COMP2611 COMPUTER ORGANIZATION
TOPIC 1 INTRODUCTION

Common Number System

- **A number system defines how a number can be represented using distinct symbols.**
- **A number can be represented differently in different systems**
  - For example, the two numbers $(2A)_{16}$ and $(101010)_2$ both refer to the same quantity, $(42)_{10}$.

Positional Notation

- Each digit position has an associated weight
- Numeric values are determined by the implicit positional values of the digits.

$$642_{(10)} = 6 \times 10^2 + 4 \times 10^1 + 2 \times 10^0$$

Base or Radix | Hundreds | Tens | Units | Position

- Positional notation as a formula

$$d_{n-1} \times R^{n-1} + d_{n-2} \times R^{n-2} + \ldots + d_1 \times R^1 + d_0 \times R^0$$

  - d_i: digit at i^{th} position
  - R: base or radix

Common Number Systems

System	Base	Symbols	Remark
Decimal	10	0, 1, … 9	used by people
Binary	2	0, 1	used by digital computer
Hexadecimal	16	0, 1, … 9, A, B, … F	great ways to concisely represent a binary sequence

Binary Number System (base 2)

- **Used to model the series of electrical signals computers use to represent information**
- **Base 2, two symbols: 0, 1 (binary digits, or just bits)**
  - 0 : no voltage or an off state
  - 1 : presence of voltage or an on state
- **A sequence of bits (a.k.a. bit sequence) usually work together**

Binary <-> Decimal Conversion

- **Binary -> Decimal**
  - Expand using positional notation

$$100101_{(2)} = 1 \times 2^5 + 0 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0$$
$$= 32 + 0 + 0 + 4 + 0 + 1$$
$$= 37_{(10)}$$

- **Decimal -> Binary**
  - Do the reverse
  - Determine largest power of 2 ≤ number, write template and solve it

$$37_{(10)} = ? \times 2^5 + ? \times 2^4 + ? \times 2^3 + ? \times 2^2 + ? \times 2^1 + ? \times 2^0$$
$$= 1 \times 2^5 + 0 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0$$

Size vs. Rate/Frequency

- **When dealing with a size (e.g., Memory or file)**
  - Kilo – 2^{10} or 1024
  - Mega – 2^{20} or 1024 Kilo
  - Giga – 2^{30} or 1024 Mega
  - Tera – 2^{40} or 1024 Giga
  - Peta - 2^{50} or 1024 Tera
  - …

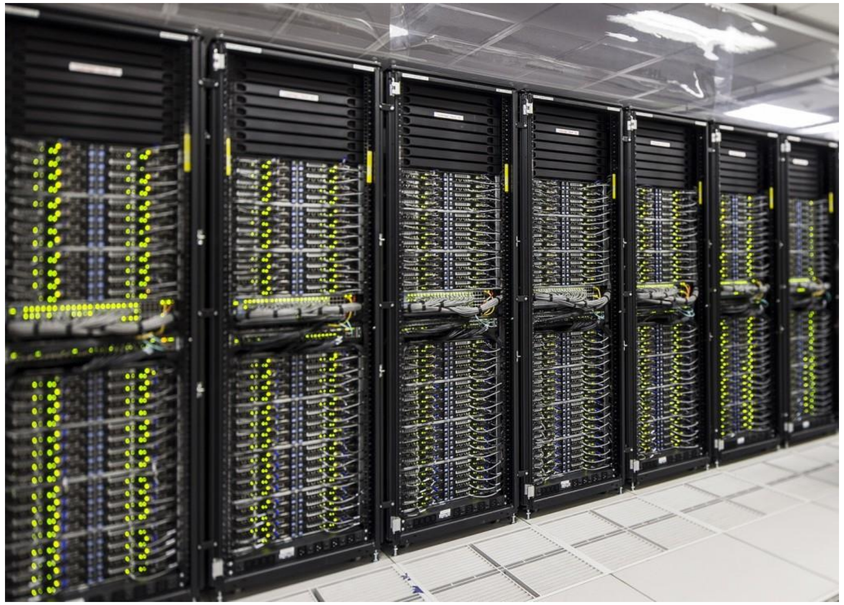
  Example:
  - The memory in my computer is 4 Gigabytes
  - The PPT file for this lecture is 2.5 Megabytes

- **When dealing with a rate/frequency (e.g., # instructions per second, # clock ticks per second)**
  - Kilo – 10^{3} or 1000
  - Mega – 10^{6} or 1000 Kilo
  - Giga – 10^{9} or 1000 Mega
  - Tera – 10^{12} or 1000 Giga
  - Peta - 10^{15} or 1000 Tera
  - …

  Example:
  - The speed of my network card is 1 Gigabit per second
  - The speed of my Intel processor is 2.89 Gigahertz

Classes of Computers

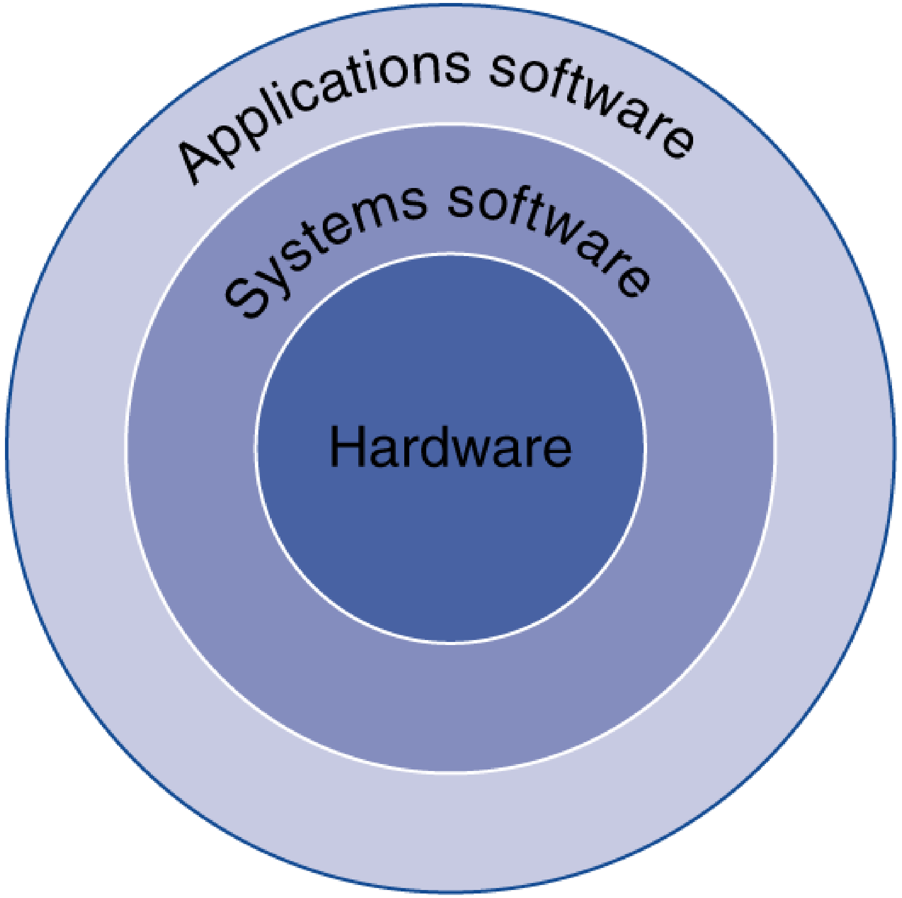
- **Personal computers**
  - General purpose, variety of software
  - Subject to cost/performance tradeoff
- **Server computers**
  - Network based
  - High capacity, performance, reliability
  - Range from small servers to building sized
- **Supercomputers**
  - High-end scientific and engineering calculations
  - Highest capability but represent a small fraction of the overall computer market
- **Embedded computers**
  - Hidden as components of systems
  - Stringent power/performance/cost constraints




