On Guaranteeing Capture Safety in At-Speed Scan Testing With Broadcast-Scan-Based Test Compression

Kazunari Enokimoto [1], Xiaoqing Wen [1], Kohei Miyase [1], Jiun-Lang Huang [2], Seiji Kajihara [1], and Laung-Terng Wang [3]

[1] Kyushu Institute of Technology, Iizuka, Fukuoka, Japan
[2] National Taiwan University, Taipei, Taiwan
[3] SynTest Technologies, Inc., Sunnyvale, CA, USA

Abstract – Capture safety has become a major concern in at-speed scan testing since strong power supply noise caused by excessive ***launch switching activity*** (LSA) at transition launching in an at-speed test cycle often results in severe timing-failure-induced yield loss. Recently, a basic ***RM*** (rescue-&-mask) test generation scheme was proposed for guaranteeing capture safety rather than merely reducing LSA to some extent. This paper extends the basic RM scheme to broadcast-scan-based test compression by uniquely solving two test-compression-induced problems, namely (1) ***input X-bit insufficiency*** (i.e., fewer input *X*-bits are available for LSA reduction due to test compression) and (2) ***output X-bit impact*** (i.e., output *X*-bits may reduce fault coverage due to test response compaction). This leads to the ***broadcast-RM*** (broadcast-scan-based rescue-&-mask) test generation scheme. Evaluations on large benchmark circuits and an industrial circuit of about 1M gates clearly demonstrate that this novel scheme can indeed guarantee capture **s**afety in at-speed scan testing with broadcast-scan-based test compression while minimizing its impact on both test quality and test costs.

Keywords – *test generation*; *test compression*; *test power*; *at-speed scan testing*; *power supply noise*; *capture safety*.

I. Introduction

Timing-related defects have become a major cause of failing *integrated circuits* (ICs) due to shrinking feature sizes, decreasing supply voltages, and increasing clock frequencies. As a result, in order to achieve sufficient IC quality, it has become mandatory to conduct delay testing, usually in the form of at-speed scan testing. However, with ever-growing circuit sizes, at-speed scan testing is being severely challenged by three problems, namely ***low test quality***, ***high test cost***, and ***excessive test power*** [1].

The problem of low test quality, mainly due to insufficient testing of small-delay defects, can be mitigated by timing-aware ATPG [2]. The problem of high test costs, due to ever-increasing test vector count, can be tackled with test compression [3]. The problem of excessive test power, due to non-functional switching characteristics in scan testing, can be addressed by low-power testing [4]. However, timing-aware ATPG usually causes test data inflation and increases test power impact. Therefore, test compression needs to be combined with low-power testing to achieve high test quality with low test costs and safe test power.

Due to its importance as stated above, ***low-power test compression*** has been intensively studied over the years [5], resulting in numerous solutions based on low-power *design for testability* (DFT) and/or low-power test generation on top of a code-based, a linear-decompression-based, or a broadcast-scan-based test compression scheme.

It is noteworthy that the purpose of low-power testing is not for reducing test power to some extent but for achieving ***test power safety***, i.e., making at-speed scan testing safe with regard to test power [6]. In the *launch-on-capture* (LOC) based at-speed scan testing, test power safety includes ***shift safety*** and ***capture safety*** as shown in Fig. 1.

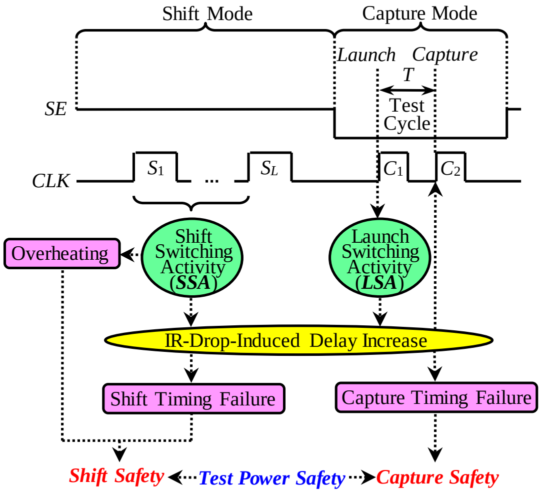


Fig. 1. Test power safety issues in LOC-based at-speed scan testing.

Generally, it is relatively easy to achieve shift safety. This is because circuit / clock change in shift mode has no impact on ATPG, CPU time, test vector count, and fault coverage. This fact makes it possible to apply effective and predictable low-power DFT techniques, such as scan chain segmentation [7], to bring ***shift switching activity*** (SSA), and thus shift power, down to a safe level.

Compared with shift safety, capture safety is highly elusive. Capture power of concern in LOC-based at-speed scan testing is due to ***launch switching activity*** (LSA) that is caused by test stimulus launching at the beginning of the at-speed test cycle *T*. As illustrated in Fig.1, the first capture C_1 may cause excessive LSA, lead to IR-drop in the *power supply network* (PSN), increase path delay, and finally result in timing failures at the second capture C_2. Since circuit / clock change in capture mode may severely impact ATPG, CPU time, test vector count, and fault coverage, it is preferable to use low-power test generation for reducing capture power [4]. However, although many techniques have been proposed for reducing LSA, most of them cannot guarantee that capture power be reduced under a safe level. This is, capture safety may not be achieved even though capture power is reduced to some extent.

