

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号
特許第5583660号
(P5583660)

(45) 発行日 平成26年9月3日(2014.9.3)

(24) 登録日 平成26年7月25日(2014.7.25)

(51) Int.Cl.

G O 6 F 12/08 (2006.01)

F I

G O 6 F 12/08 5 3 1 B

G O 6 F 12/08 5 0 7 H

G O 6 F 12/08 5 7 1

G O 6 F 12/08 5 6 1

請求項の数 4 (全 22 頁)

(21) 出願番号	特願2011-507496 (P2011-507496)	(73) 特許権者	504199127
(86) (22) 出願日	平成21年2月23日 (2009.2.23)		フリースケール セミコンダクター イン
(65) 公表番号	特表2011-519461 (P2011-519461A)		コーポレイテッド
(43) 公表日	平成23年7月7日 (2011.7.7)		アメリカ合衆国 テキサス州 7 8 7 3 5
(86) 国際出願番号	PCT/US2009/034866		オースティン ウィリアム キャノン
(87) 国際公開番号	W02009/134517		ドライブ ウェスト 6 5 0 1
(87) 国際公開日	平成21年11月5日 (2009.11.5)	(74) 代理人	100142907
審査請求日	平成24年1月30日 (2012.1.30)		弁理士 本田 淳
(31) 優先権主張番号	12/112,502	(72) 発明者	モイヤー、ウィリアム シー、
(32) 優先日	平成20年4月30日 (2008.4.30)		アメリカ合衆国 7 8 6 2 0 テキサス州
(33) 優先権主張国	米国 (US)		ドリッピング スプリングス メドウ
			リッジ ドライブ 1 1 1 1
		審査官	酒井 恭信
			最終頁に続く

(54) 【発明の名称】 データ処理システムにおけるキャッシュ・コヒーレンス・プロトコル

(57) 【特許請求の範囲】

【請求項 1】

データ処理システムであって、
キャッシュを備えた第 1 マスタと、
第 2 マスタと、
システム相互接続を介して前記第 1 マスタおよび前記第 2 マスタに操作可能に結合され
たメモリとを備え、

前記キャッシュは、キャッシュ・コントローラを備え、該キャッシュ・コントローラは
、前記キャッシュのデータ・ユニットに対する 1 組のキャッシュ・コヒーレント状態を実
行し、該キャッシュ・コヒーレント状態は、

無効状態と、

前記キャッシュのデータ・ユニットにおけるデータが修正されておらず、前記デー
タ処理システムにおける他のキャッシュにおけるデータとコヒーレントであると保証され
ないことを示す未修正非コヒーレント状態と、

前記データ・ユニットのデータが修正されておらず、前記メモリ及び前記データ処
理システムにおける他のキャッシュにおけるデータとコヒーレントであることを示す未修
正コヒーレント状態と、

前記キャッシュのデータ・ユニットにおけるデータが修正されており、前記データ
処理システムにおける他のキャッシュにおけるデータとコヒーレントであると保証されな
いことを示す修正済非コヒーレント状態と、を含み、

前記キャッシュのデータ・ユニットにおけるデータのためにライト・スルーが必要であることを示すライト・スルー要求属性に応じて、前記キャッシュ・コヒーレント状態が前記無効状態から前記未修正コヒーレント状態又は前記未修正非コヒーレント状態のいずれかに移行する、データ処理システム。

【請求項 2】

前記無効状態から前記キャッシュのデータ・ユニットの前記未修正非コヒーレント状態への移行は、前記キャッシュへの読み取りミスに応じてなされ、該読み取りミスのデータは、前記データ・ユニットに書き込まれ、キャッシュ・コヒーレンス動作は、前記読み取りミスに応じて前記データ処理システムにおける他のキャッシュ上では行なわれない請求項 1 記載のデータ処理システム。

10

【請求項 3】

書込みキャッシュ・ミスに応じて、該書込みミスのデータが書き込まれる、前記無効状態から前記キャッシュのデータ・ユニットの前記未修正非コヒーレント状態への移行は、前記データ処理システムにおける他のキャッシュにおけるキャッシュ・コヒーレンス動作を生成しない請求項 1 記載のデータ処理システム。

【請求項 4】

前記無効状態から前記キャッシュのデータ・ユニットの前記未修正非コヒーレント状態への移行は、前記データ処理システムにおける他のキャッシュが、前記データ・ユニットに書き込まれるデータに対応したメモリ・アドレスを含むか否かを判断するために、前記他のキャッシュへのルックアップ動作を生じさせない請求項 1 記載のデータ処理システム。

20

【発明の詳細な説明】

【技術分野】

【0001】

本開示は、概して、データ処理システムに関し、より詳しくは、データ処理システムにおけるキャッシュ・コヒーレンス・プロトコルに関する。

【背景技術】

【0002】

キャッシュ・コヒーレンス・プロトコルは、典型的には、多重プロセッサ・システムで使用される。そのようなプロトコルの 1 つは、MESI プロトコルである。しかしながら、MESI プロトコルを実施するために、多重プロセッサ・システムのバス・プロトコルは、特定の形式のバス・トランザクションをサポートする必要がある。さらに、MESI プロトコルは、より低レベルのシステムに対して複雑過ぎるかまたはコスト高となる場合がある。

30

【発明の概要】

【0003】

アイテム 1 . データ処理システム（例えば、10）であって、
キャッシュ（例えば、28）を備えた第 1 マスタ（例えば、14）と、
第 2 マスタ（例えば、16）と、
システム相互接続（例えば、12）を介して第 1 マスタおよび第 2 マスタに操作可能に結合されたメモリ（例えば、20）とを備え、

40

前記キャッシュは、キャッシュ・コントローラ（例えば、29）を備え、該キャッシュ・コントローラは、前記キャッシュのデータ・ユニット（例えば、キャッシュ・エントリまたはキャッシュ・ライン）に対する 1 組のキャッシュ・コヒーレント状態（例えば、I, N, C, M）を実行、

該キャッシュ・コヒーレント状態は、
無効状態（例えば、I）と、

前記キャッシュのデータ・ユニットにおけるデータが修正されておらず、前記データ処理システムのうちの少なくとも 1 つの他の記憶装置におけるデータとコヒーレントであると保証されないことを示す未修正非コヒーレント状態（例えば、N）と、

50

前記データ・ユニットのデータが修正されておらず、前記データ処理システムのうちの少なくとも1つの他の記憶装置におけるデータとコヒーレントであることを示す未修正コヒーレント状態（例えば、C）と

を含んでいるデータ処理システム。

【0004】

アイテム2． 前記無効状態から前記キャッシュのデータ・ユニットの前記未修正非コヒーレント状態への移行は、前記キャッシュへの読み取りミスに応じてなされ、該読み取りミスのデータは、前記データ・ユニットに書き込まれ、キャッシュ・コヒーレンス動作は、前記読み取りミスに応じて前記データ処理システムにおける他のキャッシュ上では行なわれないアイテム1記載のデータ処理システム。

10

【0005】

アイテム3． 書込みキャッシュ・ミスに応じて、該書込みミスのデータが書き込まれる、前記無効状態から前記キャッシュのデータ・ユニットの前記未修正非コヒーレント状態への移行は、前記データ処理システムにおける他のキャッシュにおけるキャッシュ・コヒーレンス動作を生成しないアイテム1記載のデータ処理システム。

【0006】

アイテム4． 前記無効状態から前記キャッシュのデータ・ユニットの前記未修正非コヒーレント状態への移行は、それらのキャッシュが、前記データ・ユニットに書き込まれる前記データに対応したメモリ・アドレスを含むか否かを判断するために、前記データ処理システムの他のキャッシュへのルックアップ動作を生じさせないアイテム1記載のデータ処理システム。

20

【0007】

アイテム5． 前記キャッシュのデータ・ユニットが前記未修正非コヒーレント状態であるとき、そのデータ・ユニットの方へ向かうスヌープ処理は、そのデータ・ユニットに対するコヒーレント状態の変化をもたらさないアイテム1記載のデータ処理システム。

【0008】

アイテム6． キャッシュ・ミスのデータが前記キャッシュのデータ・ユニットに書き込まれる前記キャッシュ・ミスに応じて、

前記キャッシュのデータ・ユニットの前記無効状態から前記未修正非コヒーレント状態への移行（例えば、移行110）は、ライト・スルーが前記データ・ユニットに書き込まれたデータのために必要でないことを示すライト・スルー要求属性に応じてなされ、

30

前記キャッシュのデータ・ユニットの前記無効状態から前記未修正コヒーレント状態への移行（例えば、移行128）は、ライト・スルーが前記キャッシュのデータ・ユニットに書き込まれた前記データのために必要であることを示すライト・スルー要求属性に応じてなされるアイテム1記載のデータ処理システム。

【0009】

アイテム7． 前記キャッシュ・ミスは、書込みミスであり、

前記ライト・スルーが前記書込みミスのデータのために必要であることを前記ライト・スルー要求属性が示す場合、前記書込みミスのデータは、その後のコピーバック・バス・トランザクションで前記メモリに書き込まれないアイテム6記載のデータ処理システム。

40

【0010】

アイテム8． 前記キャッシュ・コヒーレント状態は、前記キャッシュのデータ・ユニットにおけるデータが修正されており、前記データ処理システムのうちの少なくとも1つの他の記憶装置における前記データとコヒーレントであると保証されないことを示す修正済非コヒーレント状態（例えば、M）をさらに含んでいるアイテム1記載のデータ処理システム。

【0011】

アイテム9． 前記未修正非コヒーレント状態から前記キャッシュのデータ・ユニットの前記修正済非コヒーレント状態への移行（例えば、移行114）は、前記データ・ユニットのデータへの書込みヒットに応じてなされるアイテム8記載のデータ処理システム。

50

【 0 0 1 2 】

アイテム 1 0 . 前記未修正コヒーレント状態にあるとき、前記キャッシュのデータ・ユニットのデータへの書込みヒットは、前記メモリへの前記データにライト・スルーを生成し（例えば、移行 1 3 2 ）、

