

(19) 日本国特許庁 (JP)

(12) 特 許 公 報 (B2)

(11) 特許番号

特許第6724043号
(P6724043)

(45) 発行日 令和2年7月15日 (2020.7.15)

(24) 登録日 令和2年6月26日 (2020.6.26)

(51) Int. Cl.

F I

G 0 6 F 12/0895 (2016.01)

G 0 6 F 12/0895 1 0 0

請求項の数 15 (全 19 頁)

(21) 出願番号	特願2017-559370 (P2017-559370)	(73) 特許権者	507364838
(86) (22) 出願日	平成28年4月8日 (2016.4.8)		クアルコム、インコーポレイテッド
(65) 公表番号	特表2018-519570 (P2018-519570A)		アメリカ合衆国 カリフォルニア 921
(43) 公表日	平成30年7月19日 (2018.7.19)		21 サン ディエゴ モアハウス ドラ
(86) 国際出願番号	PCT/US2016/026664		イブ 5775
(87) 国際公開番号	W02016/186747	(74) 代理人	100108453
(87) 国際公開日	平成28年11月24日 (2016.11.24)		弁理士 村山 靖彦
審査請求日	平成31年3月15日 (2019.3.15)	(74) 代理人	100163522
(31) 優先権主張番号	14/716,947		弁理士 黒田 晋平
(32) 優先日	平成27年5月20日 (2015.5.20)	(72) 発明者	ヘンリー・アーサー・ペレリン・ザ・サー
(33) 優先権主張国・地域又は機関	米国 (US)		ド
			アメリカ合衆国・カリフォルニア・921
			21-1714・サン・ディエゴ・モアハ
			ウス・ドライブ・5775
			最終頁に続く

(54) 【発明の名称】 キャッシュタグ圧縮のための方法および装置

(57) 【特許請求の範囲】

【請求項 1】

キャッシュメモリであって、

複数のタグ圧縮エントリを記憶するタグ圧縮構造であって、前記複数のタグ圧縮エントリのうちの少なくとも1つが、少なくとも2つのメモリタグによって共有される所定の長さの高位ビットの一意のセットを含み、それぞれのインデックス値によって識別される、タグ圧縮構造と、

複数のタグアレイエントリを記憶するタグアレイであって、前記複数のタグアレイエントリの各々が、前記メモリタグのうちの1つの所定の長さの低位ビットのそれぞれのセットと、インデックス値とを含み、各タグアレイエントリの前記インデックス値が、それぞれの前記タグアレイエントリの前記メモリタグの前記高位ビットを含む前記タグ圧縮構造エントリの前記インデックス値に対応し、前記複数のメモリタグの各々が、データアレイ内に記憶されたデータ要素のそれぞれのメモリアドレスの少なくとも一部分を備える、タグアレイと

を備える、キャッシュメモリ。

【請求項 2】

第1のタグ圧縮エントリ内の前記メモリタグの高位ビットの前記セットが入力メモリアドレスの高位ビットのセットと一致するとの決定時に、前記タグ圧縮構造から前記第1のタグ圧縮エントリに対応する前記インデックス値を返すことと、

前記入力メモリアドレスの低位ビットの前記セットが第1のタグアレイエントリ内の前

10

20

記メモリタグの低位ビットの前記セットと一致するとの決定時に、前記第1のタグアレイエントリ内に記憶された前記インデックス値および前記メモリタグの低位ビットの前記セットを返すことと、

前記第1のタグ圧縮エントリの前記返されたインデックス値が前記第1のタグアレイエントリの前記返されたインデックス値と一致すると決定することと

によって、前記入力メモリアドレスに関する前記タグアレイ内のヒットの指示を返すように構成された論理をさらに備える、請求項1に記載のキャッシュメモリ。

【請求項3】

入力メモリアドレスの高位ビットのセットが、前記複数のタグ圧縮エントリ内の高位ビットの前記セットのうちのいずれにも一致しないこと、

10

前記入力メモリアドレスの低位ビットのセットが、前記複数のタグアレイエントリ内に記憶された低位ビットの前記セットのうちのいずれにも一致しないこと、

前記入力メモリアドレスの前記低位ビットと一致する低位ビットのセットを記憶するタグアレイエントリから返されたインデックス値が前記入力メモリアドレスの前記高位ビットと一致する高位ビットのセットを記憶するタグ圧縮エントリのインデックス値と一致しないこと

のうちの少なくとも1つの決定時に、前記入力メモリアドレスに関する前記タグアレイ内のミスの指示を返すように構成された論理をさらに備える、請求項1に記載のキャッシュメモリ。

【請求項4】

20

前記タグアレイ内の前記ミスの前記指示を返すことに応じて、

メモリ階層の次のレベルにおいて記憶されたデータを取り出すことと、

前記データを前記データアレイの第1のエントリ内に記憶することと、

前記入力メモリアドレスの高位ビットのセットを前記タグ圧縮構造内の第2のタグ圧縮エントリ内に記憶することと、

(i)前記入力メモリアドレスの低位ビットのセット、および(ii)前記第2のタグ圧縮エントリに対応するインデックス値を前記タグアレイの第2のタグアレイエントリ内に記憶することであって、前記第2のタグアレイエントリが前記データアレイの前記第1のエントリに対応する、記憶することと

を行うように構成された論理をさらに備える、請求項3に記載のキャッシュメモリ。

30

【請求項5】

前記タグ圧縮構造内の前記エントリが、(i)プロセスのアドレス空間識別子(ASID)、(ii)セキュリティビットのセット、および(iii)例外レベルビットのセットのうちの1つまたは複数を記憶するようにさらに構成される、請求項1に記載のキャッシュメモリ。

【請求項6】

第1のタグアレイエントリおよび第2のタグアレイエントリが各々、第1のタグ圧縮エントリに対応するインデックス値を特定し、前記第2のタグアレイエントリの低位ビットのセットが、前記第1のタグアレイエントリの低位ビットのセットと一致しない、請求項1に記載のキャッシュメモリ。

【請求項7】

40

前記タグ圧縮構造が、フルアソシアティブコンテンツアドレスサブルメモリを備え、前記タグアレイが、(i)データキャッシュ、(ii)トランスレーションルックアサイドバッファ、(iii)ラインフィルバッファ、および(iv)命令キャッシュのうちの少なくとも1つの一部を備える、請求項1に記載のキャッシュメモリ。

【請求項8】

前記キャッシュメモリが集積回路内に配設される、請求項1に記載のキャッシュメモリ。

【請求項9】

タグ圧縮構造内の複数のタグ圧縮エントリの第1のタグ圧縮エントリ内に、第1のメモリタグおよび第2のメモリタグによって共有される所定の長さの高位ビットのセットを記憶

50

するステップであって、前記複数のタグ圧縮エントリの各々が、それぞれのインデックス値によって識別される、記憶するステップと、

タグアレイ内の複数のタグアレイエントリの第1のタグアレイエントリ内に、前記第1のメモリタグの所定の長さの低位ビットのセットおよび前記第1のタグアレイエントリに関するインデックス値を記憶するステップであって、前記第1のタグアレイエントリの前記インデックス値が、前記第1のタグ圧縮エントリの前記インデックス値に対応し、前記第1のメモリタグが、データアレイ内に記憶されたデータ要素のメモリアドレスの少なくとも一部分を備え、前記複数のタグアレイエントリの第2のタグアレイエントリが、前記第1のタグ圧縮エントリに対応するインデックス値を特定し、前記第2のメモリタグの低位ビットのセットを記憶し、前記第2のメモリタグの低位ビットの前記セットが、前記第1のメモリタグの低位ビットの前記セットと一致しない、記憶するステップとを含む、方法。

【請求項10】

前記第1のタグ圧縮エントリ内の前記メモリタグの高位ビットの前記セットが入力メモリアドレスの高位ビットのセットと一致するとの決定時に、前記タグ圧縮構造から前記第1のタグ圧縮エントリに対応する前記インデックス値を返すことと、

前記入力メモリアドレスの低位ビットの前記セットが前記第1のタグアレイエントリ内の前記メモリタグの低位ビットの前記セットと一致するとの決定時に、前記第1のタグアレイエントリ内に記憶された前記インデックス値および前記メモリタグの低位ビットの前記セットを返すことと、

