

(12) 特許協力条約に基づいて公開された国際出願

(19) 世界知的所有権機関
国際事務局

(43) 国際公開日
2012 年 9 月 20 日(20.09.2012)



(10) 国際公開番号
WO 2012/124094 A1

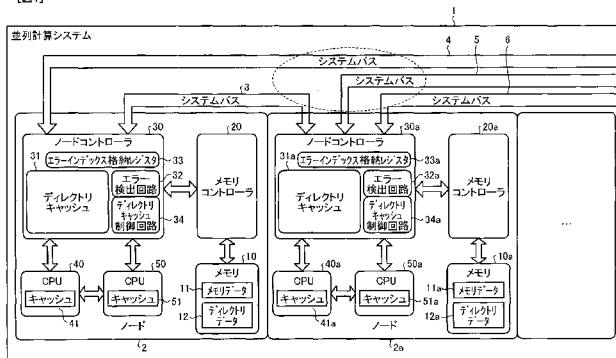
- (51) 国際特許分類:
G06F 12/08 (2006.01)
- (21) 国際出願番号: PCT/JP2011/056308
- (22) 国際出願日: 2011 年 3 月 16 日(16.03.2011)
- (25) 国際出願の言語: 日本語
- (26) 国際公開の言語: 日本語
- (71) 出願人(米国を除く全ての指定国について): 富士通株式会社(FUJITSU LIMITED) [JP/JP]; 〒2118588 神奈川県川崎市中原区上小田中 4 丁目 1 番 1 号 Kanagawa (JP).
- (72) 発明者; および
- (75) 発明者/出願人(米国についてのみ): 細川 由佳(HOSOKAWA, Yuka) [JP/JP]; 〒2118588 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 Kanagawa (JP). 畑井田 誠(HATAIDA, Makoto) [JP/JP]; 〒2118588 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 Kanagawa (JP). 石塚 孝治(ISHIZUKA, Takaharu) [JP/JP]; 〒2118588 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 Kanagawa (JP). 山本 崇史(YAMAMOTO, Takashi) [JP/JP]; 〒2118588 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 Kanagawa (JP).
- (74) 代理人: 酒井 宏明(SAKAI, Hiroaki); 〒1006020 東京都千代田区霞が関三丁目 2 番 5 号 霞が関ビルディング 酒井国際特許事務所 Tokyo (JP).
- (81) 指定国(表示のない限り、全ての種類の国内保護が可能): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PE, PG, PH, PL, PT, RO, RS, RU, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY,

[続葉有]

(54) Title: DIRECTORY CACHE CONTROL DEVICE, DIRECTORY CACHE CONTROL CIRCUIT, AND DIRECTORY CACHE CONTROL METHOD

(54) 発明の名称: ディレクトリキャッシュ制御装置、ディレクトリキャッシュ制御回路、およびディレクトリキャッシュ制御方法

[図1]



- 1 Parallel calculation system
2, 2a Node
3, 4, 5, 6 System bus
10, 10a Memory
11, 11a Memory data
12, 12a Directory data
20, 20a Memory controller
30, 30a Node controller
31, 31a Directory cache
32, 32a Error detection circuit
33, 33a Error index storage register
34, 34a Directory cache control circuit
40, 40a, 50, 50a CPU
41, 41a, 51, 51a Cache

(57) Abstract: In the present invention, a node controller (30) detects directory errors arising in a directory cache (31) that caches the directory of information recorded in a memory. Also, when an error has been detected, the node controller (30) retains the lower level address of the memory address of the memory at which information corresponding to the directory at which an error has been detected is stored. Also, when a read request for information recorded in the memory has been received, the node controller (30) determines whether or not the lower level address of the memory address that is the subject of the read request and the retained lower level address are the same. Also, when the lower level address of the memory address that is the subject of the read request and the retained lower level address have been determined to be the same, the node controller (30) controls the coherency of the information that is the subject of the read request on the basis of the directory of the information that is the subject of the read request.

(57) 要約:

[続葉有]



TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC,
VN, ZA, ZM, ZW.

- (84) 指定国 (表示のない限り、全ての種類の広域保護が可能): ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW), ユーラシア (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), ヨーロッパ (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR,

GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

添付公開書類:

— 国際調査報告 (条約第 21 条(3))

ノードコントローラ (30) は、メモリに記憶された情報のディレクトリをキャッシュするディレク
トリキャッシュ (31) において発生したディレクトリのエラーを検出する。また、ノードコントロー
ラ (30) は、エラーを検出した場合には、エラーを検出したディレクトリに対応する情報が格納され
たメモリのメモリアドレスの下位アドレスを保持する。また、ノードコントローラ (30) は、メモリ
に記憶された情報に対するリード要求を受信した場合には、リード要求の対象となるメモリアドレスの
下位アドレスと、保持した下位アドレスとが一致するか否かを判別する。そして、ノードコントローラ
(30) は、リード要求の対象となるメモリアドレスの下位アドレスと、保持した下位アドレスとが一
致しないと判別した場合には、リード要求の対象となる情報のディレクトリに基づいて、リード要求の
対象となる情報のコヒーレンスを制御する。

明 細 書

発明の名称：

ディレクトリキャッシュ制御装置、ディレクトリキャッシュ制御回路、およびディレクトリキャッシュ制御方法

技術分野

[0001] 本発明は、ディレクトリキャッシュ制御装置、ディレクトリキャッシュ制御回路、およびディレクトリキャッシュ制御方法に関する。

背景技術

[0002] 従来、キャッシュメモリを有するプロセッサとメモリとを有する複数のノードが、相互にデータを送受信する技術が知られている。このようなノードの一例として、他のノードが有するプロセッサからデータのリード要求を受信した場合には、自装置のメモリに格納されたデータを要求元のプロセッサに送信し、送信したデータを要求元のプロセッサにキャッシュさせる技術が知られている。

[0003] このようなノードは、自装置のメモリに格納されたデータと、要求元のプロセッサがキャッシュしたデータとの矛盾を防ぐ必要がある。そこで、ノードは、データをキャッシュしたプロセッサを示すディレクトリを用いて、メモリに格納されたデータとキャッシュされているデータとの同一性を保持するコヒーレンシ処理を行う。このようなコヒーレンシ処理を行うノードの一例として、メモリからディレクトリを検索する時間を短縮するためのディレクトリキャッシュを有し、ディレクトリキャッシュにキャッシュされたディレクトリを用いて、コヒーレンシ処理を行うノードが知られている。

[0004] 図6は、ディレクトリキャッシュの一例を説明するための図である。図6に示す例では、ディレクトリキャッシュが有する各キャッシュラインのうち、図6中（A）に示す範囲には、ディレクトリのデータが格納されたメモリアドレスの上位アドレスが格納されている。また、図6中（B）に示す範囲には、キャッシュ元のディレクトリデータとキャッシュしたディレクトリデ

ータとの関係を示すステータス情報が格納されている。また、図6中(C)に示す範囲には、ディレクトリデータが格納されている。なお、ディレクトリデータが格納されたメモリアドレスの上位アドレスとステータス情報とは、タグ情報として、同一キャッシュラインに格納されたディレクトリデータと対応付けられている。

[0005] このようなディレクトリキャッシュを有するノードは、ディレクトリのデータが格納されたメモリアドレスの下位アドレス（インデックス）と対応するキャッシュラインに、ディレクトリデータとタグ情報とを格納する。例えば、ノードは、図6に示す例では、インデックスが同じであって、上位アドレスが異なるメモリアドレスに格納されたディレクトリデータ「A」～「D」を、同じキャッシュラインの異なるWAYに格納する。

[0006] 以下、図7を用いて、従来のノードが実行する処理を説明する。図7は、従来のノードが実行する処理の流れを説明するためのフローチャートである。例えば、ノードは、プロセッサからデータのリード要求を受信した場合には（ステップS1）、ディレクトリが有効か否かを判別する（ステップS2）。また、ノードは、ディレクトリが有効であると判別した場合には（ステップS2肯定）、ディレクトリキャッシュからリード要求の対象であるデータのディレクトリデータを検索する（ステップS3）。

[0007] ここで、ノードは、タグ情報にエラーが発生しているか否かを判別し（ステップS4）、タグ情報にエラーが発生していないと判別した場合には（ステップS4否定）、ディレクトリがキャッシュヒットしたか否かを判別する（ステップS5）。また、ノードは、ディレクトリデータがキャッシュヒットした場合には（ステップS5肯定）、キャッシュヒットしたディレクトリデータを用いて、スヌープを発行し、データのコヒーレンスを保持する。

[0008] すなわち、ノードは、キャッシュヒットしたディレクトリデータからスヌープを発行するプロセッサを特定し、特定したプロセッサにスヌープを発行する（ステップS6）。そして、ノードは、スヌープを発行したプロセッサがキャッシュするデータと自装置のメモリに格納されたデータとの同一性を

保持した後に、データを要求元のプロセッサに送信し（ステップS 7）、処理を終了する。

[0009] また、ノードは、ディレクトリデータがキャッシュヒットしなかった場合には（ステップS 5否定）、ディレクトリデータをメモリから読み込み（ステップS 10）、ディレクトリキャッシュにディレクトリデータを格納する（ステップS 11）。その後、ノードは、スヌープを発行するプロセッサを特定し、特定したプロセッサにスヌープを発行する（ステップS 12）。

[0010] ここで、ノードは、ディレクトリキャッシュに格納されたディレクトリにエラーが発生した場合には、タグ情報を用いて、エラーが発生したディレクトリを再度キャッシュすることができる。しかし、ノードは、ディレクトリキャッシュのタグ情報においてエラーが発生した場合には、エラーが発生したタグ情報に対応付けられたディレクトリを特定することができない。例えば、ノードは、図6中（D）に示すタグ情報においてエラーが発生した場合には、同一のインデックスを有するメモリアドレスに格納されたディレクトリデータ「A」～「D」からエラーが発生したディレクトリデータを特定できない。

[0011] このため、ノードは、ディレクトリキャッシュのタグ情報においてエラーが発生した場合には（ステップS 4肯定）、ディレクトリを使用することができないと判断し、ディレクトリを無効とする（ステップS 8）。そして、ノードは、全てのプロセッサに対してスヌープを発行するブロードキャスト発行を行う（ステップS 9）。また、ノードは、他のノードから新たにリード要求を受信した場合には（ステップS 1）、ディレクトリが有効ではないと判断する（ステップS 2否定）。このため、ノードは、他のノードが有する全てのプロセッサに対してスヌープを発行する（ステップS 9）。

