

# **SEL-0415    Introdução à Organização de Computadores**

## **Estrutura de um Computador** **Aula 6**

**Profa. Luiza Maria Romeiro Codá**

# INTRODUÇÃO

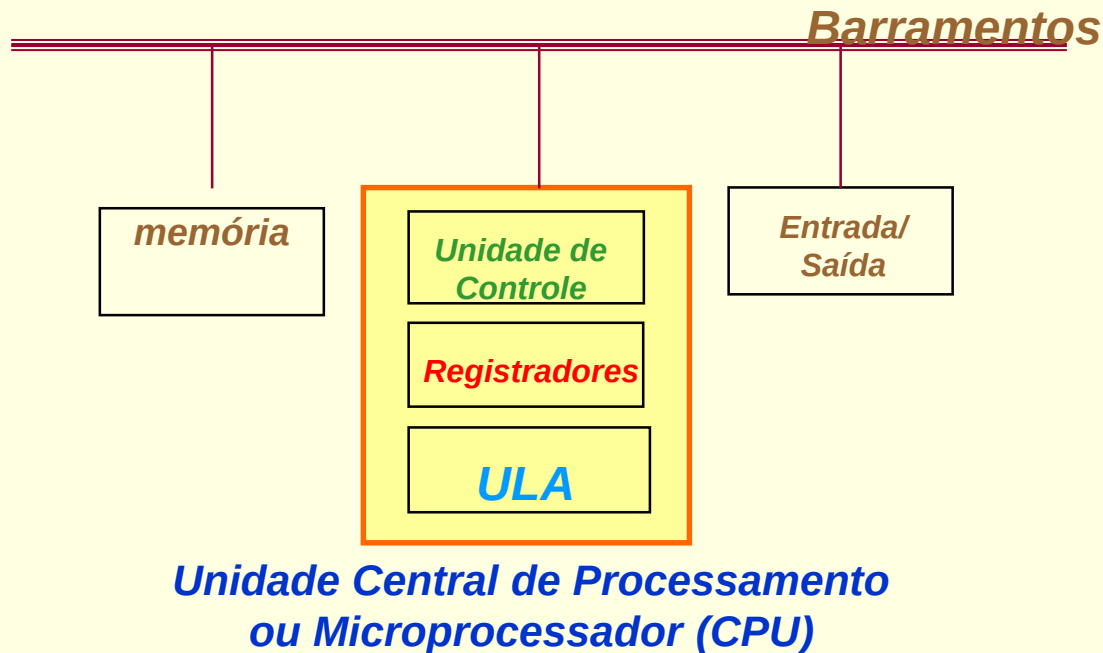
- **Organização** → implementação do hardware, componentes, construção dos dispositivos → Pouco importante ao programador;
- **Arquitetura** → Tamanho das memórias e barramentos, conjunto de instruções, modos de endereçamentos → Muito importante ao programador;
  - **Ex.** O fabricante define elementos da arquitetura de uma família de processadores, cada um com uma diferente organização, que afeta seu desempenho e custo: Família Intel MCS-51

# MODELO DE VON NEUMANN

## PRINCÍPIOS

A arquitetura de um computador consiste de 4 partes principais:

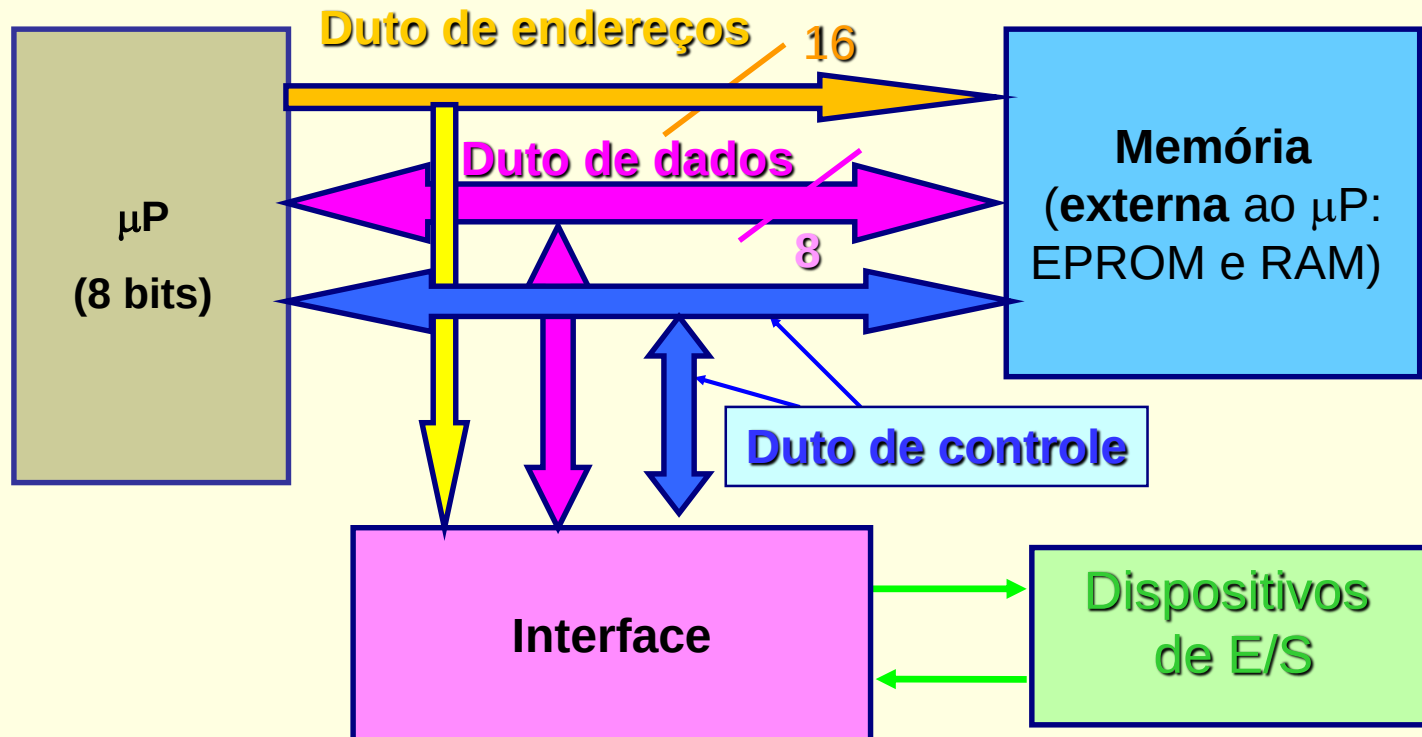
- *Unidade Central de Processamento (CPU)*
- *Memória*
- *Dispositivos de entrada/saída.*
- *Dispositivos de conexão (barramentos)*



# Microcomputador de 8 bits

## Arquitetura de Von Neumann

Diagrama em blocos mostrando um microprocessador ( $\mu P$ ) de 8 bits, interligado às demais unidades funcionais :



# MODELO DE VON NEUMANN

## Função de cada bloco:

### μP (ou CPU):

- executa instruções lidas da memória de Programa (ROM ou EEPROM)
- controla todo o fluxo de informação no duto de dados (gera sinais de /RD e /WR)
- Monitora os demais blocos do sistema

## Memória

Há dois tipos principais :

**Memória ROM(EEPROM)** - é do tipo não volátil, somente de leitura e contém o **conjunto de instruções (programa)** do sistema.

**Memória RAM:** é do tipo volátil, de leitura e gravação, é usada para **armazenamento dos dados** gerados durante a execução do programa

# MODELO DE VON NEUMANN

## Função de cada bloco (continuação):

### **Barramentos:**

Permitem o tráfego de dados e instruções entre os diversos componentes do computador

### **Dispositivos de E/S:**

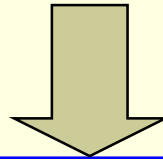
Permitem a comunicação do uP com o meio externo

### **Interfaces:**

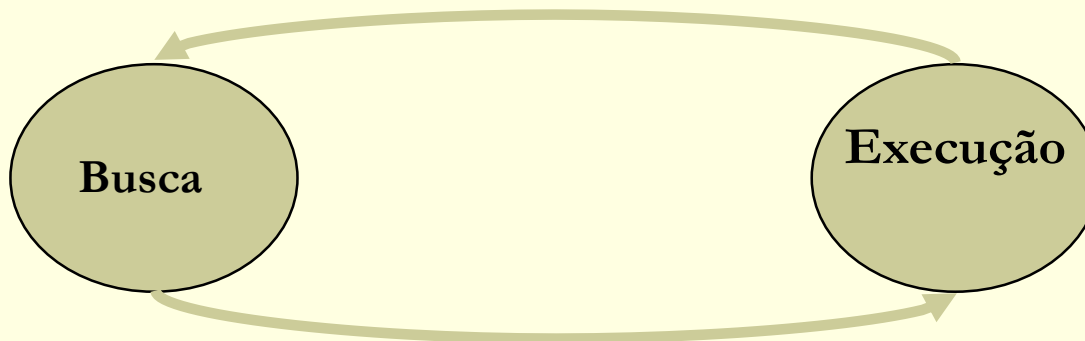
adequam os sinais do uP aos dispositivos de I/O no que se refere a tensão, corrente, frequência, etc.

