

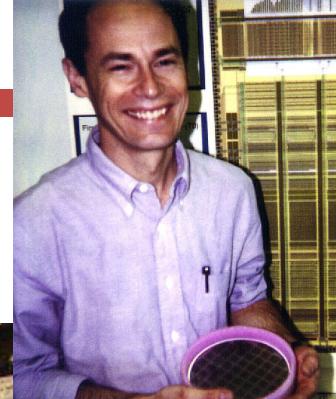
**EECS 151/251A**  
**Spring 2023**  
**Digital Design and Integrated Circuits**

Instructor:  
John Wawrzynek

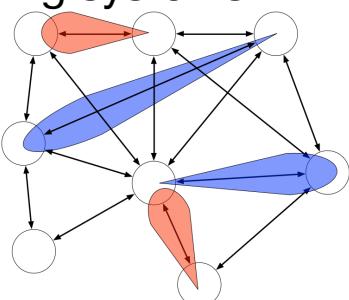
Lecture 1

# John Wawzynek

## Professor of EECS



- Profession Musician in New York
- JPL/NASA – space craft data systems
- PhD Caltech – electronic music
- Berkeley faculty since 1989
  - IC design, signal processing systems
  - High performance computer design
  - Reconfigurable computing
  - Wireless system design





## Class Goals and Outcomes

# *What this class is all about?*

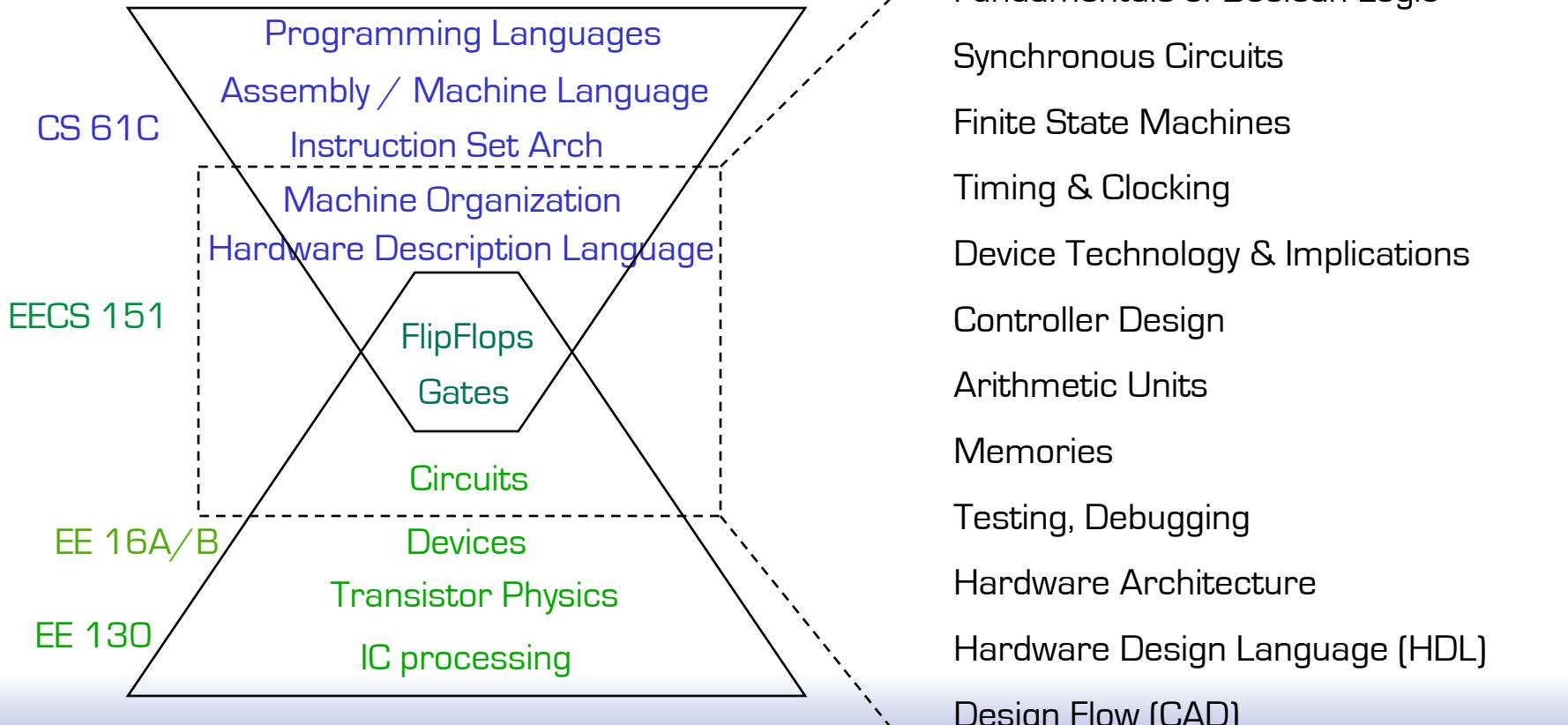
- **Introduction to digital integrated circuit and system engineering**
  - Key concepts needed to be a good digital designer
  - Discover your own creativity!
- **Learn models that allow reasoning about design**
  - Manage design complexity through abstraction and understanding of tools
  - Allow analysis and optimization of the circuit's performance, power, cost, etc.
- **Learn how to make sure your circuit and system works**
  - *Do you want to be the one that messes up a 1 billion transistor chip?*

*Digital design is not a spectator sport!  
Learn by doing.*

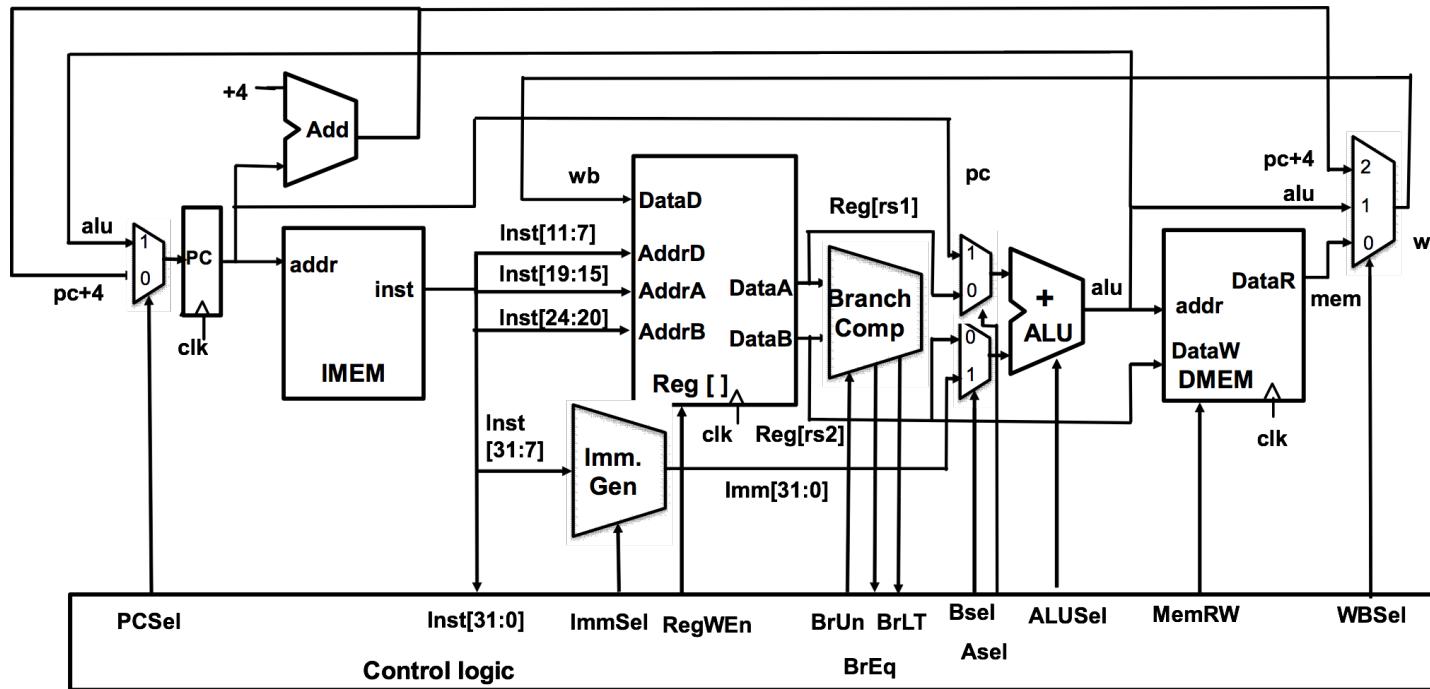
# Prerequisites

- ❑ CS61C
  - Boolean logic, RISC-V ISA
  - We will review combinational and sequential logic, and RISC-V design (with more details)
- ❑ EE16A/B
  - Digital gates, RC networks
  - We will review transistor operation and design of CMOS circuits

# Course Focus



# CS61C Background – RISC-V ISA and microarchitecture



- Used in lectures as a design example, and you'll implement in project.  
We review the microarchitecture, and discuss the design in detail.



# ADMINISTRATIVIA

# Teaching Staff



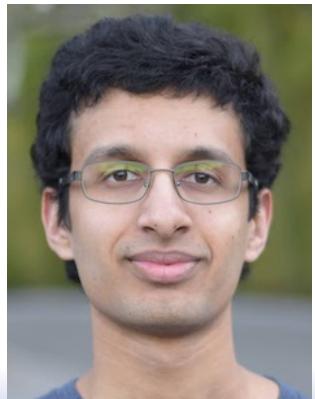
**Yukio Miyasaka**  
Discussion,  
FPGA Labs  
Office Hours:  
TBA



**Professor John  
Wawrzynek  
(Warznek)**  
[johnw@berkeley.edu](mailto:johnw@berkeley.edu)  
Office Hours:  
Th 11AM  
& by appointment.



**Reader:**  
*Daniel Endrads*



**Rahul Kumar**  
FPGA Lab,  
Discussion  
Office Hours:  
TBA



**Dhruv Vaish**  
FPGA Lab  
Website,  
Git repo  
Office Hours:  
TBA



**Lux Zhang**  
ASIC Lab  
Discussion,  
Office Hours:  
TBA

# **Enrollment:**

- Our plan is to admit all those on the waitlist with the proper prerequisites
- If you are waitlisted for the lecture, make sure you are also waitlisted for a lab session:
  - These FPGA labs will be fill up:
    - LAB 001 28444 M 11:00AM-2PM
    - LAB 002 28445 M 2:00-5PM
  - These FPGA labs will be less crowded (please consider taking these):
    - LAB 003 28446 M 5:00-8PM
    - LAB 004 30638 M 8:00-11AM
- While we are processing the waitlist, attend discussion and labs
- Concurrent Enrollment requests will only be processed after speaking with me *in person*.

# Course Information

- Basic Source of Information, class website:

<http://inst.eecs.berkeley.edu/~eecs151/sp23/>

- Lecture notes and recordings
- Assignments and solutions
- Lab and project information
- Exams
- Ed Discussion Forum
- Many other goodies ...



Print only what you need: Save a tree!

