Scan Latch Partitioning into Multiple Scan Chains for Power Minimization in Full Scan Sequential Circuits

Nicola Nicolici and Bashir M. Al-Hashimi
Electronic Systems Design Group
Department of Electronics and Computer Science
University of Southampton
Southampton SO17 1BJ, UK
{nn99r,bmah}@ecs.soton.ac.uk

Abstract

Power dissipated during test application is substantially higher than power dissipated during functional operation [22] which can decrease the reliability and lead to yield loss. This paper presents a new technique for power minimization during test application in full scan sequential circuits. The technique is based on classifying scan latches into compatible, incompatible and independent scan latches. Based on their classification scan latches are partitioned into multiple scan chains. A new test application strategy which applies an extra test vector to primary inputs while shifting out test responses for each scan chain, minimizes power dissipation by eliminating the spurious transitions which occur in the combinational part of the circuit. Unlike previous approaches [9] which are test vector and scan latch order dependent and hence are not able to handle large circuits due to the complexity of the design space, this paper shows that with low test area and test data overhead substantial savings in power dissipation during test application are achieved in very low computational time. For example, in the case of benchmark circuit s15850 it takes < 3600*s in computational time and* $< 1\%$ *in test area and test data overhead to achieve* 80% *savings in power dissipation.*

1. Introduction

Minimization of power dissipation in VLSI circuits is important to improve the reliability and reduce packaging costs. While many techniques have investigated power minimization during the normal (functional) mode of operation [2, 4, 8, 12, 13, 17–19], it is essential to examine the power dissipation during the test mode of operation mainly for the following two reasons. Firstly it was outlined in [22] that power dissipated during test application is substantially higher than power dissipated during functional operation which can decrease the reliability of the circuit under test due to higher temperature and current density. Secondly the excessive power/ground noise caused by the high rate of current flowing in power and ground lines can erroneously change the logic state of circuit lines causing some good dies to fall the test [21] leading to yield loss. Depending on level of abstraction and circuit type, high power dissipation during test application is due to the following:

a. Systems which comprise modern memory systems and multichip modules (MCMs) employ power-conscious architectural decisions where blocks are not simultaneously activated under functional operation [7]. Hence, inactive blocks do not contribute to power dissipation during the functional operation. However, when the system is in the test mode of operation, concurrent execution of tests in many blocks will result in substantially higher power dissipation when compared to functional operation.

b. Low power combinational circuits are synthesized by algorithms [2, 12, 17, 18] which seek to optimize the signal or transition probability of circuit nodes using only the spatial dependencies inside the circuit assuming the transition probabilities of primary inputs to be given. However, the complex spatiotemporal correlations which occur at the primary inputs must be considered [16]. This is of further importance during test application since correlation between consecutive test vectors generated by an automatic test pattern generator (ATPG) is very low, because a test vector is generated for a given target fault without any consideration of the previous test vector in the test sequence. The low correlation between consecutive test vectors during test application leads to substantially higher power

dissipation when compared to functional operation.

c. Low power sequential circuits are synthesized by state assignment algorithms which use state transition probabilities [4, 8, 17, 19]. The state transition probabilities are computed assuming input probability distribution and state transition graph which is valid during functional operation. These two assumptions are not valid during the test mode of operation when scan design for testability (DFT) technique is employed. While shifting out test responses, the scan latches are assigned uncorrelated values that destroy the correlation between successive states. Furthermore, in the case of data path circuits with large number of states that are synthesized for low power using the correlations between data transfers [13], in the test mode scan registers are assigned uncorrelated values which are never reached during functional operation leading to substantially higher power dissipation.

To overcome the problem of high power dissipation during test application at the system level (problem a), a power-constrained test scheduling algorithm has been proposed for high performance memories and multichip modules [7]. The algorithm is based on a resource graph formulation for the test problem and tests are scheduled concurrently without exceeding their power ratings during test application. A new ATPG tool [21] was proposed to overcome the low correlation between consecutive test vectors during test application in combinational circuits (problem b). Despite achieving the objectives of safe and inexpensive testing of low power circuits the approach in [21] increased the test application time. A different approach for minimizing power dissipation during test application in combinational circuits (problem b) is based on test vector ordering [9]. Test vector ordering is done in a post-ATPG phase with no overhead in test application time since test vectors are reordered such that correlation between consecutive test vectors matches the assumed transition probabilities of primary inputs used for switching activity computation during low power logic synthesis. However the computational time is very high due to the complexity of test vector ordering problem which is reduced to finding a minimum cost hamiltonian path in a complete, undirected, and weighted graph. To minimize power dissipation in scan sequential circuits during test application (problem c) two techniques have been proposed [9, 10]. In [10], systems equipped with a scan-based built-in self-test like the STUMPS architecture [3] are analyzed, the modules and modes with the highest power dissipation are identified, and gating logic to reduce power dissipation has been proposed. Despite substantial savings in power dissipation gating logic introduces not only suplimentary area overhead but also performance degradation. A technique which does not introduce performance degradation has been proposed in [9]. The technique based on test vector and scan latch ordering increases the correlation between consecutive states during shifting in present state part of the test vector and shifting out test responses. Further benefit of the technique proposed in [9] is that minimization of power dissipation during test application has been achieved without any increase in test application time unlike [7, 21].