前記第1のタグ圧縮エントリの前記返されたインデックス値が前記第1のタグアレイエントリの前記返されたインデックス値と一致すると決定することと

によって、前記入力メモリアドレスに関する前記タグアレイ内のヒットの指示を返すステップをさらに含む、請求項9に記載の方法。

【請求項11】

前記タグ圧縮構造内の前記複数のタグ圧縮エントリの各々が、それぞれの複数のメモリタグによって共有される所定の長さの高位ビットの一意のセットを記憶するように構成され、前記複数のタグアレイエントリの各々が、それぞれのメモリタグの所定の長さの低位ビットのセットおよびインデックス値を記憶するように構成され、各タグアレイエントリの前記インデックス値が、それぞれの前記タグアレイエントリの前記メモリタグの前記高位ビットを含む前記タグ圧縮エントリの前記インデックス値に対応し、前記方法が、

入力メモリアドレスの高位ビットのセットが、前記複数のタグ圧縮エントリ内の高位ビットの前記セットのうちのいずれにも一致しないこと、

前記入力メモリアドレスの低位ビットのセットが、前記複数のタグアレイエントリ内に記憶された低位ビットの前記セットのうちのいずれにも一致しないこと、

前記入力メモリアドレスの前記低位ビットと一致する低位ビットのセットを記憶するタグアレイエントリから返されたインデックス値が前記入力メモリアドレスの前記高位ビットと一致する高位ビットのセットを記憶するタグ圧縮エントリのインデックス値と一致しないこと

のうちの少なくとも1つの決定時に、前記入力メモリアドレスに関する前記タグアレイ内のミスの指示を返すステップをさらに含む、請求項9に記載の方法。

【請求項12】

前記タグアレイ内の前記ミスの前記指示を返すことに応じて、

メモリ階層の次のレベルにおいて記憶されたデータを取り出すステップと、

前記データを前記データアレイの第1のエントリ内に記憶するステップと、

前記入力メモリアドレスの高位ビットのセットを前記タグ圧縮構造内の第2のタグ圧縮エントリ内に記憶するステップと、

(i)前記入力メモリアドレスの低位ビットのセット、および(ii)前記第2のタグ圧縮エントリに対応するインデックス値を前記タグアレイの第2のタグアレイエントリ内に記憶するステップであって、前記第2のタグアレイエントリが前記データアレイの前記第1のエ

ントリに対応する、記憶するステップと
をさらに含む、請求項11に記載の方法。

【請求項13】

前記タグ圧縮構造内の前記エントリが、(i)プロセスのアドレス空間識別子(ASID)、(ii)セキュリティビットのセット、および(iii)例外レベルビットのセットのうちの1つまたは複数を記憶するようにさらに構成される、請求項9に記載の方法。

【請求項14】

前記タグ圧縮構造が、フルアソシアティブコンテンツアドレスサブルメモリを備え、前記タグアレイが、(i)データキャッシュ、(ii)トランスレーションルックアサイドバッファ、(iii)ラインフィルバッファ、および(iv)命令キャッシュのうちの少なくとも1つの一部を備える、請求項9に記載の方法。

【請求項15】

プロセッサによって実行されると、前記プロセッサに動作を実行させる命令を記憶した、コンピュータ可読記憶媒体であって、前記動作が、

タグ圧縮構造内の複数のタグ圧縮エントリの第1のタグ圧縮エントリ内に、第1のメモリタグおよび第2のメモリタグによって共有される所定の長さの高位ビットのセットを記憶することであって、前記複数のタグ圧縮エントリの各々が、それぞれのインデックス値によって識別される、記憶することと、

タグアレイ内の複数のタグアレイエントリの第1のタグアレイエントリ内に、前記第1のメモリタグの所定の長さの低位ビットのセットおよび前記第1のタグアレイエントリに関するインデックス値を記憶することであって、前記第1のタグアレイエントリの前記インデックス値が、前記第1のタグ圧縮エントリの前記インデックス値に対応し、前記第1のメモリタグが、データアレイ内に記憶されたデータ要素のメモリアドレスの少なくとも一部分を備え、前記複数のタグアレイエントリの第2のタグアレイエントリが、前記第1のタグ圧縮エントリに対応するインデックス値を特定し、前記第2のメモリタグの低位ビットのセットを記憶し、前記第2のメモリタグの低位ビットの前記セットが、前記第1のメモリタグの低位ビットの前記セットと一致しない、記憶することと

を含む、コンピュータ可読記憶媒体。

【発明の詳細な説明】

【技術分野】

【0001】

米国特許法第119条に基づく優先権の主張

本出願は、本出願の譲受人に譲渡され、参照により本明細書に明確に組み込まれている、2015年5月20日に出願した米国特許出願第14/716,947号の利益を主張するものである。

【0002】

本明細書で開示する態様は、(本明細書でプロセッサとも呼ばれる)コンピュータマイクロプロセッサの分野に関する。より詳細には、本明細書で開示する態様はキャッシュタグ圧縮に関する。

【背景技術】

【0003】

現代のプロセッサは、従来、処理性能を改善するためにキャッシュに依存する。キャッシュは、作業負荷の命令ストリームおよびデータストリーム内の時間的および空間的なローカリティを活用することによって動作する。キャッシュの一部分は、キャッシュタグアレイの記憶専用である。キャッシュタグは、メインメモリからフェッチされた実際のデータのアドレスを記憶する。キャッシュ内のヒットまたはミスを決める速度は、タグアレイの幅によって制約され、より大きいタグアレイは、従来、より多くのタイミング圧力を生み出す。より幅の広い(または、より大きい)タグはより多くのビットを有する。キャッシュ内にヒットがあるかどうかを決めるためには、タグのすべてのビットをブローアドレスと比較しなければならない。したがって、比較においてビットが多くなると、その比較の結果を返すためにより多くの遅延が必要となり得る。

【発明の概要】

【発明が解決しようとする課題】

【0004】

本明細書で開示する態様は、タグ圧縮キャッシュ内の高位タグビットの一意の値を記憶することによって、タグストレージのために必要とされるエリアを削減する。

【課題を解決するための手段】

【0005】

一態様では、キャッシュメモリは、タグ圧縮構造とタグアレイとを備える。タグ圧縮構造は、複数のタグ圧縮エントリを記憶する。複数のタグ圧縮エントリの各々は、メモリタグの高位ビットのそれぞれのセットを含み、それぞれのインデックス値によって識別される。タグアレイは、複数のタグアレイエントリを記憶する。複数のタグアレイエントリの各々は、メモリタグの低位ビットのそれぞれのセットと、タグ圧縮構造のインデックス値のうちの1つに対応するインデックス値とを含む。第1のメモリタグは、データアレイ内に記憶されたデータ要素のメモリアドレスの少なくとも一部分を備え得る。

10

【0006】

一態様では、方法は、タグ圧縮構造内の複数のタグ圧縮エントリの第1のタグ圧縮エントリ内に、第1のメモリタグの高位のビットのセットを記憶するステップを含む。複数のタグ圧縮エントリの各々は、それぞれのインデックス値によって識別される。この方法は、タグアレイ内の複数のタグアレイエントリの第1のタグアレイエントリ内に、第1のメモリタグの低位ビットのセットおよび第1のタグアレイエントリに関するインデックス値を記憶するステップをさらに含む。第1のタグアレイエントリのインデックス値は、タグ圧縮構造内の第1のタグ圧縮エントリのインデックス値に対応する。第1のメモリタグは、データアレイ内に記憶されたデータ要素のメモリアドレスの少なくとも一部分を備え得る。

20

【0007】

一態様では、非一時的コンピュータ可読媒体は、プロセッサによって実行されると、タグ圧縮構造内の複数のタグ圧縮エントリの第1のタグ圧縮エントリ内に、第1のメモリタグの高位ビットのセットを記憶することを含む動作をプロセッサに実行させる命令を記憶する。複数のタグ圧縮エントリの各々は、それぞれのインデックス値によって識別される。この動作は、タグアレイ内の複数のタグアレイエントリの第1のタグアレイエントリ内に、第1のメモリタグの低位ビットのセットおよび第1のタグアレイエントリに関するインデックス値を記憶することをさらに含む。第1のタグアレイエントリのインデックス値は、タグ圧縮構造内の第1のタグ圧縮エントリのインデックス値に対応する。第1のメモリタグは、データアレイ内に記憶されたデータ要素のメモリアドレスの少なくとも一部分を備え得る。

