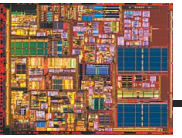
ECE411: Computer Organization and Design

Lecture 1: Course Introduction

Rakesh Kumar

ILLINOIS

Which is faster?

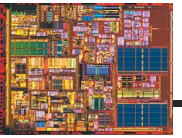


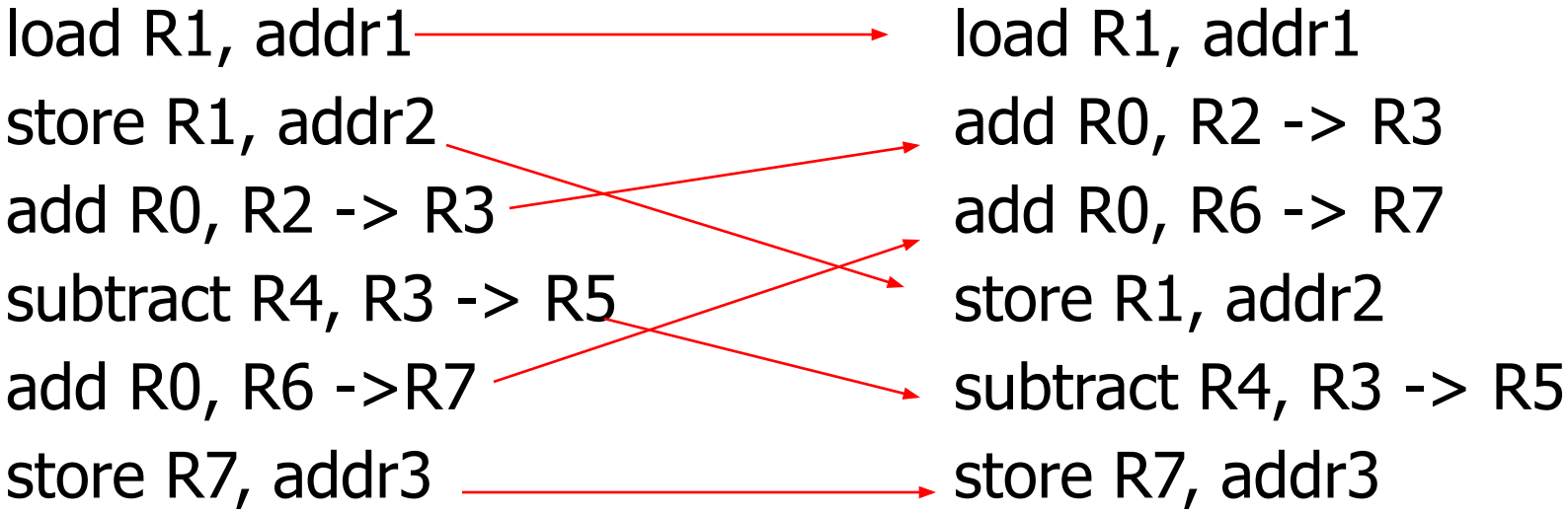
```
for (i=0; i<N; i=i+1)
   for (j=0; j<N; j=j+1) {
      r = 0;
      for (k=0; k<N; k=k+1)
         r = r + y[i][k] * z[k][j];
      x[i][j] = r;
   }
```

```
for (jj=0; jj<N; jj=jj+B)
for (kk=0; kk<N; kk=kk+B)
for (i=0; i<N; i=i+1) {
   for (j=jj; j<min(jj+B-1,N); j=j+1)
      r = 0;
      for (k=kk; k<min(kk+B-1,N); k=k+1)
         r = r + y[i][k] * z[k][j];
      x[i][j] = x[i][j] + r;
}
```

Significantly Faster!

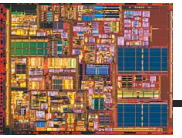
Which is faster?





Twice as fast on some machines and same on others

Which is faster?



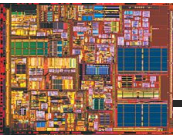
```
loop1:    add ...
          load ...
          add ...
          bne R1, loop1

loop2:    add ...
          load ...
          bne R2, loop2
```

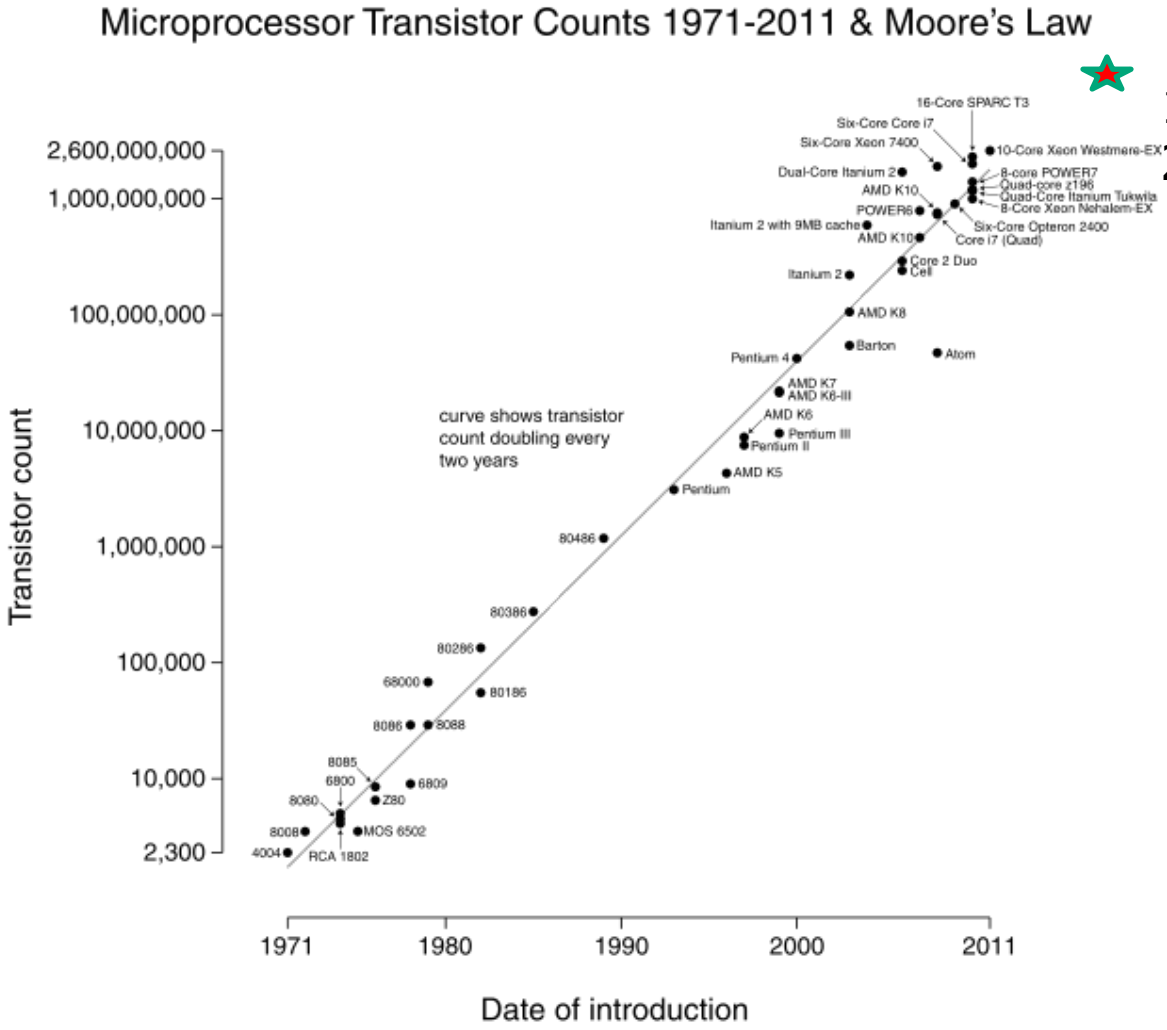
```
loop1:    add ...
          load ...
          add ...
          bne R1, loop1
          nop
          nop

loop2:    add ...
          load ...
          bne R2, loop2
```

Identical performance on several machines

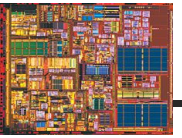


Processor Design is Hard/Interesting!