# Class Organization

- ❑ Lectures (*TuTh 9:30-11AM*)
- ❑ Discussion sessions (*F 2-3PM, 3-4PM, 540AB Cory*)
- ❑ Office hours (*check website*)
- ❑ Weekly Problem Sets
- ❑ Labs
  - ❑ FPGA (*Mon 8-11AM, 11-2PM, 2-5PM, 5-8PM*)
  - ❑ ASIC (*Tue 11-2PM*)
  - ❑ or both
- ❑ Design project
- ❑ 2 Exams (1 midterm and 1 final)

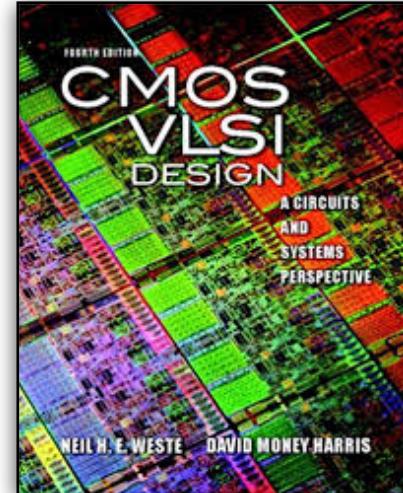
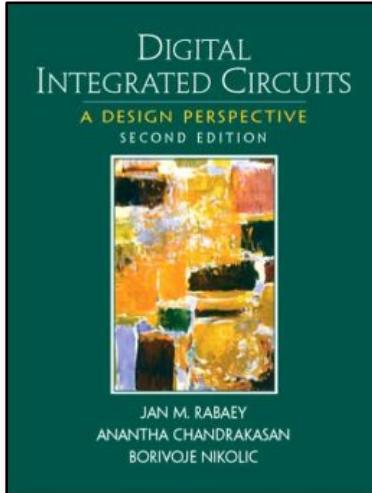
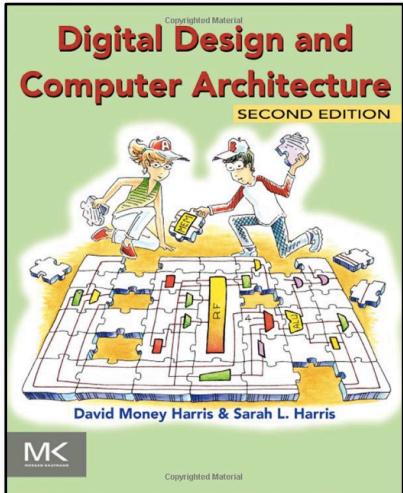
# Lectures



- ❑ Lectures are *critical*:
  - ❑ No textbook
  - ❑ All course content (minus the labs) are contained in the lectures
  - ❑ Problem sets based around lecture material
  - ❑ Discussion sessions reinforce lecture material and provide hints on problem sets
- ❑ Slides available on website before the lecture
- ❑ Best practice is to download slides before lecture and annotate them during lecture (or simply take notes)
- ❑ Lectures intended to be interactive
  - ❑ **Ask questions**, offer comments! Tell me to slow down or speed up
- ❑ Lectures to be recorded, and available on class website
- ❑ We highly encourage you to attend lectures in person. If attendance falls, we will stop recordings.

# Class Textbooks

No Required Book this semester



*Recommended  
(previously required)*

❑ Useful LA lab reference (EE151/251A):

- Erik Brunvand: Digital VLSI Chip Design with Cadence and Synopsys CAD Tools

# *Discussion Sessions*

- Start this week (Friday)!
- Review of important concepts from lecture (remember no text book)
- Help with problem sets
- Friday 2-3PM, 3-4PM, 540AB Cory
- Check piazza for zoom link.
- Will be recorded and posted.



# Problem Sets

- ❑ Approximately 12 over the course of the semester (one per week)
- ❑ Posted on Friday, due on Monday 11:59pm, 10 days later
- ❑ Essential to understanding of the material
  - Hence take them seriously!
  - Ok to discuss with colleagues but need to turn in your own work / write-up / explanations
- ❑ Late turn-in: 20% point deduction per day, except with documented medical excuse
- ❑ Solutions posted Friday of due week



# Labs

- ❑ Enroll in FPGA or ASIC or both (or another in a later semester)
- ❑ 5 FPGA / 6 ASIC lab exercises, done solo
  - Lab report (check off) due by next lab session
- ❑ Design Project lasts ~7 weeks, done with partner
  - Project demo/interview during RRR week
  - Project report due around same time
- ❑ All Labs start next week!



# **Exams**

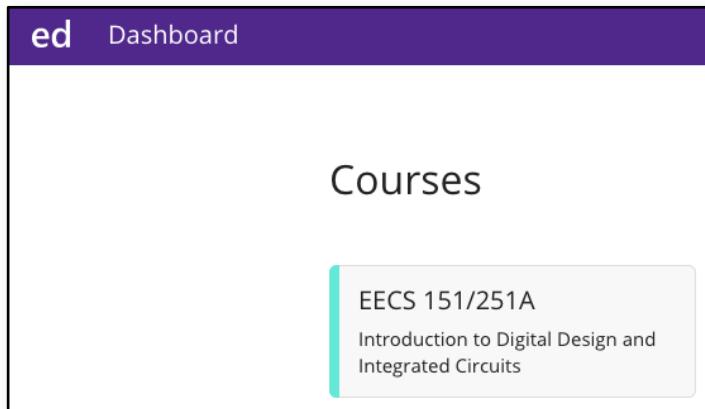
- Exam 1 scheduled in evening. No lecture that day.
- (Tentative) Thur March 9, 6-9PM
- Exam 2 during normal final exam slot:
  - Friday May 10, 11-2PM.



*Exams formats TBA*

# Class Discussions

- ❑ ed for interactions between Instructors and fellow students  
For fastest response **post your questions on ed.**



[edstem.org](https://edstem.org)

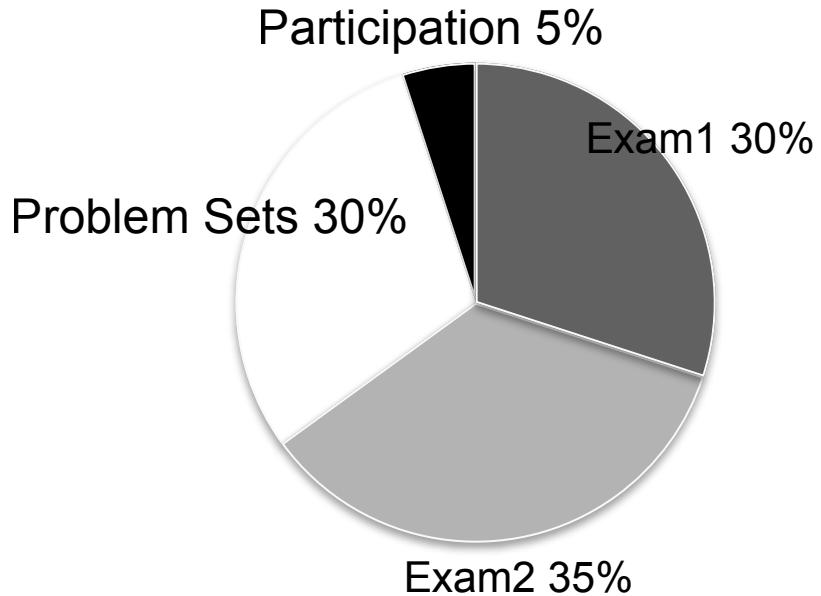
(make sure to logon asap - if you don't want to miss any of the action)  
<https://edstem.org/us/courses/34322/discussion/>

# Cheating Policy

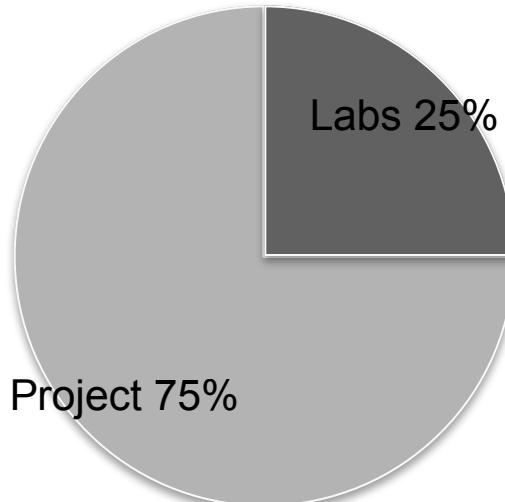
- Details of our cheating policy on the class web site. Please read it and ask questions.
- If you turn in someone else's work as if it were your own, you are guilty of cheating. This includes problem sets, answers on exams, lab exercise checks, project design, and any required course turn-in material.
- Also, if you knowingly aid in cheating, you are guilty.
- We have software that compares your submitted work to others.
- However, it is okay to discuss with others lab exercises and the project (obviously, okay to work with project partner). Okay to discuss homework with others. But everyone must turn in their own work.
- Do not post your work on public repositories like github (private o.k.)
- If we catch you cheating, you will get **negative points** on the assignment: It is better to not do the work than to cheat!  
If it is a midterm exam, final exam, or final project, you get an F in the class.  
All cases of cheating reported to the office of student conduct.

# *Grading Breakdown*

*Lecture*



*Labs*



# *Participation*

1. Be present at lectures
  - ask questions, offer comments
2. Participation in discussion sessions
3. Post to ed
  - help answer fellow student questions about problem sets, labs, project
  - contribute testing or other code to help in project debug



# *Tips on How to Get a Good Grade*

The lecture material is not the most challenging part of the course but is very important.

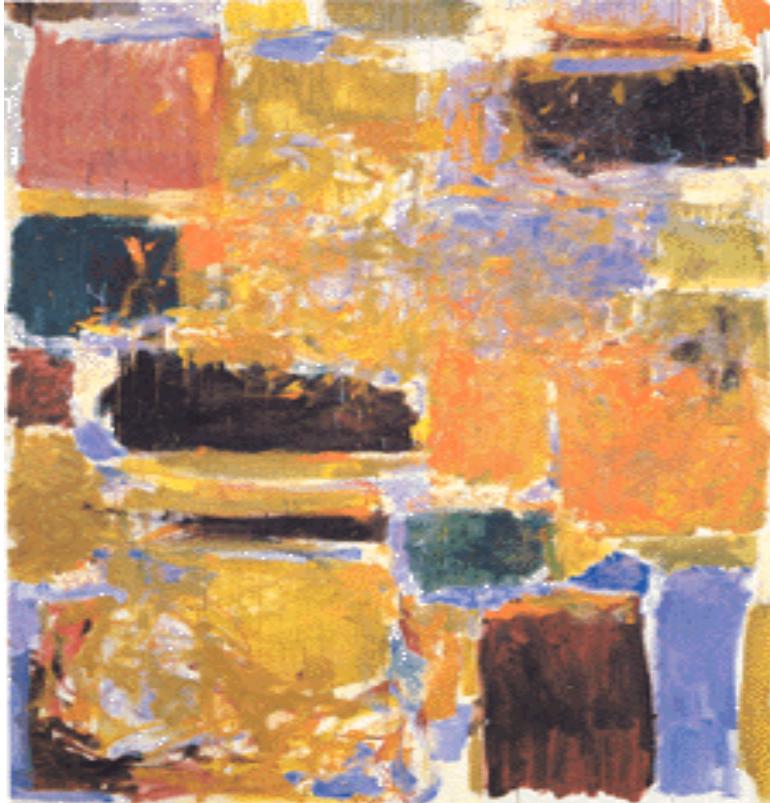
- You should be able to understand everything as we go along.
- Do not fall behind in lecture and tell yourself you “will figure it out later from the notes”.
- Notes will be online before the lecture (usually the night before). Look at them before class.
- Ask questions in class and stay involved - that will help you understand. Come to office hours to check your understanding or to ask questions.
- Complete all the homework problems - even the difficult ones. Some problems go beyond lecture.
- The exams will test your depth of knowledge. You need to understand the material well enough to apply it in new situations.

With a few exceptions, you need to enroll in both the lab and the lecture.

- Take the labs very seriously. They are an integral part of the course.
- Choose your project partner carefully. Your best friend may not be the best choice!
- Most important (this comes from 30+ years of hardware design experience):
  - Be well organized and neat with homework, labs, project.
  - In lab, add complexity a little bit at a time - always have a working design.
  - Don’t be afraid to throw away your design and start fresh.

