DDRO: A Novel Performance Monitoring Methodology Based on Design-Dependent Ring Oscillators

Tuck-Boon Chan[†], Puneet Gupta[§], Andrew B. Kahng[†‡]
and *Liangzhen Lai*[§]

UC San Diego ECE[†] and CSE[‡] Departments, La Jolla, CA 92093

UC Los Angeles EE[§] Department, Los Angeles, CA 90095

Outline

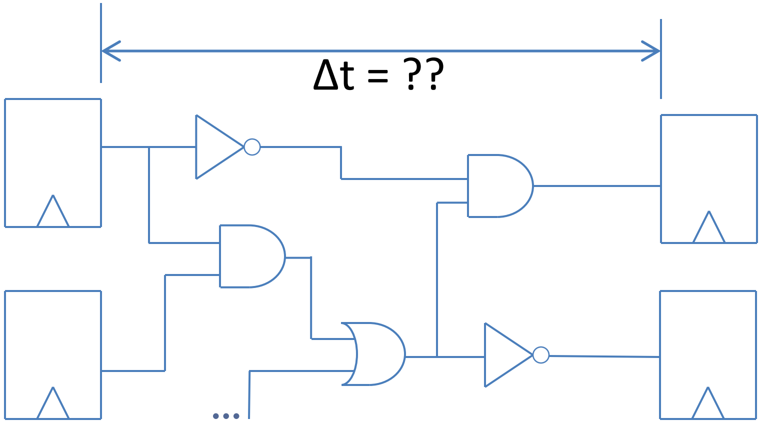
- Performance Monitoring: An Introduction
- DDRO Implementation
- Delay Estimation from Measured DDRO Delays
- Experiment Results
- Conclusions

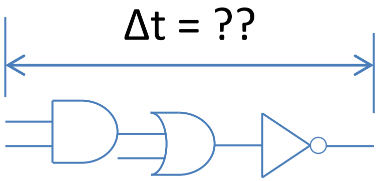
Performance Monitoring

- Process corner identification
  - Adaptive voltage scaling, adaptive body-bias
- Runtime adaptation
  - DVFS
- Manufacturing process tuning
  - Wafer and test pruning [Chan10]

Monitor Taxonomy

- In-situ monitors:
  - In-situ time-to-digital converter (TDC) [Fick10]
  - In-situ path RO [Ngo10, Wang08]
- Replica monitors:
  - One monitor: representative path [Liu10]
  - Many monitors: PSRO [Bhushan06]

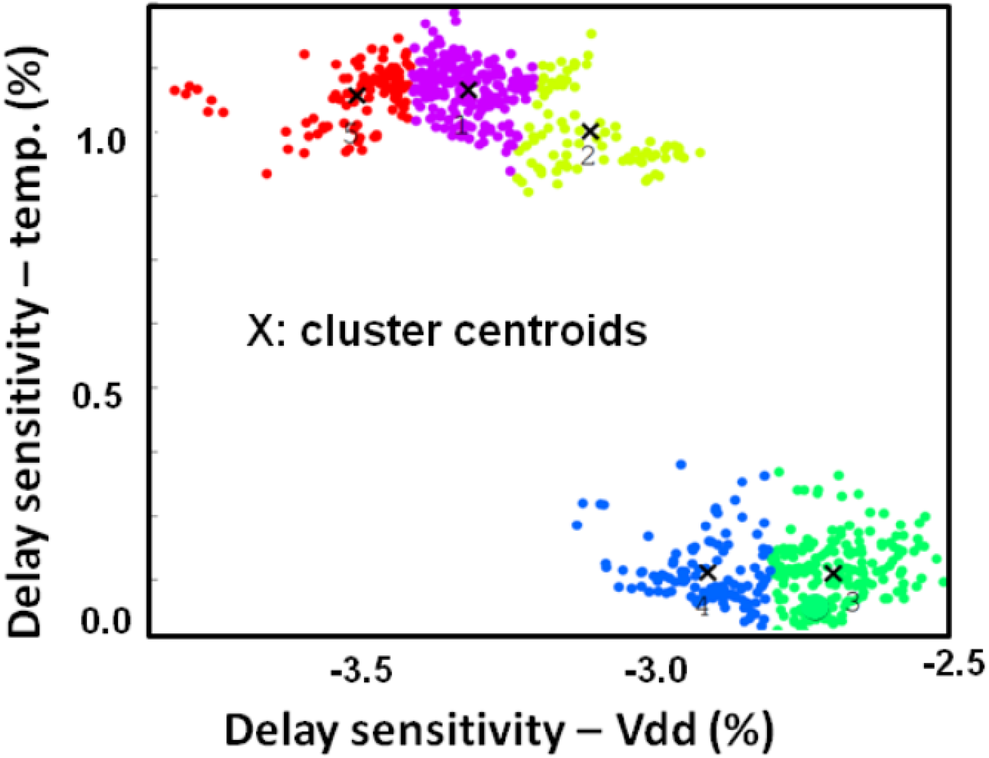




- How many monitors?
- How to design monitors?
- How to use monitors?

