

(19) 日本国特許庁(JP)

(12) 公開特許公報(A)

(11) 特許出願公開番号

特開2009-199384
(P2009-199384A)

(43) 公開日 平成21年9月3日(2009.9.3)

(51) Int.Cl.	F 1	テーマコード (参考)
G06F 9/48 (2006.01)	G06F 9/46 455D	5B033
G06F 9/34 (2006.01)	G06F 9/34 330	

審査請求 有 請求項の数 12 O L (全 16 頁)

(21) 出願番号 特願2008-40883 (P2008-40883)
(22) 出願日 平成20年2月22日 (2008.2.22)

(71) 出願人 000004237
日本電気株式会社
東京都港区芝五丁目7番1号
(74) 代理人 100124811
弁理士 馬場 資博
(72) 発明者 板橋 康雄
東京都港区芝五丁目7番1号 日本電気株式会社内
Fターム(参考) 5B033 AA01 AA15 DD04 DD09

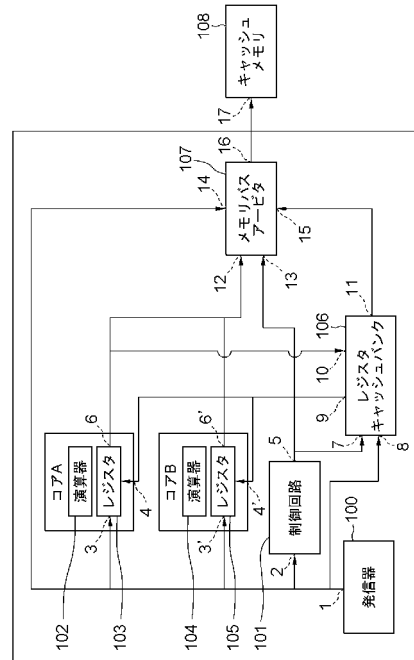
(54) 【発明の名称】 データ処理装置

(57) 【要約】

【課題】 処理の高速化を図ることができるデータ処理装置を提供すること。

【解決手段】 演算器と、当該演算器にて処理されるデータを格納するレジスタと、当該レジスタに格納されているデータの退避先となるキャッシュメモリよりもレジスタに対して高速にアクセス可能なレジスタキャッシュと、演算器がキャッシュメモリに対してアクセスしていないときにレジスタキャッシュに記憶されたデータをキャッシュメモリに記憶するメモリバスアービタと、レジスタに対するデータの入出力を制御する制御部と、を備え、上記レジスタキャッシュは、レジスタに格納されているデータの書き出し要求に応じて当該レジスタから出力されたデータを記憶する機能を有すると共に、レジスタへのデータの復元要求に応じてレジスタキャッシュ内に記憶されたデータのうち当該復元要求されたデータを選択してレジスタに復元する機能を有する。

【選択図】 図1



【特許請求の範囲】**【請求項 1】**

演算器と、当該演算器にて処理されるデータを格納するレジスタと、当該レジスタに格納されているデータの退避先となるキャッシュメモリよりも前記レジスタに対して高速にアクセス可能なレジスタキャッシュと、前記演算器が前記キャッシュメモリに対してアクセスしていないときに前記レジスタキャッシュに記憶されたデータを前記キャッシュメモリに記憶するメモリバスアービタと、前記レジスタに対するデータの入出力を制御する制御部と、を備え、

前記レジスタキャッシュは、前記レジスタに格納されているデータの書き出し要求に応じて当該レジスタから出力されたデータを記憶する機能を有すると共に、前記レジスタへのデータの復元要求に応じてレジスタキャッシュ内に記憶されたデータのうち当該復元要求されたデータを選択して前記レジスタに復元する機能を有する、
ことを特徴とするデータ処理装置。

10

【請求項 2】

前記レジスタキャッシュは、前記レジスタから出力されたデータを記憶するキャッシュ領域と、当該各キャッシュ領域にそれぞれ対応するタグ領域と、を有しており、

前記書き出し要求時に、前記レジスタから出力されたデータを前記キャッシュ領域に記憶すると共に、当該キャッシュ領域に対応する前記タグ領域に当該データを特定するデータ特定情報を記憶する機能と、

前記復元要求時に、前記タグ領域に記憶された前記データ特定情報に基づいて当該復元要求されたデータが記憶された前記キャッシュ領域を選択して、当該キャッシュ領域内のデータを前記レジスタに復元する機能を有する、
ことを特徴とする請求項 1 記載のデータ処理装置。

20

【請求項 3】

前記レジスタキャッシュは、前記書き出し要求時に、指定された書き出し先アドレス情報を前記データ特定情報として前記タグ領域に記憶し、前記復元要求時に、指定された復元要求アドレス情報と一致する前記書き出し先アドレス情報が記憶された前記タグ領域に対応する前記キャッシュ領域を選択して当該キャッシュ領域内のデータを前記レジスタに復元する、

ことを特徴とする請求項 2 記載のデータ処理装置。

30

【請求項 4】

前記レジスタキャッシュは、前記書き出し要求時に、前記レジスタから出力されたデータを記憶した前記キャッシュ領域に対応する前記タグ領域にデータ記憶フラグを記憶し、前記復元要求時に、前記データ記憶フラグが設定されている前記タグ領域に記憶された前記データ特定情報に基づいて当該復元要求されたデータが記憶された前記キャッシュ領域を選択する、

ことを特徴とする請求項 2 又は 3 記載のデータ処理装置。

【請求項 5】

前記タグ領域は、リングバッファである、

ことを特徴とする請求項 2 , 3 又は 4 記載のデータ処理装置。

40

【請求項 6】

前記レジスタキャッシュは、前記レジスタから出力されるデータを次に記憶する前記キャッシュ領域に対応する前記タグ領域を指定する出力先ポイントと、前記復元要求時に当該復元要求されたデータに対応する前記データ特定情報が記憶されているか否かを検索する対象となる前記タグ領域を指定する検索対象ポイントと、を有し、

前記復元要求時に、前記各タグ領域上を前記検索対象ポイントを走査させて、当該各タグ領域内に復元要求されたデータに対応する前記データ特定情報が記憶されているか否かを検索する、

