

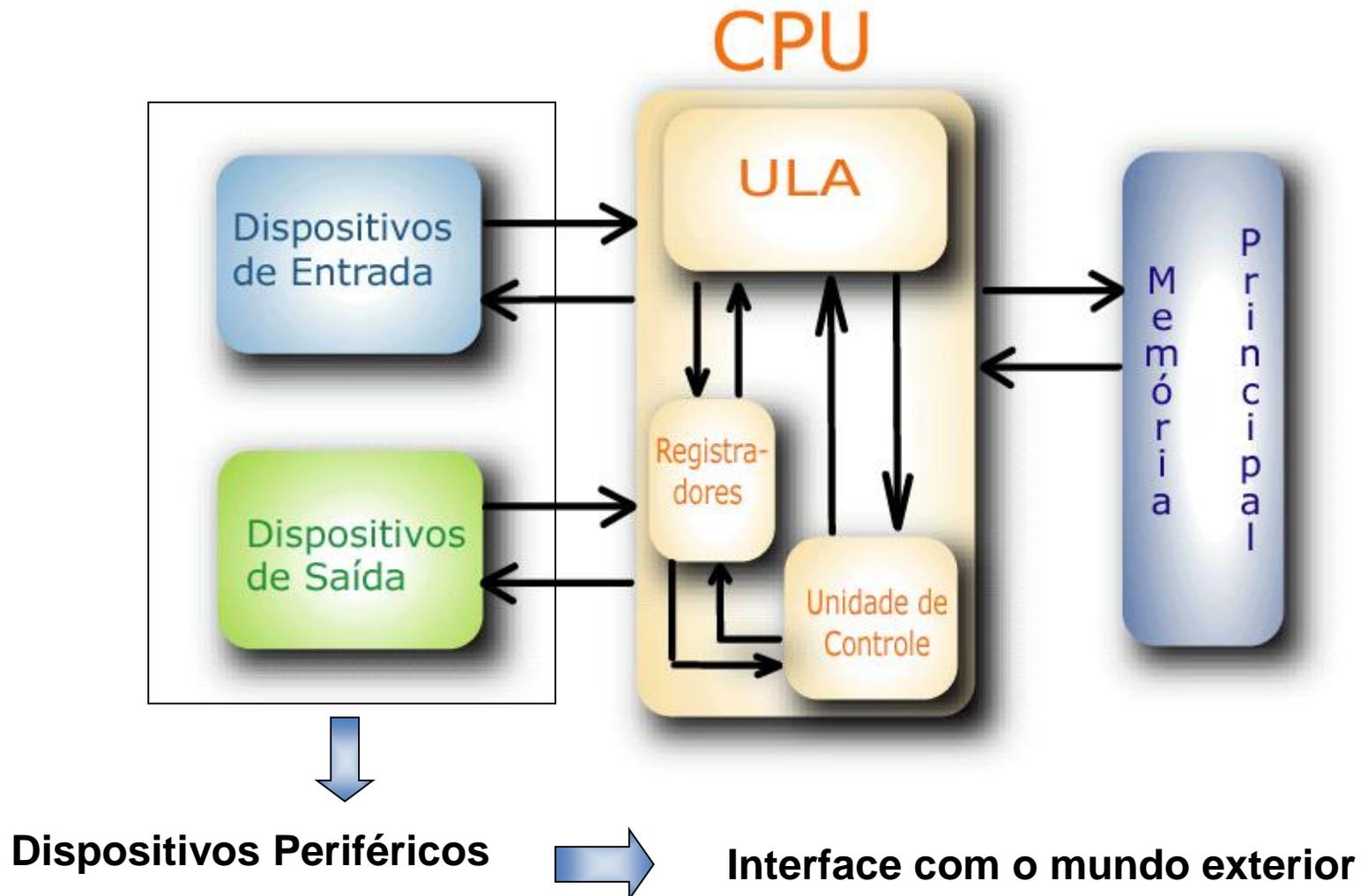
# SSC0611

# Arquitetura de Computadores

6ª Aula – Entrada e Saída

Profa. Sarita Mazzini Bruschi  
[sarita@icmc.usp.br](mailto:sarita@icmc.usp.br)

# Estrutura da máquina de von Neumann



# Dispositivos Periféricos

- Os dispositivos podem ser caracterizados por:
  - Comportamento: Entrada (teclado, mouse) / Saída (monitor, impressora) / Armazenamento (disco, memória flash)
  - Relacionamento: com o ser humano (teclado, vídeo, impressora) ou com o computador (discos, sensores, interface de rede)
  - Taxa de transferência: bytes/seg ; transferências/seg

