Constrained Signal Selection for Post-Silicon Validation

Kanad Basu, Prabhat Mishra
Computer and Information Science and Engineering
University of Florida, Gainesville, USA
email: {kbasu, prabhat}@cise.ufl.edu

Priyadarsan Patra
Post-Si Validation Architecture
Intel Corporation, USA
email: priyadarsan.patra@intel.com

***Abstract*—Limited signal observability is a major concern during post-silicon validation. On-chip trace buffers store a small number of signal states every cycle. Existing signal selection techniques are designed to select a set of signals based on the trace buffer width. In a real-life scenario, it is reasonable that a designer has determined some important signals that must be traced. In this paper, we study the constrained signal selection problem where a set of trace signals are already provided by the designer and the remaining signals have to be determined to improve overall restoration performance. Our experimental results using ISCAS'89 benchmarks demonstrate that up to 5% improvement can be obtained in restoration performance compared to existing approaches.**

I. Introduction

Post-silicon validation is widely acknowledged as an important challenge in designing modern microprocessors and System-on-Chip (SoC) designs. Once integrated circuits (ICs) are fabricated, it is not possible to observe each and every internal signal state. On-chip trace buffers help to observe some of the internal signal states. As shown in Figure 1, the activities span across three phases. During design phase, signal selection techniques try to identify a small set of important signals. The number of signals is determined based on the trace buffer width. During execution phase, the signal states are stored in the trace buffer for several cycles. The trace duration is determined based on the depth of the trace buffer. During debug phase, the traced signal states are used to reconstruct some of the untraced signal states.

The primary goal of signal selection is to carefully select the trace signals in order to improve the restorability of the untraced signals. Existing signal selection algorithms ([1], [2], [3], [4]) select a small number of trace signals based on the trace buffer width. However, it may often occur that the designer has determined some signals that are very important (for example, some critical control signals) and hence those signals must be traced. To accommodate this constraint, the modified signal selection problem would be to determine the rest of the trace signals. In other words, if the trace buffer width is w, and n ($n < w$) signals are already provided by the designer, the goal is to select the remaining $w-n$ signals for tracing. In this paper, we study how to extend existing signal selection algorithms to take care of this constraint. An obvious way to accommodate this constraint is to directly apply existing methods to select the rest $w-n$ signals. Our experimental results indicate that it is not beneficial to use existing methods to select the remaining signals. Our approach performs better because it uses the knowledge of the given n signals while selecting the remaining $w-n$ signals such that they collectively provide better restoration performance.

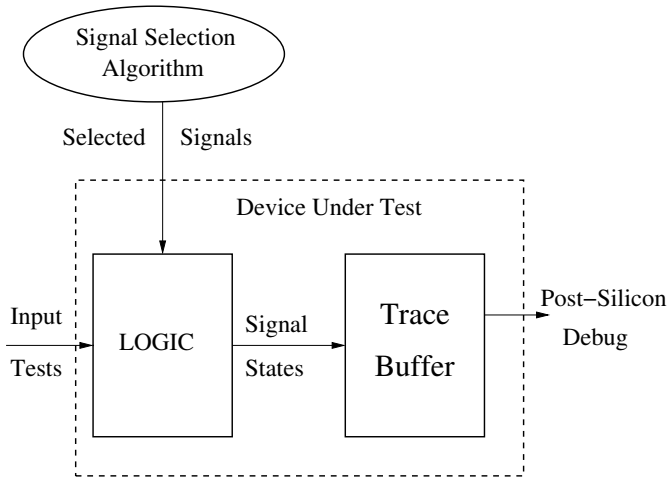

Fig. 1. Overview of post-silicon validation

The rest of the paper is organized as follows. Section II presents related work on signal selection. Section III provides the background on signal selection. Our proposed algorithms and experimental results are described in Section IV and Section V, respectively. Finally, Section VI concludes the paper.

II. Related Work

During post-silicon validation, limited observability of internal signals is a major concern since the chip has already been manufactured. In order to enhance the observability, various design-for-debug techniques like Embedded Logic Analyzer (ELA) [5] and shadow flip flops [6] have been proposed. These techniques depend on an on-chip trace buffer for debugging purposes. Since only a few internal signal states are stored using these methods, they should be carefully selected in order to improve the restoration of the untraced signal states.

Various signal selection techniques [1], [2], [3], [4] have been proposed over the years to improve the observability. None of these techniques assume any pre-determined signal constraints. Various approaches [7], [8], [9] combined trace data with scan chains to improve the observability. Multiplexed or dynamic signal selection approaches were used by [10] and [11]. In this paper, we investigate signal selection problem with the constraint that some of the trace signals are provided by the designer.

