Reducing Power Dissipation During Test Using Scan Chain Disable

Ranganathan Sankaralingam[1], Bahram Pouya[2], and Nur A. Touba[1]

[1]Computer Engineering Research Center
Dept. of Electrical and Computer Engineering
University of Texas, Austin, TX 78712

[2]Motorola, SPS, Embedded Platform Solutions
7700 West Parmer Lane
Austin, TX 78729-8084

Abstract

A novel approach for minimizing power during scan testing is presented. The idea is that given a full scan module or core that has multiple scan chains, the test set is generated and ordered in such a way that some of the scan chains can have their clock disabled for portions of the test set. Disabling the clock prevents flip-flops from transitioning, and hence reduces switching activity in the circuit. Moreover, disabling the clock also reduces power dissipation in the clock tree which often is a major source of power. The only hardware modification that is required to implement this approach is to add the capability for the tester to gate the clock for one subset of the scan chains in the core. A procedure for generating and ordering the test set to maximize the use of scan disable is described. Experimental results are shown indicating that the proposed approach can significantly reduce both logic and clock power during testing.

1. Introduction

Power dissipation during scan testing can be much greater than during normal operation. During normal circuit operation, typically a relatively small percentage of the flip-flops change value in each clock cycle. However, when scanning in test vectors, typically a much larger percentage of the flip-flops will change value in each clock cycle. This results in more switching activity in the circuit. The increased power during test can cause problems with heat dissipation. The amount of heat that can be safely dissipated without damaging the chip is limited by the chip's package. This must be taken into consideration when scheduling tests and limits either the number of cores that can be tested simultaneously [Zorian 93] or the speed at which the tests can be applied. This in turn impacts test time and test cost. With the proliferation of portable battery operated devices and the drive toward low power design and low-cost light packages, the power issues for test are becoming increasingly important [Crouch 99]. Techniques for minimizing power during test are needed.

Test scheduling algorithms that satisfy power constraints were presented in [Chou 94]. Low power BIST techniques were presented in [Wang 97a], [Hertwig 98], [Wang 99], [Gerstendörfer 99], [Girard 99, 00]. Techniques for minimizing power dissipation when testing combinational circuits were presented in [Wang 94], [Dabholkar 98] and for scan circuits in [Wang 97b], [Dabholkar 98], [Sankaralingam 00], [Whetsel 00].

The focus of this paper is on the problem of minimizing power dissipation during scan testing. [Wang 97b] modifies the controllability and observability cost functions in PODEM [Goel 81] to find a set of scan vectors that minimizes test power. [Sankaralingam 00] describes a guided algorithm for static compaction of scan vectors to minimize switching activity. [Dabholkar 98] proposes heuristic algorithms for test vector ordering and scan chain ordering that minimize average power during scan testing. [Whetsel 00] describes an adapted scan chain architecture that segments a single scan chain to minimize switching activity during scan shifting.

It should be noted that many methods that are targeted towards reducing test time during scan testing also reduce power dissipation because they result in less switching activity. Some noteworthy techniques that fall into this category are the following: [Chen 92] varies the logical length of the scan chain so that the test length is reduced by starting with a short scan chain and increasing its length as the test session proceeds. [Su 93] reduces the number of scan shifts by exploiting overlap between the new bits to be scanned in and the existing response bits waiting to be scanned out. [Higami 94] partially shifts the scan chain so that only the scan elements close to the scan input (scan output) are controlled (observed). [Lai 93] combines non-scan and scan testing to reduce scan shifting. [Gupta 91] and [Narayanan 92] describe methods for reducing test time for circuits that can be divided into disjoint portions that can be tested independently.

In this paper, we propose a new approach for minimizing power during scan testing. There are two sources of power dissipation during scan testing. One is the power dissipated when the outputs of logic gates in the circuit switch, which will be referred to here as "logic power", and the other is the power dissipated in the clock tree each time the clock makes a transition, which will be referred to here as the "clock power." The previously

1093-0167/01 $10.00 © 2001 IEEE

proposed approaches have all focused on reducing the logic power and do not do anything to reduce the clock power. Results in [Pouya 00] suggest that clock power is a significant component of the total power during testing. In this paper, we propose an approach that reduces both logic power and clock power.

