

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第4569628号
(P4569628)

(45) 発行日 平成22年10月27日 (2010.10.27)

(24) 登録日 平成22年8月20日 (2010.8.20)

(51) Int.Cl.		F I			
G06F 12/00	12/00	(2006.01)	G06F	12/00	560B
G06F 12/02	12/02	(2006.01)	G06F	12/00	560E
			G06F	12/00	571A
			G06F	12/02	590B

請求項の数 20 (全 18 頁)

(21) 出願番号	特願2007-338860 (P2007-338860)	(73) 特許権者	000004237
(22) 出願日	平成19年12月28日 (2007.12.28)		日本電気株式会社
(65) 公開番号	特開2009-157886 (P2009-157886A)		東京都港区芝五丁目7番1号
(43) 公開日	平成21年7月16日 (2009.7.16)	(74) 代理人	100103894
審査請求日	平成19年12月28日 (2007.12.28)		弁理士 冢入 健
		(72) 発明者	小林 浩二
			東京都港区芝五丁目7番1号 日本電気株式会社内
		(72) 発明者	萩原 孝
			東京都港区芝五丁目7番1号 日本電気株式会社内
		(72) 発明者	加納 健
			東京都港区芝五丁目7番1号 日本電気株式会社内

最終頁に続く

(54) 【発明の名称】 ロードストアキューの制御方法及びその制御システム

(57) 【特許請求の範囲】

【請求項1】

主記憶部に対して発行するリクエストを保持するロードストアキューの制御方法であって、

前記ロードストアキュー内の前記リクエストのうちストアリクエストを、前記ロードストアキューに対して新たに発行されるリクエストの個数が所定の個数となるまでの間前記ロードストアキュー内に保持し、

当該保持されたストアリクエストに後続するロードリクエストの前記主記憶部への前記リクエストの発行順序を、当該保持されたストアリクエストよりも先に並び替える

ロードストアキューの制御方法。

【請求項2】

前記ストアリクエストに後続して発行される前記リクエストの個数をカウントし、当該カウント値が所定の値となるまでの間、前記ロードストアキュー内に前記ストアリクエストを保持する

ことを特徴とする請求項1記載のロードストアキューの制御方法。

【請求項3】

前記ロードストアキュー内の第1のリクエストのアドレスと第2のリクエストのアドレスが前記主記憶部において同一の処理単位に含まれるアドレスである場合には、前記第1のリクエストを前記主記憶部に発行する際に、前記第2のリクエストを併せて前記主記憶部へと発行する

ことを特徴とする請求項 1 又は 2 記載のロードストアキューの制御方法。

【請求項 4】

前記第 1 のリクエストと前記第 2 のリクエストは共にロードリクエストか、又は、共にストアリクエストである

ことを特徴とする請求項 3 記載のロードストアキューの制御方法。

【請求項 5】

前記処理単位は同一ロー単位であり、当該処理単位に含まれるアドレスが同一のローアドレスに含まれるアドレスである

ことを特徴とする請求項 3 又は 4 記載のロードストアキューの制御方法。

【請求項 6】

前記処理単位は同一ランク単位であり、当該処理単位に含まれるアドレスが同一のランクアドレスに含まれるアドレスである

ことを特徴とする請求項 3 又は 4 記載のロードストアキューの制御方法。

【請求項 7】

前記ロードストアキュー内の第 1 のストアリクエストと第 2 のストアリクエストについて、当該第 2 のストアリクエストのアドレスと前記第 1 のストアリクエストのアドレスを比較し、前記第 2 のストアリクエストのアドレスが前記第 1 のストアリクエストのアドレスと同一である場合には、前記第 2 のストアリクエストを前記第 1 のストアリクエストにマージする

ことを特徴とする請求項 1 又は 2 記載のロードストアキューの制御方法。

【請求項 8】

前記主記憶部を D R A M で構成する

ことを特徴とする請求項 1 乃至 7 いずれか 1 項記載のロードストアキューの制御方法。

【請求項 9】

前記主記憶部をシンクロナス D R A M で構成する

ことを特徴とする請求項 1 乃至 7 いずれか 1 項記載のロードストアキューの制御方法。

【請求項 10】

前記主記憶部を D R A M 又はシンクロナス D R A M を用いた D I M M 若しくは S I M M で構成する

ことを特徴とする請求項 1 乃至 7 いずれか 1 項記載のロードストアキューの制御方法。

【請求項 11】

主記憶部に対して発行するリクエストを保持するロードストアキューと、当該ロードストアキューを制御する制御部と、を備えたロードストアキューの制御システムであって、前記制御部は、

前記ロードストアキュー内の前記リクエストのうちストアリクエストを、前記ロードストアキューに対して新たに発行されるリクエストの個数が所定の個数となるまでの間前記ロードストアキュー内に保持するストアリクエスト制御手段と、

当該保持されたストアリクエストに後続するロードリクエストの前記主記憶部への前記リクエストの発行順序を、当該保持されたストアリクエストよりも先に並び替えるロードリクエスト制御手段と、を備える

ロードストアキューの制御システム

【請求項 12】

前記ストアリクエスト制御手段は、

前記ストアリクエストに後続して発行される前記リクエストの個数をカウントするリクエスト計測手段を備え、

前記リクエスト計測手段によりカウントされたカウント値が所定の値となるまでの間、前記ロードストアキュー内に前記ストアリクエストを保持する

ことを特徴とする請求項 11 記載のロードストアキューの制御システム。

【請求項 13】

前記制御部は、

10

20

30

40

50

前記ロードストアキュー内の第1のリクエストのアドレスと第2のリクエストのアドレスが前記主記憶部において同一の処理単位に含まれるアドレスであるか否かを判定するアドレス判定手段を備え、

前記アドレス判定手段による判定の結果、前記第1のリクエストのアドレスと前記第2のリクエストのアドレスが前記主記憶部において同一の処理単位に含まれるアドレスである場合には、前記第1のリクエストを前記主記憶部に発行する際に、前記第2のリクエストを併せて前記主記憶部へと発行する

ことを特徴とする請求項11又は12記載のロードストアキューの制御システム。

【請求項14】

前記第1のリクエストと前記第2のリクエストは共にロードリクエストか、又は、共にストアリクエストである

10

ことを特徴とする請求項13記載のロードストアキューの制御システム。

【請求項15】

前記処理単位は同一ロー単位であり、当該処理単位に含まれるアドレスが同一のローアドレスに含まれるアドレスである

ことを特徴とする請求項13又は14記載のロードストアキューの制御システム。

【請求項16】

前記処理単位は同一ランク単位であり、当該処理単位に含まれるアドレスが同一のランクアドレスに含まれるアドレスである

ことを特徴とする請求項13又は14記載のロードストアキューの制御システム。

20

【請求項17】

前記制御部は、

前記ロードストアキュー内の第1のストアリクエストと第2のストアリクエストについて、当該第2のストアリクエストのアドレスと前記第1のストアリクエストのアドレスを比較し、当該第2のストアリクエストのアドレスが、前記第1のストアリクエストのアドレスと同一である場合には、当該第2のストアリクエストを前記第1のストアリクエストにマージする

