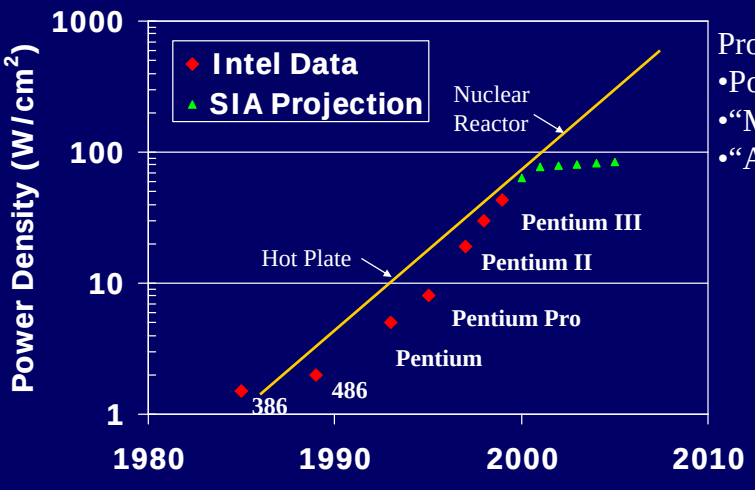
Optimizing Pipelines for Power and Performance

Viji Srinivasan, David Brooks*, Michael Gschwind,
Pradip Bose, Philip Emma, Victor Zyuban, Phil Strenski

IBM T.J. Watson Research Center
Yorktown Heights, NY 10598

*Currently at Harvard University

Power Dissipation Trends



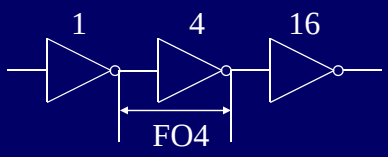
Problems:
- Power Delivery
- "Max" Power
- "Avg" Power

Motivation

- Systematic methodology needed to incorporate circuit power in the concept phase of a microarchitecture
  - Estimate and refine cycle time targets and power costs
- Parameterized power-performance models needed to:
  - Allow progressive refinements
  - Be useful in the pre-simulation phase
- One of the most critical concept-phase design decisions is to choose a pipeline depth (FO4 per pipe stage)

Background/Definitions

- Fanout-of-4 inverter metric (Horowitz)

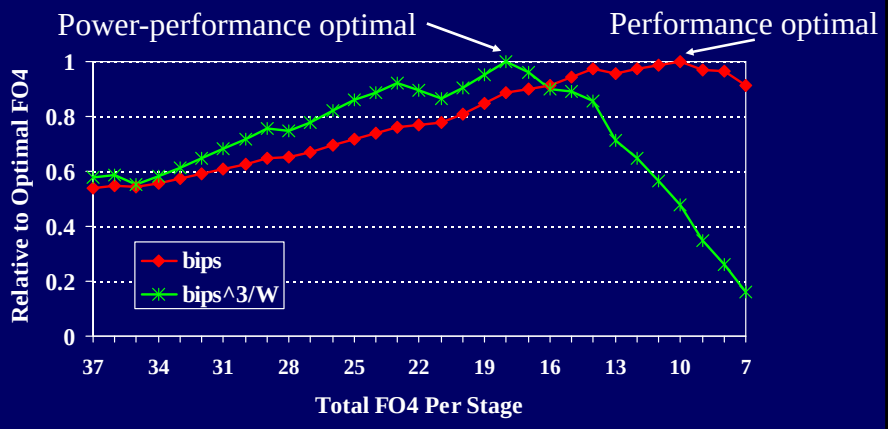


- Delay of an inverter with $C_{load}/C_{in}=4$
- More or less stable for process, voltage, temperature
- We use this to measure amount of logic per stage of the pipeline

Our Contribution

- (Recent) Prior Work -- ISCA2002
  - Three research groups studied performance-optimal pipeline depth
  - Different ISAs, different microarchitectures, different benchmarks, different simulators
  - Optimal Performance-Depth is 7-10FO4
- Our focus:
  - Intuitively, complexity/power concerns will favor shallower pipes
  - But how to quantify?

Power Matters!



