UF Herbert Wertheim College of Engineering UNIVERSITY of FLORIDA

Computer Architecture

Newman Waters
Fall 2025

EEL3701C - Digital Logic and Computer Systems

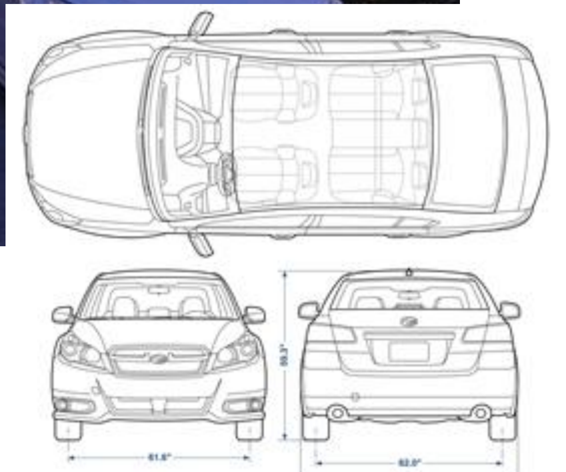
What do we mean by “architecture”?

Designing a car

- "Architecture"
  - Car has four wheels
  - Steering system
  - Power source

Can use a car w/o understanding internals

Conceptual



- Microarchitecture (blueprint)
  - Automatic/Manual
  - Disc brakes
  - 4 vs 6 cylinder engine

Can design a car w/o knowing exactly how ppl use it

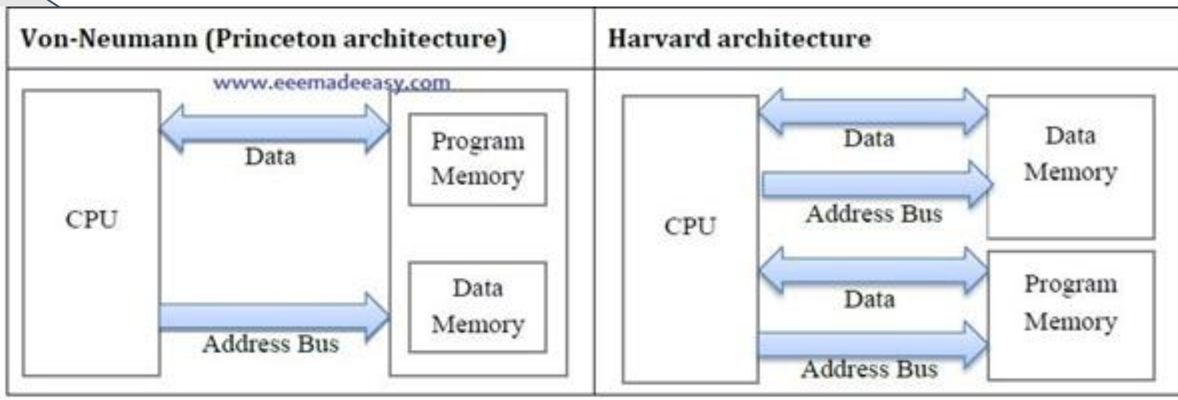
- Realization (physical implementation, materials)
  - Steel body, rubber tires

Physical

Designing a computer

- "Architecture"
  - What does a computer do?

