

Introduction to Computer Architecture

Kuliah #1, 2020-21/Genap

CSCM601252 - Pengantar Organisasi Komputer

Instructor: Erdefi Rakun dan Tim Dosen POK

Fasilkom UI



Outline

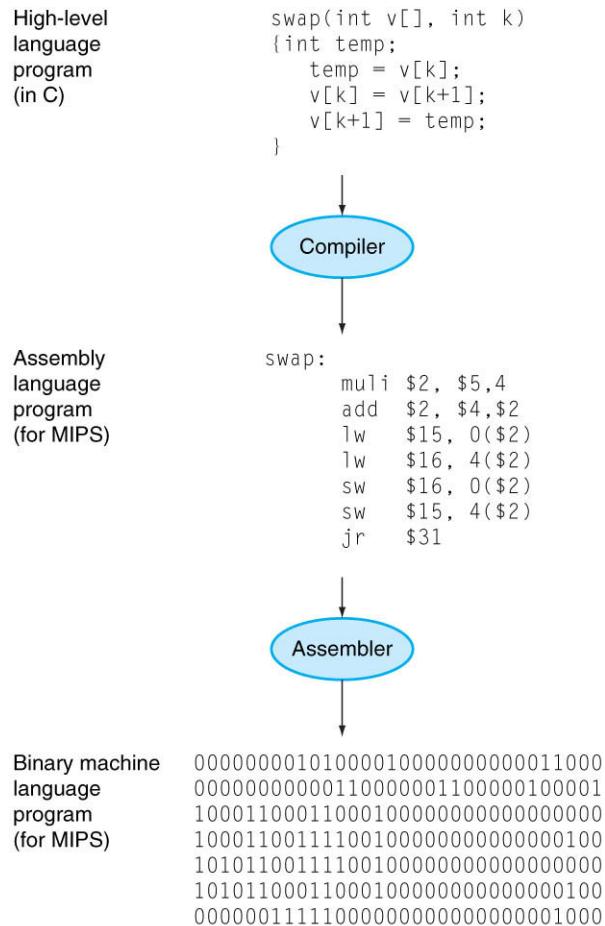
- Computer Organization/ Architecture
- von Neumann Architecture
- Buses
- Instruction Set Architecture (ISA)
- Clock Cycles
- Central Processing Unit (CPU)
- Code Execution
- Memory
- Moore's Law

Note: These slides are taken from Aaron Tan's slide

Organization & Architecture

- Computer organization: electronics engineer's view of a computer system.
- Computer architecture: assembly programmer's view of a computer system – an abstract view.
- In practice, difficult to distinguish the two.
- Who needs to study this? Software engineers, embedded systems programmers, computer engineers.
- In-depth understanding of the inner workings of computers, with emphasis on concept understanding rather than hardware implementation.

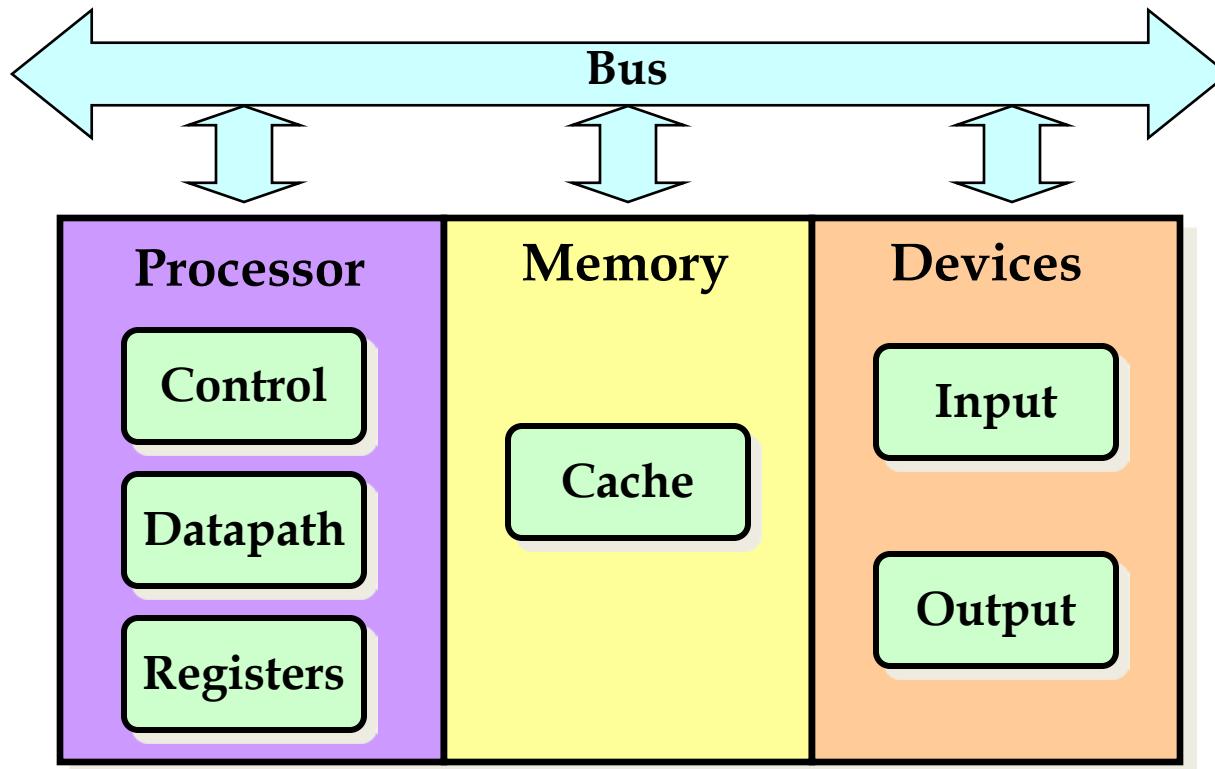
Below Your Program



- You write programs in high level programming languages, e.g., C, Java
 - $A+B$
 - Compiler translates this into assembly language statement
 - add A,B
 - Assembler translates this statement into machine language instructions that the processor can execute
 - 1000110010100000

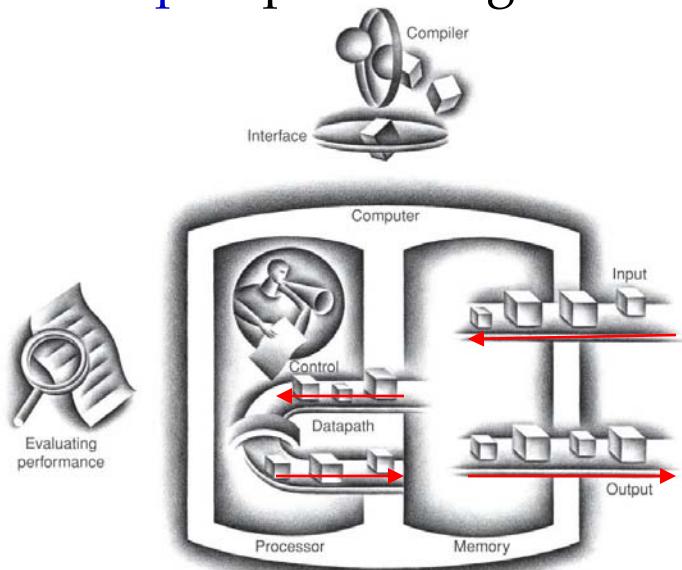
von Neumann Architecture

- von Neumann architecture: Programs and data are stored in memory (stored-memory concept).
- Consists of processor, memory and devices.
- Data are carried along **buses** between components.



Components of Computer

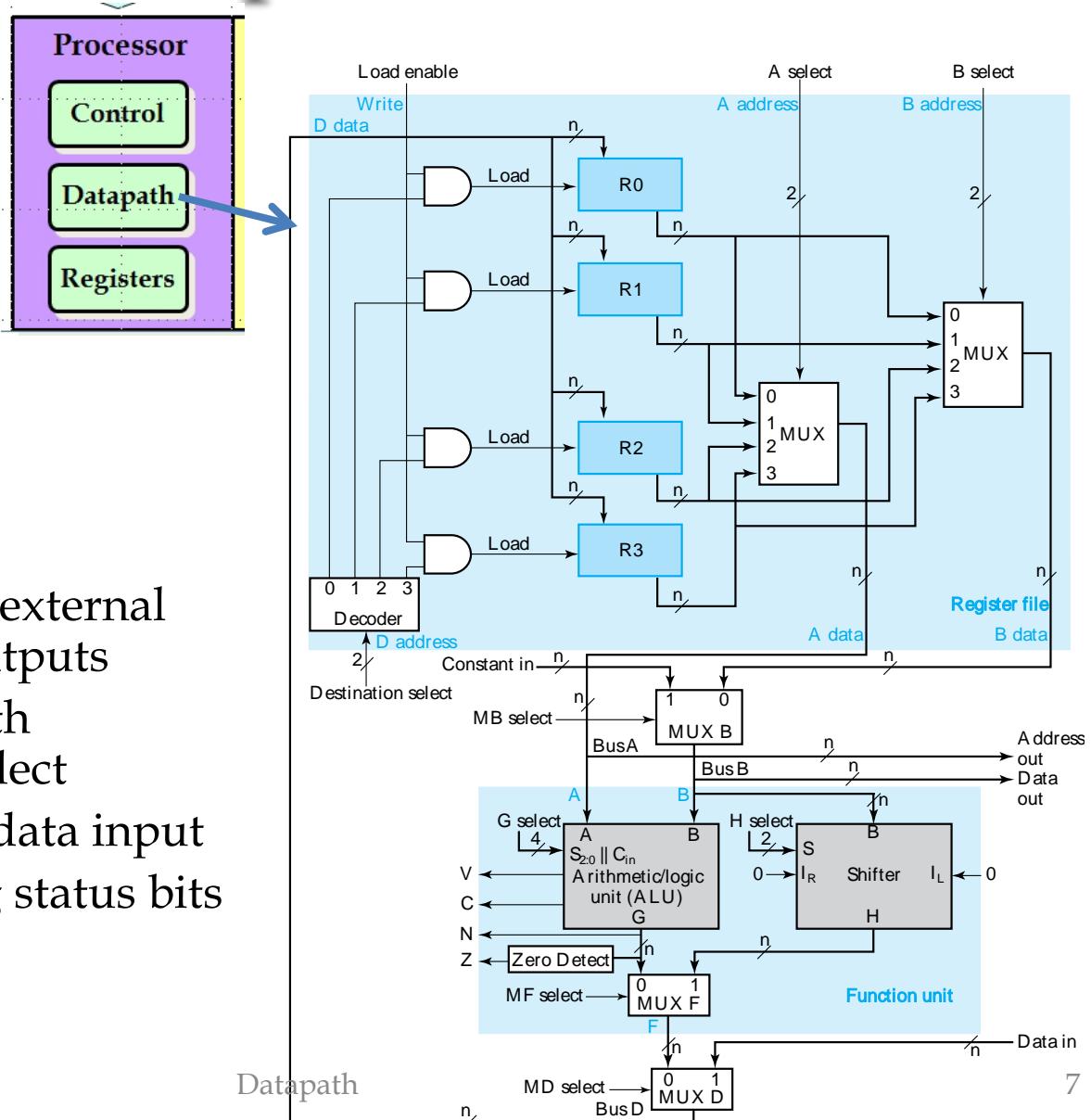
- **Memory:** stores program and data
- **Input:** feeds data (keyboard, mouse)
- **Datapath:** reads data from memory, processes it, writes it back to memory
- **Control:** Sends signals that determines the operation of datapath, memory, I/O
- **Output:** processing result to user (display)



