

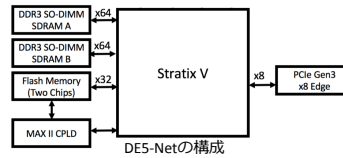
高位合成によるFPGAの高性能計算への適用

大畠 佑真⁽¹⁾, 藤田 典久⁽²⁾, 小林 諒平^(1,2), 山口 佳樹⁽¹⁾, 朴 泰祐^(1,2)
1:筑波大学大学院システム情報工学研究科 2:筑波大学計算科学研究センター

1. はじめに

- 近年、HPCの分野でFPGAがアクセラレータとして注目されてきている
- FPGAボードの通信ポートを用いて、低レイテンシのネットワークスイッチとして利用できる(TCA^[1])
- >ネットワークスイッチ+アクセラレータとして利用できないか?
- 問題点: FPGAのプログラミング言語であるVerilog HDLなどは記述が難しく実装に膨大な時間が掛かる
- >高級言語でプログラミングできる高位合成を使うことによって大幅に実装時間を短縮 -> 実用性を調査

2. 環境



DE5-Netの構成

interleave-on interleave-off

0x7FFF_FFFF 配列のアドレス

1KB

0x0000_0C00

0x0000_0800

0x0000_0400

0x0000_0000

SDRAM2

SDRAM1

SDRAM2 2GB

SDRAM1


SDRAM2

SDRAM1

SDRAM2

SDRAM1

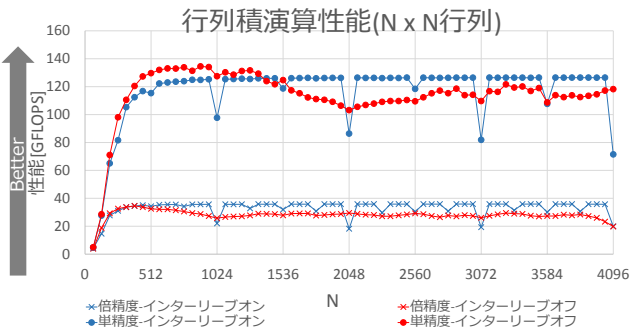
ハードウェア	
CPU	Xeon E5-2670v3
FPGA	StratixV 型番:5SGXE7N2F45C2
FPGAポート	DE5-Net DDR3-SDRAM(12.8GB/s) x2
ソフトウェア	
OS	CentOS 6.6
高位合成ツール	Altera SDK for OpenCL 16.0
論理合成ツール	Quartus16.0
言語	
ホスト	C++
デバイス	OpenCL C ^[2]



DE5-Net

3. 行列積演算 ($A \times B = C$)

- 高位合成を用いて、FPGAに行列積演算のプログラムを実装
- 単精度、倍精度およびインターリーブ有り無しとの測定
 - インターリーブ無しの場合は、 $A \times B = C$ のうち、 A と B の配列を別々のメモリに確保し、 C は余った領域に確保



	動作周波数 [MHz]	ロジック使用率[%]	DSP 使用率[%]	演算器数	理論性能 [Gflops]	最大実測値 [Gflops]
単精度-インターリーブオン	247.46	55	100	512	126.70	126.48
単精度-インターリーブオフ	267.58	55	100	512	137.00	134.51
倍精度-インターリーブオン	281.29	43	100	128	36.01	35.89
倍精度-インターリーブオフ	281.69	43	100	128	36.06	34.63

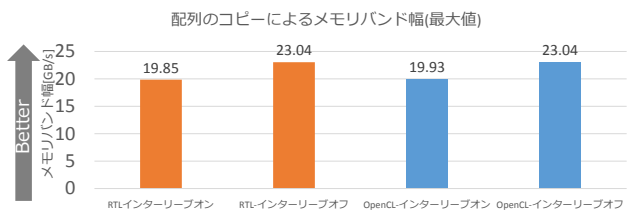
- 倍精度の演算を扱う場合、演算器のリソースが単精度の4倍必要となるため、性能は4分の1程度になる
- インターリーブの粒度が1[KB]と大きい場合インターリーブオフの方が性能が良い場合がある(4章で確認)
- 動作周波数と演算器数より求めた理論性能と同等の実測値がえられている -> 効率のよい回路が生成されているといえる