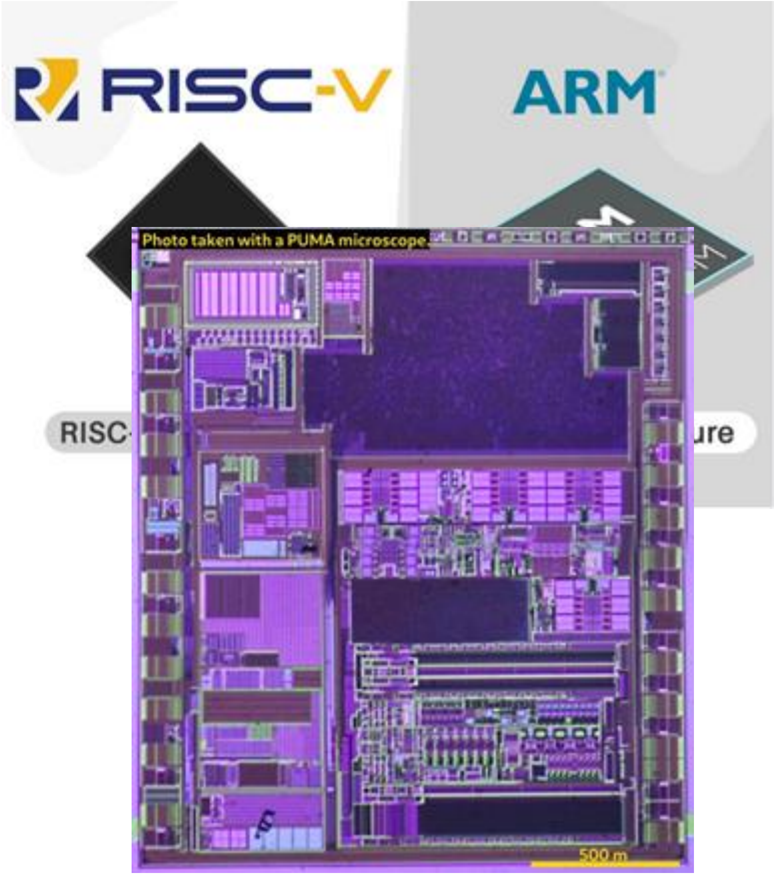
Conc



- Microarchitecture (blueprint)
  - ISA, datapath design

- Realization (physical implementation, materials)
  - CMOS, transistors

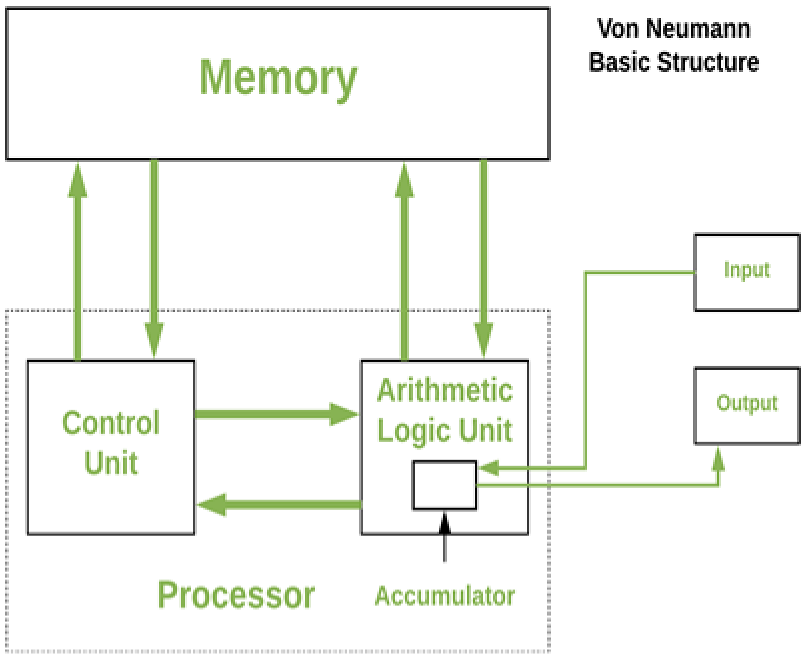
Physical

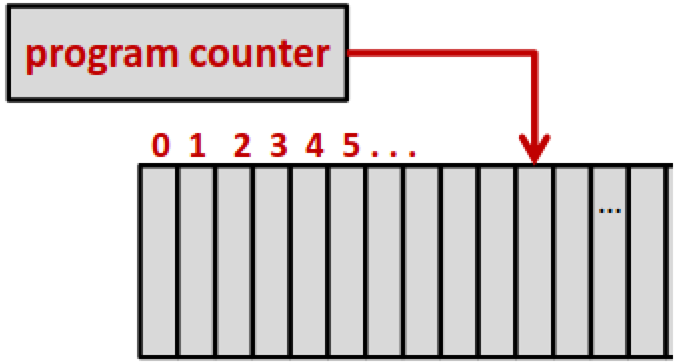


Stored Program Architecture (von Neumann)

John Von Neumann (Hungarian-American Scientist) derived the following architecture:

- Memory holds both the program you want to run and the data you need
- Sequential instruction processing
  - Program counter (PC) identifies the current instruction
  - Instruction fetched from memory
  - Update memory/increment PC (visible state)
  - repeat
  - https://tinycpu.com/coredumped

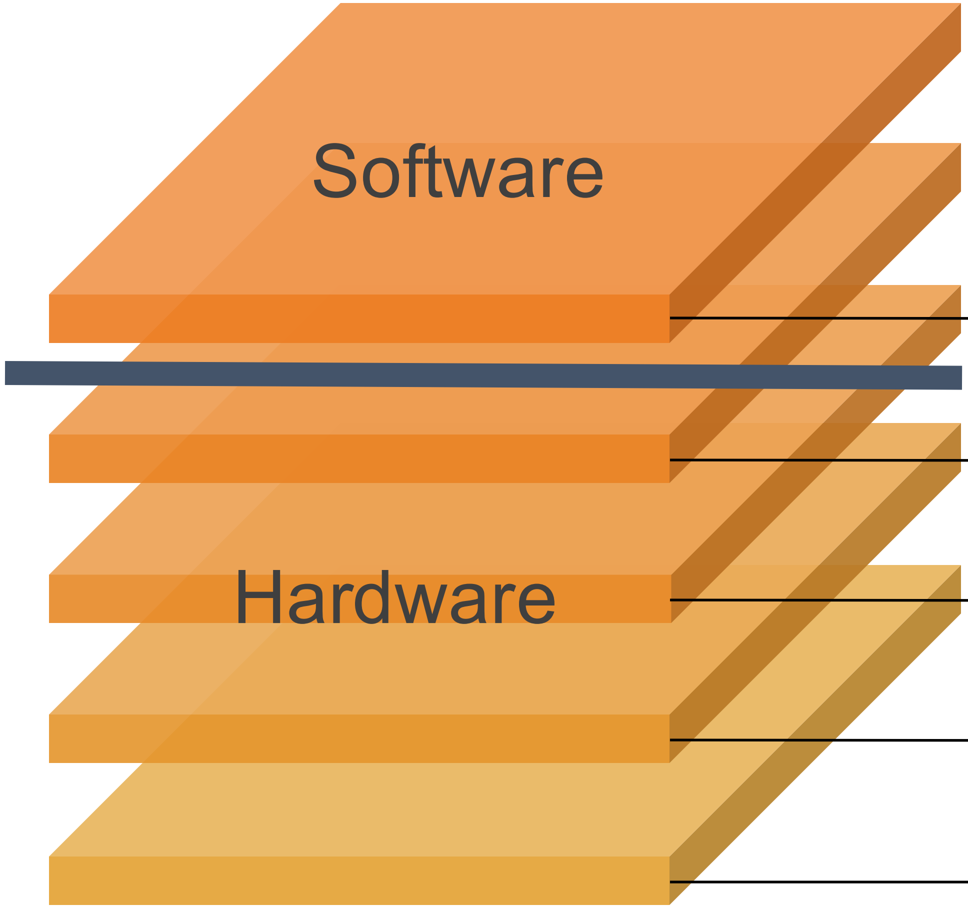




Herbert Wertheim College of Engineering
UNIVERSITY of FLORIDA

Instruction Set Architectures
The HW/SW Contract

Hardware / Software Stack



Instruction Set Architecture

Register-Transfer Level

Gate Level

Transistor Level

Silicon Geometry

ISAs in a nutshell

- Define what instructions exist on a processor, and which parts of the CPU they affect
- An interface that lets *any* program run on *any* chip that follows the same ISA
- Should last for a long time (x86 ISA invented in 1978 – 16 bit)
  - Compromise between optimization now for future scalability and compatibility
- von Neumann has been dominant paradigm, but Harvard architecture is used for certain applications requiring higher speed and less flexibility
- Atomic, sequential, and in-order instructions
- Different philosophies: RISC vs CISC

What is the main purpose of the ISA?

