EITF20: Computer Architecture

Part1.1.1: Introduction

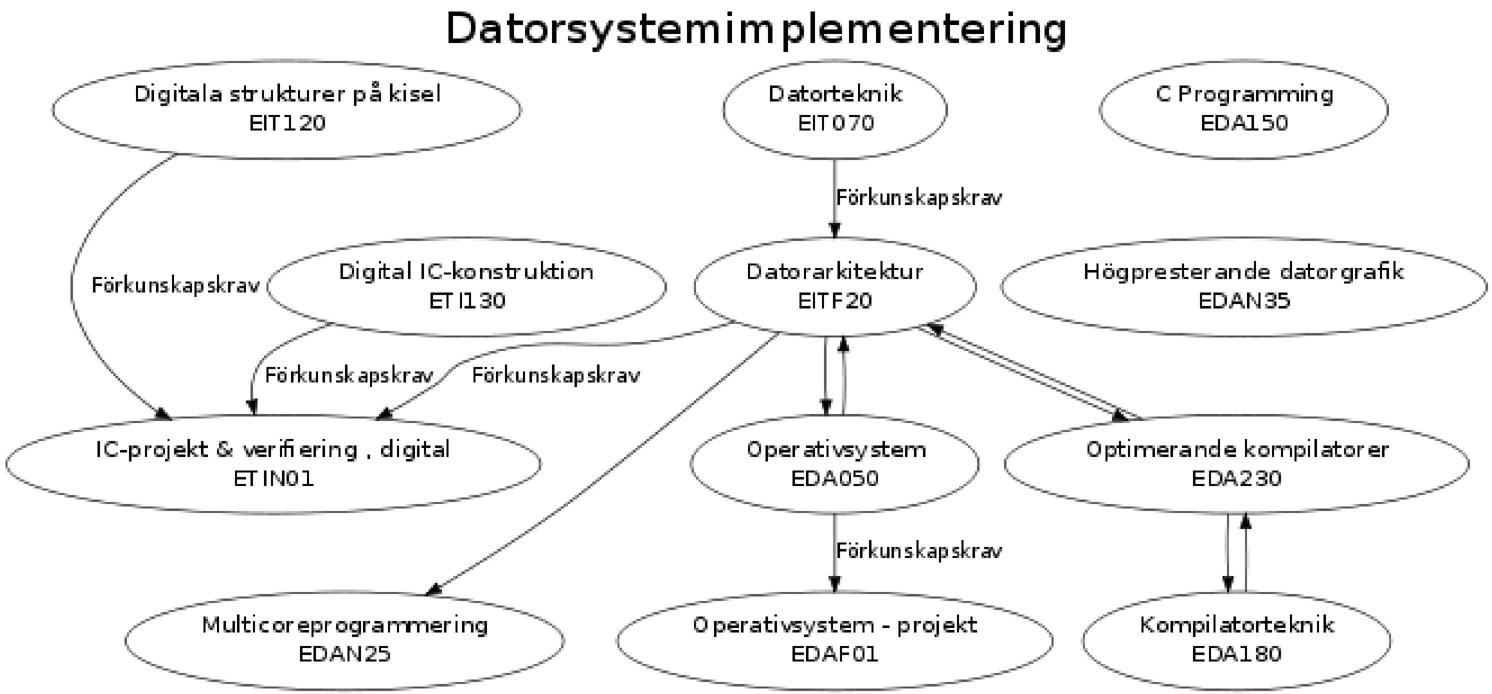
Liang Liu
liang.liu@eit.lth.se

Course Factor

- **Computer Architecture (7.5HP)**

http://www.eit.lth.se/kurs/eitf20

- **EIT's Course Service Desk (studerandeexpedition)**
  - Course secretary: Anne Andersson, Room 3152B
  - e-mail: anne.andersson@eit.lth.se



Outline

- Computers
- Computer Architecture
- This Course
- Trends
- Performance
- Quantitative Principles

Computer is everywhere

Build a Computer…

Desktop computer

Part	Price (SEK)	Example (2012-10-02)	Price
Case			490
Power supply			399
Motherboard			790
CPU			2490
Memory			698
Disk		SD	790
DVD/Blue-ray		)	590
Graphics			-
Sound, net, . . .			-
Keyboard, mouse, cables, .			?

$\sum = 3000$ - 5000 - 10000 SEK 6247

Power Consumption: 65 to 250 watts

Build a Computer…

3965 I lager för leverans inom 1 arbetsdagar

Tidigare 347,75 kr

298,19 kr

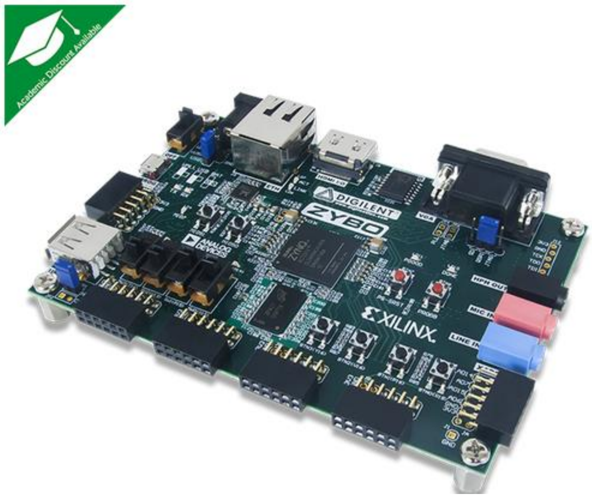
Pris (ex. moms) Each

- A 1.2GHz 64-bit quad-core ARMv8 CPU
- 802.11n Wireless LAN
- Bluetooth 4.1
- Bluetooth Low Energy (BLE)

Like the Pi 2, it also has:

- 1GB RAM
- 4 USB ports
- 40 GPIO pins
- Full HDMI port
- Ethernet port
- Combined 3.5mm audio jack and composite video
- Camera interface (CSI)
- Display interface (DSI)
- Micro SD card slot (now push-pull rather than push-push)
- VideoCore IV 3D graphics core

Build a Computer…



Digilent FPGA

Zybo Zynq-7000 ARM/FPGA SoC Trainer Board

$189.00

SKU:
410-279

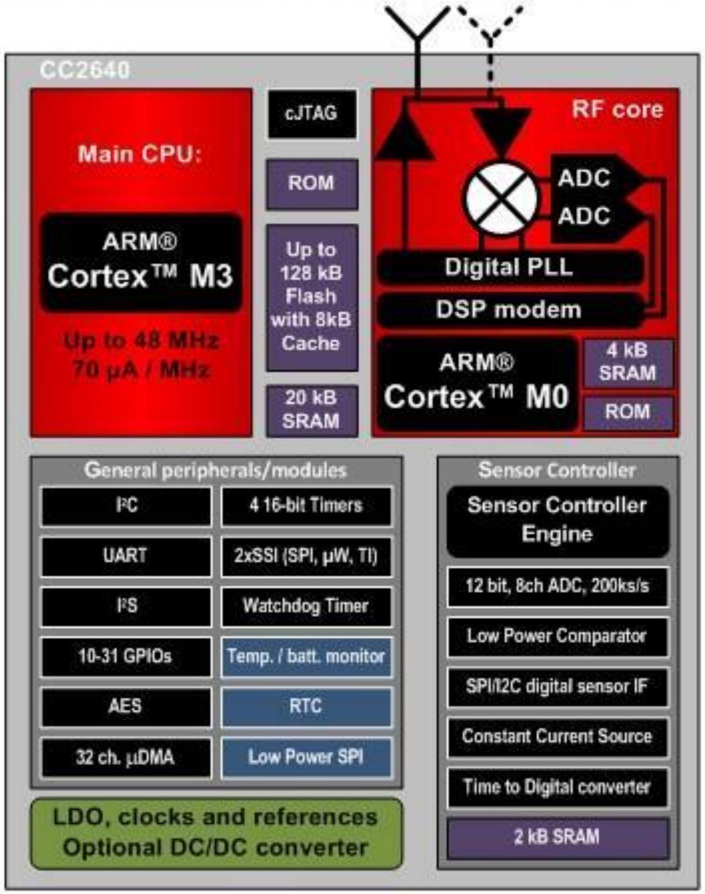
Quantity:
1

ADD TO CART

WISHLIST

Build a Computer…

CC2640 *Bluetooth low energy*



Quick Facts

Ultra-low Power Consumption
- 65 µA/MHz ARM Cortex M3
- 8.2 µA/MHz Sensor Controller
- 0.7 µA sleep with retention and RTC
- 5.9 mA RX (single-ended)
- 6.5 mA TX (single-ended)

SoC Key Features
- Autonomous sensor controller engine
- 4x4 mm to 7x7 mm QFN
- 1.65 – 3.8 V supply range
- 128 kB Flash + 8 kB Cache
- 20 kB RAM

RF Key Features
- +5 dBm output power
- -97 dBm sensitivity
- 2360 MHz – 2500 MHz
- Pin compatible with CC13xx in 4x4 and 5x5 QFN (BLE + Sub 1GHz prop)

