【図11】表示画面の一例を示す図解図である。

【符号の説明】

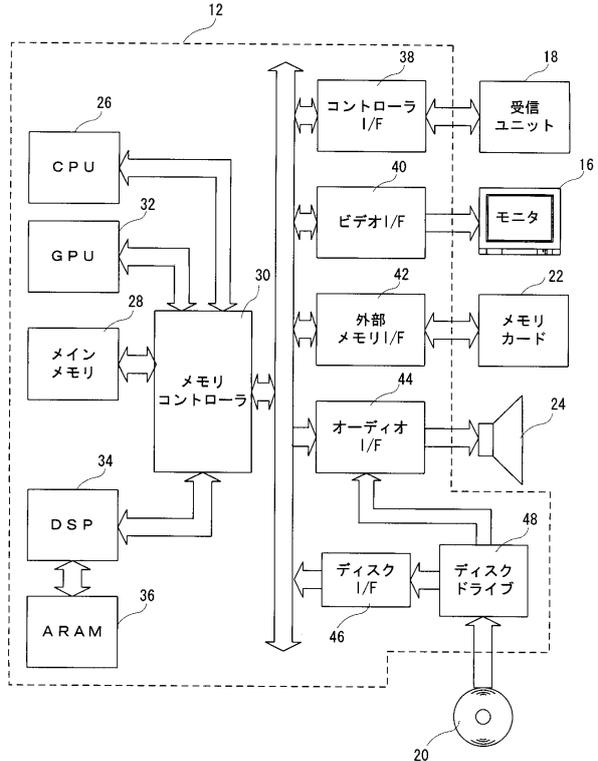
【0090】

- 10 ...ゲームシステム
- 12 ...ゲーム装置
- 14 ...コントローラ
- 16 ...モニタ
- 18 ...受信ユニット
- 20 ...光ディスク
- 24 ...スピーカ
- 26 ...CPU
- 28 ...メインメモリ
- 34 ...DSP
- 52 ...操作部
- 58 ...通信部
- 60 ...加速度センサ