A. To define the physical layout of transistors in the processor

B. To describe how software communicates with hardware, defining the instructions and visible state

C. To specify the manufacturing process used to build the CPU

D. To describe the hardware design at the register-transfer level

Instruction Types

- Each ISA can have many different formats for instructions that do different things
- Like how the ALU has two different parts, instructions have completely different roles (and need different kinds of information)
- Different formats exist because **different operations care about different kinds of data** — registers, constants, or addresses — and each instruction only has a small number of bits (like 32)
  - The design tries to **fit what's needed**, without wasting space for flexibility

Example: MIPS ISA

MIPS is a very simple 32-bit ISA:

- **R-Type:** Register
  - add, sub, and

ONLY use registers

- **I-Type:** Immediate values/Memory access
  - addi, subi, sb, beq

Use one register and need an immediate value + dest. reg.

- **J-Type:** Jump
  - j, jal

Just needs a giant address to know where to go next

R–type	op	rs	rt	rd	shamt	funct
	Arithmetic instruction format					
I–type	op	rs	rt	address/immediate		
	Transfer, branch, immediate.					
J–type	op	target address				
	Jump instruction					
Field size	6 bits	5bits	5bits	5bits	5bits	6 bits

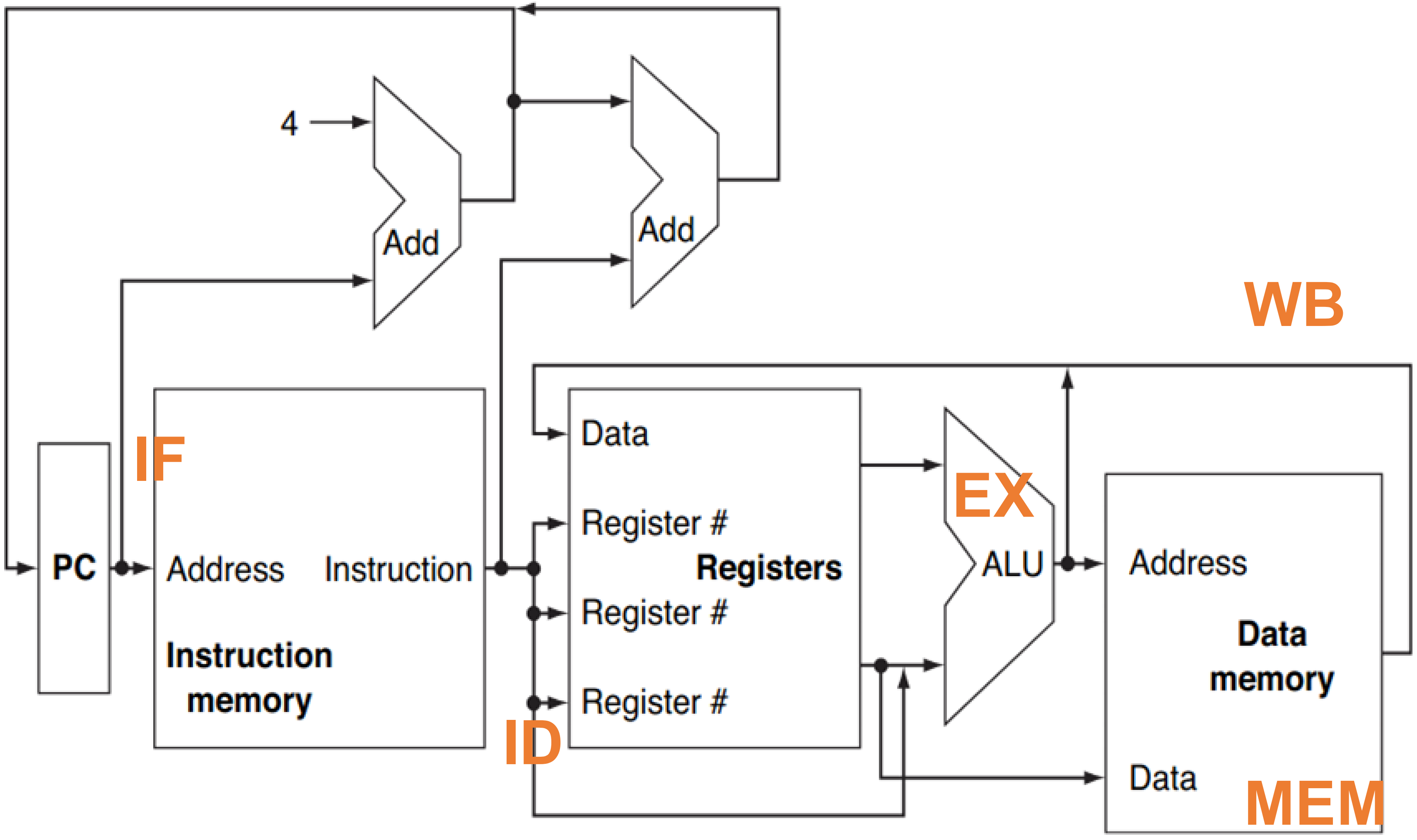
Using 32 bits to do many different actions!

Instruction Processing

Every instruction in MIPS is 32 bits (4 bytes) so program counter increments by 4.

