

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第4439572号
(P4439572)

(45) 発行日 平成22年3月24日 (2010. 3. 24)

(24) 登録日 平成22年1月15日 (2010. 1. 15)

(51) Int. Cl.			F I		
GO 1 D	1/02	(2006. 01)	GO 1 D	1/02	C
GO 1 D	3/028	(2006. 01)	GO 1 D	3/04	F
A 6 3 F	13/00	(2006. 01)	A 6 3 F	13/00	F
GO 1 P	21/02	(2006. 01)	GO 1 P	21/02	
GO 1 C	19/00	(2006. 01)	GO 1 C	19/00	Z

請求項の数 14 (全 32 頁)

(21) 出願番号	特願2008-181804 (P2008-181804)	(73) 特許権者	000233778 任天堂株式会社 京都府京都市南区上鳥羽鉾立町 1 1 番地 1
(22) 出願日	平成20年7月11日 (2008. 7. 11)	(74) 代理人	100090181 弁理士 山田 義人
(65) 公開番号	特開2010-19751 (P2010-19751A)	(74) 代理人	100130269 弁理士 石原 盛規
(43) 公開日	平成22年1月28日 (2010. 1. 28)	(72) 発明者	太田 敬三 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
審査請求日	平成21年7月21日 (2009. 7. 21)	審査官	榮永 雅夫
早期審査対象出願			

最終頁に続く

(54) 【発明の名称】 デジタルデータ補正プログラムおよびデジタルデータ補正装置

(57) 【特許請求の範囲】

【請求項 1】

所定のセンサで計測したアナログ量を A / D 変換手段によって変換したデジタルデータを補正するデジタルデータ補正装置のコンピュータに実行させるデジタルデータ補正プログラムであって、前記コンピュータを、

前記 A / D 変換手段で変換されたデジタルデータを順次記憶するバッファ手段、

前記 A / D 変換手段で変換された最新のデジタルデータに対する安定範囲を算出する安定範囲算出手段、

前記バッファ手段に記憶されたデジタルデータのうち、前記最新のデジタルデータから遡って連続して前記安定範囲内にあるデジタルデータの平均値を算出する平均値算出手段、

前記バッファ手段に記憶されたデジタルデータのうち、前記最新のデジタルデータから遡って連続して前記安定範囲内にあるデジタルデータの連続度合いを算出する連続度合い算出手段、

前記連続度合いに基づいて、前記最新のデジタルデータと前記平均値との加重平均値を算出する加重平均値算出手段、および

前記加重平均値算出手段で算出された加重平均値を用いて前記最新のデジタルデータを補正するデータ補正手段として機能させる、デジタルデータ補正プログラム。

【請求項 2】

前記連続度合い算出手段は、前記バッファ手段に記憶されたデジタルデータのうち、

前記最新のデジタルデータから遡って連続して前記安定範囲内にあるデジタルデータの連続個数を算出する連続個数算出手段を含み、

前記加重平均値算出手段は、前記連続個数に基づいて、前記最新のデジタルデータと前記平均値との加重平均値を算出する、請求項1記載のデジタルデータ補正プログラム。

【請求項3】

前記加重平均値算出手段は、前記連続個数が多いほど、前記平均値の重みを重くする、請求項2記載のデジタルデータ補正プログラム。

【請求項4】

前記加重平均値算出手段は、前記連続個数のべき乗に比例して、前記平均値の重みを重くする、請求項3記載のデジタルデータ補正プログラム。 10

【請求項5】

前記安定範囲は、前記最新のデジタルデータを含む範囲である、請求項1記載のデジタルデータ補正プログラム。

【請求項6】

前記安定範囲は、前記最新のデジタルデータを中心とした上下均等な範囲である、請求項5記載のデジタルデータ補正プログラム。

【請求項7】

前記コンピュータを、
前記連続度合いに応じて、前記データ補正手段で補正後のデジタルデータに収束する修正オフセット値を算出する修正オフセット値算出手段、および
前記データ補正手段で補正後のデジタルデータを前記修正オフセット値で補正するオフセット補正手段としてさらに機能させる、請求項1記載のデジタルデータ補正プログラム。 20

【請求項8】

前記連続度合い算出手段は、前記バッファ手段に記憶されたデジタルデータのうち、前記最新のデジタルデータから遡って連続して前記安定範囲内にあるデジタルデータの連続個数を算出する連続個数算出手段を含み、

前記修正オフセット値算出手段は、前記連続個数に応じて、前記データ補正手段で補正後のデジタルデータに収束する修正オフセット値を算出する、請求項7記載のデジタルデータ補正プログラム。 30

【請求項9】

前記修正オフセット値算出手段は、前記連続個数が多いほど、前記データ補正手段で補正後のデジタルデータに収束する度合いの強い修正オフセット値を算出する、請求項8記載のデジタルデータ補正プログラム。

【請求項10】

前記修正オフセット値算出手段は、前記連続個数のべき乗に比例して、前記データ補正手段で補正後のデジタルデータに収束する度合いの強い修正オフセット値を算出する、請求項9記載のデジタルデータ補正プログラム。

【請求項11】 40

前記データ補正手段は、前記加重平均値を用いて補正した補正後のデジタルデータを出力するデータ出力手段を含む、請求項1記載のデジタルデータ補正プログラム。

【請求項12】

前記センサはジャイロセンサを含み、前記デジタルデータは角速度データである、請求項1ないし11のいずれかに記載のデジタルデータ補正プログラム。

【請求項13】

前記ジャイロセンサはゲーム機のコントローラに設けられ、前記ゲーム機は前記デジタルデータに基づいてゲーム処理を実行する、請求項12記載のデジタルデータ補正プログラム。

【請求項14】 50

所定のセンサで計測したアナログ量を A / D 変換手段によって変換したデジタルデータを補正するデジタルデータ補正装置であって、

前記 A / D 変換手段で変換されたデジタルデータを順次記憶するバッファ手段、

前記 A / D 変換手段で変換された最新のデジタルデータに対する安定範囲を算出する安定範囲算出手段、

前記バッファ手段に記憶されたデジタルデータのうち、前記最新のデジタルデータから遡って連続して前記安定範囲内にあるデジタルデータの平均値を算出する平均値算出手段、

前記バッファ手段に記憶されたデジタルデータのうち、前記最新のデジタルデータから遡って連続して前記安定範囲内にあるデジタルデータの連続度合いを算出する連続度合い算出手段、

10

前記連続度合いに基づいて、前記最新のデジタルデータと前記平均値との加重平均値を算出する加重平均値算出手段、 および

前記加重平均値算出手段で算出された加重平均値を用いて前記最新のデジタルデータを補正するデータ補正手段を備える、デジタルデータ補正装置。

【発明の詳細な説明】

【技術分野】

【0001】

この発明は、デジタルデータ補正プログラムおよびデジタルデータ補正装置に関し、特に、アナログ値を出力するセンサからの出力値をデジタルデータに変換して出力するデバイスから出力したデジタルデータを補正する、デジタルデータ補正プログラムおよびデジタルデータ補正装置に関する。

20

【背景技術】

【0002】

たとえば、アナログ値で角速度を出力するジャイロセンサのようなセンサからの出力値（アナログ値）を、デジタルの整数値データに変換して出力するデバイスを、人が手に持って操作するゲーム機のコントローラなどに使用する場合、アナログ値とデジタルの整数値データとの対応付け、および角速度がゼロであることを示すアナログ値が予め定義されており、デバイスから得られるデジタルの整数値データが、角速度がゼロであることを示すアナログ値に対応付けられたデジタルの整数値データより小さければマイナスの角速度、大きければプラスの角速度と認識する。

30

【0003】

しかし、このようにアナログ値を出力するセンサから出力された出力値をデジタルの整数値データに変換するデバイスには以下のような問題がある。

【0004】

1つは、アナログ値を出力するセンサから出力された出力値（アナログ値）とそのアナログ値を変換したデジタルの整数値データとの対応付けにずれが生じることである。たとえば、センサが静止している状態（角速度がゼロ）のセンサ電圧が 2.9 V であるとすると、センサから得られる電圧をデジタルの整数値データに変換する場合は、2.4 V (= 2.9 - 0.5) ~ 3.4 V (= 2.9 + 0.5) の電圧範囲が、角速度がゼロであることを示す整数値データに変換されるべきとしても、センサの個体差等により必ずしもその電圧範囲である場合に角速度がゼロであることを示す整数値データに変換されず、たとえば、2.0 V (= 2.9 - 0.9) ~ 3.0 V (= 2.9 + 0.1) という電圧範囲が、角速度がゼロであることを示す整数値データに変換される場合がある。

40

【0005】

その結果、2.4 V (= 2.9 - 0.5) ~ 3.4 V (= 2.9 + 0.5) の電圧範囲が、角速度がゼロであることを示す整数値データに変換されるような理想的な環境 / 条件の場合には、2.9 V からどちらの方向に動いても、同じだけの電圧「0.5 V」が変化したときに、整数値データが「+1」または「-1」変化するが、2.0 V (= 2.9 - 0.9) ~ 3.0 V (= 2.9 + 0.1) の電圧範囲が、角速度がゼロであることを示す

50

整数値データに変換されるような環境/条件の場合には、マイナス側には0.9Vの変化が生じないと整数値データがデクリメントされないが、プラス側には0.1Vの変化が生じただけで整数値データがインクリメントされるような現象が起こる。つまり、アナログ値を出力するセンサから出力された出力値(アナログ値)とそのアナログ値を変換したデジタルの整数値データとの対応付けにずれが生じていることになる。

【0006】

2つ目の問題は、温度ドリフトの問題である。たとえばゲームをしている環境の温度が変化することによって、コントローラの使用中に上記のセンサが静止している状態(角速度がゼロ)のセンサ電圧が変化してしまう。たとえば、或る時点までは静止を示す電圧が2.9Vであったのに、いつの間にか3.3Vに変わっている、といったことが起こる。そのため、ゲームのプレイ中に、コントローラすなわちジャイロセンサを静止させているのにも拘らず、動いているかのような整数値データを出力してしまうことが起こる。

10

【0007】

このようなアナログ デジタル変換の1番目の問題に対処できる背景技術が特許文献1に開示されている。

【0008】

特許文献1の従来技術として記載されているデジタル圧力計では、入力値が所定の規定値より大きく変動する場合には、入力値をそのまま出力値として出力し、入力値の変動が所定の規定値以下の場合には、入力値を蓄積して平均化して出力している。

【特許文献1】特開平11-118651号公報[G01L19/08]

20

【発明の開示】

【発明が解決しようとする課題】

【0009】

特許文献1の背景技術では、一定個数の入力値を平均化の対象とするために、少ない一定個数の入力値を平均化する場合には、出力値の精度が低くなる。逆に、平均化する入力値の個数を多くすると、追従性が悪くなってしまふ。たとえば、入力値の変動が規定値を超えている状態から、入力値の変動が規定値より小さい状態へ変化した場合においては、多くの個数の、規定値より大きい入力値が平均化の対象となってしまう、平均化出力値は実際より大きくなることもある。

【0010】

30

それゆえに、この発明の主たる目的は、新規な、デジタルデータ補正プログラムおよびデジタルデータ補正装置を提供することである。

【0011】

この発明の他の目的は、アナログ デジタル変換の問題に有効に対処できる、デジタルデータ補正プログラムおよびデジタルデータ補正装置を提供することである。

【0012】

この発明のさらに他の目的は、出力デジタルデータの精度がよくしかも追従性が悪くない、デジタルデータ補正プログラムおよびデジタルデータ補正装置を提供することである。

【課題を解決するための手段】

40

【0013】

この発明は、上記の課題を解決するために、以下の構成を採用した。なお、括弧内の参照符号および補足説明等は、この発明の理解を助けるために後述する実施の形態との対応関係を示したものであって、この発明を何ら限定するものではない。

【0014】

第1の発明は、所定のセンサで計測したアナログ量をA/D変換手段によって変換したデジタルデータを補正するデジタルデータ補正装置のコンピュータに実行させるデジタルデータ補正プログラムであって、コンピュータを、A/D変換手段で変換されたデジタルデータを順次記憶するバッファ手段、A/D変換手段で変換された最新のデジタルデータに対する安定範囲を算出する安定範囲算出手段、バッファ手段に記憶されたデ

50

デジタルデータのうち、最新のデジタルデータから遡って連続して安定範囲内にあるデジタルデータの平均値を算出する平均値算出手段、バッファ手段に記憶されたデジタルデータのうち、最新のデジタルデータから遡って連続して安定範囲内にあるデジタルデータの連続度合いを算出する連続度合い算出手段、連続度合いに基づいて、最新のデジタルデータと平均値との加重平均値を算出する加重平均値算出手段、および加重平均値算出手段で算出された加重平均値を用いて最新のデジタルデータを補正するデータ補正手段として機能させる、デジタルデータ補正プログラムである。

【 0 0 1 5 】

第1の発明では、センサ(56:実施例で相当する部分を例示する参照符号。以下同じ。)で計測したアナログ量がA/D変換手段(58a, 58b)によってデジタルデータに変換される。このデジタルデータがコントローラ(14)からゲーム機(12)にたとえば近距離無線によって送信され、ゲーム機(12)のCPU(60)がそのデジタルデータを補正する。つまり、ゲーム機(12)のCPU(60)が、メモリ手段(62e, 66)と協働して、A/D変換手段で変換されたデジタルデータを順次記憶するバッファ手段(100)として機能する。安定範囲算出手段(S37)は、最新のデジタルデータに対して各デジタルデータが安定していると定義できる安定範囲の上限および下限(d1, d2)を算出する。平均値算出手段(S49)は、バッファ手段(100)に格納された順次のデジタルデータのうち、今回のデジタルデータから遡って連続して安定範囲内にあるデジタルデータの平均値を算出する。連続度合い算出手段(S43)が、たとえば、バッファ手段に記憶されたデジタルデータのうち、最新のデジタルデータから遡って安定範囲に入っているデジタルデータの連続個数(ct)である、連続度合いを計算する。加重平均値算出手段(S49)は、その連続度合いに基づいて最新のデジタルデータと平均値算出手段で計算した平均値との加重平均値を算出する。そして、データ補正手段(S49)は、加重平均値を用いて最新のデジタルデータを補正する。