The idea proposed here is that given a module or core that has multiple scan chains, we generate and order the test set in such a way that some of the scan chains can have their clock disabled for portions of the test set. For example, suppose a core has 8 scan chains and 300 test vectors. If we can generate and order the test vectors in such a way that we do not need to clock 4 of the scan chains for 200 of the test vectors, then we can reduce the total power (both logic power and clock power) during test by roughly 33%. When the clock is disabled for 4 of the scan chains, the contents of those scan chains remain constant which reduces circuit switching hence reducing logic power. Moreover, the clock power is reduced by a factor of two because half of the scan elements in the circuit are not clocked.

The proposed approach is highly effective because most of the test vectors in a test set target hard faults localized in regions with poor controllability or observability. Scan elements not needed to control the inputs or observe the outputs of these regions can have their clock disabled for many test vectors without losing fault coverage.

The methodology described in this paper can be used for full scan cores having multiple scan chains. The only hardware modification that is required to implement this approach is to add the capability for the tester to gate the clock for one subset of the scan chains in the core.

Note that the methodology described in this paper is different from the conventional notion of test scheduling [Chou 94] which involves scheduling groups of cores to be tested concurrently under the test power constraints imposed by the package. The methodology described here is applied to a particular core to reduce its power dissipation during test. This allows more cores to be tested concurrently without violating test power constraints imposed by the package.

2. Overview of Proposed Scheme

The only hardware modification required for the proposed scheme is to add the capability to disable the clock for a subset of the scan chains. The scan chains in the core are divided into two sets, set A and set B. An extra "disable" input is added to the core, as illustrated in Fig. 1, which allows the tester to control when the clock is disabled to the scan elements in set B. For Mux-D type scan elements, the system clock is gated and controlled by *disable* so that when *disable* is activated, the scan chains in set B will neither shift in scan mode, nor capture in system mode. They are not clocked, and simply hold a constant value. If there is a separate scan clock, then both the system clock and scan clock are gated and controlled by *disable.* So in either case, when *disable* is activated, the scan chains in set B will not receive any clock transitions either in scan mode or system mode and no clock power will be dissipated for those scan chains. Note that the scan chains in set A are never disabled and operate normally at all times.

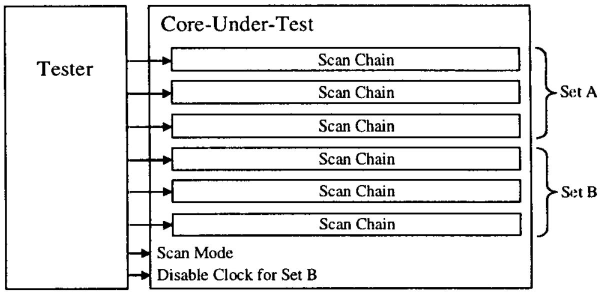

Figure 1. Block Diagram of Proposed Scheme

When *disable* is active, set B is not clocked and it doesn't shift or capture. The key is to generate and order the test vectors in such a way that for many of the scan and capture operations, set B is unused and hence power can be saved by activating *disable.*

The way *disable* is used is as follows. Some test vector t_1 is shifted into all scan chains. Then *disable* is activated before the capture cycle. In the capture cycle, only the scan chains in set A capture the response to test vector t_1. *disable* remains activated as the next test vector t_2 is shifted in, so only the scan chains in set A are loaded with new values as their captured response is shifted out. The scan chains in set B retain the same value that they had for test vector t_1. Before the capture cycle for test vector t_2, *disable* can be deactivated so that all scan chains will capture the response for test vector t_2. So there are two requirements that must be satisfied in order for it to be possible to activate *disable* between test vector t_1 and t_2:

Requirement 1: It must be sufficient to partially capture the response of test vector t_1 only in the scan elements in set A without reducing the overall fault coverage of the test set.

Requirement 2: Test vector t_1 and t_2 must have the same specified values for the scan elements in set B.

By carefully choosing which scan elements to include in set B, it is possible to generate and order the test vectors in a way that allows frequent use of *disable* to save power. This process is described in the following sections.

3. Choosing Scan Chains in Set *B*

The scan elements to be put in set *B* should be chosen judiciously. This is done by performing ATPG (automatic test pattern generation) for the core-under-test and analyzing the resulting test cubes. *Test cubes* are deterministic test vectors in which the unassigned inputs are left as X's.

