EE309(S2): Microprocessors

Spring 2025

[Week#8 Slides]

Instructor: Shalabh Gupta

Instruction Set Architecture (ISA)

Microprocessor architecture heavily depends on the ISA.

A good ISA would

- require less program memory (avoid less frequently used instructions)
- be easier to compile (less optimization required)
- involve easier processor design (smaller processor core) and circuit design
- provide faster program execution
- A small instruction set, leading to a more structured implementation

ISA depends on

- Operations supported by ALU: ADD, SUB, MULT, DIV etc.
- Location of operand values / results stored: Registers, Memory, Stack etc.
- Number of operands used by instructions (can be fixed or variable)
- Range of operands (ex. In 8051: Rn: R0-R7 Ri: R0 or R1)
- Size of an instruction: 8085 and 8051 use variable size instructions (1-3 bytes)

Instruction Classification

Based on (ALU) Operand / Result Storage

- Stack
- Accumulator (source is memory or register)
- General Purpose Register
- Register-Register (load-store) – MIPS
- Register-Memory – 80x86
- Memory-Memory (least efficient, generally not used)

Aren't the Registers placeholders for memory locations?

Operand Location

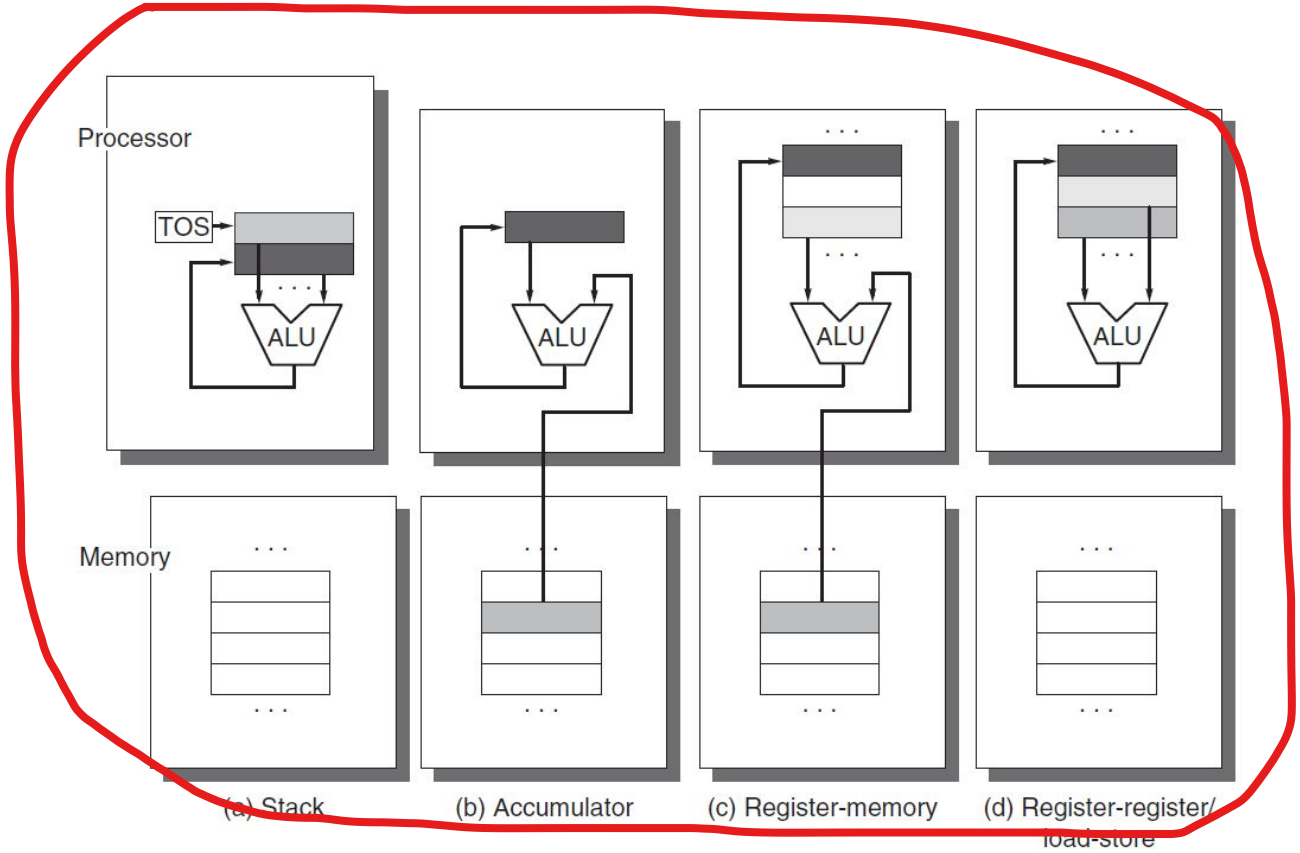


Figure A.1 **Operand locations for four instruction set architecture classes.** The arrows indicate whether the oper-

