(19)日本国特許庁(JP)　(12)特　許　公　報(B2)　(11)特許番号

特許第5542120号
(P5542120)

(45)発行日　平成26年7月9日(2014.7.9)　(24)登録日　平成26年5月16日(2014.5.16)

(51)Int.Cl.		F I		
G06F 13/16	(2006.01)	G06F	13/16	510D
G06F 13/36	(2006.01)	G06F	13/36	530B
G06F 13/38	(2006.01)	G06F	13/38	340A
G06F 12/00	(2006.01)	G06F	12/00	580

請求項の数 21　(全 23 頁)

(21)出願番号	特願2011-502109 (P2011-502109)
(86)(22)出願日	平成21年3月27日 (2009.3.27)
(65)公表番号	特表2011-516950 (P2011-516950A)
(43)公表日	平成23年5月26日 (2011.5.26)
(86)国際出願番号	PCT/US2009/038637
(87)国際公開番号	WO2009/123933
(87)国際公開日	平成21年10月8日 (2009.10.8)
審査請求日	平成24年3月27日 (2012.3.27)
(31)優先権主張番号	61/041,210
(32)優先日	平成20年3月31日 (2008.3.31)
(33)優先権主張国	米国(US)

(73)特許権者　510258658
アドバンスト・マイクロ・ディバイシズ・インコーポレイテッド
アメリカ合衆国、９４０８８　カリフォルニア州，サニィベイル，ワン　エイエムディ　プレイス（番地なし）
(74)代理人　100078916
弁理士　鈴木　由充
(74)代理人　100142114
弁理士　小石川　由紀乃
(72)発明者　モレイン，ステファン
アメリカ合衆国、９４０８８　カリフォルニア州，サニィベイル，ワン　エイエムディ　プレイス（番地なし）

最終頁に続く

(54)【発明の名称】ピアツーピア専用プロセッサのアーキテクチャ及び方法

(57)【特許請求の範囲】

【請求項１】

各々が少なくとも１つのローカルメモリを備える複数の専用プロセッサと、

前記複数の専用プロセッサのそれぞれを、当該複数の専用プロセッサのうちの少なくとも別の１つにブリッジを介して結合するホストブリッジバスと、

前記複数の専用プロセッサのそれぞれを、当該複数の専用プロセッサのうちの少なくとも１つに直接結合するダイレクトバスと、

少なくとも１つのシステムメモリと、

前記複数の専用プロセッサのそれぞれに結合されたメモリコントローラとを備え、

各専用プロセッサのローカルメモリは、所属する専用プロセッサにダイレクトバスを介して結合されている他の専用プロセッサの中の少なくとも１つに対するピアメモリとして機能し、

前記メモリコントローラは、各専用プロセッサがホストブリッジバスまたはダイレクトバスのいずれを介してデータを転送するかを決定すると共に、転送されるデータの転送先のアドレスがいずれのメモリのアドレス範囲に入るかに基づき、当該データを転送する専用プロセッサがいずれのメモリにアクセスするかを決定し、特に、前記転送先のアドレスが当該データを転送する専用プロセッサのピアメモリの範囲に入るときには、ホストブリッジバスおよびダイレクトバスのそれぞれにおけるトラフィック量に基づき、当該ピアメモリへのデータの転送にいずれのバスを使用するかを決定するように構成されている、システム。

【請求項２】
前記メモリコントローラは、複数の専用プロセッサの少なくとも１つの上に存在する、請求項１に記載のシステム。
【請求項３】
前記複数の専用プロセッサはグラフィックス処理装置（ＧＰＵ）を含む、請求項１に記載のシステム。
【請求項４】
前記ホストブリッジバスは、ＰＣＩ　Ｅｘｐｒｅｓｓ（ＰＣＩｅ（登録商標））バスを含む、請求項１に記載のシステム。
【請求項５】
前記ダイレクトバスは、ＰＣＩ　Ｅｘｐｒｅｓｓ（ＰＣＩｅ（登録商標））バスを含む、請求項１に記載のシステム。
【請求項６】
前記ダイレクトバスは、プロプライエタリバスを含む、請求項１に記載のシステム。
【請求項７】
前記メモリコントローラは、さらに、前記転送先のアドレスが前記データを転送する専用プロセッサのピアメモリの範囲に入るときには、ホストブリッジバス及びダイレクトバスのいずれかが現在満杯か否かに基づき、いずれのバスを使用するかを決定する、請求項１に記載のシステム。
【請求項８】
前記メモリコントローラは、さらに、前記転送先のアドレスが前記データを転送する専用プロセッサのピアメモリの範囲に入るときには、ホストブリッジバスおよびダイレクトバスのいずれかに対する固定マッピングに基づき、いずれのバスを使用するかを決定する、請求項１に記載のシステム。
【請求項９】
前記メモリコントローラは、さらに、ホストブリッジバス及びダイレクトバスのそれぞれにおけるトラフィック量に基づき、転送すべきデータの一部分がホストブリッジバスを介して転送され、前記データの他の部分がダイレクトバスを介して転送されるように、動的負荷分散を行なう、請求項１に記載のシステム。
【請求項１０】
各々が少なくとも１つのローカルメモリを備える複数の専用プロセッサおよびホストプロセッサにより共有されるホストブリッジバスと、前記複数の専用プロセッサのそれぞれを、当該複数の専用プロセッサのうちの少なくとも別の１つに直接結合するダイレクトバスとを備え、各ローカルメモリを、所属する専用プロセッサに前記ダイレクトバスを介して結合されている他の専用プロセッサの中の少なくとも１つに対するピアメモリとして機能させるシステムにおいて、各専用プロセッサが前記ホストブリッジバスまたはダイレクトバスのいずれを介してデータを転送又は受信するかを決定することを含み、データを転送する場合の当該決定が、転送されるデータの転送先のアドレスがいずれのメモリのアドレス範囲に入るかに基づき行われ、前記転送先のアドレスがデータを転送する専用プロセッサのピアメモリを特定する範囲内に含まれるときの当該決定が、ホストブリッジバスおよびダイレクトバスのそれぞれにおけるトラフィック量に基づき行われる、マルチプロセッサシステムにおけるデータ通信方法。
【請求項１１】
前記アドレス範囲は、前記データを転送する専用プロセッサに結合されたローカルメモリ、この専用プロセッサのピアメモリ、および前記ホストブリッジバス上のシステムメモリのうちの１つを特定する、請求項１０に記載の方法。
【請求項１２】
前記ホストブリッジバス又は前記ダイレクトバスのいずれかにピアメモリを特定する範囲内の特定アドレスが割り当てられる、請求項１１に記載の方法。
【請求項１３】

動的負荷分散を行なって、転送すべきデータの一部分がホストブリッジバスを介して転送され、前記データの他の部分がダイレクトバスを介して転送されるように、ホストブリッジバス及びダイレクトバスのそれぞれにおけるトラフィックの割合をトラフィック履歴データに基づき調整することを更に含む、請求項１１に記載の方法。

【請求項１４】

メモリアパーチャをホストプロセッサ及び複数の専用プロセッサのそれぞれに割り当てることを更に含み、メモリアパーチャはピアメモリ、システムメモリ、及びローカルメモリに適用され、アパーチャが各プロセッサのメモリの見方を決定する、請求項１１に記載の方法。

【請求項１５】

アパーチャ圧縮を行なうことを更に含み、同圧縮が、オフセットレジスタを用いて、オフセットを変更することにより全てのピアメモリを可視化することを含む、請求項１４に記載の方法。

【請求項１６】

各々が少なくとも１つのローカルメモリを備える複数の専用プロセッサおよびホストプロセッサにより共有されるホストブリッジバスと、前記複数の専用プロセッサのそれぞれを、当該複数の専用プロセッサのうちの少なくとも別の１つに直接結合するダイレクトバスとを備え、各ローカルメモリを、所属する専用プロセッサに前記ダイレクトバスを介して結合されている他の専用プロセッサの中の少なくとも１つに対するピアメモリとして機能させるマルチプロセッサシステムにおいて実行されることにより、前記マルチプロセッサシステムにおけるデータ通信方法を遂行させるインストラクションが記憶されたコンピュータ可読媒体であって、同方法が、

各専用プロセッサが前記ホストブリッジバスまたはダイレクトバスのいずれを介してデータを転送又は受信するかを決定することを含み、データを転送する場合の当該決定が、転送されるデータの転送先のアドレスが前記データを転送する専用プロセッサのローカルメモリ、この専用プロセッサのピアメモリ、およびホストブリッジバス上のシステムメモリのいずれのアドレス範囲に含まれるかに基づき行われ、前記転送先のアドレスがデータを転送する専用プロセッサのピアメモリを特定する範囲内に含まれるときの当該決定が、ホストブリッジバスおよびダイレクトバスのそれぞれにおけるトラフィック量に基づき行われる、

コンピュータ可読媒体。

【請求項１７】

前記ピアメモリを特定する範囲内の特定アドレスが、ホストブリッジバス又はダイレクトバスのいずれかに割り当てられる、請求項１６に記載のコンピュータ可読媒体。

【請求項１８】

前記実行される方法に、動的負荷分散を行なって、転送すべきデータの一部分がホストブリッジバスを介して転送され、前記データの他の部分がダイレクトバスを介して転送されるように、ホストブリッジバス及びダイレクトバスのそれぞれにおけるトラフィックの割合を、トラフィック履歴データに基づき調整することが更に含まれる、請求項１６に記載のコンピュータ可読媒体。

【請求項１９】

前記実行される方法に、メモリアパーチャをホストプロセッサ及び複数の専用プロセッサのそれぞれに割り当てることが更に含まれ、メモリアパーチャはピアメモリ、システムメモリ、及びローカルメモリに適用され、アパーチャが各プロセッサのメモリの見方を決定する、請求項１６に記載のコンピュータ可読媒体。

【請求項２０】

前記実行される方法に、アパーチャ圧縮を行なうことが更に含まれ、同圧縮が、オフセットレジスタを用いて、オフセットを変更することにより全てのローカルメモリを可視化することを含む、請求項１９に記載のコンピュータ可読媒体。

【請求項２１】

前記インストラクションは、実行する方法を遂行するための特定用途向け集積回路（ＡＳＩＣ）の作製に使用することのできるハードウェア記述言語によるインストラクションを含む、請求項１９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】

