(19)日本国特許庁(JP)　(12)特　許　公　報(B2)　(11)特許番号
特許第3696012号
(P3696012)
(45)発行日　平成17年9月14日(2005.9.14)　(24)登録日　平成17年7月8日(2005.7.8)

(51)Int.Cl.[7]　F I
G06F 13/36　G06F 13/36　310F
G06F 12/08　G06F 12/08　G
　　G06F 12/08　J
　　G06F 12/08　X

請求項の数 13　(全 13 頁)

(21)出願番号　特願平11-331526
(22)出願日　平成11年11月22日(1999.11.22)
(65)公開番号　特開2000-235543(P2000-235543A)
(43)公開日　平成12年8月29日(2000.8.29)
審査請求日　平成13年7月4日(2001.7.4)
(31)優先権主張番号　09/198833
(32)優先日　平成10年11月24日(1998.11.24)
(33)優先権主張国　米国(US)

前置審査

(73)特許権者　398038580
ヒューレット・パッカード・カンパニー
HEWLETT-PACKARD COMPANY
アメリカ合衆国カリフォルニア州パロアルト　ハノーバー・ストリート　3000
(74)代理人　100099623
弁理士　奥山　尚一
(74)代理人　100096769
弁理士　有原　幸一
(74)代理人　100107319
弁理士　松島　鉄男
(72)発明者　デレク・エイ・シャーロック
アメリカ合衆国　コロラド州，フォート・コリンズ，トレイル，オレゴン　3407
最終頁に続く

(54)【発明の名称】書込み動作順序付けバスブリッジ

(57)【特許請求の範囲】
【請求項１】
バスブリッジ内のメモリ・コヒーレンシを有するキャッシュを介して、該第２のバスから支配され該第２のバス上で完了するように予定された読み出し動作に対して、第１のバスから支配され第２のバスに予定された書込み動作を順序付けするための書込み動作順序付けバスブリッジであって、
前記第１のバスからの入出力装置への前記書込み動作を待機させるための第１の手段と、
待機させるための前記第１の手段で受け取った前記書込み動作の回数をカウントする第２の手段と、
該第２のバス上で完了した前記書込み動作の回数をカウントする第３の手段と、
前記キャッシュ内のキャッシュラインに保持されたデータの、第１のバスからフェッチしてからの時間を、前記第１の手段において受け取った前記書込み動作の回数の単位でカウント回数として測定する第４の手段と、
前記キャッシュから読み出されるべき前記データの前記カウント回数が、前記第１の手段に受け取られた前記書込み動作の回数と、前記第２のバス上で完了される前記書込み動作の回数との差よりも小さいときに、前記読み出し動作の完了を遅延させる第５の手段と
を含んでなる、書込み動作順序付けバスブリッジ。
【請求項２】
前記キャッシュ内の前記キャッシュラインに保持された前記カウント回数を、前記デー

タが第１のバスからフェッチされるたびにゼロに設定する第６の手段をさらに有する、請求項１に記載の書込み動作順序付けバスブリッジ。

【請求項３】

カウントする前記第２の手段とカウントする前記第３の手段とが、カウントを生成する際に、係数Ｍをカウンタの最大値として使用するものである請求項１に記載の書込み動作順序付けバスブリッジ。

【請求項４】

前記係数Ｍが、待機させるための前記第１の手段に待機させることができる書込み動作の最大回数よりも大きい、請求項３に記載の書込み動作順序付けバスブリッジ。

【請求項５】

バスブリッジ内のメモリ・コヒーレンシを有するキャッシュを介して、書込み動作が第１のバスから支配され第２のバスに予定され、読み出し動作が該第２のバスから支配され該第２のバス上で完了するように予定されている、前記バスブリッジによって結合された前記第１及び第２のバスの間の前記読み出し動作に対して前記書込み動作を順序付けするための書込み動作順序付け方法であって、

（ａ）前記第１のバスからの入出力装置への前記書込み動作を待機させるための書込み要求蓄積バッファを提供するステップと、

（ｂ）前記書込み要求蓄積バッファで受け取った書込み動作の回数をカウントするステップと、

（ｃ）前記第２のバス上で完了した書込み動作の回数をカウントするステップと、

（ｄ）前記キャッシュ内のキャッシュラインに保持された前記データの、第１のバスからフェッチしてからの時間を、前記書込み要求蓄積バッファで受け取った書込み動作の回数の単位でカウント回数として測定するステップと、

（ｅ）前記キャッシュから読み出されるべき前記データの前記カウント回数が、前記書込み要求蓄積バッファで受け取った前記書込み動作の回数と前記第２のバス上で完了した前記書込み動作の回数との間の差よりも小さいときに、前記読み出し動作の完了を遅延させるステップと

を含んでなる書込み動作順序付け方法。

【請求項６】

前記キャッシュ内の前記キャッシュラインに保持された前記カウント回数を、前記データを第１のバスからフェッチするたびにゼロに設定するステップをさらに有する、請求項５に記載の書込み動作順序付け方法。

【請求項７】

係数Ｍを、ステップ（ｂ）及び（ｃ）でカウントするためのカウンタの最大値として使用されるものである、請求項５に記載の書込み動作順序付け方法。

【請求項８】

前記係数Ｍが、前記書込み要求蓄積バッファに待機させることができる書込み動作の最大回数よりも大きい、請求項７に記載の書込み動作順序付け方法。

【請求項９】

バスブリッジ内のメモリ・コヒーレンシを有するキャッシュを介して、第１のバスから支配され第２のバスに予定された書込み動作を、該第２のバスから支配され該第２のバス上で完了するように予定された読み出し動作に対して順序付けするためのバスブリッジであって、

前記第１のバスからの入出力装置への前記書込み動作を待機させるための書込み記憶領域と、

該書込み記憶領域で受け取った書込み動作の回数をカウントするためのプロデューサ書込みカウンタと、

前記第２のバス上で完了した書込み動作の回数をカウントするためのコンシューマ書込みカウンタと、

前記キャッシュ内のキャッシュラインに保持されたデータの、第１のバスからフェッチ

してからの時間を、前記書込み記憶領域で受け取った書込み動作の回数の単位でカウント回数として測定するためのキャッシュラインエージャカウンタと、

前記キャッシュから読出すべきデータについての前記キャッシュラインエージャカウンタのカウント回数が、前記書込み記憶領域で受け取った書込み動作の前記回数と前記第２のバス上で完了した書込み動作の前記回数との差よりも小さいときに、前記読出し動作の完了を遅延させるための順序付けロジックと