III. Background and Motivation

During post-silicon validation, since the trace buffer size is limited, only a small set of signals are traced and the rest has to be reconstructed (or restored) based on the known signal states. Restoration proceeds in two ways - forward and backward. During forward restoration, the output signal states are restored by tracing the input signals. For example, consider the *AND* gate in Figure 2 with two inputs a and b, and the output c. If we trace a and observe that its state to be 0, it is obvious that the state of c is also 0. However, if a is 1, we cannot reconstruct the value of c unless we also know (trace) the value of b. During backward restoration, the input signal states are restored by tracing the outputs. For example, if we trace output c (of the *AND* gate) and note its state to be 1, it is obvious that both a and b are 1. However, if c is 0, we cannot reconstruct the input values, unless we also know (trace) one of the input to be 1.

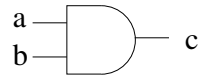

Fig. 2. An example *AND* gate

Restoration ratio is used as a measure of restoration performance. It is defined as:

$$\frac{number\ of\ states\ restored\ +\ number\ of\ states\ traced}{number\ of\ states\ traced}. \quad (1)$$

Obviously, a higher restoration ratio indicates a better restoration performance.

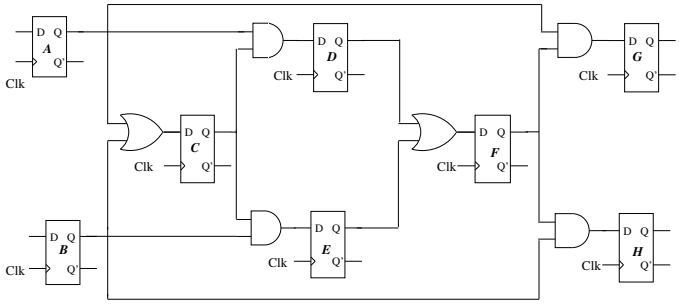

Fig. 3. Example circuit with 8 flip-flops [3]

We use the example circuit in Figure 3 in order to explain signal restoration across a circuit with eight flip-flops and six gates. Let us consider a trace buffer of width 2 (i.e., two signal states can be traced each cycle) and the number of pre-determined signals is 1. In other words, $w = 2$ and $n = 1$. Let us consider the scenario when the designer has indicated that B must be traced every cycle. The algorithm outlined in [3] will select C as the next trace signal.

TABLE I
Restored signals using B and C

Signal	Cycle 1	Cycle 2	Cycle 3	Cycle 4	Cycle 5
A	X	0	X	0	X
B	**1**	**0**	**1**	**0**	**1**
C	**1**	**1**	**0**	**1**	**0**
D	X	X	0	0	0
E	X	1	0	0	0
F	X	X	1	0	0
G	X	X	0	X	0
H	X	X	0	1	0

Table I shows the number of states restored for the untraced signals when B and C are traced over 5 cycles. The rows corresponding to B and C are highlighted (in bold) to indicate the fact that these

values are obtained from trace buffer. The values of the other signals are obtained through restoration. For example, since both B and C are 1 in cycle 1, E would be 1 in cycle 2 (using forward restoration). Note that a 0 or 1 indicates a restored state while an 'X' indicates that the state can not be restored. The number of states restored in this scenario is 27, with a restoration ratio of 2.7.

TABLE II
RESTORED SIGNALS USING A AND B

Signal	Cycle 1	Cycle 2	Cycle 3	Cycle 4	Cycle 5
A	**0**	**0**	**0**	**0**	**1**
B	**1**	**0**	**1**	**0**	**1**
C	X	1	0	1	0
D	X	0	0	0	0
E	X	X	0	0	0
F	X	X	X	0	0
G	X	X	0	0	0
H	X	X	0	X	0

With the knowledge (constraint) that B is being traced every cycle, it is more beneficial to select A (instead of C) as the next trace signal. In Table II, we show the restoration performance when A and B are traced. The number of states restored is 28, producing a restoration ratio of 2.8. This example motivates the need for constrained signal selection for improving overall restoration performance. In Section IV, we propose two signal selection algorithms that efficiently select trace signals utilizing the knowledge of the signals already provided by the designer.