However, the technique proposed in [9] is test vector and scan latch order dependent and cannot significantly reduce power dissipation despite a large computational time required to explore the large design space. Furthermore, for circuits with large number of scan latches the technique proposed in [9] is infeasible since computational time required to compute the cost function of each solution in the large design space, is unacceptably large. The aim of this paper is to introduce a new technique for power minimization during test application in full scan sequential circuits which eliminates the computational overhead associated with test vector and scan latch ordering [9]. The technique is based on partitioning scan latches into multiple scan chains and applying an extra test vector to primary inputs while shifting out test responses for each scan chain. This paper shows that with low test area and test data overhead high savings in power dissipation during test application in large full scan sequential circuits are achieved in low computational time.

2. Power Dissipation Model

Power dissipation in digital CMOS circuits is divided into static and dynamic power. The static power is considered negligible when compared to the dynamic power in digital CMOS circuits [6]. If the gate is part of a synchronous digital circuit controlled by global clock, it follows that the dynamic power dissipation P_d is calculated using:

$$P_d = 0.5 \times C_{load} \times (V_{DD}^2 / T_{cyc}) \times N_G \quad (1)$$

where C_{load} is the load capacitance, V_{DD} is the supply voltage, T_{cyc} is the global clock cycle, and N_G is the total number of gate output transitions ($0 \rightarrow 1$ or $1 \rightarrow 0$). Since supply voltage V_{DD} and global clock cycle T_{cyc} are design constraints, they are not under designer control. Thus, *node transition count*

$$NTC = \sum_{for\ all\ gates\ G} N_G \times C_{load} \quad (2)$$

is reported as quantitative measure for power dissipation throughout the paper. It has been assumed that load capacitance for each combinational gate is equal to the number of fan-outs. The node transition count in scan latches N_{SL} is considered as in [9], where it was shown that for input changes $0 \rightarrow 0$ and $1 \rightarrow 1$, $N_{SL_{min}} = 2$, whilst for input changes $0 \rightarrow 1$ and $1 \rightarrow 0$, $N_{SL_{max}} = 6$.

3. Power Minimization in Full Scan Sequential Circuits Based on Multiple Scan Chains

In this section a new technique for power minimization in full scan sequential circuits based on multiple scan chains is introduced. Section 3.1 overviews the proposed design for testability (DFT) architecture for power minimization. Section 3.2 defines compatible, incompatible and independent scan latches and their importance for partitioning scan latches into multiple scan chains, as described in section 4, is explained through examples.

3.1 Proposed Design for Testability Architecture Using Multiple Scan Chains

The proposed DFT architecture using multiple scan chains $SC_0 \ldots SC_{k-1}$ is illustrated in Figure 1. The scan input *ScanIn* is routed to all scan chains while the scan output *ScanOut* is selected from the output of each scan chain. Scan chains are operated using non overlapping clock signals which gate the scan clock using a shift register with number of flip flops equal to the number of scan chains (for the sake of simplicity the shift register and the selector logic are not presented in Figure 1). While shifting out test responses present in scan latches from scan chain SC_i, primary inputs are set to extra test vector EV_i which eliminates the spurious transitions (Definition 1 from section 3.2) that originate from scan latches from scan chain SC_i. The algorithm for partitioning scan latches into multiple scan chains is described in section 4.1, while the new test application strategy using multiple scan chains and extra test vectors is described later in section 4.2. Before describing generation of multiple scan chains the next section classifies scan latches into three broad classes.