# MODELO DE VON NEUMANN

“O programa que direciona as atividades da CPU é armazenado na mesma memória em que estão os dados, que devem ser manipulados pelo programa”



o computador é uma máquina de programas armazenados sequencialmente executados



# Sistema Microprocessado:

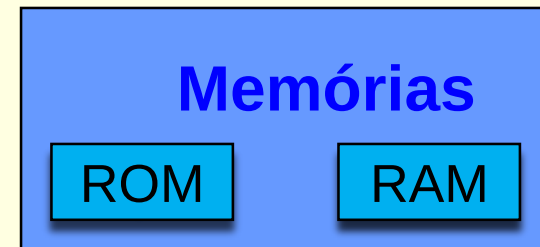
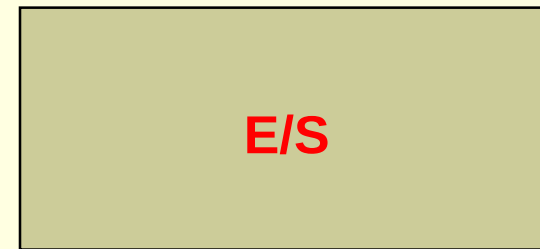
## 1. Memórias



# Sistema Microprocessado:

## 1. MEMÓRIAS

Microprocessador (CPU)



# Sistema Microprocessado:

## 1. MEMÓRIAS

---

- **Memória de Programa** (Tipo ROM ou EEPROM)
  - Instruções
  - Dados não-voláteis
  
- **Memória de Dados** (Tipo RAM)
  - Registradores Especiais (SFR – Special Function Registers)
  - Dados temporários (GPR – General Purpose Registers)

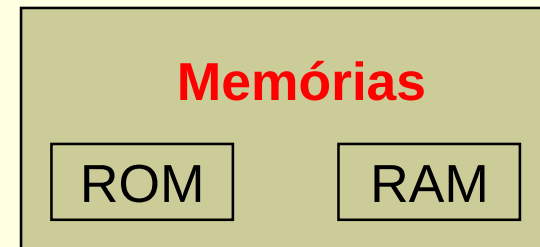
# Sistema Microprocessado :

## 2. Barramentos

# Modelo de Von Neumann:

## 2. Barramentos ou Dutos

### Microprocessador (CPU)



# Sistema Microprocessado:

## 2. BARRAMENTOS



- Canal de comunicação entre o microprocessador e os periféricos e memórias
- Todos periféricos e memória compartilham o mesmo canal de comunicação
- $\mu$ P comunica-se apenas com um dispositivo ou memória por vez
- Tamanho ➡ determina quantos bits podem ser transmitidos por vez (ex.: barramento de dados de 16 bits, de 32 bits...)
- Controle: temporizador interno à CPU

**OBS:** O duto ou barramento são divididos em três partes: barramento de dados, barramento de endereçamento e barramento de controle<sup>17</sup>

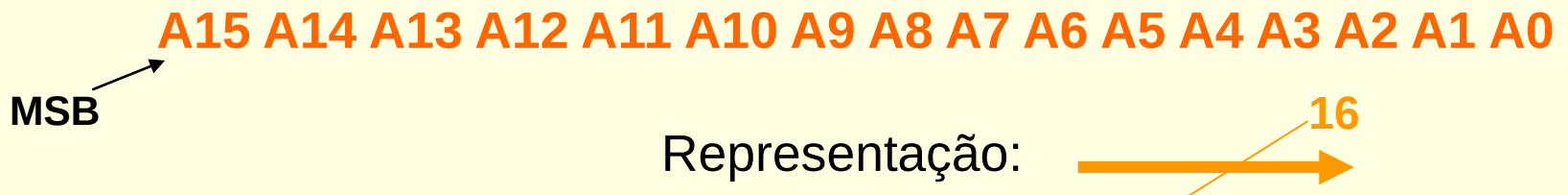
# Sistema Microprocessado:

## 2. BARRAMENTOS

- **Duto de Endereços**

é gerado pelo microprocessador

constituído por **Ne** bits de endereço. Exemplo para  $N_e = 16$  bits:



define a máxima capacidade de endereçamento do  $\mu P$

**(Espaço de Endereçamento):**

$2^{16} = 64$  Kbytes, onde 1 Kbytes = 1024 bytes

destina-se ao endereçamento e seleção de memórias e dispositivos de E/S

# Sistema Microprocessado:

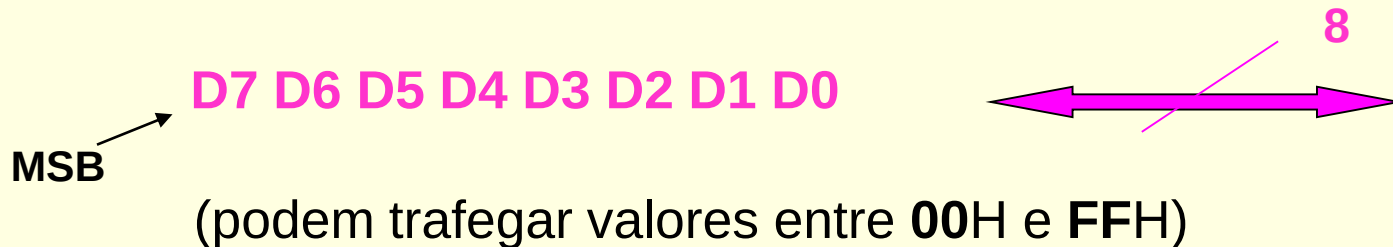
## 2. BARRAMENTOS

- **Duto de Dados**

É bidirecional : o microprocessador pode gravar ou ler dados de memórias ou dispositivos de I/O

Define o tamanho da palavra de memória **Nd** a ser usada

Exemplo para **Nd = 8 bits**:



Tipo de Informação que trafega nesse duto:

- **Instrução** (código binário do programa – ling. de máquina)
- **Dados** (temporários)

# Sistema Microprocessado:

## 2. BARRAMENTOS

- **Duto de Controle**

**Contém sinais diversos:**

- Controle de leitura e escrita
- entrada para solicitar estado de espera (aumentar duração de ciclos de leitura e escrita)
- entradas para solicitação de interrupção
- entradas para solicitação de DMA (Direct Memory Access)

Cada microprocessador pode ter parte desse conjunto de sinais no duto de controle, todos eles, ou ainda algum outro sinal específico.



# Sistema Microprocessado:

## 2. Barramentos

### Tipos:

#### ■ Síncronos

- Sincronizadas com um sinal de *clock*
- Implementação mais simples
- Todos os dispositivos devem se comunicar com a mesma velocidade
- Desvantagem: se há diferença de velocidade entre os dispositivos conectados

#### ■ Assíncronos

- Sem *clock*
- Vantagem: cada dispositivo pode se comunicar com uma velocidade diferente
- Implementação mais complexa: regras (protocolos) para início e término de comunicação

# Sistema Microprocessado:

## 3. CPU / Microprocessador

# Microprocessador X Microcontrolador

**Microprocessador** → é um dispositivo lógico programável em um único chip de silício ( concebido sob a tecnologia VLSI, ULSI ou GSI) e é composto de 3 partes principais: **ULA**, conjunto de **registradores**, unidade de **controle**.

