

(19) 日本国特許庁 (JP)

(12) 特 許 公 報 (B2)

(11) 特許番号

特許第6127907号
(P6127907)

(45) 発行日 平成29年5月17日 (2017.5.17)

(24) 登録日 平成29年4月21日 (2017.4.21)

(51) Int.Cl.

F I

G O 6 F 12/08 (2016.01)

G O 6 F 12/10 (2016.01)

G O 6 F 12/08 5 1 7 B

G O 6 F 12/08 5 1 9 E

G O 6 F 12/08 5 2 3 C

G O 6 F 12/10 5 0 1 Z

G O 6 F 12/08 5 0 9 F

請求項の数 7 (全 14 頁)

(21) 出願番号 特願2013-220675 (P2013-220675)
 (22) 出願日 平成25年10月23日 (2013.10.23)
 (65) 公開番号 特開2014-112360 (P2014-112360A)
 (43) 公開日 平成26年6月19日 (2014.6.19)
 審査請求日 平成28年7月5日 (2016.7.5)
 (31) 優先権主張番号 特願2012-248661 (P2012-248661)
 (32) 優先日 平成24年11月12日 (2012.11.12)
 (33) 優先権主張国 日本国 (JP)

(73) 特許権者 000005223
 富士通株式会社
 神奈川県川崎市中原区上小田中4丁目1番
 1号
 (74) 代理人 100105142
 弁理士 下田 憲次
 (72) 発明者 三浦 隆司
 神奈川県川崎市中原区上小田中4丁目1番
 1号 富士通株式会社内

審査官 酒井 恭信

最終頁に続く

(54) 【発明の名称】 演算処理装置及び演算処理装置の制御方法

(57) 【特許請求の範囲】

【請求項 1】

キャッシュメモリと、

前記キャッシュメモリを制御する第1のコントローラと、

前記キャッシュメモリを介さずにアクセスされるノンキャッシュ空間が割り当てられた第2のコントローラと、

を含み、前記第1のコントローラは、前記ノンキャッシュ空間への第1及び第2のアクセスリクエストの処理順序が追い越し可能であり且つ前記第1及び第2のアクセスリクエストのアクセス先が同一である条件が満たされる場合、前記第2のコントローラへ先に発行した前記第1のアクセスリクエストに対する前記第2のコントローラからの完了通知を待たずに前記第2のアクセスリクエストを前記第2のコントローラに発行し、前記条件が満たされない場合、前記第2のコントローラへ先に発行した前記第1のアクセスリクエストに対する前記第2のコントローラからの完了通知を待ってから前記第2のアクセスリクエストを前記第2のコントローラに発行することを特徴とする演算処理装置。

【請求項 2】

前記第2のコントローラは複数のコントローラを含み、前記第1及び第2のアクセスリクエストが前記複数のコントローラのうちの1つの同一のコントローラにアクセスする場合に、前記第1のコントローラは、前記第1及び第2のアクセスリクエストのアクセス先が同一であると判定することを特徴とする請求項1記載の演算処理装置。

【請求項 3】

前記第 1 のコントローラは、1 次キャッシュコントローラと 2 次キャッシュコントローラとを含み、

前記 1 次キャッシュコントローラは、前記ノンキャッシュ空間への前記第 1 及び第 2 のアクセスリクエストの処理順序が追い越し可能であると判定した場合、前記 2 次キャッシュコントローラへ先に発行した前記第 1 のアクセスリクエストに対する前記 2 次キャッシュコントローラからの応答を待ってから前記第 2 のアクセスリクエストを前記 2 次キャッシュコントローラに発行し、前記ノンキャッシュ空間への前記第 1 及び第 2 のアクセスリクエストの処理順序が追い越し可能でないと判定した場合、前記 2 次キャッシュコントローラへ先に発行した前記第 1 のアクセスリクエストに対する前記第 2 のコントローラからの完了通知を待ってから前記第 2 のアクセスリクエストを前記 2 次キャッシュコントローラに発行し、

10

前記 2 次キャッシュコントローラは、前記第 1 のアクセスリクエストを前記第 2 のコントローラに発行すると、前記完了通知を待つことなく前記応答を前記 1 次キャッシュコントローラに送信する

ことを特徴とする請求項 1 又は 2 記載の演算処理装置。

【請求項 4】

前記第 1 のコントローラは、前記ノンキャッシュ空間への複数のアクセスリクエストの処理順序が追い越し可能であるが前記複数のアクセスリクエストのアクセス先が同一でない場合、前記第 2 のコントローラへ先に発行した全てのアクセスリクエストに対する前記第 2 のコントローラからの完了通知を待ってから、次のアクセスリクエストを前記第 2 のコントローラに発行することを特徴とする請求項 1 乃至 3 何れか一項記載の演算処理装置。

20

【請求項 5】

論理アドレスを物理アドレスに変換するために用いる T L B を更に含み、

前記第 1 のコントローラは、前記 T L B に含まれる情報に基づいて、前記ノンキャッシュ空間への前記第 1 及び第 2 のアクセスリクエストの処理順序が追い越し可能であるか否かを判定することを特徴とする請求項 1 乃至 4 何れか一項記載の演算処理装置。

【請求項 6】

キャッシュメモリと、前記キャッシュメモリを制御する第 1 のコントローラと、前記キャッシュメモリを介さずにアクセスされるノンキャッシュ空間が割り当てられた第 2 のコントローラとを含む演算処理装置において、

30

前記ノンキャッシュ空間への第 1 及び第 2 のアクセスリクエストの処理順序が追い越し可能である第 1 の条件が満たされるか否かを判定し、

前記第 1 及び第 2 のアクセスリクエストのアクセス先が同一である第 2 の条件が満たされるか否かを判定し、

前記第 1 の条件及び前記第 2 の条件の両方が満たされる場合、前記第 2 のコントローラへ先に発行した前記第 1 のアクセスリクエストに対する前記第 2 のコントローラからの完了通知を待たずに、前記第 2 のアクセスリクエストを前記第 1 のコントローラから前記第 2 のコントローラに発行し、

前記第 1 の条件及び前記第 2 の条件の少なくとも一方が満たされない場合、前記第 2 のコントローラへ先に発行した前記第 1 のアクセスリクエストに対する前記第 2 のコントローラからの完了通知を待ってから、前記第 2 のアクセスリクエストを前記第 1 のコントローラから前記第 2 のコントローラに発行する