ことを特徴とする請求項 2 , 3 , 4 又は 5 記載のデータ処理装置。

【請求項 7】

50

前記レジスタキャッシュは、前記復元要求時に、前記出力先ポインタが指定する次の前記タグ領域から前記検索対象ポインタを走査して、前記各タグ領域内に復元要求されたデータに対応する前記データ特定情報が記憶されているか否かを検索する、ことを特徴とする請求項 6 記載のデータ処理装置。

【請求項 8】

前記レジスタキャッシュは、前記タグ領域のうち前記データ特定情報が最も早く記憶された前記タグ領域を指定するデータ記憶ポインタを有しており、前記復元要求時に、前記データ記憶ポインタが指定する前記タグ領域から前記検索対象ポインタを走査して、前記各タグ領域内に復元要求されたデータに対応する前記データ特定情報が記憶されているか否かを検索する、ことを特徴とする請求項 6 記載のデータ処理装置。

10

【請求項 9】

前記レジスタキャッシュは、前記出力先ポインタあるいは前記データ記憶ポインタが指定する前記タグ領域を追い越すまで前記検索対象ポインタを走査して、前記各タグ領域内に復元要求されたデータに対応する前記データ特定情報が記憶されているか否かを検索する、ことを特徴とする請求項 7 又は 8 記載のデータ処理装置。

【請求項 10】

前記演算器及び前記レジスタにより構成されたコアを複数備えた、ことを請求項 1, 2, 3, 4, 5, 6, 7, 8 又は 9 記載のデータ処理装置。

20

【請求項 11】

演算器と、当該演算器にて処理されるデータを格納するレジスタと、当該レジスタに格納されているデータの退避先となるキャッシュメモリよりも前記レジスタに対して高速にアクセス可能なレジスタキャッシュと、前記演算器が前記キャッシュメモリに対してアクセスしていないときに前記レジスタキャッシュに記憶されたデータを前記キャッシュメモリに記憶するメモリバスアービタと、前記レジスタに対するデータの入出力を制御する制御部と、を備えたデータ処理装置におけるデータ処理方法であって、

前記レジスタに格納されているデータの書き出し要求に応じて当該レジスタから出力されたデータを前記レジスタキャッシュに記憶し、前記レジスタへのデータの復元要求に応じてレジスタキャッシュ内に記憶されたデータのうち当該復元要求されたデータを選択して前記レジスタに復元する、ことを特徴とするデータ処理方法。

30

【請求項 12】

前記書き出し要求時に、前記レジスタから出力されたデータを前記レジスタキャッシュ内に設けられたキャッシュ領域に記憶すると共に、当該キャッシュ領域に対応して前記レジスタキャッシュ内に設けられた前記タグ領域に当該データを特定するデータ特定情報を記憶し、

前記復元要求時に、前記タグ領域に記憶された前記データ特定情報に基づいて当該復元要求されたデータが記憶された前記キャッシュ領域を選択して、当該キャッシュ領域内のデータを前記レジスタに復元する、ことを特徴とする請求項 11 記載のデータ処理方法。

40

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、データ処理装置にかかり、特に、マイクロプロセッサに搭載されたレジスタに入出力されるデータを処理するデータ処理装置に関する。

【背景技術】

【0002】

プロセッサは、データを保持するレジスタを備えており、当該マイクロプロセッサの実

50

行時の性能を向上させるためには、レジスタの容量を増大することが有効である。ところが、プロセッサが複数のタスクを切り替えながら実行するマルチタスク環境下では、レジスタの状態をキャッシュメモリに記録するために要する時間が相対的に長くなる傾向にある。また、近年では、1つのCPUダイに複数のCPUコアを搭載するマルチコアプロセッサが増加している。しかしながら、このような場合には、コアを跨いだコンテキスト切り替えが発生した場合、レジスタに比べて低速なキャッシュメモリや外部メモリを経由して転送が行われるため、効率向上の阻害要因となっている。

【0003】

これに対し、特許文献1では、上記アクセス時間に低減を図るべく、プロセッサにメモリよりも高速アクセス可能なメモリであるレジスタ退避用スタックを備えている。これにより、レジスタの内容の退避と復帰を高速に行っている。

10

【0004】

【特許文献1】特開平9-22355号公報

【発明の開示】

【発明が解決しようとする課題】

【0005】

一方で、タイムシェアリングシステムにおいては、特定タスクによるプロセッサ利用の独占(スタベーション)を避けるため、優先順位が高いタスクが存在している状況で、優先順位の低いタスクに対してプロセッサの割り当て時間(タイムクォンタム)を意図的に割り当てることがある(エージング)。ところが、上述したスタックメモリを用いた先入れ後出し(FILO/スタック)のコンテキスト切り替えでは、エージングを行うことができないため、選択的にコンテキスト切り替えを行うことができない。従って、結果として、処理の高速化を図ることができない、という問題があった。そして、このことは、マルチコアプロセッサにおいて、コアを跨いだコンテキスト切り替えが発生した場合には、処理の遅延が顕著である。

20

【0006】

このため、本発明の目的は、上述した課題である、処理の高速化を図ることができるデータ処理装置を提供する、ことにある。

【課題を解決するための手段】

【0007】

そこで、本発明の一形態であるデータ処理装置は、

演算器と、当該演算器にて処理されるデータを格納するレジスタと、当該レジスタに格納されているデータの退避先となるキャッシュメモリよりもレジスタに対して高速にアクセス可能なレジスタキャッシュと、演算器がキャッシュメモリに対してアクセスしていないときにレジスタキャッシュに記憶されたデータをキャッシュメモリに記憶するメモリバスアービタと、レジスタに対するデータの入出力を制御する制御部と、を備え、

上記レジスタキャッシュは、レジスタに格納されているデータの書き出し要求に応じて当該レジスタから出力されたデータを記憶する機能を有すると共に、レジスタへのデータの復元要求に応じてレジスタキャッシュ内に記憶されたデータのうち当該復元要求されたデータを選択してレジスタに復元する機能を有する、という構成を採る。

40

【0008】

また、本発明の他の形態であるデータ処理方法は、

演算器と、当該演算器にて処理されるデータを格納するレジスタと、当該レジスタに格納されているデータの退避先となるキャッシュメモリよりもレジスタに対して高速にアクセス可能なレジスタキャッシュと、演算器がキャッシュメモリに対してアクセスしていないときにレジスタキャッシュに記憶されたデータをキャッシュメモリに記憶するメモリバスアービタと、レジスタに対するデータの入出力を制御する制御部と、を備えたデータ処理装置におけるデータ処理方法であって、