前記未修正非コヒーレント状態にあるとき、前記キャッシュのデータ・ユニットの前記データへの書込みヒットは、前記データのライト・スルー動作を生成しないアイテム 9 記載のデータ処理システム。

【 0 0 1 3 】

アイテム 1 1 . 前記修正済非コヒーレント状態から前記キャッシュのデータ・ユニットの前記未修正非コヒーレント状態への移行（例えば、移行 1 1 6 ）は、前記データ・ユニットに対する無効キャッシュ動作命令なしに、前記データ・ユニットに対するフラッシュ・キャッシュ動作命令に応じてなされ、ライト・スルー要求属性は、ライト・スルーが前記データ・ユニットのデータのために必要でないことを示すアイテム 8 記載のデータ処理システム。

10

【 0 0 1 4 】

アイテム 1 2 . 前記修正済非コヒーレント状態から前記キャッシュのデータ・ユニットの前記未修正コヒーレント状態への移行（例えば、移行 1 2 6 ）は、前記データ・ユニットに対するフラッシュ・キャッシュ動作命令に応じてなされ、前記データのライト・スルー要求属性は、ライト・スルーが前記データのために必要であることを示すアイテム 8 記載のデータ処理システム。

20

【 0 0 1 5 】

アイテム 1 3 . ライト・スルーが前記データのために必要でないことをデータ・ユニットにおけるデータに対するライト・スルー要求属性が示す場合、前記修正済非コヒーレント状態から前記キャッシュの前記データ・ユニットの前記無効状態への移行は、

前記データ・ユニットに対する無効キャッシュ動作命令なしの、前記データ・ユニットに対するフラッシュ・キャッシュ動作命令であって、前記データが、前記フラッシュ・キャッシュ動作命令に応じてコピーバック・バス・トランザクションで前記メモリに書き込まれるもの（例えば、移行 1 2 4 ）、または

前記データ・ユニットに対する無効キャッシュ動作命令（例えば、移行 1 2 2 ）

に応じてなされるアイテム 8 記載のデータ処理システム。

30

【 0 0 1 6 】

アイテム 1 4 . ライト・スルーが前記キャッシュのデータ・ユニットのデータのために必要であることをライト・スルー要求属性が示す場合、前記キャッシュの前記データ・ユニットの前記未修正非コヒーレント状態から前記未修正コヒーレント状態への移行（例えば、移行 1 2 0 ）は、前記データ・ユニットに対する無効キャッシュ動作命令なしに、前記データ・ユニットに対するフラッシュ・キャッシュ動作命令に応じてなされるアイテム 1 記載のデータ処理システム。

【 0 0 1 7 】

アイテム 1 5 . ライト・スルーが前記データのために必要でないことをライト・スルー要求属性が示す場合、前記キャッシュのデータ・ユニットの前記未修正コヒーレント状態から前記未修正非コヒーレント状態への移行（例えば、移行 1 3 4 ）は、前記データ・ユニットに対する無効キャッシュ動作命令なしに、前記データ・ユニットに対するフラッシュ・キャッシュ動作命令に応じてなされるアイテム 1 記載のデータ処理システム。

40

【 0 0 1 8 】

アイテム 1 6 . 前記未修正非コヒーレント状態および前記未修正コヒーレント状態の各々から前記キャッシュのデータ・ユニットの前記無効状態への移行（例えば、移行 1 1 2 および 1 3 0 ）は、前記データ・ユニットに対する無効キャッシュ動作命令に応じてなされるアイテム 1 記載のデータ処理システム。

【 0 0 1 9 】

アイテム 1 7 . データ処理システム（例えば、1 0 ）を動作させる方法であって、

50

前記データ処理システムのキャッシュ（例えば、30）への書込みミスを生成する第1マスタ（例えば、16）によってデータの書込み処理を行なうことであって、前記第1マスタがシステム相互接続（例えば、12）に操作可能に結合され、前記データ処理システムが第2マスタ（例えば、14）と前記システム相互接続に操作可能に結合されたメモリ（例えば、20）とを備えること、

前記書込みミスに応じて、前記キャッシュに前記データを書きこむこと、

前記データのライト・スルーが要求されることを前記書込み処理の前記データに対するライト・スルー要求属性（例えば、W）が示す場合、前記メモリへの前記データのライト・スルーを行ない、前記データのライト・スルーが要求されないことを前記ライト・スルー要求属性が示す場合、前記データのライト・スルーが行なわれないこと、および

10

前記データのライト・スルーが要求されることを前記ライト・スルー要求属性が示す場合、少なくとも1つの他のキャッシュが前記書込み処理に対応したメモリ・アドレスを含むか否かを判断するために、前記データ処理システムのうちの前記少なくとも1つの他のキャッシュ（例えば、28）を探索することであって、前記データのライト・スルーが要求されないことを前記ライト・スルー要求属性が示す場合、前記少なくとも1つの他のキャッシュの探索を行なわないこと

を含む方法。

【0020】

アイテム18. データ処理システム（例えば、10）であって、

キャッシュ（例えば、28）を備えたプロセッサ（例えば、14）と、

20

第2マスタ（例えば、16）と、

前記プロセッサおよび前記第2マスタにシステム相互接続（例えば、12）を介して操作可能に結合されたメモリ（例えば、20）と、

該メモリの記憶位置に関連した複数の属性を提供するメモリ属性ロジック（例えば、MMU42内、またはプロセッサ14もしくはシステム10内のどこか他の場所）とを備え、前記複数の属性のうちの1つは、ライト・スルー要求属性（例えば、W）であり、

前記キャッシュは、キャッシュ・コントローラ（例えば、29）を備え、該キャッシュ・コントローラは、前記キャッシュのデータ・ユニットに対する1組のキャッシュ・コヒーレント状態を実行する回路を備え、前記キャッシュ・コントローラは、メモリ・コヒーレンスが前記メモリ内の一または複数の記憶位置に対して必要か否かを判断するために、前記ライト・スルー要求属性を利用するデータ処理システム。

30

【0021】

アイテム19. 前記システム相互接続に操作可能に結合されたキャッシュ・コヒーレンス・マネージャ（例えば、18）をさらに備え、メモリ・コヒーレンスが書込み処理のために必要であることを前記ライト・スルー要求属性が示す場合、前記キャッシュ・コヒーレンス・マネージャは、前記書込み処理のメモリ・アドレスに対するキャッシュ・コヒーレンス・ルックアップ動作を行なうために、前記データ・システムの他のキャッシュ（例えば、30, 41）に示すアイテム18記載のデータ・システム。

【0022】

アイテム20. メモリ・コヒーレンスが前記書込み処理のために必要ではないことを前記ライト・スルー要求属性が示す場合、前記データ・システムにおけるメモリ・コヒーレンス処理は、前記書込み処理に対して行なわれないアイテム19記載のデータ・システム。

40

【図面の簡単な説明】

【0023】

【図1】本発明の一実施形態にかかるデータ処理システムを示す図。

【図2】本発明の一実施形態にかかる、図1のデータ処理システムのプロセッサの一部を示す図。

【図3】本発明の一実施形態にかかる、図2のプロセッサによって実施されるキャッシュ・コヒーレンス・プロトコルの状態図。

50

【図4】1つのバス・プロトコルにかかる、HBURSTバス信号の符号化を規定する表を示す図。

【図5】1つのバス・プロトコルにかかる、HPROTバス信号の符号化を規定する表を示す図。

【図6】フローチャートの形態で、キャッシュ・コヒーレンス動作（例えば、スヌープすること）が行なわれるか否かを判断するための様々な実施形態を示す図。

【図7】フローチャートの形態で、キャッシュ・コヒーレンス動作（例えば、スヌープすること）が行なわれるか否かを判断するための様々な実施形態を示す図。

【図8】フローチャートの形態で、キャッシュ・コヒーレンス動作（例えば、スヌープすること）が行なわれるか否かを判断するための様々な実施形態を示す図。

【図9】フローチャートの形態で、キャッシュ・コヒーレンス動作（例えば、スヌープすること）が行なわれるか否かを判断するための様々な実施形態を示す図。

【発明を実施するための形態】

【0024】

本発明は、例示として示され、添付図面によって限定されるものではない。ここでは、同様の参照符号が同様の要素を示している。図中の要素は、簡明のために示され、必ずしも当尺には描かれてはいない。

【0025】

一実施形態では、キャッシュ・コヒーレンス・プロトコルは、一貫性および非一貫性書込みに対する個別の移行（transition）および状態（state）を使用することがある。一実施形態では、このキャッシュ・コヒーレンス・プロトコルは、コヒーレント状態間の一または複数の移行を規定するために、制御値としてメモリ管理ユニット（MMU）によって提供されるライト・スルー（write-through）・ページ属性を使用する。一実施形態では、キャッシュ・コヒーレンス・プロトコルは、コヒーレント・トラフィックに対するライト・スルー動作に依存し、非コヒーレント・データに対するコピーバック処理をサポートする。さらに、このキャッシュ・コヒーレンス・プロトコルは、キャッシュ・コヒーレンスを現在サポートしない標準バス・プロトコルと共に使用されることがある。さらに、一実施形態では、バス・トランザクションの一部は、キャッシュ・コヒーレンスが必要なものに対して選択されることが可能である。つまり、一または複数の要因に基づいて、特定のバス・トランザクションは、キャッシュ・コヒーレンス動作をもたらすこともあり、また、もたらさないこともある（例えば、スヌープ要求の生成をもたらすこともあり、また、もたらさないこともある）。さらに、一実施形態では、選択的なスヌープを行なうために使用される一または複数の要因は、既存の業界標準バス・プロトコルに存在し、その結果、標準バス・プロトコルへの追加の信号伝達機構または他の変更が、必要ではない。