Courtesy: Computer Architecture: A Quantitative Approach, 5th Edition by David A. Patterson and John L. Hennessy (2012)

Example: C = A + B
(A, B and C are memory locations)

Stack	Accumulator	Register (register-memory)	Register (load-store)
Push A	Load A	Load R1,A	Load R1,A
Push B	Add B	Add R3,R1,B	Load R2,B
Add	Store C	Store R3,C	Add R3,R1,R2
Pop C			Store R3,C

Figure A.2 The code sequence for C = A + B for four classes of instruction sets. Note

Courtesy: Computer Architecture: A Quantitative Approach, 5th Edition by David A. Patterson and John L. Hennessy (2012)

Classification of ISA

Number of memory addresses	Maximum number of operands allowed	Type of architecture	Examples
0	3	Load-store	Alpha, ARM, MIPS, PowerPC, SPARC, SuperH, TM32
1	2	Register-memory	IBM 360/370, Intel 80x86, Motorola 68000, TI TMS320C54x
2	2	Memory-memory	VAX (also has three-operand formats)
3	3	Memory-memory	VAX (also has two-operand formats)

Architecture	Advantages	Disadvantages
Load–Store	Fixed length instructions, constant CPI => faster hardware	Longer programs
Register–Memory	Shorter programs (separate load not required)	Instruction length and CPI not fixed, Source variable destroyed
Memory–Memory	Most compact code	Large variation in instruction length, CPI, Memory become bottleneck

Courtesy: Computer Architecture: A Quantitative Approach, 5th Edition by David A. Patterson and John L. Hennessy (2012)

Memory Management Unit (MMU)

MMU helps in

- Translation of address used by the program (which may not be same as on the physical memory)
- Memory protection - preventing programs from accessing each others memory locations
- Paging and segmentation
- Cache control

Memory Controller:

Responsible for managing flow of data between the memory and the CPU

Different Storage Components

Storage occupies a lot of space and it is difficult to manage addresses

Registers: Typically up to 256 Bytes (can be 1KB or more in high-performance/specialized processors)

Cache (used for storing data from Memory that is frequently used - with address mapping)

- L1 Cache: 16KB - 256 KB
- L2 Cache: 256 KB - 8MB
- L3 Cache (shared between multiple cores -- on chip or off chip): 8MB - 64MB

Physical Memory or RAM (Random Access Memory) -- independent chip(s): A few GBytes to 100s of GBytes

Secondary Memory: Hard drives (magnetic storage), SSD (solid-state drive/flash memory), Optical storage (CD/DVD)

Virtual Memory

- Extend physical memory (RAM) by using secondary memory
- Extends the address space
- Can use paging and segmentation to shorten address space
- Less frequently used data swapped between physical memory and secondary memory
- Helps in memory protection, isolation
- Can lead to performance degradation

MIPS ISA

MIPS: Microprocessor without Interlocked Pipeline Stages

–Company by this name used to license the technology

–Load-Store architecture for RISC (Reduced Inst. Set Computer) based systems

MIPS ISA (basic version)

–Fixed Instruction Width (32-bits)

–32-bit addresses (2^{30} word memory)

–4-byte words

–32 Registers (31– General Purpose, R0=0).

–16 double-precision or 32 single precision GPRs GPRs = General Purpose Resistors