謝辞

本研究の一部は、JST-CREST 研究領域「ポスト ベタスケール高性能計算に資するシステムソフトウェア 技術の創出」研究課題「ポストベタスケール時代に向けた演算加速機構・通信機構統合環境の研究開発」、及び文部 科学省研究予算「次世代計算技術開拓による学際計算科学連携拠点の創出」による。また、本研究の一部は、「Intel University Program」を通じてハードウェアおよびソフトウェアの提供を受けており、Intel の支援に謝意を表す。

4. 自作RTLモジュールの追加

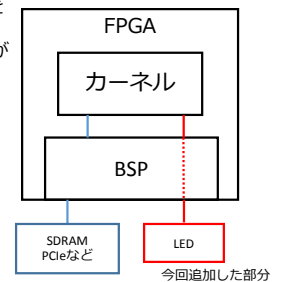
- Altera SDK for OpenCLはVerilog HDLで記述したRTLモジュールをOpenCLから関数として呼び出すことができる
 - 演算のコアとなる部分をRTLでチューニングできる
 - OpenCLでは記述できない動作ができる(可変精度など)
- 簡単なテストとして配列のコピーを行うRTLを追加
- 以下の4つのファイルを用意する必要がある
 - RTLモジュールの設定ファイル(.xml)
 - 自作RTLモジュールのI/Oや関数の名前について記述(配列のアドレス,要素数,SDRAMへのアクセスなど)
 - 自作RTLモジュール(.v)
 - xmlの設定ファイルに合わせてポートを宣言する必要がある
 - エミュレーション用ファイル(.d)
 - デバイスプログラム(.cl)
 - copy(X,Y,N); 左のように関数を呼び使用する



- 凡例のRTLが自分で作成した配列コピーのプログラムで、OpenCLが高位合成で生成したプログラムである
- 行列積演算の結果と違いインターリーブオフの方が性能が良い
- 自分で作成したプログラムと、高位合成で生成したプログラムに性能の差は見られなかったが、自作RTLモジュールを追加できることを確認した

5. 外部ペリフェラルの操作

- BSP^[3]で定義されていない外部ペリフェラルは、OpenCLから操作することができない(通信ポートなど)
- BSPを編集することでOpenCLから操作可能
- FPGAをネットワークスイッチとして利用するために重要
- 簡単なテストとしてFPGAボードのLEDをOpenCLから操作できるようにBSPを編集
- 以下のファイルを追加・編集する必要がある
 - OpenCLとI/Oおよび外部ペリフェラルとI/Oを行うRTLモジュールを作成(.v)
 - OpenCLとのI/OはAvalon Streaming形式^[4]で行う
 - BSPフォルダ内にあるsystem.qsysを編集
 - 作成したRTLと既存のOpenCLのシステムを接続
 - BSPフォルダ内にあるboard_spec.xmlを編集
 - OpenCLから作成したRTLを呼び出せるよう定義
 - BSPフォルダ内にあるtop.vを編集
 - 作成したRTLと外部のペリフェラルを接続
- 実際にOpenCLからLEDを操作できることを確認
- これを応用すれば、通信ポートなどの操作が可能



6. まとめ

- 3章では、OpenCLを用いてFPGAへHPCアプリケーションの実装ができることを、4章では、OpenCLから自作RTLモジュールを扱うことができることを、5章では、OpenCLから外部のペリフェラルを操作できることを示した
- これらの結果から、これらの技術を組み合わせることでOpenCLを用いてFPGAをネットワークスイッチ+アクセラレータとして利用できるのでないかと期待できる
- 今後、OpenCLから通信ポートを操作できるようにし、実際にネットワークスイッチとして動作できるか検証する

[1]TCA(Tightly Coupled Accelerators)
筑波大学計算科学研究センターで唱
ノードをまたぐアクセラレータ間を密に結合

[2]OpenCL
アクセラレータを用いて並列コンピューティングを行うた
めのC言語ベースのAPI
CPU,FPGAが混在するヘテロジニアスな環境でもプログラ
ミング可能

[3]BSP(Board Support Package)
FPGAボードを提供しているベンダーが提供
ボード毎にPCIeやSDRAMなどの物理的違いがあっても同じ
OpenCLのコードで実行できるようサポート

[4]Avalon Streaming形式
Altera社の提供している、ストリームデータ
を扱うためのインターフェース