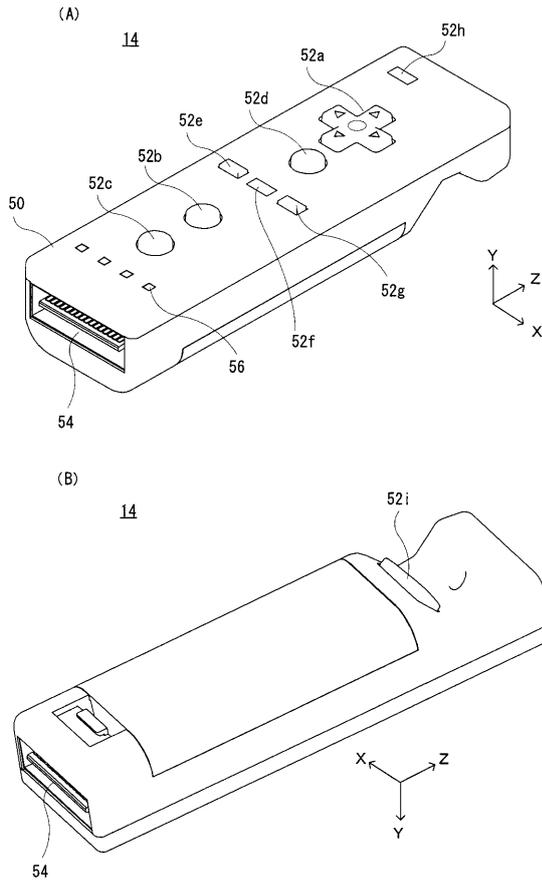
【図1】



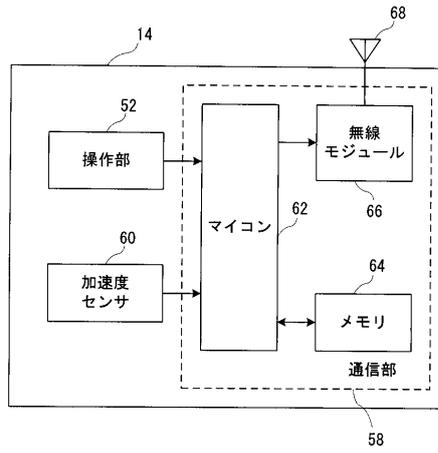
【図2】



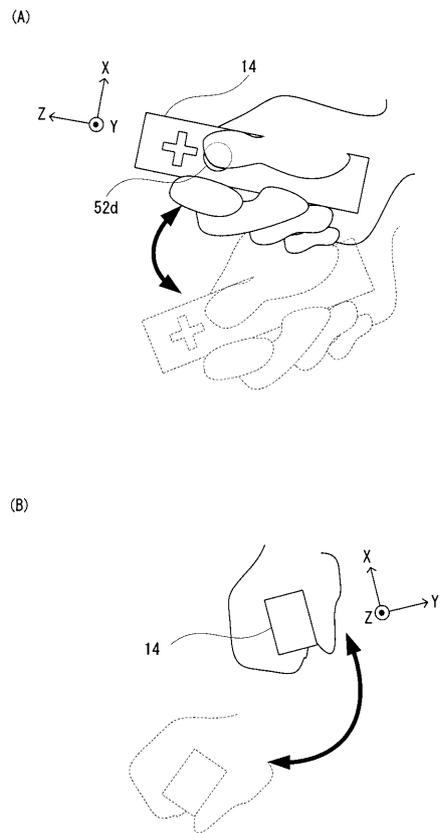
【図3】



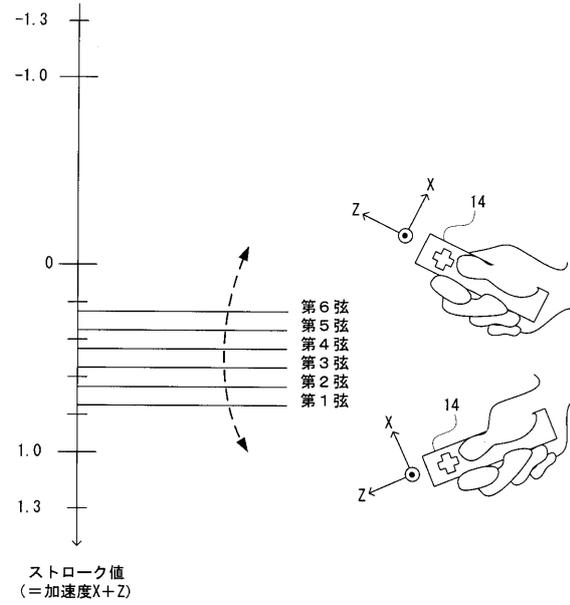
【図4】



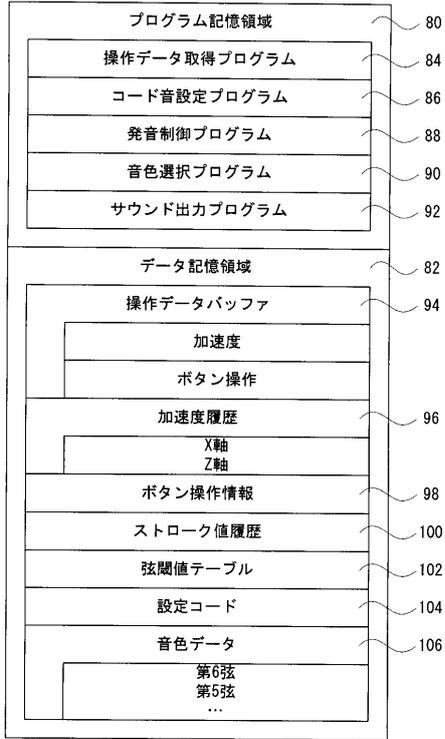
【図5】



【図6】



【図7】

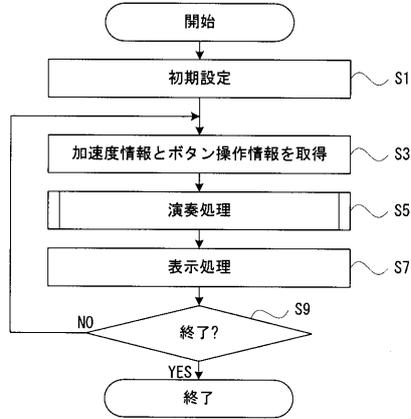


【図8】

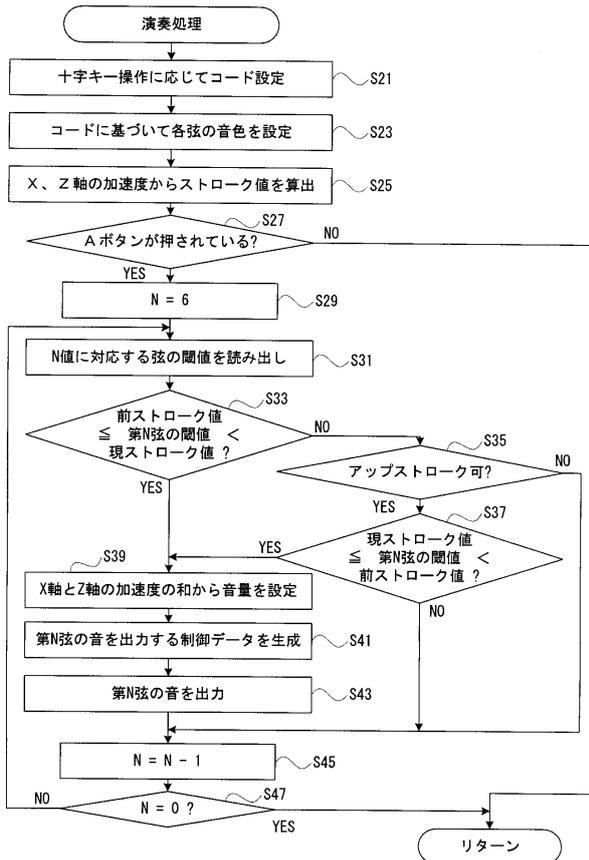
弦閾値テーブル

弦番号	閾値
第6弦	0.25
第5弦	0.35
第4弦	0.45
第3弦	0.55
第2弦	0.65
第1弦	0.75

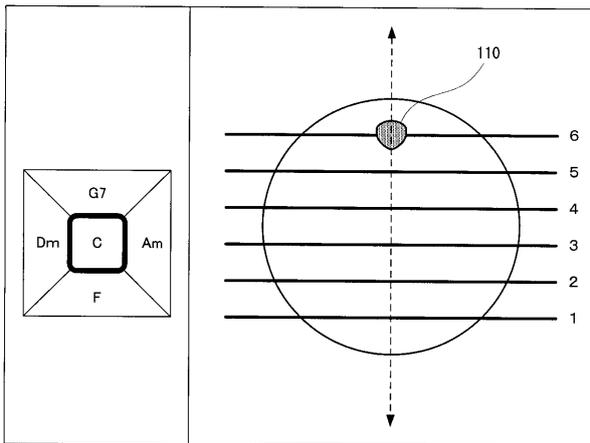
【図9】



【図10】



【図11】



フロントページの続き

- (56)参考文献 特開平04 - 184490 (JP, A)
特開2004 - 053930 (JP, A)
特開2001 - 013967 (JP, A)
特開2000 - 276141 (JP, A)
特開2003 - 076368 (JP, A)
特開平07 - 121294 (JP, A)
特開2002 - 023742 (JP, A)
特開平10 - 055174 (JP, A)
特開平05 - 080756 (JP, A)
特開2000 - 267659 (JP, A)
特開平10 - 282957 (JP, A)
特開2002 - 336547 (JP, A)

(58)調査した分野(Int.Cl., DB名)

G10H 1/00 - 7/12
A63F 9/24
A63F 13/00 - 13/12
G09B 11/00 - 15/08
G10K 15/00 - 15/12