

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第4565786号
(P4565786)

(45) 発行日 平成22年10月20日(2010.10.20)

(24) 登録日 平成22年8月13日(2010.8.13)

(51) Int.Cl.

G 0 6 F 13/362 (2006.01)

F 1

G O 6 F 13/362 5 2 O B

請求項の数 25 (全 17 頁)

(21) 出願番号 特願2001-505301 (P2001-505301)
 (86) (22) 出願日 平成12年1月11日 (2000.1.11)
 (65) 公表番号 特表2003-527652 (P2003-527652A)
 (43) 公表日 平成15年9月16日 (2003.9.16)
 (86) 國際出願番号 PCT/US2000/000605
 (87) 國際公開番号 WO2000/079404
 (87) 國際公開日 平成12年12月28日 (2000.12.28)
 審査請求日 平成18年11月20日 (2006.11.20)
 (31) 優先権主張番号 09/339,351
 (32) 優先日 平成11年6月23日 (1999.6.23)
 (33) 優先権主張国 米国(US)

(73) 特許権者 591016172
 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
 A D V A N C E D M I C R O D E V I
 C E S I N C O R P O R A T E D
 アメリカ合衆国、94088-3453
 カリフォルニア州、サンディベイル、ビイ・
 オウ・ボックス・3453、ワン・エイ・
 エム・ディ・ブレイス、メイル・ストップ
 ・68 (番地なし)
 (74) 代理人 100064746
 弁理士 深見 久郎
 (74) 代理人 100085132
 弁理士 森田 俊雄

最終頁に続く

(54) 【発明の名称】ロックの開始と終了を要求するためのバスコマンドコードポイントの使用の最小化

(57) 【特許請求の範囲】

【請求項 1】

プロセッサであって、

ロックの要求またはロック解除の要求に応答して第1の要求を生成するように構成される第1の回路と、

前記第1の回路に結合されるシステムインターフェイスコントローラとを含み、前記システムインターフェイスコントローラは、前記第1の要求に応答して第1のコマンドを伝送してロック状態を切替えるように構成される、プロセッサ。

【請求項 2】

前記第1の回路は前記第1のコマンドの後に1つまたは2つ以上のメモリアクセスを実行するように構成され、前記1つまたは2つ以上のメモリアクセスはロックされた状態で行われる、請求項1に記載のプロセッサ。 10

【請求項 3】

前記システムインターフェイスコントローラは、前記第1の回路に前記第1のコマンドがコミットされたと表示するように構成され、前記第1の回路は前記システムコントローラからの表示に応答して前記1つまたは2つ以上のメモリアクセスを実行するように構成される、請求項2に記載のプロセッサ。

【請求項 4】

前記第1の回路は、前記第1の要求を生成し、前記第1のコマンドの後に1つまたは2つ以上のメモリアクセスを行い、かつ前記1つまたは2つ以上のメモリアクセスの後に再 20

び前記第1の要求を生成することにより、ロックされたシーケンスを行うように構成される、請求項1に記載のプロセッサ。

【請求項5】

前記第1の回路は、前記第1の回路内で最も古い未処理の要求である前記ロックの要求またはロック解除の前記要求に応答して、前記第1の要求を生成するように構成される、請求項1に記載のプロセッサ。

【請求項6】

ロックを特定する第1の1つまたは2つ以上の命令を受取りさらに復号化するように結合される復号ユニットをさらに含み、前記復号ユニットは、前記第1の1つまたは2つ以上の命令に応答して、ロック指令を生成するように構成され、

10

ロック指令はロックに対する要求を示し、前記第1の回路は、前記ロック指令と前記第1の1つまたは2つ以上の命令とを前記復号ユニットから受取るように結合されるロード/記憶ユニットを含み、前記ロード/記憶ユニットは、実行のために前記第1の1つまたは2つ以上の命令を選択し、さらに、ロックを得るためにロック指令に応答してロックトグルコマンドの第1の要求を伝送するように構成される、請求項1に記載のプロセッサ。

【請求項7】

前記システムインターフェイスコントローラは複数のバッファをさらに含み、
使用中の待ち行列エントリをトラッキングするように構成されるリソースカウンタと、
未処理のプロセッサコマンドをトラッキングするように構成されるコミットカウンタと
を含む、請求項6に記載のプロセッサ。

20

【請求項8】

前記システムインターフェイスコントローラはさらに、プロセッサコマンドを発行した
ことに応答して、前記リソースカウンタを増分するように構成され、

前記システムインターフェイスコントローラはさらに、前記プロセッサコマンドを発行
したことに応答して、前記コミットカウンタを増分するように構成される、請求項7に記
載のプロセッサ。

30

【請求項9】

前記システムインターフェイスコントローラはさらに、システムデバイスからアクノレッジ信号を受取ったことに応答して、前記リソースカウンタを減分するように構成され、
前記システムインターフェイスコントローラはさらに、前記システムデバイスからコミット信号を受取ったことに応答して、前記コミットカウンタを減分するように構成される、
請求項8に記載のプロセッサ。

【請求項10】

前記アクノレッジ信号は、待ち行列エントリを解放したことに応答して、前記システム
デバイスによって送られる、請求項9に記載のプロセッサ。

【請求項11】

前記コミット信号は、特定のプロセッサコマンドに対してコピー・レンシ・ポイントに到
達したことに応答して、前記システムデバイスによって送られる、請求項9に記載のプロ
セッサ。

40

【請求項12】

バスブリッジであって、

ロック状態を記憶するように構成されるロック記憶装置と、

前記ロックレジスタに結合され、さらに第1のプロセッサから第1のコマンドを受取つ
て前記ロック状態を切替えるように結合される制御論理とを含み、前記制御論理は、前記
ロックトグルコマンドと、ロック状態がロックの欠如を表示すると判断したことに応答し
て、前記ロックレジスタ内の前記ロック状態を設定し前記第1のプロセッサによるロック
を表示するように構成され、前記制御論理は、前記第1のコマンドと、前記ロック状態が
前記第1のプロセッサによる前記ロックを表示すると判断したことに応答して、前記ロ
ック状態を設定しロックの欠如を表示するように構成される、バスブリッジ。

【請求項13】

50

前記制御論理はさらに、前記ロック状態が前記第1のプロセッサのための前記ロックを表示している間は、前記第1のプロセッサから受取ったコマンドのみを実行するように構成される、請求項1_2に記載のバスプリッジ。

【請求項14】

前記ロック記憶装置はロックレジスタである、請求項1_2に記載のバスプリッジ。

【請求項15】

前記制御論理はさらに、第2のプロセッサから前記第1のコマンドを受取ったことに応答して、前記ロック状態が前記ロックの欠如を表示する場合には、前記ロック状態を設定して第1のプロセッサと異なる前記第2のプロセッサのためのロックを表示するように構成される、請求項1_2に記載のバスプリッジ。 10

【請求項16】

前記第1のプロセッサから複数のコマンドを受取るように結合される第1の入力ポートと、

前記第1の入力ポートに結合される第1のプロセッサ待ち行列とをさらに含み、前記第1の入力ポートは前記複数のコマンドを前記第1のプロセッサ待ち行列へと転送するように構成され、前記第1のプロセッサ待ち行列は前記複数のコマンドを記憶するように構成され、前記複数のコマンドは前記ロックトグルコマンドを含み、

