

(19) 日本国特許庁(JP)

(12) 特許公報(B2)

(11) 特許番号

特許第3655438号  
(P3655438)

(45) 発行日 平成17年6月2日(2005.6.2)

(24) 登録日 平成17年3月11日(2005.3.11)

(51) Int. Cl.<sup>7</sup>

F I

A 6 3 F 13/00

A 6 3 F 13/00

A

A 6 3 F 13/06

A 6 3 F 13/06

請求項の数 2 (全 23 頁)

<p>(21) 出願番号 特願平9-192166                  (22) 出願日 平成9年7月17日(1997.7.17)                  (65) 公開番号 特開平11-33229                  (43) 公開日 平成11年2月9日(1999.2.9)                  審査請求日 平成15年5月23日(2003.5.23)</p>	<p>(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11番地1                  (74) 代理人 100090181                  弁理士 山田 義人                  (72) 発明者 西海 聡                  京都府京都市東山区福稲上高松町60番地                  任天堂株式会社内                  (72) 発明者 幸嶋 一雄                  京都府京都市東山区福稲上高松町60番地                  任天堂株式会社内                  (72) 発明者 太田 敬三                  京都府京都市東山区福稲上高松町60番地                  任天堂株式会社内</p>
---	---

最終頁に続く

(54) 【発明の名称】 ビデオゲームシステム

(57) 【特許請求の範囲】

【請求項1】

少なくともビデオゲーム機と記憶媒体とコントローラとを備えたゲームシステムであって、

前記ビデオゲーム機は前記コントローラから出力される操作データと前記記憶媒体に記憶されたゲームプログラムとに基づいて、所定周期毎にディスプレイに表示されるべきゲーム画像を発生する処理手段と、カウント値を保持するためのメモリとを含み、

前記コントローラは、

プレイヤーの手によって把持するためのハウジングと、

前記ハウジングに振動を与える振動源と、

前記ビデオゲーム機からの指令信号にตอบสนองして前記振動源を駆動する駆動回路とを備え、

前記記憶媒体は、前記処理手段に実行させるプログラムであって、前記ビデオゲーム機に、

前記コントローラから出力される前記操作データに応じてプレイヤーオブジェクトの表示を制御する、プレイヤーオブジェクト表示制御プログラムと、

前記プレイヤーオブジェクト以外のオブジェクトの表示を制御する、オブジェクト表示制御プログラムと、

前記コントローラから所定の操作データが出力されたとき、および前記プレイヤーオブジェクトが前記プレイヤーオブジェクト以外のオブジェクトに接触したときの少なくとも一

方のとき前記所定周期毎に増分値を前記カウント値に加算し、前記カウント値が所定値を超えたとき、予め定められた減算値をカウント値から減算する、カウント値制御プログラムと、

前記カウント値が前記所定値を超えた前記所定周期においてのみ前記振動源を駆動する、振動源制御プログラムとを含み、

前記ビデオゲーム機は、前記処理手段が前記プログラムを実行することに基づいてプレイヤーにゲームをプレイさせる、ビデオゲームシステム。

【請求項 2】

少なくともビデオゲーム機とプレイヤーが操作するコントローラとを備えたゲームシステムであって、

前記コントローラは、

プレイヤーの手によって把持するためのハウジングと、

前記ハウジングに振動を与える振動源と、

前記ビデオゲーム機からの指令信号に応答して前記振動源を駆動する駆動回路とを備え、

前記ビデオゲーム機は、

カウント値を保持するためのメモリと、

前記コントローラから出力される操作データに応じてプレイヤーオブジェクトの表示を制御する、プレイヤーオブジェクト表示制御手段と、

前記プレイヤーオブジェクト以外のオブジェクトの表示を制御する、オブジェクト表示制御手段と、

前記コントローラから所定の操作データが出力されたとき、および前記プレイヤーオブジェクトが前記プレイヤーオブジェクト以外のオブジェクトに接触したときの少なくとも一方のとき前記所定周期毎に増分値を前記カウント値に加算し、前記カウント値が所定値を超えたとき、予め定められた減算値をカウント値から減算する、カウント値制御手段と、

前記カウント値が前記所定値を超えた前記所定周期においてのみ前記振動源を駆動する、振動源制御手段とを含む、ビデオゲームシステム。

【発明の詳細な説明】

【0001】

【産業上の利用分野】

この発明はビデオゲームシステムに関する。より特定的には、この発明は、ビデオゲーム機に操作信号を与えるコントローラに設けられた振動源によってゲーム画像に応じて振動を発生させる、新規なビデオゲームシステムに関する。

【0002】

【従来の技術】

従来のいわゆる業務用ビデオゲーム機では、プレイヤーの座るシートに振動を発生させたり、あるいはそのシートを傾けたりして、ゲームの興趣を一層高めることができる、いわゆる体感ゲーム機が既に実用化されている。

また、たとえば昭和62年(1987)12月10日付で出願公開された実開昭62-194389号には、ジョイスティックレバー内に電磁石を内蔵し、その電磁石によってジョイスティックレバーに振動を発生させるコントローラが開示されている。このような振動発生可能なコントローラを家庭用ビデオゲーム機に接続することによって、プレイヤーの手に振動が伝達されるゲームを構成することが考えられる。

【0003】

【発明が解決しようとする課題】

しかしながら、上述の従来技術は、ディスプレイに表示されるゲーム画像と振動との関連が全く開示されておらず、したがって、この従来技術を家庭用ビデオゲームシステムに利用することはできない。また、上述の体感ゲーム機は、シートに振動を発生させるものであるため、そのようなシートを持たない家庭用ビデオゲームシステムでは利用できない。

【0004】

10

20

30

40

50

それゆえに、この発明の主たる目的は、ゲーム画像と関連してプレイヤに振動を伝達することができる、ビデオゲームシステムを提供することである。

【0005】

【課題を解決するための手段】

この発明は、少なくともビデオゲーム機と記憶媒体とコントローラとを備えたゲームシステムであって、ビデオゲーム機はコントローラから出力される操作データと記憶媒体に記憶されたゲームプログラムとに基づいて、所定周期毎にディスプレイに表示されるべきゲーム画像を発生する処理手段と、カウント値を保持するためのメモリとを含み、コントローラは、プレイヤの手によって把持するためのハウジングと、ハウジングに振動を与える振動源と、ビデオゲーム機からの指令信号に应答して振動源を駆動する駆動回路とを備え、記憶媒体は、処理手段に実行させるプログラムであって、ビデオゲーム機に、コントローラから出力される操作データに応じてプレイヤオブジェクトの表示を制御する、プレイヤオブジェクト表示制御プログラムと、プレイヤオブジェクト以外のオブジェクトの表示を制御する、オブジェクト表示制御プログラムと、コントローラから所定の操作データが出力されたとき、およびプレイヤオブジェクトがプレイヤオブジェクト以外のオブジェクトに接触したときの少なくとも一方のとき所定周期毎に増分値をカウント値に加算し、カウント値が所定値を超えたとき、予め定められた減算値をカウント値から減算する、カウント値制御プログラムと、カウント値が所定値を超えた所定周期においてのみ振動源を駆動する、振動源制御プログラムとを含み、ビデオゲーム機は、処理手段がプログラムを実行することに基づいてプレイヤにゲームをプレイさせる、ビデオゲームシステムである。

10

20

【0006】

この発明は、少なくともビデオゲーム機とプレイヤが操作するコントローラとを備えたゲームシステムであって、コントローラは、プレイヤの手によって把持するためのハウジングと、ハウジングに振動を与える振動源と、ビデオゲーム機からの指令信号に应答して振動源を駆動する駆動回路とを備え、ビデオゲーム機は、カウント値を保持するためのメモリと、コントローラから出力される操作データに応じてプレイヤオブジェクトの表示を制御する、プレイヤオブジェクト表示制御手段と、プレイヤオブジェクト以外のオブジェクトの表示を制御する、オブジェクト表示制御手段と、コントローラから所定の操作データが出力されたとき、およびプレイヤオブジェクトがプレイヤオブジェクト以外のオブジェクトに接触したときの少なくとも一方のとき所定周期毎に増分値をカウント値に加算し、カウント値が所定値を超えたとき、予め定められた減算値をカウント値から減算する、カウント値制御手段と、カウント値が所定値を超えた所定周期においてのみ振動源を駆動する、振動源制御手段とを含む、ビデオゲームシステムである。

30

【0007】

【作用】

プレイヤオブジェクト画像および他のオブジェクト画像の状態（たとえば、両者が当たっているかなど）に応じて、振動発生条件が検出される。この条件が検出されると、コントローラに含まれる振動源を駆動回路によって駆動する。

また、コントローラの操作スイッチの操作状態によって、プレイヤオブジェクト画像と他のオブジェクト画像との関係によらず、振動を発生させることもできる。

40

【0008】

【発明の効果】

この発明によれば、ゲーム画像の状態に応じてコントローラの振動源によって振動を発生させることができるので、より一層実感を伴ってゲームをプレイすることができる。

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。

【0009】

【実施例】

図1に示す実施例のビデオゲームシステムは、ビデオゲーム機10と、記憶媒体の一例の

50

R O Mカートリッジ 2 0 と、ビデオゲーム機 1 0 に接続されるディスプレイ 3 0 と、コントローラ 4 0 とを含んで構成される。コントローラ 4 0 には、振動カートリッジ 5 0 が着脱自在に装着される。

【 0 0 1 0 】

コントローラ 4 0 は、両手または片手で把持可能な形状のハウジング 4 1 に、複数のスイッチないしボタンを設けて構成される。具体的には、コントローラ 4 0 は、ハウジング 4 1 の左右端部および中央部に、それぞれ下方に延びて形成されるハンドル 4 1 L , 4 1 C および 4 1 R を含み、ハウジング 4 1 の上面が操作領域である。操作領域には、中央下部にアナログ入力可能なジョイスティック（以下、「アナログジョイスティック」という。） 4 5 が設けられ、左側に十字形のデジタル方向スイッチ（以下、「十字スイッチ」という。） 4 6 が設けられ、右側に複数のボタンスイッチ 4 7 A , 4 7 B , 4 7 C , 4 7 D , 4 7 E および 4 7 F が設けられる。