先行技術文献

特許文献

[0012] 特許文献1：特開平10-320279号公報

特許文献2：特許第3239935号公報

発明の概要

発明が解決しようとする課題

[0013] しかし、上述したディレクトリキャッシュのタグにエラーが発生した場合にディレクトリの使用を中止する技術では、リード要求を受信するたびに、他のノードが有する全てのプロセッサに対してスヌープを発行する。このため、ノード間の通信量が膨大となり、並列計算機システムの性能を低下させるという問題があった。

[0014] 本願に開示の技術は、上述した問題に鑑みてなされたものであって、スヌープによるノード間の通信量を削減する。

課題を解決するための手段

[0015] 1つの側面では、メモリに記憶された情報をキャッシュしている情報処理装置を示すディレクトリをキャッシュするディレクトリキャッシュを有し、ディレクトリキャッシュのエラーを検出する制御装置である。また、制御装置は、エラーを検出した場合には、エラーを検出したディレクトリに対応する情報が格納されたメモリのメモリアドレスの下位アドレスを保持する。また、制御装置は、エラーを検出した後でメモリに記憶された情報に対するリード要求を受信した場合には、リード要求の対象となるメモリアドレスの下位アドレスと、保持した下位アドレスとが一致するか否かを判別する。そして、制御装置は、リード要求の対象となるメモリアドレスの下位アドレスと、保持した下位アドレスとが一致しないと判別した場合には、リード要求の対象となる情報のディレクトリに基づいて、リード要求の対象となる情報のコヒーレンスを制御する。

発明の効果

[0016] 1つの側面では、スヌープによるノード間の通信量を削減する。

図面の簡単な説明

[0017] [図1] 図1は、実施例1に係る並列計算機システムを説明するための図である。

[図2] 図2は、ディレクトリデータを説明するための図である。

[図3] 図3は、ディレクトリキャッシュ制御回路がディレクトリデータを検索する処理を説明するための図である。

[図4] 図4は、ディレクトリキャッシュ制御回路が実行する処理について説明するための図である。

[図5] 図5は、ノードコントローラが実行する処理の流れを説明するためのフローチャートである。

[図6] 図6は、ディレクトリキャッシュの一例を説明するための図である。

[図7] 図7は、従来のノードが実行する処理の流れを説明するためのフローチャートである。

発明を実施するための形態

[0018] 以下に添付図面を参照して本願に係るディレクトリキャッシュ制御装置、ディレクトリキャッシュ制御回路、およびディレクトリキャッシュ制御方法について説明する。

実施例 1

[0019] 以下の実施例1では、図1を用いて、ディレクトリキャッシュ制御装置を有する複数のノードがシステムバスにより接続された並列計算機システムの一例を説明する。図1は、実施例1に係る並列計算機システムを説明するための図である。

[0020] 図1に示すように、並列計算機システム1は、それぞれ同様の構成を有するノード2、ノード2aを有する。また、図1では省略したが、並列計算機システム1は、ノード2と同様の構成を有する複数のノードをさらに有するものとする。また、各ノードは、システムバス3～6によって相互に接続されているものとする。なお、以下の説明では、ノード2が有する各部について説明し、他のノードについての説明を省略する。

[0021] ノード2は、メモリ10、メモリコントローラ20、ノードコントローラ30、CPU (Central Processing Unit) 40、CPU50を有する。メモリ10は、メモリデータ11、ディレクトリデー

タ 1 2 を記憶する。ノードコントローラ 3 0 は、ディレクトリキャッシュ 3 1、エラー検出回路 3 2、エラーインデックス格納レジスタ 3 3、ディレクトリキャッシュ制御回路 3 4 を有する。なお、ノードコントローラ 3 0 は、図 1 に示す各部 3 1 ~ 3 4 以外にも、ノード 2 と他のノードとの通信を制御する機能や、ノード 2 の機能を制御する回路を有するものとする。

[0022] メモリ 1 0 は、メモリデータ 1 1 とディレクトリデータ 1 2 とを記憶する。具体的には、メモリ 1 0 は、論理的に 2 つの領域に分割されている。そして、一方の領域にはリード要求の対象となるデータであるメモリデータ 1 1 が格納されており、他方の領域には、各メモリデータ 1 1 をキャッシュした CPU を示す情報であるディレクトリデータ 1 2 が格納されている。また、各ディレクトリデータ 1 2 が格納されている領域には、対応するメモリデータ 1 1 が格納されているメモリアドレスと同一のメモリアドレスが付与されている。つまり、各メモリデータ 1 1 と対応する各ディレクトリデータ 1 2 とは、同じメモリアドレスが付与された領域に格納されているものとする。

[0023] なお、以下の説明においては、メモリデータ 1 1 とディレクトリデータ 1 2 とが格納された領域には、同一のメモリアドレスが付与されるものとして説明するが、実施例は、これに限定されるものではない。

[0024] 図 2 は、ディレクトリデータを説明するための図である。図 2 に示すように、ディレクトリデータ 1 2 には、先頭から「Valid」、「Status」、「CPU-ID」を示すビットが格納されている。ここで、「Valid」には、ディレクトリデータ 1 2 のデータが有効であるか否かを示すバリッドビットが格納される。また、「Status」には、メモリ 1 0 に格納されたメモリデータ 1 1 とキャッシュされたメモリデータ 1 1 との同一性を示す情報が格納される。また、「CPU-ID」には、対応するメモリデータ 1 1 をキャッシュした CPU を示す識別子が格納される。

[0025] 例えば、「Status」には「M: Modify」、「E: Exclusive」、「S: Shared」、「I: Invalid」のいずれかが格納される。ここで、「M」は、「CPU-ID」に格納された識別子が示

すCPUが排他的にメモリデータ11をキャッシュしており、キャッシュされたメモリデータ11が書換えられた最新の状態に更新されていることを示す(d i r t y)。

[0026] また、「E」は、「CPU-ID」に格納された識別子が示すCPUが排他的にメモリデータ11をキャッシュしており、キャッシュされたメモリデータ11が書換えられていない状態を示す(C l e a n)。また、「S」は、「CPU-ID」に格納された識別子が示す複数のCPUが同一のメモリデータ11をキャッシュしている状態を示す。また、「I」は、キャッシュしてあるデータが無効であることを示す。

[0027] 図1に戻って、メモリコントローラ20は、メモリ10に格納されたメモリデータ11とディレクトリデータ12とを制御する。具体的には、メモリコントローラ20は、ノードコントローラ30からメモリアドレスを取得した場合には、取得したメモリアドレスに格納されているメモリデータ11をメモリ10から取得する。そして、メモリコントローラ20は、取得したメモリデータ11をノードコントローラ30へ送信する。

[0028] また、メモリコントローラ20は、ノードコントローラ30からディレクトリデータ12を取得する旨のメモリアドレスを取得した場合には、取得したメモリアドレスに格納されているディレクトリデータ12をメモリ10から取得する。そして、メモリコントローラ20は、取得したディレクトリデータ12をノードコントローラ30へ送信する。

[0029] また、メモリコントローラ20は、ノードコントローラ30から新たなメモリデータ11とメモリアドレスとを取得した場合には、取得したメモリアドレスに格納されているメモリデータ11を新たなメモリデータ11に更新する。また、メモリコントローラ20は、ノードコントローラ30から新たなディレクトリデータ12とメモリアドレスとを取得した場合には、取得したメモリアドレスに格納されているディレクトリデータ12を新たなディレクトリデータ12に更新する。

[0030] ノードコントローラ30は、メモリコントローラ20を介して、メモリ1

0に記憶されたディレクトリデータ12をキャッシュする。また、ノードコントローラ30は、キャッシュしたディレクトリデータ12のエラーを検出する。また、ノードコントローラ30は、エラーを検出した場合には、エラーが検出したディレクトリデータ12に対応するメモリデータ11が格納されたメモリアドレスの下位アドレス（インデックス）を保持する。

[0031] そして、ノードコントローラ30は、CPUからメモリデータのリード要求を受信した場合には、ディレクトリキャッシュ31からリード要求の対象となるメモリアドレスのインデックスを検索する。また、ノードコントローラ30は、ディレクトリキャッシュ31からインデックスを検索した結果、キャッシュミスヒットが発生した場合には、リード要求の対象となるメモリアドレスのインデックスと保持したインデックスとが一致するか否かを判別する。その後、ノードコントローラ30は、リード要求の対象となるメモリアドレスのインデックスと保持したインデックスとが一致しないと判別した場合には、以下の処理を実行する。すなわち、ノードコントローラ30は、リード要求の対象となるメモリデータ11のディレクトリデータ12に基づいて、リード要求の対象となるメモリデータの coherence を制御する。その後、ノードコントローラ30は、リード要求の対象となる情報を要求元のCPUへ送信する。

[0032] 以下、ノードコントローラ30が有する各部について説明する。ディレクトリキャッシュ31は、メモリ10に記憶された情報をキャッシュしているCPUを示すディレクトリデータ12をキャッシュする。また、ディレクトリキャッシュ31は、ディレクトリデータ12が格納されているメモリアドレスの上位アドレスと、ディレクトリデータ12の状態を示すステータス情報とをタグ情報とし、ディレクトリデータ12と対応付けてキャッシュする。

[0033] また、ディレクトリキャッシュ31は、メモリ10が有する各メモリアドレスの下位アドレスと対応付けられた複数のキャッシュラインを有する。また、ディレクトリキャッシュ31は、各キャッシュラインに複数のWAYを

有する。つまり、ディレトリキャッシュ 31 は、マルチウェイ方式のキャッシュメモリである。そして、ディレトリキャッシュ 31 は、インデックスが同じメモリアドレスに格納された複数のディレトリデータ 12 を、それぞれ同一キャッシュラインの異なるWAYに格納する。

[0034] エラー検出回路 32 は、ディレトリキャッシュ 31 において発生したエラーを検出する。例えば、エラー検出回路 32 は、ディレトリキャッシュ 31 が有するディレトリデータ 12 のうち、ディレトリキャッシュ制御回路 34 が検索対象としたキャッシュラインに格納された各WAYのエラーを検出する。そして、エラー検出回路 32 は、タグ情報において発生したエラーを検出した場合には、エラーを検出したタグ情報が格納されていたキャッシュラインのWAYをディレトリキャッシュ制御回路 34 に通知する。なお、エラー検出回路 32 は、任意の方法でエラーを検出することができる。

[0035] エラーインデックス格納レジスタ 33 は、エラー検出回路 32 がタグ情報のエラーを検出した場合には、エラーが検出されたタグ情報と対応付けられたディレトリデータ 12 が格納されたメモリアドレスのインデックスを保持する。つまり、ディレトリキャッシュ制御回路 34 は、エラーが検出されたタグ情報と対応付けられたディレトリデータ 12 が格納されているメモリアドレスのインデックスをエラーインデックス格納レジスタ 33 に格納する。

