ISSN 2186-5647

2-130

テスト圧縮指向ドントケア抽出を用いた静的圧縮の評価

日本生産工(院)　○山崎 紘史　日立超 LSI システムズ　若園大洋
日大生産工　細川利典　九大　吉村正義

1. はじめに

近年，大規模集積回路(Very Large Scale Integrated circuits: VLSI)の大規模化，微細化により，テストパターン数の増大，故障モデルの多様化が問題となっている．VLSI の微細化により従来の縮退故障モデル[1,2]のテストパターンでは検出できない故障モデルとしてブリッジ故障モデル[3,4,5]や遷移故障モデル[6,7]が挙げられる．そのため，ブリッジ故障モデルや遷移故障モデルのテストパターンが必要であり，これらの故障モデルを検出するテストパターンが追加される．それに伴いテストパターン数が増加し，テストコストが増加する．

テストパターン数が増加する問題を解決する手法の一つがテスト圧縮[8]である．テスト圧縮にはドントケアを利用する手法[9,10]と，故障の被覆関係に基づく手法[11,12]が存在する．ドントケアに基づくテスト圧縮は，テストパターン中のドントケアを利用して複数のテストパターンを 1 つのテストパターンに圧縮する[9,10]．故障の被覆関係に基づくテスト圧縮は，故障シミュレーションを利用して冗長なテストパターンを削減する．故障の被覆関係に基づくテスト圧縮法として逆順故障シミュレーション[11] や二重検出法[12]が存在する．

一般に生成されたテストパターンの外部入力値は全て 0,1 に設定される．しかしながら，生成されたテストパターンの中には，逆の論理値に変更しても故障検出率が低下しない外部入力値が存在する．そのような外部入力値をドントケアという．入力されたテスト集合からドントケアを判定する，ドントケア抽出技術 XID[13]や DC-XID[14]が提案されている．ドントケア抽出技術 XID と DC-XID ではドントケアと判定される外部入力が異なる．XID はテスト集合におけるテストパターンの順番により，各テストパターンに含まれるドントケア数に偏りが生じる．そのため，XID は特定の適用分野で効果が薄い可能性がある．DC-XID では，各テストパターンの検出故障数を可能な限り均一化することにより，各テストパターンのドントケア数を均一化しており，低消費電力テストの分野で適用されている．テスト圧縮の分野での応用を考えると，特定の外部入力にケアビットが集中するとテスト圧縮の妨げになる可能性がある．過去に提案されたドントケア抽出法 XID と DC-XID ではテスト集合における外部入力ごとのドントケア数の均一化を考慮していない．そのため,テスト圧縮の分野では効果が薄い可能性がある．

本論文ではテスト集合における外部入力ごとのドントケア数をできる限り均一化することがテスト圧縮に効果的であると仮定する．その仮定に基づき，テスト集合における外部入力ごとのドントケア数の偏りとテスト圧縮の関係を解析し，テスト圧縮に効果的なドントケア抽出の定式化を行う．その定式化と解析結果に基づき，テスト集合における外部入力ごとのドントケア数をできる限り均一化するためのヒューリスティックなドントケア抽出手法を提案する．

本論文の構成は次のとおりである．第 2 章で従来のドントケア抽出技術[13]についての説明を行う．第 3 章でテスト集合における外部入力ごとのドントケア数の均一化とテスト圧縮の後のテストパターン数の関係に対して解析を行う．第 4 章でテスト圧縮を考慮したドントケア抽出法の提案,第 5 章で ISACAS'89, ITC'99 ベンチマーク回路を用いてドントケア抽出率とテスト圧縮率の評価，第 6 章で結論と今後の課題について述べる．

2. ドントケア抽出技術

2.1. ドントケア

テストパターンの中には逆の論理値に変更しても故障検出率が低下しない外部入力の値が存在する．そのような値をドントケアという．またドントケアでない外部入力をケアビットという．ドントケアと判定された値は，”0”もしくは”1”のいずれの論理値に設定することができる．ドントケアは”X”または”x”と示す．

