

(19) 日本国特許庁(JP)

(12) 公開特許公報(A)

(11) 特許出願公開番号

特開2014-186397

(P2014-186397A)

(43) 公開日 平成26年10月2日(2014.10.2)

(51) Int.Cl. F I テーマコード (参考)
G06F 12/08 (2006.01) G06F 12/08 507E 5B005
 G06F 12/08 511E

審査請求 未請求 請求項の数 7 O L (全 19 頁)

(21) 出願番号 特願2013-59117(P2013-59117)
 (22) 出願日 平成25年3月21日(2013.3.21)

(71) 出願人 00005223
 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 100089118
 弁理士 酒井 宏明
 (72) 発明者 園領 琢也
 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
 Fターム(参考) 5B005 MM01 NN22 NN64 TT02

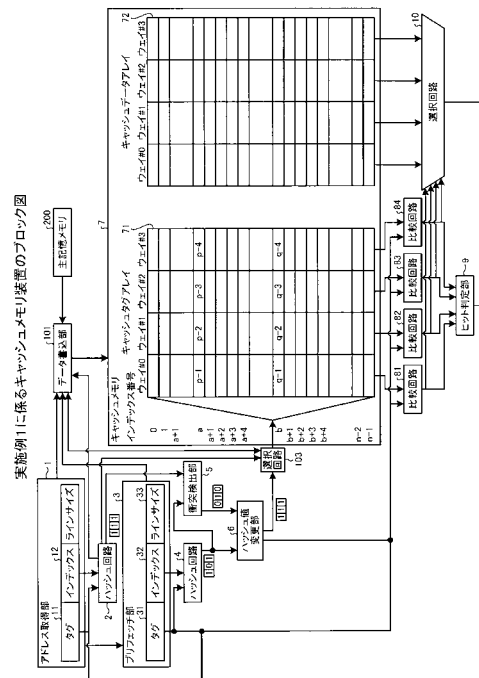
(54) 【発明の名称】 キャッシュメモリ装置、情報処理装置及びキャッシュメモリ制御方法

(57) 【要約】

【課題】 キャッシュメモリのハッシュ衝突を容易に発生させるキャッシュメモリ装置を提供する。

【解決手段】 ハッシュ回路2は、第1アドレスからキャッシュメモリ7のインデックスを示す第1インデックスをハッシュ関数により求め、第1インデックスが示すインデックスにタグを格納する。プリフェッチ部3は、第1アドレスとタグが異なる第2アドレスを生成する。ハッシュ回路4は、第2アドレスから前記キャッシュメモリ7のインデックスを示す第2インデックスをハッシュ関数により求める。衝突検出部5は、第1インデックスと第2インデックスとが一致するかどうかを判定する。一致しない場合、ハッシュ値変更部6は、第2インデックスを第1インデックスと一致するように変更し、キャッシュメモリ7における変更した第2インデックスが示すインデックスで、且つ第1アドレスのタグを格納したウェイとは異なる列にタグを格納する。

【選択図】 図1



【特許請求の範囲】

【請求項 1】

複数のウェイを有するキャッシュメモリと、
 前記主記憶メモリのアドレスである第 1 アドレスを取得する第 1 アドレス取得部と、
 前記第 1 アドレスから前記キャッシュメモリのインデックスを示す第 1 インデックスをハッシュ関数により求め、前記キャッシュメモリにおける前記第 1 インデックスが示すインデックスに前記第 1 アドレスのタグを格納する第 1 インデックス管理部と、
 前記第 1 アドレスから前記第 1 アドレスとタグが異なる第 2 アドレスを生成する第 2 アドレス生成部と、
 前記第 2 アドレスから前記キャッシュメモリのインデックスを示す第 2 インデックスを前記ハッシュ関数により求める第 2 インデックス算出部と、
 前記第 1 インデックスと前記第 2 インデックスとが一致するか否かを判定する判定部と、
 前記判定部により一致しないと判定された場合、前記第 2 インデックスを前記第 1 インデックスと一致するように変更し、前記キャッシュメモリにおける変更した前記第 2 インデックスが示すインデックスで、且つ前記第 1 アドレスのタグを格納したウェイとは異なるウェイに前記第 2 アドレスのタグを格納する第 2 インデックス管理部と、
 を備えたことを特徴とするキャッシュメモリ装置。

【請求項 2】

前記第 1 インデックス管理部及び前記第 2 インデックス算出部は、前記ハッシュ関数として排他的論理和を取る関数を用い、
 前記判定部は、前記第 1 インデックスと前記第 2 インデックスの排他的論理和を求め、
 前記第 2 インデックス管理部は、前記第 2 インデックスと前記判定部が求めた排他的論理和との排他的論理和を求め、前記第 2 インデックスを求めた排他的論理和に変更することを特徴とする請求項 1 に記載のキャッシュメモリ装置。

【請求項 3】

前記第 2 アドレス生成部は、前記第 1 アドレスに対する事前先読みを行い、前記第 2 インデックスを生成することを特徴とする請求項 1 又は 2 に記載のキャッシュメモリ装置。

【請求項 4】

前記第 2 アドレス生成部は、前記第 1 アドレスのタグを 1 つずつインクリメントしていくことで前記第 1 アドレスに対する事前先読みを行い、前記第 2 インデックスを生成することを特徴とする請求項 3 に記載のキャッシュメモリ装置。

【請求項 5】

前記第 1 インデックス管理部は、前記列毎に予め決められたハッシュ関数を用いて、前記列のそれぞれ対応する前記第 1 インデックスを求め、且つ求めた第 1 インデックスのいずれか一つが示すインデックスに、該インデックスに対応する第 1 アドレスのタグを格納し、

前記第 2 インデックス算出部は、前記ウェイ毎に予め決められたハッシュ関数を用いて、第 2 アドレスを前記ウェイ毎に順次求め、

前記判定部は、各前記第 2 インデックスと、各前記第 2 インデックスに対応するウェイに対応するハッシュ関数を用いて求められた前記第 1 インデックスとが一致するか否かを判定し、

前記第 2 インデックス管理部は、各前記第 2 インデックスを各前記第 2 インデックスに対応するウェイに対応するハッシュ関数を用いて求められた前記第 1 インデックスと一致するように変更し、前記キャッシュメモリにおける変更した各前記第 2 インデックスが示すインデックスで、且つ各前記第 2 インデックスに対応するウェイに各前記第 2 アドレスのタグを格納する

ことを特徴とする請求項 1 ~ 4 のいずれか一つに記載のキャッシュメモリ装置。

【請求項 6】

タグを含むアドレスでデータの格納場所が指定される主記憶メモリと、

10

20

30

40

50

複数のウェイを有するキャッシュメモリと、
 前記主記憶メモリのアドレスである第 1 アドレスを取得する第 1 アドレス取得部と、
 前記第 1 アドレスから前記キャッシュメモリのインデックスを示す第 1 インデックスを
 ハッシュ関数により求め、前記キャッシュメモリにおける前記第 1 インデックスが示すイン
 デックスに前記第 1 アドレスのタグを格納する第 1 インデックス管理部と、
 前記第 1 アドレスから前記第 1 アドレスとタグが異なる第 2 アドレスを生成する第 2 ア
 ドレス生成部と、
 前記第 2 アドレスから前記キャッシュメモリのインデックスを示す第 2 インデックスを
 前記ハッシュ関数により求める第 2 インデックス算出部と、
 前記第 1 インデックスと前記第 2 インデックスとが一致するか否かを判定する判定部と

10

、
 前記判定部により一致しないと判定された場合、前記第 2 インデックスを前記第 1 イン
 デックスと一致するように変更し、前記キャッシュメモリにおける変更した前記第 2 イン
 デックスが示すインデックスで、且つ前記第 1 アドレスのタグを格納したウェイとは異なる
 ウェイに前記第 2 アドレスのタグを格納する第 2 インデックス管理部と、
 を備えたことを特徴とする情報処理装置。