[0036] ディレトリキャッシュ制御回路 34 は、メモリデータ 11 に対するリード要求を受信した場合には、リード要求の対象となるメモリアドレスのインデックスとエラーインデックス格納レジスタ 33 に格納されたインデックスとが一致するか否かを判別する。そして、ディレトリキャッシュ制御回路 34 は、リード要求の対象となるメモリアドレスのインデックスとエラーインデックス格納レジスタ 33 に格納されたインデックスとが一致しない場合には、以下の処理を実行する。すなわち、ディレトリキャッシュ制御回路 34 は、リード要求の対象となるメモリデータ 11 と対応するディレトリ

データ 1 2 が示す CPU に対して、スヌープを発行し、リード要求の対象となるメモリデータ 1 1 のコヒーレンスを制御する。

[0037] また、ディレクトリキャッシュ制御回路 3 4 は、リード要求の対象となるメモリアドレスのインデックスとエラーインデックス格納レジスタ 3 3 に格納されたインデックスとが一致する場合には以下の処理を実行する。すなわち、並列計算機システム 1 が有する全ての CPU に対して、スヌープをブロードキャスト発行する。

[0038] その後、ディレクトリキャッシュ制御回路 3 4 は、スヌープを発行した結果に応じて、リード要求の対象となるメモリデータ 1 1 のコヒーレンスを制御する。また、ディレクトリキャッシュ制御回路 3 4 は、スヌープをブロードキャスト発行することにより取得した結果をディレクトリキャッシュ 3 1 にキャッシュする。

[0039] また、ディレクトリキャッシュ制御回路 3 4 は、再度リード要求を受信した場合には、ディレクトリキャッシュ 3 1 に、ブロードキャストしたスヌープの結果がキャッシュされているか否かを判別する。そして、ディレクトリキャッシュ制御回路 3 4 は、ブロードキャストしたスヌープの結果がキャッシュされていると判別した場合には、キャッシュされていたスヌープの結果が示す CPU にスヌープを発行する。

[0040] すなわち、ディレクトリキャッシュ 3 1 には、エラーが発生していないキャッシュラインに格納されているディレクトリデータ 1 2 については、メモリ 1 0 からキャッシュ前のディレクトリデータを特定することが可能である。このため、ノードコントローラ 3 0 は、エラーが発生していないキャッシュラインに格納されているディレクトリデータ 1 2 については、信頼することができるディレクトリデータ 1 2 であると判別する。

[0041] そこで、ディレクトリキャッシュ制御回路 3 4 は、エラーが発生したタグ情報と対応付けられたディレクトリデータ 1 2 のインデックス、つまり、エラーが発生したキャッシュラインに対応するインデックスをエラーインデックス格納レジスタ 3 3 に格納する。そして、ディレクトリキャッシュ 3 1 は

、新たに受信したリード要求の対象となるメモリアドレスのインデックスが、エラーインデックス格納レジスタ 33 に格納されたインデックスと一致しない場合には、ディレトリデータ 12 を用いてスヌープを送信する。

[0042] すなわち、ディレトリキャッシュ制御回路 34 は、メモリ 10 に格納されたディレトリデータ 12 又はディレトリキャッシュ 31 に格納されたディレトリデータ 12 が示す CPU に対してのみスヌープを送信する。このため、ノードコントローラ 30 は、リード要求が連続して発行される場合にも、スヌープをブロードキャスト発行しないので、ノード間の通信量を削減することができる。この結果、並列計算機システム 1 は、効率良く処理を進めることができる。

[0043] さらに、ディレトリキャッシュ制御回路 34 は、スヌープをブロードキャスト発行した結果を、ディレトリキャッシュ 31 に格納する。そして、ディレトリキャッシュ制御回路 34 は、リード要求のインデックスがエラーインデックス格納レジスタ 33 に格納されたインデックスと一致した場合には、以下の処理を実行する。

[0044] すなわち、ディレトリキャッシュ制御回路 34 は、リード要求の対象となるメモリアドレスのメモリデータ 11 に関してブロードキャスト発行されたスヌープの結果をディレトリキャッシュ 31 から検索する。そして、ディレトリキャッシュ制御回路 34 は、スヌープの結果が検索された場合には、このスヌープの結果が示す CPU のみに対してスヌープを実行する。

[0045] このため、ディレトリキャッシュ制御回路 34 は、エラーが発生したためにディレトリデータ 12 を用いることができないメモリデータ 11 に対するリード要求が連続した場合にも、スヌープのブロードキャスト発行を防ぎ、ノード間の通信量を削減する。この結果、並列計算機システム 1 は、効率よく処理をすすめることができる。

[0046] なお、ノードコントローラ 30 は、スヌープを発行した結果、メモリ 10 に格納されたメモリデータ 11 を他のノードが有する CPU にキャッシュされたメモリデータ 11 に更新する場合には、以下の処理を実行する。すなわ

ち、ノードコントローラ 30 は、新たなメモリアドレスと新たなメモリデータ 11 をメモリコントローラ 20 に送信する。また、ノードコントローラ 30 は、スヌープを発行した結果、ディレクトリデータ 12 に変更があった場合には、メモリアドレスと変更後のディレクトリデータ 12 をメモリコントローラ 20 に送信する。

[0047] また、ノードコントローラ 30 は、メモリデータ 11 をメモリ 10 から取得する場合には、メモリデータ 11 が格納されたメモリアドレスをメモリコントローラ 20 に送信する。また、ノードコントローラ 30 は、ディレクトリデータ 12 をメモリ 10 から取得する場合には、ディレクトリデータ 12 が格納されたメモリアドレスとディレクトリデータ 12 を取得する旨をメモリコントローラ 20 に送信する。

[0048] CPU 40 は、メモリ 10、10a に格納されたメモリデータ 11、11a を用いて処理を実行する情報処理装置である。ここで、CPU 40 は、キャッシュ 41 を有し、メモリ 10、10a に格納されたメモリデータ 11、11a をキャッシュする。また、CPU 40 は、ノードコントローラ 30 が他のノード 2a からスヌープのブロードキャストを取得した場合には、自装置がノード 2a のメモリ 10a からメモリデータ 11a をキャッシュしているか否かを判別する。そして、CPU 40 は、自装置がメモリデータ 11a をキャッシュしていると判別した場合には、トランザクションをノード 2a へ送信し、自装置がキャッシュしたメモリデータ 11a とメモリ 10a に格納されたメモリデータ 11a との同一性を保持する。

[0049] CPU 50 は、キャッシュ 51 を有し、メモリ 10、10a に格納されたメモリデータ 11、11a を用いて処理を実行する情報処理装置である。なお、CPU 50 は、CPU 40 と同様の機能を有するものとし、説明を省略する。

[0050] 例えば、エラー検出回路 32、ディレクトリキャッシュ制御回路 34 とは、電子回路である。ここで、電子回路の例として、ASIC (Application Specific Integrated Circuit) や FPGA (Field Programmable Gate Array) などが挙げられる。

e Array)などの集積回路、またはCPU (Central Processing Unit) や MPU (Micro Processing Unit)などを適用する。

[0051] また、ディレクトリキャッシュ31とは、RAM (Random Access Memory)、フラッシュメモリ (flash memory)などの半導体メモリ素子などの記憶装置である。また、エラーインデックス格納レジスタ33は、レジスタである。

[0052] 次に、図3を用いて、ディレクトリキャッシュ制御回路34がディレクトリキャッシュ31からキャッシュデータを検索する処理について説明する。図3は、ディレクトリキャッシュ制御回路がディレクトリデータを検索する処理を説明するための図である。なお、図3中(A)の範囲には、ディレクトリデータ12が格納されているメモリアドレスの上位アドレスが格納されている。また、図3中(B)の範囲には、ディレクトリデータ12の状態を示す情報が格納されている。また、図3中(C)の範囲には、ディレクトリデータ12が格納されている。なお、ディレクトリデータ12が格納されているメモリアドレスの上位アドレスと、ディレクトリデータ12の状態を示す情報とは、タグ情報として、ディレクトリデータ12と対応付けて記憶されている。

[0053] また、図3中の(D)に示すように、ディレクトリキャッシュ31は、マルチウェイ方式のキャッシュメモリである。また、1つのキャッシュラインに含まれる各WAYには、同一のインデックスを有するメモリアドレスに格納された複数のディレクトリデータが格納される。例えば、図3に示す例では、それぞれ同じインデックスを有するメモリアドレスに格納された複数のディレクトリデータ「A」～「D」が同一のキャッシュラインに含まれるWAYに格納される。

[0054] このため、図3中の(E)に示すタグ情報においてエラーが検出された場合には、ディレクトリデータ「A」～「D」のうち、どのデータについてエラーが発生したのか判別することができなくなる。しかし、図3中の(E)に示すタグ情報においてエラーが発生した場合にも、他のキャッシュライン

に格納されているディレクトリデータ 12 は、信用しても良いデータであると考えられる。

[0055] そこで、ディレクトリキャッシュ制御回路 34 は、タグ情報にエラーが発生したキャッシュラインのインデックスをエラーインデックス格納レジスタ 33 に格納する。そして、ディレクトリキャッシュ制御回路 34 は、リード要求の対象となるメモリアドレスのインデックスがエラーインデックス格納レジスタ 33 に格納されたインデックスと一致した場合には、ディレクトリデータ 12 を信用しない。つまり、ディレクトリキャッシュ制御回路 34 は、リード要求の対象のメモリアドレスが有するインデックスがエラーしたインデックスである場合には、ディレクトリデータ 12 を使用せずに、スヌープをブロードキャスト発行する。

[0056] 一方、ディレクトリキャッシュ制御回路 34 は、リード要求の対象となるメモリアドレスのインデックスがエラーしたインデックスと異なる場合には、リード要求の対象となるメモリアドレスのディレクトリデータ 12 を信用する。つまり、ディレクトリキャッシュ制御回路 34 は、リード要求の対象となるメモリアドレスのインデックスがエラーインデックス格納レジスタ 33 に格納されたインデックスと一致しない場合には、ディレクトリデータ 12 が示す CPU にスヌープを発行する。

[0057] 次に、図 4 を用いて、ディレクトリキャッシュ制御回路 34 が実行する処理の流れの一例を説明する。図 4 は、ディレクトリキャッシュ制御回路が実行する処理について説明するための図である。例えば、図 4 中 (A) に示すようにディレクトリキャッシュ制御回路 34 は、システムバス 3 を介して、ノード 2a が有する CPU 40a からリード要求を受信する。このような場合には、図 4 中 (B) に示すように、ディレクトリキャッシュ制御回路 34 は、リード要求の対象となるメモリアドレスと同じメモリアドレスに格納されたディレクトリデータ 12 をディレクトリキャッシュ 31 から検索する。