Consider a matrix in which each row corresponds to a test cube, and each column corresponds to a scan element. Each entry is either a 0, 1, or X. Let a *rectangle* in the matrix be defined as a set of columns and rows in which the entries in each column of the rectangle are either all 0's and X's or all 1's and X's for each row in the rectangle. Consider the example in Fig. 2 containing 5 test cubes. An example of a rectangle in this matrix would be the rectangle that includes the rows {2, 3, 4} and columns {3, 4, 5, 6}. Consider column 3, it has all X's and 1's for the set of rows in the rectangle. Column 6 has all 0's and X's. Row 1 could not be added to this rectangle because it would cause a conflict in column 3 because it has a 0 in that column whereas the other rows in the rectangle have either a 1 or an X in that column. Note that while the rows and columns in this example rectangle are consecutive, that need not be the case in general.

$$\begin{array}{cc|cccc|ccc}
\multicolumn{1}{c}{1} & \multicolumn{1}{c}{1} & \multicolumn{1}{c}{0} & 0 & X & \multicolumn{1}{c}{0} & \multicolumn{1}{c}{X} & 0 & 1 \\ \cline{3-6}
1 & 0 & X & 0 & 1 & X & 0 & 1 & 0 \\
0 & 1 & 1 & 0 & X & 0 & X & X & X \\
1 & 1 & X & 0 & X & 0 & 1 & 0 & 0 \\ \cline{3-6}
\multicolumn{1}{c}{0} & \multicolumn{1}{c}{1} & \multicolumn{1}{c}{0} & 1 & X & \multicolumn{1}{c}{0} & \multicolumn{1}{c}{0} & 1 & 1
\end{array}$$

Figure 2. Example of a Rectangle in a Test Cube Matrix

A good heuristic for choosing which scan elements to include in set *B* is to find the largest rectangle in the test cube matrix and include the scan elements corresponding to the columns of the rectangle in set *B*. The reason for this is that all of the test cubes corresponding to the rows in the rectangle would be candidates for using *disable* because they would be compatible in the scan elements in set *B*. If equal-sized scan chains are needed, then some columns can be removed from the rectangle to make the number of columns a multiple of the number of scan chains. Finding the largest rectangle in a matrix is an NP-complete problem, however there are good approximate algorithms for it [Brayton 87].

So using the heuristic of finding the largest rectangle in a test cube matrix, we can select the set of scan elements that should be included in set *B*, and then the remaining scan elements should be included in set *A*. When the scan chains are stitched together, the scan elements in set *A* should be used to construct one set of scan chains, and the scan elements in set *B* should be used to construct the other set of scan chains. The actual scan ordering does not matter and can be optimized to reduce routing complexity.

4. Test Set Generation and Ordering

In this section, we present a procedure for generating and ordering the test set to maximize the use of *disable*. The first step is to perform ATPG. Random test generation is done first to detect the easy faults, and then ATPG is performed to generate test cubes for the remaining faults. The random test vectors (which can be thought of as test cubes with no unspecified values) that detected faults are combined with the test cubes for the hard faults to form the complete set of test cubes that covers all faults. The test cubes are statically compacted, i.e., compatible test cubes are merged together. The largest rectangle in these test cubes is then identified and used to partition the scan elements into set *A* and set *B* as described in Sec. 3. The test cubes are then divided into disjoint "*B*-compatible" groups where all the test cubes in a "*B*-compatible" group have non-conflicting values for the scan elements in set *B*. An example of dividing a set of test cubes into *B*-compatible groups is shown in Fig. 3. The first 5 bit positions correspond to the scan elements in set *A*, and the last 5 bit positions correspond to the scan elements in set *B*. The test cubes are grouped together so that there are no conflicts in the last 5 bit positions between any test cubes in the same group. Note that the grouping of test cubes into *B*-compatible groups is not unique, different groupings can be obtained because *B*-compatibility is not a transitive relation.

The test cubes are now partitioned into two sets which we will refer to as the full observability and controllability (FOC) set and the partial observability and controllability (POC) set. The *B*-compatible groups containing more than one test cube are placed in the POC set, and the rest of the test cubes are placed in the FOC set. The test cubes in the POC set are candidates for using the *disable* signal because they can satisfy Requirement 2 as explained in Sec. 2, while the test cubes in the FOC set cannot possibly use *disable* because they are not compatible with any other test cube for the scan elements in set *B*.

