




# Fundamentos de Arquitetura de Computadores



Prof. Marcos Quinet

Universidade Federal Fluminense – UFF

Pólo Universitário de Rio das Ostras - PURO



# Hardware de um Sistema Computacional

- Hardware: são os componentes e dispositivos eletrônicos que operando em conjunto com outros componentes ou mesmo individualmente realizam uma das funções de um sistema de computação
  - Exs.: disco rígido, placa-mãe, teclado, monitor, etc.
- Software: é uma sequência de códigos ou instruções a ser interpretada por uma parte do hardware para executar uma tarefa requerida e gerar os resultados desejados
  - Exs.: Windows (sistema operacional), Internet Explorer (navegador da Internet), Word (editor de texto), etc.

# Estrutura Básica

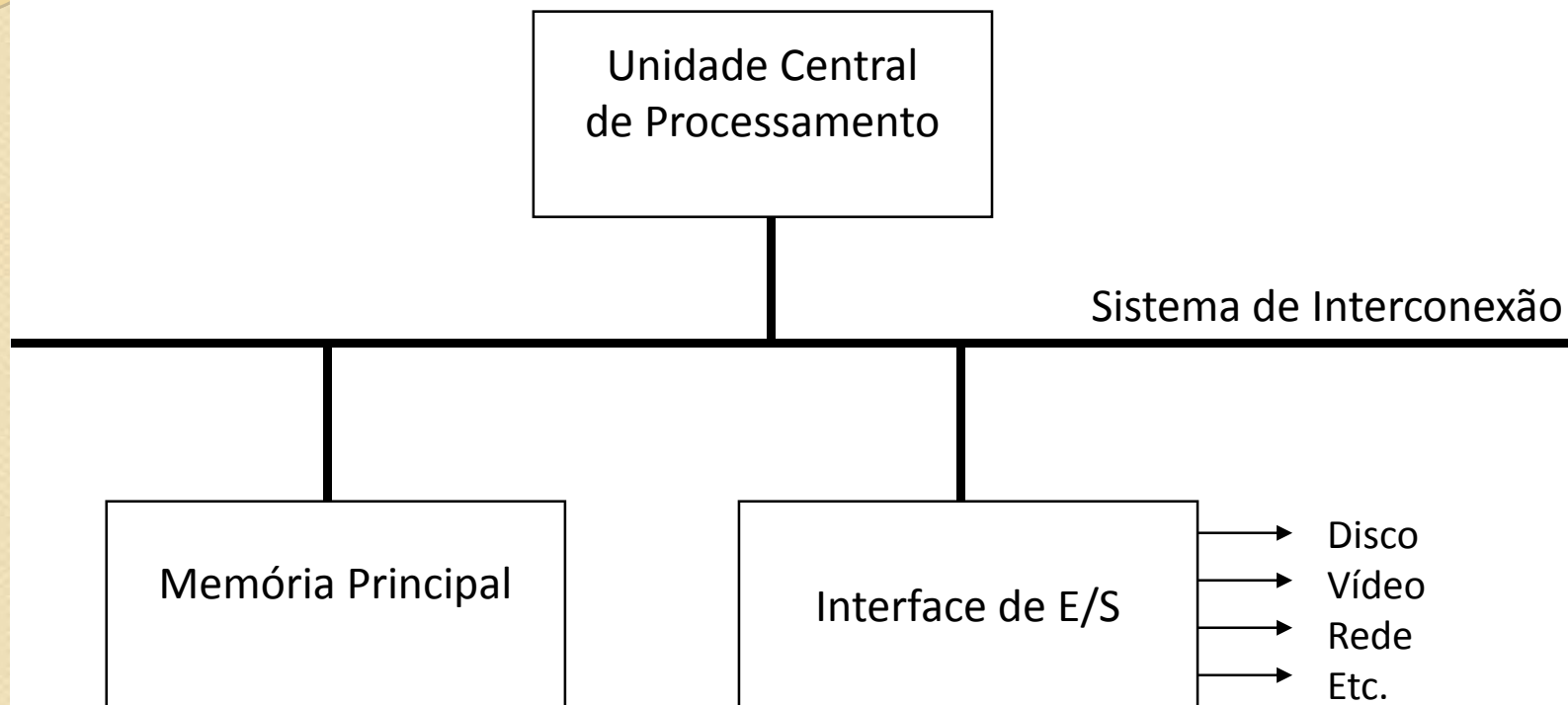
- A estrutura básica de um computador digital deve apresentar as seguintes características:
  - É necessário que existam meios de fornecer informações ao sistema (**entrada de dados**);
  - Deve haver uma forma de disponibilizar aos usuários os resultados produzidos pelo sistema (**saída de dados**);
  - Existe uma unidade responsável por efetuar operações aritméticas e lógicas (cálculos e decisões) sobre as informações fornecidas, produzindo resultados (**unidade lógica e aritmética**).



# Estrutura Básica

- O computador deve possuir um recurso que o permita reter, armazenar e recuperar informações sobre as quais serão efetuados os cálculos, resultados parciais produzidos pela ULA, ou informações de como manipular estas informações (**memória principal**)
- Deve haver um componente cuja função seja administrar todas as demais funções do computador, ou seja, dispositivos de E/S, processamento, e acessos a memória. Este componente é chamado **unidade de controle**

# Componentes Básicos





# Componentes Básicos

- A análise de um sistema de computação de um nível mais alto permite estabelecermos uma compreensão mais clara da estrutura de funcionamento de um computador, identificando claramente os seguintes pontos-chave:
  - O comportamento de cada componente, através dos dados e sinais de controle que são trocados com os demais componentes;
  - A estrutura de interconexão utilizada.



# Componentes Básicos

- Através de uma visão global da estrutura de funcionamento do sistema é possível identificar as características que melhor identificam a qualidade do mesmo, permitindo:
  - Identificação dos pontos críticos de desempenho do sistema;
  - Previsão de falhas;
  - Visualização de caminhos alternativos.
- E com uma maior compreensão do todo, define-se se melhorias ao sistema (desempenho e/ou confiabilidade) devem ser feitas através de mudanças no projeto ou pelo aumento de capacidade de componentes individuais



# Função de um Computador

- A função básica de um computador é, através de seus recursos computacionais (*hardware*), executar um programa (*software*), que é formado por um conjunto de instruções armazenadas na memória
- O modelo de arquitetura utilizado nos computadores atuais é fundamentalmente o mesmo estabelecido por *John Von Neumman* em 1946, sendo utilizado no projeto do computador EDVAC





# Arquitetura de von Neumman

- A arquitetura de *Von Neumman* foi estabelecida baseando-se nos seguintes conceitos:
  - Os dados e instruções são armazenados em uma única memória de leitura e escrita;
  - O conteúdo dessa memória é endereçado pela sua posição, independente do tipo de dados nela contidos;
  - A execução de instruções ocorre de modo sequencial, salvo quando essa sequência é explicitamente alterada de uma instrução para a seguinte.



