

Minimizing Power Consumption in Scan Testing: Pattern Generation and DFT Techniques

Kenneth M. Butler

Jayashree Saxena
Atul Jain

Tony Fryars
Jack Lewis

Graham Hetherington

Texas Instruments Inc.
Dallas, TX 75243

Abstract

It is a well-known phenomenon that test power consumption may exceed that of functional operation. ICs have been observed to fail at specified minimum operating voltages during structured at-speed testing while passing all other forms of test. Methods exist to reduce power without dramatically increasing pattern volume for a given coverage. We present case study information on ATPG- and DFT-based solutions for test power reduction.

1 Introduction

In recent years structural (mostly scan-based) tests have come to be the dominant mechanism for volume IC manufacturing. At first, low-speed scan tests measured against the stuck-at coverage metric were considered adequate. But, in recent technologies, at-speed test has become a requirement in order to maintain sufficient levels of outgoing quality [1]–[8].

Adding at-speed structural tests to the standard suite of production tests presents an array of new problems. What are the design for test (DFT) requirements that must be met in order to be able to generate high quality at-speed scan tests? What are the tester limitations that must be comprehended? If low cost testers are to be used, what additional constraints do they place on the problem? Many of the practical issues associated with structural at-speed tests were addressed by the authors and others in recent publications [7], [9]–[14].

One of these new problems is associated with switching power consumption during test. It is well-known that power consumption during the test mode of operation can exceed that during normal functional or “mission mode” operation [15], [8]. This fact is true even for relatively slow speed scan operation. In the case of structural at-speed testing such as for transition or path delay faults, power consumption is a concern not only from the standpoint of scan shifting

but also for the high speed capture operation. Very high power demands during test can cause unnecessary yield loss as well as exacerbate already troublesome issues associated with IR-drop and crosstalk [8]. Excessive power consumption can also induce unacceptably high stress-related failures.

Excessive switching power consumption during test can be tackled in multiple ways. Test patterns can be generated to reduce switching activity while possibly increasing pattern volume [16]–[18], [8]. Power grid design can be made to comprehend increased test power demands. Logical DFT techniques can be devised to reduce power; these typically involve clock gating or design partitioning. A prior publication gives a longer list of references in this area [15].

In this paper, we will begin by presenting pattern generation approaches to minimize power consumption that would be used if power-related problems were discovered after design completion. We investigate various heuristics for modifying test content to address power concerns. These techniques require no modifications to the basic design nor its test logic. We compare the heuristics in terms of the amount of switching reduction, and also the increase in pattern volume and decrease in coverage for a given fault coverage figure of merit.

We will then present a DFT approach based on partitioning a module-based design to reduce overall power consumption in the design. The advantage of this approach is that it is very simple to implement and does not involve clock gating as in previous approaches [15]. For both power reduction approaches, we provide case study data from their application to actual production devices implemented in a .13 micron ASIC library.

2 Power Consumption During Test

In order to analyze power consumption during test, it is best to break test operation into its two

sub-functions, scan shift and scan capture. During scan shifting, due to the large amount of simultaneous switching, power consumption can be excessive. Power can be kept in check by reducing shift frequency or by modifying the scan architecture in any of a number of different ways, a few examples of which we cite here [15], [19]–[21]. Scan power problems can also be mitigated by modifying the content of the patterns themselves, which will be the subject of this portion of the paper.

During scan capture, especially at high speed, new problems become apparent. IR-drop and crosstalk effects which would normally not be observed at low speed suddenly materialize [8]. Also, due to the high demands on the clock tree during the high speed captures, unintentional wave shaping and cycle stretching may occur on the clock pulses, with the net effect of either slowing the effective clock frequency [22] or suppressing the clock pulse(s) entirely. To solve this latter problem, it is possible to introduce clock gating to minimize the amount of logic switching during the capture cycle. However, doing so complicates timing closure and is thus undesirable.

Just as in the case of scan shift, pattern content can be modified in such a way as to minimize simultaneous switching, and thus power consumption, during scan capture events. Previous work by the authors in investigating IR-drop and crosstalk showed that if most of the scan-in data were made the same, e.g., all 0's or all 1's, but particularly all 0's, then the post-capture data is often similar, thus suggesting that switching activity had been reduced. Power simulations performed in the authors' earlier work corroborated this hypothesis. This procedure has been referred to by others as "pattern scrubbing" [23] and similar ideas have appeared in earlier work [16]–[18]. Depending on the exact procedure used, reductions occur during scan shift as well as scan capture. The difficulty lies in identifying which bits to modify, and doing it while minimizing any other effects.

