

Revisiting the Future ☺

Erik Hagersten
Uppsala University
Sweden

DARK2 in a nutshell

1. Memory Systems (caches, VM, DRAM, microbenchmarks, ...)
2. Multiprocessors (TLP, coherence, interconnects, scalability, clusters, ...)
3. CPUs (pipelines, ILP, scheduling, Superscalars, VLIWs, embedded, ...)
4. Future: (physical limitations, TLP+ILP in the CPU,...)

How do we get good performance?

- Creating and exploring:

- 1) Locality

- a) Spatial locality
 - b) Temporal locality
 - c) Geographical locality

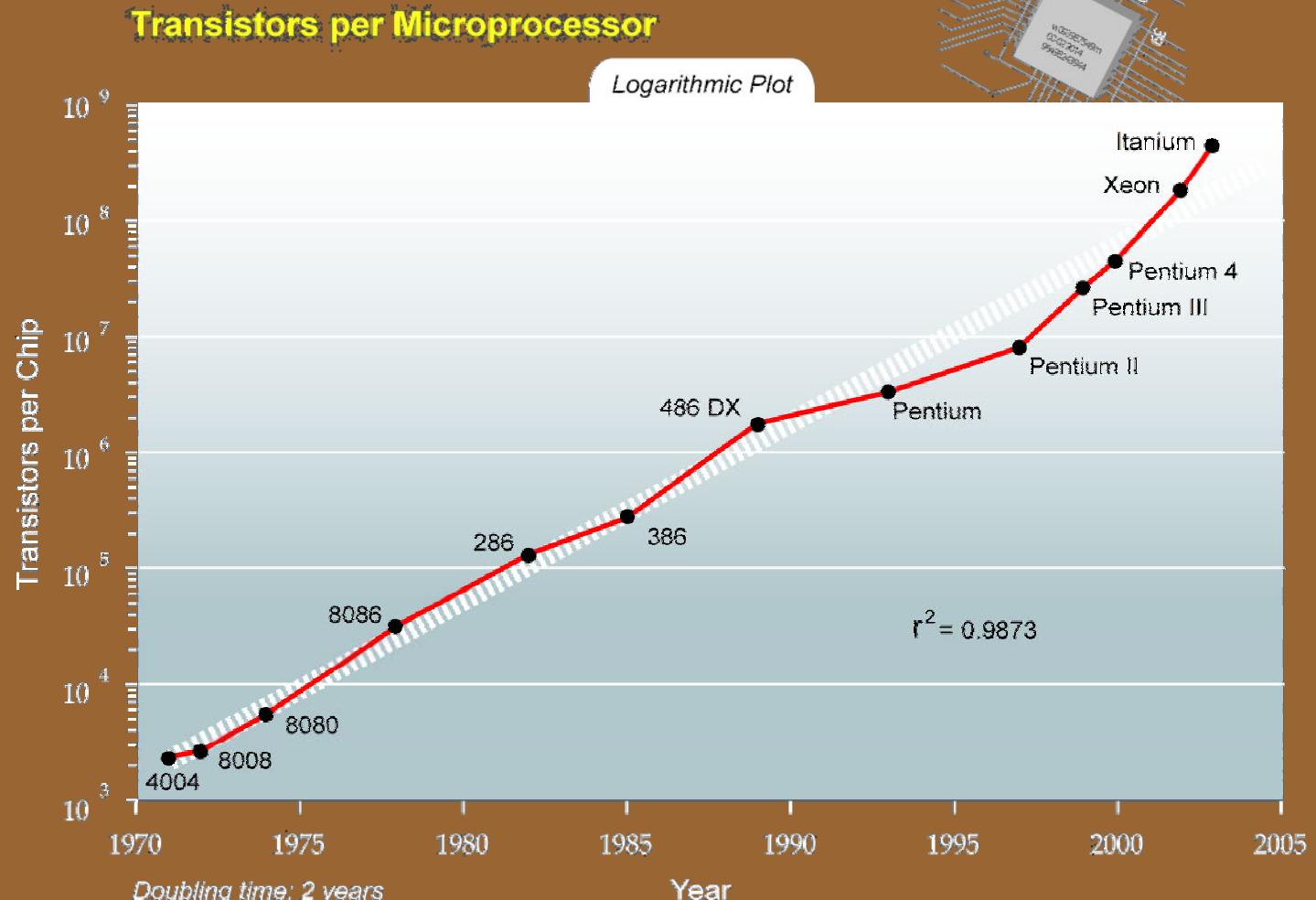
- 2) Parallelism

- a) Instruction level (ILP)
 - b) Thread level (TLP)
 - c) Memory level (MLP)



UPPSALA
UNIVERSITET

Ray Kurzweil pictures
www.KurzweilAI.net/pps/WorldHealthCongress/



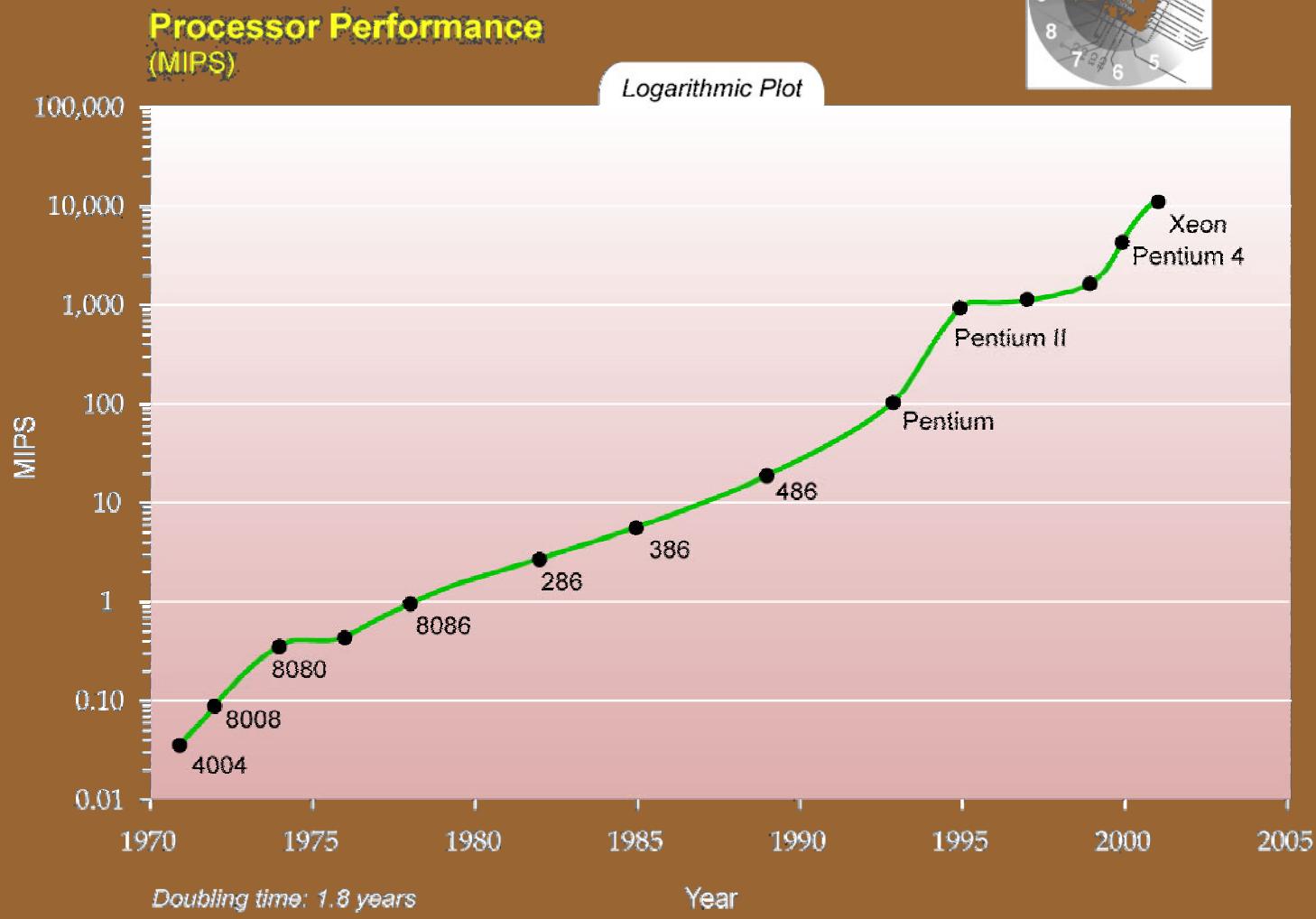
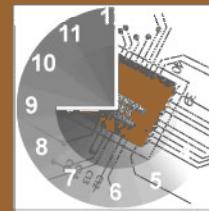
DARK
2009



UPPSALA
UNIVERSITET

DARK
2009

Ray Kurzweil pictures
www.KurzweilAI.net/pps/WorldHealthCongress/

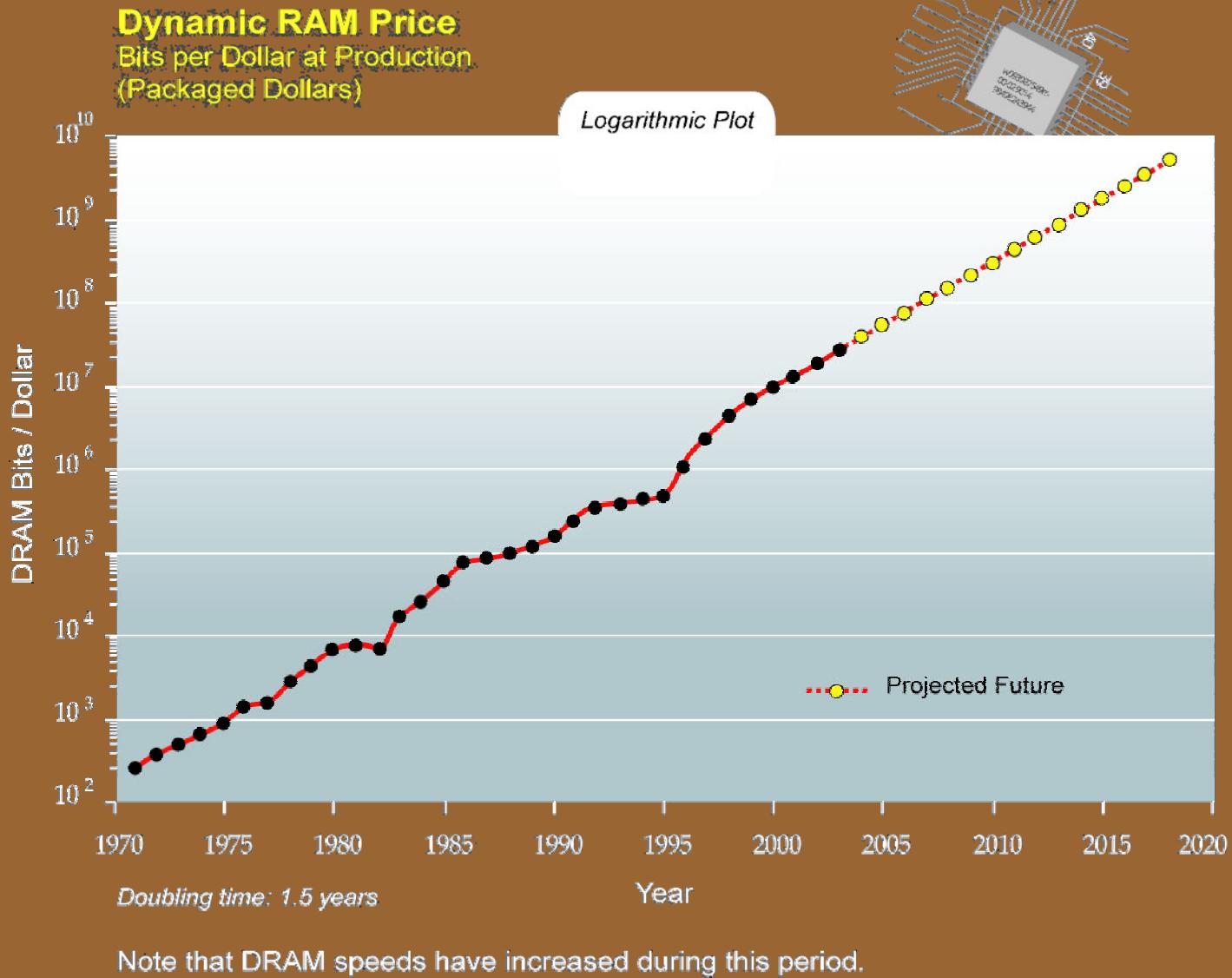




UPPSALA
UNIVERSITET

DARK
2009

Ray Kurzweil pictures
www.KurzweilAI.net/pps/WorldHealthCongress/



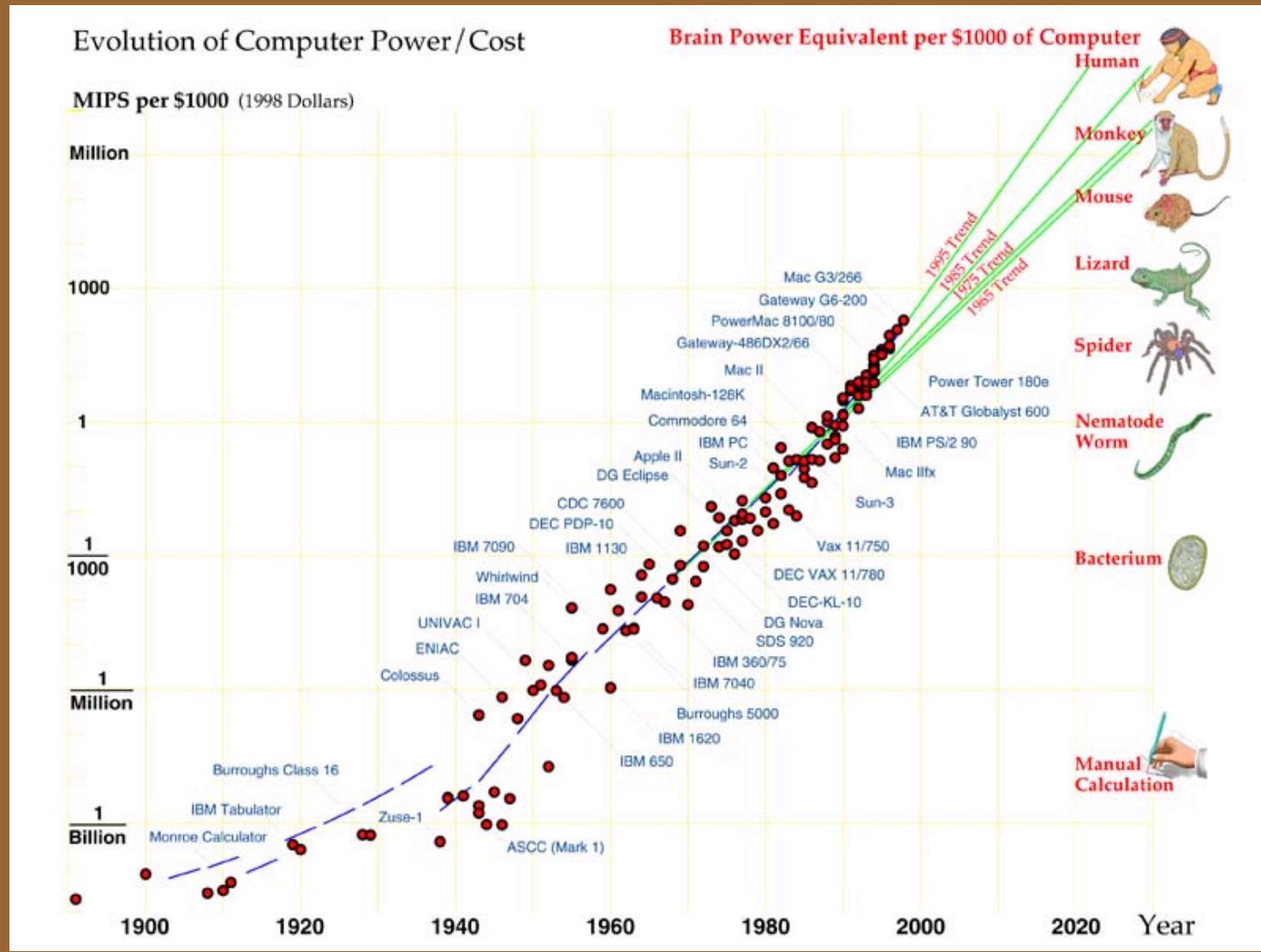


UPPSALA
UNIVERSITET

DARK
2009

Ray Kurzweil pictures

www.KurzweilAI.net/pps/WorldHealthCongress/

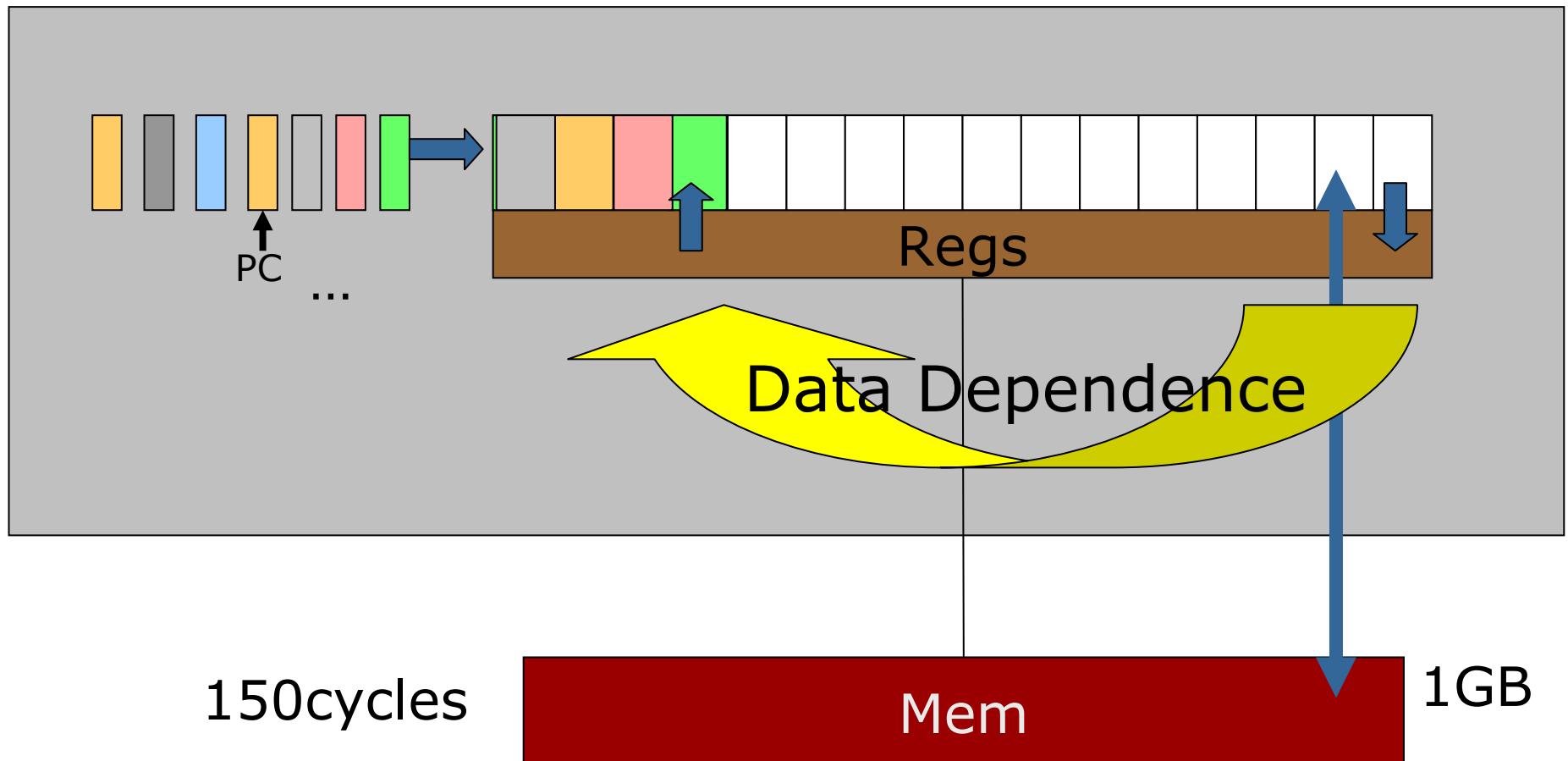


Doubling (or Halving) times

■ Dynamic RAM Memory (bits per dollar)	1.5 years
■ Average Transistor Price	1.6 years
■ Microprocessor Cost per Transistor Cycle	1.1 years
■ Total Bits Shipped	1.1 years
■ Processor Performance in MIPS	1.8 years
■ Transistors in Intel Microprocessors	2.0 years
■ Microprocessor Clock Speed	2.7 years

Old Trend1: Deeper pipelines

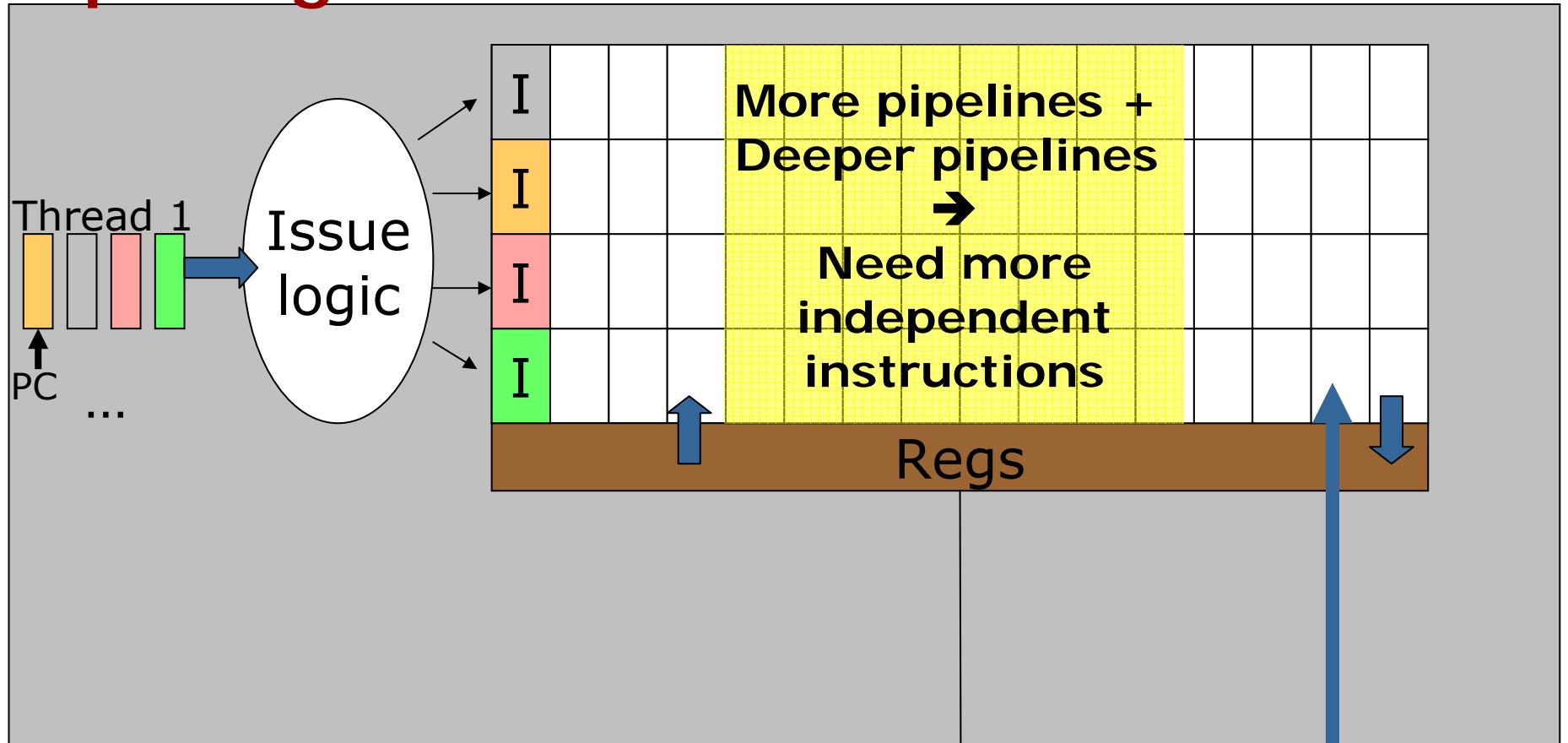
Exploring ILP (instruction-level parallelism)



