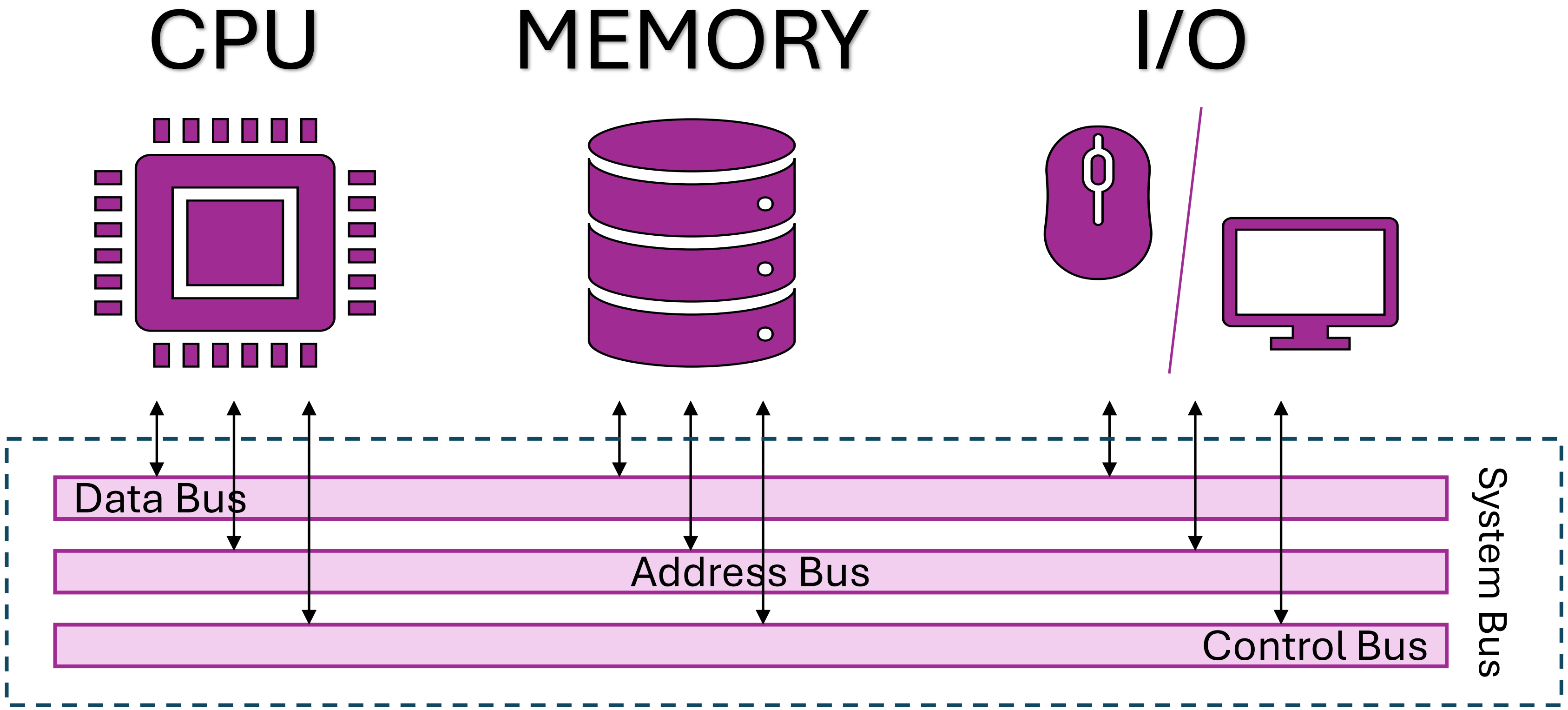
2. Computer Architectures

CPU, Memory, and I/O

Summary

- The Von Neumann Model
- System Bus
- CPU
- Instruction Cycle
- Memory
- MDR and MAR Registers
- FETCH and STORE Operations
- INPUT and OUTPUT Devices
- INPUT/OUTPUT Devices

The Von Neumann Model



System Bus

The **system bus** typically connects the CPU, RAM, I/O peripherals, and other system components. It allows for the **transfer of data**, **instructions**, and **control signals** between these components, enabling them to **collaborate** and **interact**.