# Programação por Hardware e Software

- Um computador é formado por um conjunto de componentes lógicos, que podem ser combinados de diferentes formas. Esta combinação depende da natureza de cada aplicação em particular
- Uma forma de arranjar estes componentes é desenvolver uma configuração projetada especificamente para a tarefa em questão, ou seja, através de uma configuração em particular poderemos executar a tarefa desejada, mas somente ela, nenhuma outra mais
- Este método é conhecido por programação por hardware (programa *hardwired*)

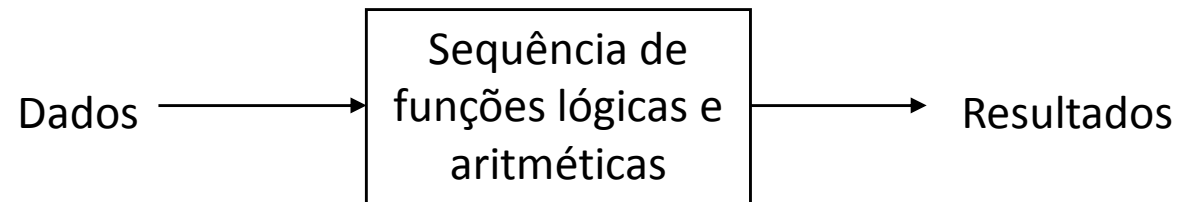


# Programação por Hardware e Software

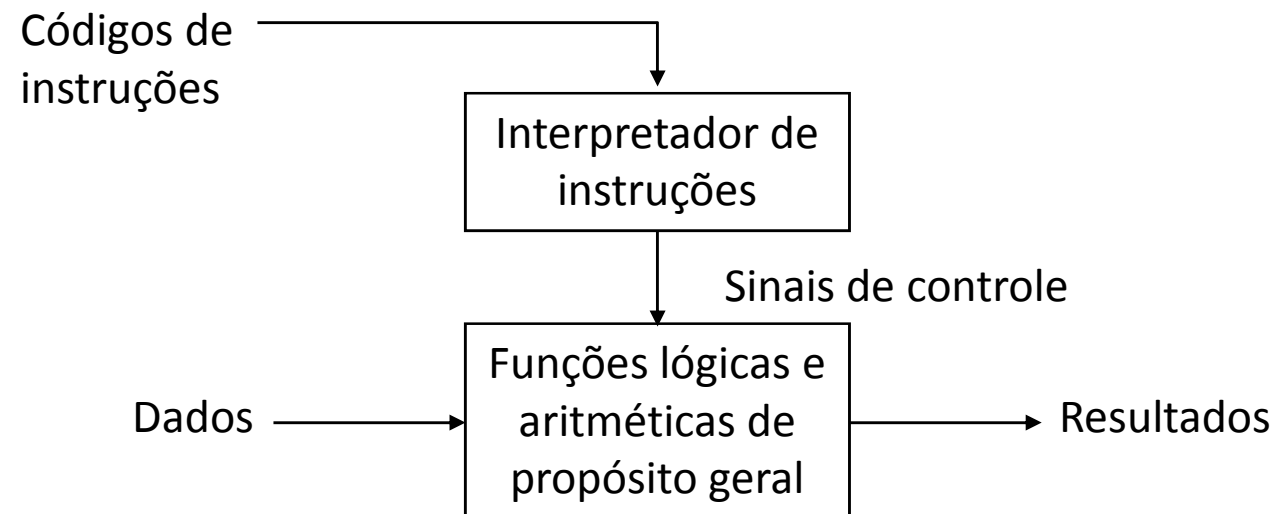
- A programação por hardware mostra-se adequada para algumas aplicações em particular, mas de forma geral é muito limitada, especialmente em um cenário onde novas tecnologias surgem rapidamente e com baixos custos, e novas técnicas de manipulação de dados são desenvolvidas, tornando os sistemas inflexíveis obsoletos em um curto período de tempo
- A solução desenvolvida foi criar um sistema de propósito geral, cuja função de cada componente seria definida por um conjunto de instruções fornecido por um usuário. Surgia assim a programação por software

# Programação por Hardware e Software

## Hardware:



## Software:





# Programação por Hardware e Software

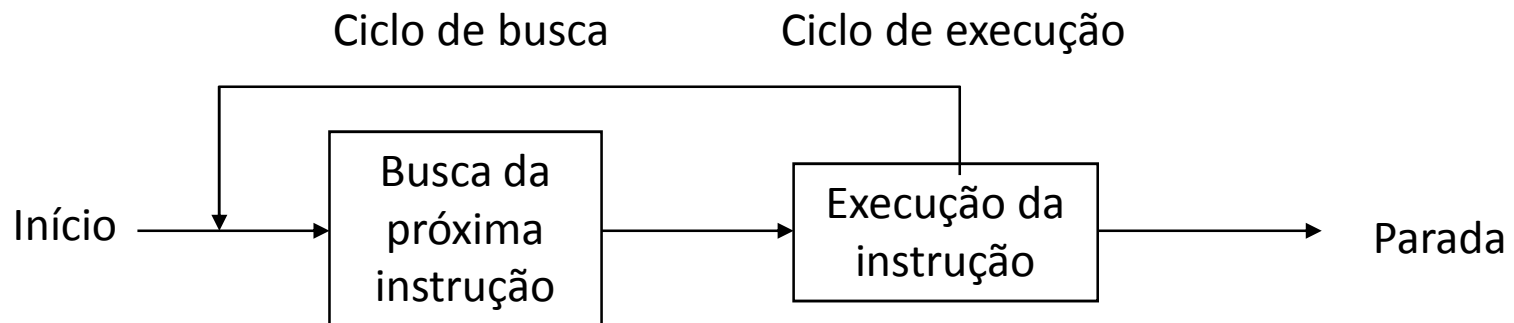
- Um programa é constituído de uma sequência de instruções, e para cada uma destas instruções uma operação lógica ou aritmética é executada sobre algum dado.
  - Para cada uma destas instruções é necessário um novo conjunto de sinais de controle, responsáveis por funções como a movimentação de dados entre registradores e ativação de funções específicas da ULA
- O interpretador de instruções é responsável por converter a sequência de passos fornecida pelo usuário em sinais de controle enviados para os componentes envolvidos nas tarefas

# Ciclo de Instruções

- Um computador tem como função básica executar programas, que são um conjunto de instruções armazenadas na memória
- O processamento de instruções, em sua forma mais simples, é feito em duas etapas:
  - O processador busca (lê) instruções na memória, uma de cada vez;
  - A instrução lida é então executada;
- Durante a execução de um programa o processo de busca e execução de instruções é feito repetidamente, até o fim do conjunto de operações

# Ciclo de Instruções

- A execução de uma instrução pode requerer várias operações, dependendo de sua natureza
- Denominamos de **ciclo de instrução** todo o processamento necessário para a execução de uma instrução; um ciclo de instrução é formado por duas etapas: um ciclo de busca e um ciclo de execução



# Execução de Instruções

- A execução de uma instrução envolve a realização de uma sequência de operações, chamadas de passos de execução, executados na seguinte ordem:


<b>busca</b>	<b>decodificação</b>	<b>execução</b>	<b>resultado</b>
--------------	----------------------	-----------------	------------------

- No passo de busca, o processador realiza o acesso ao código binário da instrução, que está armazenado na memória principal



# Execução de Instruções

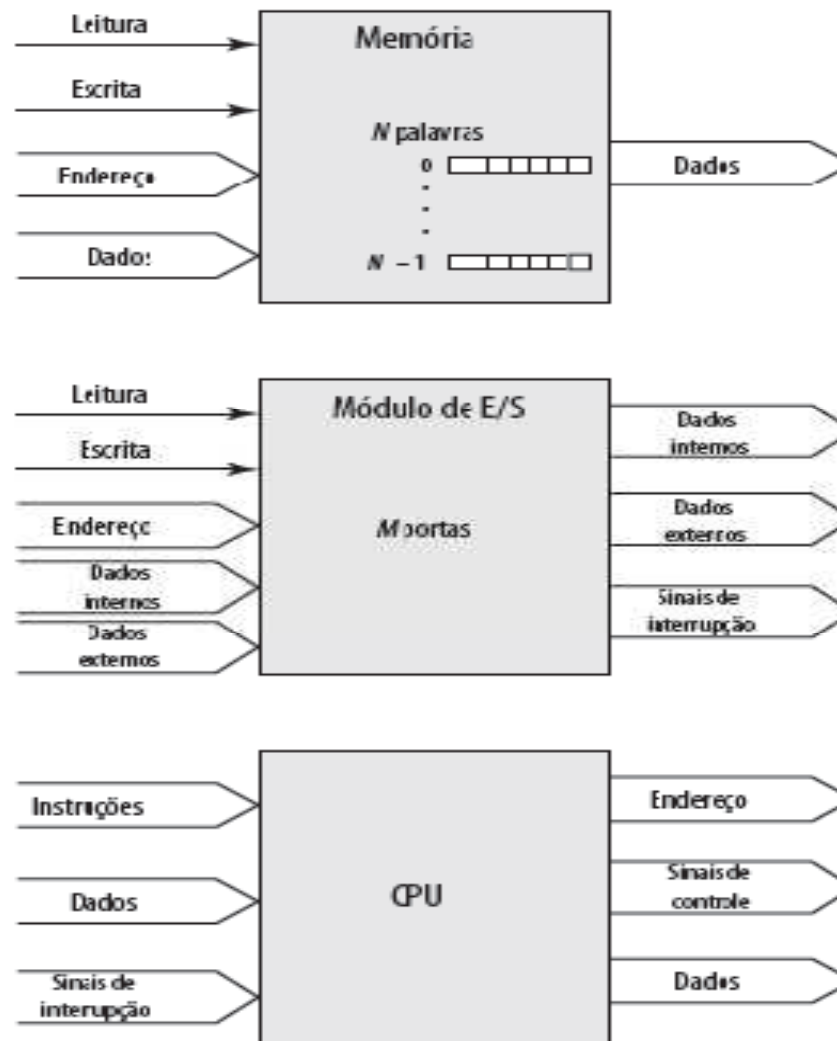
- Na etapa de decodificação, as informações contidas no código da instrução acessada no passo anterior são interpretadas
- Na execução, a operação indicada pela instrução (por exemplo, uma operação na ULA) é efetuada
- No quarto e último passo, o de resultado, é armazenado na memória ou em um registrador o resultado produzido pela instrução executada



# Módulos de um Sistema Computacional

- Como existem muitas funções a serem desempenhadas pelos diferentes componentes de um sistema computacional, estes são separados em módulos
- Cada módulo tem uma função específica, e portanto, recebe diferentes dados de entrada e produz diferentes tipos de saída
- Devido a estas diferenças, os barramentos de entrada e saída em cada módulo apresentam diferenças na capacidade de transmissão, quantidade de vias e tipos de entrada e saída gerados

# Módulos de um Sistema Computacional - exemplo



# Barramentos - definição

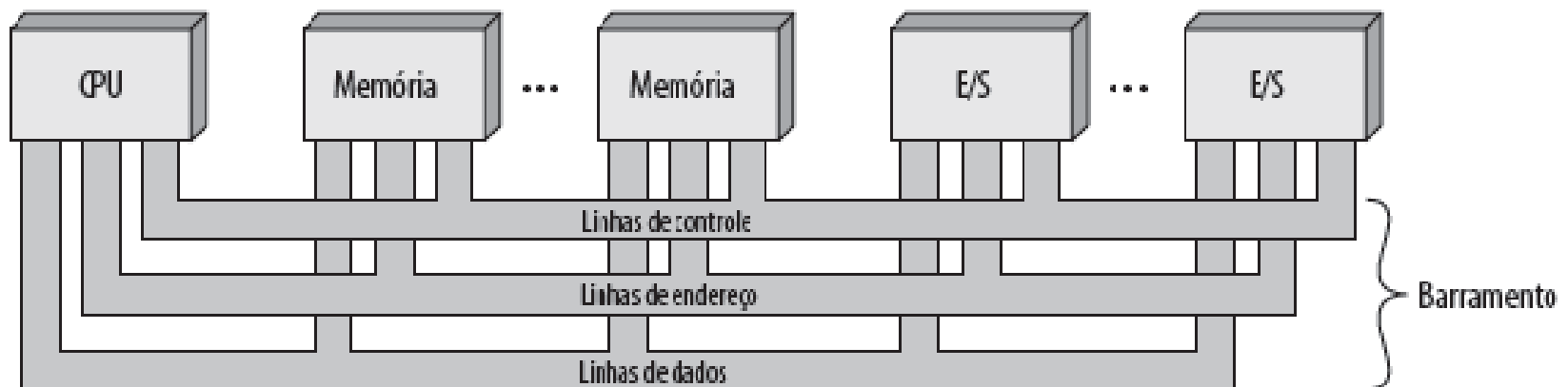
- Um caminho de comunicação conectando dois ou mais dispositivos.
- Normalmente, a transmissão é realizada em *broadcast*.
- Frequentemente, são agrupados:
  - Uma série de canais em um barramento.
  - Por exemplo, barramento de dados de 32 bits são 32 canais de bits separados.
- Linhas de potência podem não ser mostradas em projetos da parte lógica.

# Barramentos

- Barramentos são conjuntos de fios que transportam sinais de dados, endereço e controle, responsáveis pela interconexão dos componentes; existem barramentos internos e externos ao processador
- Diferentes modelos de barramentos diferenciam-se por sua capacidade de transmissão (quantidade de bits transmitidos por ciclo), modelo de endereçamento e frequência de operação; atualmente, o modelo mais usado é o *PCI Express*
- Para não sobrecarregar o processador com a tarefa de controle de todo o fluxo de informações trafegadas pelos barramentos são utilizados circuitos específicos para esta tarefa, chamados *chipsets*, presentes na placa-mãe

# Esquema de interconexão de barramento

- Podem ser transportados por barramentos:
  - Endereços de memória;
  - Dados;
  - Sinais de controle





# Barramentos de dados

- Transporta dados.
  - Lembre-se de que não existe diferença entre “dados” e “instruções” neste nível.
- Largura é um determinante fundamental do desempenho.
  - 8, 16, 32 ou 64 bits.



