

# Graph Theoretic Approach for Scan Cell Reordering to Minimize Peak Shift Power

Jaynarayan T Tudu  
Indian Institute of Science  
Bangalore, India  
jayttudu@csa.iisc.ernet.in

Erik Larsson  
Linkoping University  
Linkoping, Sweden  
erila@ida.liu.se

Virendra Singh  
Indian Institute of Science  
Bangalore, India  
viren@serc.iisc.ernet.in

Hideo Fujiwara  
Nara Institute of Science and  
Technology  
Nara, Japan  
fujiwara@is.naist.jp

## ABSTRACT

Scan circuit testing generally causes excessive switching activity compared to normal circuit operation. This excessive switching activity causes high peak and average power consumption. Higher peak power causes, supply voltage droop and excessive heat dissipation. This paper proposes a scan cell reordering methodology to minimize the peak power consumption during scan shift operation. The proposed methodology first formulate the problem as graph theoretic problem then solve it by a linear time heuristic. The experimental results show that the methodology is able to reduce up to 48% of peak power in compared to the solution provided by industrial tool.

## Categories and Subject Descriptors

B.7.3 [Reliability and Testing]: Low power testing

## General Terms

Algorithm

## Keywords

Peak power, Power droop, Scan chain reordering

## 1. INTRODUCTION

A scan chain is a set of sequential elements, which during test mode are connected to each other in a serial fashion. In scan testing, the test patterns are loaded into the scan chain by serial shifting. During the time of shifting, a number of transitions occur in the scan chain. These transitions propagate and create excessive toggling in the combinational

logic, which lead to high dynamic power consumption. The toggling activity that takes place during scan operation is in general much higher than the toggling during functional operation [18]. The excessive toggling during testing is the major concern for the modern circuit.

As most modern designs in the current era of deep sub-micron are functionally complex and operate at high speed, the power consumption becomes an important issue to address particularly during testing. Excessive average power can cause problems such as instant circuit damage, higher product cost, performance degradation and reduced battery life [13]; and excess peak power can result in IR drop and cross talk problems, leading to a good chip being falsely classified as defective. The IR drop and cross talk problems are of increasing concern for at-speed testing. During at-speed launch and capture operation, the excessive peak power causes high rate of current in the power and ground rails and hence leads to excessive noise at power and ground. This excessive noise can change the state of logic unexpectedly; hence, this may cause a good die to fail the test, which leads to yield loss [8]. Saxena et al. [3] discusses the IR drop and cross talk problems in detail.

Reduction of peak power during test becomes necessary for three important reasons: 1. high peak power causes yield loss due to power droop and cross talk, 2. to avoid chip being damaged during testing, and 3. for an SoC with multiple module, a parallel testing can be performed to reduce the test time.

The main contribution of this paper is to provide a graph theoretic problem formulation for scan cells reordering that takes power consumption during load/unload cycle and shift & launch cycle(also known as test cycle) into account. It also provides a heuristic to solve the formulated problem. The experimental results show that peak power can be significantly reduced.

The rest of the paper is organized as follows. Section II gives an overview of previous works. In Section III, a brief background to peak power problem and basic of scan reordering is presented. Section IV and V provides the problem formulation and algorithm respectively. In section VI, time and space complexity for the proposed algorithm is analyzed. Section VII presents the experimental results and the paper is concluded in section VIII.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

GLSVLSI'10, May 16–18, 2010, Providence, Rhode Island, USA.  
Copyright 2010 ACM 978-1-4503-0012-4/10/06 ...\$10.00.

## 2. PREVIOUS WORK

The problem of test power reduction has been an active area of research for quite sometime. The most straight forward approach to reduce the dynamic power is to test the circuit at reduced clock speed. However this solution is no longer a practical as modern designs are timing critical and sensitive to test application time. More over the peak power is independent of clock frequency.

Most of the previous approaches are aimed at average power reduction, however they have also achieved some reduction in peak power as a by-product [1, 11, 9, 16, 14, 17]. The approach by Chou et al. [1] proposed the scheduling of test under power constraint, address power at module level, and not at circuit level. The work by Dabholkar et al. [11] has achieved average power reduction by test vector and scan latch reordering using some of the TSP heuristics. The work in the area of pattern compaction technique to control the average as well as peak power is done by Sankaralingam et al. [9]. The idea is to carefully merge the test patterns such that the power consumption in the resultant patterns is minimized. However, the scan reordering can still be applied on power aware merged patterns to further minimize the power consumption. An technique based on scan partition called as adaptive scan architecture was proposed by [16]. Another work on average power reduction is done by Girard et al. [14] and Bonhomme et al. [12]. The basic idea behind these works is to reorder the scan cells. In both of these proposals, the problems are formulated as a graph theoretic problems. As the problems are NP-complete the solutions are heuristic based. Furthermore the proposal by Girard [14] also provides a power-area tradeoff. Wu et al. [17] proposed a graph theoretic formulation of scan cells reordering for average power reduction. In this work, the Xs are preserved during the time of scan cells reordering for further optimization using MT-fill. This work also reduces peak power consumption along with average power as by-product.

