Published in IET Computers & Digital Techniques
Received on 14th June 2007
Revised on 25th September 2007
doi: 10.1049/iet-cdt:20070088

ISSN 1751-8601

Simultaneous capture and shift power reduction test pattern generator for scan testing

H.-T. Lin J.C.-M. Li

Laboratory of Dependable Systems (LaDS), Graduate Institute of Electronics Engineering, National Taiwan University, Taiwan, Republic of China
E-mail: cmli@cc.ee.ntu.edu.tw

Abstract: An automatic test pattern generation (ATPG) technique, which simultaneously reduces capture and shift power during scan testing, is presented. This ATPG performs power reduction during dynamic test compaction so the test length overhead is very small. This low-power test generator implements several novel techniques, such as parity backtrace, confined propagation, dynamic controllability and post-fill test regeneration. The experimental data on ISCAS benchmark circuits show that the peak capture power and the peak shift power are reduced by 31% and 26%, respectively.

1 Introduction

It has been widely known that circuit power dissipation in test mode is much higher than that in function mode [1]. Since automatic test pattern generation (ATPG) software has made great progress in performance, it is now worthwhile to trade-off ATPG performance for low power. The most important reasons for low-power ATPG are given as follows. First of all, low-power testing avoids the risk of damaging the circuits under test (CUT). In addition, low-power testing enables parallel testing of multiple cores in the system on chip (SoC). Last, low-power testing prevents power integrity problems in test mode [2]. High current in test mode results in excessive V_{dd} drop or ground bounce, which may cause the CUT to malfunction.

Both average and peak power reduction are necessary for low-power ATPG because the former causes overheating problem and the latter causes power integrity problem. Average power is the time average of power consumption during test application. Instantaneous power is the power consumption at a specific moment. Peak power refers to the maximum instantaneous power during the whole test application. In scan testing, power consumption can also be classified into two categories: capture power and shift power. The former occurs when the flip-flops capture the circuit responses at the active edge of the system clock; the latter occurs when the scan flip-flops are shifting in scan mode. Shift power includes both shift-in power and shift-out power. Shift-in power is consumed when loading the scan flip-flops; shift-out power is consumed when unloading the captured responses in scan flip-flops. They are inseparable because shifting in a test pattern is accompanied by shifting out the test responses of the previous pattern. Capture power reduction is essential to enable at-speed testing in nanometer technology [3–5]. Shift power reduction is also critical to prevent scan chain failure and yield loss, which is receiving serious attention recently in the industry [6]. So far, there are few low-power ATPG available for both capture and shift power reduction [7].

Past research in low-power test generation can be summarised as follows. Wang and Gupta [8] add novel testability measure to traditional test generation algorithm and effectively reduce the average power. Genetic algorithm was proposed to remove redundant test sequences without fault coverage degradation [9], which is effective for functional (fully sequential)

testing. Reordering test patterns greatly reduces the shift power without increasing the test length [10, 11]. However, test pattern reordering alone cannot reduce the capture power. Static test compaction plus reordering effectively reduces both the shift power and the capture power [12, 13]. Bit-stripping is required if test cubes are already generated by other ATPG tools [14]. X-filling of test cubes algorithm successfully reduces the peak capture power [3, 4, 15, 16]. Besides software techniques, one of the first hardware techniques to divide scan chain for both shift and capture power reduction is proposed by Rosinger *et al.* [17].

This paper proposes a low-power test generation technique for simultaneous CASPR. This technique combines dynamic test compaction and low-power X-filling so the test length overhead is small and no bit-stripping is needed. CASPR is based on the FAN algorithm with the following novel improvements. Parity backtrace makes smart decision when selecting secondary fault in dynamic test compaction. Confined propagation guides fault effect propagation to objective outputs to reduce power. Dynamic controllability adjusts the SCOAP testability measure every time a fault is compressed into a test cube such that subsequent decisions are made in favour of power. Post-fill test regeneration removes high-power patterns and regenerates new patterns to ensure the fault coverage.

Although the power reduction of ATPG solution is not as good as that of hardware solutions, the advantage of the former is to avoid any change in design. In the SoC era, many cores are reused or purchased so it is nearly impossible to redo the DFT for power reduction. Low-power ATPG provides a good alternative to reduce the SoC test power without changing the design. Low-power ATPG hence provides a very important solution to reduce the cost and shorten the time to market.

This paper is organised as follows. Section 2 introduces the proposed novel techniques of low power ATPG. Section 3 describes the flow of each step. Section 4 shows the experimental data on ISCAS'89 benchmark circuits. Section 5 discusses the issues related to the presented idea, and finally, Section 6 concludes this paper.

