Random Access Scan - II

Virendra Singh
Associate Professor
Computer Architecture and Dependable Systems Lab
Dept. of Electrical Engineering
Indian Institute of Technology Bombay
http://www.ee.iitb.ac.in/~viren
viren@ee.iitb.ac.in

EE 709: Testing & Verification of VLSI Circuits
Lecture – 24 (Feb 28, 2012)
Random Access Scan

- First proposed by Ando in 1980
- It was considered impractical due to large area overhead.
- Baik et al. revisited it in 2004 [ITC’04]
- Proposed as a simultaneous solution to test power, test time, and test data volume
- Baik, 2005, proposed PRAS which showed around 3x speed up, reduction in test data volume with only minor increase in area compared to Serial Scan, thus making RAS practical
Random Access Scan: Architecture

A solution to test power, test time and test data volume

- Architecture
  - Each FF has unique address
  - Address shift register
  - X-Y Decoder
  - Select FF to write/read

Saluja et al [ITC’04]
Scan Operation Example

- Test vector

<table>
<thead>
<tr>
<th>Test</th>
<th>PPI(i_i)</th>
<th>PPO(o_i)</th>
</tr>
</thead>
<tbody>
<tr>
<td>t1</td>
<td>00101</td>
<td>00110</td>
</tr>
<tr>
<td>t2</td>
<td>00100</td>
<td>00101</td>
</tr>
<tr>
<td>t3</td>
<td>11010</td>
<td>11010</td>
</tr>
<tr>
<td>t4</td>
<td>00111</td>
<td>01011</td>
</tr>
</tbody>
</table>

- Scan operation for \(t_2\)

- Complete test application

  Total number of scan operation = 15
Test Vector Ordering

- Test data volume and Test application time is proportional to the random access scan operation
- **Goal**: Reduce # scan operation

<table>
<thead>
<tr>
<th>Test</th>
<th>PPI($i_i$)</th>
<th>PPO($o_i$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>t1</td>
<td>00101</td>
<td>00110</td>
</tr>
<tr>
<td>t2</td>
<td>00100</td>
<td>00101</td>
</tr>
<tr>
<td>t3</td>
<td>11010</td>
<td>11010</td>
</tr>
<tr>
<td>t4</td>
<td>00111</td>
<td>01011</td>
</tr>
</tbody>
</table>

# Scan operation = 8
Optimizing Address Scan

- The cost of address shifting
  - # of scan operation x ASR width
  - Example address set = \{1, 5, 6, 11\} for 4-bit ASR
  - \(4 \times 4 = 16\)
- Proper ordering of address can minimize shifting cost
  - Apply 11(1011) after 5(0101) \(\rightarrow\) needs only 1 left-shift
- Minimizing address shifting cost
  - Construct Address Shifting Distance Graph (ASD-graph)
  - Find min-cost Hamiltonian path using ATSP algorithm (Result: 5 shifts)

\[
G = \langle V, E \rangle \\
V = A_{ij} = \{1, 5, 6, 11\} \\
E = \{e_{ij} | e_{ij} \text{ is an edge between } v_i \text{ and } v_j\}
\]

\(w(e_{ij}) = \)

The number of minimum left-shift operation for the transition from \(v_i\) to \(v_j\).
Hamming Distance Reduction

- Don't care values in PPI do not need scan operation
  - Use Don’t care identification method
    Fully specified test vector → Vectors w/ X values on targeted bit positions without loss of fault coverage

1. Before vector ordering: Identify don’t cares in PPI
2. Vector ordering
3. Simulate test vector in order / Fill X’s with previous vectors PPO
4. Identify more X’s on targeted bit in PPI
   - odd vector
   - even vector
5. Repeat 3,4 until no more X’s are identified
Result (Test Time/Data)

Can test time be further reduced?
Random Access Scan

- Generic Architecture of RAS
- RAS in normal mode
- Application of test vector using RAS
- Reading of test response
Serial Input Random Access Scan

- Architecture
- SIRAS in functional mode is similar to RAS
- SRAM type
- Test Application
- Test Response

Adiga, VLSID’10

Feb 28, 2012

EE-709@IITB
Variable Word Length Random Access Scan

Architecture of VWLRAS

In functional mode, it is similar to RAS

Test Application using VWLRAS

Test Responses

Adiga, VLSID’10
Flip Flop Grouping

Both VWLRAS and SIRAS require Flip Flops to be grouped in order to reduce the test application time.