レジスタに格納されているデータの書き出し要求に応じて当該レジスタから出力されたデータをレジスタキャッシュに記憶し、レジスタへのデータの復元要求に応じてレジスタ

50

キャッシュ内に記憶されたデータのうち当該復元要求されたデータを選択してレジスタに復元する、
という構成を採る。

【発明の効果】

【0009】

本発明は、以上のように構成されることにより、レジスタから出力されたデータを、キャッシュメモリよりも高速アクセス可能なレジスタキャッシュから所望のデータを復元させることができ、処理の高速化を図ることができる、という優れた効果を有する。

【発明を実施するための最良の形態】

【0010】

本発明は、演算器とレジスタとを備えたマイクロプロセッサの処理をより高速にするものであり、特に、演算器及びレジスタにより構成されたコアを複数備えたマルチコアプロセッサに好適である。

【0011】

そして、本発明の一形態であるデータ処理装置は、

演算器と、当該演算器にて処理されるデータを格納するレジスタと、当該レジスタに格納されているデータの退避先となるキャッシュメモリよりもレジスタに対して高速にアクセス可能なレジスタキャッシュと、演算器がキャッシュメモリに対してアクセスしていないときにレジスタキャッシュに記憶されたデータをキャッシュメモリに記憶するメモリバスアービタと、レジスタに対するデータの入出力を制御する制御部と、を備え、

上記レジスタキャッシュは、レジスタに格納されているデータの書き出し要求に応じて当該レジスタから出力されたデータを記憶する機能を有すると共に、レジスタへのデータの復元要求に応じてレジスタキャッシュ内に記憶されたデータのうち当該復元要求されたデータを選択してレジスタに復元する機能を有する、
という構成を採っている。

【0012】

上記発明によると、まず、演算器にて処理されるレジスタに格納されたデータは、メモリバスアービタを介してキャッシュメモリに退避されるか、あるいは、当該キャッシュメモリよりも高速にアクセス可能なレジスタキャッシュに記憶される。なお、レジスタキャッシュに記憶されたデータは、メモリバスアービタによって演算器がキャッシュメモリに対してアクセスしていないときに当該キャッシュメモリに記憶される。

【0013】

そして、上記レジスタキャッシュは、レジスタに格納されているデータの書き出し要求があると、当該レジスタから出力されたデータを記憶する。その後、レジスタキャッシュは、レジスタへのデータの復元要求があると、当該復元要求されたデータがレジスタキャッシュ内に記憶されているか否かを検索し、該当するデータがあると、かかるデータを選択してレジスタに復元する。

【0014】

以上により、演算器は、レジスタから出力されたデータを、キャッシュメモリを介することなく、当該キャッシュメモリよりも高速アクセス可能なレジスタキャッシュから所望のデータを復元させることができる。従って、処理の高速化を図ることができる。特に、演算器とレジスタとによって構成されるコアを複数装備したマルチコアプロセッサに適用することで、コア間を跨ぐデータの移動がレジスタキャッシュを介して実行されるため、処理のさらなる高速化を実現できる。

【0015】

そして、具体的には、上記レジスタキャッシュは、レジスタから出力されたデータを記憶するキャッシュ領域と、当該各キャッシュ領域にそれぞれ対応するタグ領域と、を有しており、書き出し要求時に、レジスタから出力されたデータをキャッシュ領域に記憶すると共に、当該キャッシュ領域に対応するタグ領域に当該データを特定するデータ特定情報を記憶する機能と、復元要求時に、タグ領域に記憶されたデータ特定情報に基づいて当該

10

20

30

40

50

復元要求されたデータが記憶されたキャッシュ領域を選択して、当該キャッシュ領域内のデータをレジスタに復元する機能を有する、という構成を採る。

【0016】

また、上記レジスタキャッシュは、書き出し要求時に、指定された書き出し先アドレス情報をデータ特定情報としてタグ領域に記憶し、復元要求時に、指定された復元要求アドレス情報と一致する書き出し先アドレス情報が記憶されたタグ領域に対応するキャッシュ領域を選択して当該キャッシュ領域内のデータをレジスタに復元する、という構成を採る。

【0017】

さらに、上記レジスタキャッシュは、書き出し要求時に、レジスタから出力されたデータを記憶したキャッシュ領域に対応するタグ領域にデータ記憶フラグを記憶し、復元要求時に、データ記憶フラグが設定されているタグ領域に記憶されたデータ特定情報に基づいて当該復元要求されたデータが記憶されたキャッシュ領域を選択する、という構成を採る。そして、上記タグ領域は、例えば、リングバッファである。

10

【0018】

このように、レジスタキャッシュ内に記憶するデータを、当該データを特定するデータ特定情報を格納するタグを用いて管理することで、スタックメモリ方式とは異なり、任意のデータを選択して復元することが容易となる。

【0019】

さらに、具体的には、上記レジスタキャッシュは、レジスタから出力されるデータを次に記憶するキャッシュ領域に対応するタグ領域を指定する出力先ポイントと、復元要求時に当該復元要求されたデータに対応するデータ特定情報が記憶されているか否かを検索する対象となるタグ領域を指定する検索対象ポイントと、を有し、復元要求時に、各タグ領域上を検索対象ポイントを走査させて、当該各タグ領域内に復元要求されたデータに対応するデータ特定情報が記憶されているか否かを検索する。

20

【0020】

そして、上記レジスタキャッシュは、復元要求時に、出力先ポイントが指定する次の前記タグ領域から検索対象ポイントを走査して、各タグ領域内に復元要求されたデータに対応するデータ特定情報が記憶されているか否かを検索する。あるいは、レジスタキャッシュは、タグ領域のうちデータ特定情報が最も早く記憶されたタグ領域を指定するデータ記憶ポイントを有しており、復元要求時に、データ記憶ポイントが指定するタグ領域から検索対象ポイントを走査して、各タグ領域内に復元要求されたデータに対応するデータ特定情報が記憶されているか否かを検索する、という構成を採る。

30

【0021】

このとき、上記レジスタキャッシュは、出力先ポイントあるいはデータ記憶ポイントが指定するタグ領域を追い越すまで検索対象ポイントを走査して、各タグ領域内に復元要求されたデータに対応するデータ特定情報が記憶されているか否かを検索する、という構成を採る。このようにすることで、データの検索をさらに高速に実現できる。