【 0 0 1 6 】

第1の発明によれば、連続して安定範囲内に入っているデジタルデータを平均化して最新のデジタルデータを補正するようにしたので、動作時は追従性よく、静止(安定)時は滑らかに高精度な補正を行える。このとき、加重平均を用いるため、補正後のデジタルデータの信頼性が高くなる。

【 0 0 1 7 】

第2の発明は、第1の発明に従属し、連続度合い算出手段は、バッファ手段に記憶されたデジタルデータのうち、最新のデジタルデータから遡って連続して安定範囲内にあるデジタルデータの連続個数を算出する連続個数算出手段を含み、加重平均値算出手段は、連続個数に基づいて、最新のデジタルデータと平均値との加重平均値を算出する、デジタルデータ補正プログラムである。

【 0 0 1 8 】

第2の発明では、連続個数算出手段(S43)が、バッファ手段に記憶されたデジタルデータのうち、最新のデジタルデータから遡って安定範囲に入っているデジタルデータの連続個数(ct)を計算する。加重平均値算出手段(S49)は、連続個数算出手段で算出した連続個数に基づいて最新のデジタルデータと平均値算出手段で計算した平均値との加重平均値を算出し、データ補正手段はその加重平均値を補正後のデジタルデータとする。第2の発明によれば、加重平均を用いるため、補正後のデジタルデータの信頼性が高くなる。

【 0 0 1 9 】

第3の発明は、第2の発明に従属し、加重平均値算出手段は、連続個数が多いほど、平均値の重みを重くする、デジタルデータ補正プログラムである。

【 0 0 2 0 】

第3の発明では、静止(安定)しているときの補正後のデジタルデータの精度がよい。

【 0 0 2 1 】

10

20

30

40

50

第4の発明は、第3の発明に従属し、加重平均値算出手段は、連続個数のべき乗に比例して、平均値の重みを重くする、デジタルデータ補正プログラムである。

【0022】

第4の発明でも、静止（安定）しているときの補正後のデジタルデータの精度がよい。

【0023】

第5の発明は、第1の発明に従属し、安定範囲は、最新のデジタルデータを含む範囲である、デジタルデータ補正プログラムである。

【0024】

第5の発明では、安定範囲が最新のデジタルデータを含んで設定されるので、補正後のデジタルデータの信頼性が損なわれることがない。

【0025】

第6の発明は、第5の発明に従属し、安定範囲は、最新のデジタルデータを中心とした上下均等な範囲である、デジタルデータ補正プログラムである。

【0026】

第6の発明では、安定範囲の上限値および下限値を、最新のデジタルデータより所定値を増減させた値に設定する。第6の発明によれば、安定状態の判断を偏りのなく行え、高精度な補正を行える。

【0027】

第7の発明は、第1の発明に従属し、コンピュータを、連続度合いに応じて、データ補正手段で補正後のデジタルデータに収束する修正オフセット値を算出する修正オフセット値算出手段、およびデータ補正手段で補正後のデジタルデータを修正オフセット値で補正するオフセット補正手段としてさらに機能させる、デジタルデータ補正プログラムである。

第7の発明では、修正オフセット値算出手段（S53）は、連続度合いに応じて、補正後のデジタルデータに収束する修正オフセット値を算出する。そして、オフセット補正手段（S55）が補正後のデジタルデータを修正オフセット値で補正する。したがって、修正オフセット値で補正した補正後のデジタルデータを採用する。第7の発明では、連続度合いに応じた修正オフセット値を用いるので、高精度なオフセット補正が達成できる。

【0028】

第8の発明は、第7の発明に従属し、連続度合い算出手段は、バッファ手段に記憶されたデジタルデータのうち、最新のデジタルデータから遡って連続して安定範囲内にあるデジタルデータの連続個数を算出する連続個数算出手段を含み、修正オフセット値算出手段は、連続個数に応じて、データ補正手段で補正後のデジタルデータに収束する修正オフセット値を算出する、デジタルデータ補正プログラムである。

第8の発明では、連続個数算出手段（S43）が、バッファ手段に記憶されたデジタルデータのうち、最新のデジタルデータから遡って連続して安定範囲内にあるデジタルデータの連続個数を算出し、修正オフセット値算出手段（S53）は、その連続個数に応じて、補正後のデジタルデータに収束する修正オフセット値を算出する。そして、オフセット補正手段（S55）が補正後のデジタルデータを修正オフセット値で補正する。したがって、修正オフセット値で補正した補正後のデジタルデータを採用する。第8の発明では、連続個数に応じた修正オフセット値を用いるので、高精度なオフセット補正が達成できる。

【0029】

第9の発明は、第8の発明に従属し、修正オフセット値算出手段は、連続個数が多いほど、データ補正手段で補正後のデジタルデータに収束する度合いの強い修正オフセット値を算出する、デジタルデータ補正プログラムである。

【0030】

第9の発明でも、高精度なオフセット補正が達成できる。

10

20

30

40

50

【 0 0 3 1 】

第 1 0 の発明は、第 9 の発明に従属し、修正オフセット値算出手段は、連続個数のべき乗に比例して、データ補正手段で補正後のデジタルデータに収束する度合いの強い修正オフセット値を算出する、デジタルデータ補正プログラムである。

【 0 0 3 2 】

第 1 0 の発明でも、高精度なオフセット補正が達成できる。

【 0 0 3 3 】

第 1 1 の発明は、第 1 の発明に従属し、データ補正手段は、加重平均値を用いて補正した補正後のデジタルデータを出力するデータ出力手段を含む、デジタルデータ補正プログラムである。

10

【 0 0 3 4 】

第 1 1 の発明では、データ補正手段 (S 4 9) は、補正後のデジタルデータを出力する。つまり、データ補正手段は、たとえばステップ S 4 9 に示す計算式で計算した補正後のデータを出力する。

【 0 0 3 5 】

第 1 2 の発明は、第 1 ないし第 1 1 のいずれかの発明に従属し、センサはジャイロセンサを含み、デジタルデータは角速度データである、デジタルデータ補正プログラムである。

【 0 0 3 6 】

第 1 2 の発明では、ジャイロセンサを用いることができるので、ゲーム機などに好適する。

20

【 0 0 3 7 】

第 1 3 の発明は、第 1 2 の発明に従属し、ジャイロセンサはゲーム機のコントローラに設けられ、ゲーム機はデジタルデータに基づいてゲーム処理を実行する、デジタルデータ補正プログラムである。

【 0 0 3 8 】

第 1 3 の発明では、ジャイロセンサを用いて、精度よいゲーム処理が可能である。

【 0 0 3 9 】

第 1 4 の発明は、所定のセンサで計測したアナログ量を A / D 変換手段によって変換したデジタルデータを補正するデジタルデータ補正装置であって、A / D 変換手段で変換されたデジタルデータを順次記憶するバッファ手段、A / D 変換手段で変換された最新のデジタルデータに対する安定範囲を算出する安定範囲算出手段、バッファ手段に記憶されたデジタルデータのうち、最新のデジタルデータから遡って連続して安定範囲内にあるデジタルデータの平均値を算出する平均値算出手段、バッファ手段に記憶されたデジタルデータのうち、最新のデジタルデータから遡って連続して安定範囲内にあるデジタルデータの連続度合いを算出する連続度合い算出手段、連続度合いに基づいて、最新のデジタルデータと平均値との加重平均値を算出する加重平均値算出手段、および加重平均値算出手段で算出された加重平均値を用いて最新のデジタルデータを補正するデータ補正手段を備える、デジタルデータ補正装置である。

30

【 0 0 4 0 】

第 1 4 の発明によれば第 1 の発明と同様の効果が期待できる。

40

【 発明の効果 】

【 0 0 4 1 】

この発明によれば、デジタルデータの精度の向上と、追従性の向上という二律背反的な要求に効果的に対応できる。

【 0 0 4 2 】

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 】

【 0 0 4 3 】

50

図1を参照して、この発明の一実施例であるゲームシステム10は、ビデオゲーム装置(以下、単に「ゲーム装置」ということがある。)12およびコントローラ14を含む。コントローラ14は、ユーザないしプレイヤーの入力装置ないし操作装置(デバイス)として機能する。ゲーム装置12とコントローラ14とは無線によって接続される。たとえば、無線通信は、Bluetooth(登録商標)規格に従って実行されるが、赤外線や無線LANなど他の規格に従って実行されてもよい。

【0044】

ゲーム装置12は、略直方体のハウジング16を含み、ハウジング16の前面にはディスクスロット18および外部メモリカードスロットカバー20が設けられる。ディスクスロット18から、ゲームプログラムおよびデータを記憶した情報記憶媒体の一例である光ディスク22が挿入されて、ハウジング16内のディスクドライブ74(図4)に装着される。外部メモリカードスロットカバー20の内側には外部メモリカード用コネクタ82(図4)が設けられており、その外部メモリカード用コネクタ82には外部メモリカード(図示せず)が挿入される。外部メモリカードは、光ディスク22から読み出したゲームプログラム等をローディングして一時的に記憶したり、このゲームシステム10を利用してプレイされたゲームのゲームデータ(結果データまたは途中データ)を保存(セーブ)しておいたりするために利用される。また、上記ゲームデータの保存は、外部メモリカードに代えて、たとえばフラッシュメモリ64(図4)等の内部メモリに対して行うようにしてもよい。

【0045】

ゲーム装置12のハウジング16の後面には、AVケーブルコネクタ(図示せず)が設けられ、当該AVケーブルコネクタを用いて、ゲーム装置12はAVケーブル24を介してモニタ(ディスプレイ)26に接続される。このモニタ26は典型的にはカラーテレビジョン受像機であり、AVケーブル24は、ゲーム装置12からの映像信号をカラーテレビのビデオ入力端子に入力し、音声信号を音声入力端子に入力する。したがって、カラーテレビ(モニタ)26の画面上にたとえば3Dビデオゲームのゲーム画像が表示され、内蔵されるスピーカ28からゲーム音楽や効果音などのステレオゲーム音声が出力される。

【0046】

また、モニタ26の周辺(この実施例では、モニタ26の上側)には、2つの赤外LED(マーカ)30aおよび30bを備えるマーカ部30が設けられる。このマーカ部30は、電源ケーブル(図示せず)を通してゲーム装置12に接続される。したがって、マーカ部30には、ゲーム装置12から電源が供給される。これによって、マーカ30a、30bは発光し、それぞれモニタ26の前方に向けて赤外光を出力する。

【0047】

なお、ゲーム装置12の電源は、一般的なACアダプタ(図示せず)によって与えられる。ACアダプタは家庭用の標準的な壁ソケットに接続され、家庭用電源を、ゲーム装置12を駆動するのに適した低いDC電圧信号に変換する。他の実施例では、電源としてバッテリーが用いられてもよい。

【0048】

コントローラ14は、詳細は後述されるが、片手で把持可能なリモコン32およびそのリモコン32に着脱可能に装着されるジャイロセンサユニット34を含む。ジャイロセンサユニット34がリモコン32と物理的および電気的に結合されるので、リモコン32からは入力データないし操作データが出力されるとともに、ジャイロセンサユニット34で検出したリモコン32の角速度を示す角速度データが出力される。

【0049】

このゲームシステム10において、ゲーム(または他のアプリケーション)をプレイするために、ユーザはまずゲーム装置12の電源をオンし、次いで、ユーザはビデオゲーム(もしくは実行したいと思う他のアプリケーション)を記憶している適宜の光ディスク22を選択し、その光ディスク22をゲーム装置12のディスクスロット18からディスクドライブ74にローディングする。これに応じて、ゲーム装置12がその光ディスク22

10

20

30

40

50

に記憶されているソフトウェアに基づいてビデオゲームもしくは他のアプリケーションを実行し始めるようにする。ユーザはゲーム装置 1 2 に入力を与えるためにコントローラ 1 4 を操作する。

【 0 0 5 0 】

図 2 にはリモコンまたはリモコン 3 2 の外観の一例が示される。図 2 (A) は、リモコン 3 2 を上面後方から見た斜視図であり、図 2 (B) は、リモコン 3 2 を下面前方から見た斜視図である。

【 0 0 5 1 】

リモコン 3 2 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 3 6 を有している。ハウジング 3 6 は、その前後方向 (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 3 6 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、リモコン 3 2 を用いて、それに設けられたボタンを押下したり、リモコン 3 2 自体の位置や向きを変えることによって、ゲーム操作を行うことができる。

【 0 0 5 2 】

ハウジング 3 6 には、複数の操作ボタンが設けられる。すなわち、ハウジング 3 6 の上面には、十字キー 3 8 a、1 (いち) ボタン 3 8 b、2 (に) ボタン 3 8 c、A ボタン 3 8 d、- (マイナス) ボタン 3 8 e、ホーム (HOME) ボタン 3 8 f、および + (プラス) ボタンないしスタートボタン 3 8 g が設けられる。一方、ハウジング 3 6 の下面には凹部が形成されており、当該凹部の後方側傾斜面には B ボタン 3 8 h が設けられる。これら各ボタン (スイッチ) 3 8 a 3 8 h には、ゲーム装置 1 2 が実行するゲームプログラムに応じてそれぞれ適宜な機能が割り当てられる。また、ハウジング 3 6 の上面には、遠隔からゲーム装置 1 2 本体の電源をオン / オフするための電源スイッチ 3 8 i が設けられる。リモコン 3 2 に設けられる各ボタン (スイッチ) を、包括的に参照符号 3 8 を用いて操作手段または入力手段として示すこともある。

