(19)日本国特許庁(JP)　(12)特　許　公　報(B2)　(11)特許番号
特許第4454432号
(P4454432)
(45)発行日　平成22年4月21日(2010.4.21)　(24)登録日　平成22年2月12日(2010.2.12)

(51)Int.Cl.　F I
G06F 12/08 (2006.01)　G06F 12/08 541Z
G06F 12/08 501D
G06F 12/08 507Z
G06F 12/08 531Z
G06F 12/08 507H
請求項の数 32　(全 11 頁)　最終頁に続く

(21)出願番号　特願2004-236257 (P2004-236257)
(22)出願日　平成16年8月16日 (2004.8.16)
(65)公開番号　特開2005-182749 (P2005-182749A)
(43)公開日　平成17年7月7日 (2005.7.7)
審査請求日　平成18年9月20日 (2006.9.20)
(31)優先権主張番号　0329326.3
(32)優先日　平成15年12月18日 (2003.12.18)
(33)優先権主張国　英国(GB)

(73)特許権者　594154428
エイアールエム　リミテッド
イギリス国　シービー１　９エヌジェイ　ケンブリッジ，チェリー　ヒントン，フルバーン　ロード　１１０
(74)代理人　100066692
弁理士　浅村　皓
(74)代理人　100072040
弁理士　浅村　肇
(74)代理人　100094673
弁理士　林　鉐三
(74)代理人　100091339
弁理士　清水　邦明

最終頁に続く

(54)【発明の名称】キャッシュ・メモリおよびそのエラー訂正方法

(57)【特許請求の範囲】
【請求項１】
１行のデータおよびキャッシュ行について１つまたはそれ以上のエラー・チェック・ビットを記憶するように各々動作可能な複数の前記キャッシュ行と、
前記１つまたはそれ以上のエラー・チェック・ビットを使用してアクセスされる１つのキャッシュ行の少なくとも一部分をチェックするように動作できるエラー・チェック回路と、
前記エラー・チェック回路が前記キャッシュ行の部分にエラーを検出したならば、主メモリから少なくとも前記キャッシュ行の部分の補充をトリガするように動作できるエラー処理回路を含み、
各キャッシュ行もまた前記データの行のための１つまたはそれ以上のダーティ・ビットを記憶し、１つのセット・ダーティ・ビットは、前記主メモリから読取られてから前記データの行の対応する部分が変更されているが、いまだ前記主メモリへライト・バックされてないことを示し、
各キャッシュ行が各有効ビットについて１つのエラー・チェック・ビットを記憶し、前記エラー処理回路は、いずれかの有効ビットについていずれかのエラー・チェック・ビット内のエラーの検出に応答して、
（ｉ）　前記キャッシュ行のいずれかの部分にダーティ・ビットがセットされれば、前記主メモリへ前記キャッシュ行のライト・バックをトリガし、
（ｉｉ）　前記キャッシュ行のいずれの部分にもダーティ・ビットが何もセットされな

