Chapter 2

Digital Test Architectures
What is this chapter about?

- Introduce Basic Design for Testability (DFT) Techniques
- Focus on Widely Used or Emerging DFT Architectures
- Illustrate Basic Test Architectures, Low-Power Test Architectures, and At-Speed Test Architectures
Digital Test Architectures

- Introduction
- Scan Design
- Logic Built-In Self-Test
- Test Compression
- Random-Access Scan Design
- Concluding Remarks
Introduction

Evolution of DFT advances in testing digital circuits
Introduction

- **Scan Design**
  - Replace all selected storage elements with scan cells
  - Connect scan cells into multiple shift registers (scan chains)
  - Become inefficient to test deep submicron or nanometer VLSI

- **Logic Built-In Self-Test (BIST)**
  - Combine with scan approach at the design stage
  - Generate test patterns and analyze the output response
  - Crucial for safety-critical and mission-critical applications
Introduction

- **Test Compression**
  - A supplemental DFT technique to scan
  - Can reduce test data volume and test application time
  - Add some additional on-chip hardware

- **Random-Access Scan Design**
  - Randomly and uniquely addressable, similar to RAM
  - A promising alternative to Scan design for shift power reduction
Scan Design

- Widely used structured DFT architecture
- Replace all selected storage elements with scan cells
- Connect scan cells into scan chains
- Operated in three modes:
  - Normal mode
    - All test signals are turned off.
    - The scan design operates in the circuit’s original functional configuration.
  - Shift mode
    - To shift data into and out of the scan cells
  - Capture mode
    - To capture test response into scan cells
Scan Architectures

Muxed-D Scan Cell

The multiplexer uses a scan enable (SE) to select between the data input (DI) and the scan input (SI).

Sequential circuit example
Scan Architectures

Replace FF1, FF2 and FF3 with SFF1, SFF2 and SFF3.

In shift mode, SE is set to 1, the scan cells operate as a single scan chain.

In capture mode, SE is set to 0, scan cells are used to capture the test response from the combinational logic.

Muxed-D Scan Design
Input selection is conducted using two independent clocks, data clock $DCK$ and shift clock $SCK$. 

**Scan Architectures**

**Clocked-Scan Cell**
Scan Architectures

DCK and SCK are used for distinguishing shift and capture operations; while SE is used to switch the shift and capture operations in muxed-D scan design.

Clocked-Scan Design
SRL can be used as an LSSD scan cell. This scan cell contains two latches, a master two-port D latch $L_1$ and a slave D latch $L_2$. Clocks C, A, and B are used to select between D and $+L_1$ and $+L_2$.

Level-Sensitive Scan Design (LSSD) can be implemented using a single-latch design or a double-latch design.
Scan Architectures

The system clocks $C_1$ and $C_2$ should be applied in a nonoverlapping fashion.

LSSD single-latch design
Scan Architectures

During the shift operation, clocks $A$ and $B$ are applied in a nonoverlapping manner, and the scan cells, $SRL_1 \sim SRL_3$, form a single scan chain from $SI$ to $SO$.

During the capture operation, clocks $C_1$ and $C_2$ are applied in a non-overlapping manner to load the test response from the combinational logic into the scan cells.
Scan Architectures

- **Enhanced-Scan Design**
  - An alternative at-speed scan design for testing delay faults; testing of a delay fault requires a pair of test vectors in an at-speed fashion.
  - Enhanced-scan cell can store two bits of data; achieved by adding a D latch to a muxed-D scan cell or clocked-scan cell.
  - Disadvantages:
    - Higher hardware overhead
    - May activate many false paths causing an over-test problem
The first test vector $V_1$ is shifted into $SFF_1 \sim SFF_s$ and then stored into the additional latches ($LA_1 \sim LAs$) when the $UPDATE$ signal is set to 1.

Next, the second test vector $V_2$ is shifted into the scan cells while the $UPDATE$ signal is set to 0, in order to preserve the $V_1$ value in the latches ($LA_1 \sim LAs$).

