

AI Computing Systems Introduction

Instructor: Siting Liu

School of Information Science and Technology (SIST), ShanghaiTech University

Shanghai Institute of Semiconductor Innovation (SISI)

<https://faculty.sist.shanghaitech.edu.cn/liust/EE219/EE219.html>



立志成才 报国裕民

Introduction

- Development of IC industrial (computing chips)
- Development of AI
- About the course
- IC for AI: What is going on?

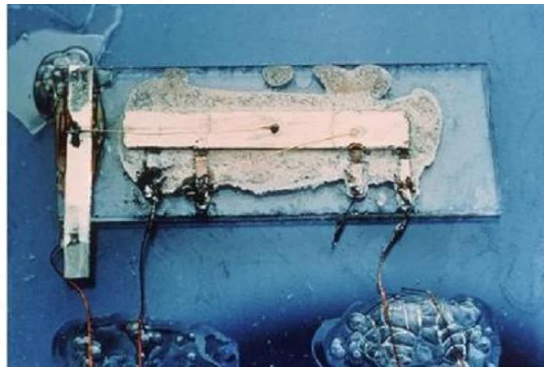


Invention of Integrated Circuit (IC)

- The first IC



Jack Kilby
2000 Nobel Prize



7/16 inch

1/16 inch

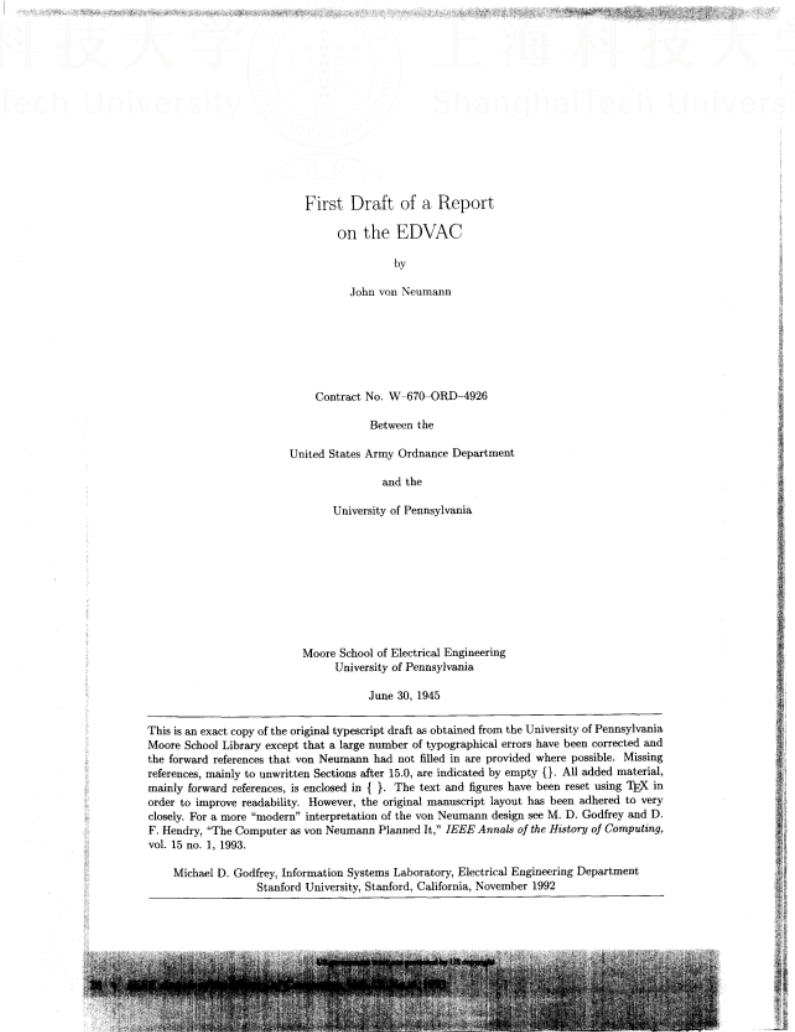
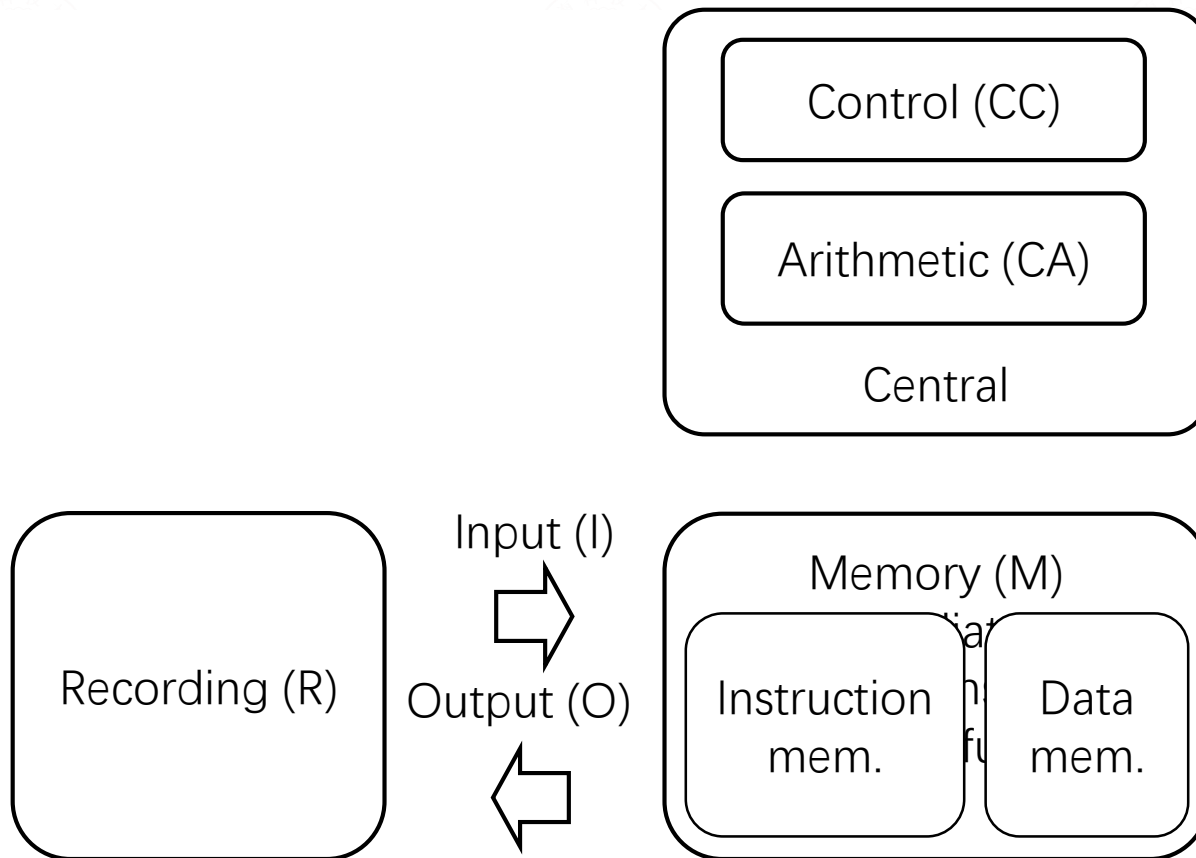
1958



Robert Noyce, 1959
Co-founder of Fairchild and Intel

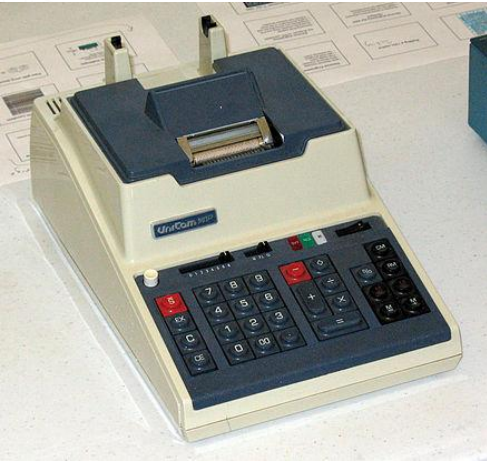
Invention of Modern CPU

- Von Neumann/Harvard architecture

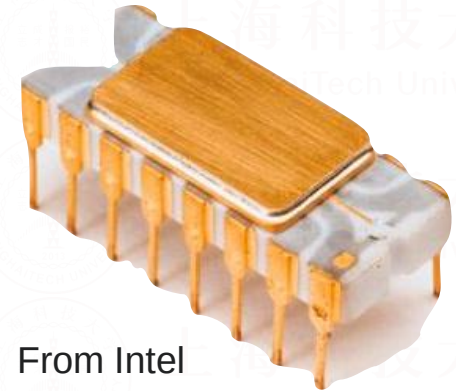


Invention of Modern CPU (Cont'd)

- The first microprocessor: Intel 4004, 1971 = IC + Harvard architecture



The first delivery of a fully operational 4004 was in March 1971 to Busicom for its 141-PF printing calculator engineering prototype (now displayed in the **Computer History Museum** in Mountain View, California).

