New Directions in Manufacturing Test

Jacob A. Abraham
Computer Engineering Research Center
The University of Texas at Austin

Shanghai Jiao Tong University

July 19, 2005

Research Areas

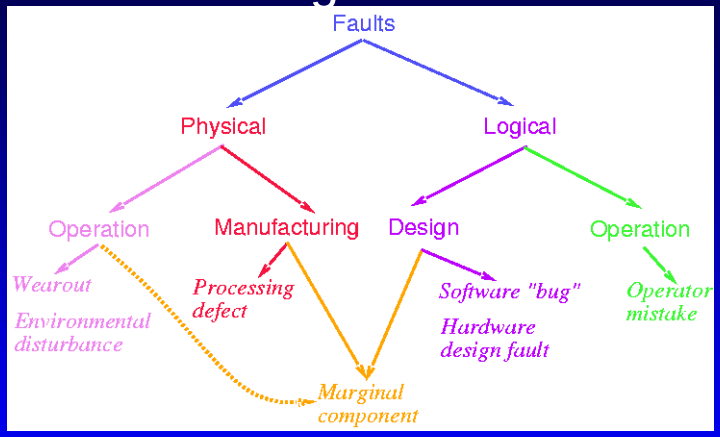
- **Manufacturing test**
  - **Dealing with faults in sub 100 nm circuits**
  - **System-on-a-chip test issues**
- Design verification
  - Formal approaches for property checking
  - Abstractions to deal with complexity
- Fault tolerance
  - Dealing with soft errors
  - Application-level techniques

Outline

- Manufacturing Test
- Trends in Silicon defects
- Hierarchy and abstractions to deal with test generation complexity
- Native-Mode Built-In Self-Test
  - Application to testing processors
  - Test of A/D and D/A converters
- Testing of mixed-signal circuits
- Future Directions

July 19, 2005 J. A. Abraham 3

Dealing with Faults



- Manufacturing faults, field failures: testing, design for testability
- Design faults: simulation and emulation, formal techniques
- Operational faults: concurrent error detection and fault tolerance

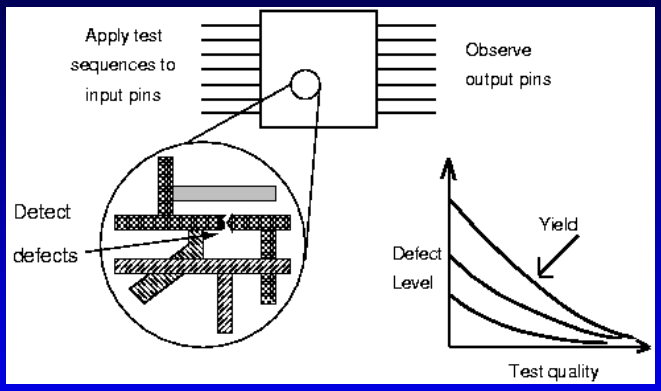
July 19, 2005 J. A. Abraham 4

Manufacturing Test

- A speck of dust on a wafer is sufficient to kill chip
- *Yield* of any chip is < 100%
  - Must test chips after manufacturing before delivery to customers to only ship good parts
- Manufacturing testers are very expensive
  - Minimize time on tester
  - Careful selection of *test vectors*

July 19, 2005 J. A. Abraham 5

Test Generation



A test for a defect will produce an output response which is different from the output when there is no defect

Test quality is high if the set of tests will detect a very high fraction of defects

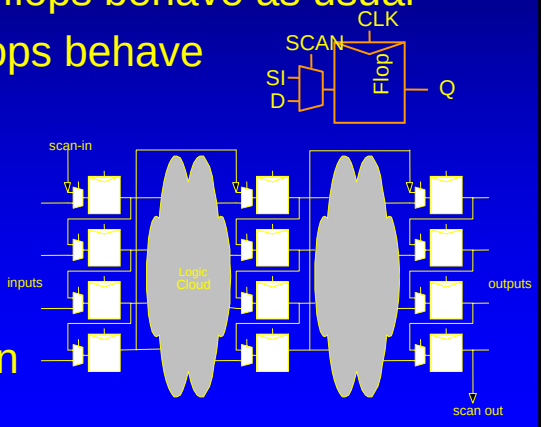
Defect level is the percentage of bad parts shipped to customers

Yield is the percentage of defect-free chips manufactured

July 19, 2005 J. A. Abraham 6

Scan – Deal with Sequential Circuits

- Convert each flip-flop to a scan register
  - Only costs one extra multiplexer
- Normal mode: flip-flops behave as usual
- Scan mode: flip-flops behave as shift register
- Contents of flops can be scanned out and new values scanned in