Flip Flops are grouped by constructing the graph using Flip Flops as nodes and edges weights are calculated using the hamming distance like calculation between test vectors using weight matrix.
Comparison of SIRAS, VWLRAS and RAS

TT Reduction

TDV Reduction

Feb 28, 2012 EE-709@IITB
T-Flip-Flop based Scan cell for RAS

From Combinational Logic

CLK

Row

Col

Row Decoder

\[ \sqrt{N_{ff}} \text{ lines} \]

\[ \{ \log_2 N_{ff} \} \]

Address

\[ \text{Master} \]

\[ \text{Slave} \]

To combinational Logic

To Bus Leading to Primary O/P

Mudlapur, VDAT'05

Feb 28, 2012

EE-709@IITB
Modified T-Flip-Flip based Cell

Gandhi, ETS’10

- Architecture
- Functional Mode
- Test Application & Simultaneous Read
- Reading Response only

Test Enable

Single Ended Read Circuits and MISR

CLK

Test Control

AOI

Address

Row Decoder

Col Decoder

\[ \sqrt{N_{ff}} \text{ lines} \]

\[ \sqrt{N_{ff}} \text{ lines} \]
Critical Path Analysis

From Combinational Logic

TO BUS

Master

Slave

To Bus
Leading to
Primary O/P

TFF Based Scan Cell

MTFF Based Scan Cell

Feb 28, 2012
EE-709@IITB
# Gate Overhead

<table>
<thead>
<tr>
<th>Circuit</th>
<th>#FF</th>
<th>#Gates</th>
<th>GOV_SS</th>
<th>GOVTFF_RAS</th>
<th>GOVMTFF_RAS</th>
</tr>
</thead>
<tbody>
<tr>
<td>S5378</td>
<td>179</td>
<td>2779</td>
<td>17.47%</td>
<td>25.3%</td>
<td>20.3%</td>
</tr>
<tr>
<td>S9234</td>
<td>228</td>
<td>5597</td>
<td>12.04%</td>
<td>17.44%</td>
<td>13.99%</td>
</tr>
<tr>
<td>S13207</td>
<td>669</td>
<td>7951</td>
<td>21.36%</td>
<td>30.75%</td>
<td>24.6%</td>
</tr>
<tr>
<td>S15850</td>
<td>597</td>
<td>9772</td>
<td>16.7%</td>
<td>24.15%</td>
<td>19.36%</td>
</tr>
<tr>
<td>S35932</td>
<td>1728</td>
<td>16065</td>
<td>25.36%</td>
<td>36.4%</td>
<td>29.1%</td>
</tr>
<tr>
<td>S38417</td>
<td>1636</td>
<td>22179</td>
<td>19.3%</td>
<td>27.8%</td>
<td>22.2%</td>
</tr>
<tr>
<td>S38584</td>
<td>1452</td>
<td>19253</td>
<td>19.7%</td>
<td>28.3%</td>
<td>22.6%</td>
</tr>
</tbody>
</table>
Efficient Decoder Design

Abhishek, ISCAS’10

- Optimize the column decoder
- Uses concept of basis vectors in linear algebra
- Minimizes the number of linear combinations of basis vectors to generate the transition vectors
- Directly related to the no. of clock cycles
- 2-3 times speed up compare to standard RAS

Feb 28, 2012

EE-709@IITB
Thank You