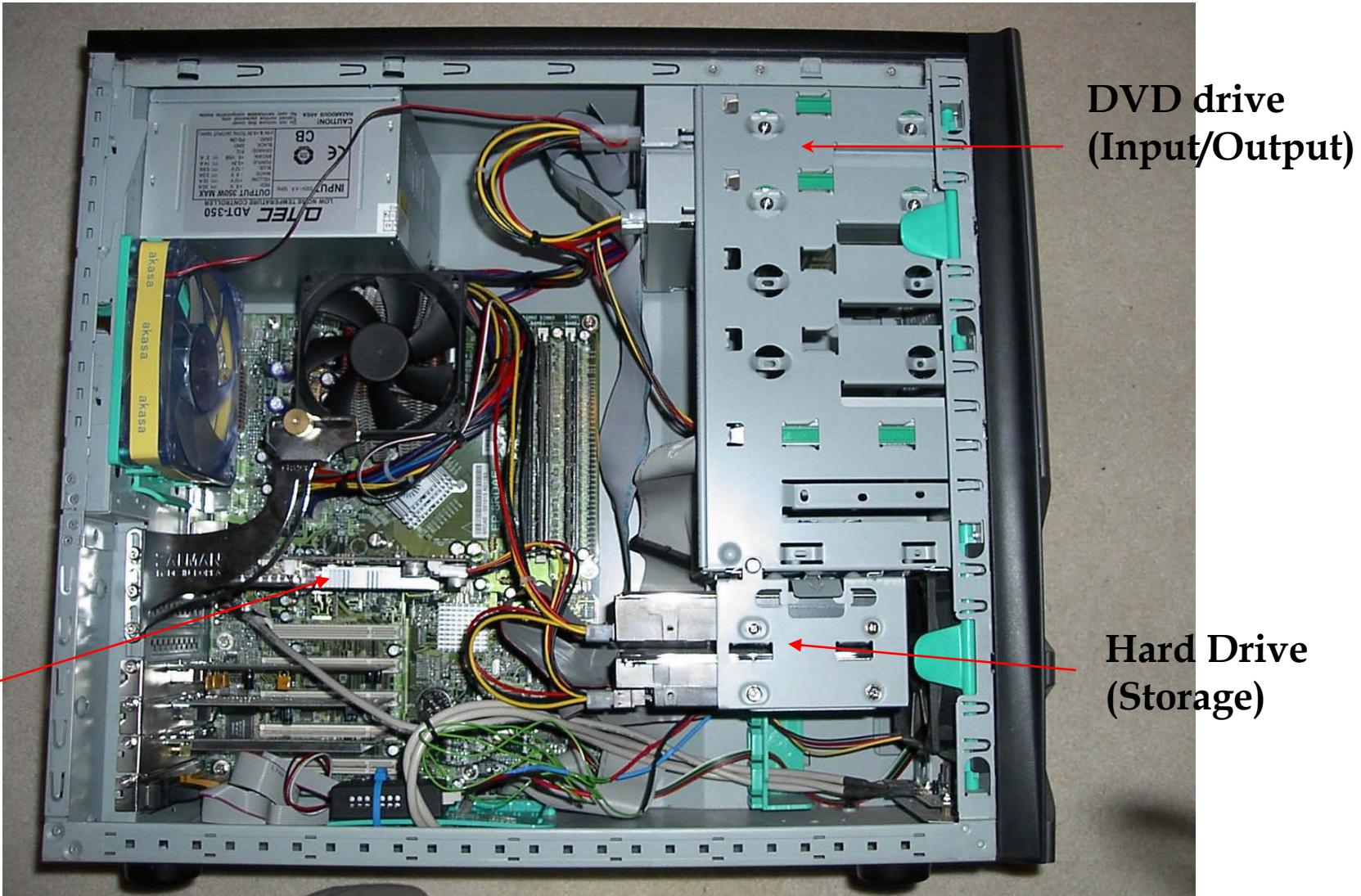
- Processor comprises datapath and control.

Datapath Example

- Four parallel-load registers
- Two mux-based register selectors
- Register destination decoder
- Mux B for external constant input
- Buses A and B with external address and data outputs
- ALU and Shifter with Mux F for output select
- Mux D for external data input
- Logic for generating status bits V, C, N, Z



Inside Your Desktop



Inside Your Laptop

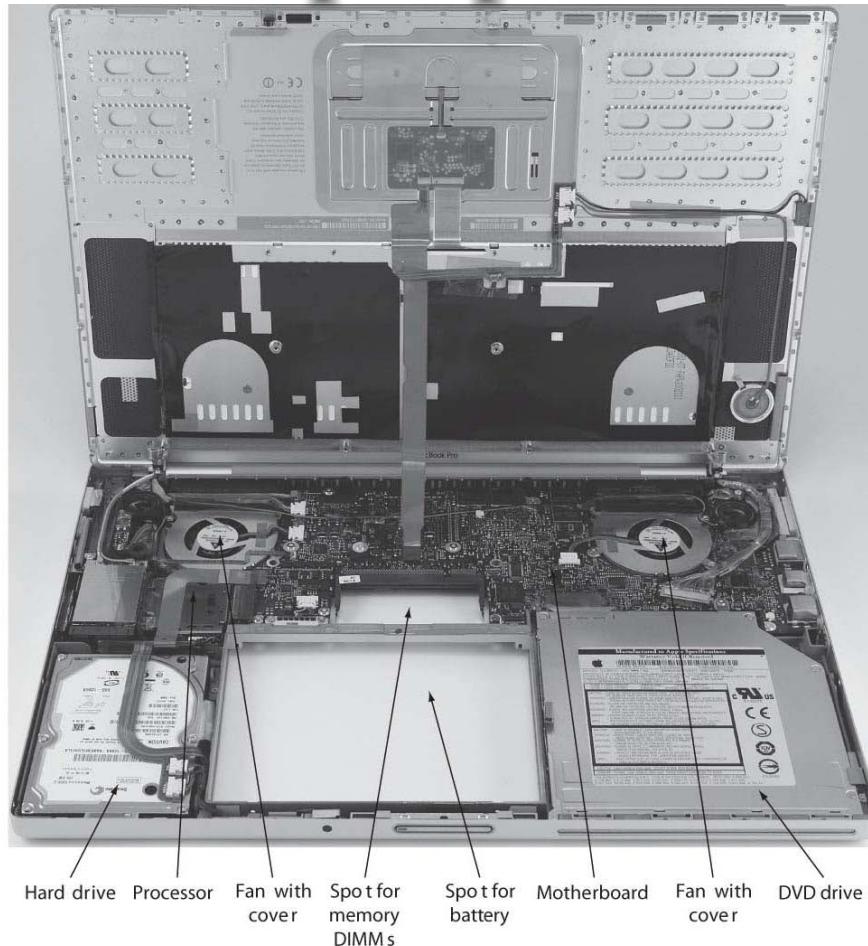


FIGURE 1.7 Inside the laptop computer of Figure 1.5.

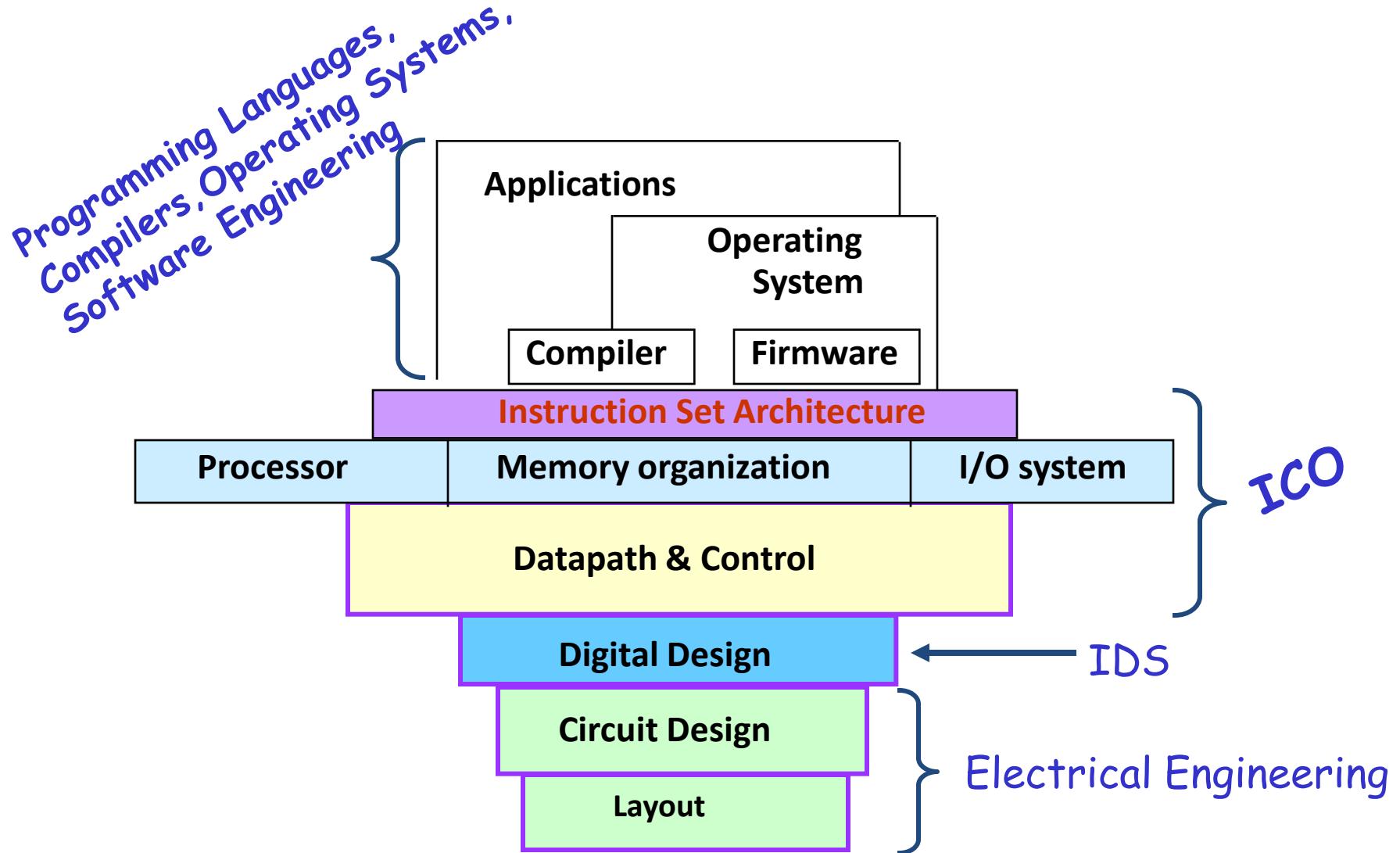
Buses

- **Bus**: a communication path between components.
 - Data bus, control bus, address bus.
- **Bus width**: the number of lines (bits).
- Data bus width usually coincides with word size, which is also usually the register size.
- Address bus width determines the addressable address range.
 - An n -bit address bus can address up to 2^n addresses (locations).

