Temperature-Aware Test Scheduling for Multiprocessor Systems-On-Chip

David R. Bild, Sanchit Misra, Thidapat Chantem, Prabhat Kumar,
Robert P. Dick, X. Sharon Hu, Li Shang, and Alok Choudhary

Department of EECS
Northwestern University
Evanston, IL 60208, USA

Department of CSE
University of Notre Dame
Notre Dame, IN 46556, USA

Department of ECE
University of Colorado at Boulder
Boulder, CO 80305, USA

November 10, 2008

Outline

Outline

1. Introduction
2. Power Analysis
3. Test Scheduling

Power Density and Temperature

High Temperature

- Process scaling leads to increasing power densities
- Temperature is strongly dependent on power density

Reduced Yield

- Causes permanent faults
- Accelerates failure processes
- Leads to timing errors

High Temperatures During Test

Power Density During Test

- Higher switching activity (e.g., scan-chain, BIST)
- Lower clock frequency

Thermal Environment

- Liquid-cooled chuck
- Thermal compounds generally not used between wafer and chuck

Temperature-Aware Test Scheduling

Goal

Minimize total test application time under a constraint on temperature

MPSoC Test Scheduling

- Cores can be tested concurrently leading to higher power consumption
- Concurrent testing of adjacent cores leads to thermal hotspots
- Resource conflicts limit the concurrency

Outline

1. Introduction
2. Power Analysis
3. Test Scheduling

Related Work

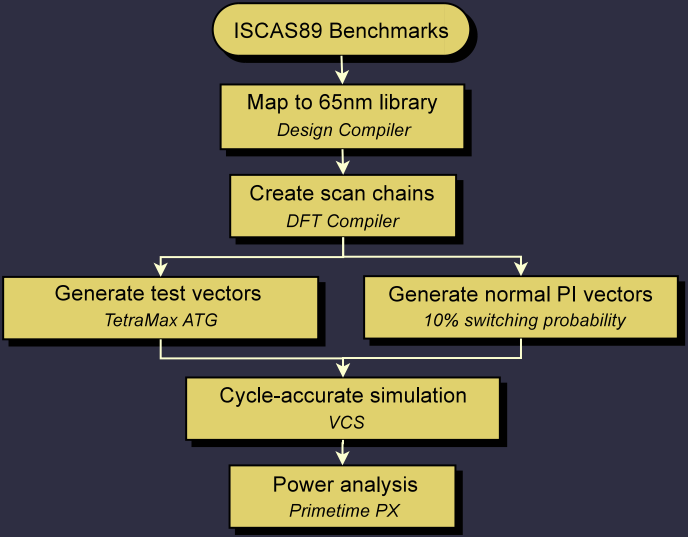
Little published data comparing power consumptions

- 2.5× increase for at-speed BIST - Y. Zorian (1993)
- 3× increase for ColdFire microprocessor core - B. Pouya and A.L. Crouch (2000)
- 3× increase historically for scan-chain - C. Shi and R. Kapur (2004)
- 30× for some designs

Main Contribution

- Comparison of normal operation and scan-chain test power consumptions for the ISCAS 89 benchmarks

Experimental Setup



ISCAS89 Benchmarks

Map to 65nm library
Design Compiler

Create scan chains
DFT Compiler

Generate test vectors
TetraMax ATG

Generate normal PI vectors
10% switching probability

Cycle-accurate simulation
VCS

Power analysis
Primetime PX

Analysis of Results

Summary

- Test: 26.9% average switching activity
- Normal: 8.5% average switching activity
- Switching Activity: 4.1× increase
- Power Consumption: 1.6× increase

Test clock frequency needs to be at least half of the normal operating frequency for the power consumption to be higher

4.1× & 1.6× Discrepancy

- Clock tree has high power consumption
- Independent of the switching activity
- Similar results seen by Pouya and Crouch (2000)

When Could High Temperatures Still Be a Problem?

