Design Automation Conference
3 June 2013

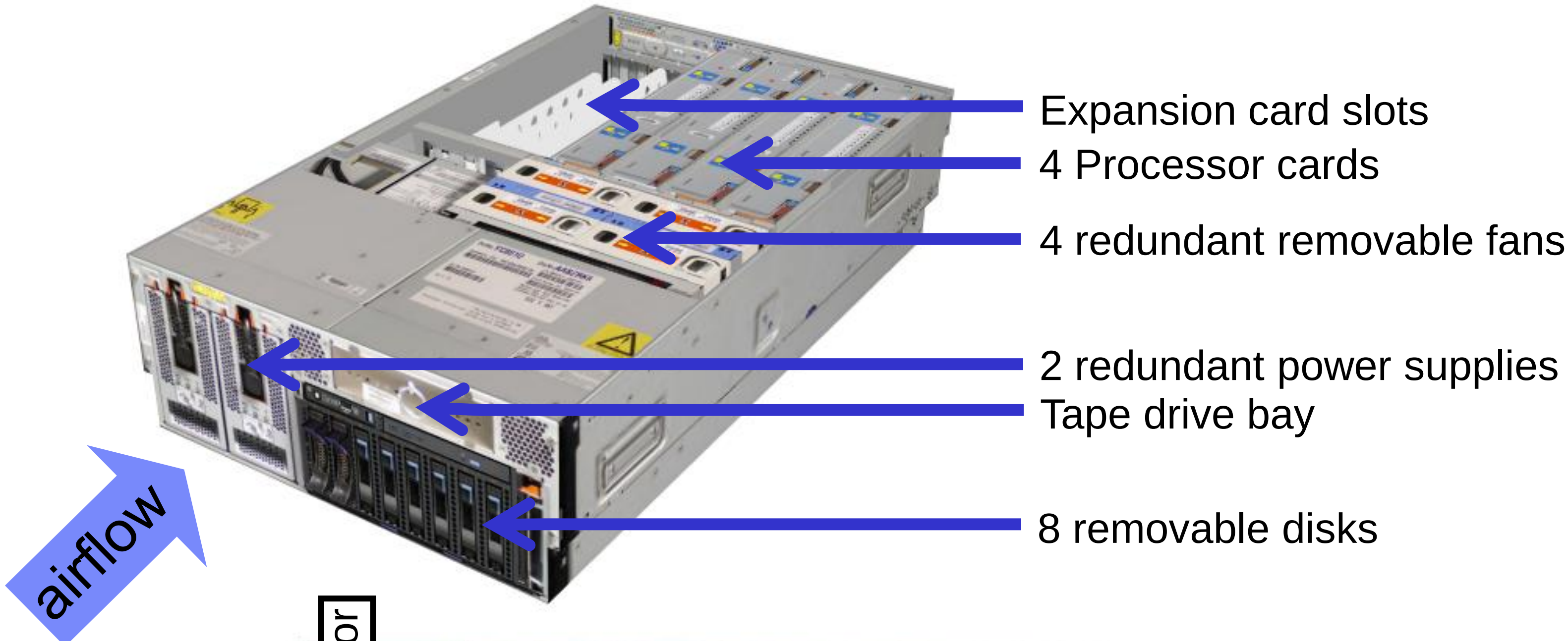
Adaptive Energy Management in POWER7 and POWER7+

Charles Lefurgy

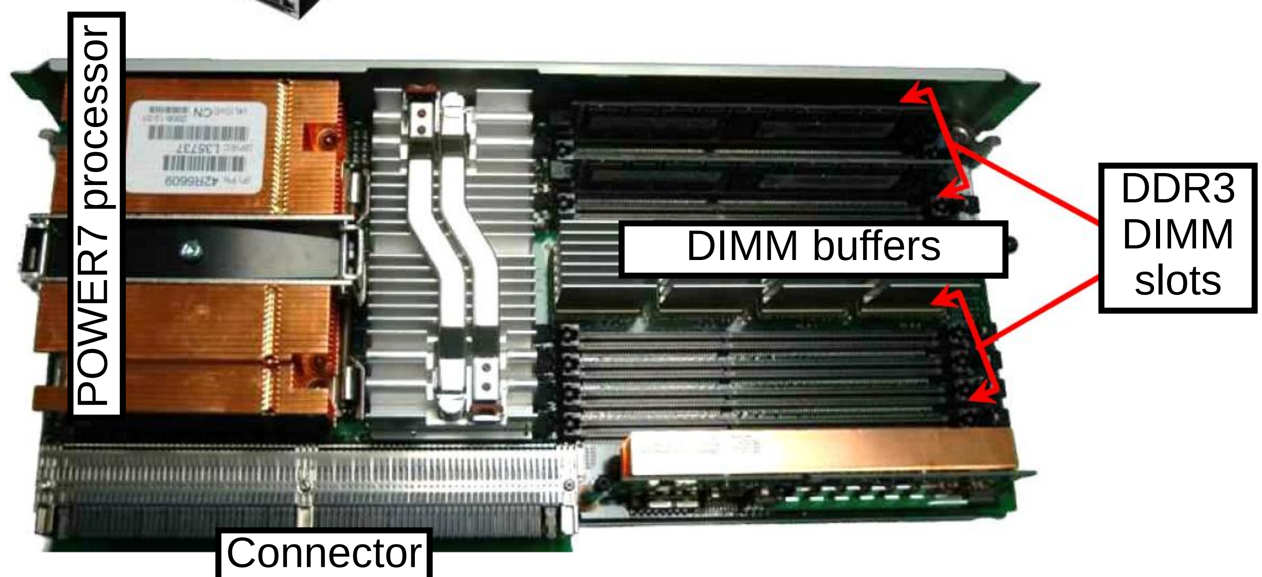
Outline

- Overview of POWER7 and POWER7+ energy management
  – Feedback control
  – Sensors
  – Actuators
- Problem #1: Server power capping
- Problem #2: Excess guardband

IBM POWER 750 Express

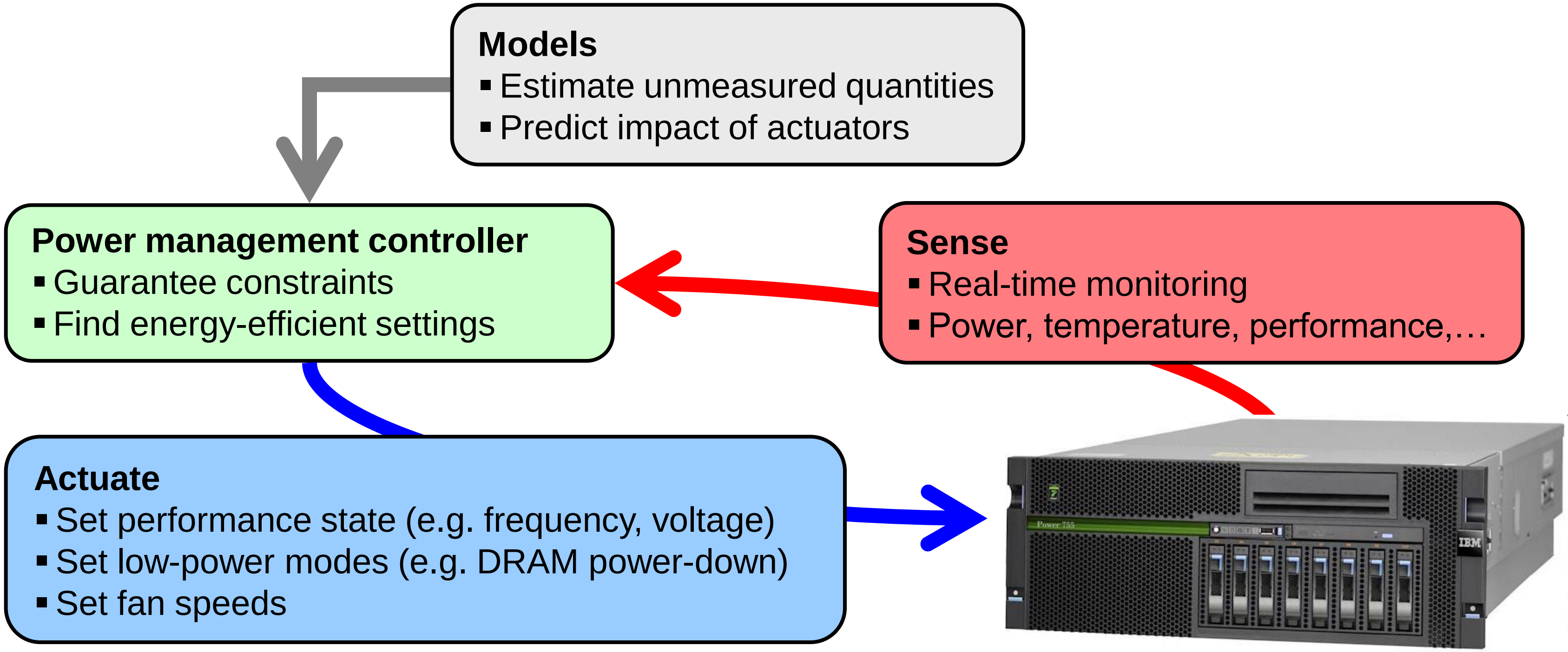


Processor card



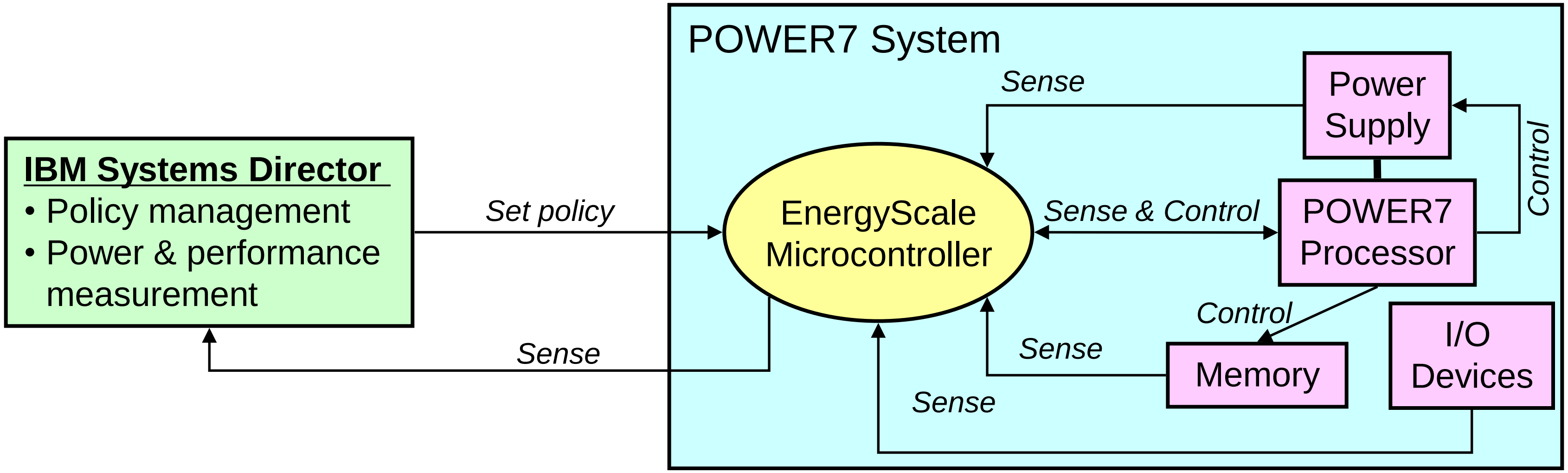
Address variability in hardware and operating environment

- Complex environment
  - Installed component count, ambient temperature, component variability, etc.
  - How to guarantee power management constraints across all possibilities?
- Feedback-driven control
  - Capability to adapt to environment, workload, varying user requirements.
  - Regulate to desired constraints even with imperfect information.



