CPE 343
Computer Organization & Architecture

Dr. Muhammad Naeem Awais

Room # 8, B-Block

COMSATS University Islamabad, Lahore Campus

Agenda

- Introduction to the Course
- Course Description File
- Difference between Computer Organization and Computer Architecture
- Functional units
- History of Computers
- Evolution of Computer Systems
- Computer Types
- Instruction Set Architecture

Course Description File (CDF)

1	Course Title	Computer Organization & Architecture
2	Course Code	CPE343
3	Credit Hours	4 (3,1)
4	Prerequisites	DLD
5	Semester	Fall 2024
6	Resource Person/Lab Engineer	Dr. Muhammad Naeem Awais/Engr. Mr. Moazzam Ali Sahi
7	Contact Hours (Theory)	3 hours per week
8	Contact Hours (Lab)	3 hours per week
9	Office Hours	10:00 - 16:30 (weekdays)

Course Contents

- Introduction to CPE343-Computer Organization & Architecture, Difference between computer architecture and computer organization, Brief history of computer evolution, Classes of computers, Basic Terminology, Functional Units of Computer, Bus Structure in CPU, Classification of Instructions, Instruction Set Architecture
- Number Systems, Conversion among number systems, Signed number representation, Addition, Operations on integers: Addition and subtraction, Hardware for Addition and subtraction, Multiplication: Multiplication Algorithm, Division: Division Algorithm, Hardware for Division, Multiplying Negative Numbers, Booth's Algorithm, Signed Division, Floating Point Representation, Floating Point Arithmetic: Addition, Subtaction, Multiplication, Division
- MIPS Architecture, MIPS R3000 ISA, Registers in MIPS Processor, Introduction to MIPS Programming, MIPS Programmer’s Model, MIPS-Register File, Assembly Variables: Registers, Comments in Assembly, Assembly Instructions: MIPS Addition and Subtraction Instructions, Memory Addressing, Addressing Modes, Load from Memory Instruction, Store to Memory Instruction, Variable Array Index, Pointers vs Values, Control instructions: conditional branches, set on less than and procedures, Jump and link instruction, Stack, Storage management on call/return, Procedures, Leaf and Non-leaf Procedures, Assembly to Machine Conversion for R-type, I-type and J-type Instructions, Machine to Assembly Conversion
- Building the Processor incrementally: Building R-Type Datapath, Building I-Type Datapath, Building for Load/Store, Building for Addi Logic Designing and Building Datapath of MIPS Processor at Block Level, Building for BEQ, Building J-Type, Datapath Building Branch Datapath Integrating, Controlling operations in Processor, Building Main Control Unit and Control Unit for ALU, Multicycle MIPS architecture: datapath and its control unit, Different stage of multicycle, Single cycle vs Multicycle
- Unpipelined Work Flow, Pipelined Work Flow, Pipelining Characteristics, Pipelining in Circuits, Unpipelined RISC Datapath, Pipelined RISC Datapath, RISC MIPS Instruction Pipeline, 5 Steps of RISC Datapath, Pipeline Hazards & their elimination: Structural hazard (Memory & Register File), Data hazards (RAW, WAR, WAW), Control hazard
- Branch Prediction: Static Brach Prediction (Branch Taken, Branch Not-taken, Delayed Branch), Dynamic Brach Prediction (1-bit, 2-bit), Branch Prediction Buffer
- Multi-cycle Floating point (FP) MIPS pipeline, MIPS FP pipeline supporting multiple FP operation
- Cache Memory: Processor Memory Performance Gap, Relationship of Caches and Pipeline, Memory Hierarchy, CPU-Cache Interaction, General Organization of Cache, Addressing Cache, Types of Cache, Block Placement, Accessing Direct-Mapped Caches, Block Identification-Direct Mapped, Accessing Set Associative Caches, Block Identification-Set Associative, Block Identification-Fully Associative, Block Replacement, Write Strategy, Write allocation, Types of Cache Misses, Problems
- Performance of CPU, Impact of Cache on CPU performance
- Virtual Address, Virtual Address Translation into Physical Address, Page Table, Translation Lookaside Buffer
- Flynn’s taxonomy for computers, Multiprocessor Organization