【 0 0 5 3 】

十字キー 3 8 a は、4 方向プッシュスイッチであり、矢印で示す 4 つの方向、前 (または上)、後ろ (または下)、右および左の操作部を含む。この操作部のいずれか 1 つを操作することによって、プレイヤーによって操作可能なキャラクタまたはオブジェクト (プレイヤキャラクタまたはプレイヤオブジェクト) の移動方向を指示したり、カーソルの移動方向を指示したり、単に方向を指示したりすることができる。

【 0 0 5 4 】

1 ボタン 3 8 b および 2 ボタン 3 8 c は、それぞれ、押しボタンスイッチである。たとえば 3 次元ゲーム画像を表示する際の視点位置や視点方向、すなわち仮想カメラの位置や画角を調整する等のゲームの操作に使用される。または、1 ボタン 3 8 b および 2 ボタン 3 8 c は、A ボタン 3 8 d および B ボタン 3 8 h と同じ操作或いは補助的な操作をする場合に用いるようにしてもよい。

【 0 0 5 5 】

A ボタンスイッチ 3 8 d は、押しボタンスイッチであり、プレイヤーキャラクタまたはプレイヤオブジェクトに、方向指示以外の動作、すなわち、打つ (パンチ)、投げる、つかむ (取得)、乗る、ジャンプするなどの任意のアクションをさせるために使用される。たとえば、アクションゲームにおいては、ジャンプ、パンチ、武器を動かすなどを指示することができる。また、ロールプレイングゲーム (R P G) やシミュレーション R P G においては、アイテムの取得、武器やコマンドの選択および決定等を指示することができる。また、A ボタンスイッチ 3 8 d は、リモコン 3 2 をポインティングデバイスとして用いる場合に、ゲーム画面上でポインタ (指示画像) が指示するアイコンないしボタン画像の決定を指示するために使用される。たとえば、アイコンやボタン画像が決定されると、これらに対応して予め設定されている指示ないし命令 (コマンド) を入力することができる。

【 0 0 5 6 】

- ボタン 3 8 e、ホームボタン 3 8 f、+ ボタン 3 8 g および電源スイッチ 3 8 i もま

10

20

30

40

50

た、押しボタンスイッチである。- ボタン 3 8 e は、ゲームモードを選択するために使用される。ホームボタン 3 8 f は、ゲームメニュー（メニュー画面）を表示するために使用される。+ ボタン 3 8 g は、ゲームを開始（再開）したり、一時停止したりするなどのために使用される。電源スイッチ 3 8 i は、ゲーム装置 1 2 の電源を遠隔操作によってオン/オフするために使用される。

【 0 0 5 7 】

なお、この実施例では、リモコン 3 2 自体をオン/オフするための電源スイッチは設けておらず、リモコン 3 2 の操作手段ないし入力手段 3 8 のいずれかを操作することによってリモコン 3 2 はオンとなり、一定時間（たとえば、3 0 秒）以上操作しなければ自動的にオフとなるようにしてある。

【 0 0 5 8 】

B ボタン 3 8 h もまた、押しボタンスイッチであり、主として、弾を撃つなどのトリガを模した入力を行ったり、リモコン 3 2 で選択した位置を指定したりするために使用される。また、B ボタン 3 8 h を押し続けると、プレイアオブジェクトの動作やパラメータを一定の状態に維持することもできる。また、一定の場合には、B ボタン 3 8 h は、通常の B ボタンと同様に機能し、A ボタン 3 8 d によって決定したアクションやコマンドなどを取り消すなどのために使用される。

【 0 0 5 9 】

ハウジング 3 6 内には、図 2 に示す X、Y および Z の 3 軸方向（すなわち左右方向、上下方向および前後方向）の加速度を検出する加速度センサ 8 4（図 5）が設けられる。なお、加速度センサ 8 4 としては、ハウジング 3 6 の形状またはリモコン 3 2 の持たせ方の限定等に応じて、上下方向、左右方向および前後方向のうちいずれか 2 方向の加速度を検出する 2 軸加速度センサが用いられてもよい。場合によっては 1 軸加速度センサが用いられてもよい。

【 0 0 6 0 】

ハウジング 3 6 の前面には光入射口 3 6 b が形成され、ハウジング 3 6 内には撮像情報演算部 4 0 がさらに設けられる。撮像情報演算部 4 0 は、赤外線撮像カメラと撮像対象の画像内での座標を算出する演算部とによって構成され、上述のマーカ 3 0 a および 3 0 b を含む被写界を赤外線で捉えて、マーカ 3 0 a および 3 0 b の被写界内における位置座標を算出する。

【 0 0 6 1 】

また、ハウジング 3 6 の後面には、コネクタ 4 2 が設けられている。コネクタ 4 2 は、リモコン 3 2 に他の機器を接続するために利用される。この実施例では、コネクタ 4 2 にはジャイロセンサユニット 3 4 のコネクタ 5 0（図 3（A））が接続される。

【 0 0 6 2 】

ハウジング 3 6 の後面にはまた、コネクタ 4 2 を左右（X 軸方向）に挟んで対向する位置に、1 対の孔 4 4 が形成されている。この 1 対の孔 4 4 には、ジャイロセンサユニット 3 4 をハウジング 3 6 の後面に固定するためのフック 5 2（図 3（A））が挿入される。ハウジング 3 6 の後面にはさらに、ストラップ（図示せず）を装着するための孔 4 6 も形成されている。

【 0 0 6 3 】

図 3 にはジャイロセンサユニット 3 4 の外観の一例が示される。図 3（A）は、ジャイロセンサユニット 3 4 を上面前方から見た斜視図であり、図 3（B）は、ジャイロセンサユニット 3 4 を下面後方から見た斜視図である。

【 0 0 6 4 】

ジャイロセンサユニット 3 4 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 4 8 を有している。ハウジング 4 8 は、略直方体形状を有しており、その長さはリモコン 3 2 のハウジング 3 6 の長さのおよそ 1 / 5、その幅および厚みはハウジング 3 6 の幅および厚みとほぼ同じである。プレイヤは、リモコン 3 2 にジャイロセンサユニット 3 4 を装着した状態で、リモコン 3 2 自体の位置や向きを変えることによって、ゲーム操作

10

20

30

40

50

を行うことができる。

【0065】

ハウジング48の前面にはコネクタ50が設けられている。コネクタ50は、このハウジング48すなわちジャイロセンサユニット34をリモコン32に一体的に装着するために、コネクタ42(図2)と嵌め合わせられる。ハウジング48の前面にはまた、コネクタ50を挟んで横方向(X軸方向)に対向する位置に、1対のフック52が設けられている。ジャイロセンサユニット34をリモコン32に装着するべく、コネクタ50をコネクタ42に接続すると、1対のフック52はハウジング36後面の1対の孔44(図2(A))に挿入され、フック52の爪がハウジング36の内壁に引っ掛かる。これによって、ジャイロセンサユニット34は、リモコン32の後面に固定される。

10

【0066】

なお、ジャイロセンサユニット34のハウジング48の後端には、必要に応じてこのジャイロセンサユニット34に別の拡張デバイスを機械的かつ電氣的に装着するための、コネクタ54が設けられる。

【0067】

また、ジャイロセンサユニット34のハウジング48内には、図5に示すジャイロセンサ56およびマイコン58が基板(図示せず)に装着されて内蔵される。このジャイロセンサ56は、一般的には、2チップ構成であり、一方のチップで2軸の角速度を検出し、他方のチップで残りの1軸の角速度を検出する。ただし、2チップ構成とは限らず、3個の1軸ジャイロセンサ(3チップ)で構成してもよく、1個の3軸ジャイロセンサ(1チップ)で構成してもよい。いずれの場合も、上述の3つの角速度を適正に検出できるように、各チップの位置や向きが決定される。マイコン58は、後述のように、ジャイロセンサ56で検出した各軸角速度に応じた出力電圧をデジタルデータに変換する。

20

【0068】

また、コントローラ14の電源は、リモコン32内に取替可能に収容されるバッテリー(図2に示すハウジング36の下面の蓋36aの中に設けられる。)によって与えられる。ジャイロセンサユニット34にはコネクタ42および50を介して電源が供給される。

【0069】

図4は図1実施例のビデオゲームシステム10の電氣的な構成を示すブロック図である。図示は省略するが、ハウジング14内の各コンポーネントは、プリント基板に実装される。図4に示すように、ゲーム装置12には、CPU60が設けられ、ゲームプロセッサとして機能する。また、CPU60には、システムLSI62が接続される。このシステムLSI62には、外部メインメモリ66、ROM/RTC68、ディスクドライブ74およびAVIC76が接続される。

30

【0070】

外部メインメモリ66は、ゲームプログラム等のプログラムを記憶したり、各種データを記憶したりして、CPU60のワーク領域やバッファ領域として用いられる。ROM/RTC68は、いわゆるブートROMであり、ゲーム装置12の起動用のプログラムが組み込まれるとともに、時間をカウントする時計回路が設けられる。ディスクドライブ74は、光ディスク18からプログラム、画像データや音声データ等を読み出し、CPU60の制御の下で、後述する内部メインメモリ62eまたは外部メインメモリ66に書き込む。

40

【0071】

システムLSI62には、入出力プロセッサ62a、GPU(Graphics Processor Unit)62b、DSP(Digital Signal Processor)62c、VRAM62dおよび内部メインメモリ62eが設けられ、図示は省略するが、これらは内部バスによって互いに接続される。入出力プロセッサ(I/Oプロセッサ)62aは、データの送受信を実行したり、データのダウンロードを実行したりする。データの送受信やダウンロードについては後述する。

【0072】

50

GPU62bは、描画手段の一部を形成し、CPU60からのグラフィクスコマンド(作画命令)を受け、そのコマンドに従ってゲーム画像データを生成する。ただし、CPU60は、グラフィクスコマンドに加えて、ゲーム画像データの生成に必要な画像生成プログラムをGPU62bに与える。

【0073】

図示は省略するが、上述したように、GPU62bにはVRAM62dが接続される。GPU62bが作画コマンドを実行するにあたって必要なデータ(画像データ:ポリゴンデータやテクスチャデータなどのデータ)は、GPU62bがVRAM62dにアクセスして取得する。ただし、CPU60は、描画に必要な画像データを、GPU62bを介してVRAM62dに書き込む。GPU62bは、VRAM62dにアクセスして描画のためのゲーム画像データを作成する。

10

【0074】

なお、この実施例では、GPU62bがゲーム画像データを生成する場合について説明するが、ゲームアプリケーション以外の任意のアプリケーションを実行する場合には、GPU62bは当該任意のアプリケーションについての画像データを生成する。

【0075】

また、DSP62cは、オーディオプロセッサとして機能し、内部メインメモリ62eや外部メインメモリ66に記憶されるサウンドデータや音波形(音色)データを用いて、スピーカ28から出力する音、音声或いは音楽に対応するオーディオデータを生成する。

【0076】

20

上述のように生成されたゲーム画像データおよびオーディオデータは、AVIC76によって読み出され、AVコネクタ78を介してモニタ26およびスピーカ28に出力される。したがって、ゲーム画面がモニタ26に表示され、ゲームに必要な音(音楽)がスピーカ28から出力される。

【0077】

また、入出力プロセッサ62aには、フラッシュメモリ64、無線通信モジュール70および無線コントローラモジュール72が接続されるとともに、拡張コネクタ80およびメモリカード用コネクタ82が接続される。また、無線通信モジュール70にはアンテナ70aが接続され、無線コントローラモジュール72にはアンテナ72aが接続される。

【0078】

30

図示は省略するが、入出力プロセッサ62aは、無線通信モジュール70を介して、ネットワークに接続される他のゲーム装置や各種サーバと通信することができる。ただし、ネットワークを介さずに、直接的に他のゲーム装置と通信することもできる。入出力プロセッサ62aは、定期的にフラッシュメモリ64にアクセスし、ネットワークへ送信する必要があるデータ(「送信データ」とする)の有無を検出し、当該送信データが有る場合には、無線通信モジュール70およびアンテナ70aを介してネットワークに送信する。また、入出力プロセッサ62aは、他のゲーム装置から送信されるデータ(「受信データ」とする)を、ネットワーク、アンテナ70aおよび無線通信モジュール70を介して受信し、当該受信データをフラッシュメモリ64に記憶する。ただし、受信データが一定の条件を満たさない場合には、当該受信データはそのまま破棄される。さらに、入出力プロセッサ62aは、ダウンロードサーバからダウンロードしたデータ(ダウンロードデータとする)をネットワーク、アンテナ70aおよび無線通信モジュール70を介して受信し、そのダウンロードデータをフラッシュメモリ64に記憶する。

40

【0079】

また、入出力プロセッサ62aは、リモコン32から送信される入力データをアンテナ72aおよび無線コントローラモジュール72を介して受信し、内部メインメモリ62eまたは外部メインメモリ66のバッファ領域に記憶(一時記憶)する。入力データは、CPU60の処理(たとえば、ゲーム処理)によって利用された後、バッファ領域から消去される。

【0080】

50

なお、この実施例では、上述したように、無線コントローラモジュール72は、Bluetooth規格に従ってリモコン32との間で通信を行う。このため、コントローラ14からデータを取得するだけでなく、ゲーム装置12からコントローラ14に所定の命令を送信し、コントローラ14の動作をゲーム装置12から制御することもできる。

【0081】

さらに、入出力プロセッサ62aには、拡張コネクタ80およびメモリカード用コネクタ82が接続される。拡張コネクタ80は、USBやSCSIのようなインターフェースのためのコネクタであり、外部記憶媒体のようなメディアを接続したり、リモコン32とは異なる他のコントローラのような周辺機器を接続したりすることができる。また、拡張コネクタ80に有線LANアダプタを接続し、無線通信モジュール70に代えて当該有線LANを利用することもできる。メモリカード用コネクタ82には、メモリカードのような外部記憶媒体を接続することができる。したがって、たとえば、入出力プロセッサ62aは、拡張コネクタ80やメモリカード用コネクタ82を介して、外部記憶媒体にアクセスし、データを保存したり、データを読み出したりすることができる。