前記制御論理は前記複数のコマンドを第1のプロセッサ待ち行列から取除くように構成され、前記ロックトグルコマンドを前記第1のプロセッサ待ち行列から取除いたことに応答して、前記制御論理は、ロック状態を検査するように構成され、前記制御論理はさらに、前記ロック状態がロックの欠如を表示する場合には、前記ロック状態を設定して前記第1のプロセッサのためのロックを表示するように構成され、前記ロック状態が前記第1のプロセッサに対して前記ロックを表示する場合には、前記ロック状態を設定して前記ロックの欠如を表示するように構成される、請求項1_2に記載のプリッジ。 20

【請求項17】

第2のプロセッサから第2の複数のコマンドを受取るように結合される第2の入力ポートと、

前記第2の入力ポートに結合される第2のプロセッサ待ち行列とを含み、前記第2の入力ポートは前記第2の複数のコマンドを前記第2のプロセッサ待ち行列へと転送するように構成され、前記第2のプロセッサ待ち行列は前記第2の複数のコマンドを記憶するように構成され、 30

前記制御論理はさらに、予め定められたプロトコルに従って、前記第1のプロセッサ待ち行列および前記第2のプロセッサ待ち行列からコマンドを選択するように構成され、前記制御論理はさらに、前記ロック状態が前記第1のプロセッサのための前記ロックを表示するとき前記第1のプロセッサ待ち行列からのみコマンドを選択するように構成される、請求項1_6に記載のプリッジ。

【請求項18】

前記第1および第2のプロセッサの各々は、別のプロセッサバスによって前記プリッジに結合される、請求項1_7に記載のプリッジ。

【請求項19】

コンピュータシステムであって、

1つまたは2つ以上のプロセッサと、

前記1つまたは2つ以上のプロセッサに結合されるプリッジとを含み、

前記1つまたは2つ以上のプロセッサのうちの第1のプロセッサは第1のコマンドを前記プリッジへと伝送して、前記プリッジ内のロック状態を切替えるように構成され、前記プリッジは、前記1つまたは2つ以上のプロセッサのうちの前記第1のプロセッサから前記第1のコマンドを受取ったことに応答して、前記ロック状態がロックの欠如を表示する場合には、前記ロック状態を設定して前記1つまたは2つ以上のプロセッサのうちの前記第1のプロセッサのための前記ロックを表示し、さらには前記ロック状態が前記1つまたは2つ以上のプロセッサのうちの前記第1のプロセッサのための前記ロックを表示する場 50

合には、前記ロック状態を設定して前記ロックの欠如を表示するように構成される、コンピュータシステム。

【請求項 2 0】

前記ブリッジはさらに、前記ロック状態が前記1つまたは2つ以上のプロセッサのうちの前記第1のプロセッサのための前記ロックを表示している間は、前記1つまたは2つ以上のプロセッサのうちの前記第1のプロセッサから受取ったコマンドのみを実行するよう構成される、請求項1_9に記載のコンピュータシステム。

【請求項 2 1】

前記ブリッジは、前記ロック状態を記憶するように構成されるロックレジスタを含む、請求項1_9に記載のコンピュータシステム。

10

【請求項 2 2】

前記ブリッジは、前記1つまたは2つ以上のプロセッサのうちの前記第2のプロセッサから前記第1のコマンドを受取ったことに応答して、前記ロック状態が前記ロックの欠如を表示する場合には、前記ロック状態を設定して前記1つまたは2つ以上のプロセッサのうちの第2のプロセッサのためのロックを表示するように構成される、請求項1_9に記載のコンピュータシステム。

【請求項 2 3】

1つまたは2つ以上のプロセッサとシステムデバイスとを含むコンピュータシステムを動作させるための方法であって、

前記1つまたは2つ以上のプロセッサのうちの第1のプロセッサから前記システムデバイスへと第1のロックトグルコマンドを発行するステップと、

20

前記第1のロックトグルコマンドに応答して、前記システムデバイス内のロック状態の表示を検査するステップと、

前記ロック状態の前記表示を検査したことおよび前記ロック状態がロックの欠如を表示すると判断したこととに応答して、前記ロック状態を設定し前記第1のプロセッサによるロックを表示するステップと、

前記ロック状態の前記表示を検査したことおよび前記ロック状態が前記ロックを表示すると判断したこととに応答して、前記ロック状態を設定し前記ロックの欠如を表示するステップとを含む、方法。

【請求項 2 4】

30

前記ロック状態が前記第1のプロセッサのための前記ロックを表示する間は、前記1つまたは2つ以上のプロセッサのうちの第2のプロセッサからのコマンドを無視するステップをさらに含む、請求項2_3に記載の方法。

【請求項 2 5】

前記ロック状態を設定して前記ロックを表示したことに応答して、前記ロックを特定する前記1つまたは2つ以上のコマンドを、前記第1のプロセッサから前記システムデバイスへと送るステップと、

前記ロックを特定する前記1つまたは2つ以上のコマンドを、前記第1のプロセッサから前記システムデバイスへと送った後に、第2のロックトグルコマンドを、前記第1のプロセッサから前記システムデバイスへと送るステップとをさらに含む、請求項2_3に記載の方法。

40

【発明の詳細な説明】

【0 0 0 1】

【技術分野】

この発明は、通信インターフェイス、より具体的にはロック状態の開始または終了のいずれかを要求するためのロックトグルコマンドに関する。

【0 0 0 2】

【背景技術】

コンピュータシステムでは、特に、メモリまたはI/O(入力/出力)スペースに、並行にアクセスしてもよい多数のプロセッサを含むコンピュータシステムでは、メモリへのア

50

トミックアクセスが他のシステムデバイスによって割り込まれないことを保証するあるメカニズムが必要とされる。たとえば、メモリに基づく変数の場合などでは、第1のシステムデバイスが、メモリ内のある場所から読出し、その同じメモリ場所に新しい値を書きもうとするであろう。第1のシステムデバイスがメモリ場所でのオペレーションを終了する前に、同じメモリ場所に、第2のシステムデバイスが読み出または書きをすることができないようにシステムを「ロック」するあるメカニズムが必要とされている。一般に、ロックオペレーションは、所与のデバイスから所与のメモリ場所または範囲への1つまたは2つ以上の読み出サイクルとそれに続く1つまたは2つ以上の書きサイクルのシーケンスであると定義することができる。ロックオペレーションを含むそのシーケンスの間、他のデバイスは1つも、少なくとも所与のメモリ場所にアクセスしない。

10

【0003】

x86プロセッサでは、LOCKプレフィックスを用いるある命令にロッキング機能が設けられている。ある他の命令は、メモリの読み出および書きがロックされることを暗に指定する。ロックオペレーションはまた、ページ表更新およびインタラプトアクノレッジサイクルをも含んでもよいことが注目される。x86ハードウェアでは、ロッキングは伝統的にx86プロセッサ上のLOCKピンを通して実現してきた。ロックオペレーションを行なうプロセッサは、ロックオペレーションを含む、読み出および書きシーケンスの間、LOCKピンをアサートする。x86プロセッサは一般に、プロセッサのメモリへのアクセスが单一の共有プロセッサバスを通して提供されるコンピュータシステム内に設計されてきたため、LOCKピンアサーションを用いて共有バスリソースをロッキングプロセッサ専用にすることができる。共有バスがロックされている間、バスリソースを共有する他のプロセッサは共有バスにアクセスすることを禁じられるため、他のプロセッサは、読み出と書きとのアトミックなシーケンスに割り込むことができない。