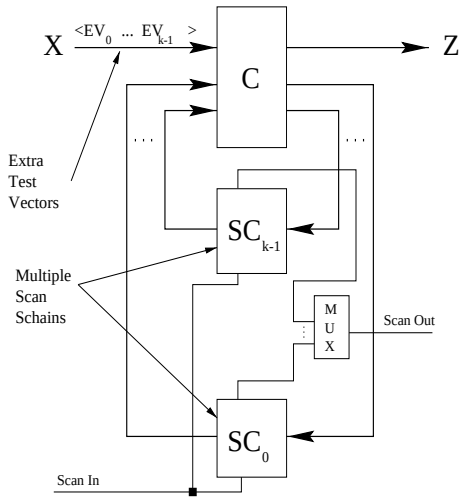

Figure 1. Proposed DFT architecture based on multiple scan chains.

3.2 Compatible, Incompatible and Independent Scan Latches

In order to partition scan latches into multiple scan chains, they need to be classified into three broad classes: compatible, incompatible and independent scan latches. This classification is also important for computing extra test vectors associated with each scan chain that eliminate spurious transitions which are defined as follows.

Definition 1 A spurious transition during test application in scan sequential circuits is a transition which occurs in the combinational part of the circuit under test while shifting out the test response and shifting in the present state part of the next test vector. This transitions do not have any influence on test efficiency since the value at the input and output of the combinational part is not useful test data.

Having defined the spurious transitions, now the compatible and incompatible scan latches are introduced.

Definition 2 Two scan latches S_i and S_j are *compatible* if all primary inputs x_k are assigned values c_k that eliminate the spurious transitions which originate from both S_i and S_j. The values c_k of primary inputs x_k constitute the *extra test vector* which eliminates spurious transitions originating from both S_i and S_j.

Definition 3 Two scan latches S_i and S_j are *incompatible* if at least one primary input x_k that is assigned value i_k to eliminate the spurious transitions which originate from S_i will propagate the transitions which originate from S_j. Two incompatible scan latches cannot be assigned to the same scan chain since there is no extra test vector that can eliminate spurious transitions which originate from both of them.

The following example illustrates compatible and incompatible scan latches.

Example 1 Consider the simple circuit of Figure 2. The $\{x_0, x_1, x_2\}$ are primary inputs, $\{S_0, S_1, S_2, S_3, S_4, S_5\}$ are scan latches, $\{y_0, y_1, y_2, y_3, y_4, y_5\}$ are present state lines, and $\{z_0, z_1, z_2, z_3, z_4, z_5\}$ are circuit outputs. To eliminate spurious transitions at gate z_0 while shifting out test responses through scan latch S_0, primary input x_0 must be assigned the controlling value 0 of gate z_0. Similarly, to eliminate spurious that originate from scan latch S_1, primary input x_0 must be assigned the controlling value 1 of

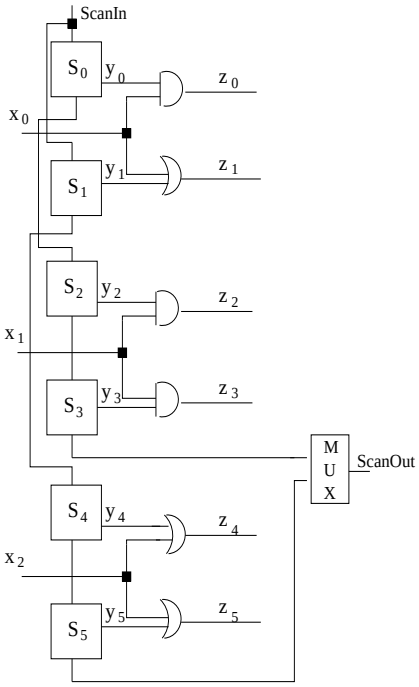

Figure 2. Example circuit illustrating compatible and incompatible scan latches.

gate z_1. Different values must be assigned to x_0 to eliminate spurios transitions which originate from scan latches S_0 and S_1. Therefore scan latches S_0 and S_1 are incompatible and are assigned to different scan chains $SC_0 = \{S_0\}$ and $SC_1 = \{S_1\}$. On the other hand, by assigning x_1 to the controlling value 0 of gates z_2 and z_3 the spurious transitions which originate from both scan latches S_2 and S_3 are eliminated. Thus, by introducing S_2 and S_3 into SC_0 and applying extra test vector $x_0x_1x_2 = \{00X\}$ while shifting out test responses from $SC_0 = \{S_0, S_2, S_3\}$ no spurious transitions will occur at gates z_0, z_2 and z_3. Similarly, scan latches S_4 and S_5 are compatible since assigning 1 to the primary input x_2 eliminates spurious transitions at gates z_4 and z_5. By introducing S_4 and S_5 into SC_1 and applying extra test vector $x_0x_1x_2 = \{1X1\}$ while shifting out test responses from $SC_1 = \{S_1, S_4, S_5\}$ no spurious transitions will occur at gates z_1, z_4 and z_5.

