

JCAD—論理合成システムFALCONの合成手法

4E-8

後藤 謙治、玉野 正剛、野村 勝幸、酒井 五雄

(株)東芝 青梅工場

1. はじめに

今日、LSI技術の進歩により、LSIの大規模化・高性能化・高密度化が進み、それにとまないハードウェア設計の設計効率の改善、設計期間の短縮、設計品質の向上が望まれている。

これらを実現するために、機能設計・論理設計のCAD化が必要である。我々は機能設計の効率化のため、ラップトップEWS (SPARC LT) 上で動作する大規模論理設計システムJCAD^[1]の開発を行った。特に論理設計の効率化を図るため、ブロック図から論理回路の生成を行う自動論理合成システムFALCONの開発を行った。以下、このシステムの合成手法について概要を述べる。

2. 合成戦略

ハードウェアの機能設計は、資源間のデータの流れを規定するデータバス系回路およびデータ操作の手順を規定する制御系回路を設計することで行う。しかし、データバスで扱う機能(ALUやADDER)は比較的高機能かつ分解困難なことから、ブール式レベルの論理最適化や局所的な最適化の適用が難しく、従来の論理合成システムではデータバスの合成が必ずしも満足のいく結果を生成するとは限らなかった。

FALCONではブロック図^[2]およびH²DL^[3]を入力として受理できる。前者は回路の構造を記述するのに適しており、後者は回路の動作を記述するのに適している。この特性に注目し、ブロック図として与えられた構造記述をデータバス系回路として、H²DLとして与えられた動作記述を制御系回路として処理することで、データバスを含む大規模な回路についても、現実的な時間内に実用的な回路を合成することが可能になった。

3. 合成方法

FALCONでは2とおりの合成方法を併用している。1つは構造記述からデータバス系回路を生成する「論理変換」であり、1つは動作記述から制御系回路を生成する「論理合成」である。論理変換はブロック図でのプリミティブを起動条件とするルールベースシステムからなっており、論理合成はブール式最小化/多段化を用いた論理最適化システムである。

4. 最適化手法

4.1 ルールの起動制御

論理変換において、一般にあるプリミティブに対する変換ルールは複数用意されている。これらのルールから、所要ゲート数・遅延時間を評価関数として、自動的に最適なルールが選択される。

4.2 ブール式最適化

論理合成においては、dead code eliminationやconstant foldingなどのコンパイラ的最適化が構文解析段階で行われる。また、ESPRESSO-II^[4]ベースの論理最小化、拡張weak-divisionによる論理多段化、およびCMOSアダプティブな否定論理素子への変換が行われる。これらの最適化処理は構文解析で得られる情報を基に動的にフローが自動決定される。

4.2.1 拡張weak-division

論理多段化の手法としてweak-division^[5]がよく知られている。この手法は比較的高速に論理の多段化を行うことができるが、ブール式としての除算は扱うことができず、多段化の結果は必ずしも優れたものではなかった。これを解決するために、ブールの除算を考慮したtransduction法^[6]等のstrong-divisionが提案されているが、計算時間が多くなることが問題であった。

そこで、従来のweak-divisionに若干の拡張を与え、計算時間のオーバーヘッドを抑えながら限定的にstrong-divisionを実行できるようにする。

Techniques for Logic Optimization in FALCON

Kenji GOTOH, Seigo TAMANO, Katsuyuki NOMURA,
Itsuo SAKAI TOSHIBA Corp. OME WORKS

内部表現として常に1である universe term ' U ' ($\forall T \subseteq U : T$ は項)を許し、含有則 ($U+T=U$, $UT=T$) や吸収則 ($A+\overline{A}B=A+B$) を weak-division の核算出操作時に適用する。核は元の式に比べると十分簡単な論理式になっているので、吸収則や含有則の適用は容易かつ効果的である。

