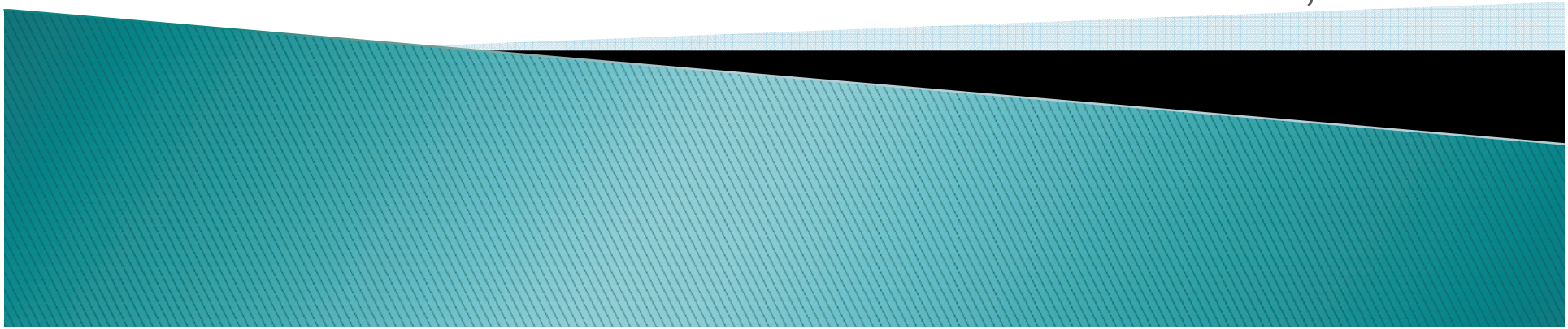
Design For Testability for Integrated Circuits

Luis Basto

Nov. 16, 2011



Outline of Presentation

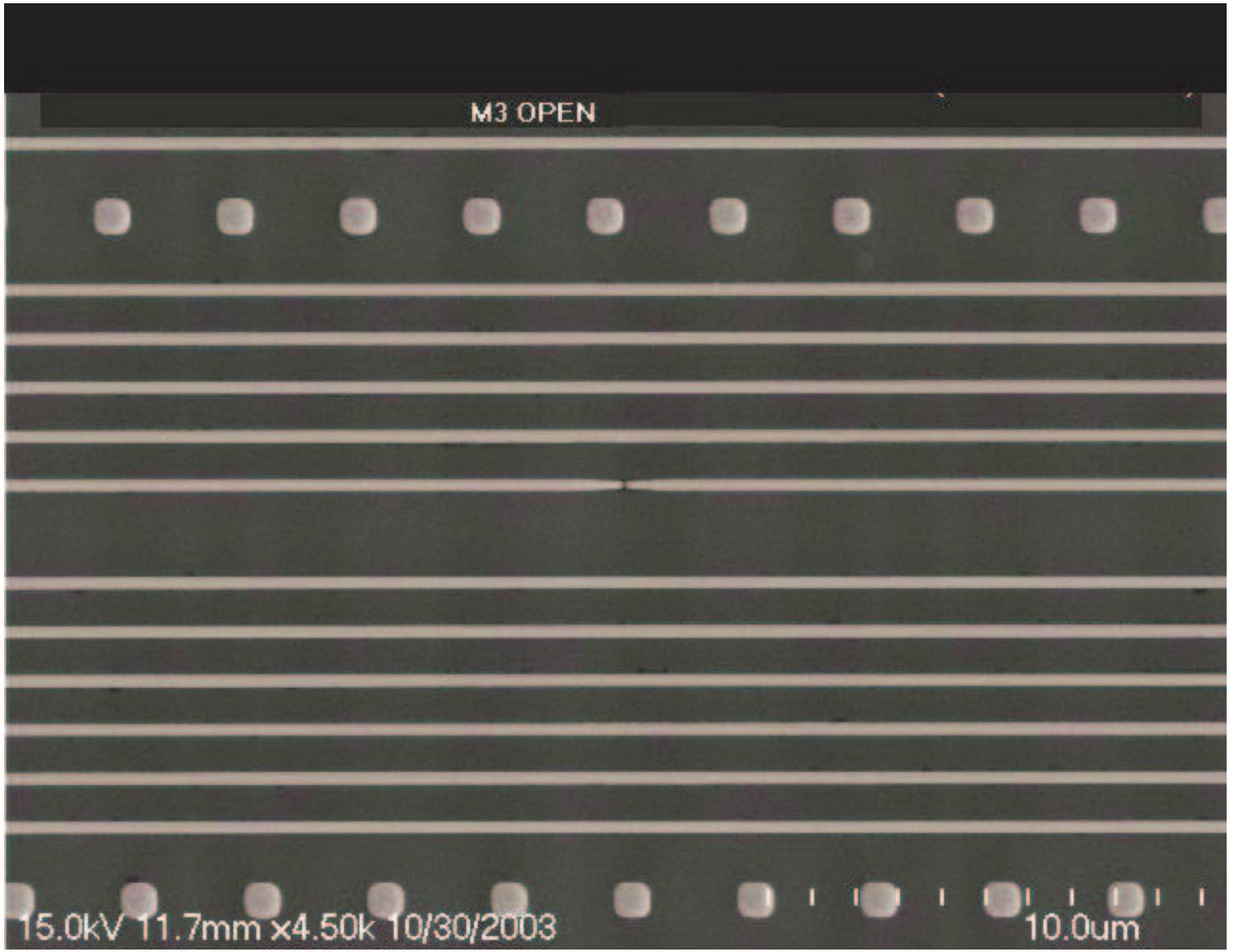
- Introduction
  - What is DFT?
  - Why do we need DFT?
  - Structural vs. functional testing
- Test Economics
  - Cost of test
- Basic Test Theory
  - Fault Models
  - Fault Coverages
- DFT Methodologies
  - Functional Tests
  - Scan design
  - ATPG
  - Memory BIST
  - Logic BIST
  - Boundary Scan (JTAG)