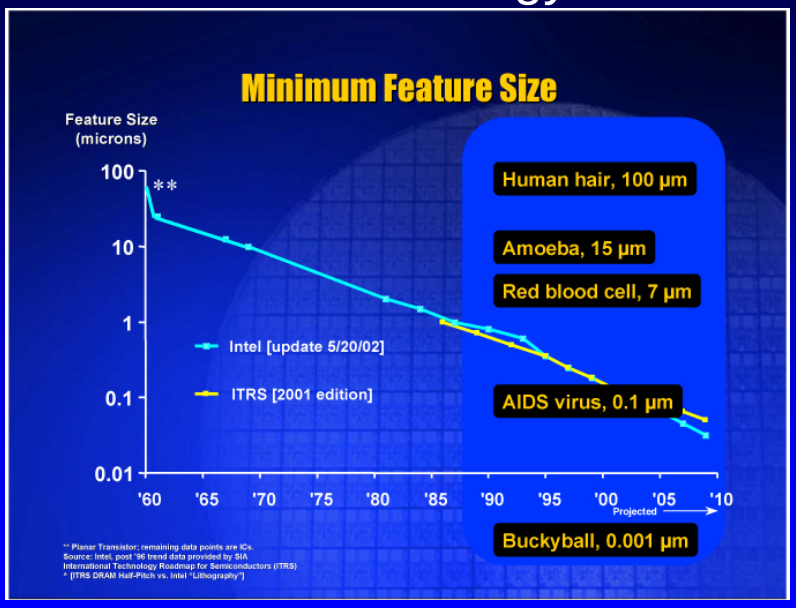
July 19, 2005 J. A. Abraham 7

What About the Future

- Are existing test solutions sufficient to deal with the defects which may exist in emerging technologies?
- Look at some of the technology trends
- Examine industry experience in new types of defects and test problems

July 19, 2005 J. A. Abraham 8

IC Technology

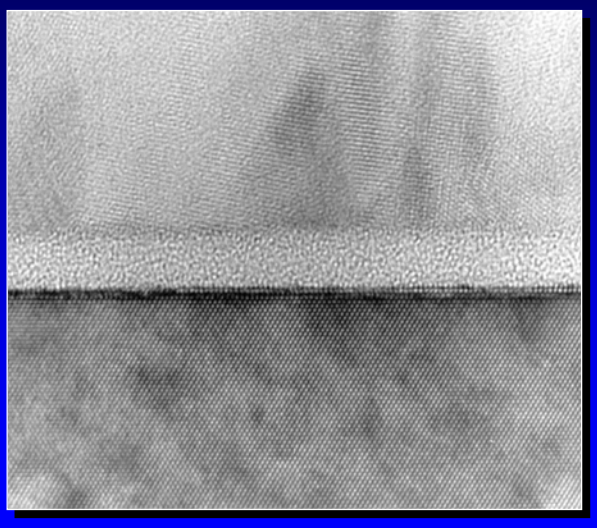


How Small Can It Get?

Polysilicon Gate

Gate Oxide

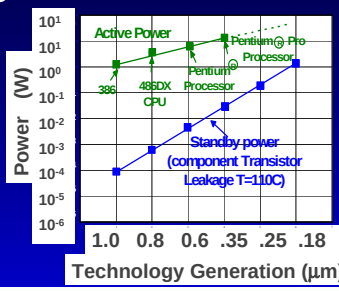
Silicon Channel

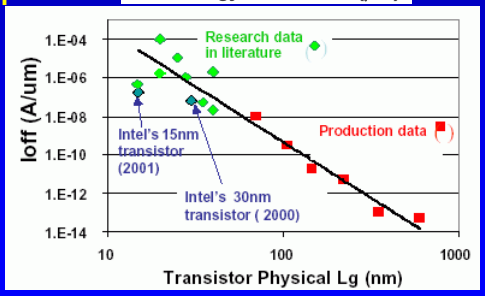


Technology Trends

- Faster circuits
- Decreasing V_t
- Increasing leakage
- Increasing parameter variations

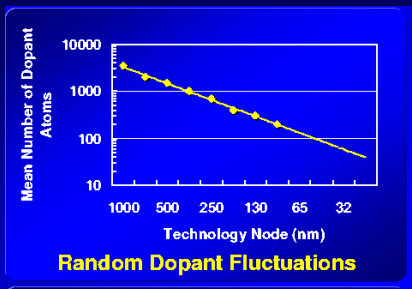
Effects exacerbated by radiation

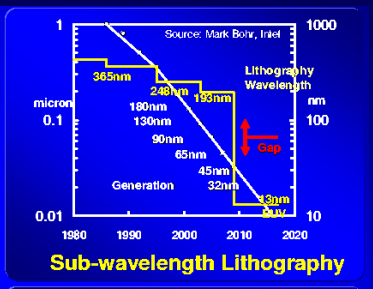


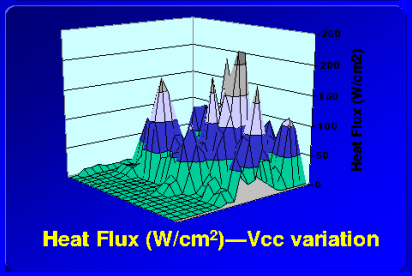


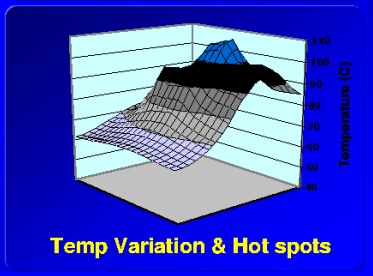
July 19, 2005 J. A. Abraham **Source: Intel** 11