# Barramento de endereço

- Identifica origem ou destino dos dados.
- Por exemplo, a CPU precisa ler uma instrução (dados) de determinado local na memória.
- Largura do barramento determina capacidade máxima da memória do sistema.
  - Por exemplo, 8080 tem barramento de endereço de 16 bits gerando espaço de endereços de 64k.





# Barramento de controle

- Informação de controle e temporização:
  - Sinal de leitura/escrita de memória.
  - Solicitação de interrupção.
  - Sinais de clock.

# Ciclo de Clock

- Em um sistema computacional, a ordem na qual os sinais de controle são ativados é crítica, alguns sinais devem obrigatoriamente preceder outros, enquanto outros sinais podem ser ativados simultaneamente.
- E ainda, para garantir que haja tempo suficiente para a transmissão de informação através dos barramentos internos, em alguns casos deve ser observado um intervalo de tempo mínimo entre dois sinais
- É necessária a sincronia para que possamos garantir que a informação seja transmitida e chegue ao seu destino antes que outra informações sejam enviadas

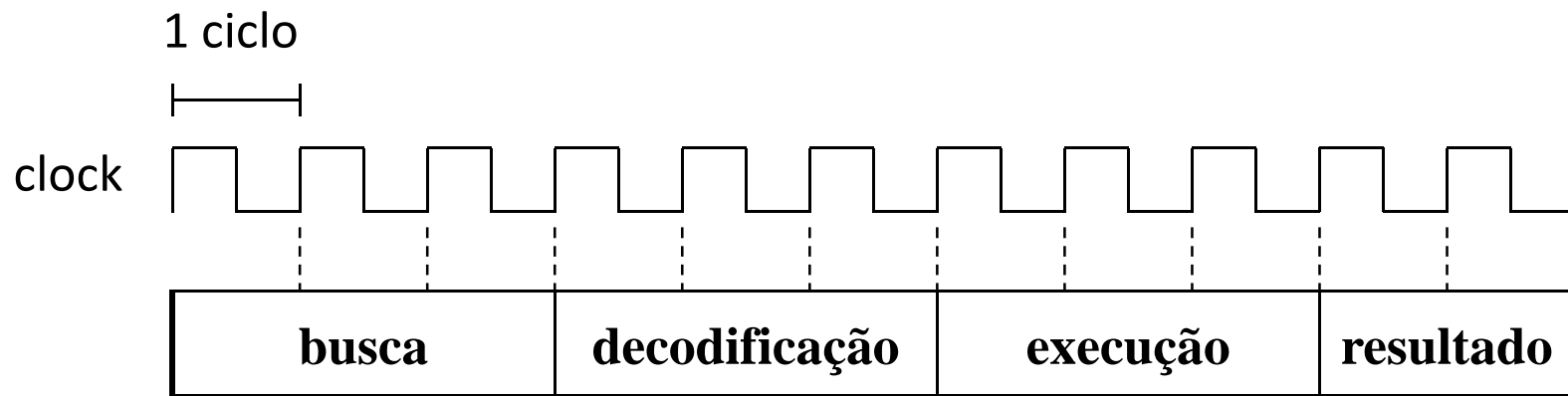
# Ciclo de Clock

- Um ciclo de clock completo é chamado de Hertz; como o tempo para que um ciclo seja completado é muito pequeno, normalmente medimos a velocidade em Mhz (Megahertz:  $10^6$ ) ou em Ghz (Gigahertz:  $10^9$ ) ciclos por segundo
- A velocidade de ciclos por segundo regulam o funcionamento da UCP; como as informações são transmitidas como sinais elétricos, é necessário um padrão de tempo para diferenciar uma informação da outra

# Ciclo de Clock

- Para atender as relações de tempo requeridas na ativação dos sinais de controle, a unidade de controle opera em sincronismo com um  sinal de clock .
  - Uma nova operação básica é executada no momento em que inicia-se um novo  ciclo de clock .
  - Em muitos casos, várias operações básicas podem ser comandadas simultaneamente, dentro de um mesmo ciclo de clock
- O intervalo entre duas transições consecutivas dos pulsos de clock é chamado de período de clock

# Ciclo de Clock



- A execução de uma instrução consome um certo número de ciclos de clock. O número de ciclos de clock por instrução não é o mesmo para todas as instruções, já que cada instrução pode envolver um número diferente de operações básicas em cada passo de execução



# Ciclo de Clock

- O tamanho do ciclo de clock é um dos fatores que determinam diretamente o desempenho de um processador. Quanto menor o tamanho do ciclo de clock, menor será o tempo de execução das instruções, e assim maior o número de instruções executadas por unidade de tempo.
- Ao longo das décadas de 70 e 80, procurava-se diminuir o tamanho do ciclo de clock com o desenvolvimento de novas tecnologias que permitissem velocidades de operação cada vez maiores. No entanto, as tecnologias de integração foram se aproximando dos limites impostos pela própria física, tornando esta evolução mais lenta e elevando os custos



# Ciclo de Clock

- Portanto, o ciclo de clock passou a ser considerado sob o ponto de vista arquitetural. Atualmente, procura-se diminuir o ciclo de clock não somente através de novas tecnologias, mas também através de simplificações na arquitetura, de modo que a arquitetura possa ser implementada através de circuitos mais simples e inerentemente mais rápidos



# Arquitetura de Entrada e Saída

- O sistema de entrada e saída de dados é o responsável pela ligação do sistema computacional com o mundo externo.
- Através de dispositivos de E/S que são fornecidos dados e instruções a serem aplicadas sobre estes e são retornados ao usuário os resultados de tais operações
- Periférico: é qualquer dispositivo ligado ao computador que permita a comunicação ou interação com o mundo externo. A partir do sentido do fluxo de dados em relação ao processador podemos classificá-lo como de entrada ou saída. Certos dispositivos podem ser dos dois tipos



# Entrada de Dados

- Entrada de dados: é feita através de dispositivos que convertem dados e informações em sinais que o computador é capaz de armazenar e processar
- Através dos dispositivos de entrada é possível fornecer ao computador instruções através de dispositivos interativos (teclado, mouse, tela sensível ao toque, microfone, etc.) ou dados através de disquetes, CDs, DVDs, scanners, leitores de código de barras, dentre vários outros

# Saída de Dados

- Saída de dados: através de dispositivos de saída o sistema computacional fornece ao usuário uma resposta para uma instrução obtida de um dispositivo de entrada
- As formas como estes dados são retornados são as mais diversas possíveis, as mais comuns são:
  - Na forma de palavras ou imagens através do monitor;
  - Para dispositivos de saída, como disquetes, CDs, pen drivers, com o propósito de armazenamento de dados;
  - Através de documentos impressos, utilizando dispositivos matriciais, de jato de tinta, laser ou cera