を含んでなるバスブリッジ。

【請求項１０】

前記第１のバスと、前記データがフェッチされるたびに前記キャッシュラインエージャカウンタをゼロにリセットするための前記キャッシュラインエージャカウンタとに通信するリセット信号線をさらに含む請求項９に記載のバスブリッジ。

【請求項１１】

前記プロデューサ書込みカウンタと前記コンシューマ書込みカウンタはカウントを生成する際に、係数Ｍをカウンタの最大値として使用する請求項９に記載のバスブリッジ。

【請求項１２】

前記係数Ｍは、前記書込み記憶領域に待機させることができる書込み動作の最大回数よりも大きい請求項１１に記載のバスブリッジ。

【請求項１３】

前記キャッシュ内の前記キャッシュラインに保持されたデータについての前記キャッシュラインエージャカウンタのカウント回数は、前記係数Ｍに等しい最大値に制限される請求項１１に記載のバスブリッジ。

【発明の詳細な説明】

【０００１】

【発明の属する技術分野】

本発明は、一般に、コンピュータ・システムにおける入出力（Ｉ／Ｏ）装置制御の分野に係り、特に、バスブリッジの書込み要求記憶領域によって２つのコンピュータ・バス間のバスブリッジを通過する書込みデータに順序違反が生じる可能性のある場合にリード・リターン・データ(read return date)の適切な順序付けを行うバスブリッジに関する。

【０００２】

【従来の技術】

コンピュータ・システムでは、入出力装置による直接メモリ・アクセス（ＤＭＡ）読み出しが、メモリ・アクセス待ち時間をマスク(mask)できるようにある形式のプリフェッチ処理を使用することが望ましい。この結果、入出力バスをより効率的に使用することができる。しかしながら、ＤＭＡ装置に供給されるプリフェッチされたデータが、プロセッサ、他の入出力装置又はバスブリッジによる最新のメモリ書込みを反映できるように、プリフェッチ処理はコヒーレント(coherent)であることが重要である。コヒーレントであるプリフェッチ・バッファは、中央処理装置（以下、ＣＰＵと称す）などの他の装置と共にシステムのコヒーレントなプロトコルに関係するキャッシュと見なすことができる。プリフェッチ・キャッシュがコヒーレントである限り、プリフェッチ・キャッシュへのＤＭＡトラフィックの完了は、順序付け及びコヒーレントの点で、メモリに対する完了と同等である。

【０００３】

【発明が解決しようとする課題】

ＣＰＵによる入出力装置への書込みは、プログラム式入出力（programmed I/O;以下、ＰＩＯと略す）としても知られ、接続された形又は書込み要求先入れ先出し記憶領域（posted write first in first out buffer;以下、書込み要求記憶領域ＦＩＦＯバッファと略す）によって直接行われることがある。入出力バスを通じ入出力装置への直接接続を経由してＰＩＯが行われる場合、プリフェッチ・キャッシュのコヒーレントは、キャッシュ・ヒット(cache hit)からのＤＭＡ読み出しデータがＰＩＯ書込みに対して適切に順序付けされていれば十分である。一方、書込み要求記憶領域ＦＩＦＯバッファを使用する場合

、ＰＩＯ書込みは、入出力バスに書き込まれる前に転記型書込みＦＩＦＯで遅延される。この遅延により、読み書きの順序違反が生じることがある。

【０００４】

この順序違反に対する脆弱性(vulnetability)は、下記の例によって説明される。ＣＰＵが２つの書込みＡ及びＢをその書込み順序で実行すると想定する。書込みＡは、入出力バスに接続された装置へのＰＩＯ書込みである。書込みＢは、メモリ内のロケーションを修正する。書込みＡは、書込み要求記憶領域ＦＩＦＯバッファに転記されるため、入出力バスへの書込みが実行されるまでに少し遅延がある。しかし、書込みＢは、すぐに有効になり、したがって、そのメモリ・ロケーションを読み出すどの入出力装置にも認識できる。したがって、書込みＡがまだ書込み要求記憶領域ＦＩＦＯバッファにある間に書込みＡを書き込まれた入出力装置が書込みＢを書き込まれたメモリ・ロケーションを読み出す必要がある場合、入出力装置には、書込みＢの影響（すなわち、メモリ値の変化）が書込みＡの影響よりも前に認識される。これにより、順序違反が起こる。

【０００５】

本発明は上記事情に鑑みてなされたものであり、コンシューマ・エージェントによって開始されるＤＭＡ読み出しを、プロデューサ・エージェントによって開始されるＰＩＯに対して順序付けするプロデューサーコンシューマ順序付けシステムを提供することを目的とする。なお、この順序付けは、書込みが、異なるバス上の混合された(mixture)異なるターゲットへのものである場合にも保証される。

【０００６】

【課題を解決するための手段】

本発明は、一般に、バスブリッジ内の書込み要求記憶領域 (write posting storage)の存在によって順序違反を引き起こす可能性のある場合に、２つのコンピュータ・バス間のインタフェースを提供し、一方のバスから支配された書込み動作を、他方のバスから支配された読み出し動作に対して適切に順序付けできるようにするバスブリッジを提供する。このバスブリッジは、バスブリッジが受け取る書込み要求記憶領域に待機させる書込み動作の回数をカウントする第１の機構を有する。さらに、バスブリッジは、第２のバス上で完了した書込み動作の回数をカウントする第２の機構を有する。また、バスブリッジの一部分として、コヒーレントなキャッシュの各キャッシュラインに保持されたデータがいつ送信されてきたか(age of data)（以下、データの送信時と称す）を測定する機構を有する。最後に、バスブリッジは、キャッシュ・データをフェッチする前に第１のバス上のバスブリッジが受け入れたすべての書込みが第２のバス上で完了するまで、キャッシュからの読み出し動作の完了を遅延させる機構を有する。これは、バスブリッジが受け取った書込み動作の回数と第２のバス上で完了した書込み動作の回数との間の差と、読み出すキャッシュラインに保持されたデータの送信時を比較することによって決定される。