The previous example has assumed a simple circuit where *all* the spurious transitions are eliminated by partitioning scan latches in two scan chains SC_0 and SC_1. However, some of the spurious transitions cannot be eliminated as described in the following example.

Example 2 Consider the circuit shown in Figure 3. The spurious transitions which originate in scan latches S_0 and

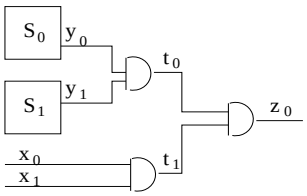

Figure 3. Example circuit illustrating spurious transitions which cannot be eliminated.

S_1 cannot be eliminated at gate t_0 since both inputs are present state lines. However, by assigning x_0 and/or x_1 to the controlling value 0 of gate t_1 the spurious transitions will be eliminated at gate t_1. Scan latches S_0 and S_1 are compatible since same primary input values eliminate the spurious transitions of gate t_1.

Example 2 has illustrated that some of the spurious transitions cannot be eliminated since all the gate inputs depend on present state lines. Computing primary input values that eliminate spurious transitions (extra test vectors introduced in Definition 2) can be viewed as an ATPG problem to a *reduced circuit* with a *specified fault list* which is divided into three steps. In the first step, *freezing signals* are identified which are the signals that depend on primary inputs and should be set to the controlling value as side inputs to the gates which eliminate transitions that originate from scan latches. In the second step, the circuit is modified to a reduced circuit which is generated as follows: gates and signals that depend only on scan latches are excluded; gates that depend on both scan latches and primary inputs are modified to gates with input signals dependent only on primary inputs (in the case of gates with two inputs of which one is a freezing signal the gate is modified to a buffer); all the freezing signals identified in the first step are set as primary outputs in the reduced circuit. Also the freezing signals are introduced in the specified fault list targeting the stuck at the non controlling value of the gate which eliminates the spurious transitions in the initial circuit. In the third step, having generated the reduced circuit and the specified fault list, any ATPG tool can be used to generate extra test vectors. The following example illustrates the three steps required to compute extra test vectors.

Example 3 For the circuit shown in Figure 3 the reduced circuit is generated as follows. In the first step, the freezing signal t_1 at the input of gate z_0 is identified to eliminate spurious transitions that originate from scan latches S_0 and S_1. In the second step, scan latches S_0 and S_1, and the AND gate t_0 are excluded from the reduced circuit as shown in Figure 4. Furthermore, gate z_0 is modified to a buffer (signals t_1 and z_0 are identical). The targeted fault

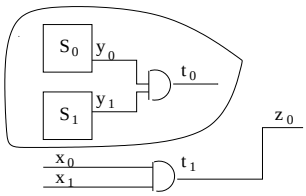

Figure 4. Reduced circuit of the example circuit from Figure 3 illustrating the three steps required to compute extra test vectors.

in the reduced circuit is t_1 $sa-0$ which eliminates the spurious transitions at gate z_0 in the original circuit. Finally in the third step, the extra test vectors (Definition 2) that eliminate the spurious transitions during test application are computed $x_0x_1 = \{0X, X0\}$.

Finally, independent scan latches are introduced.

Definition 4 A scan latches S_i is *independent* if all the gates on all the paths which originate from S_i do not have at least one side input which can be justified by primary inputs.

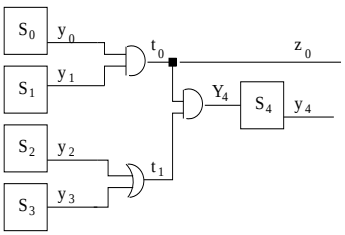

Figure 5. Example circuit illustrating independent scan latches.

The independent scan latches are grouped in the extra scan chain (ESC) for which no extra test vector can be computed and hence the spurious transitions cannot eliminated. The following example illustrates independent scan latches.

Example 4 Consider the circuit shown in Figure 5. Output z_0 depends only on scan latches S_0 and S_1, and the next state Y_4 of scan latch S_4 depends only on scan latches S_0, S_1, S_2 and S_3. There are no side inputs of gates t_0 and t_1 that can be justified by primary inputs such that spurious transitions originated from S_0, S_1, S_2 and S_3 are eliminated. Therefore scan latches S_0, S_1, S_2 and S_3 are independent.