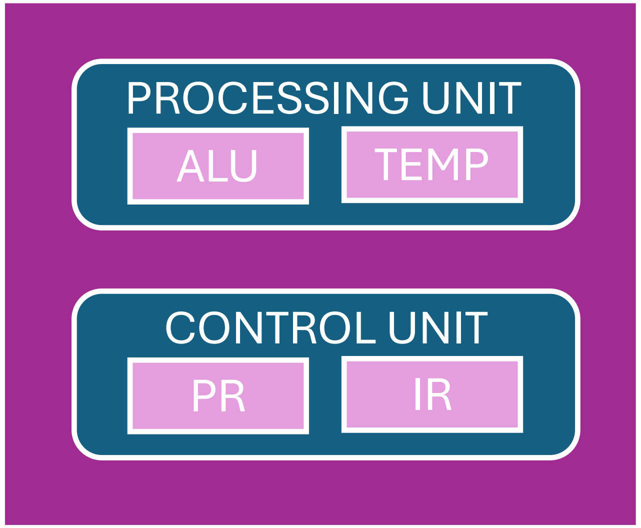
- **Data Bus:** Transmits binary data from one component to another. The data bus lines allow the transfer of information such as program instructions, data to be processed, or calculation results.

- **Address Bus:** Indicates a specific location in memory or I/O peripherals. The address bus allows system components to specify the source or destination of the data to be transferred.

- **Control Bus:** Carries control signals that coordinate and manage the operations of various system components. Control signals can include synchronization signals, interrupt signals, operation enable or disable signals, and status signals.

CPU

The **CPU (Central Processing Unit)** is the primary component of a computer that performs most of the processing inside the system. It **executes instructions** from programs, **performs calculations**, and **manages data flow** to and from other components like memory and peripherals. The CPU is often referred to as the **"brain"** of the computer, as it handles all critical tasks to ensure the system operates efficiently.



CPU – Processing Unit

The **processing unit** is a central part of the CPU responsible for **executing instructions** and managing the **data processing** tasks. It includes several key components that work together to perform **arithmetic** and **logical** operations, **control instruction flow**, and **store intermediate results**.

- **ALU (Arithmetic Logic Unit)**

The ALU is a critical component of the CPU that performs **arithmetic** operations (such as addition, subtraction, multiplication, and division) and **logical** operations (such as AND, OR, NOT, and XOR). It is essential for executing calculations and making decisions based on logical comparisons.

- **TEMP (Temporary Registers)**

Temporary registers (TEMP) are **small**, **fast storage locations** within the CPU used to hold **intermediate data** and **results** during instruction execution. They provide quick access to frequently used values and help optimize the processing speed by reducing the need to access slower main memory.

CPU – Control Unit

The **control unit** is a crucial part of the CPU that **directs the operation of the processor**. It **interprets instructions** from the program, **generates control signals**, and **coordinates the activities** of the CPU's other components, ensuring that instructions are executed in the correct sequence and timing.

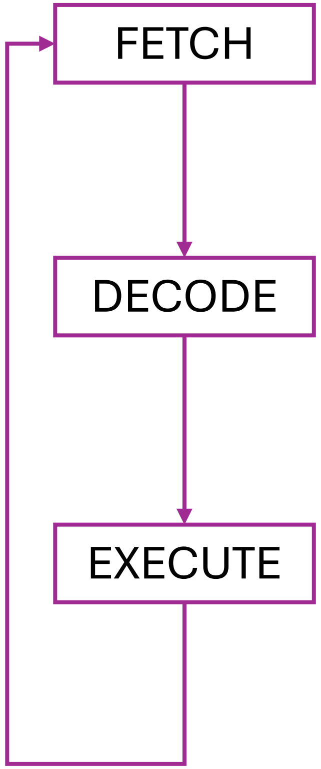
- **PC (Program Counter)**

The Program Counter (PC) is a register within the control unit that holds the **address of the next instruction to be executed**. It increments after each instruction is fetched, ensuring a sequential flow of program execution, unless a jump or branch instruction alters the flow.