Recommended Books

Textbook:

- Computer organization and design: The hardware/software interface by David A. Patterson & John L. Hennessy 5th edition

Reference Books:

- Computer organization and architecture designing for performance by William Stallings 8th edition
- Computer system organization and architecture by John D. Capinelli
- Computer Architecture – A quantitative approach by Hennessy and Patterson

Course Learning Outcomes (CLOs)

After successfully completing this course, the students will be able to:

- **Apply** suitable arithmetic algorithms to solve basic computer arithmetic using hardware, and **produce** assembly code for a high-level language code using MIPS instruction set to lay the foundations of a simplified processor design. (PLO2-C3)
- **Design** a datapath for single-cycle, multi-cycle and pipelined MIPS architecture using different hardware modules. (PLO3-C5)
- **Compute** the performance of computer programs, modern processors, the cache systems using terms related to various hardware/software constituents to highlight tradeoffs between different design choices. (PLO3-C3)
- **Follow** the procedure to assemble different parts of the MIPS 32-bit microprocessor in HDL and reproduce its response using a software tool and hardware platform. (PLO5-P3)

COMPUTER ORGANISATION & ARCHITECTURE

Computer Architecture

- Computer Architecture is a blueprint for design and implementation of a computer system. It provides the functional details and behavior of a computer system and comes before computer organization. Computer architecture deals with 'What to do?'

Computer Organization

- Computer Organization is how operational parts of a computer system are linked together. It implements the provided computer architecture. Computer organization deals with 'How to do?'

Sr. No.	Key	Computer Architecture	Computer Organization
1	Purpose	Computer architecture explains what a computer should do.	Computer organization explains how a computer works.
2	Target	Computer architecture provides functional behavior of computer system.	Computer organization provides structural relationships between parts of computer system.
3	Design	Computer architecture deals with high level design.	Computer organization deals with low level design.
4	Actors	Actors in Computer architecture are hardware parts.	Actor in computer organizaton is performance.
5	Order	Computer architecture is designed first.	Computer organization is started after finalizing computer architecture.

Computing Systems

Computers have two kinds of components:

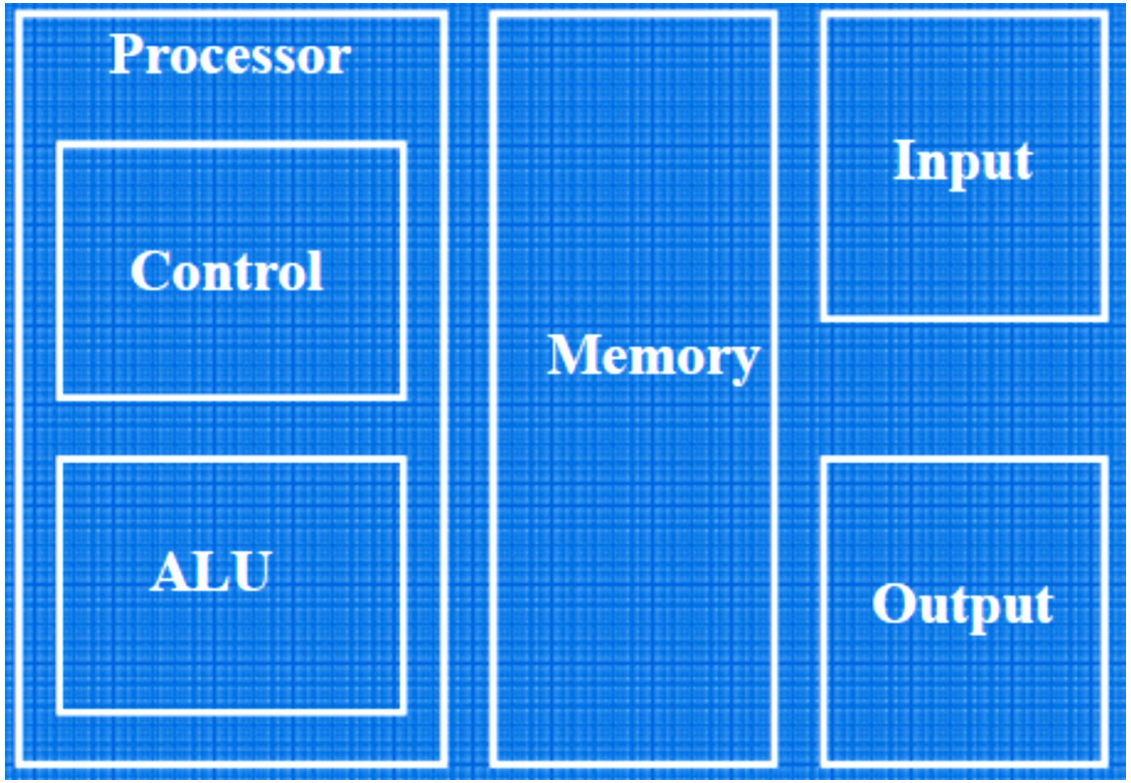
- *Hardware*, consisting of its physical devices (CPU, memory, bus, storage devices, ...)