EnergyScale

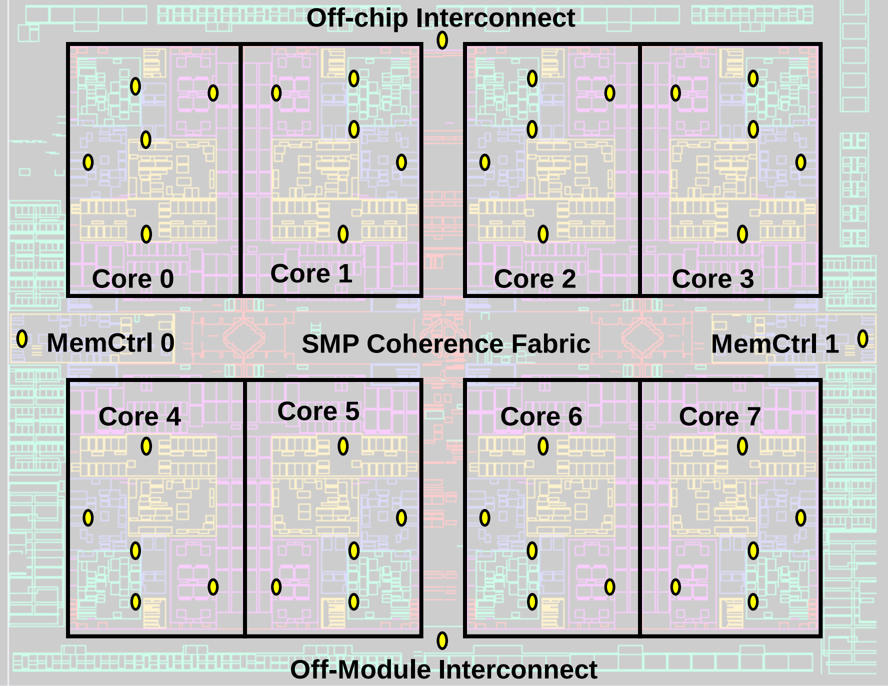
- Cooperative hardware and software solution for power management.
  - EnergyScale firmware runs on dedicated microcontroller.
    - DVFS, thermal control, power capping, guardband management, etc.
  - POWER7 microprocessor has hardware accelerators for power management.
    - Sensor gathering, thermal sensor conversion, power proxy calculation, etc.
- Goals
  - Increase performance.
  - Reduce power consumption while maintaining performance.



POWER7 sensors

- Microarchitecture activity & event counters
  - Provide performance, utilization, and activity measurements
  - Processor core, memory hierarchy, and main memory access
  - Per-thread utilization and per-core memory bandwidth (POWER7+)
- Digital Thermal Sensors
  - 44 on-chip sense points
- Critical Path Monitor
  - Detects circuit timing margin
- Power proxy
  - Estimate core power based on event counters
- System sensors
  - Fan speed
  - Power by voltage domain
  - Temperature by component
  - Ambient temperature

Physical Locations of Thermal Sensors



POWER7+ power proxies

- <u>Chip-level</u> and <u>core-level</u> power proxies.
- Per-core HW computes **activity proxy.**
  – Based on 50 activity counters.
  – Every 32 ms.
- Tracks change in voltage, frequency, temperature, and workload activity.
- POWER7+ Vdd power proxy has a mean error of 0.2% (2.6% std dev).

Active power model

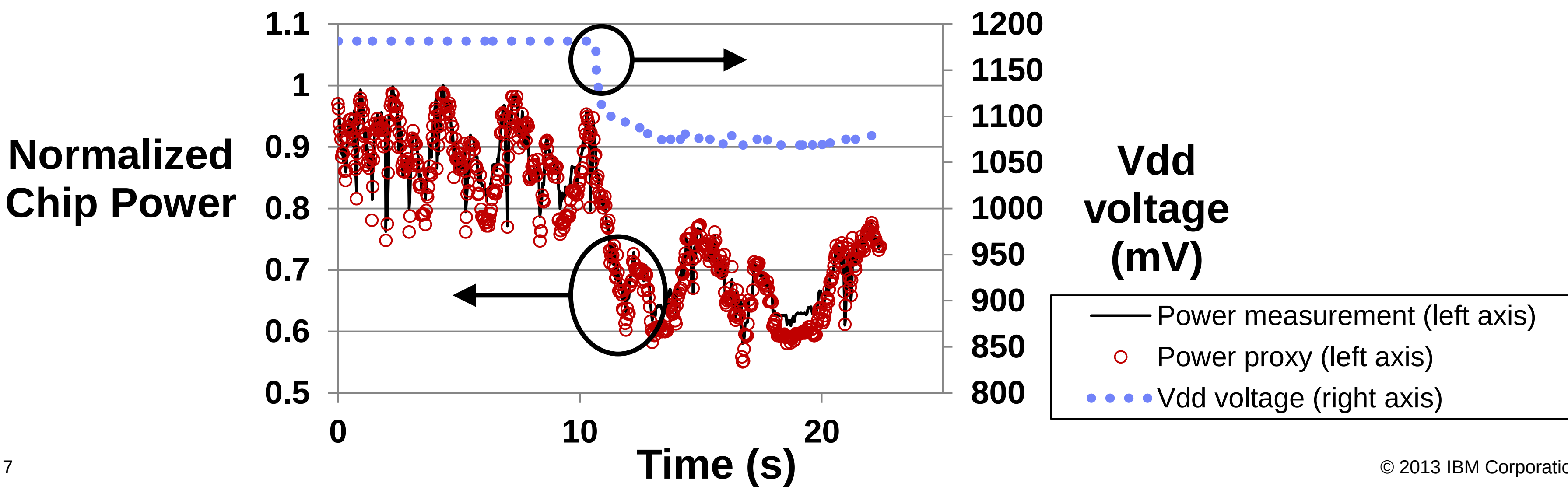
$$P_{active} = \frac{ActivityProxy}{R_0}\left(\frac{V}{V_{nom0}}\right)^{\alpha}$$

$$ActivityProxy = \sum\left(W_g \times \sum\left(W_{i_g} \times A_{i_g}\right)\right)$$

Idle power model

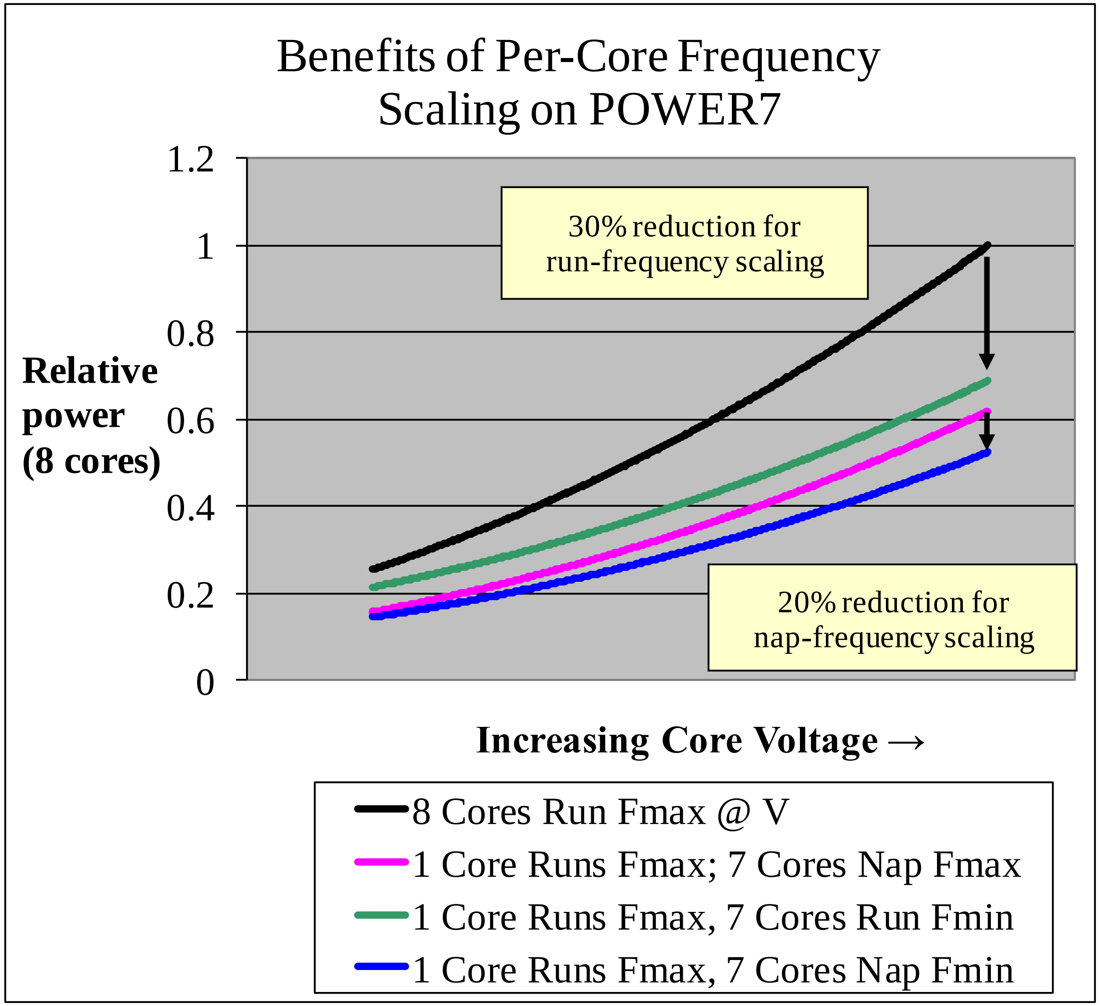
$$P_{idle} = P_{clock} + P_{leak}$$

$$= \frac{F}{S_0}\left(\frac{V}{V_{nomo}}\right)^{\beta} + P_{leak\_nom}\left(\frac{V}{V_{nom}}\right)^{\gamma}\left(1 + m_0(T - T_0)\right)$$