- capacidade de executar operações lógicas, aritméticas, e de controle (CPU).
- Inclui CPU + encapsulamento

# Microprocessador X Microcontrolador

- ❑ **Microprocessadores** precisam ser interligados com memória do tipo ROM e RAM, além dos dispositivos de E/S, para se tornarem operacionais
- ❑ **Microcontroladores** são dispositivos que possuem **em um único chip**: microprocessador, memórias, barramentos, dispositivos de E/S e interfaces (para interligar periféricos);

# Microprocessador X Microcontrolador

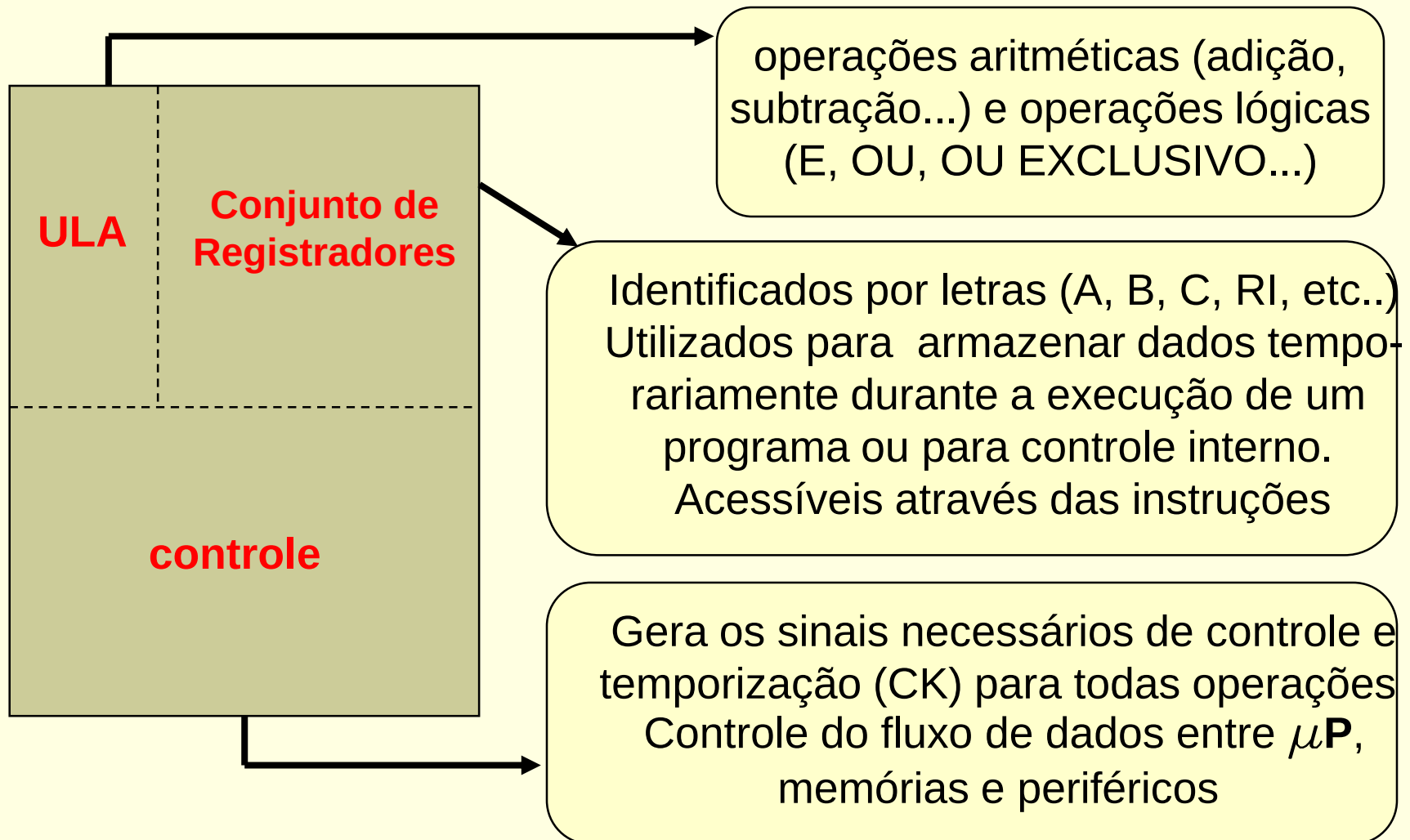
**Microcontrolador** → é um circuito integrado que possui internamente um microprocessador e todos os periféricos essenciais ao seu funcionamento, como:

- **Memória de programa** – geralmente uma memória do tipo ROM onde serão armazenadas as informações de programa,
- **Memória de dados** – geralmente uma memória do tipo RAM, onde ficarão armazenadas as informações de dados que o programa irá utilizar.
- **Portas paralelas de entrada e saída**
- **Temporizadores**
- **Conversores A/D e D/A**
- **Lógica para controle de interrupção**
- **Comunicação serial**

### 3. CPU / Microprocessador

- Dispositivo de lógica programável usado para:
  - Controlar processos
  - Ligar/desligar dispositivos
- Opera com 0s e 1s, controlado por um CLOCK
- O  $\mu\text{P}$  executa um programa que se encontra em memória do tipo **ROM (ou EEPROM)**
- Programa (armazenado em memória) ➡ contém conjunto de instruções em padrão binário ➡ **Linguagem de máquina**
- A execução é sequencial: uma única instrução por vez é executada.
- Cada  $\mu\text{P}$  tem seu próprio conjunto de instruções.

### 3. CPU / MICROPROCESSADOR



# 3. CPU / Microprocessador

## ■ ULA:

- **Operações lógicas e aritméticas:** soma, subtração, AND, OR, NAND, NOR, XOR, CMA, CMP;
- **Flags:** bits que sinalizam os resultados de operações lógicas e aritméticas.



### 3. CPU / Microprocessador

**Flags ➡ bits indicadores de estado da ULA:**

- contidos no registrador PSW (Palavra de *Status* do Programa – “*Program Status Word*”)
- são colocados em “1” ou “0” dependendo do resultado das operações da CPU
- algumas instruções testam flags para ver se elas devem ser executadas
- flags típicas: SIGN, CARRY, ZERO, OVERFLOW
- bit de flag usualmente se refere ao estado do **acumulador A**
- bit de sinal = MSB do A após a operação da ULA

# 3. CPU / Microprocessador

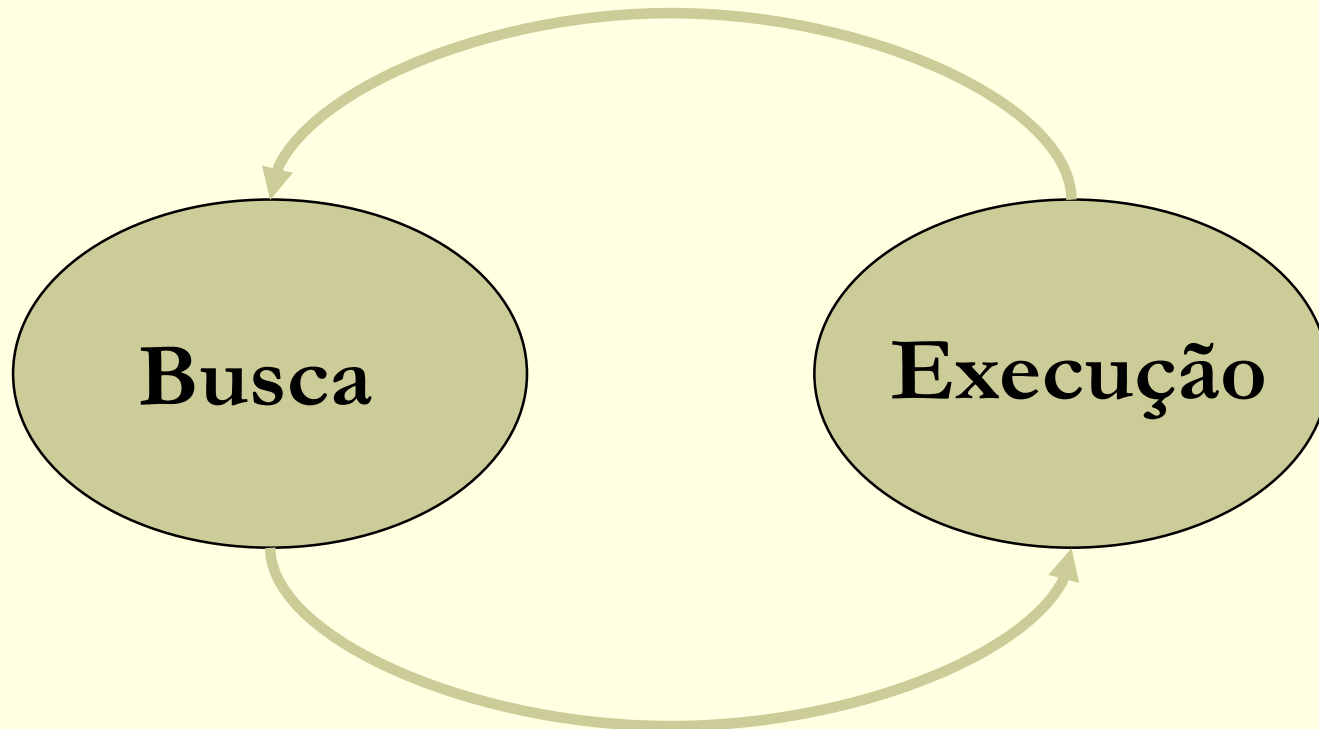
## ■ Clock (CK):