10

【 0 0 1 1 】

アナログジョイスティック 4 5 は、スティックの傾き量と方向とによって、プレイヤオブジェクト（プレイヤがコントローラ 4 0 によって操作可能なオブジェクト）の移動方向および/または移動速度ないし移動量を入力するために用いられる。十字スイッチ 4 6 は、アナログジョイスティック 4 5 に代えてプレイヤオブジェクトの移動方向を指示するために用いられる。ボタンスイッチ 4 7 A および 4 7 B は、プレイヤオブジェクトの動作を指示するために利用され、ボタンスイッチ 4 7 C - 4 7 D は、三次元画像のカメラの視点を切り換えたり、プレイヤオブジェクトのスピード調節等に用いられる。

20

【 0 0 1 2 】

操作領域のほぼ中央部にはスタートスイッチ 4 7 S が設けられ、このスタートスイッチ 4 7 S は、ゲームを開始させるときに操作される。中央部のハンドル 4 1 C の裏側にスイッチ 4 7 Z が設けられ、このスイッチ 4 7 Z は、たとえばシューティングゲームにおいてトリガスイッチとして利用される。ハウジング 4 1 の左右上部側面にはスイッチ 4 7 L および 4 7 R が設けられる。

【 0 0 1 3 】

なお、上述のボタンスイッチ 4 7 C - 4 7 F は、カメラの視点切換え以外の用途として、シューティングまたはアクションゲームにおいてプレイヤオブジェクトの移動速度を制御（たとえば、加速または減速）するためにも使用できる。しかしながら、これらのスイッチ 4 7 A - 4 7 F , 4 7 S , 4 7 Z , 4 7 L および 4 7 R の機能は、ゲームプログラムによって任意に定義することができる。

30

【 0 0 1 4 】

図 2 は図 1 実施例のビデオゲームシステムのブロック図である。ビデオゲーム機 1 0 には、中央処理ユニット（以下、「CPU」という。） 1 1 およびコプロセッサ（リアリティ・コプロセッサ；以下、「RCP」という。） 1 2 が内蔵される。RCP 1 2 には、バスの制御を行うためのバス制御回路 1 2 1 と、ポリゴンの座標変換や陰影処理等を行うための信号プロセッサ（リアリティ・シグナル・プロセッサ；以下、「RSP」という。） 1 2 2 と、ポリゴンデータを表示すべき画像にラスタライズしかつフレームメモリに記憶可能なデータ形式（ドットデータ）に変換するための描画プロセッサ（リアリティ・ディスプレイ・プロセッサ；以下、「RDP」という。） 4 6 とが含まれる。

40

【 0 0 1 5 】

RCP 1 2 には、外部 ROM 2 1 を内蔵する ROM カートリッジ 2 0 を着脱自在に装着するためのカートリッジ用コネクタ 1 3 と、ディスクドライブ 2 9 を着脱自在に装着するためのディスクドライブ用コネクタ 1 9 7 と、RAM 1 4 とが接続される。また、RCP 1 2 には、CPU 1 1 によって処理された音声信号および映像信号をそれぞれ出力するための DAC（デジタル/アナログ変換器） 1 5 および 1 6 が接続される。さらに、RCP 1 2 には、1 つまたは複数のコントローラ 4 0 の操作データおよび/または振動カートリッジ 5 0 のデータをシリアル転送するためのコントローラ制御回路 1 7 が接続される。

【 0 0 1 6 】

50

R C P 1 2 に含まれるバス制御回路 1 2 1 は、C P U 1 1 からバスを介してパラレル信号で与えられたコマンドをパラレル/シリアル変換して、シリアル信号としてコントローラ制御回路 1 7 に供給する。また、バス制御回路 1 2 1 は、コントローラ制御回路 1 7 から入力されたシリアル信号をパラレル信号に変換し、バスを介して C P U 1 1 へ出力する。コントローラ 4 0 から読み込まれた操作状態を示すデータ（操作信号ないし操作データ）は、C P U 1 1 によって処理されたり、R A M 1 4 に一時記憶される等の処理が行われる。換言すれば、R A M 1 4 は、C P U 1 1 によって処理されるデータを一時記憶する記憶領域を含み、バス制御回路 1 2 1 を介してデータの読出または書込を円滑に行うことに利用される。

【 0 0 1 7 】

音声用 D A C 1 5 には、ビデオゲーム機 1 0 の後面に設けられるコネクタ 1 9 5 が接続される。画像用 D A C 1 6 には、ビデオゲーム機 1 0 の後面に設けられるコネクタ 1 9 6 が接続される。コネクタ 1 9 5 には、ディスプレイ 3 0 のスピーカ 3 1 が接続される。コネクタ 1 9 6 には、テレビジョン受像機または C R T 等のディスプレイ 3 0 が接続される。

【 0 0 1 8 】

コントローラ制御回路 1 7 には、ビデオゲーム機 1 0 の前面に設けられるコントローラ用コネクタ 1 8 が接続される。コネクタ 1 8 には、接続用ジャックを介してコントローラ 4 0 が着脱自在に接続される。このように、コネクタ 1 8 にコントローラ 4 0 を接続することにより、コントローラ 4 0 がビデオゲーム機 1 0 と電気的に接続され、相互間のデータの送受信または転送が可能とされる。

【 0 0 1 9 】

コントローラ制御回路 1 7 は、R C P 1 2 とコントローラ用コネクタ 1 8 との間でデータをシリアルで送受信するために用いられ、図 3 に示すように、データ転送制御回路 1 7 1 , 送信回路 1 7 2 , 受信回路 1 7 3 および送受信データを一時記憶するための R A M 1 7 4 を含む。データ転送制御回路 1 7 1 は、データ転送時にデータフォーマットを変換するためにパラレル/シリアル変換回路とシリアル/パラレル変換回路を含み、さらに R A M 1 7 4 の書込/読出制御を行う。シリアル/パラレル変換回路は、R C P 1 2 から供給されるシリアルデータをパラレルデータに変換して R A M 1 7 4 または送信回路 1 7 2 に与える。パラレル/シリアル変換回路は、R A M 1 7 4 または受信回路 1 7 3 から供給されるパラレルデータをシリアルデータに変換して、R C P 1 2 に与える。送信回路 1 7 2 は、データ転送制御回路 1 7 1 から供給されるコントローラ 4 0 の信号読込のためのコマンドおよび振動カートリッジ 5 0 への書込データ（パラレルデータ）をシリアルデータに変換して、各コントローラ 4 0 のそれぞれに対応するチャンネル C H 1 ~ C H 4 へ送出する。受信回路 1 7 3 は、各コントローラ 4 0 に対応するチャンネル C H 1 ~ C H 4 から入力される各コントローラの操作データおよび振動カートリッジ 5 0 からの読出データをシリアルデータで受信し、パラレルデータに変換してデータ転送制御回路 1 7 1 に与える。データ転送制御回路 1 7 1 は、R C P 1 2 から転送されたデータまたは受信回路 1 7 3 で受信されたコントローラデータや振動カートリッジ 5 0 の読出データを R A M 1 7 4 に書込んだり、R C P 1 2 からの命令に基づいて R A M 1 7 4 のデータを読み出して R C P 1 2 へ転送する。

【 0 0 2 0 】

なお、R A M 1 7 4 は、図示を省略しているが、各チャンネル C H 1 ~ C H 4 毎の記憶場所を有し、各記憶場所に当該チャンネルのコマンド、送信データおよび/または受信データがそれぞれ記憶される。

図 4 はコントローラ 4 0 および振動カートリッジ 5 0 の詳細な回路図である。コントローラ 4 0 のハウジングには、ジョイスティック 4 5 , 各スイッチ 4 6 , 4 7 等の操作状態を検出しかつその検出データをコントローラ制御回路 1 7 へ転送するために、操作信号処理回路 4 4 等が内蔵される。操作信号処理回路 4 4 は、受信回路 4 4 1 , 制御回路 4 4 2 , スイッチ信号検出回路 4 4 3 , カウンタ回路 4 4 4 , ジョイポート制御回路 4 4 6 , リセット回路 4 4 7 および N O R ゲート 4 4 8 を含む。受信回路 4 4 1 は、コントローラ制御

10

20

30

40

50

回路17から送信される制御信号や振動カートリッジ50への書込データ等のシリアル信号をパラレル信号に変換して制御回路442に与える。制御回路442は、コントローラ制御回路17から送信される制御信号がジョイスティック45のX、Y座標のリセット信号であるとき、リセット信号を発生してNORゲート448を介してカウンタ444内のX軸用カウンタ444XとY軸用カウンタ444Yの計数値をリセット(0)させる。

#### 【0021】

ジョイスティック45は、レバーの傾き方向のX軸方向とY軸方向に分解して傾き量に比例したパルス数を発生するように、X軸用とY軸用のフォトインタラプトを含み、それぞれのパルス信号をカウンタ444Xおよびカウンタ444Yに与える。カウンタ444Xは、ジョイスティック45がX軸方向に傾けられたとき、その傾き量に応じて発生されるパルス数を計数する。カウンタ444Yは、ジョイスティック45がY軸方向に傾けられたとき、その傾き量に応じて発生されるパルス数を計数する。したがって、カウンタ444Xとカウンタ444Yとの計数値によって決まるX軸とY軸の合成ベクトルによって、プレイヤーオブジェクトまたは主人公キャラクタもしくはカーソルの移動方向と座標位置とが決定される。なお、カウンタ444Xおよびカウンタ444Yは、電源投入時にリセット信号発生回路447から与えられるリセット信号、またはプレイヤーが所定の2つのスイッチを同時に押したときにスイッチ信号検出回路443から与えられるリセット信号によって、リセットされる。