30

【0008】

一態様では、装置は、複数のタグ圧縮エントリの第1のタグ圧縮エントリ内に、第1のメモリタグの高位ビットのセットを記憶するための手段を備える。複数のタグ圧縮エントリの各々は、それぞれのインデックス値によって識別される。この装置は、複数のタグアレイエントリの第1のタグアレイエントリ内に、第1のメモリタグの低位ビットのセットおよび第1のタグアレイエントリに関するインデックス値を記憶するための手段をさらに備える。第1のタグアレイエントリのインデックス値は、第1のタグ圧縮エントリのインデックス値に対応する。第1のメモリタグは、データ要素のメモリアドレスの少なくとも一部分を備え得る。

40

【0009】

メモリタグの一部分をタグ圧縮キャッシュ内に記憶することによって、本明細書で開示する態様は、タグアレイ内の冗長ビットの記憶を低減させる。そうすることは、メモリタグのためにより少ない物理エリアを必要とし、より高速な比較結果を提供し、比較を実行するためにより少ない電力を必要とする。

【0010】

上記で具陳した態様が達成される方法が詳細に理解され得るように、添付の図面を参照

50

することによって、上記で手短に要約した、本開示の態様がより具体的に説明される。

【0011】

ただし、この開示は他の態様にも当てはまる場合があるので、添付の図面は、本開示の態様のみを示し、したがって、その範囲を限定するものと見なされるべきではないことに留意されたい。

【図面の簡単な説明】

【0012】

【図1】一態様による、キャッシュタグ圧縮を実現するプロセッサを示す図である。

【図2】一態様による、キャッシュタグ圧縮を実現するように構成されたプロセッサの構成要素の論理図である。

【図3】一態様による、タグ圧縮キャッシュ、タグアレイ、およびデータアレイ内に記憶された例示的な値を示す概略図である。

【図4】一態様による、キャッシュタグ圧縮を実現するための方法を示すフローチャートである。

【図5】一態様による、新しいタグ圧縮キャッシュエントリを確立するための方法を示すフローチャートである。

【図6】一態様による、キャッシュタグ圧縮を実現するように構成されたプロセッサを組み込んだコンピューティングデバイスを示すブロック図である。

【発明を実施するための形態】

【0013】

本明細書で開示する態様は、メモリタグ記憶のためのエリアを削減するための技法を提供する。たとえば、キャッシュは、上位アドレスビットのいくつかのセットでタグ付けされ得る。概して、キャッシュタグは、より多くのビットのためのストレージを必要とし、現代のキャッシュアーキテクチャは、従来、前の世代よりも実質的により大きいタグを必要とする。たとえば、64ビットラインを有する仮説的4ウェイ32キロバイト(KB)の命令キャッシュを考えられたい。そのようなキャッシュのライン数は、 $2^{15}/2^9$ 、または512ラインである。各ラインはタグに関連付けられる。キャッシュのセットの数は、 $2^{15}/2^6/2^2$ 、または128セット(7ビットセット復号を伴う)に等しい。少なくとも一態様では、タグビットは、7ビットセット復号または6ビットバイトオフセットを含まず、すなわち、タグビットはA[7+6]またはA[13]で開始し、この場合、「A」はタグ探索に対するアドレス入力を示す。したがって、64ビットの例示的なアドレスの場合、タグはA[63:13]と比較され、タグストレージはライン当たり51ビットである。キャッシュのための総タグストレージは、したがって、51ビット/ライン*512ライン、または26,112ビットであり、キャッシュタグを記憶するために10%のストレージオーバーヘッドをもたらす。

【0014】

タグ値は、任意であり、コード依存であるが、タグ値の各々のビットのサブセットは、ビットのサブセットが複数のキャッシュタグにわたって同じ値を有するように、若干の変動を有し得る。本明細書で開示する態様は、複数のキャッシュタグにわたって同じ値の複数のコピーの冗長なストレージを除去することによって、この傾向を活用する。具体的には、本明細書で開示する態様は、一意のタグ値を記憶するように構成されたタグ圧縮キャッシュを導入する。そうすることはキャッシュタグからいくつかのビットを除去する。キャッシュタグから除去されるビットは、タグ圧縮キャッシュエントリを指す、より少数のビットで置換され得る。タグ圧縮キャッシュは、限定はしないが、上位(または、高位)アドレスビット、アドレス空間識別子(ASID: address space identifier)、セキュリティビット、例外レベルビット(exception level bits)などを含む、ビットの任意のセットをキャッシュタグから保留することができる。

【0015】

タグ圧縮キャッシュは元のキャッシュタグの一部分を保留するため、キャッシュ内で「ヒットする」ために、アクセスはタグ圧縮キャッシュ内で「ヒット」しなければならない。同様に、タグ圧縮キャッシュ内で「ミス」するアクセスは、キャッシュ自体の中でも「

10

20

30

40

50

ミス」しなければならない。タグ圧縮キャッシュエントリをエビクトすることは、すべての対応するキャッシュライン(すなわち、エビクトされているタグ圧縮キャッシュのインデックスと一致するタグ圧縮キャッシュインデックス値を有するライン)の無効化を必要とする。しかしながら、キャッシュラインの無効化は、タグ圧縮キャッシュに対して何の更新も必要としない。

【0016】

本明細書で開示する態様は、タグ圧縮キャッシュおよびキャッシュに並行してアクセスし、入力メモリアドレスがキャッシュの「ヒット」を生じさせるか、または「ミス」を生じさせるかを決定する。具体的には、本明細書で開示する態様は、入力メモリアドレスの高位ビットのセットを使用して、タグ圧縮キャッシュにアクセスする。タグ圧縮キャッシュ内にヒットがある(すなわち、タグ圧縮キャッシュ内のエントリが入力メモリアドレスの高位ビットと一致する)場合、タグ圧縮キャッシュは、一致するエントリのインデックス値を返すように構成される。並行して、本明細書で開示する態様は、入力メモリアドレスの低位ビットのセットを使用して、キャッシュを探索する。キャッシュ内にヒットがある(すなわち、キャッシュタグアレイ内のタグエントリが入力メモリアドレスの低位ビットと一致する)場合、キャッシュは、一致するエントリ内の低位ビットおよびインデックス値を返すように構成される。本明細書で開示する態様は、次いで、キャッシュおよびタグ圧縮キャッシュから返されたインデックス値を比較することができる。インデックス値が一致しない場合、本明細書で開示する態様は、キャッシュミスの指示を返す。本明細書で開示する態様は、キャッシュタグエントリから返された下位ビットを入力メモリアドレスの下位ビットとさらに比較することができる。下位ビットが一致しない場合、本明細書で開示する態様は、キャッシュミスの指示を返す。しかしながら、インデックス値が一致し、下位ビットが一致する場合、本明細書で開示する態様は、キャッシュヒット(すなわち、要求されたデータがキャッシュ内に存在すること)の指示を返す。

【0017】

本明細書で、キャッシュは、本開示の態様の説明を円滑にするための参照例として使用される。しかしながら、本明細書で説明する技法は、プロセッサキャッシュ、データキャッシュ、命令キャッシュ、ラインフィルバッファ、トランスレーションルックアサイドバッファ(TLB)など、タグフィールドを有する他のハードウェア構造に適用されるため、キャッシュの使用は、本開示を限定するものと見なすべきではない。特定のハードウェア構造に対する任意の参照は、本開示の限定と見なすべきではない。

【0018】

図1は、一態様による、キャッシュタグ圧縮を実現するプロセッサ101を示す。示すように、プロセッサ101は、命令を実行する命令実行パイプライン112を含む。パイプライン112は、各々が様々な非アーキテクト(non-architected)レジスタ(図示せず)と、1つまたは複数の論理演算ユニット(やはり図示せず)とを含む複数の並行パイプラインを備えたスーパースカラ設計であり得る。示すように、プロセッサ101は、1つまたは複数のより高いレベルのメモリ108からデータのラインを記憶するキャッシュ102(キャッシュメモリ102とも呼ばれる)も含む。より高いレベルのメモリ108は、限定はしないが、より高いレベルのキャッシュおよび/またはメイン(システム)メモリを含み得る。少なくとも一態様では、キャッシュ102はレベル1(L1)データキャッシュである。概して、CPU101は多数の変形形態を含むことができ、図1に示すCPU101は、例示のためであり、本開示を限定するものと見なすべきではない。たとえば、CPU101はグラフィックス処理ユニット(GPU)であり得る。