【0022】

以下、本発明の具体的な構成及び動作を、実施形態にて説明する。なお、実施形態では、本発明であるデータ処理装置が2つのコアを有するマルチコアプロセッサである場合を例示して説明するが、コアの数は2つに限定されず、それ以上であってもよく、あるいは、1つのコアのみのプロセッサに適用してもよい。また、実施形態では、レジスタに入出力されるデータの一例として、コンテキストを例示しているが、これに限定されず、いかなるデータであってもよい。

40

【0023】

<実施形態1>

本発明の第1の実施形態を、図1乃至図4を参照して説明する。図1は、データ処理装置の構成を示すブロック図である。図2乃至図4は、レジスタキャッシュバンクの構成を示す概略図である。図5乃至図8は、データ処理装置の動作を示すフローチャートであ

50

る。図9乃至図14は、レジスタキャッシュバンクの具体的な構成及び動作を示す図である。

【0024】

[構成]

図1に示すように、本実施形態におけるデータ処理装置は、コアAとコアBとを備えたマルチコアプロセッサである。具体的に、データ処理装置は、発信器100と、制御回路101と、演算器102及びレジスタ103を有するCPUコアAと、演算器104及びレジスタ105を有するCPUコアBと、レジスタキャッシュバンク106と、メモリバスアービタ107と、を備えている。なお、上記発信器100は、本装置の外部に設けられていてもよい。また、本装置の外部に、メモリバスアービタ107を介してキャッシュメモリ108が接続されている。

10

【0025】

そして、上記発信器101からの信号は、当該発信器101に設けられた出力端子1より出力され、制御回路101の入力端子2、CPUコアA、Bにそれぞれ内蔵されている各レジスタ103、105の各入力端子3、3'、レジスタキャッシュバンク106の入力端子8、メモリバスアービタ107の入力端子14、にそれぞれ供給される。これにより、上記各回路は、発信器100からのクロック信号に同期して動作する。

【0026】

上記レジスタ103、105は、各レジスタが装備されているCPUコアA、Bの演算器102、104がそれぞれ演算を行うときに用いられるデータを保存する。

20

【0027】

上記制御回路101(制御部)は、上記レジスタ103、104に対するデータの入出力を制御する。具体的には、レジスタ103、105のデータを、レジスタキャッシュバンク106またはキャッシュメモリ108のどちらに書き出すかを制御する書き出し先制御信号と、書き出し先アドレスを指定する書き出し先アドレス信号とを、出力端子5から出力する。すると、制御回路101の出力端子5より出力された書き出し先制御信号と、書き出し先アドレス信号とは、メモリバスアービタ107の入力端子13と、レジスタキャッシュバンク106の入力端子7と、にそれぞれ入力される。

【0028】

また、各レジスタ103、105に格納されているデータは、各出力端子6、6'より出力され、メモリバスアービタ107の入力端子12と、レジスタキャッシュバンク106の入力端子10と、に入力される。

30

【0029】

そして、上記レジスタキャッシュバンク106は、その入力端子7に、制御回路101からの書き出し先制御信号により、書き出し先がレジスタキャッシュバンクである旨の信号を入力されると、当該信号が入力されたタイミングで入力端子10に入力されているレジスタから出力されたデータを、レジスタキャッシュバンク106内に保存する。なお、レジスタ103、105とレジスタキャッシュバンク106とは、十分に高速なバス、つまり、キャッシュメモリ108にアクセスするよりも高速なバスで接続されている。

【0030】

また、レジスタキャッシュバンク106は、それ固有のアルゴリズムにより入力されたデータを管理し、レジスタ103、105からのデータが記録しきれなくなると、キャッシュメモリ108に出力していない未出力のデータをキャッシュメモリ108に出力するまで待機する。なお、上述したレジスタキャッシュバンク106の動作は、予め組み込まれた回路によって実現されたり、あるいは、プログラムが組み込まれることによって実現されてもよい。

40

【0031】

また、メモリバスアービタ107は、CPUコアA、Bとキャッシュメモリ108との帯域に余裕ができ次第、つまり、CPUコアA、Bがキャッシュメモリ108にアクセスしていないときに、レジスタキャッシュバンク106に記憶されている上記未出力データ

50

を、キャッシュメモリ 108 に出力する。

【0032】

ここで、上記CPUコアA, Bは、現在の動作状況を保存・復元するために、全てのレジスタの内容をキャッシュメモリ 108 に出力する命令が実装される場合がある。この命令による処理に対して、上記レジスタキャッシュバンク 106 は、全てのレジスタ 103, 105 の内容を一時的に記録する。そして、レジスタキャッシュバンク 106 は、記録した全てのレジスタ 103, 105 からのデータを、制御回路 101 あるいはレジスタ 103, 105 がメモリバスアービタ 107 にデータを出力していない間に、キャッシュメモリ 108 に出力する。

【0033】

また、上述したように、メモリバスアービタ 107 は、制御回路 101 あるいはレジスタ 103, 105 からキャッシュメモリ 108 への入出力がある場合には、その動作を優先して実行する。一方で、それ以外の場合には、レジスタキャッシュバンク 106 に記憶されている未出力データを、キャッシュメモリ 108 に出力するよう作動する。

【0034】

このようにして、レジスタキャッシュバンク 106 に記憶されたレジスタ 103, 105 から出力されたデータは、制御回路 101、レジスタ 103, 105 の動作とは独立して、キャッシュメモリ 108 に出力される。

【0035】

次に、上記レジスタキャッシュバンク 106 の構成について、図 2 乃至図 4 を参照して説明する。まず、図 2 に示すように、レジスタキャッシュバンク 106 は、レジスタから出力されたデータを保存する「レジスタキャッシュ」(0~7)(キャッシュ領域)、各レジスタキャッシュに対応して用意され当該各レジスタキャッシュの有効/無効とキャッシュ先のアドレスを保持する「タグ」(0~7)(タグ領域)、を有する。また、レジスタから出力されるデータを次に書き込むレジスタキャッシュに対応するタグをポイント(指定)する「カウンタ1」(出力先ポインタ)、キャッシュメモリ 108 またはメインメモリに書き出しが完了している最新のタグをポイントしている「カウンタ2」(書き出し完了ポインタ)、レジスタキャッシュバンク 106 からレジスタ 103, 105 へ復元するデータを記憶しているレジスタキャッシュを検索するために走査するタグをポイントする「カウンタ3」(検索対象ポインタ)、を有する。ここで、上記タグは、リングバッファの形態を採っている。但し、タグは、必ずしもリングバッファ形態であることに限定されず、例えば、タグとレジスタキャッシュが関連付けられた構成であってもよく、さらには、必ずしもタグとレジスタキャッシュとを有していることにも限定されない。