10

【0082】

詳細な説明は省略するが、電源ボタンがオンされると、システムLSI62には、ゲーム装置12の各コンポーネントに図示しないACアダプタを経て電源が供給され、通常の通電状態となるモード(「通常モード」と呼ぶこととする)が設定される。一方、電源ボタンがオフされると、システムLSI62には、ゲーム装置12の一部のコンポーネントのみに電源が供給され、消費電力を必要最低限に抑えるモード(以下、「スタンバイモード」という)が設定される。

20

【0083】

この実施例では、スタンバイモードが設定された場合には、システムLSI62は、入出力プロセッサ62a、フラッシュメモリ64、外部メインメモリ66、ROM/RTC68および無線通信モジュール70、無線コントローラモジュール72以外のコンポーネントに対して、電源供給を停止する指示を行う。したがって、この実施例では、スタンバイモードにおいて、CPU60がアプリケーションを実行することはない。

【0084】

なお、システムLSI62には、スタンバイモードにおいても電源が供給されるが、GPU62b、DSP62cおよびVRAM62dへのクロックの供給を停止することにより、これらを駆動しないようにして、消費電力を低減するようにしてある。

30

【0085】

また、図示は省略するが、ゲーム装置12のハウジング14内部には、CPU60やシステムLSI62などのICの熱を外部に排出するためのファンが設けられる。スタンバイモードでは、このファンも停止される。

【0086】

ただし、スタンバイモードを利用したくない場合には、スタンバイモードを利用しない設定にしておくことにより、電源ボタンがオフされたときに、すべての回路コンポーネントへの電源供給が完全に停止される。

【0087】

また、通常モードとスタンバイモードとの切り替えは、リモコン32の電源スイッチ38iのオン/オフの切り替えによって、遠隔操作によって行うことが可能である。当該遠隔操作を行わない場合には、スタンバイモードにおいて無線コントローラモジュール72aへの電源供給を行わない設定にしてもよい。

40

【0088】

リセットボタンもまた、システムLSI62に接続される。リセットボタンが押されると、システムLSI62は、ゲーム装置12の起動プログラムを再起動する。イジェクトボタンは、ディスクドライブ74に接続される。イジェクトボタンが押されると、ディスクドライブ74から光ディスク22が排出される。

【0089】

50

図5には、リモコン32とジャイロセンサユニット34とが接続されたときのコントローラ14全体の電氣的構成の一例が示される。図5に示す通信部86は、マイクロコンピュータ(マイコン)88、メモリ90、無線モジュール92およびアンテナ94を含む。マイコン88は、処理の際にメモリ90を記憶領域(作業領域やバッファ領域)として用いながら、無線モジュール92を制御して、取得したデータをゲーム装置12に送信したりゲーム装置12からのデータを受信したりする。

【0090】

ジャイロセンサユニット34から通信部86に出力されたデータは、マイコン88を経て一時的にメモリ90に格納される。リモコン32内の操作部38、撮像情報演算部40および加速度センサ84から通信部86に出力されたデータもまた、メモリ90に一時的に格納される。マイコン88は、ゲーム装置12への送信タイミングが到来すると、メモリ90に格納されているデータをコントローラデータとして無線モジュール92へ出力する。コントローラデータには、図10に示したジャイロデータ(角速度データ)が含まれる。

【0091】

無線モジュール92は、Bluetoothのような近距離無線通信技術を用いて、所定周波数の搬送波をコントローラデータで変調し、その微弱電波信号をアンテナ94から放射する。つまり、コントローラデータは、無線モジュール92で微弱電波信号に変調されてリモコン32から送信される。微弱電波信号はゲーム装置12側の無線コントローラモジュール(Bluetooth通信ユニット)72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はコントローラデータを取得することができる。ゲーム装置12のCPU60は、コントローラ14から取得したコントローラデータに基づいてゲーム処理を行う。なお、リモコン32とゲーム装置12との無線通信は、無線LANなど他の規格に従って実行されてもよい。

【0092】

リモコン32は、操作部38すなわち上述の操作ボタンないし操作スイッチ38a 38iが操作されると、その操作を示すデータが通信部86に出力される。撮像情報演算部40からは、マーカ30aおよび30bの被写界内における位置座標を示すデータが通信部86に出力される。

【0093】

図6は、リモコン32を用いてゲームプレイするときの状態を概説する図解図である。ただし、ゲームプレイのみならず、他のアプリケーションを実行したり、DVDを再生したりする場合も同様である。図6に示すように、ビデオゲームシステム10でリモコン32を用いてゲームをプレイする際、プレイヤは、一方の手でリモコン32を把持する。厳密に言うと、プレイヤは、リモコン32の先端面(撮像情報演算部40が撮像する光の入射口36b側)がマーカ30aおよび30bの方を向く状態でリモコン32を把持する。ただし、図1からも分かるように、マーカ30aおよび30bは、モニタ26の画面の横方向と平行に配置されている。この状態で、プレイヤは、リモコン32が指示する画面上の位置を変更したり、リモコン32と各マーカ30aおよび30bとの距離を変更したりすることによってゲーム操作を行う。

【0094】

撮像情報演算部40は、赤外線フィルタ40a、レンズ40b、撮像素子40cおよび画像処理回路40dによって構成される。赤外線フィルタ40aは、リモコン32の前方から入射する光から赤外線のみを通過させる。上述したように、モニタ26の表示画面近傍(周辺)に配置されるマーカ30aおよび30bは、モニタ26の前方に向かって赤外光を出力する赤外LEDである。したがって、赤外線フィルタ40aを設けることによってマーカ30aおよび30bの画像をより正確に撮像することができる。レンズ40bは、赤外線フィルタ40aを透過した赤外線を集光して撮像素子40cへ出射する。撮像素子40cは、たとえばCMOSセンサあるいはCCDのような固体撮像素子であり、レンズ40bによって集光された赤外線を撮像する。したがって、撮像素子40cは、赤外線

10

20

30

40

50

フィルタ40aを通過した赤外線だけを撮像して画像データを生成する。撮像素子40cによって生成された画像データは、画像処理回路40dで処理される。画像処理回路40dは、撮像画像内における撮像対象(マーカ30aおよび30b)の位置を算出し、所定時間毎に、当該位置を示す各座標値をマーカ座標データとしてプロセッサ70に出力する。

【0095】

なお、図6では分かり難いが、上述したジャイロユニット34がリモコン32に接続される場合も同様である。

【0096】

一例として、撮像素子40cによって撮像された撮像画像の解像度が126×96であるとき、マーカ座標は、(0,0)から(1024,768)までの整数値で表現される。出力されたマーカ座標データは、上述したように、プロセッサ70によって入力データに含まれ、ゲーム装置12に送信される。

10

【0097】

ゲーム装置12(CPU60)は、受信した入力データからマーカ座標データを検出すると、このマーカ座標データに基づいて、モニタ26の画面上におけるリモコン32の指示位置(指示座標)と、リモコン32からマーカ30aおよび30bまでの各距離とを算出することができる。具体的には、2つのマーカ座標の midpoint の位置から、リモコン32の向いている位置すなわち指示位置が算出される。また、撮像画像における対象画像間の距離が、リモコン32と、マーカ30aおよび30bとの距離に応じて変化するので、2つのマーカ座標間の距離を算出することによって、ゲーム装置12はリモコン32と、マーカ30aおよび30bとの間の距離を把握できる。

20

【0098】

図5に戻って、加速度センサ84が検出した加速度を示すデータもまた、通信部86へ出力される。加速度センサ84は、たとえば最大100フレーム/秒程度のサンプリング周期を有する。

【0099】

コネクタ42には、ジャイロセンサユニット34のコネクタ50が接続される。ジャイロセンサユニット34は、その内部にジャイロセンサ56およびマイコン58を含む。ジャイロセンサ56は、たとえば加速度センサ84と同様のサンプリング周期を有する。マイコン58は、ジャイロセンサ56が検出した角速度を示すデータをコネクタ50およびコネクタ42を介して通信部86に出力する。

30

【0100】

なお、上述した通信部86への各出力は、たとえば1/200秒周期で実行される。したがって、任意の1/200秒間に、操作部38からの操作データと、撮像情報演算部40のからの位置座標データと、加速度センサ84からの加速度データと、ジャイロセンサ56からの角速度データとが、1回ずつ通信部86に出力される。

【0101】

図7には、図5に示した全体構成のうちジャイロセンサユニット34の要部構成が示される。上述のコネクタ42およびコネクタ50は、それぞれたとえば6ピンのコネクタであり、この6ピンの中に、コネクタ間の接続状態を示す変数“Attach”を制御するためのAttachピンが含まれている。Attachは、コネクタ42・コネクタ50間つまりリモコン32・ジャイロセンサユニット34間が切断されていることを示す“Low”と、コネクタ42・コネクタ50間つまりリモコン32・ジャイロセンサユニット34間が接続されていることを示す“High”との間で変化する。

40

【0102】

上記6ピンのうち他の2つにはI2Cバスが割り当てられており、このI2Cバスから、図8に示すジャイロセンサユニット34からの出力データ(角速度データ)がリモコン32へ入力される。

【0103】

50

ジャイロセンサユニット34のマイコン58は、A/D変換回路58aおよび58bを含んでおり、ジャイロセンサ56から出力される3軸周りの角速度信号は、これらA/D変換回路58aおよび58bの各々に与えられる。A/D変換回路58aでは、ジャイロセンサ56の検出範囲の全部(たとえば ± 360 度/秒)を対象とする高角速度モードのA/D変換処理が実行され、A/D変換回路58bでは、ジャイロセンサ56の検出範囲の一部(たとえば ± 90 度/秒)を対象とする低角速度モードのA/D変換処理が実行される。マイコン58は、これら2種類のA/D変換結果のいずれか一方を角速度データとして出力する。

【0104】

具体的には、A/D変換回路58aおよび58bからある時刻に対応する2種類の角速度データが出力されると、マイコン58は最初、このうち低角速度モードの角速度データについて、その値Aが第1閾値 T_{h1} から第2閾値 T_{h2} ($> T_{h1}$)までの範囲内にあるかどうか、すなわち条件“ $T_{h1} < A < T_{h2}$ ”が満足されるか否かを、軸毎に、つまりヨー、ロールおよびピッチの各々について判定する。次に、これら3つの判定結果に基づいて、低角速度モードおよび高角速度モードのいずれか1つを選択する。たとえば、3つの判定結果それぞれに関して、YESであれば低角速度モードを選択し、NOであれば高角速度モードを軸ごとに選択する。そして、軸ごとに選択されたモードに従う角速度データを、選択されたモードを示すモード情報と共にそれぞれ出力する。つまり、角速度に応じてデータの精度を変えることによって、同じデータ量であっても、低速のときにはより精度の高いデータを出力することができる。

【0105】

図8には、ジャイロセンサユニット34が取り扱うデータのフォーマットが示される。ジャイロセンサユニット34用のデータは、ヨー角速度データ、ロール角速度データおよびピッチ角速度データと、ヨー角速度モード情報、ロール角速度モード情報およびピッチ角速度モード情報とを含む。ただし、図9に示すように、y軸の回転がヨー角で表わされ、x軸の回転がピッチ角で表わされ、そして、z軸の回転がロール角で表わされるものとする。

【0106】

ヨー角速度データ、ロール角速度データおよびピッチ角速度データは、ジャイロセンサ56から出力されるヨー角速度信号、ロール角速度信号およびピッチ角速度信号をA/D変換して得られる、たとえば各14ビットのデータである。ヨー角速度モード情報、ロール角速度モード情報およびピッチ角速度モード情報は、それぞれ対応する角速度データのモードを示す各1ビットの情報であり、高角速度モードに対応する“0”と、低角速度モードに対応する“1”との間で変化する。

【0107】

ジャイロセンサユニット34は通信部86に、図8のフォーマットに従うジャイロ用データを、たとえば1/100秒周期で出力する。これはゲームの処理等で一般的な処理期間である1/60秒周期よりも十分短いことから、ゲーム処理でそのデータを完全に利用することができる。

【0108】

この図8に示すようなジャイロデータが、後述の角速度データdataとして、リモコン32からゲーム機12に送信される。

【0109】

このゲームシステム10では、ボタン操作だけでなく、コントローラ14自体を動かすことによって、ゲームなどのアプリケーションに対する入力を行うことができる。ゲームをプレイする際には、リモコン32には3軸方向の加速度を検出する加速度センサ84が内蔵され、リモコン32にジャイロセンサユニット34が装着されている場合には、リモコン32自身の動きを示す3軸周りの角速度がさらに検出される。

【0110】

これらの検出値は、先述したコントローラデータの態様でゲーム装置12に送信される

10

20

30

40

50

。ゲーム装置 1 2 (図 4) では、コントローラ 1 4 からのコントローラデータは、入出力プロセッサ 6 2 a によってアンテナ 7 2 a および無線コントローラモジュール 7 2 を介して受信され、受信されたコントローラデータは、内部メインメモリ 6 2 e または外部メインメモリ 6 6 のバッファ領域に書き込まれる。CPU 4 4 は、内部メインメモリ 6 2 e または外部メインメモリ 6 6 のバッファ領域に格納されたコントローラデータを読み出し、このコントローラデータから検出値つまりコントローラ 1 4 によって検出された加速度および/または角速度の値を復元する。

【 0 1 1 1 】

なお、角速度データには高角速度および低角速度の 2 つのモードがあるため、これら 2 つのモードにそれぞれ対応する 2 種類の角速度復元アルゴリズムが準備される。角速度データから角速度の値を復元するにあたっては、当該角速度データのモードに対応する角速度復元アルゴリズムが、角速度モード情報に基づいて選択される。