【0019】

示すように、キャッシュ102は、キャッシュ論理103と、タグ圧縮キャッシュ104と、タグアレイ105と、データアレイ106とを含む。キャッシュ論理103は、概して、キャッシュのヒットまたはミスが特定の動作において生じるかどうかの決定など、キャッシュ102の動作を制御する。タグ圧縮キャッシュ(TCC)104は、メモリタグの上位(または、高位)ビットのセットを保留するように構成されたハードウェア構造である。たとえば、TCC104は、メモリタグの上位アドレスビット、ASIDビット、セキュリティビット、例外レベルビット

などのうちの1つまたは複数を保留することができる。TCC104は、概して、場合によっては、タグアレイ105内に冗長的に記憶されることになる冗長値を記憶するように構成される。タグアレイ105は、概して、キャッシュ102内に記憶されたデータのアドレスを記憶するように構成される。より具体的には、タグアレイ105の各エントリは、TCC104内のメモリタグの低位ビットのセットおよびエントリのインデックス値に対応するインデックス値を記憶するように構成される。データアレイ106は、キャッシュラインのデータを記憶する。少なくとも一態様では、TCC104は、コンテンツアドレスサブルメモリ(CAM:content addressable memory)構造として実装される。同様に、タグアレイ105およびデータアレイ106は、CAMとして実装される。さらに、少なくとも一態様では、TCC104は、フルアソシアティブ(fully-associative)最長時間未使用(LRU:least recently used)置換構造である。TCC104は、2、4、または8など、任意の数のエントリを有し得る。TCC104、タグアレイ105、およびデータアレイ106内にデータを記憶するための手段は、1つまたは複数のメモリセルを含み得る。

【0020】

動作において、プロセッサ101は、より高いレベルのメモリ108のうちの1つの中に位置するデータがキャッシュ102内に存在するかどうかを決定することを探索することができる。プロセッサ101が(仮想アドレスであり得る)入力メモリアドレスをキャッシュ102に提供するとき、キャッシュ論理103は、入力メモリアドレスにおけるデータがキャッシュ102内にあるかどうかを決定することができる。そうするために、キャッシュ論理103は、入力メモリアドレスの高位ビットを使用して、TCC104のCAM探索を開始することができる。TCC104内のエントリが入力メモリアドレスの高位ビットと一致する場合、TCC104はTCC104内の一致するエントリのインデックス値を返すように構成される。キャッシュ論理103は、TCC104のCAM探索と平行して、タグアレイ105のCAM探索を開始することができる。キャッシュ論理103は、入力メモリアドレスの低位ビットを使用して、タグアレイ105を探索することができる。タグアレイ105内のエントリの低位ビットが入力メモリアドレスの低位ビットと一致する場合、タグアレイ105は、一致するエントリの低位ビットおよびインデックス値を返すように構成される。キャッシュ論理103は、次いで、TCC104およびタグアレイ105によって返されたインデックス値を比較することができる。インデックス値が一致しない場合、キャッシュ論理103はキャッシュミスの指示を返すことができる。キャッシュ論理103は、タグアレイ105によって返された低位ビットを入力メモリアドレスの低位ビットと比較することもできる。低位ビットが一致し、インデックス値が一致する場合、キャッシュ論理103はキャッシュヒットの指示を返すように構成される。

【0021】

TCC104の探索の結果がミスである場合、キャッシュ論理103は、TCC104内に入力メモリアドレスに関するエントリを作成するように構成され得る。空き(またはさもなければ、利用可能な)エントリがTCC104内に存在する場合、キャッシュ論理103は、入力メモリアドレスの高位ビットのセットを空きエントリ内に記憶することができる。TCC104内で何のエントリも利用可能でない場合、キャッシュ論理103は、TCC104内の既存のエントリをエビクトし、エビクトされたエントリのインデックス値を特定する、タグアレイ105(および、データアレイ106)内の任意の対応するエントリを無効化することができる。キャッシュ論理103は、LRUなど、TCC104に適した任意の置換ポリシーを適用し得、または、(TCC104内の各エントリに関してカウンタによって決定され得る)最も少ないキャッシュラインに関連付けられるTCC104エントリを置換する。キャッシュ論理103は、フラッシュ無効化またはライン単位の無効化によってなど、いずれかの実現可能な方法でエントリを無効化することができる。入力メモリアドレスの高位ビットを特定するエントリをTCC104に追加することに加えて、キャッシュ論理103は、入力メモリアドレスの低位ビットおよび入力メモリアドレスの高位ビットを記憶するTCC104エントリのインデックス値を特定するエントリをタグアレイ105内に作成することもできる。

【0022】

少なくとも一態様では、キャッシュ論理103はクロックサイクルごとにTCC104をブロー

10

20

30

40

50

ブすることができる。しかしながら、これは、大量の電力を引き出す可能性があり、クリティカルタイミングパス(critical timing path)内であり得る。TCC104内に構築されたエントリが多ければ多いほど、TCC104をプローブするために、より多くの電力および時間が必要とされ得る。したがって、いくつかの態様では、キャッシュ論理103は、サイクルの大部分においてTCC104を探索する必要を回避する論理を含み得る。たとえば、レジスタのセットは、TCC104に対する前のアクセスによって返されたビット(および、対応するTCC104エントリが有効な状態に留まるかどうか)を記憶するように構成され得る。概して、キャッシュ論理103は、TCC104内に保留される入力メモリアドレスビットに対応する入力メモリアドレスビットがTCC104に対する前のアクセス内と同じであるかどうかを決定することができる。キャッシュ論理103は、次いで、関連付けられたTCC104エントリが、最後のアクセス以来、無効化されていないかどうかを決定することができる。両方の決定が真である場合、キャッシュ論理103は、圧縮キャッシュタグがTCC104内にあるはずであり、前のサイクルのインデックスの再使用が安全であると決定し、現在のサイクル内でTCC104を探索することを不要にし得る。いずれかの決定が真でない場合、キャッシュ論理103は、前のサイクルのTCC104インデックスを再使用することができず、制御論理103はTCC104を探索しなければならない。TCC104、タグアレイ105、およびデータアレイ106を探索するための手段は、ハードウェアおよび/またはソフトウェアとして実装される論理を含む。同様に、ハードウェアおよび/またはソフトウェアとして実装される論理は、値を読み取るおよび/または書き込むための手段、ヒットおよび/またはミスの指示を返すための手段、ならびにTCC104、タグアレイ105、およびデータアレイ106から値を返すための手段として働き得る。そのような手段論理の例は、メモリコントローラ、キャッシュコントローラ、およびデータコントローラを含む。

【0023】

前に示したように、図1に示す圧縮方式は、他のPCタグ付けおよび/またはASIDタグ付けされたマイクロアーキテクチャ構造に適用され得る。たとえば、仮想アドレスは、ブロックを通して使用され得る形式{タグ圧縮インデックス値、部分仮想アドレス}に変換され得る。そうすることは、すべてのデータ経路要素が、トランスレーションルックアサイドバッファ(TLB)、分岐予測器、中断点、キャッシュなどを含めて、この短縮された形式で動作することを可能にすることになる。必要な場合、ブロック境界において、キャッシュコンテンツデバッグ機能に関してなど、必要なときはいつでも、変換されたアドレスを元のアドレスに変換することができる。同様に、タグ圧縮技法を命令アドレス、ロード/記憶アドレス、および物理的にタグ付けされた構造に独立して適用することができる。

【0024】

一態様では、プロセッサ101を、命令実行パイプライン112とキャッシュ102とを含む集積回路上に配設することができる。別の態様では、キャッシュ102を、プロセッサ101を含む集積回路とは別個の集積回路上に配置することができる。