【０００７】

また、本発明は、バスブリッジによって結合された２つのバス間の読み出し動作に対して書込み動作を順序付ける方法を提供する。この方法は、次に段階により広義に要約する。バスブリッジに、第１のバスから受け取った書込み動作を待機させる書込み要求記憶領域を提供する。書込み要求記憶領域で受け取った書込み動作の回数と、第２のバス上で完了した書込み動作の回数との両方のカウントを維持する。コヒーレントなキャッシュの各キャッシュラインに保持されたデータの送信時を、キャッシュラインのデータをフェッチしてからバスブリッジが受け取った書込み動作の回数の単位で測定する。最後に、読み出されるキャッシュ・データをフェッチする前に第１のバス上のバスブリッジが受け取り、第２のバス上で完了していない書込みがあるとき、コヒーレントなキャッシュからの読み出し動作が完了から遅延される。

【０００８】

本発明の利点は、キャッシュからの読み出し動作が、適切なプロデューサーコンシューマ順序を維持するために必要な最低限の時間が遅延されることである。

【０００９】

本発明の他の利点は、２バス構成及び方法に対して最低限の修正を加えるか修正を加えないかにかかわらず１対多のバスブリッジ構成に容易に拡張できること。

【００１０】

【発明の実施の形態】

本発明の原理を、入出力バスブリッジへの簡単なプロセッサ・バスを有するコンピュータ・システムに関して説明する。この例は、バス及びトランザクション・タイプを指定するために明確な用語を使用できるために選択された。図１は、プロセッサ／メモリ・バスと入出力バスとの間の通信を容易にするバスブリッジの構成を示すブロック図である。コンピュータ・システム２１は、ＣＰＵバス・インタフェース２５及び入出力バス・インタフェース２７を用いてバスブリッジ２６によって接続されたＣＰＵ／メモリ・バス２２（第１のバス）及び入出力バス２４（第２のバス）を有する。少なくとも１つのＣＰＵ２８が、ＣＰＵ／メモリ・バス２２を介してメモリ３２と通信し、入出力バス２４は入出力装置３４と入出力装置３４との通信に有用である。性能を高めるため、バスブリッジ２６は、メモリ３２にアクセスする際に入出力装置３４～３４が被る待ち時間をマスクするプリフェッチ・キャッシュ３６を有する。バスブリッジ２６は、さらに、キャッシュ３６をサポートし、プリフェッチしたデータのコヒーレンシを保証する（すなわち、入出力装置３４～３４に供給されるデータを、メモリ３２への最新の書込みに反映させる）ＤＭＡ制御ロジック３８を有する。第１の手段である書込み要求記憶領域ＦＩＦＯバッファ４２は、入出力バス２４に接続された入出力装置３４へのＣＰＵ２８からの書込み要求を受け取って待機する。

【００１１】

ここで、コンピュータ・システム２１の動作の説明に使用される用語を定義する。バスブリッジ２６を介してメモリの読み出し又は書込みをする入出力装置３４によって開始されるサイクルをすべてＤＭＡと称す。また、バスブリッジ２６を介して入出力装置３４の読み出し又は書込みをするＣＰＵ２８によって開始されるサイクルをすべてＰＩＯと称す。図１のすべての矢印の方向は、バスブリッジ２６を介したＤＭＡトラフィック及びＰＩＯ書込みトラフィックの流れを示す。ＰＩＯ読み出しトラフィックは、この種のトラフィックが本発明に関係ないため、図１には示していない。

【００１２】

コンピュータ・システム２１は、前記タイプのプロデューサーコンシューマ順序違反の影響を受けやすい。具体的には、ＣＰＵ２８が、第１の入出力装置３４へのＰＩＯ書込みと、第２のメモリ３２への書込みとの２つの書込みを行う場合、書込まれる入出力装置３４は、書込み要求記憶領域ＦＩＦＯバッファ４２による遅延のために、第１の書込みが有効になる前に第２の書込みによって変更されたメモリ３２内のロケーションのＤＭＡ読み出しを実行することがある。

【００１３】

いくつかの従来技術の解決法は、入出力装置３４～３４（すなわち、コンシューマ）が、ＣＰＵ２８（すなわちプロデューサ）によって行われる書込みを、生成された順序と同じ順序で認識するために有効である。第１の従来技術の解決法は、ＤＭＡ読み出しが完了したときに書込み要求記憶領域ＦＩＦＯバッファ４２を空にすることである。ＤＭＡ読み出しを最初に受け取ってから書込み要求記憶領域ＦＩＦＯバッファ４２が空になるまでの間にすべてのＰＩＯ書込みにフロー制御を行う状態機械(state machine)を実現することによって、このプロセスを達成することができる。書込み要求記憶領域ＦＩＦＯバッファ４２が空になれば、ＤＭＡ読み出しを完了することができる。

【００１４】

ＤＭＡ読み出し動作は、最初のＤＭＡ読み出し要求と入出力バス２４上のＤＭＡ読み出しの完了との２つの別々のトランザクションに分かれることが多い。ＰＩＯ書込みトランザクションにより分割(split)ＤＭＡ読み出しの適切な順序付けを保証するために、順序付けシステムは、ＤＭＡ読み出しの完了を遅延させる機能を必要とし、入出力バス２４上で書込み要求記憶領域ＦＩＦＯバッファ４２からの少なくとも１つのＰＩＯ書込みが許可

される。

【００１５】

しかし、ＤＭＡ読み出しを完了する前に書込み要求記憶領域ＦＩＦＯバッファ４２を使用することにより、動作上の問題が２つ生じる。第１の問題は、ＤＭＡ読み出しがコンピュータ・システム２１で頻繁に行われる場合、ＰＩＯ書込みのフロー制御も頻繁に行われる。その結果、部分的又は全体的に書込み要求記憶領域ＦＩＦＯバッファ４２を使用する利点がなくなる。第２の問題は、順序違反が生じない場合でも、書込み要求記憶領域ＦＩＦＯバッファ４２が空でないときに、読み出しサイクルの完了が遅延されるため、ＤＭＡ読み出し性能が低下する。すなわち、読み出すメモリ・ロケーションがＣＰＵ２８によって修正されなかったとき、ＤＭＡ読み出しを要求する入出力装置へのメモリ３２へのＰＩＯ書込みが行われた後でメモリ３２への書込みが行われる。