本発明はコンピュータ及びその他のデジタルシステムにおけるデータ転送の分野に関する。

【関連出願】

【０００２】

本願は２００８年3月３１日出願の米国特許仮出願第６１／０４１，２１０の優先権を主張し、その全文を参照により本明細書に援用する。

【背景技術】

【０００３】

コンピュータ及びその他のデジタルシステムの複雑化や能力の向上とともに、システムコンポーネント又は要素間のデータ転送を向上する方法及びハードウェアは絶えず進化している。転送されるデータは、データを表す信号、コマンド、又はその他のあらゆる信号を含む。データ転送の速度及び効率は、グラフィックスアプリケーションなどの非常にデータ量の多いアプリケーションを動作させるシステムにおいて特に重要である。通常のシステムでは、グラフィックス処理能力は、中央処理装置（ＣＰＵ）の能力の一部として与えられるか、又は、ＣＰＵと通信しビデオゲームなどの用途のグラフィックスデータ処理を補助するグラフィックス処理装置（ＧＰＵ）などの別個の専用プロセッサにより与えられる。１つのシステムには１又は複数のＧＰＵが含まれることがある。従来のマルチＧＰＵシステムでは、ブリッジ接続されたホストインターフェース（例えばペリフェラル・コンポーネント・インターフェース・エクスプレス（ＰＣＩｅ（登録商標））は、ホストトラフィックと帯域を共有しなければならない。図１は、ルートプロセッサ１０８（例えばＣＰＵ）、ブリッジ１０６、及び２つのエンドポイントＥＰ０、ＥＰ１（例えばＧＰＵ０、ＧＰＵ１）１０４ａ、１０４ｂを含む従来例のシステム１００のブロック図である。各エンドポイント１０４はそれぞれのメモリコンポーネント１０２ａ、１０２ｂに結合されている。

【発明の概要】

【発明が解決しようとする課題】

【０００４】

システム内の２つ以上のＧＰＵの存在を利用するため、複数のＧＰＵは処理タスクを共有しており、このためこれらＧＰＵ間のデータ転送が必要となる。マルチＧＰＵシステムでの良好な性能スケーリングの実現における問題点とは、ＧＰＵ間で往復しなければならないデータが潜在的に多く存在することである。１つの状況例として、描画に多くの時間を要する背景又はその他のテキスチャの描画をアプリケーションが要求する。この場合、背景又はその他テキスチャを次のフレーム又は同一フレームの別部分に使用できるよう、別のＧＰＵにコピーするのが望ましい。システム１００においては、ＧＰＵ１０４間でデータを転送する唯一の経路は、ブリッジ１０６への各経路１０５ａ、１０５ｂを介するものである。ルート１０８への経路は各ＧＰＵ１０４からルート１０８へのトラフィックだけでなく、２つのＧＰＵ１０４間のトラフィックも運ばなければならない。このため物理的ワイヤが共有され、ボトルネックが形成されかねない。

【図面の簡単な説明】

【０００５】

【図１】従来例のマルチプロセッサシステムのブロック図である。

【図２】ピアツーピア入出力経路を追加で有する一実施形態に係るマルチプロセッサシステムの一部のブロック図である。

【図３】ピアツーピア入出力経路を追加で有する一実施形態に係るマルチプロセッサシス

テムのより詳細なブロック図である。
【図４】一実施形態に係る入出力システムが実行する方法のフロー図である。
【図５】一実施形態に係る非対称入出力システムにおける位相情報の通信方法を示すフローチャートである。
【図６】一実施形態に係る非対称入出力システムにおけるホスト機器へのクライアント位相情報の通信を示すタイミング図である。
【発明を実施するための形態】
【０００６】
本明細書では、ピアツーピア専用プロセッサのアーキテクチャ及び方法の実施形態を説明する。実施形態は、少なくとも１つの中央処理装置にホストブリッジバスを介して結合された複数の専用プロセッサを有するシステムを含む。ダイレクトバスが複数の専用プロセッサのそれぞれを、複数の専用プロセッサの少なくとも別の１つに直接結合する。メモリコントローラが複数の専用プロセッサに結合され、メモリコントローラはホストバス又はダイレクトバスのいずれを介してデータを転送するかを決定する。
【０００７】
図２は一実施形態に係るマルチプロセッサシステム２００の一部のブロック図である。システム２００はブリッジ２０６に結合されたルートプロセッサ２０８を含む。ブリッジ２０６は各バス２０５ａ、２０５ｂを介してバスエンドポイントＥＰ０（２０４ａ）、ＥＰ１（２０４ｂ）に結合されている。各エンドポイント２０４はそれぞれのメモリ装置２０２ａ又は２０２ｂに結合されている。
【０００８】
一実施形態においては、ルートプロセッサ２０８はＣＰＵで、各エンドポイント２０４がＧＰＵであるが、実施形態はこれに限られない。以下で詳しく述べるように、実施形態は、それぞれがＰＣＩｅ（登録商標）ポートと類似のサイドポートを有するＧＰＵであるエンドポイント２０４を含む。システム２００のアーキテクチャでは、性能の向上のために、ＰＣＩｅ（登録商標）のアドレッシング機能や様々な負荷分散機能が用いられる。一般的なアーキテクチャでは、１つのＧＰＵから別のＧＰＵへの読み込み演算が特に高価になる。読み込み演算のためには、要求が送られ、しばらくしてから要求データを受信する。実施形態は、読み込み演算を含む各演算の一般的性能に渡る改良を提供する。
【０００９】
図３は一実施形態に係るマルチプロセッサシステム３００のブロック図である。システム３００はシステム２００と同様であるが、より詳細に示されている。ＣＰＵ３０８がシステム３００のルートプロセッサである。ブリッジ３０６がＣＰＵ３０８をシステムメモリ３０９に結合する。ブリッジ３０６はまた、ホストブリッジバス３０５ａを介してＣＰＵ３０８をＧＰＵ０（３０４ａ）に結合し、ホストブリッジバス３０５ｂを介してＣＰＵ３０８をＧＰＵ１（３０４ｂ）に結合している。
【００１０】
ＧＰＵ３０４は各サイドポートを通じダイレクトバス３０７を介して互いに結合されている。各ＧＰＵ３０４は更に、別々のポートを介してそれぞれのローカルメモリ３１３ａ、３１３ｂに結合されている。ＧＰＵ０（３０４ａ）は様々なクライアント３１１ａを含む。クライアント３１１ａはメモリコントローラ３０３ａと通信する。メモリコントローラ３０３ａはクライアント３１１によるローカルメモリ３１３ａ（例えばビデオメモリ）へのアクセスを制御し、クライアント３１１によるバス３０５ａへのアクセスを制御する。具体的には、メモリコントローラ３０３ａは、どのクライアント３１１がローカルメモリ３１３ａと通信するかと、どのクライアント３１１がバス３０５ａといつ通信するかとを決定する。
【００１１】
ＧＰＵ１（３０４ｂ）は同様に、メモリコントローラ３０３ｂに結合された様々なクライアント３１１ｂを含む。メモリコントローラ３０３ｂはローカルメモリ３１３ｂと通信する。ＧＰＵ０（３０４ａ）及びＧＰＵ１（３０４ｂ）はダイレクトバス３０７を介して

互いに結合されている。ダイレクトバス３０７は、各ＧＰＵ３０４の新規なサイドポート間のピアツーピア経路である。

【００１２】

一実施形態において、クライアント３１１による様々なバスへのアクセスはアドレス範囲により決定される。メモリ３０９、３１３へのアクセスはアドレス範囲に基づく。一実施形態において、メモリアドレス空間は３つの部分に区分されている。システムメモリ３０９用の部分、ローカルメモリ３１３ａ用の部分、及びローカルメモリ３１３ｂ用の部分である。メモリ３１３ａはＧＰＵ３０４ｂから見てピアメモリであり、逆もまた同様である。ピアアドレスとは、別のＧＰＵ上のメモリを参照するものである。ローカルメモリへのアクセスはブリッジ３０６により一般的方法で処理される。一実施形態において、アドレスがピアアドレス範囲に入るときは、利用可能な経路の双方（ダイレクトバス３０７及びホストブリッジバス３０５）が使用される。ある経路又は別の経路を介して送信されるデータ量は、異なる実施形態における様々な状態に左右される。例えばあるモードによれば、データが両経路間で等分され、データの５０％がバス３０７により送信され、データの５０％がバス３０５により送信される。このモードにより、従来のアーキテクチャでは不可能な性能の向上が得られる。但し、当業者には明らかであるように、バス３０５は依然として（上述のように送信されるデータに加えて）別のトラフィックにより共有されている。このため、本モードにおいては、性能を更に向上するために動的負荷分散も用いてもよい。例えば、（ダイレクトバス３０７を使用する）直接経路を好適な経路としてもよいが、この経路が満杯のときは他の経路（例えば経路３０５ａ、３０５ｂを使用する）を用いる。

【００１３】

しかし、次のような状況を考えてみる。

書き込まれるデータとして「６」が所望されるとき、あるデータ書き込みシーケンスが、以下のようになっているものとする（Ｗは書き込みを示し、ＷＡは「メモリアドレス／ロケーションＡに書き込み」を示す）。

１．ＧＰＵ０へ３０５を介してＷＡ（図３に示すようにデータは「４」）

２．ＧＰＵ０へＷＢ

３．ＧＰＵ０へＷＣ

４．ＧＰＵ０へ３０７を介してＷＡ（図３に示すようにデータは「６」）

【００１４】

上記のシーケンスでは「４」の書き込みが最初に行われたが、この書き込みはホストブリッジバス３０５ａ、３０５ｂを介している。「６」の書き込みは後から始まったが、ダイレクトバス３０７（より速い経路）を介している。「６」は最初に書き込まれるが、「４」が後から到着し「６」を上書きする。このような問題が予測される場合、この問題を回避するモードは、アドレスベースルーティングである。同一アドレスへのデータが同一ルートで送信されるようにアドレスを分割することができる。例えば、偶数のダブルワードアドレスをダイレクトバス３０７経由で送信し、奇数のダブルワードアドレスをホストブリッジバス３０５ａ、３０５ｂ経由で送信することができる。あるいは、８つのアドレスのうち７つ（バイナリデータが０００で終わらないもの）をダイレクトバス３０７経由で送信し、残りをホストブリッジバス３０５ａ、３０５ｂ経由で送信することができる。

【００１５】

別のモードによれば、全てのデータが１つの経路又は別の経路を介して送信される。本モードは、１つの経路が使われないことになるので最も望ましいとは言えない。但し、複数のクライアント３１１ａ、３１１ｂにそれぞれ独自のルールを認めることができる。このようにすれば、異なる経路を様々に使用して複数のグラフィックスハードウェア機能を持つことが可能となり、経路の完全利用が実現される。