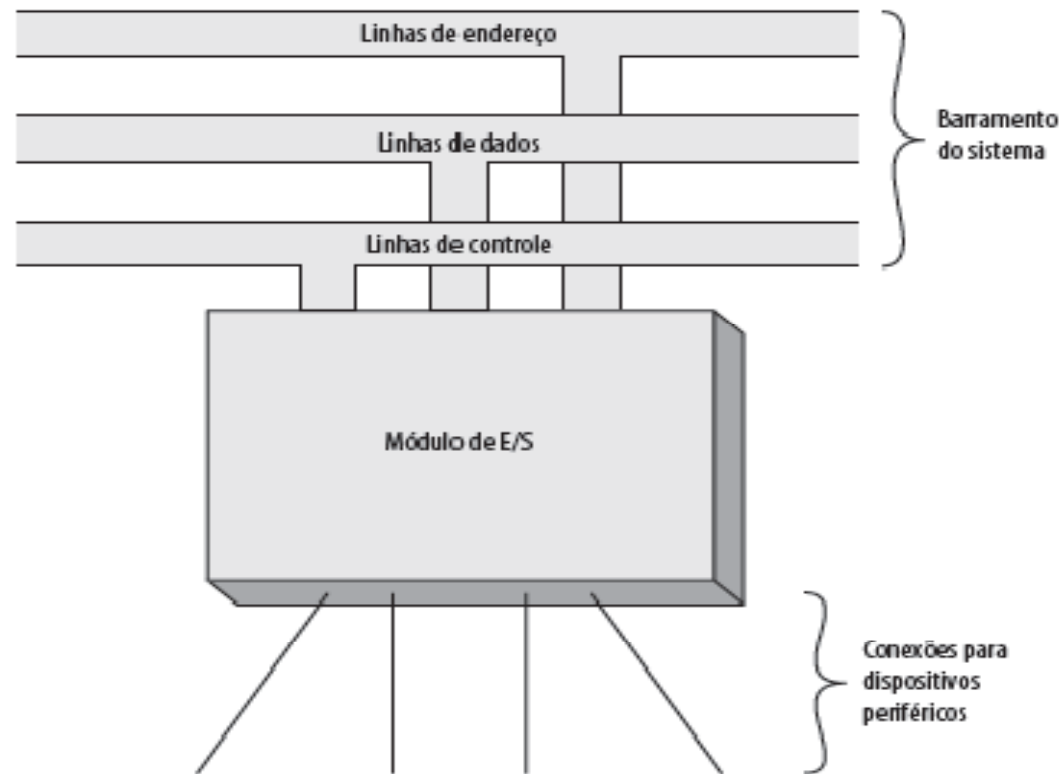


# Dispositivos de Entrada e Saída

- Por tratar-se de uma classe bastante heterogênea de dispositivos, os seguintes aspectos devem ser observados:
- Há uma grande variedade de periféricos:
  - Entregando diferentes quantidades de dados.
  - Em velocidades diferentes.
  - Em formatos diferentes.
- Todos mais lentos que CPU e RAM.
- Precisa de módulos de E/S.

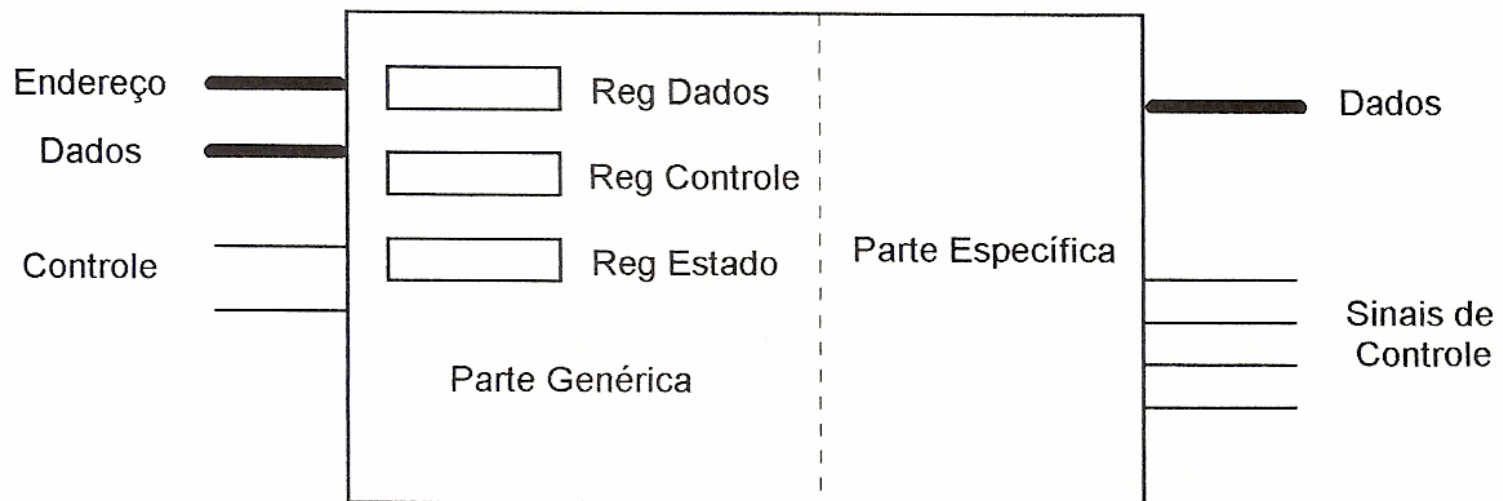
# Módulo de E/S

- Interface com CPU e memória.
- Interface com um ou mais periféricos.



# Módulo de E/S

- A principal função de uma interface de E/S é tornar transparente para a UCP os detalhes de operação e controle de dispositivos periféricos.
- Fundamentalmente, divide-se em duas partes:

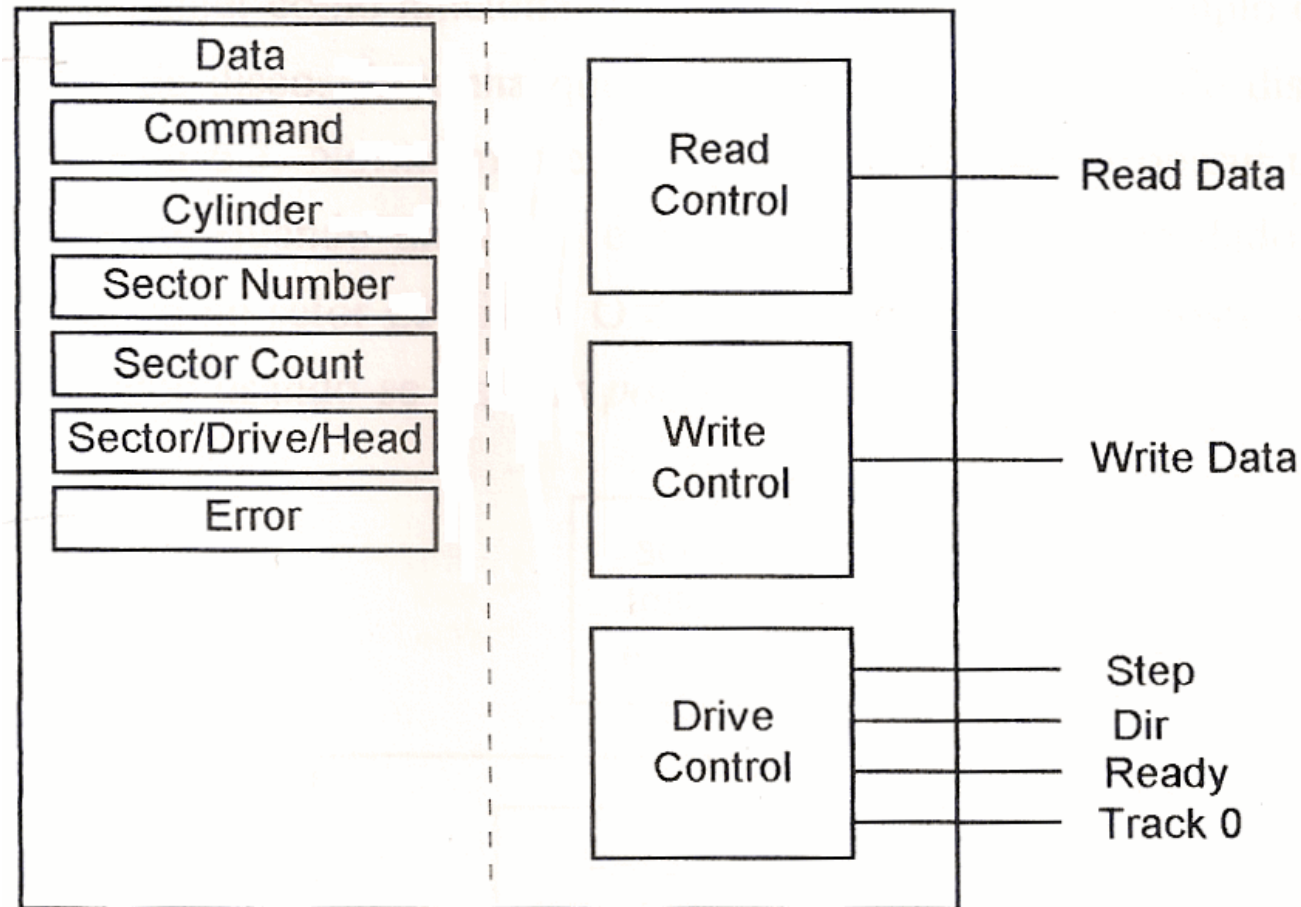




# Módulo de E/S

- A parte genérica é semelhante entre os diferentes tipos de interfaces de E/S, sendo a parte que interage com a UCP, destacando-se registradores específicos para cada tipo de dado enviado pelo barramento
- A parte específica interage diretamente com o periférico, e por isso ela difere bastante entre os vários tipos de interfaces.
- Porém, para qualquer periférico existe a parte de transmissão de dados entre a interface e o periférico, e também o conjunto de vias dos sinais de controle ao dispositivo

# Exemplo – Módulo de E/S de um disco rígido



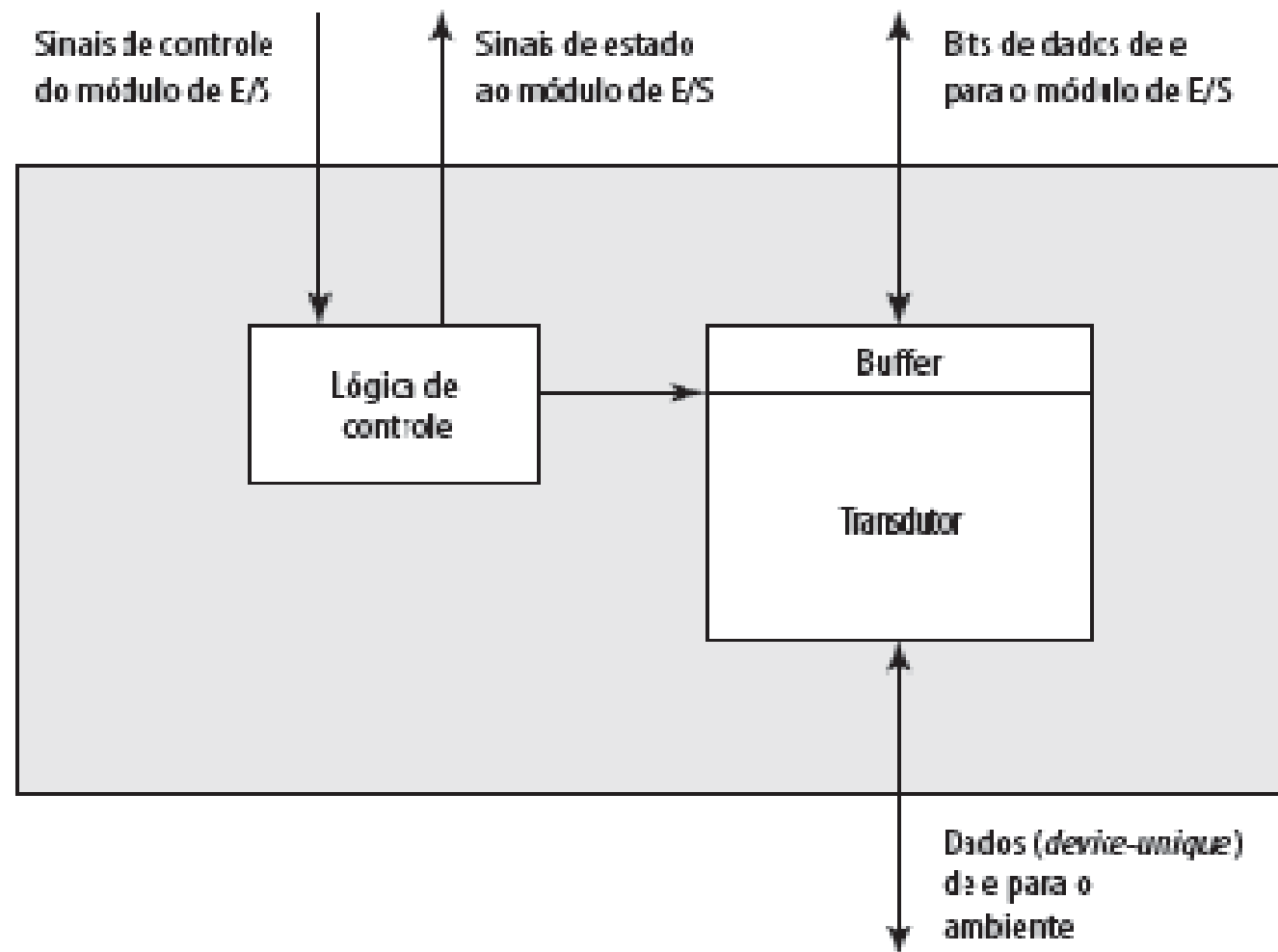


# Dispositivos Externos

- Um dispositivo externo deve pertencer a um dos seguintes grupos:
  - Legíveis ao ser humano:
    - Monitor, impressora, teclado.
  - Legíveis à máquina:
    - Monitoração e controle.
  - Comunicação (cabada e *wireless*):
    - Modem;
    - Placa de interface de rede (NIC).