【００１６】

第２の従来技術の解決法は、プリフェッチ・キャッシュ３６からのすべてのＤＭＡ読み出しデータを、強制的に書込み要求記憶領域ＦＩＦＯバッファ４２に送信することである。この方法では、ＤＭＡ読み出しデータは、ＤＭＡ読み出しに先立つＰＩＯ書込みの後で、書込み要求記憶領域ＦＩＦＯバッファ４２に待機させられる。適切なプロデューサーコンシューマ順序付けは保証されるが、いくつかの欠点が明らかになる。第１の問題は、ＰＩＯ書込みが行われてからＤＭＡ読み出しデータが修正されたかに関係なく、ＰＩＯ書込みが未処理のときには常に、ＤＭＡ読み出しの完了が遅延される。第２の問題は、多くの場合、ＤＭＡ読み出しデータによって使用領域(space)を消費するため、書込み要求記憶領域ＦＩＦＯバッファ４２は、容量が大きくなければならない。これにより、バスブリッジ２６のコストが増大する。第３の問題は、ある環境では、ＰＩＯ書込みエントリを妨げることなく書込み要求記憶領域ＦＩＦＯバッファ４２からＤＭＡ読み出しデータをフラッシュできることが望ましい。そのような環境では、入出力バス２４を効率よく利用するためにプリフェッチされたデータが書込み要求記憶領域ＦＩＦＯバッファ４２を介してバースト(burst)しているが、入出力装置３４がバーストを切断する。したがって、書込み要求記憶領域ＦＩＦＯバッファ４２からの不連続なＤＭＡ読み出しデータを選択的にフラッシュするために、複雑な制御ロジックが必要になる。しかしながら、この制御ロジックは、バスブリッジ２６のコスト及び複雑さを高める。

【００１７】

書込み要求記憶領域ＦＩＦＯバッファ４２を介してＤＭＡ読み出しデータを送る前記方法の他の方法は、書込み要求記憶領域ＦＩＦＯバッファ４２を介してＤＭＡ読み出しデータの各バーストのヘッドを示す信号トークンを送ることである。ＤＭＡ読み出しデータは、キャッシュ３６を介してアクセスされるが、状態機械は、読み出し要求と関連したトークンが書込み要求記憶領域ＦＩＦＯバッファ４２を通過するまで、ＤＭＡ読み出し要求を一次中止状態に保持することになる。この方法は、書込み要求記憶領域ＦＩＦＯバッファ４２内の使用領域の浪費ほどではないが、前の方法と同じ欠点がある。

【００１８】

図２は、本発明の一実施形態に係るＤＭＡ読み出しの完了時間をより選択的に遅延させる図１のバスブリッジを示す詳細なブロック図である。詳細には、ＤＭＡ読み出しの完了は、少なくとも１つの未処理のＰＩＯ書込みがあり、かつ最も古い未処理のＰＩＯ書込みが行われてから、読み出されるキャッシュラインがメモリ３２からフェッチされた場合にのみ遅延される。ＤＭＡ読み出しの完了は、適切なプロデューサーコンシューマ順序付けが維持されるために必要な最小時間だけ遅延されることが有利である。

【００１９】

次に、図２を参照すると、コンピュータ・システム４９は、図１に示したようなコンピュータ・システム２１を含む構成要素を含むが、さらに、１）第２の手段であるプロデューサＰＩＯ書込みカウンタ（producer PIO write counter;以下、ＰＰＣと略す）５４、２）第３の手段であるコンシューマＰＩＯ書込みカウンタ（consumer PIO write counter;以下、ＣＰＣと略す）５６、３）第４の手段であるキャッシュライン・エージャ（cache

line agers;以下、ＣＬＡと略す）５８、及び４）第５の手段である順序付けロジック６２の４つの追加の構成要素を有する修正されたプロデューサーコンシューマ順序付けバスブリッジ５０を備える。

【００２０】

ＰＰＣ５４は、係数Ｍ（modulus M）を使って、入出力バス２４に転送するためにバスブリッジ５０によって受け入れられるＰＩＯ書込みの回数をカウントするカウンタである。ＣＰＣ５６は、同じ係数Ｍを使って、入出力バス２４上で完了するＰＩＯ書込みの回数をカウントするカウンタである。入出力バス・インタフェース２７は、入出力バス２４上でＰＩＯ書込みが完了するときに線５７上にインクリメント信号を生成する。ＣＬＡカウンタ５８は、ＰＰＣ５４及びＣＰＣ５６と同じビット数を使用するカウンタであるが、係数カウンタではない。代わりに、ＣＬＡカウンタ５８は、ゼロにリセットされるまでそのまま最大値までカウントするように設計される。ＣＬＡをキャッシュ３６に接続する破線の矢印で示したプリフェッチ・キャッシュ３６の各キャッシュラインごとに１つのＣＬＡカウンタ５８がある。ＣＬＡカウンタ５８は、ＰＰＣ５４と協力して（すなわち、バスブリッジ５０によってＰＩＯ書込みを受け入れるたびに）増分するように同期される。したがって、ＣＬＡカウンタ５８とＰＰＣ５４とは、バスブリッジ５０がＰＩＯ書込みを受け入れるときにインクリメント信号をアサートするＣＰＵバス・インタフェース２５により駆動される共通のインクリメント信号線５９を共用する。ＣＬＡカウンタ５８は、ＣＰＵバス・インタフェース２５を介してＣＰＵ／メモリ・バス２２によって駆動される第６の手段であるリセット信号線６１とリセット選択信号線６３とを通して、関連したキャッシュラインのＣＰＵ／メモリ・バス２２を介してデータがフェッチされるたびにゼロにリセットされる。リセット選択信号線６３は、リセット信号線６１を介してゼロにリセットするために関連したキャッシュラインがフェッチ又はリフェッチ(re-fetch)されたＣＬＡカウンタ５８だけを選択するために使用される。