ことを特徴とする請求項11又は12記載のロードストアキューの制御システム。

【請求項18】

前記主記憶部をDRAMで構成する

30

ことを特徴とする請求項11乃至17いずれか1項記載のロードストアキューの制御システム。

【請求項19】

前記主記憶部をシンクロナスDRAMで構成する

ことを特徴とする請求項11乃至17いずれか1項記載のロードストアキューの制御システム。

【請求項20】

前記主記憶部をDRAM又はシンクロナスDRAMを用いたDIMM若しくはSIMMで構成する

ことを特徴とする請求項11乃至17いずれか1項記載のロードストアキューの制御システム。

40

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、主記憶部に対して発行するリクエストを保持するロードストアキューの制御方法に関し、特にキャッシュと主記憶部との間に設けられたロードストアキューの制御方法に関する。

【背景技術】

【0002】

近年、プロセッサからキャッシュへとロード・ストアリクエストを発行する際、或いは

50

キャッシュから主記憶部へとロード・ストアリクエストを発行する際に、アクセスレイテンシ、データ転送性能差を隠蔽するためのロードストアキューが実装されてきた。ロードストアキューは、プロセッサとキャッシュ、或いは、キャッシュと主記憶部の中間位置に設けられ、その制御方法についてこれまでも多くの工夫がなされてきた。

【 0 0 0 3 】

ロードストアキューに関するアクセスレイテンシ、データ転送性能を向上させる技術としては、例えば以下のような技術が良く知られている。

(1) ストアキュー内に発行待ちとなっているリクエストと同一アドレスの後続ロードリクエストが存在する場合には、キャッシュ若しくは主記憶部へとロードアクセスリクエストを発行する代わりに、発行待ちとなっているストアキューのデータをロードアクセス結果としてリプライすることにより、短アクセスタイム化を計っている。

(2) また、処理時間のかかるロードリクエストを、先行するストアリクエストより先に発行するなどの工夫がなされている。

(3) さらに、先行ストアリクエストと同一アドレスの後続リクエストが存在する場合には、ストアデータの置き換え、或いはマージを行い、ストアリクエストの圧縮を行っている。そして、これらの機能の高速化手法も提案されている。

【 0 0 0 4 】

また、プロセッサとキャッシュとの間に実装されるロードストアキューに関する技術として、特許文献 1 では、ストアリクエストが発生したにも関わらずそのストアデータが揃わない場合に、先行ストアリクエストと後続ロードリクエストが同一アドレスでない場合、ロードストアバッファにおいて順番を入れ替え、ロードリクエストを先に発行する主記憶アクセス制御方式を提供している。即ち、特許文献 1 に開示される主記憶アクセス制御方式は、ストアリクエストが別要因で遅れてしまい、アドレスが一致しないロードリクエストを優先して主記憶部へと発行するものである。特許文献 2 では、同一アドレスのストアリクエストをマージするキャッシュメモリアccessシステムが提案されている。特許文献 3 及び 4 では、同一アドレスを有するストアリクエスト後のロードリクエストについて、その高速化手法が提案されている。

【特許文献 1】特開平 0 6 - 1 3 1 2 3 9 号公報

【特許文献 2】特開平 0 1 - 0 5 0 1 3 9 号公報

【特許文献 3】特開 2 0 0 0 - 2 5 9 4 1 2 号公報

【特許文献 4】特開 2 0 0 2 - 2 8 7 9 5 9 号公報

【発明の開示】

【発明が解決しようとする課題】

【 0 0 0 5 】

しかしながら、上述したいずれの技術もプロセッサとキャッシュとの間に実装されるロードストアキューに関するものであり、主記憶部である D R A M、シンクロナス D R A M 及びそれらを用いた D I M M や S I M M の特性を利用することで、アクセスレイテンシ、データ転送に関して性能改善及び低消費電力化を実現する内容ではない。

【 0 0 0 6 】

プロセッサとキャッシュとの間に実装されるロードストアキューに関しては、プロセッサから発行されるリクエストは主記憶部に対して直接アクセスを伴うものではないため、主記憶部におけるリードサイクルとライトサイクル間のバス切替サイクルについて問題とはならない。一方、キャッシュと主記憶部との間に実装されるロードストアキューに関しては、主記憶部である D R A M、シンクロナス D R A M が双方向バスである場合には、リードサイクルとライトサイクルの間にはバス切替サイクルを必要とする。このため、ロードリクエストからストアリクエスト、或いはストアリクエストからロードリクエストへの切替に際しては、バス競合回避のために、1 サイクル以上の空きサイクルを設けなければならない。例えば、リクエストが、ロードリクエスト、ストアリクエスト、ロードリクエスト、ストアリクエストという順番で交互に発行される場合には、空きサイクルが毎サイクル発生してしまうなど、空きサイクルの発生により、アクセスレイテンシ、データ転送

が低下するという課題がある。

【 0 0 0 7 】

本発明の目的は、上記知見に基づいてなされたものであって、キャッシュと主記憶部との間に実装されるロードストアキューについて、リードサイクル及びライトサイクル間のバス切替における空きサイクルの発生を抑制して、主記憶部に対して効率的にリクエストを発行することで、アクセスレイテンシ、データ転送に関して性能改善及び低消費電力化を実現するロードストアキューの制御方法及び制御システムを提供することにある。

【課題を解決するための手段】

【 0 0 0 8 】

本発明に係るロードストアキューの制御方法の一態様は、主記憶部に対して発行するリクエストを保持するロードストアキューの制御方法であって、前記ロードストアキュー内における前記リクエストの順序を並び替え、連続したストアリクエストと連続したロードリクエストとなるように制御する。

10

【 0 0 0 9 】

また、本発明に係るロードストアキューの制御システムの一態様は、主記憶部に対して発行するリクエストを保持するロードストアキューと、当該ロードストアキューを制御する制御部と、を備えたロードストアキューの制御システムであって、前記制御部は、前記ロードストアキュー内における前記リクエストの順序を並び替え、連続したストアリクエストと連続したロードリクエストとなるように制御する。

【発明の効果】

20

【 0 0 1 0 】

本発明によれば、キャッシュと主記憶部との間に実装されるロードストアキューについて、主記憶部に対して効率的にリクエストを発行することで、アクセスレイテンシ、データ転送に関して性能改善及び低消費電力化を実現するロードストアキューの制御方法及び制御システムを提供することができる。

【発明を実施するための最良の形態】

【 0 0 1 1 】

以下、本発明を実施するための最良の形態について、図面を参照しながら詳細に説明する。説明の明確化のため、以下の記載及び図面は、適宜、省略及び簡潔化がなされている。各図面において同一の構成又は機能を有する構成要素及び相当部分には、同一の符号を付し、その説明を省略する。

30

【 0 0 1 2 】

図1は、本発明の各実施形態のロードストアキュー制御方法の共通した全体概念図である。ロードストアキュー10は、キャッシュ20と主記憶部30の間に実装されている。ロードストアキュー10は主記憶部30に対して発行するリクエストを保持する。尚、ロードストアキュー10は主記憶部30に対してリクエストを直接発行するロードストアキューであればよく、ロードストアキュー10に対してリクエストを発行する装置はキャッシュ20に限定されない。ロードストアキューは、図示しないプロセッサからリクエストを直接発行されるロードストアキューであってもよい。

【 0 0 1 3 】

40

キャッシュ20はロードストアキュー10に対してリクエスト50を新たに発行する。リクエスト50は、ロードリクエスト又はストアリクエストのいずれのリクエストであるかを示すリクエスト種類情報(LD/ST41)と、リクエストで使用するデータを指定するアドレス42と、主記憶部30に格納するストアデータ48と、を含む。