20

【0004】

残念なことに、共有バスシステムにはいくつかの欠点がある。たとえば、多数のデバイスが共有バスに接続されているため、バスは通常比較的低い周波数で動作する。多数の接続は、バス上で信号を駆動するデバイスに高容量性負荷を与え、多数の接続ポイントは、高周波数では比較的複雑な伝送路モデルを与える。したがって、周波数は低いままでとどまり、共有バス上で利用できる帯域幅は同様に比較的狭い。狭い帯域幅により、共有バスへの追加のデバイスの接続が妨げられる。なぜならば、利用できる帯域幅によって性能が制限されるおそれがあるからである。

30

【0005】

共有バスシステムにおける別の不利な点は、より多数のデバイスへのスケーラビリティの欠如である。上述のように、帯域幅の大きさは固定されている（追加のデバイスを付加することで、バスの動作可能周波数が減じられると、それはより狭くなるおそれもある）。バスに（直接または間接的に）接続されているデバイスの帯域幅要件が、バスの利用可能な帯域幅を一旦超えると、バスへのアクセスを試みると、デバイスは頻繁に機能停止する。このようにして、コンピュータシステムの全体の性能が衰えるおそれがある。

【0006】

x86プロセッサは、動作周波数および全体の性能において向上し続けているため、共有コンピュータバスコンピュータシステムモデルは、性能面での限界となってきている。それゆえに、非共有バスシステム内でロッキング機能を提供するための方法が望まれる。

40

【0007】

【発明の開示】

上述の問題は、ロックを設定し解除するためのトグルコマンド、つまりロックトグルを用いるためのシステムおよび方法によって大部分が解決される。例示的なコンピュータシステムでは、1つまたは2つ以上のプロセッサの各々がバスブリッジに、別々の高速接続を通して結合され、その各々は、ある実施例では、それぞれがソース同期クロック線を備える一対の单方向アドレスバスと、付随のソース同期クロック線を備える双方向データバスとを含む。ロックトグルコマンドを用いて、プロセッサからシステムコヒーレンシィポイ

50

ント、たとえばバスブリッジへとロック要求およびアンロック要求の両方を伝送する。ロックが確立または解除されたとき、システムコヒーレンシィポイントは肯定応答する。ロックがアクティブな間は、他のプロセッサは、少なくともそれについてロックが開始されたメモリ場所にアクセスすることを禁じられる。このようにして、ロックはシステムコヒーレンシィポイントで確立され、このことは、有利なことには、非共有バスシステム内のロック機能を可能にするであろう。ロックトグルコマンドの使用はまた、有利なことには、単一のコマンドコードポイントの使用を可能にし、他のポイントを他の用途に利用できるようにするであろう。

【0008】

概して、復号ユニット、ロード／記憶ユニット、およびシステムインターフェイスコントローラを含むプロセッサが企図される。復号ユニットは、ロックを特定する第1の1つまたは2つ以上の命令を受取りそれを復号化するように結合され。復号ユニットは、第1の1つまたは2つ以上の命令に応答して、ロック表示を生成するように構成される。ロード／記憶ユニットは、ロック表示と、第1の1つまたは2つ以上の命令とを復号ユニットから受取るように結合される。ロード／記憶ユニットは、第1の1つまたは2つ以上の命令を選択して実行し、さらにそれに応答してロックトグルコマンドの第1の要求を伝送するように構成される。システムインターフェイスコントローラは、ロード／記憶ユニットとバスとの間に結合される。システムインターフェイスコントローラは、ロックトグルコマンドの第1の要求をロード／記憶ユニットから受取り、さらにロックトグルコマンドの第1の要求を受取ったことに応答して、第1のロックトグルコマンドを発行するように構成される。ロード／記憶ユニットはさらに、第1の1つまたは2つ以上の命令を実行したことに応答して、ロックトグルコマンドの第2の要求を伝送するように構成される。システムインターフェイスコントローラはさらに、ロックトグルコマンドの第2の要求をロード／記憶ユニットから受取り、さらにロックトグルコマンドの第2の要求を受取ったことに応答して、第2のロックトグルコマンドを発行するように構成される。

10

20

【0009】

1つまたは2つ以上のプロセッサをコンピュータシステムに結合するためのブリッジもまた企図される。概して、ブリッジは、1つまたは2つ以上のプロセッサのうちの第1のプロセッサから複数のコマンドを受取るように結合される第1の入力ポートと、第1の入力ポートに結合される第1のプロセッサ待ち行列と、ロック状態を記憶するように構成されるロックレジスタと、第1の入力待ち行列とロックレジスタとに結合される制御論理とを含む。第1の入力ポートは、前記1つまたは2つ以上のプロセッサのうちの第1のプロセッサから第1のプロセッサ待ち行列へと複数のコマンドを転送するように構成される。第1のプロセッサ待ち行列は、1つまたは2つ以上のプロセッサのうちの第1のプロセッサからの複数のコマンドを記憶するように構成される。1つまたは2つ以上のプロセッサのうちの第1のプロセッサからの複数のコマンドは、ロックトグルコマンドを含む。制御論理は、1つまたは2つ以上のプロセッサのうちの第1のプロセッサからの複数のコマンドを、第1のプロセッサ待ち行列から取除くように構成される。ロックトグルコマンドを第1のプロセッサ待ち行列から取除いたことに応答して、制御論理は、ロック状態を検査するように構成される。制御論理はさらに、ロック状態がロックの欠如を表示する場合には、ロック状態を設定して1つまたは2つ以上のプロセッサのうちの第1のプロセッサのためのロックを表示するように構成される。制御論理はさらに、ロック状態がロックを表示する場合には、ロック状態を設定してロックの欠如を表示するように構成される。

30

40

【0010】

コンピュータシステムもまた企図される。概して、コンピュータシステムは、1つまたは2つ以上のプロセッサと、1つまたは2つ以上のプロセッサに結合されるブリッジとを含む。ブリッジは、1つまたは2つ以上のプロセッサから受取るコマンドを実行するように構成される。1つまたは2つ以上のプロセッサの各々は、ロックトグルコマンドをブリッジへと伝送して、ロック状態がロックを表示するように設定されることを要求するように構成される。ブリッジは、1つまたは2つ以上のプロセッサのうちの第1のプロセッサか

50

らロックトグルコマンドを受取ったことに応答して、ロック状態を検査するように構成される。制御論理はさらに、ロック状態がロックの欠如を表示する場合には、ロック状態を設定して1つまたは2つ以上のプロセッサのうちの第1のプロセッサのためのロックを表示し、さらにはロック状態がロックを表示する場合には、ロック状態を設定してロックの欠如を表示するように構成される。

【0011】

1つまたは2つ以上のプロセッサとシステムデバイスとを含むコンピュータシステムを動作させるための方法もまた企図される。概して、その方法は、1つまたは2つ以上のプロセッサのうちの第1のプロセッサからシステムデバイスへと第1のロックトグルコマンドを発行するステップを含む。その方法はさらに、第1のロックトグルコマンドに応答して、システムデバイス内のロック状態の表示を検査する。その方法はさらに、ロック状態の表示を検査してロック状態がロックの欠如を表示すると判断したことに応答して、ロック状態を設定して第1のプロセッサによるロックを表示するステップを含む。その方法はまた、ロック状態の表示を検査してロック状態がロックを表示すると判断したことに応答して、ロック状態を設定してロックの欠如を表示する。