IBM
z


Raspberry Pi 4 Model B
© Raspberry Pi 2018
GPIO
ETHERNET
Trxcom®
TRJG0926HENL
China M 1904
DISPLAY
RUN GLOBAL_EN
FCC ID: 2ABCB-RPI4B
IC: 20953-RPI4B
Made in the UK
HDMI
CAMERA
POWER IN
HDMI0
HDMI1

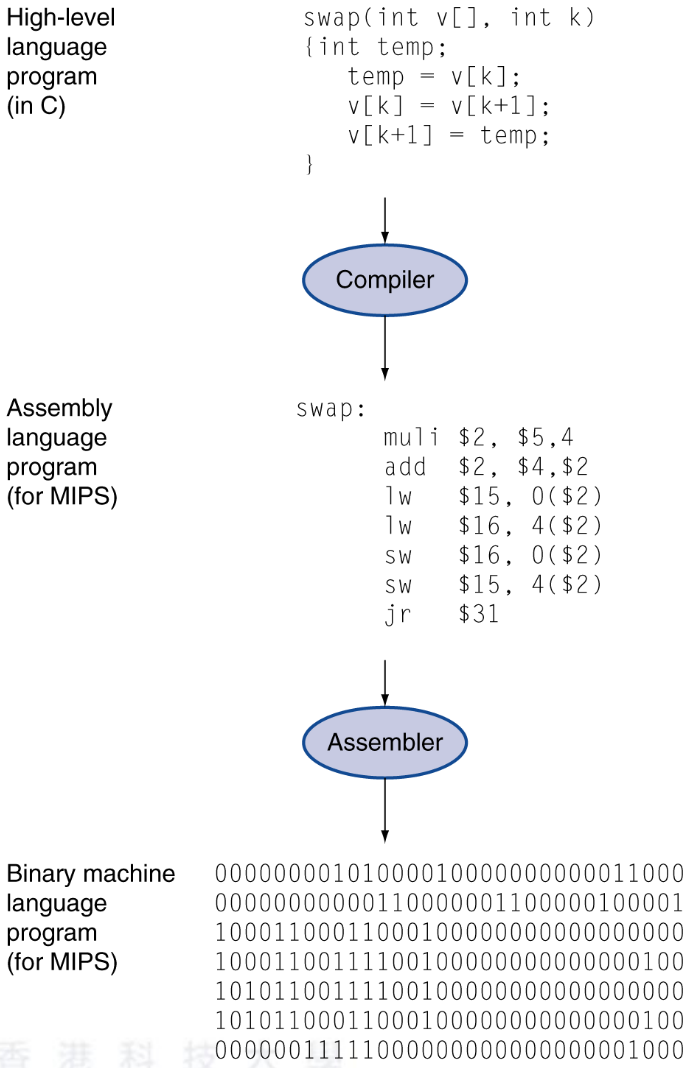
Below Your Program



- **Application software**
  - Written in high-level language
- **System software**
  - Compiler: translates HLL code to machine code
  - Operating System: service code
    - Handling input/output
    - Managing memory and storage
    - Scheduling tasks & sharing resources
- **Hardware**
  - Processor, memory, I/O controllers

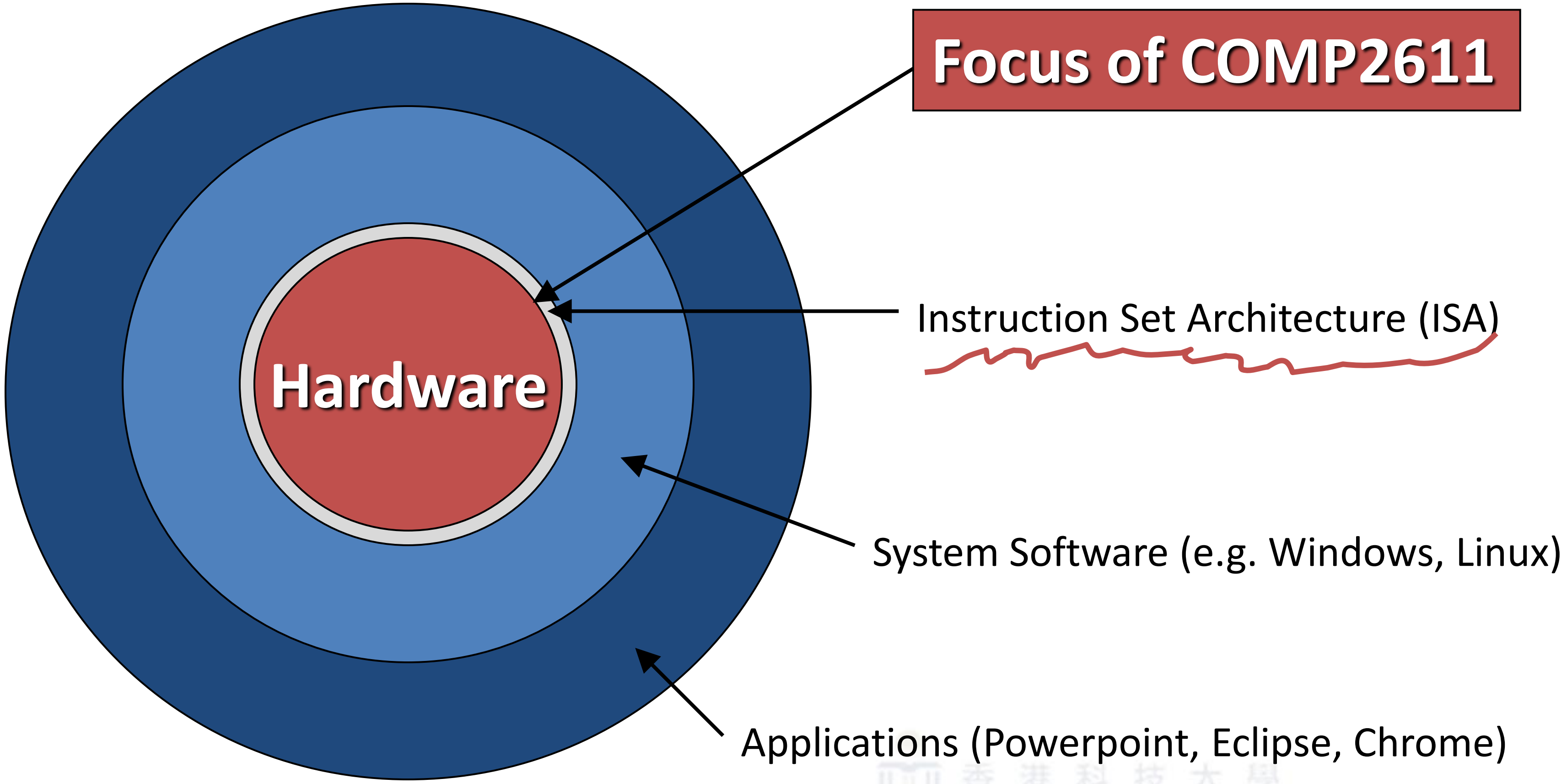
Levels of Program Code

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



Levels of Abstraction

Impossible to understand computer components by looking at every single transistor. Instead, abstraction is needed.