Butler et al. [5] and Li et al.[15] proposed X-filling technique, specifying the don't care bits in the test patterns, to minimize average power. Although the X-filling heuristic for average power also able to minimize peak power as by-product, the scan reordering can still be applied for further reduction.

As discussed in the previous section, peak power reduction is becoming increasingly important. A few works have been proposed in peak power reduction [10, 7, 6, 8, 2]. Sankaralingam et al. [10] first identify the peak power violation patterns, and then perform simulation to identify the Xs in those patterns. This kind of methodology generally involves the simulation process which is time consuming for large circuits. More over the scan reordering can still be used along with this kind of X-fill approach. The work by Badereddine et al. [7] provides two possible approaches: scan chain stitching and X-filling technique to deal with the peak power. Badereddine et al. [6] evaluate the peak power reduction using MT-fill. Corno et al. [2] also address the peak power problem. In this work X-bits are identified by simulation and specified for peak power reduction. A recent work reported by Tudu et al. [4] uses scan vector reordering to minimize peak power. However our approach of scan chain reordering can be applied along with such techniques. Hence further reduction can be achieved.

The work done by Badereddine et al. [8] is based on scan reordering to minimize peak power during test cycle. In this work, the problem is formulated as a simulated annealing problem. As the primary concern is peak power during test cycle, they have considered only test patterns and not the responses which limits their approach to reduce peak power during load/unload cycles. However, their approach could achieve nominal reduction in peak power during load/unload cycles as a by-product.

Given the above observation, in this paper we are proposing a methodology to explore the concept of scan cells reordering to minimize peak power during load/unload cycle and test cycle. The proposed methodology does not affect the stuck-at fault coverage and test time. However incur nominal routing area overhead and may affect the transition fault coverage.

## 3. BACKGROUND

### 3.1 Overview of Peak Power During Scan Operation

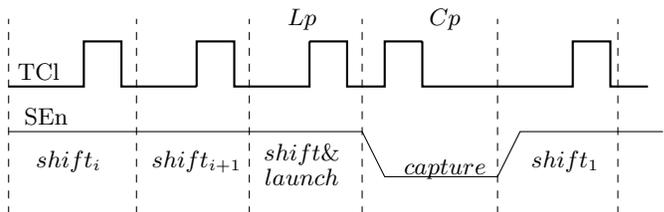


Figure 1: Timing digram for at-speed scan testing

In the above figure, TCl is test clock, SE<sub>n</sub> is scan enable, L<sub>p</sub> is launch pulse, C<sub>p</sub> is capture pulse and shift<sub>i</sub> is the *i*<sup>th</sup> shift cycle.

The power consumed during entire scan operation at various clock cycle are, *shift power* during the shift cycles (which also include the launch pulse L<sub>p</sub>) and *capture power* during the capture pulse C<sub>p</sub>. The shift power at each cycle is known as instantaneous power and the maximum among all the instantaneous power (for all vectors) is known as peak power. The peak power during shift cycle causes excessive heat dissipation (when the peak power remain for several number of cycle) and the peak power during L<sub>p</sub> causes the IR drop problem which is especially problematic for at-speed testing [13], [3]. This paper addresses both of the problems.

### 3.2 Basis of Power Reduction by Scan Cell Reordering

The scan cell reordering mechanism used for peak and average power minimization basically rely on reduction of *intra pattern transitions* (the bit difference within a pattern). By looking at the differing bits in the patterns, each scan cell can be reconnected in a suitable order to reduce the *intra pattern transitions*. The reordering of scan cells brings all 1s close to each other as well as all 0s during the scan shift operation. Figure 2 demonstrates the basic principle of scan cell reordering. In this figure T is test and R is response.

	Initial ordering of scan chain				Reordered scan chain			
	SF <sub>1</sub> → SF <sub>2</sub> → SF <sub>3</sub> → SF <sub>4</sub>				SF <sub>1</sub> → SF <sub>3</sub> → SF <sub>4</sub> → SF <sub>2</sub>			
T	0	1	0	1	0	0	1	1
R	1	0	1	1	1	1	1	0

Figure 2: Basic principle of scan reordering

## 4. PROBLEM FORMULATION

In this section a graph theoretic problem is formulated by first constructing a graph from given scan related information and then a problem resembling to scan cells ordering is defined on that graph. The following subsections explains the complete procedure.