Process Variations





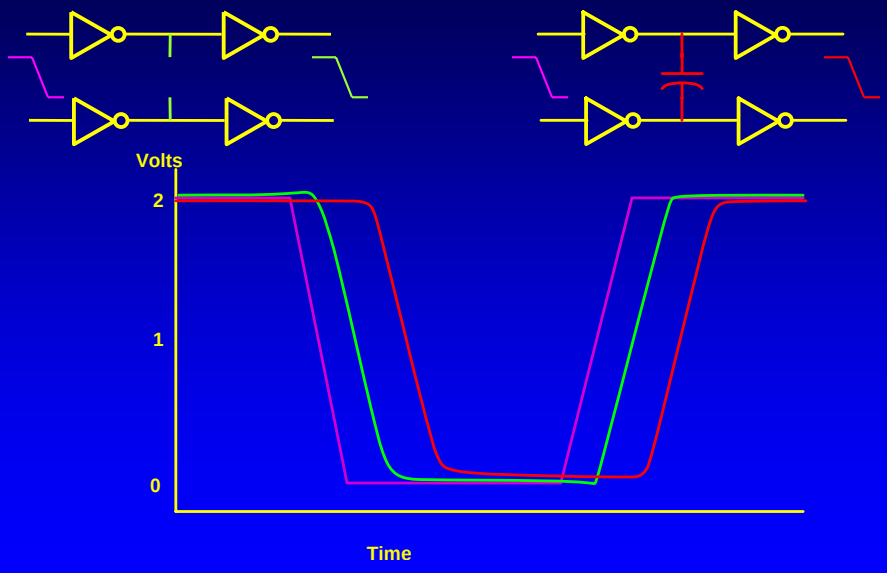




Source: Intel

July 19, 2005 J. A. Abraham 12

Delay Effects of Crosstalk



Defects in Emerging Technologies

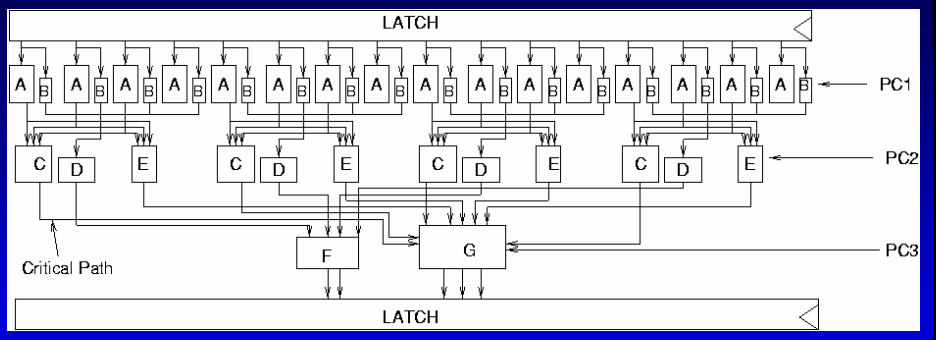
- Resistive opens comprise the bulk of test escapes in one production line
  - Likely in copper interconnect – cause delay faults
- Delay faults identified as the cause of most test escapes on another line
  - Speed differences of up to a factor of 1.5 can exist between fast and slow devices - problems with "speed binning"
- Increasing possibility of shorts and crosstalk
- Defects appearing as **delays or soft failures**

Impact on Test

- Existing Design for Test (DFT) structures are geared to “stuck” faults
  - Not easy to apply two-pattern tests necessary to check for path delays
- Performance optimizations result in a large number of paths close to the critical delay values
- Designs using statistical timing models would require the circuits to be also tested for paths close to the critical delay

July 19, 2005 J. A. Abraham 15

Static Timing Verification – Speed Paths



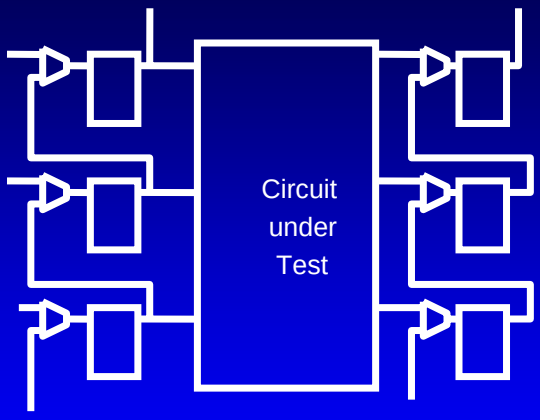
Use structure of circuit and delay to determine the **longest structural path** in block (could be a **false path**)

Only true paths on chip can be tested

Need access mechanisms to support at-speed test

July 19, 2005 J. A. Abraham 16

Scan Design and Delay Test



Need two patterns for delay test

Shifting in second pattern changes state of the nodes

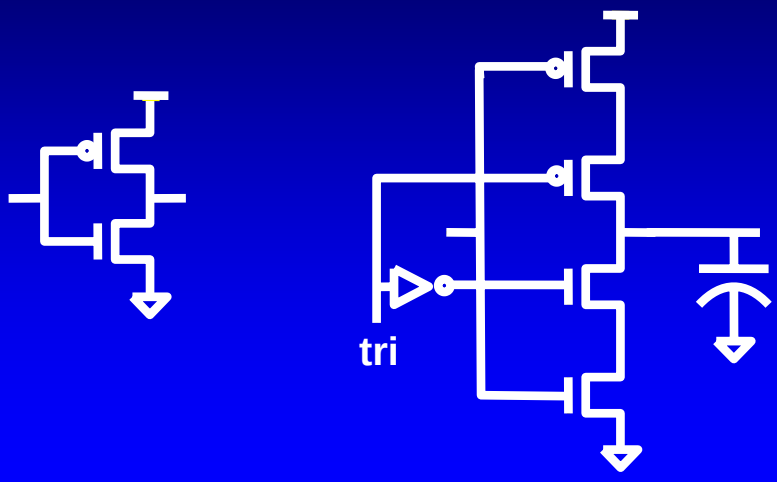
Solutions: Scan Shifting or Last Shift Launch

