Digital IC Test Research Using Custom Chips

Edward J. McCluskey & James Li

Computer Systems Laboratory
Departments of Electrical Engineering and Computer Science
Stanford University
http://crc.stanford.edu/
Production Test Issues

• Test Cost
  - Test Data Volume
  - ATE Cost
• Test Transparency (Test Escapes)
  - Test Thoroughness Metric (SSF?)
  - Timing Defects
• Functional vs. Structural Tests
Experiment Purpose

• Compare
  - Actual Production Defects
  - Fault Models
• How?
  - Collect
    • Defective chips
    • Test sets from fault models
Outline

- What we did
  - The Murphy chip experiment
  - Test set collection and application
- What we found out
- Defects
  - Fault models and test sets
- Reliability defects
- Conclusion
Murphy Chip

• Special purpose chip design
  - 5 different combinational circuit designs
    • 3 control logic designs
    • 2 data path designs
    • 4 copies of each CUT
  - Support (DFT) circuitry
Murphy Chip

• LSI Logic 150k CMOS Gate Array
  - (with Crosscheck™ embedded array)
• 25k gate design
• 120-pin Ceramic PGA package
  - 96 signal pins
• $L_{\text{eff}} = 0.7\mu\text{m}$
• Nominal supply voltage = 5 V
ELF 35 Chip

- LSI Logic G10P technology
- $L_{\text{eff}} = 0.35\,\mu\text{m}$
- 264k gates
- 272-pin Plastic BGA package
- 3.3V Nominal VDD
- 6 CUT designs
ELF 13 Chip

- LSI Logic G-flex technology
- $L_{\text{eff}} = 0.13 \mu m$
- 8 to 10 million gates
- 500 to 600 pins
- 4 CUT designs
Murphy Chip Package Test

• 1.4 million test patterns +
  - $2^{24}$ (16.8 million) exhaustive patterns
• 3 supply voltages
  - 1.7V, 2.5V, 5V
• 4 test speeds
  - “characterized” from Shmoo plot
  - 1/3 characterized
  - 1/30 characterized
  - 15% faster than characterized
Murphy Chip Package Test

• Tester
  - Advantest T6671E VLSI Test System
  - 125MHz clock rate

• Test Time: 10 minutes per die on average
  - 5 minutes per good die
  - 20 minutes per interesting die
  - 10 minutes average

• 5,491 dies probed, 309 dies packaged
Outline

• What we did
  - The Murphy chip experiment
  - Test set collection and application

• What we found out
  - Defects
    - Fault models and test sets

• Reliability defects

• Conclusion
Murphy Chip Test Sets

- 265 test sets
  - 162 single stuck fault based
  - 60 delay fault
  - 10 weighted random
  - 2 design verification vectors
  - 30 IDDQ
  - 1 exhaustive
Murphy Chip Test Sets

• Test sets modified from original test sets
  - Preceded by all-zero, all-one
  - Bit-wise complements
  - One bit shift
  - Reverse sequence
ELF35 Chip Test Sets

- 222 test sets
- 129 single stuck fault based
- 40 delay fault
- 4 weighted random
- 6 design verification vectors
- 32 IDDQ
- 6 exhaustive
- 5 BIST
Outline

• What we did
  - The Murphy chip experiment
  - Test set collection and application
• What we found out
  - Defects
    - Fault models and test sets
• Reliability defects
• Conclusion
How we got the defects

- Normal production defects
- No artificially inserted defects
Defects

• Two defect types
  - Hard defects
    • Failed chip — not functional
      • Specified operating conditions
  - Reliability defects (flaws)
    • Functional chip
    • Early life failure
Hard Defect Classification

- **Timing dependent**
  - Tester response depends on test speed
- **Sequence dependent**
  - Tester response depends on pattern sequence
- **TIC (Timing independent combinational)**
  - Neither sequence nor timing dependent
  - SSF match
    - Tester response matches SSF simulation
Murphy Chip Hard Defects

All Boolean Failures at Nominal Voltage

TIC Non-TIC

50

14* 25 11

Timing Dependent

116

Non-SSF

66

SSF

* 6 slow escapes

66

25

41
Murphy Chip Hard Defects

Fail Some Test Sets at Nominal Voltage

43 (TIC) 

23 (Non-TIC) 

7*10 

6 (Timing Dependent) 

* 6 slow escapes

20 (Non-SSF) 