[0058] ここで、ディレクトリキャッシュ制御回路 34 は、ディレクトリキャッシュ 31 からディレクトリデータ 12 を検索する場合には、受信したリード要

求の対象となるメモリアドレスを特定し、特定したメモリアドレスから上位アドレスとインデックスとを判別する。そして、ディレクトリキャッシュ制御回路34は、判別したインデックスと対応付けられたキャッシュラインをディレクトリキャッシュ31から選択し、選択したキャッシュラインの各WAYに格納されているディレクトリデータ12とタグ情報とを取得する。また、ディレクトリキャッシュ制御回路34は、取得した各WAYに格納されていた各タグ情報の上位アドレスと、リード要求の対象となるメモリアドレスから判別した上位アドレスとを比較する。

[0059] そして、ディレクトリキャッシュ制御回路34は、リード要求の対象となるメモリアドレスから判別した上位アドレスと同じ上位アドレスが格納されているタグ情報が格納されたWAYを選択する。そして、ディレクトリキャッシュ制御回路34は、選択したWAYにキャッシュされたディレクトリデータ12を取得する。一方、ディレクトリキャッシュ制御回路34は、リード要求の対象となるメモリアドレスから判別した上位アドレスと同じ上位アドレスが格納されているタグ情報が格納されたWAYが存在しない場合には、キャッシュミスヒットが発生したと判別する。

[0060] ここで、エラー検出回路32は、図4中(C)に示すように、ディレクトリキャッシュ制御回路34がディレクトリキャッシュ31から選択したキャッシュラインの各WAYにおいてエラーが発生しているか否かを判別する。そして、図4中(D)に示すように、エラー検出回路32は、いずれかのWAYにおいてエラーを検出した場合には、エラーを検出したWAYを示す情報をディレクトリキャッシュ制御回路34へ送信する。そして、ディレクトリキャッシュ制御回路34は、エラー検出回路32から取得した情報に基づいて、エラー検出の有無、および、エラーが発生したWAYを特定する。

[0061] また、ディレクトリキャッシュ制御回路34は、キャッシュヒットし、かつ、エラーが検出されなかった場合には、以下の処理を実行する。すなわち、ディレクトリキャッシュ制御回路34は、図4中(E)に示すように、キャッシュヒットしたディレクトリデータ12に格納されている「S t a t u

s」を確認する。その後、ディレクトリキャッシュ制御回路34は、「Status」が「M」である場合には、図4中（E）に示すように、キャッシュヒットしたディレクトリデータ12の「CPU-ID」が示すCPUに対してスヌープを発行する。

[0062] また、図4中（G）に示すように、ディレクトリキャッシュ制御回路34は、スヌープを発行した後、スヌープを発行したCPUから最新のメモリデータ11を受信する。そして、図4中（H）に示すように、ディレクトリキャッシュ制御回路34は、受信したメモリデータ11をリード要求元のCPUへ送信する。また、ディレクトリキャッシュ制御回路34は、メモリ10に格納されているメモリデータ11を更新する。また、ディレクトリキャッシュ制御回路34は、キャッシュヒットしたディレクトリデータ12とメモリ10に格納されているディレクトリデータ12との「Status」を更新する。

[0063] また、ディレクトリキャッシュ制御回路34は、キャッシュヒットし、かつ、キャッシュヒットしたWAY以外でタグ情報のエラーが発生している場合には、キャッシュヒットしたディレクトリデータ12を信用することができない。このため、ディレクトリキャッシュ制御回路34は、図4中（F）に示すように、選択したキャッシュラインに対応するインデックス（以下、エラーしたインデックスと記載）をエラーインデックス格納レジスタ33に格納する。そして、ディレクトリキャッシュ制御回路34は、リード要求の対象となるメモリデータ11のスヌープ要求をブロードキャスト発行する。

[0064] ここで、ディレクトリキャッシュ制御回路34は、スヌープ要求をブロードキャスト発行した結果、リード要求の対象となるメモリデータ11をキャッシュするCPUを判別した場合には、判別したCPUとの間でコヒーレンシ処理を実行する。そして、ディレクトリキャッシュ制御回路34は、メモリデータ11を更新するとともに、更新したメモリデータ11を要求元のプロセッサへ送信する。また、ディレクトリキャッシュ制御回路34は、スヌープをブロードキャスト発行した結果を、リード要求の対象となるメモリア

ドレスの上位アドレスとともに、ディレクトリキャッシュ 31 に格納する。

[0065] また、ディレクトリキャッシュ制御回路 34 は、キャッシュミスヒットした場合には、リード要求の対象となるメモリアドレスのインデックスとエラーインデックス格納レジスタ 33 に格納されているインデックスとが一致するか否かを判別する。そして、ディレクトリキャッシュ制御回路 34 は、エラー要求の対象となるメモリアドレスのインデックスとエラーインデックス格納レジスタ 33 に格納されているインデックスとが一致した場合には、以下の処理を実行する。

[0066] すなわち、ディレクトリキャッシュ制御回路 34 は、選択したキャッシュラインの各WAYからリード要求の対象となるメモリアドレスの上位アドレスに一致するアドレスがタグ情報に格納されたWAYを検索する。そして、ディレクトリキャッシュ制御回路 34 は、リード要求の対象となるメモリアドレスの上位アドレスに一致するアドレスがタグ情報に格納されたWAYに格納されたスヌープをブロードキャストした結果が示すCPUのみに対して、スヌープを発行する。また、ディレクトリキャッシュ制御回路 34 は、リード要求の対象となるメモリアドレスの上位アドレスに一致するアドレスがタグ情報に格納されたWAYが存在しない場合には、スヌープ要求をブロードキャスト発行する。

[0067] 一方、ディレクトリキャッシュ制御回路 34 は、リード要求の対象となるメモリアドレスのインデックスがエラーインデックス格納レジスタ 33 に格納されているインデックスと一致しない場合には、以下の処理を実行する。すなわち、ディレクトリキャッシュ制御回路 34 は、エラー検出の有無を判別し、エラーが検出されていない場合には、ディレクトリデータ 12 をメモリ 10 からディレクトリキャッシュ 31 にキャッシュさせる。そして、ディレクトリキャッシュ制御回路 34 は、キャッシュしたディレクトリデータ 12 が示すCPUのみにスヌープを発行する。

[0068] 一方、ディレクトリキャッシュ制御回路 34 は、エラーが検出されている場合には、エラーしたインデックスをエラーインデックス格納レジスタ 33

に格納する。また、ディレトリキャッシュ制御回路 34 は、スヌープ要求をブロードキャスト発行する。

[0069] このような各部 31～34 を有するノードコントローラ 30 は、ディレトリキャッシュ 31 が記憶するタグ情報にエラーが発生した場合には、エラーしたインデックスを保持する。そして、ノードコントローラ 30 は、リード要求の対象メモリアドレスのインデックスが保持したインデックスと異なる場合には、ディレトリキャッシュ 31、または、メモリ 10 に格納されたディレトリデータ 12 が示す CPU のみにスヌープを発行する。

[0070] このため、ノードコントローラ 30 は、リード要求を受信するたびにスヌープをブロードキャスト発行せずとも、適切なコヒーレンシ処理を実行することができる。この結果、ノードコントローラ 30 は、ノード間の通信量を抑え、並列計算機システム 1 の性能を向上させることができる。

[0071] また、ノードコントローラ 30 は、スヌープをブロードキャスト発行した場合には、スヌープ結果をディレトリキャッシュ 31 にキャッシュさせる。その後、ノードコントローラ 30 は、リード要求の対象メモリアドレスのインデックスが保持したインデックスと同じである場合には、ディレトリキャッシュ 31 からディレトリデータ 12a を検索する。そして、ノードコントローラ 30 は、ディレトリデータ 12a が示す CPU のみにスヌープを発行する。

[0072] このため、ノードコントローラ 30 は、タグ情報にエラーが発生したキャッシュラインと対応するメモリアドレスに格納されたメモリデータ 11 に対するリード要求が繰り返し発行された場合にも、特定の CPU のみにスヌープを発行することとなる。この結果、ノードコントローラ 30 は、さらにノード間の通信量を削減し、並列計算機システム 1 の性能を向上させることができる。

[0073] なお、ブロードキャスト発行したスヌープの結果をディレトリキャッシュ 31 に格納する方法は、任意の方法を用いることができるが、この実施例においては、スヌープ対象となるメモリアドレスの上位アドレスと、スヌー

プの結果とを対応付けてディレクトリキャッシュ 31 に格納するものとする。

[0074] [ノードコントローラ 30 の処理の流れ]

次に、図 5 を用いて、ノードコントローラ 30 が実行する処理の流れについて説明する。図 5 は、ノードコントローラが実行する処理の流れを説明するためのフローチャートである。例えば、ノードコントローラ 30 は、他のノードが有する CPU からリード要求を受信したことを契機として、処理を開始する（ステップ S 101）。

[0075] 次に、ノードコントローラ 30 は、リード要求の対象となるメモリデータ 11 のディレクトリデータ 12 をディレクトリキャッシュ 31 から検索する（ステップ S 102）。そして、ノードコントローラ 30 は、キャッシュヒットしたか否かを判別し（ステップ S 103）、キャッシュヒットした場合には（ステップ S 103 肯定）、ヒットした WAY 以外でタグ情報にエラーが検出されたか否かを判別する（ステップ S 104）。

[0076] 次に、ノードコントローラ 30 は、ヒットした WAY 以外でタグ情報にエラーが検出されなかった場合には（ステップ S 104 否定）、キャッシュヒットしたディレクトリデータ 12 が示す CPU にスヌープを発行する（ステップ S 105）。そして、ノードコントローラ 30 は、スヌープを発行し、コヒーレンシ処理を実行することによって同一性を保持したメモリデータ 11 を要求元の CPU へ送信し（ステップ S 106）、処理を終了する。

[0077] 一方、ノードコントローラ 30 は、ヒットした WAY 以外でタグ情報にエラーが検出された場合には（ステップ S 104 肯定）、エラーしたインデックスをエラーインデックス格納レジスタ 33 に保持する（ステップ S 107）。また、ノードコントローラ 30 は、スヌープ要求をブロードキャスト発行し（ステップ S 108）、スヌープ結果をディレクトリキャッシュ 31 に格納する（ステップ S 109）。

[0078] また、ノードコントローラ 30 は、キャッシュヒットしなかった場合には（ステップ S 103 否定）、以下の処理を実行する。すなわち、ノードコン

トローラ 30 は、リード要求対象インデックスとエラーインデックスとが一致するか否かを判別する（ステップ S 110）。そして、ノードコントローラ 30 は、リード要求対象インデックスとエラーインデックスとが一致すると判別した場合には（ステップ S 110 肯定）以下の処理を実行する。すなわち、ノードコントローラ 30 は、ディレトリキャッシュ 31 からリード要求の対象となるメモリアドレスの上位アドレスがディレトリキャッシュ 31 からヒットするか否かを判別する（ステップ S 111）。