It is risky, however, to assume that modifying pattern content will be an adequate solution in every case. It could be such that power consumption is so high that pattern modifications will be excessive and thus pattern volume high as well. Or, it could be that there is a great deal of variation in the amount of bits to be modified across the span of patterns in the test set, which will be discussed in the following section. In such cases, it is better to estimate power early in the design cycle, and plan and implement a design for test approach that maintains an acceptable level of switching activity while minimizing overall design impact. We will discuss such an approach later in this paper.

3 Power Reduction by Pattern Modification

The approach presented in this section discusses different options to create patterns that cause reduced switching activity. No additional DFT features are required to implement this solution.

3.1 Don't Care Bits in Deterministic Scan Patterns

One obvious place to look for bits to "unify" in the pattern data is the so-called "don't care bits". When a deterministic test pattern is generated, the automatic test pattern generation (ATPG) algorithm will place '0' and '1' values on scan cells and device pins as required to create test patterns for the targeted faults. These values are sometimes referred to as the "care bits". When dynamic compaction techniques are applied, multiple faults are targeted in a single test generation session, and the number of care bits increases. Scan cells and device pins not assigned as part of deterministic pattern creation are referred to as the "don't care bits".

In conventional ATPG, the don't care bits are filled in randomly, and then the resulting completely-specified pattern is simulated to confirm detection of all targeted faults and to measure the amount of "fortuitous detection" - Faults which were not explicitly targeted during pattern generation but were detected anyway. Clearly, it is desirable to maintain a rate of fortuitous detections that is as high as possible, though the rate naturally drops off as test generation proceeds, the coverage increases, and the remaining faults become increasingly random-pattern-resistant (harder to detect).

It is interesting, though somewhat non-intuitive, to note that the fraction of don't care bits in a given pattern is nearly always a very large fraction of the total available bits [24]. This observation remains true despite the application of state-of-the-art dynamic and static test pattern compaction techniques. These arguments apply to various types of target faults, including for example stuck-at and transition faults. However, it is conjectured that the average rate for don't care bits in deterministic patterns would in general be higher for transition faults than for stuck-at faults due to the increased difficulty of dynamic compaction in transition fault test generation, at least when launch-from-capture (sequential) test generation is used for the latter.

Figures 1 and 2 examine the properties of care bits for one of the modules used in this study. Deterministic ATPG was applied to the design using a

commercial tool. Launch-from-capture patterns were generated, with the further restrictions of a common clock pin for launch and capture, masked primary outputs, and constant-per-pattern primary inputs - The same low-cost-tester-compatible constraints proposed in an earlier publication [7]. However, the tool was instructed not to randomly fill in the don't care values so that they could be counted.

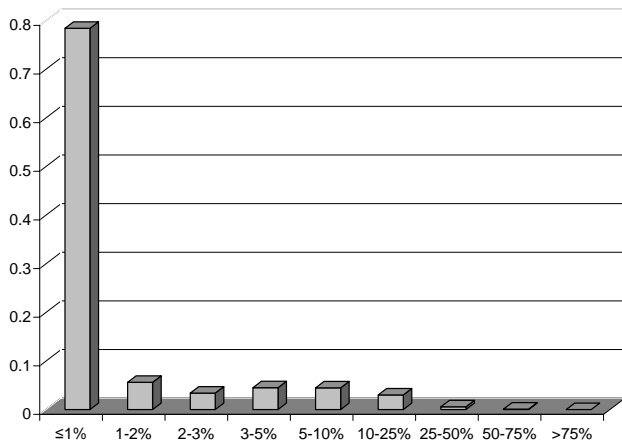


Figure 1: A histogram of the fraction of care bits in deterministic scan patterns for Module M.

Figure 1 shows the proportions of the entire pattern set with 1% or fewer care bits, 1-2%, and so forth. It is clear that the vast majority of patterns have very few care bits. Figure 2 shows good agreement with Wohl et. al that the first few patterns have larger fractions of care bits, but the numbers of care bits rapidly decline as ATPG progresses [24].

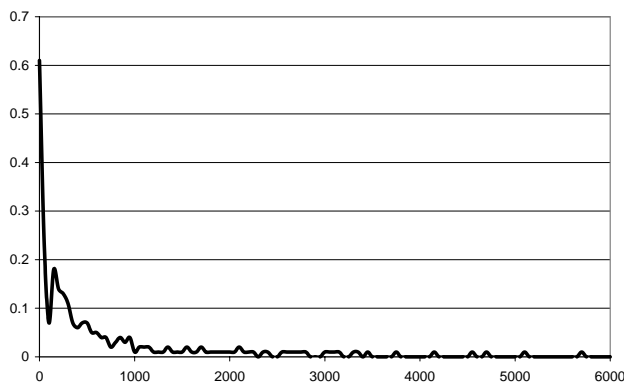


Figure 2: A plot of the fraction of care bits in deterministic scan patterns as a function of pattern number for Module M.

The presence of this significant fraction of don't care bits presents an opportunity that can be exploited for power minimization. We address that fact in the following sections.