2 Proposed novel techniques

2.1 Parity backtrace

Traditional dynamic test compaction randomly selects a target fault on the backtrace path of an output with unknown (X) [18]. To reduce shift-out power and capture power, it is important to select a 'right target fault' that causes as little power as possible. CASPR selects target faults that achieve the low-power objectives. An objective $<O, v>$ is composed of an objective output (O), to which the fault effect should propagate, and the corresponding objective value (v) of the objective output. For the example in Fig. 1, scan flip-flops are labelled as pseudo-primary outputs (PPO_1 to PPO_3), in the order from scan input to scan output. Since PPO_1 and PPO_3 are already logic ones, PPO_2 is also desired to be the one to reduce the shift-out power. On the basis of this reason, CASPR set the objective as $<PPO_2, 1>$. For another example, suppose that PPO_1 to PPO_3 are '1X0', the objective would be $<PPO_2, 1>$. Because the direction of scan is from PPO_1 to PPO_3, it is better to have PPO = '110' than '100'. Although both PPOs have only one transition, the former has shorter distance of transition travelling in the scan chain. This choice helps to reduce the average shift power. Although Fig. 1 demonstrates an objective for shift-out power reduction, the same technique can be applied to capture power reduction.

In the pre-process stage, CASPR performs a depth first search starting from every PPO all the way to primary inputs (PIs) or pseudo-PIs (PPI). The backtrace is performed on the basis of fan-out free regions (FFRs). The inversion parity from a particular fault to an output ($\text{Parity}_{\text{fault-output}}$) is equal to the inversion parity from the fault to its FFR stem ($\text{Parity}_{\text{fault-stem}}$) XORed with the inversion parity from the FFR stem to the output ($\text{Parity}_{\text{stem-output}}$). That is

$$\text{Parity}_{\text{fault-output}} = \text{Parity}_{\text{fault-stem}} \oplus \text{Parity}_{\text{stem-output}} \quad (1)$$

During the backtrace, the first undetected fault that satisfies (2) is selected as the primary target fault (PTF).

$$\text{objective value} \oplus \text{Parity}_{\text{fault-output}} = \overline{\text{stuck-at value of target fault}} \quad (2)$$

Equation (2) simply says that the fault-free value of the target fault, when propagated to the objective output,

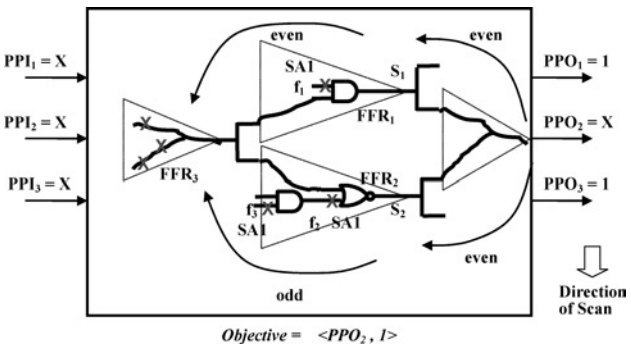

Figure 1 *Parity backtrace*

doi: 10.1049/iet-cdt:20070088

should be the same as the objective value. Fig. 1 shows an example of the parity backtrace for the objective $<PPO_2, 1>$. The inversion parity from PPO_2 to FFR_1 stem (s_1) is even and the inversion parity from s_1 to fault f_1 is even. The inversion parity from f_1 to PPO_2 is therefore even, so f_1 (stuck-at one) is not chosen as the PTF. In contrast, the inversion parity from f_2 to PPO_2 is odd, so f_2 (stuck-at one) can be chosen as the PTF.

2.2 Confined fault propagation

CASPR tries to confine the fault effect propagation within the fan-in cone of the objective output. This is implemented by marking all the gates in the fan-in cone of the objective output. Whenever CASPR searches for a propagation path, it chooses only marked gates in the D-frontier. Fig. 2 illustrates such an example of confined fault propagation. Suppose that PPO_2 is the objective output and gates in the fan-in cone of PPO_2 are marked in grey. The fault effect (denoted as 'D' in the figure) has two potential propagation paths (or X-paths). Although the upper X-path (to PPO_1) is closer to the output, CASPR chooses the lower X-path (to PPO_2) because of the confined fault propagation.

If the test generation for the PTF is successful, a secondary target fault (STF) is selected. A STF is an undetected fault that shares the same objective as the PTF. The STF must meet the condition of (2) so STF can share the same propagation path as PTF. If the current FFR has no undetected fault, CASPR continues to search for STF in the fan-in cone of the current FFR. Continued from the same example in Fig. 1, fault f_3 can be chosen as the STF of the PTF f_2. After detecting f_3, if there is no more fault available in FFR_2, we either change a new objective or continue to search FFR_3 for the next STF.