40

ことを特徴とする演算処理装置の制御方法。

【請求項 7】

前記第 2 のコントローラは複数のコントローラを含み、前記第 1 及び第 2 のアクセスリクエストが前記複数のコントローラのうちの 1 つの同一のコントローラにアクセスする場合に、前記第 1 のコントローラは、前記第 2 の条件が満たされると判定することを特徴とする請求項 6 記載の演算処理装置の制御方法。

50

【発明の詳細な説明】

【技術分野】

【0001】

本願開示は、演算処理装置及び演算処理装置の制御方法に関する。

【背景技術】

【0002】

CPUが備えるコマンドとして、ノンキャッシュ空間へのアクセス命令がある。ノンキャッシュ空間とは、キャッシュメモリを介在せずにアクセスされるメモリ空間をいう。ノンキャッシュ空間へのアクセス命令は、キャッシュメモリを介さないで、アクセス対象のデバイスに割り当てされているメモリ空間にアクセスするための命令である。ノンキャッシュ命令によるノンキャッシュ空間へのアクセスは、ノンキャッシュャブル空間として定義されたアドレス空間への読み書きとして定義される。

10

【0003】

ノンキャッシュ空間へのアクセスの動作は、レジスタの読み書きをしたり、I/O装置への動作を指示したりするものである。例えば、メモリコントローラへのノンキャッシュリクエストにより、メモリコントローラ内にあるレジスタにアクセスできる。また例えば、PCIe (Peripheral Component Interconnect Express) コントローラへのノンキャッシュリクエストにより、PCIe コントローラ内のレジスタ、又は、例えばPCIe カード等の外部装置内のレジスタにアクセスすることができる。また例えば、CPUインタフェースコントローラへのノンキャッシュリクエストにより、他CPUに繋がっているメモリコントローラやPCIe コントローラ等の装置へアクセスすることができる。

20

【0004】

デバイスのドライバからの割り込み処理として、当該デバイスに対してノンキャッシュライト動作を複数回実行し、その後同期をとるためにノンキャッシュリード動作を1回実行するアクセスパターンがよくある。従って、ノンキャッシュリクエストを連続発行する際に効率的に発行できることが望ましい。

【0005】

従来のノンキャッシュ制御において、命令コントローラから発行されたリクエストを受けた1次キャッシュコントローラは、TLB (Translation Lookaside Buffer) を用いて、アクセス先の仮想アドレスを物理アドレスへと変換する。この物理アドレス内のNCビット (キャッシュャブル空間かノンキャッシュャブル空間かを表すビット) がノンキャッシュ空間を示す場合、1次キャッシュコントローラは、ノンキャッシュリクエストを2次キャッシュコントローラへ発行する。2次キャッシュコントローラは、このノンキャッシュリクエストを、リクエスト宛先のシステムコントローラ (メモリコントローラ、PCIe コントローラ、CPUインタフェースコントローラ等) に発行する。

30

【0006】

命令コントローラから発行される次のリクエストがノンキャッシュリクエストの場合、このリクエストは1次キャッシュコントローラにおいて待機する。システムコントローラにおいて最初のリクエストの処理が完了すると、当該システムコントローラは、2次キャッシュコントローラに完了通知を発行する。完了通知を受け取った2次キャッシュコントローラは、1次キャッシュコントローラに完了通知を通知する。完了通知を受け取った1次キャッシュコントローラは、待機させておいた次のノンキャッシュリクエストを2次キャッシュコントローラに発行することができる。

40

上述の説明のように従来技術では、デバイスからの完了通知を待たなければ、1次キャッシュコントローラを内蔵するCPUコアから、2次キャッシュコントローラに対して、次のノンキャッシュリクエストを発行することができない。しかしながら、CPUコアからデバイスの完了通知を待たずに、連続でノンキャッシュリクエストを発行できる場合があるにも関わらず、一律にデバイスの完了通知を待つ方式としたのでは、ノンキャッシュリクエストの処理効率が悪くなってしまう。

【先行技術文献】

50

【特許文献】

【 0 0 0 7 】

【特許文献 1】特開 2 0 0 7 - 1 7 2 6 0 9 号公報

【発明の概要】

【発明が解決しようとする課題】

【 0 0 0 8 】

以上を鑑みると、効率よくノンキャッシュリクエストを発行できる演算処理装置が望まれる。

【課題を解決するための手段】

【 0 0 0 9 】

演算処理装置は、キャッシュメモリと、前記キャッシュメモリを制御する第 1 のコントローラと、前記キャッシュメモリを介さずにアクセスされるノンキャッシュ空間が割り当てられた第 2 のコントローラを含み、前記第 1 のコントローラは、前記ノンキャッシュ空間への第 1 及び第 2 のアクセスリクエストの処理順序が追い越し可能であり且つ前記第 1 及び第 2 のアクセスリクエストのアクセス先が同一である条件が満たされる場合、前記第 2 のコントローラへ先に発行した前記第 1 のアクセスリクエストに対する前記第 2 のコントローラからの完了通知を待たずに前記第 2 のアクセスリクエストを前記第 2 のコントローラに発行し、前記条件が満たされない場合、前記第 2 のコントローラへ先に発行した前記第 1 のアクセスリクエストに対する前記第 2 のコントローラからの完了通知を待ってから前記第 2 のアクセスリクエストを前記第 2 のコントローラに発行することを特徴とする。