- **IR (Instruction Register)**

The Instruction Register (IR) is a component of the control unit that **temporarily** holds the **current instruction being executed**. It stores the fetched instruction from memory, allowing the control unit to decode and process it, directing the necessary actions to other parts of the CPU.

Instruction Cycle



- **Fetch:**

The CPU retrieves the next instruction from memory, as indicated by the Program Counter (PC).

The instruction is loaded into the Instruction Register (IR).

- **Decode:**

The control unit interprets the fetched instruction in the IR.

It determines the operation to be performed and identifies the necessary operands.

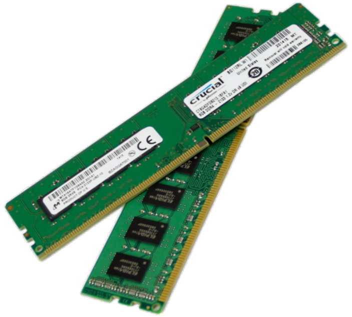
- **Execute:**

The CPU performs the operation specified by the instruction.

This may involve arithmetic or logical operations carried out by the ALU, data transfer, or control operations.

Memory/1

In the Von Neumann architecture, memory typically refers to **RAM (Random Access Memory)**, which is a critical component that **stores** both **data** and **program instructions temporarily** while the computer is running.



0000	
0001	
0010	
0011	10100010
0100	
0101	
0110	
	⋮
1101	10100010
1110	
1111	

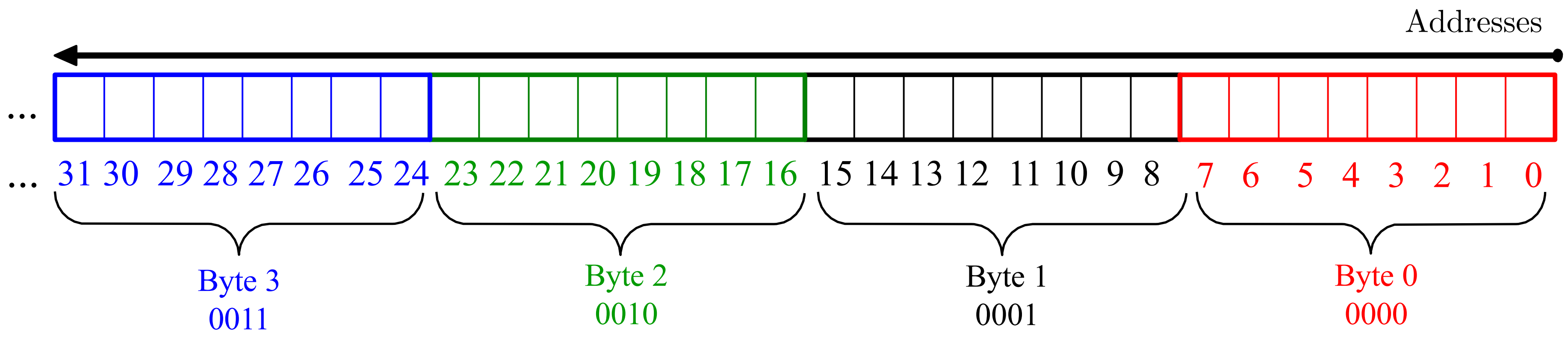
When a device is operating, both the **program** (e.g., Word application) and the **associated data** (e.g., the DOC document you are writing) are **stored in memory (RAM)**.