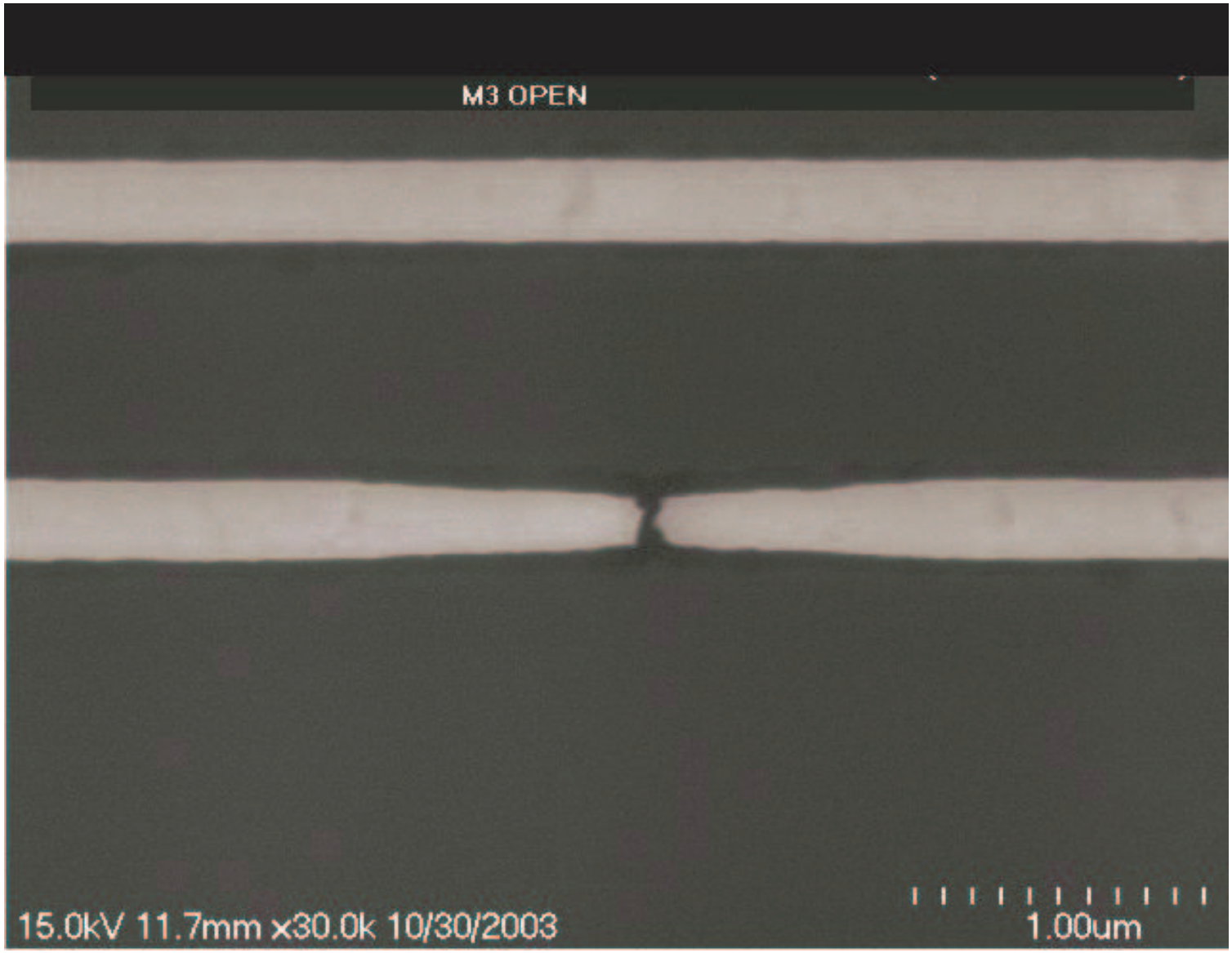
Outline of Presentation

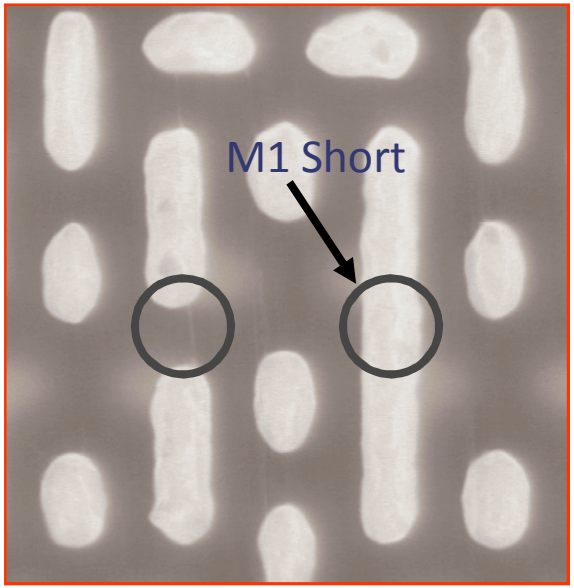
- Introduction
  - What is DFT?
  - Why do we need DFT?
  - Structural vs. functional testing
- Test Economics
  - Cost of test
- Basic Test Theory
  - Fault Models
  - Fault Coverages
- DFT Methodologies
  - Ad Hoc Techniques
  - Scan design
  - ATPG
  - Memory BIST
  - Logic BIST
  - Boundary Scan (JTAG)

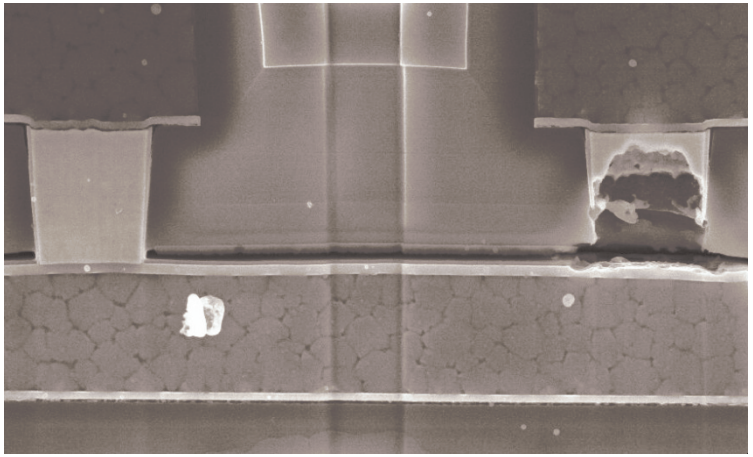
Introduction

- What is DFT?
- DFT from Google
  - DFTBA --- Don’t forget to be awesome
  - dft matrix --- Discrete Fourier transforms
  - dft interview questions --- Will discuss later
  - dft --- Design for Test or Design for Testability

