



# GERENCIAMENTO DE ENTRADA E SAÍDA

Sistemas Operacionais  
Engenharia de Software  
Prof. Jeferson Silva

# INTERRUPÇÕES

No contexto de sistemas operacionais, uma interrupção é um evento assíncrono que interrompe o fluxo normal de execução do processador para que ele possa lidar com uma situação específica.

Essas situações podem ser originadas por diversos eventos, como:

- Dispositivos de E/S
- Exceções
- Temporizadores

# INTERRUPÇÕES

Outras interrupções:

- Interrupções podem ser geradas por outros eventos, como falhas de hardware, erros de software ou solicitações específicas de um processo.

# CONTROLADOR DE INTERRUPÇÃO

Função:

- Responder às interrupções geradas pelos dispositivos.
- Interromper o processador para tratar eventos de E/S.

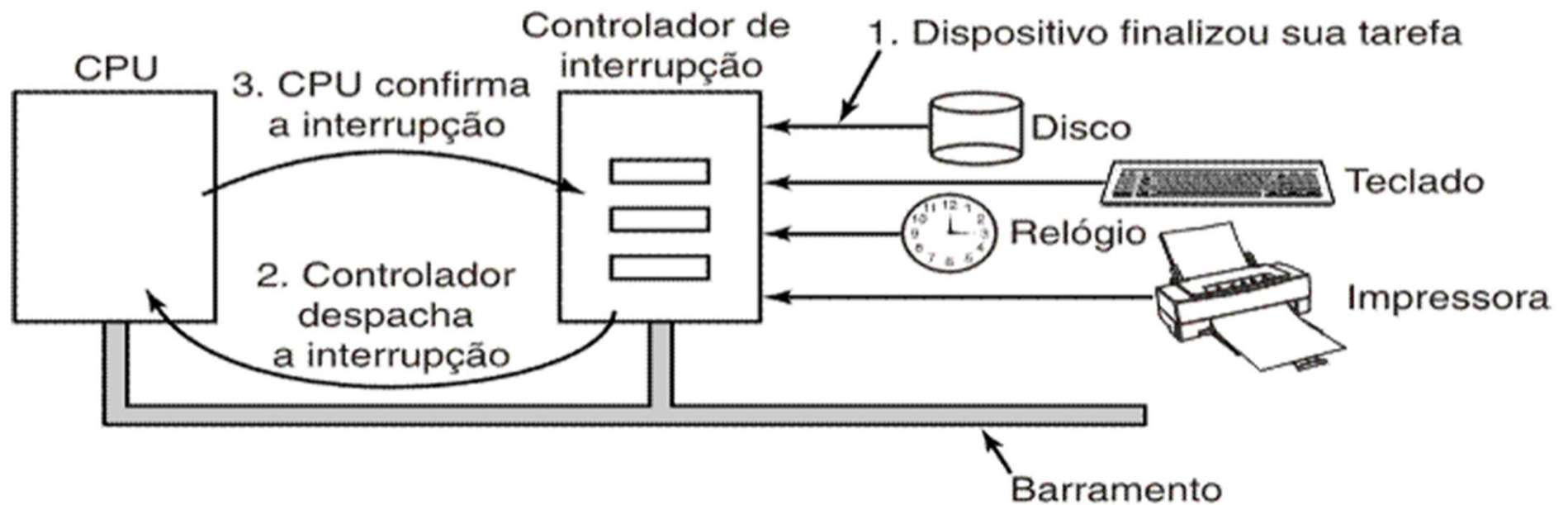
Exemplo:

- Interrupção de fim de transferência de dados.
- Interrupção de erro.

Papel do SO:

- Controlar e gerenciar as interrupções.