ISA (1/4)

- Instruction Set Architecture (ISA) is the part of computer architecture related to programming, as seen by the programmer.
- It includes a specification of a set of opcodes (machine language).
- It serves as the interface between hardware and software.

ISA (2/4)



ISA (3/4)

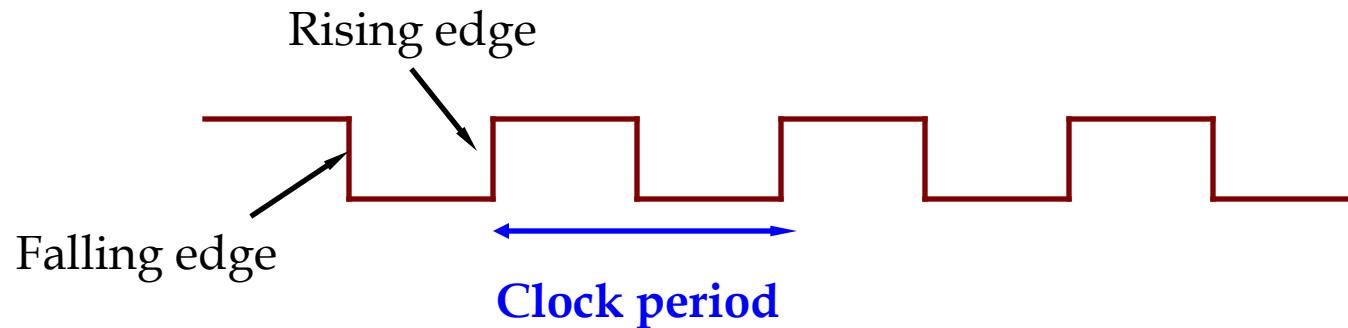
- Some examples:
 - Digital Alpha (v1, v3)
 - HP PA-RISC (v1.1, v2.0)
 - Sun Sparc (v8, v9)
 - SGI MIPS (MIPS I-V)
 - Intel (8086,80x86,Pentium,...)

ISA (4/4)

- Issues concerning ISA design:
 - Organisation of programmable storage
 - Data types and data structures: encoding and representations
 - Instruction set
 - Instruction formats
 - Modes of addressing and accessing data items and instructions
 - Exceptional conditions

Clock Cycles (1/2)

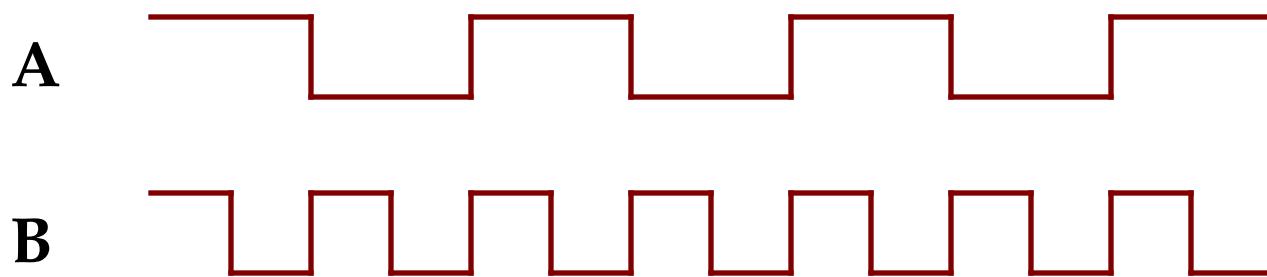
- A synchronous system is synchronised according to a **clock**.



- A **clock cycle** is the duration between two consecutive rising (or falling) edges, and its duration is also known as the **clock period**.

Clock Cycles (2/2)

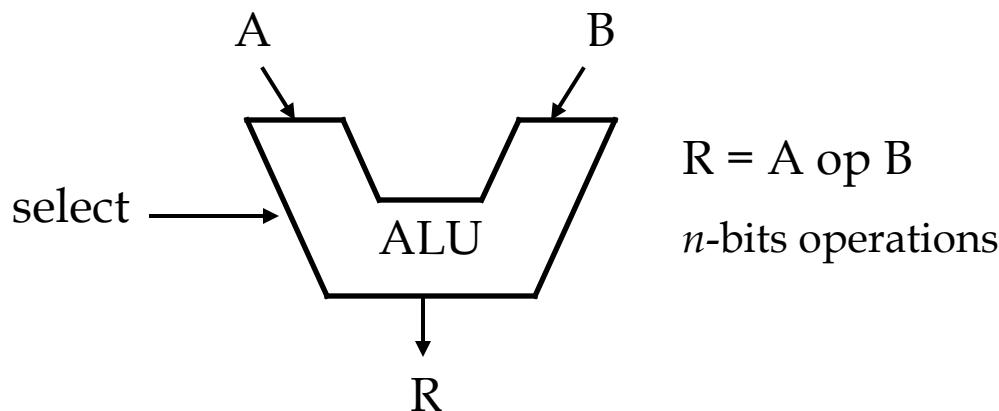
- The **clock frequency** is the reciprocal of clock period, measured in Hertz (Hz): number of cycles per second.
 - Example: A clock with period of 250ns (nanoseconds) has a frequency of 4MHz.



- Clock A has twice the clock period than that of clock B, and half the frequency of B's.

Central Processing Unit (1/3)

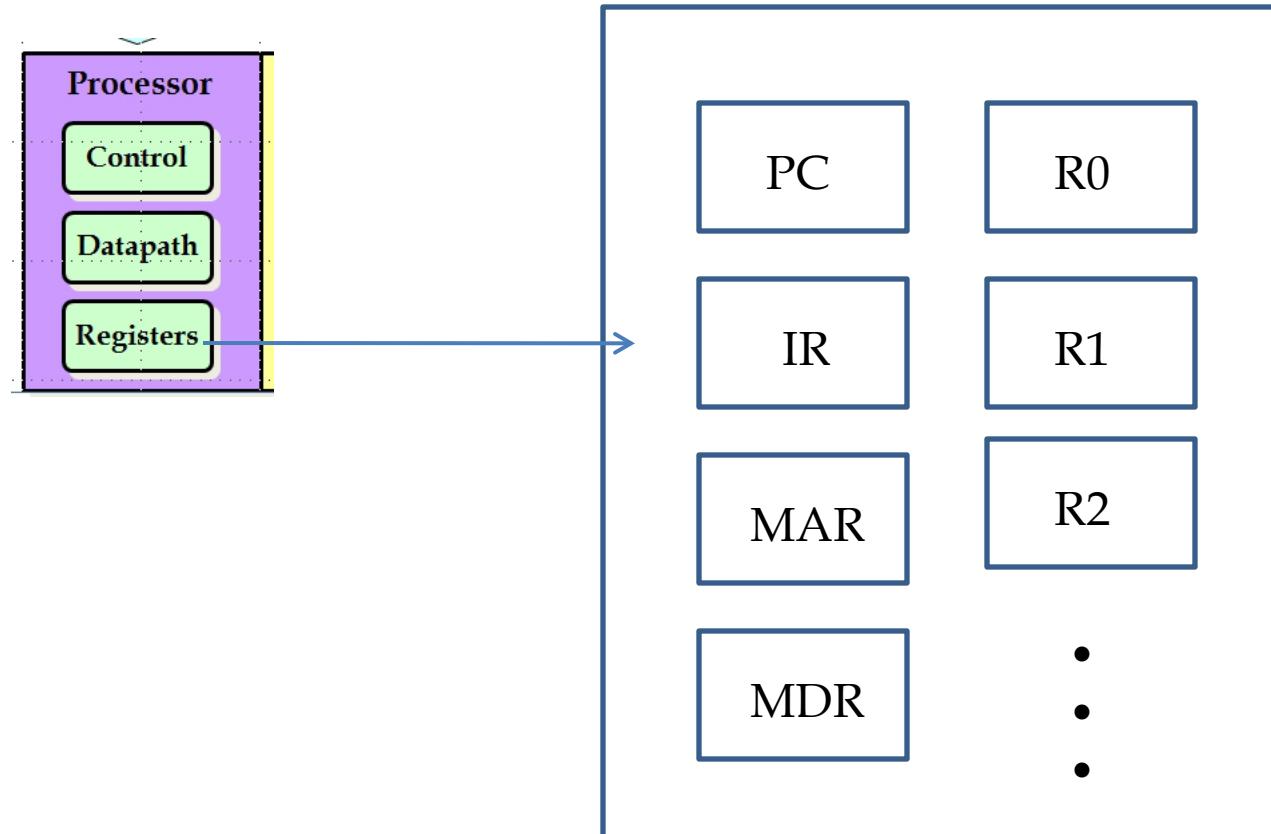
- CPU = Control Unit + ALU + Registers.
- Control unit: monitors and directs sequences of instructions.
- ALU (Arithmetic-Logic Unit): performs simple arithmetic and logical operations.
 - Examples: Add, subtract, and, or, invert, increment, etc.



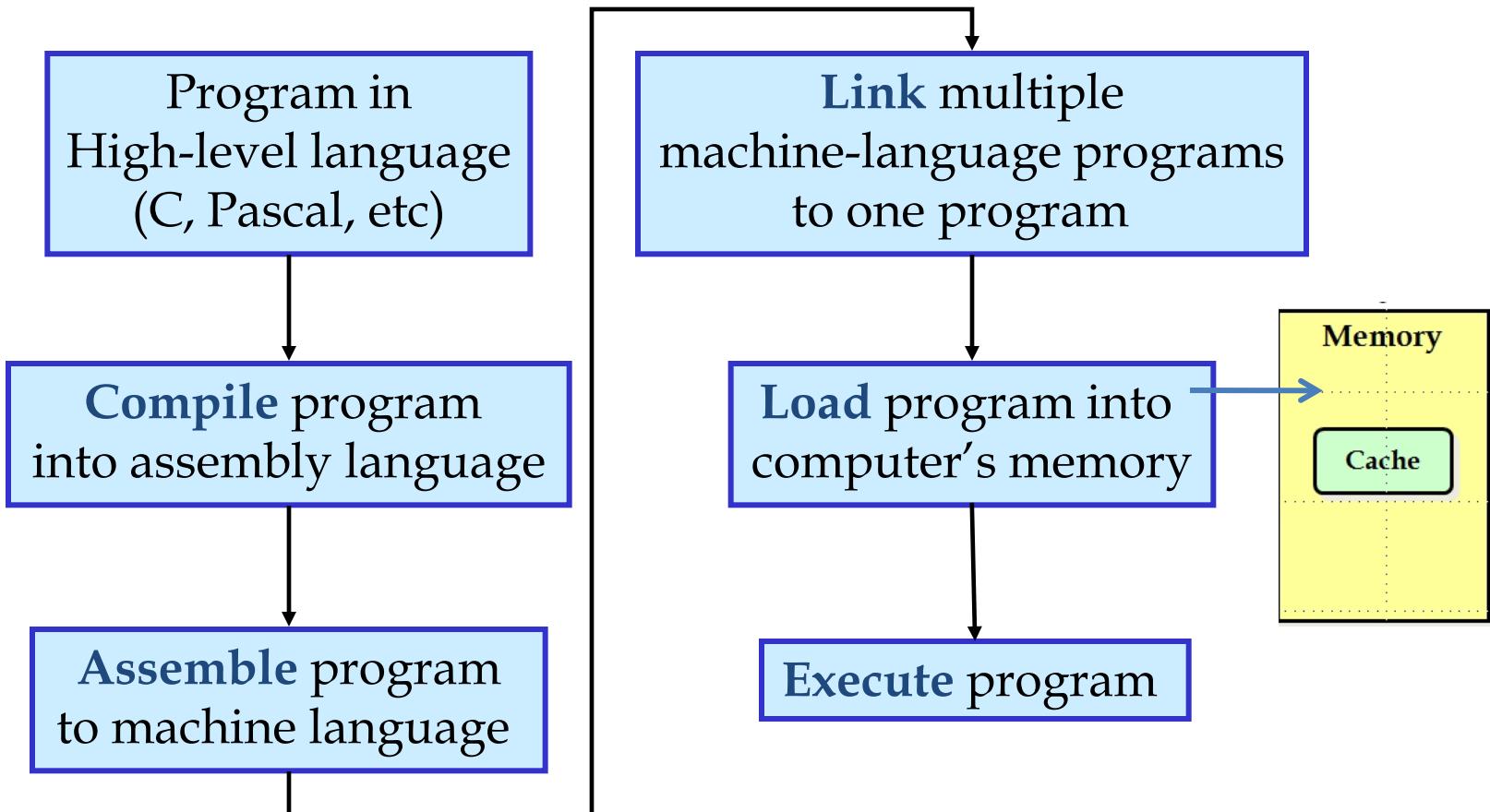
Central Processing Unit (2/3)