14 (SSF)
ELF35 Chip Hard Defects

All Boolean Failures at Nominal Voltage

TIC

Non-TIC

SSF

Non-SSF

Timing Dependent

Sequence Dependent

* all slow escapes
ELF35 Chip Hard Defects

Fail Some Test Sets at Nominal Voltage

- TIC: 28 defects
  - SSF: 1 defect
  - Non-SSF: 13 defects
- Non-TIC: 14 defects
  - Timing Dependent: 3* defects (all slow escapes)
  - Sequence Dependent: 6 + 5 defects
Outline

• What we did
  - The Murphy chip experiment
  - Test set collection and application
• What we found out
  - Defects
  - Fault models and test sets
• Reliability defects
• Conclusion
Test Sets

- Fault model based
- Single stuck faults
- Transition faults
- Stuck-open faults
- Delay faults
- Other
- Design verification
- Exhaustive
Single Stuck Fault (SSF) Model

• Fixed value (0 or 1)
  - Any gate input or output
• Elementary gates only
  - NAND, NOR, AND, OR, INV.
• Complex gate
  - Pin fault SSF Model
SSF Test Sets

- 100 % fault coverage
- Less than 100 % fault coverage
- More than 100 % fault coverage
  - Multiple detect (N Detect)
- Pin SSF model
- Weighted random
Stuck-fault Test Questions

- Is the SSF model accurate?
- Are Stuck-at fault tests effective?
- Is 100% SSF coverage enough?
  - Does the ATPG tool matter?
  - Does speed matter?
- Is 100% SSF coverage enough?
  - What’s better?
- What is the penalty
  - For less than 100% coverage?
  - For using pin faults?
Stuck-fault Test Questions

• Is the SSF model accurate?
  - Not very

• Compare the output response
  - SSF simulations
  - Silicon results

• Only 41 of 116 (34+ %) defective chips
  - Match SSF model
Are Stuck-at Fault Tests Effective?

• 100% SSF coverage test sets
  - 114 of 116 defective ICs (400 dpm)
    - Detected by the best set
  - 110 of 116 defective ICs (1200 dpm)
    - Detected by the worst set

• Exhaustive test set
  - 116 of 116 defective ICs detected
Stuck-fault Test Questions

• Is 100% SSF coverage enough?
  - Maybe not
• Does the ATPG tool matter?
  - Definitely
• Does speed matter?
  - Yes
## 100% SSF Test

<table>
<thead>
<tr>
<th>Tool</th>
<th>Test Length</th>
<th>Characterized Speed</th>
<th>Escapes</th>
<th>1/3</th>
<th>1/30</th>
<th>Speed</th>
</tr>
</thead>
<tbody>
<tr>
<td>2A</td>
<td>313</td>
<td>3</td>
<td>5</td>
<td>7</td>
<td>9</td>
<td>7</td>
</tr>
<tr>
<td>1C</td>
<td>427</td>
<td>4</td>
<td>7</td>
<td>7</td>
<td>8</td>
<td>5</td>
</tr>
<tr>
<td>6A</td>
<td>547</td>
<td>3</td>
<td>7</td>
<td>7</td>
<td>6</td>
<td>4</td>
</tr>
<tr>
<td>5C</td>
<td>571</td>
<td>4</td>
<td>6</td>
<td>4</td>
<td>5</td>
<td>7</td>
</tr>
<tr>
<td>4C</td>
<td>603</td>
<td>2</td>
<td>4</td>
<td>5</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>3C</td>
<td>1000</td>
<td>2</td>
<td>0</td>
<td>2</td>
<td>16777K</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Exhaustive</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Escapes vs. Test Length (2)

- Tool 1 (C)
- Tool 2 (A)
- Tool 3 (C)
- Tool 4 (C)
- Tool 5 (C)
- Tool 6 (A)

1/3 Speed
Characterized Speed

Test Length

Escapes
Stuck-fault Test Questions

• Is 100% SSF coverage enough?
  - Maybe Not
  - Still 2 escapes

• What’s better?
  - Detecting each SSF more than once
    • Increase the chance to detect defects
## N-Detect Test Sets