[0079] そして、ノードコントローラ 30 は、上位アドレスがディレトリキャッシュ 31 からヒットした場合には（ステップ S 111 肯定）、上位アドレスと対応付けてディレトリキャッシュ 31 に保持していたブロードキャストしたスヌープの結果が示す CPU にスヌープを発行する（ステップ S 112）。その後、ノードコントローラ 30 は、スヌープを発行し、コヒーレンシ処理を実行することによって同一性を保持したメモリデータ 11 を要求元の CPU へ送信し（ステップ S 106）、処理を終了する。

[0080] また、ノードコントローラ 30 は、上位アドレスがディレトリキャッシュ 31 からヒットしなかった場合には（ステップ S 111 否定）、スヌープ要求をブロードキャスト発行する（ステップ S 108）。そして、ノードコントローラ 30 は、スヌープの結果をディレトリキャッシュ 31 に格納する（ステップ S 109）。

[0081] また、ノードコントローラ 30 は、リード要求対象インデックスがエラーインデックスと一致しない場合には（ステップ S 110 否定）、タグ情報にエラーが検出されたか否かを判別する（ステップ S 113）。そして、ノードコントローラ 30 は、タグ情報にエラーが検出された場合には（ステップ S 113 肯定）、エラーしたインデックスをエラーインデックス格納レジスタ 33 に格納する（ステップ S 107）。一方、ノードコントローラ 30 は、タグ情報からエラーが検出されなかった場合には（ステップ S 113 否定）、メモリ 10 からディレトリデータ 12 をリードする（ステップ S 114）。そして、ノードコントローラ 30 は、リードしたディレトリデータ

12をディレクトリキャッシュ31に格納する（ステップS115）。その後、ノードコントローラ30は、ディレクトリキャッシュ31に格納されたディレクトリデータ12が示すCPUにスヌープを発行する（ステップS105）。

[0082] [実施例1の効果]

上述したように、ノードコントローラ30は、ディレクトリキャッシュ31におけるタグ情報のエラーを検出した場合には、エラーしたインデックスをエラーインデックス格納レジスタ33に格納する。また、ノードコントローラ30は、リード要求を取得した場合には、リード要求対象インデックスとエラーインデックス格納レジスタ33に格納されたインデックスとが一致するか否かを判別する。そして、ノードコントローラ30は、リード要求対象インデックスとエラーしたインデックスとが一致しない場合には、リード要求の対象となるメモリデータ11に対応するディレクトリデータ12に基づいて、メモリデータ11のコヒーレンスを制御する。

[0083] このため、ノードコントローラ30は、エラーしたインデックス以外のインデックスを含むメモリアドレスを対象とするリード要求を受信した場合には、ディレクトリデータ12を用いてコヒーレンスを制御する。つまり、ノードコントローラ30は、スヌープをブロードキャスト発行せずとも、ディレクトリデータ12に基づいて、コヒーレンスを制御することができる。この結果、ノードコントローラ30は、ノード間の通信量を削減し、並列計算機システム1の性能を向上させることができる。

[0084] また、ノードコントローラ30は、リード要求対象インデックスとエラーインデックス格納レジスタ33に格納されたインデックスとが一致しない場合には、ディレクトリデータ12が示すCPUに対してスヌープを発行する。このため、ノードコントローラ30は、リード要求を受信した際にスヌープをブロードキャストせずともメモリデータ11のコヒーレンスを適切に保持することができるので、ノード間の通信量を削減し、並列計算機システム1の性能を向上させることができる。

- [0085] また、ノードコントローラ 30 は、スヌープをブロードキャスト発行した場合には、スヌープの結果をディレクトリキャッシュ 31 に格納する。また、ノードコントローラ 30 は、再度受信したリード要求の対象インデックスとエラーインデックス格納レジスタ 33 に格納されたインデックスとが一致する場合には、ディレクトリキャッシュ 31 にスヌープをブロードキャスト発行した結果が格納されているか否かを判別する。そして、ノードコントローラ 30 は、スヌープをブロードキャストした結果が格納されていると判別した場合には、このスヌープ結果が示す CPU に対してスヌープを発行する。つまり、ノードコントローラ 30 は、信用できないディレクトリデータ 12 を使用せずに、ディレクトリキャッシュ 31 に格納しておいたスヌープの結果を用いて、コヒーレンシ処理を実行する。
- [0086] このため、ノードコントローラ 30 は、エラーが発生したタグ情報が格納されているキャッシュラインと対応付けられたメモリアドレスに対するリード要求を連続して受信した場合にも、スヌープをブロードキャスト発行することなく、コヒーレンシを保持できる。この結果、ノードコントローラ 30 は、ノード間の通信量を削減し、並列計算機システム 1 の性能を向上させることができる。
- [0087] 一方、ノードコントローラ 30 は、再度受信したリード要求の対象インデックスとエラーインデックス格納レジスタ 33 に格納されたインデックスとが一致する場合には、ディレクトリキャッシュ 31 からリード要求の対象となるメモリアドレスを検索する。そして、ノードコントローラ 30 は、ディレクトリキャッシュ 31 にリード要求対象となるメモリアドレスのスヌープ結果がキャッシュされていない場合には、並列計算機システム 1 が有する全ての CPU にスヌープをブロードキャスト発行する。
- [0088] つまり、ノードコントローラ 30 は、新たなリード要求の対処となるメモリアドレスがエラーしたインデックスを含み、かつ、ディレクトリキャッシュ 31 にスヌープ結果がキャッシュされていない場合には、スヌープをブロードキャスト発行する。このため、ノードコントローラ 30 は、タグ情報に

エラーが発生したキャッシュラインに格納される信用できないディレクトリデータ 12 を使用しない。このため、ノードコントローラ 30 は、適切にコヒーレンシ処理をおこなうことができる。

- [0089] また、ノードコントローラ 30 は、ディレクトリデータ 12 と対応するメモリデータ 11 が格納されているメモリアドレスの下位アドレスをタグ情報としてディレクトリキャッシュ 31 にキャッシュする。そして、ノードコントローラ 30 は、発生したエラーを検出する。このため、ノードコントローラ 30 は、ディレクトリキャッシュ 31 において発生したエラーのうち、メモリ 10 に格納されたディレクトリデータ 12 によって回復させることができないエラーが発生した場合にのみ、スヌープをブロードキャスト発行する。この結果、ノードコントローラ 30 は、ノード間の通信を押さえ、並列計算機システム 1 の性能を向上させることができる。

実施例 2

- [0090] これまで本発明の実施例について説明したが、上述した実施例以外にも様々な異なる形態にて実施されてよい。そこで、以下では実施例 2 として本発明に含まれる他の実施例を説明する。

- [0091] (1) 並列計算機システムが有するノードについて

上述したノード 2 は、2 つの CPU 40、50 を有していたが、実施例はこれに限定されるものではなく、任意の数の CPU を有することとしてもよい。また、上述した並列計算機システム 1 は、ノード 2 と同様の構成を有するノード 2a や他のノードを有するものとしたが、実施例はこれに限定されるものではない。例えば、各ノードは、ノードコントローラ 30 が実行する処理を同様の処理を実行する構成を有する限りは、それぞれ任意の構成を有してもよい。

- [0092] (2) ディレクトリデータについて

上述したディレクトリデータ 12 は、「Valid」、「Status」、「CPU-ID」が格納されたデータであるとしたが、実施例はこれに限定されるものではない。すなわち、ディレクトリデータ 12 は、対応するメ

メモリデータ 11 をキャッシュした CPU を示す情報と、キャッシュされたメモリデータ 11 とメモリ 10 に格納されたメモリデータ 11 との関係を示すステータス情報とが格納されていればよい。

[0093] また、上述した実施例 1 では、「S t a t u s」に格納される情報として、イリノイプロトコルによるステータス情報が格納されていた。しかし、実施例はこれに限定されるものではなく、任意のプロトコルによるステータス情報を格納することができる。

[0094] また、上述したディレクトリキャッシュ制御回路 34 は、スヌープをブロードキャスト発行した結果、スヌープ結果をディレクトリキャッシュ 31 に格納していた。しかし、実施例はこれに限定されるものではない。

[0095] 例えば、ノードコントローラ 30 は、エラーが発生したディレクトリを有するメモリアドレスに格納されたディレクトリデータ 12 をキャッシュする補助メモリをさらに有する。そして、ディレクトリキャッシュ制御回路 34 は、スヌープをブロードキャスト発行したスヌープ結果を補助メモリに格納する。その後、ディレクトリキャッシュ制御回路 34 は、リード要求の対象インデックスがエラーインデックス格納レジスタ 33 に格納されたインデックスと一致した場合には、補助メモリに格納されたスヌープ結果を用いて、スヌープを発行してもよい。

[0096] (3) ノードコントローラの処理について

上述したノードコントローラ 30 は、スヌープのブロードキャスト発行を極力行わないようにするため、リード要求対象インデックスとエラーインデックス格納レジスタ 33 に格納されたインデックスとが一致した場合には、以下の処理を実行した。すなわち、ノードコントローラ 30 は、スヌープをブロードキャスト発行したスヌープ結果がディレクトリキャッシュ 31 にキャッシュされているか否かを判別する。そして、ノードコントローラ 30 は、スヌープ結果がディレクトリキャッシュ 31 にキャッシュされている場合には、キャッシュされたディレクトリデータ 12 が示す CPU のみにスヌープを発行した。

[0097] しかし、実施例は、これに限定されるものではない。例えば、ノードコントローラ 30 は、リード要求対象インデックスとエラーインデックス格納レジスタ 33 に格納されたインデックスとが一致した場合には、即時スヌープをブロードキャスト発行しても良い。このような処理を行うことによって、ノードコントローラ 30 の構成を容易にすることができる。

[0098] なお、ディレトリキャッシュ制御回路 34 は、スヌープ結果をディレトリキャッシュ 31 にキャッシュする場合には、エラーが発生したキャッシュラインにスヌープ結果を格納することとなる。しかし、実施例はこれに限定されるものではない。例えば、ディレトリキャッシュ制御回路 34 は、スヌープ結果をエラーが発生したキャッシュラインの異なる WAY に格納しても良いし、異なるキャッシュラインに格納しても良い。

[0099] (4) メモリアドレスについて

実施例 1 においては、各メモリデータ 11 と各ディレトリデータ 12 とは、同じメモリアドレスが付与されているものとした。しかし、実施例は、これに限定されるものではない。例えば、メモリデータ 11 と対応するディレトリデータ 12 とが異なるメモリアドレスが付与されている場合には、ディレトリキャッシュ制御回路 34 は、メモリデータ 11 が格納されたメモリアドレスと、対応するディレトリデータ 12 が格納されたメモリアドレス（以下、ディレトリアドレスと記載する。）とを対応付けて記憶する。