2.3 Dynamic controllability

CASPR proposes to adjust the controllability of PPI to reduce capture transition of flip-flops. A capture transition is a change in flip-flop content before and after the system clock. CASPR dynamically updates the SCOAP controllability [19] of PPI every time a fault is compacted into a test cube. Table 1 illustrates how the dynamic controllability is computed. Following the original SCOAP definition, CC^0 and CC^1 represent the combinational zero and one controllability, respectively. A lower CC value indicates better controllability. In the beginning, CC^1 and CC^0 of every PPI are initialised to a low value. After a test generation, if a PPO changes from X to 1, CASPR dynamically decreases CC^1 of the corresponding PPI to a lower value and increases CC^0 to a higher value. The new controllability gives higher priority to the assignment PPI = 1 than the assignment PPI = 0, and hence reduces the probability of capture transition in the corresponding flip-flop. In our implementation, the high and low CC values are assigned to be the number of PPI divided by 4 and 20, respectively. Please note that CC^0 and CC^1 of PI are always zero so the backtrace procedure prefers PI to PPI. Controllability of PI remains unchanged because they do not affect the capture power.

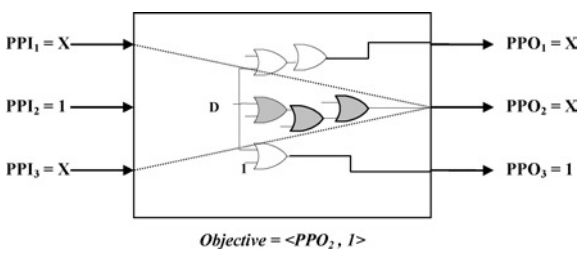

Figure 2 *Confined fault propagation*

Table 1 Dynamic testability

Condition	Before test generation		After test generation	
	CC^0	CC^1	CC^0	CC^1
PPI = X; PPO = $X \rightarrow X$	low	low	low	low
PPI = X; PPO = $X \rightarrow 1$	low	low	high	0
PPI = X; PPO = $X \rightarrow 0$	low	low	0	high
PI	0	0	0	0

2.4 Post-fill test regeneration

Tradition ATPG simply performs minimum transition X-filing to reduce shift power. However, minimum transition filling sometimes increases the capture power. CASPR performs a power simulation after the whole test set generation and then removes those test patterns of the highest instantaneous power. The test set is fault-simulated again and new test patterns are regenerated for those undetected faults due to the pattern removal. The iteration ends until the test length overhead or the iteration count exceeds user defined limits. See Section 3.2 for a step-by-step description of the post-fill test regeneration.

3 ATPG flow

Fig. 3 shows the complete CASPR flow.

(A) The circuit is pre-processed to obtain important circuit structure information, such as headlines and FFRs. For fast parity backtrace, faults in every FFR are stored in a linked list so that the undetected faults in an FFR can be quickly searched. In addition, untestable faults are proven by a Boolean satisfiability

doi: 10.1049/iet-cdt:20070088

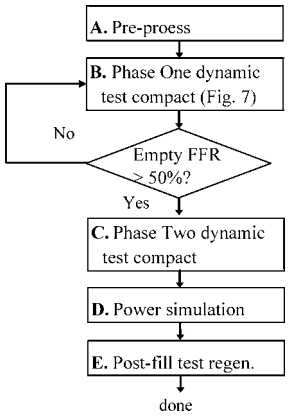

Figure 3 *Overall CASPR flow*

engine and removed from the fault list. This is a crucial step to prevent the subsequent ATPG from wasting time on untestable faults.

(B) and (C) To balance the CPU time and power, CASPR uses two-phase dynamic test compaction. Phase 1 (B) performs dynamic test compaction with power constraints; phase 2 (C) is a regular dynamic test compaction without power constraint. In the beginning of test generation, there are many undetected faults so phase 1 controls the power very carefully. After many faults are detected and dropped, the remaining faults are hard to compact so it is a waste of time to impose complicated power constraints. The details of phase 1 dynamic test compaction are described in Section 3.1.

Phase 2 test compaction is turned on when the number of empty cones, PPO fan-in cones that have no undetected fault inside, exceeds a threshold. According to the experimental results on ISCAS benchmark circuits, CASPR starts phase 2 test compaction when 50% of PPO cones are empty. Adjusting the threshold does not have significant influence on the power reduction.

(D) A logic simulation is performed to count the number of transitions in the shift mode, as well as in the capture mode. Shift power simulation requires a cycle-by-cycle simulation which is very time-consuming. CASPR performs parallel simulation on 32 patterns simultaneously to speed up this process.

(E) The last step is post-fill test regeneration. The peak power patterns are removed and new patterns are generated. The details of test regeneration are described in Section 3.2.