10

#### 【0022】

スイッチ信号検出回路443は、制御回路442から一定周期(たとえばテレビジョンのフレーム周期である1/30秒間隔)で与えられるスイッチ状態を出力するためのコマンドに应答して、十字スイッチ46およびスイッチ47A~47Zの押圧状態によって変化する信号を読み込み、それを制御回路442へ与える。制御回路442は、コントローラ制御回路17からの操作状態データの読出指令信号に应答して、各スイッチ47A~47Zの操作状態データおよびカウンタ444Xおよび444Yの計数値を所定のデータフォーマットで送信回路445に与える。送信回路445は、制御回路442から出力されたパラレル信号をシリアル信号に変換して、変換回路43および信号線42を介してコントローラ制御回路17へ転送する。制御回路442には、アドレスバスおよびデータバスならびにポートコネクタ46を介してジョイポート制御回路446が接続される。ジョイポート制御回路446は、振動カートリッジ50がポートコネクタ46に接続されているとき、CPU11の命令に従ってデータの入出力(または送受信)制御を行う。

20

30

#### 【0023】

振動カートリッジ50は、アドレスバスおよびデータバスにRAM51を接続し、RAM51に電池52を接続して構成される。RAM51は、たとえばアドレスバスを用いてアクセス可能な最大容量の半分未満の容量(たとえば256kビット)のRAMである。RAM51は、ゲームに関連するバックアップデータを記憶するものであり、振動カートリッジ50がポートコネクタ46から抜き取られても電池52からの電力供給を受けてバックアップデータを保持する。この振動カートリッジ50は、振動発生回路53を内蔵する。

#### 【0024】

次に、図5を参照して、振動カートリッジ50を詳細に説明する。振動カートリッジ50は、ケース501とそのケース501に取り付けられる裏蓋502とを含む。このケース501および裏蓋502で形成された振動カートリッジ50が、図1に示すコントローラ40の開口部に、着脱自在に装着される。

40

ケース501内には、基板503が収納される。基板503上には、前述のRAM51およびバックアップ電池52の他に、図4の振動発生回路53を構成する、電池504および駆動回路505が実装される。なお、基板503の手前側エッジには、上述のコントローラ40の開口部に形成されたコネクタ(図示せず)に接続される複数の端子506が形成される。この端子506がビデオゲーム機10のCPU11(図2)すなわちコントローラ制御回路17からのデータおよびアドレスを受ける。

50

## 【0025】

裏蓋502には、振動発生回路53を構成する振動源507が固着される。実施例では、振動源507として、振動発生用モータを用いた。しかしながら、モータの他に、電源の供給を受けて振動を発生するソレノイドや他の素子が利用可能であることは勿論である。なお、振動発生用モータの一例として、東京パーツ工業株式会社製の「FM16」,「FM23」,「FM25」あるいは「FM29」や「CM-5」等が利用可能である。「FM」モータでは、円筒型ケースに内蔵された回転軸に偏心部材を取り付け、回転軸の回転に応じて偏心部材が回転することによって、ケースに振動が生じる。「CM」モータでは、電機子コイル自体を偏心させて装着し、その電機子が回転することによって、振動を発生する。なお、ソレノイドを用いる場合には、ソレノイド内の磁心が往復運動することによって振動が発生する。

10

## 【0026】

いずれの場合も、このような振動源507は、電池504からの電源を受けて駆動回路505によって駆動され、振動を発生する。振動源507の消費電力は比較的大きく、したがって、この実施例では、RAM51のバックアップ電池52(図4)とは別に、電池504を設けた。したがって、電池504が消耗されたときには、裏蓋502に取り外し可能に設けられた電池蓋508を開いて、電池504を新しい電池に交換することができる。ただし、2つの電池52および504として、同じ電池を共通に用いることも、可能である。

## 【0027】

また、コントローラケーブル(図示せず)に電源線を含ませ、その電源線によって、画像処理装置本体すなわちビデオゲーム機10から端子506を通して振動源507に電源を供給するようにしてもよい。その場合には、電源線の容量は、振動源507の必要電力を考慮して適宜選定されることはいうまでもない。

20

さらに、この実施例では、振動源507で発生した振動が減衰することなくコントローラ40からプレイヤーの手に伝達され易いように、振動源507を裏蓋502に取り付けた。つまり、振動源507で発生した振動は、裏蓋502から、その裏蓋502に接触しているコントローラ40の開口部に伝達され、したがって、コントローラ40自体が振動する。そのため、コントローラ40を把持しているプレイヤーの手に、振動源507が発生した振動が伝達されることになる。したがって、振動源507の振動がコントローラ40を通してプレイヤーの手に伝達され得る限り、振動源507をケース501内の任意の位置に取り付けることができる。

30

## 【0028】

なお、この実施例では、コントローラ40に振動カートリッジ50を装着することによってコントローラ40に振動源507を持たせた。しかしながら、振動カートリッジ50を用いることはなく、コントローラ40のハウジング内に、振動発生回路53(図4)すなわち振動源507,駆動回路505および電池504等を内蔵するようにしてもよい。

## 【0029】

次に、図6を参照して、振動発生回路53を構成する駆動回路505を詳細に説明する。駆動回路505は、NANDゲート510から成るデコーダを含み、このNANDゲート510は、ビデオゲーム機10のCPU11(図2)からのアドレスデータA2~A14を、アドレスバスすなわち端子506(図5)を通して受ける。実施例のゲーム機システムにおいては、そのアドレスA0~A15の全てが「1」の場合、すなわち、アドレスFFFの範囲をCPU11が指定したとき、振動モードが設定され、振動源507を駆動するためのデータをCPU11から出力するようにしている。つまり、CPUアドレスFFFが指定されると、デコーダすなわちNANDゲート510の出力が「0」となる。このNANDゲート510の出力がNANDゲート511に与えられる。NANDゲート511にはさらに、CPU11からの書込信号-WEおよびチップイネーブル信号CEが与えられているので、NANDゲート511はNANDゲート510の出力および信号-WEおよびCEに应答して、ラッチ512にラッチ信号を与える。したがって、ラッチ5

40

50

12は、CPU11がFFFFアドレスを指定したとき、すなわち、振動モードにおいて、CPUデータD0を、データバスすなわち端子506を通して、ラッチする。このCPUデータD0は、振動源507で振動を発生すべきとき「1」として、振動を発生すべきでないとき「0」として出力される。ラッチ512の出力は抵抗513を介して、駆動トランジスタ514のベースに接続される。ラッチ512の出力が「1」のとき、トランジスタ514がオンし、「0」のときオフする。トランジスタ514がオンすると、電池504から振動源507（振動モータ）に駆動電流が流れ、駆動源507から振動が発生される。

#### 【0030】

図7はROMカートリッジ20（図1）に内蔵される外部ROM21のメモリ空間を示すメモリマップである。外部ROM21は、複数の記憶領域（以下、単に「領域」と呼ぶこともある）、たとえば図5に示すように、プログラム領域22、文字コード領域23、画像データ領域24およびサウンドメモリ領域25を含み、各種のプログラムを予め固定的に記憶している。

10

#### 【0031】

プログラム領域22は、ゲーム画像を処理するために必要なプログラムや、ゲーム内容に応じたゲームデータ等を記憶している。具体的には、プログラム領域22は、CPU11の動作プログラムを予め固定的に記憶するための記憶領域22a～22pを含む。メインプログラム領域22aには、後述の図9に示すゲーム等のメインルーチンの処理プログラムが記憶される。コントロールパッドデータ判断プログラム領域22bには、コントローラ40の操作データを処理するためのプログラムが記憶される。書込プログラム領域22cには、CPU11がRCP12に書込処理させるべきフレームメモリおよびZバッファへの書込プログラムが記憶される。たとえば、書込プログラム領域22cには、1つの背景画面で表示すべき複数の移動オブジェクトまたは背景オブジェクトのテクスチャデータに基づく画像データとして、色データをRAM14の画像データ領域201（図8）に書き込むプログラムが記憶される。移動プログラム領域22dには、CPU11がRCP12に作用して三次元空間中の移動物体の位置を変化させるための制御プログラムが記憶される。カメラ制御プログラム領域22eには、プレイヤーオブジェクトを含む移動オブジェクトや背景オブジェクトを三次元空間中のどの方向および/または位置で撮影させるかを制御するためのカメラ制御プログラムが記憶される。プレイヤーオブジェクトプログラム領域22fには、プレイヤーによって操作されるオブジェクトの表示制御のためのプログラムが記憶される。敵オブジェクトプログラム領域22gには、プレイヤーオブジェクトに対して攻撃を加える敵オブジェクトの表示制御のためのプログラムが記憶される。背景プログラム領域22hには、CPU11がRCP12に作用して、三次元の背景画を作成させるための背景作成プログラムが記憶される。

20

30

#### 【0032】

文字コード領域23は、複数種類の文字コードを記憶する領域であって、たとえばコードに対応した複数種類の文字のドットデータを記憶している。文字コード領域23に記憶されている文字コードデータは、ゲームの進行においてプレイヤーに説明文を表示するために利用される。

40

画像データ領域24は、背景オブジェクトおよび/または移動オブジェクトの各オブジェクト毎に複数のポリゴンの座標データおよびテクスチャデータ等の画像データをそれぞれ記憶するとともに、これらのオブジェクトを所定の位置に固定的に表示または移動表示させるための表示制御プログラムを記憶している。

#### 【0033】

サウンドメモリ領域25には、場面毎に対応して、その場面に適した上記メッセージを音声で出力するためのセリフや効果音やゲーム音楽等のサウンドデータが記憶される。

なお、記憶媒体ないし外部記憶装置としては、ROMカートリッジ20に代えてまたはROMカートリッジ20に加えて、CD-ROMや磁気ディスク等の各種記憶媒体を用いてもよい。その場合、CD-ROMや磁気ディスク等の光学式または磁気式等のディスク状

50



記憶媒体からゲームのための各種データ（プログラムデータおよび画像表示のためのデータを含む）を読み出しまたは必要に応じて書き込むために、ディスクドライブ 29（図 2）が設けられる。ディスクドライブ 29 は、外部 ROM 21 と同様のプログラムデータが磁氣的または光学的に記憶された磁気ディスクまたは光ディスクに記憶されたデータを読み出し、そのデータを RAM 14 に転送する。