<table>
<thead>
<tr>
<th>N</th>
<th>Test Length</th>
<th>Characterized Speed</th>
<th>1/3 Speed</th>
<th>1/30 Speed</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>313</td>
<td>3</td>
<td>5</td>
<td>7</td>
</tr>
<tr>
<td>2</td>
<td>671</td>
<td>2</td>
<td>5</td>
<td>7</td>
</tr>
<tr>
<td>3</td>
<td>981</td>
<td>0</td>
<td>3</td>
<td>5</td>
</tr>
<tr>
<td>4</td>
<td>1292</td>
<td>0</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>5</td>
<td>1605</td>
<td>0</td>
<td>3</td>
<td>5</td>
</tr>
<tr>
<td>7</td>
<td>2203</td>
<td>1</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>10</td>
<td>3022</td>
<td>0</td>
<td>2</td>
<td>6</td>
</tr>
<tr>
<td>12</td>
<td>3548</td>
<td>0</td>
<td>2</td>
<td>5</td>
</tr>
<tr>
<td>15</td>
<td>4396</td>
<td>0</td>
<td>2</td>
<td>5</td>
</tr>
</tbody>
</table>
Escapes vs. Test Length (4)

- Tool 1 (C)
- Tool 2 (A)
- Tool 3
- Tool 4 (C)
- Tool 5 (C)
- Tool 6 (A)
- Tool 6 (A) 3-detect

Characterized Speed

Test Length
Multiple Detect Example

- Miss the G-S short defect
  - Unless \((WXY)=(001)\) is used to detect \(Z/1\)

<table>
<thead>
<tr>
<th>W</th>
<th>X</th>
<th>Y</th>
<th>Z</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

G-S Short
Multiple Detect Example (II)

- **B=X for s.a.1**
  - B=1 for short between A & B

```
A=0
S=1
B=X
B=1
```
Stuck-fault Test Questions

- What is the penalty
- For less than 100% coverage?
Escapes vs. Coverage

- 1/30 slower
- 1/3 slower
- Characterized speed

Graph showing escapes vs. test length with different coverage percentages (80%, 90%, 95%, 98%, 99%, 100%).
What is the penalty

• For using pin faults?
## Fault Model: Pin vs. Gate

<table>
<thead>
<tr>
<th>Tool</th>
<th>Model</th>
<th>Length</th>
<th>Characterized Speed</th>
<th>1/3 Speed</th>
<th>1/30 Speed</th>
</tr>
</thead>
<tbody>
<tr>
<td>3 C</td>
<td>pin</td>
<td>857</td>
<td>3</td>
<td>1</td>
<td>7</td>
</tr>
<tr>
<td></td>
<td>gate</td>
<td>1000</td>
<td>2</td>
<td>5</td>
<td>7</td>
</tr>
<tr>
<td>4 C</td>
<td>pin</td>
<td>456</td>
<td>6</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>gate</td>
<td>603</td>
<td>2</td>
<td>4</td>
<td>5</td>
</tr>
<tr>
<td>5 C</td>
<td>pin</td>
<td>466</td>
<td>6</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>gate</td>
<td>571</td>
<td>4</td>
<td>6</td>
<td>8</td>
</tr>
</tbody>
</table>

Note: The table above shows the escape rates for different tool models and gate lengths, characterized by speed and escape rates in 1/3 and 1/30 speeds.
How about weighted random?

<table>
<thead>
<tr>
<th>Tool</th>
<th>Test Length</th>
<th>Characterized Speed</th>
<th>1/3 Speed</th>
<th>1/30 Speed</th>
</tr>
</thead>
<tbody>
<tr>
<td>2 A</td>
<td>313</td>
<td>3</td>
<td>5</td>
<td>7</td>
</tr>
<tr>
<td>4 C</td>
<td>671</td>
<td>2</td>
<td>4</td>
<td>5</td>
</tr>
<tr>
<td>Weighted Random</td>
<td>21900 A</td>
<td>2</td>
<td>7</td>
<td>NA</td>
</tr>
<tr>
<td></td>
<td>62900 C</td>
<td>2</td>
<td>7</td>
<td>NA</td>
</tr>
<tr>
<td>Exhaustive</td>
<td>1677K</td>
<td>0</td>
<td>5</td>
<td>5</td>
</tr>
</tbody>
</table>
What about