# CONTROLADOR DE INTERRUPÇÃO



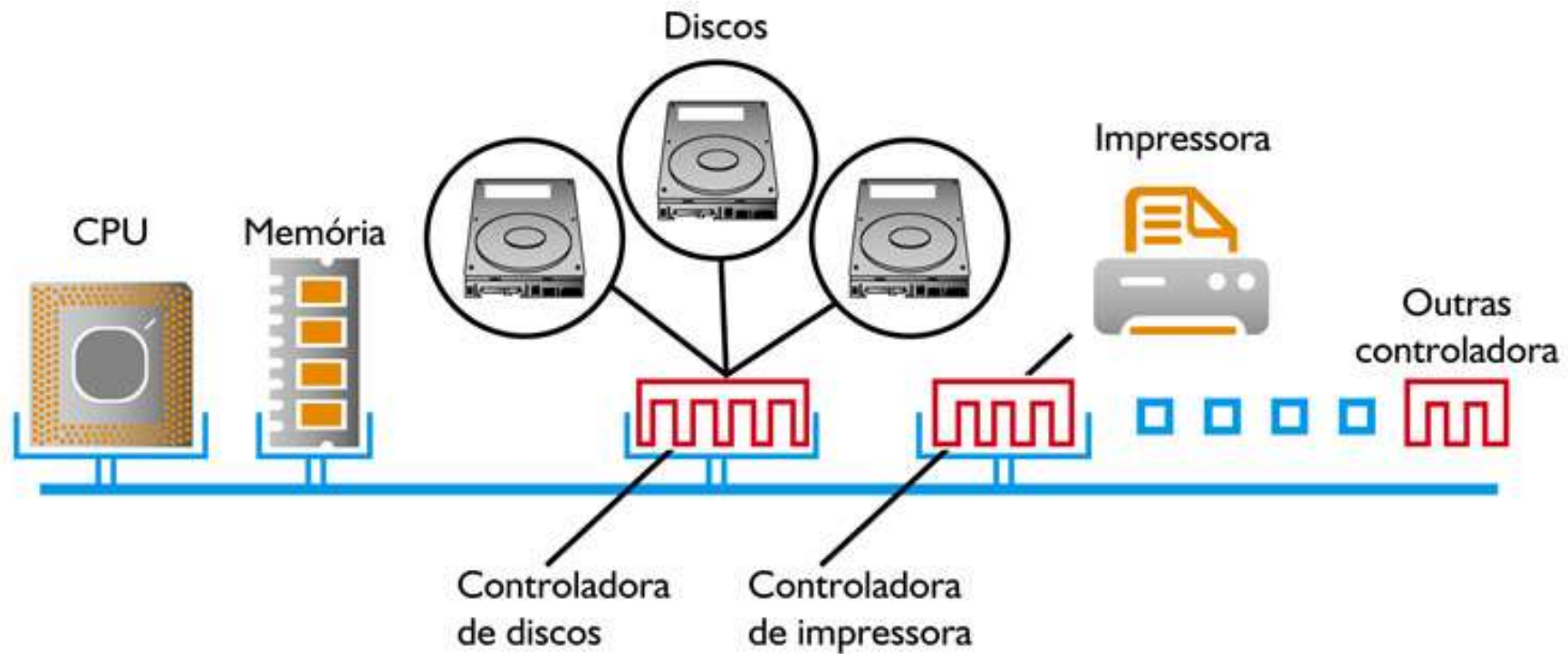
# GERENCIAMENTO DE ENTRADA E SAÍDA

O controle da entrada e saída (E/S – ou I/O, input/output) de dados dos dispositivos é uma das funções principais de um sistema operacional.

Para promover o compartilhamento seguro do uso dos recursos, contudo, não é permitido aos processos o acesso direto aos dispositivos de entrada e saída.

Assim, cabe ao SO oferecer serviços (chamadas de sistema) que permitam ler e escrever dados.

# GERENCIAMENTO DE ENTRADA E SAÍDA



# GERENCIAMENTO DE ENTRADA E SAÍDA

A interação dos programas com o SO para o acesso aos dispositivos pode ocorrer enviando e recebendo bytes de/para dispositivos de caractere, ou realizando operações de arquivos em dispositivos de bloco.