【００１６】

実施形態においては以下でより詳しく説明するように、書き込み演算について述べる。すなわち、ＧＰＵ０（３０４ａ）がＧＰＵ１（３０４ｂ）にデータを送るが、ＧＰＵ１（

３０４ｂ）からデータを取り戻さない。但し、本明細書に記載の同一の方法に従うその他の実施形態では、読み込み及び書き込み演算はいずれの方向でも実行可能である。

【００１７】

図４は一実施形態に係る２つのＧＰＵを有するシステム４００の要素のブロック図である。図４は、一実施形態に係るグラフィックスサブシステムの相互接続全体を含む（表示可能な画素をマザーボード又は表示装置に転送するための表示リンクは図示せず）。システム４００はＧＰＵ０（４０４ａ）及びＧＰＵ１（４０４ｂ）を含む。各ＧＰＵ４０４は、ダイレクトバス４０７及び各サイドポートを介して相互に直接結合されている。「ＲＣ」で示すブロックは、ルートコンプレックスのコアとして構成されている。「ＥＰ」で示すブロックは構成されるエンドポイントである。ＲＣ及びＥＰはＰＣＩ　Ｅｘｐｒｅｓｓ（ＰＣＩｅ（登録商標））ポート用の一般的用語である。適正な通信のためには、ＲＣは他のＲＣではなくＥＰと通信しなければならない。一般のポート設計で動作するピアツーピア実装を提供する一実施形態においては、ストラッピングロジックが含まれ、４０４ａの「ＲＣ」又は４０４ｂの「ＲＣ」のいずれかが真のＲＣとして働き、他方がＥＰとして働くことができる。各ＧＰＵ４０４はメモリコントローラ（ＭＣ）及びホストデータパスブロック（ＨＤＰ）を含む。図中説明文に示すように、通信経路４２１は読み込み／書き込み（Ｒ／Ｗ）＋ｄの要求の経路であり、「Ｗ＋ｄ」は「書き込み要求及びデータ」を意味する。「ＣＰＬデータ」の矢印４１３はデータと共に読み込み完了用の経路を示す。矢印４２３はＷ＋ｄの要求の経路を示す。内部レジスタ、リードオンリーメモリ（ＲＯＭ）、オーディオ、ビデオグラフィックスアレイ／アダプタ（ＶＧＡ）などのその他のＧＰＵ内部ブロックに対するホストＰＣＩｅ（登録商標）のインターフェースは図示していない。

【００１８】

ＧＰＵ４０４は更に、クロスバー４１７を通じ、各ホストブリッジバス４０５ａ、４０５ｂを介してブリッジ４０６（本実施形態ではノースブリッジ）に結合されている。ＣＰＵ４０８がブリッジ４０６に結合されている。

【００１９】

図５は一実施形態に係る４つのＧＰＵを有するシステムのブロック図である。一実施形態においては、２つのＧＰＵ、ＧＰＵＡ（５０４ａ）及びＧＰＵＢ（５０４ｂ）が１つのボード（ボード０）を占有し、更なる２つのＧＰＵ、ＧＰＵＣ（５０４ｃ）及びＧＰＵＤ（５０４ｄ）が別のボード（ボード１）を占有している。ＧＰＵ５０４ａ及びＧＰＵ５０４ｂは、ダイレクトバス５０７ａ及び表示リンク５１１ａを介して結合されている。ＧＰＵ５０４ａ及びＧＰＵ５０４ｂはそれぞれ、各ＰＣＩｅ（登録商標）バスを通して単体ブリッジ５０９ａに結合されている。

【００２０】

ＧＰＵＣ（５０４ｃ）及びＧＰＵＤ（５０４ｄ）は、ダイレクトバス５０７ｂ及び表示リンク５１１ｃを介し結合されている。ＧＰＵ５０４ｃ及びＧＰＵ５０４ｄはそれぞれ、各ＰＣＩｅ（登録商標）バスを通して単体ブリッジ５０９ｂに結合されている。

ボード０及びボード１は表示リンク５１１ｂにより結合される。ボード０及びボード１は更に、単体ブリッジ５０９ａからブリッジ５０６へのＰＣＩｅ（登録商標）バス、及び、単体ブリッジ５０９ｂからブリッジ５０６への別のＰＣＩｅ（登録商標）バスを介して、ブリッジ５０６を通して結合される。ブリッジ５０６にはＣＰＵ５０８が結合されている。システム５００では、いかなるペアのＧＰＵの間でも、ピアツーピアの同時転送が可能である。一実施形態においては、ＡＢ及びＣＤのペアの場合、各転送は持続的に最大約１２．２ＧＢ／秒で行われる。ＡＣ及びＢＤのペア又はＡＤ及びＢＣのペアで転送が両方向の場合、各転送は約６．１ＧＢ／秒で行われる。ＡＣ及びＢＤのペア又はＡＤ及びＢＣのペアで転送が同一方向の場合、各転送は約３ＧＢ／秒で行われる。

【００２１】

図６は一実施形態に係るサイドポートサブシステムのブロック図である。以下の標題を付した項目では、図６を参照して様々な実施形態に係る実装の詳細を示すが、特許請求さ

れる発明はここに記載の具体的詳細に限定されるものではない。

【００２２】

＜ホストＰＣＩｅ（登録商標）エンドポイント＞

一実施形態においては、ホストＰＣＩｅ（登録商標）インターフェースが、バスインターフェースコアロジック（ＢＩＦ）、アナログＰＨＹ、及びラッパーから構成される。インターフェースはエンドポイント（ＥＰ）として構成される。バスは、５Ｇｂｐｓ／ピンの標準的なＰＣＩ　Ｅｘｐｒｅｓｓ２．０仕様（「ＰＣＩｅ２（登録商標）」）の速度で、又は１６レーンのインターフェースでの理論上の最大速度８ＧＢ／秒で動作する。バスは（ホスト又はピアへ）送信されるメモリの書き込み用に１２８ｂのクレジット／デビットマスターインターフェースを有し、（ホスト又はピアから）受信するメモリの書き込み用に１２８ｂのクレジット／デビットスレーブインターフェースを有する。バスはリードオンリーメモリ（ＲＯＭ）、内部レジスタバス、オーディオ、及びビデオグラフィックスアレイ（ＶＧＡ）への３２ｂのインターフェースを有する。バスは、複数のＧＰＵ宛のホスト要求を自身のスレーブインターフェース（Ｓで示す）を介してルーティングする。

【００２３】

要求インターフェースは、これに伴う完了インターフェースを有する。ＢＩＦは、フレームバッファ用、レジスタ用、ＲＯＭ用のアドレスアパーチャセットによりトラフィックを案内する。ＢＩＦはコンフィギュレーションレジスタ及びストラップ用に３２ｂのインターフェースを用いる。ＢＩＦはまた、ＡＣＰＩの電力状態管理、時計、リセット、及びスキャン入力を処理するためのクロックジェネレータ（ＣＧ）へのインターフェース、及び、デバッグバスデイジーチェーン用インターフェースを有する。ＢＩＦはまた、フレームバッファ読み書き用のホストデータパスブロック（ＨＤＰ）へのインターフェースを有する。

【００２４】

＜サイドポート・ルートコンプレックス＞

サイドポート（ＸＳＰ）に用いられるＰＣＩ　Ｅｘｐｒｅｓｓのロジックは、ルートコンプレックス（ＲＣ）として構成されている。ＲＣとＥＰの演算モード間にはその構成及び能力に僅かながら違いがある。ＸＳＰバスは０．５Ｇｂｐｓ／ピンの標準的なＰＣＩｅ２（登録商標）仕様の速度で、又は１６レーンのインターフェースの理論上の最大速度８ＧＢ／秒で動作する。同バスは、ピアＧＰＵへ送信されるメモリコントローラ（ＭＣ）のメモリ空間への書き込み要求用に用いられる１２８データビットのクレジット／デビットマスターインターフェースを有し、また、ピアＧＰＵから受信するメモリの書き込み用に１２８ｂのクレジット／デビットスレーブインターフェースを有する。これらはホストＢＩＦのインスタンスに対しＨＤＰにより用いられるインターフェースである。ＸＳＰバスは、コンフィギュレーションレジスタ及びストラップ用に３２ｂのインターフェースを用いる。また、ホストインターフェースとは別個に、自身のリンクパワーステータスを維持する。ホストインターフェースと同様に、レーン数はソフトウェアにより８又は１６にプログラム可能である。ＰＣＩｅ（登録商標）の２つのＰＨＹが特定用途向け集積回路（ＡＳＩＣ）上で同速度で動作するため、単一ペアの基準クロック（ＲＥＦＣＬＫ）入力ピンを用い、バッファされたＲＥＦＣＬＫを２つのＰＨＹにオンダイで分配することができる。

【００２５】

ＸＳＰインスタンスはその他の読み書きインターフェースを一切必要とせず、また、ホストＢＩＦがサポートする処理を中断しない。サイドポートはＡＳＩＣの起動時に始動する必要がないため、ピン式ストラップをほとんど使用しない。サイドポートのコンフィギュレーションは、ＳＲＢＭバス上へのソフトウェアレジスタの書き込みを介して、ホストインターフェースとは別個にプログラム可能である。

【００２６】

ＸＳＰはレジスタの読み書き、入出力空間、又はコンフィギュレーション空間のサイクルをサポートする必要がない。本デバイス上で動作可能とするためには、単一のＰＣＩｅ

（登録商標）Ｆｕｎｃｔｉｏｎを必要とするのみである。ポステッドライトのみが発行され（読み込み無し）グローバル同期スキームを使用する一実施形態においては、完了メッセージは不要である。

【００２７】

ＭＣハブは書き込み要求をＸＳＰにルーティングする。ＸＳＰ要求には、ホストＢＩＦ要求用と同じアドレスが使用される。

【００２８】

＜ホストデータパスブロック＞

一実施形態において、ホストデータパスブロック（ＨＤＰ）はバスＩＰの一部である標準的ブロックである。ＨＤＰはホストインターフェースが行なうフレームバッファ読み書きを処理する。マルチＧＰＵシステムでは、ホスト又は他のＧＰＵが始めたピアツーピアの書き込みのいずれからもフレームバッファ読み書きが来る可能性がある。ホストからのアクセスには複数のＨＤＰのタイリングアパーチャの１つを使用することができるが、ピアツーピアからのアクセスには（ピアはリクエスタＩＤにより区別される）使用してもしなくてもよい。ＨＤＰは読み書きクライアントとして、ホストＢＩＦのスレーブポート及びＭＣへインターフェースする。ＢＩＦ及びＨＤＰは協働して読み書きの一貫性を維持する。読み込みは先行する書き込みを追い越しできない。ピアツーピアの書き込み同期のために、予約されたメモリ空間への特別の書き込みがＨＤＰによって認識され、ＭＣ読み込み及び特定の「書き込みフェーズ」用の先行する書き込みフラッシュをローカルメモリまでトリガーする。これに続き、送信元クライアントのレジスタ空間内の、フェーズごとのメールボックスへライトバックが行われる。