【発明の効果】

【 0 0 1 0 】

少なくとも 1 つの実施例によれば、効率よくノンキャッシュリクエストを発行できる演算処理装置が提供される。

【図面の簡単な説明】

【 0 0 1 1 】

【図 1】演算処理装置及び周辺デバイス等を含む演算処理システムの構成の一例を示す図である。

【図 2】IU - REQ のフォーマットの一例を示す図である。

【図 3】NC - REQ のフォーマットの一例を示す図である。

【図 4】アクセスリクエスト発行処理の流れを示すフローチャートである。

【図 5】アクセスリクエスト発行処理の動作の一例を示す図である。

【図 6】アクセスリクエスト発行処理の動作の別の一例を示す図である。

【図 7】アクセスリクエスト発行処理の動作の更に別の一例を示す図である。

【図 8】TLB の TTE のフォーマットを示す図である。

【図 9】アクセスリクエスト発行処理の回路構成を示す図である。

【発明を実施するための形態】

【 0 0 1 2 】

以下に、本発明の実施例を添付の図面を用いて詳細に説明する。

【 0 0 1 3 】

図 1 は、演算処理装置及び周辺デバイス等を含む演算処理システムの構成の一例を示す図である。図 1 の演算処理システムは、演算処理装置である CPU 10、メインメモリ 11、外部装置 12、及び他 CPU 13 を含む。CPU 10 は、複数の CPU コア 21 - 1 乃至 21 - n、2 次キャッシュコントローラ 22、システムコントローラ 23、及び 2 次キャッシュメモリ 24 を含む。複数の CPU コア 21 - 1 乃至 21 - n の各々は同一の構成を有しており、CPU コア 21 - 1 に代表して示されるように、演算器 30、命令コントローラ 31、1 次キャッシュメモリ 32、1 次キャッシュコントローラ 33、及び TLB 34 を含む。また、2 次キャッシュコントローラ内には、各 CPU コアに対応するアドレス判断部 38 - 1 乃至 38 - n を含む。システムコントローラ 23 は、メモリコントローラ 35、PCIe コントローラ 36、及び CPU インタフェースコントローラ 37 を含

10

20

30

40

50

む。

【 0 0 1 4 】

なお図 1 において、各ボックスで示される各機能ブロックと他の機能ブロックとの境界は、基本的には機能的な境界を示すものであり、物理的な位置の分離、電気的な信号の分離、制御論理的な分離等に対応するとは限らない。各機能ブロックは、他のブロックと物理的にある程度分離された 1 つのハードウェアモジュールであってもよいし、或いは他のブロックと物理的に一体となったハードウェアモジュール中の 1 つの機能を示したものであってもよい。

【 0 0 1 5 】

C P U コア 2 1 - 1 乃至 2 1 - n は、2 次キャッシュメモリ 2 4 を共有し、2 次キャッシュコントローラ 2 2 を介して 2 次キャッシュメモリ 2 4 にアクセスする。また C P U コア 2 1 - 1 乃至 2 1 - n は、2 次キャッシュコントローラ 2 2 を介してシステムコントローラ 2 3 の各コントローラにアクセスする。メモリコントローラ 3 5 は、外部メモリであるメインメモリ 1 1 を制御するコントローラである。P C I e コントローラ 3 6 は、P C I e カード等の外部装置 1 2 を制御するコントローラである。C P U インタフェースコントローラ 3 7 は、C P U 1 0 と同様の構成及び機能を有する他 C P U 1 3 との情報のやり取りを制御する。システムコントローラ 2 3 のこれらのコントローラには、キャッシュメモリを介さずにアクセスされるノンキャッシュ空間が割り当てられている。

【 0 0 1 6 】

命令コントローラ 3 1 は、1 次キャッシュメモリ 3 2 からフェッチした命令をデコードし、デコード結果に応じて演算器 3 0 による演算命令の実行を制御する。また命令コントローラ 3 1 は、ロード命令やストア命令等のアクセスリクエスト (I U - R E Q : Instruction Unit-Request) を 1 次キャッシュコントローラ 3 3 に発行し、1 次キャッシュメモリ 3 2 に対するデータのロードやデータのストアを実行する。

【 0 0 1 7 】

命令コントローラ 3 1 から発行されたアクセスリクエスト (I U - R E Q) を受けた 1 次キャッシュコントローラ 3 3 は、T L B 3 4 を参照して、アクセスリクエスト中の仮想アドレスを物理アドレスへと変換する。図 8 より T L B 3 4 の T T E (Translation Table Entry) には、アクセス空間にサイドエフェクトがあるかないかを示す E ビット 5 0 (T T E e) と物理ページ番号を示す P A 5 1 がある。アクセスリクエスト中のアクセス先のアドレスが、インオーダー処理を要件とするデバイスのメモリ空間に対応する場合には、T L B 3 4 を参照することにより T T E e = 1 が得られる。またアクセスリクエスト中のアクセス先のアドレスが、アクセスリクエストの処理順序が追い越し可能であるデバイスのメモリ空間に対応する場合には、T L B 3 4 を参照することにより T T E e = 0 が得られる。

【 0 0 1 8 】

また T L B 3 4 を参照して仮想アドレスと P A 5 1 から得られた物理アドレスには、当該アドレスがキャッシュابل空間かノンキャッシュابل空間かを示すビットである N C ビットが含まれる。1 次キャッシュコントローラ 3 3 は、N C ビットがノンキャッシュ空間を示す場合、ノンキャッシュリクエスト (N C - R E Q : NonCache-Request) を、2 次キャッシュコントローラ 2 2 へ発行する。1 次キャッシュコントローラ 3 3 は、N C ビットがキャッシュ空間を示す場合、1 次キャッシュメモリ 3 2 に対するアクセスを実行する。1 次キャッシュメモリ 3 2 と 2 次キャッシュメモリ 2 4 とは階層構造となっており、1 次キャッシュメモリ 3 2 においてアクセスがヒットしない場合、2 次キャッシュコントローラ 2 2 を介して 2 次キャッシュメモリ 2 4 へのアクセスが実行される。1 次キャッシュコントローラ 3 3 と 2 次キャッシュコントローラ 2 2 とは、キャッシュメモリ (1 次キャッシュメモリ 3 2 及び 2 次キャッシュメモリ 2 4) を制御するコントローラである。

【 0 0 1 9 】

図 2 は、I U - R E Q のフォーマットの一例を示す図である。図 2 に示されるように、命令コントローラ 3 1 から 1 次キャッシュコントローラ 3 3 に発行するアクセスリクエス

10

20

30

40

50

トであるIU - REQは、命令コード41と仮想アドレス42とを含む。命令コード（オペコード）41は、当該命令の種類を示すコードであり、例えばストア命令（書き込み命令）であることを示すコードやロード命令（読み出し命令）であることを示すコードである。仮想アドレス42は、ストア命令やロード命令等においてアクセスする先を示す仮想アドレスである。

【0020】

図3は、NC - REQのフォーマットの一例を示す図である。図3に示されるように、1次キャッシュコントローラ33から2次キャッシュコントローラ22へ発行するアクセスリクエストであるNC - REQは、ノンキャッシュ命令コード（オペコード）43と物理アドレス44とCORE - ID49を含む。ノンキャッシュ命令コード43は、当該命令の種類を示すコードであり、例えば当該命令が例えばノンキャッシュストア命令（書き込み命令）であることを示すコードや、ノンキャッシュロード命令（読み出し命令）であることを示すコードである。物理アドレス44は、NCビット45、CPU識別子であるCPU - ID46、コントローラ識別子であるCTL - ID47、及びアドレス48を含む。また、CORE - ID49はCPUコア識別子である。

