(19)日本国特許庁(JP)　(12)公開特許公報(A)　(11)公開番号
特開2023-21045
(P2023-21045A)
(43)公開日　令和5年2月9日(2023.2.9)

(51)国際特許分類		FI			テーマコード（参考）
Ｇ０６Ｆ	*8/76 (2018.01)*	Ｇ０６Ｆ	8/76		５Ｂ０８１
Ｇ０６Ｆ	*8/41 (2018.01)*	Ｇ０６Ｆ	8/41	１７０	５Ｂ３７６

審査請求　有　請求項の数　22　　ＯＬ　（全127頁）

(21)出願番号　特願2022-120245(P2022-120245)
(22)出願日　令和4年7月28日(2022.7.28)
(31)優先権主張番号　17/387,030
(32)優先日　令和3年7月28日(2021.7.28)
(33)優先権主張国・地域又は機関
米国(US)

(71)出願人　518187455
ソニー・インタラクティブエンタテインメント エルエルシー
アメリカ合衆国、カリフォルニア州、９４４０４、サン マテオ、ブリッジポイント パークウェイ ２２０７
(74)代理人　100105924
弁理士　森下 賢樹
(72)発明者　エルネスト コルヴィ
アメリカ合衆国、カリフォルニア州 ９４４０４、サン マテオ、ブリッジポイント パークウェイ ２２０７
(72)発明者　ジョージ ウェイジン
アメリカ合衆国、カリフォルニア州 ９４４０４、サン マテオ、ブリッジポイ
最終頁に続く

(54)【発明の名称】　レガシーゲーム用ＡＯＴコンパイラ

(57)【要約】　（修正有）
【課題】レガシーゲーム用の事前（ＡｏＴ）コンパイラを提供するシステム及び方法を提供する。
【解決手段】方法は、レガシーゲームの実行可能コードのためにメモリデバイスをスキャンすることと、実行可能コードを復号化することと、実行可能コードを第１のソースコードに変換し、第１のコンパイルされたコードを出力するために第１のソースコードをコンパイルすることと、レガシーゲームのプレイ中にユーザ入力が受信されるかどうかを判定することと、ユーザ入力を処理するためのファイバが作成されるかどうかを判定することと、を含む。ファイバは、第１のコンパイルされたコードと第２のコンパイルされたコードとの間で実行の方向性を提供する。方法はさらに、ユーザ入力を処理するためのファイバが作成されたと判定することに応答して、レガシーゲームの１つ以上の画像フレームを生成するためにファイバを実行する。
【選択図】なし

【特許請求の範囲】
【請求項１】
レガシーゲームのプレイ中の事前コンパイル（ＡｏＴ）のための方法であって、
レガシーゲームの実行可能コードについてメモリデバイスをスキャンすることと、
前記実行可能コードを復号化することと、
前記実行可能コードを第１のソースコードに変換することと、
第１のコンパイルされたコードを出力するために前記第１のソースコードをコンパイルすることと、
前記レガシーゲームの前記プレイ中にユーザ入力が受信されたかどうかを判定することと、
前記ユーザ入力を処理するためのファイバが作成されているかどうかを判定することであって、前記ファイバが前記第１のコンパイルされたコードと第２のコンパイルされたコードとの間で実行の方向性を提供する、前記判定することと、
第１の処理ユニットによって、前記ユーザ入力を処理するための前記ファイバが作成されたとの判定に応答して前記ファイバを実行することであって、前記ファイバが、前記レガシーゲームの１つ以上の画像フレームを生成するために実行される、前記実行することと、
を含む、方法。
【請求項２】
前記ファイバが生成されていないと判定することに応じて、前記第１のコンパイルされたコードが、１つ以上のネイティブライブラリ内に配置されているかどうかを判定することであって、前記第１のコンパイルされたコードが、前記ユーザ入力を処理するためのものである、前記判定することと、
前記１つ以上のネイティブライブラリ内で前記第１のコンパイルされたコードを特定した後に前記ファイバを作成することであって、前記ファイバを前記実行することが、前記ファイバを前記作成した後に行われている、前記作成することと、
をさらに含む、請求項１に記載の方法。
【請求項３】
前記第１のコンパイルされたコードが、前記１つ以上のネイティブライブラリ内にないという判定に応答して、前記ユーザ入力を処理するための１つ以上の基本ブロックがキャッシュされているかどうかを判定することと、
前記１つ以上の基本ブロックがキャッシュされているという判定に応答して、前記ユーザ入力を処理するための前記１つ以上の基本ブロックを実行することと、
をさらに含む、請求項２に記載の方法。
【請求項４】
前記実行可能コードのための前記スキャンが、前記レガシーゲームの前記実行可能コードを前記レガシーゲームのデータから区別することを含む方法であって、前記復号化のための前記データを提供することなく、前記復号化のための前記実行可能コードを提供することをさらに含む、請求項１に記載の方法。
【請求項５】
前記データが、画像データまたは音声データを含む、請求項４に記載の方法。
【請求項６】
前記方向性が、前記第１のコンパイルされたコード、及び前記第２のコンパイルされたコードの操作のシーケンスを提供する、請求項１に記載の方法。
【請求項７】
前記ファイバが、キャッシュの１つ以上のレジスタの１つ以上の状態を格納する操作と、
前記第１のコンパイルされたコードから前記第２のコンパイルされたコードに切り替える操作と、
前記第２のコンパイルされたコードの実行後に、前記１つ以上のレジスタの前記１つ以

上の状態を復元する操作と、
前記第２のコンパイルされたコードから前記第１のコンパイルされたコードに切り替える操作と、
を含む、請求項１に記載の方法。
【請求項８】
前記スキャンすることが、第２の処理ユニットによって実行され、前記キャッシュが、ゲームコンソールのものであり、前記ファイバが、前記キャッシュに格納され、前記キャッシュが、前記第１の処理ユニットによってアクセスされるように構成されている、請求項７に記載の方法。
【請求項９】
前記第１の処理ユニットが、前記ゲームコンソール内に配置されており、前記第２の処理ユニットが、サーバシステム内に配置されている、請求項８に記載の方法。
【請求項１０】
前記第２のコンパイルされたコードが、前記１つ以上の画像フレームのうちの１つをリフレッシュする周期的リフレッシュ画像操作、または前記１つ以上の画像フレームのうちの前記１つで出力されるリフレッシュサウンドの周期的リフレッシュオーディオ操作を含む、請求項１に記載の方法。
【請求項１１】
前記メモリデバイスを前記スキャンすること、前記実行可能コードを前記復号化すること、前記実行可能コードを前記変換すること、及び前記第１のソースコードを前記コンパイルすることが、前記ユーザ入力が受信される前に実行されている、請求項１に記載の方法。
【請求項１２】
レガシーゲームのプレイ中の事前（ＡｏＴ）コンパイルのためのシステムであって、
第１のコンピューティングデバイスであって、
前記レガシーゲームの実行可能コードについてメモリデバイスをスキャンするように構成されたスキャナと、
前記実行可能コードを復号化するように構成されたデコーダと、
前記実行可能コードを第１のソースコードに変換するように構成されたトランスレータと、
前記第１のソースコードをコンパイルして第１のコンパイルされたコードを出力するように構成されたコンパイラと、を含む、前記第１のコンピューティングデバイスと、
前記第１のコンピューティングデバイスに結合されるように構成された第２のコンピューティングデバイスであって、
前記レガシーゲームの前記プレイ中にユーザ入力が受信されたかどうかを判定し、
前記ユーザ入力を処理するためのファイバが作成されているかどうかを判定し、前記ファイバが、前記第１のコンパイルされたコードと第２のコンパイルされたコードの間で実行の方向性を提供し、
前記ユーザ入力を処理するための前記ファイバが作成されたと判定されたとき、前記ファイバを実行し、前記ファイバが、前記レガシーゲームの１つ以上の画像フレームを生成するために実行されるように構成されている、前記第２のコンピューティングデバイスと、
を備える、システム。
【請求項１３】
前記第２のコンピューティングデバイスが、
前記ファイバが作成されていないと判定されたときに、前記第１のコンパイルされたコードが、１つ以上のネイティブライブラリ内に配置されているかどうかを判定するように構成され、前記第１のコンパイルされたコードが、前記ユーザ入力を処理するためのものであり、
前記１つ以上のネイティブライブラリ内で前記第１のコンパイルされたコードを特定し

た後に、前記ファイバを作成するように構成され、前記ファイバが作成された後に前記ファイバが実行される、請求項１２に記載のシステム。
【請求項１４】
前記第２のコンピューティングデバイスが、
前記第１のコンパイルされたコードが、前記１つ以上のネイティブライブラリ内で特定されないと判定されたときに、前記ユーザ入力を処理するための１つ以上の基本ブロックがキャッシュされるかどうかを判定し、
前記１つ以上の基本ブロックがキャッシュされていると判定されたとき、前記ユーザ入力を処理するための前記１つ以上の基本ブロックを実行するように構成されている、請求項１３に記載のシステム。
【請求項１５】
前記スキャナが、前記レガシーゲームの前記実行可能コードを前記レガシーゲームのデータから区別するように構成されており、前記スキャナが、前記データを前記デコーダに提供することなく、前記実行可能コードを前記デコーダに提供するように構成されている、請求項１２に記載のシステム。
【請求項１６】
前記データが、画像データまたは音声データを含む、請求項１５に記載のシステム。
【請求項１７】
前記方向性が、前記第１のコンパイルされたコードと前記第２のコンパイルされたコードの操作のシーケンスを提供する、請求項１２に記載のシステム。
【請求項１８】
前記第２のコンピューティングデバイスが、キャッシュを含み、前記ファイバが、前記キャッシュの１つ以上のレジスタの１つ以上の状態を格納し、前記第１のコンパイルされたコードから前記第２のコンパイルされたコードに切り替え、前記第２のコンパイルされたコードが実行された後、前記１つ以上のレジスタの前記１つ以上の状態を復元し、前記第２のコンパイルされたコードから前記第１のコンパイルされたコードに切り替えるように構成されている、請求項１２に記載のシステム。
【請求項１９】
前記第１のコンピューティングデバイスが、サーバシステム内に配置され、前記第２のコンピューティングデバイスが、ゲームコンソール内に配置されている、請求項１２に記載のシステム。
【請求項２０】
前記第１のコンピューティングデバイスが、ホストコンピュータであり、前記第２のコンピューティングデバイスが、ゲームコンソールである、請求項１２に記載のシステム。
【請求項２１】
レガシーゲームのプレイ中の事前（ＡｏＴ）コンパイル用プログラム命令を含むコンピュータ可読媒体であって、コンピュータシステムの１つ以上のプロセッサによる前記プログラム命令の実行が、
前記レガシーゲームの実行可能コードについてメモリデバイスをスキャンすることと、
前記実行可能コードを復号化することと、
前記実行可能コードを第１のソースコードに変換することと、
第１のコンパイルされたコードを出力するために前記第１のソースコードをコンパイルすることと、
前記レガシーゲームの前記プレイ中にユーザ入力が受信されたかどうかを判定することと、
前記ユーザ入力を処理するためのファイバが作成されているかどうかを判定することであって、前記ファイバが前記第１のコンパイルされたコードと第２のコンパイルされたコードとの間で実行の方向性を提供する、前記判定することと、
第１の処理ユニットによって、前記ユーザ入力を処理するための前記ファイバが作成さ

れたとの判定に応答して前記ファイバを実行することであって、前記ファイバが、前記レガシーゲームの１つ以上の画像フレームを生成するために実行される、前記実行することと、
を含む複数の操作を前記１つ以上のプロセッサに実行させる、コンピュータ可読媒体。
【請求項２２】
前記複数の操作が、
前記ファイバが生成されていないと判定することに応じて、前記第１のコンパイルされたコードが、１つ以上のネイティブライブラリ内に配置されているかどうかを判定することであって、前記第１のコンパイルされたコードが、前記ユーザ入力を処理するためのものである、前記判定することと、
前記１つ以上のネイティブライブラリ内で前記第１のコンパイルされたコードを特定した後に前記ファイバを作成することであって、前記ファイバを前記実行することが、前記ファイバを前記作成した後に行われている、前記作成することと、
をさらに含む、請求項２１に記載のコンピュータ可読媒体。
【発明の詳細な説明】
【技術分野】
【０００１】
本開示は、レガシーゲーム用事前（ＡｏＴ）コンパイラを提供するためのシステム及び方法に関する。
【背景技術】
【０００２】
電子ゲーム及びネットワーキング技術がより高度になるにつれて、それに応じてゲームの複雑性が増している。その結果、より複雑な筋書きと、ゲームプレイの目的、ミッション及びタスクと、ゲームプレイのアバターに関連する能力と、スコアリングと、が存在し得る。スコアリングは、様々な方法で発生及び重み付けされ得、同様に様々なカテゴリで、または個人またはチームベースで決定され得る。
【０００３】
前述の問題の重大性は、電子ゲームの複雑性が増すにつれてのみ増加する。そのため、一部のプレイヤーは、より複雑ではない古いゲームをプレイしたいと望むことがある。
【０００４】
この文脈において、本発明の実施形態が生じる。
【発明の概要】
【０００５】
本開示の実施形態は、レガシーゲーム用の事前（ＡｏＴ）コンパイラを提供するためのシステム及び方法を提供する。
【０００６】
一実施形態では、レガシーゲームのプレイ中にＡｏＴコンパイルするための方法が提供される。本方法は、レガシーゲームの実行可能コードのためにメモリデバイスをスキャンすることと、実行可能コードを復号化することと、実行可能コードを第１のソースコードに変換することと、を含む。本方法はさらに、第１のコンパイルされたコードを出力するために第１のソースコードをコンパイルすることと、レガシーゲームのプレイ中にユーザ入力が受信されるかどうかを判定することと、ユーザ入力を処理するためのファイバが作成されるかどうかを判定することと、を含む。ファイバは、第１のコンパイルされたコードと第２のコンパイルされたコードとの間で実行の方向性を提供する。本方法は、ユーザ入力を処理するためのファイバが作成されたと判定することに応答してファイバを実行することを含む。ファイバは、レガシーゲームの１つ以上の画像フレームを生成するために実行される。
【０００７】
一実施形態では、レガシーゲームのプレイ中にＡｏＴコンパイルするためのシステムが説明される。システムは、第１のコンピューティングデバイスを含む。第１のコンピュー

ティングデバイスは、レガシーゲームの実行可能コードのためのメモリデバイスをスキャンするスキャナを含む。第１のコンピューティングデバイスは、実行可能コードを復号化するデコーダと、実行可能コードを第１のソースコードに変換するトランスレータと、第１のコンパイルされたコードを出力するために第１のソースコードをコンパイルするコンパイラと、をさらに含む。システムは、第１のコンピューティングデバイスに結合された第２のコンピューティングデバイスをさらに含む。第２のコンピューティングデバイスは、レガシーゲームのプレイ中にユーザ入力を受信しているかどうかを判定し、ユーザ入力を処理するためのファイバが作成されているかどうかを判定する。ファイバは、第１のコンパイルされたコードと第２のコンパイルされたコードとの間で実行の方向性を提供する。第２のコンピューティングデバイスは、ユーザ入力を処理するためのファイバが作成されたと判定されたときにファイバを実行する。ファイバは、レガシーゲームの１つ以上の画像フレームを生成するために実行される。

【０００８】

レガシーゲームのレガシーコードを更新されたコードに変換するための本明細書に記載のシステム及び方法の利点のいくつかは、レガシーコードの機能が更新されたマシンによって実行されることを可能にすること含む。変換しないと、セキュリティ上の問題により、レガシーコードの機能が、更新されたマシンによって実行されることができない。例えば、更新されたマシンからレガシーコードを実行し、レガシーコードの実行時に生成されたデータを更新されたマシンのレジスタに書き込む権限はない。そのため、変換を提供することにより、更新されたマシンによるレガシーコードの機能の実行が容易になる。

【０００９】

本明細書に記載のシステム及び方法のさらなる利点には、実行時間を節約することが含まれる。一例として、レガシーコードのルーチンとサブルーチンまたは２つの同様の命令などの２つ以上の命令が、更新されたコードの１つの基本ブロックに結合される。そのため、更新されたコードの実行は、レガシーコードの実行と比較して高速である。

【００１０】

レガシーコードを更新されたコードに変換するための本明細書に記載のシステム及び方法のさらなる利点は、更新されたコードの１つ以上の基本ブロックが無効であると判定したときに、更新されたコードの１つ以上の追加の基本ブロックを再コンパイルすることを含む。例えば、１つ以上の基本ブロックが無効としてマークされているとき、１つ以上の基本ブロックが実際に無効であるかどうかが判定される。そのように判定すると、１つ以上の基本ブロックを実行する代わりに、１つ以上の追加の基本ブロックがコンパイル及び実行される。１つ以上の追加の基本ブロックは、１つ以上の基本ブロックが対応する同じゲームに対応する。

【００１１】

レガシーコードを更新されたコードに変換するための本明細書で説明するシステム及び方法の別のさらなる利点は、すべての基本ブロックが無効であるかどうかチェックされる必要がないことを含む。例えば、基本ブロックのコンパイル後に無効としてマークされた基本ブロックのみが有効かどうかがチェックされる。このことは、１つ以上の基本ブロックから仮想環境を表示する際のレイテンシを短縮する。また、すべての基本ブロックの無効性をチェックするために使用される処理能力は必要とされない。

【００１２】

また、レガシーコードを更新されたコードに変換するための本明細書に記載のシステム及び方法の利点には、ゲームの更新されたコードがすでにコンパイルされているときの処理時間及び処理能力の節約が含まれる。更新されたコードがサーバまたはゲームコンソールで生成されると、更新されたコードは再コンパイルされる必要がない。むしろ、更新されたコードが、サーバまたはゲームコンソールから別のゲームコンソールに転送されることができる。そのため、他のゲームコンソールで更新されたコードを再生成する際の処理時間と処理能力が節約される。

【００１３】

レガシーゲームのプレイ中にユーザ入力をエミュレーションするための本明細書に記載のシステム及び方法の追加の利点には、レガシーゲームをプレイするために更新されたハンドヘルドコントローラを使用することが含まれる。入力エミュレータは、更新されたハンドヘルドコントローラから受信したユーザ入力をレガシーハンドヘルドコントローラのユーザ入力に変換するために使用される。更新されたハンドヘルドコントローラから受け取ったユーザ入力の変換により、更新されたハンドヘルドコントローラを使用するレガシーゲームのプレイが容易になる。

【００１４】

レガシーゲームのプレイ中にレガシーグラフィックパラメータをエミュレーションするための本明細書に記載のシステム及び方法のさらなる利点には、レガシーゲームをプレイするためにレガシーグラフィックパラメータを更新されたグラフィカルパラメータに変換することが含まれる。グラフィックプロセッシングユニット（ＧＰＵ）エミュレータは、レガシーグラフィックパラメータを更新されたグラフィックパラメータに変換するために使用される。レガシーグラフィックパラメータの変換により、高解像度グラフィックを伴うレガシーゲームのプレイが容易になる。

【００１５】

また、レガシーゲームのプレイ中にレガシーオーディオパラメータをエミュレーションするための本明細書に記載のシステム及び方法の利点には、レガシーゲームをプレイするためにレガシーオーディオパラメータを更新されたオーディオパラメータに変換することが含まれる。オーディオ処理ユニット（ＡＰＵ）エミュレータは、レガシーオーディオパラメータを更新されたオーディオパラメータに変換するために使用される。レガシーオーディオパラメータの変換により、高品質サウンドを伴うレガシーゲームが容易になる。

【００１６】

レガシーゲーム用にＡｏＴコンパイラを提供するための本明細書に記載のシステム及び方法の利点には、レガシーゲームの実行中の時間を節約することが含まれる。ＡｏＴコンパイラは、レガシーゲームのプレイを開始するためのユーザ入力を受信する前に、事前に１つ以上のコンパイルされたコードを生成する。ユーザ入力が受信されると、コンパイルされたコードの実行の方向性を提供するファイバが作成され、コンパイルされたコードは方向性に従って実行される。

【００１７】

本明細書に記載のシステム及び方法のさらなる利点には、コンパイルされたコードが利用できないときに、１つ以上の基本ブロックを使用することが含まれる。コンパイルされたコードがユーザ入力を処理するために利用できない場合、ユーザ入力を処理するためのコードの１つ以上のブロックがキャッシュされているかどうかが判定される。コードの１つ以上のブロックが、次いで実行され、ユーザ入力を処理する。

【００１８】

レガシーゲーム用のＡｏＴコンパイラを提供するための本明細書に記載のシステム及び方法の追加の利点には、レガシーコードの機能が更新されたマシンによって実行され得るようにすることが含まれる。コンパイルされたコードがないと、セキュリティの問題により、レガシーコードの機能が更新されたマシンで実行され得ない。例えば、更新されたマシンからレガシーコードを実行し、レガシーコードの実行時に生成されたデータを更新されたマシンのレジスタに書き込む権限はない。そのため、コンパイルされたコードを提供することにより、更新されたマシンによるレガシーコードの機能の実行が容易になる。

【００１９】

本開示の他の態様は、本開示に記載された実施形態の原理を例示として示す、添付の図面と併せて取られた以下の発明を実施するための形態から明らかになるであろう。

【００２０】

本開示の様々な実施形態は、添付の図面と併せて以下の説明を参照することによって最もよく理解することができる。

【図面の簡単な説明】

【００２１】
【図１】更新されたコードの基本ブロックを生成するためのシステムの実施形態を示すブロック図である。
【図２】ゲームコードの基本ブロックをコンパイル及び実行するための方法の実施形態を示すフローチャートである。
【図３】基本ブロックをコンパイル及び実行するためのシステムの実施形態を示すブロック図である。
【図４Ａ】ゲームコンソール内の基本ブロックのコンパイルを説明するためのシステムの実施形態の図である。
【図４Ｂ】サーバシステム内の基本ブロックのコンパイルを説明するためのシステムの実施形態の図である。
【図５Ａ】基本ブロックの一実施形態を説明するための図である。
【図５Ｂ】基本ブロックコンパイラによって実行されるコンパイル操作の実施形態を示すための図である。
【図６Ａ】エミュレーションプロセッサシステムの構成要素を示すシステムの実施形態の図である。
【図６Ｂ】基本ブロックをコンパイル及び実行するための方法の実施形態を示すフローチャートである。
【図６Ｃ】異なるユーザ入力に対して、異なる基本ブロックが動的にコンパイルされることを示すシステムの実施形態の図である。
【図７Ａ】メモリデバイスからのエミュレートされた処理ユニット（ＰＵ）コードの削除を示すシステムの実施形態の図である。
【図７Ｂ】エミュレートされたＰＵコードをメモリデバイスから削除するための方法の実施形態を示すフローチャートである。
【図８Ａ】基本ブロックの検証を示すエミュレーションプロセッサシステムの実施形態の図である。
【図８Ｂ】図８Ａのエミュレーションプロセッサシステムによって実行される検証操作を示す方法の実施形態のフローチャートである。
【図８Ｃ】図８Ｂの方法のフローチャートの続きである。
【図９Ａ】レガシーマシンの実施形態を示す図である。
【図９Ｂ】更新されたマシンの実施形態を示す図である。
【図１０Ａ】基本ブロックコンパイラによる複数の基本ブロックの１つの基本ブロックへの結合を示すシステムの実施形態の図である。
【図１０Ｂ】１つ以上の基本ブロックの修正を示すシステムの実施形態の図である。
【図１０Ｃ】サブルーチンに基づいて作成された基本ブロックと、サブルーチンを呼び出すエミュレートされたＰＵコード命令に基づいて生成された基本ブロックとを組み合わせることを示すシステムの実施形態の図である。
【図１０Ｄ】２つの基本ブロックの間への基本ブロックの挿入を示すシステムの実施形態の図である。
【図１０Ｅ】基本ブロックの実行の順序でスイッチを示すシステムの実施形態の図である。
【図１１Ａ】基本ブロックｎに格納されたサイクル数のカウントの使用を説明するための方法の実施形態のフローチャートである。
【図１１Ｂ】図１１Ａの方法のフローチャートの続きである。
【図１２】第１のクライアントデバイスから第２のクライアントデバイスへの基本ブロックの転送を説明するためのシステムの実施形態の図である。
【図１３】レガシーゲームのプレイ中に更新されたハンドヘルドコントローラから受信したユーザ入力のエミュレーションを説明するためのシステムの実施形態の図である。
【図１４Ａ－１】デフォルトのマッピングから現在のマッピングへの変更を説明するためのシステムの実施形態の図である。

【図１４Ａ－２】デフォルトのマッピング及び現在のマッピングを説明するためのメモリデバイスの実施形態の図である。
【図１４Ｂ】レガシーゲームのプレイ中のデフォルトのマッピングの代わりに現在のマッピングの使用を説明するためのシステムの実施形態の図である。
【図１５】更新されたハンドヘルドコントローラの右サムスティックの動きに基づいたレガシーハンドヘルドコントローラの右移動ボタンの識別を説明するためのシステムの実施形態の図である。
【図１６Ａ】タッチパッドがユーザにより選択されたときの、十字線の仮想画像の生成を説明するためのシステムの実施形態の図である。
【図１６Ｂ】ユーザによるタッチパッド上での選択の場所と画像フレーム上の十字線の場所との間の対応を説明するためのシステムの実施形態の図である。
【図１７】異なるタイプのハンドヘルドコントローラを使用した異なるマッピングの使用を説明するためのシステムの実施形態の図である。
【図１８】ゲームコンソールから別のゲームコンソールへの現在のマッピングの通信を説明するためのシステムの実施形態の図である。
【図１９Ａ】更新されたハンドヘルドコントローラのタイプとサブタイプの受信を説明する画像のスクリーンショットである。
【図１９Ｂ】更新されたハンドヘルドコントローラのタイプを受信するためのメニューを説明するための画像のスクリーンショットである。
【図２０】レガシーハンドヘルドコントローラと更新されたハンドヘルドコントローラの入力デバイス間のマッピングを説明するための画像のスクリーンショットの実施形態である。
【図２１Ａ】ユーザからの更新されたグラフィックスパラメータの受信を説明するためのシステムの実施形態の図である。
【図２１Ｂ－１】１つ以上のレガシーグラフィックパラメータの１つ以上の更新されたグラフィックパラメータへのエミュレーションを説明するためのシステムの実施形態の図である。
【図２１Ｂ－２】ブロックディスパッチャから出力されたレガシーグラフィカルフレームからの更新されたグラフィカルフレームの生成を説明するためのシステムの実施形態の図である。
【図２１Ｃ－１】レガシー及び更新されたグラフィックパラメータのリストの実施形態、ならびにレガシー及び更新されたグラフィックパラメータ間の対応である。
【図２１Ｃ－２】レンダリング操作中に仮想オブジェクトの解像度を変更するためにグラフィックプロセッシングユニット（ＧＰＵ）エミュレータによって実行される方法の実施形態を説明するための図である。
【図２１Ｄ】更新されたグラフィックパラメータに基づいて画像フレームを生成するための方法の実施形態を説明するための図である。
【図２２Ａ】ユーザからの更新されたオーディオパラメータの受信を説明するためのシステムの実施形態の図である。
【図２２Ｂ－１】１つ以上のレガシーオーディオパラメータの１つ以上の更新されたオーディオパラメータへのエミュレーションを説明するためのシステムの実施形態の図である。
【図２２Ｂ－２】ブロックディスパッチャから出力されたレガシーオーディオフレームからの更新されたオーディオフレームの生成を説明するためのシステムの実施形態の図である。
【図２２Ｃ－１】レガシーオーディオパラメータに基づいて更新されたオーディオフレームを生成するための方法の実施形態を説明するための図である。
【図２２Ｃ－２】クライエントデバイスによるレガシーゲームのオーディオフレームの処理を説明するためのシステムの実施形態の図である。
【図２２Ｃ－３】レガシーゲームの複数の仮想シーンに対する複数の画像フレームに基づ

いた画像の表示と複数のオーディオフレームに基づいたサウンドの出力の同期を説明するシステムの実施形態の図である。
【図２２Ｄ】レガシーモード及び通常モードを表示するためのディスプレイデバイスの実施形態の図である。
【図２３】メニューが表示される仮想環境を表示するためのディスプレイデバイスの実施形態の図である。
【図２４Ａ】メニューが表示される仮想環境を表示するためのディスプレイデバイスの実施形態の図である。
【図２４Ｂ】複数のスキャンラインを有する仮想環境を表示するためのディスプレイデバイスの実施形態の図である。
【図２５Ａ】メニューが表示される仮想環境を表示するためのディスプレイデバイスの実施形態の図である。
【図２５Ｂ】メニューが表示される仮想環境を表示するためのディスプレイデバイスの実施形態の図である。
【図２５Ｃ】複数のメニューが表示される仮想環境を表示するためのディスプレイデバイスの実施形態の図である。
【図２５Ｄ】十字線が表示されている仮想環境を表示するためのディスプレイデバイスの実施形態の図である。
【図２６】事前（ＡｏＴ）プロセッサシステムの実施形態の図である。
【図２７】第１及び第２のコンパイルされたコードの生成を説明するためのシステムの実施形態の図である。
【図２８】ファイバを説明するためのシステムの実施形態の図である。
【図２９Ａ】レガシーゲームの実行可能機能を説明するためのレガシーゲームの実行可能コードの実施形態の図である。
【図２９Ｂ】データの実施形態の図であり、図２７のホストプロセッサによって実行可能ではないと識別されている。
【図３０Ａ】ソースコードの実施形態を示している。
【図３０Ｂ】ソースコード及び別のソースコードの実施形態を示している。
【図３１】ブロッククリエータとファイバクリエータとの間の結合を説明するためのシステムの実施形態の図である。
【図３２Ａ】図３１のブロッククリエータ及びファイバクリエータによって実行される方法の実施形態のフローチャートである。
【図３２Ｂ】図３２Ａの方法の続きである。
【図３２Ｃ】図３２Ｂの方法の続きである。
【図３３】ＡｏＴエグゼキュータとＧＰＵエミュレータとの間の結合を説明するためのシステムの実施形態の図である。
【図３４】コンパイルされたコードジェネレータ及びＡｏＴエグゼキュータの場所を示すためのシステムの実施形態の図である。
【図３５】サーバシステム内のＡｏＴプロセッサシステムの場所を説明するためのシステムの実施形態の図である。
【図３６】本開示の実施態様に従って、クラウドビデオゲームをクライアントデバイスにストリーミングするために実行される様々な操作を概念的に示すフロー図である。
【図３７】クライアントデバイスのディスプレイデバイスとのインターフェースと互換性があり、コンピュータネットワークを介してゲームホスティングシステムと通信することができるゲームコンソールの実施形態のブロック図である。
【図３８】ヘッドマウントディスプレイ（ＨＭＤ）の構成要素を示す図である。
【図３９】情報サービスプロバイダ（ＩＮＳＰ）アーキテクチャの実施形態を示している。
【発明を実施するための形態】
【００２２】

レガシーゲーム用の事前（ＡｏＴ）コンパイラを提供するためのシステム及び方法が説明されている。本開示の様々な実施形態は、これらの具体的な詳細の一部または全部を伴わずに実施されることに留意されたい。他の事例では、本開示の様々な実施形態を不必要に不明瞭にしないために、周知のプロセス操作は詳細に説明されていない。

【００２３】

図１は、更新されたコードの基本ブロック１～ｎまでを生成するためのシステム１００の実施形態を示すブロック図であり、ここで、ｎは正の整数である。システム１００は、キャッシュ１０２と、基本ブロックコンパイラ１０４と、ゲームタイトルＧＮを有するレガシーゲームＮのエミュレートされた処理ユニット（ＰＵ）コード１０６と、を含み、ここで、Ｎは正の整数である。一例として、本明細書で使用されるキャッシュは、データを格納するハードウェアまたはソフトウェアの構成要素であり、そのため、データに対する将来の要求がより高速に処理され得る。キャッシュヒットは、要求されたデータがキャッシュ内で見つけられ得るときに発生し、一方でキャッシュミスは、見つけられ得ないときに発生する。キャッシュヒットは、キャッシュからデータを読み取ることによって処理され、これは、結果を再計算することや、またはメモリデバイスなどのより低速のデータストアから読み取ることよりも高速であり、そのため、キャッシュから処理できるリクエストが多いほど、システムがより高速に実行する。説明すると、キャッシュはレジスタのグループであり、これは、メインメモリデバイスと比較して、１０倍～１００倍など、より高速にアクセスされ得る。

【００２４】

一例として、キャッシュは、メインメモリデバイスよりも少ないメモリアドレスの数を有している。この例では、プロセッサは、操作で使用されるデータがキャッシュ内のメモリアドレスに格納されているかどうかを最初に判定し、そうでない場合は、プロセッサがメインメモリデバイスのメモリアドレスにアクセスしてデータを探す。

【００２５】

一例として、本明細書で使用される基本ブロックコンパイラは、エミュレートされたＰＵコード１０６コードを、更新されたコードの例であるゲームコードＧＣＮに変換するコンピュータプログラムである。ゲームコードＧＣＮは、レガシーゲームＮの機能を表す。コンピュータプログラムは、エミュレーションプロセッサシステムの１つ以上のプロセッサによって実行される。ゲームコードＧＣＮは、本明細書では中間コードと称される。一例として、中間コードは、ソースコードでもマシンコードでもない。説明すると、中間コードには、中央処理装置（ＣＰＵ）のアーキテクチャ、またはグラフィック処理装置（ＧＰＵ）のアーキテクチャ、または更新されたマシンのオーディオ処理装置（ＡＰＵ）のアーキテクチャに固有ではない基本ブロックが含まれており、その例が以下に提供されている。例示では、中間コードは、更新されたマシンのＣＰＵまたはＧＰＵで実行され得る基本ブロックを含み、その例には、ゲームコンソール、デスクトップコンピュータ、ラップトップコンピュータ、タブレット、スマートフォン、スマートテレビが含まれる。一例として、ソースコードは、プレーンテキストであり得る、人間が可読なプログラミング言語を使用して記述されている。一例として、本明細書で使用される基本ブロックコンパイラは、ハードウェアもしくはソフトウェア、またはそれらの組み合わせを使用して実装される。説明すると、基本ブロックコンパイラの機能は、コントローラもしくはプログラマブルロジックデバイス（ＰＬＤ）または特定用途向け集積回路（ＡＳＩＣ）を使用して実装される。

【００２６】

本明細書で使用されるコントローラの例には、プロセッサ及びメモリデバイスが含まれる。プロセッサはメモリデバイスに結合されている。本明細書で使用される場合、一例として、プロセッサは、マイクロプロセッサ、またはＣＰＵ、またはＧＰＵ、またはマイクロコントローラ、またはＡＳＩＣ、またはＰＬＤである。本明細書で使用されるメモリデバイスの例には、ランダムアクセスメモリ（ＲＡＭ）及び読み取り専用メモリ（ＲＯＭ）が含まれる。説明すると、メモリデバイスは、フラッシュメモリデバイス、またはハード

ディスク、またはソリッドステートストレージデバイス、または独立ディスクの冗長アレイ（ＲＡＩＤ）、またはそれらの組み合わせである。

【００２７】

エミュレートされたＰＵコード１０６の例は、マシンコードであり、これは操作を実行するようにレガシーマシンのＣＰＵまたはＧＰＵまたはＡＰＵなどのプロセッサに指示する。例えば、エミュレートされたＰＵコード１０６は、キャッシュ１０２内に格納されたデータに対してロード、ストア、ジャンプ、または算術論理操作ユニット（ＡＬＵ）操作などの特定の操作を実行するようにレガシーマシンのＣＰＵに指示する命令のシーケンスを含む。別の例として、エミュレートされたＰＵコード１０６は、一連の１と０を含むバイナリコードである。別の例として、エミュレートされたＰＵコード１０６は、一連の命令を含み、これは、レガシーマシンのＧＰＵが、キャッシュ１０２内に格納されたデータに対してロード、ストア、ジャンプ、またはＡＬＵ操作などの特定の操作を実行するように指示する。レガシーマシンのＧＰＵは、仮想オブジェクトに対して操作を実行し、色、強度、陰影、テクスチャ、またはそれらの組み合わせなどのパラメータを仮想オブジェクトに割り当てる。仮想オブジェクトの例には、仮想車両、仮想動物、仮想ゲームキャラクタ、仮想静止オブジェクト、仮想建物などが含まれる。色、強度、陰影、テクスチャ、またはそれらの組み合わせは、レガシーグラフィックパラメータの例であることに留意されたい。

【００２８】

さらに別の例として、エミュレートされたＰＵコード１０６は、一連の命令を含み、これは、レガシーマシンのＡＰＵに、残響量（レガシーゲームＮの仮想シーンの表示に同期する出力である）であるリバーブなどの１つ以上のレガシーオーディオパラメータを生成するように指示する。１つ以上のレガシーオーディオパラメータの別の例には、どのサウンドがレガシーゲームＮの仮想シーンと同期する出力であり得るかに基づいたサンプリングレートが含まれる。１つ以上のレガシーオーディオパラメータのさらに別の例には、サウンドが、レガシーゲームＮの仮想オブジェクトと同期して、ステレオのみとして出力されることが含まれる。

【００２９】

エミュレートされたＰＵコード１０６は、レガシーマシンのＣＰＵまたはＧＰＵのアーキテクチャに特定または独自のものである。例えば、エミュレートされたＰＵコード１０６は、更新されたマシンのＣＰＵまたはＧＰＵによって実行されることができない。別の例として、エミュレートされたＰＵコード１０６は、レガシーマシンでは実行可能であるが、更新されたマシンでは実行できない。

【００３０】

一例として、エミュレートされたＰＵコード１０６は、レガシーゲームＮで操作を実行するためにレガシーマシンのプロセッサによって実行される。説明すると、エミュレートされたＰＵコード１０６は、レガシーマシン上でレガシーゲームをプレイするために実行される。レガシーゲームの例には、戦闘機ゲーム、ダンスレッスンゲーム、アドベンチャゲーム、アクションゲームなどのビデオゲームが含まれる。説明すると、レガシーゲームはゲーム開発者によって更新されていない。さらに説明するために、レガシーゲームの現在のバージョンはない。別のさらなる説明として、レガシーゲームは、現在のゲームの作成の何年も前に作成されたものである。

【００３１】

基本ブロックコンパイラ１０４は、レガシーゲームＮのエミュレートされたＰＵコード１０６にアクセスし、エミュレートされたＰＵコード１０６を、基本ブロック１～ｎなどの１つ以上の基本ブロックに変換する。一例として、各基本ブロック１～ｎは、ある基本ブロックを別の基本ブロックと区別するための開始識別子と終了識別子を有する。基本ブロックコンパイラ１０４は、基本ブロック１～ｎをキャッシュ１０２に格納する。基本ブロック１～ｎが実行されると、レガシーゲームＮがエミュレートされる。

【００３２】

図２は、ゲームコードＧＣＮの基本ブロックをコンパイル及びディスパッチするための方法２００の実施形態を示すフローチャートである。方法２００は、更新されたマシンの１つ以上のプロセッサによって実行される。方法２００は、キャッシュヒットがあるかどうか、例えば、基本ブロックがキャッシュ１０２に格納されているかどうかを判定する操作２０２を含む。例えば、操作２０２は、レガシーゲームのプレイ中のユーザ入力が受信されたときに実行またはトリガーされる。説明すると、ユーザ入力は、レガシーゲームＮの仮想オブジェクトの位置もしくは方向、またはそれらの組み合わせを変更するために受信されている。図示では、ユーザ入力が受信されたと判定されると、基本ブロック１～ｎのうちの１つのような基本ブロックが、キャッシュ１０２に格納されているかどうかが判定されている（図１）。図示では、仮想オブジェクトの位置もしくは方向、またはそれらの組み合わせを変更するために基本ブロックが実行されている。

【００３３】

別の説明として、操作２０２は、レガシーゲームの仮想オブジェクトのルックアンドフィールなどのパラメータを変更するためのユーザ入力が受信されたときに実行される。図示では、ユーザ入力が受信されたと判定すると、基本ブロック１～ｎのうちの１つなどの基本ブロックがキャッシュ１０２に格納されているかどうかが判定される。図示では、仮想オブジェクトのパラメータを生成または変更するために基本ブロックが実行されている。ルックアンドフィールは、レガシーグラフィカルパラメータの一例であることに留意されたい。

【００３４】

さらに別の説明として、操作２０２は、どのサウンドが出力されるかに基づいて、レガシーオーディオパラメータを変更するためにユーザが入力したときに実行される。図示では、ユーザ入力が受信されると、基本ブロック１～ｎのうちの１つなどの基本ブロックがキャッシュ１０２に格納されているかどうかが判定される。図示では、基本ブロックは、レガシーゲームＮの仮想シーンの表示を伴うサウンドとして出力されるレガシーオーディオパラメータを生成または変更するために実行されるものである。

【００３５】

基本ブロックがキャッシュされていると判定することに応じて、方法２００の操作２０４では、基本ブロックがディスパッチされる。例えば、基本ブロックは、操作２０４で実行されるか、または動作する。説明すると、基本ブロックは、更新されたマシンのＣＰＵによって実行され、仮想オブジェクトをある位置から別の位置に、もしくはある方向から別の方向に、またはそれらの組み合わせで動かす。別の説明として、基本ブロックは、パラメータを生成してパラメータを仮想オブジェクトの一部に割り当てるために更新されたマシンのＧＰＵによって実行される。仮想オブジェクトの一部分の一例には、仮想オブジェクトのピクセル、または仮想オブジェクトの三角形の部分、または仮想オブジェクトの事前規定された形状の一部分が含まれる。説明すると、仮想オブジェクトは、事前に決定された数のピクセルに分割され、各ピクセルにはパラメータの値が割り当てられる。別の例として、基本ブロックは、更新されたマシンのＡＰＵによって実行され、レガシーオーディオパラメータを生成し、レガシーオーディオパラメータをレガシーゲームＮの仮想シーンを伴うサウンドとして出力する。

【００３６】

基本ブロックがキャッシュされていないとの判定に応じて、基本ブロックをコンパイルする操作２０６が方法２００で実行される。操作２０６は、基本ブロックコンパイラ１０４（図１）によって実行される。例えば、基本ブロックコンパイラ１０４は、レガシーゲームＮのエミュレートされたＰＵコード１０６（図１）を解析して、操作２０２の前に、レガシーゲームのプレイ中に受信したユーザ入力を処理するための機能を含むエミュレートされたＰＵコード命令を識別する。操作２０２の前に受信されたユーザ入力は、操作２０２をトリガーする。説明すると、基本ブロックコンパイラ１０４は、エミュレートされたＰＵコード１０６の各エミュレートされたコードＰＵ命令をトラバースして、エミュレートされたコードＰＵコード命令が、レガシーゲームのプレイ中に受信したユーザ入力を

満たす（例えば、応答を生成する）操作（機能など）を含むかどうかを判定する。図示では、機能が識別されると、基本ブロックコンパイラ１０４は、エミュレートされたコードのＰＵコード命令を変換して、基本ブロックを生成する。図示では、受信したユーザ入力に応答して、操作２０２の前に、ユーザ入力を処理するために必要とされないレガシーゲームの他のエミュレートされたコードＰＵコード命令は、基本ブロックコンパイラ１０４によって基本ブロックにコンパイルされない。

【００３７】

方法２００の操作２０８において、操作２０６で生成された基本ブロックは、基本ブロックコンパイラ１０４によってキャッシュ１０２に格納される。次に、キャッシュされた基本ブロックは、操作２０４で実行され、操作２０２の前に受信したユーザ入力を処理する。

【００３８】

一実施形態では、方法２００は、サーバシステムの１つ以上のプロセッサによって実行される。一例として、サーバシステムは、更新されたマシンをサーバとして含む。説明すると、各サーバブレードは、ゲームコンソールなどの更新されたマシンである。

【００３９】

一実施形態では、方法２００は、ユーザ入力が受信されるまで実行されない。例えば、基本ブロック１～ｎのうちの１つなどの基本ブロックが、キャッシュ１０２に格納されているかどうかの判定はされず、基本ブロックのコンパイルも、ユーザ入力が受信されるまで基本ブロックの実行もされない。

【００４０】

一実施形態では、基本ブロックコンパイラ１０４は、レガシーゲームＮのプレイ中の第１のユーザ入力に応答して、基本ブロック１～ｎのうちの１つ以上をコンパイルし、レガシーゲームＮのプレイ中の第２のユーザ入力に応答して、基本ブロック１～ｎのうちの残りの１つ以上をコンパイルする。説明すると、基本ブロックコンパイラ１０４は、第１のユーザ入力を処理するために基本ブロック１及び２を生成し、第２のユーザ入力を処理するために基本ブロック３～７を生成する。第２のユーザ入力は、第１のユーザ入力の後に受信される。

【００４１】

図３は、基本ブロック１～ｎなどの基本ブロックをコンパイル及びディスパッチするためのシステム３０４の実施形態を説明するためのブロック図である。システム３００は、基本ブロックコンパイラ１０４、キャッシュ１０２、及びブロックディスパッチャ３０２を含む。一例として、ブロックディスパッチャ３０２は、ハードウェアもしくはソフトウェア、またはそれらの組み合わせであり、ユーザ入力を処理するために基本ブロック１～ｎのうちの１つ以上の操作を実行する。説明すると、ブロックディスパッチャ３０２は、ＰＬＤまたはＡＳＩＣまたはコントローラである。別の説明として、ブロックディスパッチャ３０２は、コンピュータソフトウェアである。一例として、ブロックディスパッチャ３０２は、更新されたマシンのＧＰＵまたはＣＰＵである。

【００４２】

基本ブロックコンパイラ１０４は、操作コード（オペコード）などのエミュレートされたＰＵコード１０６（図１）の一部分を復号化し、その部分を、更新されたマシンのＣＰＵまたはＧＰＵまたはＡＰＵなどの、プロセッシングユニットのための中間表現に変換する。例えば、基本ブロックコンパイラ１０４は、１つ以上の命令など、エミュレートされたＰＵコード１０６のＣＰＵコードの部分を解析して、ＣＰＵコードの一部分がユーザ入力を処理するための機能を含むかどうかを判定する。そのように判定すると、基本ブロックコンパイラ１０４は、ＣＰＵコードの一部分を、基本ブロック１～ｎなどの１つ以上の基本ブロックに変換する。別の例として、基本ブロックコンパイラ１０４は、１つ以上の命令などのエミュレートされたＰＵコード１０６のＧＰＵコードの一部分を解析して、ＧＰＵコードの一部分が、ユーザ入力を処理する機能を含むかどうかを判定する。そのように判定すると、基本ブロックコンパイラ１０４は、ＧＰＵコードの一部分を、基本ブロッ

ク１～ｎなどの１つ以上の基本ブロックに変換する。さらに別の例として、基本ブロックコンパイラ１０４は、１つ以上の命令などのエミュレートされたＰＵコード１０６のＡＰＵコードの一部分を解析して、ＡＰＵコードの一部分が、ユーザ入力を処理する機能を含むかどうかを判定する。そのように判定すると、基本ブロックコンパイラ１０４は、ＡＰＵコードの一部分を、基本ブロック１～ｎなどの１つ以上の基本ブロックに変換する。

【００４３】

また、基本ブロックコンパイラ１０４は、エミュレートされたＰＵコード１０６の一部分から生成された各基本ブロックの実行のサイクル数を推定して、推定カウントを生成する。例えば、基本ブロックコンパイラ１０４は、基本ブロック１がジャンプ動作を含み、ジャンプ動作が所定の時間を要すると判定する。基本ブロックコンパイラ１０４は、基本ブロック１のジャンプ動作が所定の時間を要すると推定する。ブロックコンパイラ１０４は、推定されたカウントをキャッシュ１０２に格納する。例えば、ブロックコンパイラ１０４は、サイクル数が推定される推定されたカウントを基本ブロックｎに格納する。

【００４４】

基本ブロックがコンパイルされると、それらは高速ルックアップのためにキャッシュ１０２に格納される。例えば、どの基本ブロックがコンパイルされているかに応答してユーザ入力を受信した後に別のユーザ入力を受信し、同じ基本ブロックが、他のユーザ入力の処理に使用できるとき、基本ブロックは、キャッシュ１０２から迅速にアクセスすることができ、再生成される必要がない。

【００４５】

さらに、キャッシュ１０２に格納された１つ以上の基本ブロックは、コンパイル後に無効としてマークされることができる。無効としてマークされた１つ以上の基本ブロックは、その後、基本ブロックの実行中に有効化されるか、または無効化される。１つ以上の基本ブロックが無効化されるとき、１つ以上の追加の基本ブロックがコンパイルされる。１つ以上の追加の基本ブロックのコンパイルは、本明細書では、１つ以上の基本ブロックの再コンパイルと呼ばれることがある。

【００４６】

１つ以上の追加の基本ブロックの各々は、１つ以上の基本ブロック１～ｎの構造と同じ構造を有する。例えば、１つ以上の追加の基本ブロックの各々は、ソースレジスタアドレス、宛先レジスタアドレス、及び操作がある。別の例として、１つ以上の追加の基本ブロックの各々は、ソースレジスタアドレス、宛先レジスタアドレス、操作、及び追加の基本ブロックの操作の実行のサイクル数を有する。この例では、１つ以上の追加の基本ブロックのいくつかは、無効なマークを含む。さらに別の例として、１つ以上の追加の基本ブロックの各々は、ソースレジスタアドレス、宛先レジスタアドレス、操作、及び追加の基本ブロックの操作の実行のサイクル数を有する。コードの追加ブロックの各々は、基本ブロック１～ｎの各々と同じ方法で実行されることに留意されたい。

【００４７】

ブロックディスパッチャ３０２は、ユーザ入力に基づいて、基本ブロック１～ｎのうちの１つ以上を実行または動作する。例えば、ブロックディスパッチャ３０２は、基本ブロック１及び２を実行して第１のユーザ入力を処理し、第２のユーザ入力に応答して基本ブロック３～７を実行する。一例として、ブロックディスパッチャ３０２は、デジタルクロック発振器またはクロックジェネレータなどのクロックソースを含み、これは、実際のカウントを生成するためのユーザ入力に基づいて、１つ以上の基本ブロック１～ｎを実行するために使用されるサイクル数をカウントする。ブロックディスパッチャ３０２は、実際のカウントをブロックコンパイラ１０４に送信して、推定されたカウントを実際のカウントで更新する。例えば、実際のカウントは、実際のカウントが計算される基本ブロックｎに格納される。説明すると、実際のカウントは、基本ブロックｎに割り当てられたキャッシュ１０２の１つ以上のメモリレジスタに格納される。

【００４８】

一実施形態では、基本ブロックコンパイラ１０４は、任意の基本ブロックの実行のため

のサイクル数を推定しない。この実施形態では、推定されたカウントの実際のカウントへの置き換えはない。むしろ、この実施形態では、実際のカウントは、ブロックコンパイラ１０４によって、実際にカウントが判定される基本ブロックｎに格納される。

【００４９】

図４Ａは、ゲームコンソール４０２内での基本ブロック１～ｎのコンパイルを説明するためのシステム４００の実施形態の図である。システム４００は、ゲームコンソール４０２、サーバシステム４０４、コンピュータネットワーク４０８、及びディスプレイデバイス４１０を含む。サーバシステム４０４は、１つ以上のサーバを含む。一例として、サーバシステム４０４は、データセンタのハウジング内に配置されている。サーバシステム４０４は、エミュレートされたＰＵコード１０４などのエミュレートされたＰＵコードを格納するメモリデバイス４１２を含む。例えば、メモリデバイス４１２は、ゲームコード１（ｇｃ１）、ゲームコード２（ｇｃ２）及びその後のゲームコードＮ（ｇｃＮ）までを格納する。ゲームコードｇｃＮは、エミュレートされたＰＵコード１０６の例である（図１）。ゲームコード１～Ｎの各々は、レガシーゲームのレガシーコードである。説明すると、ゲームコードｇｃ１は、第１のレガシーゲームをプレイするためのマシンコードであり、ゲームコードｇｃ２は、第２のレガシーゲームをプレイするためのマシンコードである。第２のレガシーゲームは、第１のレガシーゲームとは異なっている。一例として、メモリデバイス４１２は、レガシーマシンのメモリデバイスであることに留意されたい。

【００５０】

一例として、ゲームコードｇｃ１～ｇｃＮのうちどれも、更新されたマシンで実行されることができず、かつ、レガシーマシンで実行されることができない。説明すると、更新されたマシンのＣＰＵまたはオペレーティングシステムは、ゲームコードｇｃ１～ｇｃＮの実行をサポートできない。その一方で、レガシーマシンのＣＰＵまたはオペレーティングシステムは、ゲームコードｇｃ１～ｇｃＮの実行をサポートする。本明細書で使用されるコンピュータネットワークの例には、インターネットなどのワイドエリアネットワーク（ＷＡＮ）、またはイントラネットなどのローカルエリアネットワーク（ＬＡＮ）、またはそれらの組み合わせが含まれる。

【００５１】

ゲームコンソール４０２は、更新されたマシンの一例である。ディスプレイデバイス４１０の例には、テレビ、スマートテレビ、及びコンピュータモニタが含まれる。説明すると、ディスプレイデバイス４１０は、液晶ディスプレイ（ＬＣＤ）デバイス、または発光ダイオード（ＬＥＤ）ディスプレイデバイス、または有機発光ダイオード（ＯＬＥＤ）ディスプレイデバイスである。

【００５２】

システム４００は、ユーザ１の片手または両手で保持されるハンドヘルドコントローラ（ＨＨＣ）４１４をさらに含む。本明細書で使用されるハンドヘルドコントローラの例には、ボタンを備えたコントローラ、Ｓｏｎｙ（登録商標）ＣｏｒｐｏｒａｔｉｏｎのＭｏｖｅ（登録商標）コントローラ、及銃型コントローラが含まれる。ハンドヘルドコントローラのボタンの例には、ジョイスティック、ディスプレイデバイス４１０上で仮想オブジェクトを上、下、左、または右に移動するためのボタン、及びゲームタイトルＧＮを有するレガシーゲームＮの様々な機能を選択するための他のボタンが含まれる。

【００５３】

ゲームコンソール４０２は、メモリデバイス４０６、入力プロセッサシステム４０７、及びエミュレーションプロセッサシステム４０９を含む。一例として、本明細書で使用されるプロセッサシステムは、互いに結合された１つ以上のプロセッサを含む。エミュレーションプロセッサシステム４０９は、メモリデバイス４０６に結合され、入力プロセッサシステム４０７は、エミュレーションプロセッサシステム４０９に結合される。エミュレーションプロセッサシステム４０９は、基本ブロックコンパイラ１０４及びキャッシュ１０２を含む。基本ブロックコンパイラ１０４は、キャッシュ１０２に結合されている。

【００５４】

ゲームコンソール４０２は、高精細メディアインターフェース（ＨＤＭＩ（登録商標））ケーブルなどの有線通信媒体または無線接続を介してディスプレイデバイス４１０に結合される。本明細書で使用される無線接続の例は、Ｗｉ－Ｆｉ（登録商標）接続またはＢｌｕｅｔｏｏｔｈ（登録商標）接続を含む。また、ハンドヘルドコントローラ４１４は、有線接続または無線接続を介してゲームコンソール４０２に結合されている。本明細書で使用される有線接続の例は、シリアル転送ケーブル、パラレル転送ケーブル、及びユニバーサルシリアルバス（ＵＳＢ）ケーブルを含む。

【００５５】

クライアントデバイスの一例は、ハンドヘルドコントローラ、ゲームコンソール、及びディスプレイデバイスの組み合わせを含む。クライアントデバイスの別の例は、ハンドヘルドコントローラとディスプレイデバイスの組み合わせを含む。クライアントデバイスのさらに別の例は、スマートテレビまたはヘッドマウントディスプレイなどのディスプレイデバイスを含む。クライアントデバイスの別の例は、コンピュータまたはスマートフォンを含む。

【００５６】

ユーザ１は、ユーザ識別（ＩＤ）及びパスワードがサーバシステム４０４によって認証されると、自身のユーザアカウント１にログインする。ユーザアカウント１には、サーバシステム１によってユーザＩＤ１が割り当てられる。ユーザ１がユーザアカウント１にログインすると、ユーザ１は、ゲームタイトルＧ１、ゲームタイトルＧａ、ゲームタイトルＧ２、そしてそのように続いてゲームタイトルＧＮまで、などの複数のゲームタイトルにアクセスすることができる。ゲームタイトルＧ１、Ｇ２、そしてそのように続くゲームタイトルＧＮまでは、レガシーゲームのタイトルの例である。ゲームタイトルＧａは、レガシーゲームではないゲームのタイトルである。むしろ、ゲームタイトルＧａは、レガシーマシンではプレイできないＦｏｒｔｎｉｔｅ（登録商標）など、現在のゲームのものである。

【００５７】

ユーザアカウント１にログインした後、ユーザ１は、レガシーゲームＮをプレイするためのゲームタイトルＧＮを選択するために、ハンドヘルドコントローラ４１４上の１つ以上のボタンを選択する。ユーザ１がゲームタイトルＧＮを選択すると選択を示すユーザ入力４１８が、ハンドヘルドコントローラ４１４から、ゲームコンソール４０２及びコンピュータネットワーク４０８を介してサーバシステム４０４に送信される。一例として、ユーザ入力は入力信号である。ゲームタイトルＧＮの選択を示すユーザ入力４１８を受信すると、サーバシステム４０４は、ユーザ入力４１８に基づいてゲームコードｇｃＮを識別する。例えば、サーバシステム４０４は、ゲームコードｇｃＮがゲームタイトルＧＮと同じゲームタイトルを有し、その選択がユーザ入力４１８に示されていることを識別する。

【００５８】

サーバシステム４０４は、コンピュータネットワーク４０８を介してゲームコードｇｃＮをゲームコンソール４０２に送信する。ゲームコードｇｃＮを受信すると、エミュレーションプロセッサシステム４０９は、ゲームコードｇｃＮをゲームコンソール４０２のメモリデバイス４０６に格納する。

【００５９】

ゲームコードｇｃＮを有するレガシーゲームＮのプレイ中に、ユーザ入力４１９が、無線接続を介してハンドヘルドコントローラ４１４から受信されると、入力プロセッサシステム４０７は、ユーザ入力４１９をユーザ入力４２０に変換し、ユーザ入力４２０をエミュレーションプロセッサシステム４０９に提供する。一例として、ユーザ入力４１９は、レガシーゲームＮのプレイ中に戦闘機を位置Ｐ１から位置Ｐ２に、またはある方向から別の方向に移動する要求を含む。戦闘機は仮想オブジェクトの一例である。ユーザ入力４２０を受信すると、エミュレーションプロセッサシステム４０９は、基本ブロックコンパイラ１０４を実行して、メモリデバイス４０６に格納されたゲームコードｇｃＮの一部分からゲームコードＧＣＮの一部分を生成する。ゲームコードＧＣＮの一部分は、ユーザ入力

４２０に基づいて生成される。例えば、ユーザ入力４２０が、レガシーゲームＮのプレイ中に位置Ｐ１から位置Ｐ２に戦闘機を移動する要求を含むとき、基本ブロックコンパイラ１０４は、ゲームコードｇｃＮを解析して、位置Ｐ１から位置Ｐ２までを計算する命令を特定する。基本ブロックコンパイラ１０４は、命令をゲームコードＧＣＮの基本ブロックに変換し、基本ブロックが、次いで実行され、戦闘機の位置をＰ１からＰ２に変更する。さらに、本例では、位置がＰ１からＰ２に変化するとき、戦闘機が位置Ｐ１にある仮想環境が、戦闘機が位置Ｐ２にある仮想環境に変更される。本例では、ゲームコードＧＣＮの基本ブロックは、エミュレーションプロセッサシステム４０９のＧＰＵによって実行されて、画像フレーム４２２を生成する。説明すると、画像フレーム４２２は、位置Ｐ２にある戦闘機を有する仮想環境を表示するために、ディスプレイデバイス４１０上に表示される。このようにして、ゲームコードＧＣＮの大部分または全体は、基本ブロックコンパイラ１０４によってコンパイルされ、実行のためにキャッシュ１０２に格納される。一例として、仮想シーンなどの仮想環境は、１つ以上の仮想現実（ＶＲ）画像または１つ以上の拡張現実（ＡＲ）画像を含む。

【００６０】

一実施形態では、サーバシステム４０４とゲームコンソール４０２との間のデータの通信は、インターネットプロトコル（ＴＣＰ／ＩＰ）を介した伝送制御プロトコルなどのネットワーク通信プロトコルを介して行われる。例えば、サーバシステム４０４は、データをパケットに変換するためのネットワークインターフェースコントローラ（ＮＩＣ）を含む。サーバシステム４０４のネットワークインターフェースコントローラは、メモリデバイス４１２に結合されて、メモリデバイス４１２からデータを受信する。メモリデバイス４１２からデータを受信すると、サーバシステム４０４のネットワークインターフェースコントローラは、ネットワーク通信プロトコルをデータに適用することによって、１つ以上のパケット内にデータを埋め込む。１つ以上のパケットは、サーバシステム４０４のネットワークインターフェースコントローラから、コンピュータネットワーク４０８を介してゲームコンソール４０２に転送される。ゲームコンソール４０２のネットワークインターフェースコントローラは、ネットワーク通信プロトコルを適用することによって、１つ以上のパケットからデータを抽出する。ゲームコンソール４０２のネットワークインターフェースコントローラは、エミュレーションプロセッサシステム４０９に結合されている。ゲームコンソール４０２のネットワークインターフェースコントローラは、コンピュータネットワーク４０８から受信したデータをエミュレーションプロセッサシステム４０９に提供する。さらに、ゲームコンソール４０２のネットワークインターフェースコントローラは、エミュレーションプロセッサシステム４０９からデータを受信し、ネットワーク通信プロトコルを適用することによって１つ以上のパケット内にデータを埋め込み、コンピュータネットワーク４０８を介して１つ以上のパケットをサーバシステム４０４に送信する。サーバシステム４０４のネットワークインターフェースコントローラは、ネットワーク通信プロトコルを、コンピュータネットワーク４０８から受信した１つ以上のパケットに適用して、１つ以上のパケットからデータを抽出し、格納のためにデータをメモリデバイス４１２に送信する。ネットワークインターフェースコントローラの例は、ネットワークインターフェースカード、イーサネット（登録商標）カード、及びネットワークアダプタを含む。説明すると、ゲームコンソール４０２のネットワークインターフェースコントローラは、入力プロセッサシステム４０７がコンピュータネットワーク４０８と通信することを可能にする。

【００６１】

一実施形態では、コンピュータネットワーク４０８に加えて、またはその代わりに、サーバシステム４０４とゲームコンソール４０２との間でデータを通信するためにセルラーネットワークが使用される。例えば、サーバシステム４０４とゲームコンソール４０２との間の通信は、無線技術を使用して促進される。無線技術は、例えば、４Ｇまたは５Ｇ無線通信技術を含む。本明細書で使用されるように、５Ｇはセルラーネットワーク技術の第５世代である。また、５Ｇネットワークは、デジタルセルラーネットワークであり、そこ

では、プロバイダによって網羅されるサービスエリアが、セルと呼ばれる小さな地理的エリアに分割されている。５Ｇ無線通信技術では、サウンドと画像を表すアナログ信号が電話でデジタル化され、アナログ－デジタルコンバータによって変換され、ビットのストリームとして送信される。セル内のすべての５Ｇ無線デバイスは、他のセルで再利用される周波数のプールからトランシーバによって割り当てられた周波数チャネルを介して、セル内のローカルアンテナアレイ及び低電力自動トランシーバ（送信機及び受信機）の電波によって通信する。ローカルアンテナは、高帯域幅の光ファイバまたは無線バックホール接続によってセルラーネットワークで接続される。他のセルネットワークと同様に、１つのセルから別のセルに移動するモバイルデバイスは、新しいセルに自動的に転送される。５Ｇネットワークは、通信ネットワークの単なる一例示的タイプであり、本開示の実施形態は、５Ｇに続く後続世代の有線技術または無線技術と同様に、３Ｇまたは４Ｇなどの前世代の無線通信または有線通信を利用することができることを理解されたい。

【００６２】

一実施形態では、ゲームコンソール４０２及びサーバシステム４０４のいずれかは、本明細書ではコンピューティングデバイスと称される。コンピューティングデバイスの他の例には、タブレット、スマートフォン、ラップトップコンピュータ、デスクトップコンピュータ、及びスマートテレビが含まれる。

【００６３】

一実施形態では、ゲームコードｇｃ１～ｇｃＮの各々は、サーバシステム４０４またはレガシーマシンの別個のメモリデバイスに格納されている。例えば、ゲームコードｇｃ１は、第１のレガシーマシンのメモリデバイスに格納され、ゲームコードｇｃ２は、第２のレガシーマシンのメモリデバイスに格納される。別の例として、ゲームコードｇｃ１は、サーバシステム４０４の第１のメモリデバイスに格納され、ゲームコードｇｃ２は、サーバシステム４０４の第２のメモリデバイスに格納される。