- Gera sinais de sincronismo interno;
- Permite sequência ordenada de eventos;

■ **Ciclo de máquina:** tem a duração de vários períodos de **CK**. (Ex. 8051 = 12 pulsos; PIC = 4 pulsos)

- A busca de uma instrução na memória e sua execução, pode gastar um ou mais ciclos de máquina (depende da arquitetura do  $\mu\mathbf{P}$ )

### 3. CPU / Microprocessador



**Ciclos de Máquina**

# 3. CPU / Microprocessador

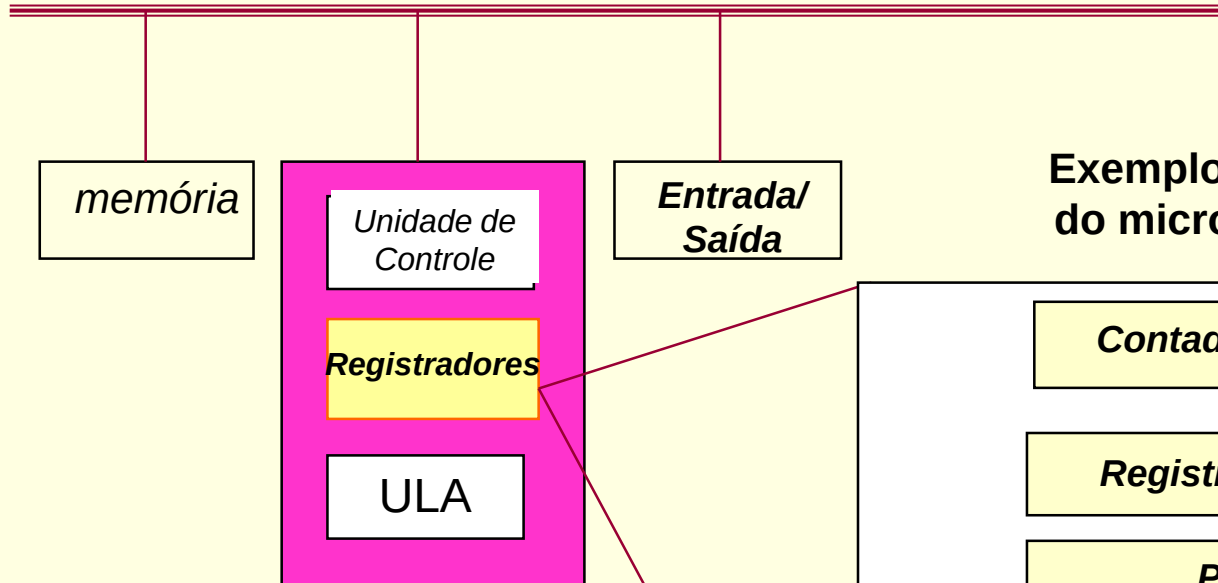
---

## ■ Registradores

- Normalmente são internos à CPU, alta velocidade
- permitem o armazenamento de valores temporários, intermediários ou informações de comando
- Cada um tem uma função própria

# 3. CPU / Microprocessador

*Barramento*



**Exemplo de registradores do microcontrolador 8051**

|                          |      |
|--------------------------|------|
| Contador de programa     | PC   |
| Registrador de Instrução | RI   |
| Ponteiro                 | DPTR |
| Acumulador               | A    |
| Timers                   | TMR  |
| Ponteiro de Pilha        | SP   |

nos **microprocessadores** os registradores são internos à CPU, e nos **microcontroladores** parte deles podem estar mapeados em memória RAM, dedicada a esses registradores.

# 3. Microprocessador / CPU

## registrador e memória principal

- Registradores se localizam no interior de um microprocessador, enquanto a memória principal é externa à CPU;
- Um registrador armazena um número limitado de bits, geralmente uma palavra de memória;
- Em algumas arquiteturas, alguns “registradores” têm funções específicas, geralmente de configuração e operação do microprocessador, que são chamados de SFRs (SFR – Special Function Register);

### 3. Microprocessador / CPU

- **Ponteiros:** são registradores que contêm informação de endereço
  - Ponteiro de Programa : **PC**
  - Ponteiro de Dados : **DPTR** (microcontrolador 8051)
  - Ponteiro de Pilha: **SP**
  
- **Pilha:** Área de Memória **RAM** para armazenamento de **endereço de retorno de subrotina ou interrupção**. Há também instruções que permitem o seu uso pelo programador
- O ponteiro **SP** indica qual a última posição em que foi armazenado um dado na pilha

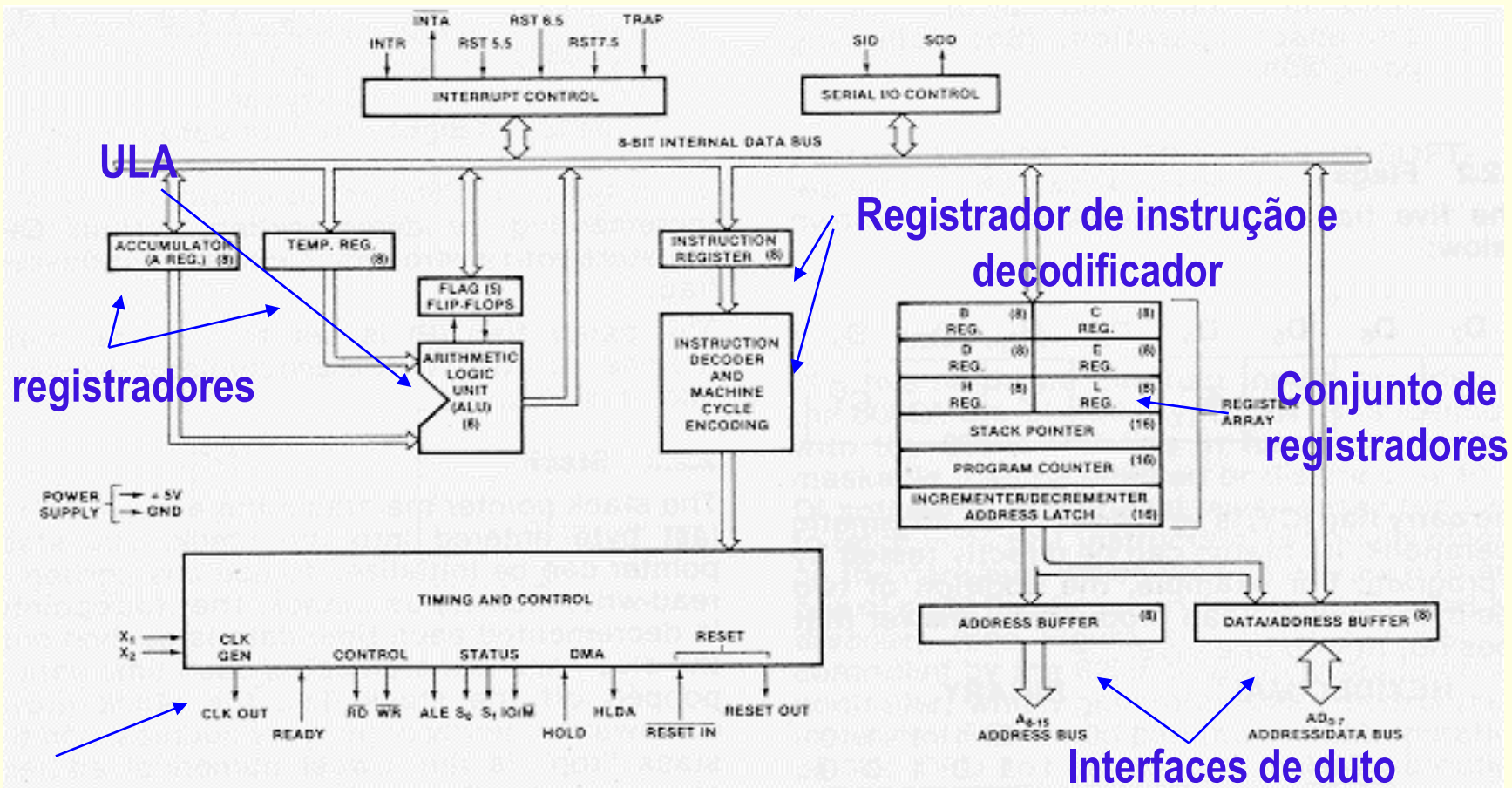
# 3. Microprocessador / CPU

## UC: Unidade de Controle

- Lê o opcode, que foi armazenado no **IR** (ou RI registrador de instruções );
- Elemento que garante a correta execução dos programas e a utilização dos dados corretos nas operações;
- Decodifica a instrução correspondente e gera os sinais para o processamento da mesma;
- Controla o acesso aos barramentos;
- Controla a execução de todas as operações no  $\mu P$ .



