Why Model Power at the Architectural Level?

- Modeling power at the architectural level allows tradeoffs between HW and SW
- Architectural decisions substantially impact both performance and power
- Architectural decisions often cannot be reversed – must estimate power early

Power-Performance Modeling Infrastructures

Analytical Tools:
CACTI, Wattch, PowerAnalyzer

Mixed-Mode Tools:
Tempest, Accupower

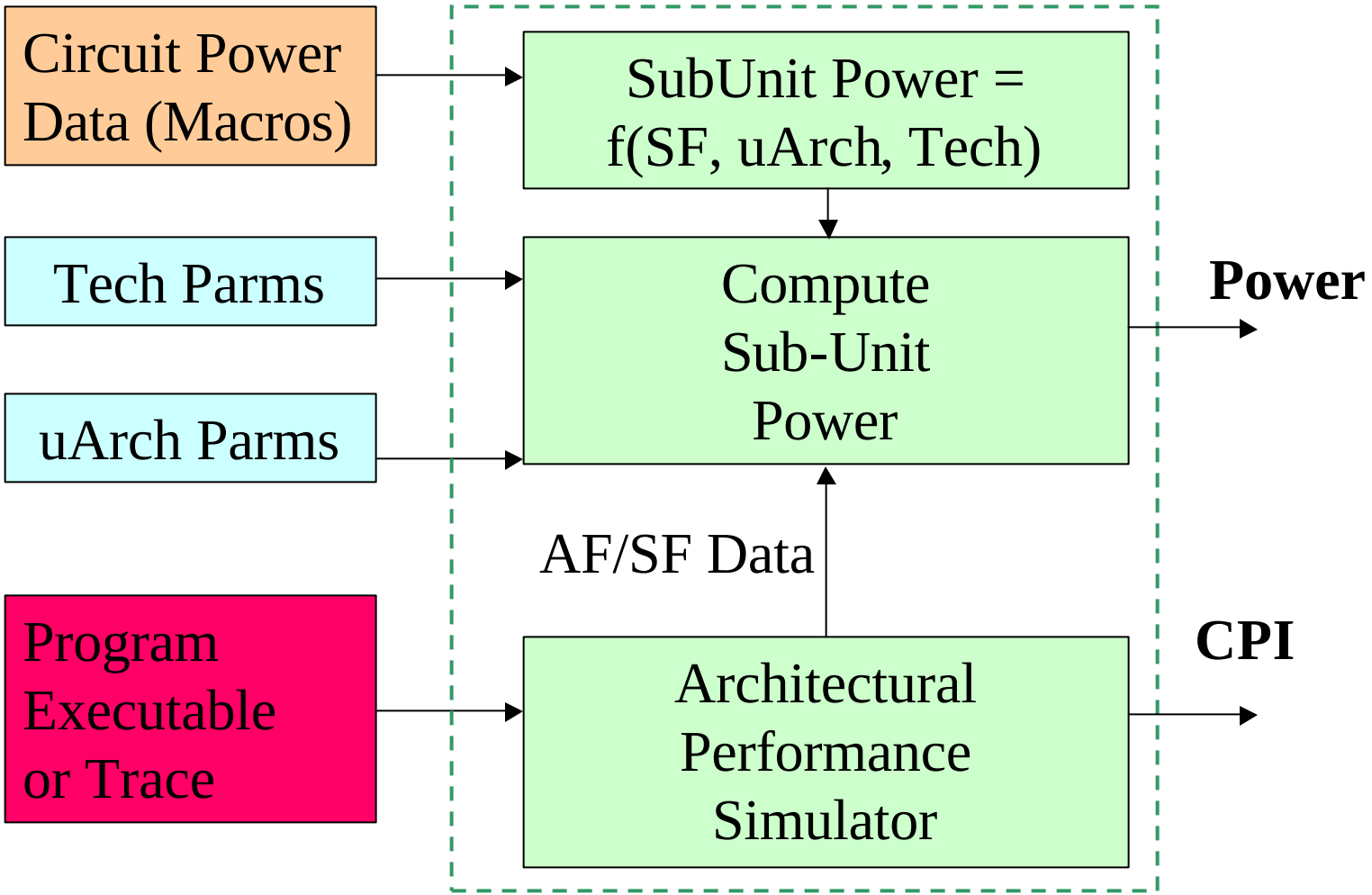
Empirical Tools:
SimplePower, PowerTimer