【0021】

NCビット45は、前述のように、アクセス先のアドレス（アドレス48が指し示す先）がキャッシュابل空間かノンキャッシュابل空間かを示すビットである。またCPU - ID46は、当該NC - REQがアクセスする先のCPUを識別する識別子である。例えばNC - REQが、図1に示すCPU10内のPCIeコントローラ36にアクセスする場合、CPU - ID46は、図1に示す当該CPU10を識別する識別子となる。またNC - REQが、図1に示す他CPU13のPCIeコントローラにアクセスする場合、CPU - ID46は、図1に示す他CPU13を識別する識別子となる。CTL - ID47は、アクセス先のコントローラを識別する識別子である。例えばNC - REQが、図1に示すCPU10内のPCIeコントローラ36にアクセスする場合、CTL - ID47は、図1に示すPCIeコントローラ36を識別する識別子となる。アドレス48は、メモリ空間中のアクセス先の物理アドレスである。例えばNC - REQが、図1に示すCPU10内のPCIeコントローラ36にアクセスする場合、アドレス48は、PCIeコントローラ36に割り当てられたメモリ空間中の特定のアドレスとなる。CORE - ID49は各CPUコア21 - 1乃至21 - nからのNC - REQから作成され、NC - REQのリクエスト元であるCPUコアを識別する識別子となる。

【0022】

図4は、アクセスリクエスト発行処理の流れを示すフローチャートである。図4を参照して、アクセスリクエスト発行処理について説明する。

【0023】

ステップS1において、命令コントローラ31がリクエスト（IU - REQ）を発行する。ステップS2において、1次キャッシュコントローラ33が、リクエスト（IU - REQ）を受け付け、リクエスト中の仮想アドレスに基づきTLB34を参照することにより、仮想アドレスに対応する物理アドレスを得る。1次キャッシュコントローラ33は更に、物理アドレス中のNCビットを確認することにより、当該リクエストがノンキャッシュ空間へのアクセスであるか否かを判断する。当該リクエストがキャッシュ空間へのアクセスである場合には、1次キャッシュメモリ32や2次キャッシュメモリ24に対する通常のキャッシュアクセス制御が実行される。当該リクエストがノンキャッシュ空間へのアクセスである場合には、ステップS3以降の処理が実行される。

【0024】

ステップS3で、1次キャッシュコントローラ33は、TLB34を参照することにより得られたTTEの値が0であるか否かを判定する。TTE = 0でない場合（TTE = 1の場合）、アクセス先のメモリ空間がインオーダー処理を要件とするデバイスであることを意味する。この場合、ステップS4で、1次キャッシュコントローラ33は、当該リクエストの直前のリクエストに対するアクセス先のデバイス又はコントローラからの

完了通知（リクエストされた処理の実行の完了を知らせる通知）を既に受領しているか否かを判定する。既に完了通知を受領していれば、１次キャッシュコントローラ３３は、２次キャッシュコントローラ２２にリクエスト（NC - REQ）を発行する（ステップＳ８）。完了通知を受領していない場合、ステップＳ５で、１次キャッシュコントローラ３３は、デバイスからの完了通知が到来するまで待機する。完了通知が到来すると、１次キャッシュコントローラ３３は、２次キャッシュコントローラ２２にリクエスト（NC - REQ）を発行する（ステップＳ８）。

【００２５】

このように１次キャッシュコントローラ３３は、ノンキャッシュ空間へのアクセスリクエストの処理順序が追い越し可能でないと判定した場合、２次キャッシュコントローラ２２へ先に発行したアクセスリクエストに対する完了通知を待つ。この完了通知は、アクセス先のデバイスからシステムコントローラ２３を介して、又はシステムコントローラ２３から到来する。１次キャッシュコントローラ３３は、先に発行したアクセスリクエストに対する完了通知を待ってから（即ち完了通知が到来すると）、現在処理中のアクセスリクエストを２次キャッシュコントローラ２２に発行する。

10

【００２６】

ステップＳ３の判定結果がTTE = 0である場合、アクセス先のメモリ空間がアクセスリクエストの処理順序が追い越し可能であるデバイスに対応することを意味する。この場合、ステップＳ６で、直前のリクエストについて２次キャッシュコントローラ２２からの応答（NC - TKN）があるか否かを確認する。この応答（NC - TKN）は、２次キャッシュコントローラ２２が、アクセスリクエストをシステムコントローラ２３に発行すると、上記の完了通知を待つことなく１次キャッシュコントローラ３３に送信するものである。

20

【００２７】

応答を受領していない場合（ステップＳ６でNo）、ステップＳ７で、１次キャッシュコントローラ３３は応答が到来するまで待機する。応答を受領すると（ステップＳ６でYes）、ステップＳ８で、１次キャッシュコントローラ３３は、２次キャッシュコントローラ２２にリクエスト（NC - REQ）を発行する。

【００２８】

このように１次キャッシュコントローラ３３は、ノンキャッシュ空間へのアクセスリクエストの処理順序が追い越し可能であると判定した場合、２次キャッシュコントローラ２２へ先に発行したアクセスリクエストに対する応答を待つ。この応答は、２次キャッシュコントローラ２２が、アクセスリクエストをシステムコントローラ２３に発行すると、完了通知を待つことなく１次キャッシュコントローラ３３に送信するものである。１次キャッシュコントローラ３３は、先に発行したアクセスリクエストに対する応答を待ってから（即ち応答が到来すると）、現在処理中のアクセスリクエストを２次キャッシュコントローラ２２に発行する。

30

【００２９】

ステップＳ９で、２次キャッシュコントローラ２２は、１次キャッシュコントローラ３３からのアクセスリクエストを受け付ける。ステップＳ１０で、２次キャッシュコントローラ２２は、システムコントローラ２３へのリクエストの発行数を確認する。ここでリクエストの発行数とは、２次キャッシュコントローラ２２からシステムコントローラ２３へ発行したリクエストであって、その完了通知がシステムコントローラ２３から届いていないリクエストの数である。