## FUNÇÕES DO GERENCIADOR DE E/S (I/O)

Gerenciar e controlar o acesso a dispositivos.

Garantir a segurança e o compartilhamento de recursos.

Fornecer interface uniforme para os programas.

# TIPOS DISPOSITIVOS DE E/S

Dispositivos de entrada de saída podem ser agrupados em duas categorias:

- Dispositivos de caractere
- Dispositivos de bloco

## DISPOSITIVOS DE CARACTERE

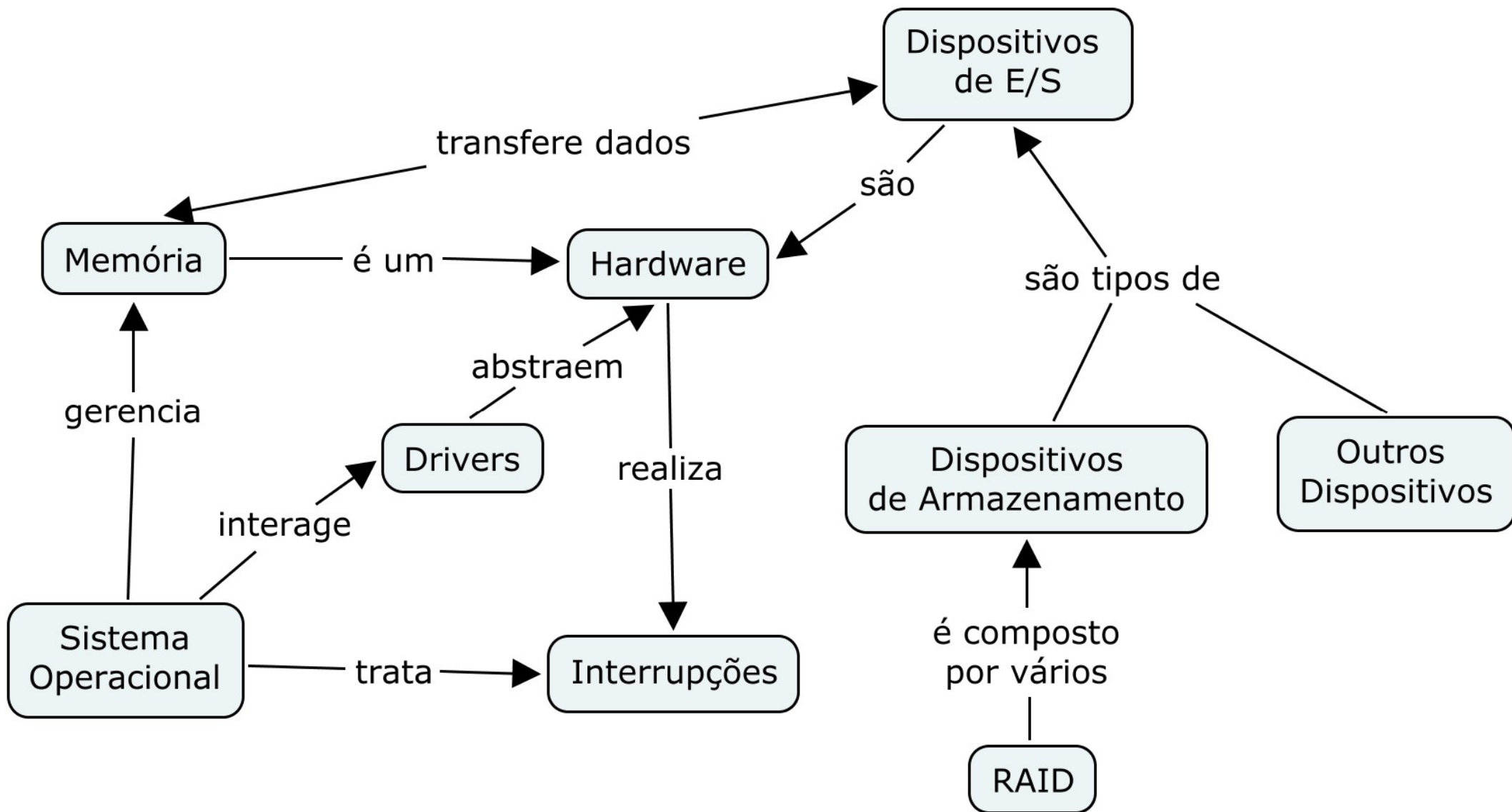
Dispositivos de caractere permitem a transferência de sequências de caracteres nas operações de leitura e escrita.