Once the second vector $V_2$ is shifted in, the $UPDATE$ signal is applied, in order to change $V_1$ to $V_2$ while capturing the output response at-speed into the scan cells by applying $CK$ after exactly one clock cycle.
Low-Power Scan Architectures

- **Serial Scan Design**
  - **Advantage:**
    - Low routing overhead
  - **Disadvantages:**
    - Scan cells cannot be controlled or observed without affecting the values of other scan cells in the same scan chain
    - High switching activities during shift and capture can cause excessive shift (or test) power dissipation

- **Low-Power Scan Design**
  - Test power is related to dynamic power, and is proportional to $V_{DD}^2 f$
    - $V_{DD}$ is the supply voltage
    - $f$ is the switching frequency of the circuit node under test
Example Low-Power Scan Architectures

- **Reduced-Voltage Low-Power Scan Design**
  - Reduce the supply voltage

- **Reduced-Frequency Low-Power Scan Design**
  - Slow down the shift clock frequency but increase test application time

- **Multi-Phase Low-Power Scan Design**
  - Split the shift clock into a number of nonoverlapping clock phases but increase routing overhead and complexity during clock tree synthesis

- **Bandwidth-Matching Low-Power Scan Design**
  - Use pairs of serial-in/parallel-out shift register and parallel-in/serial-out shift register for bandwidth matching

- **Hybrid Low-Power Scan Design**
  - Combine any of the above-mentioned low-power scan designs
Multi-Phase Low-Power Scan Design

The clock \( CK \) is split into three clock phases \( CK_1 \), \( CK_2 \), and \( CK_3 \).

Using this scheme, a 3X reduction in shift power can be achieved, assuming each clock drives an equal number of scan cells.
Bandwidth-Matching Low-Power Scan Design

Each scan chain is split into 4 sub-scan chains with the SI and SO ports of each 4 sub-scan chains connected to a serial-in/parallel-out shift register and a parallel-in/serial-out shift register, respectively.
At-Speed Scan Architectures

- **Synchronous Design**
  - A scan design if the active edges of all capture clocks controlling the clock domains can be aligned precisely or triggered simultaneously

- **Asynchronous Design**
  - A scan design if not synchronous
At-Speed Scan Architectures

- Two basic schemes for test multiple clock domain at-speed
  - Skewed-load (Launch-on-shift)
    - Use the last shift clock pulse followed immediately by a capture clock pulse to launch the transition and capture the output response
  - Double-capture (Launch-on-capture or Broad-side)
    - Use two consecutive capture clock pulses to launch the transition and capture the output test response

- Similarity
  - Can test path-delay faults and transition faults. The second capture clock pulse must be running at the domain’s operating frequency or at-speed.

- Difference
  - Skewed-load requires the domain’s SE to switch value between the launch and capture clock pulses making SE act as a clock signal.
Basic At-Apeed Test Schemes

Skewed-load

Double-capture
Clock grouping

- Can reduce test application time and test data volume during automatic test pattern generation (ATPG)
- Is a process used to analyse all data paths in the scan design in order to determine all independent or noninteracting clocks that can be grouped and applied simultaneously
Clock grouping example

CD₂ and CD₃ are independent from each other; hence their related clocks can be applied simultaneously during test as CK₂.

CD₄ through CD₇ can also be applied simultaneously during test as CK₃.

Therefore three grouped clocks instead of seven individual clocks can be used to test the circuit during the capture operation.
Clock schemes

- **One-hot clocking**
  - Apply only one grouped clock during each capture operation
  - Produce the highest fault coverage but generate most test patterns

- **Simultaneous clocking**
  - Mask off unknown values at the originating scan cells or receiving scan cells across clock domains
  - Generate the least number of patterns but may result in high fault coverage loss

- **Staggered clocking**
  - Grouped clocks are applied sequentially
  - Generate pattern count close to simultaneous clocking and fault coverage close to one-hot clocking
At-speed Clocking Scheme for Testing Two Interacting Clock Domains

(One-hot clocking)
At-speed Clocking Scheme for Testing Two Interacting Clock Domains