Increased Flexibility

Ease of Validation

- Existing tools require tradeoffs between flexibility and ease of validation/accuracy

PowerTimer

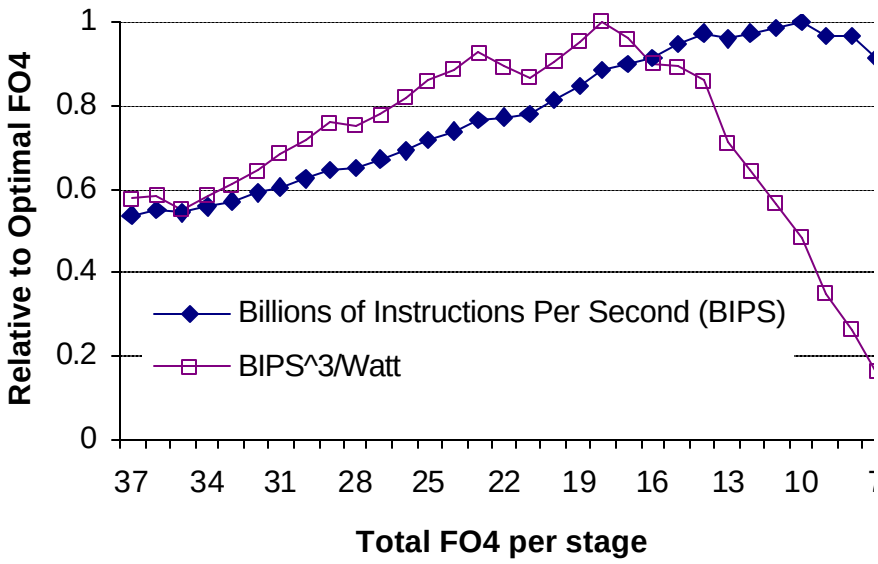

Circuit Power Data (Macros)
SubUnit Power = f(SF, uArch, Tech)
Tech Parms
Compute Sub-Unit Power
Power
uArch Parms
AF/SF Data
Program Executable or Trace
Architectural Performance Simulator
CPI

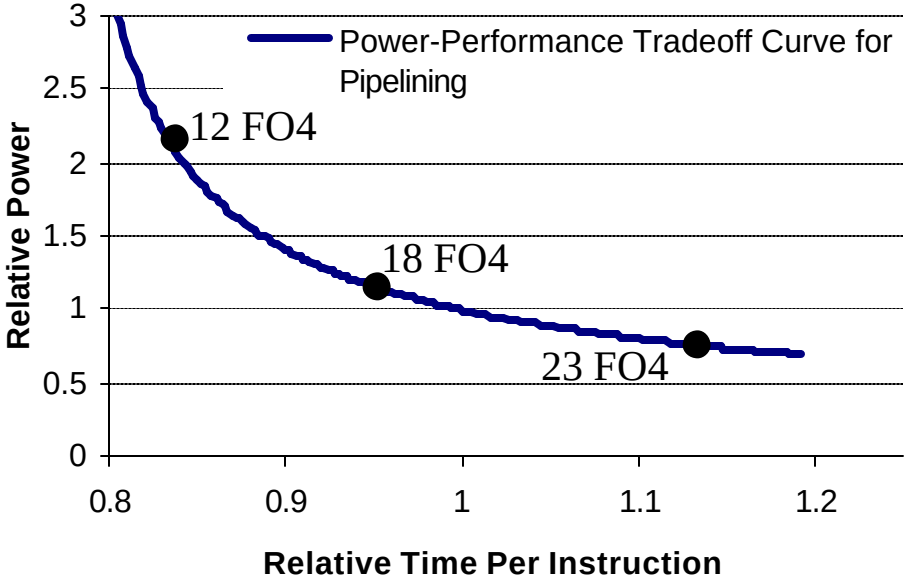
PowerTimer Observations

- PowerTimer works well for POWER4 and derivatives
  - Scales well from base microarchitecture
  - Lack of detailed (bit-level) SF not seen as a problem for high-performance chips (seen as noise)
    - Chip level SFs are quite low (5-15%)
    - Most (60-70%) power is dissipated while maintaining state (arrays, latches, clocks)
    - Much state is not high-level anyway (available in early-stage timers)
  - Validation -- Based on validation of individual pieces
    - We know how to validate the performance model (more or less)
    - Power estimates from circuits are accurate
    - Circuit designers vouch for clock gating scenarios

How are these tools used?

