VLSI Testing and Test Power

Xiaoqing Wen
Department of Computer Systems and Engineering
Kyushu Institute of Technology
Iizuka, Japan
wen@cse.kyutech.ac.jp

***Abstract*—This paper first reviews the basics of VLSI testing, focusing on test generation and design for testability. Then it discusses the impact of test power in scan testing, and highlights the need for low-power VLSI testing.**

***Keywords*—VLSI testing, test generation, design for testability, scan design, scan testing, at-speed scan testing, test power, shift power, capture power, low-power testing**

I. INTRODUCTION

Ever-growing scale, ever-increasing complexity, and ever-shrinking process feature sizes make the design of VLSI circuits highly prone to design errors and their fabrication to manufacturing defects. Therefore, VLSI design verification is needed on the design side to determine whether a design is error-free, while VLSI testing is needed on the manufacturing side to determine whether a fabricated VLSI circuit is defect-free [1]. VLSI testing is necessary because numerous causes in a manufacturing process may potentially introduce such physical defects as shorts and opens into a circuit fabricated from an error-free design.

VLSI testing is conducted by (1) applying test stimuli to the ***circuit-under-test*** (***CUT***), (2) measuring actual circuit responses, and (3) comparing them with expected circuit responses [2-4]. The CUT is declared defective if measured and expected circuit responses do not match with each other. When yield improvement or reliability enhancement is required, detailed failure analysis needs to be conducted to identify the location and the root-cause of the defect(s).

There are two basic goals in VLSI testing, namely *high test quality* and *low test costs*.

Test quality is mainly determined by the ratio of *under-test* (i.e., defective circuits passing a test) and the ratio of *over-test* (i.e., defect-free circuits failing a test). Under-test is mostly caused by inadequate contents of test stimuli, also referred to as ***test patterns***. Over-test is conventionally blamed on the circuit model difference between test mode and functional mode, which causes some functionally-benign defects to fail a circuit only during testing.

On the other hand, test costs are mainly determined by test data volume and test time. Different from design and manufacturing where reuse and parallelism significantly reduces costs due to well-scaled efficiency, testing basically needs to deal with all individual components (transistors, interconnects, etc.), and usually does not benefit from improved efficiency in design and manufacturing. As a result, test costs have been increasing over the years, and may eventually become higher than manufacturing costs.

VLSI testing technologies have been evolving around these two goals, especially in the fields of ***test generation*** and ***design for testability*** (***DFT***). The former is about how to determine the contents of test patterns so as to achieve higher test quality with fewer test patterns, while the latter is about how to modify a circuit design so as to make its testing easier (e.g., making it possible to conduct test generation and test application in practical time) and/or more efficient (e.g., making it possible to reduce test data and/or test time).

Recently, power dissipation during testing, i.e., ***test power***, has emerged as a new threat to the quality and costs of VLSI testing, especially for low-power circuits [5-8]. The reason is that test power can be much higher than functional power. Excessive test power may cause heat damage and circuit malfunction. If left uncontained, test power may lower test quality due to over-test; if inefficiently managed, test costs often increase. Therefore, ***low-power testing*** is indispensable to modern low-power VLSI circuits. In order words, no matter how attractive a low-power circuit design is, it cannot be realized without low-power testing.

In the rest of this paper, basic concepts used in test generation and design for testability will be reviewed. Based on that, the impact of test power is described. The purpose is to highlight the need for low-power VLSI testing.

II. BASICS OF TEST GENEATION

Test generation is the process of creating test patterns for a circuit-under-test or CUT [2-4]. The ultimate goal is to determine whether the CUT is free of any manufacturing defects by applying test patterns and comparing measured test responses with expected test responses. In the following, three basic concepts in test generation are briefly described.

A. Fault Model

Although the ultimate goal of test patterns is to determine whether a CUT is defect-free, it is impossible to enumerate all physical defects as direct targets in test generation. In practice, a ***fault model*** assumed in a certain circuit model is needed to represent physical defects from one perspective or another. Some typical fault models are as follows:

- ***Stuck-At Fault Model***: A stuck-at fault assumes that the logic value of a signal line in a circuit is fixed at either 0 or 1. That is, a signal line is associated with a ***stuck-at-0*** fault and a ***stuck-at-1*** fault. Intuitively, a short defect to the ground (power supply) corresponds to a stuck-at-0 (stuck-at-1) fault.

