**Overview of Microcomputer Structure and Operation**

Table of Contents

[Simple Microcomputer 2](#_Toc83226662)

[System Bus 2](#_Toc83226663)

[Address Bus 2](#_Toc83226664)

[Data Bus 3](#_Toc83226665)

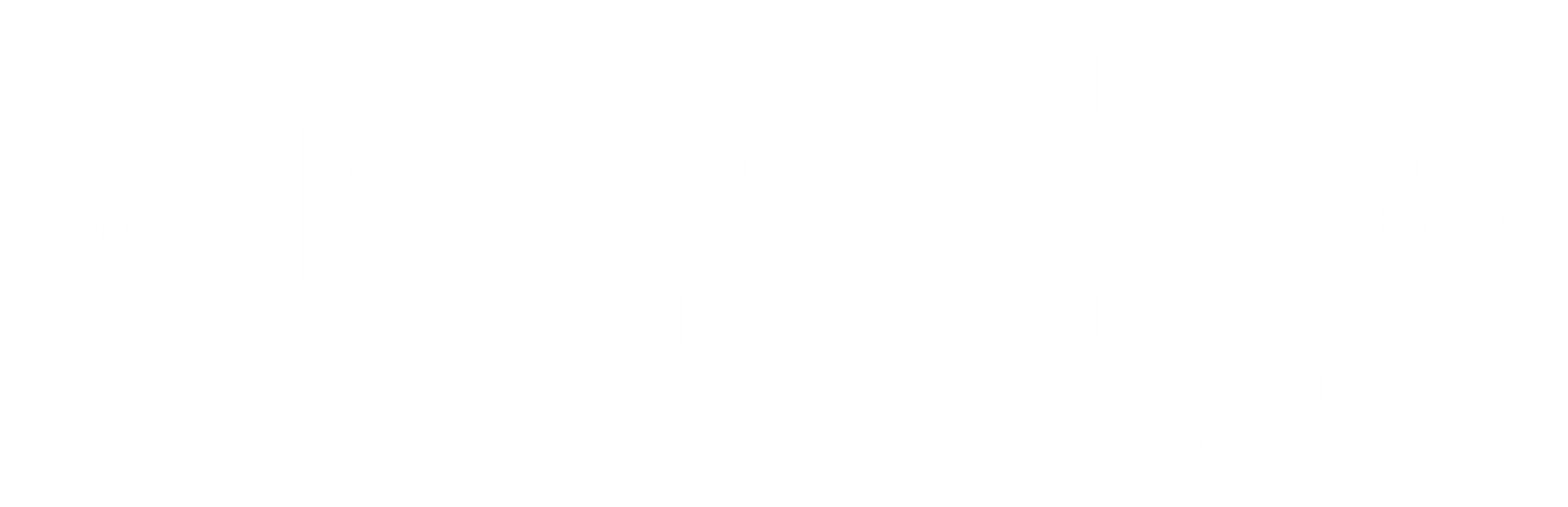
[Control Bus 4](#_Toc83226666)

[Fetching and Execution 5](#_Toc83226667)

[Fetching an Instruction 5](#_Toc83226668)

[Executing an Instruction 6](#_Toc83226669)

## Simple Microcomputer



The block diagram of a microcomputer shows the components present in a typical microcomputer. We mostly wish to concentrate on the **CPU**, but the CPU needs to interact with other components as well, so we need to know a bit about those.

Most of the components and their work is already known, so instead, we will be looking into some **design considerations** that must be made for each of the components.

### System Bus

A **Bus** is a collection of wires that creates the connectivity between different components of a computer. The **System Bus** is the combination of the **Address Bus**, the **Data Bus** and the **Control Bus**.

### Address Bus

The **Address Bus** carries the address of a specific **memory location** or **I/O element** from the CPU. This means that the Address Bus is **unidirectional**.

The Address Bus could have **multiple wires** between the CPU and another IC, such as the RAM. Each of these wires is called an **address line**, and the total number of address lines is called the **address bus size**. This is typically 16, 20, 24, 32 or 36. The 8085 microprocessors for example, has an address bus size of 16, while the 8086 microprocessor has an address bus size of 20.

The address bus size determines the maximum number of **bits** a specific memory location can have and thus determines the total **number** of memory locations that can be addressed. With 16 bits, we can identify of memory. Each location stores of memory, or .

### Data Bus

The memory locations specified by the CPU contain some data, and this data is carried in between the RAM and the CPU or in between the I/O devices and the CPU by the **Data Bus**. Thus, it is a **bidirectional** bus.

The **size** of the Data Bus varies from one microprocessor to another, but it usually matches the **word length**. Usually, this is a multiple of **8**, although earlier microprocessors had 4-bit data buses as well, which were called nibbles. Modern processors are **64-bit** processors, which means the word length is 64 bits.

### Control Bus

The **Control Bus** carries control information or **control signals**. The most important of these control signals are the **Read** and **Write** signals.

Let’s say a CPU has an pin. This is the read pin and the bar indicates that it works with logic-0. If we send a read signal to an I/O Device, it will send us data through the data bus. Similarly, if we sent a write signal through the pin, data going through the data bus will be received by the I/O device.

