Formal Test Point Insertion for Region-based Low-Capture-Power Compact At-Speed Scan Test

Stephan Eggersglüß*† Stefan Holst ‡ Daniel Tille§ Kohei Miyase‡ Xiaoqing Wen‡

*Institute of Computer Science, University of Bremen, 28359 Bremen, Germany - E-mail: segg@informatik.uni-bremen.de
†Cyber-Physical Systems, DFKI GmbH, 28359 Bremen, Germany
‡Kyushu Institute of Technology, Iizuka, Fukuoka 820-8502, Japan - E-mail: {holst,k_miyase,wen}@cse.kyutech.ac.jp
§Design-for-Test Group, Infineon Technologies AG, Munich, Germany - E-mail: daniel.tille@infineon.com

***Abstract*—Launch-Switching-Activity (LSA) is a serious problem during at-speed testing of integrated circuits, since localized LSA may lead to severe IR-drop and thus failures. The excessive LSA is conventionally mitigated by reducing the switching activity through special low-power test generation techniques, typically resulting in severe test pattern inflation and high test costs. This work introduces a novel concept of Low-Capture-Power Test Points (LCP-TPs), which are inserted to reduce switching activity in critical High-Capture-Power (HCP) regions. LCP-TPs also help in retaining high test compaction capability. An optimization-SAT based procedure is proposed to compute a small set of optimal LCP-TP locations for compact at-speed test sets with effective capture power reduction. Experimental results clearly demonstrate the advantages of LCP-TP insertion.**

I. Introduction

During manufacturing test, the power dissipation of an integrated circuit is much higher than during normal functional mode [1]–[3]. Especially during at-speed scan testing, high *Launch-Switching-Activity* (LSA) can be a serious problem. LSA is introduced by the transition launch in the initial time frame of the test. Excessive power dissipation in localized *High-Capture-Power* (HCP) regions is able to induce severe IR-drop during the capture cycle. This may cause a correct circuit under test to fail during testing [4], leading to over-testing and yield loss.

Current test techniques reduce the switching activity by generating special low-power tests or changing existing test patterns appropriately. This is done either globally or in a localized way. Generally, region- or layout-based switching activity reduction, e.g. as done in [5]–[7], is more focused and considered to be more effective.

The techniques can be roughly classified in three categories. X-filling, e.g. [8]–[10], leverages the circumstance that generated test vectors are partially specified. The unspecified values are assigned in such a way that switching activity is reduced as much as possible. However, the effectiveness is limited since it is applied as a post-ATPG process and the ATPG assignments cannot be changed anymore.

Low-Power ATPG techniques were proposed in [11], [12]. These techniques modify the search process directly to produce a test with reduced switching activity. However, these techniques suffer from severe test data inflation. A SAT-based post-processing technique has recently been proposed [13] that replaces problematic tests with new ones. The new tests detect all target faults and, by this, do not increase the pattern count while reducing the switching activity in HCP regions as much as possible. However, capture-power-safety can not fully guaranteed. In some situations, however, it is impossible to test a set of essential delay faults while reducing regional switching activity below a safe threshold because the activation and/or propagation of the target faults is tied to the activation of the HCP region. This is a problem especially for compacted test sets.

Design techniques can be used to modify the circuit in order to reduce LSA. The work in [14] uses clock-gating to disable parts of the circuit while the work in [15] disables scan chains. These techniques have in common that the controllability is reduced which leads to fault coverage degradation and/or test inflation. A general method to improve several test characteristics is the integration of *Test Points* (TPs) [16]–[25]. In particular, the approaches in [21]–[24] use TPs to improve test compaction. However, low-power aspects have not been addressed so far.

Given a fixed fault coverage, there is a direct relationship between test set size and frequency of HCP regions. If each fault is tested individually by one low-power test, there will not be any HCP region, because the switching activity necessary to sensitize each individual fault will be lower or equal the switching activity during normal operation. As the test set gets more compact, however, more faults are tested with each pattern. Since each tested fault requires a certain minimum number of transitions in the circuit to be tested, the overall switching activity per pattern will increase. Higher overall switching activity, in turn, increases the probability of HCP regions and their associated problems. This is the dilemma all low-power ATPG approaches run into. After all X-filling and pattern retargeting techniques have been applied, there is no freedom left to reduce the switching activity of a pattern further without removing essential target faults or splitting tests. This leads to either reduced fault coverage or to test inflation.

We provide an alternative to this problem by introducing *Low-Capture-Power TPs* (LCP-TPs). This type of TPs is used to reduce LSA in *High-Capture-Power* (HCP) regions and, at the same time, maintain fault coverage and a low test pattern count. In constrast to other design techniques, observability as well as controllability is enhanced locally. Based on a given compact test set and preliminary layout information, potential HCP regions are identified and a set of effective LCP-TPs locations is calculated. For the first time, the selection of effective TPs is formulated as a formal optimization problem for which powerful reasoning engines can be applied. Using this formulation, a minimal set of LCP-TPs is identified which allows for a detection of all target faults with the same number of test patterns while sufficiently reducing regional switching activity in HCP regions.