DARK
2009

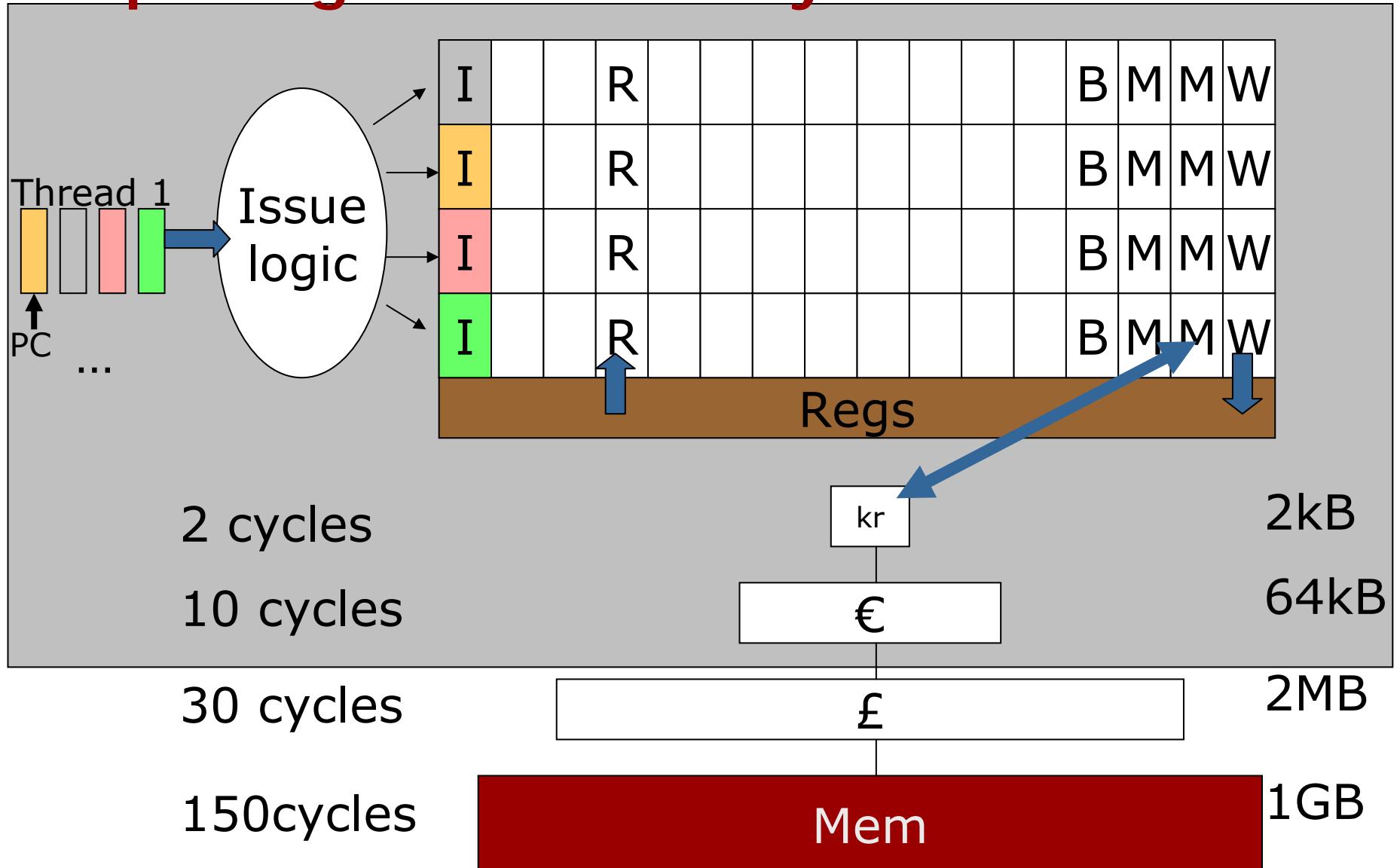


Old Trend2: Wider pipelines Exploring more ILP



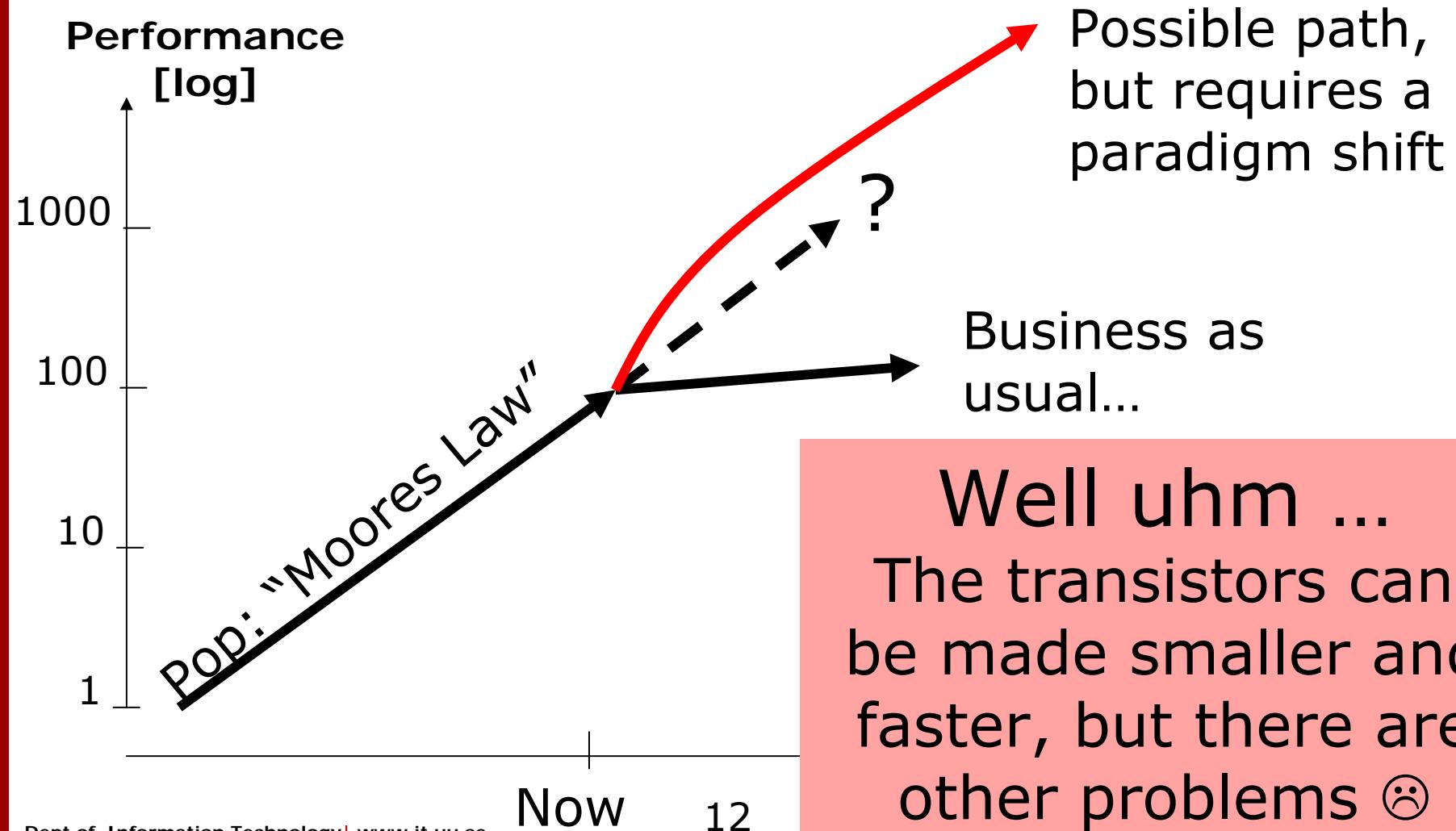


Old Trend3: Deeper memory hierarchy Exploring access locality



Are we hitting the wall now?

Pop: Can the transistors be made even smaller and faster?



Microprocessors today: Whatever it takes to run one program fast.

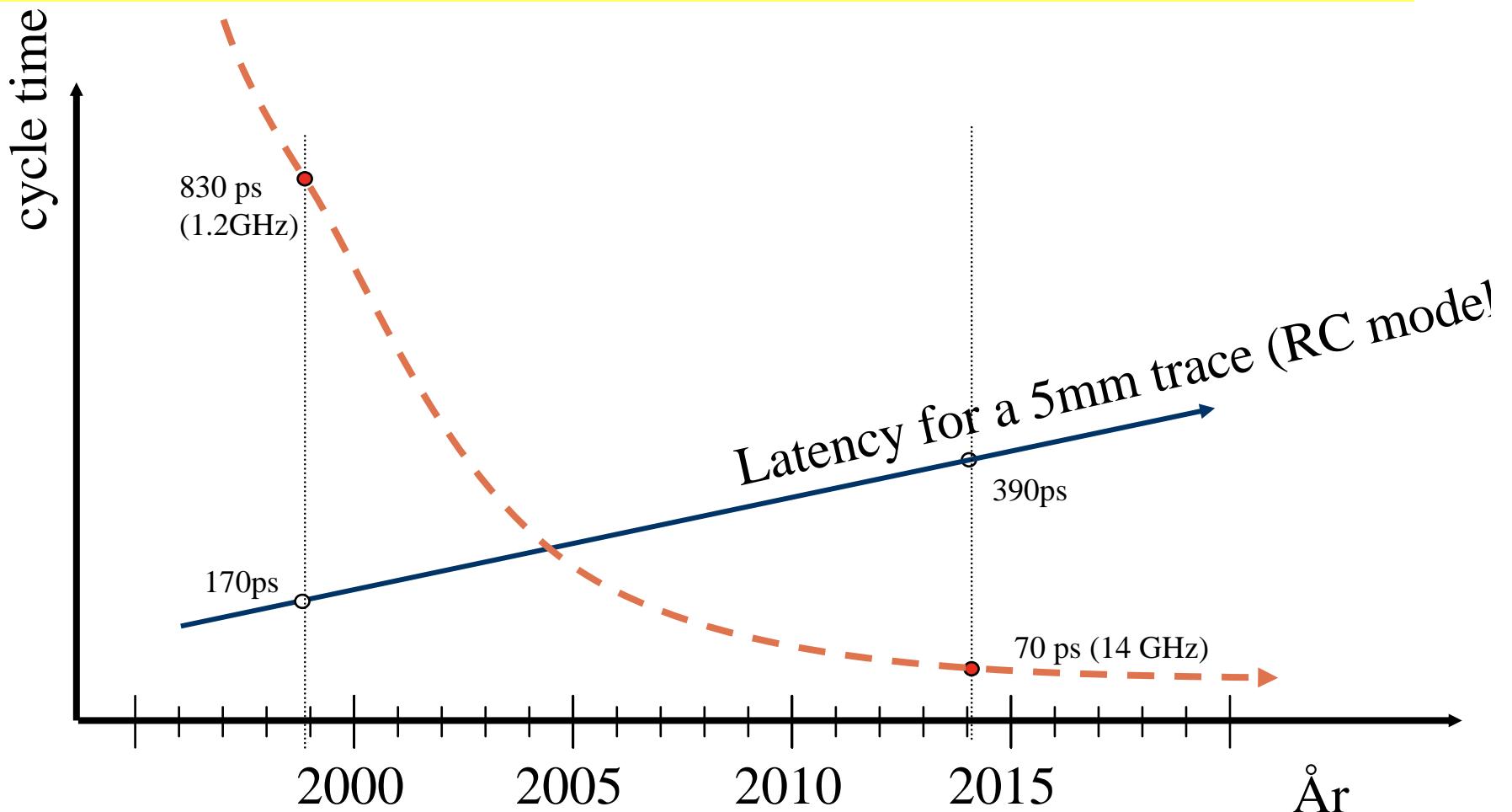
Exploring ILP (instruction-level parallelism):

- Faster clocks → Deep pipelines
- Superscalar Pipelines
- Branch Prediction
- Out-of-Order Execution
- Trace Cache
- Speculation
- Predicate Execution
- Advanced Load Addressing
- Return Address Registers
-

Bad News #1:
We have already explored most ILP
(instruction-level parallelism)

Bad News #2

Looong wire delay → slow CPUs



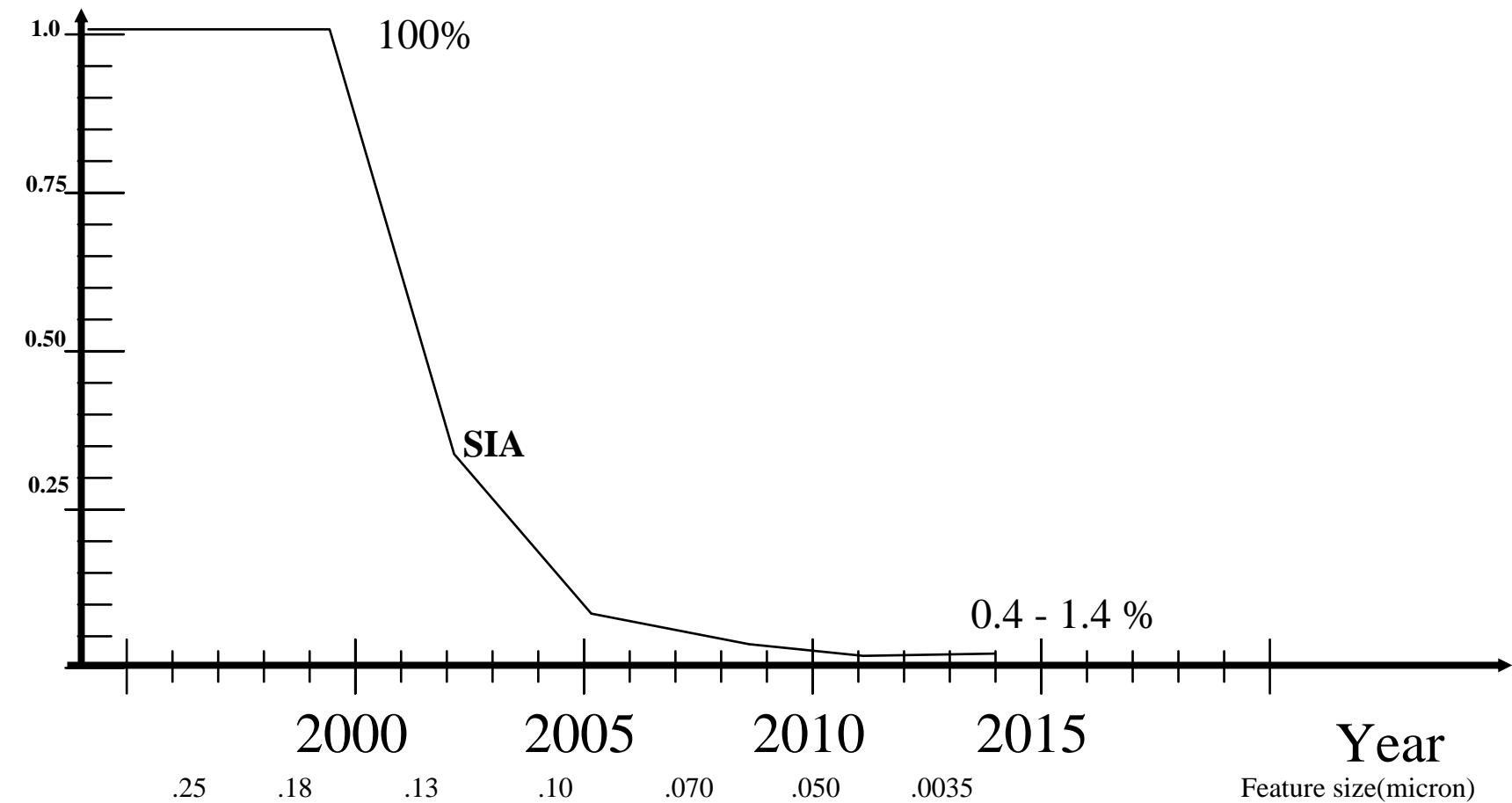
DARK
2009

*Quantitative data and trends according to V. Agarwal et al., ISCA 2000
Based on SIA (Semiconductor Industry Association) prediction, 1999*

Bad News #2

Looong wire delay → slow CPUs

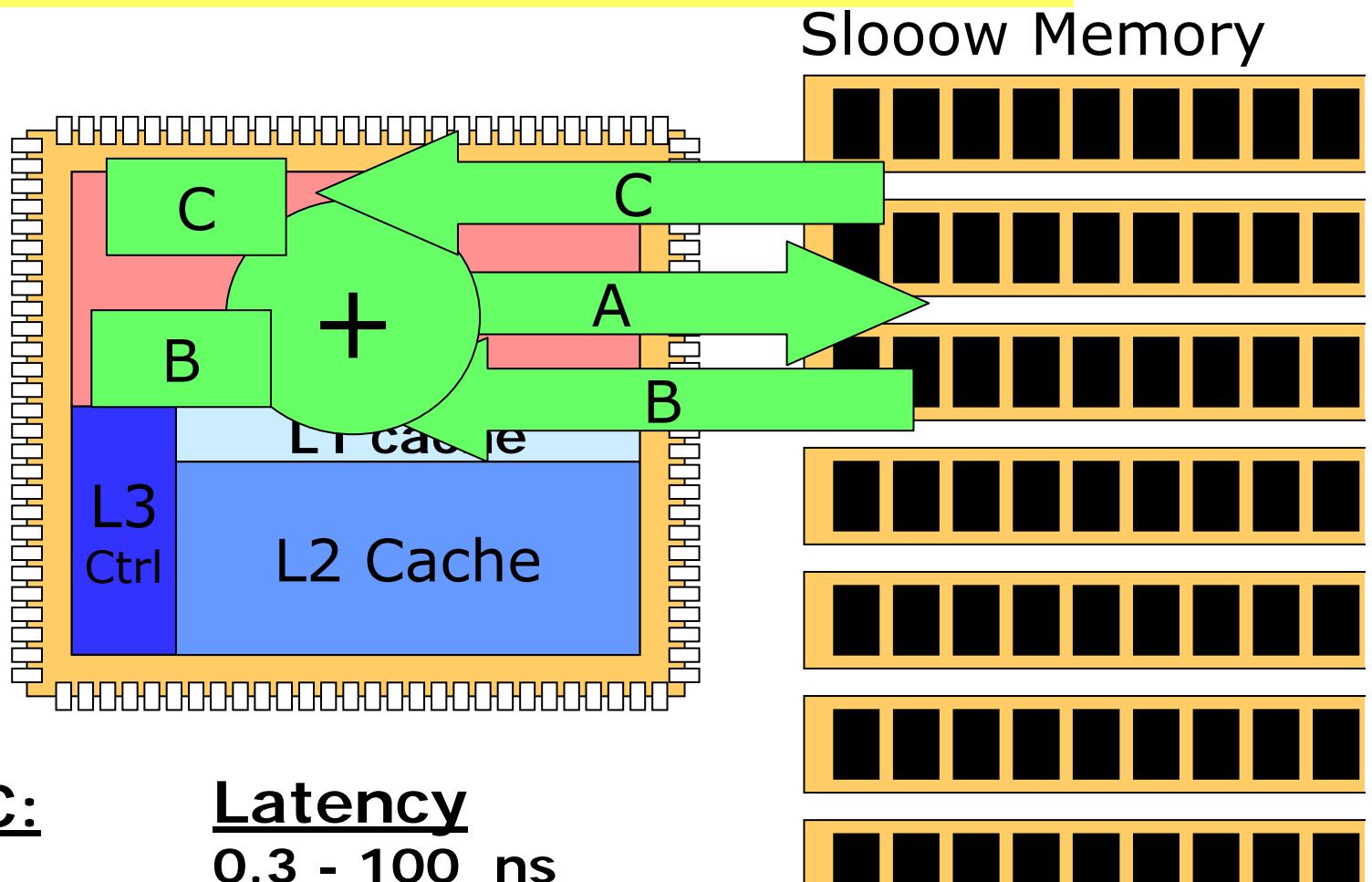
Span -- Fraction of chip reachable in one cycle



DARK
2009

Bad News #3:

Memory latency/bandwidth is the bottleneck...



A = B + C:
Read B
Read C
Add B & C
WriteA

Latency
0.3 - 100 ns
0.3 - 100 ns
0.3 ns
0.3 - 100 ns

Bad News #4: Power is the limit

- Power consumption is the bottleneck
 - ✿ Cooling servers is hard
 - ✿ Battery lifetime for mobile computers
 - ✿ Energy is money

- Dynamic effect is proportional to:

$$P_{\text{dyn}} \sim \text{Capacitance} * \text{Frequency} * \text{Voltage}^2$$

Now What?