【請求項 7】

前記主記憶メモリのアドレスである第 1 アドレスを取得し、
 複数のウェイを有するキャッシュメモリのインデックスを示す第 1 インデックスを、前
 記第 1 アドレスからハッシュ関数により求め、前記キャッシュメモリにおける前記第 1 イン
 デックスが示すインデックスに前記第 1 アドレスのタグを格納し、
 前記第 1 アドレスから前記第 1 アドレスとタグが異なる第 2 アドレスを生成し、
 前記第 2 アドレスから前記キャッシュメモリのインデックスを示す第 2 インデックスを
 前記ハッシュ関数により求め、
 前記第 1 インデックスと前記第 2 インデックスとが一致するか否かを判定し、
 一致しない場合、前記第 2 インデックスを前記第 1 インデックスと一致するように変更
 し、前記キャッシュメモリにおける変更した前記第 2 インデックスが示すインデックスで
 、且つ前記第 1 アドレスのタグを格納したウェイとは異なるウェイに前記第 2 アドレスの
 タグを格納する
 ことを特徴とするキャッシュメモリ制御方法。

20

30

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、キャッシュメモリ装置、情報処理装置及びキャッシュメモリ制御方法に関す
 る。

【背景技術】

【0002】

近年、コンピュータの高速化の一手段として、主記憶メモリ（主記憶装置）とは別に、
 CPU にキャッシュメモリと呼ばれる数キロバイトから数メガバイトにも及ぶ高速メモリ
 を配置し処理のスピードアップが図られている。キャッシュメモリは、CPU がデータや
 命令などの情報を取得/更新する際に主記憶装置やバスなどによる遅延を隠蔽化させ、C
 PU などと記憶装置との性能差を埋めるために用いる高速小容量メモリである。キャッシ
 ュメモリを使用した場合、CPU は、主記憶メモリからキャッシュメモリに読み込んだデ
 ータを、主記憶メモリ上にあるデータに比べて高速で処理することができる。

40

【0003】

例えば、キャッシュメモリを用いたシステムとして、ダイレクトマップ方式の 1 次キャ
 ッシュメモリ及び 1 次キャッシュよりも小規模かつ高速なダイレクトマップ方式の 2 次キ
 ャッシュメモリを有する従来技術が提案されている（例えば、特許文献 1 参照）。

【0004】

キャッシュメモリの容量は、キャッシュを検索するアドレス範囲とキャッシュの連想数

50

の積で表される。ここで、キャッシュを検索するアドレス範囲とは、キャッシュのインデックス範囲である。以下では、キャッシュを検索するアドレス範囲を、単に「アドレス範囲」という。また、連想数は、ウェイ数とも呼ばれる。一般的に、設計上の物理的制約から、連想数はアドレス範囲より小さくなる。そして、キャッシュのアドレスが衝突する複数のデータは、キャッシュメモリのウェイ数の上限まで格納することができる。

【0005】

ソフトウェアは、キャッシュの構造を意識して、キャッシュメモリのアドレスが異なる方向（インデックス方向）にデータを格納したり、ウェイが異なる方向（ウェイ方向）にデータを格納したりすることができる。

【0006】

情報処理装置においては、キャッシュメモリが適切に動作しているか否かを判定するための試験が行われる。このような試験では、キャッシュメモリに負荷をかけて正常な動作が行われるか否かが判定される。例えば、キャッシュメモリへの負荷は、キャッシュメモリのアドレスが衝突するメモリアクセスを多発させ、外部メモリからキャッシュメモリへのアクセスを発生させたり、キャッシュメモリのデータの置き換え処理を発生させたりすることで増加する。キャッシュメモリのデータの置き換え処理は、「リプレース」とも呼ばれる。

【0007】

さらに、近年、キャッシュメモリにおけるウェイ数がアドレス範囲の数より小さいという制約を克服するため、キャッシュメモリのアドレスの生成回路にハッシュ関数を導入することで、見かけ上の連想数を増加させる手法が採用されている。アドレス生成回路にハッシュ関数を導入した装置は、導入しない装置に比べてキャッシュのヒット率が高く、外部のメモリ参照の頻度が低くなる傾向がある。

【0008】

図6は、従来のハッシュ関数を用いない場合のインデックス計算を説明する図である。また、図7は、従来のハッシュ関数を用いた場合のインデックス計算を説明する図である。

【0009】

図6に示すように、CPUから指定されたメモリアドレス901は、タグ及びインデックスを有する。そして、キャッシュは、キャッシュタグアレイ902とキャッシュデータアレイ903とを有する。p-1~p-5は、「a」を先頭とする一連のインデックスに対応するように並んでいるタグである。また、q-1~q-5も、「a」を先頭とする一連のインデックスに対応するように並んでいるタグである。ハッシュ関数を用いない場合、タグがp-1~p-5のデータのそれぞれと、タグがq-1~q-5のデータのそれぞれとは、同じインデックス番号の異なるウェイのエントリに格納される。すなわち、キャッシュタグアレイ902では、図6に示すようにウェイ方向にp-1~p-5とq-1~q-5とが並ぶ。このため、図6の場合、ハッシュ関数を用いない場合、同じインデックスを有するアドレスのデータが5つ指定されればキャッシュミスヒットを起こす。

【0010】

一方、ハッシュ関数を使用した場合、図7のキャッシュタグアレイ904に示すように、p-1~p-5とq-1~q-5とがそれぞれ一致していても、異なるインデックスのエントリに格納される。そのため、ハッシュ前に同じインデックスを有するアドレスのデータが5つ指定されても、キャッシュミスヒットを起こさない確率が図6の場合と比べて高くなる。

【0011】

ただし、キャッシュメモリに使用されるハッシュ関数は、ハッシュ関数から逆関数を直接求めることができない。そのため、アドレスの生成にハッシュ関数を導入したキャッシュメモリの試験を行う場合、試験プログラムは、予め用意したハッシュ衝突が発生する多数のメモリアドレスを用いてキャッシュメモリに対して負荷をかけることが考えられる。

【0012】

10

20

30

40

50

ハッシュ関数が公開されており且つアルゴリズムが簡単な場合は、ソフトウェアによりハッシュ関数を使用してハッシュ衝突が発生するアドレスのリストを予めピックアップすることができる。そのため、試験プログラムは、予めピックアップされたアドレスを用いてメモリアクセス命令を実行することで、キャッシュメモリに負荷をかけることができる。

【先行技術文献】

【特許文献】

【0013】

【特許文献1】特開平5 - 324473号公報

【発明の概要】

10

【発明が解決しようとする課題】

【0014】

しかしながら、ハッシュ関数を公開していない場合や、ハッシュ関数のアルゴリズムが複雑な場合はソフトウェアからハッシュ衝突が発生するアドレスを生成することが困難となり、このような方法によるキャッシュメモリのテストを行うことは困難である。

【0015】

例えば、ダイレクトマップ方式の1次キャッシュメモリ及び小規模なダイレクトマップ方式の2次キャッシュメモリを用いた従来技術では、ハッシュを用いた場合のキャッシュメモリの試験については考慮しておらず、上記の問題を解決することは困難である。

【0016】

20

開示の技術は、上記に鑑みてなされたものであって、キャッシュメモリの負荷試験を行う際に、キャッシュメモリのハッシュ衝突を容易に発生させるキャッシュメモリ装置、情報処理装置及びキャッシュメモリ制御方法を提供することを目的とする。

【課題を解決するための手段】

【0017】

本願の開示するキャッシュメモリ装置、情報処理装置及びキャッシュメモリ制御方法は、一つの態様において、キャッシュメモリは、複数のウェイを有する。第1アドレス取得部は、前記主記憶メモリのアドレスである第1アドレスを取得する。第1インデックス管理部は、前記第1アドレスから前記キャッシュメモリのインデックスを示す第1インデックスをハッシュ関数により求め、前記キャッシュメモリにおける前記第1インデックスが示すインデックスに前記第1アドレスのタグを格納する。第2アドレス生成部は、前記第1アドレスから前記第1アドレスとタグが異なる第2アドレスを生成する。第2インデックス算出部は、前記第2アドレスから前記キャッシュメモリのインデックスを示す第2インデックスを前記ハッシュ関数により求める。判定部は、前記第1インデックスと前記第2インデックスとが一致するか否かを判定する。第2インデックス管理部は、前記判定部により一致しないと判定された場合、前記第2インデックスを前記第1インデックスと一致するように変更し、前記キャッシュメモリにおける変更した前記第2インデックスが示すインデックスで、且つ前記第1アドレスのタグを格納したウェイとは異なるウェイに前記第2アドレスのタグを格納する。

30

【発明の効果】

40

【0018】

本願の開示するキャッシュメモリ装置、情報処理装置及びキャッシュメモリ制御方法の一つの態様によれば、キャッシュメモリの負荷試験を行う際に、キャッシュメモリのハッシュ衝突を容易に発生させることができるという効果を奏する。

