

---

## Aula 07: Arquitetura de Computadores – Entrada/Saída

---

Prof. Hugo Puertas de Araújo  
[hugo.puertas@ufabc.edu.br](mailto:hugo.puertas@ufabc.edu.br)  
Sala: 509-2 (5º andar / Torre 2)



# Arquitetura de Computadores

# ■ Objetivos de aprendizagem

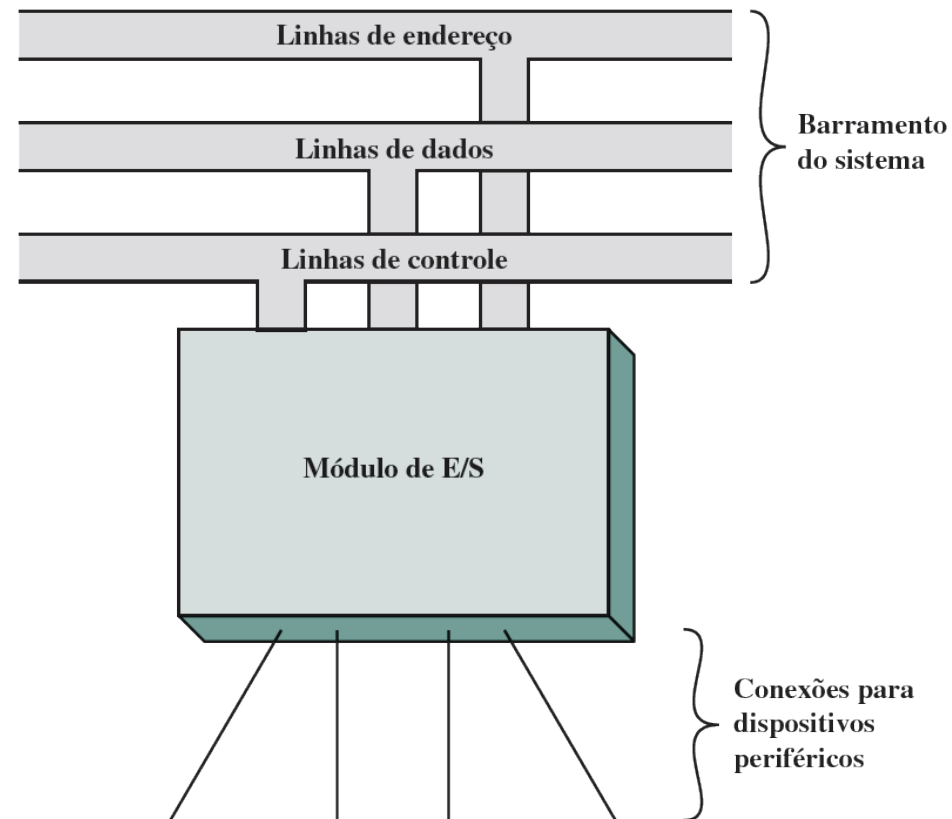
- Explicar o uso dos módulos de E/S como parte da organização de computador.
- Entender a diferença entre E/S programada e E/S controlada por interrupção e discutir suas vantagens relativas.
- Apresentar visão geral da operação do acesso direto à memória.
- Apresentar uma visão geral do acesso direto à cache.
- Explicar a função e o uso dos canais de E/S.

# ■ Dispositivos externos

- As operações de E/S são realizadas por meio de uma grande variedade de dispositivos externos.
- Um dispositivo externo conecta-se ao computador por uma conexão com um módulo de E/S.
- Podemos classificar os dispositivos externos em geral em três categorias:
  - i. Inteligíveis ao ser humano
  - ii. Inteligíveis à máquina
  - iii. Comunicação

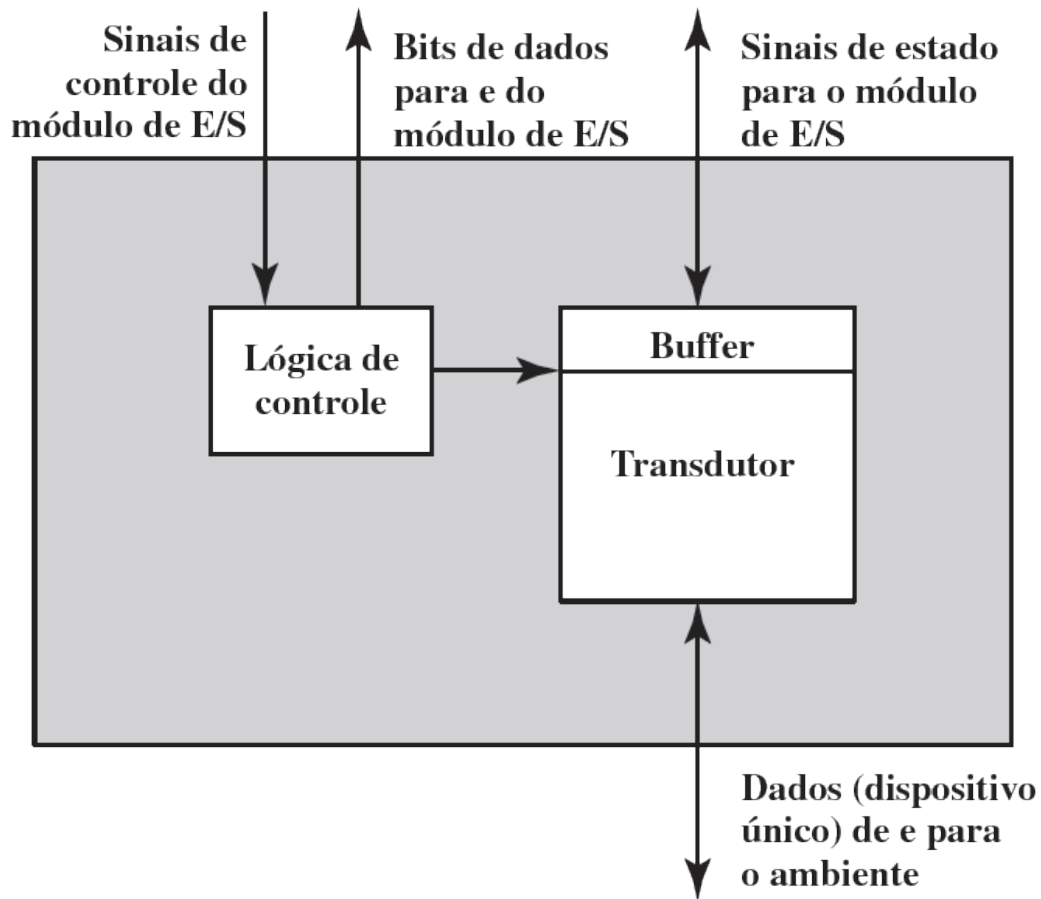
# Dispositivos externos

- Modelo genérico de um módulo de E/S:



# Dispositivos externos

- Diagrama em blocos de um dispositivo externo:



# ■ Teclado/monitor

- O meio mais comum de interação entre computador/usuário é o conjunto teclado/monitor.
- O usuário fornece entrada pelo teclado.
- A unidade de troca básica é o caractere.
- Associado a cada caractere existe um código, em geral com tamanho de 7 ou 8 bits.
- Quando o usuário pressiona uma tecla, isso gera um sinal eletrônico que é interpretado pelo transdutor no teclado e traduzido para o padrão de bits do código International Reference Alphabet - IRA correspondente.