# Dispositivos Periféricos

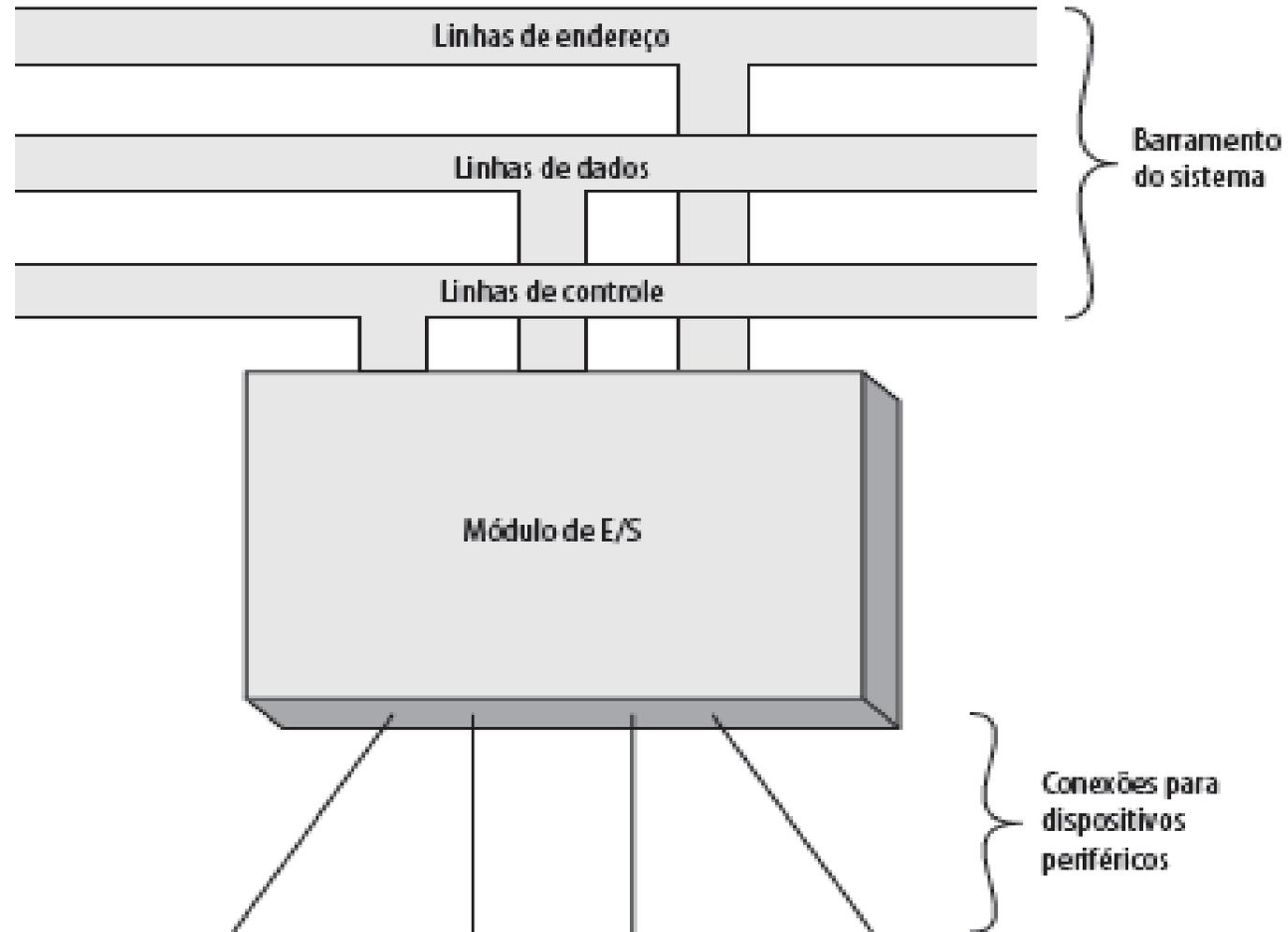
- Como conectar os dispositivos periféricos com a CPU?
  - Manipulação de diferentes quantidades de dados
  - Taxa de transferência do periférico menor que CPU ou RAM
  - Formatos de dados e tamanhos de palavras diferentes

**Necessidade de um Módulo (ou Controlador) de E/S**

# Módulos de E/S

- Os módulos de E/S são o terceiro elemento de um sistema de computação (os outros são memória e CPU)
- Comunicação entre diversos periféricos e o barramento do sistema
- Cada dispositivo externo é conectado ao computador através de uma conexão a um módulo de E/S

