

頻出ビット列の圧縮によるキャッシュメモリのデータ利用効率の向上

落合裕也[†], 小林良太郎[†], 嶋田創[‡]豊橋技術科学大学[†] 名古屋大学[‡]

1. まえがき

現在のプロセッサにおいてメモリへのアクセスレイテンシが性能向上のボトルネックとなっている。プロセッサは使用するデータや命令をメインメモリから読み出す必要がある。しかし、メインメモリからの読み出しはプロセッサの動作速度と比較し非常に大きな時間が掛かってしまう。プロセッサはデータや命令を取得するまで動作が待たされる事となる。プロセッサの性能向上をさせるために動作周波数を上げたとしても、メインメモリから読み出しにかかる絶対時間は変化しない。そのためプロセッサの動作を高速化してもメモリのアクセス速度がボトルネックとなり性能向上の妨げとなる。

このボトルネックを軽減するため、メモリの階層化が行われている。階層化によりメモリはメインメモリとキャッシュメモリにより構成されている。メモリはそれぞれ、メインメモリはDRAM, キャッシュメモリはSRAMにより作られる。さらにキャッシュメモリは階層化されている。この階層化は一般的にL1キャッシュ, L2キャッシュにより構成される。また、メモリを階層化するだけでなく商用プロセッサにおいて様々なキャッシュメモリの新技術が提案されている。ARMの互換チップとして多くのスマートフォンやタブレット等のモバイル機器で使用される, Snapdragon S4やS800のCPUアーキテクチャであるKraitコアにはL0キャッシュメモリという省電力化のためのキャッシュメモリが搭載されている[1]。

キャッシュメモリはメモリ命令(Load命令やStore命令)がある度にアクセスされる。ハードウェアが自動的にキャッシュメモリへのデータ割り当てを行うため、プログラマーやコンパイラーは意識せずにキャッシュメモリを利用可能である。しかし、メモリ命令の度にキャッシュメモリにアクセスされるという事は、1アクセス当たりの消費電力は小さいとしても、アクセス回数が膨大であるために、キャッシュメモリ

の合計消費電力は大きくなってしまふ。キャッシュメモリの消費電力を削減するには、アクセス回数を減らす方法と、キャッシュメモリの規模を小さくし1アクセス当たりの消費エネルギーを削減する方法が考えられる[2][3]。

今回の研究ではデータを保存するデータキャッシュメモリの中のL1キャッシュメモリのサイズを削減し、1アクセス当たりの消費エネルギーの削減を目指す。

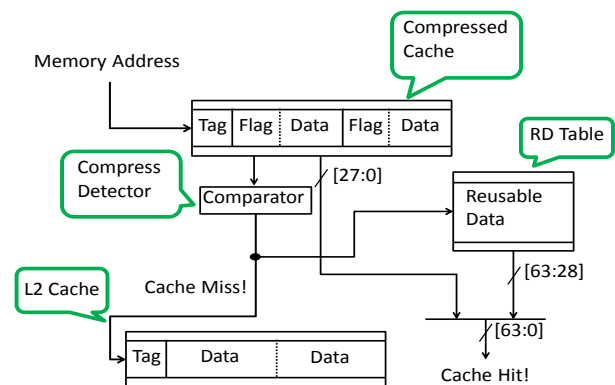


図1. 提案機構

2. 提案機構

2. 1. 方針・構成

ここでは提案機構の方針と構成について説明する。この機構ではプロセッサ内で頻繁に使用されるデータを圧縮する事でキャッシュメモリ内におけるデータの利用率向上を行う。データの上位ビットには頻繁に出現するビット列が存在しており、それらを圧縮する事でキャッシュメモリの回路規模を削減し、1アクセス当たりの消費エネルギーを削減する。提案機構の構成は図1である。従来手法と比較し、本研究では追加機構として圧縮検出器と頻出値テーブルを追加した。圧縮検出器は64bitのデータの上位ビットが圧縮可能なデータ判別し、対応するフラグを出力する。また、従来のキャッシュメモリではデータを64bit保存していたが、提案機構である圧縮キャッシュはデータを28bitしか保存しない。加えて、追加フィールドとして圧縮キャッシュは4bitのフラグを持つ。1wordの容量を半分削減したため、1ラインのサイズも半分へと削減可能となっている。従来は8wordを保

Improvement of Data Utilization Efficiency for Cache Memory by Compressing Frequent Bit Strings

[†]Hiroya OCHIAI, Ryotaro KOBAYASHI,

Toyohashi University of Technology

[‡]Hajime SHIMDA,

Nagoya University

存した場合、ラインサイズが 64 KB となったが、本機構を適用することで 8 word を保存しても 32 KB で済んでいる。フラグ比較器は従来と同様の動作で読み出されたタグが一致している場合、圧縮データを示すフラグである CF の判定を行う。