3.2 Low Power ATPG Fill Data

Since the rate of don't care values in deterministic patterns is typically high, perhaps one may consider other techniques besides pseudo-random for filling in the don't care bits. And it follows that differing techniques will result in patterns with differing properties, as measured by several important metrics:

- Pattern volume
- Fault coverage
- Power consumption (both shift and capture)

Pattern volume is impacted because poor choices for fill data can limit the number of fortuitous detections. Having fewer fortuitous detections per pattern naturally causes the overall pattern volume to increase. Fault coverage can similarly be impacted - If the pattern volume increases to the point that patterns must be truncated on the tester, then the realizable coverage may be limited. Furthermore, it could be the case that hard-to-detect faults that might have been fortuitously detected with random fill data are aborted when they must be targeted for deterministic pattern generation when non-random filling is used, thus lowering the final coverage figure even when no truncation is done.

Finally, it has been shown that certain non-random fill schemes can minimize power [16]–[18], [23]. As mentioned previously, the authors' prior work showed that constant fill data tended to dramatically reduce switching activity and power, especially when the number of care bits was very small, and that power consumption tended to track well with the number of don't care bits filled randomly [8].

It is of course the goal of this work to minimize power consumption and pattern volume while at the same time maximizing fault coverage. We explore various heuristics for fill data generation which attempt to perform this optimization.

3.3 Experimental Process

This work was an outgrowth of manufacturing situations where power problems were observed during scan-based at-speed testing. For the same devices, stuck-at testing did not manifest any power-related issues. Furthermore, pattern volume is typically far more troublesome for transition tests than it is for stuck-at, even if the stuck-at coverage exceeds the transition coverage by 20% or more. So, all of the results presented hereafter will be given for launch-from-capture transition tests only.

In order to explore various fill heuristics, it was desired to produce results on industrial circuits, and to do so using existing commercial ATPG setups. Unfortunately, at the time this work was done, neither of the two commercial ATPG tools to which we had access had all of the features necessary to complete the analysis. So, the resulting experimental flow was as follows:

1. Generate conventional transition patterns for the target device to provide baseline metrics.
2. Generate transition patterns in ATPG Tool ‘A’ where the tool is not allowed to fill in the don’t care bits (they are left as ‘X’). Write out these unfilled patterns in a format that can be read back into ATPG later.
3. Post-process the patterns using varying heuristics to produce the fill data.
4. Read the patterns into ATPG Tool ‘B’. Perform fault simulation and static compaction to find the final pattern volume and coverage.

Logic simulation was used to measure switching activity, but the run time to simulate all the patterns across multiple fill heuristics was prohibitive. Therefore, a simpler metric was implemented to estimate the overall impact of the various fill techniques. In each case, the post-static-compaction pattern files were read by a script which would compare the scan-in and scan-out data in each scan cell. If the scan-in and scan-out data differed, it was known that capture clocking must have caused the scan cell to switch at least once. If the data were the same, no determination could be made. In this way, a lower bound could be placed on the number of scan cells switching during the capture operation. Comparisons of the lower bounds across differing heuristics allows one to assess the relative merits of each one.

Note that this metric provides no information about switching activity during scan shifting. However, our experiences have been that problems are far more likely to arise during capture than shift due to the higher frequency of the capture cycles, and thus our focus for this work is limited to only the capture cycles. We should note that we have since added more conventional power measurements which will be described in a later section.

3.4 Modified Pattern Results

The data presented below was collected using two modules that are components of a larger ASIC design. Module D is comprised of approximately 600K

2-input NAND equivalent gates, and Module M is approximately 800K gates. The ASIC is implemented in TI’s SR40 (130 nm) ASIC library. Module M has a single clock and eight scan chains of length 2970 scan cells. Module D has three clocks and eight scan chains of length 3271 scan cells. ATPG was performed using all of the constraints for compatibility with low cost test equipment as set forth in a prior publication [7]. Both netlists are the versions that existed prior to physical optimization and clock tree synthesis so that the switching properties of the logic can be examined separately from the clock tree. In the case of Module D, the three clocks were made equivalent so that all captures were occur simultaneously on all portions of the logic. This modification was again done to permit study of the logic’s properties as a unit, and to avoid the difficulties of having to perform multiple observations on each clock domain separately.

In the results below, the “fill adjacent” technique will recur several times. It therefore merits explanation [25]. This random fill heuristic uses information about the care bits to fill in values for the don’t care bits, but this is done on a care-bit-by-care-bit basis, working left to right (chain order of scan-out to scan-in in most pattern formats). When applying the fill adjacent heuristic, the most recent care bit value is used to replace each ‘X’ value. When a new care bit is encountered, it’s value will be used for the adjacent X’s. This technique is very similar to the “MT-Fill” technique of Sankaralingam et. al [18].