【0036】

そして、かかる構成のレジスタキャッシュバンク 106 は、以下のように作動するよう構成されている。ここで、上述したように、発信器 100 の出力端子 1 から出力されたクロックパルス信号に同期して、制御回路 101 の出力端子 5 からは、キャッシュメモリ 108 に書き出すデータの出力元を制御する信号が出力される。この出力端子 5 からの信号は、レジスタキャッシュバンク 106 の入力端子 7 とメモリバスアービタ 107 の入力端子 13 に入力されることで、キャッシュメモリ 108 に書き出されるデータの出力元が、レジスタ 103, 105 かレジスタキャッシュバンク 106 かが決定される。つまり、レジスタ 103, 105 から出力されるデータの書き出し先が、レジスタキャッシュバンク 106 であるか、キャッシュメモリ 108 であるか、を指定する書き出し要求が出力される。また、制御回路 101 の出力端子 5 から出力されているデータの書き出し先アドレス情報は、レジスタキャッシュバンク 106 の入力端子 7 とメモリバスアービタ 107 の入力端子 13 に入力され、その内容は、キャッシュメモリ 108 にデータを書き出す際に、書き出し先アドレス情報(データ特定情報)として利用される。

【0037】

このように、制御回路 101 からレジスタキャッシュバンク 106 を書き出し先として指定する書き出し要求が出された場合には、レジスタキャッシュバンク 106 は、レジス

10

20

30

40

50

タ 1 0 3 , 1 0 5 からのデータを所定のレジスタキャッシュに記憶すると共に、このレジスタキャッシュに対応するタグに、「書き出し先アドレス」とデータが記憶されている旨を表すフラグとなる「有効」を記憶する。このとき、具体的には、「カウンタ 1」で指定されているタグに、上記「書き出し先アドレス」と「有効」を記憶し、当該タグに対応するレジスタキャッシュに、レジスタ 1 0 3 , 1 0 5 からのデータを記憶する。そして、「カウンタ 1」を一つ先のタグに進める。

【 0 0 3 8 】

また、レジスタキャッシュバンク 1 0 6 は、レジスタキャッシュに記憶されているデータがいっぱいになると、上述したようにメモリバスアービタ 1 0 7 によって、レジスタキャッシュに記憶されているデータをキャッシュメモリ 1 0 8 に書き出す。そして、「カウンタ 2」に、書き出しが完了したレジスタキャッシュに対応するタグを指定するよう設定する。

10

【 0 0 3 9 】

また、レジスタキャッシュバンク 1 0 6 は、制御回路 1 0 1 から、レジスタ 1 0 3 , 1 0 5 へのデータの復元要求があると、当該復元要求にかかるデータをレジスタキャッシュ 1 0 6 内から選択して、レジスタ 1 0 3 , 1 0 5 にロードして復元する機能を有する。具体的には、復元するデータを特定するアドレスとなる復元要求アドレスを受けると、「カウンタ 3」を「カウンタ 1」が指すタグの 1 つ先のタグを指定するよう設定し、このタグからタグ番号が増える順にタグを走査して、復元対象となるデータが格納されているレジスタキャッシュに対応するタグを検索する。なお、タグ番号 7 の次は、タグ番号 0 に戻って走査する。そして、各タグに対して、タグ内のフラグが「有効」であるか否かを調べ、「有効」であるタグ内に、復元要求アドレスと一致する書き出し先アドレスが存在するタグに対応するレジスタキャッシュに格納されたデータを選択する。そして、選択したデータを、レジスタ 1 0 3 , 1 0 5 に復元する。なお、「カウンタ 3」にてタグを走査して検索する処理は、例えば、目的のデータが選択されるまで、あるいは、「カウンタ 1」を指すタグを追い越すまで、順番にタグを走査して行われる。但し、上述した走査は、必ずしも上述したように「カウンタ 1」の 1 つ先のタグから開始されることに限定されず、いかなるタグから行ってもよく、また、いかなる順番で検索を行ってもよい。

20

【 0 0 4 0 】

[動作]

次に、上述したデータ処理装置の動作を、図 2 乃至図 8 を参照して説明する。図 2 乃至図 4 は、レジスタキャッシュバンク 1 0 6 の動作を説明する図であり、図 5 乃至図 8 は、各構成の動作を示すフローチャートである。

30

【 0 0 4 1 】

まず、図 2 は、レジスタキャッシュバンク 1 0 6 の初期状態を図示したものである。そして、上述したように、制御回路 1 0 1 からの書き出し要求があると、レジスタキャッシュバンク 1 0 6 のカウンタ 1 が指すタグが「無効」になっているレジスタキャッシュに（図 6 のステップ S 1 1 で No）、レジスタ 1 0 3 , 1 0 5 から出力されたデータを記憶する（図 6 のステップ S 1 2）。つまり、図 2 の例では、カウンタ 1 が指すタグ 0 に対応するレジスタキャッシュ 0 にデータを記憶し、そのタグのフラグを「有効」に更新する（図 6 のステップ S 1 3）。そして、その後、図 3 に示すように、カウンタ 1 を一つ先に進める（図 6 のステップ S 1 4）。一方、カウンタ 1 が指すタグのフラグが「有効」になっている場合には（図 6 のステップ S 1 1 で Yes）、レジスタキャッシュがいっぱいになっている可能性があるため、レジスタキャッシュの内容をキャッシュメモリ 1 0 8 に出力する（図 6 のステップ S 1 5）。

40

【 0 0 4 2 】

なお、上述したようにレジスタキャッシュバンク 1 0 6 からキャッシュメモリ 1 0 8 にデータを出力するときの動作は、例えば、図 7 のようになる。つまり、データがいっぱいになった場合には、カウンタ 2 が指すタグのフラグが「有効」であるかを調べ、「有効」であるタグに対応するレジスタキャッシュ内のデータを（図 7 のステップ S 2 1 で Yes