- *Software*, consisting of the programs it has (Operating system, applications, utilities, ...)

Basic Units of Computer

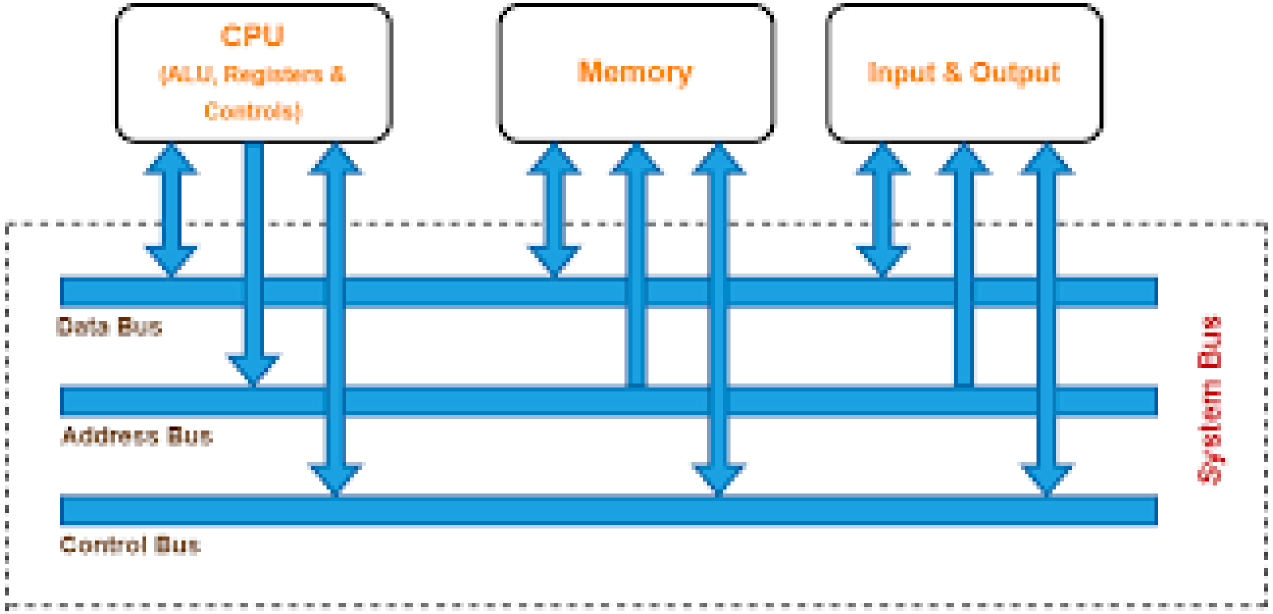
- Input Unit
- Output Unit
- Central processing Unit (ALU and Control Units)
- Memory
- Bus Structure

The Big Picture



Bus System

- The computer system bus is **the method by which data is communicated between all the internal pieces of a computer**. It connects the processor to the RAM, to the hard drive, to the video processor, to the I/O drives, and to all the other components of the computer.