- The test frequency is at least half the normal operating frequency
- The circuit has greater inter-register combinational logic depth than the ISCAS89 circuit
- The circuit is tested in an inferior thermal environment
- Other testing methodologies (e.g., BIST, sequential)

Outline

1. Introduction
2. Power Analysis
3. Test Scheduling

Section outline

3. Test Scheduling
    - Motivation
    - Optimal Formulation
    - Experimental Results
    - Heuristic

Ensuring Safe Temperatures

Minimize Power Consumption

- Test pattern sequence reorganization - Flores et al. (1999)
- Test vector reordering - Girad et al. (1997)
- MILP-based approached for power-time trade-off analysis - Nourani and Chin (2004)

Constrain Temperature, Not Power

- Power profiles have spatial variation
- Temperatures are spatially correlated
- Concurrent testing of adjacent cores may not be safe, but non-adjacent cores might be safe

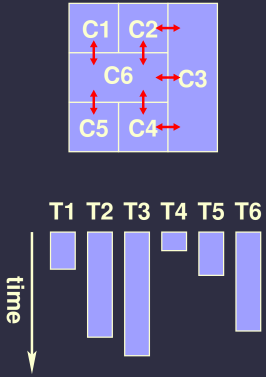
Existing Work

Clique-Set Technique

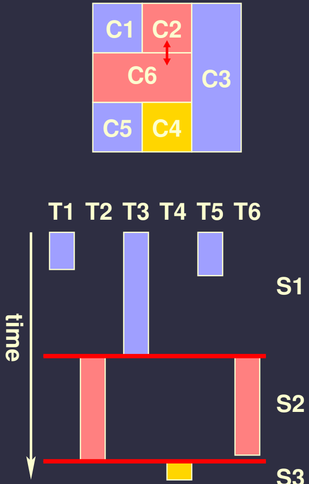
Rosinger, Al-Hashimi, Chakrabarty minimized SoC test time under temperature and resource constraints (2006)

- Identify clique sets of compatible tests
- Choose the covering set that minimizes the total time

Clique Set Weakness

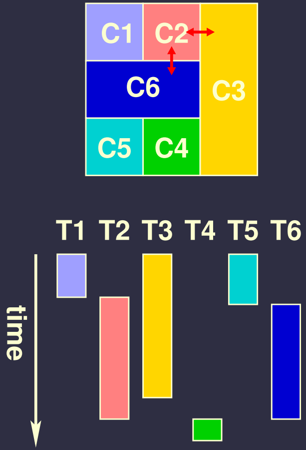


Overheated



Safe but slow

Arbitrary Start Times



Optimal: safe and fast

Section outline

3. Test Scheduling
    - Motivation
    - Optimal Formulation
    - Experimental Results
    - Heuristic

Problem Definition

Given:

- $\mathcal{C}$, the set of cores to test
- $E(c)$, the test execution time for each core
- $\Gamma(c_1, c_2)$, resource conflicts between cores
- T_{bound}, maximum allowable peak temperature

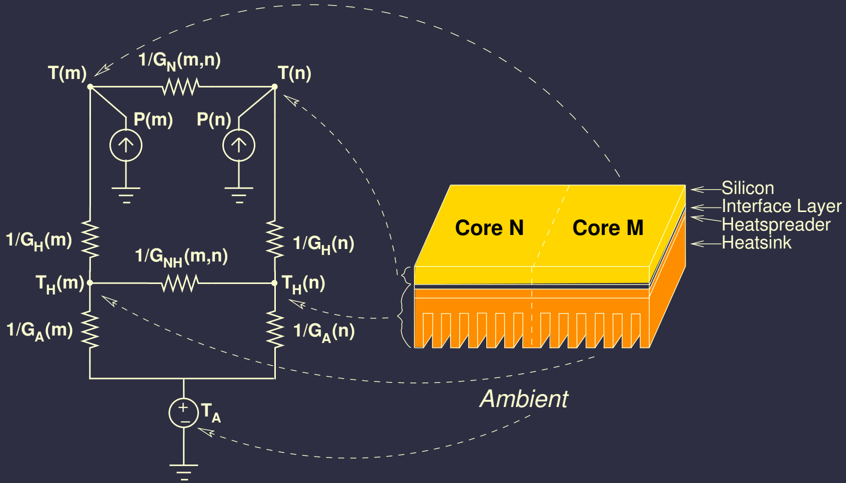
Find:

- $t_s(c)$, start time for each test

Such That:

- If $\Gamma(c_1, c_2) = 1$, c_1 and c_2 do not execute simultaneously
- T_{max}, die peak temperature, is less than T_{bound}
- The latest finish time, $max_{c \in \mathcal{C}} t_s(c) + E(c)$, is minimized

Thermal Model (1/2)



Thermal Model (2/2)