Based on detailed cycle accurate simulation

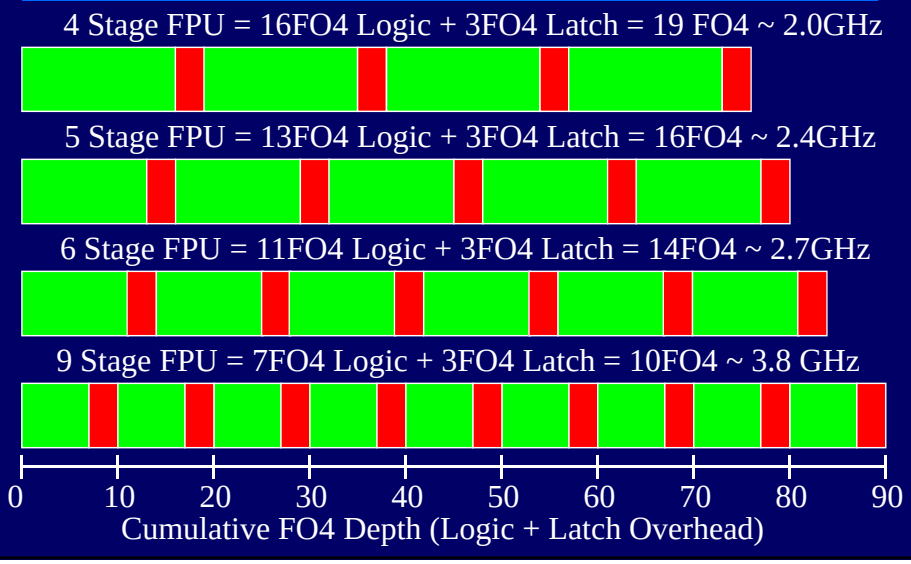
Outline

- Power-Performance modeling methodology
- Scaling Energy Models for Pipeline Depth
- Sensitivity Analysis
- Impact and Conclusions

Overall Methodology

- Start with a simple analytical pipeline model
  - Study Pipeline Optimization using workload characterization
  - Useful in the pre-simulation concept phase design
- Validate the analytical model with detailed cycle accurate simulation
- Develop energy models based on detailed circuit-level power analysis of macros
- Develop energy scaling equations for pipeline depth
- Study the sensitivity of the energy model parameters to the optimal pipeline depth

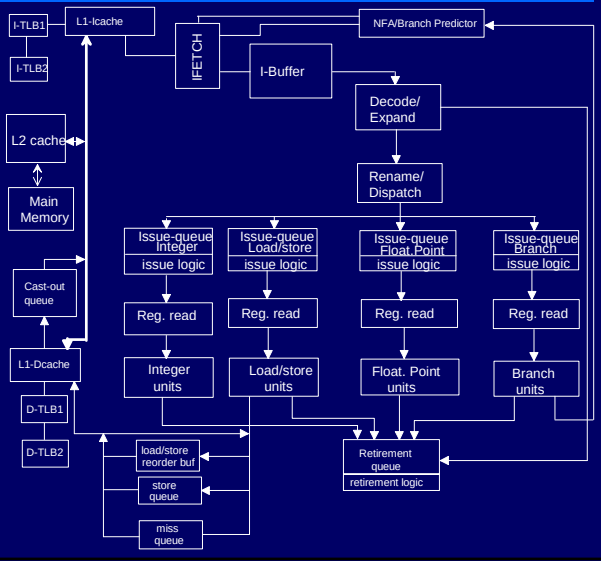
Pipeline Scaling Methodology



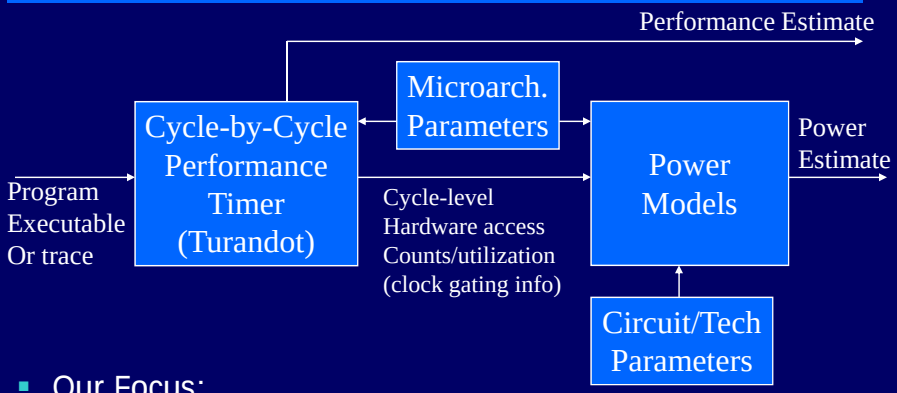
Detailed Performance Simulation: Turandot

