
Aula 03: Arquitetura de Computadores – Visão de alto nível da função e interconexão do computador

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



Arquitetura de Computadores

■ Objetivos de aprendizagem

- Entender os elementos básicos de um ciclo de instrução e o papel das interrupções.
- Descrever o conceito de interconexão dentro de um sistema computacional.
- Avaliar as vantagens relativas da interconexão ponto a ponto em comparação com a interconexão de barramento.
- Apresentar uma visão geral da QPI.
- Apresentar uma visão geral da PCIe.

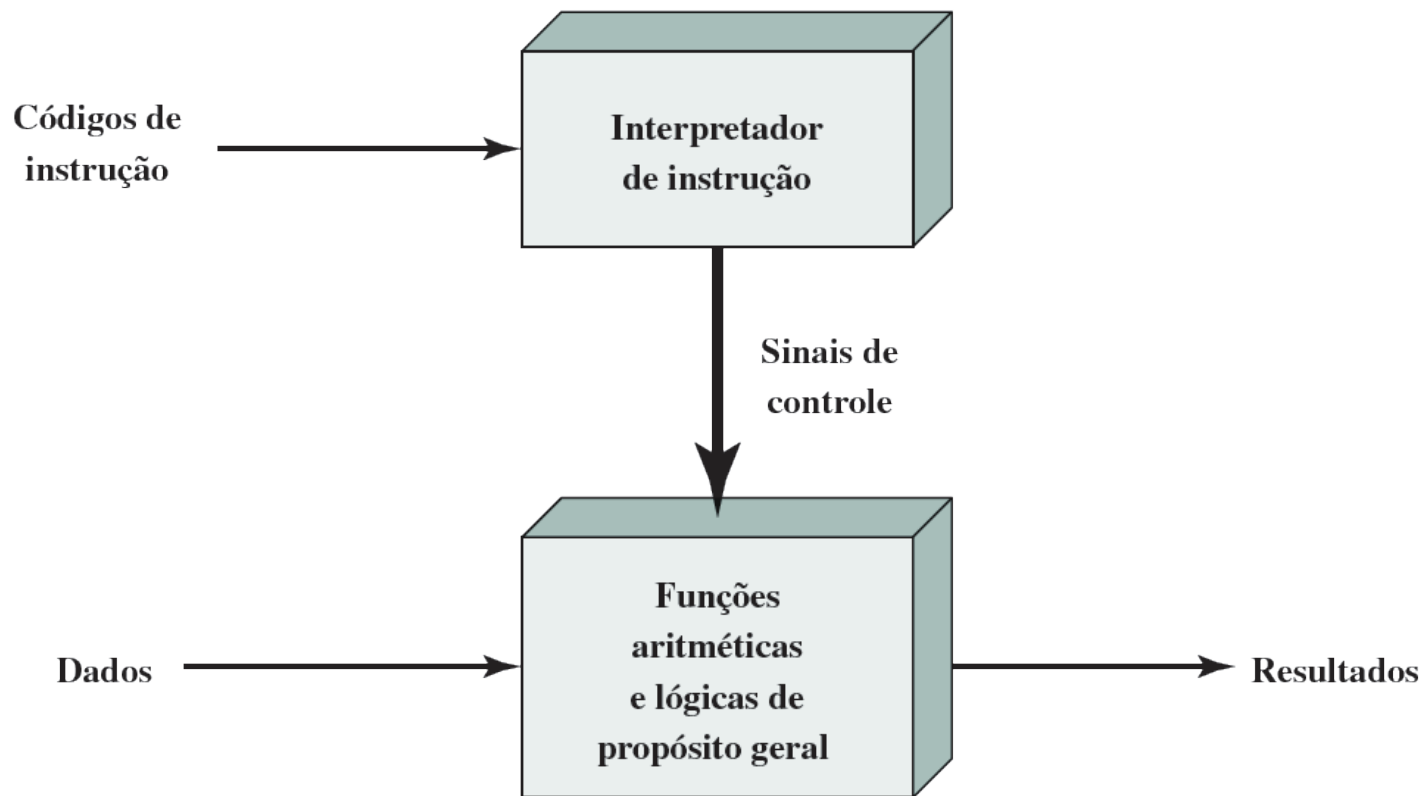
Componentes do computador

- Abordagens de hardware e software – programação no hardware:



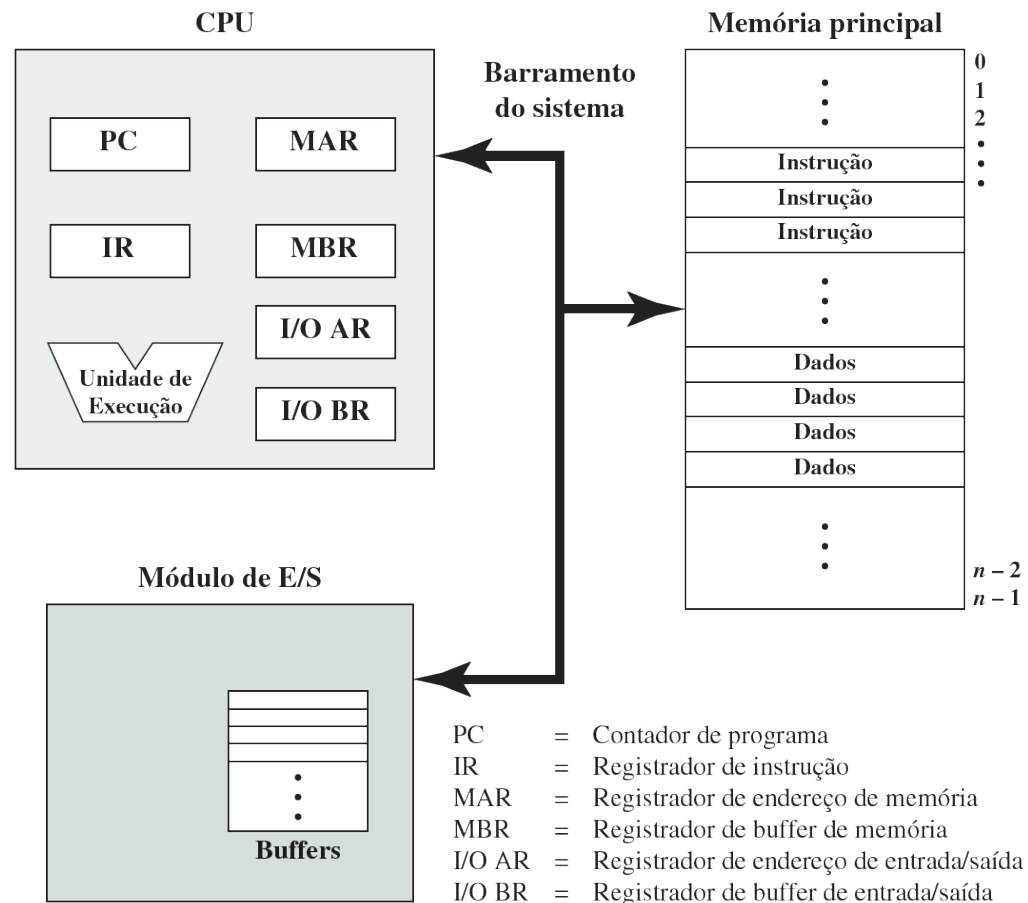
Componentes do computador

- Abordagens de hardware e software – programação no software:



Componentes do computador

- Componentes do computador – vista de nível superior:

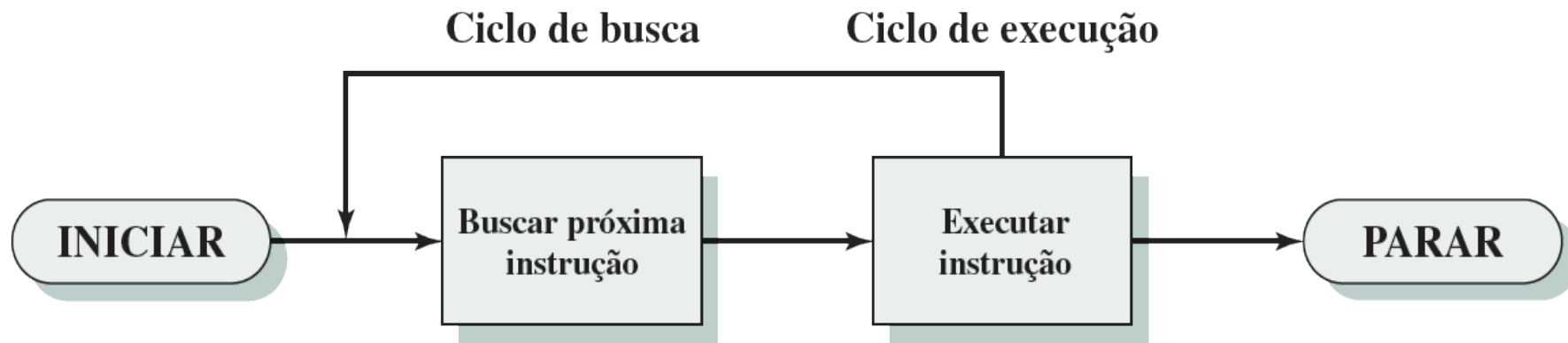


■ Função do computador

- A função básica realizada por um computador é a execução de um programa, que consiste em um conjunto de instruções armazenadas na memória.
- O processamento exigido para uma única instrução é chamado de ciclo de instrução.
- Usando a descrição simplificada em duas etapas dada anteriormente, o ciclo de instrução é representado na figura a seguir.
- As duas etapas são conhecidas como ciclo de busca (fetch) e ciclo de execução.