【図面の簡単な説明】

【0019】

【図1】図1は、実施例1に係るキャッシュメモリ装置のブロック図である。

【図2】図2は、メモリアドレスの詳細を表す図である。

【図3】図3は、実施例1に係るキャッシュメモリ装置におけるデータ書込みの試験のフローチャートである。

50

【図4】図4は、実施例2に係るキャッシュメモリ装置のブロック図である。

【図5】図5は、キャッシュメモリ装置のハードウェア構成図である。

【図6】図6は、従来のハッシュ関数を用いない場合のインデックス計算を説明する図である。

【図7】図7は、従来のハッシュ関数を用いた場合のインデックス計算を説明する図である。

【発明を実施するための形態】

【0020】

以下に、本願の開示するキャッシュメモリ装置、情報処理装置及びキャッシュメモリ制御方法の実施例を図面に基づいて詳細に説明する。なお、以下の実施例により本願の開示するキャッシュメモリ装置、情報処理装置及びキャッシュメモリ制御方法が限定されるものではない。

10

【実施例1】

【0021】

図1は、実施例1に係るキャッシュメモリ装置のブロック図である。本実施例に係るキャッシュメモリ装置は、図1に示すように、アドレス取得部1、ハッシュ回路2、プリフェッチ部3、ハッシュ回路4、衝突検出部5、ハッシュ値変更部6を有する。さらに、本実施例に係るキャッシュメモリ装置は、キャッシュメモリ7、比較回路81~84、ヒット判定部9、選択回路10を有する。

【0022】

20

アドレス取得部1は、メモリテストを実施するプログラムが生成したデータを読み出す主記憶メモリ上のアドレスを取得する。以下では、アドレス取得部1がメモリテスト用のプログラムから取得したアドレスを「デマンドアドレス」という。そして、アドレス取得部1は、デマンドアドレスを、タグ11及びインデックス12に分ける。タグ11とは、そのアドレスで指定された主記憶メモリ上のデータがキャッシュに格納されているか否かを判断するための情報である。

【0023】

タグ11及びインデックス12のビット数は予め決められており、アドレス取得部1は、予め決められたビット数に合わせてデマンドアドレスをタグ11及びインデックス12に分ける。

30

【0024】

アドレス取得部1は、タグ11の値をプリフェッチ部3へ出力する。また、アドレス取得部1は、タグ11の値及びインデックス12の値をハッシュ回路2へ出力する。

【0025】

ハッシュ回路2は、タグ11の値及びインデックス12の値の入力をアドレス取得部1から受ける。そして、ハッシュ回路2は、受信したタグ11の値及びインデックス12の値に対してハッシュ関数を用いてハッシュ値を算出する。ここで、本実施例における、ハッシュ回路2によるハッシュ演算の一例について説明する。図2は、メモリアドレスの詳細を表す図である。図2に示すメモリアドレスは、タグ201及びインデックス202を有している。各領域に対応するようにメモリアドレスの下に記載した数字は、各領域が使用するメモリアドレスを表す。すなわち、タグ201は、メモリアドレスの19~63ビットを使用する。また、インデックス202は、メモリアドレスの7~18ビットを使用する。この場合の、ラインサイズは、0~6ビットとなる。ハッシュ回路2は、タグ201のインデックス202側から3ビットずつ5つの値211~215を取得する。さらに、ハッシュ回路2は、インデックス12から値221を取得する。そして、ハッシュ回路2は、値211~215及び値221の各ビットの排他的論理和(XOR)を求めることでハッシュ値を算出する。

40

【0026】

ハッシュ回路2は、キャッシュメモリ7のキャッシュタグアレイ71における、求めたハッシュ値を有するインデックスにタグが格納されているか否か、すなわちキャッシュヒ

50

ットが発生するか否かの判定を、比較回路 8 1 ~ 8 4 の比較結果に指示する。例えば、ハッシュ回路 2 は、3 ビットのハッシュ値にインデックス 1 2 の先頭の 3 ビットを除いた値を付加するなどして 1 2 ビットのインデックスを作成する。そして、ハッシュ回路 2 は、求めた 1 2 ビットのインデックスをキャッシュタグアレイ 7 1 を検索するためのインデックス番号として用いる。キャッシュヒットが発生しなかった場合、ハッシュ回路 2 は、比較回路 8 1 ~ 8 4 にタグが格納されていないことを通知する。また、ハッシュ回路 2 は、求めたハッシュ値をデータ書込部 1 0 1 へ出力する。

【 0 0 2 7 】

また、ハッシュ回路 2 は、求めたハッシュ値を衝突検出部 5 へ出力する。

【 0 0 2 8 】

プリフェッチ部 3 は、デマンドアドレス、すなわちタグ 1 1 及びインデックス 1 2 の入力をアドレス取得部 1 から受ける。そして、プリフェッチ部 3 は、受信したタグ 1 1 を 1 つずつインクリメントしていき、インクリメントした結果にインデックス 1 2 を付加して、プリフェッチアドレスを生成する。プリフェッチアドレスは、図 1 に示すように、タグ 3 1 及びインデックス 3 2 を有する。ここで、タグ 3 1 は、タグ 1 1 を 1 つずつインクリメントしていくことで生成される値である。また、インデックス 3 2 は、インデックス 1 2 と一致する。

【 0 0 2 9 】

プリフェッチ部 3 は、タグ 3 1 及びインデックス 3 2 をハッシュ回路 4 へ出力する。

【 0 0 3 0 】

また、プリフェッチ部 3 は、ハッシュ値変更部 6 により変更されたハッシュ値を有するインデックスにタグが格納されている場合、比較回路 8 1 ~ 8 4 にタグの情報を出力する。

【 0 0 3 1 】

ハッシュ回路 4 は、タグ 3 1 及びインデックス 3 2 をプリフェッチ部 3 から受信する。そして、ハッシュ回路 4 は、タグ 3 1 及びインデックス 3 2 を用いてハッシュ値を求める。ここで、本実施例では、例えば、ハッシュ回路 4 は、ハッシュ回路 2 と同様にタグ 1 1 から取得した 5 つの 3 ビットの値及びインデックス 1 2 から取得した 3 ビットの値の各ビットの排他的論理和を求めることでハッシュ値を算出する。

【 0 0 3 2 】

ハッシュ回路 4 は、算出したハッシュ値を衝突検出部 5 及びハッシュ値変更部 6 へ出力する。

【 0 0 3 3 】

衝突検出部 5 は、デマンドアドレスのハッシュ値の入力をハッシュ回路 2 から受ける。さらに、衝突検出部 5 は、プリフェッチアドレスのハッシュ値の入力をハッシュ回路 2 から受ける。そして、衝突検出部 5 は、デマンドアドレスのハッシュ値とプリフェッチアドレスのハッシュ値との各ビットの排他的論理和を求める。図 1 中の四角で囲われた数字の列が、アドレスの遷移の一例を示している。例えば、デマンドアドレスのハッシュ値が「1 1 1」であり、プリフェッチアドレスのハッシュ値が「1 0 1」であれば、衝突検出部 5 は、各ビットの排他的論理和をもとめ、「0 1 0」を取得する。ここで、デマンドアドレスのハッシュ値とプリフェッチアドレスのハッシュ値が一致している場合、すなわち、ハッシュが衝突している場合は、衝突検出部 5 は、排他的論理和として「0 0 0」を取得する。

【 0 0 3 4 】

そして、衝突検出部 5 は、求めた排他的論理和をハッシュ値変更部 6 へ出力する。

【 0 0 3 5 】

ハッシュ値変更部 6 は、プリフェッチアドレスのハッシュ値の入力をハッシュ回路 4 から受ける。また、ハッシュ値変更部 6 は、衝突検出部 5 が求めたデマンドアドレスのハッシュ値とプリフェッチアドレスのハッシュ値との各ビットの排他的論理和の結果の入力を衝突検出部 5 から受ける。

10

20

30

40

50

【0036】

そして、ハッシュ値変更部6は、プリフェッチアドレスのハッシュ値とデマンドアドレスのハッシュ値とプリフェッチアドレスのハッシュ値との各ビットの排他的論理和の結果との各ビットの排他的論理和を求める。例えば、デマンドアドレスのハッシュ値が「111」であり、プリフェッチアドレスのハッシュ値が「101」であり、衝突検出部5により求められた排他的論理和が「010」の場合で説明する。この場合、ハッシュ値変更部6は、「101」と「010」との排他的論理和を求め、「111」を取得する。ここで、ハッシュ値変更部6の算出結果である「111」は、デマンドアドレスのハッシュ値「111」と一致する。すなわち、ハッシュ値変更部6は、デマンドアドレスのハッシュ値と衝突するハッシュ値を取得する。

