

(19) 日本国特許庁(JP)

(12) 公表特許公報(A)

(11) 特許出願公表番号

特表2017-516234

(P2017-516234A)

(43) 公表日 平成29年6月15日(2017.6.15)

(51) Int.Cl.

G06F 12/08 (2016.01)
G06F 12/0804 (2016.01)

F 1

G06F 12/08
G06F 12/08505B
501C

テーマコード(参考)

5B005

審査請求 未請求 予備審査請求 有 (全 56 頁)

(21) 出願番号 特願2016-568009 (P2016-568009)
 (86) (22) 出願日 平成27年5月21日 (2015.5.21)
 (85) 翻訳文提出日 平成28年11月16日 (2016.11.16)
 (86) 國際出願番号 PCT/US2015/031885
 (87) 國際公開番号 WO2015/179591
 (87) 國際公開日 平成27年11月26日 (2015.11.26)
 (31) 優先権主張番号 62/001,545
 (32) 優先日 平成26年5月21日 (2014.5.21)
 (33) 優先権主張国 米国(US)
 (31) 優先権主張番号 62/092,326
 (32) 優先日 平成26年12月16日 (2014.12.16)
 (33) 優先権主張国 米国(US)
 (31) 優先権主張番号 62/092,409
 (32) 優先日 平成26年12月16日 (2014.12.16)
 (33) 優先権主張国 米国(US)

(71) 出願人 507364838
 クアルコム、インコーポレイテッド
 アメリカ合衆国 カリフォルニア 921
 21 サンディエゴ モアハウス ドラ
 イブ 5775
 (74) 代理人 100108453
 弁理士 村山 靖彦
 (74) 代理人 100163522
 弁理士 黒田 晋平
 (72) 発明者 マテウス・コーネリス・アントニウス・ア
 ドリアヌス・ヘディーズ
 アメリカ合衆国・カリフォルニア・921
 21・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】次の読み取りアドレスプリフェッティングによるメモリ容量圧縮および/またはメモリ帯域幅圧縮を利用するメモリコントローラ、ならびに関連するプロセッサベースシステムおよび方法

(57) 【要約】

次の読み取りアドレスプリフェッティングによるメモリ容量圧縮および/またはメモリ帯域幅圧縮を利用するメモリコントローラ、ならびに関連するプロセッサベースシステムおよび方法が開示される。いくつかの態様では、メモリ容量圧縮をもたらし得るメモリコントローラが利用される。本明細書で開示するいくつかの態様では、次の読み取りアドレスプリフェッティング方式は、現在アクセスされているアドレスを超える別のアドレスにおいてシステムメモリからデータを推論的にプリフェッチするために、メモリコントローラによって使用され得る。したがって、メモリデータが圧縮メモリ内にアドレス指定されるとき、次の読み取りアドレスが、被アクセスアドレスにおけるメモリブロックと関連付けられたメタデータ内に記憶される場合、次の読み取りアドレスにおけるメモリデータは、中央処理装置(CPU)によって発せられる後続の読み取り動作がメモリコントローラによってプリフェッチされている場合に利用可能となるメモリコントローラによってプリフェッチされ得る。

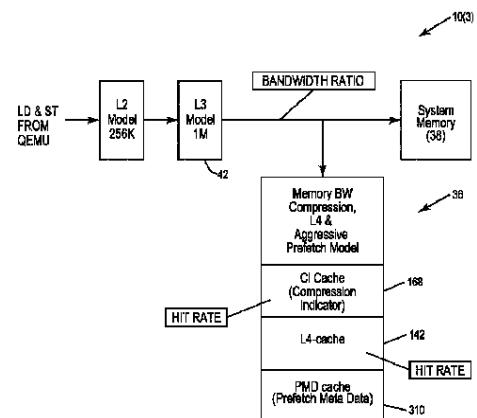


FIG. 23

【特許請求の範囲】

【請求項 1】

システムメモリにアクセスするように構成されたメモリインターフェースと、
コントローラとを備え、前記コントローラが、
システムバスにわたって前記システムメモリ内で読み取られるべき第1の物理アドレス(PA)を含む第1のメモリ読み取り要求を受信することと、

前記受信された第1のメモリ読み取り要求を実施するために、前記システムメモリ内の前記第1のPAにアクセスすることとを行うように構成され、

前記コントローラが、

前記システムバスにわたって前記システムメモリ内で読み取られるべき次のPAを含む
次のメモリ読み取り要求を受信することと、

前記受信された次のメモリ読み取り要求を実施するために、前記次のPAに基づいて前記
システムメモリにアクセスすることと、

前記第1のPAに関連する前記システムメモリ内に前記次のPAを記憶することとを行う
ようにさらに構成される、メモリコントローラ。

【請求項 2】

前記コントローラが、前記次のPAを、前記システムメモリ内の前記第1のPAと関連付け
されたメモリブロック内のメタデータとして記憶するように構成される、請求項1に記載
のメモリコントローラ。

【請求項 3】

前記コントローラが、
前記第1のPAと関連付けられた前記メモリブロックが、前記次のPAを記憶するために利
用可能な空間を含むかどうかを決定することと、

前記第1のPAと関連付けられた前記メモリブロックが前記次のPAに対して前記利用可
能な空間を含む場合に、前記次のPAを、前記システムメモリ内の前記第1のPAと関連付け
された前記メモリブロック内の前記メタデータとして記憶することとを行うように構成され
る、請求項2に記載のメモリコントローラ。

【請求項 4】

前記第1のPAと関連付けられた前記メモリブロック内の前記利用可能な空間が、前記メ
モリブロック内に記憶される圧縮メモリデータのサイズに基づく、請求項3に記載のメモ
リコントローラ。

【請求項 5】

前記コントローラが、
前記コントローラが現在のPAに対する読み取り動作を実施することに応答して、受信され
たメモリ読み取り要求に対する前記現在のPAが、レポジトリ内に含まれるかどうかを決定す
ることと、

前記次のPAが、前記レポジトリ内の前記受信されたメモリ読み取り要求に対する前記現
在のPAに関連して記憶される場合に、

前記システムメモリ内の前記現在のPAと関連付けられた前記次のPAを読み取ることと
、

前記次のPAにおいて前記システムメモリ内のメモリデータをプリフェッチし、前記次
のPAにおいて記憶される前記メモリデータをプリフェッチされたメモリデータとして取
得することとを行うようにさらに構成される、請求項1に記載のメモリコントローラ。

【請求項 6】

前記コントローラが、前記システムバスにわたって前記プリフェッチされたメモリデータ
を供給するようにさらに構成される、請求項5に記載のメモリコントローラ。

【請求項 7】

前記コントローラが、第1のPAエントリおよび対応する次のPAエントリをそれぞれが含
む複数のキャッシュエントリを含み、前記メモリコントローラに通信可能に結合されたプリ
フェッチメタデータ(PMD)キャッシュ内の前記第1のPAと関連付けられたキャッシュエン

10

20

30

40

50

トリ内の前記次のPAを記憶するようにさらに構成される、請求項1に記載のメモリコントローラ。

【請求項8】

前記コントローラが、プロセッサコアとそれぞれが関連付けられた複数のPMDキャッシュの中の前記PMDキャッシュ内の前記第1のPAと関連付けられた前記キャッシュエントリ内に前記次のPAを記憶するように構成され、前記複数のPMDキャッシュの各々が、前記第1のPAエントリおよび前記対応する次のPAエントリをそれぞれが含む前記複数のキャッシュエントリを含み、前記PMDキャッシュが、前記メモリコントローラに通信可能に結合される、請求項7に記載のメモリコントローラ。

【請求項9】

前記コントローラが、
受信されたメモリ読み取り要求に対する現在のPAが前記PMDキャッシュ内に含まれるかどうかを決定することと、

前記現在のPAが前記PMDキャッシュ内に含まれる場合に、

前記コントローラが、前記現在のPAに対する読み取り動作を実施することに応答して、前記PMDキャッシュ内の前記現在のPAと関連付けられた前記次のPAを読み取ることと、

前記次のPAにおいて前記システムメモリ内のメモリデータをプリフェッチし、前記次のPAにおいて記憶される前記メモリデータをプリフェッチされたメモリデータとして取得することとを行うようにさらに構成される、請求項7に記載のメモリコントローラ。

【請求項10】

前記コントローラが、前記システムバスにわたって前記受信されたメモリ読み取り要求に対する前記次のPAにおけるシステムメモリに記憶された前記プリフェッチされたメモリデータを供給するようにさらに構成される、請求項9に記載のメモリコントローラ。

【請求項11】

前記受信されたメモリ読み取り要求に対する前記現在のPAが前記PMDキャッシュ内に含まれない場合に、前記コントローラが、

前記システムメモリ内の前記現在のPAにアクセスして、前記システムメモリ内の前記現在のPA内に記憶された前記対応するメモリデータを取得することと、

前記プリフェッチされたメモリデータが記憶された前記PMDキャッシュ内の前記キャッシュエントリに対する前記現在のPAに対応する前記PMDキャッシュ内の前記キャッシュエントリを更新することとを行うようにさらに構成される、請求項9に記載のメモリコントローラ。

【請求項12】

前記コントローラが、
前記PMDキャッシュ内に利用可能なキャッシュエントリがあるかどうかを決定することと、

前記PMDキャッシュ内に利用可能なキャッシュエントリがない場合に、前記PMDキャッシュ内の前記キャッシュエントリを立ち退かせて、前記立ち退かされたキャッシュエントリ内の前記現在のPAを前記PMDキャッシュ内に記憶することとを行うようにさらに構成される、請求項11に記載のメモリコントローラ。

【請求項13】

アドレスエントリに関連するメモリデータを記憶するようにそれぞれが構成された複数のデータキャッシュエントリを含むデータキャッシュをさらに備え、

前記コントローラが、

前記現在のPAが、前記データキャッシュ内のデータキャッシュエントリ内のアドレスエントリと一致するかどうかを決定することと、

前記現在のPAが、前記データキャッシュの前記データキャッシュエントリ内の前記アドレスエントリ内に含まれる場合に、前記データキャッシュ内の前記現在のPAにおける前記システムメモリ内に記憶される前記対応するメモリデータを、前記受信されたメモリ読み取り要求に対する前記プリフェッチされたメモリデータとして供給することとを行うよう

10

20

30

40

50

にさらに構成される、請求項9に記載のメモリコントローラ。

【請求項 14】

前記現在のPAが、前記データキャッシュの前記データキャッシュエントリ内の前記アドレスエントリ内に含まれる場合に、前記コントローラが、

前記PMDキャッシュ内の前記現在のPAと関連付けられた前記次のPAを読み取ることと、

前記次のPAにおいて前記システムメモリ内のメモリデータをプリフェッチし、前記次のPAにおいて記憶される前記メモリデータを、前記受信されたメモリ読み取り要求に対する前記プリフェッチされたメモリデータとして取得することとを行うようにさらに構成される、請求項13に記載のメモリコントローラ。

【請求項 15】

前記現在のPAが、前記データキャッシュの前記データキャッシュエントリ内の前記アドレスエントリと一致しない場合に、前記コントローラが、

前記データキャッシュ内のデータキャッシュエントリを立ち退かせることと、

前記現在のPAにおける前記システムメモリからメモリデータを読み取って、前記メモリデータを前記データキャッシュ内に記憶することと、

次のPAが前記PMDキャッシュ内に含まれるかどうかを決定することと、

前記次のPAが前記PMDキャッシュ内に含まれる場合に、前記次のPAにおいて前記システムメモリからメモリデータを、前記受信されたメモリ読み取り要求に対する前記プリフェッチされたメモリデータとしてプリフェッチすることとを行うようにさらに構成される、請求項13に記載のメモリコントローラ。

10

20

30

30

【請求項 16】

前記現在のPAが、前記データキャッシュの前記データキャッシュエントリ内の前記アドレスエントリと一致しない場合に、前記コントローラが、

前記PMDキャッシュ内の前記現在のPAと関連付けられた前記次のPAにアクセスすることと、

前記立ち退かされたデータキャッシュエントリがデータである場合に、前記次のPAを、前記現在のPAと関連付けられた前記システムメモリ内に書き戻すことと、

前記PMDキャッシュ内の前記現在のPAと関連付けられた前記次のPAが、設定可能な確率によってデータである場合に、前記次のPAを、前記現在のPAと関連付けられた前記システムメモリ内に書き戻すこととを行うようにさらに構成される、請求項15に記載のメモリコントローラ。

30

【請求項 17】

前記次のPAが前記PMDキャッシュ内に含まれない場合に、前記コントローラが、

前記システムメモリ内の前記現在のPAと関連付けられた前記次のPAを読み取ることと、

前記システムメモリ内の前記現在のPAと関連付けられた前記次のPAが有効である場合に、前記次のPAにおいて前記システムメモリからメモリデータを、前記受信されたメモリ読み取り要求に対する前記プリフェッチされたメモリデータとしてプリフェッチすることと、

前記次のPAを、前記現在のPAに関連する前記PMDキャッシュ内に書き戻すこととを行うようにさらに構成される、請求項14に記載のメモリコントローラ。

40

【請求項 18】

前記コントローラが、

第2のPAを含むメモリ書き込み要求を受信して、前記システムバスにわたって前記システムメモリ内に記憶されるべきメモリデータを書き込むことと、

前記第2のPAが、前記データキャッシュ内のデータキャッシュエントリ内のアドレスエントリと一致するかどうかを決定することと、

前記第2のPAが、前記データキャッシュの前記データキャッシュエントリ内の前記アドレスエントリ内に含まれる場合に、前記書き込みメモリデータを前記第2のPAと関連付けられた前記データキャッシュ内に記憶することと、

前記第2のPAが、前記データキャッシュの前記データキャッシュエントリ内の前記アドレスエントリ内に含まれない場合に、データキャッシュエントリを前記データキャッシュ

50

から立ち退かせて、前記書込みメモリデータを前記第2のPAと関連付けられた前記データキャッシュ内に記憶することとを行わせるようにさらに構成される、請求項13に記載のメモリコントローラ。

【請求項 19】

前記第2のPAが、前記データキャッシュの前記データキャッシュエントリ内の前記アドレスエントリ内に含まれない場合に、前記コントローラが、

前記PMDキャッシュ内の前記第2のPAと関連付けられた前記次のPAにアクセスすることと、

前記立ち退かされたデータキャッシュエントリがダーティである場合に、前記次のPAを、前記第2のPAと関連付けられた前記システムメモリ内に書き戻すことと、

前記PMDキャッシュ内の前記第2のPAと関連付けられた前記次のPAが、設定可能な確率によってダーティである場合に、前記次のPAを、前記第2のPAと関連付けられた前記システムメモリ内に書き戻すこととを行うようにさらに構成される、請求項18に記載のメモリコントローラ。

10

【請求項 20】

集積回路(IC)に組み込まれる、請求項1に記載のメモリコントローラ。

【請求項 21】

セットトップボックス、エンターテインメントユニット、ナビゲーションデバイス、通信デバイス、固定ロケーションデータユニット、モバイルロケーションデータユニット、モバイルフォン、セルラーフォン、コンピュータ、ポータブルコンピュータ、デスクトップコンピュータ、携帯情報端末(PDA)、モニタ、コンピュータモニタ、テレビジョン、チューナ、ラジオ、衛星ラジオ、音楽プレーヤ、デジタル音楽プレーヤ、ポータブル音楽プレーヤ、デジタルビデオプレーヤ、ビデオプレーヤ、デジタルビデオディスク(DVD)プレーヤ、およびポータブルデジタルビデオプレーヤからなるグループから選択されるデバイスに組み込まれる、請求項1に記載のメモリコントローラ。

20

【請求項 22】

システムバスにわたってシステムメモリ内で読み取られるべき第1の物理アドレス(PA)を含む第1のメモリ読み取り要求を受信するための手段と、

前記受信された第1のメモリ読み取り要求を実施するために、前記システムメモリ内の前記第1のPAにアクセスするための手段と、

30

前記システムバスにわたって前記システムメモリ内で読み取られるべき次のPAを含む次のメモリ読み取り要求を受信するための手段と、

前記受信された次のメモリ読み取り要求を実施するために、前記システムメモリ内の前記次のPAに基づいて前記システムメモリにアクセスするための手段と、

前記第1のPAに関連する前記システムメモリ内に前記次のPAを記憶するための手段とを含む、メモリコントローラ。

【請求項 23】

中央処理装置(CPU)ベースシステム内のシステムメモリに対するメモリ読み取り要求に対するデータを圧縮する方法であって、

システムバスにわたってシステムメモリ内で読み取られるべき第1の物理アドレス(PA)を含む第1のメモリ読み取り要求を受信するステップと、

40

前記第1のメモリ読み取り要求を実施するために、前記システムメモリ内の前記第1のPAにアクセスするステップと、

前記システムバスにわたって前記システムメモリ内で読み取られるべき次のPAを含む次のメモリ読み取り要求を受信するステップと、

前記受信された次のメモリ読み取り要求を実施するために、前記システムメモリ内の前記次のPAにアクセスするステップと、

前記第1のPAと関連付けられた前記システムメモリ内に前記次のPAを記憶するステップとを含む、方法。

【請求項 24】

50

前記次のPAを、前記システムメモリ内の前記第1のPAと関連付けられたメモリブロック内のメタデータとして記憶するステップを含む、請求項23に記載の方法。

【請求項 25】

前記第1のPAと関連付けられた前記メモリブロックが、前記次のPAを記憶するために利用可能な空間を含むかどうかを決定するステップと、

前記第1のPAと関連付けられた前記メモリブロックが前記次のPAに対して利用可能な空間を含む場合に、前記次のPAを、前記システムメモリ内の前記第1のPAと関連付けられた前記メモリブロック内の前記メタデータとして記憶するステップとを含む、請求項24に記載の方法。

【請求項 26】

コントローラが前記現在のPAに対する読み取り動作を実施することに応答して、受信されたメモリ読み取り要求に対する現在のPAが、レポジトリ内に含まれるかどうかを決定するステップと、

前記次のPAが、前記レポジトリ内の前記受信されたメモリ読み取り要求に対する前記現在のPAに関連して記憶される場合に、

前記システムメモリ内の前記現在のPAと関連付けられた前記次のPAを読み取るステップと、

前記次のPAにおいて前記システムメモリ内のメモリデータをプリフェッチし、前記次のPAにおいて記憶される前記メモリデータをプリフェッチされたメモリデータとして取得するステップとをさらに含む、請求項23に記載の方法。

【請求項 27】

第1のPAエントリおよび対応する次のPAエントリをそれぞれが含む複数のキャッシュエントリを含み、前記コントローラに通信可能に結合されたプリフェッチメタデータ(PMD)キャッシュ内の前記第1のPAと関連付けられたキャッシュエントリ内の前記次のPAを記憶するステップをさらに含む、請求項23に記載の方法。

【請求項 28】

コントローラが前記現在のPAに対する読み取り動作を実施することに応答して、前記受信されたメモリ読み取り要求に対する現在のPAが、前記PMDキャッシュ内に含まれるかどうかを決定するステップと、

前記現在のPAが前記PMDキャッシュ内に含まれる場合に、

前記コントローラが、前記現在のPAに対する前記読み取り動作を実施することに応答して、前記PMDキャッシュ内の前記現在のPAと関連付けられた前記次のPAを読み取るステップ、および

前記次のPAにおいて前記システムメモリ内の前記メモリデータをプリフェッチし、前記次のPAにおいて記憶された前記メモリデータを前記プリフェッチされたメモリデータとして取得するステップと、

前記システムバスにわたって前記受信されたメモリ読み取り要求に対して前記次のPAにおいて前記PMDキャッシュ内に記憶された前記プリフェッチされたメモリデータを供給するステップとをさらに含む、請求項27に記載の方法。

【請求項 29】

前記受信されたメモリ読み取り要求に対する前記現在のPAが前記PMDキャッシュ内に含まれない場合に、前記コントローラが、

前記システムメモリ内の前記現在のPAにアクセスして、前記システムメモリ内の前記現在のPA内に記憶された前記対応するメモリデータを取得するステップと、

前記プリフェッチされたメモリデータが記憶された前記PMDキャッシュ内の前記キャッシュエントリに対する前記現在のPAに対応する前記PMDキャッシュ内の前記キャッシュエントリを更新するステップと、

前記PMDキャッシュ内に利用可能なキャッシュエントリがあるかどうかを決定するステップと、

前記PMDキャッシュ内に利用可能なキャッシュエントリがない場合に、前記PMDキャッシュ

10

20

30

40

50

ユ内の前記キャッシュエントリを立ち退かせて、前記立ち退かされたキャッシュエントリ内の前記現在のPAを前記PMDキャッシュ内に記憶するステップとをさらに含む、請求項28に記載の方法。

【請求項 30】

システムバスと、
前記システムバスに通信可能に結合された少なくとも1つのCPUコアと、
前記システムバスに通信可能に結合されたメモリコントローラと、
前記システムバスに通信可能に結合され、メモリデータを記憶するように構成されたシステムメモリと、
アドレスエントリおよび対応する次の読み取りアドレスをそれぞれが含む複数のキャッシュエントリを含み、前記メモリコントローラに通信可能に結合されたプリフェッチメタデータ(PMD)キャッシュとを備え、
前記メモリコントローラが、
前記システムバスにわたって前記システムメモリ内で読み取られるべき第1の物理アドレス(PA)を含む第1のメモリ読み取り要求を受信することと、
前記受信された第1のメモリ読み取り要求を実施するために、前記システムメモリ内の前記第1のPAにアクセスすることとを行うように構成されたコントローラを備え、
前記コントローラが、
前記システムバスにわたって前記システムメモリ内で読み取られるべき次のPAを含む次のメモリ読み取り要求を受信することと、
前記受信された次のメモリ読み取り要求を実施するために、前記次のPAに基づいて前記システムメモリにアクセスすることと、
前記第1のPAに関連する前記システムメモリ内に前記次のPAを記憶することとを行うようにさらに構成される、中央処理装置(CPU)システム。

【発明の詳細な説明】

【技術分野】

【0001】

優先権出願

本出願は、その全体が参照により本明細書に組み込まれている、2014年5月21日に出願した「MEMORY CONTROLLERS EMPLOYING MEMORY CAPACITY AND/OR BANDWIDTH COMPRESSION, AND RELATED PROCESSOR-BASED SYSTEMS AND METHODS」と題する米国仮特許出願第62/001,545号の優先権を主張するものである。

【0002】

本出願は、同様に、その全体が参照により本明細書に組み込まれている、2014年12月16日に出願した「MEMORY CONTROLLERS EMPLOYING MEMORY CAPACITY AND/OR BANDWIDTH COMPRESSION, AND RELATED PROCESSOR-BASED SYSTEMS AND METHODS」と題する米国仮特許出願第62/092,326号の優先権を主張するものである。

【0003】

本出願は、同様に、その全体が参照により本明細書に組み込まれている、2014年12月16日に出願した「MEMORY CONTROLLERS EMPLOYING MEMORY CAPACITY AND/OR BANDWIDTH COMPRESSION WITH NEXT READ ADDRESS PREFETCHING, AND RELATED PROCESSOR-BASED SYSTEMS AND METHODS」と題する米国仮特許出願第62/092,409号の優先権を主張するものである。

【0004】

本出願は、同様に、その全体が参照により本明細書に組み込まれている、2015年5月19日に出願した「MEMORY CONTROLLERS EMPLOYING MEMORY CAPACITY AND/OR BANDWIDTH COMPRESSION WITH NEXT READ ADDRESS PREFETCHING, AND RELATED PROCESSOR-BASED SYSTEMS AND METHODS」と題する米国特許出願第14/716,108号の優先権も主張するものである。

【0005】

開示する技術は、一般に、コンピュータメモリシステムに関し、詳細には、メモリに対するメモリアクセスインターフェースを有する中央処理装置(CPU)を提供するためのコン

10

20

30

40

50

ピュータメモリシステム内のメモリコントローラに関する。

【背景技術】

【0006】

マイクロプロセッサは、多種多様なアプリケーションにおいて計算タスクを実施する。典型的なマイクロプロセッサアプリケーションは、ソフトウェア命令を実行する1つまたは複数の中央処理装置(CPU)を含む。ソフトウェア命令は、メモリ内のロケーションからデータをフェッチし、フェッチされたデータを使用して1つまたは複数のCPU動作を実施し、結果を生成するようにCPUに命令する。次いで、結果はメモリに記憶されてもよい。例として、このメモリは、CPUに局在するキャッシュ、CPUブロックにおいてCPU間で共有されるローカルキャッシュ、複数のCPUブロック間で共有されるキャッシュ、またはマイクロプロセッサのメインメモリであり得る。

