(19)日本国特許庁(JP)　(12)特　許　公　報(B2)　(11)特許番号
特許第5996554号
(P5996554)
(45)発行日　平成28年9月21日(2016.9.21)　(24)登録日　平成28年9月2日(2016.9.2)

(51)Int.Cl.　F I
G06F 9/54 (2006.01)　G06F 9/46 480B

請求項の数 38 (全 28 頁)

(21)出願番号	特願2013-549593(P2013-549593)
(86)(22)出願日	平成24年1月13日(2012.1.13)
(65)公表番号	特表2014-505946(P2014-505946A)
(43)公表日	平成26年3月6日(2014.3.6)
(86)国際出願番号	PCT/US2012/021344
(87)国際公開番号	WO2012/097316
(87)国際公開日	平成24年7月19日(2012.7.19)
審査請求日	平成25年9月17日(2013.9.17)
審判番号	不服2015-13892(P2015-13892/J1)
審判請求日	平成27年7月23日(2015.7.23)
(31)優先権主張番号	13/007,333
(32)優先日	平成23年1月14日(2011.1.14)
(33)優先権主張国	米国(US)

(73)特許権者　595020643
クゥアルコム・インコーポレイテッド
ＱＵＡＬＣＯＭＭ　ＩＮＣＯＲＰＯＲＡＴＥＤ
アメリカ合衆国、カリフォルニア州　９２１２１－１７１４、サン・ディエゴ、モアハウス・ドライブ　５７７５
(74)代理人　100108855
弁理士　蔵田　昌俊
(74)代理人　100109830
弁理士　福原　淑弘
(74)代理人　100158805
弁理士　井関　守三
(74)代理人　100194814
弁理士　奥村　元宏

最終頁に続く

(54)【発明の名称】汎用グラフィクス処理装置における計算リソースパイプライン化

(57)【特許請求の範囲】
【請求項１】
汎用グラフィクス処理装置（ＧＰＧＰＵ）であって、
処理パイプラインのステージとして選択的に動作するように構成された前記ＧＰＧＰＵの２以上のプログラム可能な並行処理装置と、
前記並行処理装置間の移送のためにデータを保持するように構成された前記ＧＰＧＰＵの１以上のローカルメモリバッファであって、前記ローカルメモリバッファの各々が前記処理パイプラインにおける前記並行処理装置のうち少なくとも２つの間で直接接続される、１以上のローカルメモリバッファと、
ここにおいて、前記データは、前記ローカルメモリバッファを介して、前記並行処理装置の一方から前記並行処理装置の他方へと直接受け渡される、
前記処理パイプライン内のデータシーケンスを保持するように構成された制御装置であって、データセットのデータスレッドのシーケンスを記録するために前記並行処理装置のうちの少なくとも１つに前記データセットが入るときにシーケンス決定カウンタを実行し、前記シーケンス決定カウンタによって記録されるのと同じシーケンスで前記並行処理装置から前記データセットの前記データスレッドをリリースするために前記並行処理装置のうちの前記少なくとも１つから前記データセットが出るときにシーケンスエンフォーシングバリアを実行するように構成された制御装置と
を備えるＧＰＧＰＵ。
【請求項２】

前記１以上のローカルメモリバッファは、前記並行処理装置間の前記データの移送を可能にするハードウェアベースのデータフロー制御メカニズムを含む、請求項１に記載のＧＰＧＰＵ。

【請求項３】

前記１以上のローカルメモリバッファは、ハードウェアベースの先入れ先出しバッファ（ＦＩＦＯ）、後入れ先出しバッファ（ＬＩＦＯＳ）またはインデクス付けされたバッファのうちの少なくとも１つを備える、請求項１に記載のＧＰＧＰＵ。

【請求項４】

前記１以上のローカルメモリバッファにデータを送信するように前記並行処理装置のうち１以上を構成し、前記１以上のローカルメモリバッファからデータを受信するように前記並行処理装置のうち１以上を構成するように構成された制御装置、

をさらに備える請求項１に記載のＧＰＧＰＵ。

【請求項５】

前記制御装置は、前記ローカルメモリバッファにデータを送信し前記ローカルメモリバッファからデータを受信するように前記並行処理装置を構成するために１以上のアプリケーションプログラミングインタフェース（ＡＰＩ）を実行するように構成される、請求項４に記載のＧＰＧＰＵ。

【請求項６】

前記制御装置は、前記処理パイプラインにおいて前の処理装置からのデータ出力を保持するために前記ローカルメモリバッファの各々についての必要な幅を決定し、前記決定された幅を有するように前記ローカルメモリバッファの各々を構成するようにさらに構成された請求項１に記載のＧＰＧＰＵ。

【請求項７】

前記制御装置は、１以上のアプリケーションプログラミングインタフェース（ＡＰＩ）を実行して、前記ローカルメモリバッファの各々についての前記幅を決定し、前記決定された幅によって前記ローカルメモリバッファの各々を構成し、前記ローカルメモリバッファの各々の深さを決定するように構成される、請求項６に記載のＧＰＧＰＵ。

【請求項８】

前記制御装置は、前記ローカルメモリバッファの各々の深さを決定するようにさらに構成され、前記ローカルメモリバッファの各々は前記深さを前記幅とトレードすることが構成可能である、請求項６に記載のＧＰＧＰＵ。

【請求項９】

前記並行処理装置のうちの１つは、前記処理パイプラインの第１ステージとして動作し、デバイスメモリからオリジナルデータセットを取り出すように構成される、請求項１に記載のＧＰＧＰＵ。

【請求項１０】

前記並行処理装置のうちの１つは、前記処理パイプラインの最終ステージとして動作し、デバイスメモリにパイプライン処理されたデータセットを格納するように構成される、請求項１に記載のＧＰＧＰＵ。

【請求項１１】

前記並行処理装置のうちの少なくとも１つは、前記処理パイプラインの中間ステージとして動作し、前記ローカルメモリバッファのうち一方を介して前記処理パイプラインにおいて前記並行処理装置のうちの前のものからデータセットを受信し、前記ローカルメモリバッファのうち他方を介して前記処理パイプラインにおいて前記並行処理装置のうち後続のものに前記データセットを送信するように構成される、請求項１に記載のＧＰＧＰＵ。

【請求項１２】

前記並行処理装置のうちの前記少なくとも１つは、前記データセットを処理するために、デバイスメモリから補助データを取り出すように構成される、請求項１１に記載のＧＰＧＰＵ。

【請求項１３】

汎用グラフィクス処理装置（ＧＰＧＰＵ）によってデータを処理する方法であって、前記方法は、
処理パイプラインのステージとして選択的に動作するように前記ＧＰＧＰＵの２以上のプログラム可能な並行処理装置を構成することと、
前記並行処理装置間の移送のためのデータを保持するように前記ＧＰＧＰＵの１以上のローカルメモリバッファを構成することであって、なお、前記ローカルメモリバッファの各々は、前記処理パイプラインにおける前記並行処理装置のうち少なくとも２つの間で直接接続され、ここにおいて、前記データは、前記ローカルメモリバッファを介して、前記並行処理装置の一方から前記並行処理装置の他方へと直接受け渡される、構成することと、
前記処理パイプライン内のデータシーケンスを保持することであって、データセットのデータスレッドのシーケンスを記録するために前記並行処理装置のうちの少なくとも１つに前記データセットが入るときにシーケンス決定カウンタを実行することと、前記シーケンス決定カウンタによって記録されるのと同じシーケンスで前記並行処理装置から前記データセットの前記データスレッドをリリースするために前記並行処理装置のうちの前記少なくとも１つから前記データセットが出るときにシーケンスエンフォーシングバリアを実行することとを備える、前記データシーケンスを保持することと、
を備える、方法。
【請求項１４】
前記１以上のローカルメモリバッファは、前記並行処理装置間の前記データの移送を可能にするハードウェアベースのデータフロー制御メカニズムを含む、請求項１３に記載の方法。
【請求項１５】
前記１以上のローカルメモリバッファは、ハードウェアベースの先入れ先出しバッファ（ＦＩＦＯ）、後入れ先出しバッファ（ＬＩＦＯＳ）またはインデクス付けされたバッファのうちの少なくとも１つを備える、請求項１３に記載の方法。
【請求項１６】
前記１以上のローカルメモリバッファにデータを送信するように前記並行処理装置の１以上を構成することと、
前記１以上のローカルメモリバッファからデータを受信するように前記並行処理装置の１以上を構成することと、
をさらに備える請求項１３に記載の方法。
【請求項１７】
前記１以上の並行処理装置を構成することは、前記ローカルメモリバッファにデータを送信し前記ローカルメモリバッファからデータを受信するように前記並行処理装置を構成するために１以上のアプリケーションプログラミングインタフェース（ＡＰＩ）を実行することを備える、請求項１６に記載の方法。
【請求項１８】
前記１以上のローカルメモリバッファを構成することは、
前記処理パイプラインにおいて前の処理装置からのデータ出力を保持するために前記ローカルメモリバッファの各々についての必要な幅を決定することと、
前記決定された幅を有するように前記ローカルメモリバッファの各々を構成することと、
を備える、請求項１３に記載の方法。
【請求項１９】
前記１以上のローカルメモリバッファを構成することは、１以上のアプリケーションプログラミングインタフェース（ＡＰＩ）を実行して、前記ローカルメモリバッファの各々についての前記幅を決定し、前記決定された幅によって前記ローカルメモリバッファの各々を構成し、前記ローカルメモリバッファの各々の深さを決定することを備える、請求項１８に記載の方法。

【請求項２０】
前記１以上のローカルメモリバッファを構成することは、前記ローカルメモリバッファの各々の深さを決定することをさらに備え、前記ローカルメモリバッファの各々は前記深さを前記幅とトレードすることが構成可能である、請求項１８に記載の方法。
【請求項２１】
前記２以上の並行処理装置を構成することは、前記並行処理装置のうちの１つを、前記処理パイプラインの第１ステージとして動作し、デバイスメモリからオリジナルデータセットを取り出すように構成することを備える、請求項１３に記載の方法。
【請求項２２】
前記２以上の並行処理装置を構成することは、前記並行処理装置のうちの１つを、前記処理パイプラインの最終ステージとして動作し、デバイスメモリにパイプライン処理されたデータセットを格納するように構成することを備える、請求項１３に記載の方法。
【請求項２３】
前記２以上の並行処理装置を構成することは、前記並行処理装置のうちの少なくとも１つを、前記処理パイプラインの中間ステージとして動作し、前記ローカルメモリバッファのうち一方を介して前記処理パイプラインにおいて前記並行処理装置のうちの前のものからデータセットを受信し、前記ローカルメモリバッファのうち他方を介して前記処理パイプラインにおいて前記並行処理装置のうち後続のものに前記データセットを送信するように構成することを備える、請求項１３に記載の方法。
【請求項２４】
前記並行処理装置のうちの少なくとも１つを構成することは、前記データセットを処理するために、デバイスメモリから補助データを取り出すように前記並行処理装置のうちの前記少なくとも１つを構成することを備える、請求項２３に記載の方法。
【請求項２５】
汎用グラフィクス処理装置（ＧＰＧＰＵ）であって、
処理パイプラインのステージとして選択的に動作するように前記ＧＰＧＰＵの２以上のプログラム可能な並行処理装置を構成するための手段と、
前記並行処理装置間の移送のためのデータを保持するように前記ＧＰＧＰＵの１以上のローカルメモリバッファを構成するための手段と、なお、前記ローカルメモリバッファの各々は、前記処理パイプラインにおける前記並行処理装置のうち少なくとも２つの間で直接接続され、ここにおいて、前記データは、前記ローカルメモリバッファを介して、前記並行処理装置の一方から前記並行処理装置の他方へと直接受け渡される、構成するための手段と、
前記処理パイプライン内のデータシーケンスを保持するための手段であって、データセットのデータスレッドのシーケンスを記録するために前記並行処理装置のうちの少なくとも１つに前記データセットが入るときにシーケンス決定カウンタを実行するための手段と、前記シーケンス決定カウンタによって記録されるのと同じシーケンスで前記並行処理装置から前記データセットの前記データスレッドをリリースするために前記並行処理装置のうちの前記少なくとも１つから前記データセットが出るときにシーケンスエンフォーシングバリアを実行するための手段とを備える、前記データシーケンスを保持するための手段、
を備えるＧＰＧＰＵ。
【請求項２６】
前記１以上のローカルメモリバッファは、前記並行処理装置間の前記データの移送を可能にするハードウェアベースのデータフロー制御メカニズムを含む、請求項２５に記載のＧＰＧＰＵ。
【請求項２７】
前記１以上のローカルメモリバッファにデータを送信するように前記並行処理装置の１以上を構成するための手段と、
前記１以上のローカルメモリバッファからデータを受信するように前記並行処理装置の