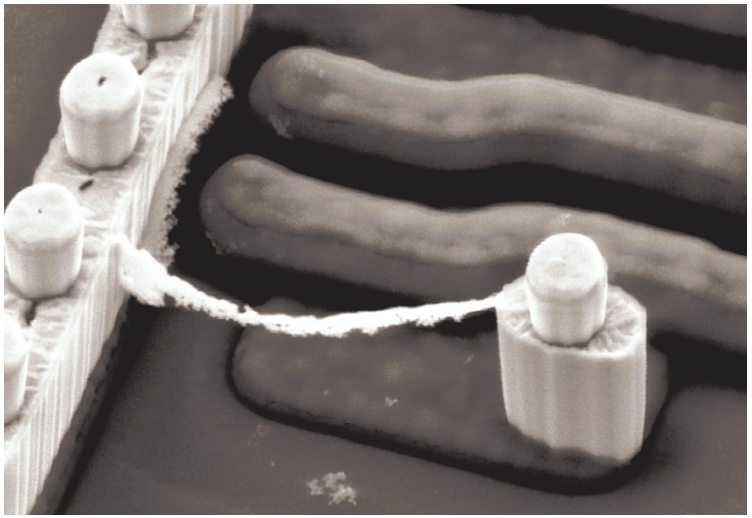

M3 OPEN
15.0kV 11.7mm x4.50k 10/30/2003
10.0um


M3 OPEN
15.0kV 11.7mm x30.0k 10/30/2003
1.00um

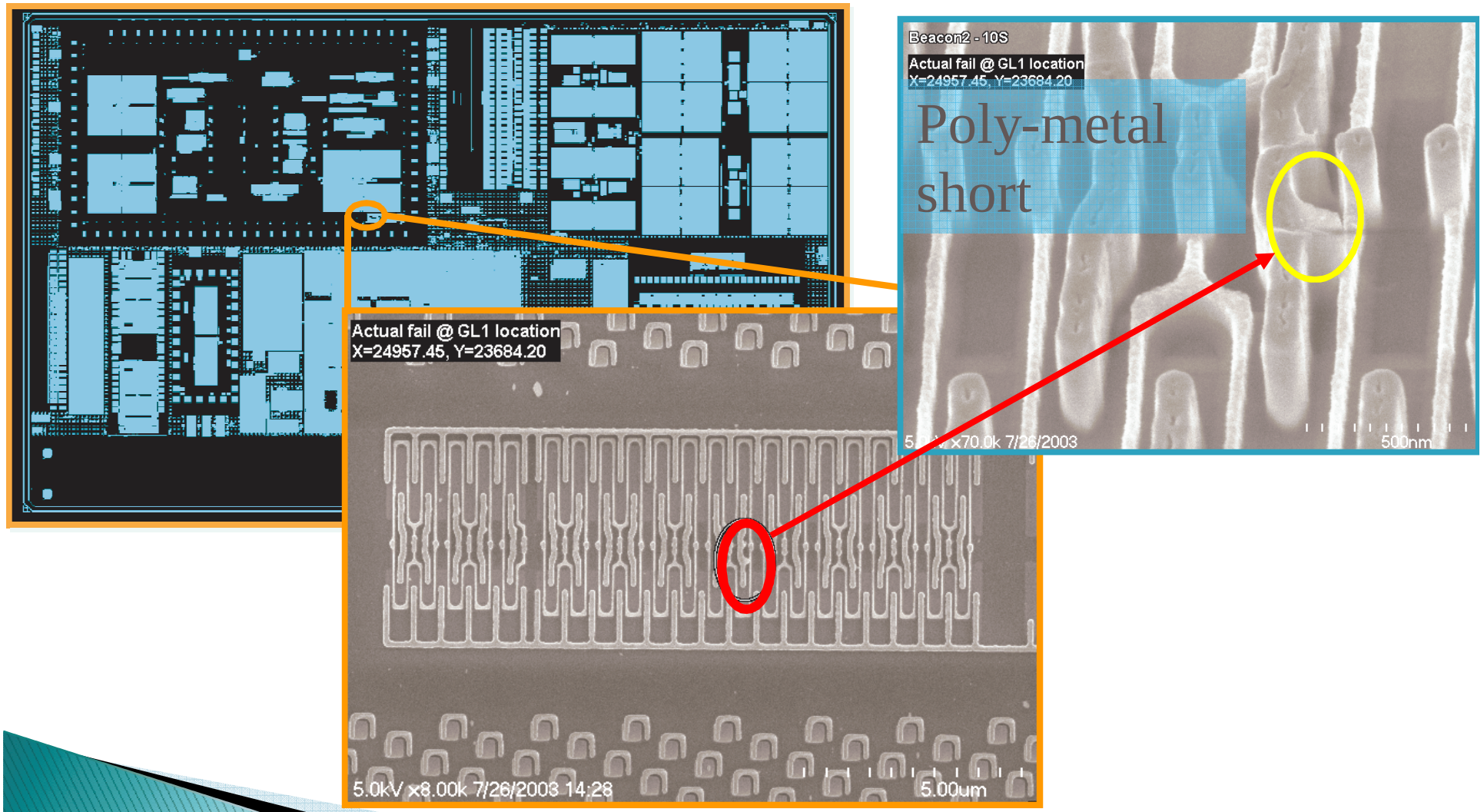

M1 Short



Resistive open



Resistive bridge


Actual fail @ GL1 location
X=24957.45, Y=23684.20
5.0kV x8.00k 7/26/2003 14:28
5.00um
Beacon2 - 10S
Actual fail @ GL1 location
X=24957.45, Y=23684.20
Poly-metal short
500nm

Issues with Traditional Test Flow

- DFT is usually added late in the design cycle, which may adversely affect area, timing, and power
- More iterations may be needed to fix RTL issues, timing, and physical implementation
- May cause more routing congestion
- When crunch comes, test is usually sacrificed to function

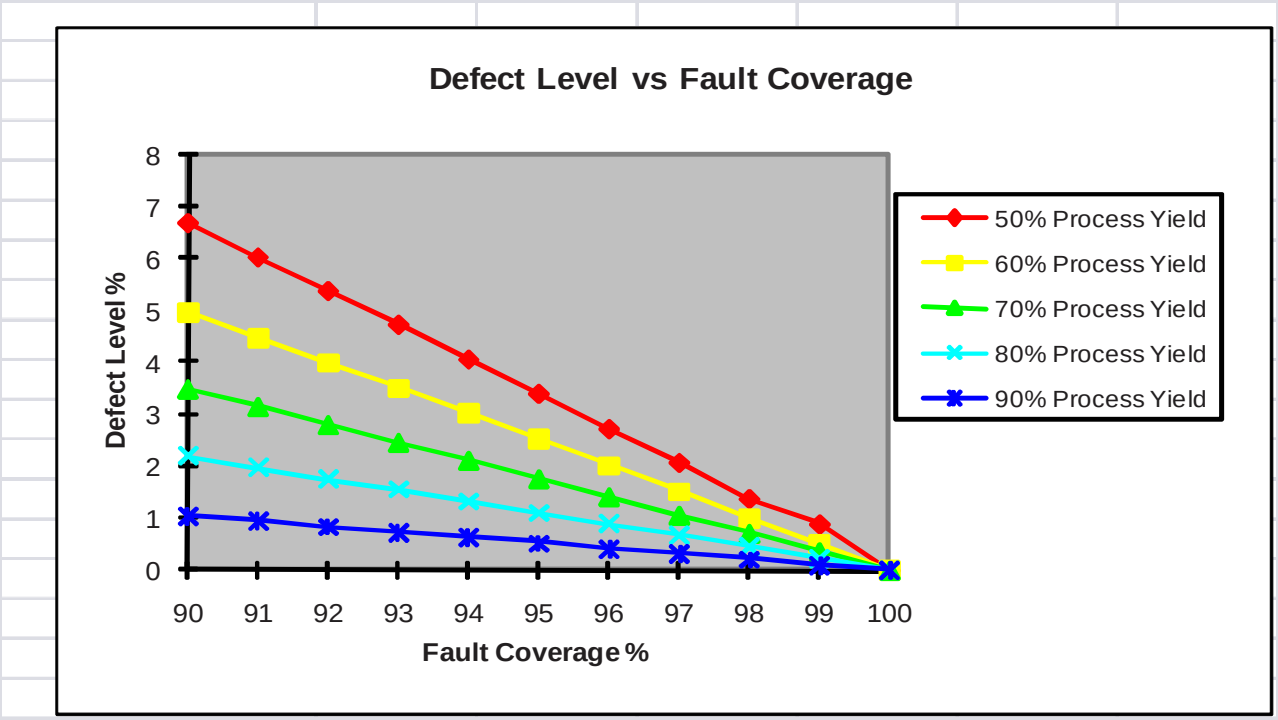
What is Design For Testability ?