10

【0007】

この点について、図1は、CPUベースシステム12を含む例示的なシステムオンチップ(SOC)10の概略図である。CPUベースシステム12は、この例では、複数のCPUブロック14(1)～14(N)を含み、「N」は、所望のCPUブロック14の任意の数に等しい。CPUブロック14(1)～14(N)の各々は、それぞれ、2つのCPU 16(1)、16(2)と共有レベル2(L2)のキャッシュ18(1)～18(N)とを含む。共有レベル3(L3)のキャッシュ20はまた、CPUブロック14(1)～14(N)の各々のいずれかによって使用されるか、またはCPUブロック14(1)～14(N)の各々の間で共有されるキャッシュデータを記憶するために設けられる。CPUブロック14(1)～14(N)の各々が共有L3キャッシュ20ならびに他の共有リソースにアクセスすることを可能にする内部システムバス22が設けられる。内部システムバス22を介してCPUブロック14(1)～14(N)によってアクセスされ得る他の共有リソースは、メイン外部メモリ(たとえば、ダブルレートダイナミックランダムアクセスメモリ(DRAM)(DDR))にアクセスするためのメモリコントローラ24と、周辺機器26と、他の記憶装置28と、高速周辺機器相互接続(PCI)(PCI-e)インターフェース30と、直接メモリアクセス(DMA)コントローラ32と、IMC34とを含むことができる。

20

【0008】

図1のCPUベースシステム12において実行するCPUベースアプリケーションが、複雑さと性能とを増すにつれて、共有L2キャッシュ18および共有L3キャッシュ20ならびにメモリコントローラ24を介してアクセス可能な外部メモリのメモリ容量要求も増加することがある。しかしながら、CPUベースシステム内に追加のメモリ容量を設けることは、集積回路上のメモリに必要なコストと面積とを増す。たとえば、図1のCPUベースシステム12などのCPUベースシステムがSOC内に設けられるとすれば、追加のメモリ容量を付加することは、SOCパッケージングを増加させることがある。データ圧縮は、物理メモリ容量を増加することなく、CPUベースシステムの有効メモリ容量を増加するために利用されてよい。しかしながら、データ圧縮は、データが圧縮された形態で記憶され、次いでメモリアクセスの間に解凍されなければならないので、CPUメモリアクセス時間を縮小し、したがってCPU性能を低下させる。

30

【発明の概要】

【発明が解決しようとする課題】

40

【0009】

したがって、CPUベースのアプリケーションの複雑さおよび性能要求が増加するときの物理メモリサイズの増加を軽減しながら、またCPU性能に及ぼす影響を最小化しながら、CPUベースシステムのメモリ容量を増加することが望ましいであろう。

【課題を解決するための手段】

【0010】

発明を実施するための形態において開示する態様は、次の読み取りアドレスプリフェッチングによるメモリ容量圧縮および/またはメモリ帯域幅圧縮を利用するメモリコントローラ、ならびに関連するプロセッサベースシステムおよび方法を含む。いくつかの態様では、メモリ容量圧縮をもたらし得るメモリコントローラが利用される。本明細書で開示する

50

いくつかの態様では、次の読み取りアドレスプリフェッティング方式は、メモリ帯域幅圧縮方式および/またはメモリ容量圧縮方式と組み合わされてよい。次の読み取りアドレスプリフェッチ方式は、現在アクセスされているアドレスを超える別のアドレスにおいてシステムメモリからデータを推論的にプリフェッチするために、メモリコントローラによって使用され得る。圧縮メモリ内で圧縮メモリデータを保持するメモリブロック内に使い残し空間が存在する所以があるので、システムまたはメインメモリからプリフェッチされるべき次の読み取りアドレスは、圧縮メモリ内に記憶される圧縮メモリデータに関連するメタデータとして記憶され得る。したがって、メモリデータが圧縮メモリ内にアドレス指定されるとき、次の読み取りアドレスが、被アクセスアドレスにおけるメモリブロックと関連付けられたメタデータ内に記憶される場合、次の読み取りアドレスにおけるメモリデータは、中央処理装置(CPU)によって発せられる後続の読み取り動作がメモリコントローラによってプリフェッチされている場合に利用可能となるメモリコントローラによってプリフェッチされ得る。メモリ帯域幅圧縮方式および/またはメモリ容量圧縮方式とともに使用されると、圧縮メモリ内でメモリブロック内の使い残し空間を使用しながら付加的性能が達成され得る。

10

【0011】

この点について、一態様では、メモリコントローラが提供される。メモリコントローラは、システムメモリにアクセスするように構成されたメモリインターフェース、およびコントローラを備える。コントローラは、システムバスにわたってシステムメモリ内で読み取られるべき第1の物理アドレス(PA)を含む第1のメモリ読み取り要求を受信するように構成される。コントローラはまた、受信された第1のメモリ読み取り要求を実施するために、システムメモリ内の第1のPAにアクセスするように構成される。コントローラは、システムバスにわたってシステムメモリ内で読み取られるべき次のPAを含む次のメモリ読み取り要求を受信するようにさらに構成される。コントローラはまた、受信された第1のメモリ読み取り要求を実施するために、次のPAに基づいてシステムメモリにアクセスするように構成される。コントローラは、第1のPAに関連するシステムメモリ内に次のPAを記憶するようにさらに構成される。

20

【0012】

別の態様では、メモリコントローラが提供される。メモリコントローラは、システムバスにわたってシステムメモリ内で読み取られるべき第1のPAを含む第1のメモリ読み取り要求を受信するための手段を含む。メモリコントローラはまた、受信された第1のメモリ読み取り要求を実施するために、システムメモリ内の第1のPAにアクセスするための手段を含む。メモリコントローラはまた、システムバスにわたってシステムメモリ内で読み取られるべき次のPAを含む次のメモリ読み取り要求を受信するための手段を含む。メモリコントローラはまた、受信された次のメモリ読み取り要求を実施するために、システムメモリ内の次のPAに基づいてシステムメモリにアクセスするための手段を含む。メモリコントローラはまた、第1のPAに関連するシステムメモリ内に次のPAを記憶するための手段を含む。

30

【0013】

別の態様では、中央処理装置(CPU)ベースシステム内のシステムメモリへのメモリ読み取り要求に対するデータを圧縮する方法が提供される。方法は、システムバスにわたってシステムメモリ内で読み取られるべき第1のPAを含む第1のメモリ読み取り要求を受信するステップを含む。方法はまた、第1のメモリ読み取り要求を実施するために、システムメモリ内の第1のPAにアクセスするステップを含む。方法はまた、システムバスにわたってシステムメモリ内で読み取られるべき次のPAを含む次のメモリ読み取り要求を受信するステップを含む。方法はまた、受信された次のメモリ読み取り要求を実施するために、システムメモリ内の次のPAにアクセスするステップを含む。方法はまた、第1のPAと関連付けられたシステムメモリ内に次のPAを記憶するステップを含む。

40

【0014】

別の態様では、CPUシステムが提供される。CPUシステムは、システムバスを備える。CPUシステムはまた、システムバスに通信可能に結合された少なくとも1つのCPUコアを備え

50

る。CPUシステムはまた、システムバスに通信可能に結合されたメモリコントローラを備える。CPUシステムはまた、システムバスに通信可能に結合され、メモリデータを記憶するように構成されたシステムメモリを備える。CPUシステムはまた、アドレスエントリおよび対応する次の読み取りアドレスをそれぞれが含む複数のキャッシュエントリを含み、メモリコントローラに通信可能に結合されたプリフェッチメタデータ(PMD)キャッシュを備える。メモリコントローラはコントローラを備える。コントローラは、システムバスにわたってシステムメモリ内で読み取られるべき第1のPAを含む第1のメモリ読み取り要求を受信するように構成される。コントローラはまた、受信された第1のメモリ読み取り要求を実施するために、システムメモリ内の第1のPAにアクセスするように構成される。コントローラはまた、システムバスにわたってシステムメモリ内で読み取られるべき次のPAを含む次のメモリ読み取り要求を受信するように構成される。コントローラはまた、受信された次のメモリ読み取り要求を実施するために、次のPAに基づいてシステムメモリにアクセスするように構成される。コントローラはまた、第1のPAに関連するシステムメモリ内に次のPAを記憶するように構成される。

10

【0015】

他の様では、小さいデータブロック圧縮に好適であり得る圧縮の方法およびフォーマットが開示される。これらの圧縮の方法およびフォーマットは、本明細書で開示するメモリ容量および/またはメモリ帯域幅の圧縮の様に利用され得る。

20

【0016】

これらのメモリコントローラおよび圧縮方式の一部または全部によって、CPUベースアプリケーションの複雑さおよび性能要件が増加するときに、物理メモリサイズの増加を軽減しながら、また帯域幅および電力の消費を低減しながら、CPUベースシステムのメモリ容量を増加することが可能である。

30

【図面の簡単な説明】

【0017】

【図1】中央処理装置(CPU)ベースシステムを含む例示的なシステムオンチップ(SOC)の概略図である。

30

【図2】複数のCPUとメモリ容量および/またはメモリ帯域幅の圧縮を利用するメモリコントローラとを有する例示的なCPUベースシステムを含むSOCの概略図である。

【図3】圧縮メモリコントローラが、さらに、メモリ容量および/またはメモリ帯域幅の圧縮をもたらすために使用される随意の内部メモリに通信可能に結合される、図2のメモリコントローラのより詳細な概略図である。

40

【図4】図3の圧縮メモリコントローラによって実装され得る例示的な行ベースバッファメモリ容量圧縮方式の概略図である。

【図5】ページベースバッファメモリ容量圧縮方式を提供するために図3の圧縮メモリコントローラによってアクセス可能な圧縮メモリに記憶される例示的なデータページ(DP)の概略図である。

40

【図6】ページベースバッファメモリ容量圧縮方式を提供するために、物理アドレス(PA)と、圧縮メモリに記憶される圧縮データページへのデータページアドレス(DPA)とを相關させるように構成され、図3の圧縮メモリコントローラによってアクセス可能な例示的なトランスレーションルックアサイドバッファ(TLB)の概略図である。

【図7】図4～図6に含まれるページベースまたは行ベースのバッファメモリ容量圧縮方式において利用され得る例示的なフリーリストバッファの概略図である。

50

【図8】図3の圧縮メモリコントローラによって実装され得る例示的なハイブリッド行/ページベースバッファメモリ容量圧縮方式の概略図である。

【図9】図8のハイブリッド行/ページベースバッファメモリ容量圧縮方式をもたらすための圧縮キャッシュラインを記憶するために、図3の圧縮メモリコントローラによってアクセス可能な圧縮メモリに記憶された、複数のメモリデータブロックをそれぞれ備える複数のデータバッファ(DB)から成る例示的なデータページ(DP)の概略図である。

【図10】メモリアクセスのために図8のハイブリッド行/ページベースバッファメモリ容

50

量圧縮方式に従って図9の割り振られたデータページ(DP)内の割り振られたDBにPAを変換する、図3の圧縮メモリコントローラの例示的なプロセスを示すフローチャートである。

【図11A】圧縮メモリコントローラ内のアドレス変換による性能損失を補償するための随意のL4キャッシュを追加で含む、図3のSOCの一例を示す図である。

【図11B】L4キャッシュ内でヒットまたはミスが決定される前に、PAをロックアップするために圧縮メモリコントローラ内でL4キャッシュおよびトランスレーショナルックアサイドバッファ(TLB)のプライミングを伴う、図11AのSOCの一例を示す図である。

【図12】例示的なダイナミックメモリサイジングの概略図である。

【図13】図3の圧縮メモリコントローラによって実装され得る例示的なメモリ帯域幅圧縮方式の概略図である。

【図14】圧縮メモリコントローラ内のアドレス変換による性能損失を補償するための随意のL4キャッシュを含む、図3のSOCの一例を示す図である。

【図15】メモリデータブロックを圧縮し、圧縮されたメモリデータブロックを展開するために圧縮メモリコントローラによって使用されてよい、例示的なデータブロック圧縮フォーマットおよび方式を示す図である。

【図16】メモリデータブロックを圧縮し、圧縮されたメモリデータブロックを展開するために圧縮メモリコントローラによって使用されてよい、例示的なデータブロック圧縮フォーマットおよび方式を示す図である。

【図17】メモリデータブロックを圧縮し、圧縮されたメモリデータブロックを展開するために圧縮メモリコントローラによって使用されてよい、例示的なデータブロック圧縮フォーマットおよび方式を示す図である。

【図18】メモリデータブロックを圧縮し、圧縮されたメモリデータブロックを展開するために圧縮メモリコントローラによって使用されてよい、例示的なデータブロック圧縮フォーマットおよび方式を示す図である。

【図19】メモリデータブロックを圧縮し、圧縮されたメモリデータブロックを展開するために圧縮メモリコントローラによって使用されてよい、例示的なデータブロック圧縮フォーマットおよび方式を示す図である。

【図20】メモリデータブロックを圧縮し、圧縮されたメモリデータブロックを展開するために圧縮メモリコントローラによって使用されてよい、例示的なデータブロック圧縮フォーマットおよび方式を示す図である。

【図21】メモリデータブロックを圧縮し、圧縮されたメモリデータブロックを展開するために圧縮メモリコントローラによって使用されてよい、例示的なデータブロック圧縮フォーマットおよび方式を示す図である。

【図22A】解凍形態におけるメモリデータブロック/行内に編成された例示的なメモリデータの図である。

【図22B】次の読み取りアドレスプリフェッチ方式によるメモリ帯域幅圧縮方式を利用するメモリコントローラによって圧縮された後の図22Aのメモリデータの図である。

【図22C】次の読み取りアドレスプリフェッチ方式によるメモリ容量圧縮方式を利用するメモリコントローラによって圧縮された後の図22Aのメモリデータの図である。

【図23】次の読み取りアドレスプリフェッチ方式によるメモリ容量圧縮方式を利用するメモリデータ圧縮を実施するように構成された例示的なメモリコントローラを利用する、図3のSOCに対する代替SOCの構造図である。

【図24A】次の読み取りアドレスプリフェッチ方式によるメモリ容量圧縮方式を利用するメモリデータ圧縮をメモリコントローラが実施することを可能にするために、図23のSOCにおいて利用され得る例示的なデータ構造の概略図である。

【図24B】次の読み取りアドレスプリフェッチ方式によるメモリ容量圧縮方式を利用するメモリデータ圧縮をメモリコントローラが実施することを可能にするために、図23のSOCにおいて利用され得る例示的なデータ構造の概略図である。

【図24C】次の読み取りアドレスプリフェッチ方式によるメモリ容量圧縮方式を利用するメモリデータ圧縮をメモリコントローラが実施することを可能にするために、図23のSOC

10

20

30

40

50

において利用され得る例示的なデータ構造の概略図である。

【図25】追加の帯域幅低減をもたらすための随意のキャッシュを含む、図23のSOCを示す図である。

【発明を実施するための形態】

【0018】

ここで図面を参照しながら、本開示のいくつかの例示的な態様について説明する。「例示的」という語は、本明細書では「一例、事例、または例示としての役割を果たすこと」を意味するために使用される。「例示的」として本明細書において説明されるいずれの態様も、必ずしも他の態様よりも好ましいか、または有利であると解釈されることは限らない。

10

【0019】

発明を実施するための形態において開示する態様は、次の読み取りアドレスプリフェッティングによるメモリ容量圧縮および/または帯域幅圧縮を利用するメモリコントローラ、ならびに関連するプロセッサベースシステムおよび方法を含む。いくつかの態様では、メモリ容量圧縮をもたらし得るメモリコントローラが利用される。本明細書で開示するいくつかの態様では、次の読み取りアドレスプリフェッティング方式は、メモリ帯域幅圧縮方式および/またはメモリ容量圧縮方式と組み合わされてよい。次の読み取りアドレスプリフェッティング方式は、現在アクセスされているアドレスを超える別のアドレスにおいてシステムメモリからデータを推論的にプリフェッチするために、メモリコントローラによって使用され得る。圧縮メモリ内で圧縮メモリデータを保持するメモリブロック内に使い残し空間が存在する事があるので、システムまたはメインメモリからプリフェッチされるべき次の読み取りアドレスは、圧縮メモリ内に記憶される圧縮メモリデータに関連するメタデータとして記憶され得る。したがって、メモリデータが圧縮メモリ内にアドレス指定されるとき、次の読み取りアドレスが、被アクセスアドレスにおけるメモリブロックと関連付けられたメタデータ内に記憶される場合、次の読み取りアドレスにおけるメモリデータは、中央処理装置(CPU)によって発せられる後続の読み取り動作がメモリコントローラによってプリフェッチされている場合に利用可能となるメモリコントローラによってプリフェッチされ得る。メモリ帯域幅圧縮方式および/またはメモリ容量圧縮方式とともに使用されると、圧縮メモリ内でメモリブロック内の使い残し空間を使用しながら付加的性能が達成され得る。

20

【0020】

次の読み取りアドレスプリフェッティングの例を説明する前に、次の読み取りアドレスプリフェッティングを利用し得る例示的なCPUベースシステムおよびメモリ圧縮方式について、図2～図21について以下で説明する。次の読み取りアドレスプリフェッティングの例は、以下の図22Aにおいて開始する。

30

【0021】

この点について、図2は、図1のCPUベースシステム12と同様に、複数のCPUブロック14(1)～14(N)を有する例示的なCPUベースシステム12'を含むSOC10'の概略図である。図2のCPUベースシステム12'は、図1のCPUベースシステム12と共にいくつかの構成要素を含み、それらは、図1と図2との間で共通の要素番号で示され、したがって再説明されない。しかしながら、図2のCPUベースシステム12'において、圧縮メモリコントローラ36が設けられる。圧縮メモリコントローラ36は、システムメモリ38へのアクセスを制御する。システムメモリ38は、非限定的な例として、1つまたは複数のダブルデータレート(DDR)ダイナミックランダムアクセスメモリ(DRAM)40(1)～40(R)(以後、DRAM40(1)～40(R))を備えてよい。この例の圧縮メモリコントローラ36は、本明細書で以下に開示する態様による、メモリ容量および/またはメモリ帯域幅の圧縮を利用する。以下の例示的な態様でより詳細に説明するように、圧縮メモリコントローラ36は、図1のCPUベースシステム12上で物理メモリサイズを増加する必要がないばかりでなく、CPU性能への影響を最小化するかまたは影響を及ぼさない様式で、CPUベースシステム12'がメモリ容量を効果的に増加することを可能にする。図1のCPUベースシステム12のメモリコントローラ24と同様に、図2のCPUベースシステム12'内の圧縮メモリコントローラ36は、内部システムバス22を介してCPUブロック14(1)

40

50

) ~ 14(N) によって共有される。

【 0 0 2 2 】

さらに、図2に関して、CPUブロック14(1) ~ 14(N)の各々の中のCPU16(1)および16(2)は、単一のハードウェアスレッドまたは複数のハードウェアスレッドを含んでよい。複数のハードウェアスレッドがCPU16(1)および16(2)の各々の中に設けられる場合、CPU16(1)および16(2)の待ち時間は、結果的に増加することがある。したがって、CPUブロック14(1) ~ 14(N)のCPU16(1)および16(2)の各々の中の複数のスレッドに対するスレッド情報を記憶するために、CPUベースシステム12'内のCPUブロック14(1) ~ 14(N)に対する共有キャッシュメモリの増加されたメモリ容量を設けることが望ましい。この点について、以下でより詳細に説明するように、CPUベースシステム12'はまた、すべてのCPUブロック14(1) ~ 14(N)に対する共有キャッシュメモリである圧縮キャッシュ42を含んでもよい。この例では、圧縮キャッシュ42は、L3キャッシュである。圧縮キャッシュ42を設けることによって、共有キャッシュのメモリ容量は、たとえば、図1のCPUベースシステム12内の共有L3キャッシュ20上の物理メモリサイズを増加する必要なしに、増加され得る。

10

【 0 0 2 3 】

したがって要約すれば、図2のCPUベースシステム12'内に圧縮メモリコントローラ36および/または圧縮キャッシュ42を設けることによって、CPUベースシステム12'のメモリ容量は、図1のCPUベースシステム12上の物理メモリサイズを増加する必要なしに、または最小限に必要な増加で、増加され得るばかりでなく、CPU性能への影響を最小化するかまたは影響を及ぼさない様式で増加され得る。本開示の残りの部分は、図2のCPUベースシステム12'内の圧縮メモリコントローラ36および/または圧縮キャッシュ42の例示的な態様および特徴を説明する。

20

【 0 0 2 4 】

この点について、図3は、図2の圧縮メモリコントローラ36の例示的な内部構成要素のより詳細な概略図を示す。この例では、圧縮メモリコントローラ36は、図2のCPUブロック14(1) ~ 14(N)を含む半導体ダイ46(1)、46(2)とは別の半導体ダイ44上に設けられる。代替として、圧縮メモリコントローラ36は、CPUブロック14(1) ~ 14(N)と共に半導体ダイ内に含まれてもよい。ダイ構成とは無関係に、CPUブロック14(1) ~ 14(N)が、内部システムバス22を介して圧縮メモリコントローラ36にメモリアクセス要求を行い、圧縮メモリコントローラ36を介してメモリからデータを受信することができるよう、圧縮メモリコントローラ36が設けられる。

30

【 0 0 2 5 】

引き続き図3を参照すると、圧縮メモリコントローラ36は、DRAM40(1)-40(R)として図2および図3に示すシステムメモリ38へのメモリアクセスに対する動作を制御する。圧縮メモリコントローラ36は、メモリアクセス要求をサービスするために使用される複数のDRAMインターフェース48(1) ~ 48(P)(たとえば、DDRインターフェース)を含む。この点について、この例の圧縮メモリコントローラ36は、圧縮コントローラ50を含む。圧縮コントローラ50は、図2のCPUブロック14(1) ~ 14(N)からのメモリアクセス要求に応答して、システムメモリ38に記憶されるデータを圧縮することと、システムメモリ38から取り出されたデータを展開することとを制御する。このようにして、CPUブロック14(1) ~ 14(N)は、圧縮メモリコントローラ36によってアクセスされるメモリの実際の容量より大きい仮想メモリアドレス空間を提供され得る。圧縮コントローラ50はまた、内部システムバス22上でCPUブロック14(1) ~ 14(N)に与えられる情報の帯域幅圧縮を実施するように構成され得る。

40

【 0 0 2 6 】

以下でより詳細に説明するように、圧縮コントローラ50は、メモリ容量および/または帯域幅の圧縮を提供するために、任意の数の圧縮技法およびアルゴリズムを実施することができる。ローカルメモリ52は、そのような圧縮技法およびアルゴリズムを実施するために圧縮コントローラ50によって必要とされるデータ構造および他の情報のために設けられる。この点について、この例のステティックランダムアクセスメモリ(SRAM)54の形態で設けられるローカルメモリ52はまた、圧縮技法およびアルゴリズムを実施するために圧縮コ

50

ントローラ50によって必要とされるデータ構造および他の情報を記憶するために、圧縮メモリコントローラ36内に設けられる。ローカルメモリ52は、圧縮技法およびアルゴリズムを実施するために圧縮コントローラ50に必要とされ得るデータ構造および他のデータ記憶に使用されるのに十分なサイズである。ローカルメモリ52はまた、圧縮メモリコントローラ36内で内部使用するための追加のキャッシュメモリを設けるために、L4キャッシュ(L4)などのキャッシュを含むように区分されてよい。したがって、L4コントローラ55もまた、L4キャッシュ(L4)へのアクセスを提供するために圧縮メモリコントローラ36内に設けられてもよい。強化された圧縮技法およびアルゴリズムは、以下でより詳細に説明するように、より大きい内部メモリを必要とすることがある。たとえば、ローカルメモリ52は、128キロバイト(kB)であることがある。

10

【0027】

さらに、図3に示し、以下でより詳細に説明するように、随意の追加の内部メモリ56もまた、圧縮メモリコントローラ36のために設けられ得る。追加の内部メモリ56は、一例として、DRAMとして設けられてもよい。以下でより詳細に説明するように、追加の内部メモリ56は、CPUベースシステム12'のメモリ容量および/またはメモリ帯域幅の圧縮を増加するためにメモリ圧縮および展開方式を提供する圧縮メモリコントローラ36のために、追加のまたはより大きい量の、データ構造およびローカルメモリ52内とは別のデータの記憶を容易にすることができます。内部メモリコントローラ58は、圧縮に使用するための追加の内部メモリ56へのメモリアクセスを制御するために、圧縮メモリコントローラ36内に設けられる。内部メモリコントローラ58は、CPUブロック14(1)～14(N)にアクセス可能でないかまたは可視でない。

20

【0028】

したがって、図3に関して要約すれば、ローカルメモリ52へのアクセスを有する圧縮メモリコントローラ36と、圧縮データ構造情報の記憶に使用するための追加の内部メモリ56とを設けることで、追加の物理メモリを設けるよりも低コストで増大したメモリ容量に対するメモリ容量圧縮が提供され得る。CPUベースシステム12'の物理メモリの増加を回避することで、追加の物理メモリを設けることと比較して、電力消費が節約され、パッケージ投影面積およびコストが低減され得る。圧縮メモリコントローラ36によってもたらされるメモリ容量および/または帯域幅の圧縮は、CPUブロック14(1)～14(N)の動作を制御するオペレーティングシステムに対してトランスペアレントである。