【００６４】

一実施形態では、メモリデバイス４１２またはメモリデバイス４０６はキャッシュではない。むしろ、メモリデバイス４１２またはメモリデバイス４０６の各々は、ＲＡＭなどのメインメモリである。

【００６５】

一実施形態では、メモリデバイス４１２は、メモリコントローラに結合されている。メモリコントローラは、メモリデバイス４１２からデータを読み取り、メモリデバイス４１２にデータを書き込む。メモリコントローラは、サーバシステム４０４のネットワークインターフェースコントローラに結合されている。メモリコントローラは、サーバシステム４０４のネットワークインターフェースコントローラから受信したデータを、格納のためにメモリデバイス４１２に送信する。メモリコントローラはまた、メモリデバイス４１２から受信したデータを、コンピュータネットワーク４０８を介してゲームコンソール４０２に送信するために、サーバシステム４０４のネットワークインターフェースコントローラにも送信する。

【００６６】

図４Ｂは、システム４５０の実施形態の図であり、エミュレーションプロセッサシステム４０９及び入力プロセッサシステム４０７が、サーバシステム４０４内に配置され、画像フレーム４２２が、コンピュータネットワーク４０８を介してサーバシステム４０４から仮想環境または仮想シーンを表示するためのディスプレイデバイス４１０へ送信されることを示している。システム４５０は、サーバシステム４０４、ディスプレイデバイス４１０、及びハンドヘルドコントローラ４１４を含む。

【００６７】

サーバシステム４０４は、メモリデバイス４１２、入力プロセッサシステム４０７、及びエミュレーションプロセッサシステム４０９を含む。メモリデバイス４１２は、エミュレーションプロセッサシステム４０９に結合され、入力プロセッサシステム４０７は、エミュレーションプロセッサシステム４０９に結合されている。ディスプレイデバイス４１

０は、ディスプレイデバイス４１０のネットワークインターフェースコントローラを介してコンピュータネットワーク４０８に結合されている。ディスプレイデバイス４１０は、ディスプレイデバイス４１０のネットワークインターフェースコントローラに結合されたプロセッサを含む。ディスプレイデバイス４１０のプロセッサは、ゲームタイトルＧＮ及びゲームコードｇｃＮを有するレガシーゲームＮのプレイ中にユーザ入力４１９を受信し、ディスプレイデバイス４１０のネットワークインターフェースコントローラにユーザ入力４１９を送信する。ディスプレイデバイス４１０のネットワークインターフェースコントローラは、コンピュータネットワーク４０８を介して、サーバシステム４０４の入力プロセッサシステム４０７にユーザ入力４１９を送信する。

【００６８】

ユーザ入力４１９を受信すると、入力プロセッサシステム４０７は、ユーザ入力４１９をユーザ入力４２０に変換し、ユーザ入力４２０をエミュレーションプロセッサシステム４０９に提供する。ユーザ入力４２０を受信すると、エミュレーションプロセッサシステム４０９は、画像フレーム４２２の生成のために基本ブロック１～ｎをコンパイルするために、ゲームコードｇｃＮに関して図４Ａを参照して上述した同じ機能を実行する。サーバシステム４０４は、ディスプレイデバイス４１０のディスプレイ画面上に仮想環境４５２などの仮想環境を表示するために、コンピュータネットワーク４０８を介して、ディスプレイデバイス４１０に画像フレーム４２２を送信する。例えば、仮想環境４５２は、戦闘機の一例である仮想オブジェクト４５４を含む。この例では、仮想環境４５２は、仮想ピラミッド４５５及び仮想構造４５６などの１つ以上の仮想オブジェクトを含む仮想背景を含む。この例では、仮想オブジェクト４５４は、ゲームコードｇｃＮを有するレガシーゲームＮのプレイ中に、仮想ピラミッド４５５及び仮想構造４５６に向けて仮想ミサイルを発射することができる。

【００６９】

一実施形態では、サーバシステム４０４とディスプレイデバイス４１０との間のデータの通信は、ネットワーク通信プロトコルを介して行われる。例えば、サーバシステム４０４は、データをパケットに変換するネットワークインターフェースコントローラを含む。サーバシステム４０４のネットワークインターフェースコントローラは、エミュレーションプロセッサシステムからデータを受信するためにエミュレーションプロセッサシステム４０９に結合され、ネットワーク通信プロトコルを適用することによって１つ以上のパケット内にデータを埋め込む。パケットは、サーバシステム４０４のネットワークインターフェースコントローラから、コンピュータネットワーク４０８を介してディスプレイデバイス４１０に転送される。ディスプレイデバイス４１０のネットワークインターフェースコントローラは、ネットワーク通信プロトコルを適用することによって、１つ以上のパケットからデータを抽出する。ディスプレイデバイスのネットワークインターフェースコントローラは、ディスプレイデバイス４１０のプロセッサに結合されている。ディスプレイデバイスのネットワークインターフェースコントローラは、コンピュータネットワーク４０８から受信したデータをディスプレイデバイス４１０のプロセッサに提供する。ディスプレイデバイス４１０のプロセッサは、画像フレーム４２２などのデータを、ディスプレイデバイス４１０のディスプレイ画面上にレンダリングする。さらに、ディスプレイデバイス４１０のネットワークインターフェースコントローラは、ディスプレイデバイス４１０のプロセッサからデータを受信し、ネットワーク通信プロトコルを適用することによって１つ以上のパケット内にデータを埋め込み、コンピュータネットワーク４０８を介して１つ以上のパケットをサーバシステム４０４に送信する。サーバシステム４０４のネットワークインターフェースコントローラは、コンピュータネットワーク４０８から受信した１つ以上のパケットへのネットワーク通信プロトコルに適用して、１つ以上のパケットからデータを抽出し、データをエミュレーションプロセッサシステム４０９に送信する。

【００７０】

一実施形態では、コンピュータネットワーク４０８に加えて、またはその代わりに、サーバシステム４０４とディスプレイデバイス４１０との間でデータを通信するためにセル

ラーネットワークが使用される。例えば、サーバシステム４０４とディスプレイデバイスとの間の通信は、無線技術を使用して促進される。

【００７１】

一実施形態では、ディスプレイデバイス４１０の代わりに、ヘッドマウントディスプレイ（ＨＭＤ）が使用される。ヘッドマウントディスプレイは、ユーザ１の頭に装着され、ＬＥＤ画面、ＯＬＥＤ画面、ＬＣＤ画面などのディスプレイ画面を含む。ＨＭＤは、ディスプレイデバイス４１０によって実行されるものと同じ機能を実行する。

【００７２】

図５Ａは、基本ブロックの一実施形態を説明するための図である。各基本ブロックは、ソースレジスタアドレス、宛先レジスタアドレス、及び操作を含む。例えば、基本ブロック１は、ソースレジスタアドレス１、宛先レジスタアドレス１、及び操作１を含む。基本ブロック２は、ソースレジスタアドレス２、宛先レジスタアドレス２、及び操作２を含み、基本ブロックｎは、ソースレジスタアドレスｎ、宛先レジスタアドレスｎ、及び操作ｎを含む。一例として、ソースレジスタアドレスは、キャッシュ１０２内の１つ以上のソースレジスタのアドレスであり、宛先レジスタアドレスは、キャッシュ１０２内の１つ以上の宛先レジスタのアドレスである。基本ブロックの操作の例には、ジャンプ操作、分岐操作、読み取り操作、書き込み操作、比較操作、リターン操作が含まれる。基本ブロックの操作のさらなる例には、加算操作、減算操作、乗算操作、及び除算操作などの算術操作が含まれる。

【００７３】

一例として、操作ｎが読み出し操作のとき、データがソースレジスタアドレスｎから読み出され、基本ブロックｎを実行する。別の例として、操作ｎが書き込み操作である場合、データが宛先レジスタアドレスｎに書き込まれ、基本ブロックｎを実行する。別の例として、操作ｎが移動操作であるとき、データはソースレジスタアドレスｎから読み取られ、操作ｎはデータに対して実行され、データは宛先レジスタアドレスｎに書き込まれ、基本ブロックｎを実行する。さらに別の例として、操作ｎが比較操作であるとき、基本ブロックｎに記載された第１のソースレジスタアドレスに格納されたデータの第１の値が、基本ブロックｎに記載された第２のソースレジスタアドレスに格納されたデータの第２の値と比較され、比較結果を生成し、この比較結果が宛先レジスタアドレスｎに格納されて基本ブロックｎを実行する。別の例として、操作ｎが加算操作であるとき、基本ブロックｎ内に記載された第１のソースアドレスに格納されたデータの第１の値が、基本ブロックｎ内に示された第２のソースアドレスに格納されたデータの第２の値に加算され、加算結果を生成し、この加算結果が宛先レジスタアドレスｎに格納され、基本ブロックｎを実行する。さらに別の例として、本明細書で説明する仮想オブジェクトが、位置Ｐ１から位置Ｐ２に移動するためのものであるとき、かつ、操作ｎが、仮想オブジェクトの位置がＰ１からＰ２に更新される書き込み操作であるとき、宛先レジスタアドレスｎでの位置Ｐ１は、基本ブロックｎを実行するために位置Ｐ２で上書きされる。この例では、基本ブロックｎの実行は、仮想オブジェクトが位置Ｐ１から位置Ｐ２に移動することであることをエミュレーションプロセッサシステム４０９に示す。また、この例では、ユーザ入力４２０（図４Ａ）は、エミュレーションプロセッサシステム４０９に仮想オブジェクトを位置Ｐ１からＰ２に移動するように指示する。同様に、別の例として、本明細書で説明する仮想オブジェクトが、方向Ｏ１から方向Ｏ２に移動するものであり、操作ｎが、仮想オブジェクトの方向がＯ１からＯ２に更新される書き込み操作であるとき、宛先レジスタアドレスｎの方向Ｏ１は、方向Ｏ２で上書きされ、基本ブロックｎを実行する。この例では、基本ブロックｎの実行は、仮想オブジェクトが方向Ｏ１から方向Ｏ２に移動することであることをエミュレーションプロセッサシステム４０９に示す。さらに、この例では、ユーザ入力４２０は、エミュレーションプロセッサシステム４０９に仮想オブジェクトを方向Ｏ１からＯ２に移動するように指示する。

【００７４】

さらに別の例として、本明細書で説明する仮想オブジェクトの一部分が、色を赤から緑

に変更するもので、操作ｎが、仮想オブジェクトの色が赤から緑に更新される書き込み操作であるとき、宛先レジスタアドレスｎで赤色を表すデータは、緑色を表すデータで上書きされ、基本ブロックｎを実行する。この例では、基本ブロックｎの実行は、仮想オブジェクトの一部分の色が赤から緑に変更されるものであることをエミュレーションプロセッサシステム４０９に示す。また、この例では、ユーザ入力４２０は、仮想オブジェクトの部分の色を赤から緑に変更するようにエミュレーションプロセッサシステム４０９に指示する。同様の方法で、強度及びテクスチャなどの他のパラメータが、ユーザ入力４２０に基づいて変更されることができる。強度とテクスチャは、レガシーグラフィカルパラメータの例である。

【００７５】

各基本ブロックは、基本ブロック実行のサイクル数を含む。例えば、基本ブロック１は、基本ブロック１実行のサイクル数１を含む。別の例として、基本ブロック２は、基本ブロック２の実行のサイクル数２を含み、基本ブロックｎは、基本ブロックｎの実行のサイクル数ｎを含む。一例として、基本ブロック実行の推定サイクル数は、基本ブロックのコンパイル時に基本ブロックコンパイラ１０４（図１）によって推定される。この例では、推定サイクル数は、基本ブロックに格納されている。また、この例では、基本ブロックがブロックディスパッチャ３０２によって実行された後（図３）、ブロックディスパッチャ３０２は、上記の方法で推定実行サイクル数を実数で更新し、実数をブロックコンパイラ１０４に提供する。推定されたサイクル数は、ブロックコンパイラ１０４によって基本ブロックの実際のカウントに置き換えられる。別の例として、操作ｎの実行のサイクル数の実際のカウントは、ブロックディスパッチャ３０２によって生成され、基本ブロックｎに格納される。この例では、操作ｎの実行のサイクル数の推定はない。

【００７６】

さらに、別の例として、基本ブロック１～ｎのうちの１つ以上は、基本ブロック１～ｎのうちの１つ以上が有効性についてチェックされることを示す無効なマークを含む。例えば、基本ブロックｎは、無効なマークｎを含む。

【００７７】

エミュレートされたＰＵコード１０６をゲームコードＧＣＮの基本ブロック１～ｎに変換することにより、フックブロックなどのフックが、基本ブロック１～ｎの任意の２つの間に挿入され得ることに留意されたい。例えば、フックブロックｎは、基本ブロック（ｎ－１）とｎの間に挿入されることができる。フックブロックｎは、基本ブロックｎの構造と同じ構造である。例えば、フックブロックには、ソースレジスタアドレス、宛先レジスタアドレス、操作、及びフックブロックの操作の実行サイクル数が含まれる。一例として、レガシーマシンに関連するセキュリティの問題のため、本明細書で説明されるフックは、レガシーマシンで実行するためにレガシーマシンに格納されたエミュレートされたＰＵコード１０６（図１）の命令の間に挿入されることができない。

【００７８】

さらに、基本ブロック１～ｎは、その中に固定されるなど、キャッシュ１０２にキー入力されることに留意されたい（図１）。例えば、基本ブロック１は、キャッシュ１０２内の基本ブロック１の開始場所を示す開始メモリアドレス１を有する。また、基本ブロック１は、キャッシュ１０２内の基本ブロック１の終了場所を示す終了メモリアドレス１を有する。別の例として、基本ブロック１の終了アドレス１は、開始メモリアドレス１からのキャッシュ１０６内のオフセットによって示される。さらに別の例として、基本ブロック２は、キャッシュ１０２内の基本ブロック２の開始場所を示す開始メモリアドレス２を有する。また、基本ブロック２は、キャッシュ１０２内の基本ブロック２の終了場所を示す終了メモリアドレス２を有する。別の例として、基本ブロック２の終了アドレス２は、開始メモリアドレス２からのキャッシュ１０６内のオフセットによって示される。同様に、別の例として、基本ブロックｎは、キャッシュ１０２内の基本ブロックｎの開始場所を示す開始メモリアドレスｎを有する。また、基本ブロックｎは、キャッシュ１０２内の基本ブロックｎの終了場所を示す終了メモリアドレスｎを有する。別の例として、基本ブロッ

クｎの終了アドレスｎは、開始メモリアドレスｎからのキャッシュ１０６内のオフセットによって示される。キャッシュ１０２に格納された基本ブロック１～ｎの開始及び終了メモリアドレスから、基本ブロックコンパイラ１０２などのエミュレーションプロセッサシステム４０９（図４Ａ）は、キャッシュ１０２内の基本ブロック１～ｎの場所を特定することができる。

【００７９】

また、更新されたマシンのブロックディスパッチャ３０２が、基本ブロック１を実行した直後に基本ブロックｎを実行することをユーザ入力が示す場合、ブロックディスパッチャ３０２は、キャッシュ１０２内の基本ブロック２～（ｎ－１）の実行をスキップし、基本ブロック１から基本ブロックｎにジャンプすることに留意されたい。この場合、基本ブロック１の実行は、ブロックディスパッチャ３０２が基本ブロックｎにジャンプするときに、ブロックディスパッチャ３０２によって閉じられる。また、この場合、次の基本ブロックの開始アドレスは、前の基本ブロックの終了アドレスに連続している。例えば、開始アドレス２は終了アドレス１に連続し、開始アドレスｎは基本ブロック（ｎ－１）の終了アドレス（ｎ－１）に連続している。

【００８０】

一実施形態では、ソースレジスタアドレス１～ｎは、キャッシュ１０２のレジスタのメモリアドレスであり、かつ、宛先レジスタアドレス１～ｎは、キャッシュ１０２内のレジスタのメモリアドレスである。

【００８１】

一実施形態では、基本ブロックは複数の操作を含む。例えば、基本ブロックｎは、第１の操作、第１のソースレジスタアドレス、及び第１の宛先レジスタアドレスを含む。基本ブロックｎは、第２の操作、第２のソースレジスタアドレス、及び第２の宛先レジスタアドレスをさらに含む。

【００８２】

一実施形態では、基本ブロックは、操作、複数のソースアドレス、及び宛先アドレスを含む。

【００８３】

一実施形態では、基本ブロックは、操作、複数の宛先アドレス、及びソースアドレスを含む。

【００８４】

一実施形態では、基本ブロックは、複数の操作、複数のソースアドレス、及び複数の宛先アドレスを含む。

【００８５】

一実施形態では、基本ブロックは、１つ以上の操作、１つ以上のソースアドレス、及び１つ以上の宛先アドレスを含む。

【００８６】

一実施形態では、基本ブロックは、ソースレジスタアドレスまたは宛先レジスタアドレスのいずれかを含むが、両方は含まない。

【００８７】

一実施形態では、ブロックコンパイラ１０２は、基本ブロックｎの実行のためのサイクル数を推定しない。むしろ、ブロックディスパッチャ３０２は、基本ブロックｎの実行のサイクル数の実際のカウントを生成し、実際のカウントを基本ブロックｎに格納する。

【００８８】

図５Ｂは、基本ブロックコンパイラ１０４（図１）によって実行されるコンパイル操作の実施形態を説明するための図である。エミュレートされたＰＵコード命令Ｍの例は、命令５５０として示され、基本ブロックｎの例は、基本ブロック５５２として示され、ここで、Ｍは正の整数である。命令５５０は、ａビットの長さを有するソースアドレスＭと、ｂビットの長さを有する宛先アドレスＭと、長さｃのビットによって表される操作Ｍと、を含み、ここで、ａ、ｂ、及びｃは正の整数である。一例として、ａは４、ｂは４、ｃは

３２である。一例として、ａビット、ｂビット、及びｃビットは、レガシーマシンの１つ以上のメモリデバイスに格納される。例えば、ａビットは、メモリデバイス４０６または４１２に格納される（図４Ａ及び図４Ｂ）。操作５５２は、ｄビットの長さを有するソースレジスタアドレスｎと、ｅビットの長さを有する宛先アドレスｎと、長さｆのビットによって表される操作ｎを含み、ここで、ｄ、ｅ、及びｆは正の整数である。一例として、ｄは８、ｅは８、ｆは６４である。一例として、ｄビット、ｅビット、及びｆビットは、更新されたマシンの１つ以上のレジスタに格納される。例として、ｄはａより大きく、ｅはｂより大きく、ｆはｃより大きい。説明すると、ａが４、ｂが４、ｃが３２のとき、ｄは８、ｅは８、ｆは６４である。別の例として、ａが４、ｂが４、ｃが３２のとき、ｄは１６、ｅは１６、ｆは１２８である。

【００８９】

コンパイル操作を実行するために、基本ブロックコンパイラ１０４（図１）は、ソースアドレスＭをソースレジスタアドレスｎに変換し、宛先アドレスＭを宛先レジスタアドレスｎに変換し、操作Ｍを操作ｎに変換して、エミュレートされたＰＵコード命令Ｍから基本ブロックｎを生成する。例えば、基本ブロックコンパイラ１０４は、ソースアドレスＭの４ビットを右にシフトして、キャッシュ１０２内の８つのソースレジスタの４つのメモリアドレスを占有し、８つのソースレジスタの残りの４つのメモリアドレスの任意のビットをマスクする。ソースアドレスＭの４ビットによって占められる４つのメモリアドレスは、キャッシュ１０２内の８つのソースレジスタの最下位位置にあり、ビットがマスクされる残りの４つのメモリアドレスは、キャッシュ１０２内のソースレジスタの８つの最上位位置にある。

【００９０】

別の例として、基本ブロックコンパイラ１０４は、宛先アドレスＭの４ビットを右にシフトして、キャッシュ１０２内の８つの宛先レジスタの４つのメモリアドレスを占有し、８つの宛先レジスタの残りの４つのメモリアドレスの任意のビットをマスクする。宛先アドレスｍの４ビットによって占められる４つのメモリアドレスは、キャッシュ１０２内の８つの宛先レジスタの最下位位置にあり、ビットがマスクされる残りの４つのメモリアドレスは、キャッシュ１０２内の８つの宛先レジスタの最上位位置にある。

【００９１】

同様に、別の例として、基本ブロックコンパイラ１０４は、操作Ｍの３２ビットを右にシフトして、キャッシュ１０２内の６４個の操作レジスタの３２個のメモリアドレスを占有し、６４個の操作レジスタの残りの３２個のメモリアドレスの任意のビットをマスクする。操作Ｍの３２ビットによって占有される３２個のメモリアドレスは、キャッシュ１０２内の３２個の操作レジスタの最下位位置にあり、ビットがマスクされる残りの３２個のメモリアドレスは、キャッシュ１０２内の６４個の操作レジスタの最上位位置にある。操作ｎは、キャッシュ１０２の操作レジスタに格納される。

【００９２】

図６Ａは、エミュレーションプロセッサシステム４０９の構成要素を説明するためのシステム６００の実施形態の図である。システム６００は、メモリデバイス４１２と、エミュレーションプロセッサシステム４０９と、を含む。エミュレーションプロセッサシステム４０９の基本ブロックコンパイラ１０４は、パーサまたはデコーダ６０２を含む。基本ブロックコンパイラ１０４は、ブロッククリエータ６０４、ブロックキャッシャ６０６Ａ、及びブロックリーダ６０８をさらに含む。エミュレーションプロセッサシステム４０９は、ブロックディスパッチャ３０２を含む。

【００９３】

一例として、パーサ６０９、ブロッククリエータ６０４、ブロックキャッシャ６０６、ブロックリーダ６０８、及びブロックディスパッチャ３０２の各々は、ソフトウェア、ハードウェア、またはそれらの組み合わせを使用して実装される。例えば、パーサ６０９、ブロッククリエータ６０４、ブロックキャッシャ６０６、ブロックリーダ６０８、及びブロックディスパッチャ３０２の各々は、ＰＬＤまたはＡＳＩＣまたはコントローラまたは

プロセッサまたはコンピュータプログラムの一部分など、別個の集積回路である。別の例として、パーサ６０９、ブロッククリエータ６０４、ブロックキャッシャ６０６、ブロッククリーダ６０８、及びブロックディスパッチャ３０２の各々は、別個のコンピュータソフトウェアプログラムである。

【００９４】

ゲームコードｇｃＮは、エミュレートされたＰＵコード命令１、エミュレートされたＰＵコード命令２というように、エミュレートされたＰＵコード命令Ｍまでの、複数の命令を含む。例えば、ゲームコードｇｃＮの各命令は、一連のビットであり、レガシーマシンのプロセッサによって実行され得、仮想オブジェクトを位置Ｐ１から位置Ｐ２に移動すること、または、仮想オブジェクトの方向をＯ１からＯ２に変更すること、または、仮想オブジェクトの一部分のパラメータを改変すること、などの機能を実装する。

【００９５】

パーサ６０２は、ブロックキャッシャ６０６及びブロックリーダ６０８に結合されているブロッククリエータ６０４に結合されている。ブロックキャッシャ６０６及びブロッククリーダ６０８はキャッシュ１０２に結合されている。ブロッククリーダ６０８は、ブロックディスパッチャ３０２に結合されている。

【００９６】

図６Ｂは、基本ブロック１～ｎをコンパイル及び実行するための方法６５０の実施形態を説明するフローチャートである。方法６５０は、図６Ａのシステム６００を使用して示されている。方法６５０の操作６５２において、ブロッククリエータ６０４は、ユーザ入力１などの任意のユーザ入力が受信されているかどうかを判定する。一例として、ユーザ入力１は、ユーザ１によって選択されたハンドヘルドコントローラ４１４（図４Ａ）の１つ以上のボタンの１つ以上の識別に基づいて生成される信号である。別の例として、ユーザ入力１は、ハンドヘルドコントローラ４１４を中心とするｘｙｚ座標系の原点に対するピッチ、ヨー、及びロールなどの動きの尺度に基づいて生成される信号である。さらに別の例として、ユーザ入力１は、ＨＭＤのｘｙｚ座標系の原点に対するピッチ、ヨー、及びロールなどの動きの尺度に基づいて生成される信号である。ユーザ入力１の例は、ユーザ入力４２０（図４Ａ）である。ユーザ入力１の別の例は、ユーザ入力４１８である（図４Ａ）。

【００９７】

ブロッククリエータ６０４は、ユーザ入力が受信されていないと判定すると、ユーザ入力が操作６５２で受信されたかどうかを引き続きチェックする。例えば、基本ブロックコンパイラ１０４は、キャッシュ１０２（図１）内に格納された基本ブロック（ｎ＋１）～ｐのいずれも実行せず、基本ブロックコンパイラ１０４（図１）は、操作６５２でユーザ入力が受信されていないとき、エミュレートされたＰＵコード命令１～Ｍ（図６Ａ）のいずれも識別しない（ここでｐは正の整数である）。操作６５２でユーザ入力を受信した時点では、基本ブロック１～ｎは生成されず、キャッシュ１０２内に格納されていないと想定されている。

【００９８】

一方で、操作６５２でユーザ入力が受信されたとの判定に応答して、方法６５０の操作６５４において、ブロッククリエータ６０４は、キャッシュ１０２内に格納された基本ブロック（ｎ＋１）～ｐのうちの１つ以上が、ユーザ入力を満たすかどうかを判定する。例えば、ブロッククリエータ６０４は、キャッシュ１０２から基本ブロック（ｎ＋１）～ｐにアクセスするために要求をブロッククリーダ６０８に送信する。この例では、ブロッククリーダ６０８は、要求を受信すると、キャッシュ１０２から基本ブロック（ｎ＋１）～ｐを読み取り、基本ブロック（ｎ＋１）～ｐをブロッククリエータ６０４に送信する。さらに、この例では、ブロッククリエータ６０４は、基本ブロック（ｎ＋１）～ｐのうちの１つ以上の機能が、処理などのユーザ入力１を満たすかどうかを判定する。説明すると、ユーザ入力１が、仮想オブジェクトの位置を位置Ｐ１から位置Ｐ２に変更することを示すとき、ブロッククリエータ６０４は、基本ブロック（ｎ＋１）～ｐのいずれかが、位置Ｐ１を

位置Ｐ２で上書きする操作が含むかどうかを判定する。１つ以上の基本ブロック（ｎ＋１）～ｐが位置Ｐ１をＰ２で上書きする操作を含むと判定すると、ブロッククリエータ６０４は、ブロック（ｎ＋１）～ｐのうちの１つ以上の１つ以上の機能がユーザ入力１を満たすと判定する。一方で、基本ブロック（ｎ＋１）～ｐのいずれも位置Ｐ１をＰ２で上書きする操作を含まないと判定すると、ブロッククリエータ６０４は、基本ブロック（ｎ＋１）～ｐの機能がユーザ入力１を満たさないと判定する。

【００９９】

別の説明として、ユーザ入力１が、仮想オブジェクトの方向を方向Ｏ１から方向Ｏ２に変更するためのものであるとき、ブロッククリエータ６０４は、基本ブロック（ｎ＋１）～ｐのいずれかが、方向Ｏ１を方向Ｏ２で上書きする操作が含まれるかどうかを判定する。基本ブロック（ｎ＋１）～ｐのうちの１つ以上が、方向Ｏ１をＯ２で上書きする操作を含むと判定すると、ブロッククリエータ６０４は、基本ブロック（ｎ＋１）～ｐのうちの１つ以上の１つ以上の機能が、ユーザ入力１を満たすと判定する。一方で、基本ブロック（ｎ＋１）～ｐのいずれも方向Ｏ１をＯ２で上書きする操作を含まないと判定した場合、ブロッククリエータ６０４は、基本ブロック（ｎ＋１）～ｐの機能が、ユーザ入力１を満たさないと判定する。さらに別の説明として、ユーザ入力１が、仮想オブジェクトの一部分のパラメータの値を第１の値から第２の値に変更することであるとき、ブロッククリエータは、基本ブロック（ｎ＋１）～ｐのいずれかが、第１の値を第２の値で上書きする操作が含まれるかどうかを判定する。基本ブロック（ｎ＋１）～ｐのうちの１つ以上が、第１の値を第２の値で上書きする操作を含むと判定すると、ブロッククリエータ６０４は、基本ブロック（ｎ＋１）～ｐのうちの１つ以上の１つ以上の機能が、ユーザ入力１を満たすと判定する。一方で、基本ブロック（ｎ＋１）～ｐのいずれも第１の値を第２の値で上書きする操作を含まないと判定したとき、ブロッククリエータ６０４は、基本ブロック（ｎ＋１）～ｐの機能がユーザ入力１を満たさないと判定する。

【０１００】

さらに別の説明として、ユーザ入力１がレガシーゲームＮを開始することであるとき、ブロッククリエータ６０４は、基本ブロック（ｎ＋１）～ｐのいずれかが、レガシーゲームＮの開始を表すためのレガシーゲームＮのゲームタイトルＧＮを含むアニメーションを生成するために実行され得るかどうかを判定する。基本ブロック（ｎ＋１）～ｐのうちの１つ以上が、アニメーションを生成するために実行され得ると判定すると、ブロッククリエータ６０４は、基本ブロック（ｎ＋１）～ｐのうちの１つ以上の１つ以上の機能が、ユーザ入力１を満たすと判定する。一方で、基本ブロック（ｎ＋１）～ｐのいずれもアニメーションを生成するために実行され得ないと判定すると、ブロッククリエータ６０４は、基本ブロック（ｎ＋１）～ｐの機能が、ユーザ入力１を満たさないと判定する。

【０１０１】

基本ブロック（ｎ＋１）～ｐのうちの１つ以上の機能が、ユーザ入力１を満たすと判定すると、方法６００の操作６５６において、ブロックディスパッチャ３０２は、基本ブロック（ｎ＋１）～ｐのうちの１つ以上を実行する。例えば、基本ブロック（ｎ＋１）～ｐのうちの１つ以上の機能がユーザ入力１を満たすと判定すると、ブロッククリエータ６０４は、基本ブロック（ｎ＋１）～ｐのうちの１つ以上を実行するためにブロックディスパッチャ３０２に命令を送信する。この例では、命令の受信に応答して、ブロックディスパッチャ３０２は、コマンドをブロックリーダ６０８に送信して、操作６５２で受信したユーザ入力を満たすために、キャッシュ１０２から基本ブロック（ｎ＋１）～ｐのうちの１つ以上を読み取る。また、この例では、コマンドに応答してキャッシュ１０２から基本ブロックのうちの１つ以上を受信すると、ブロックディスパッチャ３０２は、基本ブロック（ｎ＋１）～ｐのうちの１つ以上を実行する。

【０１０２】

一方で、ブロック（ｎ＋１）～ｐの機能がユーザ入力１を満たさないと判定すると、方法６００の操作６５８において、ブロックコンパイラ１０２は、操作６５２で受信したユーザ入力を処理するためのエミュレートされたＰＵコード命令１～Ｍの１つ以上を識別す

る。例えば、ブロック（ｎ＋１）～ｐまでの機能がユーザ入力１を満たさないと判定すると、ブロッククリエータ６０４は、パーサ６０２に要求を送信してゲームコードｇｃＮを解析して、ユーザ入力１で識別された機能を満たすゲームコードｇｃＮのエミュレートされたＰＵコード命令１～Ｍの１つ以上を識別及び取得する。この例では、要求の受信に応答して、パーサ６０２は、メモリデバイス４１２にアクセスして、ゲームコードｇｃＮを解析し、エミュレートされたＰＵコード命令１～Ｍのうちの１つ以上の機能がユーザ入力１を満たすかどうかを判定し、エミュレートされたＰＵコード命令１～Ｍのうちの１つ以上の機能が、ユーザ入力１を満たすと判定すると、パーサ６０２は、エミュレートされたＰＵコード命令１～Ｍの１つ以上をブロッククリエータ６０４に提供する。

【０１０３】

説明すると、ユーザ入力１が、仮想オブジェクトの位置を位置Ｐ１から位置Ｐ２に変更するためのものであるとき、パーサ６０２は、エミュレートされたＰＵコード命令１～Ｍのいずれかが、位置Ｐ１を位置Ｐ２で上書きする操作を含むかどうかを判定する。エミュレートされたＰＵコード命令１～Ｍのうちの１つ以上が、位置Ｐ１をＰ２で上書きする操作を含むと判定すると、パーサ６０２は、エミュレートされたＰＵコード命令１～Ｍのうちの１つ以上の１つ以上の機能が、ユーザ入力１を満たすと判定する。一方で、エミュレートされたＰＵコード命令１～Ｍのいずれも位置Ｐ１をＰ２で上書きする操作を含んでいないと判定すると、パーサ６０２は、エミュレートされたＰＵコード命令１～Ｍの機能がユーザ入力１を満たさないと判定する。

【０１０４】

別の説明として、ユーザ入力１が、仮想オブジェクトの方向を方向Ｏ１から方向Ｏ２に変更するものであるとき、パーサ６０２は、エミュレートされたＰＵコード命令１～Ｍのいずれかが、方向Ｏ１を方向Ｏ２で上書きする操作を含むかどうかを判定する。エミュレートされたＰＵコード命令１～Ｍのうちの１つ以上が、方向Ｏ１をＯ２で上書きする操作を含むと判定すると、パーサ６０２は、エミュレートされたＰＵコード命令１～Ｍのうちの１つ以上の１つ以上の機能がユーザ入力１を満たすと判定する。一方で、エミュレートされたＰＵコード命令１～Ｍのいずれも位置Ｏ１をＯ２で上書きする操作を含んでいないと判定すると、パーサ６０２は、エミュレートされたＰＵコード命令１～Ｍの機能がユーザ入力１を満たさないと判定する。さらに別の例として、ユーザ入力１が、仮想オブジェクトの一部分のパラメータの値を第１の値から第２の値に変更するものであるとき、パーサ６０２は、エミュレートされたＰＵコード命令１～Ｍのいずれかが、第１の値を第２の値で上書きする操作を含むかどうかを判定する。エミュレートされたＰＵコード命令１～Ｍのうちの１つ以上が、第１の値を第２の値で上書きする操作を含むと判定すると、パーサ６０２は、エミュレートされたＰＵコード命令１～Ｍのうちの１つ以上の１つ以上の機能が、ユーザ入力１を満たすと判定する。一方で、エミュレートされたＰＵコード命令１～Ｍのいずれも第１の値を第２の値で上書きする操作を含まないと判定すると、パーサ６０２は、エミュレートされたＰＵコード命令１～Ｍの機能が、ユーザ入力１を満たさないと判定する。

【０１０５】

方法６００の操作６６０において、ユーザ入力１を満たすエミュレートされたＰＵコード命令１～Ｍのうちの１つ以上がパーサ６０２から受信されると、ブロッククリエータ６０４は、上述のコンパイル操作を適用して、エミュレートされたＰＵコード命令１～Ｍのうちの１つ以上から１つ以上の基本ブロック１～ｎを生成する。例えば、ブロッククリエータ６０４は、１つ以上の基本ブロック１～ｎをコンパイルし、１つ以上の基本ブロック１～ｎをブロックキャッシャ６０６に送信し、ブロックディスパッチャ３０２に命令を送信して、基本ブロック１～ｎのうちの１つ以上を実行する。１つ以上の基本ブロック１～ｎを受信すると、ブロックキャッシャ６０６は、１つ以上の基本ブロック１～ｎをキャッシュ１０２に格納する。

【０１０６】

方法６００の操作６６２において、１つ以上の基本ブロック１～ｎを実行するための命

令をブロッククリエータ６０４から受信すると、ブロックディスパッチャ３０２は、１つ以上の基本ブロック１～ｎを実行して、操作６５２で受信したユーザ入力を処理する。例えば、ブロックディスパッチャ３０２は、キャッシュ１０２から基本ブロック１～ｎのうちの１つ以上を読み取るために、ブロックリーダ６０８に要求を送信する。コマンドを受信すると、ブロックリーダ６０８は、キャッシュ１０２から基本ブロック１～ｎのうちの１つ以上を読み取り、基本ブロック１～ｎのうちの１つ以上をブロックディスパッチャ３０２に提供する。基本ブロック１～ｎのうちの１つ以上を受信すると、ブロックディスパッチャ３０２は、基本ブロック１～ｎのうちの１つ以上を実行して、仮想環境４５２を生成する（図４Ｂ）。説明すると、ブロックディスパッチャ３０２は、基本ブロック１～ｎのうちの１つ以上を実行して、画像フレームを生成し、この画像フレームは、仮想オブジェクト４５４の位置及び方向などの仮想環境データと、仮想オブジェクト４５４のレガシーグラフィカルパラメータなどのパラメータ、及び仮想環境４５２内の他の仮想オブジェクトの位置と方向、及び仮想環境４５２内の他の仮想オブジェクトのレガシーグラフィカルパラメータなどのパラメータを含む。さらに、この例では、仮想オブジェクト４５４の位置もしくは方向、またはそれらの組み合わせが変化するとき、仮想環境が変化する。ブロックディスパッチャ３０２は、画像フレームをエミュレーションプロセッサシステム４０９のＧＰＵに提供して、ディスプレイデバイス４１０（図４Ａ）のディスプレイ画面上に仮想環境４５２をレンダリングなどして、表示する。別の例として、ブロックディスパッチャ３０２は、基本ブロック１～ｎのうちの１つ以上を実行して、レガシーオーディオパラメータを含む１つ以上のオーディオフレームを生成し、１つ以上のオーディオフレームをクライアントデバイスに提供し、１つ以上のオーディオフレームに基づいて音声を出力する。説明すると、各オーディオフレームは、１つ以上のレガシーオーディオパラメータを含むオーディオファイルである。操作６５２においてユーザ入力１が受信された後、ユーザ入力２などの別のユーザ入力が受信されると、方法６００が繰り返される。

【０１０７】

一実施形態では、画像フレームは、ブロックディスパッチャ３０２からディスプレイデバイス４１０のＧＰＵに提供されて、ディスプレイデバイス４１０のディスプレイ画面上に仮想環境４５２を表示する。

【０１０８】

図６Ｃは、異なるユーザ入力について、異なる基本ブロックまたは基本ブロックの異なるセットが動的にコンパイルされることを示すためのシステム６７０の実施形態の図である。システム６７０は、メモリデバイス４１２、基本ブロックコンパイラ１０４、ブロックディスパッチャ３０２、及びキャッシュ１０２を含む。基本ブロックコンパイラ１０４がユーザ入力１を受信すると、基本ブロックコンパイラ１０４は、ユーザ入力１に対応する基本ブロック１が、キャッシュ１０２に格納されているかどうかを判定する。例えば、基本ブロック１が、ユーザ入力１を処理するための機能を含むとき、基本ブロック１は、ユーザ入力１に対応する。説明すると、ユーザ入力１が、仮想オブジェクトを位置Ｐ１から位置Ｐ２に移動するものであり、基本ブロック１が、宛先レジスタアドレス１に格納された位置Ｐ１を位置Ｐ２で更新する操作を含むとき、基本ブロック１は、ユーザ入力１を処理することができる。この図では、ユーザ入力１は、ハンドヘルドコントローラ４１４（図４Ａ）上の右への移動ボタン、または左への移動ボタン、または上への移動ボタン、または下への移動ボタンの選択を示す信号である。同様に、基本ブロック２がユーザ入力１を処理するための機能を含まないとき、基本ブロック２はユーザ入力１に対応しない。説明すると、ユーザ入力１が、仮想オブジェクトの方向を変更せずに位置Ｐ１から位置Ｐ２に仮想オブジェクトを移動するためのものであり、基本ブロック２が、宛先レジスタ２に格納された方向Ｏ１を方向Ｏ２で更新する操作を含むとき、基本ブロック２はユーザ入力１を処理することができない。ユーザ入力１を処理することができる基本ブロック１がキャッシュ１０２に格納されていると判定すると、基本ブロックコンパイラ１０４は、基本ブロック１を実行するために、基本ブロック１をブロックディスパッチャ３０２に提供する。

【０１０９】
一方で、基本ブロック１がキャッシュ１０２内に格納されていないと判定すると、基本ブロックコンパイラ１０４は、メモリデバイス４１２内に格納されたエミュレートされたＰＵコード１０６を解析して、ユーザ入力１に対応するエミュレートされたＰＵコード命令１を識別する。例えば、基本ブロックコンパイラ１０４は、エミュレートされたＰＵコード１０６を解析して、エミュレートされたＰＵコード命令１が、ユーザ入力１を満たすこと（処理することなど）を識別する。説明すると、ユーザ入力１が、仮想オブジェクトを位置Ｐ１から位置Ｐ２に移動させるためのものであり、エミュレートされたＰＵコード命令１が、宛先アドレス１に格納された位置Ｐ１を位置Ｐ２で更新する機能を含むとき、エミュレートされたＰＵコード命令１は、ユーザ入力１を処理することができる。同様に、エミュレートされたＰＵコード命令２が、ユーザ入力１を処理するための機能を含まないとき、エミュレートされたＰＵコード命令２は、ユーザ入力１に対応しない。説明すると、ユーザ入力１が、仮想オブジェクトの方向を変更せずに位置Ｐ１から位置Ｐ２に仮想オブジェクトを移動するためのものであり、エミュレートされたＰＵコード命令２が、宛先アドレス２に格納された方向Ｏ１を方向Ｏ２で更新する操作を含むとき、エミュレートされたＰＵコード命令２は、ユーザ入力１を処理することができない。

【０１１０】
エミュレートされたＰＵコード命令１がユーザ入力１を処理でき、エミュレートされたＰＵコード命令２がユーザ入力１を処理できないと判定すると、基本ブロックコンパイラ１０４は、メモリデバイス４１２からエミュレートされたＰＵコード命令１にアクセス（読み取りなど）し、エミュレートされたＰＵコード命令１から基本ブロック１をコンパイルする。基本ブロックコンパイラ１０２は、ユーザ入力１の受信に応答して、エミュレートされたＰＵコード命令２から基本ブロック２をコンパイルしない。基本ブロックコンパイラ１０４は、基本ブロック１をキャッシュ１０２内に格納し、ブロックディスパッチャ３０２に命令を送信して、基本ブロック１にアクセスして実行する。命令を受信すると、ブロックディスパッチャ３０２は、キャッシュ１０２から基本ブロック１を読み取り、基本ブロック１を実行する。

【０１１１】
同様に、基本ブロックコンパイラ１０４がユーザ入力２を受信すると、基本ブロックコンパイラ１０４は、ユーザ入力２に対応する基本ブロック２が、キャッシュ１０２に格納されているかどうかを判定する。例えば、基本ブロック２が、ユーザ入力２を処理するための機能を含むとき、基本ブロック２は、ユーザ入力２に対応する。説明すると、ユーザ入力２が、仮想オブジェクトを方向Ｏ１から方向Ｏ２に移動するものであり、基本ブロック２が、宛先レジスタアドレス２に格納された方向Ｏ１を方向Ｏ２で更新する操作を含むとき、基本ブロック２は、ユーザ入力２を処理することができる。この図では、ユーザ入力２は、ハンドヘルドコントローラ４１４上の時計回りに回転するボタンか、または反時計回りに回転するボタンの選択を示す信号である。同様に、基本ブロック１がユーザ入力２を処理するための機能を含まないとき、基本ブロック１はユーザ入力２に対応しない。説明すると、ユーザ入力２が、仮想オブジェクトの位置を変更せずに方向Ｏ１から方向Ｏ２に仮想オブジェクトを移動するためのものであり、基本ブロック１が、宛先レジスタ１に格納された位置Ｐ１を位置Ｐ２で更新する操作を含むとき、基本ブロック１はユーザ入力２を処理することができない。ユーザ入力２を処理することができる基本ブロック２がキャッシュ１０２に格納されていると判定すると、基本ブロックコンパイラ１０４は、基本ブロック２を実行するために、基本ブロック２をブロックディスパッチャ３０２に提供する。

【０１１２】
一方で、基本ブロック２がキャッシュ１０２内に格納されていないと判定すると、基本ブロックコンパイラ１０４は、メモリデバイス４１２内に格納されたエミュレートされたＰＵコード１０６を解析して、ユーザ入力２に対応するエミュレートされたＰＵコード命令２を識別する。例えば、基本ブロックコンパイラ１０４は、エミュレートされたＰＵコ

ード１０６を解析して、エミュレートされたＰＵコード命令２が、ユーザ入力２を満たすこと（処理することなど）を識別する。説明すると、ユーザ入力２が、仮想オブジェクトを方向Ｏ１から方向Ｏ２に移動させるためのものであり、エミュレートされたＰＵコード命令２が、宛先アドレス２に格納された方向Ｏ１を方向Ｏ２で更新する機能を含むとき、エミュレートされたＰＵコード命令２は、ユーザ入力２を処理することができる。同様に、エミュレートされたＰＵコード命令１が、ユーザ入力２を処理するための機能を含まないとき、エミュレートされたＰＵコード命令１は、ユーザ入力２に対応しない。説明すると、ユーザ入力２が、仮想オブジェクトの方向を変更せずに、方向Ｏ１から方向Ｏ２に仮想オブジェクトを移動するためのものであり、エミュレートされたＰＵコード命令１が、宛先アドレス１に格納された位置Ｐ１を位置Ｐ２で更新する操作を含むとき、エミュレートされたＰＵコード命令１は、ユーザ入力２を処理することができない。

【０１１３】

エミュレートされたＰＵコード命令２が、ユーザ入力２を処理でき、エミュレートされたＰＵコード命令１が、ユーザ入力２を処理できないと判定すると、基本ブロックコンパイラ１０４は、メモリデバイス４１２からエミュレートされたＰＵコード命令２にアクセス（読み取るなど）して、エミュレートされたＰＵコード命令２をコンパイルして、基本ブロック２を生成する。基本ブロックコンパイラ１０２は、ユーザ入力２の受信に応答して、エミュレートされたＰＵコード命令１をコンパイルしない。基本ブロックコンパイラ１０４は、基本ブロック２をキャッシュ１０２内に格納し、ブロックディスパッチャ３０２に命令を送信して、基本ブロック２にアクセスして実行する。命令を受信すると、ブロックディスパッチャ３０２は、キャッシュ１０２から基本ブロック２を読み取り、基本ブロック２を実行する。

【０１１４】

図７Ａは、メモリデバイス４１２からのエミュレートされたＰＵコード１０６の削除を説明するためのシステム７００の実施形態の図である。システム７００は、ブロッククリエータ６０４、命令リムーバ７０２、及びメモリデバイス４１２を含む。命令リムーバ７０２の例には、プロセッサ、ＡＳＩＣ、ＰＬＤ、コンピュータプログラム、コンピュータプログラムの一部分、及びマイクロコントローラが含まれる。命令リムーバ７０２は、ブロッククリエータ６０４に結合され、かつメモリデバイス４１２に結合されている。

【０１１５】

図７Ｂは、エミュレートされたＰＵコード１０６（図６Ｃ）をメモリデバイス４１２（図６Ｃ）から削除するための方法７２０の実施形態を説明するためのフローチャートである。方法７２０は、エミュレートされたＰＵコード１０６のすべての命令がコンパイルされているかどうかを判定するための操作７０２を含む。例えば、ブロッククリエータ６０４（図７Ａ）は、ゲームコードｇｃＮのエミュレートされたＰＵコード命令１～Ｍのすべてがコンパイルされているかどうかを判定する。説明すると、ゲームコードｇｃＮのエミュレートされたＰＵコード命令１～Ｍのいずれかをコンパイルする前に、ブロッククリエータ６０４は、レガシーマシンのプロセッサに要求を送信して、ゲームコードｇｃＮのエミュレートされたＰＵコード命令１～Ｍのすべてのアイデンティティを取得する。この図では、ブロッククリエータ６０４は、１つ以上のビットなどのゲームコードｇｃＮのアイデンティティをレガシーマシンのプロセッサに送信して、ゲームコードｇｃＮのエミュレートされたＰＵコード命令１～Ｍのすべてのアイデンティティを取得する。エミュレートされたＰＵコード命令１～Ｍのアイデンティティの例には、１つ以上のビットが含まれる。説明すると、エミュレートされたＰＵコード命令Ｍのアイデンティティは、ビットの第１のシーケンスによって表され、エミュレートされたＰＵコード命令Ｍ－１のアイデンティティは、ビットの第２のシーケンスによって表され、これは第１のシーケンスとは異なっている。エミュレートされたＰＵコード命令１～Ｍのアイデンティティをレガシーマシンのプロセッサから受信すると、ブロッククリエータ６０４は、キャッシュ１０２内のテーブルにアイデンティティを格納する。エミュレートされたＰＵコード命令１～Ｍの各々がコンパイルされると、ブロッククリエータ６０４は、エミュレートされたＰＵコード命

令１～Ｍのうちのどれがコンパイルされているかを識別する指示を含むようにテーブルを更新する。ブロッククリエータ６０４は、エミュレートされたＰＵコード命令１～Ｍのうちのどれがコンパイルされているかを識別する指示から、エミュレートされたＰＵコード命令１～Ｍのすべてがコンパイルされているかどうかを判定する。

【０１１６】

ゲームコードｇｃＮのエミュレートされたＰＵコード命令１～Ｍのすべてがコンパイルされていると判定すると、ブロッククリエータ６０４は、ゲームコードｇｃＮをメモリデバイス４１２から削除するコマンドを命令リムーバ７０２（図７Ａ）に送信する。コマンドを受信すると、方法７２０の操作７２４において、命令リムーバ７０２は、メモリデバイス４１２からエミュレートされたＰＵコード命令１～Ｍを消去する。

【０１１７】

一方で、ゲームコードｇｃＮのエミュレートされたＰＵコード命令１～Ｍのうちの１つ以上がテーブルに基づいてコンパイルされていないと判定すると、方法７２０の操作７２６において、ブロッククリエータ６０４は、命令リムーバ７０２へコマンドを送信せず、メモリデバイス４１２からゲームコードｇｃＮを削除する。ゲームコードｇｃＮを削除するコマンドを受信するまで、命令リムーバ７０２は、メモリデバイス４１２からエミュレートされたＰＵコード命令１～Ｍを削除しない。

【０１１８】

一実施形態では、ブロッククリエータ６０４は、キャッシュ１０２に格納されているゲームコードｇｃＮが、ゲームコードｇｃＮがメモリデバイス４１２からアクセスされる最も遅い時間から所定の期間内に、メモリデバイス４１２（図４Ａ）からアクセスされるかどうかを判定する。一例として、ブロッククリエータ６０４は、コンピュータネットワーク４０８を介してインターネットクロックにアクセスして、最も遅い時間と所定の期間を判定することができる。別の例として、ブロッククリエータ６０４は、最も遅い時間と所定の期間をカウントするために、クロック発振器などのクロックソースを含む。ゲームコードｇｃＮが所定の期間内にメモリデバイス４１２からアクセスされないと判定すると、ブロッククリエータ６０４は、命令リムーバ７０２にコマンドを送信し、メモリデバイス４１２からゲームコードｇｃＮを削除する。一例として、ゲームコードｇｃＮが所定の期間内にアクセスされないとき、エミュレートされたＰＵコード命令１～Ｍの大部分は、基本ブロックコンパイラ１０４によってアクセスされて、基本ブロック１～ｎのうちの１つ以上をコンパイルする。別の例として、ゲームコードｇｃＮが所定の期間内にアクセスされないとき、ゲームタイトルＧＮを有するレガシーゲームＮのプレイ中に頻繁に使用されるエミュレートされたＰＵコード命令１～Ｍのうちのいくつかは、基本ブロックコンパイラ１０４によってアクセスされ、基本ブロック１～ｎのうちの１つ以上をコンパイルする。一方で、ゲームコードｇｃＮが所定の期間内にアクセスされたと判定すると、ブロッククリエータ６０４は、コマンドを命令リムーバ７０２に送信せず、ゲームコードｇｃＮは、メモリデバイス４１２から削除されない。

【０１１９】

図８Ａは、基本ブロックの検証を説明するためのエミュレーションプロセッサシステム８００の実施形態の図である。エミュレーションプロセッサシステム８００は、エミュレーションプロセッサシステム４０９の一例である（図４Ｂ）。エミュレーションプロセッサシステム８００は、基本ブロックコンパイラ１０４、キャッシュ１０２、ブロックバリデータ８０２、及びブロックフラッガ８０４を含む。ブロックバリデータ８０２は、ブロッククリエータ６０４、ブロックディスパッチャ３０２、パーサ６０２、及びブロックフラッガ８０４に結合されている。クロックバリデータ８０２及びブロックフラガー８０４のそれぞれは、キャッシュ１０２に結合されている。

【０１２０】

一例として、ブロックバリデータ８０２は、ＡＳＩＣ、またはＰＬＤ、またはマイクロコントローラ、またはプロセッサ、またはコンピュータプログラム、またはコンピュータプログラムの一部分として実装される。また、一例として、ブロックフラガー８０４は、

ＡＳＩＣ、またはＰＬＤ、またはマイクロコントローラ、またはプロセッサ、またはコンピュータプログラム、またはコンピュータプログラムの一部分として実装される。

【０１２１】

一実施形態では、ＡＳＩＣ、ＰＬＤ、マイクロコントローラ、マイクロプロセッサ、コントローラ、及びプロセッサという用語は、本明細書では互換的に使用されている。

【０１２２】

図８Ｂは、エミュレーションプロセッサシステム８００（図８Ａ）によって実行される検証操作を説明するための方法８５０の一実施形態のフローチャートである。方法８００の操作８５２において、ブロックバリデータ８０２（図８Ａ）は、基本ブロックｎがコンパイルされているかどうかを判定する。例えば、ブロックバリデータ８０２は、キャッシュ１０２にアクセス（読み取るなど）して、以前にキャッシュ１０２内に格納された基本ブロック１～ｎ－１と比較して、基本ブロックｎなどの新しい基本ブロックが、キャッシュ１０２に格納されているかどうかを判定する。ブロックバリデータ８０２は、キャッシュ１０２内に格納された基本ブロック１～ｎ－１を以前に識別した。基本ブロックｎがキャッシュされていないと判定すると、ブロックバリデータ８０２は、キャッシュ１０２へのアクセスを継続し、新しい基本ブロックがキャッシュされているかどうかを判定する。例えば、ブロックバリデータ８０２は、キャッシュ１０２内に格納されている基本ブロック１～ｎ－１を定期的に解析して、基本ブロックｎがキャッシュ１０２に格納されているかどうかを判定する。

【０１２３】

基本ブロックｎがキャッシュされていることを判定すると、方法８００の操作８５４において、ブロックバリデータ８０２は、そこから基本ブロックｎがコンパイルされる、エミュレートされたＰＵコード命令１～Ｍのうちの１つ以上からの第１のハッシュ値を生成する。例えば、ブロックバリデータ８０２は、パーサ６０２に要求を送信し、エミュレートされたＰＵコード命令１～Ｍのうちの１つ以上をメモリデバイス４１２から取得する。要求は、キャッシュ１０２内の基本ブロックｎのメモリアドレスのアイデンティティを含む。基本ブロックｎのメモリアドレスのアイデンティティは、要求によって取得されるなど、ブロックバリデータ６０２によってブロッククリエータ６０４から受信される。ブロックバリデータ８０２からの要求を受信すると、パーサ６０２は、メモリデバイス４１２からエミュレートされたＰＵコード命令１～Ｍのうちの１つ以上を読み取り、エミュレートされたＰＵコード命令１～Ｍのうちの１つ以上をブロックバリデータ８０２に提供する。説明すると、エミュレートされたＰＵコード命令１～Ｍのうちの１つ以上が、基本ブロックｎをコンパイルするために基本ブロックコンパイラ１０４に提供されるとき、パーサ６０２は、キャッシュ１０２内の基本ブロックｎによって占有された１つ以上のメモリアドレスの１つ以上のアイデンティティと、メモリデバイス４１２内のエミュレートされたＰＵコード命令１～Ｍのうちの１つ以上によって占有された１つ以上のメモリアドレスの１つ以上のアイデンティティとの間の１対１の対応を格納する。パーサ６０２は、ブロックバリデータ８０２からキャッシュ１０２内の基本ブロックｎのメモリアドレスのアイデンティティを受信し、１対１の対応からメモリデバイス４１２内のエミュレートされたＰＵコード命令１～Ｍのうちの１つ以上の１つ以上のメモリアドレスを識別する。パーサ６０２は、メモリデバイス４１２の１つ以上のメモリアドレスからエミュレートされたＰＵコード命令１～Ｍのうちの１つ以上を読み取り、エミュレートされたＰＵコード命令１～Ｍのうちの１つ以上をブロックバリデータ８０２に提供する。

【０１２４】

この例で続けると、エミュレートされたＰＵコード命令１～Ｍのうちの１つ以上をパーサ６０２から受信すると、ブロックバリデータ８０２は、エミュレートされたＰＵコード命令１～Ｍのうちの１つ以上から第１のハッシュ値を生成し、キャッシュ１０２内に第１のハッシュ値を格納する。説明すると、ブロックバリデータ８０２は、基本ブロックｎに対応するエミュレートされたＰＵコード命令１～Ｍのうちの１つ以上からダイジェストまたはチェックサムを生成する。この例では、第１のハッシュ値は、基本ブロックｎを含む

ようにキー設定された、キャッシュ１０２の１つ以上のレジスタに格納される。
【０１２５】
方法８００の操作８５６において、ブロックバリデータ８０２は、コマンドをパーサ６０２に送信して、１つ以上のエミュレートされたＰＵコード命令１～Ｍが格納されているメモリデバイス４１２内で、１つ以上のメモリアドレスが上書きされているかどうかを判定する。メモリデバイス４１２内の１つ以上のメモリアドレスは、レガシーマシンのコンパクトディスク読み取り専用メモリ（ＣＤ－ＲＯＭ）からのデータで上書きされることができる。データは、エミュレートされたＰＵコード命令１～Ｍのうちの１つ以上への更新であることができる。エミュレートされたＰＵコード命令１～Ｍのうちの１つ以上への更新は、更新された命令と呼ばれる。あるいは、データは、破損している場合があるが、これは更新された命令ではない。パーサ６０２は、ブロックバリデータ８０２からコマンドを受信すると、要求をレガシーマシンのプロセッサに送信して、エミュレートされたＰＵコード要求１～Ｍのうちの１つ以上が格納されているメモリデバイス４１２内で、１つ以上のメモリアドレスが上書きされているかどうかを判定する。
【０１２６】
パーサ６０２から要求を受信すると、レガシーマシンのプロセッサは、要求に対する応答を提供し、応答は、１つ以上のメモリアドレスが、エミュレートされたＰＵコード命令１～Ｍのうちの１つ以上が格納されているメモリデバイス４１２内で上書きされているかどうかを示す。エミュレートされたＰＵコード命令１～Ｍのうちの１つ以上が格納されているメモリデバイス４１２内の１つ以上のメモリアドレスが上書きされていないという応答を受信すると、ブロックバリデータ８０２は、コマンドをブロックフラッガ８０４（図８Ａ）に送信し、基本ブロックｎを無効としてマークしないようにする。コマンドを受信すると、方法８００の操作８５８において、ブロックフラッガ８０４は、基本ブロックｎを無効としてマークしない。
【０１２７】
一方で、エミュレートされたＰＵコード命令１～Ｍのうちの１つ以上が格納されているメモリデバイス４１２内の１つ以上のメモリアドレスが、上書きされているという応答を受信すると、方法８５０の操作８６０において、ブロックバリデータ８０２は、コマンドをブロックフラッガ８０４（図８Ａ）に送信し、基本ブロックｎを無効としてマークする。基本ブロックｎを無効としてマークするコマンドを受信すると、ブロックフラッガ８０４は、基本ブロックｎを無効としてマークする。例えば、ブロックフラッガ８０４は、キャッシュ１０２内の基本ブロックｎにアクセスし、基本ブロックｎを有するキャッシュ１０２内のメモリアドレス内に、基本ブロックｎが無効であることを示すために無効マークｎ（図５Ａ）などの識別子を含む。無効なマークｎの一例は、ビットのシーケンスである。
【０１２８】
一実施形態では、ハッシュ値は検証結果の一例である。
【０１２９】
図８Ｃは、図８Ｂの方法８００のフローチャートの続きである。方法８００の操作８６２において、ブロックバリデータ８０２（図８Ａ）は、基本ブロックｎが実行されるべきかどうかを判定する。例えば、ブロックディスパッチャ３０２は、基本ブロックｎが実行されるべきであるという指示をブロックバリデータ８０２に送信する。基本ブロックｎが実行されるべきであるという指示をブロックディスパッチャ３０２から受信すると、ブロックバリデータ８０２は、基本ブロックｎが実行されるべきであると判定する。指示が受け取られるまで、ブロックバリデータ８０２は、基本ブロックｎが検証されるべきであると判定しない。
【０１３０】
基本ブロックｎが実行されるべきであると判定すると、ブロックバリデータ８０２は、方法８００の操作８６４において、基本ブロックｎが無効としてマークされているかどうかを判定する。例えば、ブロックバリデータ８０２は、基本ブロックｎが無効としてフラ

グが立てられているかどうかを判定するために、ブロックフラッガ８０４にコマンドを送信する。コマンドを受信すると、ブロックフラッガ８０４は、基本ブロックｎが無効としてフラグが立てられているかどうかの指示をブロックバリデータ８０２に送信する。説明すると、ブロックフラッガ８０４は、基本ブロックｎにアクセスして、基本ブロックｎが無効マークｎを含むかどうかを判定して、無効の指示を生成して、ブロックバリデータ８０２に送信する。別の説明として、ブロックフラッガ８０４は、基本ブロックｎにアクセスして、基本ブロックｎが無効マークｎを含まないかどうかを判定して、無効の欠如の指示を生成し、ブロックバリデータ８０２に送信する。ブロックフラッガ８０４から、基本ブロックｎが無効としてマークされているという指示を受信すると、ブロックバリデータ８０２は、基本ブロックｎが無効であると判定する。一方で、基本ブロックｎに無効マークｎがないという指示をブロックフラッガ８０４から受信すると、ブロックバリデータ８０２は、基本ブロックｎが無効としてマークされていないと判定する。

【０１３１】

基本ブロックｎが無効としてマークされていないと判定すると、方法８００の操作８６６において、基本ブロックｎが実行される。例えば、操作８６６は操作６６２（図６Ｂ）に類似している。説明すると、ブロックバリデータ８０２は、基本ブロックｎを実行するためにブロックディスパッチャ３０２にコマンドを送信する。基本ブロックｎを実行するコマンドを受信すると、ブロックディスパッチャ３０２は基本ブロックｎを実行する。この説明では、基本ブロックｎを実行するためのコマンドがブロックバリデータ８０２から受信されるまで、ブロックディスパッチャ３０２は基本ブロックｎを実行しない。

【０１３２】

一方で、基本ブロックｎが無効としてマークされているとの判定に応答して、方法８００の操作８６６において、ブロックバリデータ８０２は、基本ブロックｎが実際に有効であるかどうかを判定する。例えば、基本ブロックｎが無効としてマークされていると判定すると、ブロックバリデータ８０２は、基本ブロックｎがコンパイルされるエミュレートされたＰＵコード命令１～Ｍのうちの１つ以上が格納されているメモリデバイス４１２内の同じメモリアドレスに格納された更新された命令または破損したデータのいずれかから第２のハッシュ値を生成する。この例では、第２のハッシュ値は、基本ブロックｎを含むようにキー設定された、キャッシュ１０２の１つ以上のレジスタに格納される。説明すると、ブロックバリデータ８０２は、パーサ６０２に要求を送信して、メモリデバイス４１２内の１つ以上のメモリアドレスに格納された更新された命令または破損したデータを取得する。パーサ６０２は、要求を受信すると、メモリデバイス４１２からの１つ以上のメモリアドレスから更新された命令または破損したデータを読み取り、更新された命令または破損したデータをブロックバリデータ８０２に提供する。この図では、ブロックバリデータ８０２は、更新された命令または破損したデータから第２のハッシュ値を生成し、この第２のハッシュ値をキャッシュ１０２に格納する。例えば、ブロックバリデータ８０２は、更新された命令または破損したデータからダイジェストまたはチェックサムを生成する。この図では、ブロックバリデータ８０２は、第２のハッシュ値を第１のハッシュ値と比較して、基本ブロックｎが有効であるかどうかを判定する。比較に基づいて、第１のハッシュ値と第２のハッシュ値との間に一致があると判定すると、ブロックバリデータ８０２は、基本ブロックｎが有効であると判定する。この一致は、更新された命令から第２のハッシュ値が生成されたときに発生する。一方で、この比較に基づいて、第１のハッシュ値と第２のハッシュ値との間に一致がないと判定すると、ブロックバリデータ８０２は、基本ブロックｎが無効であると判定する。この一致の欠如は、破損したデータから第２のハッシュ値が生成されたときに発生する。