【００２１】

要するに、ＣＬＡカウンタ５８は、関連したキャッシュラインに保持されたデータの送信時を、経過したＰＩＯ書込み受入れの単位で測定する。順序付けロジック６２は、所与のＤＭＡ読み出しがすぐに処理できるか又は入出力バス２４上のさらに他のＰＩＯ書込みの完了を可能にするためにＤＭＡ読み出しを遅延させなければならないかを決定するために使用されるハードウェア・ロジックを有する。順序付けロジックは、この決定を行う際に個々の基準でＰＰＣ５４、ＣＰＣ５６及びＣＬＡカウンタ５８の値を使用する。したがって、順序付けロジック６２及びＤＭＡ制御ロジック３８は、信号線６５上で互いに双方向通信である。ＣＬＡカウンタ５８及びＰＰＣ５４は両方とも、バスブリッジ５０のプロデューサ側（すなわち、ＣＰＵ／メモリ・バス２２上のＰＩＯの受入れ）と関連付けられ、ＣＰＣ５６は、バスブリッジ５０のコンシューマ側（すなわち、入出力バス２４上のＰＩＯの完了）と関連付けられる。一般に、プロデューサ（たとえば、ＣＰＵ２８及びＣＰＵ／メモリ・バス２２）が、コンシューマ（たとえば、入出力装置３４及び入出力バス２４）と同じ場所に必ずしも配置される必要はない。さらに、プロデューサ及びコンシューマ・バスは、互いに対して同期していなくてもよい。

【００２２】

適切な任意の時間に、未処理のＰＩＯ書込み（すなわち、入出力バス２４上で完了されるのではなくバスブリッジ５０によって受け取られた書込み）の回数は、（（ＰＰＣ－ＣＰＣ）モジューロＭ(moduloM)）によって示され、ここでＰＰＣ及びＣＰＣは、それぞれのカウンタの値である。係数Ｍは、書込み要求記憶領域ＦＩＦＯバッファ４２内の利用可能な領域によって制限される未処理のＰＩＯ書込みの最も大きな可能性の数よりも少なくとも１だけ大きくなるように選択される。一般に、係数Ｍは、実施しやすくするために２の累乗まで端数が丸められる。

【００２３】

ＣＰＵ／メモリ・バス２２、入出力バス２４及びプリフェッチ・キャッシュ３６がすべて１つのクロックを共用する場合、同期の必要はない。この場合、順序付けロジック６２

は、ＤＭＡ読み出しの完了を遅延させるか又はＤＭＡ読み出しをすぐに完了させるかの決定に、ＰＰＣ５４、ＣＰＣ５６及びＣＬＡカウンタ５８の値を直接使用することができる。

【００２４】

代替として、入出力バス２４がプリフェッチ・キャッシュ３６と非同期に動作する場合、ＣＰＣ５６の最新の値を順序付けロジック６２に提供するための何らかの機構がなければならない。入出力バス２４が遅延されたＤＭＡ読み出しを使用し、バスブリッジ５０からの再試行応答を受け取る入出力バス２４が、データが戻されるまでＤＭＡ読み出しを定期的に再試行し、各ＤＭＡ読み出し要求は、ＤＭＡ読み出し要求時にＣＰＣ５６の値を有する領域(field)を伴わなければならない。ＣＰＣ５６のこの値は、ＤＭＡ読み出しをいつ完了するかを決定する際に順序付けロジック６２によって使用される。遅延された読み出しシステムでは、ＤＭＡ読み出し要求は、一般に、クロック・ドメイン・クロスＦＩＦＯ(clockdomain crossing FIF0)などの同期リソースを介してキャッシュに送出される。

【００２５】

一方、入出力バス２４は、分割ＤＭＡ読み出しを使用することもできる。ＤＭＡ読み出しは、ＤＭＡ読み出し要求がＤＭＡ読み出し完了からの別々の入出力バス２４トランザクションであり、完了トランザクションが、読み出し要求のターゲットのエージェントによって入出力バス２４上で支配されるときに分割される。たとえば、ＤＭＡ読み出し要求が入出力バス２４により支配され、同時にＤＭＡ読み出しの完了が、バスブリッジ５０によって支配されることになる。分割ＤＭＡ読み出しシステムでは、入出力バス２４は、ＣＰＣ５６の値が変化したときには、プリフェッチ・キャッシュ３６への読み出し完了を含む順序付けロジック６２に通知しなければならない。この通知は、クロック・ドメイン・クロスＦＩＦＯなどの同期リソースにより行われなければならない。

【００２６】

図３は、本発明によりプロデューサーコンシューマ順序付けを保証する図２のハードウェア構成要素におけるバスブリッジ５０の動作を示すブロック図である。

【００２７】

まず図２及び図３を参照すると、ＤＭＡ読み出し動作は、入出力装置３４が入出力バス２４にＤＭＡ読み出し要求を出すことにより開始される。ＤＭＡ読み出し要求は、入出力バス・インタフェース２７を介してバスブリッジ５０によって受け取られる。ＤＭＡ制御ロジック３８は、ＤＭＡ読み出し要求のターゲットであるキャッシュラインと関連付けられたＣＬＡ選択線６５ａを介してＣＬＡカウンタ５８を選択し、その値を一般には順序付けロジック６２、詳細にはコンパレータ６８に提供する。ＣＬＡカウンタ５８の値は、関連するキャッシュラインがフェッチされてからバスブリッジ５０が受け入れたＰＩＯ書込みの回数を表す。ＣＰＣ５４及びＰＰＣ５６カウンタの値は、係数Ｍ算術(arithmetic)を使用するＰＰＣカウンタ５４とＣＰＣ５６との差を計算する減算器７２への入力として供給される。（（ＰＰＣ－ＣＰＣ）モジューロＭ）によって示される数は、現在未処理のＰＩＯ書込みの回数を表す。すなわち、バスブリッジ５０が受け入れたが入出力バス２４上で完了していないＰＩＯの回数を表す。減算器７２の出力は比較器６８に供給され、そこで、ＣＬＡカウンタ５８の値が、現在未処理のＰＩＯ書込みの回数よりも少ないかが判断される。（ＣＬＡ＜（ＰＰＣ－ＣＰＣ）モジューロＭ）の場合は、キャッシュ・データをフェッチしたときにバスブリッジ５０によって既に受け取られた少なくとも１つのＰＩＯ書込みが、入出力バス２４上で未完了である。これは、潜在的な順序違反である。