# EXEMPLO de Microprocessador: Intel 8085



Seção de temporização e controle

Interfaces de duto  
para o meio externo

# Sistema Microprocessado:

## 4. Dispositivos de E/S

# Sistema Microprocessado:

## ENTRADA e SAÍDA (E/S)

### (I/O - *Input/Output*)

---

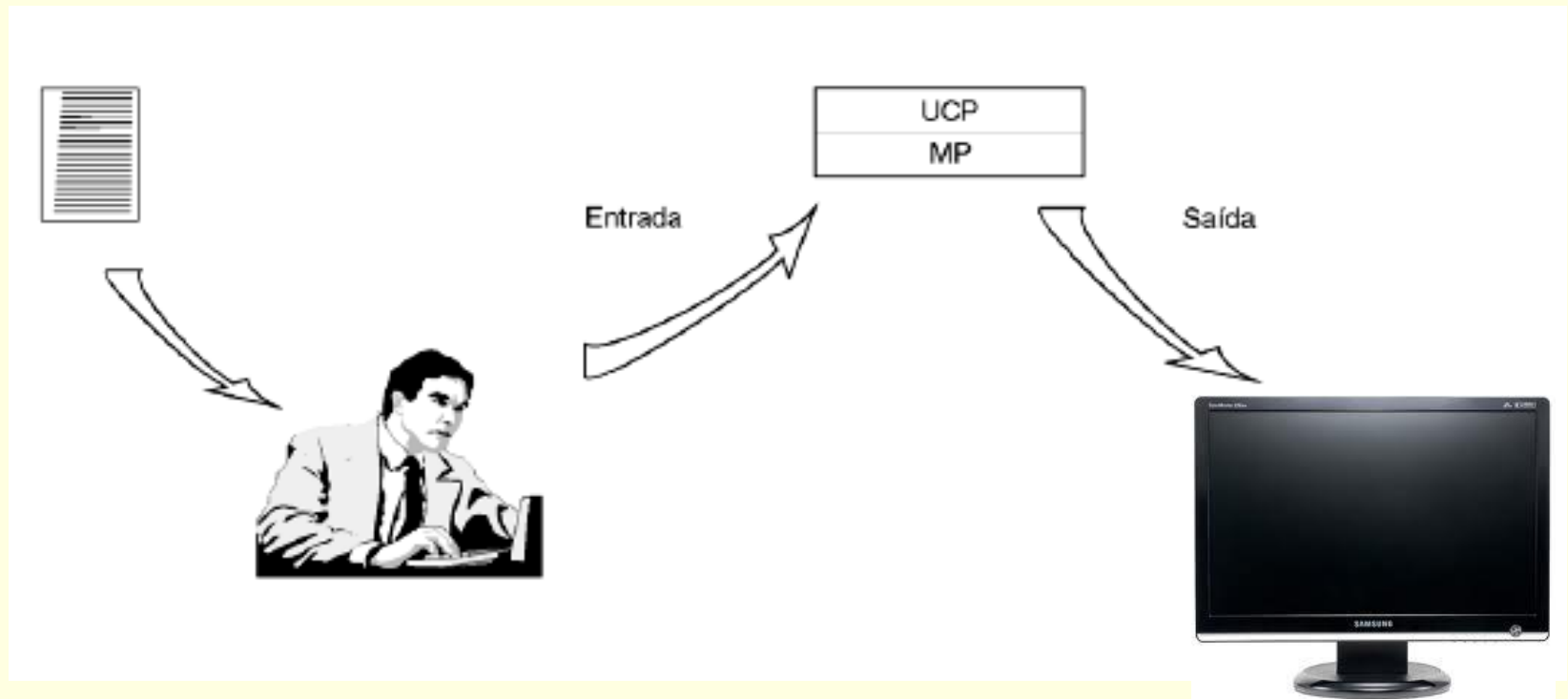
Função dos dispositivos de Entrada e Saída:

- Inserção dos dados (programa)
- Apresentação dos resultados
- Comunicação Homem/Máquina

# Sistema Microprocessado:

## ENTRADA e SAÍDA (E/S)

(I/O - *Input/Output*)



# Sistema Microprocessado:

## ENTRADA e SAÍDA (E/S)

### (I/O - *Input/Output*)

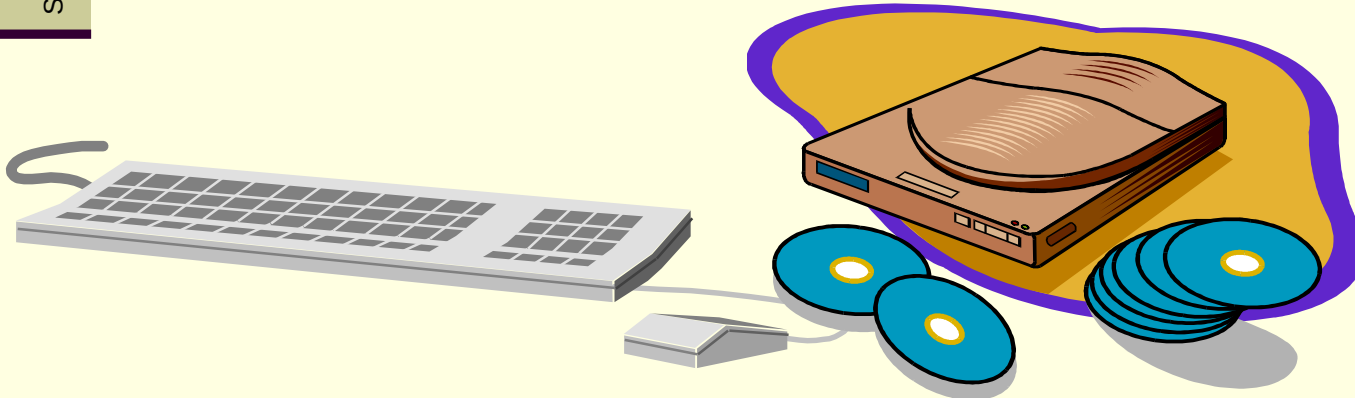
---

- **Entrada** ➡ Dispositivos (geralmente baseados em chaves) por onde informações entram na memória
  - Ex.: Teclados, Portas
- **Saída** ➡ Dispositivos que mostram o resultado da operação executada
  - Ex:
    - Monitores
    - Impressoras
    - Armazenamento secundário...

# Sistema Microprocessado: Dispositivos de Entrada

## Periféricos

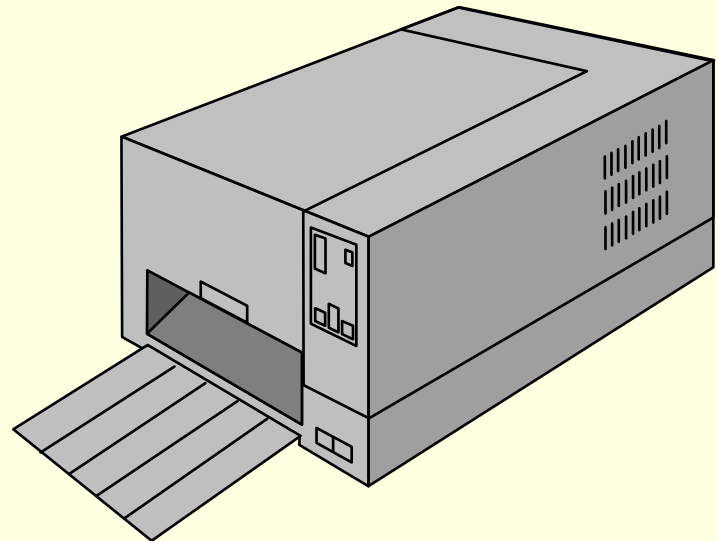
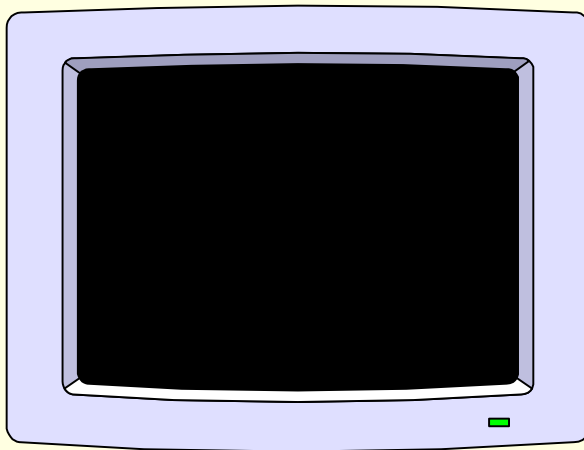
- Existem alguns que são especializados apenas em **ENTRADA**:
  - **Teclado** ➡ Lê os caracteres digitados pelo usuário
  - **MOUSE** ➡ Lê os movimentos e toque de botões
  - **Drive de CD-ROM** ➡ Lê dados de discos CD-ROM
  - **Microfone** ➡ Transmite sons para o computador
  - **SCANNER** ➡ Usado para "digitalizar" figuras ou fotos