- Architecture and software analysis and tradeoff studies, e.g. optimal pipeline depth



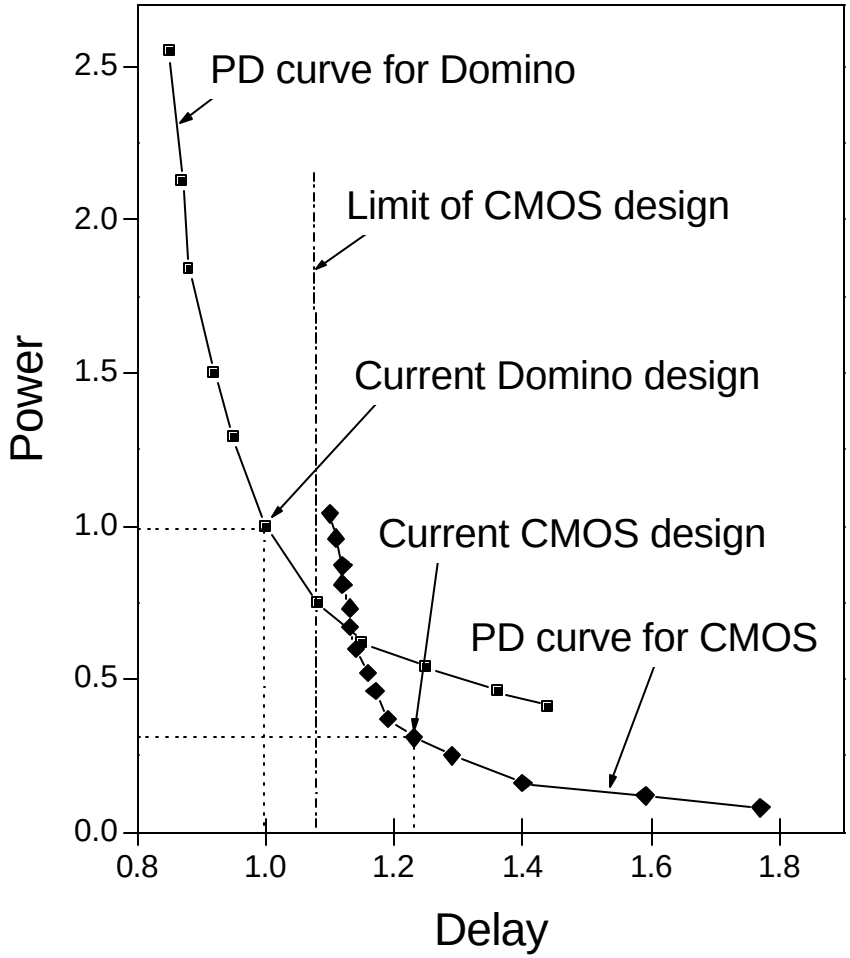


Optimal Pipeline Depth for Microprocessors, from MICRO2002

What are current tools missing?

- Models should account for circuit and architectural level power-performance tradeoffs
- Increased flexibility, accuracy, and ease of validation
- Integrated models for delay, power, and design complexity