(Simultaneous clocking)
At-speed Clocking Scheme for Testing Two Interacting Clock Domains

(Staggered clocking)
How to Generate Shift and Capture Clocks

- **Supplied from the Tester**
  - Increase test cost
  - Limited high-frequency channels

- **Generated by Phase-Locked Loop (PLL)**
  - Pipelined Scan Enable (SE) signal
  - Test clock controller
Pipelined Scan Enable (SE) Design

$SE_1$ for positive-edge scan cells

$SE_2$ for negative-edge scan cells
On-Chip Clock Controller

When `scan_en` is set to 1, `scan_clk` is directly connected to `clk_out`; when `scan_en` is set to 0, the output of the clock-gating cell is directly connected to `clk_out`.

The clock-gating cell makes sure that no glitches or spikes appear on `clk_out`.

When `scan_en` is set to 1, `scan_clk` is directly connected to `clk_out`; when `scan_en` is set to 0, the output of the clock-gating cell is directly connected to `clk_out`.
On-Chip Clock Controller - Waveform

scan_clk
scan_en
pll_clk
hs_clk_en
cgc_clk_out
clk_out
Logic Built-In Self-Test (BIST)

A typical logic BIST system

The logic BIST controller provides a pass/fail indication once the BIST operation is complete.
Logic Built-In Self-Test

- **TPG**
  - Constructed from linear feedback shift register (LFSR) or cellular automata
  - Exhaustive testing – all possible $2^n$ test patterns
  - Pseudo-random testing – a subset of $2^n$ test patterns
  - Pseudo-exhaustive testing – $2^w$ or $2^k-1$ test patterns, where $w < k < n$

- **ORA**
  - Constructed from multiple-input signature register (MISR)
Logic BIST Architectures

- Test-per-Scan BIST
  - Hardware overhead is low

- Test-per-Clock BIST
  - Execute tests faster than Test-per-Scan BIST
  - More hardware overhead
Example Logic BIST Architectures

- **Self-Testing Using MISR and Parallel SRSG (STUMPS)**
  - Based on test-per-scan BIST
  - Integrate with traditional scan architecture
  - Linear phase shifter and linear phase compactor is often used
  - Lose some fault coverage

- **Concurrent Built-In Logic Block Observer (CBILBO)**
  - Based on test-per-clock BIST
  - Signature analysis is separated from test generation
  - Possible to achieve 100% single-stuck fault coverage
  - Hardware cost is higher than STUMPS
A STUMPS-based architecture
A three-stage concurrent BILBO (CBILBO)
Example CBILBO Applications

(a) For testing a finite-state machine

(b) For testing a pipelined-oriented circuit

CBILBO Architectures
Coverage-Driven Logic BIST Architectures

- Approaches to Enhance logic BIST Fault Coverage
  - In-field coverage enhancement
    - Weighted Pattern Generation
    - Test Point Insertion
    - Mixed-Mode BIST
  - Manufacturing Coverage Enhancement
    - Hybrid BIST
Weighted Pattern Generation

Employ an LFSR

Insert a combinational circuit between the output of LFSR and the CUT

Skew the LFSR probability distribution of 0.5 to either 0.25 or 0.75

Example weighted LFSR as PRPG
Test Point Insertion

(a) Test point with a multiplexer

(b) Test point with AND-OR gates

Typical test point inserted for improving a circuit’s fault coverage
Example of Inserting Test Points to Improve Detection Probability

(a) An output RP-resistant stuck-at-0 fault

\[
\text{Min. Detection Probability } = \frac{1}{64}
\]

(b) Example inserted test points

\[
\text{Min. Detection Probability } = \frac{7}{128}
\]
Test Point Insertion

- **Test Point Placement**
  - Use fault simulation
  - Use testability measures to guide them

- **Control Point Activation**
  - During normal operation
    - Deactivated
  - During testing
    - Random activation
    - Deterministic activation
Mixed-Mode BIST

- **ROM Compression**
  - Store deterministic patterns in ROM

- **LFSR Reseeding**
  - Generate deterministic patterns by reseeding LFSR with computed seeds