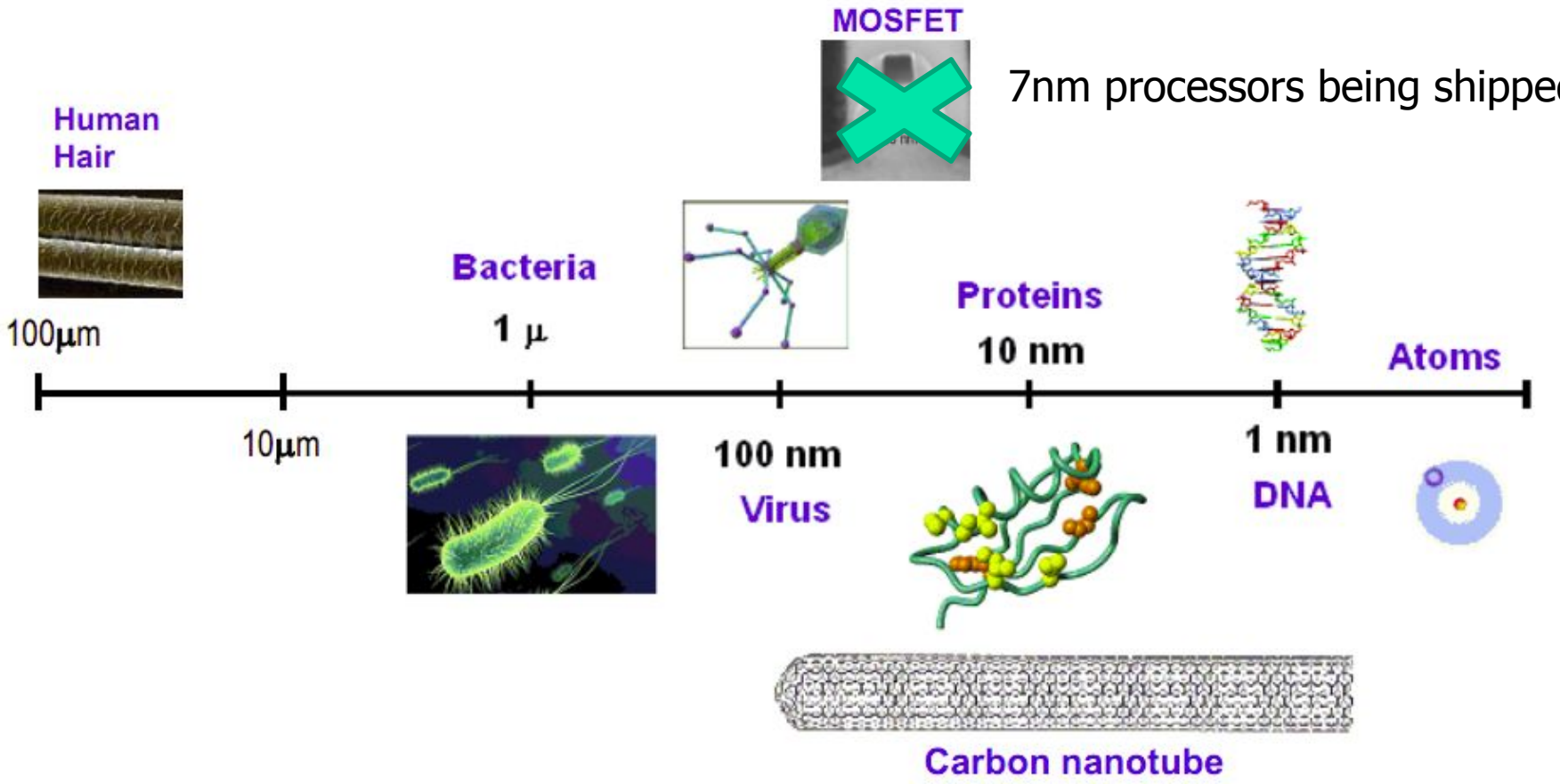


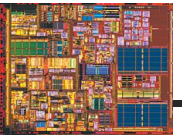
1.2 trillion (cerebras)
21billion(GV100 Volta)

Modern processors have more than a billion transistors!



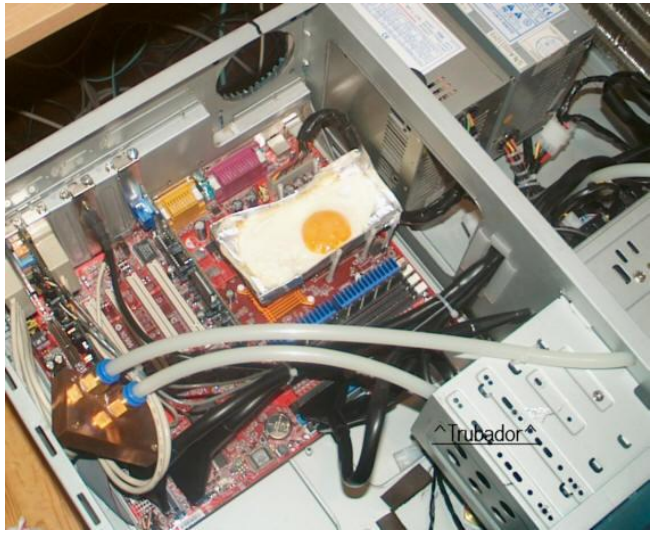
Processor Design is Hard/Interesting!





Processor Design is Hard/Interesting!

FDIV Pentium bug cost 500 million dollars!



Very recently, power density of a processor higher than a nuclear reactor!

Power Efficiency Challenges

ASCI Red @ Sandia Labs

1997 (850KW) ~1.5Teraflops 2022 (5W)

- power/energy-constrained computing devices
  - mobile: 1~5W power budget w/ 5Whr battery capacity

About the Instructor and Course Staff

- Rakesh Kumar
  - Office half hours (CSL208)
    - Tuesday 3:30-4pm CST
  - Email (rakeshk@illinois.edu, with subject title [ECE411] please)
- TAs

○ Nicholas Satchanov (nsatch3@illinois.edu), Arjun Pal (aspal2@illinois.edu) Scott Smith (scottcs2@illinois.edu), Stanley Wu (zaizhou3@illinois.edu), Collin Meyer (cmmeyer3@illinois.edu) Aditya Nebhranjani (avn5@illinois.edu), Frank Cai (zongrui3@illinois.edu), Ethan Greenwald (ethanmg4@illinois.edu), Noelle Crawford (noellec3@illinois.edu), Satvik Yellanki (satviky2@illinois.edu), Udit Pai (upai3@illinois.edu), Dhaval Purohit (dpuroh2@illinois.edu), Ian Dailis (idailis2@illinois.edu)

- Office hours are listed on class website: most hold 2 hours of OH per week
- Active on Campuswire and Discord: ○ Join information in welcome mail

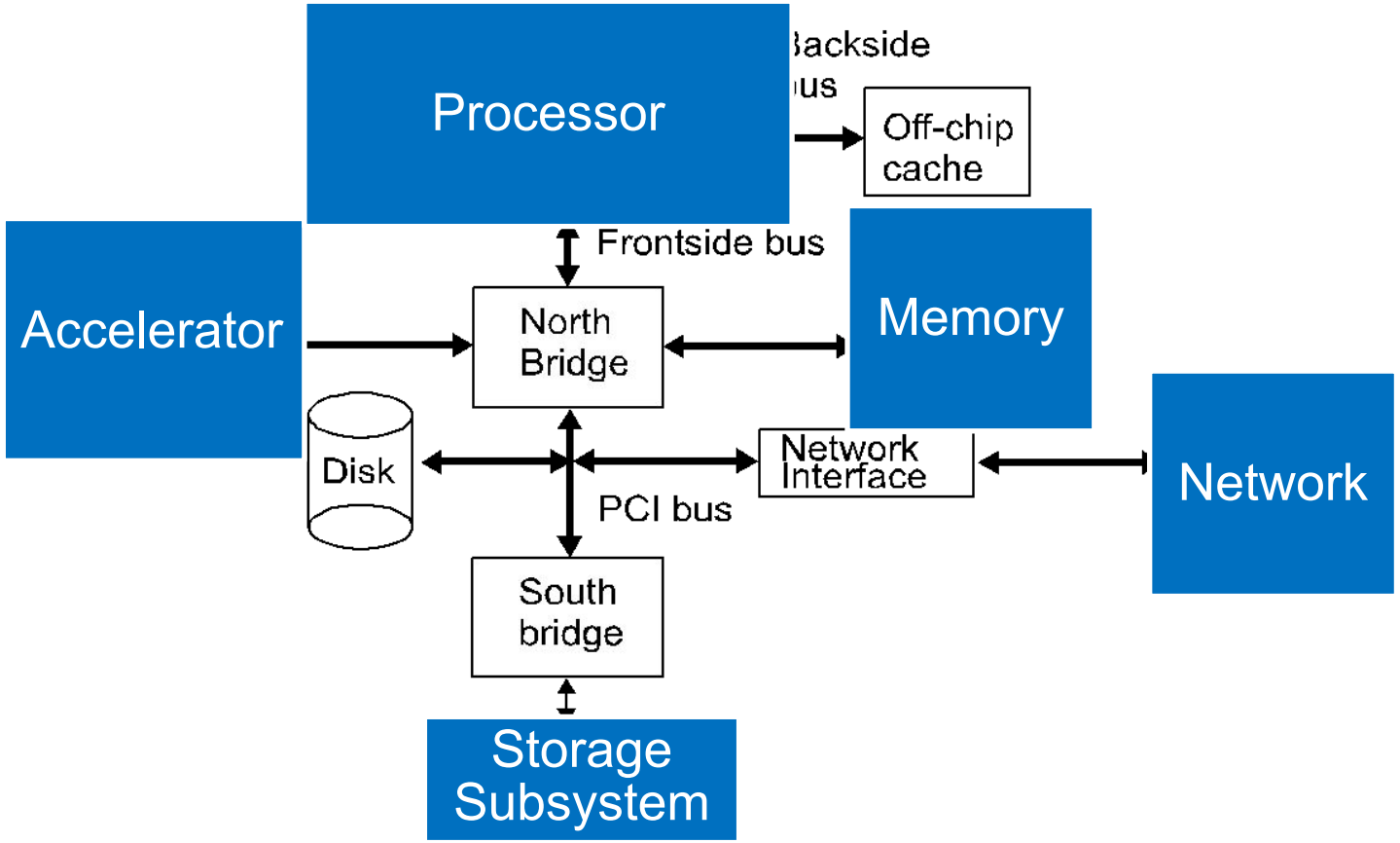
Computing is everywhere!