【 0 0 1 4 】

ロードストアキュー10は、主記憶部30に対するリクエストを実際に発行するリクエストキュー11と、ストアデータ48を保持するストアデータキュー12と、ロードリクエストに対するリプライ情報(LDリクエストリプライ情報49)を保持するリプライキュー13と、を含む。尚、図示は省略するが、ロードストアキュー10は、ロードデータを保持するロードキューを更に備えていてもよい。

50

【 0 0 1 5 】

ロードストアキューの制御方法は、ランダムに発生するロードリクエスト及びストアリクエストを、ロードストアキュー 10 内でそのリクエスト順序を並び替え、連続したロードリクエストと連続したストアリクエストとなるように制御する。このため、キャッシュ 20 からロードストアキュー 10 へと新たに発行されたリクエスト 50 に対して制御情報 43 を付与し、ロードストアキュー 10 内におけるリクエストの順序を並び替える。

【 0 0 1 6 】

ロードストアキュー 10 内のキューについて、主記憶部 30 により近いキューを上位キューとし、ロードストアキュー 10 に新たに発行されたリクエストは、ロードストアキュー 10 内において上位キューへと移動する。

10

【 0 0 1 7 】

ロードストアキューの制御方法は、ロードストアキュー 10 内でその順序を並び替えたリクエストを、主記憶部 30 に対して発行する。リクエストがストアリクエストである場合には、ストアデータ 48 を主記憶部 30 に転送し、指定されたアドレス 42 に格納する。リクエストがロードリクエストである場合には、主記憶部 30 からロードストアキュー 10 に対して、ロードデータとロードリクエストに関する LD リクエストリプライ情報 49 が転送される。リクエストキュー 11 は、同一アドレスのロードがあった場合、1 つのリクエストに圧縮して主記憶部 30 にリクエストされるが、LD リクエストリプライ情報 49 は、圧縮前のロードリクエスト情報を記憶している。主記憶部 30 からのロードデータは、LD リクエストリプライ情報と照合され、キャッシュ 20 からのロードリクエスト毎にリプライを返す。

20

【 0 0 1 8 】

尚、ロードストアキュー 10 のリクエストキュー 11 及びリプライキュー 13 は、例えばフリップフロップ (FF) で構成することができ、ストアデータキュー 12 はランダムアクセスメモリ (RAM) で構成することができる。主記憶部 30 は、DRAM やシンクロナス DRAM (SDRAM) で構成され、これら DRAM を用いた DIMM や SIMM により構成してもよい。

【 0 0 1 9 】

図 2 はロードストアキュー内におけるリクエストキューの制御内容を説明するための図である。リクエストキュー 11 内に保持されたリクエストの制御情報 43 は、リクエストの有効性を示す有効情報 (V44) と、ストア wait カウント (STwait46) と、ストア wait valid (STwaitV45) と、近接アドレスフラグコード 47 と、を含む。

30

【 0 0 2 0 】

STwait46 及び STwaitV45 は、所定の条件を満足するまでの間、ロードストアキュー 10 内のストアリクエストをロードストアキュー 10 内に保持するように制御するための制御情報である。例えば所定の条件としては、ストアリクエストに後続するリクエストの個数に応じて制御することができる。

【 0 0 2 1 】

より具体的には、ストアリクエストに後続して発行されるリクエストの個数を STwait46 を用いてカウントし、当該カウント値が規定回数となった場合には、STwaitV45 を規定の値に設定する。STwaitV45 が規定の値となった場合には、そのストアリクエストを主記憶部 30 に対して発行するように制御することができる。一方、STwaitV45 が規定の値となるまでの間は、そのストアリクエストを主記憶部 30 に対して発行せずに、ロードストアキュー 10 内に保持するように制御することができる。

40

【 0 0 2 2 】

このように、ロードストアキュー 10 に対して新たに発行されるリクエストの個数が所定の個数となるまでの間、ロードストアキュー 10 内にストアリクエストを保持することで、先行するストアリクエストと、同じアドレスの後続ストアリクエストがあった場合、

50

ロードストアキュー 10 内により多くのストアリクエストを滞留させることができるため、効率よくストアデータをマージすると共に、ストアリクエスト毎に保持する時間を変えることにより、ストアリクエストとロードリクエストを分離して主記憶部 30 に発行することができる。

【 0 0 2 3 】

近接アドレスフラグコード 47 は、主記憶部 30 において予め定めたアドレスの処理単位に基づいて、ロードストアキュー 10 内のリクエストを分類するための制御情報である。主記憶部 30 におけるアドレスは予め複数の処理単位に分割され、近接アドレスフラグコード 47 は、それら処理単位のいずれかを示す識別情報である。ロードストアキュー 10 内のリクエストに近接アドレスフラグコード 47 を付与することで、リクエストをアドレスに応じて分類して管理することができる。例えば、同一のロードアドレスを有するリクエストに対して同一の近接アドレスフラグコード 47 を付与し、同一のランクアドレスを有するリクエストに同一の近接アドレスフラグコード 47 を付与することができる。

10

【 0 0 2 4 】

より具体的には、ロードストアキューの制御方法は、ロードストアキュー 10 内のリクエストと新たに発行されるリクエストをそのアドレスについて互いに比較し、リクエストを近接アドレスフラグコード 47 に応じて分類する。そして、ロードストアキュー 10 内のリクエストを主記憶部 30 へと発行する際には、リクエストのうち同一の近接アドレスフラグコード 47 を有するリクエストをまとめて連続して主記憶部 30 へと発行する。例えばメモリリクエスト選択ユニット (M R S U 1 5) によって、ロードリクエストを主記憶部 30 へと発行する場合には、同一の近接フラグコード 47 を有するロードストアキュー 10 内のロードリクエストを選択した上で、それらロードリクエストを全て連続して発行するようにマルチプレクサを制御することができる。

20

【 0 0 2 5 】

また、ストアリクエストを主記憶部 30 へと発行する場合には、 S T w a i t V 4 5 が規定の値となったストアリクエストを主記憶部 30 へとまず発行した後、同一の近接アドレスフラグコード 47 を有するストアリクエストを連続して発行するように制御することができる。尚、ストアリクエストによっては、 S T w a i t V 4 5 が規定の値となる前に発行するように制御してもよいし、 S T w a i t V 4 5 が規定の値となったストアリクエストから常に選択して発行するように制御することもできる。

30

【 0 0 2 6 】

さらにまた、ロードストアキューの制御方法は、同一アドレスリクエスト制御部 14 によって、同一のアドレスを有するリクエストについて下記 (i) 乃至 (i i i) で説明する制御を実行することができる。

【 0 0 2 7 】

(i) ロードストアキュー 10 内において、先行ロードリクエストと後続ロードリクエストのアドレスが同一であった場合には、主記憶部 30 に対するロードリクエストを 1 つにする。より具体的には、新規に発行されたロードリクエストのアドレスとロードストアキュー 10 内の全てのロードリクエストのアドレスとを比較し、同一のアドレスを有するロードリクエストが存在する場合には、リクエストキュー 11 には 1 つのロードリクエストのみをキューイングし、リブライキュー 13 には同一のアドレスを有する全てのロードリクエストの個数分の L D リクエストリブライ情報 49 がキューイングされる。尚、本制御機能は一般的にキャッシュ 20 内に実装されることもあるが、ロードストアキュー 10 のリクエスト元がキャッシュ 20 ではない場合には、ロードストアキュー 10 に実装してもよい。