IV. CONSTRAINED SIGNAL SELECTION

We have proposed two algorithms in order to select signals under the constraint that we must trace a set of pre-determined signals. Our first algorithm is an extension of [3]. In [3], the authors used the structural characteristics of the circuit for signal selection. The first signal is selected based on an analysis of the circuit to provide the best possible restoration. A *region* is created using all the adjacent signals of the selected one. In the subsequent runs, the knowledge of the already selected signal as well as the *region* is used to recompute the restoration performance of the other signals in the circuit in order to determine the next best trace signal. This process continues until the trace buffer width is reached. The difference between [3] and our approach (Algorithm 1) is that in [3], the first signal is selected assuming the initial region is empty, whereas Algorithm 1 forms the initial region using the pre-determined signals. The rest of the signals are determined based on the *region* and the circuit structure. It is important to keep track of the *region* size. If the initial *region* comprises of only few signals (since the signals are provided by designers, there is no guarantee on the size of the initial region), it is not useful to select signals inside the *region*; rather, the signal selection algorithm should start afresh. Thus, direct application of [3] can be viewed as a specific instance of Algorithm 1. While [3] will always start selecting signals based on the region information, Algorithm 1 may or may not use the region information (ignores the information if the region is extremely small). We refer Algorithm 1 as Structural Trace Signal Selection algorithm or **StruSS**.

Algorithm 1: Structural Signal Selection (StruSS)

Input: Circuit, Trace buffer width w,
Pre-determined n signals
Output: Trace signals
1: Create a *region* using n pre-determined signals.
2: Find the edge and node values using [3]
while *Trace buffer is not full* **do**
3: Select signal with the highest node value.
4: Recompute region and node values.
end
Return Selected trace signals

Our second signal selection method (Algorithm 2) is an extension of [4]. The basic idea of [4] is to perform simulation for several cycles and then select the signals based on the restoration performance of the signals in that time frame. Initially, all the signals are selected for tracing. In each iteration, one signal from the list is deleted based on its restoration ability as well as the list of already selected signals. This process continues until the trace buffer is full. This approach operates on signal elimination. In order to extend this approach for the constrained signal selection problem, it is necessary to avoid the given signals during the elimination phase. The difference between our approach (Algorithm 2) and [4] is whether the already provided signals are chosen for elimination. Also, the contribution of the other signals in the circuit while restoring the already given signals should not be considered. In other words, if any signal can restore

one of the given signals' states, its contribution should be removed from consideration. This is because, since the signals are already selected for tracing, their states can be observed. Hence, it is not beneficial to consider their restoration when selecting subsequent signals. We refer this approach as Simulation based Trace Signal Selection or **SimuSS**.

Algorithm 2: Simulation-based Trace Signal Selection (SimuSS)

Input: Circuit, Trace buffer width w,
Pre-determined n signals
Output: Trace signals
1: Simulate the circuit for several cycles.
2: Find the restored states for each cycle.
Ignore contribution from pre-determined signals.
while *Trace buffer is not full* **do**
3: Select signal with highest restored states.
4: Recompute restored states for each signal.
Neglect contributions of states that can be restored using already selected signals.
end
Return Selected trace signals

V. Experiments

Our proposed approach *StruSS* (Algorithm 1) is derived from the signal selection technique presented by Basu et al. [3]. We compare the two approaches using 4 largest ISCAS'89 sequential benchmarks. A trace buffer of 32 is used for each approach. In each case, we assume that 16 signals are already provided by the designer. These 16 signals are selected randomly, and the same set is chosen for each of the two approaches in order to enable fair comparison. The number of states restored by these approaches are compared. We assume a trace buffer of depth 100, that is, a total simulation of 100 cycles are performed. The results are shown in Figure 4. As can be seen, up to 5% improvement in restoration performance is obtained.

Similarly, since we have modified the signal selection algorithm proposed by Chatterjee et al. [4], we would like to see how *SimuSS* (Algorithm 1) performs compared to [4]. It is important to note that in [4], the authors re-synthesized the benchmarks using Synopsys Design Compiler before performing signal selection. We, on the other hand, operate on the original benchmarks without any re-synthesis. The

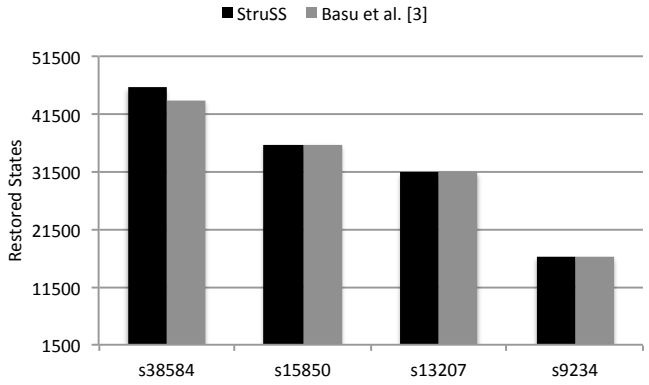

Fig. 4. Comparison with Basu et al. [3]

results are shown in Figure 5. As can be seen, our proposed method *SimuSS* provides up to 5% better restoration performance compared to [4].