ければ、前記主メモリから前記キャッシュ行の補充をトリガするキャッシュ・メモリ。
【請求項２】
請求項１に記載のキャッシュ・メモリにおいて、前記補充は、前記キャッシュ行の部分を含むキャッシュ行全体の補充であるキャッシュ・メモリ。
【請求項３】
請求項１に記載のキャッシュ・メモリにおいて、前記キャッシュ行の部分に対応するいずれかのダーティ・ビットがセットされている場合は、前記エラー処理回路が前記補充をトリガしないキャッシュ・メモリ。
【請求項４】
請求項１に記載のキャッシュ・メモリにおいて、単一のダーティ・ビットがデータの一行全体に対応するキャッシュ・メモリ。
【請求項５】
請求項１に記載のキャッシュ・メモリにおいて、複数のダーティ・ビットが前記データの行の複数の部分のそれぞれ１つに対応するキャッシュ・メモリ。
【請求項６】
請求項１に記載のキャッシュ・メモリにおいて、複数のエラー・チェック・ビットが１つのキャッシュ行に対応するキャッシュ・メモリ。
【請求項７】
請求項１に記載のキャッシュ・メモリにおいて、前記１つまたはそれ以上のエラー・チェック・ビットが、
パリティ・ビット、
重複ビット、
チェックサム・ビット、および
エラー訂正コード・ビット
の１つまたはそれ以上であるキャッシュ・メモリ。
【請求項８】
請求項３に記載のキャッシュ・メモリにおいて、前記キャッシュ行の部分に対応するいずれかのダーティ・ビットがセットされれば、前記エラー処理回路がソフトウェア処理ルーチンの実施をトリガするキャッシュ・メモリ。
【請求項９】
請求項１に記載のキャッシュ・メモリにおいて、前記データの行が１つまたはそれ以上のデータ・ワードとデータＴＡＧを含み、またエラー・チェック回路が前記データＴＡＧ内にエラーを発見するならば、前記エラー処理回路が前記キャッシュ行の補充をトリガするキャッシュ・メモリ。
【請求項１０】
請求項１に記載のキャッシュ・メモリにおいて、前記キャッシュ・メモリはマルチプル・ウェイ・キャッシュ・メモリであって、前記データの行が１つまたはそれ以上のデータ・ワードとデータＴＡＧを含み、前記キャッシュ行について１つよりも多いＴＡＧの一致が起きたときに前記エラー・チェック回路が前記データＴＡＧ内のエラーを検出するならば、前記キャッシュ行が無効にされるキャッシュ・メモリ。
【請求項１１】
請求項１に記載のキャッシュ・メモリにおいて、各キャッシュ行は前記１つまたはそれ以上のダーティ・ビットについて少なくとも１つのエラー・チェック・ビットを記憶し、また前記エラー処理回路は前記１つまたはそれ以上のダーティ・ビットについてエラー・チェック・ビット内にエラーを検出するのに応答して、前記主メモリへ向けて前記キャッシュ行のライト・バックをトリガするキャッシュ・メモリ。
【請求項１２】
請求項１に記載のキャッシュ・メモリにおいて、各キャッシュ行が１つの有効ビットを記憶し、１つのセット有効ビットが有効データを記憶する前記キャッシュ行を示すキャッシュ・メモリ。

【請求項１３】
請求項１に記載のキャッシュ・メモリにおいて、各キャッシュ行が複数の有効ビットを記憶し、１つのセット有効ビットが有効データを記憶する前記キャッシュ行の対応する部分を示すキャッシュ・メモリ。
【請求項１４】
請求項１２に記載のキャッシュ・メモリにおいて、対応する有効ビットがノン・エラーであって無効性を示すキャッシュ行内で検出されるエラーは、無視されるキャッシュ・メモリ。
【請求項１５】
請求項１に記載のキャッシュ・メモリにおいて、
（ｉ） データ・キャッシュ・メモリ、
（ｉｉ） 命令キャッシュ・メモリ、
（ｉｉｉ） ユニファイド・キャッシュ・メモリ、および、
（ｉｖ） Ｈａｒｖａｒｄ・キャッシュ・メモリの内の１つであるキャッシュ・メモリ。
【請求項１６】
請求項１に記載のキャッシュ・メモリにおいて、前記１つまたはそれ以上のエラー・チェック・ビットの少なくともいくつかは、前記キャッシュ行が書き込まれる時に更新されるキャッシュ・メモリ。
【請求項１７】
キャッシュ・メモリ内のソフト・エラーを訂正する方法であって、
複数のキャッシュ行の各々の中に１行のデータと、前記キャッシュ行のための１つまたはそれ以上のエラー・チェック・ビットを記憶するステップと、
前記１つまたはそれ以上のエラー・チェック・ビットを使用して、アクセスされる１つのキャッシュ行の少なくとも一部分をエラー・チェックするステップと、
前記エラー・チェックが前記キャッシュ行の部分内にエラーを検出するならば、ハードウエアが主メモリから前記キャッシュ行の少なくとも部分を補充することをトリガするステップを含み、
各キャッシュ行はまた前記データの行について１つまたはそれ以上のダーティ・ビットを記憶し、１つのセット・ダーティ・ビットは、前記主メモリから読取られてから前記データの行の対応する部分が変更されているが、いまだ前記主メモリにライト・バックされていないことを示し、
各キャッシュ行が各有効ビットについて１つのエラー・チェック・ビットを記憶し、いずれかの有効ビットについていずれかのエラー・チェック・ビット内でのエラー検出に応答して、
（ｉ） 前記キャッシュ・メモリのいずれかの部分についてダーティ・ビットがセットされれば、前記主メモリへの前記キャッシュ行のライト・バックがトリガされ、また、
（ｉｉ） 前記キャッシュ・メモリのいずれの部分にもダーティ・ビットが何もセットされなければ、前記主メモリからの前記キャッシュ行の補充がトリガされる方法。
【請求項１８】
請求項１７に記載の方法において、前記補充は、前記キャッシュ行の部分を含む前記キャッシュ行全体の補充である方法。
【請求項１９】
請求項１７に記載の方法において、前記キャッシュ行の部分に対応するいずれかのダーティ・ビットがセットされている場合は、前記補充がトリガされない方法。
【請求項２０】
請求項１７に記載の方法において、単一のダーティ・ビットがデータの１行全体に対応する方法。
【請求項２１】
請求項１７に記載の方法において、複数のダーティ・ビットが、前記データの行の複数