$ 2.98

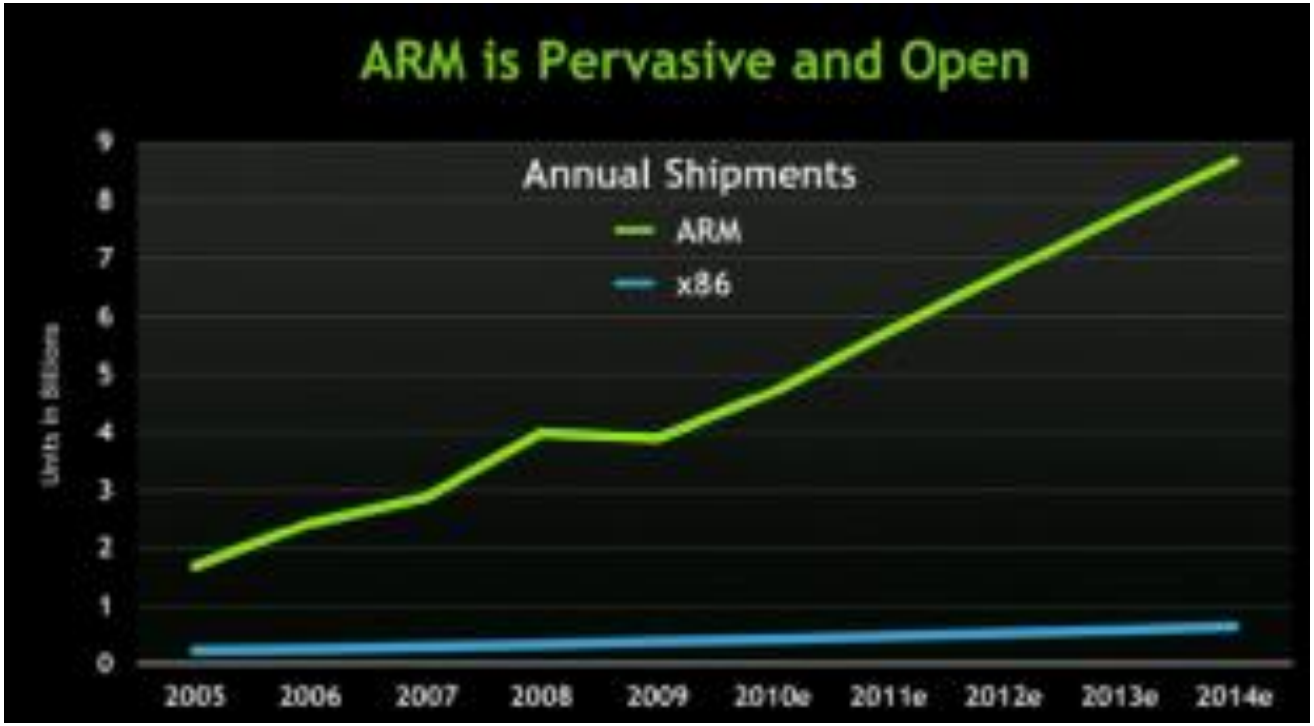
Build a Computer…

80 Megawatts

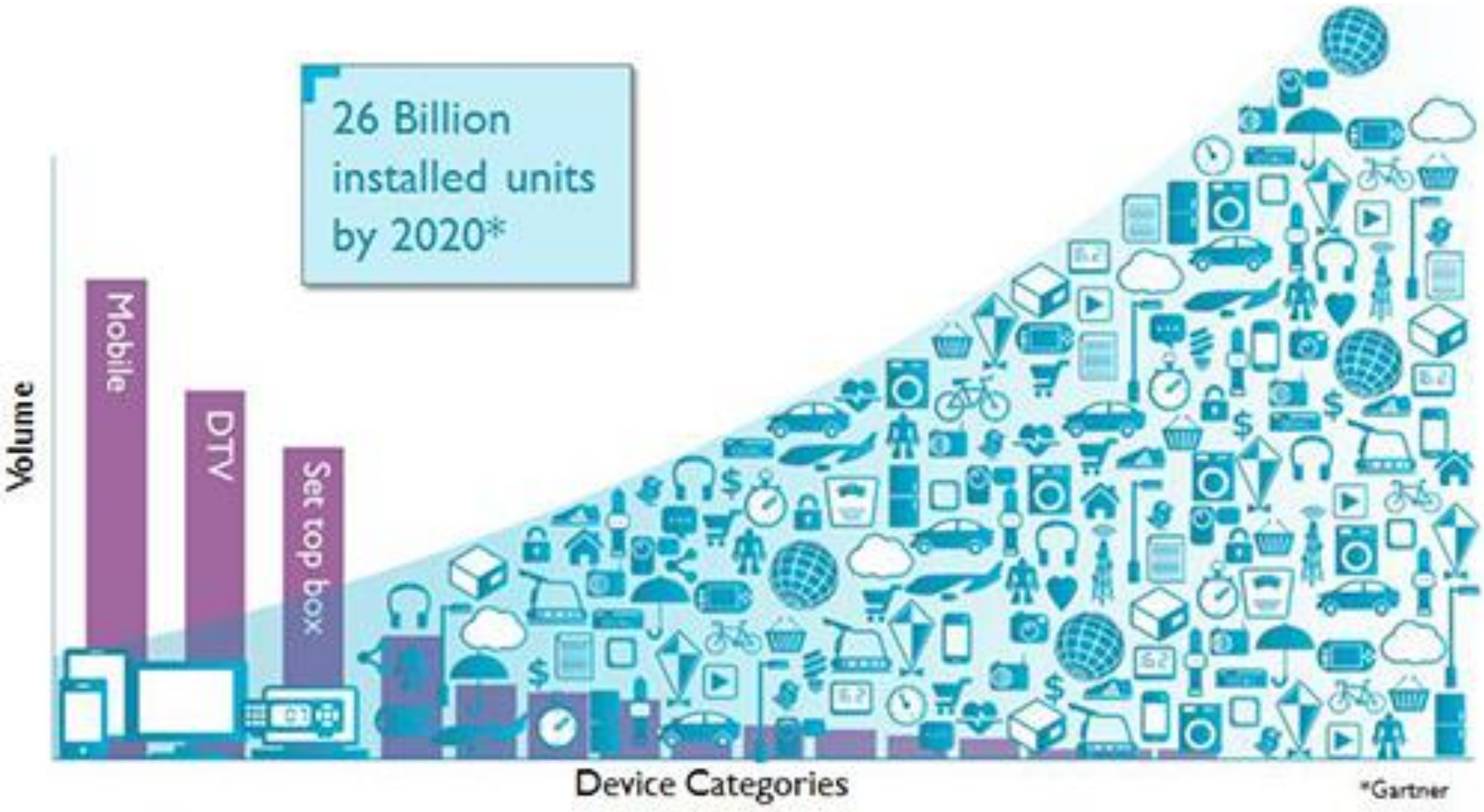
Class of Computers

Feature	Personal mobile device (PMD)	Desktop	Server	Clusters/warehouse-scale computer	Embedded
Price of system	\$100–\$1000	\$300–\$2500	\$5000–\$10,000,000	\$100,000–\$200,000,000	\$10–\$100,000
Price of micro-processor	\$10–\$100	\$50–\$500	\$200–\$2000	\$50–\$250	\$0.01–\$100
Critical system design issues	Cost, energy, media performance, responsiveness	Price-performance, energy, graphics performance	Throughput, availability, scalability, energy	Price-performance, throughput, energy proportionality	Price, energy, application-specific performance

Intel v.s. ARM

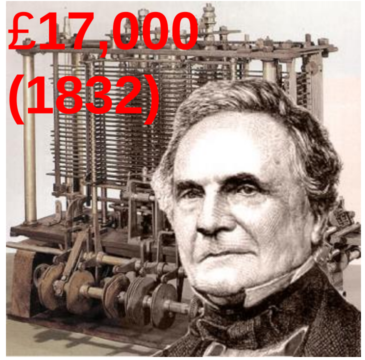

ARM is Pervasive and Open
Annual Shipments
ARM
x86
9
8
7
6
5
4
3
2
1
0
2005
2006
2007
2008
2009
2010e
2011e
2012e
2013e
2014e

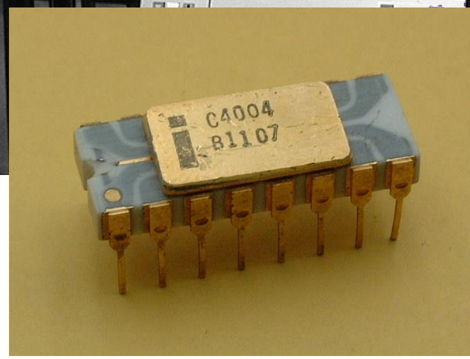
IoT - ARM


26 Billion
installed units
by 2020*
Volume
Mobile
DTV
Set top box
Device Categories
*Gartner

Time-line

- **Mid-1800 Programmable computer**
  - Charles Babbage (analytical engine)
  - Ada Lovelace (programmer)
- **1940s First modern computers**
  - Zuse, MARK, ENIAC, ...
- **1960s Mainframe**
  - 1964 IBM System/360
- **1970s Minicomputer**
  - 1971 First microprocessor

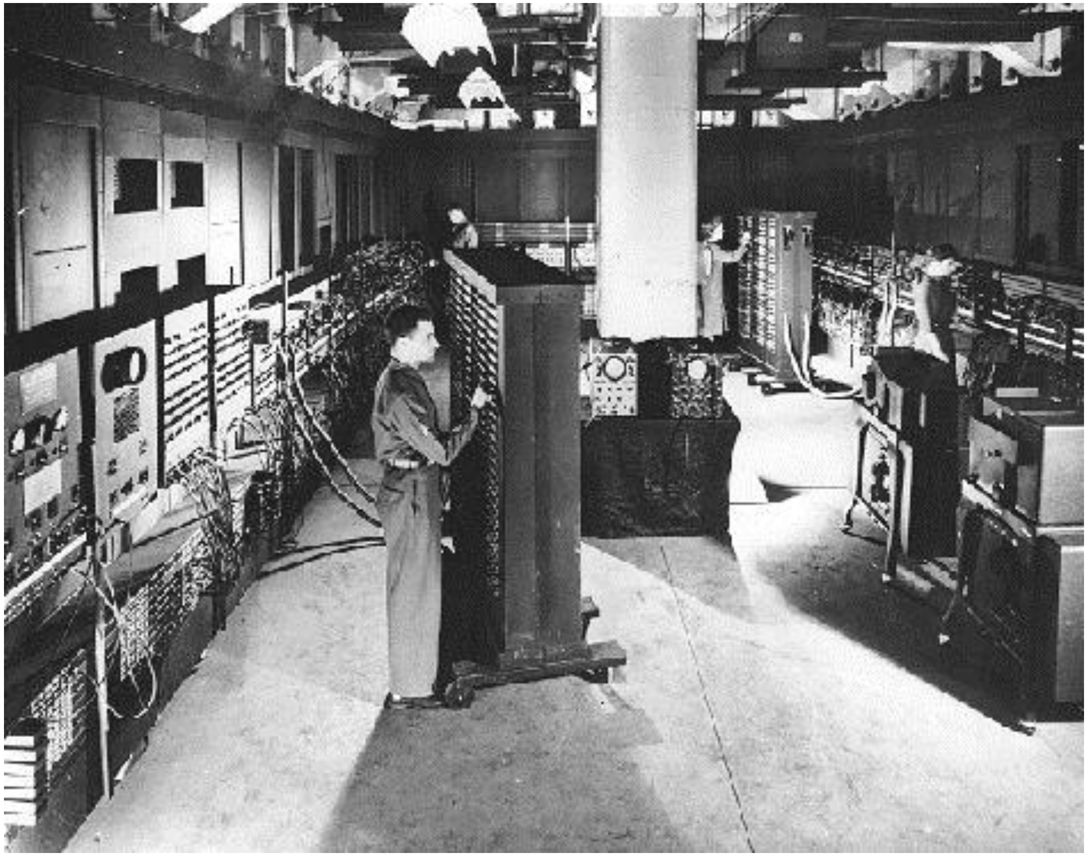




Time-line

- **1980s Desktop**
  - 1977 Apple II
  - 1981 IBM PC
- **1990s PDA**
- **2000s Embeded computers**
- **2010s Cloud computing**
- **2020s Boundless computing, Edge computing**

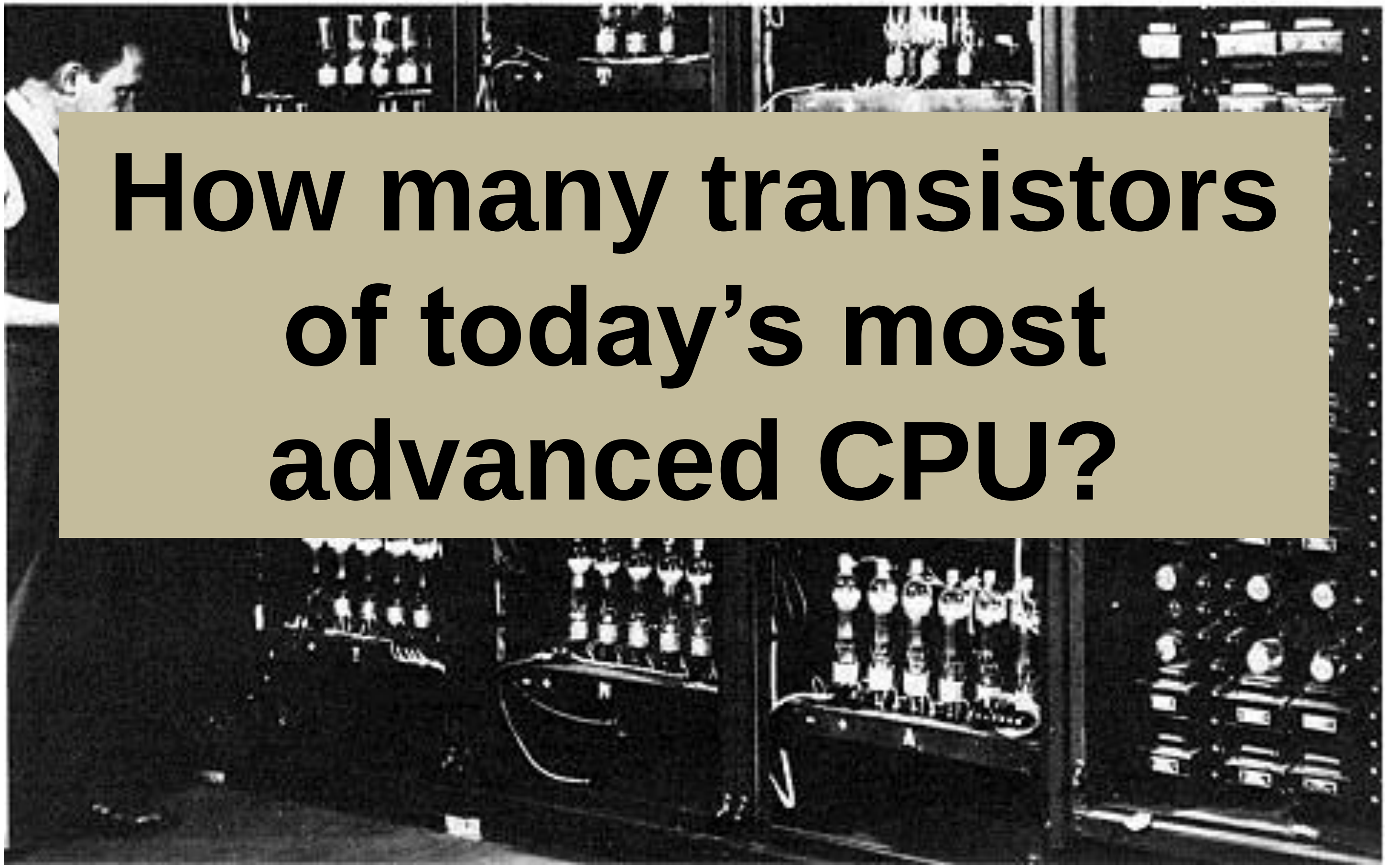
The first electronic computer



ENIAC-1946

18 000 vacuum tubes, 30 ton, 150m^2 ,140kW

The first electronic computer

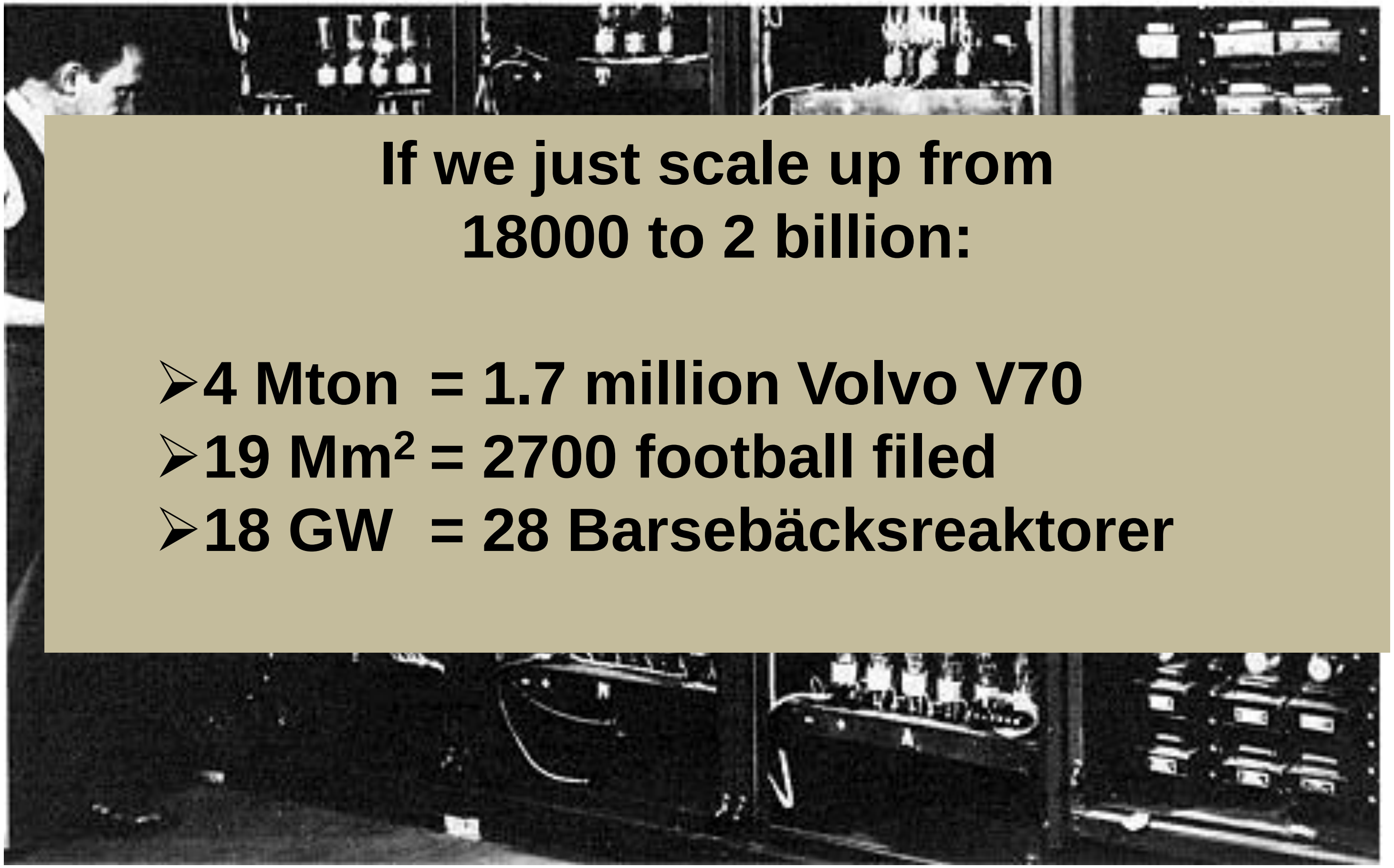


How many transistors of today’s most advanced CPU?