■ Função do computador

- Ciclo de instrução básico:



■ Busca e execução de instruções

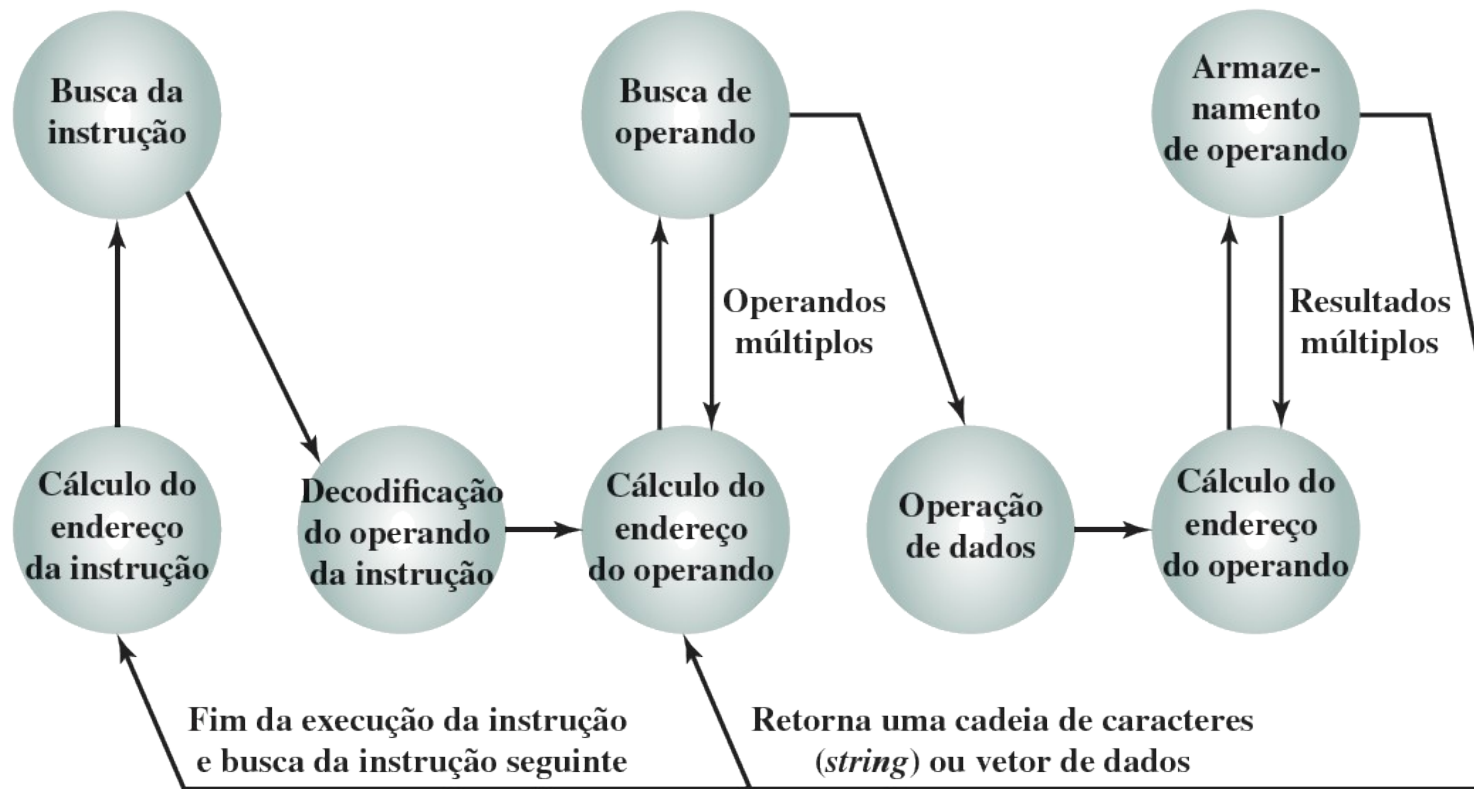
- O processador interpreta a instrução e realiza a ação solicitada.
- Em geral, essas ações estão em uma destas quatro categorias:
 - i. **Processador-memória:** os dados podem ser transferidos do processador para a memória ou da memória para o processador.
 - ii. **Processador-E/S:** os dados podem ser transferidos de ou para um dispositivo periférico, transferindo entre o processador e um módulo de E/S.
 - iii. **Processamento de dados:** o processador pode realizar alguma operação aritmética ou lógica sobre os dados.
 - iv. **Controle:** uma instrução pode especificar que a sequência de execução seja alterada.

■ Busca e execução de instruções

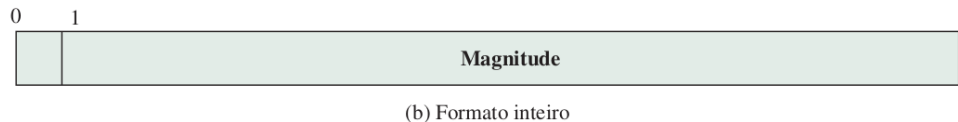
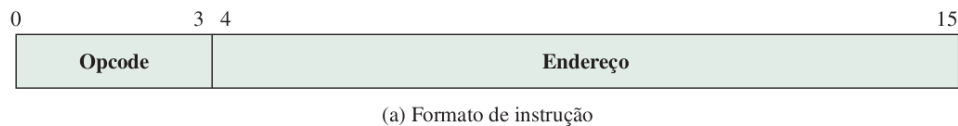
- Além disso, em vez de referências à memória, uma instrução pode especificar uma operação de E/S.
- A figura a seguir oferece uma visão mais detalhada do ciclo de instrução básico:

Busca e execução de instruções

■ Diagrama de estado de ciclo de instrução:



Busca e execução de instruções

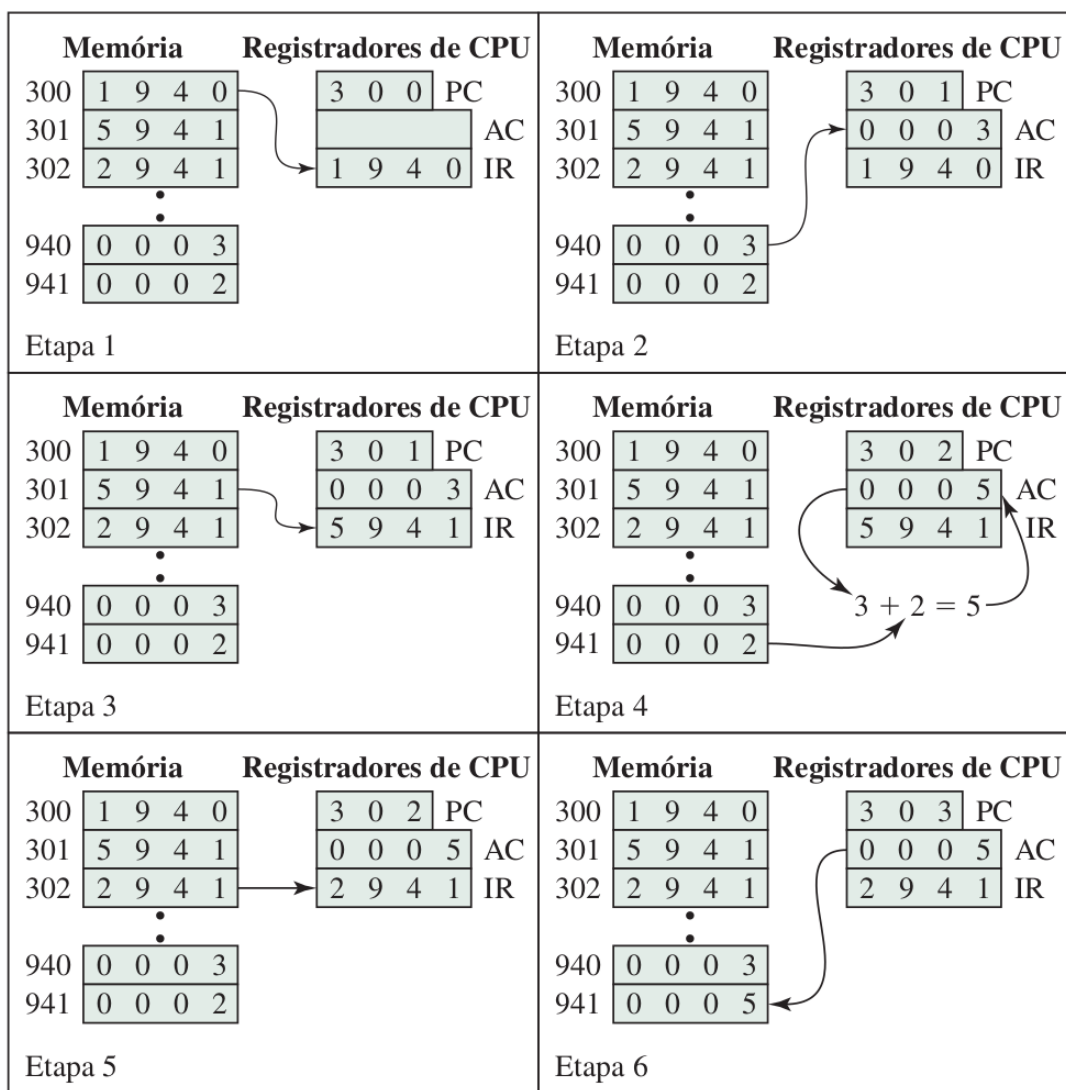


Contador de programa (PC) = endereço de instrução
 Registrador de instrução (IR) = instrução sendo executada
 Acumulador (AC) = armazenamento temporário

(c) Registradores internos da CPU

0001 = carrega AC da memória
 0010 = armazena AC na memória
 0101 = adiciona ao AC da memória

(d) Lista parcial de opcodes



■ Interrupções

- Praticamente todos os computadores oferecem um mecanismo por meio do qual outros módulos (E/S, memória) podem interromper o processamento normal do processador.
- A tabela abaixo lista as classes mais comuns de interrupções:

Programa	Gerada por alguma condição que ocorre como resultado da execução de uma instrução, como o <i>overflow</i> aritmético, divisão por zero, tentativa de executar uma instrução de máquina ilegal ou referência fora do espaço de memória permitido para o usuário.
Timer	Gerada por um timer dentro do processo. Isso permite que o sistema operacional realize certas funções regularmente.
E/S	Gerada por um controlador de E/S para sinalizar o término normal de uma operação ou para sinalizar uma série de condições de erro.
Falha de hardware	Gerada por uma falha como falta de energia ou erro de paridade de memória.