4. Multiple Scan Chains Generation and New Test Application Strategy

In this section, partitioning of scan latches in multiple scan chains based on their classification, as described in 3.2, is given. Then, a new test application strategy for power minimization during test application, based on the DFT architecture described in section 3.1, is introduced.

4.1 Partitioning Scan Latches into Multiple Scan Chains

Multiple Scan Chain Partitioning (*MSC-PARTITIONING*) algorithm identifies compatible scan latches, groups them in scan chains and computes an extra test vector for each scan chain. Figure 6 gives the pseudocode of the proposed *MSC-PARTITIONING* algorithm. Initially, the freezing signals (section 3.2) $\{FS_0, \ldots, FS_{p-1}\}$ are determined simultaneously with identifying independent scan latches (Definition 4). The independent scan latches are grouped into the extra scan chain (ESC) which consists of scan latches whose spurious transitions cannot be eliminated by computing an extra test vector. Having determined the freezing signals, the specified fault list (section 3.2) is created as shown in line 3. The next line shows how the reduced circuit C' is generated as described in section 3.2. An ATPG tool [14] is used to generate extra test vectors for the modified circuit C' and to identify compatible faults from the specified fault list L. If an extra test vector detects at least two compatible faults on different freezing signals, then the scan latches, whose spurious transitions are eliminated by setting the controlling value on the respective freezing signals, are compatible (Definition 2) and are grouped in the same scan chain as shown in line 6. If the reduced circuit is redundant then the scan latches, whose spurious transitions cannot be eliminated by the freezing signals with redundant faults, are introduced in the extra scan chain ESC. The algorithm returns the scan chains of compatible scan latches, the extra scan chain ESC and the extra test set of extra test vectors used to define a new test application strategy, as explained in the following section.

4.2 New Test Application Strategy Using Multiple Scan Chains and Extra Test Vectors

Having partitioned the scan latches into multiple scan chains with an extra test vector for each scan chain (section 4.1), this section introduces a new test application strategy for power minimization during test application in full scan sequential circuits. Multiple Scan Chain Test Application (*MSC-TEST APPLICATION*) algorithm computes the node transition count *NTC* (section 2) during the entire test application period for the given test set **S**, circuit **C**, multi-

ALGORITHM: **MSC-PARTITIONING**
INPUT: Circuit **C**
OUTPUT:Scan Chains $\{\mathbf{SC_0}, \ldots, \mathbf{SC_{k-1}}, \mathbf{ESC}\}$
Extra Test Set **ES** = $\{EV_0, \ldots, EV_{k-1}\}$

1 identify the freezing signals $\{FS_0, \ldots, FS_{p-1}\}$ where each freezing signal FS_i stops the spurious transitions from scan latch list $\{S_{i_0}, \ldots, S_{i_{m-1}}\}$
2 $\{S_{e_0}, \ldots, S_{e_{m-1}}\}$ for which no freezing signal exists are introduced in the extra scan chain **ESC**
3 create the fault list $L = \{FS_0\ sa-c_0, \ldots, FS_{p-1}\ sa-c_{p-1}\}$ where c_i is the non-controlling value of the gate G_a frozen by FS_i
4 modify the circuit **C** to **C'** as following:
- ▶ gates dependent only on scan latches are excluded
- ▶ gates dependent on both scan latches and primary inputs are modified to gates with input signals dependent only on primary inputs
- ▶ freezing signals are set as primary outputs

5 perform ATPG for fault list L of **C'** and generate extra test set **ES** = $\{EV_0, \ldots, EV_{k-1}\}$
6 $\{S_{j_0}, \ldots, S_{j_{m-1}}\}$ of detected faults $FS_j\ sa-c_j$ by extra test vector EV_i are introduced in SC_i
7 $\{S_{r_0}, \ldots, S_{r_{m-1}}\}$ of redundant faults $FS_r\ sa-c_r$ are introduced in the extra scan chain **ESC**
8 **return** $\{\mathbf{SC_0}, \ldots, \mathbf{SC_{k-1}}, \mathbf{ESC}\}$
ES = $\{EV_0, \ldots, EV_{k-1}\}$