Base Model:
PowerPC ISA
Out-of-Order
8-issue SuperScalar

Calibrated vs
Pre-RTL latch
accurate processor
model



PowerTimer



- Our Focus:
  - How to scale energy models for pipeline *depth* rather than pipeline *width*

Energy Model Formation

- Energy models based on circuit-level power analysis of structures in current high-performance PowerPC processor
- Power analysis
  - For each macro collect *ungated* power (ckt sim)
    - Clocking power (latches, LCBs, array clocking)
    - Active power (Logic, data-dependent array)
    - Leakage power
  - Clock gating factors determined based on utilization and macro-level clock gating eligibility

Factors Affecting Choice of Pipeline Depth

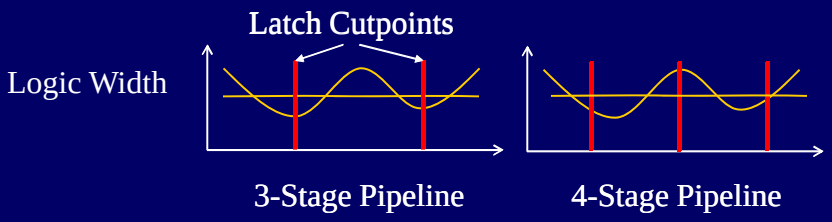
- Cycles-Per-Instruction (CPI)
- Clock Frequency
- Clock Gating Effects
- Latch-to-Logic Dynamic Power Ratio
- Latch Growth Factor
- Glitching Activity
- Leakage Power Scaling
- Power-Delay Ratios for Latches and Logic

Energy Model Scaling: CPI, Frequency, Clock Gating

- CPI impacts performance only (workload dependent)
- Clock Gating impacts power only (workload dependent)
- Frequency impacts both

Energy Model Scaling: Latch Growth Factor, Latch-Logic Ratio

- Latch growth has a big impact
  - Logic shape functions are often not flat

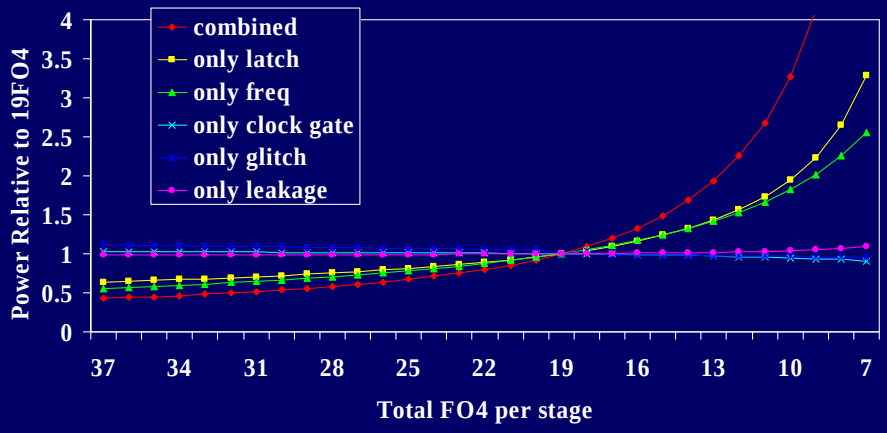


- LatchScale = (Latch-logic power ratio) * (base FO4/FO4)^{LGF}
- Latch Growth Factor slightly super-linear (1.1)
- Latch-Logic Power Ratio of current machines (70%-30%)

