

# **Course Introduction and Overview**

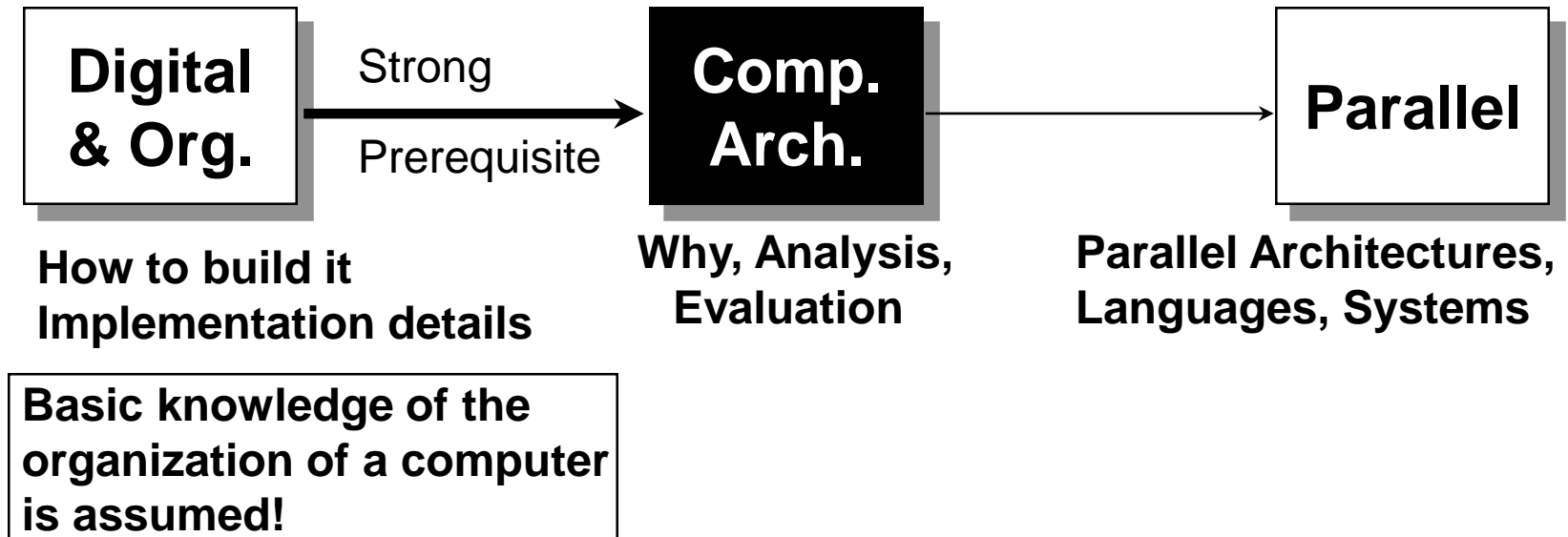
# Course Info

- Lecturer: **Aziz Qaroush**
- Email: **aqaroush@birzeit.edu**
- Office hours: **S,M,W 8 - 10 , S 12 - 2**
- Main Textbook (required)
  - Computer Organization and Design: The Hardware/Software Interface, [John L. Hennessy](#) and [David A. Patterson](#), Morgan Kaufmann, 2009 fourth edition
- **Supplement Text**
  - *Computer Architecture a Quantitative Approach 5<sup>rd</sup> Edition*, John L. Hennessy, David A. Patterson, Morgan Kaufmann 2012.
  - Modern Processor Design, John Paul Shen and Mikko H. Lipasti, McGraw-Hill, 2005.
- **Grading**
  - 15% First Exam
  - 20% Second Exam
  - 35% Final Exam
  - 30% Projects

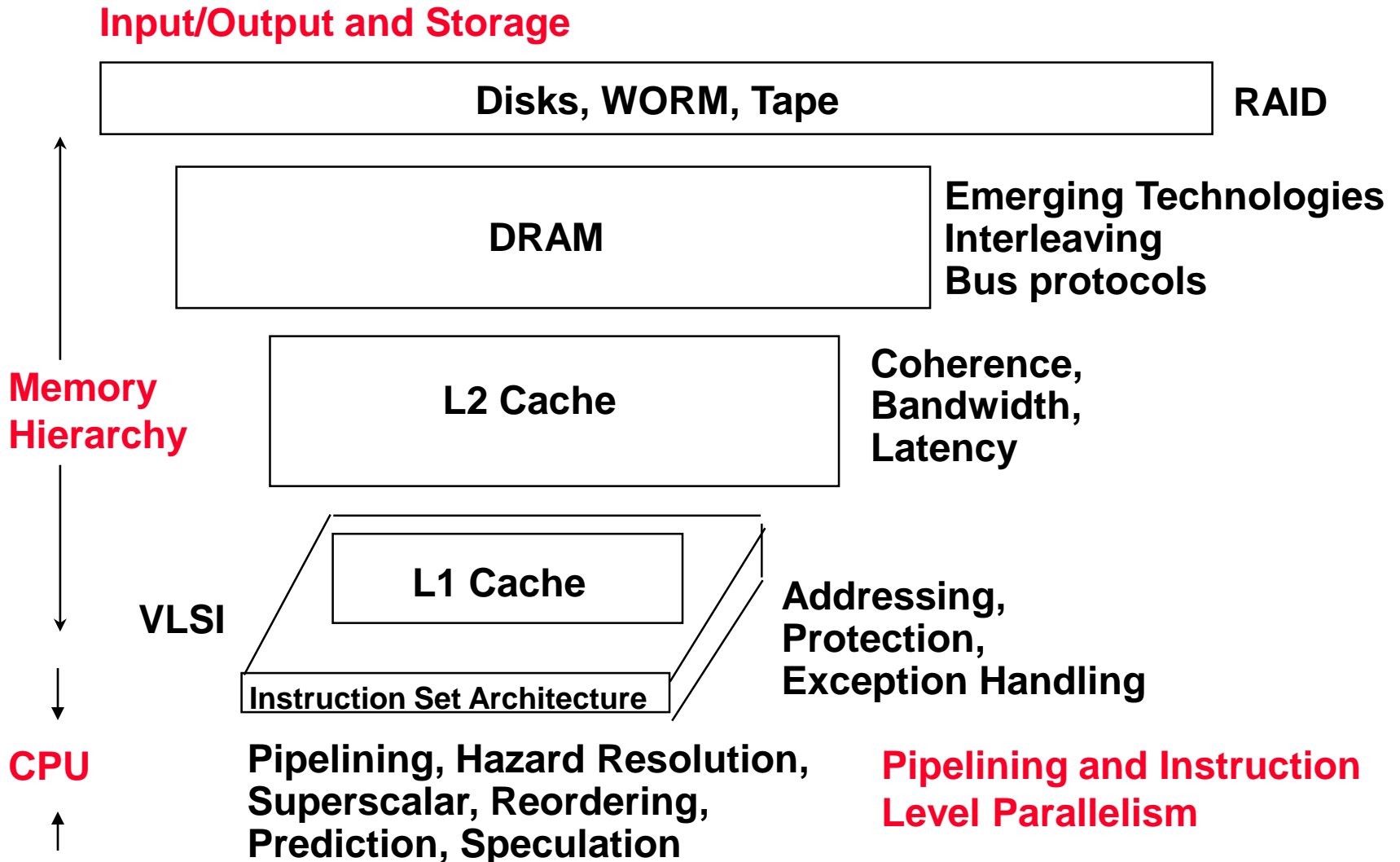
# Why Study Computer Architecture?

- **You want to be called “Computer Engineer or Scientist”**
- **You want to become an “expert” on computer hardware**
- **You want to become a “computer system designer”**
- **You want to become a “software designer” and need to understand how to improve code performance**
- **Technology is improving rapidly  $\Rightarrow$  new opportunities  
Has never been more exciting!**
- **Impacts Electrical Engineering and Computer Science**