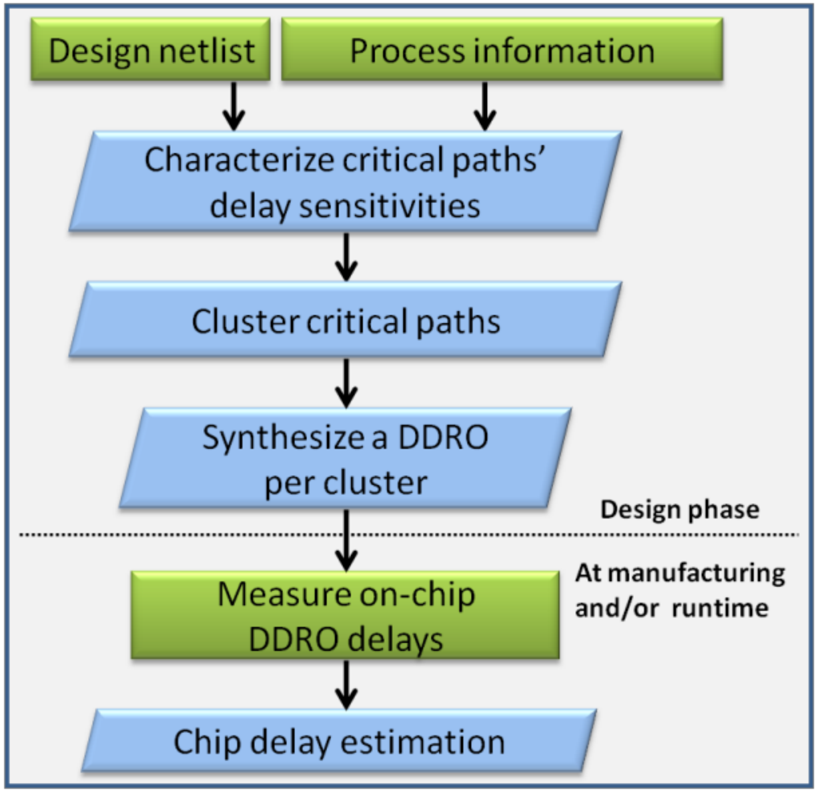
Key Observation: Sensitivities Cluster!

- Each dot represents Δdelay of a critical path under variations
- The sensitivities form natural clusters
  - – Design dependent
  - – Multiple monitors
    - One monitor per cluster



DDRO Contributions

- Systematic methodology to design ***multiple*** DDROs based on clustering
- Systematic methodology to leverage monitors to estimate chip delay


Design netlist
Process information
Characterize critical paths' delay sensitivities
Cluster critical paths
Synthesize a DDRO per cluster
Design phase
At manufacturing and/or runtime
Measure on-chip DDRO delays
Chip delay estimation

Outline

- Performance Monitoring: An Introduction
- DDRO Implementation
  - Delay model
  - Sensitivity Clustering
  - DDRO Synthesis
- Delay Estimation from Measured DDRO Delays
- Experiment Results
- Conclusions

Delay Model and Model Verification

- Assume a linear delay model for variations

Real delay

Sensitivities

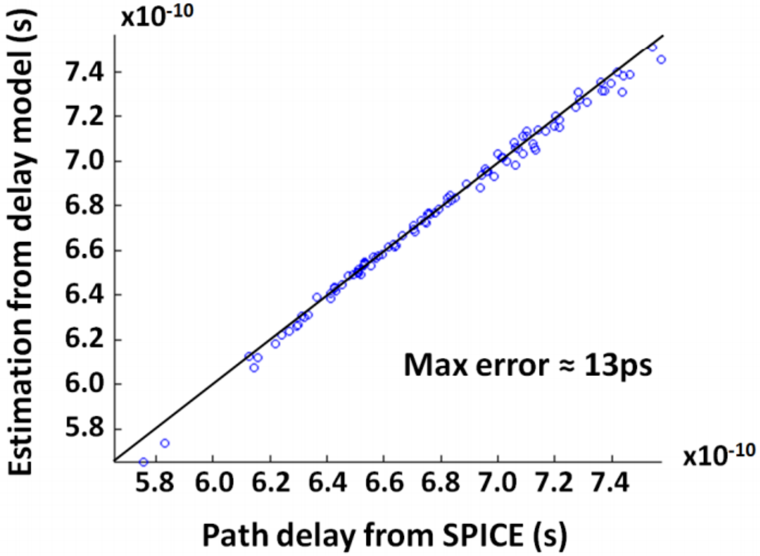
Variation magnitude

$$d = d_{nom}(1 + \sum V_j G_j)$$

Variation source index

Nominal delay

- Linear model correlates well with SPICE results



Sensitivities and Clustering

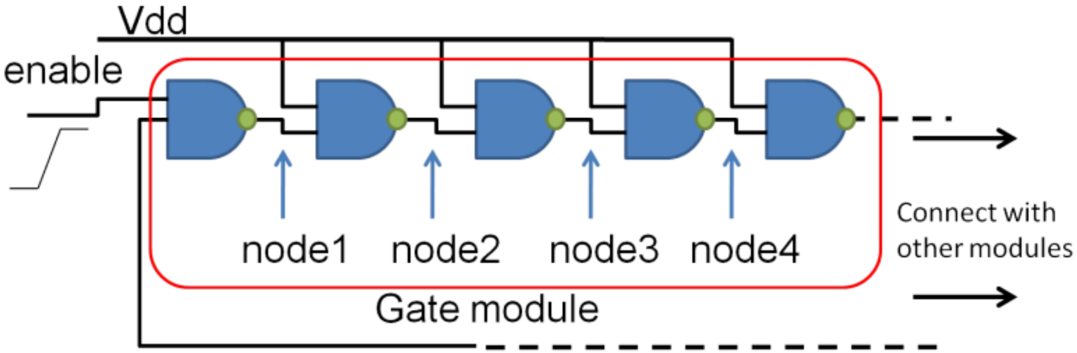
- Extract delay sensitivity based on finite difference method

