UNIT-2: THE COMPUTER HARDWARE

Contents:

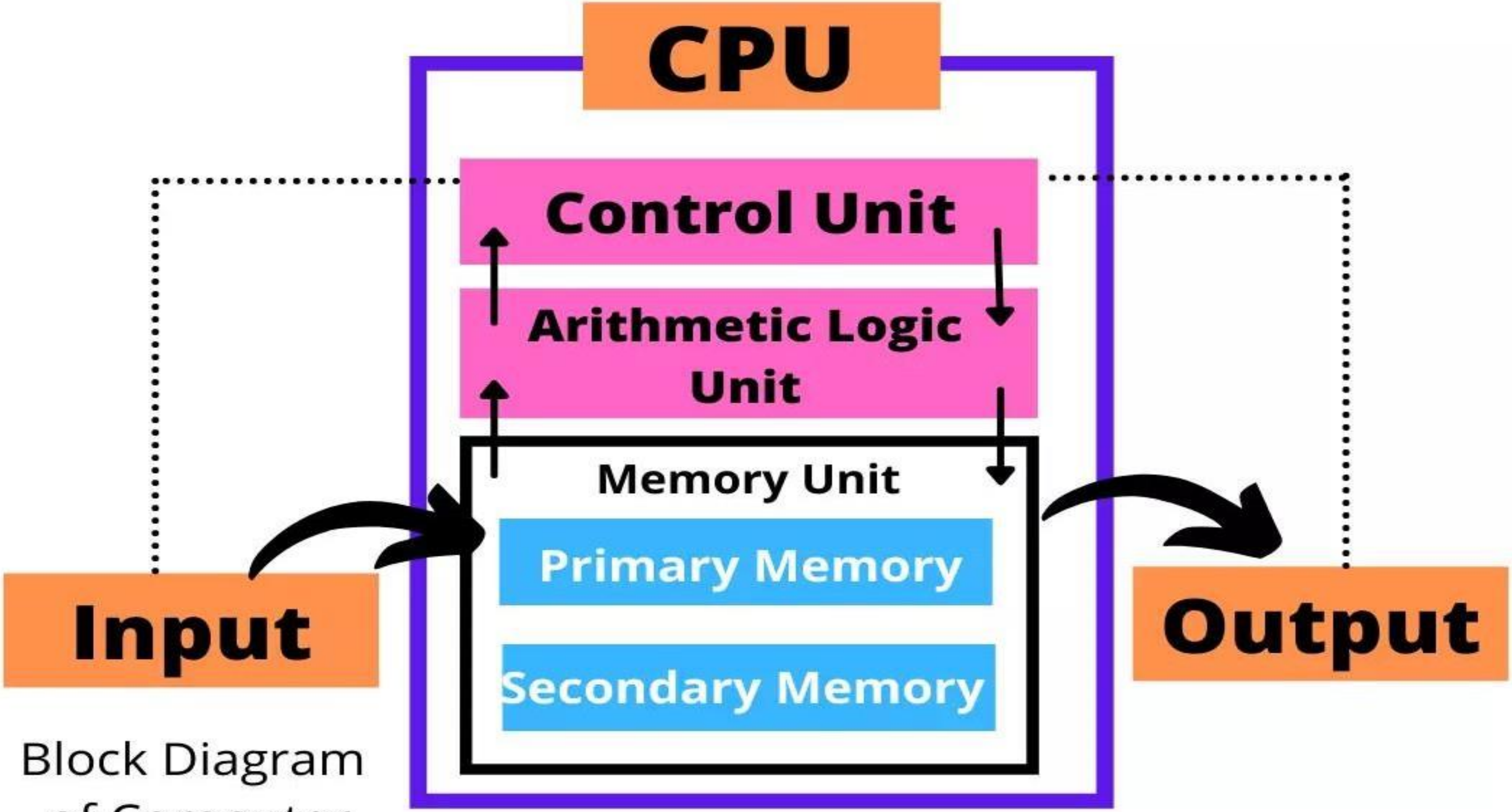
- Introduction, to computer system, Central Processing Unit, Components of CPU, Instruction Format, Instruction Set ,Instruction Cycle, Microprocessor, and Computer bus.
- Components of computer Cabinet(Power supply, motherboard, memory chips, expansion slots, ports and interfaces, processor, cables and storage devices)
- Memory unit: memory and its functions, Primary memory and Secondary memory

Computer Hardware

- Computer hardware is a collective term used to describe any of the physical components of an analog or digital computer.
- The term *hardware* distinguishes the tangible aspects of a computing device from software, which consists of written, machine-readable instructions or programs that tell physical components what to do and when to execute the instructions.
- Computer hardware includes the physical parts of a computer, such as the case, central processing unit, random access memory, monitor, mouse, keyboard, computer data storage, graphics card, sound card, speakers and motherboard.
- It encompasses everything with a circuit board that operates within a PC or laptop; including the motherboard, graphics card, CPU (Central Processing Unit), ventilation fans, webcam, power supply, and so on

WHAT IS CPU ?

- A central processing unit, also called a central processor, main processor or just processor, is the electronic circuitry that executes instructions comprising a computer program.
- The CPU performs basic arithmetic, logic, controlling, and input/output operations specified by the instructions in the program.
- It is the brain of a computer where all the calculations and process takes place

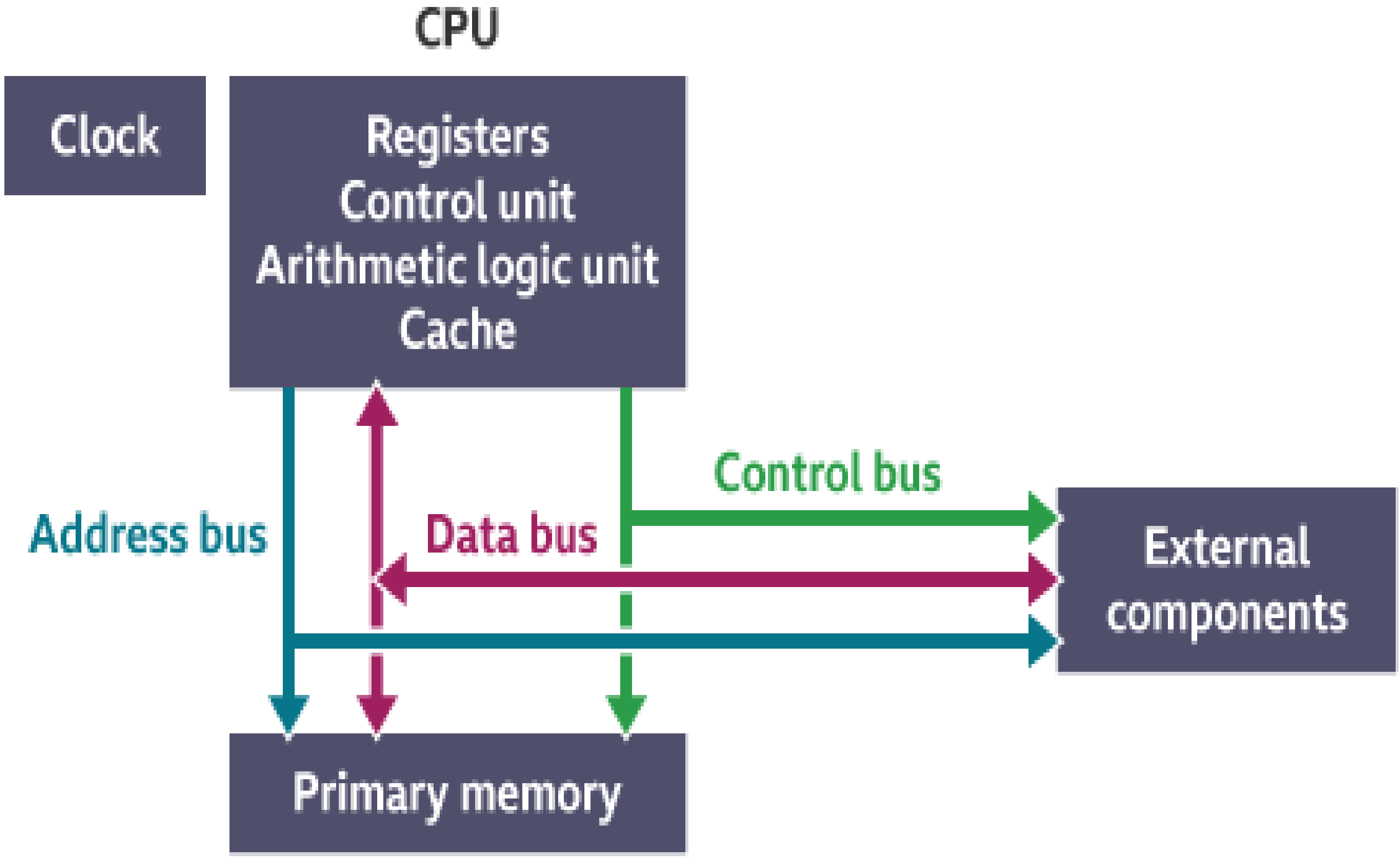


Block Diagram of Computer

COMMON CPU COMPONENTS

- The **central processing unit (CPU)** consists of six main components:
- **control unit (CU)**
- **arithmetic logic unit (ALU)**
- **registers**
- **cache**
- **buses**
- **clock**
- All components work together to allow processing and system control.

COMMON CPU COMPONENTS CONT..



CONTROL UNIT

- The CU provides several functions:
- it **fetches**, **decodes** and **executes** instructions
- it issues control signals that control **hardware**
- it moves **data** around the system
- It directs the operation of the processor
- It tells the computer's memory, arithmetic/logic unit and I/O devices on how to respond to a program instruction
- It directs the operation of the other units by providing timing and control signals

ARITHMETIC LOGIC UNIT

- The ALU has **two main functions:**
- It performs arithmetic and logical operations (decisions).
- The ALU is where calculations are done and where decisions are made.
- It acts as a gateway between primary **memory** and **secondary storage**.
- Data transferred between them passes through the ALU.
- **The ALU performs calculations and makes logical decisions**

REGISTERS

- Registers are small amounts of high-speed memory contained within the CPU. They are used by the processor to store small amounts of data that are needed during processing, such as:
- the address of the next instruction to be executed
- the current instruction being decoded
- the results of calculations
- Different processors have different numbers of registers for different purposes, but most have some, or all, of the following:
- program counter
- memory address register (MAR)
- memory data register (MDR)
- current instruction register (CIR)
- accumulator (ACC)

CACHE

- Cache is a small amount of high-speed random access memory (RAM) built directly within the processor.
- It is used to temporarily hold data and instructions that the processor is likely to reuse.
- This allows for faster processing as the processor does not have to wait for the data and instructions to be fetched from the RAM.