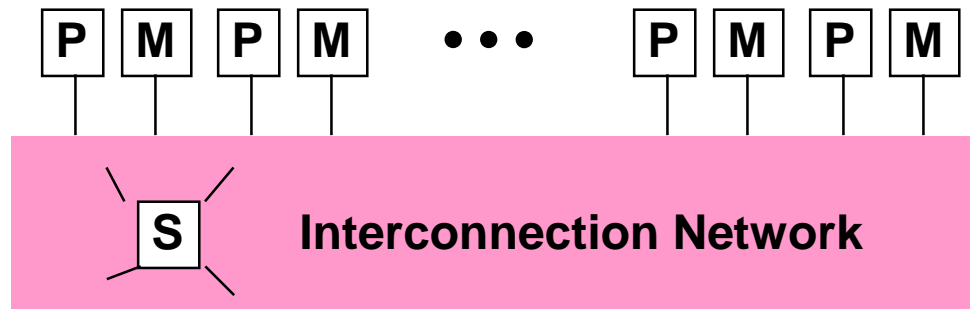
# Related Courses



# Computer Architecture Topics



# Computer Architecture Topics



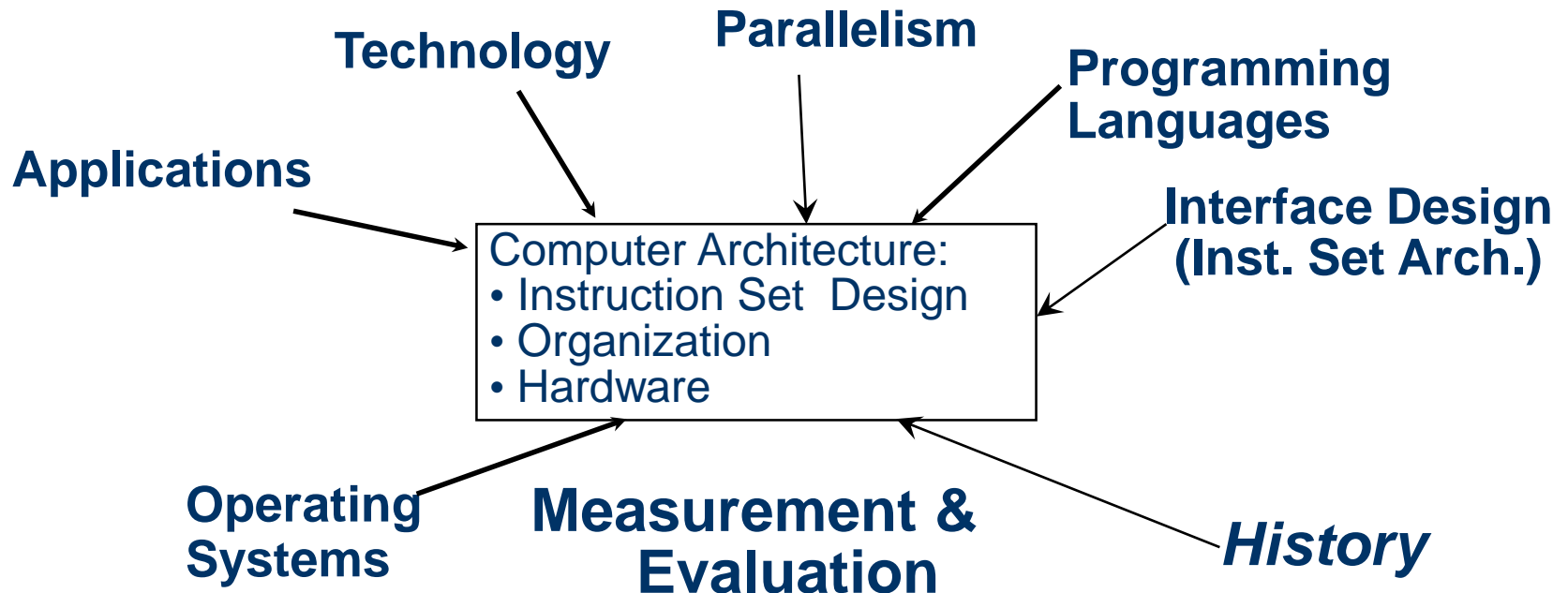
**Processor-Memory-Switch**

**Multiprocessors  
Networks and Interconnections**

**Topologies,  
Routing,  
Bandwidth,  
Latency,  
Reliability**

# Course Focus

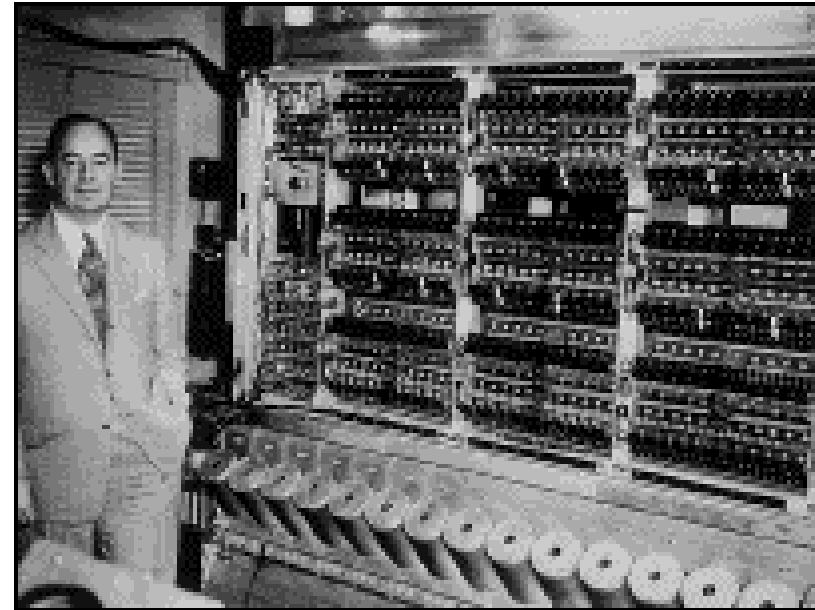
- To Understand the design techniques, machine structures, technology factors, evaluation methods that will determine the form of computers in 21st Century
- Role of a computer architect:
  - To design and engineer the various levels of a computer system to maximize performance and programmability within limits of technology and cost



# History

**Von Neumann:  
Invented EDSAC (1949).  
First Stored Program  
Computer. Uses Memory.**

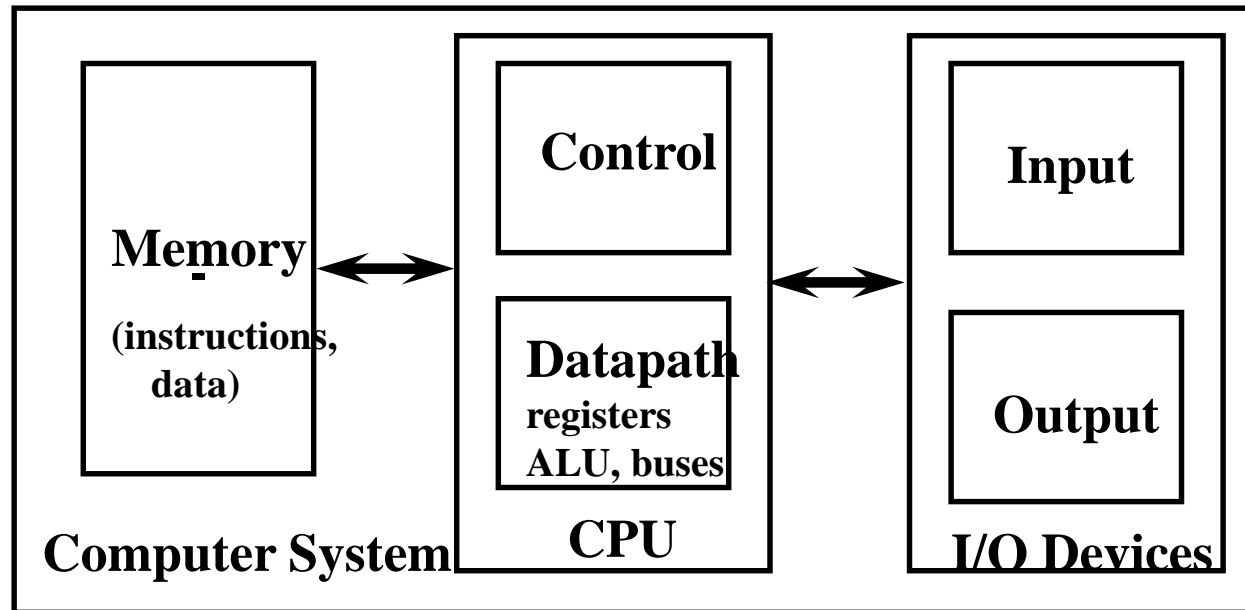
**Importance: We are still using  
The same basic design.**





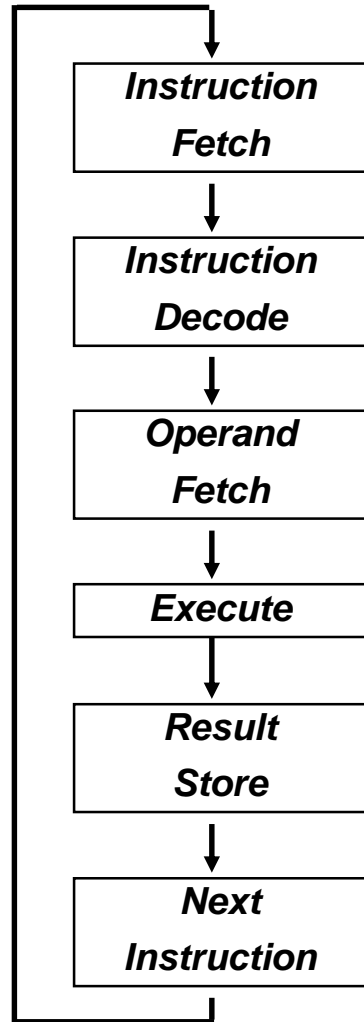
# The Von Neumann Computer Model

- **Partitioning of the computing engine into components:**
  - Central Processing Unit (CPU): **Control Unit (instruction decode , sequencing of operations), Datapath (registers, arithmetic and logic unit, buses).**
  - Memory: **Instruction and operand storage.**
  - Input/Output (I/O) sub-system: **I/O bus, interfaces, devices.**
  - The stored program concept: **Instructions from an instruction set are fetched from a common memory and executed one at a time**



**Major CPU Performance Limitation:** The Von Neumann computing model implies sequential execution one instruction at a time