# ■ Teclado/monitor

- Esse padrão de bits é, então, transmitido ao módulo de E/S no computador, onde o texto pode ser armazenado no mesmo código IRA.
- Na saída, os caracteres do código IRA são transmitidos para um dispositivo externo do módulo de E/S.
- O transdutor no dispositivo interpreta esse código e envia os sinais eletrônicos exigidos ao dispositivo de saída, ou para exibir o caractere indicado, ou para realizar a função de controle solicitada.



# ■ Drive de disco

- Uma unidade de disco contém a eletrônica para trocar sinais de dados, controle e estado com um módulo de E/S mais a eletrônica para controlar os mecanismos de leitura/gravação de disco.
- Em um disco de cabeça fixa, o transdutor é capaz de converter os padrões magnéticos na superfície do disco móvel em bits no buffer do dispositivo.
- Um disco com cabeça móvel também deve ser capaz de fazer o braço do disco se mover radialmente para dentro e fora pela superfície do disco.

# ■ Drive de disco

**Exercício**

- Uma unidade de disco contém a eletrônica para trocar sinais de dados, controle e estado com um módulo de E/S mais a eletrônica para controlar os mecanismos de leitura/gravação de disco.
- Em um disco de cabeça fixa, o transdutor é capaz de converter os padrões magnéticos na superfície do disco móvel em bits no buffer do dispositivo.
- Um disco com cabeça móvel também deve ser capaz de fazer o braço do disco se mover radialmente para dentro e fora pela superfície do disco.

**E.: O drive de disco é E/S ou faz parte da hierarquia de memória?**

# ■ Módulos de E/S

- As principais funções ou requisitos para um módulo de E/S encontram-se nas seguintes categorias:
  - ❖ Controle e temporização
  - ❖ Comunicação com o processador
  - ❖ Comunicação com o dispositivo
  - ❖ Buffering de dados
  - ❖ Detecção de erro

# ■ Módulos de E/S

- A função de E/S inclui um requisito de controle e temporização para coordenar o fluxo de tráfego entre os recursos internos e dispositivos externos.
- A comunicação do processador envolve o seguinte:
  - ❖ Decodificação de comando
  - ❖ Dados
  - ❖ Informação de estado
  - ❖ Reconhecimento de endereço

# ■ Módulos de E/S

- O módulo de E/S também deve ser capaz de realizar **comunicação com o dispositivo**.
- Ela envolve comandos, informação de estado e dados.
- Uma tarefa essencial de um módulo de E/S é o **buffering de dados**.
- Os dados são mantidos em um buffer no módulo de E/S e depois enviados ao dispositivo periférico em sua taxa de dados.
- Por fim, um módulo de E/S é responsável pela **detecção de erro** e, subsequentemente, por relatar erros ao processador.

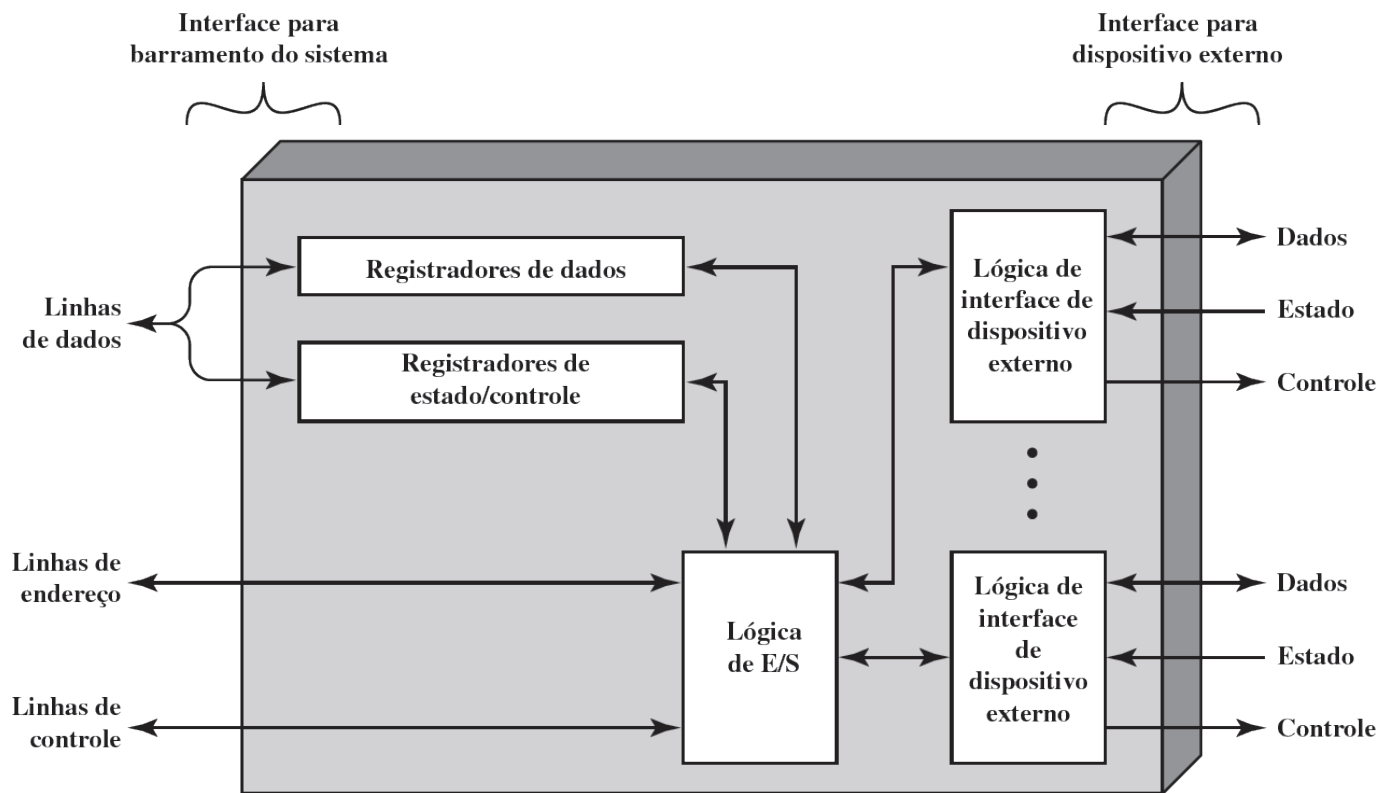
# ■ Módulos de E/S

**Exercício**

- O módulo de E/S também deve ser capaz de realizar **comunicação com o dispositivo**.
- Ela envolve comandos, informação de estado e dados.
- Uma tarefa essencial de um módulo de E/S é o **buffering de dados**.
- Os dados são mantidos em um **buffer** no módulo de E/S e depois enviados ao dispositivo periférico em sua taxa de dados.
- Por fim, um módulo de E/S é responsável pela **detecção de erro** e, subsequentemente, por relatar erros ao processador.

