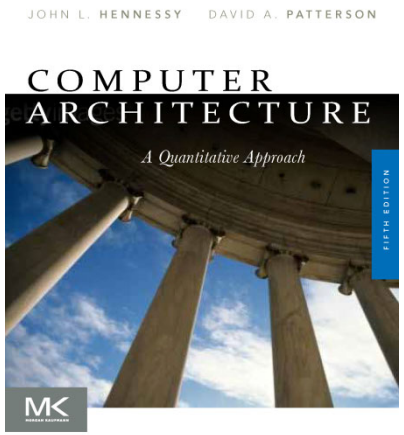


ESE 545 Computer Architecture

General Introduction to the Class and its Project

Some people may find images in this presentation distressing.

How not to become extinct as an engineer in the age of AI

These poor guys failed to learn & adapt.

About our class

- **Goal: Develop abilities for creativity, innovation, and learning** in the context of computer architecture, hardware & software design.
- **How: The emphasis is on Design**, not on software/hardware coding, neither on training you on the patterns in data called previous exams because (I'm assuming) you are not (learning like) AI bots. Check how better than AI you are **now**:

  What is common between a shoe  **and a pencil** **?**

- **The Way: Respect yourself** and do the work required in the class honestly.
- (Almost) everyone can cheat/fake, and, even if not everyone is caught on cheating immediately, I believe that once a cheater, always a cheater.

  The way how you do anything is the way how you do everything.

- **Why**: Finally, I want you to be able to write your resume proudly demonstrating your knowledge, technical skills, and abilities for critical thinking and innovation, i.e., answering the key question

  Why should someone hire & keep you, not another guy nor AI?

Let's do it.

Principal IC Design Engineer

This position requires work on site at Northrop Grumman's McClellan, CA facility.

What You'll get to Do

- Apply your skills as a circuit design engineer in development of cryogenic CMOS circuits for advanced power efficient computing applications.
- Design tasks including schematic capture, circuit simulation, circuit layout, physical verification (LVS, DRC), and parasitic extraction for a variety of analog and mixed signal circuits.
- Work in a dynamic environment with rapidly developing technologies and tools.
- Take designs from concept through modeling, layout and verification and creation of test plans based on circuit requirements.

This position can be filled at the Principal IC Design Engineer OR the Sr. Principal IC Design Engineer level.

Basic Qualifications for Principal IC Design Engineer

- Bachelor of Science STEM (Science, Technology, Engineering, Mathematics) degree with 5 years of experience in Analog, Digital or Mixed Signal Circuit design (3 years with technical MS).
- Experience with IC custom design tools (Cadence Virtuoso) for schematic capture, circuit simulation, and custom layout; physical verification using Assura or Calibre.
- Demonstrated experience in problem solving and analytical skills.
- Demonstrated experience in meeting tight deadlines and milestones according to program schedule.
- Excellent communication skills, able to efficiently disseminate information to leadership and respective working group.

Salary Range: $110,300.00 - $165,500.00

ESE545 Computer Architecture

Instructor: Prof. Mikhail Dorojevets

Office: 243 Light Eng. Bldg.

E-mail: mikhail.dorojevets@stonybrook.edu

Office Hours: Wed 10:00 am – 12:00 pm

Class: Thursday 5:00 - 7:50 pm

Web page: **http://www.ece.stonybrook.edu/~midor/ESE545/index.html**

Text: *Computer Architecture: A Quantitative Approach, 6th Edition* (2019), *ISBN: 978-0-12-811905-1 or 7th edition (2025)*

- **Exam**: There will be one ("late mid-term") exam by email in April.
- Project presentation deadline: The last week of classes.

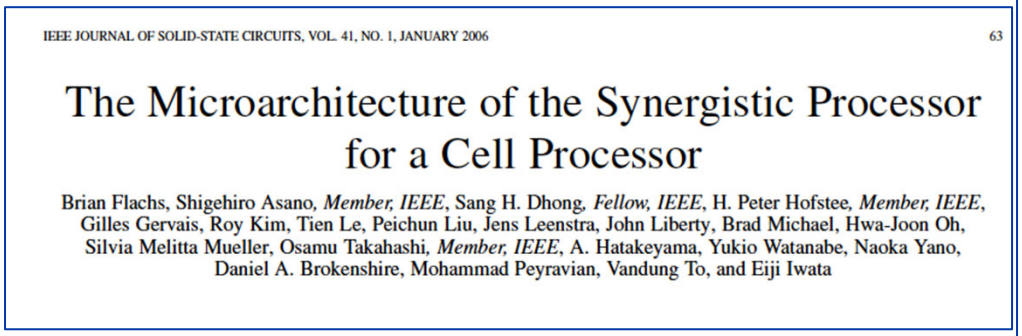
Grading:

Exam : 50%

Project (one/two student teams): 50%

ESE 545 Project

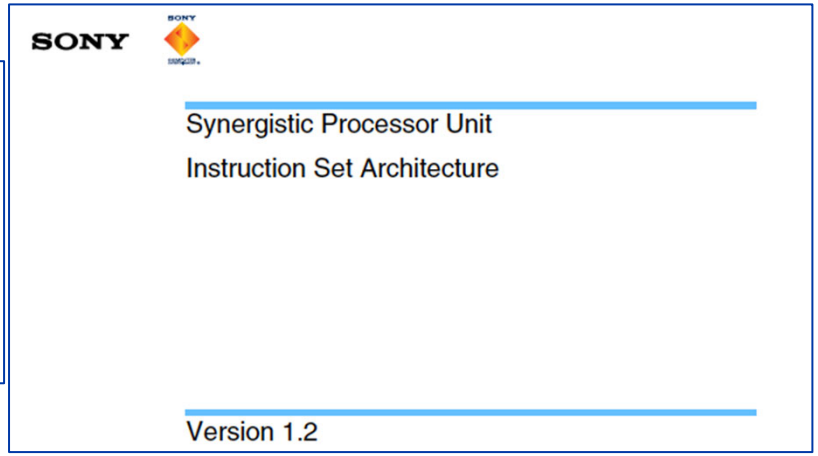
A dual-issue Cell SPU-lite pipelined multimedia processor and its VHDL/Verilog/SystemVerilog model



IEEE JOURNAL OF SOLID-STATE CIRCUITS, VOL. 41, NO. 1, JANUARY 2006 63

The Microarchitecture of the Synergistic Processor for a Cell Processor

Brian Flachs, Shigehiro Asano, *Member, IEEE*, Sang H. Dhong, *Fellow, IEEE*, H. Peter Hofstee, *Member, IEEE*, Gilles Gervais, Roy Kim, Tien Le, Peichun Liu, Jens Leenstra, John Liberty, Brad Michael, Hwa-Joon Oh, Silvia Melitta Mueller, Osamu Takahashi, *Member, IEEE*, A. Hatakeyama, Yukio Watanabe, Naoka Yano, Daniel A. Brokenshire, Mohammad Peyravian, Vandung To, and Eiji Iwata



SONY

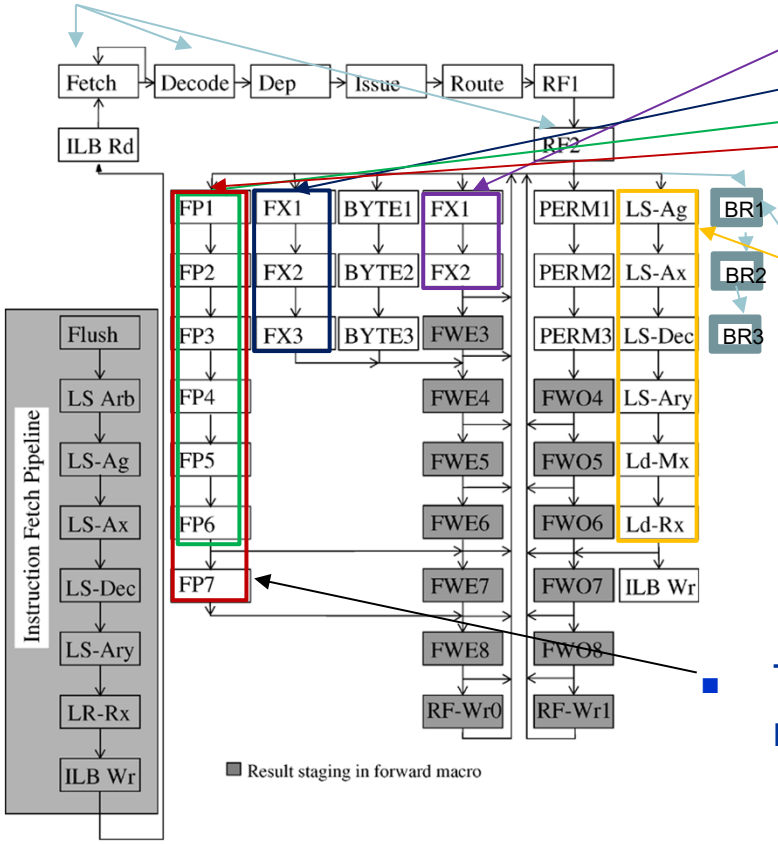
Synergistic Processor Unit

Instruction Set Architecture

Version 1.2

Cell SPU pipeline (16 stages)

Pipeline stages: two instructions/stage in the first 7 stages



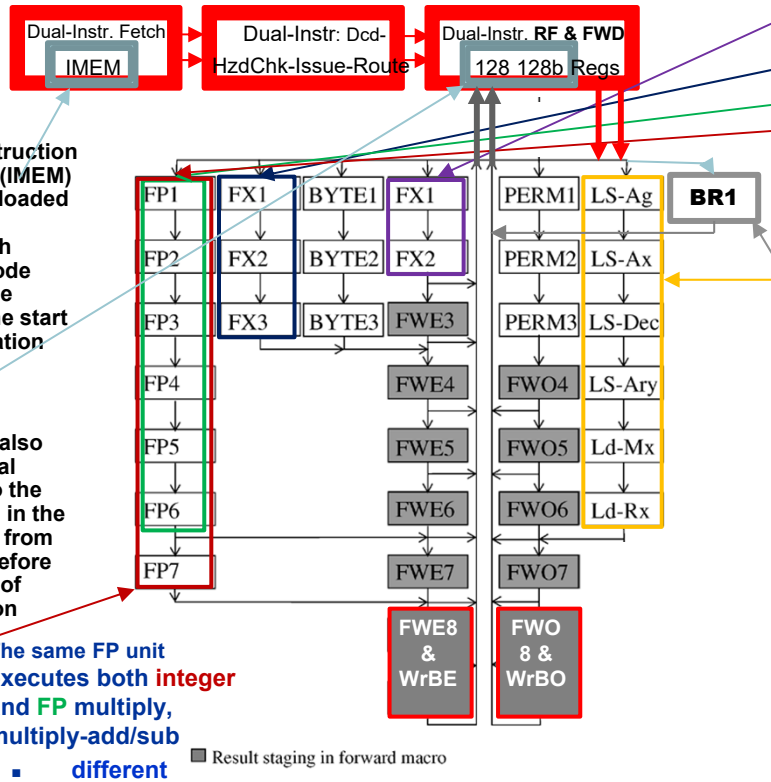
Unit	Instructions	Execution Pipe	Unit Pipeline Depth	Instruction Latency
Simple Fixed 1	word arithmetic, logicals, count leading zeros, selects, and compares	Even	2	2
Simple Fixed 2	word shifts and rotates	Even	3	4
Single Precision	multiply-accumulate	Even	6	6
Single Precision	integer multiply-accumulate	Even	7	7
Byte	pop count, absolute sum of differences, byte average, byte sum	Even	3	4
Permute	Quadword shifts, rotates, gathers, shuffles as well as reciprocal estimate	Odd	3	4
Local Store	Load and store	Odd	6	6
Channel	Channel Read/Write	Odd	5	6
Branch	Branches	Odd	3	4

- The same FP unit executes both **integer** and **FP** multiply, multiply-add/sub
  - different instruction latencies (**7** and **6** cycles) for integer and FP multiply ops

Brian Flachs et al, The Microarchitecture of the Synergistic Processor for a Cell Processor, 2006

ESE 545 SPU-lite Pipeline (11stages)

Pipeline stages: two instructions/stage in the first 3 stages



2 KB Instruction memory (IMEM) must be loaded with the testbench binary code from a file before the start of simulation

You can also load initial values to the registers in the 3rd stage from the file before the start of simulation

The earliest FW stage to forward result

Unit	Instructions	Execution Pipe	Unit Pipeline Depth	Instruction Latency
Simple Fixed 1	word arithmetic, logicals, count leading zeros, selects, and compares	Even	2	X 3
Simple Fixed 2	word shifts and rotates	Even	3	4
Single Precision	multiply-accumulate	Even	6	X 7
Single Precision	integer multiply-accumulate	Even	7	X 8
Byte	pop count, absolute sum of differences, byte average, byte sum	Even	3	4
Permute	Quadword shifts, rotates, gathers, shuffles as well as reciprocal estimate	Odd	3	4
32 KB Local Store (data only)	Load and store	Odd	6	X 7
Branch	Branches	Odd	1	2

- The same FP unit executes both integer and FP multiply, multiply-add/sub
  - different instruction latencies (7 and 6 cycles) for integer and FP multiply ops

Brian Flachs et al, The Microarchitecture of the Synergistic Processor for a Cell Processor, 2006

SPU-lite Instruction Set Requirements

The chosen SPU-lite sub-set of the SPU instructions must include:

- Instructions for all units
- RISC-style (e.g., MIPS-like) basic (core) instructions for Simple Fixed 1&2, FP, Local Store, and Branch units
  - No hint for branch instructions
  - Include No-op for even and odd pipes plus stop instructions
- Several types of data objects
  - including SP floating-point, integer word, and half word and bytes for the Byte unit
- Typical multimeidia intructions
  - including all instructions for the Byte unit, and quadword shift, rotate, gather for the Permute unit
- All binary encodings, names, and mnemonics for all instructions must be exactly as those in the Cell SPU
  - You will need to develop a simple parser to convert your test benches written with assembly instructions into binary code to be loaded into the instruction memory before the start of execution
- ~ 90-100 instruction total

Your SPU-lite Instruction Set Table Format

Name	Mnemonic	RTL Description	Exec Unit	Exec Pipe	Instruction Latency
add half word immediate	ahi rt,ra,val	s <- RepLeftBit(I10,16) for j=0 to 15 by 2 RT $^{2j:2j+1}$ <- RA $^{2j:2j+1}$+s	FX1 (1)	even	3
add word	a rt,ra,rb	for j=0 to 15 by 4 RT $^{4j:4j+3}$ <- RA $^{4j:4j+3}$ + RB $^{4j:4j+3}$	FX1 (1)	even	3

8 16-bit halfword results are calculated with an **ahi** instruction

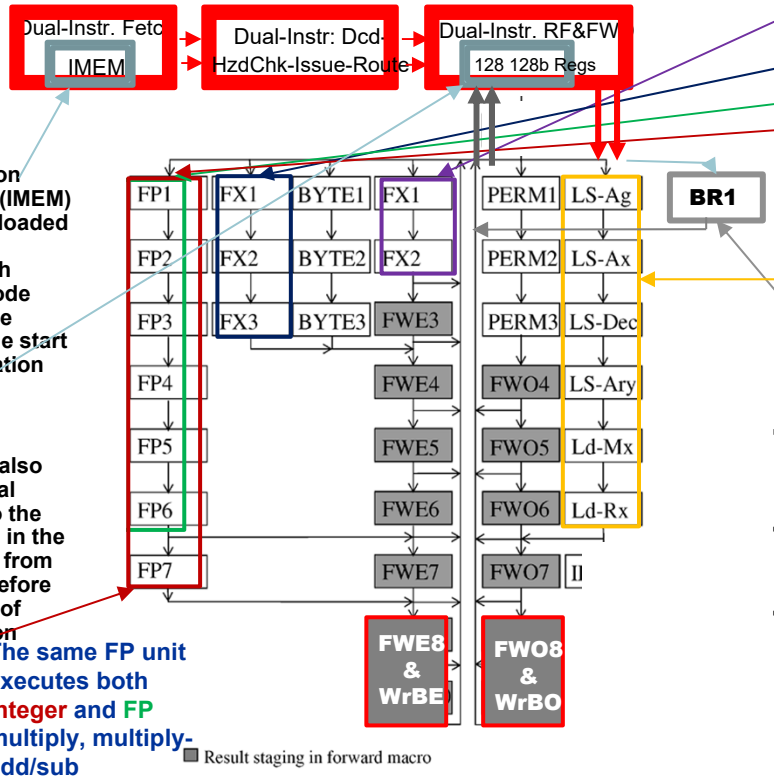
4 32-bit word results are calculated with an **a (add word)** instruction

Unit's name (Fixed-point 1) with its unit ID=1 in ( )

the earliest forwarding stage for the unit's result

ESE 545 SPU-lite Project Goals

Pipeline stages: two instructions/stage in the first 3 stages



2 KB Instruction memory (IMEM) must be loaded with the testbench binary code from a file before the start of simulation

You can also load initial values to the registers in the 3rd stage from the file before the start of simulation

- The same FP unit executes both integer and FP multiply, multiply-add/sub
  - different instruction latencies (7 and 6 cycles) for integer and FP multiply ops

Brian Flachs et al, The Microarchitecture of the Synergistic Processor for a Cell Processor, 2006

Unit	Instructions	Execution Pipe	Unit Pipeline Depth	Instruction Latency
Simple Fixed 1	word arithmetic, logicals, count leading zeros, selects, and compares	Even	2	X 3
Simple Fixed 2	word shifts and rotates	Even	3	4
Single Precision	multiply-accumulate	Even	6	X 7
Single Precision	integer multiply-accumulate	Even	7	X 8
Byte	pop count, absolute sum of differences, byte average, byte sum	Even	3	4
Permute	Quadword shifts, rotates, gathers, shuffles as well as reciprocal estimate	Odd	3	4
32 KB Local Store (data only)	Load and store	Odd	6	X 7
Branch	Branches	Odd	1	2

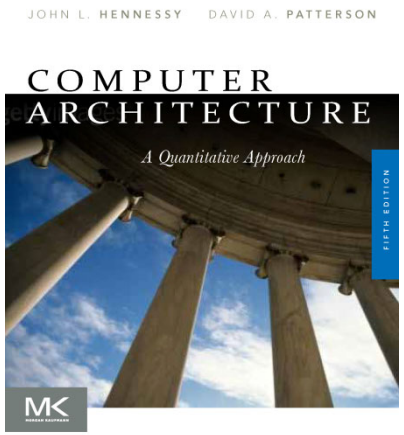
- Create a cycle-accurate model of the SPU-lite pipeline
  - Not a gate-level implementation of all stages and units!
- All operations by all units are executed in the very first stage of each unit using a behavioral-style VHDL/SystemVerilog code
- Each calculated result is a part of a unit result packet consisting:
  - A unit ID (you need to assign numbers from 1 to 7 to all seven units, using zero ID for no-op)
  - A 128-bit result
  - A register destination number (from the corresponding instruction)
  - A ready-stage number (from 2 to 7 depending on the operation) specifying the earliest stage when the result is available for forwarding (given by the instruction latency column)
  - A RegWr flag whether the result needs to be written to the destination register
- Each cycle unit result packets are shifted down the even and odd pipes until they reach the final write-back stages where they are written to the destination registers if their RegWr flags are set to 1

From one of the ESE545 former students

Good afternoon professor Dorojevets,

.. I've graduated this spring with a master's degree and wanted to thank you for teaching me in ***ESE 565 and ESE 545****. I've been working as a hardware engineer in a startup company called … in Seattle, WA after graduation. …*

The projects you gave me during the courses were especially valuable during the interviews*, and all interviewers (including the ones from Samsung) were impressed by how I was able to design and verify a cache-coherent multicore CPU, let alone* ***a dual-issue SIMD processor****.* ***I firmly believe that these projects played a pivotal role in giving me the upper hand during negotiations****,... Also, my workplace mostly consists of PhD holders, and being a startup, it demands a high level of expertise from each member.* ***As a master's degree graduate I wouldn't have gotten this position without experiences from your courses****.*



ESE 545 Computer Architecture

Introduction: Fundamentals of Quantitative Design and Analysis, Technology Trends