Recently, we proposed a basic ***RM*** (rescue-&-mask based) test generation scheme [8] for guaranteeing capture safety. The key idea is to employ a two-phase ATPG by targeting at the real cause of any capture safety problem, i.e., the excessive LSA in the neighboring areas around long sensitized paths under a test vector. The ***rescue phase*** is to reduce excessive LSA in the neighboring areas of long sensitized paths in a focused manner (instead of unfocused global reduction), and the ***mask phase*** is to exclude from use in fault detection the uncertain test response at the endpoint of any long sensitized path that still has excessive LSA in its neighboring area after the rescue phase is executed. This way, the basic RM scheme guarantees capture safety with minimum impact on test quality and test costs. However, the major limitation of the basic RM scheme is its incompatibility with test compression.

This paper extends the basic RM scheme to broadcast-scan-based test compression [3], which is one of the test compression technologies widely used in the industry [9, 10]. The resulting ***broadcast-RM*** (broadcast-scan-based rescue-&-mask) test generation scheme features novel solutions to two test-compression-induced problems: (1) ***input X-bit insufficiency*** (i.e., fewer input *X*-bits are available for LSA reduction due to test compression) and (2) ***output X-bit impact*** (i.e., output *X*-bits may reduce fault coverage due to test response compaction). Evaluations on large benchmark circuits and an industrial circuit of about 1M gates demonstrate that the broadcast-RM test generation scheme can indeed guarantee capture safety in at-speed scan testing with broadcast-scan-based test compression while minimizing its impact on test quality and test costs.

The rest of the paper is organized as follows: Sect. II describes the background. Sect. III shows the general flow of the broadcast-RM scheme, and Sect. IV presents the details of three techniques for tackling the two test-compression-induced problems stated above. Sect. V shows evaluation results, and Sect. VI concludes the paper.

II. Background

A. LSP-Based Capture Safety Checking

The basic RM scheme [8] and the broadcast-RM scheme to be proposed in this paper both judge the capture safety of a test vector based on the following definitions [8]:

Definition 1: A ***long sensitized path*** (LSP) for a test vector *V* is a path that is sensitized by *V* and whose length is longer than a threshold (e.g., 80% of the structurally longest path).

Obviously, only such LSPs are susceptive to the impact of LSA in their neighboring areas as defined below:

Definition 2: The ***aggressor region*** of a gate *G*, denoted by *AR*(*G*), is composed of aggressor cells (gates and FFs) whose transitions strongly impact the supply voltage of *G*.

AR(*G*) can be readily identified by using layout and *power supply network* (PSN) design information [11]. In the most simplest form, *AR*(*G*) can be obtained by identifying gates that share a power via with the gate *G*.

Definition 3: The ***impact area*** of *P*, denoted by *IA*(*P*), consists of the aggressor regions of all on-path gates (G_1, G_2, ..., G_n) of *P*. That is, $IA(P) = AR(G_1) \cup AR(G_2) \cup ... \cup AR(G_n)$, as illustrated in Fig. 2.

Definition 4: An LSP *P* is said to be a ***risky path*** under a test vector *V* if (1) *P* is sensitized by *V* and (2) the LSA in the *IA*(*P*) under *V* is excessive (w.r.t. a threshold).

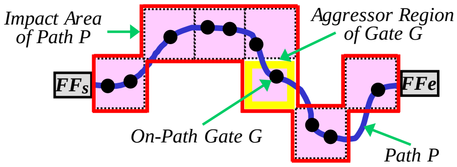


Fig. 2. Impact area.

Definition 5: A test vector *V* is said to be ***risky*** if *V* has at least one risky LSP.

B. Basic RM Test Generation Scheme

Fig. 3 illustrates the basic RM test generation scheme [8]. A test cube C_1 (with *X*s) is generated and then turned into a test vector V_1 (without *X*s) by fault-detection-oriented *X*-filling (e.g., *random-fill* for high test quality and small vector count). Conventional ATPG ends here, but the basic RM scheme continues with two more phases as follows:

- ***Rescue Phase***: *LSP-based capture safety checking* (①) is conducted to identify all ***risky paths*** under V_1. Here, suppose that P_a and P_b are risky paths. Then, ***impact-X-bit restoring*** (②) identifies those bits in V_1 that are originally *X*-bits in C_1 (before *X*-filling) and can reach the impact areas of P_a and P_b, and turns them back into *X*-bits (called ***impact-X-bits***) to create a new test cube C_2. After that, ***focused low-LSA X-filling*** (③) is conducted to turn C_2 into V_2 with reduced LSA in the impact areas of P_a and P_b. This way, some risky paths may be rescued into safe paths.
- ***Mask Phase***: *LSP-based capture safety checking* (④) is conducted again. Here, suppose that P_a is found to be safe now due to the rescue phase but P_b is found to be still risky under V_2. In this case, ***uncertain-test-response masking*** (⑤) is conducted to place an *X* at the endpoint (FF input) of P_b in the test response to V_2. This makes the uncertain value observed by the FF to be ignored in test response comparison, thus avoiding yield loss. Note that this masking needs no additional circuitry. *Also note that faults that originally detected by V_2 before masking but become undetected after masking will be targeted in subsequent test generation runs, thus avoiding fault coverage loss*.