The paper is structured as follows. The next section introduces regular TPs as well as basic information about switching activity and formal reasoning. Section III proposes the con-

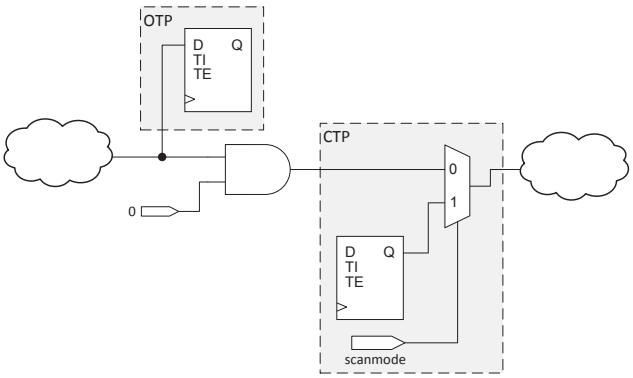

Fig. 1. Example Test Points

cept of LCP-TPs and Section IV presents the formulation as an optimization-SAT problem. Section V gives experimental results and Section VI draws conclusions.

II. Preliminaries

Test Point Insertion (TPI) is an important method in the design flow of integrated circuits. It modifies a design for improving DFT measures [16]. Conventionally, there are three main goals for TPI:

- Improvement of *random testability*, e.g. for a high coverage for BIST [17]–[20]. Areas with poor testability (which are determined by either computing signal probabilities or applying random pattern simulation) are enhanced such that observability or controllability improves.
- Improvement of *deterministic testability*, i.e. increasing the ATPG test coverage.
- Reduction of the *number of test patterns* [21]–[24]. Conflicting signal assignments are prevented which improves the feasibility of test compaction.

A further, less frequently reported goal of TPI is improving fault diagnosis [25].

TPI approaches usually directly modify the gate-level netlist. There are two different classes of test points:

- *Control Test Points* (CTP) – Here, a scan flipflop is inserted into the design to control the value of certain signals in test mode. Obviously, CTPs are introduced into locations where certain signals are hard to control, e.g. locations with a low signal probability.
- *Observation Test Points* (OTP) – An OTP is a scan flipflop, which is inserted in order to capture certain signals and, by this, make it observable. OTPs are inserted when it is difficult to propagate fault effects to observation points, e.g. in location with a low observability.

Figure 1 shows a very simple example to explain the usage of test points. Consider the AND gate. The lower input is fixed to 0 which blocks the gate. Therefore, the logic in its fanin cone is not observable and the logic in its fanout cone is not controllable. Here, we use TPI to improve testability.

An OTP is inserted in front of the AND gate which makes the logic testable. Likewise, a CTP is inserted after the AND gate such that the logic becomes controllable again. It can be observed that CTPs are more complex than OTPs. The reason is that it must be taken care that the normal mode functionality is not modified. Therefore, a MUX is inserted in order to select between normal functionality and CTP functionality. An OTP does not require such special handling. The scan flipflop has no controlling influence to the surrounding logic.

So far, test points have not been used in the field of power-aware testing. Compact test sets typically cause a larger amount of switching activity than the normal mode behavior. This could lead to IR-drop. Generally, switching activity is measured either globally or regionally. A global estimate can easily be obtained early in the design flow based on a logic netlist. The disadvantage is that the estimate is unfocused and a test with a low global switching activity can still have a very high regionally concentrated switching activity, which causes problems. However, layout information is needed for a regional estimate.

Generally, the power consumption of tests is analyzed using the *Weighted Switching Activity* (WSA) metric. Given a test vector t applied to a circuit C with signals S, the global WSA value is the sum of the WSA values for each signal $s_1, \ldots, s_n \in S$: $\sum_{i=1}^{n} \text{WSA}_{s_i}$. For a signal $s_i \in S$, the WSA value WSA_{s_i} is 0 if the signal s_i does not switch. If there is a transition on signal s_i, the WSA_{s_i} value is $1 + f$ with f as the number of fanouts of s_i.

The work in [13] presented a post-processing technique to reduce regional capture power of a test set. SAT-based optimization techniques were used to substitute a test vector which violates the switching activity threshold of a region by a test vector with regionally reduced switching activity. SAT solvers are powerful reasoning engines which work on a Boolean formula in *Conjunctive Normal Form* (CNF). A CNF Φ is a conjunction of clauses. A clause ω is a disjunction of literals. A literal is a Boolean variable in its positive (x) or negative ($\overline{x}$) form.

Besides classical SAT solvers which prove the satisfiability (by generating a solution) or unsatisfiability (by proving that no solution exists) of a CNF Φ, other solvers exist which use SAT solving techniques as basis but are able to process extended formulas, such as *Pseudo-Boolean* (PB) constraints or optimization functions.