# Generic CPU Machine Instruction Execution Steps



**(Implied by The Von Neumann Computer Model)**

**Obtain instruction from program storage**

**Determine required actions and instruction size**

**Locate and obtain operand data**

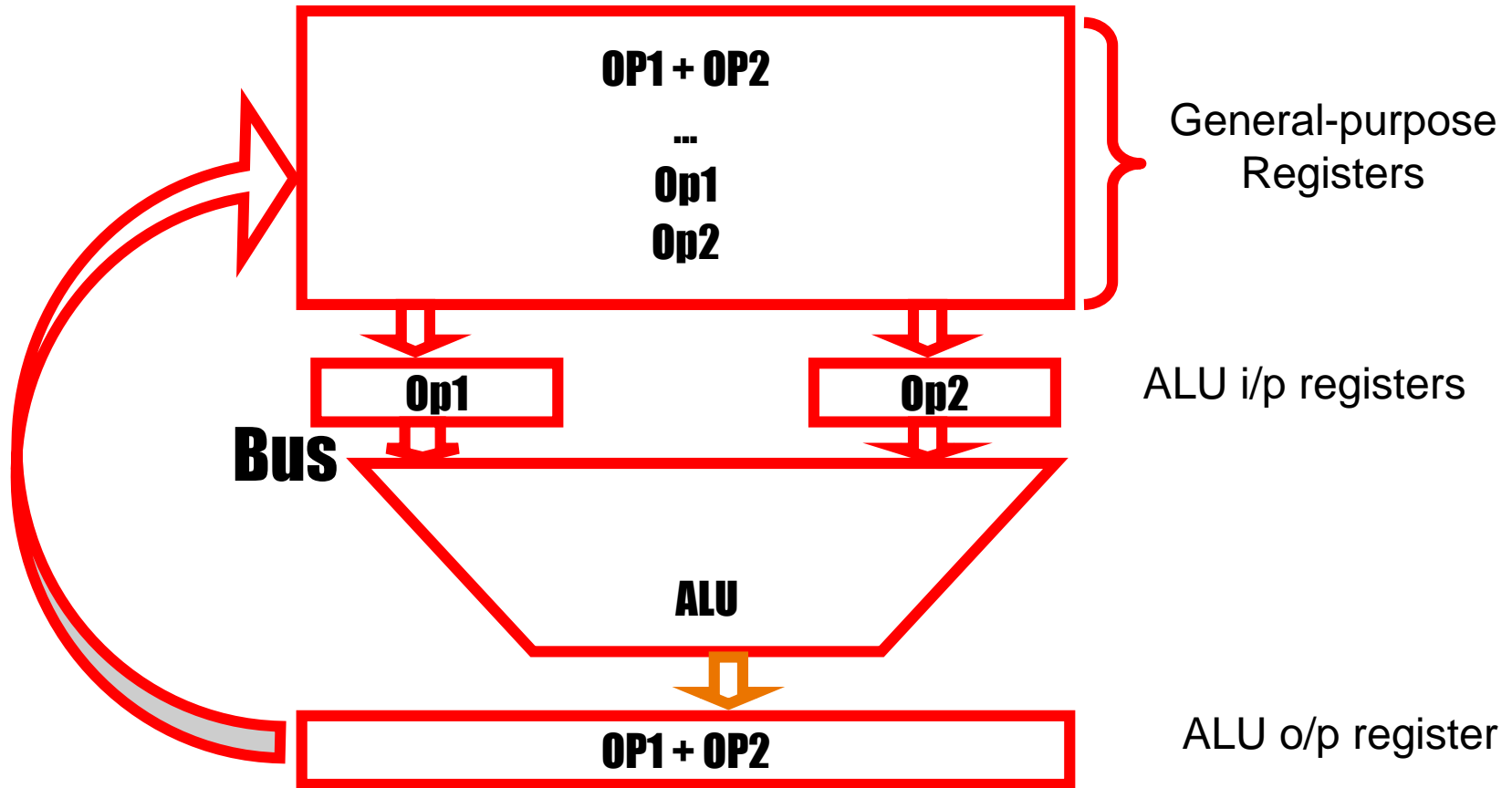
**Compute result value or status**

**Deposit results in storage for later use**

**Determine successor or next instruction**

# Computer Components

- Datapath of a von Newman machine



# General Purpose Processor/Computer System Generations

Classified according to implementation technology:

- **The First Generation**, 1946-59: Vacuum Tubes, Relays, Mercury Delay Lines:
  - ENIAC (Electronic Numerical Integrator and Computer): First electronic computer, 18000 vacuum tubes, 1500 relays, 5000 additions/sec (1944).
  - First stored program computer: EDSAC (Electronic Delay Storage Automatic Calculator), 1949.
- **The Second Generation**, 1959-64: Discrete Transistors.
  - e.g. IBM Main frames
- **The Third Generation**, 1964-75: Small and Medium-Scale Integrated (MSI) Circuits.
  - e.g. Main frames (IBM 360) , mini computers (DEC PDP-8, PDP-11).
- **The Fourth Generation**, 1975-Present: The Microcomputer. VLSI-based Microprocessors (single-chip processor)
  - First microprocessor: Intel's 4-bit 4004 (2300 transistors), 1970.
  - Personal Computer (PCs), laptops, PDAs, servers, clusters ...
  - Reduced Instruction Set Computer (RISC) 1984

**Common factor among all generations:**

**All target the The Von Neumann Computer Model or paradigm**

# What is “Computer Architecture” ?

- **Computer Architecture =**
  - Instruction Set Architecture +**
  - Computer Organization**
- **Instruction Set Architecture (ISA)**
  - **WHAT** the computer does (logical view)
- **Computer Organization**
  - **HOW** the ISA is implemented (physical view)
- **We will study both in this course**

# Instruction Set Architecture (ISA)

- Is a subset of Computer Architecture
- Definition by Amdahl, Blaaw, and Brooks – 1964

*“... 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.”*

- An ISA encompasses ...
  - Instructions and Instruction Formats
  - Data Types, Encodings, and Representations
  - Programmable Storage: Registers and Memory
  - Addressing Modes: Accessing Instructions and Data
  - Handling Exceptional Conditions

# Instruction Set Architecture – cont'd

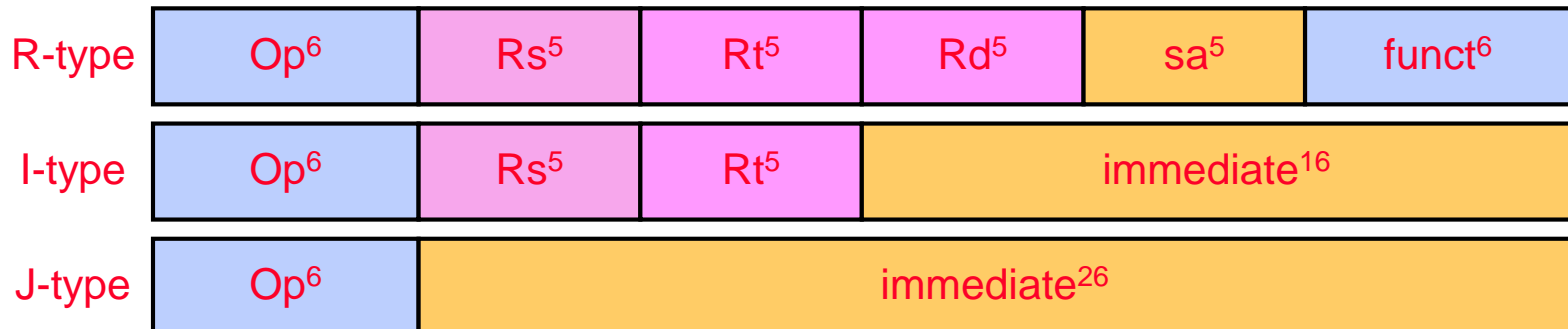
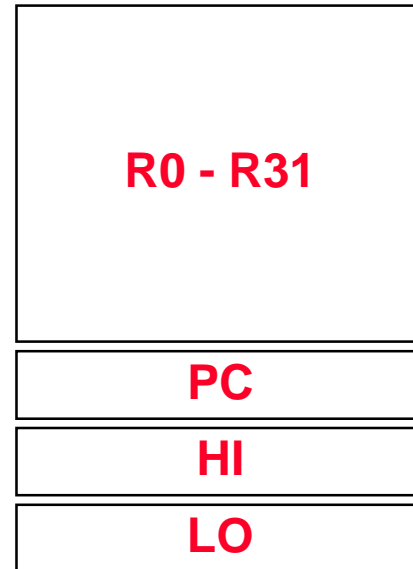
- **Critical interface between hardware and software**
  - Standardizes instructions, machine language bit patterns, etc.
  - Advantage: **different implementations of the same architecture**
  - Disadvantage: **sometimes prevents using new innovations**

<b>Examples</b>	<b>(versions)</b>	<b>Introduced in</b>
– Intel	(8086, 80386, Pentium, ...)	1978
– IBM Power	(Power 2, 3, 4, 5)	1985
– HP PA-RISC	(v1.1, v2.0)	1986
– MIPS	(MIPS I, II, III, IV, V)	1986
– Sun Sparc	(v8, v9)	1987
– Digital Alpha	(v1, v3)	1992
– PowerPC	(601, 604, ...)	1993

# Overview of the MIPS ISA

- All instructions are 32-bit wide
- Instruction Categories
  - Load/Store
  - Integer Arithmetic
  - Jump and Branch
  - Floating Point
  - Memory Management
- Three Instruction Formats