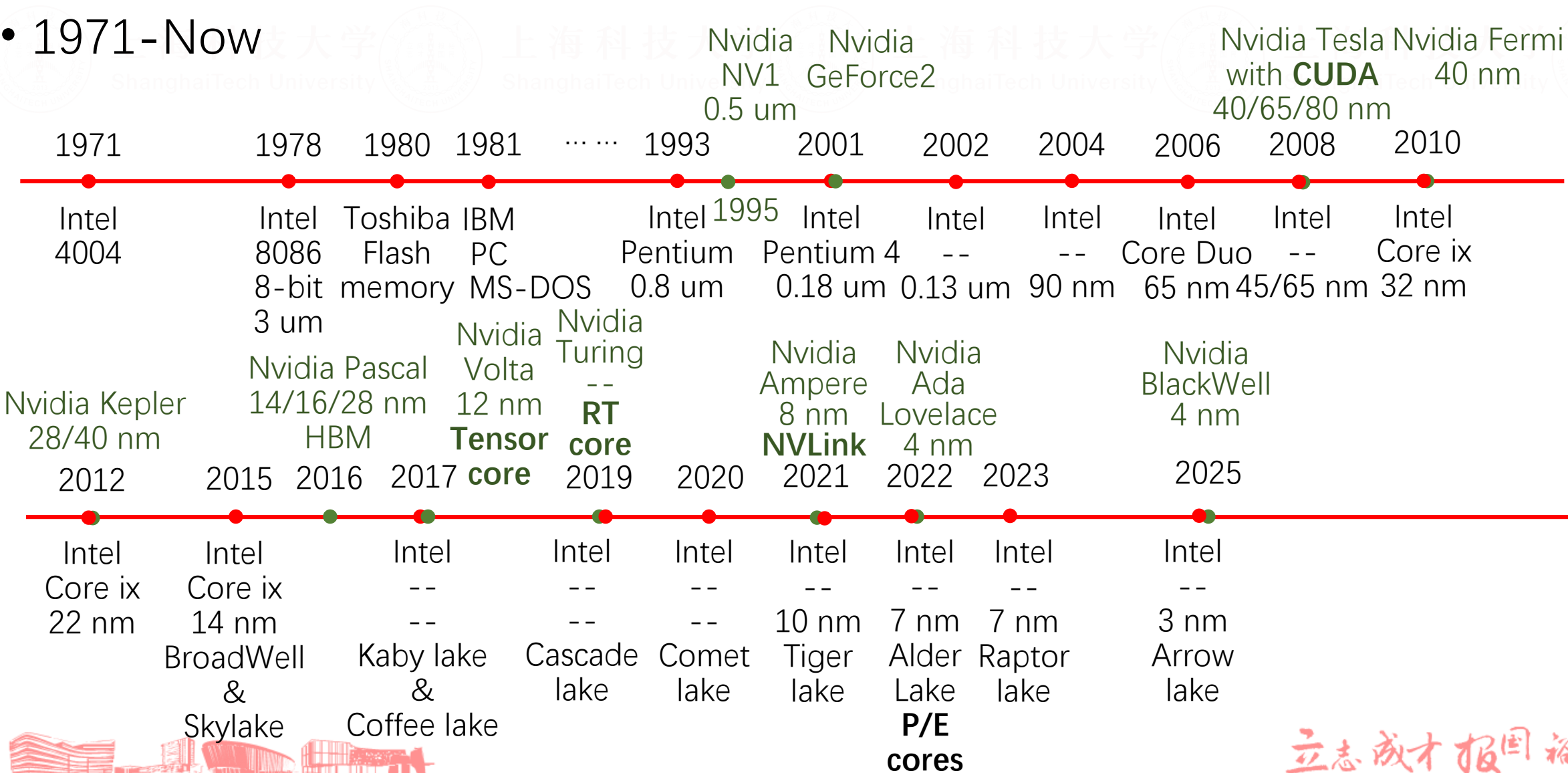


From Intel



Advancing Computing Power

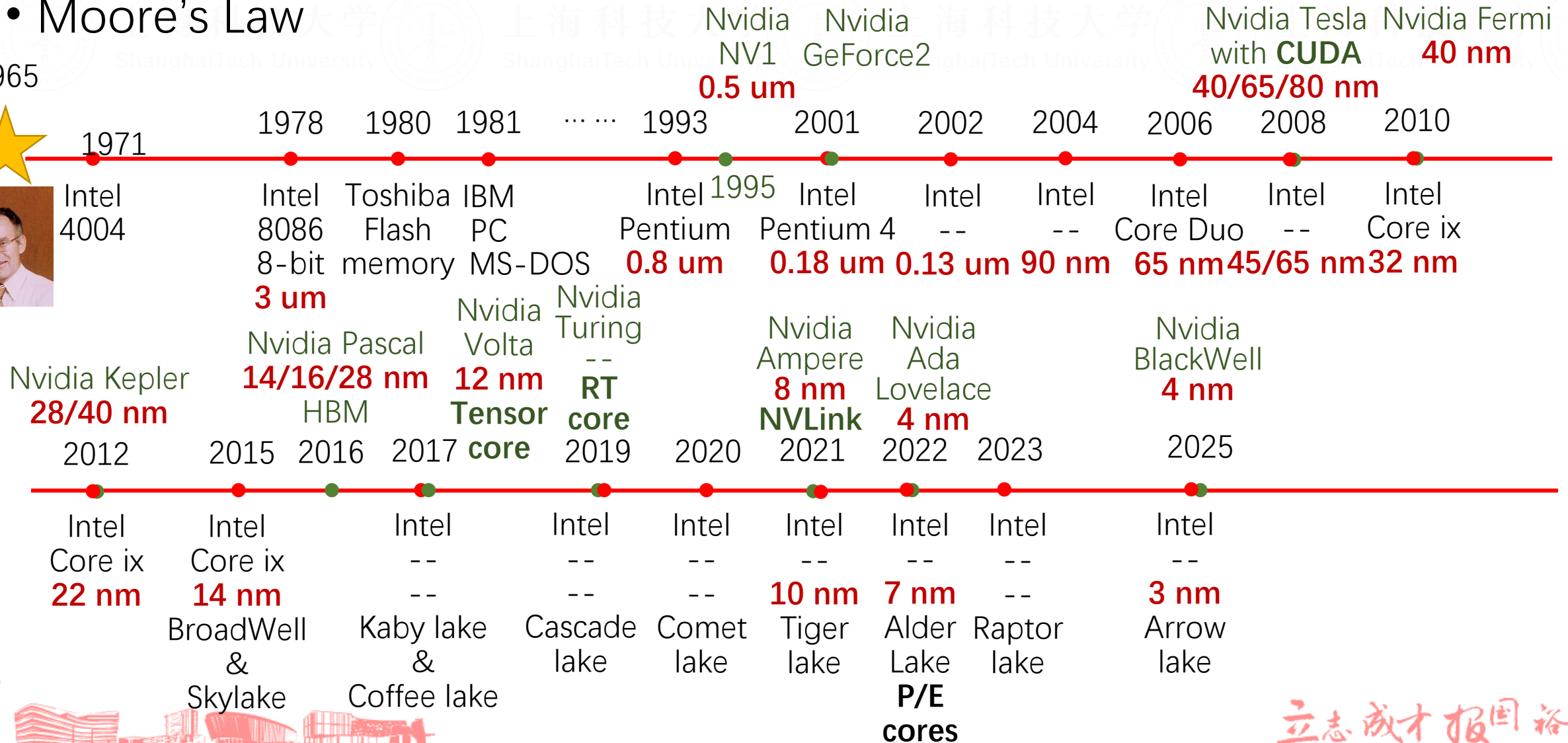
- 1971-Now



What Makes Compute Chips Tick

• Moore's Law

1965



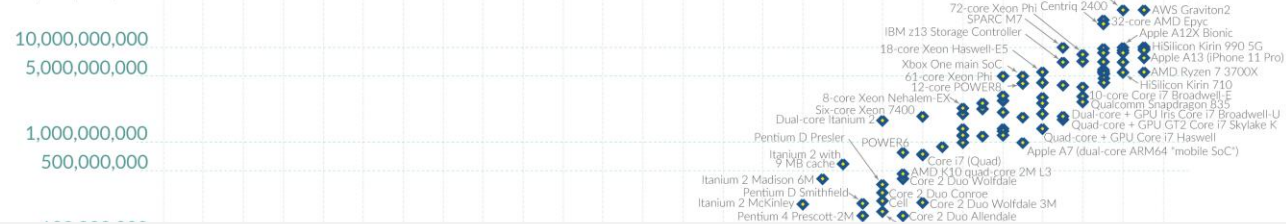
Moore's Law

Moore's Law: The number of transistors on microchips doubles every two years

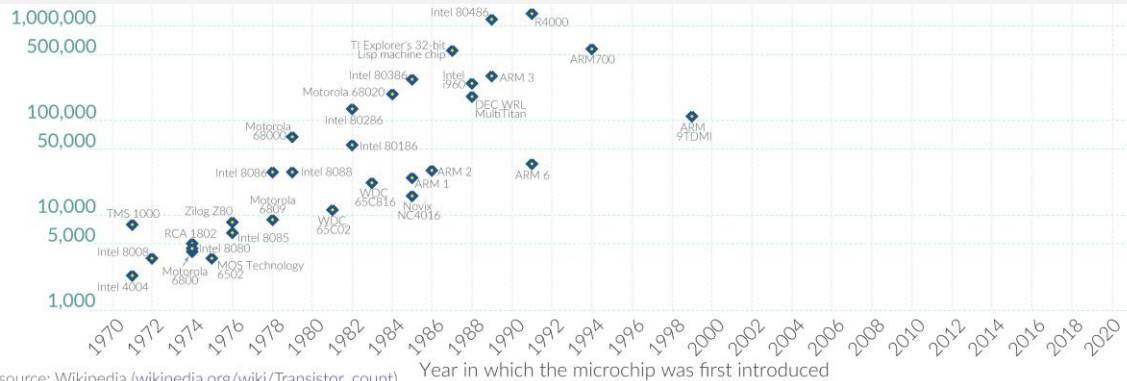
Moore's law describes the empirical regularity that the number of transistors on integrated circuits doubles approximately every two years. This advancement is important for other aspects of technological progress in computing – such as processing speed or the price of computers.