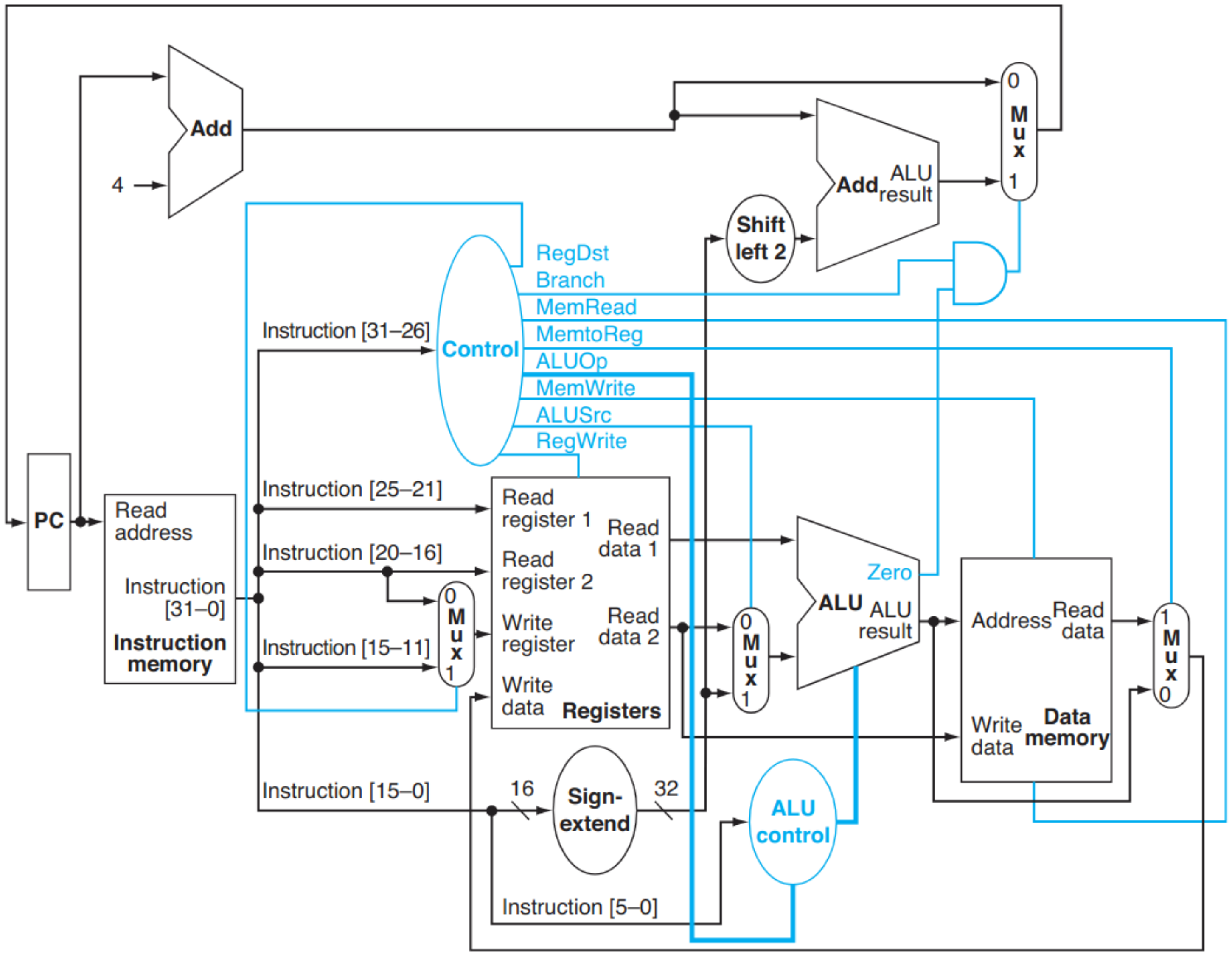
5 uniform steps regardless of instruction

- Instr. Fetch
- Instr. Decode /Register Fetch
- Execute
- Memory Access (not required by R/J types)
- Write-back