# ■ Estrutura do módulo de E/S

## ■ Diagrama do bloco de um módulo de E/S:



# E/S programada

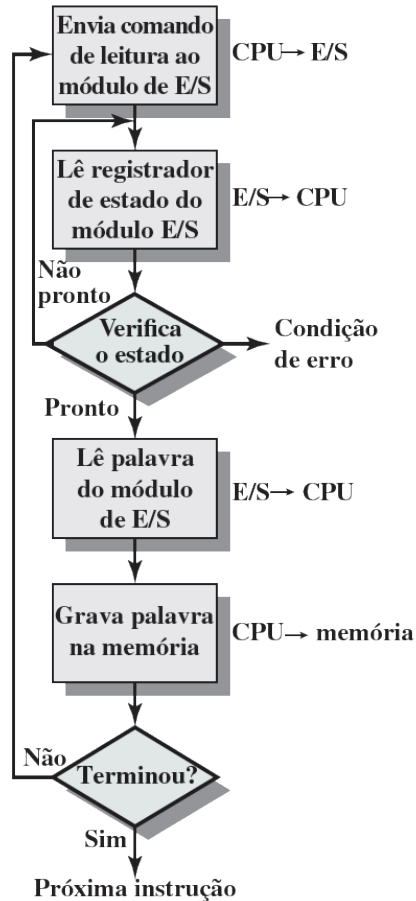
- Três técnicas são possíveis para operações de E/S:

	Sem interrupções	Uso de interrupções
Transferência de E/S para memória via processador	E/S programada	E/S controlada por interrupção
Transferência direta de E/S para memória		Acesso direto à memória (DMA)

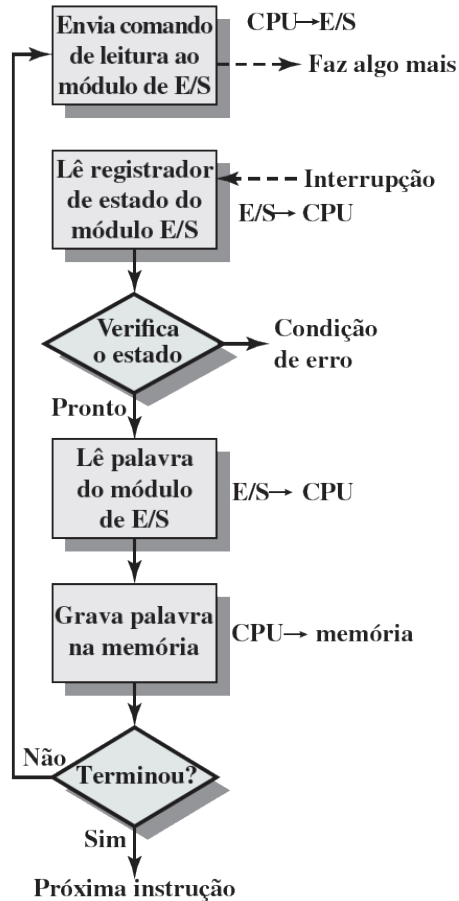
- A alternativa a estes modos é conhecida como **acesso direto à memória (DMA)**.
- Nesse modo, o módulo de E/S e a memória principal trocam dados diretamente, sem envolvimento do processador.



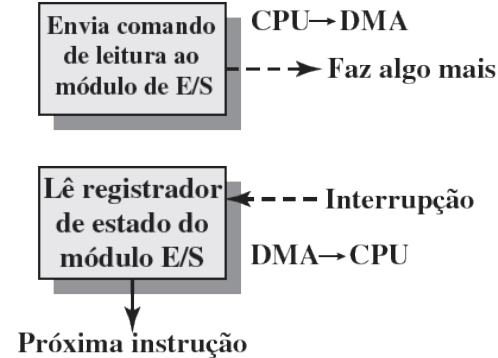
# Visão geral da E/S programada



E/S programada



E/S dirigida por interrupção



Acesso direto à memória

# ■ Comando de E/S

- Existem quatro tipos de comandos de E/S:
  - i. **Controle**: usado para ativar um periférico e dizer-lhe o que fazer.
  - ii. **Teste**: usado para testar diversas condições de estado associadas a um módulo de E/S e seus periféricos.
  - iii. **Leitura**: faz com que o módulo de E/S obtenha um item de dados do periférico e o coloque em um buffer interno.
  - iv. **Escrita**: faz com que o módulo de E/S apanhe um item de dado do barramento de dados e depois transmita-o ao periférico.

# Comando de E/S

**Exercício**

- Existem quatro tipos de comandos de E/S:
  - i. **Controle**: usado para ativar um periférico e dizer-lhe o que fazer.
  - ii. **Teste**: usado para testar diversas condições de estado associadas a um módulo de E/S e seus periféricos.
  - iii. **Leitura**: faz com que o módulo de E/S obtenha um item de dados do periférico e o coloque em um buffer interno.
  - iv. **Escrita**: faz com que o módulo de E/S apanhe um item de dado do barramento de dados e depois transmita-o ao periférico.

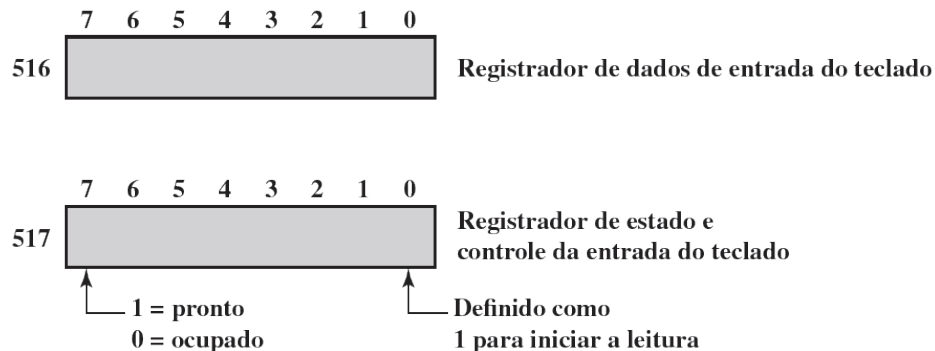
**E.: O que fazem os comandos de teste?**

# Instruções de E/S

- Com a E/S mapeada na memória, existe um único espaço de endereço para locais de memória e dispositivos de E/S.
- O processador trata os registradores de estado e dados dos módulos de E/S como locais de memória e usa as mesmas instruções de máquina para acessar a memória e os dispositivos de E/S.
- Quando o espaço de endereço para E/S é independente do espaço da memória, é chamado de E/S independente. Para esse tipo, apenas comandos especiais de E/S são permitidos.
- A figura a seguir compara essas duas técnicas.