http://images.slideplayer.com/26/8674558/slides/slide_3.jpg

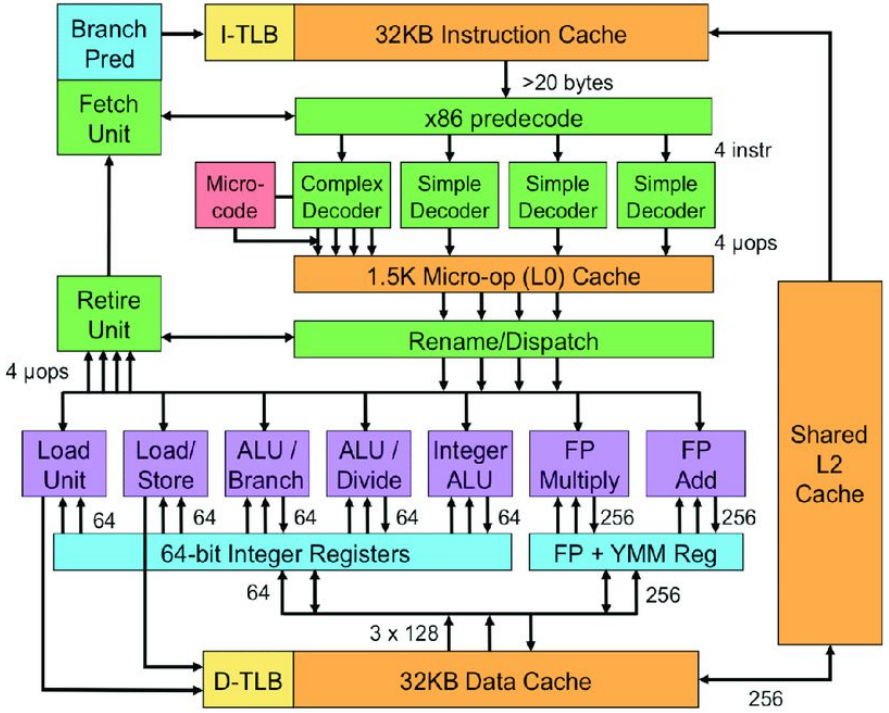
Computer Organization

Q: What are the major components we have in a computer today?

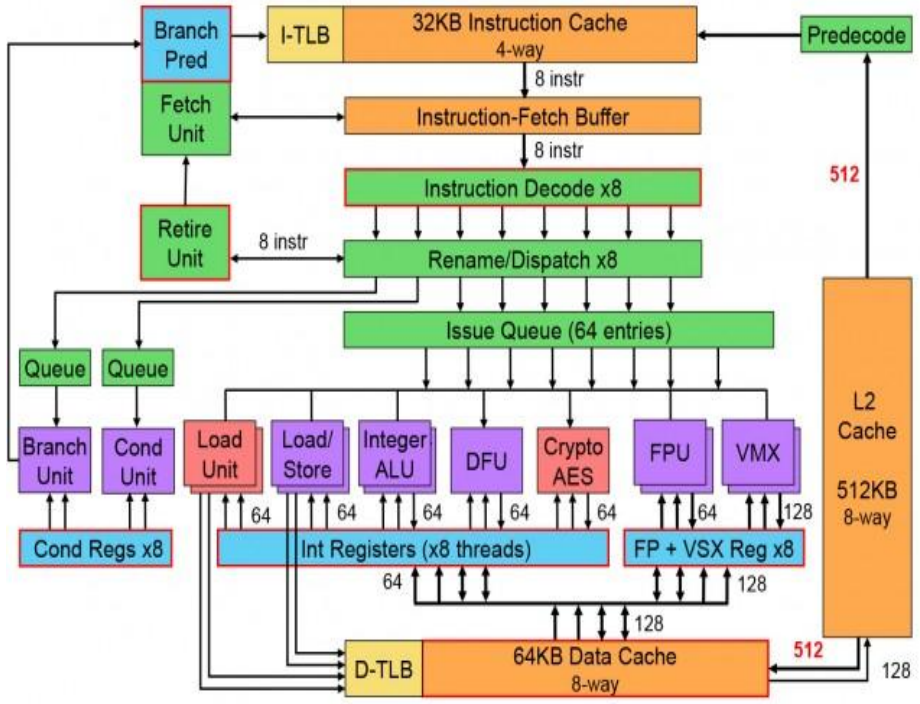
- organization of modern computers



Microprocessor Architecture



Intel Sandy Bridge Processor



IBM Power 8 Processor

Course Objective

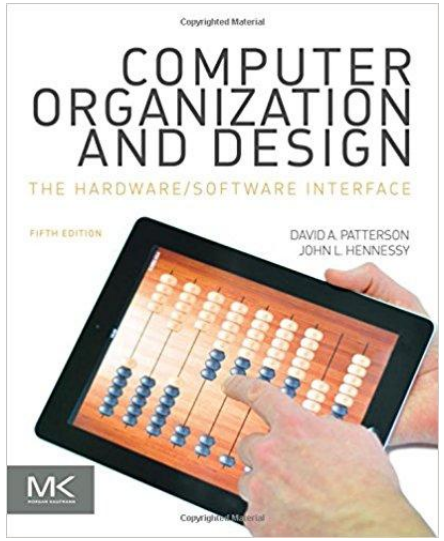
- to understand fundamental concepts in computer architecture and how they impact application performance and energy efficiency
- to become confident in programming for performance, scalability, and efficiency
- to be able to understand and evaluate architectural descriptions of even today's most complex processors
- to gain experience designing a working CPU completely from scratch
- to learn experimental techniques used to evaluate advanced architectural concepts

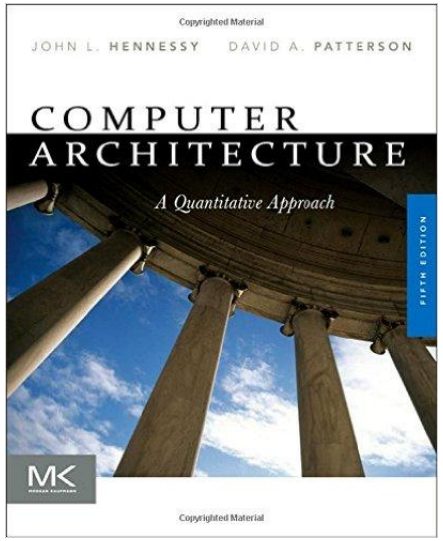
Course Content & Prerequisites

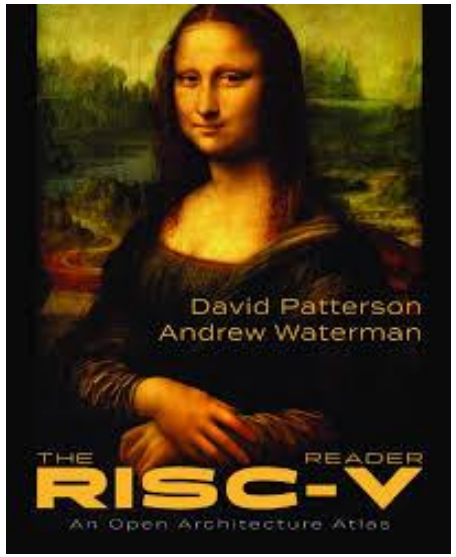
- fundamental concepts in computer organization focusing on uni-processor architecture
  - pipelining
  - memory hierarchy
  - instruction-level parallelism & dynamic scheduling
  - storage I/O subsystems
  - impact of technology on performance and energy efficiency
  - hardware accelerators
  - advanced architecture topics
- prerequisites
  - ECE 385 (programming and debugging skills)
    - ECE 391 (not a pre-req, but useful virtual memory part)
  - Linux
  - C++ programming
  - SystemVerilog

Textbooks

- course textbook
  - [HP1] Patterson & Hennessy. Computer Organization and Design (5th Ed.); Morgan Kaufmann; ISBN: 978-0-12-407726-3
- recommended textbook
  - [HP2] Hennessy & Patterson. Computer Architecture: A Quantitative Approach (5th Ed.); The Morgan Kaufmann, ISBN: 978-0123838728
  - [PW] Patterson & Waterman. The RISC-V Reader: An Open Architecture Atlas; ISBN: 978-0-9992491-1-6