2.2. ドントケア抽出の定式化

本論文では，自動テストパターン生成(Automatic Test Pattern Generator: ATPG)を用いて生成された 0，1 の論理値からなる初期テスト集合 T を入力とする．与えられた初期テスト集合 T に基づいて，ドントケアを含むテスト集合 XT を導出する．ドントケアを含むテスト集合 XT は以下の特性を持つ．

(1) XT は T を被覆する
(2)T と XT の故障検出率は等しい
(3)XT はできるだけ多くのドントケアを含む

3. ドントケアの分散とテスト圧縮の関係

3.1. 予備実験

本節では,外部入力ごとのドントケア数の偏りとテスト圧縮の関係について解析を行う．本実験では外部入力ごとのドントケア数の偏りの評価として分散を利用する．予備実験内容について説明する．まず ATPG を用いて生成した初期テスト集合 T に対して，ランダムでドントケアを割当て，外部入力におけるドントケアの分散値を操作したテスト集合 RXT を 1000 個作成する．各テスト集合 RXT の分散値はそれぞれ異なる．また故障検出率は考慮しない．その後，作成した各テスト集合 RXT に対してテスト圧縮を行う．予備実験の結果から外部入力ごとのドントケア数の偏りとテスト圧縮の関係について解析する．

$$X(rxt_i, p_j) = \begin{cases} 1 & \text{if primary input value } p_j \text{ of test pattern } rxt_i \text{ is an X-bit} \\ 0 & \text{otherwise (care bit)} \end{cases} \quad (1)$$

An Evaluation of Static Compaction using X-identification for Test Compaction

Hiroshi YAMAZAKI, Motohiro WAKAZONO, Toshinori HOSOKAWA, and Masayoshi YOSHIMURA

式(1)にドントケアを含むテストパターン $rxt_i \in RXT$ の外部入力 p_j の値を判定する関数 $X(rxt_i, p_j)$を示す. 式(1)はテストパターン rxt_i の外部入力 p_j の値がドントケアだった場合 1，ケアビットだった場合 0 を返す.

式(2)にテスト集合 RXT の各外部入力に含まれるドントケア数の平均を求める関数 $A_X(RXT)$を示す.

$$A_X(RXT) = \frac{1}{N_{PI}} \sum_{m=1}^{N_{PI}} \left(\sum_{n=1}^{N_{TP}(RXT)} X(rxt_n, p_m) \right) \quad (2)$$

ここで N_{PI} は外部入力数を示す. $N_{TP}(RXT)$はテスト集合 RXT に含まれるテストパターン数を示す.

式(1)と式(2)から，テスト集合 RXT の外部入力におけるドントケア数の分散を表す関数 $S^2(RXT)$を式(3)に示す.

$$s^2(RXT) = \frac{1}{N_{PI}} \sum_{i=1}^{N_{PI}} (A_X(RXT) - \sum_{j=1}^{N_{TP}(RXT)} X(rxt_j, p_i))^2 \quad (3)$$

3.2. 予備実験結果

図 1 に ITC'99 b14 ベンチマーク回路に対する外部入力のドントケア分散を考慮した予備実験結果を示す. 初期テスト集合 T_c は ATPG により作成し，動的圧縮と静的圧縮を行ったものである. この初期テスト集合 T_c に対して，ランダムでドントケアを割当て外部入力におけるドントケアの分散値を操作したテスト集合 RXT_c を作成した. テスト集合 RXT_c のドントケア割当て率は 70%，80%，90%の 3 種類を用いた.