■ Interrupções

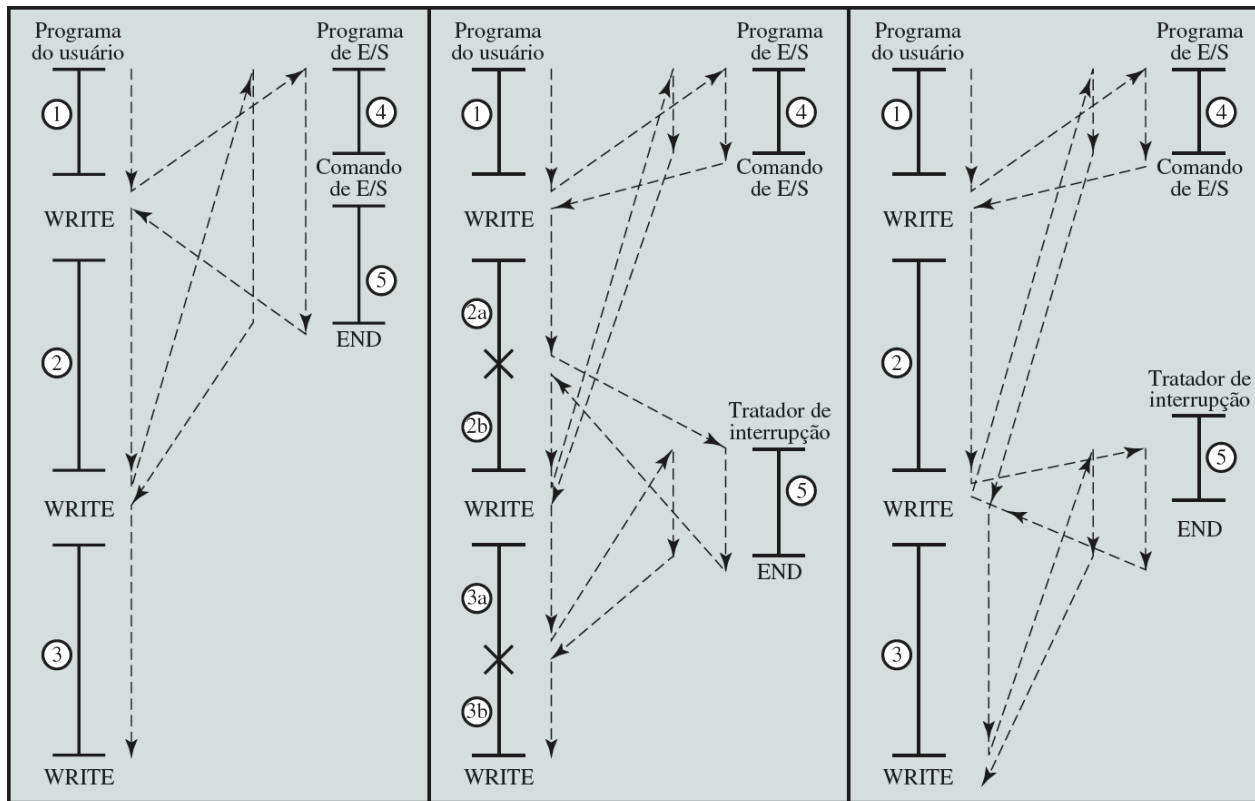
- As interrupções são fornecidas em primeiro lugar como um modo de melhorar a eficiência do processamento.
- Com as interrupções, o processador pode estar engajado na execução de outras instruções enquanto uma operação de E/S está em andamento.
- Depois que essas poucas instruções tiverem sido executadas, o controle retorna ao programa do usuário.
- Enquanto isso, o dispositivo externo está ocupado aceitando e imprimindo dados vindos da memória do computador.

■ Interrupções

- Essa operação de E/S é realizada simultaneamente com a execução de instruções no programa do usuário.
- Quando o dispositivo externo está pronto para ser atendido — ou seja, quando estiver pronto para aceitar mais dados do processador —, o módulo de E/S para o dispositivo externo envia um sinal de requisição de interrupção ao processador.
- O processador responde suspendendo a operação do programa atual, desviando para um programa para atender a esse dispositivo de E/S em particular, conhecido como tratador de interrupção, e retomando a execução original depois que o dispositivo for atendido.

■ Interrupções

- Fluxo de controle de um programa sem e com interrupções:



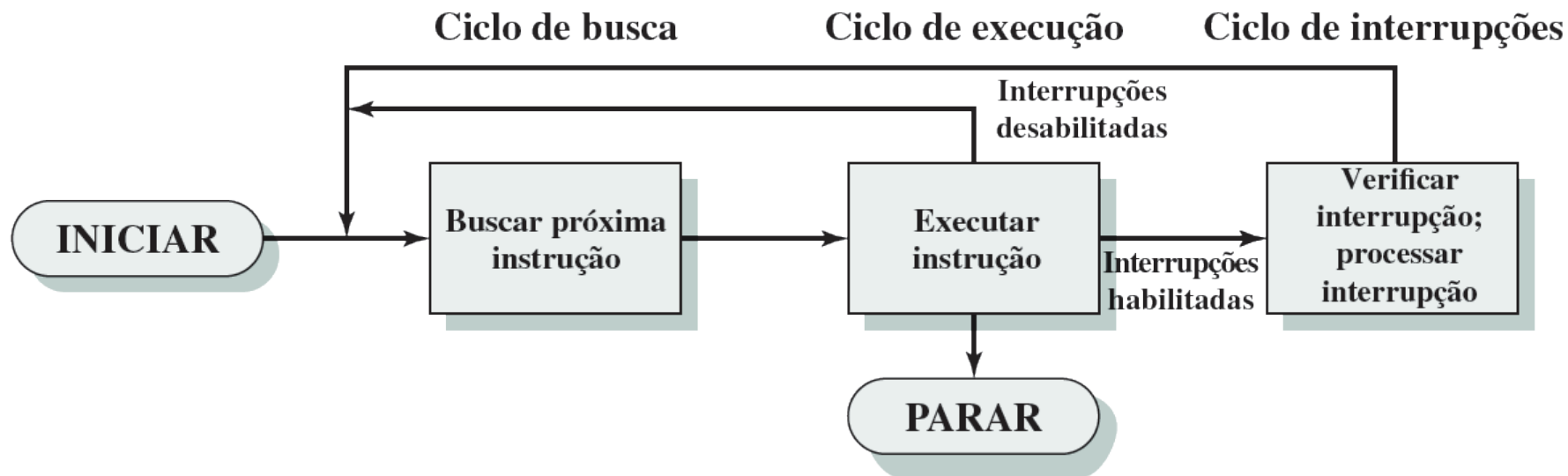
(a) Sem interrupções

(b) Interrupções; curta espera de E/S

(c) Interrupções; longa espera de E/S

■ Interrupções

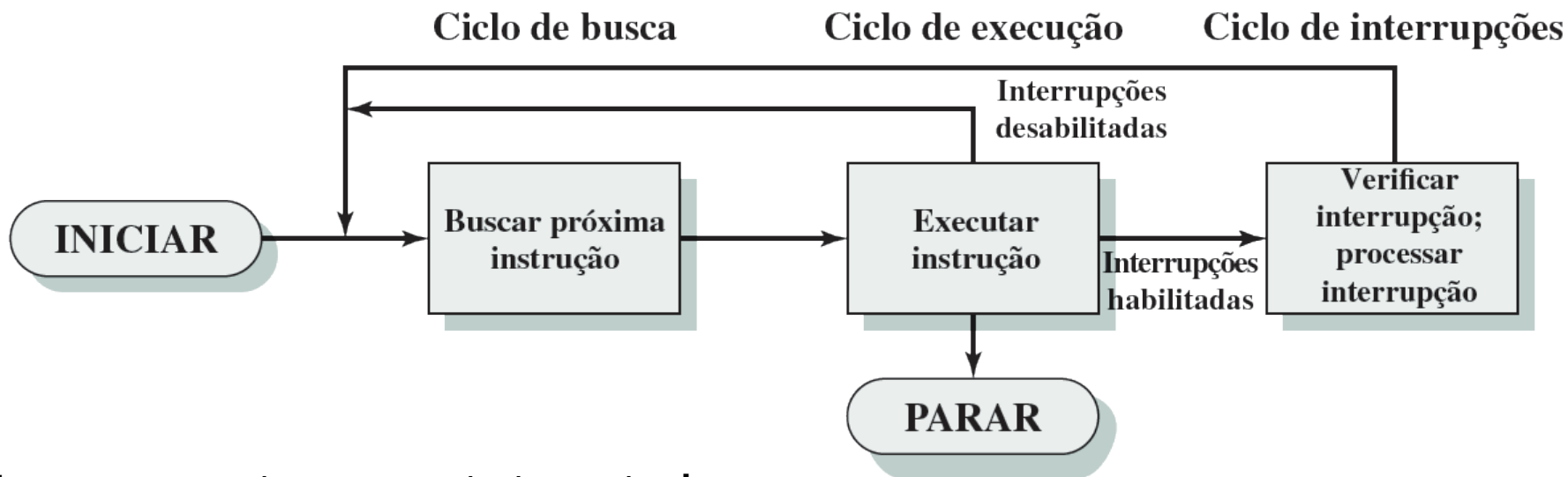
■ Ciclo de instruções com interrupção:



■ Interrupções

- Ciclo de instruções com interrupção:

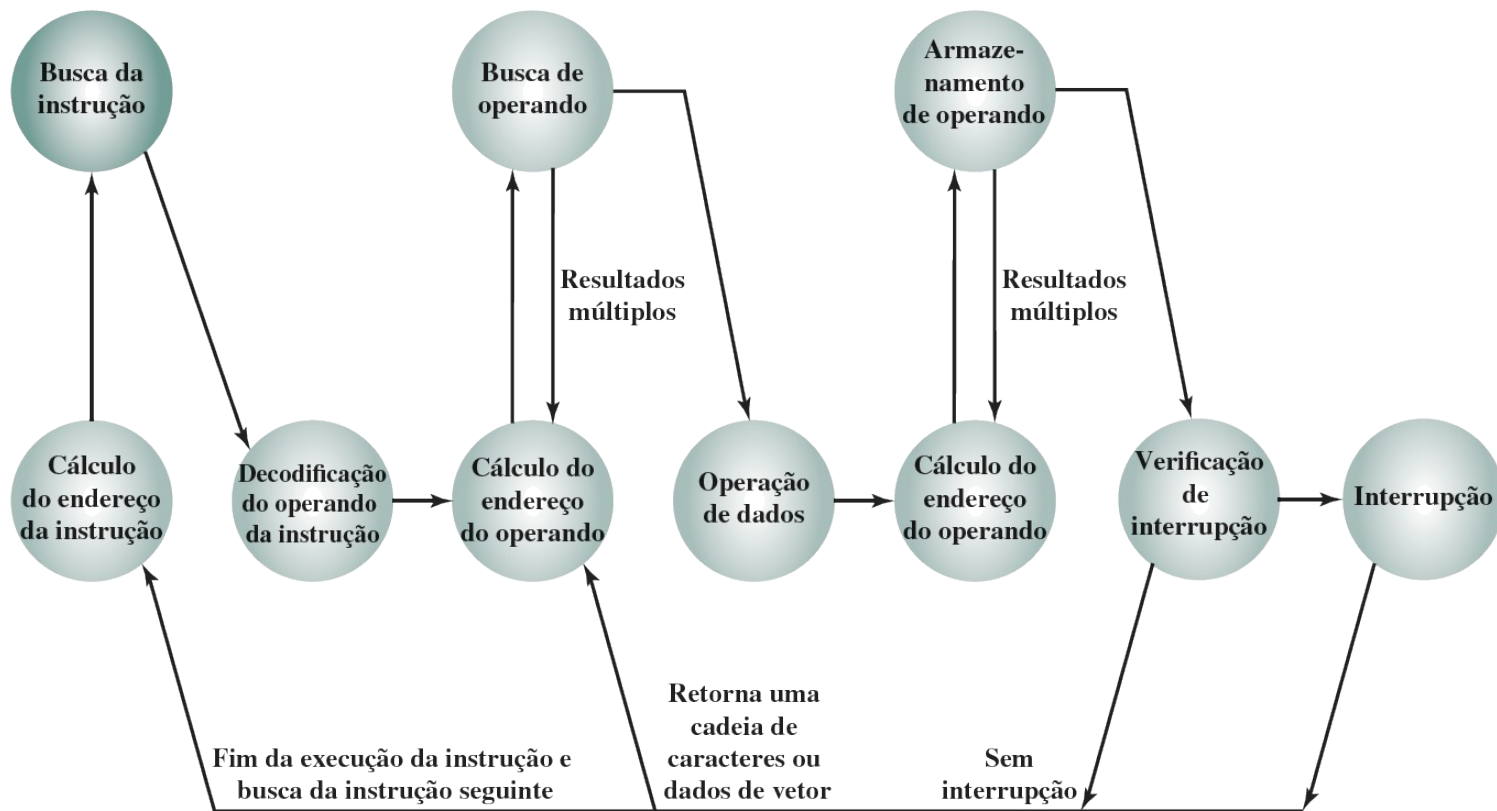
Exercício



Em que momento ocorre o tratamento da interrupção? A interrupção fica em 'espera'?

■ Interrupções

- Diagrama do estado de ciclo de instruções, com interrupções:

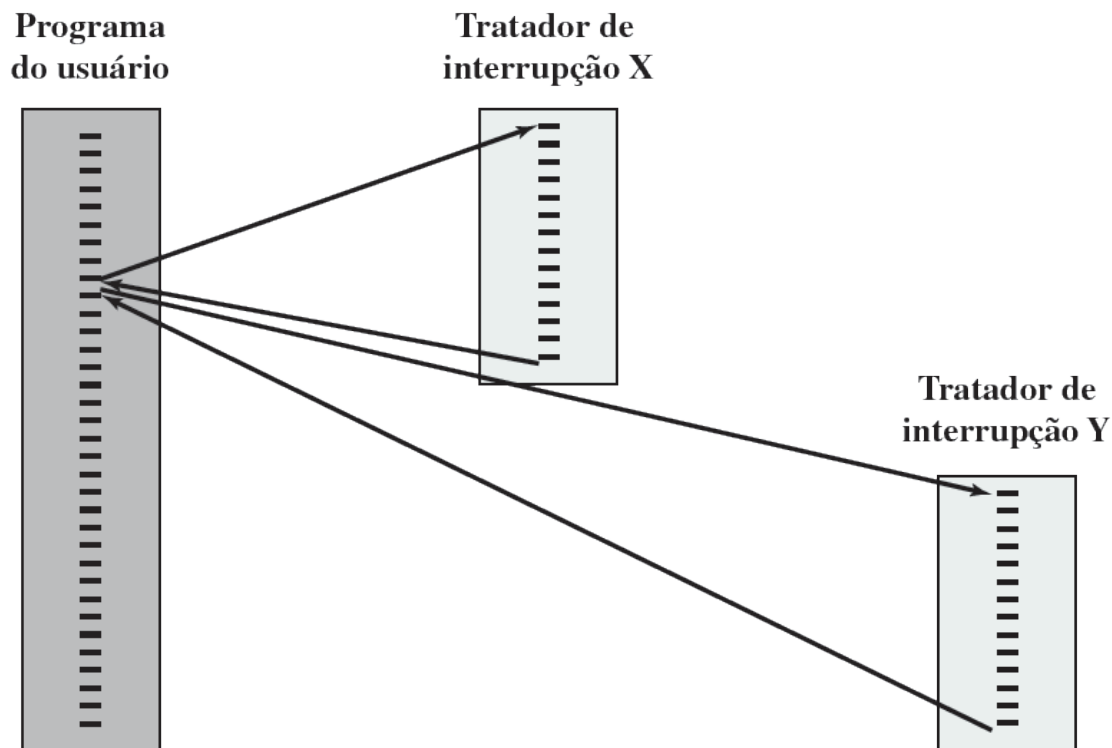


■ Interrupções múltiplas

- Duas técnicas podem ser utilizadas para lidar com múltiplas interrupções:
 - i. Desativar as interrupções enquanto uma interrupção estiver sendo processada.
 - ii. Definir prioridades para interrupções e permitir que uma interrupção de maior prioridade faça com que um tratamento de interrupção com menor prioridade seja interrompido.
- Uma interrupção desabilitada significa simplesmente que o processador pode ignorar, e ignorará, esse sinal de requisição de interrupção.

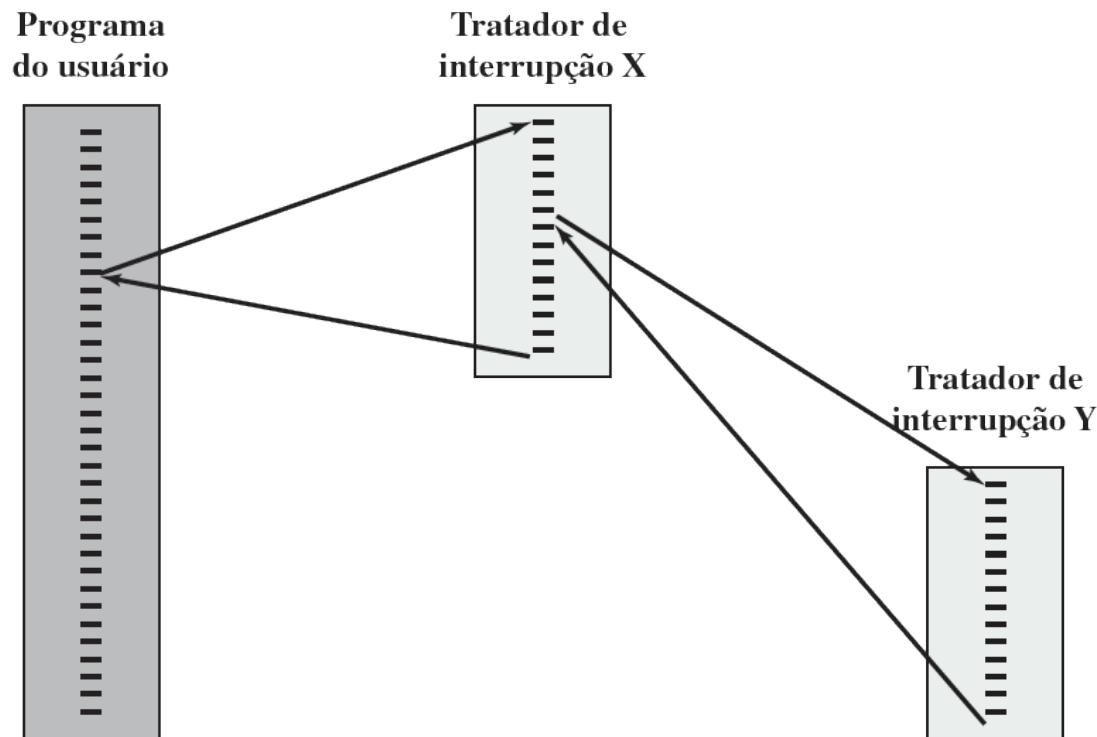
■ Interrupções múltiplas

- Transferência de controle com múltiplas interrupções – processamento de interrupção sequencial:



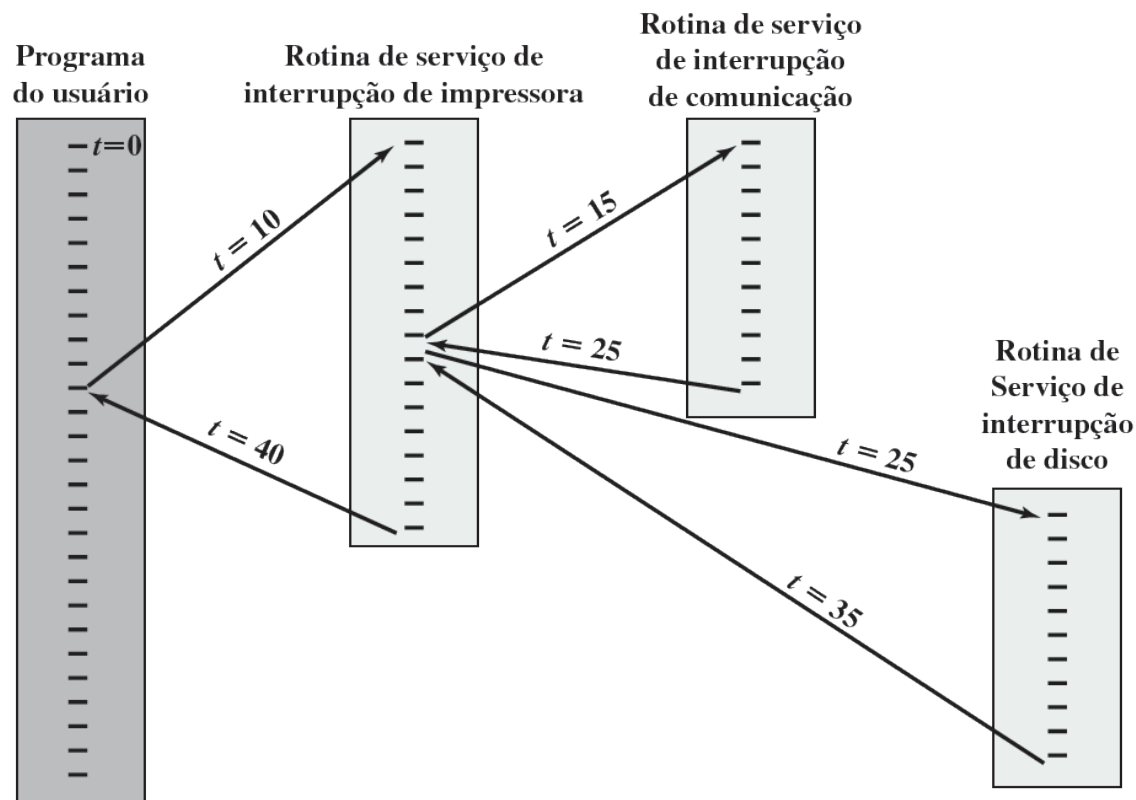
■ Interrupções múltiplas

- Transferência de controle com múltiplas interrupções – processamento de interrupção aninhado:



■ Interrupções múltiplas

- Exemplo de sequência de tempo de múltiplas interrupções:

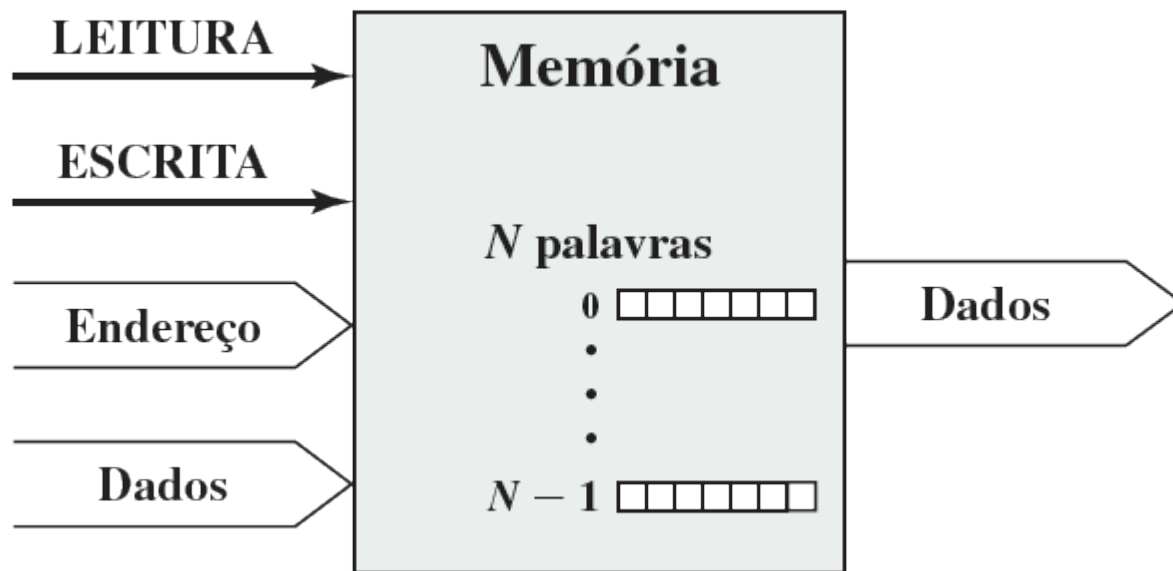


■ Estruturas de interconexão

- O conjunto de caminhos conectando os diversos módulos é chamado de estrutura de interconexão.
- O projeto dessa estrutura depende das trocas que precisam ser feitas entre os módulos.
- A figura a seguir sugere os tipos de trocas que são necessárias, indicando as principais formas de entrada e saída para cada tipo de módulo:
 - i. Memória
 - ii. Módulo de E/S
 - iii. Processador

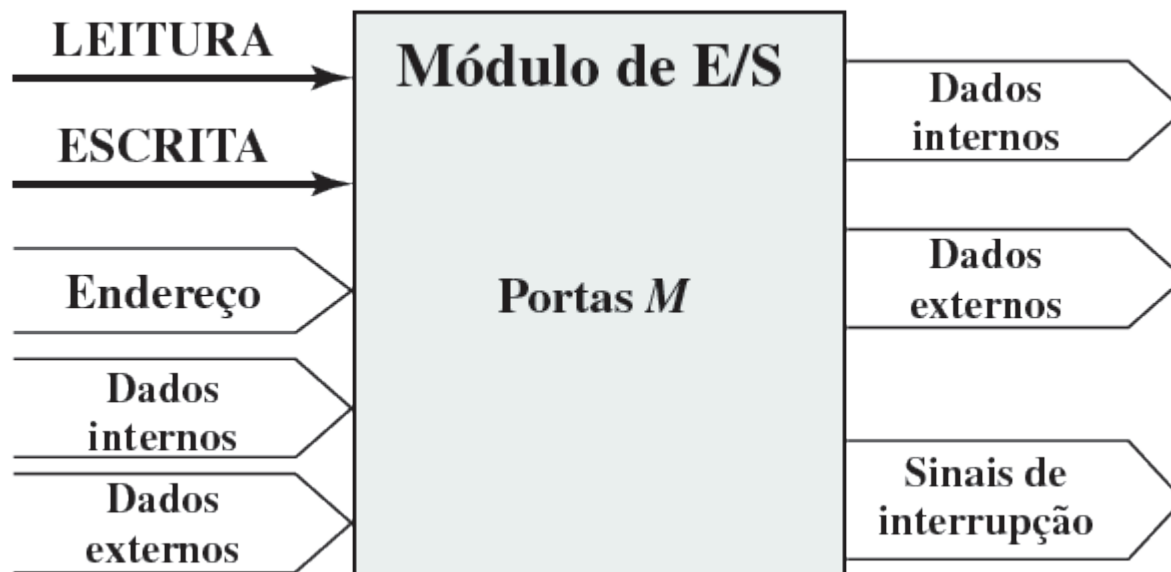
■ Estruturas de interconexão

■ Módulos do computador:



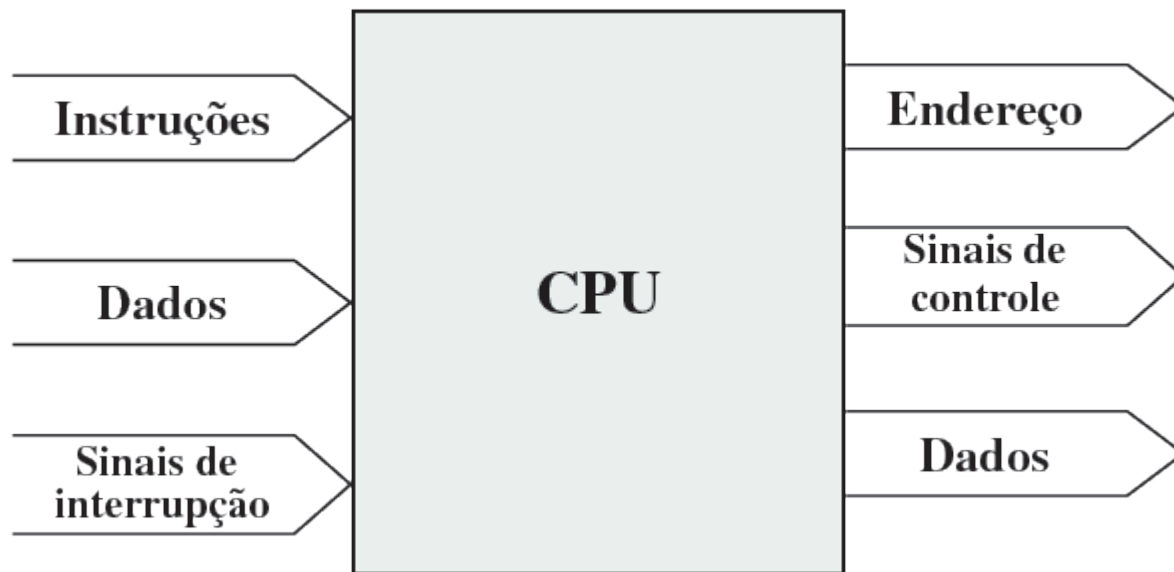
■ Estruturas de interconexão

■ Módulos do computador:



■ Estruturas de interconexão

■ Módulos do computador:



■ Estruturas de interconexão

- A estrutura de interconexão deve admitir os seguintes tipos de transferências:
 - ❖ Memória para processador
 - ❖ Processador para memória
 - ❖ E/S para processador
 - ❖ Processador para E/S
 - ❖ E/S para a memória
 - ❖ Memória para E/S

■ Interconexão de barramento

- Um barramento é um caminho de comunicação que conecta dois ou mais dispositivos.
- Uma característica-chave é que ele é um meio de transmissão compartilhado.
- Cada linha é capaz de transmitir sinais representando o binário 1 e o binário 0.
- Com o tempo, uma sequência de dígitos binários pode ser transmitida por uma única linha. Juntas, várias linhas de um barramento podem ser usadas para transmitir dígitos binários simultaneamente (em paralelo).

■ Interconexão de barramento

- As linhas de endereço são usadas para designar a origem ou o destino dos dados no barramento de dados.
- As linhas de controle são usadas para controlar o acesso e o uso das linhas de dados e endereço.