Lectures and MPs

- Lectures
  - important to attend lecture as a notable fraction of materials are not explained in textbooks

MP Overview

○ mp_setup: Release: 1/15/24 (yesterday!) Due: N/A, ○ mp_verif: Release: 1/16/24 (today!) Due: 1/29/24

○ mp_pipeline: Release: 1/26/24 Due: 2/19/24, ○ mp_cache: Release: 2/16/24 Due: 3/4/24

○ mp_bp: Release: 3/1/24 Due: 3/18/24, ○ mp_ooo: Release: 3/8/24 Competition: 4/26/24 Presentation: 4/27/24 Report: 5/1/24

● Solo/Group Work: [mp_setup, mp_bp] -> Solo, [mp_ooo] -> Groups of 3

● Late Pass: One pass to receive no late penalty on a single MP of your choice barring mp_ooo: See class site for details

● Autograder: ○ Grades MP submissions, ○ Will start running some days prior to the deadline

● Lab Sessions: ○ Planned every week on Wednesdays (stay tuned for exact room and time!), ○ Only one session (not like 220, 374, 391 discussions), ○ Recorded

Exams & Grading

- Exams
  - mid-term 1
    - 5th March (12%)
  - mid-term 2
    - 9th April (12%)
  - final
    - TBD (8%)
  - there will be a review session before each exam
  - Close-book/notes but one A4-page cheat sheet allowed.

- Graded on a curve

Website and Discussion Group

- Course website
  - Lecture notes, handouts, MP assignment, etc.
  - announcements
- Campuswire
  - posting clarification questions
  - general forum to communicate with students, TAs, instructor about aspects of the course.
  - must join by the end of the day
- Canvas
  - Grade tacking, exam grading, etc.

Academic Integrity

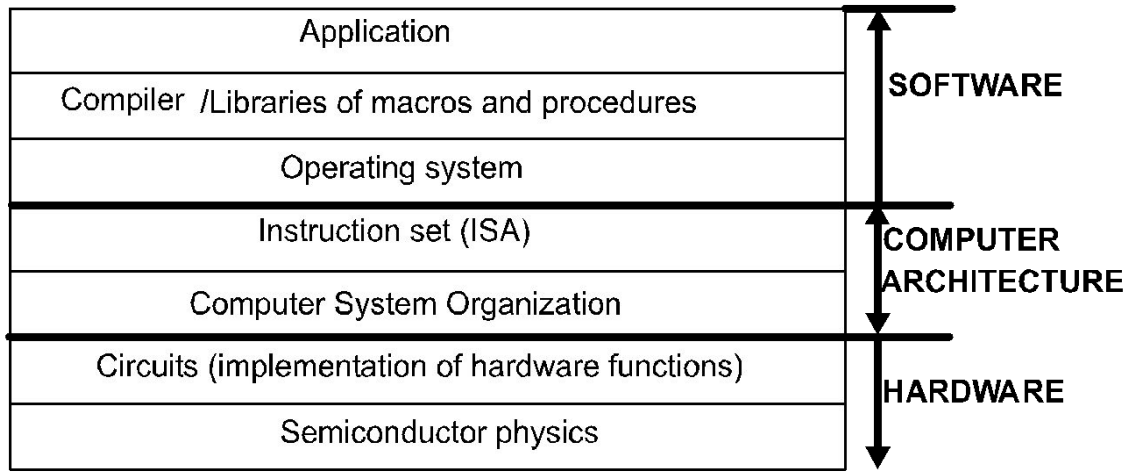
- you are **encouraged to discuss** assignments w/ anyone you feel may be able to help you (except during exams)
- everything you turn in must be your work or that of your team
- we encourage collaboration
  - verbal discussion of problems/solutions
  - diagrams, notes, other communication aids
- unacceptable
  - copying/exchanging of program/verilog code or text by any means
  - giving/receiving help on exams, or using any notes/aid beyond those explicitly permitted

Ready for ECE411?

- Requires a lot of time commitment
- Skills in SystemVerilog (programming & debugging)
- Plan your study well

What Is Computer Architecture?

- Role in the full computing system stack

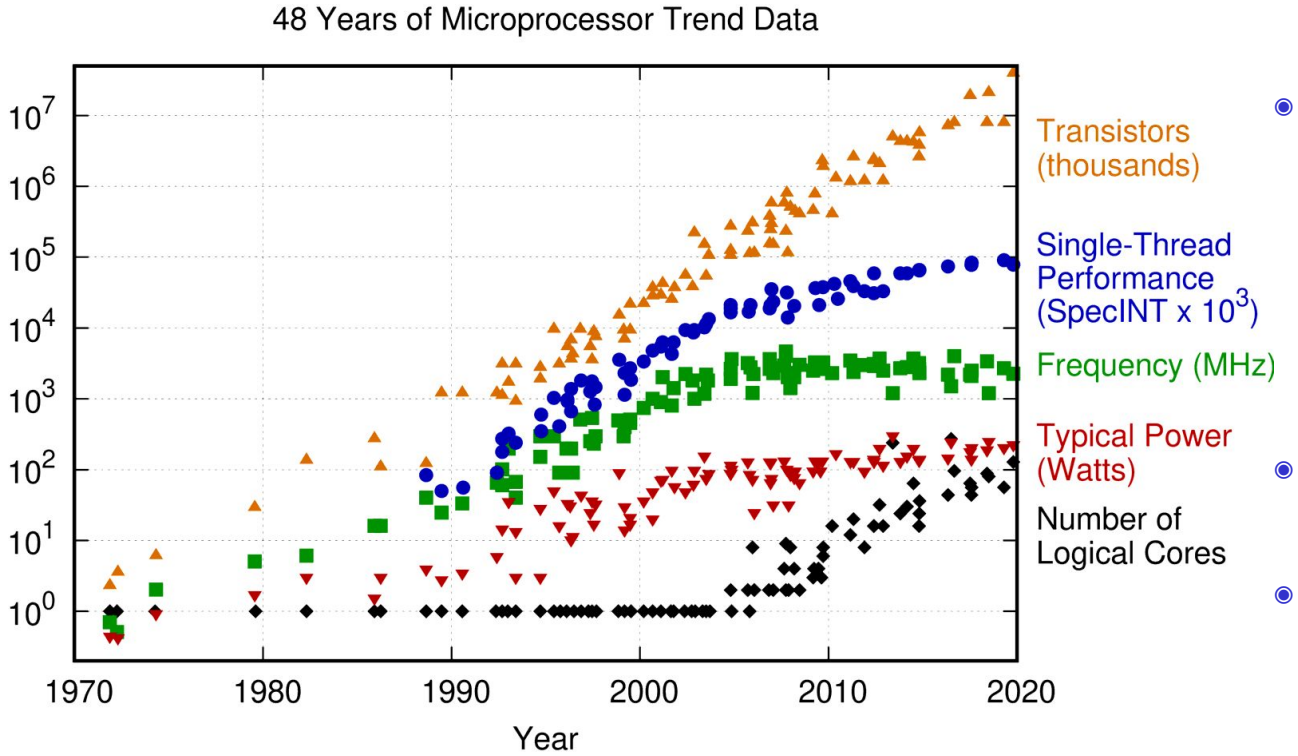


- Role of computer architects
  - to make design trade-offs across HW/SW interface to satisfy functional, performance, power/energy, and cost requirements

Processor Architecture: Historical Trend

Clock frequency: Good indicator for performance

What drove clock frequency increase in the past?