#### 【0034】

図 8 は RAM 14 のメモリ空間を示すメモリマップである。RAM 15 は、画像データ領域 201 およびプログラム領域 202 を含む。画像データ領域 201 は、図示しないが 1 フレーム分の画像データを一時記憶するフレームメモリ領域と、フレームメモリ領域のドット毎の奥行データを記憶する Z バッファ領域とを含む。プログラムデータ領域 202 は、プログラムを一時的に記憶するための領域である。上述の ROM 21 の各領域（図 7）に設定されたプログラムデータが必要に応じてプログラムデータ領域 202 に一時的に記憶され、CPU 11 および RCP 12（図 2）は、RAM 14 のプログラム領域をアクセスすることによって、ゲームを進行させる。同じように、画像データ領域 201 も、ROM 21 に記憶されている画像データを必要に応じて一時的に記憶しておくための領域であり、CPU 11 または RCP 12 によって直接アクセスされ得る。つまり、画像データ領域 201 は、外部 ROM 21 に記憶されているゲーム画像表示のための静止オブジェクトおよび / または移動オブジェクトを構成する複数のポリゴンの座標データおよびテクスチャデータを記憶するものであって、画像処理動作に先立ってたとえば 1 コースまたはステージ分のデータが外部 ROM 21 から転送される。

#### 【0035】

コントローラデータ記憶領域 141 は、コントローラ 40 から読み込まれた操作状態を示す操作状態データを一時記憶する。

また、フラグ・レジスタ領域 142 は、CPU 11 がプログラムを実行中に、必要に応じて、フラグを設定し、あるいは変数または定数を記憶する。このフラグ・レジスタ領域 143 に設定されるフラグとしては、振動ゲームフラグ F1、当たりフラグ F2、前フレームフラグ F3 および振動フラグ F4 などがある。

#### 【0036】

振動ゲームフラグ F1 は、現在プレイしているゲームが振動カートリッジ 50 の振動源 507 によって振動を発生させる場面を含むかどうかを示すフラグであり、そのような場面があるとき「1」として、それ以外るとき「0」として設定される。当たりフラグ F2 には、後述する当たり判定ルーチン（図 11）によって 2 つのオブジェクトが衝突ないし接触したとき「1」が設定され、それ以外るとき「0」が設定される。前フレーム F3 は、ディスプレイ 30 の 1 フレーム前のフレームにおいて上述の当たりフラグ F2 が「1」にセットされたかどうか、つまり直前のフレームにおいても 2 つのオブジェクトは衝突ないし接触したかどうかを設定するためのフラグである。前フレームにおいて当たり検出がなされたとき「1」が、それ以外るとき「0」がそれぞれ設定される。振動フラグ F4 は、振動源 507 によって振動を発生させるべきとき「1」として、それ以外るとき「0」として設定される。

#### 【0037】

変数  $n\_add$  は、後述の振動発生条件検出ルーチン（図 10）において、1 フレーム毎にカウンタ CT のカウント値  $n$  を増加させるための増分値であり、変数  $n\_add\_m$  は増分値  $n\_add$  の変化値である。カウンタ CT は、データサイズが 32 ビットであり、256 進カウンタ（8 ビットカウンタ）のように、そのカウント値  $n$  が「255」を超えたとき、振動源 507（図 5）を駆動して振動を発生させる。変数  $ax$ 、 $ay$  および  $az$  は、プレイヤオブジェクトの X 軸、Y 軸および Z 軸のそれぞれの加速度である。変数  $fx$ 、 $fy$  および  $fz$  は、加速度  $ax$ 、 $ay$  および  $az$  に定数  $accx$ 、 $accy$  および  $accz$  をそれぞれ乗算した値である。

#### 【0038】

なお、変数  $n\_add$  および  $n\_add\_m$  は、定数であってもよい。実施例では、変数

10

20

30

40

50

n\_\_a d dは、「255」または「150」に設定される。また、変数n\_\_a d d\_\_mは、たとえば、「20」または「10」に設定される。

図9はこの実施例のビデオゲームシステムのメインフロー図であり、電源が投入されると、最初のステップS1において、CPU11はスタートに際してビデオゲーム機10を所定の初期状態に設定する。たとえば、CPU11は、外部ROM21のプログラム領域22に記憶されているゲームプログラムのうちの立ち上げプログラムをRAM14のプログラム領域202に転送し、各パラメータを初期値に設定した後、図9の各ステップを順次実行する。

#### 【0039】

図9のメインフロー図の動作は、たとえば1フレーム(1/60秒)毎または2ないし3フレーム毎に行われるものであり、コースをクリアするまではステップS2~S14が繰り返し実行される。コースクリアに成功することなくゲームオーバーになると、ステップS14に続いて、ステップS15においてゲームオーバー処理が行われる。コースクリアに成功するとステップS13からステップS1へ戻る。

#### 【0040】

すなわち、ステップS1において、ゲームのコース画面および/またはコース選択画面の表示が行われるが、電源投入後にゲームを開始する場合は、最初のコース画面の表示が行われる。最初のコースをクリアすると、次のコースが設定される。

ステップS1に続いて、ステップS2において、コントローラ処理が行われる。この処理は、コントローラ40のジョイスティック45、十字スイッチ46、およびスイッチ47A~47Zの何れが操作されたかを検出し、その操作状態の検出データ(コントローラデータ)を読み込み、読み込んだコントローラデータをRAM14のコントローラデータ領域141に書き込む。

#### 【0041】

ステップS3において、プレイヤーオブジェクトの表示のための処理が行われる。この処理は、基本的には、プレイヤーの操作するジョイスティック45の操作状態と敵からの攻撃の有無に基づいてその姿勢、方向、形状および位置を変化させる処理である。たとえば、外部ROM21の記憶領域22f(図7)から転送されたプログラムと記憶領域24から転送されたプレイヤーオブジェクトのポリゴンデータとコントローラデータすなわちジョイスティック45の操作状態とに基づいて、変化後のポリゴンデータを演算によって求める。その結果得られた複数のポリゴンにテクスチャデータによって色を付与する。

#### 【0042】

ステップS4において、カメラ処理が行われる。たとえば、カメラのファインダを通して見たときの視線または視界がプレイヤーがジョイスティック45によって指定したアングルとなるように、各オブジェクトに対する視点の座標を演算する。

ステップS5において、敵オブジェクトの処理が行われる。この処理は、記憶領域22gおよび一部転送されたプログラムおよび記憶領域24(図2)から転送された敵オブジェクトのポリゴンデータに基づいて実行される。たとえば、プレイヤーオブジェクトの動きを判断しながらプレイヤーオブジェクトに攻撃を加えたり進行を妨げる動きとなるように、敵オブジェクトの表示位置および/またはその形状をポリゴンデータの演算によって求めて、変化した敵オブジェクトの画像が表示される。これによって、敵オブジェクトは、プレイヤーオブジェクトに対して何らかの影響を与えるように働く。

#### 【0043】

ステップS6において、背景(または静止)オブジェクトの処理が行われる。この処理は、記憶領域22hから一部転送されたプログラムと記憶領域24(図2)から転送された静止オブジェクトのポリゴンデータとに基づいて、静止オブジェクトの表示位置およびその形状を演算する。

ステップS7において、RSP122が描画処理を行う。すなわち、RCP12は、CPU11の制御の下に、RAM14の画像データ領域201に記憶されている敵オブジェクト、プレイヤーオブジェクト等の移動オブジェクトや背景等の静止オブジェクトのそれぞれ

10

20

30

40

50

のテクスチャデータに基づいて、移動オブジェクトおよび静止オブジェクトの表示のための画像データの変換処理（座標変換処理およびフレームメモリ描画処理）を行う。具体的には、複数の移動オブジェクトや静止オブジェクト毎の複数のポリゴンに色を付与する。

【 0 0 4 4 】

ステップ S 8 において、CPU 1 1 がメッセージや音楽や効果音等の音声データに基づいて、音声処理を行なう。

次のステップ S 9 において、CPU 1 1 は、振動条件検出処理を行う。すなわち、このステップ S 9 において、RCP 1 2 は、プレイヤーオブジェクトおよび敵オブジェクトもしくは静止オブジェクトの画像データに基づいて、もしくはコントローラ 4 0 からの操作データに基づいて、図 1 0 のサブルーチンを実行することによって、振動源 5 0 7 によって振動を発生させるための条件が成立したかどうかを検出する。ただし、振動条件検出サブルーチンについては、図 1 0 を参照して後に詳細に説明する。

10

【 0 0 4 5 】

次のステップ S 1 0 において、CPU 1 1 は、ステップ S 9 によって振動発生条件を検出したことに応答して、図 1 2 に示すサブルーチンを実行することによって、振動源 5 0 7 によって振動を発生させる。ステップ S 1 0 もまた、図 1 2 を参照して後に詳細に説明する。

ステップ S 1 1 において、CPU 1 1 が、ステップ S 7 において描画処理された結果により、RAM 1 4 のフレームメモリ領域に記憶されている画像データを読み出す。したがって、プレイヤーオブジェクト、移動オブジェクト、静止オブジェクトおよび敵オブジェクト等がディスプレイ 3 0（図 1，図 2）の表示画面上に表示される。

20

【 0 0 4 6 】

ステップ S 1 2 において、RCP 1 2 がステップ S 1 8 において音声処理した結果得られる音声データを読み出すことにより、音楽および効果音または会話等の音声を出力させる。

ステップ S 1 3 において、コースをクリアしたか否かが判断（コースクリア検出）され、コースをクリアしていなければステップ S 1 4 においてゲームオーバになったか否かが判断され、ゲームオーバでなければステップ S 2 へ戻り、ゲームオーバの条件が検出されるまでステップ S 2 ~ S 1 4 が繰り返される。そして、プレイヤーに許容されているミス回数が所定の回数になるか、プレイヤーオブジェクトのライフを所定数量使い切る等のゲームオーバ条件になったことが検出されると、続くステップ S 1 5 においてゲームの継続またはバックアップデータの記憶の選択等のゲームオーバ処理が行われる。