10

【0037】

また、デマンドアドレスのハッシュ値とプリフェッチアドレスのハッシュ値が衝突している場合は、ハッシュ値変更部6は、衝突検出部5から「000」を受信する。この場合、ハッシュ値変更部6は、プリフェッチアドレスのハッシュ値と「000」との排他的論理和を求めることになり、プリフェッチアドレスのハッシュ値をそのまま取得することになる。すなわち、デマンドアドレスのハッシュ値とプリフェッチアドレスのハッシュ値が衝突している場合は、ハッシュ値変更部6は、プリフェッチアドレスのハッシュ値を変更しない。

【0038】

このようにして、ハッシュ値変更部6は、プリフェッチアドレスのハッシュ値を、デマンドアドレスのハッシュ値と衝突する値に変更する。さらに、ハッシュ値変更部6は、他のプリフェッチアドレスの生成をプリフェッチ部3に指示する。

20

【0039】

選択回路103は、デマンドアドレスのハッシュ値の入力をハッシュ回路2から受ける。また、選択回路103は、プリフェッチアドレスのハッシュ値の入力をハッシュ値変更部6から受ける。また、選択回路103は、格納用のデータの入力をデータ書込部101から受ける。そして、選択回路103は、ハッシュ回路2から受信したデマンドアドレスのハッシュ値及びハッシュ値変更部6から受信したプリフェッチアドレスのハッシュ値を有するインデックスをデータ書込部101へ送信する。また、選択回路103は、データ書込部101から受信したデータのインデックスをキャッシュメモリ7へ出力する。

30

【0040】

データ書込部101は、ハッシュ回路2が求めたハッシュ値を用いた検索においてヒット判定部9によってキャッシュミスヒットと判定された場合、ハッシュ回路4からハッシュ値を含むインデックスの情報の入力を受ける。また、データ書込部101は、デマンドアドレスの入力をアドレス取得部1から受ける。次に、データ書込部101は、主記憶メモリ200のデマンドアドレスからデータを取得する。そして、データ書込部101は、キャッシュタグアレイ71における、ハッシュ回路4から受信したインデックスの空いているウェイのエントリにタグをセットする。さらに、データ書込部101は、タグをセットしたインデックス及びウェイに対応するキャッシュデータアレイ72のエントリに取得したデータを書込む。

40

【0041】

ここで、データ書込部101は、キャッシュタグアレイ71にタグをセットする場合、求めたハッシュ値を有するインデックスに対応する各ウェイのエントリに空きがあるか否かを判定する。全てのウェイにおいて求めたハッシュ値を有するインデックスに対応するエントリに空きがない場合、すなわち、エントリが枯渇している場合、データ書込部2は、そのインデックスに対応するエントリにセットされているタグから一つをリプレース対象として選ぶ。例えば、データ書込部101は、Least Recently Used (LRU)方式によってリプレース対象を選択する。そして、データ書込部101は、選択したタグがセットされているエントリからそのタグを削除し、現在セットしようとしているタグ11をセットする。

50

【0042】

また、ハッシュ値変更部6が求めたハッシュ値を用いた検索においてヒット判定部9によってキャッシュミスヒットと判定された場合、データ書込部101は、ハッシュ値変更部6からハッシュ値を含むインデックスの情報の入力を受ける。また、データ書込部101は、プリフェッチアドレスの入力をアドレス取得部1から受ける。次に、データ書込部101は、主記憶メモリ200のプリフェッチアドレスからデータを取得する。そして、データ書込部101は、キャッシュタグアレイ71における、ハッシュ値変更部6から受信したインデックスの空いているウェイのエントリにタグをセットする。さらに、データ書込部101は、タグをセットしたインデックス及びウェイに対応するキャッシュデータアレイ72のエントリに取得したデータを書込む。

10

【0043】

データ書込部101は、この場合も、キャッシュタグアレイ71にタグをセットする場合、求めたハッシュ値を有するインデックスに対応する各ウェイのエントリに空きがあるか否かを判定する。そして、全てのウェイにおいて求めたハッシュ値を有するインデックスに対応するエントリに空きがない場合、すなわち、エントリが枯渇している場合、データ書込部101は、そのインデックスに対応するエントリにセットされているタグから一つをリプレース対象として選ぶ。そして、データ書込部101は、選択したタグがセットされているエントリからそのタグを削除し、現在セットしようとしているタグ31をセットする。

20

【0044】

ここで、デマンドアドレスのハッシュ値とプリフェッチアドレスのハッシュ値が一致するため、デマンドアドレスのタグとプリフェッチアドレスのタグとが同じインデックス番号のエントリにセットされることになる。例えば、デマンドアドレスが「p-1」でプリフェッチアドレスが「p-2」～「p-5」とする。この場合、「p-1」～「p-5」のハッシュ値が一致する。そこで、図1に示すように、「p-1」～「p-4」は、同じインデックス番号のエントリにタグがセットされる。図1では、「p-1」～「p-4」のタグがセットされているエントリを、「p-1」～「p-4」で表している。この場合、p-5も同じインデックス番号を有するエントリにタグがセットされることになるが、既に、そのインデックス番号の全てのウェイが埋まっている。これにより、置き換え、すなわちリプレースが発生する。このように、ハッシュ関数を用いてインデックス番号を生成している場合でも、本実施例に係るキャッシュメモリ装置であれば、容易に衝突を発生させてリプレースを起こすことができ、キャッシュメモリの試験を迅速に行うことができる。

30

【0045】

比較回路81～84は、キャッシュタグアレイ71の各ウェイに対応するように、配置されている。図1では、比較回路81はウェイ#0に対応し、比較回路82はウェイ#1に対応し、比較回路83はウェイ#2に対応し、比較回路84はウェイ#3に対応している。

【0046】

比較回路81～84は、デマンドアドレスのタグ11及びプリフェッチ部3が算出した各プリフェッチアドレスのタグ31の入力を受ける。さらに、比較回路81～84は、ハッシュ回路2又はハッシュ値変更部6から比較の指示を受ける。そして、比較回路81～84は、入力されたタグ11又はタグ31に一致するタグが対応するウェイに有るか否かを判定する。一致するタグがある場合、比較回路81～84は、一致したタグがセットされているインデックス及びウェイの情報をヒット判定部9及び選択回路10に出力する。

40

【0047】

また、比較回路81～84は、ハッシュ値変更部6が生成したアドレスにより、変更したハッシュ値を有するインデックスにタグが格納されているか否かを判定する。

【0048】

ヒット判定部9は、タグ11又はタグ31に一致するタグがある場合、その一致するタ

50

グがセットされているインデックス及びウェイの情報を比較回路 8 1 ~ 8 4 から受ける。インデックス及びウェイの情報の入力を比較回路 8 1 ~ 8 4 のいずれかから受けると、ヒット判定部 9 は、キャッシュメモリに格納されているデータがヒットしたと判定する。これに対して、比較回路 8 1 ~ 8 4 のいずれからもインデックス及びウェイの情報の入力がない場合、ヒット判定部 9 は、キャッシュメモリに格納されているデータがヒットしなかったと判定する。

【 0 0 4 9 】

選択回路 1 0 は、タグ 1 1 又はタグ 3 1 に一致するタグがある場合、その一致するタグがセットされているインデックス及びウェイの情報の入力を比較回路 8 1 ~ 8 4 から受ける。そして、選択回路 1 0 は、受信したインデックス及びウェイに一致するキャッシュメモリのエントリからデータを取得する。そして、選択回路 1 0 は、取得したデータを出力する。

10

【 0 0 5 0 】

次に、図 3 を参照して、本実施例に係るキャッシュメモリ装置によるキャッシュテストの流れについて説明する。図 3 は、実施例 1 に係るキャッシュメモリ装置におけるキャッシュテストのフローチャートである。

【 0 0 5 1 】

アドレス取得部 1 は、デマンドアドレスを取得する (ステップ S 1 0 1)。アドレス取得部 1 は、デマンドアドレスのタグ及びインデックスをハッシュ回路 2 へ出力する。また、アドレス取得部 1 は、デマンドアドレスをプリフェッチ部 3 へ出力する。