Figure 6. Proposed algorithm for partitioning scan latches in multiple scan chains

ple scan chains $\{\mathbf{SC_0}, \ldots, \mathbf{SC_{k-1}}, \mathbf{ESC}\}$, and extra test set **ES** = $\{EV_0, \ldots, EV_{k-1}\}$. Figure 7 gives the pseudocode of the proposed *MSC-TEST APPLICATION* algorithm. The value of **NTC** is 0 at the beginning of the algorithm and it is gradually increased as the entire test set is traversed. The outer loop represents the traversal of all the test vectors V_i, with $i = 0, \ldots, n-1$, from test set **S**. Shiting out test responses through all the scan chains are then considered in the inner loop. For each scan chain SC_j, circuit **C** is simulated by applying the extra test vector EV_j to primary inputs and $NTC_{i,j}$ is added to the node transition count **NTC**. $NTC_{i,j}$ stands for node transition count while shfting in present state part of test vector V_i through scan chain SC_j and applying extra test vector EV_j to the primary inputs. After shitfting out the test responses though each scan chain SC_j the primary input part of test vector V_i is applied to primary inputs and $NTC_{i,ESC}$ is computed while shifting out test response through the extra scan chain ESC. Finally the entire test vector V_i is applied to the circuit under test and $NTC_{i,LOAD}$ required to load the test response in the scan latches, is added to **NTC**. After the completion of the inner loop, the outer loop continues until the entire test set is examined. The algorithm returns the value of **NTC** over the entire test application period. It should be noted that algorithms presented in this section are independent of test vector and scan latch order. Unlike the algorithms from [9] which require high computational time for small circuits and which are unfeasible for large circuits due the size of the design space ($n! \times m!$ where n is the number of test vectors and m is the number of scan latches), the proposed *MSC-PARTITIONING* and *MSC-TEST APPLICATION* algorithms have low computational time and can handle large circuits as shown in the following section.

ALGORITHM: **MSC-TEST APPLICATION**
INPUT: Test Set **S**=$\{V_0, \ldots, V_{n-1}\}$, Circuit **C**
Scan Chains $\{\mathbf{SC_0}, \ldots, \mathbf{SC_{k-1}}, \mathbf{ESC}\}$
Extra Test Set **ES** = $\{EV_0, \ldots, EV_{k-1}\}$
OUTPUT: Node transition count **NTC**

1 **NTC**$\leftarrow$0
2 **for** every V_i from **S** with $i = 0, \ldots, n-1$ {
3 **for** every SC_j with $j = 0, \ldots, k-1$ {
4 apply EV_j to primary inputs
5 compute $NTC_{i,j}$ by simulating **C** when shifting in the present state part of test vector V_i through scan latches from SC_j
6 **NTC**$\leftarrow$**NTC** + $NTC_{i,j}$
7 }
8 apply primary part of V_i to primary inputs
9 compute $NTC_{i,ESC}$ by simulating **C** when shifting in the present state part of test vector V_i through scan latches from **ESC**
10 **NTC**$\leftarrow$**NTC** + $NTC_{i,ESC}$
11 **NTC**$\leftarrow$**NTC** + $NTC_{i,LOAD}$
12 }
13 **return NTC**

Figure 7. Proposed test application strategy using multiple scan chains and extra test vectors

5. Experimental Results

This section demonstrates through a set of benchmark examples that multiple scan chains combined with extra test vectors, as outlined in section 3, yield savings in power dissipation during test application. The algorithms described in section 4 have been implemented on a 350 MHz Pentium II PC with 64 MB RAM running Linux and using GNU CC version 2.7. The average value of node transition count (*NTC*) reported throughout this section is calculated using the formulas from section 2 under the assumption of the zero delay model. However, the proposed technique applies equally to other delay models as unit [15] and variable delay model [11]. Furthermore, due to elimination of spurious transitions (Definition 1) the propagation of hazards and glitches is eliminated leading to even greater reductions for power dissipation in the case of unit and variable delay model. Besides, the aim of this paper is not to give exact values of power dissipation during test application, but to define a new design for testability architecture and a new test application strategy for power minimization that applies equally to every delay model.

circuit	TV	SC	ESC length	initial *NTC*	final *NTC*	CPU time (s)
s208	34	2	0	54.54	37.71	1
s298	33	3	5	103.56	45.85	1
s344	24	3	0	130.36	55.70	1
s349	22	3	0	131.90	60.30	1
s382	32	3	6	133.91	53.73	1
s400	33	3	6	135.97	58.07	1
s420	73	2	0	111.69	65.99	1
s444	33	3	6	139.92	59.21	1
s526	60	3	5	170.61	75.11	1
s641	58	2	0	166.32	73.11	1
s713	58	2	0	173.34	81.31	1
s838	148	2	0	227.46	125.91	2
s953	90	6	23	158.50	84.30	2
s1196	140	3	2	101.31	76.45	3
s1238	151	3	2	101.50	73.69	3
s1423	70	4	3	453.58	174.07	6
s5378	259	8	18	1772.07	284.95	148
s9234	366	5	8	3160.16	819.16	694
s13207	461	6	318	5949.81	2007.11	3798
s15850	436	7	53	5260.90	1030.36	3587

Table 1. Experimental results using multiple scan chains for power minimization.