40

【００３０】

ステップＳ１０での判定結果が発行数 = 0を示す場合、ステップＳ１１で、２次キャッシュコントローラ２２はシステムコントローラ２３にアクセスリクエストを発行し、且つ、１次キャッシュコントローラ３３に応答NC - TKNを送信する。２次キャッシュコントローラ２２は更に、発行したリクエストのアドレスを保持し（記憶し）、発行数のカウントを１増加する。

50

【 0 0 3 1 】

ステップ S 1 0 での判定結果が発行数が 0 より大きく且つ f u l l 未満であることを示す場合、ステップ S 1 2 で、2 次キャッシュコントローラ 2 2 は、直前のアクセスリクエストの記憶してあるアドレスと現在処理中のアクセスリクエストのアドレスとを比較する。この比較はアクセス先（「宛先」）を確認するために行われる。

【 0 0 3 2 】

直前と現在との 2 つのアクセスリクエストのアクセス先が同一である条件が満たされる場合（ステップ S 1 2 で「同一宛先」）、ステップ S 1 1 で、2 次キャッシュコントローラ 2 2 は、システムコントローラ 2 3 にアクセスリクエストを発行する。この際、2 次キャッシュコントローラ 2 2 は、システムコントローラ 2 3 へ先に発行したアクセスリクエストに対するシステムコントローラ 2 3 からの完了通知を待たずに、現在処理中のアクセスリクエストをシステムコントローラ 2 3 に発行する。なおシステムコントローラ 2 3 は複数のコントローラ（3 5 , 3 6 , 3 7 ）を含んでいる。従って、2 次キャッシュコントローラ 2 2 がシステムコントローラ 2 3 にアクセスリクエストを発行する場合には、N C - R E Q の物理アドレス（図 3 参照）の C T L - I D 4 7 が示す 1 つのコントローラにアクセスリクエストが発行されることになる。また、2 つのアクセスリクエストがこれら複数のコントローラのうちの 1 つの同一のコントローラにアクセスする場合に、2 次キャッシュコントローラ 2 2 は、2 つのアクセスリクエストのアクセス先が同一であると判定する。より具体的には、図 3 に示す物理アドレス 4 4 のうち、上位ビット側の C P U - I D 4 6 、及び C T L - I D 4 7 が 2 つのアクセスリクエストにおいて一致している場合に、2 つのアクセスリクエストのアクセス先が同一であると判定する。下位ビット側のアドレス 4 8 が 2 つのアクセスリクエストにおいて異なっている場合、2 つのアクセスリクエストのアクセス先が同一であると判定する。

【 0 0 3 3 】

直前と現在との 2 つのアクセスリクエストのアクセス先が同一である条件が満たされない場合（ステップ S 1 2 で「別宛先」）、ステップ S 1 3 で、2 次キャッシュコントローラ 2 2 は、発行数が 0 になるまで待機する。発行数は、システムコントローラ 2 3 へ既に発行したアクセスリクエストに対するシステムコントローラ 2 3 からの完了通知（リクエストされた処理実行が完了したことを知らせる通知：N C - E N D ）が到来するたびに、1 減少する。発行数が 0 になると、2 次キャッシュコントローラ 2 2 は、現在処理中のアクセスリクエストをシステムコントローラ 2 3 に発行する（ステップ S 1 1 ）。このようにして、アクセス先が同一である条件が満たされない場合、2 次キャッシュコントローラ 2 2 は、システムコントローラ 2 3 へ先に発行したアクセスリクエストに対するシステムコントローラ 2 3 からの完了通知を待機する。2 次キャッシュコントローラ 2 2 は、先に発行したアクセスリクエストに対するシステムコントローラ 2 3 からの完了通知を待ってから、現在処理中のアクセスリクエストをシステムコントローラ 2 3 に発行する。この際、システムコントローラ 2 3 へ先に発行したアクセスリクエストが複数有る場合には、全てのアクセスリクエストに対するシステムコントローラ 2 3 からの完了通知を待ってから、現在処理中のアクセスリクエストをシステムコントローラ 2 3 に発行する。

【 0 0 3 4 】

ステップ S 1 0 での判定結果が発行数 = f u l l を示す場合、ステップ S 1 3 で、2 次キャッシュコントローラ 2 2 は、発行数 < f u l l になるまで待機する。ここで f u l l とは、2 次キャッシュコントローラ 2 2 が受け付けることのできるリクエストの数であり、例えば、受け付けたリクエストを保持する 2 次キャッシュコントローラ 2 2 に内蔵のバッファの容量に依存する。

【 0 0 3 5 】

ステップ S 1 4 で、システムコントローラ 2 3 （メモリコントローラ 3 5 、P C I e コントローラ 3 6 、又は C P U インタフェースコントローラ 3 7 ）がリクエストを受け付け、必要に応じてデバイス側にリクエストを発行する。即ち、メインメモリ 1 1 や、外部装置 1 2 や、他 C P U 1 3 等にリクエストを発行する。なお、例えば P C I e コントローラ

36 がリクエストを受け付けた場合、当該リクエストが P C I e コントローラ 36 内部のレジスタにアクセスするリクエストである場合には、外部装置 12 にリクエストを発行することはない。一方、当該リクエストが外部装置 12 にアクセスするリクエストである場合には、P C I e コントローラ 36 から外部装置 12 にリクエストが発行される。

【0036】

ステップ S 15 で、デバイス側或いはシステムコントローラ 23 によりリクエスト処理が完了すると、処理完了を示す完了通知 (N C - E N D) がシステムコントローラ 23 から 2 次キャッシュコントローラ 22 に送信される。前述のように、完了通知 (N C - E N D) が到来するたびに、2 次キャッシュコントローラ 22 側では、発行数を 1 減少する処理を行う。

10

【0037】

図 5 は、アクセスリクエスト発行処理の動作の一例を示す図である。まず命令コントローラ 31 から 1 次キャッシュコントローラ 33 にリクエスト I U - R E Q 1 が発行される。このリクエスト I U - R E Q 1 に対応する T T E e が 0 である場合、ノンキャッシュリクエスト N C - R E Q 1 が、1 次キャッシュコントローラ 33 から 2 次キャッシュコントローラ 22 へと発行される。ここで、次のリクエスト I U - R E Q 2 及び I U - R E Q 3 が命令コントローラ 31 から発行され、対応する T T E e が 0 である場合、対応するノンキャッシュリクエスト N C - R E Q 2 及び N C - R E Q 3 は、1 次キャッシュコントローラ 33 で待機する。

