

FPGA 回路向けトランスダクション法の拡張

2B-2

幸田武範 上林彌彦
京都大学大学院工学研究科

1 はじめに

FPGA^[2](Field Programmable Gate Arrays) は、近年になって論理設計の各分野で用いられるようになってきた新しいタイプの素子である。この素子は従来の素子とは異なり、内部の論理ブロックの表現する関数を変更可能であるという特徴を持っており、ユーザの手によって比較的大規模な回路を実現できる素子として注目を集めている。このような背景から、FPGA の論理設計手法を確立する要求が高まってきている。

そこで、従来から用いられてきた論理回路最適化手法であるトランスダクション法^[1] の概念を応用した FPGA 回路向けの最適化手法を考案^[4] し、その後拡張を行った。

本稿では、文献^[4] の手法を元に後述のエラー補償手続きの適用順序を工夫することで、従来法と比較して面積や論理ブロック間接続コスト、段数などを小さくする手法について述べる。

2 基本的事項

2.1 FPGA 回路最適化手法の概略

FPGA 回路向け回路最適化手法の大まかな流れは、以下の通りである。

- step1 FPGA 内の論理ブロックを一つ選択
- step2 ブロックの入力結線を切断
 - 後述のエラー補償手続きを適用し、生じたエラーを修正
 - a エラー補償成功なら、step1 へ
 - b エラー補償失敗なら、step3 へ
- step3 元の状態にもどして step1 へ

以上の処理を可能な限り繰り返す。

2.2 エラー補償手続き

本稿で用いたエラー補償手続きは、以下の三種類である。

2.2.1 内部論理補償手続き

当該論理ブロックにおいて、要求された論理関数とそのブロックの内部論理を変更することによって実現する手続きである。要求出力をブロックの入力関数でシャノン展開し、手続き成功後の内部論理を計算する。

2.2.2 入力補償手続き

前節の内部論理補償を拡張したもので、要求された出力を実現するのに内部論理の変更だけでなく、当該論理ブロックの入力の論理関数を変更する手続きである。図1に本手続きの概念図を示す。

入力側ブロックの関数を変更するためその変更が回路出力に影響をおよぼさないようにする必要がある。そこで、

トランスダクション法^[1]における回路の冗長性を示す概念である CSPF によるチェックを行う。

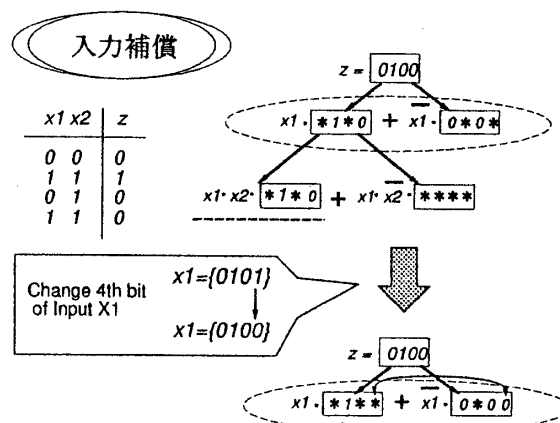


図1: 入力補償手続きの概念図

2.2.3 置換手続き

元の結線の代わりに他の結線を繋ぐことによって、要求されている出力を実現する手続きである。

もともと、置換のみでエラー補償を実現することは困難なため、単独で用いるのではなく前述の内部論理補償手続きや入力補償手続きと組み合わせて用いる。

3 エラー補償手続き適用順序の変更

前述の回路最適化の際のエラー補償手続きの適用順序について、以下の3つのアプローチについて実装を行い、面積や結線コスト、段数等のさらなる最適化を試みた。

3.1 ブロック削除優先手法

本手法は、基本法^[4] で用いられている手法である。本手法では、置換処理の際に置換対象結線を元々つながっていたブロックのファンアウト数より大きなファンアウト数を持つブロックからの出力結線に限定した。

以下に例を示す。尚、(A,3) はブロック A のファンアウト数が3であることを示している。また例では、A が元の結線のつながっていたブロック、B が置換対象として選択された結線につながっているブロックを表す。