### 4.1 Construction of Graph

The important information that are needed to be reflected in the graph are, scan cells, possible scan connection and peak power information for any pair of consecutive scan cells. Below is the procedure for graph construction:-

- For every scan cell SF<sub>i</sub> there is a corresponding node N<sub>i</sub> in the graph.
- The scan path between any two possible scan cells is represented by an undirected edge between respective nodes. *Note:* The edges are undirected because, the pairing of scan cells is symmetric with respect to the number of transition. For example, in Figure 2 a revers ordering of the reordered scan chain will also results into one transition.
- The instantaneous power consumed by any test vector and response vector due to any possible pair of scan cells is represented as a pair of weight named as *weight-pair* of the edge between respective nodes of the scan cells. The first element of the *weight-pair* is the weight corresponds to test vector and the second element corresponds to the response vector. Each element in the *weight-pair* are vector quantity, we name these element as *testvector-weight* (TVW) and *responsevector-weight* (RVW). The TVW and RVW are computed as follows:

Computation of TVW and RVW: Each TVW and RVW keeps the instantaneous power information for all tests and responses respectively. Instantaneous power information for an individual pattern (test or response) is computed by finding the bit difference between the original position of scan cell pair.

Table 2 shows the TVWs and RVWs for the tests and responses shown in Table 1 respectively. The rows in Table 1 are tests T<sub>i</sub> and their corresponding response R<sub>i</sub> and columns are scan flip flops SF<sub>i</sub>. In Table 2 the scan cell pairs

Table 1: Scan patterns and Original scan order

	→SF <sub>1</sub>	→SF <sub>2</sub>	→SF <sub>3</sub>	→SF <sub>4</sub>
T <sub>1</sub>	1	0	1	0
T <sub>2</sub>	0	1	0	1
T <sub>3</sub>	1	0	1	0
R <sub>1</sub>	1	0	1	1
R <sub>2</sub>	0	1	0	1
R <sub>3</sub>	1	0	0	0

Table 2: Computed TVW and RVW

scan cell pair [SF <sub>i</sub> , SF <sub>j</sub> ]	TVW			RVW		
	t <sub>1</sub>	t <sub>2</sub>	t <sub>3</sub>	r <sub>1</sub>	r <sub>2</sub>	r <sub>3</sub>
[SF <sub>1</sub> , SF <sub>2</sub> ]	1	1	1	1	1	1
[SF <sub>2</sub> , SF <sub>3</sub> ]	1	1	1	1	1	0
[SF <sub>3</sub> , SF <sub>4</sub> ]	1	1	1	0	1	0
[SF <sub>4</sub> , SF <sub>1</sub> ]	1	1	1	0	1	1
[SF <sub>1</sub> , SF <sub>3</sub> ]	0	0	0	0	0	1
[SF <sub>4</sub> , SF <sub>2</sub> ]	0	0	0	1	0	0

are given in column labeled with “scan cell pair” and corresponding TVW and RVW are given in columns t<sub>1</sub> to t<sub>3</sub> and r<sub>1</sub> to r<sub>3</sub> respectively. The columns t<sub>1</sub> to t<sub>3</sub> and r<sub>1</sub> to r<sub>3</sub> shows the peak power information for each test T<sub>i</sub> and response R<sub>i</sub>. The rectangular box in Table 1 shows the bit difference due to scan pair SF<sub>1</sub> and SF<sub>2</sub> and the corresponding transition information in Table 2 is shown inside a square box in t<sub>1</sub> minor column. Figure 3 shows the complete graph constructed from the data given in Table 1 & Table 2. In the process of computation of *weight-pair*, we have considered tests as well as responses. Taking both tests and responses into account is necessary to capture the peak power information as tests and responses both are shifted during load/unload operation.

### 4.2 Graph Theoretic Problem Formulation

The basic objective of problem formulation is that the formulation should consider the ordering of scan cells and the minimization of peak power.