１以上を構成するための手段と、
をさらに備える請求項２５に記載のＧＰＧＰＵ。
【請求項２８】
前記ローカルメモリバッファにデータを送信し前記ローカルメモリバッファからデータを受信するように前記並行処理装置を構成するために１以上のアプリケーションプログラミングインタフェース（ＡＰＩ）を実行するための手段、をさらに備える請求項２７に記載のＧＰＧＰＵ。
【請求項２９】
前記処理パイプラインにおいて前の処理装置からのデータ出力を保持するために前記ローカルメモリバッファの各々についての必要な幅を決定するための手段と、
前記決定された幅を有するように前記ローカルメモリバッファの各々を構成するための手段と、
をさらに備える請求項２５に記載のＧＰＧＰＵ。
【請求項３０】
１以上のアプリケーションプログラミングインタフェース（ＡＰＩ）を実行して、前記ローカルメモリバッファの各々についての前記幅を決定し、前記決定された幅によって前記ローカルメモリバッファの各々を構成し、前記ローカルメモリバッファの各々の深さを決定するための手段をさらに備える請求項２９に記載のＧＰＧＰＵ。
【請求項３１】
前記ローカルメモリバッファの各々の深さを決定するための手段をさらに備え、前記ローカルメモリバッファの各々は前記深さを前記幅とトレードすることが構成可能である、請求項２９に記載のＧＰＧＰＵ。
【請求項３２】
汎用グラフィクス処理装置（ＧＰＧＰＵ）によってデータを処理するための命令を備えるコンピュータ可読媒体であって、前記命令は、実行時に、プログラム可能なプロセッサに、
処理パイプラインのステージとして選択的に動作するように前記ＧＰＧＰＵの２以上のプログラム可能な並行処理装置を構成することと、
前記並行処理装置間の移送のためにデータを保持するように前記ＧＰＧＰＵの１以上のローカルメモリバッファを構成することであって、なお、前記ローカルメモリバッファの各々は、前記処理パイプラインにおける前記並行処理装置のうち少なくとも２つの間で直接接続され、ここにおいて、前記データは、前記ローカルメモリバッファを介して、前記並行処理装置の一方から前記並行処理装置の他方へと直接受け渡される、と、
前記処理パイプライン内のデータシーケンスを保持することと、
を行わせ、前記命令は、前記プログラム可能なプロセッサに、データセットのデータスレッドのシーケンスを記録するために前記並行処理装置のうちの少なくとも１つに前記データセットが入るときにシーケンス決定カウンタを実行することと、前記シーケンス決定カウンタによって記録されるのと同じシーケンスで前記並行処理装置から前記データセットの前記データスレッドをリリースするために前記並行処理装置のうちの前記少なくとも１つから前記データセットが出るときにシーケンスエンフォーシングバリアを実行することとを行わせる、コンピュータ可読媒体。
【請求項３３】
前記１以上のローカルメモリバッファは、前記並行処理装置間の前記データの移送を可能にするハードウェアベースのデータフロー制御メカニズムを含む、請求項３２に記載のコンピュータ可読媒体。
【請求項３４】
前記プログラム可能なプロセッサに、
前記１以上のローカルメモリバッファにデータを送信するように前記並行処理装置の１以上を構成させる命令と、
前記１以上のローカルメモリバッファからデータを受信するように前記並行処理装置の

１以上を構成させる命令と、
をさらに備える請求項３２に記載のコンピュータ可読媒体。
【請求項３５】
前記プログラム可能なプロセッサに、前記ローカルメモリバッファにデータを送信し前記ローカルメモリバッファからデータを受信するように前記並行処理装置を構成するために１以上のアプリケーションプログラミングインタフェース（ＡＰＩ）を実行させる命令、をさらに備える請求項３４に記載のコンピュータ可読媒体。
【請求項３６】
前記プログラム可能なプロセッサに、
前記処理パイプラインにおいて前の処理装置からのデータ出力を保持するために前記ローカルメモリバッファの各々についての必要な幅を決定させる命令と、
前記決定された幅を有するように前記ローカルメモリバッファの各々を構成させる命令と、
をさらに備える請求項３２に記載のコンピュータ可読媒体。
【請求項３７】
前記プログラム可能なプロセッサに、１以上のアプリケーションプログラミングインタフェース（ＡＰＩ）を実行して、前記ローカルメモリバッファの各々についての前記幅を決定し、前記決定された幅によって前記ローカルメモリバッファの各々を構成し、前記ローカルメモリバッファの各々の深さを決定させる命令、をさらに備える請求項３６に記載のコンピュータ可読媒体。
【請求項３８】
前記プログラム可能なプロセッサに、前記ローカルメモリバッファの各々の深さを決定させる命令をさらに備え、前記ローカルメモリバッファの各々は前記深さを前記幅とトレードすることが構成可能である、請求項３６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】
[0001] 本開示は、データを処理することに関し、より具体的には、汎用グラフィクス処理装置を使用してデータを処理することに関する。
【背景技術】
【０００２】
[0002] 汎用グラフィック処理装置（ＧＰＧＰＵ）は、２Ｄおよび３Ｄグラフィックスを処理するように元々設計されたグラフィック処理装置の一般化されたバージョンである。ＧＰＧＰＵは、ＧＰＵのハイパワー並行処理を、グラフィクス処理を超えて汎用データ処理アプリケーションに拡張する。一例として、ＧＰＵは、非グラフィカル計算のためにグラフィクス処理装置に一定のアプリケーションアクセスを与えるＯｐｅｎＣＬ仕様にしたがってデータを処理するように構成されうる。「OpenCL Specification, Version 1.1」は２０１０年６月にリリースされ、公的に入手可能である。
【０００３】
[0003] ＧＰＧＰＵは、処理装置間の同期またはデータ共有を許容しない高並列構造(highly parallel structure)で配置されたプログラム可能な処理装置を含む。代わりに、個々の処理装置は、外部メモリとデータセットのみ交換する。この構造により、ＧＰＧＰＵのためのアプリケーションは、本質的に並行であるものに限定される。ＧＰＧＰＵアーキテクチャは高並列処理されうるので、それらは、パイプラインベース計算の効率的な実装を阻む。この限定は各処理ステージにおける並行処理を使用する２Ｄおよび３Ｄグラフィクス処理に及ぶが、ステージ間の計算リソースのパイプライン化を必要とする。
【発明の概要】
【０００４】
[0004] 本開示は、パイプラインベースのアプリケーションの効率的な処理を許容するために並行処理装置によって汎用グラフィクス処理装置（ＧＰＧＰＵ）のアーキテクチャを

拡張するための技法を説明する。例えば、本技法は、並行処理装置間の移送についてのデータを保持するように処理パイプラインのステージとして動作する並行処理装置に接続されたローカルメモリバッファを構成することを含めうる。ローカルメモリバッファは、並行処理装置間の、オンチップ、低電力、直接なデータ移送を可能にする。ローカルメモリバッファは、並行処理装置間のデータ移送を可能にするためにハードウェアベースのデータフロー制御メカニズムを含めうる。このように、データは、ローカルメモリバッファを介して処理パイプラインにおいて１つの並行処理装置から次の並行処理装置へと直接受け渡され、実際には並行処理装置を一連のパイプラインステージに変換する。ローカルメモリバッファは、処理パイプラインにおける並行処理装置の各々がシステムメモリへの呼び出しを行ない、データを取り出すおよび／または格納する必要性を減らすまたは取り除くことによって、メモリ帯域幅使用量を著しく減らすことができる。

【０００５】

[0005] 本技法は、いくつかの例では、前の並行処理装置からのデータ出力を保持するためにバッファに必要とされる幅を有するようにローカルメモリバッファの各々を構成することを含めうる。例えば、ローカルメモリバッファは、幅と深さを交換することが構成可能であるハードウェアベースのバッファでありうる。さらに、いくつかの例では、本技法は、処理パイプライン内でデータシーケンスをプリザーブ（preserve）するためにシーケンシングバリア(sequencing barriers)を実行することを含めうる。例えば、データセットのデータスレッドのシーケンスは、並行処理装置にデータセットが入るときに記録され、データセットが処理された後で、データセットのデータスレッドは、記録されるのと同じシーケンスで並行処理装置からリリースされうる。

【０００６】

[0006] 一例では、本開示は、処理パイプラインのステージとして選択的に動作するように構成された２以上の並行処理装置と、並行処理装置間の移送のためにデータを保持するように構成された１以上のローカルメモリバッファであって、バッファの各々が並行処理装置のうち少なくとも２つの間で接続される、１以上のローカルメモリバッファと、を備えるＧＰＧＰＵを対象とする。

【０００７】

[0007] 別の例では、本開示は、処理パイプラインのステージとして選択的に動作するように２以上の並行処理装置を構成することと；行処理装置間の移送のためにデータを保持するように１以上のローカルメモリバッファを構成することと、なお、バッファの各々は、並行処理装置のうち少なくとも２つの間で接続される；を備えるＧＰＧＰＵによってタを処理する方法を対象とする。

【０００８】

[0008] さらなる例では、本開示は、処理パイプラインのステージとして選択的に動作するように２以上の並行処理装置を構成するための手段と；並行処理装置間の移送のためにデータを保持するように１以上のローカルメモリバッファを構成するための手段と、なお、バッファの各々は、並行処理装置のうち少なくとも２つの間で接続される；を備えるＧＰＧＰＵを対象とする。

【０００９】

[0009] 別の例では、本開示は、ＧＰＧＰＵによってデータを処理するための命令を備えるコンピュータ可読媒体であって、実行時に、プログラム可能なプロセッサに、処理パイプラインのステージとして選択的に動作するように２以上の並行処理装置を構成させ、並行処理装置間の移送のためにデータを保持するように１以上のローカルメモリバッファを構成させる、なお、バッファの各々が並行処理装置のうち少なくとも２つの間で接続される、コンピュータ可読媒体を対象とする。

【００１０】

[0010] １以上の例の詳細は、添付図面および下記の詳細な説明で記載されている。他の特徴、目的および利点は、明細書および図面、ならびに、特許請求の範囲から明らかとなるであろう。

【図面の簡単な説明】
【００１１】
【図１】図１は、処理パイプラインを実装することが構成可能である汎用グラフィックス処理装置（ＧＰＧＰＵ）を含むデバイスを図示するブロック図である。
【図２】図２は、並行処理を実行するように構成された並行処理装置を含む従来のＧＰＧＰＵを図示するブロック図である。
【図３】図３は、処理パイプラインを実装するように構成されたローカルメモリバッファと並行処理装置を含む図１のＧＰＧＰＵの一例を図示するブロック図である。
【図４】図４は、処理パイプラインのステージとして並行処理装置間でデータを移送するために並行処理装置に接続されたローカルメモリバッファを含むＧＰＧＰＵの例示的な動作を図示するフローチャートである。
【図５】図５は、ＧＰＧＰＵのローカルメモリバッファと並行処理装置によって実装される処理パイプライン内でデータシーケンスをプリザーブする例示的な動作を図示するフローチャートである。
【詳細な説明】
【００１２】
[0016] 本開示は、パイプラインベースのアプリケーションの効率的な処理を許容するために並行処理装置によって汎用グラフィクス処理装置（ＧＰＧＰＵ）のアーキテクチャを拡張するための技法を説明する。具体的には、本技法は、並行処理装置間の移送についてのデータを保持するように処理パイプラインのステージとして動作する並行処理装置に接続されたローカルメモリバッファを構成することを含む。ローカルメモリバッファは、並行処理装置間の、オンチップ、低電力、直接なデータ移送を可能にする。ローカルメモリバッファは、並行処理装置間のデータ移送を可能にするためにハードウェアベースのデータフロー制御メカニズムを含めうる。このように、データは、ローカルメモリバッファを介して処理パイプラインにおいて１つの並行処理装置から次の並行処理装置へと直接受け渡され、実際には並行処理装置を一連のパイプラインステージに変換する。ローカルメモリバッファは、処理パイプラインにおける並行処理装置の各々がシステムメモリへの呼び出しを行ない、データを取り出しおよび／または格納する必要性を減らすまたは取り除くことによって、メモリ帯域幅使用量を著しく減らすことができる。
【００１３】
[0017] 図１は、処理パイプライン１０を実装することが構成可能である汎用グラフィックス処理装置（ＧＰＧＰＵ）６を含むデバイス２を図示するブロック図である。以下でより詳細に説明されるように、ＧＰＧＰＵ６の処理パイプライン１０は、処理パイプライン１０のステージとして動作するように構成された２以上の並行処理装置と、処理パイプライン１０を実装するために並行処理装置間の移送のためにデータを保持するように構成された１以上のローカルメモリバッファとを含む。
【００１４】
[0018] デバイス２は、データを送信および受信すること、様々なデータ処理アプリケーションをサポートすること、および、ユーザへの提示のために処理されたデータを出力すること、が可能である。デバイス２の例は、限定されないが、モバイル無線電話、携帯情報端末（ＰＤＡ）、ビデオゲーミングデバイス、ビデオゲーミングコンソール、ビデオコンファレンシング装置(video conferencing units)、ラップトップコンピュータ、デスクトップコンピュータ、タブレットコンピュータ、テレビセットトップボックス、デジタル記録デバイス、デジタルメディアプレイヤ、および同様なものを含む。
【００１５】
[0019] 図１で図示された例では、デバイス２は、ホストプロセッサ４、処理パイプライン１０を伴うＧＰＧＰＵ６、ディスプレイ８、スピーカ１０、デバイスメモリ１２、トランシーバモジュール１４、およびユーザ入力デバイス１６を含む。他のケースでは、例えば、デバイス２がデスクトップコンピュータである場合、ディスプレイ８、スピーカ１０および／またはユーザインタフェース１６はデバイス２に外付けであってもよい。ホスト