【００２９】

バスから受信する読み書きはＢＩＦ内の専用アパーチャによりＨＤＰにルーティングされる。このアパーチャはローカルメモリの３２ＭＢ以上に相当する。任意で、ＨＤＰにより処理されるアドレスはバーチャルアドレスとして扱われ、ＨＤＰ専用のページテーブル内でルックアップされる。

【００３０】

利用可能なＰＣＩｅ（登録商標）アパーチャを各ＧＰＵ用に最大限利用し、書き込みフェーズをサポートするために、ピアツーピアの書き込みアドレスは送信元ＧＰＵのＭＣによりフェーズ及びオフセットとしてエンコードされており、またこれにフェーズ毎のベースアドレスレジスタ（ＢＡＲ）の自動更新情報を組み入れておくこともできる。ＨＤＰはＢＩＦからのリクエスタＩＤと送信されるビットとの組み合わせを用いて、どの書き込みフェーズ及び動的アドレスアパーチャを使用するかを決定する。

【００３１】

＜メモリコントローラ（ＭＣ）ブロック＞

ＭＣはサイドポートに接続するためのインターフェースを２つ有する。ＭＣクライアントから要求される書き込み用の１２８ｂのＢＩＦマスターインターフェースと、他のＧＰＵから受信するローカルメモリへの書き込み用の１２８ｂのＢＩＦスレーブインターフェースである。ＭＣはまた、マスターからの要求をＸＳＰを介してピアメモリにルーティングするため、アドレッシングロジックを内蔵している。ＸＤＰブロックはインターフェース機能を処理する。ＸＤＰはスレーブ要求を見て、アドレスオフセッティング又はビデオメモリ（ＶＭ）ルックアップのいずれかを行い、書き込み完了カウンタを維持し、ライトコンバイニング（３２ＢのＢＩＦ要求を６４ＢのＭＣ要求へ）を提供する。

【００３２】

サイドポートへの要求及びサイドポートからの要求は１６バイト（マスキング込み）から６４バイトであり得る。性能を最良とするには要求は６４Ｂであるべきである。

サイドポートＢＩＦインターフェースは、全グラフィックスバーチャルメモリがアクティブか否かに関わらず、ＧＡＲＴ変換された４０ｂシステムアドレスを通過させる。ＧＡＲＴはシステムコンテキストドメインと同様であり、３６ビットＧＰＵデバイスアドレスからの変換を行なう。

【００３３】
ＭＣは、マルチＧＰＵによるコピー演算のフラッシュを保証するようなやり方で書き込み完了ステータスを維持する。ＨＤＰと同様に、ＸＤＰにより認識される専用アドレスを用いて、特定の書き込みフェーズの書き込み完了チェックをトリガーし、送信元のクライアントのレジスタ空間内のメールボックスへライトバックを生成する。
【００３４】
＜ダイレクトメモリアクセスエンジン（ＤＲＭＤＭＡ）＞
サイドポート付きＧＰＵは、メモリのホストブロック転送（ＢＬＴ）を効率的に行なうことができるＤＭＡエンジンを含む。ＤＲＭＤＭＡは、ローカルメモリ、ホストメモリ、及びその他のＧＰＵのローカルメモリのいずれの２つの間でも転送できるようにプログラムすることができる。（後述する）アドレス処理の向上により、ＤＲＭＤＭＡはまた、ＸＳＰ及びＰＣＩｅ（登録商標）ブリッジの双方を利用してメモリのブロックを移動することができる。一実施形態の標準的転送能力は以下を含む。
・ホストからローカルへ（ホストＢＬＴ又はコピー）：１６Ｂ／最大クロックレート
・ローカルを一定値で満たす：３２Ｂ／最大クロックレート
・ローカルからホストへ（コピー）：１６Ｂ／最大クロックレート
・ローカルからリモートＧＰＵへ（コピー）：隣接ＧＰＵへは３２Ｂ／最大クロックレート、非隣接へは１６Ｂ／最大クロックレート
【００３５】
ＤＲＭＤＭＡはクロック毎に３２Ｂの読み込み及び３２Ｂの書き込みを持続できる。これは２つの１６ＢのＰＣＩｅ（登録商標）書き込みインターフェースに適合する。
【００３６】
＜ホスト対サイドポートの負荷分散＞
サイドポートを１つ追加すると、隣接ＧＰＵにはより大きな帯域が与えられる。隣接ＧＰＵでは２ＧＰＵ及び４ＧＰＵ使用モデルでは帯域の大半が必要とされる。ホストポートもまた利用可能な帯域を多く有しており、なおかつ、隣接ＧＰＵとの通信用に利用可能でなければならない。
【００３７】
負荷分散の最も簡単な方法は、例えばサイドポートに割り当てられたＤＲＭＤＭＡ及びホストポートに割り当てられたカラーバッファブロックなど、異なるクライアントへの固定割り当てに基づいて、データ転送に用いるポートを静的に割り当てることである。
【００３８】
負荷分散の２つ目の方法は、サイドポートの使用負担状況に基づき、いずれかのクライアントからのトラフィックを２つのポートのうちいずれかに動的にルーティングするものである。下記の「ＡＮＹ」負荷分散の説明を参照されたい。
【００３９】
上記２つ目の方法を用いると、同一画素アドレスへの書き込みに対して２つの異なる経路を介した競合状態があり得る。この競合状態によって、書き込みが２つの異なるプリミティブに属する場合に、画素順序ルールが破られる可能性がある。先のプリミティブ用の書き込みが後のプリミティブ用書き込みの後に到着する可能性がある。この問題を解決するため、アドレスに基づいてルーティングするというオプションがサポートされている。予備策としては、関連のクライアントに単一の経路を用いる。
【００４０】
ピア書き込み要求（ｒｅｑｓ）がホスト及びサイドポートを介して隣へ送信されている場合、これら２ポート間で遅延時間が異なり変動するために要求順序が失われる可能性がある。状況によってはホストポートがフリーズすることさえあり得る。すなわち、必要なときに要求順序が守られることを確実にするための手順を踏まなければならない。
【００４１】
ＭＣからの唯一の順序保証として、同一クライアントからの同一アドレスへの書き込みが並び替えされることはない。従ってこれが、ピア書き込みに対しても唯一のライト・ア

フター・ライト（ＷＡＷ）要求となる。これを保証する、実施形態に従う幾つかの方法がある。ＭＣはローカルメモリだけでなくホスト及びピアへの読み書きを処理する。ＭＣの初期設定の順序ルールは上述の通りである。追加のルーティングルール及びバス自身が、更なる順序保証を提供する。全ての負荷分散は、隣接ＧＰＵへ向かう要求にのみ適用される。これはアパーチャチェックにより判定され、非隣接要求は全てホストポートへ向かう。

【００４２】

＜負荷分散：ポートへの固定マッピング（「ＦＸＤ」）＞

同一アドレスへの２つの書き込みの順序に狂いが生じないよう確実にする１つの方法は、アドレスチェックにより負荷分散するものである。アドレスチェックは、任意のアドレスが常に同一のポートを使用することを確実とするために行われる。例えば、一定の粒度に対して、「偶数」アドレスは常にホストポートを使用し、「奇数」アドレスは常にサイドポートを使用するようにする。これにより、同一アドレスへの２つの要求はポートを交差はするが順序が狂うことはない。

【００４３】

使用するアドレスチェックマッピングは、いずれのアドレスビットを見るか、及び、どの閾値を分散に使用するかをプログラムすることができる。使用するアドレスビットによって分散粒度を設定する。粒度が大きいことは、おそらく書き込みバーストがいずれかのポートに渡ってとどまることを意味するのに対し、粒度が大き過ぎると、大きなバーストがいずれかのポートに送信されながら、あるポートが枯渇する可能性があるからである。使用する閾値によって、ホスト及びサイドポート間の分散率をより細かくできる。例えば３つのアドレスビットを使用する場合、使用する閾値により、ホスト：サイドの比率を０：８、１：７、２：６、…、７：１、８：０などにできる。ホストポートのみ又はサイドポートのみに全ての要求を送信するのは、この固定マッピングを縮退させた場合である。

【００４４】

この種の固定負荷分散では、あるポート（ホスト）が非常に多忙となるか又はフリーズすると、バックアップが、他のポート（サイド）に送信可能な帯域を抑制する可能性がある。また、ある種の書き込みパターンにより簡単に、所望の分散率が打ち消される可能性がある（例として、常に１つのポートにマッピングし、他のポートを使用しないままにする跨ぎ書き込みがある）。従ってこの負荷分散方法には、利用可能な帯域に反応するというより静的でありソフトウェアでプログラムされることによる、望ましくない特徴がある。以下に、動的負荷分散の代替法を説明する。

【００４５】

＜負荷分散：非制約の、又はＷＡＷが保証されないあらゆる順序（「ＡＮＹ」）＞

ＡＮＹ負荷分散は、満杯又はほぼ満杯の状態に基づいてポート間を切り替える。サイドポートは利用可能である可能性が高いため、こちらが多く選択されるはずである。各ポートにおける未処理のクレジット数を制御し、分散を調整すべきである。クライアントによってはＷＡＷ一貫性の保証を必要としないものもある。例えば、コピーは同一コピー内に同一アドレスを２度転送することはないので、コピー間のフラッシュにより一貫性を保証することができる。これをサポートすることにより、ソース及び宛先は、重複する可能性のある転送間で通常の書き込み完了同期を行なえばよいだけなので、全速力で進行することができる。このトラフィックをＡＮＹ型のトラフィックと混合することで、これらの書き込みがギャップを埋めることができる可能性があり、それぞれの弱点を幾らか軽減できる可能性がある。この場合、ホストポートはサイドポートが満杯のときのみ使用され、サイドポートが最大限使用される。

【００４６】

負荷分散の実装方法は全て、独立した転送ストリームと同時に使用することができる。いずれのタイプの負荷分散を用いるかは、フェーズごとにソフトウェアが設定することができる。

【００４７】

ＡＮＹトラフィックとＦＸＤトラフィックが混在する場合は、ＦＸＤオプションの弱点により生まれるバブルを埋めるようにＡＮＹトラフィックが補助すべきである。ハードウェアは、これらＡＮＹ型要求に、フリーズ又は待機中のＦＸＤ要求の幾つかをバイパスさせるサポートを有する。

