Issue Logic and Power/Performance Tradeoffs

Edwin Olson

Andrew Menard

December 5, 2000

The need for low-power architectures

- Low performance - PIMs
- High performance – video decoding/MP3 playback
- *And increasingly*, **both.**
  - How do you design an architecture that can do both?

A couple alternatives

- High performance processor that can be lobotomized
  - Modify Issue Logic
  - Change structure sizes
- Two separate cores
  - A high performance/high-power core
  - A low performance/low-power core

Other power throttling mechanisms

- Voltage scaling
  - Huge power savings
  - There's a limit & high performance designs are pushing towards low voltage– which doesn't leave much room for throttling.
- Burn & Coast
  - Compute at full speed, and then go into a sleep mode.
  - Simple linear power/performance throttling.

Methodology

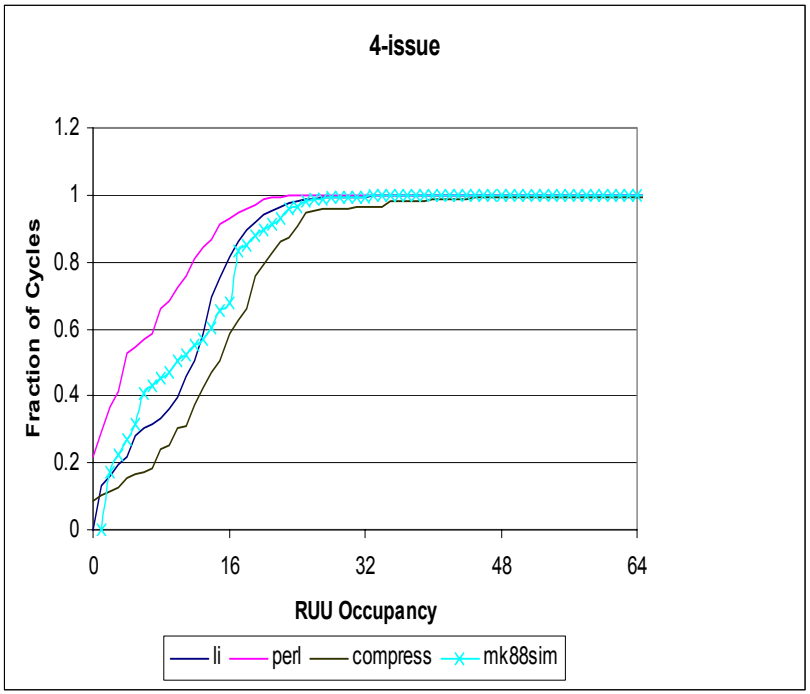
- SimpleScalar/Wattch
  - Widely used but little/no verification. Several power models available, but very large margins of error.
  - Still, the size of structures *is* correlated to power consumption.
- Industry survey
  - Look at real-world processors with the range of characteristics of interest.
- SpecInt95
  - Substantially reduced input sets to make simulation feasible.