40

【 0 0 2 8 】

(i i) 先行ストアリクエストと後続ストアリクエストのアドレスが同一であった場合には、それらストアリクエストのストアデータをマージする。より具体的には、ロードストアキュー 10 内の全てのストアリクエストについて、先行するストアリクエストと後続ストアリクエストのアドレスを比較し、同一のアドレスを有するストアリクエストが存在

50

する場合には、後続ストアリクエストのストアデータ48を先行ストアリクエストのストアデータ48にマージする。この場合、リクエストキュー11には、1つのストアリクエストのみがキューイングされ、ストアデータキュー12には、マージされた1つのストアデータ48が保持される。

【0029】

(iii) 先行ストアリクエストと後続ロードリクエストのアドレスが同一であった場合には、ストアデータキュー12に保持されたストアデータ48の内容を後続ロードリクエストのデータとしてリプライする。より具体的には、新規に発行されたロードリクエストのアドレスとロードストアキュー10内の全てのストアリクエストのアドレスとを比較し、同一のアドレスを有するストアリクエストが存在する場合には、ロード結果として、ストアデータキュー12に保持されたストアデータ48の内容をキャッシュにリプライする。

10

【0030】

実施の形態1.

実施の形態1では、ロードストアキューの制御方法は、ロードストアキュー10内におけるリクエストの順序を並び替え、連続したストアリクエストと連続したロードリクエストとなるように制御する。図3は、本実施形態のロードストアキューの制御方法が、ロードストアキュー内のリクエストを制御する手順の一例を示すフローチャートである。図3に示すようにステップS103乃至S107において、新規発行されたロードリクエストの順序を制御する。また、ステップS108乃至S117において、新規発行されたストアリクエストをロードストアキュー10内に待機させると共に、その順序を制御する。以下、図3を参照しながらロードストアキューの制御方法について詳細に説明する。尚、ロードリクエストと同一アドレスの先行ストアリクエストがリクエストキュー上にあった場合で、ロードデータの一部のデータがストアデータとして存在しないような場合に、当該ストアリクエストのストアデータをロードリクエストのデータとしてキャッシュにリプライできないとき、ストアリクエストは規定回数待つことなしに、ロードリクエストに先立ってストアリクエストを主記憶部に発行しなければならないが、当該制御は、本発明の本質ではないため、記述していない。また、実施の形態1では、リクエストキューで、ストアリクエストの直上位キューにあるのがストアリクエストであった場合で、当該直上位キューが主記憶部に発行された場合には、ストアリクエストの発行待ちをせずに主記憶部へ発行する制御の場合を示している。

20

30

【0031】

まず、ロードストアキューの制御方法は、新規発行されたリクエストの有効情報V44と、STwaitV45と、STwait46の値を初期化する(V=1, STwaitV=0, STwait=0)(ステップS101)。次いで、リクエスト種類情報(LD/ST41)より、リクエストがロードリクエストであるか否かを判定する(ステップS102)。

【0032】

ステップS102における判定の結果、リクエストがロードリクエストでない場合(即ちストアリクエストである場合)には、ステップS108へと進む。一方、リクエストがロードリクエストである場合には、ロードストアキュー10内の上位キューに有効なリクエスト(ここでは、V=1のリクエスト)が存在するか否かを判定する(ステップS103)。即ち、ロードストアキュー10内に先行する有効なリクエストが存在するか否かを確認する。

40

【0033】

ステップS103における判定の結果、上位キューに有効なリクエストが存在しない場合には、ステップS118へと進む。一方、上位キューに有効なリクエストが存在する場合には、リクエストが最上位キューに位置するものであるか否かを更に判定する(ステップS104)。即ち、リクエストが主記憶部30に対して次に発行対象となるリクエストであるか否かを確認する。

50

【 0 0 3 4 】

ステップ S 1 0 4 における判定の結果、リクエストが最上位キューに位置するものである場合にはステップ S 1 1 8 へと進む。一方、リクエストが最上位キューに位置するものではない場合には、上位キューのリクエストが全てストアリクエストであるか否かを更に判定する（ステップ S 1 0 5）。即ち、ロードリクエストに対して先行するリクエストが、全てストアリクエストであるか否かを確認する。

【 0 0 3 5 】

ステップ S 1 0 5 における判定の結果、上位キューのリクエストが全てストアリクエストである場合にはステップ S 1 1 8 へと進む。一方、上位キューのリクエストが全てストアリクエストでない場合には、直上キューのリクエストが有効なリクエスト（ここでは、 $V = 1$ のリクエスト）であるか否かを更に判定する（ステップ S 1 0 6）。即ち、ロードストアキュー 1 0 内において、直前に先行する有効なリクエストが存在するか否かを確認する。

10

【 0 0 3 6 】

ステップ S 1 0 6 における判定の結果、直上キューのリクエストが有効なリクエストである場合には再びステップ S 1 0 6 へと戻る。即ち、ロードストアキュー 1 0 内において、直前に先行する有効リクエストが存在する場合には、その先行するリクエストが無効となるまでの間、待機する。一方、直上キューのリクエストが有効なリクエストでない場合には、リクエストを上位キューへと移動して、ステップ S 1 0 3 へと戻る（ステップ S 1 0 7）。

20

【 0 0 3 7 】

他方、ステップ S 1 0 2 における判定の結果、リクエストがロードリクエストでない（即ちストアリクエストである）場合には、直上位のリクエストがストアリクエストであるか否かを更に判定する（ステップ S 1 0 8）。即ち、直前に先行するリクエストがストアリクエストであるか否かを確認する。

【 0 0 3 8 】

ステップ S 1 0 8 における判定の結果、直上位のリクエストがストアリクエストでない場合にはステップ S 1 1 0 へと進む。一方、直上位のリクエストがストアリクエストである場合には、直上位のリクエストが主記憶部 3 0 に対して発行されたか否かを更に判定する（ステップ S 1 0 9）。即ち、直前に先行するストアリクエストが発行済みであるか否かを確認する。

30

【 0 0 3 9 】

ステップ S 1 0 9 における判定の結果、直上位のリクエストが主記憶部 3 0 に対して発行された場合にはステップ S 1 1 8 へと進む。一方、直上位のリクエストが主記憶部 3 0 に対して発行されていない場合には、ロードストアキュー 1 0 に対して新たなリクエストが発行されたか否かを更に判定する（ステップ S 1 1 0）。即ち、ストアリクエストに対して後続するリクエストが発行されたか否かを確認する。

【 0 0 4 0 】

ステップ S 1 1 0 における判定の結果、新たなリクエストが発行されていない場合には、ステップ S 1 1 0 へと戻る。即ち、ロードストアキュー 1 0 に対して後続するリクエストが発行されるまでの間待機する。一方、新たなリクエストが発行された場合には、 $STwait46$ の値をインクリメントする（ $STwait = + 1$ ）（ステップ S 1 1 1）。即ち、後続するリクエストの個数をカウントする。

40

【 0 0 4 1 】

次いで、 $STwait46$ の値に基づいて、ストアリクエストがロードストアキュー 1 0 内において規定回数待ったか否かを判定する（ステップ S 1 1 2）。即ち、ストアリクエストが発行可能状態であるか否かを確認する。

【 0 0 4 2 】

ステップ S 1 1 2 における判定の結果、規定回数待っていない場合には、ステップ S 1 1 0 へと戻る。即ち、後続するリクエストの個数をカウントし、そのカウント値が規定の