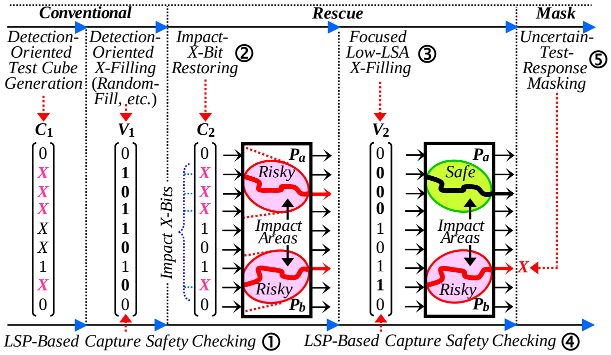
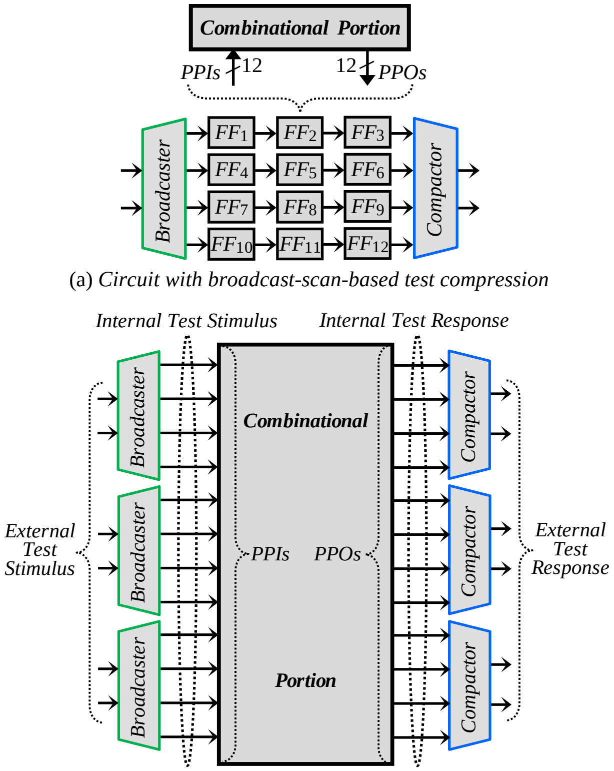

Conventional
Rescue
Mask
Detection-Oriented Test Cube Generation
Detection-Oriented X-Filling (Random-Fill, etc.)
Impact-X-Bit Restoring ②
Focused Low-LSA X-Filling ③
Uncertain-Test-Response Masking ⑤
C1
V1
C2
V2
Impact X-Bits
Pa
Pb
Risky
Safe
Impact Areas
X
LSP-Based Capture Safety Checking ①
LSP-Based Capture Safety Checking ④

Fig. 3. Basic RM test generation scheme.

It is clear that the basic RM scheme guarantees capture safety by the *mask phase*. In addition, the *recue phase* helps reduce test vector count inflation. This way, capture safety can be achieved in an effective and efficient manner.

C. Broadcast-Scan-Based Test Compression

Fig. 4 (a) illustrates a sample circuit with broadcast-scan-based test compression [10]. It has 4 internal scan chains of length 3 but only 2 external scan input/output pins. This gap is bridged by a ***broadcaster*** for input and a ***compacter*** for output. A broadcaster (compactor) is a combinational block for space expansion (space compaction). For test generation, the sample circuit is represented by a single combinational model shown in Fig. 4 (b). Note that ATPG directly generates ***external test stimuli*** (test cubes or test vectors) and ATE directly observes ***external test responses***.



(a) *Circuit with broadcast-scan-based test compression*

(b) *Circuit model for test generation*

Fig. 4. Broadcast-scan-based test compression.

D. Test-Compression-Induced Problems

Although the basic RM test generation scheme is highly effective in non-test-compression environments, it suffers from two test-compression-induced problems as follows:

- **Problem-1** (***Input X-bit Insufficiency***): The broadcaster forces severe constraints onto input *X*-bits in an external test cube, reducing their capability for reducing LSA. As a result, the rescue phase in the basic RM-based scheme becomes less effective and causes more uncertain-test-responses to be masked and more additional test vectors to be generated, resulting in test data inflation.
- **Problem-2** (***Output X-bit Impact***): The compactor makes masked internal uncertain-test-response bits interfere with other internal output bits, resulting in fault coverage loss at external output bits. As in Problem-1, this will lead to more additional test vectors, resulting in test data inflation.