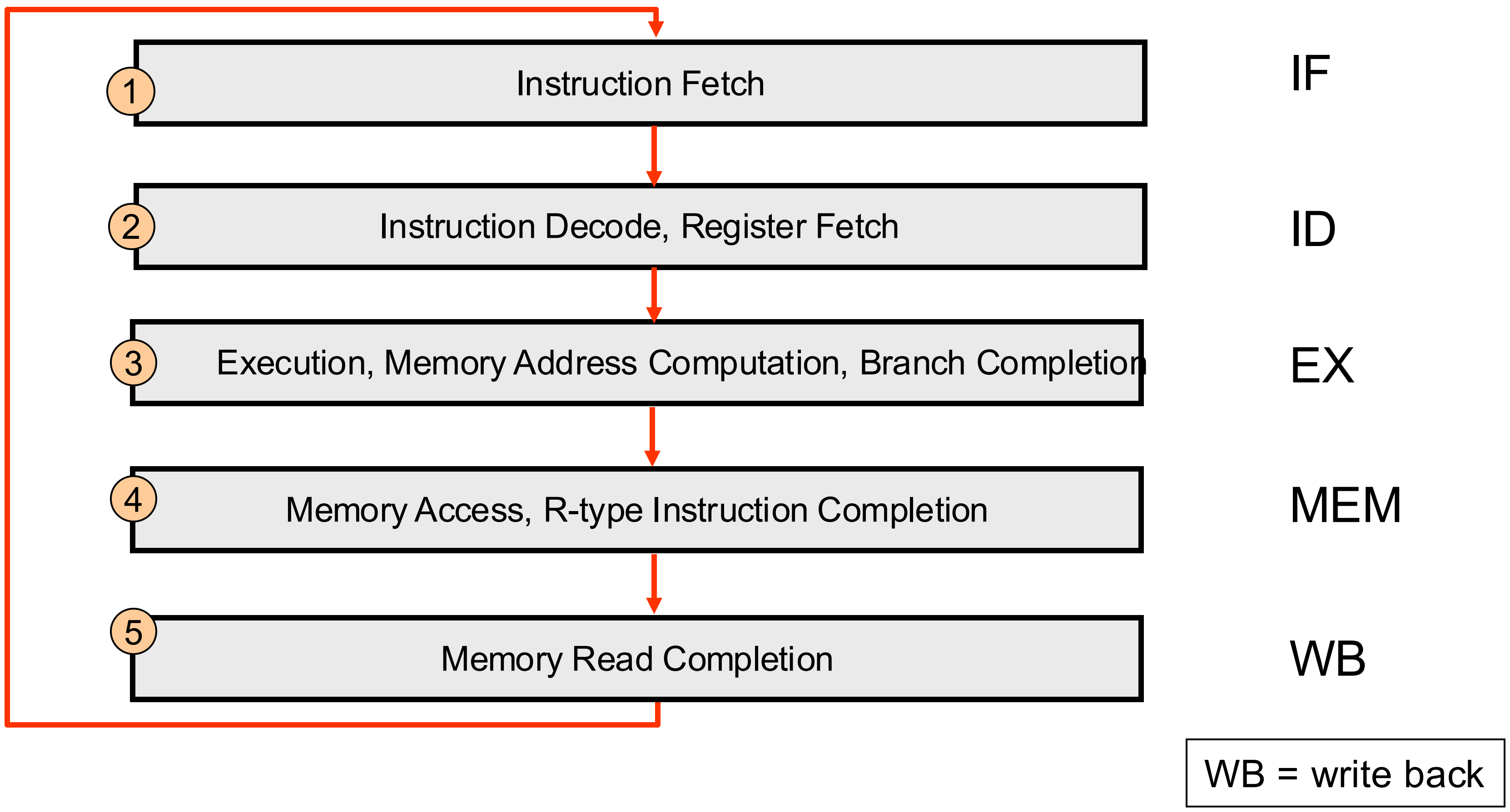
Note: Excludes all control signals

Control Signals with the Datapath

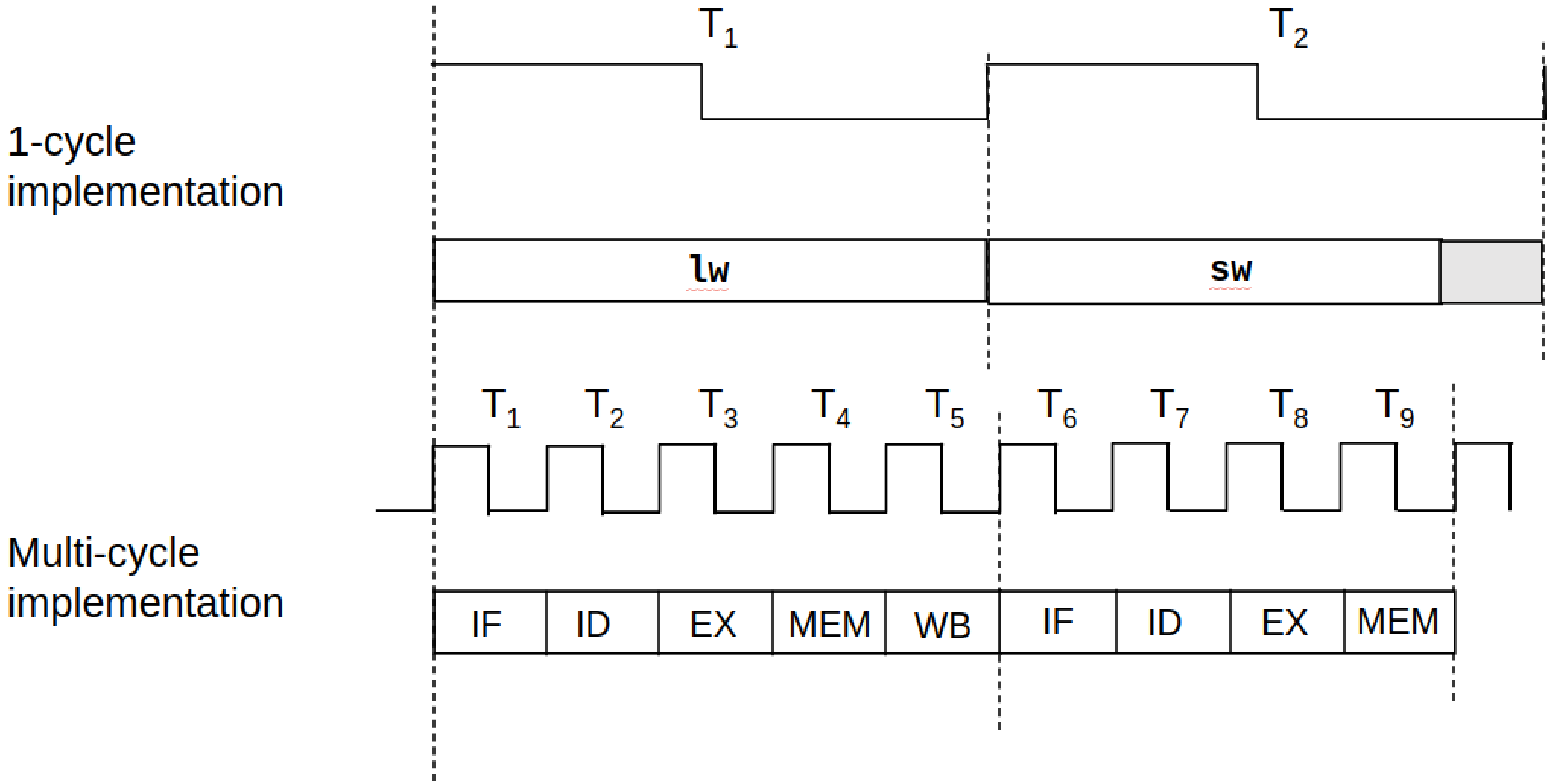


- You are building a controller for a more basic CPU in Lab 6
- If you take EEL4712C Digital Design, you will probably design this controller yourself
- If you read the ISA documentation, you can build anything!

Single-Cycle Datapath Summary



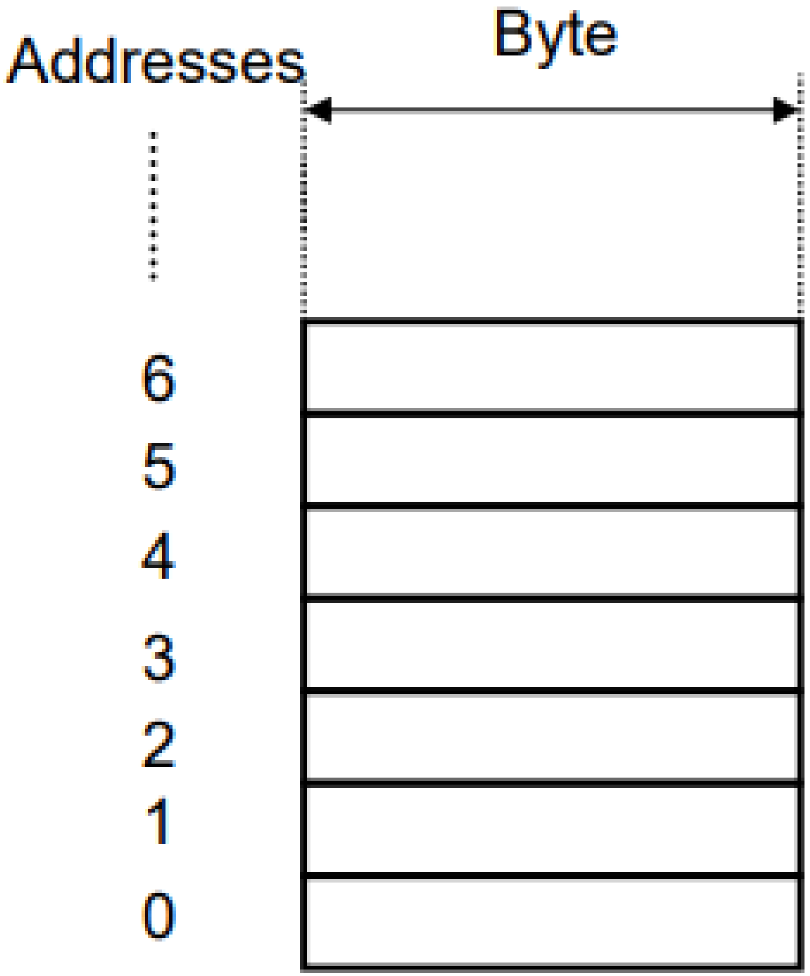
Single vs Multi-Cycle implementation



- Single cycle: each instruction takes one clock cycle to complete
- Multi-cycle: each stage takes one clock cycle