#1: Running out of ILP

#2: Wire delay is starting to hurt

#3: Memory is the bottleneck

#4: Power is the limit

DARK
2009

Solving all the problems: exploring threads parallelism

#1: Running out of ILP

→ feed one CPU with instr. from many threads

#2: Wire delay is starting to hurt

→ Multiple small CPUs with private L1\$

#3: Memory is the bottleneck

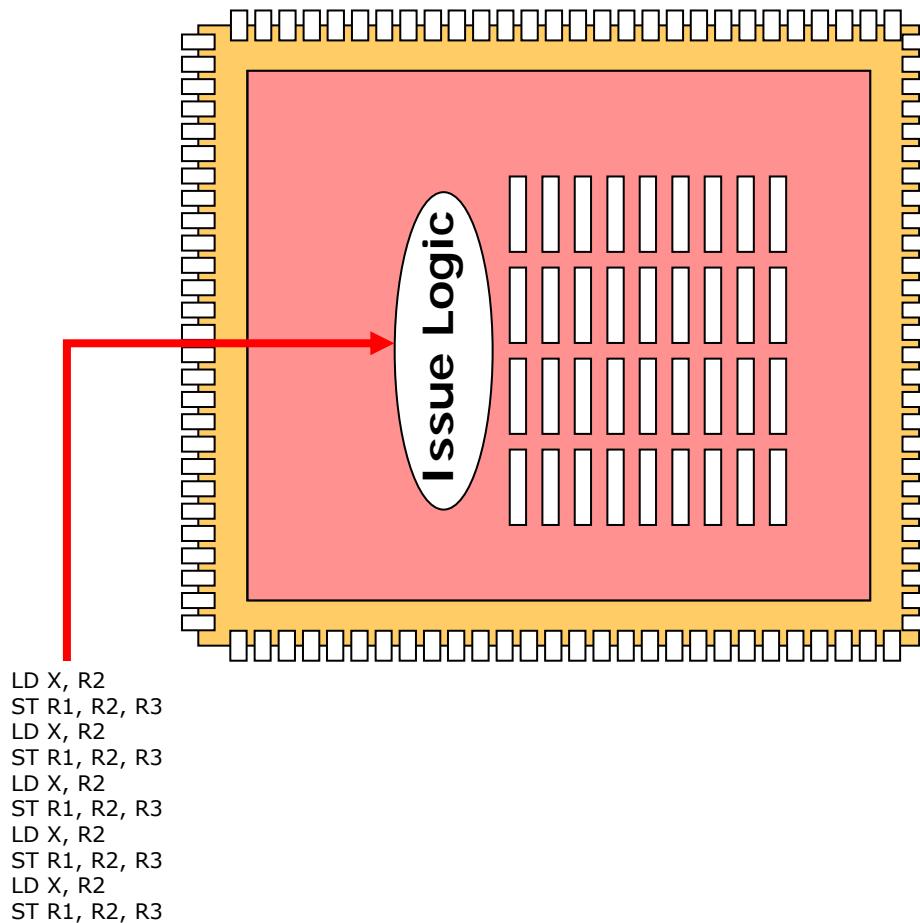
→ memory accesses from many threads (MLP)

#4: Power is the limit

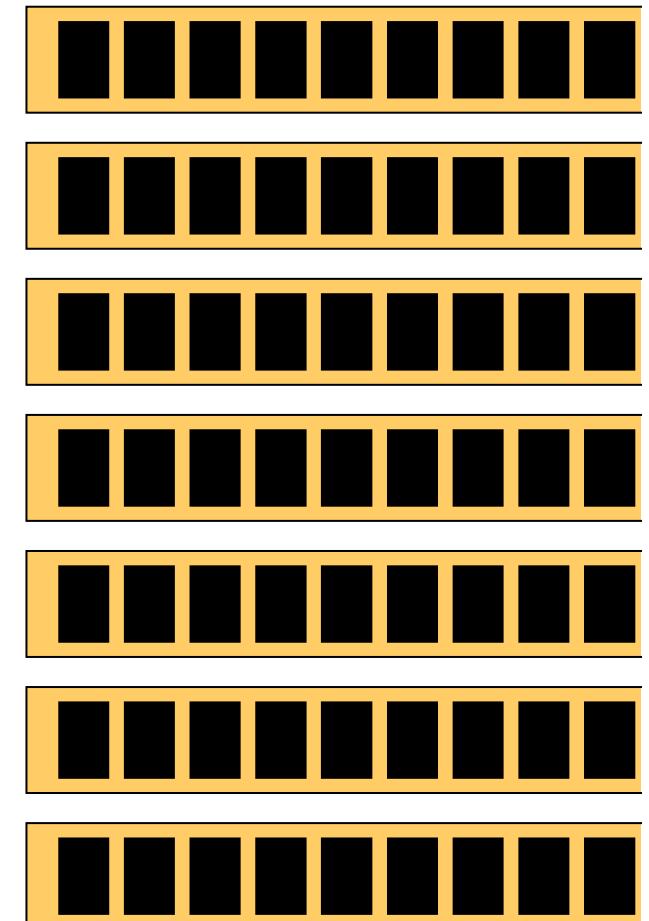
→ Lower the frequency → lower voltage

Bad News #1: Not enough ILP 1(2)

→ feed one CPU with instr. from many threads

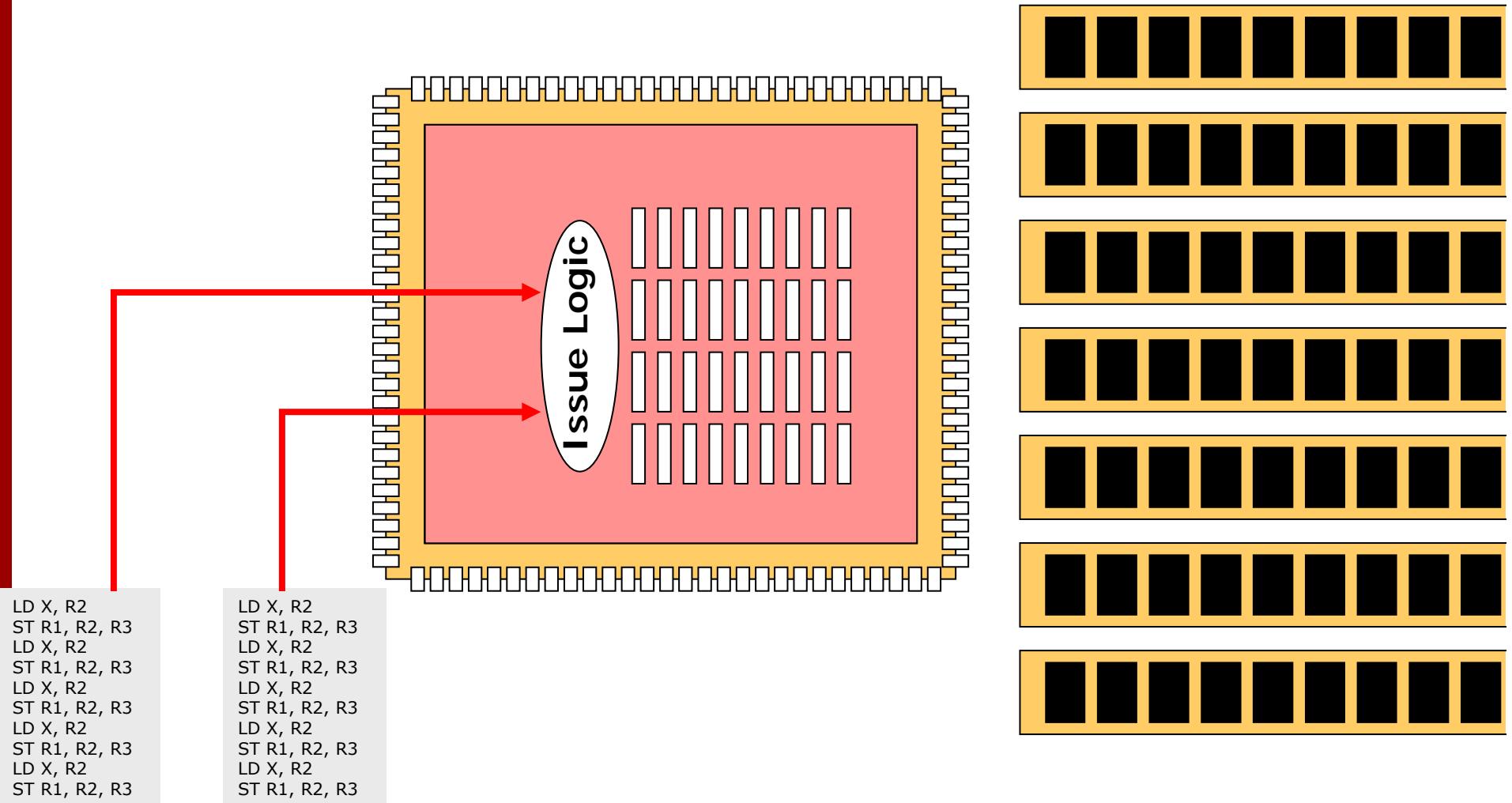


Slooow Memory



Bad News #1: Not enough ILP 2(2)

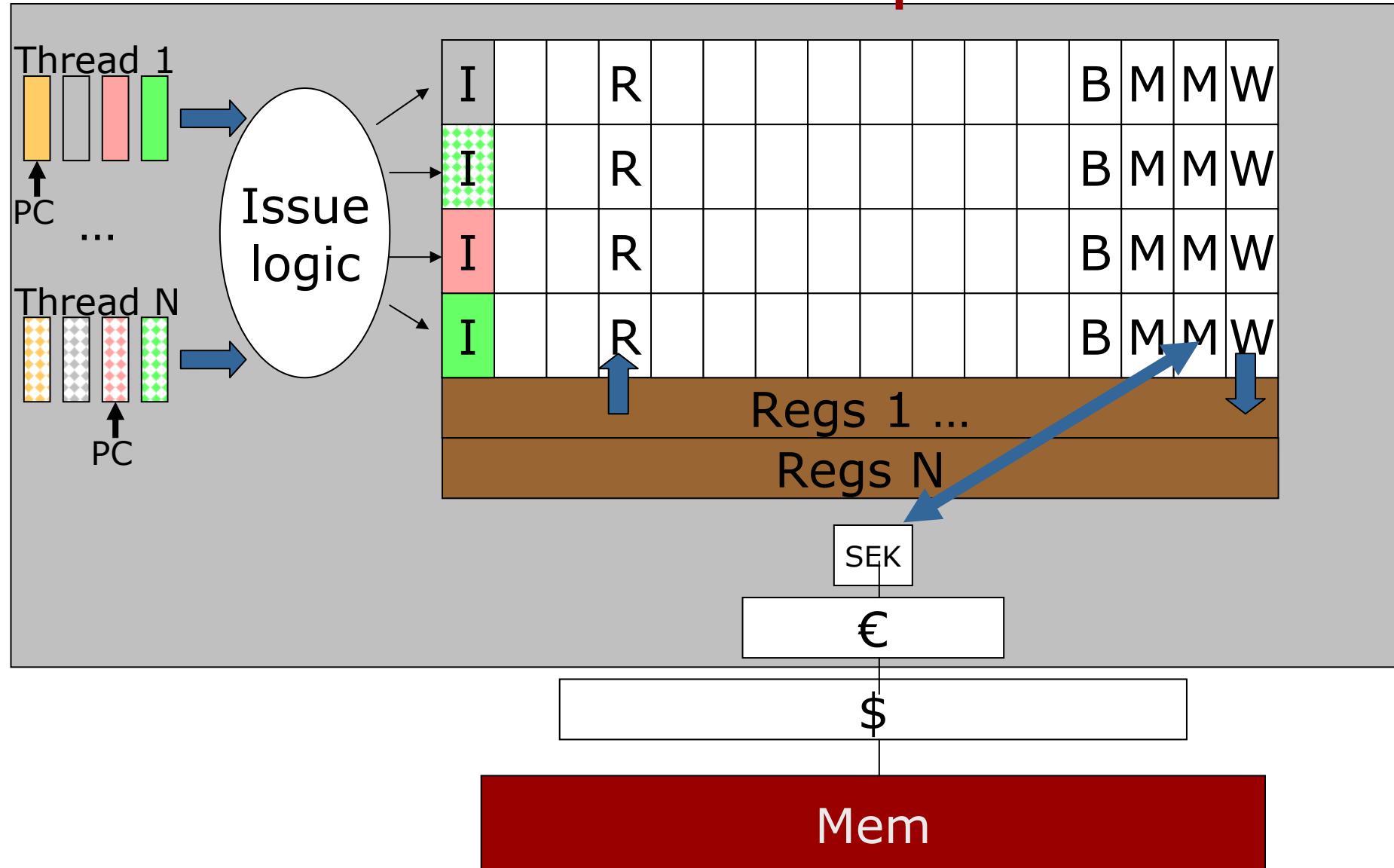
→ feed one CPU with instr. from many threads





SMT: Simultaneous Multithreading

"Combine TLP&ILP to find independent instr."



Thread-interleaved

Historical Examples:

- Denelcor, HEP, Tera Computers [B. Smith] 1984

Each thread executes every n:th cycle in a round-robin fashion

- Poor single-thread performance
- Expensive (due to early adoption)

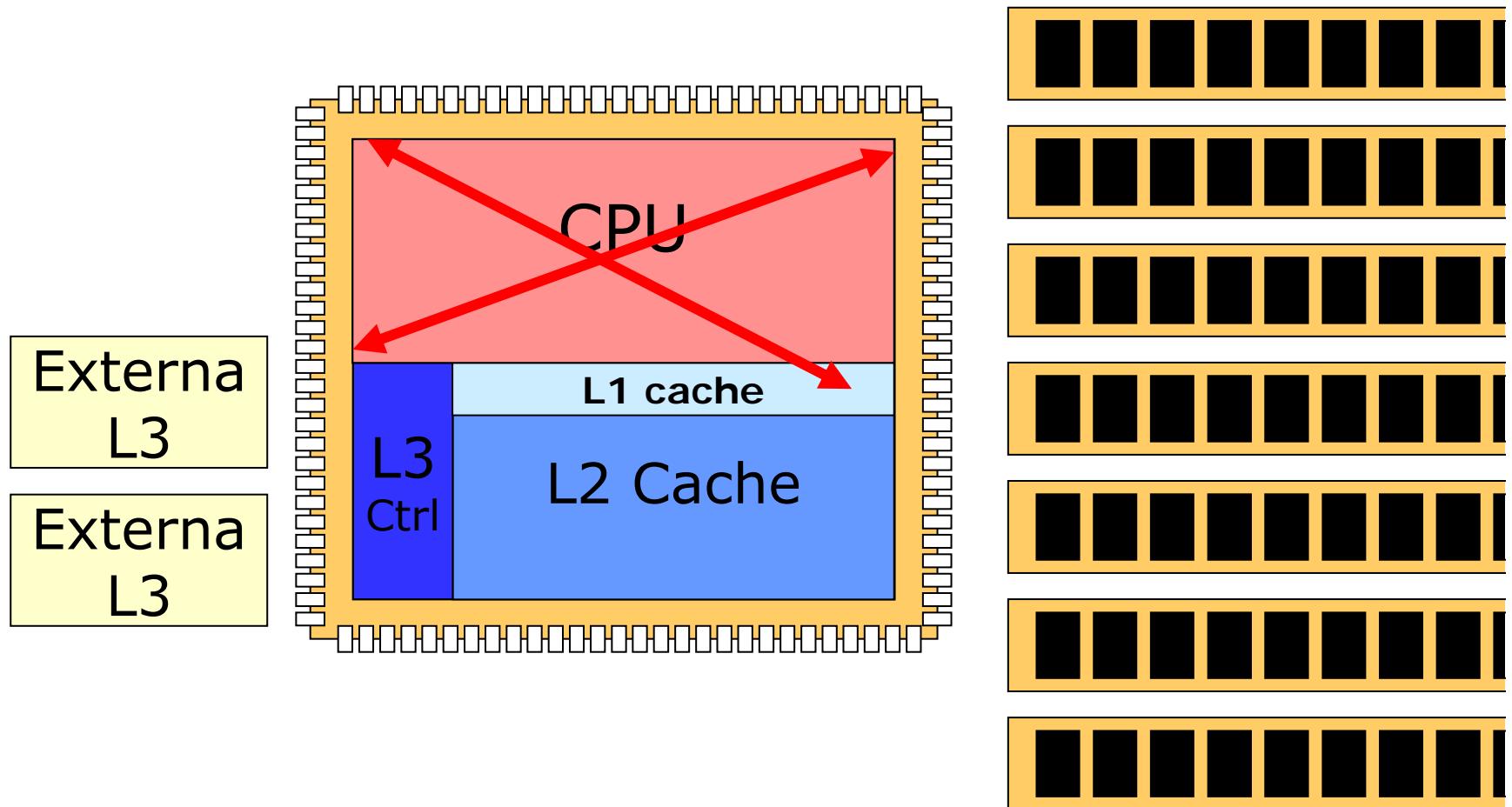
- Intel's Hyperthreading (2002??)

- Poor implementation



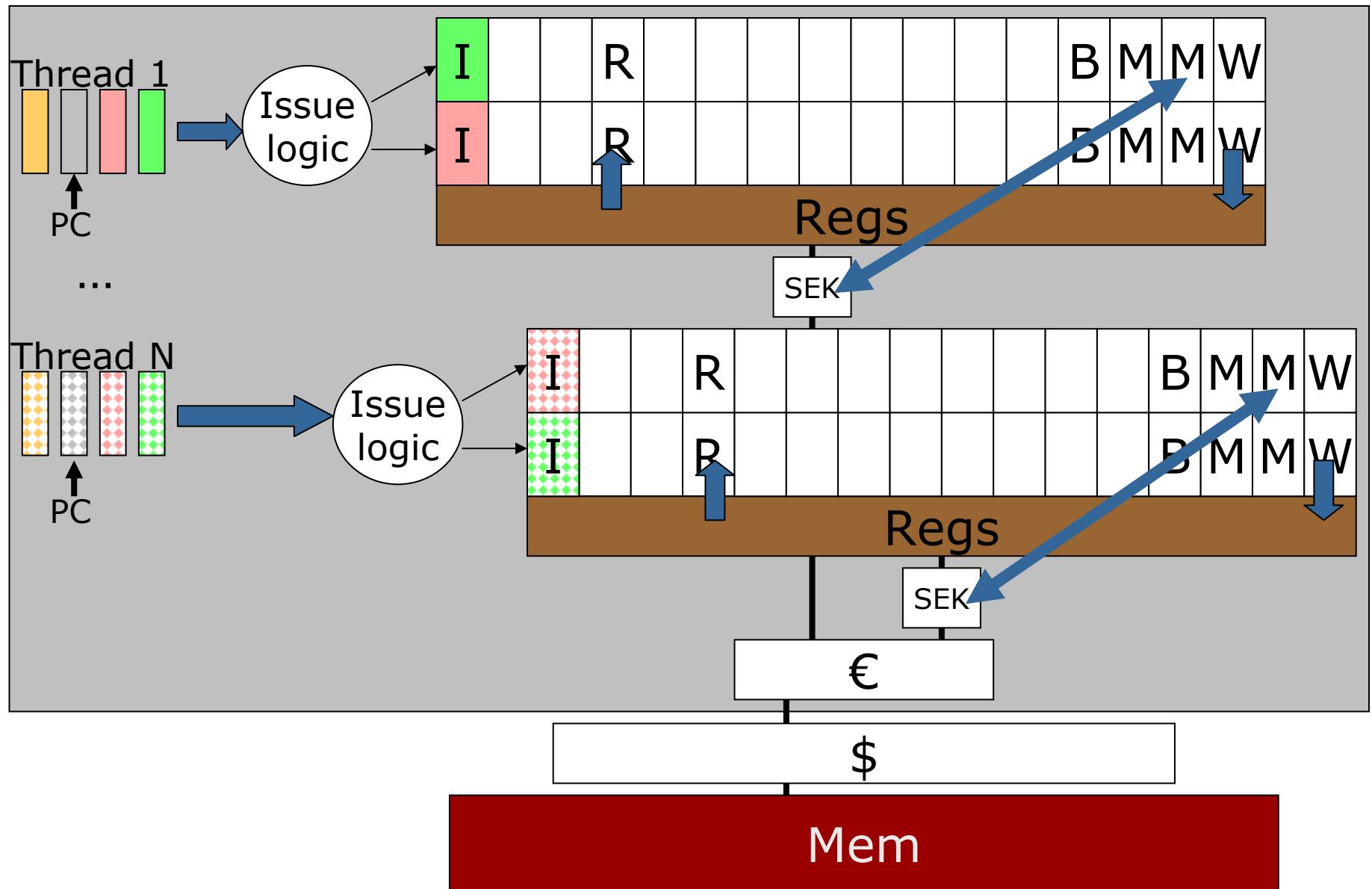
Bad News #2: wire delay

→ Multiple small CPUs with private L1\$





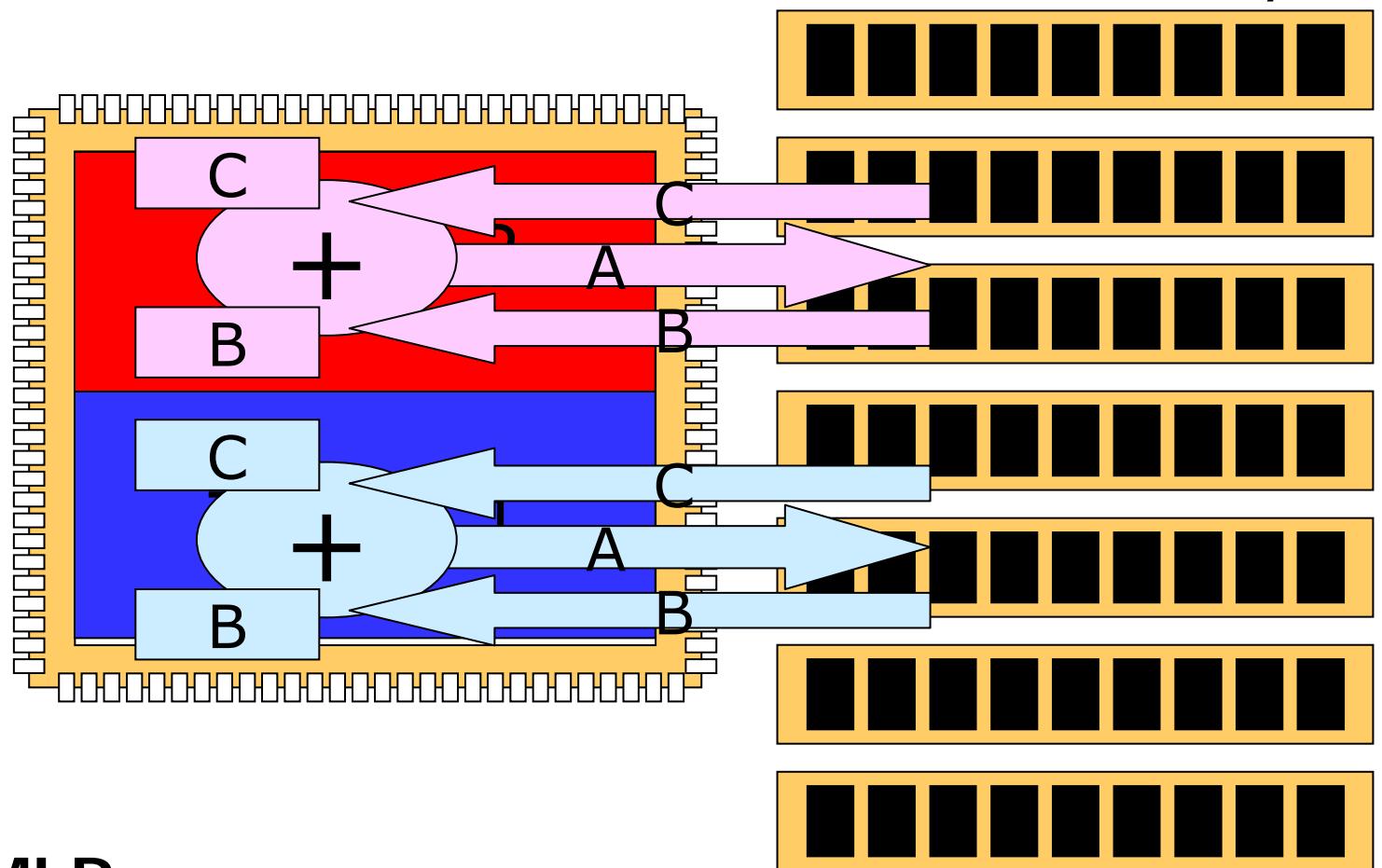
CMP: Chip Multiprocessor more TLP & geographical locality



Bad News #3: memory latency/bandwidth

→ memory accesses from many threads (MLP)

Sloooow memory



TLP → MLP

Thread-Level Parallelism → Memory-Level Parallelism (MLP)



Bad News #4: Power consumption

→ Lower the frequency → lower voltage

$$P_{\text{dyn}} = C * f * V^2 \approx \text{area} * \text{freq} * \text{voltage}^2$$

CPU
freq=f

VS.

CPU
freq=f/2

CPU
freq=f/2

$$P_{\text{dyn}}(C, f, V) = CfV^2$$

$$P_{\text{dyn}}(2C, f/2, <V) < CfV^2$$

CPU
freq=f

VS.

CPU	CPU
CPU	CPU

freq = f/2

$$P_{\text{dyn}}(C, f, V) = CfV^2$$

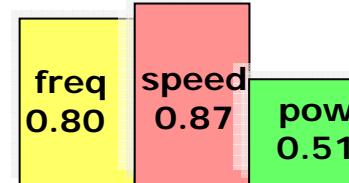
$$P_{\text{dyn}}(C, f/2, <V) < \frac{1}{2} CfV^2$$

Throughput < 2x

Example: Freq. Scaling



20% higher freq.



20% lower freq.



20% lower freq.
Two cores



Multicore CPUs:

Ageia PhysX, a multi-core physics processing unit.