Parameters	Cache Memory	Register
Definition	Cache is a small, high-speed component of a computer system's memory.	Registers are fast storage elements integrated into the computer's processor.
Data Stored	Cache stores frequently accessed data of a computer.	Registers store the data that the CPU is currently processing.
Used by CPU	The CPU uses cache to retrieve previously stored data.	The CPU uses registers to process new data and information.
Location	Cache can be located on the system's motherboard or within the CPU.	Registers are part of the computer's CPU.
Data Processing	It stores data in a processed form.	It stores data in an unprocessed form.
CPU Memory Access Speed	The system's CPU can access cache memory faster than register memory.	When operating on registers, a CPU can perform multiple operations in a single clock cycle.
Examples	Web Page Cache, Prefetch Cache, etc. are examples of Cache memory.	Base Register, Index Register are examples of registers.

BUSES

- A bus is a high-speed internal connection. Buses are used to send control signals and data between the processor and other components.
- Three types of bus are used:
- **Address bus -** carries memory addresses from the processor to other components such as primary memory and input/output devices.
- **Data bus -** carries the actual data between the processor and other components.
- **Control bus -** carries control signals from the processor to other components. The control bus also carries the clock's pulses.

BUSES CONT..

- Bus is a communication system that transfer data between components inside a computer.
- Wires and electronic pathways joins the various components of a computer system
- Network of such electronic pathways and various components that create a communication among them is known as bus.
- **Internal Bus**
  - A BUS or set of wires which connects the various components inside a computer, is known as Internal Bus.
- **External Bus**
  - A Bus or set of wires which is used to connect outer peripherals or components to computer , is known as External Bus

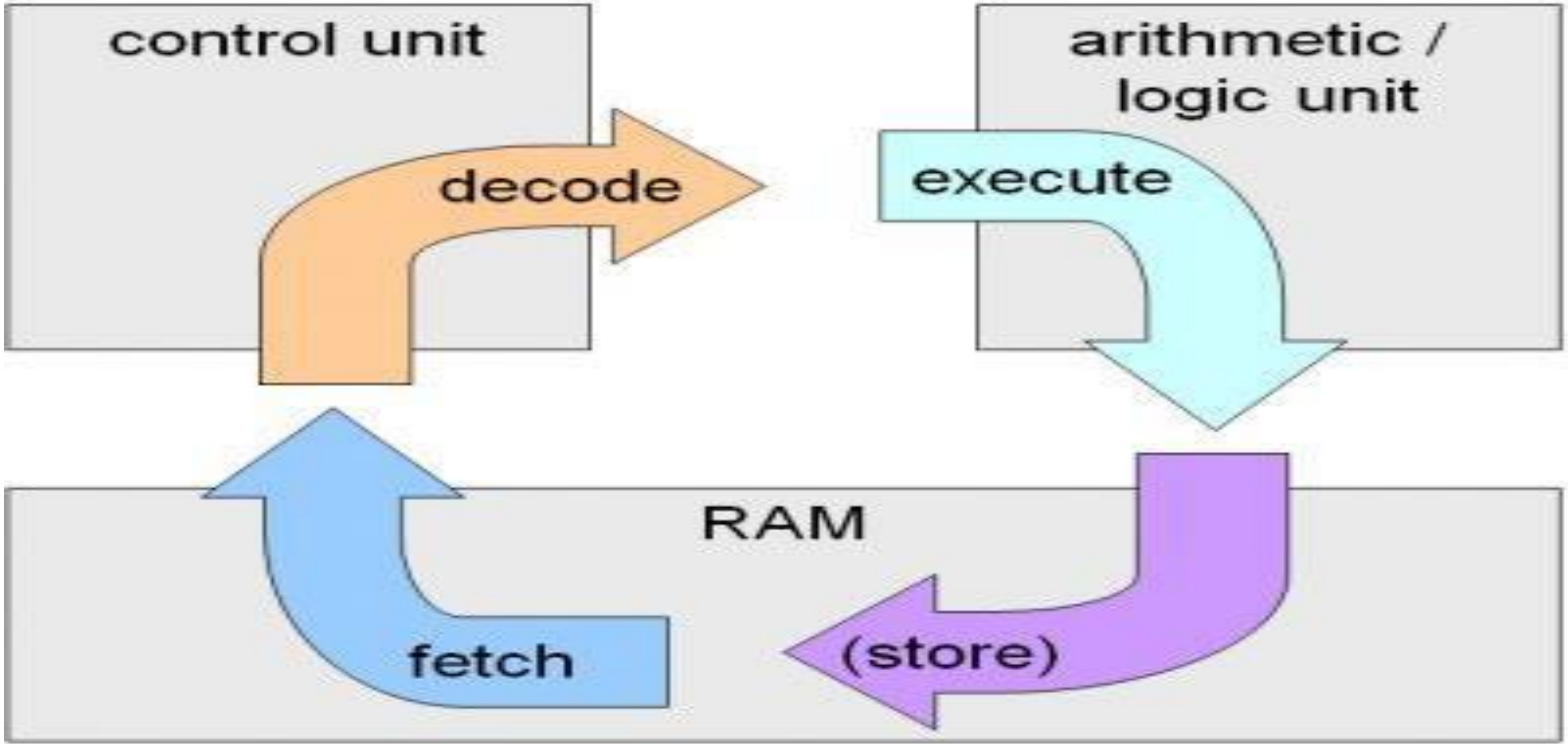
CLOCK

- The CPU contains a clock which is used to coordinate all of the computer's components.
- The clock sends out a regular electrical pulse which synchronises (keeps in time) all the components.
- The frequency of the pulses is known as the **clock speed**. Clock speed is measured in **hertz**. The higher the frequency, the more instructions can be performed in any given moment of time.

FUNCTIONS OF CPU

- **Fetch**
  - Each instruction is stored in memory and has its own address. The processor takes this address number from the program counter, which is responsible for tracking which instructions the CPU should execute next.
- **Decode**
- All programs to be executed are translated to into Assembly instructions. Assembly code must be decoded into binary instructions, which are understandable to your CPU. This step is called decoding.
- **Execute**
  - While executing instructions the CPU can do one of three things: Do calculations with its ALU, move data from one memory location to another, or jump to a different address.
- **Store**
  - The CPU must give feedback after executing an instruction and the output data is written to the memory.

CPU CYCLE



control unit
arithmetic / logic unit
decode
execute
RAM
fetch
(store)

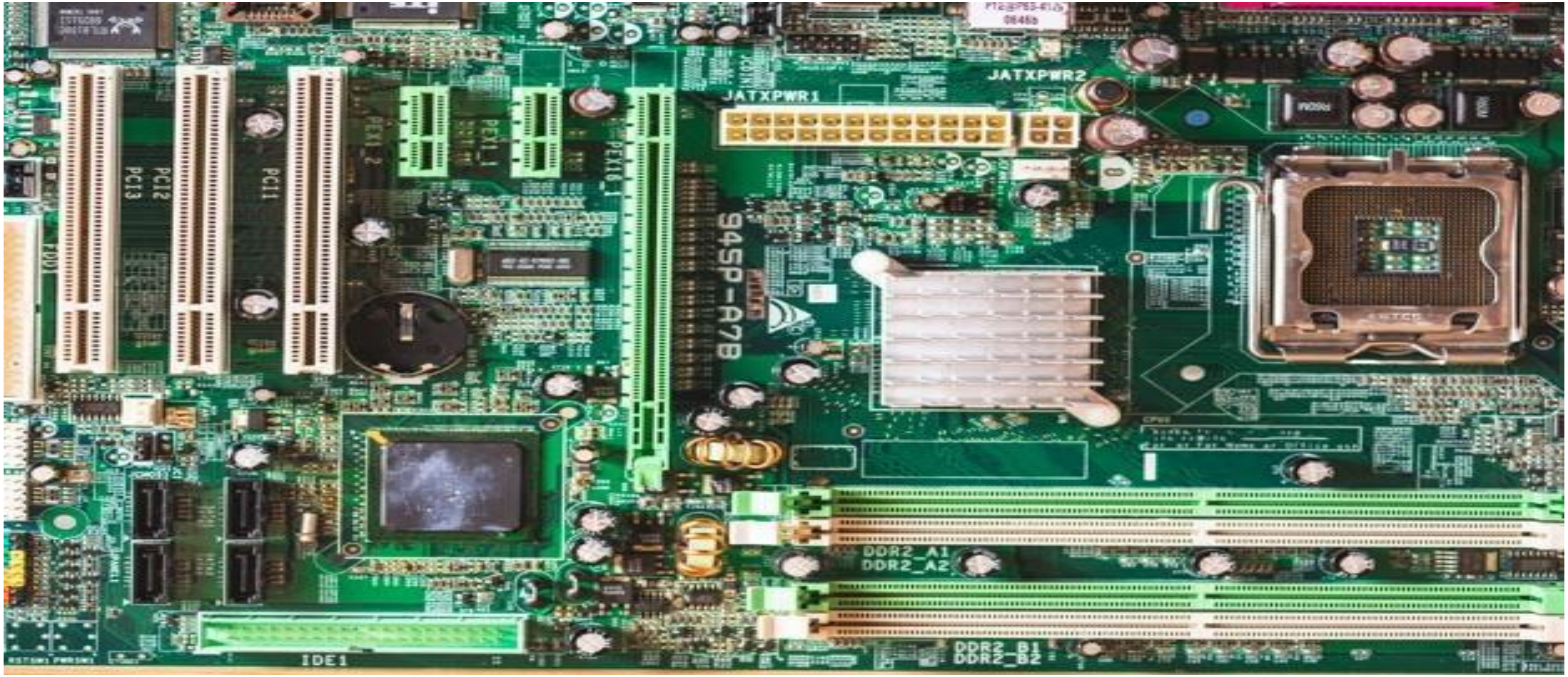
INSIDE A COMPUTER CABINET

- The computer cabinet consists of the components that are required for running the computer system effectively with fewer errors.
- There are various elements in the cabinet to which some of them are Motherboard, memory chips, cables, processors, ports, etc.