[0100] また、ディレトリキャッシュ制御回路 34 は、ディレトリデータ 12 をディレトリキャッシュ 31 が有するキャッシュラインのうち、ディレトリアドレスに応じたキャッシュラインに格納する。また、ディレトリキャッシュ制御回路 34 は、ディレトリキャッシュ 31 からエラーが検出された場合には、エラーが検出されたキャッシュラインに関わるディレトリアドレスのインデックスをエラーインデックス格納レジスタ 33 に格納する。

[0101] そして、ディレトリキャッシュ制御回路 34 は、リード要求を受信した

場合には、リード要求が示すメモリアドレスと対応付けて記憶されたディレクトリアドレスを検索する。その後、ディレクトリキャッシュ制御回路 34 は、検索したディレクトリアドレスのインデックスがエラーインデックス格納レジスタ 33 に格納されているか否かを判別する。そして、ディレクトリキャッシュ制御回路 34 は、検索したディレクトリアドレスのインデックスがエラーインデックス格納レジスタ 33 に格納されている場合には、ディレクトリキャッシュを使用せずに、スヌープをブロードキャストする。

[0102] なお、メモリアドレスと対応するディレクトリアドレスを対応付けて記憶し、変換を行う処理は、メモリコントローラ 20 に実行させるものとしてもよい。このような処理をメモリコントローラ 20 に実行させることにより、ディレクトリキャッシュ制御回路 34 は、リード要求のメモリアドレスのみを用いてメモリデータ 11 とディレクトリデータ 12 との要求をメモリコントローラ 20 に要求するだけで、対応するメモリデータ 11 とディレクトリデータ 12 とを取得することができる。この結果、ディレクトリキャッシュ制御回路 34 は、容易に実装を行うことができる。

[0103] このように、ディレクトリキャッシュ制御回路 34 は、メモリアドレスとディレクトリアドレスとが同一ではない場合にも、適切に処理を実行し、各ノード間の通信効率を改善させ、並列計算機システム 1 の効率を改善することができる。

符号の説明

- [0104] 1 並列計算機システム
 2、2a ノード
 3、4、5、6、 システムバス
 10、10a メモリ
 11、11a メモリデータ
 12、12a ディレクトリデータ
 20、20a メモリコントローラ
 30、30a ノードコントローラ

- 31、31a ディレクトリキャッシュ
- 32、32a エラー検出回路
- 33、33a エラーインデックス格納レジスタ
- 34、34a ディレクトリキャッシュ制御回路
- 40、40a、50、50a CPU
- 41、41a、51、51a キャッシュ

請求の範囲

- [請求項1] メモリに記憶された情報をキャッシュしている情報処理装置を示すディレクトリをキャッシュするキャッシュ部と、
- 前記キャッシュ部におけるディレクトリのエラーを検出する検出部と、
- 前記検出部がエラーを検出した場合には、当該エラーが検出されたディレクトリに対応する情報が格納された前記メモリのメモリアドレスを保持する保持部と、
- 前記メモリに記憶された情報に対するリード要求を受信した場合には、当該リード要求の対象となるメモリアドレスと、前記保持部が保持するアドレスとが一致するか否かを判別する判別部と、
- 前記判別部が前記リード要求の対象となるメモリアドレスと、前記保持部が保持するアドレスとが一致しないと判別した場合に、当該リード要求の対象となる情報のディレクトリに基づいて、当該リード要求の対象となる情報のコヒーレンスを制御する制御部と
- を有することを特徴とするディレクトリキャッシュ制御装置。
- [請求項2] 前記保持部は、前記メモリアドレスの下位アドレスを保持し、
- 前記判別部は、受信したリード要求の対象となるメモリアドレスの下位アドレスと、前記保持部が保持する下位アドレスとが一致するか否かを判別することを特徴とする請求項1に記載のディレクトリキャッシュ制御装置。
- [請求項3] 前記ディレクトリは、対応する情報をキャッシュしている情報処理装置を示し、
- 前記制御部は、前記判別部が前記リード要求の対象となるメモリアドレスの下位アドレスと、前記保持部が保持する下位アドレスとが一致しないと判別した場合には、当該リード要求の対象となる情報のディレクトリが示す情報処理装置に対してスヌープを発行することを特徴とする請求項2に記載のディレクトリキャッシュ制御装置。

[請求項4] 前記制御部は、前記判別部が前記リード要求の対象となるメモリアドレスの下位アドレスと、前記保持部が保持する下位アドレスとが一致すると判別した場合には、前記メモリに記憶された情報をキャッシュすることができる全ての情報処理装置に対してスヌープを発行し、当該スヌープを発行した結果を、前記キャッシュ部にキャッシュすることを特徴とする請求項3に記載のディレクトリキャッシュ制御装置。

[請求項5] 前記制御部は、前記判別部が前記リード要求の対象となるメモリアドレスの下位アドレスと、前記保持部が保持した下位アドレスとが一致すると判別した場合には、前記全ての情報処理装置に対してスヌープを発行した結果が前記キャッシュ部に格納されているか否かを判別し、スヌープを発行した結果が前記キャッシュ部に格納されていた場合には、当該スヌープを発行した結果が示す情報処理装置に対してスヌープを発行することを特徴とする請求項4に記載のディレクトリキャッシュ制御装置。

[請求項6] 前記制御部は、前記判別部が前記リード要求の対象となるメモリアドレスの下位アドレスと、前記保持部が保持した下位アドレスとが一致すると判別した場合には、前記全ての情報処理装置に対してスヌープを発行した結果を前記キャッシュ部から検索し、当該スヌープを発行した結果が前記キャッシュ部にキャッシュされていないと判別した場合には、前記メモリに記憶された情報をキャッシュすることができる全ての情報処理装置に対してスヌープを発行することを特徴とする請求項4に記載のディレクトリキャッシュ制御装置。

[請求項7] 前記キャッシュ部は、キャッシュ元のディレクトリと対応する情報が格納されているメモリアドレスの下位アドレスを有するタグ情報を、当該ディレクトリと対応付けてキャッシュし、

前記検出部は、前記タグ情報において発生したエラーを検出することを特徴とする請求項1～6に記載のディレクトリキャッシュ制御装

置。

[請求項8]

メモリに記憶された情報をキャッシュしている情報処理装置を示すディレクトリをキャッシュするキャッシュ部と、

前記キャッシュ部において発生したディレクトリのエラーを検出する検出部と、

前記検出部によってタグ情報のエラーが検出された場合には、当該エラーが検出されたディレクトリに対応する情報が格納された前記メモリのメモリアドレスのうち、所定の下位アドレスを保持する保持部と、

前記メモリに記憶された情報に対するリード要求を受信した場合には、当該リード要求の対象となるメモリアドレスの下位アドレスと、前記保持部が保持した下位アドレスとが一致するか否かを判別する判別部と、

前記判別部が前記リード要求の対象となるメモリアドレスの下位アドレスと、前記保持部が保持した下位アドレスとが一致しないと判別した場合には、当該リード要求の対象となる情報のディレクトリに基づいて、当該リード要求の対象となる情報のコヒーレンスを制御する制御部と

を有することを特徴とするディレクトリキャッシュ制御回路。

[請求項9]

メモリに記憶された情報をキャッシュしている情報処理装置を示すディレクトリをキャッシュするキャッシュ装置を有するディレクトリキャッシュ制御装置によって実行されるディレクトリキャッシュ制御方法であって、

前記キャッシュ装置において発生したディレクトリのエラーを検出し、

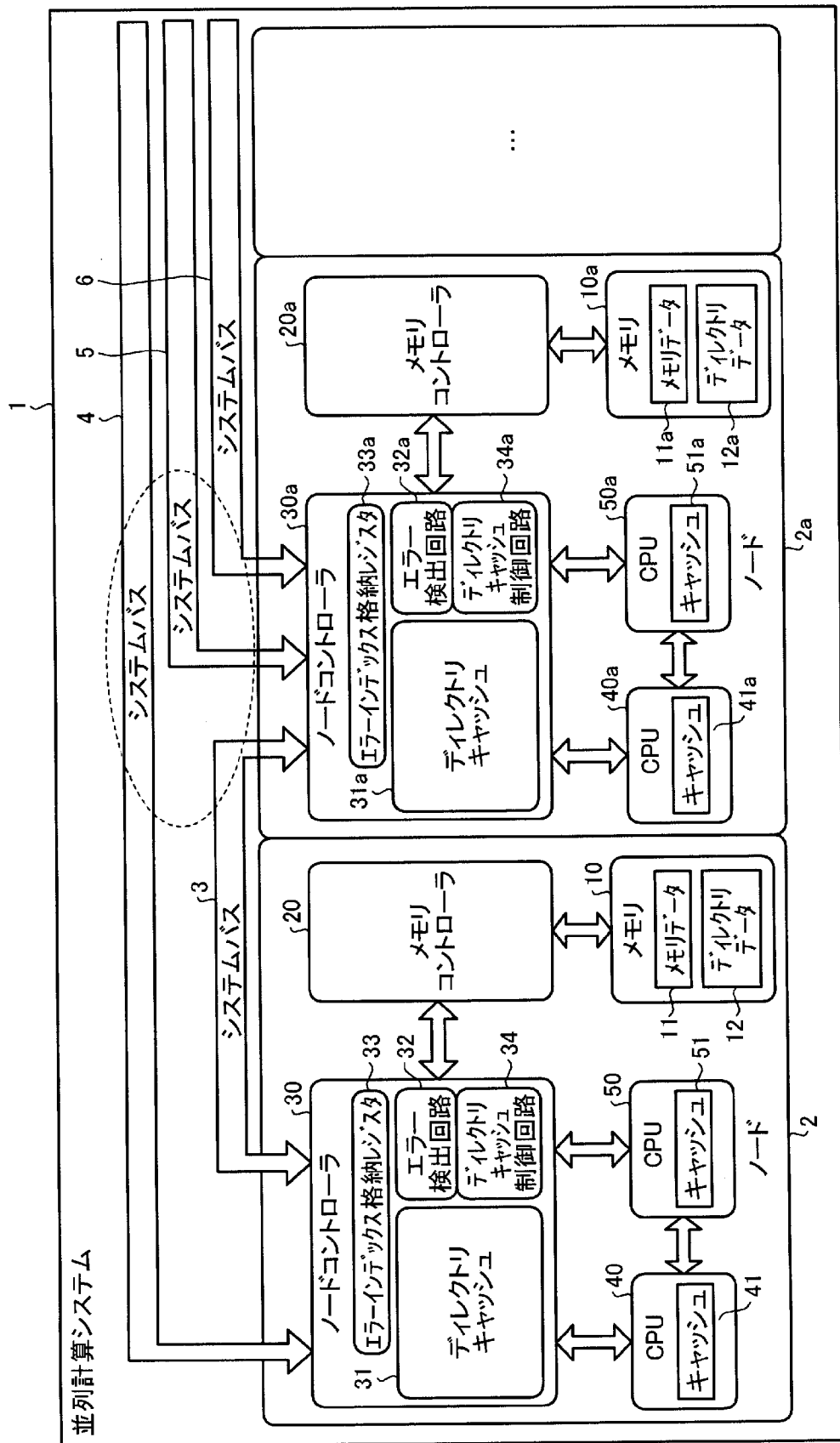
前記エラーがタグ情報のエラーであった場合には、当該エラーを検出したディレクトリに対応する情報が格納された前記メモリのメモリアドレスのうち、所定の下位アドレスを保持し、