【0025】

図2は、一態様による、キャッシュタグ圧縮を実現するように構成されたプロセッサ101のキャッシュ論理103の一部分の論理図である。示すように、プロセッサ101は、タグ圧縮キャッシュ(TCC)104とキャッシュタグアレイ105とを含む。前に説明したように、TCC104は、複数のTCCエントリ202を記憶する、インデックス付けされたハードウェア構造である。TCCエントリ202は、メモリタグの高位アドレスビットのセットを含み得る。少なくともいくつかの態様では、TCCエントリ202は、ASID、セキュリティレベル、例外レベルなどに関するビットをさらに含み得る。示すように、キャッシュタグアレイ105は、複数のエントリを含み、エントリの各々は、タグ圧縮キャッシュ(TCC)インデックス値204とキャッシュタグ205とを有する。TCCインデックス値204は、TCC104内のエントリ202のインデックス値に対応し得る。キャッシュタグ205は、メモリタグの低位ビットを含み得る。示すように、キャッシュタグ205は、メモリタグの低位ビット、すなわち、Nビットメモリタグのビット0からMを記憶する。同様に、TCC104は、Nビットのメモリタグの高位ビット、すなわち、メモリタグのビットM+1からNを記憶する。したがって、完全なメモリタグ(ビット0か

らN)は、TCC104内に記憶された高位ビット(ビットM+1からN)をキャッシュタグ205の低位ビット(ビット0からM)と組み合わせることによって、作成され得る。

【0026】

示すように、プロセッサ101は、メモリアドレス201において記憶されたデータを要求し得る。キャッシュ論理103は、メモリアドレス201において記憶されたデータがキャッシュ102内に位置するかどうかを決定するように構成され得る。少なくとも一態様では、メモリアドレス201は、Nビットの長さを有する仮想メモリアドレスであり得る。メモリアドレス201において記憶されたデータがキャッシュ102内にあるかどうかを決定するために、キャッシュ論理103は、TCC104とキャッシュタグアレイ105とを並行してプローブすることができる。より具体的には、示すように、キャッシュ論理103は、キャッシュタグアレイ105のCAM探索を実行することができる。キャッシュタグ205のうちの1つがメモリアドレス201のビット0からMと一致するエントリを含む場合、キャッシュタグアレイ105は、TCCインデックス値208およびキャッシュタグ209を返すことができる。さらに、示すように、スプリッタ213はメモリアドレス201を分割して、メモリアドレス201のビットM+1からNをTCC104に提供することができる。TCC104上でのCAM探索は、メモリアドレス201のビットM+1からNを使用して実行され得る。TCC104内のエントリ202がメモリアドレス201のビットM+1からNの値と一致する値を含む場合、TCC104は、TCCエントリ206およびTCCインデックス値207を返すように構成され、この場合、インデックス値207は一致するTCCエントリ206を識別する。TCCインデックス値207は、TCC104内の各エントリ202がそれぞれのTCCインデックス値207によって一意に識別されることを可能にするのに適した長さの任意の数のビットであり得る。したがって、たとえば、TCC104が4個のハードウェアエントリ(たとえば、4個のTCCエントリ202)を有する場合、各エントリに対応するTCCインデックス値207は2ビットの長さであり得る。そのような例では、キャッシュタグアレイ105内に記憶された各TCCインデックス204も2ビットの長さになる。少なくとも一態様では、TCCインデックス値207は、所与のTCCエントリ202のロケーション(または、アドレス)に基づいて暗示的に識別される。

【0027】

示すように、比較器210は、TCC104から返されたTCCインデックス値207をキャッシュタグアレイ105によって返されたTCCインデックス値208と比較する。さらに、比較器211は、メモリアドレス201の低位ビット0:Mをキャッシュタグアレイ105から返されたキャッシュタグエントリ209内に記憶された低位ビット0:Mと比較する。示すように、AND論理212は、比較器210、211の出力に関して論理AND演算を実行するように構成される。両方の比較器210、211が一致を示す場合、AND論理212の出力はキャッシュヒットの指示を返すように構成される。別の言い方をすれば、比較器210によって比較されたインデックス値が一致し、比較器211によって比較された低位ビット0:Mが一致する場合、要求されるメモリアドレス201のコンテンツはキャッシュ102内に記憶され、キャッシュ論理103はキャッシュヒットの指示を返すことができる。しかしながら、比較器210が比較されたインデックス値が一致しないことを示すか、または比較器211が低位ビット0:Mが一致しないことを示す場合、要求されるデータはキャッシュ102内になく、キャッシュ論理103はキャッシュミスの指示を返すことができる。したがって、示すように、キャッシュ論理103は、ビット0:Mおよびインデックス値の比較をもたらす。そうすることは、キャッシュ論理103が(ビット0:Nの)旧来のタグ比較よりも少ないビットを比較することを可能にし、旧来の比較よりも高速の比較をもたらす。

【0028】

キャッシュミスの場合、キャッシュ論理103は、上記で説明したように、要求されるデータを高位メモリからフェッチし、TCC104およびキャッシュタグアレイ105をポピュレートするように構成され得る。同様に、TCC104およびキャッシュタグアレイ105の最初のCAM探索のいずれかがミスである場合、キャッシュ論理103は、上記で説明したように、要求されるデータを高位メモリからフェッチし、TCC104およびキャッシュタグアレイ105をポピュレートするように構成され得る。

【 0 0 2 9 】

図3は、一態様による、タグ圧縮キャッシュ(TCC)104、キャッシュタグアレイ105、およびデータアレイ106内に記憶された例示的な値を示す概略図300である。示すように、TCC104内の各TCCエントリ202は、メモリタグの一部分を特定し、それぞれのTCCインデックス値207によって識別される。それぞれのインデックス値207は、各TCCエントリ202を一意に識別する。少なくとも一態様では、インデックス値207は、TCC104内の各エントリのロケーション(または、アドレス)によって暗示的に提供される。したがって、そのような態様では、インデックス値207は、TCC104内で明確に定義されず、インデックス値207は、そのような態様の説明を容易にするために、図3に示される。各TCCエントリ202のタグ部分は、メモリタグの高位アドレスビットであり得、ASIDビット、セキュリティビット、例外レベルビットなど、追加の情報を含み得る。

10

【 0 0 3 0 】

示すように、表301は、キャッシュタグアレイ105およびデータアレイ106の例示的な値を含む。データアレイ106は、キャッシュ102内の各エントリに関するデータライン302を記憶する。キャッシュタグアレイ105は、前に説明したように、インデックス値204とキャッシュタグ205とを含む。キャッシュタグアレイ105内のインデックス値204は、TCC104内の所与のTCCエントリ202に対応するインデックス値207に対応し、各対応するキャッシュタグ205はメモリタグの低位ビットを記憶する。したがって、示すように、表301内の複数のエントリは同じインデックス値204を記憶することができる。有利には、TCC104は、メモリタグの高位ビットのフルビットパターンの単一のコピーを記憶して、キャッシュタグアレイ105内に記憶された冗長情報の量(すなわち、異なるインデックス値204であって、メモリタグのより長い高位ビットではない)を低減させる。したがって、メモリタグは、高位ビットがTCC104内に記憶され、低位ビットがキャッシュタグアレイ105内に記憶された状態で、TCC104およびキャッシュタグアレイ105にわたって記憶される。概念的に言えば、TCC104内のTCCエントリ202のタグ部分は、完全なメモリタグを作成するために、キャッシュタグアレイ105のインデックス値204を置換し得る。たとえば、表301の第1の行において、インデックス204のビット「00」は、TCC104内の「00」のインデックス値207に対応するTCCエントリ202のタグビット「01...0100」によって置換されることになる。

20

【 0 0 3 1 】

例示的な一態様では、図3に示す構成は、仮定的命令キャッシュをTable 1(表1)内に示す例示的な構成で置換することができる。

30

【 0 0 3 2 】

【表 1】

Table I:

ラインサイズ	64 バイト
連合性	4 ウェイ
セットの数	128
ライン総数	512
呼び容量	32KB
アドレス幅	49 ビット
セットインデックス	仮想アドレス(7 ビット)
(ライン当たり)タグビットの数	{仮想アドレス,ASID}(36+16=52 ビット)
タグビットの数(合計)	(ライン当たり 52 ビット*512 ライン)=26,624 ビット

10

20

【 0 0 3 3 】

したがって、Table 1(表1)に示すように、仮定的命令キャッシュは、32KB(または、32,768バイト、もしくは262,144ビット)の総キャッシュストレージのうち、26,624ビットのストレージオーバーヘッドをタグストレージに対して必要とする。この例示的な構成において必要とされる命令キャッシュタグは、{PC[48:13], ASID}になる。

