

ページング機構を用いたキャッシュレベルメモリの 管理に関する研究

3H-5

中澤 剛 前田 敦司 曾和 将容

電気通信大学 情報システム学研究科

1. はじめに

集積度の向上により最近のプロセッサでは、大容量のキャッシュメモリを持つものが増えてきている。既存のプロセッサでも数MBの容量のキャッシュをプロセッサのパッケージ内に持つものも存在し、将来的にキャッシュの容量はさらに増大すると考えられる。

このようにプロセッサに搭載されるキャッシュの容量は主記憶としてとらえるのに十分な大きさになりつつある。しかしながら、通常のキャッシュは主記憶とは異なりその管理に完全にハードウェア化された機構を用いている。このような機構は性能向上に貢献する一方、固定的なりプレースメントアルゴリズム、固定的な粒度を持ち、また、キャッシュの容量の増大に伴ってキャッシュの管理のためのハードウェア量が増大するという問題がある。

本稿ではプロセッサ上に高速な主記憶であるキャッシュレベルメモリ(以下CLMと略す)を搭載し、ページング機構によりそれを管理することによりキャッシュと同等の機能を実現する方式を提案する。この方式はキャッシュと比較してハードウェア量が抑えられるという特長がある。

2. 原理

ここでは簡単化のため、キャッシュはデータアレイ、タグ、コントロールの3つの要素から成るとモデル化を行う。ここでデータアレイは実際のデータを格納するための領域、タグは仮想または物理アドレスからデータアレイ上のデータをアドレスするための機構、コントロールはキャッシュミス発生時のデータの転送などを行う機構とする。

通常のキャッシュのタグ付けの機構では、一つのキャッシュのタグのエントリが一つのキャッシュラインに対応している。この方式は、粒度などの諸条件を固定すると、データアレイサイズが増大するに従ってタグのハードウェアサイズが線形的に増大する。

本研究で提案する方式は、このタグのハードウェアを一般のページング機構で用いられるTLBとページ表に置き換えた方式である。この方式では、仮想または物理アドレスからデータアレイ上へのマッピングは一般のメモリ空間上に記録されたページ表の方針に従う。TLBはそのページ表の一部をキャッシュし、実際にアドレス変換を行う機構である。

TLBはアドレス変換に必要な情報の一部しか持たないので、随時ページ表を参照する必要があり、このため一般のキャッシュにはないTLBミスのオーバーヘッドが課せられる。しかしながら、TLBはキャッシュとは異なり複数の粒度を選ぶことができ、TLBの一つエントリで複数のキャッシュラインに対するアドレス変換を行うことができる。このため、より少ないハードウェアで大きなメモリ領域を管理できる可能性がある。さらに、この方式は一般のメモリマネジメントと同じ機構により実現されているため、一般のメモリマネジメントのためのハードウェアと統合することも可能であり、このことによるハードウェア量の削減も可能になる。

Managing Cache level memory with paging mechanism

Tsuyoshi NAKAZAWA, Atusi MAEDA, Masahiro SOWA

University of Electro-Communications

1-5-1 Chofu-ga-oka, Chofu-city, Tokyo, Japan

3. 実現例と実装上の考慮点

実現例を図1に示す。CPUよりメモリアクセスリクエストが発生すると、まずTLBが仮想アドレスからCLM内アドレスを生成する。TLBがヒットした場合にはそのCLM内アドレスをもとにCLMがアクセスされCPUにレスポンスを返す。TLBがミスした場合、ページ表の参照により<仮想アドレス、CLM内アドレス、粒度>の対を生成しそれをTLBに格納する。この例ではオーバーヘッドを小さくするため、TLBのリフィルはハードウェアにより行うことを仮定する。もしCLMにデータがロードされていない場合にはCLMミスが発生し、CLMコントローラが外部メモリからCLMへのデータ転送を行う。なお、CLMミス時にはページ表より得られる<CLM内アドレス、物理アドレス、粒度>の対にしたがってデータの転送を行う。

キャッシュをCLMとページング機構で置き換える際にはいくつかの考慮すべき点が存在する。その中で特に対処すべき点はキャッシュでは存在しなかったTLBミスやページフォルトのオーバーヘッドの縮小である。

CLMのページングの粒度をキャッシュ並に細粒度で行えるように設定すると、TLBミスやページフォルトの発生率が增大することによりオーバーヘッドが増大し、また、ページ表サイズが大きくなってしまう。そこで、通常はより大きなページサイズ(~数KBのオーダー)でページを管理し、細粒度でのメモリ管理が必要な場合のみ細粒度のページ表を必要に応じて動的に生成し細粒度のページングを行う機構を導入する。この複数の粒度による管理の実現方法はいくつかの方法が考えられるが、もっとも簡単にはTLBの各エントリごとに連想ビットを可変にするようにTLBに拡張を施すことにより実現できる。

4. おわりに

本稿では大容量キャッシュをCLMとページング機構により置き換えることによりキャッシュに関わるハードウェアを縮小するための方法を提案した。大容量キャッシュをCLMとページング機構により置き換えるという考え方そのものは以上のようにシンプルなものであるが、しかしながら、性能上の問題を扱う上でページの粒度の選定や細粒度ページ表の構成方法、ハードウェアによる支援を行う範囲など考慮すべきトレードオフは多く、現在、どのような実現手法が適切であるかをシミュレーションを通して検討を進めている。

【参考文献】

[1]Bruce L. Jacob, Trevor N. Mudge : A Look at Several Memory Management Units, TLB-Refill Mechanisms, and Page Table Organizations., Proc. ASPLOS'98,pp.295-306,(Oct. 1998)

[2]D.R.Cherton, G.A. Slavenburg, and P.D.Boyle : Software-controlled caches in VMP multiprocessor., In Proc. ISCA-13,pp.366-374,(Jan. 1986)

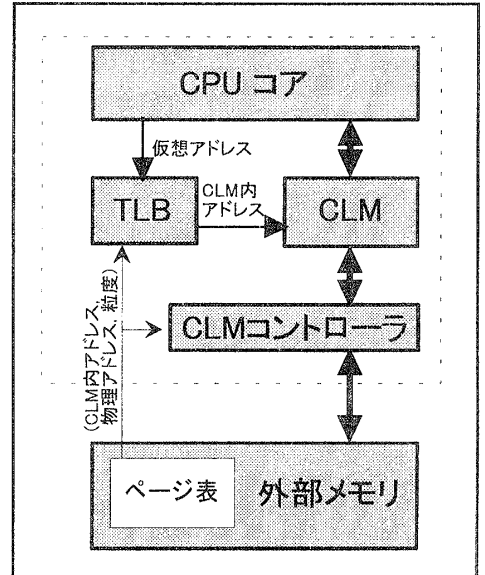


図1 実現例：CLMが大容量キャッシュにおけるデータアレイに相当し、TLBとページ表がキャッシュタグの機能を果たしている。なお、ここでは従来の主記憶を外部メモリと呼んでいる。