10

【 0 1 1 2 】

CPU 6 0 はまた、このような復元処理と並行して、復元された加速度からコントローラ 1 4 の速度を計算する処理を実行してもよい。さらに並行して、計算された速度からコントローラ 1 4 の移動距離ないし位置を求めることもできる。一方、復元された角速度からは、コントローラ 1 4 の回転角が求まる。なお、加速度を積算して速度を求めたり、角速度を積算して回転角を求めたりする際の初期値 (積分定数) は、たとえば、撮像情報演算部 4 0 のからの位置座標データに基づいて計算できる。位置座標データはまた、積算によって蓄積されていく誤差の修正にも用いることができる。

20

【 0 1 1 3 】

ゲーム処理は、こうして求められた加速度、速度、移動距離、角速度および回転角などの変数に基づいて実行される。したがって、上記の処理は全てを行わなくともよく、ゲーム処理に必要な変数を適宜算出すればよい。なお、角速度や回転角も、原理的には加速度から計算し得るが、そのためには、ゲームプログラムに複雑なルーチンが必要で、CPU 6 0 にも重い処理負荷がかかる。ジャイロセンサユニット 3 4 を利用することで、プログラム開発が容易になり、CPU 6 0 の処理負荷も軽減される。

【 0 1 1 4 】

図 1 0 は図 2 に示した内部メインメモリ 6 2 e または外部メインメモリ 6 6 のメモリマップを示す図解図である。図 1 0 に示すように、メインメモリ (6 2 e , 6 6) は、プログラム記憶領域 9 6 およびデータ記憶領域 9 8 を含む。なお、データ記憶領域 9 8 の具体的な内容については、図 1 1 に示す。

30

【 0 1 1 5 】

プログラム記憶領域 9 6 には、ゲームプログラムが記憶され、ゲームプログラムは、ゲームメイン処理プログラム 9 6 a 、画像生成プログラム 9 6 b 、画像表示プログラム 9 6 c 、角速度検出プログラム 9 6 d および加速度検出プログラム 9 6 e などによって構成される。

【 0 1 1 6 】

ゲームメイン処理プログラム 9 6 a は、この実施例の仮想ゲームのメインルーチンを処理するためのプログラムである。画像生成プログラム 9 6 b は、後述する画像データ 9 8 a (図 1 1 参照) を用いて、モニタ 2 6 にゲーム画面を表示するためのゲーム画像を生成するプログラムである。画像表示プログラム 9 6 c は、画像生成プログラム 9 6 b に従って生成されたゲーム画像をゲーム画面としてモニタ 2 6 に表示するためのプログラムである。

40

【 0 1 1 7 】

角速度検出プログラム 9 6 d は、ジャイロセンサ 5 6 で検出された角速度についての角速度データを検出するためのプログラムである。上述したように、角速度データは、リモコン 3 2 からの入力データに含まれているため、CPU 6 0 は、角速度検出プログラム 9 6 d に従って、リモコン 3 2 からの入力データに含まれる角速度データを検出する。ただし、この角速度検出プログラム 9 6 d は、後述の図 1 4 に示す補正プログラムを含む。

50

【0118】

加速度検出プログラム96eは、加速度センサ84で検出された加速度についての加速度データを検出するためのプログラムである。上述したように、加速度データは、リモコン32からの入力データに含まれているため、CPU60は、加速度検出プログラム96eに従って、リモコン32からの入力データに含まれる1つまたは2つの加速度データを検出するのである。

【0119】

なお、図示は省略するが、ゲームプログラムには、音出力プログラムやバックアッププログラムなども含まれる。音出力プログラムは、音（音楽）データを用いて、音楽（BGM）、オブジェクトの音声ないし擬声音、効果音のようなゲームに必要な音を出力するためのプログラムである。バックアッププログラムは、ゲームデータ（途中データ、結果データ）をメモリカードにセーブ（保存）するためのプログラムである。

10

【0120】

また、図11に示すように、データ記憶領域98には、画像データ98a、角速度データ98bおよび加速度データ98cなどの各種データが一時的に記憶される。また、データ記憶領域98には、タイマやレジスタさらには必要なフラグのためのフラグレジスタ領域98d、FIFO（先入れ先出し）バッファ100等が適宜設けられる。

【0121】

画像データ98aは、ゲーム画像を生成するための画像データであり、プレイヤーオブジェクトやノンプレイヤーオブジェクトなどのオブジェクト画像データ、さらにはポリゴンデータやテクスチャデータなどを含む。角速度データ98bは、角速度検出プログラム96dに従って検出し、補正プログラムに従って補正した角速度データである。ただし、この実施例では、1フレーム毎に、3ないし4個の角速度データが検出される。加速度データ98cは、加速度検出プログラム96eに従って検出されたリモコン32の加速度データである。角速度や加速度のデータは、フレーム毎に姿勢を算出するために用いられるが、補正等の目的のために複数フレーム期間分の個数（たとえば20個）を記憶しておくようにしてもよい。

20

【0122】

データ記憶領域98には、図12に詳細に示す、FIFOバッファ100が形成される。このFIFOバッファ100は、複数段の直列に接続されたデータバッファを含み、各段（記憶場所）がバッファインデックスidxによって指定される。データバッファには、角速度データ、加速度データ等のデジタルデータが格納される。

30

【0123】

図4に示したCPU60が実行するこの実施例の処理について、図13 図14を参照して説明する。図13に示すゲームメイン処理の最初のステップS1において、図11に示すデータ記憶領域98のフラグレジスタ領域98dに設定されている修正オフセット値ofsを初期化する。修正オフセット値ofsは、温度ドリフト等により、ジャイロセンサ56から出力される角速度がゼロであることを示すアナログ値に対応付けられたデジタルの整数値データが、実際に速度がゼロであることを示すアナログ値を変換したデジタルの整数値データとの間に差異が生じた際に、その差異をなくす（減らす）ための補正に用いられるオフセット値（ゼロ点オフセット値）である。この初期化の際に格納される初期値（デバイス固有値）は、たとえば、ジャイロセンサユニット34に含まれる個々のジャイロセンサ56に固有のもので、ジャイロセンサ56の工場出荷時に設定されている。個々では、そのデバイス固有の初期値を修正オフセット値ofsとして設定する。たとえば、工場で25°Cで計測した値が初期値となる。工場出荷後に、所定の方法により更新可能であってもよい。

40

【0124】

続くステップS3において、CPU60は、FIFOバッファ100の各データバッファを指定するためのバッファインデックスidxを初期化する。一例としては、バッファインデックスidxに「0」を設定する。ただし、この初期値は、「0」から「BUFS-1」の

50

範囲のどの値であってもよい。ここで、BUFSとは、F I F Oバッファ100の大きさを示す数値であり、たとえばフラグレジスタ領域98dにバッファ容量BUFSとして記憶されている。たとえば、バッファ容量BUFSが「400」で設定された大きさのF I F Oバッファ100であれば、バッファインデックスidxの初期値は「0」から「399(400-1)」の範囲で設定される。

【0125】

その後、ステップS5で、F I F Oバッファ100内を初期化する。たとえば、「-100」のように、ジャイロセンサ56から得られる値の範囲から定数C1(たとえば、「1」)よりも十分離れた値を設定する(buf[0-(BUFS-1)] - 100)。

【0126】

そして、ステップS7において、図14に示す角速度データ取得プログラムに従って角速度データを取得する。

【0127】

その後、ステップS11では、ゲーム処理中に、CPU60は、リモコン32すなわちジャイロセンサユニット34を等速で移動させる操作を行う可能性がある場面かどうかを判断する。たとえば射撃ゲームにおいて、弾を装填した銃や、矢を番えた弓によって標的を狙うときなどにおいては、プレイヤーは銃口や弓をたとえば左から右へゆっくり動かすような操作をする。銃口や弓による狙いをリモコン32で操作する場合、リモコン32を実空間内でゆっくり動かす。その場合には、リモコン32すなわちジャイロセンサユニット34を実空間内で等速操作される可能性がある。ステップS11では、現在の場面がそのような等速操作を行う可能性がある場面かどうか判断する。これは、予めゲーム内の場面ごとに設定されたフラグにより判定しても良いし、リモコン32の入力モードによって判定しても良いし、リモコン32からの入力データによって判定しても良いし、その他にリモコン32を等速操作を判定できるような条件であれば何で判定しても良い。

【0128】

このステップS11で“YES”を判断したとき、CPU60は、ステップS13でオフセット補正フラグcfgを0に設定してステップS15に進み、また、ステップS11で“NO”なら、ステップS14でオフセット補正フラグcfgを1に設定してステップS15に進む。オフセット補正フラグcfgとは、修正オフセット値ofsを補正する必要があるかどうかを示すフラグで、修正オフセット値ofsを補正する必要がない場合(等速操作を行う可能性がある場合)は0が、修正オフセット値ofsを補正する必要がある場合(等速操作を行う可能性がない場合)は1が設定される。

【0129】

ついで、ステップS15でゲーム終了かどうか判断する。たとえば、ゲームオーバーになったか、プレイヤーがゲームを終了するようにリモコン32を操作したとき、ステップS15で“YES”となり、ゲームを終了する。ステップS15で“NO”なら、先のステップS7に戻って角速度データの取得およびゲーム処理等を引き続き行う。

【0130】

先のステップS7における角速度データ取得処理を図14を参照して詳しく説明する。ただし、この図14では1軸の角速度取得処理についてのみ図示しているが、もし、軸が図9に示すように複数設定されている場合には、すべての軸について同様の処理を実行するというを、予め理解されたい。

【0131】

図14の最初のステップS31で、CPU60は、バッファインデックスidxをインクリメントする(idx (idx+1)%BUFS)。ここで、記号「%」は、剰余を示す演算子であり、モジュロ演算が行われる。たとえば、バッファ容量BUFSが「400」で設定された大きさのF I F Oバッファ100で処理が行われていれば、バッファインデックスidxは「0」から「399」の範囲を1ずつ増加するように更新(インクリメント)される。ただし、モジュロ演算が行われることにより、「399」は「0」に更新される。このように剰余を用いてバッファインデックスidxを更新している理由は、容量が予め定められたループ

10

20

30

40

50

バッファであるFIFOバッファ100に含まれる複数段の直列に接続されたデータバッファを順序立てて効率よく利用するために行うバッファインデックスidxの更新処理を効率的に行う為である。

【0132】

次のステップS33で、CPU60は、入/出力プロセッサ62aによって受け取った、コントローラ14すなわちリモコン32からの角速度データ(整数値)を取得して、フラグレジスタ領域98d(図11)に含まれる角速度データdataにその角速度データを格納する。

【0133】

続くステップS35において、CPU60は、角速度データdataに格納した角速度データ(最新の角速度データ)をステップS31でインクリメントしたバッファインデックスidxで示されるFIFOバッファ100のデータバッファに格納する(buf[idx] data)。

10

【0134】

次のステップS37で、CPU60は、過去のデータを調べる準備をする。つまり、この実施例では、コントローラ14すなわちリモコン32から取得した角速度データの補正及びジャイロセンサ56の修正オフセット値ofsのずれを補正するために行われる、角速度データdataに格納した角速度データ(最新の角速度データ)と過去の角速度データ(FIFOバッファ100のデータバッファに格納された角速度データ)との比較処理の準備(変数やカウンタの初期化または設定)をこのステップS37で実行する。

【0135】

20

CPU60は、まず、フラグレジスタ領域98dに形成されている連続個数ctに「1」を設定する。この連続個数ctは、連続数をカウントするためのカウンタであって、この連続個数ctが示す値は、後述の一定条件(ステップS39)を満たす値の入ったデータバッファが何回連続検出できたかを示す。同じくフラグレジスタ領域98dに形成される合計値sumに、ステップS33で角速度データdataに書き込んだ角速度データを格納する。つまり、初期状態としてステップS33で角速度データdataに書き込んだ角速度データが合計値sumに格納される。そして、定数C1を用いて、安定範囲(下限)d1および安定範囲(上限)d2(ともにフラグレジスタ領域98dに形成される。)を設定する。つまり、安定範囲(下限)d1に下限値data-C1を設定し、安定範囲(上限)d2に上限値data+C1を設定する。なお、安定範囲(下限)d1および安定範囲(上限)d2にそれぞれ設定する下限値および上限値によって、先のステップS33で書き込んだ角速度データdataを中心として上下均等な範囲(上下C1分の幅)を設定することになる。他の例では、定数C1は変数として、角速度データdataや連続個数ct等に応じて変化させても良いし、その他の状況に応じて変化させても良い。

30

【0136】

ステップS37における変数iは、ステップS35で角速度データを書き込んだバッファインデックスidxで示される位置のFIFOバッファ100のデータバッファから遡って調べるデータバッファの位置を指定するための値であって、フラグレジスタ領域98dに設定されている。そして、この変数iを「1」ずつ減らす方向にずらして((idx-1+BUFS)%BUFS)、その変数が指定するデータバッファ内の値を調べ、変数iが変数i2になったとき、その遡り検索を終了する。ただし、変数i2は(idx-CMAX+BUFS)%BUFSで示される。ここで、最大検索数CMAXで示す数値は、最大何個のデータを遡って調べるかを示す数値である。つまり、最大検索数CMAX分の検索を行った際に、変数iがずらされた位置を示す値が変数i2となる。したがって、バッファ容量BUFSはこの最大検索数CMAXより大きくなければならない(BUFS > CMAX)。たとえば、最大検索数CMAXはCMAX=400が設定される。このように剰余を用いて変数iおよび変数i2を設定している理由は、バッファインデックスidxを更新する際に剰余を用いている理由と同じである。また、変数iを設定する際にバッファ容量BUFSを加算している(+BUFS)理由は、バッファインデックスidxが0の場合には「idx-1」がマイナス(-)になってしまい、マイナスの値に対してモジュロ演算を行うと正しい計算結果を得ることができないためである。