However, it is well known that many other complex structural defects can also be represented by stuck-at faults.

- ***Bridging Fault Model***: A bridging fault assumes that two or more signal lines are shorted together and produce a final logic value according a certain rule. It provides a potentially better representation for short-type structural defects, although the final logic value is often hard to determine.
- ***Transition Delay Fault Model***: A transition delay fault assumes a transition (0→1 or 1→0) at a signal line cannot propagate to circuit outputs fast enough in a test cycle. A signal line is associated with a *slow-to-rise* transition delay fault and a *slow-to-fall* transition delay fault.
- ***Path Delay Fault Model***: A path delay fault assumes a transition along a path cannot propagate fast enough.

Conventionally, the stuck-at and bridging fault models represent the behaviors of structural defects that cause static logic value changes. With shrinking supply voltages and process feature sizes, timing-related defects that only affect the speed of a transition are on the rise. Such defects need to be covered by the transition and path delay fault models.

B. Automatic Test Pattern Generation (ATPG)

Test generation is the process of generating a set of test patterns to detect faults assumed under a selected fault model. As illustrated in Fig. 1, a test pattern *p* for a stuck-at fault *f* in a circuit means that the faulty circuit with *f* and the fault-free circuit without *f* produce different logic values on at least one output. Fault detection in the context of other fault models can be defined in a similar manner.

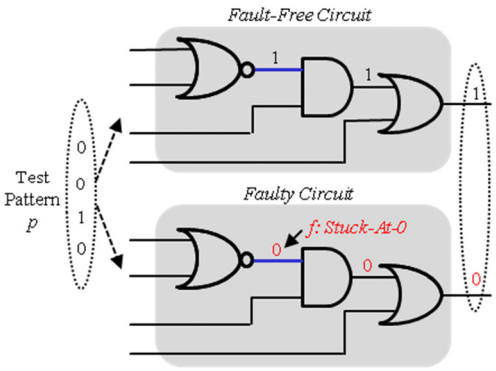


Figure 1. Concept of Fault Detection.

Fig. 2 shows an illustrative flow of test generation. The first step in test generation is to select a proper fault model, and all faults under the fault model are listed as target faults. In each test generation run, a primary fault is targeted and logic values necessary for its detection are determined. Various algorithms, such as D, PODEM, FAN, can be used for this purpose [2-4]. Note that not all inputs need to have logic values in order to detect the primary fault. If many inputs remain unspecified (indicated by *X*s in Fig. 2), one or more secondary faults can be further targeted for fault detection. This process is called ***dynamic compaction***, which helps reduce final test pattern count. Even after dynamic compaction, some inputs may still remain unspecified. Such an input combination is called a ***test cube***. ***X-filling*** is then used to assign logic values to *X*s in a test cube so as to obtain a fully-specified ***test pattern***. For example, ***random-fill*** can be conducted by assigning random logic values to all *X*s in a test cube. Random-fill has strong fortuitous fault detection capability, which helps reduce final test pattern count.

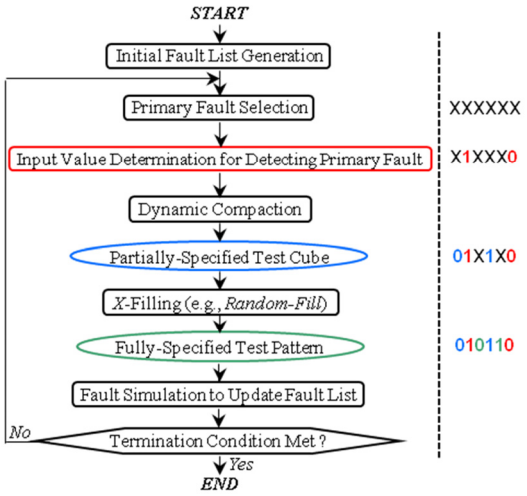


Figure 2. Illustrative Flow of Test Generaion.

The detection of a stuck-at or bridging fault needs one test pattern, while the detection of a transition or path delay fault needs a pair of test patterns. This is because in order to detect a delay fault, a path corresponding to the fault needs to be sensitized, a transition needs to be created at the start point of the path, and the test response needs to be measured at the end point of the path at the required timing.