A pseudo-Boolean constraint ψ associates constants with Boolean variables and has the following form: $c_1 \cdot x_1 + \ldots + c_n \cdot x_n \leq c_m$ with $c_1, \ldots, c_n$ as constants and $x_1, \ldots, x_n$ as associated literals. A PB constraint is satisfied when the sum of the constants whose associated literals are set, is less or equal than the constant c_m.

SAT-based optimization solvers typically accept an optimization function $\mathcal{F}$ in the following form: $\mathcal{F} = c_1 \cdot x_1 + \ldots + c_n \cdot x_n$. Instead of searching an arbitrary solution which satisfies the formula to prove the satisfiability of Φ, the result of the optimization process is the solution which satisfies Φ and, at the same time, minimizes the result of $\mathcal{F}$. By using an optimization function, it is possible to rate the solution and guide the search process towards finding the best solution. This can be leveraged in the field of ATPG in order to produce tests with specific properties, e.g. detecting a larger number of faults [26] or sensitizing longest paths [27], [28].

III. Low-Power Test Point Insertion

This section presents the basic idea and model of region-based *Low-Capture-Power Test Points* (LCP-TP). In a state-of-the-art design flow, ATPG methods receive a fault list as input and generate a test set detecting as many faults as possible with as few test vectors as possible. Especially for transition testing, this leads typically to a high switching activity, since a test vector which causes many transitions is better suited

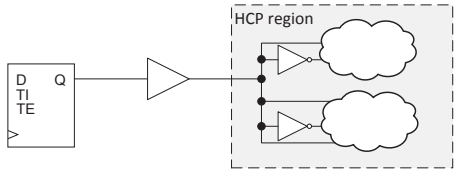

Fig. 2. Illustration of a High-Capture-Power Region

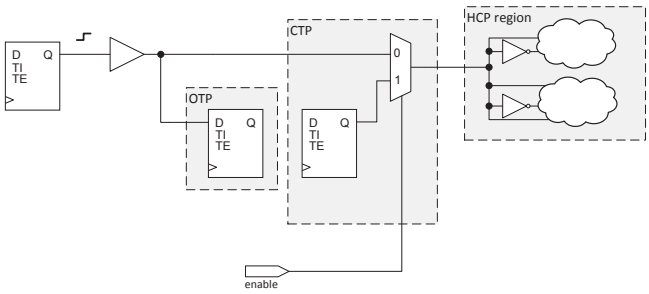

Fig. 3. Illustration of a Low-Capture-Power Test Point

to detect many transition faults. Dedicated low-power ATPG techniques typically cause severe test pattern inflation since they limit the number of transitions in an unfocused manner.

Depending on the design, the switching activity is often not evenly distributed across the circuit. Some regions are prone to high switching activity due to several reasons. One reason is that ATPG techniques often tend to excite similar easy-to-control or easy-to-observe signals in order to provide high compaction. This work presents a methodology to insert LCP-TPs in order to support the ATPG engine in generating a compact test set without *High-Capture-Power* (HCP) regions.

First, the idea of an LCP-TP is introduced. Figure 2 shows an illustration of an HCP region. Many paths can cross an HCP region for fault detection. There are three kinds of faults which cannot be tested without causing switching activity in the HCP region:

- *Region-fanIN* (RIN) faults – These are faults which lie in the fanin cone of an HCP region and whose observation path goes through the HCP region.
- *Region-fanOUT* (ROUT) faults – These are faults which lie in the fanout cone of an HCP region and whose justification paths go through the HCP region.
- *Region* (R) faults – These are faults which are located inside the HCP region.

In order to prevent that the observation paths of RIN faults go through the HCP region, an OTP can be inserted at the fanin boundary in front of of the HCP region. By this, the observation path of the transition fault ends before entering the HCP region. Since an OTP does not alter the function of the circuit, the transition will still be propagated into the HCP region. Therefore, this transition needs to be masked by introducing a CTP at the same location. In general, the OTP can be used also as a CTP. However, preliminary experiments have shown that the controllability of the R faults suffers, because two values are needed for transition testing. If the OTP is also used as a CTP, only the initial value can be introduced. The final value is provided by the observed value of the OTP in the initial time frame. In order to improve the controllability, an OTP and a CTP are introduced as an LCP-TP as shown in Figure 3. By this, the values of both time frames can be controlled.

Inserting LCP-TPs at the fanin boundary of an HCP region may result in a reduced testability of ROUT faults. This is because transitions necessary for ROUT fault detection are masked by the LCP-TPs described above. Therefore, LCP-TPs can also be inserted at the fanout boundary of an HCP region. The primary reason for inserting these LCP-TPs is not the observation of transitions (although they can be used for it) but the (re-)introduction of transitions for ROUT fault detection. These can be implemented the same way as shown in Figure 3 but using the signals at the fanout boundaries.