Memory Addressing

- Every 4 bytes (32 bits) contains a single instruction
- Program counter increments by 4 each instruction since a single address corresponds with a single byte
- BIG-ENDIAN!!!!! Most significant byte of the instruction is at the LOWER address!



How many core stages are there in the MIPS ISA?

Assembly Programming 101

- Breaking down high-level expressions into a sequence of atomic operations:
- E.g. C Code:

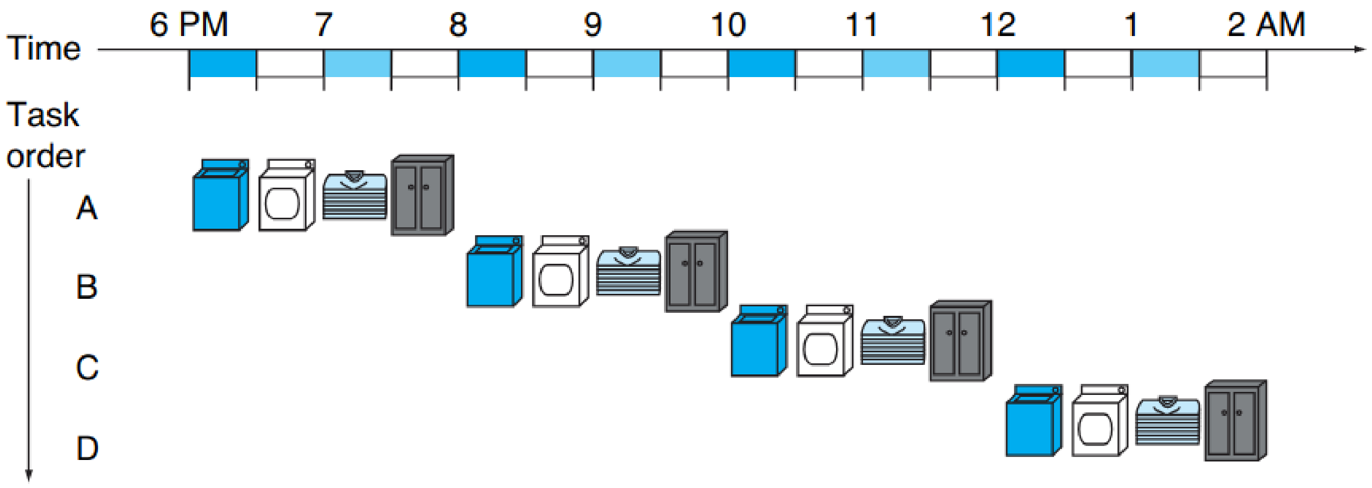
```c
int main() {
    int a = 5;
    int b = 10;
    int c = a + b;
    b = a + b;
    c += 5;
}
```

Say we only have 4 registers in our hardware, with a corresponding ISA that has basic instructions:

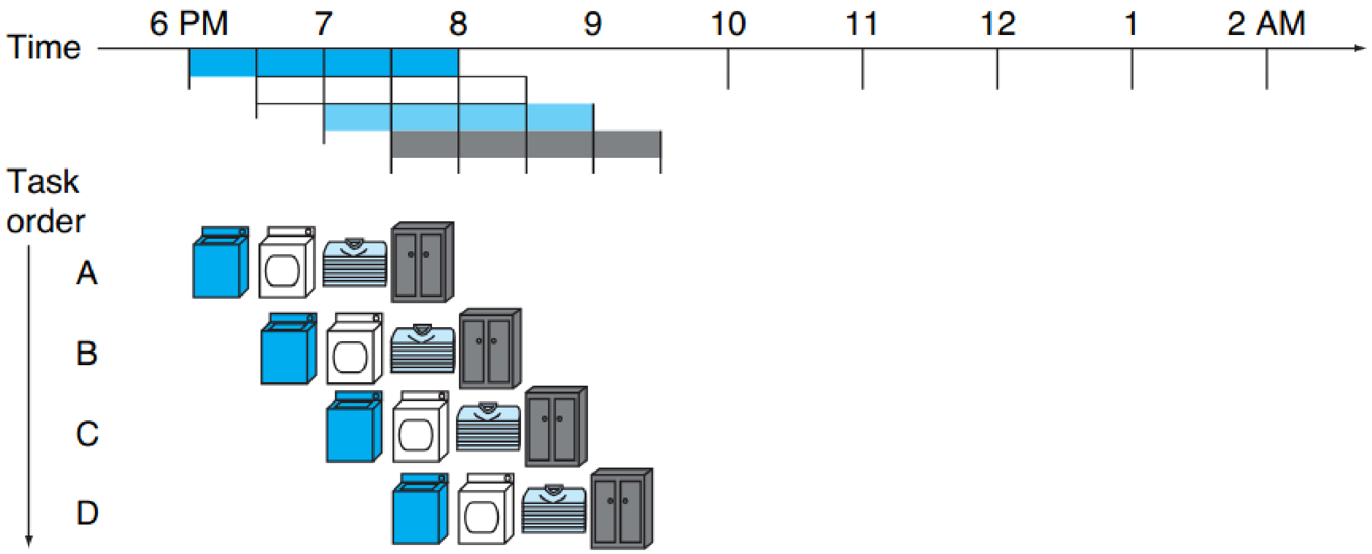
LDI, ADD, ADDI,

```
LDI R0, 5
LDI R1, 10
ADD R2, R0, R1
ADD R1, R0, R1
ADDI R2, 5
```