【００２８】

順序違反が起きないことを保証するために、比較器６８の出力に基づいた順序付けロジック６２が、少なくとも１つのＰＩＯ書込みが入出力バス２４上で完了するまで、ＤＭＡ制御ロジック３８への線６５ｂがＤＭＡ読み出しの完了を遅延させることができるようにＤＭＡ読み出しに関する信号を送る。遅延されたＤＭＡ読み出しを使用している場合は、入出力バス２４を解放(free up)してＰＩＯ書込みをさらに完了させるために再試行しなければならない。分割されたＤＭＡ読み出しを使用している場合は、少なくとも１つのＰ

ＩＯ書込みが完了するまで、リード・リターンを遅延させなければならない。少なくとも１つのＰＩＯ書込みが入出力バス２４上で完了した後、順序付けロジック６２は、増分されたＣＰＣ５６の値による演算を繰り返す。

【００２９】

順序付けロジック６２が、比較器６８によって、キャッシュラインが取り出されてからバスブリッジ５０によって受け取られたＰＩＯ書込みの回数（すなわち、ＣＬＡカウンタ５８の値）が、現在未処理のＰＩＯ書込みの回数以上であることを決定したとき、ＤＭＡ読み出しは、ＤＭＡ読み出し許可線６５ｂ上の許可信号の送信により順序付けロジック６２によって完了される。この場合、キャッシュラインに含まれるデータは、キャッシュラインがフェッチされるよりも前に行うことができた未完了のどのＰＩＯ書込みよりも前のものである。

【００３０】

ＰＩＯ書込み動作は、ＣＰＵ２８がＣＰＵ／メモリ・バス２２にＰＩＯ書込み要求を出すことによって開始される。ＰＩＯ書込み要求は、ＣＰＵバス・インタフェース２５を介してバスブリッジ５０によって受け取られる。バスブリッジ５０が、ＰＩＯ書込み要求を受け入れると、ＰＰＣ５４は、係数Ｍを使用して増分され、キャッシュ３６内のすべてのキャッシュラインのＣＬＡカウンタ５８が、インクリメント信号線５９により互いに協力して増分される。各ＣＬＡカウンタ５８のロジックは、カウンタが係数Ｍを超えたかどうかを判断するように設計される。係数Ｍを超えた場合、ＣＬＡカウンタ５８は、係数Ｍと等しい値に制限される。ＣＬＡカウンタ５８が値Ｍまでカウントしたとき、キャッシュラインのデータはとても古くなっているため、データがフェッチされたときに作動中であったＰＩＯ書込みはすべて入出力バス２４上で完了しなければならない。値Ｍは、未処理のＰＩＯ書込みの最大回数よりも少なくとも１つだけ大きくなるように選択されるため、書込み要求記憶領域ＦＩＦＯバッファ４２の記憶域は、データをフェッチしたときに処理中であった任意のＰＩＯ書込みが引き続き処理中であるためには十分でない。書込み要求記憶領域ＦＩＦＯバッファ４２からのＰＩＯ書込みが、入出力バス２４上で完了するとき、ＣＰＣ５６は、係数Ｍを使用してインクリメント信号線５７を介して増分される。

【００３１】

本発明により、各ＤＭＡ読み出しは、読み出すキャッシュラインがフェッチされたときに処理中であったすべてのＰＩＯ書込みが入出力バス２４上で完了した後で完了することができる。ＤＭＡ読み出しの進行を抑制することができる１つの要素は、キャッシュ３６の中央処理装置／メモリ・バス２２側からのスヌープ（詮索）又はキャッシュ３６メモリのセクションの最低使用頻度（ＬＲＵ）キャッシュ３６のパージである。たとえば、ＬＲＵの選択されたパージは、データが変更されていない場合でもキャッシュラインを繰返しリフェッチすることを必要とする。各リフェッチにより、キャッシュラインと関連付けられたＣＬＡカウンタ５８がゼロにリセットされる。同様に、中央処理装置２８がメモリ３２への書込みとＰＩＯ書込みとを交互に行うことによって、進行を阻止することができる。各メモリ３２への書込みは、キャッシュラインをスヌープし、ゼロにリセットされるキャッシュラインと関連したＣＬＡカウンタ５８でキャッシュラインはリフェッチされる。したがって、ＤＭＡ読み出しが完了する前に、すぐ前のＰＩＯ書込みが、入出力バス２４上で完了しなければならない。

【００３２】

本発明は、キャッシュ・メモリ・システムに一般に使用されるフォワード・プログレス・プロトコル(forward progress protocols)と互換性を持つ。まず、一般に、各ＤＭＡ読み出しバーストの第１のキャッシュラインにロック状態を適用することによって最低使用頻度を基準に選択されたパージが行われるため、リード・リターン値を提供するために各ＤＭＡ読み出しバーストが使用されるまで、各ＤＭＡ読み出しバーストはこのパージから考慮の対象とされない。

【００３３】

第２に、スヌープ（詮索）の無効化を検討する。スヌープのためにキャッシュラインを

パージする代わりに、キャッシュラインを、非コヒーレント・スナップショットの状態に移行させる。ＤＭＡ読み出しのためにフェッチされたキャッシュラインが、前に説明したようなロック状態にあるときは、スヌープの無効化により、キャッシュラインがパージされることはない。その代わりに、キャッシュラインは、キャッシュ３６のＣＰＵ／メモリ・バス２２側からのスヌープに関与しなくなり、ＤＭＡリード・リターン用のデータを供給するために使用されない限りパージされない。しかしながら、データは、ＤＭＡリード・リターンが完了するとすぐにパージされる。

