Lecture 9

Computations on graphics processors

Ingemar Ragnemalm

Information Coding, ISY

Did you find it amazing to run on 8 cores in a single desktop?

Did you find it amazing to run on 8 cores in a single desktop?

How about doing that with 1300+ cores?

This lecture:

Plan for this part of the course

GPU evolution

GPU architecture

A first intro to general computing solutions with GPUs

My part of the course:

5 lectures

1 lesson

3 labs

Lectures:

9. GPU evolution and architecture

10. Intro to CUDA

11. CUDA memory, threads, synchonization

12. More CUDA, sorting on GPU

13. Intro to OpenCL. Computing with shaders

Labs:

4. CUDA

5. Sorting with CUDA

6. OpenCL, image filter

No lab reports, demonstrations in the lab

Literature for this part

Primary source:
CUDA on-line manual

Recommended extra:
CUDA by example (Sanders & Kandrot)

Hand-outs

Lecture material

Questions

1. How can a GPU be much faster than a CPU?

2. Why is the G80 so much faster than the previous GPUs (e.g. 7000 series)?

3. A texturing unit provides access to texture memory. What more is it than just another memory?

4. Suggest two major differences in the Fermi architecture that will make a difference from the G80/G92/GT200

The decline of CPU evolution

Three ”walls”:

The decline of CPU evolution

Three ”walls”:

Tenessee Waltz

Max Wall

Wall-E

The decline of CPU evolution

Three ”walls”:

The decline of CPU evolution

Three ”walls”:

Power wall

Memory wall

ILP wall

The decline of CPU evolution

Three ”walls”:

Power wall

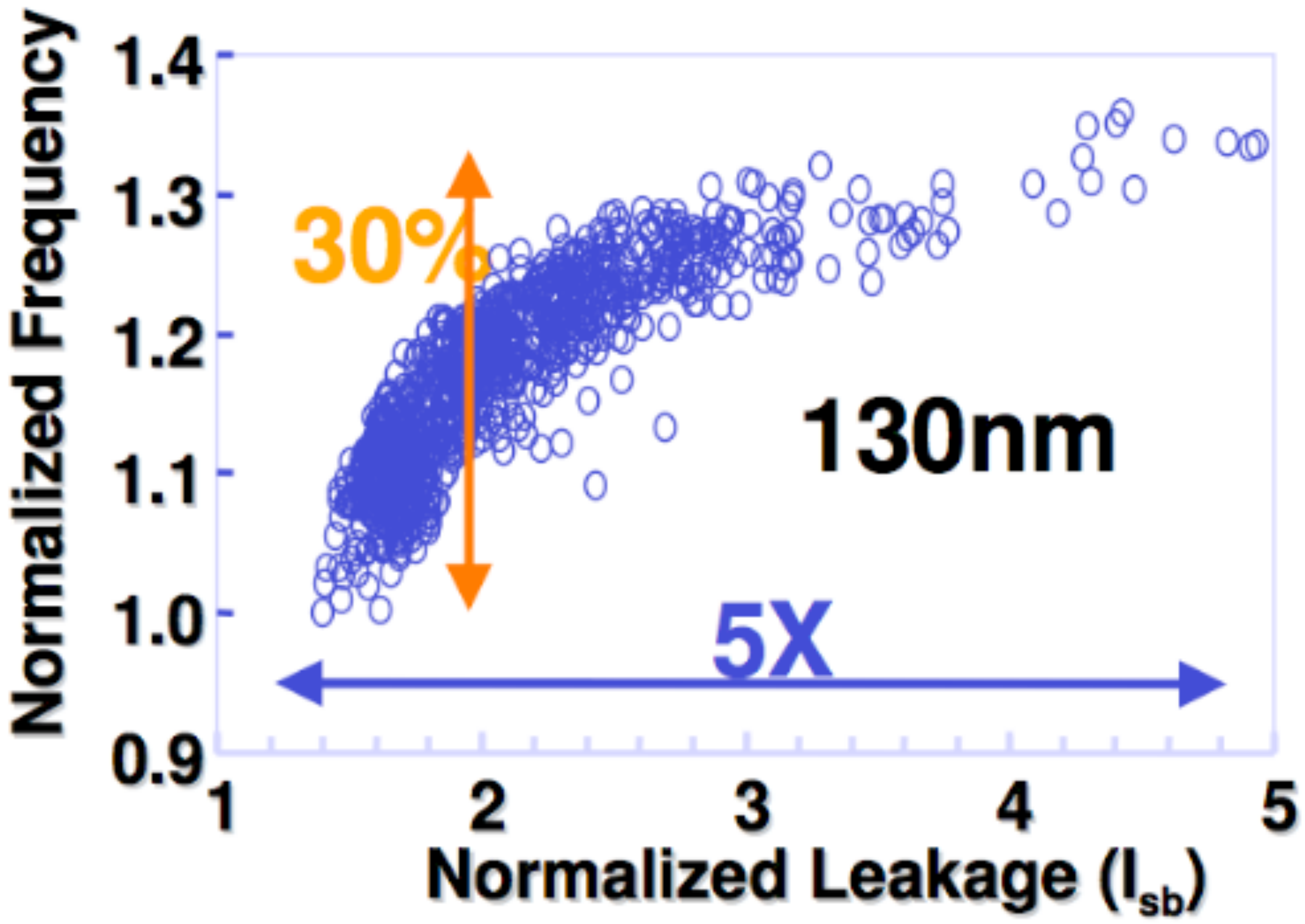
Memory wall

ILP wall

- **Clock frequency can no longer go up**
- **The memory architecture is insufficient**
- **Attempts to parallelize have failed**

Power wall

13% higher frequency = 73% more (almost double) double power consumption!



Power wall

Reverse reasoning: Lower frequency a little, win much power.

Replace one high-frequency CPU with two slightly slower - for the same cost!

Works nicely for two CPUs.

Intel promises 80 cores in a few years

BUT

this will run into the ”memory wall”

Memory wall

Already, the memory is slower than the CPU.

With more and more CPUs fighting for accessing the same RAM and caches, efficiency will degrade!

Memory bandwidth helps - if we can get it.

ILP wall

Instruction level parallelism

Writing parallel code is complicated.

Many problems are sequential by nature - or traditionally expressed as such.

ILP wall

Instruction level parallelism

Writing parallel code is complicated.

Many problems are sequential by nature - or traditionally expressed as such.

Solutions:

- **Explore algorithms in search of parallel solutions**
- **Learn how to code in parallel**
- **New programming paradigms, not optimizing for the programmer but for the computer!**