AmbriC Am2045, a 336-core Massively Parallel Processor Array (MPPA)

AMD

- Athlon 64, Athlon 64 FX and Athlon 64 X2 family, dual-core desktop processors.
- Opteron, dual- and quad-core server/workstation processors.
- Phenom, triple- and quad-core desktop processors.
- Sempron X2, dual-core entry level processors.
- Turion 64 X2, dual-core laptop processors.
- Radeon and FireStream multi-core GPU/GPGPU (10 cores, 16 5-issue wide superscalar stream processors per core)

ARM MPCore is a fully synthesizable multicore container for ARM9 and ARM11 processor cores, intended for high-performance embedded and entertainment applications.

Azul Systems Vega 2, a 48-core processor.

Broadcom SiByte SB1250, SB1255 and SB1455.

Cradle Technologies CT3400 and CT3600, both multi-core DSPs.

Cavium Networks Octeon, a 16-core MIPS MPU.

HP PA-8800 and PA-8900, dual core PA-RISC processors.

IBM

- POWER4, the world's first dual-core processor, released in 2001.
- POWER5, a dual-core processor, released in 2004.
- POWER6, a dual-core processor, released in 2007.
- PowerPC 970MP, a dual-core processor, used in the Apple Power Mac G5.
- Xenon, a triple-core, SMT-capable, PowerPC microprocessor used in the Microsoft Xbox 360 game console.

IBM, Sony, and Toshiba Cell processor, a nine-core processor with one general purpose PowerPC core and eight specialized SPUs (Synergistic Processing Unit) optimized for vector operations used in the Sony PlayStation 3.

Infineon Danube, a dual-core, MIPS-based, home gateway processor.

Intel

- Celeron Dual Core, the first dual-core processor for the budget/entry-level market.
- Core Duo, a dual-core processor.
- Core 2 Duo, a dual-core processor.
- Core 2 Quad, a quad-core processor.
- Core i7, a quad-core processor, the successor of the Core 2 Duo and the Core 2 Quad.
- Itanium 2, a dual-core processor.
- Pentium D, a dual-core processor.
- Teraflops Research Chip (Polaris), an 3.16 GHz, 80-core processor prototype, which the company says will be released within the next five years[6].
- Xeon dual-, quad- and hexa-core processors.

IntellaSys seaForth24, a 24-core processor.

Nvidia

- GeForce 9 multi-core GPU (8 cores, 16 scalar stream processors per core)
- GeForce 200 multi-core GPU (10 cores, 24 scalar stream processors per core)
- Tesla multi-core GPGPU (8 cores, 16 scalar stream processors per core)

Parallax Propeller P8X32, an eight-core microcontroller.

picoChip PC200 series 200-300 cores per device for DSP & wireless

Rapport Kilocore KC256, a 257-core microcontroller with a PowerPC core and 256 8-bit "processing elements".

Raza Microelectronics XLR, an eight-core MIPS MPU

Sun Microsystems

- UltraSPARC IV and UltraSPARC IV+, dual-core processors.
- UltraSPARC IIi, an eight-core, 32-thread processor.

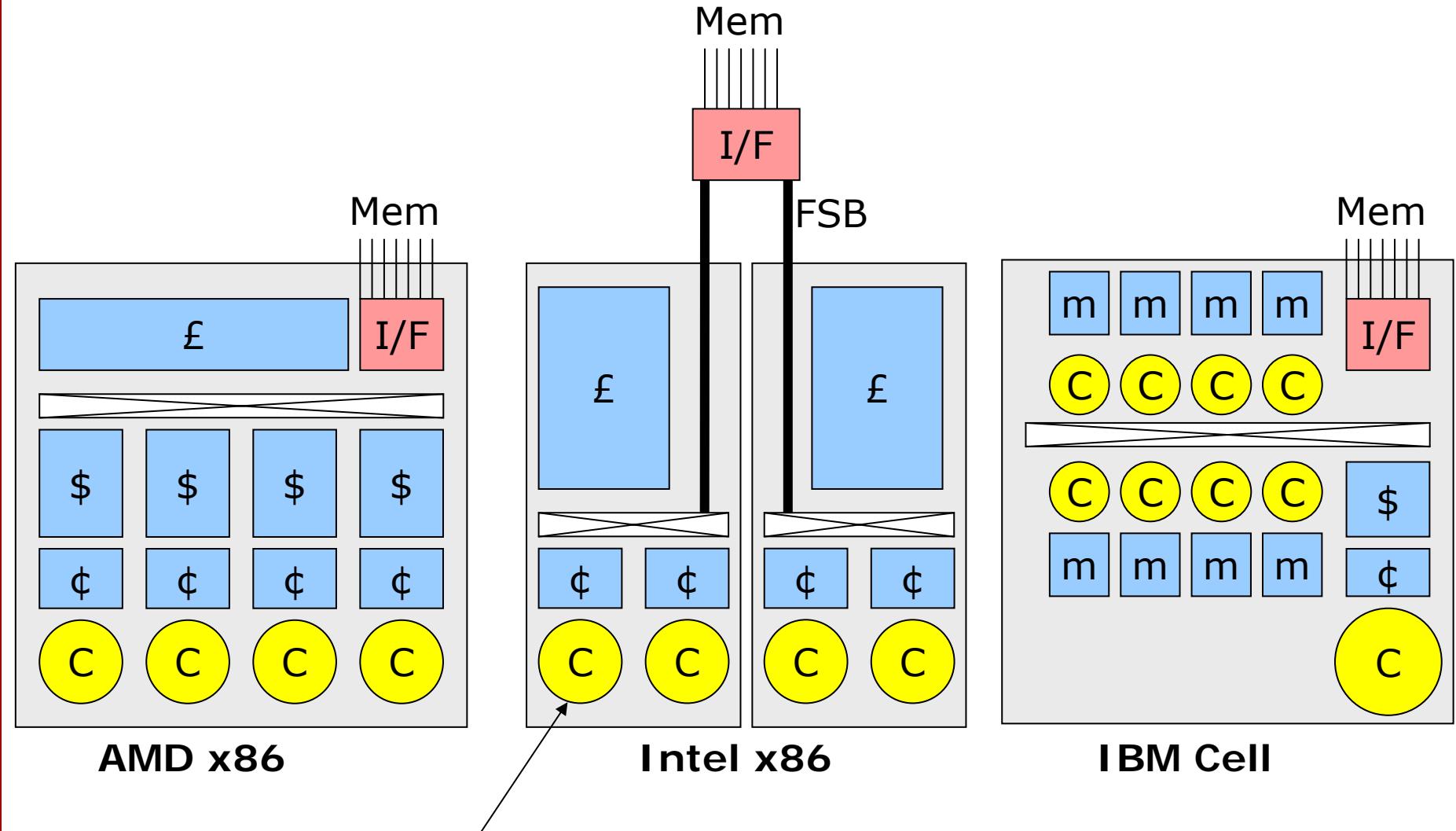


UPPSALA
UNIVERSITET

DARK
2009

High-performance single-core CPUs:

Many shapes and forms...



May run more than one thread...



Darling, I shrunk the computer

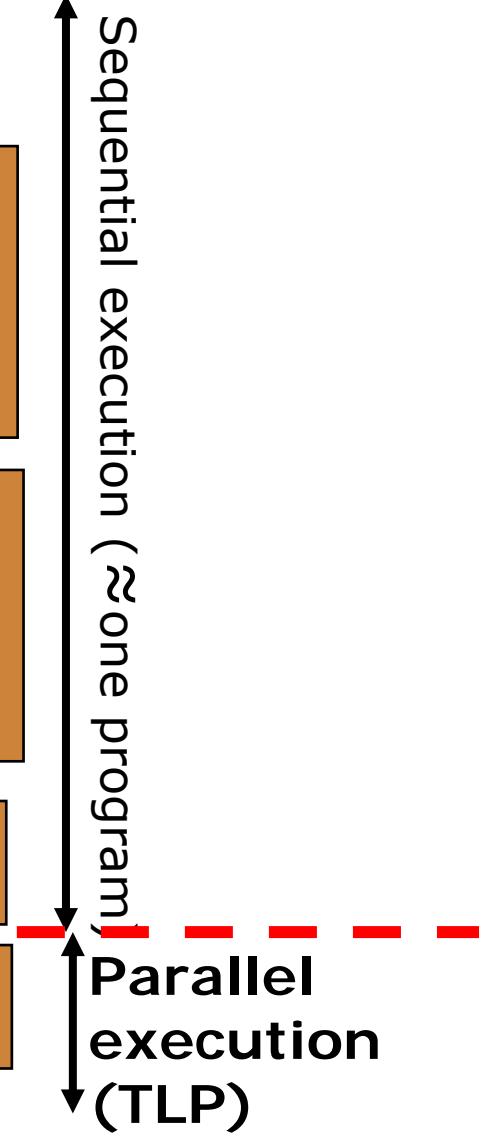
Paradigm Shift

Chip Multiprocessor (CMP): A multiprocessor on a chip!

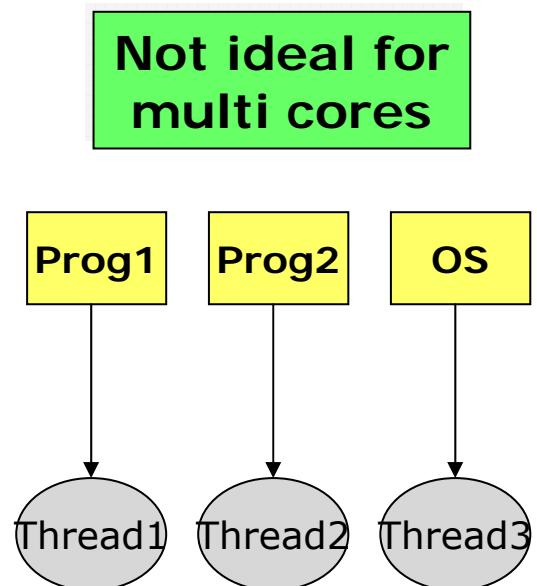
Mainframes

Super Minis:

Microprocessor:

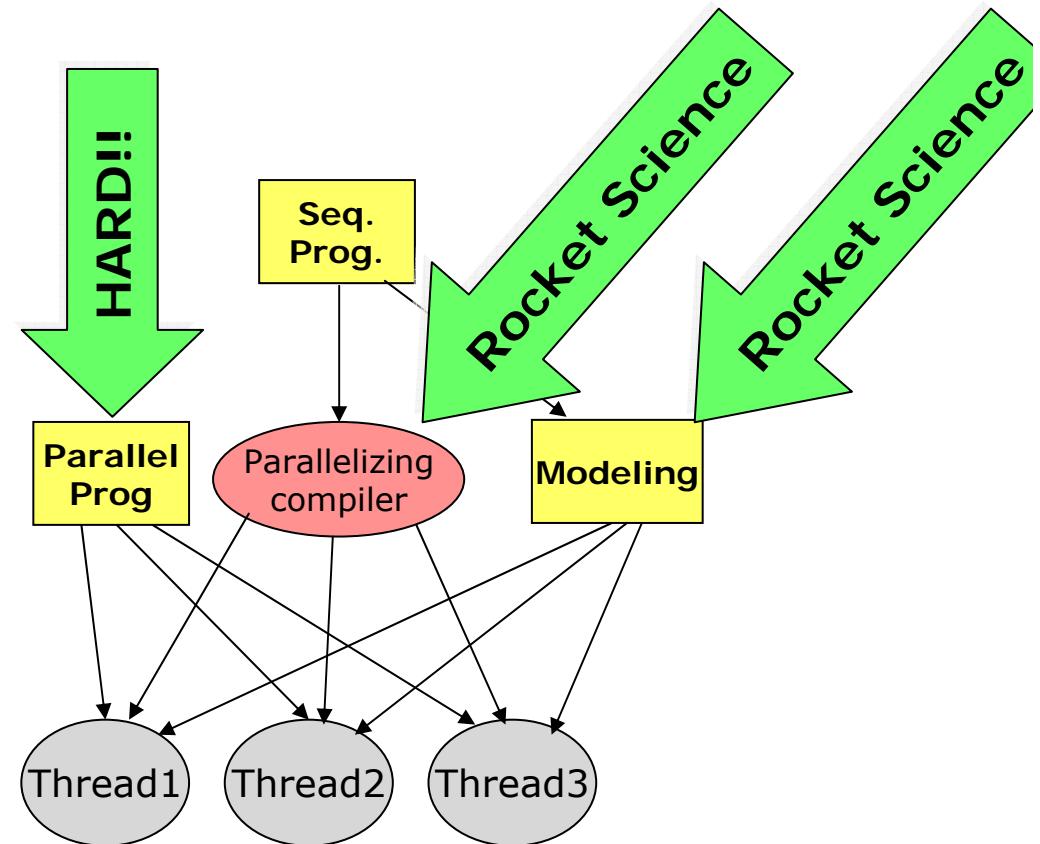


How to create threads?



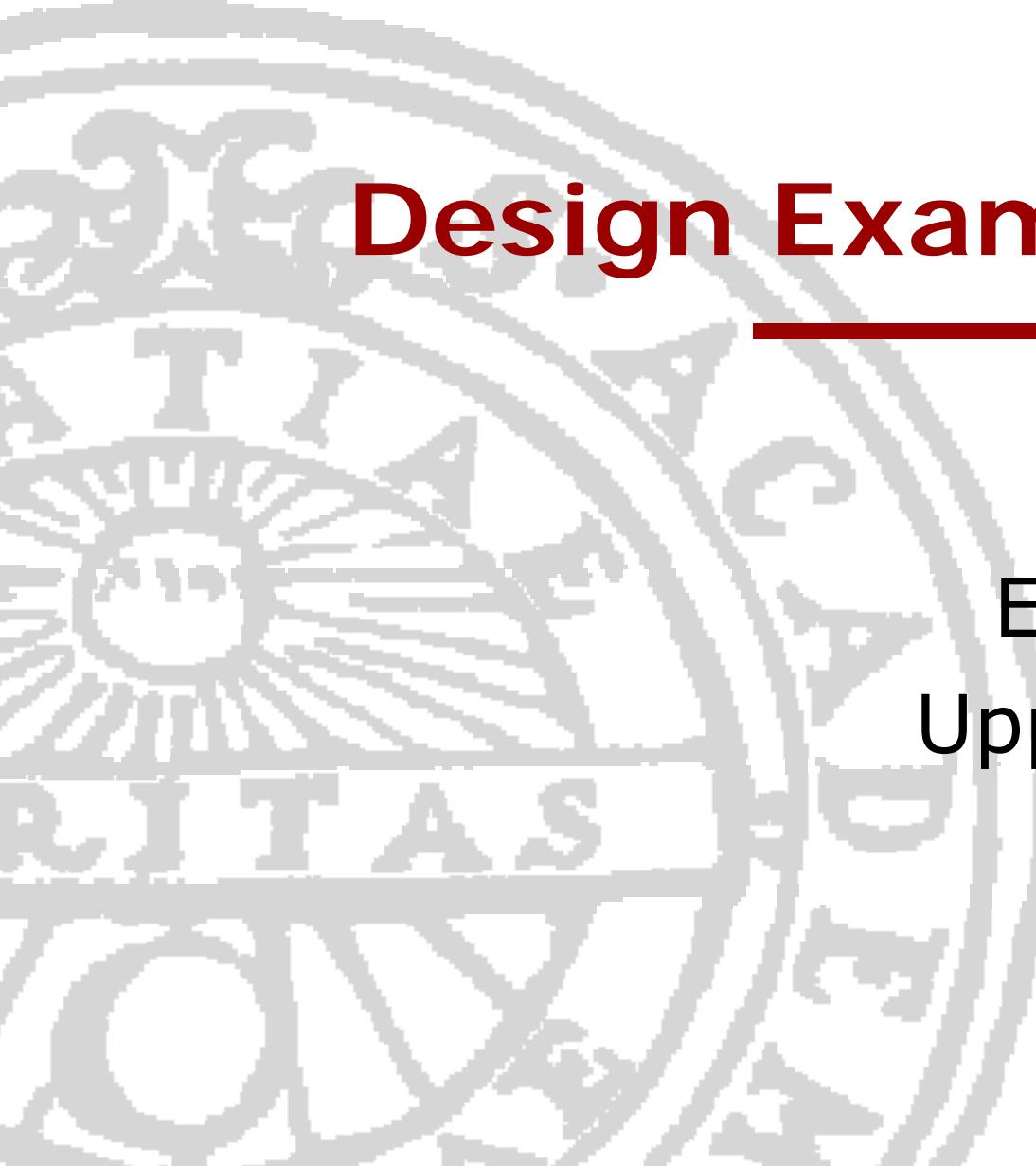
Throughput Computing

- “Multitasking”
- Virtualization
- Concurrency
- ...



Capability Computing

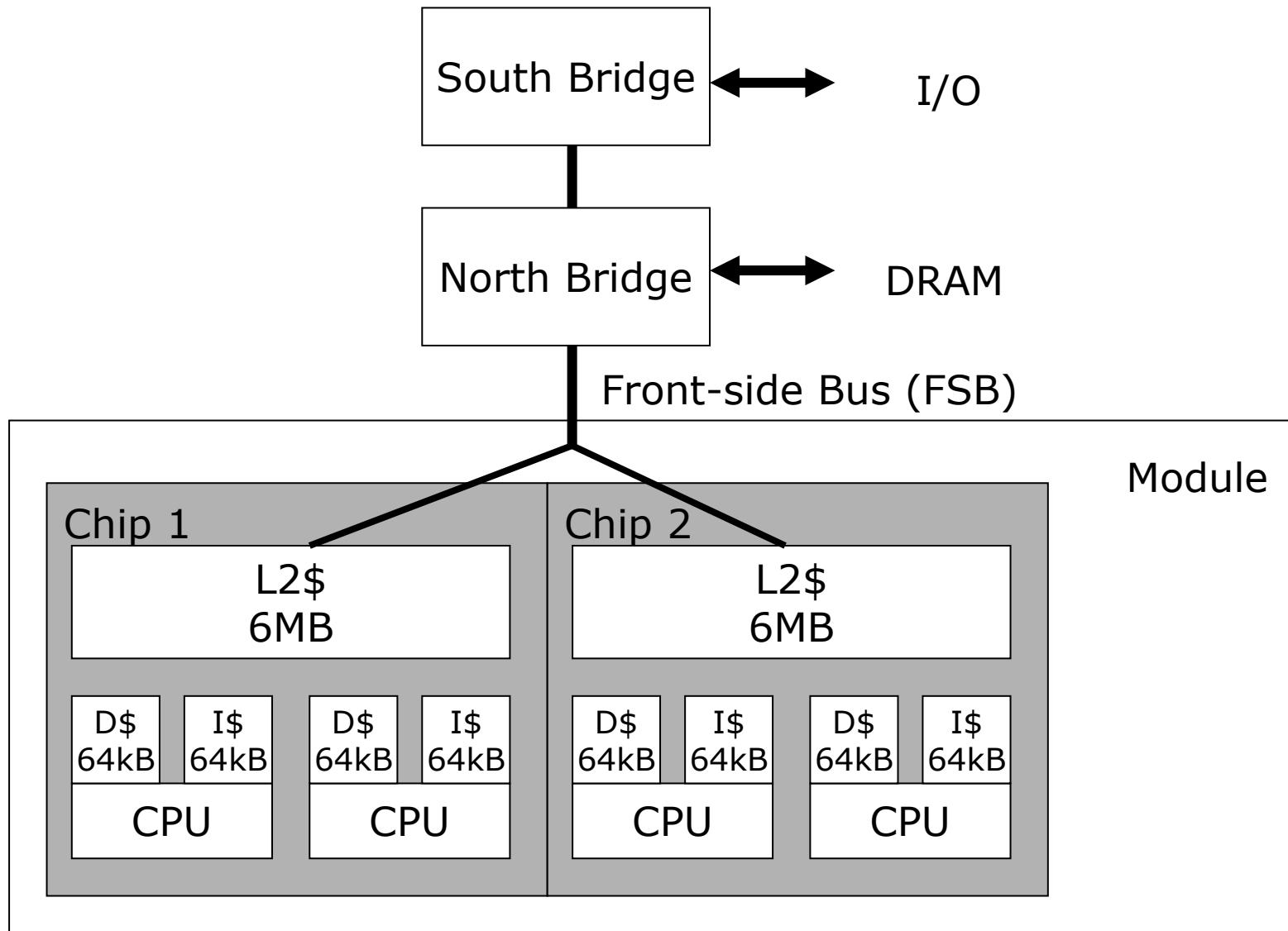
- Weather simulation
- Computer games
- ...



Design Examples CMPs

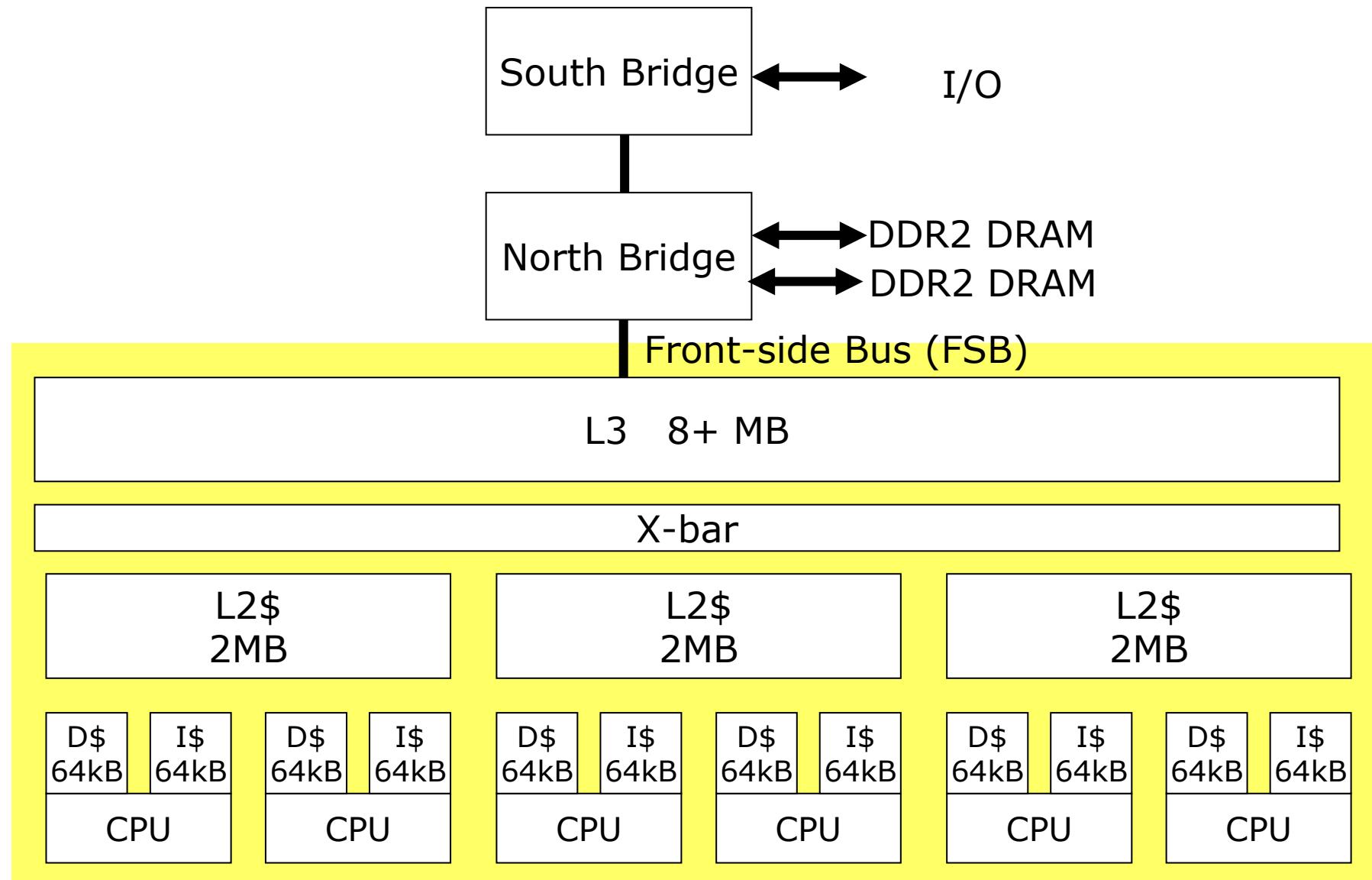
Erik Hagersten
Uppsala University
Sweden