Actuators

- Per-chip voltage selection
- Per-core frequency control
  - Digital PLL (DPLL) clock source supports -50% to +10% of nominal frequency
  - 25 MHz resolution
  - Automated fast frequency slew in excess of 50 MHz/µs
- Core + L2 cache and L3 cache power gating (POWER7+)
- Idle modes: nap, sleep, winkle
- Memory throttling
- Fan speed
- Each partition (group of cores) may use a different energy-savings policy
  - Highly utilized partitions maintain peak performance
  - Less utilized partitions run at lower frequencies



Note: highest frequency core determines the required voltage

Summary

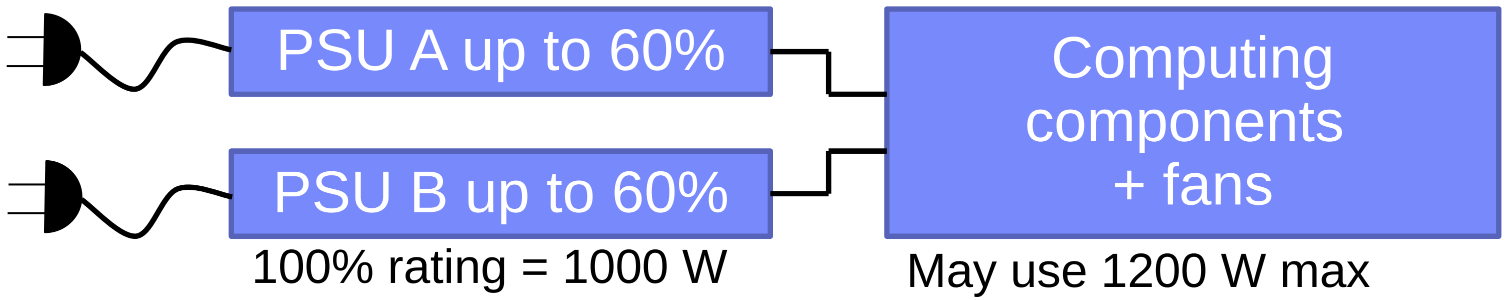
- POWER7 systems contain a rich set of sensors, actuators, and on-chip accelerators which enable the feedback controllers that manage energy use.

- POWER7 energy management features combined with new energy-saving algorithms show a **50%** improvement in SPECpower_ssj2008 score over baseline operation.

- Customers can select the best EnergyScale policy to match their needs, relying on the system to balance power consumption and performance accordingly.

Problem #1: Server power capping

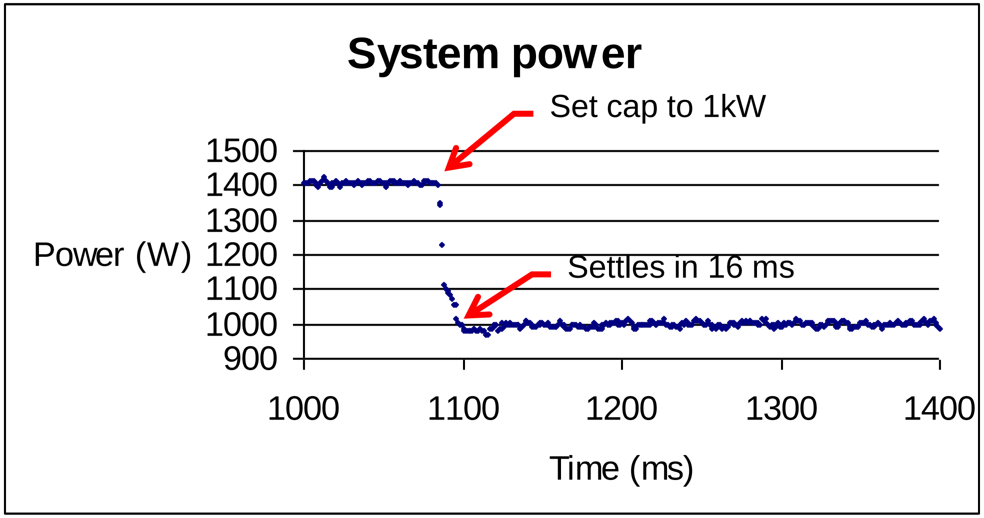
- Servers use redundant power supply units (PSU) for reliability.
- Example: Each PSU may use at most 60% of its rating.

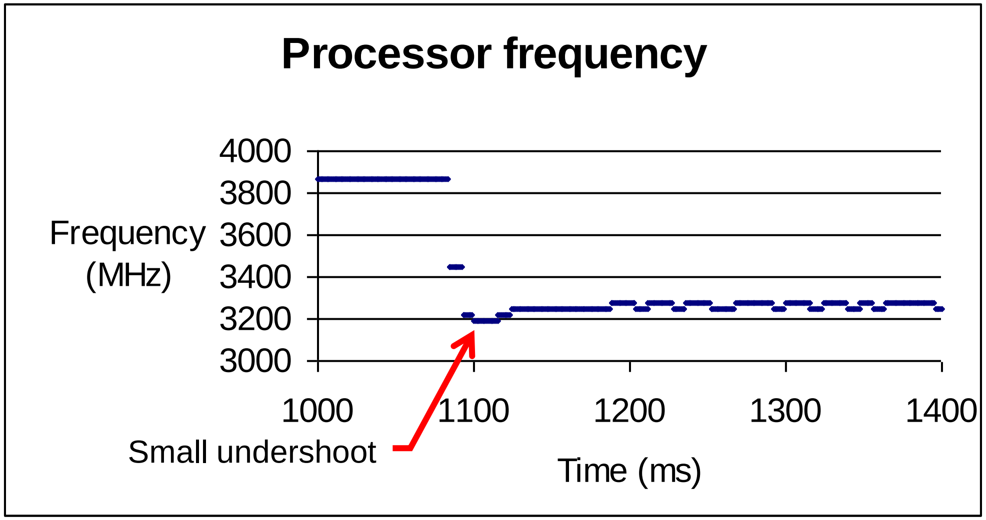


- Allows up to 120% of single-supply power to be used when both PSUs working.
  - Benefit: higher performance than using a single power supply.
- When a PSU fails, the load shifts to the remaining PSU (up to 120%).
  - Remaining PSU must reduce load to 100% rating quickly, or risk shutdown.
  - Time frame ranges from milliseconds to seconds (depends on PSU specification).
- **Power capping** is a method to control peak power consumption.
  - Objective 1: respond quickly to avoid shutdown of remaining PSU.
  - Objective 2: maximize performance within the remaining power supply limits.

Power capping controller in POWER7

- Capping situations.
  1) redundant power supply failure.
  2) customer sets power cap target.
- Control interval is 8 ms
  – Measure system power and adjust processor voltage and frequency to meet power cap.
- Power settles within 120 ms time constraint to avoid loss of remaining PSU.
- Partition-aware capping
  – Objective: Keep performance sensitive workloads at high performance.
  – Partitions are sorted based on their performance guarantees and current core clock frequency.
  – For example, turbo frequency is not guaranteed when a PSU fails.





Walk-through of first power-capping server in the industry

- IBM HS20 (Intel Xeon) blade, 2006
  - Uses clock throttling to adjust performance
    - 8 performance levels from 12.5% (slowest) to 100% (fastest)
- Settle to within 0.5 W of desired power in 1 second
  - Based on BladeCenter power supply requirements
- Note: POWER7 uses dynamic voltage and frequency scaling instead of throttling.

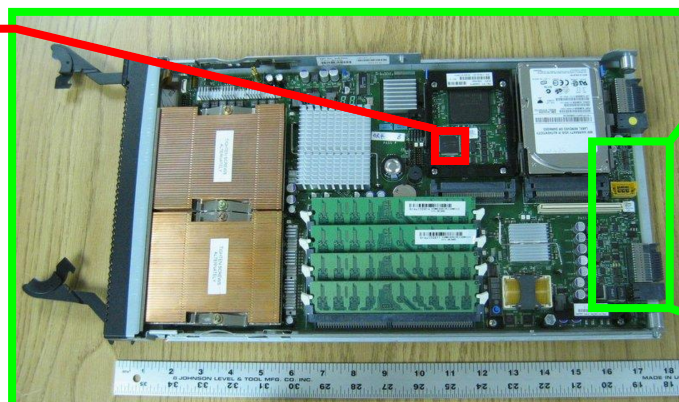
Measure 12V bulk power

0.1 W precision, 2% error

Measurement/calibration circuit

Sense resistors

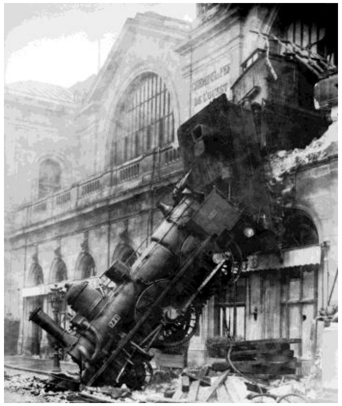
Controller firmware on service processor (Renesas H8 2168)



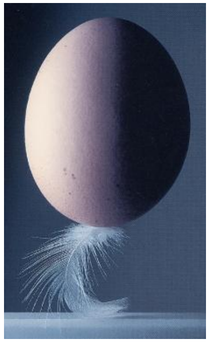
HS20 8843 (Intel Xeon blade)

Control options for power capping

Open-loop
- No measurement of power.
- Chooses fixed processor speed for a power budget.
- Based on worst-case power consumption workload.