30

【0029】

さらに、図3の圧縮メモリコントローラ36はまた、メモリ帯域幅圧縮およびゼロライン圧縮を実施し得る。ローカルメモリ52は、そのような圧縮に使用されるより大きいデータ構造を記憶するために使用され得る。たとえば、帯域幅圧縮は、データが同様に圧縮されない場合に、より大きいデータ構造を記憶する圧縮メモリコントローラ36によって達成されてよい。たとえば、128バイトが転送されるべきである場合、メモリ帯域幅を低減するために、64バイトが一方の帯域幅圧縮方式において転送されてよいが、128バイトブロックのうちの他方の64バイトは未使用のままであり、したがって増大した圧縮データ構造がもたらされる。しかしながら、メモリ容量圧縮が同様に利用される場合、128バイトブロックのうちの残りの64バイトは、別のメモリブロックのために再使用されることがある。増大したメモリ帯域幅は、メモリアクセス待ち時間を低減し、待ち時間に影響を与えることなく、または待ち時間への影響を最小化しながら、同じ数のメモリチャネルにアクセスするために、より多くのCPU16(1)、16(2)もしくはそれらのそれぞれのスレッドを可能にすることができます。または、そのような圧縮が圧縮メモリコントローラ36によって実施されない場合に、より大きい数のメモリチャネルと比較して、同様の待ち時間の結果を達成しながら、メモリチャネルの数が低減され得、それによってシステムレベル電力が低減される。

40

【0030】

図3の圧縮メモリコントローラ36による使用のために追加の内部メモリ56を設けることによって、圧縮メモリコントローラ36によって圧縮を実施するために使用される、さらに

50

大量のデータ構造およびデータが、圧縮による待ち時間の影響をさらに低減するために利用可能になり得る。追加のCPU16(1)、16(2)および/またはそれらの中のハードウェアスレッドが、図3のCPUベースシステム12'に設けられてよく、メモリチャネルの圧縮および/または低減によって有効メモリ容量が増大する。圧縮メモリコントローラ36による圧縮を実施するために使用されるデータ構造およびデータのために、より多くのメモリが必要とされる場合、追加の内部メモリ56が、上記で説明したように設けられ得る。しかしながら、追加の内部メモリ56の増大したサイズが、強化された圧縮、ならびに/あるいはより多数のCPU16(1)、16(2)および/または収容されるべきCPU16当たりに収容されるハードウェアスレッドを可能にする可能性があるとはいえ、追加の内部メモリ56は、たとえば異なる半導体ダイ上など、圧縮メモリコントローラ36の外にあるので、追加の内部メモリ56へのメモリアクセス待ち時間は、ローカルメモリ52より大きくなることがある。

10

【0031】

図3の圧縮メモリコントローラ36内の、メモリ容量および/または帯域幅の圧縮のために設けられるリソースの各々は、ローカルメモリ52および追加の内部メモリ56を含めて、リソースと、面積、電力消費、メモリ容量圧縮によって増大するメモリ容量、およびメモリ帯域幅圧縮によって増大する性能との間で所望のバランスを達成するために、個別にまたは互いに連携して使用され得る。メモリ容量およびメモリ帯域幅の圧縮は、必要に応じて有効または無効にされ得る。さらに、圧縮メモリコントローラ36による使用のために上記で説明したリソースは、メモリ容量および/または帯域幅の圧縮効率、電力消費、ならびに性能の間で所望のトレードオフを達成するために、有効または無効にされ得る。圧縮メモリコントローラ36に対して利用可能なこれらのリソースを使用する例示的なメモリ容量および/または帯域幅の圧縮技法を、次に説明する。

20

【0032】

次の読み取りアドレスプリフェッティングを用いるメモリ容量および/または帯域幅の圧縮を利用するメモリコントローラを説明する前に、図2および図3の圧縮メモリコントローラ36によって利用され得る異なるタイプのメモリ圧縮方式のうちのいくつかの例を、図4～図21に関して最初に説明する。次の読み取りアドレスプリフェッティングを用いるメモリ容量および/または帯域幅の圧縮を利用するメモリコントローラ、ならびに関連するプロセッサベースのシステムおよび方法は、これらの異なるメモリ圧縮方式とともに利用され得る。

30

【0033】

この点について、図4は、図3の圧縮メモリコントローラ36によって実装され得る例示的な行ベースバッファメモリ容量圧縮方式60(以後、「行ベース圧縮方式60」)の概略図である。以下でより詳細に説明するように、行ベース圧縮方式60は、図2のCPUベースシステム12'内の利用可能な物理メモリより多くのメモリ容量をもたらすために、CPUオペレーティングシステム内で使用される物理アドレス(PA)に対して、仮想アドレス(VA)を超える新しいレベルの変換を追加する。アクセスされるメモリロケーションの物理アドレス(PA)は、物理バッファアドレス(PBA)に付加的に変換される。PBAは、圧縮メモリコントローラ36によってアクセス可能な圧縮メモリ(たとえば、システムメモリ38)へのアドレス(図4のPB ADDRESS)を含み、そこにおいて、CPU16によって圧縮メモリコントローラ36に与えられるPAに対する圧縮データが存在する。したがって、圧縮メモリコントローラ36内にPBAのこの追加の変換レイヤを設けることによって、システムメモリ38は、システムメモリ38内で通常、物理的にアクセス可能であるよりも多くのPAを効率的に記憶することができる。

40

【0034】

この点について、図4を参照すると、この行ベース圧縮方式60では、圧縮メモリコントローラ36は、システムメモリ38との間で圧縮データを記憶し、圧縮データにアクセスする。この例では、圧縮メモリコントローラ36は、メモリアクセス内でアドレス指定される各PAに対して128バイトメモリデータブロックに一度に(解凍された形態で)アクセスするように構成される。これは、この例におけるDRAMの形態のシステムメモリ38は、128バイトに一度にアクセスするときに効率的であるからである。代替として、メモリデータブロック

50

クは、たとえば64バイトなど、他のバイト長であってもよい。本例に戻ると、圧縮メモリコントローラ36は、圧縮メモリコントローラ36によってアクセス可能な物理メモリのサイズを超えて、圧縮メモリコントローラ36を用いてアドレス可能な有効メモリ容量を増加するために、システムメモリ38内の128バイトメモリデータブロックを圧縮する。

【0035】

引き続き図4を参照すると、すべての128バイトメモリデータブロックが、圧縮メモリコントローラ36によって同じ圧縮バイトサイズに圧縮され得るとは限らないことに留意されたい。メモリデータブロック内に含まれるデータの性質および利用される圧縮方式に応じて、いくつかのメモリデータブロックは、他のものより小さいサイズに圧縮されることがある。したがって、この例では、システムメモリ38は、PBAによってそれぞれアドレス可能な複数のビット長プール62(1)～62(Q)に分割される。プール62(1)～62(Q)はそれぞれ、物理バッファ(PB)内に圧縮された形態で128バイトデータブロックの圧縮データを記憶する。PBプール62(1)～62(Q)の各々は、圧縮後に未使用のビットストレージを回避するために、より小さいサイズに圧縮され得るメモリデータブロックが一緒にグループ化され得るように、異なるサイズの圧縮データを記憶するように設けられる。図4ではPBプール62(1)～62(Q)は連続的なPBにグループ化されたように示されるが、PBプール62(1)～62(Q)が連続的なPBを含む必要性はないことに留意されたい。以下でより詳細に説明するように、PBプール62(1)～62(Q)の各々のサイズは、圧縮メモリコントローラ36によって動的に割り当てられ得る。

10

【0036】

この例では、PBプール62(1)は、32バイトまでに圧縮されて記憶され得る圧縮メモリデータブロックを記憶するために使用される。PBプール62(2)は、64バイトまでに圧縮されて記憶され得る圧縮メモリデータブロックを記憶するために使用される。PBプール62(3)は、96バイトまで記憶され得る圧縮メモリデータブロックを記憶するために使用される。PBプール62(4)は、128バイトまで記憶され得る圧縮メモリデータブロックを記憶するために使用され、128バイトは、元の128バイト長以下に圧縮され得ない任意のメモリデータブロックを含むことになる。PBプール62(1)～62(Q)の各々は、圧縮メモリコントローラ36によって、PBプール62(1)～62(Q)に対して指定されたデータ圧縮サイズ内に圧縮されたPBを含む。以下でより詳細に説明するように、PBプール62(1)～62(Q)の各々はまた、書き込みトランザクションの間にフリーPBAにおける将来のPBを記憶するために圧縮メモリコントローラ36によって使用されるべき、そのそれぞれのPBプール62(1)～62(Q)内のフリーPBAのリストを含む、それぞれのフリーノードリスト64(1)～64(Q)(図4に64(1)だけを示す)を含む。

20

30

【0037】

与えられるPBプール62(「Q」)の数は、設計およびメモリデータブロックサイズに応じて変化し得ることに留意されたい。たとえば、メモリデータブロックサイズが256バイトの場合、8個のPBプール62が与えられてよく、各々は、圧縮メモリデータブロックを32バイトきざみで(たとえば、32バイト、64バイト、96バイト、...、256バイト)記憶するように構成される。メモリデータブロックサイズはより小さいバイトサイズに分割されるので、所与のメモリデータブロックサイズに対するPBプール62の数を増加することは、増加したメモリ圧縮比をもたらすことがあるが、同時に、より多数のPBプール62を管理するオーバーヘッドを増加することがある。さらに、メモリ設計におけるメモリデータブロックサイズを増加することは、改善されたメモリ圧縮比を達成することがあるが、同時に、所望のPBプール62(1)～62(Q)サイズを可能にするために、PBプール62(1)～62(Q)の各々のサイズを減少するか、またはPBプール62(1)～62(Q)の各々を設けるために、より多くのメモリを必要とすることがある。

40

【0038】

図4の行ベース圧縮方式60の初期化に際して、すべてのPBはフリーノードリスト64内に含まれ、どのPBも、未だPAに割り振られていないことを意味する。メモリトランザクション動作の間、圧縮メモリコントローラ36は、CPU16(図2)から受信されたPAに基づくメモリ

50

アクセスのためのPBにアクセスするために、システムメモリ38にアクセスする。圧縮メモリコントローラ36は、解凍データを内部システムバス22を介して要求するCPU16に与えるために、PBを展開することができる。代替として、圧縮メモリコントローラ36は、メモリ帯域幅圧縮を同様に実施している間、解凍された形態のPBを内部システムバス22を介して要求するCPU16に与えることができる。システムメモリ38内のPBは、PBAによってアクセス可能であるので、圧縮メモリコントローラ36は、メモリアクセス要求内のPAに対応する正しいPBにアクセスするために、メモリアクセス要求に対して受信されたPAのPBAへの変換を実施しなければならない。この点について、マスターディレクトリ66は、PAからPBAへの変換を記憶するシステムメモリ38内に設けられてよい。圧縮メモリコントローラ36は、PBの圧縮サイズに基づく書き込みトランザクションの間、PBが前にPBプール62(1)～62(Q)内で確立されて記憶されたときにPAに割り当てられた、対応するPBAを取得するために、マスターディレクトリ66内でルックアップを実施するために変換されたPAを使用する。PBAは、割り当てられたPBプール62(1)～62(Q)内の任意のPBと関連付けられてよく、隣接するPBが隣接するPBAに対応する必要ななしに、メモリアクセスが発生する任意のPBに圧縮データを記憶する柔軟性が与えられる。これは、より大きい圧縮比が達成されることを可能にする。

10

【0039】

引き続き図4を参照すると、圧縮メモリコントローラ36が、マスターディレクトリ66がアクセスされる必要ななしに、PAを最近変換されたPBAに、より速やかに変換することを可能にすることが望ましい。この点について一例では、ローカルメモリ52が、トランスレーションルックアサイドバッファ(TLB)68を提供するために、この例のn-ウェイアソシエイティブキャッシュとして構成されてよい。この点について、ローカルメモリ52内のTLB68は、n-ウェイキャッシュのように編成され得るSRAM54の形態で設けられる。TLB68は、それの中に含まれているTLBエントリ74の一部として、PAのタグ72を有するタグアレイ70を含む。データアレイ76は、各タグ72に対してマスターディレクトリ66からPBAのデータエントリおよび対応するアドレス長78をそれぞれ含む複数のTLBエントリ74を含むSRAM54内に設けられる。SRAM54内に設けられたn-ウェイアソシエイティブTLB68の幅は、TLBエントリ74の各々に記憶された所望の量のPBAならびにPBAの所望のアドレス長78を容易にするように設けられる。PBAのアドレス長78は、どの粒度(たとえば、ページ、バイトなど)に対してPBがPBプール62(1)～62(Q)内でアクセスされるかを決定する。

20

【0040】

この点について引き続き図4を参照すると、圧縮メモリコントローラ36は、メモリアクセスを実施するとき、SRAM54のタグアレイ70内でルックアップを実施するために変換されたPAを使用する。タグアレイ70内のTLBエントリ74が、変換されたPAと一致するタグ72を有する場合、TLBヒットがTLB68の中で発生する。したがって、マスターディレクトリ66内に含まれるPAに対応するPBAは、タグ72に対応するTLBエントリ74内のデータアレイ76内に含まれる。このシナリオでは、圧縮メモリコントローラ36は、PAをPBAに変換するためにマスターディレクトリ66にアクセスする必要はなく、したがって、システムメモリ38内のマスターディレクトリ66にアクセスすることはSRAM54にアクセスするよりも大きくなり得るので、関連する待ち時間が低減される。圧縮メモリコントローラ36は、システムメモリ38内のどのPBをアドレス指定するか、およびすべての圧縮メモリデータを記憶するためにPBに割り当てられたバイトをいくつ読み取るかを知るために、一致するTLBエントリ74内のPBAおよび対応するアドレス長78を使用することができる。トランザクションが読み取りトランザクションである場合、圧縮されたラインが、PBAロケーションにおいてシステムメモリ38から読み取られる。トランザクションが書き込みトランザクションである場合、圧縮データは、システムメモリ38のPB内のPBAにおいて記憶される。

30

【0041】

図4においてTLBミスがTLB68内で発生する場合、圧縮メモリコントローラ36は、マスターディレクトリ66内でルックアップを実施するためにPAを使用する。TLBエントリ74は、置き換え方針に従ってTLB68から立ち退かされる。PAに対応するマスターディレクトリ66

40

50

内のPBAは、TLB68内に置かれる。PAに対するマスター・ディレクトリ66から取得されたPBの開始位置およびアドレス長78は、PB内の圧縮データにアクセスするために圧縮メモリコントローラ36によって使用される。トランザクションが読み取りトランザクションである場合、圧縮されたラインが、PBA口케ーションにおいてシステムメモリ38から読み取られる。

【0042】

トランザクションが書き込みトランザクションである場合、圧縮データは、システムメモリ38のPB内において記憶される。PBAにおけるPBの長さが圧縮されたラインの圧縮サイズと同じである場合、圧縮されたラインは、システムメモリ38内のPBA口ケーションにおける同じPBに書き込まれ得る。PBAにおけるPBの長さが圧縮されたラインの圧縮サイズと異なる場合、圧縮メモリコントローラ36は、システムメモリ38に記憶されるべき圧縮されたラインを収容することになるシステムメモリ38に基づいて、別のPBプール62(1)～62(Q)内のフリーノードリスト64内でフリーPBを見つけることになる。フリーノードリスト64の動作について、以下の図7により詳細に説明する。システムメモリ38内に所望のPBサイズのPBプール62(1)～62(Q)に対するフリーノードリスト64内に利用可能なPBがない場合、圧縮メモリコントローラ36は、PBAにおけるPBプール62からの全ページを読み取り、対応するフリーノードリスト64をPBプール62からの全ページで補充し、TLB68内の書き込み動作に対する書き込みアドレスに対応するPAに対するPBAおよびPBA長を更新し、システムメモリ38内のPBAにおいて圧縮されたPBを書き戻す。圧縮後にPBプール62(1)～62(Q)に対するフリーノードリスト64内で利用可能なPBが、依然として十分にない場合、圧縮メモリコントローラ36は、より大きいPBを含む別のPBプール62(1)～62(Q)の別のフリーノードリスト64からより大きいPBを取り出し、元のPBを元のPBプール62(1)～62(Q)に対するフリーノードリスト64に戻すことができる。圧縮サイズが、より小さいサイズのPBが使用され得るようなものである場合、より小さいサイズのPBが、同様に、より小さいサイズのPBを含む別のPBプール62(1)～62(Q)のフリーノードリスト64から取得され得る。再び、元のPBは、元のPBプール62(1)～62(Q)に対するフリーノードリスト64に戻される。

10

20

30

40

【0043】

SRAM54内のTLBエントリ74のサイズは、たとえば、64バイト、128バイト、または256バイトなど、任意の所望のサイズであり得る。この例では、図4に示すように、圧縮メモリコントローラ36は、システムメモリ38からSRAM54に4つのPBAを一度にロードするように構成されるので、TLBエントリ74は、マスター・ディレクトリ66からの4つのPBAを含む。どのPBプール62(1)～62(4)にPBAが対応するかを示すために、図4に示すように、2ビットの符号化80が各PBA内に与えられる。次いで、圧縮メモリコントローラ36は、対応するPBプール62(1)～62(4)内のPBAの口ケーションにおける正しいPBにアクセスするために、PAに対応するPBAを使用する。

【0044】

図4のこの例では、すべて0である(すなわち、ゼロライン)PAに対して、特別な場合がメモリデータに対して与えられ得る。この場合、圧縮メモリコントローラ36は、特別な場合としてSRAM54内のPAに対して、PBAをPBAアドレス0として符号化することができ、PBが割り当てられていないことを意味する。圧縮メモリコントローラ36は、PBA内の0のPBアドレスを、PAに対応するメモリデータはすべて0であるものと認識する。したがって、すべて0であるメモリデータは、PBアドレス0におけるPBは圧縮メモリコントローラ36のために使用される必要はないので、事実上0バイトに圧縮される。したがって、PBプール62(1)～62(Q)は、すべて0であるメモリデータに対してアクセスされる必要はない。また、ゼロライン圧縮の場合にシステムメモリ38にアクセスする必要がない結果として、電力消費が低減される。

【0045】

特定のPBプール62(1)～62(Q)内の利用可能なPBが激減する場合、圧縮メモリコントローラ36は、別のPBプール62(1)～62(Q)内の他のPBAにPAを割り当てるよう構成され得る。この機能は、圧縮メモリコントローラ36によるバックグラウンドタスクの一部として提供されてよい。所与のPBプール62(1)～62(Q)内に残りの利用可能なPBがなくなる前に、別の

50

PBプール62(1)～62(Q)内の他のPBAにPAを割り当てることが望ましいので、激減は、特定のPBプール62(1)～62(Q)内の利用可能なPBがしきい値より下に低下することを意味する。所与のPBプール62(1)～62(Q)が激減した場合、他のPBプール62(1)～62(Q)内の他のPBが利用可能にされ得る。たとえば一例として、メモリデータが圧縮メモリコントローラ36によって32バイト以下に圧縮され得、32バイトPBプール62(1)内にもはや利用可能なPBがない場合、圧縮メモリコントローラ36は、PB分割動作において、64バイトPBプール62(2)内のPBをPBプール62(1)に対する2つの32バイトPBに再び割り当てることができる。しかしながら、一例として128バイトPBプール62(4)内にもはや利用可能なPBがない場合、圧縮メモリコントローラ36は、マージPB動作において、PBプール62(4)に対する1つの128バイトPBを形成するために、64バイトPBプール62(2)内の2つのPBを再び割り当てることができる。しかしながら、64バイトPBプール62(2)内の2つの64バイトPBは、この例ではマージPB動作において128バイトPBプール62(4)に再び割り当たられるために隣接していなければならない。

【0046】

行ベース圧縮方式60の1つの特徴は、PBがPBプール62(1)～62(Q)内で(たとえば、ページごとの基準に反して)ラインごとの基準でアクセスされるので、PBA内で与えられるより大きいPBアドレスが、SRAM54に記憶されることである。したがって、より大きいビット数が各TLBエントリ74内の各PBAに記憶されるので、SRAM54によって与えられるTLB68のサイズは、結果的に大きくなることがある。圧縮メモリコントローラ36によって実施されるメモリ容量圧縮においてPAからPBAへのアドレス変換をもたらすために、より小さいTLB68を有することが望ましい。

【0047】

この点について図5を参照すると、ページベースバッファメモリ容量圧縮方式82(以下、「ページベース圧縮方式82」)は、圧縮データを記憶してアクセスするために、図3の圧縮メモリコントローラ36によって実施されてよい。ページベース圧縮方式82は、図4の行ベース圧縮方式60と同様であり、図2のCPUベースシステム12'において提供されるリソースと同じものを使用することができる。したがって、本明細書で論じるページベース圧縮方式82は、図2および図3を参照することになる。以下でより詳細に説明するように、ページベース圧縮方式82を設けることは、SRAM54に記憶されるアドレスはバッファラインではなくメモリページ全体のTLB68アドレスとして働くので、図4の行ベース圧縮方式60より小さいTLB68サイズが、図3のSRAM54内にもたらされることを可能にする。したがって、SRAM54に記憶されるアドレスのサイズは、フルアドレス長である必要はない。しかしながら、ページベース圧縮方式82では、たとえば、図4の例における32、64、96および128バイトのより低いバイトサイズの、ラインサイズPBの割り振られたプールの代わりに、メモリページサイズ(たとえば、1kB、2kB、3kB、4kB)のデータページ(DP)のプールが、システムメモリ38(図3)内に割り振られる。同じく、図4の行ベース圧縮方式60と同様に、ページベース圧縮方式82が、システムメモリ38内のDPプールを維持する。

【0048】

図5を参照すると、ページベース圧縮方式82に対する単一のDPを含むシステムメモリ38が示されている。この例では、DPは2kBのサイズである。システムメモリ38内のDPは、この例では32バイトである等しいサイズの64個の物理バッファ(PB)に分割される。物理バッファ(PB)は、圧縮データがDP内で連続したPBを展開し得ることを除いて、上記の行ベース圧縮方式60におけるPBと同様である。ライン0は、図5の例示的なDP内で1つの32バイトのPBのみを占有する。ライン1は、図5の例示的なDP内で3つの連続する32バイトのPBを占有する。ライン31は、2つの連続する32バイトのPBを占有する。2つ以上のPBに及ぶラインは、常に、DP内の隣接するPBに記憶される。その中に含まれるDPおよびPBは、圧縮メモリコントローラ36によってPAから変換され得るデータページアドレス(DPA)によってアドレス可能である。図4のマスター・ディレクトリ66と同様に、DPを記憶するシステムメモリ38の一部は、システムメモリ38に記憶されるアクセスされた圧縮DPに対してPAをDPAに変換するためのマスター・ディレクトリのために使用される。

【0049】

10

20

30

40

50

図6は、直近に変換されたPAに対してより速やかに、PAをデータページアドレス(DPA)に変換するために、SRAM54内に設けられ得るTLB84を示す。TLB84は、キャッシュのようになび成される。TLB84内の各キャッシュライン86は、システムメモリ38内のDPを表す。PAは、有効なキャッシュライン86の各々の中のタグアレイ90内のタグ88に含まれる。DPAは、PAに対応するTLB84内のデータアレイ92内のキャッシュライン86に記憶される。DPAはシステムメモリ38内のページをアドレス指定する必要があるので、DPAは、PBのフルアドレスビットサイズより少ないビット内に記憶され得る。図6に示すように、管理情報94が、図5に示すDPAにおけるDPに対応するメモリページ内のPBに対応する、複数のライン位置(たとえば、L0 Pos、L1 Pos、...Ln Pos)および対応するライン長(たとえば、L0 Len、L1 Len、...Ln Len)を含む各DPAに対して設けられる。ライン位置はDP内のラインの開始を示し、ライン長はDP内のライン内に含まれるPBの数を示す。TLB84内のTLBミスに際して、TLB84は、DPAと関連付けられたキャッシュライン86をロードされ、キャッシュライン86は、図4の行ベース圧縮方式60におけるTLBエントリ74より小さい。