For the test cubes in the POC set, any X's for scan elements in set *B* are specified with the appropriate value (either 0 or 1) such that all the test cubes in each *B*-compatible group have the same specified value for each scan element in set *B*. How the *B*-compatible groups from Fig. 3 would be specified is shown in Fig. 4. After this is done, any remaining X's in any of the test cubes in either the POC or FOC set are randomly filled with 0's and 1's. At this point, the test cubes are all fully specified test vectors. Fault simulation is done assuming full observability of all test vectors, and any test vector that does not detect any new faults is dropped. The order of the test vectors is reversed, and fault simulation is repeated again to drop any test vector that does not detected any new faults.

	Set A	Set B
B-Compatible Group 1	1 1 0 0 X	0 X 0 1 0
	1 0 X 1 1	X 1 0 1 X
	0 1 1 0 X	0 X X X X
	1 1 X 0 X	0 1 0 X X
B-Compatible Group 2	0 1 0 1 0	1 X 1 1 1
	1 X 0 0 X	X X 1 1 X
B-Compatible Group 3	X 1 1 0 X	0 0 X 0 X
	0 1 0 X X	0 X 1 0 X
B-Compatible Group 4	X 1 0 1 X	1 0 0 0 X
B-Compatible Group 5	1 1 0 1 X	0 1 1 1 X

Figure 3. Example of Dividing Test Cubes into *B*-Compatible Groups

		Set A	Set B
POC Set	*B*-Compatible Group 1	1 1 0 0 X	0 1 0 1 0
		1 0 X 1 1	0 1 0 1 0
		0 1 1 0 X	0 1 0 1 0
		1 1 X 0 X	0 1 0 1 0
	B-Compatible Group 2	0 1 0 1 0	1 0 1 1 1
		1 X 0 0 X	1 0 1 1 1
	B-Compatible Group 3	X 1 1 0 X	0 0 1 0 1
		0 1 0 X X	0 0 1 0 1
FOC Set	*B*-Compatible Group 4	X 1 0 1 X	1 0 0 0 X
	B-Compatible Group 5	1 1 0 1 X	0 1 1 1 X

Figure 4. Specifying Test Cubes in *B*-Compatible Groups of POC Set for Example in Fig. 3

Fault simulation is done for the test vectors in the FOC set and all of the detected faults are removed from the fault list. Fault simulation with the reduced fault list is then done for the POC set assuming full observability. The faults that are detected by the POC set assuming full observability will be referred to as $F(POC)_{full\_observe}$. Fault simulation with the faults in $F(POC)_{full\_observe}$ is then done for the POC set assuming observability only at the outputs that are captured in the scan elements in set *A*. The faults that are detected by the POC set assuming partial observability will be referred to as $F(POC)_{partial\_observe}$. The faults that are contained in $F(POC)_{full\_observe}$, but not in $F(POC)_{partial\_observe}$, are the faults that would go undetected if *disable* was used when applying the candidate test vectors in the POC set. In order to detect these faults, we need to do one of two things:

Approach 1: Move some of the scan elements in set *B* to set *A*, so they will be observed when *disable* is activated to detect the faults.

Approach 2: Keep set *B* as it is, but move some of the candidate test vectors in the POC set to the FOC set so they will be applied with full observability to detect the faults.

4.1 Approach 1 (Move Scan Elements from Set *B* to Set *A*)

Because the response of the test vectors in the POC set are only observed at the scan elements in set *A*, some faults may not be detected. Approach 1 ensures detection of those faults by moving a sufficient set of scan elements from set *B* to set *A* so that those scan elements can be observed when *disable* is activated and hence enable detection of all the faults. To determine which of the scan elements in set *B* need to be moved to set *A* to achieve complete fault coverage, fault simulation with the faults in { $F(POC)_{full\_observe}$ - $F(POC)_{partial\_observe}$ } is done for the POC set assuming full observability. For the first test vector that detects each fault, the faulty response is compared with the fault-free response to determine which scan elements the fault effect is propagated to. An *observability matrix* is formed where each row corresponds to a fault in { $F(POC)_{full\_observe}$ - $F(POC)_{partial\_observe}$ } and each column correspond to a scan element in set *B*. Each entry in the matrix is set to a '1' if the fault corresponding to the row can be propagated to the scan element corresponding to the column, and is set to a '0' otherwise. The minimum set of columns that cover all of the rows of the *observability matrix* gives the minimum set of scan elements that need to be moved from set *B* to set *A* to detect all the faults, where a column *c* is said to cover a row *r* if the matrix entry *(r,c)* is a '1'. Finding the minimum column cover of a matrix is an NP-complete problem, however, some very efficient heuristic algorithms exist [Coudert 99].