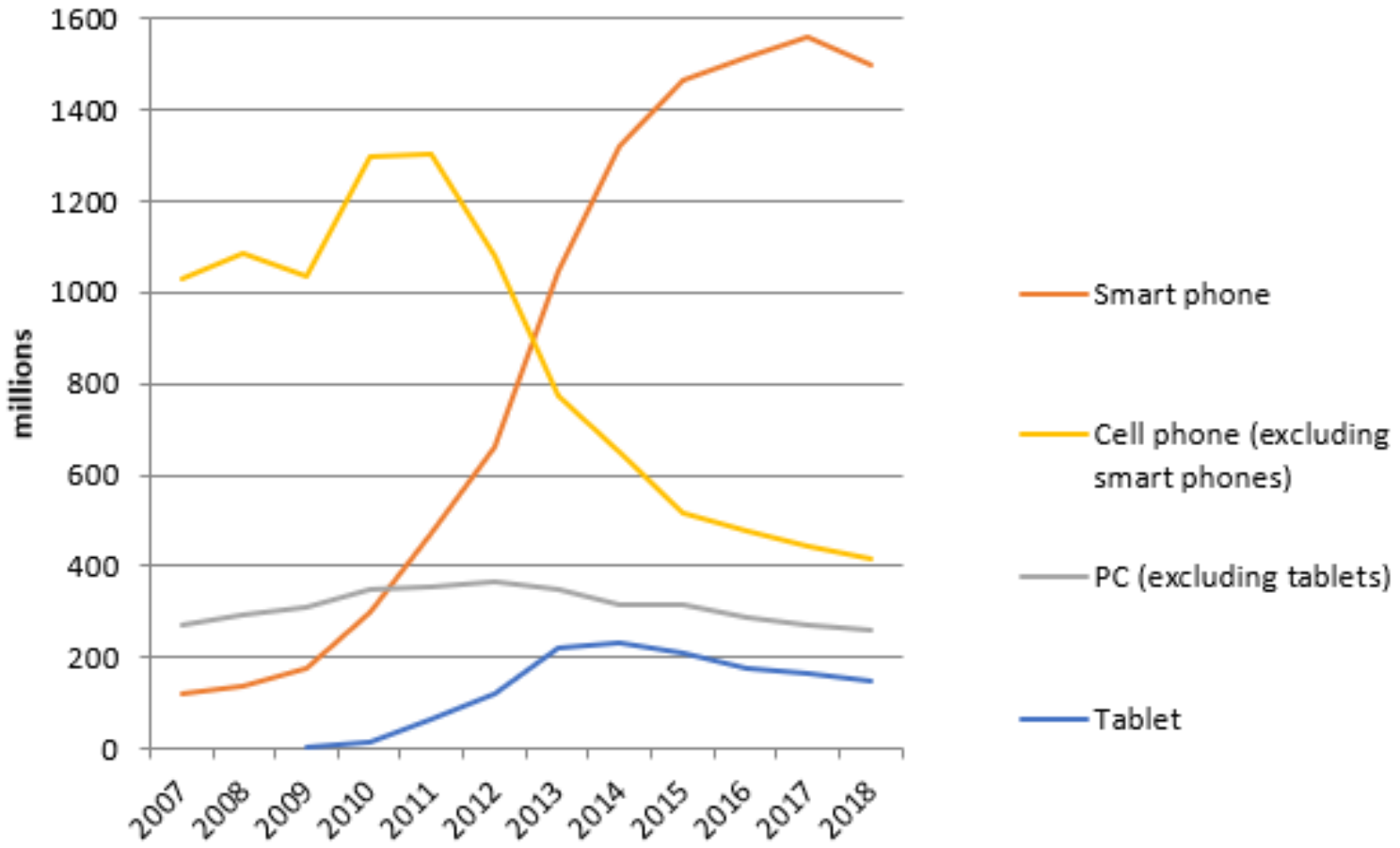
Classes of Computers

- Personal computers
  - General purpose, variety of software
  - Subject to cost/performance tradeoff
- Server computers
  - Network based
  - High capacity, performance, reliability
  - Range from small servers to building sized

Classes of Computers

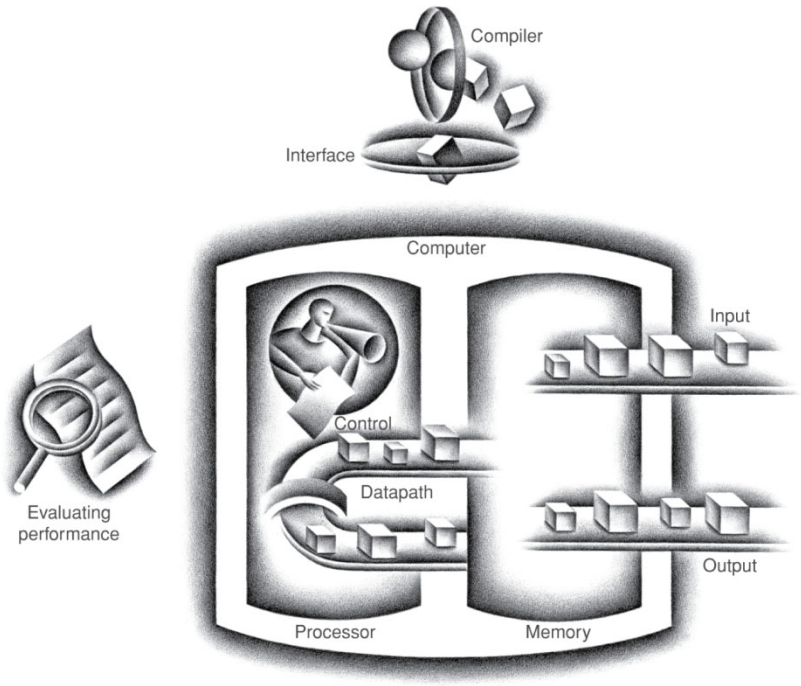
- Supercomputers
  - Type of server
  - High-end scientific and engineering calculations
  - Highest capability but represent a small fraction of the overall computer market
- Embedded computers
  - Hidden as components of systems
  - Stringent power/performance/cost constraints

The PostPC Era



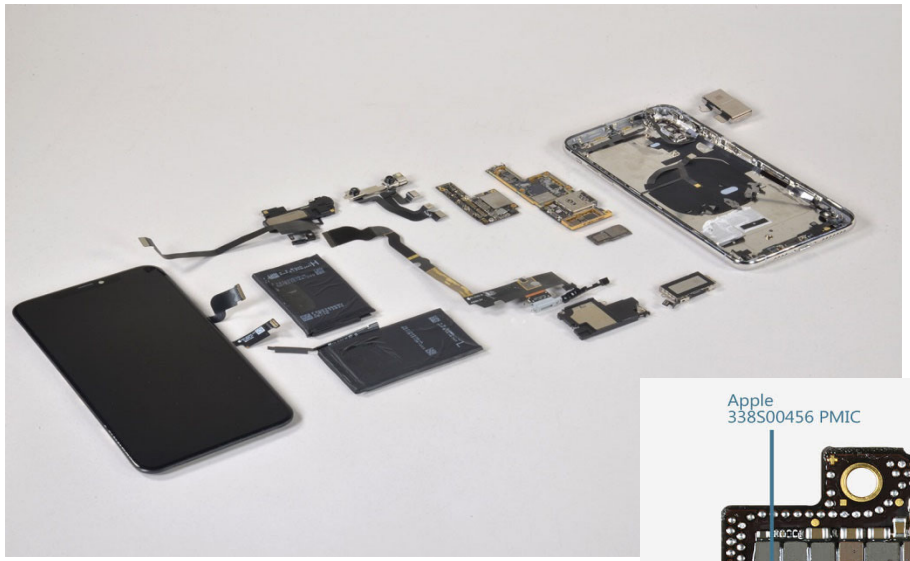
Components of a Computer

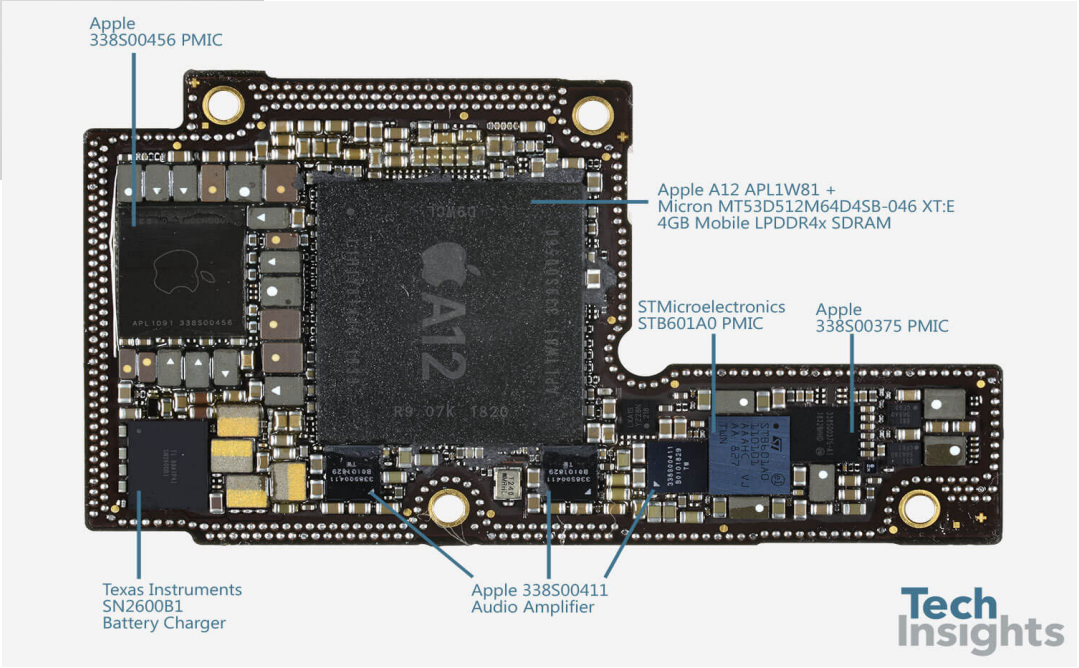
The BIG Picture



- Same components for all kinds of computer since 1946
  - Desktop, server, embedded
- Input/output includes
  - User-interface devices
    - Display, keyboard, mouse
  - Storage devices
    - Hard disk, CD/DVD, flash
  - Network adapters
    - For communicating with other computers

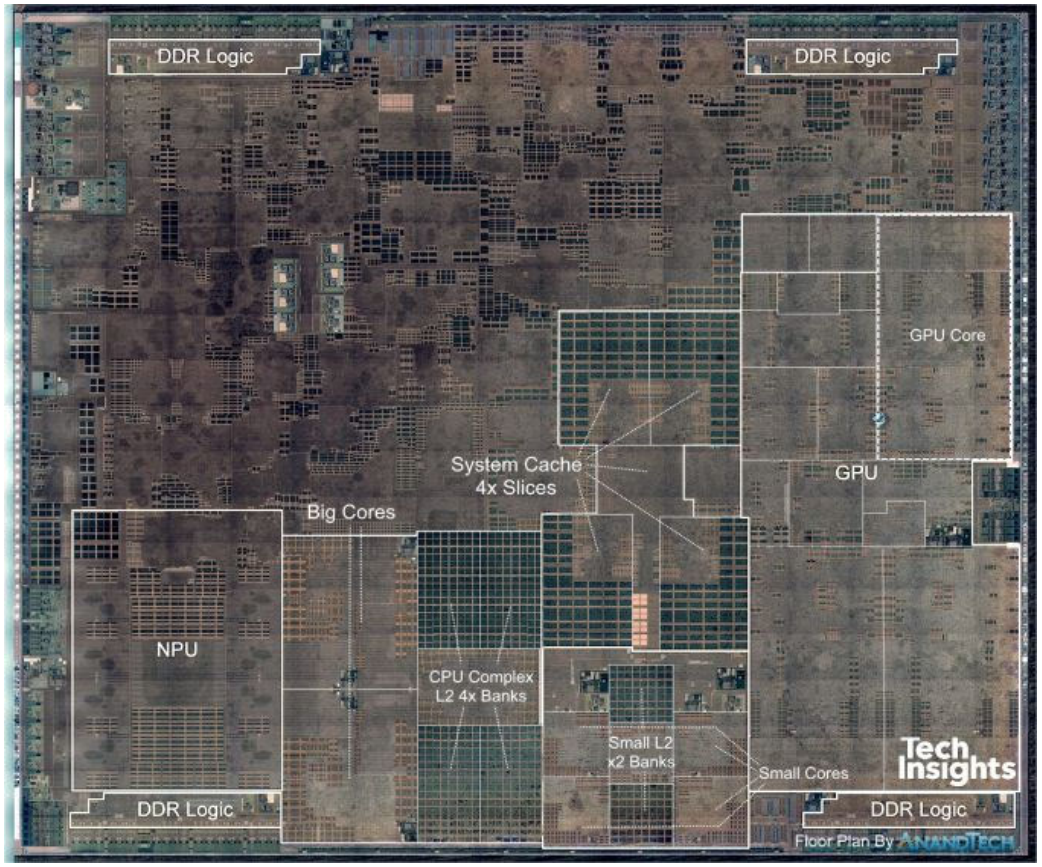
Opening the Box (iPhone XS)





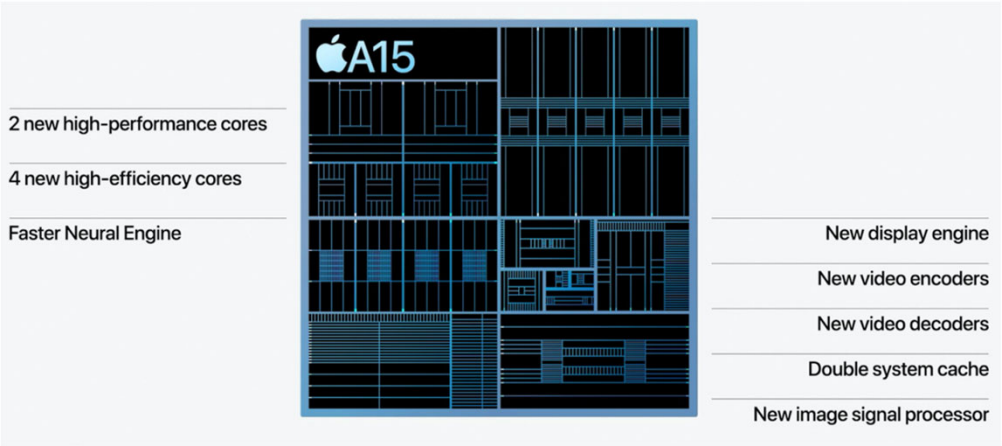
Inside the Processor (2018)

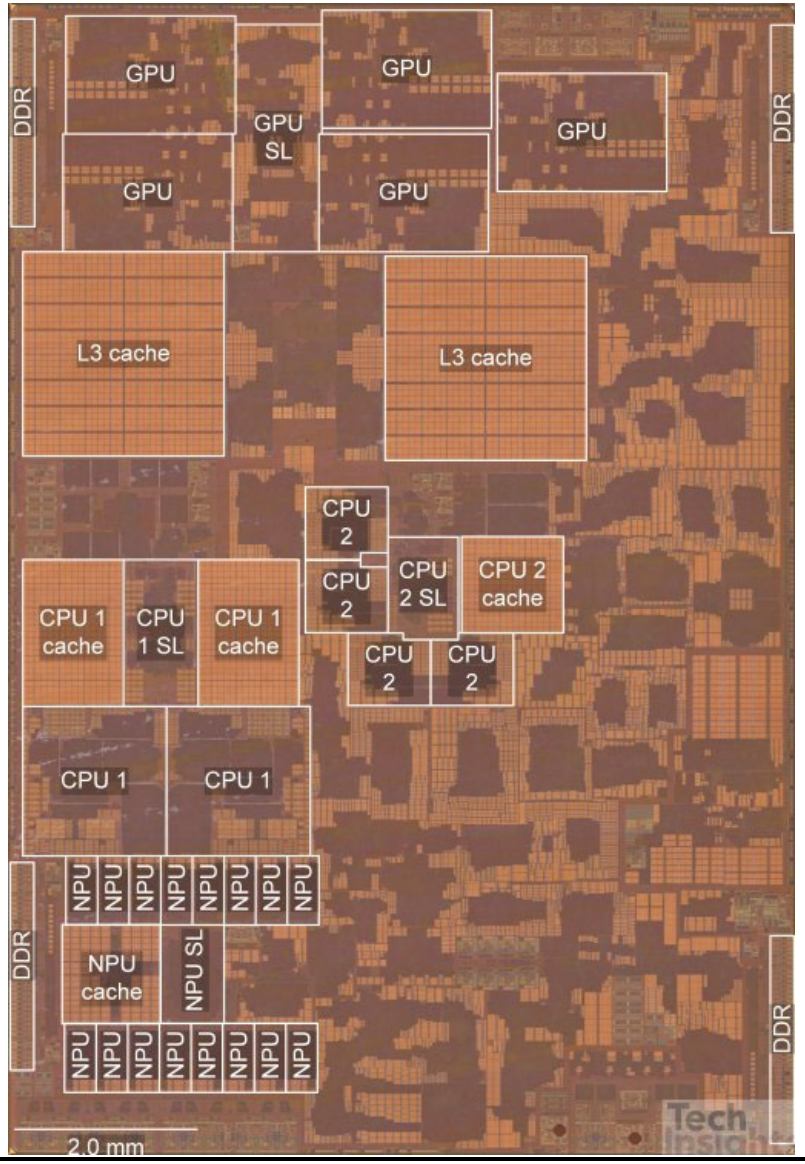
- A multicore A12 processor
  - (2 fast and 4 low-power A12 cores)



A15 in iPhone 13 (2021)

- 15B transistors
- A multicore A15 processor
  - (2 fast 3.24 GHz and 4 low-power 2.0 GHz cores)





Inside the Processor Core (CPU)

- Datapath: performs operations on data
- Control: sequences datapath, memory, ...
- Cache memory
  - Small fast SRAM memory for immediate access to data

What You Will Learn

- How programs are translated into the machine language
  - And how the hardware executes them
- The hardware/software interface
- What determines program performance
  - And how it can be improved
- How hardware designers improve performance
- What is parallel processing

What is "Computer Architecture"

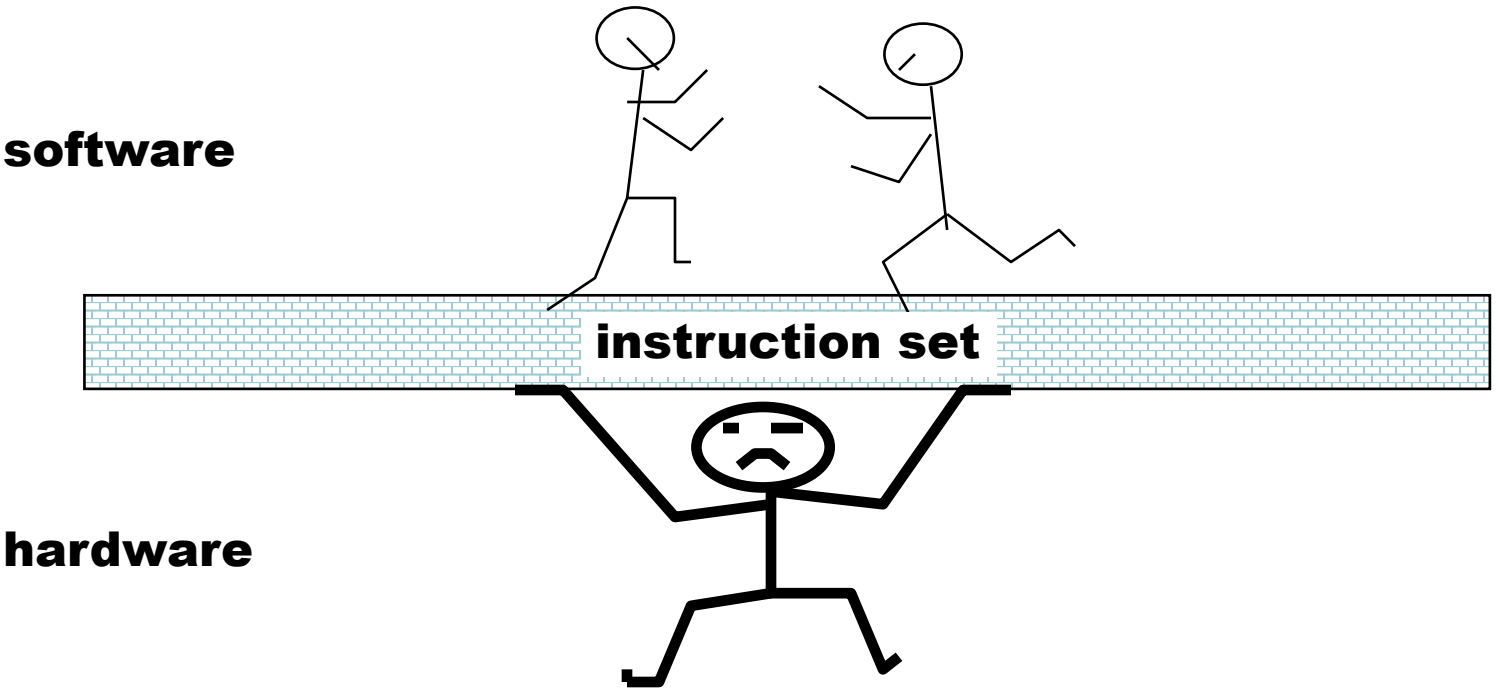
Computer Architecture =

Instruction Set Architecture +

Machine Organization (Microarchitecture) +

…..