図 1 の縦軸はテスト集合 RXT_c に対してテスト圧縮を行ったテスト集合 $CRXT_c$ に含まれるテストパターン数を示す. またテスト集合 RXT_c は Dsatur [9]を用いてドントケアに基づくテストパターン併合のテスト圧縮を行った. 図 1 の横軸はテスト集合 RXT_c の外部入力におけるドントケアの分散値を示す. また各グラフの「X-bit 70%」,「X-bit 80%」,「X-bit 90%」はそれぞれテスト集合 RXT_c 中のドントケアの割合が 70%，80%，90%を示す. また本実験は予備実験であるため，各プロットにおいて故障検出率は初期テスト集合 Tc を保証しない. 図 1 のドントケア率が 80%と 90%のグラフに着目すると，分散値が約 30,000 以下のプロットではテスト圧縮後のテストパターン数が大きく減少していることが分かる. 逆に分散値が 30,000 以上のプロットでは，ドントケア率が 90%であっても，テストパターン数は全く削減できていない. 次にドントケア率が 70%のグラフに着目すると，分散値を大きく変化させてもテストパターン数には大きな変化がないことが分かる. 図 1 の結果より外部入力におけるドントケア分散の低減はテスト圧縮に効果的であると考えられる. またテスト圧縮への効果は外部入力におけるドントケア分散に加えてドントケア数にも依存する.

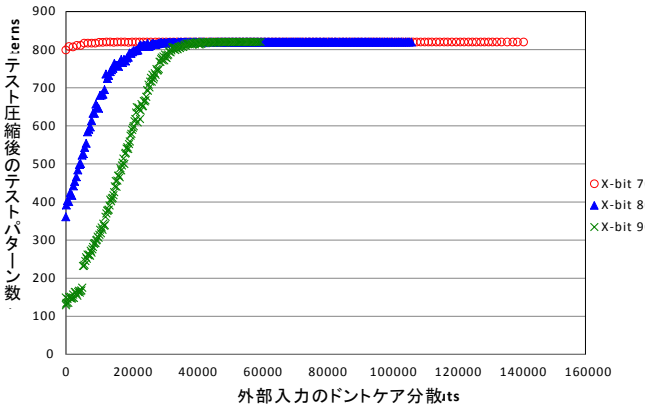

図 1. b14 回路に対する外部入力のドントケア分散とテスト圧縮後テストパターン数の関係

```
Procedure test_compaction_oriented_X-Identification(C,T,D,EXT,U,XT)
 Circuit C;  Initial_Test_Set  T ;  Fault_Dictionary D;
 Essential_Fault_Test_Set  EXT;  Undetect_Fault_Set  U;
 X_Identified_Test_Set  XT;
{
    for each testpattern t_i in T {
        D = fault_similation(C, t_i);                    Step1
    }
    EXT = essential_Xidentification(C, D,T );            Step2
    U = collect_undetected_fault(C, D, EXT );            Step3
    XT = distribution_X-filling(C, D, U, T, EXT);        Step4
    return XT;
}
```

図 2. テスト圧縮指向ドントケア抽出全体アルゴリズム

4. テスト圧縮指向ドントケア抽出

4.1. 問題の定式化

3 章より，外部入力ごとのドントケア数の分散値の低減がテスト圧縮に効果的であることがわかった. したがって，テスト圧縮に効果的なドントケア抽出は式(4)のように定式化される.

Inputs: initial test set T
Outputs: test set with X-bits XT
Constraint: X-bit ratio > n (%)
Minimization: $s^2(XT)$, subject to $D(T) = D(XT)$ (4)

ここで $s^2(XT)$はドントケア抽出後のテスト集合 XT における外部入力ごとのドントケア数の分散を表す. $D(T)$は初期テスト集合 T で検出する故障集合，$D(XT)$ はテスト集合 XT で検出する故障集合を表す. n はドントケア率の閾値を示し，XT のドントケア率は n%より大きくなければならない. しかしながら，現実的な時間で n%より大きいドントケア率かつ外部入力ごとのドントケア数の分散値を最小化する式(4)を求めることは困難である. それゆえ，本論文では外部入力ごとのドントケア数の分散値を可能な限り低減しつつ，ドントケア率を高めるヒューリスティックなドントケア抽出アルゴリズムを提案する.