C. Test Quality Assessment

In test generation, the quality of generated test patterns needs to be properly assessed. The most commonly used metric for this purpose is ***fault coverage***, which is the ratio of faults detected by a set of test patterns to the total number of faults listed under a fault model [2-4]. Since a fault model is an indirect representation of the behaviors of physical defects, in some cases there is a need to assess the capability of test patterns in detecting unmolded physical defects. A metric for this purpose is ***bridging coverage estimate*** (***BCE***) [9], which takes the number of times that a stuck-at fault is detected by a test pattern set into consideration so as to access the capability of the set to detect bridging-type defects. The delay test quality of a transition delay test set can be assessed by the ***statistical delay quality level*** (***SDQL***) metric [10], which takes the sizes of delay defects and the lengths of sensitized paths into consideration. SDQL is important since small-delay defects are becoming dominant timing-related defects and transition fault coverage alone cannot accurately assess the capability of a transition delay test pattern set in detecting small-delay defects. In addition, ***output deviation*** can be used as an effective surrogate metric [11].

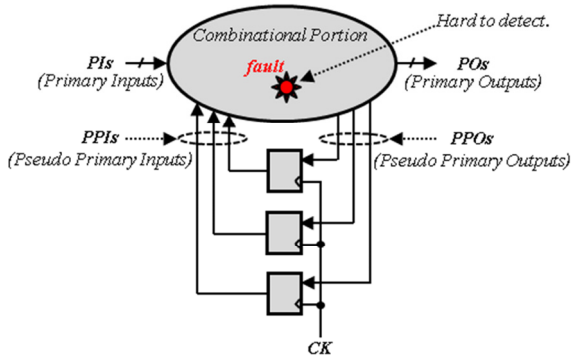
III. Basics of Design for Testability

Test generation for a large combinational circuit may take days, if not weeks, to complete. For a sequential circuit, even a modest one, its test generation time can easily become

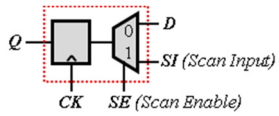
prohibitively long. This means that industrial circuits, which are mostly sequential in nature with tens of thousands of flip-flops (FFs), cannot be tested as is. A typical solution is to convert a sequential circuit into a ***scan design*** so that its testing can be conducted through targeting the combinational portion of the original circuit. Scan design is the most fundamental type of ***design for testability*** (***DFT***) [2-4].

In addition to making a sequential circuit easily testable, DFT also manifests itself as ***built-in self-test*** (***BIST***) [2-4], in which all test tasks (test generation, test application, and test response analysis) are conducted in the circuit-under-test by specially added test circuitry. Furthermore, DFT in the form of on-chip test stimulus decompressors and test response compressors / compactors can help reduce final test data volume dramatically in ***test compression*** [4, 12].

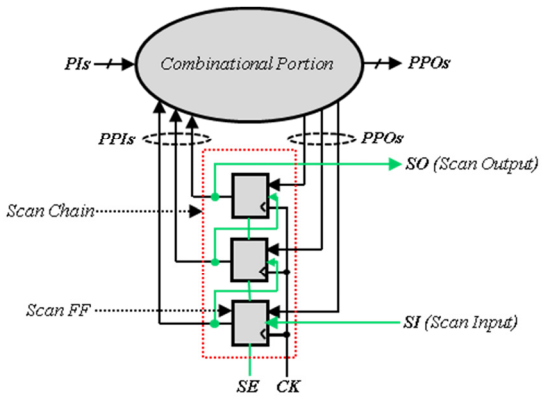
In the following, the most fundamental DFT technology, i.e., scan design, and its testing are briefly described.



(a) *Original design*



(b) *Scan FF*



(c) *Scan design*

Figure 3. Example of Scan Design.

A. Scan Design

Fig. 3 (a) shows a sample sequential circuit, which consists of a combinational portion and three FFs. For the combinational portion, external inputs are called ***primary inputs*** (***PIs***) and internal inputs from the FFs are called ***pseudo primary inputs*** (***PPIs***). In addition, external outputs are called ***primary outputs*** (***POs***) and internal outputs to the FFs are called ***pseudo primary outputs*** (***PPOs***). Since *PPI*s are hard to control and *PPO*s are hard to observe, faults in the combinational portion are usually hard to detect.