Table 1 shows the experimental results for 20 circuits from ISCAS89 benchmark set [5]. The first and second columns give the circuit name and the number of test vectors (TV) respectively generated using the ATALANTA test tool [14]. The third and fourth columns give the number of scan chains (SC) and the length of the extra scan chain (ESC) respectively computed using the *MSC-PARTITIONING* algorithm outlined in section 4.1. Column 5 shows the initial average value of *NTC*, which is the total value of *NTC* divided by the total number of clock cycles over the entire test application period. The next column 6 shows the average value of *NTC* when using multiple scan chains and extra test vectors (*MSC-TEST APPLICATION* algorithm from section 4.2). It can be clearly seen that the proposed test application strategy has significantly smaller average value of *NTC* for all the benchmark circuits when compared to initial value of *NTC* computed using the test application strategy from [1]. Furthermore, the computational time is very low ($< 3s$) for small circuits. Moreover, for large circuits which are not handled by previous approaches [9], as in the case of *s13207*, it takes $< 4000s$ to achieve substantial reduction in average value of *NTC*.

circuit	power reduction (%)	test area overhead (%)	test data overhead (%)
s208	30.81	7.39	3.26
s298	55.72	7.29	1.06
s344	57.26	7.20	4.68
s349	54.27	7.16	5.11
s382	59.87	5.34	0.78
s400	57.28	5.22	0.75
s420	40.91	3.58	1.45
s444	57.68	5.05	0.75
s526	55.97	4.34	0.41
s641	56.04	2.71	2.23
s713	53.08	2.57	2.23
s838	44.64	1.76	0.69
s953	46.81	6.71	1.97
s1196	24.54	3.03	0.62
s1238	27.40	2.94	0.57
s1423	61.62	2.14	0.80
s5378	83.91	1.50	0.44
s9234	74.07	0.61	0.15
s13207	66.26	0.36	0.09
s15850	80.41	0.43	0.17

Table 2. Power reduction and overhead in test area and test data.

To give an indication of the reductions in power dissipation, Table 2 shows the percentage reduction in power dissipation (column 1) and percentage overhead in test area and test data (columns 2 and 3). The power dissipation is considered directly proportional to the average value of *NTC*. The test area overhead represents the extra logic required to multiplex the scan output signal (Figure 1). The test data overhead represents the number of extra bits required for the extra test vectors (the number of scan chains multiplied by the number of primary inputs). The power reduction varies from approximately 84% as in the case of *s5378* down to under 25% as in the case of *s1196*. For small circuits the test area and test data overhead are up to 7% and 5% respectively as in the case of *s349*. For large circuits, as in the

case of *s15850* it takes $< 1\%$ in test area and test data overhead to achieve 80% savings in power dissipation, which clearly shows the advantage of the proposed technique for power minimization using multiple scan chains.

It should be noted that experimental results reported in this section using the simplified power model from section 2 do not consider power dissipated by the clock tree which can be up to 20% of the total power dissipation [10]. However, the power dissipated by the clock tree can be substantially reduced using low power buffered clock tree design [20] which succesfully handles both scan clock gating and scan clock trees required by the proposed design for testability architecture using multiple scan chains (Figure 1 from section 3.1).

6. Conclusions

This paper has presented a new technique for power minimization during test application in full scan sequential circuits which overcomes the computational overhead of the previous approaches [9] and substantially reduces power dissipation for large circuits. The technique is based on classifying scan latches into compatible, incompatible and independent scan latches. Based on their classification scan latches are partitioned into multiple scan chains. A new test application strategy which applies an extra test vector to primary inputs while shifting out test responses for each scan chain, minimizes power dissipation by eliminating the spurious transitions which occur in the combinational part of the circuit. For example, in the case of benchmark circuit *s15850* it takes < 3600s in computational time and $< 1\%$ in test area and test data overhead to achieve 80% savings in power dissipation which leads to higher yield and reliability.