4.2. 外部入力のドントケア分散コスト関数

本節では，提案手法であるテスト圧縮指向ドントケア抽出について説明する. 提案手法は外部入力ごとのドントケア数を可能な限り均一化するドントケア抽出技術である. 図 2 に提案手法であるテスト圧縮指向ドントケア抽出アルゴリズムを示す.

初期テスト集合 T の各テストパターン t_i に対して，故障シミュレーションを実行し故障辞書 D を作成する(Step 1). 故障辞書 D は各テストパターン t_i で検出可能な故障が記されている. Step1 で作成した故障辞書 D を用いて必須故障[12]集合を求める. 各初期テストパターン t_i に対して，必須故障の検出に必要な外部入力値を特定し，それ以外の外部入力の値をドントケアとしたテスト集合 EXT を得る(Step 2). EXT は必須故障のみを検出するテスト集合である. EXT は必須故障以外も検出する可能性がある.そのため，EXT に対して故

障シミュレーションを実行し，未検出故障集合 U を作成する(Step 3)．未検出故障集合 U に対して，外部入力に対するケアビットの分散を制御したドントケア抽出を行い，テスト集合 XT を得る(Step 4)．XT は初期テスト集合 T と等しい故障検出率なテスト集合である．本技術の詳細を 4.3 節で説明する．

4.3. テスト圧縮指向ドントケア抽出

本節では外部入力におけるドントケア分散の低減とドントケア数を考慮したドントケア抽出のヒューリスティックを説明する．この技術を用いた処理は，図 2 の Step4 である．

初期テスト集合 T に対して，ある未検出故障 f を検出することができるテストパターン数は一つとは限らない．そのため，未検出故障 f を検出するテストパターンを一つ選択する必要がある．しかしながら，テストパターン選択の際に外部入力のドントケア分散値のみを考慮するとケアビット数が増加し，ドントケア抽出率が低下する可能性がある．それゆえ，本論文では，外部入力におけるドントケア分散の低減とドントケア数増加を考慮した評価関数を提案する．

式(5)にドントケアを含むテストパターン xt_i の外部入力 p_j の値を判定する関数 $C(xt_i, p_j)$を示す．式(5) はテストパターン xt_i の外部入力 p_j の値がケアビットの場合 1，ドントケアの場合 0 を返す．

$$C(xt_i, p_j) = \begin{cases} 1 & \text{if primary input value } p_j \text{ of test pattern } xt_i \text{ is a care bit} \\ 0 & \text{otherwise (X-bit)} \end{cases} \quad (5)$$

式(6)にテスト集合 EXT の各外部入力 p_j に含まれるケアビット数を求める関数 $W(EXT,p_j)$を示す．$N_{TP}(EXT)$ はテスト集合 EXT に含まれるテストパターン数である．

$$W(EXT, p_j) = \sum_{i=1}^{N_{TP}(EXT)} C(ext_i, p_j) \quad (6)$$

式(7)に外部入力におけるドントケア分散の低減とドントケア数を考慮した評価関数 $VX(ext_i, xt'_i)$を示す．式(7)は未検出故障 f をどのテストパターンで検出するかを判定する評価関数である．本技術では，未検出故障 f を検出するテストパターンは評価関数 $VX(ext_i, xt'_i)$ の値が最小のテストパターンとする．式(7)に評価関数 $VX(ext_i, xt'_i)$を示す．

$$VX(ext_i, xt'_i) = \sum_{j=1}^{N_{PI}} W(EXT, p_j) \times X(ext_i, p_j) \times C(xt'_i, p_j) \quad (7)$$