前記メモリに記憶された情報に対するリード要求を他のコンピュータから受信した場合には、当該リード要求の対象となるメモリアドレスの下位アドレスと、前記保持した下位アドレスとが一致するか否かを判別し、

前記リード要求の対象となるメモリアドレスの下位アドレスと、前記保持した下位アドレスとが一致しないと判別した場合には、当該リード要求の対象となる情報のディレクトリに基づいて、当該リード要求の対象となる情報のコヒーレンスを制御する

ことを特徴とするディレクトリキャッシュ制御方法。

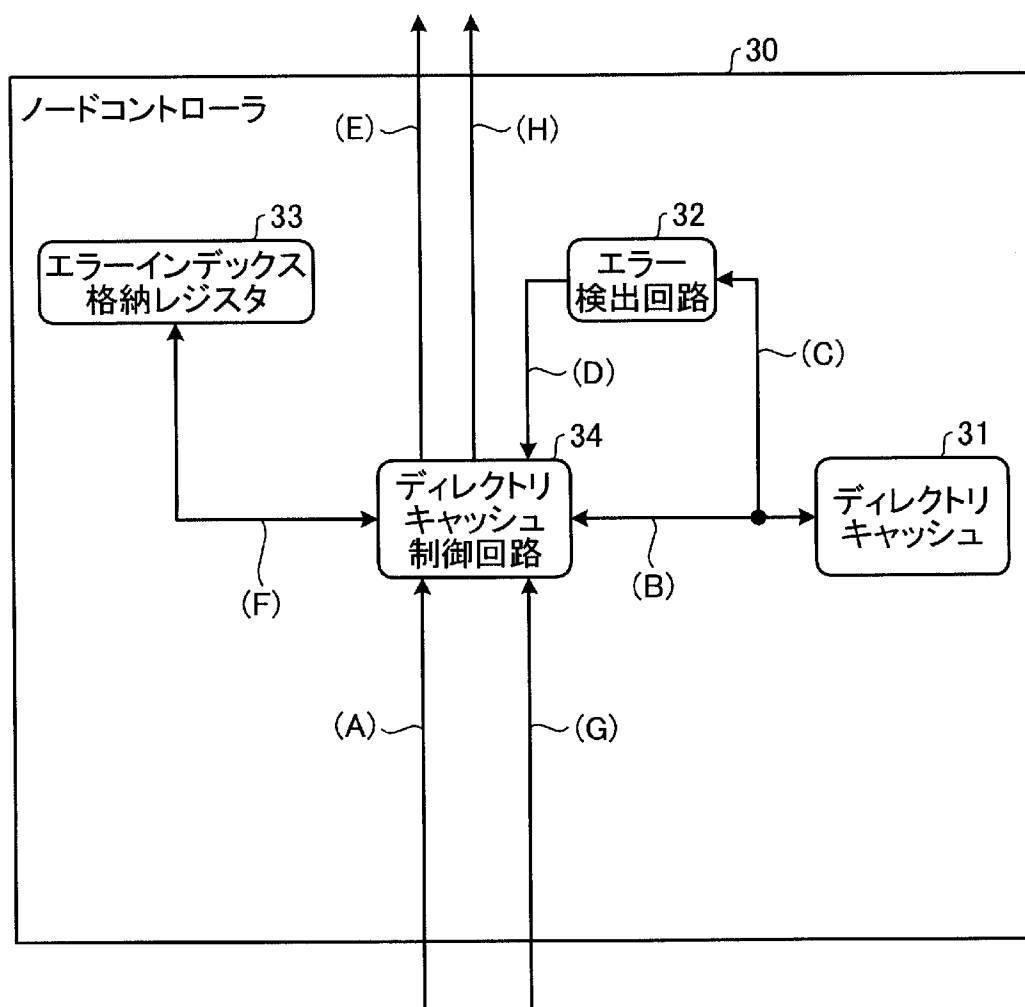
[図1]



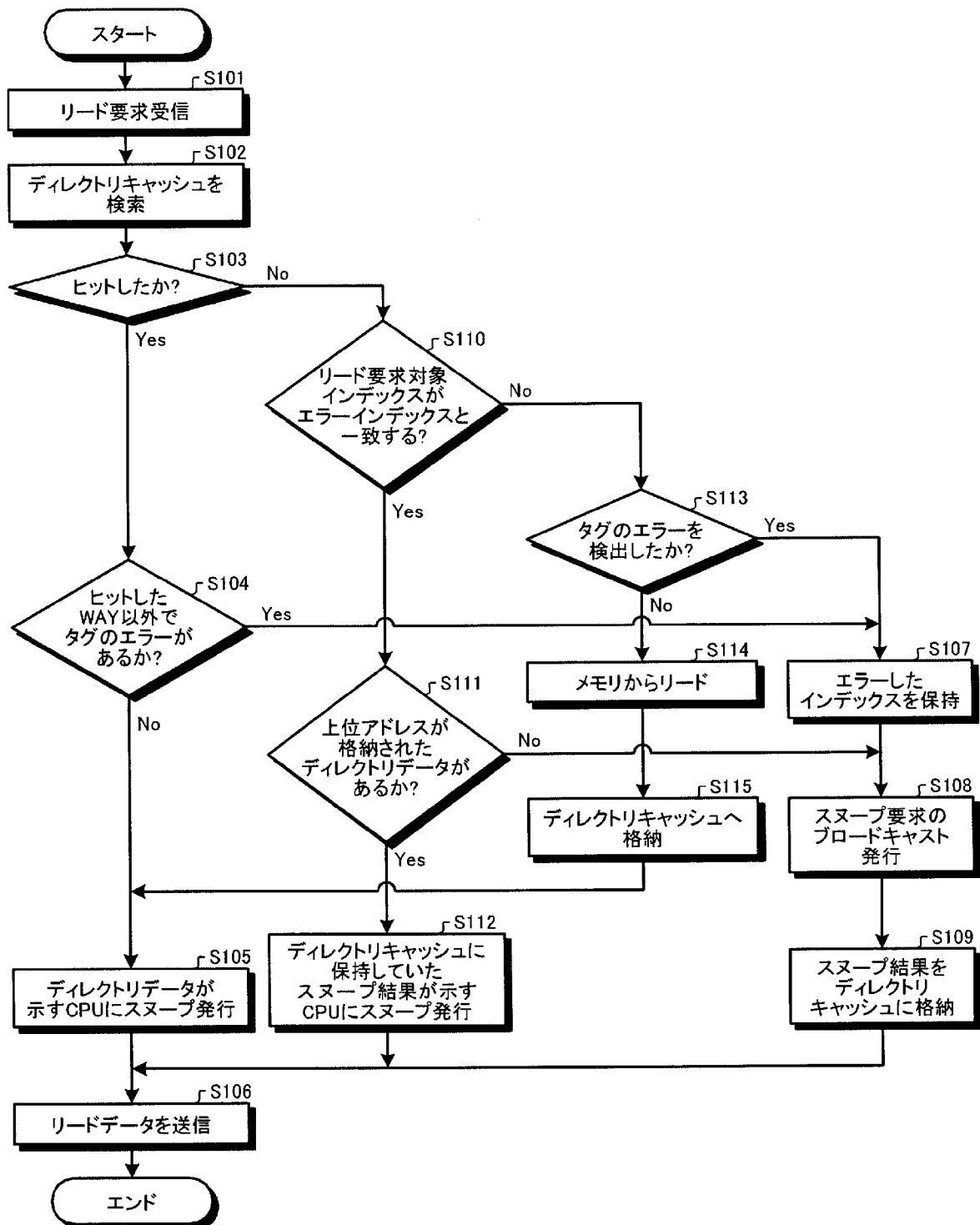
[図2]

Valid	Status	CPU-ID	CPU-ID	...
-------	--------	--------	--------	-----

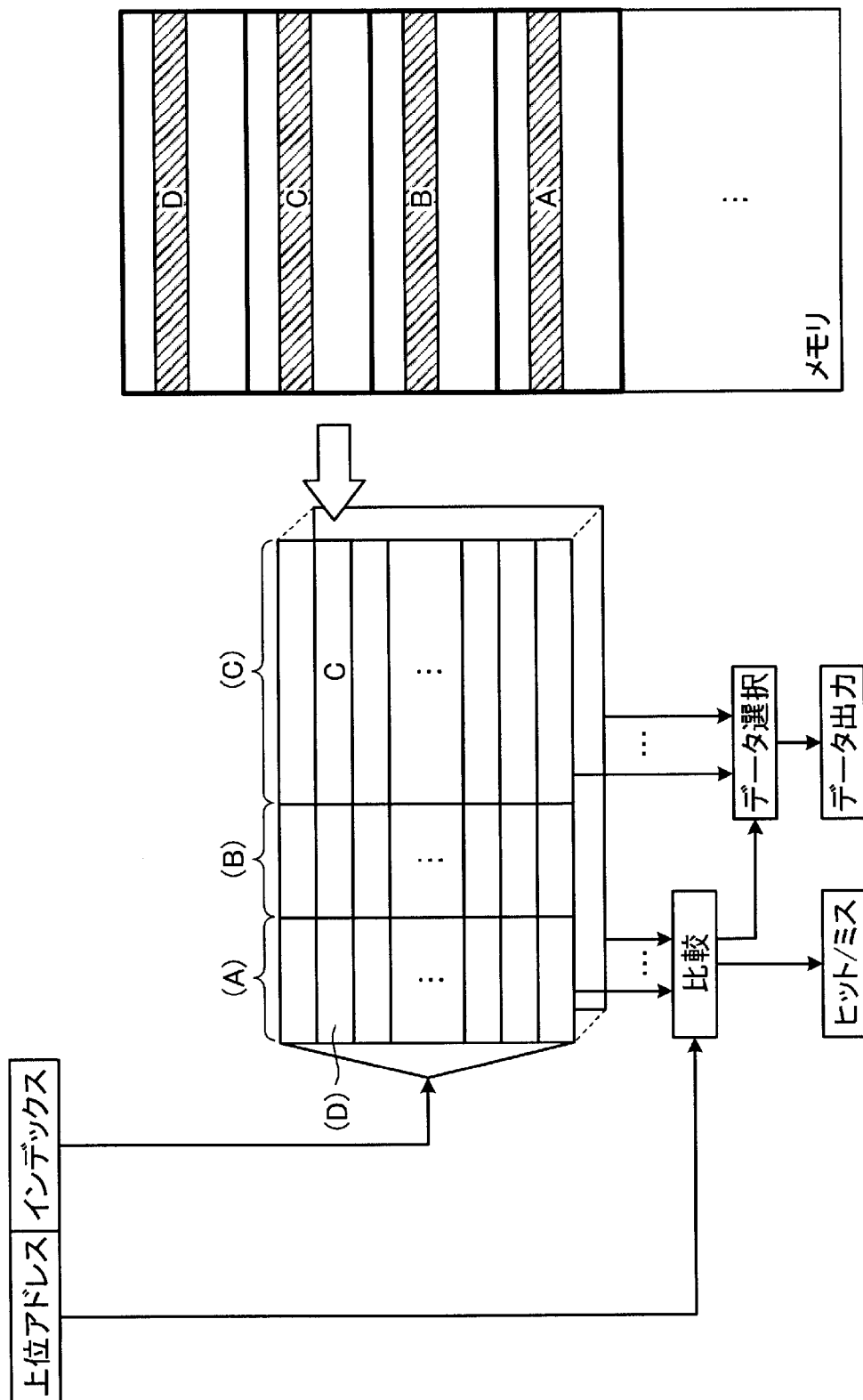
[図4]



