

プロセッサ設計コンテスト提出データの概要

児島彰[†]

プロセッサ設計コンテストで提出するデータの概要を示す。今回の提出データは、ベースのプロセッサに Xilinx の MicroBlaze を使用し、コンテストの要求仕様に適合する周辺回路を作成し、ソフトウェアのアルゴリズムの改善やデータのメモリ配置の改善などで、速度向上を行っている。さらに加速させるための拡張ハードウェアも計画しているが、現時点では完成に至っていない。

Summary of Preliminary Contest Data

AKIRA KOJIMA[†]

This manuscript describes our preliminary contest data. We use Xilinx MicroBlaze as base processor and attach peripheral I/O for the contest. At this moment, our performance improvement mainly owes to software tuning. We are planning hardware accelerator to improve performance more.

1. はじめに

コンテストで提出するプログラムと設計データの概要について述べる。今回のコンテストでは予めリファレンス設計として MIPS プロセッサを使ったシステムが提供されている。これで動く4つのプログラムが与えられて、提出する設計データは、決められた入力データに対して、リファレンスと完全に同じデータを出力することが要求されている。新規にプロセッサやコンパイラを作成するのが、本来のコンテストの意図するところであるが、リファレンスで提供されているプロセッサや既存のプロセッサ、コンパイラを利用して、ソフトウェア中心の改良を行うことも認められている。今回コンテストの期間が短いため、当方では新規のプロセッサやコンパイラの開発は行わず、これらは既存のものを利用し、ソフトウェアの改良とメモリ回路の改良を中心に行うことにした。ハードウェア設計環境とソフトウェア設計環境として、Xilinx 社の XPS と SDK を利用する。ベースのプロセッサとして Xilinx 社が提供する MicroBlaze[1]を利用し、コンテストの要求仕様に適合する周辺回路を作成し、使用メモリの配置の工夫や、ソフトウェアのアルゴリズム改善で性能向上を目指す。時間に余裕があれば、さらに実行を加速するための拡張ハードウェアを付加していく予定である。今回のコンテストは、予選データの提出日、本稿の提出日、本戦データの提出日が、約一週間の間隔である。本稿は、予選で提出したデータの説明に加えて、若干の追加の改良が行われた点を含んでいる。本戦までは時間があるので、更に改良を目指す。

2. 課題プログラム

本コンテストでは、参加者全員が同一の Xilinx 社の FPGA の Spartan6 を搭載した Digilent 社 Atlys を使用し、「整数ソート」、「行列積」、「ステンシル計算」、「最短経路問題」の4つの課題プログラムを実行し、実行時間を競う。本節では、それぞれの課題プログラムに対して、我々が行った性能改善の方法について述べる。

2.1 整数ソート

最大 4194304 要素(4M×8B=16MB)の、32 ビット整数(自然数)をソートする。リファレンスでは再帰関数を使用したクイックソートのプログラムが提供されていた。提出したプログラムではソートのアルゴリズムを変更し、関数呼び出しがないアルゴリズムを使用した。元のプログラムのデータ領域以外に作業メモリを使用し、オーバ・ヘッドもあるが、最大数のデータで速くなるように $O(n)$ の基数ソートのアルゴリズムを採用した。クイックソートは $O(n \log n)$ なので、データ数が多いと有利になる。今回使用した基数は 0x10000 (16 ビット)で、クイックソートなど内部ソートにはない作業メモリでの追加操作が必要であるが、(符号なし)32 ビットの数のソートであれば、データ転送操作は2回でソートは完了する。

2.2 行列積

コンテストのルールの指定では、最大 $n=512$ の正方行列の行列積の計算を行う。拡張ハードウェアを使い演算を加速させる予定であったが、時間の都合で、予選のデータ提出には間に合わなかった。提出したデータでは、ソフトウェア上の改善のみで性能向上させてある。具体的には、配列のアクセス順序の変更と8回ループアンローリング、ポインタアクセスへの変更を行った。実行時間を調べると、乗算の演算時間(約 13%)よりも、メモリアクセスなどその