- **Embedding Deterministic Patterns**
  - Transform the “useless” patterns into deterministic patterns
Hybrid BIST

- Perform top-up ATPG for the faults not detected by BIST
- Store the patterns directly on the tester
- Store the patterns on the tester in a compressed form and make use of the existing BIST hardware to decompress them
Low-Power Logic BIST Architecture

- **Low-Transition BIST Design**
  - Insert an AND gate and a toggle flip-flop at the scan input of the scan chain
  - **Advantages:**
    - Less design intrusive
    - no performance degradation
    - Low hardware overhead
  - **Disadvantages:**
    - Low fault coverage
    - Long test sequence
Low-Power Logic BIST Architecture

- Test-Vector-Inhibiting BIST Design
  - Inhibit the LFSR-generated pseudo-random patterns which do not contribute to fault detection from being applied to the CUT
  - Advantages:
    - Reduce test power
    - No fault coverage loss as the original LFSR
  - Disadvantage:
    - High hardware overhead
Low-Power Logic BIST Architecture

- **Modified LFSR Low-Power BIST Design**
  - Use two interleaved $n/2$-stage LFSRs
  - **Advantages:**
    - Shorter test length
    - High percentage of power reduction
    - No performance degradation
    - No test time increase
  - **Disadvantage:**
    - Require constructing special clock trees
At-Speed Logic BIST Architectures

- **Single-capture**
  - One-hot single-capture
  - Staggered single-capture
- **Skewed-load**
  - One-hot skewed-load
  - Aligned skewed-load
  - Staggered skewed-load
- **Double-capture**
  - One-hot double-capture
  - Aligned double-capture
  - Staggered double-capture
One-Hot Single-Capture

- **Advantages:**
  - No need to worry about clock skews between clock domains
  - Can be used for slow-speed testing
  - Use a global scan enable (GSE) signal – compatible with Scan

- **Disadvantage:**
  - Long test time
Staggered Single-Capture

- **Advantage:**
  - Can detect inter-clock-domain delay faults within two clock domains

- **Disadvantage:**
  - May cause some structural fault coverage loss if the sequence order of the capture clocks is fixed.
One-Hot Skewed-Load

Advantage:
- Can be used for at-speed testing of intra-clock-domain delay faults

Disadvantages:
- Cannot be used for testing of inter-clock-domain delay faults
- Long test time
Aligned Skewed-Load

Capture aligned skewed-load    Launch aligned skewed-load

- **Advantage:**
  - All intra-clock-domain and inter-clock-domain faults can be tested in synchronous clock domains

- **Disadvantage:**
  - Require more complex timing-control diagram
**Staggered Skewed-Load**

![Diagram of Staggered Skewed-Load]

- **Advantage:**
  - All intra-clock-domain and inter-clock-domain faults can be tested in both synchronous and asynchronous clock domains.

- **Disadvantage:**
  - Complicated physical implementation
One-Hot Double-Capture

- **Advantage:**
  - Can be used for true at-speed testing of intra-clock-domain delay faults

- **Disadvantages:**
  - Cannot be used for testing of inter-clock-domain delay faults
  - Long test time
Aligned Double-Capture

Capture aligned double-capture  Launch aligned double-capture

- **Advantage:**
  - Can test all intra-clock-domain and inter-clock-domain delay faults in synchronous clock domains

- **Disadvantage:**
  - Require precise alignment capture pulses

System-on-Chip Test Architectures

Ch. 2 – Digital Test Architectures - P. 58
Staggered Double-Capture

- Advantages:
  - Ease physical implementation
  - Integrate logic BIST with scan/ATPG

- Disadvantage:
  - May cause fault coverage loss due to the ordered sequence of capture clocks.
### Summary of Industry Practices for At-Speed Logic BIST