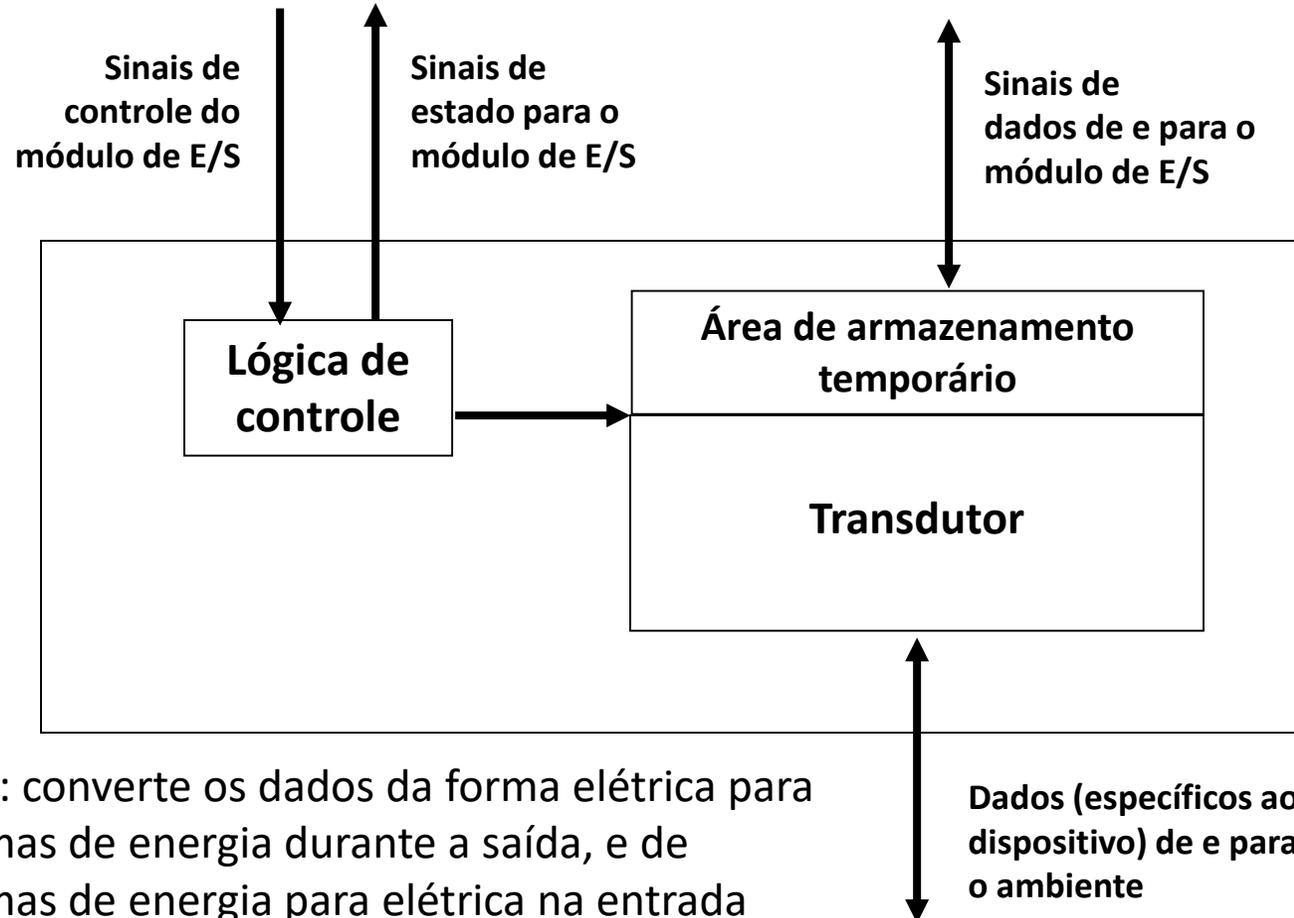
# Módulos de E/S



# Dispositivos Externos

- Os dispositivos externos fazem interface com o módulo de E/S através de sinais de controle, dados e estado
  - Controle: função a ser executada pelo dispositivo (INPUT ou READ, OUTPUT ou WRITE)
  - Estado: estado do dispositivo (READY/NOT READY)
  - Dados: conjunto de bits a serem enviados ou recebidos do módulo de E/S

# Dispositivos Externos



Transdutor: converte os dados da forma elétrica para outras formas de energia durante a saída, e de outras formas de energia para elétrica na entrada

# Módulos de E/S

- O módulo de E/S:
  - Faz a interface com o processador e a memória
  - Faz a interface com um ou mais dispositivos periféricos, através de conexões adequadas
  - Livra a CPU do gerenciamento dos dispositivos periféricos de E/S

# Módulos de E/S

- Funções:
  - Controle e temporização
  - Comunicação com o processador
  - Comunicação com dispositivos
  - Área de armazenamento temporário de dados
  - Detecção de erros