Motherboard

- The **motherboard** is the computer's **main circuit board**.
- It's a thin plate that holds the CPU, memory, connectors for the hard drive and optical drives, expansion cards to control the video and audio, and connections to your computer's ports (such as USB ports).The motherboard connects directly or indirectly to every part of the computer.

INSIDE A COMPUTER CABINET CONT..
MOTHERBOARD



INSIDE A COMPUTER CABINET CONT..

- **Ports and interfaces**
- Motherboard has a certain number of I/O sockets that are connected to the ports and interfaces found on the rear side of a computer.
- We can connect external devices to the ports and interfaces, which get connected to the computer's motherboard.
- · Serial Port— to connect old peripherals.
- · Parallel Port— to connect old printers.
- · USB Ports—to connect newer peripherals like cameras, scanners and printers to the computer. It uses a thin wire to connect to the devices, and many devices can share that wire simultaneously.
- · Fire wire is another bus, used today mostly for video cameras and external hard drives.
- · J45 connector (called LAN or Ethernet port) is used to connect the computer to a network. It corresponds to a network card integrated into the motherboard.

INSIDE A COMPUTER CABINET CONT..

Ports and interfaces Cont..

- J45 connector (called LAN or Ethernet port) is used to connect the computer to a network. It corresponds to a network card integrated into the motherboard.
- · VGA connector for connecting a monitor. This connector interfaces with the built-in graphics card.
- · Audio plugs (line-in, line-out and microphone), for connecting sound speakers and the microphone. This connector interfaces with the built-in sound card.
- · PS/2 port to connect mouse and keyboard into PC.
- · SCSI port for connecting the hard disk drives and network connectors.

INSIDE A COMPUTER CABINET CONT..

- **Expansion Slots**
- The expansion slots are located on the motherboard. The expansion cards are inserted in the expansion slots. These cards give the computer new features or increased performance.
- There are several types of slots:
- · ISA (Industry Standard Architecture) slot—To connect modem and input devices.
- · PCI (Peripheral Component Interconnect) slot—To connect audio, video and graphics. They are much faster than ISA cards.
- · AGP (Accelerated Graphic Port) slot—A fast port for a graphics card.
- · (Peripheral Component Interconnect) Express slot—Faster bus architecture than AGP and PCI buses.
- · PC Card—It is used in laptop computers. It includes Wi-Fi card, network card and external modem.

INSIDE A COMPUTER CABINET CONT..

Memory Chips

- The RAM consists of chips on a small circuit board.
- Two types of memory chips— Single In-line Memory Module (SIMM) and Dual In-line Memory Module (DIMM) are used in desktop computers.
- The CPU can retrieve information from DIMM chip at 64 bits compared to 32 bits or 16 bits transfer with SIMM chips.
- DIMM chips are used in Pentium 4 onwards to increase the access speed.

Processor

- The processor or the CPU is the main component of the computer. Select a processor based on factors like its speed, performance, reliability and motherboard support.
- Pentium Pro, Pentium 2 and Pentium 4 are some of the processors.

INSIDE A COMPUTER CABINET CONT..

Storage Devices

- The disk drives are present inside the machine.
- The common disk drives in a machine are hard disk drive, floppy drive and CD drive or DVD drive.
- High-storage devices like hard disk, floppy disk and CDs are inserted into the hard disk drive, floppy drive and CD drive, respectively.
- These storage devices can store large amounts of data, permanently.

MICROPROCESSOR

- A **Microprocessor** is an important part of a **computer architecture** without which you will not be able to perform anything on your **computer**.
- In simple words, a **Microprocessor** is a digital device on a chip which can fetch instruction from memory, decode and execute them and give results.
- A Microprocessor is an important part of a computer architecture without which you will not be able to perform anything on your computer.
- It is a programmable device that takes in input performs some arithmetic and logical operations over it and produces the desired output.
- A **microprocessor** is an electronic component that is used by a computer to do its work. It is a central processing unit on a single integrated circuit chip containing millions of very small components including transistors, resistors, and diodes that work together.

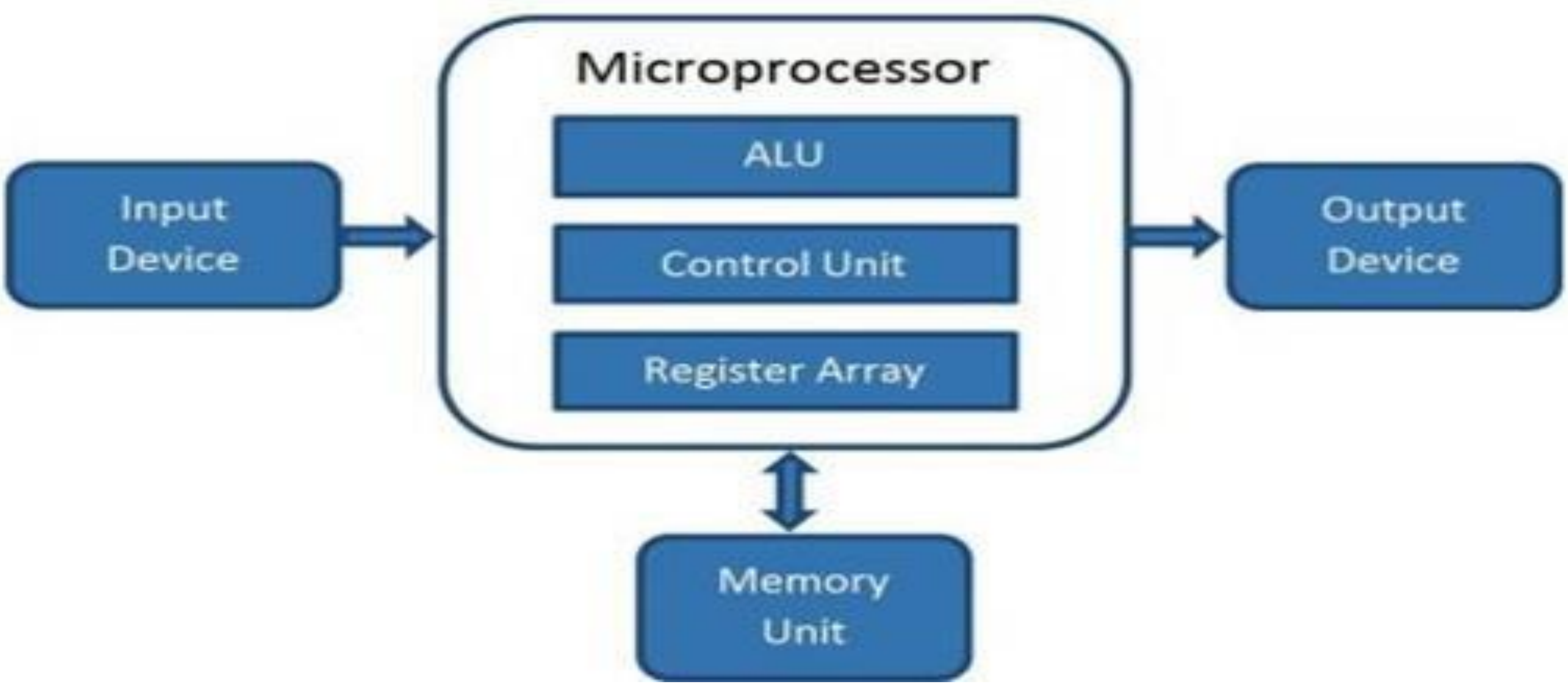
What Are the Key Components of a Microprocessor?

- **Control unit (CU)** – This part is responsible for orchestrating, coordinating and directing the execution of instructions. It picks the instructions up from the memory, understands them, and finally puts the required actions into life.
- **Arithmetic logic unit (ALU)** – This is the key element of a microprocessor responsible for calculations and logical operations orchestrated by the instructions received from the CU.
- **Registers** – Whenever your microprocessor needs to store and retrieve data quickly, it will do so in the registers. They offer limited storage space, which is perfect for information that the CPU needs right away.
- **Cache memory** – This is the access point for the microprocessor's most frequent instructions and data – it doesn't store as much information as main memory, but using cache is faster, so it accelerates the whole system.

Basic Terms used in Microprocessor

- **Instruction Set**: Instruction set is a group of instructions in the microprocessor which can be executed. It is the Interface between hardware and software.
- **Bus:** Buses are used to carry data, address and control information within a microprocessor. There are three types of Bus which are data buses, address buses, and control buses.
- **IPC (Instructions Per Cycle):** It is Measured as the number of instructions that a CPU can execute in a Single clock cycle.
- **Clock Speed:** It refers to the number of operations a processor can perform per second. It is Measured in megahertz (MHz) or gigahertz (GHz).
- **Bandwidth:** It is measured as is the number of bits processed during a single instruction.
- **Word Length** :It Shows the number of bits a processor can handle at a time.
- **Data Types:** It Supports various formats such as binary, ASCII, Signed and Unsigned Bits.

MICROPROCESSOR CONT..



Characteristics of a Microprocessor

1. **Instruction Set** : The set of complete instructions that the microprocessor executes is termed the instruction set.
2. **Word Length** : The number of bits processed in a single instruction is called word length or word size. The Greater the word size is the larger the processing power of the CPU.
3. **System Clock Speed** : A Clock speed determines how fast the single instruction can be executed in the processor.

- The microprocessor is controlled by the System Clock. A Clock speeds are generally measured in the millions of a cycles per second (MHz) and thousand million cycles per second GHz.
- A Clock speed is considered to be the very important aspect of predicting a performance of the processor.

Microprocessors and Architecture

- *Microprocessors can be classified in different categories, as follows:*

1. Based on Word Length

- Microprocessors can be based on the number of bits the processor's internal data bus or the number of bits that it can process at a time (which is known as the word length). Based on its word length, a microprocessor can be classified as 8-bit, 16-bit, 32-bit, and 64-bit.

