

6E-8 冗長故障判定アルゴリズム REDUCT

且代 三弥子 中田孝広 森脇 郁 新倉 隆夫
(株)日立製作所

1. まえがき

スキャン設計を前提とした論理回路の故障診断は組合せ回路の故障診断に帰着される。組合せ回路の故障診断において、未検出故障が冗長故障であるか、テストパターン生成困難な故障であるかを判定することは重要な課題の一つであり、この課題に対処する方法として[1]が知られている。本論文では、回路簡約手法を用いた冗長故障判定アルゴリズム REDUCT を提案する。

2. 問題と回路の変換

組合せ回路の単一縮退故障に対するテストパターン生成問題及び冗長故障判定問題の困難さは回路内の再取れんや先頭信号線の増加等に起因している。REDUCTでは、再取れんや先頭信号線の影響を把握しやすくするために、問題と回路の変換を次のように行う。最初に、問題「回路Xの故障fは冗長故障か?」を考える。この問題と問題「1出力回路Yの出力は常に0か?」が等価になるようにXをYに変換する。Yはブール微分法[2]で導出されるテストパターンを解に持つ方程式の左辺を表現した回路と同じである。次に、YをNORゲートと入力エッジのみで構成される等価回路Zに変換する。これらの変換により、冗長故障判定問題のかわりに、問題「1出力回路Zの出力は常に0か?」を考えればよい。

3. 回路簡約手法

回路Zに対して四つの簡約手法(共通ゲートのくくりだし、先頭信号線の更新、簡約処理1、簡約処理2)を用いて回路の複雑さをできるだけ除去する。共通ゲートのくくりだしと簡約処理1は再取れんを減らす役割を持ち、先頭信号線の更新と簡約処理2は先頭信号線を減らす役割を持つ。

(1) 共通ゲートのくくりだしでは、回路内の二つ以上のゲートの入力ゲートの集合が全く同じ場合に、それらのゲートを一つのゲートで代表させる。

(2) 先頭信号線の更新では、先頭信号線の出力側にあるゲートの組で互いに独立に0又は1に制御できるものを探し、それらのゲートの組が先頭信号線になるように回路を変換する。

(3) 簡約処理1では、多入力ゲートの入力側を局所的に簡約する。簡約対象多入力ゲートRに対して、「Rの出力 = 1 \Leftrightarrow I₁の出力 = \dots = I_nの出力 = 0」となるようなゲートI₁、 \dots 、I_nの中で最も入力側にあるものを含意により求め、RとI₁、 \dots 、I_nの間の不要な論理を削除し、RとI₁、 \dots 、I_nの間に直接結線をつなぐ(図1)。

(4) 簡約処理2では、対象先頭信号線hから回路の出力ゲートOに到るどの経路も偶数個のゲートを含む場合には、hの論理値を1とする。又、hからOに到るどの経路も奇数個のゲートを含む場合には、hの論理値を0とする(図2)。

4. 簡約後の処理

上記の回路簡約手法を用いた回路Zの簡約中に回路の出力を1にする入力パターンの存否が判定できなかった場合には、簡約後の回路の出力論理値を1として含意を行い、含意中に生じた未正当化信号線を後方追跡とバックトラックにより正当化する。すべての未正当化信号線が正当化されたならば、回路の出力を1にする入力パターン（テストパターン）が求まり、すべてのバックトラックに対して正当化が失敗したならば、そのような入力パターンは存在しないと判定する。

5. 評価結果

ISCAS85ベンチマーク[3]に対してテストパターン生成アルゴリズム N^2-V [4]とREDUCTを適用した(表1)。REDUCTは N^2-V で未検出となった故障に対して冗長故障判定を行う。

6. あとがき

回路簡約手法を用いた冗長故障判定アルゴリズムREDUCTを提案した。 $N^2-V+REDUCT$ はすべてのISCAS85ベンチマーク回路で検出率100%を達成した。ここで、検出率の定義は、(検出故障数 * 100) / (仮定故障数 - 冗長故障数)である。

参考文献

[1]M.H.Schulz and E.Auth, "Improved Deterministic Test Pattern Generation with Applications to Redundancy Identification", IEEE Trans. Compt., CAD-8, No.7, pp.811-816, July 1989.
 [2]E.F.Sellers, M.Y.Hsiao, and T.Arima, "Analyzing Errors with Boolean Difference", IEEE Trans. Compt., C-17, No.7, pp.676-683, July 1968.
 [3]F.Brglez and H.Fujiwara, "A Neural Netlist of 10 Combinational Benchmark Circuits and a Target Translator in FORTRAN", Proc. IEEE Int. Symp. Circuits and Systems, pp.679-682, June 1985.
 [4]且代三弥子, 新倉隆夫, 森脇 郁, " N^2 個の論理値を扱うテストパターン生成アルゴリズム", 情報処理学会論文誌, Vol.30, No.9, pp.1211-1218, Sep. 1989.

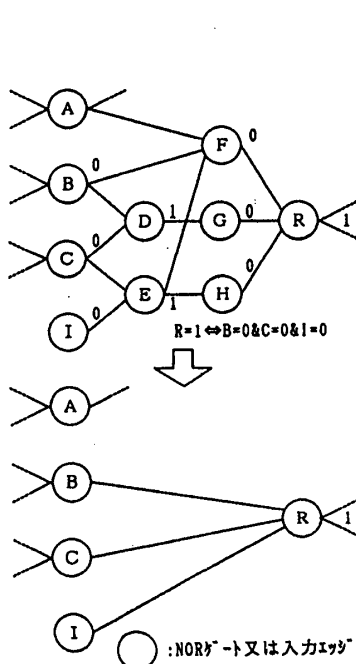


図1. 簡約処理1の例

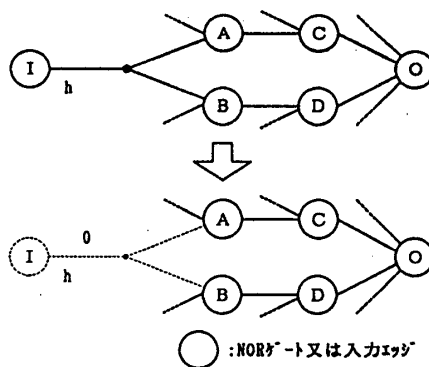


図2. 簡約処理2の例

表1. N^2-V とREDUCTの評価結果

回路	N^2-V					$N^2-V+REDUCT$				
	検出率	未検出故障数	冗長故障数	CPU時間		検出率	未検出故障数	冗長故障数	CPU時間	
				TG	TS				TG	TS
c432	98.88	3	1	8	1	100.00	0	4	19	1
c499	100.00	0	8	3	1	100.00	0	8	3	1
c880	100.00	0	0	1	1	100.00	0	0	1	1
c1355	100.00	0	8	14	1	100.00	0	8	14	1
c1908	100.00	0	9	8	2	100.00	0	9	8	2
c2670	99.68	11	106	79	4	100.00	0	117	93	4
c3540	100.00	0	137	27	6	100.00	0	137	27	6
c5315	100.00	0	59	10	7	100.00	0	59	10	7
c6288	100.00	0	34	64	5	100.00	0	34	64	5
c7552	99.18	54	77	521	14	100.00	0	131	801	14

TG:テスト生成, TS:テストシミュレーション, CPU時間の単位:秒