サポートされる負荷分散モードはＦＸＤ及びＡＮＹの２つのみであるが、その他の多くの実施形態も可能である。

【００４８】

＜負荷分散：可能性のある使用状況＞

次の使用モデルがあり得るが、実施形態はこれに限られない。

１）ピア書き込みがＤＭＡである。：データをピアＧＰＵにコピーするため１～２フェーズがＤＲＭＤＭＡに使用されている。

【００４９】

１ａ）「ＡＮＹ」型負荷分散を転送に使用できるが、重複の恐れのあるＤＭＡ間の同期を更に必要とする場合がある。重複するかどうかをソフトウェアでチェックし判定することができる。

【００５０】

１ｂ）２つのＤＭＡエンジンに対する負荷分散の代替的オプションは、１つのエンジンが「ホストのみ」を使用し、１つのエンジンが「サイドのみ」を使用するようにすることである。又は、「ホストのみ」が非隣接ピアへの書き込みにより暗示される場合もある。ホストポート用に利用可能な帯域は保証されていないため、ソフトウェアがコピーワークを最適に分割するのは困難かもしれないが、２つのフェーズが同期遅延時間を隠せば、重複する可能性のあるコピー間の同期を更に必要とすることなく両ポートが満杯に維持されるであろう。関連のリソースが重複していなければ、ハードウェアは「ホストのみ」フェーズ及び「サイドのみ」フェーズに同一のフェーズ番号を用いさせることができる。

【００５１】

２）ＷＡＷ感応性のクライアント書き込み（コピーも使用される場合とされない場合）

２ａ）上記１ｂと同様、１又は複数のピア書き込みが送信されているとき、それぞれに「ホストのみ」又は「ピアのみ」の専用経路を割り当てることができる。これにより両ポートが満杯に使用されて、順序に依存したり、ポート間でクロスカップルされるバックプレッシャー経路が形成されることがない。

【００５２】

２ｂ）このＷＡＷ感応性クライアントが転送を行なう唯一のクライアントである場合、このときに限って、ポートを満杯に利用しておくために「ＦＸＤ」を使用する必要がある。「サイドのみ」は他に問題があるとき（及び／又は別のクライアントが一時的に「ホストのみ」を使用することができるとき）の安全なバックアップとなる。

【００５３】

＜アドレッシング＞

ＧＰＵのホストインターフェースは、（メモリ空間にマッピングされた）レジスタ及びローカルメモリ（フレームバッファ）だけでなく、その他のブロックのバス要求を受信する。ＢＩＦ内のアパーチャのセット（又はアドレス範囲）が、ＰＣＩｅ（登録商標）空間アドレス（６４ｂ）をＧＰＵ内の特定の宛先にマッピングする。ローカルメモリの読み書きはＭＣへ向かう際にＨＤＰを通過する。

【００５４】

ＧＰＵ起源のメモリ要求が使用するアドレスはバーチャルアドレスであってもなくてもよい。ＷｉｎｄｏｗｓＸＰ又はＶｉｓｔａのグラフィックスドライバモデル用には、システムメモリ（ＧＡＲＴを介し更にマッピングされてもよい）又はローカルメモリを含み、アパーチャにより区分されたフラットなアドレス空間がＧＰＵ「デバイス」アドレスにより形成される。将来のドライバモデル用としては、ＧＰＵアドレスはページがマッピングされたバーチャルアドレスとして扱われ、ルックアップされる物理的アドレスはシステムメモリとローカルメモリとで別個のものとされる。

【００５５】
＜システムメモリ空間＞
ＧＰＵから見た場合、システムメモリとは通常、ＣＰＵに付随の動的ランダムアクセスメモリ（ＤＲＡＭ）を意味するが、実用上は、システムメモリはあらゆる利用可能なメモリマップド・デバイス、特に、他のＧＰＵのメモリに属するあらゆるメモリ空間を含む。システムメモリアクセスのマッピング及びルーティングは、システムコンフィギュレーションに応じて、単体ブリッジ及びチップセット（例えばノースブリッジ）のブリッジの組み合わせにより処理される。
【００５６】
ＣＰＵから見た場合、物理的アドレス空間（３２ｂ又は６４ｂ）はＣＰＵに付随の真のシステムメモリと「その他」メモリとに仕切られる。後者はメモリマップド・デバイス（例えばＰＣＩｅ（登録商標）バス上）にあるメモリである。この仕切りは、入出力（Ｉ／Ｏ）システムの最上ノード内の、コンフィギュレーションレジスタの１つのベース－リミットペアを介して行われる。レジスタ空間用には別のペアもある。そこより下のバイナリツリーの各レベルにおいて、バスアドレス範囲は各下流工程ごとにベース－リミットペアにより２つに分割される。ＧＰＵのアパーチャサイズはボードストラップにより設定される。このサイズは、現在、３２ｂ及び６４ｂのオペレーティングシステムのいずれにも適用されており、システム内にＧＰＵを幾つ挿入するかに関わらず適用されている。
【００５７】
ＧＰＵから見た場合、システムアドレスはＧＡＲＴ（システムページテーブル）を介して、又は直接、バスアドレス空間にマッピングされ、生成されない最上位ビット（ＭＳＢ）は０にセットされる。各上流ＰＣＩｅ（登録商標）ブリッジノードはベース－リミットペアをチェックし、更に上流か、又はピア接続へルーティングする。
【００５８】
図７は、２つのＧＰＵを有するシステムメモリ空間のハイレベルダイアグラムであり、各構成要素から見た場合の同空間を示す。図７ＡはＣＰＵから見た図、図７ＢはＧＰＵ０から見た図、図７ＣはＧＰＵ１から見た図である。
【００５９】
＜ビデオメモリ無効＞
ＧＰＵによりＭＣに提示される３６ビットのアドレスでは、次のオペレーションシーケンスが行われる。
【００６０】
１．ローカルメモリアパーチャをチェックする。アドレスがこの範囲内に入る場合はベースアドレスを取り出し、物理的ローカルメモリアドレスとして使用する。
【００６１】
２．ローカルメモリアパーチャ外のときは、サイドポートメモリアパーチャをチェックする。これは概して、隣接ＧＰＵにより利用されるアドレス範囲に相当する。アドレスがこの範囲内に入る場合は、負荷に応じてサイドポート又はホストインターフェース上で変更せずに用いる。アドレスがサイドポートに行く場合は、リモートＧＰＵクライアントが自身でアパーチャチェックし、このチェックは結果的にローカルメモリアパーチャ内で合格となるはずである。アドレスがホストインターフェースに行く場合は、ブリッジ又はノースブリッジ内のアパーチャチェックが要求をリモートＧＰＵにルーティングする。リモートホストインターフェースはこれらの要求をそのＨＤＰ又はＳＲＢＭに渡す。
【００６２】
３．アドレスが上述の２つのアパーチャ外のとき、それはシステムアドレスである。アドレスが非スヌープドアパーチャ内に入る場合は、変更されずにホストインターフェースに渡される。そうでなければシステムＧＡＲＴ（例えばシステムページテーブル）内でルックアップされた後、ホストインターフェースに渡される。
【００６３】
＜ビデオメモリ有効＞

ＧＰＵアドレスはバーチャルとして扱われ、ＶＭ内でページテーブルによりシステムの物理的アドレス又はローカルビデオメモリアドレスに変換される。伝統的に、システムの物理的アドレスは４０ｂで、要求の宛先は常に（ＧＡＲＴ、実際はシステムページテーブルを通過した後の）ホストインターフェースである。しかし、ピアツーピア転送用にホストインターフェース及びサイドポートの双方の使用をサポートするために、実施形態ではシステムの物理的アドレスがサイドポートにも渡ってルーティングできるようにする。

【００６４】

一実施形態に係るシーケンスは次の通りである。

１．ＧＰＵクライアント（例えばＤＭＡエンジン）が作製する要求は適切なページテーブル内でルックアップされる。

２．（システム空間用の）「Ｓ」ビットがＰＴＥ内に設定されていない場合は、要求を出しているＧＰＵのローカルメモリがアクセスされる。

３．「Ｓ」ビットが設定されている場合は、サイドポートアパーチャのアドレス範囲のチェックが行われる。

４．アドレスがＳＰＡ外の場合は、要求はホストインターフェースに行く（ＧＡＲＴルックアップ後）。

５．アドレスがＳＰＡ内の場合は、負荷分散決定が行われ、要求はサイドポート又はホストインターフェースのいずれかにルーティングされる。

６．要求がホストインターフェースを介してルーティングされる場合、それは現在行われている通常のピアツーピア要求である。

７．要求がサイドポートを介してルーティングされる場合、システムアドレスがリモートＧＰＵに渡され、そこでＨＤＰ空間のページテーブル内でルックアップされ、そこで（ほぼ常に）ローカルの物理的メモリアドレスにマッピングされる。

【００６５】

＜同期＞

マルチＧＰＵを用いたレンダリングは、データの一貫性を確実とするため描画及びデータ移動工程間での同期を必要とする。描画及びコピーは別々のコマンドストリームが行なう。オルタネートフレームレンダリング（ＡＦＲ）の一般的なシーケンスは次の通りである。

１．ＧＰＵ０上の３Ｄエンジンが動的又は永続的テクスチャ面に描画。

２．３Ｄエンジンがフラッシュして完了を通知（セマフォ信号コマンドを発行）。

３．ＧＰＵ０上のコピーエンジンがセマフォを待機、次いで結果をＧＰＵ１にコピー。

４．コピーエンジンがフラッシュして完了を通知。

５．ＧＰＵ１上の３Ｄエンジンがセマフォを待機、次いでレンダリングされたテクスチャを用いて描画。

【００６６】

ＢＩＦ及びＭＣ（のみならずノースブリッジ及びＣＰＵメモリシステム）のそれぞれは、読み込み及び書き込み演算の順序に関する一貫したルールを有している。エンドツーエンド要求の宛先への一貫性をソフトウェアの干渉なしに複数のＧＰＵ間で維持するには、これらルールを全て考慮した（又はこれらに左右されない）同期スキームが必要である。

【００６７】

マルチＧＰＵに対する同期の必要性は、シングルＧＰＵに両ＧＰＵの間のサーフェスデータのＤＭＡステップを加えた場合のものと同様である。このＤＭＡの場合及びブロードキャスト書き込み能力の場合、あるＧＰＵ上のクライアントに対して他のＧＰＵでの書き込みが完了したことを保証するという問題がある。ＭＣ及びセマフォブロックがこのタスクを補助する。

