Digital Testing: Scan-Path Design

Outline
- Problems with sequential testing
- What is scan
- Types of scan
- Types of storage devices
- Scan Architectures
- Cost of Scan
- Partial Scan
- Stitching flip-flops

Problems Impeding Testing
- Complexity testing sequential circuits due to feedback loops
- Placement of the circuit in a known state
- High chance for hazard, essential hazard
- Timing problems in general

What is Scan Design
- This DFT technique is used mainly for synchronous circuit represented by the Huffman model presented in Chapter 3
- We assume the use of D-flip-flops only
- A mux is placed at the input of each flip-flop in such a way that all flip-flops can be connected in a shift register for one mux selection and to work in normal mode in the other

The Scan-path Technique
Scan-path design is to reduce test generation complexity for circuit containing storage devices and feedback path with combinational logic
The philosophy is to divide & conquer with the purpose to:
1. Set any internal state easily
2. Observe any state through a distinguishing sequence

Adding Scan Structure

Not shown: CK or MCK/SCK feed all SFFs (scan flip-flops).
Testing strategy

Example: Principle of scan path

- The combinational part is partitioned:
  - Each input to the FF is considered an output to the circuit
  - Each output of the FF is an input to the circuit
- Connect the FF in a shift register and test them
- Test the combinational part

Testing the Combinational Part

- Repeat until all patterns are applied.
  - a. Set \( SE = 1 \), shift in the initial values on the flip-flops. (These are the signals at the output of the latches for the first test pattern.)
  - b. \( SE = 0 \), apply a pattern at the primary inputs.
  - c. Clock the circuit once and observe the results at the primary outputs.
  - d. Clock the circuit \( M \) times.
- End repeat.

Testing strategy

Example:
A realization for the double-throw switch

Internal state can be determined via the scan-out output.

Testing strategy

The procedure for circuit testing:

1. Set \( c = 1 \) to switch the circuit to shift register mode
2. Check operation as a shift register by using scan-in inputs, scan-out output and the clock
3. Set the initial state of the shift register
4. Set \( c = 0 \) to return to normal mode
5. Apply test input pattern to the combinational logic
6. Set \( c = 1 \) to return to shift register mode
7. Shift out the final state while setting the starting state for the next test

An Example

Combination Circuit

- F, Y1, Y2, Z
- D flip-flops
- Clock inputs
- Primary inputs and outputs

How Scan DFT Works

- The combinational part is partitioned:
  - Each input to the FF is considered an output to the circuit
  - Each output of the FF is an input to the circuit
- Connect the FF in a shift register and test them
- Test the combinational part
Inserting the Muxes

Combinational Circuit

Types of Storage Devices
- Multiplexed input flip-flop
- Two-port flip-flop works with two nonoverlapping clocks
- Latch-based Scan Design: requires
  - 2-latches clocked with non-overlapping clocks
  - 3-latch clocked with three phases

Scan Design Architectures
- Several architectures:
  - Multiplexed flip-flops design
  - Level-sensitive scan design
  - Scan set scan design
- Derivative of scan design:
  - Parallel scan chains
  - Partial scan

Level Sensitive Scan Design LSSD (IBM)
- Level sensitive means that state changes in FSM are independent of delays nor order of changes in input signals (if inputs are set to new values)
- Scan is ability to shift into or out of any state
  - All internal storage is implemented using hazard free polarity-hold switches

Level-sensitive Latch
- The latch works with the 3 phases A, B and C
- For normal operation, clocks B and C
- For shift operation, clocks B and A
- Two-port flip-flop works with two non-overlapping clocks

Polarity Hold Latch (IBM)
**LSSD design rules**
- Storage is implemented by hazard free polarity-hold latches
- Latches controlled by non-overlapping clocks
- Clock inputs to SRLs must be easily controlled
- Clock feeds clock inputs only
- All SRL must be connected into shift registers
- Input sensitizing conditions

**LSSD: Testing**
Repeat until all patterns are applied.
- a. Apply a pattern at the primary inputs.
- b. Clock C; then clock B once and observe the results at the primary outputs.
- c. Shift out the response

Apply the initialization for the next pattern at Si.
Clock A, then clock B, M times.
Observe at the primary outputs and the SO pins.