Intel Core2 Quad, 45 nm





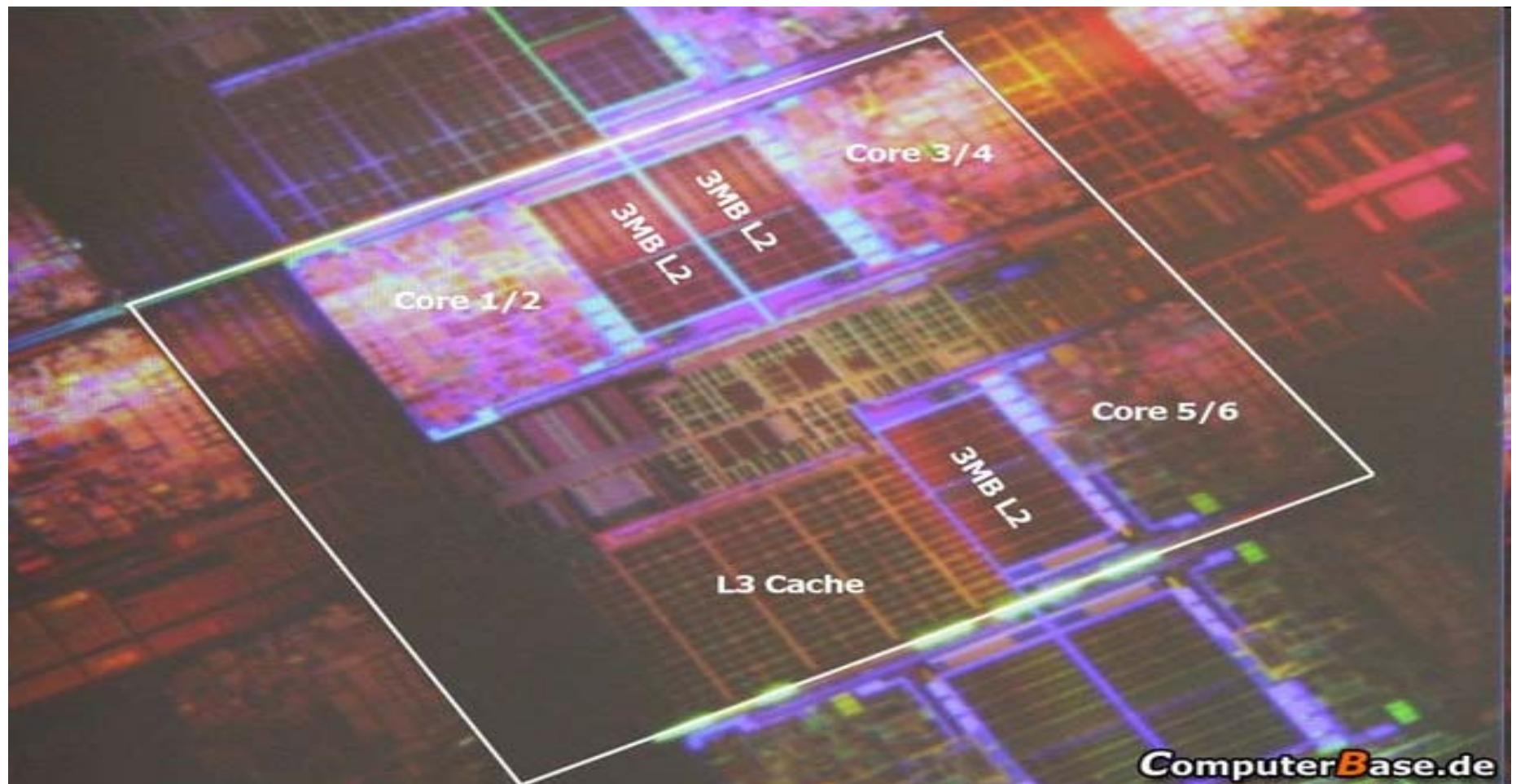
Intel: Dunnington 45nm





UPPSALA
UNIVERSITET

Intel Dunnington, 45 nm



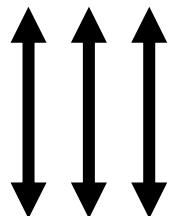
DARK
2009



UPPSALA
UNIVERSITET

AMD Barcelona, 65 nm

Hyper Transport



DDR-2



L3 2MB

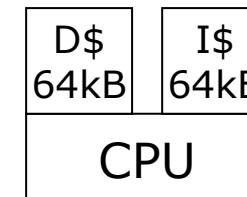
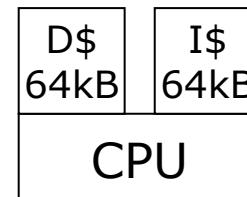
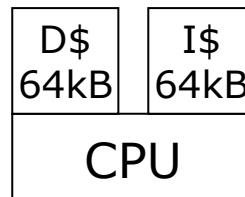
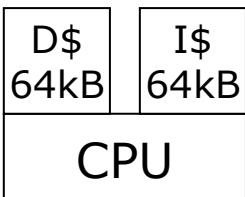
X-bar

L2\$
512kB

L2\$
512kB

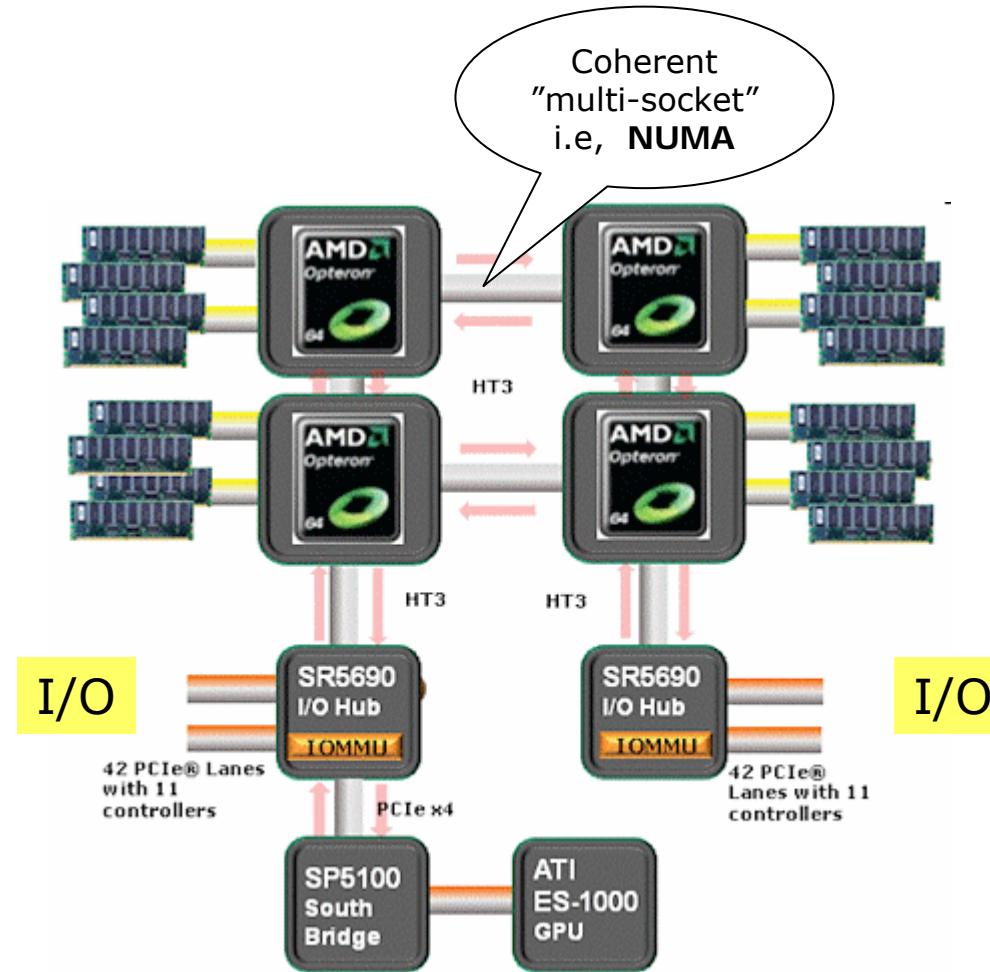
L2\$
512kB

L2\$
512kB



DARK
2009

AMD MC System Architecture



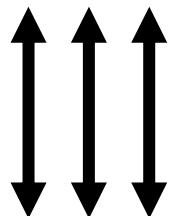
DARK
2009



UPPSALA
UNIVERSITET

AMD Shanghai, 45 nm

Hyper Transport



DDR-2, DRAM



L3 8MB

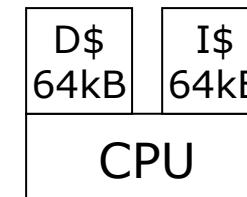
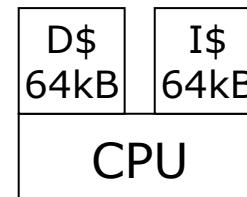
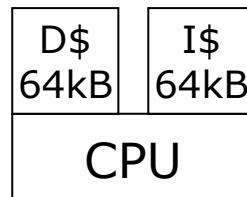
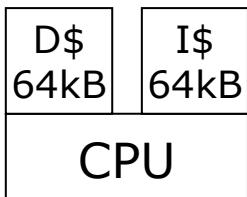
X-bar

L2\$
512kB

L2\$
512kB

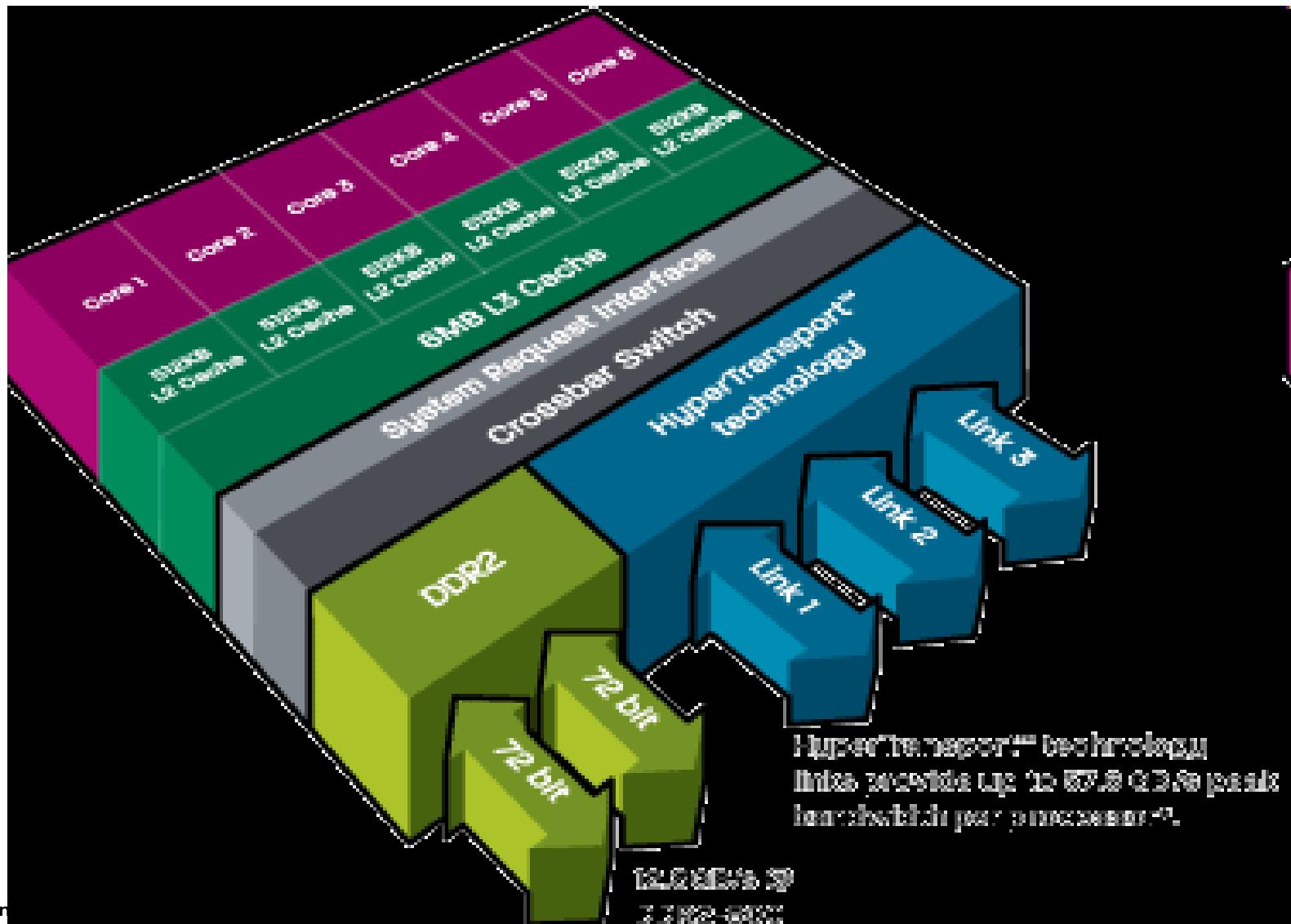
L2\$
512kB

L2\$
512kB



DARK
2009

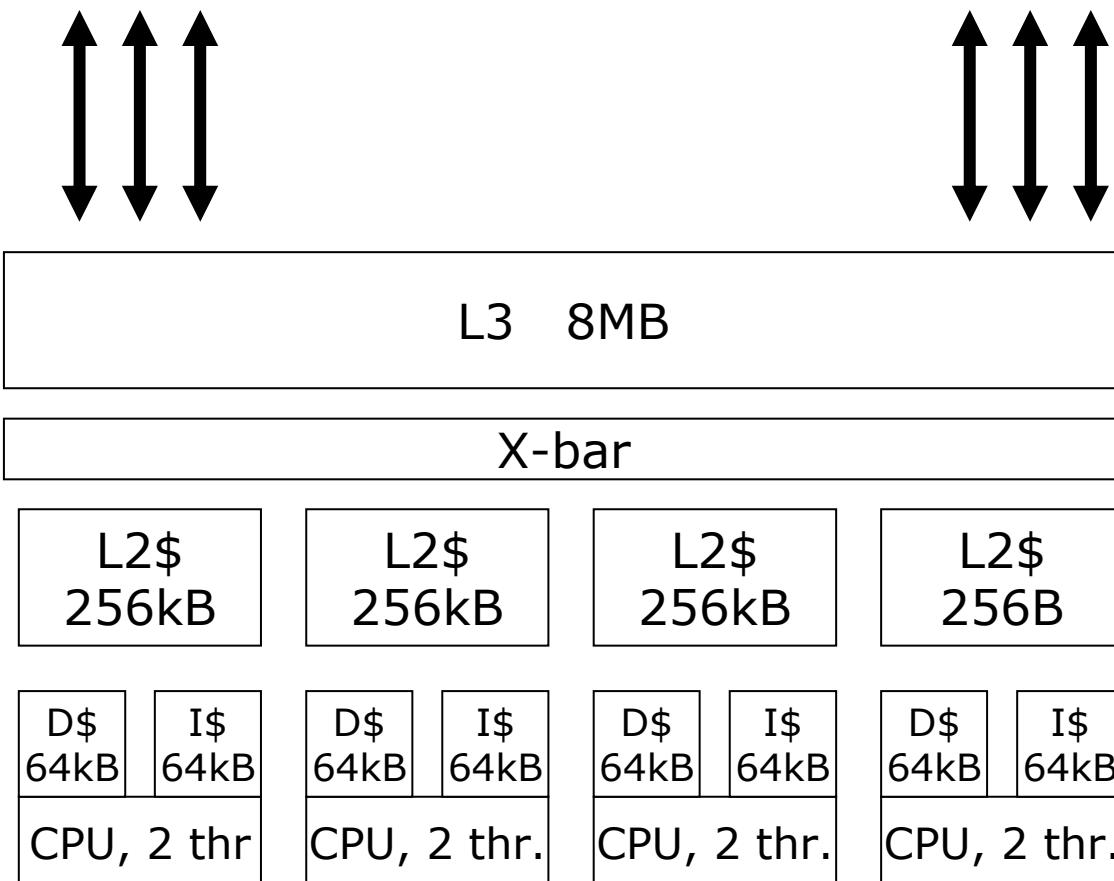
AMD Istanbul, 6 cores



Intel: Nehalem, Core i7 45 nm Q1 2009 (4 cores)

QuickPath Interconnect

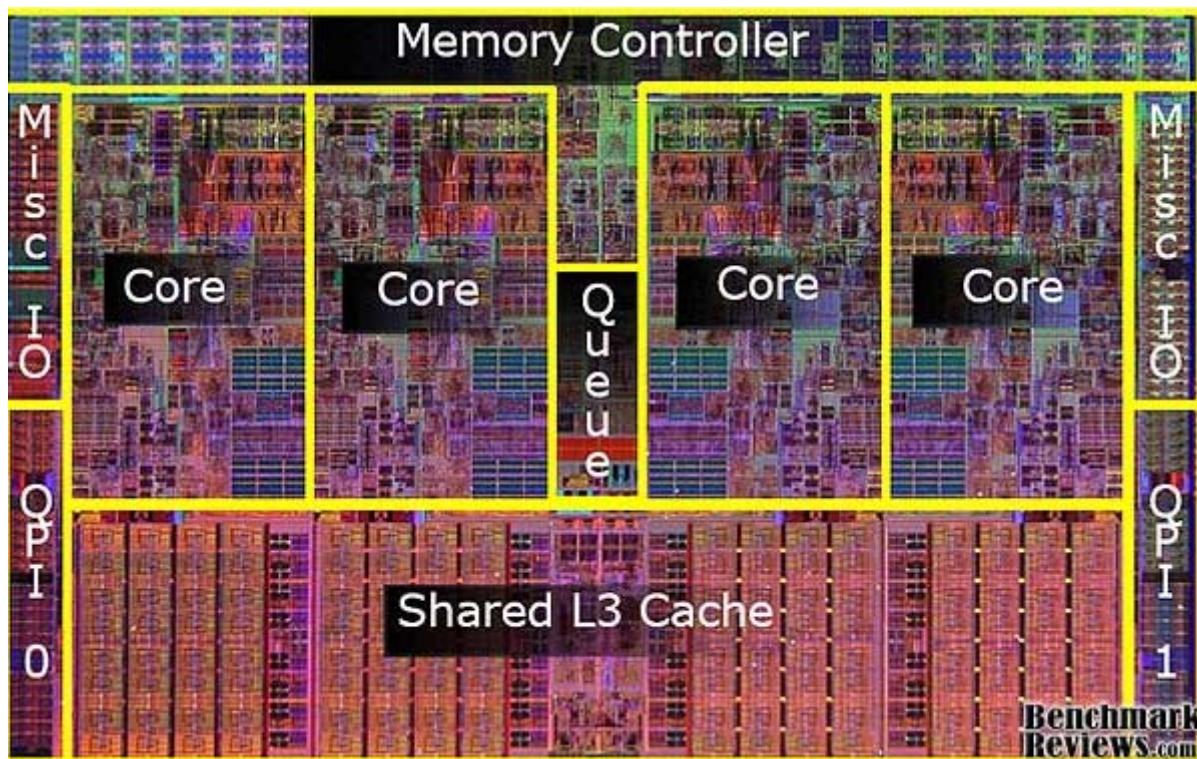
3x DDR-3 DRAM



Up to 4 cores x 2 threads

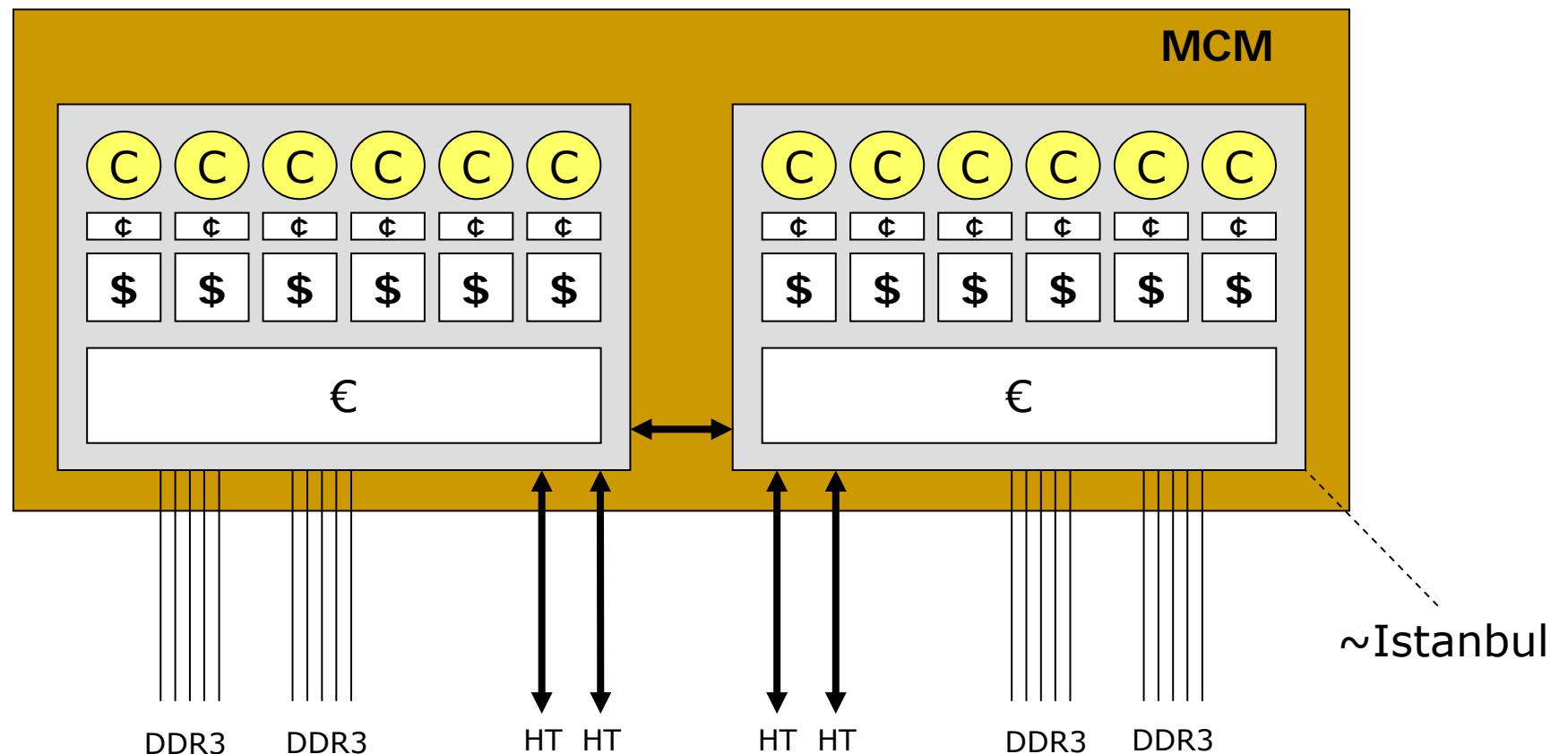
DARK
2009

Nehalem "Core i7", 45nm



DARK
2009

AMD Magny-Cours, 2010 (??)