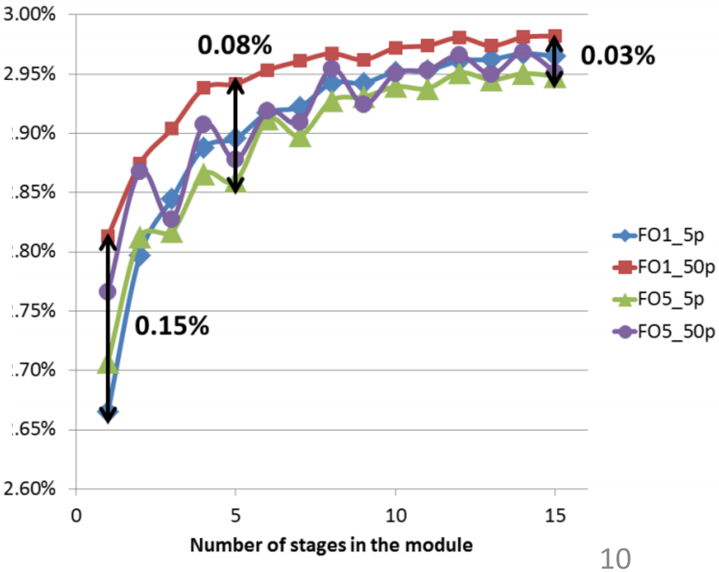
$$V_j = \frac{d_{G_j=1\sigma} - d_{nom}}{d_{nom}}$$

- Cluster the critical paths based on sensitivities
  - Use kmeans++ algorithm
  - Choose best k-way clustering solution in 100 random starts
  - Each cluster centroid = target sensitivity for a DDRO
- Synthesize DDROs to meet target sensitivities

DDRO Synthesis

- Gate module is the basic building block of DDRO
  - Consists of standard cells from qualified library
- Multiple cells are concatenated in a gate module
  - Inner cells are less sensitive to input slews and output load variation
  - Delay sensitivity is independent of other modules





ILP formulation

- Module sensitivity is independent of its location

$$\boxed{\text{RO sensitivity}} = \sum (s_h \times \boxed{\text{Module } h \text{ sensitivity}})$$

- Module number can only be integers
- Formulate the synthesis problem as integer linear programming (ILP) problem

Minimize: $\left| \boxed{\text{RO sensitivity}} - \boxed{\text{Target sensitivity}} \right|$

Subject to: $\text{Delay}_{\text{min}} < \sum (s_h \times \boxed{\text{Module 1 delay}}) < \text{Delay}_{\text{max}}$

$$\sum s_h < \text{Stage}_{\text{max}}$$

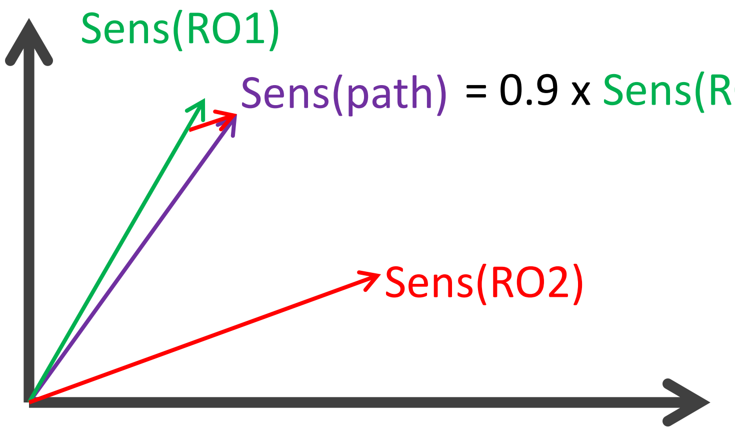
Outline

- Performance Monitoring: An Introduction
- DDRO Implementation
- Delay Estimation from Measured DDRO Delays
  - Sensitivity Decomposition
  - Path Delay Estimation
  - Cluster Delay Estimation
- Experiment Results
- Conclusions

Sensitivity Decomposition

- Based on the cluster representing RO
- User linear decomposition to fully utilize all ROs

Path sensitivity = $\sum(b_k$ x RO sensitivity)+ Sensitivity residue



Sens(path) = 0.9 x Sens(RO1) + 0.1 x Sens(RO2)

Path Delay Estimation

- Given DDRO delay, use the sensitivity decomposition
- Apply margin for estimation confidence

Predicted path delay Measured from RO Margin

$$d_i^{path} = d_{nom}^{path} \times (1 + \sum b_k \frac{d_k^{ro} - d_k^{nom}}{d_k^{nom}} + u_i)$$