**Multiple Scan Chains**
- Instead of stringing all the flip-flops or the latches in one shift register
- Partition them in several chains
- The advantages are:
  - Compatible with multiple clock designs
  - Shorten test application time
  - Simplify the stitching of the flip-flops
  - But, may require extra pins

**The Cost of Scan Design**
- Area (muxes, extra routing)
- Additional I/Os
- Performance, delays within the flip-flops
- Heat when testing at speed

**Multiple Scan Registers**
- Scan flip-flops can be distributed among any number of shift registers, each having a separate scanin and scanout pin.
- Test sequence length is determined by the longest scan shift register.
- Just one test control (TC) pin is essential.
Scan Overhead

- IO pins: One pin necessary.
- Area overhead:
  - Gate overhead = \[4 \frac{n_g(n_g+10n_f)}{n_f} \times 100\% \]
    - Example:
      - \( n_g = 100 \) gates, \( n_f = 2 \) flip-flops, overhead = 6.7%.
- More accurate estimate must consider scan wiring and layout area.
- Performance overhead:
  - Multiplexer delay added in combinational path; approx. two gate-delays.
  - Flip-flop output loading due to one additional fanout; approx. 5-6%.

Hierarchical Scan

- Scan flip-flops are chained within subnetworks before chaining subnetworks.
- Advantages:
  - Automatic scan insertion in netlist
  - Circuit hierarchy preserved; helps in debugging and design changes
- Disadvantage: Non-optimum chip layout.

Scan Area Overhead

Linear dimensions of active area:
- \( X = \frac{(C + S)}{r} \)
- \( Y = Y_y = Y + Y(1-b)/T \)

Area overhead:
- \( XY' - XY = \left(1+\frac{(1+as)(1+\frac{r}{T})-1}{T}\right) \times 100\% \)
- \( \frac{1-b}{T} \times 100\% \)

Example: Scan Layout

- 2,000-gate CMOS chip
- Fractional area under flip-flop cells, \( s = 0.478 \)
- Scan flip-flop (SFF) cell width increase, \( a = 0.25 \)
- Routing area fraction, \( b = 0.471 \)
- Cell height in routing tracks, \( T = 10 \)
- Calculated overhead = 17.24%
- Actual measured data:
  - Scan implementation     Area overhead     Normalized clock rate
  - None                             0.0          1.00
  - Hierarchical                    16.93%      0.87
  - Optimum layout                11.90%       0.91

ATPG Example: S5378

<table>
<thead>
<tr>
<th></th>
<th>Original</th>
<th>Full-scan</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of combinational gates</td>
<td>2,781</td>
<td>2,781</td>
</tr>
<tr>
<td>Number of non-scan flip-flops (10 gates each)</td>
<td>179</td>
<td>0</td>
</tr>
<tr>
<td>Gate overhead</td>
<td>0.0%</td>
<td>15.66%</td>
</tr>
<tr>
<td>Number of faults</td>
<td>4,493</td>
<td>6,993</td>
</tr>
<tr>
<td>PIP0 for ATPG</td>
<td>35/49</td>
<td>214/228</td>
</tr>
<tr>
<td>Fault coverage</td>
<td>70.5%</td>
<td>88.1%</td>
</tr>
<tr>
<td>Fault efficiency</td>
<td>70.5%</td>
<td>100.0%</td>
</tr>
<tr>
<td>CPU time on SUN Ultra II, 200MHz processor</td>
<td>5,553 s</td>
<td>3 s</td>
</tr>
<tr>
<td>Number of ATPG vectors</td>
<td>414</td>
<td>585</td>
</tr>
<tr>
<td>Scan sequence length</td>
<td>414</td>
<td>105,062</td>
</tr>
</tbody>
</table>

Automated Scan Design
Timing and Power

- Small delays in scan path and clock skew can cause race condition.
- Large delays in scan path require slower scan clock.
- Dynamic multiplexers: Skew between TC and TC signals can cause momentary shorting of D and SD inputs.
- Random signal activity in combinational circuit during scan can cause excessive power dissipation.