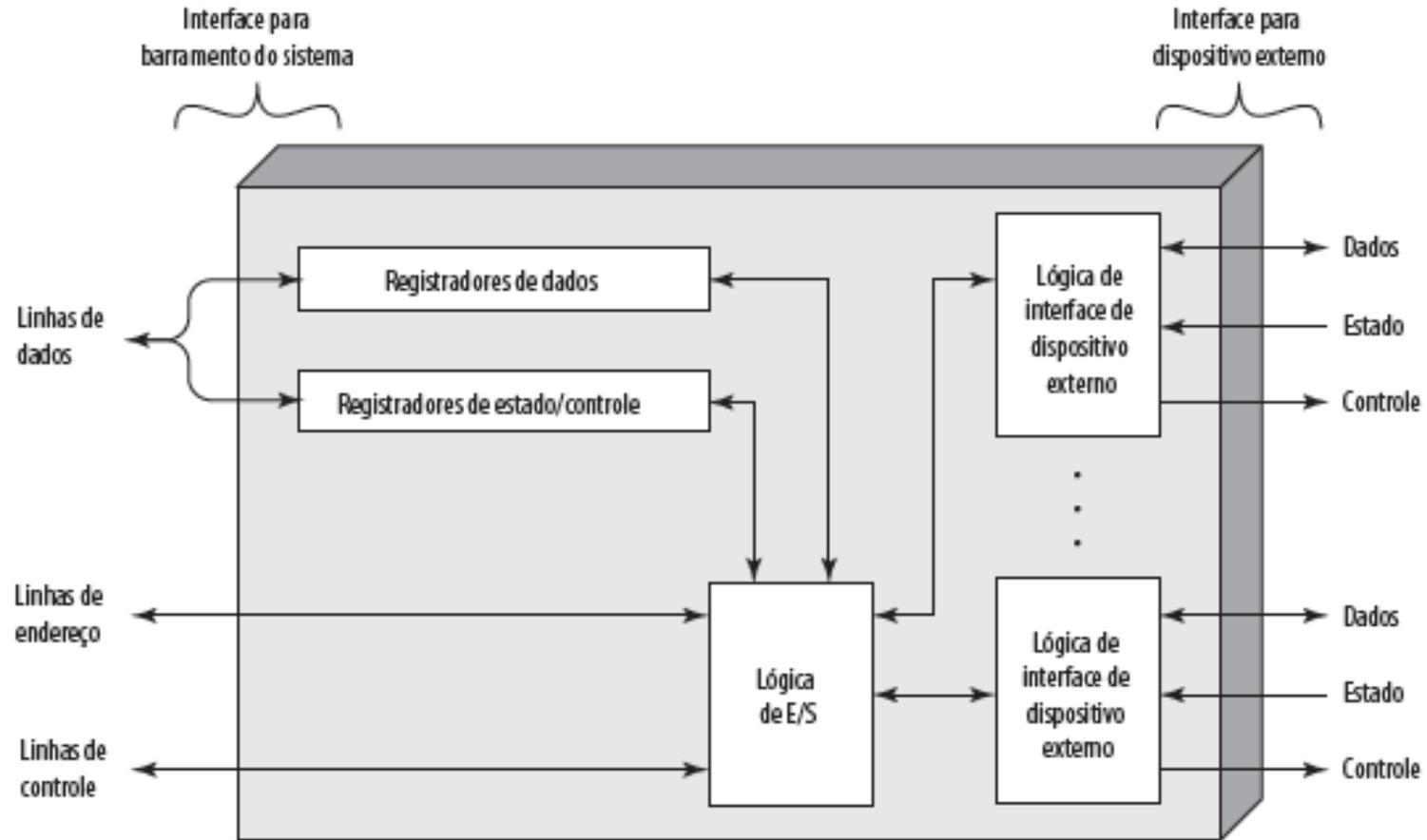
# Módulos de E/S

## Estrutura

- Complexidade de sua estrutura varia de acordo com a quantidade e dispositivos que é capaz de controlar
- Exemplo: módulo de E/S Intel 82C55A
  - Módulo conectado ao restante do sistema através de linhas de sinais
  - Dados transferidos são armazenados temporariamente em registradores de dados
  - Registradores de estado (armazenam estado atual dos dispositivos). Esses mesmos registradores funcionam como registradores de controle para a CPU (para CPU enviar comandos)

# Módulo de E/S

## Estrutura do Intel 82C55A



# Operações de E/S

- No início CPU controlava diretamente o dispositivo (*microprocessor-controlled devices*)
- Módulo de E/S (ou controlador) é adicionado
  - Assim CPU não precisa conhecer detalhes específicos dos dispositivos
  - CPU usa E/S programada sem interrupção
- Surge então a interrupção
  - CPU trabalha enquanto ocorre E/S
- O módulo de E/S ganha poder para acessar memória diretamente (DMA)

