1

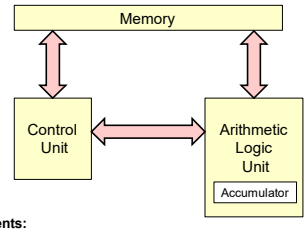
Parallel Programming: Moore's Law and Multicore

Oregon State University
Mike Bailey
mjb@cs.oregonstate.edu

1

2

Von Neumann Architecture: Basically the fundamental pieces of a CPU have not changed since the 1960s



Other elements:

- **Clock**
- Registers
- Program counter
- Stack pointer

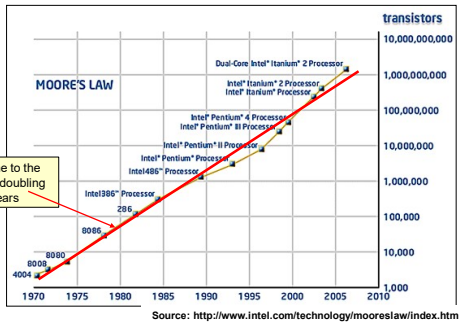
2

3

Increasing Transistor Density -- Moore's Law

"Transistor density doubles every 1.5 years."

Note: Log scale!



If I fit this line to the plot, I get a doubling every 1.6 years

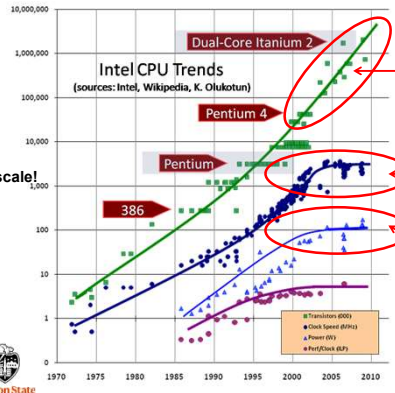
Source: http://www.intel.com/technology/mooreslaw/index.htm

Oftentimes people have (*incorrectly*) equivalenced this to: "Clock speed doubles every 1.5 years."

3

4

Increasing Clock Speed?



Note: Log scale!

Transistor count

This is what Moore's Law really deals with!

Clock speed

This is *not* what Moore's Law really deals with!

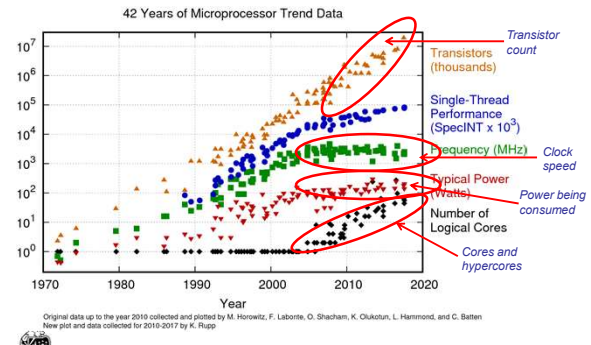
Power being consumed

Source: Intel

4

5

Increasing Clock Speed?



Original data up to the year 2010 collected and plotted by M. Horowitz, F. Labonte, O. Shacham, K. Olukotun, L. Hammond, and C. Batten
New plot and data collected for 2010-2017 by K. Rupp

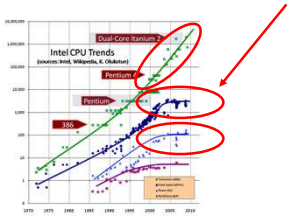
Source: Karl Rupp

5

6

Moore's Law

- Fabrication process size ("gate pitch") has fallen from 65 nm, to 45 nm, to 32 nm, to 22 nm, to 16 nm, to 11 nm, to 8 nm. This translates to more transistors on the same size die.
- From 1986 to 2002, processor performance increased an average of 52%/year, but then virtually plateaued.



6

Clock Speed and Power Consumption

7

1981	IBM PC	5 MHz
1995	Pentium	100 MHz
2002	Pentium 4	3000 MHz (3 GHz)
2007		3800 MHz (3.8 GHz)
2009		4000 MHz (4.0 GHz)

Clock speed has hit a plateau, largely because of power consumption and power dissipation.

$$PowerConsumption \propto ClockSpeed^2$$

is-proportional-to

Yikes!

Once consumed, that power becomes *heat*, which much be dissipated somehow. In general, compute systems can remove around 150 $^{watts}/_{cm}$ without resorting to exotic cooling methods.

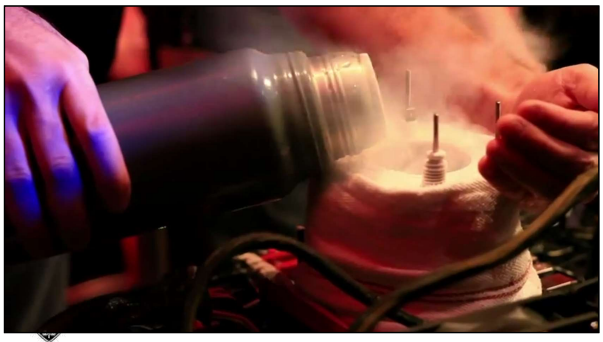
Oregon State University Computer Graphics

mjb – March 9, 2022

7

And, speaking of "exotic", AMD set the world record for clock speed (8.429 GHz) using a Liquid Nitrogen-cooled CPU

8



Source: *AMD*

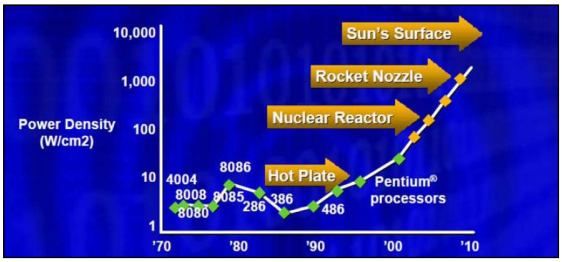
Oregon State University Computer Graphics

mjb – March 9, 2022

8

What Kind of Power Density Dissipation Would it Have Taken to Keep up with Clock Speed Trends?

