

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第4851523号
(P4851523)

(45) 発行日 平成24年1月11日(2012.1.11)

(24) 登録日 平成23年10月28日(2011.10.28)

(51) Int.Cl.

F I

G O 6 F 13/362 (2006.01)

G O 6 F 13/362 5 1 0 A

請求項の数 15 (全 17 頁)

(21) 出願番号	特願2008-526262 (P2008-526262)	(73) 特許権者	503260918
(86) (22) 出願日	平成18年8月11日 (2006.8.11)		アップル インコーポレイテッド
(65) 公表番号	特表2009-514065 (P2009-514065A)		アメリカ合衆国 95014 カリフォル
(43) 公表日	平成21年4月2日 (2009.4.2)		ニア州 クパチーノ インフィニット ル
(86) 国際出願番号	PCT/US2006/031520		ープ 1
(87) 国際公開番号	W02007/022018	(74) 代理人	100064621
(87) 国際公開日	平成19年2月22日 (2007.2.22)		弁理士 山川 政樹
審査請求日	平成21年7月29日 (2009.7.29)	(74) 代理人	100098394
(31) 優先権主張番号	11/201,581		弁理士 山川 茂樹
(32) 優先日	平成17年8月11日 (2005.8.11)	(72) 発明者	サブ라마ニアン, シュリダール・パイ
(33) 優先権主張国	米国 (US)		アメリカ合衆国・95014・カリフォル
			ニア州・クパチーノ・パークウッド ドラ
			イブ・10194・アパートメント 6

最終頁に続く

(54) 【発明の名称】 浅いエージェント毎のキューを伴うブロックしないアドレス・スイッチ

(57) 【特許請求の範囲】

【請求項 1】

複数のエージェントと、
相互接続と、

前記複数のエージェント及び前記相互接続に結合されたスイッチと
を備え、前記スイッチは、複数の記憶域を備え、前記複数の記憶域は、前記スイッチに前
記複数のエージェントにより送信された複数の要求を格納するように構成され、前記スイ
ッチは、前記複数の記憶域内に格納されている前記複数の要求のアービトレーションを行
うように構成され、前記スイッチは、前記相互接続上で選択された要求を送信するように
構成され、前記選択された要求は、前記アービトレーションの勝者であり、

前記相互接続は、前記スイッチと前記相互接続上で要求を受け取る前記複数のエージェ
ントのそれぞれとの間で結合された1つ又は複数のクロック同期記憶装置を備え、前記1
つ又は複数のクロック同期記憶装置の個数は、前記スイッチから最も遠い受け取り側エー
ジェントまでの前記要求の飛行時間に基づいており、たとえ前記複数のエージェントの少
なくとも1つが前記スイッチからの飛行時間がより短い時間を有したとしても、同じ数の
クロック同期記憶装置が前記スイッチおよび前記複数のエージェント間に備えられ、

前記クロック同期記憶装置は、フロップ、レジスタ、ラッチのうちの1つ以上を含むこ
とを特徴とするシステム。

【請求項 2】

前記複数の記憶域は、複数のキューを含み、前記複数のキューはそれぞれ、前記複数の

10

20

エージェントのうちのそれぞれ1つのエージェントに対応し、前記それぞれのエージェントにより送信される要求を格納するように構成され、前記複数のキューはそれぞれ、前記複数の記憶域のうちの少なくとも2つを含む請求項1に記載のシステム。

【請求項3】

前記スイッチは、第2の要求を選択する前に前記複数のエージェントのうちの第1のエージェントにより送信された第1の要求を選択された要求として選択するように構成され、前記第2の要求は、前記第1の要求よりも先に前記第1のエージェントにより送信される請求項1に記載のシステム。

【請求項4】

それぞれの要求は、対応する優先度を有し、前記スイッチは、前記第1の要求が前記第2の要求よりも高い優先度である場合に、前記第2の要求を選択するよりも前に前記第1の要求を選択するように構成される請求項3に記載のシステム。

【請求項5】

前記スイッチは、一組の順序付け規則に従って前記第1及び第2の要求が順序変更可能である場合に、前記第2の要求を選択するよりも前に前記第1の要求を選択するように構成され、前記スイッチは、前記第1の要求が前記第2の要求よりも高い優先度である場合でも、前記一組の順序付け規則に従って前記第1及び第2の要求が順序変更可能でない場合に、前記第2の要求よりも前に前記第1の要求を選択しないように構成される請求項3に記載のシステム。

【請求項6】

前記第1のエージェントは、前記第1の要求が前記第2の要求により順序変更可能であるかどうかを示す前記第1の要求による指示を送信するように構成され、前記スイッチは、前記第1の要求が前記第2の要求により順序変更可能であることを前記指示が示す場合に、前記第2の要求を選択するよりも前に前記第1の要求を選択するように構成される請求項5に記載のシステム。

【請求項7】

前記スイッチは、前記複数の要求のうちのそれぞれの要求に対して前記複数のエージェントのうちの1つのターゲット・エージェントを決定するように構成され、前記複数の要求のうちの1つの要求が、前記要求の前記ターゲット・エージェントにより受け取られることができない場合に、前記スイッチは、他のターゲット・エージェントへの他の要求を前記選択された要求として選択するように構成される請求項1に記載のシステム。

【請求項8】

前記スイッチは、前記複数の要求のうちのそれぞれの要求のアドレスの一部を復号化し、前記要求に対する前記ターゲット・エージェントを決定するように構成される請求項7に記載のシステム。

【請求項9】

前記複数のエージェントのうち第1のエージェントは、前記スイッチに要求を送信するために前記相互接続とは離れた別の第1の経路を介して前記スイッチに結合され、前記第1の経路はクロックサイクルよりも短い第1の飛行時間を有し、前記第1の経路はクロック記憶装置を含まず、前記スイッチに要求を送信するために、前記複数のエージェントのうち第2のエージェントは前記相互接続とは離れた別の第2の経路を介して前記スイッチに結合され、前記第2の経路はクロックサイクルよりも長い第2の飛行時間を有し、且つ少なくとも1つのクロック記憶装置を含むことを特徴とする請求項1記載のシステム。

【請求項10】

前記複数のエージェントのうち第1のエージェントは、前記スイッチに対し複数の要求を送るように構成され、前記第1のエージェントは前記スイッチにより前記相互接続から送信された前記選択された要求を受信するために結合され、前記第1のエージェントは前記相互接続上で前記選択された要求が受信されたことに応答して、前記第1のエージェントにより送信された前記複数の要求のうちのどの要求が選択されたかを判定するように構

10

20

30

40

50

成されていることを特徴とする請求項 1 記載のシステム。

【請求項 1 1】

複数の記憶域の複数のエージェントから要求をキューするステップと、

複数の要求の選択された要求を選択するために前記複数の記憶域において要求のアービトレーションを行うステップと、

相互接続上で選択された要求を送信するステップと、

から構成され、前記相互接続は前記スイッチと前記相互接続上で要求を受け取る前記複数のエージェントのそれぞれの間で結合された 1 つ又は複数のクロック同期記憶装置を備え、前記 1 つ又は複数のクロック同期記憶装置の個数は、前記スイッチから最も遠い受け取り側エージェントまでの前記要求の飛行時間に基づいており、たとえ前記複数のエージェントの少なくとも 1 つが前記スイッチからの飛行時間がより短い時間を有したとしても、同じ数のクロック同期記憶装置が前記スイッチおよび前記複数のエージェント間に備えられ、