The most common DFT technique to solve this problem is scan design [4]. The basic idea is to replace all functional FFs with ***scan FFs*** and connect them into ***scan chains***.

There are many different scan FF designs, and Fig. 3 (b) shows a typical one called the ***MUX-D scan FF***. Here, a MUX is added to the input of a D-FF, resulting in two inputs to the scan FF: one being the original D input and the other being a new input called ***scan input*** (***SI***). The selection signal for the MUX, called ***scan enable*** (***SE***), determines which input is to be used to update the output *Q*.

In a scan design, functional FFs are first replaced with scan FFs. Then, scan FFs are connected into scan chains. An example is shown in Fig. 3 (c). Here, the *D* inputs of all scan FFs come from the combinational portion, while the *SI* input of each scan FF (except the first one) is connected to the *Q* output of the preceding scan FF in a scan chain. The *SI* input of the first scan FF in a scan chain is directly controllable from the outside, and it is called the ***scan input*** (***SI***) of the scan chain. On the other hand, the *Q* input of the last scan FF in a scan chain is directly observable from the outside, and it is called the ***scan output*** (***SO***) of the scan chain. Note that a scan design may have multiple scan chains and its number is usually limited by the number of pins available for corresponding *SI*s and *SO*s of the scan chains. The *SE* inputs of all scan FFs in a scan chain are connected together to become the ***scan enable*** (***SE***) of the scan chain.

B. Scan Testing

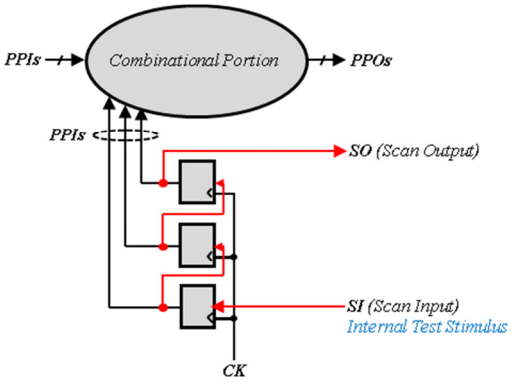
VLSI testing conducted for a scan design is called ***scan testing***. Fig. 4 illustrates how scan testing is conducted for the example scan design shown in Fig. 3 (c). Basically, two operation modes (i.e., ***shift*** and ***capture***) provided by scan design, are used for scan testing. These two modes are switched back and forth by using the *SE* signal [4].

Shift mode is entered by setting $SE = 1$. The equivalent circuit for shift mode is shown in Fig. 4 (a). It can be seen that the scan chain operates as a shift register in shift mode, in which the *PPI* values (test stimulus) for a new test pattern are shifted-in serially from *SI* and the *PPO* values (test response) for the previous test pattern are shifted-out serially from *SO*. It is clear that with this shift register function provided in shift mode, test stimuli can be easily set to all *PPI*s and test responses from all *PPO*s can be easily observed. Note that *N* shift clock pulses need to be applied, where *N* is the length of the longest scan chain in the scan design.

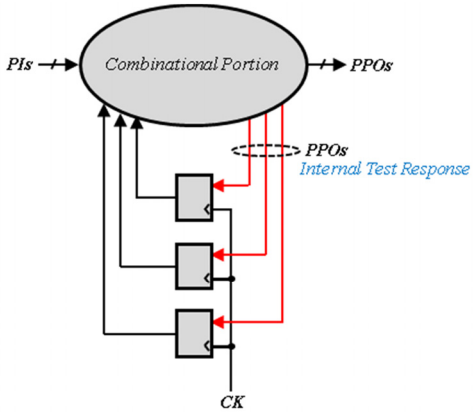
Capture mode is entered by setting $SE = 0$. The equivalent circuit for capture mode is shown in Fig. 4 (b). Obviously, this circuit configuration is the same as its original (or functional) configuration. Basically, one or two capture clock pulses are applied to load the *PPO* values (internal part of the test response for the previous test pattern) into all FFs, in order to be shifted-out in the next shift operation.