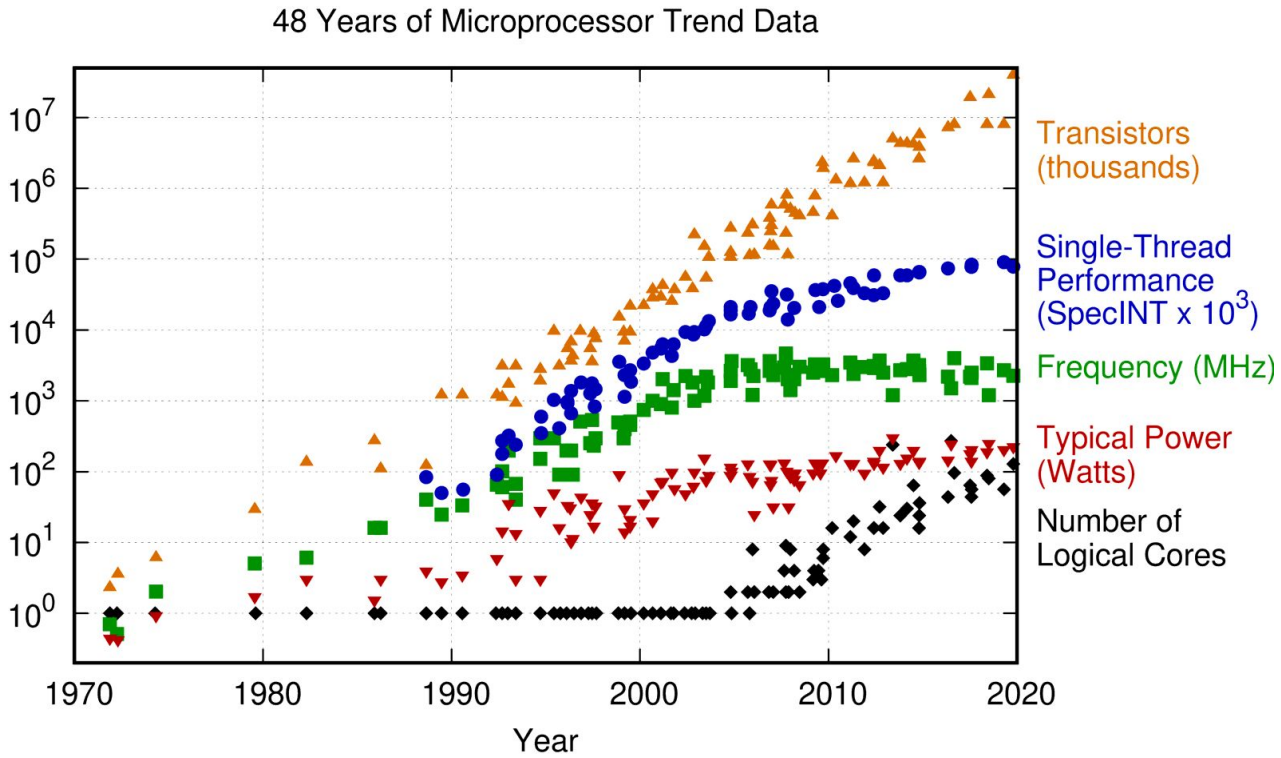
Original data up to the year 2010 collected and plotted by M. Horowitz, F. Labonte, O. Shacham, K. Olukotun, L. Hammond, and C. Batten
New plot and data collected for 2010-2019 by K. Rupp

Source data is here

- highest clock rate of Intel processors from 1990 to 2008
  - due to semiconductor technology improvements
  - deeper pipeline
  - circuit design techniques
- this historical trend has subsided over the past 10 years
- If it had kept up, today's clock rates would be more than 30GHz

Processor Architecture: Historical Trend

Why did we stop increasing the clock frequency?



Source data is here

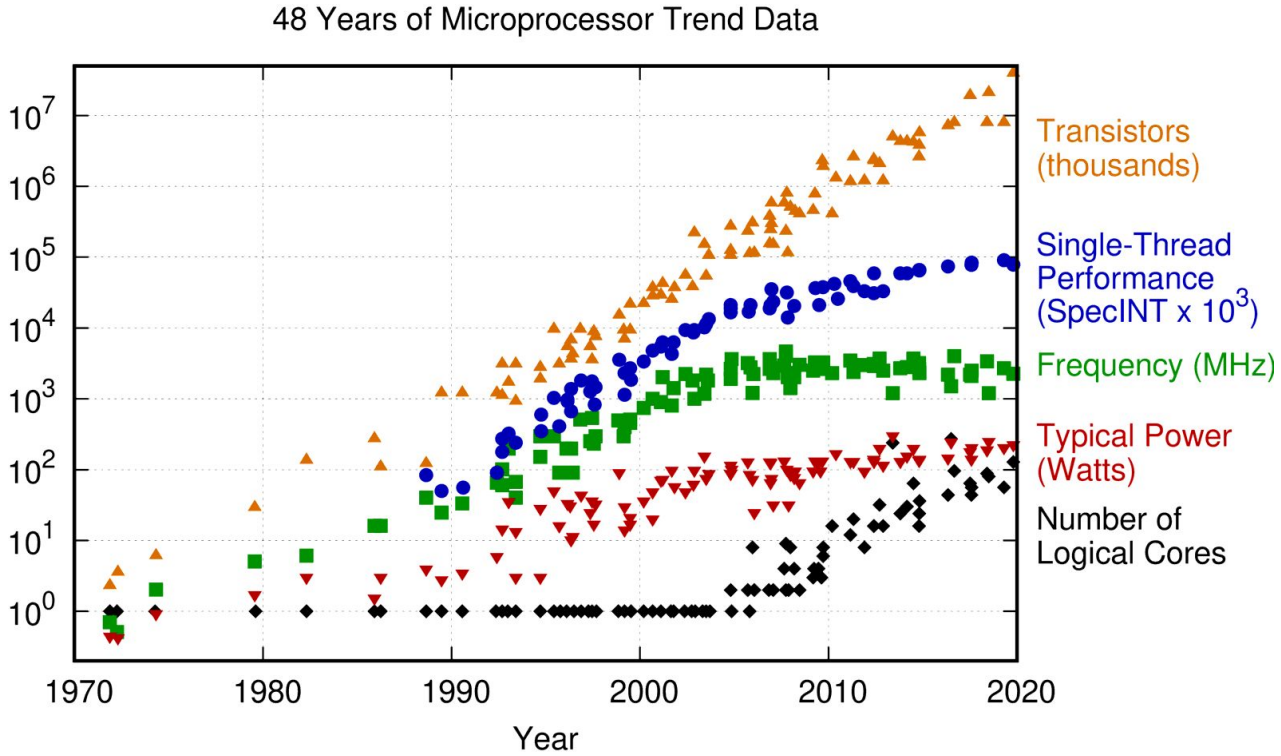
(1) Slowing down semiconductor technology scaling;

(2) excessive power consumption

Processor Architecture: Historical Trend

Moore's Law: the number of transistors in a dense integrated circuit doubles every two years

1965: double every year; 1975: double every year until 1980, after that double every two years;
2010: further slow down seen from industry; Trend is continuing (e.g., TSMC, Samsung)



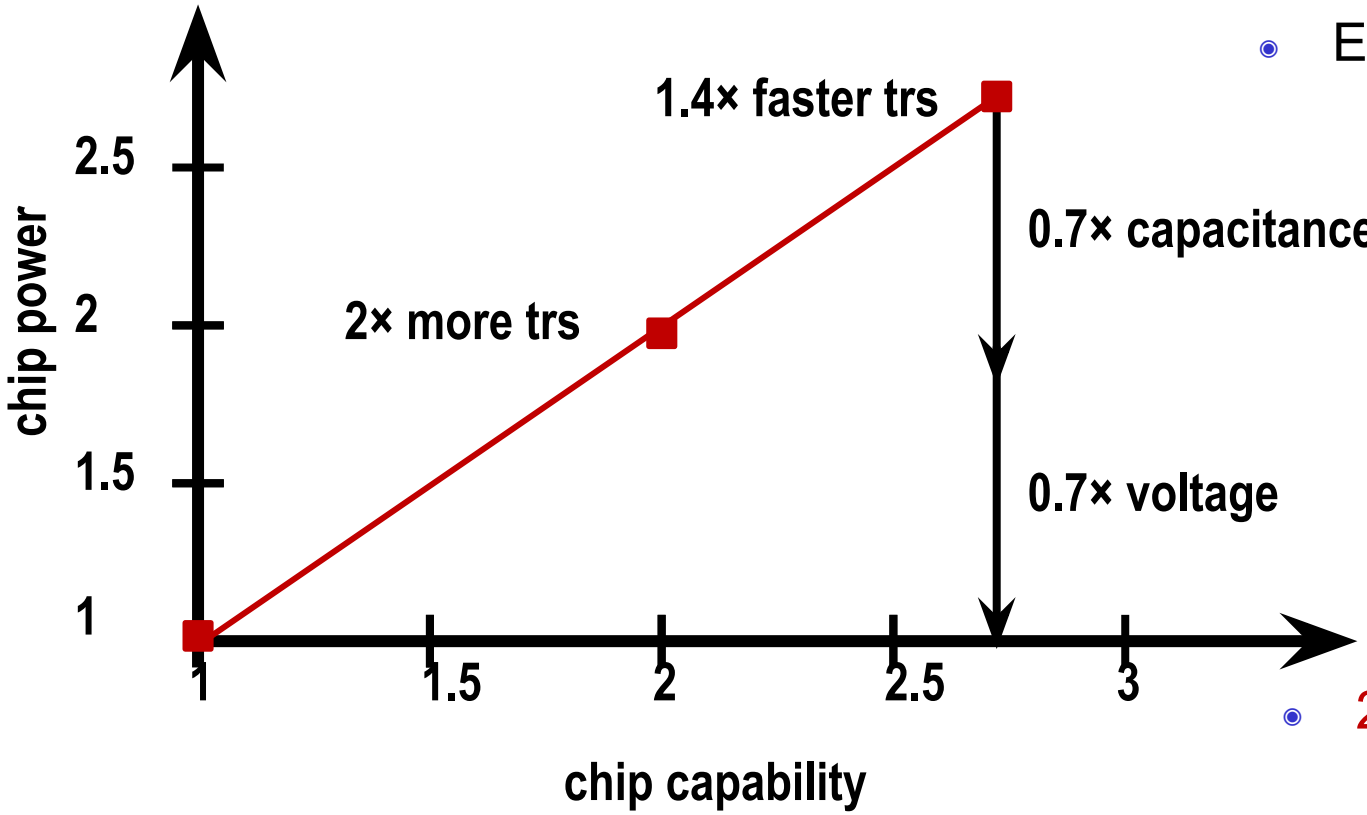
Original data up to the year 2010 collected and plotted by M. Horowitz, F. Labonte, O. Shacham, K. Olukotun, L. Hammond, and C. Batten
New plot and data collected for 2010-2019 by K. Rupp

Source data is here

Processor Architecture: Historical Trend

Classic Dennard's Scaling: when transistors get smaller, their power density stays constant such that the power consumption remains in proportion with area.