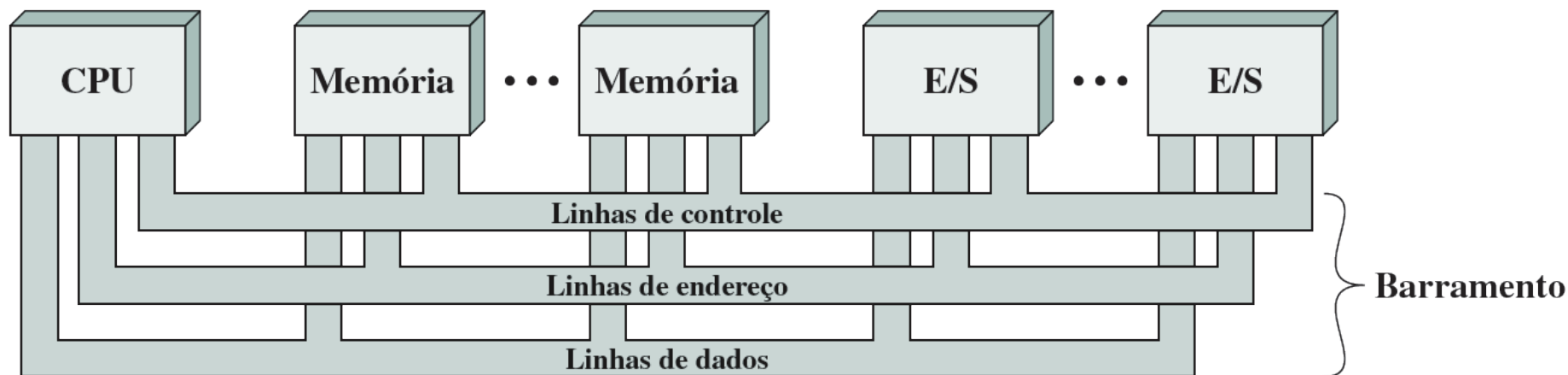
■ Interconexão de barramento

■ As linhas de controle típicas incluem:

- ❖ Escrita / Leitura de memória
- ❖ Escrita / Leitura de E/S
- ❖ ACK de transferência
- ❖ Solicitação / Concessão de barramento (bus request / bus grant)
- ❖ Requisição de interrupção (interrupt request)
- ❖ ACK de interrupção (interrupt acknowledge)
- ❖ Clock
- ❖ Reset

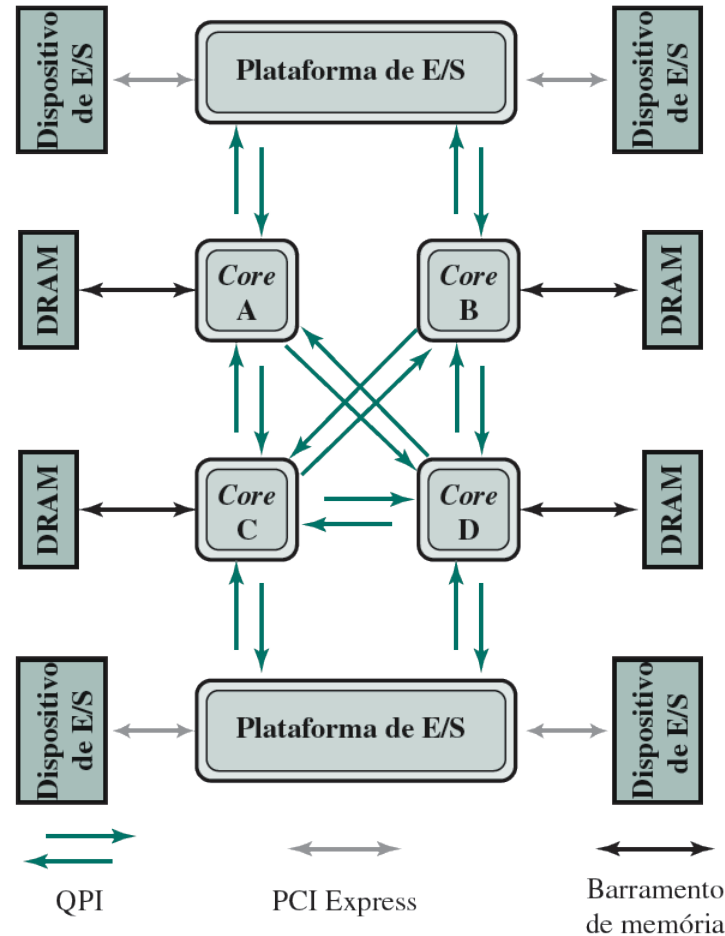
■ Interconexão de barramento

- Esquema de interconexão de barramento:



■ Interconexão ponto a ponto

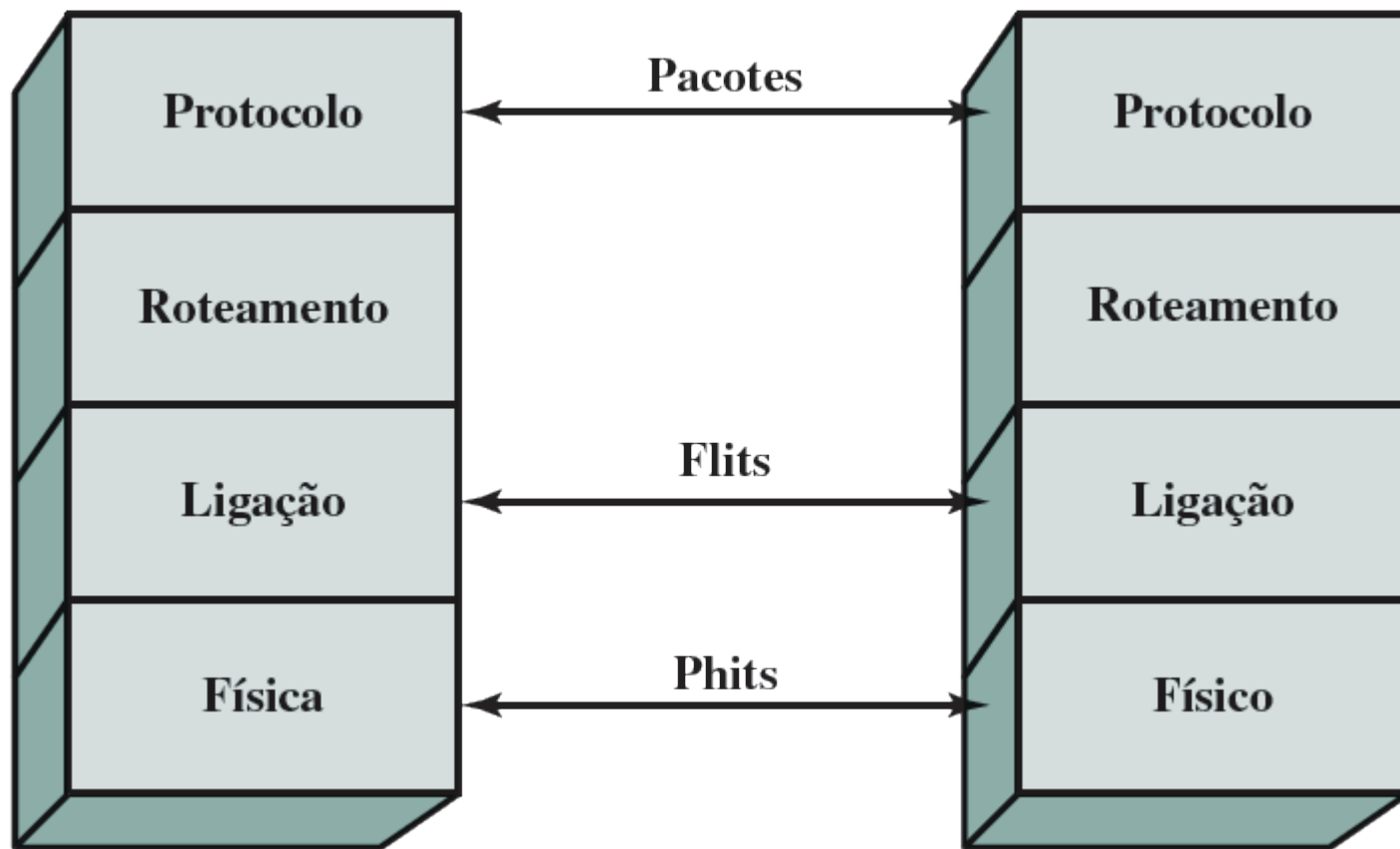
- Configuração multicore usando QPI:



■ Interconexão ponto a ponto

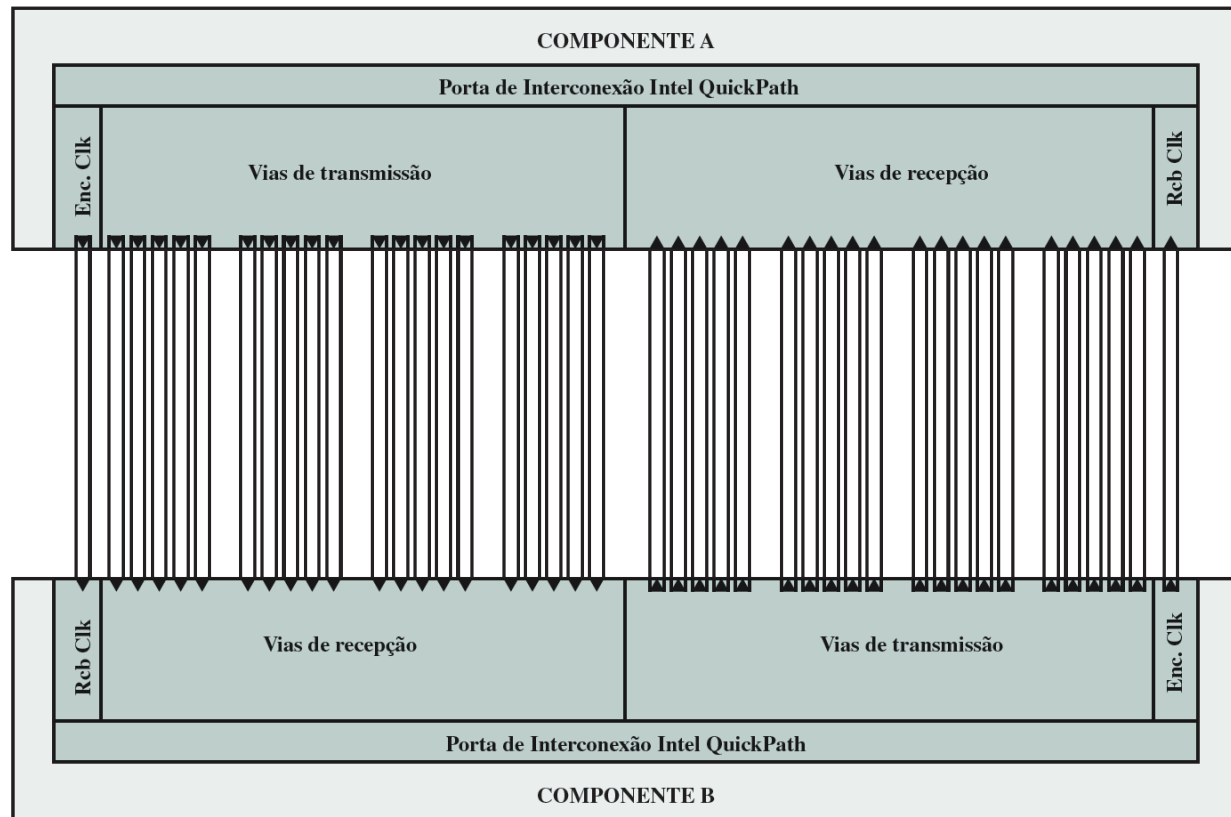
- Define-se QPI como uma arquitetura de protocolo de quatro camadas, que abrange as seguintes camadas:
 - i. Física
 - ii. Ligação
 - iii. Roteamento
 - iv. Protocolo
- A figura a seguir mostra camadas de QPI.