【0026】

ここで使用される用語「バス」は、複数の信号または導体を参照するために使用され、それは、データ、アドレス、制御または状態等の一または複数の多様な形式の情報を転送するために使用されることが可能である。ここで議論される導体は、単一の導体、複数の導体、一方向性の導体、または双方向性の導体として例示または記述されることがある。しかしながら、異なる実施形態によって、導体の実装が変更されることがある。例えば、双方向性の導体ではなく、むしろ個別の一方向性の導体を使用されることがあり、その逆もあり得る。さらに、複数の導体を、多重信号を連続的にまたは時分割多重通信の方法で転送する単一の導体で置き換えることが可能である。同様に、多重信号を搬送する単一の導体は、これらの信号の一部を搬送する種々多様な導体に分離されることが可能である。したがって、信号の転送には多くのオプションが存在する。

【0027】

用語「アサート」または「設定」および、「ネゲート」（すなわち、「デアサート」もしくは「クリア」を意味する）は、ここでは、信号のレンダリング、状態ビット、または同様の装置をそれぞれその論理上の真の状態または論理上の誤りの状態にすることを参照

10

20

30

40

50

する場合に使用される。論理上の真の状態が論理レベル 1 である場合、論理上の誤りの状態は論理レベル 0 である。また、論理上の真の状態が論理レベル 0 である場合、論理上の誤りの状態は論理レベル 1 である。

【 0 0 2 8 】

ここで記述される各信号は、正論理または負論理として設計されることがあり、負論理は、信号名の上のバーまたはその名前に続くアスタリクス (*) によって示される。負論理信号の場合、その信号は、論理上の真の状態が論理レベル 0 に対応するアクティブ・ローである。正論理信号の場合、その信号は、論理上の真の状態が論理レベル 1 に対応するアクティブ・ハイである。ここで記述される信号のいずれも、負または正の論理信号のいずれかとして設計されることができることを注記しておく。したがって、代替の実施形態では、正論理信号として記述されるそれらの信号は、負論理信号として実施されることが可能であり、負論理信号として記述されるそれらの信号は、正論理信号として実施されることが可能である。

【 0 0 2 9 】

角括弧は、ここではバスの導体または数値データのビット位置を示すために使用される。例えば、「バス 6 0 [7 : 0]」または「バス 6 0 の導体 [7 : 0]」は、バス 6 0 の 8 つの低次導体を示し、「アドレス・ビット [7 : 0]」または「ADDRESS [7 : 0]」は、アドレス値の 8 つの低次ビットを示している。数に先行するシンボル「\$」は、その数が 1 6 進法または BASE 1 6 形式で表わされていることを示している。数に先行するシンボル「%」は、その数が 2 進法または BASE 2 形式で表わされていることを示している。

【 0 0 3 0 】

図 1 は、本発明の一実施形態にかかるデータ処理システム 1 0 のブロック図を示している。システム 1 0 は、システム・オン・チップであり得る。システム 1 0 は、単一の集積回路または複数の集積回路上に実装され得る。システム 1 0 は、例えばシステム・バスである、システム相互接続 1 2 を備える。一実施形態では、システム相互接続 1 2 は、AMBA AHB または AXI プロトコル (AMBA、AXI、および AHB は、ARM 社の登録商標である) に従って動作するシステム・バスである。システム 1 0 は、プロセッサ 1 4 および 1 6 等の任意の個数のプロセッサ、システム処理アービタ 3 2、キャッシュ・コヒーレンス・マネージャ 1 8、メモリ 2 0、レベル 2 のキャッシュ 4 1、他のマスタ (複数可) 2 2、他のスレーブ (複数可) 2 4、入力 / 出力 (I / O) モジュール 2 6、および直接メモリ・アクセス (DMA) モジュール 4 0 を備え、各々は、システム相互接続 1 2 に双方向結合されている。プロセッサ 1 4 は、キャッシュ 2 8 を備え、プロセッサ 1 6 は、キャッシュ 3 0 を備えている。キャッシュ・コヒーレンス・マネージャ 1 8 は、コヒーレント・マスタ制御レジスタ 1 9 を備えている。キャッシュ・コヒーレンス・マネージャ 1 8 もまた、システム相互接続 1 2 とは別個の場合があるスヌープ制御・インタフェースを介して、L 2 キャッシュ 4 1、プロセッサ 1 4、およびプロセッサ 1 6 の各々に双方向結合されている。加えて、キャッシュ・コヒーレンス・マネージャ 1 8 は、システム 1 0 内の他のマスタに双方向結合されることも可能である。システム 1 0 の代替の実施形態は、図 1 に示されていない追加回路を備えるか、または、例示されたものよりも少ない回路を備えることも可能である。幾つかの実施形態では、キャッシュ・コヒーレンス・マネージャ 1 8 は、プロセッサ 1 4 および 1 6 内に割り振られることが可能であり、または、存在しないことも可能である。

【 0 0 3 1 】

図 2 は、図 1 のデータ処理システム 1 0 に関連するプロセッサ 1 4 のブロック図である。プロセッサ 1 4 は、パイプライン式の方法で動作を行なうように実装され、命令パイプ 2 3、実行ユニット 2 5、命令フェッチ・ユニット 2 7、制御回路 1 7、汎用レジスタ 3 1、ロード / 記憶ユニット 3 3、バス・インタフェース・ユニット (BIU) 4 8、メモリ管理ユニット (MMU) 4 2、およびキャッシュ 2 8 を備えることが可能である。一実施形態では、制御回路 1 7 は、BIU 4 8、キャッシュ 2 8、MMU 4 2、命令フェッチ

・ユニット 27、命令パイプ 23、実行ユニット 25、ロード/記憶ユニット 33の各々に双方向結合される。命令パイプ 23は、命令フェッチ・ユニット 27、実行ユニット 25、およびロード/記憶ユニット 33に双方向結合される。実行ユニット 25は、MMU 42、キャッシュ 28、および汎用レジスタ 31の各々に双方向結合される。ロード/記憶ユニットは、汎用レジスタ 31、MMU 42、およびキャッシュ 28に双方向結合される。プロセッサ 14は、BIU 48に結合されたシステム相互接続 12を介してデータ処理システム 10の他の構成要素と通信することが可能である。キャッシュ 28は、一実施形態では、MMU 42によって提供されるメモリ・ページ属性に少なくとも部分的に基づいてキャッシュ 28に対するキャッシュ・コヒーレンス・プロトコルを実施する制御回路 29を備えている。キャッシュ 28は、複数のキャッシュ・エントリ(キャッシュ・ラインとも呼ばれる)を含み、各キャッシュ・エントリは、その現在のキャッシュ・コヒーレンス・プロトコル状態に関する情報を含む、アドレス・タグ、対応したデータ、および状態情報を含んでいる。プロセッサ 14は、制御回路 29およびキャッシュ・コヒーレンス・プロトコルを実施する任意の回路を除いては、当業者に知られるように動作し、したがって、ここではさらに詳細には議論されないことを注記しておく。

10

【0032】

MMU 42は、本技術分野で知られるように、仮想アドレス(それらは、例えば、実行ユニット 25、ロード/記憶ユニット 33、および命令フェッチ・ユニット 27から受信されることが可能である)を物理アドレスに変換する。MMU 42は、変換したアドレスをキャッシュ 28に提供する。つまり、MMU 42は、物理アドレスに対応したメモリのページに対するページ属性と共に、物理アドレスを提供する。例えば、キャッシュ 28内のエントリは、メモリ 20の記憶位置に対応することが可能であり、メモリ 20は、ページと呼ばれる記憶単位を備えている。したがって、物理アドレス変換情報と共にMMU 42によって提供されるページ属性は、MMU 42の記憶回路(又はプロセッサ 14内の他の場所)にプログラムされ、提供された物理アドレスを含んだ特定のページに対応することが可能である。一実施形態では、これらのページ属性は、ライト・スルー要求属性(W)を含むとともに、次の属性のうちのいずれか1つ、すなわちキャッシュ禁止(I)、メモリ・コヒーレンスが必要(M)(全体的に共有された(SG)とも称されることがある)、保護(G)、エンディアンネス(E)、およびユーザ定義可能(例えば、U0, U1, U2, U3)のうちのいずれか1つを含んでいる。例えば、一実施形態では、MMU 42は、変換ルックアサイド・バッファ(TLB)を備え、そのエントリは、これらの属性と共にアドレス変換を記憶し、したがって、プロセッサ 14が各TLBエントリに関連したページにおける記憶アクセスを行なう方法を制御する。一実施形態では、各ページのページ属性は、WおよびMの両方を含み、WおよびM属性の各々は、互いに独立して設定されることができる。

20

30

【0033】

典型的には、キャッシュ・コヒーレンスは、マルチマスタ・システムにおけるメモリ・コヒーレンスを可能にするようにサポートされる。コヒーレンスは、キャッシュ・コヒーレンス動作の使用にも拘わらず、維持されることが可能である。一実施形態では、これらのキャッシュ・コヒーレンス動作は、プロセッサ 14等のプロセッサに提供されるスヌープ要求(つまり、スヌープ無効要求)を含んでいる。一実施形態では、これらのスヌープ要求は、キャッシュ・コヒーレンス・マネージャ 18によってプロセッサ 14および16の各々に提供される。これらのスヌープ無効要求は、プロセッサによって受信され、その結果、プロセッサは、スヌープ無効要求に対応したアドレスがキャッシュ内に記憶されているか否かを判断することができる。したがって、キャッシュ・コヒーレンス動作は、スヌープ・ルックアップを参照することが可能であり、キャッシュがスヌープ処理に関連したデータを含むか否かを判断するように、キャッシュにおけるルックアップが行なわれる。つまり、キャッシュ・コヒーレンス動作は、少なくとも1つのキャッシュが、スヌープされたバス・トランザクションのメモリ・アドレスに関連したデータを含むか否かを判断するために、データ処理システム(データ処理システム 10等)のうちの少なくとも1つ