# Instruções de E/S

## E/S mapeada na memória e isolada:



E/S mapeada na memória

ENDEREÇO	INSTRUÇÃO	OPERANDO	COMENTÁRIO
200	Carrega AC	"1"	Carrega acumulador
	Armazena AC	517	Inicia a leitura do teclado
202	Carrega AC	517	Colhe byte de estado
	Desvia, se sinal = 0	202	Loop até estar pronto
	Carrega AC	516	Carrega byte de dados

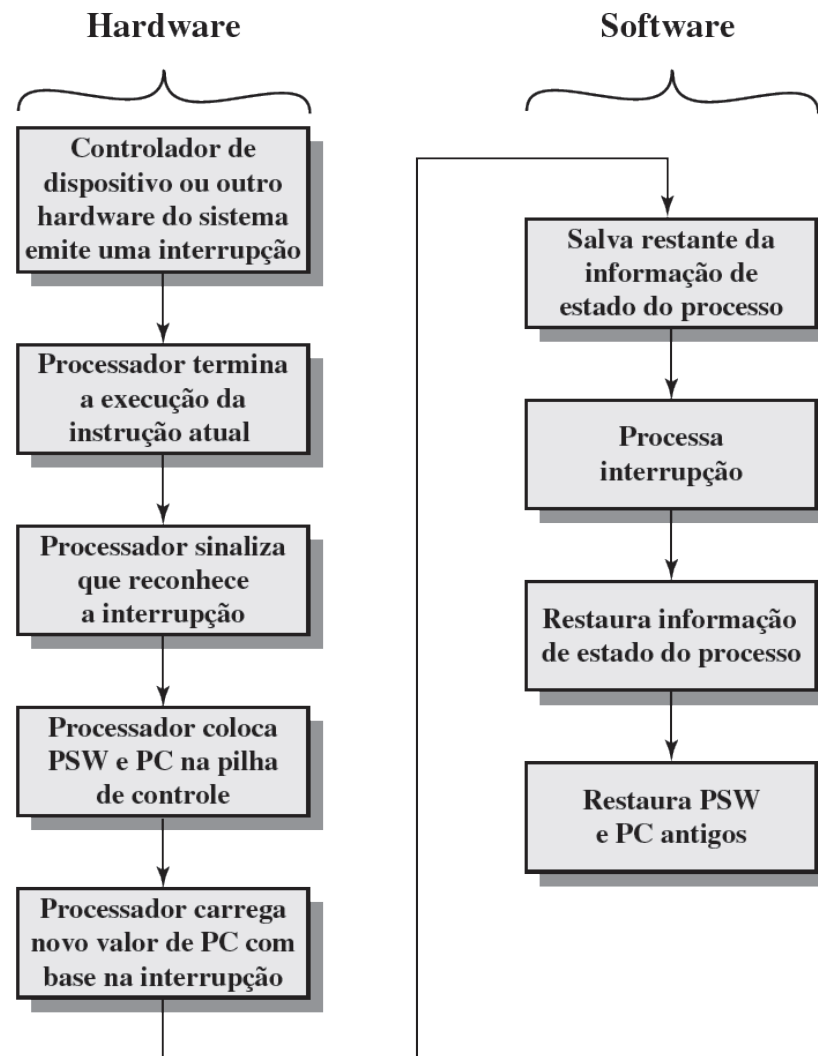
E/S independente	ENDEREÇO	INSTRUÇÃO	OPERANDO	COMENTÁRIO
	200	Carrega E/S	5	Inicia a leitura do teclado
	201	Testa E/S	5	Checa término
		Desvia se não pronto	201	Loop até estar pronto
		Entrada	5	Carrega byte de dados

# E/S controlada por interrupção

- O problema com a E/S programada é que o processador tem de esperar muito tempo para que o módulo de E/S de interesse esteja pronto para recepção ou transmissão de dados.
- Uma alternativa é que o processador envie um comando de E/S para um módulo e depois continue realizando algum outro trabalho útil.
- O módulo interromperá o processador para solicitar atendimento quando estiver pronto para trocar dados com o processador.
- O processador, então, executará a transferência de dados, como antes, e depois retomará seu processamento anterior.

# E/S controlada por interrupção

- Processamento de interrupção simples:



# E/S controlada por interrupção

- Dois aspectos de projeto surgem na implementação da E/S por interrupção:
- Considerando que quase sempre haverá vários módulos de E/S, como o processador determina qual dispositivo emitiu a interrupção?
- Se houver várias interrupções, como o processador decide qual deverá processar?
- A técnica mais simples para o problema é oferecer múltiplas linhas de interrupção entre o processador e os módulos de E/S.



# ■ E/S controlada por interrupção

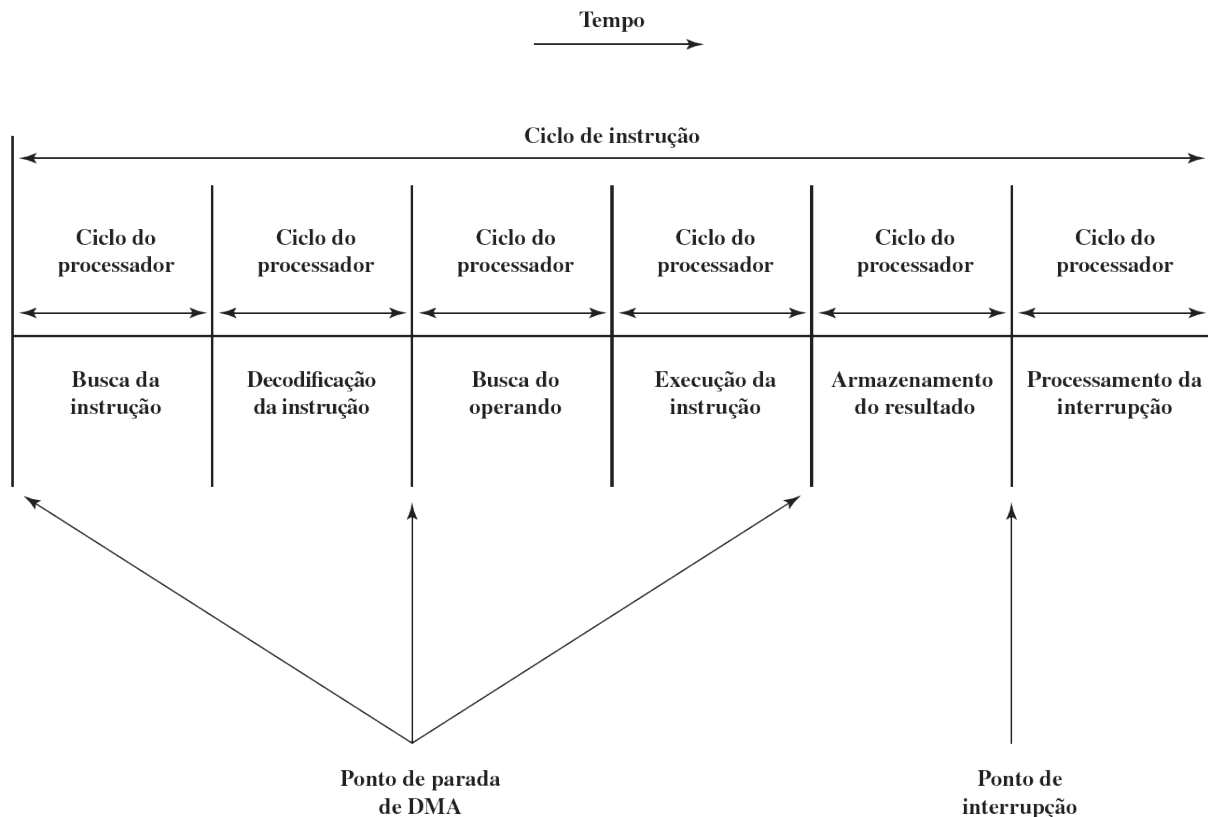
- Todavia, é impraticável dedicar mais do que algumas poucas linhas de barramento ou pinos de processador às linhas de interrupção.
- Uma alternativa é a **verificação por software**.
- A desvantagem da verificação por software é que ele é demorado.
- Uma técnica mais eficiente é usar uma configuração **daisy chain**, que oferece uma verificação por hardware.