- Registers: Fast memories in the CPU, storing operands, temporary results and status information.
- General-purpose registers and special registers:
 - PC (program counter)
 - ACC (accumulator)
 - IR (instruction register)
 - MAR (memory address register)
 - MBR (memory buffer register) or MDR (memory data register)

Central Processing Unit (2/3)

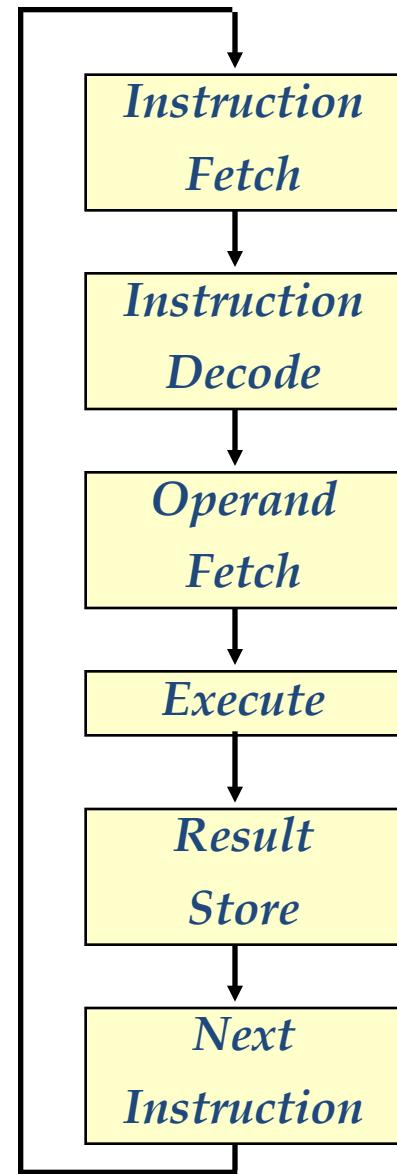


Code Execution (1/2)

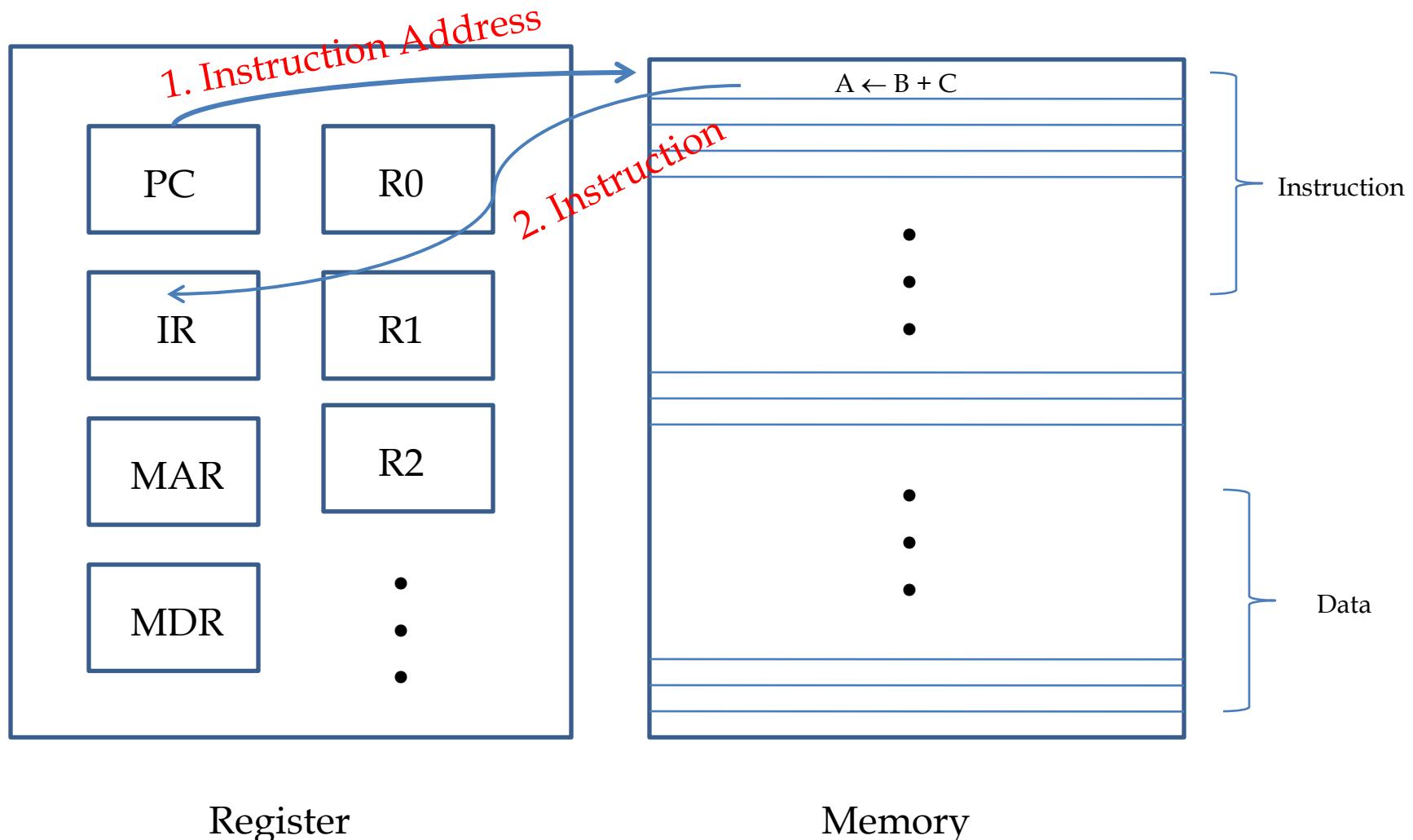


Code Execution (2/2)

- Instruction execution cycle:
fetch, decode, execute.
 - Fetch: fetch next instruction (using PC) from memory into IR.
 - Decode: decode the instruction.
 - Execute: execute instruction.