20

【 0 0 5 2 】

ハッシュ回路 2 は、デマンドアドレスの入力をアドレス取得部 1 から受ける。次に、ハッシュ回路 2 は、デマンドアドレスのタグ及びインデックスからハッシュ値を算出する (ステップ S 1 0 2)。そして、ハッシュ回路 2、比較回路 8 1 ~ 8 4 及びヒット判定部 9 は、ハッシュ回路 2 が求めたハッシュ値を用いてキャッシュメモリ 7 を検索し、キャッシュヒットしたか否かを判定する (ステップ S 1 0 3)。具体的には、比較回路 8 1 ~ 8 4 は、ハッシュ回路 2 により生成されたアドレスがキャッシュタグアレイ 7 1 におけるハッシュ値を有するインデックスにタグが格納されているか否かを判定する。タグが格納されていない場合、ヒット判定部 9 は、キャッシュミスヒットと判定する。タグが格納されている場合、比較回路 8 1 ~ 8 4 は、ハッシュ回路 2 が求めたハッシュ値を有するインデックスの各ウェイの値とタグの値とを比較する。ヒット判定部 9 は、比較結果を基にキャッシュヒットが発生したか否かを判定する。キャッシュヒットした場合 (ステップ S 1 0 3 : 肯定)、キャッシュメモリ 7 に格納されているデータが、キャッシュデータアレイ 7 2 から読み出され出力される (ステップ S 1 0 4)。

30

【 0 0 5 3 】

ヒット判定部 9 がキャッシュミスヒットと判定した場合 (ステップ S 1 0 3 : 否定)、データ書込部 1 0 1 は、ハッシュ値を有するインデックスの情報をハッシュ回路 2 から受信する。さらに、データ書込部 1 0 1 は、アドレス取得部 1 からデマンドアドレスを取得する (ステップ S 1 0 5)。そして、データ書込部 1 0 1 は、主記憶メモリ 2 0 0 における受信したデマンドアドレスに格納されているデータを読み出す (ステップ S 1 0 6)。次に、データ書込部 1 0 1 は、キャッシュデータを格納する (ステップ S 1 0 7)。具体的には、データ書込部 1 0 1 は、ハッシュ回路 2 が算出したハッシュ値を有するインデックスのエントリにタグをセットする。さらに、データ書込部 1 0 1 は、タグをセットしたキャッシュタグアレイ 7 1 の位置に対応するキャッシュデータアレイ 7 2 のエントリにデータを格納する。

40

【 0 0 5 4 】

さらに、ハッシュ回路 2 は、算出したハッシュ値を衝突検出部 5 へ出力する。プリフェッチ部 3 は、デマンドアドレスの入力をアドレス取得部 1 から受ける。そして、プリフェッチ部 3 は、デマンドアドレスのタグをインクリメントし、プリフェッチアドレスを生成する (ステップ S 1 0 8)。プリフェッチ部 3 は、生成したプリフェッチアドレスをハッ

50

ッシュ回路 4 へ出力する。

【 0 0 5 5 】

ハッシュ回路 4 は、プリフェッチアドレスの入力をプリフェッチ部 3 から受ける。次に、ハッシュ回路 4 は、受信したプリフェッチアドレスのタグ及びインデックスからハッシュ値を算出する（ステップ S 1 0 9）。そして、ハッシュ回路 4 は、算出したハッシュ値を衝突検出部 5 及びハッシュ値変更部 6 へ出力する。

【 0 0 5 6 】

衝突検出部 5 は、アドレス取得部 1 により生成されたデマンドアドレスのハッシュ値の入力をハッシュ回路 2 から受ける。また、衝突検出部 5 は、プリフェッチ部 3 により算出されたプリフェッチアドレスのハッシュ値の入力をハッシュ回路 4 から受ける。そして、衝突検出部 5 は、デマンドアドレスのハッシュ値とプリフェッチアドレスのハッシュ値との各ビットの排他的論理和を求める（ステップ S 1 1 0）。ここで、衝突検出部 5 は、排他的論理和の算出結果が「0 0 0」であれば、ハッシュが衝突しており、それ以外であれば、ハッシュは衝突していないと判定する。そして、衝突検出部 5 は、排他的論理和の算出結果をハッシュ値変更部 6 へ出力する。

【 0 0 5 7 】

ハッシュ値変更部 6 は、デマンドアドレスのハッシュ値とプリフェッチアドレスのハッシュ値との各ビットの排他的論理和の算出結果の入力を衝突検出部 5 から受ける。また、ハッシュ値変更部 6 は、プリフェッチアドレスのハッシュ値の入力をハッシュ回路 4 から受ける。ハッシュ値変更部 6 は、デマンドアドレスのハッシュ値とプリフェッチアドレスのハッシュ値との各ビットの排他的論理和の算出結果とプリフェッチアドレスのハッシュ値との各ビットの排他的論理和を算出し、ハッシュ値の変更を行う（ステップ S 1 1 1）。そして、ハッシュ値変更部 6、比較回路 8 1 ~ 8 4 及びヒット判定部 9 は、ハッシュ値変更部 6 が求めたハッシュ値、すなわち変換後のハッシュ値を用いてキャッシュメモリ 7 を検索し、キャッシュヒットが発生したか否かを判定する（ステップ S 1 1 2）。具体的には、ハッシュ値変更部 6 は、キャッシュタグアレイ 7 1 における求めたハッシュ値を有するインデックスにタグが格納されているか否かを判定する。タグが格納されていなかった場合、ヒット判定部 9 はキャッシュミスヒットと判定する。また、タグが格納されている場合、比較回路 8 1 ~ 8 4 は、ハッシュ回路 2 が求めたハッシュ値を有するインデックスの各ウェイの値とタグの値とを比較する。ヒット判定部 9 は、比較結果を基にキャッシュヒットが発生したか否かを判定する。

【 0 0 5 8 】

ヒット判定部 9 がキャッシュミスヒットと判定した場合（ステップ S 1 1 2：否定）、データ書込部 1 0 1 は、プリフェッチアドレスに対するデータの格納により、リプレースが発生するか否かを判定する（ステップ S 1 1 3）。具体的にはハッシュ値変更部 6 は、算出した排他的論理和の値を有するインデックスにおいて空いているウェイが無ければ、リプレース発生と判定する。リプレースが発生した場合（ステップ S 1 1 3：肯定）、ハッシュ値変更部 6 は、キャッシュデータのリプレース処理を行う（ステップ S 1 1 4）。

【 0 0 5 9 】

これに対して、リプレースが発生しない場合（ステップ S 1 1 3：否定）、データ書込部 1 0 1 は、キャッシュデータを格納する（ステップ S 1 1 5）。

【 0 0 6 0 】

そして、データ書込部 1 0 1 は、ハッシュ値を有するインデックスの情報をハッシュ値変更部 6 から取得する。また、データ書込部 1 0 1 は、プリフェッチ部 3 からプリフェッチアドレスを取得する（ステップ S 1 1 6）。そして、データ書込部 1 0 1 は、主記憶メモリ 2 0 0 における受信したプリフェッチアドレスに格納されているデータを読み出す（ステップ S 1 1 7）。

【 0 0 6 1 】

一方、キャッシュヒットが発生している場合（ステップ S 1 1 2：肯定）、処理はステップ S 1 0 8 へ戻る。

10

20

30

40

50

【0062】

プリフェッチ部3は、規定数のプリフェッチアドレスの算出が完了したか否かを判定する(ステップS118)。完了していない場合(ステップS118:否定)、プリフェッチ部3は、ステップS108へ戻る。

【0063】

これに対して、規定数のプリフェッチアドレスの算出が完了した場合(ステップS118:肯定)、プリフェッチ部3は、プリフェッチアドレスの算出を終了する。

【0064】

ここで、図3のフローチャートでは、1組のデマンドアドレスに対するプリフェッチアドレスの算出及びそれらを用いたテストの流れを説明したが、実際には、複数のデマンドアドレスを生成し、それら全てに対して図3のフローを繰り返すことでテストを行う。

10

【0065】

また、図3では、デマンドアドレスへのデータの格納の後に、プリフェッチアドレスの生成及びプリフェッチアドレスへのデータの格納を行うように説明したが、これらは並行して行われてもよい。また、主記憶メモリ200へのアクセスとプリフェッチアドレスの生成及びプリフェッチアドレスへのデータの格納も並行して行うことが可能である。