## Registers

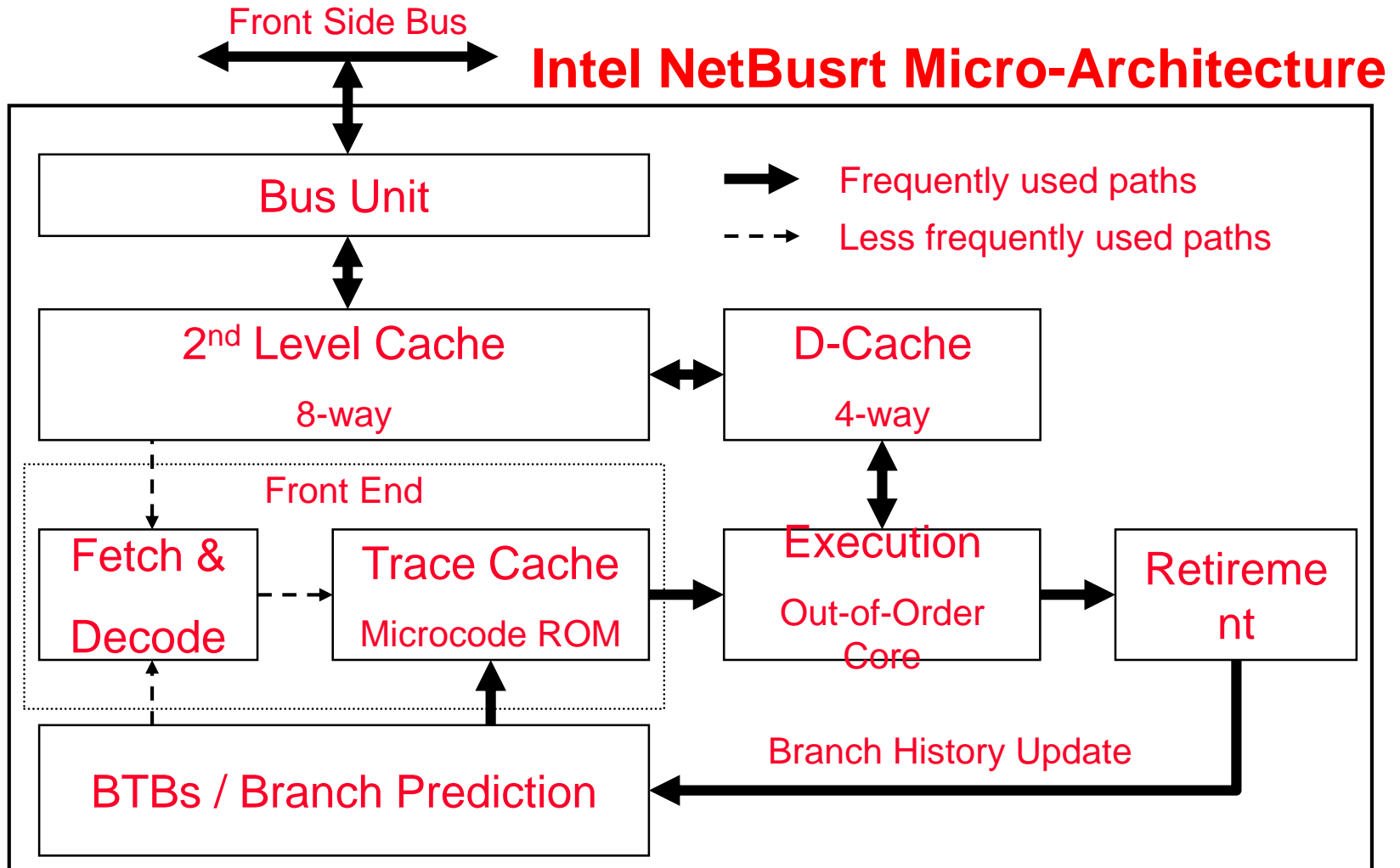




# Computer Organization

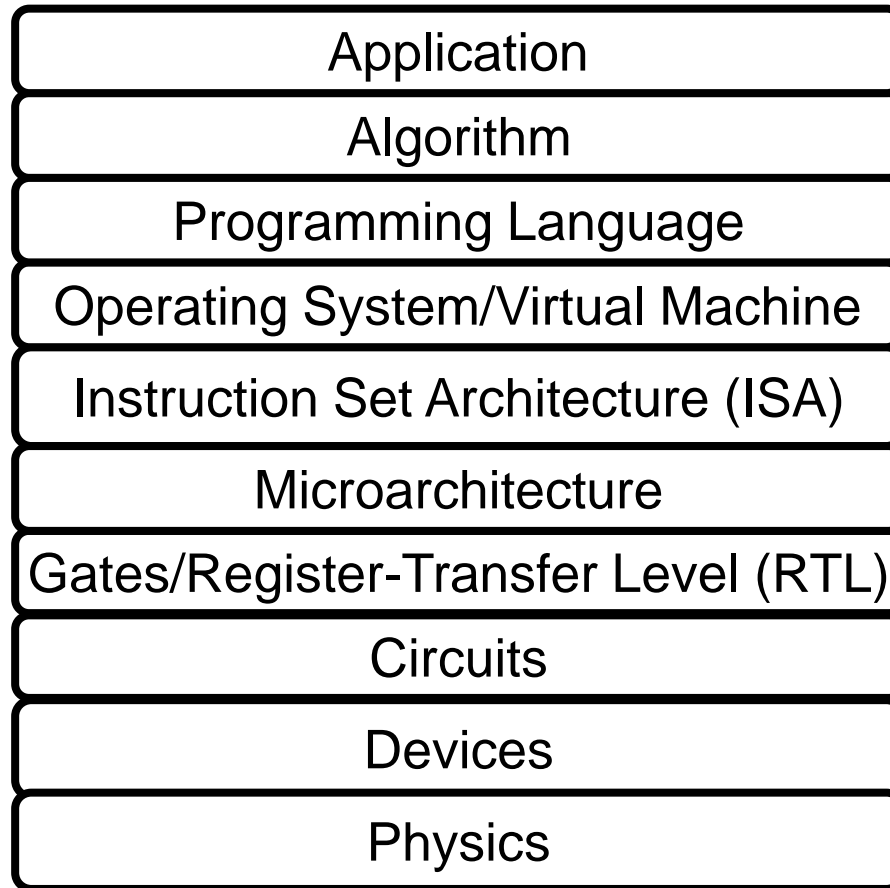
- **Realization of the Instruction Set Architecture**
- **Characteristics of principal components**
  - Registers, ALUs, FPUs, Caches, ...
- **Ways in which these components are interconnected**
- **Information flow between components**
- **Means by which such information flow is controlled**
- **Register Transfer Level (RTL) description**

# Microprocessor Organization

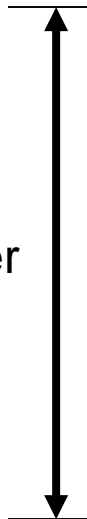




# Abstraction Layers in Modern Systems



Original domain of the computer architect ('50s-'80s)



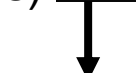
Domain of recent computer architecture ('90s)



**Parallel computing, security, ...**



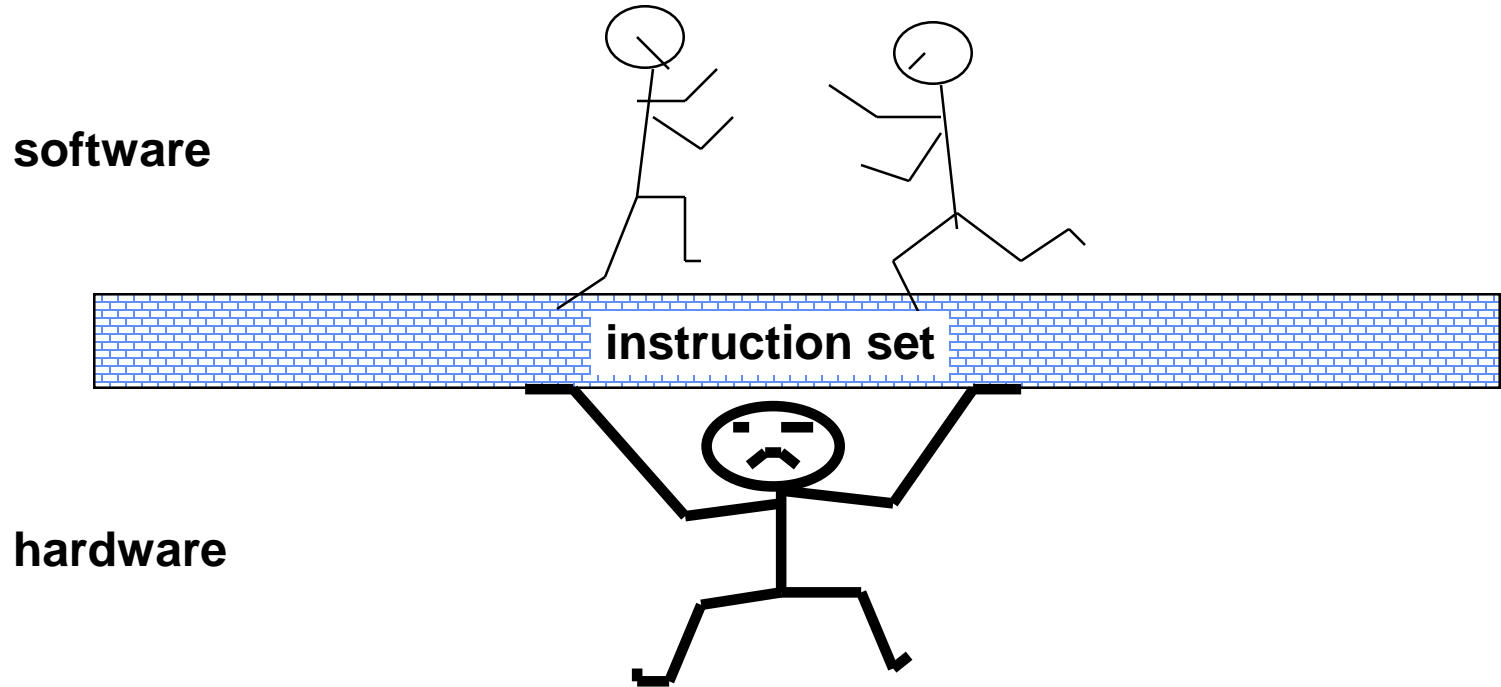
Reliability, power, ...



Reinvigoration of computer architecture, mid-2000s onward.