Given an identified HCP region, potential locations for LCP-TPs are therefore signals at the input as well as output boundary of this region. However, LCP-TPs are expensive in terms of area overhead. Therefore, it is not possible to insert LCP-TPs at all possible boundary locations. The aim of this work is to provide a small number of important LCP-TPs which can effectively be used to mask transitions and control signals to reduce the switching activity of an identified HCP region, while testing all essential faults with a compact test set. A computation method to provide such a set of LCP-TPs is presented in the next section.

IV. Overall Flow

Given is a circuit C and layout information such that the circuit die can be partitioned in regions as done in [5]–[7], [10], [13]. Each signal/gate is associated to one region. ATPG is used to generate a compact transition fault test set T. The test set is simulated and HCP regions, i.e. regions with a WSA value above a certain threshold, are identified for each $t \in T$ [13]. Test vectors that have at least one HCP region are denoted as HCP tests. The set of HCP tests is given by T_{HCP}. The set of essential faults of each HCP test $t_{\text{HCP}} \in T_{\text{HCP}}$ is denoted by f^e_{HCP}. This includes all faults which are detected by HCP tests only. The following general flow is used to identify LCP-TPs.

1) Test vector analysis – identification of HCP regions, HCP tests and essential faults
2) Test vector LCP ATPG retargeting to substitute HCP tests by non-HCP tests if possible (Section IV-A)
3) Potential LCP-TP insertion at the boundaries of remaining HCP regions
4) Optimization-SAT-based procedure to identify effective LCP-TPs (Section IV-B)

After the test vector analysis and the identification of the HCP regions and HCP tests, an ATPG procedure is started to check whether an HCP test can be substituted by a non-HCP. This includes the detection of all essential faults of the HCP test without violating the threshold of the HCP regions. This step is used to check whether the switching activity can be reduced in a sufficient manner by pure ATPG techniques and without generating additional tests. This is often not possible and there is a remaining set of HCP tests and HCP regions. These are used as a target for LCP-TP identification in the next steps.

The remaining HCP regions are analyzed and potential LCP-TPs are inserted at the input and output boundaries of all HCP regions. Note that the LCP-TPs are only inserted temporarily (in the formula) for reason of computation. Each LCP-TP has a separate enable signal. By this, LCP-TPs can be dynamically enabled/disabled during computation. Furthermore, the observation point is gated with an additional AND gate for computation only (Figure 4). By this, the enable signal is able to disable the OTP during computation. This is necessary to determine whether the OTP is used. The actual LCP-TP does not need this AND gate. If the enable signal

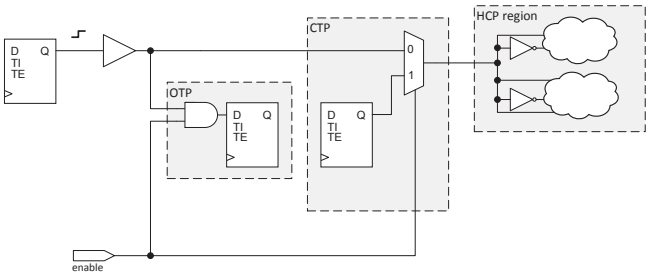

Fig. 4. Computational Model of Gated OTP

of each potential LCP-TP is set to 0, the circuit assumes its normal behavior.

The aim of the optimization-SAT procedure is to identify which LCP-TPs are necessary to detect all essential faults of an HCP test and, at the same time, do not violate the threshold of the HCP region. Therefore, the target of the optimization procedure is the minimal number of LCP-TPs which are able to satisfy both constraints. The detailed problem formulation to obtain the LCP-TPs are described in the following.

A. ATPG formulation with Threshold Constraint

This section describes how a threshold constraint for an HCP region can be incorporated into a SAT-based ATPG formulation. This constraint is necessary for the LCP ATPG retargeting process to substitute an HCP test by a non-HCP test.

The ATPG problem to retarget an HCP test t_{HCP} receives the list of essential faults F^e_{HCP} as input as well as the set of HCP regions R. Generally, the ATPG problem is formulated as a multiple-target SAT problem [13], [26]. Since it is known that all faults $f \in F^e_{\text{HCP}}$ are compatible, i.e. they can be tested with one test, the SAT instance is constructed in such a way, that the solution space consists of all tests which detect all faults $f \in F^e_{\text{HCP}}$:

$$\Phi_{F^e} = \Phi_g \cdot \Phi_{f_1} \cdot \ldots \cdot \Phi_{f_n}$$

with Φ_g as the fault-free circuit and $\Phi_{f_1} \cdot \ldots \cdot \Phi_{f_n}$ the necessary constraints to detect the essential faults $f_1, \ldots, f_n$. Detailed information about the SAT instance construction can be found in [26].