40

50

のキャッシュを探索することを含むことが可能である。その後、キャッシュが、スヌープされたトランザクションに関連したデータ（スヌープ要求に対応したメモリ・アドレスに一致するエントリを含む）を含んでいる場合、プロセッサは、そのエントリに対する適切なコヒーレンス動作またはそのエントリの上書きを行なうことができる。一実施形態では、スヌープ無効コヒーレンス動作は、キャッシュ・コヒーレンス・プロトコルにおいて利用され、この一致するエントリが無効にされる。なぜならば、別のバス・マスタによって行なわれるスヌープされたトランザクションがそのアドレスに記憶されたデータを修正すると思われるからである。これに代えて、キャッシュ・コヒーレンス動作がキャッシュ・エントリの状態の変化を参照することができることを注記しておく。

【 0 0 3 4 】

図 3 は、修正済非コヒーレント状態（M）、無効状態（I）、未修正非コヒーレント状態（N）、および未修正コヒーレント状態（C）の 4 つの状態を有したキャッシュ・コヒーレンス・プロトコル 1 0 0 を示している。この 4 つの状態のプロトコルでは、個別の移行および状態がコヒーレント書込みおよび非コヒーレント書込みに使用される。一実施形態では、キャッシュ・コヒーレンス・プロトコル 1 0 0 は、コヒーレント状態間の一または複数の移行を規定するための制御値として MMU 4 2 によって提供される W ページ属性を使用する。一実施形態では、キャッシュ・コヒーレンス・プロトコルは、コヒーレント・トラフィックに対するライト・スルー動作に依存し、非コヒーレント・データに対するコピーバック処理をサポートする。例えば、以下でさらに記述されるように、コヒーレント書込みは、I および C の状態を利用し、ライト・スルー書込みとして行なわれる。非コヒーレント書込みは、I、N および M 状態を利用し、コピーバック書込みとして行なわれる。

【 0 0 3 5 】

図 3 の状態図は、プロセッサ 1 4 に関して記述される。つまり、一実施形態では、キャッシュ 2 8 内の制御回路 2 9 は、図 3 の状態図を実行する。キャッシュ 2 8 における各キャッシュ・エントリは、キャッシュ・エントリの状態（M、I、N または C）を示すために一または複数の状態指標を含んでいる。（キャッシュ・エントリは、キャッシュ・ラインと称されることもあることを注記しておく。）図 3 に示されるように、状態 1 0 2 は I 状態に対応し、状態 1 0 4 は N 状態に対応し、状態 1 0 8 は M 状態に対応し、状態 1 0 6 は C 状態に対応している。キャッシュ・コヒーレンス・プロトコル 1 0 0 に関して使用されるように、キャッシュ・エントリが「未修正」の場合、プロセッサ 1 4 のキャッシュ 2 8 におけるキャッシュ・エントリのデータは、それが割り付けられているので、メモリ 2 0 におけるコピーに関して修正されていない。キャッシュ・エントリが「修正」される場合、それが割り付けられているので、プロセッサ 1 4 のキャッシュ 2 8 におけるキャッシュ・エントリのデータは、メモリ 2 0 におけるコピーに関して修正されている。キャッシュ・エントリが「コヒーレント」である場合、キャッシュ 2 8 におけるキャッシュ・エントリのデータは、メモリ 2 0、および同一のアドレス位置に対応したシステム 1 0 における任意のキャッシュ（キャッシュ 3 0 等）の任意の有効なエントリにおけるものと同一である。「非コヒーレント」は、他のキャッシュに対して非コヒーレントであることを参照している。つまり、キャッシュ・エントリが「非コヒーレント」である場合、キャッシュ 2 8 におけるキャッシュ・エントリのデータは、メモリ 2 0 の同一のアドレス位置に対応した、キャッシュ 3 0 等の別のキャッシュの有効なエントリとは異なる値を有することが可能である。「非コヒーレント」が他のキャッシュに対して非コヒーレントであることを参照しているため、キャッシュ・エントリが「非コヒーレント」である場合、キャッシュ・エントリのデータは、メモリ 2 0 におけるものと同一である場合があることを注記しておく。非コヒーレンスは、キャッシュ・コヒーレンス動作がメモリにおける特定のアドレス位置に対して行なわれていない場合に生じることがある。

【 0 0 3 6 】

コヒーレント・キャッシュ・エントリは、C 状態または I 状態のいずれかである。コヒーレンスを要求しないキャッシュ・エントリは、I、N または M 状態を想定することが可

10

20

30

40

50

能である。修正済または共有の状態（MESIプロトコルにおいて使用されるような）がコヒーレント・キャッシュ・エントリに対してサポートされず、したがって、スヌープ・コピーバックまたは介入操作が要求されないことを注記しておく。非コヒーレント・キャッシュ・エントリは、依然として修正済の状態を想定していることがある。一実施形態では、コヒーレンス動作を要求するメモリ領域（メモリ20内等）は、「メモリ・コヒーレンスが必要」（そのMビットを1に設定する）および「ライト・スルーが必要」（そのWビットを1に設定する）としてマーク付けされる。プロトコル100は、バス・プロトコルと共に使用されることが可能であり、それは、例えばAMBA AHBまたはAXIプロトコルなどのキャッシュ・コヒーレンス動作を包含またはサポートしないことを注記しておく。キャッシュ・コヒーレンスをサポートしないバス・プロトコルでは、MESIプロトコルを実施することができない。例えば、AMBA AHBプロトコルに存在しない「アドレス・オンリー処理」を利用するMESIプロトコルのS状態からM状態への移行を示す方法がないことがある。さらに、MESIキャッシュ・コヒーレンス処理をサポートしないそのようなバス・プロトコルは、MESIプロトコルにおけるI状態からE状態への移行を実行するために利用される「修正の意図を持った読出し」処理をサポートしていない。そのようなプロトコルに必要な他の種類の処理もまた、AMBA AHBまたはAXIバス・プロトコルによっては規定されない。したがって、一実施形態では、キャッシュ・コヒーレンス・プロトコル100が、キャッシュ・コヒーレンスを現在サポートしていない既存のバス・プロトコルに重畳されることが可能であることを注記しておく。

【0037】

図3を参照して、キャッシュ・エントリが無効状態であり、キャッシュ・ミスが生じて無効のキャッシュ・エントリへの割付けをもたらした場合、Wページ属性は、状態N（非コヒーレント状態）または状態C（コヒーレント状態）のどちらへ移行するかを判断するために使用される。キャッシュ・ミスが生じ、且つ、W=0である場合（ライト・スルーがこのアドレスに対して行なわれていないことを示す）、キャッシュ・エントリの状態は、移行110を介して状態Iから状態Nに移行する。ライト・スルーがこのアドレスに対して行なわれないので、キャッシュ・エントリのデータは、コヒーレントであるとは保証されることができない。ライト・スルーが行なわれないので、このキャッシュ・エントリへの書き込み処理は、システム相互接続12には位置せず、したがって、その処理は、キャッシュ・コヒーレンス・マネージャ19によって（または他のキャッシュによって）スヌープされることができない。したがって、コヒーレンス動作が行なわれないので、それは非コヒーレントと考えられる。キャッシュ・エントリが新しく割り付けられるので、そのデータは未修正であると考えられる。

【0038】

しかしながら、キャッシュ・ミスが生じ、且つ、W=1である場合（ライト・スルーがこのアドレスへの書き込みに対して行なわれることを示す）、キャッシュ・エントリの状態は、移行128を介して状態Iから状態Cに移行する。ライト・スルーがこのアドレスへの書き込みに対して行なわれるので、コヒーレンスが維持される。例えば、書き込みミスのライト・スルーに対して、移行128の際に新しく割り付けられたキャッシュ・エントリに記憶されたデータもまた、システム相互接続12を介してメモリ20に提供される。読み取りミスのライト・スルーに対しては、新しく割り付けられたキャッシュ・エントリへの記憶用データは、システム相互接続12を介してメモリ20によってプロセッサ14に提供される。したがって、ライト・スルーを行なう際に、新しく割り付けられたキャッシュ・エントリに記憶されたデータは、システム相互接続12を介して提供され、それは、キャッシュ・コヒーレンス・マネージャ18によって監視される。したがって、キャッシュ・コヒーレンス・マネージャ18は、ライト・スルー処理に対応したシステム相互接続12上のアドレスをスヌーピングすることによってキャッシュ・コヒーレンス動作を行なうことができる。例えば、キャッシュ・コヒーレンス・マネージャ18は、プロセッサ16にスヌープ無効要求を提供することができ、その結果、プロセッサ16は、スヌープしたアドレスがキャッシュ30においてヒットしたか否かを判断することができ、そうであれ

ば、それを無効にする。

【 0 0 3 9 】

状態Cで一度、キャッシュ・エントリに対する任意の読出しヒットまたは書込みヒットは、キャッシュ・エントリを状態Cのままにする。書込みヒットの場合、ライト・スルーが行なわれ（アドレスに対するWページ属性が設定されたので）、それは、コヒーレンスが維持されることを可能にする。つまり、ライト・スルーに際して、キャッシュ・コヒーレンス動作を行なうことができる。例えば、上述したように、キャッシュ・コヒーレンス・マネージャ18は、プロセッサ16へのスヌープ無効要求を生成することができ、その結果、コヒーレンスを維持することができる。キャッシュ・エントリ上のキャッシュ無効動作は、移行130を介してキャッシュ・エントリを状態Iに戻す。状態Cでは、キャッシュ・エントリのアドレスに関連したデータが別のプロセッサによって（プロセッサ16によって、など）修正されるとき、プロセッサ14は、キャッシュ・コヒーレンス・マネージャ18から、スヌープ無効要求などのキャッシュ・コヒーレンス動作を受信することが可能であることも注記しておく。この場合、スヌープ無効要求は、プロセッサ14におけるキャッシュ・エントリをヒットし、移行129を介してキャッシュ・エントリの状態を状態Iに移行し戻す。