10

20

30

40

【0050】

図5および図6のページベース圧縮方式82はまた、図4の行ベース圧縮方式60におけるフリーノードリスト64と同様に、メモリページの圧縮に使用するためにシステムメモリ38内にフリーディスクを割り振るためにフリーノードリストを利用する。ページベース圧縮方式82においてメモリページの圧縮に使用するために、システムメモリ38内にフリーディスクを割り振るために使用され得る例示的なフリーノードリスト96を、図7に示す。そこに示されているように、フリーノードリスト96は、連結リストである。フリーノードリスト96内のDPAフリーリスト98(1)～98(R)の数は、すべてのDPに対するすべてのDPAをカバーするのに十分であるように与えられる。DPAフリーリスト98(1)～98(R)は、DPに等しいサイズを有するDPAに分離されるように示されているが、隣接するDPAが所与のDPに割り当てられる必要はない。DPAフリーリスト98(1)～98(R)の各々の中に、フリーディスク100(1)～100(R)のリストがある。新しいDPAが、DPのPBに圧縮データを記憶するために割り当てられることが必要であるとき、DPサイズに対応するフリーノードリスト96からの1つまたは複数のフリーディスク100(1)～100(R)が、ローカルブルバッファ102に引き入れられる。ヘッドポインタ104が、フリーノードリスト96をトラバースするためにヘッドフリーDPA100(1)内に与えられる。テールポインタ106は、DP内のDBにもはや割り当てられないDPAの解放(free-up)が、ローカルブルバッファ108からフリーノードリスト96のテールにブッシュされる可能にするように与えられる。図7のページベース圧縮方式82と同様の方式がまた、図4の行ベース圧縮方式60に対するフリーノードリスト64として使用され得る。ページベースメモリ圧縮方式82に対してフリーノードリスト96から引っ張られたDPAは、メモリページを同時に割り当てられてよく、一方、図4の行ベース圧縮方式60に対してフリーノードリスト64から引っ張られたDPAは、同時にバッファラインを割り当てられてよい。同じく、次のグループのDPのローカルブルバッファ102およびローカルブルバッファ108が維持されるので、これは、圧縮メモリコントローラ36が、たとえば効率性のために、フリーノードリスト96へのフリーディスクを指しているメインポインタを、DRAMからの単一のバーストで読み取るかまたは書き込むことを可能にする。

50

【0051】

図4の行ベース圧縮方式60または図5および図6のページベース圧縮方式82のいずれについて、その中に提供され得る別の特徴は、DB(行ベース圧縮方式60に対する)またはDP(ページベース圧縮方式82に対する)を記憶するために使用されるシステムメモリ38が、DBA(行ベース圧縮方式60に対する)またはDPA(ページベース圧縮方式82に対する)のフリーノードリスト64または96を記憶するために使用され得ることである。これは、フリーノードリスト64または96を記憶するためのメモリを温存する。図7のフリーノードリスト96を参照すると、ヘッドポインタ104は、圧縮メモリコントローラ36が、システムメモリ38を有するDP内に記憶されたDPフリーノードの開始を知ることを可能にする。テールポインタ106は、圧縮メモリコントローラ36が、システムメモリ38を有するDP内に記憶されたDPフリーノードの終端を知ることを可能にする。圧縮メモリコントローラ36が、割り振られるベ

50

き新しいDPAを要求するときに、割り振られたDPAが再び割り振られることのないように、フリーノードリスト96から新しく割り振られたDPAは、システムメモリ38内のフリーノードリスト96の終端に追加される。DPAポインタは、フリーノードリスト96内で開放されると、フリーノードリスト96に戻される。図4のフリーノードリスト64は、同様に、図4のシステムメモリ38のDB内に記憶され、その中で圧縮メモリコントローラ36によってアクセスされ得る。

【0052】

より小さいDPのみがシステムメモリ38内で利用可能であるとき、必要な場合に、より大きいDPを生成するためのDP管理の別な方法は、DPのペアを保持することである。たとえば、1kBのDPは、3kBのDPとペアを組むことができる。2kBのDPは、別の2kBのDPとペアを組むことができる。システムメモリ38内で、最大ページサイズ(たとえば、4kB)より小さい占有されたDPは、対応するPAに対するバックポインタを保持することができる。バックグラウンドタスクにおいて、DPのしきいの割り振りに到達すると、圧縮メモリコントローラ36は、フリーノードリスト96を走査して、同様にフリーDPペアを有するフリーDP100を探索することができる。フリーDP100およびそのペアは、システムメモリ38内により大きいDPをもたらすために組み合わされ得る。第2のしきい値に到達する場合、バックグラウンドタスクにおいて、圧縮メモリコントローラ36は、フリーノードリスト96を走査し、対応するDPペアの他方のDPの中を見て、それが圧縮データで占有されているかどうかを調べることができる。占有されている場合、圧縮メモリコントローラ36は、PAに対するテールポインタ106を得、占有されたDPを、同じDPサイズのフリーノードリスト96からの別のDPに移動させ、2つの隣接するDPを1つのより大きいDPにマージさせ、フリーノードリスト96を更新することができる。次いで、圧縮メモリコントローラ36は、占有されたDPに対する新しいDPAを指すために、PAに対するテールポインタ106を使用して、マスター・ディレクトリ66、および隨意にTLB68を更新することができる。

【0053】

図4に関して上記で説明した行ベース圧縮方式60と図5～図7に関して上記で説明したページベース圧縮方式82の両方の特性を含むメモリ容量圧縮方式を提供することが望ましい。この点について、図8は、図3の圧縮メモリコントローラ36によって実装され得る例的なハイブリッド行/ページベースバッファメモリ容量圧縮方式110の概略図である。以下で説明するように、ハイブリッド行/ページベースバッファメモリ容量圧縮方式110は、行ベースバッファメモリ容量圧縮方式とページベースバッファメモリ容量圧縮方式の両方の特徴を含む。ハイブリッド行/ページベースバッファメモリ容量圧縮方式110は、図4に示す行ベース圧縮方式60と共に構成要素を含み、それらは共通の要素番号で示される。

【0054】

この点について図8を参照すると、ハイブリッド行/ページベースバッファメモリ容量圧縮方式110において、圧縮メモリコントローラ36は、メモリアクセスにおいてアドレス指定される各PAに対して、(解凍された形態において)一度に128バイトのメモリデータブロックにアクセスするように構成される。システムメモリ38は、TLB68のデータアレイ76内のTLBエントリ74内のバッファポインタ114によってそれぞれアドレス可能である、同じサイズ(たとえば、1kB)の複数の物理ブロック(PB)112(1)～112(Q)に分割される。各PAは、システムメモリ38内の対応する複数のPB112(1)～112(Q)(たとえば、それぞれ1kB)にそれぞれ対応する複数のバッファポインタ114(1)～114(S)にマッピングするページアドレス(たとえば、4kB)である。したがって、PAページアドレスをいくつかのPB112(1)～112(Q)に分割することによって、各PB112は、行ベースメモリ容量圧縮方式と同様の利点を達成するために同じサイズであり得るが、データ圧縮は、依然として、ページベースメモリ容量圧縮方式のようにデータページサイズ基準(たとえば、4kB)で圧縮メモリコントローラ36によって取り扱われる。さらに、以下で説明するように、各PB112内のメモリデータブロックの各々は、同じサイズ(たとえば、32バイト)である。

【0055】

図9は、図8のハイブリッド行/ページベースバッファメモリ容量圧縮方式110に従って、

10

20

30

40

50

システムメモリ38内に設けられる複数の1kBのPB112(1)～112(4)から成る4kBサイズの例示的なデータページ(DP)の概略図である。その中に示すように、PB112(1)～112(4)の各々は、それぞれ同じサイズ(たとえば、この例ではそれぞれ32バイト)である複数のメモリデータブロック116を備える。所与のPAに対応するPB112(1)～112(4)が、システムメモリ38内に隣接して記憶される必要はないことに留意されたい。データページ(DP)は、同じサイズのPB112(1)～112(4)から成るので、PB112(1)～112(4)のサイズ決めは、圧縮メモリコントローラ36によって動的に割り当てられる必要はない。

【0056】

この例では、PB112(1)～112(4)の各々は、32バイトにまで圧縮されて記憶され得る圧縮メモリデータブロックを記憶するためのものである。圧縮メモリデータブロックが32バイトまで圧縮することができない場合、所与のPB112内の隣接するメモリデータブロック116が、圧縮データを記憶するために割り当てられ得る。TLB68内の各TLBエントリ74は、複数のバッファポインタ114(1)～114(4)を含むように構成され、それらは、この例では4つまでのバッファポインタ114(1)～114(4)である(たとえば、4kBのPAアドレスページサイズで、PB112(1)～112(4)がそれぞれ1kBである場合)。バッファポインタ114(1)～114(4)は、隣接するPB112(1)～112(4)を指す必要はない。所与のPAに対するTLBエントリ74内のバッファポインタ114(1)～114(4)は、必要に応じて、割り振られるときに特定のPB112(1)～112(4)を指すように圧縮メモリコントローラ36によって設定される。TLBエントリ74は、前に説明したようにマスター・ディレクトリ66から補充される。PB112が割り振られていない場合、タグ72に対するTLBエントリ74内の対応するバッファポインタ114がヌルとなるか、または何らかの他の情報が、バッファポインタ114に対応するPB112が、未だ圧縮メモリコントローラ36によって割り振られていないことを示すことになる。

10

20

30

30

【0057】

引き続き図9を参照すると、PAが圧縮メモリコントローラ36によって受信されると、PAがTLB68内のタグ72と比較され、PAに対応するTLBエントリ74をルックアップする。圧縮メモリコントローラ36は、PAのアドレスを使用して対応するDPに対するTLBエントリ74を参照し、対応するPB内のPAに対応するPB112(1)～112(4)のアドレスを決定する。たとえば、図9に示すように、PAが、TLB68内で選択されたTLBエントリ74内の第4のバッファポインタ114(4)に対応する場合、バッファポインタ114(4)は、システムメモリ38内のPB112(4)を指してよい。各TLBエントリ74に記憶され、圧縮データのページに対応するページ情報118が、PAに基づいて圧縮メモリコントローラ36によってさらにアクセスされる。この例ではデータページ当たり32のTLBエントリ74があるので、ページ情報118は、この例では32のエントリを含む。ページ情報118におけるキャッシュラインは、PAに従ってアクセスされるべきワードに基づいてアクセスされ、選択されたPB112(4)内の開始メモリデータブロック116と圧縮メモリデータの長さとを決定する。この例では、PAに対応する圧縮メモリデータは、(ビットをセーブするために)3の長さを意味する「2」の値を有する開始メモリデータブロック116B1#「96」に対応し、圧縮メモリデータはメモリデータブロック116B1#96内で開始する3バイトであることを意味する。

【0058】

図10は、メモリアクセスのために図8のハイブリッド行/ページベースバッファメモリ容量圧縮方式110に従って図9の割り振られたDP内の割り振られたPB112にPAを変換する、図3の圧縮メモリコントローラ36の例示的なプロセスを示すフローチャートである。その中に示すように、圧縮メモリコントローラ36は、入力データ120としてPAを受信する(ブロック122)。PAは、読み取りまたは書き込みのメモリトランザクションとしてオペコードを有する(ブロック122)。入力データ120はまた、メモリトランザクションが書き込みトランザクションである場合に圧縮長さを含んでもよい(ブロック122)。圧縮メモリコントローラ36は、開始ブロック番号および読み取りまたは書き込みトランザクションのオペコードを有するPB112にPAを変換するために、図8および図9に示すTLB68を参照する(ブロック124)。PAがTLB68内に含まれない場合、圧縮メモリコントローラ36は、新しく受信されたPAのスペースを作るために、TLB68内に存在するTLBエントリ74を立ち退かせる(ブロック126)。立ち退かさ

40

50

れたTLBエントリ74は、TLBエントリ74がダーティである場合にマスターディレクトリ66に戻って記憶される(ブロック126)。受信されたPAに対応するマスターディレクトリ66内のTLBエントリ74は、マスターディレクトリ66からフェッチされてTLB68内に置かれる(ブロック126)。

【0059】

引き続き図10を参照すると、PAがTLB68内に含まれるかまたはブロック126における立ち退きプロセスが完了した場合、圧縮メモリコントローラ36は、メモリアクセスが読み取り動作であるかまたは書き込み動作であるかを決定する(ブロック128)。メモリアクセスが読み取り動作である場合、圧縮メモリコントローラ36は、TLBエントリ74を使用して、TLB68内のバッファポインタ114およびページ情報118に対応するメモリデータブロック116にアクセスし、圧縮メモリデータを読み取ってプロセスは完了する(ブロック130)。メモリトランザクションが書き込み動作である場合、圧縮メモリコントローラ36は、入力データ120からの圧縮長が変化したかどうかを決定する(ブロック132)。圧縮長が変化しなかった場合、圧縮メモリコントローラ36は、TLBエントリ74を使用して、TLB68内のバッファポインタ114およびページ情報118に対応するメモリデータブロック116にアクセスし、圧縮メモリデータを書き込んでプロセスは完了する(ブロック130)。入力データ120からの圧縮長が変化して(ブロック132)増加した(ブロック134)ものと圧縮メモリコントローラ36が決定した場合、圧縮メモリコントローラ36は、PAに対応するシステムメモリ38内の現在のメモリデータブロック116をフリーにし、新しいメモリデータブロック116をPAに割り振り(ブロック136)、TLB68および対応するTLBエントリ74に対するダーティビットを更新する(ブロック138)。入力データ120からの圧縮長が増加しなかったものと圧縮メモリコントローラ36が決定した場合(ブロック134)、圧縮メモリコントローラ36は、PAに対応するシステムメモリ38内の余分のメモリデータブロック116をフリーにし(ブロック140)(または代替として、すべてのメモリデータブロック116をフリーにしてシステムメモリ38内に新しいメモリデータブロック116を再び割り振り)、PAに対応するページ情報118におけるメモリデータブロック116の長さを更新して、TLB68内のTLBエントリ74内にダーティビットを設定する(ブロック140)。

【0060】

同じく、受信されたPAをシステムメモリ38内に存在する圧縮PAに変換するための圧縮メモリコントローラ36の動作による性能損失を補償し得る追加のキャッシュを有する図2のSOC10'を設けることも望ましい。この点について、図11Aは、図2のSOC10'と同様の代替のSOC10(1)の一例を示すが、図11AのSOC10(1)は、この例ではキャッシュ142である随意のレポジトリを付加的に含む。キャッシュ142は、この例ではL4キャッシュである。圧縮メモリコントローラ36は、待ち時間を最小化するために、L4キャッシュ142と、TLB68またはマスターディレクトリ66(本明細書では圧縮PA(CPA)144と呼ぶ)の両方の中のPAを同時にロックアップすることができる。L4キャッシュ142内のアドレスは、解凍されているPAである。L4キャッシュ142内でPAがヒットすると、CPA144内のPAロックアップは余分になる。L4キャッシュ142内でPAがミスすると、変換された圧縮PAを取得するために、CPA144内のPAロックアップが必要となる。

【0061】

L4キャッシュ142とCPA144の両方にアクセスするCPU16の追加の待ち時間を回避するため、L4キャッシュ142およびCPA144はプライミングされてよい。この点について、図11Bは、共有されたL3キャッシュ42内でヒットまたはミスが決定される前にPAをロックアップするために、圧縮メモリコントローラ36内でL4キャッシュ142およびCPA144のプライミングを伴う、図11AのSOC10(1)の一例を示す。共有されたL3キャッシュ42が、読み取り動作のためにCPU16によってアクセスされるとき、L4キャッシュ142およびCPA144は、その中でPAの早期ロックアップを行うことによってプライミングされ得る。これらのロックアップの結果は、L4キャッシュ142と関連付けられた履歴バッファ146およびCPA144と関連付けられたCPA履歴バッファ148の中に記憶され得る。一例として、履歴バッファ146および/またはCPA履歴バッファ148は、小さい待ち時間を持つレジスタから構築され得る。CPA144内にミ

10

20

30

40

50

スがあると、随意のプリフェッヂが、マスターディレクトリ66内の圧縮メモリコントローラ36によって実施されてよい。CPA144のこのプライミングは、改善されたCPA144読み取りヒットレートをもたらし得る。共有されたL3キャッシュ42内でミスが発生すると、ヒットがL4キャッシュ142内で発生していたかどうかを決定するために、履歴バッファ146が検査される。ヒットがL4キャッシュ142内で発生していた場合、L4キャッシュ142からのデータは戻される。ミスがL4キャッシュ142内で発生していた場合、CPA履歴バッファ148が、次いで検査される。CPA履歴バッファ148の検査に基づいて、ヒットがCPA144内で発生する場合、圧縮PAがCPA144から直接読み取られてよく、さもなければ(すなわち、L4キャッシュ142がミスしてCPA144がミスする場合)、実行されるべき読み取り動作に対して圧縮PAを取得するために、マスターディレクトリ66が、圧縮メモリコントローラ36によって読み取られる。

10

【0062】

本明細書で開示する圧縮メモリコントローラ36によって提供され得るメモリ容量圧縮によって、物理的に存在するメモリ空間より大きいメモリアドレス空間が、オペレーティングシステム(OS)に対して利用可能になる。たとえば、OSは16Gbのメモリ空間を与えられるが、8Gbの物理システムメモリ38しか設けられないことがある。圧縮比が変化する(すなわち、悪くなる)につれて、システムメモリ38内のすべてのフリーDPが目減りすることがある。圧縮メモリコントローラ36がゲストOSからメモリを再請求し得る能力を提供することが望ましい。しかしながら、いくつかの基本的動作原理または規則が守られる必要がある。たとえば、第1の例示的な規則は、OSにすでに割り当てられているDPはOSから取り除かれ得ないことを規定することが望ましい。同じく、別の例示的な規則は、OSによるページの閲覧は、フルアドレスの非変換アドレス(たとえば、VAはVAであり、PAはPAである)として維持されることを規定することであってよい。言い換えれば、圧縮メモリコントローラ36と、システムメモリ38内のDPをもたらし得る、下層のメモリ断片化とによって実施される圧縮は、OSから隠され、圧縮メモリコントローラ36のハードウェアによって完全に処理される。さらに、別の例示的な規則として、OSによる書き込みは、常に成功しなければならず、システムメモリ38がVA空間より小さいときでも、OSのフルPA空間が書き込み可能でなければならないことを意味する。

20

【0063】

この点について、OSがこれらの規則に適合することを可能にするために、圧縮ドライバが、動的メモリサイジングを可能にするためにCPU16上で実行する各OSに組み込まれ得る。圧縮ドライバは、OSに対する拡張機能であり、OSのメモリ管理方式に影響を及ぼさない。システムメモリ38でフリー圧縮PBが底を突きそうになると、これによって、OSからページを要求するバルーンドライバへの圧力が増加する。これらの圧縮ドライバは、OSからより多くのメモリを要求する。圧縮ドライバがより多くのメモリを取得すると、圧縮ドライバはそれをピン止めし、そのページはディスクにスワップアウトされ得ず、したがってVAからPAへの変換マッピングがOSによって維持されなければならないことを意味する。次いで、圧縮ドライバは、高度に圧縮可能なパターンをメモリに書き込む。したがって、ハイパーバイザまたは変換サポートは必要とされない。上記で説明したDP管理方式が、必要とされるフリーDPにDPを再割り振りするために使用され得る。「緊急」バルーンは、バーストを吸収するために保持され得る。

30

【0064】

図12は、例示的な動的メモリサイジングの概略図である。たとえば、図1のSOC10内のCPU16上で作動することができるホストOS150が提供される。ホストOS150は、1つまたは複数のゲストOS152(1)～152(T)をサポートする。ホストOS150は、システムメモリ38へのインターフェースをゲストOS152(1)～152(T)に与える。図12の例では、システムメモリ38は、ホストOS150に割り振られ得る32GBをアドレス可能であってよいが、予期される2倍の圧縮比を使用して16GBのサイズであってよい。ハイパーバイザとしてホストOS150内で作動するメモリコントローラドライバ154は、バッファプール、ゲストOS152(1)～152(T)属性、サービス品質(QoS)要件、割り振られたメモリ(PA)のサイズ、およびメモリ障害をセット

40

50

アップすることによってハードウェアを初期化する。

【0065】

引き続き図12を参照すると、データ圧縮ドライバ(DCD)156(1)～156(T)が、ゲストOS152(1)～152(T)の各々の上に組み込まれる。圧縮メモリコントローラ36でシステムメモリ38内のフリー圧縮メモリバッファが底を突きそうになると、圧縮メモリコントローラ36は、圧力データ158の形態でゲストOS152(1)～152(T)に表示を与え、それが、データ圧縮ドライバ156(1)～156(T)に対する圧力を増加する。圧縮メモリコントローラ36は、フリーバッファ状態をモニタし、ゲストOS152(1)～152(T)の各々に圧力データ158を与える。これは、データ圧縮ドライバ156(1)～156(T)の各々に割り当てられたメモリマップレジスタであり得る。バッファのフリープールがシステムメモリ38内で激減していることを圧縮メモリコントローラ36が検出すると、圧縮メモリコントローラ36は、属性に応じてゲストOS152(1)～152(T)の各々に対する圧力レジスタの値を増加することができる。

10

【0066】

それに応じて、ゲストOS152(1)～152(T)内のデータ圧縮ドライバ156(1)～156(T)は、ホストOS150からより多くのメモリを要求し、それが、システムメモリ38を使用する他のソースからメモリをフリーにするようにホストOS150に圧力をかける。たとえば、ホストOS150は、メモリをフリーにするために、ページキャッシングおよびプロセスヒープを使用してよい。データ圧縮ドライバ156(1)～156(T)は、一例として圧力レジスタ値に比例するレベルで、そのそれぞれのゲストOS152(1)～152(T)からページを要求するように構成されてよい。ゲストOS152(1)～152(T)はページを提供し、必要な場合、ページキャッシュなど、他のプールからページを解放する(free up)ことができる。ゲストOS152(1)～152(T)はまた、使用されたページをヒープからページアウト等することができる。

20

【0067】

データ圧縮ドライバ156(1)～156(T)がシステムメモリ38からより多くのメモリを取得すると、データ圧縮ドライバはそれをピン止めし、次いで高度に圧縮可能なパターンをシステムメモリ38に書き込むことができる。この点について、データ圧縮ドライバ156(1)～156(T)は、ゼロ(0)をページに書き込み、圧縮メモリコントローラ36は、下層のバッファを効率的にフリーにしてフリープールに戻すことができる。したがって、ハイパーテザまたは変換サポートは必要とされない。上記で説明したDP管理方式が、必要とされるDPにDPを再割り振りするために使用され得る。「緊急」バルーンは、バーストを吸収するために保持され得る。

30

【0068】

上記で説明したように、圧縮メモリコントローラ36はまた、メモリ帯域幅圧縮方式を利用することができる。たとえば、内部システムバス22上の帯域幅を低減するために、内部システムバス22などのメモリバス上の128バイトデータブロックを圧縮することが望ましい。帯域幅を低減することで、同様に、バス待ち時間を低減し、電力消費を低減することができる。CPU16、および/またはCPU16内で設けられるハードウェアスレッドの数は増加し得、CPU16の性能に影響を及ぼすこともない。または代替として、所与のバス帯域幅に対するメモリチャネルの数を低減するためのオプションが提供され得る。

40

【0069】

この点について、図13は、メモリ帯域幅圧縮を提供するための、図3の圧縮メモリコントローラ36によって実装され得る例示的なメモリ帯域幅圧縮方式160の概略図である。メモリ容量圧縮方式のこの例では、データは、圧縮形態または解凍形態のいずれかで、PA口ケーションにおいてシステムメモリ38に記憶される。この点について、複数のデータライン162から成るシステムメモリ38が設けられる。データライン162は、受信されたPAによって、圧縮メモリコントローラ36によってアクセス可能である。ECC圧縮インジケータ166が、データライン162が圧縮形態で記憶されているか否かを示すために、データライン162の各々と関連して記憶される。このようにして、システムメモリ38に対するメモリアクセス要求を実施するとき、圧縮メモリコントローラ36は、データライン162がメモリアクセス要求のプロセスの一部として圧縮されているかどうかを決定するために、アドレスされる

50

べきPAに対応するデータライン162と関連付けられECC圧縮インジケータ166を検査することができる。

【0070】

同じく、PAに対応するシステムメモリ38内でデータライン162当たり1つのエントリ165を含むマスター・ディレクトリ66が依然として設けられる。マスター・ディレクトリ66はまた、データライン162がデータライン162内に圧縮されて記憶されているかどうかを示し、圧縮されて記憶されているならば、サポートされる複数の圧縮長がある場合にその圧縮長を示すために、エントリ165当たり1つの圧縮インジケータ166を含む。たとえば、データライン162が128バイト長であり、その中に記憶されるデータが64バイト以下に圧縮され得る場合、システムメモリ38に記憶されるデータに対応するマスター・ディレクトリ66内の圧縮インジケータ166は、そのデータが、128バイトデータライン162の第1の64バイト内に記憶されていることを示すように設定されてよい。

10

【0071】

引き続き図13を参照すると、書き込み動作の間、圧縮メモリコントローラ36は、システムメモリ38の中に書き込まれるべきメモリブロックを圧縮することができる。たとえば、データ(たとえば、128バイト、または256バイト)が圧縮される。データブロックが、システムメモリ38のバーストサイズ(たとえば、64バイト)以下である場合、64バイトが書き込まれ得、さもなければ、128バイトが書き込まれる。256バイトは、圧縮データサイズに応じて、64、128、192または256バイトとして書き込まれ得る。システムメモリ38内のデータライン162と関連付けられたECC圧縮インジケータ166はまた、データライン162におけるデータが圧縮されているか否かを示すように設定され得る。