Functional Justification or Broadside Test

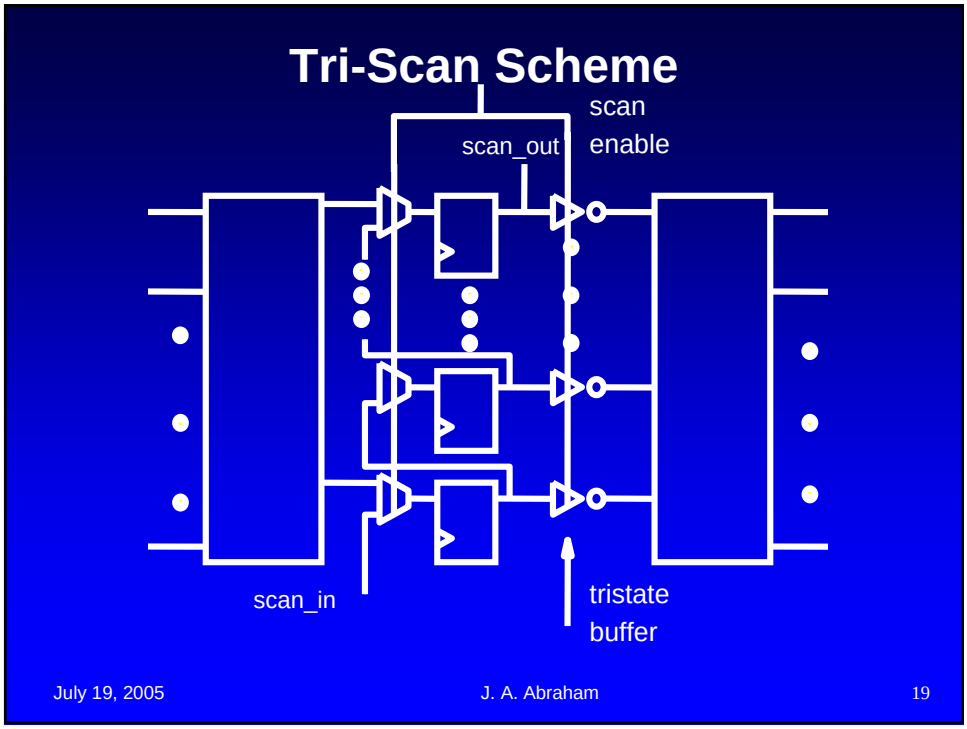
July 19, 2005 J. A. Abraham 17

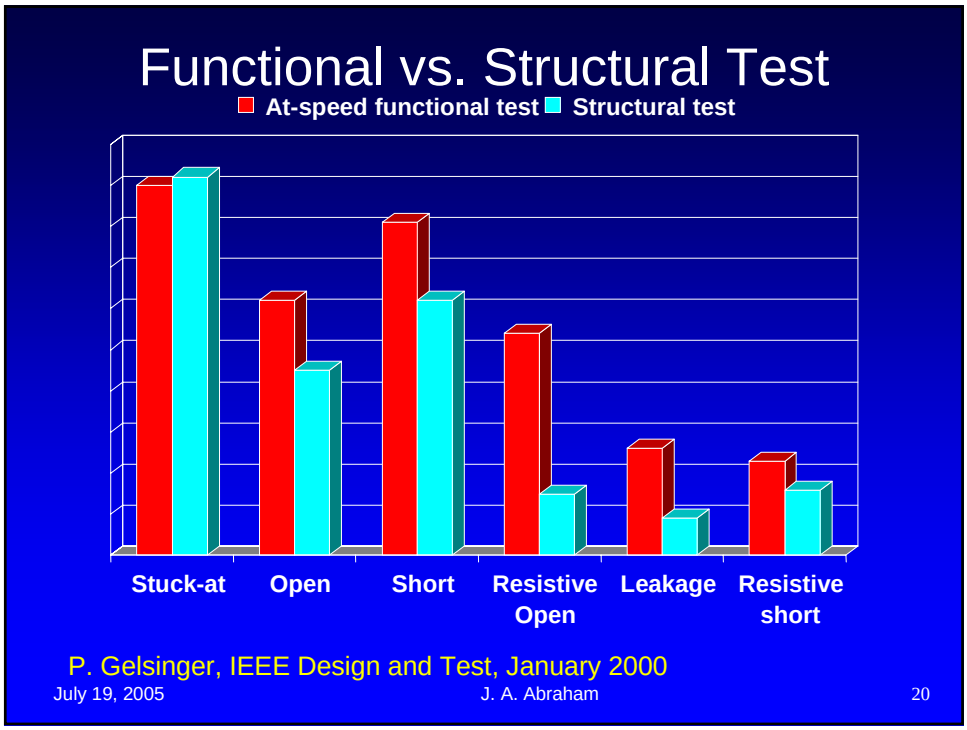
Tri-Scan Scheme

Based on state holding property of CMOS



July 19, 2005 J. A. Abraham 18


Tri-Scan Scheme
scan
enable
scan_out
scan_in
tristate
buffer
July 19, 2005
J. A. Abraham
19