50

)、キャッシュメモリ108に出力する(図7のステップS22)。そして、このレジスタキャッシュに対応するタグのフラグを「無効」に更新し(図7のステップS23)、図4に示すように、カウンタ2を1つ進める(図7のステップS24)。

【0043】

また、上記レジスタキャッシュバンク106からキャッシュメモリ108へのデータの出力は、メモリバスアービタ107によって行われるが、このメモリバスアービタ107の動作を図5に示す。まず、上述した制御回路101からの指令により、レジスタ103、105からキャッシュメモリ108に書き出すデータがある場合には(図5のステップS1でYes)、当該レジスタ103、105からのデータをキャッシュメモリ108に出力する(図5のステップS2)。一方、レジスタ103、105からキャッシュメモリ108に書き出すデータがないときに(図5のステップS1でNo)、レジスタキャッシュバンク106からキャッシュメモリ108に出力するデータがある場合には(図5のステップS3でYes)、当該レジスタキャッシュバンク106からデータをキャッシュメモリ108に出力する(図5のステップS4)。

10

【0044】

次に、制御回路101あるいはCPUコアA、Bから、データの復元要求があった場合について説明する。まず、レジスタキャッシュバンク106は、カウンタ3にカウンタ1の先のタグのデータをコピーする(図8のステップS31)。そして、カウンタ3が指すタグのフラグが「有効」であるか否かを調べ(図8のステップS32)、「有効」であるタグに対して(図8のステップS32でYes)、当該タグに記憶されている書き出し先アドレスと、復元要求にかかるデータを特定する復元要求アドレスとが一致するか否かを調べる(図8のステップS33)。このとき、一致しない場合には(図8のステップS33でNo)、カウンタ3を1つ進めて(図8のステップS34)、上述同様に、次のタグの内容を調べる。そして、アドレスが一致するタグがあると(図8のステップS33でYes)、そのタグに対応するレジスタキャッシュに記憶されているデータを、レジスタ103、105に出力して復元する(図8のステップS36)。一方、復元要求にかかるデータがレジスタキャッシュバンク106に存在しない場合には(図8のステップS32でNo)、キャッシュメモリ108あるいはメインメモリから復元要求にかかるデータを読み出して、レジスタ103、105に復元する(図8のステップS35)。

20

【0045】

[動作の具体例]

次に、CPUコアA、Bで実行されるコンテキストが切り替わるときの動作を、図9乃至図14を参照して説明する。

30

【0046】

まず、レジスタキャッシュバンク106の初期状態を図9に示す。この状態では、レジスタキャッシュは空であり、タグのフラグも全て「無効」となっている。

【0047】

そして、今、CPUコアAまたはCPUコアBで実行されているコンテキストがWであり、コンテキストの切り替えが発生したとする。具体的に、コンテキストがWからXに変化したとき、コンテキストXのレジスタ103、105のデータは、出力端子6または6'からレジスタキャッシュバンク106の入力端子10とメモリバスアービタ107の入力端子12に入力される。ここで、制御回路101の出力端子5から、レジスタ103、105のデータ出力先がレジスタキャッシュバンク106である旨と、書き出し先アドレスとが、レジスタキャッシュバンク106の入力端子7とメモリバスアービタ107の入力端子13に出力されていることとする。すると、レジスタ103、105のデータは、レジスタキャッシュバンク106のカウンタ1が指し示すタグに対応するレジスタキャッシュにコピーされ、タグに書き出し先アドレスを記録してタグが「有効」であるフラグをセットする。これにより、図10に示すように、タグ0には、上記レジスタ103、105から出力されたデータの書き出し先アドレスと、当該タグ0が「有効」である旨のフラグが記憶され、また、カウンタ1が次のタグであるタグ1に進む。

40

50

【 0 0 4 8 】

その後、CPUコアA又はCPUコアBのコンテキストがXからさらにYに切り替わると、上述同様の処理が実行され、図11に示すように、タグ1に、「有効」のフラグと書き出し先アドレスが記憶される。そして、カウンタ1が次のタグであるタグ2に進む。さらに、CPUコアA又はCPUコアBのコンテキストがYからZに切り替わると、上述同様の処理が実行され、図12に示すように、タグ2に、「有効」のフラグと書き出し先アドレスが記憶される。そして、カウンタ1が次のタグであるタグ3に進む。このように、コンテキストが切り替わり、レジスタ103, 105からデータの出力がある度に、上記動作を繰り返す。

【 0 0 4 9 】

その後、CPUコアA又はCPUコアBのコンテキストがコンテキストYに切り替わり、データの復元要求があると、まず、図13に示すように、カウンタ3にカウンタ1の次のタグのデータがコピーされる。その後、カウンタ3の指し示すタグの有効・無効をチェックし、「有効」であれば、当該タグに格納されている書き出し先アドレスが復元要求されたコンテキストYの復元要求アドレスと一致するかどうかチェックする。このとき、タグ内の書き出し先アドレスが復元要求されたコンテキストYの復元要求アドレスと一致しない場合には、図14に示すように、カウンタ3を1つ進めて、再度タグの有効・無効とアドレスのチェックを行う。そして、タグが有効でアドレスが一致しているタグが見つかり、当該タグに対応するレジスタキャッシュのデータを選択して、出力端子9からレジスタ103, 105の入力端子4または4'にロードする。なお、上記タグの走査は、タグとアドレスが一致するか、カウンタ3がカウンタ1を追い越すまで行われる。そして、アドレスが一致するタグが見つからなかった場合には、レジスタキャッシュバンク106上に復元要求にかかるコンテキストがキャッシュされていないものとして、キャッシュメモリ108やメインメモリからレジスタにデータをロードする。

【 0 0 5 0 】

以上のように、本発明によると、演算器102, 104は、レジスタ103, 105から出力されたデータを、キャッシュメモリ108を介することなく、当該キャッシュメモリ108よりも高速アクセス可能なレジスタキャッシュバンク106から所望のデータを復元させることができる。従って、処理の高速化を図ることができる。特に、上述したようなCPUコアを複数装備したマルチコアプロセッサに適用することで、コア間を跨ぐデータの移動が、高速アクセス可能なレジスタキャッシュバンク106を介して実行されるため、処理のさらなる高速化を実現できる。さらに、レジスタキャッシュバンク106内で、レジスタキャッシュに格納するデータをタグを用いて管理することで、スタックメモリ方式とは異なり、任意のデータを選択して復元することが容易となる。