Transistor count

50,000,000,000



**Predict based on observation:
2X Transistors / chip every 2 years (18 months)**



Data source: Wikipedia (wikipedia.org/wiki/Transistor_count)

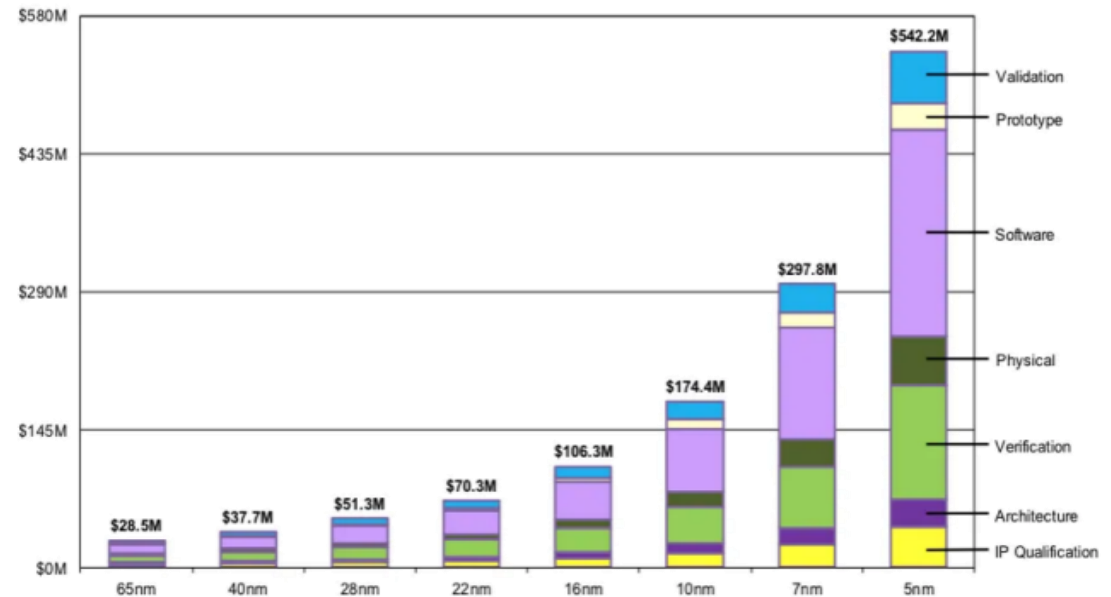
OurWorldinData.org - Research and data to make progress against the world's largest problems.

Licensed under CC-BY by the authors Hannah Ritchie and Max Roser.

Our World
in Data

Design cost at different technology nodes.

Data source: Handel Jones, IBS, 2018.

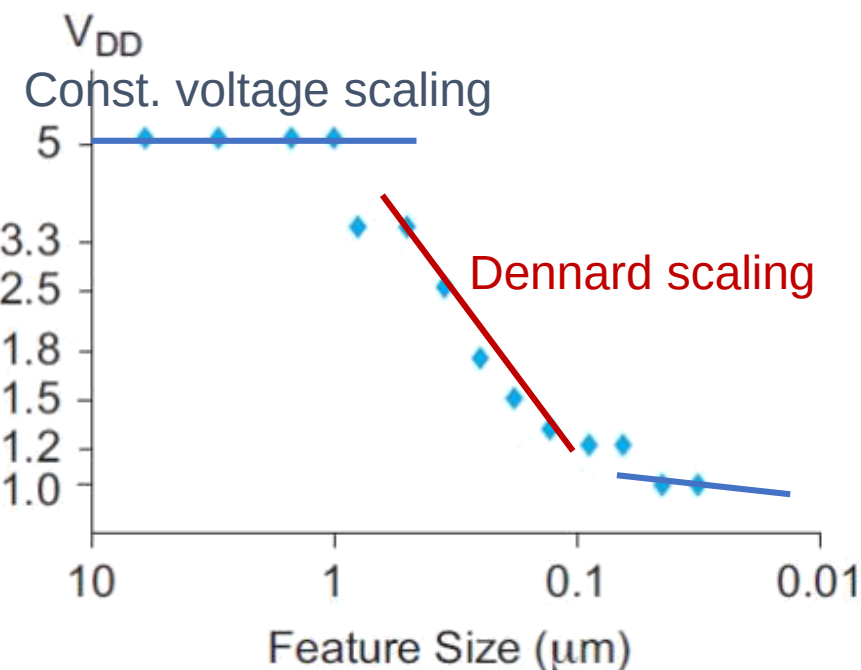


Exponentially increasing cost also, including equipment, development, double every 4 years
a.k.a, **Moore's second law**

立志成才报国裕民

Dennard Scaling

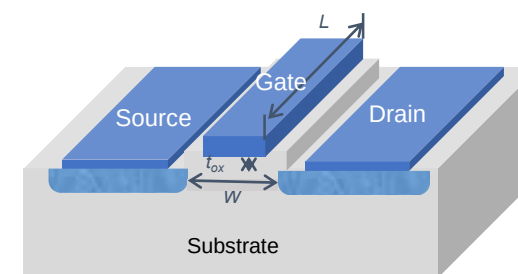
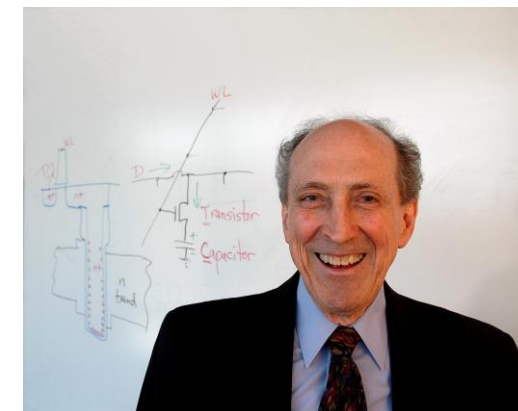
- “**Dennard scaling**, also known as MOSFET scaling, is a scaling law which states roughly that, as transistors get smaller, **their power density stays constant**, so that the power use stays in proportion with area; both voltage and current scale (downward) with length. The law, originally formulated for MOSFETs, is based on a 1974 paper co-authored by **Robert H. Dennard**, after whom it is named” from Wikipedia.



Scale of a single transistor

Gate capacitance	$C_g \propto \frac{WL}{t_{ox}} \propto S$
Saturation current	$I_{on} \propto \frac{W}{t_{ox}L} V^2 \propto \frac{1}{S}$
On resistance	$R_{on} \propto \frac{V}{I_{on}} \propto S$
Intrinsic delay	$t_d \propto R_{on}C_g \propto S^2$
Dynamic Power	$P \propto fCV^2 \propto \frac{1}{S}$
Power density	$P/A \propto \frac{1}{S^3}$

Scaling factor: $S = \frac{1}{\sqrt{2}}$



Dark Silicon

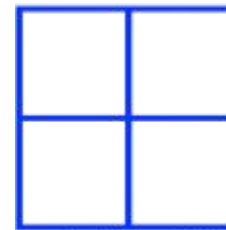
- Dark silicon: the amount of circuitry of an integrated circuit that cannot be powered-on at the nominal operating voltage for a given thermal design power (TDP) constraint.
- TDP: The thermal design power (TDP), is the **maximum amount of heat** generated by a computer chip or component (often a CPU, GPU or system on a chip) that **the cooling system in a computer is designed to dissipate** under any workload.

Utilization Wall: Dark Silicon's Effect on Multicore Scaling

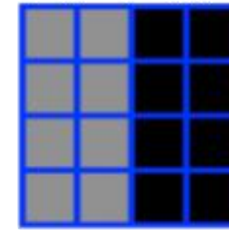
Spectrum of tradeoffs
between # of cores and
frequency

Example:
65 nm $\rightarrow$ 32 nm ($S = 2$)

4 cores @ 1.8 GHz

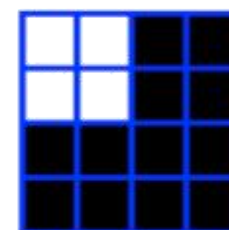


65 nm



2x4 cores @ 1.8 GHz
(8 cores dark, 8 dim)

(Industry's Choice)



4 cores @ 2x1.8 GHz
(12 cores dark)

75% dark after 2 generations;
93% dark after 4 generations

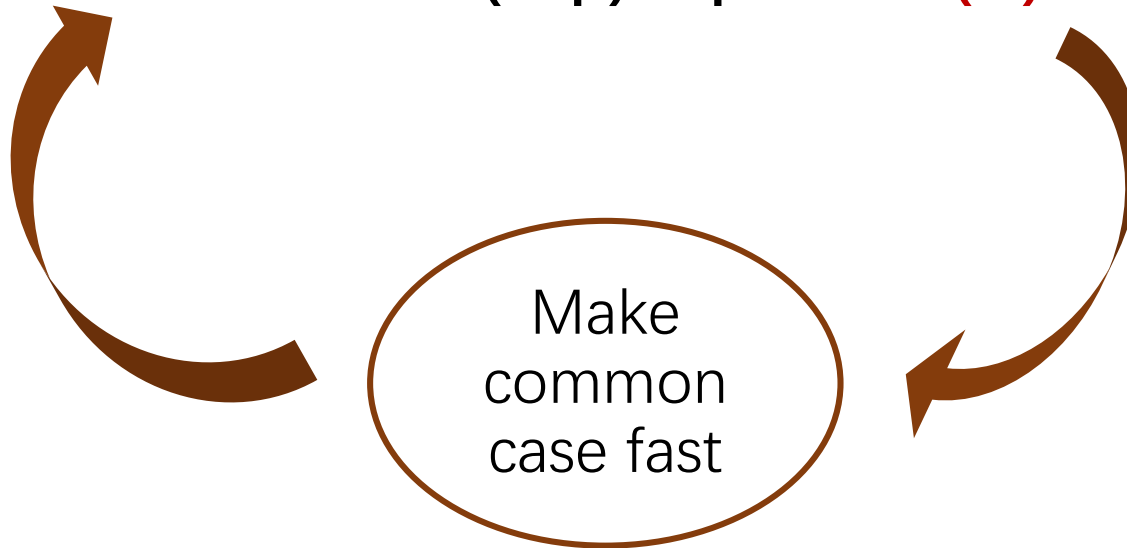
[1] M. B. Taylor, "Is dark silicon useful? Harnessing the four horsemen of the coming dark silicon apocalypse," DAC Design Automation Conference 2012, pp. 1131- 1136.

