

論理ベリファイアにおける  
設計不良箇所の追跡

6Q-2

奥沢治、宮川祐一、小田原豪太郎

(東京大学)

1. はじめに

論理検証には、i)シミュレーション、ii)組み合わせ回路と論理式の比較、iii)記号シミュレーション、iv)機能設計の検証などがあるが、本研究で扱う論理ベリファイアは、ii)の方法によるもので、図形的機能記述言語SF DL (Symbolic Functional Description Language) [1] による機能記述とゲートレベル回路設計の比較検証を行なう[2]。比較検証後、ゲートレベル設計に誤りがあるという結果が出た場合、設計者が回路を修正する際の支援として、ゲートレベル設計中、どの部分に誤りがあるかを示す手法を述べる。また、誤りを含む回路を使って、その誤りを含む範囲を限定する実験の結果について述べる。

2. 論理ベリファイアの機能と構成

論理ベリファイアの構成を図1に示す。SF DL によるレジスタトランスファレベルの機能記述と、ゲートレベル設計から、各出力端子についてのブール式を抽出し、それらを比較する。この際、順序回路も、フリップ・フロップで回路を切断し、フリップ・フロップの旧状態を入力、フリップ・フロップの新状態を出力とすることで、組み合わせ回路と同様に扱う。また2つのブール式の比較には(2つのブール式をA, Bとすると)、 $A\bar{B} = \bar{A}B = 0$ であるかどうかを検証する。これを式の展開で行なうため、2つの式が一致しない場合の入力変数の組み合わせ、すなわち、2つの出力値が異なる場合の入力パターンはすべて求めることができる。式の計算の際には、ブール式を、分割・圧縮して使用メモリが大きくなるようにしている[3]。不一致信号パ

ターンをもとにゲートレベル設計上で、誤りの含まれる部分を限定していく操作を行ない、その結果を出力するのが不良箇所追跡部である。

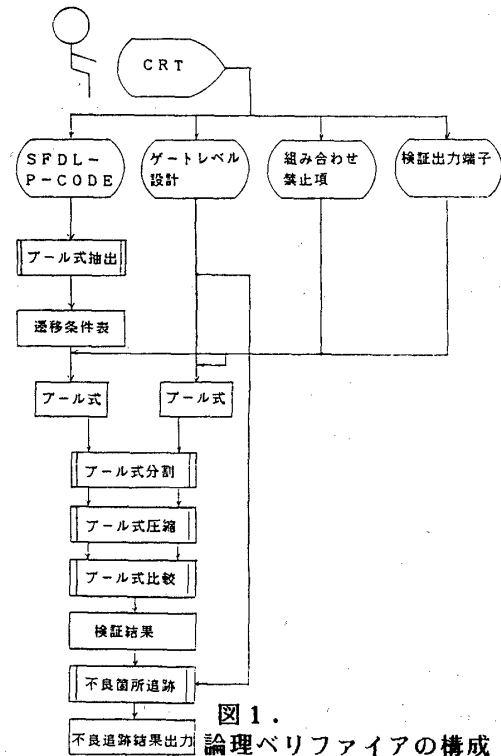


図1. 論理ベリファイアの構成

3. 設計不良箇所の追跡

3.1 不良追跡の必要性

設計されたゲート回路をベリファイアで検証した後、機能設計と一致しなかった場合、設計者は修正を行なう。本ベリファイアでは前項で述べた不一致信号パターンを出力するために、設計者はこれを見て回路を修正することも可能であるが、この作業は、回路が大規模になればなるほど困難を伴う。つまり、入力パターンの組み合わせが増え、各パターンを回路上で追うことは複雑で、時間もかかるからである。そこで、不一致入力パターンをゲートレベル設計に

入力したときの信号値の伝播を追って、設計不良箇所の限定をする。

### 3. 2 アルゴリズム

ここで扱う回路設計のミスは、問題を単純にするために、結線ミス、ゲートの不足、ゲートの種類の間違い等を想定する。また、回路設計の中で、修正すべきゲートを指定することは、困難である。ミスによっては特定できない場合もある。そこで、ミスの存在するパスを指定すること、しかも、不一致のパターンを用いて、できる限り範囲を小さくすることを目的とする。これによって、設計者は最小限の範囲をチェックすれば済むことになる。また、前項で述べたように本ベリファイアでは、順序回路も組み合わせ回路に分割して検証を行なうため、設計不良を追う時にも組み合わせ回路を対象として考える。

処理手順を以下に示す。

- i) 注目する検証出力につながるゲートをリスト・アップする。
- ii) 不一致パターンを各外部入力、フリップ・フロップのQ出力（前状態）として与える。
- iii) 入力から出力までの間の組み合わせ回路中の信号線の値を決定する。
- iv) 検証出力側から出力信号値を決定した入力信号値の方へパスを追いかけて行く。例えば、ANDゲートの場合、その出力値が0ならば、0の信号値をもつ入力線へ、出力値が1ならば、すべての入力線の方へ追跡を行なう。
- v) iv) で追ったパスを記憶する。
- vi) ii) ~ v) を繰り返し、各不一致パターンの追跡パスの共通部を求める。この共通部にミスが存在し、修正を行なわねばならない。

### 4. 実験結果と評価

110ゲートの組み合わせ回路について、設計不良箇所の追跡を行なった結果を表1に示す。表1中、(1)は、1つのインバータを取り除いたケース、(2)は、それに加え1つの素子を別の素子に変えたケースである。ここでは、1つの出力端子に80ゲ

誤りの回路	検証出力に関与するゲート数	誤りを含むとされた部分のゲート数	比率 (%)
(1)	80	26	32
(2)	80	30	37

\*実験回路  
 総ゲート数 110  
 1出力に関与する最大ゲート数 80  
 1出力に関与する最大入力数 12  
 検証処理時間 4min.

表1. 実験結果

トが関与しているが、このうち、不良が含まれると限定できたのは、全体の30-40%である。この結果から設計者が回路を修正する場合、修正すべき範囲は、狭くなっているのは事実だが、指摘の範囲をさらに狭くする必要がある。

### 5. まとめと今後の展望

機能記述とゲート設計の比較検証を行なう論理ベリファイアにおいて検証結果をもとに、ゲート設計中の不良を含む部分を限定していく方法について述べた。今後は、大きな回路についての実験例を増し、指摘した部分に誤りが必ず含まれ、かつ、どのようにしてその範囲を小さくしていくかが課題である。

また、論理ベリファイアの検証対象としては、メモリや、双方向バスなどをどう処理するかが課題である。

#### [参考文献]

- [1] G.Odawara et al. ; "A Symbolic Functional Description Language", Proc. of 21th Design Automation Conference
- [2] 小田原、富田、尾方 ; "論理ベリファイアの開発"、情報処理学会第29回全国大会
- [3] 小田原、富田、尾方、太田 ; "論理ベリファイアにおける処理回路規模拡大について"、情報処理学会第31回全国大会
- [4] G.Odawara et al. ; "A Logic Verifier Based on Boolean Comparison", Proc. of 23th Design Automation Conference