Interconexão ponto a ponto



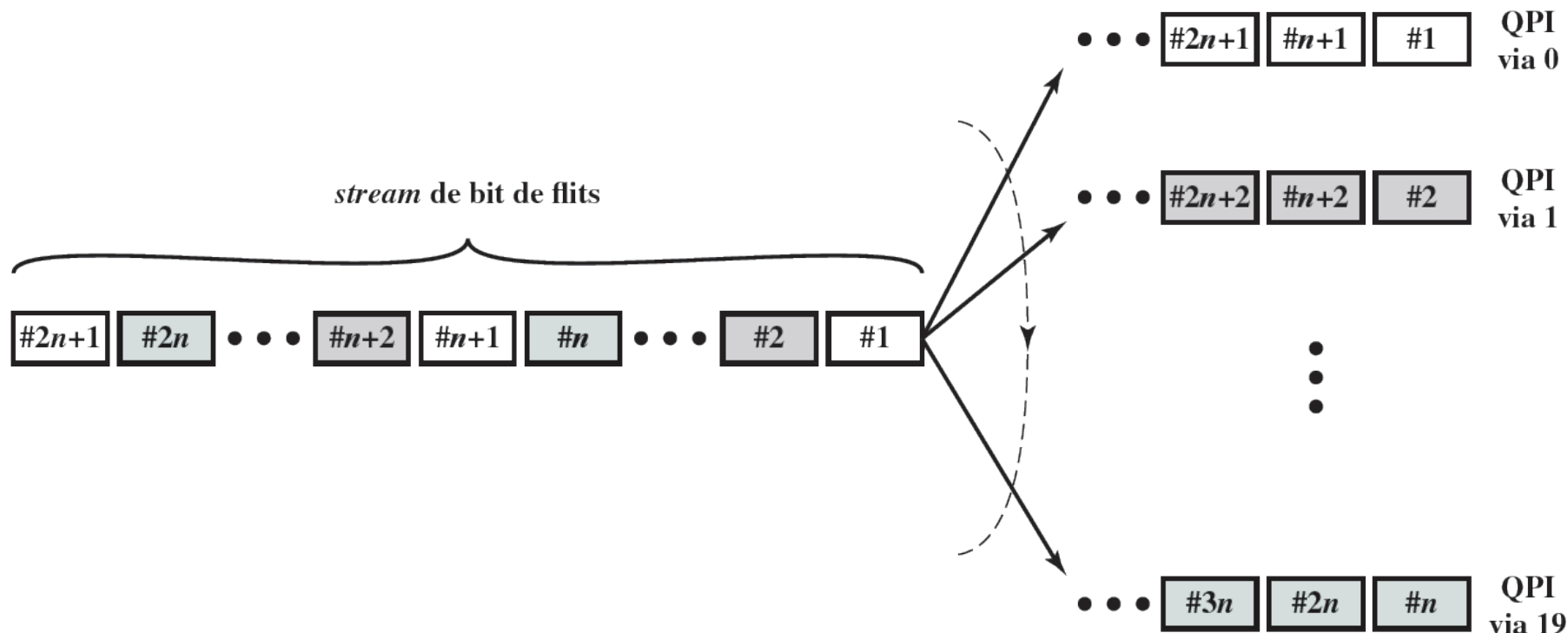
■ QPI – Camada física

- Interface física da interconexão da Intel QPI:



■ QPI – Camada física

■ Distribuição multivia da QPI:



■ QPI – Camada de ligação

- A camada de ligação QPI desempenha duas funções-chave: controle de fluxo e controle de erro.
- A função de controle de fluxo é necessária para assegurar que a entidade de QPI de envio não sobrecarregue uma entidade de QPI de recebimento ao enviar dados mais rápido do que o receptor pode processar e para limpar os buffers para receberem mais dados.
- A função de controle de erro em uma camada de ligação detecta e recupera a partir desses erros de bits, e então isola camadas mais altas a partir da experiência de erros de bits.

■ QPI – Camada de roteamento

- A camada de roteamento é usada para determinar o caminho que um pacote vai trafegar através de interconexões disponíveis do sistema.
- Em pequenas configurações as opções de roteamento são limitadas e as tabelas de roteamento são bastante simples.
- Para sistemas maiores, as opções de tabela de roteamento são mais complexas, dependendo de como (1) dispositivos são alocados na plataforma, (2) recursos do sistema são divididos e (3) eventos de confiabilidade resultam no mapeamento em torno de um recurso de falha.

■ QPI – Camada de protocolo

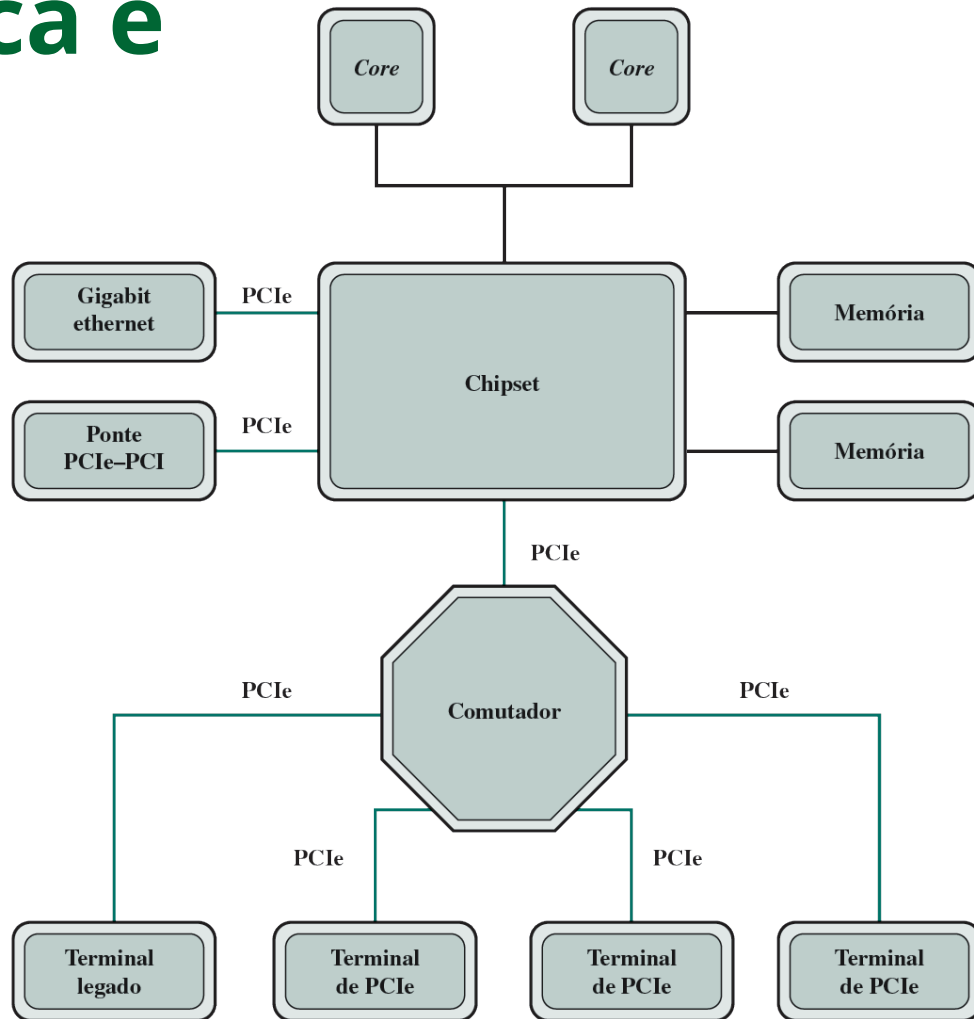
- Na camada, o pacote é definido como uma unidade de transferência.
- A definição de conteúdo de pacote é padronizada com alguma flexibilidade permitida ao atender pedidos diferentes de segmentos de mercado.
- Uma função-chave desempenhada é um protocolo de coerência de cache, que age certificando-se de que os valores da memória principal mantidos em diversas caches são consistentes.
- Uma carga útil de pacote de dados comum é um bloco de dados enviados para e a partir de uma cache.

■ PCI Express

- O barramento PCI (do inglês, *Peripheral Component Interconnect*) é um barramento de grande largura de banda, independente de processador, que pode funcionar como uma unidade intermediária ou barramento de periféricos.
- A PCI oferece melhor desempenho de sistema para subsistemas de E/S de alta velocidade.
- Uma nova versão, conhecida como PCI Express (PCIe) foi desenvolvida.
- A PCIe é um esquema de interconexão ponto a ponto que visa substituir os esquemas baseados em barramento, como a PCI.