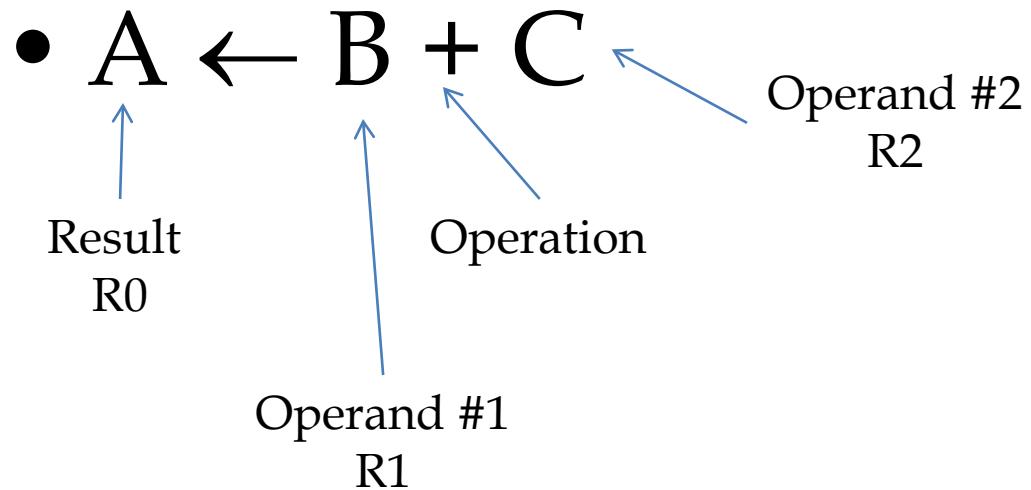
Code Execution - Example Fetch



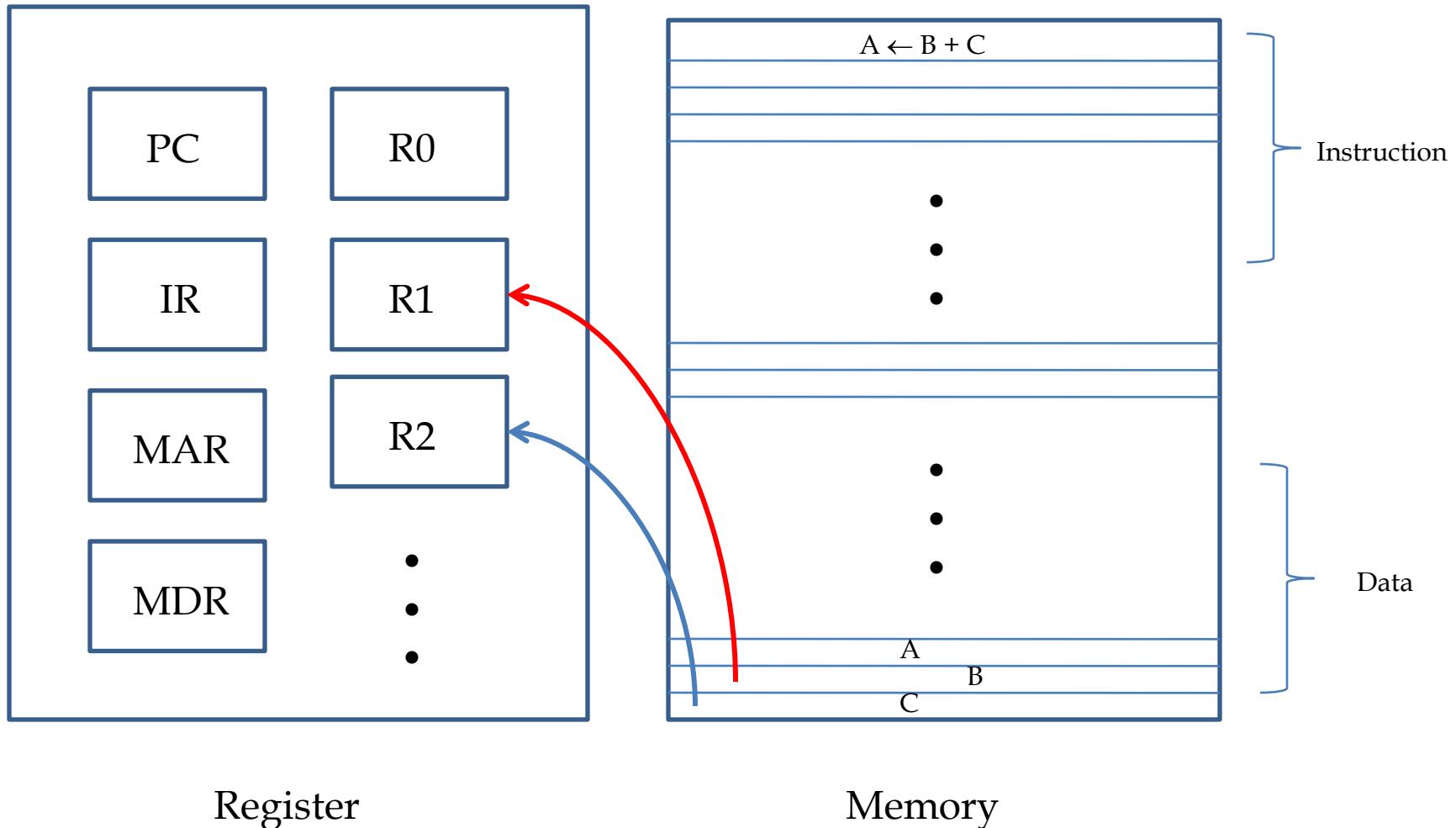
Register

Memory

Code Execution - Example Decode

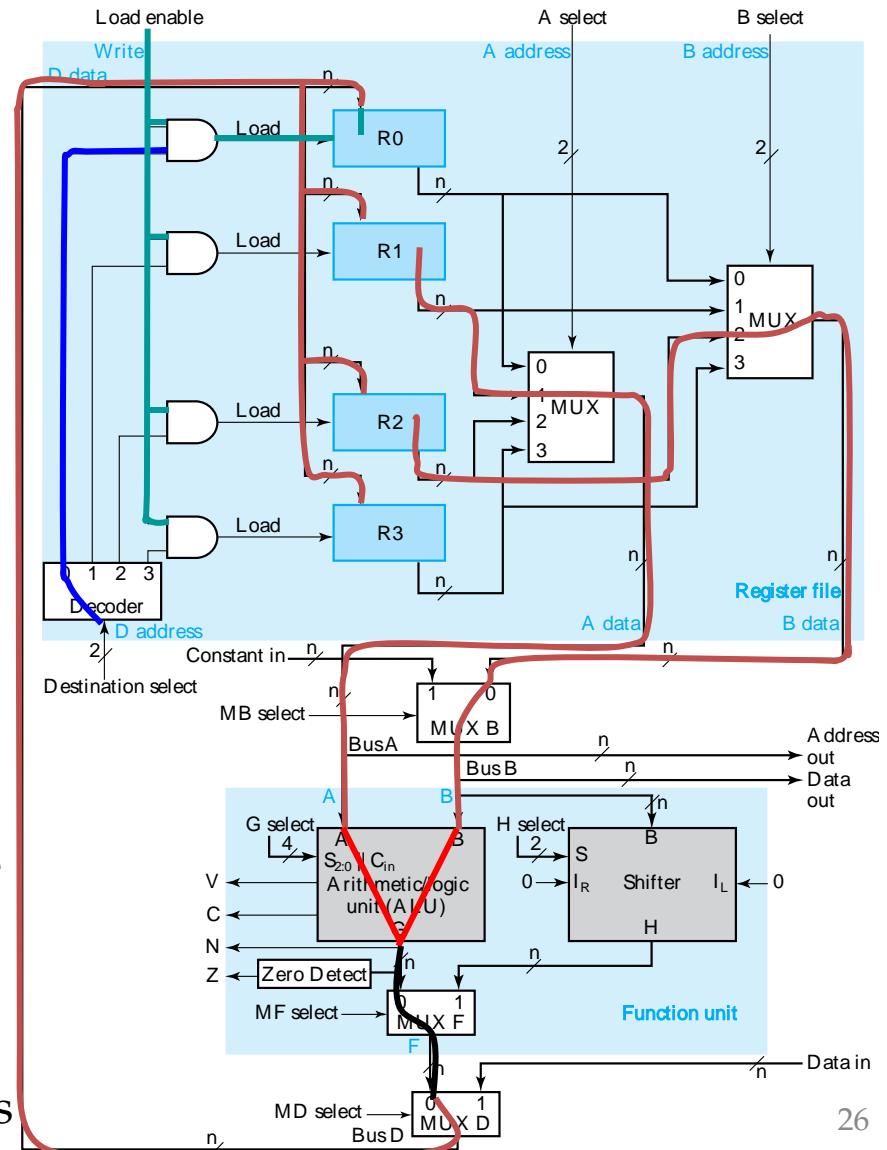


Code Execution - Example Operand Fetch



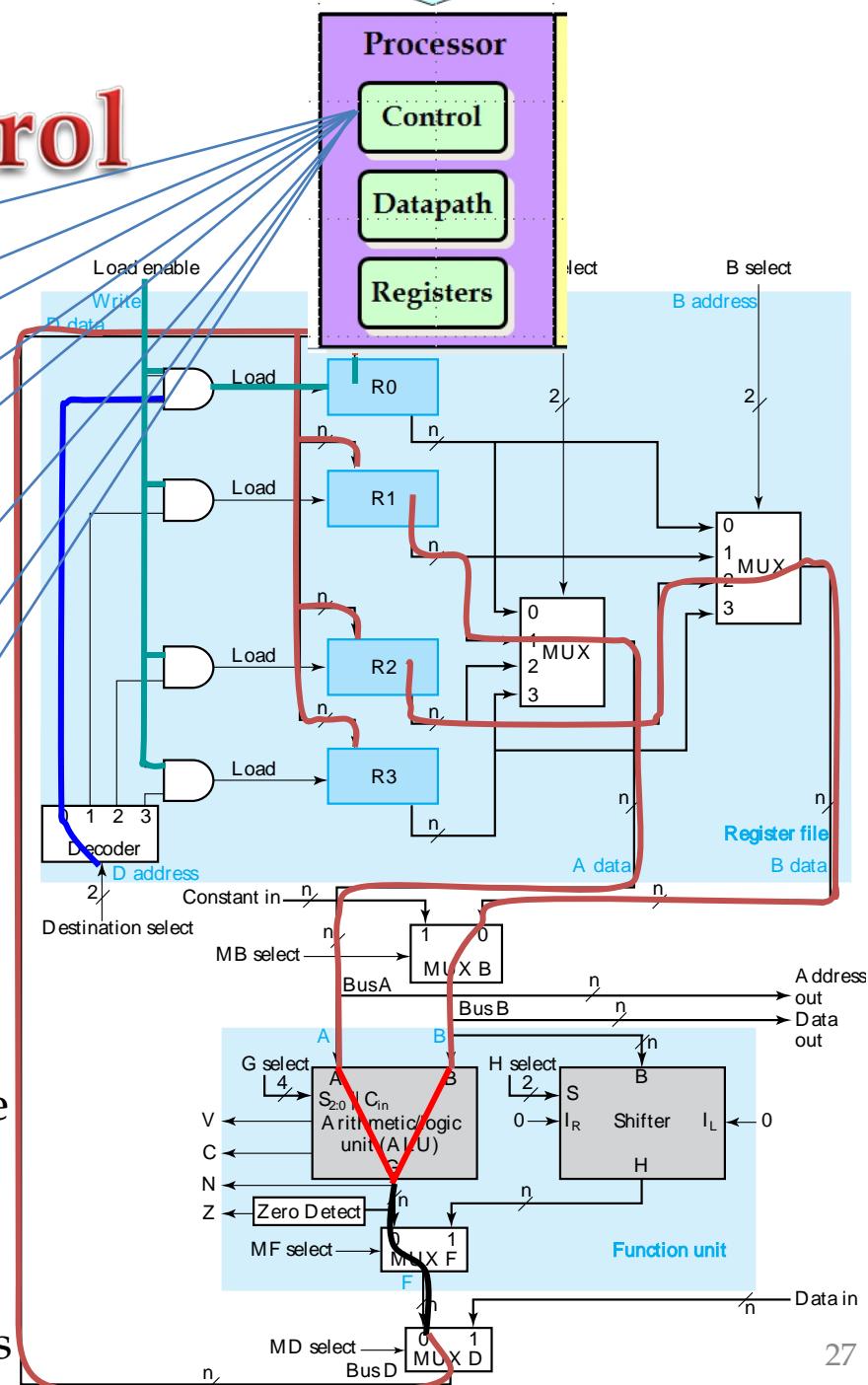
Code Execution - Example Execute

- Microoperation: $R0 \leftarrow R1 + R2$
- Apply 01 to A select to place contents of R1 onto Bus A
- Apply 10 to B select to place contents of R2 onto B data and apply 0 to MB select to place B data on Bus B
- Apply 0010 to G select to perform addition $G = \text{Bus A} + \text{Bus B}$
- Apply 0 to MF select and 0 to MD select to place the value of G onto BUS D
- Apply 00 to Destination select to enable the Load input to R0
- Apply 1 to Load Enable to force the Load input to R0 to 1 so that R0 is loaded on the clock pulse (not shown)
- The overall microoperation requires 1 clock cycle