10

【 0 0 4 0 】

さらに、キャッシュ・フラッシュ動作（それらは、フラッシュ・キャッシュ動作命令とも称されることがある）が、メモリ20における記憶位置に対してプロセッサ14または16によってそれぞれ行なわれることが可能であり、それによって、キャッシュ28またはキャッシュ30内にある任意の修正済データでメモリ20を更新する。一実施形態では、これらのキャッシュ・フラッシュ動作は、メモリ20への任意の修正済データをフラッシュし、その後、キャッシュ・エントリを有効状態（つまり、無効なしのキャッシュ・フラッシュ動作）のままにするか、または、これに代えて、メモリ20への任意の修正済データをフラッシュし、そして、このフラッシュを行なった後で、キャッシュ・エントリを無効状態（つまり、無効を伴うキャッシュ・フラッシュ動作）にするように、指定され得る。キャッシュ・フラッシュ動作は、キャッシュ内にあるアドレスに対応したデータの存在についての経験的知識なしに、または、そのデータを含んだキャッシュ・エントリの状態なしに、キャッシュにおけるルックアップに対してアドレスを提供する。したがって、一実施形態では、フラッシュ動作は、キャッシュ・エントリがいずれかの修正済データを有する場合、一致するキャッシュ・エントリがメモリに書き戻される動作である。つまり、キャッシュ・ラインまたはエントリがダーティ（dirty）である場合、それはメモリ（例えば、メモリ20）に書き戻される。修正済データがない場合、メモリへの書き戻しは必要ではない。無効を伴ってまたは無効なしに、フラッシュを行なうことも注記しておく。

20

30

【 0 0 4 1 】

依然として状態Cを参照して、無効なしのキャッシュ・フラッシュ動作を行なう際、Wページ属性は、状態間をどのように移行するかを判断するために再びチェックされる。無効なしのキャッシュ・フラッシュ動作に際して、Wページ属性が0にクリアされた場合（例えば、メモリ20における対応したページのWページ属性をソフトウェアが修正する結果として）、キャッシュ・エントリ状態は、移行134を介して状態Nに移行する。つまり、キャッシュ・エントリは、無効ではないが（無効なしにフラッシュが行なわれるので）、ここでは、ライト・スルーはもはや行なわれない。このように、再び、コヒーレンスは保証されない。また、フラッシュが要求されるので、そのデータは、フラッシュ動作の完了時に未修正であることが分かる。無効を伴ってフラッシュが行なわれた場合、そのフラッシュ動作が行なわれた後で、キャッシュ・エントリが無効にされ、したがって、（移行130を介して）状態Iに移行し戻る。無効なしにフラッシュが行なわれた場合、キャッシュ・エントリは、フラッシュ動作の後も有効なままであり、状態（状態Cのままか、または、状態Nに移行）は、Wページ属性に基づいて判断される。

40

【 0 0 4 2 】

50

状態Nで一度、キャッシュ・エントリへの任意の読出しヒットが、キャッシュ・エントリを状態Nのままにする。キャッシュ・エントリのキャッシュ無効動作は、移行112を介してキャッシュ・エントリを状態Iに戻す。無効なしのキャッシュ・フラッシュ動作は、Wページ属性の値に依存して状態Cに移行させることが可能である。例えば、無効なしのキャッシュ・フラッシュ動作が行なわれ、且つ、Wページ属性が依然として0のままである場合、キャッシュ・エントリの状態は、状態Nのままである。無効なしのキャッシュ・フラッシュ動作が行なわれ、且つ、Wページ属性が今度は1である場合（例えばソフトウェアによって変更された）、キャッシュ・エントリの状態は、状態Nから状態Cに移行し、ここでは、フラッシュによって、データは未修正のままであり、また、今行なわれているライト・スルーによって、コヒーレンスは維持される。無効を伴うフラッシュが行なわれた場合、そのフラッシュ動作の後で、キャッシュ・エントリの状態は、（移行112を介して）Iに移行し戻す。状態Nでのキャッシュ・エントリへの書込みヒットの際、キャッシュ・エントリの状態は、移行114を介して状態Nから状態Mに移行する。したがって、キャッシュ・エントリは、ライト・スルーが行なわれないので、依然として非コヒーレントであるが、ここでは、それは、状態Iから状態Nへの移行110に際して割り付けられたので、修正されたと考えられる。

【0043】

状態Mで一度、キャッシュ・エントリへの任意のヒットがキャッシュ・エントリを状態Mのままにする。無効なしのキャッシュ・フラッシュ動作が行なわれた場合、上述したように、メモリ（例えば、メモリ20）へのコピーバックが行なわれ、キャッシュ・エントリは無効にされない。その後、Wページ属性に基づいて、キャッシュ・エントリは、状態Mから状態NまたはCに移行する。その後、W=0の場合、キャッシュ・エントリ状態は、移行116を介して状態Nに移行し戻され、ここでは、データは、（フラッシュ動作により）再び未修正であり、ライト・スルーは行なわれない（したがって、コヒーレンスは保証されない）。W=1の場合、キャッシュ・エントリの状態は、移行126を介して状態Cに移行し戻され、ここでは、データは、（フラッシュ動作により）再び未修正であり、ライト・スルーが行なわれる（したがってコヒーレンスを維持される）。状態Mでは、無効を伴うフラッシュが行なわれた場合、上述したように、メモリへのコピーバックが行なわれ、キャッシュ・エントリが無効にされ、キャッシュ・エントリの状態は、移行124を介して状態Iに戻される。状態Mで置換動作がキャッシュ・エントリ上で行なわれる場合、修正済データは、まずメモリ20にコピーし戻される。その後、キャッシュ・エントリのデータは、まず無効にされ（移行124を介して状態Iに移行する）、次に、上書きされる（割付けが有効に生じるので、Wページ属性の値に依存して状態NまたはCへの移行をもたらす）。また、無効動作が行なわれる場合、キャッシュ・エントリの状態は、コピーバック動作を行なうことなしに、移行122を介して状態Mから状態Iに移行する。

【0044】

したがって、キャッシュ・コヒーレンス・プロトコルがコヒーレントおよび非コヒーレント書込みに対して個別の移行および状態をどのように使用するかを評価することができ。一実施形態では、キャッシュ・コヒーレンス・プロトコル100は、コヒーレント状態間の一または複数の移行を規定するために、制御値としてMMU42によって提供されるWページ属性を使用する。一実施形態では、キャッシュ・コヒーレンス・プロトコルは、コヒーレント・トラフィックに対するライト・スルー動作に依存し、非コヒーレント・データに対するコピーバック処理をサポートする。したがって、キャッシュ・コヒーレンス・プロトコル100は、キャッシュ・コヒーレンス動作に対して提供しない標準バス・プロトコルと共に使用されることが可能である。

【0045】

一実施形態では、キャッシュ・コヒーレンス・マネージャ18は、キャッシュ・コヒーレンスが必要なバス・トランザクションの一部を選択することが可能である。つまり、一または複数の要因に基づいて、図4～図9を参照して下で議論されるように、特定の種類

10

20

30

40

50

のトランザクションがスヌープされるかまたはスヌープされず、したがって、キャッシュ・コヒーレンス動作をもたらすこともあれば、もたらさないこともある。さらに、一実施形態では、選択的にスヌープを行なうために使用される一または複数の要因は、A M B A A H BまたはA X Iなどの既存の業界標準バス・プロトコルである。このように、追加の信号伝達機構または標準バス・プロトコルへの他の修正は必要ない。

【 0 0 4 6 】

一実施形態では、キャッシュ・コヒーレンス・マネージャ 1 8 は、スヌープし、バス・トランザクションが単一のビート処理またはバースト処理であるか否かに基づいてキャッシュ・コヒーレンス動作を行なう。例えば、一実施形態では、スヌープおよびキャッシュ・コヒーレンス動作は、バス・トランザクションが単一のビート処理であるがバースト処理ではない場合に行なわれる。これは、プロセッサによって開始されたバーストが典型的にはコピーバック処理をする可能性があるからであり、したがって、キャッシュ・コヒーレンス動作は、これらのバースト処理には必要ではない。しかしながら、一実施形態では、処理がバースト処理であるが、プロセッサ（プロセッサ 1 4 または 1 6 等）によってではなく、むしろ D M A 4 0 によって開始される場合、典型的には D M A によって開始されたバースト処理がコピーバックを行なわないので、スヌープは依然として行なわれ、これは、キャッシュ・コヒーレンス動作が依然として必要である場合があることを意味している。したがって、一実施形態では、スヌープすべきか否かの判断は、両方の種類のバス・トランザクション、および、どのマスタまたはどの種類のマスタがバス・トランザクションを開始したかに基づくことが可能である。以下に記述されるように、他の要因または要因の組合せが、スヌープが行なわれるべきか否かを判断するために、キャッシュ・コヒーレンス・マネージャ 1 8 または他のキャッシュ・コヒーレンス制御回路等によってシステム 1 0 内で使用されることが可能である。一実施形態では、マスタ識別子（マスタ I D）は、どのマスタが処理を開始したか識別する各バス・トランザクションを与えられることを注記しておく。マスタ地位の判断は、転送アドレスの属性として開始マスタによってバス・トランザクションの一部として示されるか、または、他の方法で示されるか信号伝達されることが可能である。一実施形態では、バースト書込みアクセスが開始される場合、どのマスタがコヒーレンス動作を行なうかの判断は、システム 1 0 のユーザによってプログラムで選択されることが可能である。一実施形態では、コヒーレンス動作が行なわれるべき一または複数のマスタを示すために、転送の種類（バーストもしくは非バースト、または 1 組の特定のバースト種類の中から選択）に基づいて、制御レジスタがシステム 1 0 内に設けられることが可能である。