プロセッサ４とＧＰＧＰＵ６は、デジタル信号プロセッサ（ＤＳＰ）、汎用マイクロプロセッサ、特定用途向け集積回路（ＡＳＩＣ）、フィールドプログラム可能なゲートアレイ（ＦＰＧＡ）、または他の同等な集積またはディスクリート論理回路を備えうる。

【００１６】

[0020] ホストプロセッサ４は１以上のアプリケーションを実行しうる。アプリケーションの例は、ウェブブラウザ、ｅメールアプリケーション、スプレッドシート、ビデオゲーム、オーディオおよびビデオ編集アプリケーション、または、ディスプレイ８および／またはスピーカ１０を介したユーザへの提示のためのビジュアルおよび／またはオーディオ出力を生成する他のアプリケーション、を含む。ＧＰＧＰＵ６はまた１以上のアプリケーションを実行しうる。ＧＰＧＰＵ６は、ホストプロセッサ４によって実行されたアプリケーションをサポートしてアプリケーションを実行しうる。具体的には、ＧＰＧＰＵ６は、ディスプレイ８および／またはスピーカ１０を介してユーザへの提示のためのデータを準備するためにアプリケーションを実行しうる。

【００１７】

[0021] ＧＰＧＰＵ６は、グラフィクス処理を超えて汎用データ処理アプリケーションにＧＰＵの高電力並行処理を拡張するグラフィクス処理装置（ＧＰＵ）の一般化されたバージョンである。例として、ＧＰＧＰＵ６は、非グラフィカル計算のためにＧＰＵに一定のアプリケーションアクセスを与えるＯｐｅｎＣＬ仕様にしたがってデータを処理するように構成されうる。従来のＧＰＧＰＵは、図２を参照して以下でさらに詳細に説明され、パイプラインベースのアプリケーションの効率的な実装を防ぐ高並列構造で並べられるプログラム可能な処理装置を含む。この限定は各処理ステージで並行処理を使用する２Ｄおよび３Ｄグラフィクス処理アプリケーションに及ぶが、ステージ間の計算リソースのパイプライン化を必要とする。

【００１８】

[0022] パイプラインベースのアプリケーションは、第１ステージがオリジナルデータセットを処理するように構成され、第２ステージが第１ステージの出力を処理するように構成され、第３ステージが第３ステージの出力を処理するように構成され、アプリケーションに必要とされるステージの数について同様に続くようにステージで処理されるべきデータセットを必要とする。パイプラインベースのアプリケーションの最も効率的な実装は、処理パイプラインにおいてあるステージから次のステージへと直接データセットを受け渡すことである。パイプラインベースのアプリケーションのあまり効率的でない実装は、処理パイプラインにおける各ステージについて、オフチップメモリから前回のステージによって処理されたデータを取り出し、そのあとで、次のステージのためのオフチップメモリに戻って処理されたデータを格納することである。このあまり効率的でない実装はいまだに、データセットが処理パイプラインにおいて各ステージによって正しいシーケンスで処理されることを確実にするシーケンシングメカニズムを必要とする。従来のＧＰＧＰＵは、処理パイプライン、または、パイプラインベースアプリケーションを実行するのに必要なシーケンシングメカニズムさえも実装するように構成されることができない。

【００１９】

[0023] 本開示における技法によれば、また、従来のＧＰＧＰＵとは異なり、いくつかの例において、ＧＰＧＰＵ６は、２Ｄおよび３Ｄグラフィクス処理アプリケーションを含むパイプラインベースのアプリケーションを実行するために処理パイプライン１０を実装することが構成可能である。図３を参照して以下でより詳細に説明されるように、ＧＰＧＰＵ６の処理パイプライン１０は、処理パイプライン１０のステージとして動作するように構成された２以上の並行処理装置と、処理パイプライン１０を実装するために並行処理装置間の移送のためにデータを保持するように構成された１以上のローカルメモリバッファとを含む。処理パイプライン１０に含まれるローカルメモリバッファは、並行処理装置間の、オンチップ、低電力、直接なデータ移送を可能にする。このように、データは、ローカルメモリバッファを介して処理パイプライン１０において１つの並行処理装置から次の並行処理装置へと直接受け渡され、実際には並行処理装置を一連のパイプラインステージ

に変換する。処理パイプライン１０の実装は、処理パイプライン１０における並行処理装置の各々がＧＰＧＰＵ６からオフチップで位置されるデバイスメモリ１２への呼び出しを行ない、データを取り出すおよび／または格納する必要性を減らすまたは取り除くことによって、メモリ帯域幅使用量を著しく減らすことができる。

【００２０】

[0024] 本開示の技法は、前の並行処理装置からのデータ出力を保持するためにバッファに必要とされる幅を有するように処理パイプライン１０内でローカルメモリバッファの各々を構成することを含めうる。例えば、ローカルメモリバッファは、深さを幅と交換することが構成可能であるハードウェアベースのバッファでありうる。さらに、本技法は、処理パイプライン１０内でデータシーケンスをプリザーブするためにシーケンシングバリアを実行することを含む。例えば、データセットのデータスレッドのシーケンスは、データセットが処理パイプライン１０内の並行処理装置に入るときに記録され、データセットが処理された後で、データセットのデータスレッドは、記録されるのと同じシーケンスで並行処理装置からリリースされうる。

【００２１】

[0025] 例えば、ＧＰＧＰＵ６が処理パイプライン１０を実装するように構成されるとき、ＧＰＧＰＵ６は、ウェブブラウザ、ｅメール、ビデオゲーム、およびホストプロセッサ４によって実行されるビデオ編集アプリケーションをサポートして、パイプラインベースの２Ｄおよび３Ｄグラフィクス処理アプリケーションを実行しうる。別の例として、ＧＰＧＰＵ６が処理パイプライン１０を実装するように構成されないとき、ＧＰＧＰＵ６は、画像ベースの探索アプリケーション、画像記述子生成/抽出、ラジオメトリック画像調整(radiometric image adjustments)、オーディオ処理、およびホストプロセッサ４によって一般的に実行される他の動作のような高並行構造で効率的に動作するアプリケーションを実行しうる。

【００２２】

[0026] ある場合には、ＧＰＧＰＵ６が、パイプラインベースのグラフィック処理アプリケーションをサポートしてアプリケーションを実行しうる。パイプラインベースのグラフィクス処理アプリケーションは、処理パイプライン１０を使用するＧＰＧＰＵ６自体によって、または、デバイス２に含まれる別個のＧＰＵによって実行されうる。例えば、ＧＰＧＰＵ６は、画像特殊効果アプリケーション、ＧＰＵパイプラインのための頂点（vertices）生成、およびＧＰＵパイプラインからのカラーバッファを使用するグラフィクスポスト処理アプリケーションを実行しうる。

【００２３】

[0027] ディスプレイ８およびスピーカ１０は双方とも、デバイス２のための出力デバイスを備える。あるケースでは、ディスプレイ８とスピーカ１０は、ユーザにビジュアルおよびオーディオ出力の両方を提示するために一緒に使用されうる。他のケースでは、ディスプレイ８とスピーカ１０は、ユーザに出力を提示するために、別々に使用されうる。例として、ディスプレイ８は、液晶ディスプレイ（ＬＣＤ）、ブラウン管（ＣＲＴ）ディスプレイ、プラズマディスプレイまたは別のタイプのディスプレイデバイスを備えうる。

【００２４】

[0028] ユーザ入力デバイス１６は、デバイス２のための１以上のユーザ入力デバイスを備える。例えば、ユーザ入力デバイス１６は、トラックボール、マウス、キーボード、マイクロフォン、および／または他のタイプの入力デバイスを含めうる。他の例では、ユーザ入力デバイス１６は、タッチスクリーンを備え、ディスプレイ８の一部として組み込まれうる。ユーザは、ユーザ入力デバイス１６を介してホストプロセッサ４および／またはＧＰＧＰＵ６によって実行されるべき１以上のアプリケーションを選択しうる。

【００２５】

[0029] ホストプロセッサ４は、トランシーバモジュール１４を介してホストプロセッサ４および／またはＧＰＧＰＵ６によって処理されるべきデータをダウンロードしうる。ホストプロセッサ４はまた、トランシーバモジュール１４を介してホストプロセッサ４およ

び／またはＧＰＧＰＵ６によって実行される１以上のアプリケーションをダウンロードしうる。トランシーバモジュール１４は、デバイス２と他のデバイスとの間の無線通信または有線通信、またはネットワークを可能にする回路を含めうる。トランシーバモジュール１４は、変調器、復調器、増幅器、および有線通信または無線通信のための他の当該回路を含めうる。

【００２６】

[0030] デバイスメモリ１２は、ホストプロセッサ４および／またはＧＰＧＰＵ６によって処理されるべきデータを格納し、また、ホストプロセッサ４および／またはＧＰＧＰＵ６から受信される処理されたデータを格納しうる。さらに、デバイスメモリ１２は、ホストプロセッサ４および／またはＧＰＧＰＵ６によって実行された１以上のアプリケーションを格納しうる。デバイスメモリ１２は、１以上のコンピュータ可読記憶媒体を備えうる。デバイスメモリ１２の例は、限定されないが、ランダムアクセスメモリ（ＲＡＭ）、読み出し専用メモリ（ＲＯＭ）、電子的に消去可能なプログラム可能な読み出し専用メモリ（ＥＥＰＲＯＭ（登録商標））、ＣＤ－ＲＯＭまたは他の光学ディスクストレージ、磁気ディスクストレージまたは他の磁気ストレージデバイス、フラッシュメモリ、または命令またはデータ構造の形式で所望プログラムコードを搬送または格納するために使用されることができる、また、コンピュータまたはプロセッサによってアクセスされることができる、任意の他の媒体を含む。

【００２７】

[0031] 図２は、並行処理を実行するように構成された並行処理装置２２Ａ－２２Ｄを含む従来のＧＰＧＰＵ１８を図示するブロック図である。いくつかの例では、ＧＰＧＰＵ１８は、図１を参照して上述されているデバイス２と実質的に同様なデバイス内に含まれうる。ＧＰＧＰＵ１８は、データ配信装置２０、並行処理装置２２Ａ－２２Ｄ（「並行処理装置２２」）、およびＧＰＧＰＵ１８に外付けのデバイスメモリ２６に並行処理装置２２を接続するバス２４を含む。

【００２８】

[0032] 従来のＧＰＧＰＵ１８は、２Ｄおよび３Ｄグラフィックスを処理するように元々設計されたＧＰＵの一般化バージョンである。ＧＰＧＰＵ１８は、ＧＰＵの高電力並行処理を、グラフィクス処理を超えて汎用処理アプリケーションに拡張することができる。例として、ＧＰＧＰＵ１８は、ＯｐｅｎＣＬ仕様に従ってデータを処理するように構成されうる。ＯｐｅｎＣＬ仕様は、非グラフィカルコンピューティングのためにＧＰＵに一定のアプリケーションアクセスを与える。ＯｐｅｎＣＬ用語では、データスレッドは作業項目(work item)と呼ばれ、データセットは作業グループ(work group)と呼ばれ、処理装置は計算装置(compute units)と呼ばれ、処理装置の集まりは、計算グループ(compute group)と呼ばれる。