の部分のそれぞれに対応する方法。
【請求項２２】
請求項１７に記載の方法において、複数のエラー・チェック・ビットが１つのキャッシュ行に対応する方法。
【請求項２３】
請求項１７に記載の方法において、前記１つまたはそれ以上のエラー・チェック・ビットが、
パリティ・ビット、
重複ビット、
チェック・サム・ビット、
エラー訂正コード・ビット、
の１つまたはそれ以上である方法。
【請求項２４】
請求項１９に記載の方法において、前記キャッシュ行の部分に対応するいずれかのダーティ・ビットがセットされれば、ソフトウェア処理ルーチンの実行がトリガされる方法。
【請求項２５】
請求項１７に記載の方法において、前記データの行が１つまたはそれ以上のデータ・ワードおよびデータＴＡＧを含み、前記データＴＡＧ内にエラーが検出されれば、前記キャッシュ行の補充がトリガされる方法。
【請求項２６】
請求項１７に記載の方法において、前記キャッシュ・メモリはマルチプル・ウェイ・キャッシュ・メモリであり、前記データの行は１つまたはそれ以上のデータ・ワードおよびデータＴＡＧを含み、前記キャッシュ行について１つよりも多いＴＡＧ一致が起きたときに、前記データＴＡＧ内にエラーを検出すれば、前記キャッシュ行が無効にされる方法。
【請求項２７】
請求項１７に記載の方法において、各キャッシュ行が前記１つまたはそれ以上のダーティ・ビットについて少なくとも１つのエラー・チェック・ビットを記憶し、前記１つまたはそれ以上のダーティ・ビットについてエラー・チェック・ビット内のエラー検出に応答して、前記主メモリへ前記キャッシュ行のライト・バックがトリガされる方法。
【請求項２８】
請求項１７に記載の方法において、各キャッシュ行は１つの有効ビットを含み、前記キャッシュ行が有効データを記憶することをセット有効ビットが指示する方法。
【請求項２９】
請求項１７に記載の方法において、各キャッシュ行は複数の有効ビットを含み、前記キャッシュ行の対応する部分が有効データを記憶することをセット有効ビットが指示する方法。
【請求項３０】
請求項２８に記載の方法において、対応する有効ビットがノン・エラーであって無効性を示すキャッシュ行内で検出されたエラーは無視される方法。
【請求項３１】
請求項１７に記載の方法において、前記キャッシュ・メモリが、
（ｉ） データ・キャッシュ・メモリ、
（ｉｉ） 命令キャッシュ・メモリ、
（ｉｉｉ） ユニファイド・キャッシュ・メモリ
（ｉｖ） Ｈａｒｖａｒｄ・キャッシュ・メモリ
のうちの１つである方法。
【請求項３２】
請求項１７に記載の方法において、前記キャッシュ行が書き込まれたときに、前記１つまたはそれ以上のエラー・チェック・ビットが更新される方法。
【発明の詳細な説明】