前記クロック同期記憶装置は、フロップ、レジスタ、ラッチのうちの 1 つ以上を含むことを特徴とする方法。

【請求項 1 2】

前記複数のエージェントのうち第 1 のエージェントは、前記スイッチに要求を送信するために前記相互接続とは離れた別の第 1 の経路を介して前記スイッチに結合され、前記第 1 の経路はクロックサイクルよりも短い第 1 の飛行時間を有し、前記第 1 の経路はクロック記憶装置を含まず、前記スイッチに要求を送信するために前記複数のエージェントのうち第 2 のエージェントは、前記相互接続とは別の第 2 の経路を介して前記スイッチに結合され、前記第 2 の経路はクロックサイクルよりも長い第 2 の飛行時間を有し、且つ少なくとも 1 つのクロック記憶装置を含むことを特徴とする請求項 1 1 記載の方法。

【請求項 1 3】

前記複数のエージェントのうち第 1 のエージェントが、前記スイッチに対し複数の要求を送るステップと、

前記第 1 のエージェントが前記スイッチにより前記相互接続から送信された前記選択された要求を受信するステップと、

前記第 1 のエージェントが、前記相互接続上で前記選択された要求が受信されたことに応答して、前記第 1 のエージェントによって送信された前記複数の要求のうちのどの要求が選択されたかを判定するステップと、

をさらに有することを特徴とする請求項 1 1 記載の方法。

【請求項 1 4】

前記スイッチが、第 1 の要求の前に第 1 のエージェントにより送信される第 2 の要求を選択する前に、前記選択された要求として、前記複数のエージェントのうち第 1 のエージェントにより送信された第 1 の要求を選択するステップと、

前記第 1 のエージェントが、前記第 1 の要求と前記第 2 の要求との順序変更可能か否かを示す前記第 1 の要求とともに指示を送信するステップと、
をさらに有し、

前記第 2 の要求を選択する前前記スイッチが前記第 1 の要求を選択するステップは、前記第 2 の要求と前記第 1 の要求が順序変更可能であることが指示されたことに応答するものであることを特徴とする請求項 1 1 記載の方法。

【請求項 1 5】

前記スイッチが、第 2 の要求を選択する前に前記選択された要求として前記複数のエージェントのうち第 1 のエージェントにより送信された第 1 の要求を選択するステップであって、前記第 2 の要求は前記第 1 の要求の前に前記第 1 のエージェントによって送信され、それぞれの要求は対応する優先権を有するステップと、

前記第 1 の要求が前記第 2 の要求よりも優先されることに応答して、前記スイッチが前記第 2 の要求を選択する前に前記第 1 の要求を選択するステップとをさらに有することを特徴とする請求項 1 1 記載の方法。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、集積回路の分野に関するものであり、より具体的には、集積回路内及び／又は集積回路間の相互接続のためのアービトレーション・メカニズムに関するものである。

【背景技術】

【0002】

システム内の集積回路、又は集積回路内のさまざまな回路は、典型的には、互いに通信し合う必要がある。多くの場合、システム／集積回路内のコミュニケータは、メモリ・マップ内のさまざまなアドレスを通じて通信することができる。つまり、さまざまなコミュニケータは、メモリ・マップ内にアドレスを割り当てられ、それらのアドレスへのリード／ライト操作が通信のために使用される。典型的には、そのようなコミュニケータは、コミュニケータ間の相互接続を介して送信されるリード／ライト・トランザクションを使用する。例えば、トランザクションを開始するためアドレス、コマンド、その他のトランザクション情報を送信するアドレス・バスを備えるのが普通である。さらに、データ・バスが、もしあればトランザクションに対応するデータを送信するために使用される。それらのトランザクションに対してキャッシュ・コヒーレンシが実行される場合、コミュニケータに実装されたコヒーレンシ・スキームに従ってコヒーレンシ状態を維持するために、応答インターフェイスが備えられる。

【0003】

相互接続、又はその一部がコミュニケータ間で共有される限りにおいて、相互接続を使用するためにコミュニケータのアービトレーションを行う何らかのメカニズムが必要である。以前には、集中型アービトレーション・メカニズムと分散型アービトレーション・メカニズムが使用されていた。集中型アービトレーション・メカニズムでは、すべてのコミュニケータは、要求信号を中央アービタに送信し、そこで、どのコミュニケータに相互接続の使用を許可するかを決定する（「アービトレーション勝者」）。中央アービタは、許可されたコミュニケータに許可信号を返し、許可されたコミュニケータは、相互接続上でそのトランザクションを送る。分散型アービトレーション・スキームでは、それぞれのコミュニケータは、ローカル・アービタを実装する（又はローカル・アービタが近くに含まれる）。それぞれのコミュニケータは、その要求信号をすべてのローカル・アービタにアサートする。ローカル・アービタは、同じアービトレーション勝者を独立して決定するように設計されている。許可されたコミュニケータのローカル・アービタは許可されたコミュニケータに通知し、コミュニケータはそのトランザクションを相互接続へ送る。

【発明の開示】

【発明が解決しようとする課題】

【0004】

集中型アービトレーション・メカニズムは、典型的には、分散型アービトレーション・メカニズムに比べて実装が単純である。しかし、集中型アービトレーション・メカニズムは、典型的には待ち時間の長いメカニズムでもある。集中型アービトレーション・メカニズムは、要求信号の潜在的に長い距離の伝送と、その後続く、許可信号の同じ長い距離の伝送と、さらにその後続く、許可されたコミュニケータによるトランザクションの伝送を含む。その一方で、さらに複雑な分散型アービトレーション・スキームは、（ローカル・アービタのそれぞれへの要求信号の）1つの長距離伝送のみを伴う。分散型アービトレーション・メカニズムの複雑さに関わるものとして、典型的には、特定のコミュニケータ上の許可のより複雑な「パーキング」、コミュニケータを制御するフローの複雑さ、ソース・コミュニケータの関係に基づくターゲットとなるコミュニケータ内のバッファの割り当てがある。

【課題を解決するための手段】

【0005】

一実施態様では、スイッチは、相互接続に結合されるように構成される。スイッチは、

複数の記憶域と複数の記憶域に結合されたアービタ制御回路を備える。複数の記憶域は、複数のエージェントにより送信された複数の要求を格納するように構成される。アービタ制御回路は、複数の記憶域に格納されている複数の要求の間のアービトレーションを行うように構成される。選択された要求は、アービトレーションの勝者であり、スイッチは、選択された要求を複数の記憶域のうちの1つから相互接続上に送信するように構成される。他の実施形態では、システムは、複数のエージェントと、相互接続と、複数のエージェントと相互接続に結合されたスイッチとを備える。

【0006】

さらに他の実施態様では、方法は、複数の記憶域内の複数のエージェントからの要求をキューに入れることと、複数の記憶域内の複数の要求のアービトレーションを行い、複数の要求のうちの1つの要求を選択することと、選択された要求を相互接続上で送信することを含む。

10

【発明を実施するための最良の形態】

【0007】

以下の詳細な説明では、付属の図面を参照し、図面について簡単に説明する。

【0008】

本発明は、さまざまな修正及び代替形態により異なるが、特定の実施形態は、図面の例で示され、ここで詳細に説明される。しかし、図面やその詳細説明は、本発明を開示されている特定の形態に限定することを意図しておらず、本発明は、付属の請求項により定められているような本発明の精神と範囲から逸脱しないすべての修正形態、等価形態、代替形態を対象とする。