[†] 広島市立大学
Hirosima City University

他の時間が大きなウェイトを占めることがわかった。キャッシュのヒット率を上げることが重要である。

2.3 ステンシル計算

課題のステンシル計算は、着目する点の周辺の9点のデータの平均を取り、平均で着目点を更新する処理である。我々のプログラムでは、周辺データの加算計算を3点ずつグループにして計算を行い、隣接する3つのグループをパイプライン的にデータ保持するアルゴリズムに変更することで、速度向上させている。但し、コンテストのルールで指定されている最大の $n=512$ 、最大の繰り返し 100000 回では、全く時間が足りず、時間に切れになる。また、この計算は、パイプライン的な動作が行い易い構造をしているので、現時点では試していないが、マルチプロセッサを使用するとさらに加速できる可能性がある。

2.4 最短経路問題

最大 2048 ノード、最大 8192 エッジのグラフが与えられ、指定された2つのノードの最短経路を探索する問題で、リファレンスにはダイクストラ法で解を求めるプログラムが与えられている。最短経路問題は、リファレンスプログラムの段階から、アルゴリズムが複雑で、探索順序の変更を伴うようなアルゴリズムの変更を行うと、同一長さの解が2つ以上見つかった場合にどれを出力するかなどが問題になり、細かい点まで、常にリファレンスのプログラムと同じ結果を出すのが難しくなる。今回は開発時間が不足しているので、アルゴリズムの大きな変更は行わず、メモリ配置の調整など僅かな変更のみに留めた。最も頻繁に使用するノード・アレイのデータを、アクセス時間の短い BlockRAM に配置することで速度向上を行った。

3. 性能評価

性能評価のために、「リファレンスプロセッサを使った場合」、「MicroBlaze を単純に使った場合」、さらに「メモリの使用方法やソフトウェア上の工夫を行った場合」の、3つの場合の実行時間を計測した。XPS のコンパイラは ISE 14.7 の MicroBlaze 用 gcc4.6.4 である。リファレンスのコンパイラは MIPS 用 gcc4.3.3 である。計測にはコンテストで提供されているプログラム psend を使用した。

MicroBlaze の命令セットアーキテクチャは MIPS の命令セットアーキテクチャによく似ている。一般的な5段パイプラインで構成されて、プロセッサの命令の効率自体には、リファレンスプロセッサと大きな差はないと思われる。しかし、MicroBlaze はキャッシュが実装されていて、リファレンスプロセッサよりメモリ性能が高い。

(単位:秒) n=4096*1024 seed=3 (単位:秒)			
	sort転送	sort全時間	sort実処理
reference	5.241	200.954	195.713
MicroBlaze	5.241	27.575	22.334
MB + 改良	5.241	11.213	5.972
(単位:秒) n=32*16 seed=8			
	mm転送	mm全時間	mm実処理
reference	5.242	226.446	221.204
MicroBlaze	5.241	64.010	58.769
MB + 改良	5.241	42.47	37.229
(単位:秒) n=512.iter=200.seed=8(16,100,8)			
	stencil転送	stencil全時間	stencil実処理
reference	5.241	347.324	342.083
MicroBlaze	5.241	167.753	162.512
MB + 改良	5.241	145.038	139.797
(単位:秒) n=2048 m=8192			
	spath転送	spath全時間	spath処理
reference	5.241	32.261	27.020
MicroBlaze	5.242	8.012	2.770
MB + 改良	5.241	7.710	2.469

4. おわりに

今回の提出プロセッサは、ベースのプロセッサに Xilinx の MicroBlaze をベースプロセッサに使用し、コンテストの要求仕様に適合する周辺回路を作成し、ソフトウェアのアルゴリズムの改善で速度向上を行った。予定していた付加するハードウェア拡張部分は、時間不足で現時点では完成に至っていない。

参考文献

- 1) MicroBlaze Soft Processor Core,
<http://www.xilinx.com/tools/microblaze.htm>