INPUT UNIT

- Converts the external world data to a binary format, which can be understood by CPU
  - Mouse, Joystick etc

OUTPUT UNIT

- Converts the binary format data to a format that a common man can understand
  - Monitor, Printer, LCD, LED etc

CPU

- The “brain” of the machine
- Responsible for carrying out computational task
- Contains ALU, CU, Registers
- ALU performs Arithmetic and logical operations
- CU provides control signals in accordance with some timings which in turn controls the execution process
- Register stores data and result and speeds up the operation

MEMORY

- Stores data, results, programs
- Two class of storage

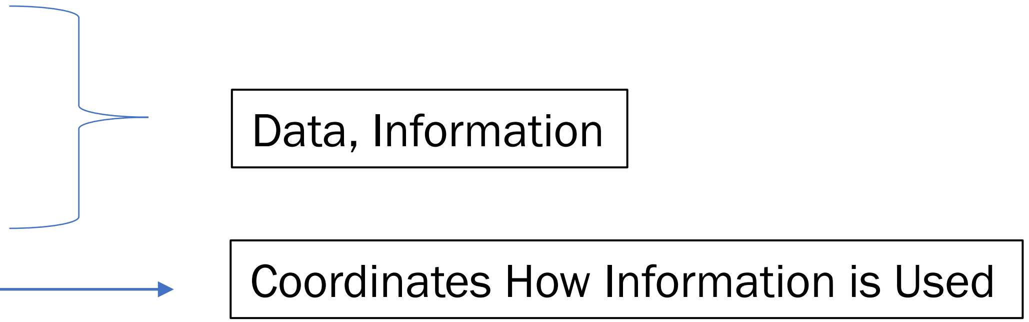
  (i) Primary (ii) Secondary
- Two types are RAM or R/W memory and ROM read only memory (Volatile Memory)
- ROM is used to store data and program which is not going to be changed (Non-Volatile Memory)
- Secondary storage is used for bulk storage or mass storage

Memory Operations

- Today, **general-purpose computers** use a set of instructions called a **program** to process data.
- A computer executes the program to create output data from input data
- Both program instructions and data operands are stored in memory
- Two basic operations requires in memory access
  - Load operation (Read or Fetch)-Contents of specified memory location are read by processor
  - Store operation (Write)-Data from the processor is stored in specified memory location

Function

- ALL computer functions are:
  - Data PROCESSING
  - Data STORAGE
  - Data MOVEMENT
  - CONTROL



Data, Information

Coordinates How Information is Used

- NOTHING ELSE!

Instruction Execution Cycle

- A computer instruction is an order given to a computer processor by a computer program. At the lowest level, each instruction is a sequence of 0s and 1s that describes a physical operation the computer is to perform.
- Although the inctruction cycle varies amongst CPUs, the stages of the instruction cycle are:
  - ➢Fetch the instruction.
  - ➢Decode the instruction.
  - ➢Execute the instruction.
  - ➢Memory access (if needed).
  - ➢Registry write-back (if needed).

Brief History of Computer Evolution

Two phases:

- Before VLSI 1945 –1978
  - ENIAC (Electronic Numerical Integrator And Computer)
  - IAS (Institute for Advanced Study)
  - IBM (International Business Machines)
  - PDP-8 (Programmed Data Processor)

- VLSI 1978 →present day
  - Microprocessors !