[2] https://en.wikipedia.org/wiki/Thermal_design_power [3] https://en.wikipedia.org/wiki/Dark_silicon

Parallelism

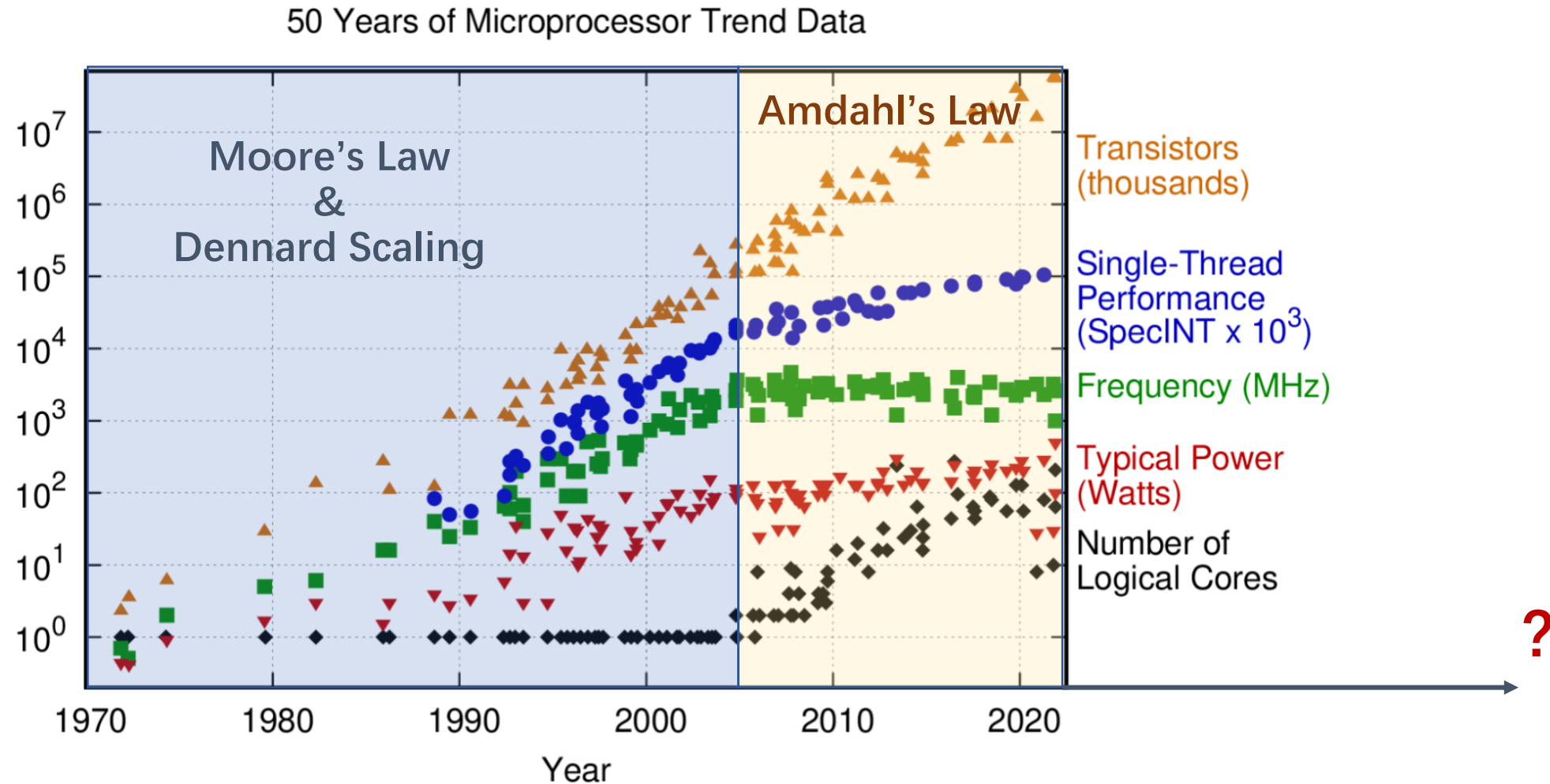
- Amdahl's Law

$$\text{Speedup} = \frac{1}{(1-p) + p/N + C(N)}$$



Gene Amdahl
Computer Pioneer

50 Years of Microprocessor Trend



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 2010-2021 by K. Rupp

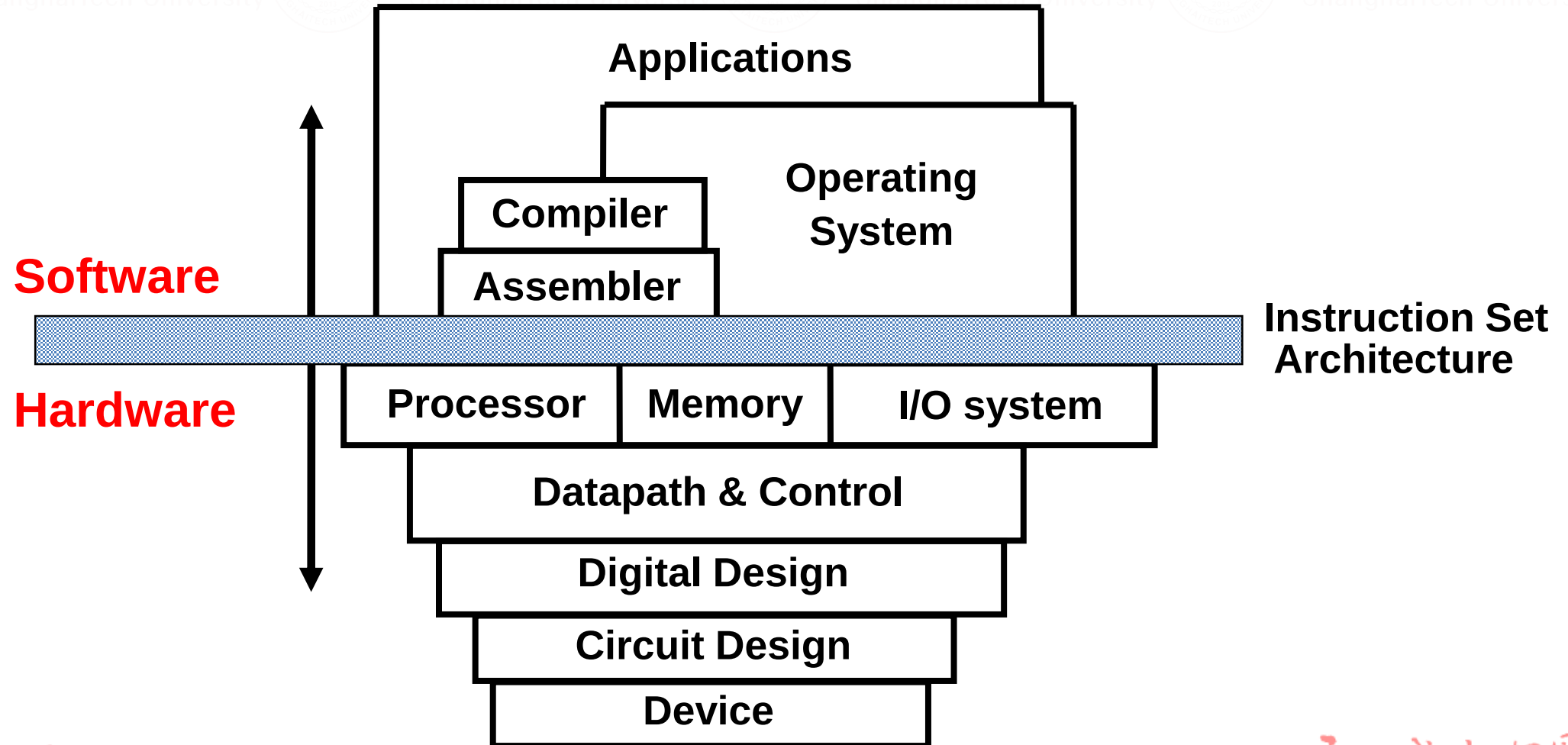
Old Wisdom

- Moore's Law + Dennard Scaling = faster, cheaper, lower-power general-purpose computers each year
- In glory days, 1%/week performance improvement!
- Dumb to compete by designing specialized computers
- By time you've finished design, next generation of general-purpose will beat you