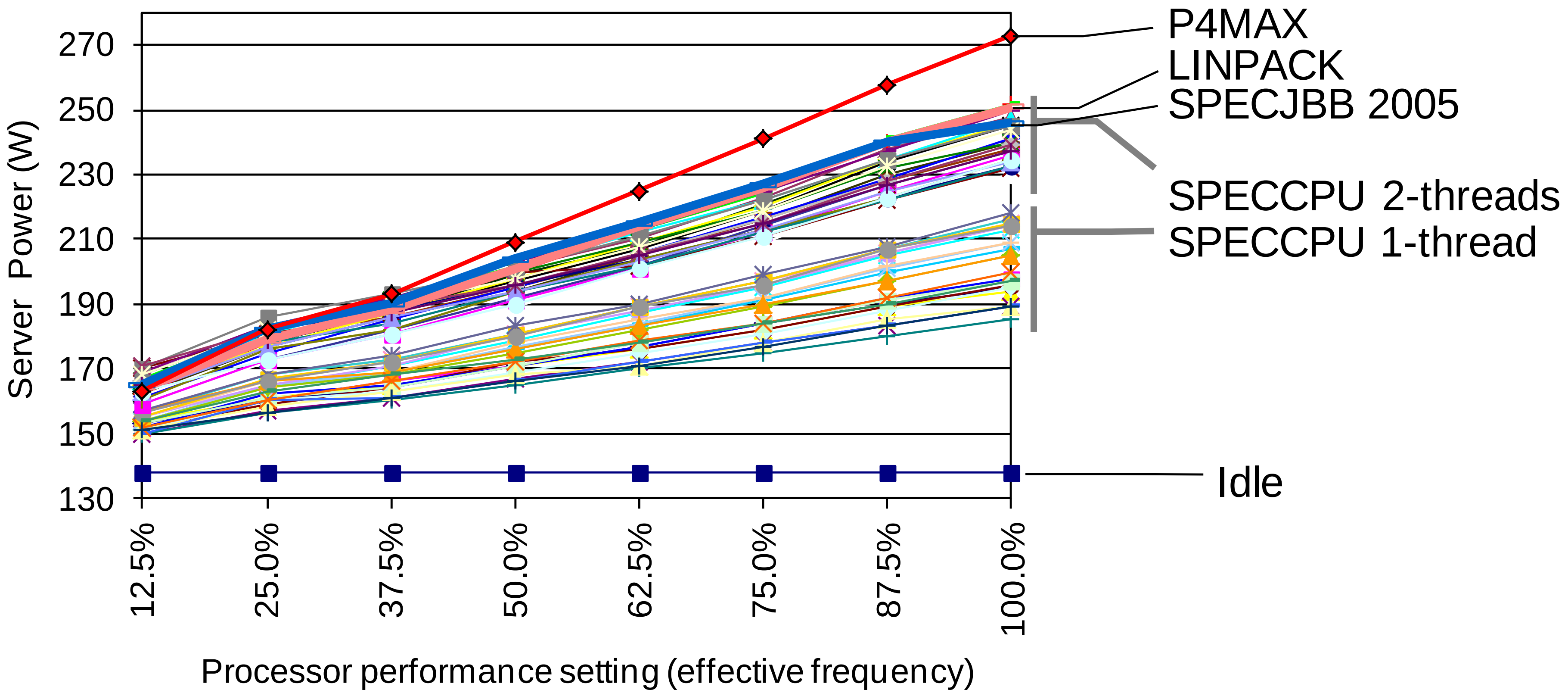
Ad-hoc
- Measures power and compares to power budget.
- +1/-1 adjustments to processor speed.



Proportional-integral-derivative (PID)
- Designed using control theory.
- Guaranteed controller performance.

Open loop design

- P4MAX workload used as basis for open-loop controller
- Leads to slowdown for all workloads, regardless of actual power consumption.
  – 250 W cap uses 75% performance setting.



Proportional controller design

- Time-domain model

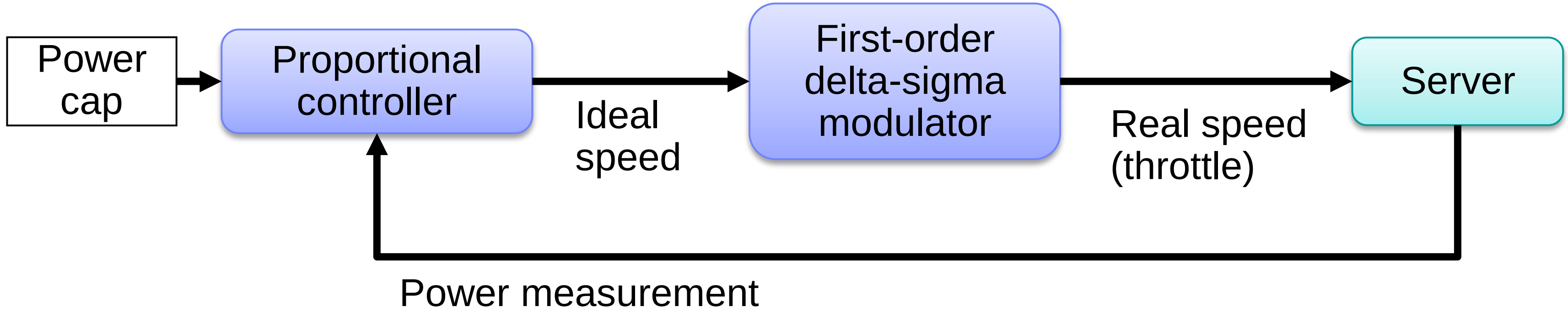
$$speed(t+1) = speed(t) + A * (Power_{cap} - Power_{measured}(t))$$

"A" parameter converts power difference to speed adjustment.

  - Selected based on average of slopes in prior chart.
  - Provably settles within 1 second.

- Control interval is 64 ms.
  - Measure power and select new throttle value.
  - Use delta-sigma modulation to achieve finer throttling resolution (units of 0.1%).

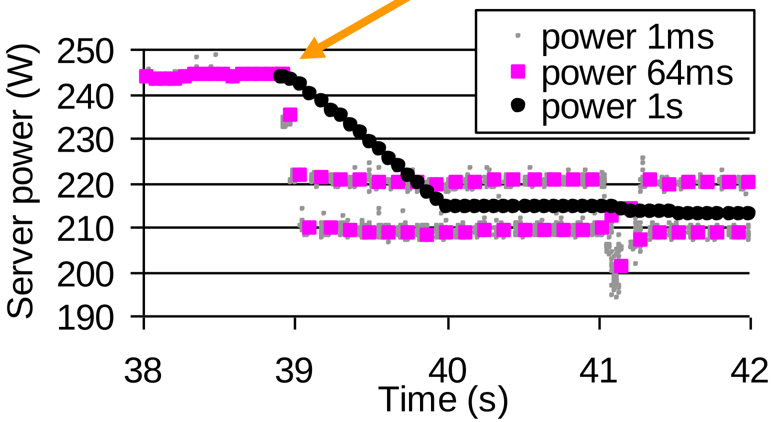
- System diagram

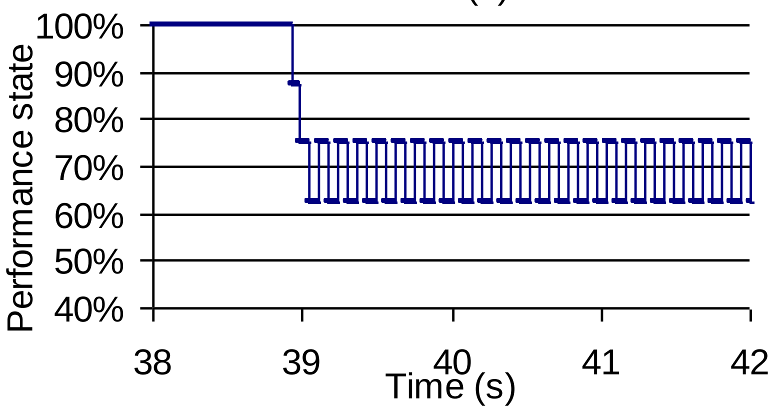


Why not use ad-hoc control?