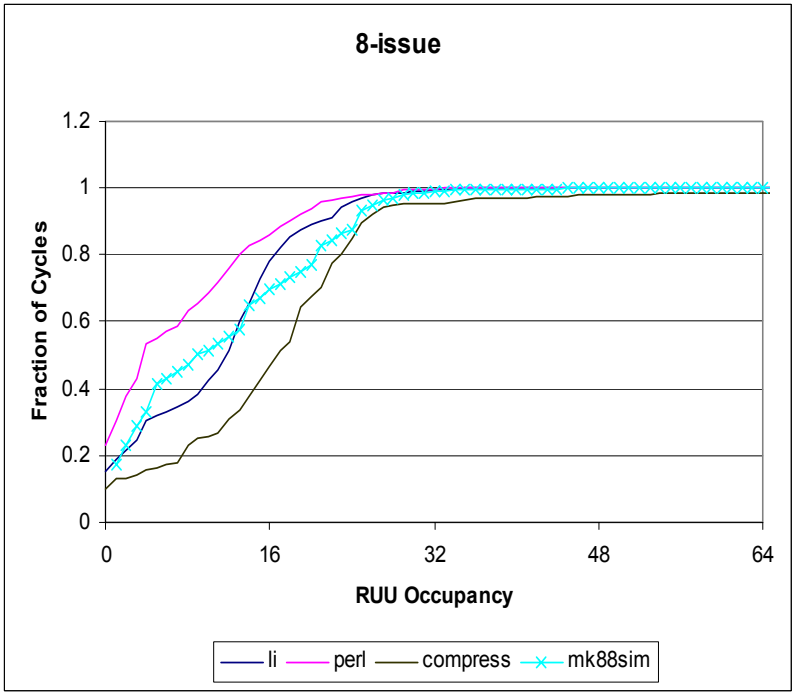
Issue Window Scaling

- Popular idea- it's a highly active chip structure. Window responsible for 20% of non-clock power (Alpha 21264 & Wattch agree)
- Does it work?
  - Let's look at RUU usage
    - What's an upper bound on the useful size?
    - How do smaller sizes impact performance and power?

RUU size upper bounds

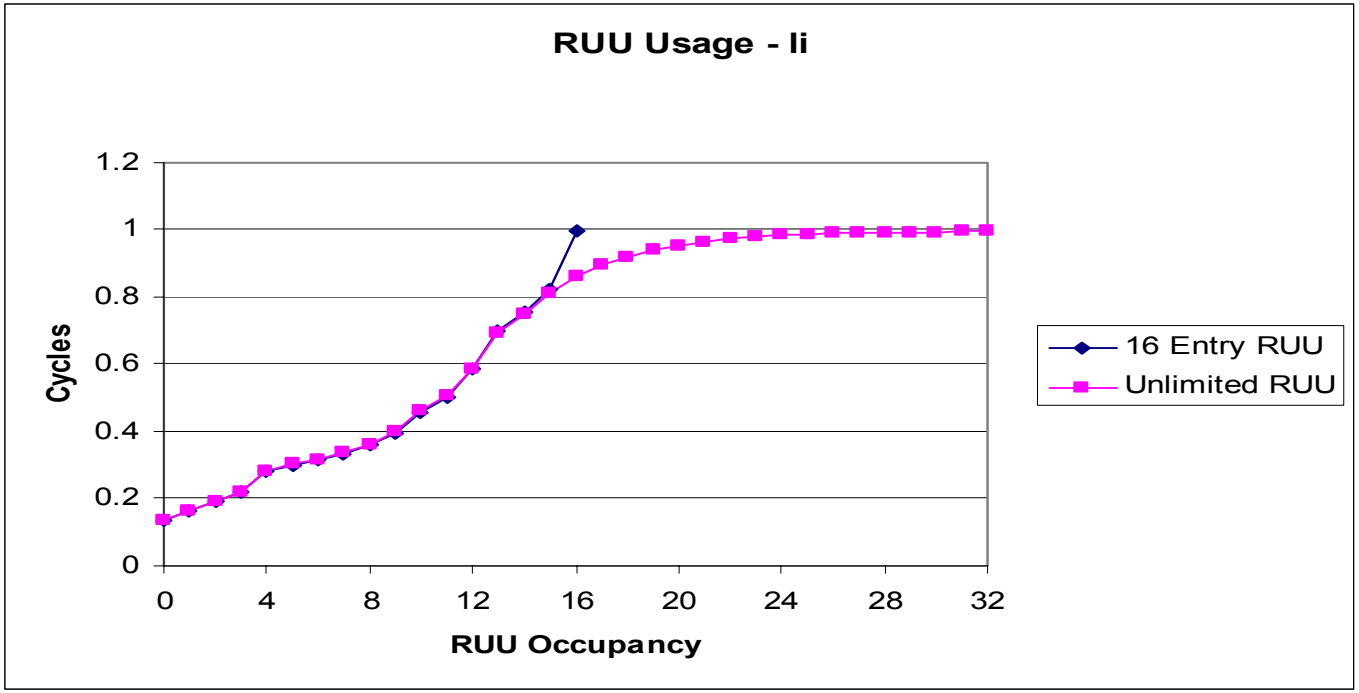
- Modified SimpleScalar, let RUU be arbitrarily big.