10

【0012】

この発明の他の目的および利点は、以下の詳細な説明を読み、さらには添付の図を参照することによって、明らかになるであろう。

【0013】

この発明には、種々の変形および代替の形の余地があるが、図における例によってその具体的な実施例を示し、それらをここで詳細に説明する。しかし、図および詳細な説明は、開示された特定の形式に発明を制限することを意図するのではなく、逆に、添付の請求項によって規定されるようなこの発明の思想と範囲とに入るすべての変形、均等物、および代替物をカバーすることを意図することが理解されるべきである。

20

【0014】

【発明の実施の態様】

図1を参照すると、一般的なコンピュータシステムの実施例のブロック図が示される。第1のプロセッサ110Aと第2のプロセッサ110Bとの各々は、別々のプロセッサバスを通してブリッジ130に結合される。第1のプロセッサ110Aと第2のプロセッサ110Bとは共に、それぞれのプロセッサバスを用いてメモリおよびI/Oオペレーションを行なうように、好ましくは構成される。ある実施例では、プロセッサ110Aおよび110Bは、×86命令セットアーキテクチャを実現する。他の実施例はいかなる好適な命令セットアーキテクチャを実現してもよい。ブリッジ130はさらにメモリ140に結合される。メモリ140は好ましくは、第1のプロセッサ110Aと第2のプロセッサ110Bとの両方にさらには他のシステムデバイスにもアクセス可能なデータと命令とを記憶するように構成される。メモリ140は、SDRAM(同期ダイナミックランダムアクセスメモリ)、RDRAM(ランバスDRAM)[RDRAMおよびRAMBUSはランバス社(Rambus, Inc.)の登録商標である]、または他のいかなる好適なメモリタイプからなっていてもよい。アドバンスドグラフィックスポートデバイス(advanced graphics port device)(AGP)150もまた任意でブリッジ130に結合される。図示されるように、ペリフェラルコンポーネントインターフェクト(Peripheral Component Interconnect)(PCI)バス160もまたブリッジ130に結合される。種々のI/OコンポーネントがPCIバス160に結合されてもよい。

30

【0015】

インダストリスタンダードアーキテクチャ(ISA)バス等の従来のバスを含むコンピュータシステムの実施例では、ブリッジ130はしばしばノースブリッジ130と呼ばれ、PCIバス160と従来のバスとの間のブリッジ(図示せず)はサウスブリッジと呼ばれることが注目される。例示される実施例では、ブリッジ130はコンピュータシステムのシステムマスターであることもまた注目される。例示される実施例は2つのプロセッサ110Aおよび110Bを含むが、所望であれば、いかなる数のプロセッサ110もコンピュ

40

50

ータシステムに含まれ得ることが注目される。

【0016】

例示される実施例では、ブリッジ130はシステムマスタとして動作し、プロセッサ110Aおよび110B、メモリ140、AGPデバイス150、およびPCIバス等の間の通信を調整する。ブリッジ130は、プロセッサ110Aおよび/または110Bを探索して、他のプロセッサ110Aまたは110B、AGPデバイス150、またはPCIバス160上のPCIデバイス等によってアクセスされるメモリ場所があるか調べることによって、コンピュータシステムのデバイス間のデータ転送のためのコピーレンシィを維持する。

【0017】

図2を参照すると、図1の一般的なコンピュータシステムの、ある構成要素の実施例のより詳細なブロック図が例示される。第1のプロセッサ110A、プロセッサバスコンポーネント126Aおよび128A、およびブリッジ130の種々の詳細が例示される。

【0018】

図示されるように、プロセッサ110Aは、L/Sコマンドバスおよびロック信号線を通して、ロード/記憶ユニット114に結合される復号ユニット112を含む。ロード/記憶ユニット114は、アドレスおよびデータバス、ロックトグル要求信号線、およびロックトグルグラント信号線を通して、システムインターフェイスコントローラ116に結合される。システムインターフェイスコントローラ116は、リソースカウンタ118(Aカウンタ)とコミットカウンタ119(Cカウンタ)とを含む。第1のプロセッサ110Aおよび第2のプロセッサ110Bの各々は、別々のプロセッサバスを通してブリッジ130に結合される。各プロセッサバスは、専用のソース同期クロック線126を備える双方向データバスと、付随のソース同期クロック線128を備える単方向アドレスインアドレスアウト線(アドレスバス)とを含む。第1のプロセッサ110Aとブリッジ130との間のプロセッサバスは、データバス126Aとアドレスバス128Aとを含み、少なくとも第1の入力ポートを通してブリッジ130へと結合する。第2のプロセッサ110Bとブリッジ130との間のプロセッサバスは、データバス126Bとアドレスバス128Bとを含み、少なくとも第2の入力ポートを通してブリッジに結合する。ブリッジ130は、第1のプロセッサ110Aからのコマンドの記憶に専用の第1のプロセッサ待ち行列134Aと、第2のプロセッサ110Bからのコマンドの記憶に専用の第2のプロセッサ待ち行列134Bとを含む。メモリ140もまたブリッジ130に結合されて示される。

【0019】

一般に、ロックが表示されない場合、コンピュータシステムの動作は以下のとおりである：プロセッサ110Aの復号ユニット112は、命令を受取りそれを復号化する。復号ユニット112は、メモリオペレーション(メモリへのロードまたは記憶を指定する命令)をロード/記憶ユニット114に運ぶ。ロード/記憶ユニット114は、メモリオペレーションを内部データキャッシュ(図示せず)に運んでもよく、システムサービスを必要とするメモリオペレーションを、システムインターフェイスコントローラ116へ運ぶ。システムインターフェイスコントローラ116は、アドレスバス128Aおよびデータバス126Aを用いて、ブリッジ130等のシステムデバイスにオペレーションを運ぶ。より具体的には、システムインターフェイスコントローラ116は、アドレスバス128Aのアドレスアウト部分上に、アドレスおよびコマンドインフォメーションを伝送する。アドレスバス128Aのアドレスイン部分上で、ブリッジ130は、対応するデータ転送の準備ができたことを合図する。そこで対応するデータがデータバス126A上で伝送される。

【0020】

ブリッジ130に送られたコマンドの大部分について、システムインターフェイスコントローラ116は、リソースカウンタ118(Aカウンタ)とコミットカウンタ119(Cカウンタ)とを増分する。ブリッジ130内の制御論理136は、第1のプロセッサ110Aおよび第2のプロセッサ110Bからコマンドを受取り、コマンドをそれぞれの待ち

10

20

30

40

50

行列、プロセッサ A 待ち行列 134A とプロセッサ B 待ち行列 134B とに置く。制御論理 136 は、予め定められたプロトコルによって規定された順序で、つまりラウンドロビンまたは最終アクセスアルゴリズムに従って、プロセッサ A 待ち行列 134A およびプロセッサ B 待ち行列 134B からコマンドを解除する。プロセッサ 110B もまた、同様の様態で動作する。リソースカウンタ 118 および / またはコミットカウンタ 119 はある種のコマンドをトラッキングしなくてもよいということが注目される。トラッキングされないコマンドの例は、ある種のプロープ応答、ある種のバッファフラッシュ、ノーオペレーションコマンド (NOPS)、および特別なブロックメモリコマンドを含み得る。