# *Getting Started*

- ❑ Discussions start this week, labs next week.
- ❑ PS 1 assigned later this week
- ❑ Login to ed as soon as possible
- ❑ Register for your EECS151 class account at  
[inst.eecs.berkeley.edu/webacct](http://inst.eecs.berkeley.edu/webacct)



# Digital Integrated Circuits and Systems – From the Past to the Future ...

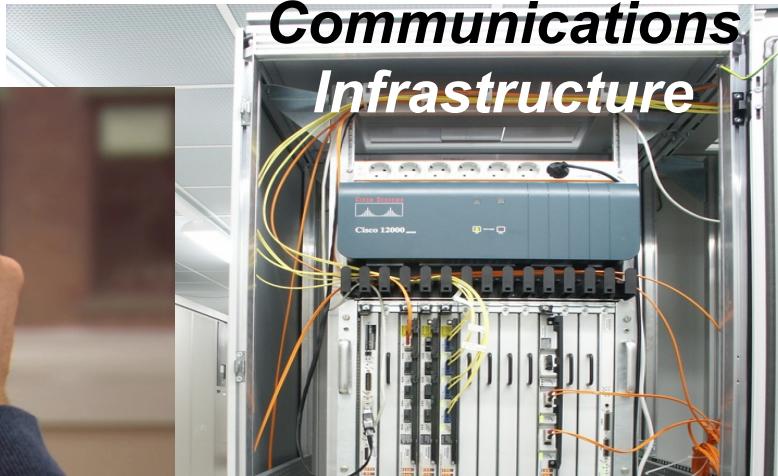
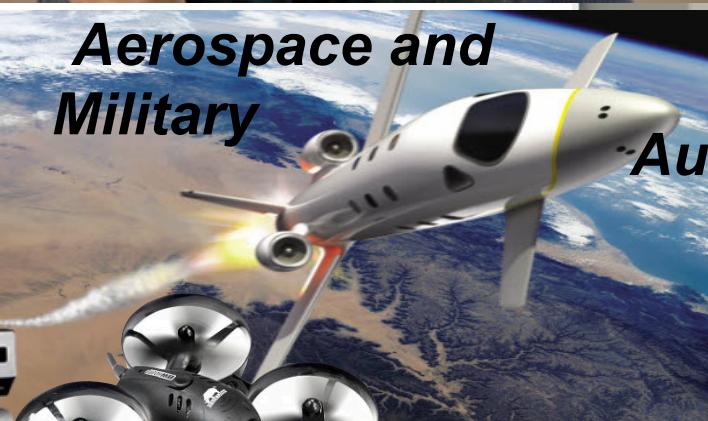
# *Electronics all around us*



**Consumer  
Products**

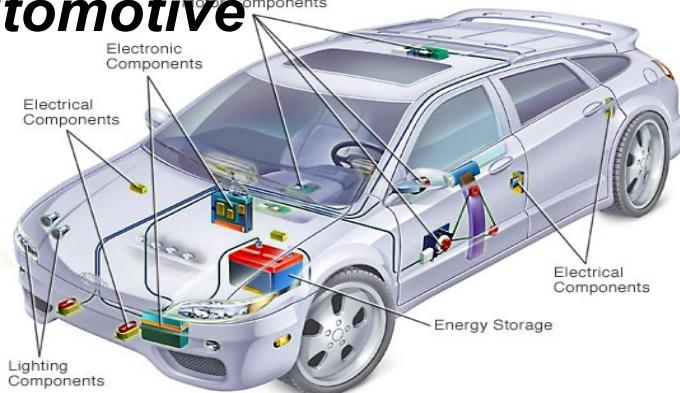


**Aerospace and  
Military**

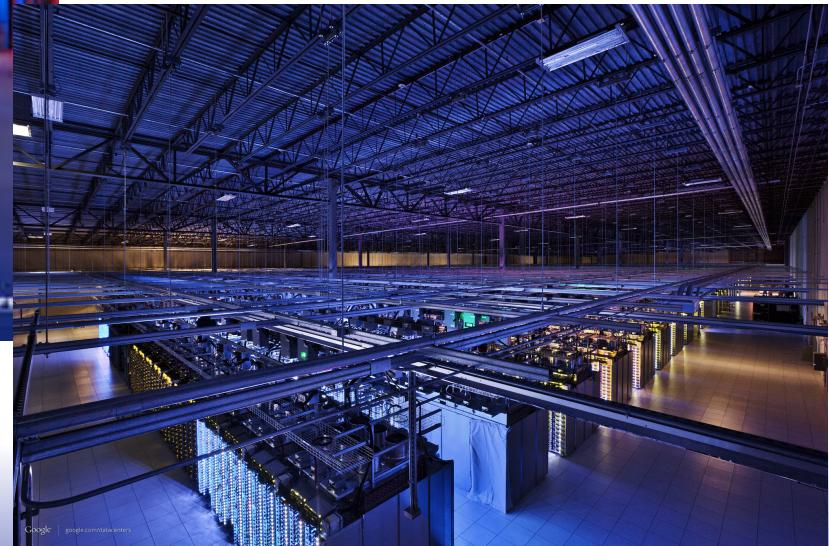


**Communications  
Infrastructure**

**Automotive**

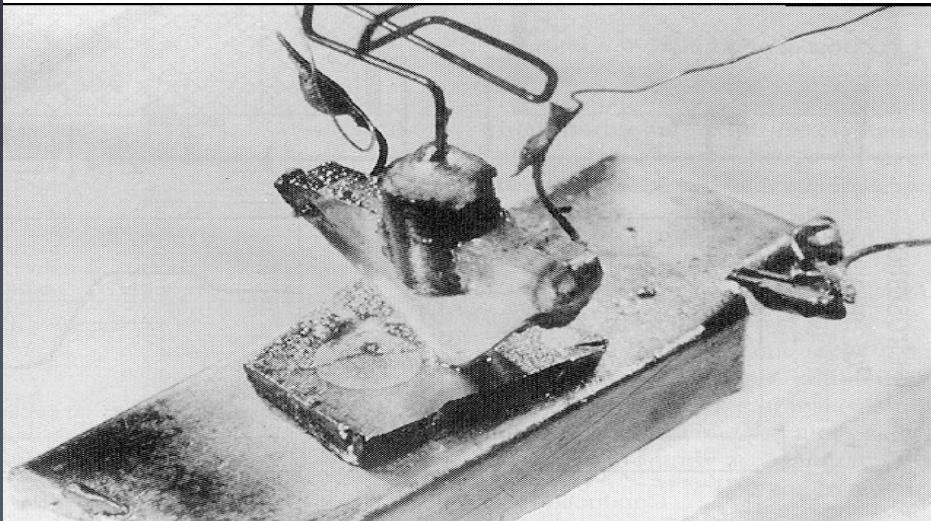


*And then plenty more ...*



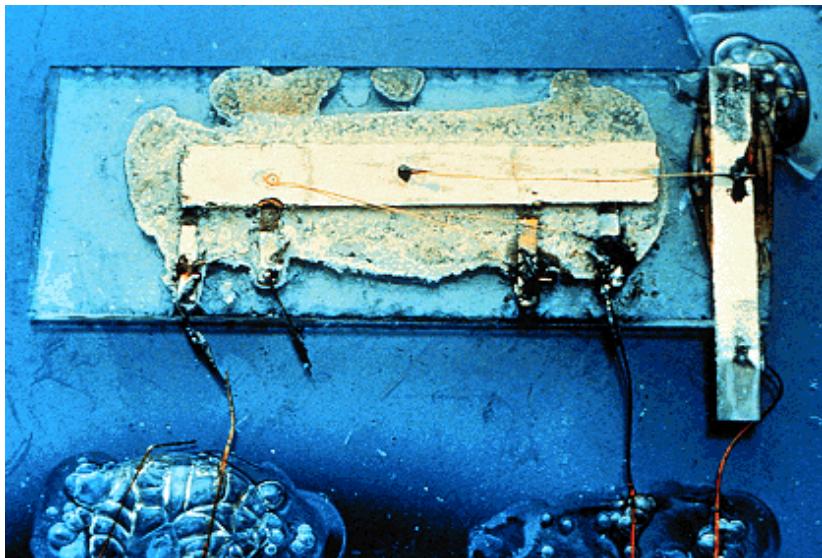
*How did this all arise?*

# *The Transistor Revolution*



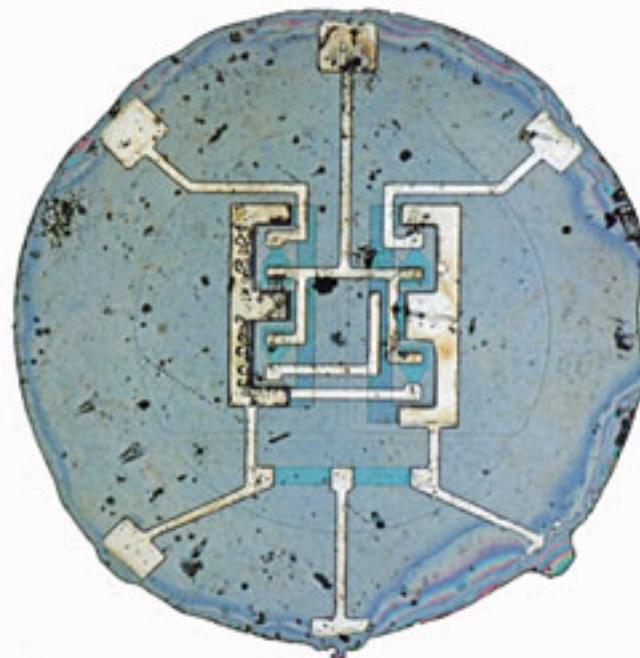
First transistor  
Bell Labs, Dec 1947

# *First Integrated Circuits (1958-59)*



*Jack Kilby, Texas Instruments*

*Bob Noyce, Fairchild*

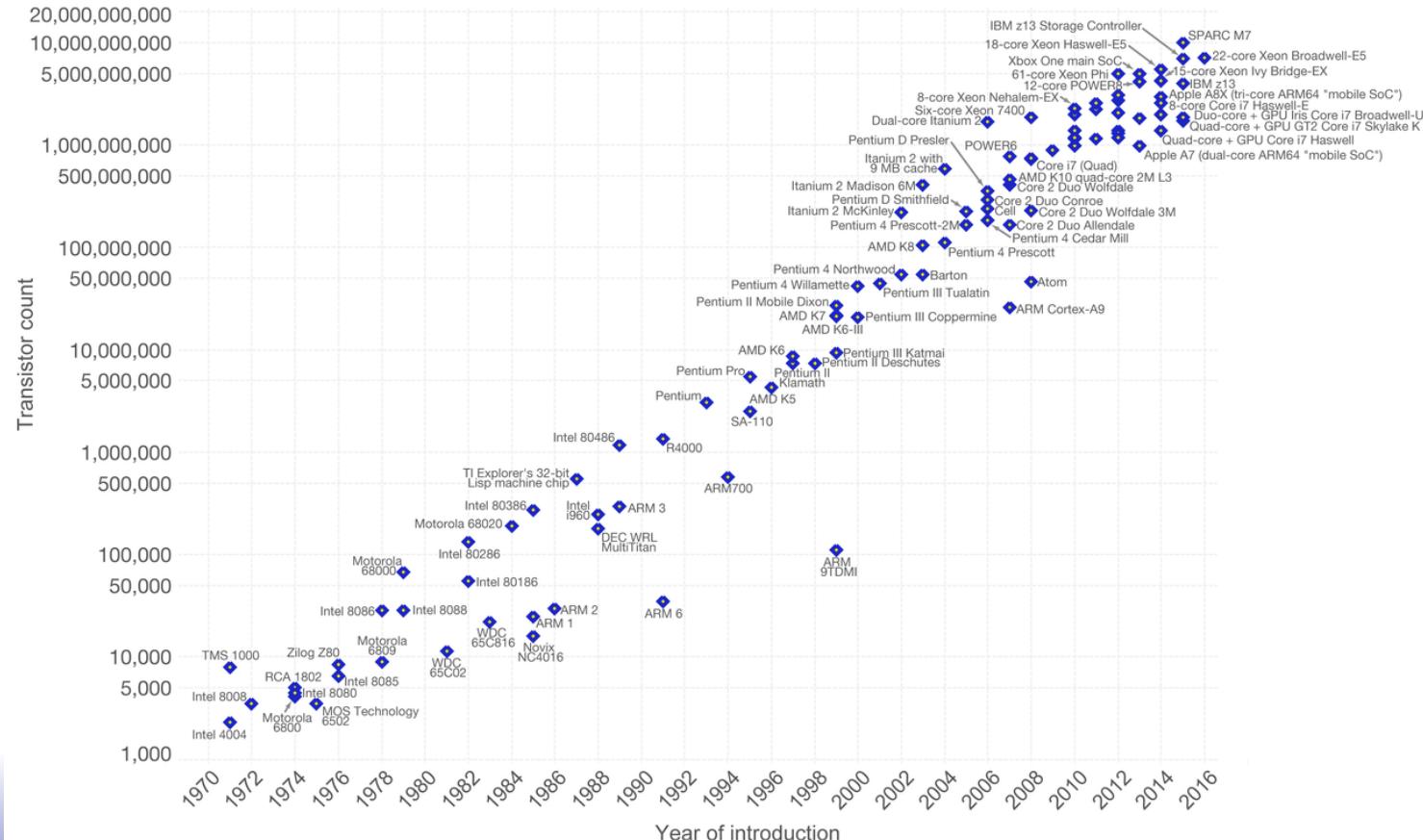


# Moore's Law – The number of transistors on integrated circuit chips (1971–2016)

OurWorld  
in Data

Moore's law describes the empirical regularity that the number of transistors on integrated circuits doubles approximately every two years.

This advancement is important as other aspects of technological progress – such as processing speed or the price of electronic products – are strongly linked to Moore's law.