30

【 0 0 4 7 】

なお、ステップ S 1 3 において、コースをクリアした条件（たとえば、ボスを倒す等）が検出されると、コースクリアの処理をした後、ステップ S 1 へ戻る。図 1 0 を参照して、振動発生条件検出サブルーチンの最初のステップ S 1 0 1 では、CPU 1 1 は、プレイヤーオブジェクトとそれに対する影響物体ないし障害物（他の移動オブジェクト、地面、海面、壁等の静止オブジェクト、敵オブジェクト、攻撃オブジェクト等）とが当たった（衝突ないし接触）か否かを判断する。この当たり検出は、図 1 1 のサブルーチンに従って実行される。

40

【 0 0 4 8 】

図 1 1 のステップ S 2 0 1 において、CPU 1 1 は、ABS (OBJ 2x - OBJ 1x) OBJ 1r であるか否か、すなわち、X 座標系で 2 つのオブジェクトが当たっているか否かを判断する。OBJ 1 は、当たり判定をする対象物のオブジェクトであり、この実施例では、プレイヤーオブジェクトのことである。OBJ 2 は、OBJ 1 に向かって近づいてくるオブジェクトであり、この実施例では、仲間オブジェクト、敵オブジェクト、静止オブジェクトおよび敵オブジェクトが発射した攻撃オブジェクトのことである。OBJ 1x は、OBJ 1 の X 座標値であり、OBJ 2x は、OBJ 2 の X 座標値である。OBJ 1x と OBJ 2x とは、同じ座標系の X 座標値であれば、ゲーム空間座標であってもプレイヤー座標であっても

50

よい。ABS( )は、( )内の数値の絶対値を表す。OBJ1rは、OBJ1を立方体として考えた場合の立方体の一辺の半分の長さを示す値である。換言すると、OBJ1rは、OBJ1の当たり判定範囲を示す値である。もし、ABS(OBJ2x - OBJ1x) OBJ1rであるならば、ステップS202に進む。

【0049】

ステップS202において、CPU11は、ABS(OBJ2y - OBJ1y) OBJ1rであるか否か、すなわち、Y座標系で2つのオブジェクトが当たっているか否かを判断する。OBJ1yは、OBJ1のY座標値であり、OBJ2yは、OBJ2のY座標値である。OBJ1yとOBJ2yとは、同じ座標系のY座標値であれば、ゲーム空間座標であってもプレイヤー座標であってもよい。もし、ABS(OBJ2y - OBJ1y) OBJ1rであるならば、ステップS203に進む。

10

【0050】

ステップS203において、CPU11は、ABS(OBJ2z - OBJ1z) OBJ1rであるか否か、すなわち、Z座標系で2つのオブジェクトが当たっているか否かを判断する。OBJ1zは、OBJ1のZ座標値であり、OBJ2zは、OBJ2のZ座標値である。OBJ1zとOBJ2zとは、同じ座標系のZ座標値であれば、ゲーム空間座標であってもプレイヤー座標であってもよい。もし、ABS(OBJ2z - OBJ1z) OBJ1rであるならば、ステップS204に進む。

【0051】

ステップS204において、CPU11は、OBJ2とOBJ1とが当たったと判断し、RAM14のフラグ領域142の当たりフラグF2を「1」にセットする。一方、ステップS201において、ABS(OBJ2x - OBJ1x) OBJ1rでないならば、元のルーチンに戻り、ステップS202において、ABS(OBJ2y - OBJ1y) OBJ1rでないならば、元のルーチンに戻り、ステップS203において、ABS(OBJ2z - OBJ1z) OBJ1rでないならば、元のルーチンに戻る。

20

【0052】

図10のステップS101において、プレイヤーオブジェクトが他のオブジェクトに当たっていないことを検出したとき、つまり、ステップS101において“NO”が判断されたとき、CPU11は、次のステップS102において、プレイヤーがコントローラ40を操作し、プレイヤーオブジェクトのアクセルをスタートさせたか否かを判断する。たとえば、「ウェーブレース」においてプレイヤーオブジェクトが「ジェットスキー」なら、Aボタン47A(図1)を押すことによって、また、プレイヤーオブジェクトが「マリオ」なら、ジョイスティック45(図1)を前に倒すことによって、アクセルスタートが実行される。したがって、CPU11は、このステップS102において、RAM14のコントローラデータ領域141のデータを参照して、Aボタン47Aやジョイスティック45が操作されたかどうかを判断する。

30

【0053】

ステップS102において、“NO”が判断されると、CPU11は、次のステップS103において、プレイヤーオブジェクト(この実施例では、「ウェーブレース」の「ジェットスキー」)が水面に接しているかどうかを判断する。ステップS103において、プレイヤーオブジェクト(「ジェットスキー」)が水面に接しているかどうかを判断するために、図11の当たり検出サブルーチンが利用される。

40

【0054】

このように、図10に示す振動発生条件検出サブルーチンにおいては、3つの振動発生条件すなわちステップS101、S102およびS103のいずれかを検出する。いずれのステップS101、S102およびS103においても“NO”が判断されると、すなわち、振動発生条件のいずれもが検出されないと、図10のステップS104において、CPU11は、RAM14のフラグ・レジスタ領域142のレジスタ値n\_\_addを「0」にリセットする。それとともに、CPU11は、次のステップS105において、フラグ・レジスタ領域142の前フレームフラグF3を「0」にクリアする。すなわち、ステッ

50

ステップS104においては、カウンタCTの増分値n\_\_addを「0」に設定するため、カウンタCTのカウント値nは1フレーム毎に「0」加算される。換言すれば、全てのステップS101、S102およびS103において“NO”が判断されると、カウンタCTは全くインクリメントされない。後で述べるように、たとえばカウンタCTのカウント値が「255」を超えると振動が発生されるのであるから、この場合には、振動源507による振動は発生されない。

#### 【0055】

なお、先のステップS101において“NO”が判断されるため、ステップS105では、前フレームフラグF3がリセットされる。

ステップS101、S102またはS103のいずれかにおいて“YES”が判断されると、振動発生条件が成立したので、それぞれの振動発生条件に従って、振動発生処理が実行される。

10

#### 【0056】

すなわち、ステップS101において、プレイヤーオブジェクトが他のオブジェクトと当たったことを検出すると、ステップS106ないしS110を実行し、振動源507から強い振動を発生させる。また、ステップS102において、アクセルスタートが検出されると、ステップS111ないしS113によって、弱い振動が発生される。プレイヤーオブジェクト（たとえば「ジェットスキー」）が水面に接していることをステップS103で検出すると、ステップS114ないしS117を実行し、「ジェットスキー」が波の上をバウンドする状態を表す弱い振動を発生させる。

20

#### 【0057】

先のステップS101において“YES”と判断されたとき、すなわち、プレイヤーオブジェクトが他のオブジェクトと衝突ないし接触したと判断したとき、CPU11は、前フレームフラグF3が「0」かどうか判断する。すなわち、このステップS106においては、前フレームにおいてもプレイヤーオブジェクトが他のオブジェクトと衝突していたかどうかを判断する。ステップS106において“YES”と判断されたとき、すなわち、前フレームではプレイヤーオブジェクトの衝突ないし接触はなく、現フレームにおいてプレイヤーオブジェクトと他のオブジェクトとの衝突ないし接触が検出されたとき、CPU11は、次のステップS107において、RAM14のフラグ・レジスタ領域142の増分値n\_\_addを、強い振動を発生させるように、たとえば「255」に設定する。

30

#### 【0058】

つまり、図13に示すように、プレイヤーオブジェクトが敵オブジェクトに衝突ないし接触した場合、あるいは、図14に示すように、プレイヤーオブジェクトが敵オブジェクトないし静止オブジェクトに衝突ないし接触した場合には、強い振動を発生するために、ステップS107において、大きい増分値n\_\_addを設定する。

#### 【0059】

次いで、ステップS108において、CPU11は、増分値n\_\_addの変化値n\_\_add\_\_mを「20」に設定する。他のオブジェクトと衝突ないし接触したときには、強い振動を発生できるように、ステップS107において増分値n\_\_addを「255」にするとともに、ステップS108において、変化値n\_\_add\_\_mを「20」に設定する。変化値n\_\_add\_\_mの値が大きいと増分値n\_\_addが短い時間で「0」になり、変化値n\_\_add\_\_mの値が小さいと、増分値n\_\_addが「0」になるまでに長い時間継続する。したがって、変化値n\_\_add\_\_mが大きいとき、振動源507による振動が長時間継続することになる。ステップS108に続くステップS109では、CPU11は、前フレームフラグF3を「1」に設定する。すなわち、現フレームにおいてプレイヤーオブジェクトの当たりが検出されたのであるから、その時点で前フレームフラグF3を「1」に設定するのである。

40

#### 【0060】

なお、先のステップS106において“NO”が検出されると、すなわち前フレームフラグF3が「1」であるとき、つまり、前フレームにおいてもプレイヤーオブジェクトの衝突

50

ないし接触が検出されているときには、ステップS110において、増分値 $n\_add$ を「0」にリセットする。すなわち、前フレームにおいてもプレイヤーオブジェクトが他のオブジェクトと接触ないし衝突したときには、継続的な振動の発生を防止するため、ステップS110において、増分値 $n\_add$ を「0」とする。

#### 【0061】

ステップS102において、プレイヤーオブジェクトのアクセルスタートが検出されると、次のステップS111において、CPU11は、増分値 $n\_add$ を、弱い振動を発生させるように、たとえば「150」に設定する。すなわち、プレイヤーオブジェクトのアクセルがスタートされたとき、先の当たり検出の場合と異なり比較的小さい振動を発生するために、カウンタCTの増分値 $n\_add$ をステップS107より小さい「150」に設定する。そして、ステップS112において、CPU11が、変化値 $n\_add\_m$ を「10」に設定する。ステップS113において、前フレームフラグF3を「0」とする。すなわち、このステップS113は、先のステップS101において“NO”と判断されたとき実行されるステップであるから、先のステップS109とは異なり、前フレームフラグF3を「0」とする。