Fig. 4 (c) shows the timing waveform for one test pattern for the scan design shown in Fig. 4 (c). Three shift clock pulses are applied in shift mode since the scan chain has three scan FFs. In capture mode, one capture clock pulse is applied long after the last shift clock pulse is applied. Such scan testing is called ***slow-speed scan testing***, which is used to test for non-delay-type faults with test patterns generated for stuck-at faults, bridging faults, etc.

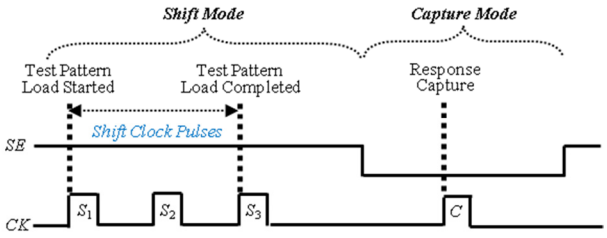
Scan testing is conducted by repeating shift and capture for all test patterns, referred to as ***scan test patterns***. Note that scan test patterns are generated for the combinational portion of a sequential circuit since scan design makes it possible to efficiently detect faults in the combinational portion in a sequential circuit. Since test generation and test application for a sequential circuit is virtually impossible, scan design is truly indispensable for VLSI testing.



(a) *Shift mode* (*SE* = 1)



(b) *Capture mode* (*SE* = 0)



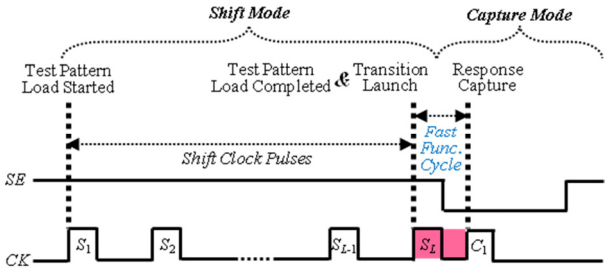
(c) *Timing waveform*

Figure 4. Example of Scan Testing.

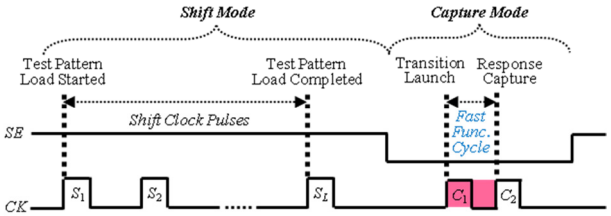
C. At-Speed Scan Testing

Shrinking feature sizes and increasing clock frequencies have made timing-related defects a major cause for failing VLSI circuits. Testing for such defects needs delay test patterns, which are usually generated with the transition delay fault model or the path delay fault model. In addition to delay test patterns, delay test application is usually conducted in the form of at-speed scan testing, in which a transition is created at the start point of a path and whether it can propagate to the end point of the path in a functional clock cycle is checked by making use of scan design [4].

There are two schemes available for realizing at-speed scan testing, namely ***launch-on-shift*** (***LOS***) and ***launch-on-capture*** (***LOC***), as described below.



(a) *Launch-On-Shift (LOS) scheme*



(b) *Launch-On-Capture (LOC) scheme*

Figure 5. At-Speed Scan Testing.

The LOS scheme is illustrated in Fig. 5 (a). This scheme generates transitions at the start points of sensitized paths by the differences of logic values loaded into FFs by the next-to-last shift clock pulse S_{L-1} and the last shift clock pulse S_L. Test responses are captured by the capture clock pulse C_1. Note that the time between the last shift clock pulse S_L and the capture clock pulse C_1 is set to be equal to the functional cycle in order to realize at-speed scan testing. Since only one capture clock pulse is used in capture mode, the LOS scheme is also referred to as the ***single-capture*** scheme.

The LOC scheme is illustrated in Fig. 5 (b). Different from the LOS scheme, this scheme generates transitions at the start points of sensitized paths by the differences of logic values loaded into FFs by the last shift clock pulse S_L and the first capture clock pulse C_1. Test responses are captured by the second capture clock pulse C_2. Note that the time between the first capture clock pulse C_1 and the second capture clock pulse C_2 is set to be equal to the functional cycle in order to realize at-speed scan testing. Since two capture clock pulses are used in capture mode, the LOC scheme is also referred to as the ***double-capture*** scheme.