2. Reduced Instruction Set Computer (RISC)

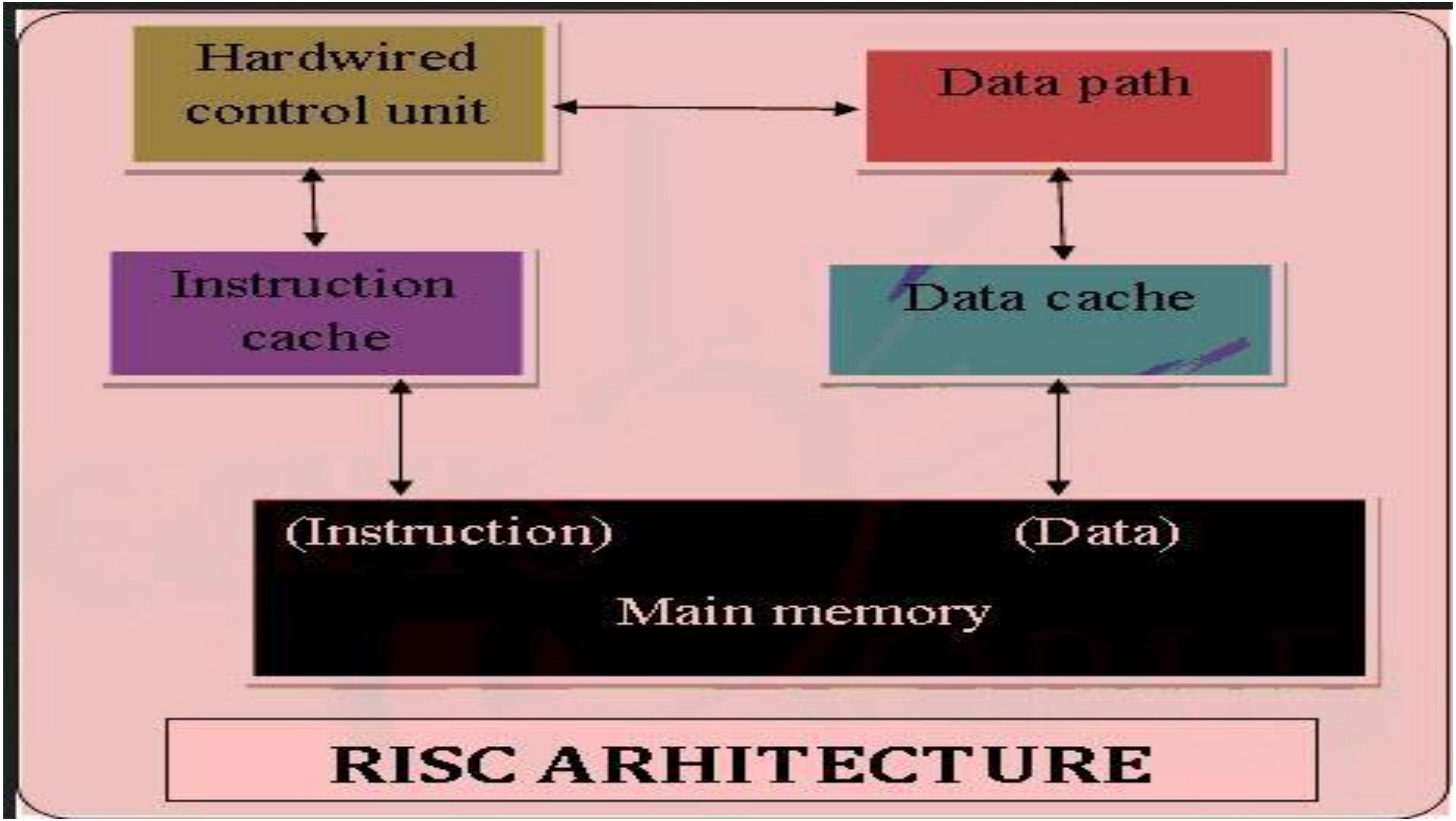
- RISC microprocessors are more general use than those that have a more specific set of instructions.
- The execution of instructions in a processor requires a special circuit to load and process data. Because RISC microprocessors have fewer instructions, they have simpler circuits, which means they operate faster.
- Additionally, RISC microprocessors have more registers, use more RAM, and use a fixed number of clock cycles to execute one instruction.

Reduced Instruction Set Architecture (RISC) Cont..

- The main idea behind this is to simplify hardware by using an instruction set composed of a few basic steps for loading, evaluating, and storing operations just like a load command will load data, a store command will store the data.

Characteristics of RISC

- Simpler instruction, hence simple instruction decoding.
- Instruction comes undersize of one word.
- Instruction takes a single clock cycle to get executed.
- More general-purpose registers.
- Simple Addressing Modes.
- Fewer Data types.
- A pipeline can be achieved.


Hardwired control unit
Data path
Instruction cache
Data cache
(Instruction)
(Data)
Main memory
RISC ARHITECTURE

Reduced Instruction Set Architecture (RISC) Cont..

Advantages of RISC

- **Simpler instructions:** RISC processors use a smaller set of simple instructions, which makes them easier to decode and execute quickly. This results in faster processing times.
- **Faster execution:** Because RISC processors have a simpler instruction set, they can execute instructions faster than CISC processors.
- **Lower power consumption:** RISC processors consume less power than CISC processors, making them ideal for portable devices.

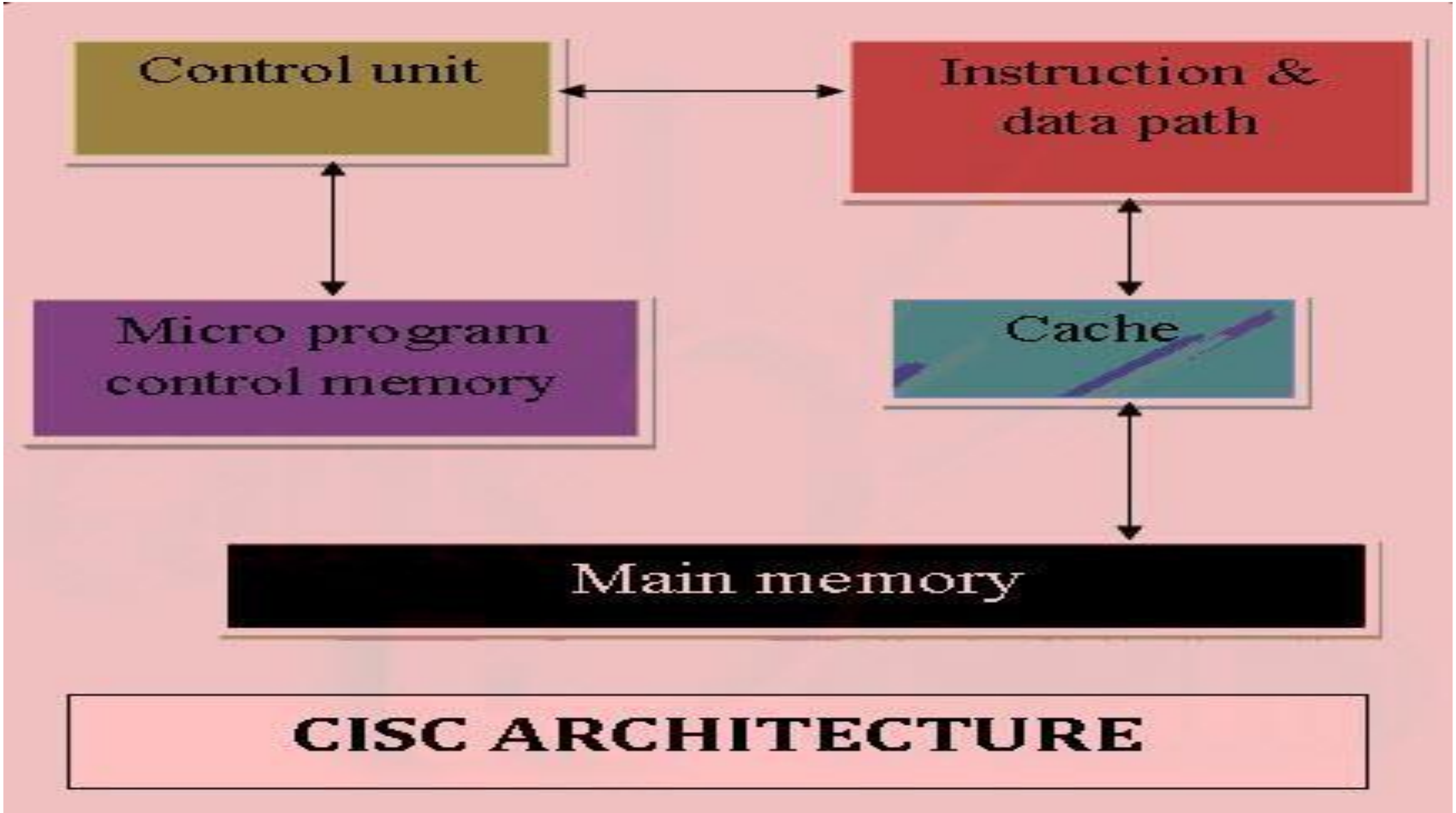
Disadvantages of RISC

- **More instructions required:** RISC processors require more instructions to perform complex tasks than CISC processors.
- **Increased memory usage:** RISC processors require more memory to store the additional instructions needed to perform complex tasks.
- **Higher cost:** Developing and manufacturing RISC processors can be more expensive than CISC processors.

3. Complex Instruction Set Computer

- CISC microprocessors are the opposite of RISC microprocessors. Their purpose is to reduce the number of instructions for each program.
- The number of cycles per instruction is ignored. Because complex instructions are made directly into the hardware, CISC microprocessors are more complex and slower.
- CISC microprocessors use little RAM, have more transistors, have fewer registers, have numerous clock cycles for each instruction, and have a variety of addressing modes.

.


Control unit
Instruction &
data path
Micro program
control memory
Cache
Main memory
CISC ARCHITECTURE

Complex Instruction Set Architecture (CISC) Cont..

- The main idea is that a single instruction will do all loading, evaluating, and storing operations just like a multiplication command will do stuff like loading data, evaluating, and storing it, hence it's complex.

Characteristics of CISC

- Complex instruction, hence complex instruction decoding.
- Instructions are larger than one-word size.
- Instruction may take more than a single clock cycle to get executed.
- Less number of general-purpose registers as operations get performed in memory itself.
- Complex Addressing Modes.
- More Data types.

Complex Instruction Set Architecture (CISC) Cont..

Advantages of CISC

- **Reduced code size:** CISC processors use complex instructions that can perform multiple operations, reducing the amount of code needed to perform a task.
- **More memory efficient:** Because CISC instructions are more complex, they require fewer instructions to perform complex tasks, which can result in more memory-efficient code.
- **Widely used:** CISC processors have been in use for a longer time than RISC processors, so they have a larger user base and more available software.

Disadvantages of CISC

- **Slower execution:** CISC processors take longer to execute instructions because they have more complex instructions and need more time to decode them.
- **More complex design:** CISC processors have more complex instruction sets, which makes them more difficult to design and manufacture.
- **Higher power consumption:** CISC processors consume more power than RISC processors because of their more complex instruction sets.