【技術分野】
【０００１】
本発明はデータ処理システムの分野に関する。特に、本発明は、キャッシュ・メモリ内に保持されるビット値に起こるエラー、たとえばイオン化放射により引き起こされるソフト・エラーなどのエラーの訂正に適したキャッシュ・メモリおよびエラー訂正方法に関する。
【背景技術】
【０００２】
処理速度を改良するためにデータ処理システムにキャッシュ・メモリを備えることが知られている。これらのキャッシュ・メモリは典型的に記憶容量がより大きくなりつつあり、集積回路内の他の要素と共に、より小さな回路ジオメトリを用いて製作される傾向になっている。この結果として、キャッシュ・メモリは、電離放射線の影響または通過により起こるような、いわゆる「ソフト・エラー」を、ますます受けやすくなってきている。そうしたソフト・エラーは、キャッシュ・メモリ内に記憶された１つまたはそれ以上のビット値を不適当に変更させる。
【０００３】
キャッシュ・メモリ内のソフト・エラーを処理する公知のシステムが、米国特許第６，３３２，１８１号に説明されている。このシステムにおいて、エラーが検出されれば、ソフトウェア・エラー処理メカニズムがトリガされる。このタイプのエラー検出およびエラー訂正メカニズムが望ましいクリティカル・システムの関係において、そうしたエラーを処理するソフトウェア・メカニズムが遅いことは、重大な短所であって、ソフトウェア内でそうした１つのエラーが処理される間に数百または数千の処理サイクルにわたって、たとえばアンチ・ロック・ブレーキ・システムまたはエンジン・システムが、その正常な機能を使用不能になる。
【発明の開示】
【発明が解決しようとする課題】
【０００４】
このタイプのエラーは、たとえばアンチ・ロック・ブレーキ・システムを制御するプロセッサ内のクリティカルで安全関連のシステムにおいて、特に重要である。そうしたシステムでは、高度な安全性が要求され、証明できなければならない。
【課題を解決するための手段】
【０００５】
１つの面から見ると、本発明はキャッシュ・メモリを供給し、このキャッシュ・メモリは、
１行のデータおよびキャッシュ行について１つまたはそれ以上のエラー・チェック・ビットを記憶するように各々動作可能な複数の前記キャッシュ行と、
前記１つまたはそれ以上のエラー・チェック・ビットを使用して、アクセスされる１つのキャッシュ行の少なくとも一部分をチェックするように動作できるエラー・チェック回路と、
前記エラー・チェック回路が前記キャッシュ行の部分にエラーを検出したならば、主メモリから少なくとも前記キャッシュの部分の補充をトリガするように動作できるエラー処理回路を含む。
【０００６】
本発明はそうしたエラーの純粋にソフトウェア的な処理に関連する問題を認識して、それによりそうしたエラーをハードウェア内に検出するだけでなく、単純なキャッシュ行補充を用いてハードウェア内でそうしたエラーを修復することもできる。キャッシュ・メモリの性質として、キャッシュ・メモリ内で改変されていない間にデータのコピーを保持することが多く、これは主メモリ内にあるので、損傷したキャッシュ行に迅速に補充することが容易であり、発生したエラーを効果的に修復できる。キャッシュ行を補充するメカニズムは正常な運転中に使用するために典型的に既設されているので、このハードウェア・

エラー処理メカニズムはまた有利にも僅かな追加回路しか必要としない。ハードウェアのために必要な追加回路はまた、このハードウェアにより置き換えられた機能の例外処理コードを記憶するのに必要な追加回路よりも少ない。

【０００７】

キャッシュ行の補充はこのエラーを含む部分の部分的な補充であり得ることが、理解できるが、多くの実施例において、エラーになっている部分があるキャッシュ行全体を単に補充するだけが、有利に簡便である。

【０００８】

キャッシュの内容が変更済みであって、主メモリへまだライト・バックされていないことを示すために、ダーティ・ビットが使用されているキャッシュをライト・バックする状況において、追加の複雑さが起きる。そうしたシステムの状況において、キャッシュ行へ補充されるデータが期限切れであるために、関係するキャッシュ行またはキャッシュ行の部分がダーティであると示されれば、エラー処理回路は補充をトリガしない。

【０００９】

キャッシュ行全体のために単一のダーティ・ビットが供給されることもあり、または、代わりに、キャッシュ行ごとおよびキャッシュ行データの各部分に対応して、２つまたはそれ以上のダーティ・ビットが供給されることもある。