<table>
<thead>
<tr>
<th>Industry Practices</th>
<th>Skewed-load</th>
<th>Double-Capture</th>
</tr>
</thead>
<tbody>
<tr>
<td>Encounter Test</td>
<td>Through service</td>
<td>Through service</td>
</tr>
<tr>
<td>ETLogic</td>
<td>✓</td>
<td>Through service</td>
</tr>
<tr>
<td>LBIST Architect</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>TurboBIST-Logic</td>
<td></td>
<td>✓</td>
</tr>
</tbody>
</table>
Test Compression

- **Decompressor**
  - Add some additional on-chip hardware before the scan chains to decompress the test stimulus
  - Use lossless compression

- **Compactor**
  - Add some additional on-chip hardware after scan chains to compact the response
  - The compaction is lossy

- **Advantages:**
  - Reduce ATE memory
  - Reduce test data volume and test application time
Test Compression Architecture

Diagram:
- Low-Cost ATE
- Compressed Stimulus
- Scan-Based Circuit (CUT)
- Stimulus
- Response
- Compacted Response
- Compactor
- Decompressor

Diagram Description:
- The diagram illustrates a test compression architecture.
- The Low-Cost ATE provides a compressed stimulus.
- This stimulus is then decompressed by the decompressor and applied to the Scan-Based Circuit (CUT).
- The response from the CUT is then compacted by the compactor.
- The compacted response is processed by the decompressor and the compactor stages.

System-on-Chip Test Architectures

Ch. 2 – Digital Test Architectures - P. 62
Circuits for Test Stimulus Compression

- **Linear-Decompression-Based Schemes**
  - Combinational linear decompressors
  - Sequential linear decompressors

- **Broadcast-Scan-Based Schemes**
  - Broadcast scan
  - Illinois scan
  - Multiple-input broadcast scan
  - Reconfigurable broadcast scan
  - Virtual scan

- **Comparison**
Linear-Decompression-Based Schemes

- **Linear Decompressor Concept**
  - Consists of only XOR gates and Flip-Flops
  - Its output space is a linear subspace that is spanned by a Boolean matrix.

- **Combinational Linear Decompressor**
  - Consists of only XOR gates

- **Sequential Linear Decompressor**
  - Consists of XOR gates and Flip-Flops
  - Flip-flops provides additional free variables for state encoding.
symbolic simulation for linear decompressor

\[
\begin{align*}
Z_9 &= X_1 \oplus X_4 \oplus X_9 \\
Z_{10} &= X_1 \oplus X_2 \oplus X_5 \oplus X_6 \\
Z_{11} &= X_2 \oplus X_3 \oplus X_5 \oplus X_7 \oplus X_8 \\
Z_{12} &= X_3 \oplus X_7 \oplus X_{10} \\
Z_5 &= X_3 \oplus X_7 \\
Z_6 &= X_1 \oplus X_4 \\
Z_7 &= X_1 \oplus X_2 \oplus X_5 \oplus X_6 \\
Z_8 &= X_2 \oplus X_5 \oplus X_8 \\
Z_1 &= X_2 \oplus X_5 \\
Z_{10} &= X_1 \oplus X_2 \oplus X_6 \\
Z_{11} &= X_2 \oplus X_3 \oplus X_6 \\
Z_{12} &= X_3 \oplus X_7 \oplus X_{10}
\end{align*}
\]
System of linear equations for the decompressor

\[ \begin{bmatrix}
0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\
1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\
1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\
1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\
0 & 1 & 1 & 0 & 1 & 0 & 1 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 
\end{bmatrix} \begin{bmatrix}
X_1 \\
X_2 \\
X_3 \\
X_4 \\
X_5 \\
X_6 \\
X_7 \\
X_8 \\
X_9 \\
X_{10} \\
X_{11} \\
X_{12} 
\end{bmatrix} = \begin{bmatrix}
Z_1 \\
Z_2 \\
Z_3 \\
Z_4 \\
Z_5 \\
Z_6 \\
Z_7 \\
Z_8 \\
Z_9 \\
Z_{10} \\
Z_{11} \\
Z_{12} 
\end{bmatrix} \]
Combinational Linear Decompressor

- **Advantage:**
  - Simpler hardware and control because only XOR gates are used