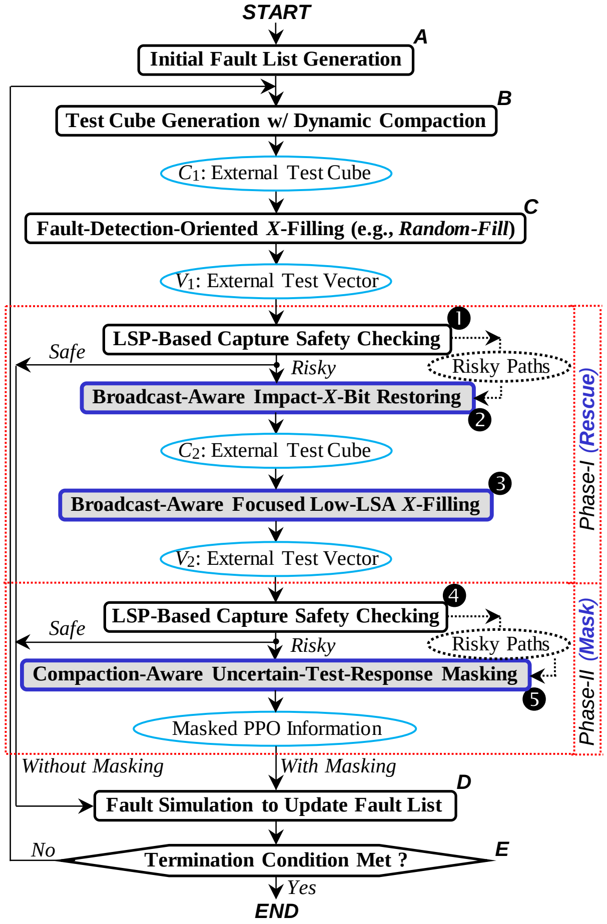
This paper extends the basic RM scheme into a test-compression-compatible low-power test generation scheme, called the ***broadcast-RM*** (broadcast-scan-based rescue-&-mask) scheme, with novel solutions to the two problems induced by broadcast-scan-based test compression.

III. Broadcast-RM Test Generation Scheme

Fig. 5 shows the broadcast-RM test generation scheme. It is similar to the basic RM scheme [8] but employs three novel techniques (❷,❸,❺) for solving the unique problems induced by broadcast-scan-based test compression.

Conventional ATPG consists of procedures ***A*** to ***E***. First, an initial fault list is generated (***A***). Then, a partially-specified external test cube C_1 is generated by detecting a primary fault and conducting subsequent dynamic compaction (***B***). After that, fault-detection-oriented *X*-filling is conducted, turning C_1 into a fully-specified external test vector V_1 (***C***). *Random-fill* is often used for this purpose since its fortuitous detection capability significantly improves test quality through unmodeled-defect detection and reduces test costs through smaller test data volume. Finally, the fault list is updated by fault simulation (***D***), and the termination condition is checked (***E***).

Conventional ATPG is enhanced to guarantee capture safety in broadcast-scan-based test compression by adding two phases (***Rescue*** and ***Mask*** that consist of ❶ to ❺), which is similar to the ones shown in Fig. 3. The differences are, instead of using *impact-X-bit restoring* (②), *focused low-LSA X-filling* (③), and *uncertain-test-response masking* (⑤) as in the basic RM scheme, the broadcast-RM scheme uses *broadcast-aware impact-X-bit restoring* (❷), *broadcast-aware focused low-LSA X-filling* (❸), and *compaction-aware uncertain-test-response masking* (❺), respectively.



START
A
Initial Fault List Generation
B
Test Cube Generation w/ Dynamic Compaction
C_1: External Test Cube
C
Fault-Detection-Oriented X-Filling (e.g., Random-Fill)
V_1: External Test Vector
❶
LSP-Based Capture Safety Checking
Safe
Risky
Risky Paths
Broadcast-Aware Impact-X-Bit Restoring
❷
C_2: External Test Cube
❸
Broadcast-Aware Focused Low-LSA X-Filling
V_2: External Test Vector
Phase-I (Rescue)
❹
LSP-Based Capture Safety Checking
Safe
Risky
Risky Paths
Compaction-Aware Uncertain-Test-Response Masking
❺
Masked PPO Information
Phase-II (Mask)
Without Masking
With Masking
D
Fault Simulation to Update Fault List
No
E
Termination Condition Met ?
Yes
END

Fig. 5. Broadcast-RM test generation scheme.

Note that *broadcast-aware impact-X-bit restoring* (❷) and *broadcast-aware focused low-LSA X-filling* (❸) are for solving Problem-1 (Sect. II. D), while *compaction-aware uncertain-test-response masking* (❺) is for solving Problem-2 (Sect. II. D). The details of these techniques will be described in the next section.

IV. Indiviual Techniques

A. Broadcast-Aware Impact-X-Bit Restoring

As shown in Fig. 5, the broadcast-RM test generation scheme starts from deterministically generating a partially-specified external test cube C_1 in $\boldsymbol{B}$ and then randomly X-filling it into a fully-specified external test vector V_1 in $\boldsymbol{C}$. Suppose that the sets of faults detected by C_1 and V_1 are $F(C_1)$ and $F(V_1)$, respectively. Generally, $F(V_1) > F(C_1)$ due to fortuitous detection in C. In Phase-I, *LSP-based capture safety checking* (❶) is conducted for V_1. As defined in Sect. II A, if some LSPs of V_1 have excessive LSA in their impact areas, those LSPs are risky paths and V_1 is risky. In this case, X-filling-based rescue will be conducted for V_1. For this purpose, ***broadcast-aware impact-X-bit restoring*** (❷) is first conducted to turn V_1 into a new external test cube C_2, which must satisfy the following two conditions:

- **Condition-1:** Preserve deterministic detection capability obtained in $\boldsymbol{C}$ by guaranteeing $F(C_2) \geq F(C_1)$.
- **Condition-2**: Each X-bit in C_2 must be able to reach the impact area of at least one risky path of V_1.