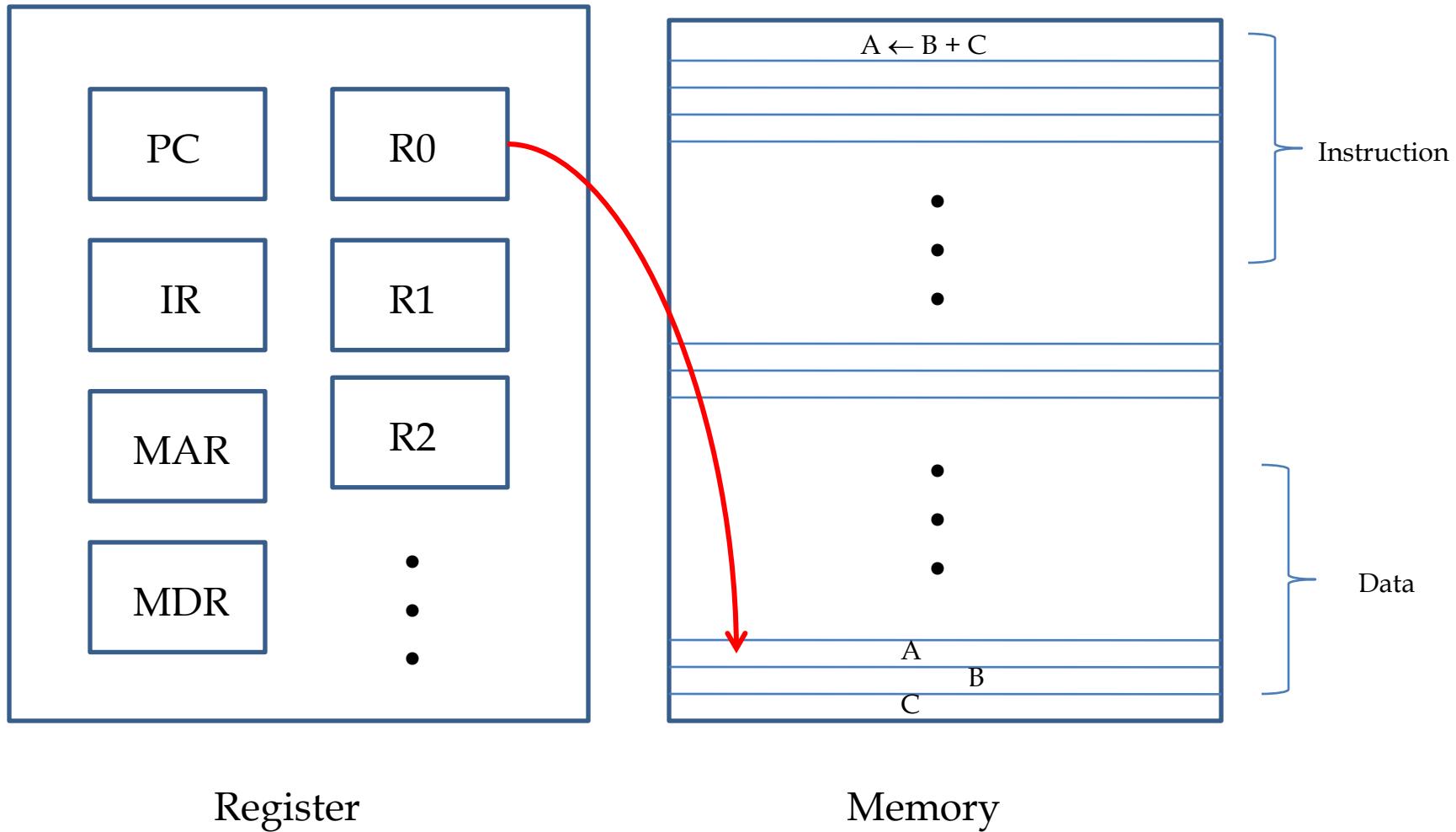


Processor - Control

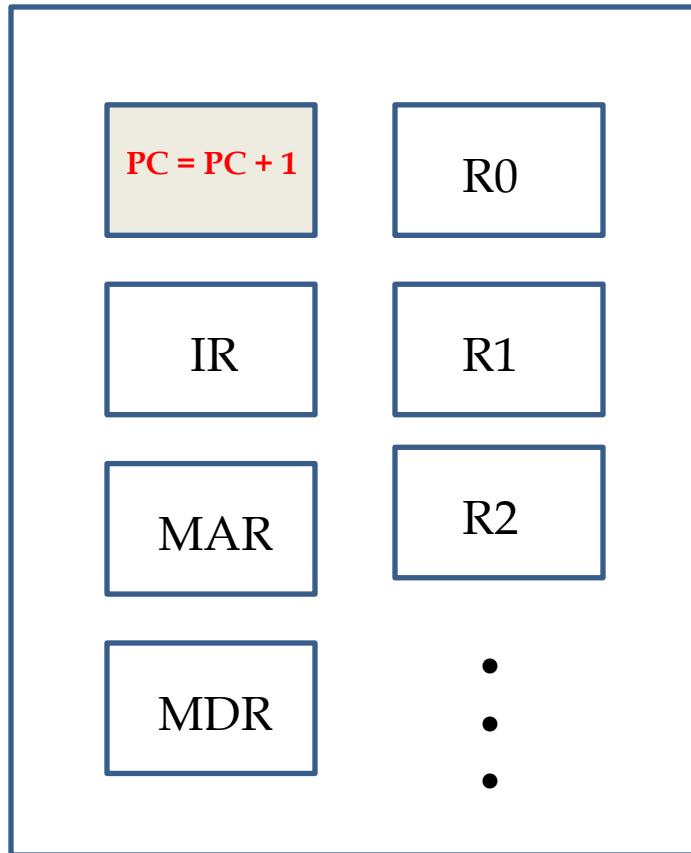
- Microoperation: $R0 \leftarrow R1 + R2$
- Apply **01 to A select** to place contents of R1 onto Bus A
- Apply **10 to B select** to place contents of R2 onto B data and apply **0 to MB select** to place B data on Bus B
- Apply **0010 to G select** to perform addition $G = \text{Bus A} + \text{Bus B}$
- Apply **0 to MF select** and **0 to MD select** to place the value of G onto BUS D
- Apply **00 to Destination select** to enable the Load input to R0
- Apply **1 to Load Enable** to force the Load input to R0 to 1 so that R0 is loaded on the clock pulse (not shown)
- The overall microoperation requires 1 clock cycle



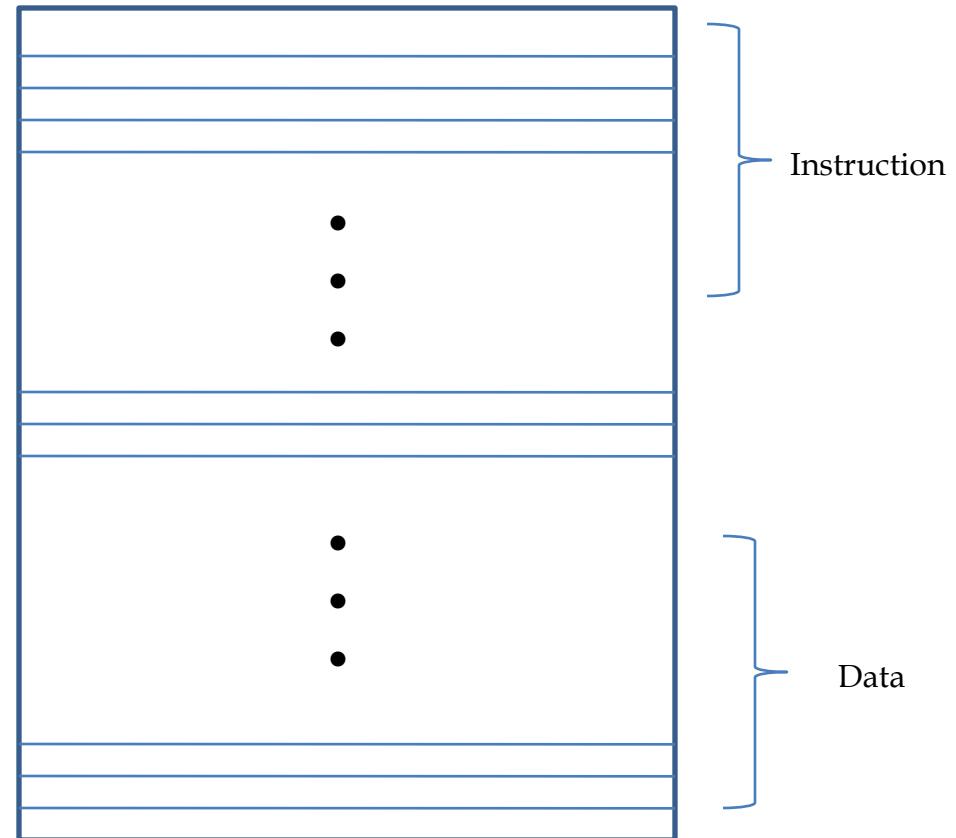
Code Execution - Example Result Store



Code Execution - Example Next Instruction



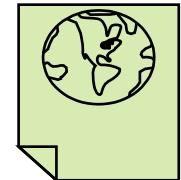
Register



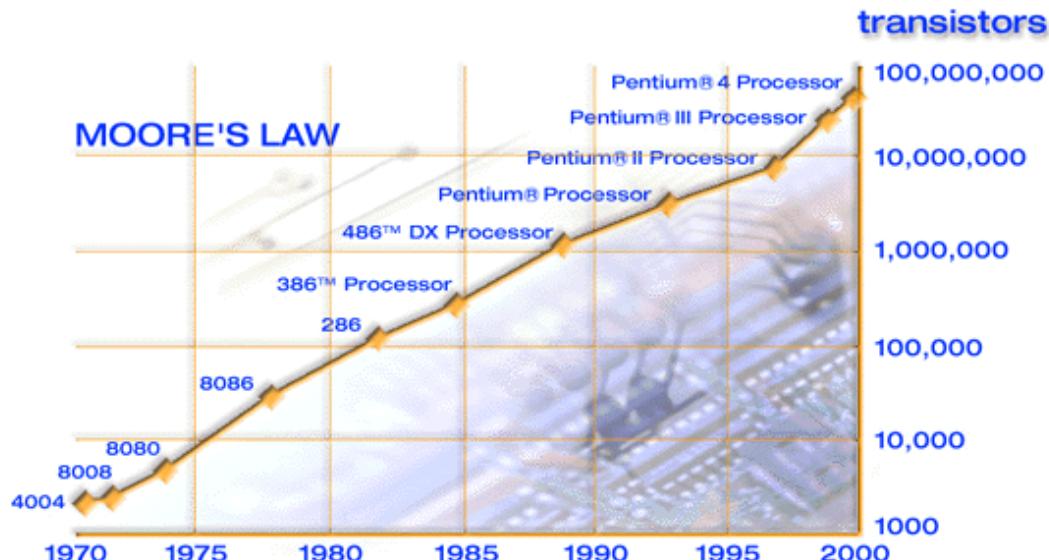
Memory

Moore's Law

- Intel co-founder Gordon Moore predicted in 1965 that Transistor density will double every 18 months → Increase in clock frequency
- Moore's Law 40th Anniversary



Gordon E. Moore, Co-founder,
Intel Corporation.
Copyright © 2005 Intel



<http://www.intel.com/research/silicon/mooreslaw.htm>

Ack: This is taken from Dr Tulika's materials.