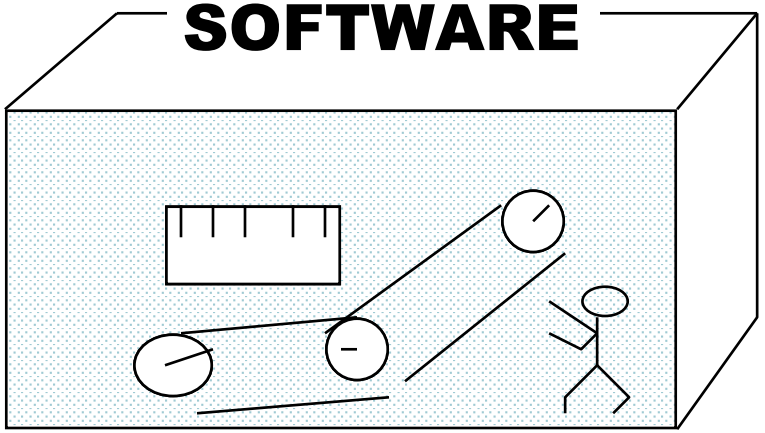
Instruction Set Architecture:Critical Interface



- Properties of a good abstraction
  - Lasts through many generations (portability)
  - Used in many different ways (generality)
  - Provides convenient functionality to higher levels
  - Permits an efficient implementation at lower levels

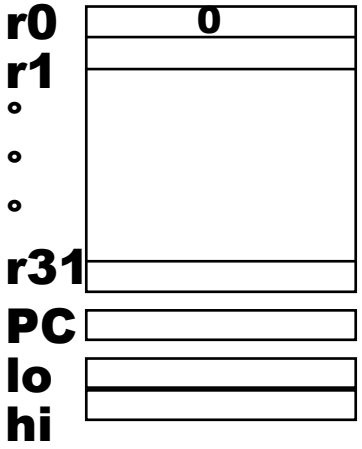
Instruction Set Architecture

"... the attributes of a [computing] system as seen by the programmer, *i.e.* the conceptual structure and functional behavior, as distinct from the organization of the data flows and controls the logic design, and the physical implementation." – Amdahl, Blaauw, and Brooks, 1964



- **Organization of Programmable Storage**
- **Data Types & Data Structures: Encodings & Representations**
- **Instruction Formats**
- **Instruction (or Operation Code) Set**
- **Modes of Addressing and Accessing Data Items and Instructions**
- **Exceptional Conditions**

Example: MIPS



Programmable storage

- 2^32 x bytes
- 31 x 32-bit GPRs (R0=0)
- 32 x 32-bit FP regs (paired DP)
- HI, LO, PC

Data types ?

Format ?

Addressing Modes?

Arithmetic logical

Add, AddU, Sub, SubU, And, Or, Xor, Nor, SLT, SLTU,
AddI, AddIU, SLTI, SLTIU, AndI, OrI, XorI, *LUI*
SLL, SRL, SRA, SLLV, SRLV, SRAV

Memory Access

LB, LBU, LH, LHU, LW, LWL,LWR
SB, SH, SW, SWL, SWR

Control

J, JAL, JR, JALR
BEq, BNE, BLEZ,BGTZ,BLTZ,BGEZ,BLTZAL,BGEZAL

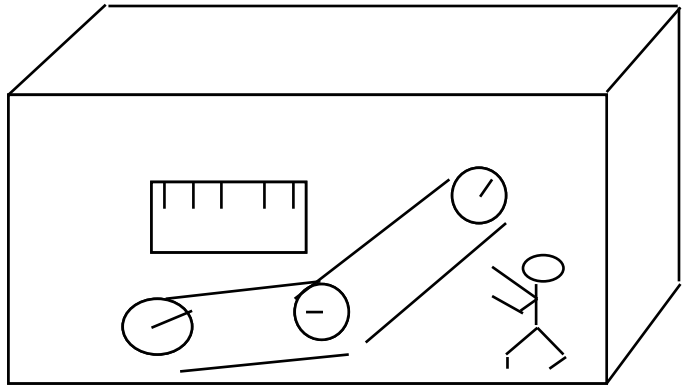
Organization (=Microarchitecture)

- Capabilities & Performance Characteristics of Principal Functional Units
  - (e.g., Registers, ALU, Shifters, Logic Units, ...)
- Ways in which these components are interconnected
- Information flows between components
- Logic and means by which such information flow is controlled.
- Choreography of FUs to realize the ISA
- Register Transfer Level (RTL) Description

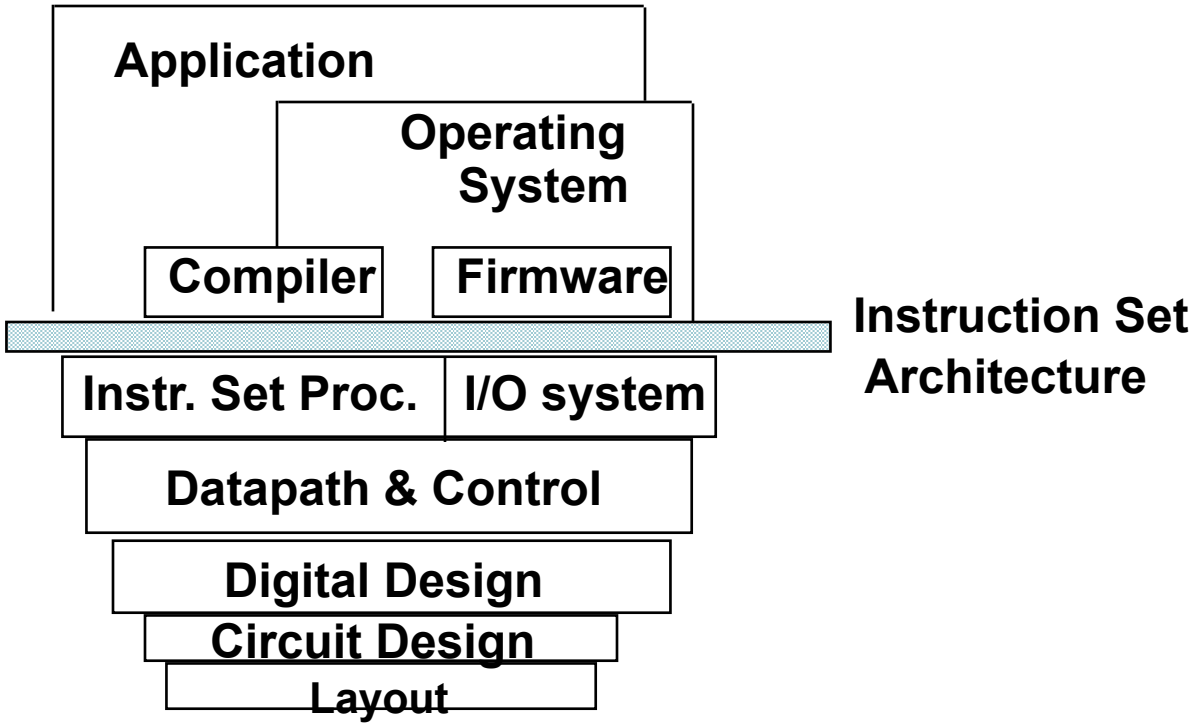
Logic Designer's View

ISA Level

FUs & Interconnect



What is "Computer Architecture"?

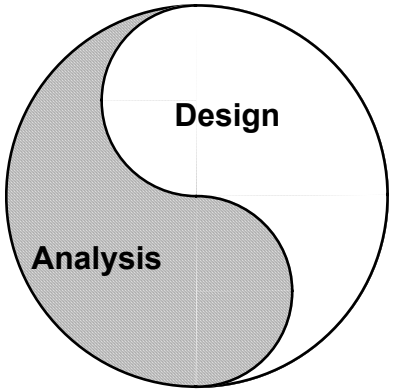


- In its broad definition, computer architecture is the *design of the abstraction layers* to maximize performance within constraints (e.g., cost, power, and availability) that allow us to implement information processing applications efficiently using available manufacturing technologies.

Computer Architecture is an Integrated Approach

- What really matters is the functioning of the complete system
  - hardware, runtime system, compiler, operating system, and application
  - In networking, this is called the “End to End argument”
- Computer architecture is not just about transistors, individual instructions, or particular implementations
  - E.g., Original RISC projects replaced complex instructions with a compiler + simple instructions

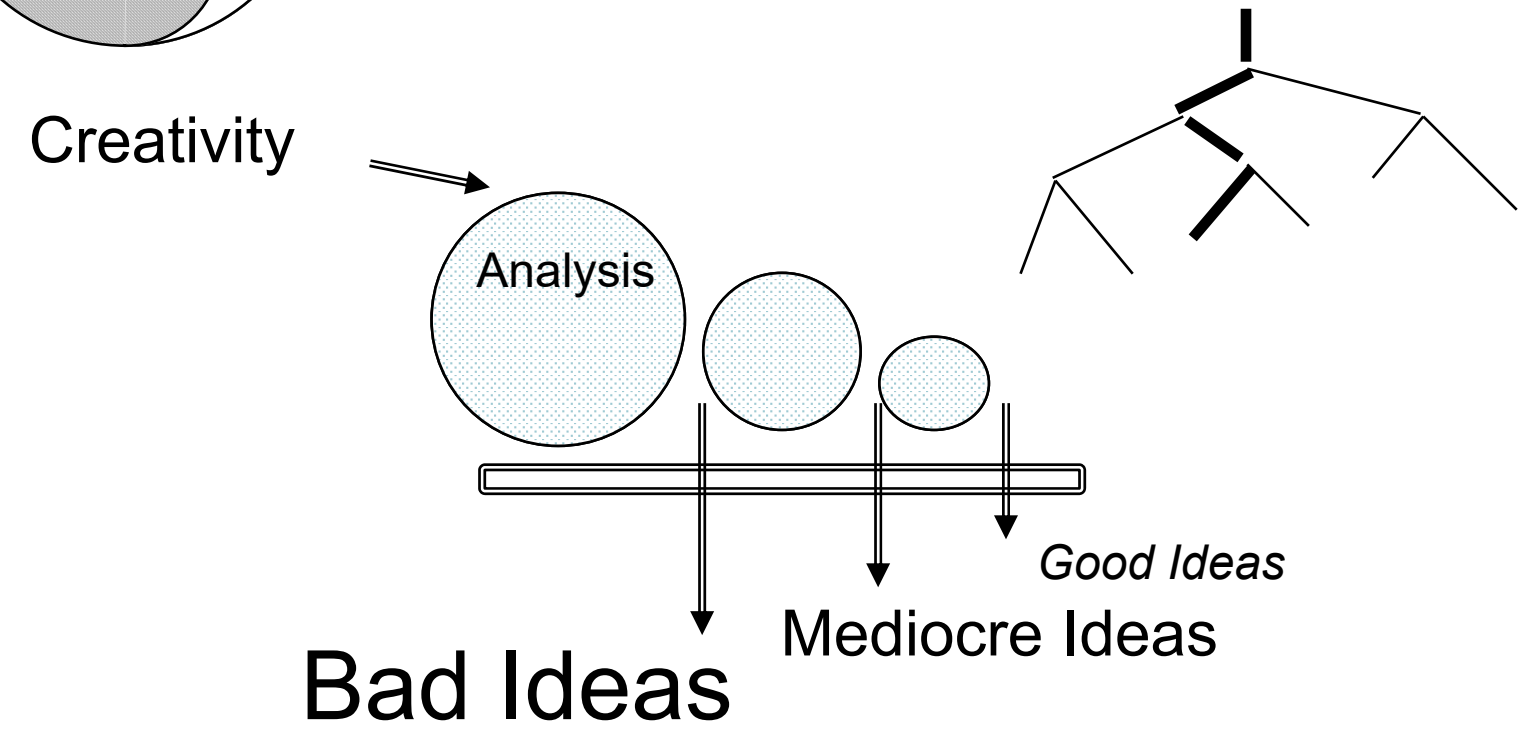
Computer Architecture is Design and Analysis



There is no single recipe for right architecture design

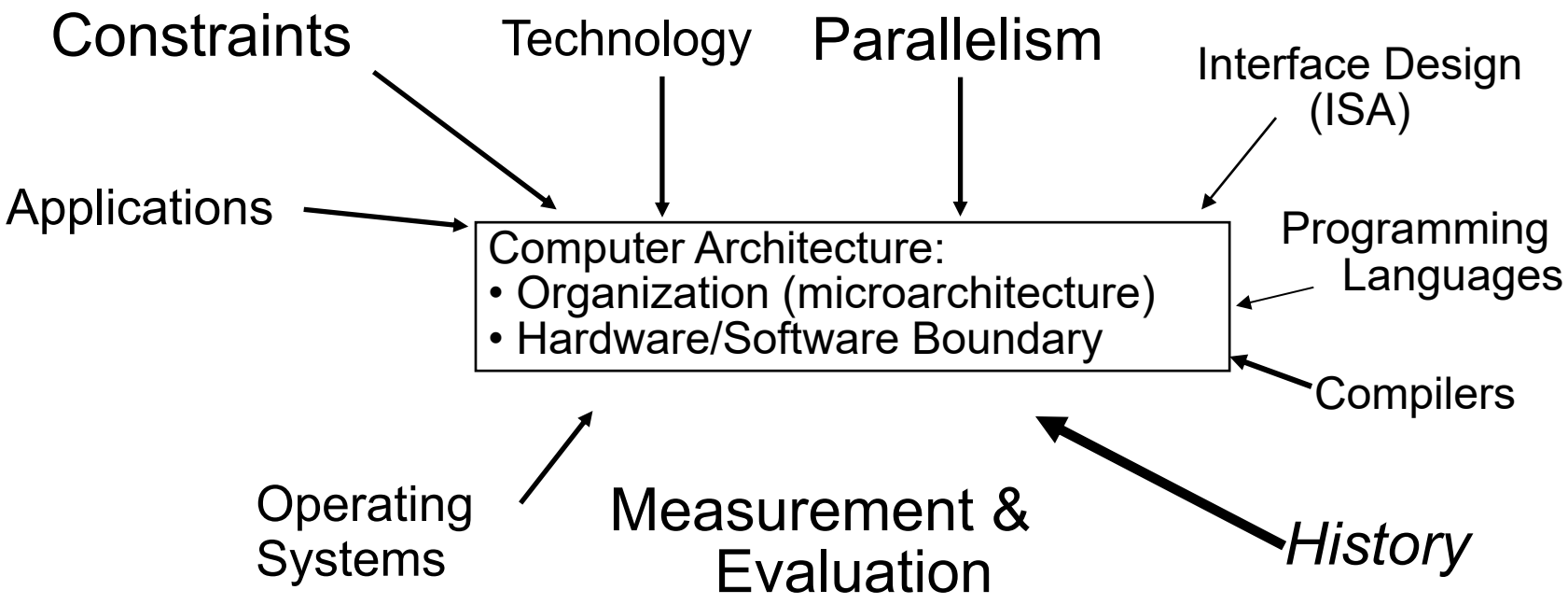
Architecture design is an iterative process:

- Searching the space of possible designs
- At all levels of computer systems



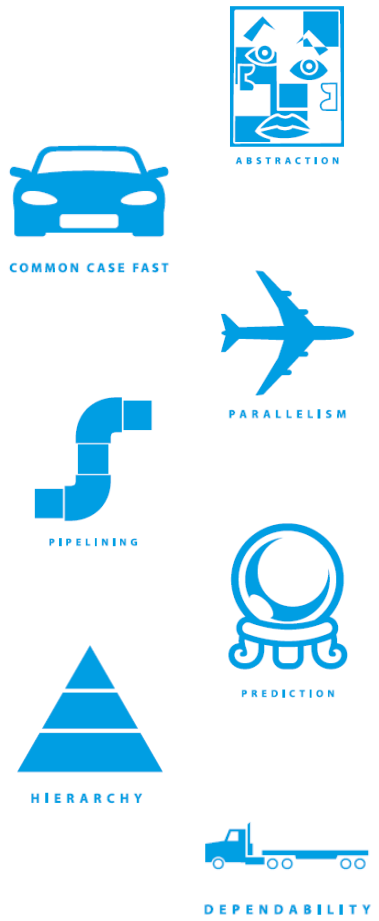
Computer Architecture Course Focus

Understanding the design techniques, machine structures, technology factors, evaluation methods that will determine the form of computers in 21st Century



Great Ideas

- Use ***abstraction*** to simplify design
- Make the ***common case fast***
- Performance *via* ***parallelism***
- Performance *via* ***pipelining***
- Performance *via* ***prediction***
- ***Hierarchy*** of memories
- ***Dependability*** *via* redundancy
- ***Technology trends***

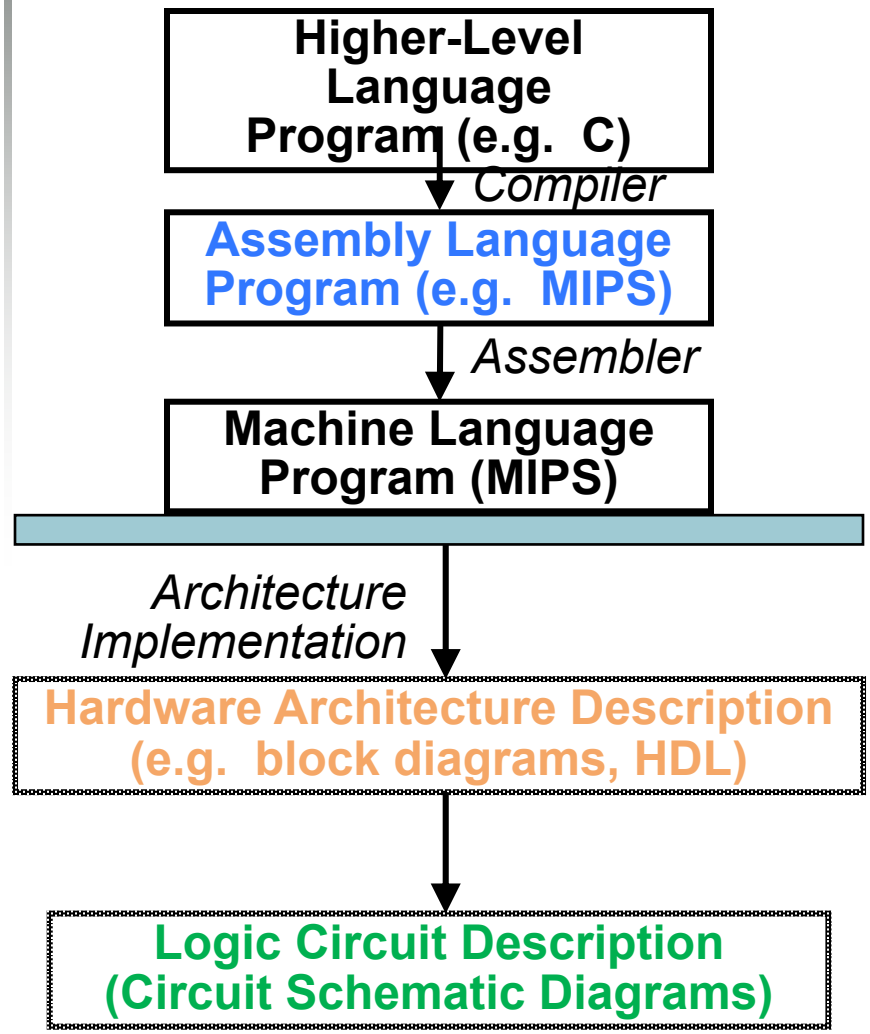


Abstractions

The BIG Picture

- Abstraction helps us deal with complexity
  - Hide lower-level detail
- Instruction set architecture (ISA)
  - The hardware/software interface
- Application binary interface (ABI)
  - The ISA plus system software interface
- Implementation (microarchitecture, …)
  - The details underlying and interface

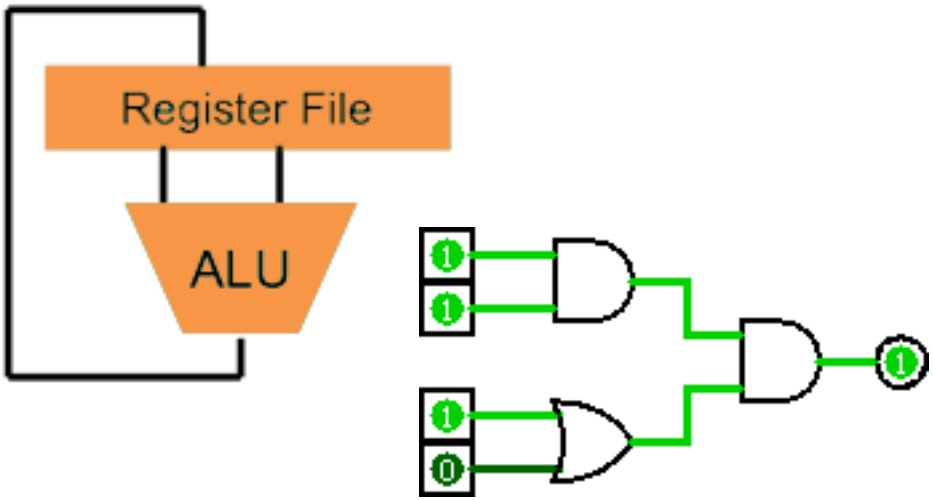
Idea #1: Levels of Representation in SW & HW




```
temp = v[k];
v[k] = v[k+1];
v[k+1] = temp;
```