# Sistema Microprocessado: Dispositivos de Saída

## Periféricos

- Outros especializados apenas em **SAÍDA**:
  - **Vídeo** ➡ Mostra ao usuário, na tela caracteres e gráficos
  - **Impressora** ➡ Imprime caracteres e gráficos
  - **Alto-falante** ➡ Realiza comunicação com o usuário através de som



# Sistema Microprocessado: Dispositivos de Entrada e Saída

## Periféricos

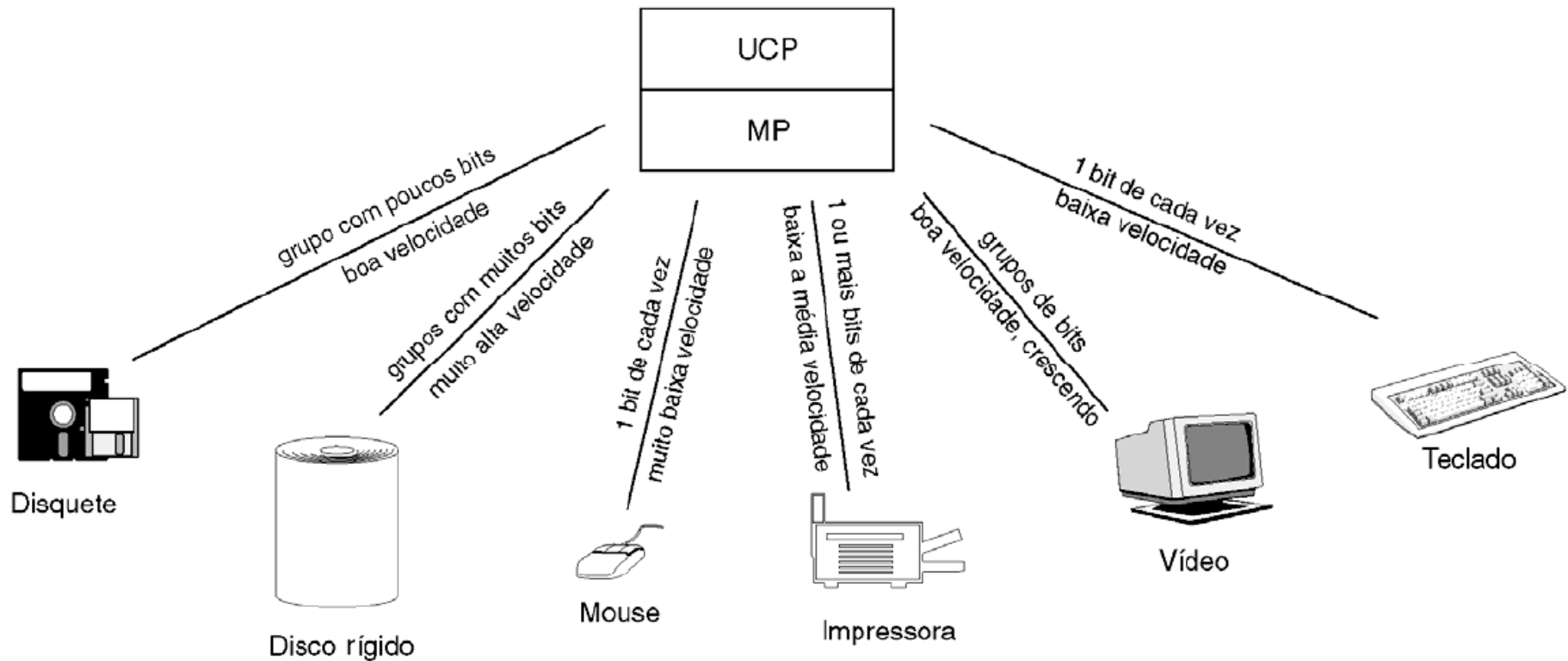
- Outros em ENTRADA E SAÍDA
  - Disco rígido - Grava e lê dados
  - Pen drive - Grava e lê dados em
  - CD-RW- Grava e lê dados em fitas magnéticas
  - MODEM - Transmite e recebe dados pela linha telefônica





# Sistema de PC:

## Dispositivos de Entrada e Saída



# Dispositivos de Entrada e Saída para Controle de Processos

## Periféricos

### ■ Menos Tradicionais (microcontroladores)

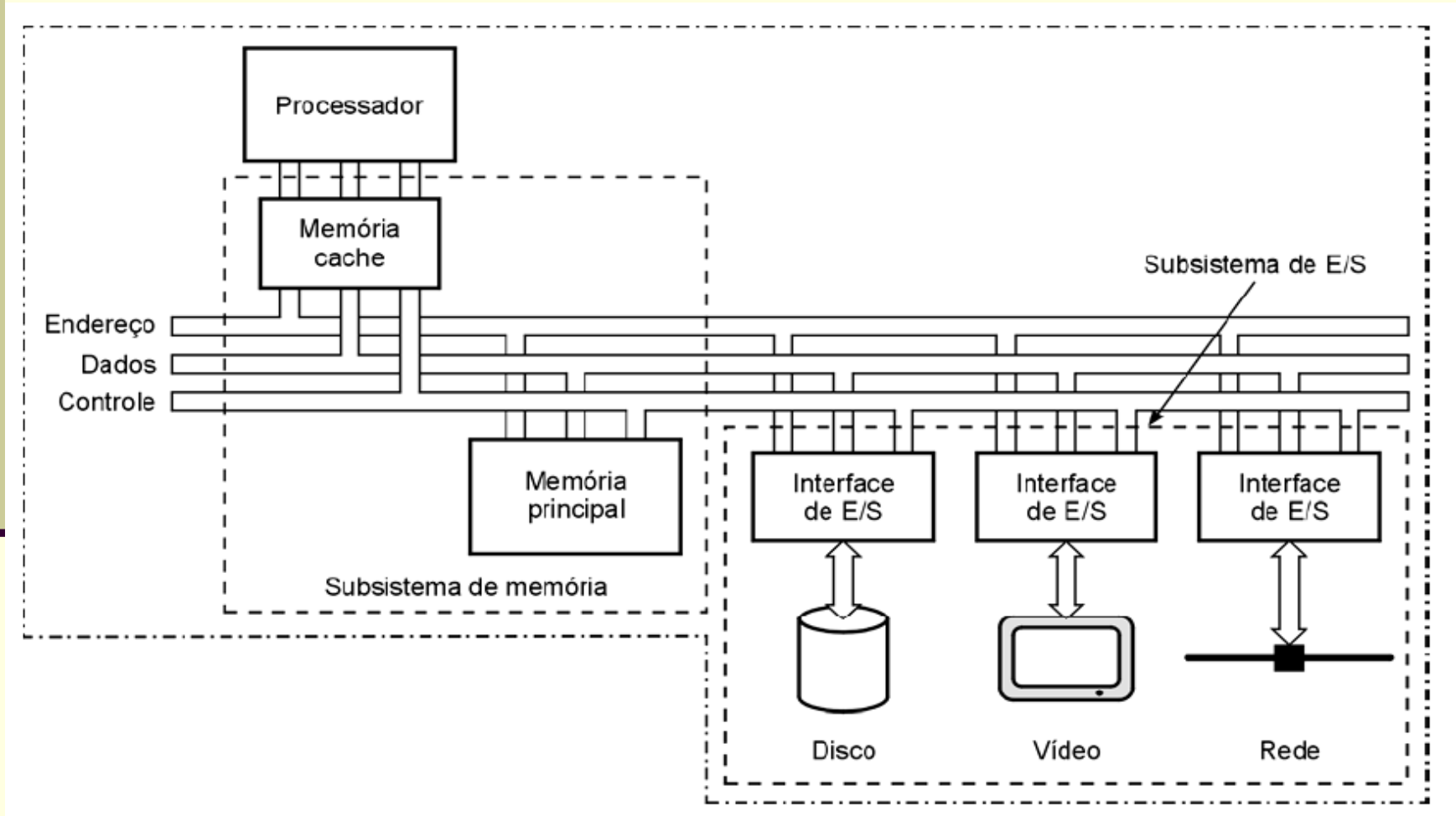
- Sensores
- Motores de Passo
- Fotocélulas
- Termostatos