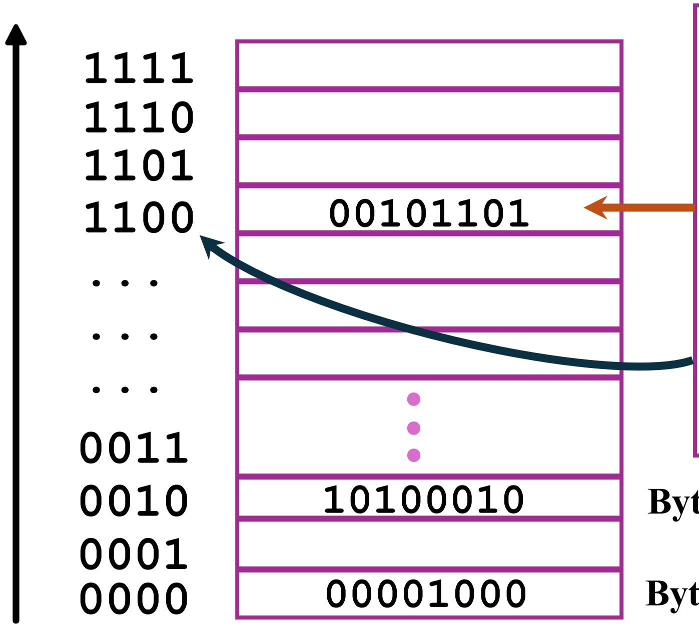
The **basic operations** that can be performed with memory include:

STORE: To save information.
FETCH: To retrieve information.

These operations are facilitated by the addressing mechanism known as **memory addresses**.

Memory/2





Memory is organized as a **sequence of cells**, each composed of a **single byte** (**8 bits**). A memory cell is characterized by:

A value, which represents the cell's content (expressed as a binary number).

An address, which represents the starting address of the set of cells related to a specific application (expressed as a binary number).

Byte 2, value 162, address 0010

Byte 0, value 8, address 0000

ICTAcademy

INFORMATION AND COMMUNICATIONS TECHNOLOGY

MDR and MAR Registers

- **MDR (Memory Data Register)**

The Memory Data Register (MDR), also known as the Memory Buffer Register (MBR), is a register within the CPU that temporarily **holds data being transferred to or from memory**.

When data is read from memory, it is first loaded into the MDR before being processed by the CPU. Similarly, when data is written to memory, it is placed in the MDR before being stored in the specified memory location. The MDR acts as a buffer, facilitating smooth data transfer between the CPU and memory.

- **MAR (Memory Address Register)**

The Memory Address Register (MAR) is a register within the CPU that **holds the address of the memory location** to be accessed next.

During the fetch stage of the instruction cycle, the MAR contains the **address of the instruction to be fetched**.

During data operations, the MAR holds the **address of the data to be read from or written to memory**. The MAR ensures that the correct memory location is accessed, enabling precise data retrieval and storage.

FETCH and STORE Operations

The operation of the Memory module involves the following two steps:

FETCH operation / To read from a location (A):

- Write the address (A) into the MAR.
- Send a "read" signal to the memory.
- Read the data from the MDR.

STORE operation / To write a value (X) to a location (A):

- Write the data (X) to the MDR.
- Write the address (A) into the MAR.
- Send a "write" signal to the memory.

INPUT and OUTPUT Devices

Input and Output (I/O) operations are crucial for allowing a computer system to **interact** with the **external environment** and **peripheral devices**.

Input:

- **Purpose:** To gather data from external sources and provide it to the computer for processing.
- **Devices:** Common input devices include keyboards, mice, scanners, microphones, and cameras.
- **Process:** Input devices convert user actions or external data into digital signals that the computer can process.

Output:

- **Purpose:** To present processed data from the computer to the external environment.
- **Devices:** Common output devices include monitors, printers, speakers, and projectors.
- **Process:** Output devices convert digital signals from the computer into a human-readable or perceivable form, such as text, images, sound, or physical output.

INPUT/OUTPUT Devices

Some devices function as **both input and output** devices, allowing for **two-way interaction** between the user and the computer system. These devices are versatile and can **both send data to** and **receive data from** the computer.

Examples of I/O Devices:

- **Touchscreen Monitors:**

- Input: The user can interact with the screen by touching it, which sends input signals to the computer.

- Output: The screen displays visual information from the computer.

- **External Hard Drives:**

- Input: Data can be written to the drive from the computer.

- Output: Data can be read from the drive to the computer.

- **Modems:**

- Input: Receive data from the internet.

- Output: Send data to the internet.