20

【0009】

次に図1を参照すると、システム10の一実施形態のブロック図が示されている。例示されている実施形態では、システム10は、エージェント12A~12Dなどの複数のエージェントを備える。システムは、さらに、アドレス・スイッチ14、アドレス相互接続16、応答/データ相互接続18を備える。エージェント12Aと12Bは、アドレス・スイッチ14に結合される(エージェント12Bは、例示されている実施形態ではフロップ20Aを通して結合される)。アドレス・スイッチ14は、さらに、アドレス相互接続16に結合され、これは、エージェント12A~12Dに(例示されている実施形態ではフロップ20B~20Iを通して)結合される。別の見方をすると、フロップ20B~20Iは、アドレス相互接続16の一部としてよい。エージェント12A~12Dは、さらに、応答/データ相互接続18に結合される。一実施形態では、システム10は、単一の集積回路チップ上に集積化される。他の実施形態では、システム10のさまざまなコンポーネントを別の集積回路上に実装することができる。さまざまな実施形態においてどのようなレベルの集積化も利用することができる。

30

【0010】

エージェント12A、12Bは、アドレス相互接続16で、アドレス・スイッチ14に送信すべき要求を送信するように構成される。それぞれの要求は、トランザクションのアドレスとコマンド(実行すべきトランザクションを識別する)を含む。コヒーレント・リード及びライト・コマンド、非コヒーレント・リード及びライト・コマンド、コヒーレント所有権コマンド、プローブ・コマンド、同期コマンド、キャッシュ管理コマンドなどのさまざまなコマンドをサポートできる。要求は、さらに、さまざまな実施形態において他の情報を含むことができる。例えば、以下で詳述される一実施形態では、要求は、(アービトレーションの)要求の優先度レベル、及びこの要求に対するデータがレベル2キャッシュにもコピーされるべきかどうかの指示を含むことができる。

40

【0011】

エージェント12A、12Bは、アドレス相互接続16に対する要求を送信することによりシステム10内のトランザクションを開始することができるため、ソース・エージェントと呼ばれる。例示的なソース・エージェントとしては、プロセッサ、外部ライト・バック・キャッシュ(修正された犠牲キャッシュ・ブロックをメモリに書き込むライト・ト

50

ランザクションのソースとなる)、入出力(I/O)ブリッジ(結合されている周辺機器デバイスに代わってランザクションのソースとなる)がある。図1において省略記号により示されているように、さまざまな実施形態は、2つよりも多いソース・エージェント(又は後述のソース/ターゲット・エージェント)を含むことができる。他のエージェントは、ランザクションのソースとはなりえないが、ランザクションのターゲットとはなりえる(つまり、ランザクションを受け取り、ランザクションのデータに関与するエージェント)。このようなエージェントは、ターゲット・エージェントと呼ばれる。リード・ランザクションでは、ターゲット・エージェントは、他のエージェントがデータの最近の(修正された)キャッシュコピーを持たない場合にデータを供給する。ライト・ランザクションでは、ターゲット・エージェントは、ソース・エージェントにより供給されるライト・データを吸い込む。ターゲット・エージェントは、例えば、メモリ・コントローラやI/Oブリッジを備えることができる。いくつかのエージェントは、あるランザクションについてはソース・エージェントでもあり、他のランザクションについてはターゲット・エージェントでもある。例示的なソース/ターゲット・エージェントは、上述のI/Oブリッジ又は外部キャッシュを備えることができる。一般に、エージェントは、アドレス相互接続16及び応答/データ相互接続18上でランザクションを介して通信するように構成された回路を備えることができる。

【0012】

それぞれのソース・エージェント12A、12B(又はソース/ターゲット・エージェント、ただしソース・エージェントはこの説明では略して使用される)は、ソース・エージェント12A、12Bが要求を送信していることを示す要求信号を使用することができる。アドレス・スイッチ14は、さらに、与えられたソース・エージェント12A、12Bへの許可信号をアサートし、そのソース・エージェント12A、12Bにより送信される要求を、アドレス相互接続16上を送ることが許されたことを示すこともできる。

【0013】

アドレス・スイッチ14は、ソース・エージェントにより送信された要求を、それらの要求がアドレス相互接続16上を送ることが許されるまで格納しておくように構成された複数の記憶域を備えることができる。一実施形態では、これらの記憶域は、複数のキューを備えることができる。それぞれのキューは、特定のソース・エージェントに対応し、そのソース・エージェントにより送信された要求を格納することのみに使用される。つまり、キューとソース・エージェントとの間に一対一対応関係がありうる。与えられたソース・エージェントに対するキューは、与えられたソース・エージェントによりアドレス・スイッチ14に送信された複数の要求を格納することができる。それぞれのソース・エージェントは、そのソース・エージェントに対応するキュー内のキュー・エントリの個数を認識することができ、キュー・エントリの個数を超える要求を送信することはできない。

【0014】

アドレス・スイッチ14は、さらに、キュー内の要求のアービトレーションを行い、アドレス相互接続16上へ送信の要求を選択するようにも構成される。アービトレーション・スキームは、どのようなものでも使用できる。例えば、いくつかの実施形態では、それぞれの要求に対し、優先度レベルを割り当てることができる。アービトレーション・スキームは、優先度の低い要求を飢餓状態にするのを回避するための飢餓予防メカニズムを備える厳密な優先度スキーム(最高優先度の要求を選択する)とすることができる。アドレス・スイッチ14は、アドレス相互接続16上に選択された要求を送ることができる。

【0015】

そのため、アドレス・スイッチ14は、アドレス相互接続16に対し集中型アービトレーションを使用することができる。しかし、それらの要求は、アドレス・スイッチ14へ送信され、アドレス・スイッチ14によりアドレス相互接続16上に送られるので、アービトレーションを獲得するように選択されたソース・エージェント(及びその許可に対する応答としてアドレス相互接続16を駆動するソース・エージェント)に許可を返すことに関連する待ち時間は、いくつかの実施形態では、軽減される。アドレス・スイッチ14

10

20

30

40

50

は、選択された要求をアドレス相互接続 16 上に送ると並行して、許可をソース・エージェントに返すことができる。さらに、いくつかの実施形態では、要求それ自体はアドレス・スイッチ 14 のキュー内に格納されているため、アドレス・スイッチ 14 内のアービトレーション回路は、それらの要求に関するさらに多くの情報を持つことができる（例えば、従来の集中型アービタ内に典型的には実装される要求 / 許可構造と比較して）。

【0016】

ソース・エージェントが、アドレス・スイッチ 14 から許可を受け取ると、ソース・エージェントは、キュー・エントリが他の要求を格納するのに利用できるという知らせを受け取る。一実施形態では、所定のソース・エージェントからの要求は、送信される順序で許可される。したがって、許可を受け取るソース・エージェントは、許可を対応する要求に関連付ける。他の実施形態では、アドレス・スイッチ 14 は、場合によっては、要求の順序を変更するように構成されてもよい（同じソース・エージェントからの以前に受け取った要求より先に、又はその前にソース・エージェントから後で受け取った要求を送信する）。このような実施形態では、ソース・エージェントは、アドレス相互接続 16 に結合され、かつどの要求が許可されたかを判定するために、送信された要求を受け取る。例えば、いくつかの実施形態において、ソース・エージェントはそれぞれの要求にソース・タグを付け、ソース・エージェントは、どの要求が許可されたかを判定するため、アドレス相互接続 16 からソース・タグを受け取る。

