Energy and Parallel Computing

Mark Greenstreet

CpSc 418 – October 12, 2018

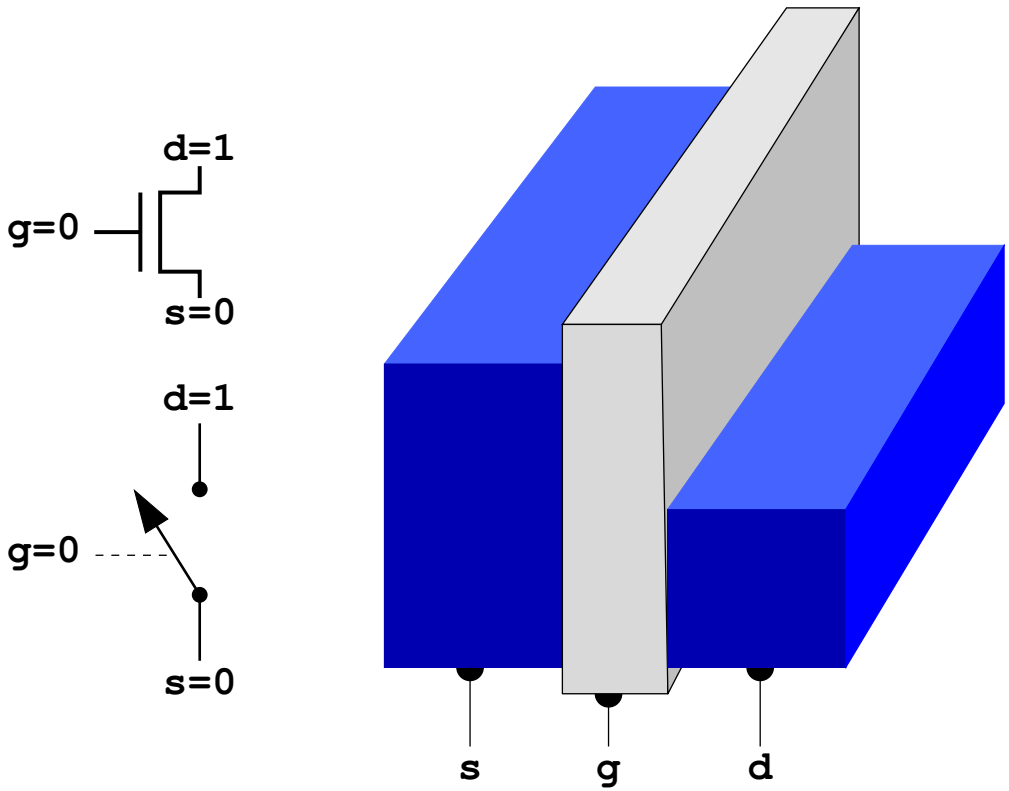
Objectives

- Understand that parallel algorithms can use less energy than their sequential counterparts.
- Familiar with the technology scaling trends that lead to this.
  - Where does Moore's Law come from?
  - What is Dennard scaling (was it first proposed by Hoeneisen & Mead?)
  - What are energy-time trade-offs for real-world computers
- Aware of how this is likely to impact computing technology in the next decade or so.
  - Buying computation by the kilowatt-hour
  - What are the opportunities
    - ★ Domain specific architectures and languages.
  - Where are exponential improvements in technology happening now
  - What are energy-time trade-offs for real-world computers
- Aware of how this is likely to impact computing technology in the next decade or so.