【0066】

ここでは、ハッシュ衝突を起こさせるテストの場合について主に説明したが、この情報処理装置は、テスト以外の通常動作も行う。すなわち、テスト時以外には、情報処理装置は以下の動作を行う。情報処理装置は、受け付けたデータ要求に対してキャッシュに格納されているか否かを判定し、キャッシュに格納されている場合には、キャッシュからデータを読み出す。また、キャッシュに格納されていない場合には、情報処理装置は、メモリからデータを読み出し、読み出したデータをキャッシュに格納しておく。この場合、情報処理装置では、プリフェッチ部3などをディセーブルにしておくことが好ましい。また、受け付けたデマンドアドレスは、ハッシュ回路2においてハッシュが求められ、比較回路81~84に送られキャッシュヒットの有無が判定される。

20

【0067】

以上に説明したように、本実施例に係るキャッシュメモリ装置は、生成した一つのデマンドアドレスに対して複数のプリフェッチアドレスを生成する。次に、キャッシュメモリ装置は、ハードウェアを用いて生成したアドレス同士の排他的論理和を求めることで衝突を検出し、衝突していない場合にはプリフェッチアドレスのハッシュ値をデマンドアドレスのハッシュ値に一致させる。その後、キャッシュメモリ装置は、ハッシュ値を一致させたデマンドアドレス及びプリフェッチアドレスを順次そのハッシュ値に一致するインデックスを有するエントリにタグをセットしていく。これにより、キャッシュメモリの負荷試験を行う際に、容易にハッシュ値の衝突を発生することができ、迅速にキャッシュメモリのテストを行うことができる。

30

【実施例2】

【0068】

図4は、実施例2に係るキャッシュメモリ装置のブロック図である。本実施例に係るキャッシュメモリ装置は、ウェイ毎に異なるハッシュ関数を使用することが実施例1と異なるものである。さらに、本実施例に係るキャッシュメモリ装置では、あるインデックスにおけるエントリが枯渇した状態で、ハッシュ衝突が発生した場合、主記憶メモリに対するリプレースを行わずに、キャッシュのウェイ間で同じインデックスを有するエントリにデータの置き換え処理を行う。このキャッシュのウェイ間でのデータの置き換え処理は「Victim処理」と呼ばれる場合がある。以下では、実施例1と同じ機能を有する各部については説明を省略する。

40

【0069】

本実施例では、図4に示すように、キャッシュメモリのウェイ毎にハッシュ回路21~24が配置されている。ここで、本実施例では、4ウェイのキャッシュメモリを例に説明しているため、ハッシュ回路21~24という4つが設けられている。また、本実施例に

50

係るキャッシュメモリ装置には、ハッシュ回路 2 1 ~ 2 4 のそれぞれに対応するように、ハッシュ回路 4 1 ~ 4 4、衝突検出部 5 1 ~ 5 4 及びハッシュ値変更部 6 1 ~ 6 4 が配置されている。さらに、本実施例に係るキャッシュメモリ装置には、選択回路 3 1 0 及びハッシュ回路 3 2 1 ~ 3 2 4 が配置されている。

【 0 0 7 0 】

アドレス取得部 1 は、生成したデマンドアドレスのタグ 1 1 及びインデックス 1 2 を、ハッシュ回路 2 1 ~ 2 4 へ出力する。本実施例では、デマンドアドレスはウェイ # 0 に格納される場合で説明する。また、アドレス取得部 1 は、生成したデマンドアドレスをプリフェッチ部 3 へ出力する。

【 0 0 7 1 】

ハッシュ回路 2 1 ~ 2 4 は、予め記憶しているハッシュ関数を用いて、デマンドアドレスのハッシュ値を算出する。ハッシュ回路 2 1 ~ 2 4 は、算出したハッシュ値をそれぞれ対応する衝突検出部 5 1 ~ 5 4 へ出力する。そして、キャッシュミスヒットの場合、データ書込部 1 0 1 は、ウェイ # 0 のハッシュ値を有するインデックスのエントリにデマンドアドレスのタグ 1 1 をセットし、デマンドアドレスのデータをウェイ # 0 に格納する。本実施例では、デマンドアドレスのデータをウェイ # 0 に格納する場合で説明しているが、デマンドアドレスのデータはどのウェイに格納されても良い。データ書込部 1 0 1 は、デマンドアドレスのデータが格納されるウェイに対応するハッシュ回路 2 1 ~ 2 4 が求めたハッシュ値を有するインデックスのエントリにタグをセットする。また、ハッシュ回路 2 1 ~ 2 4 は、求めたハッシュ値を選択回路 1 3 1 ~ 1 3 4 へ出力する。

【 0 0 7 2 】

プリフェッチ部 3 は、デマンドアドレスの入力をアドレス取得部 1 から受ける。次に、プリフェッチ部 3 は、受信したデマンドアドレスのタグ 1 1 を 1 つずつインクリメントしてプリフェッチアドレスを算出する。そして、プリフェッチ部 3 は、最初のプリフェッチアドレスをウェイ # 1 に対応するハッシュ回路 4 2 に出力する。また、プリフェッチ部 3 は、2 番目のプリフェッチアドレスをウェイ # 2 に対応するハッシュ回路 4 3 に出力する。また、プリフェッチ部 3 は、3 番目のプリフェッチアドレスをウェイ # 3 に対応するハッシュ回路 4 4 に出力する。また、プリフェッチ部 3 は、4 番目のプリフェッチアドレスをウェイ # 0 に対応するハッシュ回路 4 1 に出力する。このようにして、プリフェッチ部 3 は、データを格納するウェイを順次変更するようにプリフェッチアドレスの出力先をハッシュ回路 4 1 ~ 4 4 の中から選択し、プリフェッチアドレスを出力していく。

【 0 0 7 3 】

ハッシュ回路 4 1 ~ 4 4 は、それぞれハッシュ回路 2 1 ~ 2 4 と同じハッシュ関数を記憶している。ここで、ハッシュ回路 4 1 ~ 4 4 はそれぞれ同様の動作を行うので、以下では、ハッシュ回路 4 1 を例に説明する。ハッシュ回路 4 1 は、プリフェッチアドレスの入力をプリフェッチ部 3 から受ける。そして、ハッシュ回路 4 1 は、受信したプリフェッチアドレスのタグ 3 1 及びインデックス 3 2 に対してハッシュ関数を用いてハッシュ値を算出する。その後、ハッシュ回路 4 1 は、算出したハッシュ値を衝突検出部 5 1 及びハッシュ値変更部 6 1 へ出力する。

【 0 0 7 4 】

衝突検出部 5 1 ~ 5 4 はそれぞれ同様の動作を行うので、以下では、衝突検出部 5 1 を例に説明する。衝突検出部 5 1 は、プリフェッチアドレスのハッシュ値の入力をハッシュ回路 4 1 から受ける。また、衝突検出部 5 1 は、デマンドアドレスのハッシュ値の入力をハッシュ回路 2 1 から受ける。

【 0 0 7 5 】

衝突検出部 5 1 は、プリフェッチアドレスのハッシュ値とデマンドアドレスのハッシュ値との各ビットの排他的論理和を求め、求めた結果をハッシュ値変更部 6 1 へ出力する。ここで、例えばハッシュ値が 3 ビットで表される場合、衝突検出部 5 1 は、排他的論理和が「 0 0 0 」であればハッシュが衝突しており、排他的論理和がそれ以外であればハッシュの衝突が起こっていないと判定できる。

10

20

30

40

50

【0076】

ハッシュ値変更部61～64はそれぞれ同様の動作を行うので、以下では、ハッシュ値変更部61を例に説明する。ハッシュ値変更部61は、ハッシュを行う前のプリフェッチアドレスの入力をプリフェッチ部3から受ける。また、ハッシュ値変更部61は、プリフェッチアドレスのハッシュ値とデマンドアドレスのハッシュ値との各ビットの排他的論理和の算出結果の入力を衝突検出部51から受ける。そして、ハッシュ値変更部61は、プリフェッチアドレスと衝突検出部51による算出結果との各ビットの排他的論理和を求める。これにより、ハッシュ値変更部61は、デマンドアドレスとハッシュ衝突を起こすプリフェッチアドレスが求められる。そして、ハッシュ値変更部61は、ハッシュ衝突を起こすプリフェッチアドレスである排他的論理和の結果を選択回路310へ入力する。同様に、ハッシュ値変更部62～64も、求めたハッシュ衝突を起こすプリフェッチアドレスを選択回路310へ出力する。

10

【0077】

選択回路310は、ハッシュ衝突を起こすプリフェッチアドレスの入力をハッシュ値変更部61～64から受ける。そして、選択回路310は、入力されたプリフェッチアドレスの中から1つずつ選択し、選択したプリフェッチアドレスを順次ハッシュ回路321～324へ出力する。