【００１０】

エラー・チェック・ビットにより占められる空間と、それらがエラーを識別できるようにするための粒度との間に、バランスを得る必要がある。本発明の好ましい実施例において、各キャッシュ行のために複数のエラー・チェック・ビットが供給される。

【００１１】

エラー・チェック・ビットは、それらがエラー・チェック能力を加えるキャッシュ行内のビットの特定の性質によって、さまざまな異なった形を取り得るし、特に、パリティ・ビット、複写ビット、チェック・サム・ビットおよび／またはエラー訂正コード・ビットであり得る。

【００１２】

前に指摘したように、エラーを含むキャッシュ行の一部分の内にダーティ・ビットがセットされれば、単なる補充は不可能であり、そうした場合システムは、一層複雑な解決方法を取るソフトウエア・エラー処理をトリガするように働くか、または代わりに、エラーが修復不可能であるとして、全システムを単純にリセットする。

【００１３】

キャッシュ・メモリの状況内で理解すべきことは、データＴＡＧが提示されて、そうしたデータＴＡＧ内にエラーが起こるならば、誤ったＴＡＧと共にそれを使用できないので関連のキャッシュ行が補充され、その空間は他のデータにより使用でき、たとえばエラー・チェックがキャッシュ・アクセスに際して遂行される場合は、壊れたＴＡＧを有する行へのヒットは、そのＴＡＧを有する行が本当に必要とされていることを示し、誤ってヒットを与えられた壊れたＴＡＧにより、このキャッシュ行へ有利に補充されることを示す。

【００１４】

以前に指摘したダーティ・ビットは、それ自体エラーを受けやすく、エラー・チェック・ビットを備えている。そうしたエラーが起こると、主メモリへのキャッシュ行のライト・バックがトリガされる。

【００１５】

キャッシュ行はまた関連の有効ビットを有し、それは全キャッシュ行のための有効ビット、または各キャッシュ行のための複数の有効ビットである。そうした有効ビットはエラーを受けやすく、対応するエラー・チェック・ビットを備えている。有効ビット内にエラーが検出されると、キャッシュ行がダーティな場合はライト・バックがトリガされ、またキャッシュ行がダーティでなければ、このキャッシュ行の補充がトリガされる。

【００１６】

キャッシュ・メモリは、データ・キャッシュ・メモリ、命令キャッシュ・メモリ、ユニ

ファイド・キャッシュ・メモリまたはＨａｒｖａｒｄ・キャッシュ・メモリを含む多くの異なった形を取り得る。

【００１７】

エラー・チェック・ビットは、キャッシュ行が書き込まれるときに計算され、キャッシュ行内へ更新されるのが便利である。

【００１８】

本発明のもう１つの面から見ると、キャッシュ・メモリ内のソフト・エラーを訂正する方法を供給し、前記方法は、

複数のキャッシュ行の各々の中に１行のデータと、前記キャッシュ行のための１つまたはそれ以上のエラー・チェック・ビットを記憶するステップと、

前記１つまたはそれ以上のエラー・チェック・ビットを使用して、アクセスされる１つのキャッシュ行の少なくとも一部分をエラー・チェックするステップと、

前記エラー・チェックが前記キャッシュ行の部分内にエラーを検出するならば、ハードウエアが主メモリから前記キャッシュ行の少なくとも部分を補充することをトリガするステップを含む。

【００１９】

本発明の上記およびその他の諸目的、特徴および利点は、添付図面と共に下記の例示的実施例の詳細な説明を読めば一層明らかになる。

【実施例】

【００２０】

図１は、キャッシュ・メモリ２を図示している。この図示された例は、４ウェイ・データ・キャッシュ・メモリである。この技法は、シングル・ウェイ・キャッシュおよびマルチプル・ウエイ・キャッシュの両方に使用できる。この技法はまた、ノン・アソシエイティブ・キャッシュ、命令キャッシュ、Ｈａｒｖａｒｄ・キャッシュ、ユニファイド・キャッシュなどのような他のタイプのキャッシュ・メモリにも応用できる。図１に示すように、キャッシュ２は、データ部４、ＴＡＧ部６、有効ビット部８およびダーティ・ビット部１０を含む。このキャッシュ配列の基本的なタイプは、そうしたキャッシュの補充、ライト・バックおよび制御のためのノン・エラー関連メカニズムと共に当技術分野に知られているので、本書ではさらに詳細な説明はしない。