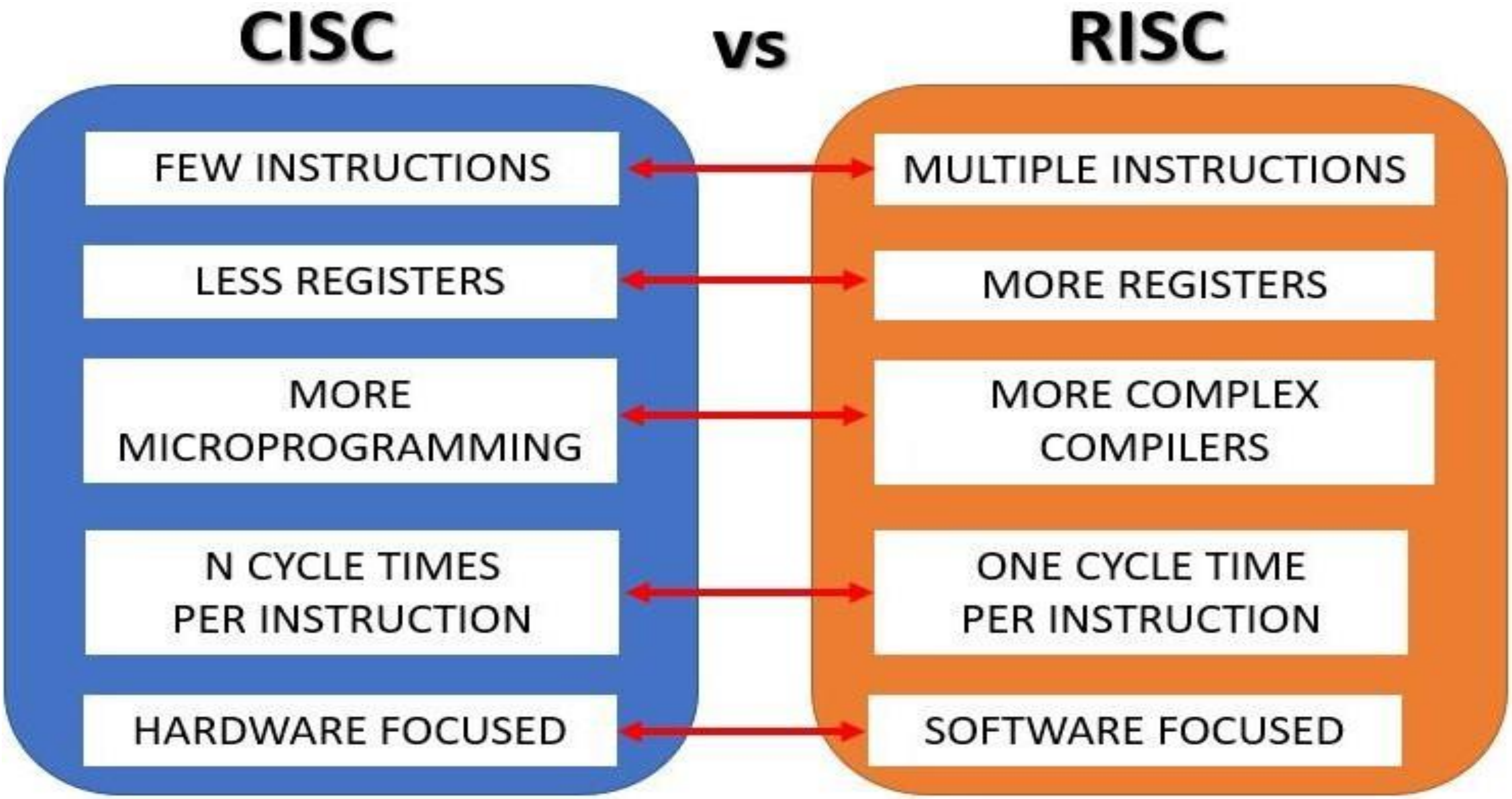
RISC Vs CISC ARCHITECTURE

- **RISC** stands for 'Reduced Instruction Set Computer whereas, **CISC** stands for Complex Instruction Set Computer. The **RISC** processors have a smaller set of instructions with few addressing nodes.
- It is a type of microprocessor that has a limited number of instructions. They can execute their instructions very fast because instructions are very small and simple.
- **RISC** chips require fewer transistors which make them cheaper to design and produce.
- The **CISC** processors have a larger set of instructions with many addressing nodes.
- A complex instruction set **computer** (**CISC**) is a **computer** in which single instructions can execute several low-level operations (such as a load from memory, an arithmetic operation, and a memory store) or are capable of multi-step operations or addressing modes within single instructions.

RISC AND CISC ARCHITECTURE CONT…

CISC – Older, slower, more complex – perfect for systems working on videos, graphics, vast amounts of complicated data or science computers.

RISC – Based on simple, mostly one-word instructions, the processors in this architecture are usually smaller, more energy-efficient, and quicker, which makes it perfect for mobile devices and embedded systems.


CISC
vs
RISC
FEW INSTRUCTIONS
MULTIPLE INSTRUCTIONS
LESS REGISTERS
MORE REGISTERS
MORE MICROPROGRAMMING
MORE COMPLEX COMPILERS
N CYCLE TIMES PER INSTRUCTION
ONE CYCLE TIME PER INSTRUCTION
HARDWARE FOCUSED
SOFTWARE FOCUSED

RISC AND CISC APPLICATION

- RISC architecture can be used with high-end applications like telecommunication, image processing, video processing, etc.
- RISC has fixed format instruction.
- CISC architecture can be used with low-end applications like home automation, security system, etc.
- CISC has variable format instruction.

Applications of Microprocessor

Computer: The Microprocessors are the CPU’s in computers.
Embedded Systems: It is utilized as the main processing block in the Embedded systems such as Washing machines, microwaves and other equipment's.
Industrial Automation: It can be used to control Industrial machinery and equipment's.
Automotive: The Modern day Vehicles uses Microprocessors in there ECUs.
Telecommunications: The Microprocessors are utilized in the Telecommunications systems such as routers, modems, and switches.

INSTRUCTION SET

- An **instruction set** is a group of commands for a CPU in machine language. ... All CPUs have **instruction sets** that enable commands to the processor directing the CPU to switch the relevant transistors.
- Some **instructions** are simple read, write and move commands that direct data to different hardware.
- The complete collection of instructions that are understood by a CPU
  - The physical hardware that is controlled by the instructions is referred to as the Instruction Set Architecture (ISA)
- The instruction set is ultimately represented in binary **machine code** also referred to as **object code**
  - Usually represented by assembly codes to human programmer

INSTRUCTION SET CONT..

Instruction sets are differentiated by the following:

- Number of bits per instruction.
- Stack-based or register-based.
- Number of explicit operands per instruction.
- Operand location.
- Types of operations.
- Type and size of operands.

INSTRUCTION SET CONT..

Instruction set architectures are measured according to:

- Main memory space occupied by a program.
- Instruction complexity.
- Instruction length (in bits).
- Total number of instruction in the instruction set.

INSTRUCTION FORMATS

- The instruction formats are a sequence of bits (0 and 1).
- These bits, when grouped, are known as fields. Each field of the machine provides specific information to the CPU related to the operation and location of the data.
- The instruction format also defines the layout of the bits for an instruction.
- In computer organization, instruction formats refer to the way instructions are encoded and represented in machine language.
- There are several types of instruction formats, including zero, one, two, and three-address instructions.
- Each type of instruction format has its own advantages and disadvantages in terms of code size, execution time, and flexibility.
- Modern computer architectures typically use a combination of these formats to provide a balance between simplicity and power.

ZERO ADDRESS INSTRUCTIONS

- These instructions do not specify any operands or addresses.
- Instead, they operate on data stored in registers or memory locations implicitly defined by the instruction.
- For example, a zero-address instruction might simply add the contents of two registers together without specifying the register names.

Advantages of Zero-Address instructions

- **Stack-based Operations:** In stack-based architectures, where operations implicitly employ the top items of the stack, zero-address instructions are commonly used.
- **Reduced Instruction Set:** It reduces the complexity of the CPU design by streamlining the instruction set, which may boost reliability.
- **Less Decoding Complexity:** Especially helpful for recursive or nested processes, which are frequently used in function calls and mathematical computations.
- **Efficient in Nested Operations:** Less bits are required to specify operands, which simplifies the logic involved in decoding instructions.
- **Compiler Optimization:** Because stacks are based on stacks, several algorithms can take use of this to improve the order of operations.

Disadvantages of Zero-Address instructions

- **Stack Dependency:** In contrast to register-based architectures, zero-address instructions might result in inefficiencies when it comes to operand access because of their heavy reliance on the stack.
- **Overhead of Stack Operations:** Performance might be negatively impacted by the frequent push and pop actions needed to maintain the stack.
- **Limited Addressing Capability:** The processing of intricate data structures may become more difficult since they do not directly support accessing memory regions or registers.
- **Difficult to Optimize:** Because operand access is implied in stack-based designs, code optimization might be more difficult. **Harder to Debug:** When compared to register-based operations, stack-based operations might be less obvious and more difficult to debug.

ONE ADDRESS INSTRUCTIONS

- These instructions specify one operand or address, which typically refers to a memory location or register.
- The instruction operates on the contents of that operand, and the result may be stored in the same or a different location. For example, a one-address instruction might load the contents of a memory location into a register.
- This uses an implied ACCUMULATOR register for data manipulation.
- One operand is in the accumulator and the other is in the register or memory location. Implied means that the CPU already knows that one operand is in the accumulator so there is no need to specify it

opcode	operand/address of operand	mode

Advantages One-address instructions

- **Intermediate Complexity:** Strikes a balance between versatility and simplicity, making it more adaptable than zero-address instructions yet simpler to implement than multi-address instructions.
- **Reduced Operand Handling:** Compared to multi-address instructions, operand fetching is made simpler by just needing to handle a single explicit operand.
- **Implicit Accumulator: O** ften makes use of an implicit accumulator register, which can expedite up some operations' execution and simplify designs in other situations.
- **Code Density: S** maller code in comparison to two- and three-address instructions, which may result in more efficient use of memory and the instruction cache.
- **Efficient Use of Addressing Modes:** Can make use of different addressing modes (such indexed, direct, and indirect) to improve flexibility without adding a lot of complexity.