式(7)において ext_i は必須故障のみを検出するテストパターン，xt'_i は未検出故障 f のみを検出するテストパターンである．また $X(ext_i, p_j)\times C(xt'_i, p_j)$ はテストパターン xt'_i の外部入力 p_j の値がケアビットかつテストパターン ext_i の外部入力 p_j の値がドントケアの時 1 を，それ以外の場合を 0 返す．つまりテストパターン ext_i で未検出故障 f を検出するために，外部入力 p_j の値をドントケアからケアビットに変更する場合 1 を，それ以外の場合は 0 を返す．そのため，未検出故障 f をテストパターン ext_i で検出する場合，テストパターン ext_i のケアビット数は $\sum_{j=1}^{N_{PI}} X(ext_i, p_j)\times C(xt'_i, p_j)$だけ増加する．そのため評価関数 $VX(ext_i, xt'_i)$の値が小さいテストパターンほど外部入力ごとのドントケア数の増加が少ないテストパターンとなる．

表 1. テスト集合 EXT と各外部入力に含まれるケアビット数 $W(EXT, p_j)$

	p_1	p_2	p_3	p_4	p_5	p_6	p_7
ext_1	x	x	c	c	x	c	x
ext_2	c	x	x	c	c	x	c
ext_3	x	x	c	c	x	x	x
ext_4	c	x	c	c	c	x	x
ext_5	x	x	c	c	x	c	x
$W(EXT, p_j)$	2	0	4	5	2	2	1

以下に未検出故障 f を検出するテストパターンの選択例を示す．表 1 にテスト集合 EXT の各外部入力に含まれるケアビット数の計算例を示す．EXT は図 2 の Step2 で求めた，必須故障のみを検出するテスト集合である．ext_1 から ext_5 はテストパターン番号，p_1 から p_7 は外部入力番号を示す．また"X"はドントケア，"C"はケアビットを表す．

例として，表 1 の外部入力 p_1 に対してケアビット数を求める．外部入力 p_1 は，テストパターン ext_2 と ext_4 においてケアビットが存在する．よって式(6)より $W(EXT, p_1)$=2 となる．

表 2 に未検出故障 f を検出するのに必要な外部入力のケアビットコストの計算例を示す．ここで未検出故障 f はテスト集合 T においてテストパターン t_1，t_3，t_5 で検出可能であるとする．例としてテストパターン ext_3 で未検出故障 f を検出する場合のケアビットコスト $VX(ext_3, xt'_3)$を示す．テストパターン ext_3 で未検出故障 f を検出する場合，外部入力 p_2，p_3，p_7 をケアビットにする必要がある．しかしながら外部入力 p_3 は表 1 の必須故障のみを検出するテスト集合の時点でケアビットであるため，外部入力 p_3 のケアビット数は増加しない．そのため式(7)のケアビット増加を示す $X(ext_3, p_3)\times C(xt'_3, p_3)$ は 0 となる．よって式(7)より $VX(ext_3, xt'_3)$=1 と計算される．同様にテストパターン ext_1，ext_5 で検出する場合のケアビットコストを算出すると $VX(ext_1, xt'_1)$=3，$VX(ext_5, xt'_5)$=2 となる．

表 2. 未検出故障 f を検出するために必要なケアビットコスト

	p_1	p_2	p_3	p_4	p_5	p_6	p_7	$VX(ext_i, xt'_i)$
xt'_1	c	c	x	x	x	x	c	3
xt'_3	x	c	c	x	x	x	c	1
xt'_5	c	c	x	x	x	c	x	2
$W(EXT, p_j)$	2	0	4	5	2	2	1	

各ケアビットコストの結果から $VX(ext_3, xt'_3)$が最小であるため，未検出故障 f を検出するテストパターンは ext_3 が選択される．そのためテストパターン ext_3 の外部入力 p_2 と p_7 をドントケアからケアビットに変更し，テスト集合 EXT を更新する．

5. 実験結果

本章では，提案手法の評価結果を示す．評価項目は，ドントケア抽出率と判定したドントケアを用いてテスト圧縮した後のテスト集合の数である．提案手法の比較対象は文献[13]と文献[14]で提案されたドントケア抽出法と，未検出故障を検出するテストパターン選択の際に外部入力のドントケア分散値が最小のものを選

択するドントケア抽出手法である. 文献[13]と文献[14]のドントケア抽出法はそれぞれ XID, DC-XID と示す. 対象回路は ITC'99 ベンチマーク回路と ISCAS'89 ベンチマーク回路である. 初期テスト集合 T は Synopsys 社の"TetraMAX™"(Synopsys)によって生成された縮退故障用のテスト集合を用いた. また初期テスト集合はテスト圧縮されたテスト集合 T_c を利用した.