Levels of Abstraction (cont'd)

- **Key ideas:**
  - Both hardware and software are organized into **hierarchical layers**.
  - Hierarchical organization helps to cope with system **complexity**.
  - Lower-level details are **hidden** to offer a simpler view at the higher levels.
  - Interaction between levels occurs only through well-defined **interface**.
    - Interface between hardware and software: Instruction set architecture (ISA)

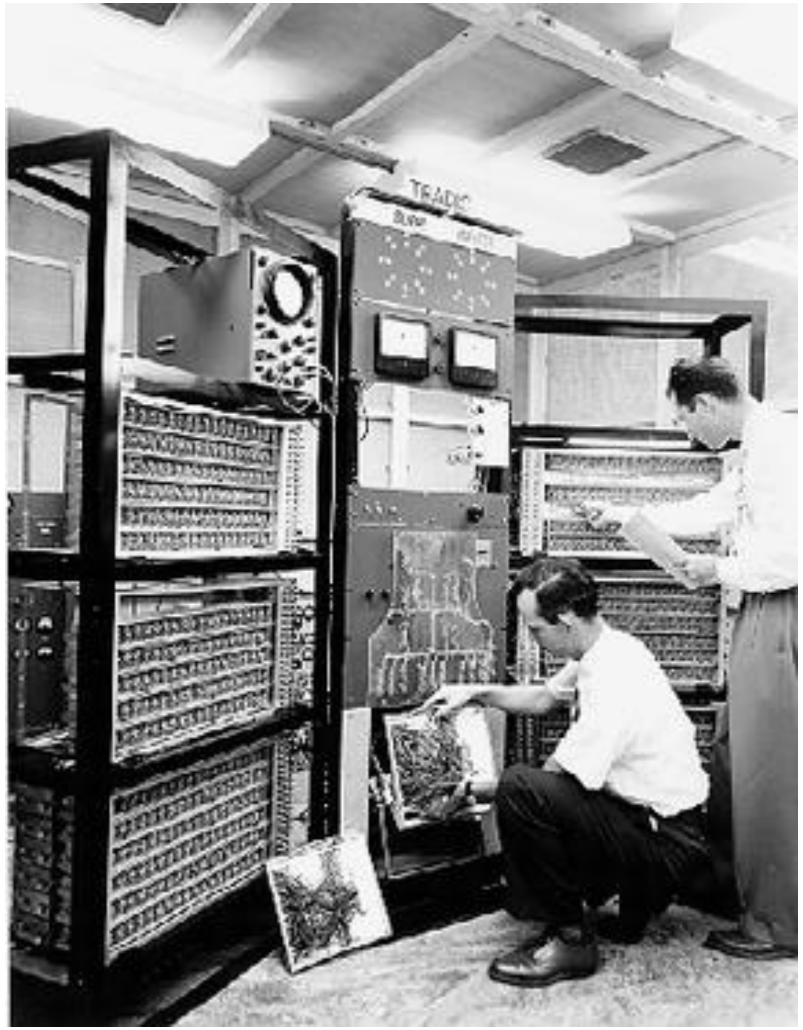
Instruction Set Architecture

An **instruction set architecture (ISA)** provides
an abstract interface between hardware and low-level software.

- **Advantage: allows different implementations of varying cost and performance to follow the same instruction set architecture (i.e., to run the same software).**
  - Example: 80x86, Pentium, Pentium II, Pentium III, Pentium 4 all implement the same ISA

- **Some instruction set architectures:**
  - Intel x86, ARM, MIPS, PowerPC, SPARC