例えば、 $f=abcd+\overline{bcd}$ (核: $ab+\overline{b}$) を多段化する場合、核を中間式とする $f=tcd$, $t=ab+\overline{b}$ (リテラル数6) が weak-division における解として得られる。当手法では核に対する吸収則の適用を行い、strong-division と同等の $f=t'cd$, $t'=a+\overline{b}$ (リテラル数5) が容易に得られる。

4.2.2 ブール式最適化処理の自動制御

拡張 weak-division を用いることにより、従来の weak-division では不可能であった論理的冗長な式の多段化が可能になった。これにより、論理最小化を行わない式についても多段化が可能になった。一般に、パリティチェック回路のような最小項が多い回路や、入力信号線や使用リテラル数が多い回路の場合には、論理最小化の計算時間がきわめて大きいものになり、現実的でなくなる。そのため、入出力信号線数やリテラル数から計算時間を予測評価し、論理最小化を行わずに多段化を行うようにフローを自動制御することで、計算時間が常に現実的になるような工夫をしている。

4.3 極性調整・ファンアウト調整

4.1 や 4.2 の操作で得られるテクノロジー非依存の generic cell は、テクノロジー毎に特有のライブラリに従いテクノロジーマッピングされる。テクノロジーマッピングと並行して、プリミティブ単位にファンアウト先の負荷を評価し、バッファ/インバータの付加する。併せて相補的な出力を持つプリミティブについては、適当な極性の出力を選択し配線する。

プリミティブは、変換ルールによりテクノロジー依存の論理回路に変換されるが、この論理回路は適当な極性の I/O のもと局所最適であることが保証されている。これらのプリミティブの接続はそれぞれの入出力は指定の極性になることが要求されるが、極性の調整は接続のもう一方の端に限り一致を取ればよく、peephole optimization の適用による冗長ゲート削減に比べて操作が容易である利点がある。

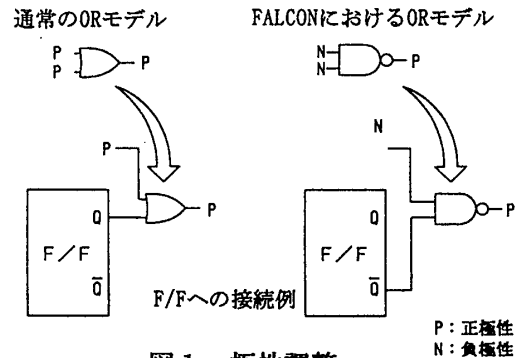


図1. 極性調整

図1. の例では入力を負極性にとった NAND ゲートを generic cell OR の変換先論理回路としている。多くのプリミティブでは任意の極性を出力することができるため、インバータなしでも信号値の反転(極性の反転)は容易に行える。このように形式上インバータフリーとすることで冗長インバータ削除の為の peephole optimization が不要になっている。

5. おわりに

N P 困難な計算時間を要するとされる最適化問題を、現実的な時間内に完了するための一方法を提示した。これらの手法を用いた実際の設計評価は別稿^[7]にゆずる。

【参考文献】

- [1] 覚井 他, “大規模論理設計システム JCAD の開発”, 情処第 44 全国大会, 1992
- [2] 木暮 他, “ラップトップ PC 上の機能図入力システム FSET”, 情処第 36 全国大会, 1988
- [3] 宮田 他, “階層的ハードウェア記述言語 H²DL の思想”, 設計自動化研究会 22-1, 1984
- [4] R.K.Brayton et al. “Logic Minimization Algorithms for VLSI Synthesis”, Kluwer Academic Publishers, USA, 1984
- [5] R.K.Brayton et al. “The decomposition and factorization of Boolean expressions”, ISCS, 1982
- [6] S.Muroga et al. “The Transduction Method—Design of Logic Networks Based on Permissible Functions”, IEEE trans.computers, Oct. 1989
- [7] 玉野 他, “JCAD—論理合成システム FALCON の開発と評価”, 情処第 44 全国大会, 1992