So, for example, consider a simple scan pattern that looks like the following: 0XXX0XX1XX0XX. If we apply the fill adjacent heuristic, the resulting filled pattern would be 0000000111000. This is a “complete” fill adjacent application in that all of the X’s were filled in according to the heuristic.

Descriptions of each of the heuristics are given in Table 1. The initial experimental results are given in Tables 2 and 3 below. The leftmost column names the heuristic used to generate the fill data.

The coverages are simply transition fault coverages of the modules. The “Volume” column is the pattern volume required to reach the fault coverage figure in the preceding column. The “Increase” column gives the ratio of pattern volume to the “Baseline A” pattern volume. The “in≠out” column gives the fraction of scan cells for which the scan-in and scan-out data differ. As stated previously, this is a lower bound on the number of scan cells switching during the capture cycles, which is a crude measure of overall switching activity.

Finally, the “%switch” columns are a more direct measure of switching activity. The technique is very similar to that reported in an earlier publication by

| | |
|-------------------------------|---|
| Baseline A | Unmodified ATPG run to determine baseline pattern volume and coverage using ATPG Tool ‘A’. |
| Don’t care ‘X’ | ATPG Tool ‘A’ results in which the tool leaves all don’t care bits as ‘X’. Such patterns would not be used in production testing and are given for information purposes only. |
| Don’t care ‘0’ | ATPG Tool ‘A’ results in which the tool sets all don’t care bits to ‘0’. |
| Don’t care ‘1’ | ATPG Tool ‘A’ results in which the tool sets all don’t care bits to ‘1’. |
| 1000 ‘0’, 1000 ‘1’, repeat | Post-compaction results where ‘X’ values are filled with all 0’s in the first 1000 patterns, then all 1’s in the next 1000, then all 0’s again, and so forth. |
| Rand. 4 chns, all ‘1’ | Post-compaction results where, for each pattern, 4 chains are selected randomly and filled with all 1’s, and the remaining 4 are filled randomly. |
| Rand. 4 chns, Fill adj. comp. | Post-compaction results where, for each pattern, 4 chains are selected randomly and filled using complete fill adjacent, and the remaining 4 are filled randomly. |
| Fill adj. comp. | Post-compaction results where all chains are filled using complete fill adjacent. |

Table 1: A table containing descriptions of heuristics for filling don’t care values in scan tests.

the authors [8]. For the data in the “%switch 1K” column, three patterns were selected from within the first one thousand patterns from the sets of patterns for each of the fill heuristics (the highest switching activity is likely in the earlier patterns based on Wohl et. al [24] and the data in Figure 2). Zero-delay parallel-load scan simulations (with the capture cycle preceded by the simulation of five shift cycles) were performed for the samples in a commercial logic simulation tool, and the simulator was instructed to output files which give the toggle counts for each of the cells in the design for the time period beginning just before the launch clock pulse and continuing until just before the capture clock pulse. This procedure gives the fraction of gates switching during that critical launch-to-capture cycle. The averages of those samples were calculated and reported in the “%switch 1K” column. It should be noted however, that the simulator is not able to count X-to-0, X-to-1, 1-to-X, and 0-to-X events, so the Don’t care ‘X’ number will be lower than it would

actually be on silicon and there are similar, but much smaller errors in the other numbers. The “%switch all” column is similar to the “%switch 1K” column, except that it represents averages of samples of eleven patterns evenly distributed across the length of each respective pattern sets to provide views of the overall switching activity for the pattern sets.

Some time after this study was first completed, one commercial tool became available which permits fill adjacent pattern filling while simultaneously supporting dynamic compaction of patterns. When run against these same designs, we observed similar amounts of switching reduction, but also some reductions in overall pattern volume. These results are included, but it should be noted that we are comparing fill adjacent results arrived at with (superior) dynamic compaction versus the other results, which arise only from static compaction. If the other fill heuristics could be applied while dynamic compaction were used, the resulting pattern volumes would be expected to decrease from what are reported here.

Using these two design modules as examples, the data seems to indicate that the fill adjacent technique does an excellent job of lowering overall switching activity while still maintaining a reasonably low pattern volume. It should be noted that, in all cases, when fill data is non-random, the final coverage number degrades. The drop in coverage is doubly troubling since it means that some true defect coverage is also lost due to the lowered likelihood of “fortuitous detections” (e. g., transition patterns detecting non-explicitly-modeled delay defects). Depending on the amount of coverage loss, this trade-off for coverage versus power may be unattractive.

Finally, this work has focused only on conventional deterministic ATPG. However, there are a variety of ATPG-like solutions which exploit the availability of don’t care bits to permit on-chip hardware compression of otherwise deterministic patterns. It seems likely that the fill heuristics proposed here would not map directly into any of these applications. However, it is conjectured that the on-chip compressors could be modified to generate fill data which simultaneously compresses the data stream and reduces switching power somewhat. The remainder of the of the paper now addresses power reduction via design.