【００２９】

[0033] 一般的なＧＰＵタスクは高度に並行であり、所与の処理装置内で処理されているデータセットのデータスレッド間での情報交換を必要としない。例えば、頂点について計算された値は、異なる頂点について計算された値から独立しており、ピクセルについて計算された値は、異なるピクセルについて計算された値から独立している。ＧＰＵの並行性質を模倣するために、ＧＰＧＰＵ１８は、高並行構造で配列された並行処理装置２２を含むように設計される。

【００３０】

[0034] ＧＰＧＰＵ１８のアーキテクチャは、並行処理装置２２間のデータ共有または同期を許容しないほど、高並行である。動作において、データ配信装置２０は、並行処理装置２２の各々に、デバイスメモリ２６に格納されたデータセットを割り当てる。処理中、割り当てられたデータセットのデータスレッドは、並行処理装置２２の各々の内で共有され同期されうる。しかしながら、異なるデータセットのデータスレッドは、並行処理装置２２間で共有または同期されることができない。代わりに、並行処理装置２２の各々は、バス２４を介してデバイスメモリ２６と割り当てられたデータセットのみ交換する。より

具体的には、並行処理装置２２の各々は、バス２４を介してデバイスメモリ２６から処理についての割り当てられたデータセットを取り出し、データセットを処理した後で、バス２４を介してデバイスメモリ２６に戻って処理されたデータセットを格納する。

【００３１】

[0035] ＧＰＧＰＵ１８の並行アーキテクチャは、並行処理装置２２間のパイプラインベースアプリケーションの効率的な実装を阻む。パイプラインベースのアプリケーションでは、処理装置は、異なる処理タスクについて１つのステージから別のステージへとデータが移動することを可能にするためにパイプラインにおいてステージとして接続される。ＧＰＧＰＵ１８におけるパイプラインベースのアプリケーションに対する限定は、２Ｄおよび３Ｄグラフィクス処理アプリケーションに拡張する、そしてそれは各処理ステージで並行処理を使用するが、ステージ間でのパイプライン化を必要とする。

【００３２】

[0036] したがって、ＧＰＧＰＵ１８のアプリケーションは、本質的に並行であるものに限定される。並行処理装置２２の各々は算術論理装置（ＡＬＵ）のクラスタまたは他の構成可能な論理素子を備えうる。したがって、並行処理装置２２は、ＧＰＧＰＵ１８によって実行されるアプリケーションに依存して異なる動作を実行することがプログラム可能または構成可能である。ＧＰＧＰＵ１８の高並行構造で効率的に動作するアプリケーションは、画像ベースの探索アプリケーション、画像記述子生成/抽出、ラジオメトリック画像調整(radiometric image adjustments)、オーディオ処理、およびデジタル信号プロセッサ（ＤＳＰ）によって一般的に実行される他の動作および同様なものを含めうる。さらに、ＧＰＧＰＵ１８によって実行されるアプリケーションは、画像特殊効果生成、ＧＰＵパイプラインのための頂点生成、ＧＰＵパイプラインからのカラーバッファを使用してグラフィクスポスト処理動作のようなパイプラインベースのグラフィクス処理アプリケーションとのインタラクションを必要としうる。

【００３３】

[0037] 図３は、図１の例示的なＧＰＧＰＵ６を図示するブロック図であり、処理パイプライン１０を実装するように構成されたローカルメモリバッファ４４Ａ－４４Ｃと並行処理装置４２Ａ－４２Ｄとを含む。他の例では、ＧＰＧＰＵ６は、より多数またはより少数の並行処理装置およびローカルメモリバッファを含めうる。

【００３４】

[0038] 図３の例では、ＧＰＧＰＵ６は、データ配信装置４０、並行処理装置４２Ａ－４２Ｄ（「並行処理装置４２」）およびＧＰＧＰＵ６に外付けのデバイスメモリ１２（図１）に並行処理装置４２を接続するバス４６を含む。従来のＧＰＧＰＵとは異なり（例えば、図３のＧＰＧＰＵ１８）、ＧＰＧＰＵ６はまた、並行処理装置４２間で接続されたローカルメモリバッファ４４Ａ－４４Ｃ（「ローカルメモリバッファ４４」）を含む。並行処理装置４２と並行処理装置４２間で接続されたローカルメモリバッファ４４の組み合わせは、処理パイプライン１０と呼ばれうる。ＧＰＧＰＵ６はまた、制御装置３０およびローカルメモリ３８を含む。ローカルメモリ３８は、ローカルメモリバッファ４４に類似したバッファ、レジスタ、またはＧＰＧＰＵ６のデータを一時的に格納するキャッシュを備えうる。制御装置３０は、アプリケーションプログラミングインタフェース（ＡＰＩ）３２、バッファマネージャ３４、およびシーケンスマネージャ３６を含む。

【００３５】

[0039] ローカルメモリバッファ４４は、並行処理装置４２間のデータ移送を可能にするハードウェアベースのデータフロー制御メカニズムを含めうる。例えば、ローカルメモリバッファ４４は、ハードウェアベースの先入れ先出し（ＦＩＦＯ）バッファ、後入れ先出し（ＬＩＦＯ）バッファまたはインデクス付けされたバッファのような他のタイプのハードウェアベースのバッファを備えうる。ローカルメモリバッファ４４ＡがハードウェアベースのＦＩＦＯを備える場合には、例えば、ローカルメモリバッファ４４Ａは、バッファにデータを書き込むスペースがあるときローカルメモリバッファ４４Ａへデータを並行処理装置４２Ａが送信し、そうでないときには書き込み要求をストールすることを可能にす

るデータフロー制御メカニズムを含む。その場合、ローカルメモリバッファ４４Ａはまた、バッファから読み出すのに利用可能なデータがあるとき、ローカルメモリバッファ４４Ａからデータを並行処理装置４２Ｂが受信し、そうでないときには読み出し要求をストールすることを可能にするデータフロー制御メカニズムを含む。ローカルメモリバッファ４４がハードウェアベースのデータフロー制御メカニズムを含むとき、あまり効率的でないソフトウェアベースのデータフロー制御は、並行処理装置４２間のデータの移送を可能にするのに必要ではない。

【００３６】

[0040] ローカルメモリバッファ４４は、並行処理装置４２間の、オンチップ、低電力、直接なデータ移送を可能にする。ローカルメモリバッファ４４は「ローカル」である、なぜならば、それらは、ＧＰＧＰＵ６内で、処理装置４２と同じチップ上で位置されるからである。このように、データは、ローカルメモリバッファ４４を介して処理パイプライン１０において並行処理装置４２の一方から並行処理装置４２の他方へと直接受け渡されうる。並行処理装置４２は、ＧＰＧＰＵ６に外付けであるまたはＧＰＧＰＵ６からオフチップに配置されているデバイスメモリ１２でデータを繰り返し取り出し格納することを必要としていない。したがって、ローカルメモリバッファ４４は、並行処理装置４２を一連のパイプラインステージに変換し、ＧＰＧＰＵ６内で処理パイプライン１０を実装する。

【００３７】

[0041] 図示された例では、ローカルメモリバッファ４４の各々は、処理パイプライン１０が純粋に直列なパイプラインであるように連続順で並行処理装置４２の２つの間で直接接続される。ローカルメモリバッファ４４は、それが２つの並行処理装置４２によってのみアクセス可能であるように、それらが接続され並行処理装置４２のいずれによってもアドレス可能なバスに、「直接」接続される。例えば、ローカルメモリバッファ４４Ａは、並行処理装置４２Ａおよび４２Ｂとの間で直接接続され、ローカルメモリバッファ４４Ｂは、並行処理装置４２Ｂと４２Ｃとの間で直接接続され、ローカルメモリバッファ４４Ｃは、並行処理装置４２Ｃと４２Ｄとの間で直接接続される。

【００３８】

[0042] 他の例では、メモリバッファ４４の各々はまた、連続順でない並行処理装置４２のうち１以上に直接接続されうる。この場合、ローカルメモリバッファ４４の各々は、クロスバー接続を介して並行処理装置４２のいずれかに直接接続されうる。例えば、ローカルメモリバッファ４４Ａは、並行処理装置４２Ａがローカルメモリバッファ４４Ａを介して並行処理装置４２Ｂ－４２Ｄのいずれかにデータを移送しうるように、クロスバー接続を介して並行処理装置４２の各々に直接接続されうる。クロスバー接続の使用は、ローカルメモリバッファ４４を並行処理装置４２に対してより幅広くアクセス可能にし、純粋に直列ではない処理パイプラインの実装を可能にする。

【００３９】

[0043] 処理パイプライン１０が純粋に直列なパイプラインを備える図示された例では、並行処理装置４２は、ローカルメモリバッファ４４の次のもの（successive one）にデータを書き込む許可のみを有し、ローカルメモリバッファ４４の前のものからデータを読み出す許可のみを有しうる。例えば、並行処理装置４２Ｂは、ローカルメモリバッファ４４Ａからデータを読み出すことのみ可能であり、ローカルメモリバッファ４４Ｂにデータを書き込むことのみ可能でありうる。処理パイプラインがクロスバー接続を含める場合、並行処理装置４２は、ローカルメモリバッファ４４のいずれかに読み出し且つ書き込む許可を有しうる。例えば、並行処理装置４２Ｂは、ローカルメモリバッファ４４Ａで、また、ローカルメモリバッファ４４Ｂで、データを読み出し書き込むことが可能であることがある。

【００４０】

[0044] 上述されるように、ローカルメモリバッファ４４は、ＦＩＦＯバッファ、ＬＩＦＯバッファ、またはインデクス付けされたバッファのうちの少なくとも１つを備えうる。ローカルメモリバッファ４４に使用されるバッファのタイプは、処理パイプライン１０で

必要とされるハードウェアベースのデータフロー制御メカニズムのタイプに依存しうる。ローカルメモリバッファ４４に使用されるバッファのタイプはまた、ローカルメモリバッファ４４が１対１接続またはクロスバー接続を介して並行処理装置４２に接続されるかに依存しうる。さらに、クロスバー接続が使用されるとき、制御装置３０のバッファマネージャ３４は、どの並行処理装置４２が所与時間にどのローカルメモリバッファ４４にアクセスするかを管理するために、いくらかのメモリ制御を実行する必要があることがある。

【００４１】

[0045] 上述されるように、ローカルメモリバッファ４４は、１対１またはクロスバー接続のいずれかを介して並行処理装置４２の少なくとも２つの間で直接接続されうる。しかしながら、ローカルメモリバッファ４４は、並行処理装置４２によってアドレス可能なバスでないことがある。このように、ローカルメモリバッファ４４の指定されたメモリコントローラは必要でないことがある。具体的には、メモリコントローラは、バスにわたってローカルメモリバッファ４４に対して読み出しおよび書き込みコマンドを処理する必要はない。

【００４２】

[0046] ローカルメモリバッファ４４は、並行処理装置４２の各々がバス４６を介してデバイスメモリ１２への呼び出しを行ない、データを取り出すおよび／または格納する必要性を減らすまたは取り除くことによって、メモリ帯域幅使用量を著しく減らすことができる。動作において、並行処理装置４２Ａは、処理パイプライン１０の第１の処理装置として、バス４６を介してデバイスメモリ１２からオリジナルデータセットを取り出す。データセットは、データ配信装置４０によって並行処理装置４２Ａに割り当てられうる。さらに、並行処理装置４２Ｄは、処理パイプライン１０の最終処理装置として、バス４６を介してデバイスメモリ１２にポストパイプラインデータセットを格納する。並行処理装置４２Ｂおよび４２Ｃは、処理パイプライン１０の中間処理装置として、ローカルメモリバッファ４４のうち一方を介して並行処理装置４２のうちの前のものからデータセットを受信し、ローカルメモリバッファ４４のうち他方を介して並行処理装置４２のうち後続のものにデータセットを送信する。したがって、中間処理装置は、データを取り出しおよび／または格納するためにデバイスメモリ１２と相互作用することを必要とされない。いくつかの場合では、中間処理装置は、処理パイプライン１０の特定のステージを実行するためにデバイスメモリから補助データを取り出しうる。しかしながら、処理用の主要なデータセットは、ローカルメモリバッファ４４を介して処理パイプライン１０に沿って直接受け渡される。