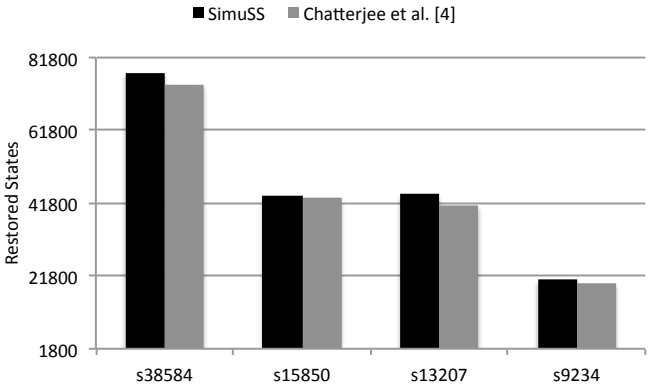

Fig. 5. Comparison with Chatterjee et al. [4]

Finally, we would like to compare the performance of our two proposed approaches, *SimuSS* and *StruSS*. The results are shown in Figure 6. As can be seen, *SimuSS* provides consistently better restoration performance compared to *StruSS*. This can be attributed to the fact that the simulation based signal selection algorithm proposed by [4] performs better than [3].

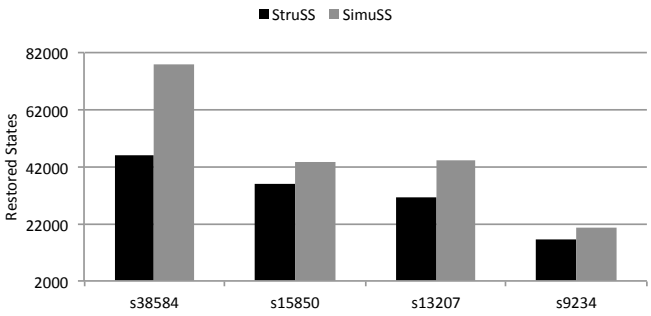

Fig. 6. Comparison of signal selection methods

VI. Conclusion

Signal selection is a critical challenge during post-silicon validation and debug. We have proposed two algorithms for constrained signal selection when a set of signals are already determined by the designer. Our proposed methods perform better restoration than direct application of existing approaches. Our experimental results demonstrate that our simulation based constrained signal selection algorithm provides better restoration compared to constrained signal selection based on structural details.

VII. Acknowledgments

This work was partially supported by NSF grants CNS-0746261 and CCF-1218629.

References

[1] H. F. Ko and N. Nicolici, "Algorithms for state restoration and trace-signal selection for data acquisition in silicon debug", *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 28, no. 2, pp. 285–297, Feb. 2009.

[2] X. Liu and Q. Xu, "Trace signal selection for visibility enhancement in post-silicon validation", in *Design Automation and Test in Europe (DATE)*, 2009, pp. 1338–1343.

[3] K. Basu and P. Mishra, "Efficient Trace Signal Selection for Post Silicon Validation and Debug", in *International Conference on VLSI Design*, 2011, pp. 352–357.

[4] D. Chatterjee, C. McCarter and V. Bertacco, "Simulation-based signal selection for state restoration in silicon debug", in *International Conference on Computer Aided Design (ICCAD)*, 2011, pp. 595–601.

[5] M. Abramovici, P. Bradley, K. Dwarakanath, P. Levin, G. Memmi and D. Miller , "A reconfigurable design-for-debug infrastructure for SoCs," in *Design Automation Conference (DAC)*, 2006, pp. 7–12.

[6] D. Josephson and B. Gottlieb, "The crazy mixed up world of silicon debug [ic validation]," in *Custom Integrated Circuits Conference*, 2004, pp. 665–670.

[7] J. Gao, Y. Han, and X. Li. A New Post-Silicon Debug Approach Based on Suspect Window. In *VLSI Test Symposium*, 2009, pages 85–90.

[8] H. Ko and N. Nicolici. Combining scan and trace buffers for enhancing real-time observability in post-silicon debugging. In *European Test Symposium (ETS)*, 2010, pages 62–67. IEEE, 2010.

[9] K. Basu, P. Mishra and P. Patra, "Efficient Combination of Trace and Scan Signals for Post Silicon Validation and Debug,", in *IEEE International Test Conference (ITC)*, 2011, pp. 1–8.

[10] S. Prabhakar and M. Hsiao, "Multiplexed trace signal selection using non-trivial implication-based correlation.," in *International Symposium on Quality Electronic Design (ISQED)*, 2010, pp. 697 – 704.

[11] X. Liu and Q. Xu, "On multiplexed signal tracing for post-silicon debug," in *Design Automation and Test in Europe (DATE)*, 2011, pp. 1 – 6.