【 0 0 4 7 】

例として A M B A A H B プロトコルを使用して、バースト情報は、8 つの可能性がある種類の処理を符号化する 3 ビットのバス信号、H B U R S T [2 : 0] を使用して提供される。これらの符号化は図 4 に提供されている。例えば、H B U R S T に対する「0 0 0」の値は、単一の転送（つまり、単一のビート処理）を示している一方で、他の 7 つの値は、異なる種類のバースト処理を示している。一実施形態では、H B U R S T の値が 0 0 0（単一の転送を示す）であることをキャッシュ・コヒーレンス・マネージャ 1 8 が判断したとき、スヌープが行なわれる。しかしながら、H B U R S T の値が 0 1 0（4 ビートのラップ・バーストを示す）、0 1 1（4 ビートのインクリメント・バーストを示す）、1 0 0（8 ビートのラップ・バーストを示す）、1 0 1（8 ビートのインクリメント・バーストを示す）、1 1 0（1 6 のビートラップ・バーストを示す）、または 1 1 1（1 6 ビートのインクリメント・バーストを示す）であると判断された場合、スヌープは行なわれな。実施形態によっては、H B U R S T の値が 0 0 1（無指定の長さのインクリメント・バーストを示す）である場合、スヌープは行なわれたり行なわれなかったりする。上で議論したように、スヌープすべきか否かの決定は、さらに、どのマスタまたはどの種類のマスタがバス・トランザクションまたは転送を開始したかに基づくことが可能である。

【 0 0 4 8 】

一実施形態では、スヌープすべきか否かの決定は、さらに、現在のアクセス（現在のバス・トランザクション）がキャッシュすることができるか否かによって特定される。例としてAMBA AHBプロトコルを依然として使用して、HPROT[3:0]信号は、バス・トランザクションのアクセス・アドレスを備え、また、例えば、転送がオペレーション・コードのフェッチまたはデータ・アクセスであるか、特権モードアクセスまたはユーザ・モード・アクセスであるかを示し、そして、MMUを備えたマスタに対して、現在のアクセスがキャッシュすることができるか否かを示す。HPROT[3:0]信号の符号化は、図5に示されている。例えば、HPROT[3]が0の値を有する場合、バス・トランザクションのアクセスはキャッシュすることができないので、スヌープは行なわれないが、その一方で、HPROT[3]が1の値を有する場合、バス・トランザクションのアクセスはキャッシュすることができるので、スヌープは行なわれる。したがって、上述した要因の使用に加えてまたはそれに代えて、HPROT信号は、スヌープが行なわれるべきか否かを判断するために使用されることも可能である。別の例の実施形態では、HPROT[2]が0の値を有する場合、バス・トランザクションのアクセスはバッファすることができないので、スヌープは行なわれない一方で、HPROT[2]が1の値を有する場合、バス・トランザクションのアクセスはバッファすることができるので、スヌープが行なわれる。幾つかの実施形態では、バス・アクセスを開始する特定のマスタに応じて、HPROTビットの値は、コヒーレンス動作が行なわれるか否かを制御するためにプログラムで使用されることが可能である。したがって、上述した要因の使用に加えてまたはそれに代えて、HPROT信号は、処理を開始するマスタの識別の判断と関連して、随意に、スヌープすべきか否かを判断するために使用されることも可能である。例えば、システム10のユーザは、HPROT[3]=1である場合にマスタXに対して、HPROT[2]=1である場合にマスタYに対して、HPROT[3]=1およびHPROT[2]=0である場合にマスタZに対して行なわれるコヒーレンス動作を達成するように、一または複数の制御レジスタまたはレジスタ・フィールドをプログラムすることが可能である。これに代えて、他の実現可能な制御の組合せが使用されることも可能である。

【0049】

図6～図9は、キャッシュ・コヒーレンス動作（例えば、スヌープ）が行なわれるべきか否かを判断するためのフローチャートの様々な例を示している。上で議論したように、キャッシュ・コヒーレンス動作は、他のキャッシュへのスヌープ要求（もしあれば）を、それらのキャッシュが次にそのスヌープ要求を処理するシステム10内で生成することを含むことが可能である。一実施形態では、キャッシュ・コヒーレンス動作は、キャッシュ・エントリの状態の変更を含むことが可能である。一実施形態では、フローの各々は、キャッシュ・コヒーレンス・マネージャ18によって実行されることが可能である。これに代えて、他のキャッシュ・コヒーレンス制御回路は、これらの機能のうちのいずれかを行ない、システム10における任意の場所に存在することが可能である。一実施形態では、個々のキャッシュまたはキャッシュ制御回路は、図6～図9に示される機能を行なうことが可能である。図6は、ブロック502から始まるフロー500を示しており、ここでは、例えばプロセッサ14、プロセッサ16、DMA40、または他のマスタ22等のバス・マスタによってバス・トランザクションが開始される。フローは判断ダイヤモンド504に進み、ここでは、書込み処理が行なわれているか否かが判断される。そうでなければ、フローは、ブロック514に進み、ここでは、キャッシュ・コヒーレンス動作は行なわれない。つまり、図6の例示の実施形態では、読出し処理はスヌープされない。判断ダイヤモンド504では、バス・トランザクションが書込み処理であると判断された場合、フローは判断ダイヤモンド506に進み、ここでは、バス・トランザクションが単一のビート（つまり、単一の転送）処理であるか否かが判断される。そうでなければ、フローは、ブロック514に進む。バス・トランザクションが単一のビート処理である場合、フローは、ブロック512に進み、ここでは、書込み処理のデータに対するキャッシュ・コヒーレンス動作が行なわれる。したがって、フロー500では、バス・トランザクションが単一のビート書込み処理である場合にのみ、キャッシュ・コヒーレンス動作が行なわれる。

10

20

30

40

50

【 0 0 5 0 】

図 7 は、例えば、プロセッサ 1 4、プロセッサ 1 6、DMA 4 0、または他のマスタ 2 2 等のバス・マスタによってバス・トランザクションが開始されるブロック 6 0 2 から始まるフロー 6 0 0 を示している。フローは判断ダイヤモンド 6 0 4 に進み、ここでは、書込み処理が行なわれているか否かが判断される。そうでなければ、フローは、ブロック 6 1 4 に進み、キャッシュ・コヒーレンス動作は行なわれない。つまり、図 7 の例示の実施形態では、読出し処理はスヌープされない。判断ダイヤモンド 6 0 4 では、バス・トランザクションが書込み処理であると判断された場合、フローは判断ダイヤモンド 6 0 6 に進み、ここでは、バス・トランザクションが単一のビート（すなわち、単一の転送）処理であるか否かが判断される。そうであれば、フローは、ブロック 6 1 2 に進み、ここでは、書込み処理のデータに対するキャッシュ・コヒーレンス動作が行なわれる。そうでなければ、フローは、ブロック 6 0 8 に進み、ここでは、その処理のマスタ ID が判断される。つまり、上述したように、一実施形態では、マスタ ID は、開始マスタを識別するために各バス・トランザクションを与えられる。次いで、フローは判断ダイヤモンド 6 1 0 に進み、ここでは、マスタ ID が DMA 4 0 等の DMA を示すか否かが判断される。そうであれば、フローは、6 1 2 ブロックに進み、ここでは、書込み処理のデータに対するキャッシュ・コヒーレンス動作が行なわれる。そうでなければ、フローは、ブロック 6 1 4 に進む。つまり、書込み処理がバースト書込み（つまり、判断ダイヤモンド 6 0 6 からの「N O」の分岐先）であっても、キャッシュ・コヒーレンス動作は、開始マスタが DMA ならば、依然として行なわれる。しかしながら、書込み処理がバースト書込みであり、且つ、開始マスタが DMA でない場合、キャッシュ・コヒーレンス動作は行なわれない。なぜならば、この場合、バースト処理がコピーバック処理であると思われるからである。

【 0 0 5 1 】

図 8 は、例えば、プロセッサ 1 4、プロセッサ 1 6、DMA 4 0、または他のマスタ 2 2 等のバス・マスタによってバス・トランザクションが開始されるブロック 7 0 2 から始まるフロー 7 0 0 を示している。フローは判断ダイヤモンド 7 0 4 に進み、ここでは、書込み処理が行なわれているか否かが判断される。そうでなければ、フローはブロック 7 1 4 に進み、ここでは、キャッシュ・コヒーレンス動作は行なわれない。つまり、図 8 の例示の実施形態では、読出し処理はスヌープされない。判断ダイヤモンド 7 0 4 では、バス・トランザクションが書込み処理であると判断された場合、フローは判断ダイヤモンド 7 0 6 に進み、ここでは、バス・トランザクションが単一のビート（つまり、単一の転送）処理であるか否かが判断される。そうであれば、フローはブロック 7 1 2 に進み、ここでは、書込み処理のデータに対するキャッシュ・コヒーレンス動作が行なわれる。そうでなければ、フローはブロック 7 0 8 に進み、ここでは、その処理のマスタ ID が判断される。つまり、上述したように、一実施形態では、マスタ ID は、開始マスタを識別するために各バス・トランザクションを与られている。その後、フローは判断ダイヤモンド 7 1 0 に進み、ここでは、マスタ ID が、選択されたマスタを示すか否かが判断される。そうであれば、フローはブロック 7 1 2 に進み、ここでは、書込み処理のデータに対するキャッシュ・コヒーレンス動作が行なわれる。そうでなければ、フローはブロック 7 1 4 に進む。一実施形態では、キャッシュ・コヒーレンス・マネージャ 1 8 は、どのマスタがコヒーレンスを要求するかを示すコヒーレント・マスタ制御レジスタ 1 9 を備えている。例えば、制御レジスタ 1 9 は、キャッシュ・コヒーレンスが望まれるかまたは要求されるシステム 1 0 におけるそれらのマスタの ID でプログラム可能であり得る。したがって、図 8 の判断ダイヤモンド 7 1 0 では、ブロック 7 0 2 でのバス・トランザクションを開始するマスタのマスタ ID は、制御レジスタ 1 9 と比較されることができ、それによって、開始マスタのマスタ ID が制御レジスタ 1 9 によって示された選択されたマスタに対応するかどうかを判断する。これに代えて、他の回路または他の方法が、キャッシュ・コヒーレンス動作が行なわれるべきである開始マスタを識別するために使用されることが可能である。したがって、書込み処理がバースト書込み（つまり、判断ダイヤモンド 6 0 6 からの「N O」の分岐先）であっても、キャッシュ・コヒーレンス動作は、開始マスタがコヒーレン