【００４３】

[0047] 上述されるように、ＧＰＧＰＵ６は、グラフィクス処理を超えて汎用データ処理アプリケーションにＧＰＵの高電力並行処理を拡張するＧＰＵの一般化されたバージョンである。例として、ＧＰＧＰＵ６は、非グラフィカル計算のためにグラフィクス処理装置に一定のアプリケーションアクセスを与えるＯｐｅｎＣＬ仕様にしたがってデータを処理するように構成されうる。ＯｐｅｎＣＬ用語では、データスレッドは作業項目（work item)と呼ばれ、データセットは作業グループ(work group)と呼ばれ、処理装置は計算装置(compute units)と呼ばれ、処理装置の集まりは、計算グループ(compute group)と呼ばれる。

【００４４】

[0048] 本開示の技法によれば、ＧＰＧＰＵ６は、２Ｄおよび３Ｄグラフィクス処理アプリケーションを含むパイプラインベースのアプリケーションを実行するために処理パイプライン１０を実装することが構成可能である。より具体的には、ＧＰＧＰＵ６の制御装置３０は、処理パイプラインのステージとして動作するように並行処理装置４２を構成する。制御装置３０はまた、並行処理装置４２間の移送のためのデータを保持するように、並行処理装置４２間で接続されたローカルメモリバッファ４４を構成する。

【００４５】

[0049] 並行処理装置４２は、ＧＰＧＰＵ６によって実行されるアプリケーションに依存

して異なる動作を実行することがプログラム可能または構成可能でありうる。制御装置３０は、アプリケーションにしたがって動作するように並行処理装置４２の各々を構成しうる。例えば、並行処理装置２２の各々は算術論理装置（ＡＬＵ）のクラスタまたは他の構成可能な論理素子を備えうる。

【００４６】

[0050] ローカルメモリバッファ４４はまた、ＧＰＧＰＵ６によって実行されるアプリケーションに依存して並行処理装置４２からの異なるタイプのデータ出力を保持することがプログラム可能または構成可能でありうる。例えば、ローカルメモリバッファ４４は、ハードウェアベースのバッファを備えうるが、構成可能な態様のセット（a set if configurable aspects）を含めうる。構成可能な態様の１つは、並行処理装置４２からの異なるタイプのデータ出力を適応させるためのローカルメモリバッファ４４の幅でありうる。例えば、ローカルメモリバッファ４４は、深さを幅とトレードすることが構成可能でありうる。制御装置３０のバッファマネージャ３４は、並行処理装置４２のうち前のもののデータ出力を保持するためにローカルメモリバッファ４４の各々に必要とされる幅を決定しうる。バッファマネージャ３４は、並行処理装置４２の各々からデータ出力のタイプを認識するので、データを保持するためにローカルメモリバッファ４４の各々によって必要とされる幅を認識する。バッファマネージャ３４は、そのあとで、決定された幅を有するようにローカルメモリバッファ４４の各々を構成しうる。

【００４７】

[0051] いったん並行処理装置４２とローカルメモリバッファ４４がＧＰＧＰＵ６内で処理パイプライン１０を実装するように構成されると、並行処理装置４２は、ローカルメモリバッファ４４を介してデータを移送しうる。制御装置３０は、ローカルメモリバッファ４４にデータを送信するように並行処理装置４２のうち１以上を構成し、ローカルメモリバッファ４４からデータを受信するように並行処理装置４４のうち１以上を構成しうる。例えば、制御装置３０は、それぞれ、ローカルメモリバッファ４４Ａ、４４Ｂ、および４４Ｃにデータを送信するように並行処理装置４２Ａ、４２Ｂおよび４２Ｃを構成しうる。制御装置３０はまた、それぞれ、ローカルメモリバッファ４４Ａ、４４Ｂ、および４４Ｃからデータ受信するように並行処理装置４２Ｂ、４２Ｃ、および４２Ｄを構成しうる。

【００４８】

[0052] ハードウェアベースのフロー制御メカニズムを有するローカルメモリバッファ４４は、新規ＡＰＩ３２を導入することによって、ＯｐｅｎＣＬ規格のようなＧＰＧＰＵ規格を使用して露出されうる(exposed)。例えば、制御装置３０は、ＡＰＩ３２の１以上を実行して、ローカルメモリバッファ４４の各々に必要とされる幅を決定し、決定された幅でローカルメモリバッファ４４の各々を構成し、ローカルメモリバッファ４４の各々の深さを決定しうる。さらに、制御装置３０は、ＡＰＩ３２の１以上を実行してローカルメモリバッファ４４にデータを送信しローカルメモリバッファ４４からデータを受信するように並行処理装置４２を構成しうる。ローカルメモリバッファ４４に含まれるハードウェアベースのデータフロー制御メカニズムは、並行処理装置４２が、さらなるソフトウェアベースのデータフロー制御なしに、ローカルメモリバッファ４４にデータを送信し、ローカルメモリバッファ４４からデータを受信することを可能にする。

【００４９】

[0053] さらにＧＰＧＰＵ６の制御装置３０は、並行処理装置４２のうち１以上内でデータシーケンスをプリザーブすることによって処理パイプライン１０内でデータシーケンスをプリザーブしうる。ＧＰＧＰＵ６によって実行されるパイプラインベースのアプリケーション、特に３Ｄグラフィクスアプリケーションは、処理パイプライン１０内で一定のシーケンスで処理されるべきデータを必要としうる。データが処理パイプラインの各ステージで処理されるとき、データは、条件、キャッシュヒットまたはミス、および同様なもののような実行課題（execution issues）に起因してシーケンスを変更しうる。制御装置３０のシーケンスマネージャ３６は、並行処理装置４２の少なくともいくつかの内でデータシーケンスをプリザーブするためにシーケンシングバリアを実行しうる。シーケンシング

バリアは、処理パイプライン１０内の処理速度を減速させることができるので、シーケンスマネージャ３６は、精確な処理のためにデータシーケンスプリザベーションを必要とするこれらの並行処理装置４２においてシーケンシングバリアのみを実行しうる。

【００５０】

[0054] シーケンスマネージャ３６によって実行されるシーケンシングバリアは、シーケンス決定カウンタ（ＳＤＣ）およびシーケンスエンフォーシングバリア（ＳＥＢ）を含めうる。例えば、シーケンシングバリアは、ＳＤＣおよびＳＥＢについてＯｐｅｎＣＬ Ｃ言語に新しい関数呼び出しを追加することによって、ＯｐｅｎＣＬ規格のようなＧＰＧＰＵ規格を使用して露出されうる(exposed)。

【００５１】

[0055] シーケンスマネージャ３６は、データセットが並行処理装置４２のいずれか１つに入るときＳＤＣを実行しうる。シーケンスマネージャ３６は、そのあとで、ローカルメモリ３８内で受信されたデータセットのデータスレッドのシーケンスを記録することによってＳＤＣ動作を実行する。例えば、シーケンスマネージャ３６は、データスレッドがデバイスメモリ１２から受信される順でデータセットの各データスレッドのインデクスを記録しうる。

【００５２】

[0056] シーケンスマネージャ３６は、データセットが並行処理装置４２のうちの１つから出るときに、ＳＥＢを実行しうる。シーケンスマネージャ３６は、そのあとで、ＳＤＣによって記録されるのと同じシーケンスで並行処理装置４２のうちの１つからデータセットのデータスレッドをリリースすることによってＳＥＢ動作を実行する。例えば、シーケンスマネージャ３６は、ローカルメモリ３８に記録されたデータスレッドインデクスにアクセスし、インデクスが記録された順にしたがって各データスレッドをリリースする。このように、データセットのデータスレッドは、データセットのデータスレッドが並行処理装置４２のうち現在のものに入るのと同じ順で並行処理装置４２の後続のものに入るであろう。

【００５３】

[0057] 一例では、制御装置３０は、パイプラインベースの３Ｄグラフィクス処理アプリケーションを実行するようにＧＰＧＰＵ６を構成しうる。その場合、制御装置３０は、３Ｄグラフィクス処理パイプラインのステージとして動作するように並行処理装置４２を構成しうる。例えば、制御装置３０は、頂点シェーダとして動作するように並行処理装置４２Ａを構成し、トライアングルラステライザとして動作するように並行処理装置４２Ｂを構成し、フラグメントシェーダとして動作するように並行処理装置４２Ｃを構成し、ピクセルブレンダとして動作するように並行処理装置４２Ｄを構成しうる。

【００５４】

[0058] 制御装置３０はまた、３Ｄグラフィクス処理パイプライン１０を実装するために並行処理装置４２間の移送のためのデータを保持するようにハードウェアベースのデータフロー制御メカニズムによってローカルメモリバッファ４４を構成しうる。例えば、制御装置３０は、頂点シェーダとして動作する並行処理装置４２Ａと、トライアングルラステライザとして動作する並行処理装置４２Ｂとの間の移送のためのポスト頂点シェーダの頂点データを保持するようにローカルメモリバッファ４４Ａを構成しうる。制御装置３０は、トライアングルラステライザとして動作する並行処理装置４２Ｂと、フラグメントシェーダとして動作する並行処理装置４２Ｃとの間の移送のためにプレフラグメントシェーダピクセルデータを保持するようにローカルメモリバッファ４４Ｂを構成しうる。最後に、制御装置３０は、フラグメントシェーダとして動作している並行処理装置４２Ｃとピクセルブレンダとして動作している並行処理装置４２Ｄとの間の移送のためにポストフラグメントシェーダピクセル値を保持するようにローカルメモリバッファ４４Ｃを構成しうる。

【００５５】

[0059] ３Ｄグラフィクス処理アプリケーションを実行するとき、データ配信装置４０は、頂点シェーダとして動作している並行処理装置４２Ａにオリジナル頂点データセットを

割り当てうる。並行処理装置４２Ａは、バス４６を介して、デバイスメモリ１２から、割り当てられたオリジナル頂点データセットを取り出す。データセットが入るとき、シーケンスマネージャ３６は、頂点データのシーケンスを記録するためにＳＤＣを実行する。並行処理装置４２Ａはそのあとで頂点シェーディング動作を実行し、ローカルメモリバッファ４４Ａにポスト頂点シェーダの頂点データを送信する。データセットが並行処理装置４２Ａから出るとき、シーケンスマネージャ３６は、ＳＤＣによって記録されるのと同じシーケンスで頂点データをリリースするためにＳＥＢを実行する。このように、頂点データは、頂点シェーダとして動作する並行処理装置４２Ａに頂点データが入ったのと同じ順で、トライアングルラステライザとして動作する並行処理装置４２Ｂに到達するであろう。

【００５６】

[0060] トライアングルラステライザとして動作する並行処理装置４２Ｂは、ローカルメモリバッファ４４Ａからポスト頂点シェーダの頂点データを受信する。いくつかの場合においては、並行処理装置４２Ｂはまた、トライアングルラステライズ化動作を実行するためにバス４６を介してデバイスメモリ１２から補助データを取り出しうる。並行処理装置４２Ｂはそのあとでトライアングルラステライズ化動作を実行し、ローカルメモリバッファ４４Ｂにプレフラグメントシェーダピクセルデータを送信する。いくつかの例では、シーケンスマネージャ３６は、頂点データが並行処理装置４２Ｂに入るときＳＤＣを実行し、ピクセルデータが並行処理装置４２Ｂから出るときＳＥＢを実行してデータシーケンスをプリザーブしうる。他の例では、シーケンシングバリアは必須ではないので、並行処理装置４２Ｂに対して実行されない。

【００５７】

[0061] 並行処理装置４２Ｃは、フラグメントシェーダを動作し、ローカルメモリバッファ４４Ｂからプレフラグメントシェーダピクセルデータを受信する。データセットが入るとき、シーケンスマネージャ３６は、ピクセルデータのシーケンスを記録するためにＳＤＣを実行する。いくつかの場合においては、並行処理装置４２Ｃはまた、フラグメントシェーダ動作を実行するためにバス４６を介してデバイスメモリ１２から補助データを取り出しうる。並行処理装置４２Ｃはそのあとでフラグメントシェーディング動作を実行し、ポストフラグメントシェーダピクセル値をローカルメモリバッファ４４Ｃに送信する。データセットが並行処理装置４２Ｃから出るとき、シーケンスマネージャ３６は、ＳＤＣによって記録されるのと同じシーケンスでピクセルデータをリリースするためにＳＥＢを実行する。このように、ピクセルデータは、フラグメントシェーダとして動作している並行処理装置４２Ｃにピクセルデータが入ったのと同じ順で、ピクセルブレンダとして動作する並行処理装置４２Ｄに到達するであろう。