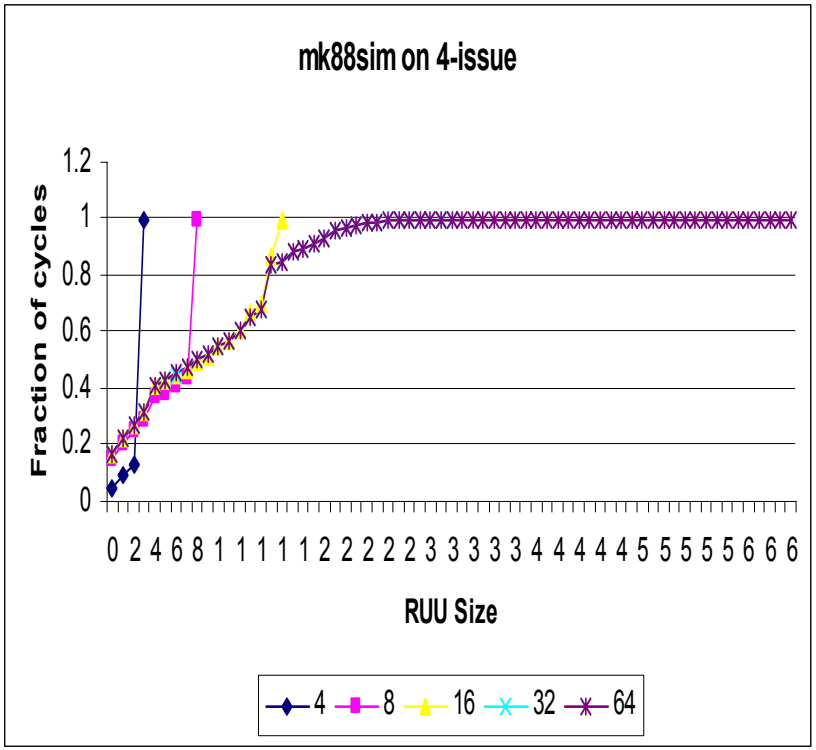


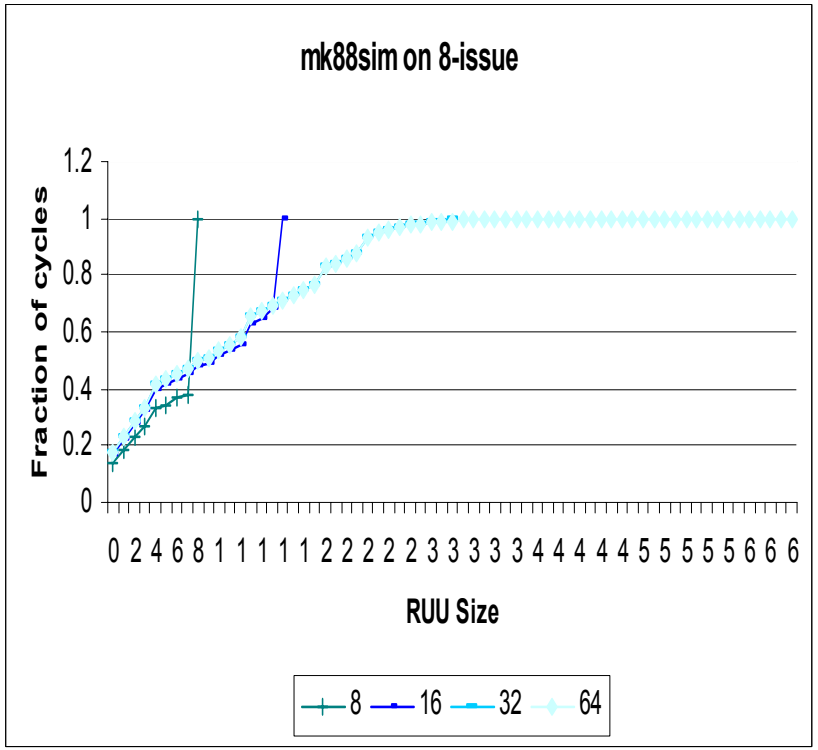
Effect of bounded RUU size

- The RUU's occupancy "saturates" as one would expect.