"Testability is a feature that is either added or not added at design time."

Why Do We Need Design For Test?

- Product Quality
  - Reduce field returns
  - Improve yield
- Time to Market
  - Test automation
  - Hit your market window
- Test Cost
  - Reduce cost of testing
  - Improve your profits

Defect Level vs. Fault Coverage



$D = [1 - Y^{(1-T)}] \times 100$ D = defect level

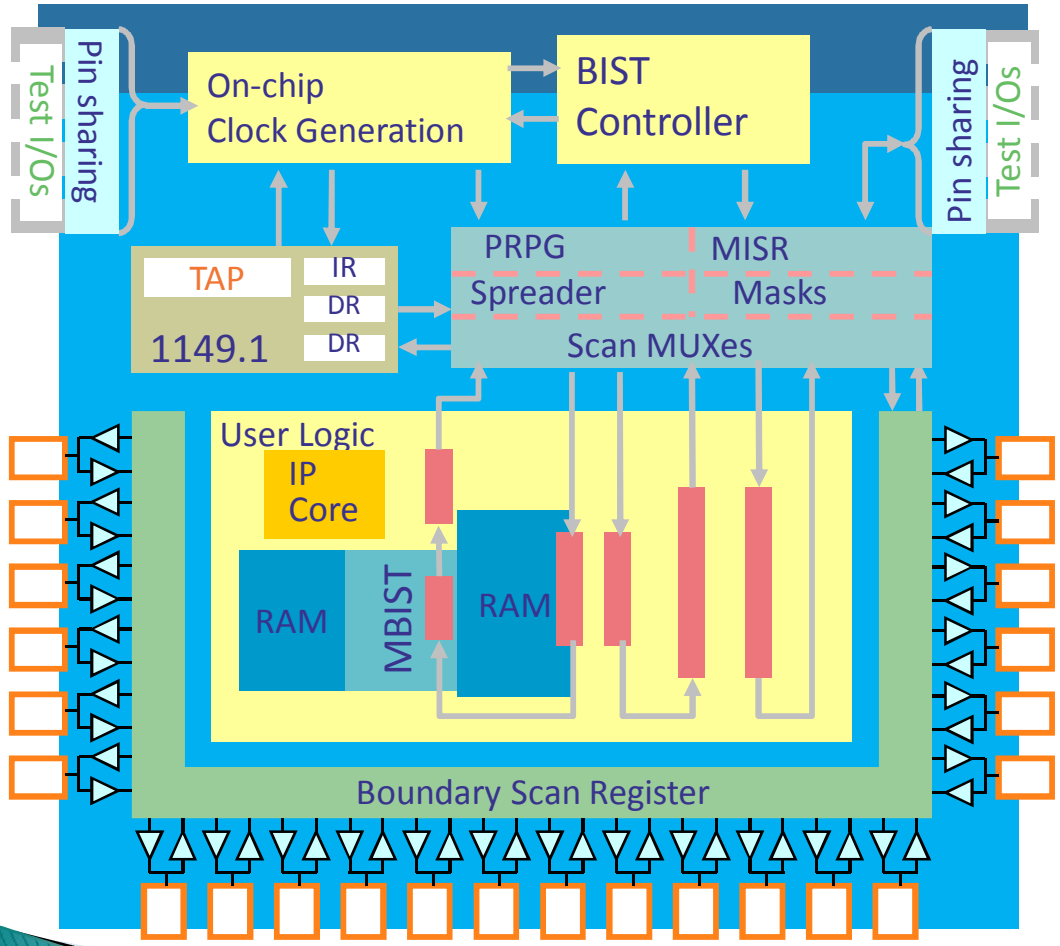
Y = process yield T = Fault coverage

Williams, IBM, 1981

Functional vs. Structural Test

- Functional Test
  - Designed to exercise the device's intended functionality
  - Usually requires manual test generation or conversion
  - Expensive to generate and run
  - Unknown test coverage
  - May be difficult to diagnose test failures
- Structural Test
  - Alters design to make it easier to test
  - Generates tests targeted at manufacturing defects
  - Automates test generation
  - Industry accepted test coverages
  - Facilitates diagnostics of failing devices

System-on-Chip DFT Requirements



- Hierarchical test insertion
- Multiple clock domains
- Mix of clock edges
- Scan chain balancing
- Embedded IPs and memories
- Memory Built-In Self-Test
- Logic Built-In Self-Test
- Internal core wrapping
- JTAG boundary scan