【００２１】

本発明は、エラー・チェック・ビットを各キャッシュ行へ加えることにより、キャッシュ・メモリの既知の形式上に構築する。一層詳細には、パリティ・ビット１２をデータ部４へ加える。この実施例において、ダブル・ワードごとに８個のパリティ・ビットがあり、３２バイト・キャッシュ行に３２個のパリティ・ビットを与える。パリティ・ビット１４はまたＴＡＧ部６へ加えられる。ＴＡＧ部６はデータ部４よりも遥かに小さいので、ＴＡＧ部６に関して３個のパリティ・ビットのみが必要である。複写有効ビット１６および複写ダーティ・ビット１８もまた、それぞれキャッシュ２の有効ビット部８およびダーティ・ビット部１０に加えられる。キャッシュ２がアクセスされて、そこからある値が読み取られると、キャッシュ２はエラー・チェック・ビット１２、１４、１６、１８を使用して、エラー・チェック回路２０によりエラー・チェックされる。パリティ・ビット１２、１４は、ＸＯＲツリーを使用するパリティを表現する対応の８ビットに対してチェックされる。複写有効ビット１６および複写ダーティ・ビット１８は、それらの対応する有効ビットおよびダーティ・ビットに対してチェックされる。検出されるエラーおよび関連キャッシュ行の状態により、エラー・チェック回路２０はまたエラー処理回路として働いて、たとえば適当な応答をトリガし、キャッシュ行補充またはライト・バック動作またはソフト・エラー・アボート信号の発行を行い、ソフト・エラー・アボート専用の割り込みハンドラをトリガする。

【００２２】

図２は、エラー検出およびエラー処理回路２０によりトリガされるさまざまな応答を示す表である。特定の補充動作において、関係する行のライト・バック（ＷＢ）動作と無効

化が、ソフト・エラー・アボート信号の発行と共にトリガされる。いくつかの状況においては、関係する行が無効なので、あるエラー信号が無視されることもある。

【００２３】

この表の上部の２つの入力は、関係するキャッシュ行が有効でダーティでない状況において、データまたはＴＡＧ内のエラーの検出を示す。データ・エラーの場合は補充動作が遂行されて、これにより主メモリから正しいデータが取られて、エラーのあるキャッシュ行へ補充される。これは迅速で効果的であって、処理時間の僅かな損失と正しい修復を結果する。エラー内にＴＡＧがある場合は、キャッシュ行の補充が遂行されるが、これは、プロセッサによりデータが要求されたが、ＴＡＧが既に壊れているので、ＴＡＧの一致が誤っているために、キャッシュへのヒットが発生したからである。しかし、データはなお要求されているので、データの補充は誤ったキャッシュ行を除去して、探しているデータへの要求を満足させる。これの例外は、マルチプル・ウェイがＴＡＧの一致を報告する場合で、この状況においては、一致が起きた他の誤りでないキャッシュ・ウェイにおいて、関連のデータが既に存在するので、誤りのＴＡＧを保持するキャッシュ行は充填されるよりも、むしろ無効にされるべきである。

【００２４】

表の５行目と６行目は、キャッシュ行が有効およびダーティである時に起きるそれぞれのデータ・エラーおよびＴＡＧエラーを示す。これらのタイプのエラーは、修復が容易でなく、システムまたは少なくとも関連の処理をリセットするなどして、これらのエラーをソフトウエアが処理できるようにするために、ハードウエアはソフト・エラー・アボート割り込み信号を発生する。

【００２５】

データおよびＴＡＧは、キャッシュ行に関連するビット記憶領域の主要部分を物理的に占有するので、これらは、ソフト・エラーが最も起こるであろう場所である。しかし、キャッシュ行の有効ビットまたはダーティ・ビット内にも、ソフト・エラーが起こり得る。図２は、これらの場合に取られる動作を図示する。理解すべきは、多重エラーの同時的な発生は、統計的に非常に起こり難い発生であって、本技法の実用的な実施例は、そうした発生およびそれに対する極端にまれな応答を取り扱わないだけである。１つの代案は、そうした状況では、多重エラーの発生を検出してシステムを強制的にリセットすることである。