【0038】

20

1 次キャッシュコントローラ 33 からノンキャッシュリクエスト N C - R E Q 1 を受け取った 2 次キャッシュコントローラ 22 は、初期状態での発行数が 0 であるので、直ちにシステムコントローラ 23 にノンキャッシュリクエスト N C - R E Q 1 を発行する。この際、発行数は 1 増加される (ステップ S 51)。また N C - R E Q 1 の発行と同時に、又は並行して、又は直前若しくは直後に、2 次キャッシュコントローラ 22 は、N C - R E Q 1 の発行を示す通知として、1 次キャッシュコントローラ 33 に応答 N C - T K N 1 を発行する。N C - T K N 1 を受け取った 1 次キャッシュコントローラ 33 は、次のノンキャッシュリクエスト N C - R E Q 2 を 2 次キャッシュコントローラ 22 に発行する。

【0039】

N C - R E Q 2 を受け取った 2 次キャッシュコントローラ 22 は、N C - R E Q 1 のアクセス先と N C - R E Q 2 のアクセス先とを比較して、同一アクセス先 (同一宛先) ならば、直ちにシステムコントローラ 23 に N C - R E Q 2 を発行する。この際、発行数は 1 増加される (ステップ S 52)。また N C - R E Q 2 の発行と同時に、又は並行して、又は直前若しくは直後に、2 次キャッシュコントローラ 22 は、N C - R E Q 2 の発行を示す通知として、1 次キャッシュコントローラ 33 に応答 N C - T K N 2 を発行する。N C - T K N 2 を受け取った 1 次キャッシュコントローラ 33 は、次のノンキャッシュリクエスト N C - R E Q 3 を 2 次キャッシュコントローラ 22 に発行する。

30

【0040】

N C - R E Q 3 を受け取った 2 次キャッシュコントローラ 22 は、N C - R E Q 2 のアクセス先と N C - R E Q 3 のアクセス先とを比較して、同一アクセス先 (同一宛先) ならば、直ちにシステムコントローラ 23 に N C - R E Q 3 を発行する。この際、発行数は 1 増加される (ステップ S 53)。また N C - R E Q 3 の発行と同時に、又は並行して、又は直前若しくは直後に、2 次キャッシュコントローラ 22 は、N C - R E Q 3 の発行を示す通知として、1 次キャッシュコントローラ 33 に応答 N C - T K N 3 を発行する。

40

【0041】

リクエスト完了通知 N C - E N D 1 及び N C - E N D 2 がシステムコントローラ 23 から 2 次キャッシュコントローラ 22 に送信されると、2 次キャッシュコントローラ 22 では各完了通知毎に発行数を 1 減算する (ステップ S 54、S 56)。

図 6 は、アクセスリクエスト発行処理の動作の別の一例を示す図である。まず命令コントローラ 31 から 1 次キャッシュコントローラ 33 にリクエスト I U - R E Q 1 が発行さ

50

れる。このリクエスト I U - R E Q 1 に対応する T T E e が 0 である場合、ノンキャッシュリクエスト N C - R E Q 1 が、1 次キャッシュコントローラ 3 3 から 2 次キャッシュコントローラ 2 2 へと発行される。ここで、次のリクエスト I U - R E Q 2 及び I U - R E Q 3 が命令コントローラ 3 1 から発行され、対応する T T E e が 0 である場合、対応するノンキャッシュリクエスト N C - R E Q 2 及び N C - R E Q 3 は、1 次キャッシュコントローラ 3 3 で待機する。

【 0 0 4 2 】

1 次キャッシュコントローラ 3 3 からノンキャッシュリクエスト N C - R E Q 1 を受け取った 2 次キャッシュコントローラ 2 2 は、初期状態での発行数が 0 であるので、直ちにシステムコントローラ 2 3 にノンキャッシュリクエスト N C - R E Q 1 を発行する。この際、発行数は 1 増加される（ステップ S 6 1）。また N C - R E Q 1 の発行と同時に、又は並行して、又は直前若しくは直後に、2 次キャッシュコントローラ 2 2 は、N C - R E Q 1 の発行を示す通知として、1 次キャッシュコントローラ 3 3 に応答 N C - T K N 1 を発行する。N C - T K N 1 を受け取った 1 次キャッシュコントローラ 3 3 は、次のノンキャッシュリクエスト N C - R E Q 2 を 2 次キャッシュコントローラ 2 2 に発行する。

【 0 0 4 3 】

N C - R E Q 2 を受け取った 2 次キャッシュコントローラ 2 2 は、N C - R E Q 1 のアクセス先と N C - R E Q 2 のアクセス先とを比較する。アクセス先が異なる（別宛先）であるので（ステップ S 6 2）、2 次キャッシュコントローラ 2 2 は、先にシステムコントローラ 2 3 へ発行した N C - R E Q 1 に対する完了通知 N C - E N D 1 が到来するまで N C - R E Q 2 を待機させる。N C - R E Q 1 に対する完了通知 N C - E N D 1 がシステムコントローラ 2 3 から到来すると、2 次キャッシュコントローラ 2 2 は、発行数を 1 減算する（ステップ S 6 3）。これにより発行数が 0 となるので、2 次キャッシュコントローラ 2 2 は、システムコントローラ 2 3 に N C - R E Q 2 を発行する。これに応じて、発行数は 1 増加される（ステップ S 6 3）。また N C - R E Q 2 の発行と同時に、又は並行して、又は直前若しくは直後に、2 次キャッシュコントローラ 2 2 は、N C - R E Q 2 の発行を示す通知として、1 次キャッシュコントローラ 3 3 に応答 N C - T K N 2 を発行する。N C - T K N 2 を受け取った 1 次キャッシュコントローラ 3 3 は、次のノンキャッシュリクエスト N C - R E Q 3 を 2 次キャッシュコントローラ 2 2 に発行する。

【 0 0 4 4 】

N C - R E Q 3 を受け取った 2 次キャッシュコントローラ 2 2 は、N C - R E Q 2 のアクセス先と N C - R E Q 3 のアクセス先とを比較する。アクセス先が異なる（別宛先）であるので（ステップ S 6 4）、2 次キャッシュコントローラ 2 2 は、先にシステムコントローラ 2 3 へ発行した N C - R E Q 2 に対する完了通知 N C - E N D 2 が到来するまで N C - R E Q 3 を待機させる。