【０１３３】

基本ブロックｎが有効であるとの判定に応答して、方法８００の操作８７０において、無効なマークｎが除去される。例えば、ブロックバリデータ８０２は、ブロックフラッガ８０４にコマンドを送信して、キャッシュ１０２に格納された基本ブロックｎから無効なマークｎを除去する。ブロックバリデータ８０２からコマンドを受信すると、ブロックフ

ラッガ８０４は、キャッシュ１０２から基本ブロックｎにアクセスし、基本ブロックｎから無効なマークｎを消去する。説明すると、ブロックフラッガ８０４は、無効なマークｎが格納されているキャッシュ１０２の１つ以上のメモリアドレスから無効なマークｎを消去する。無効なマークｎを除去すると、ブロックフラッガ８０４は、基本ブロックｎを実行するためにブロックディスパッチャ３０２にコマンドを送信し、コマンドを受信すると、操作８６６において、ブロックディスパッチャ３０２は、基本ブロックｎを実行する。一例として、ブロックディスパッチャ３０２は、基本ブロックｎを実行するためのコマンドがブロックフラッガ８０４から受信されるまで、基本ブロックｎを実行しない。

【０１３４】

一方で、基本ブロックｎが有効でないと判定すると、方法８００の操作８７２において、基本ブロックｎと同じ機能または操作を有する追加の基本ブロックがコンパイルされる。例えば、ブロックバリデータ８０２は、基本ブロックｎを実行しないようにブロックディスパッチャ３０２にコマンドを送信し、基本ブロックｎが無効であるという指示をブロッククリエータ６０４に送信する。指示を受信すると、ブロッククリエータ６０４は、基本ブロックｎを再コンパイルする。説明すると、追加の基本ブロックが、追加のレガシーマシンのメモリデバイス内に格納された、エミュレートされたＰＵコード命令１～Ｍの１つ以上などのエミュレートされたＰＵコード命令の追加セットからコンパイルされることを除いて、ブロッククリエータ６０４は、メモリデバイス４１２に格納されたエミュレートされたＰＵコード命令１～Ｍのうちの１つ以上から基本ブロックｎをコンパイルするのと同じ方法で追加の基本ブロックをコンパイルする。追加のレガシーマシンは、メモリデバイス４１２を含むレガシーマシンとは異なっている。また、エミュレートされたＰＵコード命令の追加セットは、ゲームコードｇｃＮの追加セットと同じゲームタイトルＧＮのものである。別の説明として、追加の基本ブロックが、エミュレートされたＰＵコード命令１～Ｍが格納されている、メモリアドレスとは異なるメモリデバイス４１２のメモリアドレス内に格納された、エミュレートされたＰＵコード命令１～Ｍのうちの１つ以上などのメモリデバイス４１２のエミュレートされたＰＵコード命令の追加セットからコンパイルされることを除いて、ブロッククリエータ６０４は、メモリデバイス４１２に格納されたエミュレートされたＰＵコード命令１～Ｍのうちの１つ以上から基本ブロックｎをコンパイルするのと同じ方法で追加の基本ブロックをコンパイルする。さらに別の説明のように、追加の基本ブロックが、メモリデバイス４１２とは異なるメモリデバイス内に格納された、エミュレートされたＰＵコード命令１～Ｍの１つ以上などのエミュレートされたＰＵコード命令の追加セットからコンパイルされることを除いて、ブロッククリエータ６０４は、メモリデバイス４１２に格納されたエミュレートされたＰＵコード命令１～Ｍのうちの１つ以上から基本ブロックｎをコンパイルするのと同じ方法で追加の基本ブロックをコンパイルする。

【０１３５】

追加の基本ブロックは、実行のためにブロッククリエータ６０４からブロックディスパッチャ３０２に送信される。例えば、ブロックディスパッチャ３０２は、追加の基本ブロックを実行して、仮想環境４５２（図４Ｂ）などの追加の仮想環境の一部分を生成する。説明すると、ブロックディスパッチャ３０２は、追加の基本ブロックを実行して、画像フレームの一部を生成し、この画像フレームは、仮想オブジェクト４５４の位置及び方向、仮想オブジェクト４５４のパラメータ、仮想環境４５２内の他の仮想オブジェクトの位置及び方向、及び追加の仮想環境内の他の仮想オブジェクトのパラメータなどの追加の仮想環境データを含む。ブロックディスパッチャ３０２は、画像フレームをエミュレーションプロセッサシステム４０９のＧＰＵに提供して、ディスプレイデバイス４１０のディスプレイ画面上に追加の仮想環境をレンダリングするなどして表示する。

【０１３６】

方法８５０は、基本ブロックｎ及び追加の基本ブロックを参照して示されているが、方法８５０は、追加の基本ブロックが生成されるのと同じ方法で生成される他の基本ブロック１～（ｎ－１）及びより多くの追加の基本ブロックに等しく適用可能であることに留意

されたい。

【０１３７】

一実施形態では、エミュレートされたＰＵコード命令１～Ｍのセットは、レガシーマシンの外部に配置された１つ以上のメモリデバイスの第１のセットに格納され、エミュレートされたＰＵコード命令の追加のセットは、追加のレガシーマシンの外部に配置された１つ以上のメモリデバイスの第２のセットに格納されている。

【０１３８】

図９Ａは、レガシーマシン９００の実施形態を説明するための図である。レガシーマシン９００は、レガシーＣＰＵ９０２、レガシーＧＰＵ９０４、レガシーＡＰＵ９１０、メモリデバイス９０６、及びＣＤ－ＲＯＭドライブ９０８を含む。メモリデバイス９０６は、メモリデバイス４１２（図４Ａ）の一例である。レガシーＣＰＵ９０２の一例は、１クロックサイクル中に最大３２ビットを処理することができる３２ビットＣＰＵである。また、レガシーＧＰＵ９０４の一例は、１クロックサイクル中に最大３２ビットを処理できる３２ビットＧＰＵである。レガシーＡＰＵ９１０の一例は、１クロックサイクル中に最大１６ビットを処理して１つ以上のレガシーオーディオパラメータを生成できる１６ビットＡＰＵである。メモリデバイス９０６の一例は、２メガバイト（ＭＢ）ＲＡＭである。

【０１３９】

レガシーＣＰＵ９０２、レガシーＡＰＵ９１０、及びレガシーＧＰＵ９０４は、ＣＤ－ＲＯＭドライブ９０８に結合されているメモリデバイス９０６に結合されている。エミュレートされたＰＵコード１０６は、メモリデバイス９０６内に格納されている。

【０１４０】

レガシーＣＰＵ９０２、レガシーＡＰＵ９１０、またはレガシーＧＰＵ９０４は、メモリデバイス９０６からエミュレートされたＰＵコード１０６にアクセスし、エミュレートされたＰＵコード１０６を処理する。ＣＤ－ＲＯＭドライブ９０８は、更新された命令または破損したコードを含むＣＤ－ＲＯＭを受け入れる。更新された命令または破損したコードは、レガシーＣＰＵ９０２によってＣＤ－ＲＯＭからメモリデバイス９０６に転送され得る。

【０１４１】

一実施形態では、レガシーマシン９００はキャッシュを含まない。一実施形態では、レガシーマシン９００は、４キロバイト（ＫＢ）のキャッシュなどの限られた容量のキャッシュを含む。

【０１４２】

一実施形態では、レガシーマシン９００は、現在、会社または企業などのエンティティによって製造されていないが、過去にエンティティによって製造された。

【０１４３】

図９Ｂは、更新されたマシン９２０の実施形態を説明するための図である。更新されたマシン９２０は、ＣＰＵ９２２、ＧＰＵ９２４、ＡＰＵ９３１、メモリシステム９２６、及びキャッシュ１０２（図１）の一例であるキャッシュ９２８を含む。ゲームコンソール４０２（図４Ｂ）は、更新されたマシン９２０の一例である。一例として、ＣＰＵ９２２は２つのクアッドコアモジュールを含み、各モジュールは、各クロックサイクル中に６４ビットを処理することができる。各コアは、３２キロバイト（ＫＢ）のキャッシュを有する。ＣＰＵ９２２の別の例は、１クロックサイクル中に最大６４ビットを処理できる６４ビットＣＰＵである。一例として、ＧＰＵ９２４は１１５２コアを有し、各コアは１クロックサイクル中に６４ビットを処理できる。別の例として、レガシーＧＰＵ９２４は、６４ビットＧＰＵであり、１クロックサイクル中に最大６４ビットを処理できる。さらに別の例として、ＡＰＵ９３１は、１６ビットＡＰＵであり、１クロックサイクル中に最大１６ビットを処理できる。さらに別の例として、ＡＰＵ９３１は、２４ビットＡＰＵであり、１クロックサイクル中に最大２４ビットを処理できる。さらに別の例として、ＡＰＵ９３１は、３２ビットＡＰＵであり、１クロックサイクル中に最大３２ビットを処理できる。

【０１４４】
ＣＰＵ９２２、ＧＰＵ９２４、及びＡＰＵ９３１は、メモリシステム９０６に結合されている。一例として、エミュレートされたＰＵコード１０６は、レガシーメモリシステム９０６内に格納されている。メモリシステム９２６が含むものの一例は、５００ギガバイト（ＧＢ）または２テラバイト（ＴＢ）のストレージを提供するハードドライブである。ＣＰＵ９２２、ＧＰＵ９２４、キャッシュ９２８、及びメモリシステム９２６は、バス９３０を介して互いに結合されている。
【０１４５】
ＣＰＵ９２２またはＧＰＵ９２４は、メモリシステム９２６にアクセスする前に、最初にキャッシュ９２８にアクセスする。キャッシュ９２８が、ＣＰＵ９２２またはＧＰＵ９２４によって要求されたデータを含まないと判定すると、ＣＰＵ９２２またはＧＰＵ９２４またはＡＰＵ９３１は、メモリシステム９２６にアクセスする。
【０１４６】
一実施形態では、ゲームコードｇｃＮは、ＣＰＵ９２２またはＧＰＵ９２４またはＡＰＵ９３１によって実行されることはできないが、レガシーＣＰＵ９０２またはレガシーＧＰＵ９０４によって実行されることができることに留意されたい。また、基本ブロック１～ｎは、ＣＰＵ９２２またはＧＰＵ９２４またはＡＰＵ９３０によって実行され得るが、レガシーＣＰＵ９０２またはレガシーＧＰＵ９０４またはレガシーＡＰＵ９１０によっては実行され得ない。
【０１４７】
一実施形態では、キャッシュ９２８は、ＣＰＵ９２２内に位置する。
【０１４８】
一実施形態では、キャッシュ９２８は、ＧＰＵ９２４内に位置する。
【０１４９】
一実施形態では、キャッシュ９２８は、ＡＰＵ９３１内に位置する。
【０１５０】
一実施形態では、キャッシュ９２８は、ＣＰＵ９２２内に位置し、キャッシュ１０２などの別のキャッシュは、ＧＰＵ９２４内に位置する。
【０１５１】
一実施形態では、更新されたマシン９２０は、現在、エンティティによって製造されているか、またはエンティティによって最近製造された。例えば、更新されたマシン９２０は、レガシーマシン９００の製造の数年後にエンティティによって製造されている。
【０１５２】
図１０Ａは、基本ブロックコンパイラ１０４（図１）による複数の基本ブロックの１つの基本ブロックへの組み合わせを説明するためのシステム１０００の実施形態の図である。ブロッククリエータ６０４（図６Ａ）は、仮想オブジェクト４５４（図４Ｂ）が、ゲームタイトルＧＮを有するレガシーゲームＮのレベル１において、位置Ｐ１及び方向Ｏ１で開始されることを示す信号などのユーザ入力１を受信する。ユーザ入力１を受信すると、基本ブロック１は、エミュレートされたＰＵコード命令１に基づいて基本ブロックコンパイラ１０４によって生成され、ユーザ入力１を処理する。同様に、ブロッククリエータ６０４は、仮想オブジェクト４５４がレガシーゲームＮのレベル２で位置Ｐ１及び方向Ｏ１で開始されることを示す信号などのユーザ入力２を受信する。ユーザ入力２を受信すると、基本ブロック２は、エミュレートされたＰＵコード命令２に基づいて基本ブロックコンパイラ１０４によって生成され、ユーザ入力２を処理する。
【０１５３】
仮想オブジェクト４５４がレガシーゲームＮのレベル３で位置Ｐ１と方向Ｏ１で開始されることを示す信号などのユーザ入力３が受信されると、ブロックコンパイラ１０４は、基本ブロック１及び２をユーザ入力３の処理として識別する。そのように識別すると、ブロッククリエータ６０４は、基本ブロック１及び２を、基本ブロック１または基本ブロック２などの単一の基本ブロックに組み合わせるなどして統合する。この統合は、キャッシ

ュ１０２内のメモリスペースを節約し、かつ、基本ブロック１及び基本ブロック２にアクセスする代わりに、単一の基本ブロックにアクセスする際の効率も高める。

【０１５４】

ブロッククリエータ６０４はさらに、キャッシュ１０２のポインタに値を生成する。本明細書で使用されるポインタの例は、レジスタである。ポインタは、基本ブロック１または基本ブロック２を処理するためのユーザ入力が受信されたときに、単一の基本ブロックを実行するようにブロックディスパッチャ３０２に示す。ブロックディスパッチャ３０２が基本ブロック１または２を実行するとき、ブロックディスパッチャ３０２は、ポインタにアクセスし、基本ブロック１または２の代わりに単一の基本ブロックを実行する。

【０１５５】

図１０Ｂは、基本ブロック１～ｎのうちの１つ以上の修正を説明するためのシステム１０２０の実施形態の図である。システム１０２０は、ブロックインターフェース１０２２、キャッシュ１０２、及びブロックディスパッチャ３０２を含む。一例として、基本ブロックｎは、推測航法の操作ｎが含まれる。別の例として、基本ブロックｎは、ディスプレイデバイス４１０（図４Ｂ）のディスプレイ画面の端から十字線を跳ね返す操作ｎを含む。ブロックインターフェース１０２２の例は、ＡＳＩＣ、またはＰＬＤ、またはマイクロプロセッサ、またはマイクロコントローラ、またはコンピュータプログラム、またはコンピュータプログラムの一部分である。ブロックインターフェース１０２２は、キャッシュ１０２に結合されている。

【０１５６】

ブロックインターフェース１０２２は、キャッシュ１０２に格納された１つ以上の基本ブロック１～ｎへのアクセスをユーザに提供する。例えば、ユーザは、キーボード、マウス、またはキーパッドなどの入力デバイス上の１つ以上のボタンを選択して、修正入力１０２４を生成する。一例として、修正入力１０２４は、基本ブロックｎを修正するためのソースコードの形式の１つ以上のユーザ命令を含む。説明すると、修正入力１０２４は、基本ブロックｎから推測航法の操作ｎを除去するためのユーザ命令を含む。別の例として、修正入力１０２４は、十字線がディスプレイデバイス４１０のディスプレイ画面の端から滑り落ち、ディスプレイデバイス４１０のディスプレイ画面の反対側の端に表示されることを含むように操作ｎを変更する命令を含む。反対側のエッジは、十字線が滑り落ちる端と対角線上にある。

【０１５７】

入力デバイスは、ブロックインターフェース１０２２に結合されている。修正入力１０２４は、入力デバイスからブロックインターフェース１０２２に送信されて、基本ブロックｎを修正して、修正された基本ブロックｎを出力する。一例として、修正された基本ブロックｎは、推測航法の操作を含んでいない。別の例として、修正された基本ブロックｎは、十字線を端から反対側の端にスライドさせる操作ｎを含む。さらに別の例として、修正された基本ブロックｎは、操作ｎの実行のクロックサイクル数を計算し、その数を基本ブロックｎに格納する操作を含む。説明すると、サイクル数は、基本ブロック１が格納されているキャッシュ１０２のメモリアドレスに格納されている。クロックサイクル数を計算する操作は、操作ｎを実行することに加えて、ブロックディスパッチャ３０２（図３）によって実行される。ブロックディスパッチャ３０２は、操作ｎの実行時にクロックサイクル数を計算し、その数を基本ブロックｎに格納する。

【０１５８】

ブロッククリエータ６０４はさらに、キャッシュ１０２内のポインタに値を生成し、ポインタは、基本ブロックｎを処理するためのユーザ入力が受信されたときに、変更された基本ブロックｎを実行するようにブロックディスパッチャ３０２に示す。ブロックディスパッチャ３０２が基本ブロックｎを実行するとき、ブロックディスパッチャ３０２はポインタにアクセスし、基本ブロックｎの代わりに変更された基本ブロックｎを実行する。

【０１５９】

図１０Ｃは、サブルーチンに基づいて作成された基本ブロックと、サブルーチンを呼び

出すエミュレートされたＰＵコード命令に基づいて生成された基本ブロックとを組み合わせることを示すシステム１０３０の実施形態の図である。ブロッククリエータ６０４（図６Ａ）は、仮想オブジェクト４５４の破壊になる仮想オブジェクト４５４（図４Ｂ）の位置及び／または方向の変化を示す信号などのユーザ入力１を受信する。ユーザ入力１を受信すると、基本ブロック１は、エミュレートされたＰＵコード命令１に基づいて基本ブロックコンパイラ１０４によって生成され、ユーザ入力１を処理する。エミュレートされたＰＵコード命令１は、サブルーチン１への関数呼び出しを含む。このように、ユーザ入力１を受信すると、基本ブロック２は、エミュレートされたＰＵコード命令２の一例であるサブルーチン１に基づいて、基本ブロックコンパイラ１０４によって生成される。一例として、基本ブロック２は、位置Ｐ１及び方向Ｏ１で仮想オブジェクト４５４を再生成する操作２を含む。

【０１６０】

仮想オブジェクト４５４の破壊になる、仮想オブジェクト４５４（図４Ｂ）の位置及び／または方向の変化を示す信号などのユーザ入力２が受信されるとき、ブロックコンパイラ１０４は、ユーザ入力２を処理するものとして基本ブロック１及び２を識別する。そのように識別すると、ブロッククリエータ６０４は、基本ブロック１及び２を、基本ブロック１または基本ブロック２などの単一の基本ブロックに組み合わせるなどして統合する。例えば、ユーザ入力２を受信すると、仮想オブジェクト４５４の破壊及び仮想オブジェクト４５４の再生成がトリガーされる。基本ブロック１と２を組み合わせると、サブルーチン１はスキップされる。

【０１６１】

ブロッククリエータ６０４はさらに値を生成し、その値をキャッシュ１０２内のポインタに格納する。この値は、基本ブロック１及び２を処理するためのユーザ入力が受信されたときに、単一の基本ブロックを実行するようにブロックディスパッチャ３０２に示す。ブロックディスパッチャ３０２が、基本ブロック１及び２を実行するためであるときに、ブロックディスパッチャ３０２は、ポインタにアクセスし、基本ブロック１の代わりに単一の基本ブロックを実行する。

【０１６２】

図１０Ｄは、２つの基本ブロックの間への基本ブロックの挿入を示すシステム１０４０の実施形態の図である。システム１０４０は、ブロックインターフェース１０２２及びキャッシュ１０２を含む。基本ブロック１及び２がキャッシュ１０２に格納された後、修正入力１０４２が、入力デバイスを介してユーザから受信される。一例として、修正入力１０４２は、ソースコードを含む信号であり、ソースコードは、基本ブロック１．１を定義し、これは、キャッシュ１０２内のソースレジスタアドレス１．１、キャッシュ１０２内の宛先レジスタアドレス１．１、及びソースレジスタアドレス１．１内に格納され、または宛先レジスタアドレス１．１上に格納され、またはソースレジスタアドレスと宛先レジスタアドレス１．１の両方に格納されているデータに実行される操作１．１を含む。操作１．１の例には、ジャンプ、格納、ロード、分岐、及び算術操作が含まれる。

【０１６３】

修正入力１０４２は、基本ブロック１．１の場所の指示、及び基本ブロック１．１と基本ブロック１または２との関連付けをさらに含む。例えば、修正入力１０４２は、基本ブロック１．１がキャッシュ１０２内の基本ブロック１と２の間に挿入されることを示す信号と、基本ブロック１．１が基本ブロック１及び／または２にリンクされることを示す信号と、を含む。説明すると、基本ブロック１．１は、基本ブロック２の無効なマーク２を受信するために、１つ以上のレジスタアドレスなどのキャッシュ１０２内の場所を含むために挿入される。別の説明として、基本ブロック１．１は、基本ブロック２の実行のサイクル数を受信するために、キャッシュ１０２内の１つ以上のレジスタアドレスなどの場所を含むために挿入される。この説明では、ブロックインターフェース１０２２は、修正入力１０４２を受信し、基本ブロック１．１が基本ブロック１と２の間に挿入される修正入力１０４２から識別し、基本ブロック１と２との間に基本ブロック１．１を挿入する。別

の例として、ブロックインターフェース１０２２は、基本ブロック１．１が、ゲームタイトルＧＮを有するレガシーゲームＮのレベルをロック解除するための操作を含むことを判定する。レベルは、基本ブロック１の操作で識別される第１のレベルと、基本ブロック２の操作で識別される第２のレベルとの間にある。第１レベルと第２レベルは、ゲームタイトルＧＮを有するレガシーゲームＮにある。第１レベルと第２レベルの間に挿入されたレベルは、ゲームコードｇｃＮの一部ではないが、レガシーゲームＮの新しいレベルである。第１のレベルと第２のレベルとの間に挿入されるレベルの例は、仮想オブジェクト４５４がミサイルの代わりにレーザ銃を発射するレベルである。

【０１６４】

さらに、ブロックインターフェース１０２２は、修正入力１０４２からポインタの値を識別し、その値をキャッシュ１０２に格納する。一例として、ポインタの値は、基本ブロック１．１の実行が、基本ブロック２の実行の直前に発生することを示す。ブロックディスパッチャ３０２が基本ブロック２を実行するとき、ブロックディスパッチャ３０２は、基本ブロック１．１を指すポインタ内の値を識別し、基本ブロック２を実行する直前に基本ブロック１．１を実行する。別の例として、ポインタの値は、基本ブロック１．１の実行が、基本ブロック２を実行する直後に発生することを示している。ブロックディスパッチャ３０２が基本ブロック２を実行した後、ブロックディスパッチャ３０２は、基本ブロック１．１を指すポインタ内の値を識別し、基本ブロック２を実行した直後に基本ブロック１．１を実行する。

【０１６５】

図１０Ｅは、基本ブロックの実行の順序での切り替えを示すシステム１０５０の実施形態の図である。システム１０５０は、キャッシュ１０２及びブロックインターフェース１０２２を含む。キャッシュ１０２は、キャッシュ１０２のポインタ内に値を含み、その値は、基本ブロック１の実行後に基本ブロック２が実行されることを示している。ユーザは、入力デバイスを使用して、基本ブロック１及び２の実行順序が切り替えられることを示す信号を含む修正入力１０５２を提供する。信号を受信すると、ブロックインターフェース１０２２は、キャッシュ１０２内のポインタの値を変更して、基本ブロック２を実行した後に基本ブロック１が実行されることを示す。

【０１６６】

図１１Ａは、基本ブロックｎに格納されたサイクル数の実際のカウントの使用を説明するための方法１１００の実施形態のフローチャートである。方法１１００は、基本ブロックｎを実行する操作６６２を含む。方法１１００の操作１１０２において、ブロックディスパッチャ３０２（図３）は、操作６６２において、基本ブロックｎの実行のサイクル数をカウントして、第１のカウントを生成する。ブロックディスパッチャ３０２は、方法１１００の操作１１０４において、基本ブロックｎに第１のカウントを格納する。例えば、ブロックディスパッチャ３０２は、基本ブロックｎに割り当てられた、キャッシュ１０２内のレジスタアドレスのうちの１つを有するレジスタに実数を書き込む。

【０１６７】

方法１１００の操作１１０６において、ブロッククリエータ６０４（図６Ａ）は、同じ基本ブロックｎが再度実行されるかどうかを判定する。例えば、ブロッククリエータ６０４は、同じ基本ブロックｎを処理するためのユーザ入力が、クライアントデバイスから受信されるかどうかを判定する。ブロッククリエータ６０４は、ユーザ入力が受信されるまで、同じ基本ブロックｎを処理するためのユーザ入力が受信されるかどうかを判定し続ける。

【０１６８】

図１１Ｂは、図１１Ａの方法１１００のフローチャートの続きである。基本ブロックｎを処理するためのユーザ入力が再び受信されたと判定すると、基本ブロックｎは、ブロックディスパッチャ３０４によって再び実行される。方法１１００の操作１１０８では、ブロックディスパッチャ３０４は、操作１１０６において、基本ブロックｎの実行のサイクル数をカウントして、第２のカウントを計算する。

【０１６９】
ブロックディスパッチャ３０４は、第２のカウントが、方法１１００の操作１１０８における第１のカウントからの所定の制限内にあるかどうかを判定する。所定の制限は、キャッシュ１０２に格納される。第２のカウントが第１のカウントからの所定の制限内にないと判定したことに応答して、方法１１００の操作１１１０において、ブロックディスパッチャ３０４は通知を生成する。例えば、ブロックディスパッチャ３０４がゲームコンソール４０２（図４Ａ）内に位置するとき、ゲームコンソール４０２のＧＰＵ９２４（図９Ｂ）は、クライアントデバイスのディスプレイデバイス４１０（図４Ａ）に通知の表現を表示し、サーバシステム４０４に通知するために、コンピュータネットワーク４０８を介してサーバシステム４０４に通知を送信する。別の例として、ブロックディスパッチャ３０４が、サーバシステム４０４（図４Ｂ）内に位置するとき、ブロックディスパッチャ３０４は通知を生成し、サーバシステム４０４のＧＰＵは、サーバシステム４０４に結合されたディスプレイデバイス上に通知の表現を表示する。一方で、第２のカウントが、第１のカウントから所定の制限内にあると判定すると、方法１１００の操作１１１２において、ブロックディスパッチャ３０４は通知をトリガーしない。

【０１７０】
図１２は、第１のクライアントデバイスから第２のクライアントデバイスへの基本ブロック１～ｎの転送を説明するためのシステム１２００の実施形態の図である。システム１２００は、ゲームコンソール４０２、コンピュータネットワーク４０８、サーバシステム４０４、ゲームコンソール１２０２、及びディスプレイデバイス１２０４を含む。ゲームコンソール１２０２は、ゲームコンソール４０２に類似している。例えば、ゲームコンソール４０２及び１２０２の両方は、更新されたマシンの例である。別の例として、ゲームコンソール１２０２はレガシーマシンではない。また、ディスプレイデバイス１２０４は、図４Ａのディスプレイデバイス４１０に類似している。例えば、ディスプレイデバイス４１０は、ＨＭＤまたはテレビまたはスマートテレビまたはコンピュータモニタである。

【０１７１】
ゲームコンソール４０２は、ネットワークインターフェースコントローラ１２１２を含む。ゲームコンソール１２０２は、エミュレーションプロセッサシステム１２０６、入力プロセッサシステム１２０７、キャッシュ１２０８、及びネットワークインターフェースコントローラ１２１０を含む。エミュレーションプロセッサシステム１２０６は、エミュレーションプロセッサシステム４０９と同じ構造及び同じ機能を有する。入力プロセッサシステム１２０７は、入力プロセッサシステム４０７（図４Ａ）と同じ構造及び同じ機能を有する。また、キャッシュ１２０８は、キャッシュ１０２と同じ構造及び同じ機能を有する。エミュレーションプロセッサシステム１２０６、入力プロセッサシステム１２０７、キャッシュ１２０８、及びネットワークインターフェースコントローラ１２１０は、バス１２１８を介して互いに結合されている。

【０１７２】
基本ブロック１～ｎがキャッシュ１０２に格納されると、エミュレーションプロセッサシステム４０９の基本ブロックコンパイラ１０４は、基本ブロック１～ｎをネットワークインターフェースコントローラ１２１２に送信する。ネットワークインターフェースコントローラ１２１２は、ネットワーク通信プロトコルを基本ブロック１～ｎに適用して、基本ブロック１～ｎを埋め込む１つ以上のパケットを生成し、コンピュータネットワーク４０８を介して１つ以上のパケットをサーバシステム４０４に送信する。１つ以上のパケットを受信すると、サーバシステム４０４のネットワークインターフェースコントローラは、ネットワーク通信プロトコルを１つ以上のパケットに適用して、ゲームタイトルＧＮを有するレガシーゲームＮの基本ブロック１～ｎを抽出し、基本ブロック１～ｎをサーバシステム４０４の１つ以上のメモリデバイス内に格納する。

【０１７３】
ユーザ２は、ハンドヘルドコントローラ１２１２を使用して、ハンドヘルドコントローラ１２１２上の１つ以上のボタンを選択して、サーバシステム４０４によってユーザ２に

割り当てられたユーザアカウント２にログインする。ユーザ２は、ユーザＩＤ２及びパスワードがサーバシステム４０４によって認証されると、ユーザアカウント２にログインする。ユーザ２がユーザアカウント２にログインすると、ユーザ２は、ゲームタイトルＧ１、ゲームタイトルＧａ、ゲームタイトルＧ２、そしてそのように続いてゲームタイトルＧＮまで、などの複数のゲームタイトルにアクセスすることができる。

【０１７４】

ユーザアカウント２にログインすると、ユーザ２は、ハンドヘルドコントローラ１２１２を使用して、ハンドヘルドコントローラ１２１２上の１つ以上のボタンを選択して、ユーザ入力１２１４を生成する。ユーザ入力１２１４は、ディスプレイデバイス１２０４上に表示されたタイトルＧＮを有するレガシーゲームＮの選択時に生成される。ユーザ入力１２１４が生成されるとき、キャッシュ１２０８は、基本ブロック１～ｎを含まない。例えば、ユーザ入力１２１４を受信するとエミュレーションプロセッサシステム１２０６の基本ブロックコンパイラは、キャッシュ１０２をチェックして、キャッシュ１０２が、ユーザ入力１２１４を処理するためのゲームコードＧＣＮの基本ブロック１～ｎのうちの１つ以上を含むかどうかを判定する。キャッシュ１０２が、ユーザ入力１２１４を処理するための基本ブロック１～ｎのうちの１つ以上を含まないと判定すると、エミュレーションプロセッサシステム１２０６は、基本ブロック１～ｎのうちの１つ以上に対する要求１２２０を生成し、ネットワークインターフェースコントローラ１２１０への要求１２２０を送信する。

【０１７５】

要求１２２０を受信すると、ネットワークインターフェースコントローラ１２１０は、ネットワーク通信プロトコルを要求１２２０に適用することによって要求１２２０を埋め込む１つ以上のパケットを生成し、コンピュータネットワーク４０８を介してサーバシステム４０４に１つ以上のパケットを送信する。サーバシステム４０４のネットワークインターフェースコントローラは、１つ以上のパケットを受信し、ネットワーク通信プロトコルを適用して、１つ以上のパケットから要求１２２０を抽出する。サーバシステム４０４のプロセッサは、要求１２２０を分析して、基本ブロック１～ｎが要求されていることを識別する。

【０１７６】

基本ブロック１～ｎが要求されたとの判定に応答して、サーバシステム４０４は、サーバシステム４０４の１つ以上のメモリデバイスに格納された基本ブロック１～ｎにアクセスして、基本ブロック１～ｎをサーバシステム４０４のネットワークインターフェースコントローラに提供する。サーバシステム４０４のネットワークインターフェースコントローラは、ネットワーク通信プロトコルを適用して、基本ブロック１～ｎを埋め込む１つ以上のパケットを生成して、コンピュータネットワーク４０８を介して１つ以上のパケットをゲームコンソール１２０２に送信する。

【０１７７】

ゲームコンソール１２０２のネットワークインターフェースコントローラ１２１０は、基本ブロック１～ｎを有する１つ以上のパケットを受信し、ネットワーク通信プロトコルを適用して、１つ以上のパケットから基本ブロック１～ｎを抽出し、基本ブロック１～ｎをエミュレーションプロセッサシステム１２０６に送信する。エミュレーションプロセッサシステム１２０６の基本ブロックコンパイラは、基本ブロック１～ｎをキャッシュ１２０８に格納する。

【０１７８】

ゲームタイトルＧＮを有するゲームのプレイ中にハンドヘルドコントローラ１２１２からユーザ入力１２２４を受信すると、入力プロセッサシステム１２７０は、入力プロセッサシステム４０７が、ユーザ入力４１９をユーザ入力４２０に変換するのと同じ方法で、ユーザ入力１２２４をユーザ入力１２２５に変換する。ユーザ入力１２２５を受信すると、エミュレーションプロセッサシステム１２０６の基本ブロックコンパイラは、ユーザ入力１２２５を処理するために、キャッシュ１２０８内の基本ブロック１～ｎのうちの１つ

以上を識別する。エミュレーションプロセッサシステム１２０６のブロックディスパッチャは、ユーザ入力１２２５を処理するために、基本ブロック１～ｎのうちの１つ以上を実行する。このようにして、基本ブロック１～ｎがエミュレーションプロセッサシステム４０９によってコンパイルされると、基本ブロック１～ｎは、エミュレーションプロセッサシステム１２０６によってコンパイルされる必要はないが、エミュレーションプロセッサシステム１２０６によって、サーバシステム４０４からアクセスされることができる。

【０１７９】

図１３は、レガシーゲームＮのプレイ中にハンドヘルドコントローラ４１４から受信されたユーザ入力４１９のエミュレーションを説明するためのシステム１３００の実施形態の図である。システム１３００は、入力プロセッサシステム４０７、エミュレーションプロセッサシステム４０９、ハンドヘルドコントローラ４１４、及びハンドヘルドコントローラ１３０６を含む。

【０１８０】

ハンドヘルドコントローラ１３０６は、レガシーハンドヘルドコントローラの一例である。一例として、ハンドヘルドコントローラ１３０６は、レガシーマシン９００（図９Ａ）と共に使用されることができるが、更新されたマシン９２０（図９Ｂ）と共に使用することはできない。説明すると、入力プロセッサシステム４０７は、入力プロセッサ１３１４を含み、これは、ハンドヘルドコントローラ１３０６によって生成され、ハンドヘルドコントローラ１３０６から受信されるユーザ入力を解釈することができない。さらに説明するために、入力プロセッサ１３１４は、６４ビットまたは１２８ビットで表されるユーザ入力４１９を解釈することができるが、ハンドヘルドコントローラ１３０６から受信した３２ビットを有するユーザ入力を解釈することはできない。別の例として、ハンドヘルドコントローラ１３０６は、レガシーコントローラの一例であり、ハンドヘルドコントローラ４１４は、更新されたコントローラの一例である。説明すると、ハンドヘルドコントローラ１３０６は、現在エンティティによって製造されておらず、ハンドヘルドコントローラ４１４は、現在エンティティによって製造されている。さらに説明するために、ハンドヘルドコントローラ４１４は最新バージョンであり、ハンドヘルドコントローラ１３０６は古いバージョンである。別のさらなる説明として、ハンドヘルドコントローラ１３０６は、ハンドヘルドコントローラ４１４がエンティティによって製造される数年前に、エンティティによって製造されている。さらに別の例として、ハンドヘルドコントローラ１３０６は、サムスティックを含まないが選択ボタンを含み、ハンドヘルドコントローラ４１４は選択ボタンを含まない。入力プロセッサ１３１４の例は、シリアル入力／出力（Ｉ／Ｏ）チップ内に一体化されたプロセッサである。

【０１８１】

入力プロセッサシステム４０７は、入力エミュレータ１３１８、メモリデバイス１３１６、及び通信デバイス１３２０をさらに含む。メモリデバイス１３１６内には、デフォルトのマッピング１３０８が格納されている。デフォルトのマッピング１３０８は、ユーザアカウント１及び２に関連付けられていることに留意されたい。例えば、レガシーゲームＮがユーザアカウント１を介してアクセスされるとき、メモリデバイス１３１６に格納され、ユーザアカウント１で使用するために選択された現在のマッピングがない限り、デフォルトのマッピング１３０８が入力エミュレータ１３１８によって適用される。

【０１８２】

一例として、本明細書で説明する入力エミュレータは、ハードウェアもしくはソフトウェア、またはそれらの組み合わせである。説明すると、入力エミュレータは、プロセッサ、またはＰＬＤ、またはＡＳＩＣ、またはコントローラである。さらに説明するために、入力エミュレータ１３１８は、更新されたマシン９２０内に実装されると、更新されたマシン９２０が、ハンドヘルドコントローラ４１４から受信したユーザ入力を受信して処理し、レガシーゲームＮのプレイを容易にすることを可能にする。入力エミュレータ１３１８は、レガシーゲームＮをプレイするためのハンドヘルドコントローラ４１４の使用を容易にする。通信デバイス１３２０の一例は、無線転送プロトコルまたは有線転送プロトコ

ルなどの近距離通信プロトコルを適用する通信回路である。有線転送プロトコルの一例は、ＵＳＢプロトコルであり、無線転送プロトコルの一例はＢｌｕｅｔｏｏｔｈ（登録商標）プロトコルである。通信デバイス１３２０は、入力エミュレータ１３１８に結合されている入力プロセッサ１３１４に結合されている。入力エミュレータ１３１８は、基本ブロックコンパイラ１０４に結合されている。入力プロセッサ１３１４と入力エミュレータ１３１８の両方は、メモリデバイス１３１６に結合されている。

【０１８３】

また、ハンドヘルドコントローラ４１４は、有線接続または無線接続などの接続を介して入力プロセッサ１３１４に結合されている。有線接続の一例には、並列転送接続が含まれ、そこでは、情報は、ハンドヘルドコントローラ４１４から入力プロセッサ１３１４への同時並列方式での通信である。

【０１８４】

ユーザ１は、ハンドヘルドコントローラ４１４上で１つ以上の入力デバイスを操作して、ゲームタイトルＧＮを選択し、ユーザ入力４１８を生成する（図４Ａ）。ハンドヘルドコントローラ４１４の入力デバイスの例には、左サムスティック１３０２Ａ、右サムスティック１３０２Ｂ、及び１つ以上のボタンが含まれる。説明すると、ゲームタイトルＧＮを選択するために、ハンドヘルドコントローラ４１４上のボタンが、押される、または押し下げられるなど、ユーザ１によって選択される。ユーザ入力４１８に基づいて、サーバシステム４０４は、ゲームコンソール４０２のメモリデバイス４０６（図４Ａ）での格納のために、コンピュータネットワーク４０８を介してゲームコンソール４０２にゲームコードｇｃＮを送信する。

【０１８５】

レガシーゲームＮのプレイ中、ハンドヘルドコントローラ４１４の１つ以上の入力デバイスは、ユーザ１によって操作され、対応する１つ以上の電気信号を生成し、これが最終的にユーザ入力４１９を形成する。例えば、ユーザ１が、ハンドヘルドコントローラ４１４のサムスティックを前方向または後方向または右方向または左方向に動かすと、電気信号が生成される。説明すると、順方向及び逆方向は、ハンドヘルドコントローラ４１４のｘｙｚ座標系のｙ軸に沿っており、右方向及び左方向は、ｘｙｚ座標系のｘ軸に沿っている。この説明では、順方向は逆方向に反対であり、右方向は左方向に反対である。

【０１８６】

ハンドヘルドコントローラ４１４は、ハンドヘルドコントローラ４１４の１つ以上の入力デバイスに結合された、１つ以上の磁力計、１つ以上の加速度計、１つ以上のポテンショメータ、及び１つ以上のジャイロスコープなどのモーションセンシティブハードウェアを含む。ハンドヘルドコントローラ４１４は、モーションセンシティブハードウェアに結合されたアナログ－デジタル変換器をさらに含む。ハンドヘルドコントローラ４１４は、アナログ－デジタル変換器に結合されたエンコーダなどのプロセッサと、プロセッサに結合された通信デバイスと、を含む。ハンドヘルドコントローラ４１４の１つ以上の入力デバイスがユーザ１によって操作されるとき、１つ以上のアナログモーションセンシティブ信号などの対応する電気信号は、モーションセンシティブハードウェアによって生成される。１つ以上のアナログモーションセンシティブ信号は、モーションセンシティブハードウェアからハンドヘルドコントローラ４１４のアナログ－デジタル変換器に送信される。ハンドヘルドコントローラ４１４のアナログ－デジタル変換器は、アナログモーションセンシティブ信号をアナログ形式からデジタル形式に変換して、デジタルモーションセンシティブ信号を出力する。デジタルモーションセンシティブ信号は、アナログ－デジタル変換器からハンドヘルドコントローラ４１４のプロセッサに送信される。

【０１８７】

ハンドヘルドコントローラ４１４のプロセッサは、符号化プロトコルを適用して、デジタルモーションセンシティブ信号を符号化して、符号化された信号を出力する。例えば、第１のデジタルモーションセンシティブ信号が、ハンドヘルドコントローラ４１４上のオプションボタンの選択を表すと判定すると、ハンドヘルドコントローラ４１４のプロセッ

サは、第１の周波数を使用して第１のデジタルモーションセンシティブ信号を符号化する。この例では、第２のデジタルモーションセンシティブ信号が、ハンドヘルドコントローラ４１４の左サムスティック１３０２Ａの動きを表すと判定することに応答して、ハンドヘルドコントローラ４１４のプロセッサは、第２のデジタルモーションセンシティブ信号を第１の周波数より大きい、またはより小さいなど、第１の周波数とは異なる第２の周波数を使用して第２のデジタルモーションセンシティブ信号を符号化する。また、この例では、第３のデジタルモーションセンシティブ信号が、ハンドヘルドコントローラ４１４の右サムスティック１３０２Ｂの動きを表すと判定したことに応答して、ハンドヘルドコントローラ４１４のプロセッサは、第１及び第２の周波数とは異なる第３の周波数を使用して第３のデジタルモーションセンシティブ信号を符号化する。

【０１８８】

ハンドヘルドコントローラ４１４のプロセッサは、符号化された信号をハンドヘルドコントローラ４１４の通信デバイスに送信する。ハンドヘルドコントローラ４１４の通信デバイスは、Ｂｌｕｅｔｏｏｔｈ（登録商標）プロトコルなどの無線転送プロトコルを符号化された信号に適用して、ユーザ入力４１９の例である１つ以上のデータパケットを生成する。ハンドヘルドコントローラ４１４の通信デバイスは、無線接続を介して入力プロセッサシステム４０７の通信デバイス１３２０にユーザ入力４１９を送信する。

【０１８９】

ユーザ入力４１９を受信すると、通信デバイス１３２０は、無線転送プロトコルを適用して、１つ以上のデータパケットから符号化された信号を抽出し、符号化された信号を入力プロセッサ１３１４に送信する。入力プロセッサ１３１４は、符号化された信号を、復号化及び分析するなどして解析し、ハンドヘルドコントローラ４１４の操作に関連付けられた初期入力デバイス情報を判定する。例えば、入力プロセッサ１３１４は、１つ以上のデジタルモーションセンシティブ信号に基づいて、ハンドヘルドコントローラ４１４の入力デバイスのどれが、ユーザ１によって操作されているか、入力デバイスの動きの量、及び動きの量の方向を識別する。説明すると、入力プロセッサ１３１４は、ハンドヘルドコントローラ４１４の加速度計から受信した１つ以上のアナログモーションセンシティブ信号に基づいて生成された１つ以上のデジタルモーションセンシティブ信号から、ハンドヘルドコントローラ４１４上の特定のボタン（オプションボタンなど）がユーザ１によって選択されているかを判定する。さらに説明するために、第１のデジタルモーションセンシティブ信号が、第１の周波数を使用して符号化されていると判定すると、入力プロセッサ１３１４は、オプションボタンが選択されていると判定し、第２のデジタルモーションセンシティブ信号が、第２の周波数を使用して符号化されていると判定することに応答して、入力プロセッサ１３１４は、左サムスティック１３０２Ａがユーザ１によって操作されていると判定する。

【０１９０】

別の説明として、入力プロセッサ１３１４は、ハンドヘルドコントローラ４１４の加速度計、ジャイロスコープ、及び磁力計から受信した１つ以上のアナログモーションセンシティブ信号に基づいて生成された１つ以上のデジタルモーションセンシティブ信号から、サムスティックが、前方、後方、右方、左方などの特定の方向に動かされたか、そして移動量が特定の方向で事前判定された閾値を超えているかを判定する。この説明では、特定の方向での動き、及び動きの量が、サムスティックの機能の例である。さらに説明するために、入力プロセッサ１３１４は、左サムスティック１３０２Ａの移動量が所定の移動量を超えると判定して、移動量が所定の閾値を超えると判定する。所定の移動量は、メモリデバイス１３１６に記憶される。ハンドヘルドコントローラ４１４のボタンの機能の別の例は、ユーザ１によるボタンの選択（押す、または押し込むなど）である。この例では、ボタンの機能は、ボタンが、入力プロセッサ１３１４によってユーザ１によって操作されていると識別されたときに判定される。ハンドヘルドコントローラ４１４のボタンの機能の別の例として、入力プロセッサ１３１４は、ユーザ１がボタンを操作したときにデジタルモーションセンシティブ信号が生成される回数を判定する。この例では、回数は、デジ

タルモーションセンシティブ信号が、ハンドヘルドコントローラ４１４から入力プロセッサ１３１４によって受信されると判定された回数に等しい。また、この例では、回数は、ボタンがユーザ１によって選択された回数と同じである。

【０１９１】

入力プロセッサ１３１４は、初期入力デバイス情報を入力エミュレータ１３１８に送信する。レガシーゲームＮのプレイ中に初期入力デバイス情報を受信すると、入力エミュレータ１３１８は、メモリデバイス１３１６内に記憶されたデフォルトマッピング１３０８にアクセスして、ハンドヘルドコントローラ４１４の更新された入力デバイスのうちの１つ以上の識別子１３１２に基づいてハンドヘルドコントローラ１３０６の１つ以上のレガシー入力デバイスを識別する。例えば、初期入力デバイス情報を受信すると、入力エミュレータ１３１８は、デフォルトマッピング１３０８から、ユーザ１によって操作されるハンドヘルドコントローラ４１４の更新された入力デバイスのうちの１つ以上の更新された入力デバイス識別子１３１２のうちの１つ以上に対応する１つ以上のレガシー入力デバイス識別子１３１０を判定する。デフォルトのマッピング１４０８は、ハンドヘルドコントローラ１３０６のレガシー入力デバイスのそれぞれと、ハンドヘルドコントローラ４１４の更新された入力デバイスの対応する１つとの間のリンクなどの１対１の対応を含む。

【０１９２】

更新された入力デバイス識別子１３１２の例は、ハンドヘルドコントローラ４１４の更新された入力デバイスのうちの１つ以上を識別する英数字及び記号またはビットなどの識別文字を含む。説明すると、更新された入力デバイス識別子１３１２は、左サムスティック１３０２Ａの記号及び右サムスティック１３０２Ｂの記号を含む。別の例として、更新された入力デバイス識別子１３１２は、ハンドヘルドコントローラ４１４上のタッチパッド用の記号を含む。

【０１９３】

レガシー入力デバイス識別子１３１０の例は、ハンドヘルドコントローラ１３０６のレガシー入力デバイスの１つ以上を識別する英数字及び記号またはビットなどの識別文字を含む。説明すると、更新された入力デバイス識別子１３１０には、ハンドヘルドコントローラ１３０６の左移動ボタン１３０４Ａの記号、ハンドヘルドコントローラ１３０６の前方移動ボタン１３０４Ｂの記号、ハンドヘルドコントローラ１３０６の右移動ボタン１３０４Ｃの記号、及びハンドヘルドコントローラ１３０６の後方移動ボタン１３０４Ｄの記号が含まれる。別の説明として、レガシー入力デバイス識別子１３１０は、ハンドヘルドコントローラ１３０６の選択ボタンの記号を含む。

【０１９４】

さらに、入力エミュレータ１３１８は、ユーザ１によって操作されるハンドヘルドコントローラ４１４の１つ以上の機能から、ハンドヘルドコントローラ１３０６の１つ以上のレガシー入力デバイスの１つ以上の機能を判定する。説明すると、入力エミュレータ１３１８は、所定の閾値を超える左方向への左サムスティック１３０２Ａの移動量が、ユーザ１による左移動ボタン１３０４Ａの３回の選択に対応すると判定する。別の説明として、入力エミュレータ１３１８は、第１の所定の閾値を超える左方向への左サムスティック１３０２Ａの移動量が、ユーザ１による左移動ボタン１３０４Ａの２回の選択に対応すると判定する。この説明では、入力エミュレータ１３１８は、第２の所定の閾値を超える左方向への左サムスティック１３０２Ａの移動量が、ユーザ１による左移動ボタン１３０４Ａの３回の選択に対応すると判定する。第２の事前に判定された閾値は、第１の事前に判定された閾値よりも大きいか、または高くなっている。さらに別の説明として、ハンドヘルドコントローラ４１４の１つ以上の機能から判定される、ハンドヘルドコントローラ１３０６の１つ以上のレガシー入力デバイスの１つ以上の機能は、ハンドヘルドコントローラ４１４の１つ以上の機能によって実現されるような同じ結果を実現する。この説明では、ハンドヘルドコントローラ１３０６の１つ以上のレガシー入力デバイスの１つ以上の機能と、ハンドヘルドコントローラ４１４の１つ以上の機能は、仮想環境４５２内の仮想オブジェクトの同じ移動量になる。同じ移動量の例は、１つの位置から別の位置への動き、ま

たは１つの向きから別の向きへの動き、またはそれらの組み合わせである。
【０１９５】
入力エミュレータ１３１８は、ユーザ１によって操作される１つ以上の更新された入力デバイス識別子１３１２に対応する１つ以上のレガシー入力デバイス識別子１３１０を有するユーザ入力４２０を生成する。また、ユーザ入力４２０は、１つ以上のレガシー入力デバイス識別子１３１０によって識別される１つ以上のレガシー入力デバイスの１つ以上の機能を含む。入力エミュレータ１３１８は、ユーザ入力４２０を基本ブロックコンパイラ１０４に送信する。
【０１９６】
基本ブロックコンパイラ１０４は、ユーザ入力４２０に基づいて、ユーザ入力４２０を処理するための基本ブロック１～ｎのうちの１つ以上がキャッシュ１０２に格納されているかどうかを判定する（図１）。例えば、基本ブロックコンパイラ１０４は、ユーザ入力４２０に基づいて、キャッシュ１０２に格納された基本ブロック１～ｎのうちの１つ以上が、レガシーゲームＮのプレイ中に戦闘機を位置Ｐ１から位置Ｐ２に移動させるために実行され得るかを判定する。説明すると、基本ブロックコンパイラ１０４は、左移動ボタン１３０４Ａの選択に基づいて、戦闘機が位置Ｐ１から位置Ｐ２に移動されることを判定する。基本ブロックコンパイラ１０４は、左移動ボタン１３０４Ａの選択に従って、キャッシュ１０２に格納された基本ブロック１～ｎのうちの１つ以上の実行が、戦闘機の位置Ｐ１から位置Ｐ２への移動を可能にするかどうかをさらに判定する。
【０１９７】
ユーザ入力４２０を処理するための基本ブロック１～ｎのうち１つ以上がキャッシュ１０２に格納されていると判定すると、基本ブロックコンパイラ１０４は、ブロックディスパッチャ３０２に命令を送信する（図３）。ブロックディスパッチャ３０２は、命令に応答して、ユーザ入力４２０を処理するために、キャッシュ１０２から基本ブロック１～ｎのうちの１つ以上を実行する。例えば、基本ブロック１～ｎのうちの１つ以上は、戦闘機を位置Ｐ１から位置Ｐ２に移動させるために実行される。さらに、戦闘機が位置Ｐ１から位置Ｐ２に移動する１つ以上の画像フレームを生成するために、基本ブロック１～ｎのうちの１つ以上が実行される。
【０１９８】
一方で、ユーザ入力４２０を処理するための基本ブロック１～ｎのうちの１つ以上がキャッシュ１０２に格納されていないとの判定に応答して、基本ブロックコンパイラ１０２は、ユーザ入力４２０を処理するためのエミュレートされたＰＵコード命令１～Ｍのうちの１つ以上を識別する。基本ブロックコンパイラ１０２は、エミュレートされたＰＵコード命令１～Ｍのうちの１つ以上をコンパイルして、基本ブロック１～ｎのうちの１つ以上を生成し、ユーザ入力４２０を処理し、基本ブロック１～ｎのうちの１つ以上を格納のためにキャッシュ１０２に提供する。また、基本ブロックコンパイラ１０２は、ブロックディスパッチャ３０２に命令を送信して、キャッシュ１０２内に最近格納された基本ブロック１～ｎのうちの１つ以上を実行する。基本ブロック１～ｎのうちの１つ以上は、次いで、ユーザ入力４２０を処理するために、ブロックディスパッチャ３０２によって実行される。
【０１９９】
一実施形態では、ハンドヘルドコントローラ４１４は、レガシーマシン９００（図９Ａ）と互換性がない。例えば、レガシーマシン９００の入力プロセッサシステムは、ハンドヘルドコントローラ１３０６によって生成され、そこから受信したユーザ入力を解釈することができる入力プロセッサを含む。レガシーシステム９００の入力プロセッサシステムは、ハンドヘルドコントローラ４１４によって生成され、そこから受信したユーザ入力４１９を解釈することができない。さらに説明すると、レガシーシステム９００の入力プロセッサは、ハンドヘルドコントローラ１３０６から受信され、３２ビットで表されるユーザ入力を解釈することができるが、６４ビットまたは１２８ビットで表されるユーザ入力４１９は、解釈することはできない。

【０２００】
一実施形態では、入力エミュレータ１３１８は、入力プロセッサ１３１４の一部分である。例えば、入力エミュレータ１３１８は、入力プロセッサ１３１４の制御ユニット及び算術論理ユニット（ＡＬＵ）による実行のために、入力プロセッサ１３１４のキャッシュ内に格納されたコンピュータプログラムである。
【０２０１】
一実施形態では、ハンドヘルドコントローラ４１４の通信デバイスは、ハンドヘルドコントローラ４１４のモーションセンシティブハードウェアに結合されている。通信デバイスは、有線転送プロトコルを適用して、有線接続の異なるチャネルを介して入力プロセッサシステム４０７の通信デバイスにデジタルモーションセンシティブ信号を送信する。例えば、ハンドヘルドコントローラ４１４上のボタンの動きを表す第１のデジタルモーションセンシティブ信号を受信すると、ハンドヘルドコントローラ４１４の通信デバイスは、有線接続の第１の接続リンクまたは第１の導体などの第１のチャネルを介して、第１のデジタルモーションセンシティブ信号を転送する。この例では、ハンドヘルドコントローラ４１４の左サムスティック１３０２Ａの動きを表す第２のデジタルモーションセンシティブ信号の受信に応答して、ハンドヘルドコントローラ４１４の通信デバイスは、第２の接続リンクまたは第２の導体などの有線接続の第２のチャネルを介して第２のデジタルモーションセンシティブ信号を転送する。また、この例では、ハンドヘルドコントローラ４１４の右サムスティック１３０２Ｂの動きを表す第３のデジタルモーションセンシティブ信号を受信することに応答して、ハンドヘルドコントローラ４１４の通信デバイスは、有線接続の第３の接続リンクや第３のワイヤなどの第３のチャネルを介して第３のデジタルモーションセンシティブ信号を転送する。
【０２０２】
実施形態を続けると、入力プロセッサシステム４０７の通信デバイスは、デジタルモーションセンシティブ信号を受信し、チャネル情報と共にデジタルモーションセンシティブ信号を入力プロセッサ１３１４に提供する。例えば、入力プロセッサ４０７の通信デバイスは、入力プロセッサ１３１４に、第１のデジタルモーションセンシティブ信号が第１のチャネルを介して受信され、第２のデジタルモーションセンシティブ信号が第２のチャネルを介して受信され、第３のデジタルモーションセンシティブ信号が第３のチャネルを介して受信されることを示す。入力プロセッサ１３１４は、チャネルに基づいて、ハンドヘルドコントローラ４１４の入力デバイスのどれがユーザ１によって操作されるかを識別する。
【０２０３】
一実施形態では、ハンドヘルドコントローラ４１４は、それ自体が入力デバイスである。例えば、ハンドヘルドコントローラ４１４は、ユーザ１によって、順方向、逆方向、右方向、または左方向に動かされて、その方向に動く仮想オブジェクトを制御する。
【０２０４】
図１４Ａ－１は、ユーザ１によるデフォルトマッピング１３０８から現在のマッピング１４０４への変更を説明するためのシステム１４００の実施形態の図である。システム１４００は、ハンドヘルドコントローラ４１４、入力プロセッサシステム４０７、ＮＩＣ１２１２、コンピュータネットワーク４０８、及びサーバシステム４０４を含む。
【０２０５】
サーバシステム４０４のメモリデバイス４１２は、ユーザアカウント１及び２に関する情報を格納する。例えば、メモリデバイス４１２は、ユーザアカウント１及び２、ならびにユーザアカウント１及び２へのログインを可能にするための検証情報を格納する。別の例として、メモリデバイス４１２は、レガシーゲームＮ及びゲームタイトルＧａを有する現在のゲームのプレイ中に、ユーザ１によってユーザアカウント１内に蓄積されたポイント数を格納する。この例では、メモリデバイス４１２は、レガシーゲームＮ及びゲームタイトルＧａを有する現在のゲームのプレイ中に、ユーザ２によってユーザアカウント２内に蓄積されたポイントの数を格納する。

【０２０６】
ユーザ１は、ハンドヘルドコントローラ４１４の１つ以上の更新された入力デバイスを操作し、１つ以上の更新された入力デバイスが操作されると、１つ以上のデジタルモーションセンシティブ信号が、上に示したのと同様の方法で生成される。例えば、ユーザ１は、ハンドヘルドコントローラ４１４の更新された入力デバイスのうちの１つ以上を操作して、レガシーゲームＮを一時停止する。この例では、レガシーゲームＮが一時停止されると、ユーザ１は、ハンドヘルドコントローラ４１４の更新された入力デバイスのうちの１つ以上を操作して、デフォルトマッピング１３０８を現在のマッピング１４１４に変更する。別の例として、レガシーゲームＮをプレイする前に、ユーザ１は、ハンドヘルドコントローラ４１４の更新された入力デバイスのうちの１つ以上を操作して、デフォルトのマッピング１３０８を現在のマッピング１４１４に変更する。
【０２０７】
一例として、デフォルトマッピング１３０８は、コントローラ４１４の更新された入力デバイスの第１の第１の識別子を、コントローラ１３０６のレガシー入力デバイスの第１の第１の第１の識別子にマッピングする（図１３）。また、一例として、デフォルトマッピング１３０８は、コントローラ４１４の更新された２つ目の入力デバイスの第２の識別子を、コントローラ１３０６の２つ目のレガシー入力デバイスの第２の識別子にマッピングする。
【０２０８】
ハンドヘルドコントローラ４１４の通信デバイスは、１つ以上のデジタルモーションセンシティブ信号を埋め込む１つ以上のデータパケットを生成し、これは、デフォルトのマッピング１３０８が現在のマッピング１４１４に変更されることを示す。ハンドヘルドコントローラ４１４の通信デバイスは、無線転送プロトコルを適用して、１つ以上のデータパケットを生成する。デフォルトマッピング１３０８を現在のマッピング１４１４に変更するための１つ以上のデータパケットは、ユーザ入力１４０２としてハンドヘルドコントローラ４１４の通信デバイスから無線接続を介して通信デバイス１３２０へ送信される。
【０２０９】
通信デバイス１３２０は、無線転送プロトコルを適用して、デフォルトのマッピング１３０８を１つ以上のパケットから現在のマッピング１４１４に変化させるために１つ以上のデジタルモーションセンシティブ信号を抽出する。通信デバイス１３２０は、デフォルトマッピング１３０８を変化させるための１つ以上のデジタルモーションセンシティブ信号を入力プロセッサ１３１４に送信する。１つ以上のデジタルモーションセンシティブ信号を受信すると、入力プロセッサ１３１４は、デフォルトのマッピング１３０８から現在のマッピング１４０４を生成するために判定する。例えば、コントローラ４１４の更新された入力デバイスのうちの１つ目の第１の識別子が、コントローラ１３０６のレガシー入力デバイスのうちの２つ目の第２の識別子にマッピングされたという指示を受信したことに応答して、入力プロセッサ１３１４は、第１の識別子を第２の識別子にリンクして、現在のマッピング１４０４を生成し、現在のマッピング１４０４を入力エミュレータ１３１８及びユーザアカウント１に関連付ける。別の例として、コントローラ４１４の更新された入力デバイスのうちの２つ目の第２の識別子が、コントローラ１３０６のレガシー入力デバイスのうちの１つ目の第１の識別子にマッピングされたという指示を受信したことに応答して、入力プロセッサ１３１４は、第２の識別子を第１の識別子にリンクして、現在のマッピング１４０４を生成し、現在のマッピング１４０４を入力エミュレータ１３１８に関連付ける。
【０２１０】
さらに、現在のマッピング１４０４がユーザアカウント１を介して生成された後、入力プロセッサ１３１４は、現在のマッピング１４０４をメモリデバイス１３１６に格納し、現在のマッピング１４０４をユーザアカウント１に関連付ける。例えば、ユーザアカウント１にアクセスすると、ユーザ１は、デフォルトのマッピング１３０８に加えて、現在のマッピング１４０４にアクセスすることができる。説明すると、ユーザ１がユーザアカウ

ント１にログインするとき、ゲームタイトルＧＮの選択を示すデジタルモーションセンシティブ信号を受信した後、入力プロセッサ１３１４は、ユーザ１が、デフォルトマッピング１３０８とレガシーゲームＮをプレイするための現在のマッピング１４０４との間で選択する選択肢を提供する。別の例として、現在のマッピング１４０４の生成に応答して、入力プロセッサ１３１４は、入力エミュレータ１３１８に指示して、レガシーゲームＮのプレイ中にハンドヘルドコントローラ４１４から受信した将来のユーザ入力をエミュレートする。この説明では、ユーザ１がユーザアカウント１にログインした後、将来のユーザ入力が入力プロセッサシステム４０７によって受信される。また、この説明では、将来のユーザ入力は、デフォルトのマッピング１３０８の代わりに、現在のマッピング１４０４に従ってエミュレートされている。さらに説明するために、ユーザ１は、ハンドヘルドコントローラ４１４の更新された入力デバイスの１つ以上を操作した後、ユーザアカウント１からログアウトして、デフォルトマッピング１３０８が現在のマッピング１４０４に変更されることを示す。さらなる説明では、ユーザ１は、ユーザアカウント１からログアウトした後、ユーザアカウント１にログインする。さらなる説明を続けると、ユーザｓ１がユーザアカウント１にログインした後、入力エミュレータ１３１８は、デフォルトマッピング１３０８の選択がハンドヘルドコントローラ４１４から受信されるまで、デフォルトマッピング１３０８の代わりに現在のマッピング１４０４を将来のユーザ入力に適用する。

【０２１１】

図１４Ａ－２は、デフォルトマッピング１３０８及び現在のマッピング１４０４を説明するためのメモリデバイス１３１６の実施形態の図である。デフォルトマッピング１３０８は、ハンドヘルドコントローラ１３０６（図１３）のレガシー入力デバイスのレガシー入力デバイス識別子１３１０を、レガシー入力デバイスの機能識別子、ハンドヘルドコントローラ４１４（図１３）の更新された入力デバイスの更新された入力デバイス識別子１３１２、及び更新された入力デバイスの機能の識別子と共にマッピングする。例えば、デフォルトマッピング１３０８は、ハンドヘルドコントローラ１３０６のレガシー入力デバイスのレガシー入力デバイス識別子ＬＩＤ１、レガシー入力デバイスの機能ＬＩＦ１、ハンドヘルドコントローラ１３０６の更新された入力デバイスの更新された入力デバイス識別子ＵＩＤ１、及び更新された入力デバイスの機能ＵＩＦ１の間の１対１の対応を含む。別の例として、デフォルトマッピング１３０８は、ハンドヘルドコントローラ１３０６のレガシー入力デバイスのレガシー入力デバイス識別子ＬＩＤ２、レガシー入力デバイスの機能ＬＩＦ２、ハンドヘルドコントローラ１３０６の更新された入力デバイスの更新された入力デバイス識別子ＵＩＤ２、及び更新された入力デバイスの機能ＵＩＦ２の間の１対１の対応を含む。別の、さらなる別の例として、デフォルトマッピング１３０８は、ハンドヘルドコントローラ１３０６のレガシー入力デバイスのレガシー入力デバイス識別子ＬＩＤ５、レガシー入力デバイスの機能ＬＩＦ５、ハンドヘルドコントローラ１３０６の更新された入力デバイスの更新された入力デバイス識別子ＵＩＤ５、及び更新された入力デバイスの機能ＵＩＦ５の間の１対１の対応を含む。