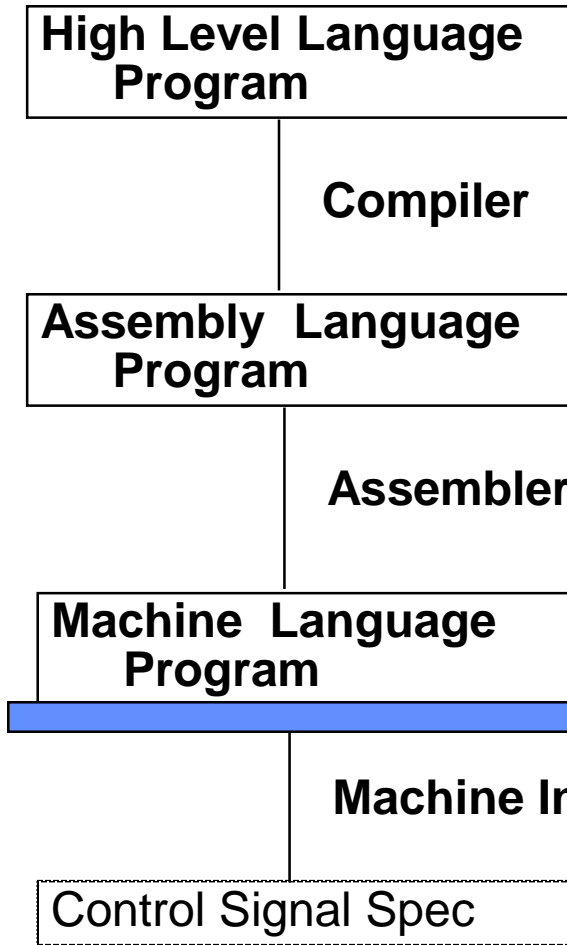


# Instruction Set Architecture: Critical Interface



- **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

# How to Speak Computer



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

```
lw $15, 0($2)  
lw $16, 4($2)  
sw $16, 0($2)  
sw $15, 4($2)
```

```
10001100011000100000000000000000  
10001100111100100000000000000100  
10101100111100100000000000000000  
10101100011000100000000000000100
```

```
ALUOP[0:3] <= InstReg[9:11] & MASK
```

**Need translation from application to physics**

# Computer Architecture's Changing Definition

- **1950s to 1960s: Computer Architecture Course: Computer Arithmetic**
- **1970s to mid 1980s: Computer Architecture Course: Instruction Set Design, especially ISA appropriate for compilers**
- **1990s: Computer Architecture Course: Design of CPU, memory system, I/O system, Multiprocessors, Networks**
- **2010s: Computer Architecture Course: Self adapting systems? Self organizing structures? DNA Systems/Quantum Computing?**

# Crossroads: Conventional Wisdom in Comp. Arch

- Old Conventional Wisdom: Power is free, Transistors expensive
  - New Conventional Wisdom: “Power wall” Power expensive, Xtors free  
(Can put more on chip than can afford to turn on)
  - Old CW: Sufficiently increasing Instruction Level Parallelism via compilers, innovation (Out-of-order, speculation, VLIW, ...)
  - New CW: “ILP wall” law of diminishing returns on more HW for ILP
  - Old CW: Multiplies are slow, Memory access is fast
  - New CW: “Memory wall” Memory slow, multiplies fast  
(200 clock cycles to DRAM memory, 4 clocks for multiply)
  - Old CW: Uniprocessor performance 2X / 1.5 yrs
  - New CW: Power Wall + ILP Wall + Memory Wall = **Brick Wall**
    - Uniprocessor performance now 2X / 5(?) yrs
- ⇒ **Sea change in chip design: multiple “cores”**  
(2X processors per chip / ~ 2 years)
- » More simpler processors are more power efficient

# Technology Change

- **Technology changes rapidly**
  - **HW**
    - » **Vacuum tubes: Electron emitting devices**
    - » **Transistors: On-off switches controlled by electricity**
    - » **Integrated Circuits( IC/ Chips): Combines thousands of transistors**
    - » **Very Large-Scale Integration( VLSI): Combines millions of transistors**
    - » **What next?**
  - **SW**
    - » **Machine language: Zeros and ones**
    - » **Assembly language: Mnemonics**
    - » **High-Level Languages: English-like**
    - » **Artificial Intelligence languages: Functions & logic predicates**
    - » **Object-Oriented Programming: Objects & operations on objects**



# Technology Improvements

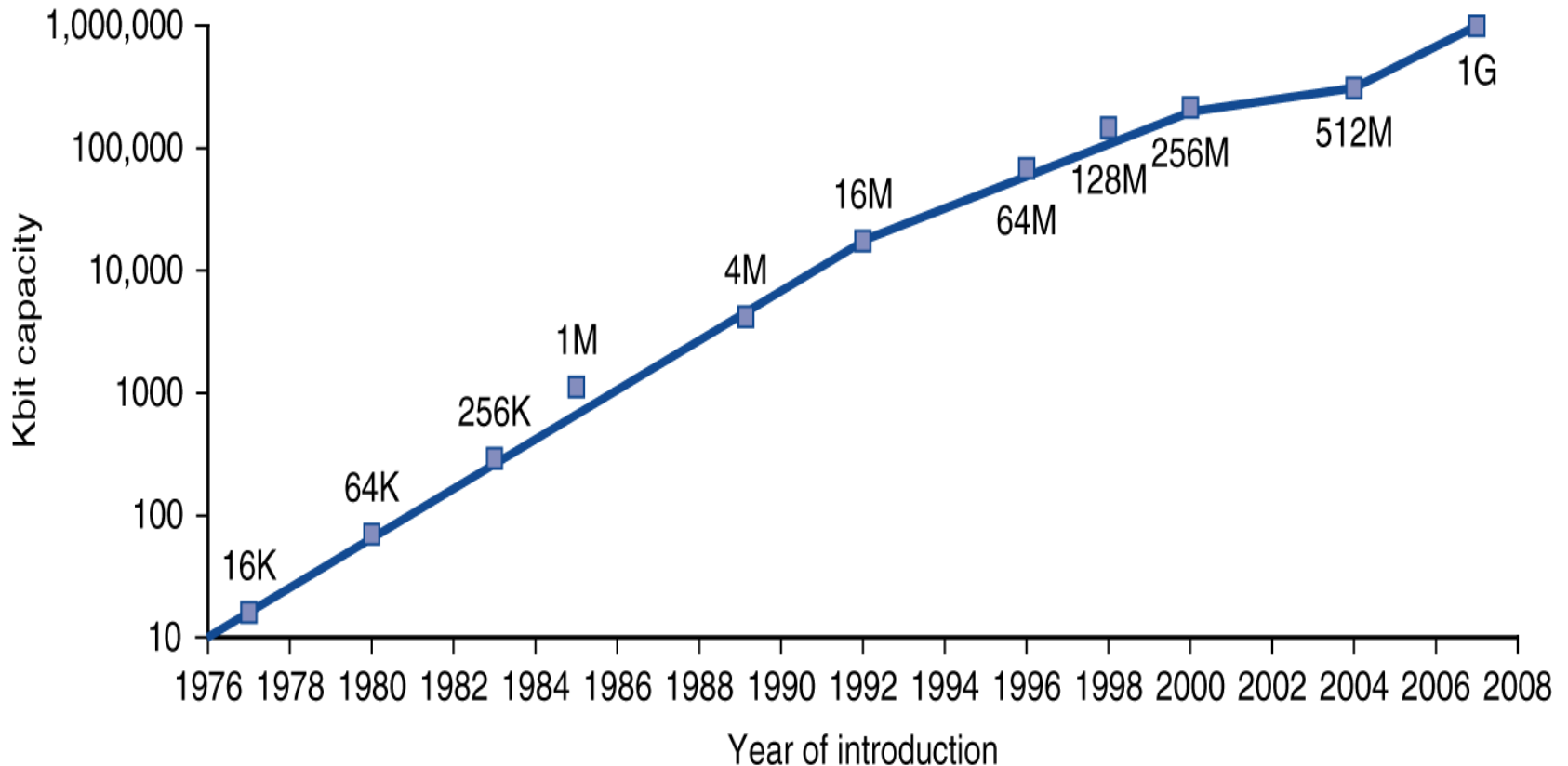
Year	Technology	Relative performance/cost
1951	Vacuum tube	1
1965	Transistor	35
1975	Integrated circuit (IC)	900
1995	Very large scale IC (VLSI)	2,400,000
2005	Ultra large scale IC	6,200,000,000

- ❖ Processor transistor count: about 30% to 40% per year
- ❖ Memory capacity: about 60% per year (4x every 3 years)
- ❖ Disk capacity: about 60% per year
- ❖ Opportunities for new applications
- ❖ Better organizations and designs