【0021】

プロセッサ A 待ち行列 134A からコマンドを解除する等によって、待ち行列エントリを解放したことに応答して、制御論理 136 は、それぞれのプロセッサ 110A のシステムインターフェイスコントローラ 116 に肯定応答信号を送る。システムインターフェイスコントローラ 116 は、ブリッジ 130 から肯定応答信号を受取ったことに応答して、リソースカウンタ 118 を減分する。システムマスターとして働くブリッジ 130 が特定のプロセッサコマンドに対してコヒーレンシィポイントに一旦到達すると、制御論理 136 は、コミット信号を適切なシステムインターフェイスコントローラ 116 に送る。システムインターフェイスコントローラ 116 は、コミット信号をブリッジ 130 から受取ったことに応答して、コミットカウンタ 119 を減分するように構成される。好ましい実施例では、プロセッサ 110 は、コミット信号をいかなる特定のプロセッサコマンドにも関連させないことが注目される。プロセッサコマンドのいずれかがコヒーレンシィポイントに到達したとき、ブリッジ 130 は単にコミット信号を返すのみである。アクノレッジ信号およびコミット信号は、アドレスバス 128A のアドレスイン部分上のアドレスインコマンドの一部であってもよい。

10

20

【0022】

ある実施例では、リソースカウンタ 118 および / またはコミットカウンタ 119 内の値が予め定められた値に到達すると、プロセッサ 110A および 110B は、新しいコマンドをブリッジ 130 に送ることをやめるように構成されている。リソースカウンタ 118 は、各プロセッサ 110 に、ブリッジ 130 に送ったコマンドのうちいくつがブリッジ 130 によって肯定応答されていないのかを、トラッキングさせてもよい。コミットカウンタ 119 は、各プロセッサ 110 に、コヒーレンシィポイントにまだ到達していない、ブリッジ 130 に送られた未処理のコマンドの数を制限させてもよい。

30

【0023】

種々の実施例において、リソースカウンタ 118 の用途と解釈とは異なっていてもよいことが注目される。ある実施例では、4 エントリ共通プロセッサ待ち行列 134 は、一方は読み出のためであり他方は書きのためである、2 つより大きな待ち行列を送る。肯定応答リミット（まだ肯定応答されていない、プロセッサ 110 が発行したコマンドの数に対するリミット）は、4 に設定される。別の実施例では、1 つのプロセッサ待ち行列は、読み出と書きとの両方を保持する。共通の待ち行列エントリの割当が解除されたとき、A ピットがブリッジ 130 からプロセッサ 110 へと返される。

【0024】

40

概して、1 つまたは 2 つ以上のプロセッサ 110 のうちの第 1 のプロセッサ 110A が、メモリ 140 へのアトミックアクセスを行なうとするか、またはロックを必要とする 1 つまたは 2 つ以上の何らかの他のオペレーションを行なうとするとき、第 1 のプロセッサ 110A は、ブリッジ 130 等のシステムマスターに、第 1 のロックトグルコマンドを送る。第 1 のロックトグルコマンドが完遂されさらにロックが表示されるまでは、第 1 のプロセッサ 110A は通常それ以上のコマンドをブリッジ 130 に送ることを避けるが、ある状況下では、ロックトグルコマンドの後に、ある種のコマンドを送ることができる。第 1 のロックトグルコマンドが到達するまでは、ブリッジ 130 は 1 つまたは 2 つ以上のプロセッサ待ち行列 134 からコマンドを選択するように動作する。第 1 のロックトグルコマンドを実行する際に、ブリッジ 130 はロックレジスタ 132 内のロック状態を調べ、

50

第1のプロセッサ110Aのためのロックを開始し、さらには第1のプロセッサ110Aにロックトグルコマンドを完遂したことを知らせる。ブリッジ130は、ロック状態が第1のプロセッサ110Aのためのロックを表示している間は、他のすべてのプロセッサ110からのコマンドを無視する。第1のプロセッサ110Aは、ロックを指定した1つまたは2つ以上の命令を、実行のために、ブリッジ130に伝送する。第1のプロセッサ110Aは、その後また第2のロックトグルコマンドをブリッジ130へと伝送し、ロック状態をアンロックに設定する。ロックを指定した命令を完了させさらに第2のロックトグルコマンドを実行した後、ブリッジ130は第1のプロセッサ110Aにロック状態が再びアンロックに設定されたことを知らせる。そこでブリッジ130は、予め定められたプロトコルに従って、1つまたは2つ以上のプロセッサ待ち行列134のすべてからコマンドを選択することに戻る。

10

【0025】

例示されるシステムにおいて、ソース同期クロックは付随のデータと同じ方向に伝送される。ソース同期クロックとその付随のデータとは一緒に受取られる。「ソース同期クロック」はまた「フォワードクロック」とも呼ばれることが注目される。例示的な実施例ではソース同期クロックが示されているが、コンピュータシステムに適切ないかなるクロックメカニズムをも用いてもよいということもまた注目される。データバス126Aおよびアドレスバス128Aもまた単なる例示的なものにすぎない。図2で示されるコンピュータシステムのオペレーションに関する種々の詳細が、以下の図3から図5の説明に関して、提供される。

20

【0026】

図3を参照すると、図1および図2で示されるコンピュータシステム等のコンピュータシステム内におけるロックオペレーションを行なうための全体のオペレーションを示すフローチャートの1つの実施例が例示される。図3に示されるオペレーションはフローチャートの形式で示されているが、図3の種々のオペレーションは異なる順序で起こってもよく、または全く起こらなくてもよいということが注目される。

【0027】

未処理のロックがない場合には、1つまたは2つ以上のプロセッサ110のうちの第1のプロセッサ110Aが、ロックトグルコマンドをブリッジ130へと発行しロックを開始する。プロセッサ110Aはまた、リソースカウンタ118およびコミットカウンタ119を増分させ(ステップ410)、通例、他のコマンドを開始することを避けるが、いくつかの実施例では投機的なコマンドの開始を選ぶことがあってもよい。ブリッジ130は、適切なプロセッサ待ち行列134内のロックトグルコマンドをバッファリングする(ステップ415)。ブリッジは、ロックトグルコマンドが到達するまでは、1つまたは2つ以上のプロセッサ待ち行列134からの要求されたオペレーションを行なう(ステップ420)。他の大部分のオペレーションと同様に、ブリッジ130は、ロックトグルコマンドがプロセッサ待ち行列134Aから取除かれたとき、肯定応答信号(たとえばAビット)を適切なプロセッサ110Aに送る(ステップ425)。プロセッサ110AはAビットを受取り、リソースカウンタ118を減分する(ステップ430)。

30

【0028】

ブリッジ130は、たとえば以下の図5で説明されるようにロックトグルコマンドを処理する(ステップ435)。ブリッジ130は、ロックが確立されると、コミット信号(たとえばCビット)をプロセッサ110Aに送る(ステップ440)。プロセッサ110AはCビットを受取り、コミットカウンタ119を減分する(ステップ445)。コミットカウンタ119がゼロに一旦到達すると、プロセッサ110Aは、ロックが確立されたということを知らされる。その後、プロセッサ110Aは、ロックがアクティブな間に行なわれるべき1つまたは2つ以上のコマンドをブリッジ130に送り、第2のロックトグルコマンドがこれに続く(ステップ450)。ブリッジ130は、ロックがアクティブなままで、1つまたは2つ以上のコマンドを処理する(ステップ455)。次に、ブリッジ130は第2のロックトグルコマンドを処理し、ロックを取消す(ステップ460)。第2