# Operações de E/S

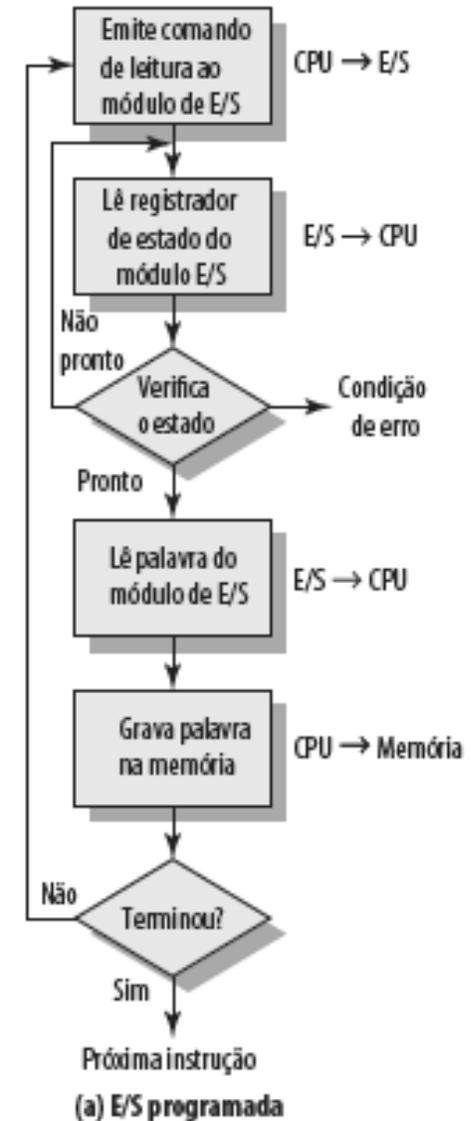
- As operações de E/S podem ser realizadas
  - por meio do processador ou diretamente entre a memória e o módulo; e
  - com interrupção ou não
- Três técnicas:
  - E/S programada (processador, sem interrupção)
  - E/S dirigida por interrupção (processador, com interrupção)
  - Acesso direto à memória (DMA – *Direct Memory Access*) (diretamente com interrupção)

# E/S Programada

- Dados são trocados entre o processador e o módulo de E/S
  - Há um programa, executado pelo processador, para manipular todas as operações de E/S
- O processador é mais rápido que o módulo de E/S, então perde-se muito tempo, pois processador espera pela operação de E/S

# E/S Programada

- CPU solicita operação de E/S
- Módulo de E/S realiza operação
- Módulo de E/S define bits de estado
- CPU verifica bits de estado periodicamente
- Módulo de E/S não informa à CPU diretamente
- Módulo de E/S não interrompe CPU
- CPU pode esperar ou voltar mais tarde



# E/S Programada

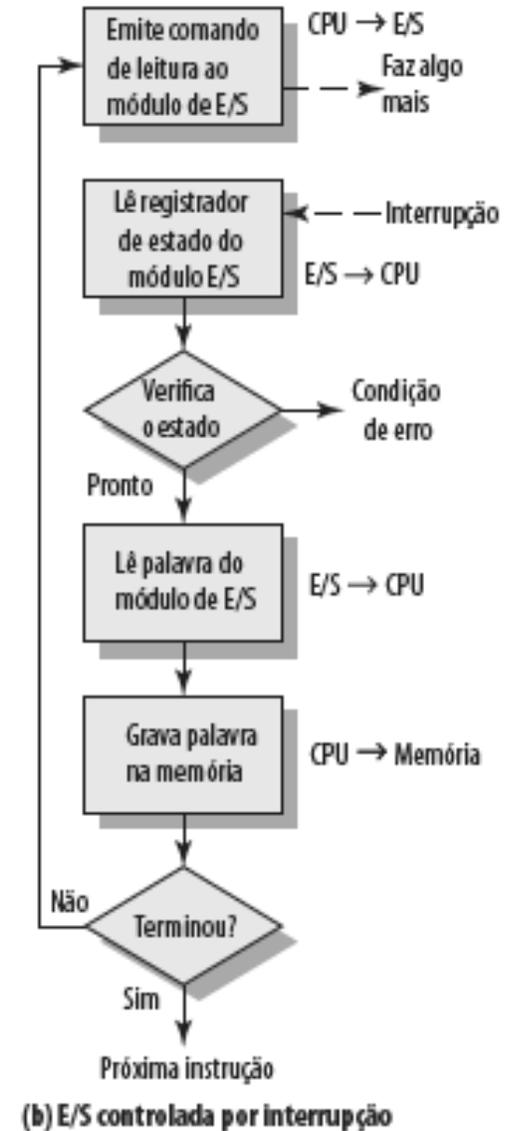
- Desvantagem: CPU espera pelo dispositivo
  - Dispositivos lentos: perda de tempo de processamento
  - Se a CPU iria ficar parada: não tem problema
  - Caso contrário, deve-se optar por outra forma de E/S
- Para evitar que o sistema fique travado em casos de falha nos dispositivos de E/S pode implementar um sistema de monitoração por um temporizador

# E/S Dirigida por Interrupção

- Objetivo: reduzir o tempo gasto nas operações de E/S
- Inicialização da transferência vem do dispositivo
- Interrupção: programa em execução é interrompido para execução de uma tarefa mais urgente!