Disadvantages One-address instructions

- **Accumulator Bottleneck:** Often uses an accumulator, which can act as a bottleneck and reduce efficiency and parallelism.
- **Increased Instruction Count:** Multiple instructions may be needed for complex processes, which would increase the overall number of instructions and code size.
- **Less Efficient Operand Access:** There is just one operand that is specifically addressed, which might result in inefficient access patterns and extra data management instructions.
- **Complex Addressing Modes:** The instruction set and decoding procedure get more complicated when several addressing modes are supported.
- **Data Movement Overhead:** Moving data between memory and the accumulator could need more instructions, which would increase overhead.

TWO ADDRESS INSTRUCTIONS

- These instructions specify two operands or addresses, which may be memory locations or registers. The instruction operates on the contents of both operands, and the result may be stored in the same or a different location.
- For example, a two-address instruction might add the contents of two registers together and store the result in one of the registers.
- This is common in commercial computers. Here two addresses can be specified in the instruction. Unlike earlier in one address instruction, the result was stored in the accumulator, here the result can be stored at different locations rather than just accumulators, but require more number of bit to represent the address.

opcode	Destination address	Source address	mode

Advantages of Two-address instructions

- **Improved Efficiency:** Allows for the execution of operations directly on memory or registers, which reduces the amount of instructions required for certain activities.
- **Flexible Operand Use:** Increases programming variety by offering more options for operand selection and addressing modes.
- **Intermediate Data Storage:** May directly store interim results, increasing some algorithms' and calculations' efficiency.
- **Enhanced Code Readability:** Produces code that is frequently easier to read and comprehend than one-address instructions, which is beneficial for maintenance and troubleshooting.
- **Better Performance:** Better overall performance can result from these instructions because they minimize the amount of memory accesses required for certain processes.

Disadvantages of Two-address instructions

- **Operand Overwriting:** Usually, the result overwrites one of the source operands, which might lead to an increase in the number of instructions needed to maintain data.
- **Larger Instruction Size:** Because two-address instructions are bigger than zero- and one-address instructions, the memory footprint may be increased.
- **Intermediate Results Handling:** It is frequently necessary to handle intermediate outcomes carefully, which can make programming more difficult and result in inefficiencies.
- **Decoding Complexity:** The design and performance of the CPU may be impacted by the greater complexity involved in decoding two addresses.
- **Inefficient for Some Operations:** The two-address style could still be inefficient for some tasks, needing more instructions to get the desired outcome.

THREE ADDRESS INSTRUCTIONS

- These instructions specify three operands or addresses, which may be memory locations or registers. The instruction operates on the contents of all three operands, and the result may be stored in the same or a different location.
- For example, a three-address instruction might multiply the contents of two registers together and add the contents of a third register, storing the result in a fourth register.
- This has three address fields to specify a register or a memory location. Programs created are much short in size but number of bits per instruction increases.
- These instructions make the creation of the program much easier but it does not mean that program will run much faster because now instructions only contain more information but each micro-operation (changing the content of the register, loading address in the address bus etc.) will be performed in one cycle only.

opcode	Destination address	Source address	Source address	mode

Advantages of Three-address instructions

- **Direct Representation of Expressions:** Reduces the need for temporary variables and extra instructions by enabling the direct representation of complicated expressions.
- **Parallelism:** Allows for the simultaneous fetching and processing of several operands, which facilitates parallelism in CPU architecture.
- **Compiler Optimization:** Makes it possible for more complex compiler optimizations to be implemented, which improve execution efficiency by scheduling and reordering instructions.
- **Reduced Instruction Count:** May increase execution performance even with bigger instruction sizes by perhaps lowering the overall number of instructions required for complicated processes.
- **Improved Pipeline Utilization:** More information in each instruction allows CPU pipelines to be used more efficiently, increasing throughput overall.
- **Better Register Allocation:** Permits direct manipulation of several registers inside a single instruction, enabling more effective usage of registers.

Disadvantages of Three-address instructions

- **Largest Instruction Size:** Has the highest memory requirements per instruction, which can put strain on the instruction cache and increase code size.
- **Complex Instruction Decoding:** Three addresses to decode adds complexity to the CPU architecture, which might affect power consumption and performance.
- **Increased Operand Fetch Time:** Each instruction may execute more slowly if obtaining three operands takes a long period.
- **Higher Hardware Requirements:** Has the potential to raise cost and power consumption since it requires more advanced hardware to handle the higher operand handling and addressing capabilities.
- **Power Consumption:** Higher power consumption is a crucial factor for devices that run on batteries since it can be caused by more complicated instructions and increased memory utilization.
- **Overall, the choice of instruction format depends on the specific requirements of the computer architecture and the trade-offs between code size, execution time, and flexibility.**

ADVANTAGES OF ZERO-ADDRESS, ONE-ADDRESS, TWO-ADDRESS AND THREE-ADDRESS INSTRUCTIONS

- **Zero-address instructions**
- They are simple and can be executed quickly since they do not require any operand fetching or addressing. They also take up less memory space.
- **One-address instructions**
- They allow for a wide range of addressing modes, making them more flexible than zero-address instructions. They also require less memory space than two or three-address instructions.
- **Two-address instructions**
- They allow for more complex operations and can be more efficient than one-address instructions since they allow for two operands to be processed in a single instruction. They also allow for a wide range of addressing modes.
- **Three-address instructions**
- They allow for even more complex operations and can be more efficient than two-address instructions since they allow for three operands to be processed in a single instruction. They also allow for a wide range of addressing modes.

DISADVANTAGES OF ZERO-ADDRESS, ONE-ADDRESS, TWO-ADDRESS AND THREE-ADDRESS INSTRUCTIONS

- **Zero-address instructions**
- They can be limited in their functionality and do not allow for much flexibility in terms of addressing modes or operand types.
- **One-address instructions**
- They can be slower to execute since they require operand fetching and addressing.
- **Two-address instructions**
- They require more memory space than one-address instructions and can be slower to execute since they require operand fetching and addressing.
- **Three-address instructions**
- They require even more memory space than two-address instructions and can be slower to execute since they require operand fetching and addressing.

Example Of format Of Instruction

1. Zero Address Instruction examples

Assembly language instruction – PUSH A, PUSH B etc.

Stack transfer operation – TOS <- A, TOS <- B etc.

2. One Address Instruction examples

Assembly language instruction – LOAD C, ADD B, STORE T etc.

Operation Register instruction – AC <- M[T], AC <- M[C] etc.

3. Two Address Instruction examples

Assembly language instruction – MOV R1, A; ADD R1, B etc.

Operation Register instruction – R1 <- M[A], R2 <- M[C] etc.

4.Three Address Instruction examples

Assembly language instruction – ADD R1, A, B etc.

Operation Register instruction – R1 <- M[A] + M[B] etc.

Parts Of Instruction Format

1. Addressing Mode

The data is represented in the instruction format with the help of addressing mode

The addressing mode is the first part of the instruction format

The data can either be stored in the memory of a computer or it can be located in the register of the CPU

2. Operation Code(OPCODE)

The operation code gives instructions to the processor to perform the specific Operation

The operation code is the second part of the instruction format

3. OPERAND

It is the part of the instruction format that specifies the data or the address of the data

Depending upon the processor of the computer the instruction format contains zero to three operands

INSTRUCTION FORMATS CONT..

In designing an instruction set, consideration is given to:

- Instruction length.
  - Whether short, long, or variable.
- Number of operands.
- Number of addressable registers.
- Memory organization.
  - Whether byte- or word addressable.
- Addressing modes.
  - Choose any or all: direct, indirect or indexed.

ELEMENTS OF AN INSTRUCTION

- Operation code (Op code)
  - Do this
- Source Operand reference(s)
  - To this
- Result Operand reference(s)
  - Put the answer here
- Next Instruction Reference
  - When you are done, do this instruction next

WHERE ARE THE OPERANDS?

- Main memory
- CPU register
- I/O device
- In instruction itself
- To specify which register, which memory location, or which I/O device, we’ll need some addressing scheme for each

HOW MANY OPERANDS IS BEST?

- More operands
  - More complex (powerful?) instructions
  - Fewer instructions per program
- More registers
  - Inter-register operations are quicker
- Fewer operands
  - Less complex (powerful?) instructions
  - More instructions per program
  - Faster fetch/execution of instructions

INSTRUCTION TYPES

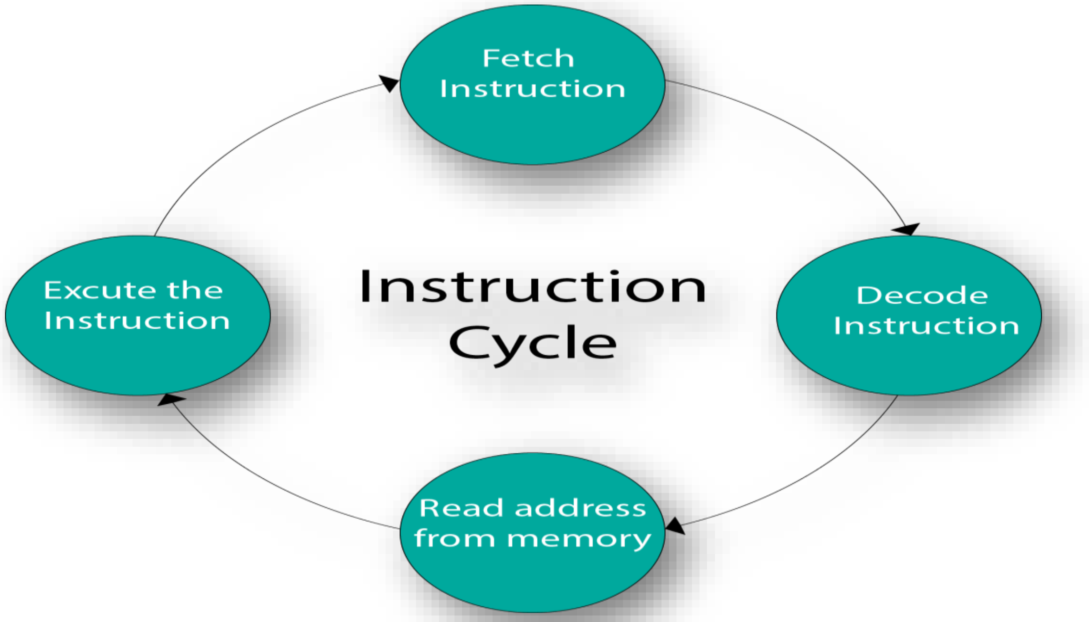
Instructions fall into several broad categories that you should be familiar with:

- Data movement
- Arithmetic
- Boolean
- Bit manipulation
- I/O
- Control transfer
- Special purpose

Can you think of some examples of each of these?