The instantaneous power consumed by scan patterns during scan shift operation due to scan cell pair [SF<sub>i</sub>, SF<sub>j</sub>] is represented as the *weight-pair* of the *edge*(N<sub>i</sub>, N<sub>j</sub>) in the graph. For example, the instantaneous power consumed by scan pair [SF<sub>4</sub>, SF<sub>2</sub>] is represented as *weight-pair* ([0 0 0],[1 0 0]) shown in Table 2. As the graph is complete graph, it is always possible to find an order of scan cells by constructing a *Hamiltonian path* (which visits each node of the graph exactly once). Each Hamiltonian path in the graph has a corresponding cost, *cost*(N<sub>s</sub>, N<sub>e</sub>), of constructing the path N<sub>s</sub>~N<sub>e</sub>, where N<sub>s</sub> and N<sub>e</sub> are start and end nodes in the path. The computation of cost function *cost*(N<sub>s</sub>, N<sub>e</sub>) for a Hamiltonian path (let say H<sub>p</sub>) is carried out in following manner:

- Sum up all the TVWs along the path H<sub>p</sub> in vector addition manner and find out the maximum value in summation result.
- Similarly sum all the RVWs along path H<sub>p</sub> and find out the maximum value in the summation result.

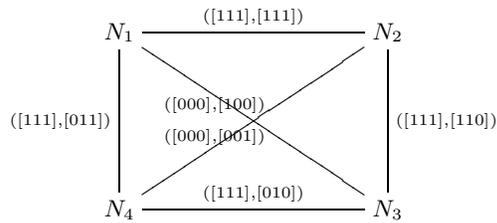


Figure 3: A complete vector-weighted graph

- The maximum of above computed maxima will be the  $cost(N_s, N_e)$  for path  $N_s \rightsquigarrow N_e$  which resembles the peak power value for corresponding scan order  $S_s \rightsquigarrow S_e$ .

In a more formal way,

$$PeakPower = cost(N_s, N_e) = Max(T_{max}, R_{max}) \quad (1)$$

where  $T_{max} = Max\left(\sum_{i=1}^{l-1} TVW(n_i, n_{i+1})\right)$  and

$R_{max} = Max\left(\sum_{i=1}^{l-1} RVW(n_i, n_{i+1})\right)$ , where  $n_i$  and  $n_{i+1}$  are

the  $i^{th}$  and  $(i+1)^{st}$  visited nodes in the path  $N_s \rightsquigarrow N_e$ , and  $l$  is the length of scan chain(or total number of node). The following example illustrates the above described procedure.

**Example 1** Let  $N_1 \xrightarrow{([111],[011])} N_4 \xrightarrow{([000],[001])} N_2 \xrightarrow{([111],[110])} N_3$ , be a Hamiltonian path from Figure 3. Then

the  $T_{max}$  and  $R_{max}$  can be computed as follow:

$$T_{max} = Max(TVW(n_1, n_2) + TVW(n_2, n_3) + TVW(n_3, n_4)) = Max([1 \ 1 \ 1] + [0 \ 0 \ 0] + [1 \ 1 \ 1]) = Max([2 \ 2 \ 2]) = 2.$$

$$R_{max} = Max(RVW(n_1, n_2) + RVW(n_2, n_3) + RVW(n_3, n_4)) = Max([0 \ 1 \ 1] + [0 \ 0 \ 1] + [1 \ 1 \ 0]) = Max([1 \ 2 \ 2]) = 2.$$

In this example  $n_1 = N_1$ ,  $n_2 = N_4$ ,  $n_3 = N_2$ , and  $n_4 = N_3$ . Finally the  $cost(N_s, N_e) = Max(T_{max}, R_{max}) = 2$ .

Hence from the above illustration the graph theoretic problem can formally be stated in following way:

**Problem Statement:** Given an undirected graph  $G(V, E, W)$ , find a Hamiltonian path which has minimum cost over all other possible paths. Where  $V$  is set of vertices  $V_i$ s,  $E$  is set of edges, and  $W$  is set of *weight-pairs*.

*Note:* The hardness of problem is NP-complete as this can be reduced to TSP which is a known NP-complete problem.

## 5. PROPOSED ALGORITHM

As the problem is NP-complete we are proposing a greedy based polynomial time (with respect to  $|V|$ ) algorithm. The complete algorithm has two parts, Part-1 to find the Hamiltonian cycle and Part-2 to find the path from Hamiltonian cycle. Following subsequent subsection gives the details of algorithm for Part-1 and Part-2.

### 5.1 Construction of Hamiltonian cycle

**Problem Statement:** Given a graph  $G(V, E, W)$  find a Hamiltonian cycle having minimum cost  $cost(N_s, N_e)$ , in this case  $s = e$ .

**Algorithm Part-1**(with reference to Figure 3):

**Step 1:** Chose an initial node randomly and assign it to *current-node* and initialize corresponding *cumulative-node-cost* to  $([0 \ 0 \ 0], [0 \ 0 \ 0])$ .

*Note:* *cumulative-node-cost* is an array of integers computed dynamically for a node visited currently.