【００６８】

＜アパーチャ圧縮＞

効率の良いＧＰＵシステムを設計する上での１つの課題は、ＧＰＵの数及び各ＧＰＵのローカルメモリ量が増加するに従い、全てをシステムメモリ空間に収めることが難しくな

り、デバイスアパーチャのサイズを縮小せざるを得ないことである。システム空間にマッピングできるＧＰＵローカルメモリ量が限られると、マッピングされていないメモリの有用性はローカルＧＰＵに限られる。ＣＰＵにマッピングできるＧＰＵ空間の範囲が限られていても、予備策として、ローカルビデオメモリへの固定アクセスの代わりにシステムメモリにサーフェスをコピーするので、大きな懸念はない。

【００６９】

一実施形態による解決法では、オフセットレジスタ（Ｐ２Ｐ　ＢＡＲ）を加えることによってローカルメモリの残りにピアがアクセスできるようにする。宛先ＢＩＦがそのソースからの書き込み要求を見ると、ＨＤＰに３６ｂデバイスアドレスを生成するメモリアパーチャを適用した後、オフセットが追加される（４ＫＢページ粒度）。ソースが見ることのできる連続的ローカルバーチャルメモリ空間の量は以前に比べて大きいわけではないが、これが、Ｐ２Ｐ　ＢＡＲを書き換えることにより変更することのできる全てのローカルメモリへの窓となる。理論上は、変更が書き込みストリーム内の順序に従っていれば、ＢＡＲをいつでも移動することができる。また、複数のオフセットは、ソース（例えば書き込みフェーズ）により区別さえできれば、全て同期している必要はなく、又は空間的に離間している必要はない。

【００７０】

図８はオフセットを有するメモリ空間の図である。図８Ａはシステムメモリ空間のＧＰＵ０から見た図であり、図８ＢはオフセットなしのＧＰＵ１　ＬＶＭマッピングであり、図８ＣはオフセットありのＧＰＵ１マッピングである。

【００７１】

＜書き込みフェーズ＞

単一のＧＰＵが複数セットの書き込みを発行する場合には、仕掛り中の複数セットの書き込みトラフィック（例えば３Ｄパイプ及びＤＲＭＤＭＡコピー）が同時に生じる場合、又は同一クライアントからの書き込みセットが連続する場合のいずれであっても、書き込み完了を保証することが望ましい。後者の場合はフラッシュ／同期演算の遅延時間を隠すことで達成される。二次的に、複数のソースＧＰＵが単一の宛先ＧＰＵに書き込み、各ソースごとに別個に完了を追跡できるようにすることも望ましい。

【００７２】

所望の結果を提供する一実施形態によれば、（宛先）ＢＩＦは各セットごとに３ビットのフェーズＩＤとともに８セットのベース／リミットレジスタ（フェーズ範囲）を与えられる。これらはソースＧＰＵにより様々なやり方で書き込むことができる。

【００７３】

例えば、

・８つまでのＧＰＵのそれぞれが１つのフェーズ範囲を専用のために保持することができる。単一の宛先ＧＰＵ上に範囲を重複して有するのは不明瞭である（「ストリームの交差」）。よりよいスキームとしては、フェーズを単にＰＣＩｅ（登録商標）ソースＩＤにより決定させることで、それにより重複による限定を除くことができる。

・４つまでのＧＰＵのそれぞれが２つの重複しないフェーズ範囲を保持することができる。

・２つのＧＰＵのそれぞれが８つのフェーズ範囲を全て使用できる（実際はＧＰＵソースごとに２つのみが有用と考えられる）。

【００７４】

一実施形態においては、単一のＧＰＵに、明確なフェーズに割り当てることのできるトラフィックのソースが７つある。

・カラーバッファブロック又はデプスバッファブロック
・シェーダーエクスポートブロック
・コマンドプロセッサ
・ＤＲＭＤＭＡ（２つの交互フェーズ）

【００７５】

従って、最終的なフェーズ値は、システム内のＧＰＵの数に応じて、ソースＩＤと範囲チェック結果との組み合わせとなるはずである。

【００７６】

＜書き込みフェーズ及びスモールアパーチャの組み合わせ＞

これらの機能（書き込みフェーズ及びスモールアパーチャ）を同時に可能とすることができる。すなわち、ソースＧＰＵが全ての宛先サーフェスを見て、ＧＰＵごとに複数ソースを有することが可能である。一実施形態では、Ｐ２Ｐ　ＢＡＲをフェーズごとに用いる。これにより、宛先アドレスが単一のアパーチャに収まるには離れ過ぎている単一のソースＧＰＵによって、複数のフェーズを使用することが可能になる。しかし、宛先でうまく範囲チェックを行なうにはＰＣＩｅ（登録商標）空間のアパーチャの大きさが充分ではないことから、Ｐ２Ｐ　ＢＡＲの選択には問題をはらむことがある。

【００７７】

１つの解決法は、フェーズ範囲のレジスタ及び比較を送信元側にて複製し、比較結果を、ａ）結果として得られたフェーズ情報をフェーズごとのＰ２Ｐ　ＢＡＲからのオフセットとともに転送し、ｂ）要求がそのフェーズの現在の範囲外にある場合にはＰ２Ｐ　ＢＡＲを更新するために用いることである。Ｐ２Ｐのアパーチャサイズは予め別個のグローバルレジスタを介して固定される。このＢＡＲの動き方がある程度の量のヒステリシスを内蔵し、更新頻度を最小とする。

【００７８】

この解決法の第２の部分は、拡張された（３６ｂ）バーチャル空間を利用して、リモートＧＰＵメモリを全てマッピングできるようにすること、及び、ソースＢＩＦに行くＰＣＩｅ（登録商標）空間の分解図を利用することである。しかし、フェーズビットを転送する必要性があるために、Ｐ２Ｐ　ＢＡＲを変更せずにアドレッシングできるサーフェスの最大サイズは低減している。ＧＰＵ用の実存ＰＣＩｅ（登録商標）アパーチャは、特にアラインメントを必要としない。フェーズは、利用可能なアドレス範囲の１／ｎパーティションを有することで（宛先側の追加コンパレータを犠牲にして）合理的な粒度に決定される。物理的サーフェスページのスキャッター／ギャザーは宛先ＨＤＰページテーブル又はＧＡＲＴ内で行われ、ＰＣＩｅ（登録商標）空間内ではサーフェスアドレスは連続していると考えられる。

【００７９】

図９はフェーズ圧縮の仕組みを示す図である。フェーズは２つのみ示している。ＧＰＵごとに２５６ＭＢのローカルメモリ及び１２８ＭＢのみのＰＣＩｅ（登録商標）アパーチャがあるものとする。図９Ａはシステムメモリ空間をＧＰＵ０から見た図（ＧＰＵバーチャルメモリ空間）である。図９Ｂはシステムメモリ空間をＰＣＩｅ（登録商標）から見た図（ファブリック物理メモリ空間）である。図９ＣはＬＶＭをＧＰＵ１から見た再構築図（ＨＤＰメモリ空間）である。

【００８０】

限られたＰＣＩｅ（登録商標）アパーチャサイズを最大限利用するため、同一ソースＧＰＵからの複数フェーズの使用を停止してもよい。通常の大きなサーフェスは、１コンポーネントにつき２Ｋ×２Ｋ×３２ｂ又は６４ＭＢである。これはちょうど５１２ＭＢ／８ＧＰＵアパーチャに収まる。一実施形態においては、アパーチャのエンコーディング適用が送信元ＢＩＦの直前に行われる。

【００８１】

本明細書に記載のアーキテクチャ及び方法は、ホストブリッジバスへ結合される複数の専用プロセッサと、複数の専用プロセッサのそれぞれを、複数の専用プロセッサの少なくとも別の１つに直接結合するダイレクトバスと、複数の専用プロセッサの少なくとも１つに結合されたメモリコントローラとを備えたシステムを含む。このシステムでは、少なくとも１つのメモリコントローラが、ホストバス又はダイレクトバスのいずれを介してデータを送信するかと、ホストバス又はダイレクトバスのいずれを介してデータを受信するかとを決定する。

【００８２】
実施形態において、メモリコントローラは、複数の専用プロセッサの少なくとも１つの上に存在する。
【００８３】
実施形態において、複数の専用プロセッサはグラフィックス処理装置（ＧＰＵ）から成る。
【００８４】
実施形態において、ホストバスは、ＰＣＩ　Ｅｘｐｒｅｓｓ（ＰＣＩｅ（登録商標））バスから成る。
【００８５】
実施形態において、ダイレクトバスは、ＰＣＩ　Ｅｘｐｒｅｓｓ（ＰＣＩｅ（登録商標））バスから成る。
【００８６】
実施形態において、ダイレクトバスは、プロプライエタリバスから成る。
【００８７】
実施形態において、システムは少なくとも１つのローカルメモリと、少なくとも１つのシステムメモリと、少なくとも１つのピアメモリとを更に備え、少なくとも１つのピアメモリが専用プロセッサ上に存在し、メモリコントローラが、各メモリのアドレス範囲に基づきいずれのメモリにアクセスするかを決定するように構成される。
【００８８】
実施形態において、アドレスがピアメモリの範囲に入るときには、ホストバス及びダイレクトバスのそれぞれにおけるトラフィック量に基づき、いずれのポートを使用するかのメモリコントローラによる決定が行われる。
【００８９】
実施形態において、アドレスがピアメモリの範囲に入るときには、ホストバス及びダイレクトバスのいずれかが現在満杯か否かに基づき、いずれのポートを使用するかのメモリコントローラによる決定が行われる。
【００９０】
実施形態において、アドレスがピアメモリの範囲に入るときには、特定アドレスのホストバス又はダイレクトバスのいずれかへの割り当てに基づき、メモリコントローラによる決定が行われる。
【００９１】
実施形態において、システムは少なくとも１つのローカルメモリと、少なくとも１つのシステムメモリと、少なくとも１つのピアメモリとを更に備え、少なくとも１つのピアメモリが専用プロセッサ上に存在し、メモリコントローラが、各メモリのアドレス範囲に基づきいずれのメモリにアクセスするかを決定するよう構成され、メモリコントローラが更に、ホストバス及びダイレクトバスのそれぞれのトラフィック量に基づき動的負荷分散を行なう。
【００９２】
一実施形態はマルチプロセッサシステムにおけるデータ通信方法であって、同方法は、ホストプロセッサ及び複数の専用プロセッサにより共有されるホストバス、又は複数の専用プロセッサ間のダイレクトバスのいずれを介してデータを転送又は受信するかを、転送のメモリアドレス範囲により決定することを含む。
【００９３】
実施形態は、専用プロセッサ上に存在するピアメモリ、ホストバス上のシステムメモリ、及び複数の専用プロセッサ間に結合されたローカルメモリのうちの１つを特定するアドレス範囲を含む。
【００９４】
実施形態において、範囲内でアドレスを特定することは、ホストバス又はダイレクトバスのいずれかに割り当てられるピアメモリを特定することを含む。