Functional vs. Structural Test
At-speed functional test
Structural test
Stuck-at
Open
Short
Resistive Open
Leakage
Resistive short
P. Gelsinger, IEEE Design and Test, January 2000
July 19, 2005
J. A. Abraham
20

Dealing With the Entire Design

- Interest in chip-level (functional) tests
- Delay tests generated at the block level may be too conservative
  - "Long" paths at the block level may be false at the next level
- Problem of dealing with the entire design:
  - **Design with around 300 memory elements has more states than the number of protons in the universe!**

Dealing With the Entire Design

- Experiments on benchmark processors (target embedded module in processor)
  1. Tests for faults in module under test (MUT) by itself
  2. Tests for MUT faults when embedded in processor

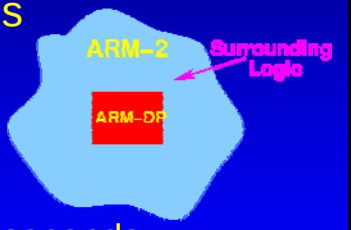
	Comb. Gates	Seq. Elems.	PIs	POs	Faults
ARM-2	16029	1270	63	67	99198
ARM-DP	8893	295	199	161	51824

Test Generation on Module

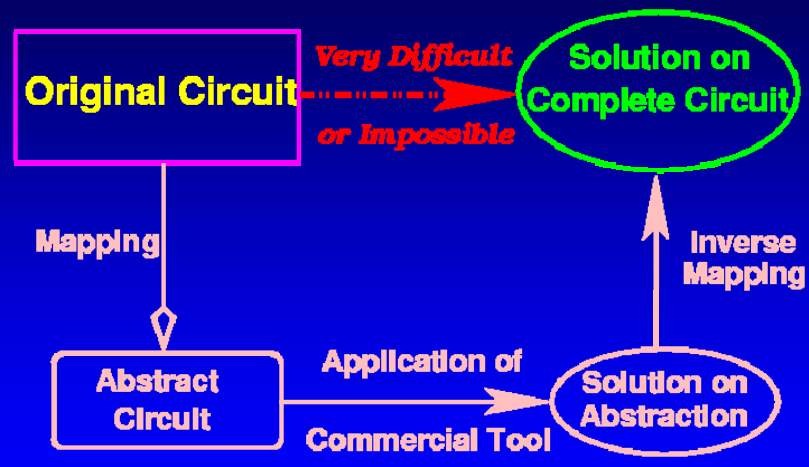
- Sequential ATPG can easily deal with a module
- Example: ARM-DP by itself
- Results using commercial ATPG tool (on HPUX-715, 125 MHz processor)
  - Fault Coverage: 99.70%
  - ATPG Efficiency: 99.93%
  - Test generation time: 33.1 seconds
  - Test length: 822 cycles

Test Generation on Embedded Module

- Sequential ATPG cannot deal with a module **when it is embedded in even a moderately complex design**
- Results on ARM-DP when it is embedded in ARM-2
  - Fault Coverage: 17.66%
  - ATPG Efficiency: 17.66%
  - Test generation time: 316,199 seconds

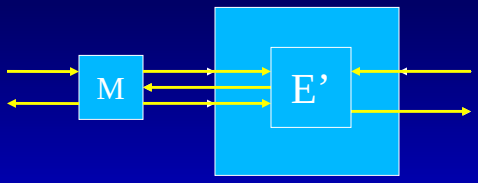


Abstraction to Deal With Complexity



July 19, 2005 J. A. Abraham 25

Test Generation Methodology



- Abstract HDL description (currently synthesizable Verilog) of environment, **synthesize to gate level**
- Commercial ATPG targeting faults in Module Under Test (MUT)
- ***Program slicing* for sound and complete abstraction**

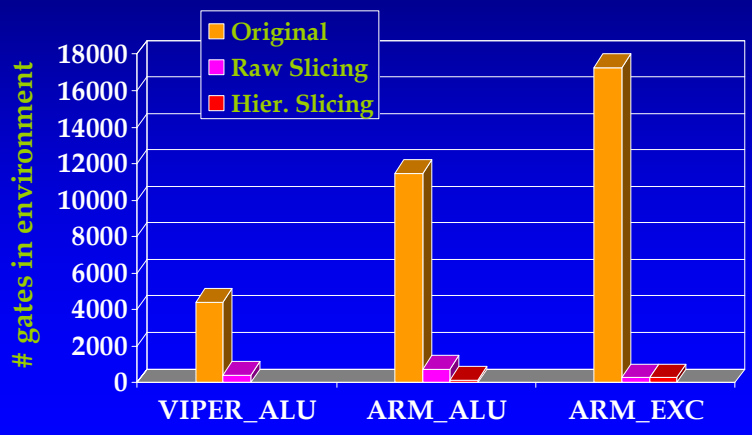
July 19, 2005 J. A. Abraham 26

Key Theory

- *Lemma* 1: Union of constraint slices provides abstracted environment
- *Lemma* 2: Process trace for signals defined in a MUT preserved
- *Theorem* 1: Valid set of patterns for the MUT using constraint slices
- *Theorem* 2: Hierarchical composition yields the desired constraint slice