40

50

のロックトグルコマンドは、第1のロックトグルコマンドと同様の態様で処理される。

【0029】

図4を参照すると、ロックオペレーションを行なうためにブリッジ130と交信する、図2のプロセッサ110Aおよび110Bのうちの第1のプロセッサ110Aのオペレーションを示すフローチャートの1つの実施例が詳細に示される。図4で示されるオペレーションは単なる例示的なものにすぎず、例示されるオペレーションフローの一部ではないプロセッサ110またはブリッジ130の追加の特徴またはオペレーションを含まない。図4に示されるオペレーションはフローチャートの形式で示されているが、図4の種々のオペレーションは異なる順序で起こってもよく、または全く起こらなくてもよいということが注目される。

10

【0030】

復号ユニット112は、ロックを特定する1つまたは2つ以上の命令を復号化し、それを識別する(ステップ310)。次に、復号ユニット112は、ロード/記憶ユニット114に、ロックを特定する1つまたは2つ以上のメモリオペレーションを知らせる(ステップ315)。より具体的には、復号ユニット112は、1つまたは2つ以上のメモリオペレーションをロード/記憶ユニット114に伝送し、ロック信号をアサートする。ロード/記憶ユニット114は、メモリオペレーションおよび対応するロックの要求をバッファリングする(ステップ320)。ロックの要求が最も古い未処理のオペレーションである場合、ロード/記憶ユニット114はシステムインターフェイスコントローラ116にロックトグルを要求する(ステップ325)。

20

【0031】

システムインターフェイスコントローラ116は、ロックのために、第1のロックトグルコマンドをブリッジ130に発行する(ステップ330)。システムインターフェイスコントローラ116は、ブリッジ130が第1のロックトグルコマンドを実行するまで、たとえば、コミットカウンタ119がゼロにまで減分するまで、待つ(ステップ335)。ブリッジ130が第1のロックコマンドを完遂した後に、システムインターフェイスコントローラ116は、ロックトグルグラント信号をロード/記憶ユニット114に返す(ステップ340)。

【0032】

ロックトグルグラント信号をシステムインターフェイスコントローラ116から受取ったことに応答して、ロード/記憶ユニット114は、ロックを特定する1つまたは2つ以上のオペレーションを行なう(ステップ345)。ロード/記憶ユニット114は、ロックを解除するための第2のロック要求をシステムインターフェイスコントローラ116に送る(ステップ350)。システムインターフェイスコントローラ116は、アンロックのために、第2のロックトグルコマンドをブリッジ130に発行する(ステップ355)。システムインターフェイスコントローラ116は、(アンロックのための)第2のロックトグル要求をロード/記憶ユニット114から受取ったことに応答して、アンロックのための第2のロックトグルコマンドをブリッジ130に発行してもよい。システムインターフェイスコントローラ116は、ブリッジ130がアンロックのためのロックトグルコマンドを実行するまで待ち(ステップ360)、それからロックトグルグラント信号を返す。

30

【0033】

図5は、制御論理136が種々のプロセッサ待ち行列134からコマンドを削除するときの、ブリッジ130のオペレーションのハイレベルフローチャートの実施例を例示する。図5に示されるオペレーションはフローチャートの形式で示されるが、図5の種々のオペレーションは異なる順序で起こってもよく、または全く起こらなくてもよいということが注目される。

【0034】

制御論理136はまず、未処理のロックが存在するかどうかを検査する(判断ブロック510)。未処理のロックは、ロックレジスタ132内のロック状態によって、またはロッ

40

50

ク状態を表わす他の所望の手段によって、表示されてもよい。より具体的には、ロックレジスタ132内のロック状態は、ノーロック、プロセッサ110Aのためのロック、またはプロセッサ110Bのためのロックを表示してもよい。所望であれば、ロック状態の他の符号化で、いかなる数のプロセッサ110のためのロックを識別してもよい。未処理のロックが存在する場合には、制御論理136は、ロックを発行したプロセッサ待ち行列からのみコマンドを選択する（ステップ515）。制御論理136は、コマンドを調べそれがロックトグルコマンドであるかどうかを確かめる（判断ブロック520）。コマンドがロックトグルコマンドである場合には、制御論理136は未処理のロックをリセットする（ステップ525）。制御論理136は、既存のロックがありさらにロックトグルコマンドが実行される場合には、ロック状態をノーロックに設定する。そのとき制御論理136は動作可能になり、ブリッジ130の通常のオペレーションを続ける。10

【0035】

判断ブロック520で、コマンドがロックトグルコマンドでない場合には、ブリッジ130は、ロックを続けたままで、選択されたコマンドを処理する（ステップ530）。

【0036】

判断ブロック510で、未処理のロックが存在しない場合には、制御論理136は、利用可能なプロセッサ待ち行列134から適切なコマンドを選択する（ステップ535）。どのコマンドを、どの待ち行列134から、ということを決定するためのプロトコルは、所望の、いかなる好適なプロトコルであってもよい。制御論理136は、判断ブロック540で、コマンドがロックトグルコマンドであるかどうかを確かめるために検査する。コマンドがロックトグルコマンドである場合には、制御論理はロックを設定し（ステップ545）、たとえばロックレジスタ132のロック状態を設定して、対応するプロセッサ110のためのロックを表示する。20

【0037】

判断ブロック540のコマンドがロックトグルコマンドでない場合には、制御論理136は選択されたコマンドを処理するのみである（ステップ530）。

【0038】

図6を参照すると、図1のコンピュータシステム内の、ここではSysAddOutコマンドフォーマットと呼ばれるプロセッサによって開始されたコマンドのためのフォーマットの実施例のブロック図が例示される。好ましい実施例では、プロセッサ110が、読み出、書き込み、データ移動を伴わないプロープ応答、キャッシュロック状態遷移ブロードキャスト、およびロックトグルコマンドのためのコマンドをシステム130に発行するときに、SysAddOutコマンドフォーマットが用いられる。SysAddOutコマンドフォーマットは好ましくは、プロセッサアドレスバス128のアドレスアウト線の線[14:2]上を送られる。30

【0039】

図示されるように、SysAddOutコマンドフォーマットは、転送されたクロック信号の立上り端縁と立下り端縁との両方の上でクロックされる実施例で2つの完全な転送されたクロックサイクルにわたる4ビットタイムの広がりを含む。図6ではADDRESSで示される物理アドレスの種々のビットが、4ビットタイムサイクルにわたって分配される。他のコマンドフィールドは、M1ビット、COMMAND[4:0]フィールド、M2ビット、MASK[7:0]フィールド、CHフィールド、ID[2:0]フィールド、およびRVビットを含む。40

【0040】

図示されるように、ビットタイム0は、M1ビット、COMMAND[4:0]フィールド、およびADDRESSの一部を含む。M1ビットは、初期プロープミスインディケータである。M1が[1]に設定されている場合には、このプロセッサ110が受取る最も古いプロープはミスという結果になる。プロープがビットタイム0という結果になる場合には、M1がアサートされる。COMMAND[4:0]フィールドは、プロセッサ110からシステム130へのコマンド符号化を表示するのに用いられる。COMMAND[50