# Sistema Microprocessado: Interfaces de Entrada e Saída

## Interfaces de (I/O)

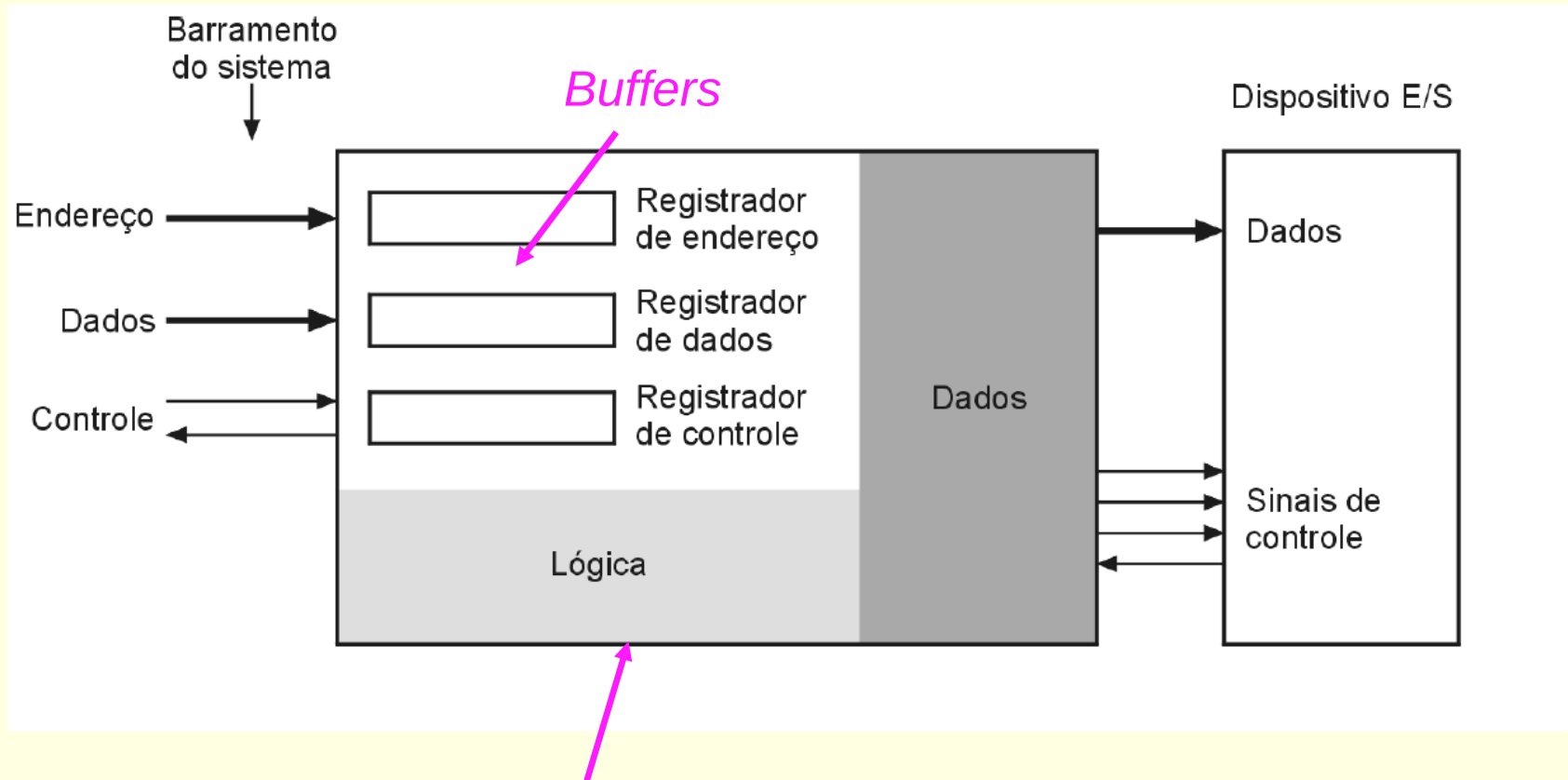
- Geralmente a CPU não pode se comunicar diretamente com os periféricos  $\Rightarrow$  a comunicação é feita com a ajuda de circuitos chamados de **Interfaces** ou **Módulos** de I/O
- Funções:
  - Presentes entre o barramento e o periférico
  - Compatibilidade entre os dispositivos e o  $\mu P$
  - Controle da comunicação
  - Ex.: controlador de vídeo, controlador de disco, etc...

# Sistema Microprocessado: Interfaces de Entrada e Saída



# Sistema Microprocessado

## Interface de Entrada e Saída



# Operações de Entrada em Saída no Processador

# Operações de I/O

## Métodos para realização de operações de I/O

- ◆ Três tipos principais:
  - ◆ Programada (*Pooling*)
  - ◆ Interrupção
  - ◆ Acesso Direto à Memória (DMA)

# EXEMPLO FIGURATIVO



O EMPREGADO ESTÁ  
LIMPANDO A CASA E TEM  
COMO FUNÇÃO RECEBER O  
RECADO DE QUEM LIGAR.





# EXEMPLO FIGURATIVO



**PROGRAMADA (ou Varredura)**  
(telefone SEM campainha): o empregado de tempos em tempos verifica se há alguém querendo lhe falar ao telefone

**INTERRUPÇÃO** (telefone COM campainha): o empregado pára de fazer o serviço quando o telefone toca, pois há alguém querendo lhe falar ao telefone



**DMA - ACESSO DIRETO À MEMÓRIA**  
(telefone COM campainha e COM secretária eletrônica): o telefone toca, a secretária eletrônica armazena o recado e o empregado pára de fazer o serviço quando lhe convier para ouvir o recado.

# Varredura

## I/O por Programa (Varredura)

- ◆ A CPU controla diretamente todas as etapas da comunicação
- ◆ O programa deve verificar os dispositivos de entrada e saída e parar o processamento durante a transmissão
- ◆ Subrotina de verificação dos dispositivos de entrada e saída
- ◆ Tempo de transmissão dos dispositivos de I/O são muito altos comparados ao  $\mu P$
- ◆ Processo muito pouco eficiente

# Interrupção

## I/O por Interrupção

- ◆ A CPU aguarda a interface de I/O requisitar uma transmissão
- ◆ Enquanto isso o  $\mu P$  pode realizar outras tarefas
- ◆ Quando a interface está pronta para a transmissão ela avisa o  $\mu P$
- ◆ O  $\mu P$  interrompe a atividade corrente e inicia a comunicação com o dispositivo de I/O
- ◆ Processo mais eficiente do que a operação por varredura, mas ainda sobrecarrega o  $\mu P$  durante a comunicação com o periférico

# INTERRUPÇÃO

1. Atende à acontecimentos assíncronos (imprevisível);
2. Não precisa esperar para que ele ocorra – o microprocessador não deixa de ser utilizado para outras funções;
3. Pode ser interna ou externa
4. Interna: divisão por zero, overflow, etc.
5. Externa: Interface de I/O