Dynamic Effects

- Heat capacity is not modeled
- Valid as long as test sequence times are relatively long (e.g., milliseconds)

Phased Steady-State

- Temperature can only increase when test sequences start
- Only evaluate the temperature profile at these times

Temperature Equations

- For this model, the system is linear
- Can be integrated into an MILP formulation

MILP Formulation

- Phased steady-state and linear temperature model allowed for integration in an MILP formulation
- Developed an MILP formulation in AMPL specification language
- Suitable for small problem instances

Section outline

3. Test Scheduling
    - Motivation
    - Optimal Formulation
    - Experimental Results
    - Heuristic

Experimental Setup

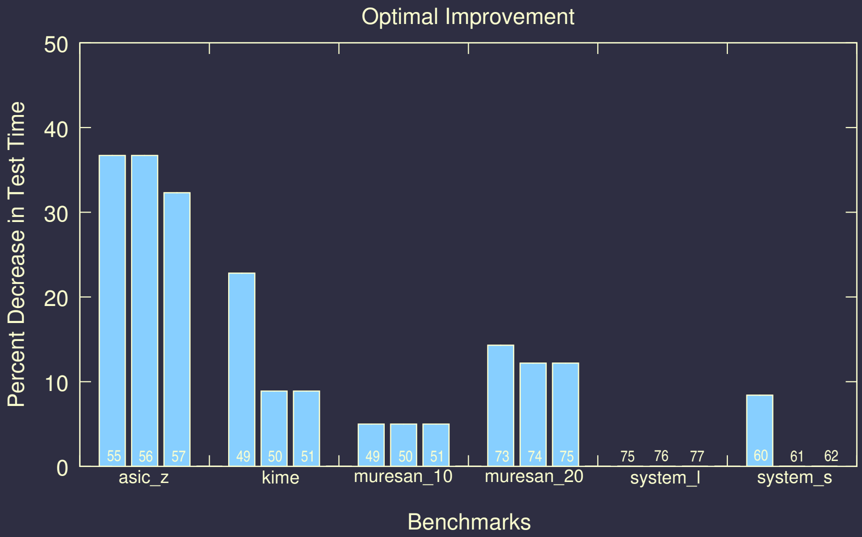
Optimal Solution

- Solved using AMPL and CPLEX
- Same set of benchmarks as Rosinger, Al-Hashimi, and Chakrabarty
- Five different temperature bounds to show the discontinuous relationship between temperature bound and optimal schedule length

Comparison with Clique-Set

- Implemented the clique-set technique in an MILP using our thermal model
- Solved for the same set of benchmarks and temperature bounds

Table of Results



Summary of Results

Summary

- 10.8% average improvement
- 36.7% maximum improvement

Analysis

- For most designs, the schedule is limited only by resource conflicts
- For system_l, no improvement for any temperature because it is severely resource constrained

Extensions to the Model

Test Sequence Granularity

- Real cores will have a finer test sequence granularity
- Our technique can handle this with no modification
  1. Split each test set into atomic subsequences
  2. Add resource conflicts between atomic subsequences
  3. Schedule as separate tests

Physical ATE Limitations

- Automated Test Equipment will have physical resource limitations
- For example, number of probe pins available
- Easily handled by adding constraints

Section outline

Need for Heuristic

- $\mathcal{NP}$-hard by reduction from task scheduling
- Can only optimally solve small problem instances
- Design muresan_20, with 20 cores, took 30 minutes

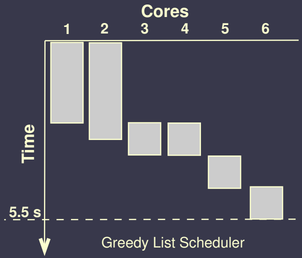
List Scheduler

List Scheduler Intuition

- Schedule high temperature impact tests first
- Maximize the chance that a temperature-compatible test will exist to schedule concurrently