【００３４】

非コヒーレント・スナップショットの状態にあるキャッシュラインからのデータはキャッシュ３６のコヒーレント・プロトコルに含まれないため、そのデータは一般的な用途には適さない。しかし、読み出し要求のときにキャッシュ３６がコヒーレントであったため、非コヒーレント・スナップショットの状態のデータを使用して、最初にデータがフェッチされるようにしたＤＭＡ読み出しを完了することはできる。

【００３５】

本明細書において、本発明を、入出力バス２４のブリッジへの単一のＣＰＵ／メモリ・バス２２に適用されるように説明したが、本発明を１対多(one-to-many)のバスブリッジ環境にも適用できる。すなわち、バスブリッジ５０を使用して、単一のＣＰＵ／メモリ・バス２２にいくつかの入出力バス２４を接続することができる。

【００３６】

１対多のバスブリッジは、様々な構成で実現することができる。まず、各入出力バス２４が、それ自体の独立したキャッシュ３６を有する場合は、それぞれの入出力バス２４ごとに簡単に本発明を複製(duplicate)することができる。すなわち、各入出力バス２４は、前に説明したように個別に動作する１組のＰＰＣ５４、ＣＰＣ５６及びＣＬＡカウンタ５８とそれに割り当てられた順序付けロジック６２とを備える。

【００３７】

代替として、すべての入出力バス２４に１つのキャッシュ３６を使用する場合は、１つのキャッシュラインにつき１つのＣＬＡカウンタ５８を使用するか、１つのキャッシュラインにつき複数のＣＬＡカウンタ５８を使用するか、２つの手法を用いることができる。

【００３８】

最も簡単な構成は、各入出力バス２４ごとに各キャッシュラインに１つのＣＬＡカウンタ５８を割り当てることである。各入出力バス２４は、１組のＰＰＣ５４及びＣＰＣ５６カウンタと順序付けロジック６２とを有する。キャッシュラインデータがフェッチされるとき、そのキャッシュラインと関連したすべてのＣＬＡカウンタ５８は、ゼロにリセットされる。ＰＩＯ書込みがバスブリッジ５０によって受け入れられるときは、その特定の入出力バス２４のＰＰＣ５４は、各キャッシュラインの宛先入出力バスに対応するＣＬＡカウンタ５８と共に増分する。ＤＭＡ読み出しが完了するかを決定するとき、その検査は、ヒットしたキャッシュラインと関連した入出力バス２４の固有のＣＬＡカウンタ５８と、特定の入出力バス２４に関連したＰＰＣ５４及びＣＰＣ５６カウンタを使って適用される。

【００３９】

最後に、１つのキャッシュ３６が複数の入出力バス２４に共用されている場合、各キャッシュラインは、１つの入出力バス２４につき１つのＣＬＡカウンタ５８を使用せずに、１つのＣＬＡカウンタ５８を使用することができる。この構成では、キャッシュラインデータがフェッチされるとき、関連したＣＬＡカウンタ５８がゼロにリセットされ、キャッシュラインが、特定の入出力バス２４に結合(bound)される。したがって、キャッシュラインはすべて、ＤＭＡ読み出しのためのデータの最初のフェッチにより、キャッシュラインが、読み出し動作を開始した入出力バス２４に結合されるように、いくつかの入出力バス２４に結合される。ＣＬＡカウンタ５８は、キャッシュラインが結合される入出力バス２４を対象とするＰＩＯ書込みのためにのみ増分される。したがって、現在のＰＩＯ書込みが目的とする入出力バス２４に結合されたＣＬＡカウンタ５８だけが増分されるように

、インクリメント５９線をフィルタリングする付加的なロジックが含まれる。ロック状態又は非コヒーレント・スナップショット状態でキャッシュラインをヒットする他の入出力バス２４からのＤＭＡ読み出し要求は、キャッシュラインがその状態にある限り一時中止状態に留まる。キャッシュラインが、ロック状態か又は非コヒーレント・スナップショット状態でなくなると、ＤＭＡ読み出しを開始した入出力バス２４に再結合される。しかしながら、キャッシュラインが非コヒーレント・スナップショット状態でなくなると、すぐにパージされ、新しいデータが、メモリ３２からフェッチされる。ロック状態又は非コヒーレント・スナップショット状態でキャッシュラインをヒットする他の入出力バス２４からのＤＭＡ読み出し要求によって、キャッシュラインが再結合され、入出力バスがＤＭＡ読み出し要求を出す。

【００４０】

結合するプロセスは、ＣＬＡカウンタ５８をゼロにリセットし、後で説明するプロセスを使用してＤＭＡ読み出し完了を表示する段階を有する。結合するときに処理中のＰＩＯ書込みは、キャッシュラインデータの送信時よりも長く未処理になるように控えめに想定される。前記のように、ロックされた非コヒーレント・スナップショット状態の線は、進行を保証するために、解放されるまで再結合することができない。

【００４１】

当業者には、本発明の原理から実質上逸脱することなしに好ましい実施形態に多くの変形及び変更を行うことができることが明らかである。たとえば、本発明は、プロセッサ／メモリ・バスと少なくとも１つの入出力バスとの間の通信を容易にするためにバスブリッジにおいて実施するように説明した。しかし、本発明は、ブリッジの一方の側から支配された書込み用の書込み要求記憶領域と、ブリッジの反対の側から支配された読み出しサイクルに応えるキャッシュやその他のコヒーレントなプリフェッチされたリソースとを、ある任意の構成に適用することができる。そのような変形と修正はすべて、併記の特許請求の範囲で説明するような本発明の範囲内に含まれるものである。

【００４２】

【発明の効果】

以上の説明から明らかなように、本発明によれば、キャッシュからの読み出し動作が、適切なプロデューサーコンシューマ順序を維持するために必要な最低限の時間を遅延することができ、２バス構成及び方法に対して最低限の修正を加えるか修正を加えないかにかかわらず１対多のバスブリッジ構成に容易に拡張することができる。