3.1 Dynamic test compaction flow

Fig. 4 shows the flow chart of phase 1 dynamic test compaction. The flow begins with a low density test

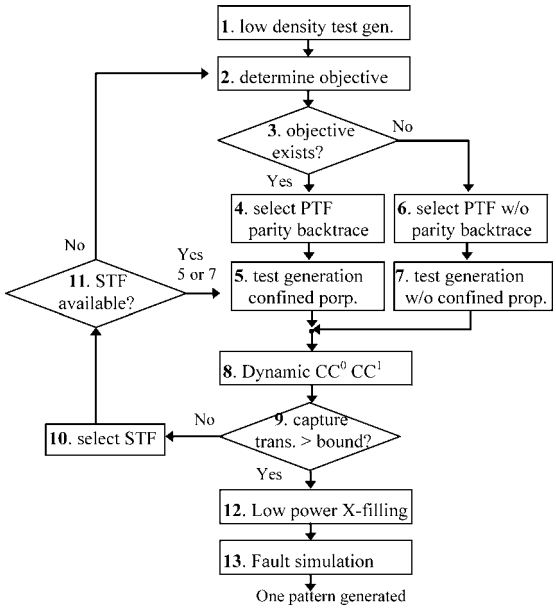

Figure 4 *Phase 1 dynamic test compaction*

generation (step 1), which is a regular dynamic test compaction without any power constraint imposed. CASPR randomly selects target faults on fan-out stems of different FFRs. The low-density test generation keeps compacting faults until the number of specified PPOs exceeds 10% of the total number of PPOs. The low-density test generation provides a starting point for the following dynamic test compaction.

After the low-density test generation, CASPR determines an objective based on the instantaneous capture power or shift-out power of the current test cube (step 2). The former is estimated by the number of capture transitions and the latter by the number of transitions between two neighbour scan flip-flops. When there is a tie, capture power reduction has higher priority over shift power reduction because the shift power can be greatly reduced by the subsequent low power X-filling in step 12. If a low-power objective exists (step 3), a PTF is selected using parity backtrace followed by a test generation with confined fault propagation (steps 4 and 5). In case there is no objective available, a PTF is selected using traditional backtrace without parity, such as [18], followed by a test generation without confinement (steps 6 and 7). In step 8, the controllability (CC^0 and CC^1) are dynamically updated according to the compacted test cube.

After every single compaction, CASPR monitors the capture transition of the current test cube (step 9). Unlike previous techniques, which use don't care bit ratio as the power bound [4], CASPR uses capture transition as the power bound because we find it a better power indicator than the X ratio. If the current test cube does not exceed the maximum of 40% capture transition, CASPR selects an STF and

doi: 10.1049/iet-cdt:20070088

continues the dynamic test compaction process (steps 10 and 11). If there is no more STF available, CASPR returns step 2 to choose another objective. The above dynamic test compaction (steps 2–11) is repeated until the capture transition exceeds the 40% bound. When the capture transition exceeds the power bound, the corresponding target fault is delayed for a later test generation. Please note that the 40% capture transition bound is chosen based on our experimental data. Because the high power test patterns will be removed later, the overall results are not very sensitive to this 40% bound. For example, experiments on large ISCAS benchmark circuits show that changing the bound to either 30% or 50% would not affect the power by $>1\%$, which is not very significant. Also, the difference of test length is below 5.

Once the test cube is compacted, CASPR starts low power X-filling (step 12). In low-power X-filling, CASPR first tries to reduce the capture power by filling the flip-flops that have $X \rightarrow 1$ or $X \rightarrow 0$ transition during capture. The circuit is simulated and X-fill in the same way again. After two such iterations, CASPR starts to perform the minimum transition X-filling, where the PPI are filled with the same value as its downstream neighbour. Finally, a fault simulation is performed to drop all the detected faults (step 13) and a filled test pattern is generated.

3.2 Post-fill test regeneration flow

Fig. 5 shows the post-fill test regeneration flow, which is divided into capture power reduction and shift power reduction. After the power simulation, the pattern which causes the highest capture power is removed. The whole test set is fault simulated (in reverse pattern order with fault dropping) to identify two things: (1) the redundant patterns that detect no fault and (2) the undetected faults due to the pattern removal. The redundant patterns are removed. Also, a new pattern is generated for undetected faults. The new test set is power simulated again. If the peak capture power does not decrease, the new pattern is abandoned. The undetected faults are divided into several groups and more new patterns are generated again. During the capture power reduction stage, X-filling is performed in favour of capture power reduction. After the capture power reduction, the same procedure is repeated for shift power reduction. During the shift power reduction stage, CASPR makes sure that the capture power of regenerate patterns does not exceed the value that we previously achieved in the capture power reduction stage.