Generally, the LOC scheme is easier to implement than the LOS scheme since the latter requires an at-speed *SE* signal that needs to be treated in a similar way as a clock signal in physical design. On the other hand, the LOS scheme usually achieves higher fault coverage than the LOC scheme. This is because it provides better controllability and observability due to the use of a single capture clock.

Transition delay test patterns are widely used for at-speed scan testing. However, high transition fault coverage may not directly translate into high at-speed scan test quality. This is because at-speed scan test quality also depends on the relations among (1) the lengths of sensitized paths, (2) the extra delays caused by timing-related defects, and (3) the length of the test cycle, as illustrated in Fig. 6. Especially, shrinking process feature sizes and increasing clock frequencies have increased the occurrence possibility of small-delay defects, which only cause a slightly increased delay at the defect site. As shown in Fig. 6, if the sensitized path passing through a small-delay defect site is relatively short with respect to the test cycle, at-speed scan testing cannot detect such a defect. Only when the sensitized path is long enough can such a small-delay defect be detected in at-speed scan testing. In order to achieve this goal, transition test generation needs to make sensitized paths as long as possible. Such test generation is called ***small-delay test generation*** [13]. The quality of a small-delay test pattern set can be assessed with the SDQL metric [10].

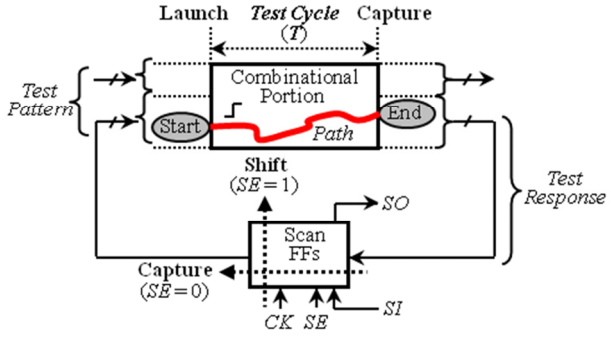


Figure 6. Sensitzied Path and At-Speed Scan Test Quality.

IV. Impact of Scan Test Power

Power dissipation occurs during both functional and scan testing operations. However, functional power and test power have so different characteristics that, even if functional power is reduced to a low level, scan test power may still be excessively high to cause serious problems in testing [5-8].

A. Power Disspipation

There are two types of power: ***dynamic power*** and ***static power***. Dynamic power depends on *operation-independent factors* (supply voltage, load capacitance, frequency) and *operation-dependent factors* (switching activity level), while static power depends on *operation-independent factors* (supply voltage, threshold voltage, channel dimension, doping profile, drain-source junction depth, pate-oxide thickness temperature) and *operation-dependent factors* (input vector contents), as summarized in Fig. 7.

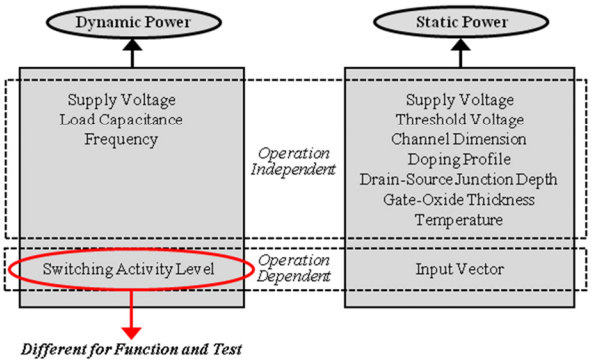


Figure 7. Gap between Functinoal Power and Test Power.

Operation-dependent factors, especially switching activity level for dynamic power, can have significantly different characteristics during functional operations and scan testing. As a result, a gap between functional power and test power often exists, which may cause serious problems during scan testing. As illustrated in Fig. 8, compared with standard design, low-power design makes full use of various hardware /software-based power management techniques (such as power gating, clock gating, etc.) to reduce functional power to a significantly low level. However, this effort does not help reduce scan test power. Due to constraints from test time and test costs, scan testing needs to use non-functional clocking / test data as well as fault / block parallelism (i.e., testing for multiple faults and blocks simultaneously) in order to increase test efficiency. This inevitably increases switching activity, resulting in excessive test power. For example, it was reported in 1993 that test power could be 2X higher than functional power [14]. This gap has been steadily growing and was found to have reached 5X in 2008 [15]. Excessive test power is starting to cause various problems in scan testing, especially in at-speed scan testing.

