The State-of-the-Art Test Compression and Test Response Compaction Techniques

2008.11.15

Hong-Sik Kim
Contents

- Introduction
- Test Stimulus Compression
- Test Response Compaction
- Industrial Practice
- Conclusion
Test Data Increase

- **Increase in the number of test patterns**
  - Increasing number of embedded IP cores
  - Increasing number of target fault models for DSM technologies
    - Transition fault testing requires 3~5 times more test patterns than stuck at fault testing

- **Newer chips with more pins and functionality but older ATE equipment**
  - Less test channels
  - Not enough ATE memory

- **Low-cost ATE : reduced pin count test**
- **Keep down test cost**
Test Data Volume vs Technology


Computer Systems & Reliable SoC Lab.
Test Quality Improvement for Nano Technology

- **Multiple detect**
  - Shown to improve test quality in 200K production run experiment

- **DFM-oriented test**
  - Extractions from physical database (such as Calibre)
  - For example, deterministic bridge fault model

- **Timing aware test**
  - Use SDF to test longest paths
Test Data Volume vs Technology

Source: www.elecdesign.com
Why Test Compression?

Introduction

Test Quality Requirement

Compression

Standard ATPG

180nm  130nm  90nm
Full Scan VS Compressed Scan

Introduction
**Test Data Compression**

**Introduction**

- **Advantage**
  - Complete set of ATPG test patterns can be applied
  - Compatible with the convectional design rules and test generation flow for scan testing

- **Benefits**
  - Reduce amount of test data
    - Life cycle of older tester with limited memory is extended
  - Test time reduces for a given test data bandwidth
    - With test compression, larger number of scan chains can be used
Test Data Compression

Test (Stimulus) Compression
- Code-based compression
- Linear-decompression based scheme
- Broadcast scan

Test Response Compaction
- Space compaction
- Time compaction
- Mixed one
❖ Introduction
❖ **Test Stimulus Compression**
❖ Test Response Compaction
❖ Industrial Practice
❖ Conclusion
Categories of TC

❖ **Code-based Schemes**
  - Traditional coding algorithm
  - Entropy coding

❖ **Linear-Decompression-based Schemes**
  - Combinational
    - XOR networks
  - Sequential (Static/Dynamic)
    - LFSR reseeding

❖ **Broadcast-Scan-Based Schemes**
  - Traditional
    - Scan forest/Illinois scan
  - Reconfigurable
    - Static/Dynamic
Code-Based TC

- **Dictionary coding**
  - fixed to fixed
- **Huffman coding**
  - fixed to variable
- **Run-length coding**
  - variable to fixed
- **Golomb coding**
  - variable to variable
- **Arithmetic coding**
  - fixed to variable
Code-Based TC

- **Run-length coding based test compression**
Golomb coding based test compression

Dictionary coding

Code-Based TC

- Selective HC based test compression

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Freq.</th>
<th>Pattern</th>
<th>Huffman Code</th>
<th>Selected Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>S_0</td>
<td>22</td>
<td>0010</td>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td>S_1</td>
<td>13</td>
<td>0100</td>
<td>00</td>
<td>110</td>
</tr>
<tr>
<td>S_2</td>
<td>7</td>
<td>0110</td>
<td>110</td>
<td>111</td>
</tr>
<tr>
<td>S_3</td>
<td>5</td>
<td>0111</td>
<td>010</td>
<td>00111</td>
</tr>
<tr>
<td>S_4</td>
<td>3</td>
<td>0000</td>
<td>0110</td>
<td>00000</td>
</tr>
<tr>
<td>S_5</td>
<td>2</td>
<td>1000</td>
<td>0111</td>
<td>01000</td>
</tr>
<tr>
<td>S_6</td>
<td>2</td>
<td>0101</td>
<td>11100</td>
<td>00101</td>
</tr>
<tr>
<td>S_7</td>
<td>1</td>
<td>1011</td>
<td>111010</td>
<td>01011</td>
</tr>
<tr>
<td>S_8</td>
<td>1</td>
<td>1100</td>
<td>111011</td>
<td>01100</td>
</tr>
<tr>
<td>S_9</td>
<td>1</td>
<td>0001</td>
<td>111100</td>
<td>00001</td>
</tr>
<tr>
<td>S_10</td>
<td>1</td>
<td>1101</td>
<td>111110</td>
<td>01101</td>
</tr>
<tr>
<td>S_11</td>
<td>1</td>
<td>1111</td>
<td>111110</td>
<td>01111</td>
</tr>
<tr>
<td>S_12</td>
<td>1</td>
<td>0011</td>
<td>111111</td>
<td>00011</td>
</tr>
<tr>
<td>S_13</td>
<td>0</td>
<td>1110</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>S_14</td>
<td>0</td>
<td>1010</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>S_15</td>
<td>0</td>
<td>1001</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>
Linear Decompression Based Scheme