30

【 0 0 3 4 】

しかしながら、これらのビットのサブセット({PC[48:28], ASID}など)は、現実世界の作業負荷において若干の変動を示す傾向がある。この例示的な命令キャッシュの場合、4個のエントリTCC104を作成することによって、各TCCエントリ202は、21 PC[48:28]ビットおよびASIDの16ビットの総計37ビットを記憶することになる。TCC104に対して必要とされる総記憶量は、したがって、4エントリ×37ビット、すなわち、148ビットになる。TCCエントリ202は、TCC104内の各それぞれのTCCエントリ202のロケーション(または、アドレス)によって暗示的にインデックス付けされ得る(たとえば、4エントリTCC104において、ロケーション0、1、2、および3は、それぞれ、00、01、10、および11のバイナリインデックス値に対応する)。一方、キャッシュタグアレイ105のインデックス204は、2ビットインデックス値を記憶することになり、キャッシュタグ205は、タグビットPC[27:13]、または15ビットの総計17ビットを記憶することになる。命令キャッシュの512本のラインにわたって、キャッシュタグアレイ105内の総記憶域は、512個のエントリ×17ビットの総計8,704ビットになる。TCC104およびキャッシュタグアレイ105にわたって必要とされる総記憶域は、8,852ビット(148+8,704)になり、これは、Table 1(表1)で説明する命令キャッシュによって必要とされる26,624ビットよりもかなり少ない。

40

【 0 0 3 5 】

図4は、一態様による、キャッシュタグ圧縮を実現するための方法400を示すフローチャートである。方法400はステップ410で開始し、ここで、キャッシュ論理103は長さNビットのメモリアドレスを受信することができる。メモリアドレスは、メモリアドレスにおい

50

て記憶されたデータがキャッシュ102内に位置するかどうかを決定するための要求の一部として受信され得る。キャッシュ論理103は、次いで、TCC104とキャッシュタグアレイ105とを並行してプローブすることができる。より具体的には、ステップ415において、キャッシュ論理103は、ステップ410で受信されたメモリアドレスの下位Mビットを使用して、キャッシュタグアレイ105上でCAM探索を実行することができる。キャッシュタグアレイ105内のエントリがメモリアドレスの下位Mビットと一致する値を記憶している場合、キャッシュタグアレイ105内にヒットが存在し、方法はステップ460に進む。加えて、キャッシュタグアレイ105は、メモリアドレスの下位Mビットおよび一致するエントリ内に記憶されたインデックス値を返すことができる。メモリアドレスの下位Mビットと一致する値を記憶するキャッシュタグアレイ105内に何のエントリも存在しない場合、キャッシュタグアレイ105内にミスが存在し、方法はステップ480に進む。

10

【0036】

ステップ420で、キャッシュ論理103は、要求されるメモリアドレスの高位ビット(M+1からN)を使用して、TCC104をプローブすることができる。TCC104内のエントリが要求されるメモリアドレスの高位ビットと一致する値を記憶する場合、方法はステップ460に進む。TCC104内のヒットの場合、TCC104は、メモリアドレスの高位ビットと一致する、TCC104内のエントリのインデックス値を返すことができる。TCC104内にミスが存在する場合、方法はステップ430に進み、ここで、キャッシュ論理103は、TCC104内に利用可能なエントリが存在するかどうかを決定することができる。エントリが利用可能でない場合、方法はステップ440に進み、ここで、キャッシュ論理103は、TCC104内のエントリをエビクトし、キャッシュタグアレイ105内の任意の対応するエントリを無効化することができる。キャッシュ論理103は、最長時間未使用のエントリをエビクトすること、またはキャッシュタグアレイ105内の最小エントリに関連付けられた、TCC104内のエントリをエビクトすることなど、何らかの適切なエビクションポリシーを使用することができる。キャッシュ論理103は、フラッシュ無効化またはライン単位の無効化など、何らかの実現可能な方法でキャッシュタグアレイ105内のエントリを無効化することができる。図5を参照してより詳細に説明するステップ450において、キャッシュ論理103はTCC104内に新しいエントリを確立することができる。概して、キャッシュ論理103は、メモリアドレスをビット0:Mの下位セットとビットM+1:Nの高位セットに分割することができる。キャッシュ論理103は、次いで、ビットの下位セットをキャッシュタグアレイ105内に記憶し、ビットの高位セットをTCC104内に記憶することができる。

20

30

【0037】

ステップ460で、キャッシュ論理103は、TCC104およびキャッシュタグアレイ105によって返されたインデックス値を比較することができる。キャッシュ論理103は、キャッシュタグアレイ105によって返された低位ビットを要求されるメモリアドレスの低位ビットとさらに比較することができる。両方の比較が一致をもたらす場合、この方法はステップ470に進み、ここで、キャッシュ論理103はキャッシュヒットの指示を返すことができる。ステップ460において実行された比較のうちの1つ(または、両方)が一致をもたらさない場合、この方法はステップ480に進み、ここで、キャッシュ論理103はキャッシュミスの指示を返す。

40

【0038】

図5は、一態様による、タグ圧縮キャッシュ104内に新しいエントリを確立するためのステップ450に対応する方法500を示すフローチャートである。示すように、方法500はステップ510で開始し、ここで、プロセッサ101は、ステップ410で提供されたメモリアドレスに関連付けられるデータをより高いレベルのメモリ(より高いレベルのキャッシュまたはメインメモリなど)から取り出すことができる。ステップ520で、キャッシュ論理103は、メモリアドレスの高位ビット(ビットM+1:Nなど)をTCC104の第1のエントリ内に記憶することができる。ステップ530で、キャッシュ論理103は、TCC104の第1のエントリのインデックス値およびメモリアドレスの低位ビット(ビット0:Mなど)をタグアレイ105の第1のエントリ内に記憶することができる。ステップ540で、キャッシュ論理103は、ステップ510で

50

取り出されたデータをタグアレイ105の第1のエントリに対応するデータアレイ106内に記憶する。

【0039】

図6は、一態様による、キャッシュタグ圧縮を実現するように構成されたプロセッサ101を組み込んだコンピューティングデバイス601を示すブロック図である。図1～図5に示した装置および方法のすべては、コンピューティングデバイス601内に含まれ得るか、またはコンピューティングデバイス601によって実行され得る。コンピューティングデバイス601は、ネットワーク630を介して他のコンピューティングデバイスに接続されてもよい。概して、ネットワーク630は、電気通信ネットワークおよび/またはワイドエリアネットワーク(WAN)であり得る。特定の態様では、ネットワーク630はインターネットである。概して、コンピューティングデバイス601は、限定はしないが、デスクトップコンピュータ、ラップトップコンピュータ、タブレットコンピュータ、およびスマートフォンを含む、キャッシュタグ圧縮を実装するように構成されたプロセッサを含む任意のデバイスであってよい。

10

【0040】

コンピューティングデバイス601は、概して、バス620を介して、メモリ608、ネットワークインターフェースデバイス618、ストレージ609、入力デバイス622、および出力デバイス624に接続されたプロセッサ101を含む。コンピューティングデバイス601は、概して、オペレーティングシステム(図示せず)の制御下にある。本明細書で説明する機能をサポートする任意のオペレーティングシステムが使用され得る。プロセッサ101は、単一のCPU、複数のCPU、複数の処理コアを有する単一のCPUなどを表すように含まれる。ネットワークインターフェースデバイス618は、コンピューティングデバイス601がネットワーク630を介して他のコンピューティングデバイスと通信することを可能にする任意のタイプのネットワーク通信デバイスであり得る。

20

【0041】

ストレージ609は、永続ストレージデバイスであってよい。ストレージ609は単一のユニットとして示されるが、ストレージ609は、固定ディスクドライブ、固体ドライブ、SANストレージ、NASストレージ、取外し可能メモ리카ード、または光ストレージなど、固定および/または取外し可能ストレージデバイスの組合せであり得る。メモリ608およびストレージ609は、複数の1次ストレージデバイスおよび2次ストレージデバイスにまたがる1つの仮想アドレス空間の一部であり得る。

30

【0042】

入力デバイス622は、コンピューティングデバイス601に入力を提供するための任意のデバイスであり得る。たとえば、キーボードおよび/またはマウスを使用することができる。出力デバイス624は、コンピューティングデバイス601のユーザに出力を提供するための任意のデバイスであり得る。たとえば、出力デバイス624は、任意の従来のディスプレイスクリーンまたはスピーカのセットであってよい。入力デバイス622とは別に示されているが、出力デバイス624および入力デバイス622は結合されてよい。たとえば、一体化タッチスクリーンを備えたディスプレイスクリーンが使用され得る。