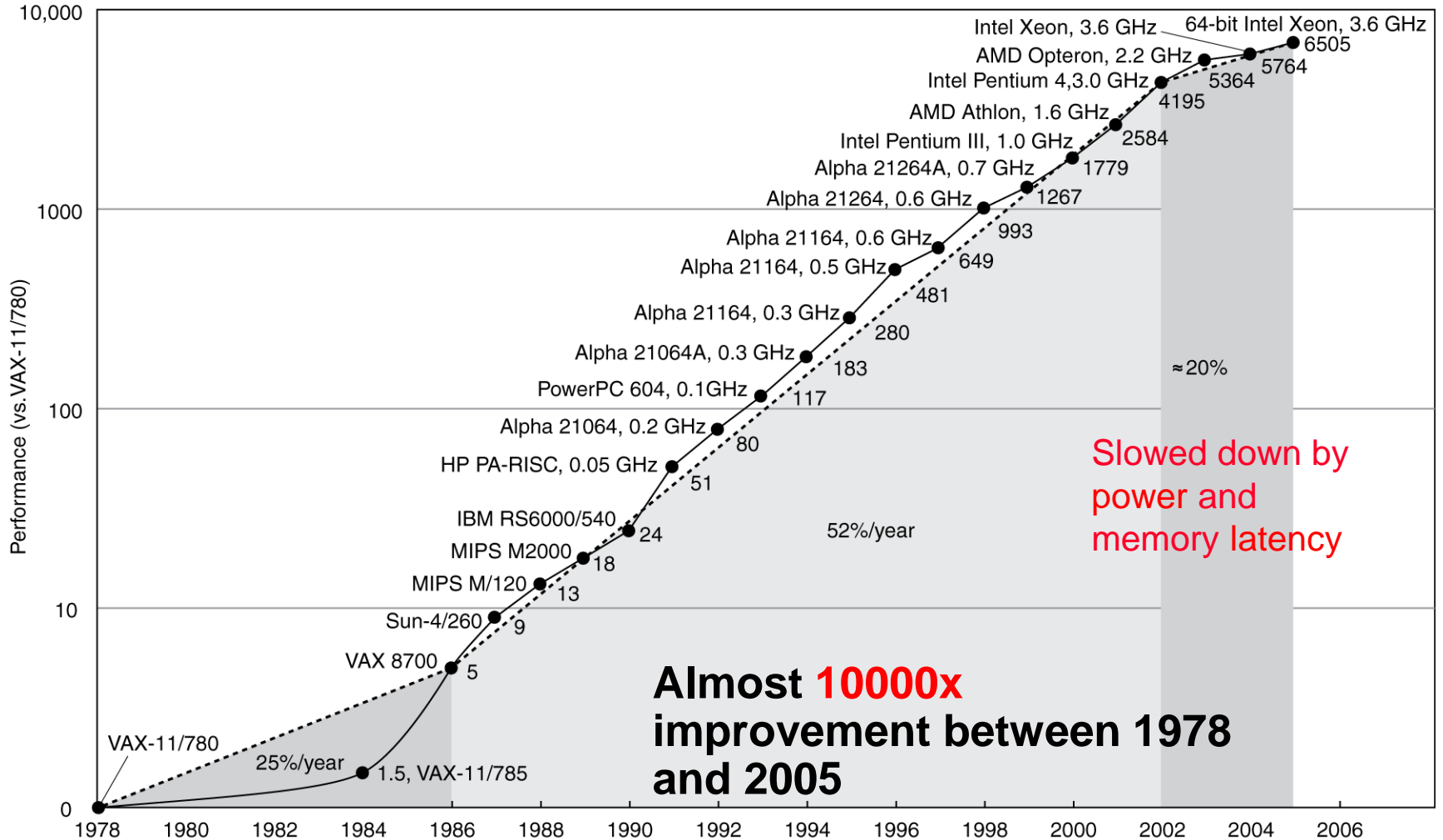
# Growth of Capacity per DRAM Chip

❖ DRAM capacity quadrupled almost every 3 years

★ 60% increase per year, for 20 years

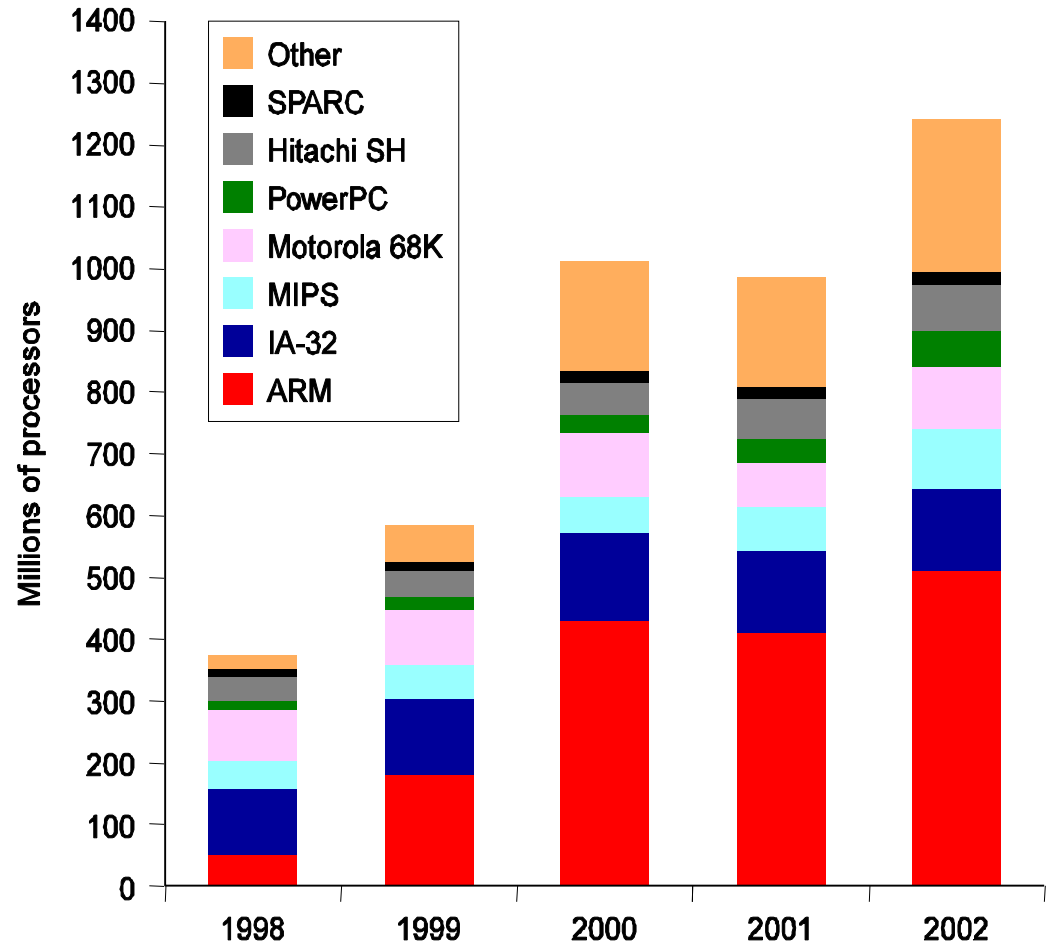


# Processor Performance (1978-2005)



# Microprocessor Sales (1998 – 2002)

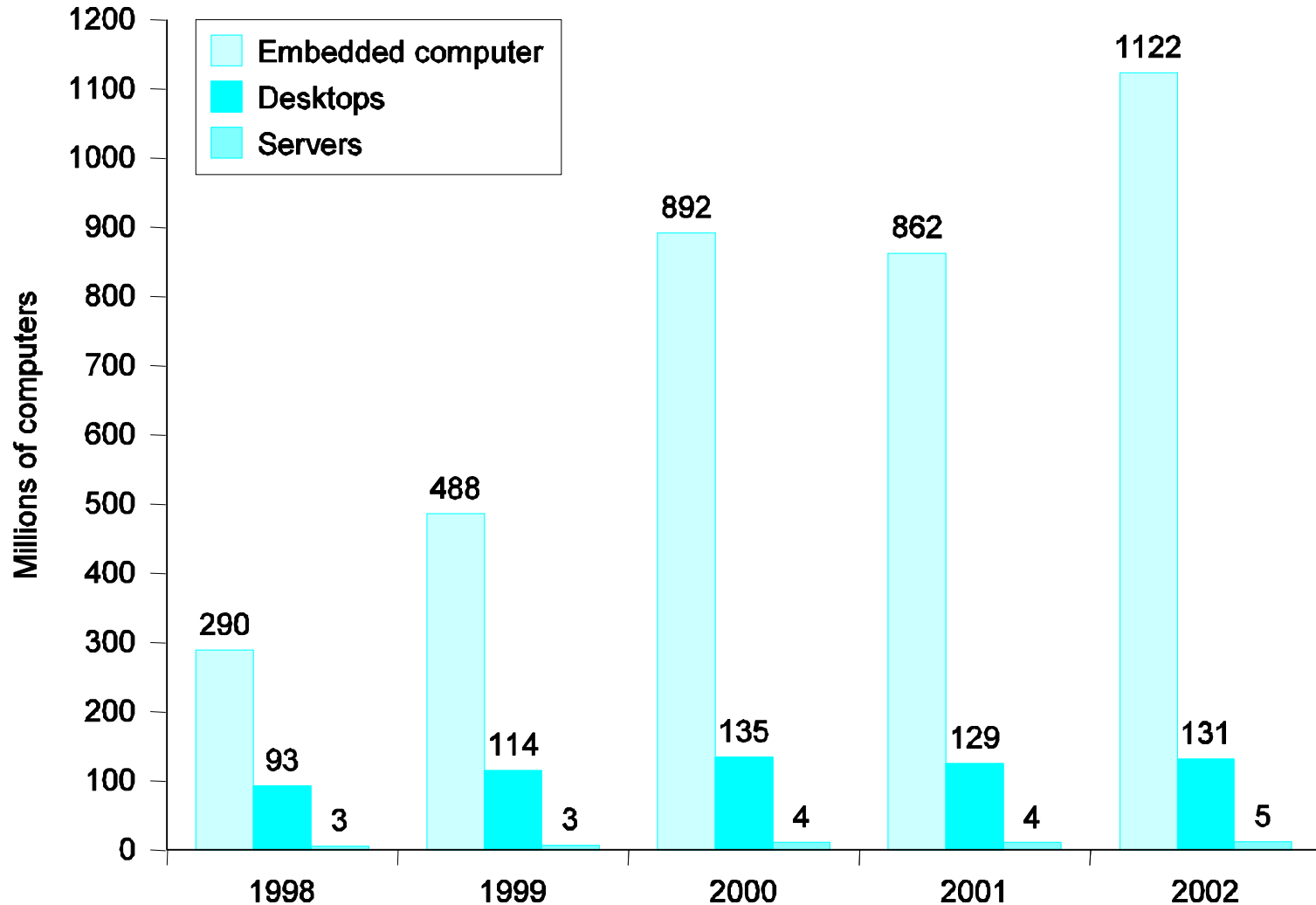
- ❖ ARM processor sales exceeded Intel IA-32 processors, which came second
- ❖ ARM processors are used mostly in cellular phones
- ❖ Most processors today are embedded in cell phones, digital TVs, video games, and a variety of consumer devices