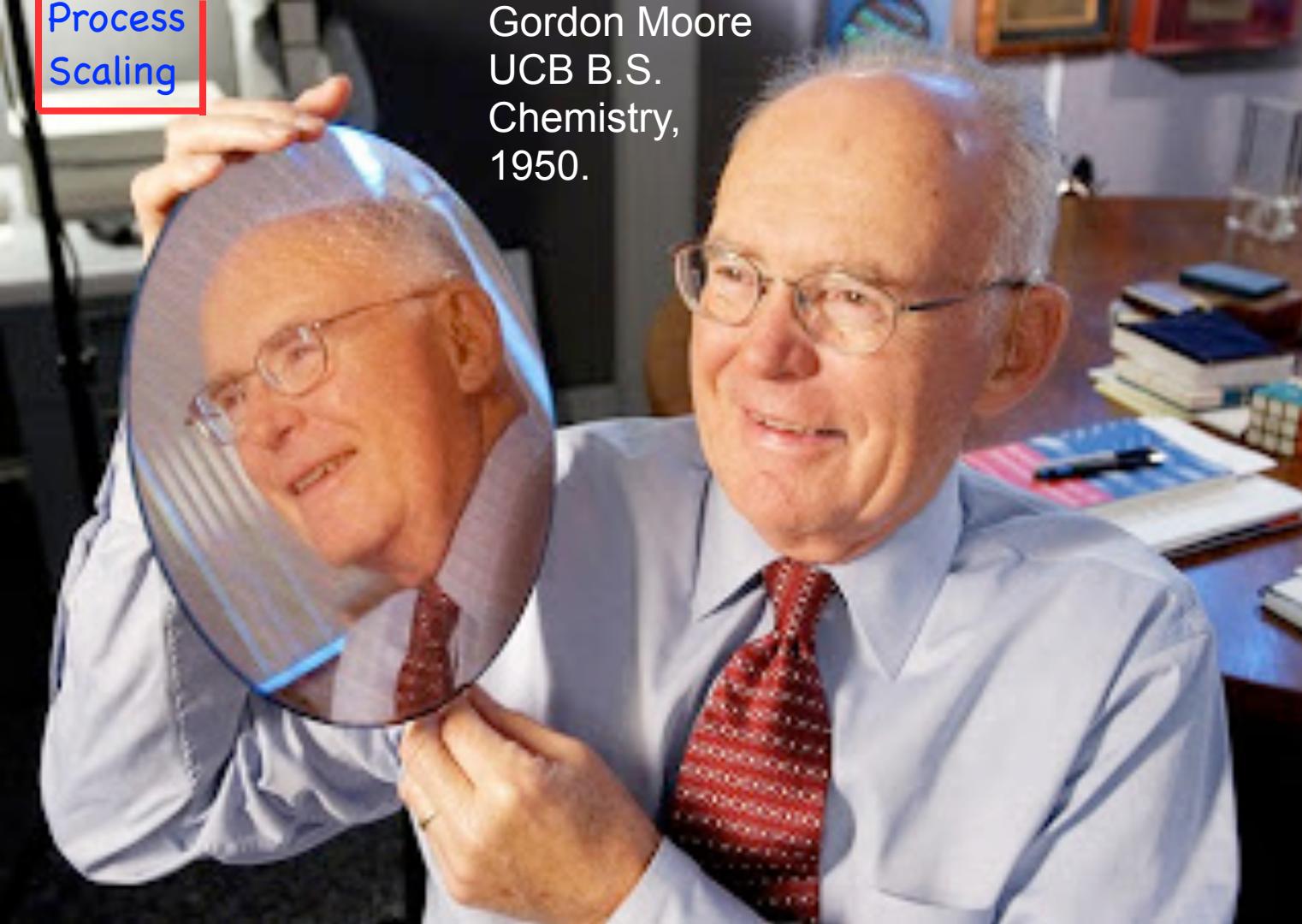


Data source: Wikipedia ([https://en.wikipedia.org/wiki/Transistor\\_count](https://en.wikipedia.org/wiki/Transistor_count))

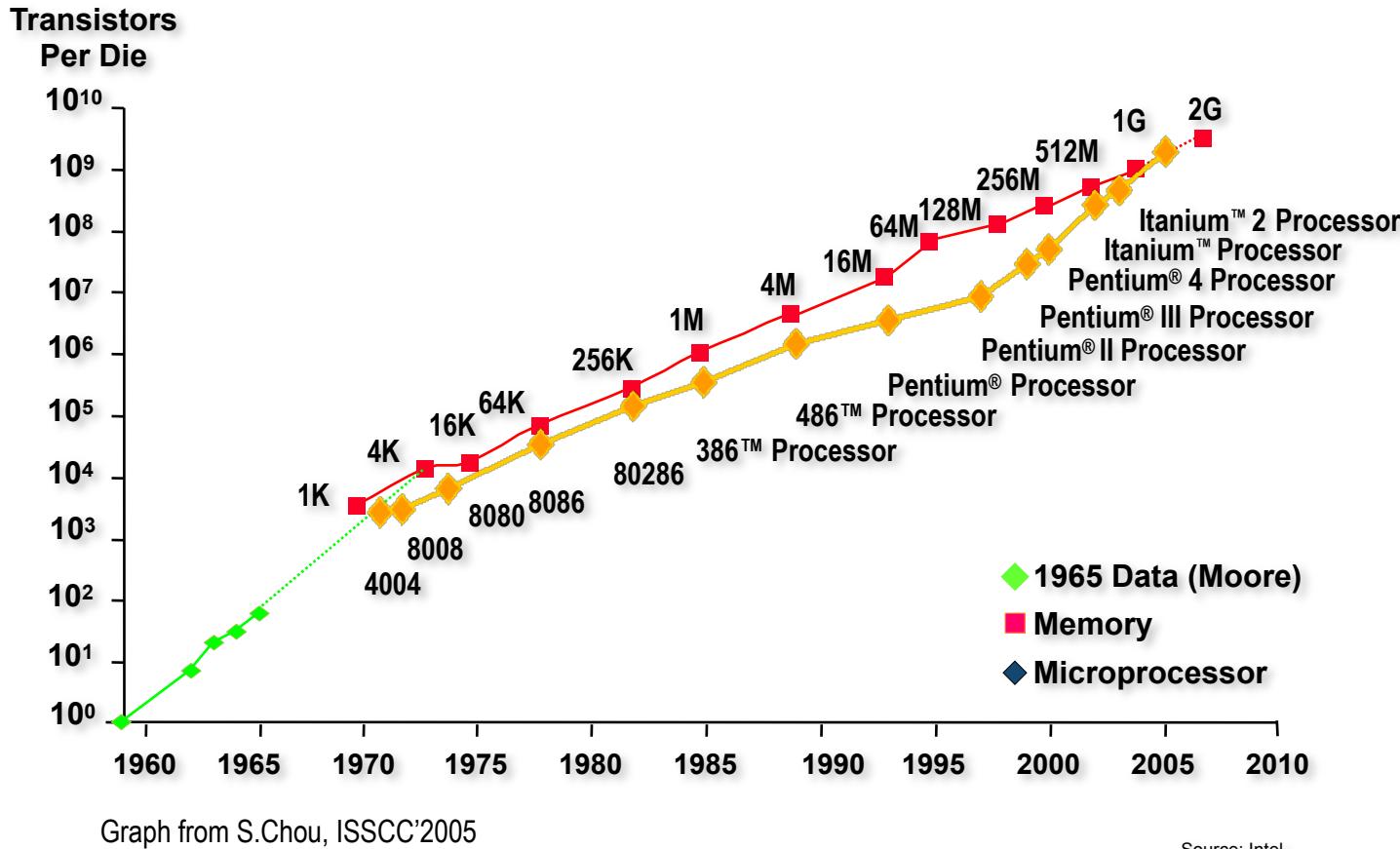
The data visualization is available at [OurWorldInData.org](http://OurWorldInData.org). There you find more visualizations and research on this topic.

Process  
Scaling

Gordon Moore  
UCB B.S.  
Chemistry,  
1950.



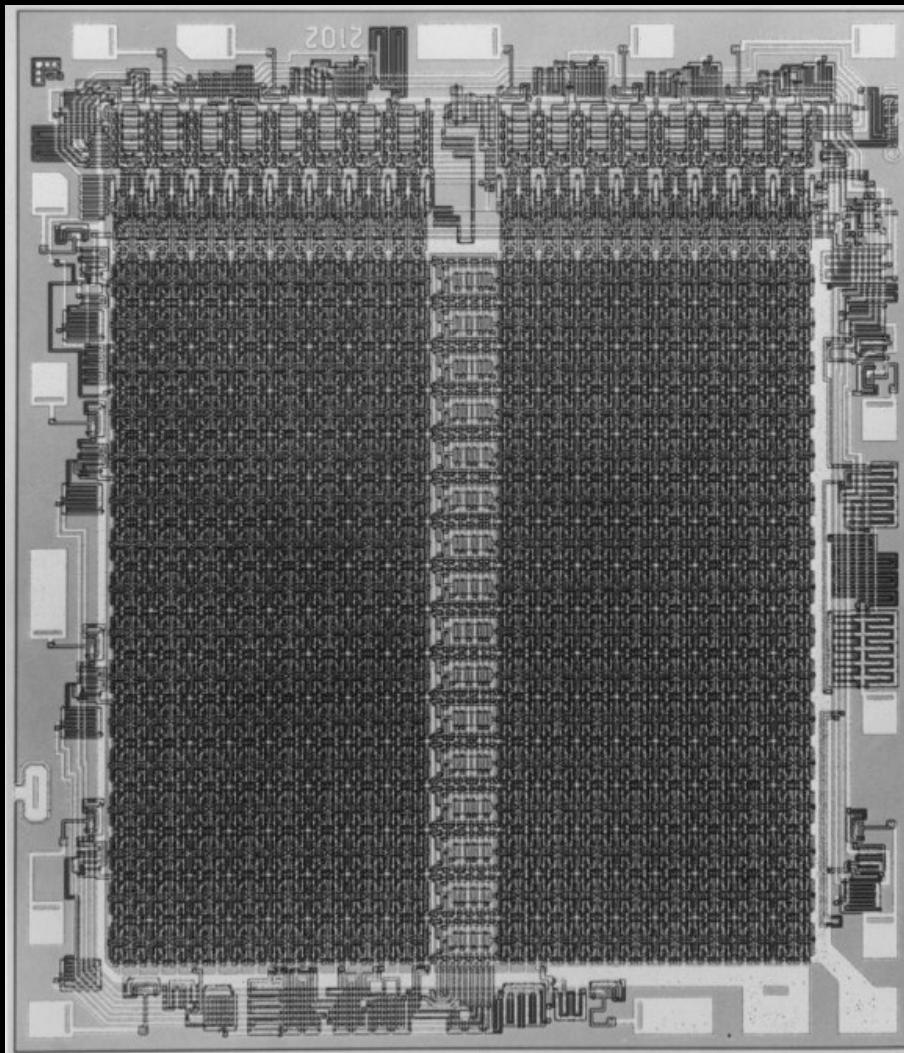
# Moore's Law - applied to memory and logic



## MOS in the 70s

1971 state of the art.

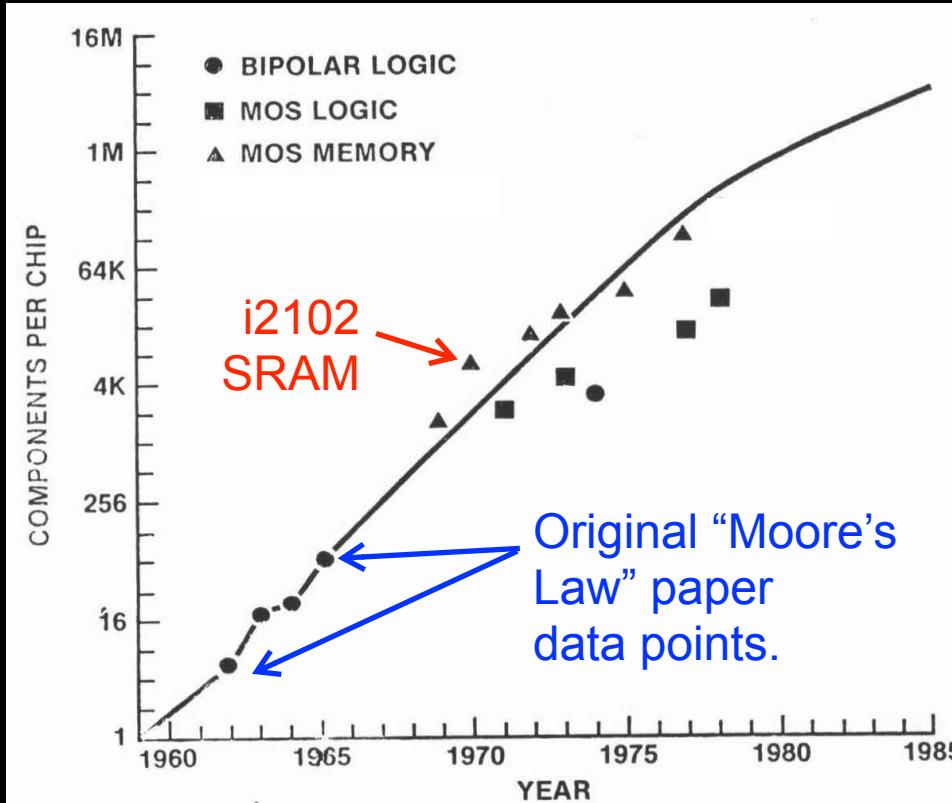
Intel 2102, a 1kb, 1 MHz static RAM chip with 6000 nFETs transistors in a 10 µm process.