【００２６】

図３は、発生するエラーをどのように分析して、図２のそれに従って適当な動作のコースを検出するかを、図式的に示すフローチャートである。何らかの形のエラーが検出されると、これらは更に検査されて、ダーティ・ビット・エラーが発生したかどうか、有効ビットが発生したかどうか、またはこのエラーが必然的にデータまたはＴＡＧのエラーでなければならないかどうかが決定される。ダーティ・ビット・エラーを処理する場合は、有効ビットもまたダーティ・ビット・エラーが無視されるのと違った仕方で、セットされる。同じことがデータまたはＴＡＧのエラーの訂正においても、真実である。ステップ２２に達する処理により示されるデータまたはＴＡＧエラーの場合は、キャッシュ行がダーティでなければ、ハード・ウエア・メカニズムにより制御され遂行されて、ステップ２４でキャッシュ行補充動作がトリガされる。

【００２７】

図４は、図２に示した適当な状況において、種々なエラー応答をトリガするためのエラー検出回路およびエラー処理回路２０内に備えられる回路を図示する。発生する信号の論理的な組み合わせが比較的に簡単であり、このことがこの技法を効率的にし、また迅速に実施するようにしていることが理解される。

【００２８】

図５は、図４の回路により遂行される評価の代数的表現である。

【００２９】

添付図面を参照しながら本発明の例示的実施例を、本書に詳細に説明してきたが、本発

明はこれらの実施例に厳密に制限されるものではなく、添付の特許請求の範囲により定義される本発明の範囲と精神から離れることなく、当業者がそれに種々の変更と修正を加えることが可能であることを理解すべきである。

【図面の簡単な説明】

【００３０】

【図１】エラー・チェック・ビット、エラー検出回路およびエラー処理回路を組み込んだ４ウェイ・キャッシュ・メモリを図式的に示す。

【図２】特定エラーが発生した場合にトリガされるアクションを示す表である。

【図３】エラー処理選択メカニズムの単純化された形式を図式的に示すフローチャートである。

【図４】種々のエラー条件の組み合わせが発生したときに、異なった応答をトリガするハードウェア回路を図示する。

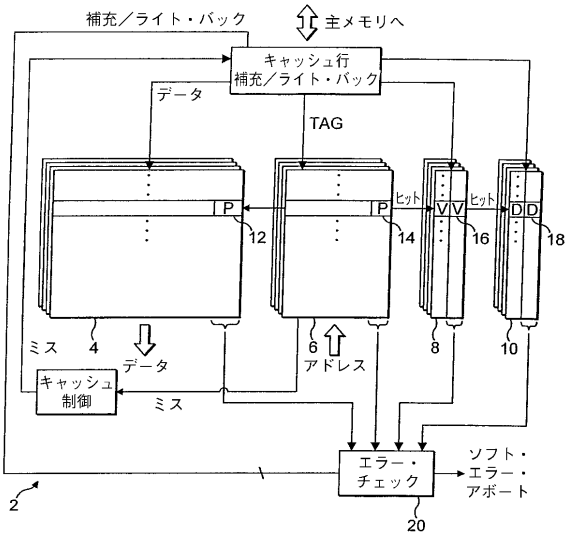
【図５】図４の回路により遂行される評価の代数的表現である。

【符号の説明】

【００３１】

２　キャッシュ・メモリ
４　データ部
６　ＴＡＧ部
８　有効ビット部
１０　ダーティ・ビット部
１２　データ部４のパリティ・ビット
１４　ＴＡＧ部６のパリティ・ビット
１６　有効ビット部８の複写有効ビット
１８　ダーティ・ビット部１０の複写有効ビット
２０　エラー・チェック回路