10

#### 【0062】

さらに、ステップS103において、プレイヤーオブジェクト、実施例においては「ジェットスキー」が水面に接していることが検出されると、次のステップS114において、乗算値 $f_x$ 、 $f_y$ および $f_z$ を計算する。そして、ステップS115において、CPU11は、カウンタCTの増分値 $n\_add$ として、「 $f_x \times f_x + f_y \times f_y + f_z \times f_z$ 」の平方根（小数点以下切り捨て）を計算する。つまり、このステップS114およびS115では、プレイヤーオブジェクトのX軸方向、Y軸方向およびZ軸方向の加速度 $a_x$ 、 $a_y$ および $a_z$ を求め、その加速度に比例した値 $f_x$ 、 $f_y$ および $f_z$ を求め、この比例値ないし乗算値に応じて、「0」～「255」の範囲の増分値 $n\_add$ を計算する。そして、ステップS116において、増分値 $n\_add$ の変化値 $n\_add\_m$ として、たとえば「10」を設定する。すなわち、プレイヤーオブジェクト「ジェットスキー」が水面に接している場合には比較的長時間衝撃を与えるために、変化値 $n\_add\_m$ としては相対的に小さい「10」を設定する。そして、この場合にも先のステップS101において“NO”が判断されたのであるから、続くステップS117において、前フレームフラグF3を「0」にする。

20

30

#### 【0063】

振動発生サブルーチンを示す図12の最初のステップS120において、CPU11は、RAM14のフラグ・レジスタ領域142の振動ゲームフラグF1が「1」であるかどうか、すなわち、現在プレイ内のゲームが振動付加ゲームであるかどうかを判断する。そして、このステップS120において“YES”が判断されると、CPU11は、RAM14のコントローラデータ領域141を参照して、振動カートリッジ50（図1、図4）がコントローラ40に装着されているかどうかを判断する。振動ゲームでありかつ振動カートリッジ50がコントローラ40に装着されているとき、次のステップS122において、CPU11は、振動発生条件が成立したにも拘わらず振動を強制的に停止すべきかどうかすなわち振動がリセットされたかどうかを判断する。このステップS122において“NO”が判断されると、CPU11は、次のステップS123において、カウンタCTのカウント値 $n$ を「 $n + n\_add$ 」とする。すなわち、カウンタCTのカウント値 $n$ を増分値 $n\_add$ に従って増加する。

40

#### 【0064】

そして、次のステップS124において、その増分値 $n\_add$ を、変化値 $n\_add\_m$ に従って修正する。すなわち、このステップS124においては、ディスプレイ30の1フレーム毎に増分値 $n\_add$ の値を変化値 $n\_add\_m$ だけ減算するため、カウンタCTのカウント値 $n$ の値の増分値がフレーム毎に小さくなり、やがて、カウント値 $n$ が増加しなくなる。換言すると、振動源507による振動は、最初は大きく（強く）かつ徐々に小さく（弱く）なり、やがて振動しなくなるのである。

50

## 【 0 0 6 5 】

そして、ステップ S 1 2 5 においては、カウンタ C T のカウント値 n が「 2 5 5 」を超えたかどうかを判断する。このステップ S 1 2 5 において“ Y E S ”が判断されると、カウンタ C T のカウント値 n を「 n - 2 5 6 」とする。すなわち、ステップ S 1 2 5 において、カウンタ C T のカウント値 n が「 2 5 5 」を超えたと判断されると、次のステップ S 1 2 6 においては、そのカウント値 n から「 2 5 6 」を減算する。そして、ステップ S 1 2 7 において、 C P U 1 1 は、振動フラグ F 4 を「 1 」に設定する。先のステップ S 1 2 5 においてカウンタ C T のカウント値 n が「 2 5 5 」を超えたことが検出されたので、このステップ S 1 2 7 においては、振動フラグ F 4 を「 1 」に設定する。

## 【 0 0 6 6 】

そして、次のステップ S 1 2 8 においては、振動フラグ F 4 が「 1 」かどうかを判断する。先のステップ S 1 2 7 において、振動フラグ F 4 が「 1 」に設定されていると、振動を発生するために、次のステップ S 1 2 9 において、 C P U 1 1 は、アドレス A 1 5 を除く全てのアドレス A 2 - A 1 4 に「 1 」を出力するとともに、書込信号を出力し、かつ、チップイネーブル信号を出力する。したがって、このステップ S 1 2 9 において、レコーダないし N A N D ゲート 5 1 1 ( 図 6 ) から信号が出力され、ラッチ 5 1 2 が C P U 1 1 のデータビット D 0 をラッチする。ステップ S 1 2 9 においては、振動源 5 0 7 において振動を発生すべきであるので、 C P U 1 1 のデータビット D 0 は「 1 」として出力される。したがって、ラッチ 5 1 2 ( 図 6 ) には「 1 」がラッチされ、応じて、トランジスタ 5 1 4 がオンし、振動源ないし振動モータ 5 0 7 に電池 5 0 4 から電力が供給され、振動源 5 0 7 すなわち振動カートリッジ 5 0、つまりコントローラ 4 0 が振動する。

## 【 0 0 6 7 】

なお、先のステップ S 1 2 5 において“ N O ”が判断されると、ステップ S 1 3 0 において、振動フラグ F 4 が「 0 」にリセットされる。すなわち、カウンタ C T のカウント値 n が「 2 5 5 」を超えないときには、振動フラグ F 4 はリセットされたままである。そして、ステップ S 1 2 7 において、振動フラグ F 4 が「 1 」でないとき、すなわち振動フラグ F 4 が「 0 」であるときには、ステップ S 1 3 1 において振動を停止するために、 C P U 1 1 は、データビット D 0 に「 0 」を出力する。したがって、ラッチ 5 1 2 に「 0 」がラッチされ、トランジスタ 5 1 4 がオフする。したがって、振動源 5 0 7 には電流が流れず、振動源 5 0 7 から振動が発生されることはない。

## 【 0 0 6 8 】

たとえば、プレイヤーオブジェクトが敵オブジェクトや静止オブジェクトに衝突ないし接触した場合 ( 図 1 3 , 図 1 4 )、ステップ S 1 0 7 において、カウンタ C T のカウント値 n の増分値 n \_ \_ a d d として「 2 5 5 」が設定されかつステップ S 1 0 8 において、変化値 n \_ \_ a d d \_ \_ m として「 2 0 」が設定されるため、表 1 および図 1 5 に示すように、カウンタ C T のカウント値 n が最初のフレームを除く 4 フレーム連続して「 2 2 5 」を超えるため、振動源 5 0 7 が 4 フレーム連続して駆動され、その後 2 フレーム毎にカウント値 n が「 2 5 5 」を超え、振動源 5 0 7 は 2 フレーム毎に駆動される比較的強い振動を発生する。

## 【 0 0 6 9 】

ところが、プレイヤーオブジェクトのアクセルスタートのときには、比較的弱い振動を発生させるために、ステップ S 1 1 1 において増分値 n \_ \_ a d d として「 1 5 0 」が設定され、ステップ S 1 1 2 において、変化値 n \_ \_ a d d \_ \_ m として「 1 0 」が設定される。したがって、この場合には、表 2 および図 1 6 に示すように、2 フレーム毎に、カウント値 n が「 2 5 5 」を超えるため、振動源 5 0 7 は 2 フレーム毎に駆動され、そして、次の 2 フレームは振動を休止し、次の 1 フレーム振動源 5 0 7 を駆動した後、さらに、次の 2 フレーム振動を休止する。

## 【 0 0 7 0 】

すなわち、強い振動を発生する場合には、振動源 5 0 7 は数フレーム連続して駆動され、その後 2 フレーム毎に駆動されるように振動が漸減する。弱い振動を発生すべきときには

10

20

30

40

50

、2フレーム毎に駆動源507が駆動され、その後3フレーム毎に駆動されるように振動が漸減する。ただし、増分値  $n\_add$  や変化値  $n\_add\_m$  すなわちこのような振動発生パターンは、任意に設定できることはいうまでもない。

【0071】

なお、第3の振動発生条件すなわち、プレイヤーオブジェクト(「ジェットスキー」)が水面に接している場合には、増分値  $n\_add$  はプレイヤーオブジェクトの加速度の関数として設定されるため、その加速度に応じて強い振動または弱い振動が発生される。

【0072】

【表1】

強い振動の例				
フレーム	n	n_add	n_add m	振動信号
1	255	255	20	0
2	234	235	20	1
3	193	215	20	1
4	132	195	20	1
5	51	175	20	1
6	206	155	20	0
7	85	135	20	1
8	200	115	20	0
9	39	95	20	1
10	114	75	20	0
11	169	55	20	0
12	204	35	20	0
13	219	15	20	0
14	219	0	20	0
15	219	0	20	0
16	219	0	20	0
17	219	0	20	0
18	219	0	20	0
19	219	0	20	0
20	219	0	20	0

10

20

30

【0073】

【表2】



弱い振動の例				
フレーム	n	n_add	n_add_m	振動信号
1	150	150	10	0
2	34	140	10	1
3	164	130	10	0
4	28	120	10	1
5	138	110	10	0
6	238	100	10	0
7	72	90	10	1
8	152	80	10	0
9	222	70	10	0
10	26	60	10	1
11	76	50	10	0
12	116	40	10	0
13	146	30	10	0
14	166	20	10	0
15	176	10	10	0
16	176	0	10	0
17	176	0	10	0
18	176	0	10	0
19	176	0	10	0
20	176	0	10	0

10

20

## 【0074】

なお、上述の実施例においては、図15または図16に示す強い振動または弱い振動を発生させるために、図12のフロー図を実行して、カウンタCTのカウンタ値nを増分値n\_addや変化値n\_add\_mに従って変化させ、そのカウンタ値nが「255」を超えたとき振動フラグF4を「1」に設定して、振動源507を駆動するようにしている。つまり、先の実施例では、振動源507によって振動を発生させるかどうかはリアルタイムで計算しながら決定した。

