

データ駆動型プロセッサのFPGA実装におけるフロアプラン最適化の検討

1200290 井上 聡 【コンピュータ構成学研究室】

1 はじめに

近年, IoT エッジ機器の高機能化・高性能化が益々望まれつつある. この実現にあたり, 複数ストリームデータの多重処理性能が高いデータ駆動型プロセッサDDP(Data Driven Processor)[1] を, 回路構成を改変可能なFPGA(Field Programmable Gate Array)上に実装することが有望である.

一方, 現行のFPGA回路設計ツールは同期回路専用であり, DDPをセルフタイム型パイプラインSTP(Self-Timed Pipeline)により実装する際, 回路の最適化を充分に行えない. そのため, 先行研究[2]では, 回路ブロック単位での配置・配線を手動で最適化する手法が提案されている.

本研究は, 先行研究の最適化手法の中で, 各regionを巧妙に配置(フロアプラン)してDDP全体の面積を削減できる自動化アルゴリズムを提案し, その効果を評価した.

2 DDPの最適化に関する現状と提案

DDPのSTP実装では, 隣接ステージ間でハンドシェイク通信を行うことでパケットの転送制御を行う. 各ステージ(region)での転送時間は, それを構成する回路の遅延で決まるため, フロアプラン結果に基づく配線遅延の影響を受ける. 先行研究の手法では, DDPの各regionに必要な回路規模から必要な論理アレイブロックLAB数を算出し, 隣接ステージが隣接するように手動で配置されていた.

本研究では, DDPのregion単位でのフロアプランを自動化することにより, 設計者の熟練度に依存せずに, 安定したフロアプランを自動生成することを目指す.

3 フロアプランの最適化手法

提案するフロアプランは, 以下の手順で実施する.

1. FPGA上の配置可能領域に関して, 基準座標(x,y), 領域の幅, メモリBlock等専用回路ブロック列のx座標を与える. また, 設計対象DDPの情報として, 各regionに必要なLAB数, メモリBlock数, DSPBlock数を入力する.
2. $\lceil \sqrt{\text{LAB数}} \rceil$ を計算し, 各regionのWidthとHeightを仮決定する.
3. ステージの隣接関係に従って配置順を決定し, その順に各regionの配置座標(x,y)を仮決定する.
4. 各regionにおいて, (Width*Height > LAB数)を満たす範囲で整数WidthとHeightを微調整.

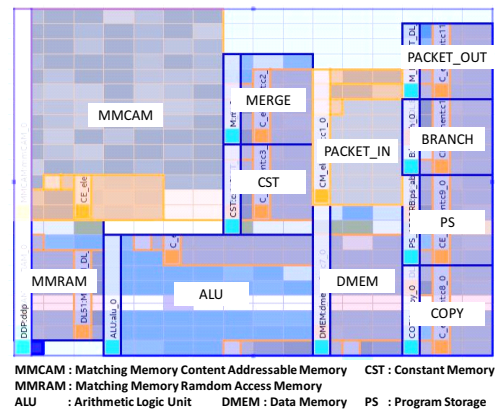


図1 DDPの配置・配線結果

表1 提案フロアプランに基づくDDP実装結果

	提案手法	Quartus
配置領域(W*H)	23*16	23*19
LAB使用率[%]	73.6	61.8
スループット [packet/sec] *	92.6M	94.6M

* Slow条件 (V_{DD} 1.2V, 85°C)での評価

5. 再度各ステージの隣接関係を維持したまま, 全面積が削減されるよう各regionの(x,y)を調整する.

以上のアルゴリズムを実装したpythonスクリプトで自動生成したフロアプランを図1に示す.

4 評価

提案フロアプランと回路設計ツールIntel社Quartus Prime 18.0の自動配置機能によるフロアプランでそれぞれDDPをIntel社MAX10-50上に実装し, 比較評価した. 評価結果を表1に示す. この結果から, 提案手法に基づく回路実装は, Quartusよりも面積が約16%削減できることが判った. 一方, スループットは約2%劣化してしまった. 今回は, 特定のDDP構成ならびにMAX10-50に依存したアルゴリズムが一部含まれているため, 今後, 一般化する必要がある.

参考文献

- [1] H. Terada, et al., "DDMP's: Self-Timed Super-Pipelined Data-Driven Multimedia Processors," Proc. IEEE, vol. 87, no. 2, pp.282-296, Feb. 1999.
- [2] K. Nagano and M. Iwata, "Area-Efficient FPGA Implementation of Self-Timed Data-Driven Processor," ISFT2019, Aug. 2019.