Acknowledgement

The authors wish to acknowledge Dr. Dong S. Ha of Virginia Polytechnic Institute and State University for providing ATALANTA [14] test tool.

References

[1] M. Abramovici, M. Breuer, and A. Friedman. *Digital Systems Testing and Testable Design*. IEEE Press, 1990.

[2] R. Bahar, H. Cho, G. Hachtel, E. Macii, and F. Somenzi. Symbolic timing analysis and resynthesis for low power of combinational circuits containing false paths. *IEEE Transactions on CAD*, 16(10):1101–1115, Oct 1997.

[3] P. Bardell, W. McAnney, and J. Savir. *Built-In Self Test - Pseudorandom Techniques*. John Wiley & Sons, 1986.

[4] L. Benini and G. D. Micheli. State assignment for low power dissipation. *IEEE Journal of Solid-State Circuits*, 30(3):258–268, Mar 1995.

[5] F. Brglez, D. Bryan, and K. Kozminski. Combinational profiles of sequential benchmark circuits. In *Proc. International Symposium on Circuits and Systems*, pages 1929–1934, 1989.

[6] A. Chandrakasan and R. Brodersen. *Low Power Digital CMOS Design*. Kluwer Academic Publishers, 1995.

[7] R. Chou, K. Saluja, and V. Agrawal. Scheduling tests for VLSI systems under power constraints. *IEEE Transactions on VLSI*, 5(2):175–184, Jun 1997.

[8] S.-H. Chow, Y.-C. Ho, T. Hwang, and C. Liu. Low power realization of finite state machines - a decomposition approach. *ACM Transactions on Design Automation of Electronic Systems*, 1(3):313–340, Jul 1996.

[9] V. Dabholkar, S. Chakravarty, I. Pomeranz, and S. Reddy. Techniques for minimizing power dissipation in scan and combinational circuits during test application. *IEEE Transactions on CAD*, 17(12):1325–1333, Dec 1998.

[10] S. Gerstendorfer and H. Wunderlich. Minimized power consumption for scan-based BIST. In *Proc. IEEE International Test Conference*, pages 77–84, 1999.

[11] M. Hsiao, E. Rudnick, and J. Patel. Effects of delay models on peak power estimation of VLSI sequential circuits. In *Proc. International Conference on Computer Aided Design*, pages 45–51, 1997.

[12] S. Iman and M. Pedram. An approach for multi-level logic optimization targeting low power. *IEEE Transactions on CAD*, 15(8):889–901, Aug 1996.

[13] G. Lakshminarayana, A. Raghunathan, N. Jha, and S. Dey. Power management in high level synthesis. *IEEE Transactions on VLSI*, 7(1):7–15, Mar 1999.

[14] H. K. Lee and D. S. Ha. On the generation of test patterns for combinational circuits. Technical Report No. 12-93, Department of Electrical Engineering, Virginia Polytechnic Institute and State University, 1991.

[15] S. Manich and J. Figueras. Maximizing the weighted switching activity in CMOS combinational circuits under the variable delay model. In *Proc. European Design and Test Conference*, pages 597–602, 1997.

[16] R. Marculescu, D. Marculescu, and M. Pedram. Probabilistic modelling of dependecies during switching activity analysis. *IEEE Transactions on CAD*, 17(2):73–83, Feb 1998.

[17] K. Roy and S. Prasad. Circuit activity based CMOS logic synthesis for low power reliable operations. *IEEE Transactions on VLSI*, 1(4):503–513, Dec 1993.

[18] J.-M. Tseng and J.-Y. Jon. Two-level logic minimization for low power. *ACM Transactions on Design Automation of Electronic Systems*, 4(1):52–69, Jan 1999.

[19] C.-Y. Tsui, M. Pedram, and A. Despain. Low-power state assignment targeting two- and multilevel logic implementations. *IEEE Transactions on CAD*, 17(2):1281–1291, Dec 1998.

[20] A. Vital and M. Marek-Sadowska. Low-power buffered clock tree design. *IEEE Transactions on CAD*, 16(9):965–975, Sep 1997.

[21] S. Wang and S. Gupta. ATPG for heat dissipation minimization during test application. *IEEE Transactions on Computers*, 47(2):256–262, Feb 1998.

[22] Y. Zorian. A distributed BIST control scheme for complex VLSI devices. In *Proc. 11th IEEE VLSI Test Symposium*, pages 4–9, 1993.