# Acesso direto à memória

- Desvantagens da E/S programada e controlada por interrupção:
  - i. A taxa de transferência de E/S é limitada pela velocidade com a qual o processador pode testar e atender a um dispositivo.
  - ii. O processador fica ocupado no gerenciamento de uma transferência de E/S; diversas instruções precisam ser executadas para cada transferência de E/S.
- O DMA envolve um módulo adicional no barramento do sistema.
- O módulo de DMA é capaz de imitar o processador e, na realidade, assumir o controle do sistema do processador.

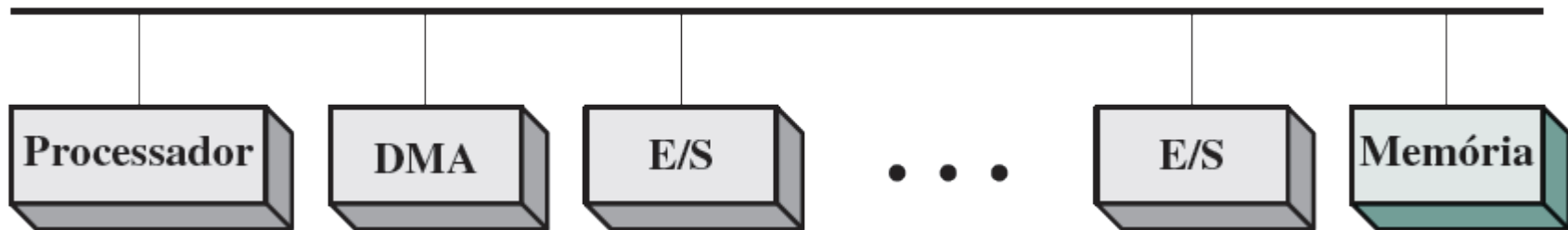
# Acesso direto à memória

- DMA e pontos de interrupção durante um ciclo de instrução:



# Acesso direto à memória

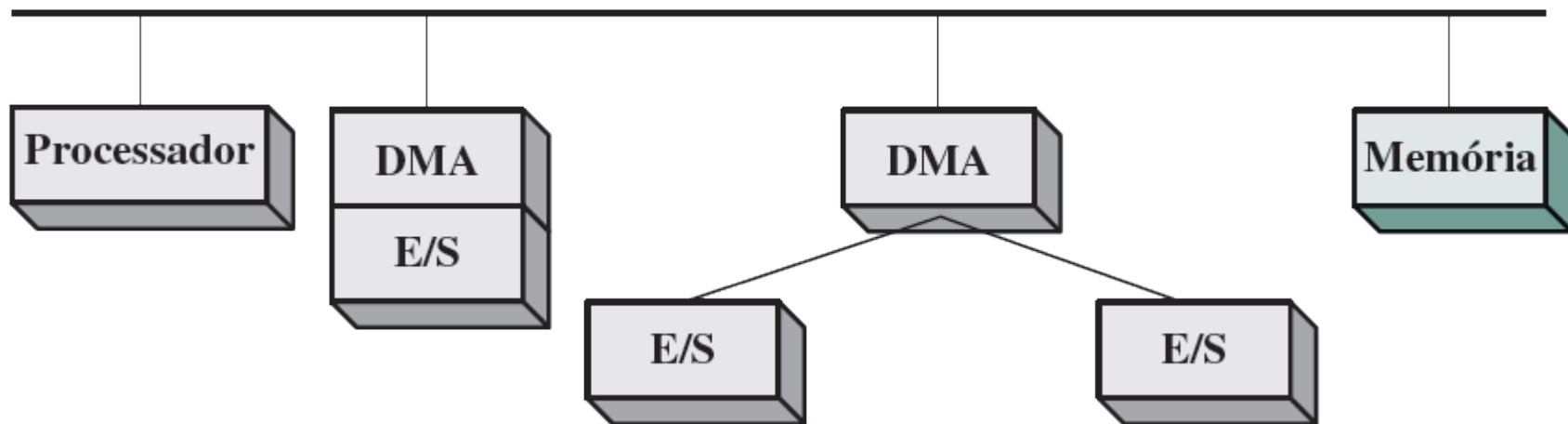
## ■ Configurações de DMA alternativas:



Único barramento, DMA separado

# Acesso direto à memória

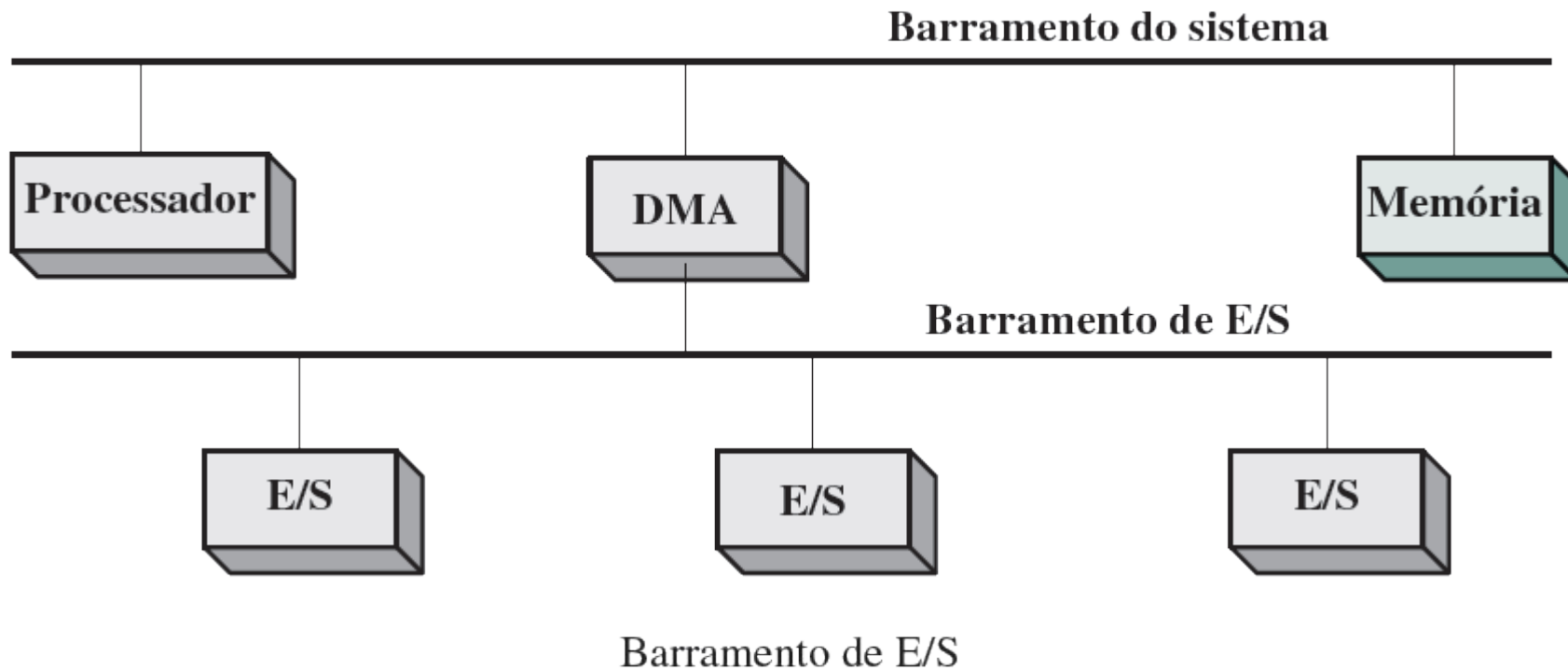
## ■ Configurações de DMA alternativas:



Único barramento, DMA-E/S integrados

# Acesso direto à memória

## ■ Configurações de DMA alternativas:



# Acesso direto à cache

- Os sistemas multicore modernos incluem tanto a cache dedicada a cada core como um nível adicional da cache compartilhada, seja L2 ou L3.
- Para esclarecer a interação do DMA e da cache, será útil primeiro descrever uma arquitetura de sistema específica.

## Processador multicore Xeon

- Trata-se de uma família de processadores de tecnologia de ponta e de alto desempenho usada nos servidores, estações de trabalho de alto desempenho e supercomputadores.

# Acesso direto à cache

- O E5-2600/4600 pode ser configurado com até oito cores em um chip único.
- Existe uma cache L3 de até 20 MB.
- A cache L3 é dividida em faixas, cada uma associada com cada core, embora cada um possa se referir a toda a cache.
- Cada faixa tem seu próprio pipeline de cache, de tal maneira que os pedidos podem ser enviados em paralelo a essas faixas.
- A interconexão em anel bidirecional de alta velocidade liga os cores, cache de último nível, PCIe e IMC.



# Acesso direto à cache

- Para saída, quando o controlador de E/S emite uma requisição de leitura, a MCH primeiro checa para ver se os dados estão na cache L3.
- A MCH direciona os dados a partir da cache L3 ao controlador de E/S; não são necessários acessos à memória principal.
- Desse modo, a operação de E/S procede de modo eficiente porque ela não requer acesso à memória principal.
- Todavia, se uma aplicação de fato precisar desses dados no futuro, eles devem ser lidos de volta na cache L3 a partir da memória principal.

# Acesso direto à cache

- Para esclarecer a questão do desempenho e explicar o benefício de DCA como uma maneira de aprimorar o desempenho, vamos analisar o processo de tráfego de protocolo em mais detalhes para o tráfego de entrada. Em termos gerais, ocorrem as seguintes etapas:
  - i. Chegada de pacote
  - ii. DMA
  - iii. NIC interrompe o host
  - iv. Cabeçalhos e descritores de recuperação
  - v. Ocorrências de falha de cache
  - vi. O cabeçalho é processado
  - vii. Transferência do bloco

# Acesso direto à cache

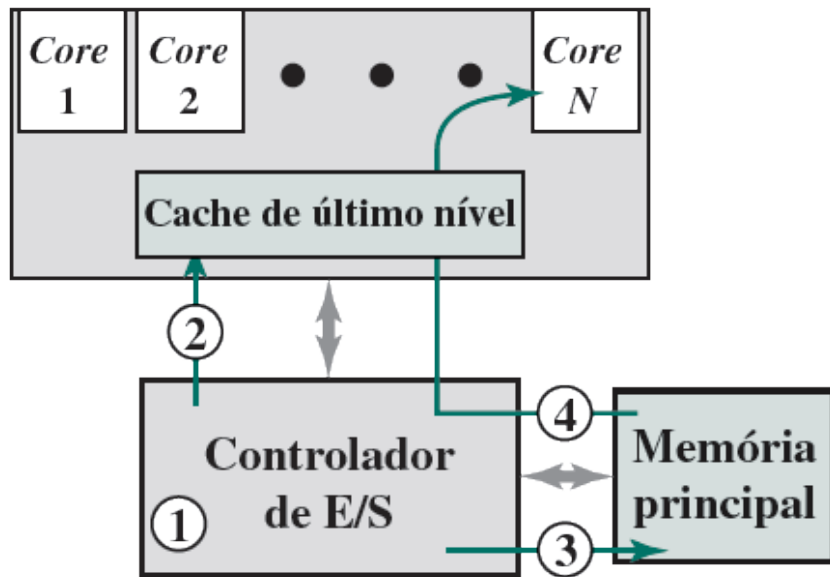
- Uma sequência similar de etapas ocorre para o tráfego de saída de pacote, mas há algumas diferenças que afetam o modo como a cache é gerenciada.
- Para o tráfego de saída, ocorrem as seguintes etapas:
  - i. Solicitação de transferência de pacote
  - ii. Criação do pacote
  - iii. Chamada de uma operação de saída
  - iv. Transferência de DMA
  - v. Sinais de Network Interface Controller – NIC de finalização
  - vi. O driver libera buffer

# E/S de Dados Diretos

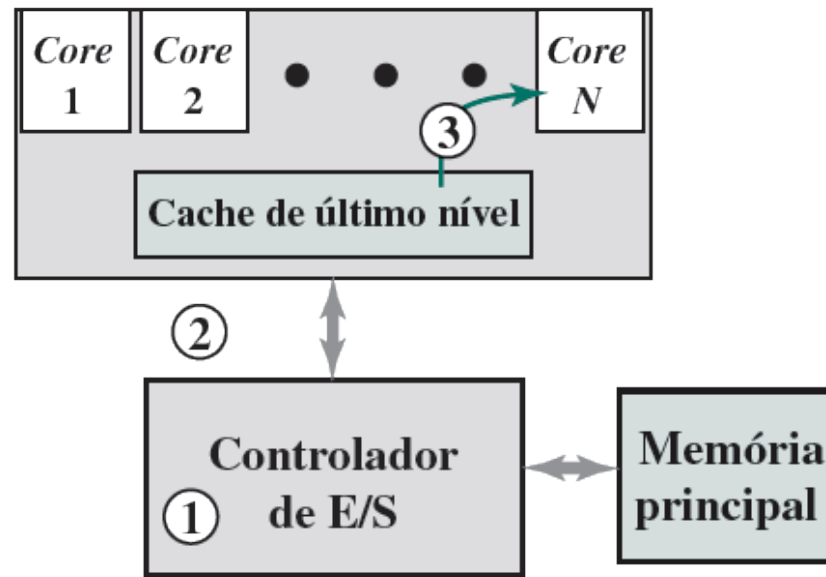
- A E/S de Dados Diretos da Intel (DDIO) é implementada em toda a família Xeon E5 de processadores.
- Sua operação é mais bem explicada com a comparação lado a lado de transferências com e sem DDIO.
- A estratégia de DDIO é efetivamente para a aplicação do protocolo de rede, porque os dados de entrada não precisam ser retidos para uso futuro.
- A aplicação do protocolo vai gravar os dados em um buffer de aplicação, e não há necessidade de armazenar temporariamente em um buffer de sistema.