【0017】

アドレス相互接続 16 は、さまざまな実施形態において、通信媒体を含む。例えば、アドレス相互接続 16 は、パケット・インターフェイスを備えることができ、要求は、アドレス相互接続 16 上で 1 つ又は複数のクロック・サイクルにわたって 1 パケットとして送信される。特に、一実施形態では、アドレス・パケットは、アドレス相互接続 16 上で 1 クロック・サイクル以内に送信することができる。そのような実施形態では、アドレス・スイッチ 14 を、いくぶん、トランザクションのアドレス・フェーズのプロトコルから絶縁することができる。他の実施形態は、アドレス相互接続 16 をバスとして実装することができ、アドレスは、アドレス・フェーズにおいて転送されるコマンドや他の制御情報を示すさまざまな制御信号とともに送信される。

【0018】

要求は、アドレス相互接続 16 上でエージェント 12 A ~ 12 D にブロードキャストされる。いくつかの実施形態では、アドレス相互接続 16 上で最も遠いエージェント 12 A ~ 12 D までの飛行時間（物理的距離に関して）は、アドレス相互接続 16 に関連するクロックの 1 クロック・サイクルを超えることがある。フロップ 20 B ~ 20 I は、要求を取り込み、エージェント 12 A ~ 12 D への伝播を続けるために使用される。したがって、アドレス・スイッチ 14 と与えられたエージェント 12 A、12 B との間に備えられるフロップ 20 B ~ 20 I の個数は、最も遠いエージェントまでの飛行時間に基づく（アドレス相互接続 16 に使用されるクロック信号のクロック・サイクル数で）。例示されている実施形態では、飛行時間は、2 クロック・サイクルを超え、したがって 2 つのフロップが使用される。他の実施形態は、0 個のフロップ（飛行時間が 1 クロック・サイクルよりも小さい場合）、1 個のフロップ（飛行時間が、1 クロック・サイクルを超え、2 クロック・サイクル未満である場合）、又は 2 個よりも多いフロップ（飛行時間に依存する）を含むことができる。与えられた要求が、必ず、同じクロック・サイクルでそれぞれのエージェント 12 A ~ 12 D により論理的に受信されるようにするために、いくつかのエージェントが、アドレス・スイッチ 14 に物理的に近く、要求が短い飛行時間内に近いエージェントに物理的に到達できるとしても、等しい数のフロップ 20 B ~ 20 I をアドレス・スイッチ 14 とそれぞれのエージェント 12 A ~ 12 D との間に備えることができる。遠いエージェントまでのフロップ 20 B ~ 20 I は、アドレス・スイッチ 14 と遠いエージェントとの間の距離にそって物理的に分配される。図 1 は、図面において簡単のため、フロップ 20 B ~ 20 I の物理的分配を示そうとするものではない。

【0019】

それぞれのエージェント 12A ~ 12D は、同じクロック・サイクルでアドレス相互接続 16 上で送信された要求を論理的に受信するので、アドレス相互接続 16 は、いくつかの実施形態において、コヒーレント・トランザクションに対する空間内のコヒーレンシ点とすることができる。つまり、アドレス相互接続 16 上で正常に送信された要求の順序が、コヒーレンシを目的とするトランザクションの順序を定めることができる。

【0020】

同様に、いくつかの実施形態において、ソース・エージェント 12A、12B からアドレス・スイッチ 14 までの要求の飛行時間は、1クロック・サイクルを超える可能性がある。いくつかの実施形態では、アドレス・スイッチ 14 を、要求の最高の帯域幅を持つと予想されるソース・エージェントに最も近いところに物理的に配置することができる（例えば、プロセッサ・エージェントは、典型的には、キャッシュ・エージェントが I/O エージェントである場合よりも高い要求帯域幅を持つことができる）。図 1 の実施形態では、ソース・エージェント 12B からの要求の飛行時間は、1クロック・サイクルを超える場合があり、したがって、フロップ 20A は、要求を取り込み、アドレス・スイッチ 14 への伝播を続けるために使用することができる。同様に、アドレス・スイッチ 14 により返された許可信号は、フロップ 20A により取り込まれ、続くクロック・サイクルで伝播させられる。

【0021】

本発明では、アドレス相互接続 16 は、コヒーレント・トランザクションに対するコヒーレンシ点であるため（また全体として要求の順序を決めることもできる）、異なるエージェントからアドレス・スイッチ 14 に送信された複数の要求の間には順序付けはない。したがって、フロップ 20A などのフロップが、1つのソース・エージェントからの飛行時間に使用される場合、要求に対する飛行時間が 1クロック・サイクル未満である他のエージェントについて、フロップが挿入される必要はない。

【0022】

上述のように、ソース・エージェントは、与えられたエージェントからのアドレス・スイッチ 14 において未解決の複数の要求のうちのどの要求が実際にアドレス相互接続 16 上に許可されて送られたかを判定するために、いくつかの実施形態では、アドレス相互接続 16 上で要求を受け取る。さらに、いくつかの実施形態では、データをキャッシュすることもできる（したがって、コヒーレント・トランザクションに参加することができる）ソース・エージェントは、さらに、コヒーレンシを目的としてアドレス相互接続 16 上で他のソース・エージェントの要求をスヌープすることもできる。エージェント 12C ~ 12D などのターゲット・エージェントは、アドレス相互接続 16 に結合され、これにより、それらがターゲットである要求を受信する。

【0023】

一実施形態では、アドレス・スイッチ 14 は、さらに、さまざまなターゲット・エージェント 12C ~ 12D へのフロー制御を管理するように構成することもできる。例えば、アドレス・スイッチ 14 を、どのターゲット・エージェントがそれぞれの要求によりアドレスされるかを判定するように構成することができる（例えば、要求アドレスの粗粒度復号化や復号化に基づくターゲット・エージェントへのアドレスのマッピングを介して）。アドレス・スイッチ 14 は、ターゲット・エージェントのキューに置くことができる要求の個数を認識することができ（アドレス相互接続 16 から要求を受け取った後）、ターゲット・エージェントの入力キューが要求でオーバーフローしないことを保証する。与えられた要求が、入力キューが満杯である与えられたターゲット・エージェントをターゲットとする場合、アドレス・スイッチ 14 は、入力キュー・エントリが与えられたターゲット・エージェントにおいて利用可能になるまで、与えられた要求がアービトレーションの勝者として選択されないようにする。アドレス・スイッチ 14 は、このような状況において他の要求をブロックしない。つまり、アドレス・スイッチ 14 は、ターゲット・エージェントが要求を受け取ることができないため以前の要求又は優先度の高い要求がアービトレーションを獲得するのに適しない場合に、他のターゲット・エージェントをターゲットと

する他の要求をそのまま選択する。いくつかの実施形態では、アドレス・スイッチ 14 は、さらに、ソース・エージェント間に公正さをもたらすか、又はターゲット・エージェントへのアクセスの最適化を試みることもできる。

【0024】

エージェント 12A ~ 12D は、さらに、アドレス相互接続 16 上で要求を介して開始されたトランザクションの応答フェーズとデータ・フェーズを伝達するために応答 / データ相互接続 18 に結合される。いくつかのトランザクションは、さまざまな実施形態において、データ・フェーズを含まない場合がある。応答フェーズは、例えば、コヒーレント・トランザクションについてキャッシング・エージェントからの応答を含むことがある。応答は、トランザクションに対応するデータの受け取り側においてどのコヒーレンシ状態が確立されなければならないかを示す情報を与える。トランザクションのデータ・フェーズは、ソース・エージェント（リードの場合）又はターゲット・エージェント（ライトの場合）へのデータの転送を伴う。応答 / データ相互接続 18 は、さまざまな実施形態において、通信媒体を含む。