Scan Summary

- Scan is the most popular DFT technique:
  - Rule-based design
  - Automated DFT hardware insertion
  - Combinational ATPG
- Advantages:
  - Design automation
  - High fault coverage; helpful in diagnosis
  - Hierarchical - scan-testable modules are easily combined into large scan-testable systems
  - Moderate area (~10%) and speed (~5%) overhead
- Disadvantages:
  - Large test data volume and long test time
  - Basically a slow speed (DC) test

Overview: Partial-Scan & Scan Variations

- Definition
- Partial-scan architecture
- Scan flip-flop selection methods
- Cyclic and acyclic structures
- Partial-scan by cycle-breaking
- Scan variations
- Scan-hold flip-flop (SHFF)
- Summary

Partial-Scan Definition

- A subset of flip-flops is scanned.
- Objectives:
  - Minimize area overhead and scan sequence length, yet achieve required fault coverage
  - Exclude selected flip-flops from scan:
    - Improve performance
    - Allow limited scan design rule violations
  - Allow automation:
    - In scan flip-flop selection
    - In test generation
    - Shorter scan sequences

Partial-Scan Architecture

What & Why Partial Scan Design

- To scan only a subset of the flip-flops
  - The circuit is easier to test by the sequential ATPG.
  - The area overhead is minimized.
  - The placement of the flip-flops is such that the interconnects are minimized.
  - The delays are shortened.
Partial Scan Design
  - To scan only a subset of the flip-flops
  - How to select this subset?
  - It is an NP-complete problem
  - Heuristics on graph model to select the minimum feedback vertex set (MFVS) to transform the FSM into an acyclic graph

Scan Flip-Flop Selection Methods
  - Testability measure based
    - Use of SCOAP: limited success.
  - Structure based:
    - Cycle breaking
    - Balanced structure
      + Sometimes requires high scan percentage
  - ATPG based:
    - Use of combinational and sequential TG

Cycle Breaking
  - Difficulties in ATPG
  - S-graph construction and MFVS problem
  - Test generation and test statistics
  - Partial vs. full scan
  - Partial-scan flip-flop

Difficulties in Seq. ATPG
  - Poor initializability
  - Poor controllability/observability of state variables.
  - Gate count, number of flip-flops, and sequential depth do not explain the problem.
  - Cycles are mainly responsible for complexity.
  - An ATPG experiment:

<table>
<thead>
<tr>
<th>Circuit</th>
<th>Number of gates</th>
<th>Number of flip-flops</th>
<th>Sequential depth</th>
<th>ATPG CPU s</th>
<th>Fault coverage</th>
</tr>
</thead>
<tbody>
<tr>
<td>TLC</td>
<td>355</td>
<td>21</td>
<td>14*</td>
<td>1,247</td>
<td>89.01%</td>
</tr>
<tr>
<td>Chip A</td>
<td>1,112</td>
<td>39</td>
<td>14</td>
<td>269</td>
<td>98.80%</td>
</tr>
</tbody>
</table>
* Maximum number of flip-flops on a PI to PO path

Benchmark Circuits

Cycle-Free Example
  - All faults are testable. See Example 8.6.
Relevant Results

Theorem 8.1: A cycle-free circuit is always initializable. It is also initializable in the presence of any non-flip-flop fault.

Theorem 8.2: Any non-flip-flop fault in a cycle-free circuit can be detected by at most $d_{seq} + 1$ vectors.

ATPG complexity: To determine that a fault is untestable in a cyclic circuit, an ATPG program using nine-valued logic may have to analyze $9^N_{ff}$ time-frames, where $N_{ff}$ is the number of flip-flops in the circuit.

A Partial-Scan Method

Select a minimal set of flip-flops for scan to eliminate all cycles.

Alternatively, to keep the overhead low only long cycles may be eliminated.

In some circuits with a large number of self-loops, all cycles other than self-loops may be eliminated.

The MFVS Problem

For a directed graph find a set of vertices with smallest cardinality such that the deletion of this vertex-set makes the graph acyclic.

The minimum feedback vertex set (MFVS) problem is NP-complete; practical solutions use heuristics.

A secondary objective of minimizing the depth of acyclic graph is useful.