20

【0072】

たとえば読み取り動作の間、圧縮メモリコントローラ36は、システムメモリ38からアクセスされるべきデータの一部を読み取ることができる。読み取られるべきデータが、ECC圧縮インジケータ166によって示されるようにシステムメモリ38内で圧縮された場合、読み取られるデータの部分は全データブロックであるが圧縮された形態であるので、読み取り動作が行われる。圧縮データは、帯域幅圧縮を達成するために圧縮された形態で内部システムバス22(図2)上に圧縮メモリコントローラ36によって供給され得る。読み取られるデータの部分がシステムメモリ38内で圧縮されていなかった場合、読み取られるべきメモリラインの追加の部分もまた、システムメモリ38から読み取られなければならないので、これは待ち時間の影響を有する。トレーニング機構が、いくつかのアドレス範囲に対して配置され得、システムは、システムメモリ38から2アクセスでデータを読み取ることがベターであるかどうか、または待ち時間の影響を回避するためにシステムメモリ38から全データ量を読み取ることがベターであるかどうかを「学習する」。

30

【0073】

この例では、キャッシュエントリ170を含む圧縮インジケータ(CI)キャッシュ168もまた、設けられてよい。CIキャッシュ168は、システムメモリ38の外部の別々のキャッシュ内に設けられる。CIキャッシュ168は、システムメモリ38内のデータライン162が圧縮形態で記憶されているか否かを示すために、システムメモリ38内にデータライン162当たり1つのキャッシュエントリ170を含むキャッシュである。このようにして、システムメモリ38に対するメモリアクセス要求を実施するとき、圧縮メモリコントローラ36は、データライン162を読み取る必要なしに、システムメモリ38内のPAにおけるデータライン162が、メモリアクセス要求のプロセスの一部として圧縮されているかどうかを決定するために、アドレスされるべきPAに対応するCIキャッシュ168内のキャッシュエントリ170を最初に検査することができる。したがって、データライン162が圧縮されて記憶されていることをCIキャッシュ168が示す場合、圧縮メモリコントローラ36は、全データライン162を読み出す必要はなく、したがって待ち時間が低減される。データライン162が解凍されて記憶されていることをCIキャッシュ168が示す場合、圧縮メモリコントローラ36は、全データライン162を読み出すことができる。CIキャッシュ168内でミスが発生する場合、マスター・ディレクトリ66内のアクセスされたデータライン162に対応する当該のECC圧縮インジケータ166は

40

50

、参照され、同じPAに対する後続のメモリアクセス要求のためにCIキャッシュ168にロードされ得る。

【0074】

たとえば、CIキャッシュ168は、レギュラーキャッシュとして編成されてよい。CIキャッシュ168は、タグアレイを含んでよく、n-ウェイ、または他のウェイで編成されてもよい。CIキャッシュ168は、圧縮メモリコントローラ36によって実装され得る立ち退きポリシーを有することになる。CIキャッシュ168内のキャッシュライン172の各々は、複数のキャッシュエントリ170を記憶することができる。キャッシュエントリ170の各々は、キャッシュエントリ170と関連付けられたシステムメモリ38内のデータライン162が、圧縮されているかどうか、および/またはキャッシュエントリ170に対応するデータの圧縮サイズに対するオプション(たとえば、データが32、64、96または128バイトに圧縮され得る場合に4つのオプションに対して2つのキャッシュインジケータ)を表すかどうかを示すために、1つの圧縮インジケータ174を含む。この例では、この情報はまた、キャッシュエントリ170内の圧縮インジケータ174に記憶されるので、ECC圧縮インジケータ166は冗長であることに留意されたい。たとえば、データライン162が128バイト長であり、その中に記憶されるデータが64バイト以下に圧縮され得る場合、システムメモリ38内のデータライン162に対応するCIキャッシュ168内のキャッシュエントリ170内の圧縮インジケータ174は、そのデータが、128バイトデータライン162の第1の64バイトに記憶されていることを示すように設定されてよい。

【0075】

CIキャッシュ168が利用されるときの読み取り動作の別の例では、圧縮メモリコントローラ36は、データライン162内のいくつかの128バイトメモリブロックに対して、データブロックが圧縮されているかどうかを覚えているCIキャッシュ168を維持する(たとえば、128バイトブロック当たり1ビット、または256バイトブロック当たり2ビット)。CIキャッシュ168は、この例では、4Kページ当たり、1つのページアドレスおよび32ビットのフラグに対するエントリをアドレスされ得る。読み取り動作に対して、CIキャッシュ168は、アクセスされるべきアドレスがCIキャッシュ168内に含まれるかどうかを調べるために検査される。要求されたデータがCIキャッシュ168内にある場合、圧縮メモリコントローラ36は、CIキャッシュ168内のキャッシュエントリ170に従ってシステムメモリ38からのデータ量を読み取ることができる。要求されたデータがCIキャッシュ168内にない場合、2つの例示的なオプションが存在する。最悪の場合のシナリオを仮定すると、圧縮メモリコントローラ36は、フルキャッシュエントリ170サイズ(たとえば、128バイト)を読み取ることができる。代替として、圧縮メモリコントローラ36は、上記で説明した2-アクセス読み取りプロセスに従って読み取ることができる。ラーニング拡張も利用され得る。読み取られるべきデータがCIキャッシュ168内にない場合、CIキャッシュ168は、次のように更新され得る。キャッシュエントリ170が、知られている技法(たとえば、LRU、擬似LRU、など)に従ってCIキャッシュ168から立ち退かされ得る。立ち退かされたキャッシュエントリ170は、メモリに書き込まれる必要はなく、単にCIキャッシュ168から除去され得ることに留意されたい。新しいキャッシュエントリ170が生成されてよく、そこにおいて、すべてのキャッシュエントリ170は、システムメモリ38から単に読み取られただけのブロックを除いて、最大読み取りサイズに設定され、それは、その実際のサイズに設定される。最適化として、この立ち退きは、システムメモリ38から読み取られたブロックが解凍されている(すなわち、その最大サイズ)場合はスキップされ得る。

【0076】

引き続き図13を参照すると、設けられたCIキャッシュ168を用いてシステムメモリ38内のデータライン162に書き込むときに、キャッシュエントリ170がシステムメモリ38内で圧縮されたか否かを示すためのECC圧縮インジケータ166を用いて、圧縮メモリコントローラ36は、常に、圧縮されたサイズを書き込むように構成され得る。圧縮インジケータ174が、データライン162と関連付けられたCIキャッシュ168内のキャッシュエントリ170内に存在する場合、圧縮インジケータ174が、CIキャッシュ168内で更新される。圧縮インジケー

10

20

30

40

50

タ174がデータライン162と関連付けられたキャッシュエントリ170に対するCIキャッシュ168内に存在しない場合、CIキャッシュ168が更新されてよい。圧縮長がビンサイズ(たとえば、64バイト)より大きい場合、CIキャッシュ168が更新される。1つのキャッシュエントリ170が立ち退かされて、新しいキャッシュライン172が、マスター・ディレクトリ66からCIキャッシュ168によって組み込まれる。対応するデータライン162に対する圧縮インジケータ174は、CIキャッシュ168内で更新される。圧縮長がビンサイズ以下である場合、CIキャッシュ168は、更新される必要はないが、更新されてもよい。たとえば、CIキャッシュ168は、一定の確率(それは、100%、すなわち常時、に構成され得る)で更新されてよい。この確率が100%に設定される場合、マスター・ディレクトリ66は、いくつかの場合には、常に正確であり得る。さもなければ、マスター・ディレクトリ66は、実際の圧縮ライン長より大きい圧縮ライン長を表すことになる。これは、マスター・ディレクトリ66の更新における帯域幅と、メモリ読み取り中にセーブする帯域幅との間のトレードオフである。

10

【0077】

引き続き図13を参照すると、システムメモリ38から読み取るときに、ECC圧縮インジケータ166が、アクセスされたデータライン162に対するCIキャッシュ168内のキャッシュエントリ170内に存在する場合、これは、このデータライン162に対する圧縮サイズの情報が知られていることを意味する。この点について、キャッシュライン172は、読み取りサイズを決定するためにCIキャッシュ168からアクセスされ、そのデータ量が、対応するデータライン162から読み取られ得る。圧縮インジケータ174が、CIキャッシュ168内のアクセスされたキャッシュライン172内に存在しない場合、圧縮メモリコントローラ36は、システムメモリ38からフルラインサイズを読み取り、データが圧縮されているか否かを決定するためにECC圧縮インジケータ166を使用する。同時に、圧縮メモリコントローラ36は、設定可能な確率(たとえば、10%)でマスター・ディレクトリ66からCIキャッシュ168をリフレッシュしてよい。

20

【0078】

代替として、圧縮メモリコントローラ36は、システムメモリ38内でECC圧縮インジケータ166を利用しないように構成され得る。この例では、マスター・ディレクトリ66は、システムメモリ38内でラインが圧縮されているか否かを、常に反映することになる。システムメモリ38に対する書き込みトランザクションのときに、圧縮データは、常に、圧縮メモリコントローラ36によって書き込まれることになる。圧縮メモリコントローラ36は、CIキャッシュ168内の圧縮インジケータ174を常に更新することになり、そのことは、必要であれば、立ち退きとマスター・ディレクトリ66とがアクセスされることを引き起こす。システムメモリ38に対する読み取りトランザクションのときに、圧縮メモリコントローラ36は、常に、CIキャッシュ168内の圧縮インジケータ174から最初に読み取るように構成され、存在しない場合、キャッシュライン172はCIキャッシュ168から立ち退かされ、CIキャッシュ168がアクセスされることになる。次いで、ラインが、ECC圧縮インジケータ166内で指定された長さに基づいてシステムメモリ38から読み取られる。

30

【0079】

読み取り動作の別の例では、圧縮メモリコントローラ36は、いくつかの128バイトメモリブロックに対して、ブロックが圧縮されているかどうかを覚えているCIキャッシュ168を維持することができる(たとえば、128バイトブロック当たり1ビット、または256バイトブロック当たり2ビット)。CIキャッシュ168は、4Kページ当たり、1つのページアドレスおよび32ビットのフラグに対するエントリをアドレスされ得る。読み取り動作に対して、CIキャッシュ168は、アクセスされるべきアドレスがCIキャッシュ168内に含まれるかどうかを調べるために検査される。要求されたデータがCIキャッシュ168内にある場合、圧縮メモリコントローラ36は、CIキャッシュ168内のキャッシュエントリ170に従ってシステムメモリ38からのデータ量を読み取ることができる。要求されたデータがCIキャッシュ168内にない場合、2つの例示的なオプションが存在する。最悪の場合のシナリオを仮定すると、圧縮メモリコントローラ36は、フルキャッシュエントリ170サイズ(たとえば、128バイト)を読み取ることができる。代替として、圧縮メモリコントローラ36は、上記で説明した2-ア

40

50

クセス読み取りプロセスに従って読み取ることができる。ラーニング拡張も利用され得る。読み取られるべきデータがCIキャッシュ168内にない場合、CIキャッシュ168は、次のように更新され得る。キャッシュエントリ170が、知られている技法(たとえば、LRU、擬似LRU、など)に従ってCIキャッシュ168から立ち退かされ得る。立ち退かされたキャッシュエントリ170は、メモリに書き込まれる必要はなく、単にCIキャッシュ168から除去され得ることに留意されたい。新しいキャッシュエントリ170が生成されてよく、そこにおいて、すべてのキャッシュエントリ170は、システムメモリ38から単に読み取られただけのブロックを除いて、最大読み取りサイズに設定され、それは、その実際のサイズに設定される。最適化として、この立ち退きは、システムメモリ38から読み取られたブロックが解凍されている(すなわち、その最大サイズ)場合はスキップされ得る。

10

【0080】

引き続き図13を参照すると、システムメモリ38に書き込むときに、キャッシュエントリ170がシステムメモリ38内で圧縮されたか否かを示すためのECC圧縮インジケータ166を用いて、圧縮メモリコントローラ36は、常に、圧縮されたサイズを書き込むように構成され得る。圧縮インジケータ174が、アクセスされたキャッシュエントリ170に対するCIキャッシュ168内に存在する場合、圧縮インジケータ174は、CIキャッシュ168内で更新される。圧縮インジケータ174がCIキャッシュ168内に存在しない場合、CIキャッシュ168は更新されてよい。圧縮長がビンサイズ(たとえば、64バイト)より大きい場合、CIキャッシュ168が更新される。1つのキャッシュエントリ170が立ち退かされて、新しいキャッシュライン172が、マスター・ディレクトリ66によって組み込まれる。対応するデータライン162に対する圧縮インジケータ174は、CIキャッシュ168内で更新される。圧縮長がビンサイズ以下である場合、CIキャッシュ168は、更新される必要はないが、更新されてもよい。たとえば、CIキャッシュ168は、一定の確率(それは100%、すなわち常時構成され得る)で更新されてよい。この確率が100%に設定される場合、マスター・ディレクトリ66は、いくつかの場合には、常に正確であり得る。さもなければ、マスター・ディレクトリ66は、実際の圧縮ライン長より大きい圧縮ライン長を表すことになる。これは、マスター・ディレクトリ66の更新における帯域幅と、メモリ読み取り中にセーブする帯域幅との間のトレードオフである。

20

【0081】

引き続き図13を参照すると、システムメモリ38から読み取るときに、ECC圧縮インジケータ166が、アクセスされたデータライン162に対するCIキャッシュ168内に存在する場合、これは、このデータライン162に対する圧縮サイズの情報が知られていることを意味する。この点について、キャッシュライン172は、読み取りサイズを決定するためにCIキャッシュ168からアクセスされ、そのデータ量が、対応するデータライン162から読み取られ得る。圧縮インジケータ174が、CIキャッシュ168内のアクセスされたキャッシュライン172内に存在しない場合、圧縮メモリコントローラ36は、システムメモリ38からフルラインサイズを読み取り、データが圧縮されているか否かを決定するためにECC圧縮インジケータ166を使用する。同時に、圧縮メモリコントローラ36は、設定可能な確率(たとえば、10%)でマスター・ディレクトリ66からCIキャッシュ168をリフレッシュしてよい。

30

【0082】

代替として、圧縮メモリコントローラ36は、システムメモリ38内でECC圧縮インジケータ166を利用しないように構成され得る。この例では、マスター・ディレクトリ66は、システムメモリ38内でラインが圧縮されているか否かを、常に反映することになる。システムメモリ38に対する書き込みトランザクションのときに、圧縮データは、常に、圧縮メモリコントローラ36によって書き込まれることになる。圧縮メモリコントローラ36は、CIキャッシュ168内の圧縮インジケータ174を常に更新することになり、そのことは、必要であれば、立ち退きとマスター・ディレクトリ66とがアクセスされることを引き起こす。システムメモリ38に対する読み取りトランザクションのときに、圧縮メモリコントローラ36は、常に、CIキャッシュ168内の圧縮インジケータ174から最初に読み取るように構成され、存在しない場合、キャッシュライン172はCIキャッシュ168から立ち退かされ、マスター・ディレクトリ66がアクセスされることになる。次いで、ラインが、ECC圧縮インジケータ166内で指定

40

50

された長さに基づいてシステムメモリ38から読み取られる。

【0083】

同じく、メモリ帯域幅圧縮方式のために追加のキャッシュを設けることが望ましい。この点について、図14は、図2のSOC10'のような代替のSOC10(2)の一例を示すが、図14のSOC10(2)は、この例ではL4キャッシュである随意のキャッシュ142を追加で含む。圧縮メモリコントローラ36は、待ち時間を最小化するために、L4キャッシュ142とCIキャッシュ168の両方の中のPAを同時にルックアップすることができる。L4キャッシュ142内のアドレスは、解凍されているPAである。L4キャッシュ142内でPAがヒットすると、CIキャッシュ168内のPAルックアップは余分になる。L4キャッシュ142内でPAがミスすると、システムメモリ38からデータを取得するために、CIキャッシュ168内のPAルックアップが必要となる。同じく、L4キャッシュ142とCIキャッシュ168の両方にアクセスするCPU16の追加の待ち時間を回避するために、L4キャッシュ142およびCIキャッシュ168は、図11Bに関して上記で説明したプライミングと同様にプライミングされてよい。

10

【0084】

上記で説明したように、圧縮を最適化するために、特別な場合としてすべて0のメモリデータブロックの圧縮をもたらすことが望ましい。たとえば、メモリデータブロック(たとえば、128バイト)がすべて0である場合、データは読み取りまたは書き込みを行われないが、別個のデータ構造においてすべて0として示される。メモリラインが圧縮されているかどうかを示すために、ゼロビットがシステムメモリ38内に設けられてよく、したがって、メモリデータブロック当たり1ビットが追加される。圧縮メモリコントローラ36は、たとえばいくつかの128バイトメモリブロックに対して、ブロックがすべて0であるかどうかを覚えているキャッシュを維持することができる。書き込み動作に対して、メモリデータブロックの1つのラインがすべて0である場合、圧縮メモリコントローラ36は、そのラインをシステムメモリ38に書き込まないが、CIキャッシュ168は更新される。読み取り動作に対して、CIキャッシュ168内のゼロビットが、ラインに対して検査される。ラインがCIキャッシュ168内にある場合、ゼロビットに従って、ラインがシステムメモリ38から読み取られるか、またはすべての0が戻される。ラインがCIキャッシュ168内にない場合、それは、システムメモリ38から読み取られる。ラインがゼロである場合、CIキャッシュ168は更新され得る。CIキャッシュ168は、知られている技法(LRU、擬似LRU、など)に従って、TLBエントリ74を立ち退かせるために更新され得る。ラインが立ち退かされる場合、ゼロビットを設定されたすべてのラインは、メインメモリに書き込まれる必要がある。これは、オールゼロ書き込み(write-all-zeros)キューとして編成され得る。可能な最適化は、2ビットを使用することであり、1つのビットは、ラインがすべてゼロであるかどうかを示し、別のビットは、ラインがダーティである(すなわち、未だメインメモリに書き込まれていない)かどうかを示す。圧縮メモリコントローラ36のバックグラウンドタスクは、CIキャッシュ168を精査し、ダーティビットを有するラインを「オールゼロ書き込み」キュー内にキューイングすることができる。上記で説明したメモリ容量圧縮方式とともに、異なるメモリ圧縮方式が、必要に応じて圧縮メモリコントローラ36によって利用され得る。たとえば、64、128および256バイトなど、小さいデータブロックに対して最適化されるメモリ圧縮方式を利用することが望ましく、その理由は、上記の例で説明したキャッシュラインは、例としてこれらのサイズを含むからである。

20

【0085】

この点について、図15は、頻出パターン圧縮データ圧縮方式180を示す。この点について、圧縮されるべきソースデータフォーマット182内のソースデータは、本例では128バイトとして示されている。圧縮データフォーマット184を、以下で示す。圧縮データフォーマット184は、プレフィックスコードPxおよびプレフィックスの後ろのデータDataxのフォーマットにおいて与えられる。プレフィックスは3ビットである。プレフィックスコードは、プレフィックスコード列186内の所与のプレフィックスコードに対するパターン符号化列190内の符号化されたパターンを示す頻出パターン符号化表188内のプレフィックスコード列186内に示される。符号化されたパターンに対するデータサイズは、頻出パターン

30

40

50

符号化表188のデータサイズ列192内に与えられる。

【0086】

図16は、32ビット頻出パターン圧縮データ圧縮方式194を示す。この点について、圧縮されるべきソースデータフォーマット196内のソースデータは、本例では128バイトとして示されている。圧縮データフォーマット198を、以下で示す。圧縮データフォーマット198は、プレフィックスPxおよびプレフィックスの直後のデータDataxのフォーマットにおいて与えられる。新しい圧縮データフォーマット200は、プレフィックスコードPx、データDatax、フラグ、およびパターンの異なるフォーマットにおいて与えられ、それらは、効率を目的として一緒にグループ化されるように編成される。プレフィックスコードは、3ビットである。プレフィックスコードは、プレフィックスコード列202内の所与のプレフィックスコードに対するパターン符号化列206内の符号化されたパターンを示す頻出パターン符号化表204内のプレフィックスコード列202内に示される。符号化されたパターンに対するデータサイズは、頻出パターン符号化表204のデータサイズ列208内に与えられる。プレフィックスコード000は、解凍パターンを示し、それは、新しい圧縮データフォーマット200内の32ビットのフルサイズのデータである。プレフィックスコード001は、すべてゼロのデータブロックを示し、それは、新しい圧縮データフォーマット200のデータ中の0ビットとして与えられ得る。3ビットプレフィックスに対して、プレフィックスコード010～111は、ソースデータ内で認識される他の特定のパターンを符号化するために使用されてよく、本例では、それはそれぞれ、0、4、8、12、16および24ビットのパターンである。

【0087】

図17は、32ビット頻出パターン圧縮データ圧縮方式210の一例を示す。この点について、圧縮されるべきソースデータフォーマット212内のソースデータは、本例では128バイトとして示されている。圧縮データフォーマット214を、以下で示す。圧縮データフォーマット214は、プレフィックスPxおよびプレフィックスの後のデータDataxのフォーマットにおいて与えられる。新しい圧縮データフォーマット216は、プレフィックスコードPx、データDatax、フラグ、およびパターンの異なるフォーマットにおいて与えられ、それらは、効率を目的として一緒にグループ化されるように編成される。プレフィックスコードは、3ビットである。プレフィックスコードは、プレフィックスコード列218内の所与のプレフィックスコードに対するパターン符号化列222内の符号化されたパターンを示す頻出パターン符号化表220内のプレフィックスコード列218内に示される。符号化されたパターンに対するデータサイズは、頻出パターン符号化表220のデータサイズ列224内に与えられる。プレフィックスコード000は、解凍パターンを示し、それは、新しい圧縮データフォーマット216内の32ビットのフルサイズのデータである。プレフィックスコード001は、すべてゼロのデータブロックを示し、それは、新しい圧縮データフォーマット216のデータ中の0ビットとして与えられ得る。プレフィックスコード010は、パターン0xFFFFFFFFを示し、それは特定のパターンであり、したがって、新しい圧縮データフォーマット216に従って圧縮データ内に0ビットデータサイズを必要とする。他のパターンが、プレフィックスコード011～111に対する頻出パターン符号化表220内で示される。新しい圧縮データフォーマット216内のフラグフィールドは、プレフィックスコード001～111に対するどのパターンが圧縮データのデータ部(すなわち、Datax)内に存在するかを示す。パターンが圧縮データ内に存在する場合、パターンは、新しい圧縮データフォーマット216内に記憶され、そのパターンは次いで、解凍データを再生成するために参照され得る。データフィールドは、新しい圧縮データフォーマット216内のデータフィールドと関連付けられたプレフィックスコードによる圧縮データを含む。

【0088】

図18は、64ビット頻出パターン圧縮データ圧縮方式226の別の例を示す。この点について、圧縮されるべきソースデータフォーマット228内のソースデータは、本例では128バイトとして示されている。新しい圧縮データフォーマット230は、プレフィックスコードPx、データDatax、フラグ、およびパターンの異なるフォーマットにおいて与えられ、それらは、効率を目的として一緒にグループ化されるように編成される。プレフィックスコード

10

20

30

40

50

ドは、4ビットである。プレフィックスコードは、プレフィックスコード列232、234内の所与のプレフィックスコードに対するパターン符号化列238、240内で符号化されたパターンを示す頻出パターン符号化表236内のプレフィックスコード列232、234内に示される。符号化されたパターンに対するデータサイズは、頻出パターン符号化表236のデータサイズ列242、244内に与えられる。プレフィックスコード0000は、すべてゼロのデータブロックを示し、それは、新しい圧縮データフォーマット230のデータ中の0ビットとして与えられ得る。他のパターンは、プレフィックスコード0001～1111に対する頻出パターン符号化表236内に示され、それらのパターンは、頻繁に発生するASCIIパターンに対するASCIIパターンを含む。新しい圧縮データフォーマット230内のフラグフィールドは、プレフィックスコード0001～1111に対するどのパターンが圧縮データのデータ部(すなわち、Datax)内に存在するかを示す。パターンが圧縮データ内に存在する場合、パターンは、新しい圧縮データフォーマット230内に記憶され、そのパターンは次いで、解凍データを再生成するために参照され得る。データフィールドは、新しい圧縮データフォーマット230内のデータフィールドと関連付けられたプレフィックスコードによる圧縮データを含む。