10

【0025】

アドレス・スイッチ 14 内の記憶域は、上記のいくつかの実施形態ではエージェント毎のキューとして説明されているが、他の実施形態では、記憶域を他の様式で実現することができる。例えば、これらの記憶域は、ソース・エージェントが要求を格納する単一のキューとすることができる。キュー・エントリは、アドレス・スイッチ 14 によりソース・エージェントに柔軟に割り当てられ、またそれぞれのソース・エージェントから利用可能なキュー・エントリの個数を示す追加のシグナリングがあり（例えば、キュー・エントリの個数、又は少なくとも 1 つの追加のキュー・エントリがエージェントに利用可能であることなどを示す、アドレス・スイッチ 14 からそれぞれのエージェントへのシグナリング）。エージェントは、グループ化され、またキューを共有することができるか、又はキューをそれぞれの要求のターゲットに基づいて割り当てることができる。

20

【0026】

フロップ 20A ~ 20I が図 1 の実施形態において例示されているが、一般的には、どのようなクロック同期記憶装置でもデバイス 20A ~ 20I として使用することができることに留意されたい。例えば、レジスタ、ラッチなどを使用できる。クロック同期記憶装置は、クロック信号に応答する記憶装置に対する値を取り込むように構成された任意の記憶装置を含むことができる。本発明の実施形態では、フロップ 20A ~ 20I に対するクロック信号入力は、アドレス相互接続 16 に使用されるクロックである。いくつかのエージェントは、内部においてそのクロックの倍数で動作できる。本明細書で説明されている他のフロップも、任意のクロック同期記憶装置により実現できる。一般的に、それぞれのフロップ 20A ~ 20I は、その入力の幅に等しいビット幅を持つ。例えば、フロップ 20A は、アドレス・スイッチ 14 への要求 / 許可インターフェイスの幅であり、またフロップ 20B ~ 20I の幅は、アドレス相互接続 16 の幅でもよい。

30

【0027】

厳密な優先度アービトレーション・スキームは、上記の一例として使用されるが、他の実施形態は、他のアービトレーション・スキームを実装することができる。例えば、他のアービトレーション・スキームは、ラウンドロビン、優先度重み付けラウンドロビン、ラウンドロビンと優先度スキームとの組み合わせなどを含むことができる。

40

【0028】

上記の説明では、アドレス相互接続 16 に対する要求を受け取り、要求間のアービトレーションを行ってアドレス相互接続 16 上で送信されるべきアービトレーション勝者を決定するアドレス・スイッチを参照しているが、他の実施形態では、データ相互接続に対し類似のスイッチを実装する。データ・スイッチへの要求は、データ・ソースから受け取ることができる。これらの要求は、データが対応するアドレス要求を識別するタグとともに、トランザクションに対するデータを含む。要求は、さまざまな実施形態において、アドレス要求と同じ優先度である優先度、又は異なる優先度を含むことができる。データ・ス

50

イチは、キューに入れられたデータ要求間のアービトレーションを行い、データ相互接続上で推進すべきアービトレーション勝者を選択する。

【0029】

次に図2を参照すると、アドレス・スイッチ14の一実施形態のブロック図が示されている。例示されている実施形態では、アドレス・スイッチ14は、それぞれエージェント12A、12Bに対応するキュー30A、30Bなどのキューを含む。アドレス・スイッチ14は、さらに、マルチプレクサ(mux)32、出力フロップ34、アービタ制御回路36を備える。キュー30A、30Bは、それぞれのエージェント12A、12Bから要求を受け取るように結合される。さらに、アービタ制御回路36とmux32は、例示されている実施形態では、要求を受け取るように結合される。キュー30A、30Bは、さらに、アービタ制御回路36とmux32に結合される。追加のソース・エージェントを含む実施形態では、mux32とアービタ制御回路36は、さらに、要求を受取り、かつアドレス・スイッチ14内の対応するキューの出力を受け取るように結合されている。アービタ制御回路36は、エージェント12A、12Bから要求信号を受け取り、許可信号をエージェント12A、12Bに供給するように結合される。アービタ制御回路36は、さらに、mux32に選択制御権を渡すように結合されている。mux32の出力は、出力フロップ34に結合され、これは、さらに、アドレス相互接続16に結合される。アービタ制御回路36は、さらに、エージェント12C~12D(及びいくつかの実施形態では、他のターゲット・エージェント又はソース/ターゲット・エージェント)からクレジット指示を受け取るように結合される。

【0030】

例示されている実施形態では、それぞれの要求は、アドレス(Addr)、コマンド(Command)、優先度(Pri)、トランザクションID(TID)、ReorderOKビット(ROK)を含むことができる。キュー30A、30Bは、要求を格納するように構成される。つまり、それぞれのキュー・エントリは、要求を格納するのに十分な記憶域を含むことができる。例示されている実施形態では、それぞれのキュー30A、30Bは、2つのエントリを含む。他の実施形態では、それぞれのキュー30A、30Bに、さらに多くのキューを含めることができる。アドレスは、要求の影響を受けるメモリ・マップ内のアドレスであり、ターゲット・エージェントを識別する。コマンドは、開始されるトランザクションを識別する。優先度は、要求の優先度レベルを示す。一実施形態では、3つの優先度レベルが使用される。最高優先度レベルは、要求の待ち時間がクリティカルである場合に使用できる。例えば、一実施形態では、イーサネット(登録商標)・インターフェイスなどのパケット・インターフェイスに結合されるいくつかのI/Oブリッジでは、パケット・インターフェイスを駆動するブリッジ又は回路内のバッファの空き容量が少なくなりつつあるときに、最高優先度レベルを使用してディスクリプタ又はパケット・データを読み込むことができる。中間優先度レベルは、I/Oブリッジによるプロセッサ・リードとダイレクト・メモリ・アクセス(DMA)ディスクリプタ・リードに使用することができる。中間優先度は、さらに、ソース・エージェントのライト・バッファが満杯に近づくつつある場合に書き込みに使用できる。低優先度レベルは、他のすべての要求に使用できる(例えば、プロセッサ・ライト、DMAリード及びライトなどの高帯域リード及びライトなど)。他の実施形態では、使用する優先度レベルを加減することができる。reorderOKビットは、前の要求がまだエージェントのキュー30A、30B内にある場合に、同じエージェントからの前の要求よりも先に来るように要求の順序が変更されるかどうかを示すために使用できる(つまり、前の要求が、アドレス相互接続16上にまだ許可されて送られていない)。ソース・エージェントは、エージェントにより実行される一組の順序付け規則に従ってreorderOKビットを生成することができる。一例が、図4に示されており、以下でさらに詳しく説明される。

【0031】

エージェント12A、12Bは、アドレス・スイッチ14に要求を送信するときに対応する要求信号をアサートする。つまり、アサートされた要求信号は、その要求、エージェ

10

20

30

40

50

ント 12 A、12 B に対応するキュー 30 A、30 B の書き込み、アービタ制御回路 36 への要求の指示に対する有効なビットとして役立つ。アービタ制御回路 36 は、さらに、許可信号を発生することもできる（ソース・エージェント 12 A、12 B 毎に 1 つ）。アービタ制御回路 36 は、許可信号を与えられたソース・エージェント 12 A、12 B にアサートし、ソース・エージェント 12 A、12 B からの要求がすでに許可されており、アドレス相互接続 16 上に送り出されることをも示す。アサートされた許可信号は、他の要求を受け入れられるようにエージェントのキュー 30 A、30 B 内のキュー・エントリが解放されていることを、エージェント 12 A、12 B に示す。