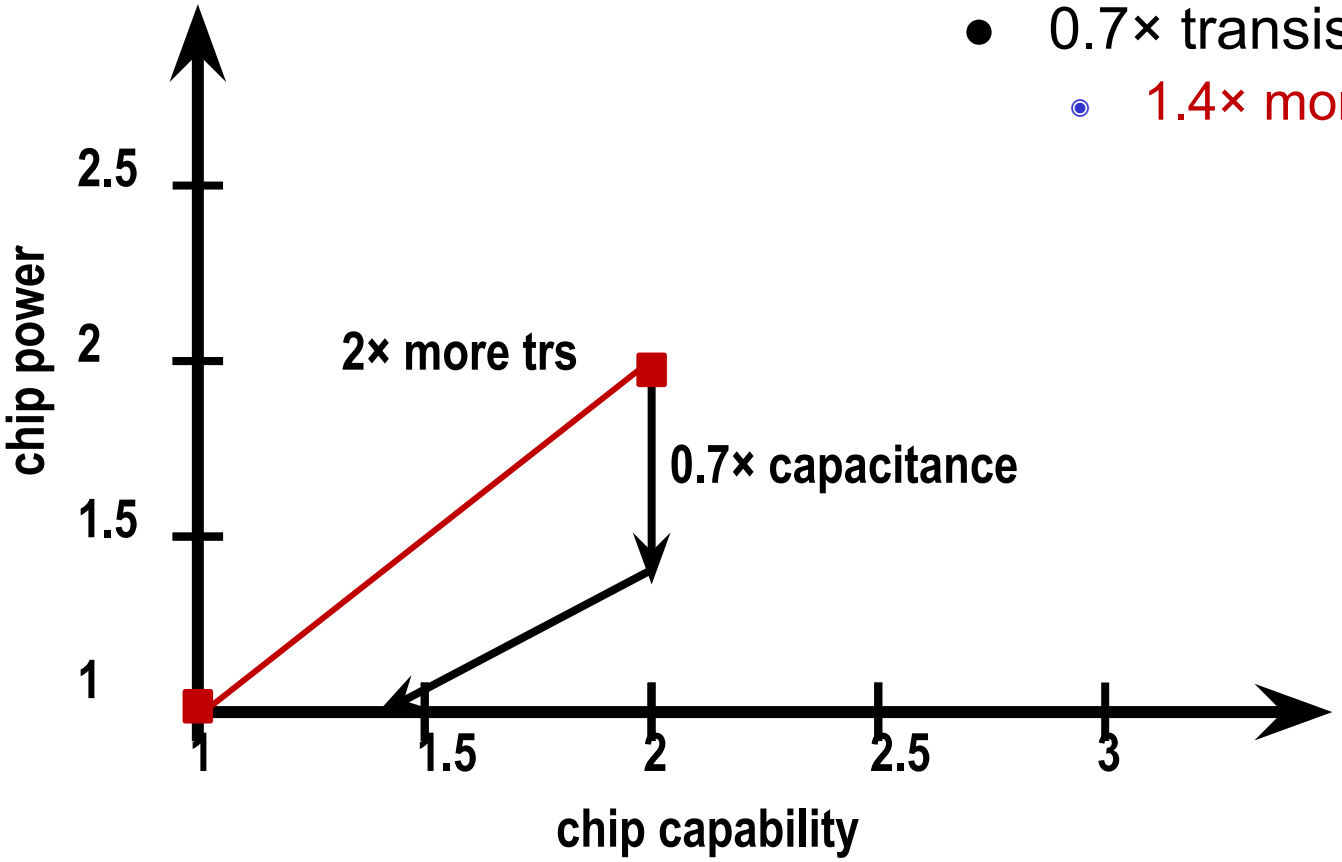
Proposed in 1974 by Robert H. Dennard at IBM



- Each technology generation
  - Transistor dimensions scaled by 30% (0.7x)
  - Area reduced by 0.5x
  - V (voltage) is reduced by 30% (0.7x)
  - Circuit delay reduced by 30% (0.7x)
  - Frequency = 1 / delay = 1.4x
  - Capacitance = area / distance = 0.5/0.7 = 0.7x
  - Power = CV^2f = 0.5x
- 2.8× more chip capability at the same power

Processor Architecture: Historical Trend

Dennard's Scaling ended in 2005-2007

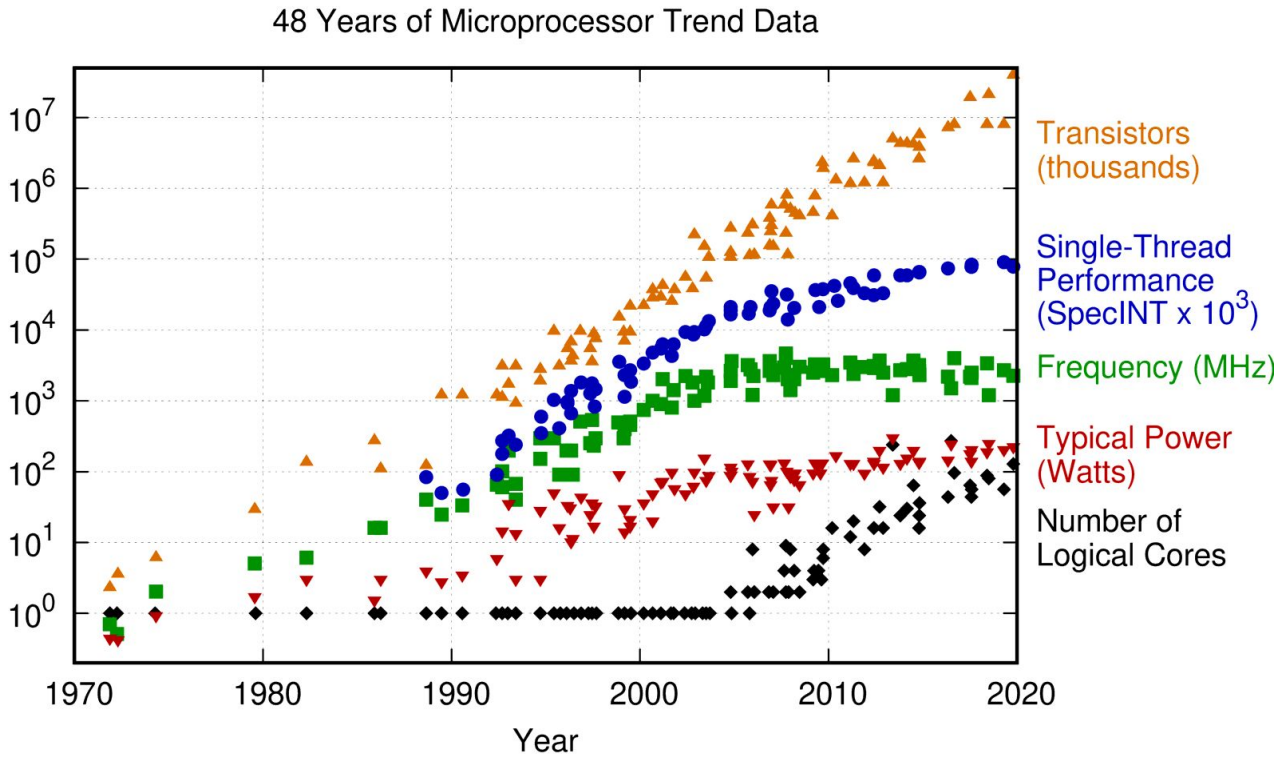


- 0.7× transistor feature size scaling per generation
  - 1.4× more chip capability at the same power