## 【0075】

これに対して、たとえば図17に示すように、RAM14のフラグ・レジスタ領域142に外部ROM21のプログラム領域22aから読み出した強振動パターンデータおよび弱振動パターンデータをそれぞれ記憶しておき、これらの振動パターンデータを選択的に読み出してそれに基づいて振動源507を制御するようにしてもよい。なお、強振動パターンデータとしては、先の表1におけるフレーム1～16のフレーム順次の振動信号「0111101010000000」を用いることが考えられる。また、弱振動パターンデータとしては、先の表2におけるフレーム1～16のフレーム順次の振動信号「0101001001000000」を用いることが考えられる。なお、必要に応じて、中振動パターンデータも記憶してもよい。

30

## 【0076】

たとえば図10のステップS101でプレイヤオブジェクトの衝突ないし接触を検出したときには強振動パターンデータを読み出すと、最初のフレームでは振動源507は駆動されず、続く4フレーム連続して振動源507が駆動され、その後2フレーム毎に振動源507が駆動され、以後振動源507の駆動が停止される。たとえば図10のステップS102でアクセルスタートを検出したときまたはステップS103で水面との接触を検出したときには弱振動パターンデータを読み出す。したがって、最初の4フレームでは振動源507は2フレーム毎に駆動され、その後3フレーム毎に振動源507が駆動され、以後振動源507の駆動が停止される。

40

## 【0077】

さらに、上述の実施例では、プレイヤオブジェクトが他のオブジェクトと当たったとき等において、コントローラ40に装着された振動カートリッジ50の振動源507によって

50

振動を発生させた。この振動と時間的に関連するタイミングで、ゲーム画像に視覚上の振動を付与するようにしてもよい。この場合、ディスプレイ30上でのゲーム画像の振動はコントローラ40の振動カートリッジ50によって生じる機械的振動より早く認識されるので、図9のステップS10で振動カートリッジ50による機械的振動を発生させた後、1ないし2フレーム以上遅れて、図9のステップS4またはステップS7によってゲーム画像に振動を付与すればよい。

【0078】

たとえば図10のステップS101でプレイヤオブジェクトの他のオブジェクトとの接触ないし衝突が検出されると、強い振動をゲーム画像上に発生させるために、ディスプレイ30の表示画面全体を振動させる。この場合、図9のステップS4のカメラ処理において、先に述べたカメラの視点座標を小刻みに変化させるようにすればよい。

10

【0079】

図10のステップS102またはS103において弱い振動を発生させるべき条件を検出すると、弱い振動をゲーム画像上に発生させるために、ディスプレイ30に表示されているプレイヤオブジェクト画像にのみ振動を付与する。この場合、図9のステップS7の描画処理において、プレイヤオブジェクトを形成する複数のポリゴン集合体の中心座標位置を小刻みに変化させるようにすればよい。

【0080】

なお、ゲーム画像に振動を付与するかどうかは、図12のステップS128によって「F4=1」が検出されたかどうかを条件とするようにすればよい。つまり、振動フラグF4が「1」のときゲーム画像に振動を付与し、「0」のときにはゲーム画像の振動を発生させない。

20

【図面の簡単な説明】

【図1】この発明の一実施例のビデオゲームシステムを示す概略図解図である。

【図2】図1のビデオゲーム機を詳細に示すブロック図である。

【図3】図2のコントローラ制御回路をより詳細に示すブロック図である。

【図4】図2のコントローラ制御回路および振動カートリッジを詳細に示すブロック図である。

【図5】振動カートリッジを詳細に示す図解図である。

【図6】振動カートリッジに含まれる振動発生回路を示す回路図である。

30

【図7】外部ROMのメモリマップを示す図解図である。

【図8】RAMのメモリマップを示す図解図である。

【図9】図1実施例の全体の動作を示すフロー図である。

【図10】振動発生条件検出サブルーチンを示すフロー図である。

【図11】当たり判定サブルーチンを示すフロー図である。

【図12】振動発生サブルーチンを示すフロー図である。

【図13】プレイヤオブジェクトと敵オブジェクトとの衝突を示す図解図である。

【図14】プレイヤオブジェクトと静止オブジェクトとの衝突を示す図解図である。

【図15】強く短い振動を発生させる振動パターンを示すグラフである。

【図16】弱く長い振動を発生させる振動パターンを示すグラフである。

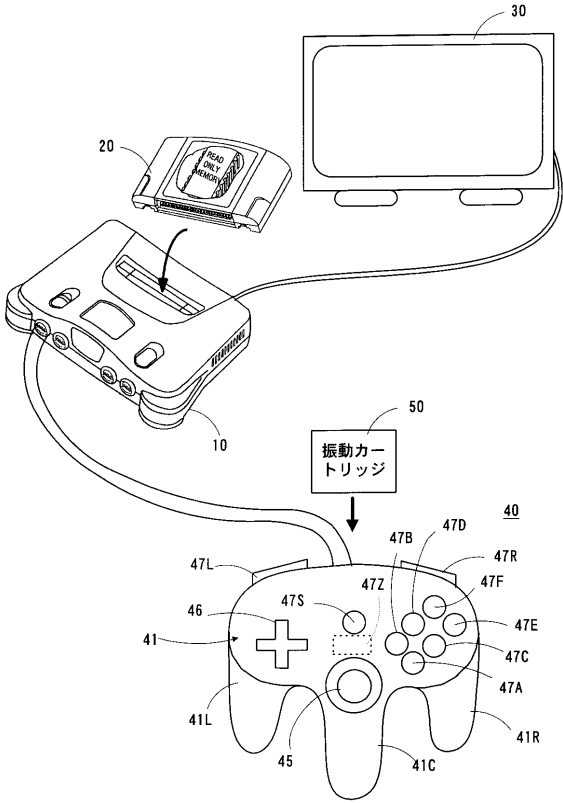
40

【図17】この発明の実施例におけるRAMのメモリマップを示す図解図である。

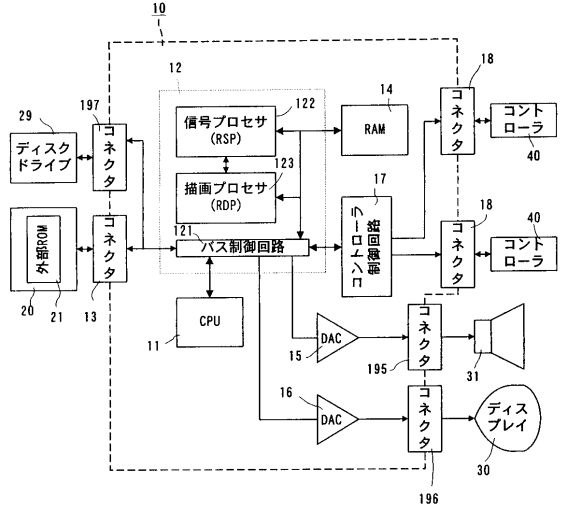
【符号の説明】

- 10 ... ビデオゲーム機
- 11 ... CPU
- 17 ... コントローラ制御回路
- 20 ... ROMカートリッジ
- 40 ... コントローラ
- 50 ... 振動カートリッジ