4 Power Reduction via Design Partitioning

In this section, a DFT technique will be presented that reduces switching activity in portions of the circuit not being tested. We will illustrate the technique

| Fill heuristic | Covg. | Volume | Increase | %switch 1K | %switch all | in≠out |
|-------------------------------|--------|--------|----------|------------|-------------|--------|
| Baseline A | 77.26% | 6825 | N/A | 17.34% | 10.37% | 34.05% |
| Don't care 'X' | 76.38% | 10874 | 1.59 | 6.34% | 1.71% | 4.38% |
| Don't care '0' | 72.00% | 10015 | 1.47 | 11.01% | 3.42% | 2.04% |
| Don't care '1' | 74.50% | 9228 | 1.35 | 17.04% | 9.58% | 27.13% |
| 1000 0, 1000 1, repeat | 73.83% | 9220 | 1.35 | 12.28% | 4.45% | 13.11% |
| Rand. 4 chns, all '1' | 76.35% | 7174 | 1.05 | 19.60% | 12.79% | 34.35% |
| Rand. 4 chns, Fill adj. comp. | 76.56% | 6893 | 1.01 | 15.45% | 8.15% | 19.48% |
| Fill adj. comp. | 77.10% | 5740 | 0.84 | 14.58% | 6.99% | 22.43% |

Table 2: A table comparing fault coverage, pattern volume, and switching activity for transition patterns with varying fill heuristics for Module D.

| Fill heuristic | Covg. | Volume | Increase | %switch 1K | %switch all | in≠out |
|-------------------------------|--------|--------|----------|------------|-------------|--------|
| Baseline A | 92.13% | 2261 | N/A | 26.26% | 24.98% | 37.60% |
| Don't care 'X' | 91.89% | 6140 | 2.72 | 11.02% | 3.00% | 30.89% |
| Don't care '0' | 91.89% | 5689 | 2.52 | 18.53% | 3.88% | 4.35% |
| Don't care '1' | 92.04% | 5602 | 2.48 | 26.27% | 10.48% | 26.48% |
| 1000 0, 1000 1, repeat | 92.04% | 5453 | 2.41 | 17.73% | 8.04% | 15.68% |
| Rand. 4 chns, all '1' | 92.20% | 3809 | 1.68 | 24.86% | 20.19% | 30.64% |
| Rand. 4 chns, Fill adj. comp. | 92.20% | 3291 | 1.46 | 21.64% | 18.01% | 26.00% |
| Fill adj. comp. | 92.04% | 2212 | 0.98 | 17.13% | 11.18% | 27.36% |

Table 3: A table comparing fault coverage, pattern volume, and switching activity for transition patterns with varying fill heuristics for Module M.

diagrammatically and give results from its implementation on a recent ASIC design.

4.1 Module-based Designs

It is common for large ASIC designs to be subdivided into modules to enable a hierarchical ATPG strategy. Figure 3 depicts one such partitioning.

Scan pins are shared between scan chains belonging to different cores or modules as shown in Figure 3. Scan chain A belongs to Module A and scan chain B to Module B. The signal `scan_in` is the scan input signal connected to both scan chain inputs. The signal `scan_out` is the scan output signal for both chains and is driven either by chain A or B depending on the value of the “select” signal. This configuration can be generalized to multiple chains per core or module.

Power consumption can be high even if cores are tested independently due to the fact that the shared scan configuration causes scan data as well clock pulses to be sent to all modules, including those not under test. Clock gating to turn off the clocks of the modules not being tested is one possible solution, but may not be preferred due to added complexity,

such as in clock tree synthesis and/or timing closure flows.

The technique described in this paper provides a method of reducing power consumption in designs with shared scan configuration without adding any clock gating circuitry [26]. The technique relies on providing constant data to scan inputs of chains not used in testing by adding muxes and tie logic to scan inputs of all chains as shown in Figure 4. It includes logic to ensure all modules other than the module being tested are always in scan shift mode (`scan_enable=1`) to avoid capturing arbitrary data that overwrites the otherwise constant data being scanned in. The design changes required to permit this mode of operation are shown in Figure 4.