# Dispositivos externos





# Funções do módulo de E/S

- Controle e temporização.
- Comunicação com CPU.
- Comunicação com dispositivo.
- *Buffering* de dados.
- Detecção de erro.



# Etapas da E/S

- CPU verifica estado do dispositivo do módulo de E/S.
- Módulo de E/S retorna o estado.
- Se estiver pronto, CPU solicita transferência de dados.
- Módulo de E/S recebe dados do dispositivo.
- Módulo de E/S transfere dados à CPU.
- Variações para saída, DMA etc.

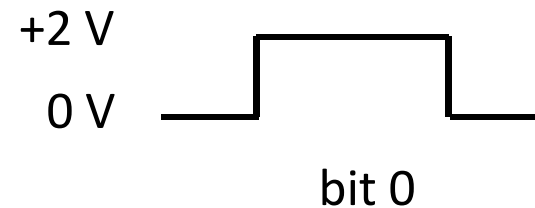
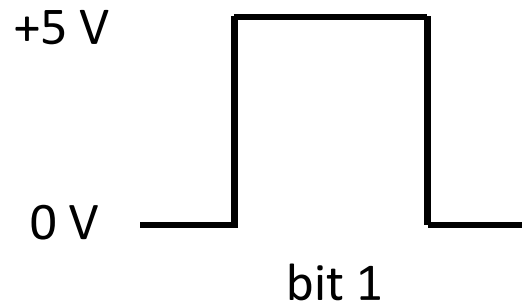
# Transmissão de Dados

- Um subsistema de entrada e saída de dados deve ser capaz de executar duas funções básicas:
  - Receber ou enviar informações para o meio exterior;
  - Converter as informações (de entrada ou saída) em uma forma inteligível para o computador (se estiver recebendo) ou para o usuário(se estiver enviando);
- É feita a conversão dos símbolos utilizados pelos seres humanos para representar informações para os “símbolos” usados pelo computador, que são apenas ‘0’ e ‘1’, representados através de intensidades de sinais elétricos

# Transmissão de Dados

A ; 9 ) #  
- A @ + 3

Símbolos utilizados pelo ser humano; são convertidos em ...



...símbolos utilizados na linguagem dos computadores

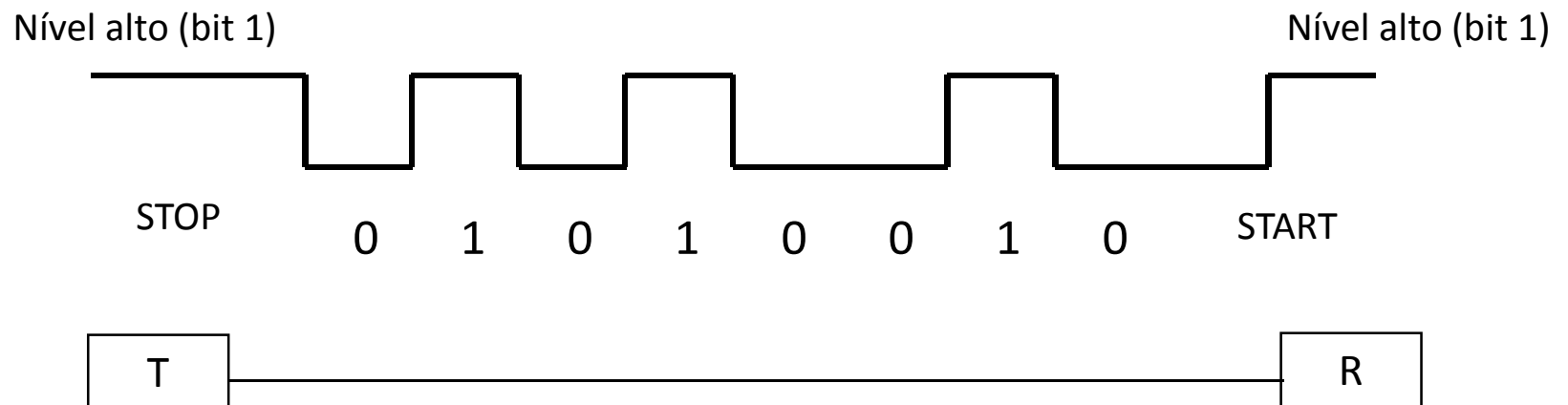
# Transmissão Serial

- Existem duas maneiras básicas de realizar a transmissão e recepção de dados entre periféricos/ interfaces e dispositivos de MP/UCP, as quais veremos a seguir
- Na **transmissão serial**, o periférico é conectado ao dispositivo controlador ou de E/S por uma única linha de transmissão
  - Como a transmissão é realizada bit a bit, receptor e transmissor devem estar sincronizados, ou seja, o transmissor transmite os bits sempre com a mesma velocidade, todos com a mesma duração no tempo
  - Para o receptor receber os dados, ele precisa saber quando um bit inicia e sua duração; o receptor deve trabalhar na mesma velocidade do transmissor

# Transmissão Serial – Assíncrona

- Apenas receber os bits não é suficiente, é necessário que o receptor saiba identificar grupos de bits que tenham um significado, por exemplo, um caracter; na transmissão serial duas formas podem ser utilizadas para a separação das informações: transmissão síncrona e assíncrona
- Na transmissão assíncrona, cada caracter é acrescido de um pulso de dois bits no início com valor de tensão correspondente ao bit 0, chamado START, e outro pulso de dois bits no final, com tensão correspondente ao bit 1, denominado STOP
- Enquanto não há transmissão, o transmissor envia continuamente bits 1 pela linha; assim que um caracter é enviado, é detectada uma queda de tensão pelo receptor

# Transmissão Serial – Assíncrona

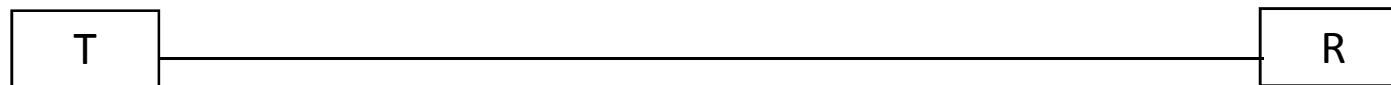




# Transmissão Serial – Síncrona

- A transmissão **síncrona** é mais eficiente, pois são transmitidos blocos de caracteres, sem intervalos entre eles e sem pulsos de START e STOP, reduzindo a quantidade de bits trafegados apenas para controle
  - Receptor e transmissor precisam trabalhar em sincronia para conhecerem o início e o final de cada bloco, funcionando com a mesma frequência de relógio
  - Neste modelo é necessário a utilização de identificadores de início e final de bloco; no início é inserido um grupo de bits, utilizado para marcar o início da contagem de bits a serem recebidos, e um grupo no final
  - Dispositivos USB trabalham utilizando transmissão síncrona