50

値となるまでの間、ロードストアキュー 10 内にストアリクエストを保持する。一方、規定回数待った場合には、S T w a i t V 4 5 の値を有効値へと変更する (S T w a i t V = 1) (ステップ S 1 1 3)。即ち、ストアリクエストを発行可能状態とする。

【 0 0 4 3 】

次いで、ロードストアキュー 10 内の上位キューに有効なリクエスト (ここでは、V = 1 のリクエスト) が存在するか否かを判定する (ステップ S 1 1 4)。判定の結果、上位キューに有効なリクエストが存在しない場合にはステップ S 1 1 8 へと進む。一方、上位キューに有効なリクエストが存在する場合には、リクエストが最上位キューに位置するものであるか否かを更に判定する (ステップ S 1 1 5)。

【 0 0 4 4 】

ステップ S 1 1 5 における判定の結果、リクエストが最上位キューに位置するものである場合にはステップ S 1 1 8 へと進む。一方、リクエストが最上位キューに位置するものでない場合には、直上キューのリクエストが有効なリクエスト (ここでは、V = 1 のリクエスト) であるか否かを更に判定する (ステップ S 1 1 6)。

【 0 0 4 5 】

ステップ S 1 1 6 における判定の結果、直上キューのリクエストが有効なリクエストである場合には再びステップ S 1 1 6 へと戻る。即ち、ロードストアキュー 10 内において、直前に先行する有効リクエストが存在する場合には、その先行するリクエストが無効となるまでの間、待機する。一方、直上キューのリクエストが有効なリクエストでない場合には、リクエストを上位キューへと移動して、ステップ S 1 0 8 へと戻る (ステップ S 1 1 7)。

【 0 0 4 6 】

他方、ステップ S 1 0 3、S 1 0 4、S 1 0 5、S 1 0 9、S 1 1 4、S 1 1 5 の判定の結果、リクエストが主記憶部 30 へ発行対象となった場合には、リクエストの有効情報 V 4 4 の値をクリアし (V = 0) (ステップ S 1 1 8)、主記憶部 30 に対してロードストアキュー 10 からリクエストを発行し (ステップ S 1 1 9)、リクエストキューからエントリを解除する。

【 0 0 4 7 】

このように、後続リクエスト個数が所定の個数となるまでの間、ロードストアキュー 10 内のストアリクエストを主記憶部 30 に対して発行せずに保持すると共に、そのストアリクエストの後続ロードリクエストの順序を先に並び替えることで、ロードストアキュー 10 内のストアリクエストを連続して保持することができる。このため、ロードストアキュー 10 から主記憶部 30 に対してリクエストを発行する際には、ストアリクエストを連続して発行することができると共に、ストアリクエスト間のロードリクエストについても連続して発行することができる。従って、リードサイクル及びライトサイクル間のバス切替における空きサイクルの発生を抑制して、主記憶部 30 に対して効率的にリクエストを発行することができ、アクセスレイテンシ、データ転送に関して性能改善及び低消費電力化を実現することができる。

【 0 0 4 8 】

実施の形態 2 .

次に、本発明の実施の形態 2 について図面を参照して詳細に説明する。実施の形態 2 では、ロードストアキューの制御方法は、ロードストアキュー 10 内のリクエストのアドレスが主記憶部 30 において同一の処理単位に含まれるか否かを判定し、主記憶部 30 へとリクエストを発行する際には、同一の処理単位に含まれるアドレスを有するリクエストをまとめて連続して発行するよう制御する。図 4 は、本実施形態のロードストアキューの制御方法が、ロードストアキュー内のリクエストを制御する手順の一例を示すフローチャートである。以下、図 4 を参照しながらロードストアキューの制御方法について詳細に説明する。

【 0 0 4 9 】

まず、ロードストアキューの制御方法は、新規発行されたリクエストの有効情報 V 4 4

10

20

30

40

50

の値を初期化する ($V = 1$) (ステップ S 2 0 1)。次いで、ロードストアキュー 1 0 内の上位キューに存在する全てのリクエストのアドレスと、新規発行されたリクエストのアドレスとを比較して検索を行う (ステップ S 2 0 2)。

【 0 0 5 0 】

ステップ S 2 0 2 における検索の結果、ロードストアキュー 1 0 内の上位キューに、新規発行リクエストのアドレスと同一のローアドレス或いは同一のランクアドレスを有するリクエストが存在するか否かを判定する (ステップ S 2 0 3)。即ち、新規発行リクエストに先行するリクエストであって、新規リクエストと同一のローアドレス或いは同一のランクアドレスを有しており、主記憶部 3 0 に発行する際に、新規発行リクエストと同時に発行可能なリクエストがロードストアキュー 1 0 内に既に存在しているか否かを確認する。

10

【 0 0 5 1 】

ステップ S 2 0 3 における判定の結果、新規発行リクエストのアドレスと同一のローアドレス或いは同一のランクアドレスを有するリクエストが既に存在している場合には、ロードストアキュー 1 0 内のそのリクエストに付与された近接アドレスフラグコード 4 7 を新規発行リクエストに付与する (ステップ S 2 1 0)。一方、新規発行リクエストのアドレスと同一のローアドレス或いは同一のランクアドレスを有するリクエストが存在しない場合には、新たな近接アドレスフラグコード 4 7 を生成して、その新規発行リクエストに付与する (ステップ S 2 0 4)。

【 0 0 5 2 】

20

次いで、ロードストアキュー 1 0 内の上位キューに有効なリクエスト (ここでは、 $V = 1$ のリクエスト) が存在するか否かを判定する (ステップ S 2 0 5)。判定の結果、上位キューに有効なリクエストが存在しない場合には、ステップ S 2 1 1 へと進む。一方、上位キューに有効なリクエストが存在する場合には、リクエストが最上位キューに位置するものであるか否かを更に判定する (ステップ S 2 0 6)。

【 0 0 5 3 】

ステップ S 2 0 6 における判定の結果、リクエストが最上位キューに位置するものである場合にはステップ S 2 1 1 へと進む。一方、リクエストが最上位キューに位置するものではない場合には、同一の近接アドレスフラグコード 4 7 を有するリクエストが、主記憶部 3 0 に対して発行されたか否かを更に判定する (ステップ S 2 0 7)。即ち、同一の近接アドレスフラグコード 4 7 を有するリクエストをまとめて連続して発行するか否かを確認する。

30

【 0 0 5 4 】

ステップ S 2 0 7 における判定の結果、同一の近接アドレスフラグコード 4 7 を有するリクエストが、主記憶部 3 0 に対して発行された場合にはステップ S 2 1 1 へと進む。一方、同一の近接アドレスフラグコード 4 7 を有するリクエストが、主記憶部 3 0 に対して発行されていない場合には、直上キューのリクエストが有効なリクエスト (ここでは、 $V = 1$ のリクエスト) であるか否かを更に判定する (ステップ S 2 0 8)。

【 0 0 5 5 】

ステップ S 2 0 8 における判定の結果、直上キューのリクエストが有効なリクエストである場合には再びステップ S 2 0 8 へと戻る。一方、直上キューのリクエストが有効なリクエストでない場合には、リクエストを上位キューへと移動して、ステップ S 2 0 5 へと戻る (ステップ S 2 0 9)。

40

【 0 0 5 6 】

他方、ステップ S 2 0 5、S 2 0 6、S 2 0 7 の判定の結果、リクエストが主記憶部 3 0 への発行対象となった場合には、リクエストの有効情報 V 4 4 の値をクリアし ($V = 0$) (ステップ S 2 1 1)、ロードストアキュー 1 0 から主記憶部 3 0 に対してリクエストを発行する (ステップ S 2 1 2)。

【 0 0 5 7 】

このように、ロードストアキュー 1 0 内のリクエストについて、そのアドレスが同一の

50

ロードアドレス或いは同一のランクアドレスである場合には、同一の近接アドレスフラグコード47を有するリクエストを併せて主記憶部30へと発行することで、同一のロードアドレス或いは同一のランクアドレスにより主記憶部30に対して連続してアクセスすることができる。このため、ロードストアキュー10から主記憶部30に対してリクエストを発行する際には、同一のロードアドレスの転送について、RASを1回のみアクティベートすれば良く、RASのアクティベート実行回数を削減することができる。また、DIMM等が主記憶部30に使用されており、同一ランクアドレスアクセスが連続する場合の方が、異ランクアドレスアクセスが連続する場合より、高速にアクセスできる場合は、同一ランクアドレスアクセスを連続させることができ、主記憶部30の処理を高速化できる。

【0058】

実施の形態3.