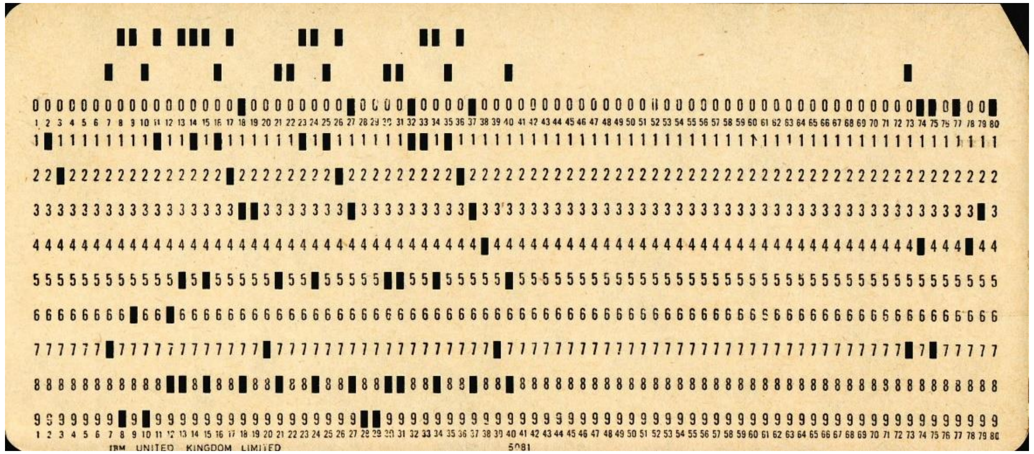
Computer in the Ooooooooooold Days



IBM model 350 disk file in 1950’s

Programmer in the Old Days

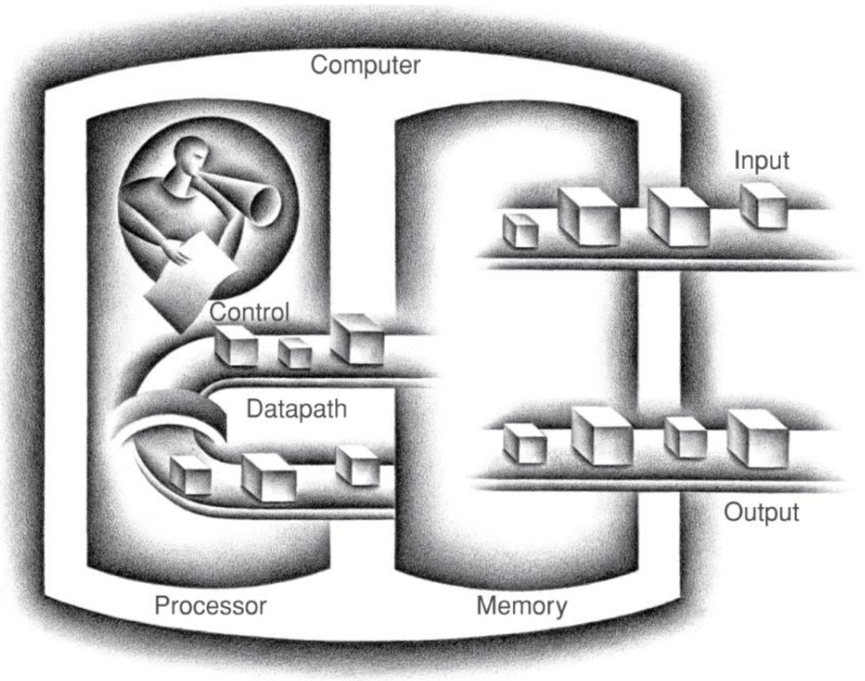
Programmers in 1970's with punch card



Components of Computer

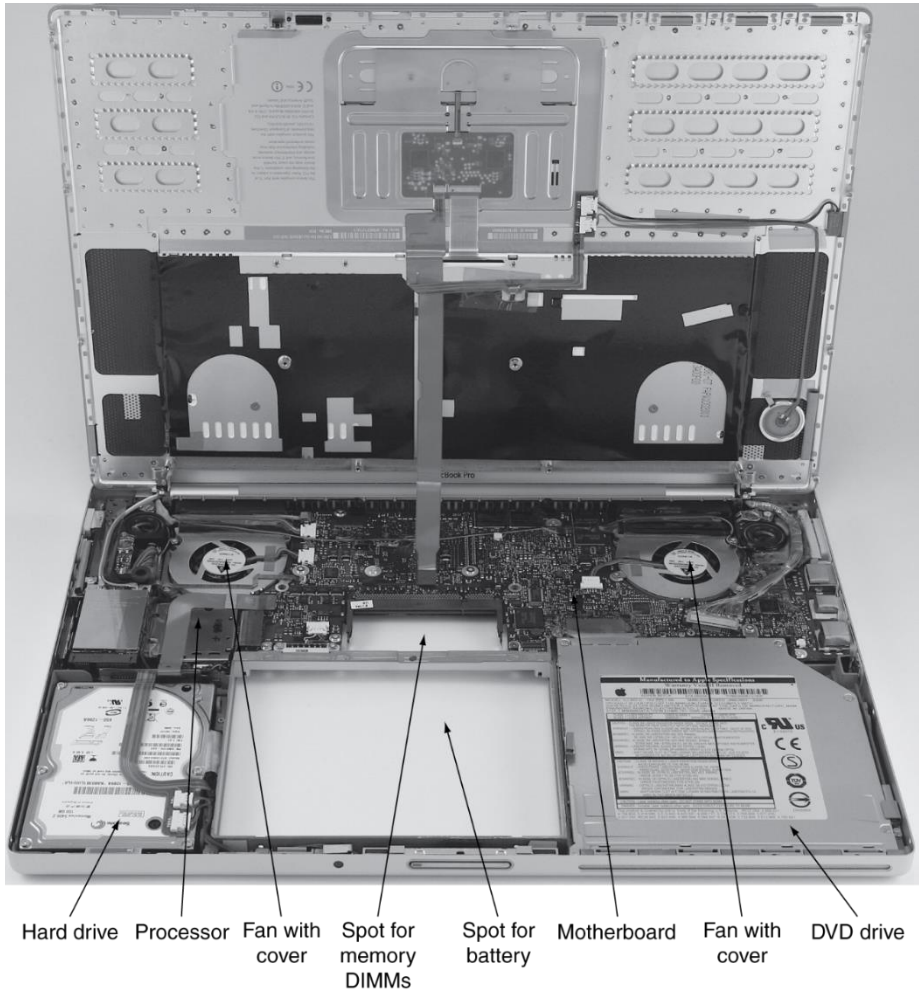
Five Basic Components (all kinds of computers)

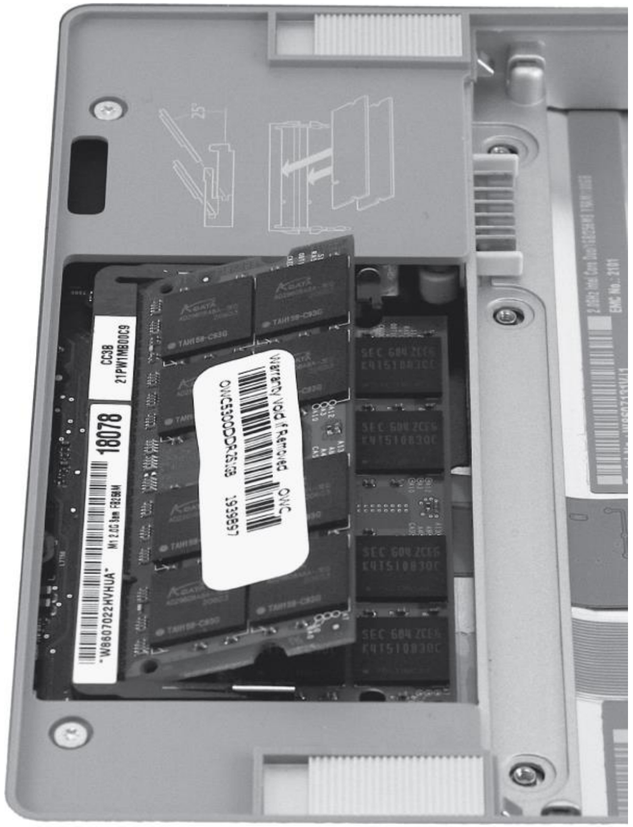
- **Input:**
  - To communicate with the computer
  - Data and instructions transferred to the memory
- **Output:**
  - To communicate with the user
  - Data is read from the memory
- **Memory:**
  - Large store to keep instructions and data
- **Processor, which consists of:**
  - **Datapath**: processes data according to instructions.
  - **Control**: commands the operations of input, output, memory, and datapath according to the instructions.