4 : 0] フィールドの種々の符号化は、読出、書込、プローブ応答等を識別する。ある特定の符号化がロックトグルコマンドを表示する。

【 0 0 4 1 】

図示されるように、ビットタイム 1 は ADDRESS の別の部分を含む。

図示されるように、ビットタイム 2 は、M2 ビット、MASK [7 : 0] フィールド、CH フィールド、および ID [2 : 0] フィールドを含む。M2 ビットは、レート(late)プローブミスインディケータまたはキャッシュヒット確認である。M2 が [1] に設定されている場合、このプロセッサ 110 が受取る最も古いプローブはミスという結果になる。ビットタイム 0 の後、かつビットタイム 2 の前に、プローブが決定される場合には、M2 がアサートされる。M2 はまた、プローブヒットを表示するがデータ移動を表示しない CH ビットを確認してもよい。MASK [7 : 0] フィールドは、データ転送マップである。¹⁰ MASK [7 : 0] フィールドは、すべてのサブキャッシュロックコマンド(バイト、LW、QW)のために用いられ、それは、データバス 128 の、どのバイト(たとえば 8 ビット)、ロングワード(たとえば 16 ビット)、またはクワッドワード(たとえば 32 ビット)が、要求されるデータ転送のために有効であるのかを表示する。CH ビットはキャッシュヒットビットである。最も古い未処理のプローブが、システム 130 へのデータ移動が必要とされないプロセッサキャッシュヒットという結果になった場合には、キャッシュヒットビットは [たとえば 1] に設定される。キャッシュヒットビットが設定されると、M2 ビットが設定される。ID [2 : 0] フィールドは、バッファ識別フィールドである。ID [2 : 0] フィールドは、COMMAND [4 : 0] フィールド内のコマンドに対応するミスアドレスバッファ(MAB)、ビクテムデータバッファ(VDB)、または書込データバッファ(WDB)エントリを指定する。²⁰ ID [2 : 0] フィールドは、コマンドに依存して、WDB または VDB を暗にマップする。

【 0 0 4 2 】

図示されるように、ビットタイム 3 は、RV ビット、および ADDRESS の別の部分を含む。RV ビットは、COMMAND [4 : 0] フィールド内の投機的なコマンドを確認する書込有効ビットである。

【 0 0 4 3 】

図 7 を参照すると、図 1 のコンピュータシステム内、ここでは SysAddIn コマンドフォーマットと呼ばれる、メモリコヒーレンシイを保持しさらにデータを移動させるような、システムによって開始されたコマンドのためのフォーマットの実施例のブロック図が例示される。好ましい実施例では、プリッジ 130 が、プロセッサ 110 にキャッシュを探索させ、またはプロセッサ 110 への、およびプロセッサ 110 からのデータ移動を開始させるコマンドをプロセッサ 110 に発行するととき、SysAddIn コマンドフォーマットが用いられる。³⁰ SysAddIn コマンドフォーマットは好ましくは、プロセッサアドレスバス 128 のアドレスイン線 [14 : 2] 上を送られる。

【 0 0 4 4 】

図示されるように、SysAddIn コマンドフォーマットは、転送されたクロック信号の立上り端縁と立下り端縁との両方の上でクロックされる実施例で 2 つの完全な転送されたクロックサイクルにわたる 4 ビットタイムの広がりを含む。図 7 で ADDRESS で示される物理アドレスが、4 ビットタイムサイクルにわたって分配される。他のコマンドフィールドは、プローブタイプ PROBE [4 : 0] フィールド、システムデータコマンド SYSDC [4 : 0] フィールド、解除ビクティムバッファ RVB ビット、解除プローブバッファ RPB ビット、アクノリッジ A ビット、バッファ識別番号 ID [3 : 0] フィールド、およびコミット C ビットを含む。⁴⁰

【 0 0 4 5 】

プローブタイプ PROBE [4 : 0] フィールドは、プロセッサ 110 がデータをプリッジ 130 に戻すべき条件を表示し、プローブがキャッシュロックヒットの結果になる場合には、プロセッサがキャッシュロックをそれに変えなければならないキャッシュ状態を表示する。システムデータコマンド SYSDC [4 : 0] フィールドは、プロセッサへ⁵⁰

の、およびプロセッサからのデータ移動を制御する。プローブタイプ P R O B E [4 : 0] フィールドとシステムデータコマンド S Y S D C [4 : 0] フィールドとの種々の符号化が考えられる。R V B ビットは、アサートされると、I D [3 : 0] フィールドに対応するV D B またはW D B エントリを解除するようプロセッサ 110 に信号を送る。R P B ビットは、アサートされると、I D [2 : 0] に対応する、V D B 内のプローブデータエントリを解除するようプロセッサに信号を送る。

【 0 0 4 6 】

アクノリッジビットは、プロセッサによって発行されるコマンドに肯定応答する。そこでプロセッサはリソースカウンタを減分する。バッファ識別番号フィールドは、書込およびバッファ解除コマンドのためのR V B およびR P B ビットに関連のバッファ I D を識別する。I D [3] は設定されるとW D B エントリを表示し、一方でクリアされたI D [3] はV D B エントリを表示する。コミットビットはシステム 130 によってアサートされ、プロセッサが生成したコマンドがいつコヒーレンシィポイントに到達したのかを表示する。
。

10

【 0 0 4 7 】

好みの実施例では、プロセッサバス 126 A および 128 A は、マサチューセッツ州、マイナード (Mass. Maynard) のデジタルイクイップメント社 (Digital Equipment Corp.) からのE V 6 バスのバージョンと互換性がある。E V 6 バスは、これもデジタルイクイップメント社から入手可能なA L P H A プロセッサのために設計された。ロックトグルコマンドは好みの場合は、プロセッサがプロセッサ 110 である場合には、M B コードポイントを用いてロックまたはアンロックを要求する。なぜならば、プロセッサ 110 A および 110 B はメモリバリアトランザクションを行なわないからである。このようにして、ロックトグルコマンドは、プロセッサ 110 がE V 6 バスを用いることを可能にしながら依然としてA L P H A プロセッサとの互換性を維持することを可能にする。

20

【 0 0 4 8 】

上述の開示が完全に理解されると、多くの変更および変形が当業者にとって明らかになるであろう。以下の請求項がこのようなすべての変更および変形を含むと解釈されることが意図される。