ト・マスタ制御レジスタ１９によって選択されたマスタならば、依然として行なわれることが可能である。代替の実施形態では、判断ステップ７１０が、システム１０内の他の所定の選択基準を利用して行なわれることが可能である。一実施形態では、この選択基準は、システム１０のユーザによってプログラムされることが可能である。

【００５２】

図９は、例えば、プロセッサ１４、プロセッサ１６、DMA４０、または他のマスタ２２等のバス・マスタによってバス・トランザクションが開始されるブロック８０２から始まるフロー８００を示している。フローは判断ダイヤモンド８０４に進み、ここでは、書込み処理が行なわれているか否かが判断される。そうでなければ、フローはブロック８１４に進み、ここでは、キャッシュ・コヒーレンス動作は行なわれない。つまり、図９の例示の実施形態では、読出し処理はスヌープされない。判断ダイヤモンド８０４では、バス・トランザクションが書込み処理であると判断された場合、フローは判断ダイヤモンド８０６に進み、ここでは、バス・トランザクションのキャッシュ可能性フィールドまたは信号が読み出される。例えば、これは、上述したHPROT[３]またはHPROT[２]を参照することができる。その後、フローは判断ダイヤモンド８０８に進み、ここでは、キャッシュ可能性フィールドまたは信号が、キャッシュ可能な処理を示すか否かが判断される。そうでなければ、フローはブロック８１４に進む。そうであれば、フローは判断ダイヤモンド８１０に進み、ここでは、書込み処理が単一のビート書込みであるか否かが判断される。そうでなければ、フローはブロック８１４に進む。そうであれば、フローはブロック８１２に進み、ここでは、キャッシュ・コヒーレンス動作が書込み処理に対して行なわれる。つまり、図９の例では、キャッシュ・コヒーレンス動作は、キャッシュ可能な単一の書込み処理に対してのみ行なわれる。つまり、バースト処理は、例えば、システム相互接続１２のバス・プロトコルに応じて与えられる信号によってキャッシュ可能でない処理として示される任意の処理もそうであるように、キャッシュ・コヒーレンス動作をもたらさない。代替の実施形態では、判断ステップ８０８は、図８のステップ７０８および７１０によって行なわれる判断と同様の処理を開始するマスタを識別する判断ステップの後に続くことが可能である。一実施形態では、この判断の結果は、キャッシュ・コヒーレンス動作がバス・トランザクションに対して行なわれるべきか否かを判断するさらなる要因として作用することが可能である。

【００５３】

したがって、キャッシュ・コヒーレンス動作（例えば、スヌープ）が行なわれるか否かを判断するために、単独でまたは互いに様々な組合せで、種々多様な要因がどのように使用されることがあるかを認識することができる。キャッシュ・コヒーレンス動作のこの選択的な動作は、例えば、既存の標準バス・プロトコルを使用して実行されることが可能である。

【００５４】

本発明を実施する装置が、大部分は、当業者に知られる電子部品および回路からなるので、回路詳細は、本発明の基本的概念についての理解および認識のために、および、本発明の教示を不明瞭にしたり混乱させたりしないために、上に例示したような必要と考えられるものよりも広い範囲では説明されない。

【００５５】

本発明は、比導電率型または電位の極性に関して記述されたが、当業者は、比導電率型および電位の極性が逆にされることがあることを認識する。

さらに、本記述および特許請求の範囲における用語「前」、「後」、「上」、「下」、「上側」、「下側」など（もしあれば）、説明の目的のために使用されるものであって、必ずしも恒久的な相対的位置について記述するためのものではない。そのように使用される用語は、適切な状況下では置換可能であり、その結果、ここで記述される本発明の実施形態は、例えば、ここで例示されたものまたは記述されたものの以外の他の配置で実施されることができると理解される。

【００５６】

ここで使用される用語「プログラム」は、コンピュータ・システム上で実行するように設計された一連の命令として定義される。プログラムまたはコンピュータ・プログラムは、サブルーチン、関数機能、手順、オブジェクト方法、オブジェクト実施、実行可能なアプリケーション、アプレット、サーブレット、ソース・コード、オブジェクト・コード、共有ライブラリ/ダイナミック・ロード・ライブラリ、および/またはコンピュータ・システム上で実行するように設計された他の一連の命令を含むことが可能である。

【 0 0 5 7 】

上記の実施形態のうちの幾つかは、適宜状況に合わせて、種々多様な情報処理システムを使用して実施されることが可能である。例えば、図 1 およびその議論は、典型的な情報処理アーキテクチャについて記述しているが、この典型的なアーキテクチャは、本発明の様々な態様についての議論の際に有用な参照を単に提供するために示されている。勿論、そのアーキテクチャの記述は、議論の目的のために簡略化されており、それは、本発明に従って使用されることが可能である多くの異なる種類の適切なアーキテクチャの単に 1 種類である。当業者は、ロジック・ブロック間の境界が単に例示であり、その代替の実施形態がロジック・ブロックまたは回路要素を結合するか、様々なロジック・ブロックまたは回路要素に機能の代替の分解を課することが可能であることを認識する。

【 0 0 5 8 】

このように、ここで図示されるアーキテクチャが単に典型例であり、実際には、同一の機能を達成する多くの他のアーキテクチャを実施することができることが理解される。抽象的な意味で、しかしながら依然として有限の意味で、同一の機能を達成する構成要素の任意の配置は、所望の機能が達成されるように有効に「関連付けられる」。したがって、特定の機能を達成するためにここで組み合わせられた任意の 2 つの構成要素は、互いに「関連付けられる」ものと見なされることができ、その結果、所望の機能は、アーキテクチャまたは中間構成要素に関係なく、達成される。同様に、そのように関連付けられた任意の 2 つの構成要素は、所望の機能を達成するために、互いに「操作可能に接続される」または「操作可能に結合される」ものとして見られることもできる。

【 0 0 5 9 】

さらに、例えば、一実施形態では、システム 10 の図示の要素は、単一の集積回路上にまたは同一の装置内に配置された回路である。これに代えて、システム 10 は、互いに相互接続された任意の数の個別の集積回路または個別の装置を備えることが可能である。例えば、メモリ 20 は、プロセッサ 14 および 16 として同一の集積回路上に、または、個別の集積回路上に配置されるか、または、システム 10 の他の要素から個別に分離された別の周辺機器またはスレーブ内に配置されることが可能である。他のスレーブ 24 および I/O 回路 26 は、個別の集積回路または装置上に配置されることが可能である。さらに、例えば、システム 10 またはその一部は、物理的な回路または該物理的な回路に変換可能な論理的表現のソフトもしくはコード表現であることが可能である。そのため、システム 10 は、任意の適切な種類のハードウェア記述言語で具現化されることが可能である。

【 0 0 6 0 】

さらに、当業者は、上述した動作の機能間の境界が単に実例であることを認識する。複数の動作の機能は、単一の動作に組み合わせられることが可能であり、および/または、単一の動作の機能は、追加の動作に分散されることが可能である。さらに、代替の実施形態は、特定の動作の複数の例を含むことが可能であり、動作の順序は、様々な他の実施形態において変更されることが可能である。

【 0 0 6 1 】

ここで記述されたソフトウェアのすべてまたは幾つかは、例えば、メモリ 35 または他のコンピュータ・システム上の他の媒体等のコンピュータ読取り可能な媒体からシステム 10 の要素を受信することが可能である。そのようなコンピュータ読取り可能な媒体は、システム 10 等の情報処理システムに恒久的に取外し可能にまたは遠隔に結合されることが可能である。コンピュータ読取り可能な媒体は、幾つか例を挙げると、例えば、限定なしに、ディスクおよびテープ記憶媒体を含む磁気記憶媒体；コンパクトディスク媒体（例

10

20

30

40

50

えば、CD-ROM、CD-Rなど)等の光記憶媒体およびデジタル・ビデオ・ディスク記憶媒体;FLASHメモリ、EEPROM、EPROM、ROMなどの半導体ベースのメモリ・ユニットを含む不揮発性メモリ記憶媒体;強磁性デジタル・メモリ;MRAM;レジスタ、バッファまたはキャッシュ、メイン・メモリ、RAMなどを含む揮発性記憶媒体;コンピュータ・ネットワーク、ポイント・ツー・ポイント通信機器および搬送波送信媒体を含むデータ伝送媒体、のうちの任意のものを含むことが可能である。

【0062】

一実施形態では、システム10は、パーソナル・コンピュータ・システム等のコンピュータ・システムである。他の実施形態は、異なる種類のコンピュータ・システムを含むことが可能である。コンピュータ・システムは、一または複数のユーザに対して独立した演算能力を与えるように設計されることができ、情報処理システムである。コンピュータ・システムは、多くの形態で表されることがあり、メインフレーム、ミニコンピュータ、サーバ、ワークステーション、パーソナル・コンピュータ、ノートパッド、携帯情報端末、コンピュータ・ゲーム、自動車および他の内蔵システム、携帯電話および様々な他の無線装置を含むが、これらに限定されるものではない。典型的なコンピュータ・システムは、少なくとも1つの処理装置、関連するメモリ、および多数の入力/出力(I/O)装置を備えている。

10

【0063】

コンピュータ・システムは、プログラムに従って情報を処理し、I/O装置を介して結果としての出力情報を生成する。プログラムは、特定のアプリケーション・プログラムおよび/またはオペレーティング・システム等の命令のリストである。コンピュータ・プログラムは、典型的には、コンピュータ読取り可能な記憶媒体上で内部に記憶されるか、または、コンピュータ読取り可能な送信媒体を介してコンピュータ・システムに送信される。コンピュータ・プロセスは、典型的には、実行(ラン)プログラムまたはプログラムの一部、現在のプログラム値および状態情報、ならびにそのプロセスの実行を管理するためにオペレーティング・システムによって使用されるリソースを含んでいる。親処理は、親処理の全体的な機能を行なうのを支援するために、他の子処理を生じさせることが可能である。親処理が、特に、該親処理の全体的な機能の一部を行なうために子処理を生じさせるので、子処理(および孫処理など)によって行なわれる機能は、親処理によって行なわれるように場合によっては記述されることがある。