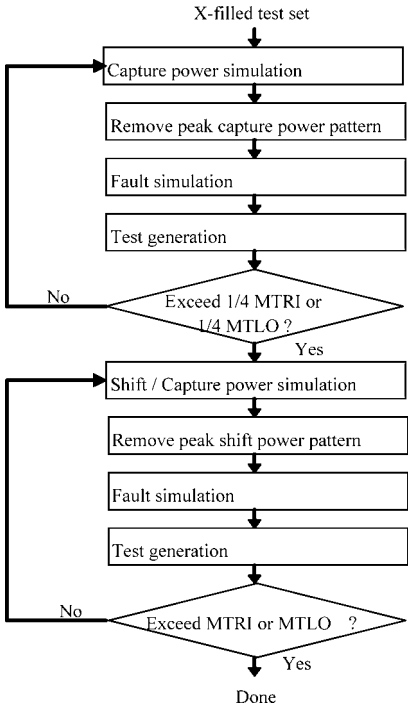

Figure 5 *Post-fill Test regeneration flow*

Each of the two procedure ends when either the number of iterations or the number of additional patterns is larger than two user-defined limits: maximum test regeneration iteration (MTRI, the largest number of iteration allowed) and maximum test length overhead (MTLO, the largest percentage of additional patterns allowed). MTRI controls the CPU time and MTLO controls the test length of post-fill test regeneration. CASPR reserves one quarter of MTRI/MTLO for capture power reduction and the rest three quarters for shift power reduction. For example, if the user-defined MTLO is 8%, then 2% of the budget is reserved for capture power reduction and the other 6% for shift power reduction.

Note that the selection of MTLO can be adjusted based on the users' constraint on test data size. The selection of MTRI can be adjusted based on users' tolerance of runtime. There is, of course, a tradeoff between runtime, test length and power reduction. For full scan circuits, however, the selection of MTLO and MTRI does not depend on the sequential depth, because only combinational ATPG is used.

4 Experimental results

To validate the effectiveness of CASPR, experiments are performed on ISCAS benchmark circuits. The power dissipation in our experiments is estimated by the weighted transition, which is the summation of transitions on each net time its associated fan-out load. That is

$$\text{weighted transition} = \sum_{\text{allnet}i} (C_{\text{load}\,i} \cdot T_i) \qquad (3)$$

where T_i is the transition on net i and $C_{\text{load}\,i}$ the total load capacitance connected to net i. In this implementation, $C_{\text{load}\,i}$ is approximated by the number of fan-outs on net

doi: 10.1049/iet-cdt:20070088

Table 2 Comparison of regular ATPG and CASPR

CUT	Regular ATPG					CASPR (MTLO = 15%; MTRI = 80)				
	test length	fault coverage, %	peak power		average power	test length	fault coverage, %	peak power		average power
			capture	shift				capture	shift	
s5378	118	99.13	1737	1767	1277	138	99.13	999	1 473	986
s9234	162	93.48	3059	3551	2533	185	93.48	1881	2856	1801
s13207	243	98.46	4045	5126	4052	249	98.46	2598	4125	2294
s15850	117	96.68	3397	5677	4030	133	96.68	2559	4682	2279
s35932	24	89.81	11 376	11 215	6867	27	89.81	9578	7010	2891
s38417	111	99.47	12 687	14 273	10 174	125	99.47	8770	9680	4179
s38584	138	95.85	11 753	11 906	10 097	158	95.85	6762	9672	4870
Average	130	96.13	6865	7645	5576	145 (+11.2%)	96.13 (−0.0%)	4735 (−31.0%)	5643 (−26.2%)	2757 (−50.6%)

i. Note that, although weighted switching is not a perfect metric to measure the absolute power, it can be used to compare the relative power of our technique with original ATPG. It serves the purpose as a relative measure so it was used in many previous publications [20, 21]. We also adopt this measure so that we can make a fair comparison with other previous techniques.

Table 2 compares the test patterns generated by regular ATPG and CASPR. The regular ATPG was implemented based on the FAN algorithm and a traditional dynamic test compaction [18]. CASPR patterns are generated with the limit MTLO = 15% and MTRI = 80 in the post-fill test regeneration. The fault coverage of CASPR is exactly the same as the regular ATPG. On average, CASPR reduces the peak capture power by 31% and the peak shift power by 26% compared with the regular ATPG. CASPR test length is only 11.2% longer than that of regular ATPG. The actual test length overhead is shorter than MTLO (15%) because some circuits hit the MTRI limit before the test length overhead reaches 15%. Note that shift peak power is always higher than the capture peak power except s35932. This could be because the s35932 circuit has extremely large fan-out and many faults are detected by a few patterns. Table 3 again

Table 3 Comparison of regular ATPG and CASPR (DC only)

CUT	Regular ATPG					CASPR (DC Only)				
	test length	fault coverage, %	peak power		average power	test length	fault coverage, %	peak power		average power
			capture	shift				capture	shift	
s5378	118	99.13	1737	1767	1277	122	99.13	1317	1765	1082
s9234	162	93.48	3059	3551	2533	165	93.48	2402	3170	1917
s13207	243	98.46	4045	5126	4052	247	98.46	3184	4548	2317
s15850	117	96.68	3397	5677	4030	118	96.68	2867	5290	2467
s35932	24	89.81	11 376	11 215	6867	24	89.81	10 386	7010	3393
s38417	111	99.47	12 687	14 273	10 174	108	99.47	8791	12 107	4585
s38584	138	95.85	11 753	11 906	10 097	144	95.85	7888	10 884	5148
Average	130	96.13	6865	7645	5576	133 (+1.6%)	96.13 (−0.0%)	5262 (−23.3%)	6396 (−16.3%)	2987 (−46.4%)