## **By 1971, “Moore’s Law” paper was already 6 years old ...**

But the result was empirical.

Understanding the physics of scaling MOS transistor dimensions was necessary ...



# 1974: Dennard Scaling



IEEE JOURNAL OF SOLID-STATE CIRCUITS, VOL. SC-9, NO. 5, OCTOBER 1974

## Design of Ion-Implanted MOSFET's with Very Small Physical Dimensions

ROBERT H. DENNARD, MEMBER, IEEE, FRITZ H. GAENSSLER, HWA-NIEN YU, MEMBER, IEEE, V. LEO RIDEOUT, MEMBER, IEEE, ERNEST BASSOUS, AND ANDRE R. LEBLANC, MEMBER, IEEE

If we scale the gate length by a factor  $\kappa$ , how should we scale other aspects of transistor to get the “best” results?

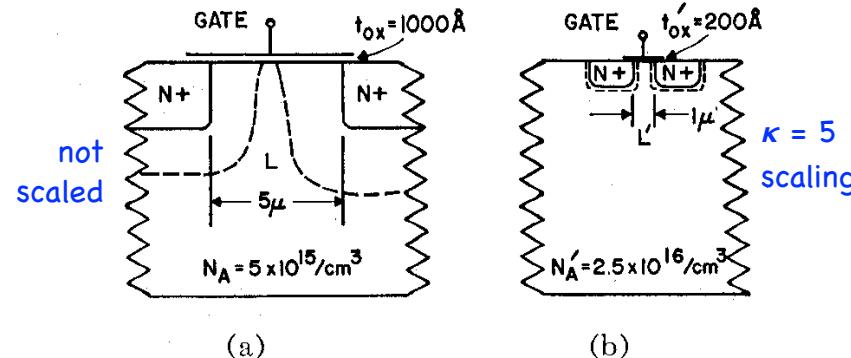
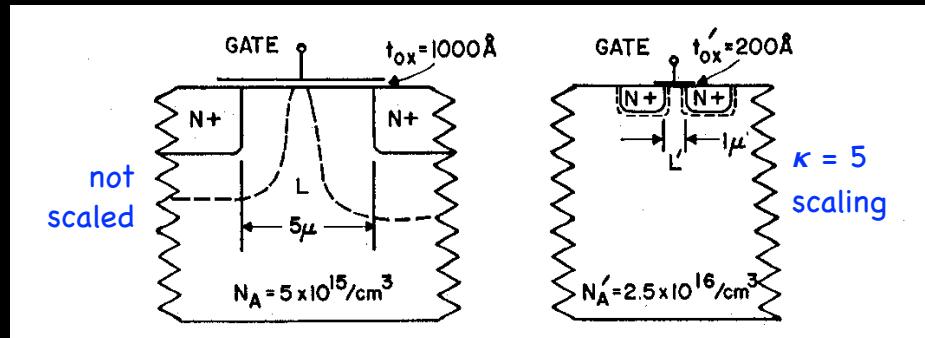


Fig. 1. Illustration of device scaling principles with  $\kappa = 5$ . (a) Conventional commercially available device structure. (b) Scaled-down device structure.

# Dennard Scaling

Things we do: scale dimensions, doping, Vdd.



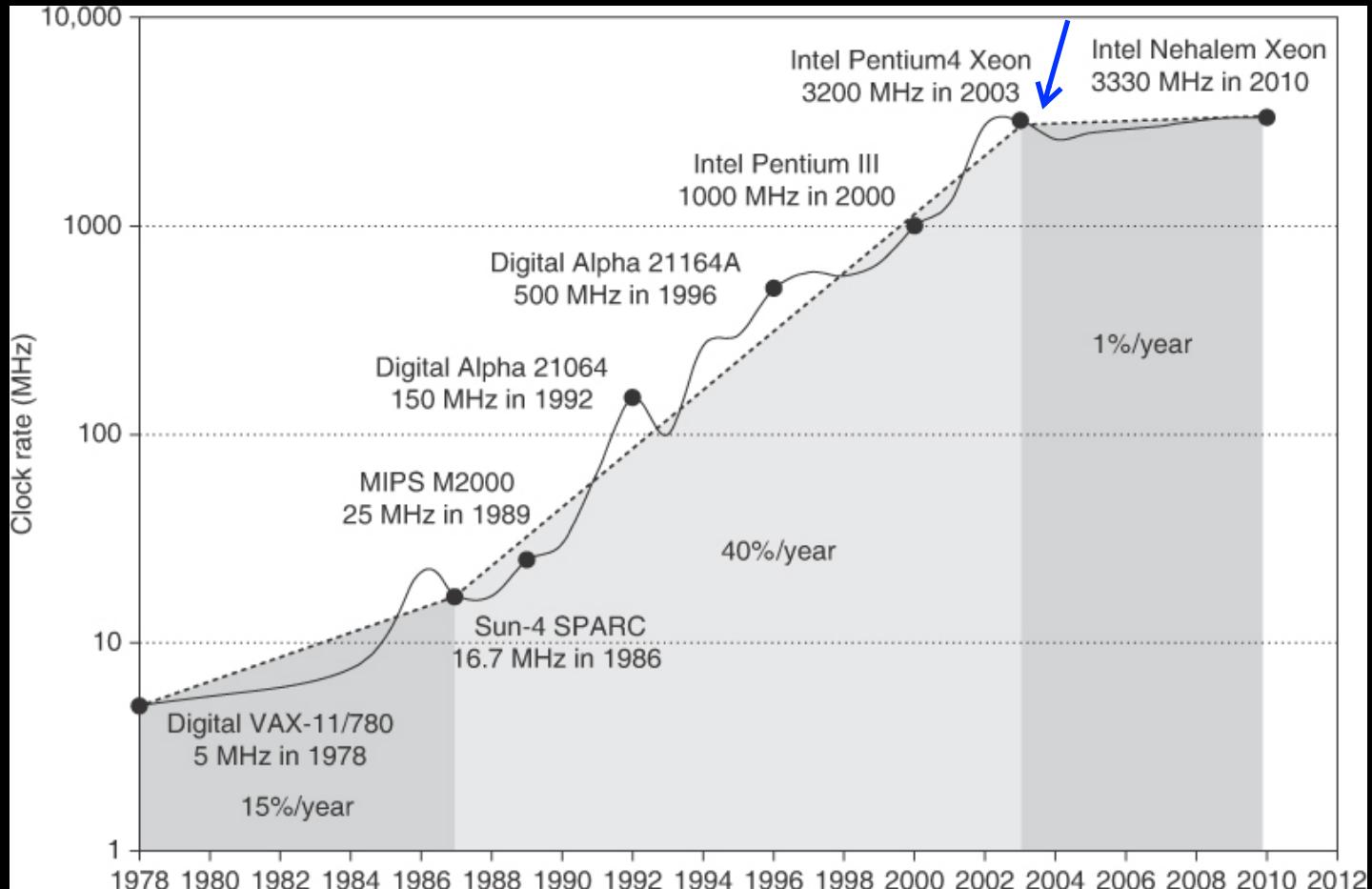
What we get:  
 $\kappa^2$  as many transistors  
at the same power density!

Whose gates switch  $\kappa$  times faster!

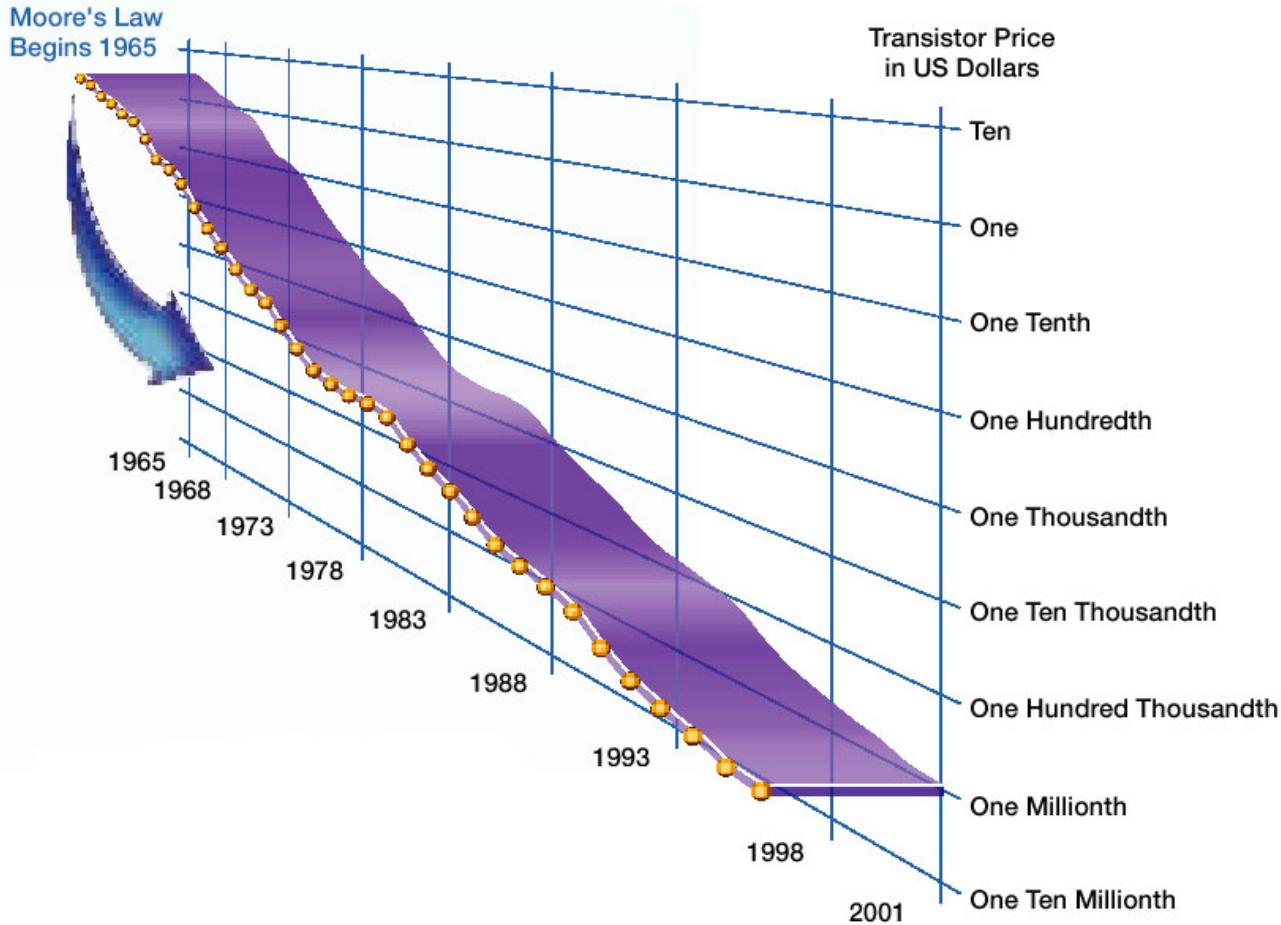
Device or Circuit Parameter	Scaling Factor
Device dimension $t_{ox}, L, W$	$1/\kappa$
Doping concentration $N_a$	$\kappa$
Voltage $V$	$1/\kappa$
Current $I$	$1/\kappa$
Capacitance $\epsilon A/t$	$1/\kappa$
Delay time/circuit $VC/I$	$1/\kappa$
Power dissipation/circuit $VI$	$1/\kappa^2$
Power density $VI/A$	1

Power density scaling ended in 2003  
(Pentium 4: 3.2GHz, 82W, 55M FETs).