Memory Capacity Increase

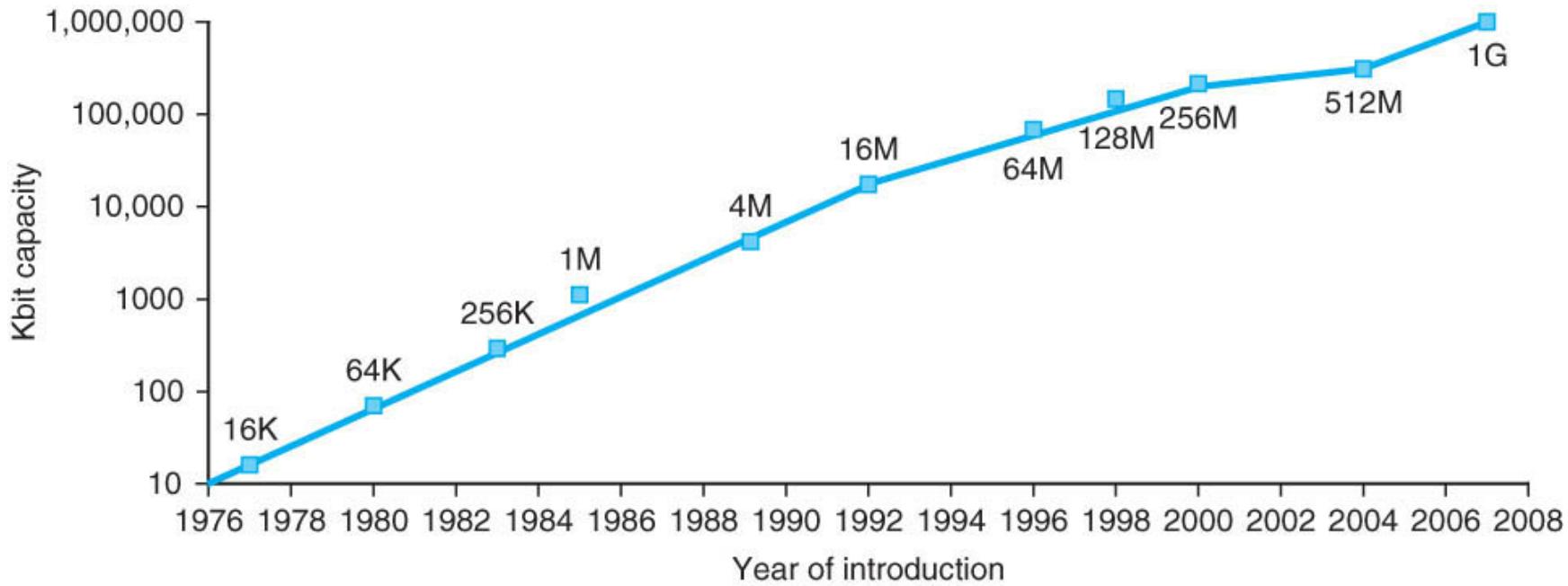


FIGURE 1.12 Growth of capacity per DRAM chip over time. The y-axis is measured in Kilobits, where $K = 1024$ (2^{10}). The DRAM industry quadrupled capacity almost every three years, a 60% increase per year, for 20 years. In recent years, the rate has slowed down and is somewhat closer to doubling every two years to three years. Copyright © 2009 Elsevier, Inc. All rights reserved.

Processor Performance Increase

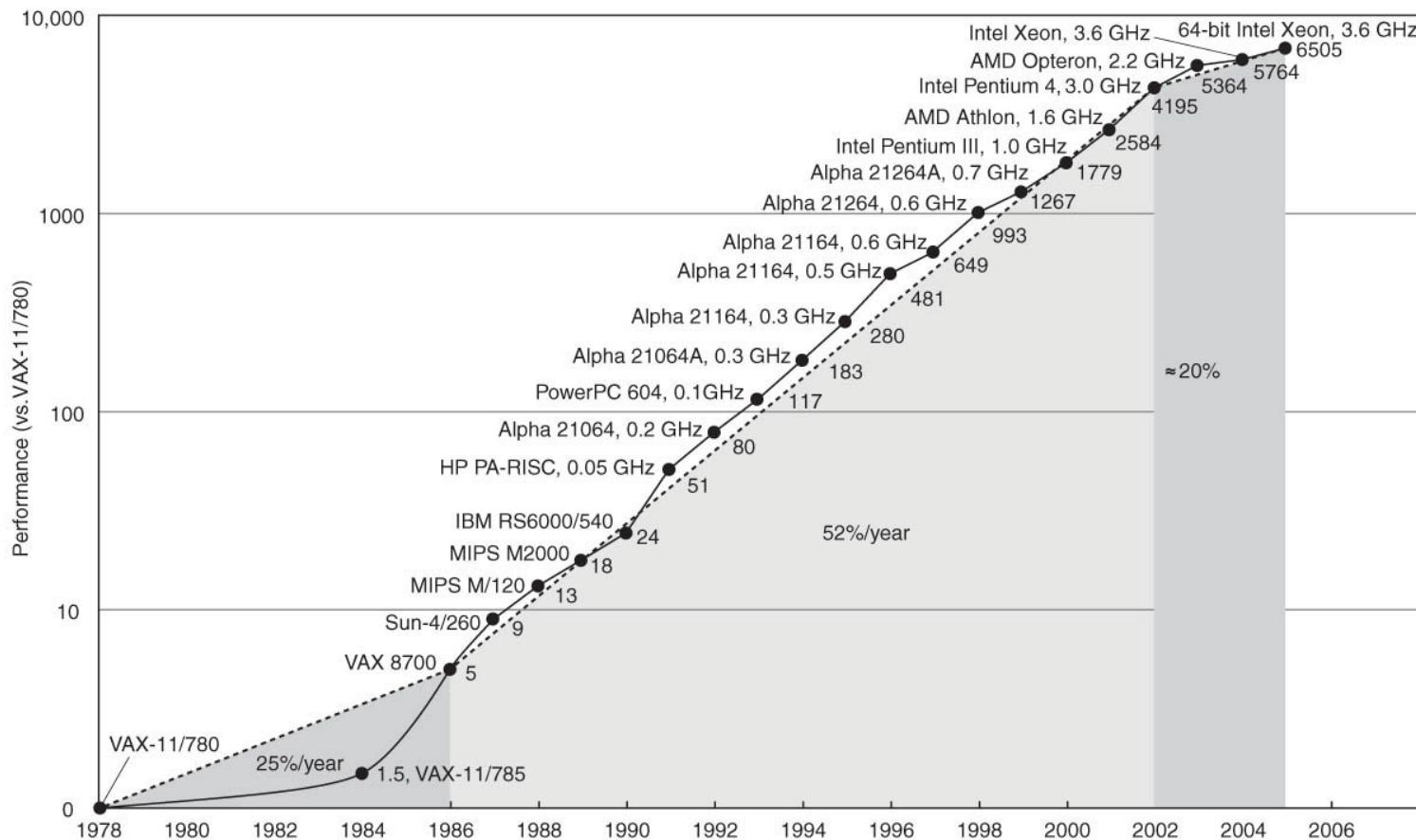
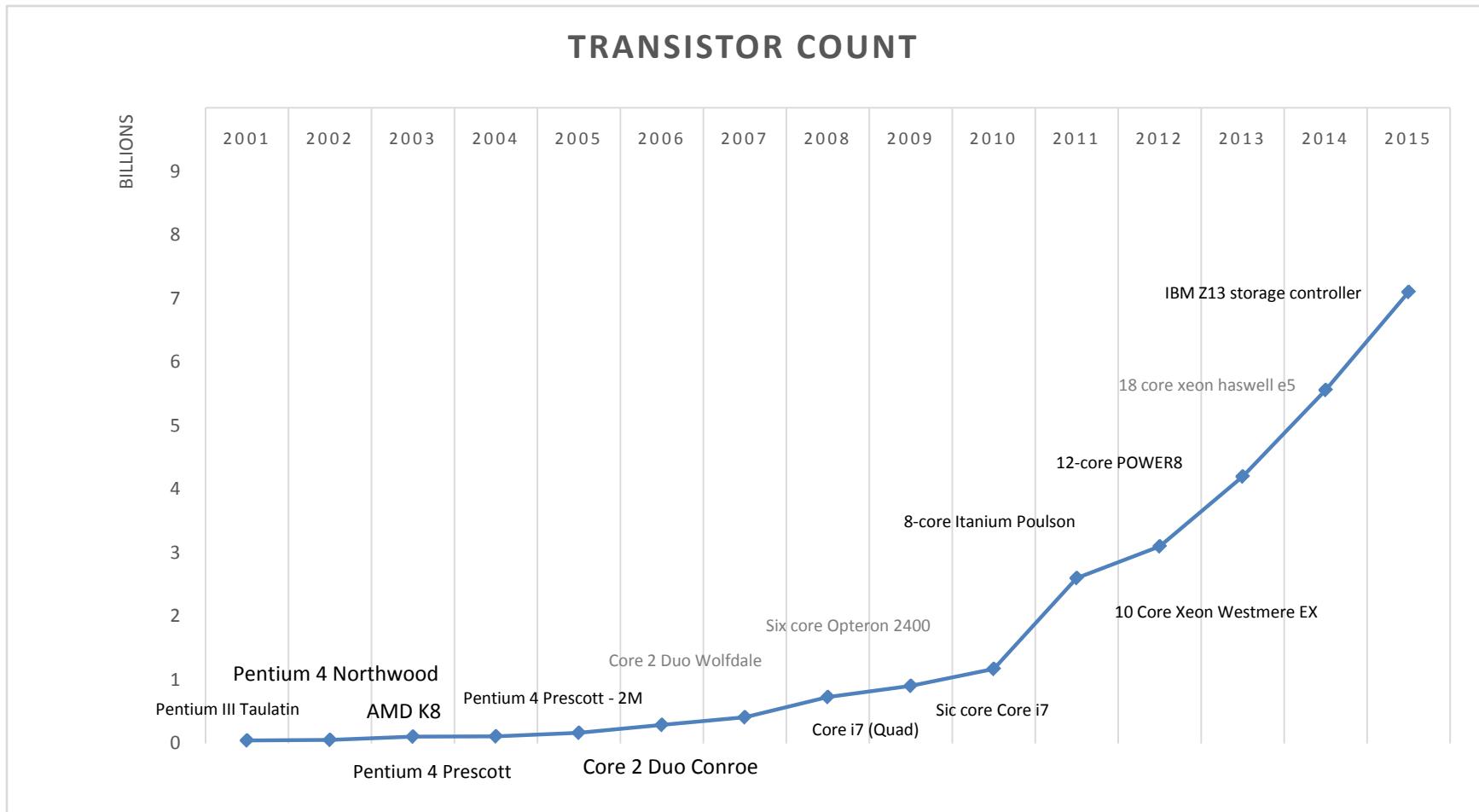


FIGURE 1.16 Growth in processor performance since the mid-1980s. This chart plots performance relative to the VAX 11/780 as measured by the SPECint benchmarks (see Section 1.8). Copyright © 2009 Elsevier, Inc. All rights reserved.