【図１】

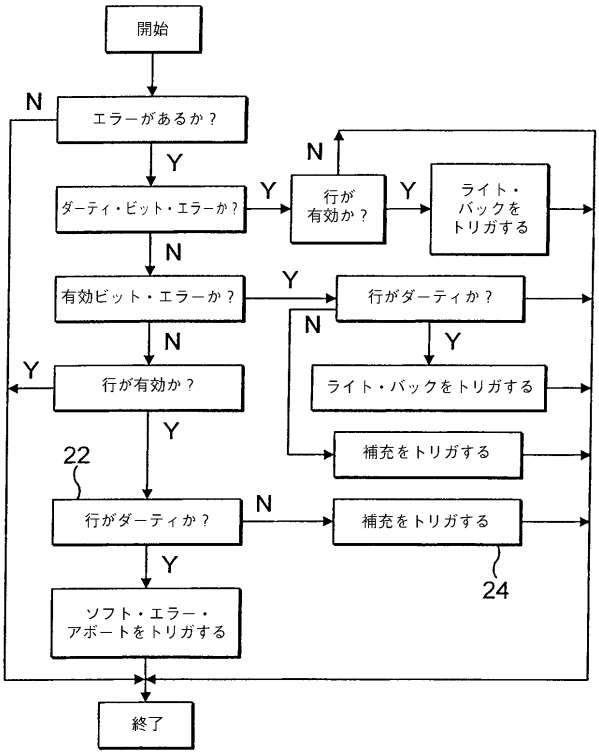


【図２】

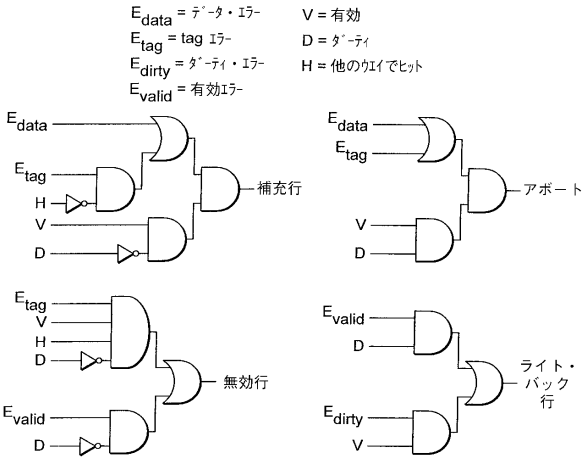
データ・エラー	TAG エラー	有効 エラー	ダーティ・エラー	有効	ダーティ	動作
✓	-	-	-	✓	X	補充
-	✓	-	-	✓	X	補充
✓	-	-	-	X	X	無視
-	✓	-	-	X	X	無視
✓	-	-	-	✓	✓	アボート
-	✓	-	-	✓	✓	アボート
-	-	✓	-	?	X	無効化
-	-	✓	-	?	✓	ライト・バック
-	-	-	✓	X	?	無視
-	-	-	✓	✓	?	ライト・バック

注：
有効＋ダーティ
は不可能

【図３】



【図４】



【図５】

補充 = $E_{data}.V.\overline{D} \vee E_{tag}.V.\overline{D}.\overline{H}$

アボート = $E_{data}.V.D \vee E_{tag}.V.D$

無効化 = $E_{valid}.\overline{D} \vee E_{tag}.V.\overline{D}.H$

ライト・バック = $E_{valid}.D \vee E_{dirty}.V.D$

フロントページの続き

(51)Int.Cl. FI
G06F 12/08 501C

(72)発明者 デーヴィッド、ケヴィン、ハート
イギリス国、ケンブリッジ、チェリー ヒントン、 ローリス コート 33
(72)発明者 パトリック ジェラード マックグレウ
イギリス国、ケンブリッジ、ウィリングハム、 ヘーデン ウェイ 37
(72)発明者 アンドリュー バーダス
イギリス国、ケンブリッジシャー、イーリー、 ウィリアムズ クローズ 51

審査官 清木 泰

(56)参考文献 特開平10－240628（JP, A）
特開平07－248970（JP, A）
特開平02－123440（JP, A）
特開2003－216493（JP, A）
特開平11－039217（JP, A）
特表平09－511088（JP, A）
特開平09－016473（JP, A）
特開平07－168760（JP, A）
特開平06－124244（JP, A）
特開平06－110783（JP, A）
特開平05－265859（JP, A）
特開平05－165719（JP, A）
特開昭64－012348（JP, A）
特開昭60－179858（JP, A）
特開昭54－011647（JP, A）
国際公開第2005／010760（WO, A1）

(58)調査した分野(Int.Cl., DB名)
G06F12／08－12／12