# E/S Dirigida por Interrupção

- CPU emite comando de leitura
- Módulo de E/S recebe dados do periférico enquanto CPU faz outro trabalho
- Módulo de E/S interrompe CPU
- CPU solicita dados
- Módulo de E/S transfere dados



# E/S com Acesso Direto à Memória (DMA)

- Desvantagens da E/S programada e E/S dirigida por interrupção
  - Taxa de transferência de E/S é limitada pela velocidade do processador
  - Processador fica sobrecarregado

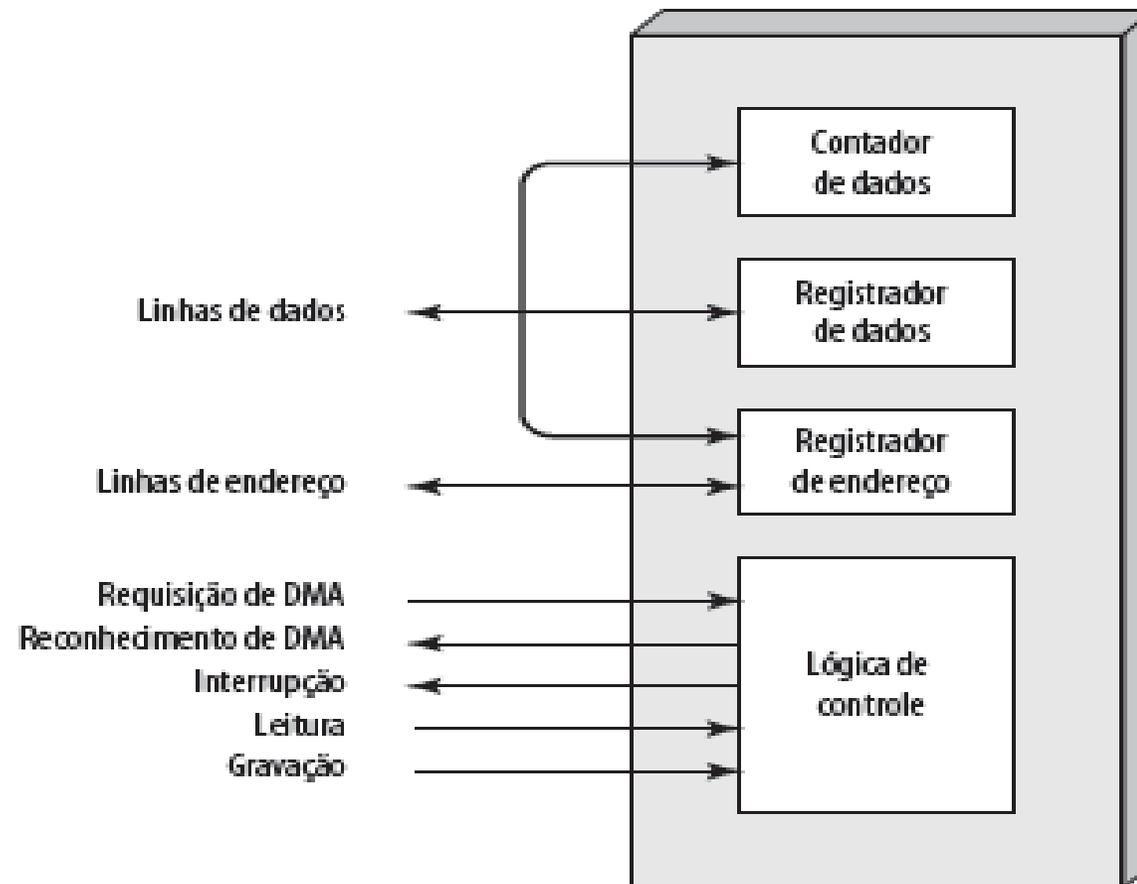
# E/S com Acesso Direto à Memória (DMA)

- Dados trocados entre Memória e dispositivos não precisam passar pela CPU
  - Ex: Memória Principal e Disco
- Com quantidade de dados elevada tem-se ganho de desempenho com DMA
- Exige lógica adicional razoavelmente complexa: processador dedicado à DMA que substitui a CPU nas operações de E/S com DMA
- Módulo adicional (hardware) conectado ao barramento