【 0 0 4 5 】

リクエスト完了通知 N C - E N D 2 がシステムコントローラ 2 3 から 2 次キャッシュコントローラ 2 2 に送信されると、2 次キャッシュコントローラ 2 2 では発行数を 1 減算する（ステップ S 6 5）。

【 0 0 4 6 】

図 7 は、アクセスリクエスト発行処理の動作の更に別の一例を示す図である。図 7 に示す例では、命令コントローラ 3 1、1 次キャッシュコントローラ 3 3、及び 2 次キャッシュコントローラ 2 2 の動作は、図 5 に示す動作と同じである。図 7 においては、図 5 の場合と異なり、システムコントローラ 2 3 又はデバイスによるリクエスト処理が、N C - R E Q 1 よりも N C - R E Q 2 に対して先に完了している。その結果、N C - R E Q 2 に対する完了通知 N C - E N D 2 が、N C - R E Q 1 に対する完了通知 N C - E N D 1 よりも先に、システムコントローラ 2 3 から 2 次キャッシュコントローラ 2 2 に発行されている。N C - R E Q 1 及び N C - R E Q 2 は、T T E e が 0 であり、アクセス先のメモリ空間がアクセスリクエストの処理順序が追い越し可能であるデバイスに対応している。従って、図 7 に示す動作例の場合のように、後にリクエストが発行された N C - E N D 2 が、先

10

20

30

40

50

にリクエストが発行された $NC - REQ1$ を追い越して、先に処理が完了してしまってもよい。このようにアクセスリクエストの処理順序が追い越し可能である場合、後に発行されたアクセスリクエストが、先に発行されたアクセスリクエストよりも、先に処理が開始されて先に処理が完了したり、後に処理が開始されて先に処理が完了したりしてよい。即ち、アクセスリクエストの処理順序が追い越し可能である場合、処理順序がリクエスト発行順序と異なってもよく、処理順序がリクエスト発行順序と異なったことによる結果として、処理結果に支障が生じることがない。

【0047】

図9はアクセスリクエスト発行処理の回路構成を示す図である。複数のCPUコア21-1乃至21-nからそれぞれ2次キャッシュコントローラ22に $NC - REQ$ が発行される。2次キャッシュコントローラ22には各CPUコア21-1乃至21-nに対応したアドレス判断部38-1乃至38-nを有しており、それぞれのアドレス判断部38-1乃至38-nで宛先確認を行う。アドレス判断部38-1乃至38-nでは、CPUコア21-1乃至21-nから $NC - REQ$ が来た場合、そのリクエストのアドレスを保持しておく。宛先確認は、現在処理中の $NC - REQ$ と保持していた直前の $NC - REQ$ のCPU-ID46、及びCTL-ID47を比較して同一宛先かどうかを判定する。システムコントローラ23へリクエスト発行後もアドレスは保持しておく。選択部39では発行可能となったCPUコア21-1乃至21-nからのリクエストを任意に選択してシステムコントローラ23へと $NC - REQ$ を発行する。また、システムコントローラ23からのリクエスト完了通知を受けた2次キャッシュコントローラ22は、応答部40で完了通知に付随されたCORE-IDを確認して、リクエスト発行元のCPUコアへと $NC - END$ を返す。

【0048】

以上、本発明を実施例に基づいて説明したが、本発明は上記実施例に限定されるものではなく、特許請求の範囲に記載の範囲内で様々な変形が可能である。

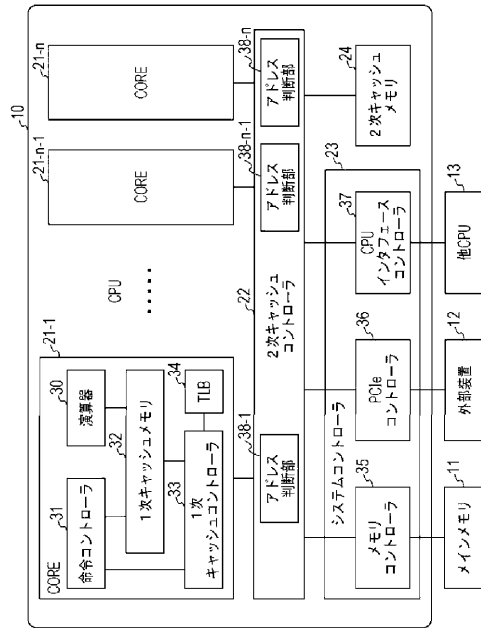
【符号の説明】

【0049】

- 10 CPU
- 11 メインメモリ
- 12 外部装置
- 13 他CPU
- 21-1乃至21-n CPUコア
- 22 2次キャッシュコントローラ
- 23 システムコントローラ
- 24 2次キャッシュメモリ
- 30 演算器
- 31 命令コントローラ
- 32 1次キャッシュメモリ
- 33 1次キャッシュコントローラ
- 34 TLB
- 35 メモリコントローラ
- 36 PCIeコントローラ
- 37 CPUインタフェースコントローラ
- 38-1乃至38-n アドレス判断部
- 39 選択部
- 40 応答部