【0089】

図19は、64ビット頻出パターン圧縮データ圧縮方式246の別の例を示す。この点について、圧縮されるべきソースデータフォーマット248内のソースデータは、本例では128ビットとして示されている。新しい圧縮データフォーマット250は、プレフィックスコードPx、データDatax、フラグ、およびパターンの異なるフォーマットにおいて与えられ、それらは、効率を目的として一緒にグループ化されるように編成される。プレフィックスコードは、4ビットである。プレフィックスコードは、プレフィックスコード列252、254内の所与のプレフィックスコードに対するパターン符号化列258、260内で符号化されたパターンを示す頻出パターン符号化表256内のプレフィックスコード列252、254内に示される。符号化されたパターンに対するデータサイズは、頻出パターン符号化表256のデータサイズ列262、264内に与えられる。プレフィックスコード0000は、すべてゼロのデータブロックを示し、それは、新しい圧縮データフォーマット250のデータ中の0ビットとして与えられ得る。他のパターンは、プレフィックスコード0001～1111に対する頻出パターン符号化表256内に示され、それらのパターンは、固定パターンの組合せを含むことができる。新しい圧縮データフォーマット250内のフラグフィールドは、プレフィックスコード0001～1111に対するどのパターンが圧縮データ内のデータ部(すなわち、Datax)内に存在するかを示す。パターンが圧縮データ内に存在する場合、パターンは、新しい圧縮データフォーマット250内に記憶され、そのパターンは次いで、解凍データを再生成するためにデータ圧縮の間に参照され得る。プレフィックスコードP0～P31はパターンとリンクすることができ、それらのパターンは、解凍フォーマットにおけるフルレンジスデータを再生成するために、対応するデータ(Datax)とともに使用される。データフィールドは、新しい圧縮データフォーマット250内のデータフィールドと関連付けられたプレフィックスコードによる圧縮データを含む。

【0090】

図19の頻出パターン圧縮データ圧縮方式246を用いて使用され得る固定パターンの例が、図20の表266に示され、固定パターンは、長さ列270のその長さおよびパターン定義列272内のパターンの定義とともに、パターン列268内に与えられる。プレフィックスコードにリンクされた所与のパターンを、解凍データを生成するために使用される定義と相關させることを、圧縮メモリコントローラ36などのメモリコントローラが可能にするために、フラグ定義がフラグ定義表274内に示される。フラグ定義表274は、フラグ列276内の所与のフラグに対するビットと、フラグ値列278内の所与のフラグに対するビットの値と、フラグ定義列280内の所与のフラグに対するフラグ定義とを含む。

【0091】

図21は、64ビット頻出パターン圧縮データ圧縮方式282の別の例を示す。この点について、圧縮されるべきソースデータフォーマット284内のソースデータは、本例では128ビットとして示されている。新しい圧縮データフォーマット286は、プレフィックスコードPx

10

20

30

40

50

、データDatax、フラグ、およびパターンの異なるフォーマットにおいて与えられ、それらは、効率を目的として一緒にグループ化されるように編成される。プレフィックスコードは、4ビットである。プレフィックスコードは、プレフィックスコード列288、290内の所与のプレフィックスコードに対するパターン符号化列294、296内で符号化されたパターンを示す頻出パターン符号化表292内のプレフィックスコード列288、290内に示される。符号化されたパターンに対するデータサイズは、頻出パターン符号化表292のデータサイズ列298、300内に与えられる。プレフィックスコード0000は、すべてゼロのデータブロックを示し、それは、新しい圧縮データフォーマット286のデータ中の0ビットとして与えられ得る。プレフィックスコード1111は、新しい圧縮データフォーマット286内で圧縮されていないデータブロックを示す。他のパターンは、プレフィックスコード0001～1110に対する頻出パターン符号化表292内に示され、それらのパターンは、その中に示される定義パターンの組合せを含むことができる。新しい圧縮データフォーマット286内のフラグフィールドは、プレフィックスコード0000～1110に対するどのパターンが圧縮データのデータ部(すなわち、Datax)内に存在するかを示す。パターンが圧縮データ内に存在する場合、パターンは、新しい圧縮データフォーマット286内に記憶され、そのパターンは次いで、解凍データを再生成するために参照され得る。新しい圧縮データフォーマット286は、パターン0～5だけを含むように示されるが、その理由は、本例では、これらのパターンだけが、ソースデータ内に存在するプレフィックスコード0000～1110内に存在するパターンであるからである。データフィールドは、新しい圧縮データフォーマット286内のデータフィールドと関連付けられたプレフィックスコードによる圧縮データを含む。

10

20

30

【0092】

上記で説明したように、メモリ容量圧縮方式および/またはメモリ帯域幅圧縮方式が、所望の圧縮比を達成するために利用され得る。たとえば、図22Aは、システムメモリ38内で解凍形態でメモリデータブロック/行内に編成された例示的なメモリデータ302の図であり、メモリデータ302は、所望の圧縮比を達成するためにメモリ容量圧縮方式および/またはメモリ帯域幅圧縮方式を使用して圧縮され得る。図22Bは、メモリ帯域幅圧縮方式を利用する圧縮メモリコントローラ36によって圧縮された後の、図22Aのメモリデータ302の図である。上記の例で説明したように、メモリ帯域幅圧縮は、バス上の帯域幅を低減するためにバス上のデータを圧縮することを伴う。バス上の帯域幅を低減することで、バス待ち時間および電力消費を低減することができる。さらに、データはPAロケーションにおいて、単に圧縮形態または解凍形態のいずれかで記憶されているので、メモリ帯域幅圧縮方式は、メモリ容量圧縮方式において実施されるようなPAのさらなる変換の必要性を伴わない。図22Cは、メモリ容量圧縮方式を利用する圧縮メモリコントローラ36によって圧縮された後の、図22Aのメモリデータ302の図である。同じく上記の例で説明したように、メモリ容量圧縮は、データを圧縮形態でメモリに記憶することを伴う。しかしながら、待ち時間は、この圧縮の結果として、圧縮メモリコントローラ36と関連付けられた追加のオーバーヘッドによって影響を受けることがある。性能に対して幾つかの圧縮効率のトレードオフを行うことが望ましい。

30

【0093】

この点について、次の読み取りアドレスプリフェッチング方式は、図22Bおよび図22Cにそれぞれ示すように、メモリ帯域幅圧縮方式および/またはメモリ容量圧縮方式と組み合わされてよい。次の読み取りアドレスプリフェッチング方式は、現在アクセスされているPAを超える別のアドレスにおいてシステムメモリ38からデータを推論的にプリフェッチするために、圧縮メモリコントローラ36によって使用される。圧縮メモリデータを保持するメモリブロック306内に使い残し空間が存在することがあり得るので、システムメモリ38からプリフェッチされるべき次の読み取りアドレスは、図22Bおよび図22Cに示すシステムメモリ38内に記憶されるメモリデータ302に関連するメタデータ304として記憶され得る。たとえば、特定のメモリブロック306において記憶される圧縮メモリデータは、そのメモリブロック306の限界まで圧縮しないことがあり、したがって追加のメモリ空間がメモリブロック306内に使い残される。たとえば、メモリブロック306が32バイトのサイズであり、圧

40

50

縮メモリデータが50バイトまで圧縮される場合、圧縮メモリデータは、合計64バイトの2つのメモリブロック306にわたって記憶され、14バイトが未使用となる。したがって、メモリデータ302がシステムメモリ38内にアドレス指定されるとき、次の読み取りアドレスがメモリブロック308と関連付けられたメタデータ304に記憶されるならば、次の読み取りアドレスにおけるメモリデータ302は、次の読み取りアドレスが圧縮メモリコントローラ36によって受信される次のPAである場合に利用可能となるシステムメモリ38内で、圧縮メモリコントローラ36によってプリフェッチされ得る。メモリ帯域幅圧縮方式および/またはメモリ容量圧縮方式とともに使用されると、システムメモリ38内でメモリブロック306内の使い残し空間を使用しながら付加的性能が達成され得る。

【0094】

10

この点について、図23は、次の読み取りアドレスプリフェッチ方式によるメモリ容量圧縮方式を利用するメモリデータ圧縮を実施するように構成された圧縮メモリコントローラ36を利用する、図2のSOC10'に対する代替SOC10(3)内に設けられ得るメモリ関連構成要素の構造図である。その中に示すように、この例では、プリフェッチメタデータ(PMD)キャッシュ310が追加で設けられる。CIキャッシュ168は、この例におけるメモリ帯域幅圧縮方式を実施するために設けられる。隨意のL4キャッシュ142もまた設けられる。図24A～図24Cは、次に説明する次の読み取りアドレスプリフェッチ方式によるメモリ容量圧縮方式を利用することによって、圧縮メモリコントローラ36がメモリデータ圧縮を実施することを可能にするために、図23のSOC10(3)において利用され得る例示的なデータ構造の概略図である。これらのデータ構造は、各ハードウェアスレッドがメモリ容量圧縮をもたらすことができるよう、各CPU16(図2参照)内に各ハードウェアスレッドを設けられることがある。

20

【0095】

この点について、図24Aは、システムメモリ38からプリフェッチされるべき現在の以前読み取りアドレスを含む以前アドレスデータレジスタ312であり、現在の以前読み取りアドレスは、この例では0x9800である。図24Bは、被アクセスアドレス314に関連して記憶された次の読み取りアドレス316を有するメモリアクセスのために、圧縮メモリコントローラ36によってアクセスされる被アクセスアドレス314を含むいくつかのキャッシュラインを有するPMDキャッシュ310を示す。図24Cは、前述のメモリデータをキャッシュするため隨意に設けられることがあるL4キャッシュ142を示す。図25は、図23のSOC10(3)に対するメモリ構成要素を示す。図24A～図24Cに戻って参考すると、たとえば、被アクセスアドレス314が最初に0x1000である場合、次の読み取りアドレス316は、この被アクセスアドレス314と関連付けられることはない。しかしながら、圧縮メモリコントローラ36によって受信された、アクセスされるべき次のメモリアドレスが0x1080である場合、圧縮メモリコントローラ36は、この例では、PMDキャッシュ310内の被アクセスアドレス0x1000に関連するアドレス0x1080を記憶する。したがって、次にメモリアドレス0x1000がシステムメモリ38からアクセスされるために圧縮メモリコントローラ36に与えられるときに、メモリアドレス0x1000は、それに関連する次の読み取りアドレス316が、アクセスされる実際の次の読み取りアドレス316である場合に推論的にプリフェッチされるべき関連する次の読み取りアドレス316を有するかどうかを決定するために、圧縮メモリコントローラ36は、PMDキャッシュ310内でルックアップを実施することができる。PMDキャッシュ310は、メモリアドレスアクセスが発生するときに、最終的に満たされることになる。

30

【0096】

40

圧縮メモリコントローラ36はまた、圧縮されたPAにおいてメモリブロック内に利用可能な空き場所がある場合、図22Bおよび図22Cに示すように、圧縮データが圧縮されたPAに対して記憶されるとき、圧縮されたPAに関連するシステムメモリ38内のメタデータ304として次の読み取りアドレス316を記憶することになる。したがって、圧縮メモリコントローラ36は、次の読み取りアドレス316がPMDキャッシュ310内にない場合、プリフェッチするためにシステムメモリ38内の被アクセス圧縮データに対する次の読み取りアドレス316にアクセスすることができる。システムメモリ38に記憶されている次の読み取りアドレス316はまた、以下で説明するように、PMDキャッシュ310における立ち退き発生時にPMDキャッシュ310を

50

満たすために使用され得る。

【0097】

したがって、メモリ読取り動作時に、圧縮メモリコントローラ36は、PMDキャッシュ310内の現在の読取りアドレスに関連する以前読取りアドレスを記憶するように構成される。PMDキャッシュ310内のキャッシュラインは、必要ならば立ち退かされることになる。現在の読取りアドレスに対してL4キャッシュ142に対するヒット発生時、圧縮メモリコントローラ36は、L4キャッシュ142への次の読取りアドレス316においてシステムメモリ38からメモリデータをプリフェッチするために、PMDキャッシュ310内の現在の読取りアドレスにアクセスするときにヒットが発生する場合、PMDキャッシュ310から次の読取りアドレス316を得る。L4キャッシュ142に対するミス発生時、キャッシュラインがL4キャッシュ142から立ち退かされ、データが、システムメモリ38から圧縮されたPAにおける読取りアドレスにおいて読み取られ、L4キャッシュ142に記憶される。L4キャッシュ142と関連付けられた次の読取りアドレス316がPMDキャッシュ310内にある場合、PMDキャッシュ310内に記憶されている次の読取りアドレス316が、次の読取りアドレス316におけるメモリデータをプリフェッチするために使用される。L4キャッシュ142と関連付けられた次の読取りアドレス316がPMDキャッシュ310内にない場合、システムメモリ38内の次の読取りアドレス316において圧縮されたPAと関連付けられたメタデータ304に記憶されている次の読取りアドレス316が、次の読取りアドレス316におけるメモリデータをプリフェッチするために使用される。この後の状況において、メタデータ304内の次の読取りアドレス316がシステムメモリ38内の圧縮されたPAにおいて有効である場合、この次の読取りアドレス316もまた、アクセスされる読取りアドレスに関連するPMDキャッシュ310内に記憶されることになる。

10

20

30

30

40

50

【0098】

メモリ書き込み動作時、およびアクセスされるべきアドレスに基づくL4キャッシュ142に対するヒット時に、圧縮メモリコントローラ36は、古いキャッシュラインを、システムメモリ38からアクセスされたメモリデータを有する新しいキャッシュラインと置き換える。

【0099】

L4キャッシュ142に対するミス発生時、キャッシュラインがL4キャッシュ142から立ち退かされ、システムメモリ38からの次のメモリデータがL4キャッシュ142に記憶される。L4キャッシュ142から立ち退かされると、次の読取りアドレス316は、PMDキャッシュ310から、L4キャッシュ142内の立ち退かされたキャッシュラインと関連付けられたアドレスに対して取得される。L4キャッシュ142から立ち退かされたキャッシュラインがダーティである場合、次の読取りアドレス316は、システムメモリ38内のメタデータ304内に書き戻される。そうではなく、次の読取りアドレス316がダーティである場合、次の読取りアドレス316は、設定可能な確率によってメタデータ304に書き戻される。

【0100】

本明細書において開示される態様に関して説明される種々の例示的な論理ブロック、モジュール、回路、およびアルゴリズムは、電子ハードウェア、メモリもしくは別のコンピュータ可読媒体に記憶され、プロセッサもしくは他の処理デバイスによって実行される命令、または両方の組合せとして実現できることを、当業者はさらに理解されよう。本明細書において開示されるメモリは、任意のタイプおよびサイズのメモリとすることができる、また、任意のタイプの所望の情報を記憶するように構成することができる。この交換可能性を明確に示すために、様々な例示的な構成要素、ブロック、モジュール、回路、およびステップについては上記で、それらの機能の観点から一般的に説明してきた。そのような機能がどのように実現されるかは、特定の用途、設計上の選択、および/または、システム全体に課された設計上の制約によって決まる。当業者は、説明した機能を特定の適用例ごとに様々な方法で実装し得るが、そのような実装の決定は、本開示の範囲からの逸脱を引き起こすものと解釈されるべきではない。

【0101】

本明細書において開示される態様に関して説明される種々の例示的な論理ブロック、モジュール、および回路は、プロセッサ、デジタルシグナルプロセッサ(DSP)、特定用途向

50

け集積回路(ASIC)、フィールドプログラマブルゲートアレイ(FPGA)もしくは他のプログラマブル論理デバイス、個別ゲートもしくはトランジスタ論理、個別ハードウェア構成要素、または本明細書において説明される機能を実行するように設計されたそれらの任意の組合せにおいて実現または実行することができる。プロセッサは、マイクロプロセッサとすることができるが、代替形態では、プロセッサは、任意の従来のプロセッサ、コントローラ、マイクロコントローラ、または状態機械とすることができます。プロセッサはまた、コンピューティングデバイスの組合せ、たとえば、DSPとマイクロプロセッサの組合せ、複数のマイクロプロセッサ、DSPコアと連携した1つもしくは複数のマイクロプロセッサ、または任意の他のそのような構成として実装され得る。

【0102】

10

本明細書において開示される態様は、ハードウェアにおいて、また、ハードウェアに記憶された命令において具現される場合があり、命令は、たとえば、ランダムアクセスメモリ(RAM)、フラッシュメモリ、読み取り専用メモリ(ROM)、電気的プログラマブルROM(EPROM)、電気的消去可能プログラマブルROM(EEPROM)、レジスタ、ハードディスク、リムーバブルディスク、CD-ROM、または当技術分野において知られている任意の他の形態のコンピュータ可読媒体内に存在することができる。例示的な記憶媒体は、プロセッサが記憶媒体から情報を読み取り、記憶媒体に情報を書き込むことができるよう、プロセッサに結合される。代替として、記憶媒体は、プロセッサに一体化されてもよい。プロセッサおよび記憶媒体は、ASICに存在してもよい。ASICはリモート局に存在してもよい。代替として、プロセッサおよび記憶媒体は、個別構成要素として、リモート局、基地局、またはサーバに存在してもよい。

【0103】

20

本明細書の例示的な態様のいずれかにおいて説明された動作ステップは、例および考察を提供するために説明されることにも留意されたい。説明される動作は、図示されるシーケンス以外の多数の異なるシーケンスにおいて実行することができる。さらに、単一の動作ステップに記載した動作は、実際にはいくつかの異なるステップで実行されてもよい。さらに、例示的態様において論じられる1つまたは複数の動作ステップが組み合わせられる場合がある。フローチャート図において図示した動作ステップは、当業者に容易に明らかとなるような多くの異なる修正を受けてもよいことを理解されたい。また、情報および信号を種々の異なる技術および技法のいずれかを使用して表すことは、当業者には理解されよう。たとえば、上記の説明全体にわたって参照される場合があるデータ、命令、コマンド、情報、信号、ビット、シンボル、およびチップは、電圧、電流、電磁波、磁場もしくは磁性粒子、光場もしくは光学粒子、またはそれらの任意の組合せによって表すことができる。

30

【0104】

本開示のこれまでの説明は、当業者が本開示を作製または使用することを可能にするために提供される。本開示に対する種々の変形態が当業者には容易に明らかとなり、本明細書において規定される一般原理は、本開示の趣旨または範囲から逸脱することなく、他の変形態に適用することができる。したがって、本開示は、本明細書において説明される例および設計に限定されるものではなく、本明細書において開示される原理および新規の特徴に合致する最も広い範囲を与えられるべきである。

40

【符号の説明】

【0105】

50

- 10 システムオンチップ(SOC)
- 10(1) SOC
- 10(2) SOC
- 10(3) SOC
- 10' SOC
- 12 CPUベースシステム
- 12' CPUベースシステム

14(1) CPU ブロック	
14(N) CPU ブロック	
16 CPU	
16(1) CPU	
16(2) CPU	
18(1) キャッシュ	
18(N) キャッシュ	
20 キャッシュ	
22 内部システムバス	
24 メモリコントローラ	10
26 周辺機器	
28 記憶装置	
30 高速周辺機器相互接続(PCI)(PCI-e)インターフェース	
32 直接メモリアクセス(DMA)コントローラ	
34 IMC	
36 圧縮メモリコントローラ	
38 システムメモリ	
40(1) ダブルデータレート(DDR)ダイナミックランダムアクセスメモリ(DRAM)	
40(R) ダブルデータレート(DDR)ダイナミックランダムアクセスメモリ(DRAM)	
42 圧縮キャッシュ	20
44 半導体ダイ	
46(1) 半導体ダイ	
46(2) 半導体ダイ	
48(1) DRAMインターフェース	
48(2) DRAMインターフェース	
48(P) DRAMインターフェース	
50 圧縮コントローラ	
52 ローカルメモリ	
54 SRAM	
55 L4コントローラ	30
56 内部メモリ	
58 内部メモリコントローラ	
60 行ベースバッファメモリ容量圧縮方式	
62 ビット長プール、PBプール	
62(1) ビット長プール、PBプール	
62(2) ビット長プール、PBプール	
62(3) PBプール	
62(4) PBプール	
62(Q) ビット長プール、PBプール	
64 フリーノードリスト	40
64(1) フリーノードリスト	
66 マスター・ディレクトリ	
68 トランスレーションルックアサイドバッファ(TLB)	
70 タグアレイ	
72 タグ	
74 TLBエントリ	
76 データアレイ	
78 アドレス長	
80 2ビットの符号化	
82 ページベースメモリ容量圧縮方式	50

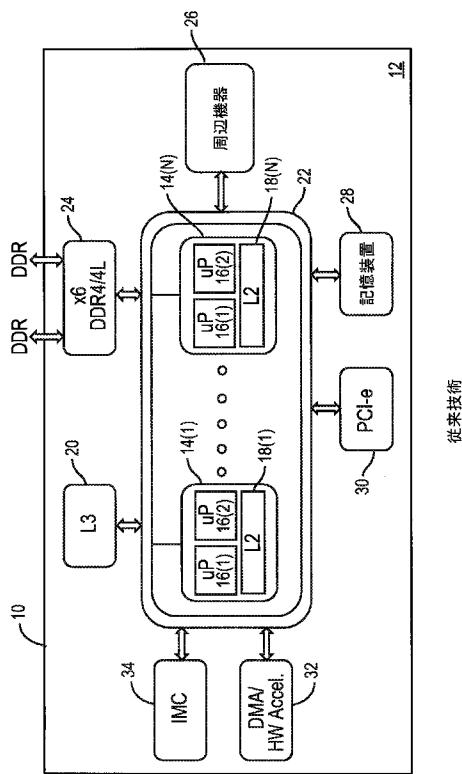
84	TLB	
86	キャッシュライン	
88	タグ	
90	タグアレイ	
92	データアレイ	
96	フリー ノードリスト	
98(1)	DPAフリー リスト	
98(2)	DPAフリー リスト	
98(R)	DPAフリー リスト	
100(1)	フリー DPA	10
100(2)	フリー DPA	
100(R)	フリー DPA	
102	ローカル ブルバッファ	
104	ヘッド ポインタ	
106	テール ポインタ	
108	ローカル ブッショバッファ	
110	ハイブリッド行/ページベースバッファメモリ容量圧縮方式	
112(1)	物理ブロック	
112(2)	物理ブロック	
112(3)	物理ブロック	20
112(4)	物理ブロック	
112(Q)	物理ブロック	
114	バッファポインタ	
114(1)	バッファポインタ	
114(4)	バッファポインタ	
114(S)	バッファポインタ	
116	メモリデータブロック	
118	ページ情報	
142	キャッシュ、L4キャッシュ	
144	圧縮PA(CPA)	30
146	履歴バッファ	
148	CPA履歴バッファ	
150	ホストOS	
152(1)	ゲストOS	
152(2)	ゲストOS	
152(T)	ゲストOS	
154	メモリコントローラドライバ	
156(1)	データ圧縮ドライバ(DCD)	
156(2)	データ圧縮ドライバ(DCD)	
156(T)	データ圧縮ドライバ(DCD)	40
158	圧力データ	
160	メモリ帯域幅圧縮方式	
162	データライン	
165	エントリ	
166	ECC圧縮インジケータ	
168	圧縮インジケータ(CI)キャッシュ、CIキャッシュ	
170	キャッシュエントリ	
172	キャッシュライン	
174	圧縮インジケータ	
180	頻出パターン圧縮データ圧縮方式	50

182	ソースデータフォーマット	
184	圧縮データフォーマット	
186	プレフィックスコード列	
188	頻出パターン符号化表	
190	パターン符号化列	
192	データサイズ列	
194	頻出パターン圧縮データ圧縮方式	
196	ソースデータフォーマット	
198	圧縮データフォーマット	
200	圧縮データフォーマット	10
202	プレフィックスコード列	
204	頻出パターン符号化表	
206	パターン符号化列	
208	データサイズ列	
210	頻出パターン圧縮データ圧縮方式	
212	ソースデータフォーマット	
214	圧縮データフォーマット	
216	圧縮データフォーマット	
218	プレフィックスコード列	
220	頻出パターン符号化表	20
222	パターン符号化列	
224	データサイズ列	
226	頻出パターン圧縮データ圧縮方式	
228	ソースデータフォーマット	
230	圧縮データフォーマット	
232	プレフィックスコード列	
234	プレフィックスコード列	
236	頻出パターン符号化表	
238	パターン符号化列	
240	パターン符号化列	30
242	データサイズ列	
244	データサイズ列	
246	頻出パターン圧縮データ圧縮方式	
248	ソースデータフォーマット	
250	圧縮データフォーマット	
252	プレフィックスコード列	
254	プレフィックスコード列	
256	頻出パターン符号化表	
258	パターン符号化列	
260	パターン符号化列	40
262	データサイズ列	
264	データサイズ列	
266	表	
268	パターン列	
270	長さ列	
272	パターン定義列	
274	フラグ定義表	
276	フラグ列	
278	フラグ値列	
280	フラグ定義列	50