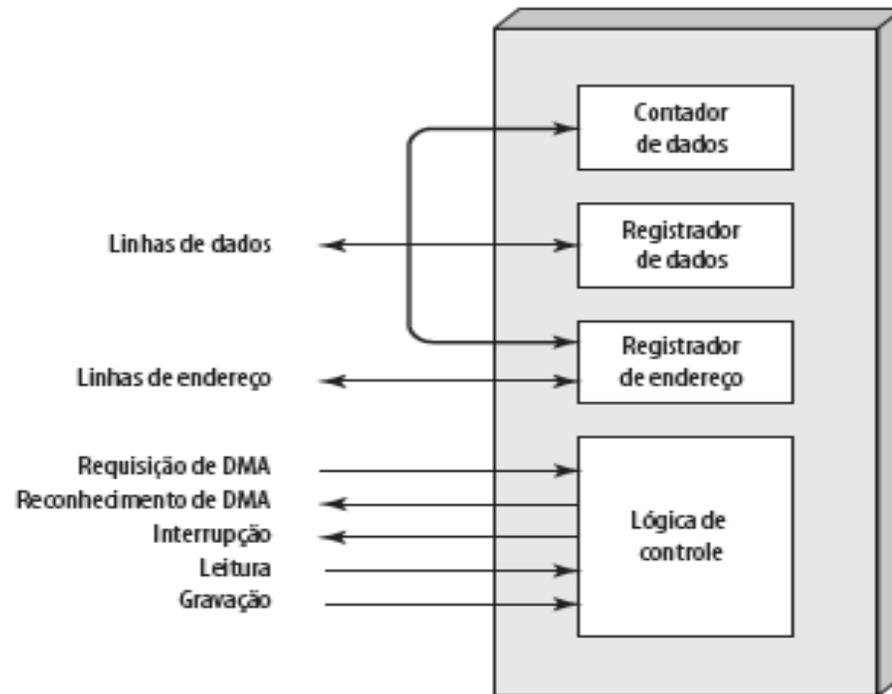
# E/S com Acesso Direto à Memória (DMA)

- Controlador de DMA ou Módulo de DMA



# E/S com Acesso Direto à Memória (DMA)

- Controlador de DMA ou Módulo de DMA

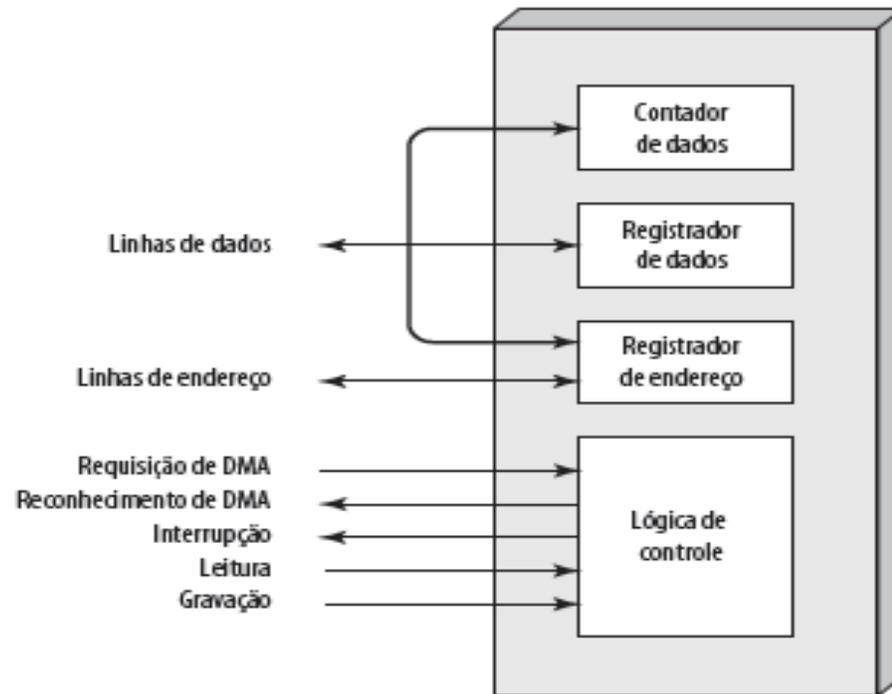


Processador envia:

- indicação de operação de E/S pela linha de Leitura ou Escrita
- endereço do dispositivo de E/S pelas linhas de endereço
- Endereço inicial de memória pela linha de dados -> Registrador de Endereço
- Número de palavras a serem lidas /escritas pela linha de dados -> Contador de Dados

# E/S com Acesso Direto à Memória (DMA)

- Controlador de DMA ou Módulo de DMA



DMA:

- Transfere o bloco de dados, uma palavra por vez, diretamente, de ou para a memória principal
- Sinal de Interrupção