Problem

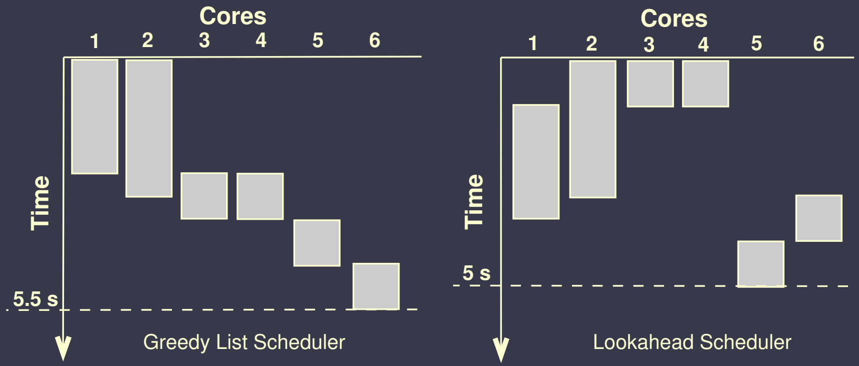
- Ignores future effects of resource conflicts



Look-Ahead Scheduler

Intuition

- Examine future impacts by growing groups of concurrent cores
- Schedule the seed from the best group



Heuristic

Algorithm

1. Ensure all cores can be scheduled alone
2. Select legal cores as seeds
3. Build groups using list scheduler (tests ordered by increasing temperature)
4. Schedule the seed with the largest group

Runtime

- Solve phased-steady state thermal model: $\mathbf{A} \times T + B = 0$ in $\mathcal{O}(|\mathcal{C}|^2)$
- Total runtime: $\mathcal{O}(|\mathcal{C}|^5)$

Experimental Results (1/2)

Design	Threshold Temp (°C)	Optimal Test Length (s)	Heuristic Test Length (s)	Increase Over Optimal (%)
asic_z	55	0.204	0.209	2.5
asic_z	56	0.204	0.204	0.0
asic_z	57	0.191	0.191	0.0
kime	49	3.180	3.180	0.0
kime	50	3.180	3.180	0.0
kime	51	3.180	3.180	0.0
muresan_10	49	1.900	1.900	0.0
muresan_10	50	1.900	1.900	0.0
muresan_10	51	1.900	1.900	0.0
muresan_20	73	3.600	4.000	11.1
muresan_20	74	3.600	3.600	0.0
muresan_20	75	3.600	3.600	0.0
system_l	75	2.880	2.880	0.0
system_l	76	2.880	2.880	0.0
system_l	77	2.880	2.880	0.0
system_s	60	8.448	8.448	0.0
system_s	61	8.448	8.448	0.0
system_s	62	8.448	8.448	0.0

Experimental Results (2/2)

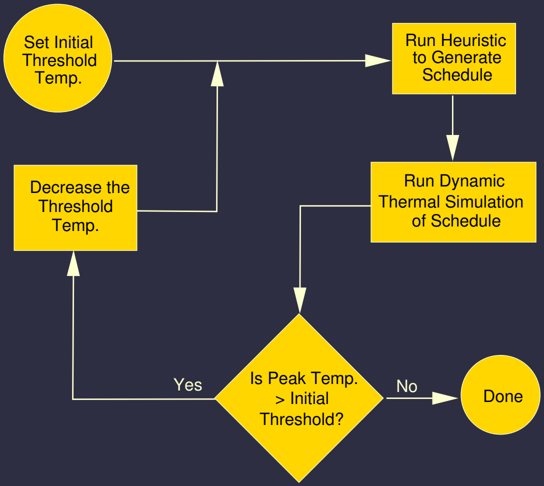
Summary

- Within 0.5% of optimal average
- Not worse than 11.1%
- Always as good as or better than existing prior work
- 10.5% better on average

Runtime

- 10.5 s for all 30 problem instances
- 0.3 s per instance on average

Dynamic Thermal Effects



Conclusions

Power Analysis

- Compared the normal mode and test mode power consumptions of the ISCAS89 benchmarks
- Found a 4.1× increase in switching activity lead to a 1.6× increase in power consumption

Test Scheduling

- Developed an optimal MILP formulation for MPSoC test scheduling
- Developed a heuristic which can quickly generate near optimal solutions
- Test schedule length improved by 10.8% over best existing approach

Acknowledgments

- We would like to thank Paul Rosinger and Bashir M. Al-Hashimi from the University of Southampton and Krishnendu Chakrabarty from Duke University for their advice and access to their benchmarks.
- We would also like to acknowledge Niraj K. Jha from Princeton University and Gokhan Memik from Northwestern University for their advice and support.
- This work is supported in part by the SRC under award 2007-TJ-1589 and in part by the NSF under awards CCF-0702761, CNS-0347941, CNS-0410771, CNS-0404341, IIS-0536994, and CCF-0444405.

Questions

Thank You!