Exemplos desse tipo de dispositivo incluem terminais, impressoras de linha, etc., em que o acesso é puramente sequencial.

# DISPOSITIVOS DE BLOCO

Dispositivos de bloco, como os discos rígidos e as unidades de armazenamento em memória externa, manipulam informação em blocos de tamanho fixo, normalmente com tamanho múltiplo de 512 bytes.

Diferentemente dos dispositivos de caractere, contudo, dispositivos de bloco permitem acessar cada bloco de armazenamento de maneira independente.



# CONTROLOADORES DE DISPOSITIVOS

Unidades de E/S apresentam dois componentes, um componente mecânico, que corresponde ao dispositivo que realiza as operações, e um componente eletrônico, chamado de controlador de dispositivo, que é conectado ao computador e atua como interface para o dispositivo.

# CONTROLADORES DE DISPOSITIVOS

Alguns controladores podem intermediar a interação com mais de um dispositivos de E/S.

Um controlador de discos SATA, por exemplo, pode tratar da interação com vários discos SATA.

Do mesmo modo, um controlador USB pode lidar com vários dispositivos USB.

# FUNÇÕES DO CONTROLADOR

Receber comandos do SO.

Controlar o funcionamento do dispositivo.

Gerenciar a transferência de dados entre o dispositivo e a memória.



# TIPOS DE CONTROLADORES

Controlador de disco rígido.

Controlador de rede.

Controlador de vídeo.

Exemplos:

- Controlador SATA, Controlador USB.

# INTERCONEXÃO DE HARDWARE

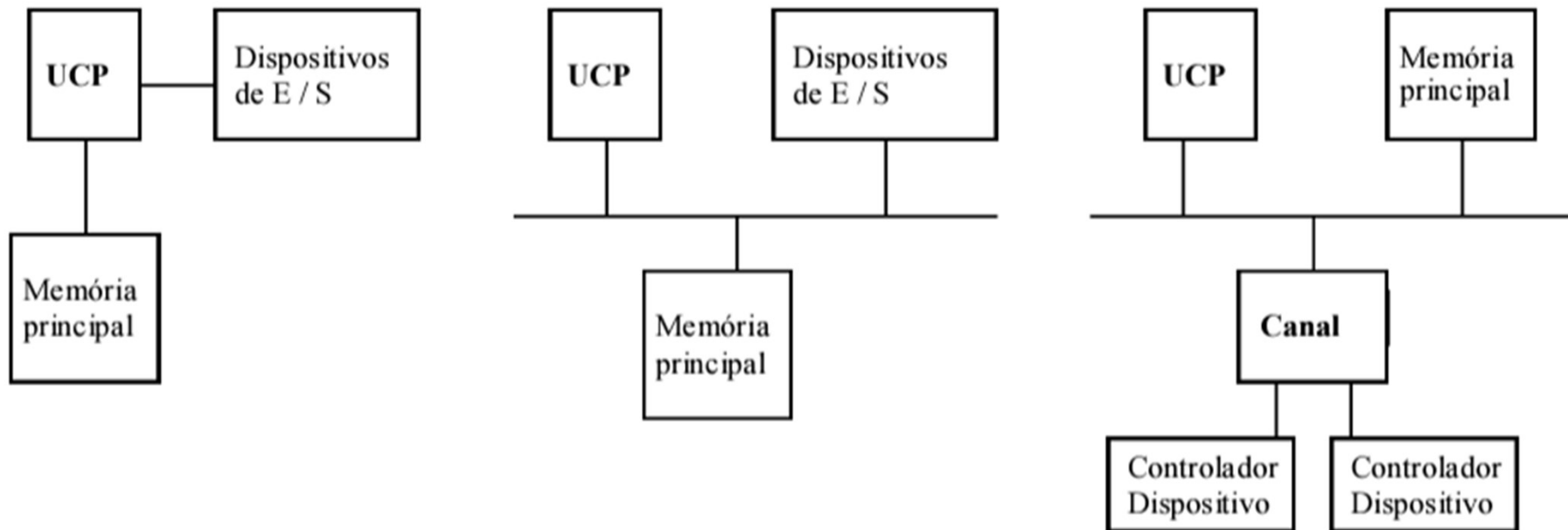
## ○ Barramento:

- Tipos: PCI, USB, SATA.
- Estrutura: Linhas de endereço, linhas de dados, linhas de controle.

## Diagrama da Interconexão:

- Processador conectado à Memória Principal.
- Controladores de dispositivos conectados ao barramento.
- Comunicação entre os componentes via sinais elétricos.

# CONTROLADORES DE DISPOSITIVOS



# ACESSO DIRETO À MEMÓRIA (DMA)

Para evitar que o processador tenha que intermediar cada transferência de valores de/para um controlador de dispositivo, usando o barramento, é possível usar um dispositivo de controle que permite o acesso direto à memória pelos controladores, sem intermediação do processador.

Esse dispositivo é o controlador de acesso direto à memória (DMA).

# DMA

DMA: Transferência de dados entre o controlador e a memória sem a intervenção do processador.

Benefícios:

- Maior eficiência na transferência de dados.
- Libera o processador para outras tarefas.

# DMA

## Funcionamento:

- O SO configura o DMA com endereço da memória, tamanho do bloco e operação.
- O DMA transfere dados diretamente para a memória.

# ACESSO DIRETO SEM DMA

SO envia comando de transferência para controlador do dispositivo

O controlador lê o bloco (pensando em dispositivo de bloco) bit a bit e o armazena num buffer interno

A verificação de erro é efetuada

Se não houve erros, o controlador gera uma interrupção

A CPU (executando código do SO) lê os dados, bytes ou palavras, via barramento, e os armazena na memória

Problema: gasto de tempos da CPU para ler informações byte a byte pelo barramento

## ACESSO COM DMA

SO envia comando de transferência para o controlador do dispositivo

SO passa endereço da memória e tamanho como parâmetros extras ao controlador de DMA



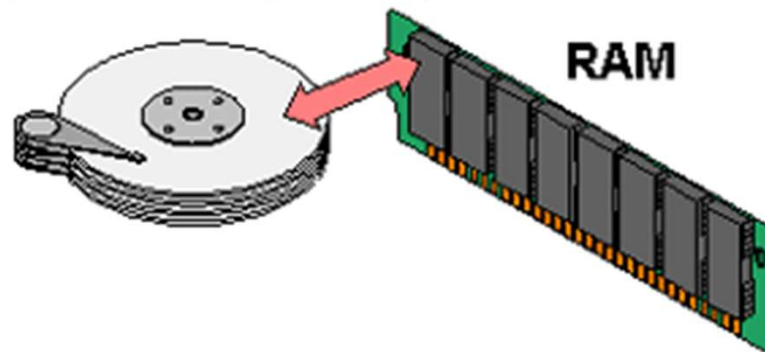
## ACESSO COM DMA

Dados já verificados, armazenados no buffer pelo controlador, são transferidos para a memória diretamente, ou por intermédio do controlador de DMA