When the SAT instance Φ_{F^e} is given to a SAT solver, the result is a variable assignment from which a test vector can be extracted which detects all target faults. Since this does not guarantee that no region threshold is violated, additional constraints are necessary for each HCP region. The switching activity of a region can be encoded similarly to the approach in [13]. Given an HCP region r and a set of all signals S_r located in the region, a Boolean variable s^t is assigned to each signal $s \in S$ in addition to the Boolean variables s^1 and s^2 which encode the value of the signal s in both time frames. Implications are added which guarantee that s^t is assigned to 1 if the assignments of s^1 and s^2 differ. If s^1 and s^2 have the same value, the value of s^t is 0 (per implication).

In order to exclude all solutions, i.e. tests, which violate the region WSA threshold, from the solution space, a pseudo-Boolean constraint is added for each HCP region to the SAT instance. This constraint associates the WSA value of a signal to its assigned value. Since the WSA value is either 0 or has a static value, the constraint can be integrated into the ATPG problem formulation. For an HCP region r, a set of signals $s_1, \ldots, s_m \in S_r$, their associated WSA values (in case of an active transition) $w_1, \ldots, w_m$ and the pseudo-Boolean constraints for the HCP region r with WSA threshold w_t is formulated as follows:

$$\psi_r = w_1 \cdot s^t_1 + \ldots + w_m \cdot s^t_m \leq w_t$$

with $s^t_1, \ldots, s^t_m$ as positive literals of the corresponding Boolean variables. When the constraint is evaluated, all constants w_i are accumulated if the corresponding literal s^t_i is assigned to 1, i.e. a transition occurs. That means, the result of the evaluation is the WSA value for the region r. Since this value is compared to the WSA threshold of the region, the constraint evaluates to false when the WSA value of the assignment violates w_t. In summary, the following (PB-)SAT instance is given to a (PB-)SAT solver:

$$\Phi^{w_t}_{F^e} = \Phi_{F^e} \cdot \psi_r.$$

If $\Phi^{w_t}_{F^e}$ is proven to be satisfiable, a test can be extracted which detects all target faults and does not violate the WSA threshold of the target HCP region.[1] Therefore, there is no need to perform LCP-TP insertion for this set of faults. If the problem instance is proven to be unsatisfiable, there exists no test which detects all faults and does not violate the targeted WSA thresholds. In this case, the target faults and HCP regions are used for the LCP-TP identification.

B. LCP-TP Identification Using Optimization

This section describes the procedure to identify effective LCP-TPs, given a set of HCP tests which could not be replaced by a non-HCP test. As described above, potential LCP-TPs are inserted into the circuit structure first which could help to make these faults testable under the WSA threshold constraint.

The same procedure as presented in Section IV-A can be used as a basis problem formulation. The only difference is that the SAT instance, i.e. the Boolean formula, is extracted for the updated circuit structure with all potential LCP-TPs included. This is a straight-forward extension.

If the LCP-TPs can be used to detect the essential faults without violating a region WSA threshold, the solving engine will generate such a test using the basis formulation as a (PB-)SAT problem. However, it is most likely that the solver will enable all or a large number of the inserted test points since these can easily be used to control necessary signals without time-consuming justification, although many of them are actually not needed.

Since LCP-TPs are costly in terms of hardware overhead, the aim is to identify a small number of relevant LCP-TPs which can be effectively used to reduce the switching activity in the HCP regions. Therefore, the problem formulation is extended with an optimization function $\mathcal{F}$. This function is used to minimize the number of enabled potential test points which are needed to achieve the specified goal. Given to a solver, the search process internally enumerates solutions in an iterative manner until the minimum is found.

The proposed LCP-TP modelling allows for an easy formulation of the minimization function. Since each LCP-TP has an enable signal, $\mathcal{F}$ is formulated over the variables which correspond to the enable signals of the LCP-TPs. Although, all TPs can be enabled by the same signal after implementation, e.g. the scan mode signal, the computational model assigns a separate enable signal to each potential LCP-TP.

[1] The generated test has to be analyzed as well whether the WSA thresholds of other regions are violated. If this is the case, then the procedure has to be repeated with updated HCP regions.

Given a set of potential LCP-TPs $tp_1, \ldots, tp_k$ and the corresponding enable signals/variables $e_1, \ldots, e_k$, the minimization function is formulated as follows:

$$\mathcal{F} = 1 \cdot e_1 + ... + 1 \cdot e_k$$

If $e_i = 1$ holds for an enable signal, the LCP-TP tp_i is used. If $e_i = 0$ holds, the LCP-TP tp_i is disabled. The result of the evaluation of $\mathcal{F}$ for an assignment is the number of enabled signals.

The (PB-)SAT instance $\Phi_{F^e}^{w_t}$ as well as the minimization function $\mathcal{F}$ is given to a pseudo-Boolean optimization solver. The (PB-)SAT instance describes the search space. Each solution to the instance corresponds to a test detecting all target faults without violating the target HCP regions. The minimization function is used to rate the solution according to the number of used LCP-TP enable signals.