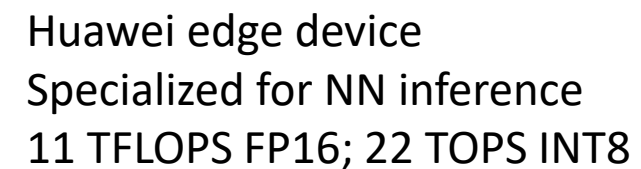
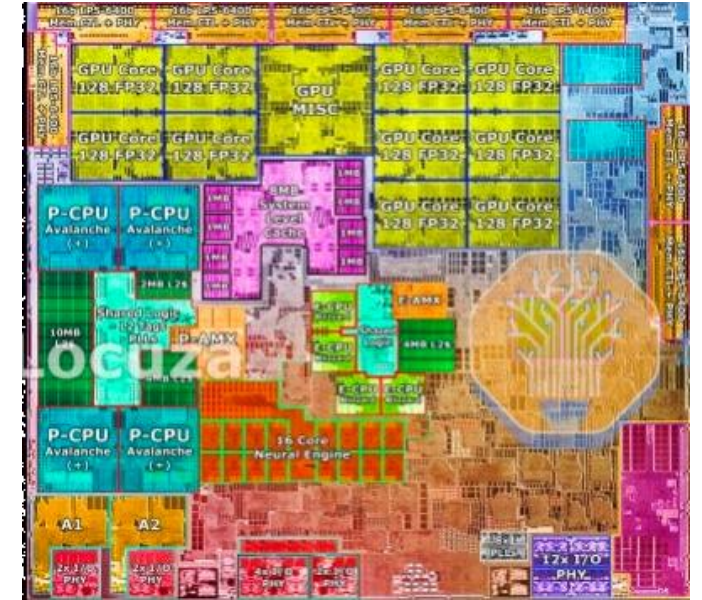
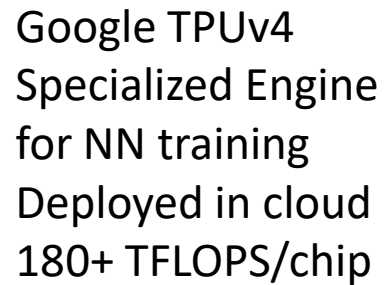


Golden Age of Compute Chips

- Abstraction of computers

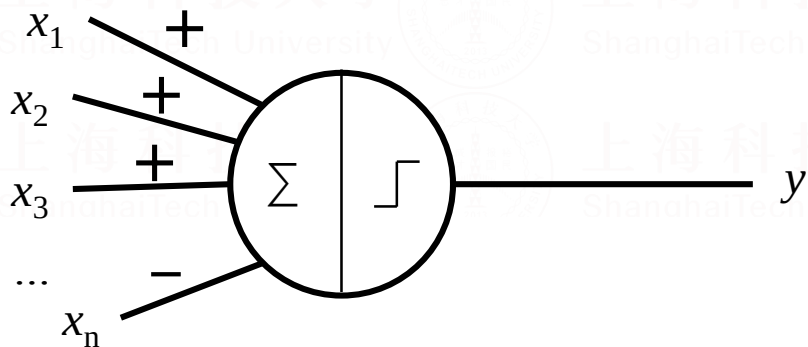


- Heterogonous
- Highly parallel
- Application-specific IC (ASIC)

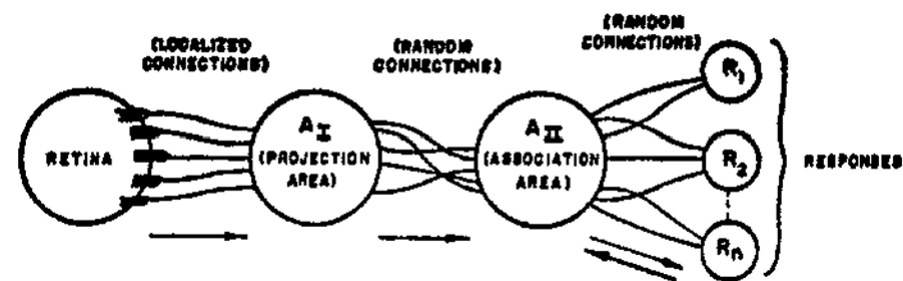


AI Evolution

- Perceptron



McHulloch-Pitts neuron model.



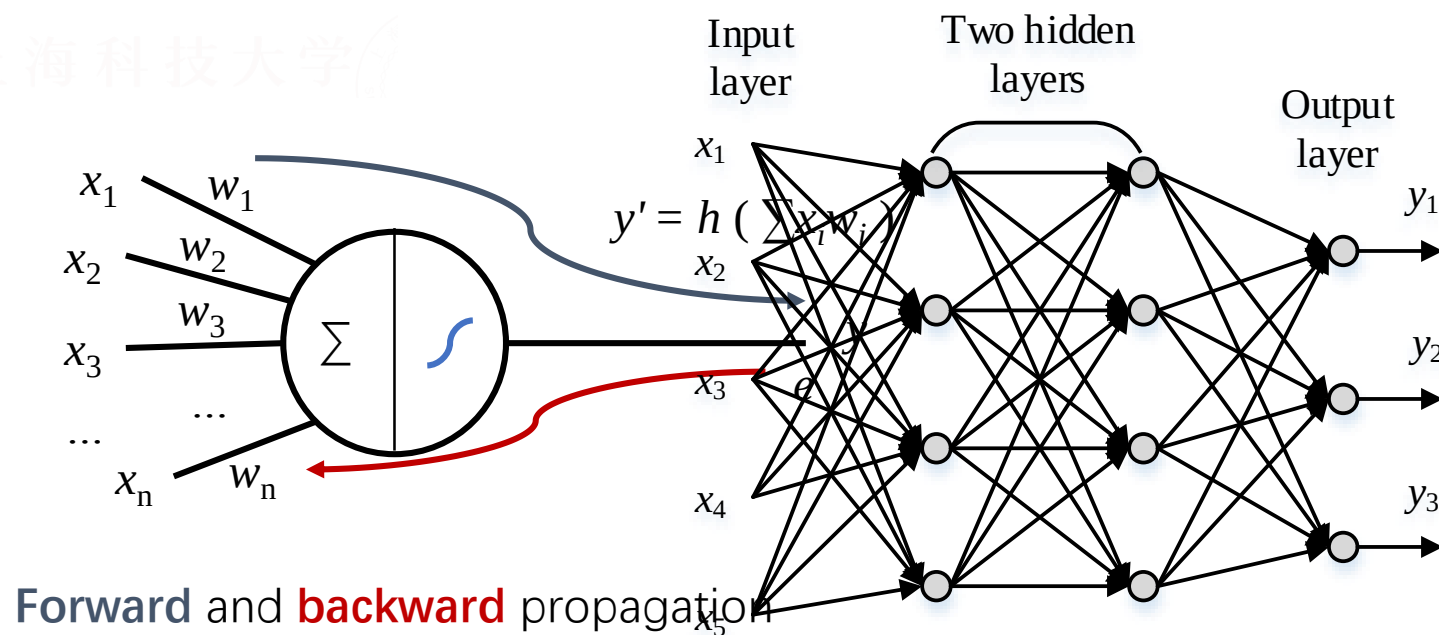
Organization of a perceptron.

Rosenblatt, Frank. "The perceptron: a probabilistic model for information storage and organization in the brain." Psychological review 65.6 (1958): 386.

McCulloch, Warren S., and Walter Pitts. "A logical calculus of the ideas immanent in nervous activity." The bulletin of mathematical biophysics 5.4 (1943): 115-133.

AI Evolution (Cont'd)

- Backpropagation

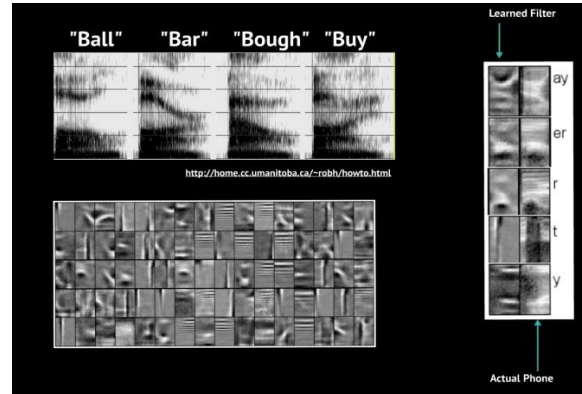
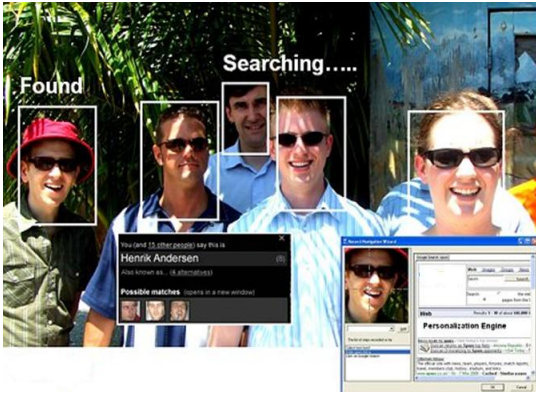


Rumelhart, D., Hinton, G. & Williams, R. Learning representations by back-propagating errors. Nature 323, 533–536 (1986).

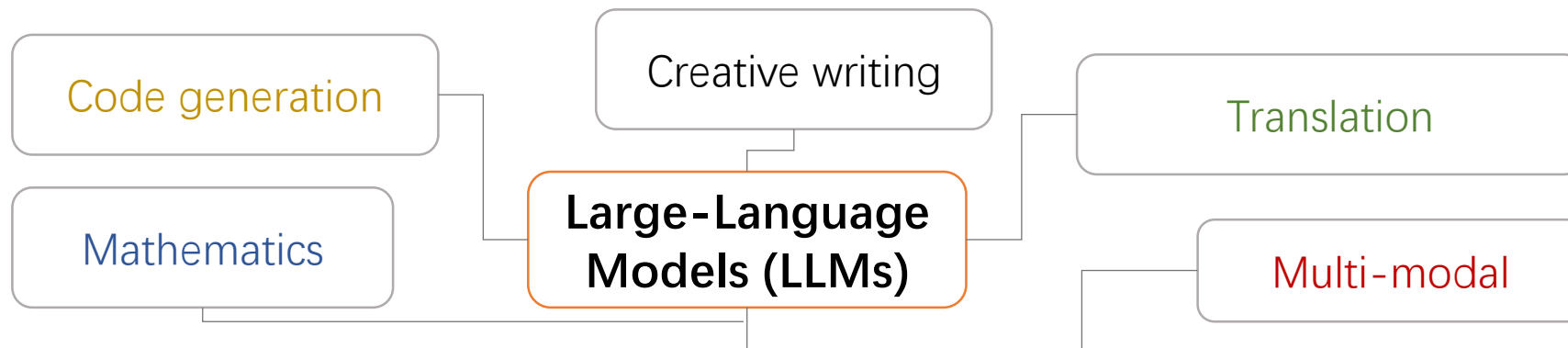
Neural Networks, Deep Learning and Beyond

- Accuracy outperforms human beings in many aspects.