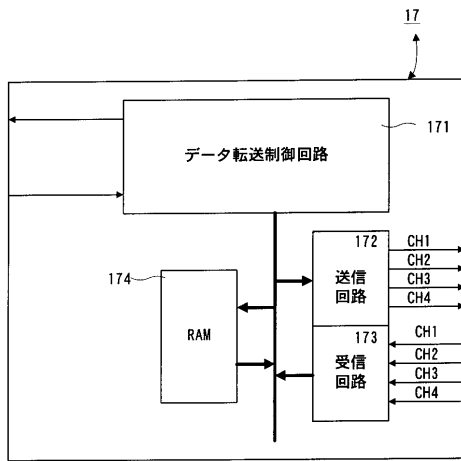
【 図 1 】



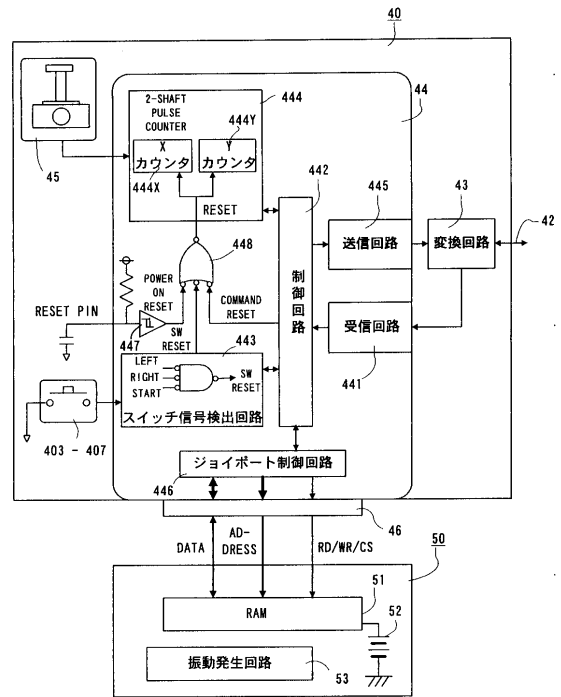
【 図 2 】



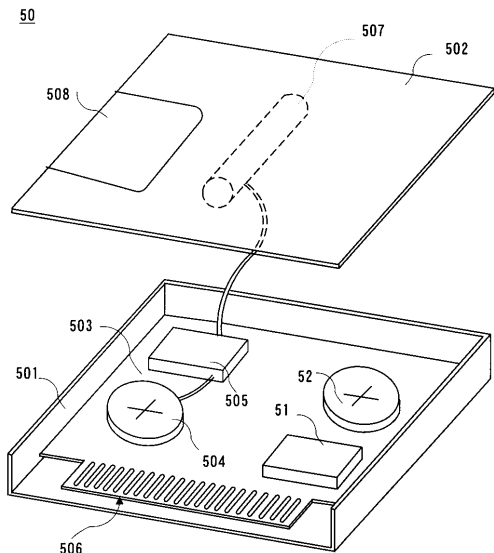
【 図 3 】



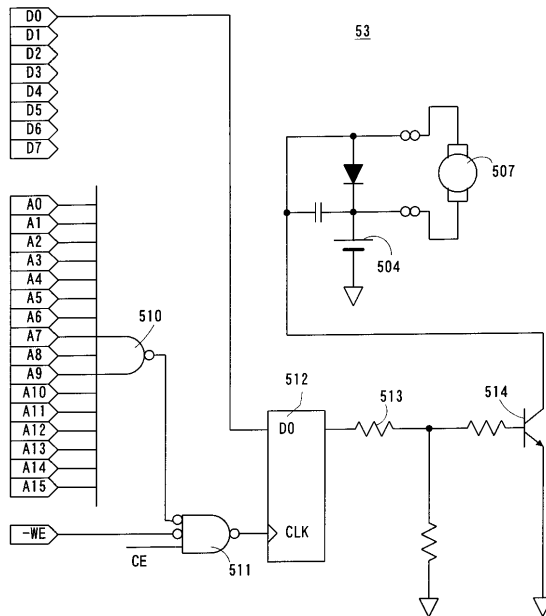
【 図 4 】



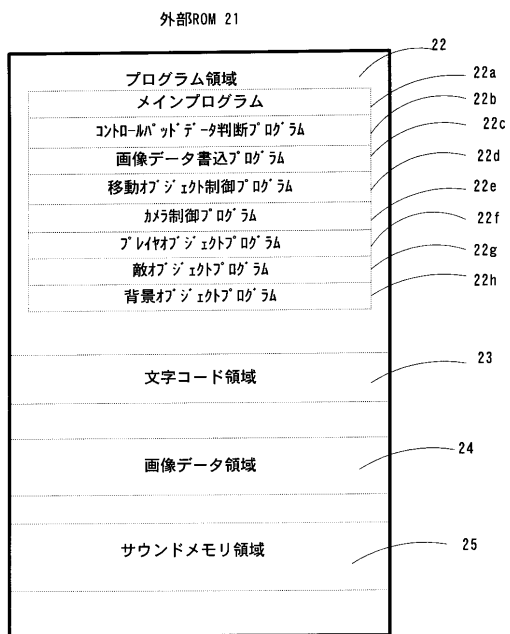
【図5】



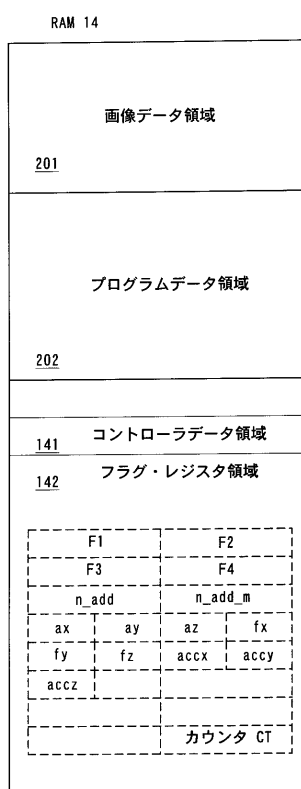
【図6】



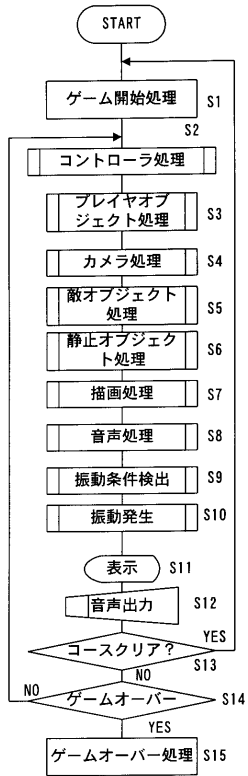
【図7】



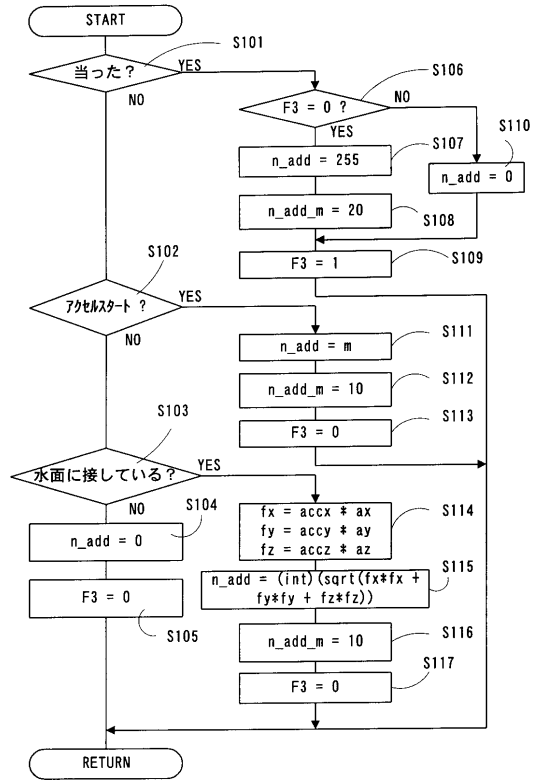
【図8】



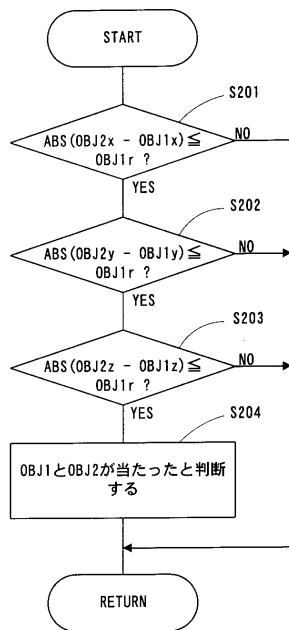
【図9】



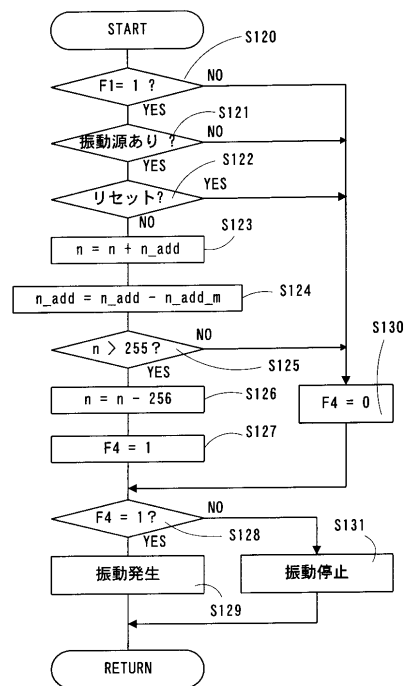
【図10】



【図11】

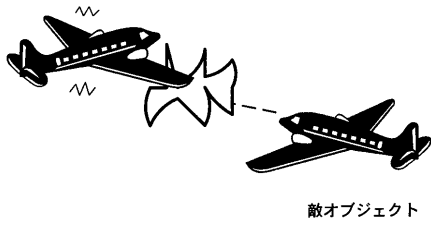


【図12】

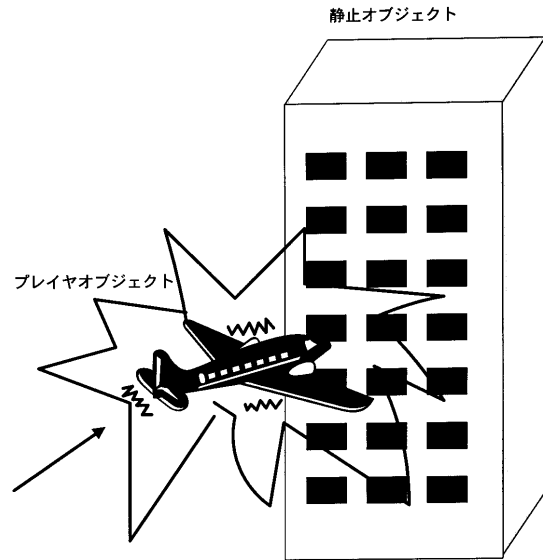


【 図 1 3 】

プレイヤーオブジェクト

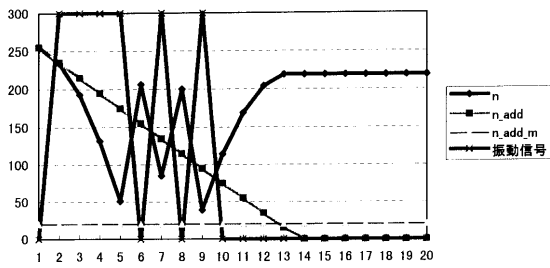


【 図 1 4 】



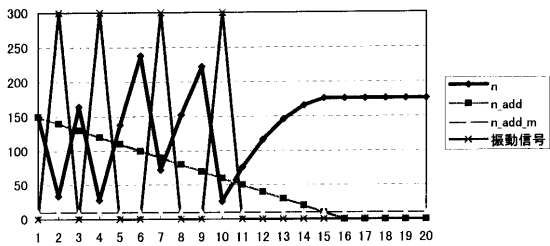
【 図 1 5 】

強い振動例

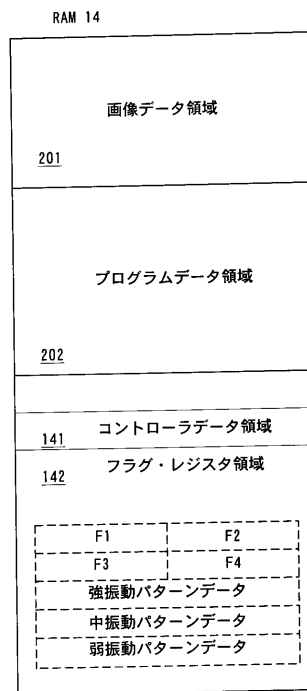


【 図 1 6 】

弱い振動例



【 図 1 7 】



フロントページの続き

審査官 宮本 昭彦

(56)参考文献 特開平09 - 164270 (JP, A)

(58)調査した分野(Int.Cl.<sup>7</sup>, DB名)

A63F 13/00 - 13/12