Note that for Condition-1, $F(V_1) > F(C_2)$ may occur, but faults detected by V_1 but undetected by C_2 will be detected in subsequent test generation runs, thus avoiding fault coverage loss. Also note that Condition-2 enables *focused low-LSA X-filling* (❸) for effectively reducing excessive LSA in the impact areas of risky paths in a pinpoint manner.

Impact-X-bit restoring (❷) in the basic RM scheme without test compression is simple and straightforward as illustrated in Fig. 3. However, *broadcast-aware impact-X-bit restoring* (❷) is more complex due to the existence of the broadcaster. The detailed procedure for this technique is described by using the example shown in Fig. 6.

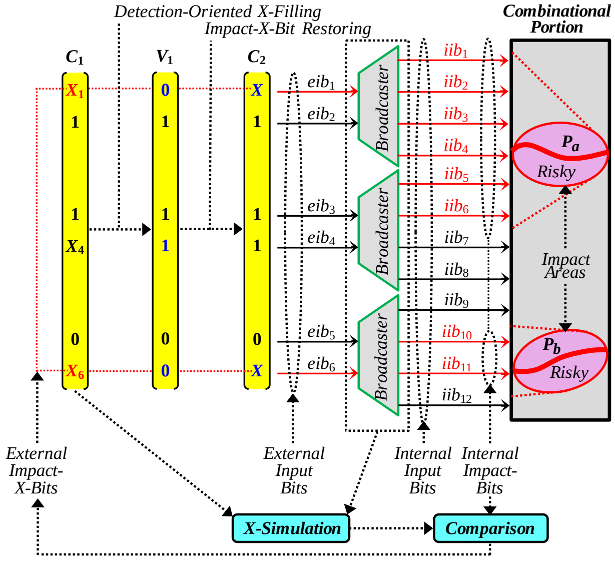


Fig. 6. Broadcast-aware impact-*X*-restoring.

In the example of Fig. 6, C_1 = <X_1 1 1 X_4 0 X_6> is a partially-specified external test cube generated in $\boldsymbol{B}$ as shown in Fig. 5. Logic values at external input bits eib_2, eib_3, and eib_5 in C_1 are deterministically generated for detecting targeted faults, while the other external input bits eib_1, eib_4, and eib_6 in C_1 are X-bits, denoted by X_1, X_4, and X_6, respectively. These X-bits are filled by a fault-detection-oriented X-filling method (e.g., *random-fill*) in $\boldsymbol{C}$ as shown in Fig. 5 for increasing fortuitous fault detection, resulting in a fully-specified external test vector V_1 = <**0** 1 1 **1** 0 **0**>. Suppose that *LSP-based capture safety checking* (❶) finds that V_1 has two risky paths: P_a and P_b. From the impact areas (as defined in Definition 3) of P_a and P_b, ***internal impact-bits*** for V_1 can be identified to be iib_1 through iib_6, iib_{10}, and iib_{11}, which are internal inputs bits that can reach the impact area of at least one risky path (P_a or P_b).

The next step is to identify ***external impact-X-bits*** for V_1, which are X-bits in C_1 that can reach at least one internal impact-bit for V_1. This can be simply done by conducting a static analysis on the broadcaster module to see if each X-bit in C_1 can structurally reach at least one internal impact-bit for V_1. However, the existence of the broadcaster often makes this simple method to yield inaccurate results. That is, an X-bit in C_1 may be found not to show up at any output of the broadcaster when logic values at other bits in C_1 are taken into consideration. This inaccuracy will not only result in the restoration of X-bits useless for reducing excessive LSA in impact areas but also cause significant test vector inflation. In order to address this issue that is part of Problem-1 (Sect. II. D), the procedure for *broadcast-aware impact-X-bit restoring* (❷) employs a partial symbolic simulation technique, called ***X-simulation*** [12].

In X-simulation, two types of X-symbols are propagated through the broadcaster module, which is a combination of three broadcaster copies. The initial X-bits X_1, X_4, and X_6 in C_1 are called ***primary X-symbols***. Any logic function (other than logic inversion) that involves primary X-symbols and cannot resolve into a logic constant (0 or 1) creates a ***compound X-symbol***. For example,

$$\text{NAND}(X_1, 1, X_6) = X(1, 6)$$

where $X(1, 6)$ is a compound X-symbol and (1, 6) means that it is originated from primary X-symbols X_1 and X_6. In addition, any non-constant logic function (other than logic inversion) involving primary and compound X-symbols also creates a *compound X-symbol*. For example,

$$\text{NOR}(X_4, 0, X(1, 6)) = X(1, 4, 6)$$

As for logic inversion, it is preserved as illustrated below:

$$\text{NAND}(X1(b1, b2), 1) = \overline{X}1(b1, b2)$$

By reviewing the result of X-simulation conducted for C_1 on the broadcaster module, external impact-X-bits for V_1 can be identified. For example, if $\overline{X}1(b1, b2)$ appears on an internal impact-bit, then both X_1 and X_2 are external impact-X-bits. As confirmed in evaluation experiments, X-simulation-based external impact-X-bit identification can achieve a good balance in terms of effect and CPU time.