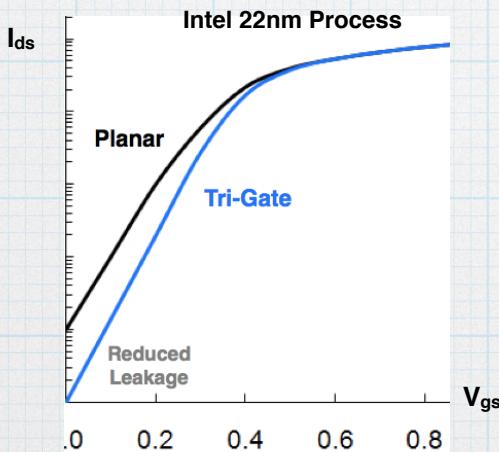
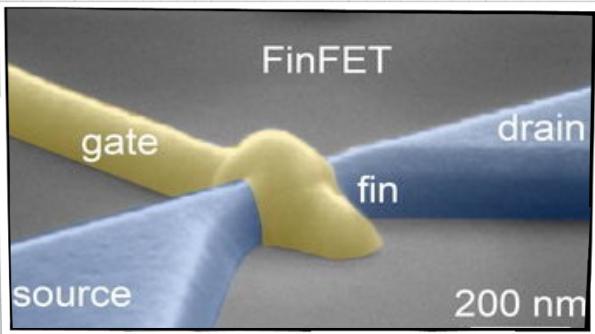
## Dennard Scaling ended ... when we hit the “power wall”



# The Key Benefit of Moore's Law Scaling: Cost



# Modern IC Process



Transistor channel is a raised fin.

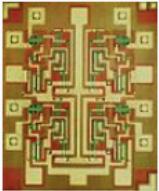
Gate controls channel from sides and top.

(12) United States Patent  
Hu et al. Filed: Oct. 23, 2000

(54) FINFET TRANSISTOR STRUCTURES HAVING A DOUBLE GATE CHANNEL EXTENDING VERTICALLY FROM A SUBSTRATE AND METHODS OF MANUFACTURE

(75) Inventors: Chenming Hu, Alamo; Tsu-Jae King, Fremont; Vivek Subramanian, Redwood City; Leland Chang, Berkeley; Xuejue Huang, Yang-Kyu Choi, both of Albany; Jakub Tadeusz Kedzierski, Hayward; Nick Lindert, Berkeley; Jeffrey Bokor, Oakland, all of CA (US); Wen-Chin Lee, Beaverton, OR (US)

## Semiconductor device fabrication



### MOSFET scaling (process nodes)

10  $\mu\text{m}$  – 1971

6  $\mu\text{m}$  – 1974

3  $\mu\text{m}$  – 1977

1.5  $\mu\text{m}$  – 1981

1  $\mu\text{m}$  – 1984

800 nm – 1987

600 nm – 1990

350 nm – 1993

250 nm – 1996

180 nm – 1999

130 nm – 2001

90 nm – 2003

65 nm – 2005

45 nm – 2007

32 nm – 2009

22 nm – 2012

14 nm – 2014

10 nm – 2016

7 nm – 2018

5 nm – 2020

3 nm – 2022

Future

2 nm ~ 2024

# State of the Art

## ▶ 7nm

As of September 2018, mass production of 7 nm devices has begun. The first mainstream 7 nm mobile processor intended for mass market use, the [Apple A12 Bionic](#), was released at their September 2018 event. Although [Huawei](#) announced its own 7 nm processor before the Apple A12 Bionic, the Kirin 980 on August 31, 2018, the [Apple A12 Bionic](#) was released for public, mass market use to consumers before the Kirin 980. Both chips are manufactured by [TSMC](#). AMD is currently working on their "Rome" workstation processors, which are based on the 7 nanometer node and feature up to 64 cores.

## ▶ 5nm

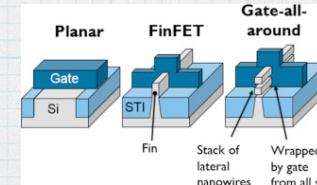
The 5 nm node was once assumed by some experts to be the end of [Moore's law](#).

Transistors smaller than 7 nm will experience [quantum tunnelling](#) through the gate oxide layer. Due to the costs involved in development, 5 nm is predicted to take longer to reach market than the two years estimated by Moore's law. Beyond 7 nm, it was initially claimed that major technological advances would have to be made to produce chips at this small scale. In particular, it is believed that 5 nm may usher in the successor to the [FinFET](#), such as a [gate-all-around](#) architecture.

Although Intel has not yet revealed any specific plans to manufacturers or retailers, their 2009 roadmap projected an end-user release by approximately 2020. In early 2017, [Samsung](#) announced production of a 4 nm node by 2020 as part of its revised roadmap. On January 26th 2018, [TSMC](#) announced production of a 5 nm node by 2020 on its new fab 18. In October 2018, TSMC disclosed plans to start risk production of 5 nm devices in April 2019.

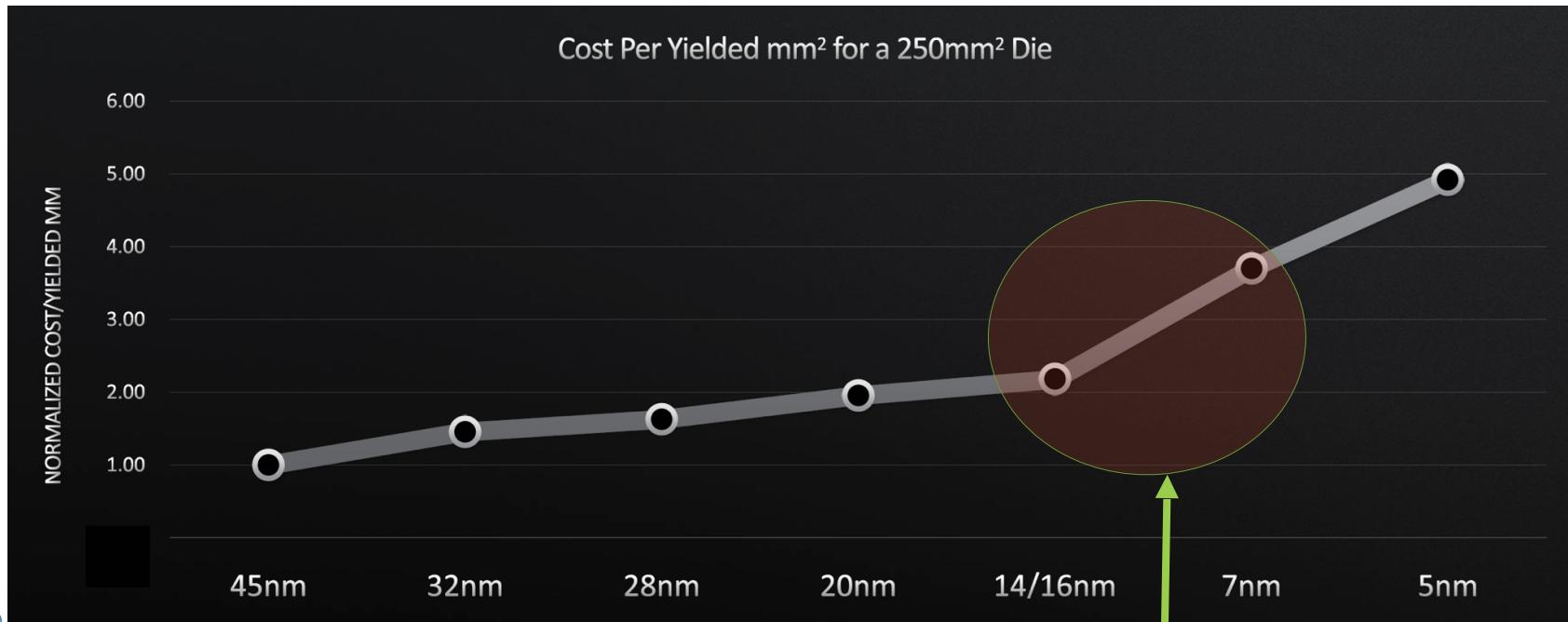
## ▶ 3.5nm

3.5 nm is a name for the first node beyond 5 nm. In 2018, [IMEC](#) and [Cadence](#) had taped out 3 nm test chips. Also, [Samsung](#) announced that they plan to use Gate-All-Around technology to produce 3 nm FETs in 2021.



\* From Wikipedia

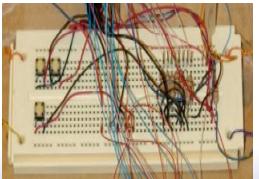
# Recent Cost Trend



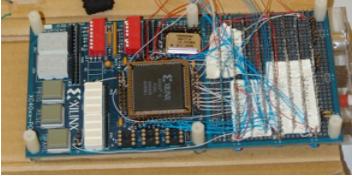
L. Su, HotChips, August 2019.

Cost nearly doubled!

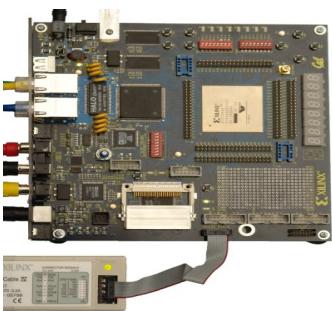
# CS150/EECS151 Project Complexity



1980 Pong game  
10's of logic gates



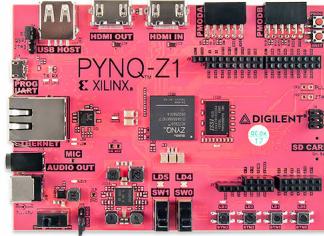
1995 MIDI synthesizer  
1000's of logic gates