# INTERRUPÇÃO

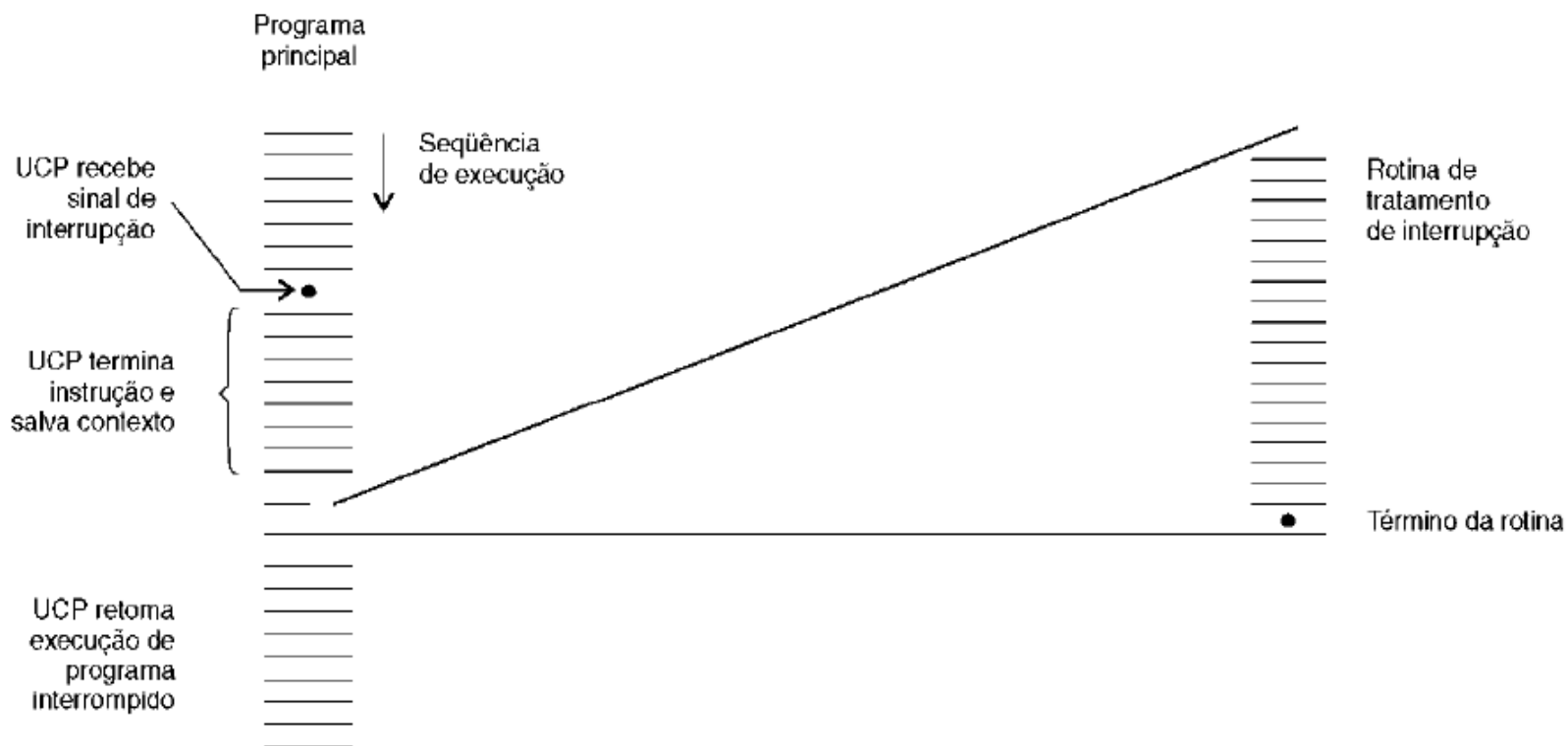
6. Um evento qualquer envia um sinal de pedido de interrupção (INTERRUPT REQUEST – IRQ) ao  $\mu P$  por meio de uma linha de controle do barramento externo do sistema
7. O  $\mu P$  pode aceitar ou rejeitar o pedido, gerando um sinal de reconhecimento de interrupção (INTERRUPT ACKNOWLEDGE – IACK) numa linha de controle do barramento externo do sistema
8. O  $\mu P$  pára a execução do programa (via hardware), grava o endereço de retorno (PC+1) na pilha e atende à subrotina de interrupção
9. Após a execução da subrotina de interrupção, a microprocessador volta ao ponto onde parou no programa principal
10. Nem sempre é possível prever o local exato de retorno da interrupção

# INTERRUPÇÃO

## **Diferença entre uma subrotina convencional e a subrotina de interrupção:**

- A subrotina convencional é chamada por uma instrução do microprocessador (instrução CALL), em posições definidas pelo programador, no programa principal.
- a subrotina de interrupção está relacionada à ocorrência de um evento que pode ser imprevisível .

# INTERRUPÇÃO



# INTERRUPÇÃO

- quando uma aplicação de microcomputador começa a se tornar mais complexa, as perdas de tempo começam a se tornar mais críticas. Se levar 50 microssegundos para ler um dado de um dispositivo externo e mais 50 microssegundos para transmitir um dado para o mesmo dispositivo externo, então um sistema microcomputador poderia executar uma centena de transferências de dados por segundo, mas não sobraria tempo para fazer mais nada.

**Portanto, existirão diversas aplicações onde o desperdício de tempo em processamento de interrupções se tornaria intolerável.**



# DMA

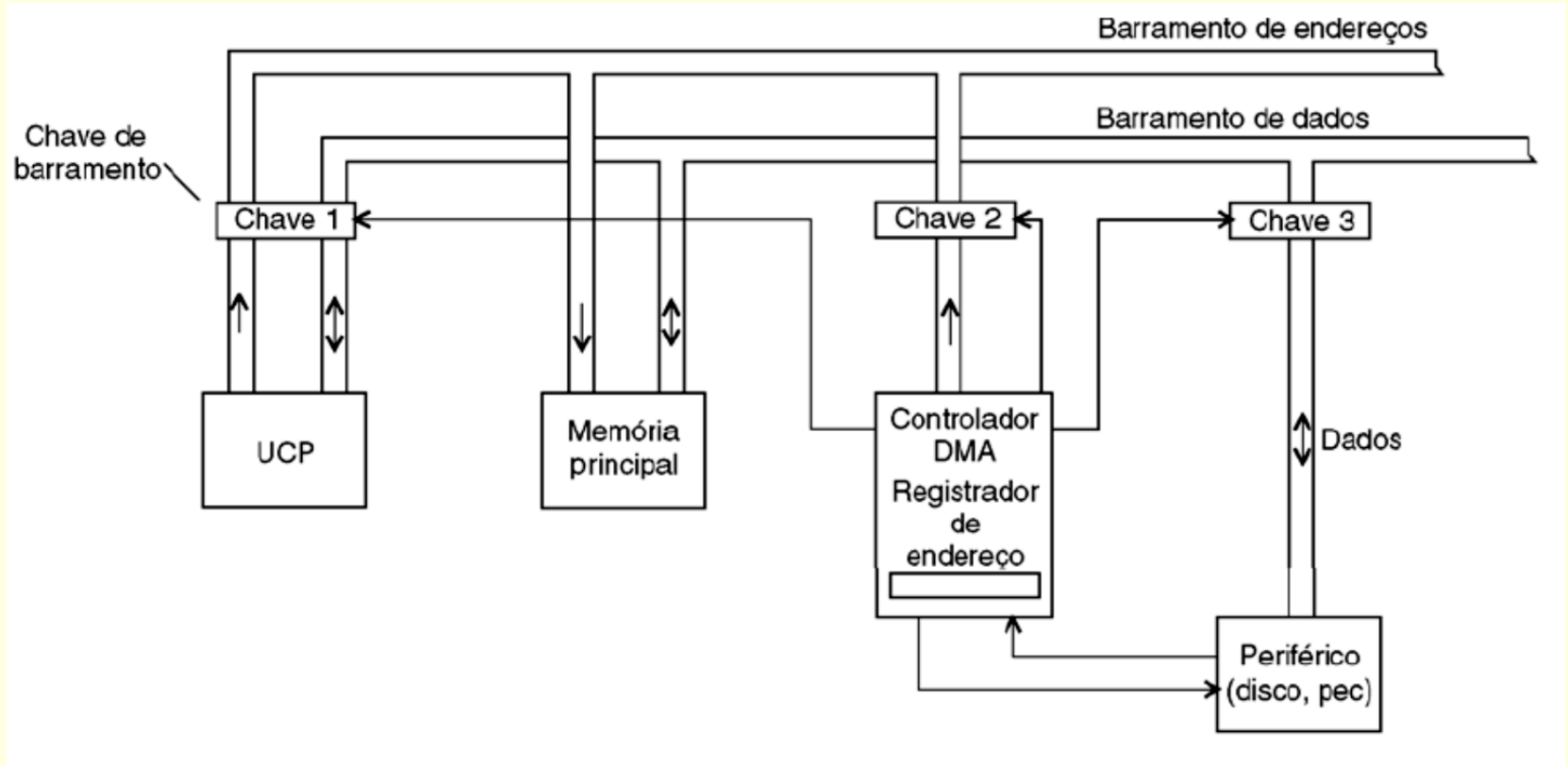
## I/O por DMA (Direct Memory Access)

- ◆ Permite a movimentação de dados entre os dispositivos de I/O e a memória do microcomputador sem envolver o processador nesta transferência
- ◆ Processo mais eficiente do que todos os outros, pois não utiliza o  $\mu P$  e não sobrecarrega o barramento.

# DMA - ACESSO DIRETO À MEMÓRIA

- ◆ Dispositivo de hardware dedicado à operação de transferência de dados entre um dispositivo de I/O e a memória;
- ◆ Coloca a saída do microprocessador em estado de alta impedância (desligado) para permitir a um dispositivo externo o Acesso Direto à Memória – *Bus Request*
- ◆ Acesso direto à memória (DMA) permite uma forma mais rápida de mover dados entre as portas de I/O e a memória.

# DMA - ACESSO DIRETO À MEMÓRIA



# FIM