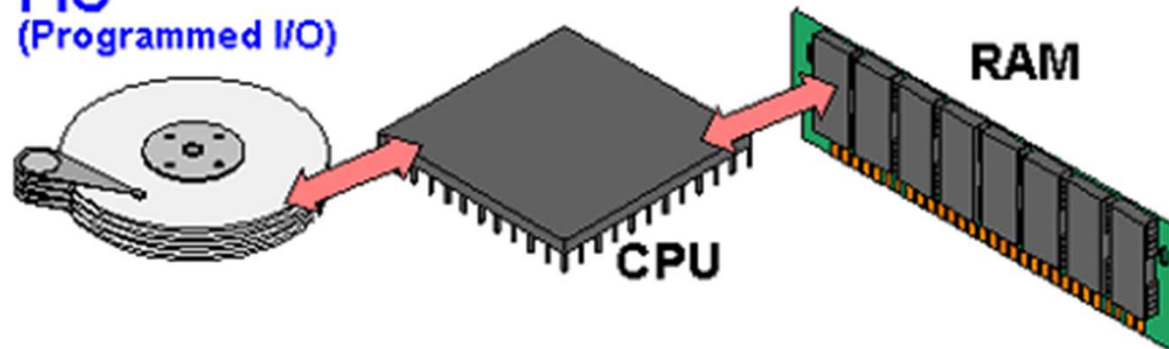
Interrupção é gerada à UCP quando transferência foi concluída

# ACESSO COM E SEM DMA

**DMA**  
(Direct Memory Access)



**PIO**  
(Programmed I/O)



# SOFTWARE DE E/S

Camadas de Software:

- Nível mais baixo: Tratadores de interrupção.
- Nível intermediário: Controladores de dispositivos (device drivers).
- Nível superior: Software de E/S independente de dispositivo.

# TRATADORES DE INTERRUPÇÃO

Função:

- Responder às interrupções geradas pelos dispositivos.
- Interromper o processador para tratar eventos de E/S.

# TRATADORES DE INTERRUPÇÃO

Exemplo:

- Interrupção de fim de transferência de dados.
- Interrupção de erro.

Papel do SO:

- Controlar e gerenciar as interrupções.

# CONTROLOADORES DE DISPOSITIVOS (DEVICE DRIVERS)

Função:

- Código específico para cada tipo de dispositivo.
- Traduzir as operações de E/S do SO em comandos para o dispositivo.
- Gerenciar o acesso ao hardware.

Exemplo:

- Driver de disco rígido.
- Driver de rede.

# SOFTWARE DE E/S INDEPENDENTE DE DISPOSITIVO

Função:

- Fornecer uma interface uniforme para os programas.
- Esconder as diferenças entre os dispositivos.
- Gerenciar recursos de E/S, como buffers e alocação de espaço.

Exemplo:

- Funções de leitura e gravação de arquivos.

# OBJETIVOS DO SOFTWARE DE E/S

Independência de Dispositivo:

- Os programas podem usar o mesmo código para diversos dispositivos.

Nomeação Uniforme:

- Arquivos e dispositivos podem ser endereçados por nomes.

Manipulação de Erros:

- Erros de E/S devem ser tratados e notificados.



# OBJETIVOS DO SOFTWARE DE E/S

Transferência Síncrona (Bloqueante):

- O programa espera a conclusão da operação de E/S antes de continuar.

Transferência Assíncrona (Orientada a Interrupções):

- O programa pode continuar executando enquanto a operação de E/S ocorre.

Compartilhamento de Dispositivos:

- O SO controla o acesso compartilhado a dispositivos.