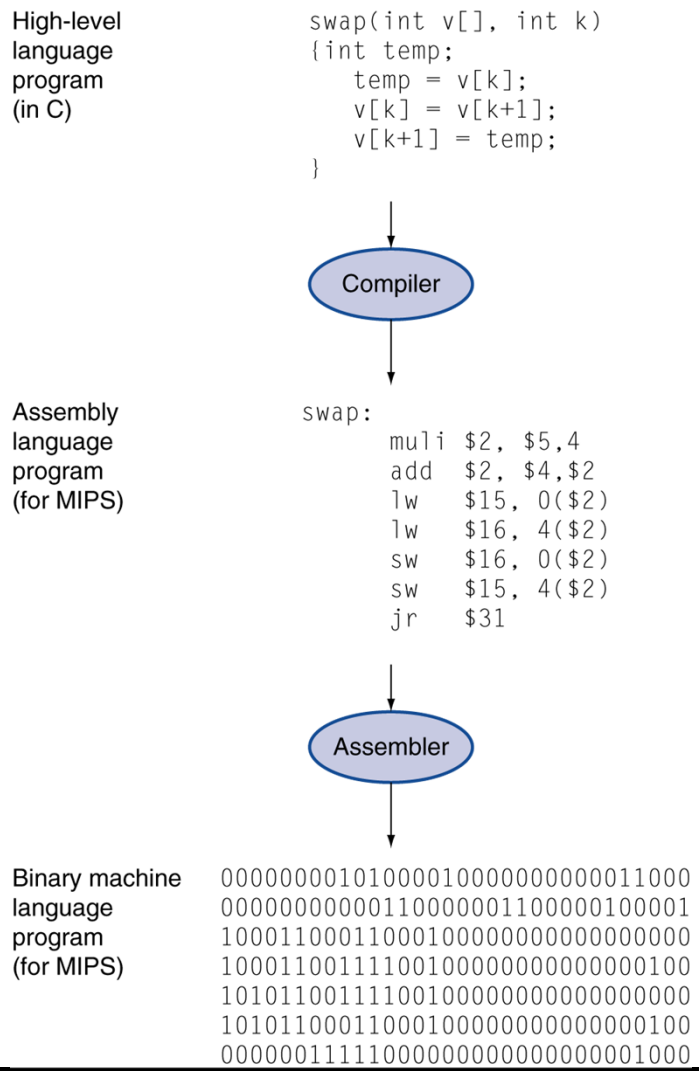
```
lw    $t0, 0($2)
lw    $t1, 4($2)
sw    $t1, 0($2)
sw    $t0, 4($2)
```

```
0000 1001 1100 0110 1010 1111 0101 1000
1010 1111 0101 1000 0000 1001 1100 0110
1100 0110 1010 1111 0101 1000 0000 1001
0101 1000 0000 1001 1100 0110 1010 1111
```

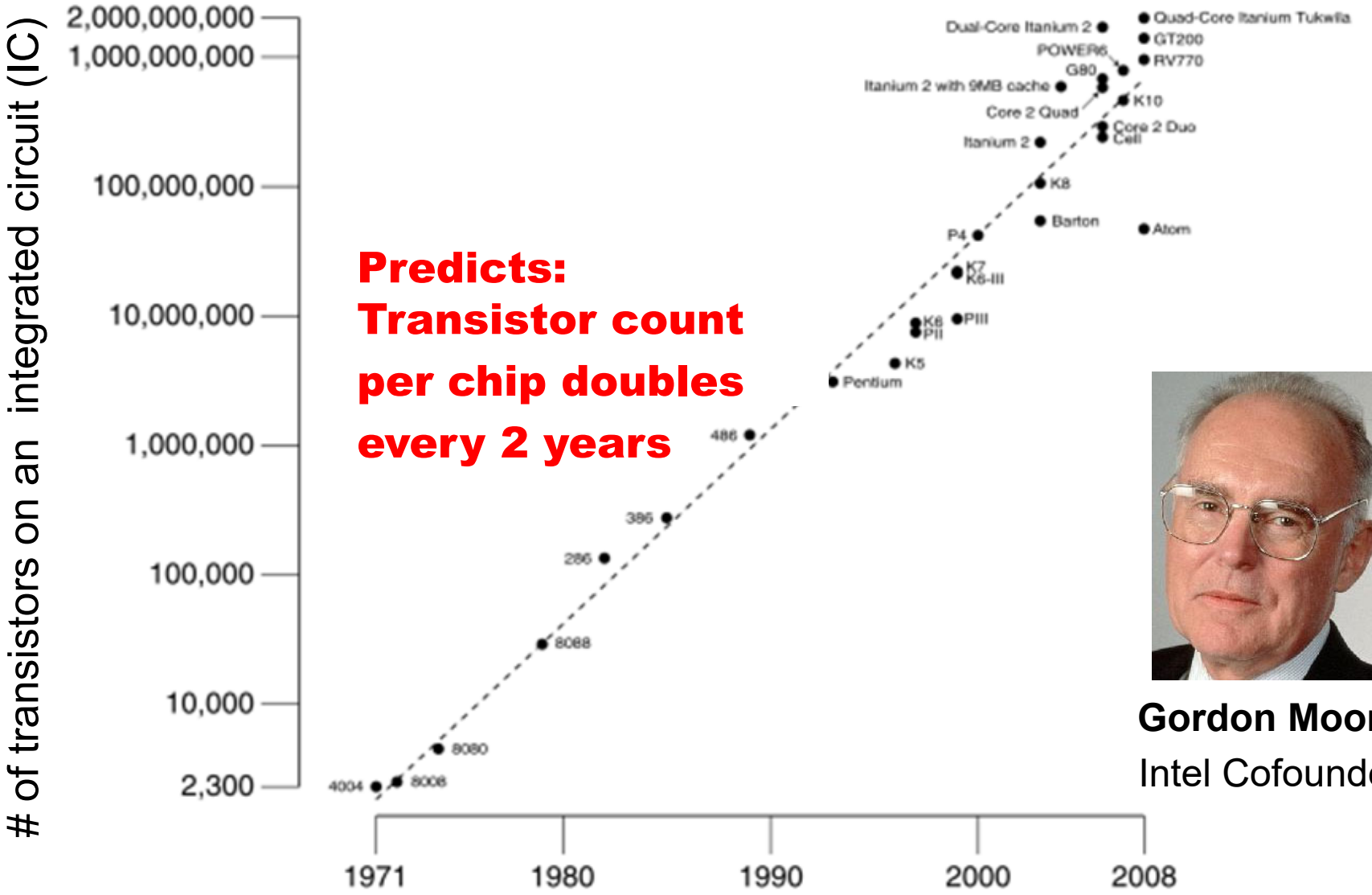


Levels of Program Code

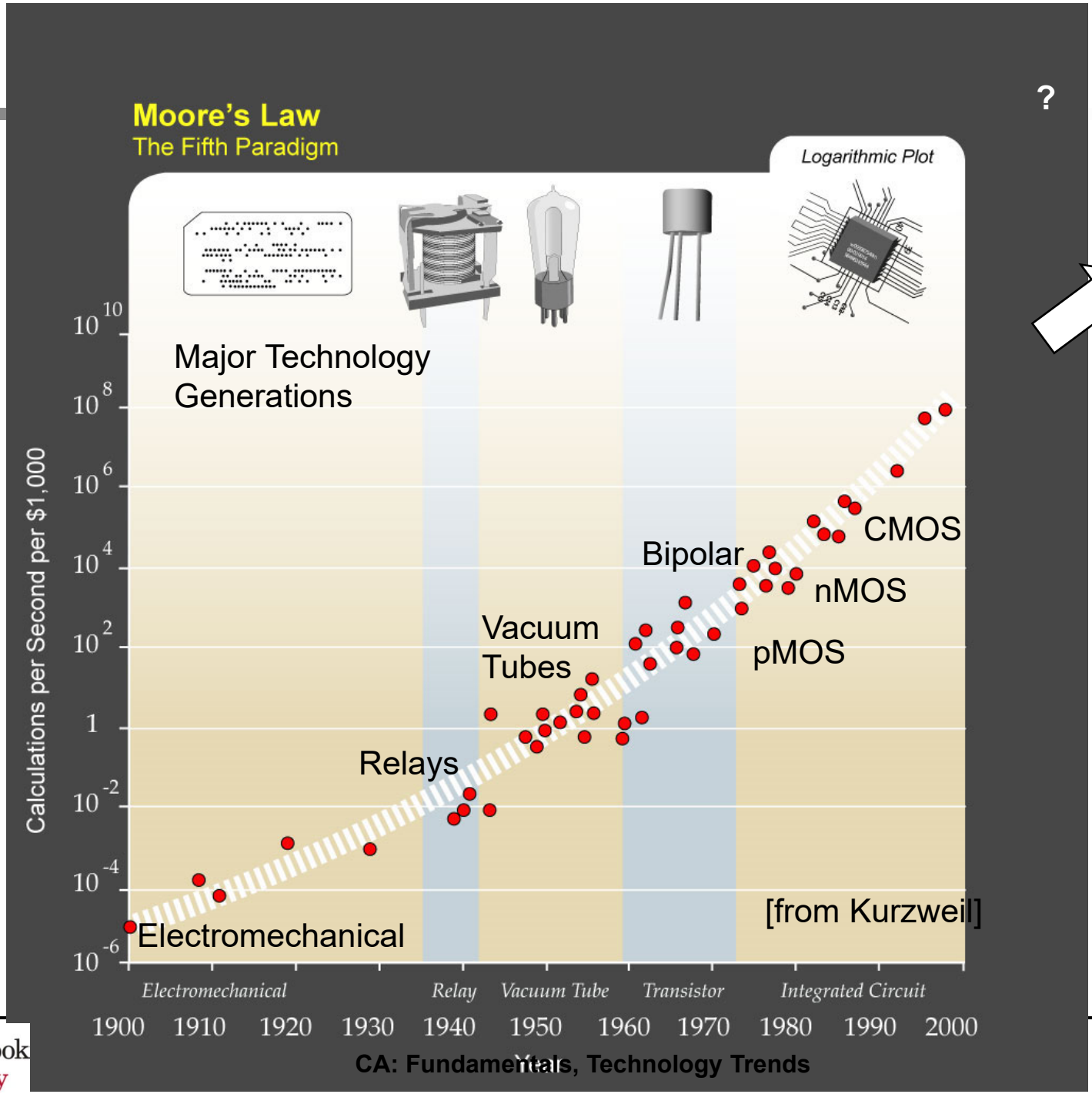
- High-level language
  - Level of abstraction closer to problem domain
  - Provides for productivity and portability
- Assembly language
  - Textual representation of instructions
- Hardware representation
  - Binary digits (bits)
  - Encoded instructions and data



Idea #2: Technology Trends

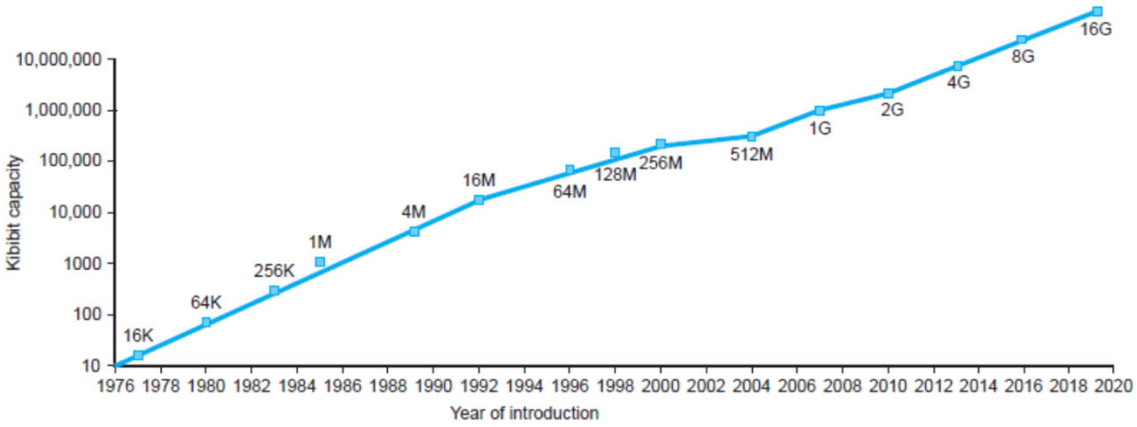


Gordon Moore
Intel Cofounder


Moore's Law
The Fifth Paradigm
Logarithmic Plot
?
Major Technology Generations
Calculations per Second per $1,000
10^{10}
10^{8}
10^{6}
10^{4}
10^{2}
1
10^{-2}
10^{-4}
10^{-6}
Electromechanical
Relays
Vacuum Tubes
Bipolar
nMOS
pMOS
CMOS
[from Kurzweil]
Electromechanical
Relay
Vacuum Tube
Transistor
Integrated Circuit
1900
1910
1920
1930
1940
1950
1960
1970
1980
1990
2000
Years

Technology Trends

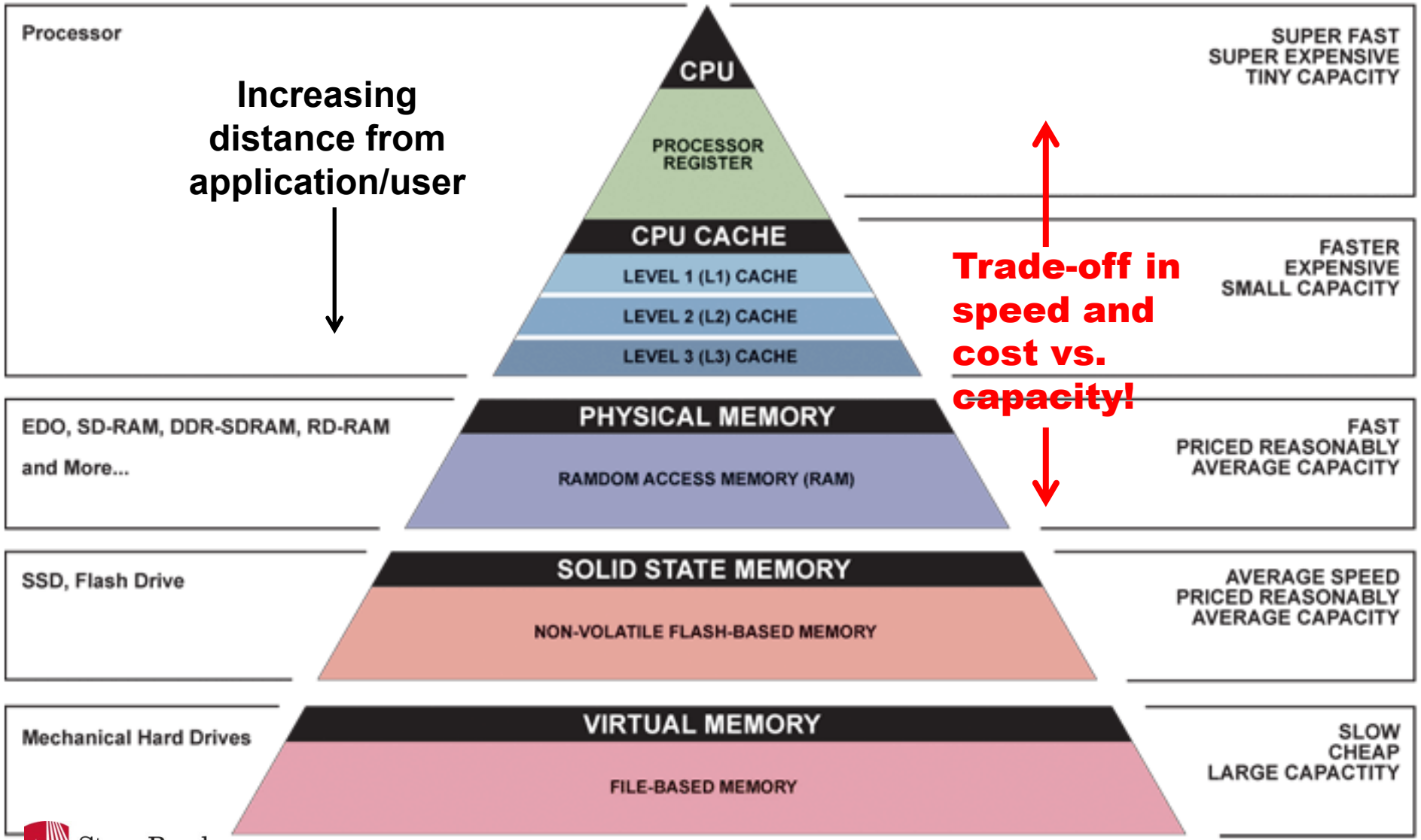
- Electronics technology continues to evolve
  - Increased capacity and performance
  - Reduced cost



DRAM capacity

Year	Technology	Relative performance/cost
1951	Vacuum tube	1
1965	Transistor	35
1975	Integrated circuit (IC)	900
1995	Very large scale IC (VLSI)	2,400,000
2020	Ultra large scale IC	500,000,000,000

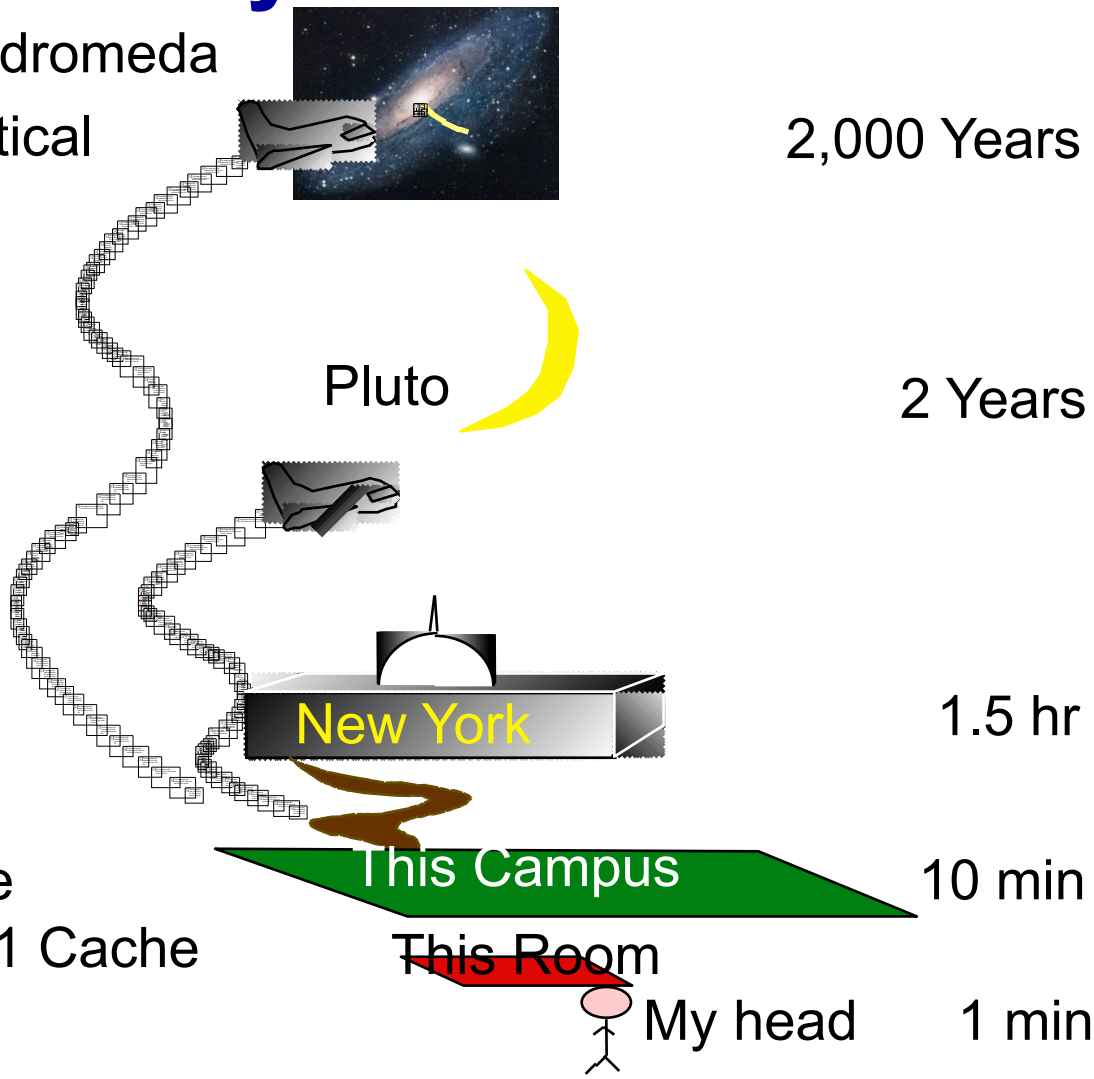
Idea #3: Principle of Locality/ Memory Hierarchy



Memory Technology

- Static RAM (SRAM)
  - 0.5ns – 2.5ns, $500 – $1000 per GB
- Dynamic RAM (DRAM)
  - 50ns – 70ns, $3 – $6 per GB
- Magnetic disk
  - 5ms – 20ms, $0.01 – $0.02 per GB
- Ideal memory
  - Access time of SRAM
  - Capacity and cost/GB of disk

Storage Latency Analogy: How Far Away is the Data?