Other variation components Sensitivity residue

$$where \;:\; u_i = l_i^{path} + V_{res} G$$

- *One estimation per path*

Cluster Delay Estimation

- For run-time delay estimation, may be impractical to make one prediction per path
- Reuse the clustering
  - Assume a pseudo-path for each cluster

$$d_X^{cluster} = \max\{d_i^{path}, path\ i \in cluster\ X\}$$

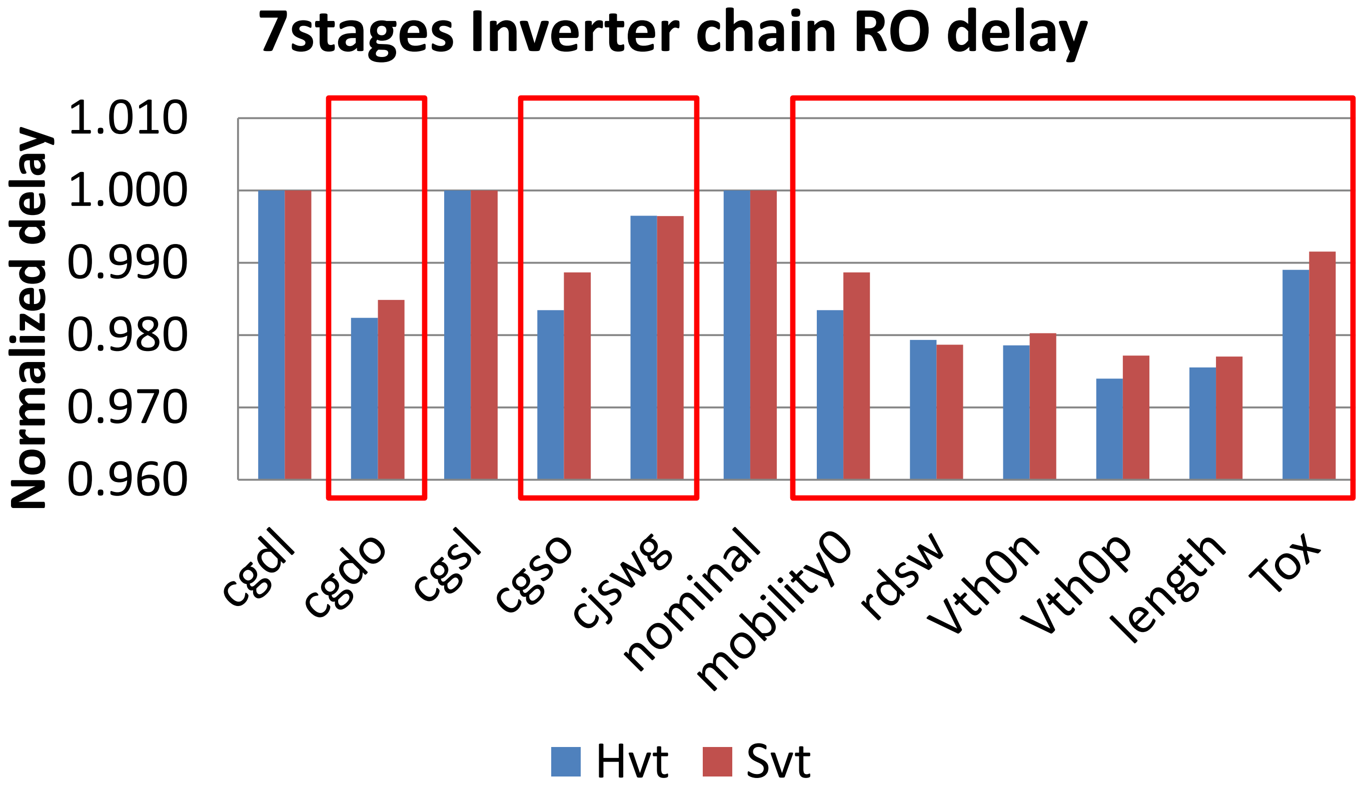
  - Use statistical method to compute the nominal delay and delay sensitivity of the pseudo-path
  - Estimate the pseudo-path delay
- *One estimation per cluster*

Outline

- Introduction
- Implementation
- Delay Estimation
- Experiment Results
- Conclusion