【００５８】

[0062] 並行処理装置４２Ｄは、ピクセルブレンダとして動作し、ローカルメモリバッファ４４Ｃからポストフラグメントシェーダピクセル値を受信する。並行処理装置４４Ｄは、ピクセルブレンディング動作を実行し、バス４６を介してデバイスメモリ１２にポストパイプラインデータセットを格納する。いくつかの例では、シーケンスマネージャ３６は、ピクセルデータが並行処理装置４２Ｄに入るときＳＤＣを実行し、画像データが並行処理装置４２Ｄから出るときＳＥＢを実行してデータシーケンスをプリザーブしうる。他の例では、シーケンシングバリアは必須ではないので、並行処理装置４２Ｄに対して実行されない。３Ｄグラフィクス処理アプリケーションの上述された例は、単なる例示であり、開示された技法は、ＧＰＧＰＵ６において様々なパイプラインベースのアプリケーションを実行するために使用されうる。

【００５９】

[0063] 図４は、処理パイプライン１０のステージとしての並行処理装置間でデータを移送するために並行処理装置４２に接続されたローカルメモリバッファ４４を含むＧＰＧＰＵ６の例示的な動作を図示するフローチャートである。図示される動作は、図３のＧＰＧＰＵ６を参照して説明される。

【００６０】

[0064] ＧＰＧＰＵ６の制御装置３０は、処理パイプライン１０のステージとして動作するように並行処理装置４２を構成する（５０）。例えば、制御装置３０は、３Ｄグラフィクス処理パイプラインのステージとして動作するように並行処理装置４２を構成する。その例では、制御装置３０は、頂点シェーダとして動作するように並行処理装置４２Ａを構成し、トライアングルラステライザとして動作するように並行処理装置４２Ｂを構成し、フラグメントシェーダとして動作するように並行処理装置４２Ｃを構成し、ピクセルブレンダとして動作するように並行処理装置４２Ｄを構成しうる。

【００６１】

[0065] 制御装置３０はまた、並行処理装置４２間の移送のためのデータを保持するようにローカルメモリバッファ４４を構成し、結果、並行処理装置４２を処理パイプライン１０に変換する（５２）。ローカルメモリバッファ４４は、並行処理装置４２間のデータ移送を可能にするためにハードウェアベースのデータフロー制御メカニズムを含めうる。例えば、ローカルメモリバッファ４４は、ハードウェアベースのＦＩＦＯ、ＬＩＦＯ、またはインデクス付けされたバッファを備えうる。ローカルメモリバッファ４４は、並行処理装置４２の少なくとも２つの間で直接接続されうる。例えば、３Ｄグラフィックス処理パイプラインの場合、ローカルメモリバッファ４４Ａは、頂点シェーダとして動作する並行処理装置４２Ａと、トライアングルラステライザとして動作する並行処理装置４２Ｂとの間で直接接続され、ポスト頂点シェーダの頂点データ(post-vertex shader vertex data)を保持するように構成されうる。ローカルメモリバッファ４４Ｂは、トライアングルラステライザとして動作する並行処理装置３２Ｂと、フラグメントシェーダとして動作する並行処理装置４２Ｃとの間で直接接続され、プレフラグメントシェーダのピクセルデータを保持するように構成されうる。最後に、ローカルメモリバッファ４４Ｃは、フラグメントシェーダとして動作する並行処理装置４２Ｃと、ピクセルブレンダとして動作する並行処理装置４２Ｄとの間で直接接続され、ポストフラグメントシェーダピクセル値を保持するように構成されうる。

【００６２】

[0066] さらに、制御装置３０のバッファマネージャ３４は、並行処理装置４２のうち前のものからのデータ出力を保持するためにローカルメモリバッファ４４の各々に必要とされる幅を決定しうる（５４）。バッファマネージャ３４は、並行処理装置４２の各々からデータ出力のタイプを認識するので、データを保持するためにローカルメモリバッファ４４の各々によって必要とされる幅を認識する。バッファマネージャ３４は、そのあとで、決定された幅を有するようにローカルメモリバッファ４４の各々を構成しうる（５６）。ある場合においては、ローカルメモリバッファ４４は、ハードウェアベースでありうるが、構成可能な態様のセット(a set of configurable aspects)を含む。例えば、ローカルメモリバッファ４４は、深さを幅とトレードすることが構成可能でありうる。

【００６３】

[0067] 例えば、バッファマネージャ３４は、頂点シェーダとして動作している並行処理装置４２Ａがポスト頂点シェーダの頂点データを出力するということを認識し、ポスト頂点シェーダの頂点データを保持するのに必要とされた幅を有するようにローカルメモリバッファ４４Ａを構成しうる。バッファマネージャ３４はまた、トライアングルラステライザとして動作している並行処理装置４２Ｂがプレフラグメントシェーダピクセルデータを出力するということを認識し、プレフラグメントシェーダ画素データを保持するのに必要とされた幅を有するようにローカルメモリバッファ４４Ｂを構成しうる。さらに、バッファマネージャ３４は、フラグメントシェーダとして動作している並行処理装置４２Ｃがポストフラグメントシェーダピクセル値を出力するということを認識し、ポストフラグメントシェーダピクセル値を保持するのに必要とされる幅を有するようにローカルメモリバッファ４４Ｃを構成しうる。

【００６４】

[0068] いったん並行処理装置４２とローカルメモリバッファ４４がＧＰＧＰＵ６内で処理パイプライン１０を実装するように構成されると、並行処理装置４２は、ローカルメモ

リバッファ４４を介して互いの間でデータを移送しうる（５８）。より具体的には、制御装置３０は、ローカルメモリバッファ４４にデータを送信するように並行処理装置４２のうち１以上を構成し、ローカルメモリバッファ４４からデータを受信するように並行処理装置４４のうち１以上を構成しうる。例えば、制御装置３０は、それぞれ、ローカルメモリバッファ４４Ａ、４４Ｂ、および４４Ｃにデータを送信するように並行処理装置４２Ａ、４２Ｂ、および４２Ｃを構成しうる。制御装置３０はまた、それぞれ、ローカルメモリバッファ４４Ａ、４４Ｂ、および４４Ｃからデータ受信するように並行処理装置４２Ｂ、４２Ｃ、および４２Ｄを構成しうる。

【００６５】

[0069] 図５は、ＧＰＧＰＵ６の並行処理装置４２とローカルメモリバッファ４４とによって実装される処理パイプライン内でデータシーケンスをプリザーブする例示的な動作を図示するフローチャートである。ＧＰＧＰＵ６の制御装置３０は、並行処理装置４２のうち１以上内でデータシーケンスをプリザーブすることによって処理パイプライン内でデータシーケンスをプリザーブしうる。図示される動作は、図３のＧＰＧＰＵ６の並行処理装置４２Ａを参照して説明される。同様な動作が他の並行処理装置４２のうちのいずれについても実行されうる。

【００６６】

[0070] 例として、並行処理装置４２およびローカルメモリバッファ４４は、３Ｄグラフィクス処理パイプラインを実装するように構成されうる。その例では、並行処理装置４２Ａは、頂点シェーダとして動作するように構成され、並行処理装置４２Ｂは、トライアングルラステライザとして動作するように構成され、並行処理装置４２Ｃは、フラグメントシェーダとして動作するように構成され、並行処理装置４２Ｄは、ピクセルブレンダとして動作するように構成されうる。

【００６７】

[0071] 処理パイプライン１０のステージ、例えば頂点シェーダとして動作するように構成された並行処理装置４２Ａは、処理のためにデータセットを受信する（６２）。例えば、データ配信装置４０は、頂点データのデータセットを並行処理装置４２Ａに割り当て、並行処理装置４２Ａは、バス４６を介してデバイスメモリ１２から割り当てられたデータセットを受信しうる。データセットが並行処理装置４２Ａに入る時に、制御装置３０のシーケンスマネージャ３６は、シーケンス決定カウンタ（ＳＤＣ）を実行する（６４）。ＳＤＣにしたがって、シーケンスマネージャ３６は、ローカルメモリ３８内で受信されたデータセットのデータスレッドのシーケンスを記録する（６６）。例えば、シーケンスマネージャ３６は、データスレッドがデバイスメモリ１２から受信される順でデータセットの各データスレッドのインデクスを記録しうる。

【００６８】

[0072] 頂点シェーダとして動作するように構成された並行処理装置４２Ａは、そのあとで、ポスト頂点シェーダの頂点データを生成するためにデータセットを処理する（６８）。上述されているように、並行処理装置４２Ａは、トライアングルラステライザとして動作するように構成された並行処理装置４２Ｂにデータセットを移送するために、ローカルメモリバッファ４４Ａにポスト頂点シェーダの頂点データを送信するように構成されうる。データセットが並行処理装置４２Ａから出る時に、シーケンスマネージャ３６は、シーケンスエンフォーシングバリア（ＳＥＢ）を実行する（７０）。ＳＥＢにしたがって、シーケンスマネージャ３６は、ＳＤＣによって記録されるのと同じシーケンスで並行処理装置４２Ａからデータセットのデータスレッドをリリースする（７２）。例えば、シーケンスマネージャ３６は、ローカルメモリ３８に記録されたデータスレッドインデクスにアクセスし、インデクスが記録された順にしたがって各データスレッドをリリースする。このように、頂点シェーダとして動作するように構成された並行処理装置４２Ａに複数の頂点が入ったのと同じ順でトライアングルラステライザとして動作するように構成された並行処理装置４２Ｂに複数の頂点が入るであろう。

【００６９】

[0073] １以上の例では、説明された機能は、ハードウェア、ソフトウェア、ファームウェア、またはそれらのいずれの組み合わせにおいて実装されうる。ソフトウェアで実装される場合には、機能または動作は、非一時的なコンピュータ可読媒体で１以上の命令またはコードとして格納され、ハードウェアベースの処理装置によって実行されうる。コンピュータ可読媒体は、データ記憶媒体のようなタンジブル媒体に対応するコンピュータ可読媒体、または、例えば通信プロトコルにしたがって、１つの場所から別の場所へとコンピュータプログラムの移送を容易にする任意の媒体を含む通信媒体、を含めうる。このように、コンピュータ可読媒体は一般的に、（１）非一時的であるタンジブルコンピュータ可読記憶媒体または（２）信号または搬送波のような通信媒体に対応しうる。データ記憶媒体は、本開示で説明される技法の実装についての命令、コードおよび／またはデータ構造を取り出すために１以上のコンピュータまたは１以上のプロセッサによってアクセスされることができる任意の利用可能な媒体でありうる。コンピュータプログラムプロダクトは、コンピュータ可読媒体を含めうる。

【００７０】

[0074] 例として、また限定されないが、そのようなコンピュータ可読媒体は、ＲＡＭ、ＲＯＭ、ＥＥＰＲＯＭ、ＣＤ－ＲＯＭあるいは他の光学ディスクストレージ、磁気ディスクストレージあるいは他の磁気ストレージデバイス、フラッシュメモリ、のようなノントランジトリ媒体、あるいは、命令あるいはデータス構造の形態で所望プログラムコードを格納または搬送するために使用されることができる、また、コンピュータによってアクセスされることができる、任意の他の媒体を備えることができる。また、いずれの接続もコンピュータ可読媒体と適切に名付けられる。例えば、命令がウェブサイト、サーバ、あるいは、同軸ケーブル、光ファイバーケーブル、ツイストペア、デジタル加入者ライン（ＤＳＬ）、あるいは赤外線、無線、およびマイクロ波のような無線技術を使用している他の遠隔ソース、から送信される場合には、そのときには、同軸ケーブル、光ファイバーケーブル、ツイストペア、ＤＳＬ、あるいは赤外線、無線、およびマイクロ波のような無線技術は、媒体の定義に含まれる。しかしながら、コンピュータ可読記憶媒体およびデータ記憶媒体は接続、搬送波、信号、または他のトランジエント媒体を含まないが、代わりに、ノントランジエント、タンジブル記憶媒体を対象としているということは理解されるべきである。ここに使用されているように、ディスク(disk)とディスク(disc)は、コンパクトディスク(compact disc)（ＣＤ）、レーザーディスク(登録商標)(laser disc)、光学ディスク(optical disc)、デジタル汎用ディスク(digital versatile disc)（ＤＶＤ）、フロッピー（登録商標)ディスク(disk)およびブルーレイ（登録商標）ディスクを含んでおり、「ディスク(disks)」は、大抵、データを磁気で再生し、「ディスク(discs)」は、レーザーで光学的に再生する。上記のものの組み合わせも、コンピュータ可読媒体の範囲内に含まれるべきである。