- **Disadvantages:**
  - Low Encoding Efficiency
    - Because no free variables are used
    - Can be improved by dynamically adjusting the number of scan chains that are loaded in each clock cycle.
Sequential Linear Decompressor

- Based on linear finite-state machines
  - Examples: LFSRs, cellular automata, ring generators

- Advantages:
  - Allow free variables from earlier clock cycles
  - Much greater flexibility than combinational linear decompressor

- Two classes
  - Static reseeding
    - Drawbacks
      - The tester is idle while the LFSR is running in autonomous mode.
      - The LFSR must be at least as large as the number of specified bits in the test cube.
  - Dynamic reseeding
Typical Sequential Linear Decompressor

Dynamic reseeding calls for the injection of free variables coming from the tester into the LFSR as it loads the scan chains.
Broadcast-Scan-Based Schemes

- Broadcast scan
- Illinois Scan
- Multiple input broadcast scan
- Reconfigurable broadcast scan
- Virtual scan
Broadcast Scan

- Broadcasting to scan chains driving independent circuit
- Won’t affect fault coverage if all circuits are independent
Illinois Scan

- Consists of two modes of operations
  - Broadcast mode
  - Serial scan mode

- Main Drawback
  - No test compression in serial scan mode

- Ways to reduce number of patterns
  - Multiple-Input broadcast scan
  - Reconfigurable broadcast scan
Illinois Scan Architecture

Two Mode of Illinois Scan Architecture

(a) Broadcast mode

(b) Serial chain mode
Multiple-input broadcast scan

- Use more than one channel to drive all scan chains

- The shorter each scan chain is, the easier to detect more faults because fewer constraints are placed on the ATPG
Reconfigurable Broadcast Scan

- Reduce the number of required channels compared to multiple-input broadcast scan
- Provide the capability to reconfigure the set of scan chains
- Two possible reconfiguration schemes
  - Static reconfiguration
  - Dynamic reconfiguration
    - Need more control information *versus* static reconfiguration
Example MUX Network with Control Line(s) connected only to select pins of the multiplexers.
Virtual Scan

- Use Combinational logic network for stimulus decompression – called Broadcaster
  - Buffers, inverters, AND/OR gates, MUXs, XOR gates

- Advantages
  - One-Step ATPG – No need to solve linear equations as required in sequential linear decompressor.
  - Dynamic compaction can be effectively utilized during the ATPG process.
Example Virtual Scan Broadcast Using an XOR Network

Broadcaster using an example XOR network with additional VirtualScan Inputs to reduce coverage loss
Example Virtual Scan Broadcaster Using a MUX Network

Broadcaster using an example MUX network with additional VirtualScan inputs that can also be connected to data pins of the multiplexers.
Comparison

Encoding flexibility among combinational decompression schemes

System-on-Chip Test Architectures

Ch. 2 – Digital Test Architectures - P. 80
Circuits for Test Response Compaction

- Performed at the output of scan chains
- To reduce the amount of test response
- Grouped into three categories
  - Space compaction
  - Time compaction
  - Mixed space and time compaction
Space Compaction

- Space compactor is combinatorial
- Inverse procedure of linear expansion
- Compaction Techniques
  - X-Compact
  - X-Blocking
  - X-Masking
  - X-Impact
X-tolerant Response Compaction

An X-compactor with 8 inputs and 5 outputs
X-compactor

- **Theorem 2.1**
  - If only a single scan chain produces an error at any scan-out cycle, the X-compactor is guaranteed to produce errors at the X-compactor outputs at that scan-out cycle, if and only if no row of the X-compact matrix contains all 0’s.