【０２１２】

同様に、現在のマッピング１４０４は、ハンドヘルドコントローラ１３０６のレガシー入力デバイスのレガシー入力デバイス識別子１３１０を、レガシー入力デバイスの機能の識別子、ハンドヘルドコントローラ４１４の更新された入力デバイスの更新された入力デバイス識別子、及び更新された入力デバイスの機能の識別子と共にマッピングする。例えば、現在のマッピング１４０４は、レガシー入力デバイス識別子ＬＩＤ１、機能ＬＩＦ１、更新された入力デバイス識別子ＵＩＤ２、及び更新された入力デバイスの機能ＵＩＦ２の間の１対１の対応を含む。別の例として、現在のマッピング１４０４は、レガシー入力デバイス識別子ＬＩＤ２、機能ＬＩＦ２、更新された入力デバイス識別子ＵＩＤ１、及び機能ＵＩＦ１の間の１対１の対応を含む。現在のマッピング１４０４では、デフォルトマッピング１３０８の識別子ＬＩＤ１とＵＩＤ１との間のマッピングが、識別子ＬＩＤ１とＵＩＤ２との間のマッピングを生成するため切り替えられることに留意されたい。また、

現在のマッピング１４０４では、デフォルトマッピング１３０８の識別子ＬＩＤ２とＵＩＤ２との間のマッピングが切り替えられて、識別子ＬＩＤ２とＵＩＤ１との間のマッピングを生成する。

【０２１３】

一実施形態では、複数のレガシー入力デバイス識別子が、１つ以上の更新された入力デバイス識別子にマッピングされる。例えば、左移動ボタン１３０４Ａ及び右移動ボタン１３０４Ｃを識別する複数のレガシー入力デバイス識別子は、ハンドヘルドコントローラ４１４のサムスティック（左サムスティック１３０２Ａまたは右サムスティック１３０２Ｂ（図１３）など）の単一の更新された入力識別子にマッピングされる。別の例として、前方移動ボタン１３０４Ｂ及び後方移動ボタン１３０４Ｄを識別する複数のレガシー入力デバイス識別子が、ハンドヘルドコントローラ４１４のサムスティック（左サムスティック１３０２Ａまたは右サムスティック１３０２Ｂなど）の単一の更新された入力識別子にマッピングされる。

【０２１４】

一実施形態では、デフォルトマッピング１３０８は、ハンドヘルドコントローラ１３０６のレガシー入力デバイスの機能の識別子、及びハンドヘルドコントローラ４１４の更新された入力デバイスの機能の識別子を含まない。

【０２１５】

一実施形態では、マッピングは、本明細書で説明されるように、更新されたハンドヘルドコントローラの同じ数の更新された入力デバイスに対する任意数の識別子を含む。また、マッピングには、レガシーハンドヘルドコントローラの同数のレガシー入力デバイスの任意数の識別子が含まれる。

【０２１６】

図１４Ｂは、レガシーゲームＮのプレイ中にデフォルトマッピング１３０８（図１３）の代わりに現在のマッピング１４０４を使用することを説明するためのシステム１４５０の実施形態の図である。ユーザ１は、ユーザアカウント１にログインし、ハンドヘルドコントローラ４１４の更新された入力デバイスのうちの１つ以上を介して現在のマッピング１４０４を選択する。ハンドヘルドコントローラ４１４から現在のマッピング１４０４の選択を示すユーザ入力を受信すると、入力プロセッサ１３１４は、デフォルトマッピング１３０８の代わりに現在のマッピング１４０４を適用するように入力エミュレータ１３１８に命令する。

【０２１７】

レガシーゲームＮのプレイ中、ハンドヘルドコントローラ４１４の１つ以上の入力デバイスは、ユーザ１によって操作され、対応する１つ以上の電気信号を生成し、これが最終的にユーザ入力１４５２を形成する。例えば、ユーザ１が、ハンドヘルドコントローラ４１４のサムスティックを前方向または後方向または右方向または左方向に動かすと、電気信号が生成される。

【０２１８】

ハンドヘルドコントローラ４１４の１つ以上の入力デバイスがユーザ１によって操作されるとき、１つ以上の追加のアナログモーションセンシティブ信号などの対応する電気信号は、モーションセンシティブハードウェアによって生成される。１つ以上の追加のアナログモーションセンシティブ信号は、モーションセンシティブハードウェアからハンドヘルドコントローラ４１４のアナログ－デジタル変換器に送信される。ハンドヘルドコントローラ４１４の追加のアナログ－デジタル変換器は、アナログモーションセンシティブ信号をアナログ形式からデジタル形式に変換して、追加のデジタルモーションセンシティブ信号を出力する。追加のデジタルモーションセンシティブ信号は、アナログ－デジタル変換器からハンドヘルドコントローラ４１４のプロセッサに送信される。

【０２１９】

ハンドヘルドコントローラ４１４のプロセッサは、符号化プロトコルを適用して、追加のデジタルモーションセンシティブ信号を符号化して、追加の符号化された信号を出力す

る。例えば、第１の追加のデジタルモーションセンシティブ信号が、ハンドヘルドコントローラ４１４上のオプションボタンの動作を表すと判定すると、ハンドヘルドコントローラ４１４のプロセッサは、第１の周波数を使用して第１の追加のデジタルモーションセンシティブ信号を符号化する。この例では、第２の追加のデジタルモーションセンシティブ信号が、ハンドヘルドコントローラ４１４の左サムスティック１３０２Ａの動きを表すと判定することに応答して、ハンドヘルドコントローラ４１４のプロセッサは、第２の周波数を使用して第２の追加のデジタルモーションセンシティブ信号を符号化する。また、この例では、第３の追加のデジタルモーションセンシティブ信号が、ハンドヘルドコントローラ４１４の右サムスティック１３０２Ｂの動きを表すと判定したことに応答して、ハンドヘルドコントローラ４１４のプロセッサは、第３の周波数を使用して第３の追加のデジタルモーションセンシティブ信号を符号化する。

【０２２０】

ハンドヘルドコントローラ４１４のプロセッサは、追加の符号化された信号をハンドヘルドコントローラ４１４の通信デバイスに送信する。ハンドヘルドコントローラ４１４の通信デバイスは、無線転送プロトコルを追加の符号化された信号に適用して、ユーザ入力１４５２の例である１つ以上の追加のデータパケットを生成する。ハンドヘルドコントローラ４１４の通信デバイスは、無線接続を介して入力プロセッサシステム４０７の通信デバイス１３２０にユーザ入力１４５２を送信する。

【０２２１】

ユーザ入力１４５２を受信すると、通信デバイス１３２０は、近距離通信プロトコルを適用して、１つ以上のデータパケットから追加の符号化された信号を抽出し、追加の符号化された信号を入力プロセッサ１３１４に送信する。入力プロセッサ１３１４は、追加の符号化された信号を、復号化及び分析するなど、解析して、ハンドヘルドコントローラ４１４の操作に関連付けられた追加の入力デバイス情報を判定する。例えば、入力プロセッサ１３１４は、１つ以上の追加のデジタルモーションセンシティブ信号に基づいて、ハンドヘルドコントローラ４１４の入力デバイスのどれが、ユーザ１によって操作されているか、入力デバイスの動きの量、及び動きの量の方向を識別する。説明すると、入力プロセッサ１３１４は、ハンドヘルドコントローラ４１４の加速度計から出力された追加のアナログモーションセンシティブ信号から生成された１つ以上の追加のデジタルモーションセンシティブ信号から、ハンドヘルドコントローラ４１４上の特定のボタン（オプションボタンなど）がユーザ１によって選択されているかを判定する。さらに説明するために、第１の追加のデジタルモーションセンシティブ信号が、第１の周波数を使用して符号化されていると判定すると、入力プロセッサ１３１４は、オプションボタンが選択されていると判定し、第２の追加のデジタルモーションセンシティブ信号が、第２の周波数を使用して符号化されていると判定することに応答して、入力プロセッサ１３１４は、左サムスティック１３０２Ａがユーザ１によって操作されていると判定する。

【０２２２】

別の説明として、入力プロセッサ１３１４は、ハンドヘルドコントローラ４１４の加速度計、ジャイロスコープ、及び磁力計から出力された追加のアナログモーションセンシティブ信号から生成された１つ以上の追加のデジタルモーションセンシティブ信号から、サムスティックが、前方、後方、右方、左方などの特定の方向に動かされたか、そして移動量が特定の方向で事前判定された閾値を超えているかを判定する。さらに説明するために、入力プロセッサ１３１４は、左サムスティック１３０２Ａの移動量が所定の移動量を超えると判定して、移動量が所定の閾値を超えると判定する。この説明では、特定の方向での動き、及び動きの量が、サムスティックの機能の例である。ハンドヘルドコントローラ４１４のボタンの機能の別の例は、ユーザ１によるボタンの選択である。この例では、ボタンの機能は、ボタンが、入力プロセッサ１３１４によってユーザ１によって選択されていると識別されたときに判定される。ハンドヘルドコントローラ４１４のボタンの機能の別の例として、入力プロセッサ１３１４は、ユーザ１がボタンを操作したときに追加のデジタルモーションセンシティブ信号が生成される回数を判定する。回数は、追加のデジタ

ルモーションセンシティブ信号が、ハンドヘルドコントローラ４１４から入力プロセッサ１３１４によって受信されると判定された回数に等しい。また、この回数は、ボタンがユーザ１によって選択された回数と同じである。

【０２２３】

レガシーゲームＮのプレイ中に追加の入力デバイス情報を受信すると、入力エミュレータ１３１８は、メモリデバイス１３１６内に格納された現在のマッピング１４０４にアクセスして、ハンドヘルドコントローラ４１４の更新された入力デバイスのうちの１つ以上の識別子１３１２に基づいてハンドヘルドコントローラ１３０６の１つ以上のレガシー入力デバイスを識別する。例えば、追加の入力デバイス情報を受信すると、入力エミュレータ１３１８は、現在のマッピング１４０４から、ユーザ１によって操作されるハンドヘルドコントローラ４１４の更新された入力デバイスのうちの１つ以上の更新された入力デバイス識別子１３１２のうちの１つ以上に対応する１つ以上のレガシー入力デバイス識別子１３１０を判定する。

【０２２４】

さらに、入力エミュレータ１３１８は、追加の入力デバイス情報を生成するためにユーザ１によって操作されるハンドヘルドコントローラ４１４の１つ以上の機能から、ハンドヘルドコントローラ１３０６の１つ以上のレガシー入力デバイスの１つ以上の機能を判定する。説明すると、入力エミュレータ１３１８は、所定の閾値を超える左方向への左サムスティック１３０２Ａの移動量が、ユーザ１による左移動ボタン１３０４Ａの３回の選択に対応すると判定する。別の説明として、入力エミュレータ１３１８は、第１の所定の閾値を超える左方向への左サムスティック１３０２Ａの移動量が、ユーザ１による左移動ボタン１３０４Ａの２回の選択に対応すると判定する。この説明では、入力エミュレータ１３１８は、第２の所定の閾値を超える左方向への左サムスティック１３０２Ａの移動量が、ユーザ１による左移動ボタン１３０４Ａの３回の選択に対応すると判定する。

【０２２５】

入力エミュレータ１３１８は、ユーザ１によって操作されたハンドヘルドコントローラ４１４の１つ以上の更新された入力デバイスの１つ以上の更新された入力デバイス識別子１３１２に対応する１つ以上のレガシー入力デバイス識別子１３１０を有するユーザ入力１４５４を生成する。また、ユーザ入力１４５４は、１つ以上のレガシー入力デバイス識別子１３１０によって識別される１つ以上のレガシー入力デバイスの１つ以上の機能を含む。入力エミュレータ１３１８は、ユーザ入力１４５４を基本ブロックコンパイラ１０４に送信する。

【０２２６】

基本ブロックコンパイラ１０４は、ユーザ入力１４５４に基づいて、ユーザ入力１４５４を処理するための基本ブロック１～ｎのうちの１つ以上がキャッシュ１０２に格納されているかどうかを判定する（図１）。例えば、基本ブロックコンパイラ１０４は、ユーザ入力１４５４に基づいて、レガシーゲームＮのプレイ中に戦闘機を位置Ｐ１から位置Ｐ２に移動させるための基本ブロック１～ｎのうちの１つ以上が、キャッシュ１０２に格納されていることを判定する。説明すると、基本ブロックコンパイラ１０４は、左移動ボタン１３０４Ａの選択に基づいて、戦闘機が第１の位置から第２の位置に、または第１の方向から第２の方向に、またはそれらの組み合わせで動かされることを判定する。基本ブロックコンパイラ１０４は、左移動ボタン１３０４Ａの選択に従って、キャッシュ１０２に格納された基本ブロック１～ｎのうちの１つ以上の実行が、戦闘機の第１の位置から第２の位置への移動、または第１の方向から第２の方向への移動、またはこれらの組み合わせを可能にするかどうかをさらに判定する。

【０２２７】

ユーザ入力１４５４を処理するための基本ブロック１～ｎのうち１つ以上がキャッシュ１０２に格納されていると判定すると、基本ブロックコンパイラ１０４は、ブロックディスパッチャ３０２に命令を送信する（図３）。ブロックディスパッチャ３０２は、命令に応答して、ユーザ入力１４５４を処理するために、キャッシュ１０２から基本ブロック１

～ｎのうちの１つ以上を実行する。例えば、基本ブロック１～ｎのうちの１つ以上は、戦闘機を第１の位置から第２の位置に移動するため、または第１の方向から第２の方向に移動するため、またはそれらの組み合わせのために実行される。さらに、基本ブロック１～ｎのうちの１つ以上は、戦闘機が第１の位置から第２の位置に移動する、または第１の方向から第２の方向に移動する、またはその組み合わせである、１つ以上の画像フレームを生成するために実行される。

【０２２８】

一方で、ユーザ入力１４５４を処理するための基本ブロック１～ｎのうちの１つ以上がキャッシュ１０２に格納されていないとの判定に応答して、基本ブロックコンパイラ１０２は、ユーザ入力１４５４を処理するためのエミュレートされたＰＵコード命令１～Ｍのうちの１つ以上を識別する。基本ブロックコンパイラ１０２は、エミュレートされたＰＵコード命令１～Ｍのうちの１つ以上をコンパイルして、ユーザ入力１４５４を処理するための基本ブロック１～ｎのうちの１つ以上を生成し、基本ブロックの１つ以上を格納のためにキャッシュ１０２に提供する。また、基本ブロックコンパイラ１０２は、ブロックディスパッチャ３０２に命令を送信して、キャッシュ１０２内に最近格納された基本ブロック１～ｎのうちの１つ以上を実行する。

【０２２９】

一実施形態では、ハンドヘルドコントローラ４１４の通信デバイスは、ハンドヘルドコントローラ４１４のモーションセンシティブハードウェアに結合されている。通信デバイスは、有線転送プロトコルを適用して、有線転送媒介の異なるチャネルを介して入力プロセッサシステム４０７の通信デバイスに追加のデジタルモーションセンシティブ信号を送信する。例えば、ハンドヘルドコントローラ４１４上のボタンの動きを表す第１の追加のデジタルモーションセンシティブ信号を受信すると、ハンドヘルドコントローラ４１４の通信デバイスは、有線接続の第１のチャネルを介して、第１の追加のデジタルモーションセンシティブ信号を転送する。この例では、ハンドヘルドコントローラ４１４の左サムスティック１３０２Ａの動きを表す第２の追加のデジタルモーションセンシティブ信号の受信に応答して、ハンドヘルドコントローラ４１４の通信デバイスは、有線接続の第２のチャネルを介して第２の追加のデジタルモーションセンシティブ信号を転送する。また、この例では、ハンドヘルドコントローラ４１４の右サムスティック１３０２Ｂの動きを表す第３の追加のデジタルモーションセンシティブ信号を受信することに応答して、ハンドヘルドコントローラ４１４の通信デバイスは、有線接続の第３のチャネルを介して第３の追加のデジタルモーションセンシティブ信号を転送する。

【０２３０】

実施形態を続けると、入力プロセッサシステム４０７の通信デバイスは、追加のデジタルモーションセンシティブ信号を受信し、チャネル情報と共に追加のデジタルモーションセンシティブ信号を入力プロセッサ１３１４に提供する。例えば、入力プロセッサ４０７の通信デバイスは、入力プロセッサ１３１４に、第１の追加のデジタルモーションセンシティブ信号が第１のチャネルを介して受信され、第２の追加のデジタルモーションセンシティブ信号が第２のチャネルを介して受信され、第３の追加のデジタルモーションセンシティブ信号が第３のチャネルを介して受信されることを示す。入力プロセッサ１３１４は、チャネルに基づいて、ハンドヘルドコントローラ４１４の入力デバイスのどれがユーザ１によって操作されるかを識別する。

【０２３１】

図１５は、右サムスティック１３０２Ｂの動きに基づいた右移動ボタン１３０２Ｃの識別を説明するためのシステム１５００の実施形態の図である。システム１５００は、ハンドヘルドコントローラ４１４、ハンドヘルドコントローラ１３０６、入力プロセッサシステム４０７、及びエミュレーションプロセッサシステム４０９を含む。

【０２３２】

ハンドヘルドコントローラ１３０６は、方向パッド（Ｄｐａｄ）１５１０及びパッド１５１２を含む。方向パッド１５１０上に、ボタン１３０２Ａ～１３０２Ｂが配置されてい

る。また、パッド１５１２上には、四角ボタン、三角ボタン、丸ボタン、及びＸボタンが配置されている。さらに、ハンドヘルドコントローラ１３０６は、左ショルダーボタン１５０６及び右ショルダーボタン１５０８を含む。

【０２３３】

ユーザ１は、右サムスティック１３０２を右方向１５０２に動かす。右サムスティック１３０２Ｂの拡大図に示されるように、ユーザが右サムスティックを右方向１５０２に動かすと、モーメントは所定の閾値１５０４を超える。所定の閾値１５０４は、右サムスティック１３０２Ｂの中心を通過する垂直軸である中心軸１５１６から測定される。例えば、中心軸１５１６は、右サムスティック１３０２Ｂの重心を通過する。説明すると、中心軸１５１６と所定の閾値１５０４との間に所定の水平距離がある。一例として、所定の閾値１５０４は垂直軸である。

【０２３４】

ハンドヘルドコントローラ４１４は、右サムスティック１３０２が右方向１５０２に移動して所定の閾値１５０４を超えると、１つ以上のデジタルモーションセンシティブ信号を含むユーザ入力１５１４を生成する。１つ以上のデジタルモーションセンシティブ信号は、右サムスティック１３０２Ｂが右方向１５０２に移動し、所定の閾値１５０４を超えることを示す。ユーザ入力１５１４は、ユーザ入力４１９の例である（図１３）。ユーザ入力１５１４は、ハンドヘルドコントローラ４１４から入力プロセッサシステム４０７に送られる。

【０２３５】

入力プロセッサ１３１４は、ユーザ入力１５１４内に埋め込まれた１つ以上のデジタルモーションセンシティブ信号を受信し、周波数チャネルに基づいて、１つ以上のデジタルモーションセンシティブ信号が、右サムスティック１３０２Ｂがユーザ１によって動かされたことを識別すると判定する。周波数チャネルは、ハンドヘルドコントローラ４１４から入力プロセッサシステム４０７にユーザ入力１５１４内に埋め込まれた１つ以上のデジタルモーションセンシティブ信号を無線で転送するために使用される。第１の周波数を有するチャネルまたは第２の周波数を有するチャネルなどの周波数チャネルの識別子に基づいて、１つ以上のデジタルモーションセンシティブ信号は、右サムスティック１３０２Ｂの動きがあることを識別する。例えば、１つ以上のデジタルモーションセンシティブ信号が、第１の周波数でチャネルを介して転送されるとき、入力プロセッサ１３１４は、右サムスティック１３０２Ｂが動かされたと判定し、１つ以上のデジタルモーションセンシティブ信号が第２の周波数でチャネルを介して転送されるとき、入力プロセッサ１３１４は、左サムスティック１３０２Ａが動かされたと判定する。この例では、第２の周波数は、第１の周波数よりも大きい、または小さい。

【０２３６】

さらに、入力プロセッサ１３１４は、１つ以上のデジタルモーションセンシティブ信号によって示される方向性に基づいて、右サムスティック１３０２Ｂが右方向１５０２に動かされたことをさらに判定する。例えば、１つ以上のデジタルモーションセンシティブ信号は、ハンドヘルドコントローラ４１４のジャイロスコープによって生成された１つ以上のアナログモーションセンシティブ信号に基づいて出力され、ジャイロスコープは、右サムスティック１３０２Ｂの動きの方向性を測定する。

【０２３７】

また、入力プロセッサ１３１４は、ハンドヘルドコントローラ４１４のポテンショメータによって生成された１つ以上のアナログモーションセンシティブ信号に基づいて出力された１つ以上のデジタルモーションセンシティブ信号に基づいて、右サムスティック１３０２Ｂの右方向１５０２での移動量を判定する。入力プロセッサ１３１４はさらに、右サムスティック１３０２Ｂの移動量が、メモリデバイス１３１６に格納されている所定の閾値１５０４を超えると判定する。

【０２３８】

右サムスティック１３０２Ｂの動き、右方向１５０２の動き、及び右方向１５０２の動

きが、所定の閾値１５０４を超えることは、初期入力デバイス情報または追加の入力デバイス情報などの入力デバイス情報の例である。入力プロセッサ１３１４は、入力デバイス情報を入力エミュレータ１３１８に提供する。入力エミュレータ１３１８は、入力デバイス情報を受信し、入力デバイス情報に基づいて、デフォルトのマッピング１３０８（図１４Ａ－２）または現在のマッピング１４０４（図１４Ａ－２）などのマッピング１５３０にアクセスする。例えば、右サムスティック１３０２Ｂが右方向１５０２に動かされたという指示を受信すると、入力エミュレータ１３１８は、デフォルトマッピング１３０８から、右サムスティック１３０２Ｂの動きが右移動ボタン１３０２Ｃの選択に対応することを識別する。さらに、入力エミュレータ１３１８は、デフォルトのマッピング１３０８から、右サムスティック１３０２Ｂの移動量が、右移動ボタン１３０２Ｃの選択の所定の回数（２回、３回、または４回など）に対応することを判定する。

【０２３９】

入力エミュレータ１３１８は、右移動ボタン１３０２Ｃの操作を示す情報を含むユーザ入力１５２０を生成する。この情報は、右移動ボタン１３０２Ｃの選択と、右移動ボタン１３０２Ｃが選択される所定の回数を示す。右移動ボタン１３０２Ｃは、実際にはユーザ１によって選択されないことに留意されたい。むしろ、右サムスティック１３０２Ｂの動きは、右移動ボタン１３０２Ｃの選択としてエミュレートされる。ユーザ入力１５２０は、ユーザ入力４２０の例である（図１３）。入力エミュレータ１３１８は、ユーザ入力１５２０を基本ブロックコンパイラ１０４に提供し、これは、図１３を参照して上に示した方法で操作する。

【０２４０】

図１５を参照して上で説明された実施形態は、右サムスティック１３０２Ｂを右方向１５０２に動かすためのものであることに留意されたい。一実施形態では、右サムスティック１３０２Ｂの代わりに、左サムスティック１３０２Ａが、右方向、左方向、前方向、または後方向に動かされ、その動きに基づいて、ユーザ入力がハンドヘルドコントローラ４１４によって生成される。一実施形態では、右方向１５０２への移動の代わりに、右、前方、後方、またはそれらの組み合わせなどの他の任意の方向への移動が発生する。

【０２４１】

一実施形態では、ハンドヘルドコントローラ４１４の代わりに、ホイールコントローラが使用される。ホイールコントローラはドライビングホイールの形をしている。一例として、ホイールコントローラは、左サムスティック１３０２Ａ及び右サムスティック１３０２Ｂを含まない。代わりに、ドライビングホイールは、ユーザ１によって操作されて、レガシーゲームＮの仮想オブジェクトを左方向、右方向、前方向、及び後方向に動かす。ドライビングホイールは、時計回りまたは反時計回りに回転されることができる。ドライビングホイールがユーザ１によって時計回りに回転されると、入力プロセッサ１３１４は、１つ以上のデジタルモーションセンシティブ信号に基づいて、ドライビングホイールが回転されたこと、及びドライビングホイールが、反時計回りの方向又は時計回りの方向に回転されたかどうかを判定する。入力プロセッサ１３１４はさらに、回転量が事前設定された閾値を超えるかどうかを判定する。入力プロセッサ１３１４は、ドライビングホイールの識別子、回転方向、及び回転量が事前設定された閾値を超えるかどうかの指示などの情報を入力エミュレータ１３１８に提供する。入力エミュレータ１３１８は、デフォルトマッピングまたは現在のマッピングなどのマッピングにアクセスして、ハンドヘルドコントローラ１３０６の対応するボタン、及び対応するボタンの操作を識別し、ユーザ入力を生成する。例えば、入力エミュレータ１３１８は、情報、及びマッピングに基づいて、ハンドヘルドコントローラ１３０６の対応するボタンが選択されたこと、及び対応するボタンがユーザ入力を生成するために選択された回数を識別する。入力エミュレータ１３１８は、ユーザ入力を図１３または１４Ｂを参照して上に示した方法で操作する基本ブロックコンパイラ１０４に提供する。

【０２４２】

図１６Ａは、タッチパッド１６０２がユーザ１によって選択されたときの十字線の仮想

画像の生成を説明するためのシステム１６００の実施形態の図である。タッチパッド１６０２は、ボタンの一例である。システム１６００は、入力プロセッサシステム４０７、ハンドヘルドコントローラ４１４、及びエミュレーションプロセッサシステム４０９を含む。ハンドヘルドコントローラ４１４は、タッチパッド１６０２、方向パッド１６０４、パッド１６０６、左サムスティック１３０２Ａ、及び右サムスティック１３０２Ｂを含む。

【０２４３】

レガシーゲームＮのプレイ中に、ユーザ１は、自身の指を使用して、タッチパッド１６０２を選択する。タッチパッド１６０２は、ハンドヘルドコントローラ４１４の更新された入力デバイスのうちの１つの例である。ハンドヘルドコントローラ４１４のモーションセンシティブハードウェアは、タッチパッド１６０２の選択に応答して、１つ以上のアナログモーションセンシティブ信号を生成する。例えば、１つ以上のアナログモーションセンシティブ信号は、タッチパッド１６０２の選択、及びタッチパッド１６０２の基準座標系に対するタッチパッド１６０２の選択場所を示す。説明すると、ｘ軸に沿った選択の場所に基づいて、ハンドヘルドコントローラ４１４の第１のポテンショメータは、第１の電気信号を生成し、ｙ軸に沿った選択の場所に基づいて、ハンドヘルドコントローラ４１４の第２のポテンショメータは、第２の電気信号を生成する。タッチパッド１６０２に関連するｘ軸及びｙ軸は、以下でさらに説明されている。１つ以上のアナログモーションセンシティブ信号は、ハンドヘルドコントローラ４１４のアナログ－デジタル変換器によって、１つ以上のデジタルモーションセンシティブ信号に変換される。ハンドヘルドコントローラ４１４のプロセッサは、符号化プロトコルを１つ以上のデジタルモーションセンシティブ信号に適用して、１つ以上の符号化信号を出力する。例えば、１つ以上の符号化信号は、固有の関係を有するような、タッチパッド１６０２に対応する周波数を有する。ハンドヘルドコントローラ４１４の通信デバイスは、無線転送プロトコルを適用して、１つ以上の符号化信号を１つ以上のデータパケットに埋め込み、無線接続を介して通信デバイス１３２０に１つ以上のデータパケットを送信する。１つ以上のデータパケットが、ユーザ入力１６１２として図１６Ａに示されている。通信デバイス１３２０は、無線転送プロトコルを、ハンドヘルドコントローラ４１４から受信した１つ以上のデータパケットに適用して、符号化された信号を取得し、符号化された信号を入力プロセッサ１３１４に送信する。

【０２４４】

入力プロセッサ１３１４は、図１３を参照して説明したのと同様の方法で、タッチパッド１６０２がユーザ１によって選択されたことを判定または識別する。例えば、１つ以上の符号化された信号内に符号化された１つ以上のデジタルモーションセンシティブ信号が、所定の周波数チャネルに一致するハンドヘルドコントローラ４１４からの周波数で受信されると判定することに応答して、入力プロセッサ１３１４は、タッチパッド１６０２が選択されたと判定する。

【０２４５】

この例では、所定の周波数チャネルは、メモリデバイス１３１６内に格納されている。また、この例では、入力プロセッサ１３１４は、符号化された信号にフーリエ変換を適用することによって周波数を判定する。

【０２４６】

入力プロセッサ１３１４はさらに、タッチパッド１６０２の基準座標系に関して、ユーザ１によるタッチパッド１６０２上の選択場所を判定する。例えば、入力プロセッサ１３１４は、ハンドヘルドコントローラ４１４の第１のポテンショメータから受信した電圧量に基づいて、タッチパッド１６０２が、タッチパッド１６０２の基準座標系に関して座標（ｘ１，０）においてユーザ１によって選択されていると判定する。さらに、この例では、入力プロセッサ１３１４は、ハンドヘルドコントローラ４１４の第２のポテンショメータから受信した電圧量に基づいて、タッチパッド１６０２が、タッチパッド１６０２の基準座標系に関して座標（０，ｙ１）において、ユーザ１によって選択されていると判定する。この例では、第１及び第２の電圧量が、１つ以上の符号化された信号内で受信される

。説明すると、ハンドヘルドコントローラ４１４のモーションセンシティブハードウェアから出力される１つ以上のアナログモーションセンシティブ信号は、第１及び第２の電圧量を表す情報を含む。選択されたタッチパッド１６０２の識別子及びユーザ１によるタッチパッド１６０２上の選択場所は、初期入力デバイス情報及び追加の入力デバイス情報などの入力デバイス情報の例である。

【０２４７】

入力プロセッサ１３１８は、選択されたタッチパッド１６０２の識別子及びタッチパッド１６０２上の選択場所を入力エミュレータ１３１８に提供する。入力エミュレータ１３１８は、マッピング１６１０にアクセスして、選択されたタッチパッド１６０２の識別子に基づいて、ハンドヘルドコントローラ１３０６上の対応するボタンを識別する。例えば、入力エミュレータ１３１８は、マッピング１６１０から、ハンドヘルドコントローラ１３０６上の選択ボタンが、タッチパッド１６０２と固有の関係を有することを判定する。マッピング１６１０は、デフォルトのマッピング１３０８（図１３）または現在のマッピング１４０４（図１４Ａ－２）の一例である。

【０２４８】

また、入力エミュレータ１３１８は、ブロックディスパッチャ３０２に要求を送信して、レガシーゲームＮの仮想シーンを有する画像フレームのサイズを取得する。一例として、サイズが要求されるレガシーゲームＮの画像フレームは、ユーザ入力１６１２に基づいてユーザ入力１６１４が生成された後に生成される。説明すると、サイズが要求されるレガシーゲームＮの第１の画像フレームは、ユーザ入力１６１４に基づいて生成され、第１の画像フレームが生成される前に、ユーザ入力１６１４に基づいて他の画像フレームは生成されない。ブロックディスパッチャ３０２は、画像フレームのサイズを入力エミュレータ１３１８に提供する。例えば、ブロックディスパッチャ３０２は、ユーザ入力１６１４に基づいて生成される画像フレームの直前に生成された先行する画像フレームと同じになるように画像フレームのサイズを提供する。説明すると、ユーザ入力１６１４に基づいて生成される画像フレームの生成と前の画像フレームとの間に、ブロックディスパッチャ３０２によって生成される他の画像フレームはない。入力エミュレータ１３１８は、ブロックディスパッチャ３０２に結合されている。

【０２４９】

入力エミュレータ１３１８は、ユーザ入力１６１４に基づいて生成される画像フレーム内の十字線の仮想画像の場所を判定する。画像フレーム内の十字線の仮想画像の場所は、タッチパッド１６０２上の選択場所に対応する。画像フレーム内の十字線の仮想画像の場所が、タッチパッド１６０２上の選択場所と１対１の関係を有する方法の説明が、図１６Ｂを参照して説明されている。

【０２５０】

入力エミュレータ１３１８は、ハンドヘルドコントローラ１３０６上の選択ボタンの指示と、ユーザ入力１６１４に基づいて生成される画像フレーム内の十字線の仮想画像の場所と、を有するユーザ入力１６１４を生成する。ユーザ入力１６１４は、入力エミュレータ１３１８からブロックコンパイラ１０４に提供される。

【０２５１】

ユーザ入力１６１４を受信すると、ブロックコンパイラ１０４は、ユーザ入力１６１４を処理するための基本ブロック１～ｎのうちの１つ以上がキャッシュ１０２に格納されているかどうかを判定する（図１）。基本ブロック１～ｎのうちの１つ以上がキャッシュ１０２に格納されていると判定したことに応答して、ブロックディスパッチャ３０２（図３）は、１つ以上の基本ブロック１～ｎを実行して、ユーザ入力１６１４を処理するための１つ以上の画像フレームを生成する。例えば、ブロックディスパッチャ３０２（図３）は、ブロックコンパイラ１０４から生成される画像フレーム内の十字線の仮想画像の場所を受信し、画像フレーム内の仮想オブジェクトとしての十字線を生成するために、１つ以上の基本ブロック１～ｎを実行する。さらに、十字線は、ユーザ１によるタッチパッド１６０２の選択場所に対応する場所で画像フレーム内に生成される。

【０２５２】
一実施形態では、入力エミュレータ１３１８は、ブロックディスパッチャ３０２が画像フレーム内の十字線の仮想画像の位置を判定するためのコマンドを生成し、コマンドをブロックディスパッチャ３０２に送信する。例えば、コマンドは、ブロックディスパッチャ３０２によって生成される画像フレームが、ユーザ１によるタッチパッド１６０２上の選択場所に比例する場所において、十字線の仮想画像を含むものであることをブロックディスパッチャ３０２に示す。コマンドは、ユーザ１によるタッチパッド１６０２上の選択場所を含む。入力エミュレータ１３１８は、ブロックディスパッチャ３０２に結合されている。この実施形態では、ユーザ入力１６１４は、画像フレーム内の十字線の仮想画像の場所を含まない。ブロックディスパッチャ３０２は、基本ブロック１～ｎのうちの１つ以上を実行して、画像フレームを生成する。さらに、ブロックディスパッチャ３０２は、図１６Ｂを参照してなど、本明細書で説明するのと同じ方法で、画像フレーム内の十字線の仮想画像の場所を判定し、そこでは、入力エミュレータ１３１８が、ユーザ１によるタッチパッド１６０２上の選択場所に基づいて、画像フレーム内の十字線の仮想画像の位置を判定する。
【０２５３】
図１６Ｂは、システム１６５０の実施形態の図であり、ユーザ１によるタッチパッド１０６２上の選択場所と、仮想環境４５２を有する画像フレーム１６５４上の十字線１６５２の場所との間の対応を示す。この対応は、入力エミュレータ１３１８によって判定される（図１６Ａ）。画像フレーム１６５４は、ディスプレイデバイス４１０のディスプレイ画面に表示される（図４Ｂ）。システム１６５０は、タッチパッド１６０２及び画像フレーム１６５４を含む。
【０２５４】
タッチパッド１６０２は、上部エッジ１６６６Ａ、右エッジ１６６６Ｂ、下部エッジ１６６６Ｃ、及び左エッジ１６６６Ｄを有する。上部エッジ１６６６Ａは、右エッジ及び左エッジに隣接し、接続している。また、下部エッジ１６６６Ｃは、右エッジ及び左エッジに隣接し、接続している。下部エッジ１６６６Ｃは、上部エッジ１６６６Ａに面し、右エッジ１６６６Ｂは、左エッジ１６６６Ｄに面している。
【０２５５】
タッチパッド１６０２上の選択場所は、座標（ｘ１，ｙ１）である。タッチパッド１６０２は、タッチパッド１６０２の隅に原点（０，０）を有する基準座標系に関連付けられている。タッチパッド１６０２は、長方形の４つの頂点などの４つの隅を有する。一例として、タッチパッド１６０２は、頂点を有する長方形である。タッチパッド１６０２の基準座標系のｘ軸は、タッチパッド１６０２の下部エッジ１６６６Ｃに沿っており、基準座標系のｙ軸は、タッチパッド１６０２の左エッジ１６６６Ｄに沿っている。座標ｘ１は、原点（０、０）からｘ軸に沿った距離であり、座標ｙ１は、原点（０、０）からｙ軸に沿った距離である。基準座標系のｘ軸は、基準座標系のｙ軸に垂直である。
【０２５６】
同様に、画像フレーム１６５４は、上部エッジ１６６８Ａ、右エッジ１６６８Ｂ、下部エッジ１６６８Ｃ、及び左エッジ１６６８Ｄを有する。上部エッジ１６６８Ａは、右エッジ及び左エッジに隣接し、接続している。また、下部エッジ１６６８Ｃは、画像フレーム１６５４の右エッジ及び左エッジに隣接し、接続している。下部エッジ１６６８Ｃは上部エッジ１６６８Ａに面し、右エッジ１６６８Ｂは左エッジ１６６８Ｄに面している。
【０２５７】
十字線１６５２の場所は、座標（Ｘ１，Ｙ１）にある。画像フレーム１６５４は、画像フレーム１６５４の隅に原点（０，０）を有する基準座標系に関連付けられている。画像フレーム１６５４は、長方形の４つの頂点などの４つの角を有する。一例として、画像フレーム１６５４は、頂点を有する長方形の画像フレームである。画像フレーム１６５４の基準座標系のＸ軸は、下部エッジ１６６８Ｃに沿っており、画像フレーム１６５４の基準座標系のＹ軸は左エッジ１６６８Ｄに沿っている。座標Ｘ１は、画像フレーム１６５４の

隅にある原点（０，０）からのＸ軸に沿った距離であり、座標Ｙ１は、原点（０，０）からのＹ軸に沿った距離である。

【０２５８】

入力エミュレータ１３１８は、画像フレーム１６５４の基準座標系のＸ軸に沿った距離が、タッチパッド１６０２の基準座標系のｘ軸に沿った距離に比例していることを判定する。例えば、入力エミュレータ１３１８は、画像フレーム１６５４の幅がタッチパッド１６０２の幅のＰ倍であると判定し、ここで、Ｐはゼロより大きい整数である。画像フレーム１６５４の幅は、画像フレーム１６５４の基準座標系のＸ軸に沿って測定され、タッチパッド１６０２の幅は、タッチパッド１６０２の基準座標系のｘ軸に沿って測定される。この例では、入力エミュレータ１３１８は、画像フレーム１６５４の高さがタッチパッド１６０２の高さのＱ倍であると判定し、ここで、Ｑはゼロより大きい整数である。画像フレーム１６５４の高さは、画像フレーム１６５４の基準座標系のＹ軸に沿って測定され、タッチパッド１６０２の高さは、タッチパッド１６０２の基準座標系のｙ軸に沿って測定される。この例では、入力エミュレータ１３１８は、Ｘ軸に沿った距離が、タッチパッド１６０２の基準座標系のｘ軸に沿った距離のＰ倍であると判定する。また、この例では、入力エミュレータ１３１８は、Ｙ軸に沿った距離が、タッチパッド１６０２の基準座標系のｘ軸に沿った距離のＱ倍であると判定する。

【０２５９】

図１７は、システム１７００の実施形態の図であり、異なるタイプのハンドヘルドコントローラを用いた異なるマッピングの使用を示している。システム１７００は、ホイールコントローラ１７０２、ハンドヘルドコントローラ４１４、及び入力プロセッサシステム４０７を含む。ホイールコントローラ１７０２は、ドライビングホイールの形状を有し、ステアリングホイール１７０８、方向パッド１７０６、正方形ボタン、三角形ボタン、円形ボタン、及びＸボタンなどの更新された入力デバイスを含む。

【０２６０】

ホイールコントローラ１７０２は、ハンドヘルドコントローラ４１４とは異なるタイプであることに留意されたい。例えば、ハンドヘルドコントローラ４１４は、左ハンドル１７０４Ａ及び右ハンドル１７０４Ｂを有し、ステアリングホイールを含まない。ハンドル１７０４Ａ及び１７０４Ｂは、ハンドヘルドコントローラ４１４の本体から下向きに延在する。比較すると、ホイールコントローラ１７０２は、ステアリングホイール１７０８を有するが、左右のハンドルを含まない。

【０２６１】

さらに、第１のハンドヘルドコントローラと第２のハンドヘルドコントローラの両方が同じタイプのものであるが、異なるエンティティによって製造されているとき、第１のハンドヘルドコントローラのサブタイプは、第２のハンドヘルドコントローラのサブタイプとは異なるサブタイプである。例えば、Ｌｏｇｉｔｅｃｈ（登録商標）ｃｏｒｐｏｒａｔｉｏｎが製造したホイールコントローラは、Ｓｏｎｙ（登録商標）ｃｏｒｐｏｒａｔｉｏｎが製造したホイールコントローラとは異なるホイールコントローラのサブタイプである。別の例として、Ｓｏｎｙ（登録商標）ｃｏｒｐｏｒａｔｉｏｎによって製造されたハンドヘルドコントローラは、Ｍｉｃｒｏｓｏｆｔ（登録商標）ｃｏｒｐｏｒａｔｉｏｎによって製造されたハンドヘルドコントローラとは異なるタイプのハンドヘルドコントローラである。

【０２６２】

ホイールコントローラ１７０２は、ハンドヘルドコントローラ４１４の内部構成要素と同じまたは類似の内部構成要素を有する。例えば、ホイールコントローラ１７０２は、モーションセンシティブハードウェア、アナログ－デジタル変換器、プロセッサ、及び通信デバイスを含む。ホイールコントローラ１７０２の内部構成要素は、ハンドヘルドコントローラ４１４の内部構成要素が互いに結合されるのと同じ方法で互いに結合される。

【０２６３】

ユーザ１が、ハンドヘルドコントローラ４１４またはホイールコントローラ１７０２な

どの更新されたコントローラを入力プロセッサシステム４０７と結合するとき、ユーザ１は、更新されたコントローラの１つ以上の更新された入力デバイスを操作して、更新されたコントローラのタイプ及びサブタイプを提供する。例えば、ユーザ１は、ハンドヘルドコントローラ４１４を入力プロセッサシステム４０７から切り離し、代わりに、ホイールコントローラ１７０２を入力プロセッサシステム４０７に結合する。更新されたコントローラがユーザ１によって操作されると、更新されたコントローラは、更新されたコントローラのタイプを有する識別信号を生成し、識別信号を入力プロセッサシステム４０７に送信する。また、更新されたコントローラは、更新されたコントローラのサブタイプを有する識別信号を生成し、入力プロセッサシステム４０７に識別信号を送信する。入力プロセッサ１３１４は、更新されたコントローラのタイプ及びサブタイプを受信し、タイプ及びサブタイプを入力エミュレータ１３１８に提供する。

【０２６４】

更新されたコントローラのタイプ及びサブタイプを受信すると、入力エミュレータ１３１８は、タイプ及びサブタイプに基づいて、マッピング１７１０またはマッピング１７１２を選択する。例えば、入力エミュレータ１３１８は、更新されたコントローラのタイプがハンドヘルドコントローラ４１４であり、更新されたコントローラのサブタイプがＳｏｎｙ（登録商標）ｃｏｒｐｏｒａｔｉｏｎ製のハンドヘルドコントローラ４１４であるときに、マッピング１７１０を選択する。更新されたコントローラのタイプが、ホイールコントローラ１７０２であり、更新されたコントローラのサブタイプが、Ｓｏｎｙ（登録商標）ｃｏｒｐｏｒａｔｉｏｎ製のホイールコントローラ１７０２であるときに、入力エミュレータはマッピング１７１２を選択する。マッピング１７１０の例は、ユーザアカウント１のデフォルトのマッピング、またはユーザアカウント１の現在のマッピングである。マッピング１７１２の例は、ユーザアカウント１のデフォルトのマッピング、またはユーザアカウント１の現在のマッピングである。

【０２６５】

レガシーゲームＮのプレイ中、入力プロセッサシステム４０７は、ハンドヘルドコントローラ４１４を切り離した後に、入力プロセッサシステム４０７に結合されたホイールコントローラ１７０２からユーザ入力４１９を受信する。ユーザ入力４１９は、ホイールコントローラ１７０２によって同じ方法で生成され、そこでは、ユーザ入力４１９は、サムスティック１３０２Ａ及び１３０２Ｂの代わりにドライビングホイール１７０８が回転されることを除いて、ハンドヘルドコントローラ４１４によって生成される。例えば、ユーザ１は、ホイールコントローラ１７０２の更新された入力デバイスのうちの１つ以上を操作する。ホイールコントローラ１７０２のモーションセンシティブハードウェアは、ホイールコントローラ１７０２の１つ以上の更新された入力デバイスの操作に応答して、１つ以上のアナログモーションセンシティブ信号を生成する。ホイールコントローラ１７０２のアナログ－デジタル変換器は、１つ以上のアナログモーションセンシティブ信号を１つ以上のデジタルモーションセンシティブ信号に変換する。ホイールコントローラ１７０２のプロセッサは、符号化プロトコルを適用して、１つ以上の符号化信号を出力するために１つ以上のデジタルセンシティブ信号を符号化する。ホイールコントローラ１７０２の通信デバイスは、無線転送プロトコルを１つ以上の符号化された信号に適用して、１つ以上のデータパケットを生成し、１つ以上のデータパケットを通信デバイス１３２０に送信する。

【０２６６】

通信デバイス１３２０は、１つ以上の符号化信号を抽出するために無線転送プロトコルを１つ以上のデータパケットに適用し、１つ以上の符号化信号を入力プロセッサ１３１４に送信する。入力プロセッサ１３１４は、１つ以上の符号化された信号から、ホイールコントローラ１７０２の更新された入力デバイスのどれがユーザ１によって操作されているかを識別し、１つ以上の更新された入力デバイスの１つ以上の識別子を入力エミュレータ１３１８に提供する。入力エミュレータ１３１８は、マッピング１７１２にアクセスし、マッピング１７１２を適用して、ハンドヘルドコントローラ１３０６（図１３）の対応す

る１つ以上のレガシー入力デバイスを識別して、ユーザ入力４２０を生成する。

【０２６７】

図１８は、ゲームコンソール４０２からゲームコンソール１２０２への現在のマッピングの通信を説明するためのシステム１８００の実施形態の図である。システム１８００は、ゲームコンソール４０２、コンピュータネットワーク４０８、サーバシステム４０４、及びゲームコンソール１２０２を含む。

【０２６８】

ゲームコンソール１２０２は、入力プロセッサシステム１２０７を含み、これは、入力プロセッサ１８０４、入力エミュレータ１８０６、メモリデバイス１８０８、及び通信デバイス１８１０を含む。説明として、入力エミュレータ１８０６は、ゲームコンソール１２０２内に実装されると、ハンドヘルドコントローラ１２１２を使用することによってレガシーゲームＮのプレイを可能にする。通信デバイス１８１０の例は、近距離通信プロトコルを適用する通信デバイスである。

【０２６９】

通信デバイス１８１０は、入力プロセッサ１８０４に結合されている。入力プロセッサ１８０４及び入力エミュレータ１８０６は、メモリデバイス１８０８に結合されている。入力プロセッサシステム１２０７は、バス１２１８に結合されている。

【０２７０】

ゲームコンソール４０２の入力プロセッサ１３１４は、メモリデバイス１３１６からの現在のマッピング１４０４にアクセス（読み取りなど）し、現在のマッピング１４０４をＮＩＣ１２１２に送信する。例えば、現在のマッピング１４０４が生成された後の所定の期間内に、入力プロセッサ１３１４は、現在のマッピング１４０４をＮＩＣ１２１２に送信する。説明すると、現在のマッピング１４０４が作成された直後に、現在のマッピング１４０４は、メモリデバイス１３１６に格納され、同時にＮＩＣ１２１２に送信される。ＮＩＣ１２１２は、１つ以上のデータパケットを生成するためにネットワーク通信プロトコルを現在のマッピング１４０４に適用し、サーバシステム４０４の１つ以上のメモリデバイスに格納するために、コンピュータネットワーク４０８を介して１つ以上のデータパケットをサーバシステム４０４に送信する。

【０２７１】

ユーザ１は、ハンドヘルドコントローラ１２１２を使用して、ユーザアカウント１にログインし、ゲームコンソール１２０２を介してレガシーゲームＮにアクセスする。さらに、レガシーゲームＮのプレイの前またはプレイ中に、ユーザ１は、ハンドヘルドコントローラ１２１２の１つ以上の更新された入力デバイスを選択して、サーバシステム４０４から現在のマッピング１４０４にアクセスするための表示（要求など）を生成する。ハンドヘルドコントローラ１２１２の１つ以上の更新された入力デバイスの選択を受信すると、ハンドヘルドコントローラ１２１２は、無線転送プロトコルを適用して１つ以上のデータパケットを生成し、１つ以上のデータパケットを通信デバイス１８１０に送信する。通信デバイス１８１０は、無線転送プロトコルを適用して、現在のマッピング１４０４がユーザアカウント１について取得されるべきであるという指示を取得し、その指示をゲームコンソール１２０２のメモリデバイス１８０８に格納する。

【０２７２】

ゲームコンソール１２０２のＣＰＵ９２２（図９Ｂ）などのＣＰＵは、メモリデバイス１８０８から現在のマッピング１４０４を取得するための指示にアクセスし、その指示をＮＩＣ１２１０に送信する。ＮＩＣ１２１０は、現在のマッピング１４０４を取得するための指示にネットワーク通信プロトコルを適用して、１つ以上のデータパケットを生成し、コンピュータネットワーク４０８を介してサーバシステム４０４に１つ以上のデータパケットを送信する。

【０２７３】

ゲームコンソール１２０２から現在のマッピング１４０４の指示を受信することに応答して、サーバシステム４０４は、サーバシステム４０４の１つ以上のメモリデバイスから

現在のマッピング１４０４にアクセスし、コンピュータネットワーク４０８を介して現在のマッピング１４０４をゲームコンソール１２０２に送信する。例えば、サーバシステム４０４のプロセッサは、現在のマッピング１４０４を取得するための指示が、ユーザアカウント１に対応することを判定する。説明すると、サーバシステム４０４のプロセッサは、ユーザ１がユーザアカウント１にログインしている間に指示が受信されたことを判定して、指示がユーザアカウント１に対応することを判定する。この例を続けると、サーバシステム４０４のプロセッサは、サーバシステム４０４の１つ以上のメモリデバイスから現在のマッピング１４０４にアクセスし、サーバシステム４０４のネットワークインターフェースコントローラに現在のマッピング１４０４を送信する。サーバシステム４０４のネットワークインターフェースコントローラは、ネットワーク通信プロトコルを現在のマッピング１４０４に適用して、１つ以上のデータパケットを生成し、コンピュータネットワーク４０８を介してゲームコンソール１２０２に１つ以上のデータパケットを送信する。

【０２７４】

ゲームコンソール１２０２のＮＩＣ１２１０は、サーバシステム４０４から受信した１つ以上のデータパケットから現在のマッピング１４０４を抽出し、現在のマッピング４０４をメモリデバイス１８０８に格納する。ユーザ入力１４５２は、ユーザ入力１４５２が、入力プロセッサシステム４０７によってハンドヘルドコントローラ４１４から受信されるのと同じ方法で、入力プロセッサシステム１２０７によってハンドヘルドコントローラ１２１２から受信される。ユーザ入力１４５２が受信されると、入力エミュレータ１８０６は、メモリデバイス１８０８から現在のマッピング１４０４にアクセスして、ユーザ入力１４５４を判定する。例えば、ユーザ入力１４５４は、ユーザ入力１４５４が、ユーザ入力１４５２からの入力エミュレータ１３１８によって判定されるのと同じ方法で、ユーザ入力１４５２からの入力エミュレータ１８０６によって判定される。入力エミュレータ１８０６は、バス１２１８を介してエミュレーションプロセッサシステム１２０６にユーザ入力１４５４を送信する。

【０２７５】

ユーザ入力１４５４を受信すると、エミュレーションプロセッサシステム１２０６の基本ブロックコンパイラは、エミュレーションプロセッサシステム１２０６のキャッシュ１２０８をチェックして、キャッシュ１２０８が、ユーザ入力１４５４を処理するためのゲームコードＧＣＮの基本ブロック１～ｎのうちの１つ以上を含むかどうかを判定する。キャッシュ１２０８が、ユーザ入力１４５４を処理するための基本ブロック１～ｎのうちの１つ以上を含まないと判定すると、エミュレーションプロセッサシステム１２０６の基本ブロックコンパイラは、ユーザ入力１４５４を処理するためのエミュレーションプロセッサシステム１２０６の１つ以上のメモリデバイスからエミュレートされたＰＵコード命令１～Ｍのうちの１つ以上にアクセスし、エミュレートされたＰＵコード命令１～Ｍの１つ以上をコンパイルして、ユーザ入力１４５４を処理するための基本ブロック１～ｎの１つ以上を生成する。エミュレーションプロセッサシステム１２０６の基本ブロックコンパイラは、エミュレーションプロセッサシステム１２０６のブロックディスパッチャに基本ブロック１～ｎのうちの１つ以上を送信する。１つ以上の基本ブロック１～ｎを受信すると、エミュレーションプロセッサシステム１２０６のブロックディスパッチャは、基本ブロック１～ｎのうちの１つ以上を実行して、ユーザ入力１４５４に基づいて１つ以上の画像フレームを生成する。一方で、キャッシュ１２０８が、ユーザ入力１４５４を処理するための基本ブロック１～ｎの１つ以上を含むと判定すると、エミュレーションプロセッサシステム１２０６の基本ブロックコンパイラは、エミュレーションプロセッサシステム１２０６のブロックディスパッチャに命令を送信する。命令を受信することに応答して、エミュレーションプロセッサシステム１２０６のブロックディスパッチャは、ユーザ入力１４５４を処理するために基本ブロック１～ｎの１つ以上を実行する。

【０２７６】

図１９Ａは、画像１９００のスクリーンショットであり、ハンドヘルドコントローラ４１４またはホイールコントローラ１７０２（図１７）またはハンドヘルドコントローラ１

２１２（図１８）などの更新されたハンドヘルドコントローラのタイプ及びサブタイプの受信を示している。画像１９００は、レガシーゲームＮの仮想環境４５２（図４Ｂ）を含む。

【０２７７】

レガシーゲームＮのプレイ中またはプレイ前に、ユーザ１は、第１の更新されたハンドヘルドコントローラを入力プロセッサシステム４０７から切り離し、第２の更新されたハンドヘルドコントローラを入力プロセッサシステム４０７に結合する。次に、ユーザ１は、第２の更新されたハンドヘルドコントローラの１つ以上の更新された入力デバイスを操作して、設定オプションを含むメニュー１９０２にアクセスする。エミュレーションプロセッサシステム４０９のＣＰＵ９２２（図９Ｂ）などのＣＰＵは、メニュー１９０２をディスプレイデバイス４１０（図４Ｂ）のディスプレイ画面に表示するように、エミュレーションプロセッサシステム４０９のＧＰＵ９２４（図９Ｂ）などのＧＰＵに指示する。ユーザ１はさらに、第２の更新されたハンドヘルドコントローラの１つ以上の更新された入力デバイスを操作して、設定オプションを選択する。設定オプションが選択されると、エミュレーションプロセッサシステム４０９のＣＰＵは、ディスプレイデバイス４１０（図４Ｂ）のディスプレイ画面に別のメニューを表示するように、エミュレーションプロセッサシステム４０９のＧＰＵに指示する。他のメニューには、制御と周辺機器のオプションが含まれている。次に、ユーザ１は、第２の更新されたハンドヘルドコントローラの１つ以上の更新された入力デバイスを操作して、制御及び周辺機器オプションを選択する。制御及び周辺機器オプションが選択されると、エミュレーションプロセッサシステム４０９のＣＰＵは、エミュレーションプロセッサシステム４０９のＧＰＵに指示して、ディスプレイデバイス４１０のディスプレイ画面にメニュー１９１２（図１９Ｂ）を表示する。

【０２７８】

図１９Ｂは、メニュー１９１２を説明するための画像１９１０のスクリーンショットである。メニュー１９１０は、ボタンを再割り当てするためのボタンの再割り当てオプションと、ガン形状コントローラが入力プロセッサシステム４０７と共に第２の更新されたハンドヘルドコントローラとして使用されることを入力プロセッサシステム４０７（図４Ｂ）に示すための銃型ハンドヘルドコントローラオプションと、ホイールコントローラが入力プロセッサシステム４０７（図４Ｂ）と共に第２の更新されたハンドヘルドコントローラとして使用されることを入力プロセッサシステム４０７に示すためのホイールハンドヘルドコントローラオプションと、を含む。銃型コントローラは、更新されたハンドヘルドコントローラのタイプの一例である。

【０２７９】

ユーザ１は、第２の更新されたハンドヘルドコントローラの１つ以上の更新された入力デバイスを操作して、ホイールハンドヘルドコントローラオプションを選択する。ホイールハンドヘルドコントローラオプションが選択されると、エミュレーションプロセッサシステム４０９のＣＰＵは、第１のサブタイプコントローラオプション及び第２のサブタイプコントローラオプションを含むメニューをディスプレイデバイス４１０のディスプレイ画面に表示するようにエミュレーションプロセッサシステム４０９のＧＰＵに命令する。第１のサブタイプコントローラ及び第２のサブタイプコントローラは、ホイールコントローラのサブタイプの例である。

【０２８０】

ユーザ１は、第２の更新されたハンドヘルドコントローラの１つ以上の更新された入力デバイスを操作して、第２のサブタイプコントローラオプションを選択する。第２のサブタイプコントローラオプションの選択を受信すると、エミュレーションプロセッサシステム４０９のＣＰＵは、レガシーゲームＮのプレイを継続または開始するようにエミュレーションプロセッサシステム４０９のＧＰＵに指示する。

【０２８１】

図２０は、デフォルトマッピングなどのマッピング２００２を説明するための画像２０００のスクリーンショットの実施形態である。画像２０００は、レガシーゲームＮの仮想

環境４５２を含む。マッピング２００２は、デフォルトのマッピング１３０８（図１３）の例である。レガシーゲームＮのプレイ中またはプレイ前に、エミュレーションプロセッサシステム４０９のＣＰＵは、ディスプレイデバイス４１０のディスプレイ画面にマッピング２００２を表示するようにエミュレーションプロセッサ４０９のＧＰＵに指示する。

【０２８２】

マッピング２０２２は、レガシーハンドヘルドコントローラの入力デバイスと更新されたハンドヘルドコントローラの入力デバイスとの間の対応または１対１の関係などの固有の関係を含む。例えば、三角形ボタンなどのレガシーハンドヘルドコントローラのレガシー入力デバイス１は、三角形ボタンなどの更新されたハンドヘルドコントローラの更新された入力デバイス１にマッピングされる。別の例として、更新されたハンドヘルドコントローラのタッチパッド１６０２（図１６Ａ）などのレガシー入力デバイス２は、レガシーハンドヘルドコントローラの選択ボタンなどの更新された入力デバイス２と対応している。さらに別の例として、レガシーハンドヘルドコントローラのレガシー入力デバイス３は、更新されたハンドヘルドコントローラの更新された入力デバイス２と１対１の関係を有する。

【０２８３】

図２１Ａは、ユーザ１からの更新されたグラフィックスパラメータの受信を説明するためのシステム２１００の実施形態の図である。システム２１００は、ディスプレイデバイス４１０、ハンドヘルドコントローラ４１４、及びゲームコンソール４０２を含む。

【０２８４】

ゲームコンソール４０２は、エミュレーションプロセッサシステム４０９、メモリデバイス４０６、及びＧＰＵメモリデバイス２１０４を含む。エミュレーションプロセッサシステム４０９は、ＣＰＵ９２２及びＧＰＵ９２４を含む。ＧＰＵメモリデバイス２１０４は、バス９３０に結合されている。ＧＰＵメモリデバイス２１０４は、メモリシステム９２６（図９Ｂ）のメモリデバイスの一例である。

【０２８５】

ユーザ１がユーザアカウント１にログインした後、レガシーゲームＮのプレイ中またはプレイ前に、ＧＰＵ９２４は、ＵＧＰ１及びＵＧＰ２などの１つ以上の更新されたグラフィックパラメータのリストを表示するようにディスプレイデバイス４１０を制御する。例えば、レガシーゲームＮのプレイ中に、ＧＰＵ９２４は、更新されたグラフィックパラメータＵＧＰ１及びＵＧＰ２をディスプレイデバイス４１０上にレンダリングする。

【０２８６】

更新されたグラフィックパラメータがディスプレイデバイス４１０に表示された後、ユーザ１は、ハンドヘルドコントローラ４１４の１つ以上の更新された入力デバイスを操作して、１つ以上の更新されたグラフィックパラメータを選択し、ハンドヘルドコントローラ４１４は、操作に基づいてユーザ入力２１０２を生成する。ユーザ入力２１０２は、１つ以上の更新されたグラフィックパラメータの選択の指示を含む。ユーザ入力２１０２は、ハンドヘルドコントローラ４１４からゲームコンソール４０２のＣＰＵ９２２に送信される。ＣＰＵ９２２は、１つ以上の更新されたグラフィックパラメータをユーザ入力２１０２から取得し、１つ以上の更新されたグラフィックパラメータをＧＰＵ９２４に提供する。ＧＰＵ９２４は、１つ以上のレガシーグラフィックパラメータを１つ以上のグラフィックパラメータにエミュレートする。

【０２８７】

一実施形態では、任意数の更新されたグラフィックパラメータがディスプレイデバイス４１０に表示される。

【０２８８】

図２１Ｂ－１は、１つ以上のレガシーグラフィックパラメータの１つ以上の更新されたグラフィックパラメータへのエミュレーションを説明するためのシステム２１２０の実施形態の図である。システム２１２０は、ハンドヘルドコントローラ４１４及びゲームコンソール４０２を含む。

【０２８９】
ＧＰＵ９２４には、ＧＰＵエミュレータ２１２２が含まれている。一例として、本明細書で説明するように、ＧＰＵエミュレータは、ハードウェアもしくはソフトウェア、またはそれらの組み合わせである。説明すると、ＧＰＵエミュレータは、プロセッサ、ＰＬＤ、ＡＳＩＣ、またはコントローラである。別の例として、ＧＰＵエミュレータはコンピュータプログラムである。
【０２９０】
１つ以上の更新されたグラフィックパラメータの選択の指示がハンドヘルドコントローラ４１４から受信されると、ＣＰＵ９２２は、更新されたグラフィックパラメータをＧＰＵ９２４に提供する。ＧＰＵ９２４は、更新されたグラフィックパラメータをＧＰＵメモリデバイス２１０４内に格納する。ユーザ入力４２０に応答して、ＧＰＵ９２４は、更新されたコードの基本ブロック１～ｎのうちの１つ以上を実行して、レガシーグラフィックパラメータＬＧＰ１及びレガシーグラフィックパラメータＬＧＰ２などの１つ以上のレガシーグラフィックパラメータが、レガシーゲームＮの画像フレーム２１２３内の仮想オブジェクト及び仮想背景に適用されていると判定する。画像フレーム２１２３は、更新された画像フレームである。１つ以上の更新されたグラフィックパラメータを適用するためのユーザ入力２１０２が受信される前に、ユーザ入力４２０が生成されるユーザ入力４１９が受信されることに留意されたい。
【０２９１】
ＧＰＵ９２４によって、１つ以上のレガシーグラフィックパラメータを判定して、画像フレーム４２２（図４Ｂ）などの１つ以上のレガシー画像フレームを生成すると、ＧＰＵエミュレータ２１２２は、１つ以上のレガシーグラフィックパラメータの代わりに１つ以上の更新されたグラフィックパラメータをレガシー画像フレームに適用し、画像フレーム２１２３を出力する。１つ以上の更新されたグラフィックパラメータの適用は、レガシーゲームＮの仮想環境の画像フレームのレンダリングと呼ばれることがある。例えば、ＧＰＵエミュレータ２１２２は、ＧＰＵメモリデバイス２１０４から、更新されたグラフィックパラメータＵＧＰ１をレガシーグラフィックパラメータＬＧＰ１に対応するものとして、かつ、更新されたグラフィックパラメータＵＧＰ２をレガシーグラフィックパラメータＬＧＰ２に対応するものとして識別する。この例では、更新されたグラフィカルパラメータＵＧＰ１とレガシーグラフィカルパラメータＬＧＰ１は同じタイプである。説明すると、更新されたグラフィカルパラメータＵＧＰ１は、仮想オブジェクトの仮想画像の更新された解像度であり、レガシーグラフィカルパラメータＬＧＰ１は、仮想オブジェクトの仮想画像のレガシー解像度である。別の例として、更新されたグラフィカルパラメータＵＧＰ２は、３２ビットカラーパレットから仮想オブジェクトに適用される色のセットであり、レガシーグラフィカルパラメータＬＧＰ２は、１６ビットカラーパレットから仮想オブジェクトに適用される色のセットである。別の例として、グラフィカルパラメータＬＧＰ１とＵＧＰ１の両方は、あるタイプのグラフィカルパラメータの同じ識別子（英数字コードなど）を含む。この例では、ＧＰＵエミュレータ２１２２は、グラフィカルパラメータＵＧＰ１及びＬＧＰ１の両方が同じ識別子を含むとの判定に応答して、更新されたグラフィカルパラメータＵＧＰ１がレガシーグラフィカルパラメータＬＧＰ１に対応していると判定する。ＧＰＵ９２４は、画像フレーム２１２３に従って、仮想オブジェクトを有する仮想環境及び仮想環境４５２の仮想背景などの仮想環境の画像をディスプレイデバイス４１０上に表示する。例えば、ＧＰＵ９２４は、ディスプレイデバイス４１０の発光体を制御して、画像フレーム２１２３に従って画像を表示する。仮想環境の画像は、レガシーゲームＮのプレイを表示するためにディスプレイデバイス４１０上に表示される。
【０２９２】
一実施形態では、レガシーＧＰＵ９０４（図９Ａ）は、エミュレートされたＰＵコード命令１～Ｍのうちの１つ以上を実行する。エミュレートされたＰＵコード命令１～Ｍを実行すると、更新されたグラフィックパラメータは生成され得ない。むしろ、レガシーグラフィカルパラメータが生成される。