表 3 にテスト集合 T_c に対するドントケア抽出率, 外部入力に対するドントケア分散値, テスト圧縮後のテストパターン数を示す. 表 3 の *Circuits* は回路名, N_{PI} は外部入力数, $N_{TP}(Tc)$はテスト集合 T_c のテストパターン数, *%X-bit* はテスト集合 XT_c に含まれているドントケアの割合, $S^2(XT_c)$はテスト集合 XT_c の外部入力のドントケア分散, *#Dsatur* は XT_c に対して Dsatur[9]を用いて圧縮した後のテストパターン数, *#DD* は XT_c に対して Dsatur[9]を用いて圧縮した後のテストに対して二重検出法[12]を用いて再度圧縮した後のテストパターン数を示す. また *XID* が文献[13]のドントケア抽出手法, *DC-XID* が文献[14]のドントケア抽出手法, *Proposed* が提案手法によるドントケア抽出を示す. 提案手法と XID の*%X-bit* を比較すると s13207, s15850, s35932, s38417, b15, b17 に対しては, 提案手法のほうが約 2%高い. s38584, b14, b20, b21, b22 に対しては XID のほうが約 1%高い. 提案手法と DC-XID の*%X-bit* を比較すると, 全ての回路に対して提案手法のほうが約 1%高い. $s^2(XT_c)$では, 全ての回路において XID, DC-XID より提案手法の方が低い分散値であり, 3~60%(平均 13%)分散値を削減できた. *#Dsatur* では b22 を除く全ての回路で提案手法が最も少ないテストパターン数であり, XID, DC-XID と比較すると最大 26 パターン(平均 8 パターン)の削減となった. b22 回路に対しては, XID が最も少ないテストパターン数であった. *#DD* では b22 を除く全ての回路で提案手法が最も少ないテストパターン数であり, XID, DC-XID と比較すると最大 15 パターン(平均 6 パターン)の削減となった. b22 回路に対しては, XID が最も少ないテストパターン数であった. また提案手法の*#Dsatur* と*#DD* を比較すると, 最大 20 パターン(平均 4 パターン)が削減された

6. まとめ

本論文では外部入力におけるドントケア分散がテスト圧縮にどのように関係するのかを示した. 予備実験の結果からテスト圧縮に効果的なドントケア抽出を定式化した. またテスト圧縮に有効なドントケア抽出のヒューリスティックを提案し, 評価実験を行った. 実験結果では 1 つの回路を除いた全ての回路においてテストパターン数を削減した. 外部入力における分散値では, 全ての回路において提案手法の方が従来手法よりも低い結果となった. また予備実験結果から, 外部入力におけるドントケアの分散値とドントケア抽出率がテスト圧縮に影響を与えることがわかった. 今後の課題として, よりテスト圧縮に効果的な指標の解析や, 他の故障モデルでの実験が挙げられる.

文 献

[1] H. Fujiwara, "Logic Testing and Design for Testability," MIT Press, 1985.

[2] M. Abramovici, M. A. Breuer and A. D. Friedman "Digital Systems Testing and Testable Design," IEEE Press, 1995.

[3] J. Ferguson and J. Shen, "Extraction and Simulation of Realistic CMOS Faults Using Inductive Fault Analysis," International Test Conference, pp. 475-484, 1988.

[4] M. Renovell, P. Huc, and Y. Bertrand, "The Concept of Resistance Interval: A New Parametric Model for Realistic Resistive Bridging Fault," 13th IEEE VLSI Test Symposium, pp. 184-189, 1995.