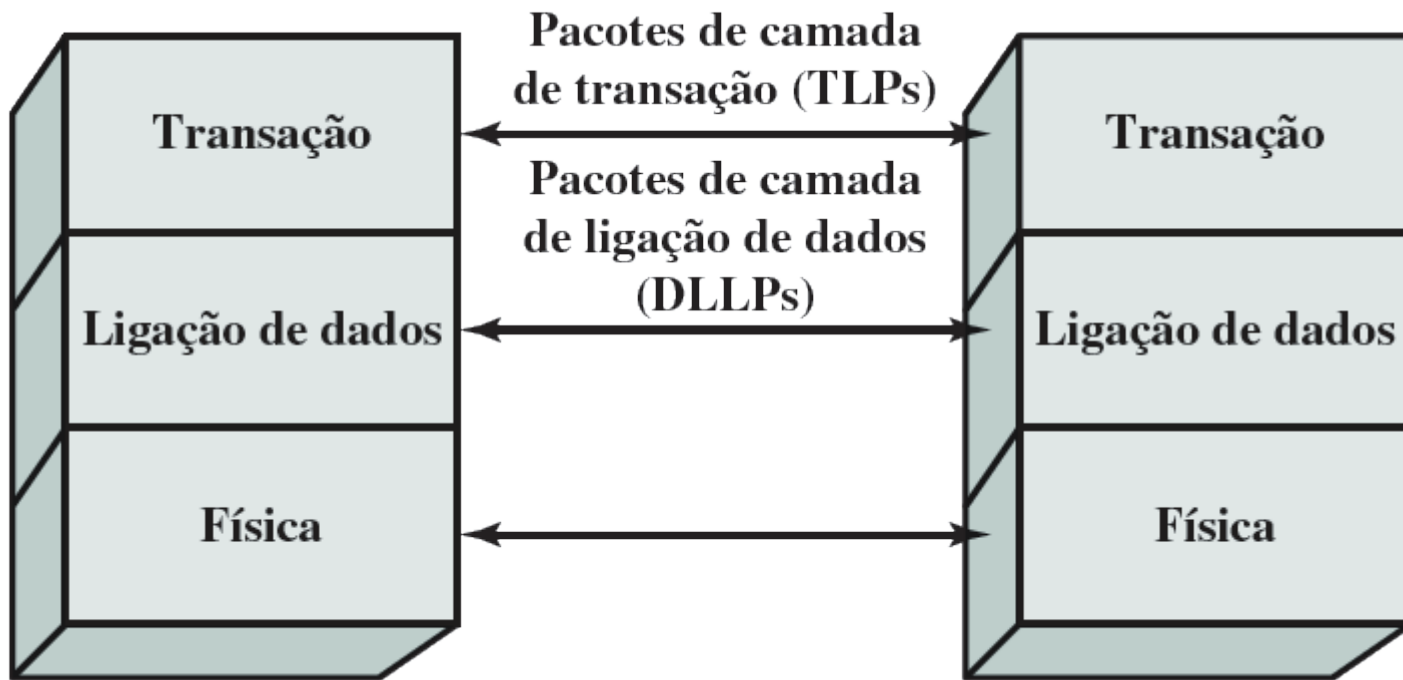
■ PCI – Arquitetura física e lógica

- Configuração comum usando PCIe:



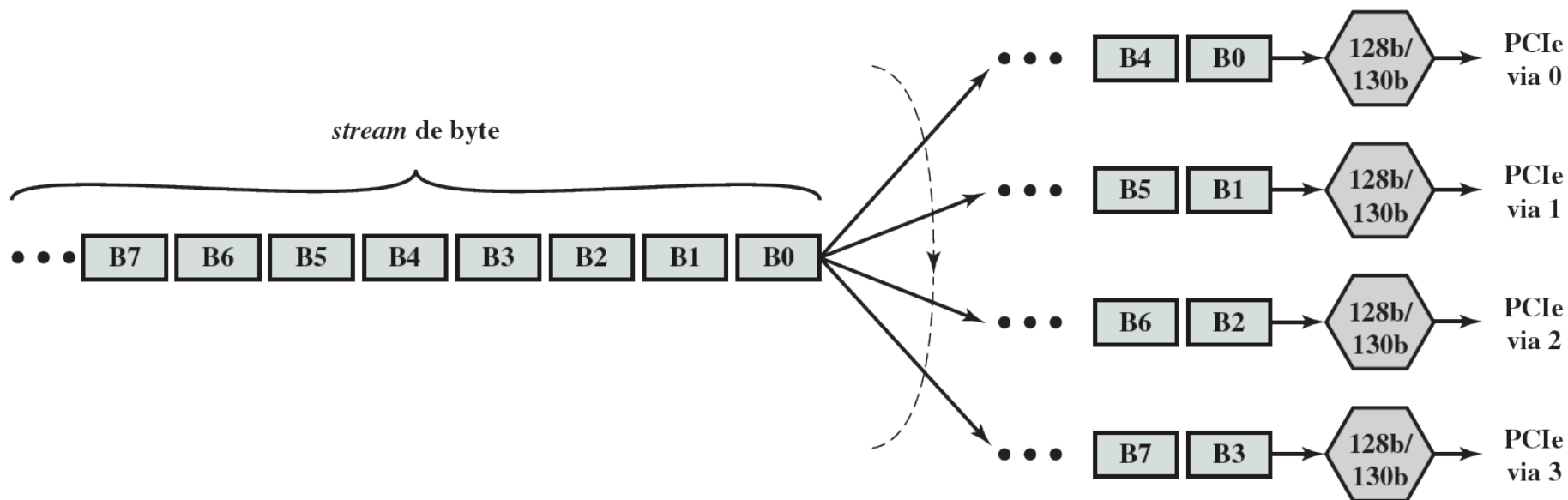
■ PCI – Arquitetura física e lógica

■ Camadas de protocolo de PCIe:



■ PCIe – Camada física

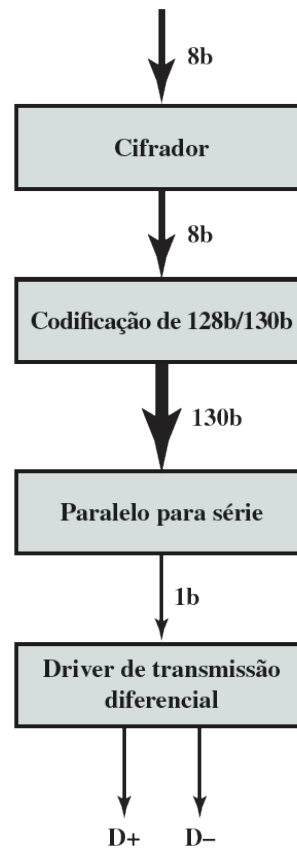
■ Técnica de distribuição multivia do PCIe:



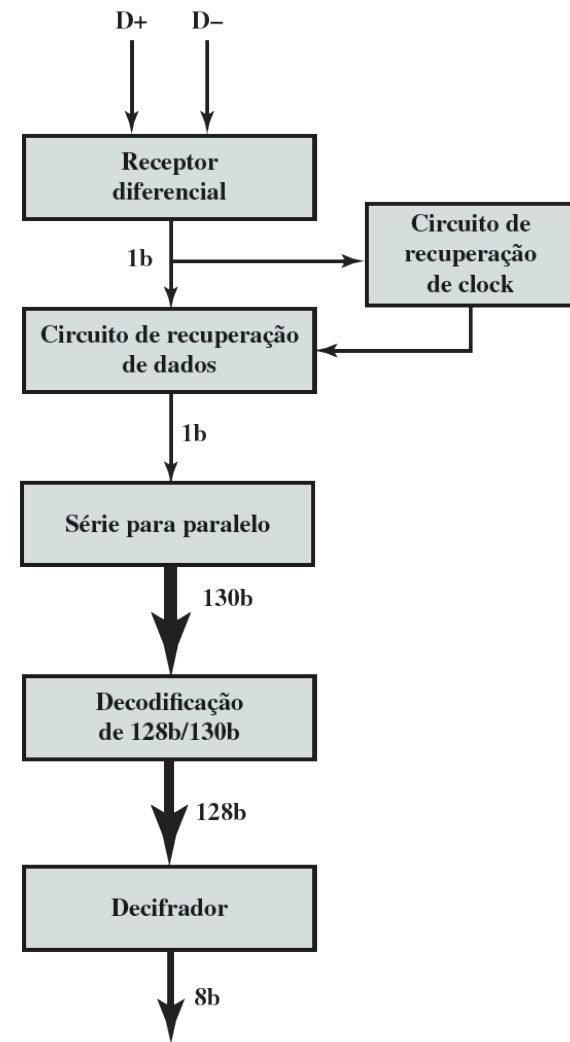
■ PCIe – Camada física

- Uma técnica comum, e a única usada na PCIe 3.0, para superar o problema de uma longa cadeia de bits de um valor é a cifragem.
- A cifragem é uma técnica de mapeamento que tende a fazer os dados aparecerem de modo mais aleatório.
- Outra técnica que pode auxiliar na sincronização é a codificação, em que bits adicionais são inseridos no stream de bits para forçar transições.
- A figura a seguir ilustra o uso da cifragem e da codificação.

PCIe - Camada física



(a) Transmissor



(b) Receptor

■ PCIe – Camada de transação

- A camada de transação (TL) recebe pedidos de leitura e escrita a partir do software acima da TL e cria pacotes de solicitação de transmissão para um destino por meio da camada de ligação.
- A TL suporta quatro espaços endereçados:
 - i. Memória
 - ii. E/S
 - iii. Configuração
 - iv. Mensagem

Tipos de transação

PCIe TLP

Espaço de endereço	Tipo de TLP	Propósito
Memória	Pedido de leitura de memória	Transfere os dados para ou a partir de um local no mapa de memória do sistema
	Pedido de bloqueio de leitura de memória	
	Pedido de escrita de memória	
E/S	Pedido de leitura de E/S	Transfere os dados para ou a partir de um local no mapa de memória do sistema para dispositivos legado
	Pedido de escrita de E/S	
Configuração	Pedido de leitura de tipo de configuração 0	Transfere os dados para e a partir de um local no espaço de configuração de um dispositivo PCIe
	Pedido de escrita de tipo de configuração 0	
	Pedido de leitura de tipo de configuração 1	
	Pedido de escrita de tipo de configuração 1	
Mensagem	Pedido de mensagem	Proporciona uma mensagem em banda e um relato de evento
	Pedido de mensagem com dados	
Memória, E/S, Configuração	Finalização	Retornado para certos pedidos
	Finalização com dados	
	Finalização bloqueada	
	Finalização bloqueada para dados	

■ PCIe – Camada de ligação de dados

- O propósito da camada de ligação de dados PCIe é assegurar a entrega confiável pela ligação de PCIe.
- Os pacotes de camada de ligação de dados originam-se na camada de ligação de dados de um dispositivo de transmissão e terminam na DLL do dispositivo no outro final da ligação.
- Há três grupos importantes de DLLPs usados para gerenciar a ligação:
 - i. pacotes de controle de fluxo,
 - ii. pacotes de gerenciamento de potência e
 - iii. pacotes TLP ACK e NAK.