【０２９３】
一実施形態では、ＣＰＵ９２２は、基本ブロックコンパイラ１０４を実行し、ＧＰＵ９２４は、ブロックディスパッチャ３０２を実行する。
【０２９４】
一実施形態では、ＣＰＵ９２２は、基本ブロックコンパイラ１０４及びブロックディスパッチャ３０２を実行する。
【０２９５】
一実施形態では、更新されたグラフィカルパラメータは、レガシーグラフィカルパラメータに対応していない。例えば、更新されたグラフィカルパラメータＵＰＧ１に対応するレガシーグラフィカルパラメータはない。レガシーグラフィックパラメータのいずれかを適用する代わりに、ＧＰＵエミュレータ２１２２は、レガシーゲームＮのプレイ中に実行されるレンダリング操作中に、更新されたグラフィックパラメータＵＰＧ１を適用する。
【０２９６】
一実施形態では、ＧＰＵエミュレータ２１２２は、レガシー画像フレームに対するニアレスト、またはバイリニア、またはバイキュービック、またはそれらの組み合わせなどの補間操作を適用して、レガシーグラフィックパラメータが不明なフレームのためにレガシー画像のレガシー画像ピクセルに適用される更新されたグラフィックパラメータを推定する。例えば、ＧＰＵエミュレータ２１２２は、仮想環境４５２の第１の画像ピクセルの色及び強度値を仮想環境４５２の第２の画像ピクセルに補間して、第１及び第２の画像ピクセルを有する更新された画像フレームを出力する。この例では、更新されたコードの１つ以上の基本ブロック１～ｎの実行時に、第２の画像ピクセルの色や強度の値などのレガシーグラフィックパラメータは不明である。この例では、第１及び第２の画像ピクセルは、更新された画像フレーム２１２３の１つなど、更新された画像フレームの部分である。
【０２９７】
一実施形態では、ＧＰＵエミュレータ２１２２は、１つ以上の補間操作及び１つ以上の畳み込み操作をレガシー画像フレームに適用して、レガシー画像フレームのレガシー画像ピクセルに適用される更新されたグラフィックパラメータを推定する。例えば、バイキュービック補間を適用した後、ＧＰＵエミュレータ２１２２は、９Ｘ９畳み込み、１Ｘ１畳み込み、及び５Ｘ５畳み込みをレガシー画像フレームに適用して、更新された画像フレームを出力する。例えば、ｎＸｍ畳み込みにおいて、ＧＰＵエミュレータ２１２２は、レガシー画像フレームのｎＸｍ画像ピクセルを取得し、ｎＸｍ画像ピクセルをｎＸｍカーネルと畳み込み、更新されたグラフィカルパラメータの結果値を出力する。ここで、ｎ及びｍは正の整数であり、ｎは画像ピクセルまたはカーネルピクセルの行番号を表し、ｍは画像ピクセルのカーネルピクセルの列番号を表している。例えば、ｎは４であり、ｍは２である。別の例として、ｍとｎの両方が４である。ｎＸｍカーネルが適用されるレガシー画像フレームのｎＸ１ピクセルの更新されたグラフィックパラメータの値は、結果値に置き換えられる。
【０２９８】
一実施形態では、レガシー画像フレームの画像ピクセルに畳み込みを適用する代わりに、ＧＰＵエミュレータ２１２２は、画像ピクセルをサブピクセルに分割し、畳み込みをサブピクセルに適用する。この実施形態では、畳み込みは、画像ピクセルに畳み込みを適用する前または後にサブピクセルに適用される。
【０２９９】
一実施形態では、ＧＰＵ９２４は、深層学習手法を適用する。深層学習手法では、ＧＰＵエミュレータ２１２２は、レガシー画像フレームを受信し、比較画像フレームが格納されているデータベースにアクセスし、レガシー画像フレームを比較画像フレームと比較して、更新された画像フレームを出力する。例として、比較画像フレームは、レガシー画像フレームと同じ仮想オブジェクトが１つ以上を有している。この例では、比較画像フレームは比較戦闘機を有し、レガシー画像フレームは仮想オブジェクト４５４を有する。この例を続けると、ＧＰＵエミュレータ２１２２は、比較戦闘機を仮想オブジェクト４５４と

比較して、仮想オブジェクト４５４が比較戦闘機に基づいて変更されるかどうかを判定する。この例をさらに続けると、ＧＰＵエミュレータ２１２２は、仮想オブジェクト４５４の少なくとも一部分（翼など）が、比較戦闘機の少なくとも対応する部分（翼など）と置き換えられると判定する。この例では、ＧＰＵエミュレータ２１２２から判定を受信すると、ＧＰＵ９２４は、仮想オブジェクト４５４の少なくとも一部分を比較戦闘機の少なくとも対応する部分に置き換えて、結果の戦闘機を出力し、更新された画像フレームをさらに出力する。更新された画像フレームは、ＧＰＵ９２４によってデータベース内に格納される。データベースはＧＰＵメモリデバイス２１０４内に格納されていることに留意されたい。

【０３００】

一実施形態では、ＧＰＵは再帰的学習手法を適用する。再帰学習手法には深層学習手法が含まれる。また、再帰学習手法では、ＧＰＵエミュレータ２１２２は、追加のレガシー画像フレームを受信し、更新された画像フレームが格納されているデータベースにアクセスし、追加のレガシー画像フレームを更新された画像フレームと比較して、追加の更新された画像フレームを出力する。一例として、更新された画像フレームは、追加のレガシー画像フレームのものと同じ仮想オブジェクトを有している。この例では、更新された画像フレームは、結果の戦闘機を有しており、追加のレガシー画像フレームには仮想オブジェクト４５４を有している。この例を続けると、ＧＰＵエミュレータ２１２２は、結果の戦闘機を仮想オブジェクト４５４と比較して、仮想オブジェクト４５４が、結果の戦闘機に基づいて修正されるかどうかを判定する。この例をさらに続けると、ＧＰＵエミュレータ２１２２は、仮想オブジェクト４５４の少なくとも一部（翼など）が、結果の戦闘機の少なくとも対応する部分（翼など）と置き換えられるべきであると判定する。この例では、ＧＰＵエミュレータ２１２２から判定を受信すると、ＧＰＵ９２４は、仮想オブジェクト４５４の少なくとも一部分を結果の戦闘機の少なくとも対応する部分に置き換えて、追加の結果の戦闘機を出力し、さらに追加の更新された画像フレームを出力する。追加の更新された画像フレームは、次のレガシー画像フレームに適用するためにＧＰＵ９２４によってデータベース内に格納される。次のレガシー画像フレームは、ＧＰＵエミュレータ２１２２によって、追加のレガシー画像フレームの受信に続いて受信される。

【０３０１】

図２１Ｂ－２は、ブロックディスパッチャ３０２から出力されたレガシー画像フレームなどの画像フレーム４２２からの更新された画像フレーム２１２３の生成を説明するためのシステム２１３０の実施形態の図である。システム２１３０は、ブロックディスパッチャ３０２及びＧＰＵエミュレータ２１２２を含む。ブロックディスパッチャ３０２は、ＧＰＵエミュレータ２１２２に結合されている。ブロックディスパッチャ３０２は、更新されたコードの基本ブロック１～ｎのうちの１つ以上を実行して、レガシーグラフィックパラメータを含む画像フレーム４２２を生成する。ＧＰＵエミュレータ２１２２は、画像フレーム４２２を受信し、画像フレーム４２２を解析してレガシーグラフィックパラメータを識別し、レガシーグラフィックパラメータを更新されたグラフィックパラメータに変換して、更新された画像フレーム２１２３を出力する。

【０３０２】

一実施形態では、各画像フレームは画像ファイルである。例えば、第１の画像フレームは第１の画像ファイルであり、第２の画像フレームは第２の画像ファイルである。

【０３０３】

図２１Ｃ－１は、レガシー及び更新されたグラフィックパラメータのリストの実施形態、ならびにレガシー及び更新されたグラフィックパラメータ間の対応である。レガシーグラフィックパラメータＬＧＰ１またはＬＧＰ２の一例は、仮想環境の仮想画像の６４０Ｘ４８０ピクセル解像度、７２０Ｘ４８０ピクセル解像度、及び７２０Ｘ５８６ピクセル解像度などのレガシー画像解像度である。この例では、レガシーグラフィックパラメータＬＧＰ１に対応する更新されたグラフィックパラメータＵＧＰ１、またはレガシーグラフィックパラメータＬＧＰ２に対応する更新されたグラフィックパラメータＵＧＰ２は、更新

された画像解像度である。説明すると、更新された解像度は、レガシーグラフィックパラメータのピクセル解像度の４倍である高解像度（ＨＤ）解像度と、レガシーグラフィックパラメータのピクセル解像度の８倍である４Ｋ解像度を含む。

【０３０４】

レガシーグラフィカルパラメータＬＧＰ１またはＬＧＰ２の別の例は、レガシーラインの太さである。レガシーラインは、レガシーゲームＮの仮想シーンの１つ以上の仮想オブジェクトの１つ以上の部分と、仮想シーンの仮想背景の一部分のライン画像である。この例では、レガシーグラフィカルパラメータＬＧＰ１に対応する更新されたグラフィカルパラメータＵＧＰ１、またはレガシーグラフィカルパラメータＬＧＰ２に対応する更新されたグラフィカルパラメータＵＧＰ２は、更新されたライン太さである。説明として、更新されたライン太さは、レガシーライン太さの倍数または係数である。

【０３０５】

レガシーグラフィカルパラメータＬＧＰ１またはＬＧＰ２のさらに別の例は、レガシーラインカウントであり、これは、レガシーゲームＮの仮想環境の画像を形成するピクセルを有するレガシーラインの数によって規定される。この例では、レガシーグラフィカルパラメータＬＧＰ１に対応する更新されたグラフィカルパラメータＵＧＰ１、またはレガシーグラフィカルパラメータＬＧＰ２に対応する更新されたグラフィカルパラメータＵＧＰ２は、更新されたラインカウントである。説明として、更新されたラインカウントは、レガシーラインカウントの倍数または係数である。

【０３０６】

更新されたグラフィカルパラメータＵＧＰ１またはＵＧＰ２のさらに別の例は、更新されたスキャンライン太さであり、これは、レガシーゲームＮの仮想環境の仮想画像の画像フレームにオーバーレイされる各スキャンライン太さによって規定されている。説明すると、各スキャンラインは、レガシー画像フレームにオーバーレイされる。各スキャンラインは、仮想画像の一方の端から仮想イメージの反対側の端まで延在するようにオーバーレイされる。この例では、更新されたスキャンライン太さに対応する対応するレガシーグラフィカルパラメータがないことに留意されたい。

【０３０７】

更新されたグラフィカルパラメータＵＧＰ１またはＵＧＰ２の別の例は、更新されたスキャンラインカウントであり、これは、レガシーゲームＮの仮想環境の仮想イメージにオーバーレイされるスキャンラインの数によって規定される。この例では、更新されたスキャンラインカウントに対応する対応するレガシーグラフィカルパラメータがないことに留意されたい。

【０３０８】

しかし、レガシーグラフィカルパラメータＬＧＰ１またはＬＧＰ２の例は、レガシーアンチエイリアシングフィルタとレガシービデオフィルタである。レガシーアンチエイリアシングフィルタ及びレガシービデオフィルタは、レガシーゲームＮの仮想環境の仮想画像のピクセル内で出力され、かつ明らかな、ぼやけた色や輝度などのノイズを低減または除去する。この例では、レガシーグラフィカルパラメータＬＧＰ１に対応する更新されたグラフィカルパラメータＵＧＰ１、またはレガシーグラフィカルパラメータＬＧＰ２に対応する更新されたグラフィカルパラメータＵＧＰ２は、更新されたアンチエイリアシングフィルタ、及び更新されたビデオフィルタである。説明として、更新されたアンチエイリアシングフィルタ及びビデオフィルタは、レガシーアンチエイリアシングフィルタ及びレガシービデオフィルタがノイズを除去または低減する解像度よりも高い解像度でノイズを低減または除去する。この説明では、レガシーゲームＮの仮想オブジェクトを表すために使用されるポリゴンのポリゴンエッジの平滑化によって、または、レガシーゲームＮの仮想シーンの仮想背景のピクセルの輝度を補間することによって、ノイズが除去または低減されている。

【０３０９】

レガシーグラフィックパラメータＬＧＰ１またはＬＧＰ２のさらに別の例は、レガシー

カラースペースまたはレガシーカラーパレットなどのレガシーカラースケールであり、これは、レガシーカラースケール内の各色を表すビット数によって定義される。レガシー色空間の説明は、ビデオグラフィックアレイ（ＶＧＡ）色空間である。レガシーカラーパレット内の各色は１６ビットで表されている。レガシーカラーパレット内の色は、レガシーゲームＮの仮想環境の画像を表すために適用され得る。この例では、レガシーグラフィカルパラメータＬＧＰ１に対応する更新されたグラフィカルパラメータＵＧＰ１、またはレガシーグラフィカルパラメータＬＧＰ２に対応する更新されたグラフィカルパラメータＵＧＰ２は、更新されたカラースケールである。説明として、更新されたカラースケールは、３２ビットカラーパレットである。更新されたカラースケールの各色は、１６ビットではなく３２ビットで表される。

【０３１０】

レガシーグラフィカルパラメータＬＧＰ１またはＬＧＰ２の別の例は、２０フレーム／秒（ｆｐｓ）または３０ｆｐｓなどのレガシーレンダリングレートであり、このレンダリングレートで、レガシー画像フレームがレンダリングされ、ディスプレイデバイス４１０上に表示される。この例では、レガシーグラフィカルパラメータＬＧＰ１に対応する更新されたグラフィカルパラメータＵＧＰ１、またはレガシーグラフィカルパラメータＬＧＰ２に対応する更新されたグラフィカルパラメータＵＧＰ２は、更新されたレンダリングレートであり、これはレガシーレンダリングレートの倍数である。説明として、更新されたレンダリングレートは、６０ｆｐｓまたは１２０ｆｐｓであり、このレンダリングレートで、更新された画像フレームがレンダリングされ、ディスプレイデバイス４１０上に表示される。

【０３１１】

レガシーグラフィックパラメータＬＧＰ１またはＬＧＰ２のさらに別の例は、レガシーアップスケールファクタであり、これは、レガシー画像解像度と、レガシーマシン９００（図９Ａ）で使用される陰極線管などの標準テレビのサイズによって定義される。レガシーアップスケール係数は、仮想オブジェクトまたは仮想背景の画像を拡大縮小して、レガシー解像度が画像に適用された後、標準のテレビの表示ピクセルを塗りつぶす。この例では、レガシーグラフィカルパラメータＬＧＰ１に対応する更新されたグラフィカルパラメータＵＧＰ１、またはレガシーグラフィカルパラメータＬＧＰ２に対応する更新されたグラフィカルパラメータＵＧＰ２は、更新されたアップスケール係数である。説明すると、更新されたアップスケール係数には、ＨＤ解像度または４Ｋ解像度が含まれる。また、この説明では、更新されたアップスケール係数は、仮想オブジェクトまたは仮想背景の画像をスケーリングし、ＨＤディスプレイデバイスまたは４Ｋディスプレイデバイスの表示ピクセルを塗りつぶす。ＨＤディスプレイデバイス及び４Ｋディスプレイデバイスは、ディスプレイデバイス４１０の例である。

【０３１２】

図２１Ｃ－２は、レンダリング操作中に仮想オブジェクトの解像度を変更するためにＧＰＵエミュレータ２１２２（図２１Ｂ－１）によって実行される方法の実施形態を説明するための図である。ＧＰＵエミュレータ２１２２がない場合、レガシーゲームＮの仮想シーンにおける仮想オブジェクト２１５２の位置及び方向２１５０は、より大きくなるようにサイズが再変更され、４色ピクセルを使用してＧＰＵ９２４によってレンダリングされる。例えば、仮想オブジェクト２１５２は、ＧＰＵエミュレータ２１２２を含まないレガシーＧＰＵ９０４（図９Ａ）によってレンダリングされる。

【０３１３】

ＧＰＵエミュレータ２１２２が使用されるとき、位置及び向き２１５０は、サイズ変更され、レガシーゲームＮの仮想シーンにおいて仮想オブジェクト２１５４としてレンダリングされる。一例として、仮想オブジェクト２１５４は、仮想オブジェクト２１５２よりも高い解像度を有する。説明すると、仮想オブジェクト２１５４は、仮想オブジェクト２１５２よりも多数の画像ピクセルを有する。さらに説明するために、ＧＰＵエミュレータ２１２２は、仮想オブジェクト２１５２の１つ以上のより大きなカラーピクセルを、より

小さなピクセルの１つ以上の対応するセットで効果的に置き換えて、仮想オブジェクト２１５４を出力する。別の例として、仮想オブジェクト２１５４は、仮想オブジェクト２１５２の輪郭２１５８と比較して、より顕著な輪郭２１５６を有する。さらに別の例として、仮想オブジェクト２１５４は、仮想オブジェクト２１５２と比較して漫画のように見える。説明すると、仮想オブジェクト２１５４は、２次元仮想オブジェクトとしてレンダリングされ、仮想オブジェクト２１５２は、３次元仮想オブジェクトとしてレンダリングされる。別の例として、輪郭２１５６は、輪郭２１５８と比較してより滑らかなエッジ及び頂点を有する。別の例として、仮想オブジェクト２１５４は、仮想オブジェクト２１５２と比較してより高いコントラストを有する。さらに別の例として、仮想オブジェクト２１５４は、仮想オブジェクト２１５２の強度レベルと比較して、より多様な強度レベルを有する。

【０３１４】

図２１Ｄは、更新されたグラフィックパラメータに基づいて更新された画像フレーム２１６０を生成するための方法の実施形態を説明するための図である。ＣＰＵ９２２（図２１Ｂ－１）は、ゲームコードＧＣＮの基本ブロック１～ｎのうちの１つ以上を実行して、レガシーゲームＮの仮想シーンにおける仮想オブジェクト及び仮想背景の位置及び方向を判定する。画像フレーム２１６０は、画像フレーム２１２３の１つの例である（図２１Ｂ－１）。ＧＰＵ９２４（図２１Ｂ－１）は、ゲームコードＧＣＮの基本ブロック１～ｎのうちの１つ以上を実行して、レガシーグラフィックパラメータを有する画像フレーム４２２などのレガシー画像フレームを生成する。レガシーグラフィックパラメータは、レガシーゲームＮの仮想シーンにおける仮想オブジェクト及び仮想背景の位置及び方向にＧＰＵ９２４によって適用される。ＧＰＵ９２４は、仮想環境４５２の画像を表示するためにレガシー画像フレームをディスプレイデバイス４１０に提供する代わりに、ＧＰＵエミュレータ２１２２を適用（実行するなど）して、レガシーグラフィックパラメータに対応する更新されたグラフィックパラメータを判定する。ＧＰＵ９２４は、レガシーグラフィックパラメータの代わりに更新されたグラフィックパラメータに基づいて更新された画像フレーム２１６０を生成し、レガシーゲームＮの画像を表示するために更新された画像フレームをディスプレイデバイス４１０に送信する。

【０３１５】

図２２Ａは、ユーザ１からの更新されたオーディオパラメータの受信を説明するためのシステム２２００の実施形態の図である。システム２２００は、ディスプレイデバイス４１０、ハンドヘルドコントローラ４１４、及びゲームコンソール４０２を含む。

【０３１６】

ゲームコンソール４０２は、エミュレーションプロセッサシステム４０９及びオーディオメモリデバイス２２０６を含む。エミュレーションプロセッサシステム４０９は、オーディオエミュレータ２２０４を含むＡＰＵ９３１を含む。一例として、本明細書で説明するオーディオエミュレータは、ハードウェアもしくはソフトウェア、またはそれらの組み合わせである。説明すると、オーディオエミュレータは、プロセッサ、またはＰＬＤ、またはＡＳＩＣ、またはコントローラである。別の説明として、オーディオエミュレータはコンピュータプログラムである。

【０３１７】

オーディオメモリデバイス２２０６及びＡＰＵ９３１は、バス９３０に結合されている。一例として、ＡＰＵ９３１は、更新されたマシン９２０内に実装されている。また、一例として、オーディオメモリデバイス２２０６は、メモリシステム９２６（図９Ｂ）内に実装される。

【０３１８】

ユーザ１がユーザアカウント１にログインした後、レガシーゲームＮのプレイ中またはプレイ前に、ＧＰＵ９２４は、ディスプレイデバイス４１０を制御して、ＵＡＰ１及びＵＡＰ２などの１つ以上の更新されたオーディオパラメータのリストを表示する。例えば、レガシーゲームＮのプレイ中に、ＧＰＵ９２４は、更新されたオーディオパラメータＵＡ

Ｐ１及びＵＡＰ２をディスプレイデバイス４１０上にレンダリングする。

【０３１９】

更新されたオーディオパラメータがディスプレイデバイス４１０に表示された後、ユーザ１は、ハンドヘルドコントローラ４１４の１つ以上の更新された入力デバイスを操作して、１つ以上の更新されたオーディオパラメータを選択し、ハンドヘルドコントローラ４１４は、操作に基づいてユーザ入力２２０８を生成する。ユーザ入力２２０８は、１つ以上の更新されたオーディオパラメータの選択の指示を含む。ユーザ入力２２０８は、ハンドヘルドコントローラ４１４からＣＰＵ９２２に送られる。ＣＰＵ９２２は、ユーザ入力２２０８から１つ以上の更新されたオーディオパラメータを取得し、１つ以上の更新されたオーディオパラメータをＡＰＵ９３１のＡＰＵエミュレータ２２０４に送信する。ＡＰＵエミュレータ２２０４は、１つ以上のレガシーオーディオパラメータを１つ以上の更新されたオーディオパラメータにエミュレートする。

【０３２０】

一実施形態では、任意数の更新されたオーディオパラメータがディスプレイデバイス４１０に表示される。

【０３２１】

図２２Ｂ－１は、１つ以上のレガシーオーディオパラメータの１つ以上の更新されたオーディオパラメータへのエミュレーションを説明するためのシステム２２５０の実施形態の図である。システム２１２０は、ハンドヘルドコントローラ４１４及びゲームコンソール４０２を含む。

【０３２２】

１つ以上の更新されたオーディオパラメータの選択の指示が、ハンドヘルドコントローラ４１４からのユーザ入力２２０８を介して受信されると、ＣＰＵ９２２は、バス９３０を介してＡＰＵ９３１に更新されたオーディオパラメータを提供する。ＡＰＵ９３１は、更新されたグラフィックパラメータをオーディオメモリデバイス２２０６内に格納する。ユーザ入力４２０に応答して、ＡＰＵ９３１は、更新されたコードの基本ブロック１～ｎのうちの１つ以上を実行して、レガシーグラフィックパラメータＬＡＰ１及びレガシーグラフィックパラメータＬＡＰ２などの１つ以上のレガシーグラフィックパラメータが、レガシーゲームＮの画像フレーム２１２３内の仮想オブジェクト及び仮想背景に適用されていると判定する。ＡＰＵ９３１はさらに、画像フレーム２１２３（図２１Ｂ－１）と同期して出力される１つ以上のレガシーオーディオパラメータに基づいて、１つ以上のレガシーオーディオフレームを生成する。１つ以上の更新されたオーディオパラメータを適用するためのユーザ入力２２０８が受信される前に、ユーザ入力４２０が生成されるユーザ入力４１９が受信されることに留意されたい。

【０３２３】

ＡＰＵ９３１によって、画像フレーム２１２３と共に出力されるレガシーオーディオフレームが生成されると、オーディオエミュレータ２２０４は、更新されたオーディオフレームである１つ以上のオーディオフレーム２２５２を生成するための１つ以上のレガシーオーディオパラメータの代わりに、１つ以上の更新されたオーディオパラメータをレガシーオーディオフレームに適用する。１つ以上の更新されたオーディオパラメータの適用は、レガシーゲームＮの仮想環境で出力されるオーディオフレームの生成と呼ばれることがある。例えば、オーディオエミュレータ２２０４は、オーディオメモリデバイス２２０６から、更新されたオーディオパラメータＵＡＰ１をレガシーオーディオパラメータＬＡＰ１に対応するものとして、かつ、更新されたオーディオパラメータＵＡＰ２をレガシーオーディオパラメータＬＡＰ２に対応するものとして識別する。この例では、更新されたオーディオパラメータＵＡＰ１とレガシーオーディオパラメータＬＡＰ１は同じタイプである。説明すると、更新されたオーディオパラメータＵＡＰ１は、更新されたサンプリングレートであり、これに基づいて、レガシーゲームＮの仮想オブジェクトの仮想イメージからサウンドが出力され、レガシーオーディオパラメータＬＡＰ１は、レガシーサンプリングレートであり、これに基づいてサウンドが仮想イメージから出力され得る。更新された

サンプリングレートは、レガシーサンプリングレートよりも大きくなっている。さらに説明すると、レガシーサンプリングレートは、１０，０００サンプル／秒であり、更新されたサンプリングレートは、９６，０００サンプル／秒である。別の説明として、更新されたオーディオパラメータＵＡＰ２は、レガシーゲームＮの仮想オブジェクトからステレオ、モノラル、またはサラウンドサウンドとしてサウンドが出力されることを可能にする。この説明では、レガシーオーディオパラメータＬＧＰ２は、仮想オブジェクトからステレオとしてのみサウンドが出力されることを可能にする。さらに別の説明として、更新されたオーディオパラメータＵＡＰ１は、レガシーオーディオパラメータＬＡＰ１のオーディオサウンドのサンプルスペースと比較して、オーディオサウンドのサンプルスペースが大きくなっている。より大きなサンプルスペースには、レガシーオーディオパラメータＬＡＰ１のオーディオサウンドのサンプルスペースと比較して、より多様なオーディオサウンドがある。別の説明として、更新されたオーディオパラメータＵＡＰ２は、レガシーオーディオパラメータＬＡＰ２と比較して異なるリバーブを有している。さらに説明するために、更新されたオーディオパラメータＵＡＰ２は、レガシーオーディオパラメータＬＡＰ２に基づいて出力されるエコーの量と比較してより多くか、またはより少ないエコーを示す。

【０３２４】

別の例として、オーディオパラメータＬＡＰ１とＵＡＰ１の両方は、オーディオパラメータのタイプの同じ識別子（英数字コードなど）を含む。この例では、オーディオエミュレータ２２０４は、オーディオパラメータＵＡＰ１及びＬＡＰ１の両方が同じ識別子を含むとの判定に応答して、更新されたオーディオパラメータＵＡＰ１がレガシーオーディオパラメータＬＡＰ１に対応すると判定する。ＧＰＵ９２４は、画像フレーム２１２３を送信し、ＡＰＵ９３１は、音声フレーム２２５２をディスプレイデバイス４１０に送信して、画像フレーム２１２３に従って、かつ、オーディオフレーム２２５２に基づいたサウンド出力に従って、仮想環境（仮想環境４５２など）の画像をディスプレイデバイス４１０上に表示する。例えば、ディスプレイデバイス４１０のＧＰＵは、ディスプレイデバイス４１０の発光体を制御して画像フレーム２１２３に従って画像を表示し、ディスプレイデバイス４１０のＡＰＵは、ディスプレイデバイス４１０のスピーカーを制御してオーディオフレーム２２５２に従ってサウンドを出力する。仮想環境の画像は、レガシーゲームＮのプレイを表示するためにディスプレイデバイス４１０上に表示される。

【０３２５】

一実施形態では、オーディオエミュレータ２２０４を含まないＡＰＵ９３１は、レガシーマシン９００（図９Ａ）の構成要素である。例えば、ＡＰＵ９３１は、メモリデバイス９０６に結合されている。別の例として、同じオーディオ処理ハードウェアが、レガシーマシン９００と更新されたマシン９２０の両方で使用されている（図９Ｂ）。さらに別の例として、ＡＰＵ９３１は、レガシーＡＰＵ９１０と同じハードウェア及びソフトウェアの仕様を有している。

【０３２６】

一実施形態では、ＣＰＵ９２２、もしくはＧＰＵ９２４、もしくはＡＰＵ９３１、またはそれらの２つ以上の組み合わせは、基本ブロックコンパイラ１０４を実行する。

【０３２７】

一実施形態では、ＣＰＵ９２２、もしくはＧＰＵ９２４、もしくはＡＰＵ９３１、またはそれらの２つ以上の組み合わせは、ブロックディスパッチャ３０２を実行する。

【０３２８】

一実施形態では、更新されたオーディオパラメータは、レガシーオーディオパラメータに対応していない。例えば、更新されたオーディオパラメータＵＡＰ１に対応するレガシーオーディオパラメータはない。レガシーオーディオパラメータのいずれかを適用する代わりに、ＡＰＵ９３１は、更新されたオーディオパラメータＵＡＰ１を判定して、レガシーゲームＮのオーディオフレーム２２５２を生成する。

【０３２９】

一実施形態では、レガシーＡＰＵ９０４（図９Ａ）は、エミュレートされたＰＵコード命令１～Ｍのうちの１つ以上を実行する。エミュレートされたＰＵコード命令１～Ｍを実行すると、更新されたオーディオパラメータは生成され得ない。むしろ、レガシーオーディオパラメータが生成される。

【０３３０】

一実施形態では、エミュレーションプロセッサシステム４０９がサーバシステム４０４（図４Ｂ）内に実装されるとき、ディスプレイデバイス４１０またはＨＭＤなどのクライアントデバイスが使用される。エミュレーションプロセッサシステム４０９は、コンピュータネットワーク４０８を介してクライアントデバイスに結合されている。サーバシステム４０４のＮＩＣは、ネットワーク通信プロトコルを更新された画像フレーム２１２３に適用して、１つ以上のデータパケットを生成し、コンピュータネットワーク４０８を介してクライアントデバイスにデータパケットを送信する。クライアントデバイスのＮＩＣは、ネットワーク通信プロトコルをデータパケットに適用して、データパケットから更新された画像フレーム２１２３を抽出し、レガシーゲームＮの仮想環境の表示のために更新された画像フレーム２１２３をクライアントデバイスのディスプレイ画面に提供する。

【０３３１】

図２２Ｂ－２は、ブロックディスパッチャ３０２から出力されたレガシーオーディオフレーム２２６６からの更新されたオーディオフレーム２２５２の生成を説明するためのシステム２２６４の実施形態の図である。システム２２６４は、ブロックディスパッチャ３０２及びＡＰＵエミュレータ２２０４を含む。ブロックディスパッチャ３０２は、ＡＰＵエミュレータ２２０４に結合されている。ブロックディスパッチャ３０２は、基本ブロック１～ｎのうちの１つ以上を実行して、レガシーオーディオパラメータを含むオーディオフレーム２２６６を生成する。ＡＰＵエミュレータ２２０４は、レガシーオーディオフレーム２２６６を受信し、レガシーオーディオフレーム２２６６を解析してレガシーオーディオパラメータを識別し、レガシーオーディオパラメータを更新されたオーディオパラメータに変換して、更新されたオーディオフレーム２２５２を出力する。

【０３３２】

図２２Ｃ－１は、レガシーオーディオパラメータに基づいて更新されたオーディオフレーム２２６０を生成するための方法の実施形態を説明するための図である。ＡＰＵ９３１は、ブロックディスパッチャ３０２（図３）を実行して、レガシーオーディオパラメータを判定し、レガシーオーディオパラメータに基づいてレガシーオーディオフレームを生成する。ＡＰＵエミュレータ２２０４（図２２Ｂ－１）は、レガシーオーディオフレームからレガシーオーディオパラメータを抽出し、レガシーオーディオフレームのレガシーオーディオパラメータを更新されたオーディオパラメータに変換するなど、エミュレートする。ＡＰＵ９３１は、更新されたオーディオパラメータを有する更新されたオーディオフレーム２２６０を生成する。更新されたオーディオフレーム２２６０は、レガシーゲームＮの仮想シーンの画像フレームと共に出力されるものである。例えば、更新されたオーディオフレーム２２６０は、ＡＰＵ９３１から送信され、更新された画像フレーム２１６０（図２１Ｄ）は、サウンドを有するレガシーゲームＮの仮想シーンの画像を表示するために、ＧＰＵ９２４からディスプレイデバイス４１０に送信される。更新されたオーディオフレーム２２６０は、更新されたオーディオフレーム２２５２（図２２Ｂ－１）の１つの一例である。同様の方法で、追加の更新されたオーディオパラメータは、追加のレガシーオーディオパラメータに基づいて判定され、追加の更新されたオーディオパラメータは、更新されたオーディオフレーム２２６０内に統合される。

【０３３３】

図２２Ｃ－２は、エミュレーションプロセッサシステム４０９がゲームコンソール４０２内に実装されているときの、クライアントデバイス２２７２によるレガシーゲームＮのオーディオフレームの処理を説明するためのシステム２２７０の実施形態の図である（図４Ａ）。クライアントデバイス２２７２の一例は、ディスプレイデバイス４１０またはＨＭＤである。クライアントデバイス２２７２は、デジタル－アナログ変換器（ＤＡＣ）２

２７４、フィルタ２２７６、増幅器２２７８、及びスピーカー２２８０を含む。ＤＡＣ２２７４は、増幅器２２７８に結合されているフィルタ２２７６に結合されている。増幅器２２７８は、スピーカー２２８０に結合されている。ＤＡＣ２２７４はまた、オーディオエミュレータ２２０４に結合されている。

【０３３４】

ＤＡＣ２２７４は、オーディオエミュレータ２２０４から更新されたオーディオフレーム２２６０を受信する。更新されたオーディオフレーム２２６０を受信すると、更新されたオーディオフレーム２２６０のデジタルオーディオデータは、ＤＡＣ２２７４によってデジタル形式からアナログ形式に変換されて、アナログオーディオデータを出力する。フィルタ２２７６は、アナログオーディオデータをフィルタリングして、アナログオーディオデータからのノイズを低減または除去するなどしてフィルタリングし、ノイズが低減されたオーディオデータを出力する。増幅器２２７８は、ノイズ低減されたオーディオデータの振幅を変更するなどして増幅し、増幅されたオーディオデータを出力する。スピーカー２２８０は、増幅されたオーディオデータの電気エネルギーをサウンドに変換する。同様の方法で、更新されたオーディオフレーム２２５２（図２２Ｂ－２）は、クライアントデバイス２２７２によって処理される。

【０３３５】

一実施形態では、クライアントデバイス２２７２は、増幅器２２７８に結合された複数のスピーカーを含む。

【０３３６】

一実施形態では、エミュレーションプロセッサシステム４０９がサーバシステム４０４（図４Ｂ）内に実装されるとき、ゲームコンソール４０２とディスプレイデバイス４１０の組み合わせ、もしくはゲームコンソール４０２とＨＭＤの組み合わせなどの別のクライアントデバイス、またはクライアントデバイス２２７２が使用される。この実施形態では、サーバシステム４０４のＮＩＣは、オーディオエミュレータ２２０４から更新されたオーディオフレーム２２６０を受信し、ネットワーク通信プロトコルを更新されたオーディオフレーム２２６０に適用して、１つ以上のデータパケットを生成し、コンピュータネットワーク４０８を介して他のクライアントデバイスにデータパケットを送信する。サーバシステム４０４のＮＩＣは、バス９３０を介してＡＰＵ９３１に結合されている。他のクライアントデバイスのＮＩＣは、データパケットを受信する。一例として、他のクライアントデバイスのＮＩＣは、ＨＭＤまたはディスプレイデバイス４１０またはゲームコンソール４０２内に配置されている。他のクライアントデバイスのＮＩＣは、ネットワーク通信プロトコルをデータパケットに適用して、データパケットから更新されたオーディオフレーム２２６０を取得し、更新されたオーディオフレーム２２６０をＤＡＣ２２７４に送信する。他のクライアントデバイスのＮＩＣはＤＡＣ２２７４に結合されている。

【０３３７】

図２２Ｃ－３は、更新された画像フレーム２１２３に基づいた画像の表示と、レガシーゲームＮの複数の仮想シーンに対して更新されたオーディオフレーム２２５２に基づいたサウンドの出力との同期を説明するシステム２２８０の実施形態の図である。システム２２８０は、ＡＰＵ９３１、ＧＰＵ９２４、及びクライアントデバイス２２７２を含む。クライアントデバイス２２７２は、ビデオオーディオ（ＶＡ）シンクロナイザ２２８２、ＧＰＵ２２８４、及びディスプレイ画面２２８６を含む。ＶＡシンクロナイザ２２８２は、ＤＡＣ２２７４に結合され、かつ、ＧＰＵ２２８４に結合されており、ＧＰＵ２２８４は、ディスプレイ画面２２８６に結合されている。

【０３３８】

エミュレーションプロセッサシステム４０９が、ゲームコンソール４０２内に実装されるとき、ＡＰＵ９３１及びＣＰＵ９２４は、ゲームコンソール４０２の通信デバイスに結合される。通信デバイスは、オーディオフレーム２２５２及び画像フレーム２１２３を受信し、近距離通信プロトコルをフレーム２２５２及び２１２３に適用して、１つ以上のデータパケットを生成し、１つ以上のデータパケットをディスプレイデバイス４１０に送信

する。ディスプレイデバイス４１０は、通信デバイスを含み、これは、１つ以上のデータパケットを受信し、近距離通信プロトコルをデータパケットに適用して、フレーム２２５２及び２１２３を抽出する。ディスプレイデバイス４１０の通信デバイスは、フレーム２２５２及び２１２３をＶＡシンクロナイザ２２８２に送信する。

【０３３９】

ＶＡシンクロナイザ２２８２は、画像フレーム２１２３の表示順序をオーディオフレーム２２５２の表示順序と同期させる。例えば、ＶＡシンクロナイザ２２８２は、更新されたオーディオフレーム２２６０内に埋め込まれた第１の時間識別子、及び更新された画像フレーム２１６０内に埋め込まれた第２の時間識別子に基づいて、更新されたオーディオフレーム２２６０及び更新された画像フレーム２１６０が、同時に出力されていることを判定する。説明として、ＶＡシンクロナイザ２２８２は、第１の時間識別子が、第２の時間識別子と一致することを判定して、更新されたオーディオフレーム２２６０と更新された画像フレーム２１６０が同時に出力されていることを判定する。また、この説明では、第１及び第２の識別子の各々は、英数字コードである。この説明では、第１の時間識別子はＡＰＵ９３１によって生成されており、第２の時間識別子はＧＰＵ９２４によって生成されている。また、この説明では、ＡＰＵ９３１とＧＰＵ９２４の両方が、同じクロック信号に基づいて第１と第２の時間識別子を生成する。さらに説明すると、クロック信号は、ＡＰＵ９３１によって生成されてＧＰＵ９２４に送信されるか、またはＧＰＵ９２４によって生成されてＡＰＵ９３１に送信されるか、またはクロックソースによって生成されてＡＰＵ９３１及びＧＰＵ９２４に送信される。クロックソースの一例はインターネットである。

【０３４０】

ＶＡシンクロナイザ２２８２は、画像フレーム２１２３と２２５２と同期させた後、画像フレーム２１２３をＧＰＵ２２８４に出力し、オーディオフレーム２２５２をＤＡＣ２２７４に出力する。例えば、ＶＡシンクロナイザ２２８２は、更新されたオーディオフレーム２２６０をＤＡＣ２２７４に提供すると同時に、更新された画像フレーム２１６０をＧＰＵ２２８４に提供する。ＧＰＵ２２８４は、ディスプレイ画面２２８６の表示ピクセルを制御して、画像フレーム２１２３に従って、レガシーゲームＮの複数の仮想シーンの画像を表示する。また、オーディオフレーム２２５２は、レガシーゲームＮの複数の仮想シーンの画像の表示と同期してサウンドを出力するために、更新されたオーディオフレーム２２６０が処理されるのと同じ方法で、ＤＡＣ２２７４、フィルタ２２７６、増幅器２２７８、及びスピーカー２２８０（図２２Ｃ－２）によって処理される。

【０３４１】

一実施形態では、エミュレーションプロセッサシステム４０９がサーバシステム４０４（図４Ｂ）内に実装されるとき、オーディオフレーム２２５２は、ＡＰＵ９３１からサーバシステム４０４のＮＩＣに送信される。ＡＰＵ９３１及びＧＰＵ９２４は、サーバシステム４０４のＮＩＣに結合されている。また、画像フレーム２１２３は、ＧＰＵ９２４からサーバシステム４０４のＮＩＣに送信される。サーバシステム４０４のＮＩＣは、ネットワーク通信プロトコルをフレーム２１２３及び２２５２に適用して、１つ以上のパケットを生成し、コンピュータネットワーク４０８（図４Ｂ）を介して図２２Ｃ－２を参照して上で説明した他のクライアントデバイスにパケットを送信する。他のクライアントデバイスのＮＩＣは、ネットワーク通信プロトコルを１つ以上のパケットに適用して、パケットからフレーム２２５２及び２１２３を抽出する。他のクライアントデバイスのＮＩＣは、図２２Ｃ－３を参照して上で説明したのと同じ方法でフレーム２２５２及び２１２３を処理するために、フレーム２２５２及び２１２３をＶＡシンクロナイザ２２８２に送信する。

【０３４２】

図２２Ｄは、レガシーモード及び通常モードを表示するためのディスプレイデバイス４１０の実施形態の図である。ユーザ１は、ハンドヘルドコントローラ４１４を操作して、ユーザ入力を生成する。レガシーゲームＮのプレイ前またはプレイ中にユーザ入力を受信

すると、ＧＰＵ９２４は、画像フレームを生成し、画像フレームをディスプレイデバイス４１０に送信する。ディスプレイデバイス４１０は、画像フレームに基づいて、レガシーモードの選択を受信するためのアイテム２２９２、及び通常モードの選択を受信するためのアイテム２２９４を含む画像２２９０を表示する。ユーザ１が、ハンドヘルドコントローラ４１４を操作してアイテム２２９２を選択すると、ＧＰＵ９２４は、更新されたグラフィックパラメータに基づいて画像フレーム２１２３（図２１Ｂ－１）を生成する。一方で、ユーザ１が、ハンドヘルドコントローラ４１４を操作してアイテム２２９４を選択すると、ＧＰＵ９２４は、レガシーグラフィックパラメータに基づいて画像フレーム４２２（図４Ｂ）を生成する。

【０３４３】

図２３は、メニュー２３０２が表示される仮想環境２３００を表示するためのディスプレイデバイス４１０の実施形態の図である。仮想環境２３００は、仮想環境２３００がメニュー２３０２を含むことを除いて、仮想環境４５２（図４Ｂ）と同じである。メニュー２３０２は、レガシーゲームＮのプレイ中にハンドヘルドコントローラ４１４（図４Ｂ）からユーザ入力が受信されたときに、ディスプレイデバイス４１０のＧＰＵによって表示される。

【０３４４】

メニュー２３０２は、Ｘ位置、Ｙ位置、アップスケール係数、及びアスペクト比などの複数の更新されたグラフィックパラメータを含む。ユーザ１は、ハンドヘルドコントローラ４１４を操作して、メニュー２３０２内に表示される更新されたグラフィックパラメータのうちの１つ以上を選択する。例えば、ユーザ１は、Ｘ位置を選択して、仮想環境４５２のディスプレイデバイス４１０のディスプレイ画面の水平レベルに対する位置を変更する。別の例として、ユーザ１は、Ｙ位置を選択して、ディスプレイデバイス４１０のディスプレイ画面の垂直レベルに対する仮想環境４５２の位置を変更する。ディスプレイ画面の水平レベルは垂直レベルに垂直である。説明すると、水平レベルはｘ軸に沿っており、垂直レベルはｙ軸に沿っている。別の例として、ユーザ１はアップスケール係数を選択して、更新されたアップスケール係数を変更する。さらに別の例として、ユーザ１は、アスペクト比を選択して、ディスプレイデバイス４１０のディスプレイ画面上で仮想環境４５２の高さと仮想環境４５２の幅との比を変更する。高さはディスプレイ画面の垂直レベルに沿って測定され、幅はディスプレイ画面の水平レベルに沿って測定される。

【０３４５】

一実施形態では、メニュー２３０２は、メニュー２３０２内に表示される更新されたグラフィックパラメータの各々についてのスライディングスケールを含む。ユーザ１は、ハンドヘルドコントローラ４１４を使用して、メニュー２３０２の更新されたグラフィックパラメータのうちの１つの対応するスライディングスケールのスライダをスライドさせて、更新されたグラフィックパラメータのレベルまたは量を選択する。

【０３４６】

図２４Ａは、メニュー２４０２が表示される仮想環境２４００を表示するためのディスプレイデバイス４１０の実施形態の図である。仮想環境２４００は、仮想環境２４００がメニュー２４０２を含むことを除いて、仮想環境４５２（図４Ｂ）と同じである。メニュー２４０２は、レガシーゲームＮのプレイ中にハンドヘルドコントローラ４１４（図４Ｂ）からユーザ入力が受信されたときに、ディスプレイデバイス４１０のＧＰＵによって表示される。

【０３４７】

メニュー２４０２は、スキャンラインを有効にする、スキャンライン深度、スキャンライン幅、スキャンライン輝度、ディザ深度、などの複数の更新されたグラフィックパラメータを含む。ユーザ１は、ハンドヘルドコントローラ４１４を操作して、メニュー２４０２内に表示される更新されたグラフィックパラメータのうちの１つ以上を選択する。例えば、ユーザ１は、「スキャンラインを有効にする」を選択して、仮想環境４５２の上にスキャンラインをオーバーレイする。「スキャンラインを有効にする」オプションの選択の

指示を受信すると、ＧＰＵ９２４は、ＧＰＵエミュレータ２１２２を適用してスキャンラインをオーバーレイする。別の例として、ユーザ１は、「スキャンライン深度」を選択して、各「スキャンラインオーバーレイド」の深度を変更するか、または仮想環境４５２上にオーバーレイさせる。スキャンラインの深さは、ディスプレイデバイス４１０のディスプレイ画面の水平レベルに沿っている。説明すると、スキャンラインの深度は、ｘ軸とｙ軸の両方に垂直なｚ軸に沿っている。別の例として、ユーザ１は、スキャンライン幅を選択して、各「スキャンラインオーバーレイド」の幅を変更するか、仮想環境４５２上にオーバーレイさせる。スキャンラインの幅は、ディスプレイデバイス４１０のディスプレイ画面の垂直レベルに沿っている。さらに別の例として、ユーザ１は、「スキャンライン輝度」を選択して、スキャンラインオーバーレイドの輝度の量を変更するか、仮想環境４５２上にオーバーレイさせる。スキャンラインの輝度変更の指示を受信すると、ＧＰＵ９２４は、ＧＰＵエミュレータ２１２２を適用して輝度のレベルを変更する。さらに別の例として、ユーザ１は、ディザ深度を選択し、ノイズを追加して、スキャンライン生成における量子化誤差を低減する。量子化誤差は、量子化をランダム化することによって低減される。ハンドヘルドコントローラ４１４からディザ深度の変更の指示を受信すると、ＧＰＵ９２４は、ＧＰＵエミュレータ２１２２を適用して、スキャンラインを生成するためのディザ深度のレベルを変更する。説明すると、ＧＰＵエミュレータ２１２２は、第２のサンプリングレートよりも小さい第１のサンプリングレートでスキャンラインを生成するためにデータをサンプリングする。第２のサンプリングレートはまた、スキャンラインを生成するためのデータのサンプリングのためのものでもある。第１のサンプリングレートが適用されるとき、サンプリングから出力される第１の量のサンプルポイントは、第２のサンプリングレートから出力される第２の量のサンプリングポイントと比較して低くなっている。説明すると、第１のサンプリングレートは、ナイキストサンプリングレートよりも低くなっている。ＧＰＵエミュレータ２１２２は、ノイズを増加させるために、ディザ深度の一定量を第１の量のサンプリングポイントに適用し、それによって、スキャンラインがより顕著であるようにする。スキャンラインは、ディザ深度が適用されていない場合に比べてより顕著である。

【０３４８】

一実施形態では、メニュー２４０２は、メニュー２４０２内に表示される更新されたグラフィックパラメータの各々についてのスライディングスケールを含む。ユーザ１は、ハンドヘルドコントローラ４１４を使用して、メニュー２４０２の更新されたグラフィックパラメータのうちの１つの対応するスライディングスケールのスライダをスライドさせて、更新されたグラフィックパラメータのレベルまたは量を選択する。

【０３４９】

図２４Ｂは、スキャンライン２４１２Ａ、スキャンライン２４１２Ｂ、及びスキャンライン２４１２Ｃなどの複数のスキャンライン２４１２を有する仮想環境２４１０を表示するためのディスプレイデバイス４１０の実施形態の図である。仮想環境２４１０は、仮想環境２４１０がスキャンライン２４１２を含むことを除いて、仮想環境４５２（図４Ｂ）と同じである。スキャンライン２４１２の各々は、仮想環境４５２の画像の左エッジ２４１４から仮想環境４５２の画像の右エッジ２４１６まで延在している。例えば、各スキャンライン２４１２は水平である。左エッジ２４１４は、仮想環境４５２の画像の上部エッジ２４１８に対して、かつ、仮想環境４５２の画像の下部エッジ２４２０に対して垂直である。

【０３５０】

一実施形態では、スキャンライン２４１２の代わりに、垂直に向けられたスキャンラインが生成され、仮想環境４５２上にオーバーレイされる。例えば、垂直に向けられたスキャンラインは、上部エッジ２４１８から下部エッジ２４２０まで延在している。垂直に向けられたスキャンラインは、ディスプレイデバイス４１０上に表示するためにＧＰＵ９２４によって生成される。

【０３５１】

図２５Ａは、メニュー２５０２が表示される仮想環境２５００を表示するためのディスプレイデバイス４１０の実施形態の図である。仮想環境２５００は、仮想環境２５００がメニュー２５０２を含むことを除いて、仮想環境４５２（図４Ｂ）と同じである。メニュー２５０２は、レガシーゲームＮのプレイ中にハンドヘルドコントローラ４１４（図４Ｂ）からユーザ入力が受信されたときに、ディスプレイデバイス４１０のＧＰＵによって表示される。

【０３５２】

メニュー２５０２は、輝度、コントラスト、ガンマ、色相、彩度などの複数の更新されたグラフィックパラメータを含む。ユーザ１は、ハンドヘルドコントローラ４１４を操作して、メニュー２４０２内に表示される更新されたグラフィックパラメータのうちの１つ以上を選択する。例えば、ユーザ１は、仮想環境４５２の画像ピクセルの輝度を増減するなど、変更する輝度を選択する。「輝度」の選択の指示を受信すると、ＧＰＵ９２４は、ＧＰＵエミュレータ２１２２を適用して輝度を変更する。別の例として、ユーザ１は、仮想環境４５２の仮想背景と比較して、仮想環境４５２の仮想オブジェクトのコントラストを変更するために「コントラスト」を選択する。別の例として、ユーザ１は、ガンマを選択して、仮想環境４５２の画像において色がどれだけ速く変化するかについての指標を提供する。ガンマの選択の指示を受信すると、ＧＰＵ９２４は、ＧＰＵエミュレータ２１２２を適用して、より高いか、またはより低い周波数で色を変更する。さらに別の例として、ユーザ１は、色相を選択して、仮想環境４５２を満たす色を変更する。スキャンライン輝度の変更の指示を受信すると、ＧＰＵ９２４は、仮想環境４５２の画像ピクセルを生成するために使用される色の量にＧＰＵエミュレータ２１２２を適用する。説明すると、ＧＰＵエミュレータ２１２２は、赤、青、緑、及び黄色を含む第１の色相を、赤、青、緑、オレンジ、及び黄色を含む第２の色相に変更する。さらに別の例として、ユーザ１は、彩度を選択して、仮想環境４５２の各画像ピクセルの色の強度または純度の量を変更する。色の彩度が増すと、色はより純粋に見える。彩度が低下すると、色が灰色に置き換わったように見える。

【０３５３】

一実施形態では、メニュー２５０２は、メニュー２５０２内に表示される更新されたグラフィックパラメータの各々についてのスライディングスケールを含む。ユーザ１は、ハンドヘルドコントローラ４１４を使用して、メニュー２５０２の更新されたグラフィックパラメータのうちの１つの対応するスライディングスケールのスライダをスライドさせて、更新されたグラフィックパラメータのレベルまたは量を選択する。

【０３５４】

図２５Ｂは、メニュー２５１２が表示される仮想環境２５１０を表示するためのディスプレイデバイス４１０の実施形態の図である。仮想環境２５１０は、仮想環境２５１０がメニュー２５１２を含むことを除いて、仮想環境４５２（図４Ｂ）と同じである。メニュー２５１２は、レガシーゲームＮのプレイ中にハンドヘルドコントローラ４１４（図４Ｂ）からユーザ入力が受信されたときに、ディスプレイデバイス４１０のＧＰＵによって表示される。

【０３５５】

メニュー２５１２は、セルシェーダによって適用されるシェーディングレベル、セルシェーダによって適用される彩度、セルシェーダによって適用される輝度、セルシェーダによって適用される輪郭、セルシェーダによって適用される強度、及びセルシェーダによって適用されるスムーズ、などの複数の更新されたグラフィックパラメータを含む。セルシェーダは、ＧＰＵエミュレータ２１２２の一例である（図Ｂ）。ユーザ１は、ハンドヘルドコントローラ４１４を操作して、メニュー２５１２内に表示される更新されたグラフィックパラメータのうちの１つ以上を選択する。例えば、ユーザ１は、仮想環境４５２内の仮想オブジェクトの輪郭（周囲または境界など）の画像ピクセルのシェーディングの量を増加または減少など、変更するためにシェーディングレベルを選択する。説明すると、仮想オブジェクトは、ハンドヘルドコントローラ４１４を介してユーザ１によって制御され

る仮想キャラクタである。さらに説明すると、仮想オブジェクトは戦闘機である。シェーディングは、輪郭の色または色の強度を変更する。例として、仮想シーン内の仮想オブジェクトのアウトラインは、仮想オブジェクトを他の仮想オブジェクト及び仮想シーン内の仮想背景から区別する。シェーディングレベルのオプションの選択の指示を受信すると、ＧＰＵ９２４は、ＧＰＵエミュレータ２１２２を適用して、仮想オブジェクトの輪郭のシェーディングレベルを変更する。別の例として、ユーザ１は、彩度を選択して、仮想環境４５２の仮想オブジェクトの輪郭の各画像ピクセルの色の強度または純度の量を変更（増加させる、または減少させるなど）する。彩度のオプションの選択の指示を受信すると、ＧＰＵ９２４は、ＧＰＵエミュレータ２１２２を適用して、仮想オブジェクトの輪郭の各画像ピクセルの色の強度または純度の量を変更する。

【０３５６】

別の例として、ユーザ１は、輝度を選択し、仮想環境４５２内の仮想オブジェクトの輪郭の画像ピクセルの輝度を変更する（増加させる、または減少させるなど）。彩度のオプションの選択の指示を受信すると、ＧＰＵ９２４は、ＧＰＵエミュレータ２１２２を適用して、仮想オブジェクトの輪郭の画像ピクセルの輝度を変更する。

【０３５７】

さらに別の例として、ユーザ１は、アウトラインを選択して、仮想環境で仮想オブジェクトの輪郭を生成するかどうかを示す。輪郭の指示を受信すると、ＧＰＵ９２４は、ＧＰＵエミュレータ２１２２を適用して、仮想オブジェクトの輪郭を含める。輪郭は、仮想オブジェクトの周囲を形成し、仮想オブジェクトの境界を対比して、仮想オブジェクトの仮想背景から、及び仮想環境４５２内の他の仮想オブジェクトから仮想オブジェクトを区別する。さらに別の例として、ユーザ１は、強度を選択して、仮想環境４５２内の仮想オブジェクトの輪郭の透明度量を変更する。輪郭の透明度が低下すると、輪郭は不透明に見え、透明度が増加すると、輪郭は透明に見える。このようにして、輪郭は不透明から半透明に、さらに透明に切り替わるか、またはその逆に切り替わる。輪郭の指示を受信すると、ＧＰＵ９２４は、ＧＰＵエミュレータ２１２２を適用して、仮想環境４５２内の仮想オブジェクトの輪郭の透明度量を変更する。

【０３５８】

別の例として、ユーザ１は、スムーズを選択し、仮想環境４５２内の仮想オブジェクトの輪郭の画像ピクセルの平滑性の量を（増加させる、または減少させるなど）変更する。説明すると、平滑性は、輪郭の画像ピクセルの数を増やすことによって増加し、輪郭の画像ピクセルの数を減らすことによって減少する。スムーズのオプションの選択の指示を受け取ると、ＧＰＵ９２４は、ＧＰＵエミュレータ２１２２を適用して、仮想オブジェクトの輪郭の画像ピクセルの平滑性を変更する。

【０３５９】

一実施形態では、メニュー２５１２は、メニュー２５１２内に表示される更新されたグラフィックパラメータの各々についてのスライディングスケールを含む。ユーザ１は、ハンドヘルドコントローラ４１４を使用して、メニュー２５１２の更新されたグラフィックパラメータのうちの１つの対応するスライディングスケールのスライダをスライドさせて、更新されたグラフィックパラメータのレベルまたは量を選択する。

【０３６０】

図２５Ｃは、メニュー２５２２及びメニュー２５２４が表示される仮想環境２５２０を表示するためのディスプレイデバイス４１０の実施形態の図である。仮想環境２５２０は、仮想環境２５２０がメニュー２５２２及び２５２４を含むことを除いて、仮想環境４５２（図４Ｂ）と同じである。メニュー２５２２及び２５２４は、レガシーゲームＮのプレイ中にハンドヘルドコントローラ４１４（図４Ｂ）からユーザ入力が受信されたときに、ディスプレイデバイス４１０のＧＰＵによって表示される。

【０３６１】

メニュー２５２２は、ビデオフィルタのタイプ１及びビデオフィルタのタイプ２などの複数の更新されたグラフィックパラメータを含む。ユーザ１は、ハンドヘルドコントロー

ラ４１４を操作して、メニュー２５２２内に表示される更新されたグラフィックパラメータのうちの１つ以上を選択する。例えば、ユーザ１は、仮想環境４５２の画像ピクセルに適用されるタイプ１フィルタまたはタイプ２フィルタを選択する。タイプ１フィルタの説明は、Ｍｉｔｃｈｅｌｌ　Ｎｅｔｒａｖａｌｉ（登録商標）フィルタであり、タイプ２フィルタの説明は、Ｂｕｔｔｅｒｗｏｒｔｈ（登録商標）フィルタである。ビデオフィルタのフィルタタイプの他の説明には、Ｃｈｅｂｙｓｈｅｖ（登録商標）及びＢｅｓｓｅｌ（登録商標）が含まれる。ビデオフィルタのタイプ１またはタイプ２の選択の指示を受信すると、ＧＰＵ９２４は、ＧＰＵエミュレータ２１２２を適用して、ビデオフィルタのタイプを適用して、レガシーゲームＮの、仮想オブジェクトの画像ピクセル、または仮想背景の画像ピクセル、またはそれらの組み合わせを生成する。

【０３６２】

同様に、メニュー２５２４は、アンチエイリアシングフィルタのタイプ１、及びアンチエイリアシングフィルタのタイプ２、などの複数の更新されたグラフィカルパラメータを含む。ユーザ１は、ハンドヘルドコントローラ４１４を操作して、メニュー２５２４内に表示される更新されたグラフィックパラメータのうちの１つ以上を選択する。例えば、ユーザ１は、仮想環境４５２の画像ピクセルに適用されるタイプ１フィルタまたはタイプ２フィルタを選択する。タイプ１フィルタの説明は、ＦＸＡＡ（登録商標）フィルタであり、タイプ２フィルタの説明は、Ｌｉｎｅ　Ｔｈｉｃｋｎｅｓｓ（登録商標）フィルタである。アンチエイリアシングフィルタのタイプ１またはタイプ２の選択の指示を受信すると、ＧＰＵ９２４は、ＧＰＵエミュレータ２１２２を適用して、アンチエイリアシングフィルタのタイプを適用して、レガシーゲームＮの、仮想オブジェクトの画像ピクセル、または仮想背景の画像ピクセル、またはそれらの組み合わせを生成する。

【０３６３】

一実施形態では、各メニュー２５２２及び２５２４は、メニュー内に表示される更新されたグラフィックパラメータのそれぞれについてのスライディングスケールを含む。ユーザ１は、ハンドヘルドコントローラ４１４を使用して、メニュー２５２２または２５２４の更新されたグラフィックパラメータのうちの１つの対応するスライドスケールのスライダをスライドさせて、更新されたグラフィックパラメータのレベルまたは量を選択する。

【０３６４】

図２５Ｄは、十字線２５３２が表示されている仮想環境２５３０を表示するためのディスプレイデバイス４１０の実施形態の図である。仮想環境２５３２は、仮想環境２５３２が十字線２５３２を含むことを除いて、仮想環境４５２（図４Ｂ）と同じである。十字線２５３２は、レガシーゲームＮのプレイ中にハンドヘルドコントローラ４１４（図４Ｂ）からユーザ入力が受信されたときに、ディスプレイデバイス４１０のＧＰＵによって表示される。例えば、ユーザ入力は十字線２５３２のサイズを示す。

【０３６５】

ユーザ１は、ハンドヘルドコントローラ４１４を使用して、レガシーゲームＮを一時停止する。レガシーゲームＮが一時停止されると、ユーザ１は、ハンドヘルドコントローラ４１４を操作して、十字線２５３２のサイズが変更されるべきであるという指示を含むユーザ入力を生成する。十字線のサイズは、更新されたグラフィカルパラメータの一例である。指示を受信すると、ＧＰＵ９２４はＧＰＵエミュレータ２１２２（図２１Ｂ－１）を適用して、ユーザ入力を介して受信したサイズと一致するように十字線２５３２のサイズを変更する。

【０３６６】

図２６は、事前（ＡｏＴ）プロセッサシステム２６００の実施形態の図である。ＡｏＴプロセッサシステム２６００は、コンパイルされたコードジェネレータ（ＣＣＧ）２６０１と、ランタイムコンポーネント２６０３と、を含む。ＣＣＧ２６０１は、スキャナ２６０２、デコーダ及びトランスレータ２６０４、ならびにコンパイラ２６０６を含む。ランタイムコンポーネント２６０３には、ファイバクリエータ２６０８、スイッチ２６１０、及びＡｏＴエグゼキュータ２６１２が含まれる。一例として、ＡｏＴコンパイルでは、エ

ミュレートされたＰＵコード１０６の実行可能部分は、実行前にコンパイルされる。このように、実行中にエミュレートされたＰＵコード１０６の実行可能部分をコンパイルする必要はなく、時間が節約される。

【０３６７】

一例として、スキャナ２６０２、デコーダ及びトランスレータ２６０４、及びコンパイラ２６０６の各々は、ホストコンピュータのホストＣＰＵなどのホストプロセッサ内に実装されている。説明すると、スキャナ２６０２、デコーダ及びトランスレータ２６０４、及びコンパイラ２６０６の各々は、ホストプロセッサによって実行されるコンピュータプログラムである。別の例として、スキャナ２６０２、デコーダ及びトランスレータ２６０４、ならびにコンパイラ２６０６の各々は、ＡＳＩＣまたはＰＬＤまたはマイクロコントローラなどのホストハードウェア構成要素である。ホストコンピュータの例には、サーバシステム４０４（図４Ａ）の１つ以上のサーバ、デスクトップコンピュータ、ラップトップコンピュータ、スマートフォン、及びタブレットが含まれる。説明すると、デスクトップコンピュータまたはラップトップコンピュータは、サーバシステム４０４の一部である。ホストコンピュータとホストコンピューティングデバイスという用語は、本明細書では互換的に使用され、ホストプロセッサとホスト処理装置という用語は、本明細書では互換的に使用されることに留意されたい。