doi: 10.1049/iet-cdt:20070088

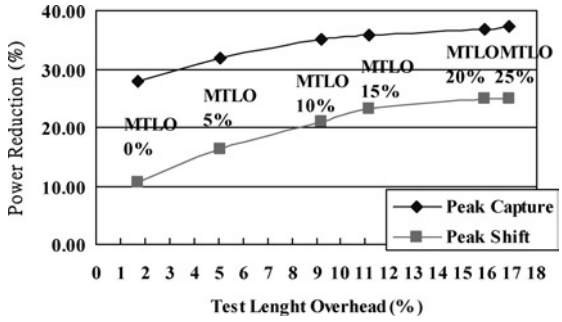

Figure 6 *Power reduction in post-fill reduction*

shows the comparison of regular ATPG with CASPR without test regeneration. This time, the peak capture power and peak shift power reduction are 23% and 16%, respectively. The results show that test regeneration contributes about 8% and 10% improvement in peak CASPR, respectively.

The following figure shows the trend of power reduction against test length overhead (with respect to regular ATPG). The upper curve shows the capture power reduction and the lower curve shows the shift power reduction (with respect to regular ATPG). Each curve represents the average of six ISCAS benchmark circuits in Tables 2 and 3. Every dot represents the MTLO of 0%, 5%, 10%, 15%, 20% and 25% from left to right (Fig. 6). It is observed that the first three MTLOs are very effective in terms of power reduction. After MTLO becomes larger than 15%, however, the marginal power reduction becomes less significant. On the basis of the results, it is recommended that setting MTLO to 15% is good enough for ISCAS benchmark circuits.

Table 4 compares the CPU time and memory usage of different ATPG techniques. If we perform only dynamic compaction, it requires 44% more CPU time than the regular ATPG. If we perform both dynamic compaction and post-fill test regeneration (MTLO = 15%, MTRI = 80), it requires 133% more CPU time than the regular ATPG. Most of the CPU time overhead in test regeneration is attributed to the shift power simulation, which is a cycle-by-cycle simulation.

From this table, it can be seen that the DC runtime does not grow out of control for large circuits. For example, the original ATPG on s38417 takes 1200 s and our overhead is 30% (DC only). Among the four proposed techniques, parity backtrace is linear time and is performed only once in the pre-processing stage. The confined fault propagation and dynamic controllability are also linear time. The test regeneration is probably the most time-consuming process so it is limited by user-defined MTRI parameter.

Dynamic compaction and test regeneration requires only 30% and 49% more memory than the regular ATPG, respectively. Previous research used a fault simulation without fault dropping to generate a complete fault dictionary for test pattern regeneration. To make a comparison, we perform a parallel fault simulator without fault dropping and the memory overhead is listed in the last column. In our implementation, a 4B pointer is assumed to store each entry in the complete fault dictionary. It is observed that a complete fault dictionary requires more than 1.5 times as much memory as the regular ATPG.

Fig. 7*a* shows the peak power of every individual test pattern of s38417. The horizontal axis is the pattern index and the vertical axis is the peak power (the maximum of capture and shift power). The triangle dots represent the regular ATPG patterns and the diamond dots represent the CASPR patterns (after dynamic compaction, before post-fill test

Table 4 CPU time and memory usage

CUT	Regular ATPG		CASPR DC only		CASPR DC+ regen.		complete fault dic.
	CPU time, s	memory, MB	CPU time overhead, s	memory overhead, MB	CPU time overhead, s	memory overhead, MB	memory overhead, MB
s5378	3.2	1.23	0.3	0.42	12.1	1.09	2.11
s9234	73.7	2.15	8.3	0.55	62.6	0.94	2.90
s13207	61.2	3.41	11.1	0.35	431.6	2.03	8.34
s15850	189.1	4.44	81.2	3.37	192.7	3.5	4.86
s35932	562.2	7.70	104.2	2.49	168.5	3.95	4.02
s38417	1192.8	9.61	360.2	1.40	581.2	1.93	12.51
s38584	787.2	9.02	705.8	2.69	2368.8	4.8	18.79
Average	409.9	5.37	181.6 (+44%)	1.61 (+30%)	545.4 (+133%)	2.61 (+49%)	7.64 (143%)