Note that the solver does not explicitly enumerate all solutions but uses effective learning techniques to traverse the search space towards finding the best solution, i.e. the solution with the minimal number of enabled LCP-TPs.

In order to provide a small LCP-TP set, the LCP-TPs which were computed for one HCP test are fixed as active in the upcoming calculations for the other HCP tests. By this, only the minimal number of additional needed TPs will be computed. The computed LCP-TPs are saved and can be used for test point insertion as relevant TPs for regional switching activity reduction as well as a countermeasure against severe pattern inflation.

V. Experiments & Discussion

This section presents the experimental results of the proposed approach. The procedure was implemented in C++. Experiments were carried out on an Intel Xeon E3-1240 (GNU/Linux, 64bit, 32GByte RAM, 3.4 GHz) in single-threaded mode. Layout design of the benchmark circuits was conducted by using a commercial tool to obtain the layout information (DEF). We used the techniques in [26] to obtain two different (launch-on-capture) transition fault test sets with different compaction settings.

Table I and Table II present the results for the two different test sets. The number of regions for each circuit can be found in the column *#Reg*. A functional simulation over thousands of clock cycles was carried out. Based on these results, the WSA thresholds were determined. This value is given in column *WSA th*. Column *#Pat* gives the overall number of generated tests. The number of test vectors which violate the WSA threshold is shown in column *#HCP t*, while the total number of regions which were violated at least once is given in column *#HCP Reg*.

There are results for two steps. The results for the first step of SAT-based test generation with regional thresholds is given in *Step 1 - SAT*. In this step, no LCP-TPs are introduced. The data given in this column represents the remaining violations after applying this step. Based on the results of Step 1, potential LCP-TPs were inserted into the netlist. The number of these potential test points is given in *# Pot. LCP-TPs*. The remaining violating tests after test point insertion are given in column *Rem. HCP t* and the number of actual used LCP-TPs is given in column *# LCP-TPs*. The overall run time of the analysis, Step 1 and Step 2 is also given.

It can be seen that the proposed methodology is very effective. After the optimization-based TPI insertion, no violating test vector is left for both test sets. Depending on the circuit, a large share of potential TPs can be ruled out. However, Step 1 is very important to reduce the number of violating tests and violated regions, respectively. Here, a difference between the test sets can be observed. The larger test set typically has also more HCP tests. This can be explained with the larger number of test vectors. After applying Step 1, the more compact test set has slightly more HCP tests in total. The HCP tests of the larger test set were obviously easier to eliminate.

However, due to the high compactness of Test Set #2, it is also harder to reduce the switching activity below the WSA threshold. This can be seen from the fact that the number of LCP-TPs is increased by a factor of 5 for Test Set #2 compared to Test Set #1. Only about 15% of the potential LCP-TPs are needed in total. Test Set #2 needs nearly 30% of the potential LCP-TPs. Note that Table I contains circuits where HCP tests can be eliminated in Step 2 without any LCP-TPs. This is due to the fact that the set of essential faults slightly changed due to the newly generated tests in Step 1.

Overall, the results show that LCP-TPs can be inserted to reduce the regional switching activity very effectively without any test inflation. The results also show that the compactness of the test set is an important factor to consider and has large influence on the efforts which are necessary for eliminating HCP regions.

A. Discussion & Future Work

Test point insertion generally influences the layout of the design. After desired TPs have been identified, the layout of the circuit has to be updated to include these TPs. This creates the risk that the HCP regions can be changed. However, depending on the number of TPs, it should be possible for the layouter to mitigate the risk that the circuit layout may change drastically. Future work will target possible solutions to address this problem, e.g. by the use of spare cells or specific layout constraints.

Another future work will also deal with the consideration of critical paths. This paper proposes the general LCP-TP methodology. However, inserting TPs may shorten critical paths, which is disadvantageous for the test quality. Therefore, we will address this problem by including constraints on preserving critical paths.

Furthermore, heuristics can be used to allow for a small amount of additional test vectors in order to further reduce the number of necessary LCP-TPs and save hardware overhead.

VI. Conclusions

Low-power ATPG techniques typically lead to severe test inflation and increased test costs. There is a direct connection between the test set size and the frequency of *High-Capture-Power* (HCP) regions. We have introduced *Low-Capture-Power Test Points* (LCP-TPs). These LCP-TPs can be inserted in order to reduce the switching activity in HCP regions during testing. A formal optimization method has been proposed to calculate relevant positions to insert LCP-TPs. Experimental results show that LCP-TPs can effectively be used to reduce the switching activity of HCP regions without test inflation.