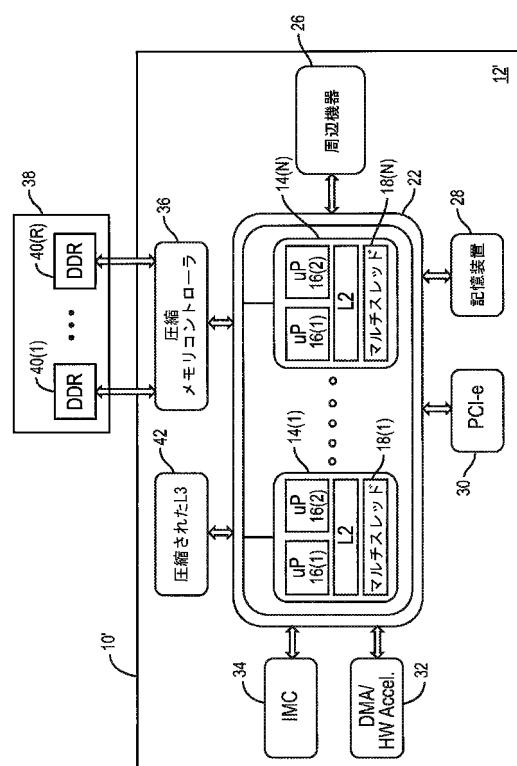
- | | |
|-----|------------------------|
| 282 | 頻出パターン圧縮データ圧縮方式 |
| 284 | ソースデータフォーマット |
| 286 | 圧縮データフォーマット |
| 288 | プレフィックスコード列 |
| 290 | プレフィックスコード列 |
| 292 | 頻出パターン符号化表 |
| 294 | パターン符号化列 |
| 296 | パターン符号化列 |
| 298 | データサイズ列 |
| 300 | データサイズ列 |
| 302 | メモリデータ |
| 304 | メタデータ |
| 306 | メモリロック |
| 308 | メモリロック |
| 310 | プリフェッヂメモリデータ(PMD)キャッシュ |
| 312 | 以前アドレスデータレジスタ |
| 314 | 被アクセスアドレス |
| 316 | 次の読み取りアドレス |

10

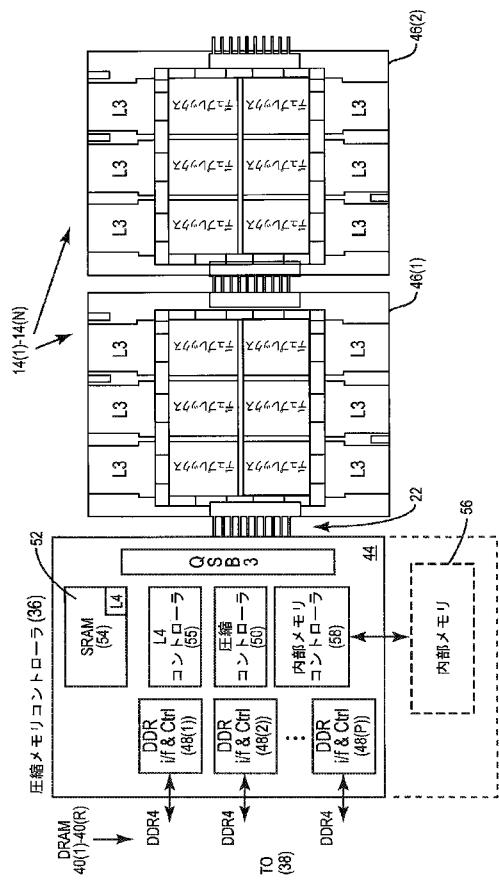
【 図 1 】



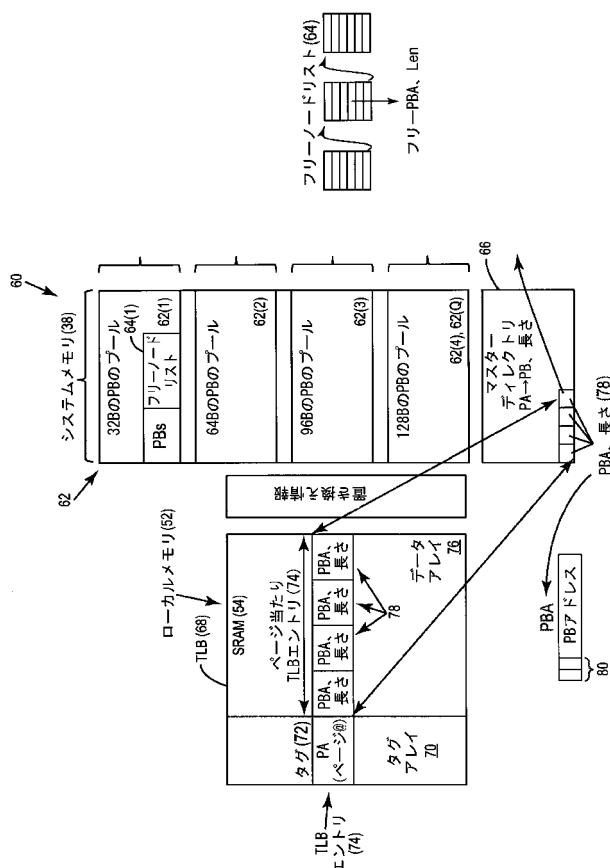
【 図 2 】



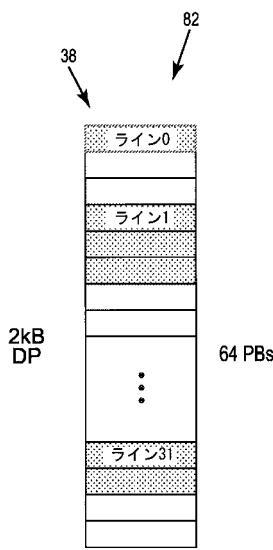
【図3】



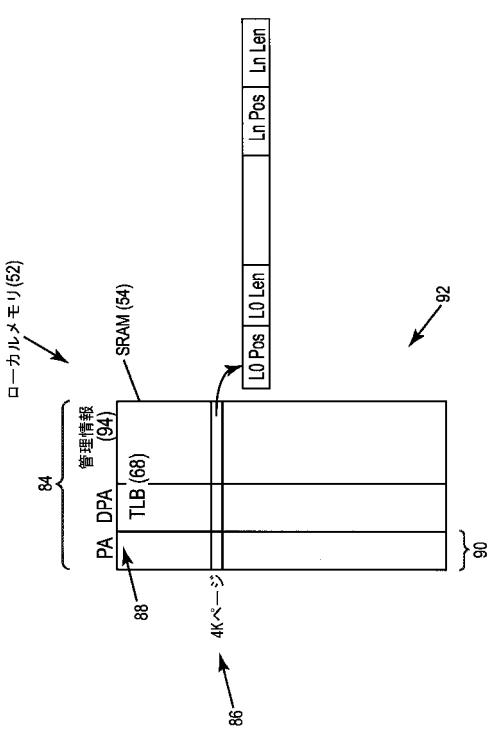
【図4】



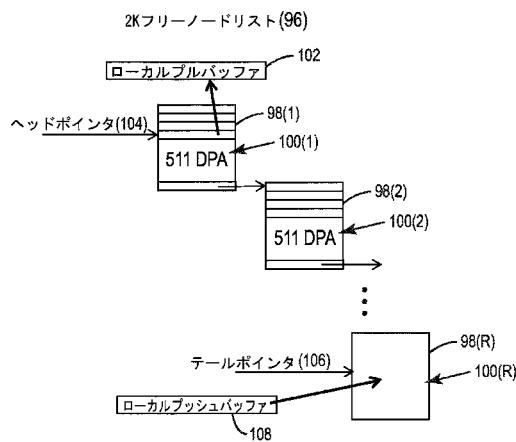
【図5】



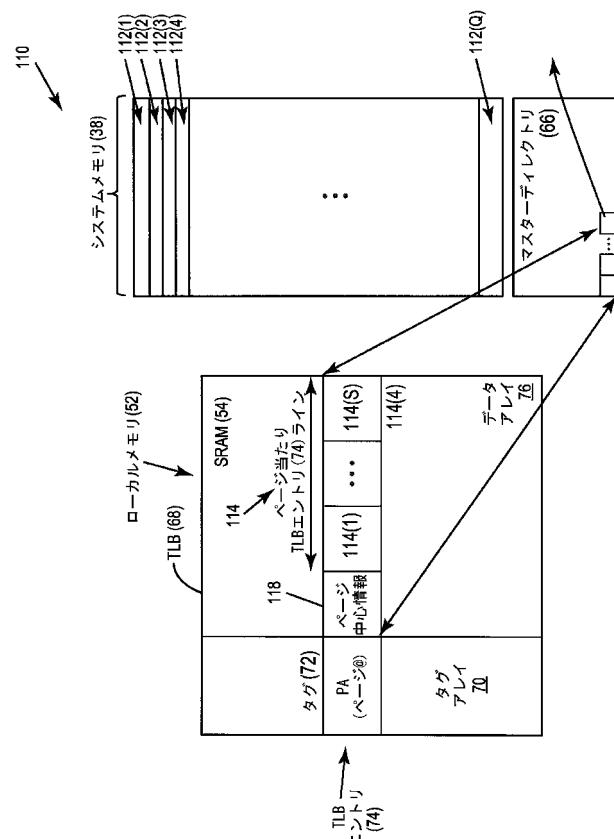
【図6】



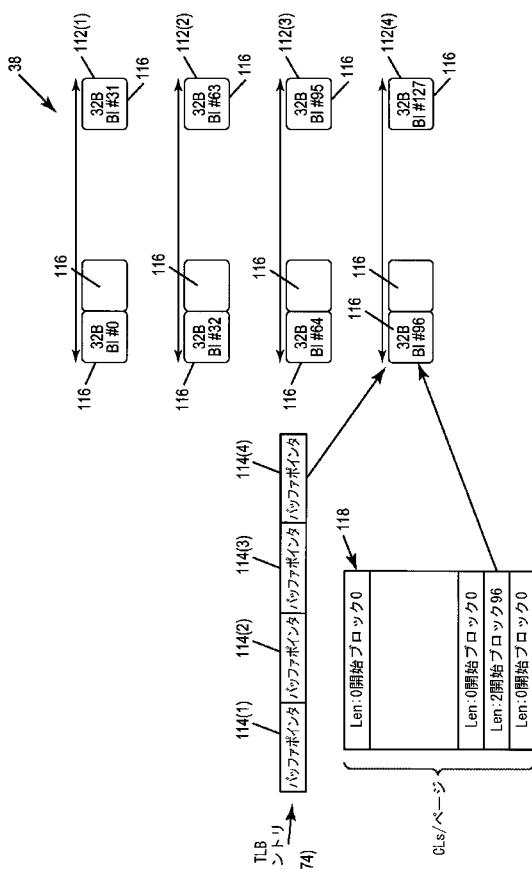
【図7】



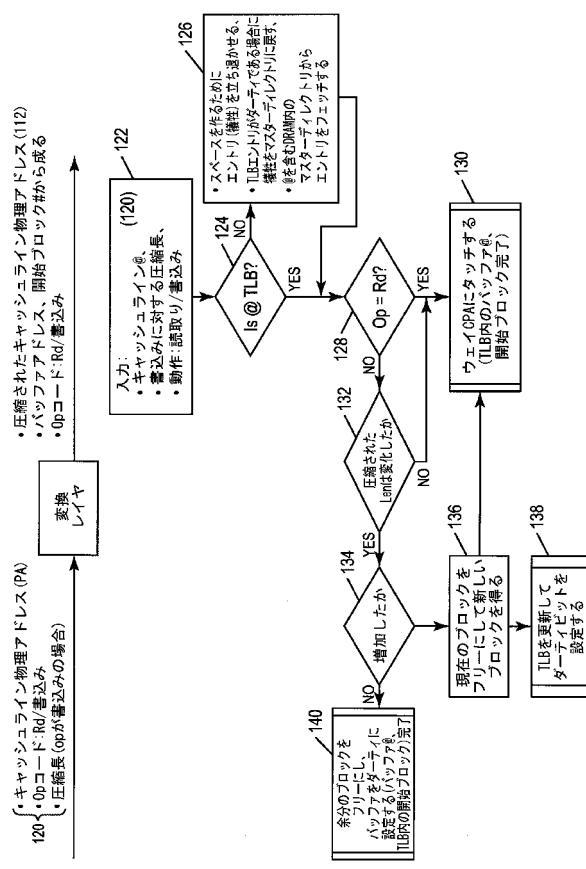
【 四 8 】



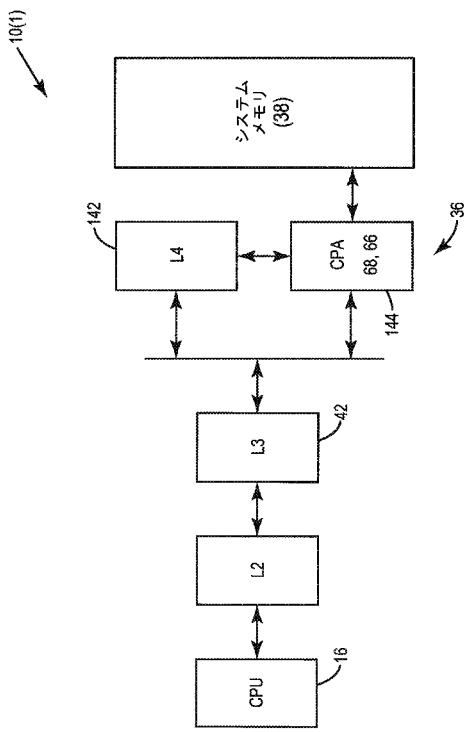
〔 図 9 〕



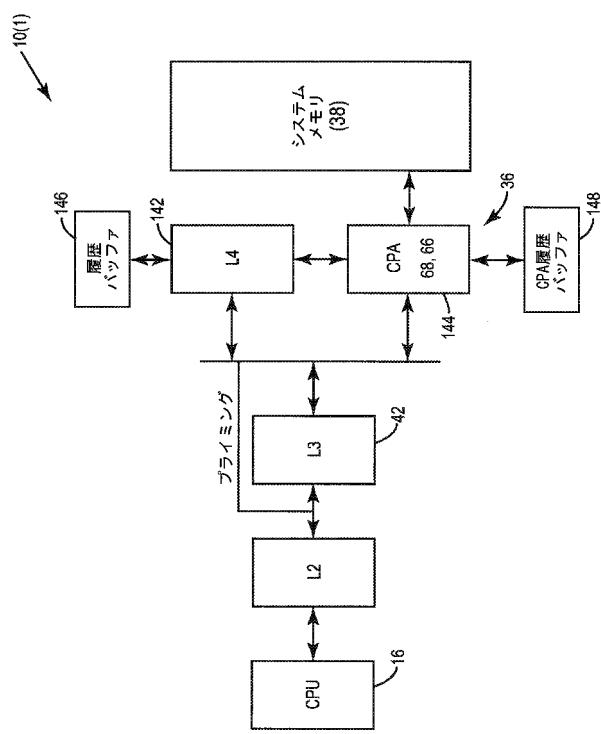
【 図 1 0 】



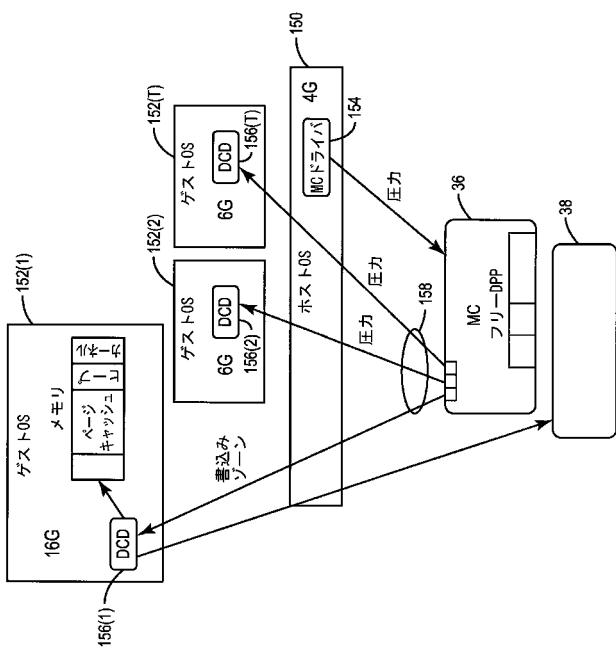
【図 1 1 A】



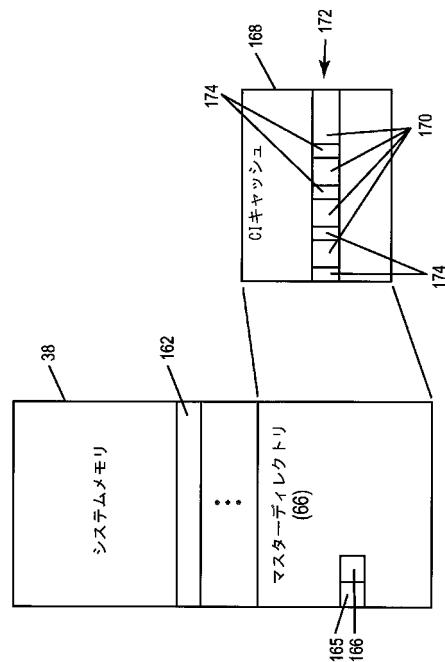
【図 1 1 B】



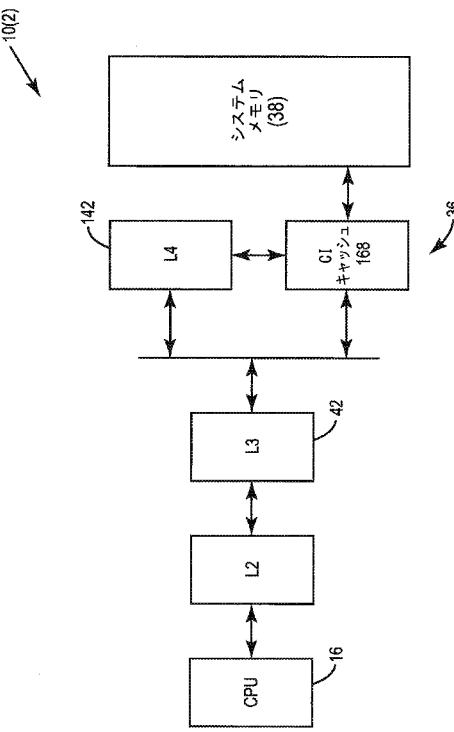
【図 1 2】



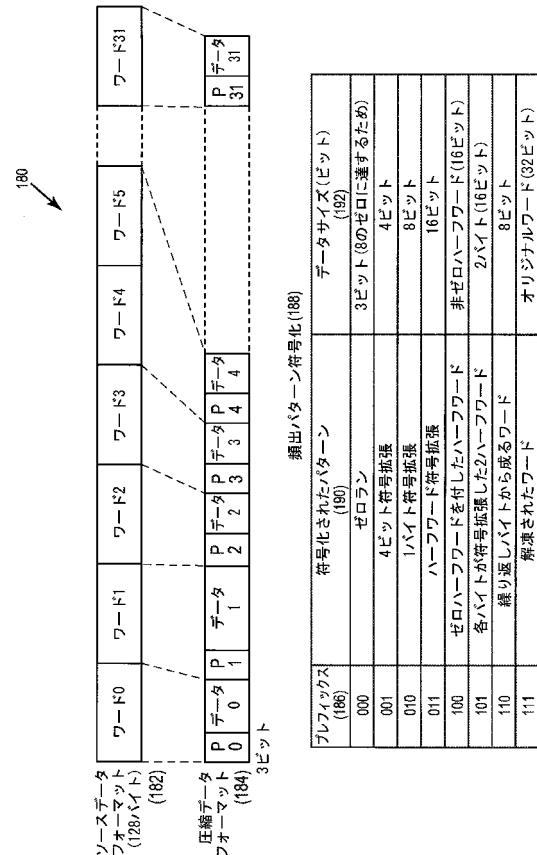
【図 1 3】



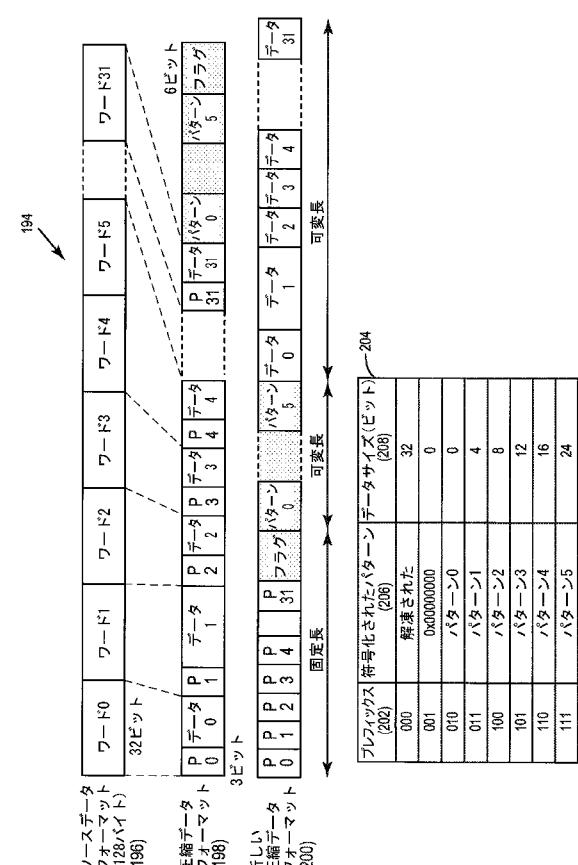
【 図 1 4 】



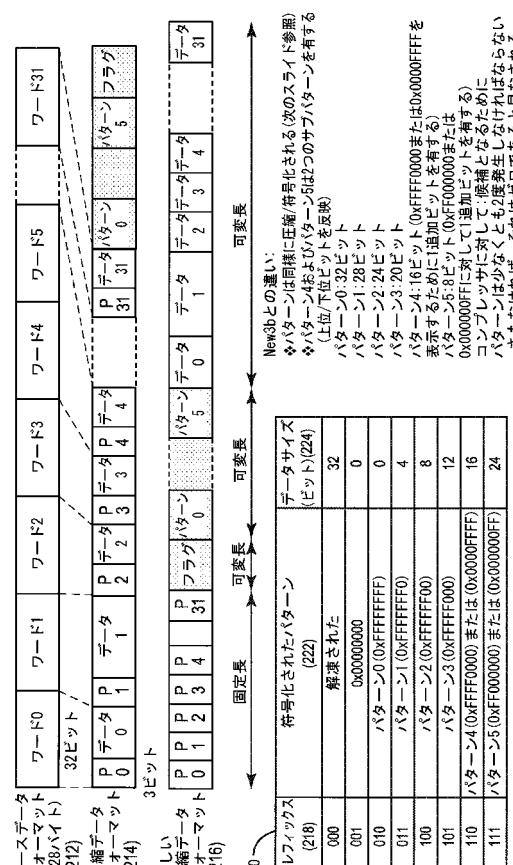
【図15】



【 図 1 6 】



【 図 1 7 】



【 図 1 8 】

【 図 2 0 】

バターン	長さ	定義(212)	266
バターン0	32	64ビットワードのうちのビット63~32をカバーする	270
バターン1	16	64ビットワードのうちのビット31~16をカバーする	268
バターン2	16	64ビットワードのうちのビット15~0をカバーする	266

【 図 19 】

プロパティ (238)	符号化されたパターン (238)	データサイズ (ビット) (242)	フレックス (234)	符号化されたパターン (240)	データサイズ (ビット) (244)
0000	0x0000000000000000	0	1000	0x20~0x3Fの範囲内に2個のASCII CHR	44
0001	0x000000000000YYYY	16	1001	0x00~0x7Fの範囲内に2個のASCII CHR	
0010	0x0000000000YYYYYY	40	1010	0x00~0x7Fの範囲内に2個のASCII CHR	44
0011	バーチ-0	0	1011	0x00~0x7Fの範囲内に1個のASCII CHR	
0100	バーチ-1	16	1100	64の所定の語書から8個のASCII CHR (小文字を重複)	48
0101	バーチ-2	32	1101	64の所定の語書から8個のASCII CHR (大文字を重複)	48
0110	0x20~0x3Fの範囲内に2個のASCII CHR	40	1110	0x00~0x7Fの範囲内に3個のASCII CHR	56
0111	0x00~0x7Fの範囲内のASCII	40	1111	解凍された-	64

【 図 2 1 】

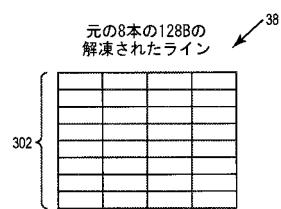
ワード0	ワード1	ワード2	ワード3	ワード4	ワード5	ワード15
64ビット						
4ビット						

(45)

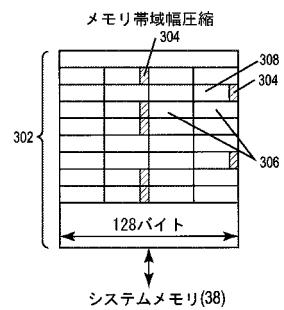
データサイズ (ビット)	データ化されたノット (258)	データ化されたノット (254)	データ化されたノット (260)	データサイズ (ビット) (264)
0000	0x0000000000000000	0	1000	解乗された
0001	0x0000000000000001	0	1001	ノット0を使用するワード
0010	0x0000000000000000y	4	1010	ノット1を使用するワード
0011	0x0000000000000000yy	8	1011	ノット-1+ノット0を使用するワード
0100	0x00000000000000yy	12	1100	ノット-2を使用するワード
0101	0x00000000000000yyyy	16	1101	ノット-2+ノット-1を使用するワード
0110	0x00000000000000yyyy	20	1110	ノット-2+ノット-1+ノット0を使用するワード
0111	0x00000000000000yyyy	20	1111	ノット-2+ノット-1+ノット0+ノット1を使用するワード