Set point = 211.0 W

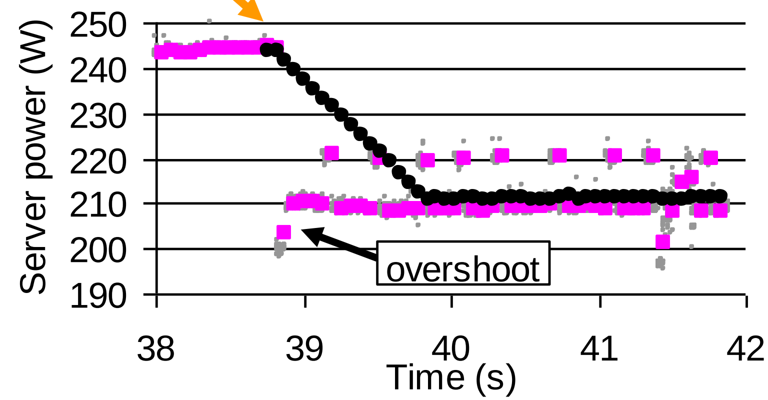
Ad-hoc

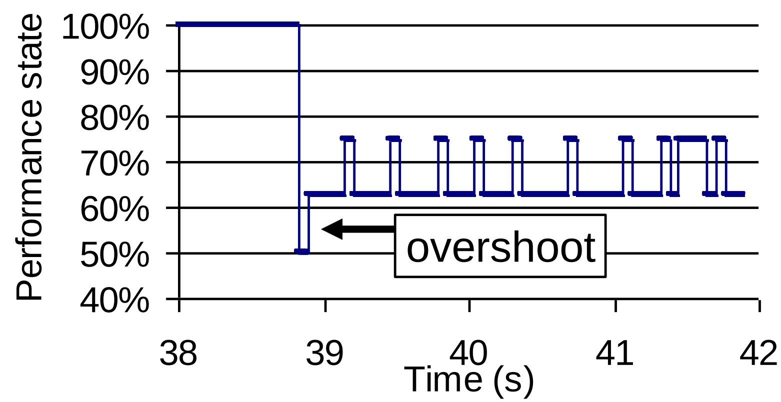




Settles to 216.0 W 5 W Violation
CPU speed: 68.8%

P Controller

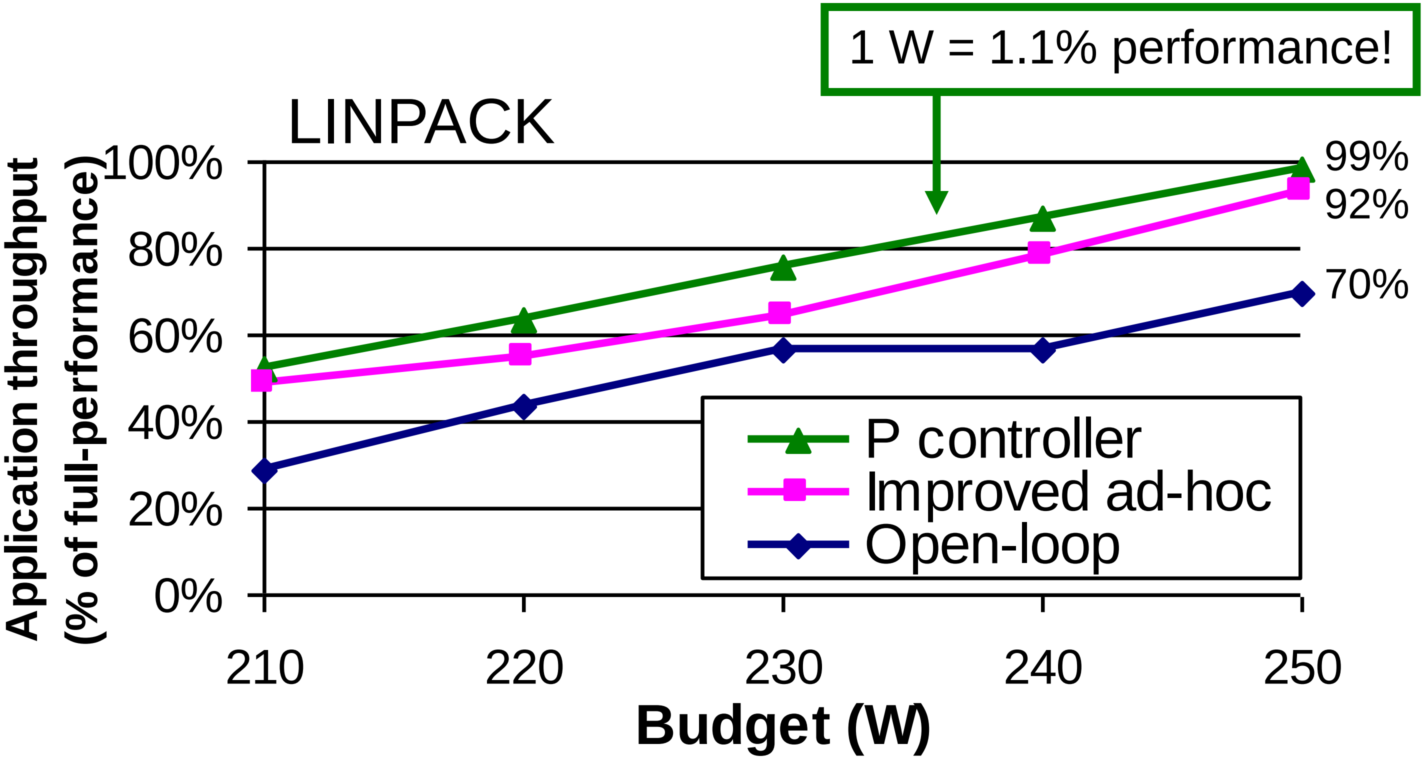




Settles to 211.0 W No violation
CPU speed: 65.8%

Comparison of controller types

- Improved ad-hoc controller use 6 W of guardband to avoid violations.
  - – Internally, 6 W subtracted from set point.
- P controller
  - – Up to 82% higher performance than open-loop controller.
  - – Up to 17% higher performance than ad-hoc controller.
  - – Zero steady state error.

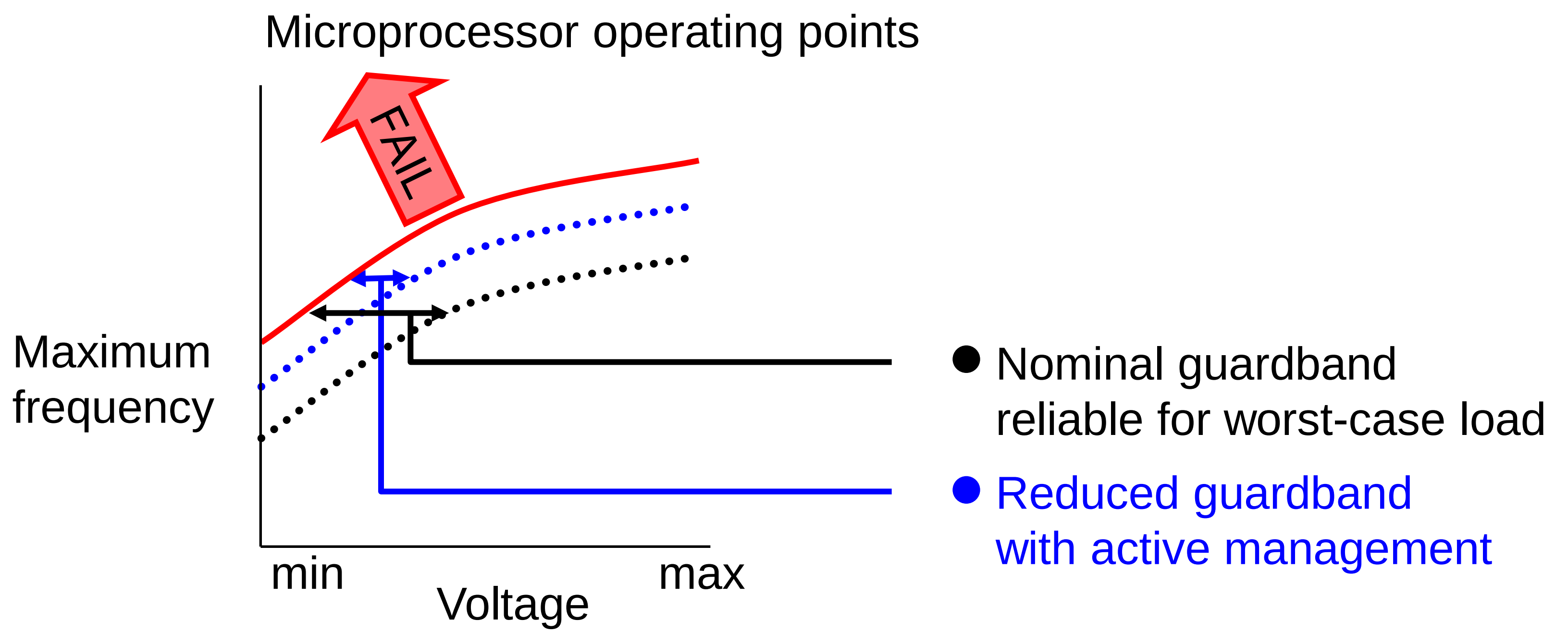


Power capping summary

- Power is a 1st class resource that can be managed.
  - Power consumption is no longer the accidental result component configuration, manufacturing variation, and workload.
- Better-than-worst-case design for power supply and cooling.
  - Size for important workloads, not power viruses.
  - Lower manufacturing cost.
- Power control is a fundamental mechanism managing a power-constrained datacenter.
  - Enables shifting power to critical workloads.
- Can be applied to server sub-systems (per-voltage regulator, per-core, etc.)

Problem #2: Excess guardband

- The voltage used on a microprocessor is conservative to provide a safe timing margin under worst-case conditions
  - workload-induced voltage droops (dI/dt or load line)
  - high temperature
- **Concern**: Energy-efficiency is reduced to guarantee reliability.
- **Opportunity**: Worst-case conditions rarely occur. Can actual timing margin be controlled?



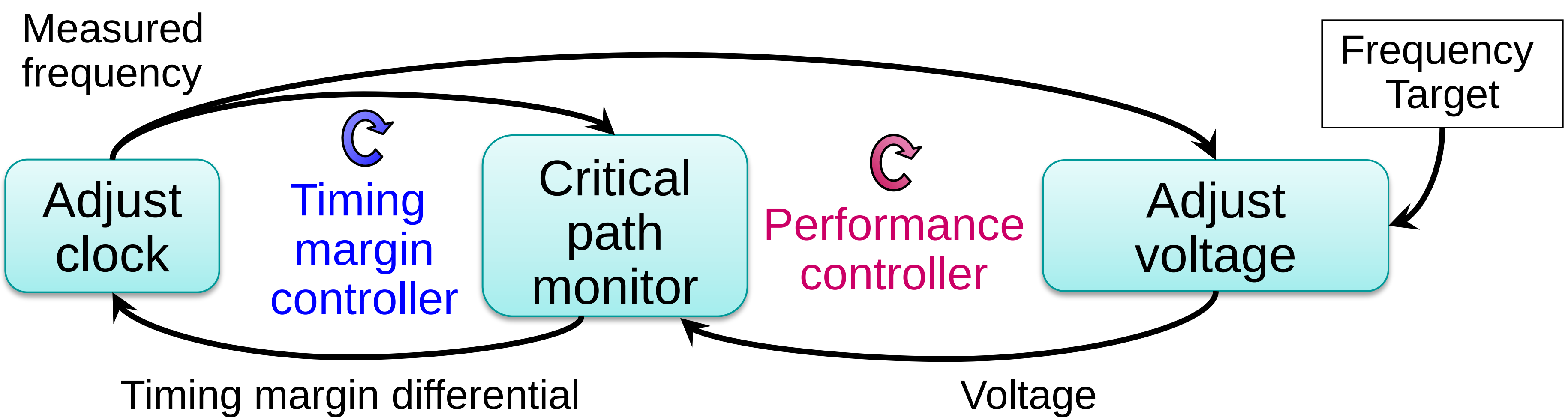
Active guardband management

- New capability to keep timing margin nearly constant
  – Convert excess timing margin into a voltage reduction
  – Reduce traditional voltage margin when conditions are not worst-case
  (Some voltage margin is retained for aging, calibration inaccuracy, etc.)