After all external impact-X-bits for V_1 are identified, the corresponding logic bits in V_1 are turned into X-bits. This results in a new external test cube C_2 = <$\boldsymbol{X}$ 1 1 1 0 $\boldsymbol{X}$>, which satisfies both Condition-1 and Condition-2.

B. Broadcast-Aware Focused Low-LSA X-Filling

It is clear that all X-bits in the new external test cube C_2 obtained by *broadcast-aware impact-X-bit restoring* (❷) are related to the impact areas of the risky paths of V_1. Therefore, low-LSA X-filling on these X-bits can reduce excessive LSA exactly in those impact areas in a focused manner, as illustrated in Fig. 7.

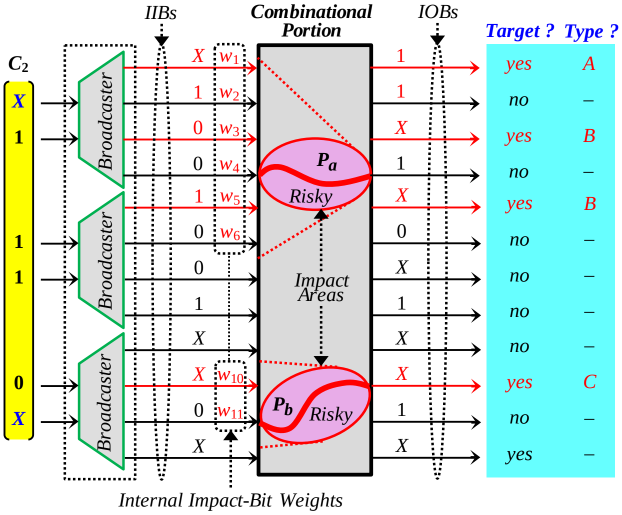


Fig. 7. Broadcast-aware focused low-LSA *X*-filling.

In Fig. 7, C_2 = <X 1 1 1 0 X> is applied to the circuit model. 3-valued simulation for C_2 obtains values (0, 1, X) at ***IIBs*** (*Internal Input Bits*) and ***IOBs*** (*Internal Output Bits*). Obviously, low-LSA X-filling only needs to target an IIB-IOB pair that has an X as either the IIB or IOB value and whose IIB is an internal impact-bit. The information on target IIB-IOB pairs is also shown in Fig. 7.

Target IIB-IOB pairs can be classified into three types: ***Type-A*** (X–0/1), ***Type-B*** (0/1–X), and ***Type-C*** (X–X), as shown in Fig. 7. Generally, X-filling for reducing LSA in the impact areas of the risky paths of V_1 is conducted by determining proper logic values for X-bits in C_2 to realize the goal of making the one X in a *Type-A*/*Type-B* IIB-IOB pair to have the same logic value as the corresponding binary bit and making the two Xs in a *Type-C* IIB-IOB pair have the same logic value, as much as possible.

Numerous previous low-LSA X-filling techniques have been proposed to realize the above goal by using the assignment/justification-based approach (e.g., *LCP-fill* [13]) or the signal-probability-based approach (e.g., *preferred-fill* [14]). However, these effective low-LSA X-filling techniques cannot be directly applied in the broadcast-scan-based test scheme. The reason is that the first bit in a bit-pair can no longer be directly controlled (through direct logic value assignment) in such a test compression scheme due to the existence of the broadcaster module as shown in Fig. 7. *This means that only the justification operation is applicable for realizing the above goal of low-LSA X-filling in the broadcast-scan-based test compression scheme*.

In order to address this issue that is part of Problem-1 (Sect. II. D), ***broadcast-aware focused low-LSA X-filling*** (❸) is proposed. The detailed procedure for this technique is illustrated in Fig. 7, and described below:

◆ ***Broadcast-Aware Focused Low-LSA X-Filling*** ◆

(1) For each internal impact-bit, identify the number of its reachable gates in the impact areas of risky paths. Use this number as its ***internal impact-bit weight***.

(2) Select the target IIB-IOB pair whose IIB has the largest internal impact-bit weight. Use the justification operation to make both IIB and IOB have the same logic value, as much as possible.

(3) If two target IIB-IOB pairs have equal internal impact-bit weights for their IIBs, select a target IIB-IOB pair in the order of *Type-A*→*Type-B*→*Type-C*. *Type-A* (X–0/1) as the highest priority since it requires justification to be conducted for an X-bit that is closer to the input side. This makes justification more likely to succeed and consume fewer X-bits in the test cube.