【0078】

ハッシュ回路321～324は、それぞれ異なったハッシュ関数を記憶している。ハッシュ回路321～324は、プリフェッチアドレスの入力を受ける。ハッシュ回路321～324は、予め記憶しているハッシュ関数を用いて、受信したプリフェッチアドレスのハッシュ値を算出する。そして、ハッシュ回路321～324それぞれの対応するウェイのいずれかでキャッシュがヒットしない場合、データ書込部101は、次の動作を行う。すなわち、データ書込部101は、ハッシュ回路321～324のうちのいずれかに対応するウェイにおけるそのハッシュ値を有するインデックスに応じたエントリにプリフェッチアドレスのタグ31をセットする。

20

【0079】

このようにすることで、デマンドアドレスのハッシュ値と各プリフェッチアドレスのハッシュ値が一致する。すなわち、いずれも同じ値を用いて算出されるハッシュ値となる。そのため、例えば、図4のように4ウェイの場合、1つのデマンドアドレス及び3つのプリフェッチアドレスを格納することで、4ウェイにおける同じ値を用いて算出されるハッシュ値を有するエントリは一応枯渇する。ただし、ウェイ毎にハッシュ関数が異なっているため、ウェイ毎に算出されるハッシュ値は異なり使用しているインデックスは異なっている。そこで、本実施例では、同じ値を用いて算出されるハッシュ値を有するエントリが枯渇しても他のウェイの同じインデックスのエントリにデータが置き換えられる。例えば、図4におけるp-1～p-4は、同じ値を用いて計算されたハッシュ値をインデックスとしているエントリに格納されている場合で考える。この場合、p-2と同じエントリにp-5のタグをセットしようとした場合、p-2が既にセットされているので、そのままではセットできない。また、他のウェイにおいても同じ値で算出されたハッシュ値を有するエントリには既にタグがセットされているので、そのままではp-5のタグをセットできない。そこで、本実施例に係るデータ書込部101は、p-2をウェイ#0、#2～#3のいずれかの同じインデックスを有するエントリに移動させることで、p-2のタグが格納されていたエントリにp-5のタグを格納する。このようなことから、1つのデマンドアドレス及び3つのプリフェッチアドレスを格納しても、未だ主記憶メモリ200との間でのデータのリプレースは起こらない。そこで、プリフェッチ部3は、最大で15個のプリフェッチアドレスを生成し、その生成された15個のプリフェッチアドレスがハッシュ値変更部61～64により格納されれば、キャッシュメモリに格納できないデータが発生し、データのリプレースが発生する。すなわち、本実施例に係るキャッシュメモリ装置は、デマンドアドレスとの間でハッシュ衝突を発生させるプリフェッチアドレスの生成及びデータの格納を繰り返すことで、最終的にキャッシュメモリのウェイ間置き換え処理が

30

40

50

できない状況に達することができる。これにより、本実施例に係るキャッシュメモリ装置は、主記憶メモリに対するデータのリプレースを発生させることができる。

【0080】

さらに、ハッシュ回路321～324は、求めた排他的論理和の結果をプリフェッチアドレスのハッシュ値として選択回路102へ出力する。また、ハッシュ回路321～324は、求めた排他的論理和の結果をプリフェッチアドレスのハッシュ値として選択回路131～134のへそれぞれ出力する。

【0081】

選択回路131～134は、デマンドアドレスのハッシュ値の入力をハッシュ回路2から受ける。また、選択回路131～134は、プリフェッチアドレスのハッシュ値の入力をハッシュ回路321～324からそれぞれ受ける。また、選択回路131～134は、格納用のデータの入力をデータ書込部101から受ける。そして、選択回路131～134は、ハッシュ回路2から受信したデマンドアドレスのハッシュ値及びハッシュ回路321～324から受信したプリフェッチアドレスのハッシュ値を有するインデックスをキャッシュタグアレイ71へ出力する。また、選択回路103は、データ書込部101から受信したデータのインデックスをキャッシュタグアレイ71へ出力する。

10

【0082】

選択回路102は、プリフェッチアドレスのハッシュ値であるハッシュ回路321～324が求めた排他的論理和の算出結果をハッシュ回路321～324から受信する。さらに、選択回路102は、ヒットしたウェイの情報を比較回路81～84から受信する。そして、選択回路102は、キャッシュデータアレイ72内のヒットしたウェイの情報及びハッシュ回路321～324から受信したハッシュ値の中で、ヒットしたウェイに対応するハッシュ値をデータ書込部101へ出力する。

20

【0083】

例えば、ハッシュ回路321から出力されたハッシュ値を有するインデックスにプリフェッチアドレスのタグ31がセットされた場合、ウェイ#0においてタグがヒットする。そこで、選択回路102は、ウェイ#0の情報を比較回路81から受信する。そして、選択回路102は、ハッシュ値変更部61から受信したハッシュ値及びウェイ#0の情報をデータ書込部101へ出力する。

【0084】

データ書込部101は、選択回路102からハッシュ値及びウェイの情報の入力を受ける。また、データ書込部101は、デマンドアドレスで指定されたデータがキャッシュミスヒットの場合、デマンドアドレスの入力をアドレス取得部1から受ける。また、データ書込み部101は、プリフェッチアドレスで指定されたデータがキャッシュミスヒットの場合、プリフェッチアドレスの入力をプリフェッチ部3から受ける。次に、データ書込部101は、主記憶メモリ200のデマンドアドレス又はプリフェッチアドレスからデータを取得する。そして、データ書込部101は、選択回路102から受信したインデックス及びウェイが示すキャッシュデータアレイ72のエントリに取得したデータを書込む。

30

【0085】

比較回路81～84は、デマンドアドレスのタグ及びプリフェッチアドレスのタグのうち現在セットされたタグの情報の入力をプリフェッチ部3から受ける。そして、比較回路81～84は、受信したタグが対応するウェイに存在するか否かを判定する。存在する場合には、比較回路81～84は、タグが存在するウェイの情報を選択回路102へ出力する。

40

【0086】

以上に説明したように、実施例2に係るキャッシュメモリ装置は、ウェイ毎に異なるハッシュ関数を持っている場合でもハッシュ衝突を容易に発生させることができ、キャッシュメモリの試験を迅速に実行することができる。

【0087】

また、以上の実施例2の説明では、キャッシュメモリのウェイ間でデータの置き換え処

50

理を行う場合を例に説明したが、この処理を行わず、単にウェイ毎にハッシュ関数が異なるキャッシュメモリにおいても、本実施例における各機能は同様の効果を有する。その場合、キャッシュメモリの試験の際に、キャッシュメモリのウェイ間でのデータの置き換え処理が行われる場合に比べて、より早くデータのリプレースを発生させることができる。

【0088】

(ハードウェア構成)

次に、図5を参照して、以上で説明した各実施例に係るキャッシュメモリ装置のハードウェア構成について説明する。図5は、キャッシュメモリ装置のハードウェア構成図である。

【0089】

本実施例に係るキャッシュメモリ装置は、CPU (Central Processing Unit) 401、メモリ402及びキャッシュ403を有している。また、キャッシュメモリ装置は、ハッシュ回路404、衝突検出回路405、ハッシュ値変更回路406、比較回路407及び選択回路408を有している。

10

【0090】

ハッシュ回路404は、図1に例示したハッシュ回路2及び4、並びに、図4に例示したハッシュ回路21~24及び41~44などの機能を実現する。衝突検出回路405は、図1に例示した衝突検出部5及び図4に例示した衝突検出部51~54などの機能を実現する。ハッシュ値変更回路406は、図1に例示したハッシュ値変更部6及び図4に例示したハッシュ値変更部61~64などの機能を実現する。比較回路407は、図1及び図4に例示した比較回路81~84などの機能を実現する。選択回路408は、図1及び図4に例示した選択回路10などの機能を実現する。

20

【0091】

メモリ402は、図1及び図4に例示した主記憶メモリ200の機能を実現する。

【0092】

また、キャッシュ403は、図1及び図4に例示したキャッシュメモリ7の機能を実現する。ここで、本実施例では、CPU401の外部にキャッシュ403を配置しているが、キャッシュ403は、CPU401の内部に配置されてもよい。