A Partial-Scan Example

<table>
<thead>
<tr>
<th>Circuit: TLC</th>
<th>355 gates</th>
<th>21 flip-flops</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scan flip-flops</td>
<td>Max. cycle length</td>
<td>Depth</td>
</tr>
<tr>
<td>0</td>
<td>4</td>
<td>14</td>
</tr>
<tr>
<td>4</td>
<td>2</td>
<td>10</td>
</tr>
<tr>
<td>9</td>
<td>1</td>
<td>5</td>
</tr>
<tr>
<td>10</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>21</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Test Generation

Scan and non-scan flip-flops are controlled from separate clock PIs:

- Normal mode - Both clocks active
- Scan mode - Only scan clock active

Seq. ATPG model:

- Scan flip-flops replaced by Pi and Po
- Seq. ATPG program used for test generation
- Scan register test sequence, 001100... length $n_{ff} + 4$ applied in the scan mode
- Each ATPG vector is preceded by a scan-in sequence to set scan flip-flop states
- A scan-out sequence is added at the end of each vector sequence

Test length = $(n_{ATPG} + 2) n_{ff} + n_{ATPG} + 4$ clocks

Partial vs. Full Scan: S5378

<table>
<thead>
<tr>
<th>Original</th>
<th>Partial-scan</th>
<th>Full-scan</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of combinational gates</td>
<td>2,781</td>
<td>2,781</td>
</tr>
<tr>
<td>Number of non-scan flip-flops (10 gates each)</td>
<td>179</td>
<td>149</td>
</tr>
<tr>
<td>Number of scan flip-flops (14 gates each)</td>
<td>0</td>
<td>30</td>
</tr>
<tr>
<td>Gate overhead</td>
<td>0.0%</td>
<td>2.63%</td>
</tr>
<tr>
<td>Number of faults</td>
<td>4,603</td>
<td>4,603</td>
</tr>
<tr>
<td>PIPO for ATPG</td>
<td>35/49</td>
<td>65/79</td>
</tr>
<tr>
<td>Fault coverage</td>
<td>78.0%</td>
<td>91.7%</td>
</tr>
<tr>
<td>Fault efficiency</td>
<td>70.5%</td>
<td>99.5%</td>
</tr>
<tr>
<td>CPU time on SUN Ultra II 200MHz processor</td>
<td>5.53s</td>
<td>7.7s</td>
</tr>
<tr>
<td>Number of ATPG vectors</td>
<td>144</td>
<td>1,117</td>
</tr>
<tr>
<td>Scan sequence length</td>
<td>414</td>
<td>34,691</td>
</tr>
</tbody>
</table>
Flip-flop for Partial Scan

- Normal scan flip-flop (SFF) with multiplexer of the LSSD flip-flop is used.
- Scan flip-flops require a separate clock control.

Scan Variations

- Integrated and Isolated scan methods
  - Scan path: NEC 1968
  - Serial scan: 1973
  - LSSD: IBM 1977
  - Scan set: Univac 1977
  - RAS: Fujitsu/Amdahl 1980

Scan Set

Scan Set Applications

- Advantages
  - Potentially useable in delay testing.
  - Concurrent testing: can sample the system state while the system is running
    - Used in microrollback
- Disadvantages
  - Higher overhead due to routing difficulties

Random-Access Scan (RAS)

RAS Flip-Flop (RAM Cell)
**RAS Applications**

- Logic test: reduced test length.
- Delay test: easy to generate single-input-change (SIC) delay tests.

**Advantage:**
- RAS may be suitable for certain architecture, e.g., where memory is implemented as a RAM block.

**Disadvantages:**
- Not suitable for random logic architecture
- High overhead - gates added to SFF, address decoder, address register, extra pins and routing

---

**Scan-Hold Flip-Flop (SHFF)**

- The control input HOLD keeps the output steady at previous state of flip-flop.

**Applications:**
- Reduce power dissipation during scan
- Isolate asynchronous parts during scan test
- Delay testing

---

**Partial-Scan Summary**

- Partial-scan is a generalized scan method; scan can vary from 0 to 100%.
- Elimination of long cycles can improve testability via sequential ATPG.
- Elimination of all cycles and self-loops allows combinational ATPG.
- Partial-scan has lower overheads (area and delay) and reduced test length.
- Partial-scan allows limited violations of scan design rules, e.g., a flip-flop on a critical path may not be scanned.