Sensitivity Extraction

- All variability data from a commercial 45nm statistical SPICE model

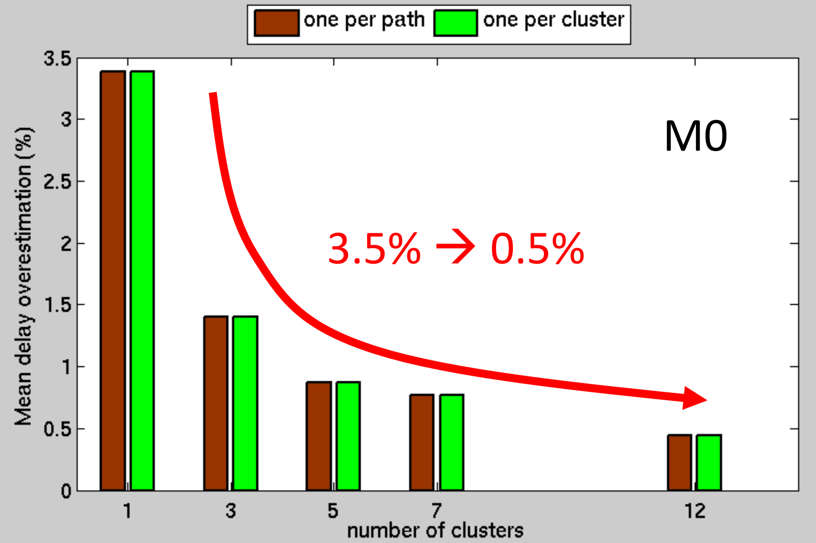


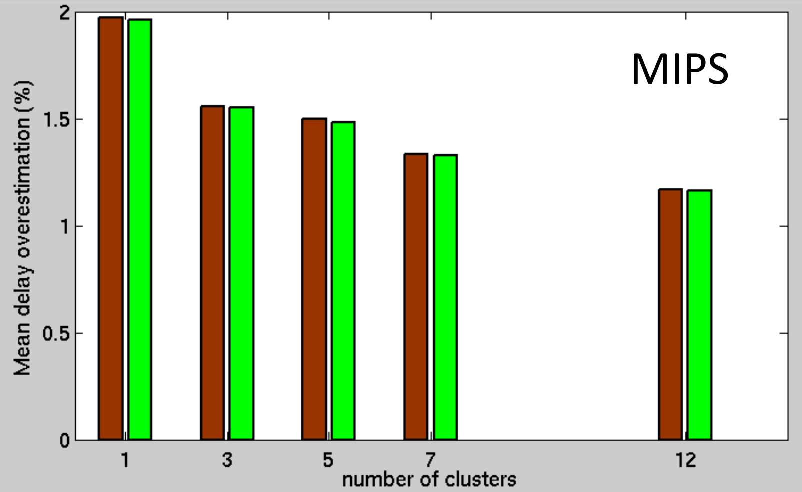
Experiment Setup

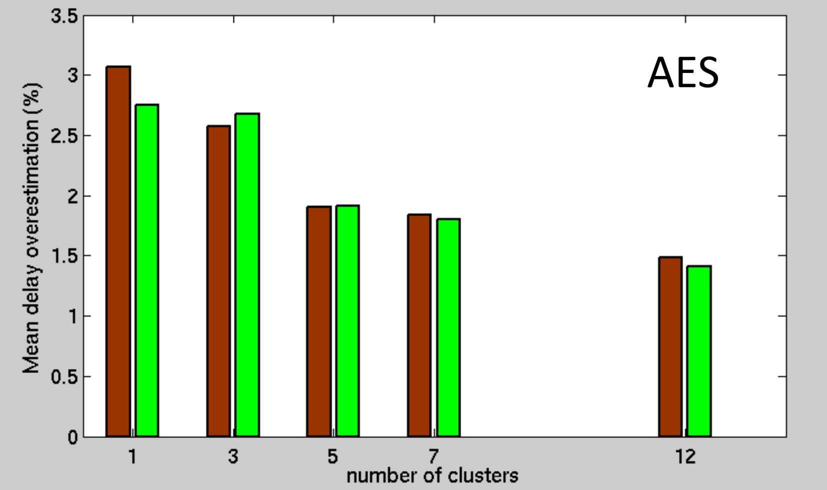
- Use Monte-Carlo method to simulate critical path delays and DDRO delays
- Apply delay estimation methods with certain estimation confidence
  - 99% in all experiments
- Compare the amount of delay over-prediction
  - Delay from DDRO estimation vs. Delay from critical paths