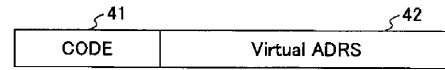
【図 1】

演算処理装置及び周辺デバイス等を含む
演算処理システムの構成の一例を示す図



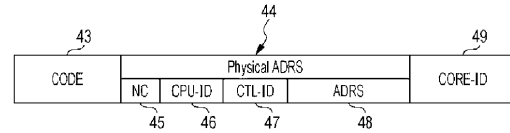
【図 2】

IU-REQのフォーマットの一例を示す図



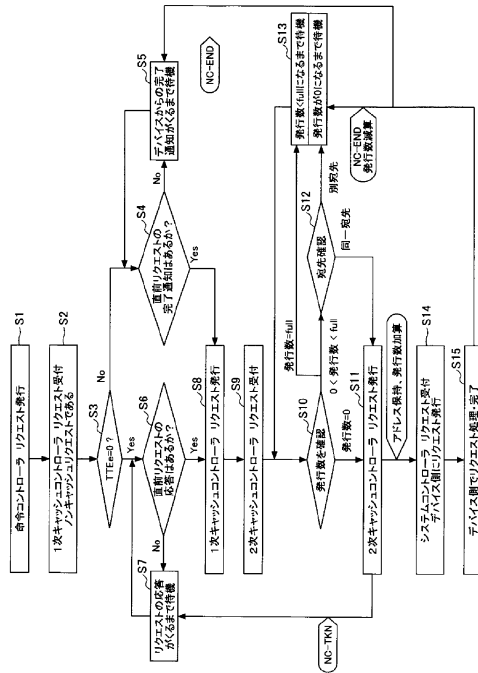
【図 3】

NC-REQのフォーマットの一例を示す図



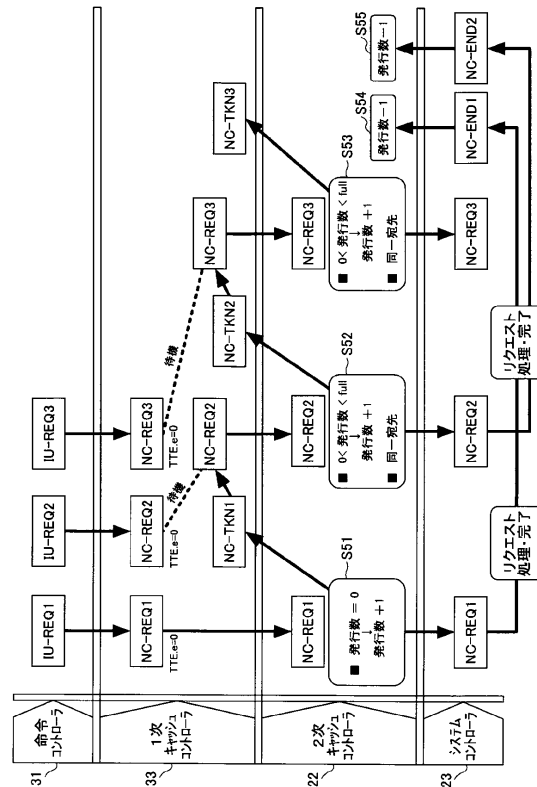
【図 4】

アクセスリクエスト発行処理の流れを示すフローチャート



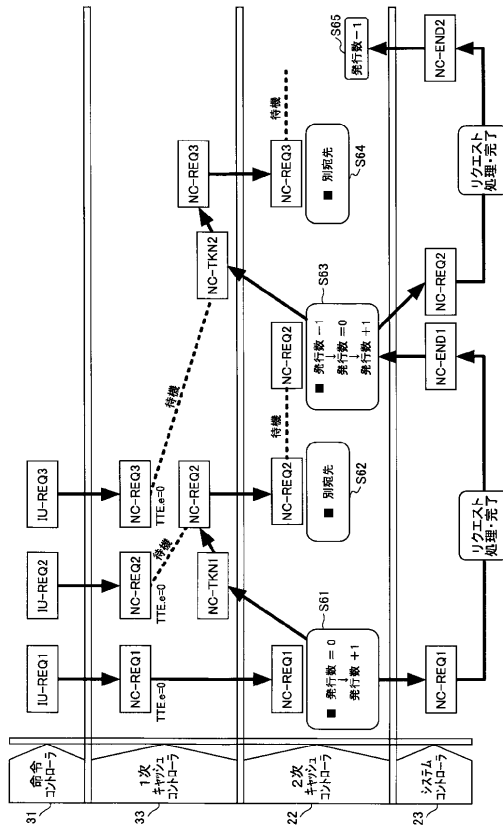
【図 5】

アクセスリクエスト発行処理の動作の一例を示す図



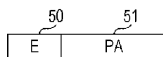
【図 6】

アクセスリクエスト発行処理の動作の別の一例を示す図



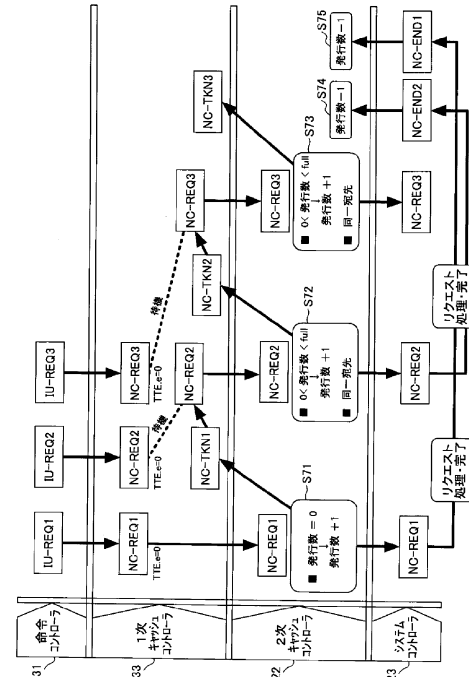
【図 8】

TLBのTTEのフォーマットを示す図



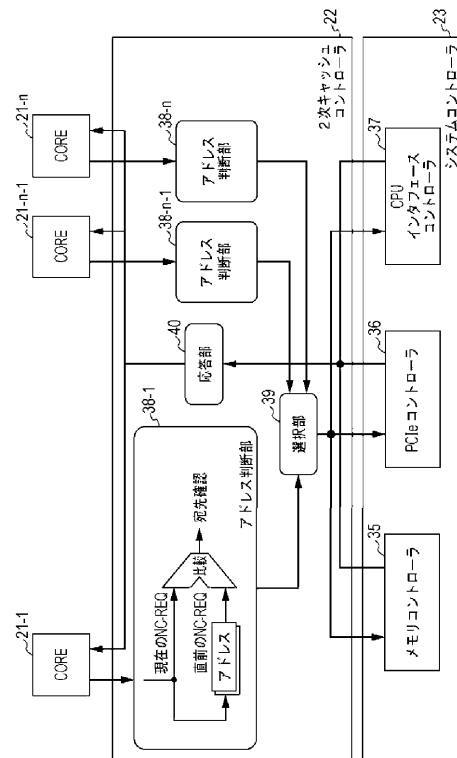
【図 7】

アクセスリクエスト発行処理の動作の更に別の一例を示す図



【図 9】

アクセスリクエスト発行処理の回路構成を示す図



フロントページの続き

(56)参考文献 国際公開第2010/032433(WO, A1)

特開2001-117859(JP, A)

米国特許第6697899(US, B1)

特開平05-265916(JP, A)

特開2001-154914(JP, A)

米国特許第5751996(US, A)

米国特許第6014737(US, A)

(58)調査した分野(Int.Cl., DB名)

G06F 12/08 - 12/10