# E/S com Acesso Direto à Memória (DMA)

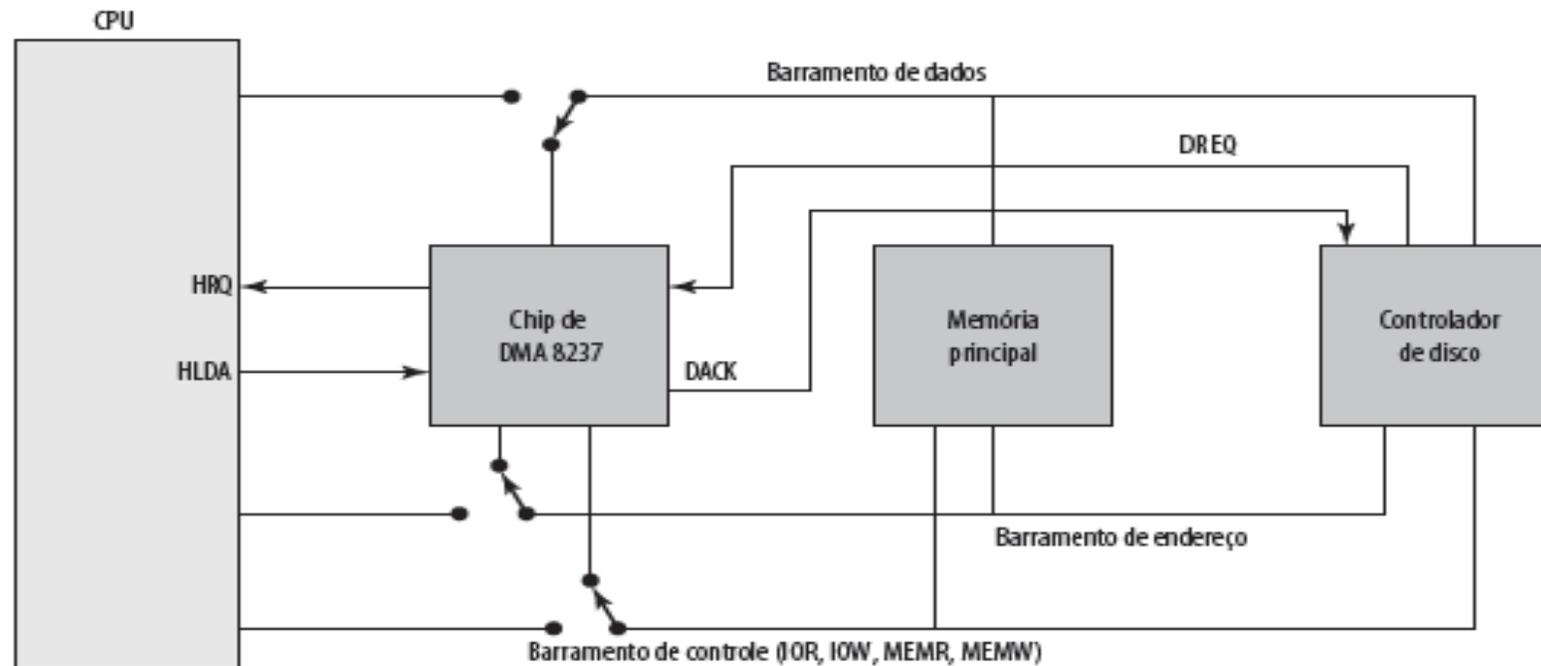
- CPU programa a DMA para transferir bloco de dados
  - CPU continua processamento
  - DMA executa transferência
- DMA encerra transferência
  - Envia interrupção ao processador
  - CPU lê estado do Módulo de DMA
- Processador executa próxima instrução

# E/S com Acesso Direto à Memória (DMA) - Exemplo

- Controlador de DMA Intel 8237A
  - Interfaces com família 80x86 e DRAM
  - Quando o módulo de DMA precisa de barramentos, ele envia sinal HOLD ao processador
  - CPU responde HLDA (*hold acknowledge*)
  - Módulo de DMA pode usar barramentos

# E/S com Acesso Direto à Memória (DMA) - Exemplo

- Ex: transferir dados da memória para o disco



DACK = DMA *acknowledge* (reconhecimento de DMA)  
DREQ = DMA *request* (requisição de DMA)  
HLDA = HOLD *acknowledge* (reconhecimento de HOLD)  
HRQ = HOLD *request* (requisição de HOLD)

# E/S com Acesso Direto à Memória (DMA) - Exemplo

1. Dispositivo requisita serviço de DMA levantando DREQ (requisição de DMA)
2. DMA levanta sua linha HRQ (hold request)
3. CPU termina ciclo de barramento e levanta linha HDLA (hold acknowledge). HOLD permanece ativo pela duração do DMA

# E/S com Acesso Direto à Memória (DMA) - Exemplo

4. DMA ativa DACK (DMA acknowledge), para dispositivo iniciar transferência.
5. DMA inicia transferência:
  - endereço do primeiro byte no barramento de endereço
  - ativa MEMR
  - ativa IOW para escrever no periférico.

# E/S com Acesso Direto à Memória (DMA) - Exemplo

6. DMA decrementa contador e incrementa ponteiro de endereço. Repete até contagem chegar a zero.
7. DMA desativa HRQ, retornando o controle do barramento de volta à CPU.