【 0 0 4 9 】

【産業上の適用性】

30

この発明は一般に、プロセッサおよびコンピュータシステムに適用可能である。

【図面の簡単な説明】

【図 1】 プロセッサをブリッジに連結する別々のバスを備える 2 つのプロセッサを含むコンピュータシステムの実施例のブロック図である。

【図 2】 1 つまたは 2 つ以上のロックトランザクションを処理するように構成される、図 1 のプロセッサおよびブリッジの実施例のブロック図である。

【図 3】 図 2 のプロセッサとブリッジとの間の交信の実施例を示すフローチャートである。

【図 4】 図 2 のコンピュータシステム内でロックオペレーションを行なうための方法の実施例を示すフローチャートである。

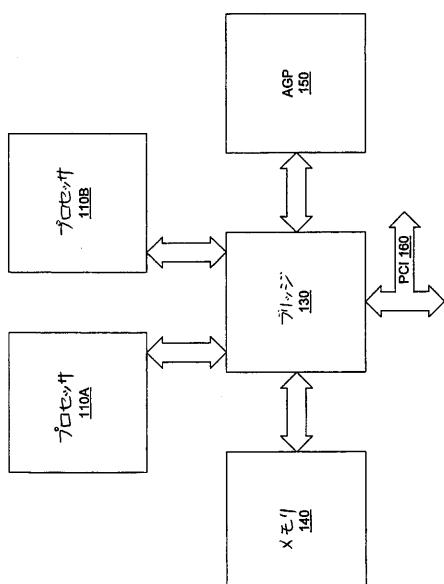
40

【図 5】 図 1 のブリッジのオペレーションの実施例を示すフローチャートである。

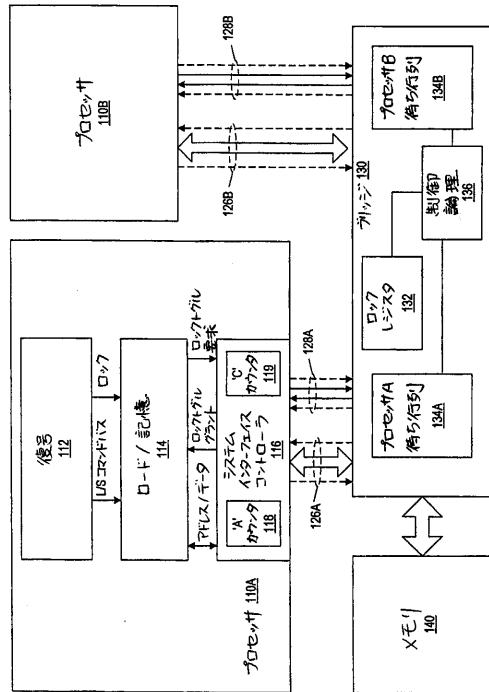
【図 6】 図 1 のコンピュータシステム内における、プロセッサによって開始されたコマンドのためのフォーマットの実施例を示す図である。

【図 7】 メモリのコヒーレンシィを維持しさらに図 1 のコンピュータシステム内のデータを移動させるために、ブリッジによって用いられるコマンドのためのフォーマットの実施例を示す図である。

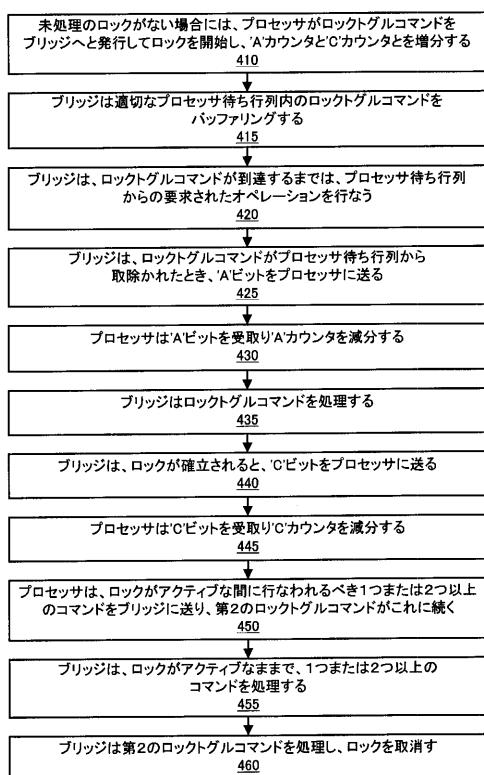
【図1】



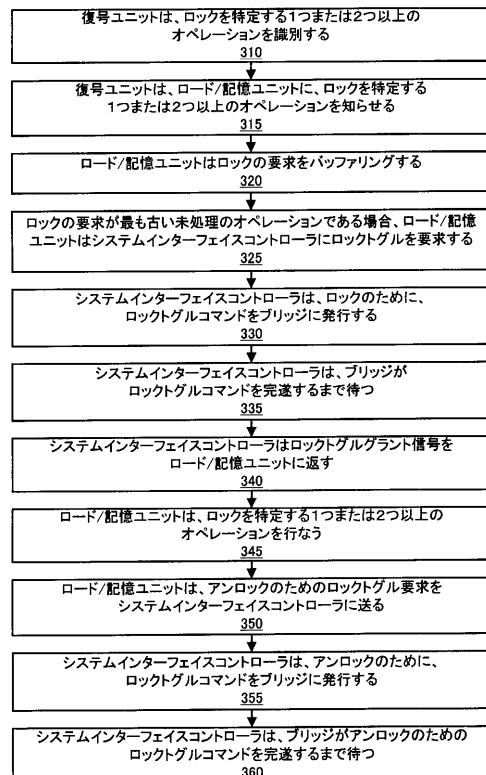
【図2】



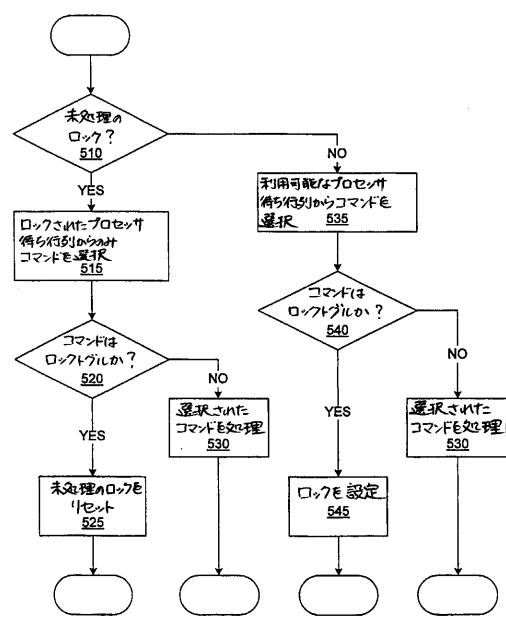
【図3】



【図4】



【 図 5 】



【 四 6 】

| | | | | |
|----|----|--------------|----------------|---------|
| | | | SADDOUT[14:2#] | |
| 14 | 13 | 12 | 11 | 10 |
| 9 | 8 | 7 | 6 | 5 |
| 4 | 3 | 2 | | |
| 0 | M1 | COMMAND[4:0] | | ADDRESS |
| 1 | | | ADDRESS | |
| 2 | M2 | | MASK[7:0] | CH |
| 3 | RV | | ADDRESS | ID[2:0] |

【 図 7 】

| E_{in} | 91_C | SADIN[14:2]# | | | | | | | | E_{out} |
|----------|--------|--------------|--|--|-----|-----|---|--|---------|-----------|
| 0 | 1 | PROB[4:0] | | | | | | | ADDRESS | |
| 1 | | | | | | | | | ADDRESS | |
| 2 | 0 | SYSDC[4:0] | | | RVB | RPB | A | | ID[3:0] | |
| 3 | C | | | | | | | | ADDRESS | |

フロントページの続き

(74)代理人 100083703

弁理士 仲村 義平

(74)代理人 100091409

弁理士 伊藤 英彦

(74)代理人 100096781

弁理士 堀井 豊

(74)代理人 100096792

弁理士 森下 八郎

(72)発明者 マイヤー, デリック・アール

アメリカ合衆国、78733 テキサス州、オースティン、クリークス・エッジ・パークウェイ、
2728

(72)発明者 ルーチャック, ウィリアム・ケー

アメリカ合衆国、78759-5183 テキサス州、オースティン、シエラ・オーツ、108
08

審査官 木村 貴俊

(56)参考文献 特開平02-232747(JP, A)

特開平03-058164(JP, A)

国際公開第99/019801(WO, A1)

(58)調査した分野(Int.Cl., DB名)

G06F 13/20-13/378