Processor Architecture: Historical Trend

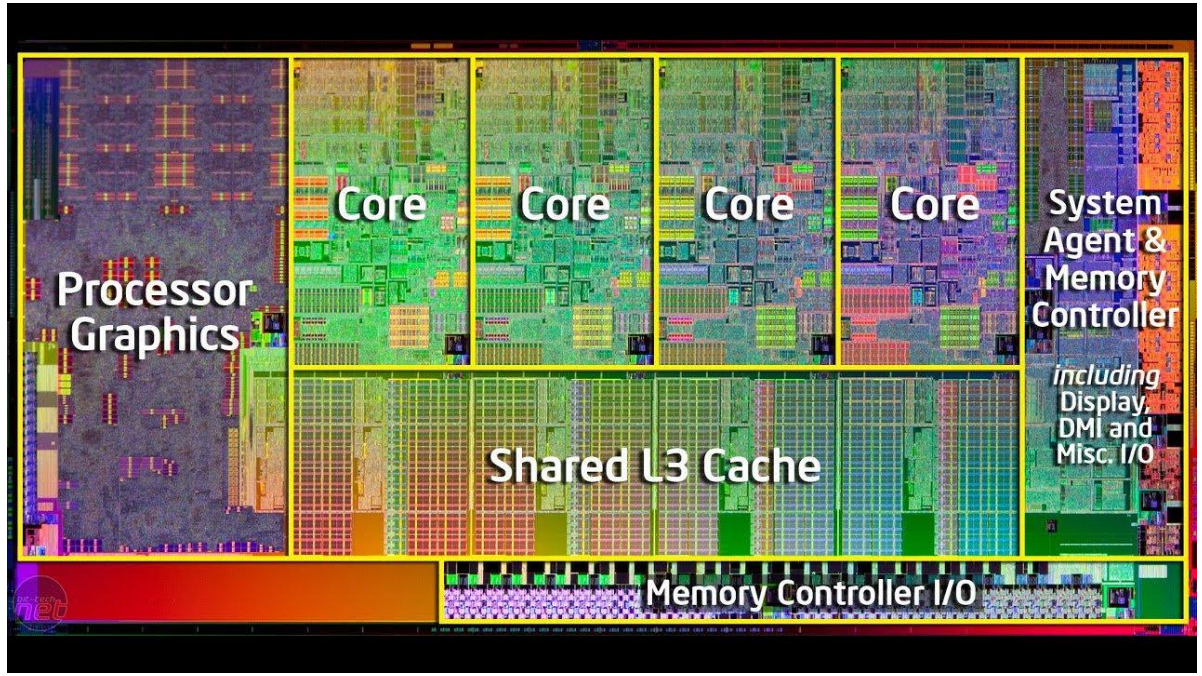
What did we do to improve performance after stopping clock frequency increase?

(1) Increase the number of cores;



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 2010-2019 by K. Rupp

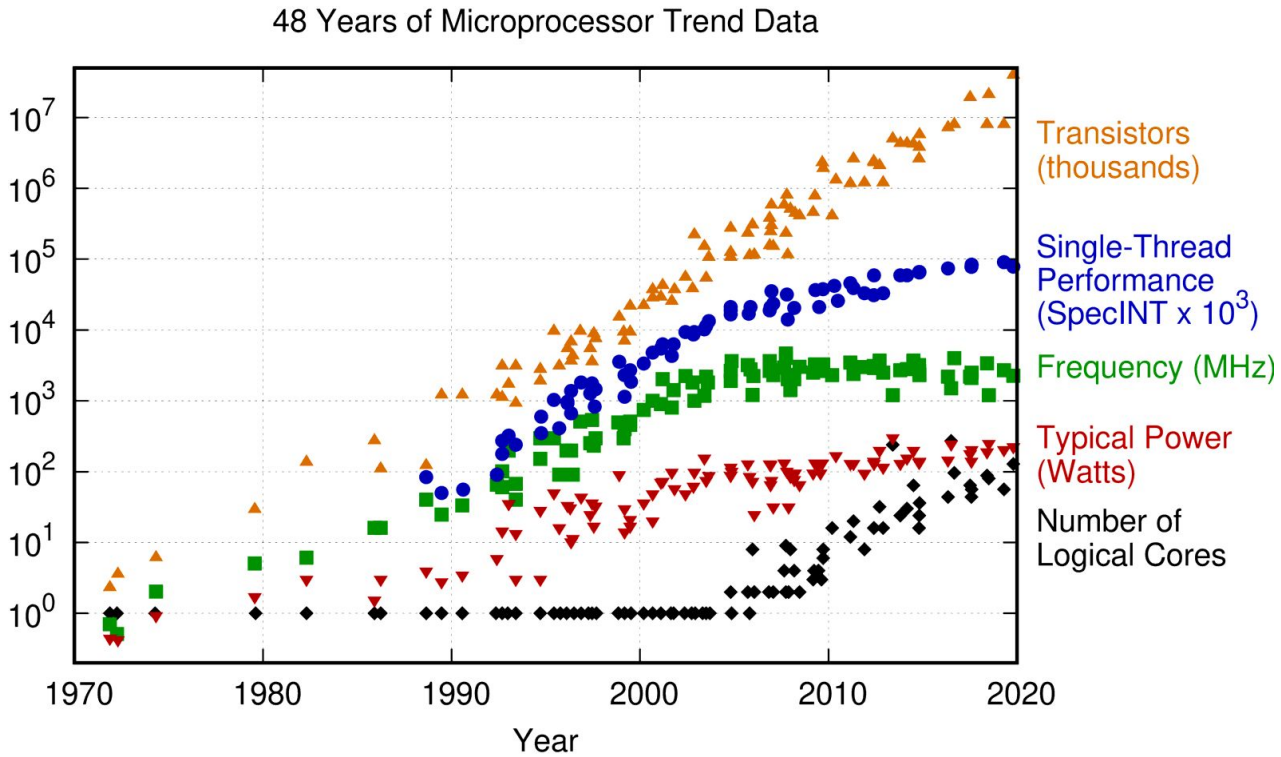
Source data is here



Processor Architecture: Historical Trend

What did we do to improve performance after stopping clock frequency increase?

(1) Increase the number of cores; (2) specialized hardware, GPU, accelerators.



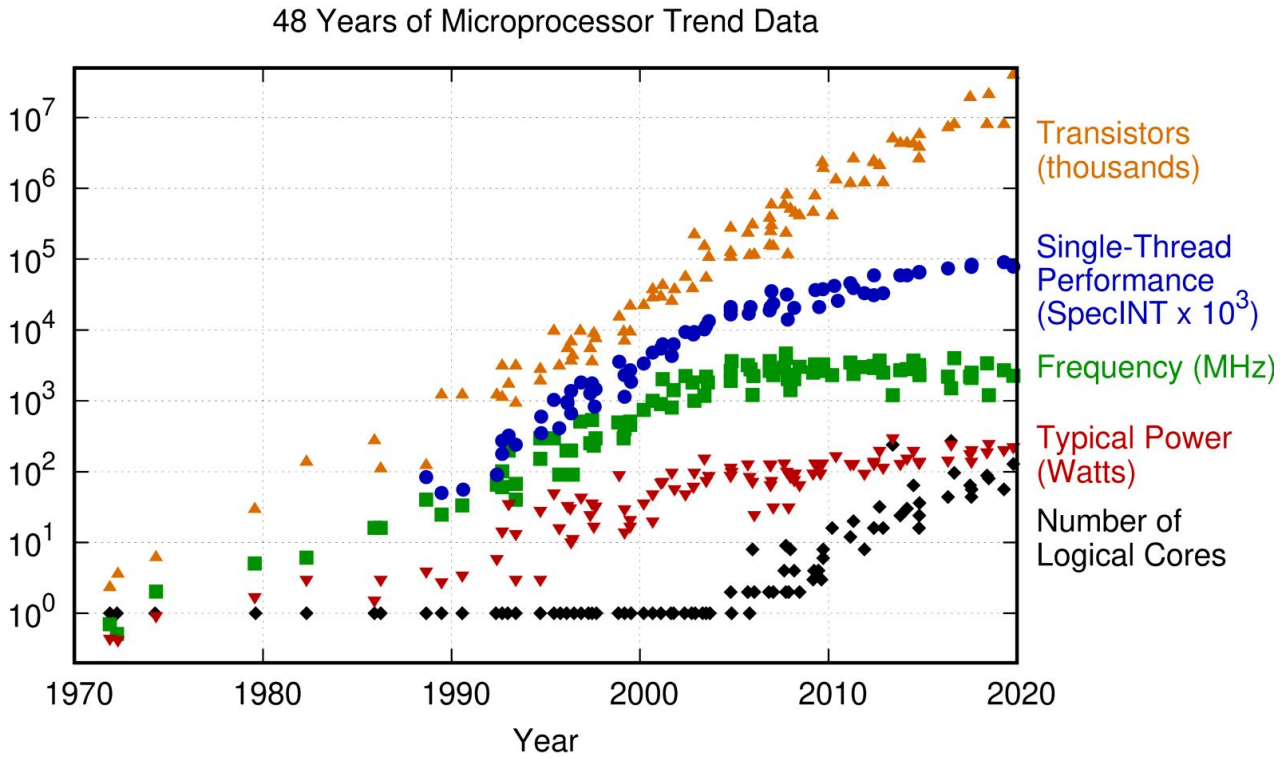
Original data up to the year 2010 collected and plotted by M. Horowitz, F. Labonte, O. Shacham, K. Olukotun, L. Hammond, and C. Batten
New plot and data collected for 2010-2019 by K. Rupp

Source data is here

Processor Architecture: Historical Trend

What did we do to improve performance after stopping clock frequency increase?