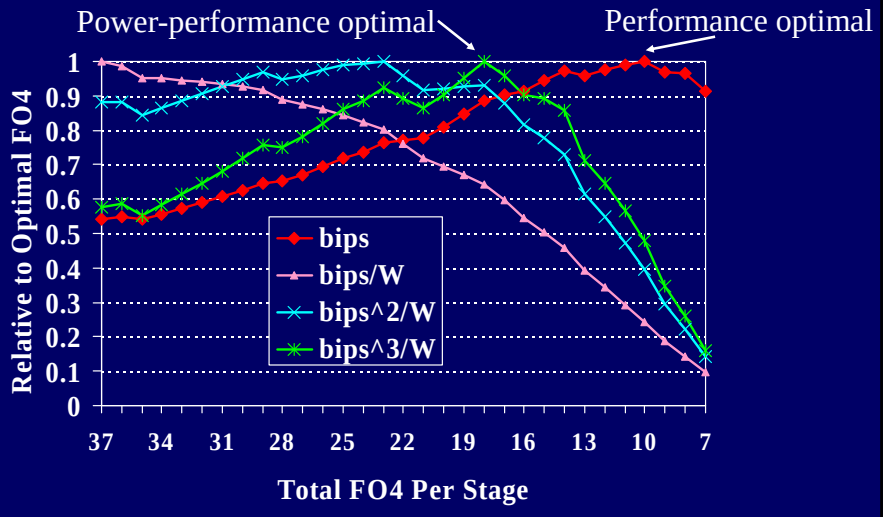
Energy Model Scaling: Glitching and Leakage

- Glitching *reduces* with deeper pipelines
  - More pipeline latches stop glitch propagation
- Leakage power component grows more slowly than dynamic power component with deeper pipelines
  - Leakage does not scale with frequency
  - Leakage growth is proportional to overall width of latches rather than overall power of latches
    - Overall Latch width % << Overall Latch power %