上述した実施の形態1では、ロードストアキューの制御方法は、所定の条件が満足されるまでの間、ロードストアキュー10内にストアリクエストを保持することで、連続したストアリクエストと連続したロードリクエストとなるように制御する。また、実施の形態2では、ロードストアキューの制御方法は、ロードストアキュー10内のリクエストのアドレスが主記憶部30において同一の処理単位に含まれるか否かを判定し、主記憶部30へとリクエストを発行する際には、同一の処理単位に含まれるアドレスを有するリクエストをまとめて連続して発行するように制御する。実施の形態3では、ロードストアキューの制御方法は、ロードストアキュー内におけるリクエストの順序を並び替え、連続したストアリクエストと連続したロードリクエストとなるように制御すると共に、リクエストを主記憶部30へと発行する際には、ロードストアキュー10内のリクエストのアドレスが主記憶部30において同一の処理単位に含まれるアドレスである場合には、同一の処理単位に含まれるアドレスを有するリクエストを併せて発行するように制御する。

【0059】

より具体的には、まず、図3に示したフローチャートにおいて説明したように、後続リクエスト個数が所定の個数となるまでの間、ロードストアキュー10内のストアリクエストを主記憶部30に対して発行せずに保持すると共に、そのストアリクエストの後続ロードリクエストの順序を先に並び替える。そして、図4に示したフローチャートにおいて説明したように、ロードストアキュー10内のリクエストについて、そのアドレスが同一のロードアドレス或いは同一のランクアドレスである場合には、同一の近接アドレスフラグコード47を有するリクエストを併せて主記憶部30へと発行する。ロードストアキューの制御方法は、例えばS T w a i t 4 6及びS T w a i t V 4 5を用いることで、所定の条件を満足するまでの間、ストアリクエストをロードストアキュー10内に保持することができ、例えば近接アドレスフラグコード47を用いることで、主記憶部30において同一のアドレスの処理単位に基づいて、ロードストアキュー10内のリクエストを分類して管理することができる。

【0060】

このようにすると、ロードストアキュー10から主記憶部30に対してリクエストを発行する際には、連続したロードリクエストと連続したストアリクエストをより効率的に発行することができるため、アクセスレイテンシ、データ転送に関して性能改善及び低消費電力化を実現することができる。

【0061】

尚、ストアリクエストを主記憶部30へと発行する場合には、S T w a i t V 4 5が規定の値となったストアリクエストを主記憶部30へと先に発行した後、同一の近接アドレスフラグコード47を有するストアリクエストを連続して発行するよう制御し、S T w a i t V 4 5が規定の値となる前に発行するように制御してもよい。

【0062】

また、ロードストアキューの制御方法は、まず、図4に示したフローチャートにおいて説明したように、ロードストアキュー10内のリクエストについて、そのアドレスが同一のロードアドレス或いは同一のランクアドレスである場合には、同一の近接アドレスフラグ

10

20

30

40

50

コード47を有するリクエストが連続するようにリクエストの順序を並び替える。そして、図3に示したフローチャートにおいて説明したように、後続ストアリクエスト個数が所定の個数となった場合に、同一の近接アドレスフラグコード47を有するリクエストの中で、ロードリクエストとストアリクエストを分離して、主記憶部30へと発行するようにしてもよい。

【0063】

実施の形態4.

実施の形態4では、ロードストアキューの制御方法は、まず、図3に示したフローチャートにおいて説明したように、後続リクエスト個数が所定の個数となるまでの間、ロードストアキュー10内のストアリクエストを主記憶部30に対して発行せずに保持すると共に、そのストアリクエストの後続ロードリクエストの順序を先に並び替える。そして更に、ロードストアキュー10内の全てのストアリクエストについて、先行するストアリクエストと後続ストアリクエストのアドレスを比較し、同一のアドレスを有するストアリクエストが存在する場合には、後続するストアリクエストのストアデータ48を先行するストアリクエストのストアデータ48にマージして1つのストアリクエストとするように制御する。

10

【0064】

このように、ロードストアキュー10内のストアリクエストを所定の条件が満足されるまで発行しない構成とすることで、より多くのストアリクエストをロードストアキュー10内に滞留させることができる。従って、ロードストアキュー10に対して発行されるストアリクエストのマージ確率をより向上させることができ、主記憶部30に対して更に効率的にストアリクエストを発行することができる。

20

【0065】

実施の形態5.

実施の形態5では、ロードストアキューの制御方法は、まず、図3に示したフローチャートにおいて説明したように、後続リクエスト個数が所定の個数となるまでの間、ロードストアキュー10内のストアリクエストを主記憶部30に対して発行せずに保持すると共に、そのストアリクエストの後続ロードリクエストの順序を先に並び替える。そして更に、新規発行ロードリクエストのアドレスとロードストアキュー10内の全てのストアリクエストのアドレスとを比較し、同一のアドレスを有するストアリクエストが存在する場合には、ロードリクエストを主記憶部30に対して発行せずに、ロード結果として、ストアデータキュー12に保持されたストアデータ48の内容をリプライする。

30

【0066】

このように、ストアリクエストの発行を遅延させることで、連続したロードリクエストと連続したストアリクエストを主記憶部30にリクエストすることができる。また、ロードストアキュー10内により多くのストアリクエストを滞留させることで、先行ストアリクエストと同一アドレスを有する後続ロードリクエストがロードストアキュー10内に存在する確率を向上させることができ、主記憶部30に対して更に効率的にリクエストを発行することができる。

【0067】

実施の形態6.

実施の形態6では、ロードストアキューの制御方法は、まず、図3に示したフローチャートにおいて説明したように、後続リクエスト個数が所定の個数となるまでの間、ロードストアキュー10内のストアリクエストを主記憶部30に対して発行せずに保持すると共に、そのストアリクエストの後続ロードリクエストの順序を先に並び替える。そして更に、新規発行されたロードリクエストのアドレスとロードストアキュー10内の全てのロードリクエストのアドレスとを比較し、同一のアドレスを有するロードリクエストが存在する場合には、リクエストキュー11には1つのロードリクエストのみをキューイングするように制御する。

40

【0068】

50

このように、ストアリクエストの発行を遅延させることで、連続したロードリクエストと連続したストアリクエストを主記憶部 30 にリクエストすることができる。更に、同一のアドレスを有するロードリクエストを 1 つにすることで、主記憶部 30 に対してより効率的にリクエストを発行することができる。

【 0 0 6 9 】

実施の形態 7 .