**Step 2:** In this step, the decision parameter *peak-power* and *average-power* are computed. These are used in Step 3 to select the next node to visit. The *peak-power* and *average-power* are computed with respect to the set of nodes which are not yet visited and are neighbors of the current node. We call these node as *candidate-node* and any edge from

current node to a *candidate-node* is called as *candidate-edge*. A vector addition operation is performed between the *cumulative-node-cost* and *weight-pair* of each *candidate-edge* separately. We call these sum *probe-path-weight*. The *peak-power* for each of the *candidate-edge* is computed by finding  $Max(\text{probe-path-weight})$  along the *candidate-edge*. And *average-power* is computed by finding  $Sum(\text{probe-path-weight})$  along the *candidate-edge*.

**Step 3:** A node  $N_i$  from the set of *candidate-node* is selected as the next node if the *peak-power* along the *edge(current-node,  $N_i$ )* is minimum among all other *peak-power*. If there are more than one node having equal *peak-power* then the corresponding *average-power* is used as a tie breaker. In case of equal average power, the next node will be chosen randomly. Following example helps understanding the above procedure.

**Example 2** Considering Figure 3, the initial node will be  $N_1$  with *cumulative-node-cost*  $([0 \ 0 \ 0], [0 \ 0 \ 0])$ . The neighboring node of  $N_1$  are  $N_2$ ,  $N_3$ , and  $N_4$  which satisfies the property of being *candidate-node* because these are not yet visited. Based on the *peak-power* and *average-power*, a next node will be selected from these *candidate-nodes*. The corresponding *candidate-edges* are *edge( $N_1, N_2$ )*, *edge( $N_1, N_3$ )*, and *edge( $N_1, N_4$ )*. The *weight-pair* of each of these *candidate-edges* are  $([1 \ 1 \ 1], [1 \ 1 \ 1])$ ,  $([0 \ 0 \ 0], [0 \ 0 \ 1])$ , and  $([1 \ 1 \ 1], [0 \ 1 \ 1])$  respectively. *Probe-path-weight* along each of the *candidate-edges* will be vector addition of *cumulative-node-cost* of current node with the *weight-pair* of each of these *candidate-edge* which are,  $([0 \ 0 \ 0], [0 \ 0 \ 0]) + ([1 \ 1 \ 1], [1 \ 1 \ 1]) = ([1 \ 1 \ 1], [1 \ 1 \ 1])$ ,  $([0 \ 0 \ 0], [0 \ 0 \ 0]) + ([0 \ 0 \ 0], [0 \ 0 \ 1]) = ([0 \ 0 \ 0], [0 \ 0 \ 1])$ , and  $([0 \ 0 \ 0], [0 \ 0 \ 0]) + ([1 \ 1 \ 1], [0 \ 1 \ 1]) = ([1 \ 1 \ 1], [0 \ 1 \ 1])$  respectively.

The *peak-power* and *average-power* along each *candidate-edge* can now be computed as follows,

*peak-power:*

$$\text{along } edge(N_1, N_2) = Max(Max([1 \ 1 \ 1]), Max([1 \ 1 \ 1])) = 1$$

$$\text{along } edge(N_1, N_3) = Max(Max([0 \ 0 \ 0]), Max([0 \ 0 \ 1])) = 1$$

$$\text{along } edge(N_1, N_4) = Max(Max([1 \ 1 \ 1]), Max([0 \ 1 \ 1])) = 1$$

*average-power:*

$$\text{along } edge(N_1, N_2) = Sum([1 \ 1 \ 1]) + Sum([1 \ 1 \ 1]) = 6$$

$$\text{along } edge(N_1, N_3) = Sum([0 \ 0 \ 0]) + Sum([0 \ 0 \ 1]) = 1$$

$$\text{along } edge(N_1, N_4) = Sum([1 \ 1 \ 1]) + Sum([0 \ 1 \ 1]) = 5$$

The next task is to select a next node based on the above computed *peak-power* and *average-power*. In this example *peak-power* along each *candidate edge* is equal *i.e.* 1, hence *average-power* is used as a tie breaker. *average-power* along *edge( $N_1, N_3$ )* is 1 which is less than the *average-power* of other *candidate edge*. Hence node  $N_3$  will be selected as the next node to be visited.

**Step 4:** In this step *cumulative-node-cost* and *current-node* are updated. The *cumulative-node-cost* for next node to visit = *cumulative-node-cost* of *current-node* + *weight-pair* of *edge(current-node, next-node-to-visit)*. For Example 2, *current-node* will be updated with node  $N_3$  and *cumulative-node-cost* for  $N_3 = ([0 \ 0 \ 0], [0 \ 0 \ 0]) + ([0 \ 0 \ 0], [0 \ 0 \ 1]) = ([0 \ 0 \ 0], [0 \ 0 \ 1])$ .