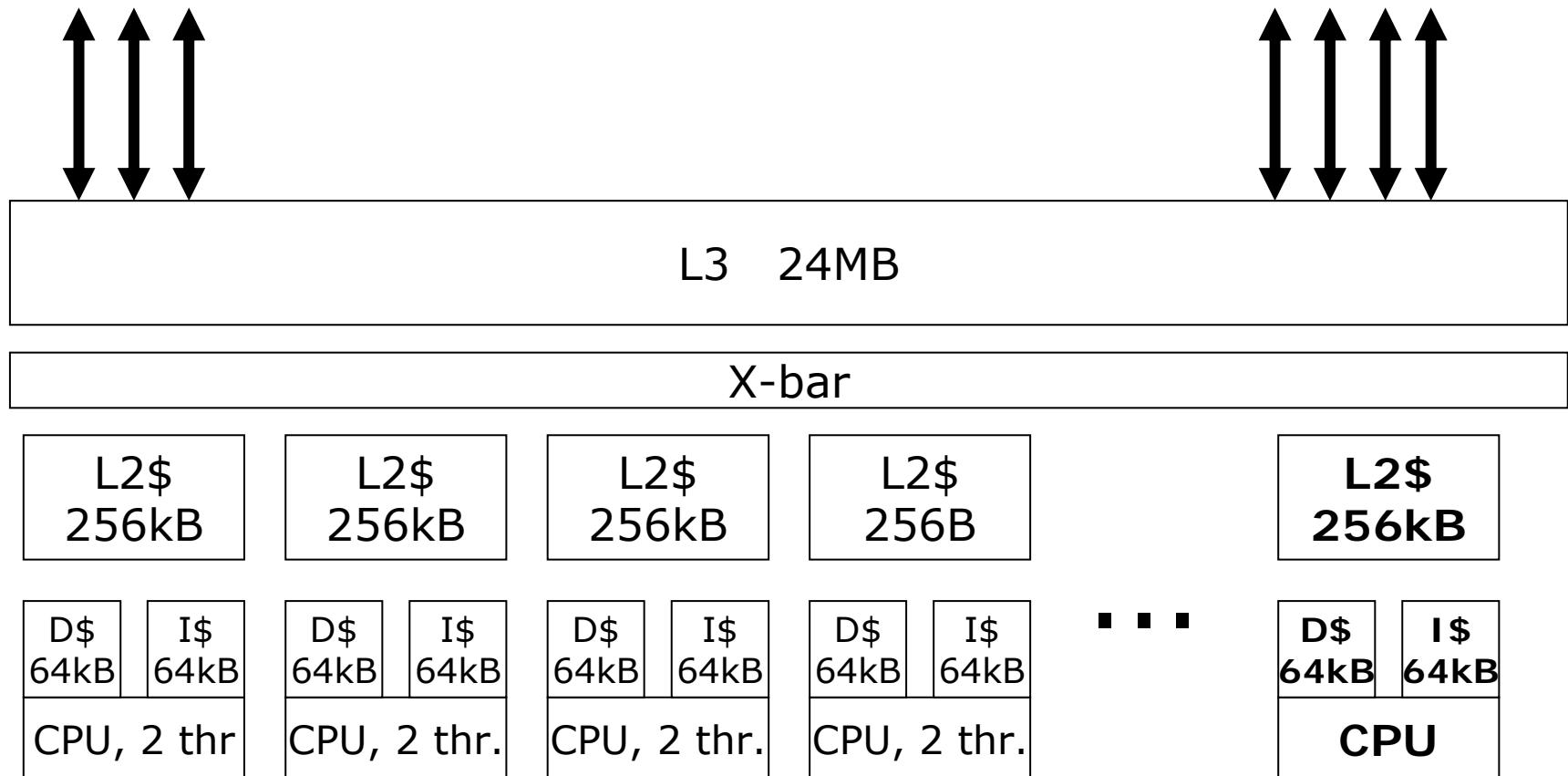


DARK
2009



Intel: Core i7 2010 (??)

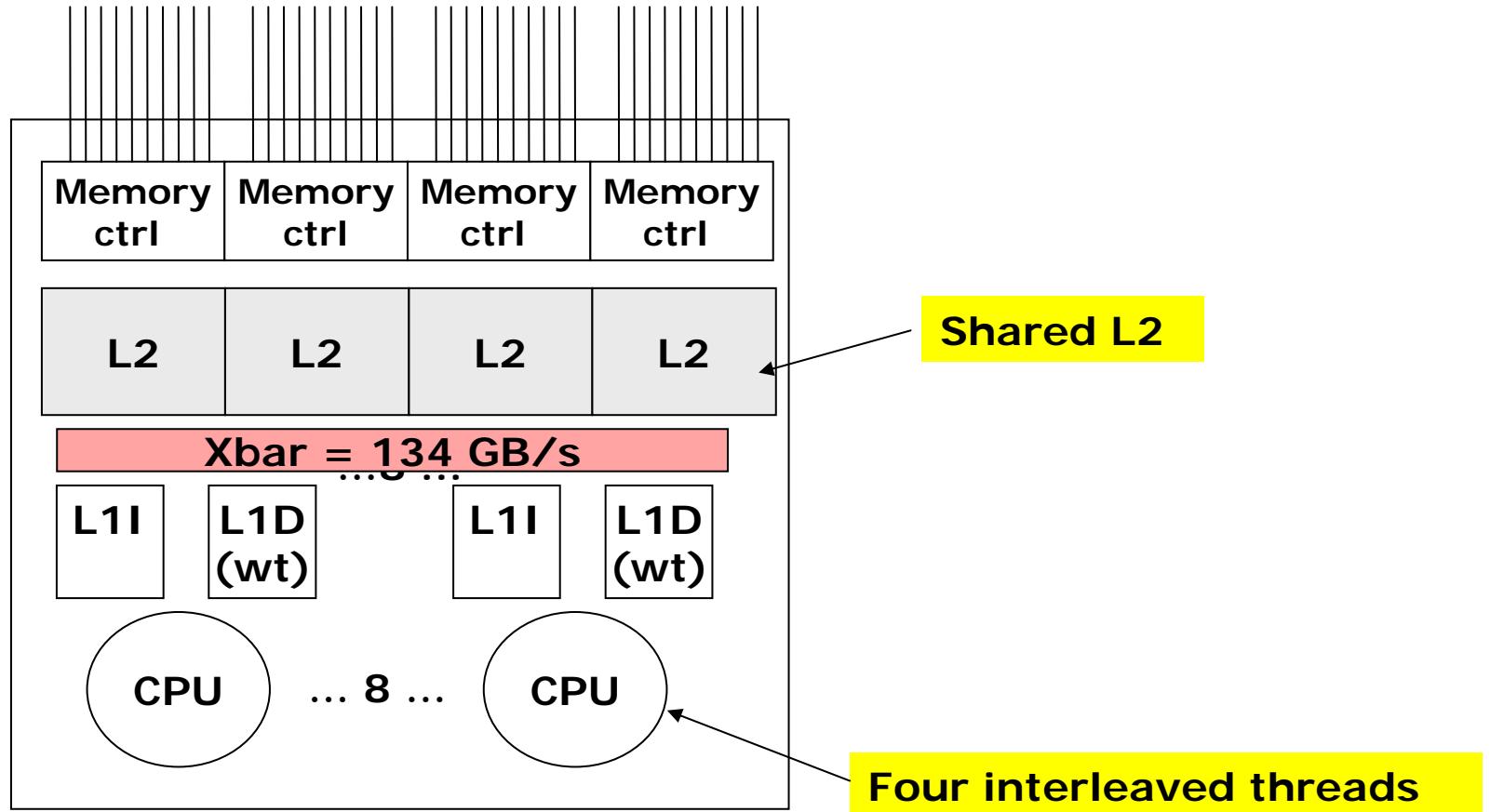
QuickPath Interconnect



8 cores x 2 threads

Sun Niagara, 2005!!

$4 \times \text{DDR-2} = 25\text{GB/s} (!)$

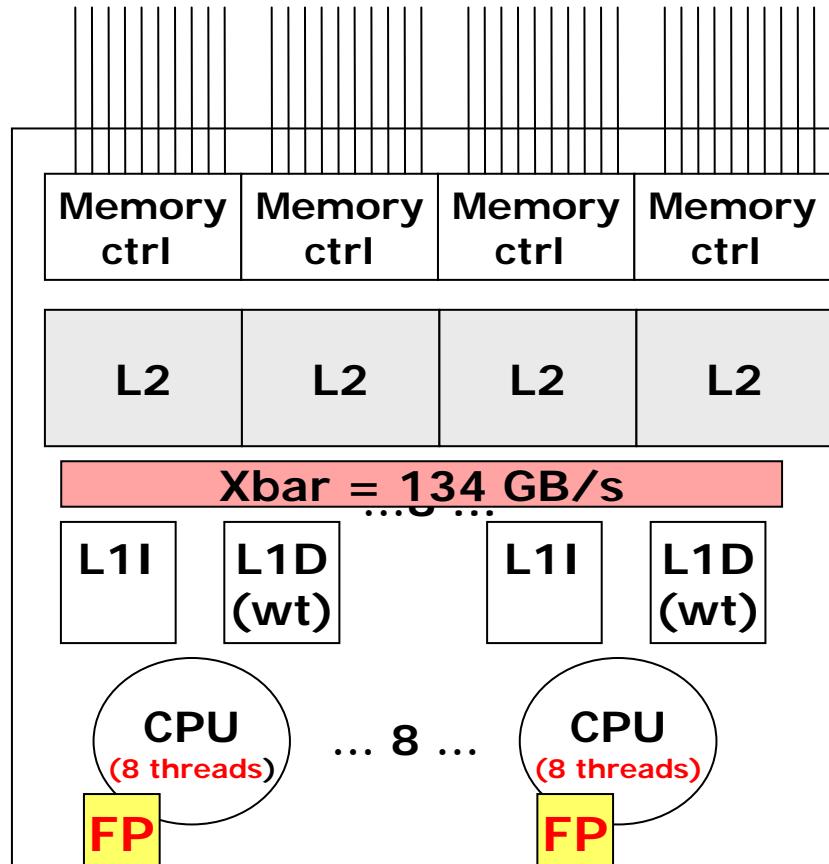


DARK
2009

Now: Victoria's falls: 16 core with 16 threads each

Niagara 2

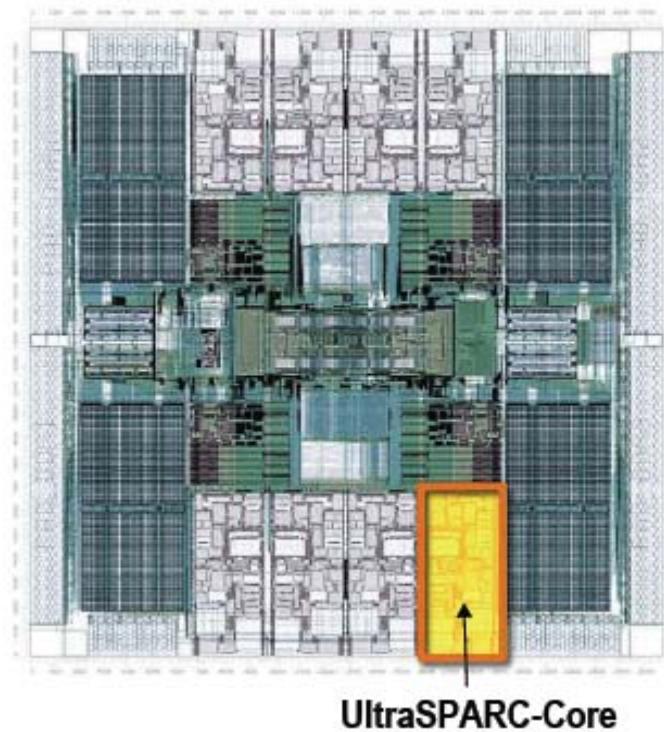
$4 \times \text{DDR-2} = 25\text{GB/s} (!)$



DARK
2009

Now: Niagra 3, 16 cores,

Niagara Chip

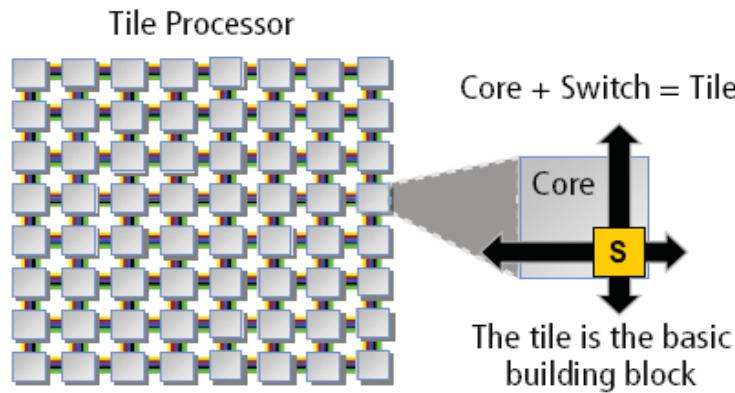


DARK
2009

Sun Microsystems

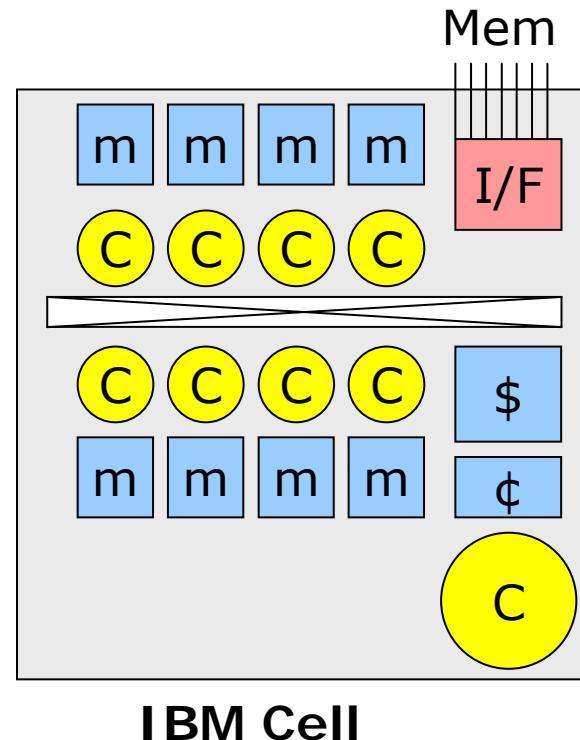


TILER A Architecture



- 64 cores connected in a mesh**
- Local L1 + L2 caches**
- Shared distributed L3 cache**
- Linux + ANSI C**
- New Libraries**
- New IDE**
- Stream computing**
- ...

IBM Cell Processor



DARK
2009

So-called accelerators

- Sits on the IO bus (!!)
- GP Graphics processors, aka GPGPU?
[e.g. NVIDIA, AMD/ATI]
- Specialized accelerators?
[e.g., FPGA/Mitrionics, ASIC/ClearSpeed]
- Specialized languages for the above?
[CUDA, Ct, Rapid Mind, Open-CL, ...]

So-called accelerators

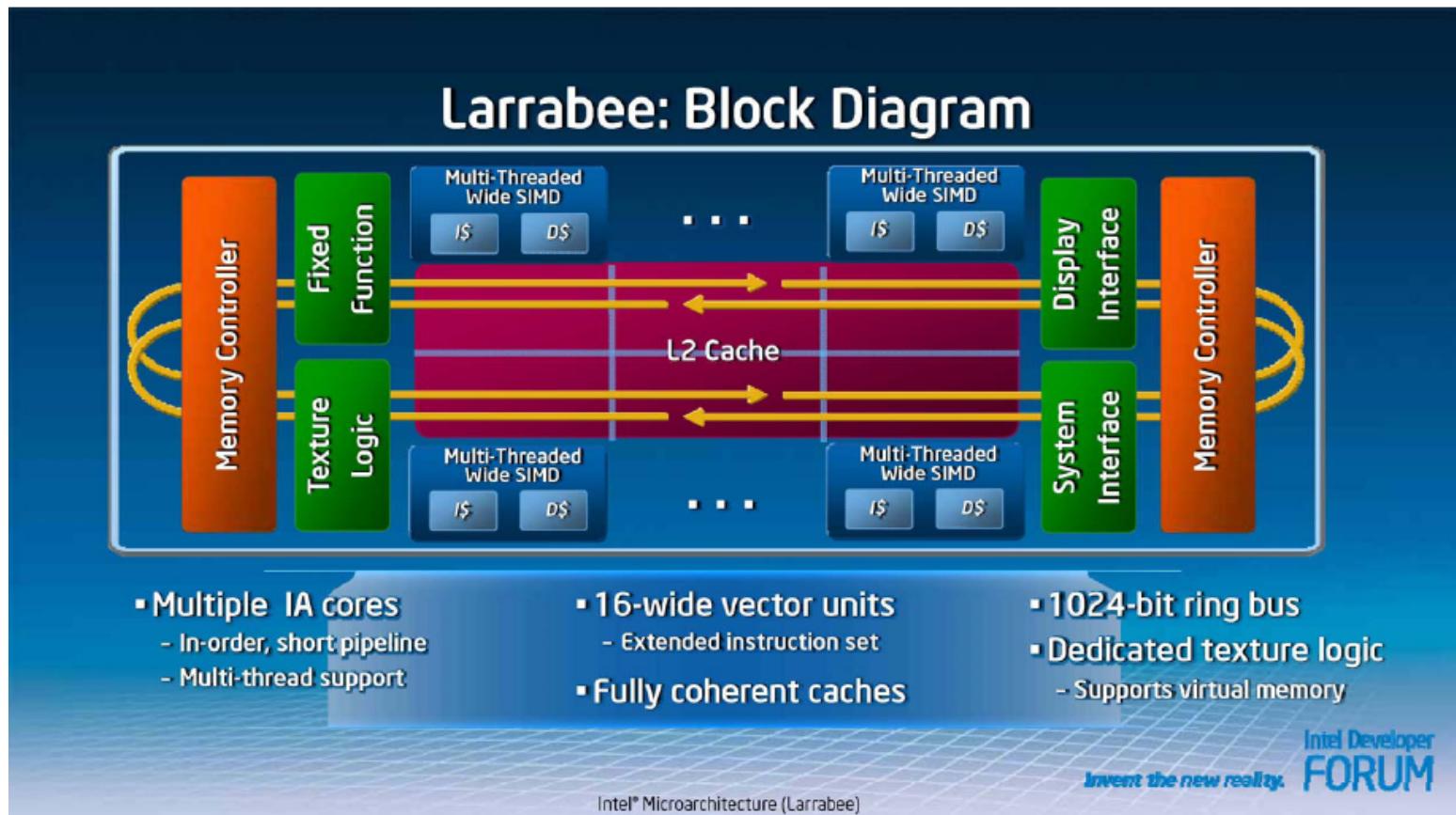
- Sits on the IO bus (!!)
 - My view: Not very general purpose yet
- GP G [e.g.
 - Fits well for a few VERY IMPORTANT app domains!
 - Limited applicability?
 - Programmer productivity?
- Speci [e.g.,
 - Application life time?
 - Better architectures around the corner
 - Fermi by NVIDIA
 - Larrabee by Intel
- Speci [CUDA]
 - SW standards emerging, e.g. OpenCL

Intel Larrabee 2010 (??)

32 single in-order cores

4 threads /core

16 wide SIMD (single precision)/8 wide (double)



Rumors on the net. Not based of actual facts.



Multicore CPUs:

Ageia PhysX, a multi-core physics processing unit.

AmbriC Am2045, a 336-core Massively Parallel Processor Array (MPPA)

AMD

- Athlon 64, Athlon 64 FX and Athlon 64 X2 family, dual-core desktop processors.
- Opteron, dual- and quad-core server/workstation processors.
- Phenom, triple- and quad-core desktop processors.
- Sempron X2, dual-core entry level processors.
- Turion 64 X2, dual-core laptop processors.
- Radeon and FireStream multi-core GPU/GPGPU (10 cores, 16 5-issue wide superscalar stream processors per core)

ARM MPCore is a fully synthesizable multicore container for ARM9 and ARM11 processor cores, intended for high-performance embedded and entertainment applications.

Azul Systems Vega 2, a 48-core processor.

Broadcom SiByte SB1250, SB1255 and SB1455.

Cradle Technologies CT3400 and CT3600, both multi-core DSPs.

Cavium Networks Octeon, a 16-core MIPS MPU.

HP PA-8800 and PA-8900, dual core PA-RISC processors.

IBM

- POWER4, the world's first dual-core processor, released in 2001.
- POWER5, a dual-core processor, released in 2004.
- POWER6, a dual-core processor, released in 2007.
- PowerPC 970MP, a dual-core processor, used in the Apple Power Mac G5.
- Xenon, a triple-core, SMT-capable, PowerPC microprocessor used in the Microsoft Xbox 360 game console.

IBM, Sony, and Toshiba Cell processor, a nine-core processor with one general purpose PowerPC core and eight specialized SPUs (Synergistic Processing Unit) optimized for vector operations used in the Sony PlayStation 3.

Infineon Danube, a dual-core, MIPS-based, home gateway processor.

Intel

- Celeron Dual Core, the first dual-core processor for the budget/entry-level market.
- Core Duo, a dual-core processor.
- Core 2 Duo, a dual-core processor.
- Core 2 Quad, a quad-core processor.
- Core i7, a quad-core processor, the successor of the Core 2 Duo and the Core 2 Quad.
- Itanium 2, a dual-core processor.
- Pentium D, a dual-core processor.
- Teraflops Research Chip (Polaris), an 3.16 GHz, 80-core processor prototype, which the company says will be released within the next five years[6].
- Xeon dual-, quad- and hexa-core processors.

IntellaSys seaForth24, a 24-core processor.

Nvidia

- GeForce 9 multi-core GPU (8 cores, 16 scalar stream processors per core)
- GeForce 200 multi-core GPU (10 cores, 24 scalar stream processors per core)
- Tesla multi-core GPGPU (8 cores, 16 scalar stream processors per core)

Parallax Propeller P8X32, an eight-core microcontroller.

picoChip PC200 series 200-300 cores per device for DSP & wireless

Rapport Kilocore KC256, a 257-core microcontroller with a PowerPC core and 256 8-bit "processing elements".

Raza Microelectronics XLR, an eight-core MIPS MPU

Sun Microsystems

- UltraSPARC IV and UltraSPARC IV+, dual-core processors.
- UltraSPARC II, an eight-core, 32-thread processor.

Design Issues for Multicores

Erik Hagersten
Uppsala University
Sweden

CMP bottlenecks/points of optimization

- Performance per Watt?
- Performance per memory byte?
- Performance per bandwidth?
- Performance per \$?
- ...

- How large fraction of a CMP system cost is the CPU chip?
- Should the execution (MIPS/FLOPS) be viewed as a scarce resource?

DRAM issues

"Rock will have more than 1000 memory chips per Rock chip" [M. Trembley, Sun Fellow at ICS 2006]

→ Memory will dominate cost?