【００７１】

[0075] 命令は、１以上のＤＳＰ、汎用マイクロプロセッサ、ＡＳＩＣ、ＦＰＧＡ、または他の同等な集積またはディスクリートな論理回路のような１以上のプロセッサによって実行されうる。したがって、ここで使用される用語「プロセッサ」は、前述の構造のうちのいずれかまたはここで説明される技法の実施に適切な任意の他の構造を指す。さらに、いくつかの態様では、ここで説明される機能は、符号化および復号のために構成された専用のハードウェアおよび／またはソフトウェアのモジュール内で提供されうる、または、組み合わせられたコーデックに組み込まれうる。また、本技法は、１つまたは複数の回路または論理構成要素において十分に実装されることができる。

【００７２】

[0076] 本開示の技法は、無線ハンドセット、集積回路（ＩＣ）または１セットのＩＣ（例えばチップセット）を含む種々さまざまなデバイスまたは装置で実装されうる。様々なコンポーネント、モジュールまたは装置は、開示された技法を実行するように構成されたデバイスの機能的態様を強調するために本開示で説明されており、異なるハードウェア装置による実現を必ずしも必要としていない。むしろ、上述されているように、様々な装置

は、コーデックハードウェア装置で組み合わせられ、または、適切なソフトウェアおよび／またはファームウェアと併せて上述されるような１以上のプロセッサを含むインタオペラティブハードウェア装置の集まりによって与えられうる。

【００７３】

[0077] 様々な例が説明されている。これらおよび他の例は、特許請求の範囲内にある。

以下に本件出願当初の特許請求の範囲を付記する。

［Ｃ１］

汎用グラフィック処理装置（ＧＰＧＰＵ）であって、

処理パイプラインのステージとして選択的に動作するように構成された２以上の並行処理装置と、

前記並行処理装置間の移送のためにデータを保持するように構成された１以上のローカルメモリバッファであって、前記バッファの各々が前記並行処理装置のうち少なくとも２つの間で接続される、１以上のローカルメモリバッファと、

を備えるＧＰＧＰＵ。

［Ｃ２］

前記１以上のローカルメモリバッファの各々は、前記処理パイプラインにおいて前記並行処理装置のうち前記少なくとも２つの間で直接接続される、［Ｃ１］に記載のＧＰＧＰＵ。

［Ｃ３］

前記１以上のローカルメモリバッファは、前記並行処理装置間の前記データの移送を可能にするハードウェアベースのデータフロー制御メカニズムを含む、［Ｃ１］に記載のＧＰＧＰＵ。

［Ｃ４］

前記１以上のローカルメモリバッファは、ハードウェアベースの先入れ先出しバッファ（ＦＩＦＯ）、後入れ先出しバッファ（ＬＩＦＯＳ）またはインデクス付けされたバッファのうちの少なくとも１つを備える、［Ｃ１］に記載のＧＰＧＰＵ。

［Ｃ５］

前記１以上のローカルメモリバッファにデータを送信するように前記並行処理装置のうち１以上を構成し、前記１以上のローカルメモリバッファからデータを受信するように前記並行処理装置のうち１以上を構成するように構成された制御装置、

をさらに備える［Ｃ１］に記載のＧＰＧＰＵ。

［Ｃ６］

前記制御ユニットは、前記ローカルメモリバッファにデータを送信し前記ローカルメモリバッファからデータを受信するように前記並行処理装置を構成するために１以上のアプリケーションプログラミングインタフェース（ＡＰＩ）を実行するように構成される、［Ｃ５］に記載のＧＰＧＰＵ。

［Ｃ７］

前記処理パイプラインにおいて前の処理装置からのデータ出力を保持するために前記ローカルメモリバッファの各々についての必要な幅を決定するように、前記決定された幅を有するように前記ローカルメモリバッファの各々を構成するように、構成された制御装置、

をさらに備える［Ｃ１］に記載のＧＰＧＰＵ。

［Ｃ８］

前記制御装置は、１以上のアプリケーションプログラミングインタフェース（ＡＰＩ）を実行して、前記ローカルメモリバッファの各々についての前記幅を決定し、前記決定された幅によって前記ローカルメモリバッファの各々を構成し、前記ローカルメモリバッファの各々の深さを決定するように構成される、［Ｃ７］に記載のＧＰＧＰＵ。

［Ｃ９］

前記ローカルメモリバッファの各々は深さを幅とトレードすることが構成可能である、［Ｃ７］に記載のＧＰＧＰＵ。

［Ｃ１０］
前記処理パイプライン内でデータシーケンスをプリザーブする制御装置、をさらに備える［Ｃ１］に記載のＧＰＧＰＵ。
［Ｃ１１］
前記制御装置は、
前記データセットのデータスレッドのシーケンスを記録するために前記並行処理装置のうちの少なくとも１つにデータセットが入るときにシーケンス決定カウンタを実行するように、
前記シーケンス決定カウンタによって記録されるのと同じシーケンスで前記並行処理装置から前記データセットの前記データスレッドをリリースするために前記並行処理装置のうちの前記少なくとも１つから前記データセットが出るときにシーケンスエンフォーシングバリアを実行するように、
構成される、［Ｃ１０］に記載のＧＰＧＰＵ。
［Ｃ１２］
前記並行処理装置のうちの１つは、前記処理パイプラインの第１ステージとして動作し、デバイスメモリからオリジナルデータセットを取り出すように構成される、［Ｃ１］に記載のＧＰＧＰＵ。
［Ｃ１３］
前記並行処理装置のうちの１つは、前記処理パイプラインの最終ステージとして動作し、デバイスメモリにパイプライン処理されたデータセットを格納するように構成される、［Ｃ１］に記載のＧＰＧＰＵ。
［Ｃ１４］
前記並行処理装置のうちの少なくとも１つは、前記処理パイプラインの中間ステージとして動作し、前記ローカルメモリバッファのうち一方を介して前記処理パイプラインにおいて前記並行処理装置のうちの前のものからデータセットを受信し、前記ローカルメモリバッファのうち他方を介して前記処理パイプラインにおいて前記並行処理装置のうち後続のものに前記データセットを送信するように構成される、［Ｃ１］に記載のＧＰＧＰＵ。
［Ｃ１５］
前記並行処理装置のうちの前記少なくとも１つは、前記データセットを処理するために、デバイスメモリから補助データを取り出すように構成される、［Ｃ１４］に記載のＧＰＧＰＵ。
［Ｃ１６］
汎用グラフィック処理装置（ＧＰＧＰＵ）によってデータを処理する方法であって、前記方法は、
処理パイプラインのステージとして選択的に動作するように２以上の並行処理装置を構成することと；
前記並行処理装置間の移送のためのデータを保持するように１以上のローカルメモリバッファを構成することと、なお、前記バッファの各々は、前記並行処理装置のうち少なくとも２つの間で接続される；
を備える、方法。
［Ｃ１７］
前記ローカルメモリバッファの各々は、前記処理パイプラインにおいて前記並行処理装置のうち前記少なくとも２つの間で直接接続される、［Ｃ１６］に記載の方法。
［Ｃ１８］
前記１以上のローカルメモリバッファは、前記並行処理装置間の前記データの移送を可能にするハードウェアベースのデータフロー制御メカニズムを含む、［Ｃ１６］に記載の方法。
［Ｃ１９］
前記１以上のローカルメモリバッファは、ハードウェアベースの先入れ先出しバッファ（ＦＩＦＯ）、後入れ先出しバッファ（ＬＩＦＯＳ）またはインデクス付けされたバッフ

ァのうちの少なくとも１つを備える、［Ｃ１６］に記載の方法。

［Ｃ２０］

前記１以上のローカルメモリバッファにデータを送信するように前記並行処理装置の１以上を構成することと、

前記１以上のローカルメモリバッファからデータを受信するように前記並行処理装置の１以上を構成することと、

をさらに備える［Ｃ１６］に記載の方法。

［Ｃ２１］

前記１以上の並行処理装置を構成することは、前記ローカルメモリバッファにデータを送信し前記ローカルメモリバッファからデータを受信するように前記並行処理装置を構成するために１以上のアプリケーションプログラミングインタフェース（ＡＰＩ）を実行することを備える、［Ｃ２０］に記載の方法。

［Ｃ２２］

前記１以上のローカルメモリバッファを構成することは、

前記前記処理パイプラインにおいて前の処理装置からのデータ出力を保持するために前記ローカルメモリバッファの各々についての必要な幅を決定することと、

前記決定された幅を有するように前記ローカルメモリバッファの各々を構成することと、

を備える、［Ｃ１６］に記載の方法。

［Ｃ２３］

前記１以上のローカルメモリバッファを構成することは、１以上のアプリケーションプログラミングインタフェース（ＡＰＩ）を実行して、前記ローカルメモリバッファの各々についての前記幅を決定し、前記決定された幅によって前記ローカルメモリバッファの各々を構成し、前記ローカルメモリバッファの各々の深さを決定することを備える、［Ｃ２２］に記載の方法。

［Ｃ２４］

前記ローカルメモリバッファの各々は深さを幅とトレードすることが構成可能である、［Ｃ２２］に記載の方法。

［Ｃ２５］

前記処理パイプライン内でデータシーケンスをプリザーブすること、

をさらに備える［Ｃ１６］に記載の方法。

［Ｃ２６］

前記データシーケンスをプリザーブすることは、

前記データセットのデータスレッドのシーケンスを記録するために前記並行処理装置のうちの少なくとも１つにデータセットが入るときにシーケンス決定カウンタを実行することと、

前記シーケンス決定カウンタによって記録されるのと同じシーケンスで前記並行処理装置から前記データセットの前記データスレッドをリリースするために前記並行処理装置のうちの前記少なくとも１つから前記データセットが出るときにシーケンスエンフォーシングバリアを実行することと、

をさらに備える、［Ｃ２５］に記載の方法。

［Ｃ２７］

前記２以上の並行処理装置を構成することは、前記並行処理装置のうちの１つを、前記処理パイプラインの第１ステージとして動作し、デバイスメモリからオリジナルデータセットを取り出すように構成することを備える、［Ｃ１６］に記載の方法。

［Ｃ２８］

前記２以上の並行処理装置を構成することは、前記並行処理装置のうちの１つを、前記処理パイプラインの最終ステージとして動作し、デバイスメモリにパイプライン処理されたデータセットを格納するように構成することを備える、［Ｃ１６］に記載の方法。

［Ｃ２９］

前記２以上の並行処理装置を構成することは、前記並行処理装置のうちの少なくとも１つを、前記処理パイプラインの中間ステージとして動作し、前記ローカルメモリバッファのうち一方を介して前記処理パイプラインにおいて前記並行処理装置のうちの前のものからデータセットを受信し、前記ローカルメモリバッファのうち他方を介して前記処理パイプラインにおいて前記並行処理装置のうち後続のものに前記データセットを送信するように構成することを備える、［Ｃ１６］に記載の方法。

［Ｃ３０］

前記並行処理装置のうちの少なくとも１つを構成することは、前記データセットを処理するために、デバイスメモリから補助データを取り出すように前記並行処理装置のうちの前記少なくとも１つを構成することを備える、［Ｃ２９］に記載の方法。

［Ｃ３１］

汎用グラフィック処理装置（ＧＰＧＰＵ）であって、

処理パイプラインのステージとして選択的に動作するように２以上の並行処理装置を構成するための手段と；

前記並行処理装置間の移送のためのデータを保持するように１以上のローカルメモリバッファを構成するための手段と、なお、前記バッファの各々は、前記並行処理装置のうち少なくとも２つの間で接続される；

を備えるＧＰＧＰＵ。

［Ｃ３２］

前記ローカルメモリバッファの各々は、前記処理パイプラインにおいて前記並行処理装置のうち前記少なくとも２つの間で直接接続される、［Ｃ３１］に記載のＧＰＧＰＵ。

［Ｃ３３］

前記１以上のローカルメモリバッファは、前記並行処理装置間の前記データの移送を可能にするハードウェアベースのデータフロー制御メカニズムを含む、［Ｃ３１］に記載のＧＰＧＰＵ。