# Transmissão Serial – Síncrona



C<sub>1</sub>, C<sub>2</sub>, ... C<sub>n</sub> – caracteres de dados  
CC – caracteres especiais de controle

# Transmissão paralela

- Este modelo define um conjunto de sinais que fluem através de um conjunto de linhas de conexão simultaneamente; a comunicação de todos os circuitos internos do computador adota este modelo
- É necessário existir uma sincronia entre transmissor e receptor, realizada através de um pulso de frequência do sistema



# Transmissão paralela

- Apesar do modelo de transmissão paralela ser conceitualmente melhor do que a transmissão serial, existem certos fatores que devem ser levados em consideração na sua adoção:
  - Interfaces paralelas requerem maior número de fios, o que leva a cabos mais caros e largos, além de conectores com maior número de pinos;
  - Um cabo com mais vias necessita de blindagem para evitar interferências elétricas entre os fios;
  - As interfaces paralelas requerem a sincronização entre os fios, o que se torna mais grave à medida que são utilizados cabos mais longos.

# Controle de E/S programada

- As técnicas de utilizadas em operações de Entrada e Saída dividem-se em três categorias:
- E/S programada: as operações de E/S são realizadas diretamente e de forma contínua pelo programa que requisitou a operação; quando o processador envia um comando para o módulo de E/S, ele tem que esperar até que a operação seja completada, e no caso do processador ser mais rápido que o módulo de E/S, essa espera representará um desperdício no tempo de processamento; durante todo o tempo é testado o estado do módulo de E/S, degradando substancialmente o desempenho do sistema

# Controle de E/S por interrupção

- E/S dirigida por interrupção: a comunicação utilizando a técnica de interrupção é realizada da seguinte forma:
  - A UCP emite a instrução de E/S para a interface, e não havendo uma resposta imediata, ao invés de ficar checando continuamente o estado do dispositivo, vai executar outra atividade, normalmente, de outro programa;
  - Quando a interface está pronta para enviar os dados, ela avisa a UPC através de um sinal especial chamado de interrupção, que interrompe a atividade corrente da UCP, para ser utilizada pelo dispositivo que disparou o sinal;
  - A UCP inicia então o programa de E/S.



# Controle de E/S por interrupção

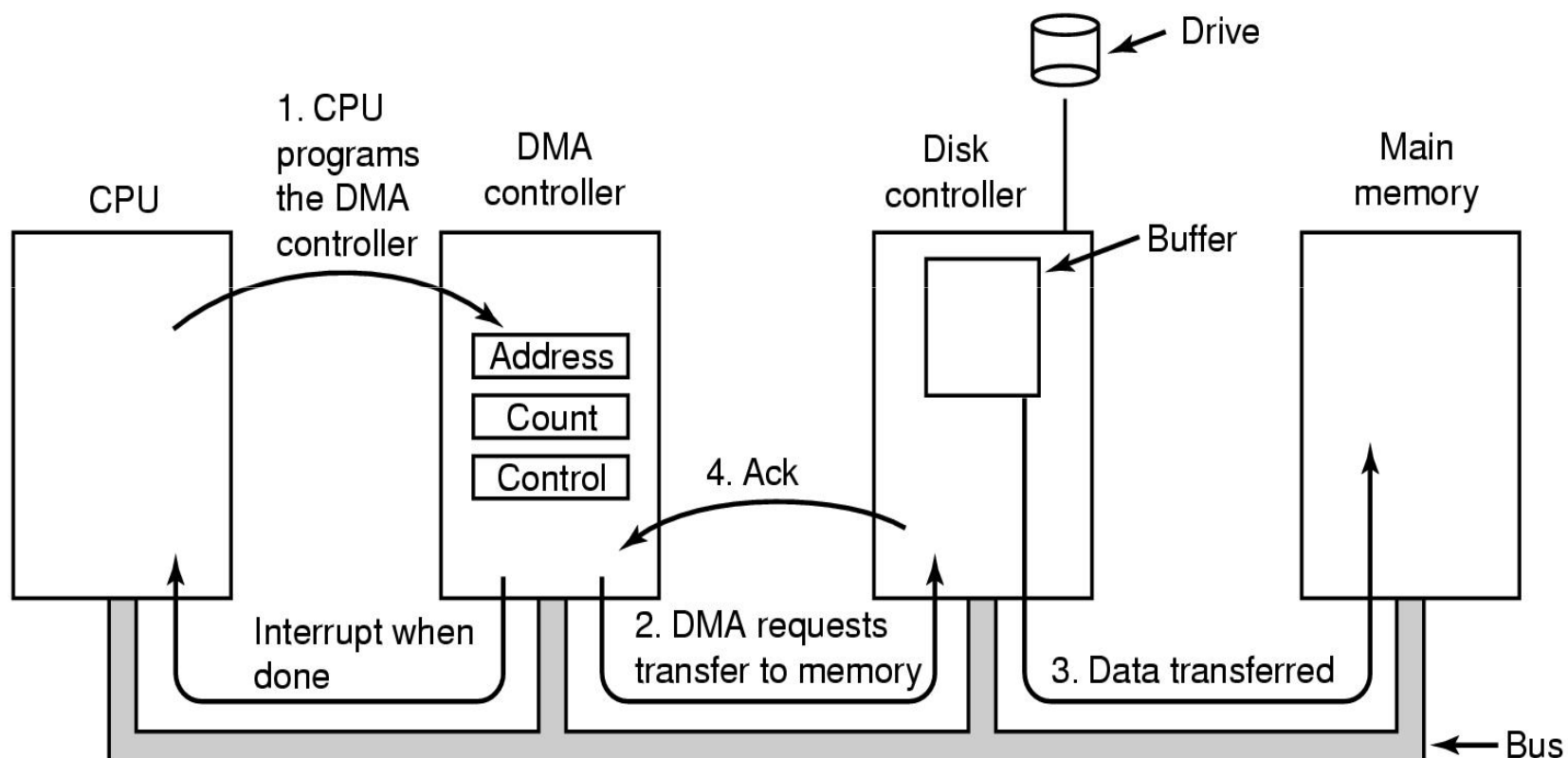
- O problema deste método é que toda vez que um programa é interrompido, é necessário salvar seu contexto, isto é, seu estado no momento da interrupção, para que posteriormente, quando for dada continuidade a sua execução, recomece de onde parou, e não desde o início

# Controle de E/S por DMA

- Acesso direto a memória (DMA): a E/S é controlada por um módulo especializado de E/S, chamado de controlador de DMA, que se encarrega da transferência dos blocos de dados diretamente de uma interface para a memória através de um barramento do sistema
- A UCP fica liberada para realizar outras atividades, apenas solicita ao dispositivo DMA que controle o tráfego de dados através do barramento; quando o controlador termina a transferência, sinaliza para a UCP através de uma interrupção



# Exemplo de operação de E/S com DMA



# Fluxogramas das técnicas de controle de E/S