doi: 10.1049/iet-cdt:20070088

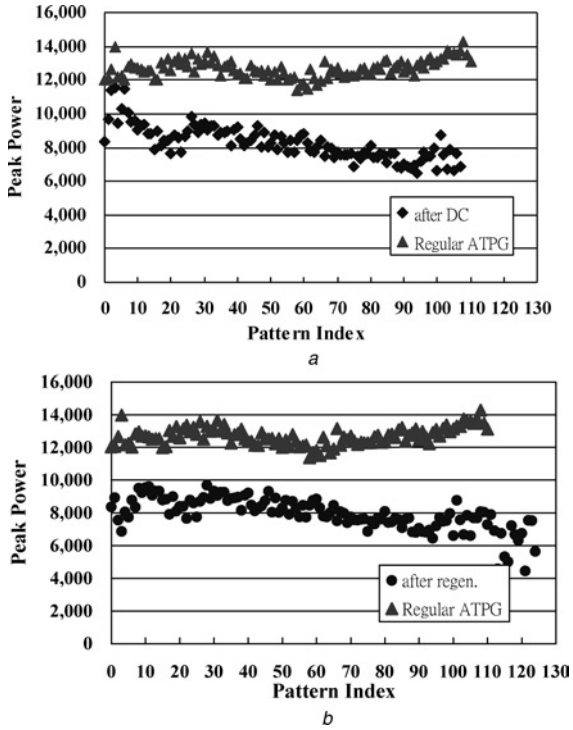

Figure 7 *Peak power of s38417*

a After DC, before test regeneration
b After test regeneration

regeneration). It is observed that the peak power of regular ATPG patterns is much higher than that of the CASPR patterns. This figure shows clearly that dynamic compaction alone successfully reduces the peak power. However, the first few CASPR patterns have extraordinarily high power because too many faults are compacted in the first few patterns. Fig. 7*b* shows the same comparison except that the round dots now represent the CASPR patterns after post-fill test regeneration. The high-power CASPR patterns are removed at the cost of a few new patterns. The overall peak power of CASPR is now much lower than that of the regular ATPG patterns. It should be noted that modifying a regular ATPG test set to reduce power, as suggested by previous techniques, can be very time-consuming since almost every test pattern from regular ATPG needs modification. These results justify the need of a low-power ATPG from scratch.

The following table compares the test length of CASPR and two pure software low-power ATPG. The test length overhead of [20] is compared with their original test length because they did not perform test compaction. The test length overheads of CASPR and [3] are compared against our regular ATPG. The power reduction in both [3] and [20] is obtained from their original papers. Note that Wang and Gupta [20] do not perform peak power reduction and [3] does not perform shift power reduction (Table 5).

5 Discussion and future work

5.1 Transition fault ATPG

Although the presented idea is demonstrated on single stuck-at fault model, the same technique can be

Table 5 Comparison with previous techniques

CUT	CASPR				[20]			[3]		
	test length[a]	FC loss,	average power reduction, %	peak capture power reduction, %	test length	FC loss, %	average power reduction[b], %	Test length	FC loss, %	peak capture power reduction[b], %
s5378	138	0.0	−22.8	−42.5	738	0.0	−73.0	115	0.0	−13.8
s9234	185	0.0	−28.9	−38.5	854	0.0	−70.0	145	0.0	−6.1
s13207	249	0.0	−43.4	−35.8	1155	0.3	−77.0	262	0.0	−19.5
s15850	133	0.0	−43.4	−24.7	1016	0.5	−83.0	119	0.0	−23.5
s35932	27	0.0	−57.9	−15.8	66	0.3	−92.0	38	0.0	−29.8
s38417	125	0.0	−58.9	−30.9	2406	0.0	−78.0	118	0.0	−21.1
s38584	158	0.0	−51.7	−42.7	4977	0.0	−75.0	135	0.0	−37.9
Average	145 (+11.2%)	0.0	−50.6	−31.0	519 (+298%)[a]	0.2	−78.2	133 (+2.1%)[a]	0.0	−25.7

[a]Compared with our regular ATPG; [b]Obtained from original papers

doi: 10.1049/iet-cdt:20070088

applied to delay fault test generation, such as transition fault ATPG. For launch-on-capture pattern generation, we can expand the circuit into two time frames: before and after capture clock. The parity backtrace technique can be applied in the second time frame to select low-power secondary faults. The confined propagation technique can be applied across two time frames to guide fault effect propagation to objective outputs. Dynamic controllability can still be used in both time frames to adjust the SCOAP testability measure. Post-fill test regeneration, of course, can still be used to regenerate those patterns that have excessive power. Overall speaking, the proposed techniques are still valid for low-power delay fault test generation.

To reduce the power in both time frames simultaneously, the transition in two time frames has to be counted separately. In the test compaction stage, if any time frame exceeds the 40% capture transition bound, dynamic compaction ends. In the test regeneration stage, if any time frame has the peak power, the corresponding test must be removed and regenerated.

5.2 Working with test compression

Test data volume keeps increasing and test compression becomes a very important feature of commercial tools. These tools first generate test cubes and then insert extra hardware to perform on-chip decompression of cubes [21]. The proposed dynamic compaction technique can be applied in the test cube generation stage such that the capture power is reduced. The test regeneration technique can be applied after test cubes compression such that shift power can be reduced. There is, indeed, tradeoff between test power and compression ratio, which can be a topic for future research.