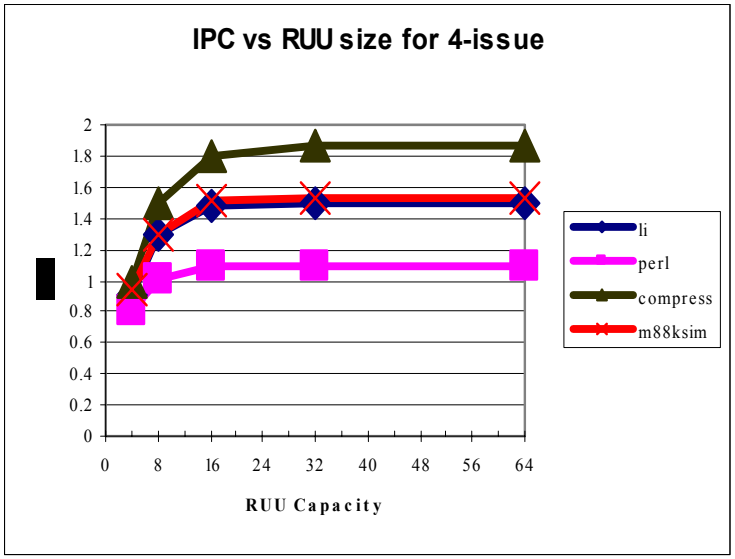
Effect of Bounded RUU Size

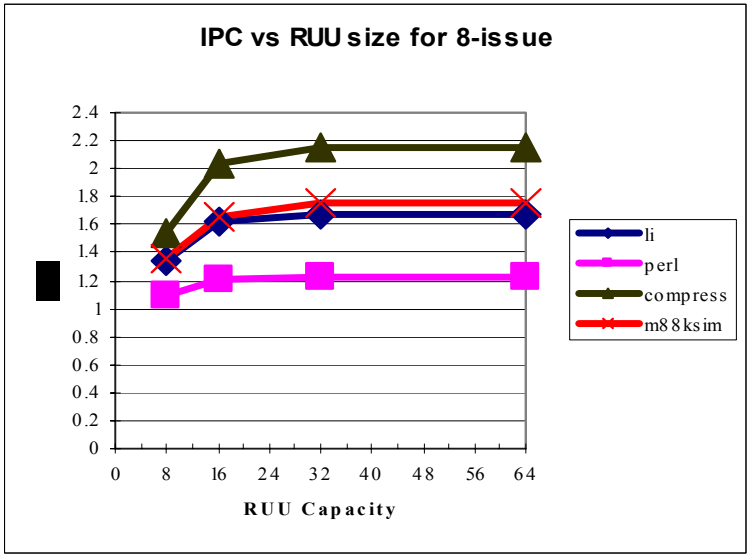




Bounded RUU Impact on Performance

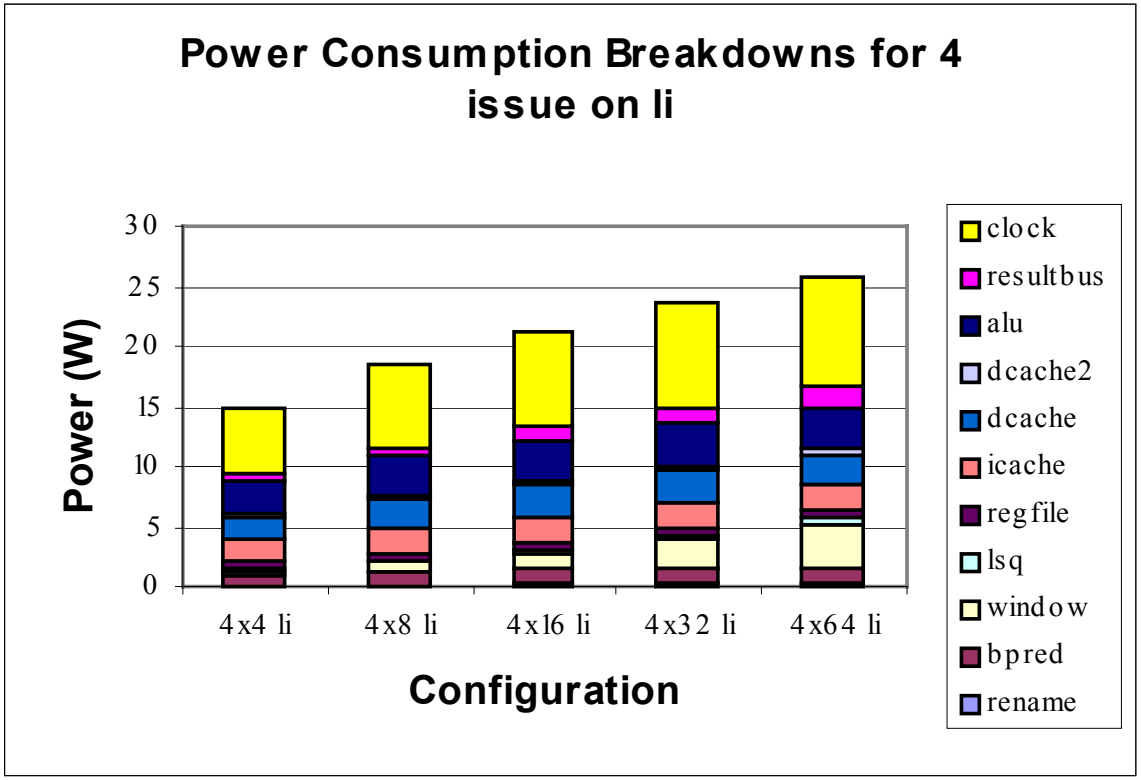
- Performance rapidly approaches maximum.
- 8-issue needs a slightly larger RUU, as expected.