Brief History of Computer Evolution

- **ENIAC** (Electronic Numerical Integrator and Computer)

ENIAC
THE WORLD'S FIRST ELECTRONIC, LARGE SCALE, GENERAL-PURPOSE DIGITAL COMPUTER
2005/12/13 12:49 pm

100   K Hz Clock

17500 Vacuum Tubes

30 ton weight

Evolution of Computers

FIRST GENERATION (1945 –1955)

- Program and data reside in the same memory (stored program concepts –John von Neumann)
- ALP was made used to write programs
- Vacuum tubes were used to implement the functions (ALU & CU design)
- Magnetic core and magnetic tape storage devices are used
- Using electronic vacuum tubes, as the switching components

Evolution of Computers

SECOND GENERATION (1955 –1965)

- Transistor were used to design ALU & CU
- HLL is used (FORTRAN)
- To convert HLL to MLL compiler were used
- Separate I/O processor were developed to operate in parallel with CPU, thus improving the performance
- Invention of the transistor which was faster, smaller and required considerably less power to operate

Evolution of Computers

THIRD GENERATION (1965-1975)

- IC technology improved
- Improved IC technology helped in designing low-cost, high-speed processor and memory modules
- Multiprogramming, pipelining concepts were incorporated
- DOS allowed efficient and coordinate operation of computer system with multiple users
- Cache and virtual memory concepts were developed
- More than one circuit on a single silicon chip became available

Evolution of Computers

FOURTH GENERATION (1975-1985)

- CPU –Termed as microprocessor
- INTEL, MOTOROLA, TEXAS, NATIONAL semiconductors started developing microprocessor
- Workstations, microprocessor (PC) & Notebook computers were developed
- Interconnection of different computer for better communication LAN, MAN, WAN
- Computational speed increased by 1000 times
- Specialized processors like Digital Signal Processor were also developed

Evolution of Computers

BEYOND THE FOURTH GENERATION(1985 –TILL DATE)

- E-Commerce, E-banking, home office
- ARM, AMD, INTEL, MOTOROLA
- High speed processor -GHz speed
- Because of submicron IC technology lot of added features in small size

COMPUTER TYPES

Computers are classified based on the parameters like

- Speed of operation
- Cost
- Computational power
- Type of application

Classes of computers

- Personal mobile devices (PMDs)
- Desktop computers
- Servers
- Cluster/warehouse scale computers
- Embedded computers

Computer classes

- **Personal Mobile Devices (PMDs)**
  - Wireless devices with multimedia interface such as smart phones and tablet computers
  - Constraints
    - Cost
    - Energy efficiency
    - Size requirement
- **Desktop computing**
  - Low end note books to high end workstations, battery operated lap tops etc..
  - Constraints
    - Cost
    - performance

Computer classes

- **Servers**
  - Large scale and more reliable file computing services
  - Key features:

1)Availability,

2) Scalability,

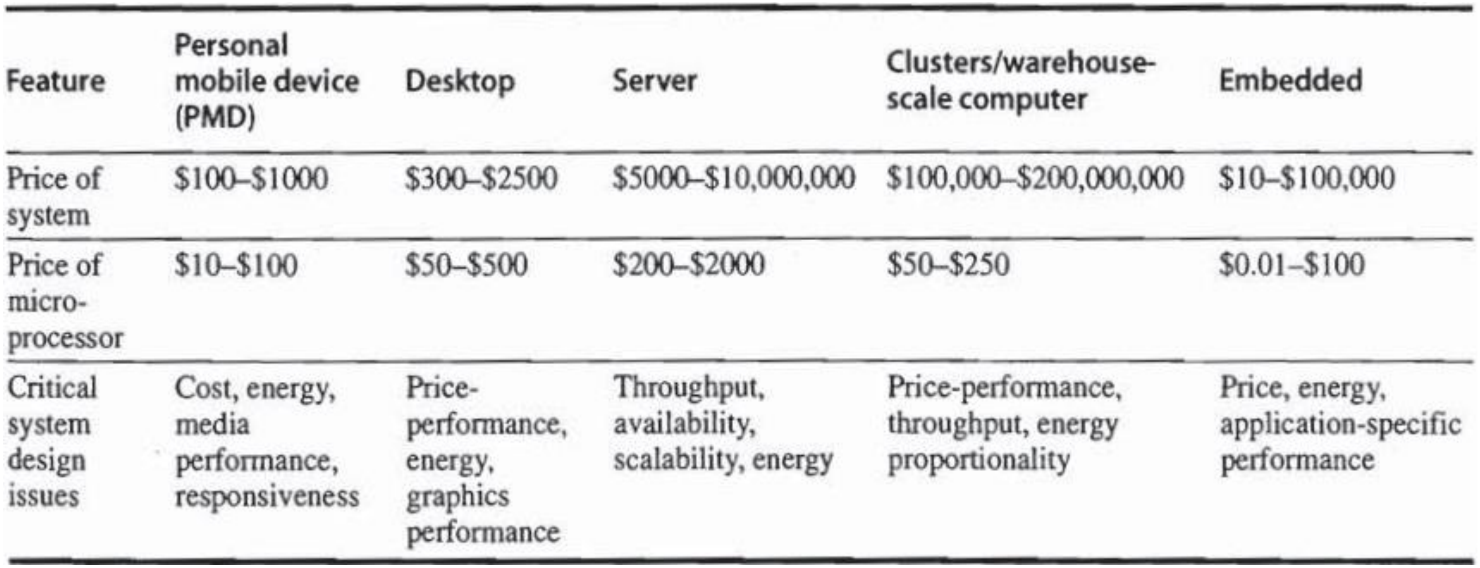
3) Throughput e.g. transactions per minute or results per minute, capability of handling number of requests per unit time