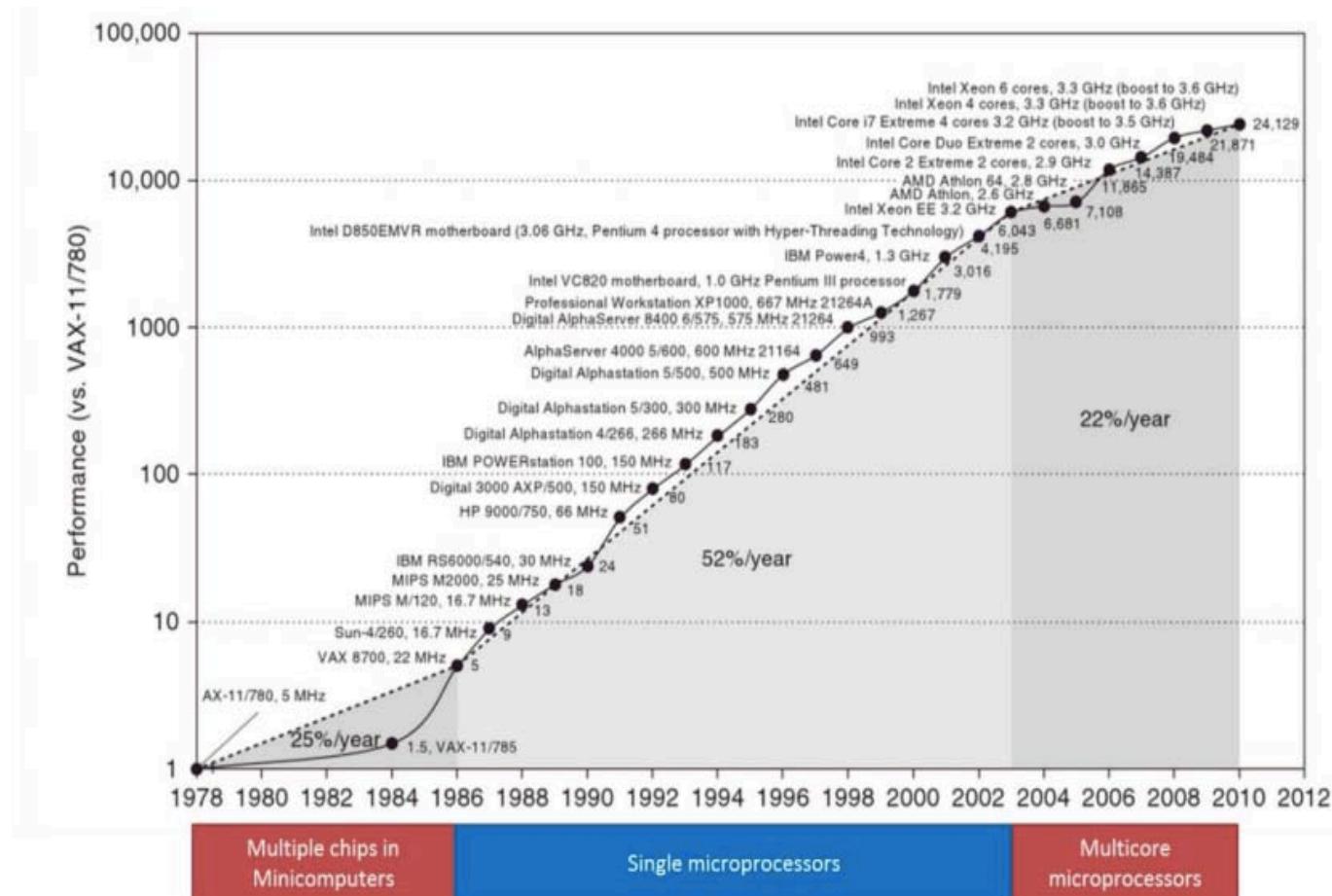
Transistor Count

KI angkatan 2015



Performance

KI angkatan 2015



Source: Computer Architecture, A Quantitative Approach by Hennessy and Patterson

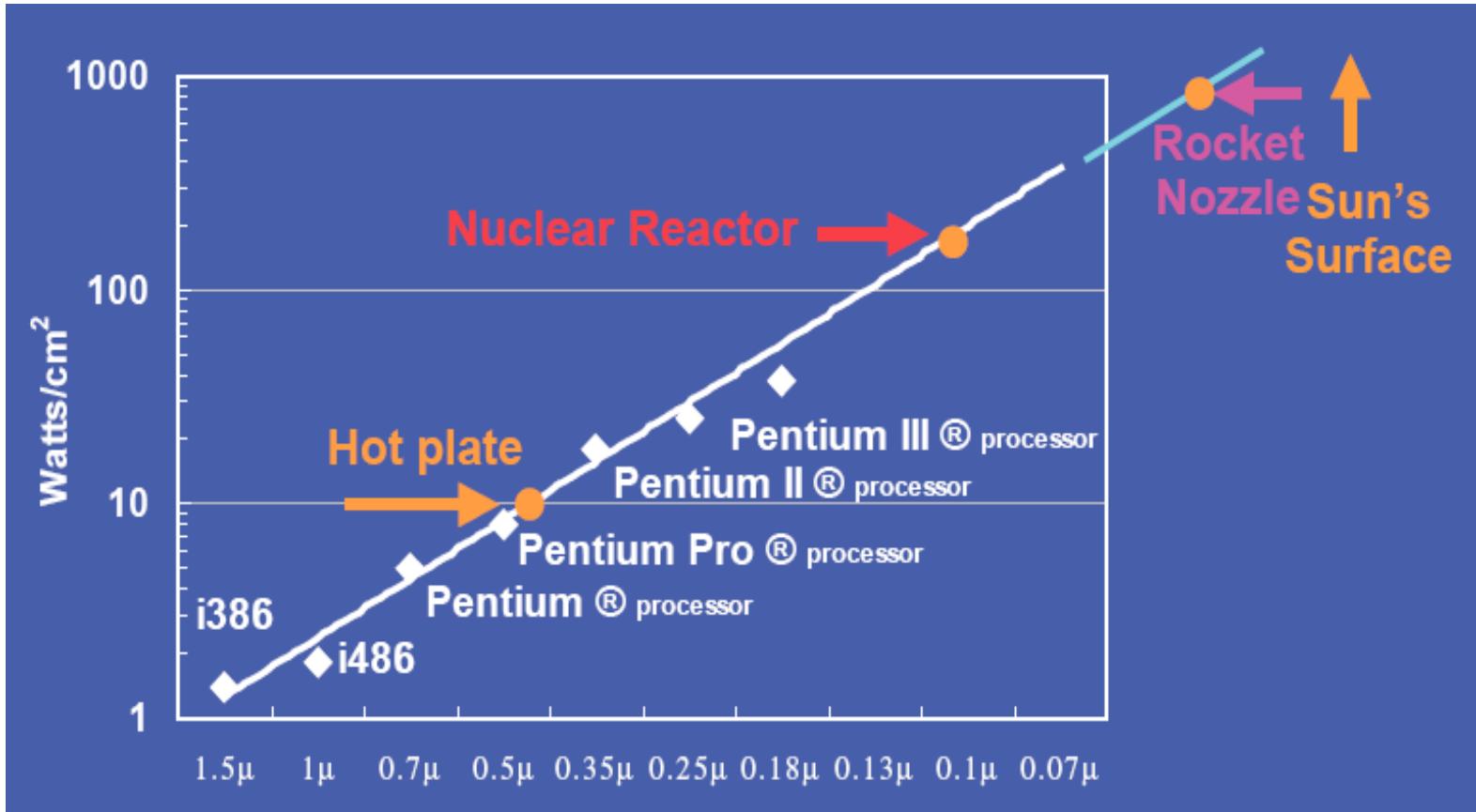
Performance Growth in Perspective

- Doubling every 2 years (1971-2005):
 - Total of 36,000X
 - If transportation industry matched this improvement, we could have traveled LA to NYC in about a second for roughly a few cents!

Ack: This is taken from Dr Tulika's materials.

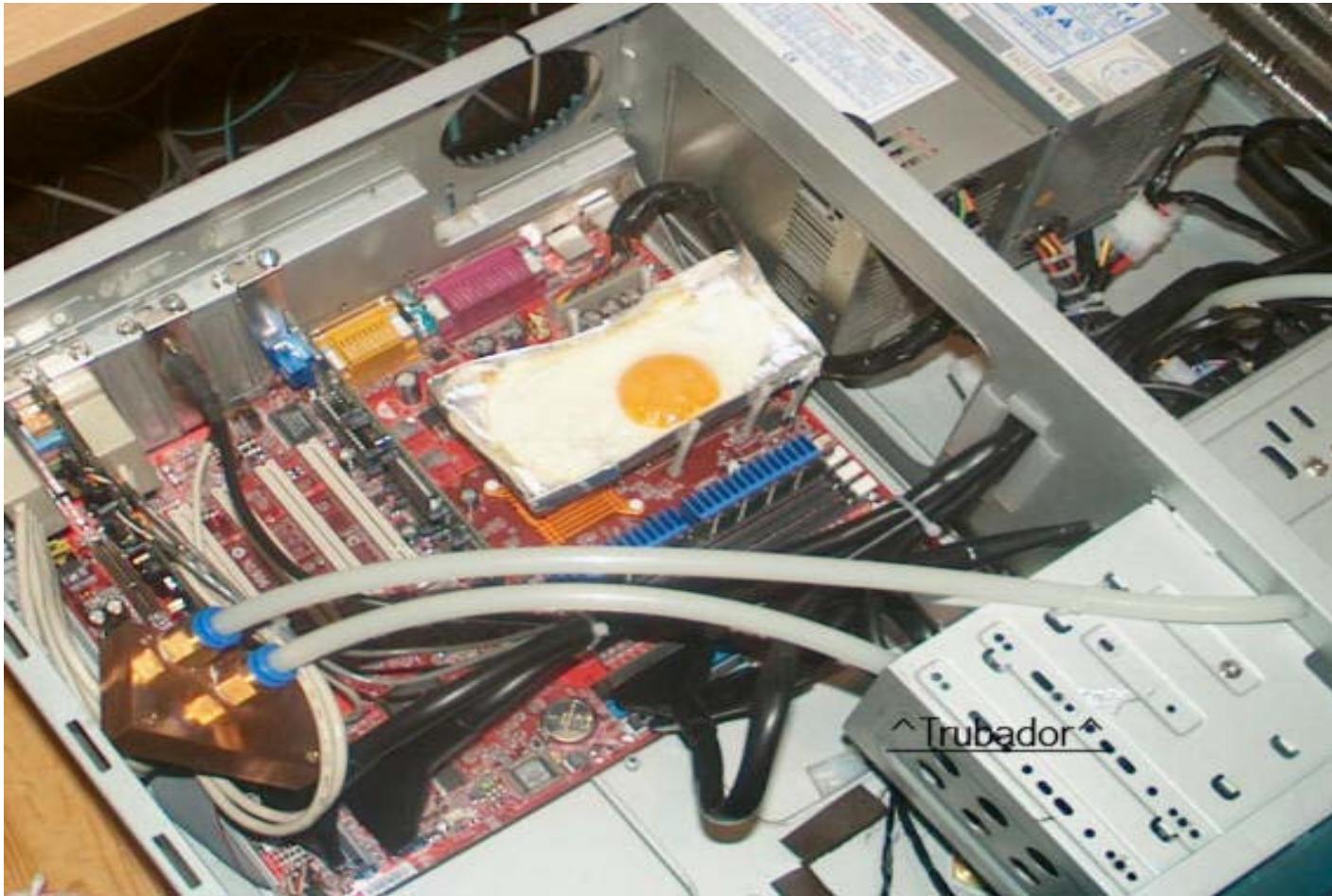
Roadblock: Power Density

- Bad for cooling, reliability, environment



Ack: This is taken from Dr Tulika's materials.

Breakfast Anyone?



Ack: This is taken from Dr Tulika's materials.