

SIMD 拡張ソフトコアプロセッサのための効率的なメモリシステムの検討

菊池智也[†] 大津金光^{††} 大川猛^{††} 横田隆史^{††}[†] 宇都宮大学工学部情報工学科 ^{††} 宇都宮大学大学院工学研究科情報システム科学専攻

1 はじめに

FPGA はハードウェアリソース量の増加や動作速度の向上, 消費電力の低下などの要因によって組み込みシステムで利用される機会が増加している. FPGA を利用した開発は一般的に難易度が高く, 開発期間が長くなることが多い. しかし, データ並列性が高いプログラムを対象としたシステムを開発する場合は, SIMD 演算や自動並列化コンパイラの手助けを借りることで比較的短期間で高性能化が達成できると考えられる.

そこで我々は, SIMD 演算機能を拡張したソフトコアプロセッサである MIQS (MIPS Instruction processor with Quadword SIMD extension)[1] を開発している. MIQS はオンチップメモリを使用することで, オフチップメモリへのアクセス回数を削減して高効率なメモリアクセスを実現している. しかし, 使用できるオンチップメモリには限りがあるため, 依然としてオフチップメモリの性能はシステム全体の性能に大きな影響を与える. オフチップメモリの性能を十分活用するための機能が求められるが, MIQS にはそのための機能が未だ不十分である. そこで MIQS のオフチップメモリの性能を十分に引き出すために, オフチップメモリへのメモリアクセスを高速に行うための機能を組み込んだ効率的なメモリシステムを検討する.

2 MIQS

MIQS とは 4 並列 SIMD 演算機能とオンチップメモリを備えたソフトコアプロセッサである. MIPS 互換命令を実行する MIPS サイドと SIMD 演算命令を実行する SIMD サイドから構成されており, MIPS サイドでは 1 ワード (32bit), SIMD サイドでは 4 ワード (32bit × 4) のデータをそれぞれ処理する.

また, MIQS は処理するデータをアクセスレイテンシが小さいが小容量なオンチップメモリと, アクセスレイテンシが大きいが大容量なオフチップメモリに分けて格納している. できる限りレイテンシが小さいオンチップメモリへアクセスをすることで, レイテンシが大きいオフチップメモリへのアクセスを削減し, 低いメモリ性能を隠蔽している.

しかし, 依然としてオフチップメモリへのアクセスが無くなることはないため, システム全体の性能を向上させるためにはオフチップメモリの性能を十分に引

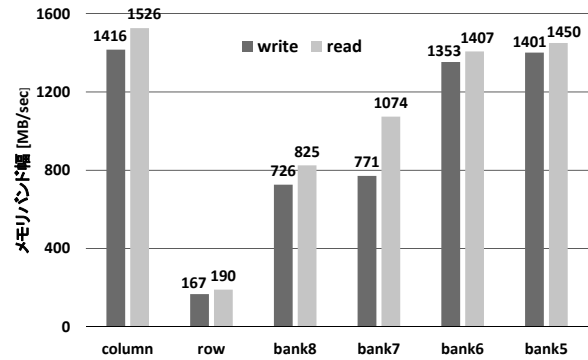


図 1: メモリバンド幅の測定結果

き出す必要がある. つまり, オフチップメモリに対して発生するメモリアクセスをできる限り高速に行うための仕組みが求められる.

3 SDRAM の初期評価

どのようなメモリアクセスの場合に高速に行えるかを調べるために, アクセスパターンごとのメモリバンド幅を比較した. メモリは, 標準的な FPGA ボードである Xilinx 社の Nexys Video に搭載されている DDR3 SDRAM を使用した. このメモリのデータバス幅は 16bit であり, 動作周波数は 400MHz として計測を行った. なおメモリバンド幅は, メモリコントローラへ連続してアクセス命令を発生させる回路を作成して, メモリアccessに掛かったクロック数を計測することで算出した.

SDRAM のメモリアドレスはバンク, ロウ, カラムの 3 つの要素から構成されている. バンクアドレスで該当するバンクを選び, ロウアドレスでアクセス対象が含まれる 1 行分のデータを読み出し, カラムアドレスでその行データの 1 つにアクセスする. なお, 同一バンクの同一行に対して連続してアクセスする際には, ロウアドレスの指定が省略できる. また, 同一バンクは 1 度に 1 つのアクセスしか受け付けないが, 異なるバンクに対しては並列してアクセスを行うことができる. つまり, DDR3 SDRAM への連続するメモリアccessのパターンとして「同一行へのアクセスが異なる行へのアクセス」, 「単一バンクへのアクセスが複数バンクへのアクセス」がある. よって, それぞれのメモリアccessパターンを測定, 比較, 評価した.