【０３６８】

一例として、ファイバクリエータ２６０８、スイッチ２６１０、及びＡｏＴエグゼキュータ２６１２の各々は、ＣＰＵ９２２（図９Ｂ）内に実装されている。説明すると、ファイバクリエータ２６０８、スイッチ２６１０、及びＡｏＴエグゼキュータ２６１２の各々は、ＣＰＵ９２２によって実行されるコンピュータプログラムである。別の例として、ＡｏＴエグゼキュータ２６１２は、ＣＰＵ９２２、ＧＰＵ９２４（図９Ｂ）、またはそれらの組み合わせの中に実装されている。説明すると、ＣＰＵ９２２が、ＧＰＵ９２４に画像フレーム４２２を生成するように命令するときに、ＡｏＴエグゼキュータ２６１２は実装される（図４Ａ）。別の例として、ファイバクリエータ２６０８、スイッチ２６１０、及びＡｏＴエグゼキュータ２６１２の各々は、ゲームコンソール４０２（図４Ａ）内に実装されるＡＳＩＣまたはＰＬＤまたはマイクロコントローラなどのハードウェア構成要素である。

【０３６９】

スキャナ２６０２は、コンパイラ２６０６に結合されているデコーダ及びトランスレータ２６０４に結合されている。コンパイラ２６０６は、ファイバクリエータ２６０８に結合されており、ファイバクリエータ２６０８は、スイッチ２６１０に結合されている。スイッチ２６１０は、ＡｏＴエグゼキュータ２６１２に結合されている。

【０３７０】

スキャナ２６０２は、レガシーゲームＮのすべての実行可能機能に対してエミュレートされたＰＵコード１０６をスキャンして、スキャンされた機能を出力する。例えば、スキャナ２６０２は、メモリデバイス４１２（図６Ｃ）からエミュレートされたＰＵコード１０６にアクセスし、エミュレートされたＰＵコード１０６の各行を解析またはトラバースして、画像フレーム４２２を生成するために実行されることができる実行可能関数などの演算である、エミュレートされたＰＵコード１０６の部分を識別し、この部分を、データであるエミュレートされたＰＵコード１０６の他の部分から区別する。この例では、データ自体は、画像フレーム４２２を生成するために実行されることはできない。別の例として、データは、ユーザ入力４２０（図４Ｂ）をそれ自体で処理できないが、ユーザ入力４１８を処理してレガシーゲームＮのプレイを開始することができる。むしろ、この例では、実行可能関数は、データを適用して、ユーザ入力４１８または４２０を処理する。説明すると、スキャナ２６０２は、エミュレートされたＰＵコード１０６をスキャンして、レガシーゲームＮのすべての可能な仮想オブジェクトの動き（位置及び向きなど）を識別し、レガシーゲームＮのすべての可能な仮想背景の動き（位置及び向きなど）を識別する。この説明では、識別された仮想オブジェクトと仮想背景の動きは、レガシーゲームＮのプ

レイ中にＡｏＴエグゼキュータ２６１２によって受信されるとスキャナ２６０２によって予想されるユーザ入力４２０（図４Ａ）などのユーザ入力に応答している。エミュレートされたＰＵコード１０６の実行可能機能の説明には、書き込み操作、読み取り操作、ジャンプ操作、及びロード操作などの操作が含まれる。

【０３７１】

エミュレートされたＰＵコード１０６のデータの例には、画像データ及びオーディオデータが含まれる。説明すると、画像データは、アニメーションのアイコンまたはルックアンドフィールである。別の例として、画像データは、レガシーゲームＮの仮想オブジェクト及び仮想背景の色及び強度などのグラフィックデータである。別の説明として、オーディオデータは、サウンドが画像データとともに出力されることを可能にする。さらに説明するために、オーディオデータは処理され、レガシーゲームＮの１つ以上の仮想オブジェクトからサウンドを出力するか、またはレガシーゲームＮの１つ以上の仮想背景と共にサウンドを出力する。

【０３７２】

スキャナ２６０２は、実行可能機能などのスキャンされた機能をデコーダ及びトランスレータ２６０４に送信する。例えば、スキャナ２６０２は、データをデコーダ及びトランスレータ２６０４に提供しない。

【０３７３】

デコーダ及びトランスレータ２６０４は、スキャナ２６０２から出力されたスキャンされた機能を復号化し、Ｃ＋＋ソースコード、Ｐｙｔｈｏｎ（登録商標）ソースコード、またはＪａｖａ（登録商標）ソースコードなどの第１のソースコードに変換する。例えば、デコーダ及びトランスレータ２６０４は、スキャンされた機能が、アセンブリ言語プログラムの読み取り及び書き込み操作を含むことを識別するなどして復号化し、読み取り及び書き込み操作をタイトルの読み取り及び書き込みを有する機能に変換する。タイトルの読み取り及び書き込みを有する機能は、本明細書では、読み取り及び書き込み機能と呼ばれることがある。読み取り及び書き込み機能は、読み取り及び書き込み機能を実行するためのデータが格納されており、読み取り及び書き込み操作を実行するための操作をさらに含んでいるホストコンピュータ内のホストメモリデバイスのメモリ場所を識別する。ホストメモリデバイスは、ホストプロセッサに結合されている。スキャンされた機能は、エミュレートされたＰＵコード１０６の操作を識別するために復号化され、次いで第１のソースコードに変換される。

【０３７４】

コンパイラ２６０６は、デコーダ及びトランスレータ２６０４から第１のソースコードを受信する。さらに、コンパイラ２６０６は、ホスト入力デバイスから第２のソースコードを受信する。例えば、プログラマは、ホスト入力デバイスを介してホストプロセッサに第２のソースコードを提供する。ホスト入力デバイスの例には、キーボード、マウス、スタイラス、タッチパッド、またはそれらの組み合わせなどのホスト周辺機器が含まれる。ホスト入力デバイスは、ホストコンピュータの一部であり、ホストコンピュータのホストプロセッサに結合されている。第２のソースコードの例には、画像フレーム４２２の１つを周期的な時間間隔でリフレッシュするためのソースコード、またはオーディオフレームから周期的な時間間隔で生成されるサウンドをリフレッシュするためのソースコード、またはそれらの組み合わせが含まれる。画像フレーム４２２のうちの１つをリフレッシュするための周期的な時間間隔の例は、ディスプレイデバイス４１０またはＨＭＤ上での画像フレーム４２２の表示のフレームレートが、毎秒６０フレームであるときの１６ミリ秒の期間である。コンパイラ２６０６は、第１及び第２のソースコードを受信し、第１のソースコードを第１のコンパイルされたコードにコンパイルし、第２のソースコードを第２のコンパイルされたコードにコンパイルする。第１のコンパイルされたコードまたは第２のコンパイルされたコードの例は、オブジェクトコードである。

【０３７５】

レガシーゲームＮのプレイ期間を含むＡｏＴランタイム中に、ファイバクリエータ２６

０８は、コンパイラ２６０６から第１及び第２のコンパイルされたコードを受信し、第１及び第２のコンパイルされたコードからファイバを作成する。一例として、ファイバは、ＣＰＵ９２２による、第１のコンパイルされたコードの１つ以上の部分と、第２のコンパイルされたコードの１つ以上の部分の実行順序である。ファイバクリエータ２６０８は、ファイバをキャッシュ１０２内に格納する。

【０３７６】

スイッチ２６１０は、ＡｏＴランタイム中に実行される。スイッチ２６１０が実行されると、実行は、第１のコンパイルされたコードの１つ以上の部分から第２のコンパイルされたコードの１つ以上の部分に、またはその逆に変化させる。ＡｏＴエグゼキュータ２６０２が実行されると、第１及び第２のコンパイルされたコードは、ファイバに加えてＣＰＵ９２２によって実行され、レガシーゲームＮの画像フレーム４２２を出力する。一例として、ＣＰＵ９２２は、ＡｏＴエグゼキュータ２６１２を実行して、ＧＰＵ９２４に画像フレーム４２２を生成するように命令する。

【０３７７】

一実施形態では、デコーダ及びトランスレータ２６０４の代わりに、別個のデコーダ及び別個のトランスレータが使用される。例えば、デコーダはトランスレータとは別個のコンピュータプログラムである。別の例として、デコーダは、ＡＳＩＣなどの別個の集積回路、またはトランスレータからの別個のＰＬＤである。

【０３７８】

図２７は、第１及び第２のコンパイルされたコードを含む、コンパイルされたコードセット２７０２の生成を説明するためのシステム２７００の実施形態の図である。システム２７００は、ホストプロセッサ２７０４及びメモリデバイス２７０６を含む。一例として、ホストプロセッサ２７０４は、サーバシステム４０４（図４Ｂ）に結合されている。説明すると、ホストコンピュータは、コンピュータネットワーク４０８（図４Ｂ）を介してサーバシステム４０４に結合されている。ホストプロセッサ２７０４は、ホストコンピュータのプロセッサである。メモリデバイス２７０６の例は、ＣＤ－ＲＯＭドライブ９０８によって読み取り可能であるＣＤ－ＲＯＭを含む。メモリデバイス２７０６は、レガシーゲームＮのエミュレートされたＰＵコード１０６を格納する。

【０３７９】

ホストプロセッサ２７０４は、メモリデバイス２７０６に結合されている。スキャナ２６０２（図２６）は、ホストプロセッサ２７０４によって実行され、メモリデバイス２７０６からエミュレートされたＰＵコード１０６をスキャンする。例えば、エミュレートされたＰＵコード１０６は、メモリデバイス２７０６からホストプロセッサ２７０４によって読み取られ、エミュレートされたＰＵコード１０６の実行可能機能をエミュレートされたＰＵコード１０６のデータから区別して、スキャンされた機能を出力する。ホストプロセッサ２７０４は、デコーダ及びトランスレータ２６０４（図２６）を実行して、スキャンされた機能を復号化し、第１のソースコードに変換する。第１のソースコードは、デコーダ及びトランスレータ２６０４からコンパイラ２６０６（図２６）に提供される。

【０３８０】

また、軽量エミュレータプログラムなどの第２のソースコードは、ホスト入力デバイスからコンパイラ２６０６によって受信される。ホストプロセッサ２７０４は、コンパイラ２６０６を実行して、第１及び第２のソースコードをコンパイルされたコードセット２７０２にコンパイルする。一例として、コンパイルされたコードセット２７０２は、複数のネイティブライブラリを含み、各ネイティブライブラリは、コンパイルされたコードセット２７０２の部分を格納するための１つ以上のファイルを含む。この例では、コンパイルされたコードセット２７０２は非実行可能プログラムである。説明すると、コンパイルされたコードセット２７０２は、自動的に動作または実行しない。この説明では、コンパイルされたコードセット２７０２は、ユーザ入力４１８または４２０などのユーザ入力が受信されたときに実行する。

【０３８１】

一実施形態では、ネイティブライブラリは、第２のコンパイルされたコードを含まない。

【０３８２】

図２８は、コンピュータスレッドなどのファイバ２８０２を説明するためのシステム２８００の実施形態の図である。ファイバ２８０２の一例は、第１及び第２のコンパイルされたコードのシーケンスまたは一連の実行、ならびに第１及び第２のコンパイルされたコードの実行の間に実行される操作である。ファイバ２８０２は、システム２８００において破線で示されている。システム２８００は、コンパイルされたコード（ＣＣ）１及びコンパイルされたコード２を含む。一例として、コンパイルされたコード１は、レガシーゲームＮの第１のソースコードからコンパイルされた第１のコンパイルされたコードであり、コンパイルされたコード２は、第２のソースコードからコンパイルされた第２のコンパイルされたコードである。

【０３８３】

コンパイルされたコード１は、開始アドレスと終了アドレスを有する。コンパイルされたコード１の開始アドレスは、コンパイルされたコード１の開始の指示が格納されるキャッシュ１０２（図１）内の場所である。例えば、コンパイルされたコード１の開始アドレスは、コンパイルされたコード１の名前または英数字の識別子などの開始識別子が格納されるアドレスである。コンパイルされたコード１の終了アドレスは、コンパイルされたコード１の終了の指示が格納されるキャッシュ１０２内の場所である。例えば、コンパイルされたコード１の終了アドレスは、コンパイルされたコード１の名前または英数字の識別子などの終了識別子が格納されるアドレスである。

【０３８４】

同様に、コンパイルされたコード２は、開始アドレスと終了アドレスを有する。コンパイルされたコード２の開始アドレスは、コンパイルされたコード２の開始の指示が格納されるキャッシュ１０２内の場所である。例えば、コンパイルされたコード２の開始アドレスは、コンパイルされたコード２の名前または英数字の識別子などの開始識別子が格納されるアドレスである。コンパイルされたコード２の終了アドレスは、コンパイルされたコード２の終了の指示が格納されるキャッシュ１０２内の場所である。例えば、コンパイルされたコード２の終了アドレスは、コンパイルされたコード２の名前または英数字の識別子などの終了識別子が格納されるアドレスである。

【０３８５】

コンパイルされたコード１は、レガシーゲームＮのコンパイルされたコード部分１（ＣＣＰ１）、ＣＣＰ２、ＣＣＰ３などの複数の部分を含む。コンパイルされたコード部分ＣＣＰ１の一例は、レガシーゲームＮのアニメーションであり、アニメーションは、レガシーゲームＮをユーザ１に紹介する。説明すると、アニメーションは、ユーザ入力４１８（図４Ａ）が、ハンドヘルドコントローラ４１４（図４Ａ）から受信されたときに、レガシーゲームＮのタイトルＧＮ、１つ以上の画像、及びレガシーゲームＮの紹介のオーディオを提供する。コンパイルされたコード部分ＣＣＰ２の例は、レガシーゲームＮのプレイ中のユーザ入力４２０（図４Ａ）に応答したレガシーゲームＮの仮想オブジェクト４５４（図４Ｂ）の動きを含む。コンパイルされたコード部分ＣＣＰ３の例は、仮想オブジェクト４５４の上向きの動きまたは下向きの動きなどの動きを含む。この例では、仮想オブジェクト４５４は、ユーザ入力４２０を受信することなく移動する。説明すると、仮想オブジェクト４５４は、レガシーゲームＮ内で自動的に移動し、この移動は、ユーザ入力４２０を必要としない。

【０３８６】

コンパイルされたコード２の例には、レガシーゲームＮのイメージの周期的なリフレッシュが含まれている。説明すると、ディスプレイデバイス４１０（図４Ａ）またはＨＭＤに表示された画像または画像フレームは、ディスプレイデバイス４１０またはＨＭＤが画像または画像フレームを保持する能力を有さないためにリフレッシュされる。この説明では、ＧＰＵ９２４（図９Ｂ）は、ディスプレイデバイス４１０またはＨＭＤのＬＣＤ素子

またはＬＥＤなどの光源を制御して、ディスプレイデバイス４１０のディスプレイ画面またはＨＭＤのディスプレイ画面全体にわたり光源を定期的に掃引し、画像または画像フレームを再描画する。この説明では、ＧＰＵ９２４は、ディスプレイデバイス４１０またはＨＭＤの１つ以上のトランジスタなどの複数のディスプレイドライバに制御信号を提供することによって光源を制御する。この説明では、各ディスプレイドライバは、対応する光源に結合されて、対応する光源を制御する。また、この説明では、各ディスプレイドライバがＧＰＵ９２４に結合されている。さらにこの説明では、制御信号の受信に応答して、ディスプレイドライバは、対応する光源をオンまたはオフにして、画像を再描画する。またこの説明では、ディスプレイドライバは、ディスプレイデバイス４１０またはＨＭＤのディスプレイコントローラの一部である。別の説明として、コンパイルされたコード２は、周期的リフレッシュ画面を含み、これは、毎１６ミリ秒（ミリ秒）など、レガシーゲームＮの画像を周期的にリフレッシュする。この説明では、画像は、コンパイルされたコード１のコンパイルされたコード部分の実行に基づいて生成される。さらに、この説明では、コンパイルされたコード１のコンパイルされたコード部分は、ＡｏＴエグゼキュータ２６１２によるコンパイルされたコード２の実行以前に、ＡｏＴエグゼキュータ２６１２（図２６）によって実行される。さらに説明すると、コンパイルされたコード２は、コンパイルされたコード部分ＣＣＰ２がＡｏＴエグゼキュータ２６１２によって実行されるときに生成される画像を定期的にリフレッシュする。

【０３８７】

コンパイルされたコード２の別の例は、レガシーゲームＮのオーディオフレームから生成されたサウンドの定期的なリフレッシュを含む。説明すると、ディスプレイデバイス４１０のスピーカー（図４Ａ）またはＨＭＤのスピーカーを介したサウンド出力は、ディスプレイデバイス４１０のスピーカーまたはＨＭＤのスピーカーが、サウンドを表す電気エネルギーを保持する容量を有さないためにリフレッシュされる。この説明では、ＣＰＵ９２２（図９Ｂ）は、ディスプレイデバイス４１０のスピーカーまたはＨＭＤのスピーカーを制御して、スピーカーによってサウンドに変換される電気エネルギーを周期的にリフレッシュして、サウンドをリフレッシュする。この説明では、ＣＰＵ９２４は、ディスプレイデバイス４１０またはＨＭＤのＤＡＣ２２７４、フィルタ２２７６、及び増幅器２２７８（図２２Ｃ－２）の組み合わせなどのオーディオドライバに制御信号を提供することによって、ディスプレイデバイス４１０またはＨＭＤのスピーカーを制御する。この説明では、オーディオドライバがＣＰＵ９２２に結合されている。さらにこの説明では、制御信号の受信に応答して、オーディオドライバはサウンドを再出力する。また、この説明では、オーディオドライバは、ディスプレイデバイス４１０またはＨＭＤのオーディオコントローラの一部である。別の例として、コンパイルされたコード２は、サウンドの周期的なリフレッシュを含み、これは、レガシーゲームＮの画像と共に出力されるサウンドを周期的にリフレッシュする。この説明では、コンパイルされたコード２は、コンパイルされたコード１のコンパイルされたコード部分の実行に基づいて生成されたレガシーゲームＮのサウンドを周期的にリフレッシュする。さらに、この説明では、コンパイルされたコード１のコンパイルされたコード部分は、ＡｏＴエグゼキュータ２６１２によるコンパイルされたコード２の実行以前に、ＡｏＴエグゼキュータ２６１２（図２６）によって実行される。さらに説明すると、コンパイルされたコード２は、コンパイルされたコード部分ＣＣＰ２がＡｏＴエグゼキュータ２６１２によって実行されるときに生成されるサウンドを定期的にリフレッシュする。別の説明として、コンパイルされたコード２は、周期的リフレッシュサウンドを含み、これはレガシーゲームＮのサウンドを周期的にリフレッシュする。この説明では、コンパイルされたコード２は、コンパイルされたコード１のコンパイルされたコード部分の実行に基づいて生成されたレガシーゲームＮのサウンドを周期的にリフレッシュする。さらに、この説明では、コンパイルされたコード１のコンパイルされたコード部分は、ＡｏＴエグゼキュータ２６１２によるコンパイルされたコード２の実行以前に、ＡｏＴエグゼキュータ２６１２（図２６）によって実行される。さらに説明すると、コンパイルされたコード２は、コンパイルされたコード部分ＣＣＰ２がＡｏＴエグゼキ

ュータ２６１２によって実行されるときに生成されるサウンドを周期的にリフレッシュする。

【０３８８】

ファイバ２８０２は、ＡｏＴエグゼキュータ２６１２によって実行されると、コンパイルされたコード１の開始アドレスで始まるコンパイルされたコード部分ＣＣＰ１の実行を指示する。例えば、ＡｏＴエグゼキュータ２６１２は、ユーザ入力４１８が、ハンドヘルドコントローラ４１４（図４Ａ）から受信されるかどうかを判定する。この例では、ユーザ入力４１８が受信されたと判定すると、ＡｏＴエグゼキュータ２６１２は、コンパイルされたコード１のコンパイルされたコード部分ＣＣＰ１を実行する。説明すると、コンパイルされたコード部分ＣＣＰ１が実行されると、ゲームタイトルＧＮを有するレガシーゲームＮのアニメーションを有する１つ以上の画像フレームが、ディスプレイデバイス４１０（図４Ａ）またはＨＭＤに表示される。

【０３８９】

コンパイルされたコード部分ＣＣＰ１が実行されると、ＡｏＴエグゼキュータ２６１２は、ユーザ入力４２０（図４Ａ）が入力プロセッサシステム４０７（図４Ａ）から受信されるかどうかを判定する。ユーザ入力４２０が受信されたと判定すると、ＡｏＴエグゼキュータ２６１２は、ファイバ２８０２を実行して、コンパイルされたコード部分ＣＣＰ２に実行を指示する。説明すると、コンパイルされたコード部分ＣＣＰ２は、ユーザ入力４２０に応答して、戦闘機を上方向、下方向、または左方向、または右方向などの方向に移動させるために実行される。

【０３９０】

コンパイルされたコード部分ＣＣＰ２が実行された後、ＡｏＴエグゼキュータ２６１２は、ファイバ２８０２を実行して、コンパイルされたコード部分ＣＣＰ２の実行の終わりにキャッシュ１０２のレジスタの状態を格納する操作２８０４を実行する。例えば、コンパイルされたコード部分ＣＣＰ２の実行が終了するまでにユーザ１によって獲得されたポイントの数は、キャッシュ１０２の第１のレジスタから第１のレジスタの外側のメモリ場所にコピーされる。第１のレジスタの外側のメモリ場所の例には、キャッシュ９２８内の別の場所またはメモリシステム９２６（図９Ｂ）内の場所が含まれる。別の例として、ファイバ２８０２がコンパイルされたコード１の実行から離れるように指示するコンパイルされたコード部分ＣＣＰ２の終了サブアドレスの識別は、キャッシュ１０２の第２のレジスタから第２のレジスタの外側のメモリ場所にコピーされる。第２のレジスタの外側のメモリ場所の例には、キャッシュ９２８内の別の場所またはメモリシステム９２６内の場所が含まれる。

【０３９１】

操作２８０４の後のファイバ２８０２の継続的な実行中、操作２８０６において、スイッチ２６１０（図２６）は、操作２８０４の実行をコンパイルされたコード２の実行に切り替える。一例として、操作２８０６の間、ファイバ２８０２が実行されると、スイッチ２６１０は、操作２８０６からコンパイルされたコード２の実行に実行を指示する。別の例として、第１及び第２のレジスタの状態が、第１及び第２のレジスタの外側のメモリ場所にコピーされた後、ファイバ２８０２は、操作２８０４からコンパイルされたコード２の開始アドレスに切り替え、コンパイルされたコード２が実行される。また、この例では、第１レジスタと第２レジスタは、コンパイルされたコード２を実行して生成されたデータによって上書きされる。

【０３９２】

コンパイルされたコード２は、コンパイルされたコード２の終了アドレスに到達するまで実行される。コンパイルされたコード２が実行された後、ファイバ２８０２の実行は、操作２８０８を続行する。操作２８０８の間、コンパイルされたコード２を実行して生成されたデータで上書きされるキャッシュ１０２のレジスタは、レジスタの状態で上書きされる。そのため、レジスタは状態とともに復元される。例えば、コンパイルされたコード部分ＣＣＰ２の実行が終了するまでにユーザ１が獲得したポイント数は、第１のレジスタ

の外側のメモリ場所から第１のレジスタにコピーされる。また、この例では、ファイバ２８０２がコンパイルされたコード１を離れるように指示するコンパイルされたコード部分ＣＣＰ２の終了サブアドレスの識別が、第２のレジスタの外側のメモリ場所から第２のレジスタにコピーされる。

【０３９３】

操作２８０８後のファイバ２８０２の継続実行中、操作２８１０において、スイッチ２６１０は、操作２８０８の実行からコンパイルされたコード部分ＣＣＰ３の実行に切り替える。例えば、第１及び第２のレジスタの状態が、第１及び第２のレジスタの外側のメモリ場所から第１及び第２のレジスタにコピーバックされた後、ファイバ２８０２は実行の指示をＣＣＰ２の終了サブアドレスに切り替える。ファイバ２８０２は、実行が継続されると、コンパイルされたコード部分ＣＣＰ３の開始サブアドレスにおいて開始するコンパイルされたコード部分ＣＣＰ３を実行するように指示する。

【０３９４】

このようにして、ファイバクリエータ２６０８（図２６）は、ファイバ２８０２を作成する。例えば、ファイバクリエータ２６０８は、コンパイルされたコード部分ＣＣＰ１を最初に、コンパイルされたコード部分ＣＣＰ２を２番目に、操作２８０４を３番目に、操作２８０６を４番目に、コンパイルされたコード２を５番目に、操作２８０８を６番目に、操作２８１０を７番目に、そしてコンパイルされたコード部分ＣＣＰ３を８番目に実行する順序またはシーケンスまたは方向を作成する。説明すると、ファイバ２８０２は、操作２８０４、２８０６、２８０８、及び２８１０を含む。別の例として、ファイバ２８０２が実行されるとき、コンパイルされたコード１及び２は、同時になど、並行して実行されない。むしろ、コンパイルされたコード１と２は、順次に、または連続的に実行される。一実施形態では、コンパイルされたコード１及び２のうちの１つ以上は、キャッシュ９２８（図９Ｂ）に格納される代わりに、メモリシステム９２６（図９Ｂ）に格納される。

【０３９５】

一実施形態では、レガシーゲームＮの画像または画像フレームを伴うサウンド出力のリフレッシュは、画像または画像フレームのリフレッシュが実行される前または後に実行される。例えば、コンパイルされたコード３は、ファイバ２８０２によって指示されるように、コンパイルされたコード２が実行される前または直後に、ＡｏＴエグゼキュータ２６１２によって実行される。この例では、コンパイルされたコード２を実行して画像を周期的に更新するとき、コンパイルされたコード３が実行されて画像を伴うサウンド出力が周期的に更新される。別の例として、コンパイルされたコード３は、レガシーゲームＮの画像または画像フレームを伴ったサウンド出力を周期的にリフレッシュするための機能を含む。この例では、コンパイルされたコード部分ＣＣＰ３が実行された後、キャッシュ１０２のレジスタの状態をキャッシュ１０２のレジスタの外側のメモリ場所に格納する別の操作が、ＡｏＴエグゼキュータ２６１２によって実行され、続いて、ファイバ２８０２の指示に従って、コンパイルされたコード３に切り替える操作が行われる。コンパイルされたコード３に切り替える操作は、スイッチ２６１０によって実行され、スイッチ２６１０は、ファイバ２８０２に、コンパイルされたコード３への実行に切り替えるように指示する。次に、コンパイルされたコード３は、ファイバ２８０２の指示に従って、ＡｏＴエグゼキュータ２６１２によって実行される。さらに、この例では、コンパイルされたコード３の実行後、キャッシュ１０２のレジスタ内に格納された状態は、コンパイルされたコード３の実行に基づいて生成されたデータで更新される。この例では、更新後、ＡｏＴエグゼキュータ２６１２によってデータがレジスタの状態で上書きされると、レジスタの状態が復元される。また、この例では、データが上書きされた後、スイッチ２６１０は、ファイバ２８０２によって提供される指示に従って、コンパイルされたコード１のコンパイルされたコード部分４（ＣＣＰ４）に実行を切り替える。この例では、ファイバ２８１２は、レジスタの状態を格納する他の操作、コンパイルされたコード３への切り替えの操作、コンパイルされたコード３の実行、レジスタ内のデータをレジスタの状態で上書きする操作、及びＣＣＰ４の実行の実行順序を含む。

【０３９６】
一実施形態では、ファイバ２８０２は、コンパイルされたコード部分ＣＣＰ１、コンパイルされたコード部分ＣＣＰ２、操作２８０４、操作２８０６、コンパイルされたコード２、操作２８０８、操作２８１０、及びコンパイルされたコード部分ＣＣＰ３を含む。
【０３９７】
一実施形態では、ファイバ２８０２は、操作２８０４、操作２８０６、コンパイルされたコード２、操作２８０８、及び操作２８１０を含む。
【０３９８】
一実施形態では、コンパイルされたコード部分ＣＣＰ１の実行の終了時にキャッシュ１０２のレジスタの１つ以上の状態を格納する操作２８０４に類似したレジスタ状態の格納操作は、コンパイルされたコード部分ＣＣＰ１のランニング後にＡｏＴエグゼキュータ２６１２によって実行される。また、レジスタ状態の格納操作の後、操作２８０６が繰り返され、コンパイルされたコード２の実行に切り替わる。例えば、ユーザ入力なしでタイトルＧＮを有するレガシーゲームＮのアニメーションを提供する画像フレームは、周期的に更新されるか、またはアニメーションと同時のオーディオフレームに基づいたサウンド出力は、周期的に更新されるか、またはそれらの組み合わせになる。コンパイルされたコード２が実行されると、１つ以上の状態がデータで更新される。さらに、コンパイルされたコード２が実行された後、１つ以上の状態でデータを置き換える操作２８０８などのレジスタ状態の復元操作が、ＡｏＴエグゼキュータ２６１２によって実行される。さらに、レジスタ状態の復元操作後、操作２８１０が繰り返され、コンパイルされたコード部分ＣＣＰ２の実行に切り替わる。この実施形態では、ファイバ２８０２は、コンパイルされたコード部分ＣＣＰ１、レジスタ状態の格納操作、操作２８０６、コンパイルされたコード２、レジスタ状態の復元操作、操作２８１０、及びコンパイルされたコード部分ＣＣＰ２の実行順序を含む。
【０３９９】
一実施形態では、ファイバ２８０２は、コンパイルされたコード部分ＣＣＰ１、レジスタ状態の格納操作、操作２８０６、コンパイルされたコード２、復元レジスタ状態の格納操作、操作２８１０、及びコンパイルされたコード部分ＣＣＰ２を含む。
【０４００】
一実施形態では、ファイバ２８０２は、レジスタ状態の格納操作、操作２８０６、コンパイルされたコード２、及び復元レジスタ状態の格納操作、操作２８１０を含む。
【０４０１】
一実施形態では、コンパイルされたコード部分ＣＣＰ３の実行の終了時にキャッシュ１０２のレジスタの１つ以上の状態を格納する操作２８０４に類似したレジスタ状態の格納操作は、コンパイルされたコード部分ＣＣＰ３のランニング後にＡｏＴエグゼキュータ２６１２によって実行される。また、レジスタ状態の格納操作の後、操作２８０６が繰り返され、コンパイルされたコード２の実行に切り替わる。例えば、戦闘機が、いずれのユーザ入力も伴わずに動く画像フレームが、周期的に更新されるか、または戦闘機の動きと同時に音声フレームに基づいたサウンド出力が、周期的に更新されるか、またはそれらの組み合わせが行われる。コンパイルされたコード２が実行されると、１つ以上の状態がデータで更新される。さらに、コンパイルされたコード２が実行された後、１つ以上の状態でデータを置き換える操作２８０８などのレジスタ状態の復元操作が、ＡｏＴエグゼキュータ２６１２によって実行される。さらに、レジスタ状態の復元操作の後、操作２８１０が繰り返され、コンパイルされたコード部分の実行に切り替わる。後続のコンパイルされたコード部分は、コンパイルされたコード部分ＣＣＰ３の直後に続く。コンパイルされたコード部分ＣＣＰ３と後続のコンパイルされたコード部分との間には、他のコンパイルされたコード部分はない。この実施形態では、ファイバ２８０２は、コンパイルされたコード部分ＣＣＰ３、レジスタ状態の格納操作、操作２８０６、コンパイルされたコード２、レジスタ状態の復元操作、操作２８１０、及び後続のコンパイルされたコード部分の実行順序を含む。

【０４０２】
一実施形態では、ファイバ２８０２は、コンパイルされたコード部分ＣＣＰ３、レジスタ状態の格納操作、操作２８０６、コンパイルされたコード２、レジスタ状態の復元操作、操作２８１０、及び後続のコンパイルされたコード部分を含む。
【０４０３】
一実施形態では、ファイバ２８０２は、レジスタ状態の格納操作、操作２８０６、コンパイルされたコード２、レジスタ状態の復元操作、及び操作２８１０を含む。
【０４０４】
図２９Ａは、レガシーゲームＮの実行可能機能のいくつかを説明するための、レガシーゲームＮのアセンブリ言語プログラムなどの実行可能コード２９００の実施形態の図である。実行可能コード２９００は、「．ｃｏｄｅ」という用語で始まり、メモリデバイス２７０６（図２７）の読み取り中に、実行可能コード２９００が開始されようとしていることをホストプロセッサ２７０４に示す。実行可能コード２９００は、読み取りオペコード、書き込みオペコード、リセットオペコード、ジャンプオペコード、追加オペコード、及びループオペコードを含む。一例として、読み取りオペコードは、レガシーマシン９００（図９Ａ）のキャッシュのレジスタ、またはメモリデバイス９０６（図９Ａ）内のメモリ場所、またはＣＤ－ＲＯＭドライブ９０８（図９Ａ）に結合されたＣＤ－ＲＯＭからデータまたは値などの情報を読み取るために実行される。別の例として、書き込みオペコードは、レガシーマシン９００のキャッシュのレジスタ、またはメモリデバイス９０６内のメモリ場所、またはＣＤ－ＲＯＭドライブ９０８に結合されたＣＤ－ＲＯＭに情報を書き込むために実行される。さらに別の例として、ジャンプオペコードは、レガシーマシン９００のキャッシュのレジスタ、またはメモリデバイス９０６内のメモリ場所、またはＣＤ－ＲＯＭドライブ９０８に結合されたＣＤ－ＲＯＭにジャンプするために実行される。別の例として、追加オペコードは、レガシーマシン９００のキャッシュの第１のレジスタ、またはメモリデバイス９０６内の第１のメモリ場所、またはＣＤ－ＲＯＭドライブ９０８に結合されたＣＤ－ＲＯＭ内の第１の場所の中に格納された値を、レガシーマシン９００のキャッシュの第２のレジスタ、またはメモリデバイス９０６内の第２のメモリ場所、またはＣＤ－ＲＯＭ内の第２の場所の中に格納された値に追加するために実行される。別の例として、ループオペコードは、条件が満たされるまで操作を複数回繰り返すために実行される。説明すると、ループオペコードは、レガシーマシン９００のキャッシュのレジスタｒ１内に格納された値を、レガシーマシン９００のキャッシュのレジスタｒ０内に格納された値と乗算するために実行される。各オペコードは、レガシーゲームＮの実行可能機能の例である。
【０４０５】
一実施形態では、ホストプロセッサ２７０２は、「．ｃｏｄｅ」などの開始用語がオペコードに先行することを判定する必要なしに、本明細書で説明されるオペコードのいずれかを実行可能機能として識別する。オペコードをデータから区別するためホストプロセッサ２７０２がオペコードを識別するために、オペコードの前に用語を付ける必要はない。
【０４０６】
図２９Ｂは、データ２９１０の実施形態の図であり、これは、ホストプロセッサ２７０４（図２７）によって実行可能ではないと識別される。データ２９１０は、レガシーゲームＮのものである。データ２９１０は、「．ｄａｔａ」という用語で始まり、メモリデバイス２７０６（図２７）の読み取り中に、データ２９１０が開始されようとしていることをホストプロセッサ２７０４に示す。一例として、データ２９１０は、レジスタｒ１内に格納された４の値と、レジスタｒ１内に格納された６の値とを含む。メモリデバイス２７０６内に格納されたレガシーゲームＮのデータの他の例には、サウンドデータ及びゲームアイコンなどの画像データが含まれる。
【０４０７】
図３０Ａは、ソースコード３０００の実施形態を示している。ソースコード３０００には、ｍａｉｎ（）という名称が含まれている。名称には、開いた中括弧開くがさらに続き

、これには、さらに、整数を格納するためのキャッシュ１０２のレジスタａの割り当て、及び整数を格納するためのキャッシュ１０２のレジスタｂの割り当てが続く。割り当ての後には、レジスタａからの値の読み取り、レジスタｂへの値１の書き込みが続く。ソースコードは中括弧閉じるで終わる。ソースコード３０００は、レガシーゲームＮの第１のソースコードの例である。

【０４０８】

図３０Ｂは、ソースコード３０１０及び別のソースコード３０１２の実施形態を示している。ソースコード３０１０は、第１のソースコードの例であり、ソースコード３０１２は第２のソースコードの例である。ソースコード３０１０は、コンパイルされたコード１を生成するためにコンパイルされている。ソースコード３０１２は、コンパイルされたコード２、またはコンパイルされたコード３、またはそれらの組み合わせを生成するためにコンパイルされている。

【０４０９】

図３１は、ブロッククリエータ６０４とファイバクリエータ２６０８との間の結合を説明するためのシステム３１００の実施形態の図である。ブロッククリエータ６０４は、ファイバクリエータ２６０８に結合されている。ブロッククリエータ６０４及びファイバクリエータ２６０８によって実行される操作は、図３２Ａ～３２Ｃを参照して以下に説明されている。

【０４１０】

図３２Ａは、図３１のブロッククリエータ６０４及びファイバクリエータ２６０８によって実行される方法３２００の実施形態のフローチャートである。方法３２００では、操作３２０２は、ファイバクリエータ２６０８によって実行される。操作３２０２において、ファイバクリエータ２６０８は、ファイバ２８０２（図２８）がセットアップされているかどうかを判定する。例えば、ユーザ入力４１８の受信に応答して、ファイバクリエータ２６０８は、ユーザ入力４１８を処理するためのファイバ２８０２が作成され、キャッシュ１０２に格納されているかどうかを判定する。説明すると、ファイバクリエータ２６０８は、ユーザ入力４１８を解析して、ユーザ入力４１８が、レガシーゲームＮのプレイを開始するための要求を含むかを判定し、コンパイルされたコード部分ＣＣＰ１が、レガシーゲームＮのアニメーションを含むかを判定して、ユーザ入力４１８を処理するためのファイバ２８０２が作成されるかを判定する。別の例として、ユーザ入力４２０を受信することに応答して、ファイバクリエータ２６０８は、ユーザ入力４２０を処理するためのファイバ２８０２が作成され、キャッシュ１０２内に格納されているかどうかを判定する。説明すると、ファイバクリエータ２６０８は、ユーザ入力４２０を解析して、ユーザ入力４２０が、仮想戦闘機をある方向に動かす要求を含むことを判定し、コンパイルされたコード部分ＣＣＰ２が、仮想戦闘機のある方向での動きを表す１つ以上の画像フレーム４２２を生成する操作を含むことを判定し、コンパイルされたコード部分ＣＣＰ２を実行した後に、操作２８０４及び２８０６、コンパイルされたコード２、ならびに操作２８０８及び２８１０が実行され、ユーザ入力４２０を処理するためのファイバ２８０２が作成されていることを判定する。

【０４１１】

ファイバ２８０２がセットアップされていることを判定すると、次のコンパイルされたコードに切り替える操作３２０４が、スイッチ２６１０（図２６）によって実行される。操作３２０４において、スイッチ２６１０は、実行を次のコンパイルされたコードに切り替える。例えば、スイッチ２６１０は、ユーザ入力４１８を受信すると、実行をコンパイルされたコード１の開始アドレスに切り替える。別の例として、スイッチ２６１０は、ユーザ入力４２０を受信した後、実行をコンパイルされたコード２の開始アドレスに切り替える。この例では、コンパイルされたコード部分ＣＣＰ１及びＣＣＰ２と操作２８０４が実行された後、実行はコンパイルされたコード２に切り替えられる（図２８）。

【０４１２】

方法３２００の操作３２０６は、ＡｏＴエグゼキュータ２６１２（図２６）によって実

行されている。例えば、ＡｏＴエグゼキュータ２６１２は、次にコンパイルされたコードを実行する。説明すると、ＡｏＴエグゼキュータ２６１２は、ユーザ入力４１８が受信されたときに、コンパイルされたコード１の開始アドレスでコンパイルされたコード部分ＣＣＰ１を実行する。別の説明として、ＡｏＴエグゼキュータ２６１２は、コンパイルされたコード部分ＣＣＰ２が実行され、操作２８０４が実行された後、コンパイルされたコード２の開始アドレスでコンパイルされたコード２を実行する。操作３２０２、３２０４、及び３２０６は、操作３２０６の後にこのように繰り返される。

【０４１３】

一方で、キャッシュ１０２に格納されていないなど、ファイバ２８０２がセットアップされていないと判定すると、ファイバクリエータ２６０８は、方法３２００の操作３２０８において、次のコンパイルされたコードが、ネイティブライブラリから特定されることができると判定する。例えば、ファイバクリエータ２６０８は、ネイティブライブラリを解析または検索して、ユーザ入力４１８が受信されたときにユーザ入力４１８を処理するにコンパイルされたコード１を識別する。説明すると、ファイバクリエータ２６０８は、ネイティブライブラリを解析し、コンパイルされたコード１の機能が、ユーザ入力４１８を処理することが可能かどうかを判定する。さらに説明すると、ファイバクリエータ２６０８は、ユーザ入力４１８を解析して、ユーザ入力４１８が、レガシーゲームＮのプレイを開始するための要求を含むかを判定し、コンパイルされたコード部分ＣＣＰ１が、レガシーゲームＮのアニメーションを含むかを判定して、ユーザ入力４１８を処理するためのコンパイルされたコード１が、ネイティブライブラリに格納されているかを判定する。別の例として、ファイバクリエータ２６０８は、ネイティブライブラリを解析または検索して、ユーザ入力４２０が受信されたときにユーザ入力４２０を処理するにコンパイルされたコード１を識別する。説明すると、ファイバクリエータ２６０８は、ネイティブライブラリを解析し、コンパイルされたコード１の１つ以上の機能が、ユーザ入力４２０を処理することが可能かどうかを判定する。さらに説明すると、ファイバクリエータ２６０８は、ユーザ入力４２０を解析して、ユーザ入力４２０が仮想戦闘機をある方向に動かす要求を含むことを判定し、コンパイルされたコード部分ＣＣＰ２が、ある方向での戦闘機の動きを表す１つ以上の画像フレーム４２２を生成する操作を含むことを判定し、ユーザ入力４２０を処理するためのコンパイルされたコード１がネイティブライブラリ内に格納されていることを判定する。

【０４１４】

次のコンパイルされたコードが、ネイティブライブラリ内で特定され得ると判定すると、方法３２００の操作３２１０において、ファイバクリエータ２６０８は、コンパイルされたコードの開始アドレスなどのエントリポイントを用いてファイバ２８０２をセットアップする。例えば、ファイバクリエータ２６０８は、ユーザ入力４１８を受信すると、コンパイルされたコード１の開始アドレスを用いてファイバ２６０８を作成する。別の例として、ファイバクリエータ２６０８は、コンパイルされたコード部分ＣＣＰ２の開始サブアドレスを用いてファイバ２８０２を作成し、その後に操作２８０４が続き、操作２８０６がさらに続き、コンパイルされたコード２の開始アドレスがさらに続き、操作２８０８が続き、加えて、ユーザ入力４２０を受信すると、操作２８１０が続く。操作３２０４は、操作３２１０の後に実行される。

【０４１５】

一方で、次にコンパイルされたコードがネイティブライブラリ内で特定できないと判定すると、方法３２００の実行は図３２Ｂに続く。例えば、次のコンパイルされたコードがネイティブライブラリ内で特定できないことを示す信号が、ファイバクリエータ２６０８から基本ブロッククリエータ６０４に送信される。

【０４１６】

図３２Ｂは、方法３２００の続きである。方法３２００の操作６５４において、次のコンパイルされたコードがネイティブライブラリ内で特定できないことを示す信号を受信すると、基本ブロッククリエータ６０４は、ユーザ入力４１８または４２０などのユーザ入

力を処理するための基本ブロック（ｎ＋１）～ｐのうちの１つ以上が、キャッシュ１０２内に格納されているかどうかを判定する。ユーザ入力を処理するためのブロック（ｎ＋１）～ｐのうちの１つ以上がキャッシュされていると判定すると、基本ブロック（ｎ＋１）～ｐのうちの１つ以上を動作する操作６５６が実行される。例えば、ブロックディスパッチャ３０２は、キャッシュ１０２から基本ブロック１～ｎのうちの１つ以上を読み取るために、ブロックリーダ６０８に要求を送信する。コマンドを受信すると、ブロックリーダ６０８（図６Ａ）は、キャッシュ１０２から基本ブロック（ｎ＋１）～ｐのうちの１つ以上を読み取り、基本ブロック（ｎ＋１）～ｐのうちの１つ以上をブロックディスパッチャ３０２（図３）に提供する。

【０４１７】

基本ブロック１～ｎのうちの１つ以上を受信すると、ブロックディスパッチャ３０２は、基本ブロック（ｎ＋１）～ｐのうちの１つ以上を実行して、レガシーゲームＮのアニメーションまたは仮想環境４５２（図４Ｂ）を生成する。説明すると、ユーザ入力４２０が受信されると、ブロックディスパッチャ３０２は、基本ブロック（ｎ＋１）～ｐのうちの１つ以上を実行して、１つ以上の画像フレーム４２２を生成し、この画像フレームは、仮想オブジェクト４５４の位置及び方向、仮想オブジェクト４５４のパラメータ、仮想環境４５２内の他の仮想オブジェクトの位置及び方向、及び仮想環境４５２内の他の仮想オブジェクトのパラメータなどの仮想環境データを含む。ブロックディスパッチャ３０２は、エミュレーションプロセッサシステム４０９のＧＰＵに１つ以上の画像フレーム４２２を提供し、ディスプレイデバイス４１０（図４Ａ）のディスプレイ画面上にまたはＨＭＤに仮想環境４５２を表示する。別の例として、ブロックディスパッチャ３０２は、基本ブロック（ｎ＋１）～ｐのうちの１つ以上を実行して１つ以上のオーディオフレームを生成し、１つ以上のオーディオフレームをクライアントデバイスに提供して、１つ以上のオーディオフレームに基づいてサウンドを出力する。方法３２００は、操作６５６の後に繰り返される。別の説明として、ユーザ入力４１８が受信されると、基本ブロック（ｎ＋１）～ｐのうちの１つ以上が実行され、レガシーゲームＮのアニメーションを表示するための画像データを生成し、画像データは、ブロックディスパッチャ３０２からアニメーションを表示するためのディスプレイデバイス４１０またはＨＭＤに提供されている。一方で、ユーザ入力４１８を処理するためのブロック（ｎ＋１）～ｐのうちの１つ以上がキャッシュされていないと判定すると、方法３２００は図３２Ｃに続く。

【０４１８】

図３２Ｃは、図３２Ｂの方法３２００のフローチャートの続きである。方法３２００では、操作６５８、６６０、及び６６２が実行される。

【０４１９】

一実施形態では、操作３２０８において次のコンパイルされたコードがネイティブライブラリ内で特定できず、図３２Ｂに示される操作が実行されないとき、方法３２００がクラッシュすることに留意されたい。方法３２００がクラッシュすると、レガシーゲームＮがクラッシュして実行できなくなる。

【０４２０】

図３３は、システム３３００の一実施形態の図であり、ＡｏＴエグゼキュータ２６１２とＧＰＵエミュレータ２１２２との間の結合を説明している。システム３３００は、ＡｏＴエグゼキュータ２６１２及びＧＰＵエミュレータ２１２２を含む。ＡｏＴエグゼキュータ２６１２は、ＧＰＵエミュレータ２１２２に結合されている。

【０４２１】

ユーザ入力４１８またはユーザ入力４２０（図４Ａ）などのユーザ入力に応答して、ＡｏＴエグゼキュータ２６１２は、ユーザ入力を処理する。例えば、ＡｏＴエグゼキュータ２６１２は、コンパイルされたコード１、またはコンパイルされたコード２、またはそれらの組み合わせなどの１つ以上のコンパイルされたコードを実行する。この例では、ＡｏＴエグゼキュータ２６１２は、操作２８０４、または操作２８０８、またはそれらの組み合わせなどの、レジスタ状態を格納するための１つ以上の操作をさらに実行する。また、

この例では、ＡｏＴエグゼキュータ２６１２は、画像フレーム４２２を生成するために、ファイバ２８０２（図２８）などのファイバによって提供される方向性に従って、操作２８０６、または操作２８１０、またはそれらの組み合わせなどのコンパイルされたコード間で切り替えるための１つ以上の操作を実行する。画像フレーム４２２は、ＡｏＴエグゼキュータ２６１２からＧＰＵエミュレータ２１２２に提供される。ＧＰＵエミュレータ２１２２は、１つ以上の更新されたグラフィックパラメータを画像フレーム４２２に適用して、画像フレーム２１２３を出力する。

【０４２２】

図３４は、コンパイルされたコードジェネレータ２６０１及びランタイム構成要素２６０３の場所を説明するためのシステム３４００の実施形態の図である。システム３４００は、サーバシステム４０４、ゲームコンソール４０２、コンピュータネットワーク４０８、及びハンドヘルドコントローラ４１４を含む。サーバシステム４０４は、コンパイルされたコードジェネレータ２６０１及びＮＩＣ３４０２を含む。ＮＩＣ３４０２は、コンパイルされたコードジェネレータ２６０１に結合されている。ゲームコンソール４２は、コンパイルされたコード１、コンパイルされたコード２（図２８）、及びコンパイルされたコード３などの複数のコンパイルされたコードを含むメモリデバイス４０６を含む。例えば、本明細書で説明されるコンパイルされたコードは、ゲームコンソール４０２内に事前に格納されている。説明すると、ユーザ１によるゲームコンソール４０２の購入時に、コンパイルされたコードは、メモリデバイス４０６に格納されている。

【０４２３】

同様に、レガシーゲームコードｇｃＮは、メモリデバイス４０６に格納されている。例えば、ゲームコードｇｃＮは、ゲームコンソール４０２内に事前に格納されている。説明すると、ユーザ１によるゲームコンソール４０２の購入時に、ゲームコードｇｃＮは、メモリデバイス４０６に格納されている。また、ランタイム構成要素２６０３は、ゲームコンソール４０２内に配置されている。説明すると、ユーザ１によるゲームコンソール４０２の購入時に、ランタイム構成要素２６０３は、ゲームコンソール４０２内にある。ランタイムコンポーネント２６０３は、入力プロセッサシステム４０７、メモリデバイス４０６、及びＮＩＣ１２１２に結合されている。

【０４２４】

入力プロセッサシステム４０８は、ハンドヘルドコントローラ４１４から、ユーザ入力４１８またはユーザ入力４１９などのユーザ入力を受信する。ユーザ入力４１８が、入力プロセッサシステム４０７によって受信されると、入力プロセッサシステム４０７は、ユーザ入力４１８を変換する必要がないと判定し、ランタイムコンポーネント２６０３にユーザ入力４１８を提供する。ユーザ入力４１８を受信すると、ランタイムコンポーネント２６０３は、ユーザ入力４１８を処理する。ユーザ入力４１８の処理中に、ＡｏＴエグゼキュータ２６１２は、メモリデバイス４０６からの１つ以上のコンパイルされたコードにアクセスし、ゲームコードｇｃＮからの画像データまたはオーディオデータなどのデータにアクセスし、ファイバ２８０２（図２８）によって提供される方向性に従ってコンパイルされたコードをデータに適用し、１つ以上の画像フレーム４２２を生成する。例えば、コンパイルされたコード１は、ファイバ２８０２の方向性に従って、ＡｏＴエグゼキュータ２６１２によって実行される。この例では、コンパイルされたコード１の実行中に、アニメーション画像データなどの画像データが、ゲームコードｇｃＮからアクセスされて、レガシーゲームＮの１つ以上の画像フレーム４２２を生成する。さらにこの例では、１つ以上の画像フレーム４２２が、レガシーゲームＮのアニメーションを表示するために、ゲームコンソール４０２からディスプレイデバイス４１０に送信される。

【０４２５】

さらに、ユーザ入力４１９が入力プロセッサシステム４０７によって受信されると、入力プロセッサシステム４０７は、ユーザ入力４１９をユーザ入力４２０に変換し、ユーザ入力４２０をランタイムコンポーネント２６０３に提供する。ユーザ入力４２０を受信することに応答して、ランタイムコンポーネント２６０３は、ユーザ入力４２０を処理する

。ユーザ入力４２０の処理中に、ＡｏＴエグゼキュータ２６１２は、メモリデバイス４０６から、コンパイルされたコード１及び２などの１つ以上のコンパイルされたコードにアクセスし、ゲームコードｇｃＮから、画像データまたはオーディオデータなどのデータにアクセスし、ファイバ２８０２によって提供される方向性に従ってコンパイルされたコードをデータに適用して、１つ以上の画像フレーム４２２を生成する。また、この例では、１つ以上の画像フレーム４２２が、レガシーゲームＮの表示のために、ゲームコンソール４０２からディスプレイデバイス４１０に送られる。

【０４２６】

一実施形態では、コンパイルされたコードは、サーバシステム４０４のコンパイルされたコードジェネレータ２６０１によって生成され、サーバシステム４０４の１つ以上のメモリデバイス内に格納される。ユーザ入力４１８は、ゲームコンソール４０２によって受信されると、ゲームコンソール４１８からコンピュータネットワーク４０８を介してコンパイルされたコードジェネレータ２６０１に送信される。ユーザ入力４１８を受信することに応答して、コンパイルされたコードジェネレータ２６０１は、サーバシステム４０４の１つ以上のメモリデバイスからコンパイルされたコードにアクセスし、コンパイルされたコードをＮＩＣ３４０２に送信する。ＮＩＣ３４０２は、ネットワーク通信プロトコルをコンパイルされたコードに適用して、１つ以上のパケットを生成し、１つ以上のパケットを、コンピュータネットワーク４０８を介してゲームコンソール４０２に送信する。ＮＩＣ１２１２は、１つ以上のパケットを受信し、ネットワーク通信プロトコルを１つ以上のパケットに適用して、コンパイルされたコードを抽出し、コンパイルされたコードをランタイムコンポーネント２６０３に提供する。ランタイムコンポーネント２６０３は、ＮＩＣ１２１２からコンパイルされたコードを受信し、コンパイルされたコードをメモリデバイス４０６内に格納する。ユーザ入力４２０が入力プロセッサシステム４０７から受信されると、ランタイムコンポーネント２６０３のファイバクリエータ２６０８は、ファイバ２８０２を生成し、コンパイルされたコードの実行の方向性を提供する。

【０４２７】

一実施形態では、ＡｏＴエグゼキュータ２６１２は、更新されたマシン９２０（図９Ｂ）のＣＰＵ９２２によって実行されるコンピュータプログラムである。一実施形態では、ＡｏＴエグゼキュータ２６１２は、メモリシステム９２６内に、またはキャッシュ９２８（図９Ｂ）内に格納される。ＣＰＵ９２２は、メモリシステム９２６またはキャッシュ９２８からＡｏＴエグゼキュータ２６１２にアクセスし、ＡｏＴエグゼキュータ２６１２を実行する。

【０４２８】

一実施形態では、ランタイムコンポーネント２６０３は、ゲームコンソール４０２のＣＰＵ９２２による実行のために、メモリデバイス４０６に事前に格納されている。

【０４２９】

図３５は、サーバシステム４０４内のＡｏＴプロセッサシステム２６００の場所を説明するためのシステム３５００の実施形態の図である。システム３５００は、サーバシステム４０４、ディスプレイデバイス４１０、コンピュータネットワーク４０８、及びハンドヘルドコントローラ４１４を含む。サーバシステム４０４は、メモリデバイス４１２、ＡｏＴプロセッサシステム２６００、及び入力プロセッサシステム４０７を含む。メモリデバイス４１２は、入力プロセッサシステム４０７に結合されているＡｏＴプロセッサシステム２６００に結合されている。入力プロセッサシステム４０７は、コンピュータネットワーク４０８に結合されているＮＩＣ３４０２に結合されている。ＮＩＣ３４０２は、ＡｏＴプロセッサシステム２６００に結合されている。

【０４３０】

ユーザ入力４１８は、ハンドヘルドコントローラ４１４からディスプレイデバイス４１０に送信される。ディスプレイデバイス４１０のＮＩＣは、ネットワーク通信プロトコルをユーザ入力４１８に適用して、１つ以上のパケットを生成し、コンピュータネットワーク４０８を介してＮＩＣ３４０２にパケットを送信する。ＮＩＣ３４０２は、ネットワー

ク通信プロトコルを１つ以上のパケットに適用して、１つ以上のパケットからユーザ入力４１８を抽出し、ユーザ入力４１８を入力プロセッサシステム４０７に送信する。

【０４３１】

入力プロセッサシステム４０７は、ユーザ入力４１８を変換することなく、ユーザ入力４１８をＡｏＴプロセッサシステム２６００に提供する。ユーザ入力４１８が受信される前に、コンパイルされたコードは、ＡｏＴプロセッサシステム２６００によって生成される。ＡｏＴプロセッサシステム２６００は、ユーザ入力４１８を処理して、ファイバ２８０２（図２８）によって提供される指示に従って、１つ以上のコンパイルされたコードを実行する。ユーザ入力４１８は、処理されて、１つ以上の画像フレーム４２２を生成する。画像フレーム４２２は、ＡｏＴプロセッサシステム２６００からＮＩＣ３４０２に提供される。ＮＩＣ３４０２は、ネットワーク通信プロトコルを１つ以上の画像フレーム４２２に適用して、１つ以上のパケットを生成し、コンピュータネットワーク４０８を介してディスプレイデバイス４１０にパケットを送信する。１つ以上のパケットを受信すると、ディスプレイデバイス４１０のＮＩＣは、ネットワーク通信プロトコルを１つ以上のパケットに適用して、１つ以上のパケットから１つ以上の画像フレーム４２２を取得する。ディスプレイデバイス４１０のＧＰＵは、１つ以上の画像フレーム４２２を、レガシーゲームＮのアニメーションの画像などの画像として表示する。

【０４３２】

さらに、レガシーゲームＮのプレイ中に、ハンドヘルドコントローラ４１４は、ユーザ入力４１９を生成し、ユーザ入力４１９をディスプレイデバイス４１０に送信する。ディスプレイデバイス４１０のＮＩＣは、ネットワーク通信プロトコルをユーザ入力４１９に適用して、１つ以上のパケットを生成し、コンピュータネットワーク４０８を介してＮＩＣ３４０２にパケットを送信する。ＮＩＣ３４０２は、ネットワーク通信プロトコルを１つ以上のパケットに適用して、１つ以上のパケットからユーザ入力４１９を抽出し、ユーザ入力４１９を入力プロセッサシステム４０７に送信する。

【０４３３】

入力プロセッサシステム４０７は、ユーザ入力４１９をユーザ入力４２０に変換し、ユーザ入力４２０をＡｏＴプロセッサシステム２６００に提供する。ＡｏＴプロセッサシステム２６００は、ユーザ入力４２０を処理して、１つ以上の画像フレーム４２２を生成する。画像フレーム４２２は、ＡｏＴプロセッサシステム２６００からＮＩＣ３４０２に提供される。ＮＩＣ３４０２は、ネットワーク通信プロトコルを１つ以上の画像フレーム４２２に適用して、１つ以上のパケットを生成し、コンピュータネットワーク４０８を介してディスプレイデバイス４１０にパケットを送信する。１つ以上のパケットを受信すると、ディスプレイデバイス４１０のＮＩＣは、ネットワーク通信プロトコルを１つ以上のパケットに適用して、１つ以上のパケットから１つ以上の画像フレーム４２２を取得する。ディスプレイデバイス４１０のＧＰＵは、１つ以上の画像フレーム４２２を、仮想環境４５２（図４Ａ）などの画像として表示する。

【０４３４】

一実施形態では、ディスプレイデバイス４１０の代わりに、ＨＭＤが使用される。

【０４３５】

図３６は、本開示の実装に従って、クラウドビデオゲームをクライアントデバイスにストリーミングするために実行される様々な操作を概念的に示すフロー図である。クライアントデバイスの例には、ハンドヘルドコントローラとディスプレイデバイスの組み合わせ、スマートフォン、ハンドヘルドコントローラとディスプレイデバイスとゲームコンソールの組み合わせ、ハンドヘルドコントローラとＨＭＤの組み合わせ、ハンドヘルドコントローラとゲームコンソールとＨＭＤの組み合わせ、タブレット、及びコンピュータが含まれる。ゲームサーバ３６０２は、ビデオゲームまたはゲームコードＧＣＮなどのゲームプログラムを実行し、生の（非圧縮の）ビデオ３６０４及びオーディオ３６０６を生成する。仮想環境４５２（図４Ａ）及び仮想環境４５２の表現中のオーディオ出力は、ビデオ３６０４及びオーディオ３６０６の例である。ゲームサーバ３６０２は、サーバシステム４

０４（図４Ａ）の一例である。示された図内の参照番号３６０８に示されているように、ビデオ３６０４及びオーディオ３６０６は、ストリーミング目的でキャプチャされ符号化される。符号化は、ビデオ及びオーディオストリームの圧縮を提供して、帯域幅の使用量を低減し、ゲーム体験を最適化する。符号化形式の例には、Ｈ．２６５／ＭＰＥＧ－Ｈ、Ｈ．２６４／ＭＰＥＧ－４、Ｈ．２６３／ＭＰＥＧ－４、Ｈ．２６２／ＭＰＥＧ－２、ＷＭＶ、ＶＰ６／７／８／９、等が含まれる。符号化されたオーディオ３６１０及び符号化されたビデオ３６１２は、コンピュータネットワーク４０８の一例であるコンピュータネットワーク３６２０を介した送信の目的のために、参照番号３６１４に示されるように、ネットワークパケットにさらにパケット化される（図４Ａ）。いくつかの実施形態では、ネットワークパケット符号化プロセスはまた、データ暗号化プロセスを採用し、それにより、強化されたデータセキュリティを提供する。図示した実施態様では、オーディオパケット３６１６及びビデオパケット３６１８は、コンピュータネットワーク３６２０を介した転送のために生成される。

【０４３６】

ゲームサーバ３６０２はさらに、触覚フィードバックデータ３６２２を生成し、これもまた、ネットワーク送信のためにネットワークパケットにパケット化される。図示した実装形態では、触覚フィードバックパケット３６２４は、コンピュータネットワーク３６２０を介した転送のために生成される。

【０４３７】

生のビデオ及びオーディオ及び触覚フィードバックデータを生成する前述の操作は、データセンタのゲームサーバ３６０２上で実行され、ビデオ及びオーディオを符号化し、符号化されたオーディオ／ビデオ及び触覚フィードバックデータを転送のためにパケット化する操作は、データセンタのストリーミングエンジンによって実行される。示されるように、オーディオ、ビデオ、及び触覚フィードバックパケットは、コンピュータネットワーク３６２０を介して転送される。参照番号３６２６において示されているように、オーディオパケット３６１６、ビデオパケット３６１８、及び触覚フィードバックパケット３６２４は、クライアントデバイスによって、例えば、解析されるなどして分解され、クライアントデバイスにおいて、ネットワークパケットから符号化されたオーディオ３６２８、符号化されたビデオ３６３０、及び触覚フィードバックデータ３６２２を抽出する。データが暗号化されている場合、データも暗号化される。符号化されたオーディオ３６２８及び符号化されたビデオ３６３０は、次いで、参照番号３６３４に示されるように、クライアントデバイスによって復号化され、クライアントデバイスのディスプレイデバイス３６４０上でレンダリングするためのクライアント側の生のオーディオ及びビデオデータを生成する。触覚フィードバックデータ３６２２は、クライアントデバイスのプロセッサによって処理されて、コントローラデバイス３６４２または他のインターフェースデバイス、例えば、ＨＭＤなどで触覚フィードバック効果を生成し、それらを介して触覚効果がレンダリングされることができる。コントローラデバイス３６４２は、クライアントデバイスのハンドヘルドコントローラ４１４（図４Ａ）または１２１２（図１２）などのハンドヘルドコントローラの一例である。触覚効果の一例は、コントローラデバイス３６４２のバイブレーションまたはランブルである。

【０４３８】

ビデオゲームは、プレイヤーまたはユーザの入力に応答し、そのため、プレイヤー入力の転送及び処理について上述した同様の手順フローが実行されるが、クライアントデバイスからサーバへの逆方向であることが理解されよう。図示されるように、コントローラデバイス３６４２または別の更新されたコントローラ、例えば、ＨＭＤなど、またはそれらの組み合わせは、入力データ３６４８を生成する。入力データ３６４８は、ユーザ入力としてコンピュータネットワーク３６２０を介してデータセンタに転送するために、クライアントデバイスでパケット化される。入力データパケット３６４６は、ゲームサーバ３６０２によってパケット解除及び再構成されて、データセンタ側で入力データ３６４８を定義する。入力データ３６４８は、ゲームサーバ３６０２に供給され、ゲームサーバ３６０

２は、入力データ３６４８を処理して、レガシーゲームＮのゲーム状態を生成する。

【０４３９】

オーディオパケット３６１６、ビデオパケット３６１８、及び触覚フィードバックパケット３６２４のコンピュータネットワーク３６２０を介した転送中に、いくつかの実施形態では、コンピュータネットワーク３６２０を介したデータの送信が監視されて、サービス品質を保証する。例えば、アップストリーム及びダウンストリームネットワーク帯域幅の両方を含む、コンピュータネットワーク３６２０のネットワーク状態は、参照番号３６５０によって示されるように監視され、ゲームストリーミングは、利用可能な帯域幅の変化に応じて調整される。すなわち、ネットワークパケットの符号化及び復号化は、参照番号３６５２によって示されるように、現在のネットワーク状態に基づいて制御される。