• Non SSF test sets?
  - Not very effective
## Non-SSF Test Sets

<table>
<thead>
<tr>
<th>Tool</th>
<th>Model</th>
<th>Characterized Speed</th>
<th>1/3 Speed</th>
<th>CUT Tested</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 C</td>
<td>Transition</td>
<td>3</td>
<td>7</td>
<td>116</td>
</tr>
<tr>
<td>2 A</td>
<td>Stuck Open</td>
<td>4</td>
<td>9</td>
<td>116</td>
</tr>
<tr>
<td>3 A</td>
<td>Gate Delay</td>
<td>9</td>
<td>11</td>
<td>78</td>
</tr>
<tr>
<td></td>
<td></td>
<td>8</td>
<td>9</td>
<td>78</td>
</tr>
<tr>
<td>4 A</td>
<td>Robust Path Delay</td>
<td>5</td>
<td>5</td>
<td>60</td>
</tr>
<tr>
<td></td>
<td></td>
<td>5</td>
<td>5</td>
<td>60</td>
</tr>
<tr>
<td>5 A</td>
<td>Robust Path Delay</td>
<td>3</td>
<td>5</td>
<td>60</td>
</tr>
<tr>
<td>6 C</td>
<td>Non Robust Path Delay</td>
<td>8</td>
<td>9</td>
<td>60</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2</td>
<td>5</td>
<td>60</td>
</tr>
<tr>
<td>7</td>
<td>Verification</td>
<td>6</td>
<td>7</td>
<td>56</td>
</tr>
</tbody>
</table>
At-Speed vs. 2-Pattern Delay Tests

- Close Call
  - At-speed defects 3 more defective ICs
Outline

• What we did
  - The Murphy chip experiment
  - Test set collection and application
• What we found out
  - Defects
  - Fault models and test sets
• Reliability defects
• Conclusion
Reliability Defects or Flaws

• Chip functional during test
• Early life failure
• Intermittent Failures
Reliability Defect Screens

- Voltage or Temperature Burn-in
- IDDq
- VLV
- minVdd
- SHOVE
Voltage or Temperature Burn-in

• **Voltage burn-in**
  - Effective for Oxide defects
  - Acceleration decreases at lower $V_{\text{DD}}$

• **Temperature burn-in**
  - Effective for Metal defects
  - **Misses low temperature sensitive defects**
    • Silicide opens
IDDq for Reliability Defects or Flaws

• Chip quiescent
  - Measure supply current
  - High current
    • Discard

• Variants
  - IDDt—dynamic IDD
  - Current Signature (sort by value)
  - Delta IDDq (difference method)
VLV for Reliability Defects or Flaws

• Lower $V_{DD}$
• Between 2 and 2.5 normal $V_T$
  - Scaled speed
• Apply test patterns
  - Discard failing chips
minVdd for Reliability Defects or Flaws

• Lower $V_{DD}$
  - Find minimum value
    • Chip still operates
• Discard outliers
SHOVE for Reliability Defects or Flaws

• Dynamic elevated voltage stress
• Called SHOVE, stress test
  - New reliability screen
  - Dynamic voltage screen
SHOVE for Reliability Defects or Flaws

• Raise $V_{DD}$
  - Apply patterns
    • $<$Collect response$>$

• Lower $V_{DD}$ to normal
  - Apply test patterns
    • Collect response
## Effectiveness of Test methods

<table>
<thead>
<tr>
<th>Failure mode</th>
<th>VLV</th>
<th>I&lt;sub&gt;DDQ&lt;/sub&gt;</th>
<th>Burn-in</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>T</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>V</td>
</tr>
<tr>
<td>V&lt;sub&gt;t&lt;/sub&gt; shift</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
</tr>
<tr>
<td>Gate oxide shots</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>Metal shorts</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>High Z interconnect</td>
<td>N</td>
<td>N</td>
<td>*</td>
</tr>
<tr>
<td>Tunneling opens</td>
<td>Y</td>
<td>*</td>
<td>N</td>
</tr>
</tbody>
</table>

* * defect dependent

* *
Issues
  - Values
  - Thresholds
  - Voltages
  - Replacing normal voltage test

Reliability Defects Screens
IDDQ & VLV Test Results (1530 CUTs)

Nominal Voltage Boolean Failures

VLV Boolean Failures

NV IDDQ Failures (3uA)
Reliability Screen Questions

- Can burn-in be avoided?
- How many IDDQ vectors are enough?
- How good are IDDQ test pattern sets?
Future Research

• Alternatives to AT-SPEED testing
• Alternatives to Burn In
  - New IDD Techniques
  - Very Low Voltage Techniques
  - New approaches to SHOVE testing
• BIST Techniques
• Fault models & fault grading capability
Outline

- What we did
  - The Murphy chip experiment
  - Test set collection and application
- What we found out
- Defects
  - Fault models and test sets
- Reliability defects
- Conclusion
Conclusions

• Single stuck fault model isn’t accurate
• Single stuck fault model is very useful
  - Generates good test sets
• Multiple-detect test sets are effective
  - Competitive length

• The Conventional Wisdom
  - Is often WRONG !!!