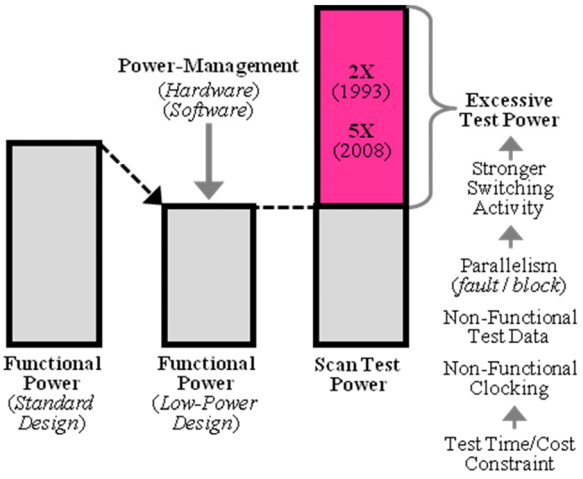


Figure 8. Gap between Functinoal Power and Test Power.

Since scan testing consists of shift and capture operations, there are two types of scan test power, namely ***shift power*** and ***capture power***. Their impacts are illustrated in Fig. 9.

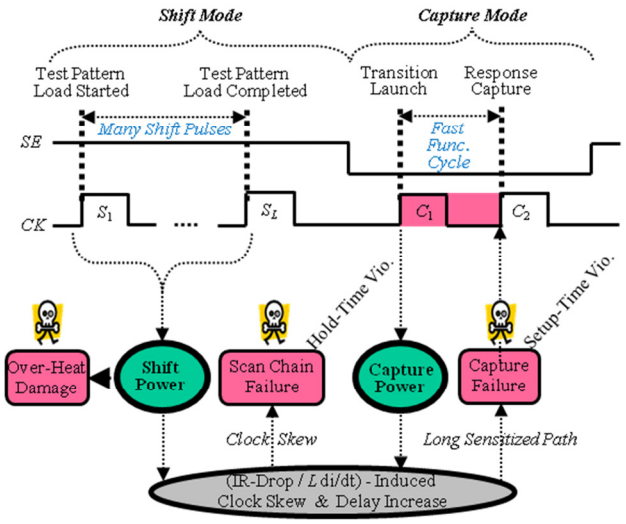


Figure 9. Test Power Impact in LOC-Based At-Speed Scan Testing.

B. Impact of Shift Power

The shift operation is usually conducted at a slower speed, even in at-speed scan testing that needs a very fast test cycle that is equal to the functional cycle. However, the shift operation requires N shift clock pulses, where N is the length of the longest scan chain. Therefore, shift time is usually much longer than capture time, and the accumulative impact of shift power is dominant. High average shift power leads to high heat dissipation, which may cause chip/package damage as well as reliability degradation due to hot spots. This problem is especially serious for low-power consumer devices, which have low heat dissipation bounds. In addition, high peak shift power causes shift clock skew and scan path delay increase induced by IR-drop or $L\ di/dt$. This may cause scan chain failures, rendering scan testing impossible.

In order to address the shift power problem, providing better cooling and/or using better package materials help but are too expensive to be a viable solution. Slowing down the shift clock speed also helps but results in longer test time. More sophisticated solutions include reducing switching activity caused by the shift operation through low-shift-power test generation or DFT techniques [5-8, 16].

C. Impact of Capture Power

The capture operation in at-speed scan testing only needs one (in the LOS scheme) or two (in the LOC scheme) capture clock pulses. Therefore, the accumulative impact of capture power is not a problem. However, instantaneous switching activity caused by the launching of transitions at the start points of sensitized paths (by S_L in the LOS scheme shown in Fig. 5 (a) or by C_1 in the LOC scheme shown in Fig. 5 (b)) results in IR-drop and ground bounce in the power supply network, which reduces effective power supply voltages to logic elements. As a result, the delay of such a logic element increases, resulting in increased total delay along sensitized paths. When this path delay increase is excessively large, false logic values will be captured at the end points of sensitized paths by C_1 in the LOS scheme shown in Fig. 5 (a) or by C_2 in the LOC scheme shown in Fig. 5 (b). Clearly, this leads to unacceptably high test-power-induced yield loss.