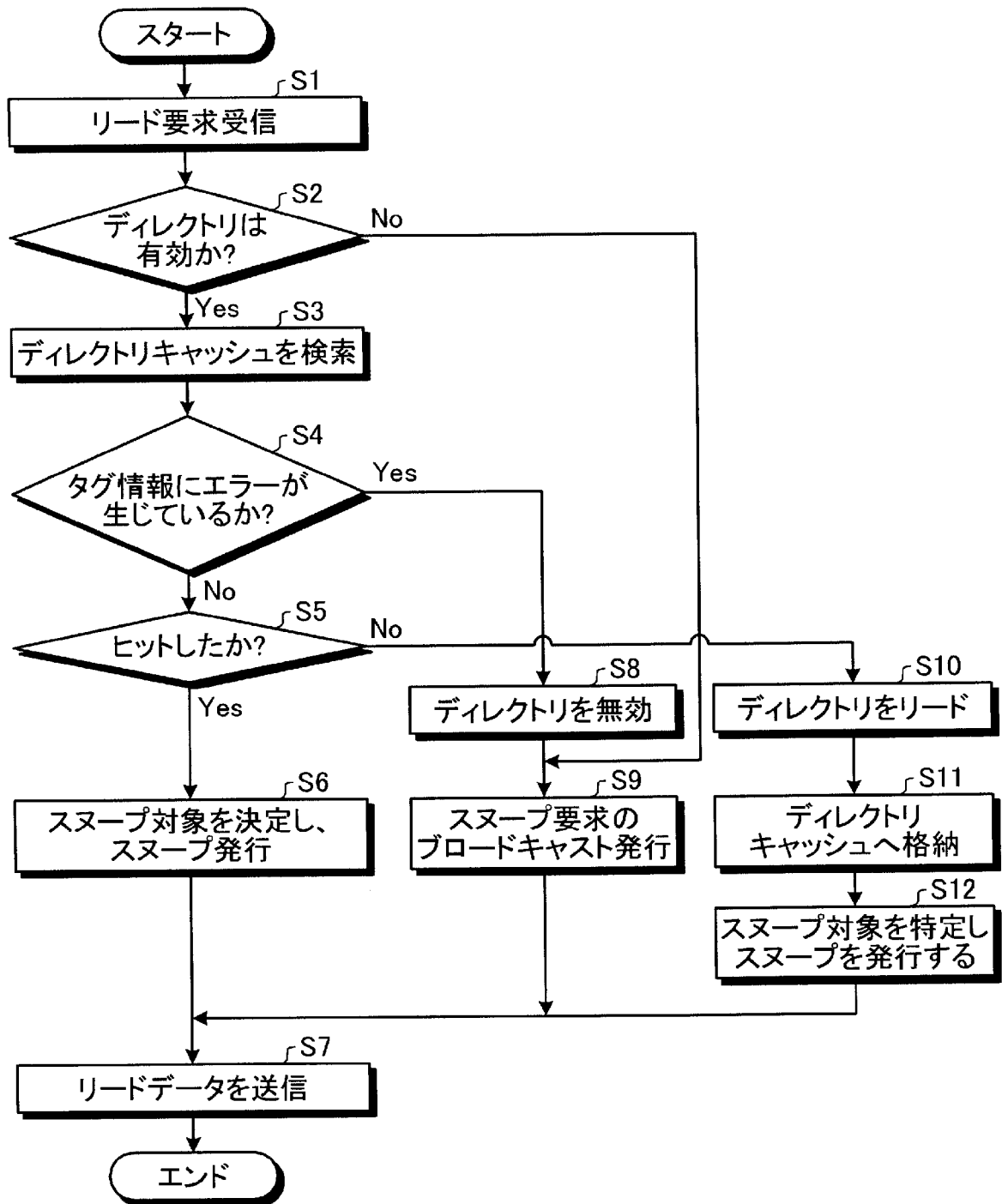
[図5]



[図6]



[図7]



INTERNATIONAL SEARCH REPORT

International application No.

PCT/JP2011/056308

A. CLASSIFICATION OF SUBJECT MATTER

G06F12/08 (2006.01) i

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

G06F12/08-12/12

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Jitsuyo Shinan Koho	1922-1996	Jitsuyo Shinan Toroku Koho	1996-2011
Kokai Jitsuyo Shinan Koho	1971-2011	Toroku Jitsuyo Shinan Koho	1994-2011

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y A	JP 2003-162447 A (Sun Microsystems, Inc.), 06 June 2003 (06.06.2003), entire text; all drawings & US 2003/0084369 A1 & EP 1310873 A1 & KR 10-2003-0033109 A	1-3, 7-9 4-6
Y A	JP 7-182238 A (SGS-Thomson Microelectronics, Inc.), 21 July 1995 (21.07.1995), entire text; all drawings & US 5708789 A	1-3, 7-9 4-6
A	JP 2008-234531 A (NEC Computertechno, Ltd.), 02 October 2008 (02.10.2008), entire text; all drawings (Family: none)	1-9



Further documents are listed in the continuation of Box C.



See patent family annex.

* Special categories of cited documents:

“A” document defining the general state of the art which is not considered to be of particular relevance

“E” earlier application or patent but published on or after the international filing date

“L” document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

“O” document referring to an oral disclosure, use, exhibition or other means

“P” document published prior to the international filing date but later than the priority date claimed

“T” later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

“X” document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

“Y” document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

“&” document member of the same patent family

Date of the actual completion of the international search
12 April, 2011 (12.04.11)

Date of mailing of the international search report
26 April, 2011 (26.04.11)

Name and mailing address of the ISA/
Japanese Patent Office

Authorized officer

Facsimile No.

Telephone No.

A. 発明の属する分野の分類 (国際特許分類 (I P C))

Int.Cl. G06F12/08 (2006.01) i

B. 調査を行った分野

調査を行った最小限資料 (国際特許分類 (I P C))

Int.Cl. G06F12/08-12/12

最小限資料以外の資料で調査を行った分野に含まれるもの

日本国実用新案公報	1 9 2 2 - 1 9 9 6 年
日本国公開実用新案公報	1 9 7 1 - 2 0 1 1 年
日本国実用新案登録公報	1 9 9 6 - 2 0 1 1 年
日本国登録実用新案公報	1 9 9 4 - 2 0 1 1 年

国際調査で使用した電子データベース (データベースの名称、調査に使用した用語)

C. 関連すると認められる文献

引用文献の カテゴリー*	引用文献名 及び一部の箇所が関連するときは、その関連する箇所の表示	関連する 請求項の番号
Y A	JP 2003-162447 A (サン・マイクロシステムズ・インコーポレイテッド) 2003.06.06, 全文, 全図 & US 2003/0084369 A1 & EP 1310873 A1 & KR 10-2003-0033109 A	1-3, 7-9 4-6
Y A	JP 7-182238 A (エスジーエーストムソン マイクロエレクトロニクス, インコーポレイテッド) 1995.07.21, 全文, 全図 & US 5708789 A	1-3, 7-9 4-6

☒ C欄の続きにも文献が列挙されている。☐ パテントファミリーに関する別紙を参照。

* 引用文献のカテゴリー

「A」 特に関連のある文献ではなく、一般的技術水準を示すもの

「E」 国際出願日前の出願または特許であるが、国際出願日以後に公表されたもの

「L」 優先権主張に疑義を提起する文献又は他の文献の発行日若しくは他の特別な理由を確立するために引用する文献 (理由を付す)

「O」 口頭による開示、使用、展示等に言及する文献

「P」 国際出願日前で、かつ優先権の主張の基礎となる出願

の日の後に公表された文献

「T」 国際出願日又は優先日後に公表された文献であって出願と矛盾するものではなく、発明の原理又は理論の理解のために引用するもの

「X」 特に関連のある文献であって、当該文献のみで発明の新規性又は進歩性がないと考えられるもの

「Y」 特に関連のある文献であって、当該文献と他の1以上の文献との、当業者にとって自明である組合せによって進歩性がないと考えられるもの

「&」 同一パテントファミリー文献

国際調査を完了した日

1 2 . 0 4 . 2 0 1 1

国際調査報告の発送日

2 6 . 0 4 . 2 0 1 1

国際調査機関の名称及びあて先

日本国特許庁 (I S A / J P)

郵便番号 1 0 0 - 8 9 1 5

東京都千代田区霞が関三丁目4番3号

特許庁審査官 (権限のある職員)

野田 佳邦

5 M

3 4 5 0

電話番号 0 3 - 3 5 8 1 - 1 1 0 1 内線 3 5 9 9

C (続き) . 関連すると認められる文献		
引用文献の カテゴリー*	引用文献名 及び一部の箇所が関連するときは、その関連する箇所の表示	関連する 請求項の番号
A	JP 2008-234531 A (エヌイーシーコンピュータテクノ株式会社) 2008.10.02, 全文, 全図 (ファミリーなし)	1-9