Idea #4: Parallelism

Software | *Hardware*

- Parallel Requests
  - Assigned to computer
  - e.g. search “Garcia”
- Parallel Threads
  - Assigned to core
  - e.g. lookup, ads
- Parallel Instructions
  - > 1 instruction @ one time
  - e.g. 5 pipelined instructions
- Parallel Data
  - > 1 data item @ one time
  - e.g. add of 4 pairs of words
- Hardware descriptions
  - All gates functioning in parallel at same time

Leverage Parallelism & Achieve High Performance

Warehouse Scale Computer

Smart Phone

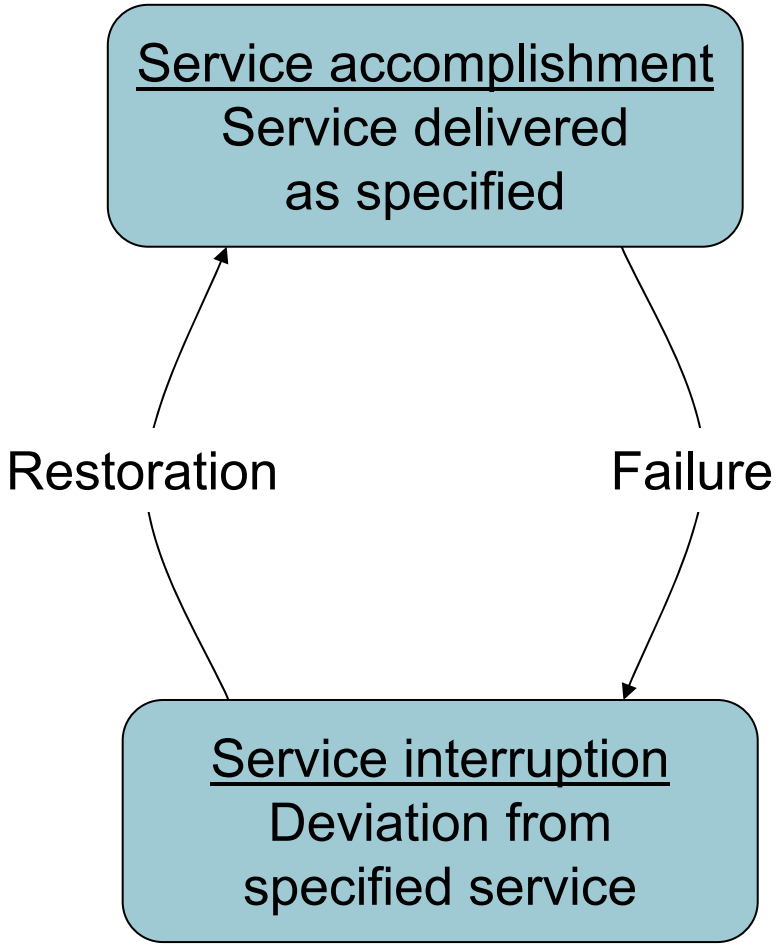
Idea #5: Performance Measurement and Improvement

- Allows direct comparisons of architectures and quantification of improvements
- Most common measures are *time to finish* (latency) and *rate of execution* (throughput)
- Match application and hardware to exploit:
  - Locality, parallelism, special hardware features

Understanding Performance

- Algorithm
  - Determines number of operations executed
- Programming language, compiler, architecture
  - Determine number of machine instructions executed per operation
- Processor and memory system
  - Determine how fast instructions are executed
- I/O system (including OS)
  - Determines how fast I/O operations are executed

Idea #6: Dependability



- Fault: failure of a component
  - May or may not lead to system failure

Dependability Measures

- Reliability: mean time to failure (MTTF)
- Service interruption: mean time to repair (MTTR)
- Mean time between failures
  - MTBF = MTTF + MTTR
- Availability = MTTF / (MTTF + MTTR)
- Improving Availability
  - Increase MTTF: fault avoidance, fault tolerance, fault forecasting
  - Reduce MTTR: improved tools and processes for diagnosis and repair

Idea #6: Dependability via Redundancy

- Applies to everything from datacenters to storage to memory
  - Redundant datacenters so that can lose 1 datacenter but Internet service stays online
  - Redundant disks so that can lose 1 disk but not lose data (Redundant Arrays of Independent Disks/RAID)
  - Redundant memory bits of so that can lose 1 bit but no data (Error Correcting Code/ECC Memory)
- Increasing transistor density reduces the cost of redundancy

Example Calculating Reliability

- If modules have *exponentially distributed lifetimes* (age of module does not affect probability of failure), overall failure rate is the sum of failure rates of the modules
- Calculate FIT and MTTF for 10 disks (1M hour MTTF per disk), 1 disk controller (0.5M hour MTTF), and 1 power supply (0.2M hour MTTF):

$$FailureRate =$$

$$MTTF =$$

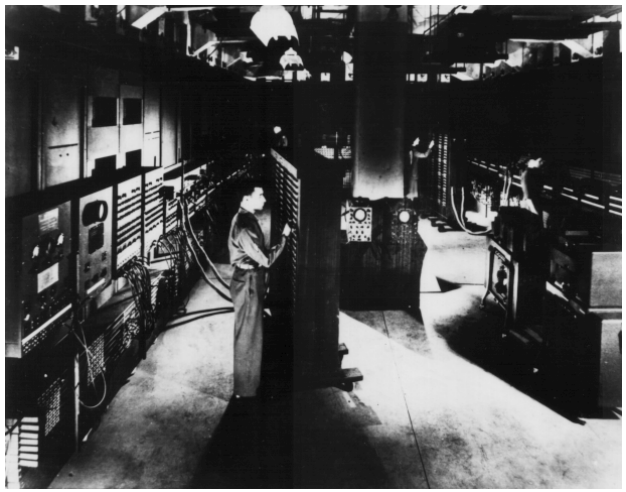
Example Calculating Reliability

- If modules have *exponentially distributed lifetimes* (age of module does not affect probability of failure), overall failure rate is the sum of failure rates of the modules
- Calculate FIT and MTTF for 10 disks (1M hour MTTF per disk), 1 disk controller (0.5M hour MTTF), and 1 power supply (0.2M hour MTTF):

$$
\begin{aligned}
FailureRate &= 10 \times (1/1{,}000{,}000) + 1/500{,}000 + 1/200{,}000 \\
&= 10 + 2 + 5/1{,}000{,}000 \\
&= 17/1{,}000{,}000 \\
&= 17{,}000\,FIT \\
MTTF &= 1{,}000{,}000{,}000/17{,}000 \\
&\approx 59{,}000\,hours
\end{aligned}
$$

Electronic Numerical Integrator and Computer (ENIAC)

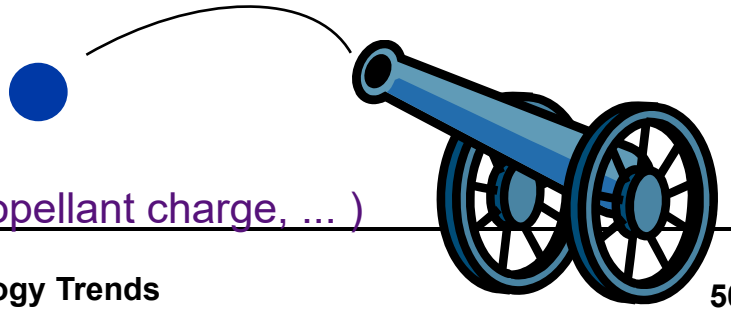
- Inspired by Atanasoff and Berry, Eckert and Mauchly designed and built ENIAC (1943-45) at the University of Pennsylvania
- The first, completely electronic, operational, general-purpose analytical calculator!
  - 30 tons, 72 square meters, 200KW
  - Each of the twenty 10 digit registers was 2 feet long
  - Used 18,000 vacuum tubes
- Performance
  - Read in 120 cards per minute
  - Addition took 200 ms, Division 6 ms
  - 1000 times faster than Mark I
- Not very reliable!



The slogan was: The ENIAC is able to calculate the trajectory of a large-caliber naval shell in less time than the shell takes to reach its target!

Application: Ballistic calculations

angle = f (location, tail wind, cross wind,
air density, temperature, weight of shell, propellant charge, ...)



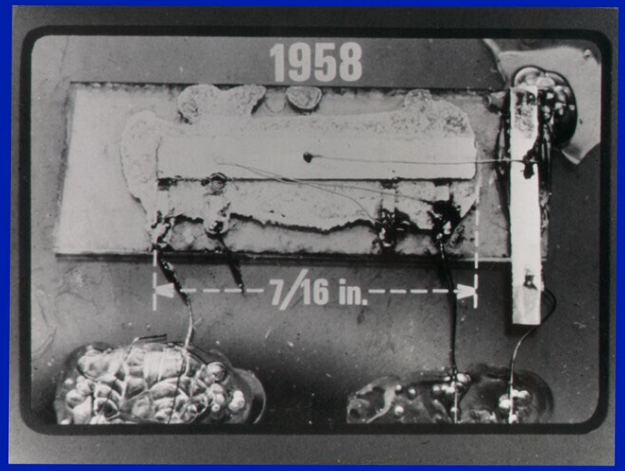
Semiconductor Technology

- Silicon: semiconductor
- Add materials to transform properties:
  - Conductors
  - Insulators
  - Switch

First Integrated Circuits

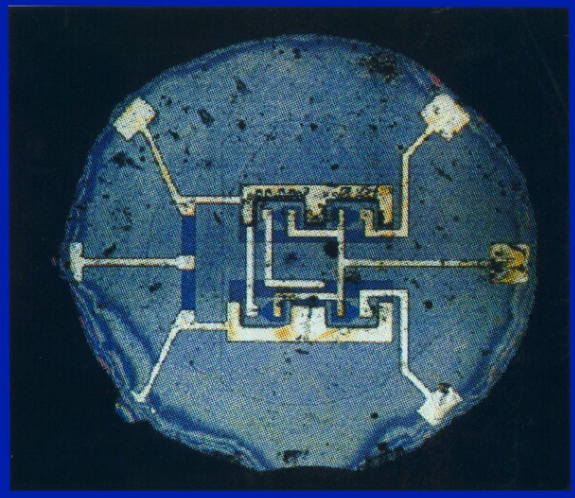
FIRST INTEGRATED CIRCUIT BY J. S. KILBY

(US Patent 3,138,763 filed Feb. 1959, granted 1964)



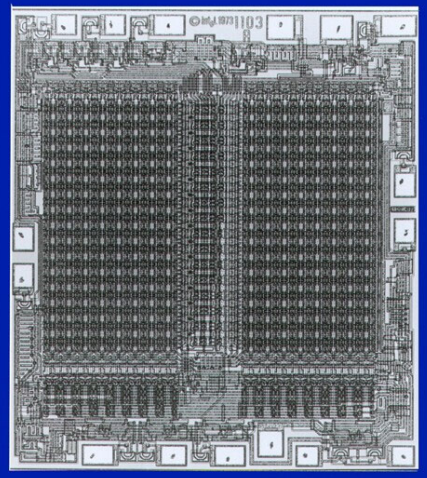
FIRST MONOLITHIC IC BY R. N. NOYCE

(US Patent 2,981,877 filed July 1959, granted 1961)

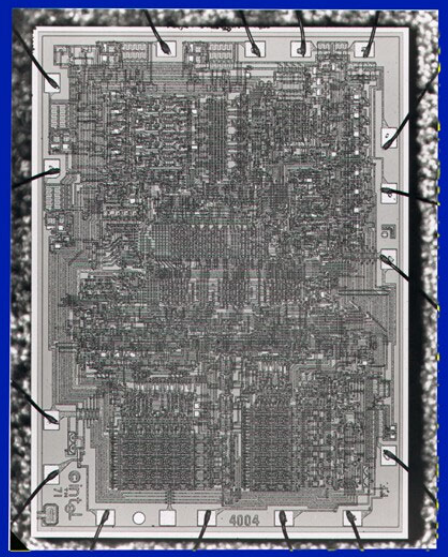


First 1 Kbit DRAM and 4-bit Microprocessor

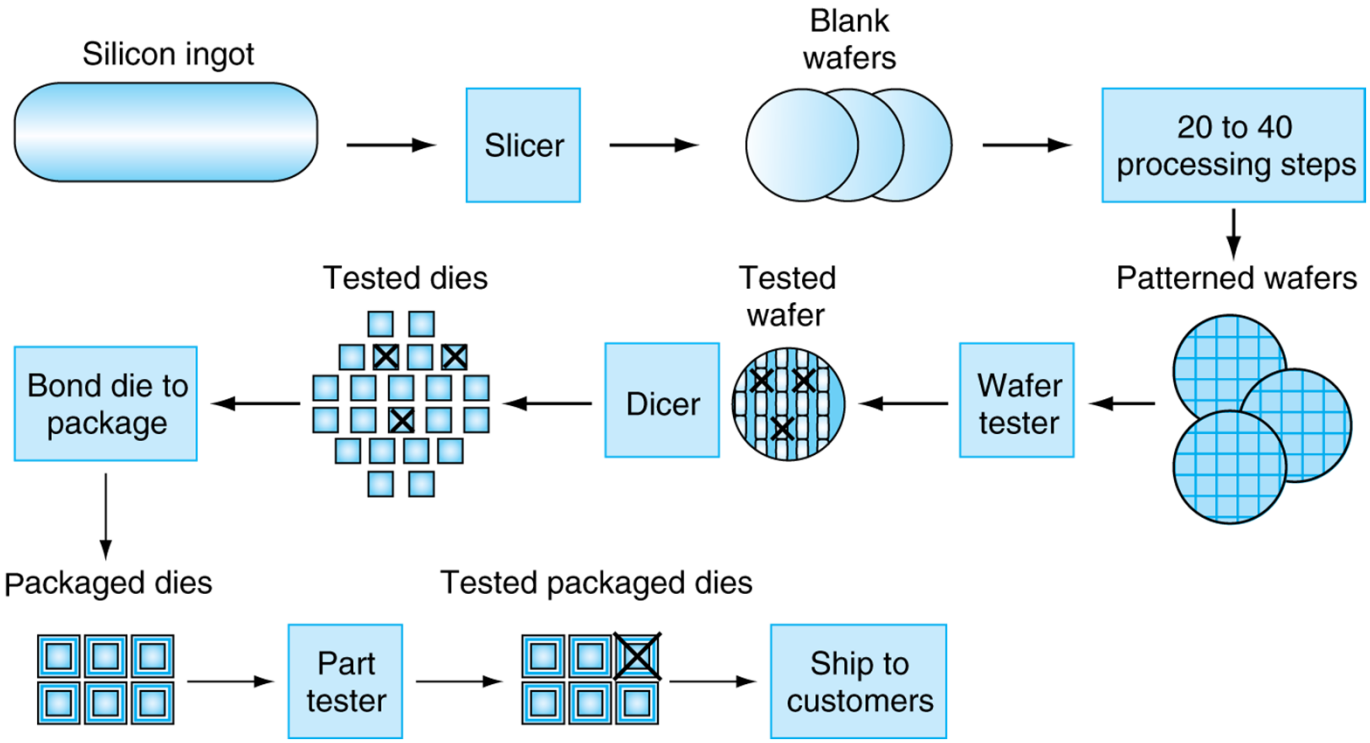
FIRST DRAM
(1103 by Intel 1970)



FIRST MICROPROCESSOR
(4004 by Intel 1971)

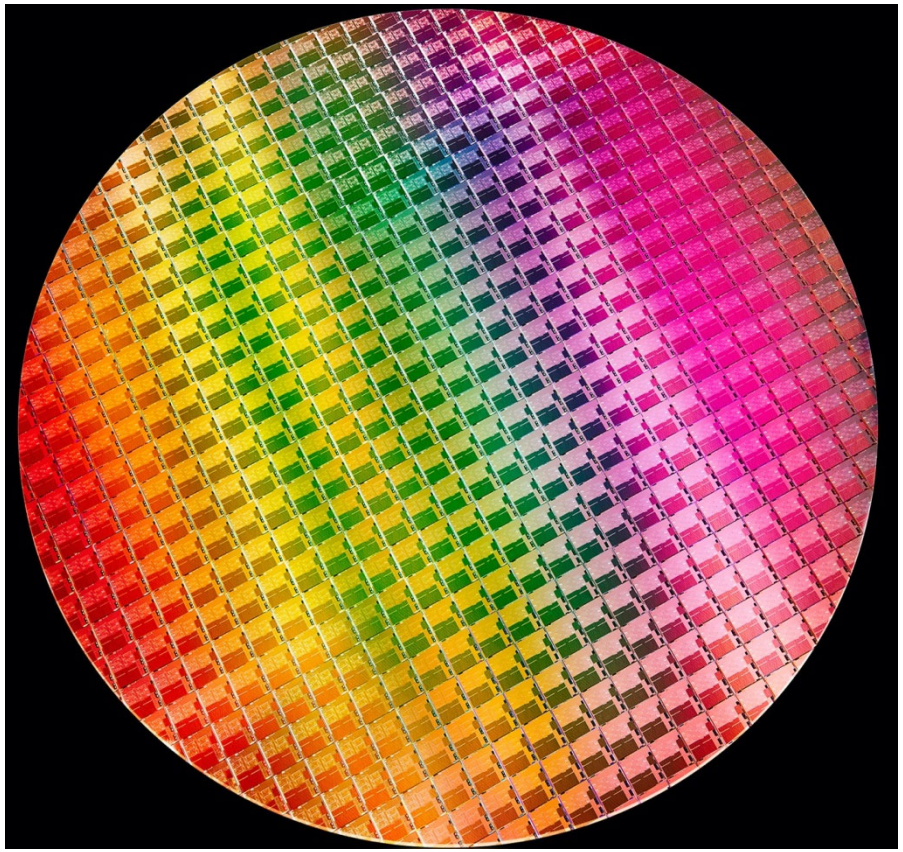


Manufacturing ICs



- Yield: proportion of working dies per wafer

Intel® Core 10th Generation



- 300mm wafer, 506 chips, 10nm technology
- Each chip is 11.4 x 10.7 mm

Integrated Circuit Cost

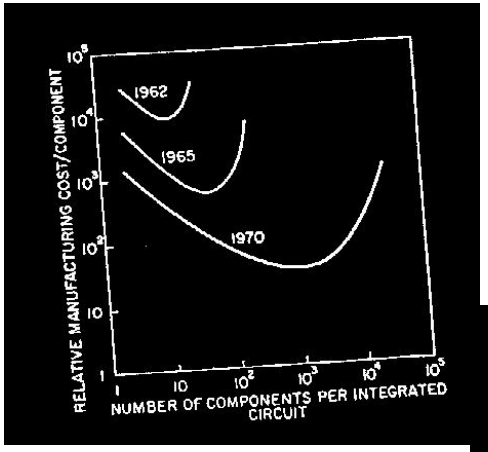
$$\text{Cost per die} = \frac{\text{Cost per wafer}}{\text{Dies per wafer} \times \text{Yield}}$$

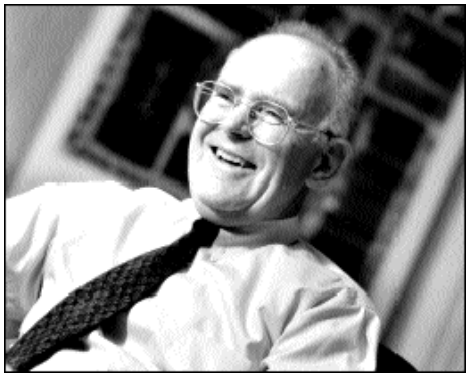
$$\text{Dies per wafer} \approx \text{Wafer area}/\text{Die area}$$

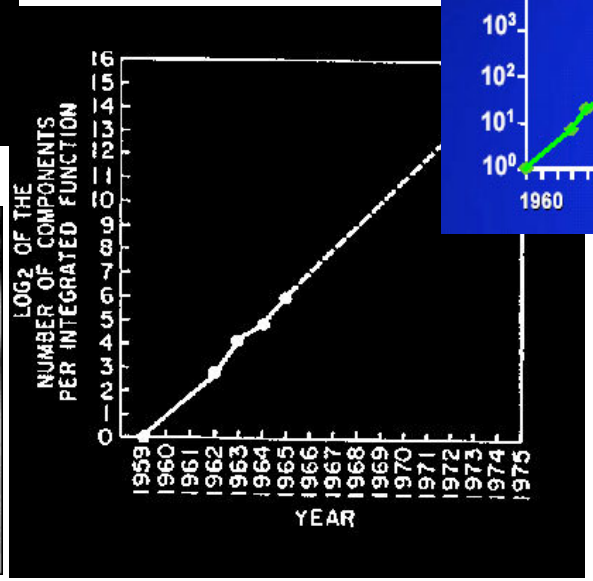
$$\text{Yield} = \frac{1}{(1+(\text{Defects per area} \times \text{Die area}/2))^2}$$