【０４４０】

図３７は、ゲームコンソール３７００の実施形態のブロック図であり、ゲームコンソール３７００は、クライアントデバイスのディスプレイデバイスとのインターフェースと互換性があり、コンピュータネットワーク３６２０（図３６）を介して、サーバシステム４０４（図４Ａ）などのゲームホスティングシステムと通信することができる。ゲームコンソール３７００は、ゲームコンソール４０２（図４Ａ）及びゲームコンソール１２０２（図１２）の例である。ゲームコンソール３７００は、データセンタ内に配置されているか、またはユーザ１または２などのプレイヤーが位置する場所に配置されている。いくつかの実施形態では、ゲームコードＧＣＮまたはファイバ２８０２（図２８）に従ってコンパイルされたコードを実行するために、ゲームコンソール３７００が使用され、ＨＭＤ３７０５上にレガシーゲームＮを表示する。ゲームコンソール３７００には、ゲームコンソール３７００に接続可能な様々な周辺機器が設けられている。ゲームコンソール３７００は、セルプロセッサ３７２８、ダイナミックランダムアクセスメモリ（ＸＤＲＡＭ）ユニット３７２６、専用ビデオランダムアクセスメモリ（ＶＲＡＭ）ユニット３７３２を備えたリアリティシンセサイザグラフィックプロセッサユニット３７３０、及び入出力（Ｉ／Ｏ）ブリッジ３７３４を有する。ゲームコンソール３７００はまた、ディスク３７４０ａから読み取るためのＢｌｕＲａｙ（登録商標）ディスク読み取り専用メモリ（ＢＤ－ＲＯＭ）光ディスクリーダ３７４０と、Ｉ／Ｏブリッジ３７３４を介してアクセス可能なリムーバブルスロットインハードディスクドライブ（ＨＤＤ）３７３６と、を有する。オプションとして、ゲームコンソール３７００はまた、コンパクトフラッシュ（登録商標）メモリカードを読み取るためのメモリカードリーダ３７３８、メモリスティック（登録商標）メモリカード、等も含み、これは、Ｉ／Ｏブリッジ３７３４を介して同様にアクセス可能である。Ｉ／Ｏブリッジ３７３４は、ＵＳＢ２．０ポート３７２４、ギガビットイーサネットポート３７２２、ＩＥＥＥ８０２．１１ｂ／ｇワイヤレスネットワーク（Ｗｉ－Ｆｉ（登録商標））ポート３７２０、及びＢｌｕｅｔｏｏｔｈ接続をサポートできるＢｌｕｅｔｏｏｔｈ（登録商標）ワイヤレスリンクポート３７１８にも接続する。

【０４４１】

操作中、Ｉ／Ｏブリッジ３７３４は、ゲームコントローラからの、及びＨＭＤ３７０５からのデータを含む、すべてのワイヤレス、ＵＳＢ、及びイーサネットデータを取り扱う。例えば、プレイヤーが、ゲームコードＧＣＮなどのゲームコードの一部分の実行によって生成されたレガシーゲームＮをプレイしているとき、Ｉ／Ｏブリッジ３７３４は、ゲームコントローラ３６４２から、及び／またはＨＭＤ３７０５から、Ｂｌｕｅｔｏｏｔｈリンクを介して本明細書で説明する入力データまたは入力信号を受信し、入力データをセルプロセッサ３７２８に向け、セルプロセッサ３７２８は、それに応じて、レガシーゲームＮの現在の状態を更新する。例として、ＨＭＤ３７０５内のカメラは、プレイヤーのジェスチャをキャプチャして、ジェスチャを表す画像を生成する。

【０４４２】

ワイヤレス、ＵＳＢ、及びイーサネットポートはまた、ゲームコントローラ３６４２及び３７０３、ＨＭＤ３７０５、及び、例えば、リモートコントロール３７０４、キーボード３７０６、マウス３７０８、例えば、Ｓｏｎｙ　Ｐｌａｙｓｔａｔｉｏｎ　Ｐｏｒｔａ

ｂｌｅ（登録商標）エンターテインメントデバイス、等などのポータブルエンターテインメントデバイス３７１０、例えば、ＥｙｅＴｏｙ（登録商標）ビデオカメラ３７１２、等などのビデオカメラ、マイクロフォンヘッドセット３７１４、及びマイクロフォン３７１５などの、その他の周辺機器用の接続を提供する。ポータブルエンターテインメントデバイス３７１０は、ゲームコントローラの一例である。いくつかの実施形態では、そのような周辺機器は、無線でゲームコンソール３７００に接続され、例えば、ポータブルエンターテインメントデバイス３７１０は、Ｗｉ－Ｆｉ（登録商標）アドホック接続を介して通信し、一方で、マイクロフォンヘッドセット３７１４は、Ｂｌｕｅｔｏｏｔｈリンクを介して通信する。

【０４４３】

これらのインターフェースの提供は、ゲームコンソール３７００が、デジタルビデオレコーダ（ＤＶＲ）、セットトップボックス、デジタルカメラ、ポータブルメディアプレイヤー、ボイスオーバーインターネットプロトコル（ＩＰ）電話、モバイル電話、プリンタ、スキャナなどの他の周辺機器とも互換性がある可能性があることを意味する。

【０４４４】

さらに、レガシーメモリカードリーダ３７１６は、ＵＳＢポート３７２４を介してゲームコンソール３７００に接続され、ゲームコンソール３７００によって使用される種類のメモリカード３７４８の読み取りを可能にする。ゲームコントローラ３６４２及び３７０３、ならびにＨＭＤ３７０５は、Ｂｌｕｅｔｏｏｔｈリンク３７１８を介してゲームコンソール３７００と無線で通信するように操作可能であるか、またはＵＳＢポート３７２４に接続されるように操作可能であり、それによって電力も受け取り、これによってゲームコントローラ３６４２及び３７０３、ならびにＨＭＤ３７０５の電池を充電する。いくつかの実施形態では、ゲームコントローラ３６４２及び３７０３の各々、ならびにＨＭＤ３７０５は、メモリ、プロセッサ、メモリカードリーダ、例えば、フラッシュメモリなどの固定メモリ、例えば、照光式球形セクション、発光ダイオード（ＬＥＤ）、または赤外線ライト、等、などの発光体、超音波通信用のマイクロフォン及びスピーカー、音響チャンバ、デジタルカメラ、内部クロック、例えば、ゲームコンソール３７００に面する球形セクションなどの認識可能な形状、及び、例えば、Ｂｌｕｅｔｏｏｔｈ、Ｗｉ－Ｆｉ、等、などのプロトコルを使用する無線デバイスを含む。

【０４４５】

ゲームコントローラ３６４２は、ユーザ１または２などのプレイヤーが両手で使用するように設計されたコントローラであり、ゲームコントローラ３７０３は、アタッチメントを備えた片手コントローラである。ＨＭＤ３７０５は、頭上やプレイヤーの眼前にフィットするように設計されている。１つ以上のアナログジョイスティック及び従来の制御ボタンに加えて、各ゲームコントローラ３６４２及び３７０３は、３次元場所判定の影響を受けやすい。同様に、ＨＭＤ３７０５は、３次元場所判定の影響を受けやすくなっている。その結果として、いくつかの実施形態では、ゲームコントローラ３６４２及び３７０３、ならびにＨＭＤ３７０５を用いるユーザによるジェスチャ及び動作は、従来のボタンまたはジョイスティックコマンドに加えて、またはその代わりに、ゲームへの入力として変換される。オプションで、例えば、Ｐｌａｙｓｔａｔｉｏｎ（登録商標）ポータブルデバイスなどの他のワイヤレス対応周辺デバイスが、コントローラとして使用される。Ｐｌａｙｓｔａｔｉｏｎ（登録商標）ポータブルデバイスの場合、追加のゲームまたは制御情報、例えば、制御命令または生存数などが、デバイスのディスプレイ画面上に提供されている。いくつかの実施形態では、例えば、ダンスマット（図示せず）、ライトガン（図示せず）、ステアリングホイール及びペダル（図示せず）、特注のコントローラ、などの他の代替的または補足的な制御デバイスが使用される。特注のコントローラの例は、高速応答クイズゲーム用の単一の、またはいくつかの大きなボタン（同様に図示せず）を含む。

【０４４６】

リモートコントロール３７０４はまた、Ｂｌｕｅｔｏｏｔｈリンク３７１８を介してゲームコンソール３７００と無線で通信するように操作可能である。リモートコントロール

３７０４は、ＢｌｕＲａｙ（登録商標）ディスクＢＤ－ＲＯＭリーダ３７４０の操作に適した、かつ、ディスクコンテンツのナビゲーションに適した制御を含む。

【０４４７】

ＢｌｕＲａｙ（登録商標）ディスクＢＤ－ＲＯＭリーダ３７４０は、従来の録音済み及び録音可能なＣＤ、及びいわゆるスーパーオーディオＣＤに加えて、ゲームコンソール３７００と互換性のあるＣＤ－ＲＯＭを読み取るように操作可能である。ＢｌｕＲａｙ（登録商標）ディスクＢＤ－ＲＯＭリーダ３７４０はまた、従来の事前記録及び記録可能なＤＶＤに加えて、ゲームコンソール３７００と互換性のあるデジタルビデオディスク－ＲＯＭ（ＤＶＤ－ＲＯＭ）を読み取るように操作可能である。ＢｌｕＲａｙ（登録商標）ディスクＢＤ－ＲＯＭリーダ３７４０はさらに、ゲームコンソール３７００と互換性のあるＢＤ－ＲＯＭ、ならびに、従来の事前記録された、及び記録可能なＢｌｕＲａｙディスクを読み取るように操作可能である。

【０４４８】

ゲームコンソール３７００は、オーディオコネクタ３７５０及びビデオコネクタ３７５２を介して、例えば、モニターまたはテレビなどの、ディスプレイ画面３７４４及び１つ以上のラウドスピーカー３７４６を有するディスプレイ及びサウンド出力デバイス３７４２に、リアリティシンセサイザーグラフィックスユニット３７３０を通じて生成されたか、または復号されたオーディオ及びビデオを供給するように操作可能であり、または、Ｂｌｕｅｔｏｏｔｈ（登録商標）ワイヤレスリンクポート３７１８を介して、ＨＭＤ３７０５のディスプレイデバイスにオーディオ及びビデオを供給するように操作可能である。オーディオコネクタ３７５０は、様々な実施形態において、従来のアナログ及びデジタル出力を含み、一方で、ビデオコネクタ３７５２は、コンポーネントビデオ、Ｓ－ビデオ、コンポジットビデオ、及び１つ以上のハイデフィニションマルチメディアインターフェース（ＨＤＭＩ）出力を様々に含む。その結果、ビデオ出力は、位相交互線（ＰＡＬ）や全米テレビジョンシステム委員会（ＮＴＳＣ）などの形式、または３７２０ｐ、１０８０ｉ、または１０８０ｐの高解像度であり得る。オーディオ処理、例えば、生成、復号化などは、セルプロセッサ３７０８によって実行される。ゲームコンソール３７００のオペレーティングシステムは、Ｄｏｌｂｙ（登録商標）５．１サラウンドサウンド、Ｄｏｌｂｙ（登録商標）シアターサラウンド（ＤＴＳ）、及びＢｌｕ－Ｒａｙ（登録商標）ディスクからの７．１サラウンドサウンドの復号化をサポートしている。ディスプレイ及びサウンド出力デバイス３７４２は、ディスプレイデバイス４１０の一例である（図４Ａ）。

【０４４９】

いくつかの実施形態では、例えば、ビデオカメラ３７１２などのビデオカメラは、単一の電荷結合デバイス（ＣＤＤ）、ＬＥＤインジケータ、及びハードウェアベースのリアルタイムデータ圧縮及び符号化装置を含み、圧縮されたビデオデータは、ゲームコンソール３７００による復号化のためのイントラ画像ベースの動画専門家集団（ＭＰＥＧ）規格などの適切な形式で送信されるようになっている。ビデオカメラ３７１２のＬＥＤインジケータは、ゲームコンソール３７００からの適切な制御データに応答して、例えば、著しく不利な照明状態などを照射するように配置されている。ビデオカメラ３７１２のいくつかの実施形態は、ＵＳＢ、Ｂｌｕｅｔｏｏｔｈ、またはＷｉ－Ｆｉ通信ポートを介してゲームコンソール３７００に接続する。ビデオカメラの様々な実施形態は、１つ以上の関連するマイクロフォンを含み、オーディオデータを送信することもできる。ビデオカメラのいくつかの実施形態では、ＣＣＤは、高解像度ビデオキャプチャに適した解像度を有する。使用時、ビデオカメラによってキャプチャされた画像は、ゲーム内に組み込まれる、またはゲーム制御入力として解釈される。別の実施形態では、ビデオカメラは、赤外線光を検出するのに適した赤外線カメラである。

【０４５０】

様々な実施形態において、例えば、ビデオカメラまたはリモートコントロールなどの周辺デバイスとのゲームコンソール３７００の通信ポートの１つを介したデータ通信を成功させるために、デバイスドライバなどの適切なソフトウェアが提供されている。

【０４５１】
いくつかの実施形態では、ゲームコンソール３７００、ゲームコントローラ３６４２（図３６）または３７０３、及びＨＭＤ３７０５を含む前述のシステムデバイスは、ＨＭＤ３７０５が、レガシーゲームＮのインタラクティブセッションのビデオを表示及びキャプチャすることを可能にする。システムデバイスは、レガシーゲームＮの対話型セッション、プレイヤー１と他のプレイヤーとの間の対話性を規定する対話型セッション、及びレガシーゲームＮの対話型セッションを開始する。システムデバイスはさらに、プレイヤーによって操作されるゲームコントローラ３６４２（図３６）もしくは３６０３、及び／またはＨＭＤ３７０５の初期位置及び向きを判定する。ゲームコンソール３７００は、プレイヤーとレガシーゲームＮとの間の双方向性に基づいてゲームの現在の状態を判定する。システムデバイスは、レガシーゲームＮとのプレイヤーの対話型セッション中に、ゲームコントローラ３６４２、３７０３及び／またはＨＭＤ３７０５の位置及び向きを追跡する。システムデバイスは、レガシーゲームＮの現在の状態、ならびにゲームコントローラ３６４２、３７０３及び／またはＨＭＤ３７０５の追跡された位置及び方向に基づいて、インタラクティブセッションの観客ビデオストリームを生成する。いくつかの実施形態では、ＨＨＣは、観客ビデオストリームをＨＨＣのディスプレイ画面上にレンダリングする。様々な実施形態において、ＨＭＤ３７０５はＨＭＤ３７０５のディスプレイ画面上に観客ビデオストリームをレンダリングする。
【０４５２】
図３８を参照すると、ＨＭＤ３８００の構成要素を示す図が示されている。ＨＭＤ３８００は、ＨＭＤ３７０５（図３７）の例である。ＨＭＤ３８００は、プログラム命令を実行するためのプロセッサ３８０１を含む。メモリデバイス３８０２は、記憶目的のために提供されている。メモリデバイス３８０２の例には、揮発性メモリ、不揮発性メモリ、またはそれらの組み合わせが含まれる。例えば、プレイヤーが見る保存データなどから生成された画像フレームの表示などの視覚的インターフェースを提供するディスプレイデバイス３８０４が含まれている。電池３８０６は、ＨＭＤ３８００の電源として提供されている。モーション検出モジュール３８８０は、磁力計３８１０、加速度計３８１２、及びジャイロスコープ３８１４などの様々な種類のモーションセンシティブハードウェアのいずれかを含む。
【０４５３】
加速度計は、加速度と重力によって引き起こされる反力を測定するためのデバイスである。単軸及び多軸モデルは、異なる方向における加速度の大きさ及び方向を検出するために利用可能である。加速度計は、傾き、バイブレーション、衝撃を感知するために使用される。一実施形態では、３つの加速度計３８１２は、重力の方向を提供するために使用され、この方向は、２つの角度、例えば、ワールド空間ピッチ及びワールド空間ロールなどの絶対基準を与える。
【０４５４】
磁力計は、ＨＭＤ３８００の近傍の磁場の強度と方向を測定する。いくつかの実施形態では、３つの磁力計３８１０が、ＨＭＤ３８００内で使用され、ワールド空間のヨー角の絶対基準を確実にする。様々な実施形態において、磁力計は、±８０マイクロテスラである地球磁場にまたがるように設計されている。磁力計は金属により影響を受け、実際のヨーと単調なヨー測定値を提供する。いくつかの実施形態では、磁場は、実世界の環境において金属のために歪められ、これは、ヨー測定において歪みを引き起こす。様々な実施形態において、この歪みは、例えば、ジャイロスコープ３８１４、カメラ３８１６などの他のセンサからの情報を使用して較正される。一実施形態では、加速度計３８１２は、磁力計３８１０と共に使用されて、ＨＭＤ３８００の傾斜及び方位角を取得する。
【０４５５】
ジャイロスコープは、角運動量の原理に基づいて、向きを測定または維持するためのデバイスである。一実施形態では、ジャイロスコープ３８１４の代わりに、３つのジャイロスコープが、慣性感知に基づいて、それぞれの軸（ｘ、ｙ、及びｚ）を横切る動きに関す

る情報を提供する。ジャイロスコープは、高速回転の検出に役立つ。しかしながら、ジャイロスコープは、いくつかの実施形態では、絶対基準が存在しないと、時間の経過と共にドリフトする。このことが、ジャイロスコープを周期的にリセットし、これは、オブジェクト、加速度計、磁力計などの視覚的な追跡に基づいた位置／方向の判定など、他の利用可能な情報を使用して実行され得る。

【０４５６】

カメラ３８１６は、例えば、部屋、キャビン、自然環境など、プレイヤーを取り巻く実世界環境の画像及び画像ストリームをキャプチャするために提供される。様々な実施形態において、複数のカメラが、ＨＭＤ３８００に含まれており、複数のカメラは、例えば、プレイヤーがＨＭＤ３８００などのディスプレイを見ているときに、プレイヤーから離れる方向に向けられた後ろ向きのカメラ、及び例えば、プレイヤーがＨＭＤ３８００のディスプレイを見ているときにプレイヤーに向けられる、正面向きのカメラを含んでいる。さらに、いくつかの実施形態では、深度カメラ３８１８は、実世界環境内のオブジェクトの深度情報を感知するためにＨＭＤ３８００に含まれている。

【０４５７】

ＨＭＤ３８００は、オーディオ出力を提供するためのスピーカー３８２０を含む。また、いくつかの実施形態では、周囲環境からのサウンドを含む実世界環境からのオーディオ、及びプレイヤーによって行われたスピーチ、などをキャプチャするために、マイクロフォン３８２２が含まれている。ＨＭＤ３８００は、触覚フィードバックをプレイヤーに提供するための、例えば、バイブレーションデバイスなどの触覚フィードバックモジュール３８２４を含む。一実施形態では、触覚フィードバックモジュール３８２４は、ＨＭＤ３８００の動き及び／またはバイブレーションを起こして、プレイヤーに触覚フィードバックを提供することができる。

【０４５８】

ＬＥＤ３８２６は、ＨＭＤ３８００のステータスの視覚的インジケータとして提供されている。例えば、ＬＥＤは、バッテリレベル、電源オンなどを示し得る。カードリーダ３８２８は、ＨＭＤ３８００がメモリカードに情報を書き込み、そこから情報を読み取ることを可能にするために提供される。ＵＳＢインターフェース３８３０は、周辺デバイスの接続、または他の携帯デバイス、コンピュータなどの他のデバイスへの接続を可能にするためのインターフェースの一例として含まれている。ＨＭＤ３８００の様々な実施形態では、ＨＭＤ３８００のより大きな接続性を可能にするために、様々な種類のインターフェースのいずれかが含まれ得る。

【０４５９】

Ｗｉ－Ｆｉ（登録商標）モジュール３８３２は、無線ネットワーク技術を介したインターネットへの接続を可能にするために含まれている。また、ＨＭＤ３８００は、他のデバイスへのワイヤレス接続を可能にするＢｌｕｅｔｏｏｔｈ（登録商標）モジュール３８３４を含む。いくつかの実施形態では、他のデバイスに接続するための通信リンク３８３６も含まれている。一実施形態では、通信リンク３８３６は、無線通信のために赤外線送信を利用する。他の実施形態では、通信リンク３８３６は、他のデバイスとの通信のための様々な無線または有線伝送プロトコルのいずれかを利用する。

【０４６０】

入力ボタン／センサ３８３８は、プレイヤーに入力インターフェースを提供するために含まれている。ボタン、タッチパッド、ジョイスティック、トラックボールなど、様々な種類の入力インターフェースが含まれている。様々な実施形態において、超音波通信モジュール３８４０が、超音波技術を介して他のデバイスとの通信を容易にするためにＨＭＤ３８０２に含まれている。

【０４６１】

プレイヤーからの生理学的データの検出を可能にするために、生体センサ３８４２が含まれている。一実施形態では、バイオセンサー３８４２は、プレイヤーの皮膚を通して、プレイヤーの生体電気信号を検出するための１つ以上の乾式電極を含む。

【０４６２】
ＨＭＤ３８００の前述の構成要素は、ＨＭＤ３８００に含まれ得る単なる例示的な構成要素として説明されてきた。様々な実施形態において、ＨＭＤ３８００は、様々な前述の構成要素のいくつかを含むか、または含まない。
【０４６３】
図３９は、情報サービスプロバイダ（ＩＮＳＰ）アーキテクチャの実施形態を示している。ＩＮＳＰ３９０２は、地理的に分散し、例えば、ＬＡＮ、ＷＡＮ、またはそれらの組み合わせなどのコンピュータネットワーク３９０６を介して接続されたプレイヤーに多数の情報サービスを提供する。コンピュータネットワーク３９０６は、コンピュータネットワーク３６２０（図３６）の例である。ユーザ１は、クライアントデバイス３９２０－１を操作し、ユーザ２は、別のクライアントデバイス３９２０－２を操作し、ユーザ３は、さらに別のクライアントデバイス３９２０－３を操作する。
【０４６４】
いくつかの実施形態では、各クライアントデバイス３９２０－１、３９２０－２、及び３９２０－３は、ＣＰＵ、ディスプレイ、及び入出力（Ｉ／Ｏ）インターフェースを含む。各クライアントデバイス３９２０－１、３９２０－２、及び３９２０－３の例には、パーソナルコンピュータ（ＰＣ）、携帯電話、ネットブック、タブレット、ゲームシステム、携帯情報端末（ＰＤＡ）、ゲームコンソール３７００とディスプレイデバイスの組み合わせ、ＨＭＤ３８００（図３８）、ゲームコンソール３７００とＨＭＤ３８００の組み合わせ、デスクトップコンピュータ、ラップトップコンピュータ、及びスマートテレビ、等が含まれる。いくつかの実施形態では、ＩＮＳＰ３９０２は、クライアントデバイスのタイプを認識し、使用される通信方法を調整する。
【０４６５】
いくつかの実施形態では、ＩＮＳＰ３９０２は、株価更新などの１つのタイプのサービス、または放送メディア、ニュース、スポーツ、ゲームなどのような様々なサービスを提供する。さらに、各ＩＮＳＰによって提供されるサービスは動的であり、つまり、サービスはどの時点でも追加または削除できる。したがって、特定の個人に特定のタイプのサービスを提供するＩＮＳＰは、時間の経過とともに変化することがある。例えば、クライアントデバイス３９２０－１は、クライアントデバイス３９２０－１に近いＩＮＳＰによってサービス提供され、クライアントデバイス３９２０－１は、ユーザ１の地元にあり、クライアントデバイス３９２０－１は、ユーザ１が別の都市に移動するときに、異なるＩＮＳＰによってサービス提供される。地元のＩＮＳＰは、要求された情報とデータを新しいＩＮＳＰに転送し、情報が、新しい都市へとクライアントデバイス３９２０－１を「追従」し、データをクライアントデバイス３９２０－１に近づけてアクセスしやすくする。様々な実施形態では、マスターとサーバの関係は、クライアントデバイス３９２０－１の情報を管理するマスターＩＮＳＰと、マスターＩＮＳＰからの制御下でクライアントデバイス３９２０－１と直接インターフェースするサーバＩＮＳＰとの間に確立される。いくつかの実施形態では、クライアントデバイス３９２０－１が世界中を移動すると、ＩＮＳＰをより良い位置に置いて、クライアントデバイス３９２０－１にサービス提供して、これらのサービスを提供するものにするために、データは１つのＩＳＰから別のＩＳＰに転送される。
【０４６６】
ＩＮＳＰ３９０２は、コンピュータネットワーク３９０６を介して顧客にコンピュータベースのサービスを提供するアプリケーションサービスプロバイダ（ＡＳＰ）３９０８を含む。ＡＳＰモデルを使用して提供されるソフトウェアは、オンデマンドソフトウェアまたはｓｏｆｔｗａｒｅ　ａｓ　ａ　ｓｅｒｖｉｃｅ（ＳａａＳ）とも称されることもある。例えば、顧客関係管理などのコンピュータベースのサービスへのアクセスを提供する簡単な形式は、例えば、ハイパーテキスト転送プロトコル（ＨＴＴＰ）などの標準プロトコルを使用することによる。アプリケーションソフトウェアは、ベンダーのサーバ上にあり、ベンダー、及び／またはシンクライアントなどの他のリモートインターフェースによっ

て提供されている専用クライアントソフトウェアによって、ハイパーテキストマークアップ言語（ＨＴＭＬ）などを使用したウェブブラウザを介して各クライアントデバイス３９２０－１、３９２０－２、及び３９２０－３によってアクセスされる。

【０４６７】

広い地理的領域で提供されるサービスは、多くの場合、クラウドコンピューティングを使用する。クラウドコンピューティングは、動的にスケーラブルで、多くの場合、仮想化されたリソースがコンピュータネットワーク３９０６を介してサービスとして提供されるコンピューティングのスタイルである。ユーザ１から３は、それらをサポートする「クラウド」のテクノロジインフラストラクチャの専門家である必要はない。いくつかの実施形態では、クラウドコンピューティングは、Ｉｎｆｒａｓｔｒｕｃｔｕｒｅ ａｓ ａ Ｓｅｒｖｉｃｅ（ＩａａＳ）、Ｐｌａｔｆｏｒｍ ａｓ ａ Ｓｅｒｖｉｃｅ（ＰａａＳ）、及びＳｏｆｔｗａｒｅ ａｓ ａ Ｓｅｒｖｉｃｅ（ＳａａＳ）などの異なるサービスに分割される。クラウドコンピューティングサービスは、多くの場合、ウェブブラウザからアクセスされる一般的なビジネスアプリケーションをオンラインで提供し、一方で、ソフトウェア及びデータはサーバに格納されている。クラウドという用語は、コンピュータネットワーク３９０６が、どのようにコンピュータネットワーク図に描かれているか、かつ、秘匿する複雑なインフラストラクチャの抽象化であるかに基づいて、例えば、サーバ、ストレージ、ロジックなどを使用して、コンピュータネットワーク３９０６のメタファーとして使用される。

【０４６８】

さらに、ＩＮＳＰ３９０２にはゲーム処理プロバイダ（ＧＰＰ）３９１０が含まれており、これは、本明細書では、レガシーゲームＮなどの、シングルプレイヤー及びマルチプレイヤービデオゲームをプレイするためにクライアントデバイス３９２０－１、３９２０－２、及び３９２０－３によって使用されるゲーム処理サーバと呼ばれることもある。コンピュータネットワーク３９０６上でプレイされるほとんどのビデオゲームは、ゲームサーバへの接続を介して操作する。通常、ゲームは、クライアントデバイス３９２０－１、３９２０－２、及び３９２０－３からデータを収集し、それらを他のユーザによって操作されている他のクライアントに分配する専用のサーバアプリケーションを使用する。これはピアツーピアの配置よりも効率的で効果的であるが、別のサーバが、サーバアプリケーションをホストするために使用される。いくつかの実施形態では、ＧＰＰ３９１０は、クライアントデバイス３９２０－１、３９２０－２、及び３９２０－３間の通信を確立し、集中型ＧＰＰ３９１０にさらに依存することなく、情報を交換する。

【０４６９】

専用のＧＰＰは、クライアントとは独立して実行されるサーバである。このようなサーバは、通常、データセンタに配置された専用ハードウェア上で実行し、より多くの帯域幅及び専用処理能力を提供する。専用サーバは、ほとんどのＰＣベースのマルチプレイヤーゲームのためのゲームサーバをホストする方法である。大規模なマルチプレイヤーオンラインゲームは、通常、ゲームタイトルを所有しているソフトウェア会社がホストする専用サーバで実行され、コンテンツの制御及び更新をできるようにする。

【０４７０】

ブロードキャスト処理サーバ（ＢＰＳ）３９１２は、本明細書ではブロードキャスト処理プロバイダとも呼ばれ、オーディオまたはビデオ信号を視聴者に配信する。非常に狭い範囲の視聴者へのブロードキャストは、ナローキャスティングと呼ばれることがある。ブロードキャスト配信の最終レッグは、信号がクライアントデバイス３９２０－１、３９２０－２、及び３９２０－３に到達する方法であり、いくつかの実施形態では、ラジオ局やテレビ局からアンテナ及び受信器に放送で配信されるか、またはケーブルテレビまたはケーブルラジオまたはステーションを介した「無線ケーブル」を通じて配信される。コンピュータネットワーク３９０６はまた、様々な実施形態において、ラジオ信号またはテレビ信号のいずれかをクライアントデバイス３９２０－１、３９２０－２、及び３９２０－３に持ち込み、特に、マルチキャスティングで信号や帯域幅が共有されることを可能にする

。歴史的に、ブロードキャストは、いくつかの実施形態では、例えば全国放送、地域放送など、地理的領域によって区切られている。しかしながら、高速インターネットの急増により、コンテンツは世界のほぼすべての国に到達できるため、ブロードキャストは地域によって定義されていない。

【０４７１】

ストレージサービスプロバイダ（ＳＳＰ）３９１４は、コンピュータストレージスペースと関連管理サービスを提供する。ＳＳＰ３９１４はまた、周期的なバックアップとアーカイブも提供する。ストレージをサービスとして提供することにより、クライアントデバイス３９２０－１、３９２０－２、及び３９２０－３は、ストレージがサービスとして使用されていないときと比較して、より多くのストレージを使用する。別の大きな利点は、ＳＳＰ３９１４が、バックアップサービスを含み、クライアントデバイス３９２０－１、３９２０－２、及び３９２０－３は、ハードドライブが故障した場合でもデータを損失しない。さらに、いくつかの実施形態では、複数のＳＳＰは、クライアントデバイス３９２０－１、３９２０－２、及び３９２０－３から受信したデータの全体的または部分的なコピーを有し、クライアントデバイス３９２０－１、３９２０－２、及び３９２０－３が配置されている場所や、クライアントのタイプに関係なく、クライアントデバイス３９２０－１、３９２０－２、及び３９２０－３が効率的な方法でデータにアクセスすることを可能にする。例えば、プレイヤーは、自宅のコンピュータを介して、かつ、プレイヤーの移動中は携帯電話を介して個人ファイルにアクセスする。

【０４７２】

通信プロバイダ３９１６は、クライアントデバイス３９２０－１、３９２０－２、及び３９２０－３への接続を提供する。通信プロバイダ３９１６の一種は、コンピュータネットワーク３９０６へのアクセスを提供するインターネットサービスプロバイダ（ＩＳＰ）である。ＩＳＰは、ダイヤルアップ、デジタル加入者線（ＤＳＬ）、ケーブルモデム、ファイバ、ワイヤレスまたは専用の高速相互接続などのインターネットプロトコルデータグラムを送達するために適切なデータ転送技術を使用して、クライアントデバイス３９２０－１、３９２０－２、及び３９２０－３を接続する。通信プロバイダ３９１６はまた、いくつかの実施形態では、電子メール、インスタントメッセージング、及びショートメッセージサービス（ＳＭＳ）テキストメッセージなどのメッセージングサービスを提供する。別のタイプの通信プロバイダは、ネットワークサービスプロバイダ（ＮＳＰ）であり、これは、コンピュータネットワーク３９０６への直接バックボーンアクセスを提供することによって、帯域幅またはネットワークアクセスを販売する。ネットワークサービスプロバイダの例には、電気通信会社、データキャリア、ワイヤレス通信プロバイダ、インターネットサービスプロバイダ、高速インターネットアクセスを提供するケーブルテレビ事業者などが含まれる。

【０４７３】

データ交換３９１８は、ＩＮＳＰ６０２内のいくつかのモジュールを相互接続し、これらのモジュールを、コンピュータネットワーク３９０６を介してクライアントデバイス３９２０－１、３９２０－２、及び３９２０－３に接続する。データ交換３９１８は、様々な実施形態において、ＩＮＳＰ３９０２のすべてのモジュールが近接している小さな領域をカバーするか、または異なるモジュールが地理的に分散しているときに、広い地理的領域をカバーする。例えば、データ交換３９０２には、データセンタのキャビネット内の高速ギガビットイーサネットまたは大陸間仮想ＬＡＮが含まれる。

【０４７４】

いくつかの実施形態では、サーバシステム４０４（図４Ａ）とクライアントデバイス３９２０－１～３９２０－３との間の通信は、無線技術を使用して促進され得る。そのような技術には、例えば、５Ｇ無線通信技術が含まれ得る。

【０４７５】

一実施形態では、本明細書で説明するレガシーゲームＮなどのビデオゲームは、ゲーム機、パーソナルコンピュータのいずれかでローカルに実行されるか、またはサーバ上で実

行される。場合によっては、ビデオゲームは、データセンタの１つ以上のサーバによって実行される。ビデオゲームが実行されるとき、ビデオゲームのいくつかのインスタンスは、ビデオゲームのシミュレーションであり得る。例えば、ビデオゲームは、ビデオゲームのシミュレーションを生成する環境またはサーバによって実行され得る。いくつかの実施形態では、シミュレーションは、ビデオゲームのインスタンスである。他の実施形態では、シミュレーションは、エミュレータによって生成され得る。いずれの場合も、ビデオゲームがシミュレーションとして表される場合、そのシミュレーションは、ユーザ入力によってインタラクティブにストリーミング、実行、及び／または制御されることができるインタラクティブコンテンツをレンダリングするために実行されることが可能である。

【０４７６】

様々な実施形態において、本明細書に記載されるいくつかの実施形態の１つ以上の特徴は、本明細書に記載される残りの実施形態の１つ以上の１つ以上の特徴と組み合わされることに留意されたい。

【０４７７】

本開示に記載の実施形態は、ハンドヘルドデバイス、マイクロプロセッサシステム、マイクロプロセッサベースもしくはプログラム可能な消費者向け電気製品、小型コンピュータ、メインフレームコンピュータなどを含む様々なコンピュータシステム構成で実施されてよい。一実施態様では、本開示に記載の実施形態は、有線または無線ネットワークを通してリンクされたリモート処理デバイスによってタスクが行われる分散コンピューティング環境で実践される。

【０４７８】

前述の実施形態を念頭に置いて、一実施態様では、本開示に記載の実施形態は、コンピュータシステムに記憶されたデータを伴う様々なコンピュータ実施動作を採用することを理解されたい。これらの動作は、物理量の物理的操作を要する動作である。本開示に記載の実施形態の一部を形成する本明細書で説明される動作のうちのいずれも、有用な機械動作である。本開示に記載のいくつかの実施形態はまた、これらの動作を行うためのデバイスまたは装置に関する。装置は、必要な目的のために特別に構築される、または装置は、コンピュータに記憶されたコンピュータプログラムにより選択的に起動または構成される汎用コンピュータである。具体的には、一実施形態において、様々な汎用マシンは、本明細書の教示に従って書かれたコンピュータプログラムと共に使用される、または、必要な動作を行うためにさらに特化した装置を構築するほうがより好都合な場合がある。

【０４７９】

ある実施態様では、本開示に記載のいくつかの実施形態は、コンピュータ可読媒体上のコンピュータ可読コードとして具体化される。コンピュータ可読媒体は、データを格納することができる任意のデータ格納装置であり、これは、その後、コンピュータシステムによって読み取られることができる。コンピュータ可読媒体の例には、ハードドライブ、ネットワーク接続ストレージ（ＮＡＳ）、ＲＯＭ、ＲＡＭ、ＣＤ－ＲＯＭ、ＣＤ－ｒｅｃｏｒｄａｂｌｅ（ＣＤ－Ｒ）、ＣＤ－ｒｅｃｏｒｄａｂｌｅ（ＣＤ－ＲＷ））、磁気テープ、光学データ記憶装置、非光学データ記憶装置などが含まれる。一例として、コンピュータ可読媒体は、ネットワーク結合コンピュータシステム上に分散されたコンピュータ可読有形媒体を含み、コンピュータ可読コードが分散方式で格納及び実行されるようになっている。

【０４８０】

さらに、上述の実施形態のいくつかは、ゲーム環境に関して説明されているが、いくつかの実施形態では、ゲームの代わりに、例えば、ビデオ会議環境などの他の環境が使用される。

【０４８１】

本方法の操作は特定の順序で説明されているが、他のハウスキーピング操作が、操作の間に実行され得、もしくは、操作が、わずかに異なる時間に発生するように調整されることができるか、またはシステムに分散されることができ、このことが、重複操作の処理が

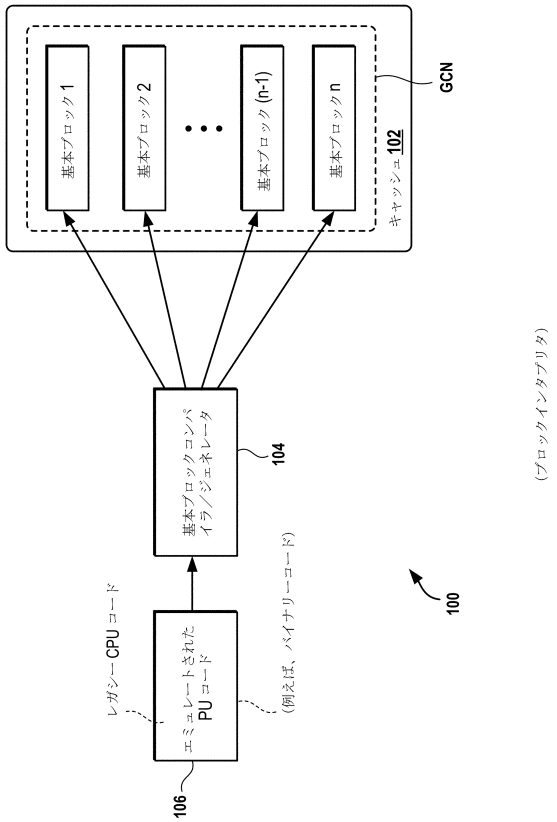
所望の方法で実行される限り、処理に関連する様々な間隔での処理操作の発生を可能とすることが理解されるべきである。

【０４８２】

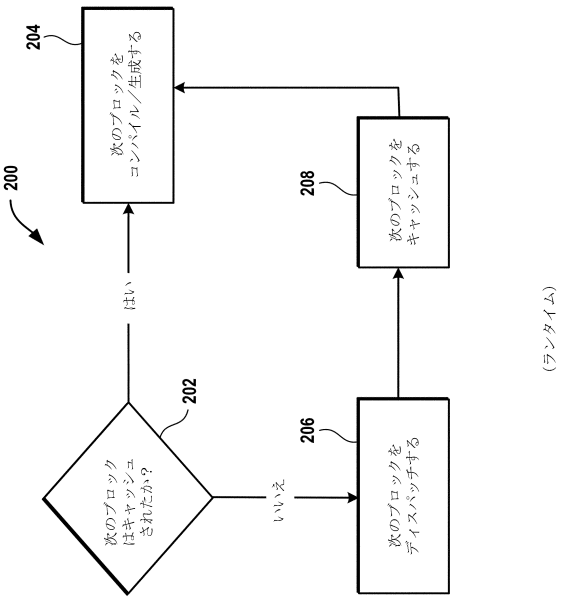
本開示に記載の前述の実施形態は、理解を明確にするためにある程度詳細に説明したが、添付の特許請求の範囲内で特定の変更及び修正を実施できることは明らかであろう。従って、本実施形態は、限定ではなく例示としてみなされるべきであり、本実施形態は、本明細書に記載される詳細に限定されるべきではなく、添付の請求項の範囲及び均等物の中で変更されてよい。