[5] T. M. Storey and W. Maly, "CMOS Bridging Fault Detection," International Test Conference, pp. 842-851, 1990.

[6] S.J. Wang, Y.T. Chen, and K. Shu-Min Li, "Low Capture Power Test Generation for Launch-off-Capture Transition Test Based on Don't-care Filling," IEEE International Symposium on Circuits and Systems, pp. 3683-3686, 2007

[7] A. Krstic and K.T. Cheng, "Delay Fault Testing for VLSI Circuits," Kluwer Academic Pub Press, 1998.

[8] N. Jha and S. Gupta, "Testing of Digital Systems," Cambridge University Press, 2002.

[9] D. Brelaz, "New Methods to Color the Vertices of a Graph," Communications of the ACM, Vol.22, Issue 4, pp. 251-256, 1979.

[10] P.Goel and B.C.Rosales, "Test Generation and Dynamic Compaction of Tests," Digest of papers of Test Conf., pp.189-192, 1979.

[11] L. N. Reddy, I. Pomeranz and S. M. Reddy, "ROTCO: A Reverse Order Test Compaction Technique," IEEE EURO-ASIC Conf., pp. 189-194, 1992.

[12] S. Kajihara, I. Pomeranz, K. Kinoshita and S. M. Reddy, "Cost-effective Generation of Minimal Test Sets for Stuck-at Faults in Combinational Logic Circuits," 30th ACM/IEEE Design Automation Conference, pp. 102-106, 1993.

[13] K. Miyase and S. Kajihara, "XID: Don't Care Identification of Test Patterns for Combinational Circuits," IEEE Trans. Computer-Aided Design of Integrated Circuits and Systems, Vol. 23, No. 2, pp. 321-326, 2004.

[14] K. Miyase, K. Noda, H. Ito, K. Hatayama, T. Aikyo, Y. Yamato, H. Furukawa, X. Wen and S. Kajihara, "Effective IR-Drop Reduction in At-Speed Scan Testing Using Distribution-Controlling X-Identification," IEEE/ACM International Conference on Computer-Aided Design, pp. 52-58, 2008.

表 3. 各ドントケア抽出法に対するドントケア抽出率とドントケア分散と圧縮後テストパターン数

Circuits	N_{PI}	$N_{TP}(T_c)$	XID				DC-XID				Proposed			
			%X-bit	$s^2(XT_c)$	#Dsatur	#DD	%X-bit	$s^2(XT_c)$	#Dsatur	#DD	%X-bit	$s^2(XT_c)$	#Dsatur	#DD
s13207	650	267	87.86	1676	262	262	91.13	1384	261	260	**91.66**	**1315**	**259**	**259**
s15850	600	130	79.02	430	127	126	79.74	451	127	127	**81.32**	**383**	**124**	**121**
s35932	1763	21	56.81	5	**21**	**21**	58.61	3	**21**	**21**	**59.09**	**2**	**21**	**21**
s38417	1524	104	75.54	401	104	104	76.78	387	104	104	**77.38**	**351**	**103**	**103**
s38584	1462	143	**83.65**	451	140	139	82.46	482	143	143	83.61	**407**	**135**	**135**
b14	277	749	**77.14**	21755	728	710	74.97	23272	737	708	75.73	**20961**	**715**	**703**
b15	485	459	85.41	5086	393	357	84.29	5916	398	367	**86.22**	**4641**	**372**	**352**
b17	1452	1056	91.33	13530	1033	1022	90.73	14229	1028	1022	**91.88**	**12302**	**1015**	**1012**
b20	522	754	**72.90**	18401	733	729	70.68	18921	728	727	71.49	**17296**	**721**	**717**
b21	522	762	**72.92**	20468	739	737	70.90	20537	737	737	71.80	**19194**	**726**	**725**
b22	767	640	**72.14**	18375	**621**	**621**	70.65	18694	638	638	71.25	**17415**	625	624