1. Measure excess operational margin with timing margin sensor
  – Difference from a calibrated reference point

2. Protect timing margin against voltage droop by adjusting frequency
  – Hardware-based **timing margin controller**

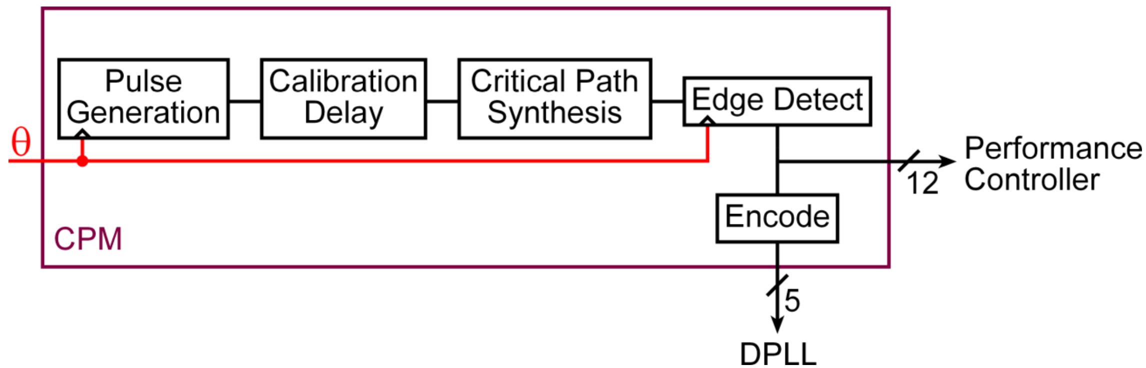
3. Save energy by converting excess timing margin into voltage reduction
  – Software-based **performance controller**



Measure timing margin

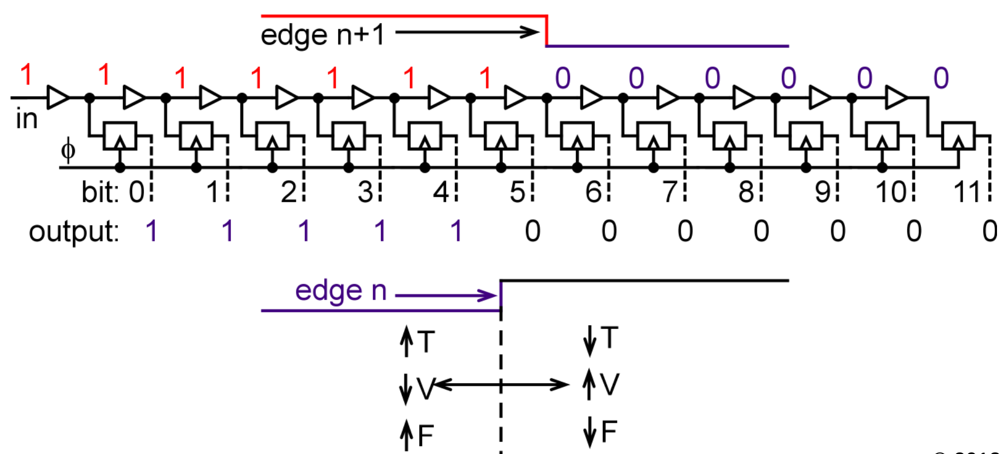
- Use **Critical Path Monitor** (CPM) circuit. Mimics behavior of real critical path.
- Each cycle: generate pulse, traverse synthesized critical path and calibrated delay, capture in edge detector

Critical Path Monitor



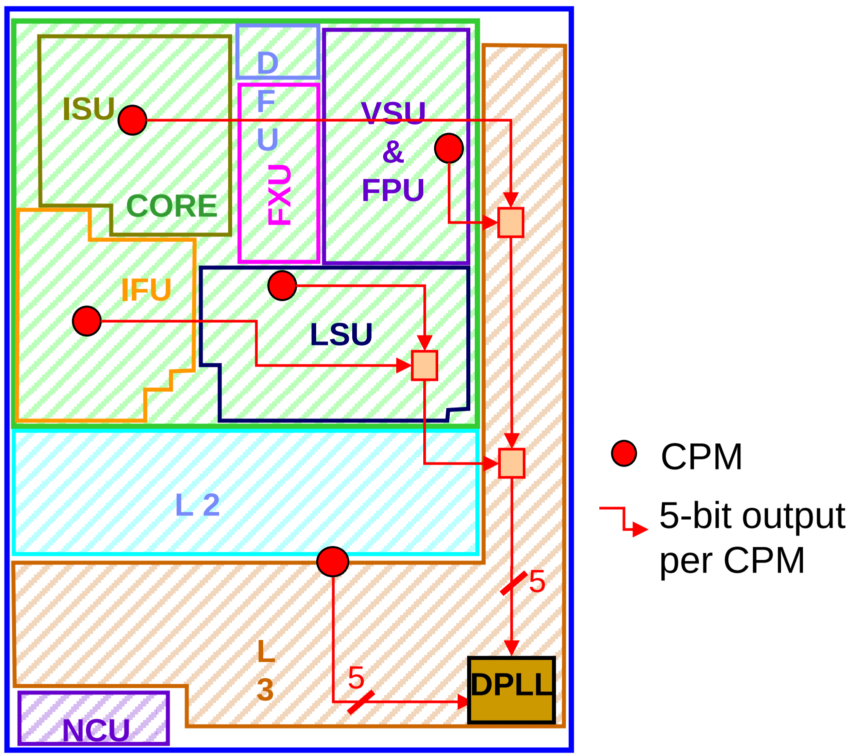
- Edge detector 12-bit output: (bit 0 = less margin, bit 11 = more margin)

Edge Detector



Critical path monitor

- 5 Critical Path Monitors per core in POWER7 (8 core chip)
- Middle bits of edge detector are forwarded to DPLL



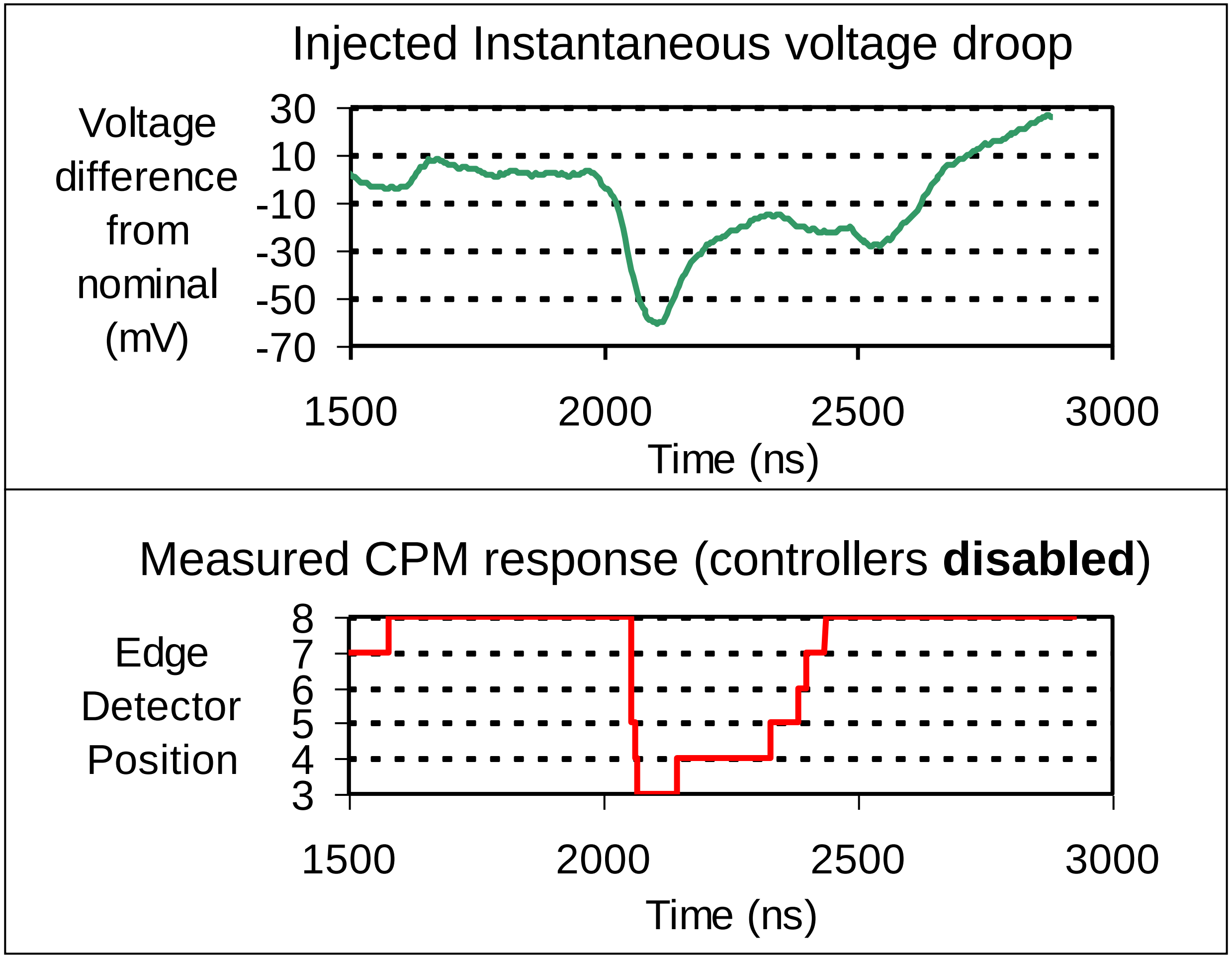
POWER7 core chiplet

CPM output

“11111” = large margin
“11110” = some margin
“11100” = ideal margin
“11000” = margin too small
“10000” = not enough margin