20

30

【0064】

本発明は、ここでは特定の実施形態に関して記述されているが、様々な改変および変更が、下の特許請求の範囲で述べられるような本発明の範囲から逸脱することなしになされることができる。したがって、本明細書および図面は、限定的な意味ではなく、むしろ実例的な意味とみなさるべきであり、すべてのそのような改変は、本発明の範囲に含まれるものとして意図されている。特定の実施形態に関してここで記述される如何なる恩恵、利点、または問題に対する解決策も、いずれかまたはすべての請求項の重大な、要求された、本質的な特徴または要素として解釈されるようには意図されていない。

【0065】

ここで使用される用語「結合」は、直接的な結合または機械的な結合に限定されるようには意図されていない。

40

さらに、ここで使用される用語「a」または「an」は、1つまたは1以上として定義される。また、特許請求の範囲での「少なくとも1つ」および「一または複数」などの導入句の使用は、不定冠詞「a」または「an」による別の請求項要素の導入が、同一の請求項が導入句「一または複数」または「少なくとも1つ」および「a」または「an」などの不定冠詞を含む場合であっても、1つだけそのような要素を含んだ発明への導入請求項要素を含んだ任意の特定の請求項を限定することを意味するように解釈されるべきではない。同じことが定冠詞の使用にも該当する。

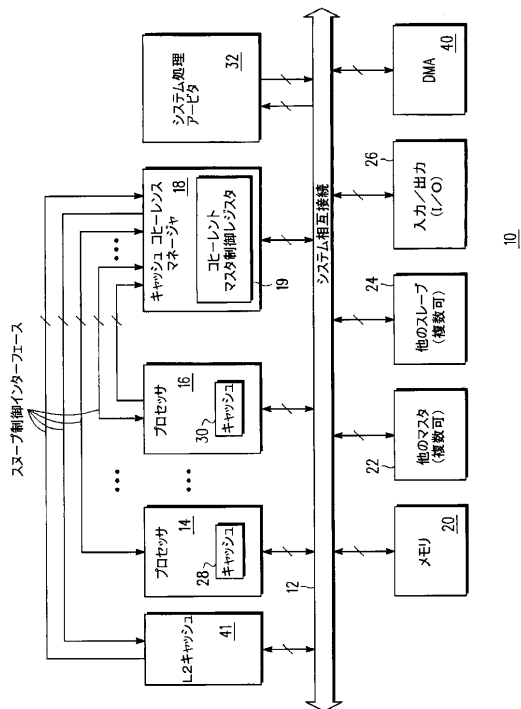
【0066】

特に述べない限り、用語「第1」および「第2」は、そのような用語が記述する要素間

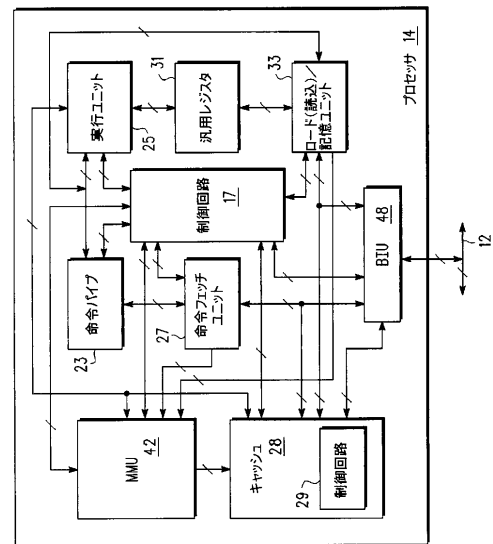
50

の任意の識別に使用される。このように、これらの用語は、そのような要素の時間的または他の優先順位を示すようには必ずしも意図されていない。

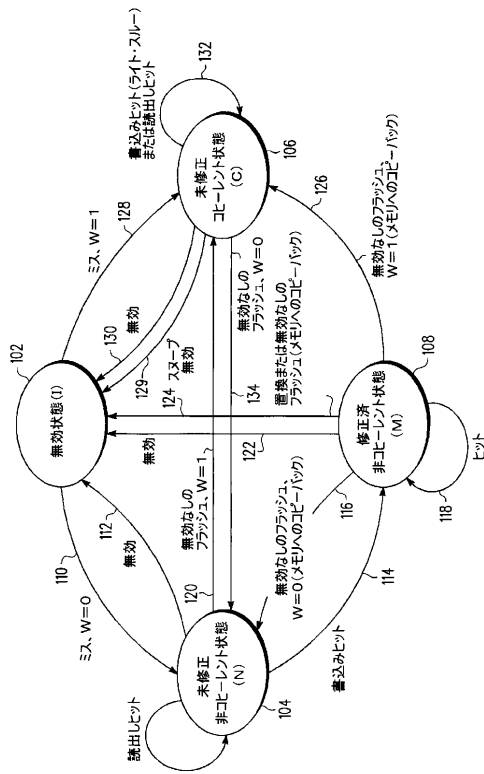
【図 1】



【図 2】



【図 3】



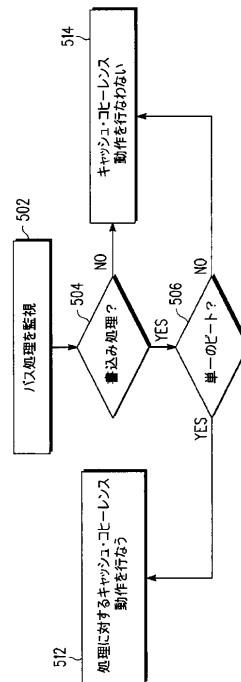
【図 5】

HPROT[3]	HPROT[2]	HPROT[1]	HPROT[0]	説明
キャッシュ可能	バフア可能	特権あり	データ/オPCODE	オペコードフェッチ
-	-	-	0	データアクセス
-	-	-	1	ユーザアクセス
-	-	0	-	特権アクセス
-	-	1	-	バフア可能
-	0	-	-	バフア可能
-	1	-	-	バフア可能
0	-	-	-	キャッシュ不可能
1	-	-	-	キャッシュ可能

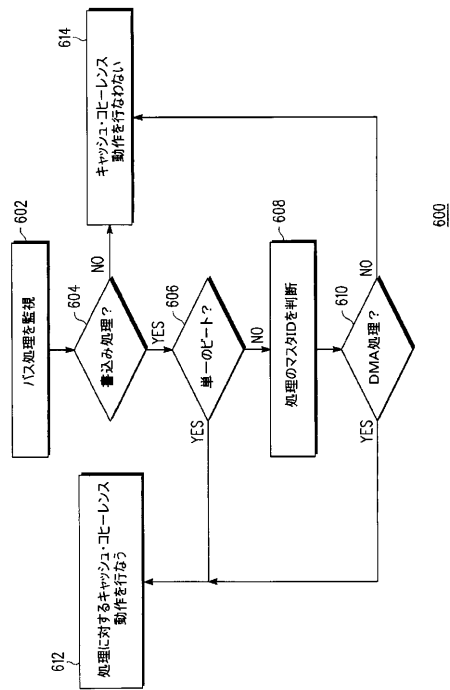
【図 4】

HBURST[2:0]	種類	説明
000	SINGLE	単一の転送
001	INCR	無指定の長さのインクリメント・バースト
010	WRAP4	4ビットのラップ・バースト
011	INCR4	4ビットのインクリメント・バースト
100	WRAP8	8ビットのラップ・バースト
101	INCR8	8ビットのインクリメント・バースト
110	WRAP16	16ビットのラップ・バースト
111	INCR16	16ビットのインクリメント・バースト

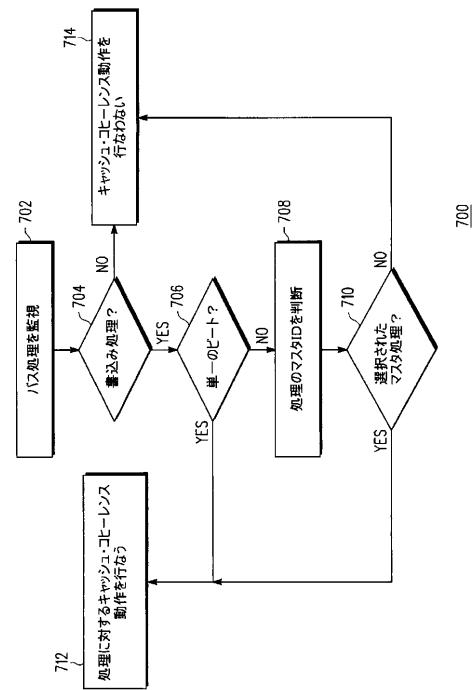
【図 6】



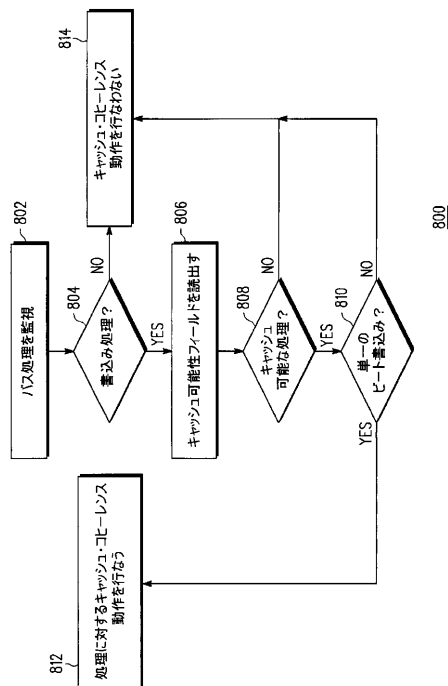
【図 7】



【図 8】



【図 9】



フロントページの続き

(56)参考文献 特開平04 - 137052 (JP, A)
特開2005 - 025726 (JP, A)
特開平05 - 241954 (JP, A)
米国特許出願公開第2008 / 0091884 (US, A1)
米国特許出願公開第2005 / 0071573 (US, A1)

(58)調査した分野(Int.Cl., DB名)
G06F 12 / 08