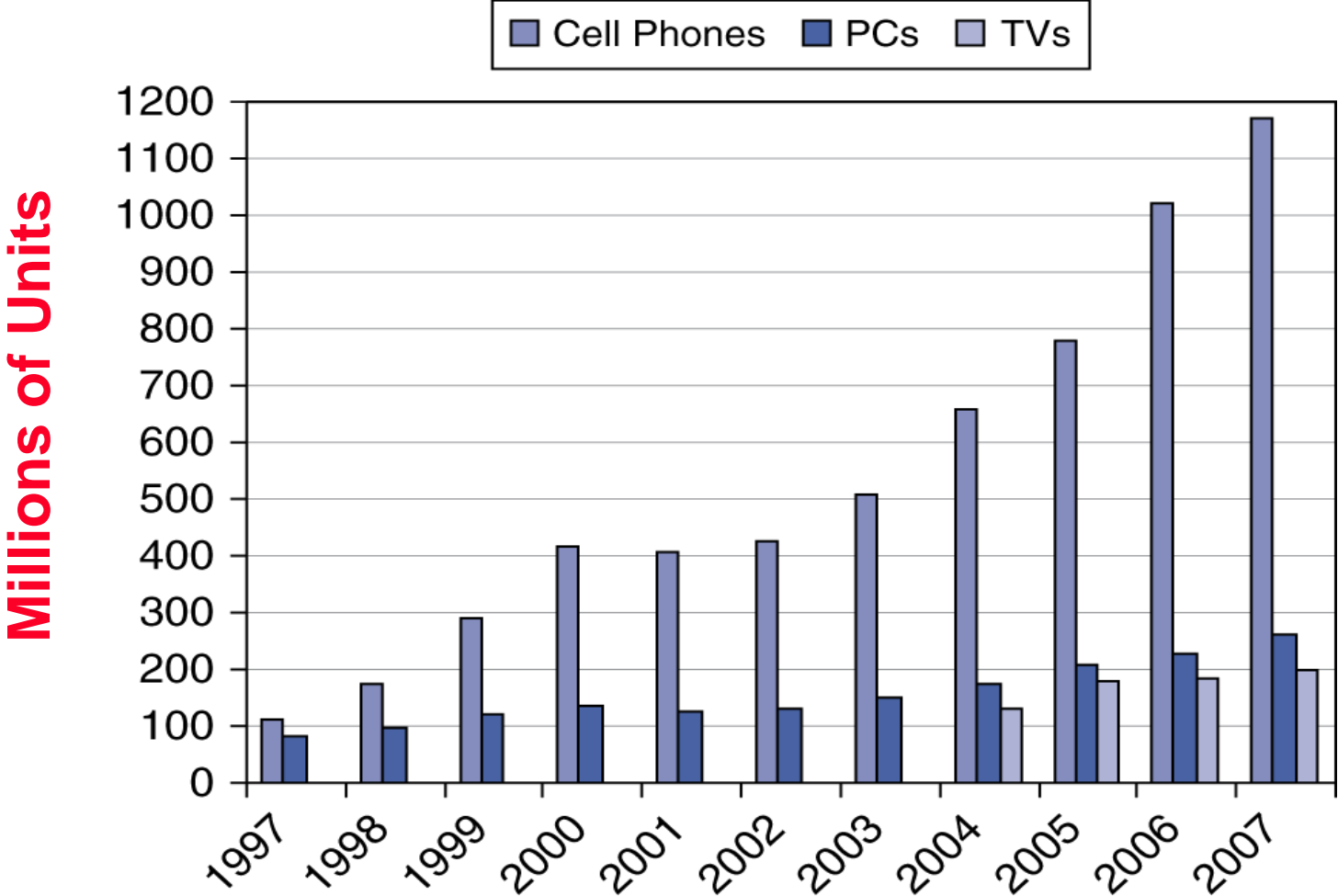
# Classes of Computers

- **Desktop / Notebook 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
- **Embedded Computers**
  - Hidden as components of systems
  - Stringent power/performance/cost constraints

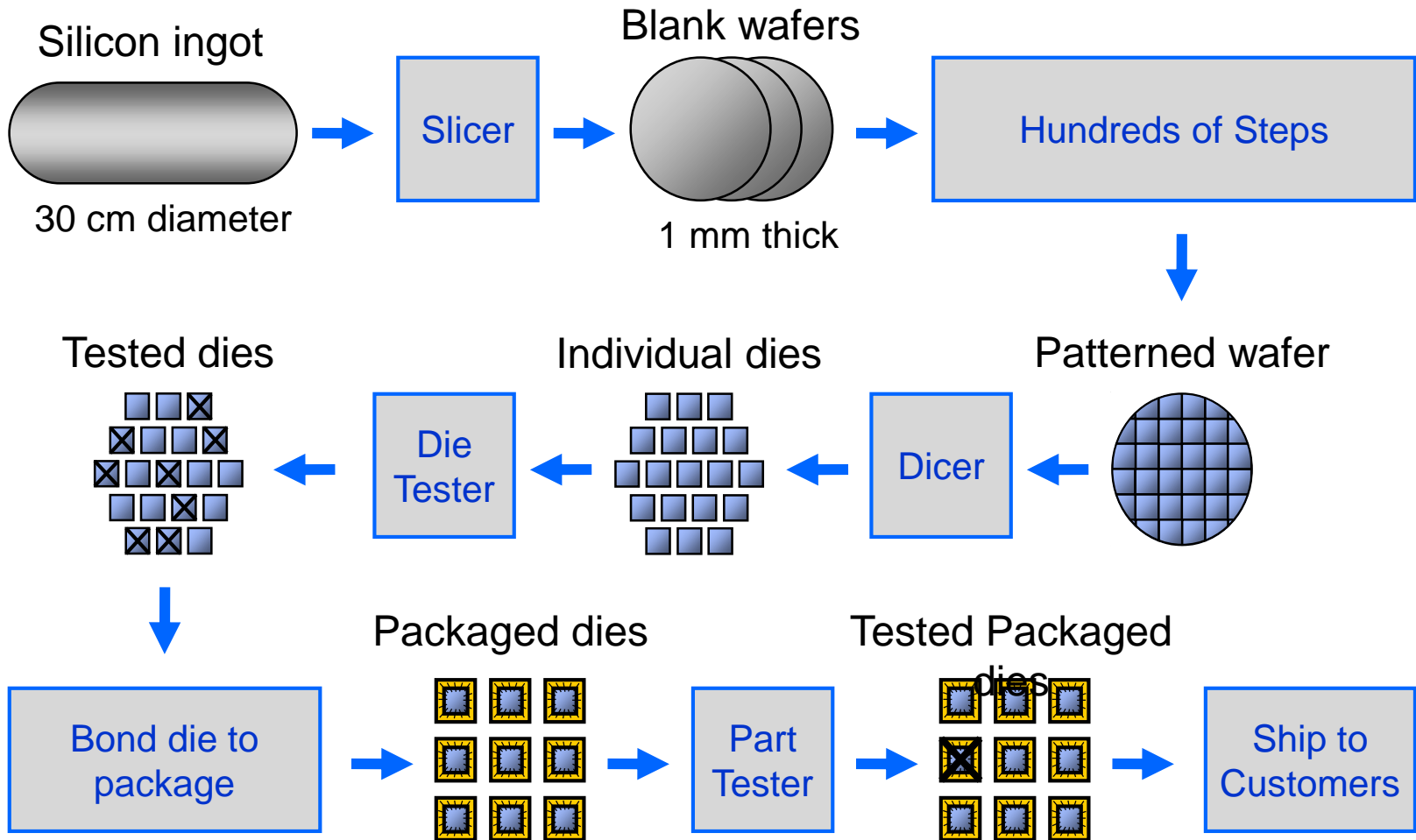
# Computer Sales (1998 – 2002)