ENIAC-1946

18 000 tubes, 30 ton, 150m^2 ,140kW

The first electronic computer



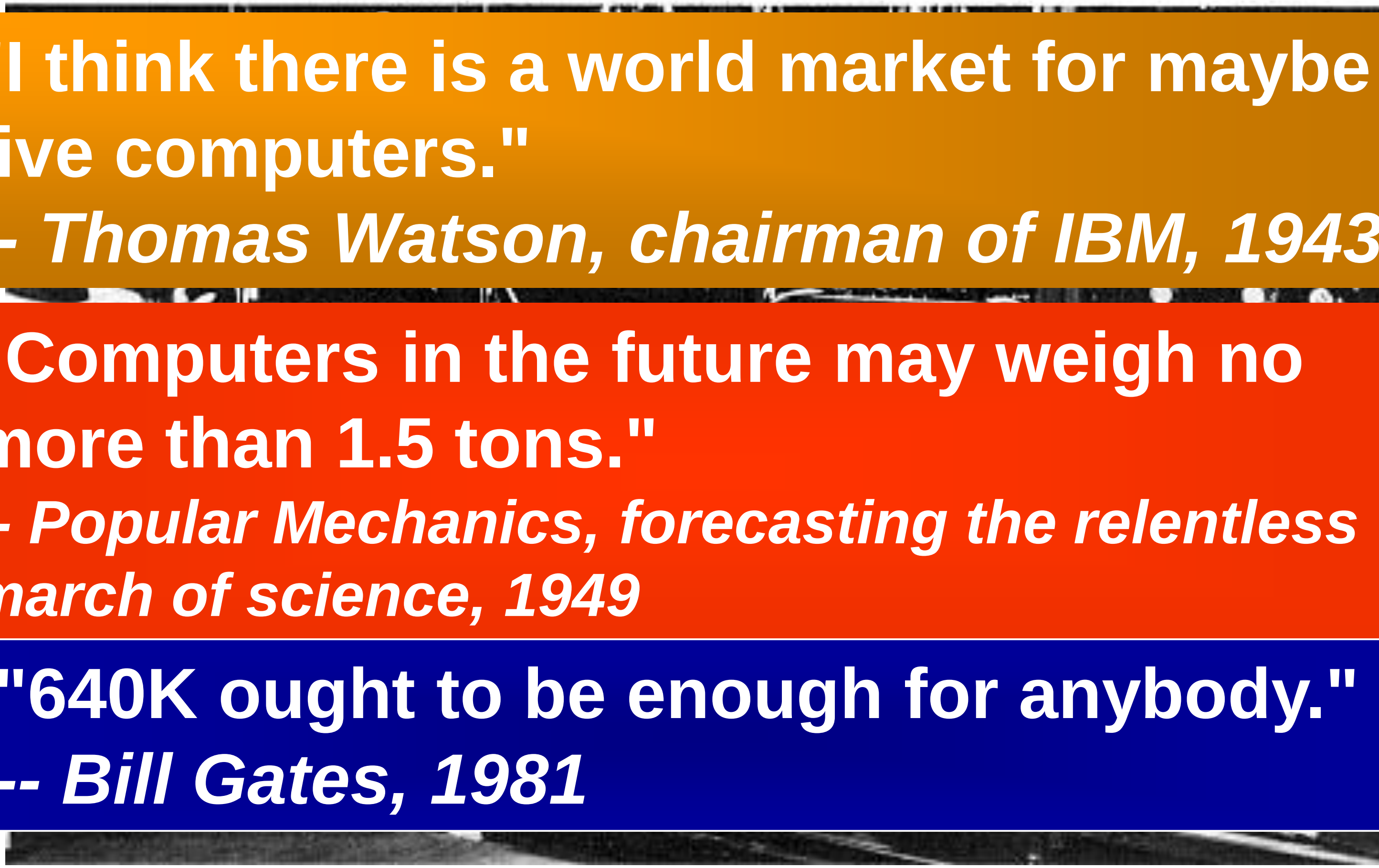
If we just scale up from 18000 to 2 billion:

- **4 Mton = 1.7 million Volvo V70**
- **19 Mm^2 = 2700 football filed**
- **18 GW = 28 Barsebäcksreaktorer**

ENIAC-1946

18 000 tubes, 30 ton, 150m^2 ,140kW

The first electronic computer



"I think there is a world market for maybe five computers."
-- Thomas Watson, chairman of IBM, 1943

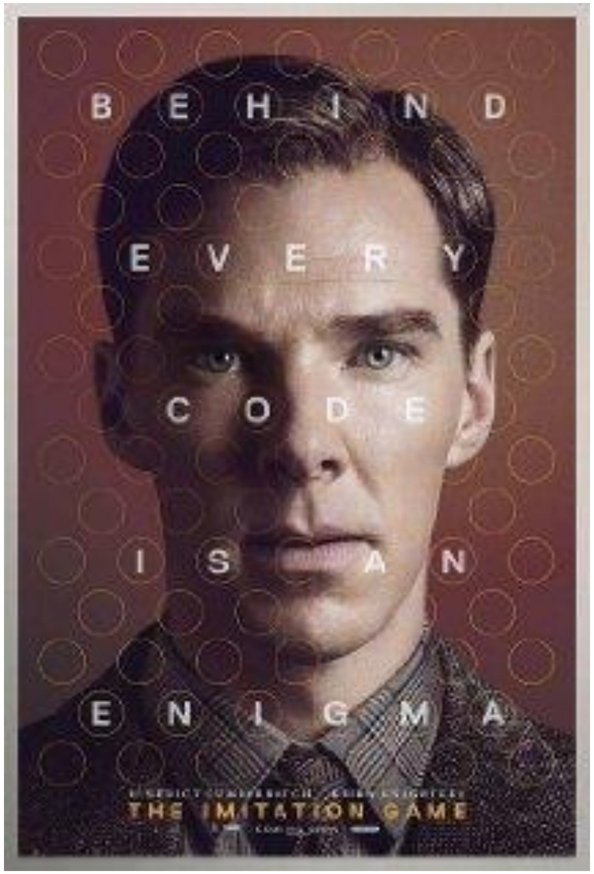
"Computers in the future may weigh no more than 1.5 tons."
-- Popular Mechanics, forecasting the relentless march of science, 1949

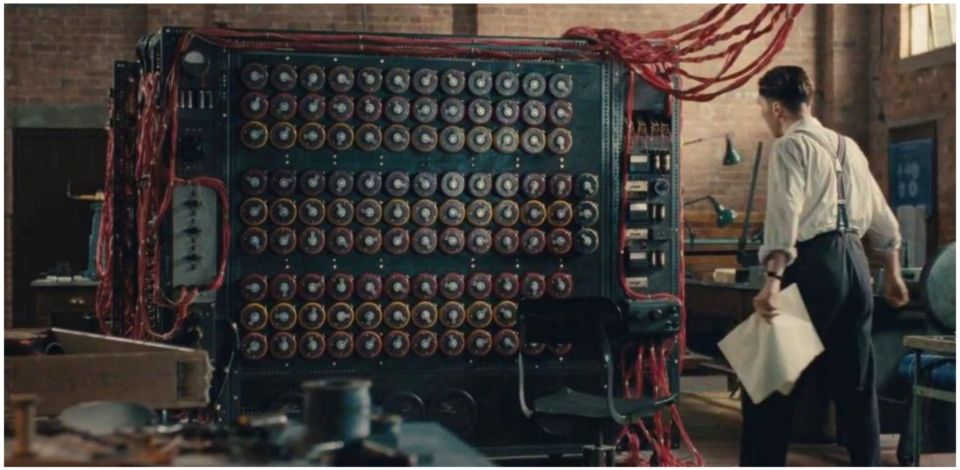
"640K ought to be enough for anybody."
-- Bill Gates, 1981

ENIAC-1946

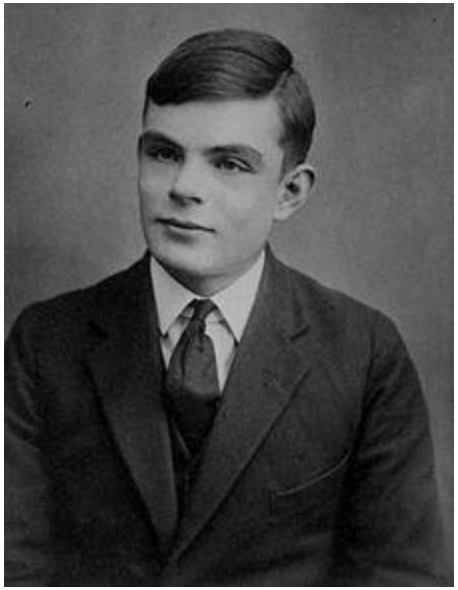
18 000 tubes, 30 ton, 150m^2 ,140kW

Interlude: The imitation game



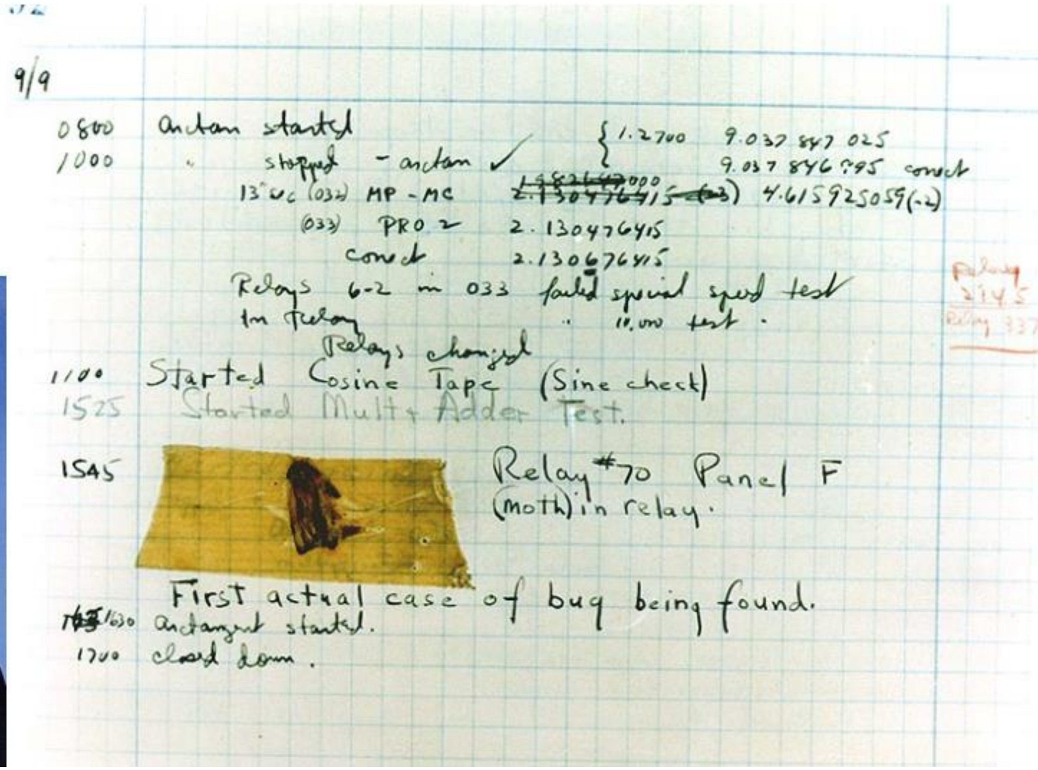


Interlude: Alan Turing



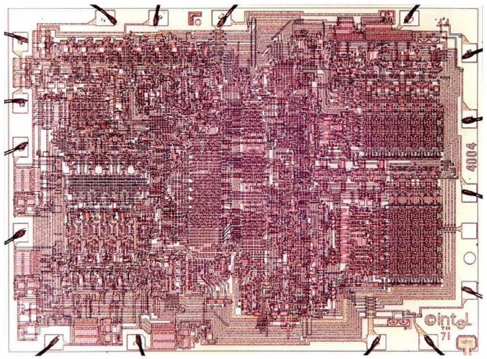
Interlude: Debug

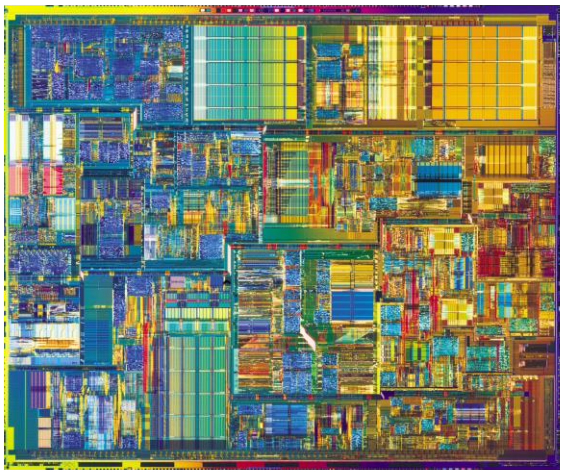
In 1947, Rear Admiral Grace Murray Hopper and associates was working on Mark II, the machine was experiencing problems. An investigation showed that there was a moth trapped in a relay. The operators removed the moth and affixed it to the log. The computer had been “debugged”.