Fewer "open pages" accessed due to interleaving of several threads

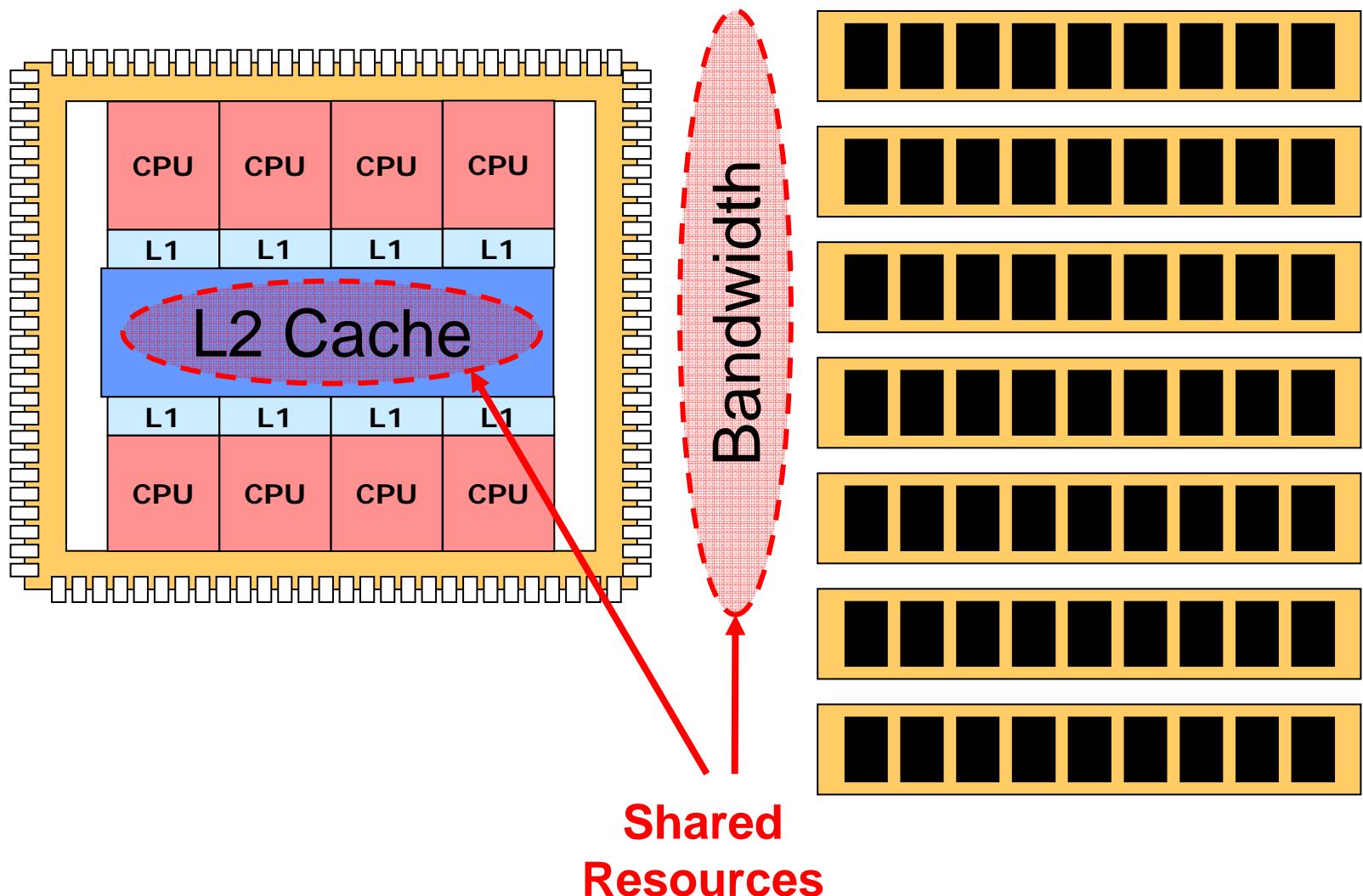
Far memory → long latency & low per-pin BW

Pushing dense memory technology/packaging?

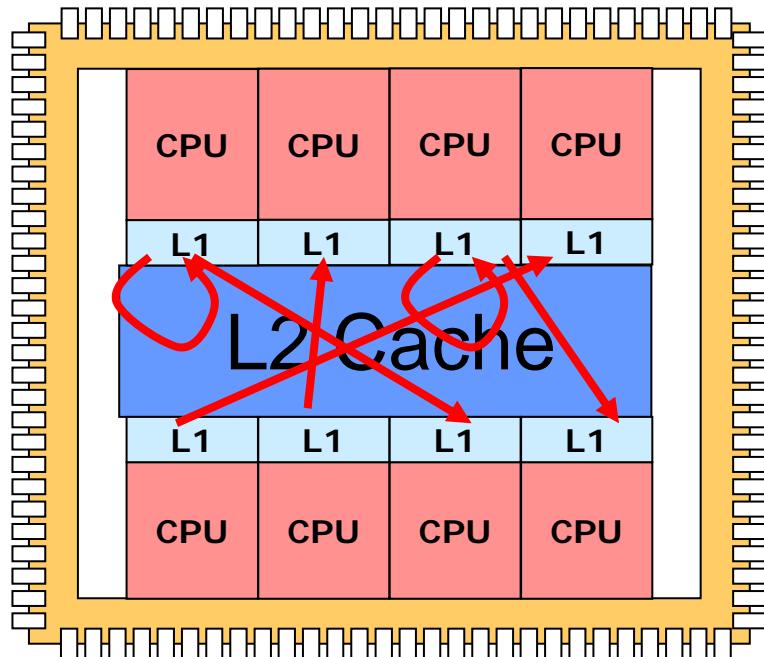
Bandwidth issues

- #pins is a scarce resource!
 - every pin should run at maximum speed
-
- ➔ external memory controllers?
 - ➔ off-chip cache?
 - ➔ is there room for multi-CMP?
 - ➔ is this maybe a case for multi-CMP?

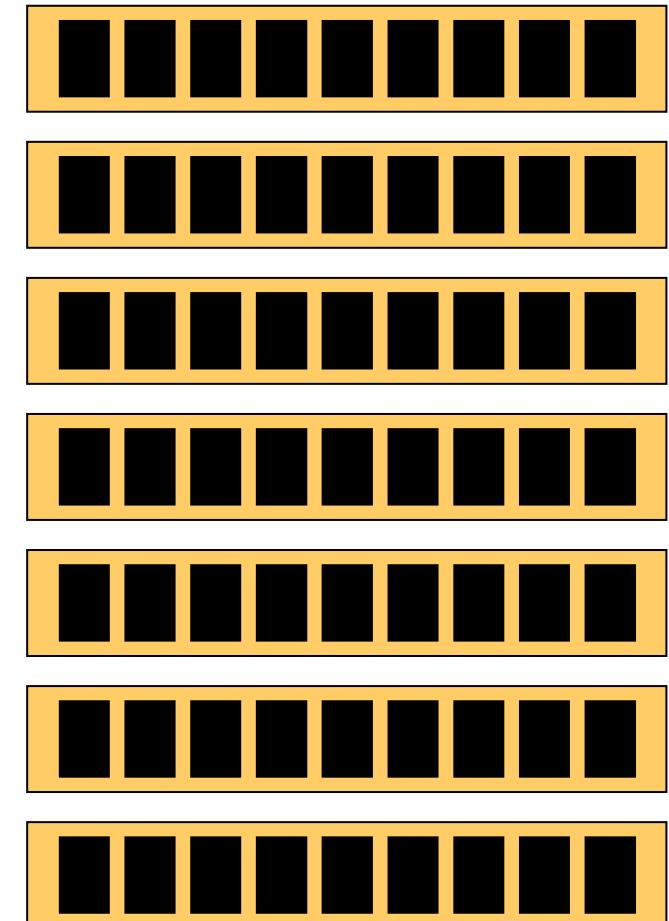
Shared Bottlenecks



Thread Interaction

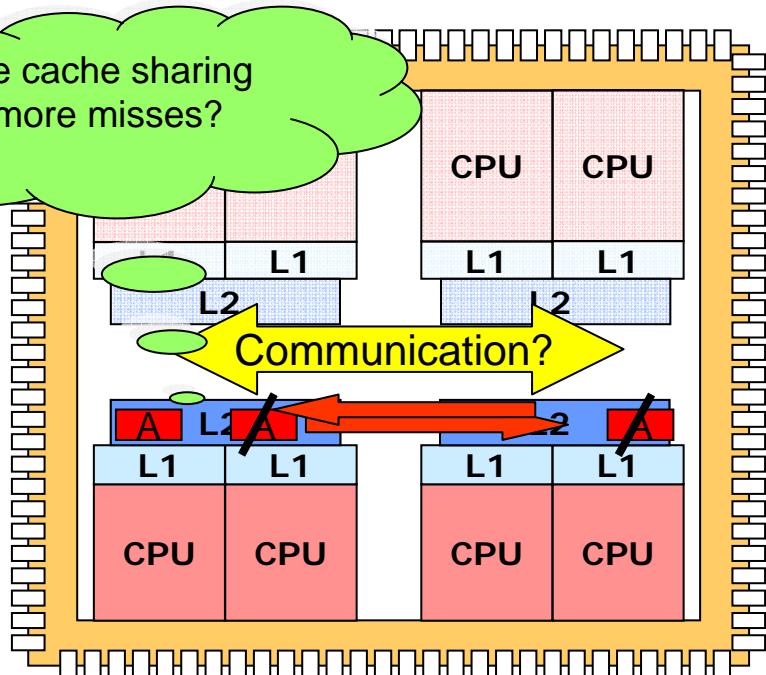


- Coherence traffic
- Communication utilization
- Load imbalance
- Synchronization
- False sharing
- ...



Example: Thread Interaction (True Communication)

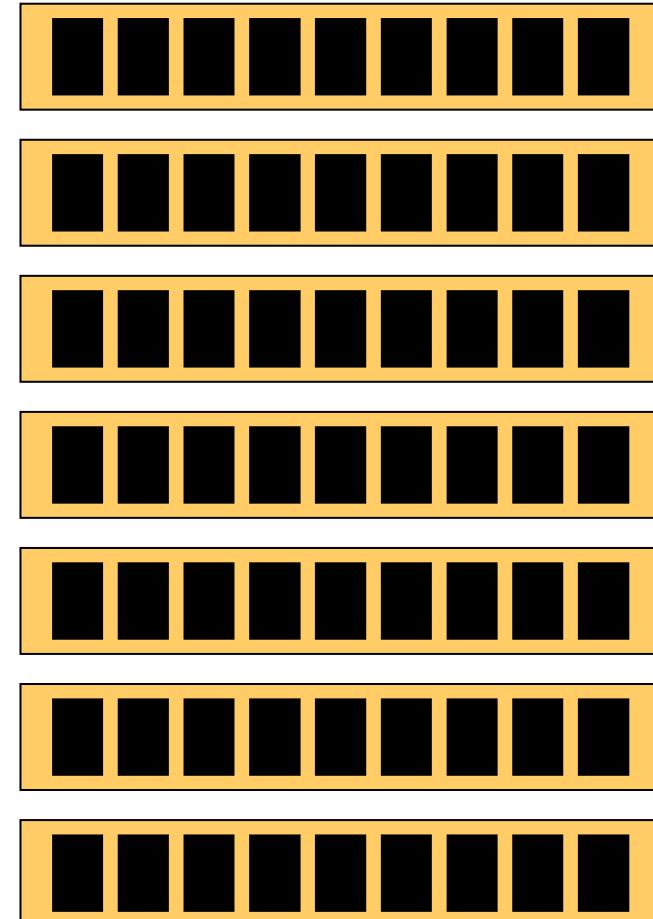
More cache sharing
→ more misses?



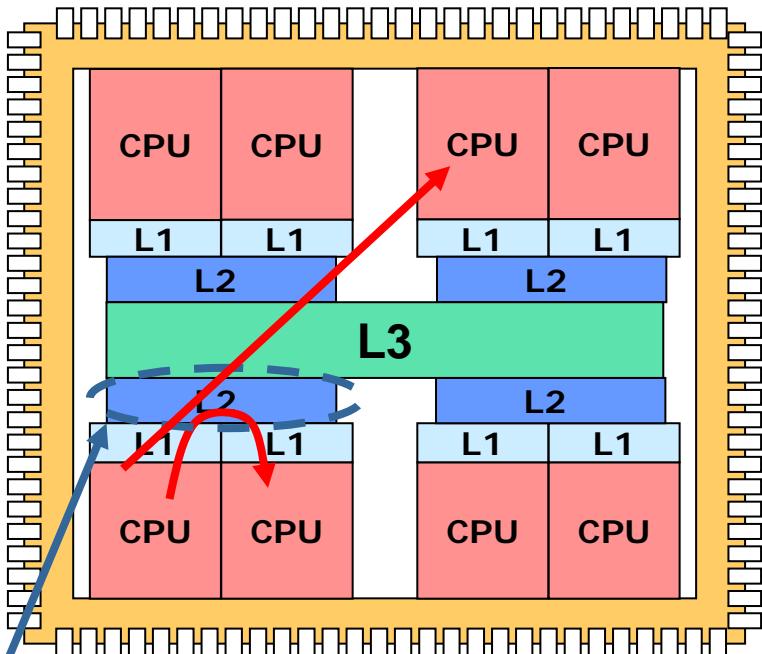
A := 1

A := 2

A := 3



Multicore Challenges: Non-uniformity Communication



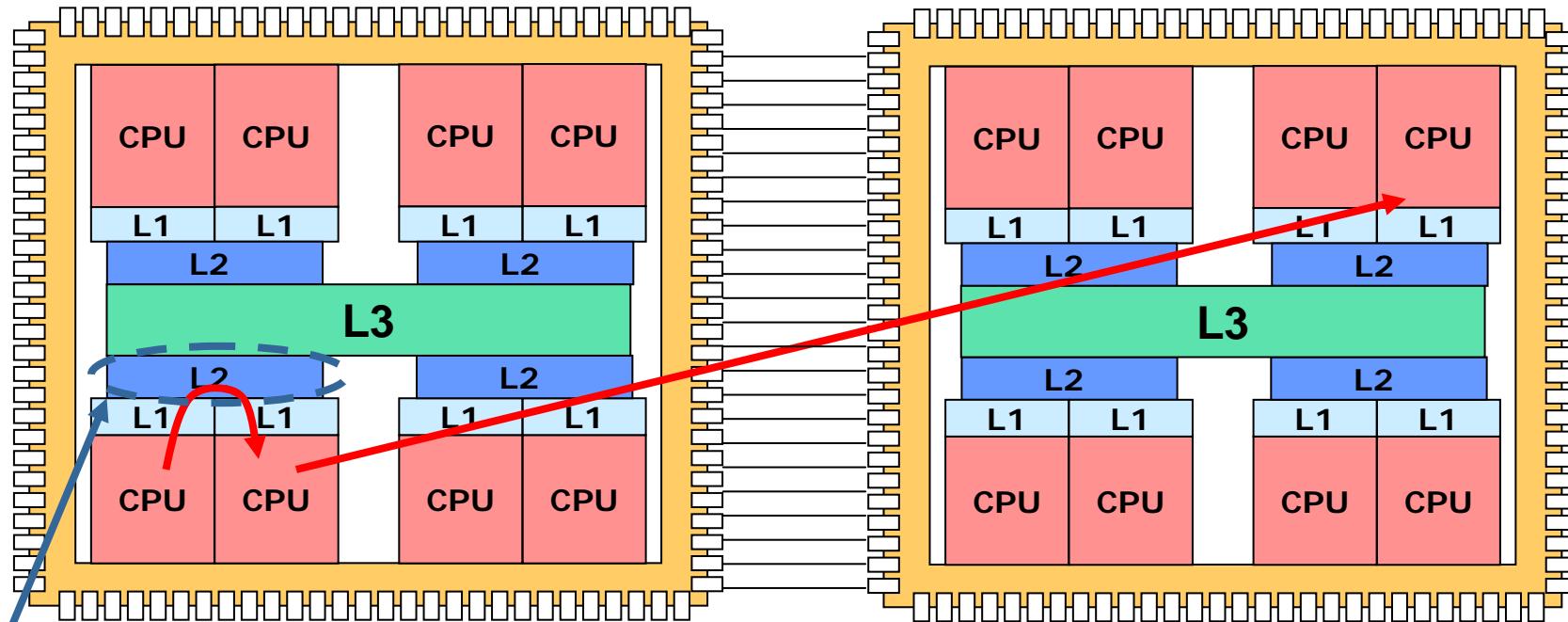
L2 sharing
(here: pair-wise)

DARK
2009



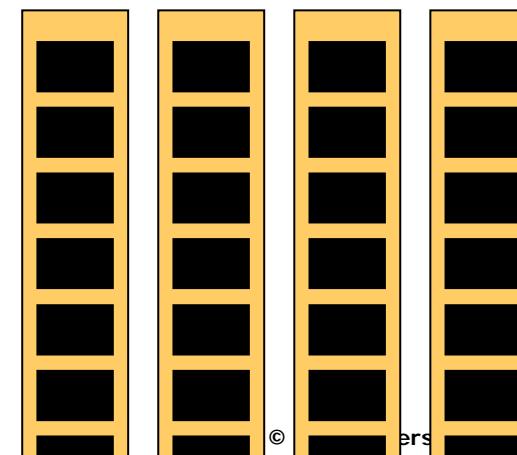
UPPSALA
UNIVERSITET

Multicore Challenges(Multisocket) Non-uniformity Communication



L2 sharing
(here: pair-wise)

DARK
2009



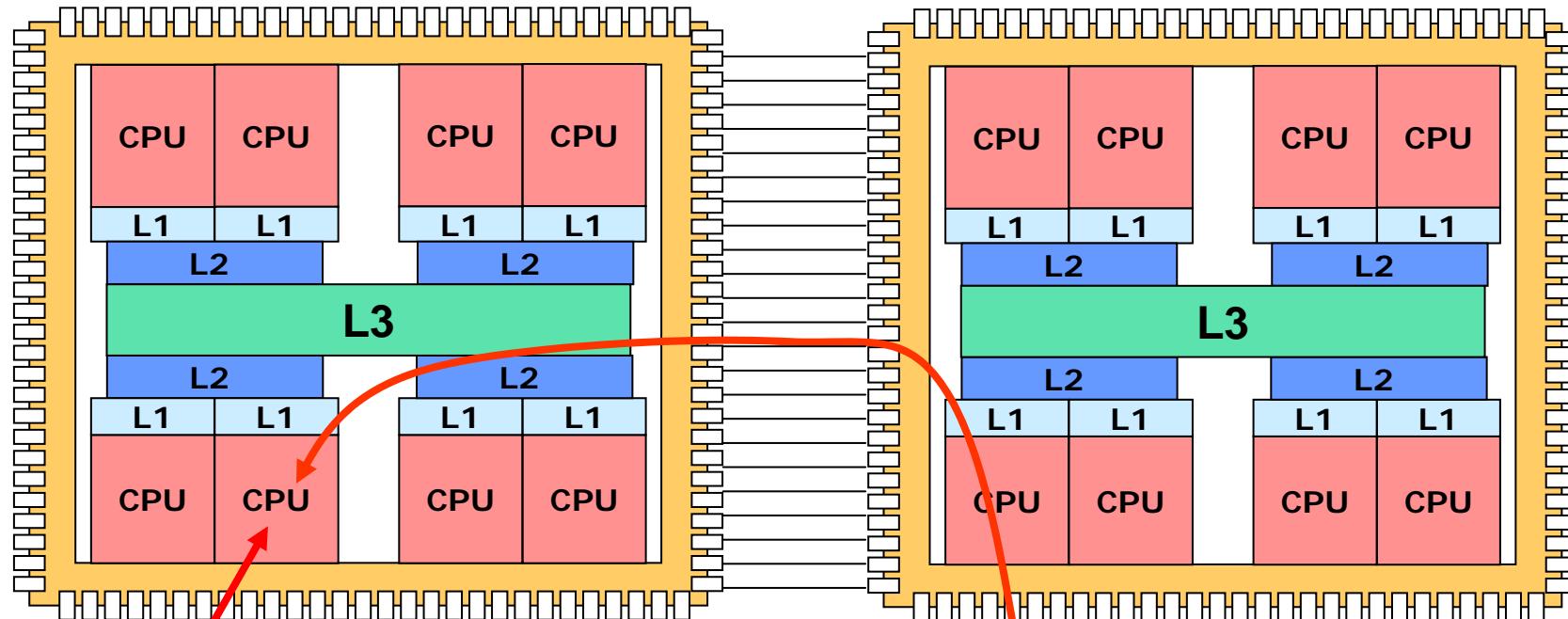


UPPSALA
UNIVERSITET

DARK
2009

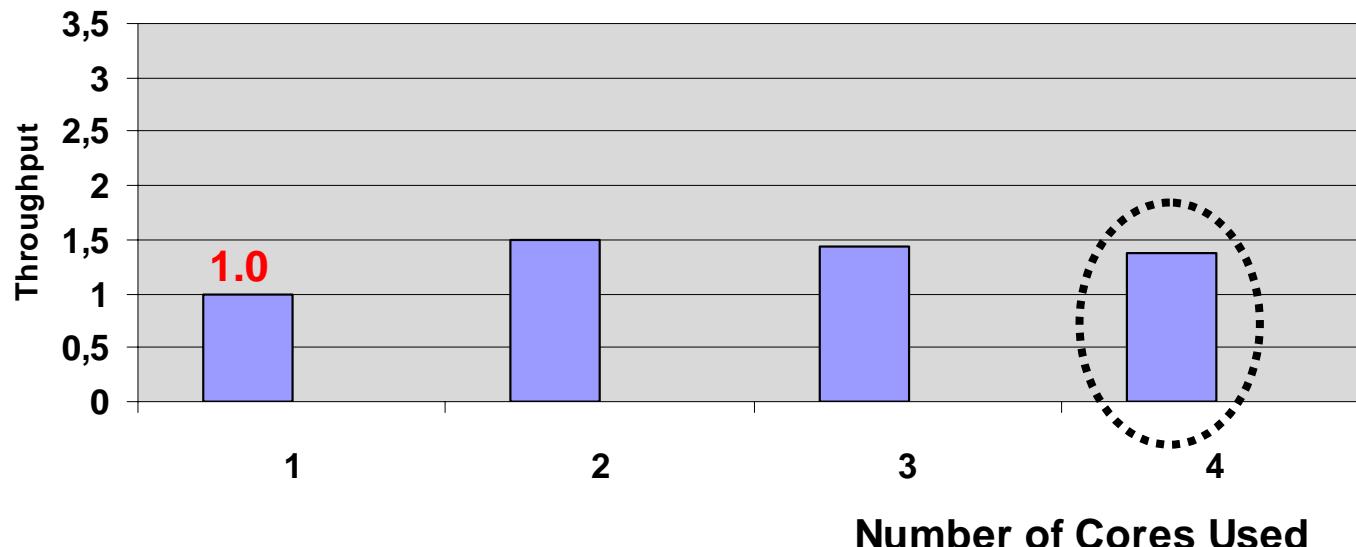
Multicore Challenges (Multisocket)

Non-uniformity Memory



Example: Poor Throughput Scaling!

Example: 470.lbm



Throughput (as defined by SPEC):

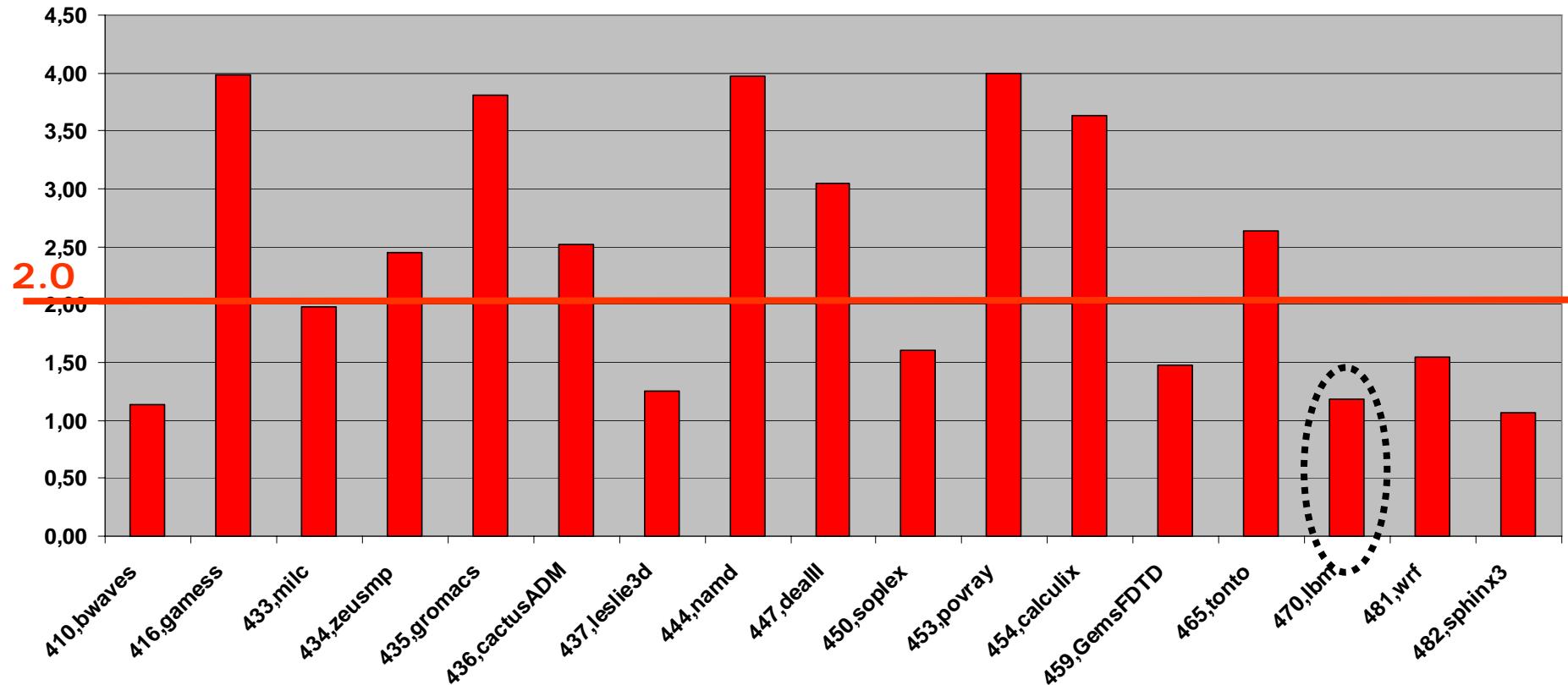
Amount of work performed per time unit when several instances of the application is executed simultaneously.