# E/S de Dados Diretos

## ■ Comparação de DMA e DDIO:



Transferência normal de DMA à memória

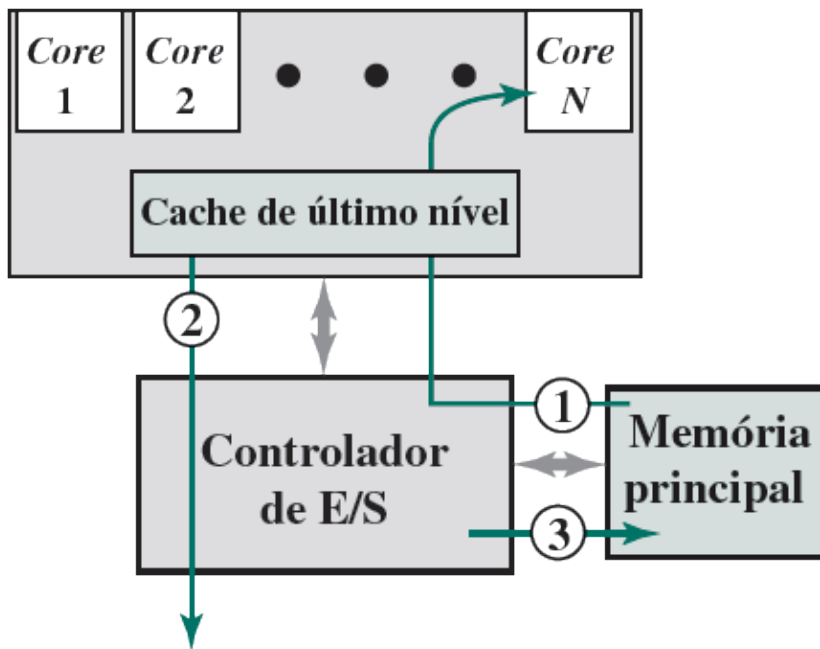


Transferência de DDIO à cache

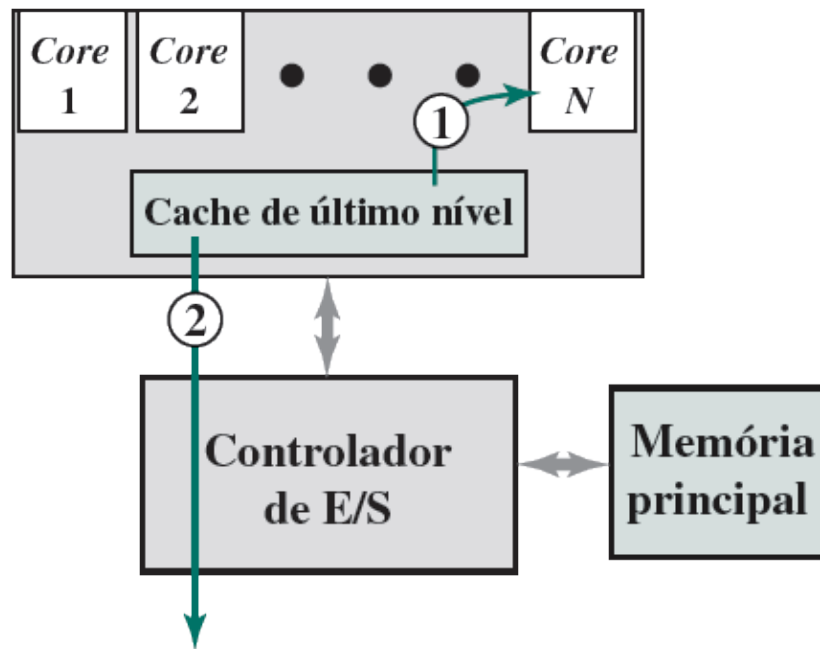
# E/S de Dados Diretos

## ■ Comparação de DMA e DDIO:

DDIO não envolve atualização da memória principal



Transferência normal de DMA à E/S



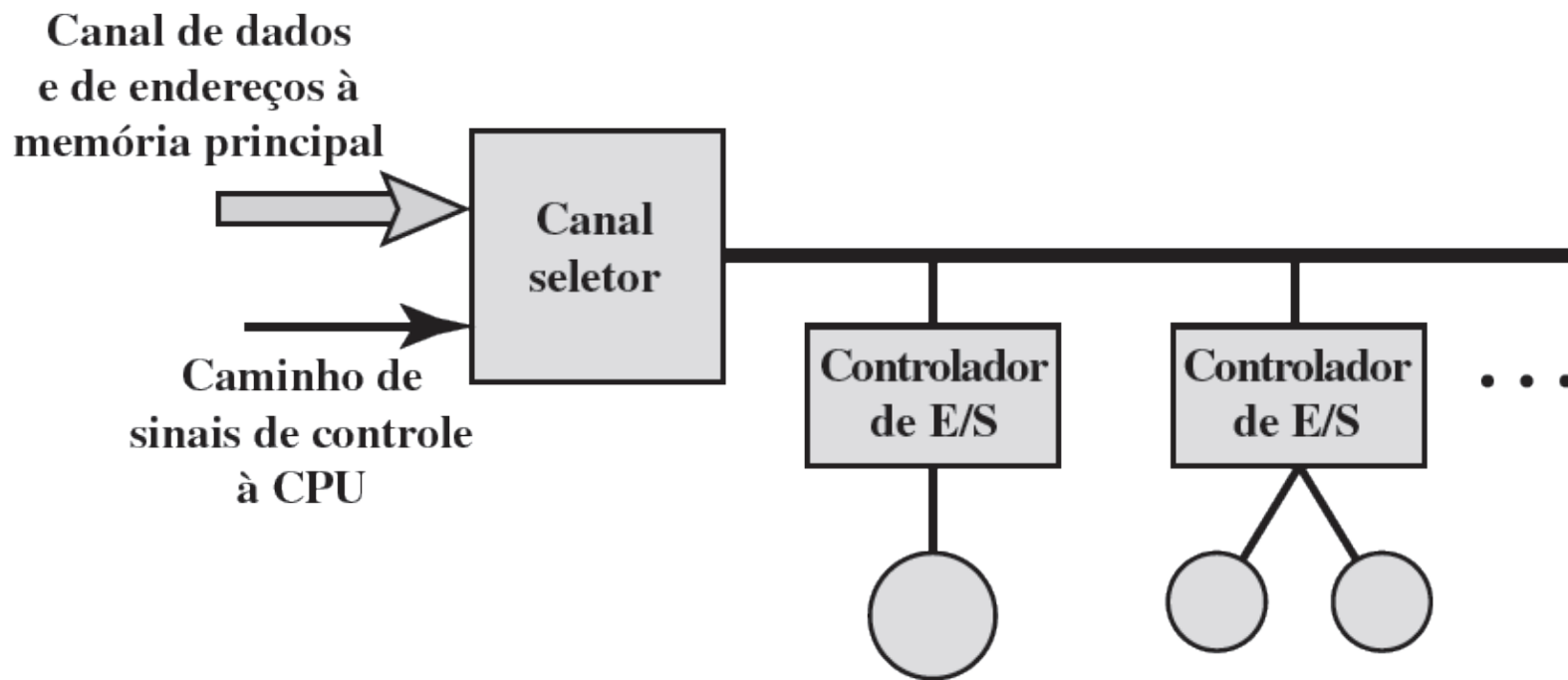
Transferência de DDIO à E/S

# ■ Processadores e canais de E/S

- Enquanto se prossegue no caminho de evolução, cada vez mais a função de E/S é realizada sem envolvimento da CPU.
- A CPU fica cada vez mais livre das tarefas relacionadas a E/S, melhorando o desempenho.
- Ocorre uma grande mudança com a introdução do conceito de um módulo de E/S capaz de executar um programa.
- O módulo de E/S em geral é conhecido como um canal de E/S.

# ■ Características dos canais de E/S

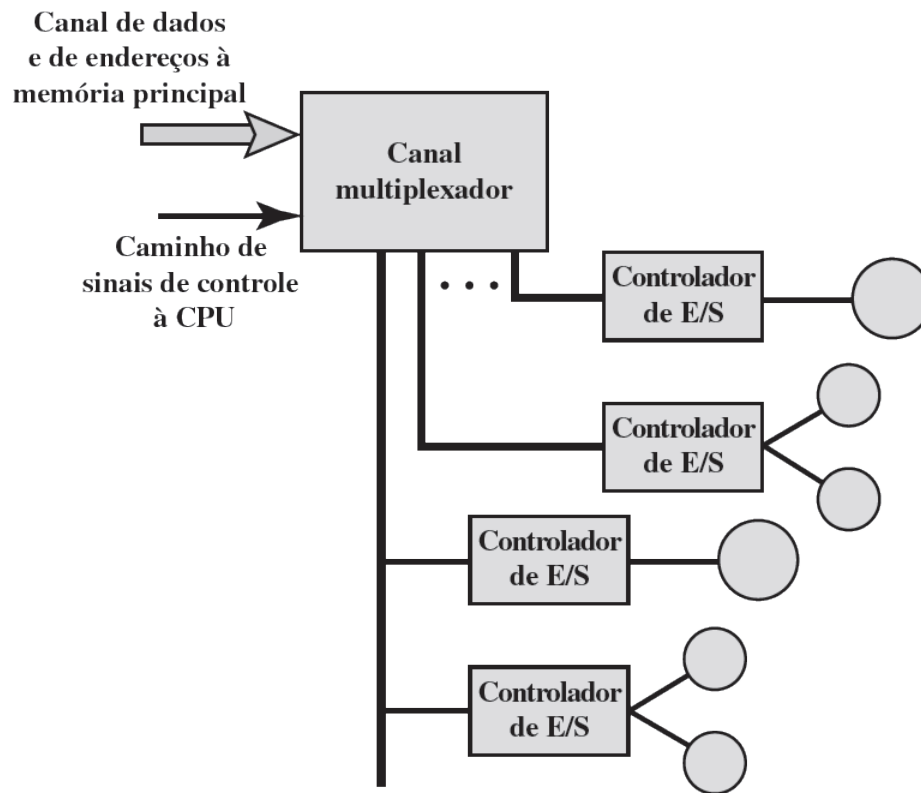
- Um canal seletor controla diversos dispositivos de alta velocidade e, a qualquer momento, é dedicado à transferência de dados com um desses dispositivos:





# ■ Características dos canais de E/S

- Um canal multiplexador pode tratar da E/S com vários dispositivos ao mesmo tempo:



# ■ Padrões de interconexão externa

- O **USB** é bastante usado para conexões periféricas.
- É a interface padrão para dispositivos de velocidade mais lenta, como teclado e dispositivos apontadores, mas também é comumente usada para E/S de alta velocidade, incluindo impressoras, drives de disco e adaptadores de rede.
- Ele vem de várias gerações.
- O sistema USB é controlado por um controlador host central, que é conectado aos dispositivos para criar uma rede local com uma topologia hierárquica em árvore.

# ■ Padrões de interconexão externa

- A interface **SCSI** é um padrão comum para conectar dispositivos periféricos (discos, modems, impressoras etc.) a computadores pequenos e médios.
- A organização física da SCSI é um barramento compartilhado, que pode suportar até 16 ou 32 dispositivos, dependendo da geração do padrão.
- A velocidade varia de 5 Mbps na especificação original de SCSI-1 a 160 Mbps na SCSI-3 U3.