【0093】

さらに、CPU401、メモリ402及びキャッシュ403は、図1及び図4に示したアドレス取得部1、プリフェッチ部3及びデータ書込部101の機能を実現する。具体的には、メモリ402は、上規各機能などを実現するための各種プログラムを格納している。そして、CPU401は、メモリ402から各種プログラムを読み出し実行することで、アドレス取得部1、プリフェッチ部3及びデータ書込部101などの各機能を実現する。

30

【符号の説明】

【0094】

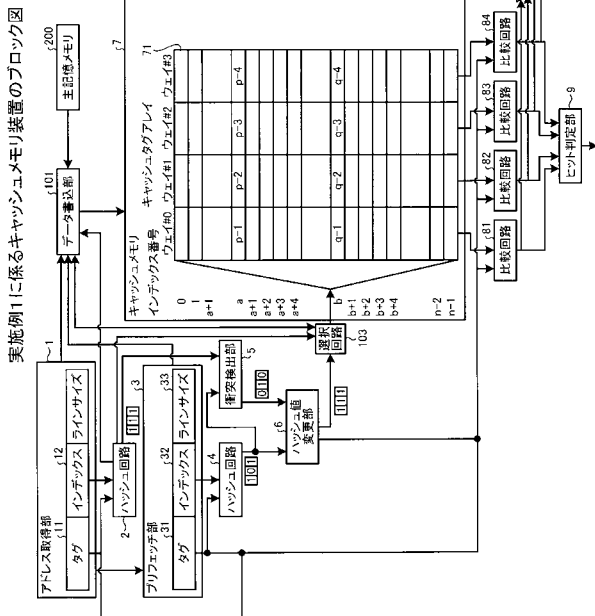
- 1 アドレス取得部
- 2 ハッシュ回路
- 3 プリフェッチ部
- 4 ハッシュ回路
- 5 衝突検出部
- 6 ハッシュ値変更部
- 7 キャッシュメモリ
- 9 ヒット判定部
- 10 選択回路
- 21~24 ハッシュ回路
- 41~44 ハッシュ回路
- 51~54 ハッシュ衝突検出部
- 61~64 ハッシュ値変更部
- 71 キャッシュタグアレイ

40

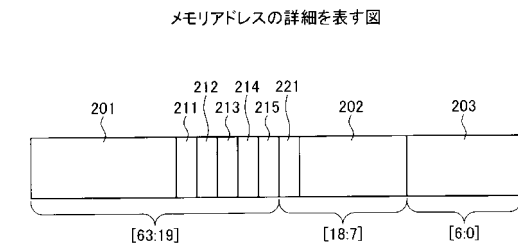
50

- 7 2 キャッシュデータアレイ
- 8 1 ~ 8 4 比較回路
- 1 0 1 データ書込部
- 1 0 2 選択回路
- 2 0 0 主記憶メモリ

【 図 1 】

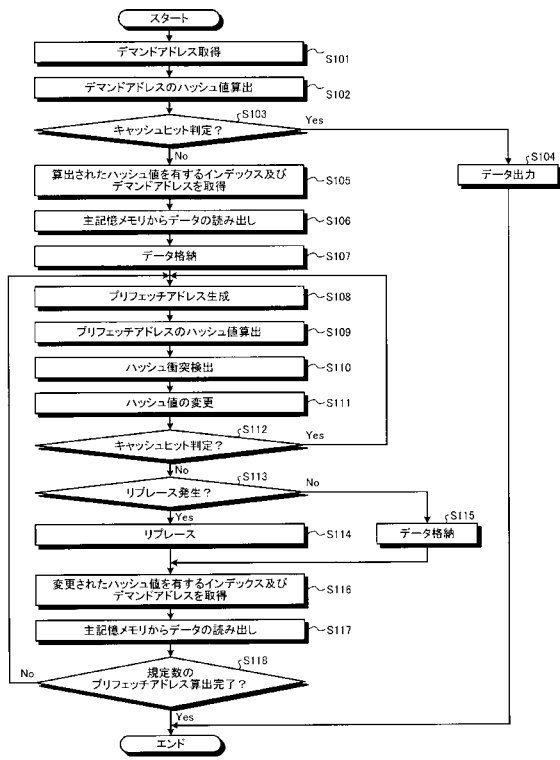


【 図 2 】

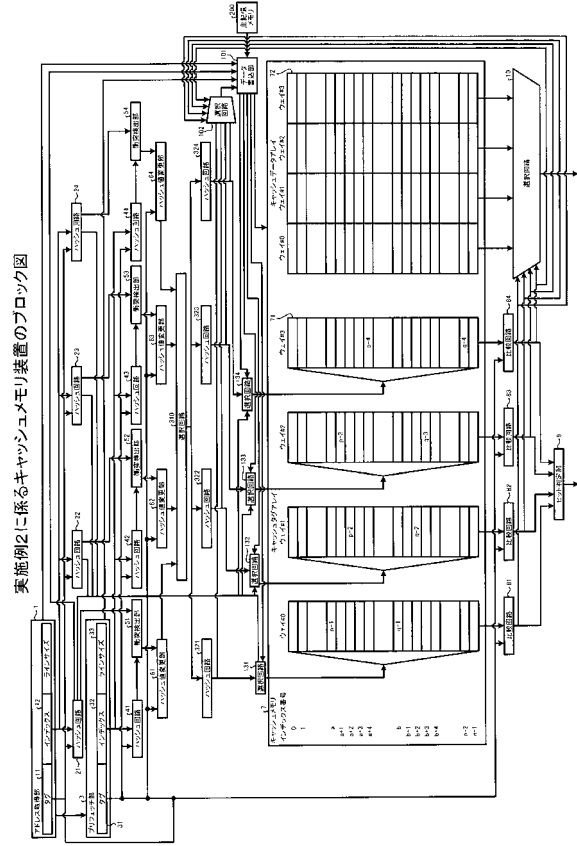


【 図 3 】

実施例1に係るキャッシュメモリ装置におけるデータ書込みの試験のフローチャート

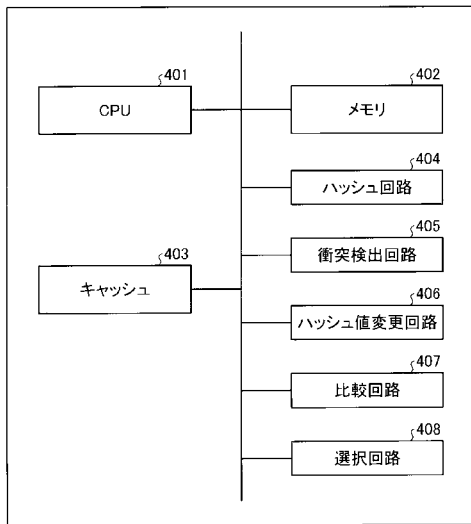


【 図 4 】

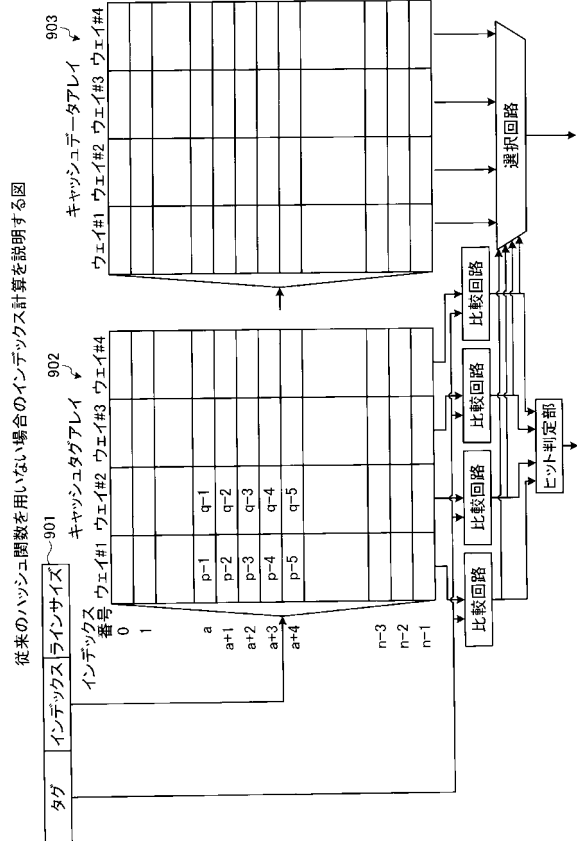


【 図 5 】

キャッシュメモリ装置のハードウェア構成図



【 図 6 】



【 図 7 】

従来のハッシュ関数を用いた場合のインデックス計算を説明する図