Some other signals include:

* **Interrupts**, such as if an input device needs to send data, in which case it will need to interrupt the CPU first, since the CPU is probably executing some other program. The interrupt request is sent through one pin while the interrupt acknowledgement is sent through another.
* Bus Control signals for **DMA**, where the I/O Device will request DMA through one pin and the CPU will acknowledge it with another.

## Fetching and Execution

A **Fetch Cycle** is a computer cycle during which an **instruction** is brought in from memory and stored in the **instruction register**. The **program counter** is also moved so that it points to the next instruction.

An **Execute Cycle** is a computer cycle during which the actual execution of an instruction occurs. The actual actions that occur depend on the **instruction** and the **addressing mode** being used to access any data that is required.

### Fetching an Instruction

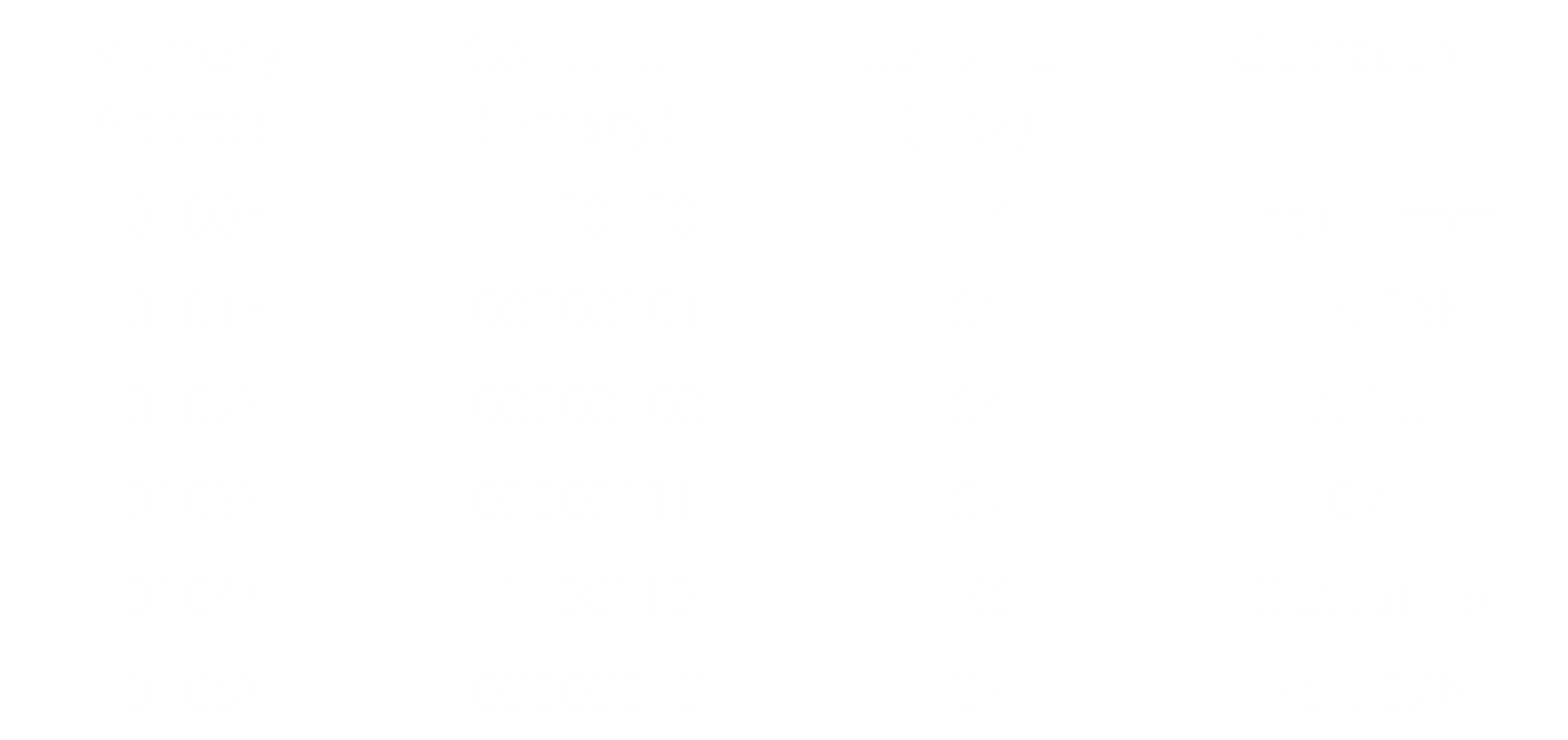
1. The **instruction pointer (IP)** or **program counter** is a special register that holds the address of the next instruction that needs to be fetched. This address is sent along the **Address Bus** to the **Memory Access Register (MAR)**, which is a special register in the **RAM**.
2. The data in the memory location specified by the **MAR** is sent along the **Data Bus** to the **Instruction Register**, which is a special register in the CPU that holds the instruction to be executed. This instruction will go on to be decoded and executed.

### Executing an Instruction

Consider a program with **three instructions**:

1. Input a value from a keyboard connected to the port at address 05h (IN AL, 05h).
2. Add 07h to the value read in (ADD AL, 07h).
3. Output the result to a display connected to the port at address 02h (OUT 02h, AL).

The corresponding **Machine Language** instructions will be:



Each of these instructions take . The microprocessor however, needs to load the instructions one byte at a time, meaning 2 bytes need to be loaded before each instruction can be executed.