Outline of Presentation

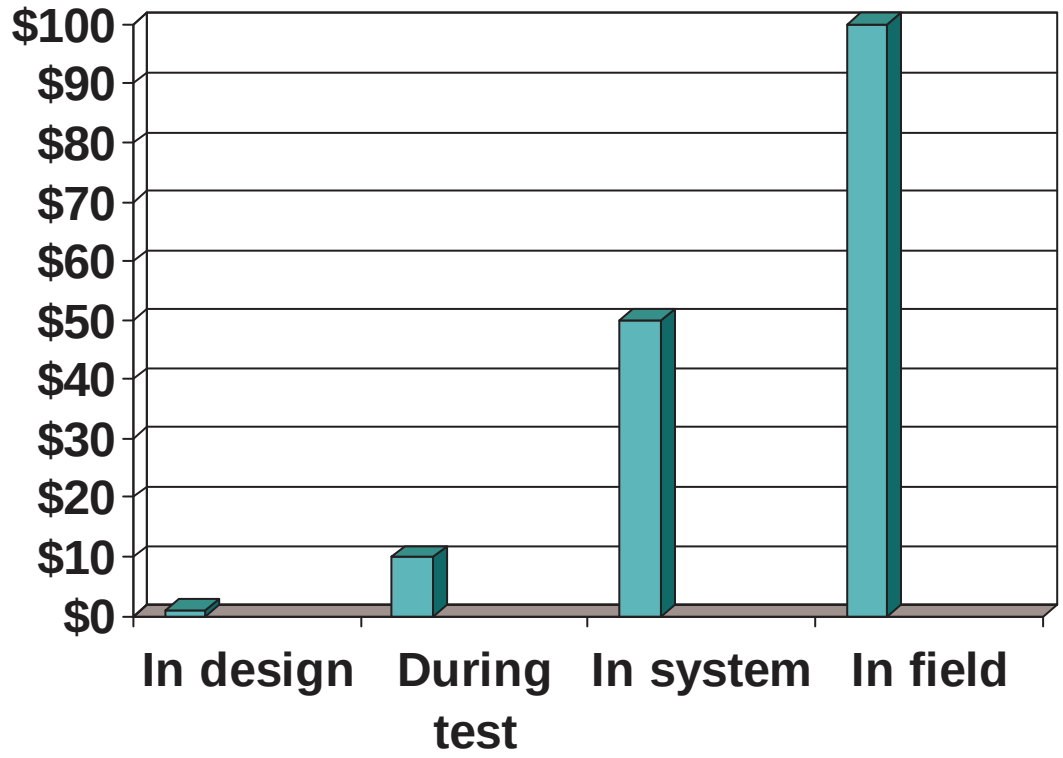
- Introduction
  - What is DFT?
  - Why do we need DFT?
  - Structural vs. functional testing
- Test Economics
  - Cost of test
- Basic Test Theory
  - Fault Models
  - Fault Coverages
- DFT Methodologies
  - Ad Hoc Techniques
  - Scan design
  - ATPG
  - Memory BIST
  - Logic BIST
  - Boundary Scan (JTAG)

DFT Reduces Test Cost

- DFT reduces test volume
  - Reduces ATE memory requirements
  - Reduces test time
  - Minimizes or eliminates tester reloads
  - Improves tester throughput
- DFT provides higher quality and lower cost

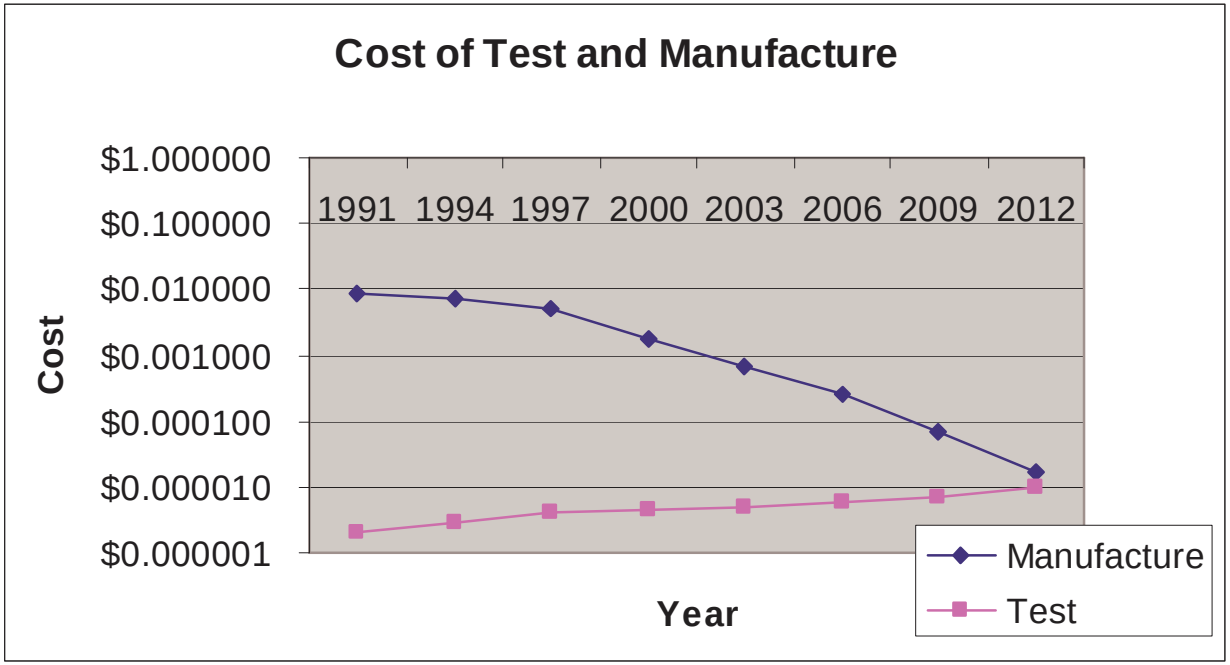
DFT Reduces Test Cost (cont'd)

- Cost increases exponentially with respect to where undetected defects are found