Linear Model Results

Global variation only



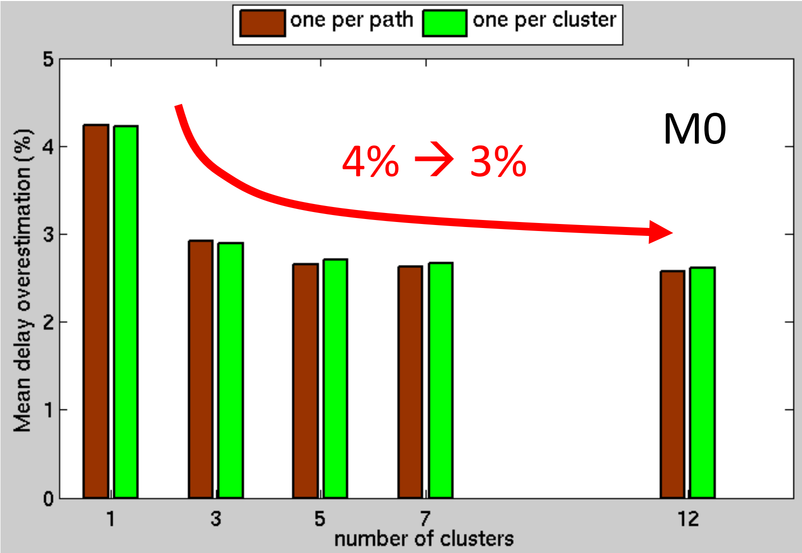


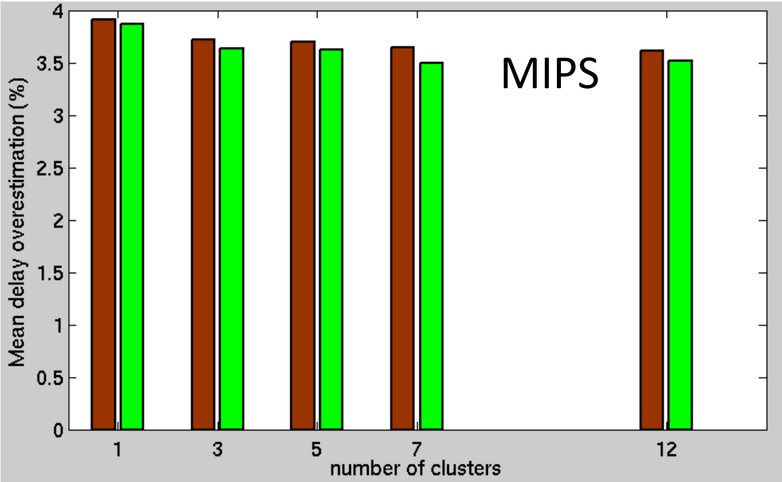


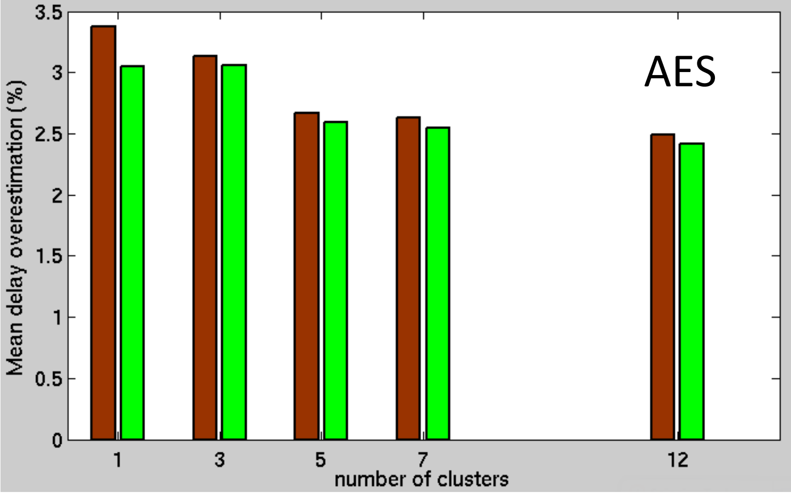
- Overestimation reduces as the number of cluster and RO increases
- The two estimation methods perform similarly

Linear Model Results

Global and local variations



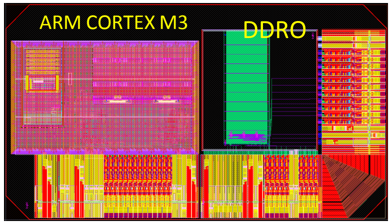




- With local variation, the benefit of having more ROs saturates
- Local variation can only be captured by in-situ monitors

Conclusion and Future Work

- A systematic method to design multiple DDROs based on clustering
- An efficient method to predict chip delay
- By using multiple DDROs, delay overestimation is reduced by up to 25% (from 4% to 3%)
  - Still limited by local variations
- Test chip tapeout using 45nm technology
  - With an ARM CORTEX M3 Processor



Acknowledgments

- Thanks to Professor Dennis Sylvester, Matt Fojtik, David Fick, and Daeyeon Kim from University of Michigan

SRC® Semiconductor Research Corporation

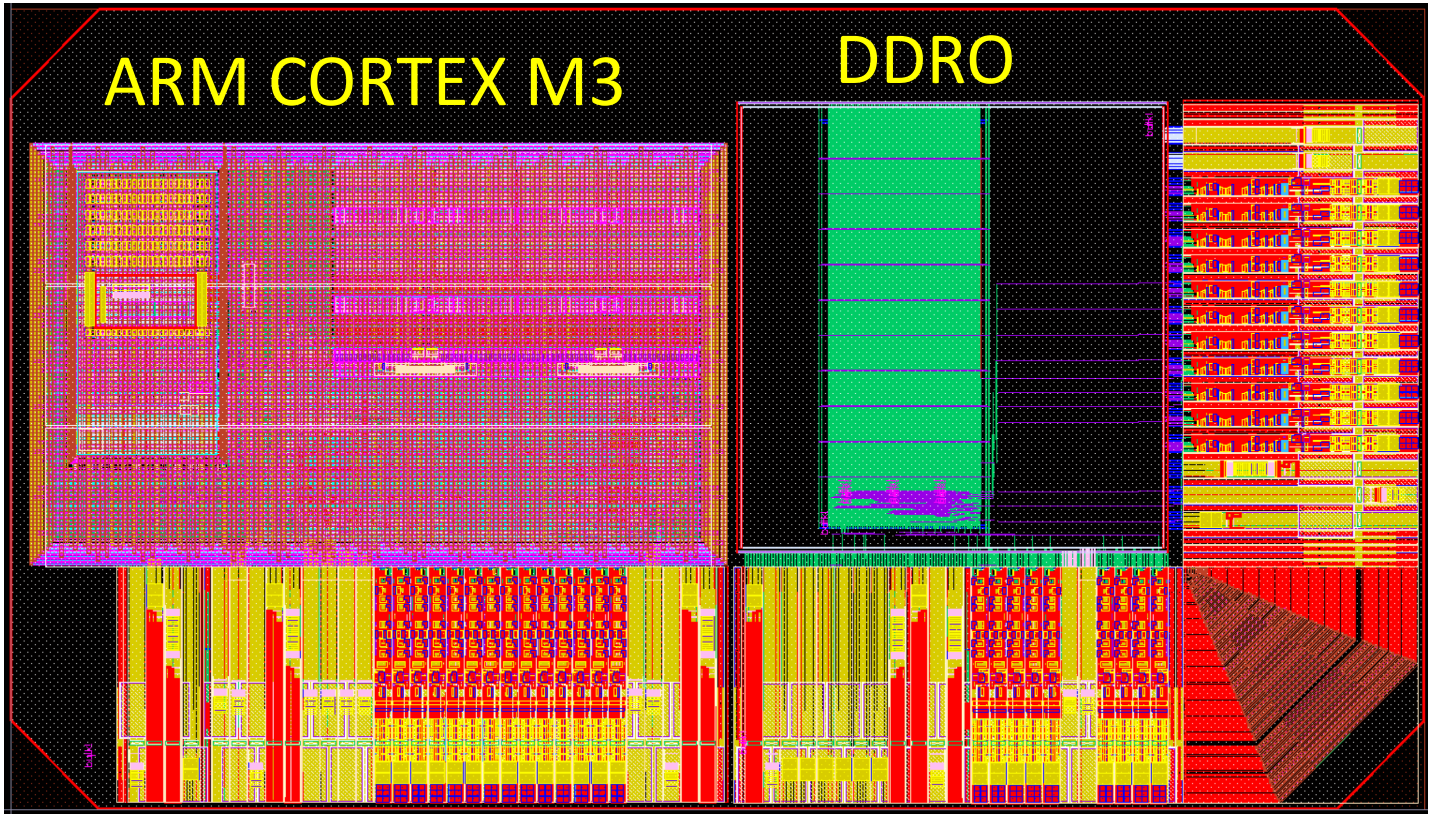
Variability Expedition

NSF

Thank you!