Circuit-level Power-Performance Tradeoffs

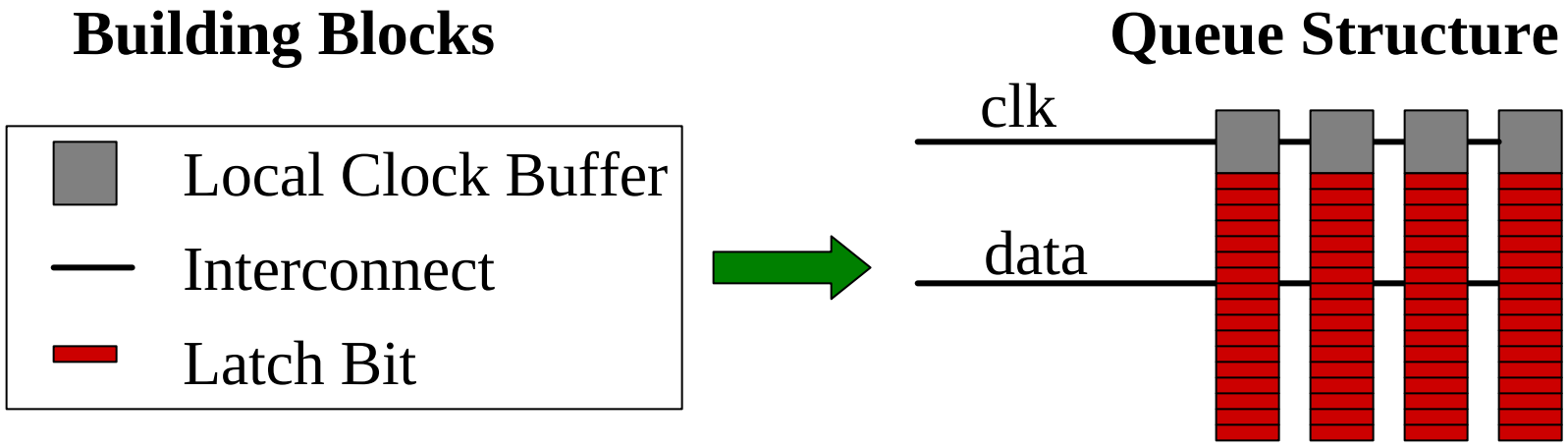


- Circuit/architecture decisions should be made together
- Allows joint-optimization of the power-delay curves

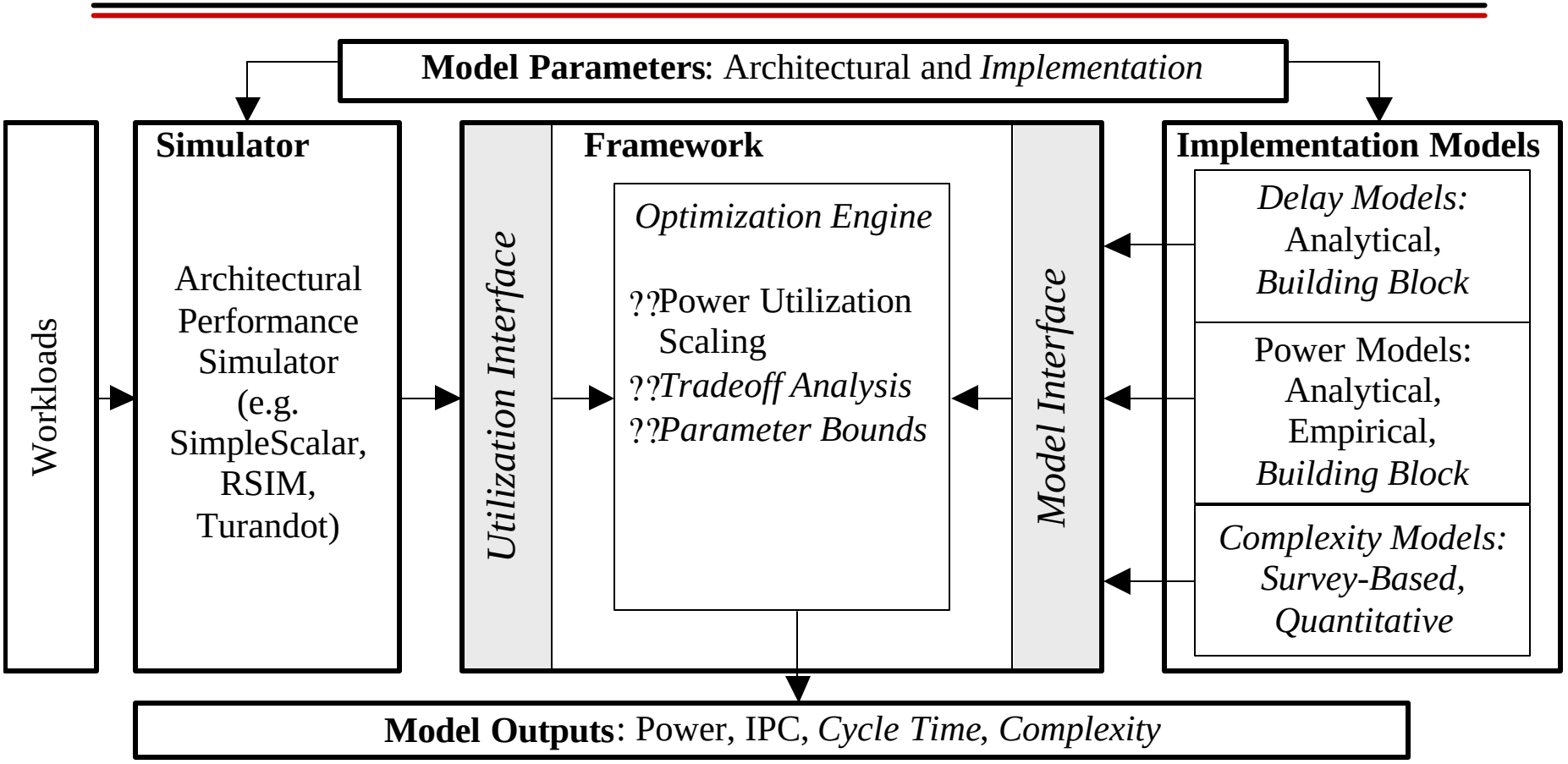
32-bit Adder Power/Delay varying Circuit Style and Transistor Sizings
From Tiwari, et. al. DAC98