9



Source: Intel

Oregon State University Computer Graphics

mjb – March 9, 2022

9

MultiCore -- Multiprocessing on a Single Chip

10

So, to summarize:

Moore's Law of transistor density is still going, but the "Moore's Law" of clock speed has hit a wall. Now what do we do?

We keep packing more and more transistors on a single chip, but don't increase the clock speed. Instead, we increase computational throughput by using those transistors to pack multiple processors onto the same chip.

This is referred to as ***multicore***.

Vendors have also reacted by adding SIMD floating-point units on the chip as well. We will get to that later.

Oregon State University Computer Graphics

mjb – March 9, 2022

10

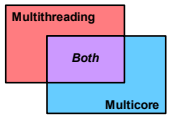
MultiCore and Multithreading

11

Multicore, even without multithreading too, is still a good thing. It can be used, for example, to allow multiple programs on a desktop system to always be executing concurrently.

Multithreading, even without multicore too, is still a good thing. Threads can make it easier to logically have many things going on in your program at a time, and can absorb the dead-time of other threads.

But, the big gain in performance is to use *both* to speed up a *single program*. For this, we need ***a combination of both multicore and multithreading***.



Multicore is a very hot topic these days. It would be hard to buy a CPU that doesn't have more than one core. We, as programmers, get to take advantage of that.

We need to be prepared to convert our programs to run on ***MultiThreaded Shared Memory Multicore*** architectures.

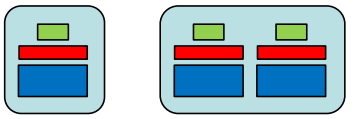
Oregon State University Computer Graphics

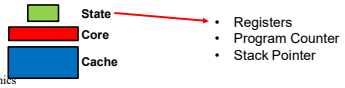
mjb – March 9, 2022

11

Each of the Multiple Cores keeps its own State

12

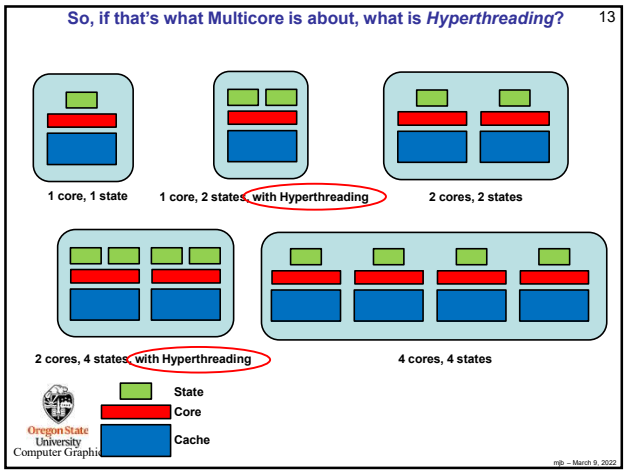




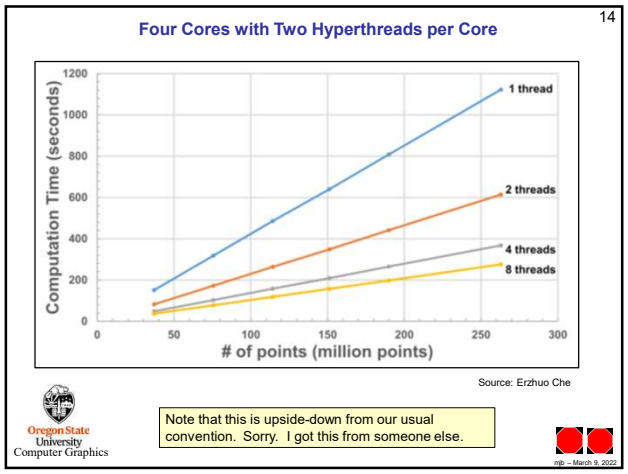
Oregon State University Computer Graphics

mjb – March 9, 2022

12


So, if that's what Multicore is about, what is *Hyperthreading*?
13
1 core, 1 state
1 core, 2 states, with Hyperthreading
2 cores, 2 states
2 cores, 4 states, with Hyperthreading
4 cores, 4 states
State
Core
Cache
Oregon State University Computer Graphics
mjb – March 9, 2022

13


Four Cores with Two Hyperthreads per Core
14
Computation Time (seconds)
# of points (million points)
1 thread
2 threads
4 threads
8 threads
Source: Erzhuo Che
Oregon State University Computer Graphics
Note that this is upside-down from our usual convention. Sorry. I got this from someone else.
mjb – March 9, 2022

14