40

50

【 0 1 3 7 】

次のステップS 3 9からS 4 5においては、角速度データdataに格納した角速度データ(最新の角速度データ)と過去の角速度データ(F I F Oバッファ1 0 0のデータバッファに格納された角速度データ)との比較処理が行われる。まず、C P U 6 0は、ステップS 3 9において、ステップS 3 7でインクリメントした変数iで指定されるデータバッファ内の値が、安定範囲(下限)d1および安定範囲(上限)d2で設定された安定範囲内にあるかどうかを判断する($d1 \leq buf[i] \leq d2$)。もし、ステップS 3 9で“ Y E S ”が判断されるとき、つまり、変数iで指定されるデータバッファ内の値が安定範囲(下限)d1および安定範囲(上限)d2で設定された安定範囲内に含まれる(角速度データdataに近い値である)ときは、続くステップS 4 1において、上記安定範囲内に含まれる値つまり、その変数iで指定されるデータバッファ内の値を合計値sumに加算する。つまり、変数iで指定されるデータバッファ内の値が上記安定範囲の条件(安定範囲(下限)d1および安定範囲(上限)d2で設定された安定範囲内に含まれる)を満たすときは、そのデータバッファの値によって合計値を更新する。その後、ステップS 4 3において、連続個数ctをインクリメントする。これにより、上記安定範囲の条件を連続して満たすデータバッファの個数が連続個数ctによってカウントされる。さらにこのステップS 4 3においては、上記変数iを「1」減らす方向にずらして更新する($i = (i-1+BUFS)\%BUFS$)。変数iを更新する際にバッファ容量BUFSを加算している(+BUFS)理由は、ステップS 3 7で変数iの設定を行う時と同様の理由である。そして、ステップS 4 5において、最大検索数C MAX分の検索を行ったかどうかを判断する。すなわち、変数iが更新されて、変数i2に達したかどうかを判断している。ここで、最大検索数C MAX分の検索を行っていれば“ Y E S ”が判断され、ステップS 4 7に進む。一方、最大検索数C MAX分の検索を行っていなければ“ N O ”が判断され、その場合には、先のステップS 3 9に戻って検索を継続する。

10

20

【 0 1 3 8 】

これに対して、ステップS 3 9で“ N O ”が判断されるときは、最大検索数C MAX分の検索を行う前に、データバッファ内の値で安定範囲(下限)d1および安定範囲(上限)d2で設定された安定範囲外の値が検出されたことを意味し、そのことは、換言すれば、リモコン3 2すなわちジャイロセンサユニット3 4に速度変化の大きい動きを与えられた(急に動かされた)ことを意味する。この場合には、ステップS 3 9からステップS 4 7に進む。

30

【 0 1 3 9 】

以上のステップS 3 9からステップS 4 5を繰り返すことにより、連続個数ctには、角速度データdataが格納されたデータバッファから遡って連続して安定範囲(下限)d1および安定範囲(上限)d2で設定された安定範囲内に収まると判断されたデータバッファの数が格納され、合計値sumには、そのデータバッファ内の値の合計値が格納されていることになる。つまり、ステップS 3 5で角速度データ書き込んだデータバッファの位置から遡って、新しいデータバッファから古いデータバッファへ順に、連続して上記安定範囲内($d1 \leq buf[i] \leq d2$)に含まれるデータバッファを繰り返し取得して、その個数(連続個数ct)及び合計値(合計値sum)を算出する(ただし、検索する上限数は最大検索数C MAX)。

40

【 0 1 4 0 】

ステップS 3 9で“ N O ”が判断されたか、ステップS 4 5で“ Y E S ”が判断された後に、ステップS 4 7において、C P U 6 0は、そのときのリモコン3 2すなわちジャイロセンサユニット3 4の静止具合値(静止の程度)aを求める。この静止具合値aは「0」～「1」の範囲内の数値であり、リモコン3 2すなわちジャイロセンサユニット3 4の動きの変化が少ない(安定している)期間が長ければ長いほど「1」に近づく値で、連続個数ctでカウントした連続個数が最大検索数C MAXと等しいとき($ct=C MAX$ のとき)、すなわち角速度データdataが格納されたデータバッファから遡って連続して最大検索数C MAX分のデータバッファの値がすべて安定範囲(下限)d1および安定範囲(上限)d2で設定された範囲内に入っていたとき(ステップS 4 5で“ Y E S ”が判断されたとき)、最大値「1」になるようにして正規化している。

50

【 0 1 4 1 】

具体的には、静止具合値 a は、最大検索数 C_{MAX} に対する連続個数 ct の割合 ($ct \div C_{MAX}$) で決まる。たとえば、最大検索数 C_{MAX} が 4 0 0 のときに 3 9 9 個目のデータでステップ S 3 9 にて “ N O ” に分岐した場合 ($ct=3 9 9$) は、静止具合値 a は $3 9 9 / 4 0 0$ になる。また、最大検索数 C_{MAX} が 4 0 0 のときに 1 個目のデータでステップ S 3 9 にて “ N O ” に分岐した場合 ($ct=1$) は、静止具合値 a は $1 / 4 0 0$ になる。つまり、静止具合値 a は、角速度データ $data$ が格納されたデータバッファから遡って連続してどれくらい前までデータバッファの値が上記安定所定範囲内に収まっていたか (安定していたか) を示す指標である。

【 0 1 4 2 】

そして、さらに、 k 乗する ($k > 1$) (a^k) ことによって、変動の少ない状態が長く続いたとき、つまりステップ S 3 9 で “ Y E S ” と判断される回数 (連続個数 ct の値) が多い (大きい) ときと少ない (小さい) ときとの差を大きくする。すなわち、静止具合値 a をそのまま使うと連続個数 ct に比例するだけであるが、べき乗たとえば、 k 乗することによって、連続個数 ct に対して累乗的に静止具合値 a が変化するように設定でき、連続個数 ct を用いた処理を行う場合に、連続個数 ct が大きいときと小さいときとの差を明確に出すことができる。実施例では、 $k=3 2$ に設定した。ただし、 $k=1$ の時ように、静止具合値 a が連続個数 ct に比例するような値であっても実施可能である。さらに他の例では、 k を変数として、角速度データ $data$ や連続個数 ct 等に応じて変化させても良いし、その他の状況に応じて変化させても良い。

【 0 1 4 3 】

次に、ステップ S 4 9 において、C P U 6 0 は、静止具合値 a に応じた平均値を用いて、角速度データ $data$ の補正を行う ($data - data + (sum \div ct - data) \times a$)。つまり、合計値 sum に格納している合計値を連続個数 ct で割り、その商から角速度データ $data$ を減じ、その結果に静止具合値 a を乗算して角速度データ $data$ に加算する。このことは換言すれば、静止具合値 a の値に応じて、角速度データ $data$ を採用するか平均値 ($sum \div ct$) を採用するかその重み付けが決まるということである。したがって、静止具合値 a が大きければ大きいほど平均値の重みが重くなるように (平均値の影響が大きくなるように)、平均値と角速度データ $data$ の加重平均値が算出される。また、静止具合値 a が小さければ小さいほど角速度データ $data$ の重みが重くなるように (角速度データ $data$ の影響が大きくなるように)、平均値と角速度データ $data$ の加重平均値が算出される。この算出された加重平均値で角速度データ $data$ が補正される。つまり、静止具合値 a が大きければ大きいほど、すなわち、リモコン 3 2 すなわちジャイロセンサユニット 3 4 の動きの変化が少ない (安定している) 期間が長ければ長いほど、平均値の重みが重くなる (平均値の影響が大きくなる) ので、角速度データ $data$ は、精度が高く、ぶれを抑えた滑らかな信頼度の高い値に補正される。また、静止具合値 a が小さければ小さいほど、すなわち、リモコン 3 2 すなわちジャイロセンサユニット 3 4 に変化が大きい (速度変化が大きい) 動きが与えられた際には、角速度データ $data$ の重みが重くなる (角速度データ $data$ の影響が大きくなる) ので、角速度データ $data$ は、変化の大きい動きを迅速に反映するような追従性の高い (レスポンスのよい) 値に補正される。

【 0 1 4 4 】

ただし、ステップ S 4 9 では、上記の計算を行うことにより角速度データ $data$ を上書きして補正するようにしたが、角速度データ $data$ はそのままの状態を更新を行わずに、補正後のデータを別途出力するようにすることも考えられ、コネクタ等で用いられる場合には、このステップ S 4 9 で補正を行った角速度データ $data$ を別途出力しても良い。このように、別途出力を行うことにより、角速度データ $data$ を別の目的で利用することが可能となる。

【 0 1 4 5 】

また、このステップ S 4 9 では、静止具合値 a を用いることにより重み付けを行っているが、ステップ S 3 9 において、角速度データ $data$ が格納されたデータバッファから遡っ

10

20

30

40

50

て連続して安定範囲（下限）d1および安定範囲（上限）d2で設定された安定範囲内に収まると判断されたデータバッファのみが、連続回数ctと合計値sumに影響を与え、変化の大きい動きが行われた際のデータは影響を与えにくい為、常に静止具合値aを1として、すなわち角速度データdataに平均値（ $sum \div ct$ ）を代入することにより、角速度データdataの補正を行っても良い。更には、静止具合値aに対して所定の閾値を設けて、静止具合値aが所定の閾値を上回ったときのみ、ステップS49による補正を行ったり、逆に静止具合値aが所定の閾値を下回ったときのみ、ステップS49による補正を行ったりしてもよい。

【0146】

このように、この実施例では、取得した角速度データdataに対して設定された安定範囲（下限）d1および安定範囲（上限）d2で示される範囲（安定範囲）内に含まれるデータバッファが連続している度合い（静止具合値a）に応じた重み付けに基づいて、角速度データdataと平均値データとの加重平均が算出されるので、連続している度合い（静止具合値a）が大きいときには平均値データが重い重み付けで採用される（平均値の影響が大きくなる）ために、精度が高く、ぶれを抑えた滑らかな信頼度の高い角速度データdataが得られる。逆に、連続している度合い（静止具合値a）が小さいときには角速度データdataが重い重み付けで採用される（角速度データdataの影響が大きくなる）ために、たとえば急激な変動が生じている場合には、角速度データdataはその変動を迅速に反映するように補正されるので、変化の大きい動きを迅速に反映するような追従性の高い（レスポンスのよい）角速度データdataが得られる。

【0147】

つまり、このような処理によって、リモコン32すなわちジャイロセンサユニット34の動きの変化が少ない（安定している）時には精度が高く、ぶれを抑えた滑らかな信頼度の高いデジタルデータ（出力データ）が得られる反面、反応を鈍化させる平滑処理を行いながらも、機敏な反応が求められるときには、変化の大きい動きを迅速に反映するような追従性の高い（レスポンスのよい）デジタルデータ（出力データ）が得られ、二律背反的な要求に効果的に応えることができる。

【0148】

さらに、前述の静止具合値aが大きければ大きいほど（リモコン32すなわちジャイロセンサユニット34の動きの変化が少ない（安定している）期間が長ければ長いほど）、ジャイロセンサ56から出力される角速度におけるゼロ点のオフセットとして用いている修正オフセット値ofsを、ステップS49で補正された角速度データdataに近づける補正も行うことで、温度ドリフトによりゼロ点のずれを軽減することができる。

【0149】

まず、ステップS51において、修正オフセット値ofsを補正するかどうかを判定する。具体的には、図13のステップS11において、ゲーム処理中に、リモコン32すなわちジャイロセンサユニット34を等速で移動させる操作を行う可能性がある場面かどうかを判断した結果に基づき、図13のステップS13または図13のステップS14でセット（リセット）されたオフセット補正フラグcfgに基づいて判定する。オフセット補正フラグcfgが1、すなわち、図13のステップS11で“NO”に分岐し、ステップS14でオフセット補正フラグcfgを1に設定した場合であり、リモコン32すなわちジャイロセンサユニット34を等速で移動させる操作を行う可能性がない場合には、ステップS53で修正オフセット値ofsの補正を行う。また、オフセット補正フラグcfgが0、すなわち、図13のステップS11で“YES”に分岐し、ステップS13でオフセット補正フラグcfgを0に設定した場合であり、リモコン32すなわちジャイロセンサユニット34を等速で移動させる操作を行う可能性がある場合には、修正オフセット値ofsの補正は行わずに、ステップS55に進む。

【0150】

ステップS53では、静止具合値aが大きければ大きいほど（リモコン32すなわちジャイロセンサユニット34の動きの変化が少ない（安定している）期間が長ければ長いほど

ど)、リモコン32すなわちジャイロセンサユニット34の動きの変化が少ない(安定している)といえるので、静止具合値aの大きさに応じて、ステップS49で補正された角速度データdataが、コントローラの静止状態における角速度データ(ゼロ点)であるように補正を行う度合いを算出して、その度合いに応じたオフセット値(修正オフセット値ofs)を補正する。