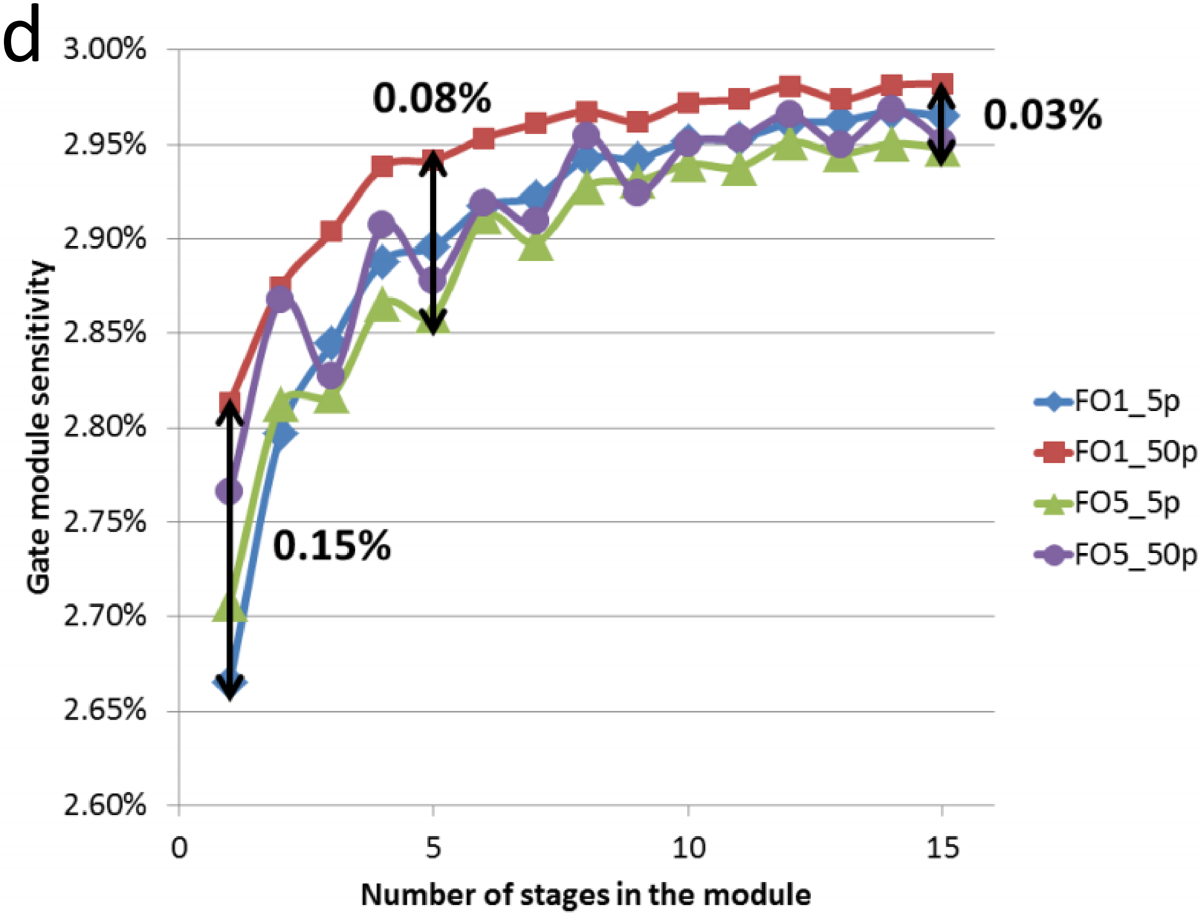
Test Chip

- Test chip tapeout using 45nm technology
  - With an ARM CORTEX M3 Processor



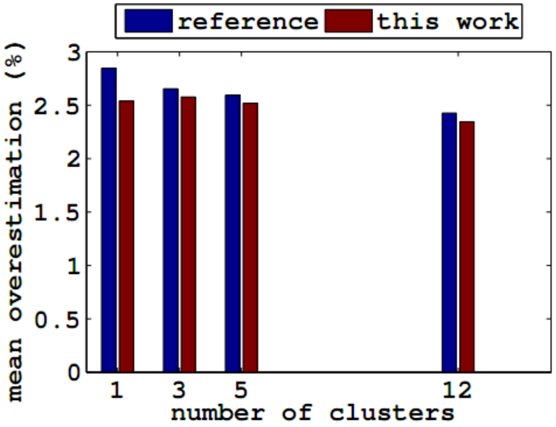
Gate-module

- The delay sensitivities for different input slew and output load combinations.
- Use 5 stages as trade-off between module area and stability