Anatomy of a Desktop Computer

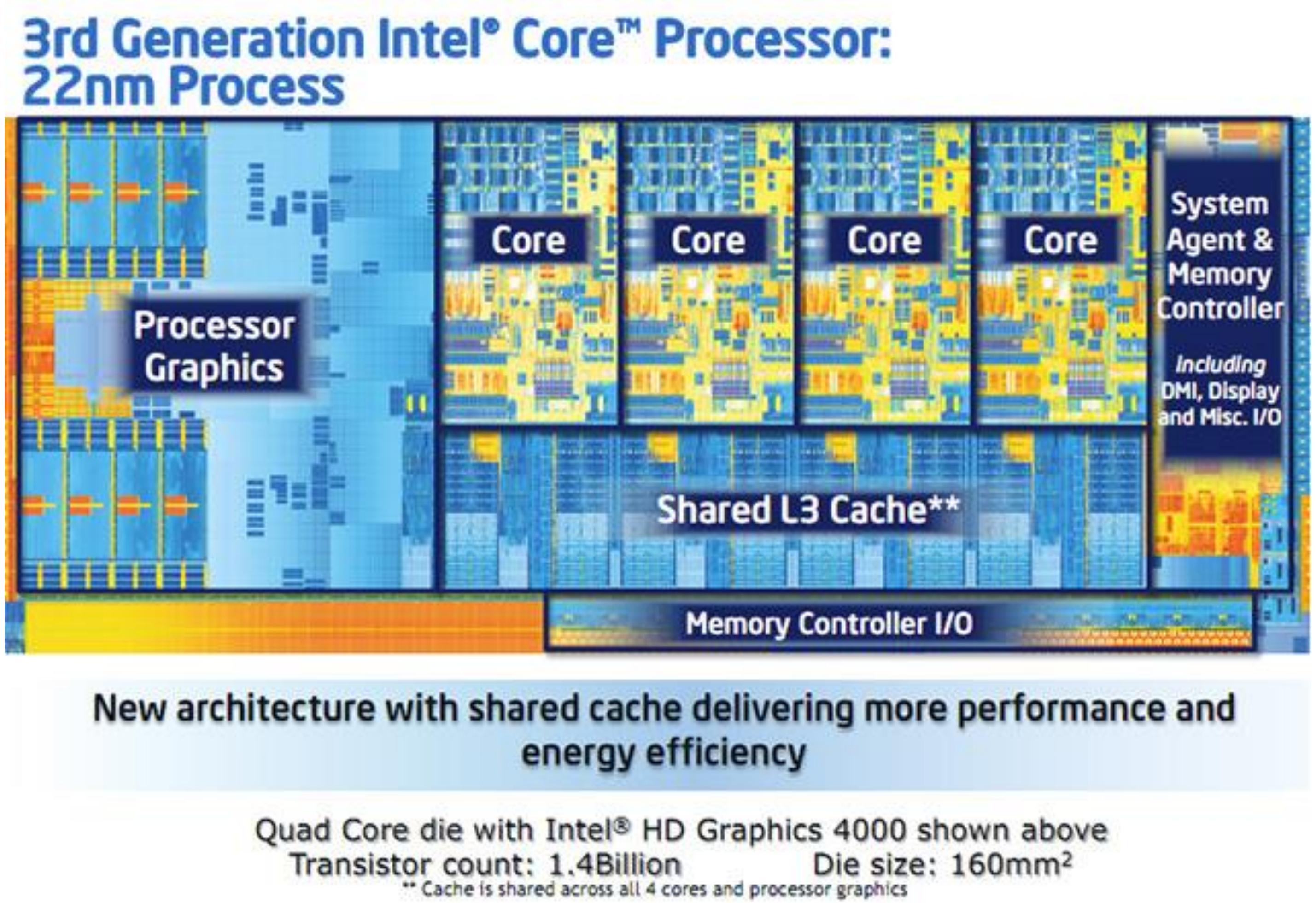
Anatomy of a Laptop Computer





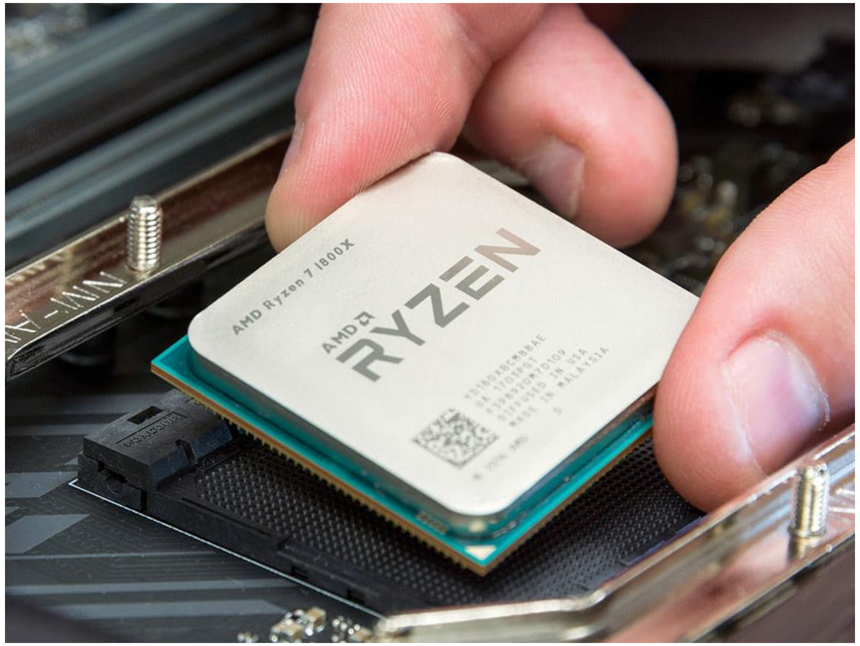
Anatomy of a Computer Processor

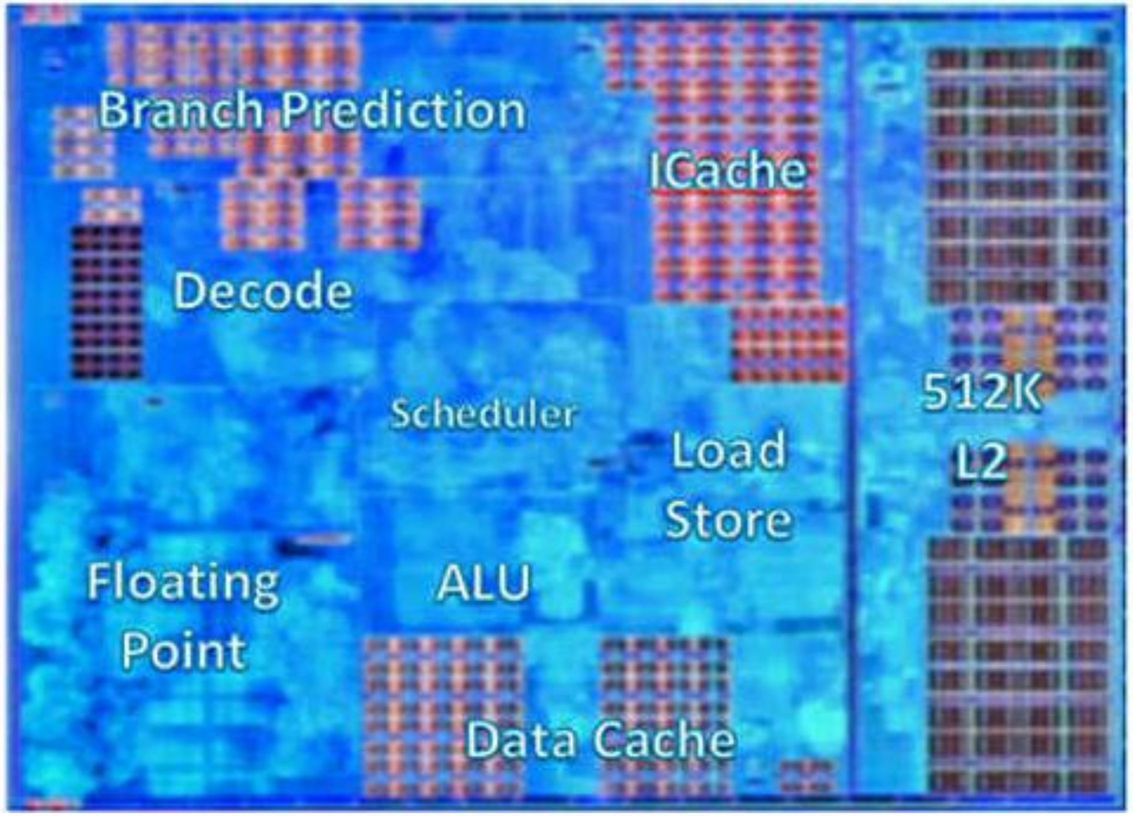
- **Intel core i7**