【００９５】
一実施形態において、アドレスがピアメモリを特定する範囲内のとき、ホストバス及びダイレクトバスのそれぞれにおけるトラフィック量に基づき決定が行われる。
【００９６】
一実施形態において、方法は更に、動的負荷分散を行って、ホストバス及びダイレクトバスのそれぞれのトラフィックの割合を、トラフィック履歴データに基づき調節することを含む。
【００９７】
一実施形態において、方法は更に、メモリアパーチャをホストプロセッサ及び複数の専用プロセッサの各トラフィックに割り当てることを含み、メモリアパーチャはピアメモリ、システムメモリ、及びローカルメモリに適用され、アパーチャが各プロセッサのメモリの見方を決定する。
【００９８】
一実施形態において、方法は更に、アパーチャ圧縮を行なうことを含み、同圧縮は、オフセットレジスタを用いて、オフセットを変更することにより全てのピアメモリを可視化することを含む。
【００９９】
一実施形態において、インストラクションは、当該方法を遂行するための特定用途向け集積回路（ＡＳＩＣ）の作製に使用することのできるハードウェア記述言語インストラクションから成る。
【０１００】
本明細書に記載のアーキテクチャ及び方法は、実行されるとマルチプロセッサシステムにおけるデータ通信方法を遂行させるインストラクションが記憶されたコンピュータ可読媒体を含む。同方法には、ホストプロセッサ及び複数の専用プロセッサにより共有されるホストバス、又は複数の専用プロセッサ間のダイレクトバスのいずれを介してデータを転送又は受信するかを、転送のメモリアドレスの範囲により決定することが含まれる。
【０１０１】
一実施形態において、アドレスの範囲は、専用プロセッサ上に存在するピアメモリ、ホストバス上のシステムメモリ、及び専用プロセッサに結合されたローカルメモリのうち１つを特定する。
【０１０２】
一実施形態において、範囲内でアドレスを特定することは、ホストバス又はダイレクトバスのいずれかに割り当てられるピアメモリを特定することを含む。
【０１０３】
一実施形態において、アドレスがピアメモリを特定する範囲内にあるとき、ホストバス及びダイレクトバスのそれぞれにおけるトラフィック量に基づき前記決定が行われる。
【０１０４】
一実施形態において、同方法には更に、動的負荷分散を行なって、ホストバス及びダイレクトバスそれぞれのトラフィック割合を、トラフィック履歴データに基づき調整することが含まれる。
【０１０５】
一実施形態において、同方法には更に、ホストプロセッサ及び複数の専用プロセッサのそれぞれのトラフィックにメモリアパーチャを割り当てることが含まれる。メモリアパーチャはピアメモリ、システムメモリ、及びローカルメモリに適用され、アパーチャが各プロセッサのメモリの見方を決定する。
【０１０６】
一実施形態において、同方法には更に、アパーチャ圧縮を行なうことが含まれる。同圧縮は、オフセットレジスタを用いて、オフセットを変更することにより全てのローカルメモリを可視化することを含む。
【０１０７】

一実施形態において、インストラクションは、同方法を遂行するための特定用途向け集積回路（ＡＳＩＣ）の作製に使用することのできるハードウェア記述言語インストラクションから成る。

【０１０８】

本明細書に記載のアーキテクチャ及び方法は更に、マルチプロセッサシステムにおけるデータ通信方法を含む。同方法は、ホストプロセッサ及び複数の専用プロセッサにより共有されるホストバス、又は複数の専用プロセッサ間のダイレクトバスのいずれを介してデータを転送又は受信するかを、転送のメモリアドレス範囲により決定することと、ダイレクトバスを介してプロセッサ間で直接データを通信することとを含む。専用プロセッサはフレームバッファアパーチャを有するグラフィックス処理装置から成り、直接データを通信することには、ダイレクトバスを介してフレームバッファアパーチャ外のアクセスを提供することが含まれる。

【０１０９】

上述の実施形態の態様は、様々な回路のいずれかにプログラミングされた機能として実施することができる。これら回路には、フィールド・プログラマブル・ゲートアレイ（ＦＰＧＡ）、プログラマブル・アレイ論理（ＰＡＬ）素子、電気的プログラマブル・論理／メモリ素子、及び標準的なセルベース素子などのプログラマブル論理素子（ＰＬＤ）と、特定用途向け集積回路（ＡＳＩＣ）や完全カスタムの集積回路が含まれるが、これらには限られない。本実施形態の態様を実現するその他の可能性としては、メモリ（電子的に消去可能なプログラマブル・リードオンリー・メモリ（ＥＥＰＲＯＭ）やフラッシュメモリ等）を有するマイクロコントローラ、組み込み式マイクロプロセッサ、ファームウェア、ソフトウェアなどがある。更に、本実施形態の態様は、ソフトウェアベースの回路エミュレーションを有するマイクロプロセッサ、離散論理回路（順序及び組み合わせ）、カスタムデバイス、ファジィ（ニューラル）論理回路、量子素子、及び上述の種類の素子のハイブリッドにて具現化することができる。基礎となるデバイス技術は様々な種類の構成要素において提供できるのは勿論である。例えば、相補型金属酸化物半導体（ＣＭＯＳ）などの金属酸化物半導体電界効果トランジスタ（ＭＯＳＦＥＴ）技術、エミッタ結合論理回路（ＥＣＬ）などのバイポーラ技術、ポリマー技術（例えばシリコン共役ポリマー及び金属共役ポリマー金属構造）、アナログ／デジタル混合などである。

【０１１０】

本明細書及び特許請求の範囲で使用する「プロセッサ」という用語は、プロセッサコア又はプロセッサの一部を含む。更に、本明細書では１又は複数のＧＰＵ及び１又は複数のＣＰＵに通常は別々に言及しているが、実施形態ではＧＰＵ及びＣＰＵの双方が単一の集積回路パッケージ又は単一のモノリシックダイ上に含まれている。従ってかかる実施形態では、単一のデバイスが、特許請求されている方法を実行する。

【０１１１】

説明及び特許請求の範囲を通じて、文脈上明らかに別のものを要していない限り、「備える」「含む」などの用語は、排他的又は網羅的な意味とは反対の包含的な意味に解釈されるべきである。すなわち、「を含むが、限定されない」の意味に解釈するものとする。単数又は複数を使用する用語も、それぞれ、複数又は単数を含む。加えて、「本明細書において」、「以下に」、「上記」、「下記」という用語及び同様の意味の用語は、本出願で用いる場合、本出願全体を指すのであって本出願のいずれか特定部分を指すのではない。「又は」という用語が、２つ以上の事項を列挙したものに関して使用されているとき、その用語は、当該用語の以下の解釈の全てを包含する。即ち、列挙された事項のいずれか、列挙された事項の全て、及び列挙した事項の任意の組み合わせの全てを包含する。

【０１１２】

方法及びシステムの例示した実施形態の上記説明は、網羅的であることを意図するものでもなければ、開示した厳密な形態に限定することを意図するものでもない。方法及びシステムの具体的な実施形態及び例は、本明細書では例示の目的で説明されているが、当業者であれば認識するように、本発明の範囲内で様々な均等の変更が可能である。本明細書

で提供した開示の教示は他のシステムに適用することができ、上述したようなグラフィックス処理又はビデオ処理を含むシステムについてのみ適用されるものではない。説明した様々な演算は、非常に多様なアーキテクチャにて実行され、説明したものと異なる方法で分配することができる。加えて、本明細書では多くの構成について説明したが、そのいずれも限定的又は排他的であることを意図するものではない。

【０１１３】

その他の実施形態では、本明細書で説明したハードウェア及びソフトウェアの能力の一部又は全てが、プリンタ、カメラ、テレビ、デジタル他用途ディスク（ＤＶＤ）プレーヤ、ＤＶＲ又はＰＶＲ、ハンドヘルド装置、携帯電話又はその他の装置内に存在していてもよい。上述の様々な実施形態の要素及び動作は、更なる実施形態を提供するように組み合わせることができる。これらの変更及び他の変更は、上記の詳細な説明に鑑み、方法及びシステムに対し行うことができる。

【０１１４】

一般に、以下の特許請求の範囲では、使用される用語は、本明細書及び特許請求の範囲で開示される具体的実施形態の方法及びシステムに限定するように解釈されるべきではなく、特許請求の範囲の下で動作するあらゆる処理システム及び処理方法を含むように解釈されるべきである。従って、方法及びシステムは開示によって限定されるものではなく、その代わり、方法及びシステムの範囲は、全面的に特許請求の範囲によって決定されることになる。

【０１１５】

方法及びシステムの一定の態様は、以下の一定の請求項の形で提示されるが、本発明者らは方法及びシステムの様々な態様を、任意の個数の請求項の形で考えている。例えば、方法及びシステムの１つの態様のみがコンピュータ可読媒体において具現化されるものとして記述されるかもしれないが、その他の態様も同様にしてコンピュータ可読媒体において具現化することができる。かかるコンピュータ可読媒体は、演算装置（例えばパーソナルコンピュータ、携帯情報端末、ＰＶＲ、移動装置など）により実行されることになる命令を記憶してもよいし、又は、実行されると動作時に上述の態様を遂行するデバイス（ＧＰＵ、ＡＳＩＣなど）又はソフトウェアアプリケーションを生成するよう設計された命令（例えばＶｅｒｉｌｏｇ又はハードウェア記述言語）を記憶していてもよい。特許請求される発明は、ＧＤＳＩＩデータ（又はその均等物）を生成するために開発、記憶、合成、及び使用されるコンピュータコード（例えばＨＤＬ、Ｖｅｒｉｌｏｇなど）により具現化することができる。次いでこのデータに基づきＡＳＩＣを製造することができる。