【図面の簡単な説明】

【図１】本発明の一実施形態に係るプロセッサ／メモリ・バスと入出力バスとの間の通信を容易にするバスブリッジの構成を示すブロック図である。

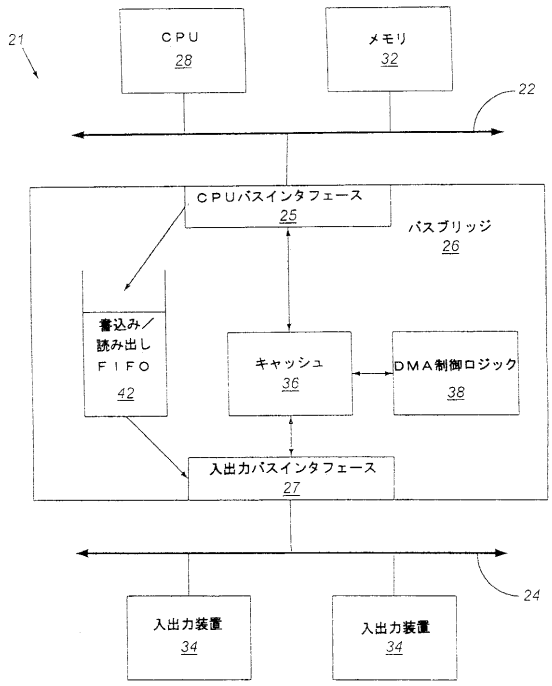
【図２】本発明の一実施形態に係るＤＭＡ読み出しの完了時間をより選択的に遅延させる図１のバスブリッジを示す詳細なブロック図である

【図３】本発明によりプロデューサーコンシュマ順序付けを保証する図２のハードウェア構成要素におけるバスブリッジ５０の動作を示すブロック図である。

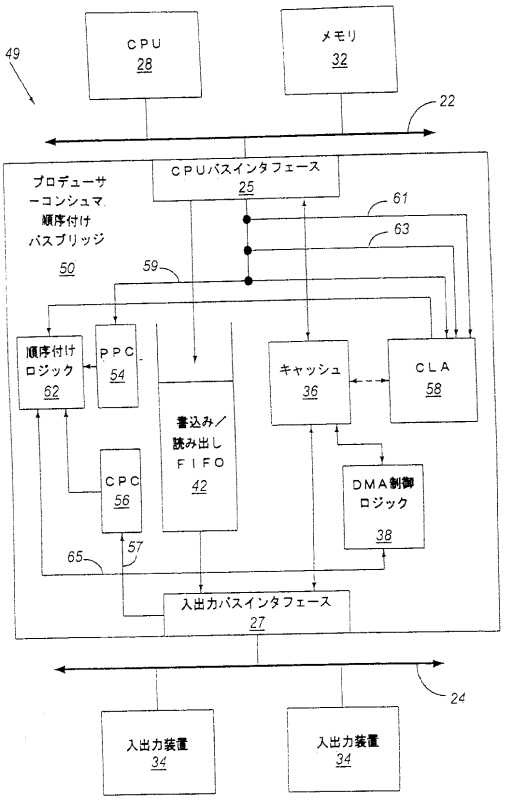
【符号の説明】

２２　ＣＰＵ／メモリ・バス
２４　入出力バス
３６　キャッシュ
４２　書込み／読み出しＦＩＦＯ記憶領域
５０　バスブリッジ
５４　ＰＰＣ
５６　ＣＰＣ
５８　ＣＬＡカウンタ
６２　順序付けロジック
６１　リセット信号線
６３　リセット選択信号線

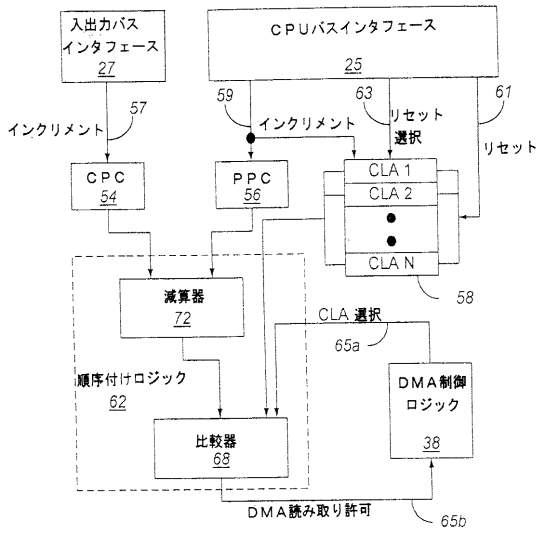
【図１】


21
CPU
28
メモリ
32
22
CPUバスインタフェース
25
バスブリッジ
26
書込み／
読み出し
FIFO
42
キャッシュ
36
DMA制御ロジック
38
入出力バスインタフェース
27
24
入出力装置
34
入出力装置
34

【図２】


49
CPU
28
メモリ
32
22
プロデューサ
ーコンシュマ
順序付け
バスブリッジ
50
CPUバスインタフェース
25
61
63
59
順序付け
ロジック
62
PPC
54
書込み／
読み出し
FIFO
42
キャッシュ
36
CLA
58
CPC
56
DMA制御
ロジック
38
65
57
入出力バスインタフェース
27
24
入出力装置
34
入出力装置
34

【図３】


入出力バス
インタフェース
27
CPUバスインタフェース
25
59
63
61
インクリメント
57
インクリメント
リセット
選択
リセット
CPC
54
PPC
56
CLA 1
CLA 2
CLA N
58
減算器
72
CLA 選択
65a
順序付けロジック
62
DMA制御
ロジック
38
比較器
68
DMA読み取り許可
65b

フロントページの続き

(72)発明者　トーマス・ブイ・スペンサー
アメリカ合衆国　コロラド州，フォート・コリンズ，モッキンバード・ドライブ　８０３１
(72)発明者　フランシスコ・コレラ
アメリカ合衆国　カリフォルニア州，サクラメント，キンカイド・ウェイ　１９２１

審査官　五十嵐　努

(56)参考文献　特開平０３－０４８３５５（ＪＰ，Ａ）
特開平０９－１９８３３９（ＪＰ，Ａ）
特開平０９－２３１１６４（ＪＰ，Ａ）

(58)調査した分野(Int.Cl.[7]，ＤＢ名)
G06F 13/36　310
G06F 12/08