Anatomy of a Processor Core

- AMD Ryzen



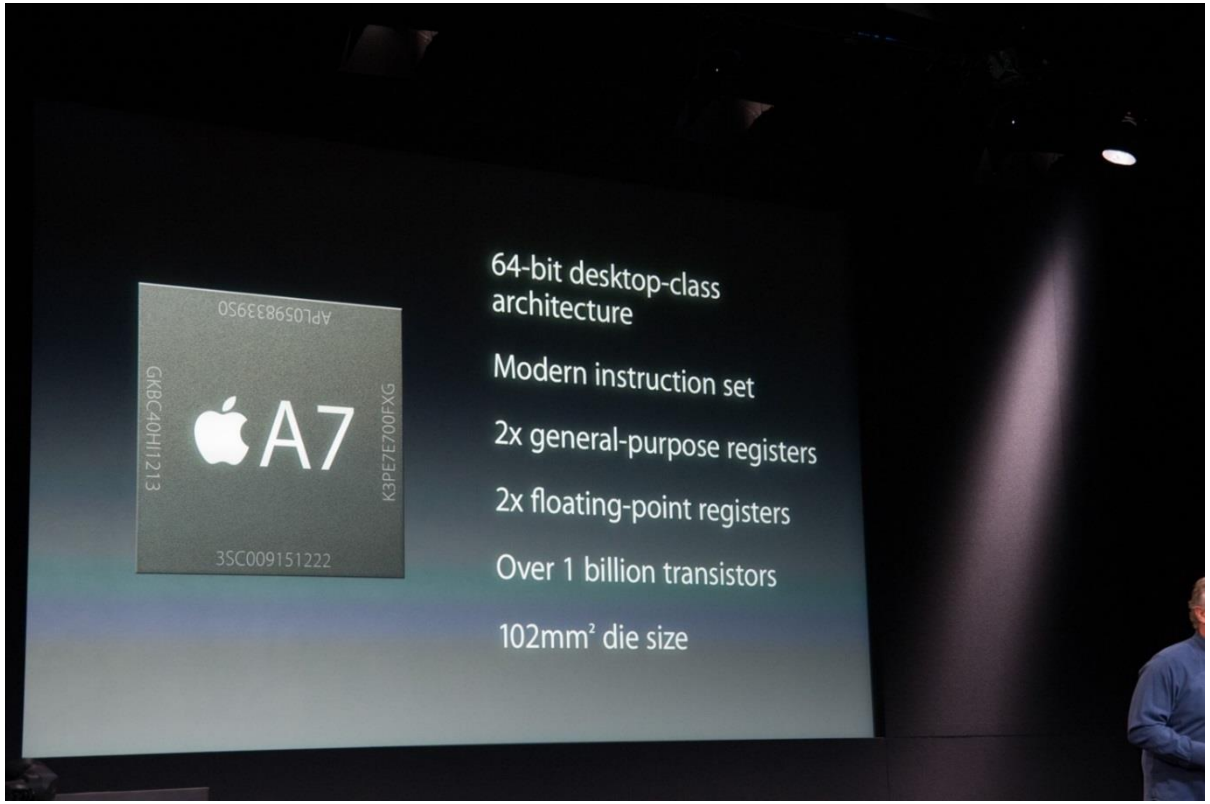


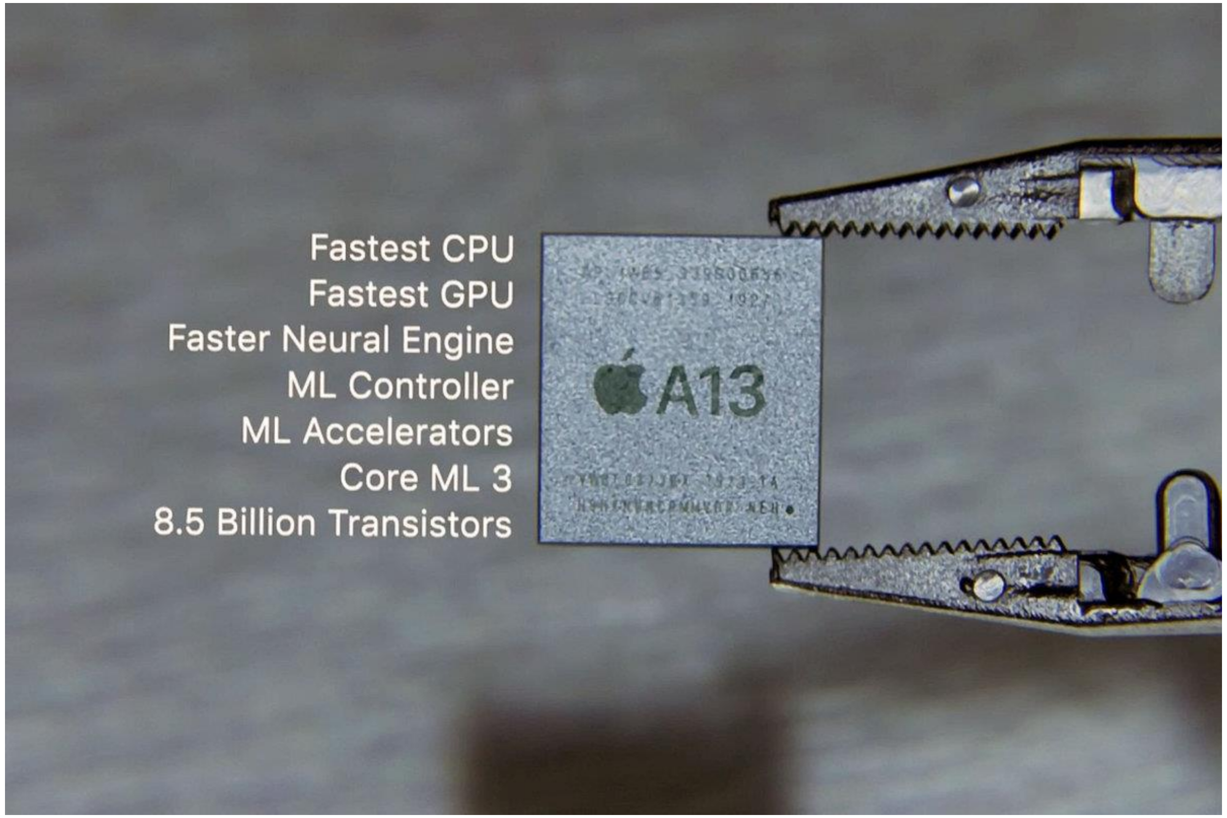
The Computer Revolution

Computers have led to a third revolution for civilization:
agricultural -> industrial -> information