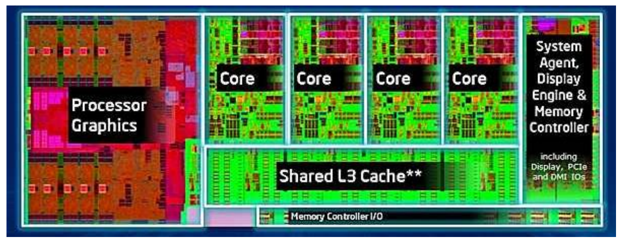


Development of Microprocessor

	Year	Transistors	Frequency	cores	Cache
Intel4004	1971	2300	108 kHz	"1"	None
Z80	1976	8500	2.5 MHz	1	None
Intel386	1985	280 000	16 MHz	1	None
Intel486	1989	1 185 000	20 - 50 MHz	1	8 kB
Pentium 4	2000	44 000 000	1 - 2 GHz	1	256 kB
Nehalem	2008	731 000 000	> 3.6 GHz	4	8 MB
Sandy Bridge	2011	995 000 000	3.8 GHz	4+	8 + 1 MB
Haswell	2013	1 860 000 000	> 3.6 GHz	6	15 + 1.5 M
Itanium 9560	2012	3 100 000 000	2.5 GHz	8	32 + 6 MB





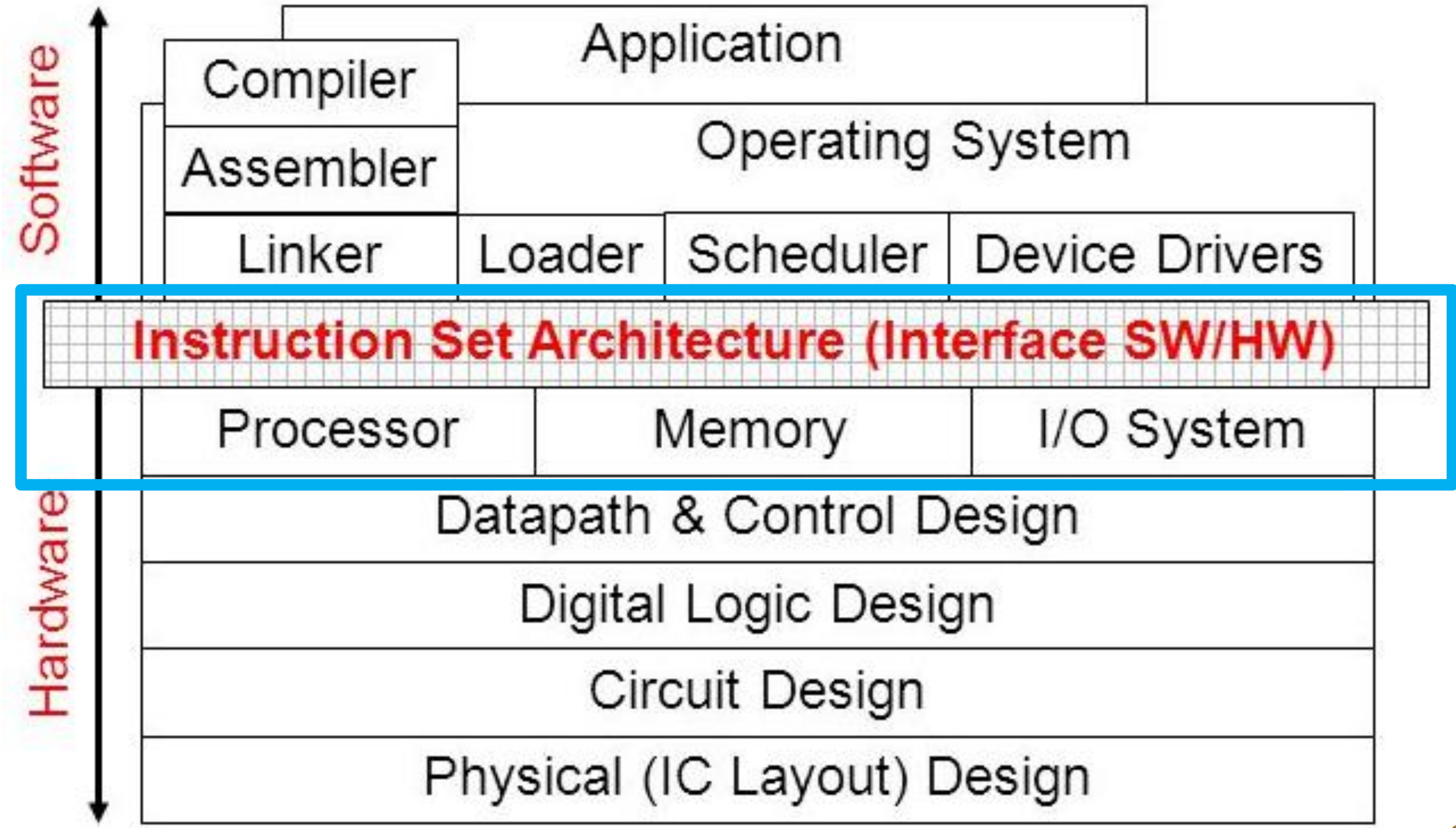


Outline

- Computers
- **Computer Architecture**
- This Course
- Trends
- Performance
- Quantitative Principles

The art of designing computers is based on engineering principles and quantitative performance evaluation

Computer abstraction levels



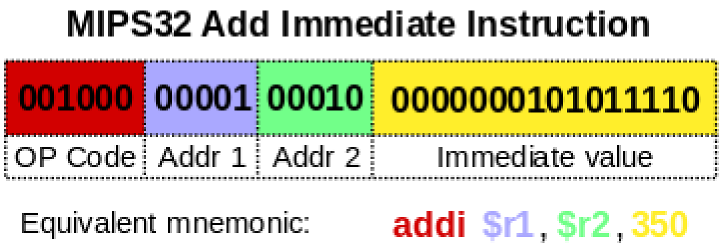
Computer Architecture

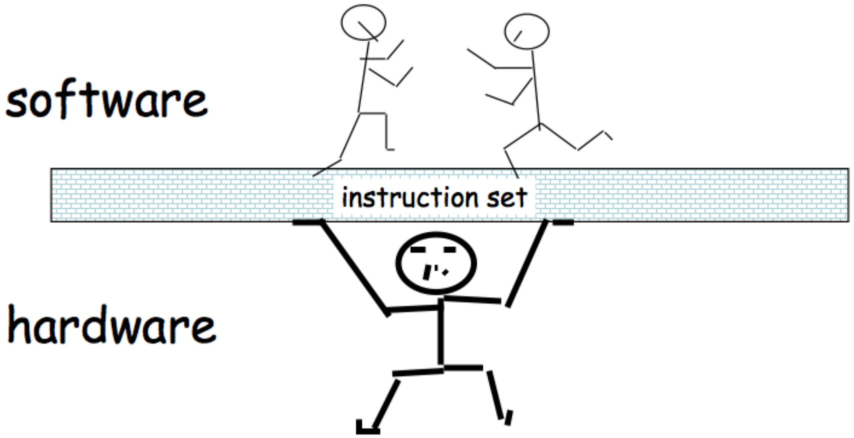
Computer architecture is a set of disciplines that describe the functionality, organization and implementation of computer systems.

- **ISA: Instruction-set architecture**
- **Computer orginization: micro architecture**
- **Specific implementation**

ISA

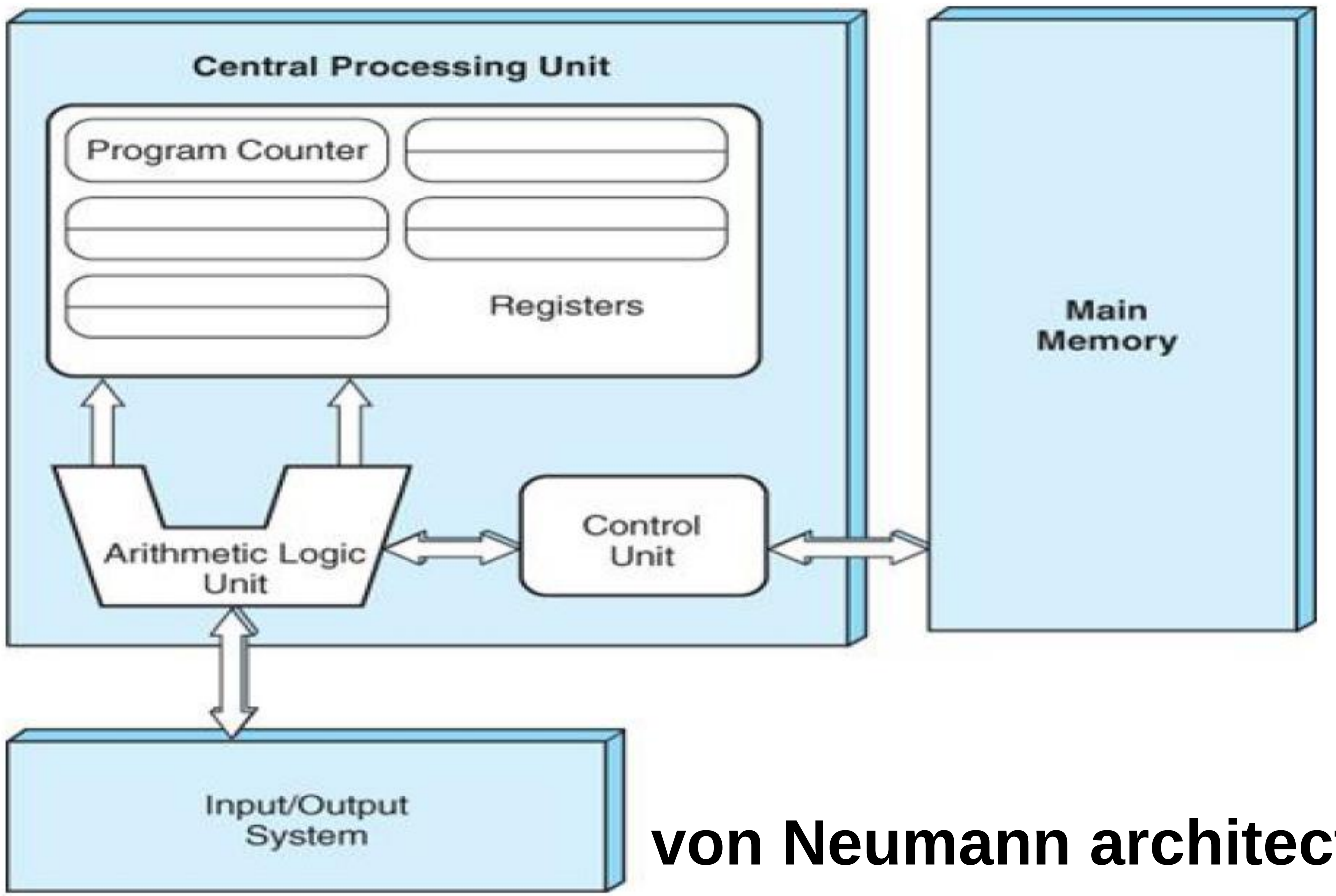
An instruction set architecture (ISA) is the interface between the computer's software and hardware and also can be viewed as the programmer's view of the machine.





Microarchitecture

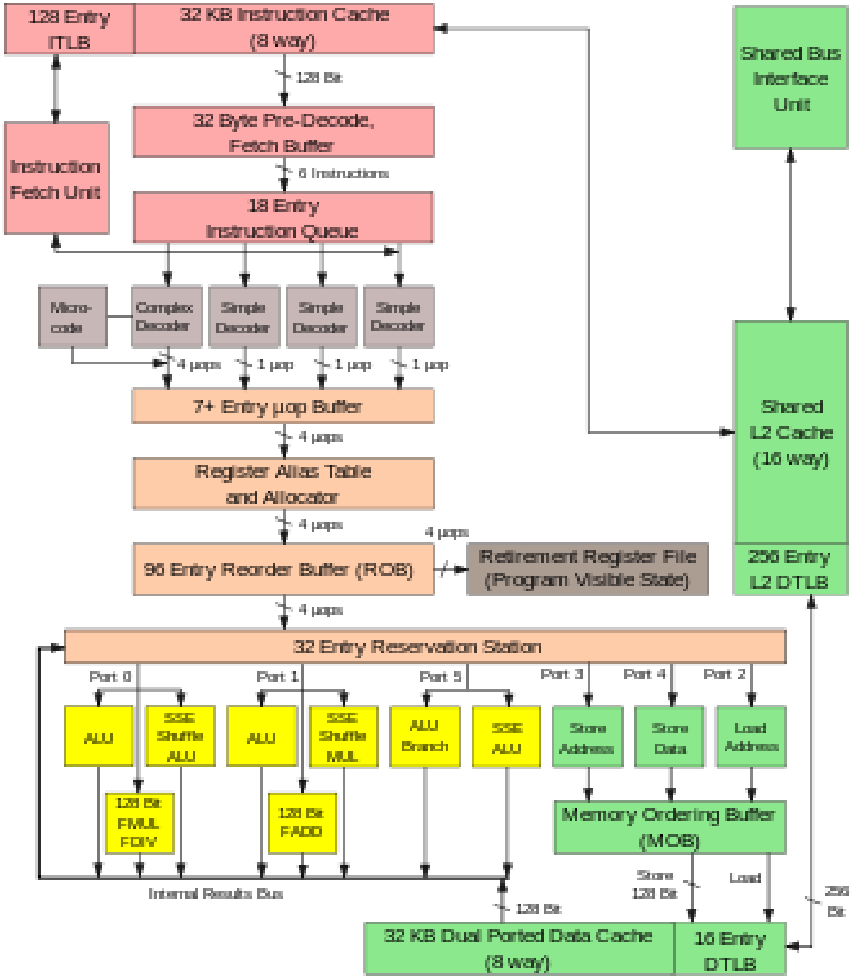
Microarchitecture is the way a given instruction set architecture (ISA) is implemented on a processor.