【0032】

それぞれのエージェント 12 A、12 B は、要求を、最大でもキュー 30 A、30 B の中のキュー・エントリの個数（例示されている実施形態では 2 つ）に等しい数まで送信するように構成されている。一実施形態では、それぞれのエージェント 12 A、12 B は、キュー 30 A、30 B 内の未解決の要求の個数をエントリの個数に制限する。つまり、エージェントは、2 つの要求を送信し、次いで、キュー・エントリが解放されることをアサートされた許可信号が示すまでさらに要求を送信することを抑制する。他の実施形態では、それぞれのエージェント 12 A、12 B は、そのキュー 30 A、30 B に書き込み、もう 1 つ要求を送信し、エージェント 12 A、12 B は、前の要求が許可され、それによりアサートされた要求がキュー・エントリに書き込まれるまで送信を続ける。

【0033】

アービタ制御回路 36 は、キュー 30 A、30 B 内の要求のアービトレーションを行い、アドレス相互接続 16 上で送信すべき要求を選択する。アービタ制御回路 36 は、mux 32 に対する選択制御を行い、要求を選択し、選択された要求を出力フロップ 34 に供給する。出力フロップ 34 は、要求をアドレス相互接続 16 上に送る。出力フロップ 34 は、クロック・サイクルの始めに要求がアドレス相互接続 16 上に必ず送り出されるように設けられる。他の実施形態では、出力フロップ 34 を取り除き、mux 32 を通じて選択後に要求を送り出す。選択された要求は、さらに、キュー 30 A、30 B から削除され、アービタ制御回路 36 は、対応するソース・エージェント 12 A、12 B に対する許可信号をアサートする。

【0034】

アービタ制御回路 36 は、複数の要求のうちから選択するために、アービトレーション・スキームを実装する。例えば、上述のように、アービタ制御回路 36 は、飢餓予防がなされる厳密な優先度選択を実装する。このようなスキームでは、最高優先度要求は、通常、アービトレーションの勝者として選択される。しかし、多数の高い優先度の要求が長期間にわたってキュー 30 A、30 B 内に低い優先度の要求が留まる原因となる場合（つまり、低い優先度の要求は、「飢餓」状態になる）、低い優先度の要求を選択する。飢餓予防メカニズムは、さまざまな方法で実現する。例えば、それぞれの要求には、タイマーが関連付けられるか、又は要求がキュー 30 A、30 B 内にどれだけの期間留まっていたかを示すタイムスタンプが関連付けられる。要求がキュー 30 A、30 B に留まっている期間が閾値期間よりも長い場合（固定されるか、又はプログラム可能なものとすることができる）、要求を選択する。実際、要求の優先度は、キュー 30 A、30 B 内のその要求の経過時間により上げることができる。他の例では、高い優先度の要求の決められた数が連続して選択されると（この数は、固定であるか、又はプログラム可能なものとすることができる）、低い優先度の要求が自動的に選択される。与えられたアービトレーションに対し、複数の要求がキュー 30 A、30 B 内で最高優先度である場合、要求を選択するためのどのようなメカニズムでも使用できる（例えば、ソース・エージェント間の固定された優先度、ソース・エージェント間のラウンドロビン、最も古い要求を選択できる、など）。他の実施形態では、他のアービトレーション・スキームを実装することができる（例えば、優先度のないラウンドロビン、優先度に基づく重み付けラウンドロビンなど）。

【0035】

高優先度要求が、キュー 30 A、30 B 内の他の要求の「背後」にある場合、その高優

10

20

30

40

50

先度要求に対する `ReorderOK` ビットが、前の要求に先立つ要求の順序変更が許容されることを示すように設定されていないと、その高優先度要求は、アービトレーションに不適であると考えられる。つまり、アービタ制御回路 36 は、順序変更が許容されることを `ReorderOK` ビットが示していない場合、同じキュー内の前の低い優先度の要求よりも先に高優先度要求を選択することはできない。`ReorderOK` ビットにより示されるように、前の要求よりも先に順序変更が許容される場合、高優先度要求は、高い優先度の要求がキュー内の低い優先度の要求の背後にある場合に選択することができる。つまり、低い優先度の要求は、エージェントにより、高優先度要求を送信する前に、アドレス・スイッチに送信されている。

【0036】

いくつかの実施形態では、アービタ制御回路 36 は、さらに、それぞれのターゲット・エージェント（例えば、図 1 の実施形態におけるエージェント 12C ~ 12D）へのフロー制御を実行する。アービタ制御回路 36 は、それぞれの要求に対するターゲット・エージェントを決定する（例えば、本発明の実施形態におけるアドレスを使用する）。アービタ制御回路 36 は、ターゲット・エージェントへのアドレスのマッピングによりプログラム可能である。例えば、1 つ又は複数のレジスタ 38 は、アドレス空間をターゲット・エージェントにマッピングするようにプログラムすることができる。このアドレス・マッピングに基づき、アドレス制御回路 36 は、いくつかの最上位アドレス・ビットの粗粒度復号化を実行し、ターゲット・エージェントを決定する。本発明の実施形態において復号化が粗粒度であるのは、比較的大きな連続するアドレス範囲が同じターゲット・エージェントにプログラムされると予想されるからである。他の実施形態では、細かい粒度の復号化を使用することができる。さらに、この実施形態では、復号化は、プログラム可能であるが、他の実施形態では、固定されたアドレス・マップを備え、アービタ制御回路 36 は、その固定されたアドレス・マップに従ってアドレスを復号化する。

【0037】

それぞれのターゲット・エージェントは、特定の数までのトランザクションを受け入れる能力を持つ（例えば、ターゲット・エージェント内に実装されたバッファの個数に応じて）。いくつかの実施形態では、トランザクションは、トランザクション・タイプ別にグループ化することができ、またトランザクションのそれぞれのグループの個数は、ターゲット・エージェント毎に指定することができる。例えば、一実施形態では、トランザクションは、コヒーレント・リード、コヒーレント・ライト、非ポスト非コヒーレント・コマンド、ポスト非コヒーレント・コマンドとしてグループ化することができる。それぞれのターゲット・エージェントは、トランザクションの上記のグループのそれぞれについて特定の数 of バッファを実装する。

【0038】

アービタ制御回路 36 は、ターゲット・エージェントへのフロー制御を実行するように構成され、それによりターゲット・エージェントのバッファがオーバーフローしないようにする。例えば、それぞれのバッファが対応するトランザクション・タイプに対するクレジットにより表されるクレジットベースのシステムを使用することができる。アービタ制御回路 36 は、利用可能なクレジットを追跡する（例えば、図 2 の 1 つ又は複数のレジスタ 40 を使用して）。アービタ制御回路 36 が、与えられたタイプの、また与えられたターゲット・エージェントをターゲットとする要求を選択した場合、アービタ制御回路 36 は、対応するクレジット・カウントを 1 だけ減らすことができる。ターゲット・エージェントは、さらに、バッファが空いたときにクレジットの返却を伝達することもできる（図 2 においてクレジットとして示されている）。そのため、与えられた時点に、アービタ制御回路 36 は、それぞれのトランザクション・タイプに対するそれぞれのターゲット・エージェントにおいてバッファが利用可能であることを認識することができる。アービタ制御回路 36 は、その要求により消費されるクレジットが利用可能でない場合、要求が選択されるのを妨げることができる。他の要求（さらに低い優先度の要求も）は、その他の要求の対応するクレジットが利用できる場合に代わりに選択することができる。