(4) Repeat steps (2) and (3) until there are no more X-bits in the test cube.

As confirmed in evaluation experiments, the number of target IIB-IOB pairs is usually not large. This fact makes justification-based low-LSA X-filling both feasible (i.e., CPU time will not be too long) and necessary (i.e., the LSA-reducing effect of a small number of X-bits needs to be maximized through deterministic justification).

C. Compaction-Aware Uncertain-Test-Response Masking

The *rescue phase* in the broadcast-RM scheme results in a new external test vector V_2 after *broadcast-aware impact-X-bit restoring* (❷) and *broadcast-aware focused low-LSA X-filling* (❸). However, some risky paths may still remain under V_2. Fig. 8 shows such an example.

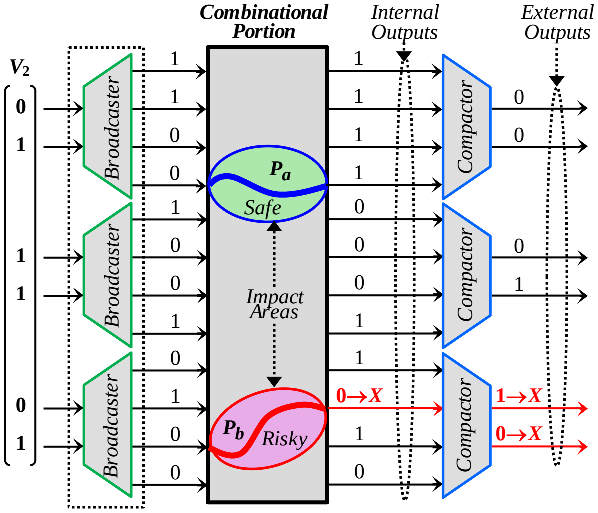


Fig. 8. Compaction-aware uncertain-test-response masking.

In Fig. 8, one risky path (P_b) remains. That is, an uncertain (possibly erroneous) output value may appear at the endpoint of P_b. In order to avoid undue yield loss, the value should be masked [8]. There are two issues due to the existence of the compactor: (1) masking one internal output may cause multiple external outputs (compactor outputs) to be masked, and (2) fault detection capability may decrease. In order to address these issues which are part of Problem-2 (Sect. II. D), ***compaction-aware uncertain-test-response-masking*** (❺) is proposed. The basic ideas are as follows:

(1) Use the number of compactor outputs that need to be masked due to a risky path as its ***mask cost***. In the *rescue phase*, try to reduce the excessive LSA in the impact areas of the risky paths of high mask costs.

(2) Use *X*-tolerant compactors such as *X*-compact [15] to mitigate fault detection capability degradation.

V. Evaluation Results

The proposed broadcast-RM test generation scheme illustrated in Fig. 5 was implemented in C with a commercial test compression tool (VirtualScan^{TM}) as the base ATPG with a 1:4 compression ratio. Six large ITC'99 benchmark circuits and an industrial circuit of about 1M gates were used in experiments on a workstation (Intel Xeon$^{®}$ 3.33 GHz CPU / 64GB main memory).

Table I shows circuit statistics & evaluation results. Test costs were evaluated by test data volume, while test quality was evaluated by transition-delay *fault coverage* (*FC*), *bridging coverage estimate* (*BCE*) [16] for assessing the capability of detecting unmodeled structural defects (e.g., bridging defects), and *statistical delay quality level* (*SDQL*) [17] for assessing the capability of detecting unmodeled small-delay defects. Changes in test data volume and test quality metrics (*FC*, *BCE*, *SDQL*) are shown under "*ATPG Results %Change*". In addition, the average number of LSPs per vector (*Ave. # of LSPs / Vec.*), the average number of risky paths per vector (*Ave. # of Risky Paths / Vec.*), the percentage of risky vectors (*% of Risky Vec.*) are shown under "*Capture Safety Checking*"; the percentage of impact-*X*-bits (*% of Impact-X-Bits*) and the ratio of focused low-LSA *X*-filling making risky paths into safe paths (*Rescue Ratio (%)*) are shown under "*Rescue*"; the average number of masked compactor outputs per vector (*Ave. # of Masked Comp. Outputs / Vec.*) are shown under "*Mask*".

Note that the proposed broadcast-RM test generation scheme always guarantees capture safety. Table I shows that test data volume increases in the broadcast-RM scheme due to test-compression-induced constraints. For comparison, there is virtually no change in test data volume in the basic RM scheme without test compression support [8]. However, this increase is modest and far from being excessive w.r.t. previous solutions [5]. It is also noteworthy that the large industrial circuit (*dpro*) caused a relatively small change in test data volume. In addition, the rescue success ratio for *dpro* was very high and the number of masked compactor outputs is very small. *This favorably indicates that the proposed broadcast-RM test generation scheme is effective and scalable for large industrial circuits*.

VI. Conclusions

This paper has presented the ***broadcast-RM*** (broadcast-scan-based rescue-&-mask) test generation scheme for guaranteeing capture safety (instead of just reducing global capture power to some extent) in at-speed scan testing with broadcast-scan-based test compression. Novel compression-aware low-power test generation techniques have been proposed so as to minimize the impact on test quality and test costs. Current evaluation results demonstrate the effectiveness and scalability of the broadcast-RM test generation scheme. Future work will be focused on fine-turning the proposed techniques to further reduce test vector count inflation, more evaluations will be conducted on industrial circuits.