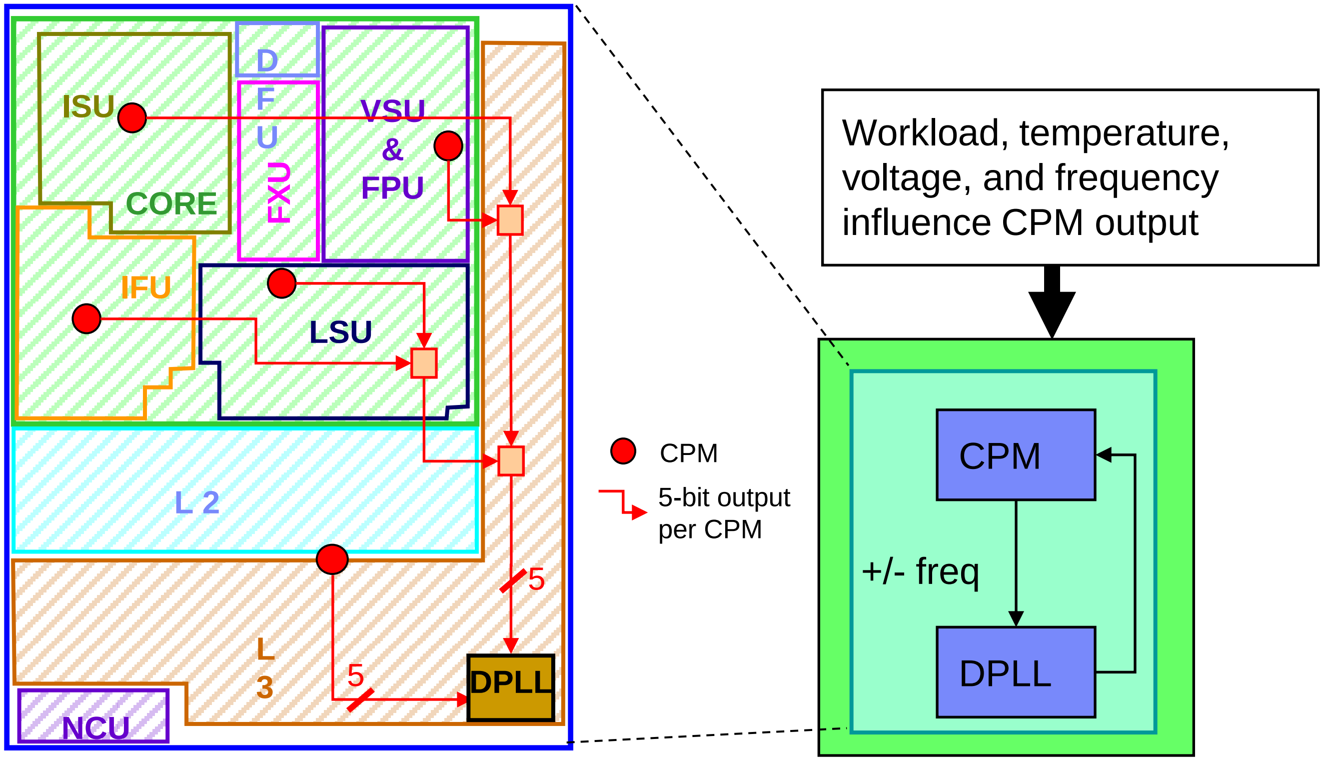
Example of critical path monitor output

- Inject 60 mV droop into Power 755 Express Server (with no load-line)
  – Instruction fetch throttling
- Critical path sensor follows on-chip voltage reduction



Protect timing margin

- **Timing margin controller** responds to changing operating conditions by adjusting frequency to maintain timing margin target.
  - Implemented in hardware of POWER7.
  - Can reduce frequency by -7% in about 5 ns to handle fast voltage droop.



POWER7 core chiplet

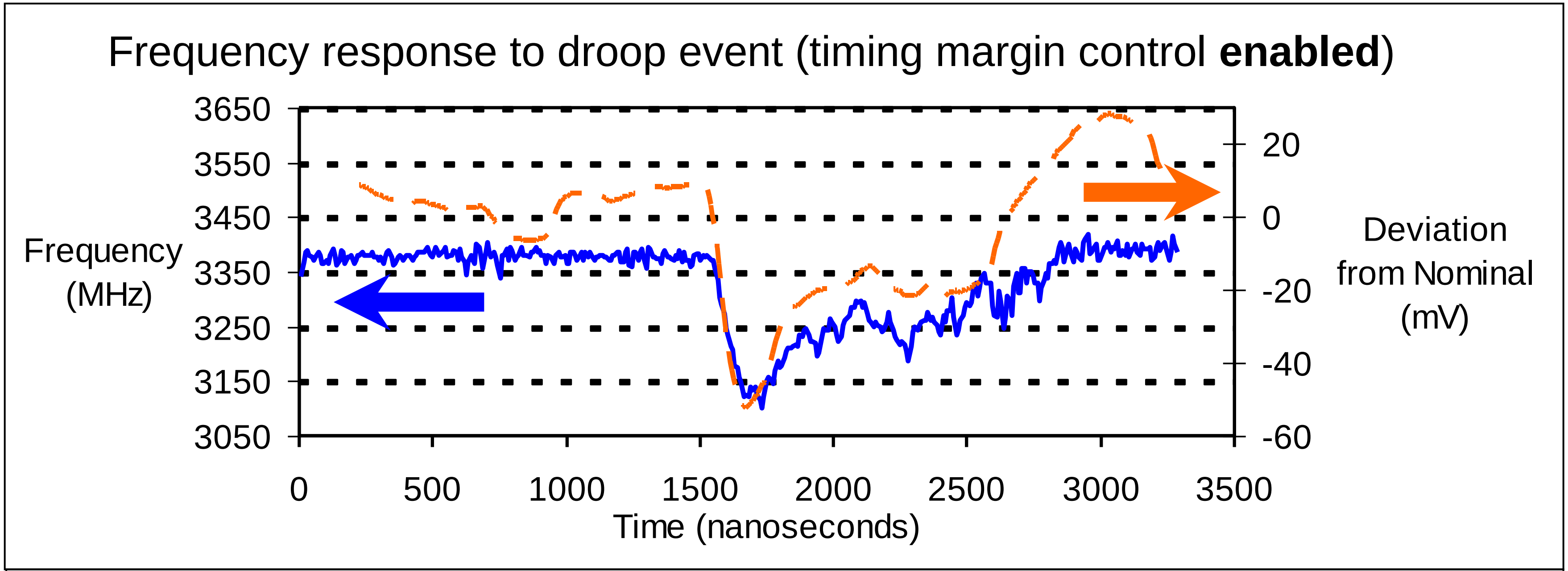
POWER7

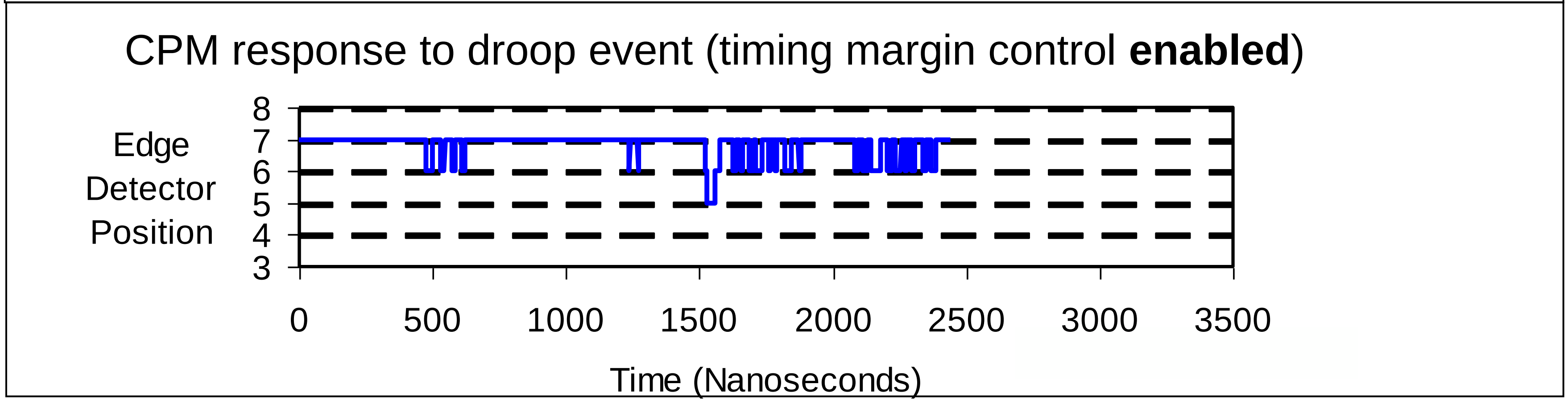
Calibration of critical path

- Teach the chip the desired timing margin to use during field operation
- Done once during manufacturing of chip
- Run chip at desired timing margin
  - Set voltage, frequency, and temperature
  - Run stressful workload
- Find delay setting that places timing edge on position 6 in edge detector
  - Position 6 is the setpoint for the timing margin controller