# The Processor Market (1997-2007)



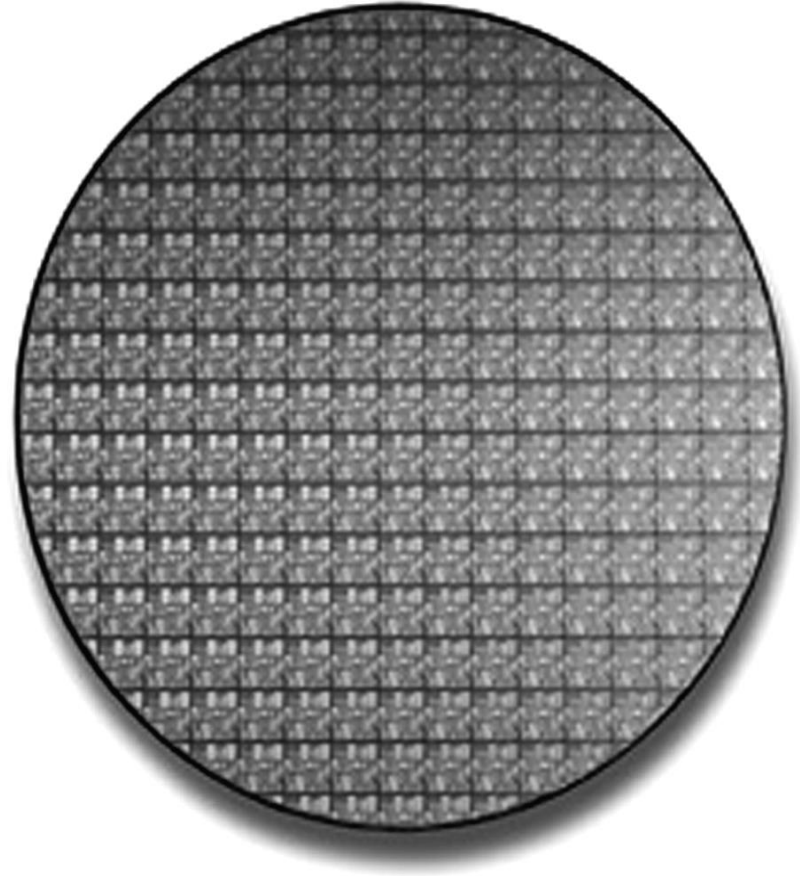
# Chip Manufacturing Process



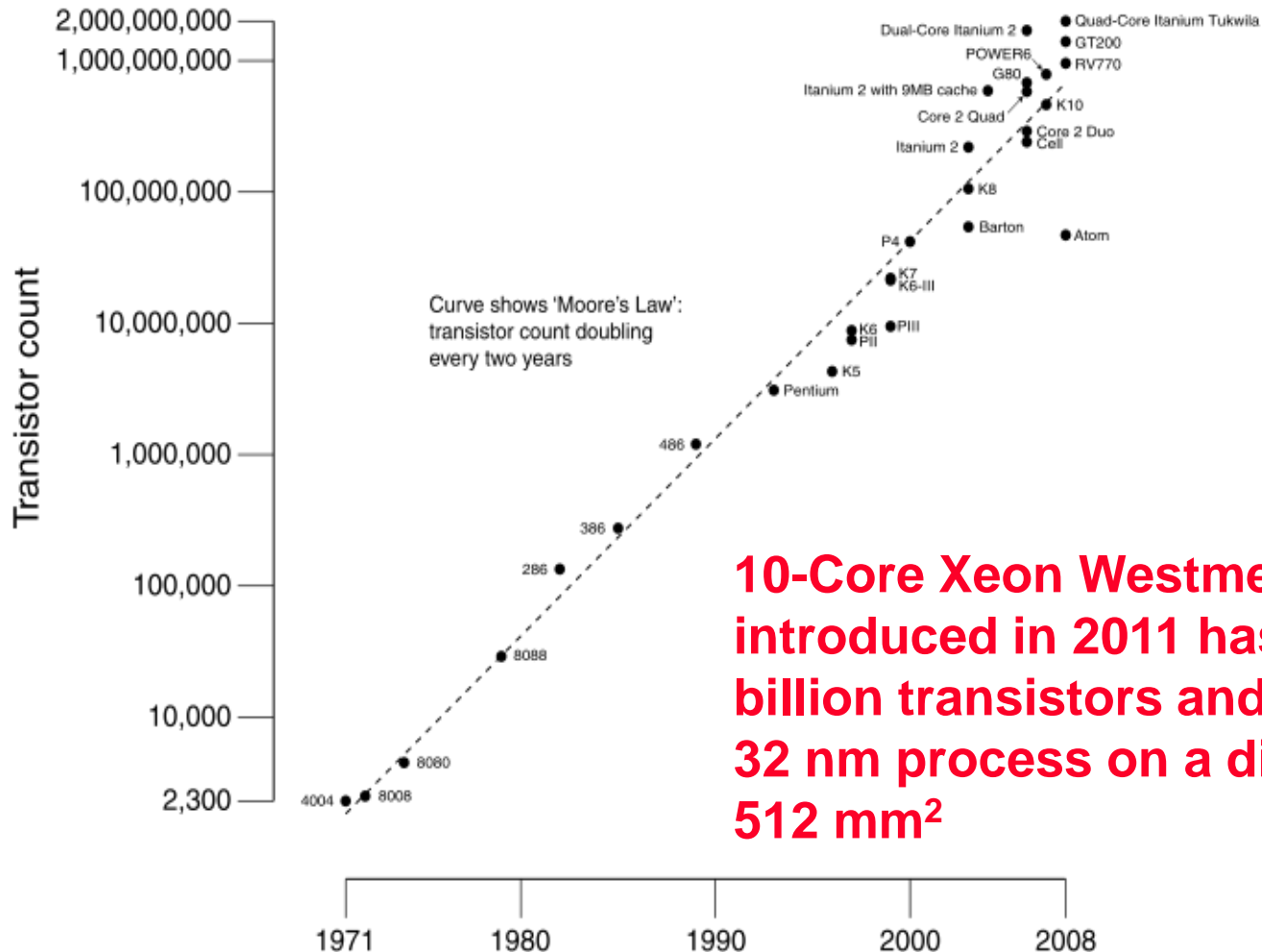


# Wafer of Pentium 4 Processors

- 8 inches (20 cm) in diameter
- Die area is 250 mm<sup>2</sup>
  - About 16 mm per side
- 55 million transistors per die
  - 0.18 μm technology
  - Size of smallest transistor
  - Improved technology uses
    - » 0.13 μm and 0.09 μm
- Dies per wafer = 169
  - When yield = 100%
  - Number is reduced after testing
  - Rounded dies at boundary are useless



# CPU Transistor Count (1971 – 2008)

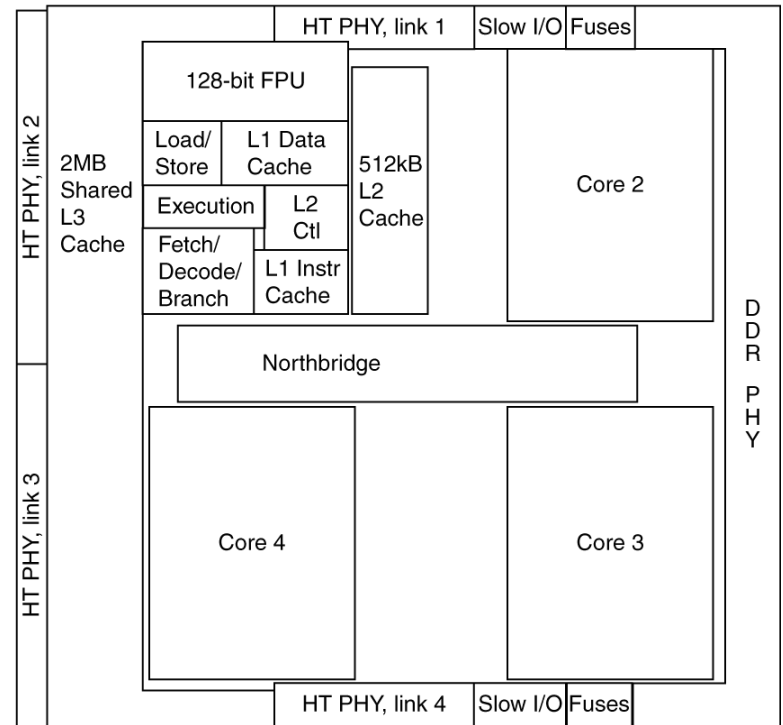
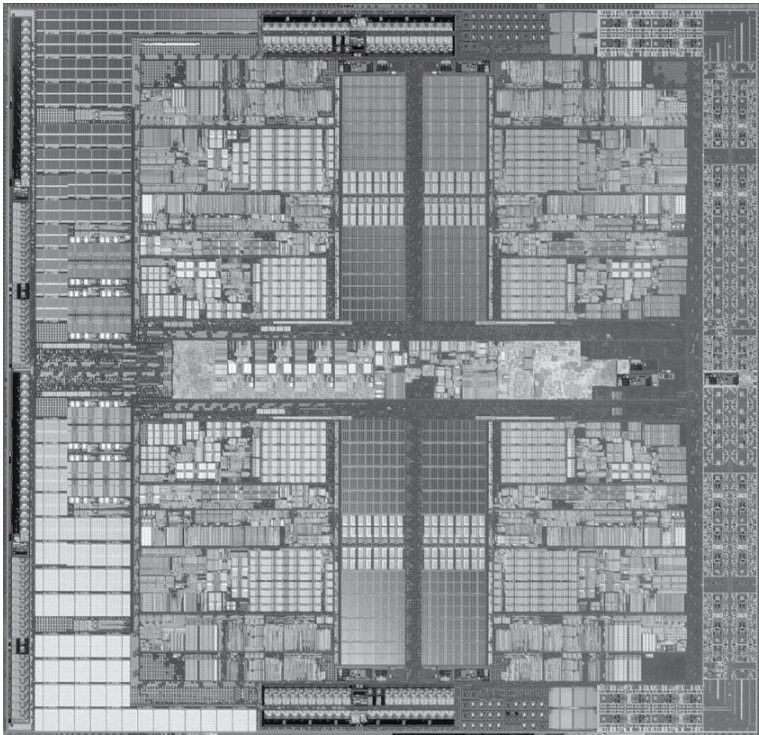


**10-Core Xeon Westmere-EX introduced in 2011 has 2.6 billion transistors and uses a 32 nm process on a die size = 512 mm<sup>2</sup>**

# Inside a Multicore Processor Chip

AMD Barcelona: 4 Processor Cores

3 Levels of Caches



# Course Roadmap

- **Instruction set architecture**
- **Performance issues**
- **Constructing a processor**
- **Pipelining to improve performance**
- **Memory: caches and virtual memory**
- **Introduction to Parallel Architectures**