References

[1] L.-T. Wang, et al., *System-on-Chip Test Architectures: Nanometer Design for Testability*, Morgan Kaufmann, San Francisco, 2008.

[2] X Lin, et al., "Timing-Aware ATPG for High Quality At-Speed Testing of Small Delay Defects," *Proc. ATS*, pp.139-146, 2006.

[3] N. A. Touba, "Survey of Test Vector Compression Technique," *IEEE Design & Test*, Vol. 23, Issue 4, pp. 294-303, 2006.

[4] J. Saxena, et al., "A Scheme to Reduce Power Consumption during Scan Testing," *Proc. ITC*, pp. 670-677, 2001.

[5] P. Girard, et al., *Power-Aware Testing and Test Strategies for Low Power Devices*, Springer, New York, 2009.

[6] V.R. Devanathan, et al., "A stochastic pattern generation and optimization framework for variation tolerant, power-safe scan test," *Proc. ITC*, Paper 13.1, 2007.

[7] L. Whetsel, "Adapting Scan Architectures for Low Power Operation," *Proc. ITC*, pp. 863-872, 2000.

[8] X. Wen, et al., "Power-Aware Test Generation with Guaranteed Launch Safety for At-Speed Scan Testing," *Proc. VTS*, pp.166-171, 2011.

[9] N. Sitchinava, et al., "Changing the Scan Enable during Shift," *Proc. ITC*, pp. 916-925, 2004.

[10] L.-T. Wang, et al., "VirtualScan: A New Compressed Scan Technology for Test Cost Reduction," *Proc. ITC*, pp. 916-925, 2004.

[11] M. Tehranipoor, et al., "Power Supply Noise: A Survey on Effects and Research," *IEEE Design&Test*, Vol.27, Issue 2, pp. 51-67, 2010.

[12] X. Wen, et al., "On Per-Test Fault Diagnosis Using the *X*-Fault Model," *Proc. ICCAD*, pp. 633-640, 2004.

[13] X. Wen, et al., "On Low-Capture-Power Test Generation for Scan Testing," *Proc. VTS*, pp. 265-270, 2005.

[14] S. Remersaro, et al., "Preferred Fill: A Scalable Method to Reduce Capture Power for Scan Based Designs," *Proc. ITC*, Paper 32.2, 2006.

[15] S. Mitra, et al., "*X*-Compact: An Efficient Response Compaction for Test Cost Reduction," *Proc. ITC*, Paper 11.2, 2002.

[16] B. Benware, et al., "Impact of Multiple-Detect Test Patterns on Product Quality," *Proc. ITC*, pp. 1031-1040, 2003.

[17] Y. Sato, et al., "Invisible Delay Quality - SDQM Model Lights Up What Could Not Be Seen," *Proc. ITC*, Paper 47.1, 2005.

TABLE I. Evaluation Results

Circuit	*# of Gates (K)*	*# of FFs*	***Conventional Flow*** (with capture risk)				***Proposed Flow*** (with guaranteed capture safety)										
			ATPG Result (Baseline)				*ATPG Result (% Change)*				*Capture Safety Checking*			*Rescue*		*Mask*	*CPU (Sec.)*
			Test Data Volume (MB)	*FC*	*BCE*	*SDQL*	*Δ Test Data Volume*	*Δ FC*	*Δ BCE*	*Δ SDQL*	*Ave. # of LSPs / Vec.*	*Ave. # of Risky Paths / Vec.*	*% of Risky Vec.*	*% of Impact-X-Bits*	*Resuce Ratio (%)*	*Ave. # of Masked Comp. Outputs / Vec.*	
b17	40	1,415	0.85	82.8	39.3	1358.4	+7.14	+0.55	+0.57	+0.16	2.49	0.32	0.59	31.08	0.00	0.08	2267
b18	99	3,320	0.88	78.0	46.6	1292.8	+9.88	+0.43	+0.86	+1.00	0.68	0.29	0.38	20.40	0.00	0.02	8016
b19	196	6,130	1.3	76.0	45.6	3329.1	+7.69	+0.26	+0.39	+0.29	5.58	5.38	0.90	29.59	0.00	0.13	64,837
b20	16	490	2.8	80.9	43.1	282.3	+9.93	+0.35	+0.05	+0.30	0.39	0.29	0.92	32.96	13.33	0.09	681
b21	17	490	8.8	83.1	42.9	187.8	+10.11	+0.30	+0.47	–0.16	3.95	3.74	3.48	36.96	0.00	0.32	1,262
b22	23	735	26.0	81.4	44.2	255.7	+9.57	–0.07	+0.45	+0.01	2.16	1.63	6.70	16.29	1.85	0.60	1,264
dpro	967	99,836	44.0	83.1	35.1	1383.0	+7.69	+0.47	+0.93	+3.17	2.46	0.25	0.39	16.98	60.00	0.03	55,688