von Neumann architecture

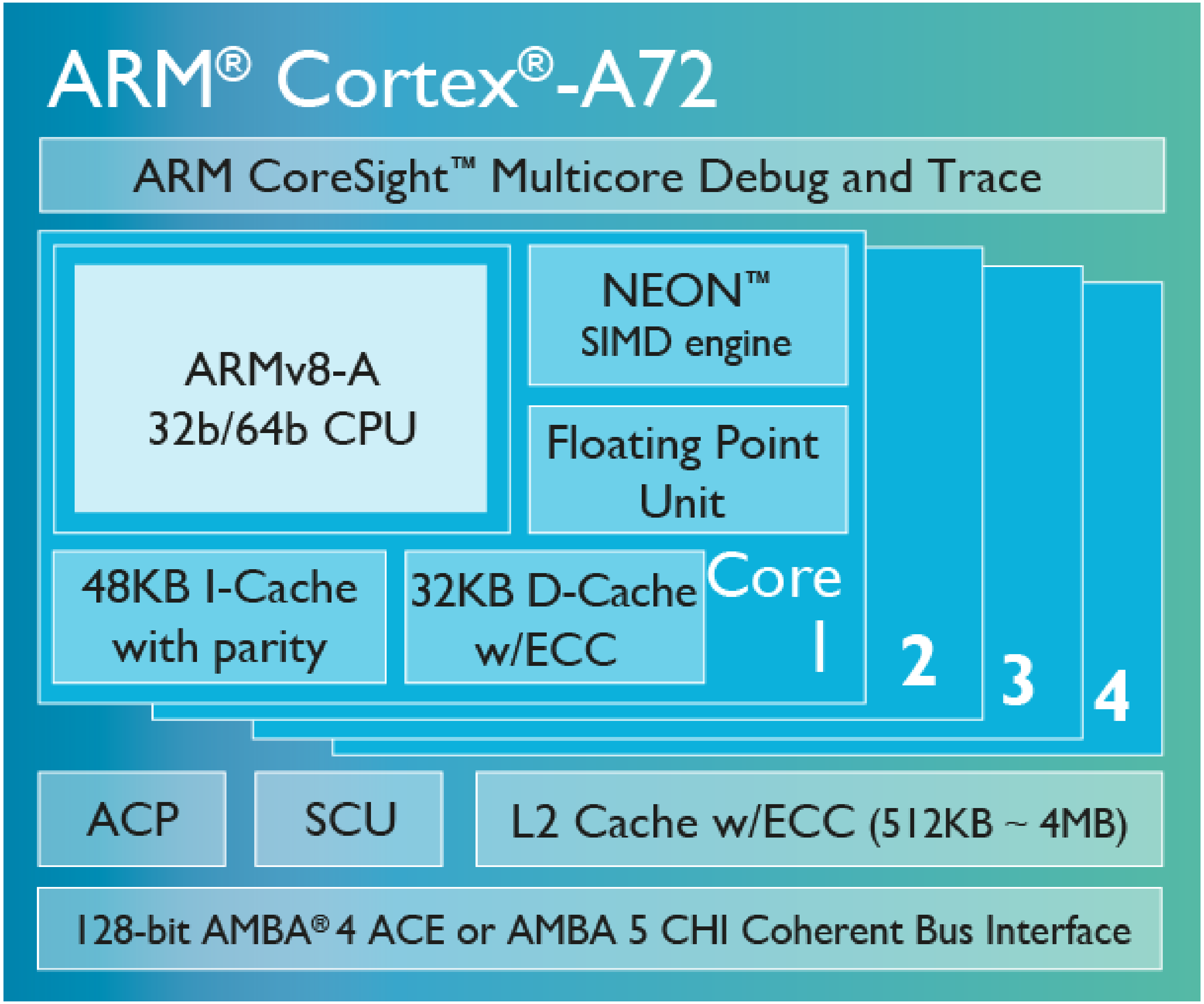
Microarchitecture

Microarchitecture is the way a given instruction set architecture (ISA) is implemented on a processor.

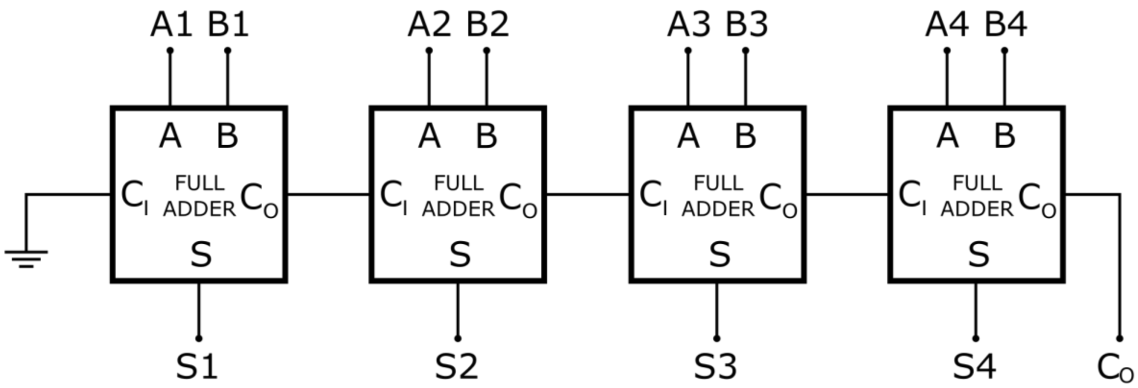


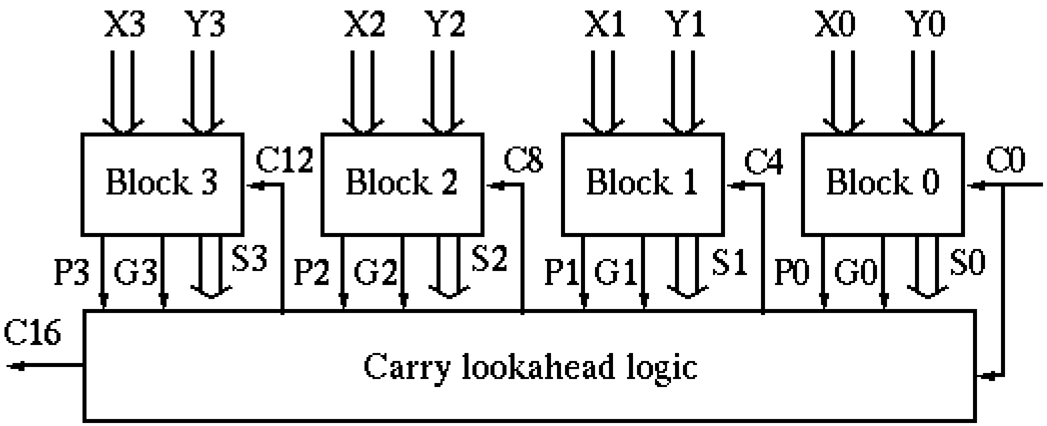
Intel Core 2 Architecture

Microarchitecture



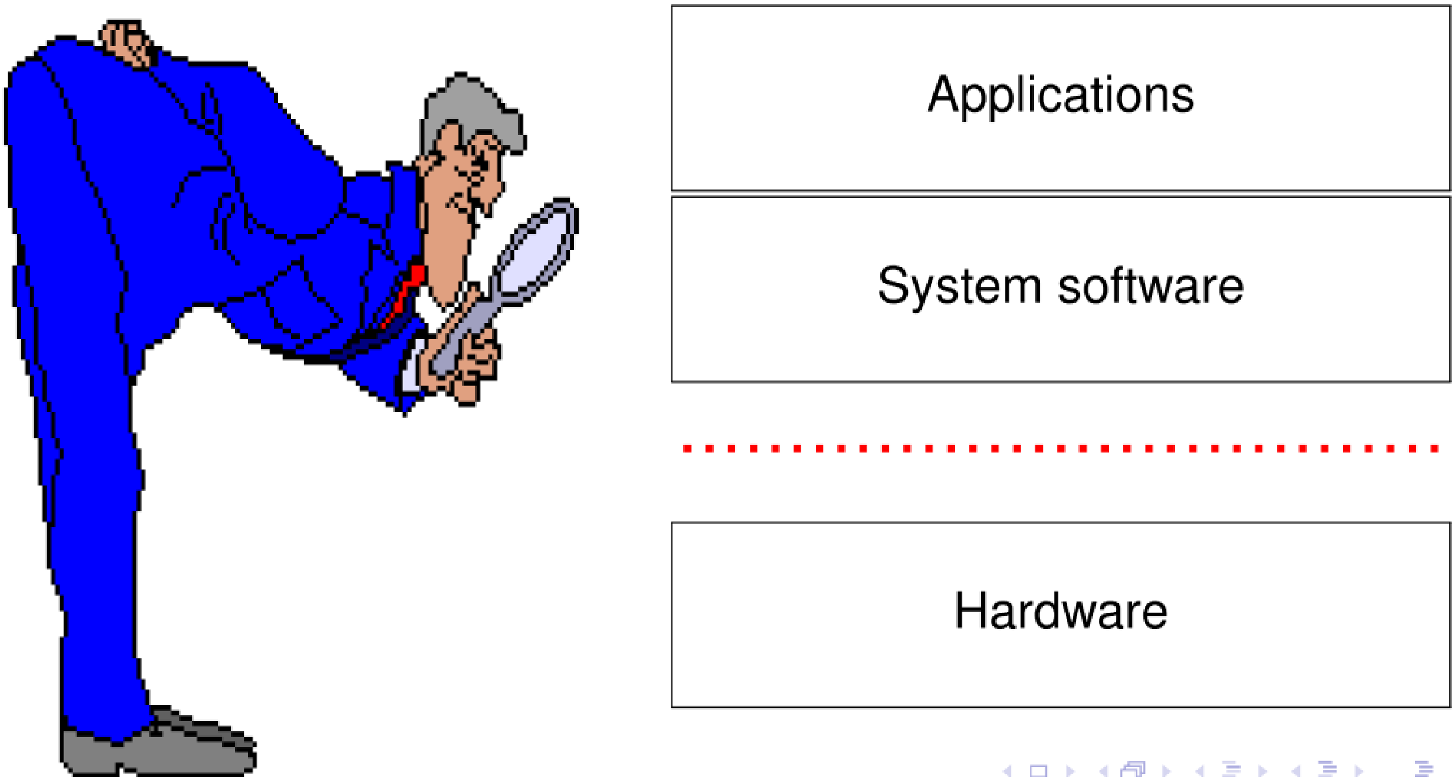
Implementation





The role of computer architecture?

Make design decisions across the interface between hardware and software in order to meet functional and performance goals.



Why computer architecture?

Understand how to evaluate and choose

- What do we mean ”one computer is faster than another”?
- How can Gene Amdahl help you decide which enhancement is the best?
- Is a larger cache better than higher clock frequency?
- Why is pipelining faster than combinational circuits?
- Different levels of caches - why?
- ...

Design your own specialized architecture

- Embedded special purpose processors
  - Axis Communications/Ericsson/Nokia/ARM/SAAB
  - ...

Write better program

What computer architecture?

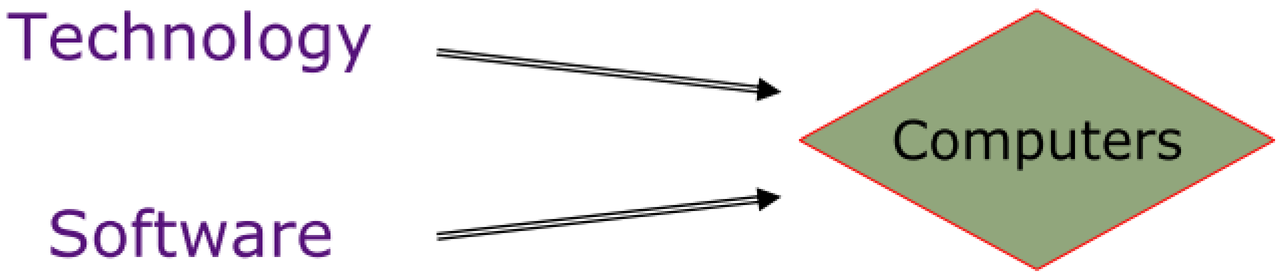
Design and analysis

- ISA
- Orgnization (microarchitecture)
- Implementation

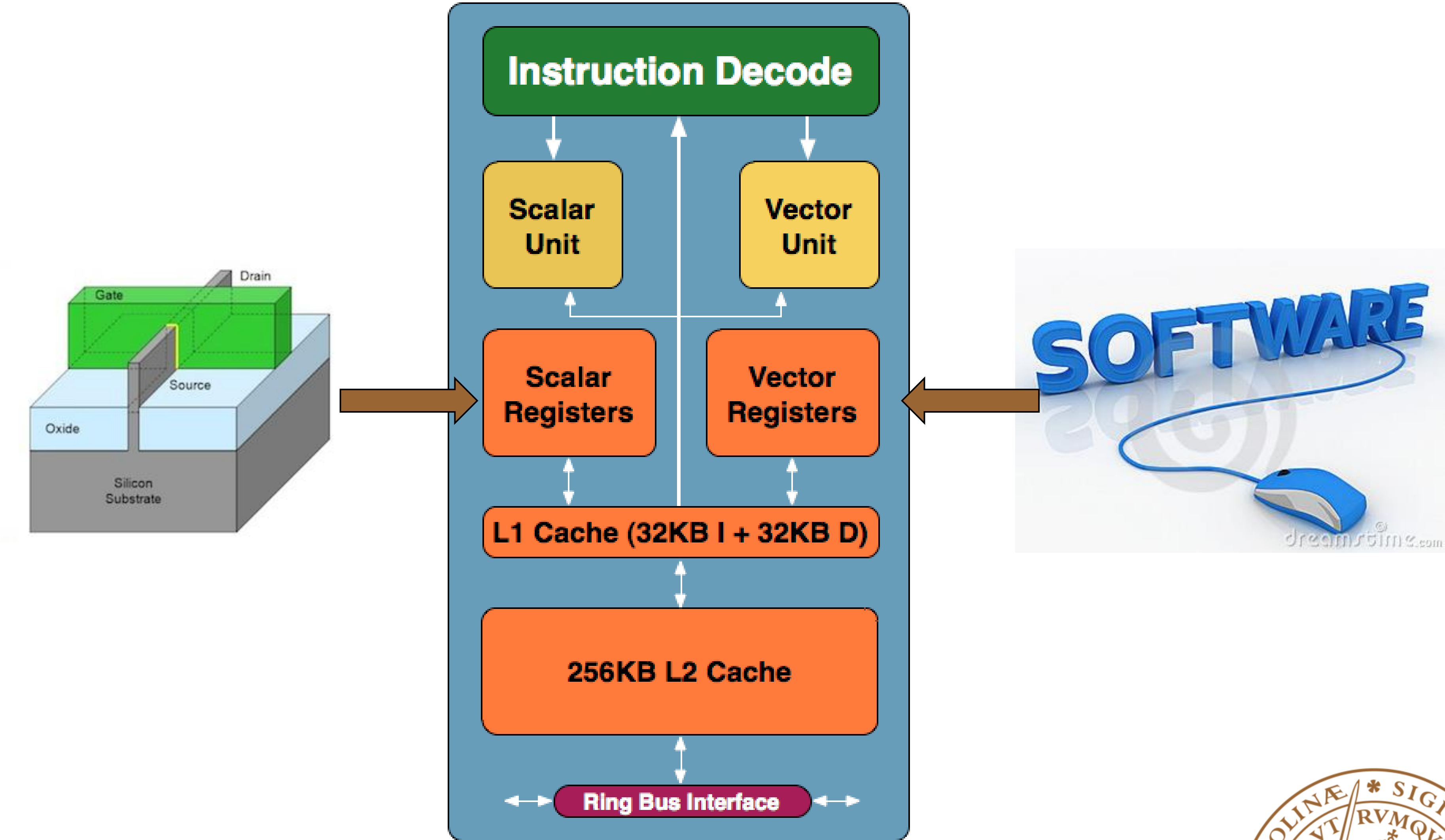
To meet requirements of

- Functionality (application, standards…)
- Price
- Performance
- Power
- Reliability
- Dependability
- Compatability
- ..