Outline

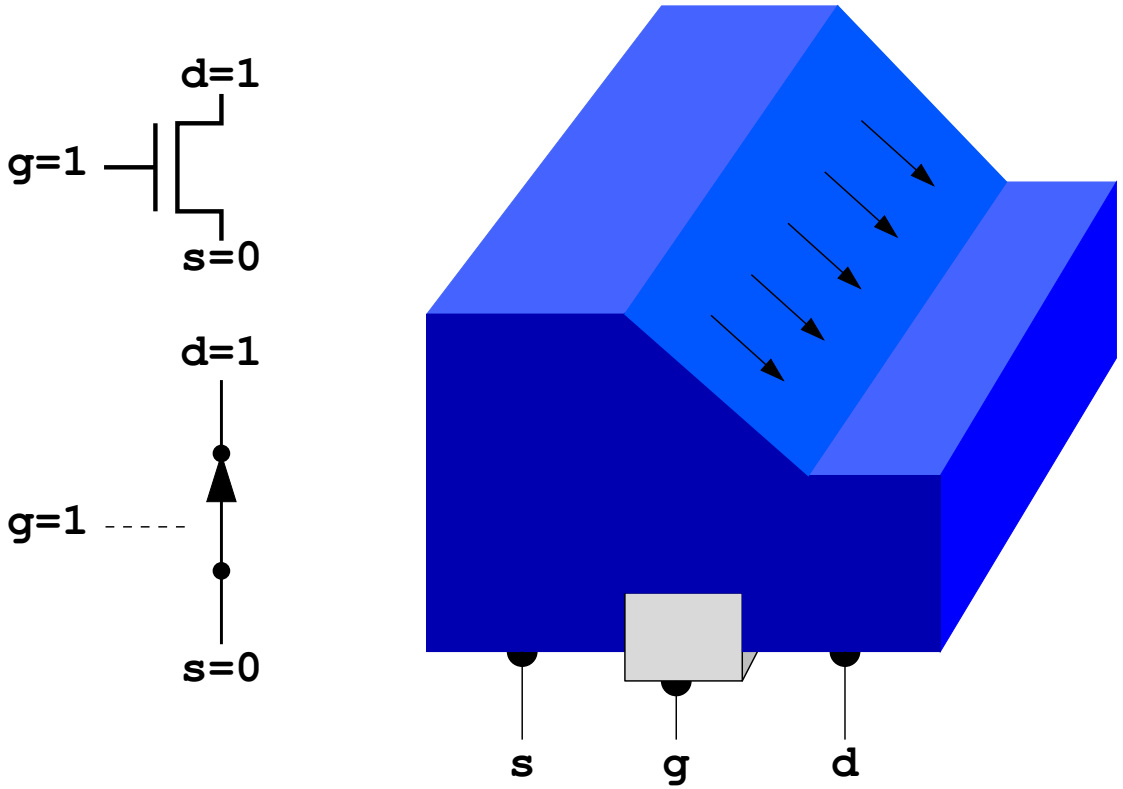
- From silicon atoms to computers.
  - Dam transistors
  - How to make a computer
  - Classical scaling, and why it no longer applies.
- Energy performance trade-offs in real computers.
  - Going fast takes lots of energy.
  - Many slow parallel tasks can be more energy efficient than one, fast sequential task.
  - The case for dedicated co-processors.
- Guessing about the future
  - Optical technology has a bright future.
  - Dedicated co-processors means domain-specific architectures and programming models.

Dam Electronics



- When the gate, `g`, is low, it has a negative charge that repels electrons. This is like the dam being high, and no water flows from the source, `s`, to the drain, `d`. The switch is "open" – it makes no connection.

Dam Electronics

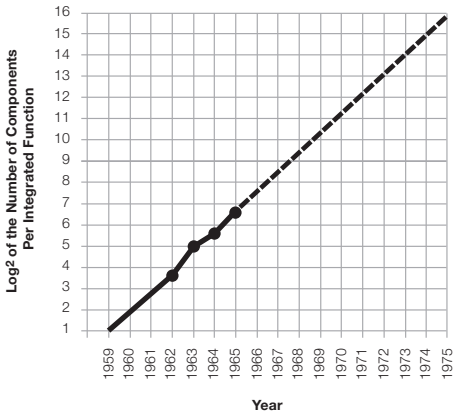


- When the gate is high, it has a postive charge that attracts electrons. This is like the dam being low, and water flows from the source to the drain. The switch is “closed” – it makes a connection.
- Once we have switches, we can build gates, registers, and all the other wonders of the digital world.