40

【0043】

有利には、本明細書で開示する態様は、メモリタグ値を記憶するために必要とされる空間の量を低減させる。メモリタグの高位ビットの冗長値をより小さなタグ圧縮キャッシュ内に記憶することによって、より小さなタグアレイ構造を使用して、メモリタグの低位ビットを記憶することができる。そうすることは、より小さな全体的なシリコンダイを使用してそのような構造が作成されることを可能にし得、これは、他のハードウェア構造がより近くに一緒に配置されることを可能にすることによって、製造コストを削減し、漏れ電流を低減し、回路タイミングを改善する。加えて、または代替として、節約された空間を他のマイクロアーキテクチャ構造上で使うことができる。

【0044】

いくつかの態様が説明されてきた。しかしながら、これらの構成に対する様々な変更が

50

可能であり、本明細書で提示した原理は他の構成にも同様に適用され得る。そのような方法の様々なタスクは、マイクロプロセッサ、埋込み型コントローラ、またはIPコアなど、論理要素の1つまたは複数のアレイによって実行可能な命令のセットとして実装され得る。

【 0 0 4 5 】

上記で説明した方法の様々な動作は、プロセッサ、ファームウェア、特定用途向け集積回路(ASIC)、ゲート論理/レジスタ、メモリコントローラ、またはキャッシュコントローラなど、動作を実行することが可能な任意の適切な手段によって実行され得る。概して、図に示された任意の動作は、その動作を実行することが可能な対応する機能的手段によって実行され得る。

【 0 0 4 6 】

上記で開示したデバイスおよび機能は、コンピュータ可読媒体上に記憶されたコンピュータファイル(たとえば、RTL、GDSII、GERBERなど)に設計および構成されてもよい。いくつかまたはすべてのそのようなファイルは、そのようなファイルに基づいてデバイスを製造する製造者に提供されてもよい。結果として得られる製品は、半導体ウエハを含み、次いで、半導体ウエハは半導体ダイに切断され、半導体チップにパッケージングされる。いくつかのまたはすべてのそのようなファイルは、本明細書で説明したデバイスを作製するために設計データを使用して作製機器を構成する製造者に提供されてよい。コンピュータファイルから形成された結果得られた製品は半導体ウエハを含み、半導体ウエハは、次いで、半導体ダイ(たとえば、プロセッサ101)に切断され、パッケージングされ、限定はしないが、モバイルフォン、スマートフォン、ラップトップ、ネットブック、タブレット、ウルトラブック、デスクトップコンピュータ、デジタルビデオレコーダ、セットトップボックス、および集積回路が使用される任意の他のデバイスを含む製品にさらに組み込まれる。

【 0 0 4 7 】

一態様では、コンピュータファイルは、上記で説明し、物理的な設計レイアウト、概略図、ハードウェア記述言語(たとえば、Verilog、VHDLなど)の形態で、図で示した回路を含む設計構造を形成する。たとえば、設計構造は、テキストファイルまたは上記で説明し、図に示したような回路のグラフィカル表現であってよい。設計プロセスは、好ましくは、下記で説明する回路をネットリストに合成し(たとえば、変換し)、この場合、ネットリストは、たとえば、集積回路設計内の他の要素および回路に対する接続を記述し、機械可読媒体のうちの少なくとも1つの上に記録される、ワイヤ、トランジスタ、論理ゲート、制御回路、I/O、モデルなどのリストである。たとえば、媒体は、CD、コンパクトフラッシュ(登録商標)、他のフラッシュメモリ、またはハードディスクドライブなどの記憶媒体であり得る。別の態様では、本明細書で説明したハードウェア、回路、および方法は、プロセッサによって実行されると、上記で説明し、図に示した回路の機能をシミュレートするコンピュータファイルに構成され得る。これらのコンピュータファイルは、回路シミュレーションツール、回路図エディタ、または他のソフトウェアアプリケーション内で使用され得る。

【 0 0 4 8 】

本明細書で開示した態様の実装形態は、(たとえば、本明細書に記載する1つまたは複数のコンピュータ可読記憶媒体の有形なコンピュータ可読特徴中で)論理要素のアレイ(たとえば、プロセッサ、マイクロプロセッサ、マイクロコントローラ、または他の有限状態機械)を含む機械によって実行可能な命令の1つまたは複数のセットとしても有形に実施され得る。「コンピュータ可読媒体」という用語は、情報を記憶または転送することができる、揮発性、不揮発性、取外し可能および取外し不可能な記憶媒体を含む、任意の媒体を含み得る。コンピュータ可読媒体の例は、電子回路、半導体メモリデバイス、ROM、フラッシュメモリ、消去可能ROM(EROM)、フロッピーディスクもしくは他の磁気ストレージ、CD-ROM/DVDもしくは他の光ストレージ、ハードディスクもしくはは所望の情報を記憶するのに使われ得る任意の他の媒体、光ファイバー媒体、無線周波(RF)リンク、または所望の

情報を運ぶために使用され得るとともにアクセスされ得る任意の他の媒体を含む。コンピュータデータ信号は、電子ネットワークチャネル、光ファイバー、無線リンク、電磁リンク、RFリンクなどの伝送媒体を介して伝播することができる、任意の信号を含み得る。コードセグメントは、インターネットまたはイントラネットなどのコンピュータネットワークを介してダウンロードされ得る。いずれの場合も、本開示の範囲は、そのような態様によって限定されると解釈すべきではない。

【 0 0 4 9 】

開示した態様の上記の説明は、開示した態様を当業者が作成または使用できるようにするために提供される。これらの態様への様々な修正は当業者には容易に明らかになり、本明細書で定義された原理は、本開示の範囲から逸脱することなく、他の態様に適用されてもよい。したがって、本開示は、本明細書に示す態様に限定されることを意図するものではなく、以下の特許請求の範囲によって定義される原理および新規な特徴と可能な限り一致する最も広い範囲が与えられるべきである。

【 符号の説明 】

【 0 0 5 0 】

- 101 プロセッサ、CPU
- 102 キャッシュ、キャッシュメモリ
- 103 キャッシュ論理
- 104 タグ圧縮キャッシュ、TCC
- 105 キャッシュタグアレイ
- 106 データアレイ
- 108 メモリ
- 112 命令実行パイプライン、パイプライン
- 201 メモリアドレス
- 202 TCCエントリ
- 204 TCCインデックス値
- 205 キャッシュタグ
- 206 TCCエントリ
- 207 TCCインデックス値、インデックス値
- 208 TCCインデックス値
- 209 キャッシュタグ
- 210 比較器
- 211 比較器
- 212 AND論理
- 213 スプリッタ
- 300 概略図
- 400 方法
- 500 方法
- 601 コンピューティングデバイス
- 608 メモリ
- 609 ストレージ
- 618 ネットワークインターフェースデバイス
- 620 バス
- 622 入力デバイス
- 624 出力デバイス
- 630 ネットワーク