What affect computer architecture?



X86 Architecture

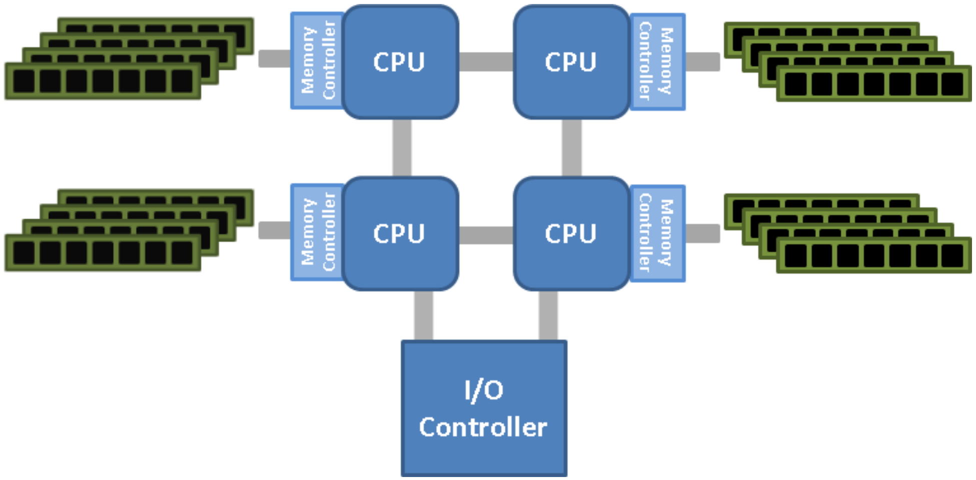


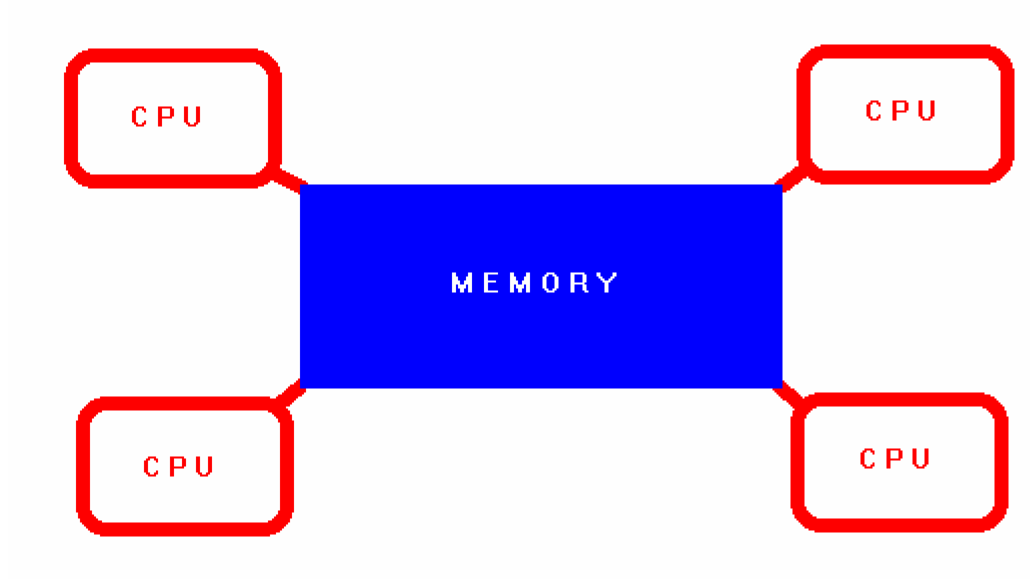
Architecture change due to new applications

ARM

?

Architecture change due to new applications





Outline

- Computers
- Computer Architecture
- **This Course**
- Trends
- Performance
- Quantitative Principles

Course Objectives

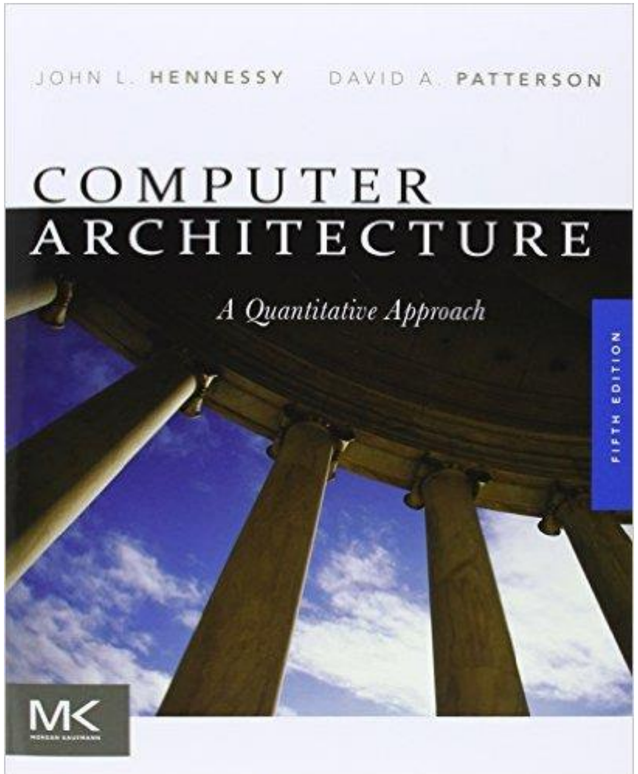
After this course, you will...

- **Have a thorough knowledge about the design principles for modern computer systems**
- **Have an understanding of the relations between**
  - The design of the instruction set of a processor
  - The microarchitecture of a processor
- **Be able to evaluate design alternatives towards design goals using quantitative evaluation methods**
- **Side effects...**
  - Better digital IC designer
  - Better understanding of compiler, operating system, high-performance programming

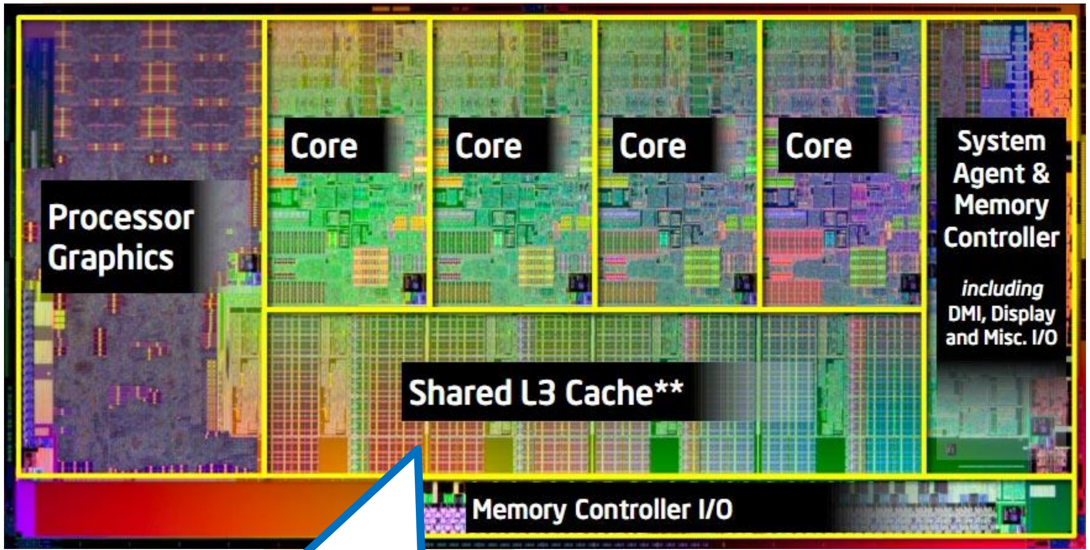
Book Recommendation

Computer Architecture – A Quantitative Approach

- Hennessy, Patterson
- 5th Edition



Course Content & Schedule



- Overview
- Instruction set architecture
- Pipeline
- Memory System
- Storage System
- I/Os
- Multiprocessor

- Concept &Theory
- Assignment & Project
- Exams

Teachers

Lecture

- Liang Liu, Associate Professor
- Email: liang.liu@eit.lth.se
- Room: E2342
- Homepage: http://www.eit.lth.se/staff/Liang.Liu

Teaching Assistants

- Mojtaba Mahdavi
- Steffen Malkowsky

Steffen Malkowsky

Mojtaba Mahdavi

Lectures and Labs

Lectures (10)

- Tuesday : 13:15-15:00 **E:B (V:B)**
- Thursday: 08:15-10:00 **E:1406 (E:B)**
- Covers design principles and analysis methodology
- Read the literature before each lecture
- Does not cover all of the literature
- Ask many questions!

Labs (4)

- Tuesday: 08:15-12:00 **E:4118-E:4119**
- Friday: 08:15-12:00 (except for last one) **E:4118-E:4119**
- 2 students/group
- Read manual and literature before the lab
- Do Home Assignments before lab (**or be sent home**)
- Experiment and discuss with assistants
- Understand what you have done (**or FAIL the exam**)
- Finish Lab before **DEADLINE**

Examination (Written)

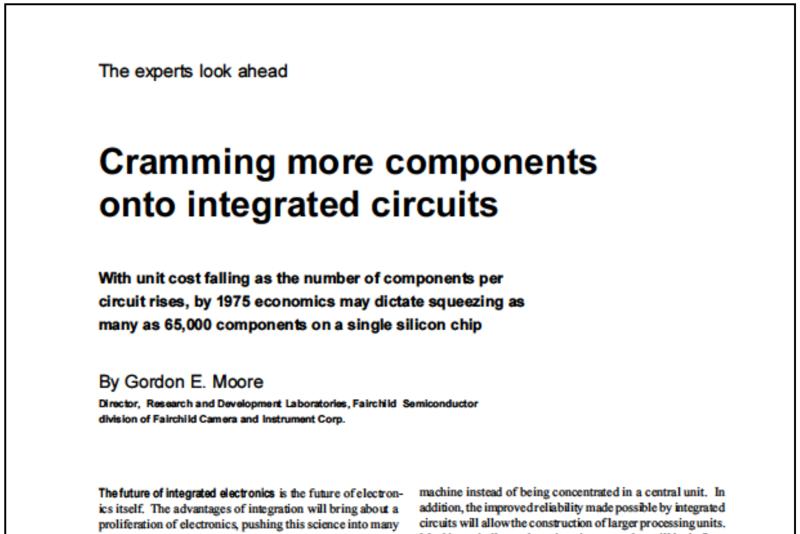
- **Anonymous exam**
- **Pass all labs to be able to attend written exam**
- **Five problems**
  - Highly lab related
  - Problem solving
  - Descriptive nature

Questions?

Outline

- Computers
- Computer Architecture
- This Course
- **Trends**
- Performance
- Quantitative Principles

Moore's Law


The experts look ahead

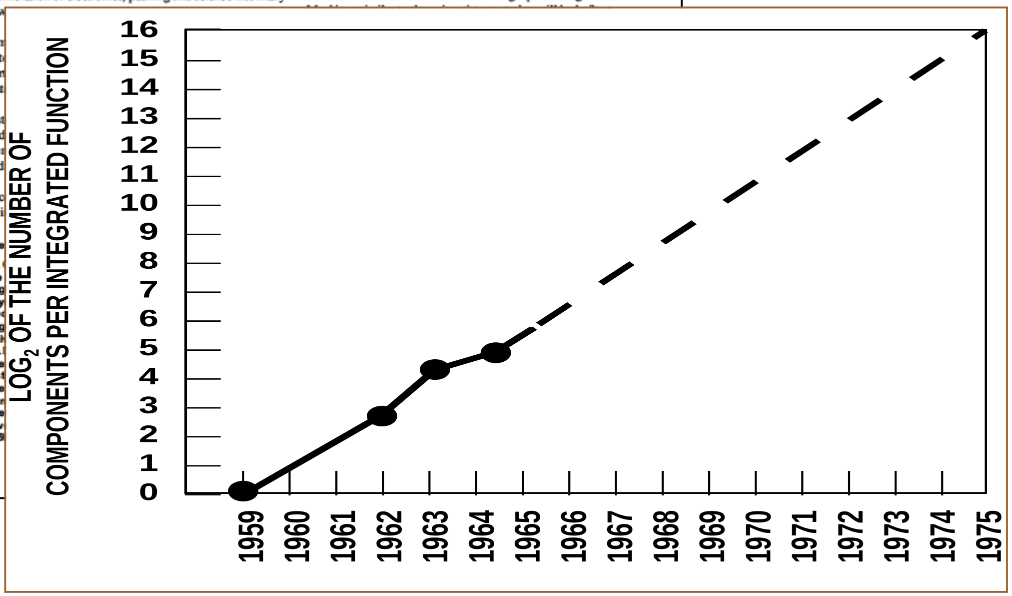
Cramming more components onto integrated circuits

With unit cost falling as the number of components per circuit rises, by 1975 economics may dictate squeezing as many as 65,000 components on a single silicon chip

By Gordon E. Moore
Director, Research and Development Laboratories, Fairchild Semiconductor division of Fairchild Camera and Instrument Corp.