例1. $A \geq B$ の際に置換を行った場合

置換前: (A,4)(B,2) → 置換後: (A,3)(B,3)
置換してもブロック数が減る可能性はあまりかわらない。

例2. $A < B$ の際に置換を行った場合

置換前: (A,1)(B,3) → 置換後: (A,0)(B,4)
置換すればするほどファンアウト0のブロックの出現の可能性が増える。

これは、ファンアウト数が元々大きかったものは大きく、小さかったものはより小さくなるようにすることで、ファンアウト数が0となるブロックの出現する可能性が高

| 回路名 | 初期回路 | ブロック削除優先手法 | 接続コスト優先手法 | 段数優先手法 |
|--------|------------------|-----------------------|-----------------------|-----------------------|
| alu2 | 137/517/23/2439 | 125/467/23/2190/12751 | 90/299/18/1174/153 | 131/489/23/2162/240 |
| alu4 | 250/1009/23/4022 | 235/900/23/3452/987 | 239/924/23/3416/610 | 244/940/23/3517/1466 |
| b9 | 87/176/4/281 | 83/160/4/263/22 | 83/160/4/263/22 | 87/176/4/281/9 |
| c8 | 76/191/5/317 | 73/180/5/304/42 | 72/173/5/293/42 | 75/188/5/315/31 |
| cordic | 44/82/5/119 | 37/55/5/85/5 | 37/55/5/85/5 | 44/82/5/119/3 |
| dalu | 489/1772/11/3812 | 420/1447/10/3030/3321 | 388/1298/10/2693/3893 | 451/1560/10/3211/5488 |
| f51m | 52/173/10/442 | 46/152/10/372/30 | 44/142/10/351/29 | 16/30/4/54/2 |
| i9 | 351/1121/6/2563 | 346/977/6/2137/393 | 346/977/6/2137/405 | 351/971/5/1867/799 |
| lal | 84/214/6/383 | 61/118/6/198/12 | 61/118/6/198/11 | 66/139/6/220/11 |
| pcler8 | 53/96/4/182 | 53/96/4/184/8 | 52/87/4/166/2 | 53/96/4/184/2 |
| sct | 61/164/4/256 | 54/139/4/226/25 | 52/131/4/214/24 | 61/164/4/256/12 |
| term1 | 171/549/8/1113 | 97/250/8/567/80 | 120/333/8/600/102 | 116/327/8/680/113 |
| ttt2 | 100/300/4/445 | 81/219/4/326/49 | 78/200/4/289/57 | 86/241/4/356/61 |
| vda | 583/1866/5/2404 | 377/1367/5/1840/474 | 583/1866/5/2404/1222 | 583/1866/5/2404/1343 |
| x3 | 379/1073/9/1748 | 349/948/8/1492/551 | 347/931/8/1455/1768 | 363/1000/8/1555/1113 |

各欄：(ブロック数 / 結線数 / 段数 / 接続コスト / 計算時間(秒)) 太字は結果が良かった箇所

表 1: 各手法の実験結果

まり、回路全体のブロック数を減らす効果などが高まるものと考えたからである。

3.2 論理ブロック間接続コスト優先手法

一般にFPGAからなる回路は動作速度が他種のゲートアレイによる回路と比べて劣っている。

本手法は、遅延の一因にブロック間結線網を実現する際に用いるスイッチのスイッチング速度の遅さがあげられることに注目し、結線上のスイッチ数(主に各ブロックの段数差に比例)を減らすことを目的とした手法である。

実装の際には、各ブロックの入力からの深さ(段数)をもとに接続コストを計算し、置換手続きの際に最も接続コストの小さいものを選択するようにした。

3.2.1 段数優先手法

上記の手法と同様に、回路の動作速度はその回路の段数にも大きく影響される。

そこで、回路の段数を可能な限り小さくするために置換手続きの際に入力からの深さが最小となるブロックと置換を行うようにした。

4 実験結果および考察

4.1 実験結果

前節でのべた3つの目的別最適化手法をC言語を用いて実現し、SUN ULTRA1にて実験を行った。実装の際には、FPGAのモデルに5入力1出力の任意の論理関数を実現可能な論理ブロックからなる素子を用いた。さらに実験の際の初期回路には、MCNCベンチマーク回路をSISを用いて5入力の論理ブロックにマッピングした回路を用いた。またエラー補償手続きには、前述の三種類の手続きを組み合わせて用いた。

表1にその実験結果を示す。

4.2 考察

以下に、各手法についての考察を述べる。

まず、ブロック削除優先手法についてであるが、面積を小さくする効果は確かめられたが、予想していたほどの効果は得られなかった。

次に、論理ブロック間接続コスト最小化手法については、他手法と比べて多くの回路において接続コストが小さくなったことから望み通りの結果であったといえる。

最後に段数優先手法において、f51mなど初期回路の段数が10段程度の回路なら効果があるが、段数が小さいものについては効果が現れにくいことが示された。

5 結論

今回はFPGA回路の特徴を検討の上で、エラー補償手続きの適用順序に特徴を工夫することで、生成回路をより最適なものに

得られた実験結果は必ずしも満足いくものではなかったが、この結果を元にさらに改良を行っていきたいと思っている。

また、本提案手法が単に回路設計時のみFPGAの特徴を利用した手法であったことから、FPGAの「必要に応じて再プログラム可能」という特徴をすべて生かしているとはいえない。回路の再利用や実行時の動的回路変形など、FPGAでなければ実現が困難な用途も多い。これらの用途に用いるための設計手法についても考えていきたいと思っている。

謝辞

SBDDパッケージを使用させていただいた矢島研究室の皆様へ深謝します。尚、本研究は新エネルギー・産業技術総合開発機構(NEDO)の提案公募型・最先端分野研究開発のプロジェクトとしての支援を受けている。

参考文献

- [1] S.Muroga, Y.Kambayashi, H.C.Lai, J.N.Culliney, "The Transduction Method-Design of Logic Networks Based on Permissible Functions", IEEE Trans.Comput., pp.356-359, Nov.1989
- [2] S.D.Brown, R.J.Francis, J.Rose, Z.G.Vranesic, "FIELD PROGRAMMABLE GATE ARRAYS", Kluwer Academic Publishers, 1992
- [3] 山下茂, 上林彌彦, 室賀三郎, "許容関数に基づいた表参照型FPGAの最適化手法", 電子情報通信学会論文誌, D-1, Vol. J78-D-1, No.11, pp.878-885, Nov.1995
- [4] 幸田武範, 上林彌彦, "内部論理補償と入力補償によるFPGA回路設計", 情報処理学会 第52回全国大会, Vol. 6, 1K-10, pp.19, Nov. Mar.1996