October 2015: 1,202 CPUs and 176 GPUs
March 2016: 1,920 CPUs and 280 GPUs

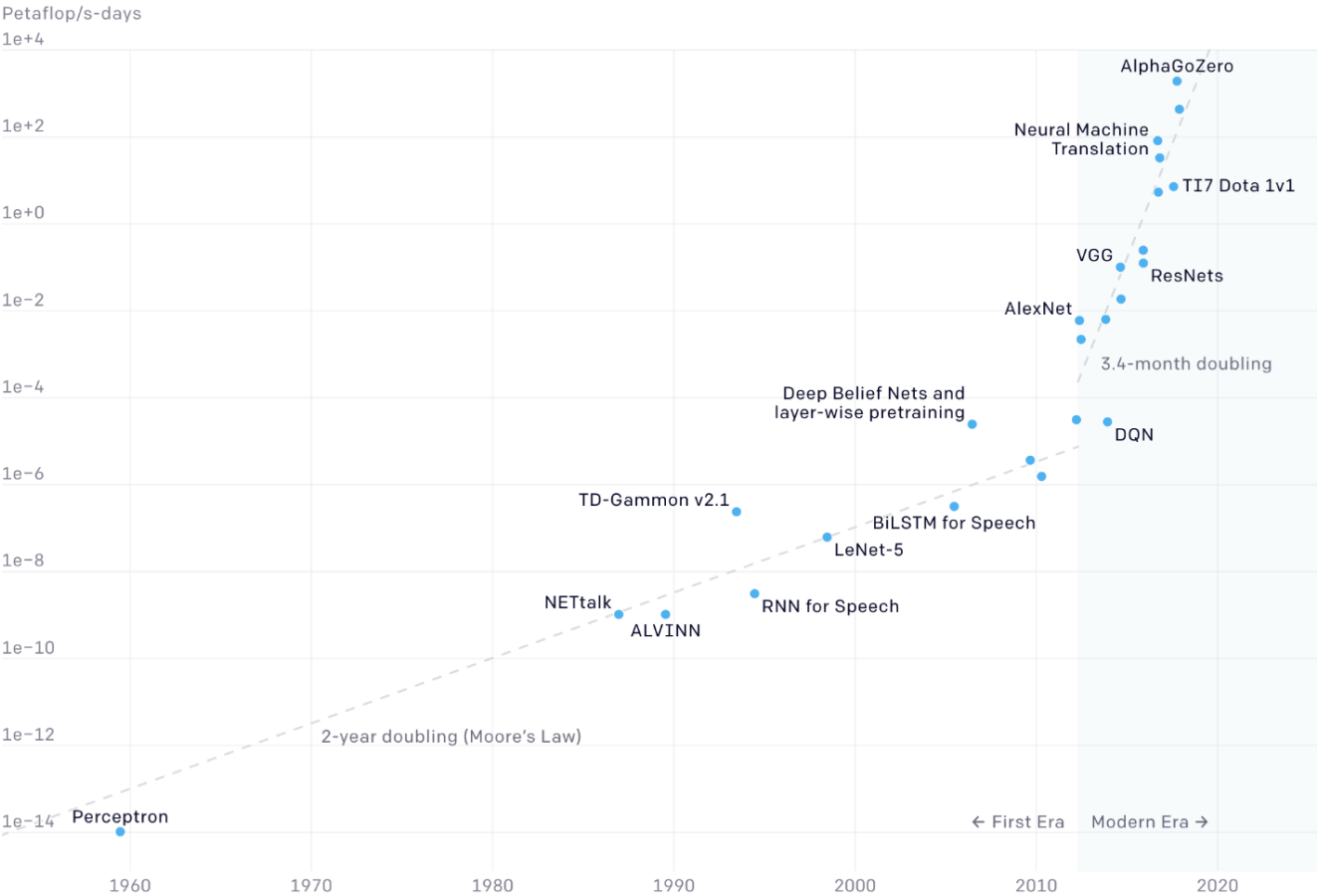


Multilayer perceptron → Convolutional neural networks → Recurrent neural networks → Reinforcement learning

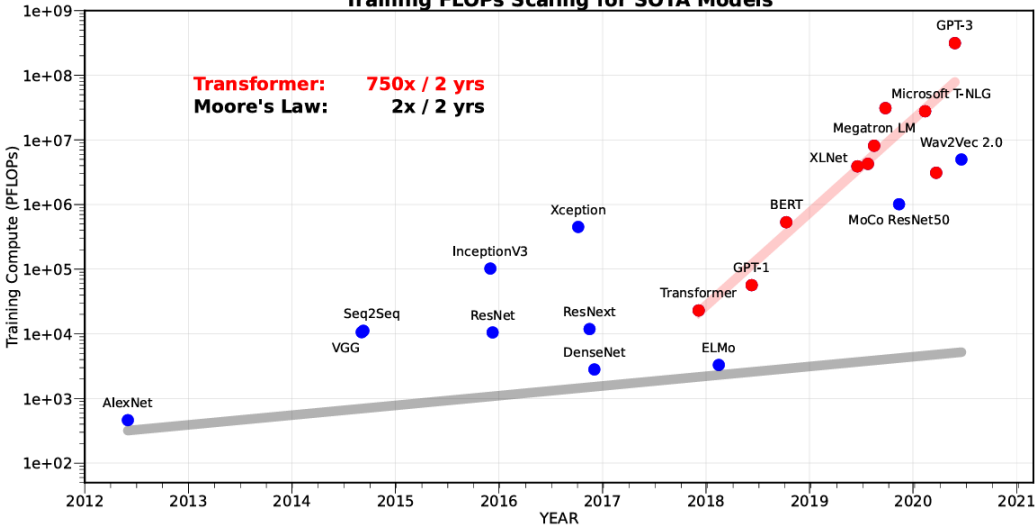


AI Evolution (Cont'd)

Two Distinct Eras of Compute Usage in Training AI Systems



Training FLOPs Scaling for SOTA Models

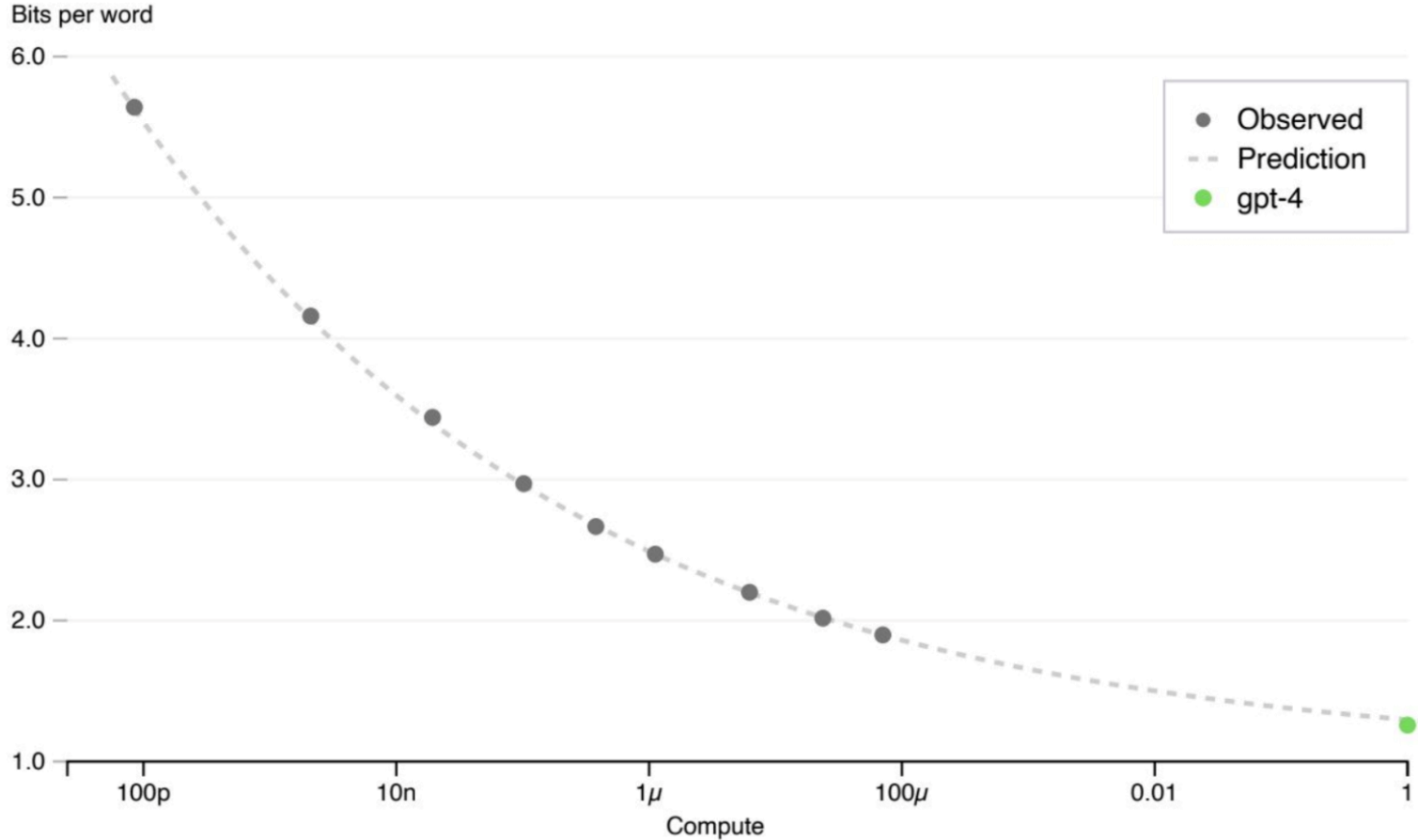


Gholami A, Yao Z, Kim S, Mahoney MW, Keutzer K. AI and Memory Wall. RiseLab Medium Blog Post, University of California Berkeley, 2021, March 29.

Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton. "Imagenet classification with deep convolutional neural networks." Advances in neural information processing systems 25 (2012). Cited nearly 160,000 times as of 2026-09.

Scaling Law for AI Models

OpenAI codebase next word prediction



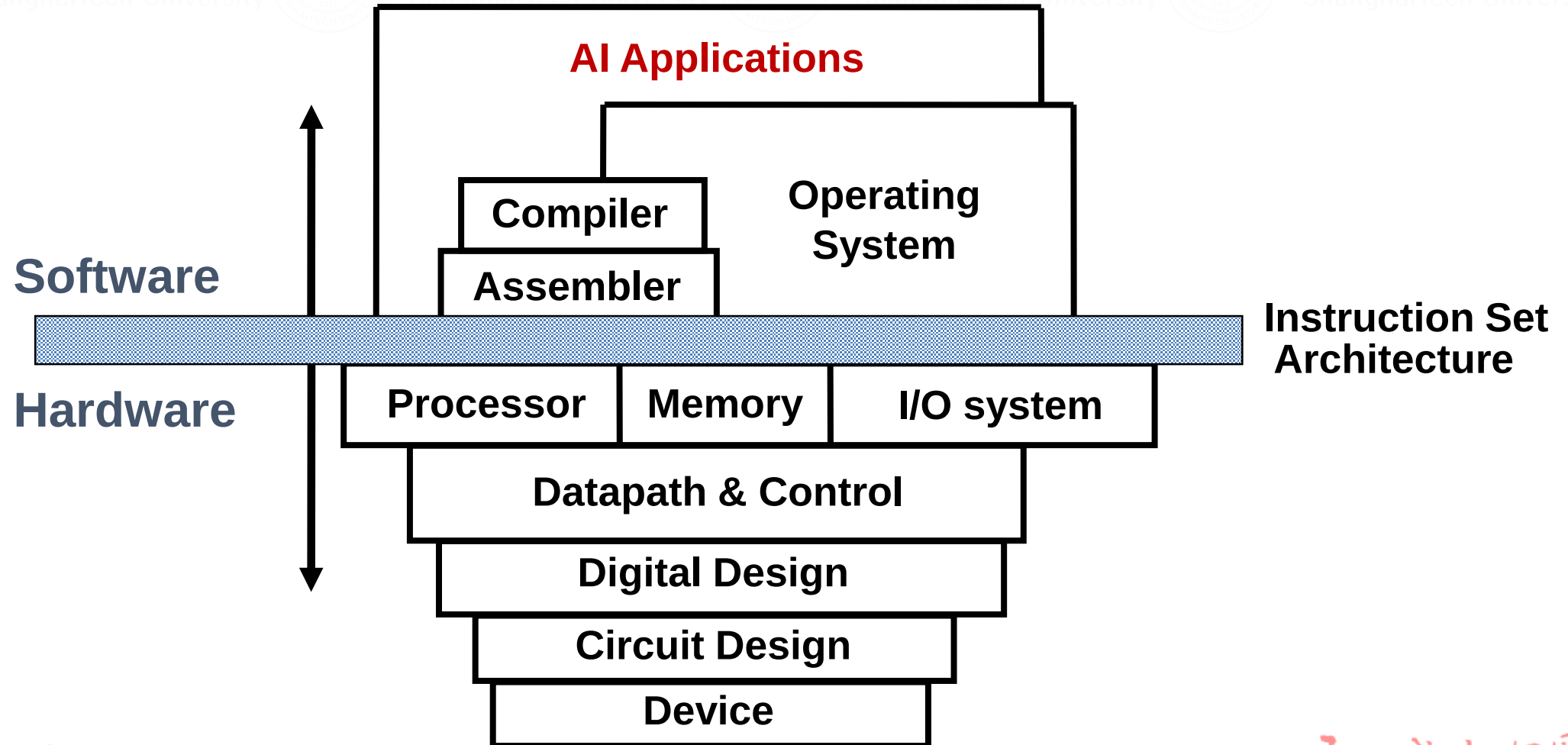
T. Cai, "Predictable Scaling and Infrastructure," IEEE Hot Chips 36 Symposium (HCS), Stanford, CA, USA, 2024,

Bull case for AI compute

- Compute scaling has been predictable and looks to continue.
- Intelligence drives inference demand.
- Technology and economics are ripe for scale this decade.

Golden Age of AI Computing Systems

- Abstraction of computers



About this course

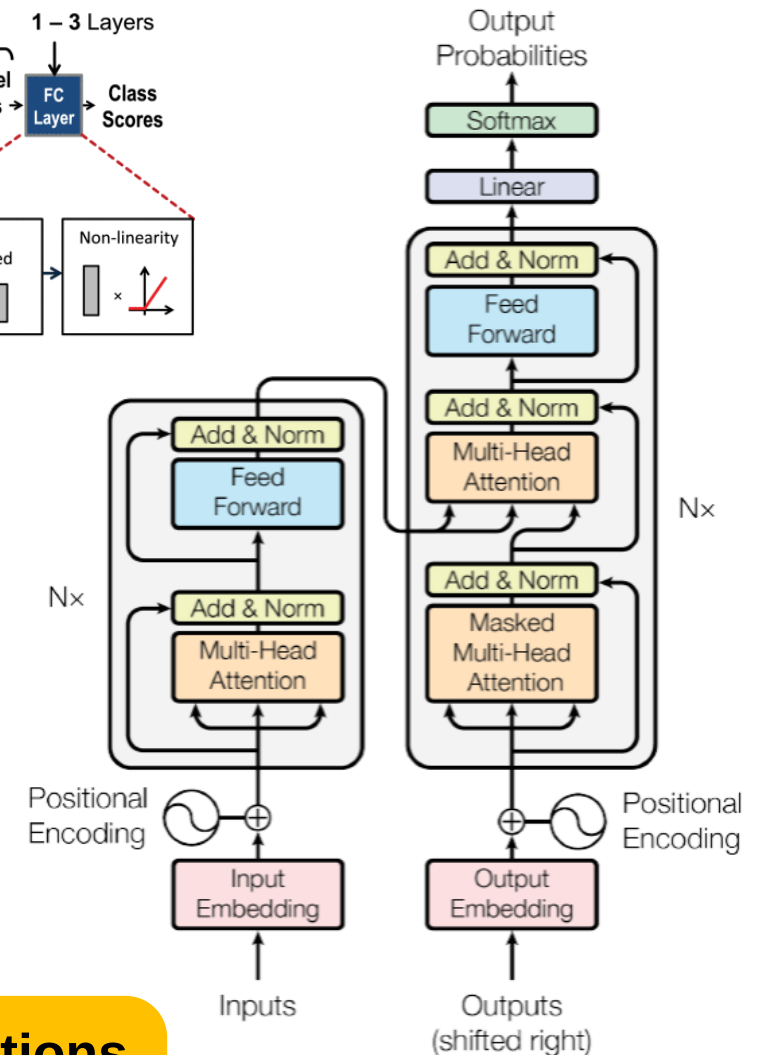
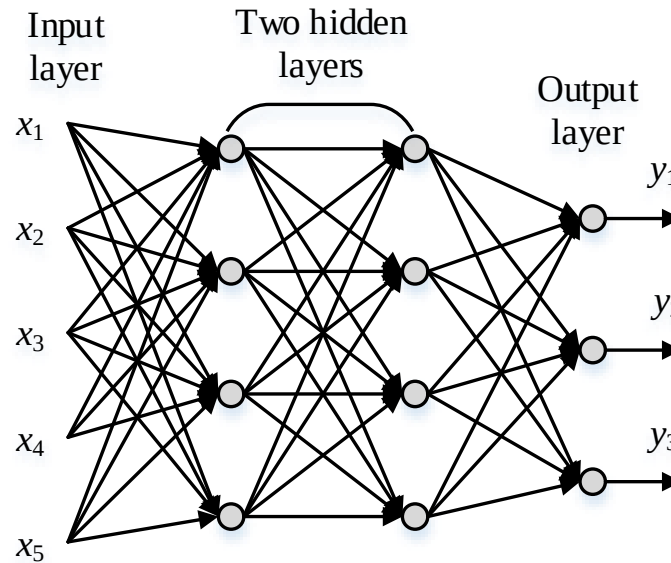
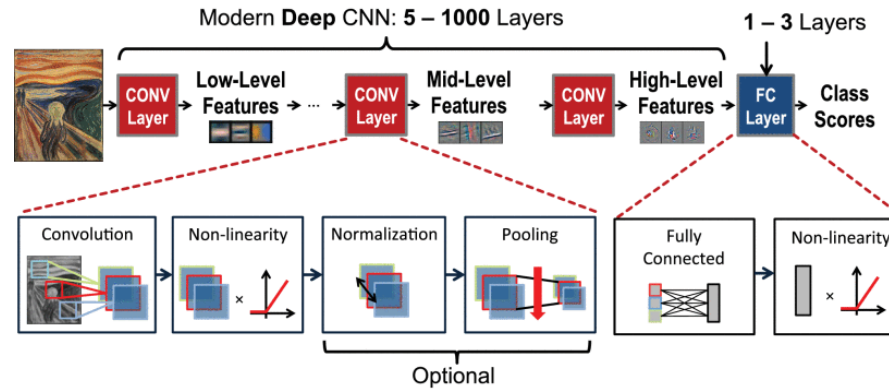
Overview: AI Computing Systems - primarily focus on neural-network hardware accelerator design from a broad architectural perspective

- Basic neural network knowledge (MLP/CNN/LSTM/attention, etc.);
- Model compression (quantization/pruning);
- Kernel computation (algorithms by transformation);
- Datapath design (systolic array/adder tree/special function unit);
- Vector processor (SIMD) & roofline model;
- Memory hierarchy;
- Design cases;
- Programming language, software stack and beyond;
- Topics on compute-in-memory (CIM), sparsity, neuromorphic, etc.