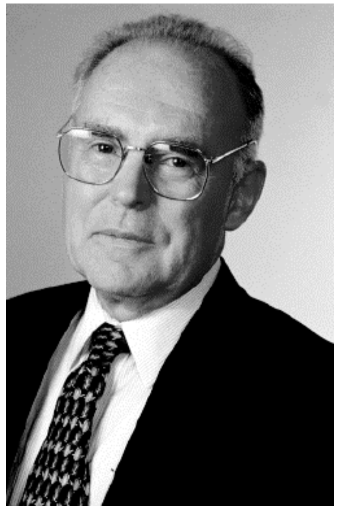
The future of integrated electronics is the future of electronics itself. The advantages of integration will bring about a proliferation of electronics, pushing this science into many

machine instead of being concentrated in a central unit. In addition, the improved reliability made possible by integrated circuits will allow the construction of larger processing units.



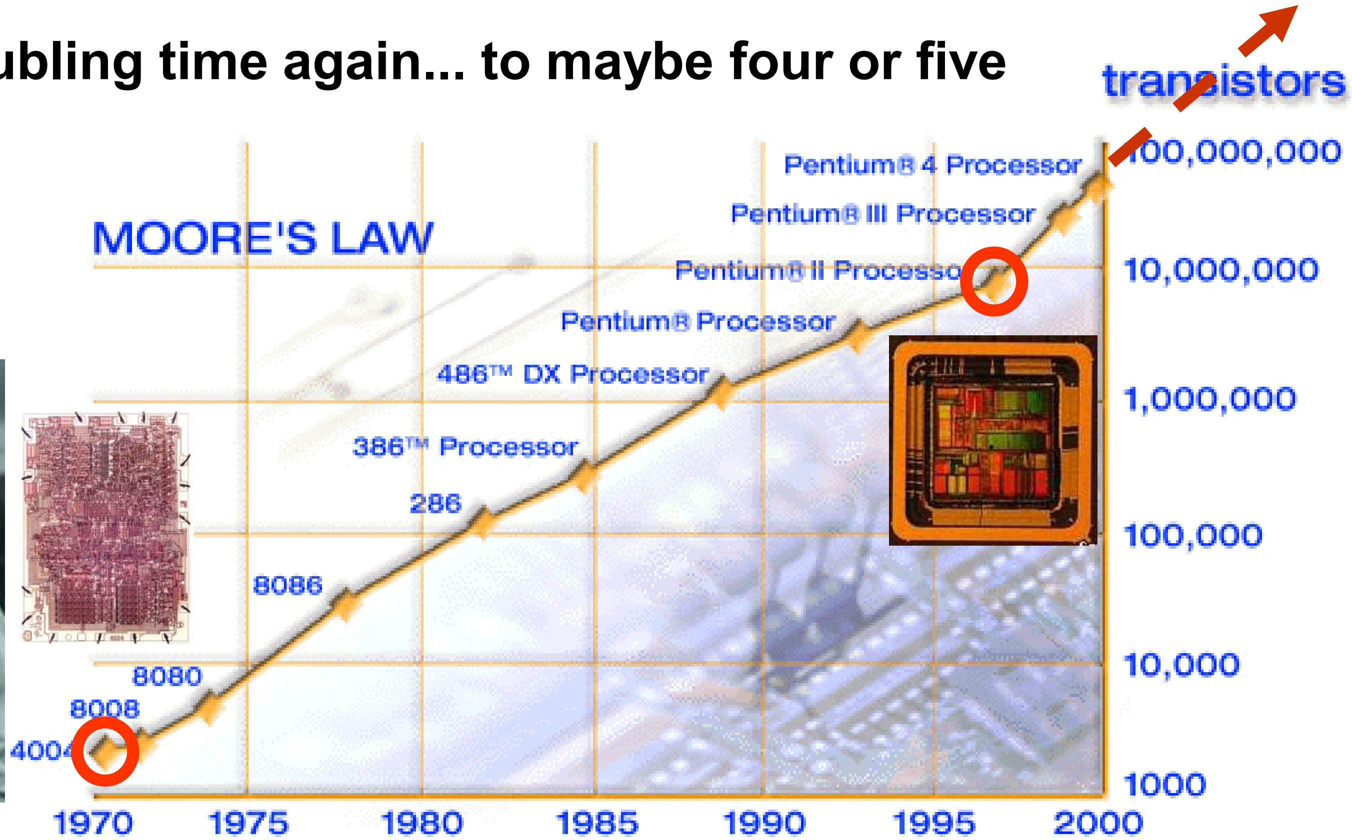
- **Electronics, Apr. 19, 1965**

Gordon Moore (co-founder of Intel) described a doubling every year in the number of components per integrated circuit



Moore's Law

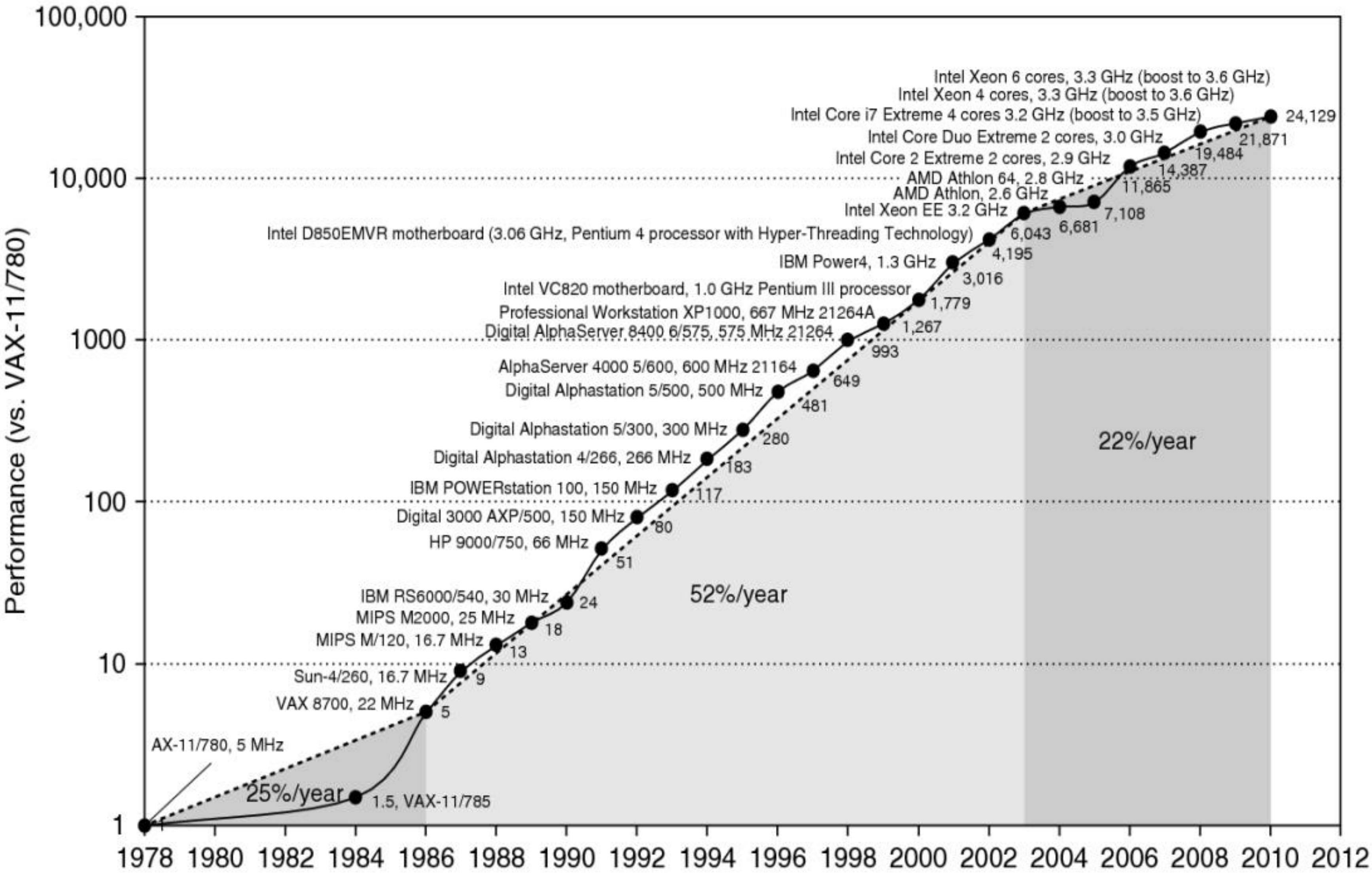
Moore reformulates to a doubling every 2 years. (1975)
Interview 2000:
"...change the doubling time again... to maybe four or five years."



Gordon Moore
Co-founder of Intel

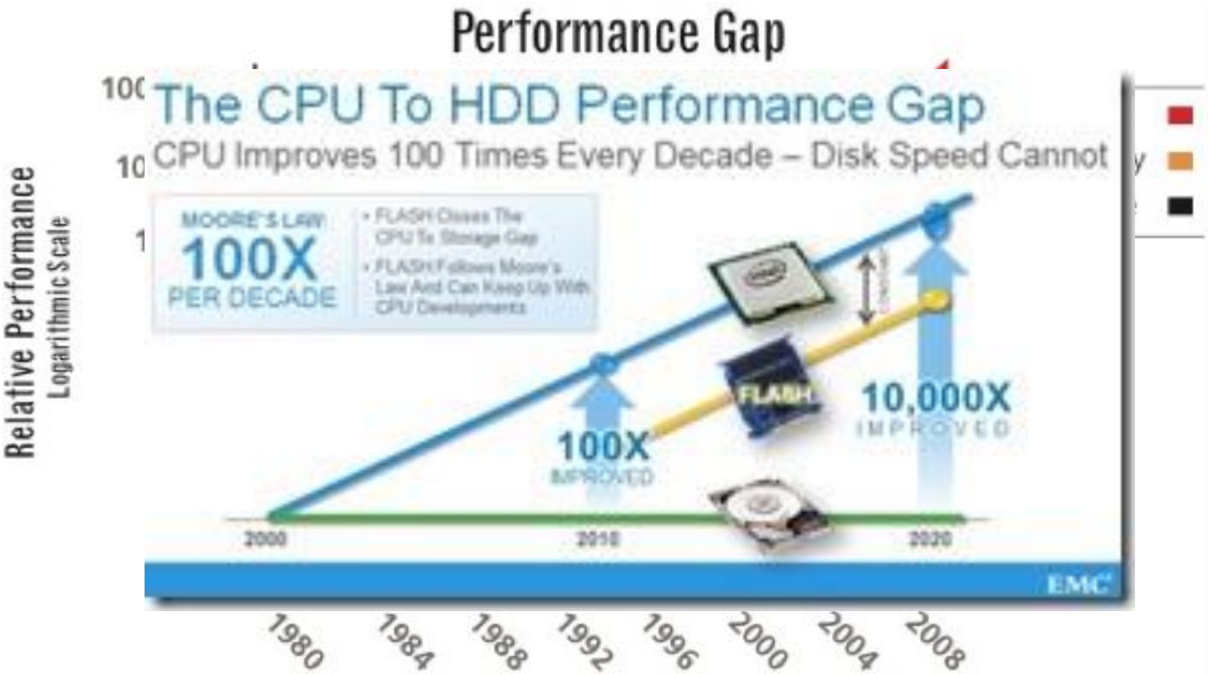
ca. 1 billion transistors in 2007

Performance of Microprocessor

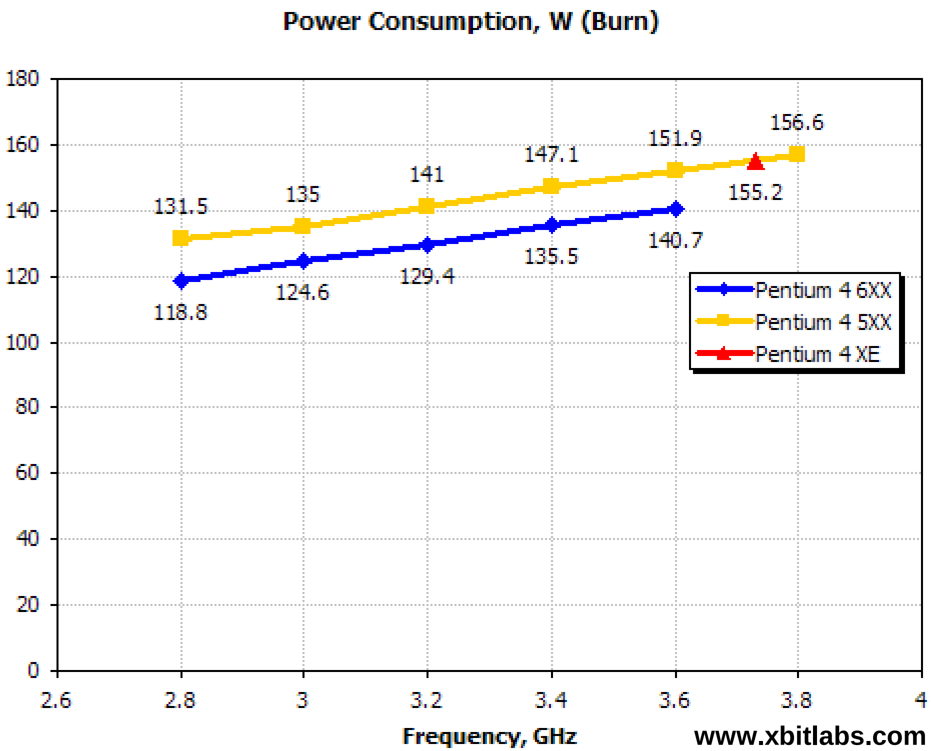


Does not Apply to All

- Processing power doubles every 18 months
- Memory size doubles every 18 months
- Disk capacity doubles every 18 months
- Disk positioning rate (seek & rotate) doubles every ten years!
- Speed of DRAM and disk improves a few % per year