# ■ Padrões de interconexão externa

- O **FireWire** foi desenvolvido como uma alternativa para a interface SCSI.
- O FireWire usa uma configuração daisy-chain, com até 63 dispositivos conectados em uma única porta.
- Até 1.022 barramentos FireWire podem ser interconectados usando pontes.
- O FireWire permite o que é conhecido como conexão a quente (hot plugging), que significa que é possível conectar e desconectar periféricos sem ter que desligar o sistema de computação ou reconfigurar o sistema.

# ■ Padrões de interconexão externa

- A mais recente, e uma das mais rápidas, tecnologia de conexão de periféricos a se tornar disponível para uso de propósito geral é o **Thunderbolt**, desenvolvido pela Intel em colaboração com a Apple.
- A tecnologia combina dados, vídeos, áudios e energia em uma única conexão de alta velocidade para periféricos como drives de disco rígidos, conjuntos de RAID, caixas de captura de vídeo e interfaces de rede.
- Isso proporciona uma taxa de transferência de até 10 Gbps em cada direção e até 10 watts de potência aos periféricos conectados.

# ■ Padrões de interconexão externa

- **InfiniBand** é uma especificação de E/S, voltada para o mercado de servidores de ponta.
- O **PCI Express** é um sistema de barramento de alta velocidade que conecta periféricos de uma grande variedade de tipos e velocidades.
- **Serial ATA** é uma interface para sistemas de armazenamento de disco.
- **Ethernet** é uma tecnologia de rede predominantemente com fios, usada em casas, escritórios, centros de dados, empresas e redes de área ampla.

# ■ Padrões de interconexão externa

- O **Wi-Fi** é uma tecnologia de acesso à internet predominantemente sem fio, usado em casas, escritórios e espaços públicos.
- O Wi-Fi em casa agora conecta computadores, tablets, smartphones e hosts de dispositivos eletrônicos.
- O Wi-Fi nas empresas tem se tornado um meio essencial para aumentar a produtividade dos colaboradores.
- Os hotspots de Wi-Fi público expandiram-se de modo significativo para proporcionar acesso livre à internet em locais públicos.