

粗粒度再構成可能アーキテクチャのFPGA実装

1150147 毛利 真崇 (密山研究室)

1 はじめに

LSI 設計において、FPGA (Field Programmable Gate Array) を用いたプロトタイピングによる機能検証は極めて重要なプロセスである。粗粒度再構成可能アーキテクチャ (Coarse-Grained Reconfigurable Architecture: CGRA) の開発においても同様に、FPGA を用いたプロトタイピングの要求が高い。しかし、FPGA 上に CGRA を実現する場合、組み合わせループが基本セル間に多数発生し得るため、コンパイル処理に膨大な時間を要する。先行研究 [1] では、セル間でループ構造が発生しないようデータのやり取りを片方向に制限しているため、厳密にはプロトタイプとはいえない。そこで本研究では、ループ構造に起因する諸問題の解決方法を提案する。また、複数の FPGA に分割して実装することで、実装可能な回路規模を拡大させる手法について検討を行う。

2 再構成可能アーキテクチャ

本研究で対象とする CGRA を図 1 に示す。CGRA は、基本セルのアレイ構造で構成され、基本セルは演算部と配線部で構成される。演算部は、16bit ALU (Arithmetic Logic Unit), 入力選択用 16 入力 MUX, バイパス可能なレジスタを持つ。配線部は 18 入力 MUX で構成され、ALU の出力先と隣接セル間の接続構造を切り替える。

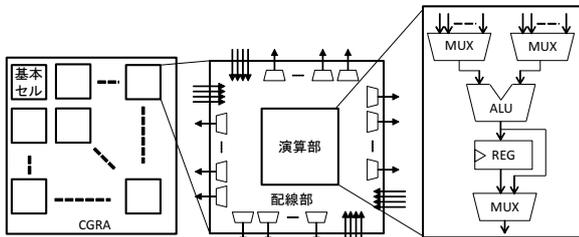


図 1: 対象 CGRA の概要

3 CGRA の FPGA 実装における問題の抽出

プロトタイピング用 FPGA ボードとして、Terasic 社 DE2-115 (Cyclone IV) を用いた。設計環境として、Intel Core-i7 クラッドコア 4.0GHz, 32GB RAM, Windows7 を搭載する PC 上で、Quartus II v13.1.4 を用いた。論理合成時に複数の基本セル同士で回路が意図せず最適化されることを防ぐため、基本セル毎に Design Partition を設定した (手法 1)。CGRA の RTL 記述をコンパイルすると、図 2 に示すような潜在的なループ構造が多数検出され、コンパイル処理に約 28 時間という膨大な時間を要した。このため、コンパイル時の潜在的ループ構造に起因する問題の解決が不可欠であるとわかる。

また、1 つの基本セルの実装に必要な LE 数は 4,369 であり、FPGA 全体の約 4% を占める。すなわち今回用いた FPGA では理論上でも最大で 5x5 個の基本セルまでしか搭載できず、実際はさらに少なくなる。このため実用的なアレイサイズの CGRA を FPGA に実装するには、この問題の解決が不可欠であることがわかる。

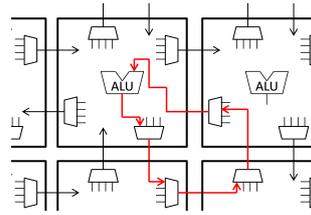


図 2: ループ構造の例

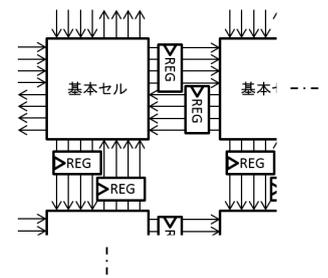


図 3: セル間レジスタ挿入

4 組み合わせループの解決

ループ構造を解決する手法として、開発ツールのコンパイルオプションを変更し、組み合わせループ構造を無視してコンパイルを行う手法 (手法 2) ならびに、基本セル間にレジスタを挟むことで、強制的に組み合わせループを抑制する手法 (手法 3) を検討した。両手法を適用し、演算粒度、アレイサイズを変えてコンパイルを行った。16bit 粒度の 2x2 セルアレイの場合のコンパイル時間を表 1 に示す。手法 2 では、論理合成の時間は短縮し、全体のコンパイル時間を 37% 削減できたが、配置配線以降の処理時間に変化は見られなかった。一方、手法 3 では、手法 1 に対して 99% 以上削減することができた。

表 1: コンパイル時間の比較

	コンパイル時間 (分)			
	A&S*	Fitter	Others	Total
手法 1	1,091	474	94	1,659
手法 2	478	475	96	1,049
手法 3	1	2	1	4

* Analysis & Synthesis

5 回路規模の解決

100x100 といった実用的なアレイサイズの CGRA を FPGA に実現するため、複数の FPGA に CGRA を分割実装する手法を提案した。本研究では、2 枚の FPGA に基本セルアレイを実装し、FPGA 間を直接接続することでホモジニアスな CGRA を 2 つの FPGA にまたがって実装した。さらに、多数の FPGA を接続するために、ボード間接続に起因する遅延を測定し、接続する際に考慮すべき制約を明らかにした。

6 まとめ

FPGA を用いた CGRA のプロトタイプ設計手法について検討した。CGRA が潜在的にもつループ構造に起因するコンパイル時間の増大ならびに、FPGA の回路規模の制約に対して、その解決法を提案した。今後は、複数の FPGA を接続する場合の、クロック供給方法について検討し、ボード間で同期をとる手法を検討する。

参考文献

- [1] 小川裕喜, 尼崎太樹, 飯田全広, 久我守弘, 末吉敏則, “仮想 CGRA への Java ソフトウェアのマッピングと FPGA 実装,” 電子情報通信学会技術報告 RECONF2013-47, pp. 45-50, 2013 年 11 月