- Compress test data by using linear equation and solver

- Classification
  - Combinational linear de-compressor
  - Sequential linear de-compressor
    - Fixed length
    - Variable length
Linear Decompression Based Scheme

- Compress test data by using linear equation and solver
Linear Decompression Based Scheme

- Compress test data by using linear equation and solver

\[
\begin{align*}
Z_1 &= X_2 \oplus X_5 \\
Z_2 &= X_3 \\
Z_3 &= X_1 \oplus X_4 \\
Z_4 &= X_1 \oplus X_6 \\
Z_5 &= X_3 \oplus X_7 \\
Z_6 &= X_1 \oplus X_4 \\
Z_7 &= X_1 \oplus X_2 \oplus X_5 \oplus X_6 \\
Z_8 &= X_2 \oplus X_5 \oplus X_8 \\
Z_9 &= X_1 \oplus X_4 \oplus X_9 \\
Z_{10} &= X_1 \oplus X_2 \oplus X_5 \oplus X_6 \\
Z_{11} &= X_2 \oplus X_3 \oplus X_5 \oplus X_7 \oplus X_8 \\
Z_{12} &= X_3 \oplus X_7 \oplus X_{10}
\end{align*}
\]
Compress test data by using linear equation and solver.
Linear Decompression Based Scheme

- Combinational Linear Decompression
Linear Decompression Based Scheme

- LFSR Reseeding Scheme
  - B. Koenemann, “LFSR-coded Test Patterns for Scan Designs,” ETC, 1991

![Diagram of LFSR Reseeding Scheme](image)

Seed for C₀

A system of linear equations:

\[
\begin{align*}
  a₀ &= 0 \\
  a₁ &= 1 \\
  a₂ &= 0
\end{align*}
\]

\[
\begin{align*}
  a₀ + a₂ &= 0 \\
  a₀ + a₁ &= 1
\end{align*}
\]
Linear Decompression Based Scheme

- **Ring Generator**

![Diagram of a ring generator with XOR gates and levels of logic]

<table>
<thead>
<tr>
<th>Type</th>
<th>XOR gates</th>
<th>Levels of logic</th>
<th>Fan-out</th>
</tr>
</thead>
<tbody>
<tr>
<td>Internal feedback LFSR</td>
<td>$k$</td>
<td>$\log_2 k$</td>
<td>2</td>
</tr>
<tr>
<td>External feedback LFSR</td>
<td>$\lambda$</td>
<td>1</td>
<td>$\lambda + 1$</td>
</tr>
<tr>
<td>Hybrid LFSR</td>
<td>$(k+1)/2$</td>
<td>$(1, \log_2 k)$</td>
<td>$(2, k+1)$</td>
</tr>
<tr>
<td>Cellular automata</td>
<td>$2n - 2$</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>Ring generators</td>
<td>$k$</td>
<td>1</td>
<td>2</td>
</tr>
</tbody>
</table>
Linear Decompression Based Scheme

- Variable rank LFSR with BF
Broadcasting

**Concept**
- Input the same test data to multiple scan chains

**Various TC Schemes based Broadcast**
- ILLINOIS scan
- Scan forest
- Multicast based TC
Concept

- Input the same test data to multiple scan chains

Force ATPG tool to generate patterns for broadcast scan

Scan input

<table>
<thead>
<tr>
<th>SC_1</th>
<th>SC_2</th>
<th>SC_k</th>
</tr>
</thead>
<tbody>
<tr>
<td>C_1</td>
<td>C_2</td>
<td>C_k</td>
</tr>
</tbody>
</table>

Inputs

- | 1 | 2 | 3 | 4 |
- | C_1 |

- | 1 | 2 | 3 |
- | C_2 |
ISA (Illinois scan architecture)

- I. Hamzaoglu et al, “Reducing Test Application Time for Full Scan Embedded Cores,” FTCS, 1999
Scan forest


Scan forest is constructed based on structural analysis.