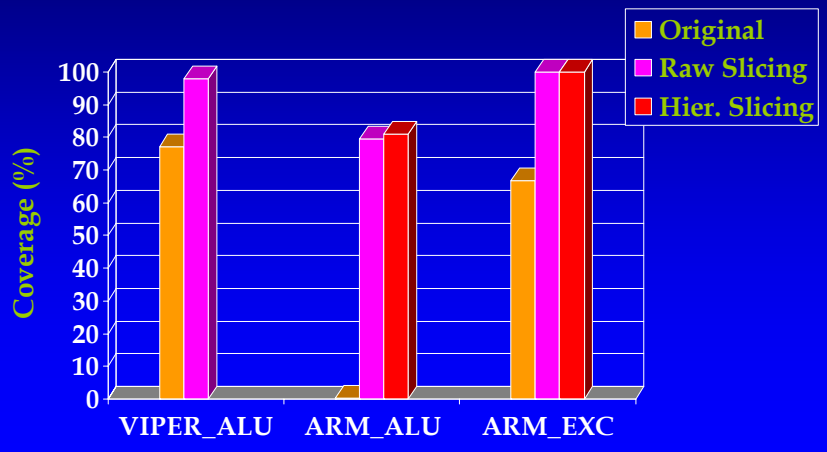
Slicing Experiments

- Automated into FACTOR (DATE 2002)
- Verilog models of Viper and ARM-2



Test Coverage Improvement

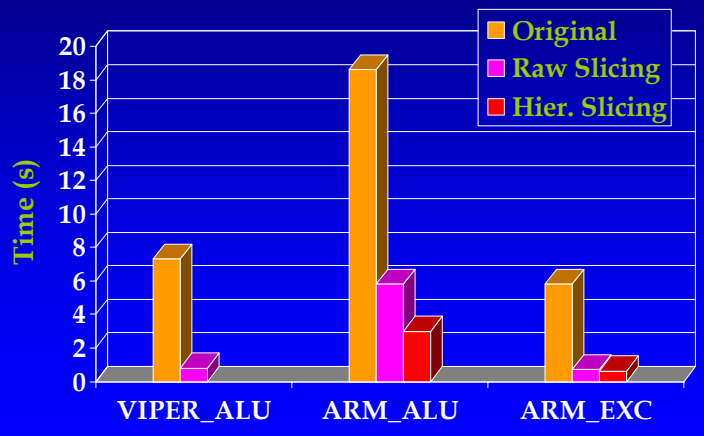
- 450 MHz dual UltraSPARC-II with 1 GB RAM
- Commercial sequential ATPG tools used



July 19, 2005 J. A. Abraham 29

Reduction in ATPG Time

- Measured in system CPU seconds

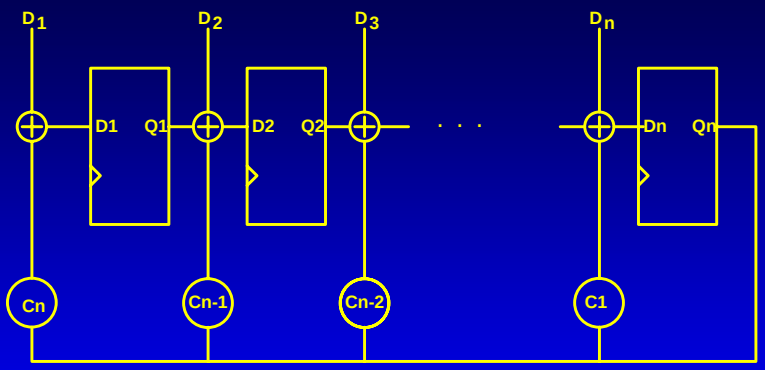


July 19, 2005 J. A. Abraham 30

Native-Mode Built-In Self Test

- Functional capabilities of processors can be used to replace BIST hardware – **(Shen and Abraham, ITC 1998)**
- Application to self-test of processors at Intel – **FRITS method applied to Pentium 4, Itanium (ITC 2002)**
- Embedded processor can be used to test other modules in a System-on-a-Chip (research at Texas, Santa Barbara, San Diego)
- Extend approach to test embedded analog and mixed-signal modules

Native-Mode Signature Compression




```
// S: general register or memory, holds signature
 for each register Ri to be compressed {
      Shift_Right_Through_Carry(Ri);
      if (Carry) { S = XOR(S, feedback_polynomial) }
      S = XOR(S, Ri);
   }
```

Intel Functional BIST

- Functional tests have good "collateral coverage" for detecting unmodeled defects
- Technique of Functional Random Instruction Testing at Speed (FRITS) applied to Itanium processor family and Pentium 4 line
  - Parvathala et al., Int'l Test Conference, 2002
- Tests (kernels) are instruction sequences
  - **Kernels loaded into cache** and executed in real time during test application
  - They **generate and execute** pseudo-random or directed sequences of machine code

July 19, 2005 J. A. Abraham 33

FRITS Results

- On Pentium 4, full chip-level coverage of single-stuck faults was ≈ 70% with FRITS
  - Added 5% unique coverage to manually generated tests
- Helped reduce test "holes"
  - Screen 10%–15% of chips which passed wafer sort/package tests by failed system tests
- Enables low cost testers
  - 40% increase in defect screening on structural tester
- Kernels execute 20 loops in ≈ 8 mSecs