When “select” is 0, Module A in Figure 4 is ready to be tested, scan chain A in Module A is selected, while scan chain B in Module B is setup to receive a constant ‘0’ data. During the shift of scan chain A, the clock of Module B is not gated off, but because of the constant input at the scan in of chain B, signal transitions in the circuit are minimized thus reducing power. This condition requires Module B not to

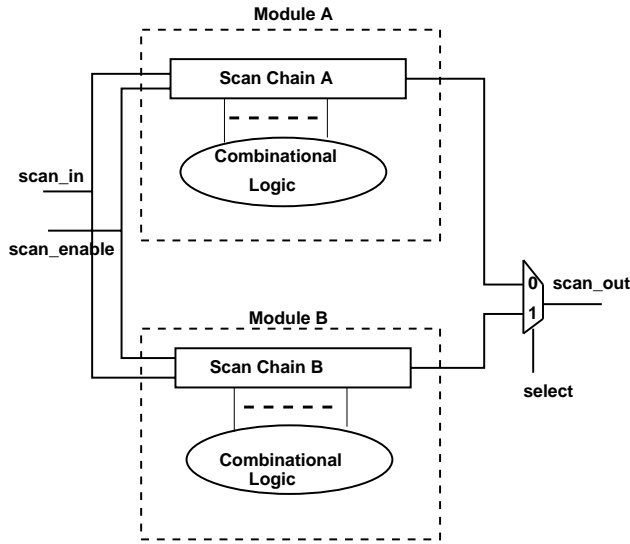


Figure 3: A diagram illustrating the hierarchical scan implementation.

capture from the combinational logic when the capture clock is applied to Module A with `scan_enable=0`. The `scan_enable` of Module B has to be maintained at a '1' value when "select" is 0 to achieve this behavior.

When "select" is 1, Module B is tested using scan chain B, while the input to scan chain A is held constant thus reducing power. The scan-out path is not affected - When "select" is 0, Module A is scanned out, else Module B is scanned out. This example can be generalized to designs that use multiple scan chains to test each core or module.

It should be noted that, for most practical design situations, the coverages achievable in a modular test approach may be deemed insufficient due to the likely poor coverage of any glue logic between the modules. So, most hierarchical scan situations will also include a "full chip" scan mode where all scan chains are concatenated and accessible simultaneously. Obviously, power again can be a concern for this mode of operation. However, experience shows that the number of patterns required in this mode is frequently a very small fraction of the total pattern set, and the shift frequency can be modulated downward to compensate for additional power consumption as required.

4.2 Experimental Results

The power reduction scheme was implemented on a .13 micron 5M gate ASIC. There were four sub-modules in the design and a "pwr_mode" signal controlled whether conventional scan was performed as in Figure 3 or a power-conscious scan was performed as in Figure 4. For this design, the shift and capture

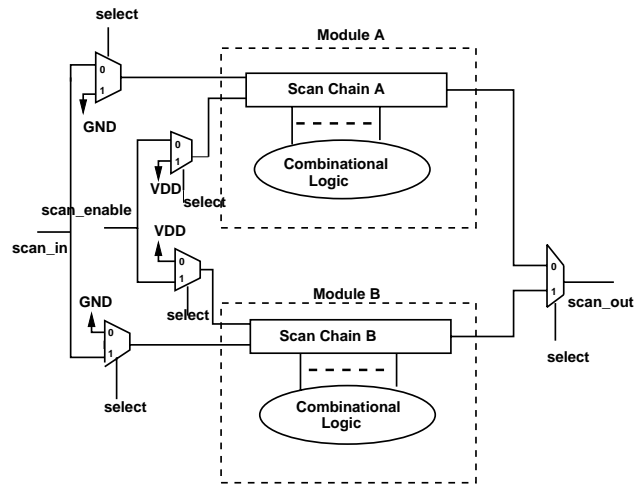


Figure 4: A diagram illustrating the hierarchical scan implementation with constant data inputs.

frequencies were the same (60 MHz). Power estimation via simulation was performed for power-conscious scan with "pwr_mode" = 1. When "pwr_mode" = 1, the sub-modules not being tested are receiving constant data as in Figure 4. The following procedure was used to estimate power for "pwr_mode" = 1:

- Serially simulate scan chain test (001100..) for the sub-module with the longest scan chain.
- At the end of the serial simulation, the circuit is assumed to be adequately initialized.
- After complete simulation of chain test, simulate a few additional (ten) cycles still applying the 0011 pattern.
- Measure toggle activity for these few cycles using simulation with timing SDF.
- Provide toggle activity information to a power estimation tool.
- Provide load information for the nets in the design to the power estimation tool.
- Estimate average power

Simulation results revealed that with "pwr_mode" = 1, the average power is 1.2W. The simulations were not performed for the whole chip with "pwr_mode" = 0, but simulations were performed for one module by itself. This module is roughly $\frac{1}{6}$ th the size of the entire design. Based on the power estimated for this module, the overall rough estimate for power was expected to be about 2.5W-3W. It should be noted that the power estimations may vary significantly based

on whether load information for nets is used or not. This information is typically not available until after layout.

4.3 Silicon Observations

Two types of patterns were generated for comparison per sub-module. One set of patterns (containing about 700 scan patterns) held “pwr_mode” at 1. The second set was a replica of the first except “pwr_mode” was set to 0 and one additional flip-flop masked. The additional masking was to account for a mismatch caused by a functional flip-flop that was fed by the functional pin shared as “pwr_mode” in test mode.