- Reduced number of scan leaf decreases the number of XOR gate in response compactor.

- Reduce both test data volume and power consumption.
**Multicast based TC**


<table>
<thead>
<tr>
<th>Scan types</th>
<th>Type description (mode)</th>
</tr>
</thead>
<tbody>
<tr>
<td>2'b00</td>
<td>Broadcasting</td>
</tr>
<tr>
<td>2'b01</td>
<td>Multicasting (top-down Golomb coding)</td>
</tr>
<tr>
<td>2'b10</td>
<td>Multicasting (bottom-up Golomb coding)</td>
</tr>
<tr>
<td>2'b11</td>
<td>Single-casting (for casting a single scan chain)</td>
</tr>
</tbody>
</table>

**Table:**

- Data volume (Mbits):
  - Original data volume: 10 Mbits
  - Universal multicasting: 1.3x reduction
  - Control subpattern coding: 13.3x reduction
  - Control subpattern skipping: 41.1x reduction
  - Partial data reuse: 51.3x reduction

**Diagram:**

- Example: Scan chains 1, 3, 5
- Single test channel
Contents

- Introduction
- Test Stimulus Compression
- **Test Response Compaction**
- Industrial Practice
- Conclusion
Categories of TC

- **Space compaction**
  - Reducing the number of output bit size

- **Time compaction**
  - Reducing the number of output response patterns

- **Mixed space and time compaction**
Categories of TC

- **Space compaction**
  - Reducing the number of output bit size

- **Time compaction**
  - Reducing the number of output response patterns

- **Mixed space and time compaction**
Response Compaction Issues

- **Aliasing problem**

- **X propagation**
  - Many sources of unknown X’s in output response data
    - Unmodeled ATPG logics: RAM’s, mixed signal logic, black boxes etc
    - Uninitialized memory elements (non scanned FF’s)
    - Floating tri-states
    - Multi-cycle paths
    - Etc
X propagation Problem

- **Conventional Scan**
  - Easy to handle X values in test response by masking them on tester

- **Logic BIST and test response compaction**
  - X’s corrupt final signature
  - Prevents observation of other scan cells
  - Output compression ratios and ATPG results are degraded by the capture of unknown value

![Diagram showing X propagation and test response compaction](image)
Handling X’s

- **X-Bounding (X-blocking)**
  - Insert DFT to prevent X’s from propagating to output

- **X-tolerant compactor**
  - Make compactor resilient to one or several X’s propagated to the compactor

- **X-Masking**
  - Mask X’s at the input to compactor
  - Mask data required
X-Tolerant TRC

- **X-Compact**

- Combinational compactor
- Tolerates one X per scan slice
- Detects 1, 2, or any odd errors
- Corrupted outputs will be masked on tester
- **X-Canceling**
X-Masking TRC

X-Masking

- X’s can be masked off right before the response compactor
- Mask data is required to indicate when the masking should take place
- Mask data can be compressed
  - LFSR reseeding, run-length coding
X-Masking TRC

Masking compression based on HC


Diagram showing the architecture of the X-Masking TRC system.
X-Masking TRC

X-Block

Contents

- Introduction
- Test Stimulus Compression
- Test Response Compaction
- Industrial Practice
- Conclusion
**Embedded Deterministic Test (EDT)**

- TestKompress
- First commercially available on-chip test compression product
Virtual scan

- TestKompress
- First commercial product based on the broadcast scan scheme using combinational logic for pattern decompression
Adaptive scan
- DFTMAX
## Summary of Commercial Solutions

<table>
<thead>
<tr>
<th>Industrial Practice</th>
<th>Stimulus Compression</th>
<th>Response Compression</th>
</tr>
</thead>
<tbody>
<tr>
<td>EDT (Mentor)</td>
<td>Ring generator</td>
<td>XOR tree</td>
</tr>
<tr>
<td>Virtual Scan (Syntest)</td>
<td>Combinational logic network (broadcast)</td>
<td>XOR tree</td>
</tr>
<tr>
<td>DFTMAX (Synopsys)</td>
<td>Combinational MUX network (broadcast)</td>
<td>XOR tree</td>
</tr>
</tbody>
</table>

---

*Computer Systems & Reliable SoC Lab.*
Conclusion

- **Test Compression**
  - Effective method to reduce test data volume and test application time
  - Good solution for test cost reduction
  - Easy to implement and capable of producing high-quality tests
  - Part of design flow