Manufacturing Integrated Circuits

- Transistors and wires are made through a sequence of chemical processing steps.
  - Start with a flat, thin, sheet of silicon, a wafer.
  - Photographically print a pattern onto the wafer.
  - Use chemical processes to change electrical properties of the silicon, deposit metal for wires or glass as an insulator, etch the metal to make wires, etc.
- By making tranistors smaller:
  - We can put more processor cores, more memory, etc. onto the chip.
  - This creates a profit motive for making smaller and smaller transistors.
  - Today's chips have transistors where the "dam" is 14 nanometers across.
    - ★ To compare, a hair is about 20 microns (if you're blond) to about 100 microns (if your hair is black) in diameter. 1 micron is one micro-meter.
    - ★ That means we can fit about 1000 tranistors across the diameter of one hair.
- By making tranistors smaller:

Moore's Law



- Moore's Law (original): the number of transistors on a chip will double every year from 1965 through 1975.
- Justification
  - Moore took four data points and found they could be fit reasonable well with a line on a semi-log plot. ☺
  - More seriously, Moore observed that
    - ★ Putting more transistors onto a chip allowed you do build new kinds of electronic devices.
    - ★ There would be a large market for these devices.
    - ★ The profits made from selling the chips would allow semiconductor companies to improve their manufacturing processes.
    - ★ Transistors would shrink a lot, chips would get bigger.
    - ★ Moore extrapolated until 1975 because the various technical challenges seemed solvable given plausible estimates of sales an profit.

Moore's Law – Beyond 1975

- Moore's law has enjoyed many extensions as key manufacturing issues were solved.
- The rate has gradually slowed from doubling every year to doubling every 3 or 4 years.
- Power blocked clock frequency from scaling with transistor size from roughly 2003 and beyond.
- There is a limit to scaling
  - Current products in transistors with 14nm channel length (the thickness of the "dam").
    nm $=$ nanometer $= 10^{-9}$ meter.
  - Chip designer working on designs with 7nm channel length.
  - Shrinking to 5nm or 3.5nm looks really difficult.
  - The spacing of silicon atoms in a silicon crystal is around 0.3nm.

Denard Scaling – Dams

What happens if we scale transistor dimensions and operating voltage by a factor λ?

- The resevoirs get smaller.
  - The resevoir's height, V, goes as λ.
  - The resevoir's volume, C, goes as λ^2.
  - The stored energy in the resevoir goes as λ^3.
- The aspect ratio of the dam, $1/R$ stays constant
  - The height difference between the source and drain, V, goes as λ.
  - The rate of flow over the dam, I, goes as λ – Ohm's law: $I = V/R$.
- The time to fill/drain the resevoir is volume/flow
  - That goes as $\lambda^2/\lambda = \lambda$.

Denard Scaling – slightly quantitative

- E.g. $\lambda = 0.5$ is shrinking everything to half its previous size.
- Gate delay scales as λ.
- Clock frequency scales as $1/\text{delay} = 1/\lambda$.
- Energy per signal transition scales as λ^3 – this is amazing!
- Power is $\frac{\text{energy}}{/\text{time}}$. Power scales as λ^2.
- Number of devices on a chip scales as λ^{-2}.
- Power density (i.e. watts per square centimeter) is constant.
- **Conclusion:** everything gets **way** better as we shrink transistors.
  - Of course, this requires very precise manufacturing, so it took many rounds of the Moore's Law positive feedback cycle to get to where we are today.

What went wrong: The Power Wall

- To disconnect the source from the drain of the transistor, the "dam" must be above the level of the upper reservoir.
- But, the reservoirs have "waves"
  - The waves are the thermal energy of the electrons.
  - To turn off a transistor, the dam needs to be about $10\times$ higher than the average wave.
  - The dam height can be at most $\sim 40\%$ of the operating voltage.
  - This sets a lower bound for operating voltage (at room temperature) of about $0.6V$.
- Voltage hasn't scaled as predicted by classical scaling since the early 1990's.
  - Chips are faster than they should be by Denard scaling. ☺
  - They are also **way** hotter. ☹

Power is the Primary Design Concern

- In the old days, processors were designed primarily for speed.
- Now, they are designed to satisfy a power requirement.
- This impacts all forms of computing:
  - mobile devices and battery life
  - desktop devices and gaming consoles are limited by cooling
  - data centers and cloud services are limited by building cooling.
    - ★ The power bill is a major part of the operating expenses for cloud services.
    - ★ Indirectly, cloud users are buying computation by the kilowatt hour.
    - ★ Although the power bill is indirect in the billing, the financial consequences are very real.