Pipelining

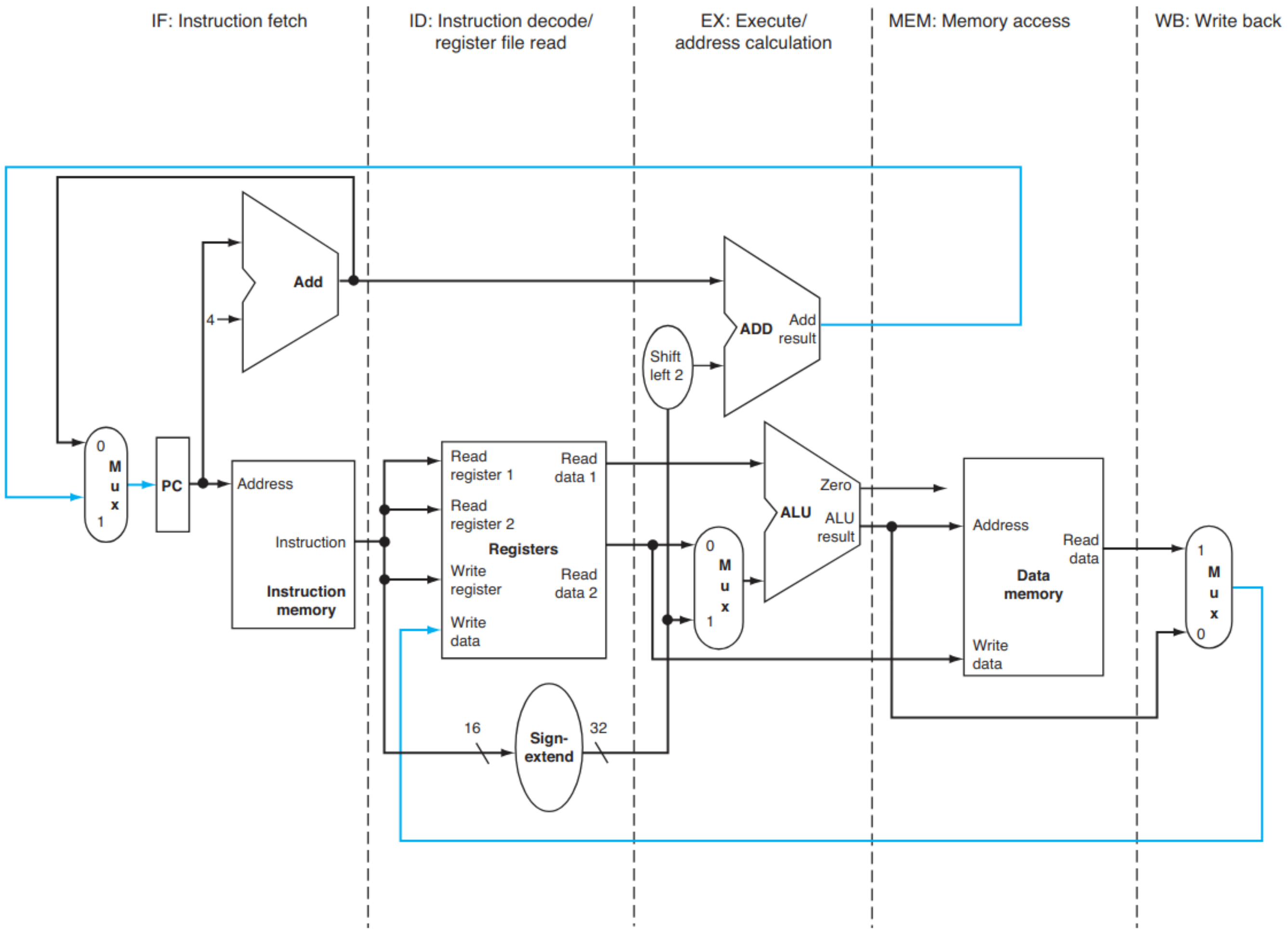


Time
6 PM
7
8
9
10
11
12
1
2 AM
Task order
A
B
C
D



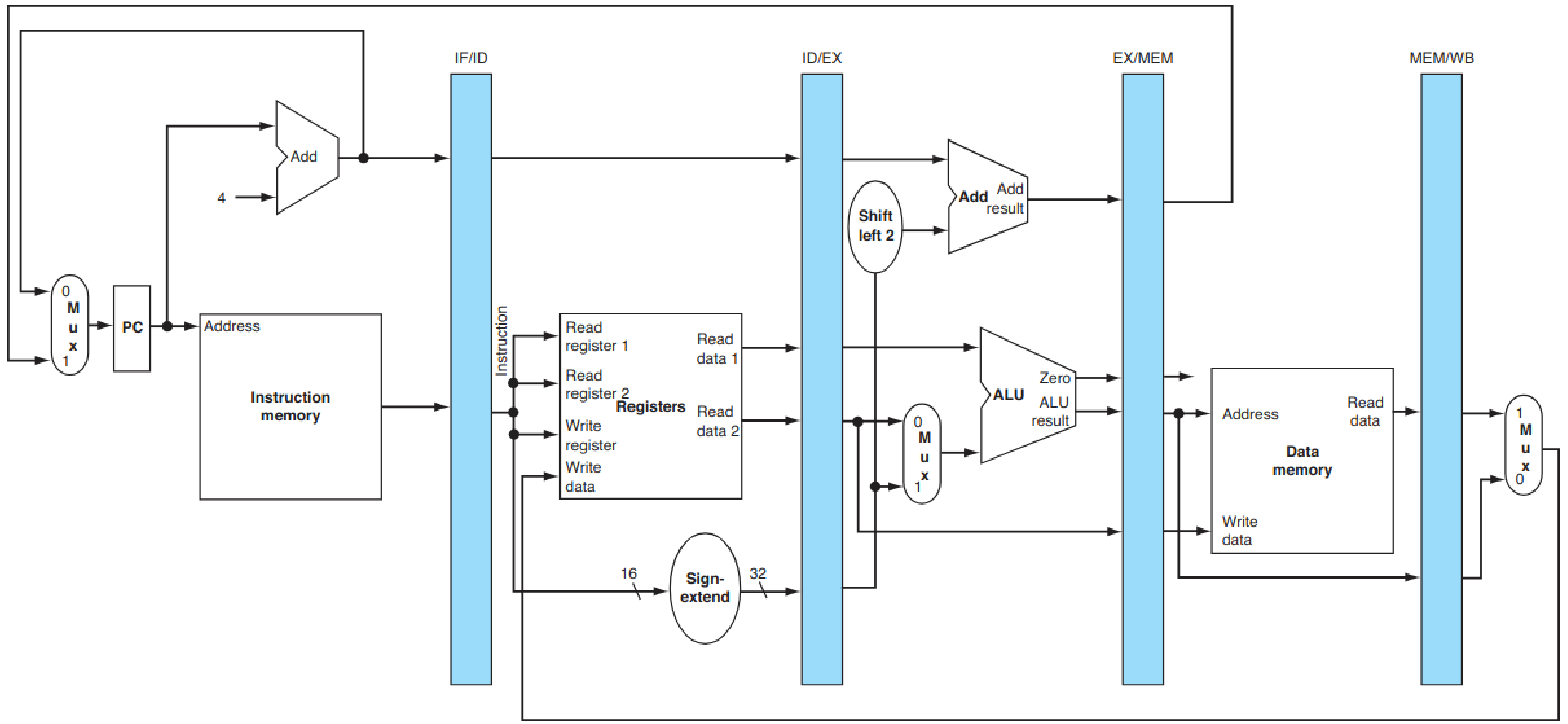
Time
6 PM
7
8
9
10
11
12
1
2 AM
Task order
A
B
C
D