【 0 0 5 1 】

ここで、上述したレジスタキャッシュバンク106に、高速なメモリ、マルチポートRAM、非同期回路などを用いることで、より一層の高速化や低電力化を図ることができる。加えて、上記発信器100を外部に設置したり複数個設置したりすることでも、より一層の高速化や低電力化を図ることができる。

【 0 0 5 2 】

また、上記では、「カウンタ3」にて「カウンタ1」の次のタグを指定して、このタグから走査を開始する場合を例示したが、これに限定されない。例えば、現在「有効」となっているタグのうち、最も早く「有効」となったタグを記憶する「カウンタ4」（データ記憶ポインタ）をレジスタキャッシュバンク106に実装し、当該「カウンタ4」に「カウンタ3」を当てて、タグの走査を開始してもよい。例えば、「カウンタ4」は、カウンタ2の一つ先のタグを指定することで、現在記憶しているもののうち、最も早く「有効」となったタグを記憶することができる。そして、この場合には、「カウンタ3」が「カウンタ4」を追い越すまで検索が実行される。

【 産業上の利用可能性 】

【 0 0 5 3 】

本発明は、マルチコアプロセッサをはじめ、種々のプロセッサに利用することができ、産業上の利用可能性を有する。

【図面の簡単な説明】

【0054】

【図1】データ処理装置の構成を示すブロック図である。

【図2】レジスタキャッシュバンクの構成を示す概略図である。

【図3】レジスタキャッシュバンクの構成を示す概略図である。

【図4】レジスタキャッシュバンクの構成を示す概略図である。

【図5】データ処理装置の動作を示すフローチャートである。

【図6】データ処理装置の動作を示すフローチャートである。

10

【図7】データ処理装置の動作を示すフローチャートである。

【図8】データ処理装置の動作を示すフローチャートである。

【図9】レジスタキャッシュバンクの具体的な動作を説明する説明図である。

【図10】レジスタキャッシュバンクの具体的な動作を説明する説明図である。

【図11】レジスタキャッシュバンクの具体的な動作を説明する説明図である。