フレーマークス符号化されたバーバーン (294) (288)	データサイズ (ビット) (298)	フレーマークス符号 (280)
00000	0	1000
00001	4	1001
00010	8	1010
00011	12	1011
00000	16	1100
00001	16	1101
00010	20	1110
00011	32	1111

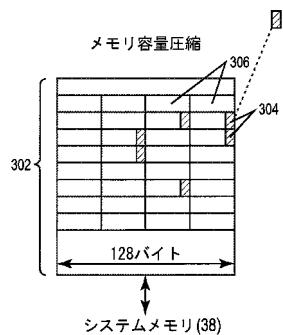
【図 2 2 A】



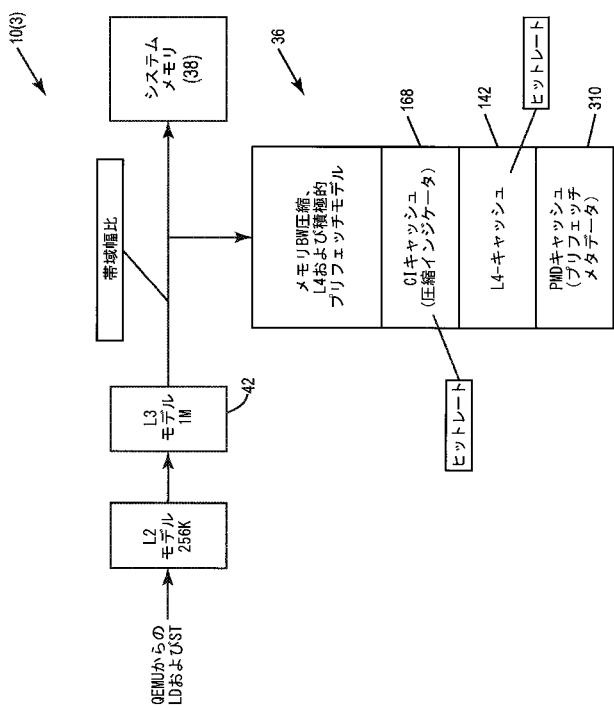
【図 2 2 B】



【図 2 2 C】



【図 2 3】



【図 2 4 A】

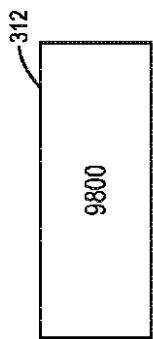


FIG. 24A

【図 2 4 B】

314	316	314	316
1000 → 1080	2200 → 9800		
1080 → 1200			
1200 → 0080			
0080 → 2200			

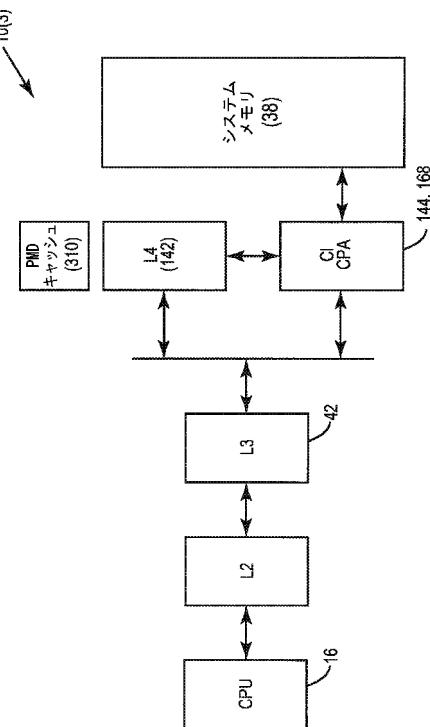
PMDキャッシュ
(310)

【図 2 4 C】

1000 + Data	2200 + Data
1080 + Data	9800 + Data
1200 + Data	
0080 + Data	

L4-キャッシュ
(142)

【図 2 5】



【手続補正書】

【提出日】平成28年11月22日(2016.11.22)

【手続補正1】

【補正対象書類名】特許請求の範囲

【補正対象項目名】全文

【補正方法】変更

【補正の内容】

【特許請求の範囲】

【請求項1】

システムメモリにアクセスするように構成されたメモリインターフェースと、
コントローラとを備え、前記コントローラが、

システムバスにわたって前記システムメモリ内で読み取られるべき第1の物理アドレス(PA)を含む第1のメモリ読み取り要求を受信することと、
前記受信された第1のメモリ読み取り要求を実施するために、前記システムメモリ内の前記第1のPAにアクセスすることとを行うように構成され、

前記コントローラが、

前記システムバスにわたって前記システムメモリ内で読み取られるべき次のPAを含む次のメモリ読み取り要求を受信することと、

前記受信された次のメモリ読み取り要求を実施するために、前記次のPAに基づいて前記システムメモリにアクセスすることと、

前記第1のPAに関連する前記システムメモリ内に前記次のPAを記憶することとを行うようにさらに構成される、メモリコントローラ。

【請求項2】

前記コントローラが、前記次のPAを、前記システムメモリ内の前記第1のPAと関連付けられたメモリブロック内のメタデータとして記憶するように構成される、請求項1に記載

のメモリコントローラ。

【請求項3】

前記コントローラが、

前記第1のPAと関連付けられた前記メモリブロックが、前記次のPAを記憶するために利用可能な空間を含むかどうかを決定することと、

前記第1のPAと関連付けられた前記メモリブロックが前記次のPAに対して前記利用可能な空間を含む場合に、前記次のPAを、前記システムメモリ内の前記第1のPAと関連付けられた前記メモリブロック内の前記メタデータとして記憶することとを行うように構成される、請求項2に記載のメモリコントローラ。

【請求項4】

前記第1のPAと関連付けられた前記メモリブロック内の前記利用可能な空間が、前記メモリブロック内に記憶される圧縮メモリデータのサイズに基づく、請求項3に記載のメモリコントローラ。

【請求項5】

前記コントローラが、

前記コントローラが現在のPAに対する読み取り動作を実施することに応答して、受信されたメモリ読み取り要求に対する前記現在のPAが、レポジトリ内に含まれるかどうかを決定することと、

前記次のPAが、前記レポジトリ内の前記受信されたメモリ読み取り要求に対する前記現在のPAに関連して記憶される場合に、

前記システムメモリ内の前記現在のPAと関連付けられた前記次のPAを読み取ることと、

前記次のPAにおいて前記システムメモリ内のメモリデータをプリフェッチし、前記次のPAにおいて記憶される前記メモリデータをプリフェッチされたメモリデータとして取得することとを行うようにさらに構成される、請求項1に記載のメモリコントローラ。

【請求項6】

前記コントローラが、前記システムバスにわたって前記プリフェッチされたメモリデータを供給するようにさらに構成される、請求項5に記載のメモリコントローラ。

【請求項7】

前記コントローラが、第1のPAエントリおよび対応する次のPAエントリをそれぞれが含む複数のキャッシュエントリを含み、前記メモリコントローラに通信可能に結合されたプリフェッチメタデータ(PMD)キャッシュ内の前記第1のPAと関連付けられたキャッシュエントリ内の前記次のPAを記憶するようにさらに構成される、請求項1に記載のメモリコントローラ。

【請求項8】

前記コントローラが、プロセッサコアとそれぞれが関連付けられた複数のPMDキャッシュの中の前記PMDキャッシュ内の前記第1のPAと関連付けられた前記キャッシュエントリ内に前記次のPAを記憶するように構成され、前記複数のPMDキャッシュの各々が、前記第1のPAエントリおよび前記対応する次のPAエントリをそれぞれが含む前記複数のキャッシュエントリを含み、前記PMDキャッシュが、前記メモリコントローラに通信可能に結合される、請求項7に記載のメモリコントローラ。

【請求項9】

前記コントローラが、

受信されたメモリ読み取り要求に対する現在のPAが前記PMDキャッシュ内に含まれるかどうかを決定することと、

前記現在のPAが前記PMDキャッシュ内に含まれる場合に、

前記コントローラが、前記現在のPAに対する読み取り動作を実施することに応答して、前記PMDキャッシュ内の前記現在のPAと関連付けられた前記次のPAを読み取ることと、

前記次のPAにおいて前記システムメモリ内のメモリデータをプリフェッチし、前記次のPAにおいて記憶される前記メモリデータをプリフェッチされたメモリデータとして取得

することとを行うようにさらに構成される、請求項7に記載のメモリコントローラ。

【請求項 10】

前記コントローラが、前記システムバスにわたって前記受信されたメモリ読取り要求に対する前記次のPAにおけるシステムメモリに記憶された前記プリフェッчされたメモリデータを供給するようにさらに構成される、請求項9に記載のメモリコントローラ。

【請求項 11】

前記受信されたメモリ読取り要求に対する前記現在のPAが前記PMDキャッシュ内に含まれない場合に、前記コントローラが、

前記システムメモリ内の前記現在のPAにアクセスして、前記システムメモリ内の前記現在のPA内に記憶された前記対応するメモリデータを取得することと、

前記プリフェッчされたメモリデータが記憶された前記PMDキャッシュ内の前記キャッシュエントリに対する前記現在のPAに対応する前記PMDキャッシュ内の前記キャッシュエントリを更新することとを行うようにさらに構成される、請求項9に記載のメモリコントローラ。

【請求項 12】

前記コントローラが、

前記PMDキャッシュ内に利用可能なキャッシュエントリがあるかどうかを決定することと、

前記PMDキャッシュ内に利用可能なキャッシュエントリがない場合に、前記PMDキャッシュ内の前記キャッシュエントリを立ち退かせて、前記立ち退かされたキャッシュエントリ内の前記現在のPAを前記PMDキャッシュ内に記憶することとを行うようにさらに構成される、請求項11に記載のメモリコントローラ。

【請求項 13】

アドレスエントリに関連するメモリデータを記憶するようにそれぞれが構成された複数のデータキャッシュエントリを含むデータキャッシュをさらに備え、

前記コントローラが、

前記現在のPAが、前記データキャッシュ内のデータキャッシュエントリ内のアドレスエントリと一致するかどうかを決定することと、

前記現在のPAが、前記データキャッシュの前記データキャッシュエントリ内の前記アドレスエントリ内に含まれる場合に、前記データキャッシュ内の前記現在のPAにおける前記システムメモリ内に記憶される前記対応するメモリデータを、前記受信されたメモリ読取り要求に対する前記プリフェッчされたメモリデータとして供給することとを行うようにさらに構成される、請求項9に記載のメモリコントローラ。

【請求項 14】

前記現在のPAが、前記データキャッシュの前記データキャッシュエントリ内の前記アドレスエントリ内に含まれる場合に、前記コントローラが、

前記PMDキャッシュ内の前記現在のPAと関連付けられた前記次のPAを読み取ることと、

前記次のPAにおいて前記システムメモリ内のメモリデータをプリフェッчし、前記次のPAにおいて記憶される前記メモリデータを、前記受信されたメモリ読取り要求に対する前記プリフェッчされたメモリデータとして取得することとを行うようにさらに構成される、請求項13に記載のメモリコントローラ。

【請求項 15】

前記現在のPAが、前記データキャッシュの前記データキャッシュエントリ内の前記アドレスエントリと一致しない場合に、前記コントローラが、

前記データキャッシュ内のデータキャッシュエントリを立ち退かせることと、

前記現在のPAにおける前記システムメモリからメモリデータを読み取って、前記メモリデータを前記データキャッシュ内に記憶することと、

次のPAが前記PMDキャッシュ内に含まれるかどうかを決定することと、

前記次のPAが前記PMDキャッシュ内に含まれる場合に、前記次のPAにおいて前記システムメモリからメモリデータを、前記受信されたメモリ読取り要求に対する前記プリフェッ

チされたメモリデータとしてプリフェッчすることとを行うようにさらに構成される、請求項13に記載のメモリコントローラ。

【請求項 16】

前記現在のPAが、前記データキャッシュの前記データキャッシュエントリ内の前記アドレスエントリと一致しない場合に、前記コントローラが、

前記PMDキャッシュ内の前記現在のPAと関連付けられた前記次のPAにアクセスすることと、

前記立ち退かされたデータキャッシュエントリがダーティである場合に、前記次のPAを、前記現在のPAと関連付けられた前記システムメモリ内に書き戻すことと、

前記PMDキャッシュ内の前記現在のPAと関連付けられた前記次のPAが、設定可能な確率によってダーティである場合に、前記次のPAを、前記現在のPAと関連付けられた前記システムメモリ内に書き戻すこととを行うようにさらに構成される、請求項15に記載のメモリコントローラ。

【請求項 17】

前記次のPAが前記PMDキャッシュ内に含まれない場合に、前記コントローラが、

前記システムメモリ内の前記現在のPAと関連付けられた前記次のPAを読み取ることと、

前記システムメモリ内の前記現在のPAと関連付けられた前記次のPAが有効である場合に、前記次のPAにおいて前記システムメモリからメモリデータを、前記受信されたメモリ読み取り要求に対する前記プリフェッчされたメモリデータとしてプリフェッчすることと、

前記次のPAを、前記現在のPAに関連する前記PMDキャッシュ内に書き戻すこととを行うようにさらに構成される、請求項14に記載のメモリコントローラ。

【請求項 18】

前記コントローラが、

第2のPAを含むメモリ書き込み要求を受信して、前記システムバスにわたって前記システムメモリ内に記憶されるべきメモリデータを書き込むことと、

前記第2のPAが、前記データキャッシュ内のデータキャッシュエントリ内のアドレスエントリと一致するかどうかを決定することと、

前記第2のPAが、前記データキャッシュの前記データキャッシュエントリ内の前記アドレスエントリ内に含まれる場合に、前記書き込みメモリデータを前記第2のPAと関連付けられた前記データキャッシュ内に記憶することと、

前記第2のPAが、前記データキャッシュの前記データキャッシュエントリ内の前記アドレスエントリ内に含まれない場合に、データキャッシュエントリを前記データキャッシュから立ち退かせて、前記書き込みメモリデータを前記第2のPAと関連付けられた前記データキャッシュ内に記憶することとを行わせるようにさらに構成される、請求項13に記載のメモリコントローラ。

【請求項 19】

前記第2のPAが、前記データキャッシュの前記データキャッシュエントリ内の前記アドレスエントリ内に含まれない場合に、前記コントローラが、

前記PMDキャッシュ内の前記第2のPAと関連付けられた前記次のPAにアクセスすることと、

前記立ち退かされたデータキャッシュエントリがダーティである場合に、前記次のPAを、前記第2のPAと関連付けられた前記システムメモリ内に書き戻すことと、

前記PMDキャッシュ内の前記第2のPAと関連付けられた前記次のPAが、設定可能な確率によってダーティである場合に、前記次のPAを、前記第2のPAと関連付けられた前記システムメモリ内に書き戻すこととを行うようにさらに構成される、請求項18に記載のメモリコントローラ。

【請求項 20】

集積回路(IC)に組み込まれる、請求項1に記載のメモリコントローラ。

【請求項 21】

セットトップボックス、エンターテインメントユニット、ナビゲーションデバイス、通

信デバイス、固定ロケーションデータユニット、モバイルロケーションデータユニット、モバイルフォン、セルラーフォン、コンピュータ、ポータブルコンピュータ、デスクトップコンピュータ、携帯情報端末(PDA)、モニタ、コンピュータモニタ、テレビジョン、チューナ、ラジオ、衛星ラジオ、音楽プレーヤ、デジタル音楽プレーヤ、ポータブル音楽プレーヤ、デジタルビデオプレーヤ、ビデオプレーヤ、デジタルビデオディスク(DVD)プレーヤ、およびポータブルデジタルビデオプレーヤからなるグループから選択されるデバイスに組み込まれる、請求項1に記載のメモリコントローラ。

【請求項22】

システムバスにわたってシステムメモリ内で読み取られるべき第1の物理アドレス(PA)を含む第1のメモリ読み取り要求を受信するための手段と、

前記受信された第1のメモリ読み取り要求を実施するために、前記システムメモリ内の前記第1のPAにアクセスするための手段と、

前記システムバスにわたって前記システムメモリ内で読み取られるべき次のPAを含む次のメモリ読み取り要求を受信するための手段と、

前記受信された次のメモリ読み取り要求を実施するために、前記システムメモリ内の前記次のPAに基づいて前記システムメモリにアクセスするための手段と、

前記第1のPAに関連する前記システムメモリ内に前記次のPAを記憶するための手段とを含む、メモリコントローラ。

【請求項23】

中央処理装置(CPU)ベースシステム内のシステムメモリに対するメモリ読み取り要求に対するデータを圧縮する方法であって、

システムバスにわたってシステムメモリ内で読み取られるべき第1の物理アドレス(PA)を含む第1のメモリ読み取り要求を受信するステップと、

前記第1のメモリ読み取り要求を実施するために、前記システムメモリ内の前記第1のPAにアクセスするステップと、

前記システムバスにわたって前記システムメモリ内で読み取られるべき次のPAを含む次のメモリ読み取り要求を受信するステップと、

前記受信された次のメモリ読み取り要求を実施するために、前記システムメモリ内の前記次のPAにアクセスするステップと、

前記第1のPAと関連付けられた前記システムメモリ内に前記次のPAを記憶するステップとを含む、方法。

【請求項24】

前記次のPAを、前記システムメモリ内の前記第1のPAと関連付けられたメモリブロック内のメタデータとして記憶するステップを含む、請求項23に記載の方法。

【請求項25】

前記第1のPAと関連付けられた前記メモリブロックが、前記次のPAを記憶するために利用可能な空間を含むかどうかを決定するステップと、

前記第1のPAと関連付けられた前記メモリブロックが前記次のPAに対して利用可能な空間を含む場合に、前記次のPAを、前記システムメモリ内の前記第1のPAと関連付けられた前記メモリブロック内の前記メタデータとして記憶するステップとを含む、請求項24に記載の方法。

【請求項26】

コントローラが現在のPAに対する読み取り動作を実施することに応答して、受信されたメモリ読み取り要求に対する前記現在のPAが、レポジトリ内に含まれるかどうかを決定するステップと、

前記次のPAが、前記レポジトリ内の前記受信されたメモリ読み取り要求に対する前記現在のPAに関連して記憶される場合に、

前記システムメモリ内の前記現在のPAと関連付けられた前記次のPAを読み取るステップと、

前記次のPAにおいて前記システムメモリ内のメモリデータをプリフェッチし、前記次

のPAにおいて記憶される前記メモリデータをプリフェッчされたメモリデータとして取得するステップとをさらに含む、請求項23に記載の方法。

【請求項 27】

第1のPAエントリおよび対応する次のPAエントリをそれぞれが含む複数のキャッシュエントリを含み、コントローラに通信可能に結合されたプリフェッチメタデータ(PMD)キャッシュ内の前記第1のPAと関連付けられたキャッシュエントリ内の前記次のPAを記憶するステップをさらに含む、請求項23に記載の方法。

【請求項 28】

コントローラが現在のPAに対する読み取り動作を実施することに応答して、前記受信されたメモリ読み取り要求に対する前記現在のPAが、前記PMDキャッシュ内に含まれるかどうかを決定するステップと、

前記現在のPAが前記PMDキャッシュ内に含まれる場合に、

前記コントローラが、前記現在のPAに対する前記読み取り動作を実施することに応答して、前記PMDキャッシュ内の前記現在のPAと関連付けられた前記次のPAを読み取るステップ、および

前記次のPAにおいて前記システムメモリ内の前記メモリデータをプリフェッチし、前記次のPAにおいて記憶された前記メモリデータを前記プリフェッチされたメモリデータとして取得するステップと、

前記システムバスにわたって前記受信されたメモリ読み取り要求に対して前記次のPAにおいて前記PMDキャッシュ内に記憶された前記プリフェッチされたメモリデータを供給するステップとをさらに含む、請求項27に記載の方法。

【請求項 29】

前記受信されたメモリ読み取り要求に対する前記現在のPAが前記PMDキャッシュ内に含まれない場合に、前記コントローラが、

前記システムメモリ内の前記現在のPAにアクセスして、前記システムメモリ内の前記現在のPA内に記憶された前記対応するメモリデータを取得するステップと、

前記プリフェッチされたメモリデータが記憶された前記PMDキャッシュ内の前記キャッシュエントリに対する前記現在のPAに対応する前記PMDキャッシュ内の前記キャッシュエントリを更新するステップと、

前記PMDキャッシュ内に利用可能なキャッシュエントリがあるかどうかを決定するステップと、

前記PMDキャッシュ内に利用可能なキャッシュエントリがない場合に、前記PMDキャッシュ内の前記キャッシュエントリを立ち退かせて、前記立ち退かされたキャッシュエントリ内の前記現在のPAを前記PMDキャッシュ内に記憶するステップとをさらに含む、請求項28に記載の方法。

【請求項 30】

システムバスと、

前記システムバスに通信可能に結合された少なくとも1つのCPUコアと、

前記システムバスに通信可能に結合されたメモリコントローラと、

前記システムバスに通信可能に結合され、メモリデータを記憶するように構成されたシステムメモリと、

アドレスエントリおよび対応する次の読み取りアドレスをそれぞれが含む複数のキャッシュエントリを含み、前記メモリコントローラに通信可能に結合されたプリフェッチメタデータ(PMD)キャッシュとを備え、

前記メモリコントローラが、

前記システムバスにわたって前記システムメモリ内で読み取られるべき第1の物理アドレス(PA)を含む第1のメモリ読み取り要求を受信することと、

前記受信された第1のメモリ読み取り要求を実施するために、前記システムメモリ内の前記第1のPAにアクセスすることとを行うように構成されたコントローラを備え、

前記コントローラが、

前記システムバスにわたって前記システムメモリ内で読み取られるべき次のPAを含む次のメモリ読み取り要求を受信することと、

前記受信された次のメモリ読み取り要求を実施するため、前記次のPAに基づいて前記システムメモリにアクセスすることと、

前記第1のPAに関連する前記システムメモリ内に前記次のPAを記憶することとを行うようにさらに構成される、中央処理装置(CPU)システム。

【国際調査報告】

INTERNATIONAL SEARCH REPORT		International application No PCT/US2015/031885						
A. CLASSIFICATION OF SUBJECT MATTER INV. G06F12/02 G06F12/10 ADD.								
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) G06F								
Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched								
Electronic data base consulted during the international search (name of data base and, where practicable, search terms used) EPO-Internal								
C. DOCUMENTS CONSIDERED TO BE RELEVANT <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: left; padding: 2px;">Category*</th> <th style="text-align: left; padding: 2px;">Citation of document, with indication, where appropriate, of the relevant passages</th> <th style="text-align: left; padding: 2px;">Relevant to claim No.</th> </tr> </thead> <tbody> <tr> <td style="text-align: center; padding: 2px;">X</td> <td style="padding: 2px;">US 2013/138867 A1 (CRAFT DAVID JONES [US] ET AL) 30 May 2013 (2013-05-30) figure 1 paragraph [0017] - paragraph [0019] -----</td> <td style="text-align: center; padding: 2px;">1-30</td> </tr> </tbody> </table>			Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.	X	US 2013/138867 A1 (CRAFT DAVID JONES [US] ET AL) 30 May 2013 (2013-05-30) figure 1 paragraph [0017] - paragraph [0019] -----	1-30
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.						
X	US 2013/138867 A1 (CRAFT DAVID JONES [US] ET AL) 30 May 2013 (2013-05-30) figure 1 paragraph [0017] - paragraph [0019] -----	1-30						
<input type="checkbox"/> Further documents are listed in the continuation of Box C.		<input checked="" type="checkbox"/> 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								
Date of the actual completion of the international search 12 August 2015		Date of mailing of the international search report 24/08/2015						
Name and mailing address of the ISA/ European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Fax: (+31-70) 340-3016		Authorized officer Filip, Liviu						

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No
PCT/US2015/031885

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2013138867 A1	30-05-2013	NONE	-----

フロントページの続き

(31) 優先権主張番号 14/716,108

(32) 優先日 平成27年5月19日(2015.5.19)

(33) 優先権主張国 米国(US)

(81) 指定国 AP(BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), EA(AM, AZ, BY, KG, KZ, RU, T, J, TM), EP(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, R, O, RS, SE, SI, SK, SM, TR), OA(BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG), AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, 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, H, N, HR, HU, ID, IL, IN, IR, IS, JP, KE, KG, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US

(72) 発明者 ナタラジャン・ヴァイディアナタン

アメリカ合衆国・カリフォルニア・92121・サン・ディエゴ・モアハウス・ドライブ・577
5

(72) 発明者 コリン・ピートン・ヴェリリ

アメリカ合衆国・カリフォルニア・92121・サン・ディエゴ・モアハウス・ドライブ・577
5

F ターム(参考) 5B005 LL19 MM01 NN22 PP03