Pipelining

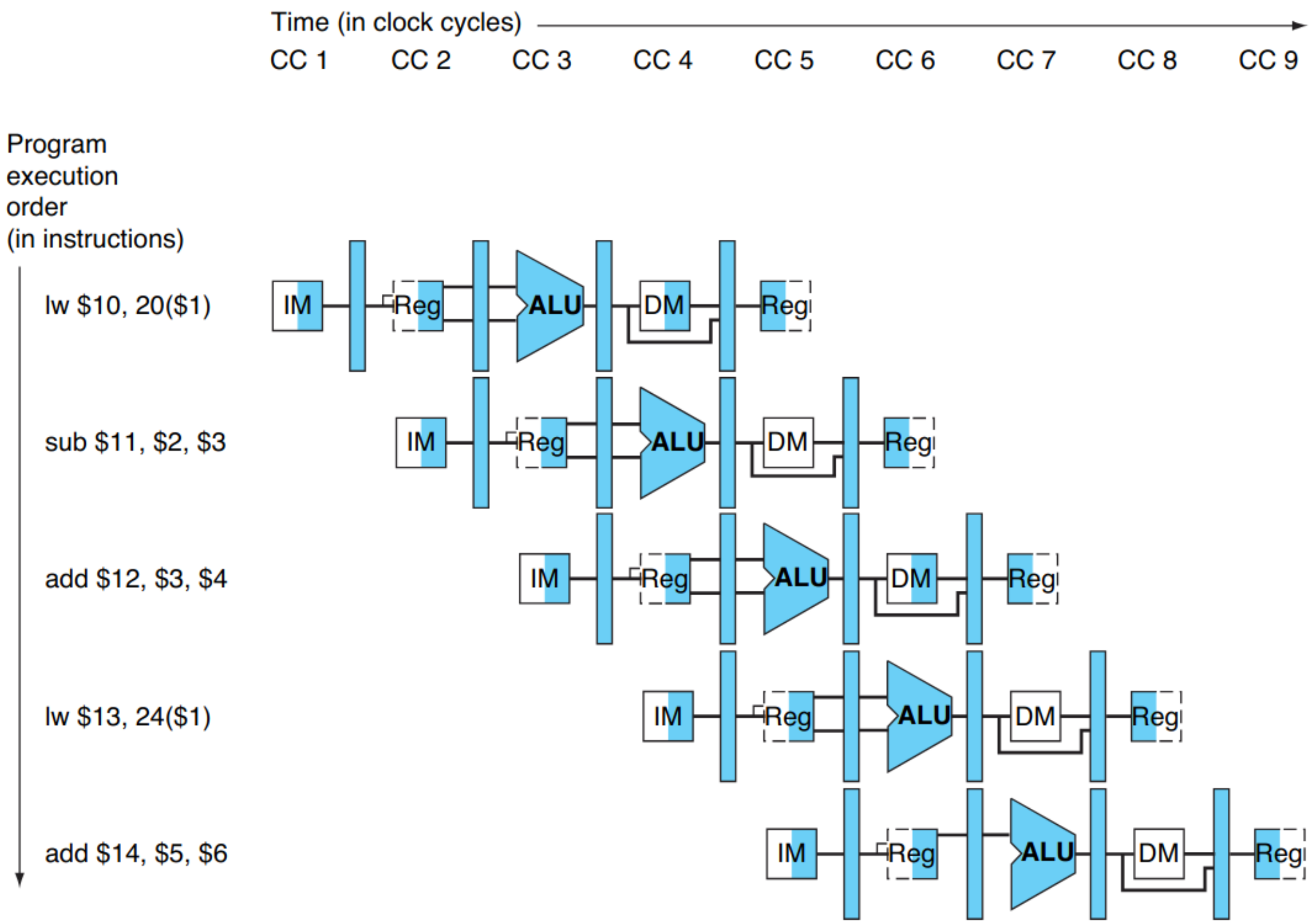


IF: Instruction fetch
ID: Instruction decode/
register file read
EX: Execute/
address calculation
MEM: Memory access
WB: Write back
Add
4
ADD
Add
result
Shift
left 2
0
M
u
x
1
PC
Address
Instruction
Instruction
memory
Read
register 1
Read
register 2
Registers
Write
register
Write
data
Read
data 1
Read
data 2
16
Sign-
extend
32
0
M
u
x
1
ALU
Zero
ALU
result
Address
Read
data
Data
memory
Write
data
1
M
u
x
0

Do multiple things at once!



Different stages happen at the same time



Data Hazards....

There are certain dangers to pipelining though

What happens if two resources are needed for a single instruction at the same time?

CPU blows up

Additional Resources

- Computer Organization and Design by Patterson and Hennessy
- TinyCPU
- Emugator by Christopher Tressler and team


UF
Herbert Wertheim
College of Engineering
UNIVERSITY of FLORIDA
LEADING THE CHARGE, CHARGING AHEAD