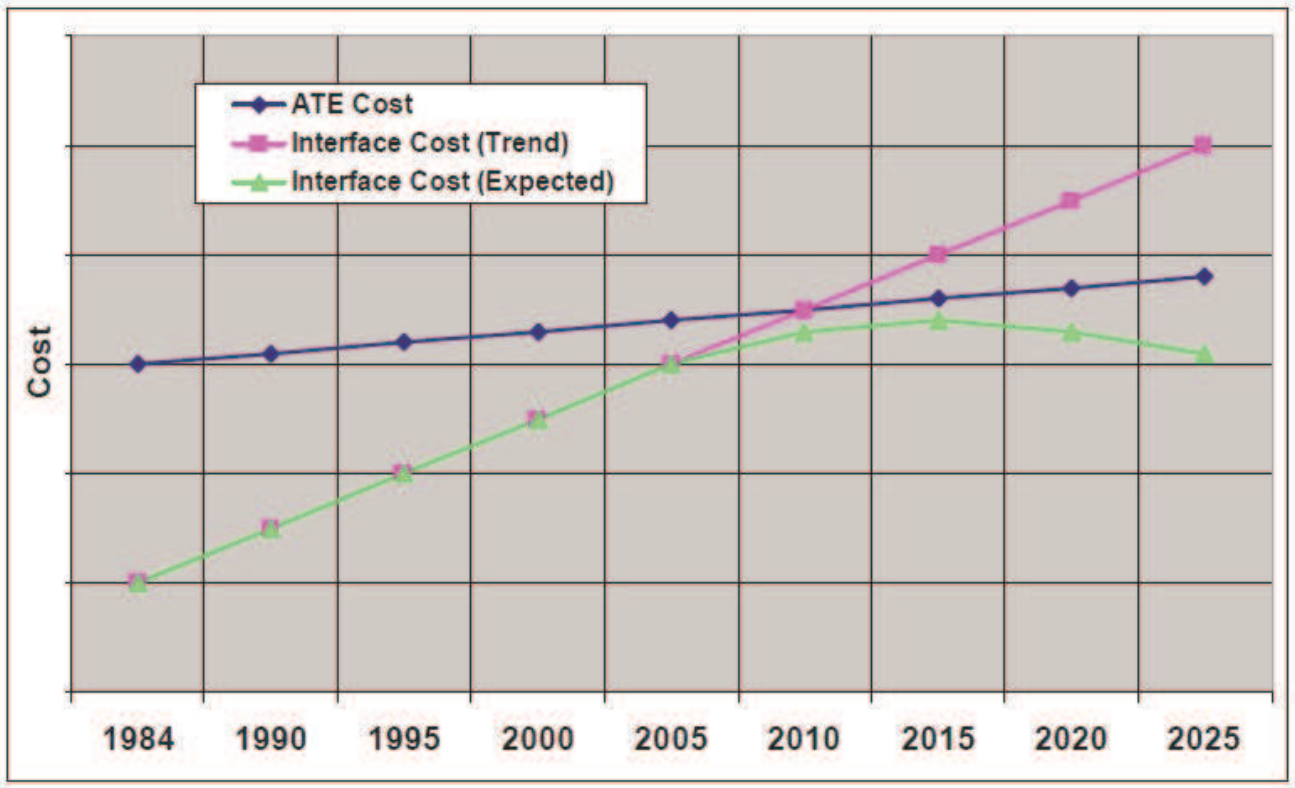


Cost of Test Increasing

- Chip operation at full speed depends on the manufacturing process
- According to Silicon Industry Association (SIA), cost to test will soon *equal* the cost to manufacture



Cost of Test Increasing (2)



Test cost/unit vs. Interface Cost.

International Technology Roadmap for Semiconductors (ITRS) 2009.

Outline of Presentation

- Introduction
  - What is DFT?
  - Why do we need DFT?
  - Structural vs. functional testing
- Test Economics
  - Cost of test
- Basic Test Theory
  - Fault Models
  - Fault Coverages
- DFT Methodologies
  - Ad Hoc Techniques
  - Scan design
  - ATPG
  - Memory BIST
  - Logic BIST
  - Boundary Scan (JTAG)

What is Manufacturing Test

- High volume production testing
- ATE generally targets pass/fail
- ATE runs a test program that serially steps through multiple tests
- Only those devices that pass EVERY test are shipped
- Devices that failed are either sent to failure analysis or discarded

What is a Physical Defect?

- A physical defect is a flaw introduced into a chip during manufacturing which makes the chip fails or behaves different from specification
- Some common physical defects
  - Surface effects and gate oxide breakdown
  - Metalization and electromigration
  - Impurities
  - Opens and shorts
  - Bridging

Fault Models

Fault models attempt to represent manufacturing defects.

A fault model is a logical model representing physical defects.

Types of fault models:

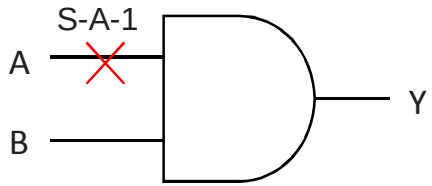
- Stuck-at fault - most common
- Bridging
- IDDQ - transistor shorts cause higher current
- Delay faults –
  - Transition delay
  - Path delay

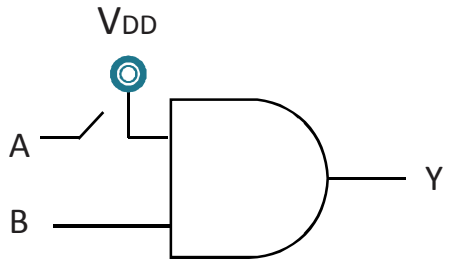
Stuck-at Fault Model 1/3

- Many physical defects behave like shorts to power or ground
- This have the appearance of being stuck-at logic 1 or stuck-at logic 0
- Faults only at input and output pins of logic gates
- Each pin of device modeled as stuck-at-1 (s-a-1) or stuck-at-0 (s-a-0)
- Test required to sensitize inputs to detect faults and propagate results to outputs

Stuck-at Fault Model 2/3

- Examples of logic gates for stuck faults
  - Input stuck fault





Good circuit

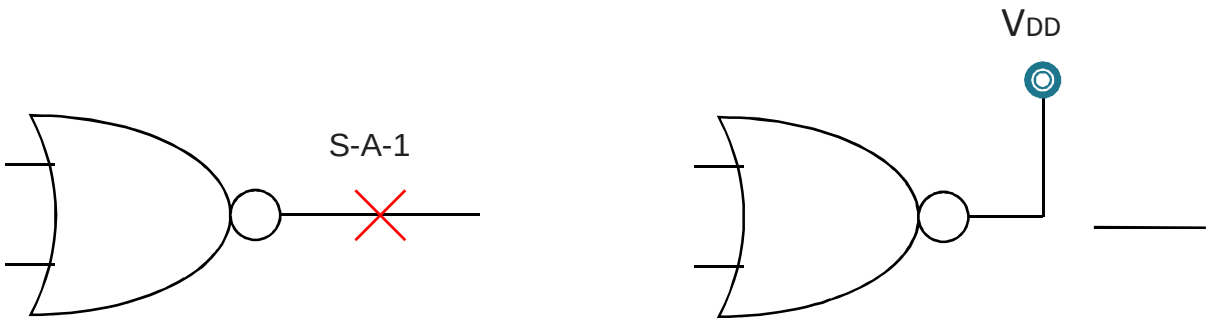
A	B	Y
0	0	0
0	1	0
1	0	0
1	1	1

Defective circuit

A	B	Y
0	0	0
0	1	1
1	0	0
1	1	1

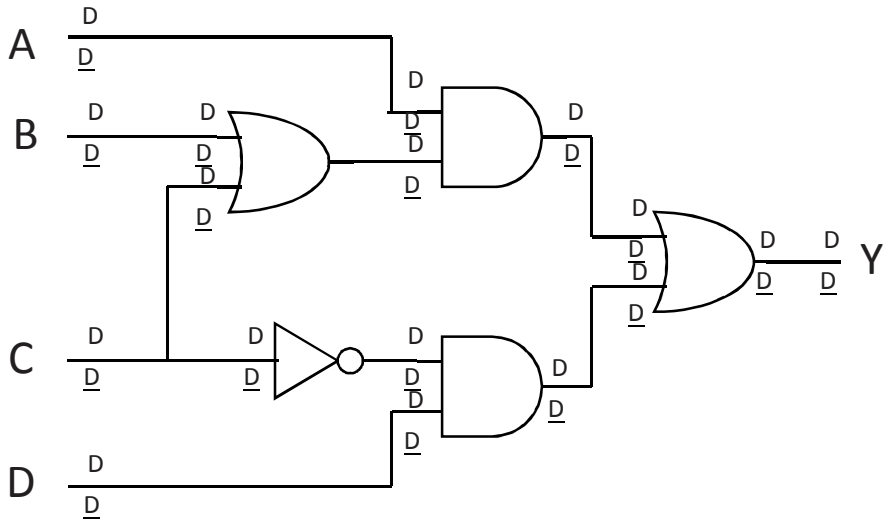
Stuck-at Fault Model 3/3

- Examples of logic gates for stuck faults
  - Output stuck fault



Number of Stuck-at Faults

- How many are there in a circuit?
- Considered only at the boundary of a logic gate or cell