［Ｃ３４］

前記１以上のローカルメモリバッファにデータを送信するように前記並行処理装置の１以上を構成するための手段と、

前記１以上のローカルメモリバッファからデータを受信するように前記並行処理装置の１以上を構成するための手段と、

をさらに備える［Ｃ３１］に記載のＧＰＧＰＵ。

［Ｃ３５］

前記ローカルメモリバッファにデータを送信し前記ローカルメモリバッファからデータを受信するように前記並行処理装置を構成するために１以上のアプリケーションプログラミングインタフェース（ＡＰＩ）を実行するための手段、をさらに備える［Ｃ３４］に記載のＧＰＧＰＵ。

［Ｃ３６］

前記処理パイプラインにおいて前の処理装置からのデータ出力を保持するために前記ローカルメモリバッファの各々についての必要な幅を決定するための手段と、

前記決定された幅を有するように前記ローカルメモリバッファの各々を構成するための手段と、

をさらに備える［Ｃ３１］に記載のＧＰＧＰＵ。

［Ｃ３７］

１以上のアプリケーションプログラミングインタフェース（ＡＰＩ）を実行して、前記ローカルメモリバッファの各々についての前記幅を決定し、前記決定された幅によって前記ローカルメモリバッファの各々を構成し、前記ローカルメモリバッファの各々の深さを決定すること、をさらに備える［Ｃ３６］に記載のＧＰＧＰＵ。

［Ｃ３８］

前記ローカルメモリバッファの各々は深さを幅とトレードすることが構成可能である、［Ｃ３６］に記載のＧＰＧＰＵ。

［Ｃ３９］
前記処理パイプライン内でデータシーケンスをプリザーブするための手段、をさらに備える［Ｃ３１］に記載のＧＰＧＰＵ。
［Ｃ４０］
前記データセットのデータスレッドのシーケンスを記録するために前記並行処理装置のうちの少なくとも１つにデータセットが入るときにシーケンス決定カウンタを実行するための手段と、
前記シーケンス決定カウンタによって記録されるのと同じシーケンスで前記並行処理装置から前記データセットの前記データスレッドをリリースするために前記並行処理装置のうちの前記少なくとも１つから前記データセットが出るときにシーケンスエンフォーシングバリアを実行するための手段と、
をさらに備える［Ｃ３９］に記載のＧＰＧＰＵ。
［Ｃ４１］
汎用グラフィクス処理装置（ＧＰＧＰＵ）によってデータを処理するための命令を備えるコンピュータ可読媒体であって、実行時に、プログラム可能なプロセッサに、
処理パイプラインのステージとして選択的に動作するように２以上の並行処理装置を構成させる；
前記並行処理装置間の移送のためにデータを保持するように１以上のローカルメモリバッファを構成させる、なお、前記バッファの各々は、前記並行処理装置のうち少なくとも２つの間で接続される；
コンピュータ可読媒体。
［Ｃ４２］
前記ローカルメモリバッファの各々は、前記処理パイプラインにおいて前記並行処理装置のうち前記少なくとも２つの間で直接接続される、［Ｃ４１］に記載のコンピュータ可読媒体。
［Ｃ４３］
前記１以上のローカルメモリバッファは、前記並行処理装置間の前記データの移送を可能にするハードウェアベースのデータフロー制御メカニズムを含む、［Ｃ４１］に記載のコンピュータ可読媒体。
［Ｃ４４］
前記プログラム可能なプロセッサに、
前記１以上のローカルメモリバッファにデータを送信するように前記並行処理装置の１以上を構成させる命令と、
前記１以上のローカルメモリバッファからデータを受信するように前記並行処理装置の１以上を構成させる命令と、
をさらに備える［Ｃ４１］に記載のコンピュータ可読媒体。
［Ｃ４５］
前記プログラム可能なプロセッサに、前記ローカルメモリバッファにデータを送信し前記ローカルメモリバッファからデータを受信するように前記並行処理装置を構成するために１以上のアプリケーションプログラミングインタフェース（ＡＰＩ）を実行させる命令、をさらに備える［Ｃ４４］に記載のコンピュータ可読媒体。
［Ｃ４６］
前記プログラム可能なプロセッサに、
前記処理パイプラインにおいて前の処理装置からのデータ出力を保持するために前記ローカルメモリバッファの各々についての必要な幅を決定させる命令と、
前記決定された幅を有するように前記ローカルメモリバッファの各々を構成させる命令と、
をさらに備える［Ｃ４１］に記載のコンピュータ可読媒体。
［Ｃ４７］
前記プログラム可能なプロセッサに、１以上のアプリケーションプログラミングインタ

フェース（ＡＰＩ）を実行して、前記ローカルメモリバッファの各々についての前記幅を決定し、前記決定された幅によって前記ローカルメモリバッファの各々を構成し、前記ローカルメモリバッファの各々の深さを決定させる命令、をさらに備える［Ｃ４６］に記載のコンピュータ可読媒体。

［Ｃ４８］

前記ローカルメモリバッファの各々は深さを幅とトレードすることが構成可能である、［Ｃ４６］に記載のコンピュータ可読媒体。

［Ｃ４９］

前記プログラム可能なプロセッサに前記処理パイプライン内でデータシーケンスをプリザーブさせる命令、をさらに備える［Ｃ４１］に記載のコンピュータ可読媒体。

［Ｃ５０］

前記プログラム可能なプロセッサに、

前記データセットのデータスレッドのシーケンスを記録するために前記並行処理装置のうちの少なくとも１つにデータセットが入るときにシーケンス決定カウンタを実行させる命令と、

前記シーケンス決定カウンタによって記録されるのと同じシーケンスで前記並行処理装置から前記データセットの前記データスレッドをリリースするために前記並行処理装置のうちの前記少なくとも１つから前記データセットが出るときにシーケンスエンフォーシングバリアを実行させる命令と、

をさらに備える［Ｃ４９］に記載のコンピュータ可読媒体。

【図１】

図１

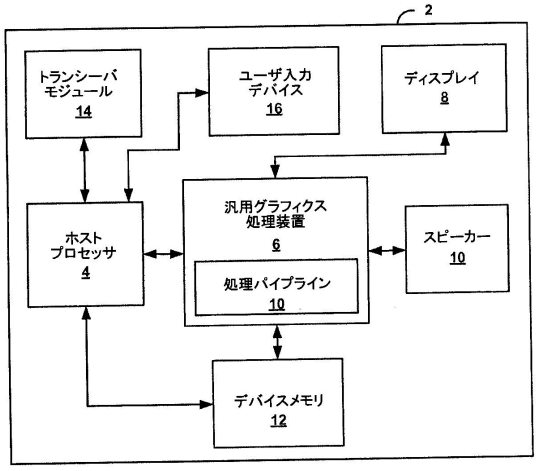

FIG. 1

【図２】

図２

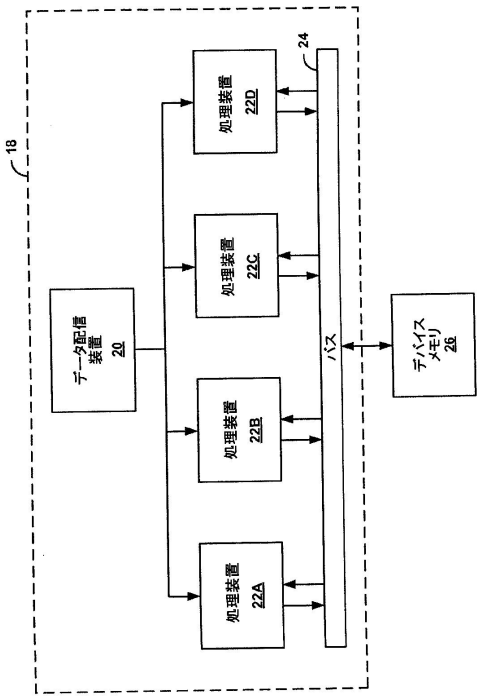

FIG. 2

【図３】

図３

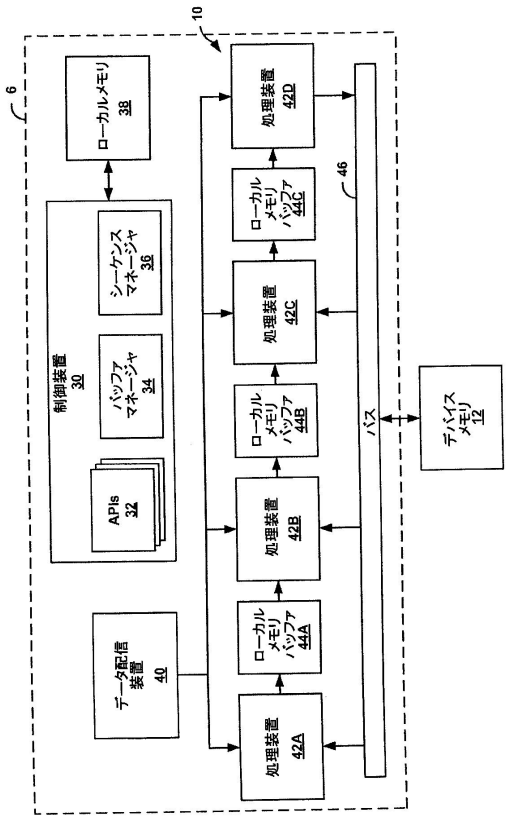

FIG. 3

【図４】

図４

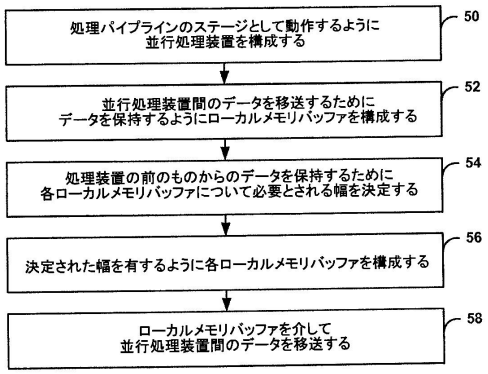

FIG. 4

【図５】

図５

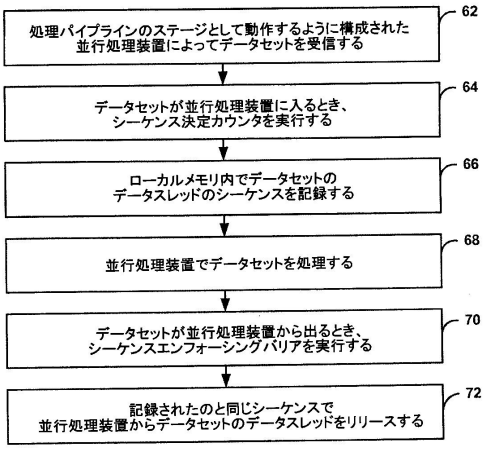

FIG. 5

フロントページの続き

(72)発明者　ボウルド、アレクセイ・ブイ.
アメリカ合衆国、カリフォルニア州　９２１２１、サン・ディエゴ、モアハウス・ドライブ　５７７５
(72)発明者　グルバー、アンドリュー
アメリカ合衆国、カリフォルニア州　９２１２１、サン・ディエゴ、モアハウス・ドライブ　５７７５
(72)発明者　クルスティク、アレクサンドラ・エル.
アメリカ合衆国、カリフォルニア州　９２１２１、サン・ディエゴ、モアハウス・ドライブ　５７７５
(72)発明者　シンプソン、ロバート・ジェイ.
アメリカ合衆国、カリフォルニア州　９２１２１、サン・ディエゴ、モアハウス・ドライブ　５７７５
(72)発明者　シャープ、コリン
アメリカ合衆国、カリフォルニア州　９２１２１、サン・ディエゴ、モアハウス・ドライブ　５７７５
(72)発明者　ユ、チュン
アメリカ合衆国、カリフォルニア州　９２１２１、サン・ディエゴ、モアハウス・ドライブ　５７７５

合議体
審判長　高木　進
審判官　辻本　泰隆
審判官　須田　勝巳

(56)参考文献　特表２００４－５１５８５６（ＪＰ，Ａ）
特開２００８－９６９７（ＪＰ，Ａ）
特開平６－２８２４４７（ＪＰ，Ａ）
特開２０１０－２８７１１０（ＪＰ，Ａ）
特開２０１０－２４４０９６（ＪＰ，Ａ）

(58)調査した分野(Int.Cl.，ＤＢ名)
G06F9/46-9/54