実施の形態 7 では、ロードストアキューの制御システムは、ロードストアキュー 10 内におけるリクエストの順序を並び替え、連続したストアリクエストと連続したロードリクエストとなるように制御する。図 5 は、実施の形態 7 のロードストアキュー制御システムの機能ブロック図である。ロードストアキューの制御システム 100 は、主記憶部 30 に対して発行するリクエストを保持するロードストアキュー 10 と、ロードストアキュー 10 を制御する制御部 110 と、を備える。

10

【 0 0 7 0 】

制御部 110 は、ロードストアキュー 10 内における前記リクエストの順序を並び替え、連続したストアリクエストと連続したロードリクエストとなるように制御する。制御部 110 は、ストアリクエスト制御部 120 と、ロードリクエスト制御部 130 と、リクエスト判定部 140 と、アドレス判定部 150 と、を備える。ストアリクエスト制御部 120 は、更にリクエスト計測部 121 を備える。

【 0 0 7 1 】

ストアリクエスト制御部 120 は、ロードストアキュー 10 内のリクエストのうちストアリクエストを所定の条件を満足するまでの間、ロードストアキュー 10 内に保持する。ストアリクエスト制御部 120 は、例えば、ロードストアキュー 10 に対して新たに発行されるリクエストの個数が所定の個数となるまでの間ロードストアキュー 10 内にストアリクエストを保持する。より具体的には、ストアリクエスト制御部 120 は、リクエスト計測部 121 によりストアリクエストに後続して発行されるリクエストの個数をカウントし、カウントされたカウント値が所定の値となるまでの間、ロードストアキュー 10 内にストアリクエストを保持する。尚、ストアリクエスト制御部 120 は、ロードストアキュー 10 内のストアリクエストをロードストアキュー 10 内に所定の時間保持するように制御してもよい。

20

【 0 0 7 2 】

ロードストアリクエスト制御部 130 は、ロードストアキュー 10 内に保持されたストアリクエストに後続するロードリクエストの順序を、その保持されたストアリクエストよりも先に並び替える。

30

【 0 0 7 3 】

また、制御部 130 は、リクエスト判定部 140 により、ロードストアキューから主記憶部 30 に対して発行対象となるリクエストがストアリクエスト又はロードリクエストのいずれかであるかを判定し、リクエストがストアリクエストである場合には、ロードストアキュー 10 内にそのストアリクエストを保持する。

【 0 0 7 4 】

さらにまた、制御部 110 は、アドレス判定部 160 により、ロードストアキュー 10 内の第 1 のリクエストのアドレスと第 2 のリクエストのアドレスが主記憶部 30 において同一の処理単位に含まれるアドレスであるか否かを判定し、第 1 のリクエストのアドレスと第 2 のリクエストのアドレスがロードストアキュー 10 内の処理単位に含まれるアドレスである場合には、第 1 のリクエストを主記憶部 30 に発行する際に、その第 2 のリクエストを併せて主記憶部 30 へと発行する

40

【 0 0 7 5 】

その他の実施の形態 .

上述した実施の形態 1 乃至 6 においては、所定の条件が満足されるまでの間、ロードストアキュー 10 内にストアリクエストを保持するように制御する構成としたが、本発明はこれに限定されない。例えば、ロードストアキュー 10 内において、リクエストが主記憶

50

部 30 に対して発行対象となった場合に、そのリクエストがストアリクエスト又はロードリクエストのいずれかであることを判定し、当該判定の結果、リクエストがストアリクエストである場合には、ロードストアキュー 10 内に当該ストアリクエストを保持するように制御してもよい。また、上述した実施の形態においては、後続リクエストの個数に応じてストアリクエストを保持するように制御する構成としたが、ロードストアキュー 10 内に所定の時間ストアリクエストを保持するように制御してもよい。

【0076】

以上説明してきたように、本発明のロードストアキューの制御方法の一態様は、主記憶部 30 に対して発行するリクエストを保持するロードストアキュー 10 の制御方法であって、ロードストアキュー 10 内におけるリクエストの順序を並び替え、連続したストアリクエストと連続したロードリクエストとなるように制御する。

10

【0077】

このように、ロードストアキュー 10 内におけるリクエストの順序を並び替え、連続したストアリクエストと連続したロードリクエストとなるように制御することで、ロードストアキュー 10 内により多くのストアリクエストをまとめて滞留させることができる。このため、ロードストアキュー 10 から主記憶部 30 に対してリクエストを発行する際には、連続したストアリクエストと連続したロードリクエストを発行することができる。従って、リードサイクル及びライトサイクル間のバス切替における空きサイクルの発生を抑制して、主記憶部 30 に対して効率的にリクエストを発行することができ、アクセスレイテンシ、データ転送に関して性能改善及び低消費電力化を実現することができる。

20

【0078】

また、本発明のロードストアキューの制御システムの一態様は、主記憶部 30 に対して発行するリクエストを保持するロードストアキュー 10 と、ロードストアキュー 10 を制御する制御部 110 と、を備えたロードストアキューの制御システム 100 であって、制御部 110 は、ロードストアキュー 10 内におけるリクエストの順序を並び替え、連続したストアリクエストと連続したロードリクエストとなるように制御する。

【0079】

これにより、ロードストアキュー 10 から主記憶部 30 に対してリクエストを発行する際には、ストアリクエストを連続して発行することができると共に、ロードリクエストについても連続して発行することができる。従って、リードサイクル及びライトサイクル間のバス切替における空きサイクルの発生を抑制して、主記憶部 30 に対して効率的にリクエストを発行することができ、アクセスレイテンシ、データ転送に関して性能改善及び低消費電力化を実現することができる。