4.2 Approach 2 (Move Test Vectors from POC Set to FOC set)

Instead of moving scan elements from set *B* to set *A*, an alternative approach is to move test vectors from the POC set to the FOC set to detect the faults. In this approach, the size of set *B* is not reduced, but rather a sufficient set of test vectors is moved from the POC set to the FOC set so that they will be applied with *disable* deactivated and hence can be fully observed and allow detection of the faults. To determine which of the candidate test vectors in the POC set must be moved to the FOC set to achieve complete fault coverage, fault simulation with the faults in { $F(POC)_{full\_observe}$ - $F(POC)_{partial\_observe}$ } is done for the POC set assuming full observability. Any test vector in the POC set that detects one of these faults is moved to the FOC set. This ensures that all faults will be detected when the remaining test vectors in the POC set are applied with *disable* activated, and the test vectors in the FOC set are applied with *disable* deactivated.

4.3 Ordering Test Vectors

After using either of the two approaches described above to obtain the POC set and FOC set, the last step is to order the test vectors in a way that allows maximal use of *disable*. The test vectors in each *B*-compatible group are ordered sequentially with the ones in the POC set coming before the ones in the FOC set. If all of the test vectors in a *B*-compatible group are in the POC set, then the last one in the sequence is moved to the FOC set since *disable* cannot be used for this vector because the subsequent test vector is not *B*-compatible with it.

When the test set is applied, *disable* is activated when any of the test vectors in the final POC set are applied, and it is deactivated when any of the test vectors in the final FOC set are applied.

5. Experimental Results

Experiments were performed on the largest ISCAS-89 benchmark circuits using both of the test generation approaches described in Sec. 4. It was assumed that the primary inputs and primary outputs of each circuit where part of a core boundary scan chain (i.e., "core wrapper") and hence were not considered when partitioning the scan elements into set *A* and set *B*. Only the flip-flops in the benchmark circuits were considered as candidates for scan disable.

The results for Approach 1 where scan elements are moved from set *B* to set *A* to provide 100% fault coverage of detectable faults are shown in Table 1. For each circuit, the total number of scan elements is shown, and then results are given for using the proposed approach and for using the conventional approach followed by the percentage reduction in the number of logic and clock transitions at the flip-flops. For the proposed approach, the number of flip-flops in Set *B* is shown, followed by the total number of test vectors in the test set, and the number of test vectors for which *disable* can be activated. The total number of flip-flop output transitions that occur when applying all the test vectors with the proposed approach is shown, followed by the total number of flip-flop clock transitions that occur. These two transition counts give an approximate measure of the power that will be dissipated during test. The more flip-flop output transitions that occur, the more gates in the circuit that will switch, and the more flip-flop clock transitions that occur, the more power that is dissipated in the clock tree. The conventional approach involves doing normal ATPG with random fill of all the X's, and then applying all test vectors with no clock disable taking place. The number of test vectors and the number of flip-flop and clock transitions that occur when applying the test are shown. As can be seen, the total number of test vectors for the proposed approach is usually slightly larger than that for the conventional approach. The reason for this is that in the proposed approach, for the test cubes in the POC set, the X's for the scan elements in set *B* are not filled randomly. This results in fewer test vectors being dropped during reverse fault simulation than the conventional approach where all the X's are filled randomly. However, as can be seen, the effect of this slight reduction in randomness results in only a small increase in the number of test vectors. The additional transitions that occur due to the extra vectors are more than offset by the reduction from the scan disable. It is interesting to note that for circuit *s15850*, the number of test vectors was actually smaller for the proposed method. This means that the constrained fill was actually better for detecting faults than the random fill for that particular circuit.