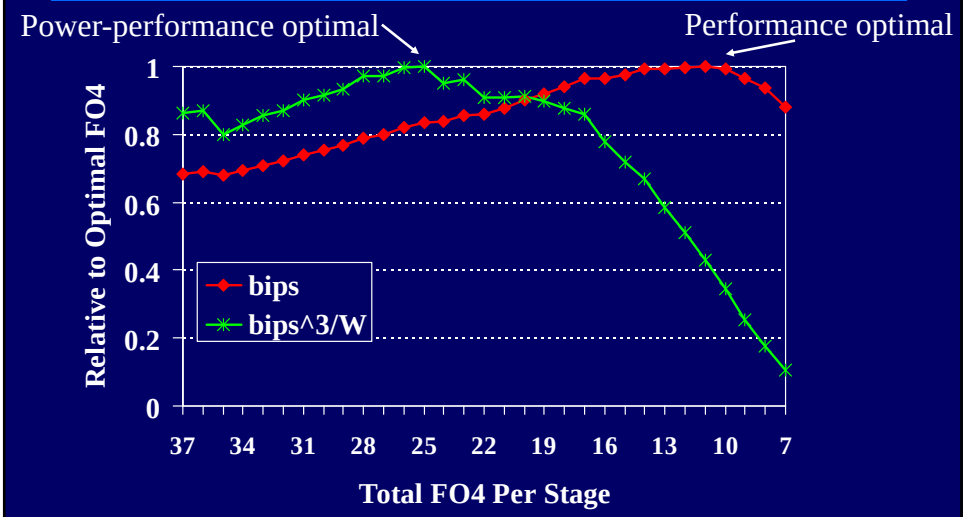
Power Scaling Effects



Scaling Results: Average of SPEC2K



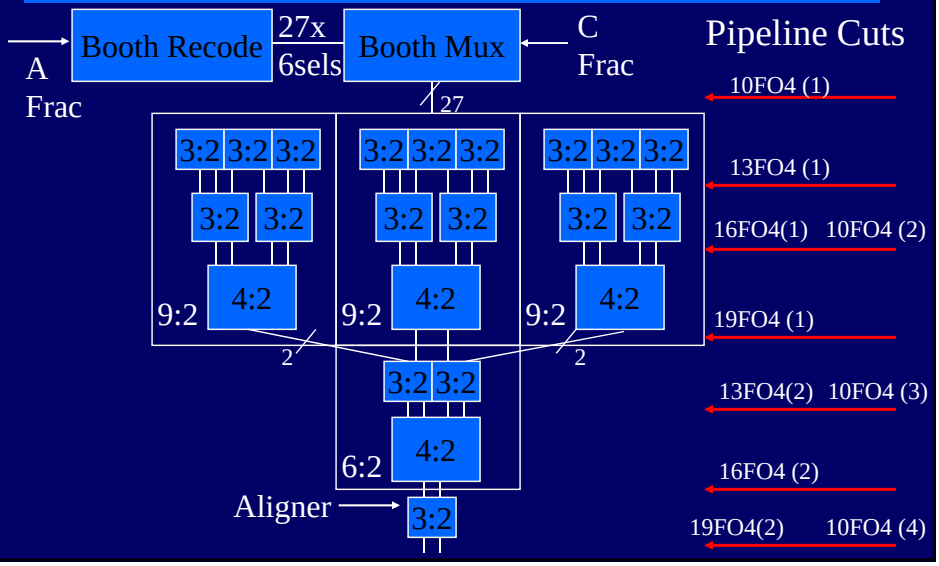
Workload impact: TPCC Trace



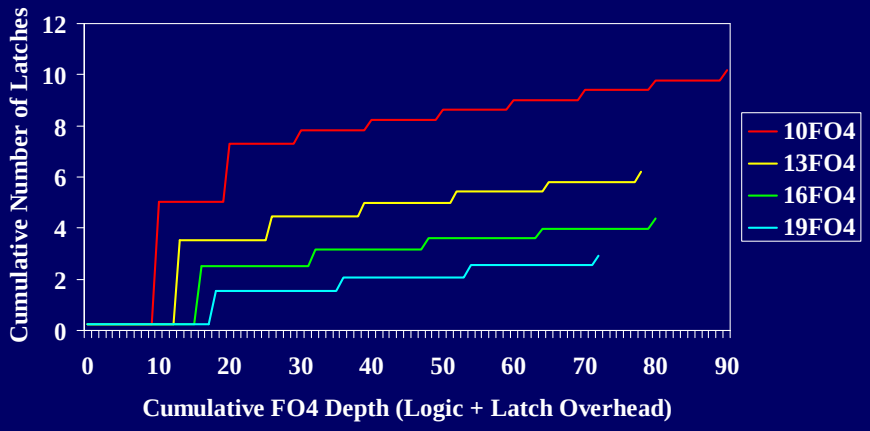
Sensitivity Analysis

- Baseline model discussed
- Now we consider sensitivity of the results to uncertainty in the parameters
  - Either difficult to measure (LGF, glitching, etc)
  - May vary from processor to processor (latch vs. logic power, leakage ratio)

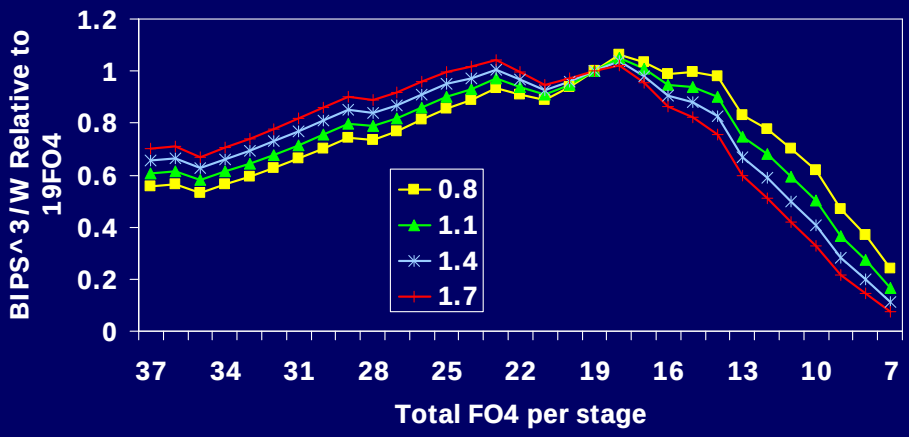
Sensitivity Analysis: Latch Growth Factor



Sensitivity Analysis: Latch Growth Factor



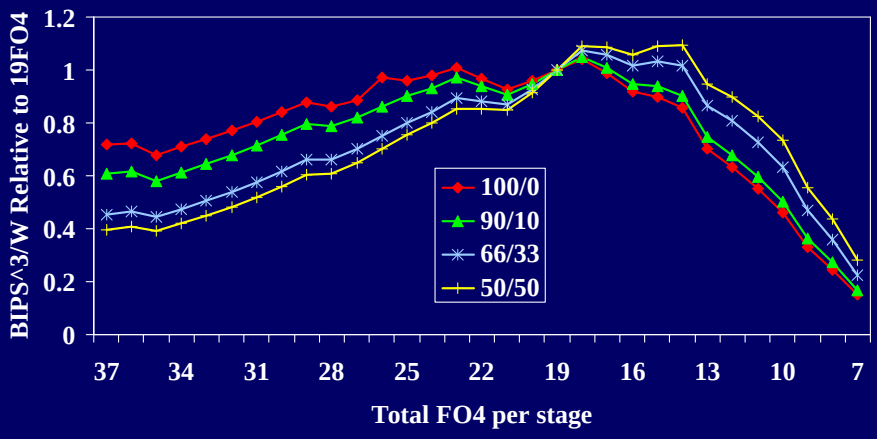
Sensitivity Analysis: Optimality varying Latch Growth Factor