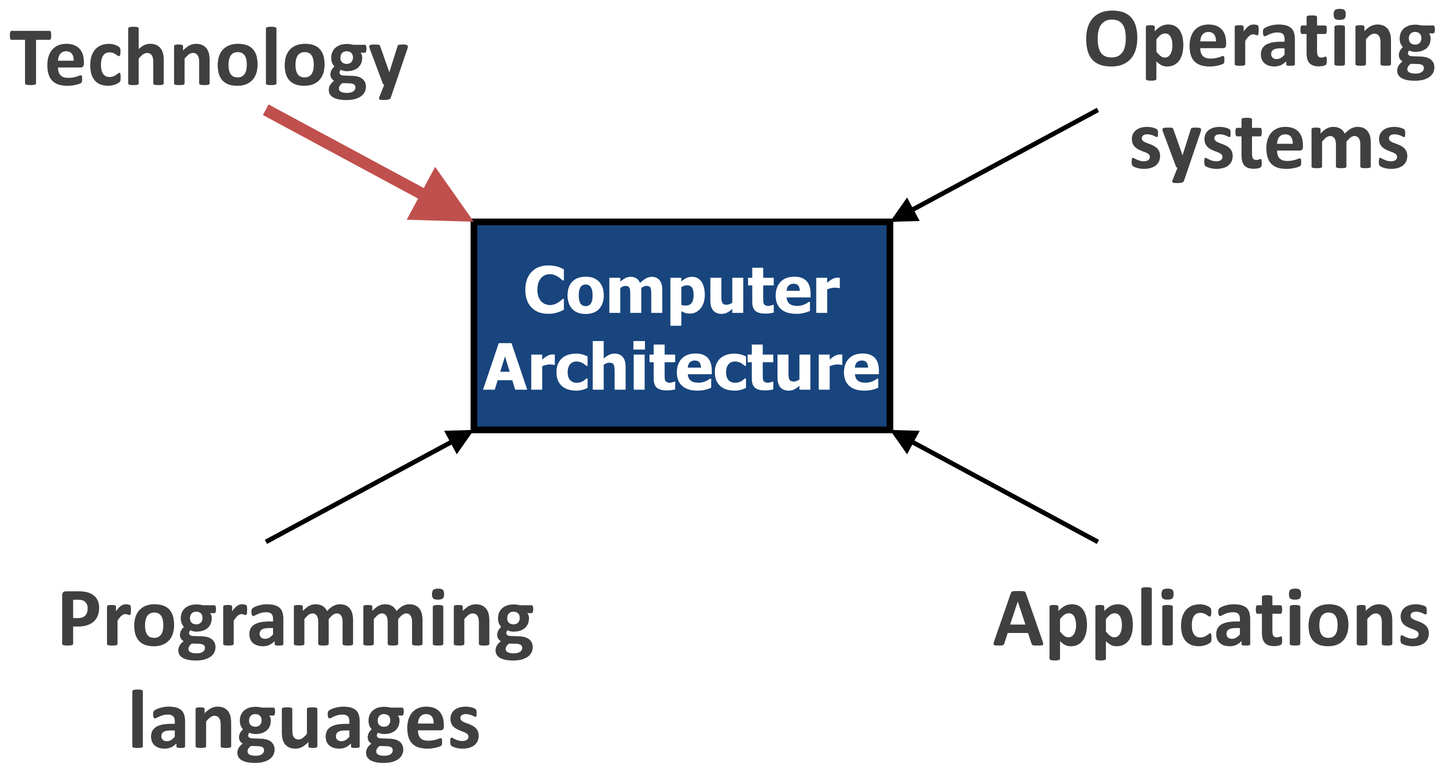
- **Progress in computer technology**
  - Underpinned by Moore's Law
- **Makes novel applications feasible**
  - Computers in automobiles
  - Artificial Intelligence
  - Human genome project
  - World Wide Web
  - Search Engines
- **Computers are pervasive**

What are these highlights?





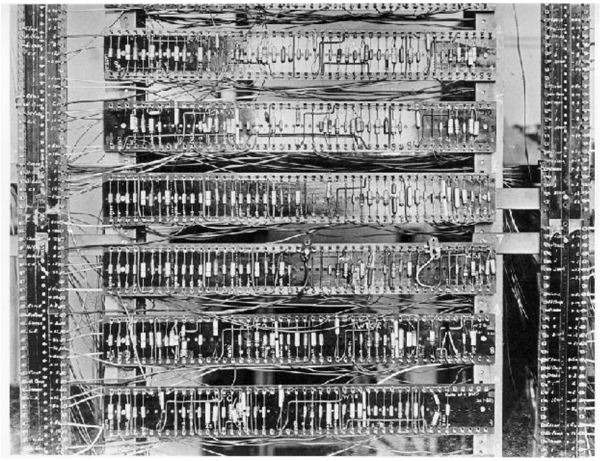
Rapidly Changing Forces on Computer Architecture



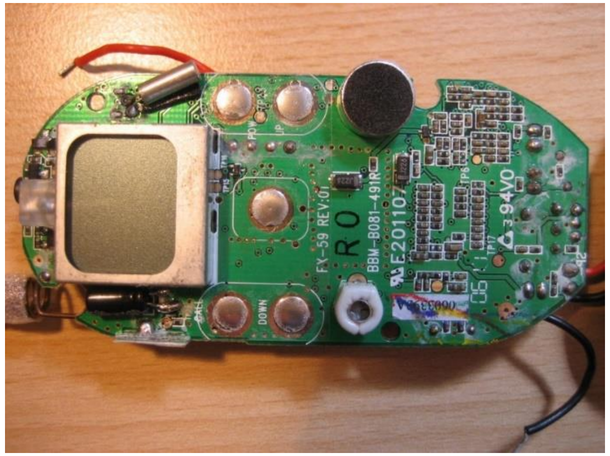
Technology Trend

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

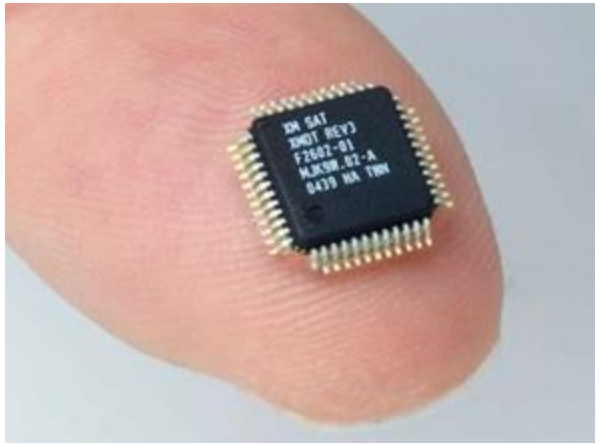
Vacuum Tubes (1950s)