2. 2. 動作

ここでは提案機構の動作について説明する。まず始めに、プロセッサがあるアドレスの 64 bit のデータについてキャッシュメモリから読み出す動作について示す。そのアドレスから、タグ、インデックス、オフセットが生成される。インデックスを基にタグアレイとデータアレイの対応するラインへとアクセスされる。タグアレイからタグが読み出され、アドレスから生成されたタグとの比較が行われる。従来手法では、タグが一致した場合はキャッシュヒットと判定されるが、本研究の機構ではここではまだタグヒットと確定しない。その後、データアレイから圧縮により 28 bit となった下位ビットデータに相当するデータを読み出す。そのとき、提案機構では、圧縮データと同時に CF も読み出される。この CF には格納されているデータが圧縮されているのか、あるいは、圧縮不可能なデータであるのかを示すフラグが保存されている。この CF を CF 比較器で判定することで圧縮キャッシュのデータアレイに格納されていたデータが圧縮されているのか判明する。タグヒットでかつ圧縮データの場合、フラグからデータが復元可能であるため、キャッシュヒットとなる。

キャッシュヒットだった場合、CF の値は圧縮された上位ビットのデータを格納しているテーブルのインデックスも示している。このテーブルの中にはデータを復元するために必要な圧縮されたデータの上位ビットに相当する 36 bit データが格納されている。そして 64 bit のデータの下位ビットはデータアレイから読み出されたデータ、上位ビットはテーブルから読み出されたデータとなり復元が可能となる。それらを連結しプロセッサに渡す事で、キャッシュヒット時におけるキャッシュメモリの読み込みに関する動作を完了する。

一方、CF 判定器により圧縮不可能な値を示していた場合、タグが一致していても、格納されているデータが不完全であり復元が不可能なため、キャッシュミスとなる。その場合、キャッシュメモリは通常と同様のキャッシュミス動作を行い、下位層へのメモリへと問い合わせやキャッシュラインの入れ替えを行う。

3. 評価

評価にはSimCoreをSuperScalar化し、提案機構を適用したシミュレータを使用する[4]。その測定結果とCacti 6.5を用いて消費電力を算出する。図2は提案モデルにおけるキャッシュメモリの消費電力削減率の結果である。図の縦軸は従来機構に対するDL1・共有L2キャッシュの消費電力削減率であり、横軸はベンチマークとその平均を示す。

この図より、libquantum以外のベンチマークで従来機構に対し消費電力を削減できている事が分かる。従来機構から平均して9.2%の電力削減を達成した。最大で42.4%と消費電力削減を行う事が出来た。これは提案機構が圧縮キャッシュを搭載し、従来機構のL1 キャッシュに比べて1アクセス当たりの消費エネルギーを削減した事による。

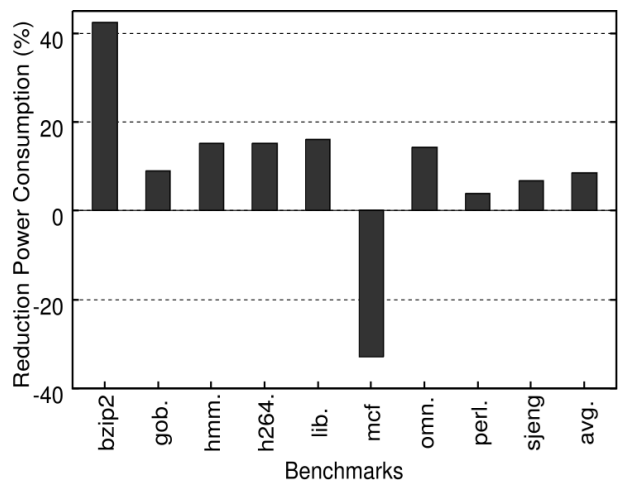


図2. 測定結果

謝辞 本研究の一部は、JSPS 科研費 25330060 および 26330063 の支援により行った。

文 献

- [1] P. Yongtae, et al., "Demo: Software Defined Radio: On a Smartphone, As an App!," MobiCom '14, pp. 313-316, 2014.
- [2] 岡慶太郎ほか, "データ値の局所性を利用したイン共有キャッシュ," 情報処理学会論文誌コンピュータインテグレーションシステム(ACS), Vol. 5, No. 4, pp. 36-47, 2012.
- [3] J. Kin, et al., "The filter cache: an energy efficient memory structure," In Proc. MICRO-30, pp. 184-193, 1997.
- [4] K. Kise, et al., "SimAlpha Version 1.0: Simple and Readable Alpha Processor Simulator," LNCS, Vol. 2823, pp. 122-136, 2003.