V_{min} , the minimum voltage at which the scan patterns pass on the tester, was recorded for both sets of sub-chips. V_{min} is an indication of the switching activity during the test [8]. The larger the switching activity, the higher the V_{min} . The expectation was that the V_{min} for patterns with “pwr_mode” = 0 would be higher than those with “pwr_mode” = 1. The results indicate that this is true as shown in Table 4 for a subset of patterns, especially for the stuck-at patterns. This data represents averages for samples taken from 24 wafers across split lot material. Tester accuracy for these voltage measurements is reported to be +/- 15 mV.

| Module and test | pwr_mode 1 | pwr_mode 0 |
|-----------------------|---------------|---------------|
| Module 1 Stuck-at | 1.18V | 1.39V |
| Module 1 Trans. fault | 1.25V | 1.33V |
| Module 2 Stuck-at | 1.23V | 1.39V |
| Module 2 Trans. fault | 1.27V | 1.29V |

Table 4: A table of V_{min} values for two modules with and without power conscious test.

A much later sample of material was examined, but this time only using two units. The variations observed between “pwr_mode” on/off were in the range of 60-100 mV for stuck-at tests for the two modules in Table 4. In this latter case, the differences for transition testing were almost negligible. We note that this design is atypical in that its shift and capture frequencies are the same, whereas most designs shift is significantly slower than capture. It should also be pointed out that the transition patterns contain dummy cycles between the completion of shift and the launch event, which is not true for the stuck-at patterns.

5 Future Work

The results shown here are promising, but more work remains to identify further heuristics which reduce switching power but add minimal test length. Furthermore, as the industry moves from stored deterministic pattern testing towards BIST techniques, a new class of low-power BIST pattern generators will be needed. Some work has been done in this area, such as the recent work by Rosinger et al. [27], but more is needed.

Reliable power estimates are also required early on in the design cycle to be able to architect power reduction schemes. More work remains to be done to obtain initial power estimates effectively.

In addition, in order to get most effective delay defect screening from the test, the switching activity should resemble as closely as possible functional switching activity and, more importantly, must not be significantly lower. One example might be that reduced switching activity increases the available slack on many/most paths in the design, limiting (increasing) the sizes of defects that are detectable. The process of “training” DFT and ATPG to produce a test which matches the functional operating conditions is not well-understood, and thus requires further research. The work we have presented here represents only a small step towards a more comprehensive solution to power management.

6 Conclusions

In this paper we have presented case study data on power issues during test, with emphasis on at-speed test power concerns. We analyzed various heuristics for creating lower power scan-based deterministic ATPG patterns which can reduce overall switching activity and thus power while minimizing the increase in pattern volume to maintain coverage. Of the various heuristics which could be considered practical, the fill adjacent technique seems to hold the most promise as far as lowering switching activity while keeping pattern volume in check.

The DFT technique for reduced power scan also seems to be effective and is relatively simple to implement due to the the complete avoidance of clock gating. The challenges that remain are in identifying further design techniques which can address areas of the design which have intense switching activity, and the difficulty in matching functional and test activity levels, neither of which this technique addresses.

7 Acknowledgments

Cy Hay of Synopsys is acknowledged for multiple useful conversations on low power pattern generation schemes.