図 1 にバンド幅の測定結果を示す. なお, column は同一行内への連続アクセス, row は異なる行への連続アクセス, bank8~5 は複数バンクに対する連続したメモリアccessのことを表している. バンクの数字は, 利

Consideration of efficient memory system for SIMD-extended soft-core processor

[†]Tomoya Kikuchi, ^{††}Kanemitsu Ootsu, ^{††}Takeshi Ohkawa, ^{††}Takashi Yokota

Department of Information Science, Faculty of Engineering, Utsunomiya University ([†])

Department of Information Systems Science, Graduate School of Engineering, Utsunomiya University (^{††})

用したバンク数がそれぞれ8~5つであることを示している。column は理論バンド幅 1.6GB/sec に対してバンド幅の利用率は最大 95.4%であり、row のバンド幅の利用率は最大 11.9%であった。また、複数バンクへのメモリアクセスにおけるバンド幅の利用率は、バンクの最大数である8つの場合は最大 51.5%、7つの場合は最大 67.1%、6つの場合は最大 88.0%、5つの場合は最大 90.6%であった。

4 効率的なメモリシステムの検討

評価結果を踏まえると、メモリアクセスを効率的に行うためにはできる限り異なるロウアドレスへのメモリアクセスを避けるべきである。そこで、同一行へのアクセスや異なるバンクへの連続したメモリアクセスを行うメモリシステムを目指して、ハードウェアプリフェッチャとストアバッファの導入を検討した。

4.1 ハードウェアプリフェッチャ

一般的に、キャッシュを利用する場合はキャッシュミス率を下げるために、プリフェッチ機構 [2] を用いる。プリフェッチには大きく分けてソフトウェアプリフェッチと、ハードウェアプリフェッチの2種類がある。ソフトウェアプリフェッチと比較して高速化が簡単であるという点から、ハードウェアプリフェッチャの検討を行った。

ハードウェアプリフェッチのプリフェッチ方式は、シーケンシャルプリフェッチを選択することが適切であると考えられる。シーケンシャルプリフェッチは基本的に、あるブロックがアクセスされた時に隣接するブロックのプリフェッチを開始するような方式である。このようなプリフェッチが連続して発生すると、同一バンク内の同一行に対するアクセスが連続して行われるため、高速なメモリアクセスが行えると考えられるためである。

また、シーケンシャルプリフェッチ以外にも、配列データを扱うようなプログラムの場合にはストライドプリフェッチなどの方式も高速なメモリアクセスに繋がると考えられる。

4.2 ストアバッファ

MIQS にはデータを格納するためのオンチップメモリがあり、このオンチップメモリはキャッシュメモリやスクラッチパッドメモリとして使用される。オンチップメモリからはオフチップメモリに対してブロック単位でのストアが発生するが、連続するストアの場合でもその順番を変えることはできない。そこで、オンチップメモリからオフチップメモリへストアされるブロックを一時的にバッファして、高速にメモリアクセスを行えるようにストアする順番を管理するようなストアバッファの検討を行った。

図2は検討しているストアバッファの構造を示したものである。ストアバッファはオンチップメモリからブロックが送られてきたら、アドレス情報(タグやイン

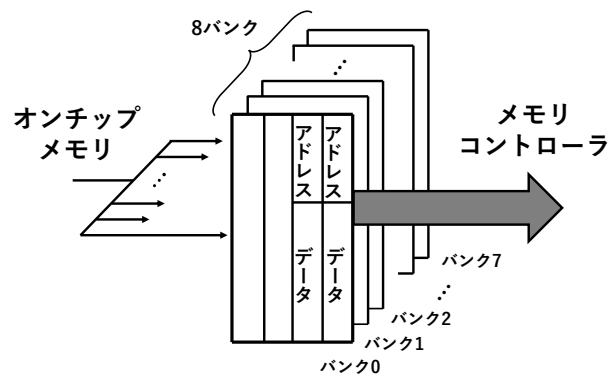


図 2: ストアバッファの構造

デックス) を参照してストア先のバンクアドレスごとにブロックを分けて管理する。複数のブロックが保持されていて、それらをストアをする際には、異なるバンクアドレスに連続して行うことで高速なアクセスを実現する。また、ある1つのバンクに対してブロックが2つ以上ある場合、先頭の2ブロックのアドレス情報を比較して同一行へのアクセスである時には、連続してストアさせる。ただし比較処理はむしろボトルネックになる可能性があるため、注意する必要がある。

5 おわりに

本稿では、MIQS システムのオフチップメモリに対する効率的なメモリアクセスのための初期評価として、SDRAM のメモリ性能をメモリアクセスパターンごとに評価した。その評価を元に、ストアバッファとハードウェアプリフェッチャを用いた効率的なメモリシステムについて検討した。各機構を導入することで、オフチップメモリの性能をより引き出すことが可能になる。また、オフチップメモリへのアクセス順を、カラムアドレスまたは異なるバンクアドレスに対して連続になるように管理することで、効率的なメモリアクセスが可能になると考えられる。

謝辞

本研究は、一部 JSPS 科研費 15K00068, 16K00068, 17K00072 の助成による。

参考文献

- [1] 平石 康祐, 橋本 瑛大, 大津 金光, 大川 猛, 横田 隆史: “SIMD 命令拡張による高速化の検討 – プロセッサ設計コンテストを題材として –”, 信学技報, Vol.114, No.345 (CPSY2014-85), pp.1-6, 2014.
- [2] Steven P. Vanderwiel, David J. Lilja: “Data Prefetch Mechanisms”, ACM Computing Surveys, Vol.32, No.2, pp.174-199, June 2000.