Moore’s Law: power density

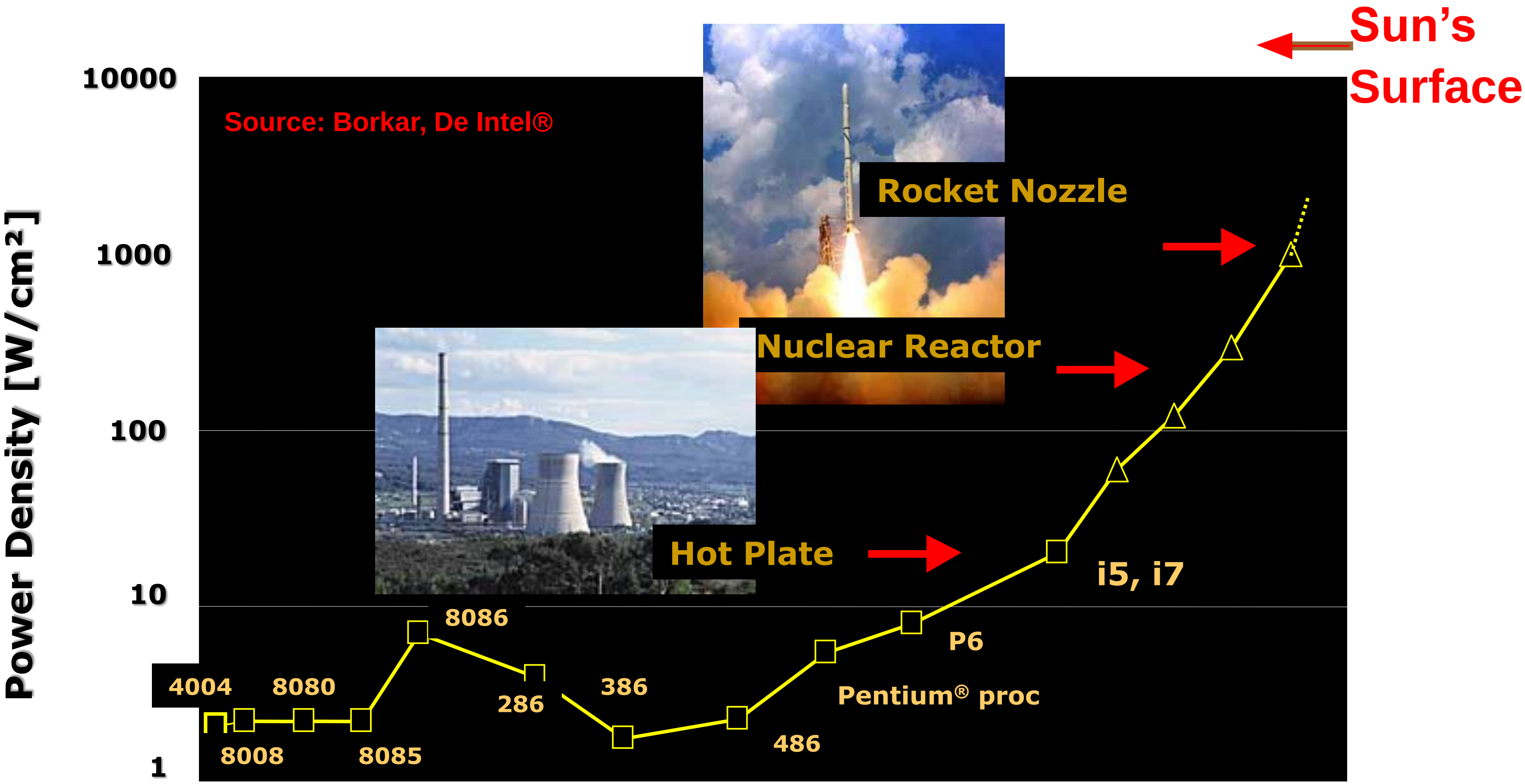


Pentium IV chip area (in 130 nm technology) **1.3 cm²**

This gives about 100 W/cm² that needs to be transported away (cooling)

Comparison: This little thing operates at about 10 W/cm².

Moore’s Law: power density



Outline

- Computers
- Computer Architecture
- This Course
- Trends
- **Performance**
- Quantitative Principles

Performance

$$Performance(X) = \frac{1}{T_{exe}(X)}$$

"X is n times faster than Y" means:

$$\frac{T_{exe}(Y)}{T_{exe}(X)} = \frac{Performance(X)}{Performance(Y)} = n$$

How to define execution time?

Performance

Application	$\Longleftarrow$	Answers/month
Programming	$\Longleftarrow$	Response time (seconds)
language	$\Longleftarrow$	Operations/second
Compiler		
Instruction set	$\Longleftarrow$	MIPS/MFLOPS
Data-path control	$\Longleftarrow$	Megabytes/second
Functional units		
Transistors, wires, pins	$\Longleftarrow$	Cycles per second (clock rate)

MIPS = millions of instructions per second
MFLOPS = millions of FP operations per second

Program to evaluate performance

- **Real programs: e.g. TeX, spice, SPEC benchmarks, ...**
- **Kernels - small, key pieces of real applications**
- **Toy programs - sort, prime number generation**
  - Something 100-line programs
- **Synthetic benchmarks - “The average program”**
  - Fake programs to mathc the behaviour of real applications
  - **Real programs are the only true measurement objects**
- **SPEC benchmarks will be used here (*plus some toy programs*)**
  - Real programs modified to be portable and to minimize the effect of IO

Which Computer is Faster?

Execution time			
Computer	**A**	**B**	**C**
Program P1	1	10	20
Program P2	1000	100	20
Total time	1001	110	40

- ☐ **A is 10 times faster than B for P1**
- ☐ **B is 10 times faster than A for P2**
- ☐ **A and B are faster than C for P1**
- ☐ **C is faster than A and B if both P1 and P2 are run**

Which Computer is Faster?

Execution time			
Computer	**A**	**B**	**C**
Program P1	1	10	20
Program P2	1000	100	20
Total time	1001	110	40

- Arithmetic mean of execution time: $\frac{\sum T_i}{n}$
  or weighted execution time $\frac{\sum W_i * T_i}{n}$

Outline

- Computers
- Computer Architecture
- This Course
- Trends
- Performance
- **Quantitative Principles**

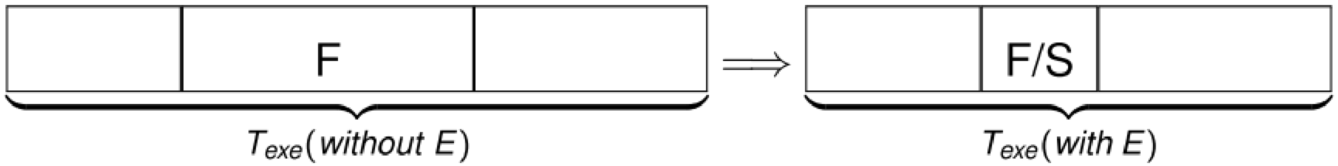
Quantitative Principles

This is intro to design and analysis

- Take advantage of parallelism
  - ILP, DLP, TLP, ...
- Principle of locality
  - 90% of execution time in only 10% of the code
- Focus on the common case
  - In makeing a design trade-off, favor the frquent case ove the infrequent case
- Amdahl's Law
  - The performance improvement gained from uisng faster mode is limited by the fraction of the time the faster mode can be used
- The Processor Performance Equation

Amdahl's Law

Enhancement E accelerates a fraction F of a program by a factor S



Speedup due to enhancement E:

$$Speedup(E) = \frac{T_{exe}(without\ E)}{T_{exe}(with\ E)} = \frac{Performance(with\ E)}{Performance(without\ E)}$$

$$T_{exe}(with\ E) = T_{exe}(without\ E) * [(1 - F) + F/S]$$

$$Speedup(E) = \frac{T_{exe}(without\ E)}{T_{exe}(with\ E)} = \frac{1}{(1-F)+F/S}$$

Best you could ever hope to do:

$$Speedup_{maximum} = \frac{1}{(1 - Fraction_{enhanced})}$$

Amdahl's Law: example

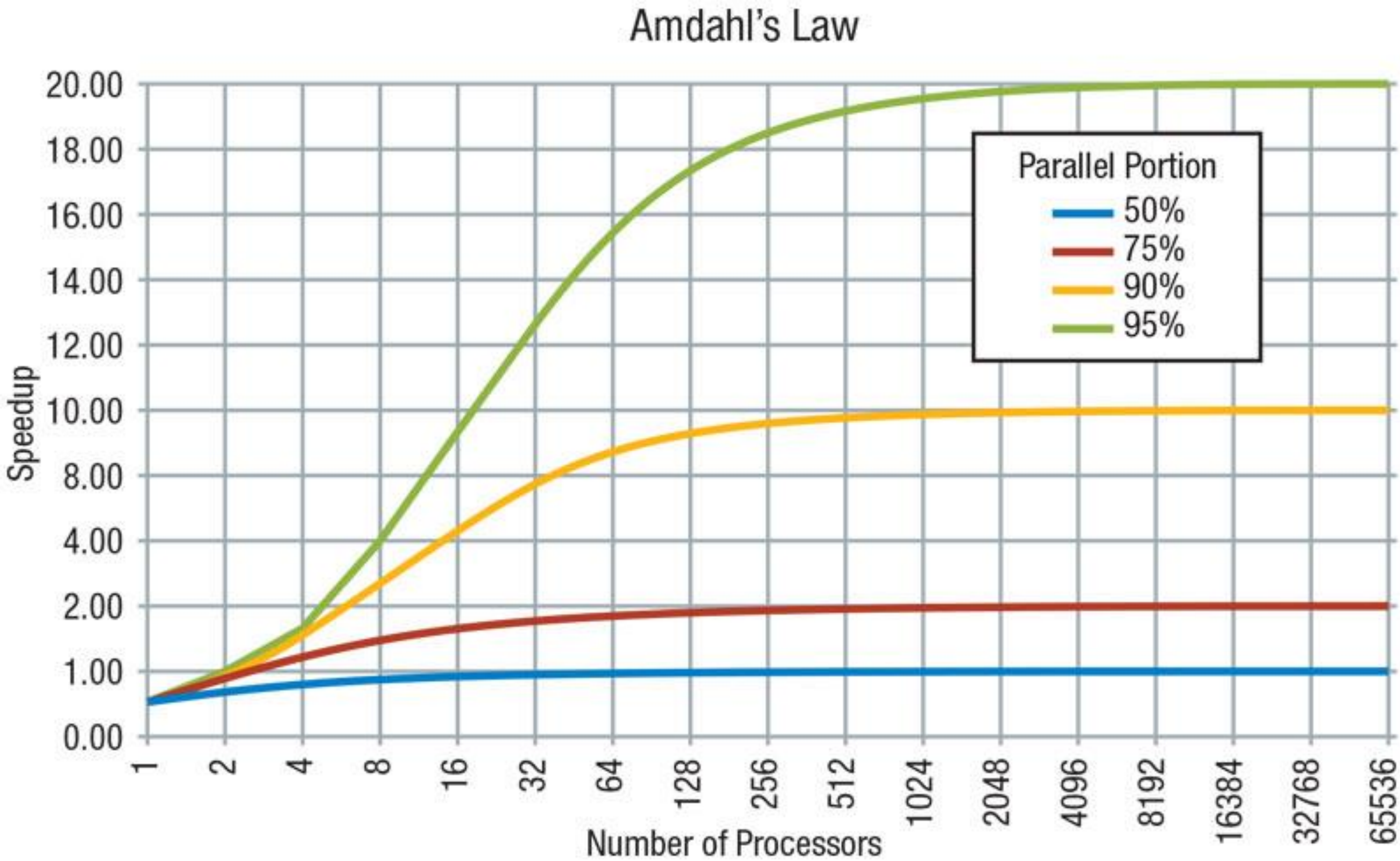
- New CPU is 10 times faster!
- 60% for I/O which remains almost the same…

$$\text{Speedup}_{\text{overall}} = \frac{1}{\left(1 - \text{Fraction}_{\text{enhanced}}\right) + \frac{\text{Fraction}_{\text{enhanced}}}{\text{Speedup}_{\text{enhanced}}}}$$

$$= \frac{1}{\left(1 - 0.4\right) + \frac{0.4}{10}} = \frac{1}{0.64} = 1.56$$

Apparently, its human nature to be attracted by 10X faster, vs. keeping in perspective its just 1.6X faster

Amdahl’s Law: example



http://www-inst.eecs.berkeley.edu/~n252/paper/Amdahl.pdf

Aspect of CPU performance

$$\text{CPUtime} = \text{Execution time} =$$
$$seconds/program =$$

$$\underbrace{(executed)instr./program}_{IC} * \underbrace{cycles/instr.}_{CPI} * \underbrace{seconds/cycle}_{T_c}$$

	IC	CPI	T_c
Program	X		
Compiler	X	(X)	
Instr. Set	X	X	
Organization		X	X
Technology			X

Instructions are not created equally

"Average Cycles per Instruction"

CPI_{op} = Cycles per Instruction of type *op*

IC_{op} = Number of executed instructions of type *op*

$$CPUtime = T_c * \sum (CPI_{op} * IC_{op})$$

"Instruction frequency"

$$\overline{CPI} = \sum (CPI_{op} * F_{op}) \text{ where } F_{op} = IC_{op}/IC$$

Average CPI: example

Op	F_{op}	CPI_{op}	$F_{op} * CPI_{op}$	% time
ALU	50 %	1	0.5	(33 %)
Load	20 %	2	0.4	(27 %)
Store	10 %	2	0.2	(13 %)
Branch	20 %	2	0.4	(27 %)

$$\overline{CPI} = 1.5$$

Invest resources where time is spent!