INSTRUCTION CYCLE

- A program residing in the memory unit of a computer consists of a sequence of instructions. These instructions are executed by the processor by going through a cycle for each instruction.
- In a basic computer, each instruction cycle consists of the following phases:
- Fetch instruction from memory.
- Decode the instruction.
- Read the effective address from memory.
- Execute the instruction.



INSTRUCTION-LEVEL PIPELINING

- Some CPUs divide the fetch-decode-execute cycle into smaller steps.
- These smaller steps can often be executed in parallel to increase throughput.
- Such parallel execution is called *instruction-level pipelining*.
- This term is sometimes abbreviated *ILP* in the literature.

INSTRUCTION PRE FETCH

- Simple version of Pipelining – treating the instruction cycle like an assembly line
- Fetch accessing main memory
- Execution usually does not access main memory
- Can fetch next instruction during execution of current instruction
- Called instruction prefetch

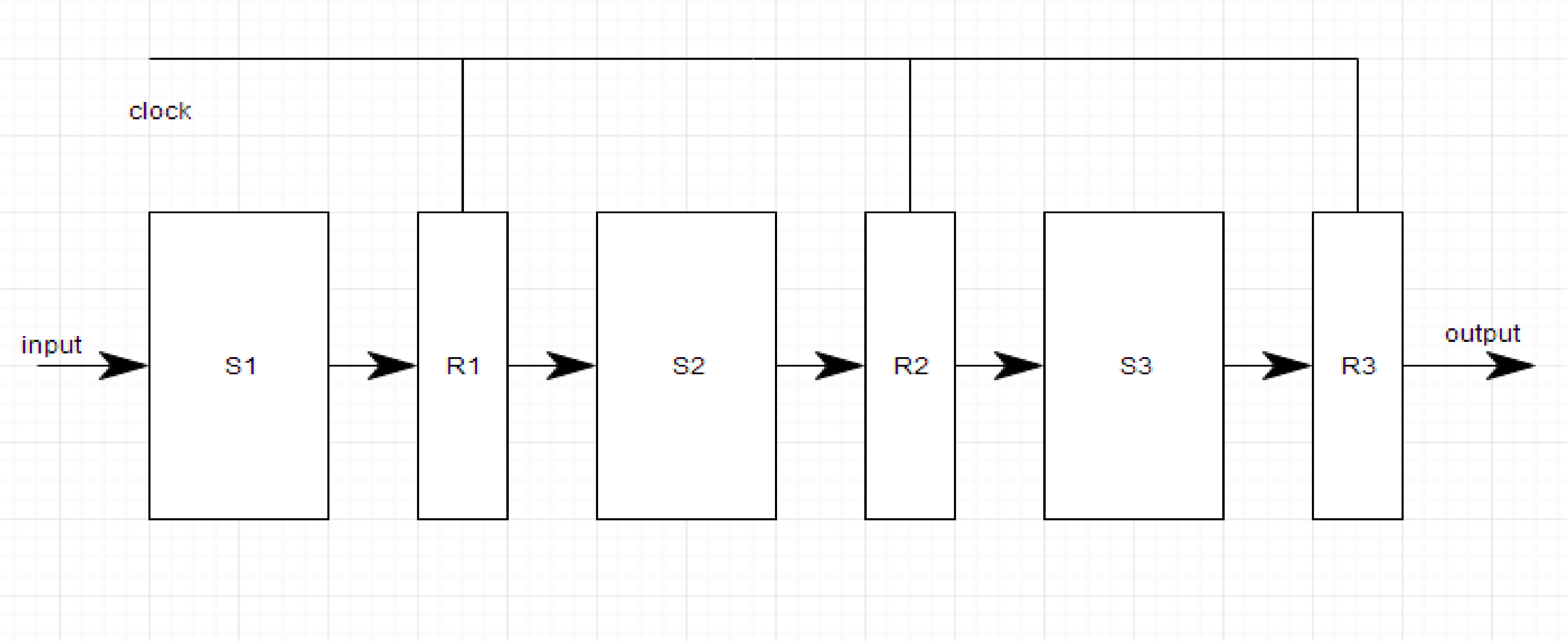
PIPELINING

- Pipelining is the process of accumulating instruction from the processor through a pipeline.
- It allows storing and executing instructions in an orderly process. It is also known as **pipeline processing**.
- Pipelining is a technique where multiple instructions are overlapped during execution.
- Pipeline is divided into stages and these stages are connected with one another to form a pipe like structure. Instructions enter from one end and exit from another end.
- Pipelining increases the overall instruction throughput.

PIPELINING CONT..

- In pipeline system, each segment consists of an input register followed by a combinational circuit. The register is used to hold data and combinational circuit performs operations on it.
- The output of combinational circuit is applied to the input register of the next segment.
- Pipeline system is like the modern day assembly line setup in factories.
- For example in a car manufacturing industry, huge assembly lines are setup and at each point, there are robotic arms to perform a certain task, and then the car moves on ahead to the next arm.
- **Pipelining** is an implementation technique where multiple instructions are overlapped in execution.
- The computer **pipeline** is divided in stages. Each stage completes a part of an instruction in **parallel**.
- **Pipelining** does not decrease the time for individual instruction execution. .

PIPELINING CONT..



Advantages of Pipelining

- **Increased Throughput:** Pipelining enhance the throughput capacity of a CPU and enables a number of instruction to be processed at the same time at different stages. This leads to the improvement of the amount of instructions accomplished in a given period of time, thus improving the efficiency of the processor.
- **Improved CPU Utilization:** From superimposing of instructions, pipelining helps to ensure that different sections of the CPU are useful. This gives no time for idling of the various segments of the pipeline and optimally utilizes hardware resources.
- **Higher Instruction Throughput:** Pipelining occurring because when one particular instruction is in the execution stage it is possible for other instructions to be at varying stages of fetch, decode, execute, memory access, and write-back. In this manner there is concurrent processing going on and the CPU is able to process more number of instructions in a given time frame than in non pipelined processors.
- **Better Performance for Repeated Tasks:** Pipelining is particularly effective when all the tasks are accompanied by repetitive instructions, because the use of the pipeline shortens the amount of time each task takes to complete.
- **Scalability:** Pipelining is RSVP implemented in different types of processors hence it is scalable from simple CPU's to an advanced multi-core processor.

Disadvantages of Pipelining

- **Pipeline Hazards:** Pipelining may result to data hazards whereby instructions depends on other instructions; control hazards, which arise due to branch instructions; and structural hazards whereby there are inadequate hardware facilities. Some of these hazards may lead to delays hence tough strategies to manage them to ensure progress is made.
- **Increased Complexity:** Pipelining enhances the complexity of processor design as well as its application as compared to non-pipelined structures. Pipelining stages management, dealing with the risks and correct instruction sequence contribute to the design and control considerations.
- **Stall Cycles:** When risks are present, pipeline stalls or bubbles can be brought about, and this produces idle times in certain stages in the pipeline. These stalls can actually remove some of the cycles acquired by pipelining, thus reducing the latter's efficiency.
- **Instruction Latency:** While pipelining increases the throughput of instructions the delay of each instruction may not necessarily be reduced. Every instruction must still go through all the pipeline stages and the time it takes for a single instruction to execute can neither reduce nor decrease significantly due to overheads.
- **Hardware Overhead:** It increases the complexity in designing the pipelining due to the presence of pipeline registers and the control logic used in managing the pipe stages and the data. This not only increases the cost of the wares but also forces integration of more complicated, and thus costly, hardware.

Memory unit: memory and its functions,

A memory unit is a small storage device that holds a memory for a computer and can be accessed through the software.
Memory units are typically used to store the information of the programs.
The computer has two types of storage systems: volatile and non-volatile.

TYPES OF COMPUTER MEMORY

- **Computer memory** plays vital role in the computer industry because without **computer memory** entire system like as plastic box.
- There are two types
- **Primary Memory (Storage Device)**
- **Secondary Memory (Storage Device)**

What is Primary Memory?

- It is also referred to as main memory or internal memory. It is a computer system’s temporary storage component which is directly accessible by the central processing unit (CPU). It houses data for immediate processing.

Characteristics

- volatile: Data is lost upon power loss.
- High-speed access.
- Limited capacity relative to secondary storage.
- Examples: Random Access Memory(RAM), Read-Only Memory, Cache memory.

Primary Memory Cont..

Advantages:

- **High-speed access:** Data can be retrieved and stored very quickly.
- **Directly accessible by CPU:** No intermediate steps are required for data transfer.

Disadvantages:

- **Volatile:** Data is lost when power is turned off.
- **Limited storage capacity**: compared to secondary storage, primary memory is relatively small.
- **Expensive**: Cost per unit is higher than secondary storage.

Applications:

- Temporary storage
- Multitasking
- Buffering
- Caching

What is Secondary Memory?

- Secondary memory or external memory serves as long-term storge for data and programs. Unlike primary memory, it is not directly accessible by the CPU and requires input/output operations.

Characteristics

- Non-volatile: Data persists even when the system is powered off.
- slower access speeds compared to primary memory.
- High storage capacity.
- Examples: Hard Disk Drives(HDD), Solid-State Drives(SSD), Optical drives(CD, DVD, Blu-ray).

Secondary Memory Cont..

Advantages:

- **Non-volatile:** Data persists even when the power is turned off.
- **Large storage capacity:** can store vast amount of data.
- **Relatively Inexpensive:** cost-effective for storing large volumes of data.