- LatchScale = (Latch-logic power ratio) * (base FO4/FO4)^{LGF}

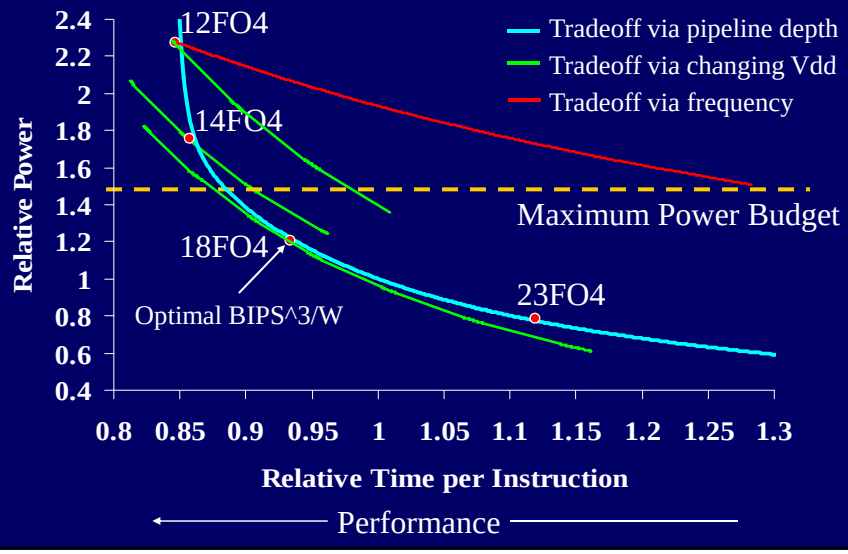
Vary the LGF exponent (1.0=linear)

Sensitivity Analysis: Varying Leakage Ratios



Vary the ratio of leakage to active power

Impact on Design



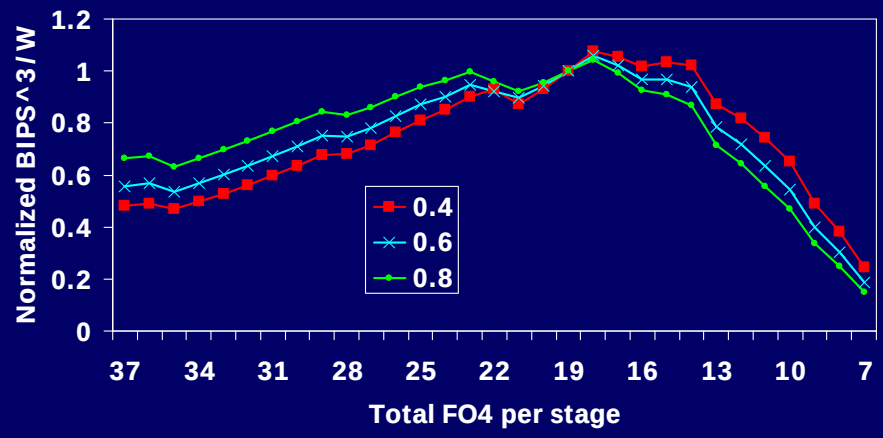
Conclusions

- Concept phase key decisions significantly impact power-performance of machine
- Deeper pipelines are one way to improve performance (but only to a point)
  - Going for the smallest FO4 and highest Vdd upfront ➔ may lead in the end to unacceptably low net performance as Vdd or frequency is lowered to contain power!
  - Considering power and performance together ➔ leads to more balanced choice of FO4 depth at nominal Vdd, and leaves room for post-tapeout growth in performance via Vdd scaling and circuit tuning

Backup Slides: Default Latency Values

Fetch Latencies		Decode Latencies		Exe Pipe Latencies		Mem Latencies	
Parameter	Cycles	Parameter	Cycles	Parameter	Cycles	Parameter	Cycles
NFA Predictor	1	Multiple Decode	2	Integer Execute	1	L1 D-Load	3
L2 Icache	11	Millicode Decode	2	Float Execute	4	L2 D-Load	9
L3 (Instr.)	85	Expand String	2	Branch Execute	1	L3 (Data)	77
I-TLB Miss	10	Mispredict Cycles	3	Float Divide	12	Load Float	2
L2 I-TLB Miss	50	Register Read	1	Integer Multiply	7	D-TLB Miss	7
				Integer Divide	35	L2 D-TLB Miss	50
				Retire Delay	2	StoreQ Forward	4