AI Computing Systems: Algorithm-side

- Artificial intelligence (AI)
- Machine learning (ML)
- Neural Networks (NNs)
- Deep NNs (DNNs)
- Attention mechanism
- Emerging technologies (Spiking NNs, etc.)
- Hardware-aware optimization, e.g., quantization and pruning



Focus on the practical implementations rather than the theory.

AI Computing Systems: Hardware-side

- Datapath & controller for specific NN models (dataflow, data reuse, locality)
- Computer architecture (basics, overview, RISC-V and customization)
- Parallelism (ILP, SIMD, **roofline model**)
- Memory consideration (decoupled execute/access architecture)
- Hardware/algorithm codesign
- Design examples (CPU, GPU, FPGA, TPU, NVDLA, etc.)



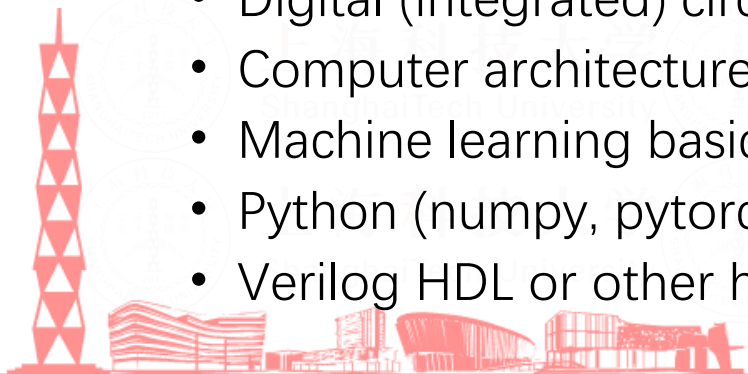
AI Computing Systems: Goal

- Think from an architectural perspective
- General knowledge to design a domain/application-specific IC
- Hardware/software codesign and optimization techniques
- Creative thinking, critical thinking, reading/writing & presentation skills, etc.



Course Info.

- ShanghaiTech SIST and SISI (创芯学院) share most of the course materials. Course website:
<https://faculty.sist.shanghaitech.edu.cn/liust/EE219/EE219.html>
- Lectures (1-12 week, 13-16 week for project and group presentation)
 - **ShanghaiTech Teaching center 201**: Tue./Thur. 1500pm-1640pm;
 - **SISI: SISI building room 307**: Wed. 1015am-1155am; Fri. 0815am-0955am;
- Office hour: by appointment or online through Piazza
- Prerequisite
 - Programming: C/C++, assembly
 - Digital (integrated) circuit
 - Computer architecture basics (recommended)
 - Machine learning basics (recommended)
 - Python (numpy, pytorch, matplotlib etc.)
 - Verilog HDL or other hardware description language



Course Grading

Evaluation (100% in total)

- Labs: tentatively 4-5 labs, 60% of total score;
- Participation: 10%, in-lecture quiz and Piazza participation;
- Group project (20%) and presentation (10%)
- For those who want A+, do as much bonus as you can!



Lab/Project Platform

- Virtual platform <https://vd-new.shanghaitech.edu.cn/> will be used for labs and projects (Ubuntu 22.04, get yourself familiar with Ubuntu!)
- It is used for the third year, so there might be problems. Bonus for who report bugs (email or cc the instructor & TAs)!
- Lab 0 will be released soon. Keep an eye on Piazza/course website.
- Piazza and Gradescope will be used
 - Piazza for class forum
 - Gradescope for lab/project submission



The Teaching Team

- Instructor: Siting Liu (刘思廷)
- Email: liust@shanghaitech.edu.cn;
- B.Eng. & M.Eng. Harbin Institute of Technology;
- Ph.D., University of Alberta, Canada;
- Postdoc, McGill University, MEIL
- Have been teaching *Computer Architecture, Fault-tolerant Computing*
- Research interest: circuits and systems for emerging technology and applications in AI, published on TC/DAC/DATE/TCAD etc.
- Recently focuses on AI Infra, sparse AI accelerators, edge LLM accelerators.



The Teaching Team (Cont'd)

- TAs



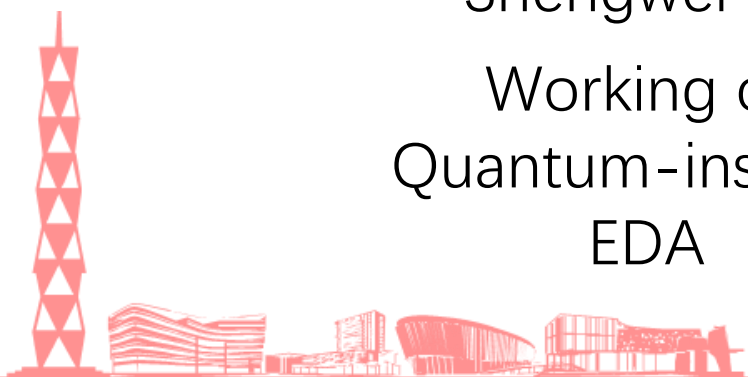
Shengwei Gu

Working on
Quantum-inspired
EDA



Zhilian Hong

Working on edge
LLM deployment



Course Design Philosophy

- The course serves as an introduction to AI computing system field and develop basic skills of AI accelerator/system design
- Feedbacks are very much appreciated



立志成才 报國裕民

Academic Integrity

- Intellectual property (IP), independent work, academic integrity;
- Breaking academic integrity leads to **“0” mark on lab/paper representation/project/quiz, fail of the course or even expulsion**. See <https://sist.shanghaitech.edu.cn/2019/1107/c2842a46321/page.htm> & <https://openinfo.shanghaitech.edu.cn/2015/0706/c87a271/page.htm> for more details;
- AI-generated code is allowed. But please submit a report stating how AI helps with your assignment, specify what the AI agent/LLM you use is.
- **Giver and receiver are treated equally!**



Learning Materials

Learning python (pytorch, numpy, matplotlib)

- Python: <https://www.runoob.com/python3/python3-tutorial.html>
- Pytorch: <https://pytorch.org/tutorials/beginner/basics/intro.html>
- Numpy: <https://numpy.org/numpy-tutorials/content/tutorial-svd.html>;
- <https://numpy.org/numpy-tutorials/content/mooreslaw-tutorial.html>
- #Matplotlib: <https://matplotlib.org/stable/gallery/index.html>

Learning Verilog HDL

- <http://www.asic-world.com/verilog/veritut.html>
- <https://www.runoob.com/w3cnote/verilog-tutorial.html>
- Verilog 设计示例 (intel.cn)

Reference courses

- <https://cs217.stanford.edu/>
- <https://inst.eecs.berkeley.edu/~ee290-2/sp20/> (Lecture videos available at <https://www.bilibili.com/video/av545145111> Sophia Shao, UC Berkeley)
- <https://elearning.fudan.edu.cn/courses/26051> (Lecture videos available at <https://www.bilibili.com/video/BV1ff4y1X7kP> 陈迟晓, Fudan)
- <https://www.bilibili.com/video/BV1WE411A7tv> (陈云霁, UCAS)
- <https://space.bilibili.com/1663273796/channel/collectiondetail?sid=499979> (陈天奇, CMU)



立志成才 报国裕民

Reading Materials & Books

- 《智能计算系统第二版》陈云霁等。
- Computer Organization and Design RISC-V Edition: The Hardware Software Interface
- Computer Architecture: A Quantitative Approach, 6th Edition. By John Hennessy, David Patterson.
- V. Sze, T.-J. Yang, Y.-H. Chen, J. Emer, "Efficient Processing of Deep Neural Networks: A Tutorial and Survey," Proceedings of the IEEE, vol. 105, no. 12, pp. 2295-2329, December 2017.
- <http://eyeriss.mit.edu/tutorial.html>
- <https://media.nips.cc/Conferences/2015/tutorialslides/Dally-NIPS-Tutorial-2015.pdf>
- Papers from top-tier conferences
 - Computer architecture: ISCA, HPCA, MICRO, ASPLOS, Hot Chips, etc.
 - EDA: DAC, DATE, ICCAD, etc.
 - Machine learning (not very "hardware"): NeurIPS, ICML, ICLR, CVPR, etc.
 - Circuits & systems: ISSCC, ISCAS, JSSCC (journal), etc.
 - New conferences: MLSys, AICAS, etc.



Q & A



立志成才 报國裕民