**Step 5:** Repeat the Algorithm from Step 2 until a Hamiltonian cycle is constructed.

At the end Algorithm part-1 will produce a Hamiltonian cycle. For Figure 3 the Algorithm will give the Hamiltonian cycle shown in Figure 4 as output.

## 5.2 Construction of Hamiltonian path

**Problem Statement:** Given a Hamiltonian cycle  $H(V, E, W)$ , find out a Hamiltonian path which have minimum cost  $cost(N_s, N_e)$ .

**Algorithm Part-2(Computation and Selection):**

**Computation:** In this step the peak and average power consumption will be computed for every possible Hamiltonian path. Peak power is the primary objective of power minimization where as average power is used a tie breaker which is the secondary objective. For a given  $H$  having total node  $|V|$  there can be  $|V|$  number of possible Hamiltonian path. For each path the peak and average power is computed. Computation of peak and average power for a Hamiltonian path  $H_p$  are done as follows:

Peak power is computed according to the Equation 1. Average power is computed according to the weighted transition metric method [9] as follows,

$$AveragePower = T_{sum} + R_{sum} \quad (2)$$

$$where T_{sum} = Sum \left( \sum_{i=1}^{l-1} TVW(n_i, n_{i+1}) * i \right).$$

$$R_{sum} = Sum \left( \sum_{i=1}^{l-1} RVW(n_i, n_{i+1}) * (n - i) \right), \text{ where } l = |V|$$

is the total number of node in  $H$ , and  $n_i$  and  $n_{i+1}$  are the  $i^{th}$  and  $(i + 1)^{st}$  node in  $H_p$ .

**Example 3** Computation of peak and average power for all possible Hamiltonian path in Figure 4.

$$Path1: N_1 \xrightarrow{([111],[111])} N_2 \xrightarrow{([000],[100])} N_4 \xrightarrow{([111],[010])} N_3$$

$$\begin{aligned} PeakPower &= Max(T_{max}, R_{max}) \\ T_{max} &= Max([1 \ 1 \ 1] + [0 \ 0 \ 0] + [1 \ 1 \ 1]) = 2 \\ R_{max} &= Max([1 \ 1 \ 1] + [1 \ 0 \ 0] + [0 \ 1 \ 0]) = 2 \\ PeakPower &= Max(2, 2) = 2 \\ AveragePower &= T_{sum} + R_{sum} \\ T_{sum} &= Sum([1 \ 1 \ 1] + [0 \ 0 \ 0] * 2 + [1 \ 1 \ 1] * 3) = 12 \\ R_{sum} &= Sum([1 \ 1 \ 1] * 3 + [1 \ 0 \ 0] * 2 + [0 \ 1 \ 0]) = 12 \\ AveragePower &= 12 + 12 = 24 \end{aligned}$$

$$Path2: N_2 \xrightarrow{([000],[100])} N_4 \xrightarrow{([111],[010])} N_3 \xrightarrow{([000],[001])} N_1$$

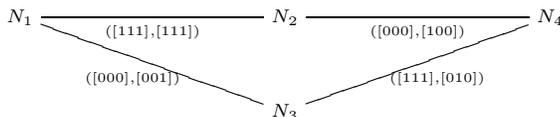
$$\begin{aligned} PeakPower &= Max(T_{max}, R_{max}) \\ T_{max} &= Max([0 \ 0 \ 0] + [1 \ 1 \ 1] + [0 \ 0 \ 0]) = 1 \\ R_{max} &= Max([1 \ 0 \ 0] + [0 \ 1 \ 0] + [0 \ 0 \ 1]) = 1 \\ PeakPower &= Max(1, 1) = 1 \\ AveragePower &= T_{sum} + R_{sum} \\ T_{sum} &= Sum([0 \ 0 \ 0] + [1 \ 1 \ 1] * 2 + [0 \ 0 \ 0] * 3) = 6 \\ R_{sum} &= Sum([1 \ 0 \ 0] * 3 + [0 \ 1 \ 0] * 2 + [0 \ 0 \ 1]) = 6 \\ AveragePower &= 6 + 6 = 12 \end{aligned}$$

$$Path3: N_4 \xrightarrow{([111],[010])} N_3 \xrightarrow{([000],[001])} N_1 \xrightarrow{([111],[111])} N_2$$

$$PeakPower = 2 \text{ and } AveragePower = 20$$

$$Path4: N_3 \xrightarrow{([000],[001])} N_1 \xrightarrow{([111],[111])} N_2 \xrightarrow{([000],[100])} N_4$$

$$PeakPower = 2 \text{ and } AveragePower = 16.$$



**Figure 4: Hamiltonian cycle from Algorithm Part-1**

**Selection:** In this step the optimal Hamiltonian path will be selected based on the following criteria,

- The path which have lesser peak power than other will be the final Hamiltonian path
- In case of multiple path having equal peak power value, the minimum average power path will the final Hamiltonian path
- In case of failure of above criteria, the Hamiltonian path will be chosen with random choice.