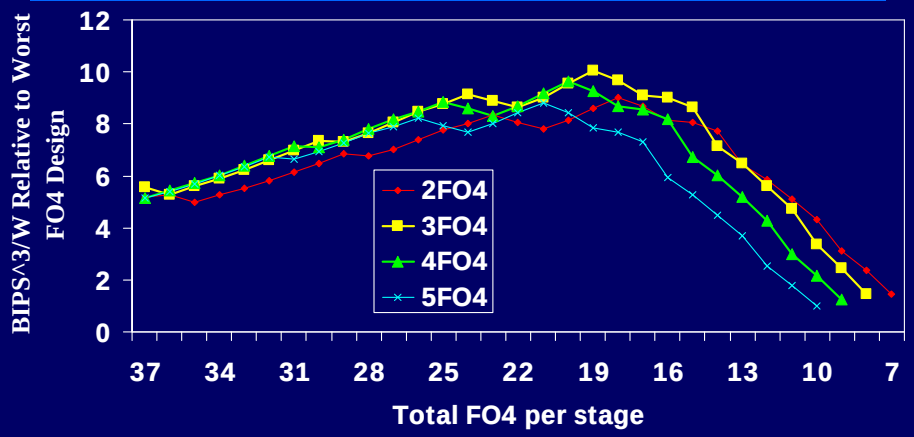
(Backup) Sensitivity Analysis: Varying Latch-Logic Power Ratio



LatchScale = LatchRatio * $(FO4^{base}/FO4)^{LGF}$

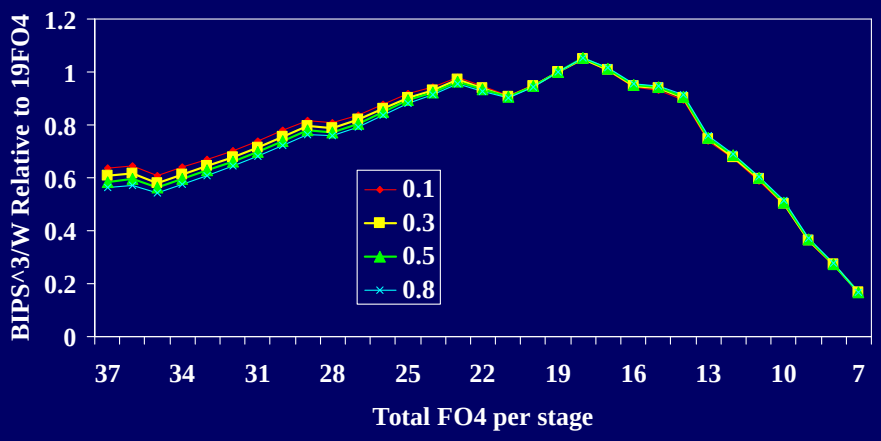
Vary the amount of power spent in latch vs. logic (0.7 = 70%)

(Backup) Sensitivity Analysis: Varying Power-Delay Ratios

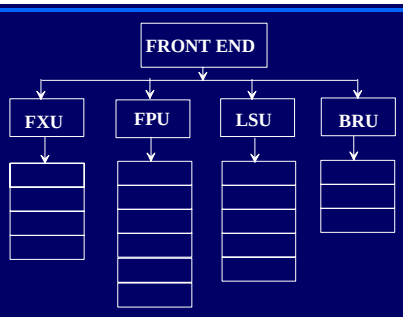


Latch FO4	2	3	4	5
Relative Clocking Energy	1.0	0.62	0.49	0.43

(Backup) Sensitivity Analysis: Varying Glitching Activity



(Backup) Analytical Model (pre-simulation)



stages:	s1	s2	s3	s4
completion time: (no latches)	t1	t2	t3	t4
latch delay/stage:	c1	c2	c3	c4

- Workload analysis determines data dependent stalls
- Throughput $G = \frac{u_1}{T_{fxu}} + \frac{u_2}{T_{fpu}} + \frac{u_3}{T_{lsu}} + \frac{u_4}{T_{bru}}$

(Backup) Analytical Model vs. Simulation