Disadvantages:

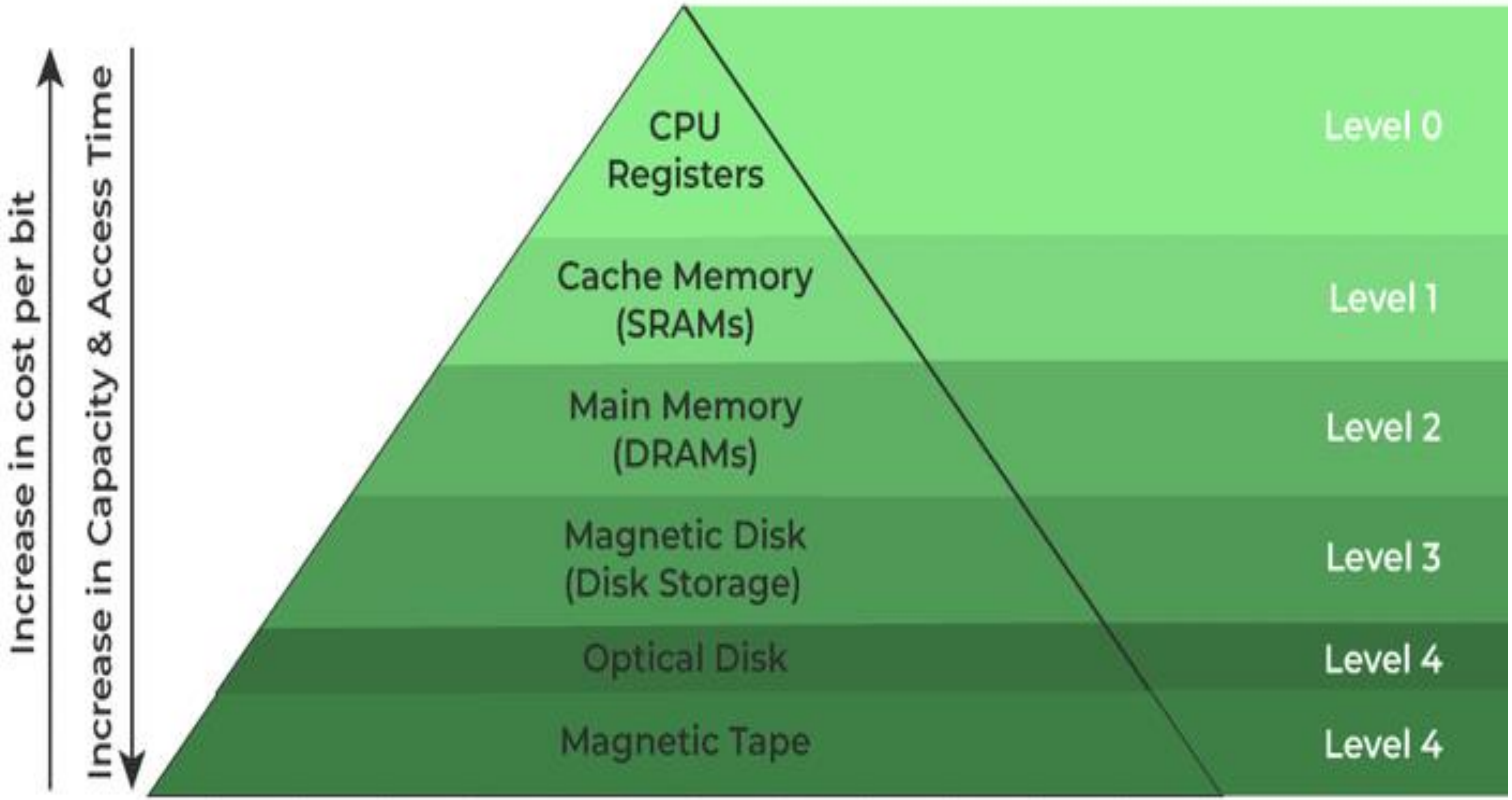
- **Slower access time:** Data retrieval is slower compared to primary memory.
- **Requires input/output operations**: Data transfer involves additional steps.

Applications:

- Long-term storage
- Operating system installation
- Software installation
- Data backup
- Media storage

Types of Memory Hierarchy

This Memory Hierarchy Design is divided into 2 main types:
External Memory or Secondary Memory: Comprising of Magnetic Disk, Optical Disk, and Magnetic Tape i.e. peripheral storage devices which are accessible by the processor via an I/O Module.
Internal Memory or Primary Memory: Comprising of Main Memory, Cache Memory & CPU registers .
This is directly accessible by the processor.



Memory Hierarchy Design

Characteristics of Memory Hierarchy

- **Capacity:** It is the global volume of information the memory can store. As we move from top to bottom in the Hierarchy, the capacity increases.
- **Access Time:** It is the time interval between the read/write request and the availability of the data. As we move from top to bottom in the Hierarchy, the access time increases.
- **Performance:** Earlier when the computer system was designed without a Memory Hierarchy design, the speed gap increased between the CPU registers and Main Memory due to a large difference in access time. This results in lower performance of the system and thus, enhancement was required. This enhancement was made in the form of Memory Hierarchy Design because of which the performance of the system increases. One of the most significant ways to increase system performance is minimizing how far down the memory hierarchy one has to go to manipulate data.
- **Cost Per Bit:** As we move from bottom to top in the Hierarchy, the cost per bit increases i.e. Internal Memory is costlier than External Memory.

Primary memory	Secondary memory
The primary memory of a computer is the main memory that is utilized to store data temporarily.	Secondary memory defines to additional storage devices that are utilized to store data permanently.
Primary memory is temporary.	Secondary memory is permanent.
Primary memory is faster than secondary memory because it is directly accessible to the CPU.	Secondary memory is non-volatile, which means it retains data even when the power is off.
Primary memory is directly accessible by Processor/CPU.	Secondary memory is not directly accessible by the CPU.
Nature of Parts of Primary memory varies, RAM- volatile in nature. ROM- Non-volatile.	It's always Non-volatile in nature.
Primary memory is volatile, which means it is wiped out when the computer is turned off.	Since it is non-volatile, data can be retained in case of a power failure.
Primary memory devices are more expensive than secondary storage devices.	Secondary memory devices are less expensive when compared to primary memory devices.
The memory devices used for primary memory are semiconductor memories.	The secondary memory devices are magnetic and optical memories.
It can hold data/information currently being used by the processing unit.	It can hold data/information that are not currently being used by the processing unit.
The capacity of primary memory is usually within the range of 16 to 32 GB.	It stores a considerable amount of data and information. The capacity of secondary memory ranges from 200 GB to some terabytes.
Primary memory is also known as Main memory or Internal memory.	Secondary memory is also known as External memory or Auxiliary memory.
It can be accessed by a data bus.	It can be accessed using I/O channels.
Examples: RAM, ROM, Cache memory, PROM, EPROM, Registers, etc.	Examples: Hard Disk, Floppy Disk, Magnetic Tapes, etc.

Units of Memory

Memory units are used to measure the size and represent data. Some of the commonly used memory units are:

1. Bit

The first memory location in a computer is bit. The smallest measurement unit for data held in primary memory and storage devices is a bit. Out of the binary values 0 and 1, a bit can only have one.

The smallest measurement unit for data in primary memory and storage devices.

Represents binary values 0 and 1.

2. Nibble

It means the group of 4 bits.

3. Word

It is a fixed number of bits, it is different from computer to computer, but the same for each device.

Compute store information in the form of words.

A fixed number of bits that varies across computers but remains consistent within each device.

Used to store information in computers.

4. Bytes

The fundamental unit used to measure data is the byte. It has 8 bits in it. A byte can therefore represent 2 * 8 or 256 values. They determine the size of files, documents, photos, and other kinds of data.

The fundamental unit for measuring data, consisting of 8 bits.

Represents 256 values and determines file, document, photo, and data sizes.

Units of Memory Cont..

5. Kilobyte

1024 bytes is equal to one kilobyte. It is widely used to denote small file sizes and data storage capacities. One kilobyte can hold a small image or around 1024 characters of text.

- It frequently shows up in text documents, spreadsheets, and small image files.
- Equal to 1024 bytes.
- Denotes small file sizes and storage capacities.
- Can hold small images or around 1024 characters of text.

6. Megabyte

A megabyte is 1024 kilobytes in size. It contains more info as compared to a kilobyte. A megabyte can hold longer texts, high-resolution images, and short audio clips. It is used to calculate the size of files comprising music and short films, software packages, and documents. Megabytes are still important and frequently used, even though larger units of measurement are being used more frequently as a result of the growing number of data files.

- Comprising 1024 kilobytes.
- Contains more information compared to a kilobyte.
- Holds longer texts, high-resolution images, and short audio clips.
- Measures file sizes of music, short films, software packages, and documents.

Units of Memory Cont..

7. Gigabyte

- 1024 megabytes is equal to one gigabyte. It has a substantial amount of data storage space. Larger files, such full photo albums, high-definition movies, and software programs can fit within a gigabit. The storage capabilities of hard drives, solid-state drives, and other forms of data storage devices are routinely assessed utilizing this technique.
- Equal to 1024 megabytes.
- Offers substantial data storage space.
- Suitable for larger files, such as full photo albums, high-definition movies, and software programs.

8. Terabyte

- A terabyte is made up of 1024 gigabytes. It has a substantial amount of data storing capacity. A terabyte can hold a lot of data in large databases, massive media collections, and enterprise-level storage systems. It is frequently used by data centers, cloud storage services, and external hard drives with large storage capacities. As the demand for large-scale data processing and storage grows, terabytes are becoming more and more important.
- Comprising 1024 gigabytes.
- Provides substantial data storing capacity.
- Holds large databases, media collections, and enterprise-level storage systems.

Units of Memory Cont..

9. Petabyte

A petabyte is a colossal unit of data storage capacity. A petabyte may hold massive amounts of data, including significant video libraries, sizable databases, and sizable collections of high-resolution pictures. It is often used in data centers, cloud storage, and scientific research that uses a lot of data.

- A colossal unit of data storage capacity.
- Stores massive data quantities, like video libraries and large databases.

10. Exabyte (1024 petabytes)

An exabyte is equal to one EB. It has a substantial amount of data storage space. Exabytes can store vast film archives, massive data warehouses, and global internet traffic. It is extensively used in large-scale scientific simulations, cloud computing infrastructures, and enterprise-level storage systems.

- Equal to 1024 petabytes.
- Holds vast film archives, data warehouses, and global internet traffic.

Units of Memory Cont..

11. Zettabyte (1024 exabytes)

A zettabyte. It represents a capacity for data storage that is almost unimaginable. Zettabytes have the capacity to store unfathomably large amounts of data, including worldwide internet content, long-term archival storage, and in-depth global data analysis.

- Represents an almost unimaginable data storage capacity.
- Stores worldwide internet content, long-term archival data, and extensive global analysis.

12. Yottabyte

1024 zettabytes make up a yottabyte (abbreviated YB). It stands for an incredible amount of data storage. Unimaginable amounts of data, such as the equivalent of storing all of the material on the internet numerous times or tracking vast amounts, may be stored in yottabytes.

- Comprising 1024 zettabytes.
- Stands for an incredible amount of data storage.
- Can hold vast amounts equivalent to storing internet content numerous times.


Thank you!