【 0 0 3 9 】

いくつかの実施形態では、アービタ制御回路 3 6 は、さらに、ソース・エージェントによる与えられたターゲット・エージェントへのアクセスの公平性を確実なものとすることを試みる。アービタ制御回路 3 6 は、それぞれのターゲット・エージェントに対するさまざまなタイプのクレジットの全体的な使用とともにそれぞれのソース・エージェントによるクレジットの使用を追跡する。ターゲット・エージェントに対するクレジットの全体的使用度が高く（ターゲット・エージェントがトランザクションに関して「ビジー」状態であることを示す）、特定のソース・エージェントがそのターゲット・エージェントと高速通信している場合（そのターゲット・エージェントに対するクレジットの使用により示される）、アービタ制御回路 3 6 は、そのソース・エージェント / ターゲット・エージェントの対によるクレジットの使用を制限し、他のソース・エージェントによるターゲット・エージェントへのアクセスを改善する。

10

【 0 0 4 0 】

図 2 の実施形態に例示されているように、mux 3 2 とアービタ制御回路 3 6 は、エージェント 1 2 A、1 2 B によりアドレス・スイッチ 1 4 に現在送信されている要求を受け取るように結合される。アービタ制御回路 3 6 は、与えられた要求に対するキュー 3 0 A、3 0 B をバイパスし、要求が送信された（またターゲット・エージェントの対応するクレジットが消費に使用可能である）ときにキュー 3 0 A、3 0 B が空である場合に mux 3 2 を通じて要求を選択するように構成される。キューを通る待ち時間は、このような場合に回避できる。他の実施形態では、バイパスを 1 つのソース・エージェント、又はソース・エージェントの部分集合に対してのみ用意することができる（例えば、プロセッサは、バイパスを持つことができるが、他のエージェントは持つことができない）。さらに他の実施形態では、バイパスを実装せず、mux 3 2 とアービタ制御回路 3 6 への受信要求の接続をなくすることができる。

20

【 0 0 4 1 】

本発明の実施形態では Reorder OK ビットを使用して、与えられた要求が同じエージェントからのすでに送信されている要求よりも前に順序変更できるかどうかを示すが、他の実施形態では他の指示を使用することがあることに留意されたい。例えば、2 つよりも多いキュー・エントリがキュー 3 0 A、3 0 B 内に実装される場合、キュー内のそれぞれのエントリに対応する Reorder OK ビットがある。それぞれの Reorder OK ビットは、要求が対応するキュー・エントリ内の要求に関して順序変更可能であるかどうかを示すことができる。それとは別に、アービタ制御回路 3 6 は、与えられた要求がすでに送信されている要求の前に来るように順序変更できるかどうかを判定する場合に、エージェントに適用される一組の順序付け規則を実装する。

30

【 0 0 4 2 】

図 3 は、アービトレーション制御回路 3 6 の一実施形態のオペレーションを例示する流れ図である。これらのブロックは、理解しやすいように特定の順序で示されているが、どのような順序でも使用できる。さらに、アービトレーション制御回路 3 6 内の組み合わせ論理回路により、さまざまなブロックを並列実行することができる。他のブロック、ブロックの組み合わせ、又は流れ図は、全体として、望み通り、複数のクロック・サイクルにわたってパイプライン化することができる。

40

【 0 0 4 3 】

アービトレーション制御回路 3 6 は、飢餓制御が現在のアービトレーション・サイクルに対しアクティブ化されている場合に「通常」（例えば、優先度ベース）のアービトレーションをオーバーライドする（決定ブロック 5 0）。上述のように、飢餓制御は、与えられた低い優先度の要求が長時間キュー内に留まっている場合にアクティブ化される。それとは別に、飢餓制御は、高優先度要求が多数の連続するアービトレーションにまたがって選択される場合にアクティブ化される。飢餓制御がアクティブ化された場合（決定ブロック 5 0、「はい」の選択肢）、アービトレーション制御回路 3 6 は、時間の経った要求（又は低い優先度の要求）をアービトレーション勝者として選択する（ブロック 5 2）。

50

【 0 0 4 4 】

飢餓制御がアクティブ化されていない場合（決定ブロック 50、「いいえ」の選択肢）、アービトレーション制御回路 36 は、さまざまな要求にアービトレーションに不適であるとしてマスクし、マスクされていない要求のアービトレーションを行う。例えば、キュー内の要求が、順序変更を許可しないことを示す `ReorderOK` ビットを有し、同じキュー内に前の要求がある場合、要求をマスクし、その要求が前の要求よりも先に選択されることを防ぐ（例えば、その要求が、前の要求よりも高い優先度を持つ場合）（ブロック 54）。さらに、要求が、その要求に対する適切なタイプの利用可能なクレジットがないターゲット・エージェントをターゲットとする場合、その要求をマスクする（ブロック 56）。アービトレーション制御回路 36 が、要求に対応するソース・エージェントにより消費されるクレジットを制限しており、その制限に達した場合に、消費に使用できるクレジットがある場合でも、その要求をマスクすることもできる（ブロック 56）。アービトレーション制御回路 36 は、最高優先度の非マスク要求（ブロック 58）をアービトレーション勝者として選択する。

10

【 0 0 4 5 】

アービトレーション制御回路 36 は、アドレス相互接続 16 上で `mux` 32 を通してアービトレーション勝者を選択する。さらに、アービトレーション制御回路 36 は、選択された要求のソースとなったソース・エージェントへの許可信号をアサートし、キュー 30A、30B から選択された要求を削除する。

20

【 0 0 4 6 】

図 4 は、さまざまな実施形態において、ソース・エージェント、又はアービトレーション制御回路 36、又はその両方により実行することができる一組の順序付け規則の一実施形態のブロック図である。トランザクション・タイプは、表の最上行左から右へと表の左側の上から下に向かって示されている。行と列との交差点のところに、その行のタイプのトランザクションがその列のタイプの前のトランザクションよりも前に順序変更することが許されるかどうかに関する規則がある。

【 0 0 4 7 】

したがって、コヒーレント・リード及びライト要求は、アドレスの一致がない限り、自由に順序変更できる。この文脈では、アドレス一致は、コヒーレンシが維持される粒度に関して検出される（例えば、キャッシュ・ブロック）。コヒーレント・リード要求は、非コヒーレント・ポスト要求及び非コヒーレント完了よりも先になるように順序変更できるが、非コヒーレント非ポスト要求よりも先になるように順序変更できない（コヒーレント・リード行及び非コヒーレント・ポスト、非コヒーレント非ポスト、及び非コヒーレント完了の列）。場合によっては、要求の順序が変更可能かどうかは、要求の特定のインスタンスに依存する（a）と b）の答えを含む交差点）。a）及び b）に対する要求のタイプは、図 5 の表の下で定義されている。Y/N は、順序変更が許容されるが、許可する必要はないことを意味する。そのため、要求は、図 4 が「はい」又は「Y/N」を示す場合に順序変更可能である。

30

【 0 0 4 8 】

次に図 5 を参照すると、アドレス相互接続上で要求を伝達する方法の高水準の流れ図が示されている。ソース・エージェントからの要求は、アドレス・スイッチのキューに入れられる（ブロック 70）。アドレス・スイッチは、要求を選択するためにキューに入っている要求のアービトレーションを行う（ブロック 72）。選択された要求は、アドレス相互接続上で送信される（ブロック 74）。