Table 1. Results for Approach 1 for Generating Test Vectors for Scan Disable

Circuit		Proposed Approach 1					Conventional Approach			Power Reduction	
Name	Scan Elem.	FFs in Set B	Total Test Vect.	Vectors w/ Scan Disable	FF Output Transitions	FF Clock Transitions	Test Vectors	FF Output Transitions	FF Clock Transitions	FF Output Transitions	FF Clock Transitions
s9234	228	81	296	227	2924877	6847782	289	3530986	7594920	17.2%	9.8%
s13207	638	313	329	288	19231964	38770626	320	33024885	65911040	41.8%	41.2%
s15850	534	217	233	181	12865496	23729679	244	17221721	34947144	25.3%	32.1%
s38417	1636	658	364	307	156173476	373214771	348	205628135	471860160	24.1%	20.9%

Table 2. Results for Approach 2 for Generating Test Vectors for Scan Disable

Circuit		Proposed Approach 2					Conventional Approach			Power Reduction	
Name	Scan Elem.	FFs in Set B	Total Test Vect.	Vectors w/ Scan Disable	FF Output Transitions	FF Clock Transitions	Test Vectors	FF Output Transitions	FF Clock Transitions	FF Output Transitions	FF Clock Transitions
s9234	228	102	296	42	3337584	7317914	289	3530986	7594920	5.5%	3.6%
s13207	638	354	329	190	22022240	44003431	320	33024885	65911040	33.3%	33.2%
s15850	534	249	233	89	13699310	27853569	244	17221721	34947144	20.5%	20.3%
S38417	1636	724	364	182	170455613	398679024	348	205628135	471860160	17.1%	15.5%

The last two columns show the power reduction in terms of percentage reduction in the number of flip-flop output transitions and flip-flop clock transitions of the proposed approach compared to the conventional approach. As can be seen, the proposed approach provides a substantial reduction in switching activity in the circuit, and hence reduction in power. Both logic power and clock power are reduced substantially.

The results for Approach 2 where test vectors are moved from the POC set to the FOC set to provide 100% fault coverage of detectable faults are shown in Table 2. The number of flip-flops in the *B* set is larger compared with Approach 1, but the number of vectors applied with scan disable goes down.

We also did some experiments on the Motorola V3 ColdFire^{TM} core using commercial ATPG tools. This core is a full-scan design with 32 scan chains. We took the already existing partition of scan elements into scan chains and simply divided the 32 scan chains into set *A* and set *B*. Due to tool limitations, we were not able to generate test cubes, we could only generate fully specified test vectors. This greatly reduced the effectiveness of our technique, but nonetheless, we were still able to get up to a 16% reduction in power dissipation. The results are shown in Table 3. The conventional approach required 597 vectors. We tried placing 8 and 16 out of the 32 scan chains in set *B*.

Table 3. Results for Motorola V3 ColdFire^{TM} Core with 32 Scan Chains and 597 Test Vectors

Proposed Approach				**Power Red.**
Scan Chains in Set B	FFs in Set B	Total Test Vectors	Vectors with Scan Disable	FF Clock Transitions
8	1184	697	577	7.5%
16	2392	753	506	16%

6. Conclusions

In comparing the proposed approach with other approaches, it should be noted that the proposed approach reduces both logic power and clock power. In circuits where clock power is a significant component of the total power, the proposed approach can provide a lot of benefit. As clock frequencies continue to rise, techniques such as the one presented here will be needed.

Acknowledgements

This material is based on work supported in part by the National Science Foundation under Grant No. MIP-9702236 and in part by the Texas Advanced Technology Program under Grant No. 003658-0644-1999.

References

[Brayton 87] Brayton, R.K., R. Rudell, A. Sangiovanni-Vincentelli, and A.R. Wang, "Multi-Level Logic Optimization and the Rectangular Covering Problem," *Proc. of Int. Conf. on CAD*, pp. 66-69, 1987.

[Chen 92] Chen P-C., Liu B-D. and Wang, J-F., "Overall Consideration of Scan Design and Test Generation," *Proc. International Conference on Computer-Aided Design (ICCAD)*, pp. 9-12, 1992.

[Chou 94] Chou, R.M., K.K.Saluja, and V.D. Agrawal, "Power Constraint Scheduling of Tests," *Proc. of International Conference on VLSI Design*, pp. 271-274, 1994.

[Crouch 99] Crouch, A. *Design-For-Test for Digital IC's and Embedded Core Systems*, Prentice Hall, ISBN 0-13-084827-1.