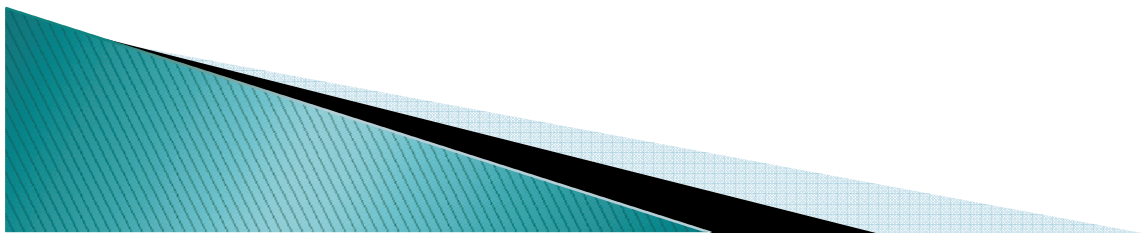
$N_{faults} = 2 (N_{pins} + N_{ports})$

Undetectable Faults

- Why can't we get 100% coverage?
- When there is fanout, we can no longer guarantee that all faults are detectable.
- Fault models are approximations.
- Many other reasons for undetectable faults.

Bridging Fault Model

- Defects that crosses two or more wires or nodes
- Needs physical layout information for correct modeling



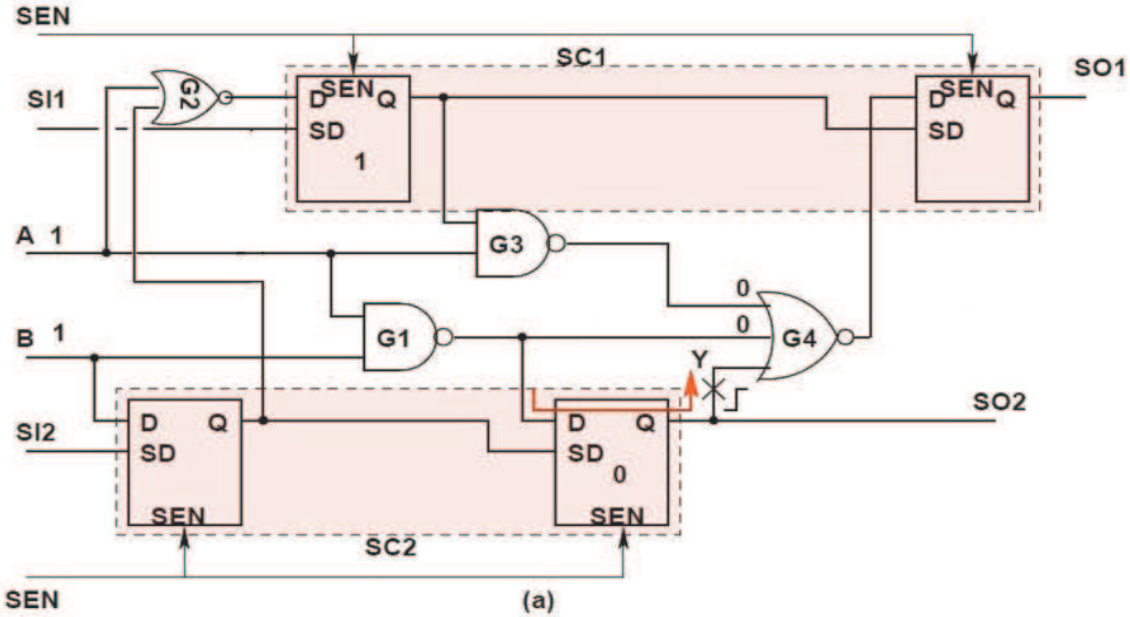
Iddq Fault Model

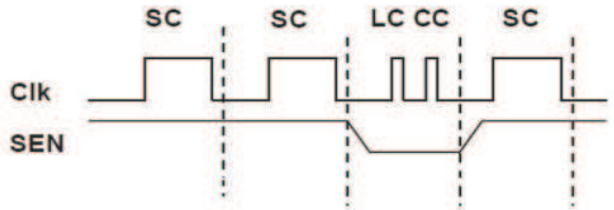
- Single SAF model cannot cover all defects
- Single SAF model is voltage-based
- Iddq fault model is current-based
- Physical defects such as bridging and stuck-on transistors cause a rise in static current
- Failure is detected by measuring current and noting that its level higher than fault-free threshold

Transition Fault Model

- At-speed test
- Model large delay defects
- Behaves as a temporary stuck-at fault
- Slow-to-rise and slow-to-fall model
- Two test patterns required - initialization vector and transition vector