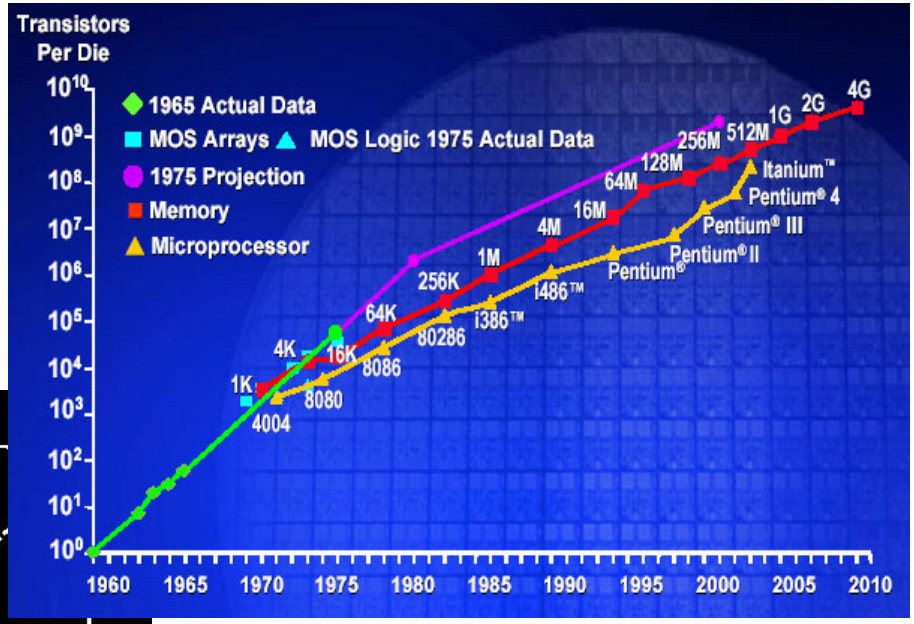
- Nonlinear relation to area and defect rate
  - Wafer cost and area are fixed
  - Defect rate determined by manufacturing process
  - Die area determined by architecture and circuit design

Moore's Law: 2X transistors / "year"



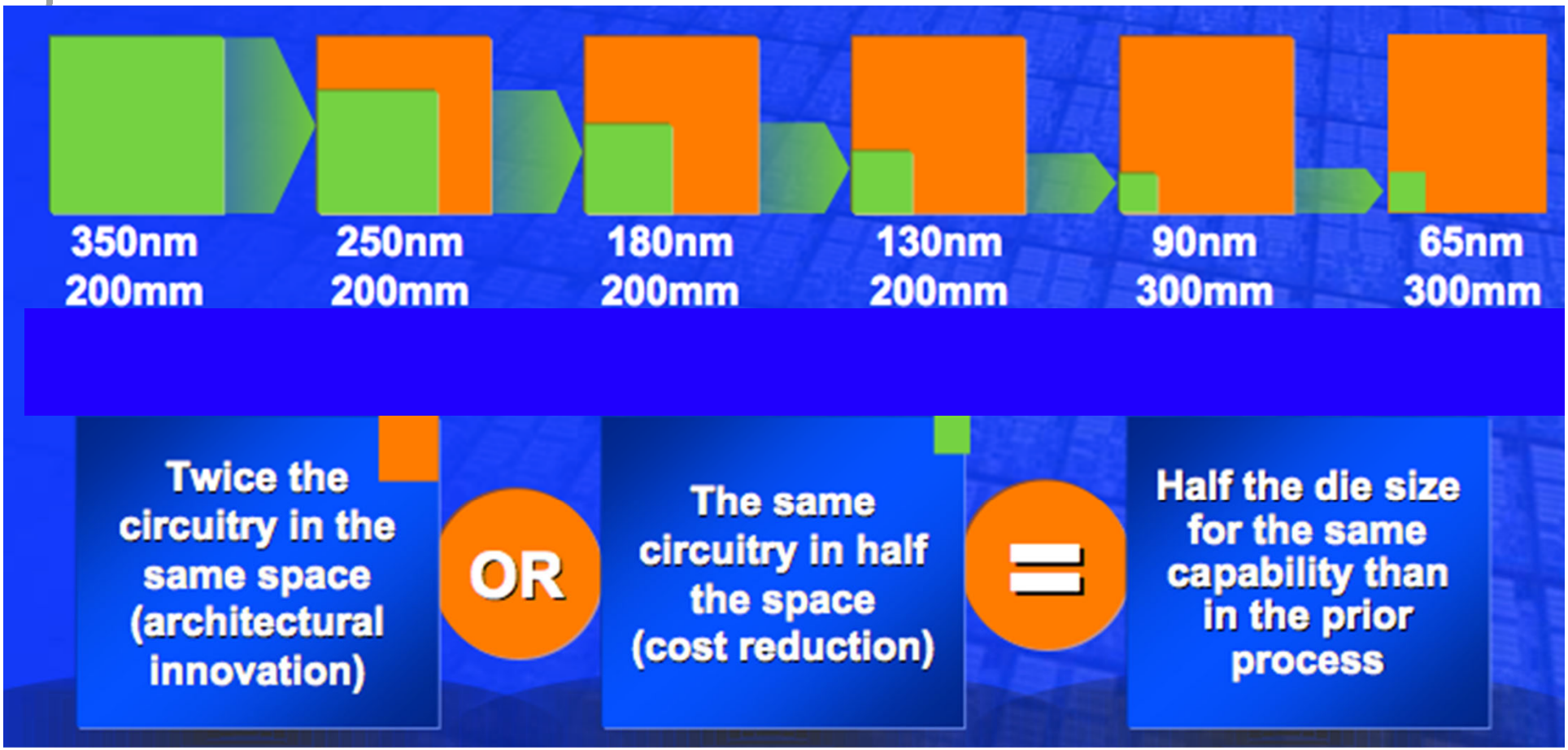






- "Cramming More Components onto Integrated Circuits"
  - Gordon Moore, Electronics, 1965
- # on transistors / cost-effective integrated circuit double every N months ($12 \leq N \leq 24$)

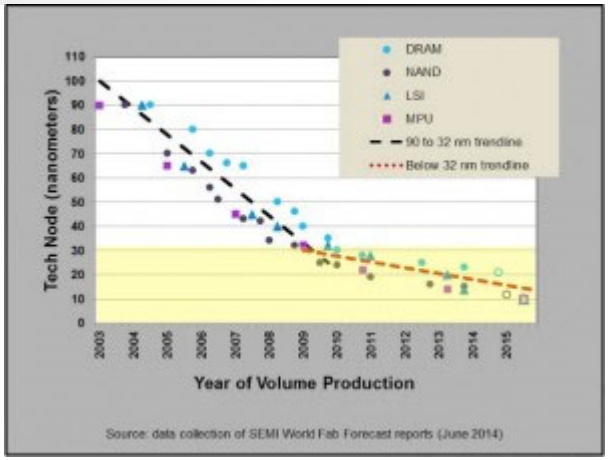
Main Driver: Device Scaling



From: “Facing the Hot Chips Challenge Again”, Bill Holt, Intel, presented at Hot Chips 17, 2005.

Trends in Technology

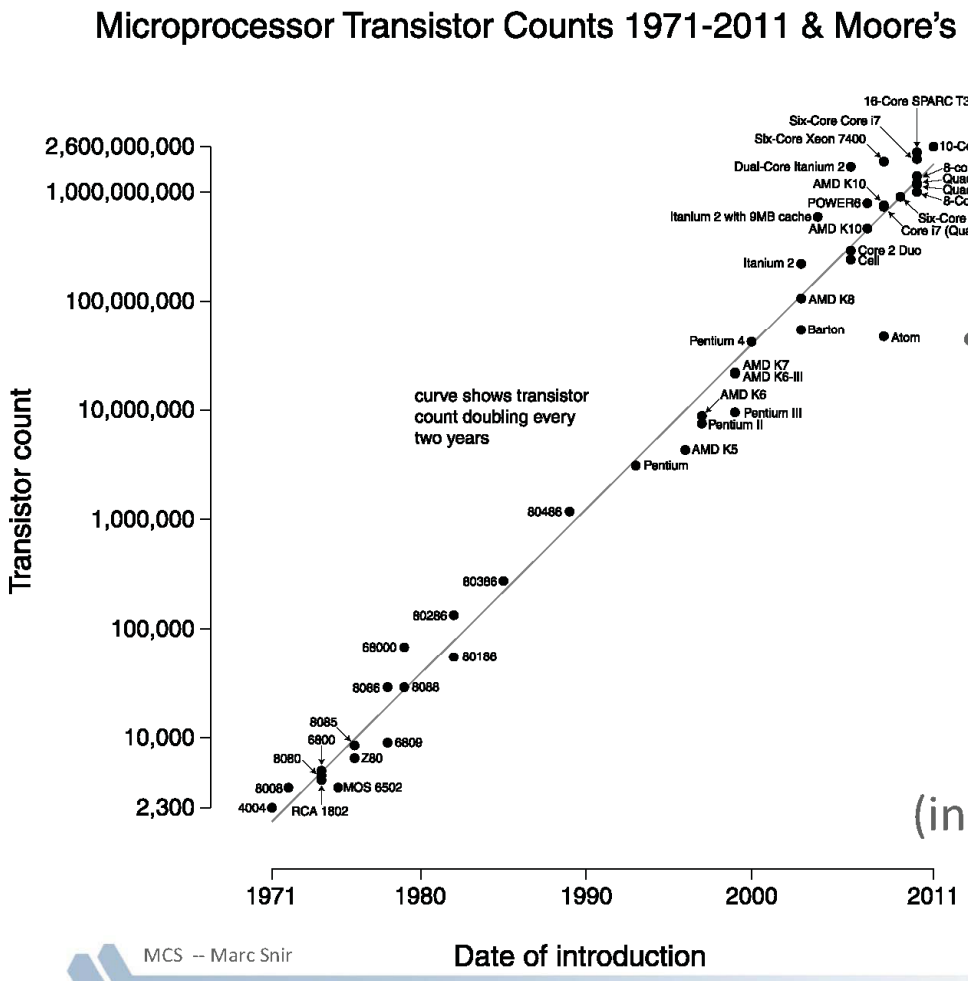
- Integrated circuit technology
  - Transistor density: 35%/year (slowing)
  - Die size: 10-20%/year
  - Integration overall: 40-55%/year (slowing)
  - Moore's Law is no more!



- DRAM capacity: was 25-40%/year (slowed dramatically now)
  - 4Gbit in 2012, 8Gbit in 2015, 16Gbit in 2018, 32Gbit (Samsung) in 2023
- Flash capacity: 50-60%/year
  - 8-21X cheaper/bit than DRAM
- Magnetic disk technology: was 40%/year (~5% now)
  - 15-25X cheaper/bit then Flash
  - 300-500X cheaper/bit than DRAM

The CMOS Age: Moore's Law & Dennard Scaling

Moore's Law: *The number of transistors per chip doubles every 2+ years*



Dennard Scaling:

- Decrease feature size by a factor of λ and decrease voltage by a factor of λ; then
  - # transistors increase by λ^2
  - Clock speed increases by λ
  - Energy consumption does not change

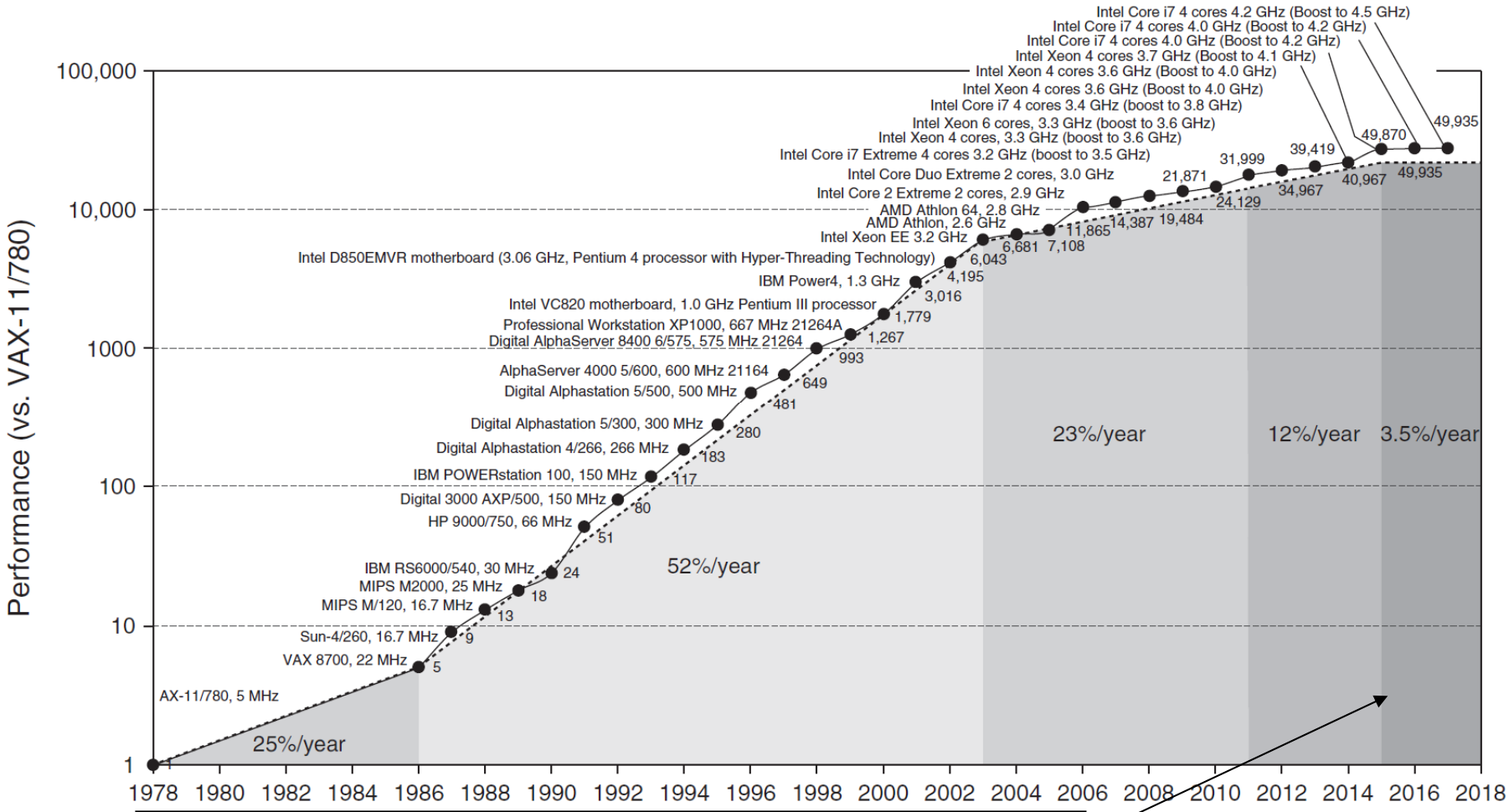
(in reality, voltage decrease was slower; clock speed and energy consumption increased faster)

The Twilight of Moore's Law: Economics

130nm	90nm	65nm/55nm	45/40nm	32/28nm	22/20nm
Altis Semiconductor	Dongbu Hitek	Freescale	Fujitsu	Globalfoundries	Globalfoundries
Dongbu Hitek	Freescale	Fujitsu	Panasonic	Intel	Intel
Freescale	Fujitsu	Globalfoundries	Globalfoundries	Panasonic	Samsung
Fujitsu	Globalfoundries	IBM	IBM	Samsung	ST Microelectronics
Globalfoundries	Grace Semiconductor	Infineon	Intel	ST Microelectronics	TSMC
Grace Semiconductor	IBM	Intel	Renesas (NEC)	TSMC	
IBM	Infineon	Panasonic	Samsung	UMC	
Infineon	Intel	Renesas (NEC)	SMIC		
Intel	Panasonic	Samsung	ST Microelectronics		
Panasonic	Renesas (NEC)	SMIC	Toshiba		
Renesas (NEC)	Samsung	Sony	TSMC		
Samsung	Seiko Epson	ST Microelectronics	UMC		
Seiko Epson	SMIC	Texas Instruments			
SMIC	Sony	Toshiba			
Sony	ST Microelectronics	TSMC			
ST Microelectronics	Texas Instruments	UMC			
Texas Instruments	Toshiba				
Toshiba	TSMC				
TSMC	UMC				
UMC					

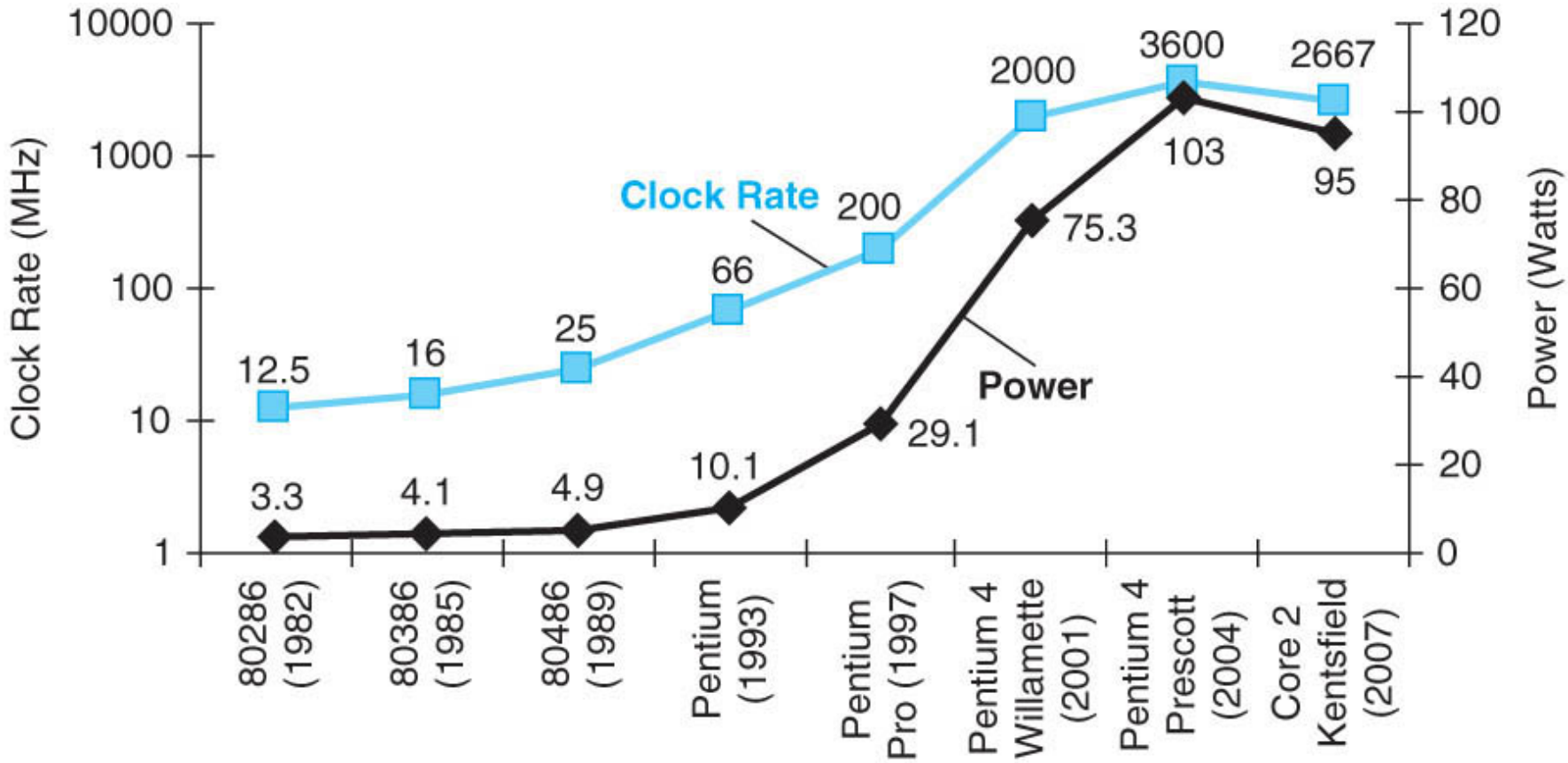
Market volume wall: only the largest volume products will be manufactured with the most advanced technology