【図面】

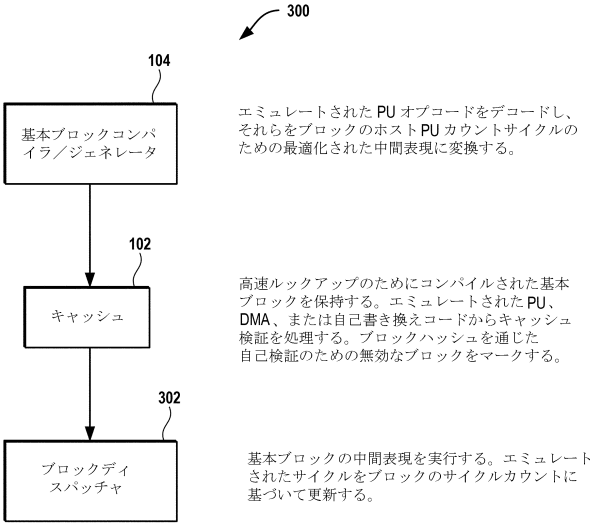
【図１】



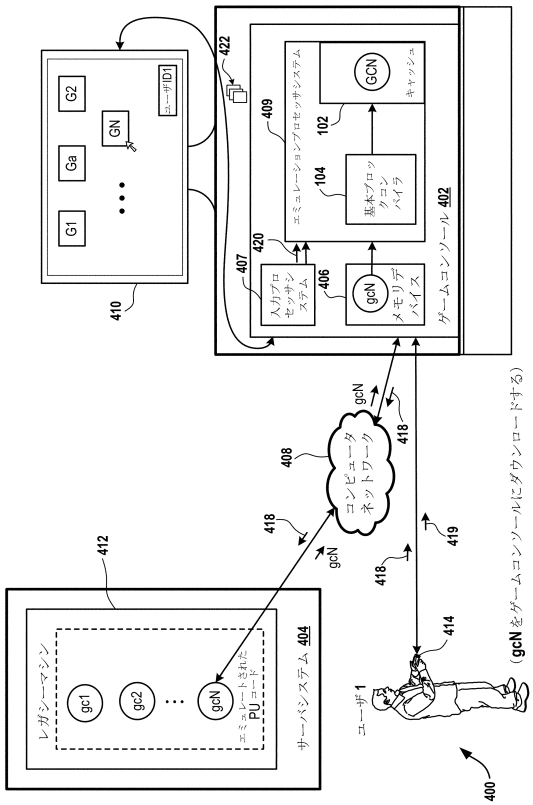
【図２】



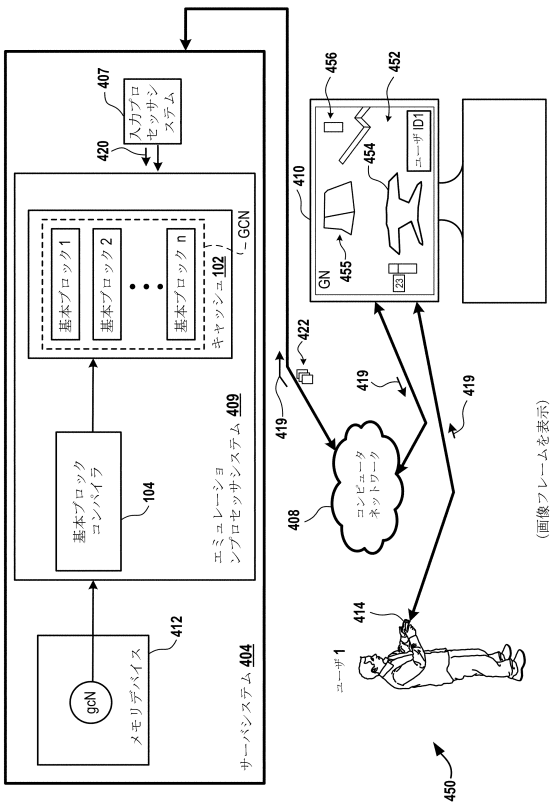
【図３】



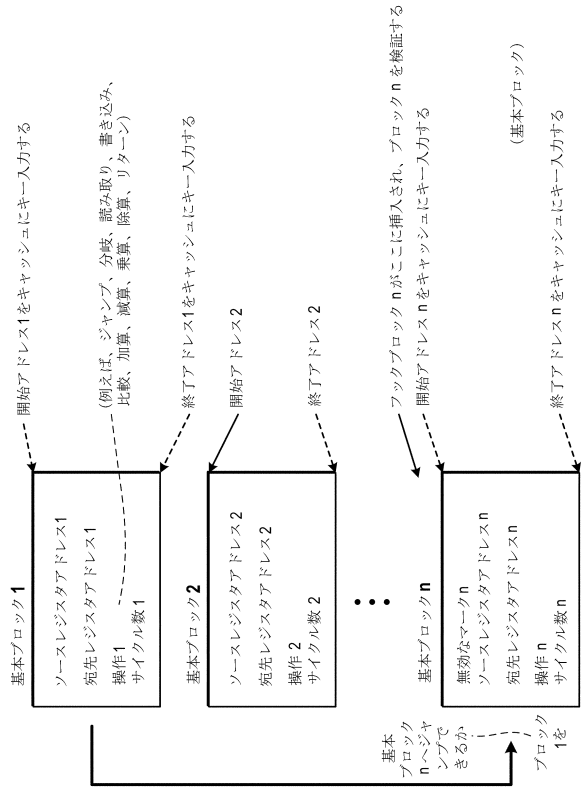
【図４Ａ】



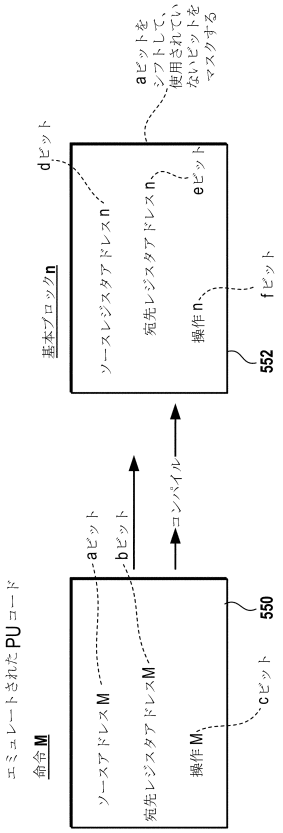
【図４Ｂ】



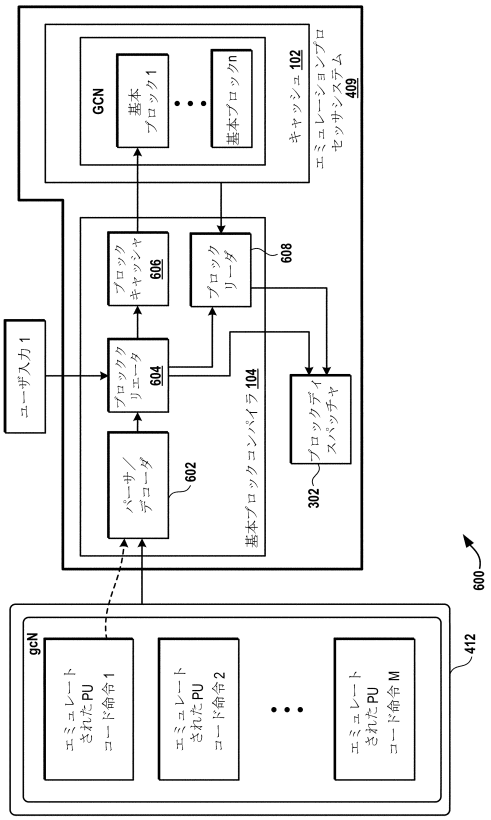
【図５Ａ】



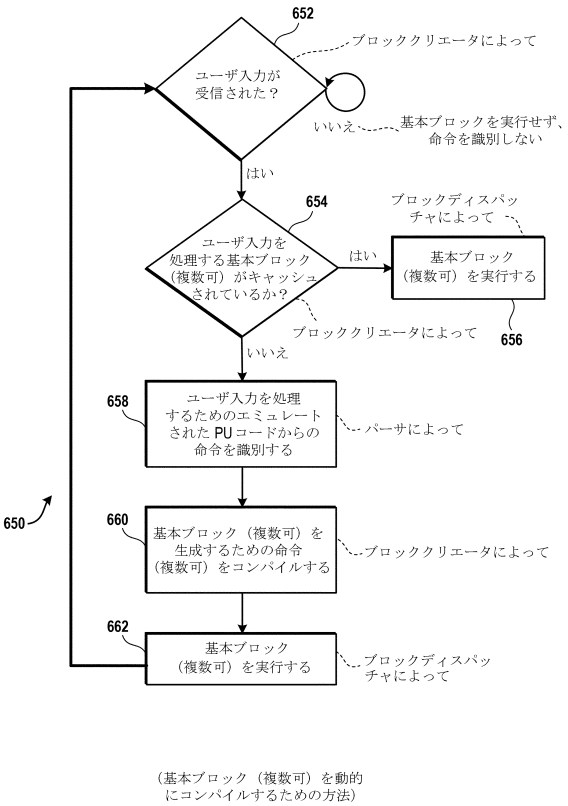
【図５Ｂ】



【図６Ａ】

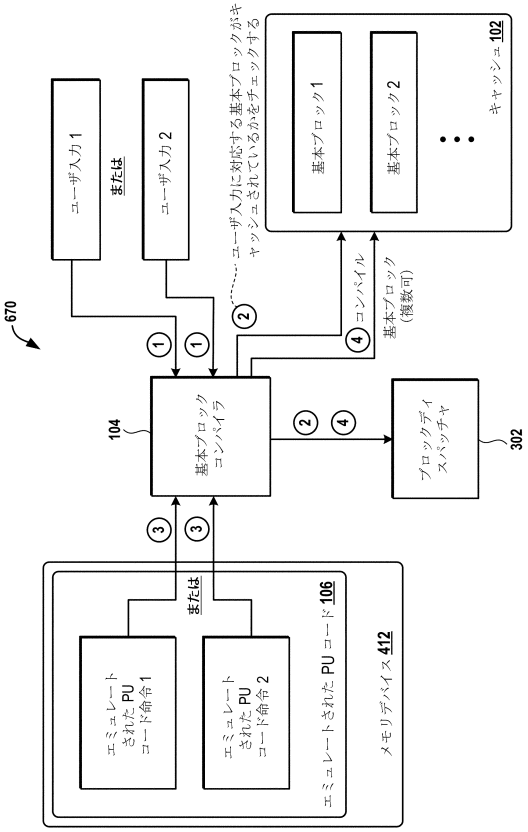


【図６Ｂ】

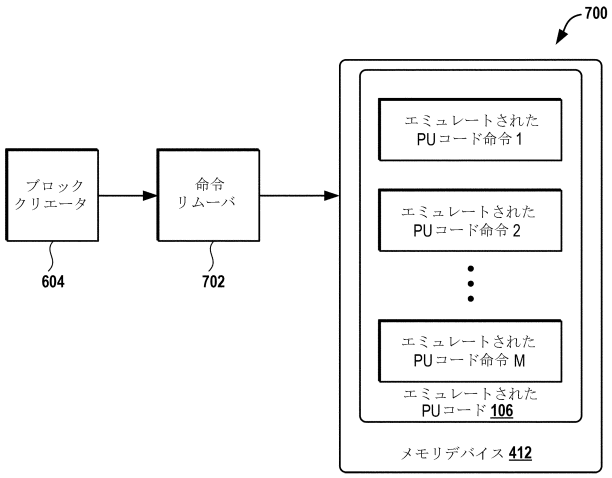


（基本ブロック（複数可）を動的にコンパイルするための方法）

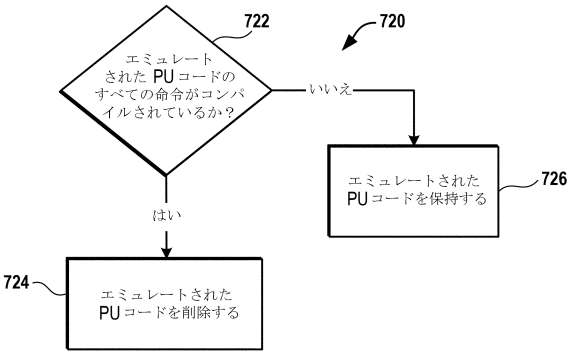
【図６Ｃ】



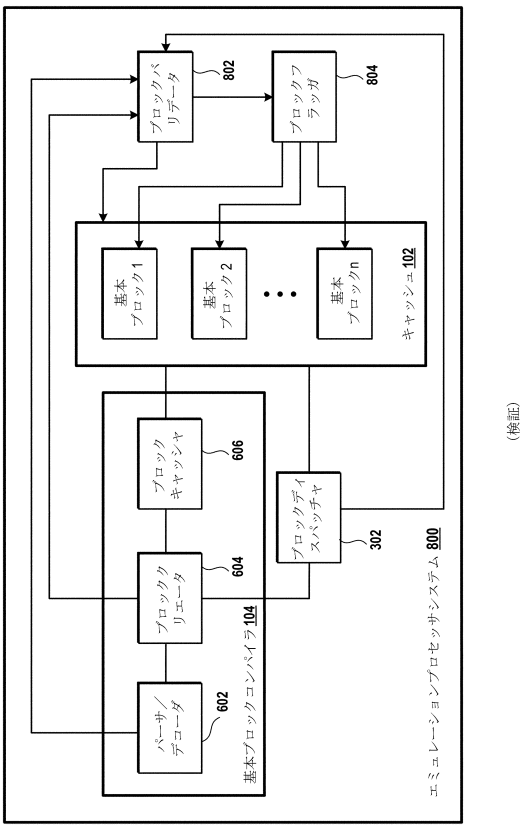
【図７Ａ】



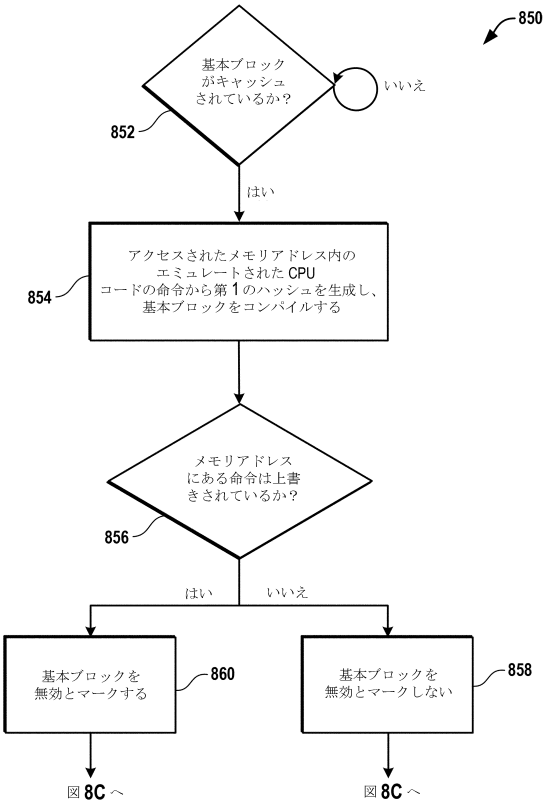
【図７Ｂ】



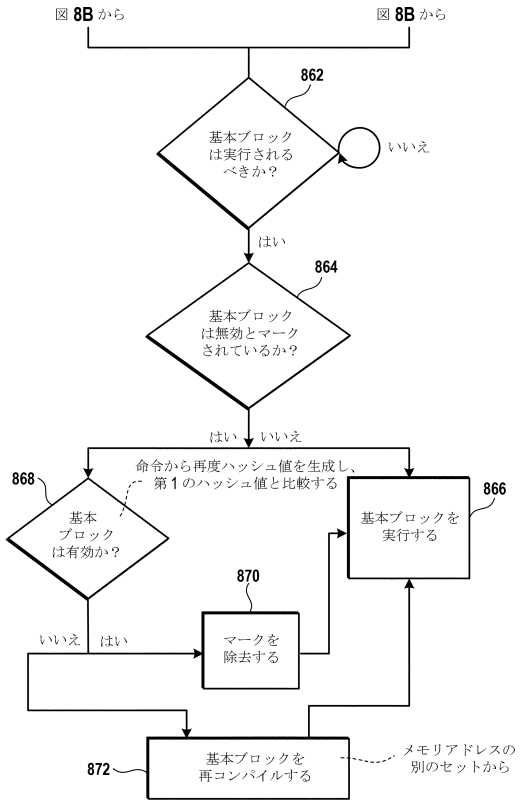
【図８Ａ】



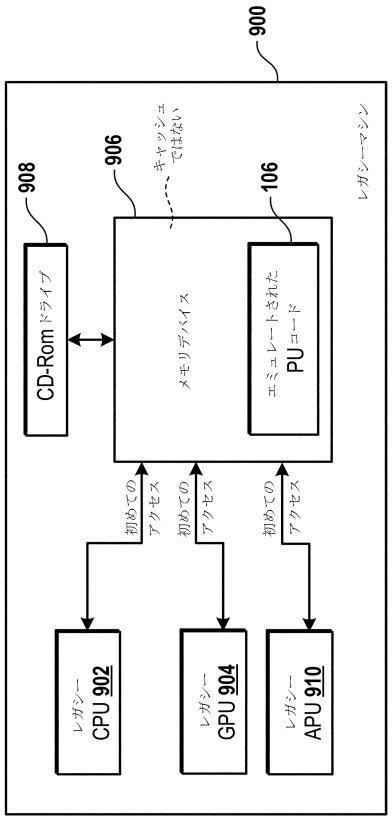
【図８Ｂ】



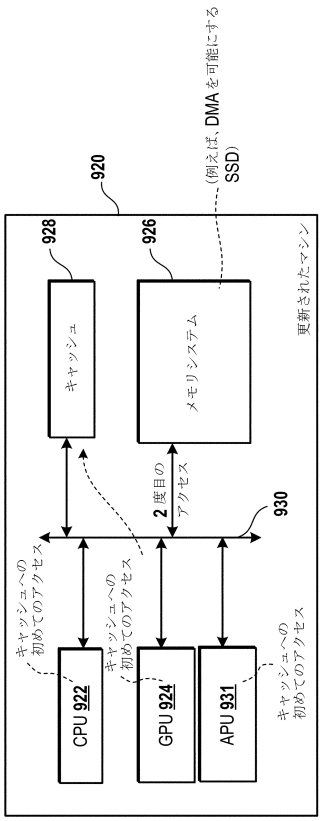
【図８Ｃ】



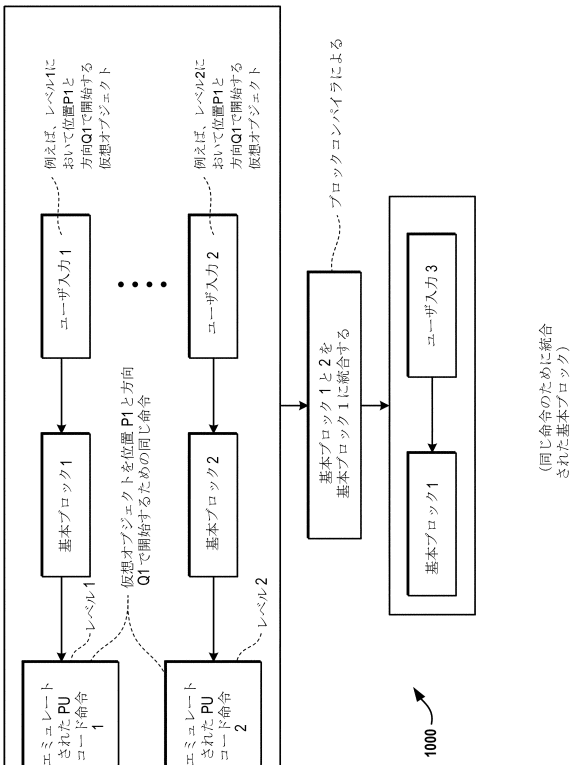
【図９Ａ】



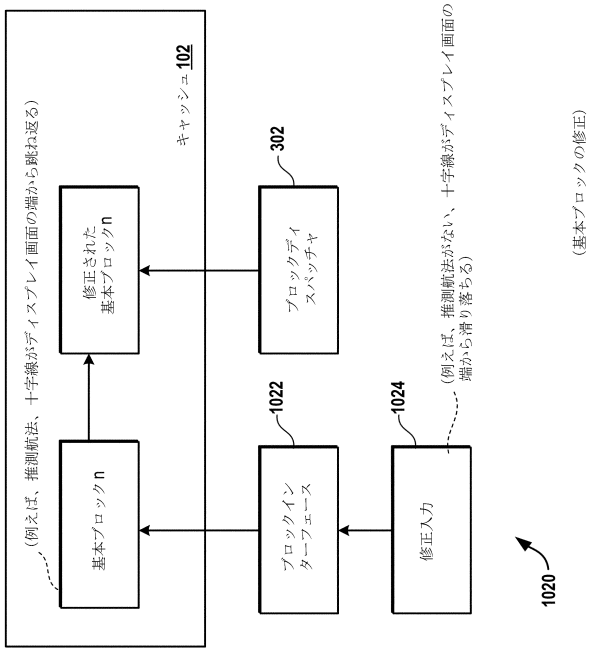
【図９Ｂ】



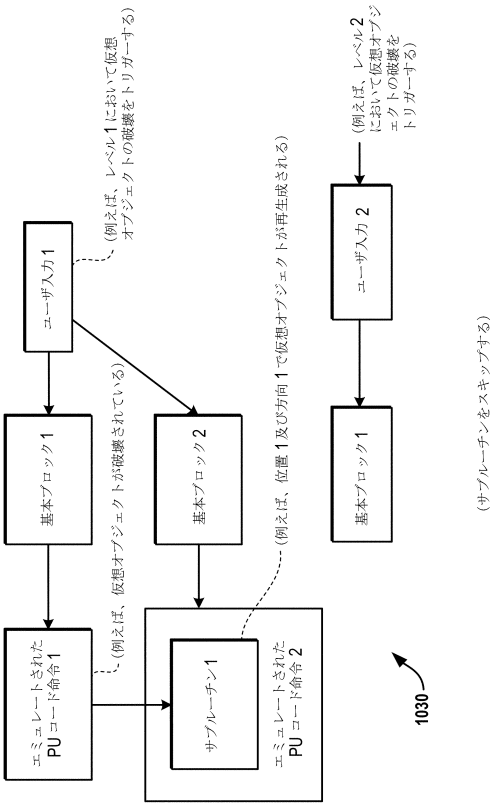
【図１０Ａ】



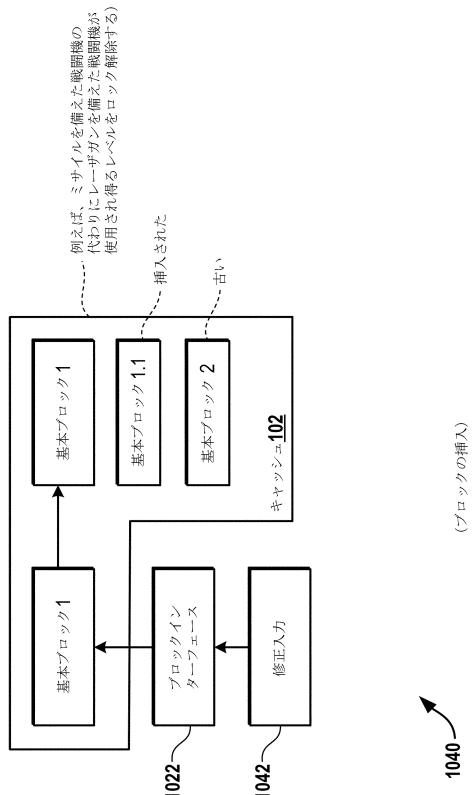
【図１０Ｂ】



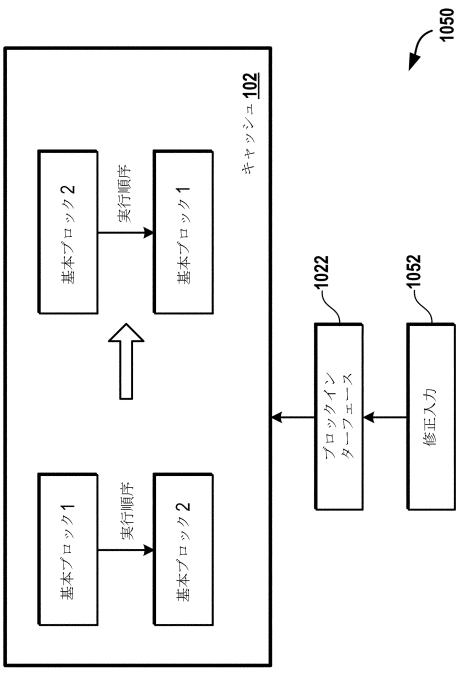
【図１０Ｃ】



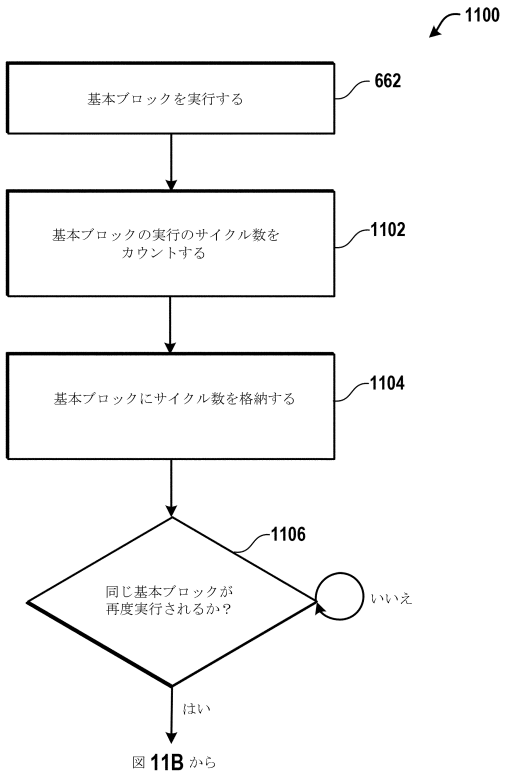
【図１０Ｄ】



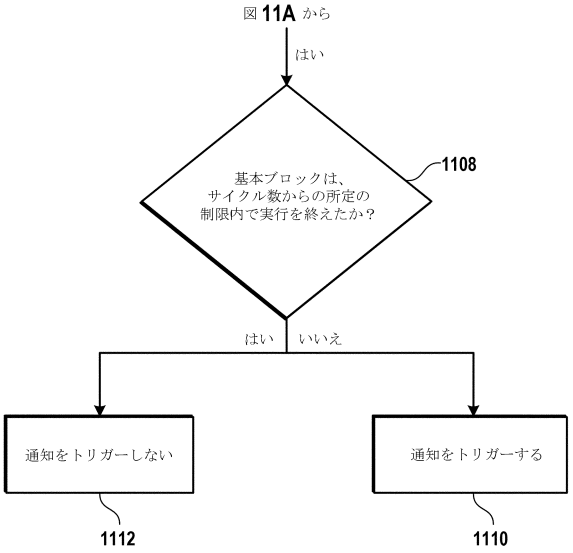
【図１０Ｅ】



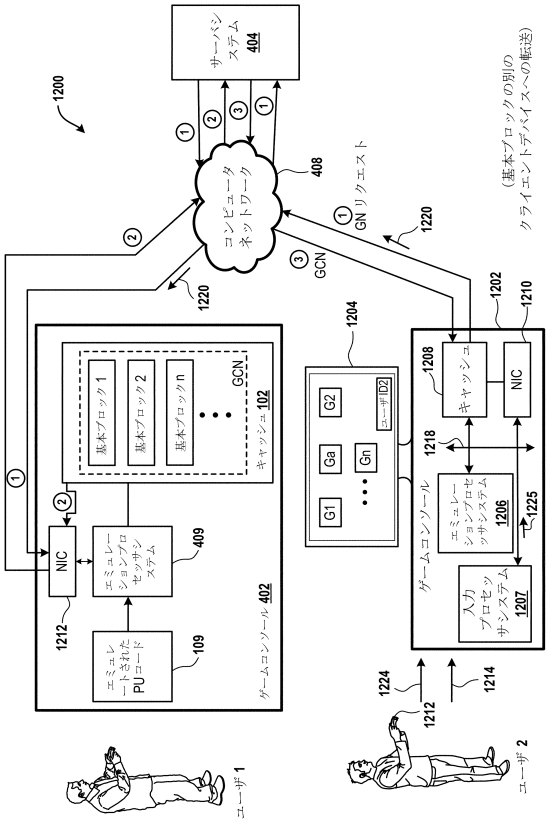
【図１１Ａ】



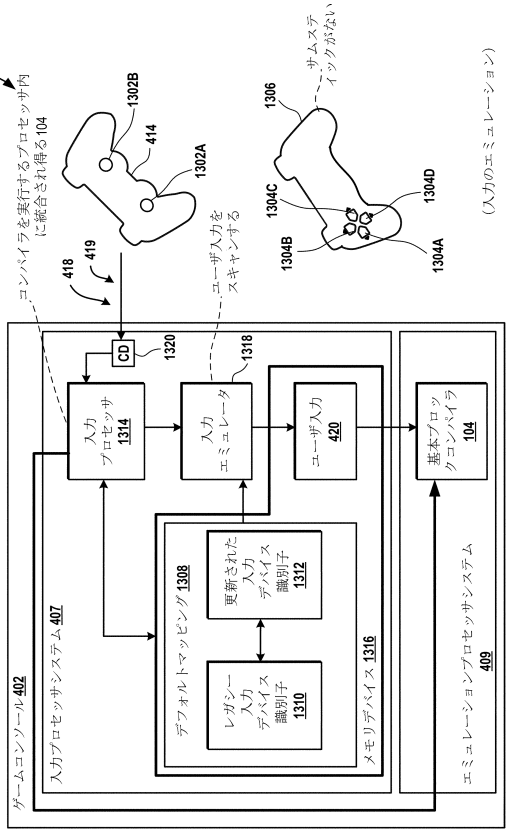
【図１１Ｂ】



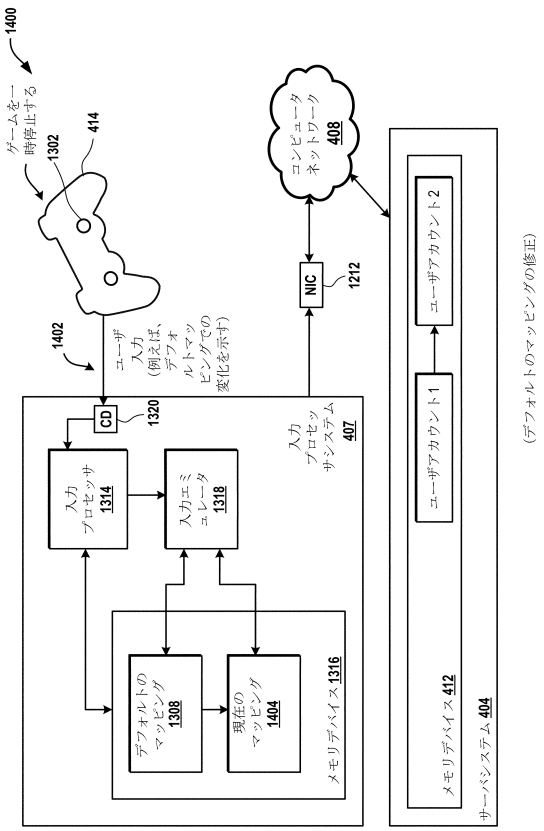
【図１２】



【図１３】

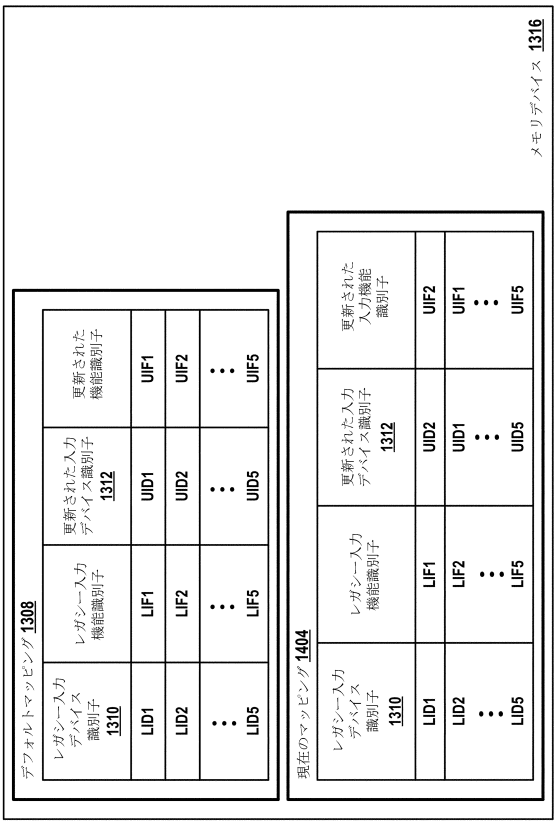


【図１４Ａ－１】

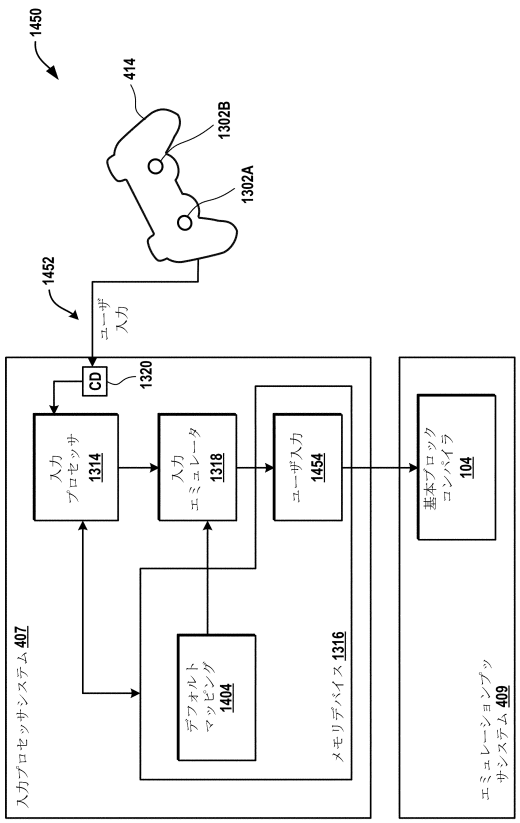


（デフォルトのマッピングの修正）

【図１４Ａ－２】

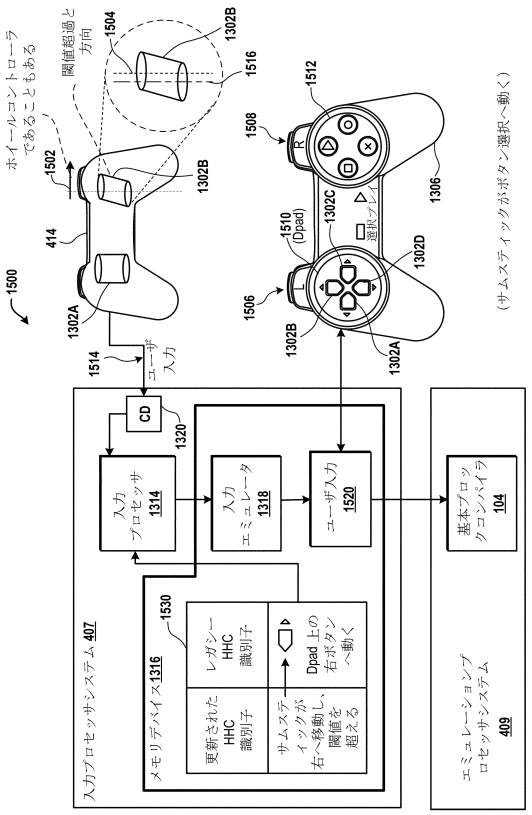


【図１４Ｂ】



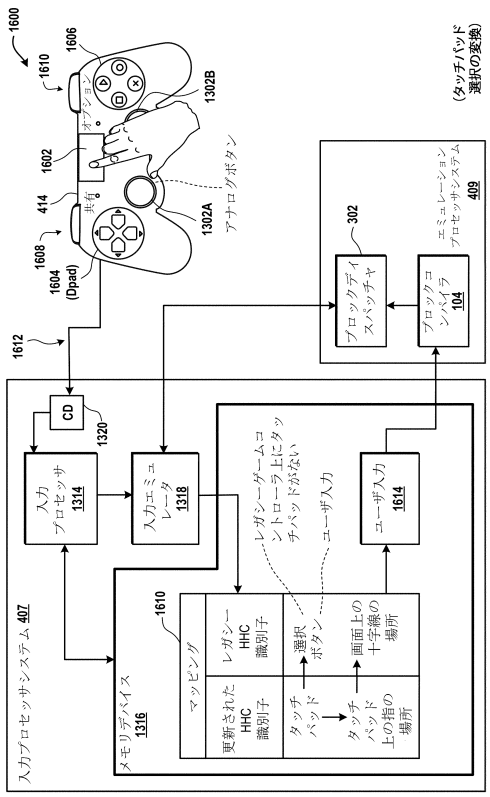
（現在のマッピングの使用）

【図１５】

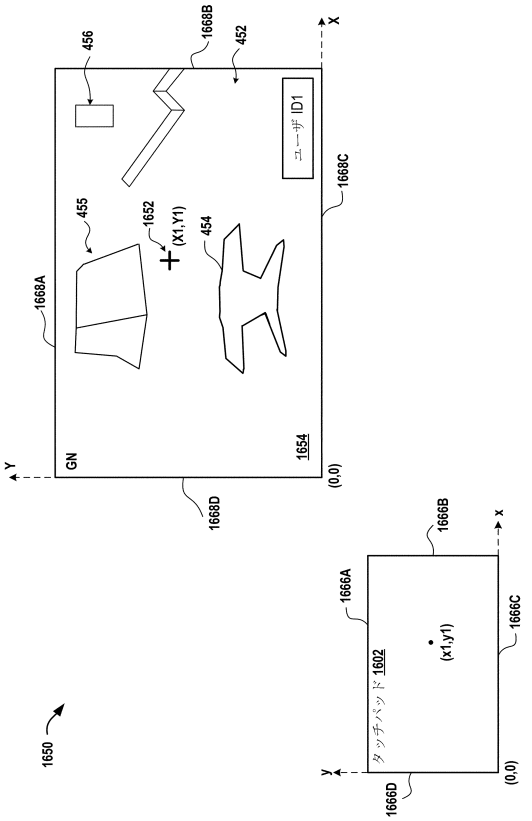


（サムスティックがボタン選択へ動く）

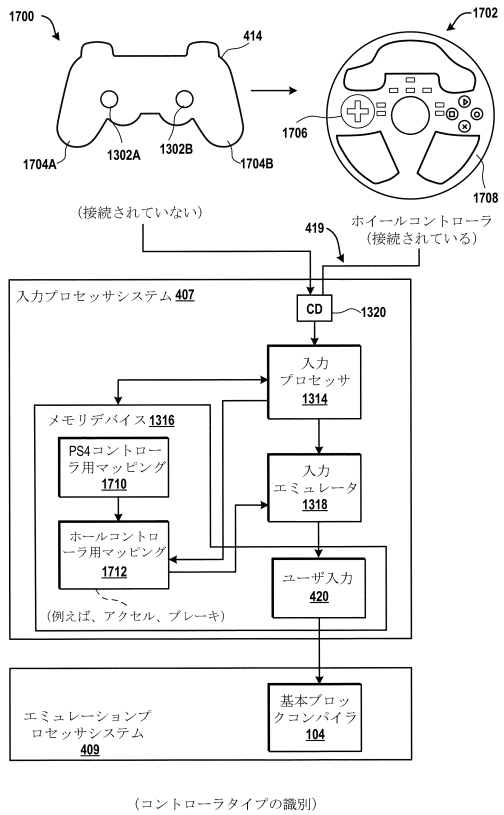
【図１６Ａ】



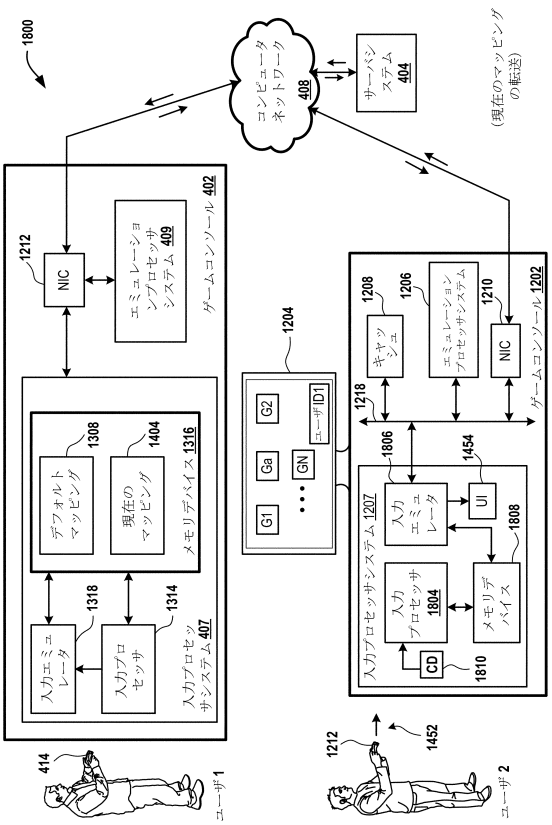
【図１６Ｂ】



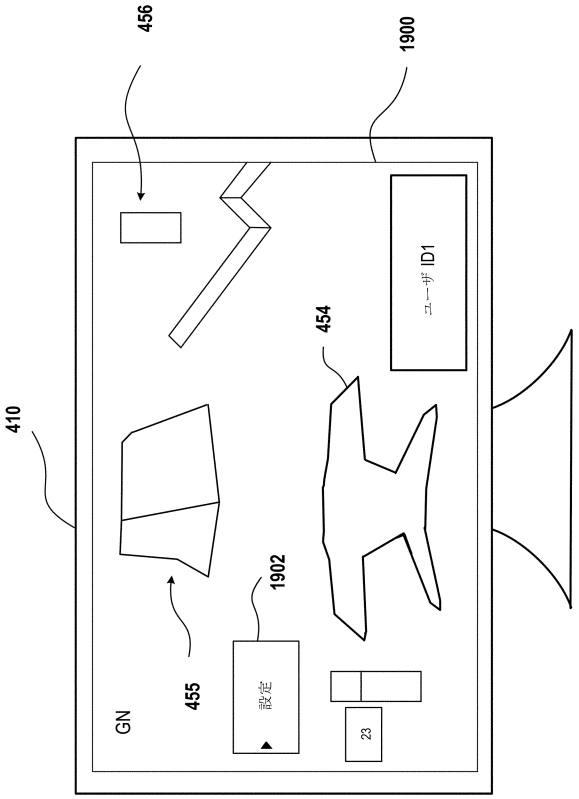
【図１７】



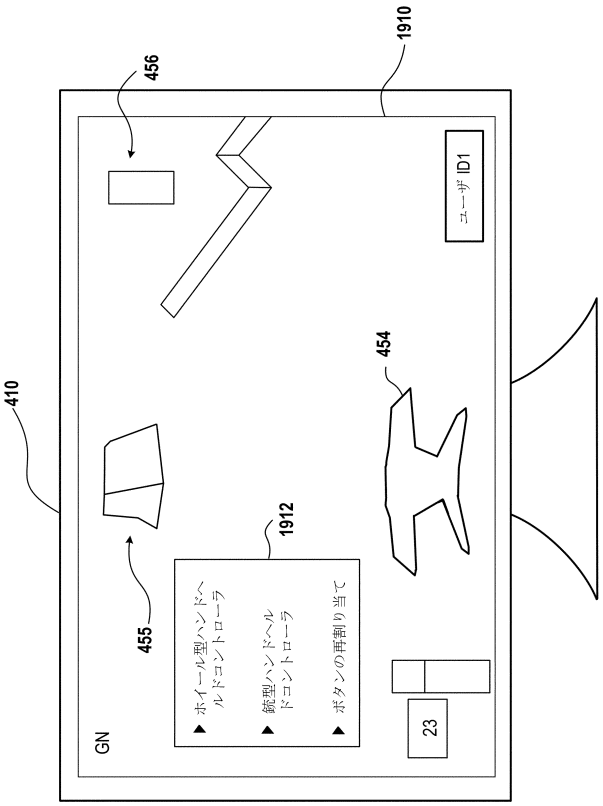
【図１８】



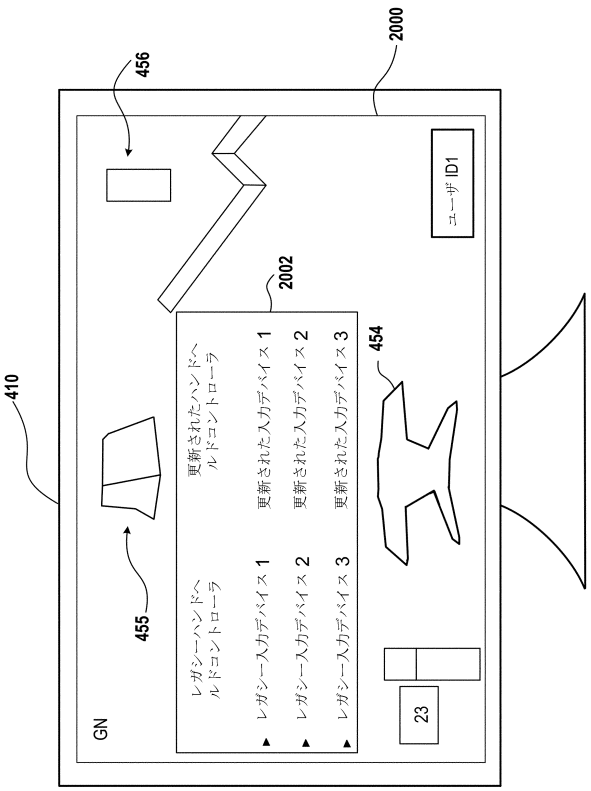
【図１９Ａ】



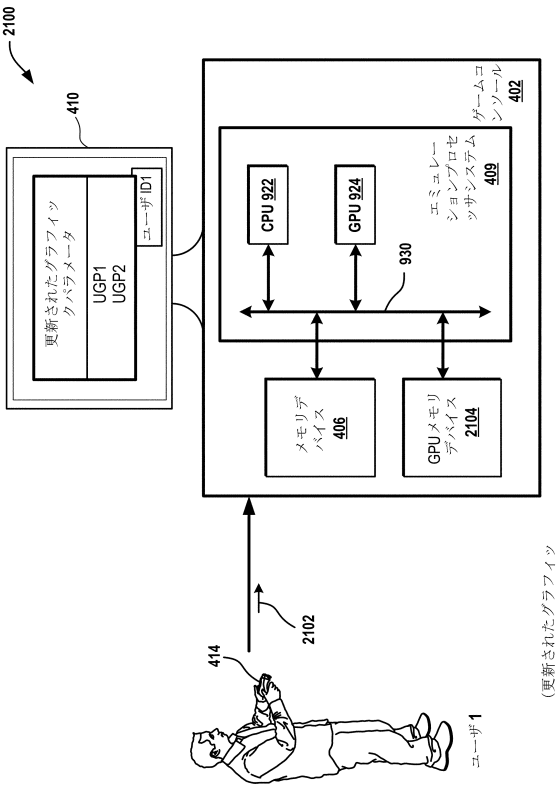
【図１９Ｂ】



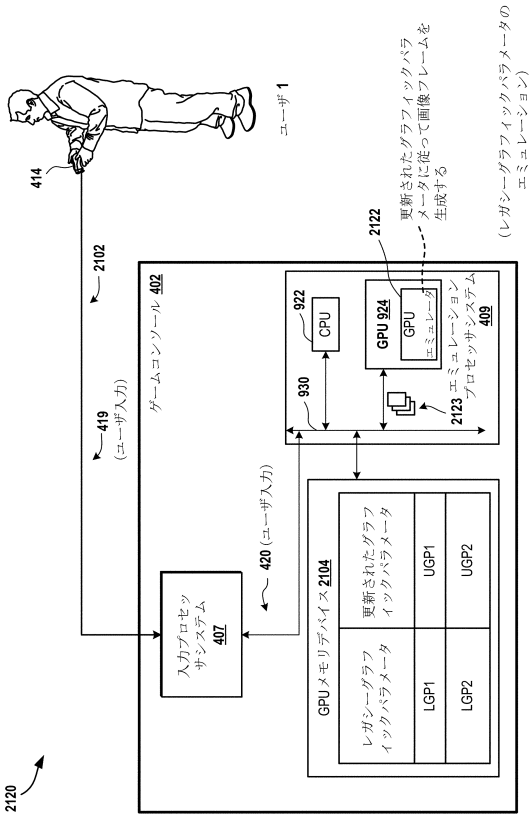
【図２０】



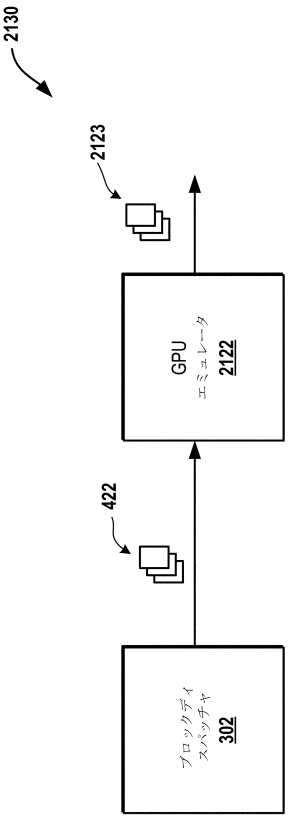
【図２１Ａ】



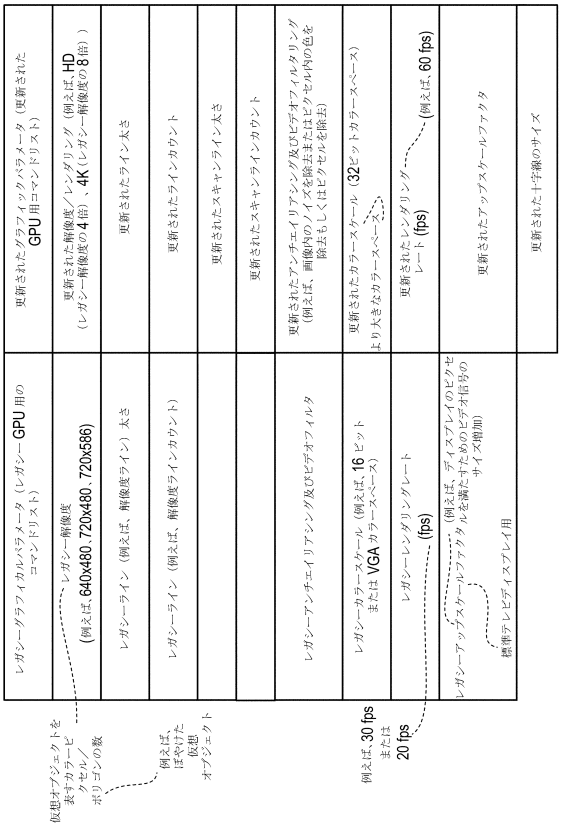
【図２１Ｂ－１】



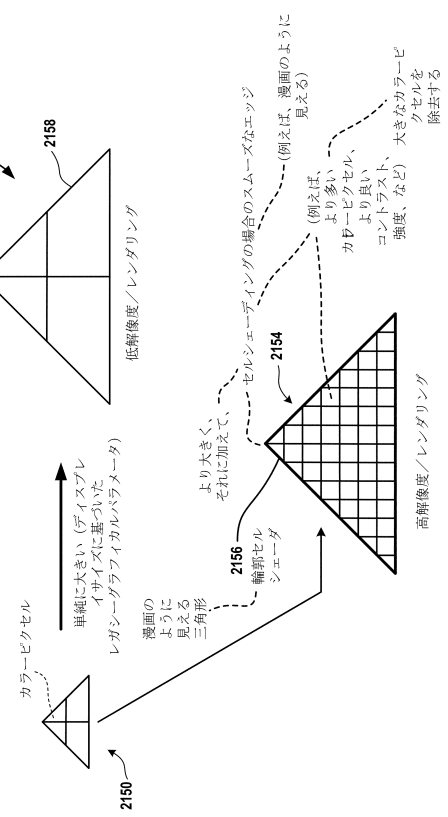
【図２１Ｂ－２】



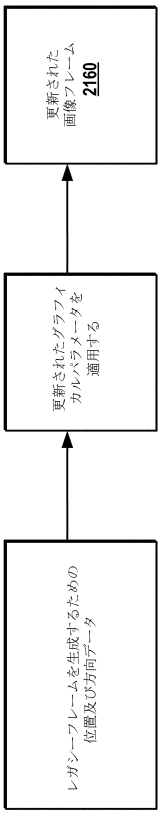
【図２１Ｃ－１】



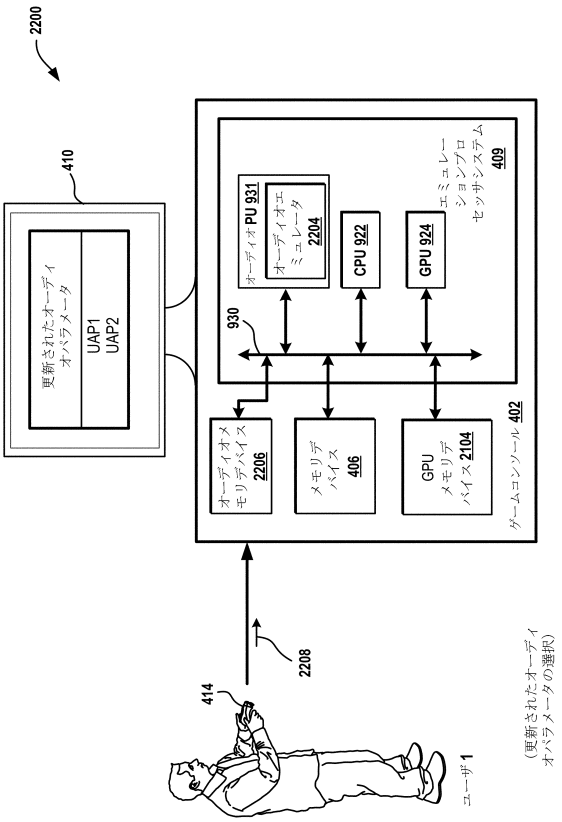
【図２１Ｃ－２】



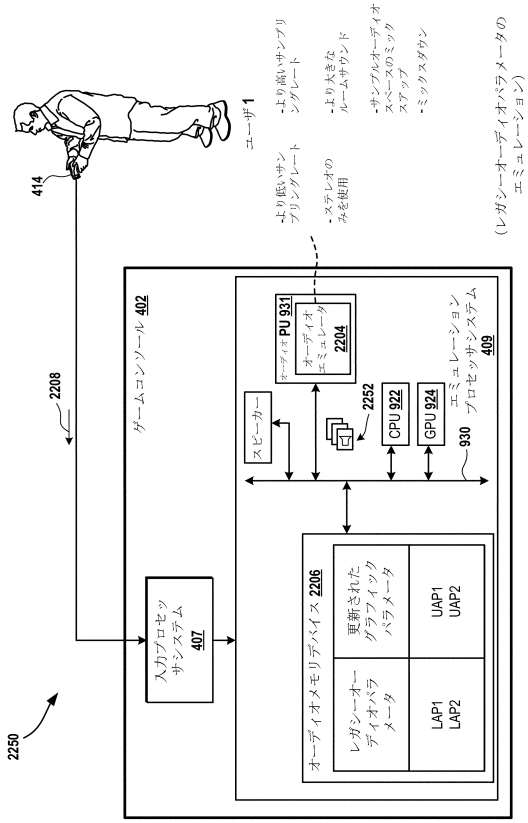
【図２１Ｄ】



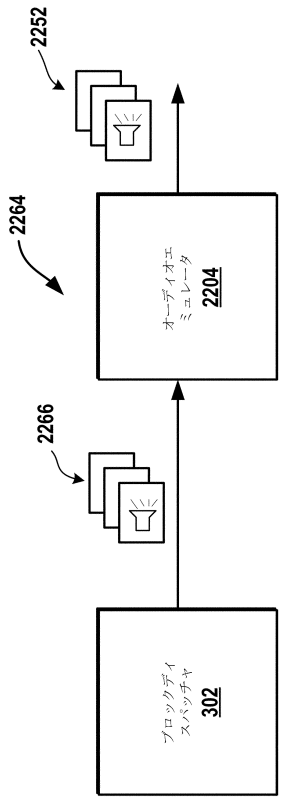
【図２２Ａ】



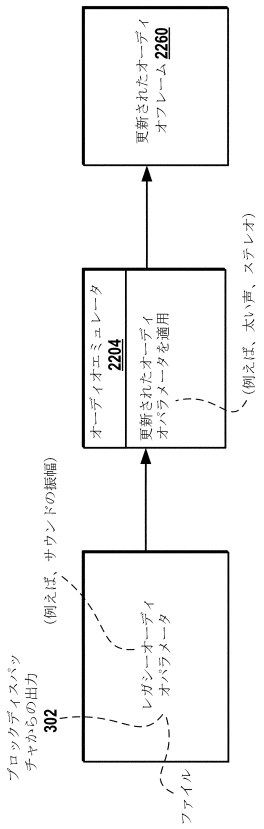
【図２２Ｂ－１】



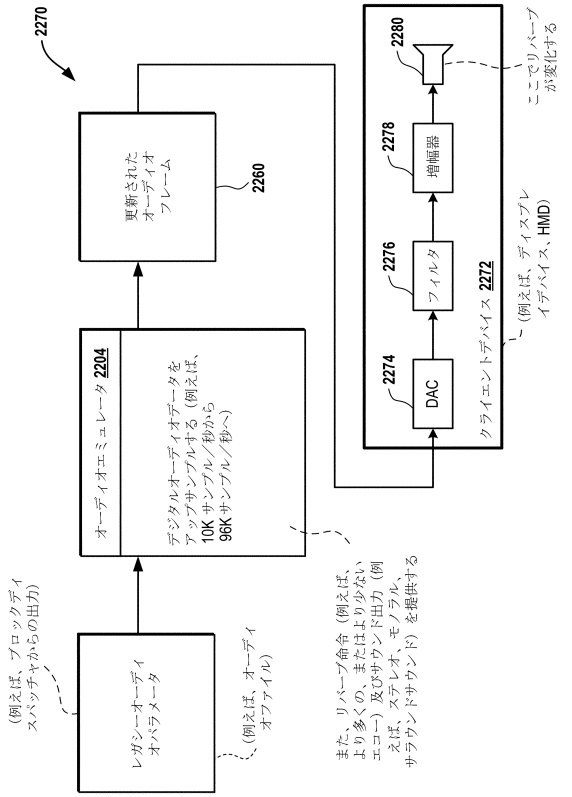
【図２２Ｂ－２】



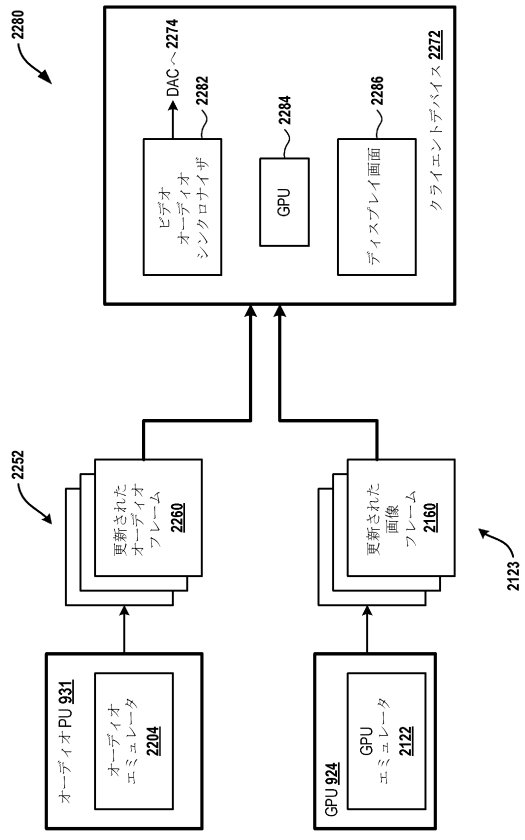
【図２２Ｃ－１】



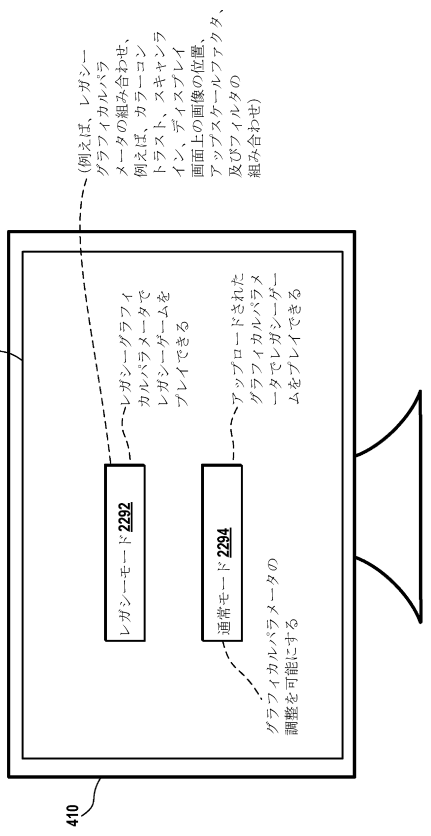
【図２２Ｃ－２】



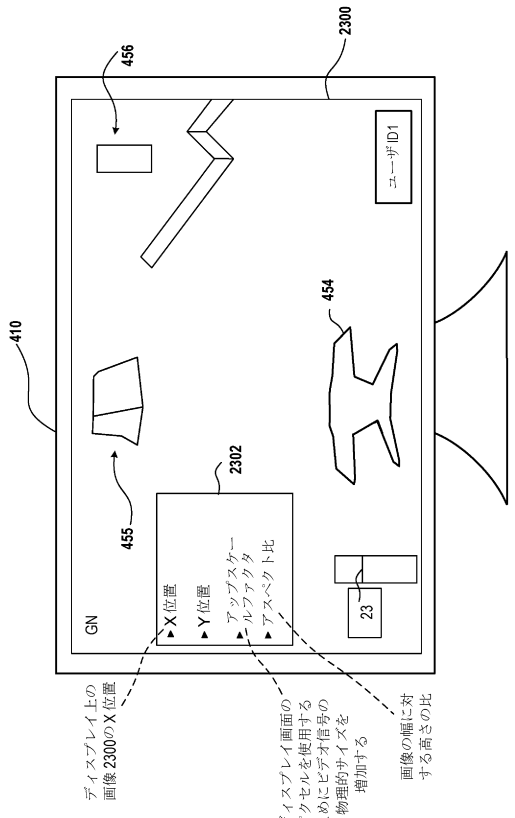
【図２２Ｃ－３】



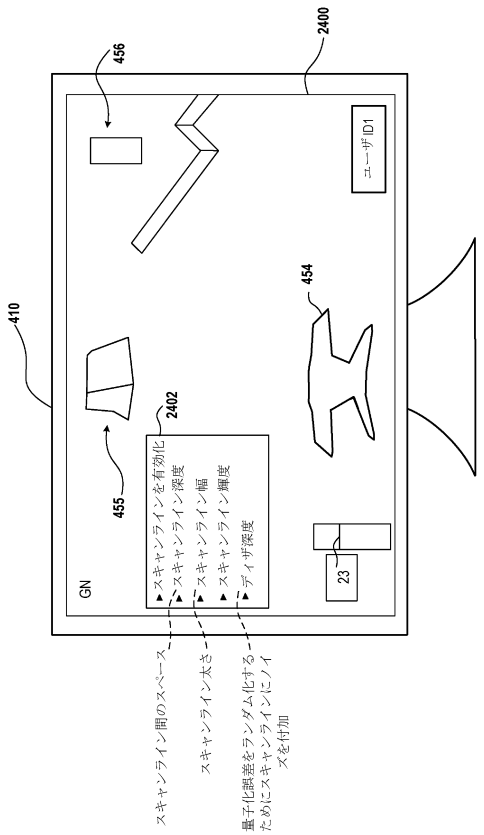
【図２２Ｄ】



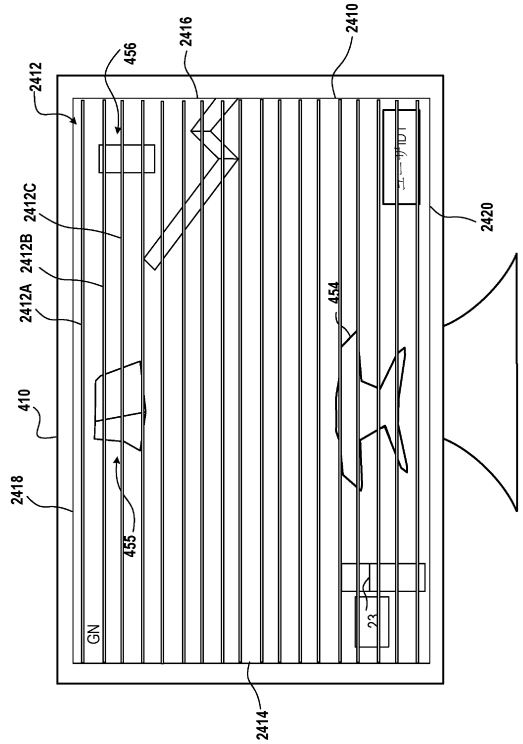
【図２３】



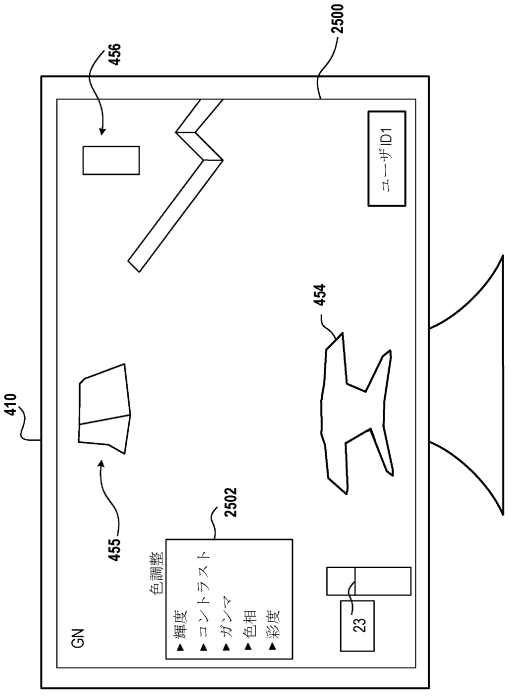
【図２４Ａ】



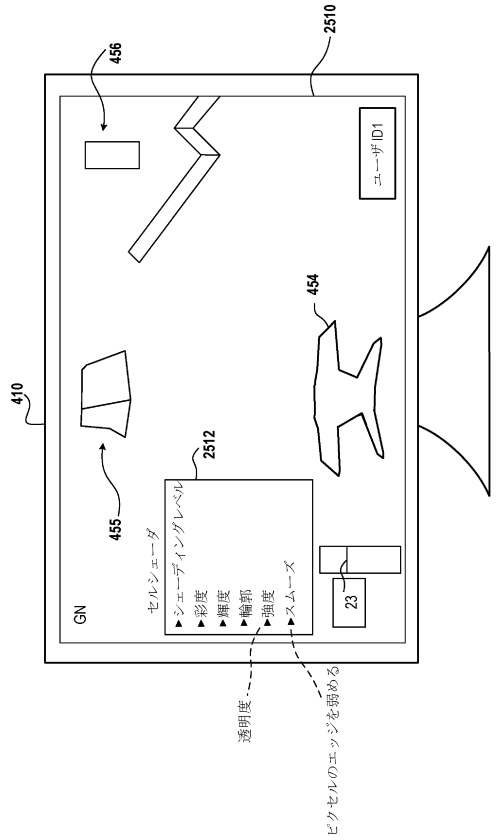
【図２４Ｂ】



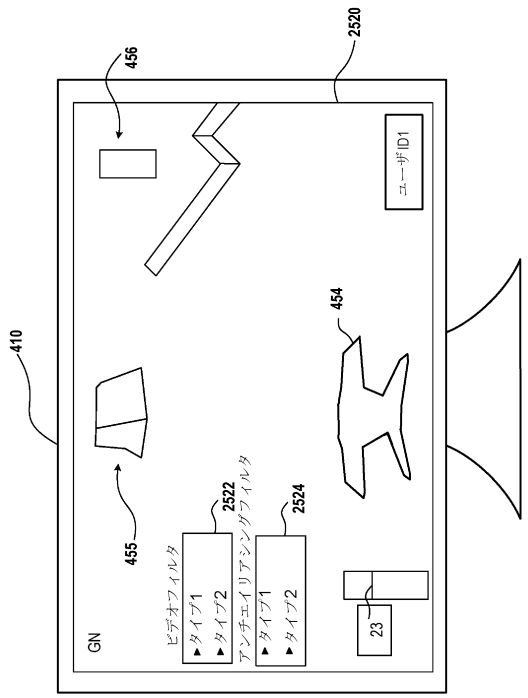
【図２５Ａ】



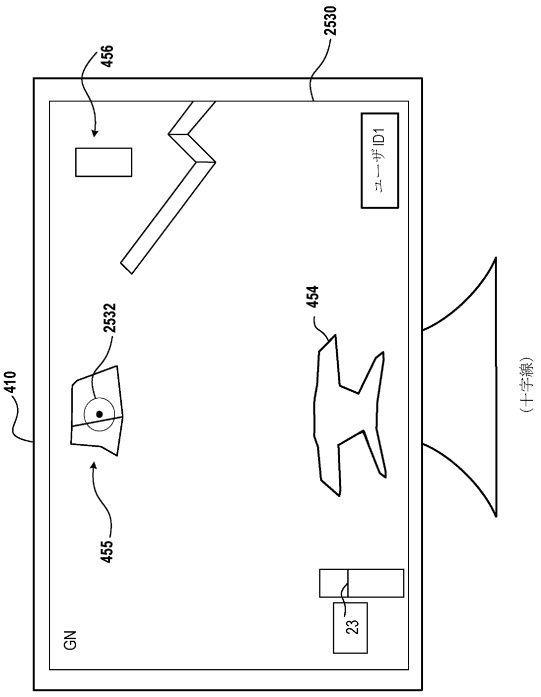
【図２５Ｂ】



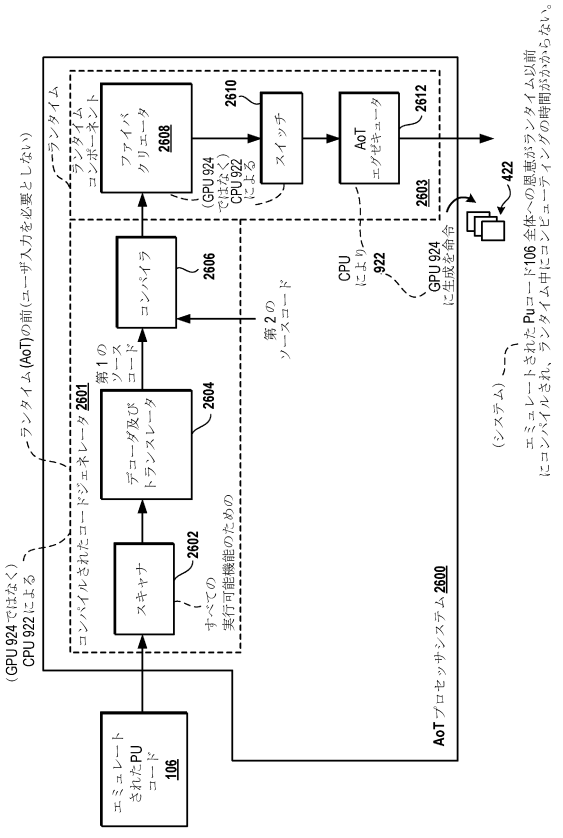
【図２５Ｃ】



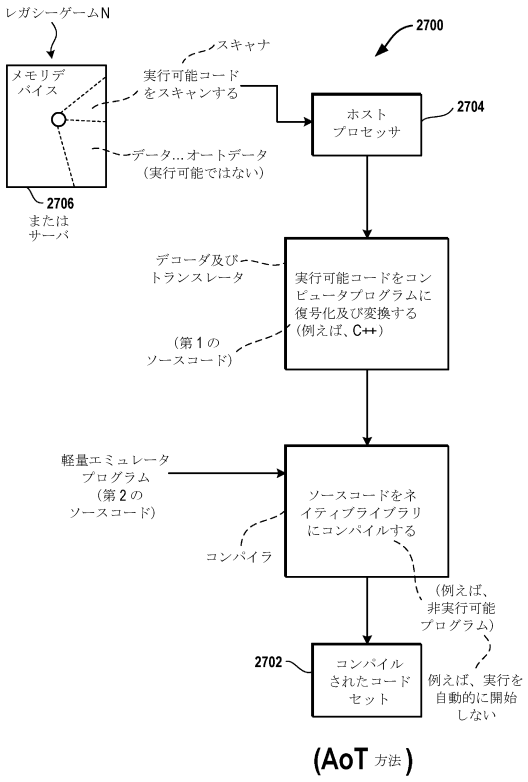
【図２５Ｄ】



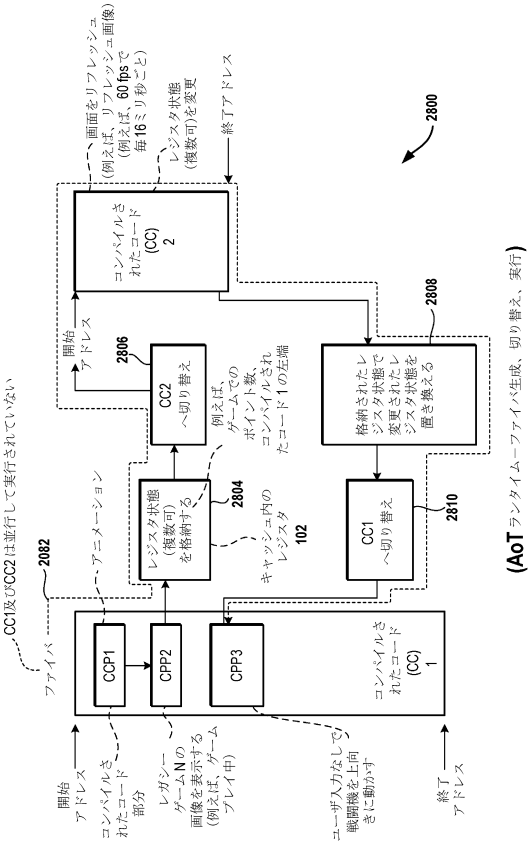
【図２６】



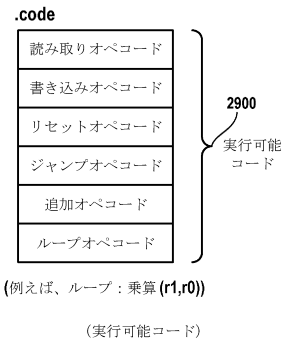
【図２７】



【図２８】



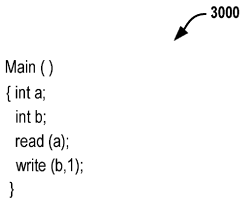
【図２９Ａ】



【図２９Ｂ】

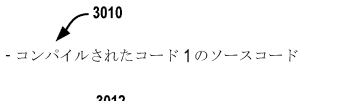


【図３０Ａ】



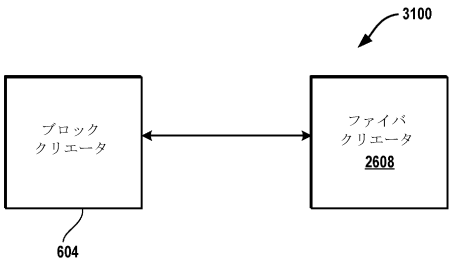
（ソースコード）

【図３０Ｂ】



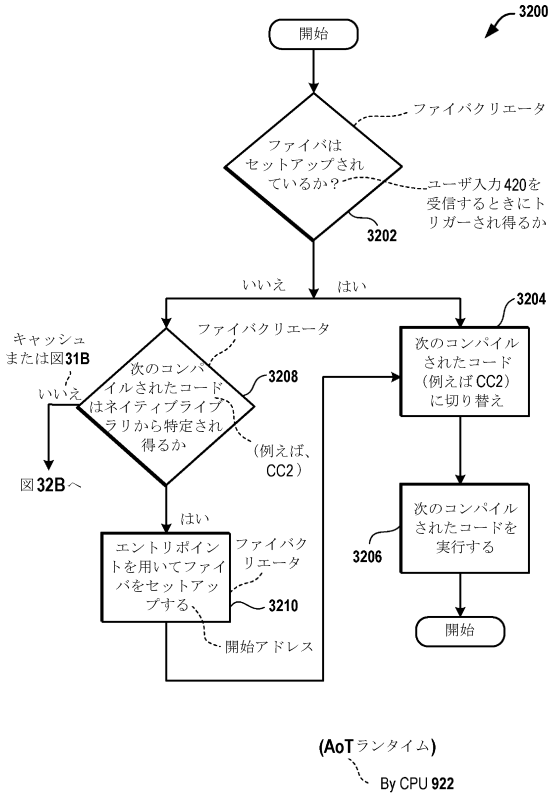
（追加されたソースコード）

【図３１】

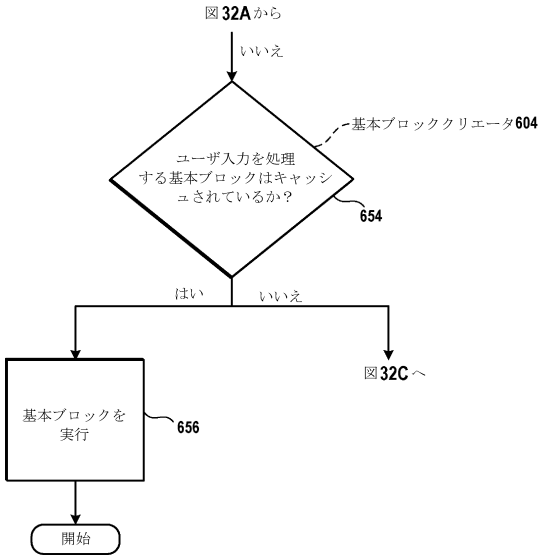


（システム）

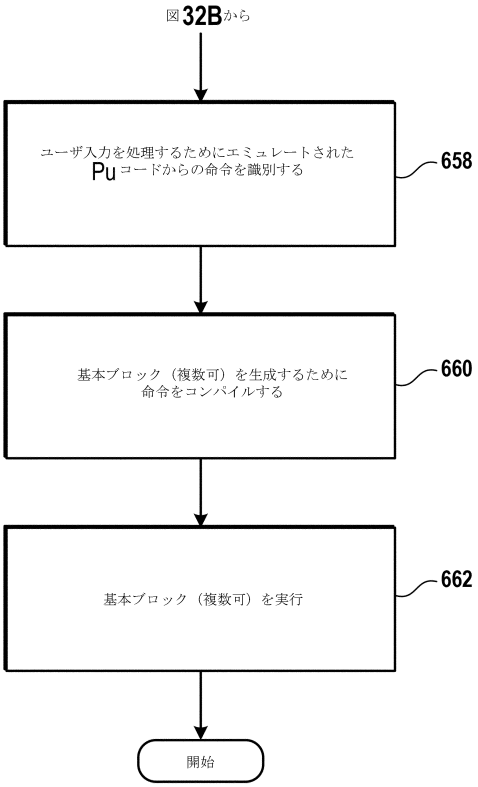
【図３２Ａ】



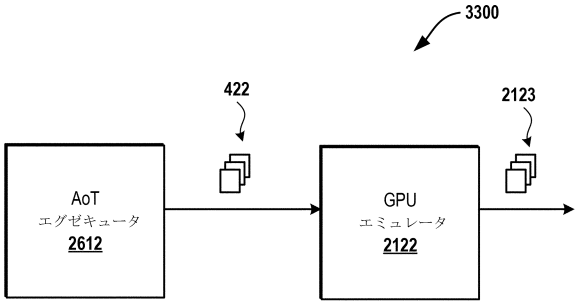
【図３２Ｂ】



【図３２Ｃ】

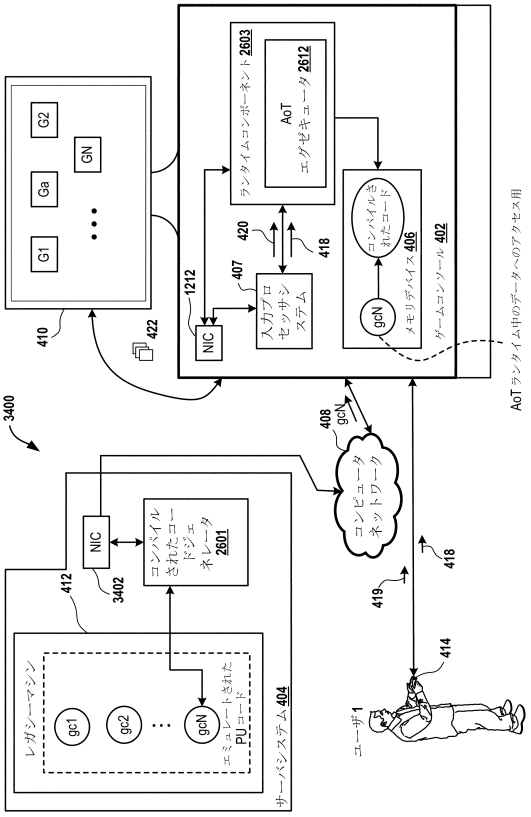


【図３３】

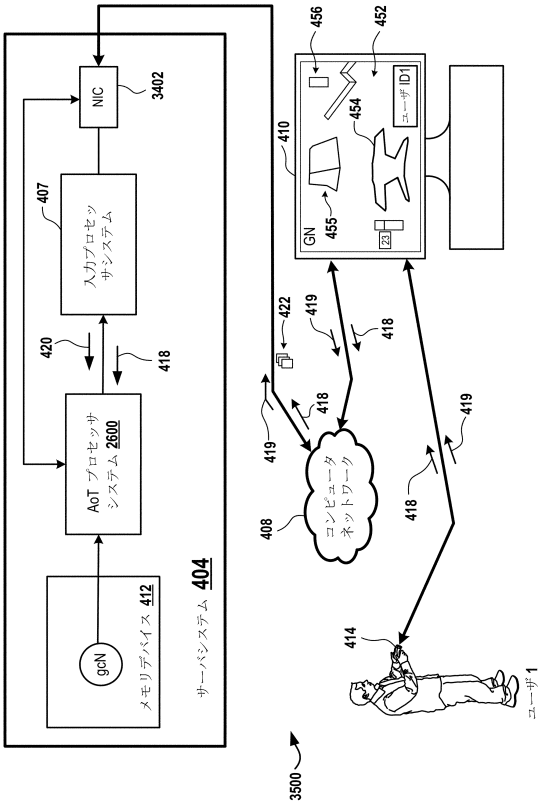


（システム）

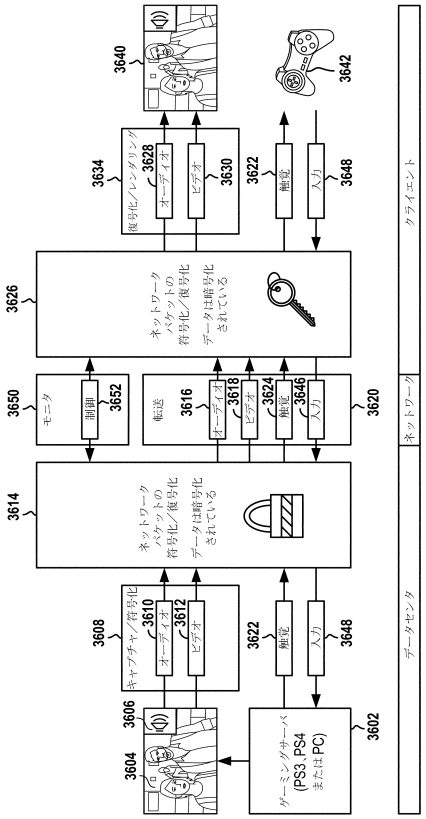
【図３４】



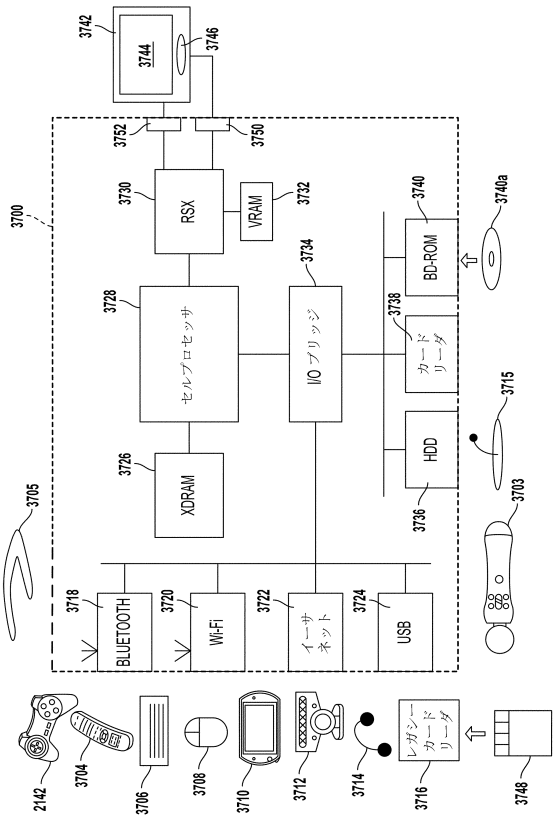
【図３５】



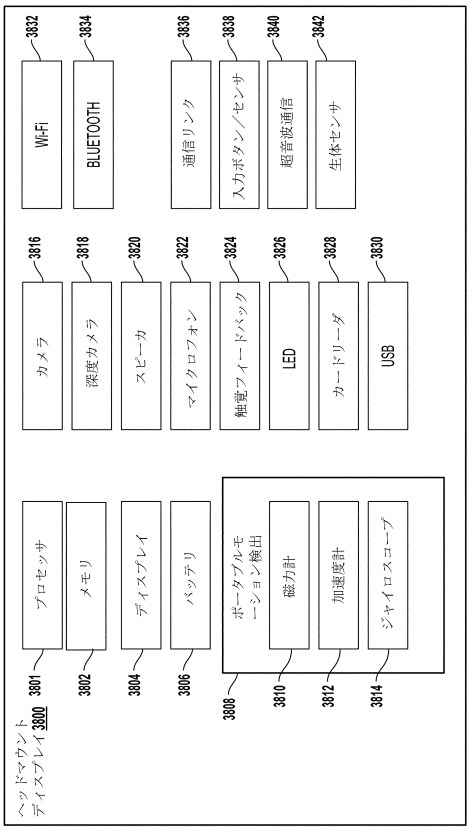
【図３６】



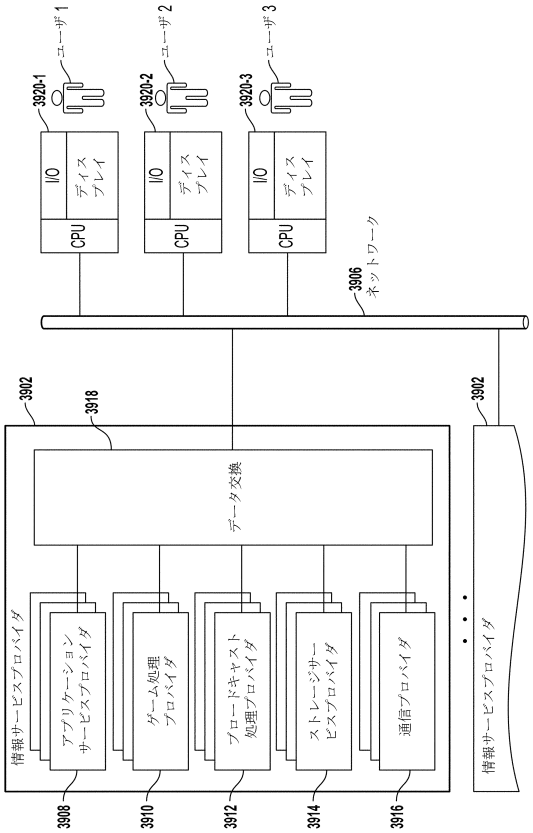
【図３７】



【図３８】



【図３９】


ユーザ 1
ユーザ 2
ユーザ 3
3920-1
3920-2
3920-3
I/O
ディスプレイ
CPU
3906
ネットワーク
3902
3918
データ交換
情報サービスプロバイダ
アプリケーションサービスプロバイダ
ゲーム処理プロバイダ
ブロードキャスト処理プロバイダ
ストレージサービスプロバイダ
通信プロバイダ
情報サービスプロバイダ
3908
3910
3912
3914
3916

フロントページの続き

ント パークウェイ ２２０７
(72)発明者 デーヴィッド タック
アメリカ合衆国、カリフォルニア州 ９４４０４、サン マテオ、ブリッジポイント パークウェイ ２２０７

Ｆターム（参考） 5B081 CC41
5B376 AA14 BC57