Our TP study: compare TP improvement when you go from 1 core to 4 cores



Throughput Scaling, More Apps

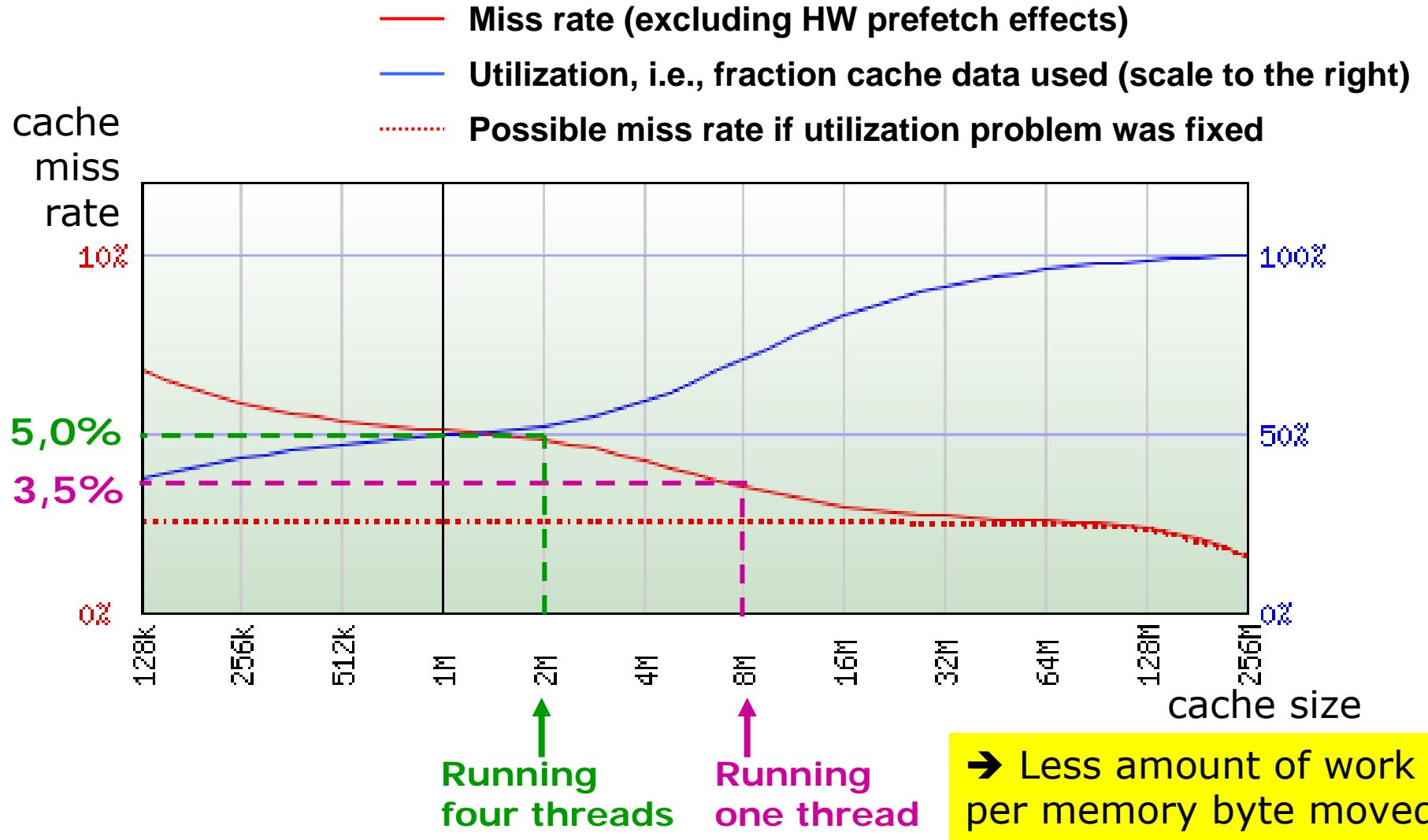
SPEC CPU 20006 FP Throughput improvements on 4 cores



DARK
2009

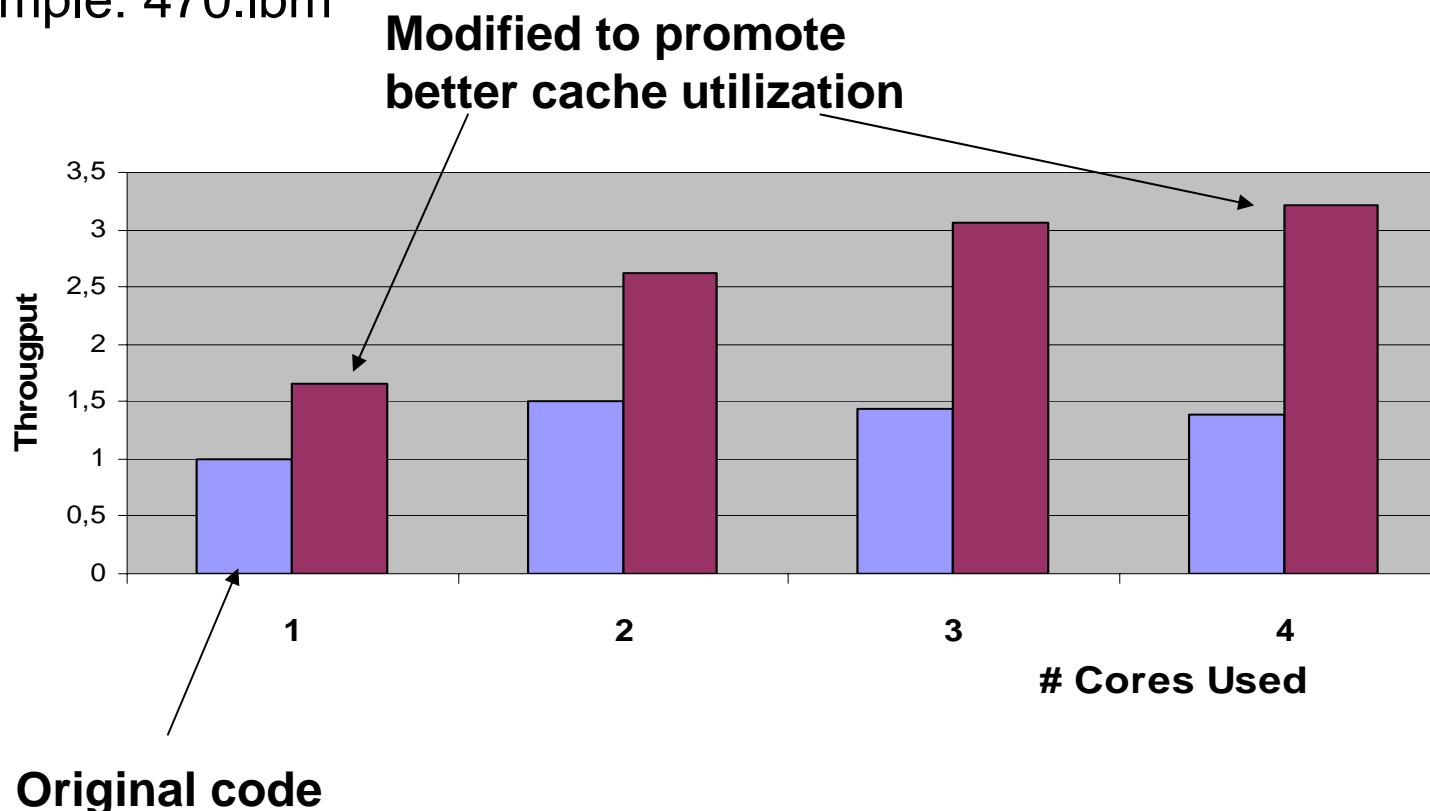
Intel X5365 3GHz, 1333 MHz FSB, 8MB L2.
(Based on data from the SPEC web)

Nerd Curve: 470.Ibm

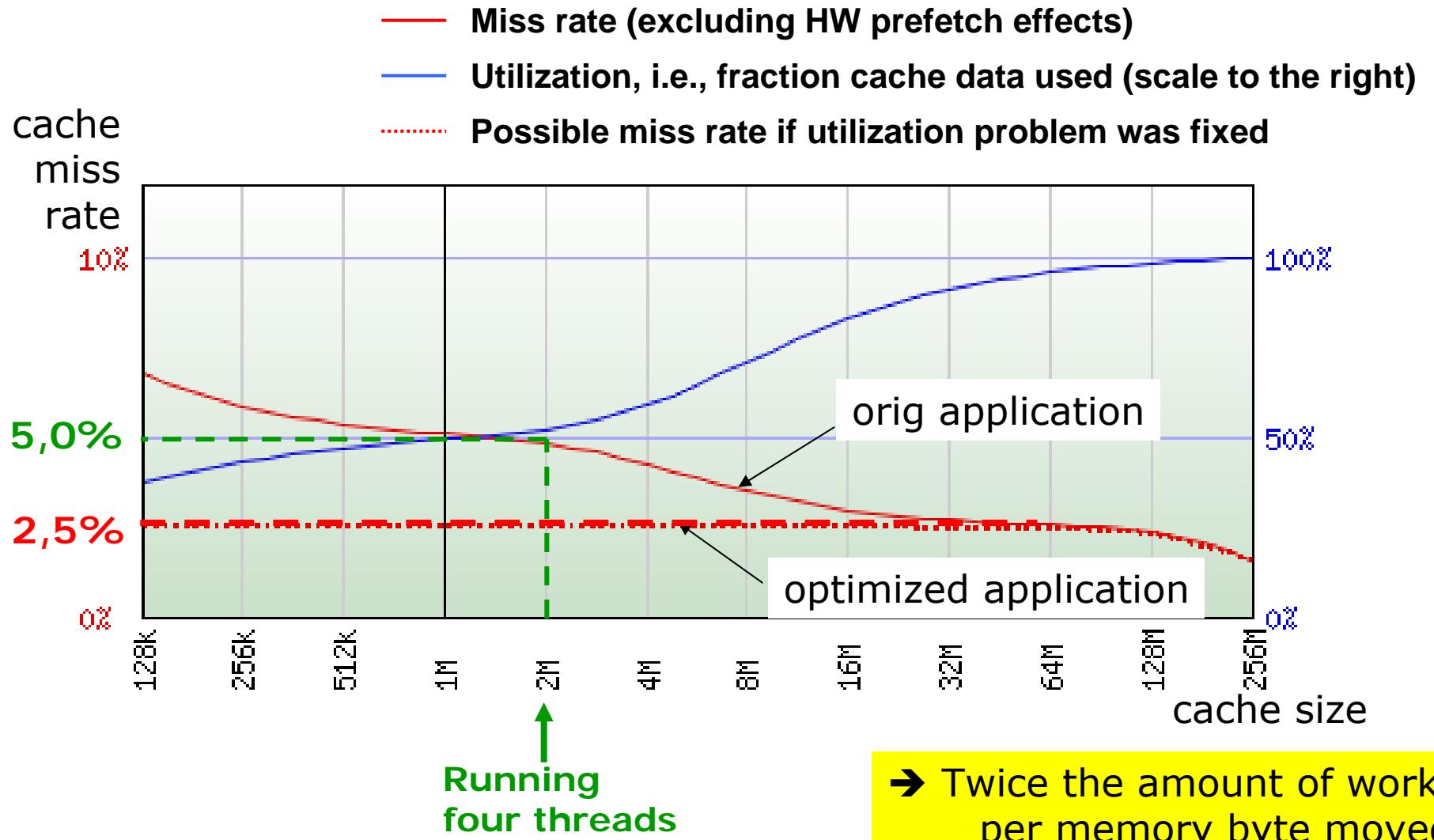


→ Better Memory Usage!

Example: 470.lbm



Nerd Curve (again)



More transistors → More Threads

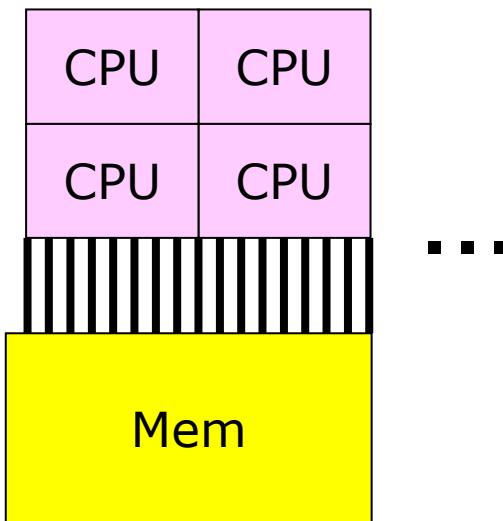
■ Warning:

- * # transistors grows exponentially
 - # threads can grow exponentially
- * Can memory BW keep up?

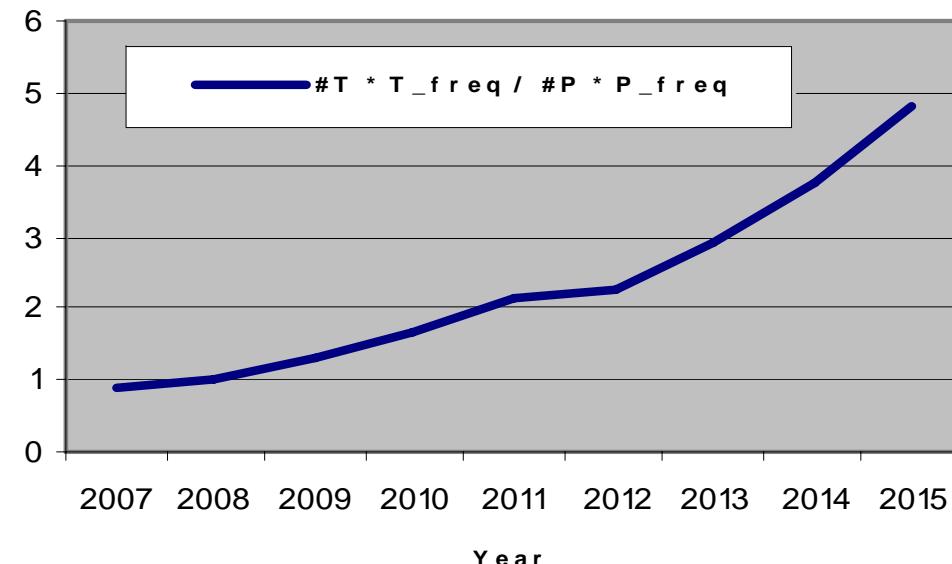
DARK
2009

BW in the Future?

#Cores ~ #Transistors



Computation vs Bandwidth



Source: International Technology Roadmap for Semiconductors (ITRS)

See also: Karlsson et al. *Conserving Memory Bandwidth in Chip Multiprocessors with Runahead Execution*. IPDPS March 2007.

See also IDC presentation from ISC June 2007

Capacity or Capability Computing?

Capacity? (\approx several sequential jobs)

or

Capability? (\approx one parallel job)

Issues:

- ✿ Memory requirement?
- ✿ Sharing in cache?
- ✿ Memory bandwidth requirement?

Memory: the major cost of a CMP system!

How do we utilize it the best?

- ✿ Once the workingset is in memory, work like crazy!

➔ Capability computing suits CMPs the best
(in general)

Fat or narrow cores?

- Fat:
 - Fewer cores but...
 - wide issue?
 - O-O-O?
- Narrow: More cores but...
 - narrow issue?
 - in-order?
 - have you ever heard of Amdahl?
 - SMT, run-ahead, execute-ahead ... to cure shortcomings?

Read:

Maximizing CMP Throughput with Mediocre Cores

Davis, Laudon and Olukotun, PACT 2006

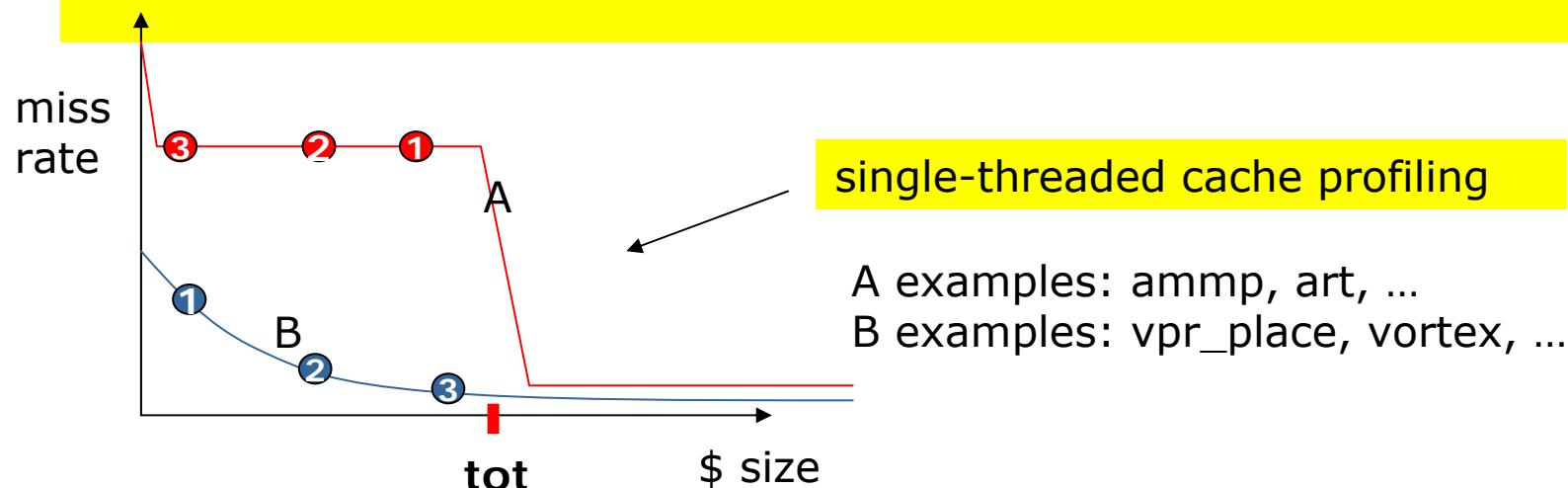
Cores vs. caches

- Depends on your target applications...
- Niagara's answer: go for cores
 - ✿ In-order 5-stage pipeline
 - ✿ 8 cores a' 4 SMT threads each → 32 threads,
 - ✿ 3MB shared L2 cache (96 kB/thread)
 - ✿ SMT to hide memory latency
 - ✿ Memory bandwidth: 25 GB/s
 - ✿ Will this approach scale with technology?
- Others: go for cache
 - ✿ 2-4 cores for now

Cache Interference in Shared Cache

■ Cache sharing strategies:

1. Fight it out!
2. Fair share: 50% of the cache each
3. Maximize throughput: who will benefit the most?



Read:

STATSHARE: A Statistical Model for Managing Cache Share via Decay
Pavlos Petoumenos et al in MOBS workshop ISCA 2006

Predicting the inter-thread cache contention on a CMP
Chandra et al in HPCA 2005

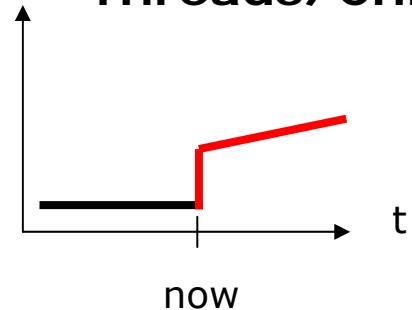
Hiding Memory Latency

- O-O-O
- HW prefetching
- SMT
- Run-ahead/Execute-ahead

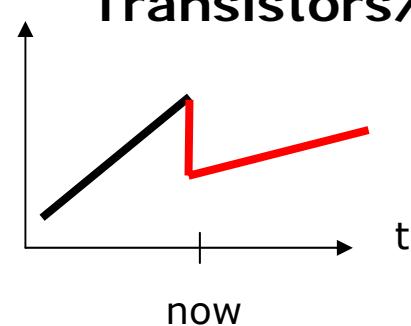


Trends (my guess!)

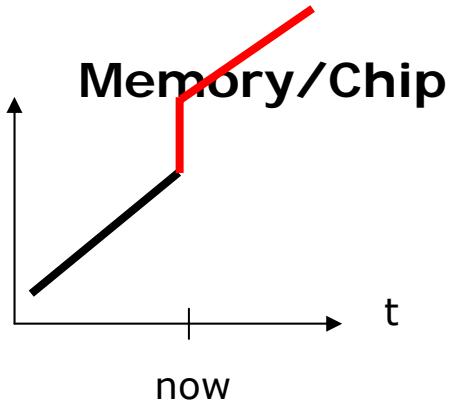
Threads/Chip



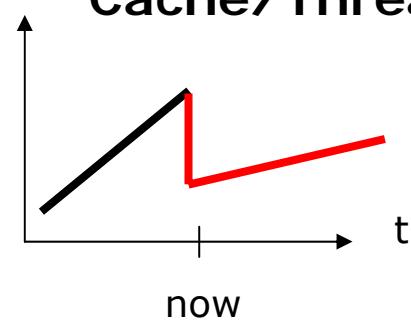
Transistors/Thread



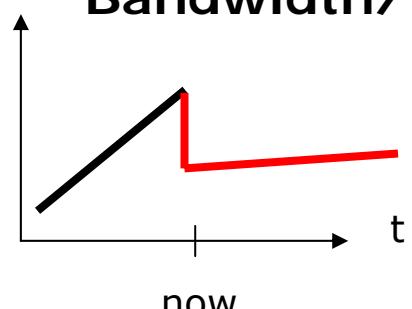
Memory/Chip



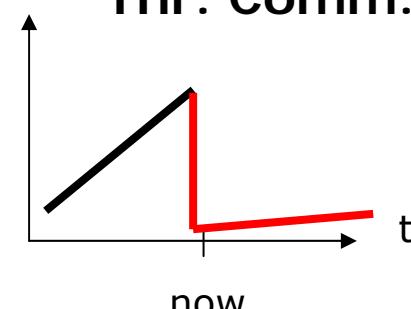
Cache/Thread



Bandwidth/Thread



Thr. Comm. Cost (temporal)



Questions for the Future

- What applications?
- How to get parallelism and data locality?
- Will funky languages see a renascence?
- Will automatic parallelizing return?
- Are we buying:
 - ✿ compute power,
 - ✿ memory capacity, or
 - ✿ memory bandwidth?
- Will the CPU market diverge into desktop/capacity CPUs again?
- How to debug? ...
- A non-question: will it happen?