6 Summary

This paper presents a low-power ATPG that simultaneously reduces the shift power and capture power. Four novel techniques are proposed to minimise the test length overhead. The experimental results on ISCAS'89 benchmark circuits show that a reduction of 31% and 26% of peak capture and peak shift power is obtained at a cost of only 11% test length overhead.

7 References

[1] GIRARD P: 'Survey of low-power testing of VLSI circuits', *IEEE Des. Test Comput.*, 2002, 19, (3) pp. 80–90

[2] BUTLER KM, JAIN JSA, JACK LEWIS TF, *ET AL.*: 'Minimizing power consumption in scan testing: pattern generation and DFT techniques'. Proc. IEEE Int. Test Conf., 2004, pp. 77–84

[3] WEN X, YAMASHITA Y, KAJIHARA S, *ET AL.*: 'On low-capture-power test generation for scan testing'. Proc. IEEE 23th VLSI Test Symp., 2005

[4] WEN X, YAMASHITA Y, MORISHIMA S, *ET AL.*: 'Low-capture-power test generation for scan-based at-speeding testing'. Proc. IEEE Int. Test Conf., 2005

[5] WEN X, KAJIHARA S, MIYASE K, *ET AL.*: 'A new ATPG method for efficient capture power reduction during scan testing'. Proc. IEEE 24th VLSI Test Symp., 2006

[6] HUANG Y, CHENG W-T, REDDY SM, *ET AL.*: 'Statistical diagnosis for intermittent scan chain hold-time fault'. Proc. IEEE Int. Test Conf., 2003, pp. 319–327

[7] REMERSARO S, LIN X, ZHANG Z, *ET AL.*: 'Low shift and capture power scan tests'. Proc. Int. Conf. on VLSI Design, 2007, pp. 793–798

[8] WANG S, GUPTA SK: 'ATPG for heat dissipation minimization during test application', *IEEE Trans. Comput.*, 1998, **47**, (2), pp. 256–262

[9] CORNO F, REBAUDENGO M, REORDA MS, SQUILLERO G: 'Low power BIST via non-linear hybrid cellular automata'. IEEE Proc. VLSI Test Symp., 30 April–4 May 2000, pp. 29–34

[10] DABHOLKAR RV, CHAKRAVARTY S, POMERANZ I, *ET AL.*: 'Techniques for reducing power dissipation during test application in full scan circuits', *IEEE Trans. Comput.-Aided Des.*, 1998, **17**, (12), pp. 1325–1333

[11] GIRARD P, LANDRAULT C, PRAVOSSOUDOVITCH S, *ET AL.*: 'Reducing power consumption during test application by test vector ordering'. Proc. Int. Symp. Circuits and Systems (ISCAS 98), Part II, 1998, pp. 296–299

[12] SANKARALINGAM R, ORUGANTI RR, TOUBA NA: 'Static compaction techniques to control scan vector power dissipation'. Proc. IEEE 18th VLSI Test Symp., 2000, pp. 319–324

[13] SANKARALINGAM R, TOUBA NA: 'Controlling peak power during scan testing'. Proc. IEEE 20th VLSI Test Symp., 2002, pp. 319–324

[14] KAJIHARA S, ISHIDA K, MIYASE K: 'Test vector modification for power reduction during scan testing'. Proc. IEEE 20th VLSI Test Symp., 2002

[15] LI W, REDDY SM, POMERANZ I: 'On reducing peak current and power during test'. Proc. IEEE Symp. VLSI, 2005, pp. 156–161

[16] REMERSARO S, LIN X, ZHANG Z, *ET AL*.: 'Preferred fill: a scalable method to reduce capture power for scan based designs'. Proc. IEEE Int. Test Conf., 2006

[17] ROSINGER PM, AL-HASHIMI BM, *ET AL*.: 'Scan architecture with mutually exclusive scan segment activation for shift and capture power reduction', *IEEE Trans. Comput.-Aided Des.*, 2004, **23**, (7), pp. 1142–1153

[18] GOEL P, ROSALES BC: 'Dynamic test compaction with fault selection using "sensitizable" path tracing', *IBM Tech. Disclosure Bull.*, 1980, **23**, (5), pp. 1954–1958

[19] GOLDSTEIN LH, THIGPEN EL: 'SCOAP: Sandia controllability/observability analysis program'. Proc. Des. Automation Conf., 1980, pp. 190–196

[20] WANG S, GUPTA SK: 'An automatic test patten generator for minimizing switching activity during scan testing activity', *IEEE Trans. Comput.-Aided Des.*, 2002, **21**, (8), pp. 954–968

[21] WANG L-T, WU C-W, WEN X: 'VLSI test principles and architectures: design for testability' (Morgan Kaufmann, 2006)

doi: 10.1049/iet-cdt:20070088