Bounded RUU impact on Power

- Power consumption increased in RUU as size increases



Power/Performance

- There's a minimum! And it's pretty much where maximum performance is. Hmmm.

Structure	8x8	8x16	8x32	8x64
Energy/Inst (li)	13.8	**12.5**	13.4	14.9
Energy/Inst (perl)	15.1	**14.7**	15.8	17.6
Energy/inst (compress)	12.4	**11.4**	11.9	13.3
Energy/inst (m88ksim)	13.0	**12.1**	12.9	14.4

Analysis

- Some groups have advocated a variable 16-32 capacity RUU. Even if scaling is perfect, there's little to be gained.
- A power-conscious architect is likely to be cornered into just one reasonable RUU size.

Adding a separate core

- If we can't lobotomize, perhaps we can add a completely separate CPU.
- Sounds like a good idea
  - Intuition: a simple in-order processor should have lower energy/instruction than a complex out-of-order one.
  - Small area overhead, around 1mm^2.
- Opportunity for more energy savings
  - Smaller register file
  - No issue window
  - Separate low-power caches (though this increases area)

Methodology

- SimpleScalar/Wattch is all but useless
  - Availability of only one parameterizable power model (Wattch) and we don't know what trade-offs the designer made.
  - Wattch doesn't support sim-inorder
  - E.g., Cacti cache model uses 10x greater energy than Krste.
- Industry Survey

PowerPC Statistics

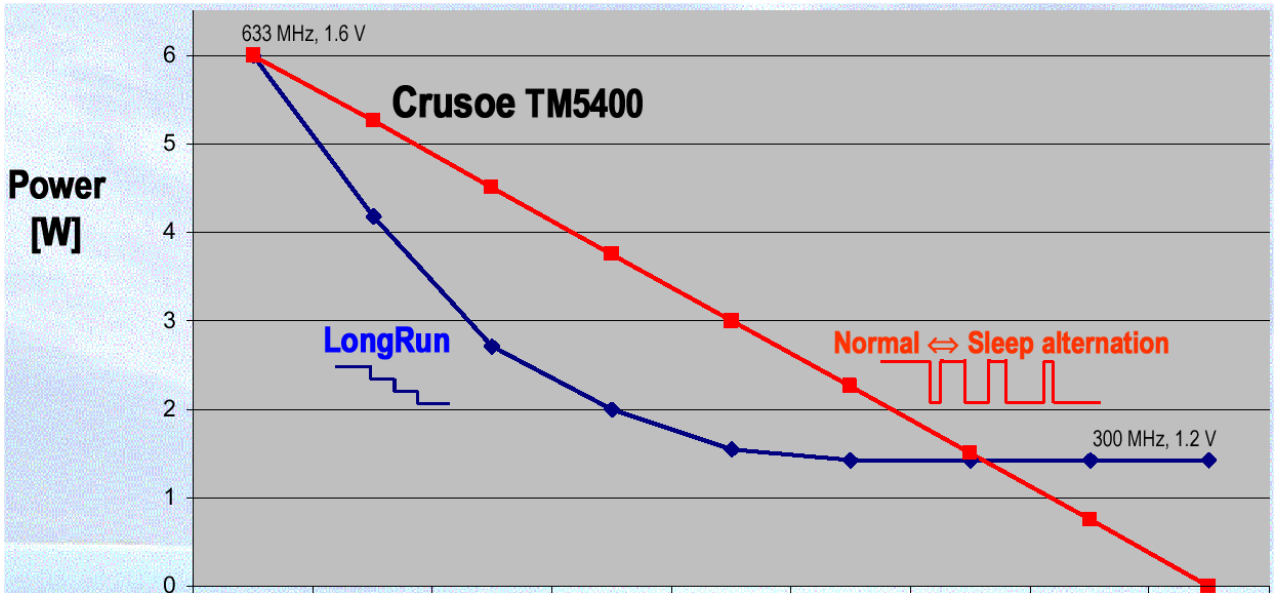
- PPC440 is 2-issue, out of order
- PPC405 is single issue, in-order
- Both use same technology

- The 440 is twice as fast, but uses only 1.66 times the power!