Energy time trade-offs in real life

- The tradeoff that $E \propto T^{-2}$ from the text assumes classical scaling.
  - We can't push the operating voltage as low as assumed by such scaling laws.
  - Emperically, we get $E \propto T^{-1}$ through a combination of voltage scaling, circuit design, and architectural tradeoffs.
- Parallel computing can still be a big-win for saving energy
  - Let's say we can build processors that run $\frac{1}{10}$ the speed of a fast sequential machine. They will each use $\frac{1}{100}$ of the power.
  - If a parallel version of the computation gets perfect speed-up, we can run it on 10 slow processors in the same time as running the sequential code on one fast processor.
  - The parallel version will use $\frac{1}{10}$ of the energy.

Where does the energy go

- For a general purpose processor: instruction fetch, decode, and other control.
- For a GPU: register file accesses.
- Compared with full-custom hardware:
  - A CPU can be 1000$\times$ less energy efficient.
  - A GPU can be 100$\times$ less energy efficient — that's better than a CPU, but there is still plenty of room for improvement.
- The factor of 100$\times$ energy waste of current architectures is begging for the next breakthrough.
  - What will that breakthrough be?

What went wrong: The Atom Wall

- Chips are now being designed where the gate length (i.e. dam thickness) is about 20 atoms.
- We need to squeeze a low concentration of dopant atoms into the channel.
- It's very hard to manufacture circuits where a few atoms makes a big difference.
  - All edges are jagged.
  - Photo-lithography (printing the circuit structures with light) is challenging because the transistors are much smaller than a wavelength of the UV light that is used.
  - Quantum mechanics becomes a big deal.
  - ...

What's next? (part 1)

- Parallel computing: how to make good use of Moore transistors without using more power.
- Optics:
  - Computer performance is often limited by chip-to-chip interconnect, e.g. the connection between a CPU an memory.
  - Glass is **much** better than copper.
  - Optical networking is standard in large data centers.
  - Optical interconnect between chips is emerging – there are clever ways to make modulate and detect light beams with silicon.
  - Wavelength-division multiplexing (WDM) is awesome – we can have hundreds of simultaneous channels on a single glass fibre by using different wavelengths of light.

What's next? (part 2)

- Higher bandwidth channels to memory
  - GPUs now use HBM and HBM2.
    - ★ This involves stacking 16 or 18 memory chips next to the GPU.
    - ★ The memory chips are connected to each other by polishing each chip down to a few tenths of a millimeter thick, etching holes in the chip, filling the holes with metal, and making connections.
    - ★ This allows 10× the number of connections between the memory chips and between the memory and the GPU.
  - Cryogenic memory?
    - ★ I've read recently about a joint project between Microsoft and Rambus to look at memory that runs in liquid nitrogen.
    - ★ Silicon in liquid nitrogen has wonderful electrical properties – the waves are much smaller.
    - ★ But, making reliable systems has been a show-stopper because wires become extremely brittle.
    - ★ I haven't seen how Microsoft and Rambus plan to address this.
  - Nanotubes, graphene, spintronics, molecular computing, quantum computing
    - ★ Many long-shots are being explored.

Preview

October 12: Homework 3 released, later today

October 15: Sorting Networks

October 17: The 0-1 Principle

October 18: HW 3 earlybird (11:59pm).

October 19: Midterm Review

Homework: HW3 due: **12 noon**.

October 22: Midterm

October 24-26: Sorting (second half)

October 29-November 30: Data Parallelism with CUDA

Summary (part 1)

- Transistors are voltage-controlled switches made of silicon
  - We can use controlled switches to make gates, registers, and everything digital.
- Making chips has been a great way to make money:
  - More money means better manufacturing processes.
  - Better manufacturing means Moore, smaller transistors on a chip.
  - Moore transistors, means Moore functionality, and Moore performance.
  - Better chips mean Moore profit

Summary (part 2)

- Moore's Law
  - The positive feedback loop described above leads to an exponential growth in number of transistors per chip, clock speed, memory capacity, etc.
  - Moore's Law is an economic law.
  - Exponential trends inevitably collide with physics.
- The end(?) of Moore's Law
  - The power wall – chips are at the cooling limit.
  - The atom wall – transistor sizes are now a few tens of atoms.
- Why Parallelism matters
  - Greater **throughput** with a huge number of of simpler, lower clock frequency processors.
  - The **only** way go grow performance is with more parallelism.
  - For the next 10-20 years, "the next big thing" will be parallel, nearly every time.