2000-2010 eTV tuner  
10K's logic gates



2010-2017 MIPS CPU or BYO  
1M logic gates

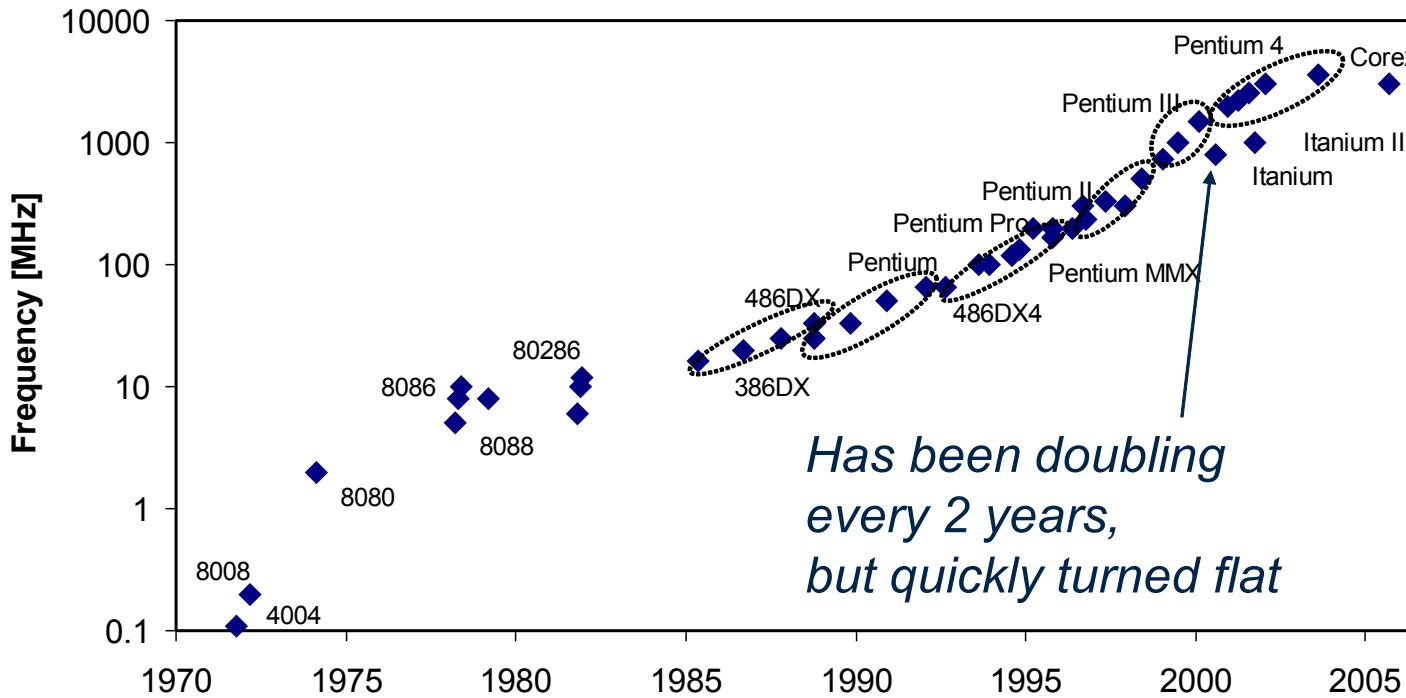


2018 MIPS CPU  
Programmable SOC:  
dual-core ARM, 85K  
logic cells, 220 MACC

# *The other outcomes*

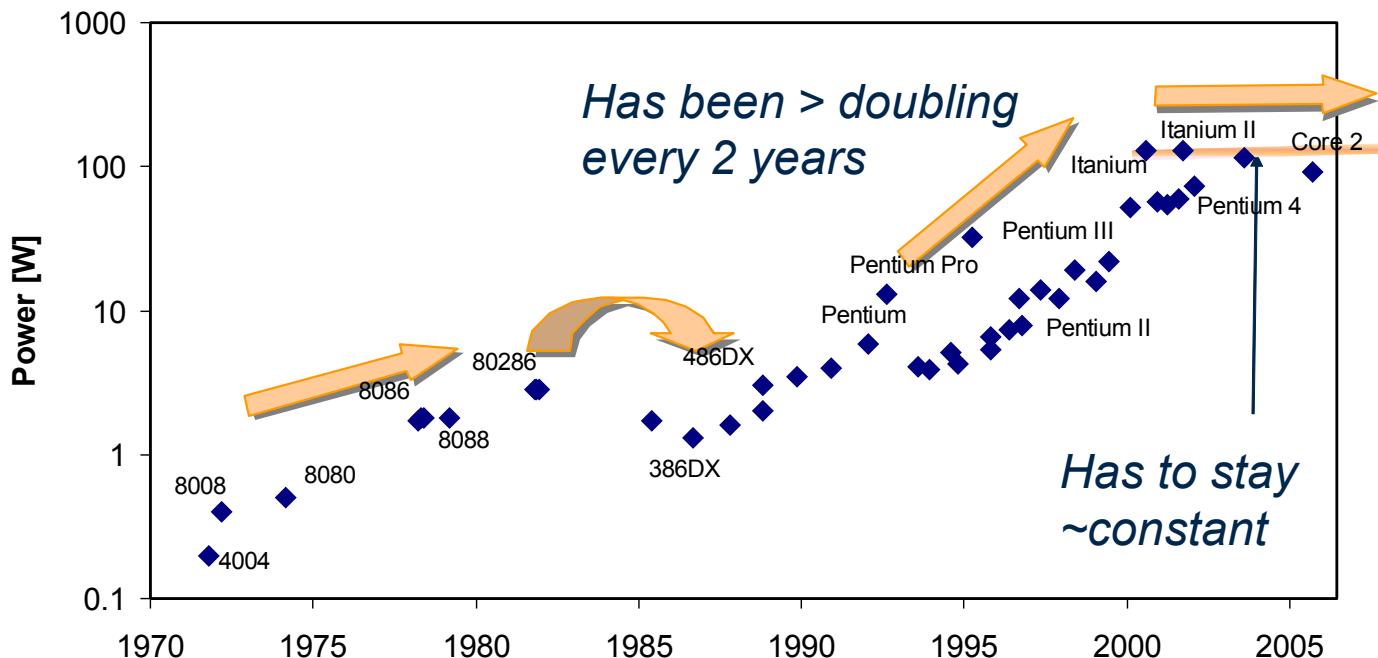
# Frequency

Frequency Trends in Intel's Microprocessors

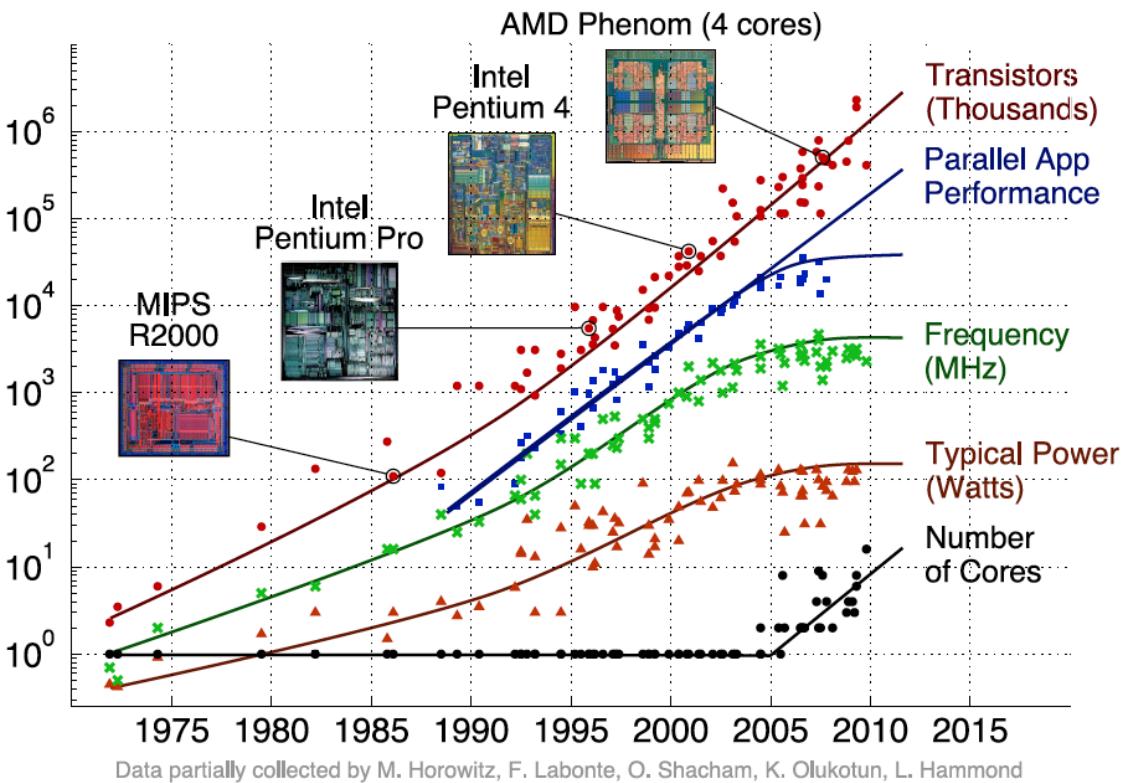


# Power Dissipation

Power Trends in Intel's Microprocessors



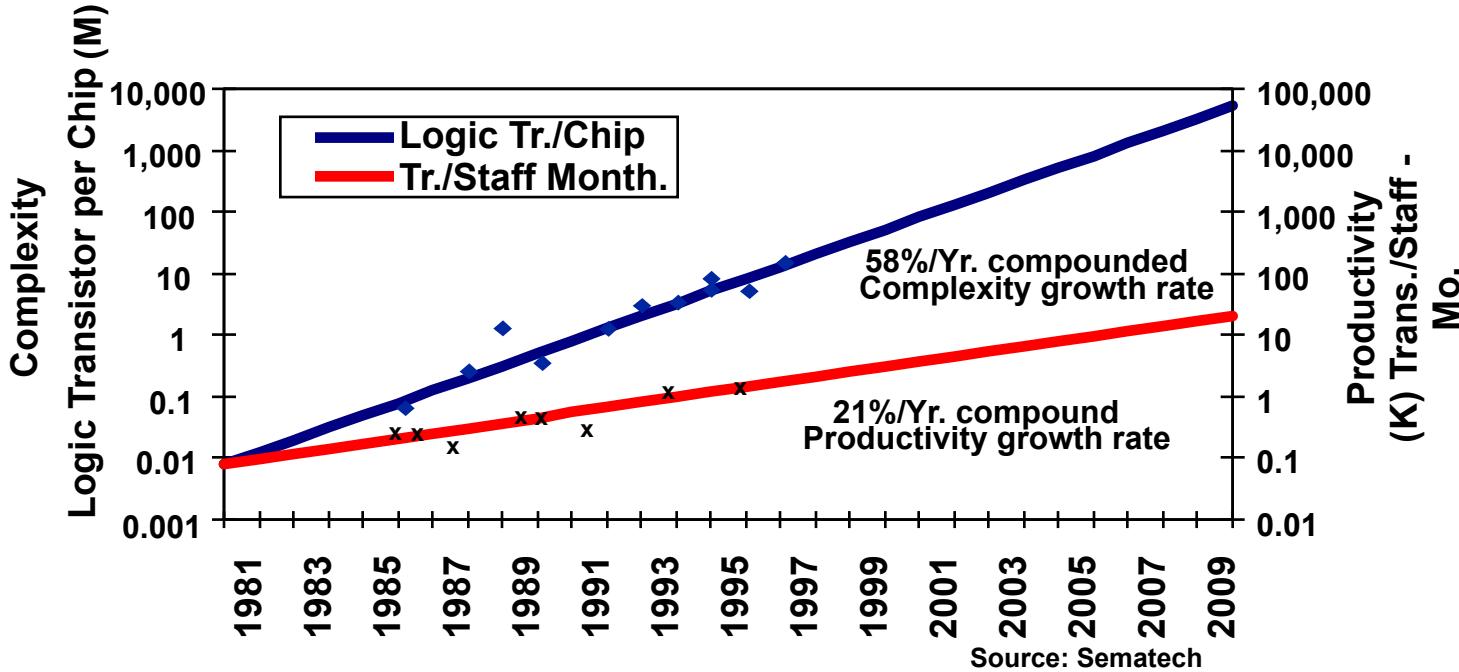
# Power and Performance Trends



*For reasons of power efficiency, performance scaling now comes from multiple cores and “accelerators”, not from higher clock frequency.*

# *The other Demon: Complexity*

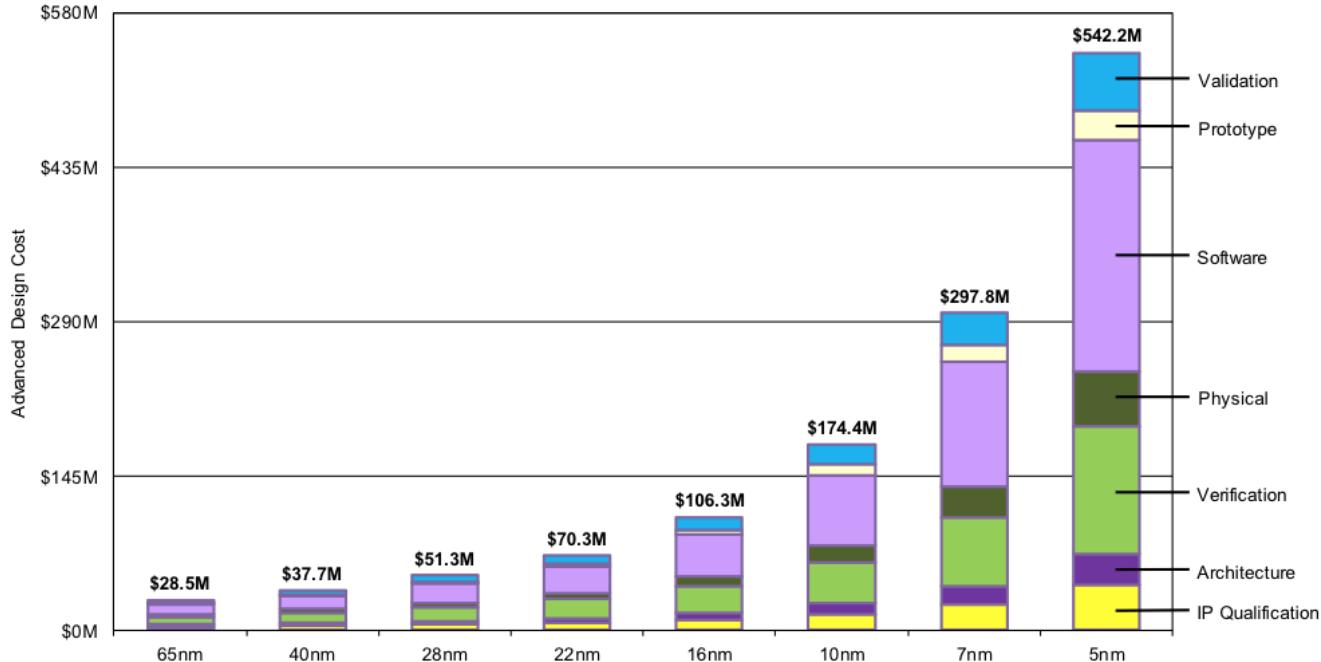
# Complexity and Productivity Trends



**Complexity outpaces design productivity**

Courtesy, ITRS Roadmap

# Cost Of Developing New Products



- These are non-recurring (NRE) costs, need to be amortized over the lifetime of a product

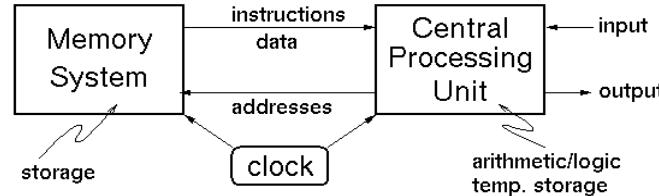
# *The answers*

- Design methodology!
  - Abstraction
  - Hierarchy
  - Reuse
- Computer Aided Design tools

# *Digital System Design: A few basic concepts*

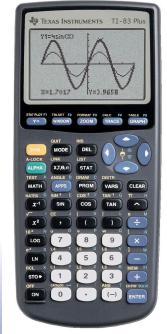
# Example Digital Systems

- General Purpose Server



- *Designed to maximize performance* -  
“Optimized for speed”.
- *Expensive and high power*