【０１１６】

従って、本発明者らは、本願の出願後に追加の請求項を加えて、方法及びシステムの他の態様に対するこのような追加請求項の形式を追求する権利を留保している。

【図１】

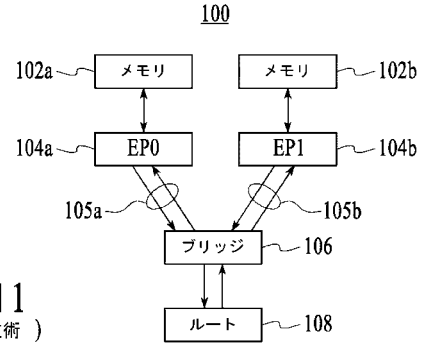


図１
（ 従来技術 ）

【図２】

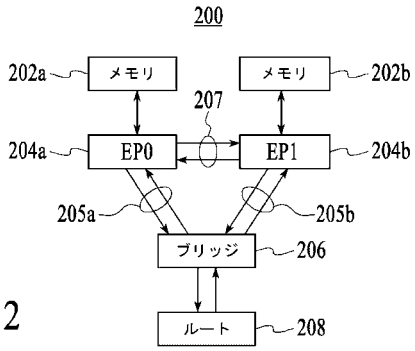


図２

【図３】

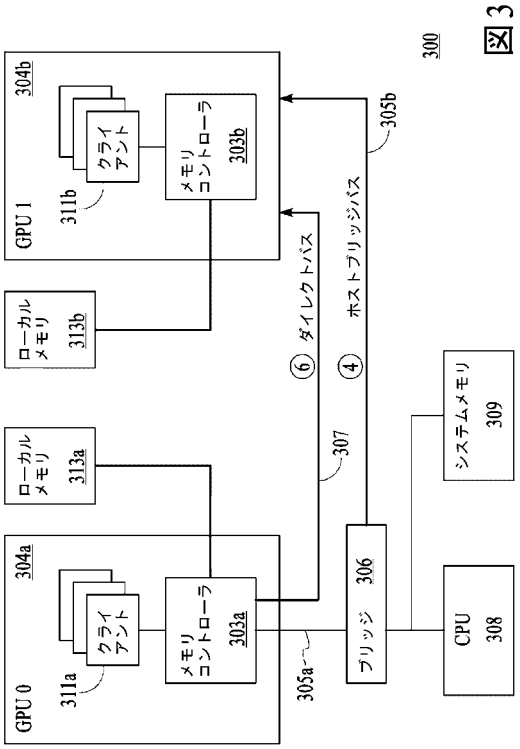


図３

【図４】

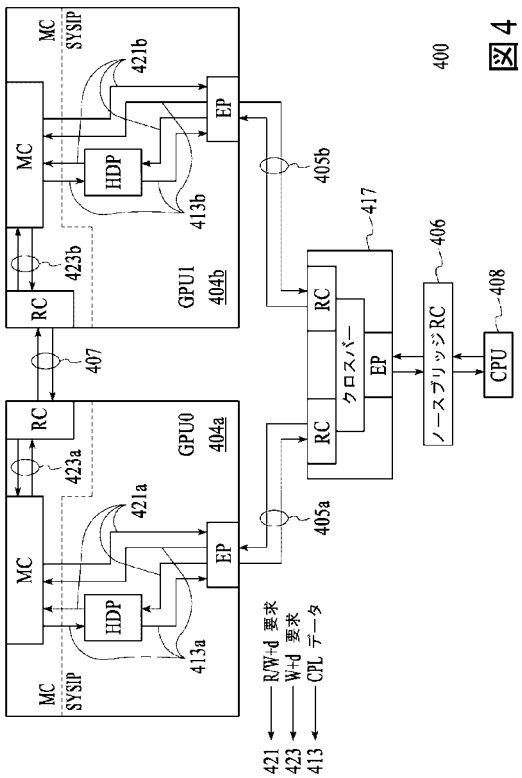


図４

【図５】

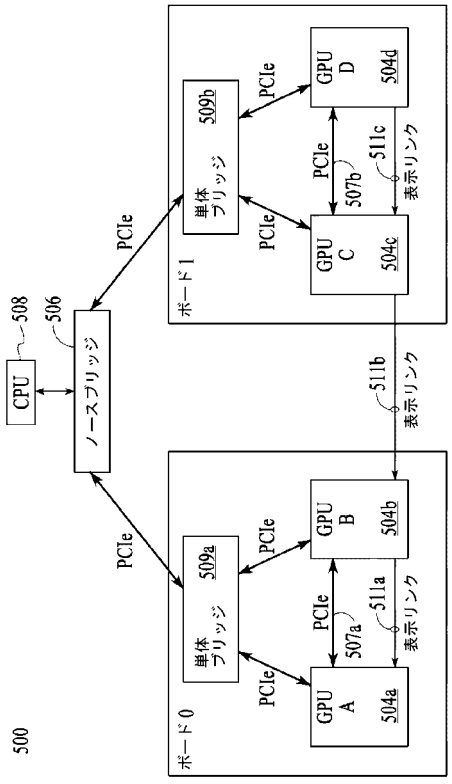


図５

【図６】

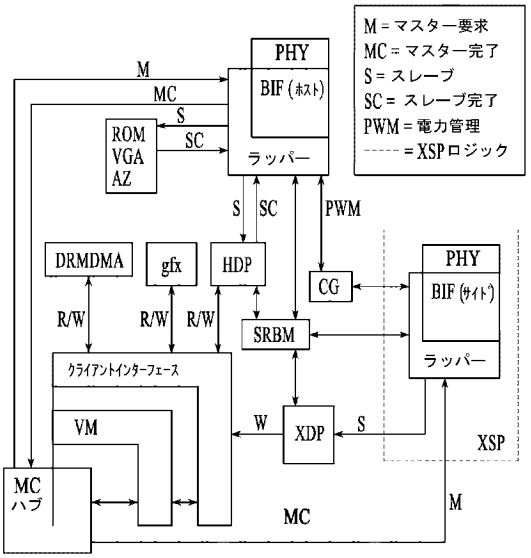

M = マスター要求
MC = マスター完了
S = スレーブ
SC = スレーブ完了
PWM = 電力管理
= XSPロジック
PHY
BIF (ホスト)
ラッパー
M
MC
S
SC
ROM
VGA
AZ
S
SC
PWM
DRMDMA
gfx
HDP
CG
PHY
BIF (サイト)
R/W
R/W
R/W
SRBM
ラッパー
クライアントインターフェース
VM
W
XDP
S
XSP
MC
ハブ
MC
M

図６

【図７】

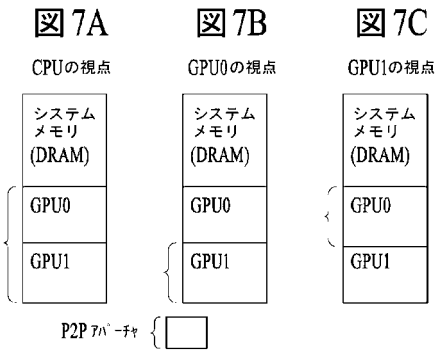


図7A
図7B
図7C
CPUの視点
GPU0の視点
GPU1の視点
システム
メモリ
(DRAM)
GPU0
GPU1
P2P アパーチャ

【図８】

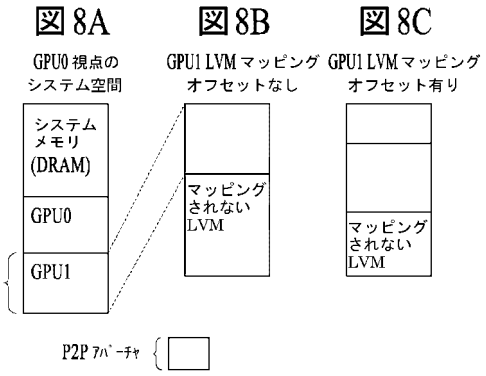
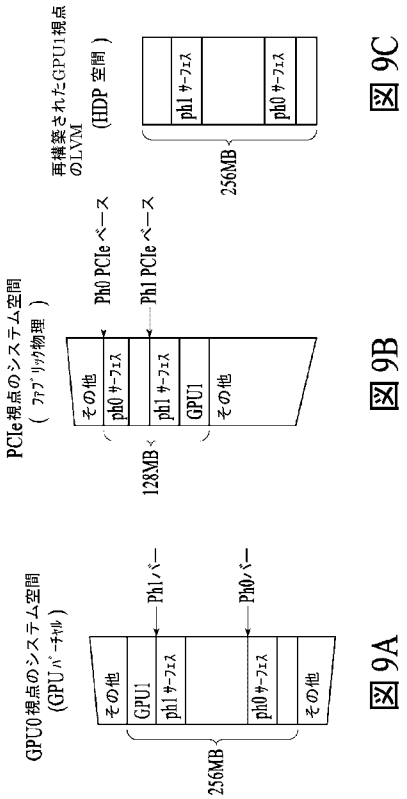


図8A
図8B
図8C
GPU0 視点の
システム空間
GPU1 LVM マッピング
オフセットなし
GPU1 LVM マッピング
オフセット有り
システム
メモリ
(DRAM)
GPU0
GPU1
マッピング
されない
LVM
マッピング
されない
LVM
P2P アパーチャ

【図９】


再構築されたGPU1視点
のLVM
(HDP 空間)
ph1 サーフェス
ph0 サーフェス
256MB
図9C
Ph0 PCIeベース
Ph1 PCIeベース
PCIe視点のシステム空間
(ファブリック物理)
その他
ph0 サーフェス
ph1 サーフェス
GPU1
その他
128MB
図9B
GPU0視点のシステム空間
(GPUバーチャル)
Ph1バー
Ph0バー
その他
GPU1
ph1 サーフェス
ph0 サーフェス
その他
256MB
図9A

フロントページの続き

(72)発明者　グロスマン，マーク，エス
アメリカ合衆国、９４０８８　カリフォルニア州，サニィベイル，ワン　エイエムディ　プレイス（番地なし）
(72)発明者　クルーガー，ウォーレン，フリッツ
アメリカ合衆国、９４０８８　カリフォルニア州，サニィベイル，ワン　エイエムディ　プレイス（番地なし）
(72)発明者　エトシャイト，ブライアン
アメリカ合衆国、９４０８８　カリフォルニア州，サニィベイル，ワン　エイエムディ　プレイス（番地なし）

審査官　原　秀人

(56)参考文献　米国特許第０６４７３０８６（ＵＳ，Ｂ１）
米国特許出願公開第２００７／０２９４４５８（ＵＳ，Ａ１）
特開２００６－０７２５９２（ＪＰ，Ａ）
特開２００７－０２６２８４（ＪＰ，Ａ）

(58)調査した分野(Int.Cl.，ＤＢ名)
Ｇ０６Ｆ　１３／１６
Ｇ０６Ｆ　１２／００
Ｇ０６Ｆ　１３／３６
Ｇ０６Ｆ　１３／３８