In order to address the capture power problem, over-designing the power supply network is not a viable solution. Generally, this problem has to be solved by reducing switching activity caused by transition launch (also referred to as ***launch switching activity*** (***LSA***)) through low-capture-power test generation or DFT techniques [6-9, 16].

V. Conclusions

This paper reviewed the basics of VLSI testing, focusing on test generation and scan design. It then discussed the impact of scan test power. As made clear in this paper, low-power VLSI testing has become indispensable in testing, especially for low-power VLSI circuits.

References

[1] L.-T. Wang, Y.-W. Chang, and K.-T. (Tim) Cheng, Eds., *Electronic Design Automation: Synthesis, Verification, and Test*, San Francisco: Morgan Kaufmann, 2008.

[2] M. Abramovici, M. A. Breuer, and A. D. Friedman, *Digital Systems Testing and Testable Design*, Piscataway, New Jersey: IEEE Press, Revised Printing, 1994.

[3] M. L. Bushnell and V. D. Agrawal, Essentials of Electronic Testing for Digital, Memory & Mixed-Signal VLSI Circuits, New York: Springer Science, 2000.

[4] L.-T. Wang, C.-W. Wu, and X. Wen, Eds., *VLSI Test Principles and Architectures: Design for Testability*, San Francisco: Morgan Kaufmann, 2006.

[5] P. Girard, N. Nicolici, and X. Wen, Eds., *Power-Aware Testing and Test Strategies for Low Power Devices*, New York: Springer, 2009.

[6] P. Girard, X. Wen, and N. A. Touba, "Low-Power Testing," in *Advanced SOC Test Architectures – Towards Nanometer Designs*, L.-T. Wang, C. Stroud, and N. A. Touba, Eds., San Francisco: Morgan Kaufmann, 2007, pp. 307–350.

[7] C. P. Ravikumar, M. Hirech, and X. Wen, "Test Strategies for Low-Power Devices," *J. of Low Power Electronics*, Vol. 4, No.2, pp. 127-138, August 2008.

[8] M. Tehranipoor and K.M. Butler, "*Power Supply Noise: A Survey onEffects and Research*," *IEEE Design & Test of Computers*, Vol. 27, No.2, pp. 51-67, March-April 2010.

[9] B. Benware, C. Schuermyer, S. Ranganathan, R. Madge, P. Krishnamurthy, N. Tamarapalli, K.-H. Tsai, and J. Rajski, "Impact of Multiple-Detect Test Patterns on Product Quality," *Proc. IEEE Intl. Test Conf.*, pp. 1031-1040, 2003.

[10] Y. Sato, S. Hamada, T. Maeda, A. Takatori, Y. Nozuyama, and S. Kajihara, "Invisible Delay Quality - SDQM Model Lights Up What Could Not Be Seen," *Proc. IEEE Int'l Test Conf.*, Paper 47.1, 2005.

[11] Z. Wang and K. Chakrabarty, "Test-Quality/Cost Optimization Using Output-Deviation-Based Reordering of Test Patterns," *IEEE Trans. TCAD,* Vol. 27, No. 2, pp. 352-365, Feburary 2008.

[12] N.A. Touba, "Survey of Test Vector Compression Techniques," *IEEE Design & Test Magazine,* Vol. 23, issue 4, pp. 294-303, 2006.

[13] X Lin, K. Tsai, C. Wang, M. Kassab, J. Rajaski, T. Kobayashi, R. Klingenberg, Y. Sato, S. Hamada, and T. Aikyo, "Timing-Aware ATPG for High Quality At-Speed Testing of Small Delay Defects," *Proc. IEEE Asian Test Symp.*, pp.139-146, 2006.

[14] Y. Zorian, "A Distributed BIST Control Scheme for Complex VLSI Devices," *Proc. IEEE VLSI Test Symp.*, pp. 4-9, 1993.

[15] S. Sde-Paz and E. Salomon, "Frequency and Power Correlation between At-Speed Scan and Functional Tests," *Proc. IEEE Int'l Test Conf.*, Paper 13.3, 2008.

[16] X. Wen, "Low-Power Testing for Low-Power Devices," *WRTLT 2010*, http://aries3a.cse.kyutech.ac.jp/~wen/WRTLT_120510.pdf