Acknowledgment

The work of S. Eggersglüß has been supported by the Institutional Strategy of the University of Bremen, funded by the German Excellence Initiative and by the German Research Foundation (DFG) under contract number EG 290/5-1. The work of S. Holst, K. Miyase and X. Wen has been supported by JSPS Grant-in-Aid for Scientific Research (B) #25280016 and JSPS Grant-in-Aid for Scientific Research on Innovative Areas #15K12003.

TABLE I. EXPERIMENTAL RESULTS – LCP TEST POINT INSERTION - TEST SET #1

Setup						Step 1 - SAT		Step 2 - Opt. TPI			Overall
Circ	#Reg	WSA th.	#Pat	#HCP Reg	#HCP t	#HCP Reg	Rem. #HCP t	# Pot. LCP-TPs	# LCP-TPs	Rem. HCP t	Run time
s15850_lay	64	150	157	3	28	2	2	434	1	0	30.9s
s35932_lay	196	135	129	21	54	6	4	863	6	0	173.4s
s38417_lay	156	195	197	14	45	6	6	1281	411	0	2529.3s
s38584_lay	132	155	325	9	96	6	16	1591	127	0	2390.2s
b11_lay	16	105	147	4	13	2	2	257	1	0	1.7s
b12_lay	36	105	452	5	32	3	4	776	270	0	183.4s
b13_lay	30	45	54	1	2	1	2	47	1	0	0.1s
b14_lay	132	220	934	14	167	9	20	334	261	0	9.9s
b15_lay	225	275	2018	1	98	1	14	565	0	0	10.0s
b17_lay	441	330	2167	13	98	3	14	1173	246	0	2177.8s
b20_lay	289	270	1021	5	173	4	4	2031	0	0	48.5s
b21_lay	289	270	1074	7	210	3	16	1650	283	0	2428.9s
b22_lay	342	275	1065	7	237	2	28	225	184	0	124.72s
total				104	1253	48	132	11227	1791	0	

TABLE II. EXPERIMENTAL RESULTS – LCP TEST POINT INSERTION - TEST SET #2

Setup						Step 1 - SAT		Step 2 - Opt. TPI			Overall
Circ	#Reg	WSA th.	#Pat	#HCP Reg	#HCP t	#HCP Reg	Rem. #HCP t	# Pot. LCP-TPs	# LCP-TPs	Rem. HCP t	Run time
s15850_lay	64	150	117	3	17	2	2	434	89	0	54.3s
s35932_lay	196	135	86	11	43	14	13	2116	836	0	504.4s
s38417_lay	156	195	142	14	31	8	7	1530	628	0	708.2s
s38584_lay	132	155	182	7	47	5	18	1317	332	0	3759.0s
b11_lay	16	105	125	3	13	2	2	257	2	0	14.8s
b12_lay	36	105	366	5	13	4	5	982	290	0	167.7s
b13_lay	30	45	52	1	3	1	2	47	1	0	0.1s
b14_lay	132	220	649	11	98	10	15	3223	1280	0	8542.8s
b15_lay	225	275	1440	1	63	1	10	565	13	0	440.8s
b17_lay	441	330	1592	11	53	5	12	1621	466	0	4991.5s
b20_lay	289	270	693	3	98	2	3	1474	297	0	1432.9s
b21_lay	289	270	743	4	110	2	9	1414	250	0	224.1s
b22_lay	342	275	705	15	223	9	45	3930	1089	0	1899.8s
total				89	812	65	143	18910	5573	0	

REFERENCES

[1] J. Saxena, K. M. Butler, V. B. Jayaram, S. Kundu, N. V. Arvind, P. Sreeprakash, and M. Hachinger, "A case study of IR-drop in structured at-speed testing," in *International Test Conf.*, 2003, pp. 1098–1104.

[2] P. Girard, N. Nicolici, and X. Wen(Eds.), *Power-Aware Testing and Test Strategies for Low Power Devices*. Springer, 2009.

[3] M. Tehranipoor and K. M. Butler, "Power supply noise: A survey on effects and research," *IEEE Design & Test of Computers*, vol. 27, no. 2, pp. 51–67, 2010.

[4] K. Asada, X. Wen, S. Holst, K. Miyase, S. Kajihara, M. A. Kochte, E. Schneider, H.-J. Wunderlich, and J. Qian, "Logic/clock-path-aware at-speed scan test generation for avoiding false capture failures and reducing clock stretch," in *IEEE Asian Test Symp.*, 2015, pp. 103–108.

[5] V. R. Devanathan, C. P. Ravikumar, and V. Kamakoti, "Glitch-aware pattern generation and optimization framework for power-safe scan test," in *VLSI Test Symp.*, 2007, pp. 167–172.

[6] J. Lee and M. Tehranipoor, "A novel pattern generation framework for inducing maximum crosstalk effects on delay-sensitive paths," in *International Test Conf.*, 2008, pp. 1–10.

[7] S. Kiamehr, F. Firouzi, and M. B. Tahoori, "A layout-aware X-filling approach for dynamic power supply noise reduction in at-speed scan testing," in *IEEE European Test Symp.*, 2013, pp. 52–57.