40

【 0 0 4 9 】

上記開示が完全に理解されれば、当業者には、多くの変更形態及び修正形態が明白なものとなるであろう。付属の請求項は、このようなすべての変更形態及び修正形態を包含するものと解釈することが意図されている。

【 図面の簡単な説明 】

【 0 0 5 0 】

50

【図 1】集積回路の一実施形態のブロック図である。

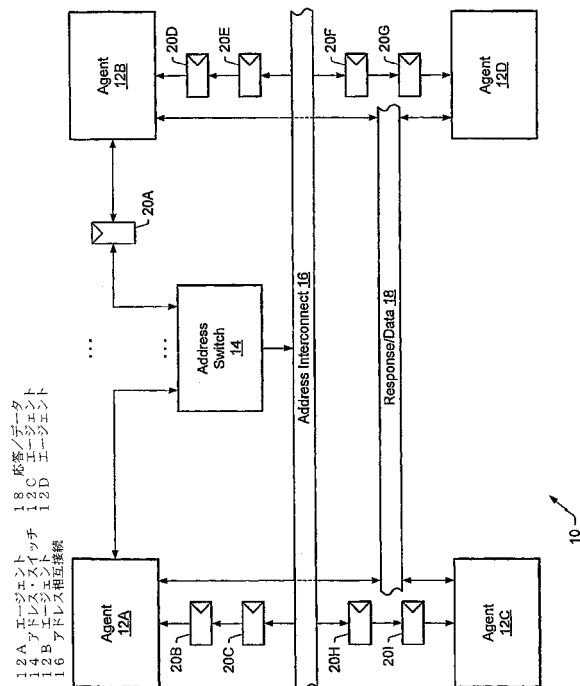
【図 2】図 1 に示されているアービタ/アドレス・スイッチの一実施形態のブロック図である。

【図 3】要求のアービトレーションを行うための図 2 に示されているアービタ制御ユニットの一実施形態のオペレーションを例示する流れ図である。

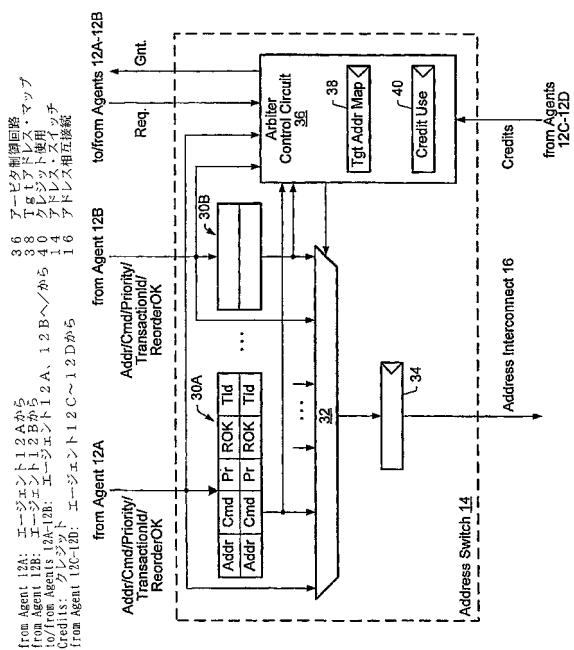
【図 4】一実施形態の順序付け規則を例示する表である。

【図 5】アドレス相互接続上で要求を伝達する方法の高水準の流れ図である。

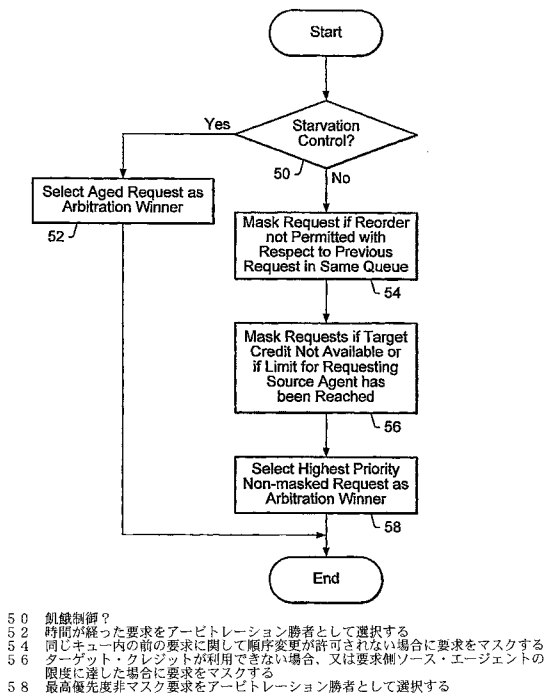
【図 1】



【図 2】



【 図 3 】

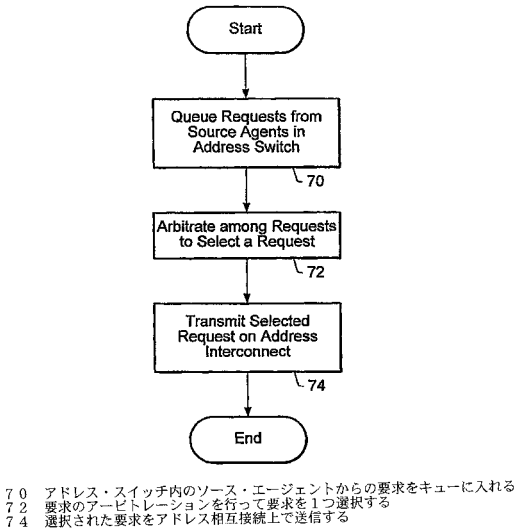


【 図 4 】

トランザクション・タイプ	コヒーレント・リード	コヒーレント・ライト	非コヒーレント・ポスト	非コヒーレント非ポスト	非コヒーレント完了
コヒーレント・リード	はい (アドレス一致がない場合)	はい (アドレス一致がない場合)	はい	いいえ	はい
コヒーレント・ライト	はい (アドレス一致がない場合)	はい (アドレス一致がない場合)	いいえ	はい	はい
非コヒーレント・ポスト	はい	いいえ	a) いいえ b) Y/N	はい	a) いいえ b) Y/N
非コヒーレント非ポスト	いいえ	いいえ	いいえ	Y/N	Y/N
非コヒーレント完了	はい	いいえ	a) いいえ b) Y/N	はい	a) Y/N

a) リード要求
b) IOライト、構成ライト、CRライト

【 図 5 】



フロントページの続き

- (72)発明者 ケラー, ジェイムズ・ビー
アメリカ合衆国・94062-3828・カリフォルニア州・レッドウッド シティ・パロマー
ドライブ・950
- (72)発明者 イウ, ジョージ・コン
アメリカ合衆国・94086・カリフォルニア州・サニベイル・ケンモア アベニュー・338
- (72)発明者 ワドハワン, ルチ
アメリカ合衆国・94086・カリフォルニア州・サニベイル・アシロマ テラス・977・ナ
ンバー 2
- (72)発明者 ゲンナ, ラメシュ
アメリカ合衆国・95129・カリフォルニア州・サンノゼ・ブルー リッジ ドライブ・469
8

審査官 坂東 博司

- (56)参考文献 特開平11-096108(JP, A)
特開平08-077104(JP, A)
特開平11-191076(JP, A)
特開2002-041449(JP, A)
米国特許出願公開第2003/0065843(US, A1)
欧州特許出願公開第01308862(EP, A1)
PCI EXPRESS BASE SPECIFICATION REVISION 1.0a, 2003年 4月15日, 1~426ペー
ジ

- (58)調査した分野(Int.Cl., DB名)
G06F 13/362