Computer classes

- **Clusters/Warehouse scale computers (WSC)**
  - Software as a Service (SaaS) like social networking, search, multiplayer games, video sharing etc has led to *clusters.*
  - Clusters are collections of desktop computers or servers connected by local area networks to act as a single larger computer.
  - Largest clusters are termed as warehouse scale computers
  - Critical factors
    - Price-performance
    - Power
    - Availability just like servers
    - Scalability through LANs
  - WSCs are as expensive as Super computers. But their objectives are different

Computer classes

- **Embedded computers**
  - Found in everyday machines: microwave ovens, washing machines, printers
  - Constraints
    - Cost
    - Performance but not as critical as cost



Feature	Personal mobile device (PMD)	Desktop	Server	Clusters/warehouse-scale computer	Embedded
Price of system	\$100–\$1000	\$300–\$2500	\$5000–\$10,000,000	\$100,000–\$200,000,000	\$10–\$100,000
Price of micro-processor	\$10–\$100	\$50–\$500	\$200–\$2000	\$50–\$250	\$0.01–\$100
Critical system design issues	Cost, energy, media performance, responsiveness	Price-performance, energy, graphics performance	Throughput, availability, scalability, energy	Price-performance, throughput, energy proportionality	Price, energy, application-specific performance

Instruction Set Architecture (ISA)

- The portion of the computer visible to the programmer or compiler writer.
- A set of all instruction that a processor can do.
- The type of internal storage in a processor is the most basic differentiation:

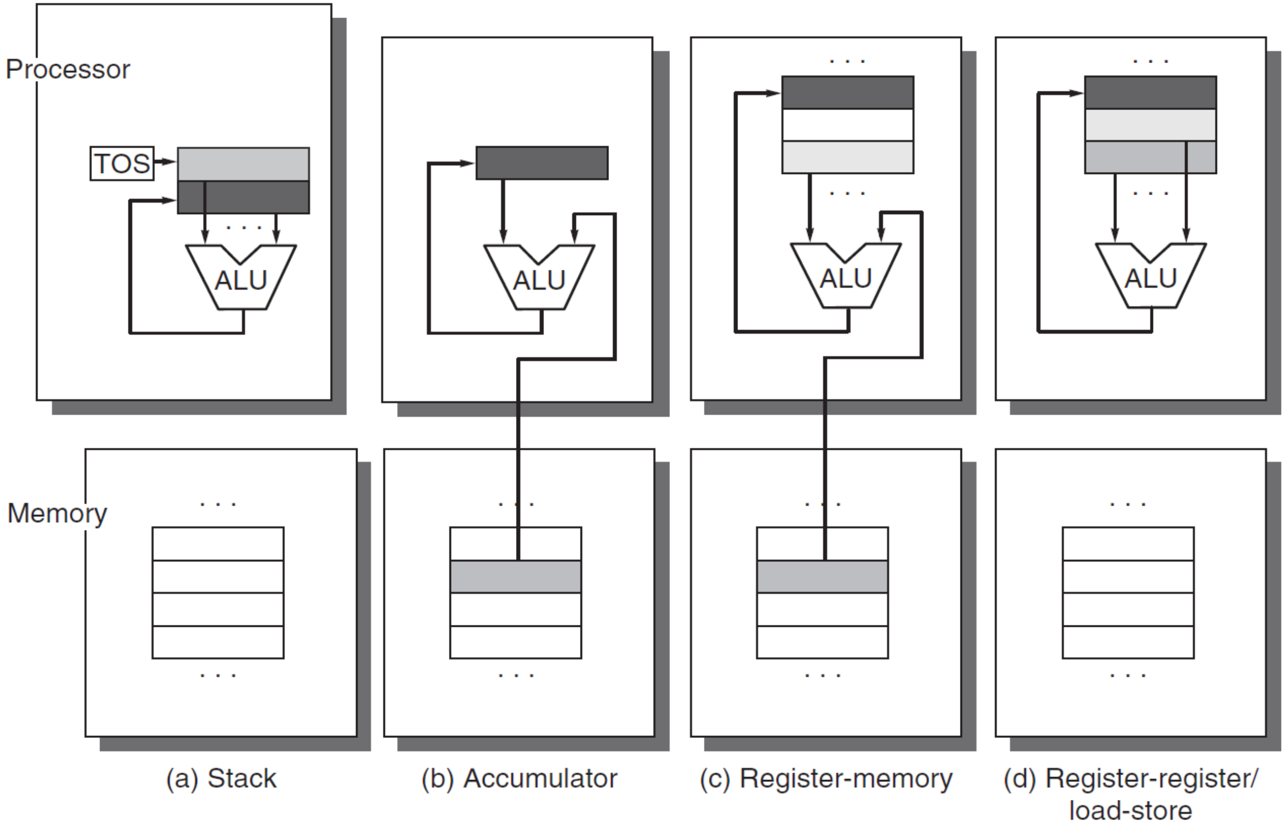
Classification:

a) Stack Architecture

b) Accumulator Architecture

c) Register-memory Architecture

d) Register-register/load-store Architecture



(a) Stack (b) Accumulator (c) Register-memory (d) Register-register/ load-store

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.

Instruction Set Architecture

- RISC (Reduced Instruction Set Computer) Architectures
  - Memory accesses are restricted to load and store instruction, and data manipulation instructions are register to register.
  - Addressing modes are limited in number.
  - Instruction formats are all of the same length.
  - Instructions perform elementary operations

- CISC (Complex Instruction Set Computer) Architectures
  - Memory access is directly available to most types of instruction.
  - Addressing mode are substantial in number.
  - Instruction formats are of different lengths.
  - Instructions perform both elementary and complex operations.

Instruction Set Architecture

- RISC (Reduced Instruction Set Computer) Architectures
  - Large register file
  - Control unit: simple and hardwired
  - pipelining

- CISC (Complex Instruction Set Computer) Architectures
  - Register file: smaller than in a RISC
  - Control unit: often micro-programmed
  - Current trend
    - CISC operation →a sequence of RISC-like operations

CISC Examples

- Examples of CISC processors are the
  - System/360(excluding the 'scientific' Model 44),
  - VAX,
  - PDP-11,
  - Motorola 68000 family
  - Intel x86 architecture based processors.

Characteristics of RISC Vs CISC processors

No	RISC	CISC
1	Simple instructions taking one cycle	Complex instructions taking multiple cycles
2	Instructions are executed by hardwired control unit	Instructions are executed by microprogramed control unit
3	Few instructions	Many instructions
4	Fixed format instructions	Variable format instructions
5	Few addressing mode, and most instructions have register to register addressing mode	Many addressing modes
6	Multiple register set	Single register set
7	Highly pipelined	Not pipelined or less pipelined