- **Theorem 2.2**
  - Errors from any one, two, or an odd number of scan chains at the same scan-out cycle are guaranteed to produce errors at the X-compactor outputs at that scan-out cycle, if every row of the X-compact matrix is nonzero, distinct, and contains an odd number of 1’s.
X-Blocking (X-Bounding)

- Block X’s before reaching the response compactor
- Scan design rule checker for identifying potential X-generators

Impact
- No X’s will be observed
- Fault coverage loss
- Add area overhead
- May impact delay due to the inserted logic
X-Masking

Mask off X’s right before the response compactor

An example X-masking circuit
X-Impact

Handling of X-Impact
X-Impact

Handling of Aliasing
Time compaction

- Uses sequential logic to compact test response
- No unknown (X) values are allowed to reach the compactor; otherwise X-bounding, X-masking must be employed.
- MISR is most widely used
Mixed Time and Space Compaction

- Combine the advantages of a time compactor and a space compactor but with high area overhead
- Examples of mixed time and space compactors
  - OPMISR
  - Convolutional Compactor
  - q-compactor  
    - No feedback path
q-compactor

An example q-compactor with single output
Low-Power Test Compression Architectures

- **Low-Power architectures**
  - The Bandwidth-match low-power scan design can be used for test compression

- **An Example – The UltraScan Architecture**
  - Time-Division Demultiplexer (TDDM)
  - Time-Division Multiplexer (TDM)
  - Clock Controller
  - The TDDM/TDM circuit operates at 10 MHz and slow down the shift clock frequency to 1 MHz resulting in 10X reduction in shift power dissipation
UltraScan
# Summary of Industry Practices for Test Compression

<table>
<thead>
<tr>
<th>Industry Practices</th>
<th>Stimulus Decompressor</th>
<th>Response Compactor</th>
</tr>
</thead>
<tbody>
<tr>
<td>XOR Compression or OPMISR+</td>
<td>Combinational XOR network or Fanout Network</td>
<td>XOR network with or Without MISR</td>
</tr>
<tr>
<td>TestKompress</td>
<td>Ring generator</td>
<td>XOR network</td>
</tr>
<tr>
<td>VirtualScan</td>
<td>Combinational logic network</td>
<td>XOR network</td>
</tr>
<tr>
<td>DFT MAX</td>
<td>Combinational MUX network</td>
<td>XOR network</td>
</tr>
<tr>
<td>ETCompression</td>
<td>(Reseeding)PRPG</td>
<td>MISR</td>
</tr>
<tr>
<td>UtralScan</td>
<td>TDDM</td>
<td>TDM</td>
</tr>
</tbody>
</table>
### Summary of Industry Practices for At-Speed Delay Fault Testing

<table>
<thead>
<tr>
<th>Industry Practices</th>
<th>Skewed-Load</th>
<th>Double-Capture</th>
</tr>
</thead>
<tbody>
<tr>
<td>XOR Compression or OPMISR+</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>TestKompress</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>VirtualScan</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>DFT MAX</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>ETCompression</td>
<td>✓</td>
<td>Through service</td>
</tr>
<tr>
<td>UtralScan</td>
<td>✓</td>
<td>✓</td>
</tr>
</tbody>
</table>
Random-Access Scan Design

- Eliminate problems in serial scan mode
  - Excessive dynamic power during capture
  - Difficult fault diagnosis

- Scan cell randomly and uniquely addressable
  - Similar to storage cell in random-access memory (RAM)

- Impacts
  - Low shift power dissipation with an increase in routing overhead
  - Combinational logic diagnosis techniques for locating faults
Random-Access Scan Architectures

- **Traditional Random-Access Scan (RAS) Architecture**
  - All scan cells are organized into a two-dimensional array
  - Advantage
    - Can reduce shift power dissipation
  - Disadvantages
    - No guarantee to reduce test application time or test data volume
    - High area overhead

- **Progressive Random-Access Scan Design (PRAS)**
  - Use a structure similar to SRAM or a grid-addressable latch

- **Shift-Addressable Random-Access Scan Design (STAR)**
Traditional RAS Architecture

Access to a scan cell by decoding a full address with a row decoder (X) and a column decoder (Y)
Traditional RAS Scan Cell

Traditional Scan Cell Design
Broadcast the external SI port to all scan cells, cause routing problem

Toggle Scan Cell Design
Require a clear mechanism to reset all scan cells prior to testing
In normal mode, $RE$ is set to 0, forcing each scan cell to act as a normal $D$ flip-flop.

In test mode, $RE$ is set to 0 and a pulse is applied on clock $\Phi$.