For Example 3, the *Path2* will be chosen as the final Hamiltonian path as the peak power of this path is lesser than others. Hence the resultant scan chain which corresponds to Path2 will be  $SF_2 \rightarrow SF_4 \rightarrow SF_3 \rightarrow SF_1$ .

## 6. TIME AND SPACE COMPLEXITY

### 6.1 Time Complexity

**Algorithm Part-1(Hamiltonian cycle):** The proposed heuristic has complexity of  $O(n * l^2)$  where  $l$  is the length of scan chain and  $n$  is number of test vector. The proposed heuristic takes additional  $2n$  time to compute peak and average power at *Step 2* in comparison to any other heuristic of same nature whose edge weight in graph are just a scalar quantity unlike in our case.

**Algorithm Part-2 (Hamiltonian path):** Time complexity for this part of algorithm is  $O(n * l)$ .

### 6.2 Space Complexity

Proposed heuristic does not use explicit space to represent the graph as the weight of edges are computed dynamically. An array of size  $l$  is used to keep the visited information for each node. To store the *cumulative-node-cost* requires  $2n$  size of array. A temporary space to keep the *weight-pair* needs  $2n$  size of array. Hence the overall complexity will be  $O(l + n)$ .

Hence the proposed heuristic does not suffer from any strain related to time and space.

## 7. EXPERIMENTAL RESULTS

The experiments are carried out on ITC99 and ISCAS89 circuits. The algorithm is implemented in C++. Scan insertion, circuit synthesis, and test generation are performed using industrial tools. Test patterns are low power adjacent X-filled and compacted stuck-at fault pattern. Specification of benchmark circuits and experimental results are provided in Table 3 and Table 4 respectively.

**Table 3: Specification of Benchmark Circuit**

Benchmark circuit	#Test pattern	Scan chain length	#PI	#PO	#Gates	Fault coverage
b04	49	66	11	8	628	92.85
b07	53	45	1	8	427	99.95
b08	40	21	9	4	171	100
b10	44	17	11	6	180	100
s420	53	16	18	1	218	99.32
s5378	107	179	35	49	2779	99.87
s9234	123	211	36	39	5597	99.93
s13207	115	638	62	152	7951	99.97
s15850	96	534	77	150	9772	100
s35932	13	1728	35	320	16065	100
s38417	69	1636	28	106	22179	100
s35854	111	1426	38	304	19523	100

**Table 4: Experimental results**

Benchmark circuit	#Peak trans. Ind. soln.	#Peak trans. [12]	%of reduct. by [12] wrt Ind. soln.	#Peak trans. Proposed soln.	%of reduct. wrt Ind. soln.	%of reduct. wrt [12]
b04	33	26	21.21	22	33.33	15.28
b07	27	19	29.63	17	37.04	10.53
b08	16	12	25	11	31.25	8.33
b10	12	11	8.33	10	16.67	9.09
s420	11	8	27.27	7	36.36	12.5
s5378	84	87	-3.57	68	20.24	21.82
s9234	118	99	16.1	89	24.58	11.11
s13207	176	148	15.9	91	48.29	28.51
s15850	108	117	-8.33	97	10.18	17.09
s35932	90	22	75.56	17	81.11	22.73
s38417	470	493	-4.9	364	22.55	26.16
s38584	440	421	4.32	353	19.77	16.15
Avg. Data	132.08	121.92	7.91	95.5	31.79	21.67

To show the effectiveness of the proposed work we have compared the experimental results with the scan order provided by industrial tool and with the average power scan order proposed by Bonhomme et al. [12]. We have chosen [12] as the suitable reference to compare our results reason being this is the only work on scan cell reordering for minimization of average and peak power. Another nearest candidate would be the methodology by Badereddine et al.[8]. However, the careful observation of experimental results of [12] and [8] indicates that for most of the benchmarks the work by Bonhomme et al. [12] is performing better for shift cycle peak power.