[8] S. Remersaro, X. Lin, Z. Zhang, S. M. Reddy, I. Pomeranz, and J. Rajski, "Preferred fill: A scalable method to reduce capture power for scan based designs," in *International Test Conf.*, 2006, pp. 1–10.

[9] X. Wen, K. Miyase, T. Suzuki, S. Kajihara, Y. Ohsumi, and K. K. Saluja, "Critical-path-aware X-filling for effective IR-drop reduction in at-speed scan testing," in *Design Automation Conf.*, 2007, pp. 527–532.

[10] W.-W. Hsieh, S.-L. Chen, I.-S. Lin, and T. Hwang, "A physical-location-aware X-filling method for IR-drop reduction in at-speed scan test," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 29, no. 2, pp. 289–298, 2010.

[11] X. Wen, S. Kajihara, K. Miyase, T. Suzuki, K. Saluja, L.-T. Wang, K. Abdel-Hafez, and K. Kinoshita, "A new ATPG method for efficient capture power reduction during scan testing," in *VLSI Test Symp.*, 2006, pp. 58–65.

[12] M.-F. Wu, K.-S. Hu, and J.-L. Huang, "LPTest: a flexible low-power test pattern generator," *Journal of Electronic Testing: Theory and Applications*, vol. 25, no. 6, pp. 323–335, 2009.

[13] S. Eggersglüß, K. Miyase, and X. Wen, "SAT-based post-processing for regional capture power reduction in at-speed scan test generation," in *IEEE European Test Symp.*, 2016, pp. 1–6.

[14] K. Chakravadhanula, V. Chickermane, B. Keller, P. Gallagher, and P. Narang, "Capture power reduction using clock gating aware test generation," in *International Test Conf.*, 2009, pp. 1–9.

[15] L.-J. Lee, "Observation-oriented ATPG and scan chain disabling for capture power reduction," *Journal of Electronic Testing: Theory and Applications*, vol. 29, no. 5, pp. 625–634, 2013.

[16] J. P. Hayes and A. D. Friedman, "Test point placement to simplify fault detection," *IEEE Transactions on Computers*, vol. C-23, no. 7, pp. 727–735, 1974.

[17] C. Schotten and H. Meyr, "Test point insertion for an area efficient BIST," in *International Test Conf.*, 1995, pp. 515–523.

[18] N. Tamarapalli and J. Rajski, "Constructive multi-phase test point insertion for scan-based BIST," in *International Test Conf.*, 1996, pp. 649–658.

[19] N. A. Touba and E. J. McCluskey, "Test point insertion based on path tracing," in *VLSI Test Symp.*, 1996, pp. 2–8.

[20] J.-S. Yang, N. A. Touba, and B. Nadeau-Dostie, "Test point insertion with control points driven by existing functional flip-flops," *IEEE Transactions on Computers*, vol. 61, no. 10, pp. 1473–1483, 2012.

[21] M. J. Geuzebroek, J. T. van der Linden, and A. J. van de Goor, "Test point insertion that facilitates ATPG in reducing test time and data volume," in *International Test Conf.*, 2002, pp. 138–147.

[22] M. Yoshimura, T. Hosokawa, and M. Ohta, "A test point insertion method to reduce the number of test patterns," in *IEEE Asian Test Symp.*, 2002, pp. 298–304.

[23] S. Remersaro, J. Rajski, T. Rinderknecht, S. M. Reddy, and I. Pomeranz, "ATPG heuristics dependant observation point insertion for enhanced compaction and data volume reduction," in *IEEE International Symp. on Defect and Fault Tolerance in VLSI Systems*, 2008, pp. 385–393.

[24] C. Acero, D. Feltham, F. Hapke, E. Moghaddam, N. Mukherjee, V. Neerkundar, M. Patyra, J. Rajski, J. Tyszer, and J. Zawada, "Embedded deterministic test points for compact cell-aware tests," in *International Test Conf.*, 2015, pp. 1–8.

[25] S. Udar and D. Kagaris, "Minimizing observation points for fault location," in *IEEE International Symp. on Defect and Fault Tolerance in VLSI Systems*, 2009, pp. 263–267.

[26] S. Eggersglüß, K. Schmitz, R. Krenz-Bååth, and R. Drechsler, "On optimization-based ATPG and its application for highly compacted test sets," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 2016, available (early access).

[27] M. Sauer, S. Reimer, T. Schubert, I. Polian, and B. Becker, "Efficient SAT-based dynamic compaction and relaxation for longest sensitizable paths," in *Design, Automation and Test in Europe*, 2013, pp. 448–453.

[28] M. Sauer, B. Becker, and I. Polian, "PHAETON: A SAT-based framework for timing-aware path sensitization," *IEEE Transactions on Computers*, vol. 65, no. 6, pp. 1869–1881, 2016.