July 19, 2005 J. A. Abraham 34

Automatic Mapping of Module Tests to Instruction Sequences

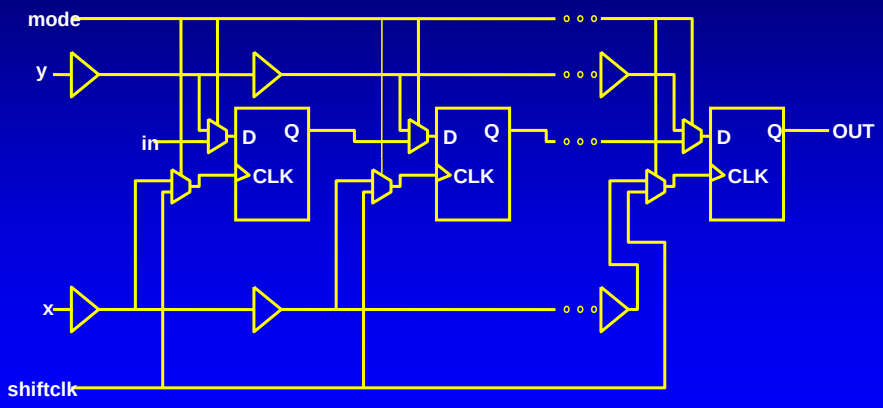
- Functional and directed random tests provide good fault coverage (95% – 96%) in high-performance microprocessors
- Need to develop tests for the remaining, hard-to-detect faults
- Able to generate tests for these faults at the **module level**
- **Developing new techniques for mapping module-level tests to instruction sequences**

Change in Test Requirements

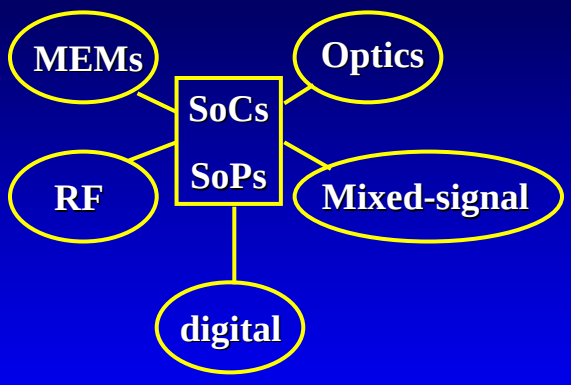
- Analog and mixed-signal circuits need to be tested for conformance with specifications
- Delay tests (speed binning) for digital circuits are also specification-based tests
- Designing for process variations will require sophisticated tests to screen for parts falling outside the acceptance limits

On-Chip Delay Measurement

Modified Vernier Delay Line



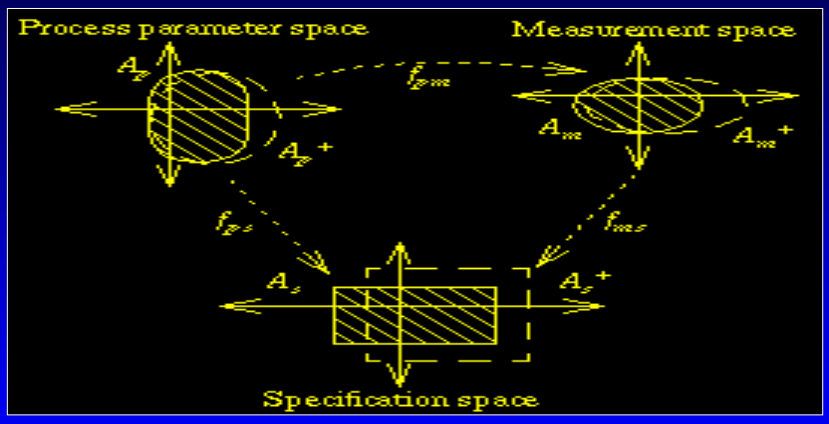
Testing Mixed-Signal Circuits



Have to deal with continuous signals

Need to test for specifications

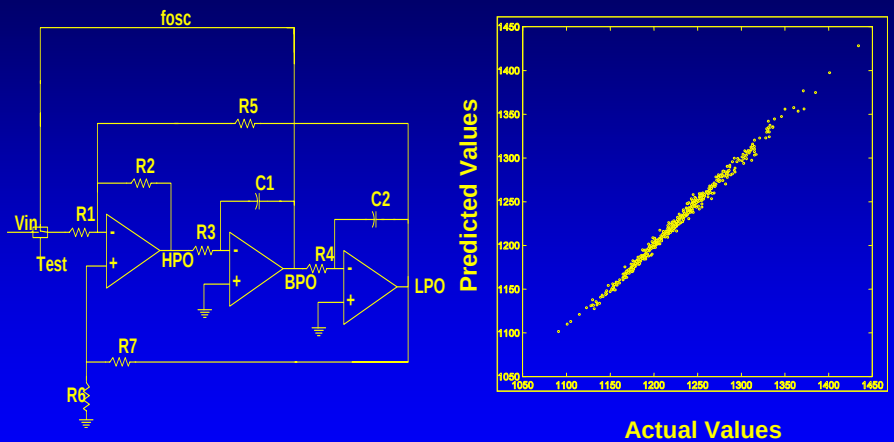
Alternate Tests: Key Idea



- The mapping f_{ms} is derived using regression (MARS)