Transition Fault Testing

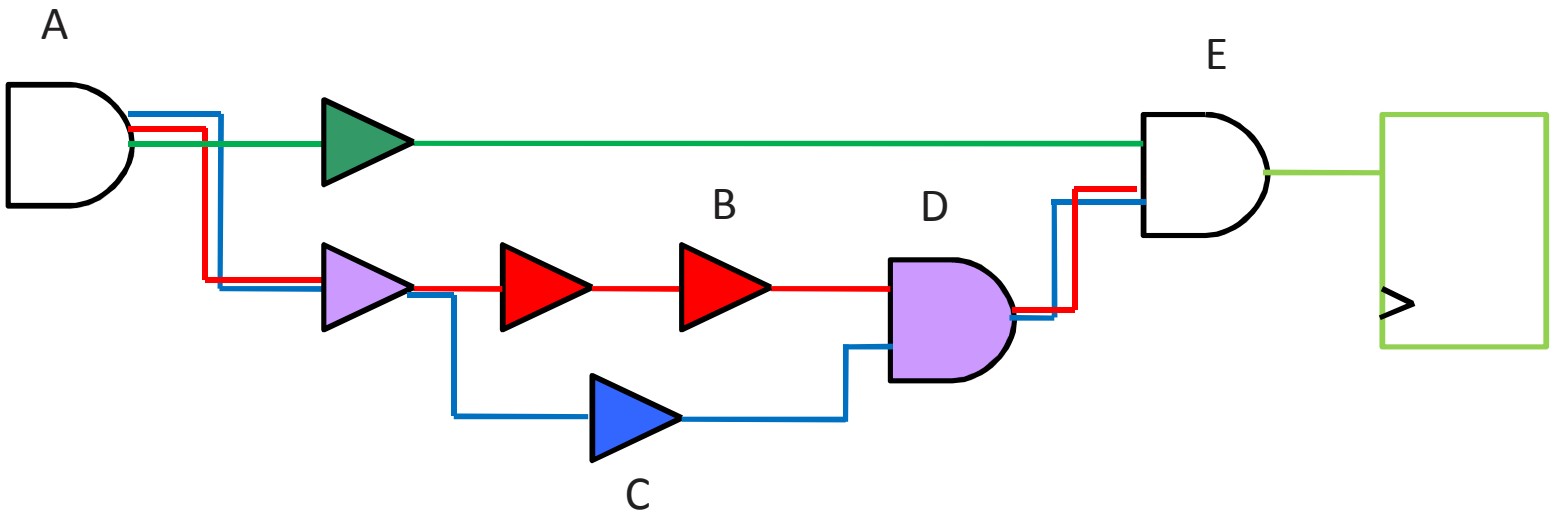




Path Delay Fault Model

- At-speed critical path test
- Targets specific paths in design
  - Path topology - launch and capture points are either scan registers or device pins
  - Rising or falling edge
- Two test patterns required – launch and capture vectors
- Disadvantage – large number of potential paths; grows exponential with size of design

Path Delay Fault Testing



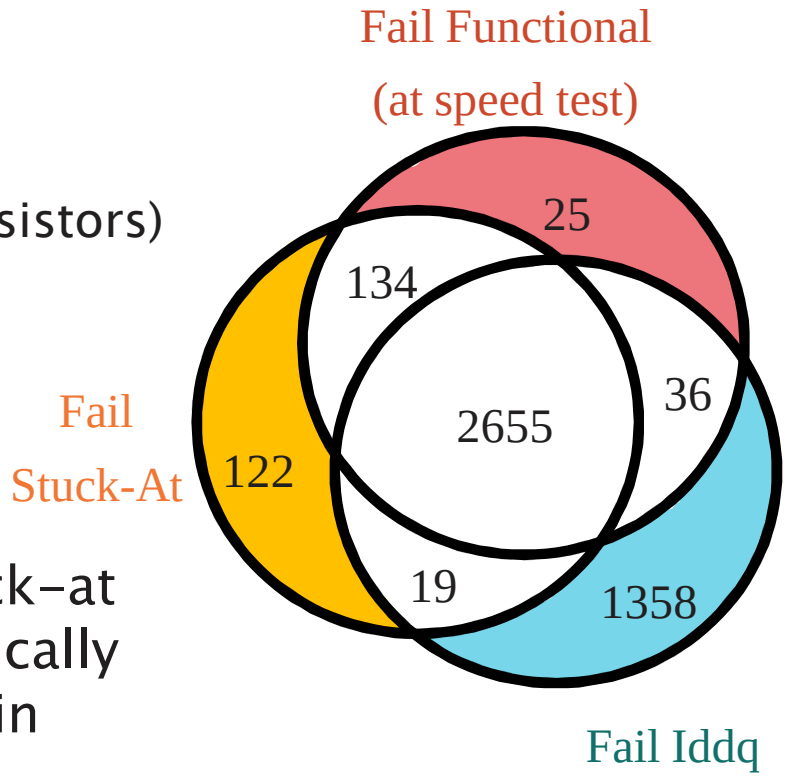
- Paths A – E (shortest)
- Paths A – B – D – E (longest)
- Paths A – C – D – E
- All of the above

Dynamic (AC) Fault Models Benefits

- Transition Fault Model
  - Finds gross dynamic defects not identified by stuck-at fault model
  - Similar to stuck-at model, targets entire chip
- Path Delay Model
  - Targets critical paths only
  - Used to find dynamic defects and for speed binning

Why multiple fault models?

- Test Chip* (experimental results)
  - Standard cell (8577 gates/36k transistors)
  - 436 state element (flip flops)
  - Test of 3 wafer lots, 26,415 die (excludes wafer test fails)



CONCLUSION

"… a combination of high static stuck-at coverage, Iddq tests and automatically generated timing tests will result in excellent quality levels…"

*P.C. Maxwell, R.C. Aitken, V. Hohansen, I. Chiang

Uncover defects not identified by Stuck-At fault model.

Outline of Presentation

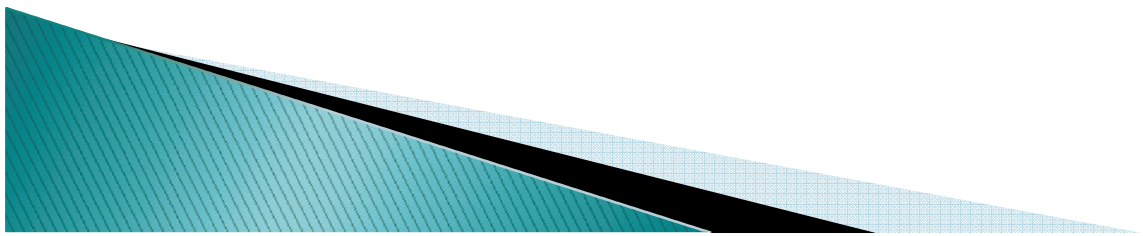
- Introduction
  - What is DFT?
  - Why do we need DFT?
  - Structural vs. functional testing
- Test Economics
  - Cost of test
- Basic Test Theory
  - Fault Models
  - Fault Coverages
- DFT Methodologies
  - Ad Hoc Techniques
  - Scan design
  - ATPG
  - Memory BIST
  - Logic BIST
  - Boundary Scan (JTAG)

DFT Methodologies

- Functional Testing
- Scan Design
- Compression
- ATPG
- Memory BIST
- Logic BIST
- Boundary Scan

Major EDA Vendors

- Synopsys
- Cadence Design Systems
- Mentor Graphics



Other Topics and Newer Challenges

- Core Wrapping
- Adaptive testing
- Small delay defects
- 3-D testing
- Debug and Diagnostics

DFT Careers

- Covers many disciplines
- Design knowledge
- Understands DFT concepts
- Manufacturing and test experience
- Scripting/automation experience

Job Openings in DFT – from LinkedIn

- Field Application Engineer - Noida
- Physical design/DFT/RTL/STA - Bangalore
- DFT Lead Engineer – Bangalore
- DFT Specialist – Pune and Noida
- DFT Lead Engineer – RIM, Waterloo
- Physical Design Engineer – RTL to GDSII – Hyderabad
- DFT Lead Architect – Bangalore
- DFT Engineer - Hyderabad

End of Presentation

Questions and Comments?