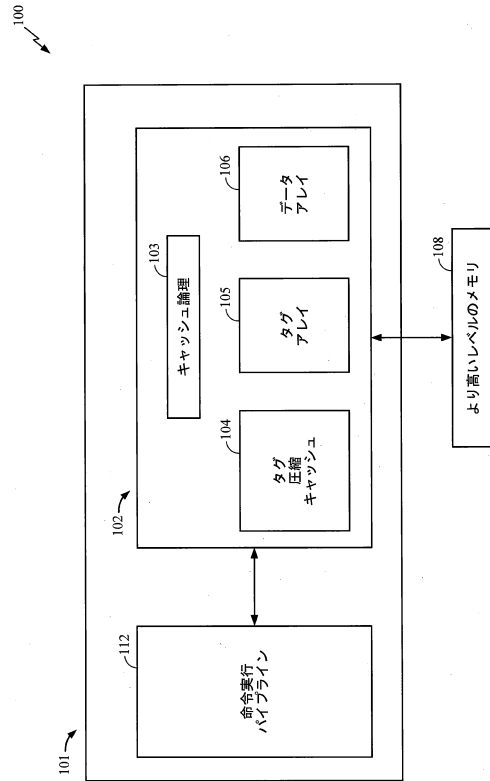
10

20

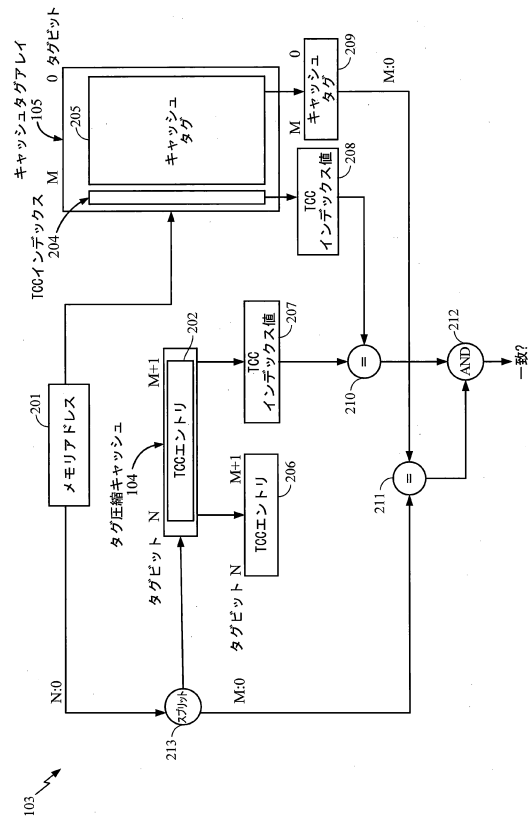
30

40

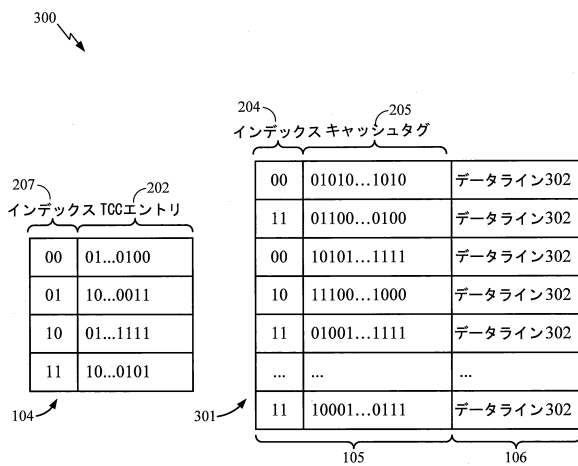
【図 1】



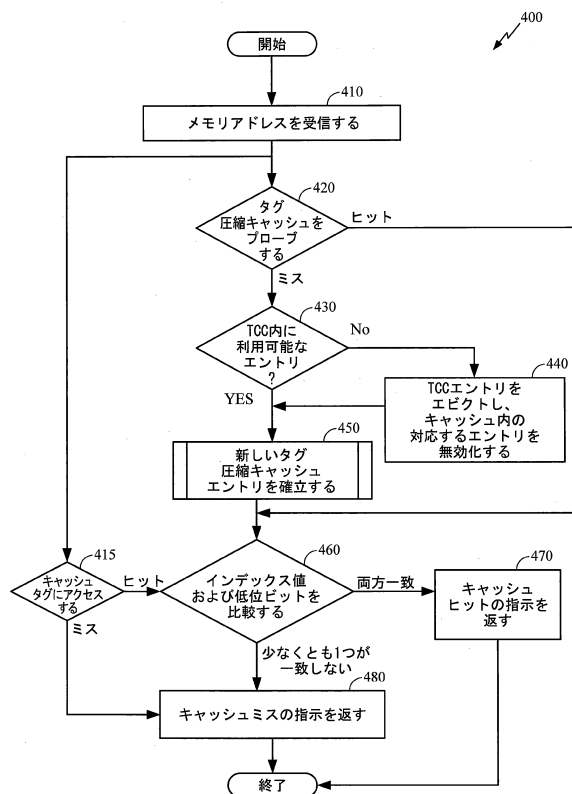
【図 2】



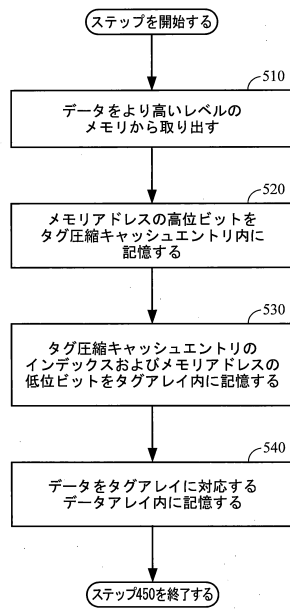
【図 3】



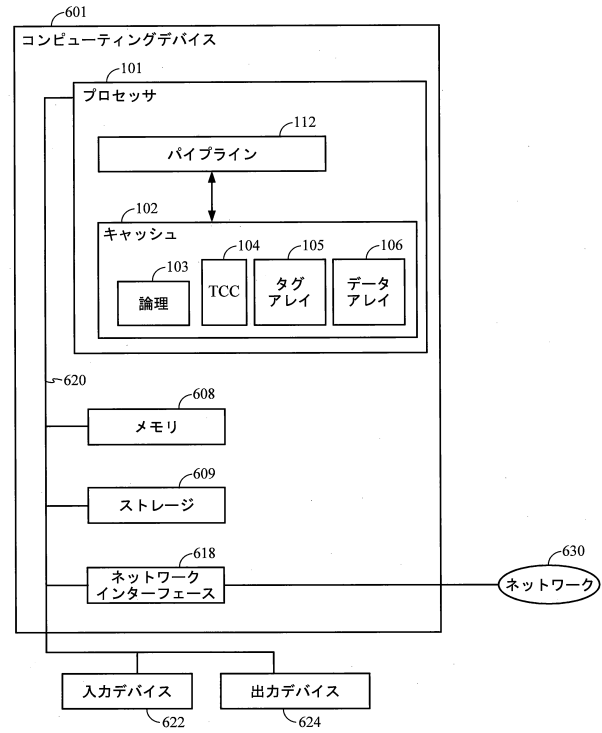
【図 4】



【図 5】



【図 6】



フロントページの続き

- (72)発明者 トーマス・フィリップ・スパイアー
アメリカ合衆国・カリフォルニア・9 2 1 2 1 - 1 7 1 4・サン・ディエゴ・モアハウス・ドライ
ヴ・5 7 7 5
- (72)発明者 トーマス・アンドリュー・サートリウス
アメリカ合衆国・カリフォルニア・9 2 1 2 1 - 1 7 1 4・サン・ディエゴ・モアハウス・ドライ
ヴ・5 7 7 5
- (72)発明者 マイケル・ウィリアム・モロー
アメリカ合衆国・カリフォルニア・9 2 1 2 1 - 1 7 1 4・サン・ディエゴ・モアハウス・ドライ
ヴ・5 7 7 5
- (72)発明者 ジェームズ・ノリス・ディーフェンデルファー
アメリカ合衆国・カリフォルニア・9 2 1 2 1 - 1 7 1 4・サン・ディエゴ・モアハウス・ドライ
ヴ・5 7 7 5
- (72)発明者 ケネス・アラン・ドックス
アメリカ合衆国・カリフォルニア・9 2 1 2 1 - 1 7 1 4・サン・ディエゴ・モアハウス・ドライ
ヴ・5 7 7 5
- (72)発明者 マイケル・スコット・マキルヴェイン
アメリカ合衆国・カリフォルニア・9 2 1 2 1 - 1 7 1 4・サン・ディエゴ・モアハウス・ドライ
ヴ・5 7 7 5

審査官 後藤 彰

- (56)参考文献 米国特許第0 6 1 2 2 7 0 9 (US, A)
特開2 0 1 2 - 0 0 3 3 1 4 (JP, A)
特開平0 8 - 0 0 6 8 5 7 (JP, A)
国際公開第2 0 1 1 / 0 4 9 0 5 1 (WO, A1)
米国特許出願公開第2 0 1 3 / 0 0 0 7 3 5 8 (US, A1)
特開2 0 1 5 - 0 6 9 6 4 9 (JP, A)
岩田 英次, 可変構造型並列計算機のキャッシュ・アーキテクチャ, 情報処理学会論文誌 第3
2 巻 第6号, 社団法人情報処理学会, 1 9 9 1年 6月, 国内学会論文1998-00290-008

- (58)調査した分野(Int.Cl., DB名)
G 0 6 F 1 2 / 0 8 9 5