【図12】レジスタキャッシュバンクの具体的な動作を説明する説明図である。

【図13】レジスタキャッシュバンクの具体的な動作を説明する説明図である。

【図14】レジスタキャッシュバンクの具体的な動作を説明する説明図である。

【符号の説明】

【0055】

20

100 発信器

101 制御回路

102, 104 演算器

103, 105 レジスタ

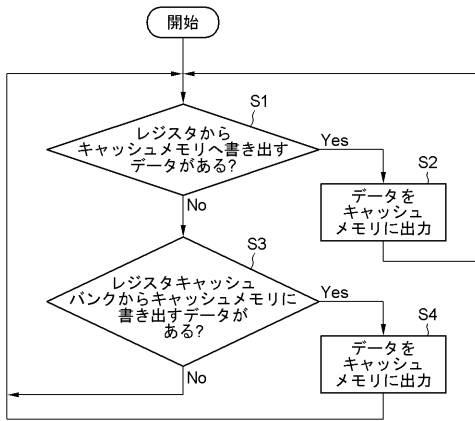
106 レジスタキャッシュバンク

107 メモリバスアービタ

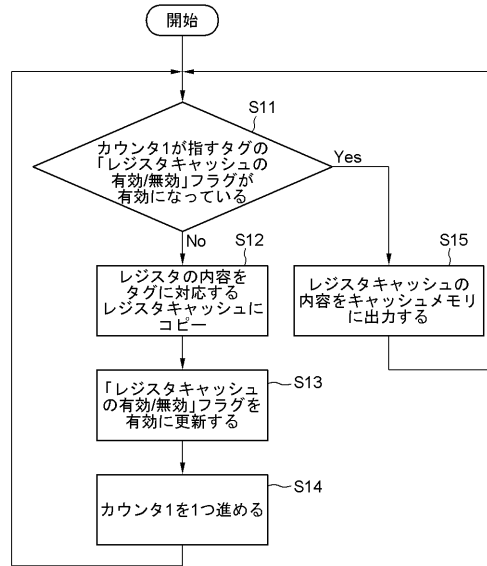
108 キャッシュメモリ

A, B CPUコア

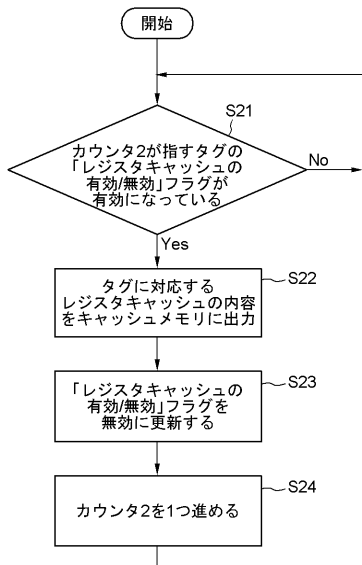
【 図 5 】



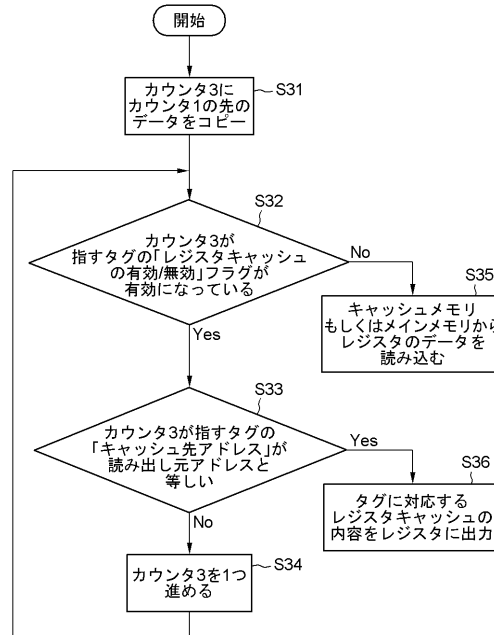
【 図 6 】



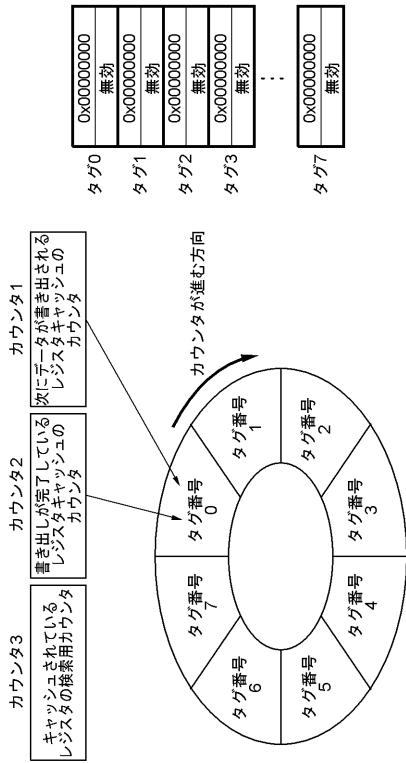
【 図 7 】



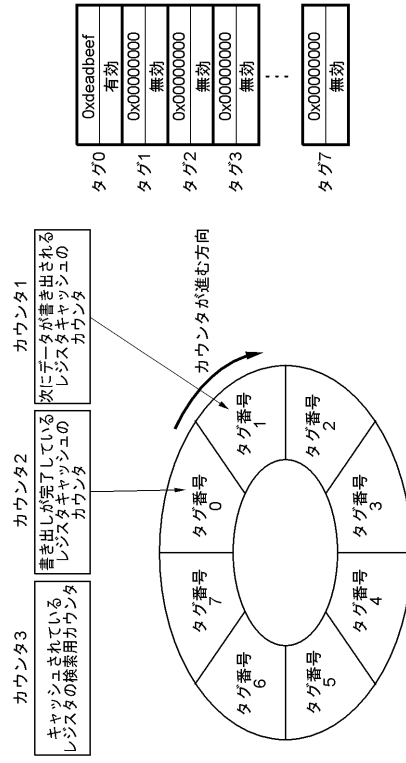
【 図 8 】



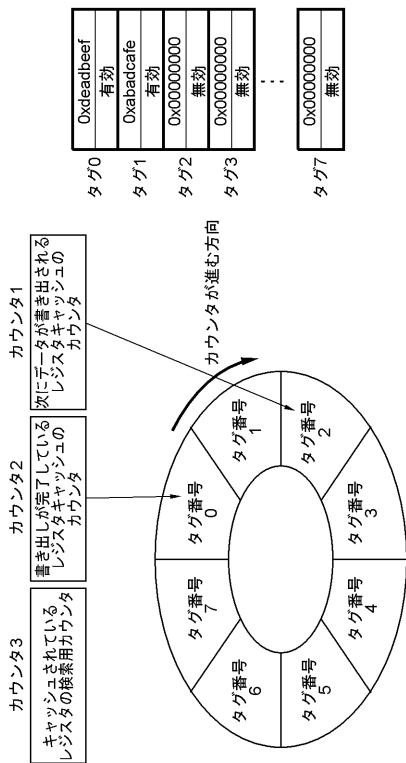
【 図 9 】



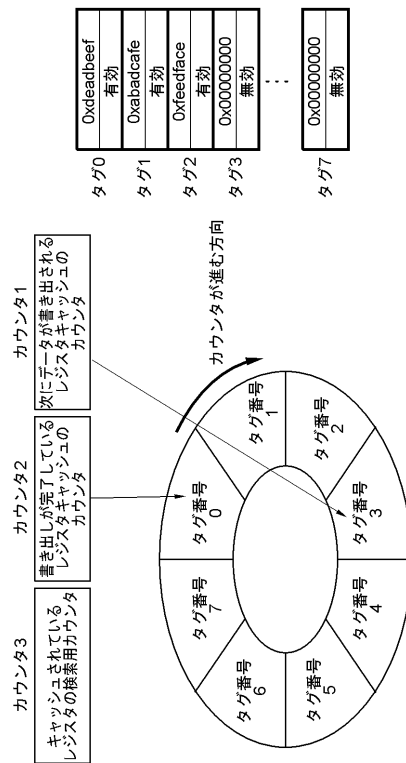
【 図 10 】



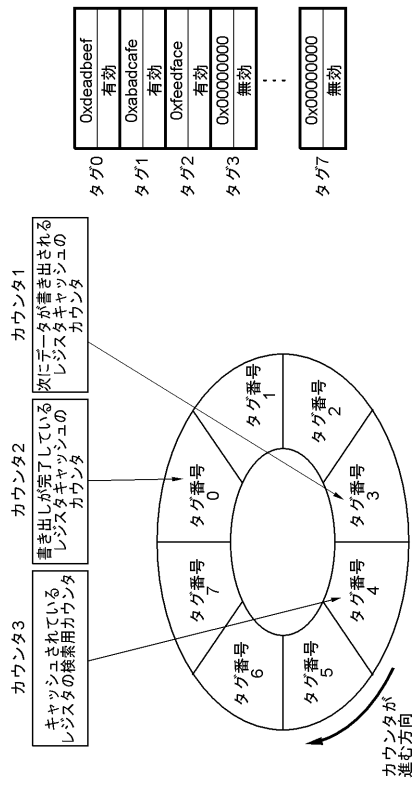
【 図 11 】



【 図 12 】

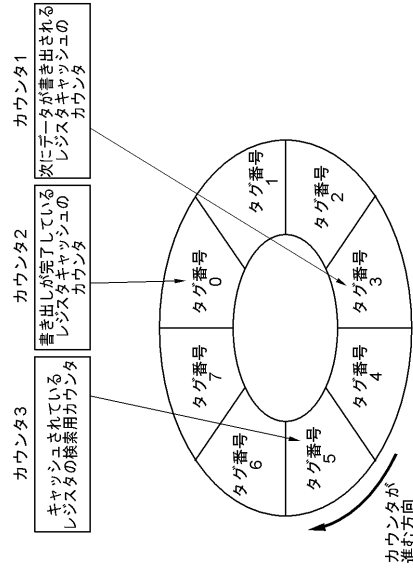


【 図 1 3 】



タグ0	Oxdeadbeef
タグ1	有効
タグ2	Oxabadcafe
タグ3	有効
タグ4	Oxfeedface
タグ5	有効
タグ6	Ox00000000
タグ7	無効

【 図 1 4 】



タグ0	Oxdeadbeef
タグ1	有効
タグ2	Oxabadcafe
タグ3	有効
タグ4	Oxfeedface
タグ5	有効
タグ6	Ox00000000
タグ7	無効