【0151】

具体的には、静止具合値aに定数C2を乗算して静止具合値aを変更する。定数C2は、たとえば「0.01」として設定され、静止具合値aは $0.01 \times a$ に変更される。定数C2を乗じることにより、静止具合値aを小さい値に変更している理由は、ステップS47で算出された静止具合値aをそのまま利用すると、短時間で角速度データdataが、コントローラの静止状態における角速度データ(ゼロ点)であるように補正されてしまい、角速度データdataの値が出にくくなってしまうためである。また、ゼロ点のずれの原因となる温度ドリフトという現象は変化の激しい現象ではなく、この補正は反応良く行う必要がないためでもある。他の例では、定数C2は変数として、角速度データdataや連続個数ct等に応じて変化させても良いし、その他の状況に応じて変化させても良い。そして、その変更された静止具合値aを使って、修正オフセット値ofsを補正する。つまり、修正オフセット値ofsに対して、ステップS49で補正された角速度データdataから修正オフセット値ofsを減算した結果に、定数C2を乗算して変更した静止具合値aを掛けた値を加算する。したがって、静止具合値aに応じて、ステップS49で補正された角速度データdataと修正オフセット値ofsとの差分を、所定の割合(定数C2を乗算して変更した静止具合値a)に応じて、修正オフセット値ofsに加算される。その結果、静止具合値(静止の程度)aが大きければ大きいほど(リモコン32すなわちジャイロセンサユニット34の動きの変化が少ない(安定している)期間が長ければ長いほど)、修正オフセット値ofsがステップS49で補正された角速度データdataに近い値に補正される、換言すれば、修正オフセット値ofsがステップS49で補正された角速度データdataに収束する割合が大きくなる(度合いが強くなる)。また他の例では、所定の条件を満たす時(例えば、角速度データdataから修正オフセット値ofsを減算した結果が所定値以下の場合)にのみ、修正オフセット値ofsを補正しても良い。

【0152】

その後、ステップS55では、ステップS49で補正された角速度データdataから修正オフセット値ofsを減じることにより、ジャイロセンサ56から出力される角速度におけるゼロ点の補正を考慮した上で、角速度データdataが再度補正され、図13のステップS9にリターンする。なお、図13のステップS11で“YES”に分岐し、ステップS13でオフセット補正フラグcfgを0に設定した場合であり、リモコン32すなわちジャイロセンサユニット34を等速で移動させる操作を行う可能性がある場合には、ステップS53で補正が行われなままの修正オフセット値ofsをそのまま用いて、角速度データdataの補正が行われる。

【0153】

上述のように、静止具合値aの大きさ(リモコン32すなわちジャイロセンサユニット34の動きの変化が少ない(安定している)期間の長さ)により実施例のようなリモコン32すなわちジャイロセンサユニット34の安定状態の判定を行うと、リモコン32すなわちジャイロセンサユニット34を等速で移動させ続けた場合にも、静止具合値(静止の程度)aが大きくなる(リモコン32すなわちジャイロセンサユニット34の動きの変化が少ない(安定している)期間が長い)と判定されるという現象が起こる。ステップS49における角速度データdataの補正に関しては、等速で移動させ続けた場合にコントローラ14すなわちリモコン32から出力される角速度データは同じ(または近い)データが出力され続け、変化の大きい動きを迅速に反映するような追従性は求められない為に、平均値データが重い重み付けで採用される(平均値の影響が大きくなる)ことには問題はないが、ステップS53における修正オフセット値ofsの補正に関しては、修正オフセット値ofsがステップS49で補正された角速度データdataに近い値に補正される割合が大きくなる(度

10

20

30

40

50

合いが強く) になってしまい、リモコン 3 2 すなわちジャイロセンサユニット 3 4 の移動している状態を、コントローラの静止状態における角速度データ(ゼロ点)であるように補正されてしまうという問題が生じる。ただし、実施例のようなゲームシステムでは、リモコン 3 2 すなわちジャイロセンサ 5 6 を人が手に持って操作することを前提にしているため、角速度におけるゼロ点のオフセット修正が起こるほど、機械のように正確にリモコン 3 2 すなわちジャイロセンサユニット 3 4 を等速で移動させ続けられないということで、実用においては問題になることはない、ということもできる。

【 0 1 5 4 】

それでももし、そのような都合の悪い状況が起こりやすい操作を必要とする場合には、修正オフセット値ofsの補正を行うタイミングを工夫する必要がある。たとえば、リモコン 3 2 すなわちジャイロセンサユニット 3 4 でゲーム画面上の弓を動かして何かを慎重に狙う場合、リモコン 3 2 すなわちジャイロセンサユニット 3 4 が静かに等速で動かされる可能性がある。したがって、この実施例においては、たとえば矢を番える前まではゼロ点オフセットの修正処理機能をONにしておき、矢を番えて慎重に狙いだしたらOFFにするようにしている。それらの判定基準として用いられているのが、前述のオフセット補正フラグcfgである。

【 0 1 5 5 】

なお、ゼロ点のオフセット値は、温度センサを別途準備すれば、正確に温度ドリフトを補償できるように修正することができるが、それにはコストがかかる。これに対して、静止具合値(静止の程度)aに応じてゼロ点のオフセット値(修正オフセット値ofs)を補正しているため、簡易的ではあるが、温度センサを用いなくても温度ドリフトの影響を軽減するようにゼロ点のオフセット値(修正オフセット値ofs)を補正できる。

【 0 1 5 6 】

なお、上述の実施例では、ジャイロセンサからの出力値をデジタルデータに変換した場合について説明した。しかしながら、この発明は、このようなA/D変換を用いてデジタルデータを取得する任意のセンサに適用できる。たとえば、加速度センサ、速度センサ、変位センサ、回転角センサなど、他のモーションセンサでもよい。モーションセンサ以外にも、傾斜センサ、イメージセンサ、光センサ、圧力センサ、磁気センサ、温度センサなどがあり、いずれのセンサの場合でも、センサからの検出アナログ値をデジタルデータに変換し、そのデジタルデータを使う任意の装置にこの発明を適用できる。ただし、修正オフセット値ofsの補正については必ずしもすべてのセンサが該当するという訳ではなく、たとえば加速度センサについてはゼロ点のオフセットという考え方はない。

【 0 1 5 7 】

ここで、上記の実施例において具体的数値を当てはめて、この実施例の効果を検証する。

【 0 1 5 8 】

数値の確認をし易くするために、ここでは次のようなパラメータを用いる。

【 0 1 5 9 】

$C1 = 1$ 、 $C2 = 1$ 、 $C_{MAX} = 10$ 、 $k = 1$

正確なゼロ値が安定して出力される例

まず、或る状態として、ジャイロセンサのゼロ点の値が「2.4」であり、プログラム中における修正オフセット値ofsも「2.4」である正常な状況を示す。この場合、リモコン 3 2 すなわちジャイロセンサユニット 3 4 を手に持って静止させていると(完全に固定していない)、入力されるデジタルデータは、統計的に平均値が「2.4」になるように、たまに変動して、たとえば次のようになる。

デジタルデータ: 2 2 2 3 3 2 2 2 3 3

この場合、最新のデジタルデータは「2」または「3」となるが、いずれの場合でも、過去に安定範囲(下限)d1および安定範囲(上限)d2で設定された範囲内の値が続く回数(連続個数ct)は最大検索数C_{MAX}の「10」と等しくなり、図14のステップS47において算出される静止具合値aは「1」になる。

【 0 1 6 0 】

その結果、平均値が 2 . 4 になるようなデジタルデータが出力されているという前提の上記のような状況下においては、ステップ S 4 9 で補正される角速度データ data は、sum/ct で算出されるので、最新値が「 2 」でも「 3 」でも「 2 . 4 」になる。

【 0 1 6 1 】

さらに、ステップ S 5 3 の修正オフセット値 ofs の計算結果も「 2 . 4 」のまま変化せず、ステップ S 5 5 の最終出力データとしては、以下のように、「 2 . 4 - 2 . 4 」 = 0 . 0 が続くことになる。

デジタルデータ : 2 2 2 3 3 2 2 2 3 3

アナログ出力 : 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0

10

ゼロ点が変わったときに補正される例

次に、上記状態からセンサのゼロ点が「 3 . 6 」に変化した場合、上記同様にリモコン 3 2 すなわちジャイロセンサユニット 3 4 を静止させていると、平均値が「 3 . 6 」となる次のような変動を生じる。

デジタルデータ : 2 2 2 3 3 (4) 4 4 3 3 4 4 3 3

上記デジタルデータの左から 6 番目の括弧付の「 4 」でゼロ点変化が生じたとすると、このときの各ステップ中の変数は表 1 のようになる。

【 0 1 6 2 】

【表 1】

data(S33)	2	2	2	3	3	4	4	4	3	3	4	4	4	3	3
ct(S47)	10	10	10	10	10	3	4	5	10	10	8	9	10	10	10
a(S47)	1.0	1.0	1.0	1.0	1.0	0.3	0.4	0.5	1.0	1.0	0.8	0.9	1.0	1.0	1.0
sum(S49)	24	24	24	24	24	10	14	18	30	30	28	32	36	36	36
data(S49)	2.4	2.4	2.4	2.4	2.4	3.8	3.8	3.8	3.0	3.0	3.6	3.6	3.6	3.6	3.6
ofs(S53)	2.4	2.4	2.4	2.4	2.4	2.8	3.2	3.5	3.0	3.0	3.5	3.6	3.6	3.6	3.6
data(S55)	0.0	0.0	0.0	0.0	0.0	1.0	0.6	0.3	0.0	0.0	0.1	0.0	0.0	0.0	0.0

20

30

【 0 1 6 3 】

つまり、次のような結果となり、ゼロ点が変わっても出力データはやがて「 0 . 0 」に落ち着くことになる。

デジタルデータ : 2 2 2 3 3 4 4 4 3 3 4 4 4 3 3

アナログ出力 : 0.0 0.0 0.0 0.0 0.0 0.0 (1.0) 0.6 0.3 0.0 0.0 0.1 0.0 0.0 0.0 0.0

したがって、左から 7 番目のデータからゼロ点の補正がかかり始め、左から 1 2 番目のデータで補正が完了する。

センサを動かしたときにすばやく出力データに反映される例

さらに、上記状態から正方向への回転を行った場合、たとえば次のような入力になる。

デジタルデータ : 4 4 4 3 3 4 (5) 6 7 8 10 12 16 20 25

40

上記デジタルデータの左から 7 番目の括弧付きの「 5 」で回転を始めたとすると、このときの各ステップ中の変数は表 2 のようになる。

【 0 1 6 4 】

【表 2】

data(S33)	4	4	4	3	3	4	5	6	7	8	10	12	16	20	25
ct(S47)	8	9	10	10	10	10	2	2	2	2	1	1	1	1	1
a(S47)	0.8	0.9	1.0	1.0	1.0	1.0	0.2	0.2	0.2	0.2	0.1	0.1	0.1	0.1	0.1
sum(S49)	28	32	36	36	36	36	9	11	13	15	10	12	16	20	25
data(S49)	3.6	3.6	3.6	3.6	3.6	3.6	4.9	5.9	6.9	7.9	10.0	12.0	16.0	20.0	25.0
ofs(S53)	3.5	3.6	3.6	3.6	3.6	3.6	3.9	4.3	4.8	5.4	5.9	6.5	7.4	8.7	10.3
data(S55)	0.1	0.0	0.0	0.0	0.0	0.0	1.0	1.6	2.1	2.5	4.1	5.5	8.6	11.3	14.7

10

【0165】

つまり、次のような結果となり、入力に追従する結果が得られる。

デジタル入力：4 4 4 3 3 4 5 6 7 8 10 12 16 20 25

アナログ出力：0.1 0.0 0.0 0.0 0.0 0.0 0.0 (1.0) 1.6 2.1 2.5 4.1 5.5 8.6 11.3 14.7

したがって、左から7番目の括弧付の「1.0」で、回転が生じる。

【0166】

なお、この例では出力の変化に若干遅れを感じるが、それは、ここの例で用いているパラメータが、少ないデータサンプル数でもゼロ点補正が確認できるように極端な値にしたためである。ゼロ点の補正が即座にかかるような調整は、ジャイロセンサを動かしたときの追従性が鈍くなる。しかしながら、実際の場合は、ゼロ点の補正は数百というデータサンプルが行われる中で変化していく調整がされており、その分ジャイロセンサを動かしたときの追従性は、ここの例に現れたような遅れは生じない。

20

【0167】

上述の実施例では、リモコン32にジャイロセンサユニット34（ジャイロセンサ56）を接続するようにしたが、ジャイロセンサ56をリモコン32に内蔵するようにしてもよい。

【0168】

また、上述の実施例では、所定のセンサ（上記例ではジャイロセンサ）から出力されるデジタルデータ（角速度データ）を、同一のセンサ（ジャイロセンサ）から出力されるデジタルデータ（角速度データ）を用いて静止具合等を算出した上で補正を行っていたが、異なるセンサから出力されるデジタルデータを用いて静止具合等を設定した上で補正を行っても良い。例えば、加速度センサから出力されるデジタルデータ（加速度データ）から静止具合等を算出した上で、ジャイロセンサから出力されるデジタルデータ（角速度データ）の補正を行ってもよい。

30

【図面の簡単な説明】

【0169】

【図1】図1はこの発明の一実施例の構成を示すブロック図である。

【図2】図2は図1実施例に適用されるコントローラ（リモコン）の外観を示す図解図であり、図2（A）はリモコンを上面後方から見た斜視図であり、図2（B）はリモコンを下面前方から見た斜視図である。

40

【図3】図3は図1実施例に適用されるジャイロセンサユニットの外観を示す図解図であり、図3（A）はジャイロセンサユニットを上面前方から見た斜視図であり、図3（B）はジャイロセンサユニットを下面後方から見た斜視図である。