Source: Chatterjee

Oscillation-Based Test

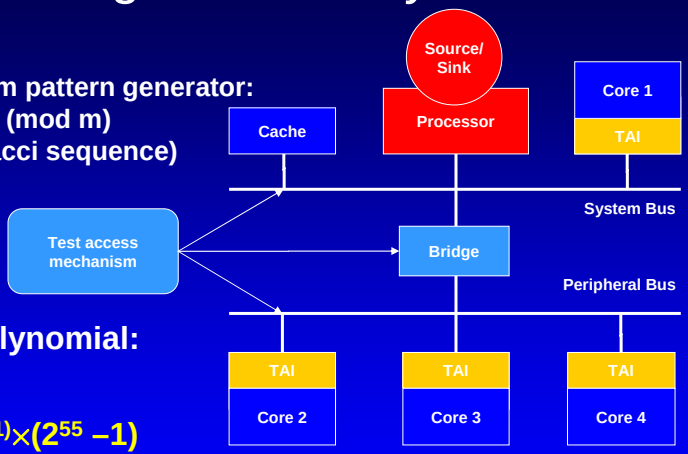


3dB Frequency

July 19, 2005 J. A. Abraham

Testing Cores in System

Example random pattern generator:
$X_n = X_{n-24} + X_{n-55}$ (mod m)
(Lagged Fibonacci sequence)



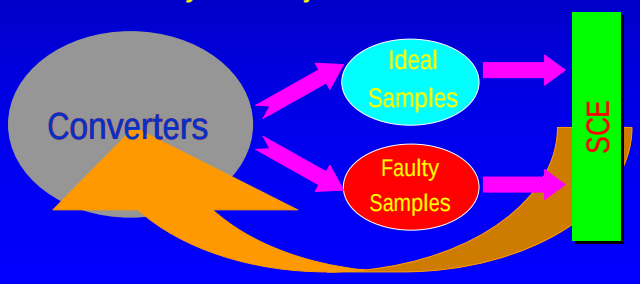
Primitive polynomial:
$X^{55} + X^{24} + 1$

Period: $2^{(32-1)} \times (2^{55} - 1)$

Source/Sink: Embedded Processor
Test access mechanism: System/Peripheral Bus
Test Access Interface (TAI): Core test wrapper

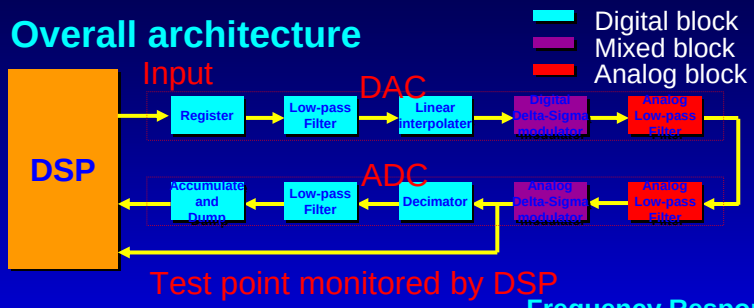
DSP-Based BIST

- Use **Statistical Characteristic Equations** (SCEs)
- Statistical Parameters
  - Good correlation between the inputs and the outputs
  - Physical or mathematical parameter
- Real faulty samples and ideal samples
  - Generated by fault injection on simulation models

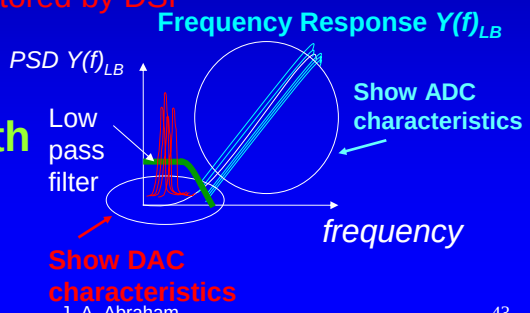


BIST for Σ-Δ D/A – A/D Converters

Overall architecture



DSP only deals with digital signals



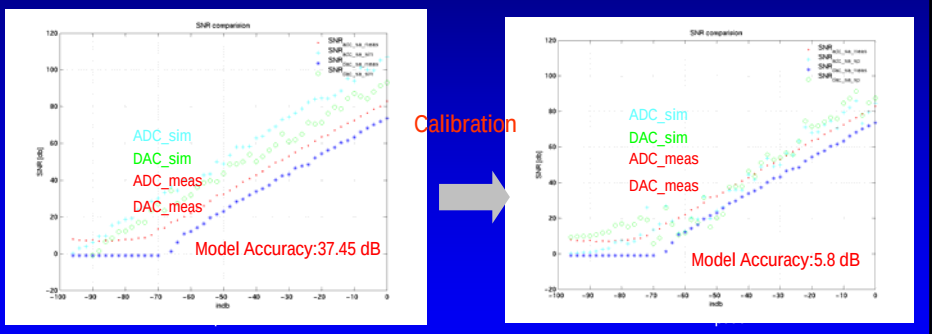
Implementation

- Applied BIST technique to industry chip for wireless applications
  - Oversampling Δ-Σ converters
  - Analog loop-back
  - SNR measurement

Results

Application: industry chip for wireless applications

Oversampled Σ-Δ converters, SNR measurement



The setup for SNR measurement was a little noisy, but was used to get these preliminary results

July 19, 2005 J. A. Abraham 45

Ongoing Work in Test

- New approaches for dealing with process variations and sub-100 nm defects
- Testability features for improved DSP-based BIST
- Applications to high-frequency designs (PLLs, RF, etc)
- Algorithms for mapping module-level tests to instructions
- Improved abstraction techniques to speed up test for complex designs

July 19, 2005 J. A. Abraham 46