Single-Thread Processor Performance



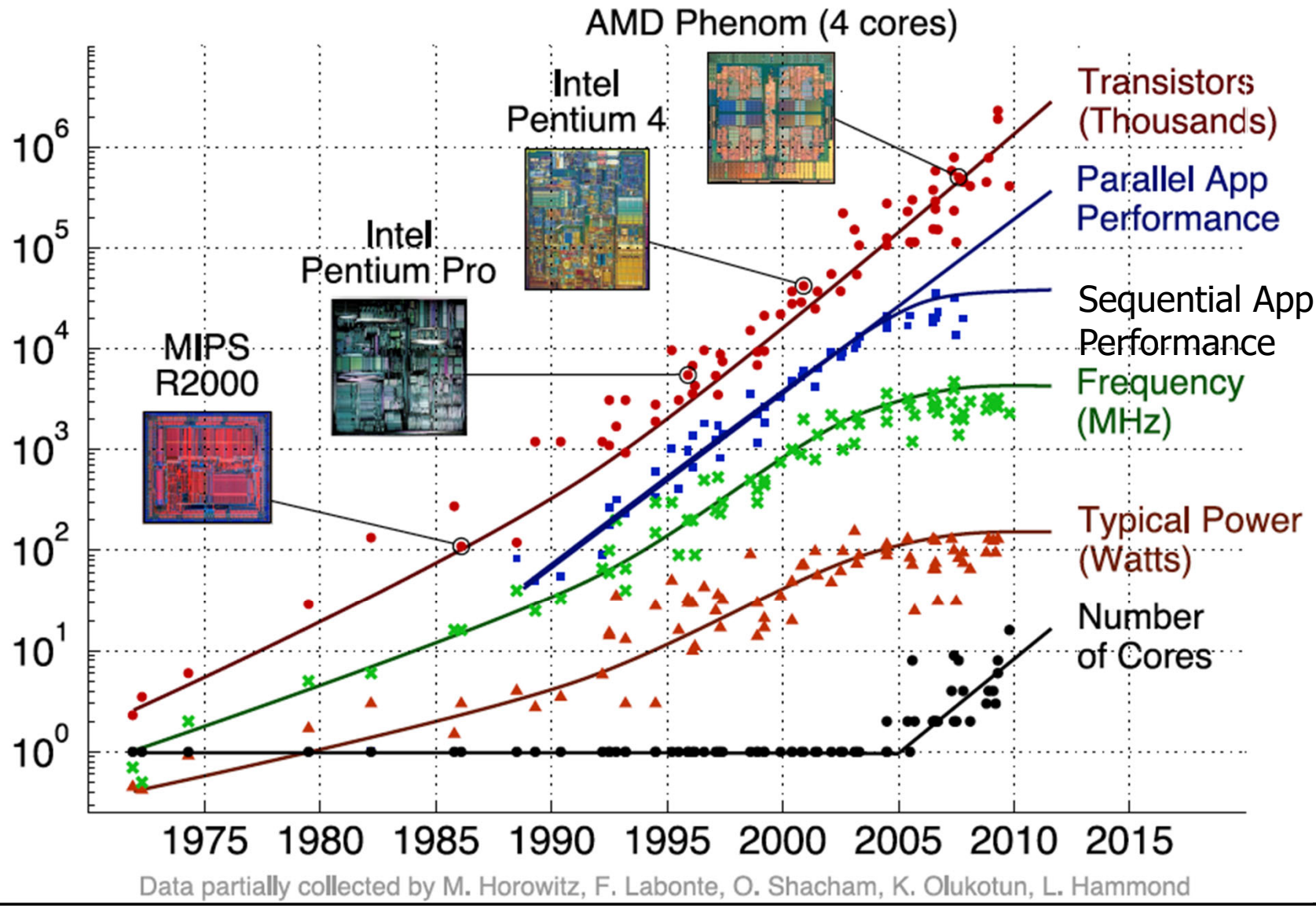
Constrained by power, instruction-level parallelism, memory latency

Limits to Performance: Faster Means More Power



Water Cooling in a Google Data Center

Transition to Multicore



Multicore Machines Today

Examples from Apple's product line:

Mac Pro
28 Intel Xeon W cores

iMac Pro
18 Intel Xeon W cores

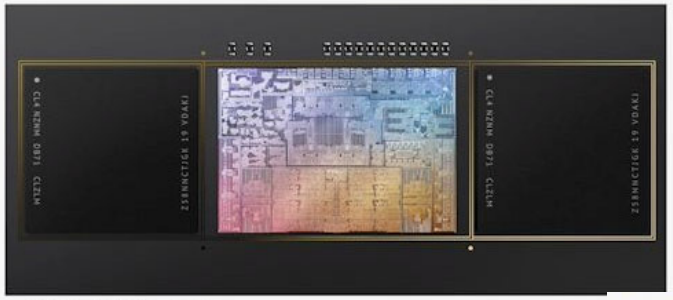
MacBook Pro Retina 15"
8 Intel Core i9 cores

iPad Pro
8 A12X cores
(4 fast + 4 low-power)

iPhone XS
6 A12 cores
(2 fast + 4 low-power)

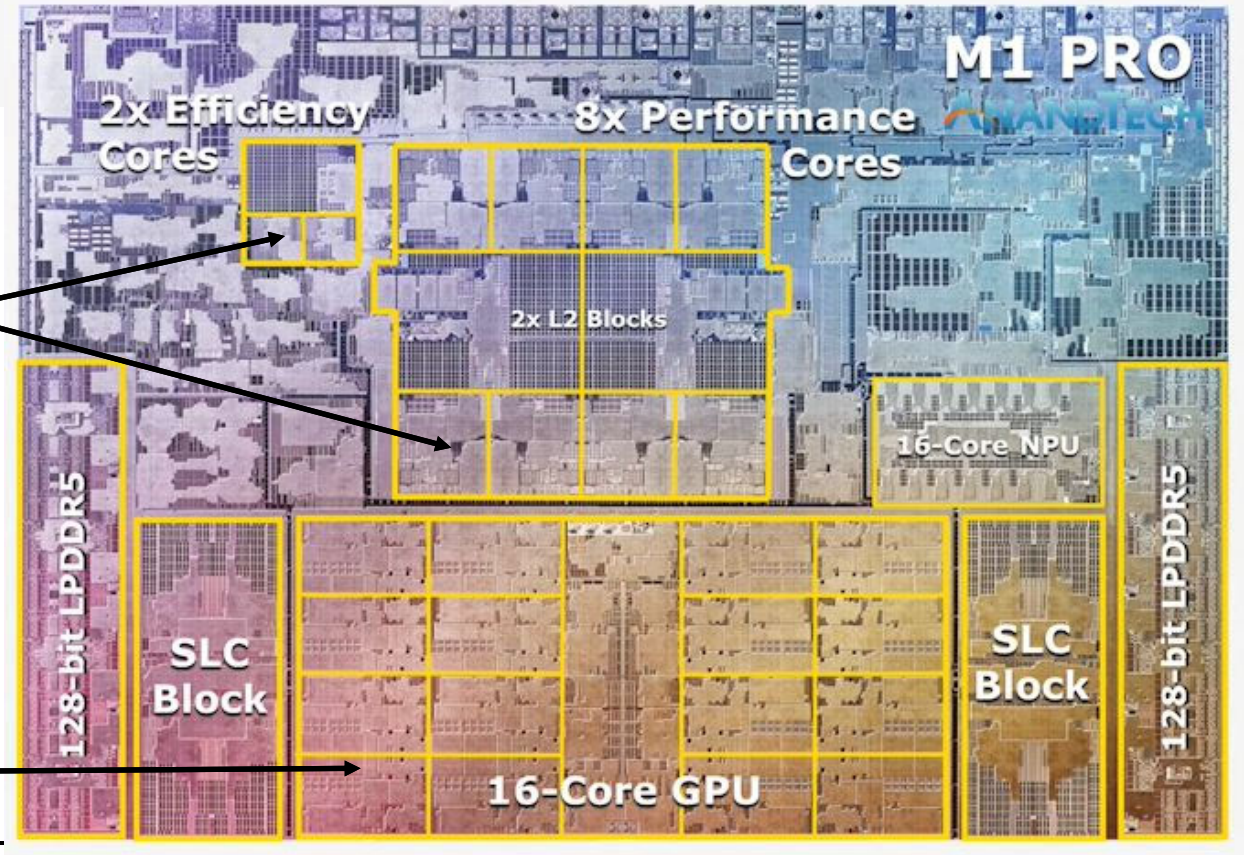
(images from apple.com)

Apple M1 Pro System-on-Chip (2021)

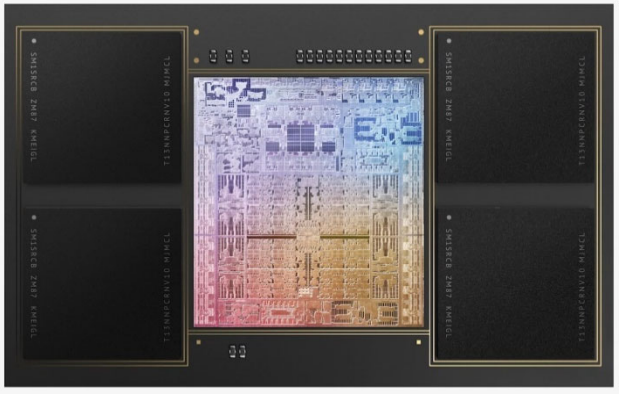


- M1 Pro and 2 DRAM chips on SoC
- M1 Pro: 5 nm process, 33.7B transistors on a 18.95 x 12.98 mm (~245.92 mm^2) die

- ESE 545 focus is on a single-core designs
- ESE 565 will talk about multi-core designs including GPUs

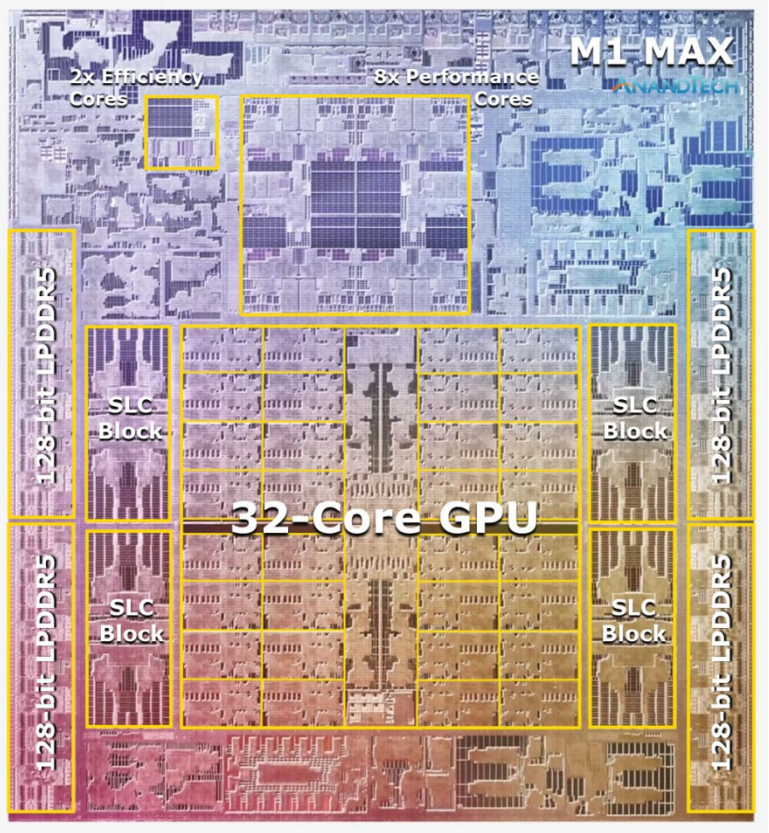


Apple M1 Max System-on-Chip (2021)

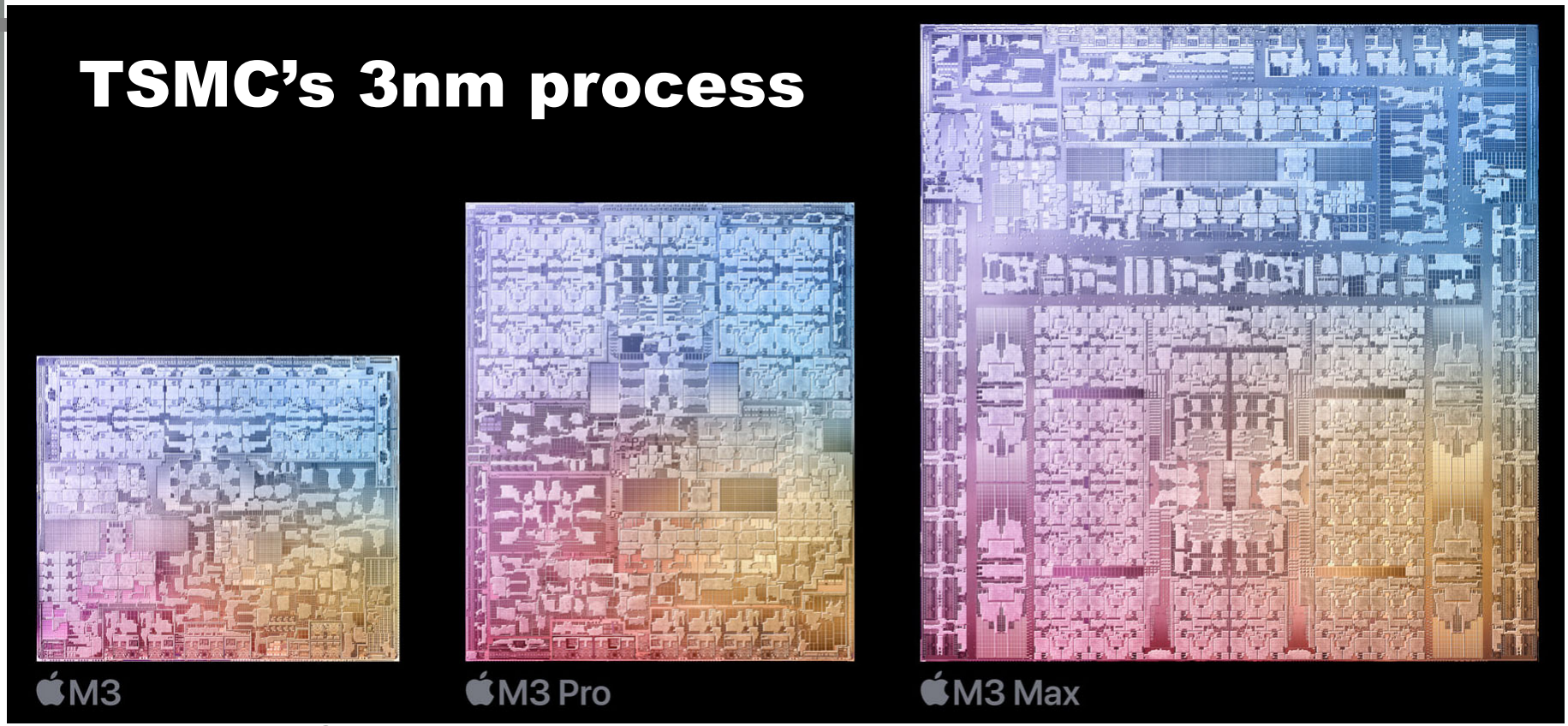


- M1 Max and 4 DRAM chips on SoC
- M1 Max: 5 nm process, 57B transistors on a 19.96 x 21.66 mm (~432.35 mm^2) die

- Intended applications for M1 Pro and M1 Max SoCs are: smartphone, tablet and laptop designs



Apple M3 Chip Series (2023)

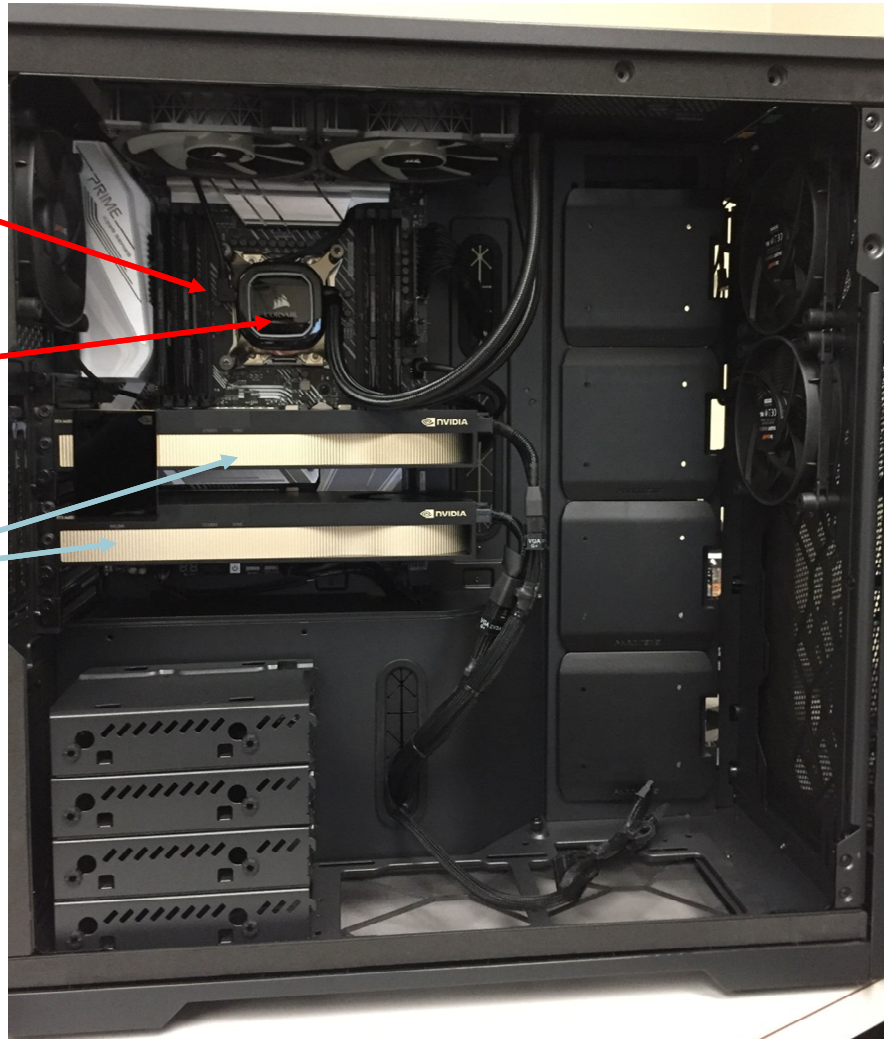


Max. clock: 4.05 GHz

25B transistors	**37B** transistors	**92B** transistors
CPU cores: 6/8 (2/4+4)	CPU cores: 11/12 (5/6+6)	CPU cores: 14/16 (10/12P+4E)
GPU cores: 10	GPU cores: 14/18	GPU cores: 38/40 cores
Power: **20W**	Power: **27W**	Power: **78W**

RTX A6000 GPU Workstation (2021)

- An 18-core Intel 10th generation Core i9-10980XE Cascade Lake host CPU
  - 14 nm technology
  - 3.0-4.8 GHz frequency
  - 165W power consumption (when not overclocked)
- **Liquid cooling** of the Intel i9 CPU for extra stability and low noise
- Two air-cooled RTX A6000 graphics cards, each of which with an 84-core GA102 GPU
  - 8 nm Samsung technology
  - 28.3B transistors
  - 1.4-1.8 GHz frequency
  - 300W max. power consumption/GPU



Intel 14th Generation (2024)

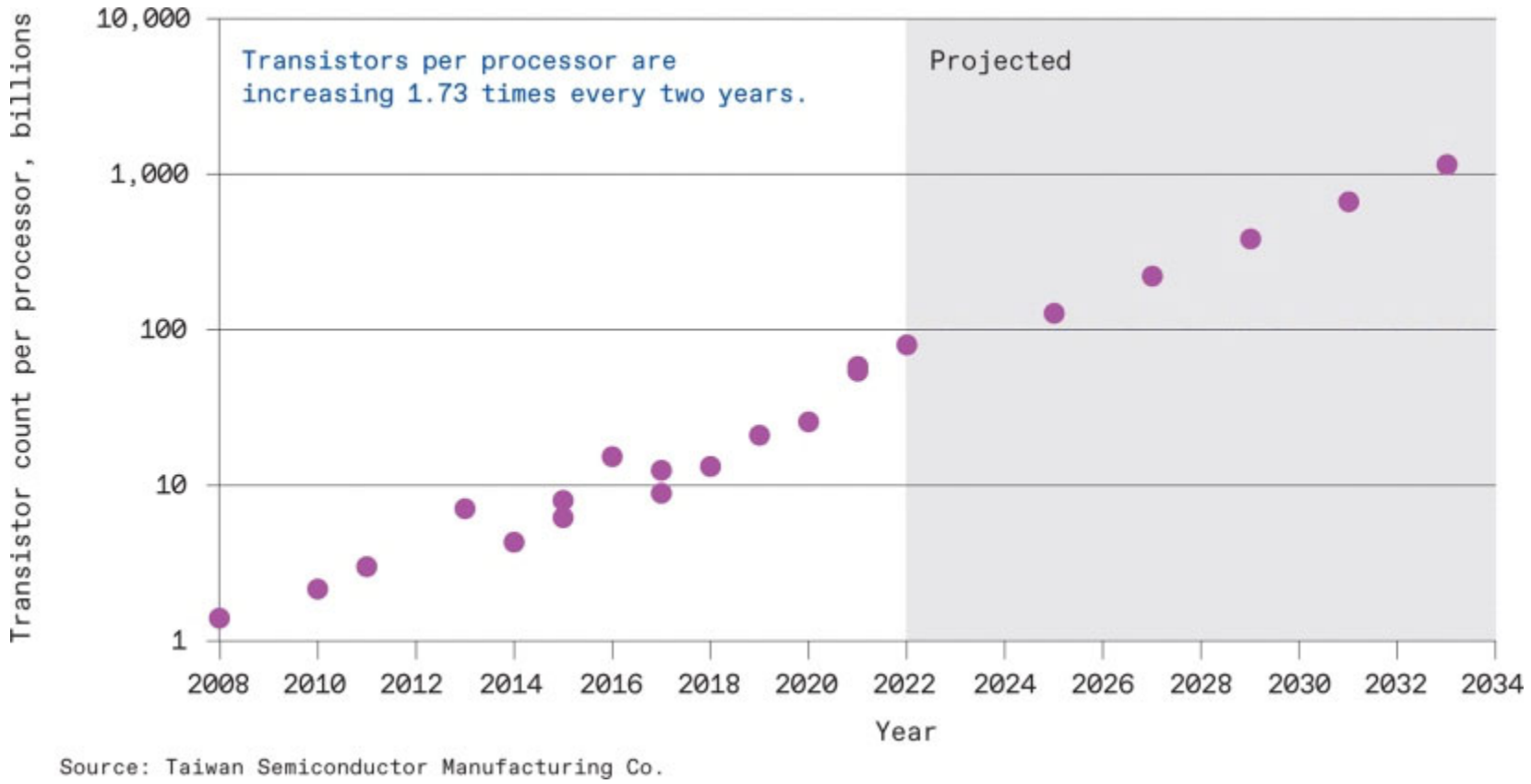
Processor	Processor Cores (P+E)	Processor Threads	Intel Smart Cache (LLC)	Max Turbo Frequency P-cores	Max Turbo Frequency E-cores	Base Frequency P-cores	Base Frequency E-cores	Max Memory Speed	Memory Capacity	Processor Base Power (W)	Max Turbo Power (W)
Core i9-14900HX	**24** (8+16)	32	36MB	Up to **5.8**	Up to 4.1	Up to 2.2	Up to 1.6	DDR5-5600, DDR4-3200	192GB	55	157
Core i7-14700HX	**20** (8+12)	28	33MB	Up to **5.5**	Up to 3.9	Up to 2.1	Up to 1.5	DDR5-5600, DDR4-3200	192GB	55	157
Core i7-14650HX	**16** (8+8)	24	30MB	Up to **5.2**	Up to 3.7	Up to 2.2	Up to 1.6	DDR5-5600, DDR4-3200	192GB	55	157
Core i5-14500HX	**14** (6+8)	20	24MB	Up to **4.9**	Up to 3.5	Up to 2.6	Up to 1.9	DDR5-5600, DDR4-3200	192GB	55	157
Core i5-14450HX	**10** (6+4)	16	20MB	Up to **4.8**	Up to 3.5	Up to 2.4	Up to 1.8	DDR5-5600, DDR4-3200	192GB	55	157

Intel's 7nm process

What's made possible AI revolution?