AM5x86 vs. K6

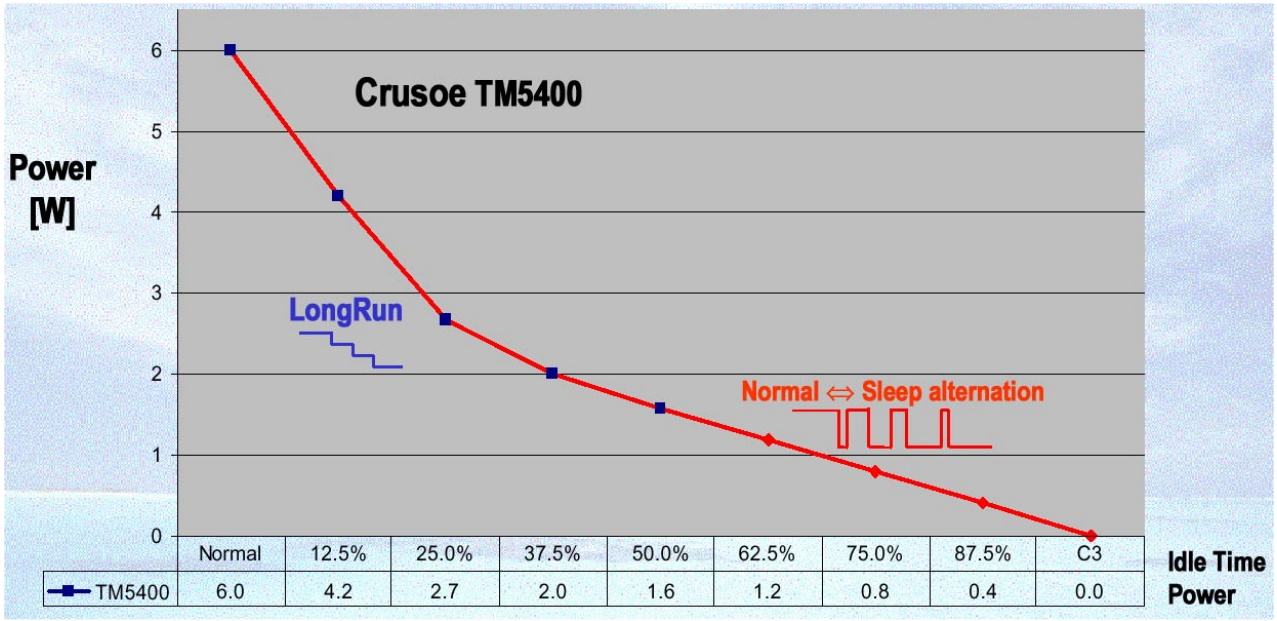
- 5x86 is in-order
- K6 is out-of-order, 6 issue, 24 entry window
- K6 has slightly better power/performance
  - But it's on a newer process (0.25um rather than 0.35)

Crusoe's Voltage Scaling & Coast and Burn



	Normal	12.5%	25.0%	37.5%	50.0%	62.5%	75.0%	87.5%	C3
LongRun	6.0	4.2	2.7	2.0	1.6	1.4	1.4	1.4	1.4
C3	6.0	5.3	4.5	3.8	3.0	2.3	1.5	0.8	0.0

Crusoe's Voltage Scaling & Coast and Burn



	Normal	12.5%	25.0%	37.5%	50.0%	62.5%	75.0%	87.5%	C3
TM5400	6.0	4.2	2.7	2.0	1.6	1.2	0.8	0.4	0.0

Big Proviso

- CPUs available today, even the “low power” ones, are still after speed.
  - Low power IA32 is just a slower, high-power IA32.
- If you designed your simple core for super-low power (without very little regard for speed), how might this change?

Conclusion

- Smaller issue windows are not a win on power; they lower the amount of ILP found by too much.
- Multiple cores are not a win on power; the faster core tends to be more energy efficient.