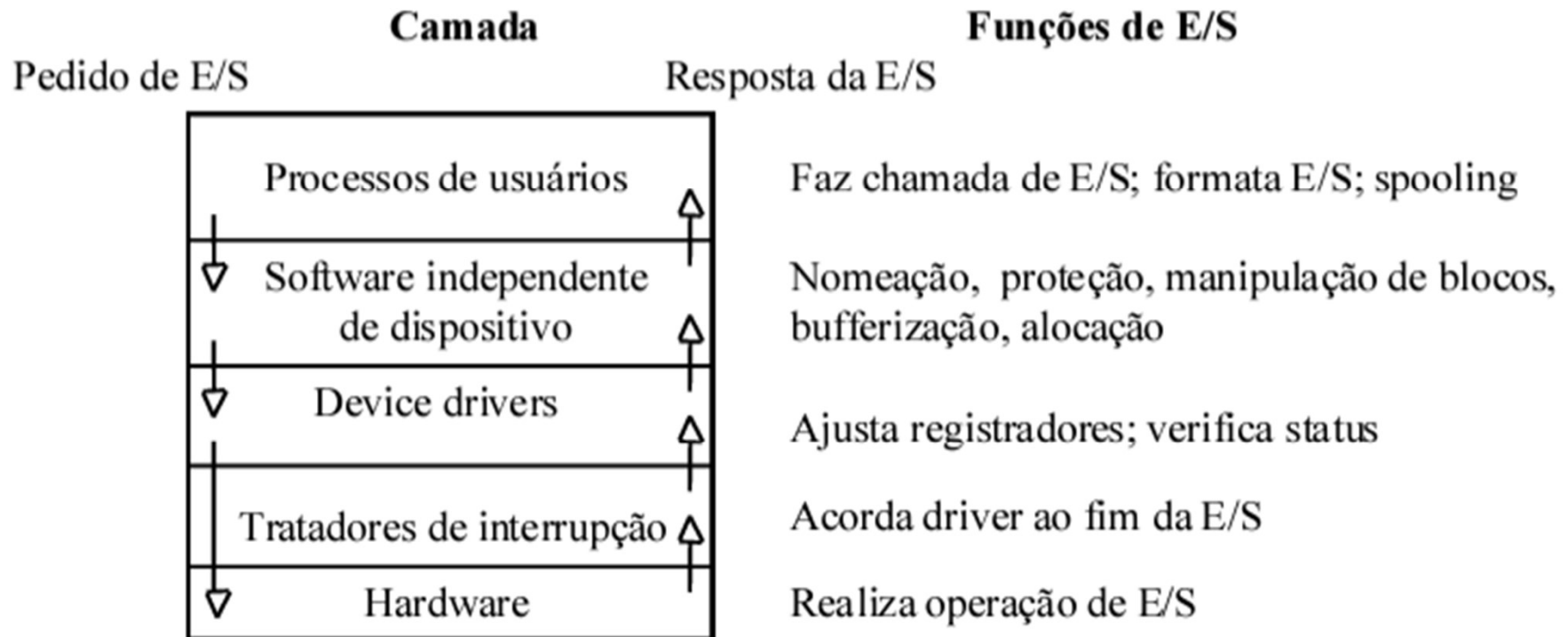
# INTERAÇÃO ENTRE AS CAMADAS DE SOFTWARE

Como as camadas de software interagem entre si.

Exemplo:

- Um programa faz uma chamada de sistema para ler um arquivo.
- O sistema operacional chama a função do driver de disco rígido.
- O driver envia comandos ao controlador do disco rígido.

# INTERAÇÃO ENTRE AS CAMADAS DE SOFTWARE



# ENTRADA E SAÍDA PADRÃO

STDIN: Arquivo padrão de entrada.

STDOUT: Arquivo padrão de saída.

STDERR: Arquivo padrão de saída de erros.

Exemplo:

- Um programa lê dados do teclado (STDIN).
- O programa imprime resultados na tela (STDOUT).
- O programa imprime mensagens de erro na tela (STDERR).

# IMPACTO DE E/S NO DESEMPENHO DO SISTEMA

Tempo de resposta:

- Tempo que o sistema leva para responder a uma solicitação do usuário.

Taxa de transferência:

- Quantidade de dados transferidos por unidade de tempo.

Throughput:

- Número de solicitações processadas por unidade de tempo.