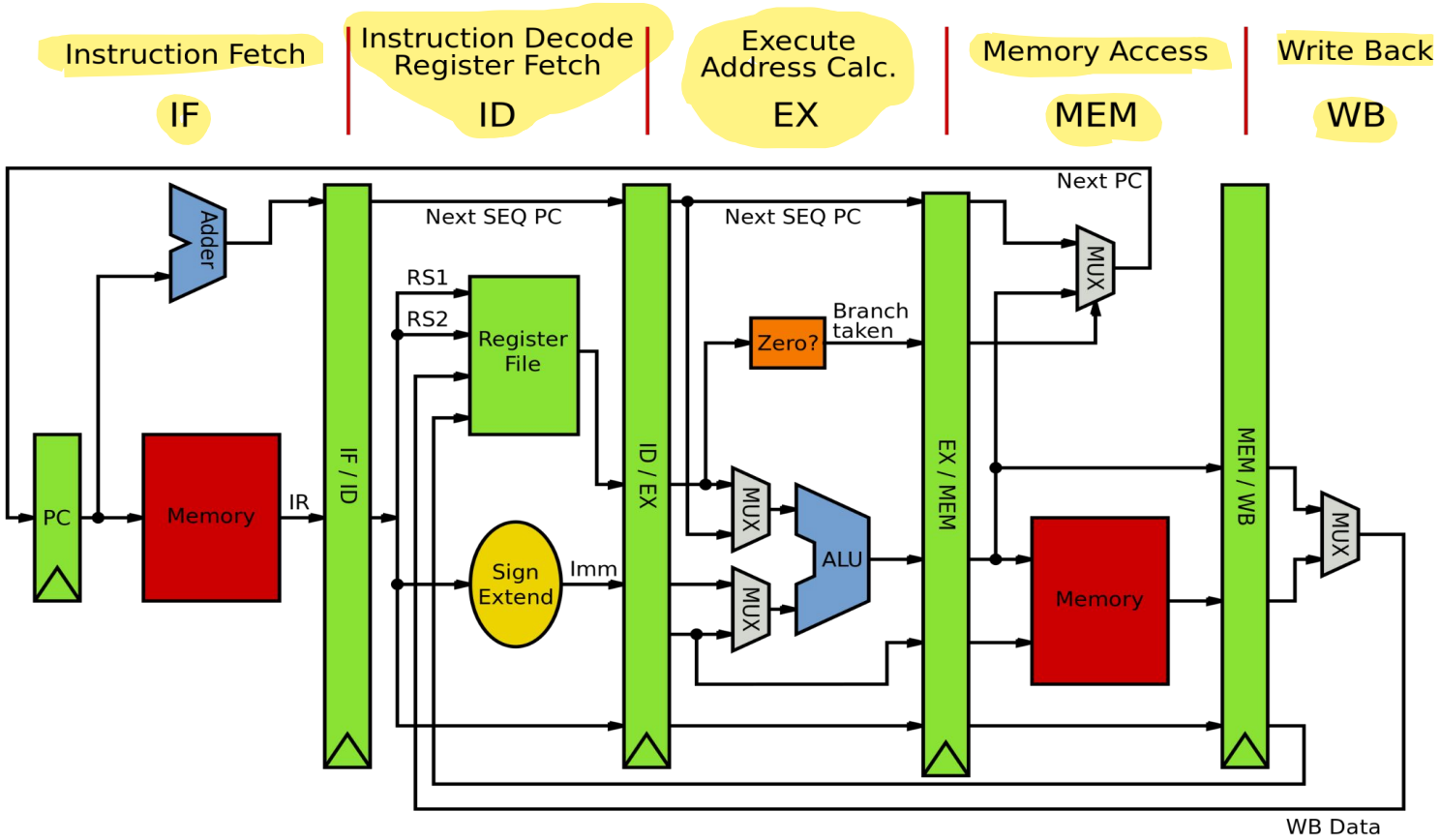
MIPS Instructions

Three types of Instructions (each instruction is of 32-bits)

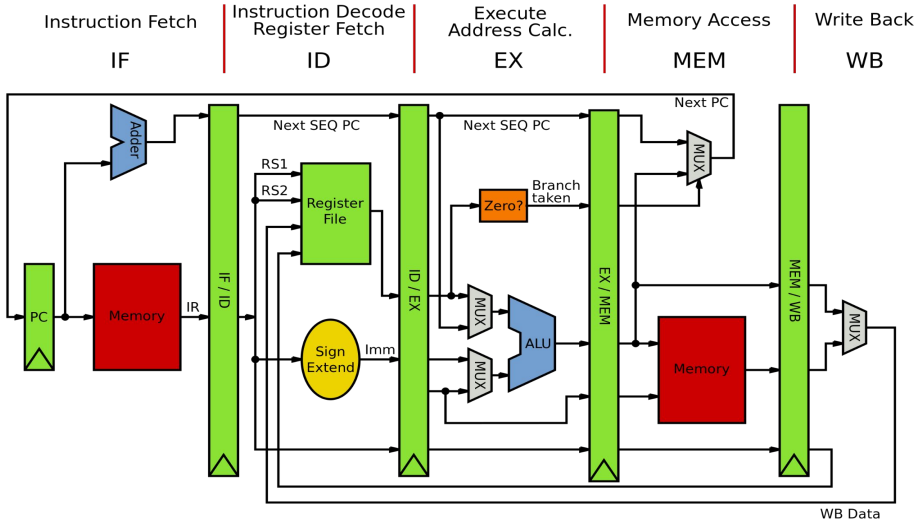
- R-type (all Register operands)
- I-type (Immediate data)
- J-type (Jump)

(Bits) R – Type	(31-26) Opcode	(25-21) rs	(20-16) rt	(15-11) rd	(10-6) shift amount	(5-0) function
(Bits) I – Type	(31-26) Opcode	(25-21) rs	(20-16) rt	(15-0) Immediate		
(Bits) J –Type	(31-26) Opcode	(25-0) Psuedodirect jump address				

MIPS Pipelined Architecture



MIPS Instruction Pipeline



	Clock number								
Instruction number	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
Instruction *i*	IF	ID	EX	MEM	WB				
Instruction *i* + 1		IF	ID	EX	MEM	WB			
Instruction *i* + 2			IF	ID	EX	MEM	WB		
Instruction *i* + 3				IF	ID	EX	MEM	WB	
Instruction *i* + 4					IF	ID	EX	MEM	WB