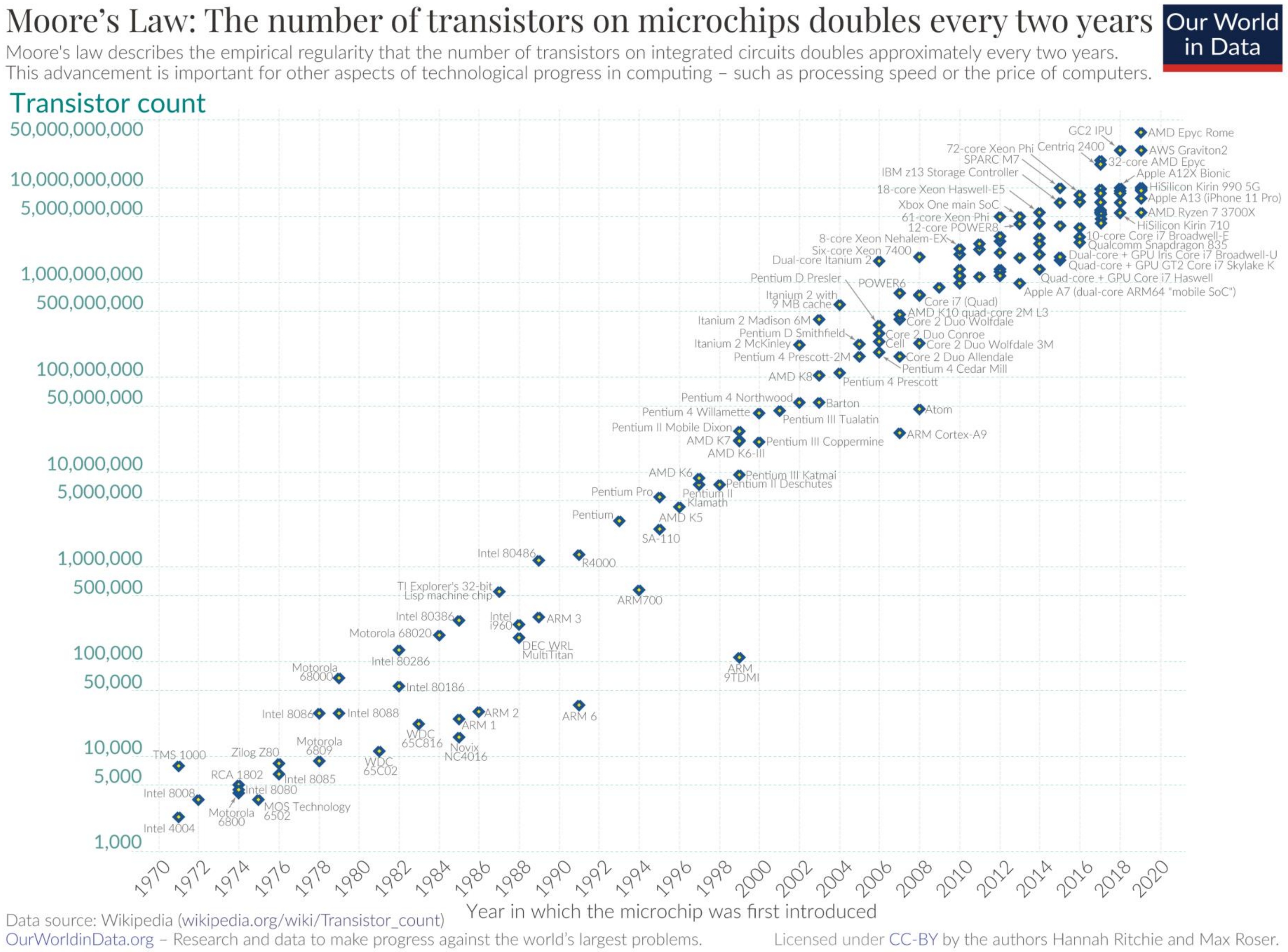
Transistors (1950s and 1960s)



Integrated Circuits (1960s and 70s)



Very Large Scale Integrated (VLSI) Circuit (1980s and on)



Moore’s Law: The number of transistors on microchips doubles every two years
Moore's law describes the empirical regularity that the number of transistors on integrated circuits doubles approximately every two years.
This advancement is important for other aspects of technological progress in computing – such as processing speed or the price of computers.
Our World in Data
Transistor count
50,000,000,000
10,000,000,000
5,000,000,000
1,000,000,000
500,000,000
100,000,000
50,000,000
10,000,000
5,000,000
1,000,000
500,000
100,000
50,000
10,000
5,000
1,000
GC2 IPU
AMD Epyc Rome
72-core Xeon Phi
Centriq 2400
AWS Graviton2
SPARC M7
32-core AMD Epyc
IBM z13 Storage Controller
Apple A12X Bionic
18-core Xeon Haswell-E5
HiSilicon Kirin 990 5G
Apple A13 (iPhone 11 Pro)
Xbox One main SoC
AMD Ryzen 7 3700X
61-core Xeon Phi
12-core POWER8
HiSilicon Kirin 710
8-core Xeon Nehalem-EX
10-core Core i7 Broadwell-E
Six-core Xeon 7400
Qualcomm Snapdragon 835
Dual-core Itanium 2
Dual-core + GPU Iris Core i7 Broadwell-U
Quad-core + GPU GT2 Core i7 Skylake K
Pentium D Presler
POWER6
Quad-core + GPU Core i7 Haswell
Itanium 2 with 9 MB cache
Apple A7 (dual-core ARM64 "mobile SoC")
Core i7 (Quad)
AMD K10 quad-core 2M L3
Itanium 2 Madison 6M
Core 2 Duo Wolfdale
Pentium D Smithfield
Core 2 Duo Conroe
Itanium 2 McKinley
Cell
Core 2 Duo Wolfdale 3M
Pentium 4 Prescott-2M
Core 2 Duo Allendale
Pentium 4 Cedar Mill
AMD K8
Pentium 4 Prescott
Pentium 4 Northwood
Barton
Pentium 4 Willamette
Atom
Pentium III Tualatin
Pentium II Mobile Dixon
ARM Cortex-A9
AMD K7
Pentium III Coppermine
AMD K6-III
AMD K6
Pentium III Katmai
Pentium II Deschutes
Pentium Pro
Pentium II Klamath
Pentium
AMD K5
SA-110
Intel 80486
R4000
TI Explorer's 32-bit Lisp machine chip
ARM700
Intel 80386
Intel i960
ARM 3
Motorola 68020
DEC WRL MultiTitan
Intel 80286
ARM 9TDMI
Motorola 68000
Intel 80186
Intel 8086
Intel 8088
ARM 2
ARM 1
ARM 6
WDC 65C816
Motorola 6809
Novix NC4016
TMS 1000
Zilog Z80
WDC 65C02
RCA 1802
Intel 8085
Intel 8008
Intel 8080
MOS Technology 6502
Motorola 6800
Intel 4004
1970
1972
1974
1976
1978
1980
1982
1984
1986
1988
1990
1992
1994
1996
1998
2000
2002
2004
2006
2008
2010
2012
2014
2016
2018
2020
Year in which the microchip was first introduced
Data source: Wikipedia (wikipedia.org/wiki/Transistor_count)
OurWorldinData.org – Research and data to make progress against the world’s largest problems.
Licensed under CC-BY by the authors Hannah Ritchie and Max Roser.

Concluding Remarks

- Recognize the five basic components of a computer
  - input, output, memory, processor (datapath + control)
- Understand the Principle of abstraction
  - Help cope with design complexity by hiding low level details
  - Levels of program code: high-level language, low-level language (e.g. assembly), machine code
- Instruction set architecture
  - The hardware/software interface
- Recognize the technology trend
  - Cost/performance is improving due to underlying technology development