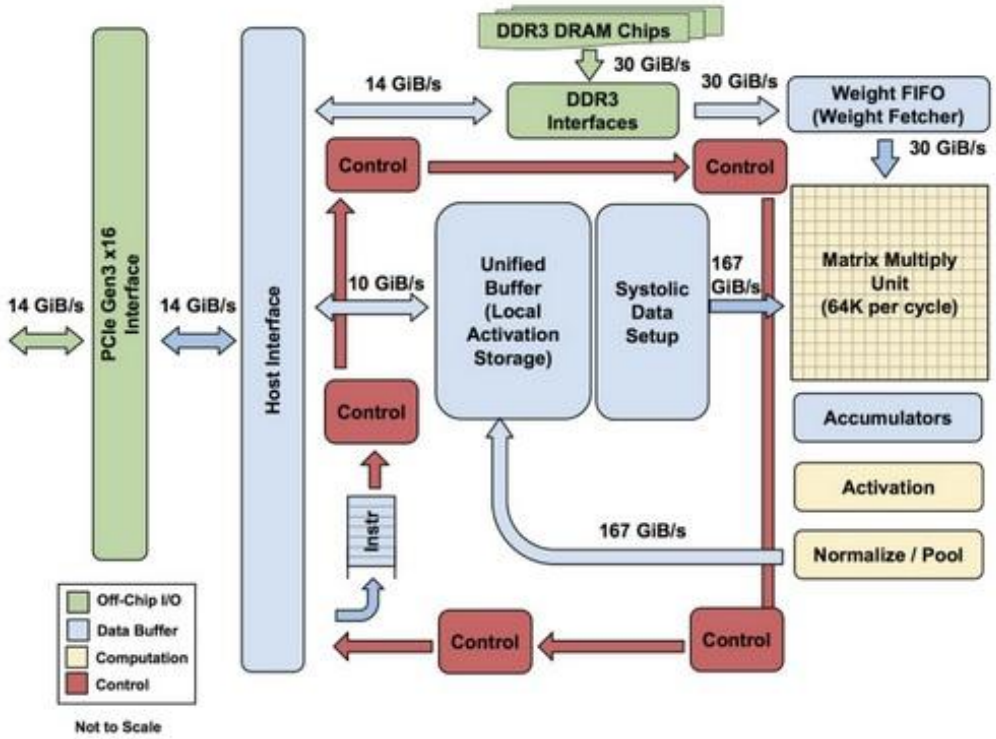
(1) Increase the number of cores; (2) specialized hardware, GPU, accelerators.



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 2010-2019 by K. Rupp

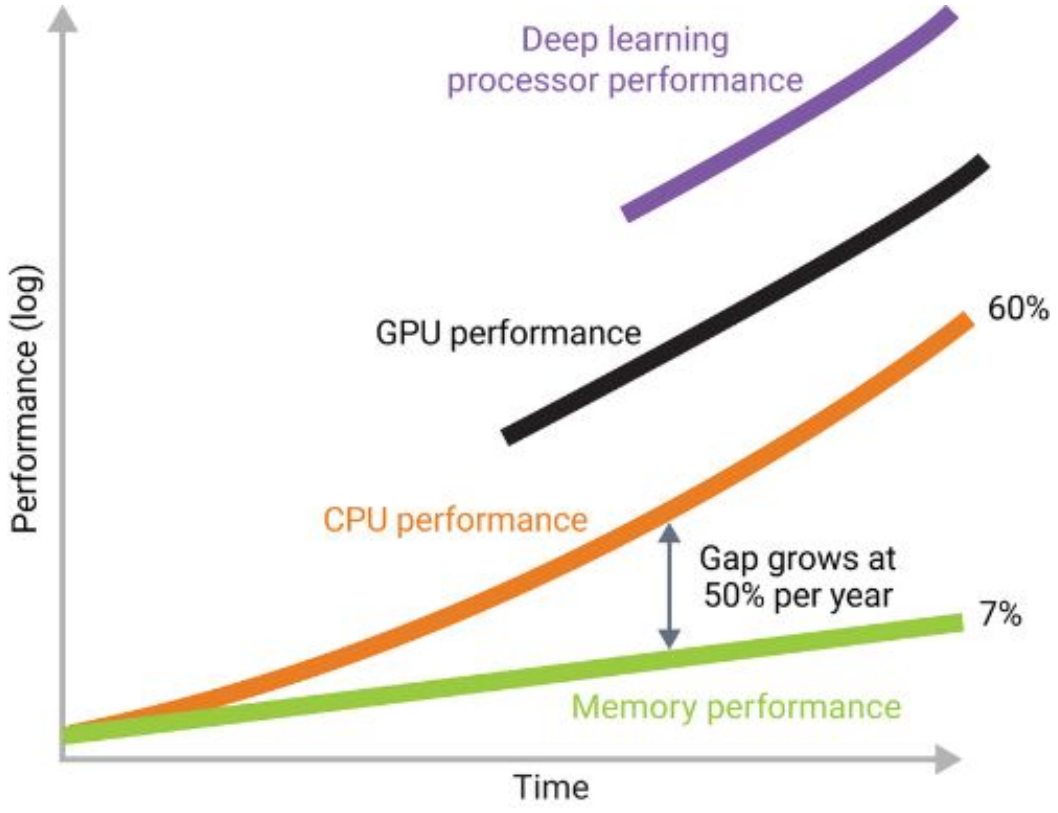
Source data is here

Google TPU

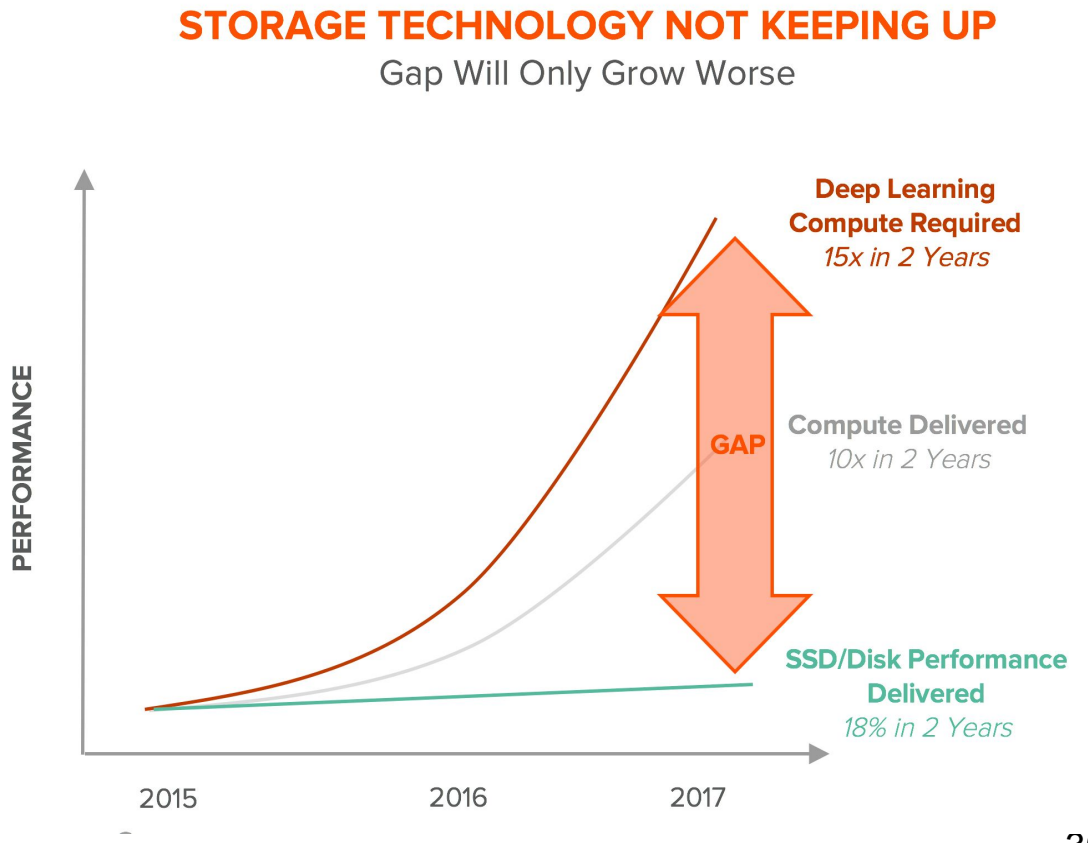


Memory/Storage Architecture: Historical Trend

The gap between compute and memory/storage is increasing



https://medium.com/@abruyns/memory-holds-the-keys-to-ai-adoption-5acd5e06508b



Pure Storage Inc.