To read out, clock $\Phi$ is held at 1, $RE$ for the selected scan cell is set to 1, and the content of the scan cell is read out through the bidirectional scan data signals $SD$ and $SD_\bar{}$.

To write or update a scan value into the scan cell, clock $\Phi$ is held at 1, $RE$ for the selected scan cell is set to 1, and the scan value and its complement are applied on $SD$ and $SD_\bar{}$, respectively.
Rows are enabled in a fixed order. It is only necessary to supply a column address to specify which scan cell in an enabled row to access.
PRAS Test Procedure

for each test vector $v_i$ ($i = 1, 2, \ldots, N$) {
    /* Test stimulus application */
    /* Test response compression */
    enable TM;
    for each row $r_j$ ($j = 1, 2, \ldots, m$) {
        read all scan cells in $r_j$ / update MISR;
        for each scan cell SC in $r_j$
            /* $v(SC)$: current value of SC  */
            /* $v_i(SC)$: value of SC in $v_i$ */
            if $v(SC) \neq v_i(SC)$
                update SC;
    }
    /* Test response acquisition */
    disable TM;
    apply the normal clock;
}
scan-out MISR as the final test response;
Use only one row (X) decoder and support two or more SI and SO ports.

All rows are enabled (selected) in a fixed order one at a time by rotating a 1 in the row enable shift register.

When a row is enabled, all columns (or scan cells) associated with the enabled row are selected at the same time.
STAR Test Procedure

for each test vector $v_i$ ($i = 1, 2, \ldots, N$) {
    /* Test stimulus application */
    /* Test response compression */
    enable TM;
    for each row $r_j$ ($j = 1, 2, \ldots, m$) {
        read all scan cells in $r_j$ / update MISR;
        /* Update selected rows */
        update all scan cells in $r_j$;
    }
    /* Test response acquisition */
    disable TM;
    apply the normal clock;
}
scan-out MISR as the final test response;
Test Compression RAS Architecture

- RAS design is effective in reducing shift power dissipation
- RAS is achieved by reducing power at the cost of increased area and routing overhead
- RAS cannot significantly reduce test data volume and test application time
- Test compression schemes are applicable for RAS design
STAR Compression Architecture

A decompressor is used to decompress the ATE-supplied stimuli.

A compactor is used to compact the test responses.
Reconfigured STAR Compression

The multiplexer allows transmitting scan-in stimulus from one column to another column.

The AND gate enables or disables the scan-out test response on the column to be fed to the compactor in serial scan mode.
At-Speed RAS Architectures

- **Major advantages of RAS design**
  - Significant shift power reduction
  - Facilitating fault diagnosis

- **Additional benefit for at-speed delay fault testing**
  - Launch-on-shift (a.k.a. skewed-load)
  - Launch-on-capture (a.k.a. double-capture)

- **Enhanced-scan based at-speed RAS Design**
  - Maximize delay fault detection capability
  - Long vector count problem
At-Speed RAS Architectures

- Approaches to overcoming long vector count problem
  - Using Enhanced-scan-based at-speed RAS architecture
  - Using Conventional launch-on-capture schemes

- Launch-on-capture based at-speed RAS architecture
  - Allow multiple transitions on the initialization vector; thereby reducing the vector count.

- Hybrid at-speed RAS architecture
  - First generate transition fault tests using launch-on-capture
  - Then supplement the tests using enhanced scan

- Faster-than-at-speed RAS architecture
  - To catch small delay defects that escape traditional transition fault tests.
Concluding Remarks

- Scan and Logic built-in self-test (BIST) are two most widely used DFT techniques.
- ATPG can no longer guarantee adequate product quality; at-speed delay testing and test compression become a requirement for 90-nanometer designs and below.
- Physical failures can escape detection of ATPG; logic BIST and low-power testing are gaining more industry acceptance in VLSI designs at 65-nanometer and below.
- Challenges lie ahead whether pseudo-exhaustive testing will become a preferred BIST pattern generation technique and random-access scan will be a promising DFT technique for test power reduction.