- Handheld Calculator



- *Usually designed to minimize cost.*  
“Optimized for low cost”
- *Of course, low cost comes at the expense of speed.*

# Example Digital Systems

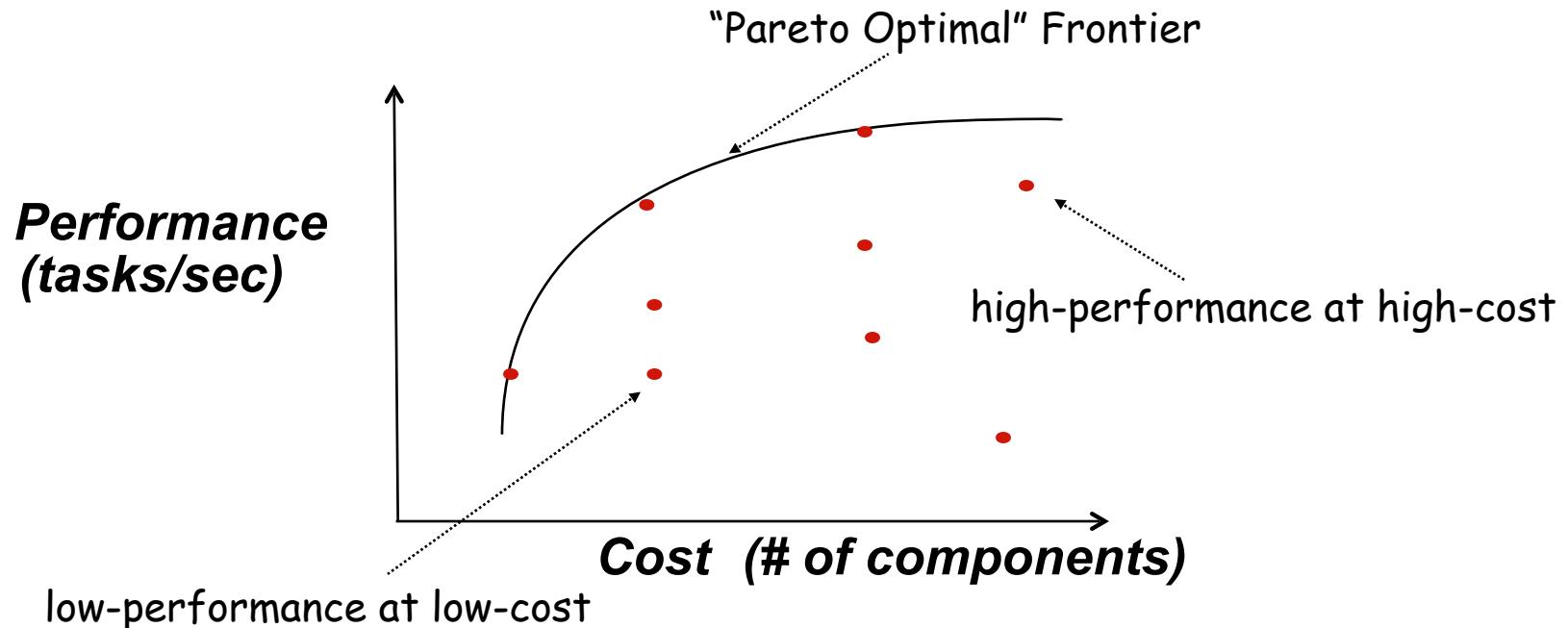
## □ Digital Watch



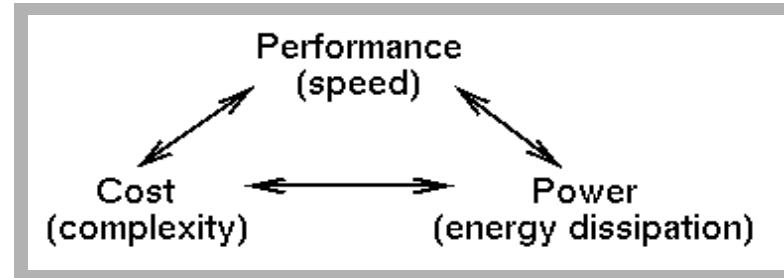
*Designed to minimize power.  
Single battery must last for years.*

- Low power operation comes at the expense of:
  - lower speed
  - higher cost

# *Design Space & Optimality*

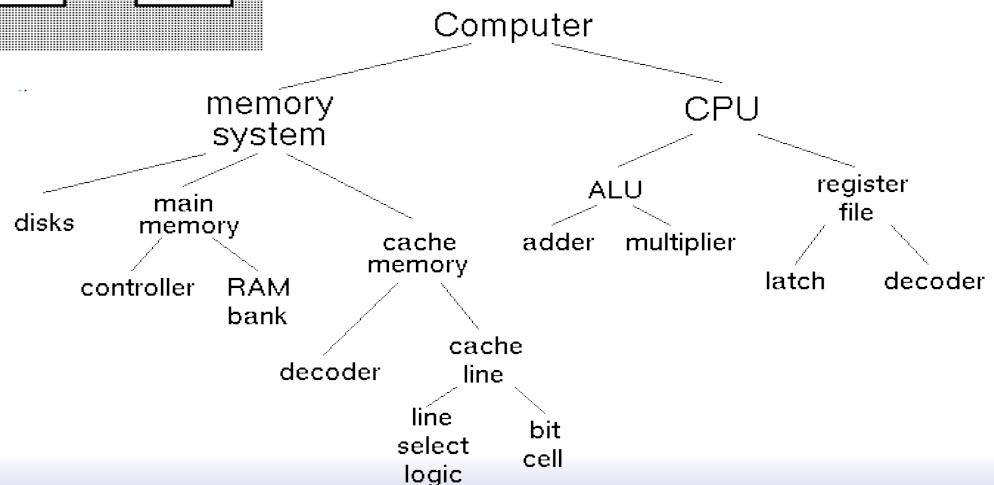
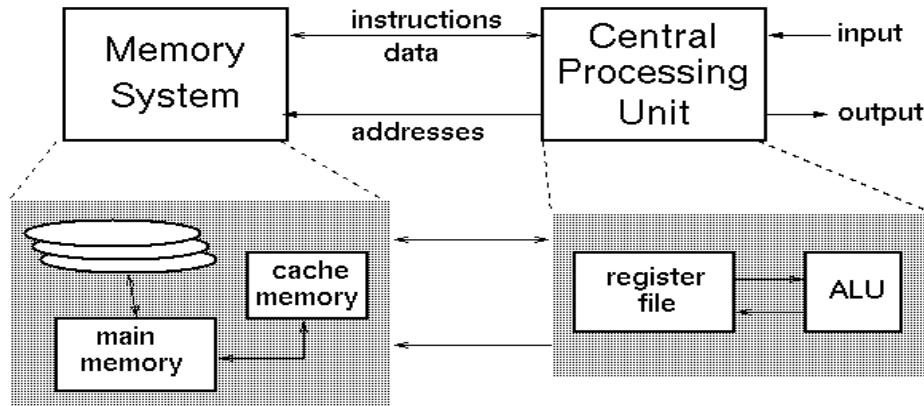


# Basic Design Tradeoffs



- Improve on one at the expense of the others
- Tradeoffs exist at every level in the system design
- Design Specification
  - Functional Description
  - Performance, cost, power constraints
- Designer must make the tradeoffs needed to achieve the function within the constraints

# Hierarchy & Design Representation



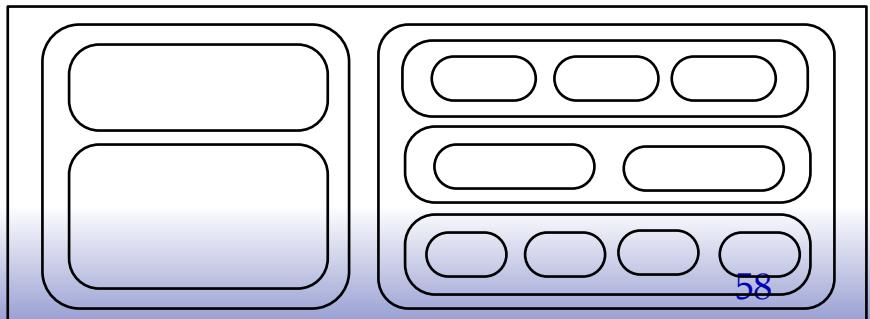
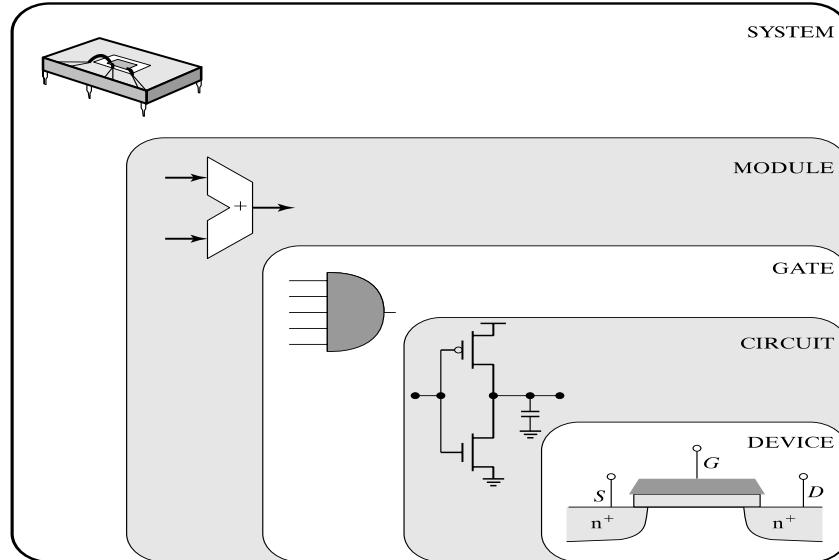
# Hierarchy in Designs – Complexity Control

## ❑ Design Abstraction

- Hide details and reduce number of things to handle at any time

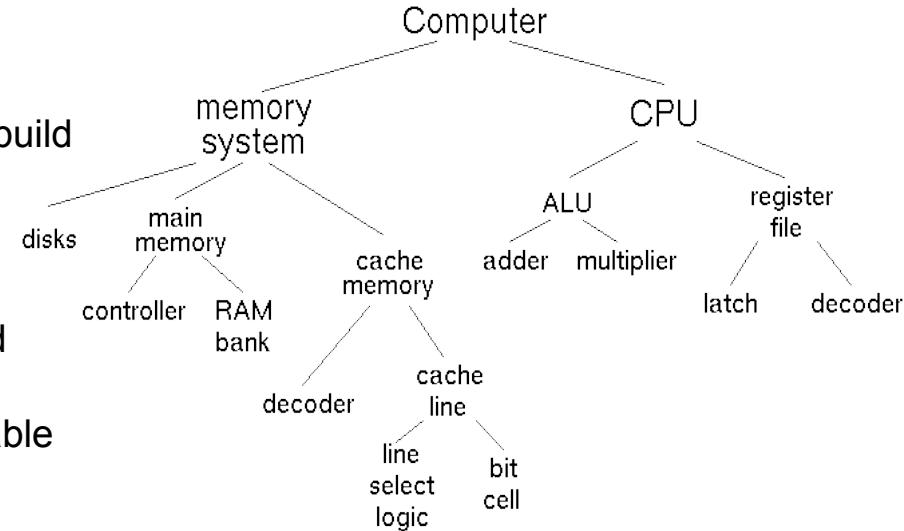
## ❑ Modular design

- Divide and conquer
- Simplifies implementation and debugging



# *Design Methodologies*

- ❑ Top-Down Design
  - Starts at the top (root) and works down by successive refinement.
- ❑ Bottom-up Design
  - Starts at the leaves & puts pieces together to build up the design.
- ❑ Which is better?
  - In practice both are needed & used
  - Top-down to handle the complexity (divide and conquer)
  - Bottom-up since structure influenced by available primitives  
(in a well designed system)



# Digital Design: What's it all about?

Given a functional description and performance, cost, & power constraints, come up with an implementation using a set of primitives.

- How do we learn how to do this?
  1. Learn about the primitives and how to use them.
  2. Learn about design representations.
  3. Learn formal methods and tools to manipulate the representations.
  4. Look at design examples.
  5. Use trial and error - CAD tools and prototyping. Practice!
- Digital design is in some ways more an art than a science. The creative spirit is critical in combining primitive elements & other components in new ways to achieve a desired function.
- However, unlike art, we have objective measures of a design:

*Performance Cost Power*

# *End of Lecture 1*