【図4】図4は図1実施例の電氣的な構成を示すブロック図である。

【図5】図5は図1実施例に適用されるコントローラ全体の電氣的な構成を示すブロック図である。

【図6】図6は図1に示すジャイロユニットが接続されたりモコンを用いてゲームプレイするときの状態を概説するための図解図である。

50

【図7】図7は図1のコントローラにおいて、リモコンに接続したジャイロセンサユニットの電気的な構成を示すブロック図である。

【図8】図8はジャイロセンサユニットが取り扱うデータのフォーマットを示す図解図である。

【図9】図9はジャイロセンサが検出できるヨー角、ピッチ角、ロール角を示す図解図である。

【図10】図10は図4に示すメインメモリのメモリマップを示す図解図である。

【図11】図11は図10に示すデータ記憶領域の具体例を示す図解図である。

【図12】図12はFIFOバッファの一例を示す図解図である。

【図13】図13は実施例において図4に示すCPUのゲーム処理を示すフロー図である

10

【図14】図14は図13に示す角速度データ取得処理の動作を示すフロー図である。

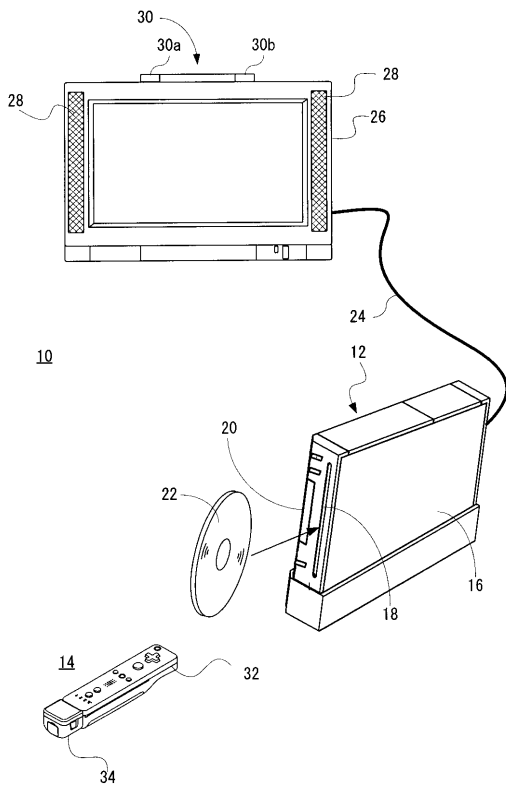
【符号の説明】

【0170】

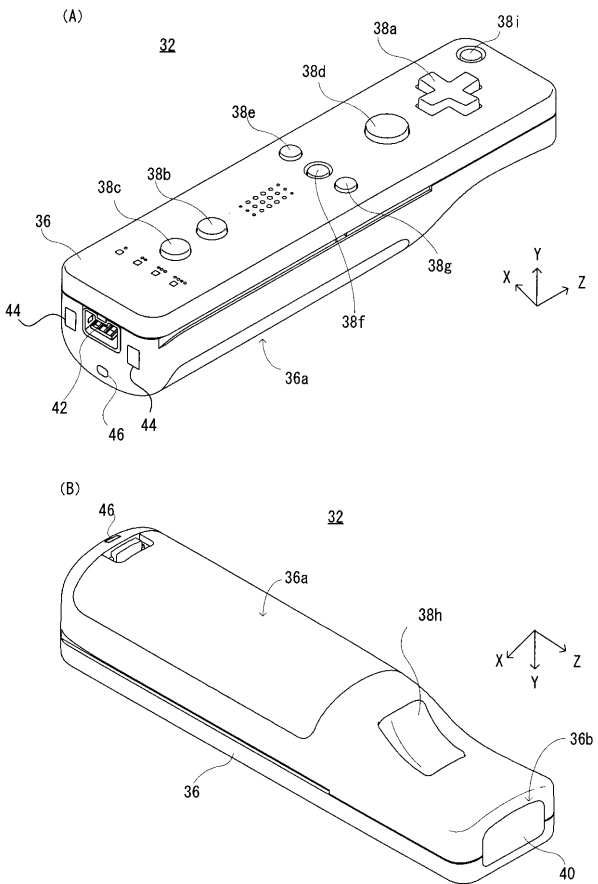
- 10 ...ゲームシステム
- 12 ...ゲーム装置
- 14 ...コントローラ
- 32 ...リモコン
- 34 ...ジャイロセンサユニット
- 56 ...ジャイロセンサ
- 60 ...CPU
- 62 ...システムLSI
- 100 ...FIFOバッファ

20

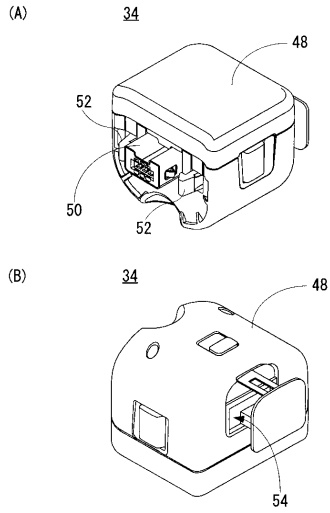
【図1】



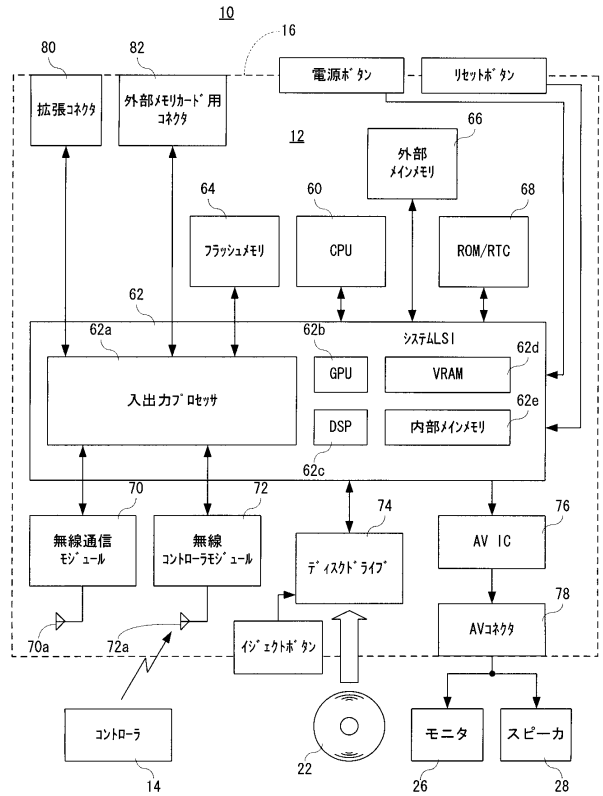
【図2】



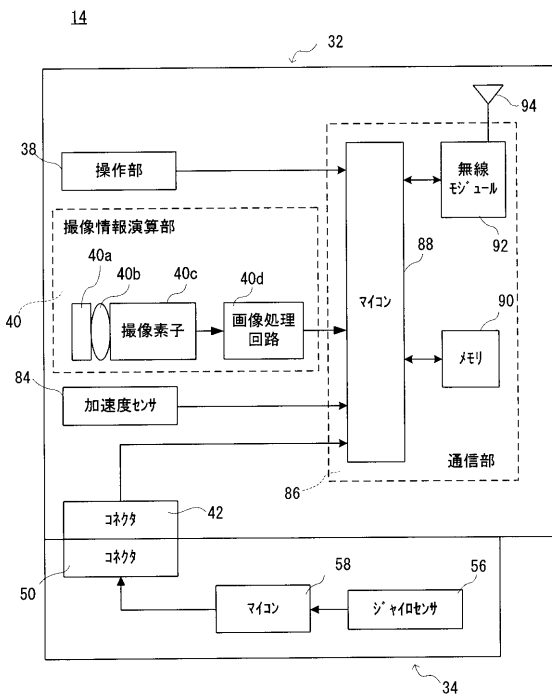
【図3】



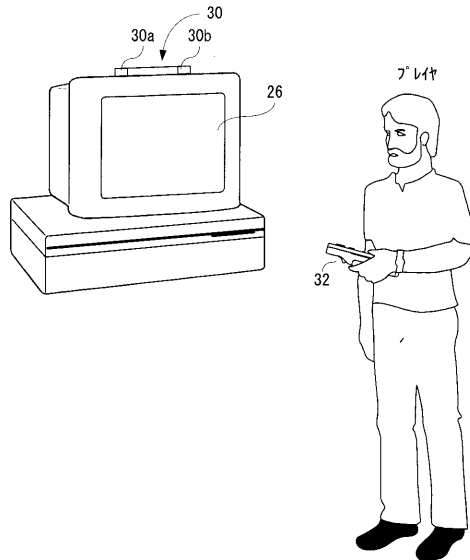
【図4】



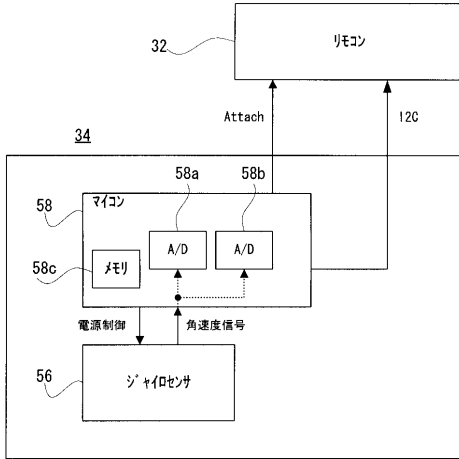
【図5】



【図6】



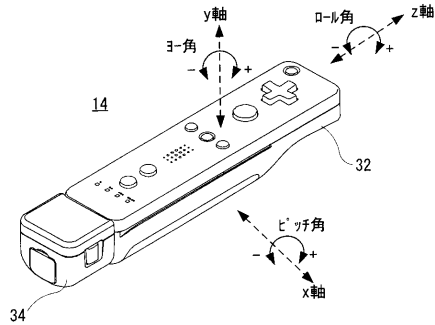
【図7】



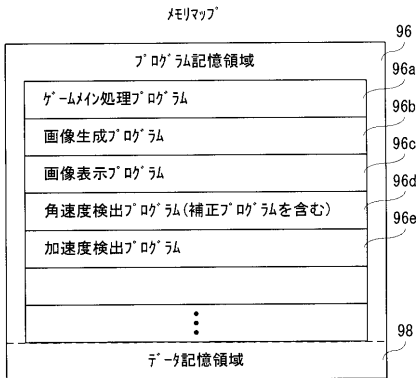
【図8】

ヨー角速度データ
ロール角速度データ
ピッチ角速度データ
ヨー角速度モード情報
ロール角速度モード情報
ピッチ角速度モード情報

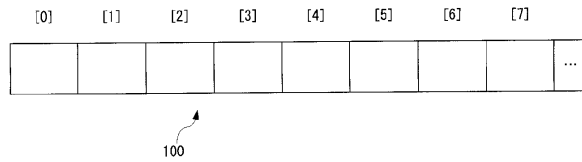
【図9】



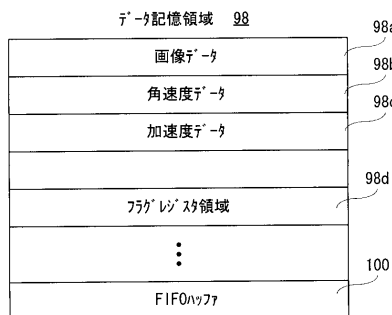
【図10】



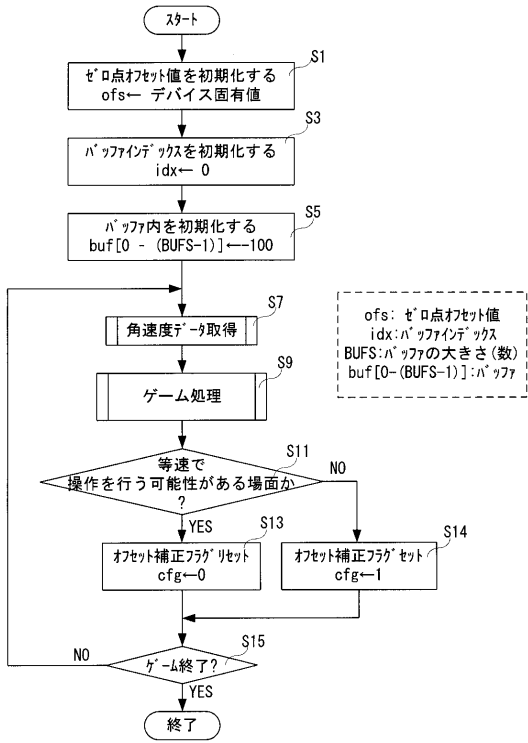
【図12】



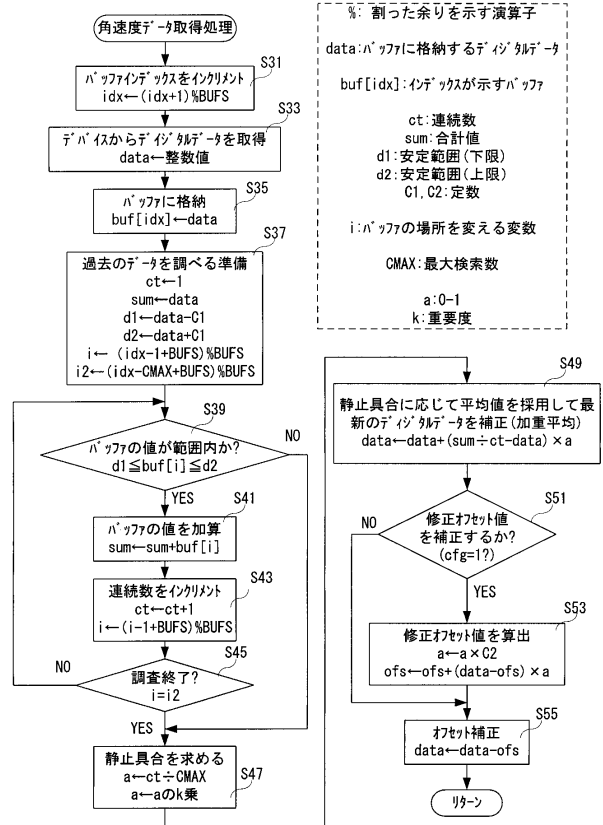
【図11】



【図13】



【図14】



フロントページの続き

(56)参考文献 特開2002-023919(JP,A)
特開昭59-094017(JP,A)
特開平10-097375(JP,A)

(58)調査した分野(Int.Cl., DB名)

G01D	1/02	
G01D	3/00	- 028
G01P	21/02	
G01C	19/00	
A63F	13/00	
A63F	13/06	
G06F	3/01	
G06F	3/048	