[Coudert 96] Coudert, O., "On Solving Covering Problems," *Proc. Design Automation Conference*, pp. 197-202, 1996.

[Dabholkar 98] Dabholkar, V., S. Chakravarty, I. Pomeranz, and S.M. Reddy, "Techniques for Minimizing Power Dissipation in Scan and Combinational Circuits During Test Application," *IEEE Trans. on Computer-Aided Design*, Vol. 17, No. 12, pp. 1325-1333, Dec. 1998.

[Gerstendörfer 99] Gerstendörfer, S., and H.-J. Wunderlich, "Minimized Power Consumption for Scan-Based BIST," *Proc. of International Test Conference*, pp. 77-84, 1999.

[Girard 99] Girard, P., L. Guiller, C. Landrault, and S. Pravossoudovitch "A Test Vector Inhibiting Technique for Low Energy BIST Design," *Proc. of VLSI Test Symposium*, pp. 407-412, 1999.

[Girard 00] Girard, P., L. Guiller, C. Landrault, and S. Pravossoudovitch "Low Power BIST by Hypergraph Partitioning: Methodology and Architectures," *Proc. of Int. Test Conference*, pp. 652-661, 2000.

[Goel 81] Goel, P., "An Implicit Enumeration Algorithm to Generate Tests for Combinational Logic Circuits," *IEEE Trans. on Computers*, pp. 215-222, Mar. 1981.

[Gupta 91] Gupta, R. and M. Breuer, "Ordering Storage Elements in Single Scan Chain," *Proc. Int. Conf. Comp. Aided Design*, pp. 408-411, 1991.

[Hertwig 98] Hertwig, A., and H.-J. Wunderlich, "Low Power Serial Built-In Self-Test," *Proc. of European Test Workshop*, pp. 49-53, 1998.

[Higami 94] Higami, Y., S. Kajihara and K. Kinoshita, "Reduced Scan Shift: A New Testing Method for Sequential Circuits," *Proc. Int. Test Conf.*, pp. 624-630, 1994.

[Lai 93] Lai, W-J., C-P. Kung and C-S. Lin, "Test Time Reduction in Scan Designed Circuits," *Proc. Euro. Design Automation Conf.*, pp. 489-493, 1993.

[Pouya 00] Pouya, B., and A. Crouch, "Optimization Trade-offs for Vector Volume and Test Power," *Proc. of Int. Test Conf.*, pp. 873-881, 2000.

[Narayanan 92] Narayanan, S., R. Gupta and M. Breuer, "Configuring Multiple Scan Chains for Minimum Test Time," *Proc. Int. Conf. Comp. Aided Design*, pp. 4-8, 1992.

[Sankaralingam 00] Sankaralingam, R., R.R. Oruganti, and N.A. Touba, "Static Compaction Techniques to Control Scan Vector Power Dissipation", *Proc. of VLSI Test Symposium*, pp. 35-40, 2000.

[Su 93] Su, C., and K. Hwang, "A Serial Scan Test Vector Compression Methodology," *Proc. Int. Test Conf.*, pp. 981-988, 1993.

[Wang 94] Wang, S., and S.K. Gupta, "ATPG for Heat Dissipation Minimization during Test Application," *Proc. of International Test Conference*, pp. 250-258, 1994.

[Wang 97a] Wang, S., and S.K. Gupta, "DS-LFSR: A New BIST TPG for Low Heat Dissipation," *Proc. of Int. Test Conf.*, pp. 848-857, 1997.

[Wang 97b] Wang, S., and S.K. Gupta, "ATPG for Heat Dissipation Minimization for Scan Testing," *Proc. of Design Automation Conference*, pp. 614-619, 1997.

[Wang 99] Wang, S., and S.K. Gupta, "LT-RTPG: A New Test-Per-Scan BIST TPG for Low Heat Dissipation," *Proc. of International Test Conference*, pp. 85-94, 1999.

[Whetsel 00] Whetsel, L., "Adapting Scan Architectures for Low Power Operation," *Proc. Int. Test Conf.*, pp. 863-872, 2000.

[Zorian 93] Zorian, Y., "A Distributed BIST Control Scheme for Complex VLSI Devices," *Proc. VLSI Test Symp.*, pp. 4-9, 1993.