Table 4 show that the proposed work is able to reduce the peak power up to 48.29% in comparison to the industrial solution and 28.51% in comparison to [12] for circuit s13207. For circuit s35932 the proposed work and the work by [12] are able to reduce the power significantly due to the favourable combination of smaller number of pattern set and large scan chain. Also the proposed solution has achieved appreciable reduction for larger benchmark circuit. For some bench mark the %of reduction by [12] is negative because the peak power was not considered explicitly where as in this proposed work the peak power is considered as a primary parameter.

## 8. CONCLUSION

This paper has proposed a graph theoretic approach for scan cell reordering to minimize peak power during scan shift operation. The results show that the proposed methodology is capable of minimizing peak power in compared to industrial solution and [12]. The proposed work keeps stuck-at fault coverage and test application time unaffected. The scan reordering may affect the transition fault coverage in case of *skewed-load* testing and may incur small area overhead due to routing. In this work we have not taken these two parameters into consideration. However, the problem of routing can be solved by integrating our work with the methodology proposed by Girard et al. [14], where the proposed methodology can be applied within the cluster to take care of power and cluster ordering can be used to minimize routing congestion. The transition fault coverage requires further work to co-optimize it with power and routing area.

The proposed scan reordering methodology is basically pattern dependent. It is the limitation of this proposed work that if some additional patterns are required on top of the existing patterns then the reordered scan chain may not be

able to reduce the peak power effectively. This issue needs further work.

## 9. REFERENCES

- [1] R. Chou, K. Saluja, and V.D. Agrawal. Scheduling test for vlsi systems under power under power constraints. *IEEE Transactions on VLSI Systems*, 5(2):175 – 185, 1997.
- [2] F. Corno et al. On reducing the peak power consumption of test sequences. In *Proc. European Conf. on Circuit Theory and Design*, pages 247 – 250, 1999.
- [3] J. Saxena et al. A case study of ir-drop in structured in at-speed testing. In *Proc. IEEE Int. Test Conference*, pages 1098 – 1104, September 2003.
- [4] J. Tudu et al. On minimization of peak power during soc test. In *Proc. European Test Symposium*, pages 25 – 30, 2009.
- [5] K.M. Butler et al. Minimizing power consumption in scan testing: Pattern generation and dft techniques. In *Proc. IEEE International Test Conference*, pages 335 – 364, 2004.
- [6] N. Badereddine et al. Minimizing peak power consumption during scan testing: Test pattern modification with x-filling heuristics. In *Proc. of Intl Conf. on Design and Test of Integrated System in Nano Technology*, pages 359 – 364.
- [7] N. Badereddine et al. *Controlling Peak Power During Scan Testing: Power-Aware Dft and Test Set Perspective*. Springer Berlin / Heidelberg, August 2005.
- [8] N. Badereddine et al. Scan cell reordering for peak power reduction during scan test cycles. *IFIP International Federation for Information Processing*, 240:267 – 281, 2007.
- [9] R. Sankaralingam et al. Static compaction technique to control scan vector power dissipation. In *Proc. of VLSI Test Symposium*, pages 35 – 40.
- [10] R. Sankaralingam et al. Controlling peak power during scan testing. In *Proc. IEEE VLSI Test Symposium*, pages 153 – 159, 2002.
- [11] V. Dabholkar et al. Techniques for minimizing power dissipation in scan and combinational circuits during test application. *IEEE Trans. on Comp.-Aided Design*, 17(12):1325 – 1333, December 1998.
- [12] Y. Bonhomme et al. Power driven chaining of flip-flops in scan architectures. In *Proc. of International Test Conference*, pages 796 – 802, 2002.
- [13] P. Girard. Survey of low-power testing of vlsi circuits. *IEEE Design & Test of Computers*, pages 82 – 92, May 2002.
- [14] P. Girard and Y. Bonhomme. Low power scan chain design: A solution for an efficient tradeoff between test power and scan routing. *Journal of Low Power Electronics*, 1.
- [15] J. LI, Q. XU, Yu HU, and X. LI. On reducing both shift and capture power for scan-based testing. In *Proc. Design Automation Conference*, pages 653 – 658, 2008.
- [16] L. Whetsel. Adapting scan architecture for low power scan operation. In *Proceedings International Test Conference*, pages 863–873, 2000.
- [17] Y. Wu and M. Chao. Scan-chain reordering for minimizing scan shift power based on non-specified test cubes. In *Proc. IEEE VLSI Test Symposium*, pages 147 – 174, April-May 2008.
- [18] Y Zorian. A distributed bist control scheme for complex vlsi devices. In *Proceedings of the 11th IEEE VLSI Test Symposium*, pages 4 – 9, 1993.