Timing margin controller response time

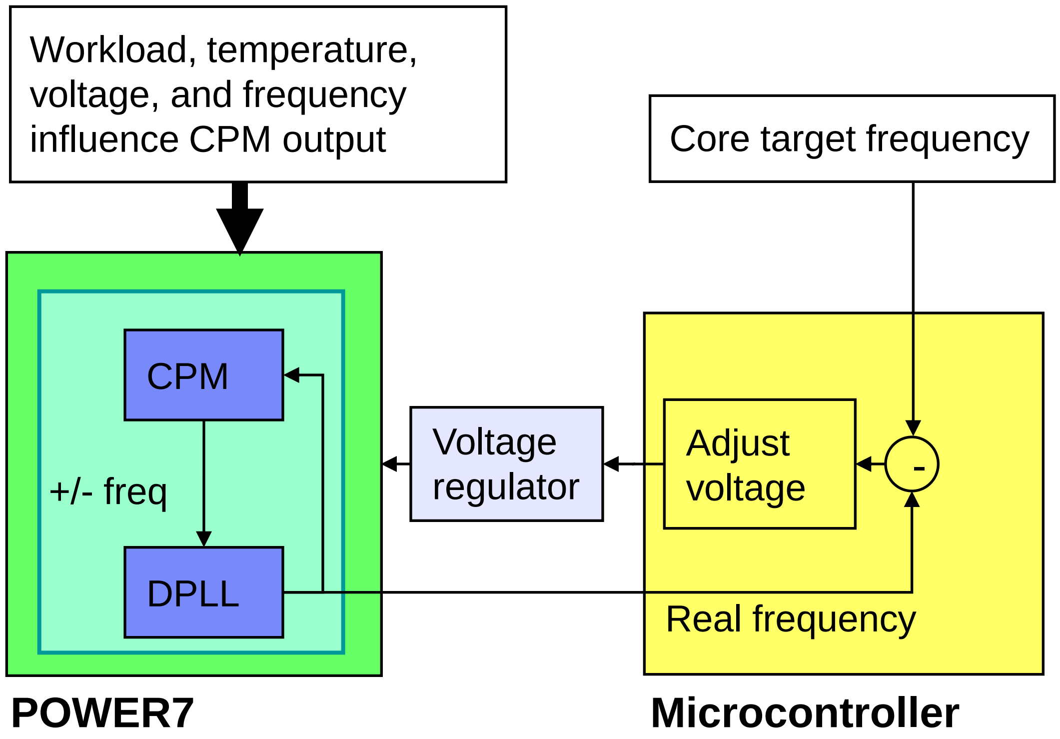
- Quick enough to follow voltage droops



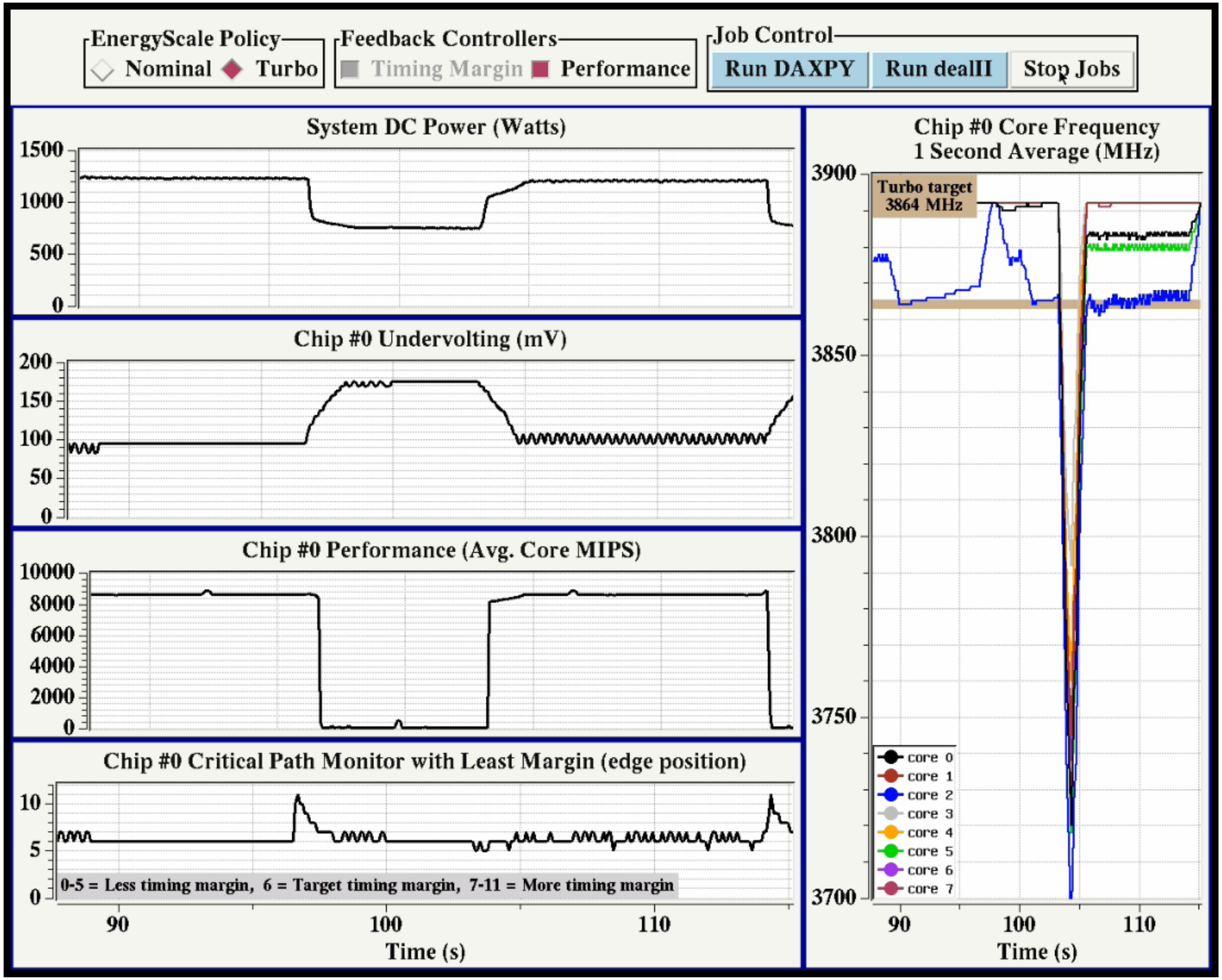


Save energy

- **Performance controller** adjusts voltage to meet desired clock frequency target.
  - Implemented in firmware of on-board microcontroller
  - Frequency is capped at target + 28 MHz (clock resolution)
    - Prevent energy waste
    - Allow for detection of excess timing margin for voltage reduction

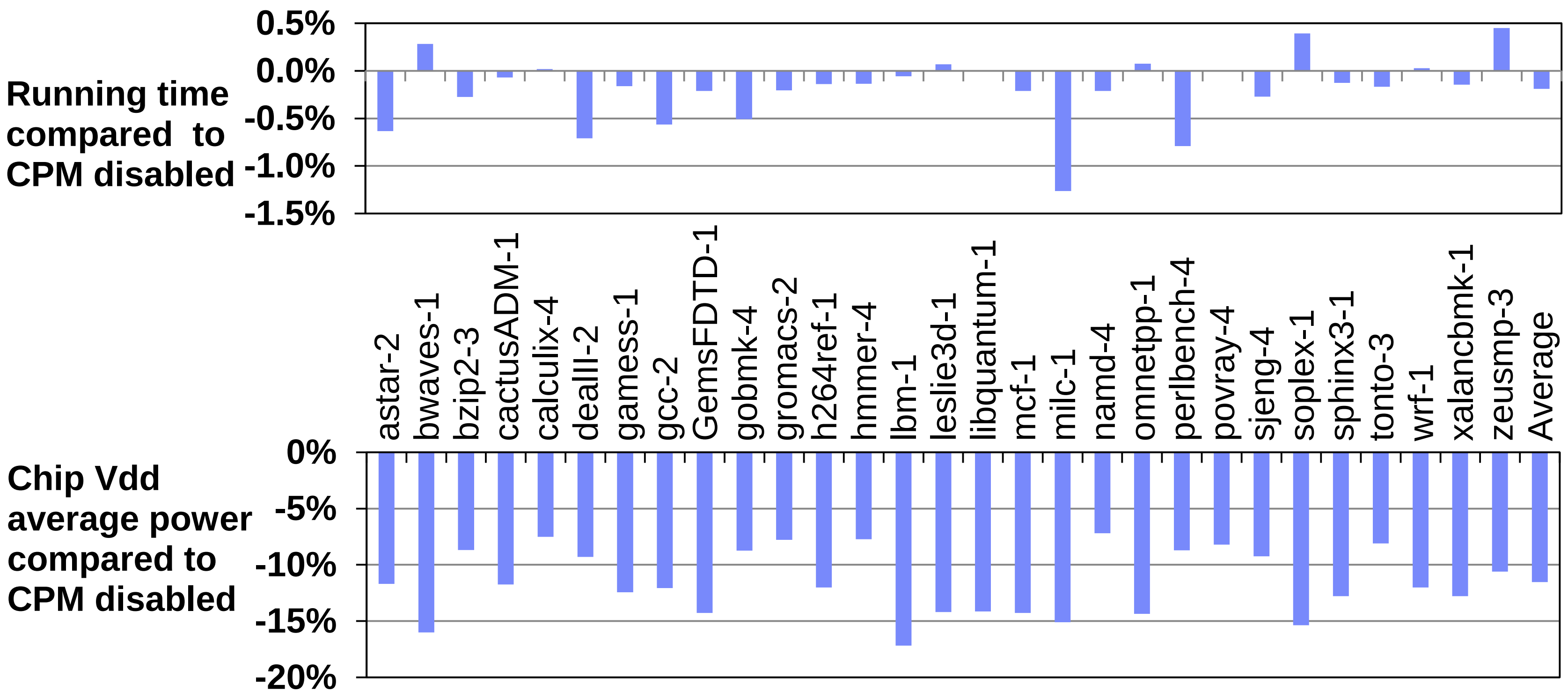


Demonstration



POWER7+ results

- IBM Power 780 Server
  - 4-socket 4.1GHz (32 cores)
  - 128 GB
  - 30 C ambient
- Vdd power reduced by 11%
- Reduced fan power due to lower temperature processor
- Negligible performance impact



Summary of guardband management

- Demonstration of a new capability to keep timing margin nearly constant
- Architecture combines two feedback controllers
  - Hardware-based timing margin controller (safety)
  - Software-based performance controller (undervolting)
- Used in production POWER7+ servers
  - Reduces chip Vdd power by 11% for SPEC CPU2006
  - Improves performance during power capping
- IP portfolio for licensing

Bibliography

- Michael Floyd, Bishop Brock, Malcolm Allen-Ware, Karthick Rajamani, Bishop Brock, Charles Lefurgy, Alan J. Drake, Lorena Pesantez, Tilman Gloekler, Jose A. Tierno, Pradip Bose, and Alper Buyuktosunoglu, "Introducing the Adaptive Energy Management Features of the POWER7 Chip ", IEEE Micro, vol. 31, no. 2, March/April, 2011.
- Charles Lefurgy, Xiaorui Wang, and Malcolm Ware, "Server-level Power Control ", *4th IEEE Conference on Autonomic Computing (ICAC'07),* 2007.
- Wei Huang, Charles Lefurgy, William Kuk, Alper Buyuktosunoglu, Michael Floyd, Karthick Rajamani, Malcolm Allen-Ware, and Bishop Brock, "Accurate Fine-Grained Processor Power Proxies", Proceedings of the 45th Annual International Symposium on Microarchitecture, December 2012.
- Charles Lefurgy, Alan Drake, Michael Floyd, Malcolm Allen-Ware, Bishop Brock, Jose Tierno, and John Carter, "Active Management of Timing Guardband to Save Energy in POWER7", Proceedings of the 44th Annual International Symposium on Microarchitecture, December 2011.

Above papers are available at
http://researcher.watson.ibm.com/researcher/view_person_subpage.php?id=2758