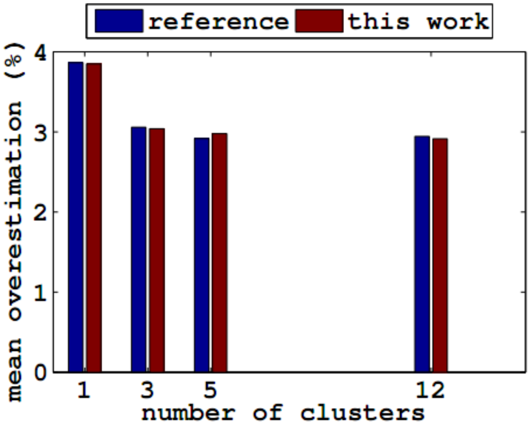


SPICE Results

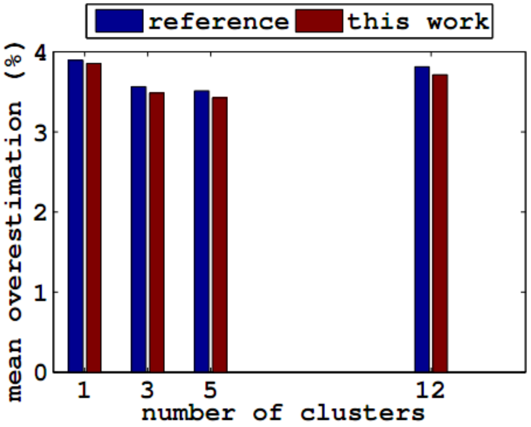
Global and local variations



AES



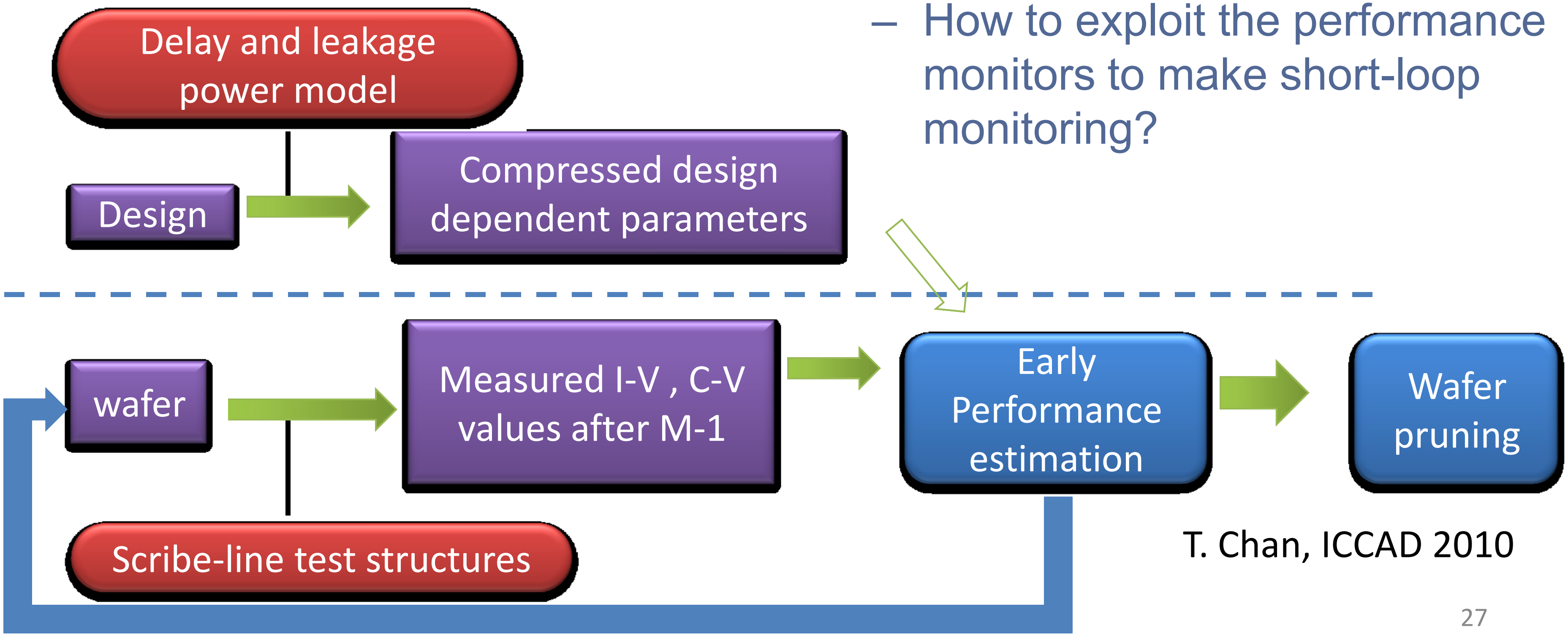
M0



MIPS

Process Tuning

- Circuit performance monitoring is potentially helpful as test structure for manufacturing process tuning
  - How to exploit the performance monitors to make short-loop monitoring?


Delay and leakage power model
Design
Compressed design dependent parameters
wafer
Measured I-V , C-V values after M-1
Early Performance estimation
Wafer pruning
Scribe-line test structures

T. Chan, ICCAD 2010

Existing Monitors

	Generic	**Design-dependent**
Many monitors	N/A	Representative path [Xie10] In-situ monitors [Fick10] Critical-path replica [Black00, Shaik11] In-situ path RO [Ngo10, Wang08]
Multiple monitors	PSRO [Bhushan06] RO [Tetelbaum09]	**This work** TRC [Drake08] Process monitors [Burns08, Philling09]
One monitor	PLL [Kang10]	Representative Path [Liu10]