Implementation Models

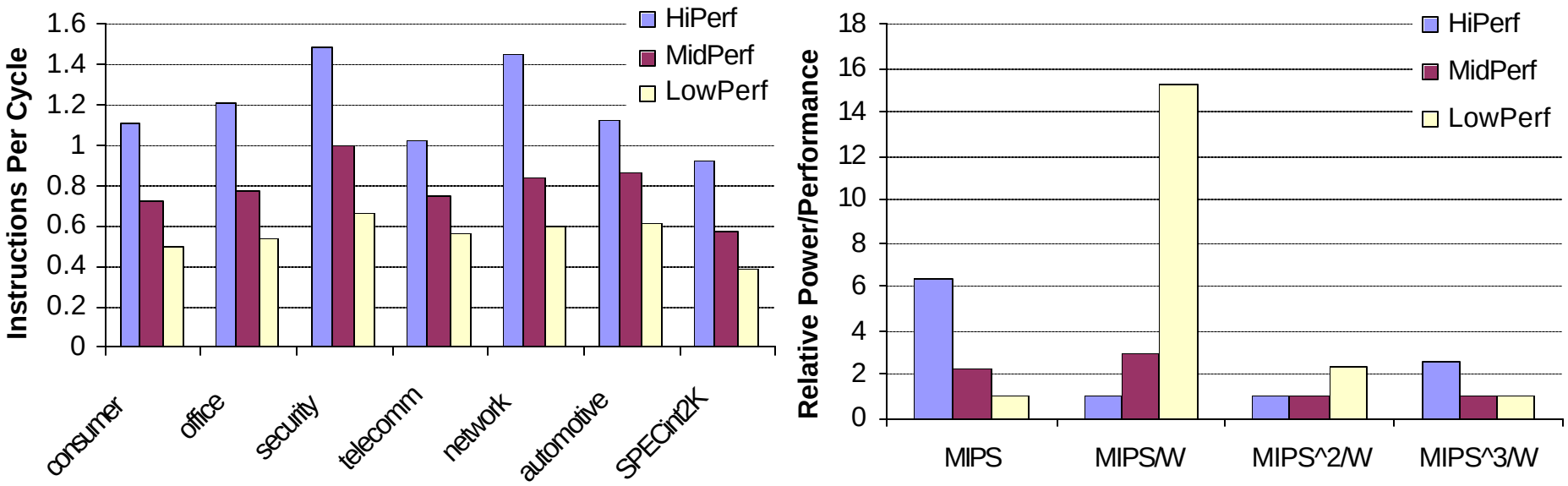
- Building block methodology can provide flexibility with ease of validation
- Develop/Validate models for common intrinsic blocks (latch, mux, interconnect, etc)
  - Chain building blocks together to model higher level structures (issue queue, register file)



Model Framework



Example of Experiments



- Consider three architectures of varying complexity
- Which one is “optimal” for power-performance efficiency?
- What about design points between these choices?

Future Work

- Scalable models for power/performance allow seamless analysis of high-performance embedded architectures
- Development of design complexity models
- Optimization infrastructure to study joint tradeoffs