30

【0080】

尚、本発明は上述した実施の形態のみに限定されるものではなく、本発明の要旨を逸脱しない範囲において種々の変更が可能であることは勿論である。

【図面の簡単な説明】

【0081】

【図 1】本発明の実施の形態 1 のロードストアキュー制御方法を示す全体概念図である

【図 2】本発明の実施の形態 1 のリクエストキューの制御内容を説明するための図である

40

【図 3】本発明の実施の形態 1 のロードストアキュー制御方法による制御手順の一例を示すフローチャートである。

【図 4】本発明の実施の形態 2 のロードストアキュー制御方法による制御手順の一例を示すフローチャートである。

【図 5】本発明の実施の形態 7 のロードストアキュー制御システムの機能ブロック図である。

【符号の説明】

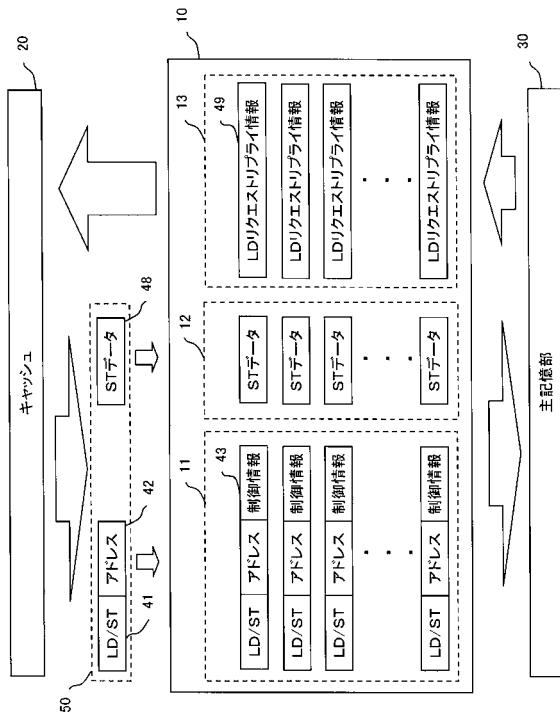
【0082】

10 ロードストアキュー、

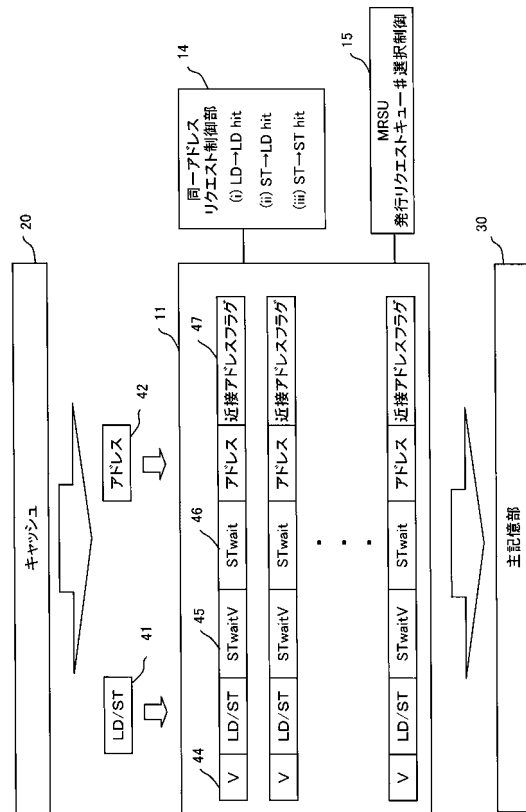
50

- 1 1 リクエストキュー、 1 2 ストアデータキュー、 1 3 リプライキュー、
- 1 4 同一アドレスリクエスト制御部、 1 5 MRSU、
- 2 0 キャッシュ、
- 3 0 主記憶部、
- 4 1 リクエスト種類情報 (LD / ST)、 4 2 アドレス、 4 3 制御情報、
- 4 4 有効情報、 4 5 STwaitV、 4 6 STwait、
- 4 7 近接アドレスフラグコード、 4 8 ストアデータ、
- 4 9 LDリクエストリプライ情報、
- 1 0 0 ロードストアキューの制御システム、
- 1 1 0 制御部、 1 2 0 ストアリクエスト制御部、 1 2 1 リクエスト計測部、
- 1 3 0 ロードリクエスト制御部、 1 4 0 リクエスト判定部、
- 1 5 0 アドレス判定部

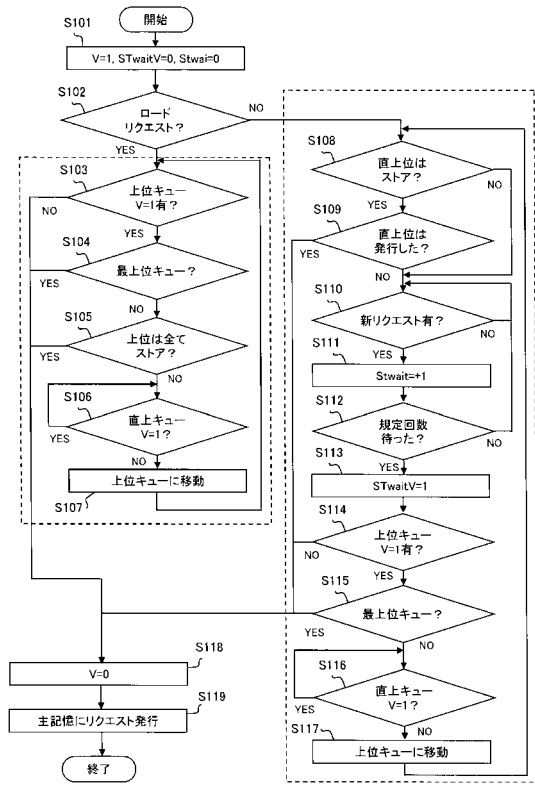
【図 1】



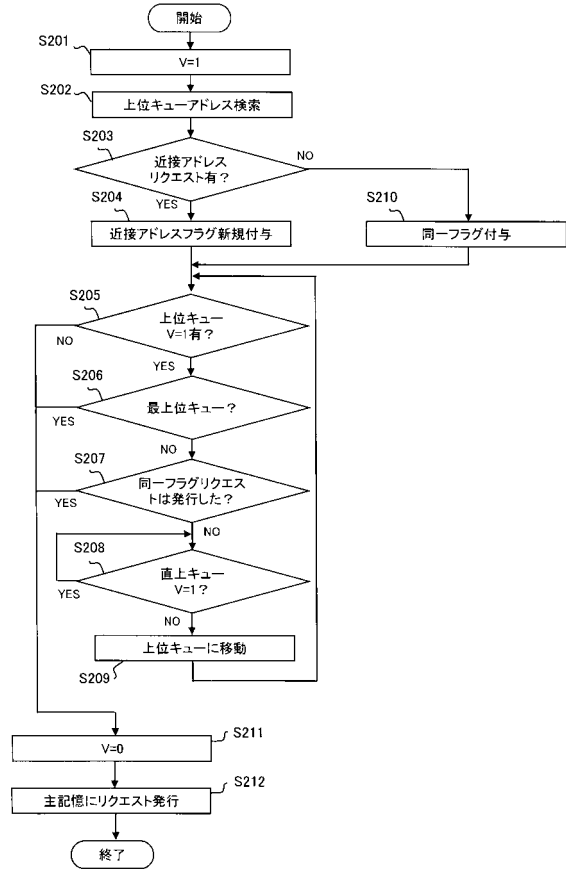
【図 2】



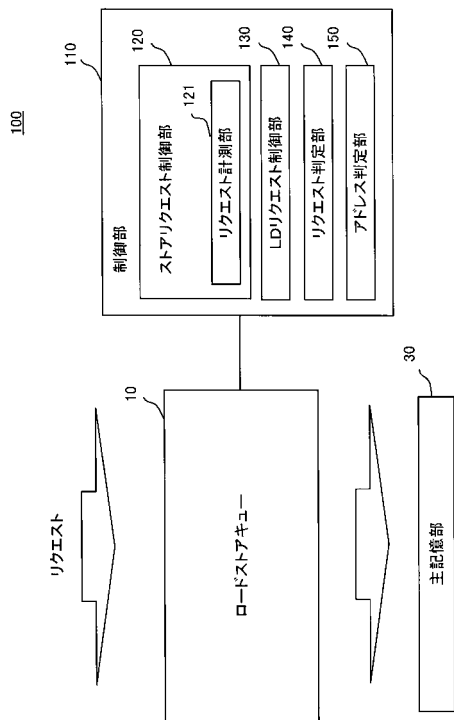
【図3】



【図4】



【図5】



フロントページの続き

審査官 多賀 実

- (56)参考文献 特開平10-228417(JP,A)
特開2003-271445(JP,A)
米国特許第6564304(US,B1)
特開2000-181780(JP,A)
特開昭58-208982(JP,A)
特開2001-222463(JP,A)
特表2005-505854(JP,A)
特開2004-252985(JP,A)
米国特許出願公開第2006/0248261(US,A1)

- (58)調査した分野(Int.Cl.,DB名)
G06F12/00-12/06
G06F13/16-13/18