- Innovation in efficient ML algorithms
- The availability of massive amount of data on which to train neural networks
- **The progress in energy-efficient computing though advancement of semiconductor technology**
  - The IBM Deep Blue Supercomputer which defeated world chess champion Garry Kasparov was implemented with a mix of 0.6 – and 0.35-micrometer technology
  - AlphaGo concurred the game of Go using 40-nm technology
  - Initial version of chatGPT is powered by servers using 5-nm technology, the most recent uses 4-nm technology
  - Within the decade AI revolution will need something close to 1-trillion-transistor GPU (~10 times more than in 2024)

Transistors per processor

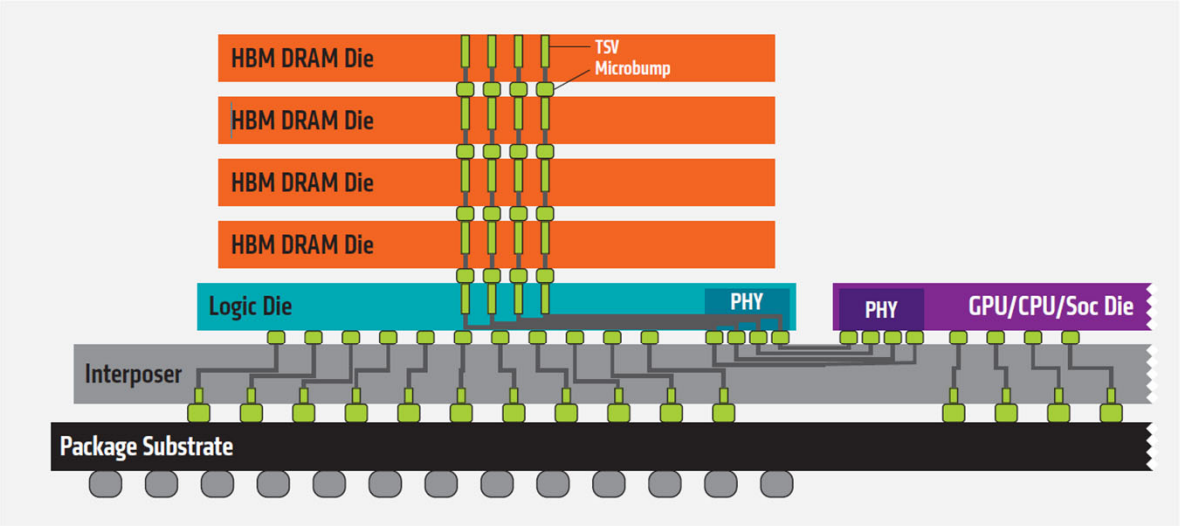


Source: Taiwan Semiconductor Manufacturing Co.

3D DRAM Stacking Technologies

Recent enabling technology: 3D stacking of DRAM chips

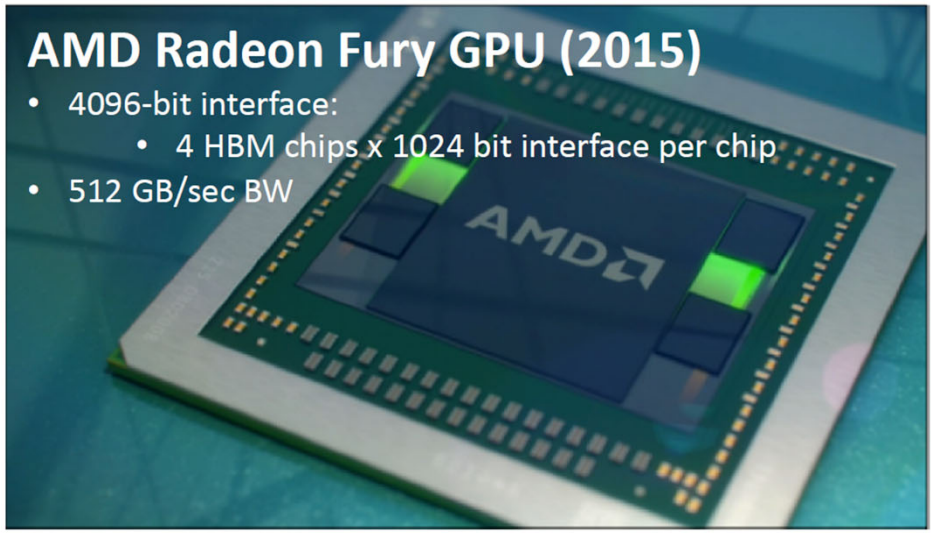
- DRAMs connected via through-silicon-vias (TSVs) that run through the chips
  - TSVs provide highly parallel connection between logic layer and DRAMs
- Base layer of stack "logic layer" is memory controller, manages requests from processor
- Silicon "interposer" serves as high-BW interconnect between DRAM stack and processor

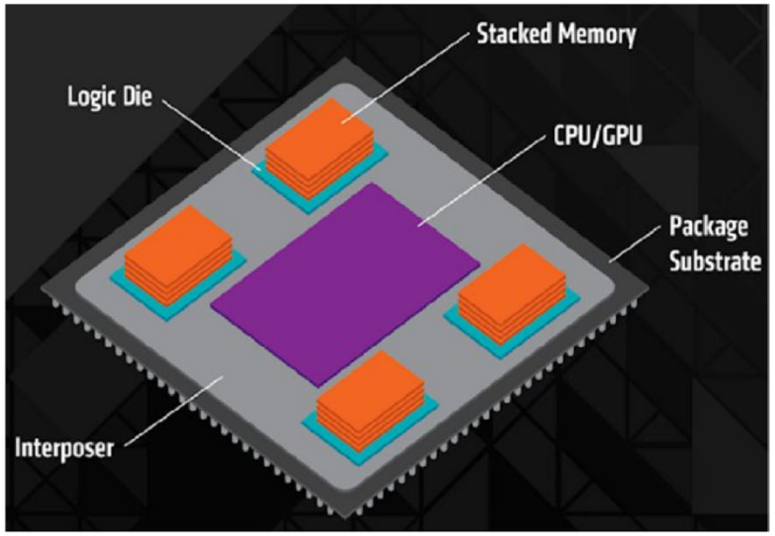


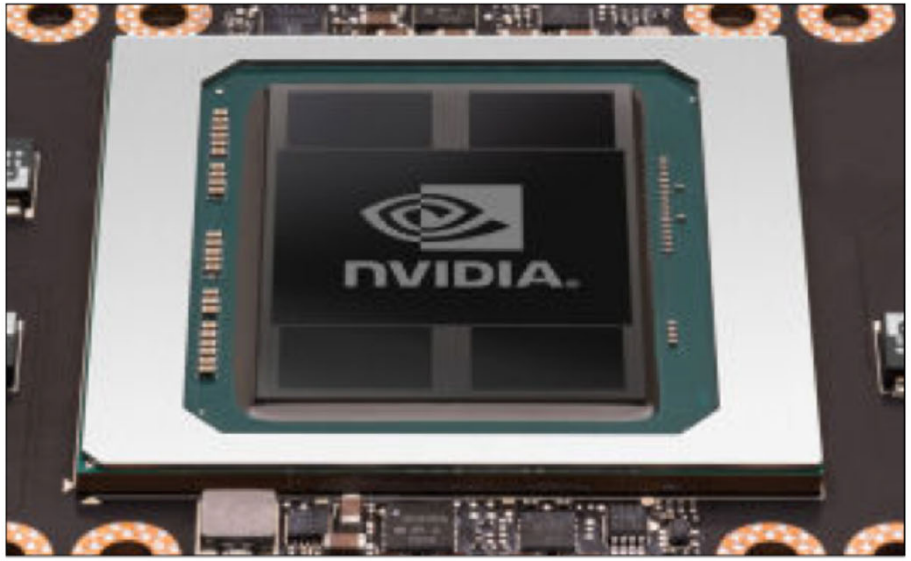
Technologies:

- Micron/Intel's Hybrid Memory Cube (HMC)
- AMD's High-Bandwidth Memory (HBM): 1024 bit interface to stack

3D DRAM Stacking in GPUs





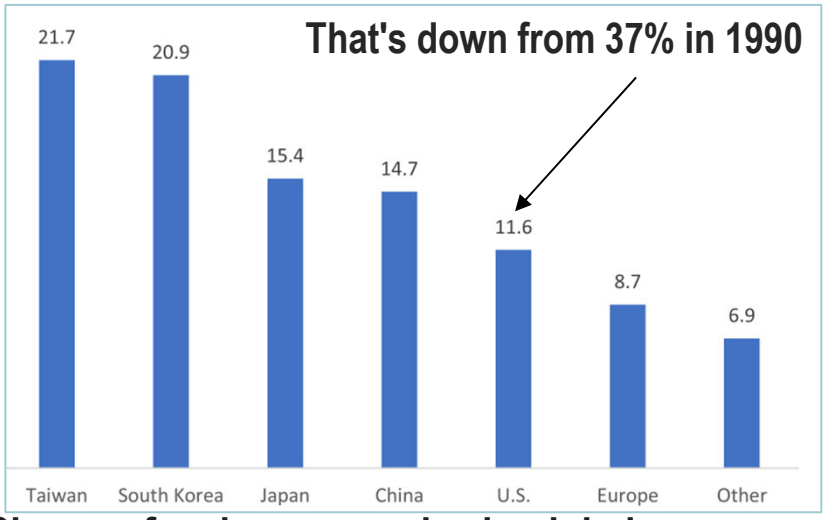


NVIDIA P100 GPU (2016)

- 4096-bit interface:
  - 4 HBM2 chips x 1024 bit interface per chip
- 720 GB/sec peak BW
- 4 x 4 GB = 16 GB capacity

Current and Next-Generation Chips

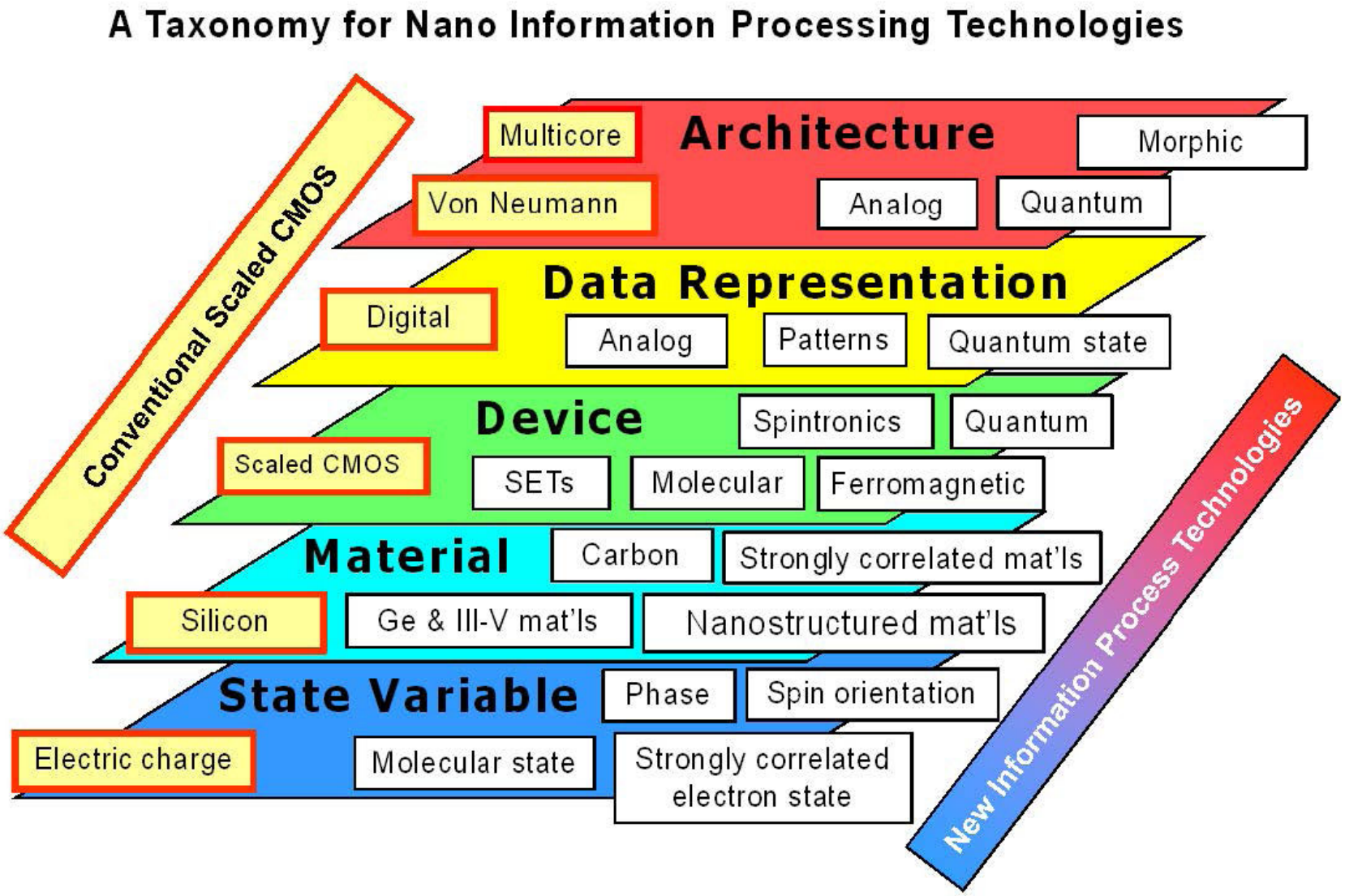
- **In 2020, TSMC and Samsung introduced 5 nm chips and 3 nm chips in 2022**
  - TSMC produces chips for AMD, Apple, Qualcomm, and Huawey
- **Intel 7 nm chips were delayed until 2022**
- **IBM announced its first 7 nm 18B transistors POWER10 chips in Sept. 2021 (manufactured by Samsung)**
  - In May 2021, IBM announced the the development of the first 2 nm nanosheet technology



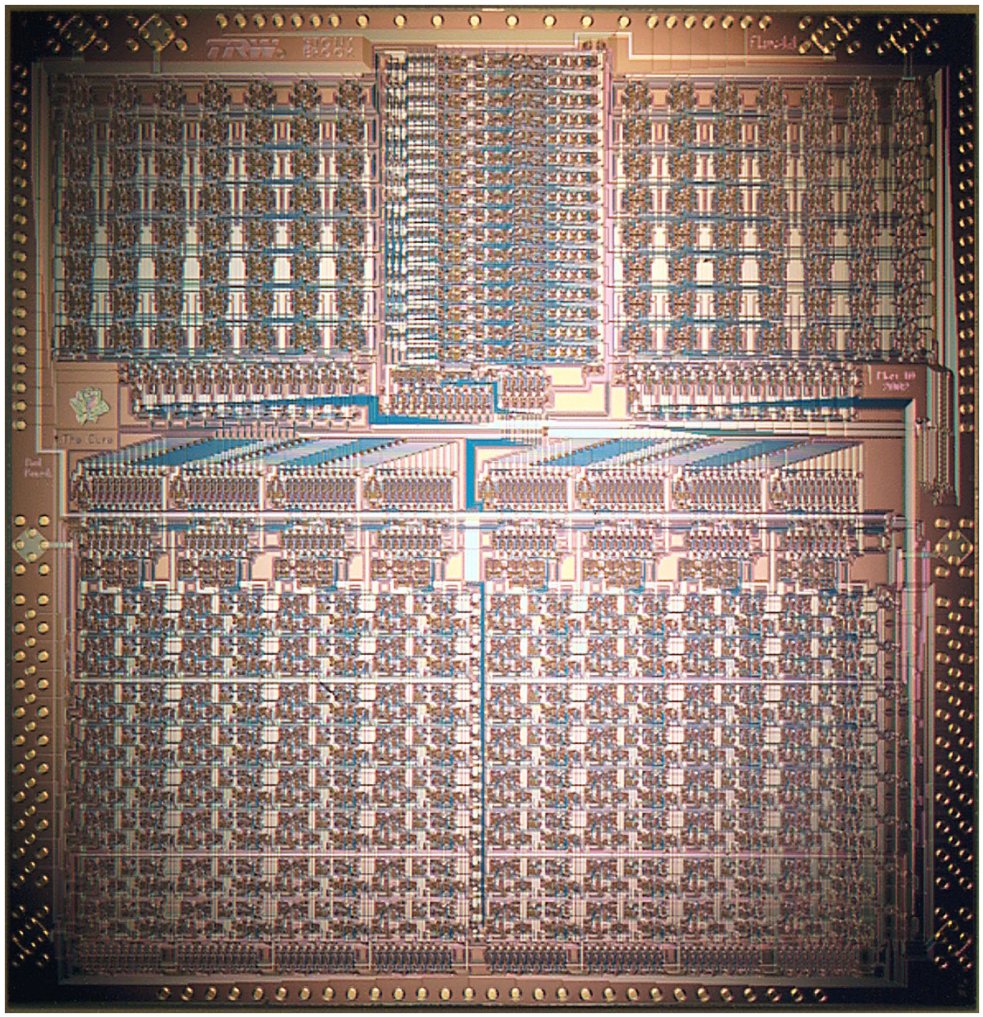
Shares of major economies in global semiconductor manufacturing capacity (2020)

- **In 2020, TSMC announced a plan to spend $12 billion to build a semiconductor plant in Arizona**
- **In 2021, Samsung announced a plan to spend $17 billion to build a semiconductor plant in Texas**
- **In 2022, Intel announced a plan to build a $20B semiconductor foundry in Columbus, Ohio**

What's Next After CMOS?



Superconductor Flux Quantum Computing: 20 GHz 8-bit RSFQ Microprocessor (2002)



M. Dorojevets, "An 8-bit FLUX-1 RSFQ microprocessor built in 1.75-μm technology," 2002

- **Designed at Stony Brook University (ECE and Physics dept. teams) in collaboration with TRW (CA)**
- **Technology**: 1.75-μm min. JJ size (TRW)
- **Circuit family**: RSFQ
- **Target Clock**: 20 GHz
- **JJ count**: 63,107
- **Power**: ~9.5 mW at 4.2K
- **Chip size**: 10.35 mm x 10.65 mm

- More about flux quantum computing at **www.quantarctic.com**

What About Bio/Neuro-Inspired Computing?

Human Brain

Laid out flat – 0.25 m^2 (not 3D) 2mm thick with 6 layers
wire diameter 10 nm, neuron diameter 4 microns
100 billion neurons (1 x 10^{11}),
0.15 quadrillion connections (1.5 x 10^{14})
150,000 km of wiring
Wt 1.3 Kg
Frequency 30 hz
Power 25 W ← System architects eat your heart out

Continuous failure – 85,000 die/day, millions misfire/sec

Errors in calculations – make mistakes (examples visual illusions)

Self correction – sensor fusion, redundant calculation

Self healing – reprogram around physical damage

From A. Geist's presentation at Exaflops workshop (CA), Aug. 3, 2011

Technology Trends Summary

- Most of last 50 years, Moore's Law ruled
  - Technology scaling allowed continual performance/energy improvements without changing software model
- Last decade, technology scaling slowed/stopped
  - Dennard (voltage) scaling over (supply voltage ~fixed)
  - Moore's Law (cost/transistor) over?
  - No competitive replacement for CMOS anytime soon
  - Energy efficiency constrains everything
- No "free lunch" for software developers, must consider:
  - Parallel systems
  - Heterogeneous systems

Acknowledgements

- These slides contain material developed and copyright by:
  - Morgan Kauffmann (Elsevier, Inc.)
  - Arvind (MIT)
  - Krste Asanovic (MIT/UCB)
  - Joel Emer (Intel/MIT)
  - James Hoe (CMU)
  - John Kubiatowicz (UCB)
  - David Patterson (UCB)
  - Justin Hsia (UCB)
  - Marc Snir (ANL)
  - Mikhail Dorojevets (SBU)