8 References

- [1] P. C. Maxwell, R. C. Aitken, K. R. Kollitz and A. C. Brown, " I_{DDQ} and AC scan: The war against unmodelled defects," in *Proc. 1996 IEEE Int. Test Conf.*, Oct. 1996, pp. 250–258.
- [2] P. Nigh, W. Needham, K. Butler, P. Maxwell and R. Aitken, "An experimental study comparing the relative effectiveness of functional, scan, I_{DDQ} , and delay-fault testing," in *Proc. 1997 IEEE VLSI Test Symp.*, Apr.-May 1997, pp. 459–464.
- [3] ———, "So what *is* an optimal test mix? A discussion of the SEMATECH methods experiment," in *Proc. 1997 IEEE Test Conf.*, Nov. 1997, pp. 1037–1038.
- [4] P. Maxwell, I. Hartanto and L. Bentz, "Comparing functional and structural tests," in *Proc. 2000 IEEE Int. Test Conf.*, Oct. 2000, pp. 400–407.
- [5] C-W. Tseng and E. J. McCluskey, "Multiple-output propagation transition fault test," in *Proc. 2001 IEEE Int. Test Conf.*, Oct.-Nov. 2001, pp. 358–366.
- [6] K. Tumin, C. Vargas, R. Patterson and C. Nappi, "Scan vs. functional testing - A comparative effectiveness study on Motorola's MMC2107TM," in *Proc. 2001 IEEE Int. Test Conf.*, Oct.-Nov. 2001, pp. 443–450.
- [7] J. Saxena, K. M. Butler, J. Gatt, R. R., S. P. Kumar, S. Basu, D. J. Campbell and J. Berech, "Scan-based transition fault testing - Implementation and low cost test challenges," in *Proc. 2002 IEEE Int. Test Conf.*, Oct. 2002, pp. 1120–1129.
- [8] J. Saxena, K. M. Butler, V. B. Jayaram, S. Kundu, N. V. Arvind, P. Sreepakash and M. Hachinger, "A case study of IR-drop in structured at-speed testing," in *Proc. 2003 IEEE Int. Test Conf.*, Sep.-Oct. 2003, pp. 1098–1104.
- [9] K. S. Kim, S. Mitra and P. G. Ryan, "Delay defect characteristics and testing strategies," *IEEE Design and Test of Comput.*, vol. 20, no. 5, pp. 8–16, Sep-Oct. 2003.
- [10] X. Lin, R. Press, J. Rajski, P. Reuter, T. Rinderknecht, B. Swanson and N. Tamarapalli, "High-frequency, at-speed scan testing," *IEEE Design and Test of Comput.*, vol. 20, no. 5, pp. 17–25, Sep-Oct. 2003.
- [11] S. Pateras, "Achieving at-speed structural test," *IEEE Design and Test of Comput.*, vol. 20, no. 5, pp. 26–33, Sep-Oct. 2003.
- [12] A. L. Crouch, J. C. Potter and J. Doege, "AC scan path selection for physical debugging," *IEEE Design and Test of Comput.*, vol. 20, no. 5, pp. 34–40, Sep-Oct. 2003.
- [13] B. D. Cory, R. Kapur and B. Underwood, "Speed binning with path delay test in 150-nm technology," *IEEE Design and Test of Comput.*, vol. 20, no. 5, pp. 41–45, Sep-Oct. 2003.
- [14] R. Madge, B. R. Benware and W. R. Daasch, "Obtaining high defect coverage for frequency-dependent defects in complex ASICs," *IEEE Design and Test of Comput.*, vol. 20, no. 5, pp. 46–53, Sep-Oct. 2003.
- [15] J. Saxena, K. M. Butler and L. Whetsel, "An analysis of power reduction techniques in scan testing," in *Proc. 2001 IEEE Int. Test Conf.*, Oct.-Nov. 2001, pp. 670–677.
- [16] S. Wang and S. K. Gupta, "ATPG for heat dissipation minimization during scan testing," in *Proc. 1994 IEEE Int. Test Conf.*, Oct. 1994, pp. 250–258.
- [17] ———, "ATPG for heat dissipation minimization during scan testing," in *Proc. ACM/IEEE 34th Design Automation Conf.*, June 1997, pp. 614–619.
- [18] R. Sankaralingam, R. R. Oruganti and N. A. Touba, "Static compaction techniques to control can vector power dissipation," in *Proc. 2000 IEEE VLSI Test Symp.*, Apr.-May 2000, pp. 35–40.
- [19] B. B. Bhattacharya, S. C. Seth and S. Zhang, "Double-tree scan: A novel low-power scan-path architecture," in *Proc. 2003 IEEE Int. Test Conf.*, Sep.-Oct. 2003, pp. 470–479.
- [20] T. Yoshida and M. Watati, "A new approach for low power scan testing," in *Proc. 2003 IEEE Int. Test Conf.*, Sep.-Oct. 2003, pp. 480–487.
- [21] Y. Bonhomme, P. Girard, L. Guiller, C. Landrault and S. Pravossoudovitch, "Efficient scan chain design for power minimization during scan testing under routing constraint," in *Proc. 2003 IEEE Int. Test Conf.*, Sep.-Oct. 2003, pp. 488–493.

- [22] J. Rearick, personal communication, July 2003.
- [23] A. L. Crouch, personal communication, May 2002.
- [24] P. Wohl, J. A. Waicukauski, S. Patel and M. B. Amin, "Efficient compression and application of deterministic patterns in a logic BIST architecture," in *Proc. ACM/IEEE 40th Design Automation Conf.*, June 2003, pp. 566–569.
- [25] Synopsys Corporation, *Synopsys TetraMAX™ on-line help*. Mountain View, CA, Dec. 2003.
- [26] J. Saxena, K. M. Butler, A. K. Jain, A. Fr-yars and G. G. Hetherington, "Power reduction in module-based scan testing," US Patent and Trade Office, Pub. No. US 2002/0170010 A1, Nov. 2002.
- [27] P. Rosinger, B. M. Al-Hashimi and N. Nicolici, "Dual multiple-polynomial LFSR for low-power mixed-mode BIST," *IEE Proc.*, vol. 150, no. 4, pp. 209–217, Jul. 2003.