

Dispositivos de Entrada e Saída

Profa. Débora Christina Muchaluat Saade
debora@midia.com.uff.br

<http://www.midia.com.uff.br/debora>

1

Problemas com Entrada e Saída

- ✓ **Periféricos possuem características diferentes**
 - *Geram diferentes quantidades de dados*
 - *Em velocidades diferentes*
 - *Em formatos diferentes*
- ✓ **Periféricos são mais lentos que UCP e Memória**
- ✓ **Necessita-se de módulos de Entrada/Saída**

2

Dispositivos de E/S

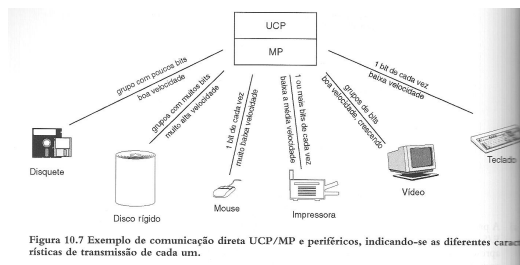


Figura 10.7 Exemplo de comunicação direta UCP/MP e periféricos, indicando-se as diferentes características de transmissão de cada um.

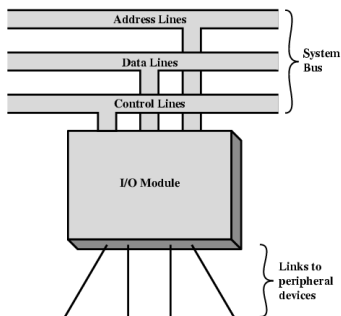
3

Módulo de Entrada/Saída

- ✓ **Interface para UCP e memória**
- ✓ **Interface para um ou mais periféricos**

4

Modelo Genérico de um Módulo de E/S



5

Estrutura do sistema

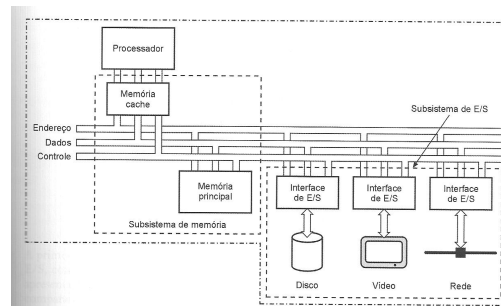


Figura 10.3 Modelo de estrutura de um sistema de computação.

6

Funções do Módulo de E/S

Fundamentos de Arquiteturas de Computadores

- ✓ Controle & Temporização
- ✓ Comunicação com UCP
- ✓ Comunicação com dispositivo
- ✓ Bufferização de dados
- ✓ Detecção de erros

7

Operação de E/S

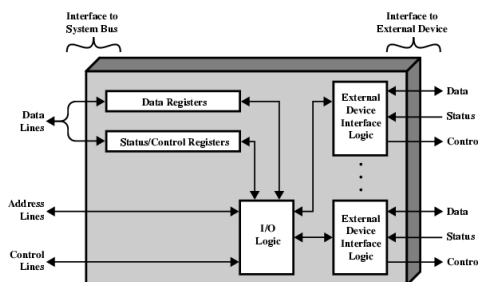
Fundamentos de Arquiteturas de Computadores

- ✓ UCP solicita estado do dispositivo para módulo de E/S
- ✓ Módulo de E/S retorna estado
- ✓ Caso dispositivo pronto, UCP solicita transferência de dados
- ✓ Módulo de E/S obtém dados do dispositivo
- ✓ Módulo de E/S transfere dados para UCP

8

Diagrama do Módulo de E/S

Fundamentos de Arquiteturas de Computadores



9

Decisões do Módulo de E/S

Fundamentos de Arquiteturas de Computadores

- ✓ Esconder ou revelar propriedades do dispositivo para UCP
- ✓ Suportar um ou múltiplos dispositivos
- ✓ Controlar funções do dispositivo ou deixar para UCP

11

Comandos de E/S

Fundamentos de Arquiteturas de Computadores

- ✓ Envio de endereço
 - *Identifica módulo (endereço do dispositivo, caso exista mais de um dispositivo por módulo)*
- ✓ Envio de comando
 - *Controle – indica ao módulo o que fazer*
 - Desloca cabeça de leitura e gravação
 - *Teste – verifica estado do dispositivo*
 - Ligou? Erro?
 - *Leitura/escrita*
 - O módulo transfere dados via buffer de/para dispositivo

12

Mapeamento de E/S

Fundamentos de Arquiteturas de Computadores

- ✓ E/S mapeada na memória (*memory-mapped*)
 - *Dispositivos e memória compartilham espaço de endereçamento*
 - *Operações de escrita/leitura para E/S são executadas da mesma forma que para a memória*
 - *Não existem comando especiais de E/S*
 - Todos os comandos de acesso à memória podem ser utilizados para E/S
- ✓ E/S isolada (*isolated I/O*)
 - *Espaços de endereçamento separados (portas de E/S)*
 - *Necessita de linhas diferentes para selecionar memória e E/S*
 - *Comandos especiais de E/S*
 - Conjunto limitado (*in e out*)

Memória mapeada x isolada

Fundamentos de Arquiteturas de Computadores

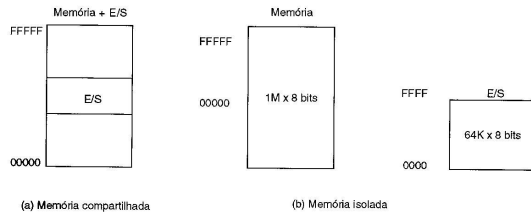


Figura 10.31 Exemplo de organização de memória nos microprocessadores Intel 8086/8088.

14

Técnicas de Entrada e Saída

Fundamentos de Arquiteturas de Computadores

- ✓ Programada
- ✓ Por interrupção
- ✓ Acesso Direto à Memória (DMA – *Direct Memory Access*)

15

E/S Programada

Fundamentos de Arquiteturas de Computadores

- ✓ UCP controla diretamente o dispositivo de E/S
 - *Verifica estado*
 - *Comandos de escrita/leitura*
 - *Transfere dados*
- ✓ UCP espera pela finalização da operação do módulo de E/S
- ✓ Gasta tempo de processamento da UCP
 - *Método ineficiente*

16

E/S Programada

Fundamentos de Arquiteturas de Computadores

- ✓ UCP solicita operação de E/S
- ✓ Módulo de E/S realiza a operação
- ✓ Módulo de E/S seta bits de estado
- ✓ UCP verifica bits de status periodicamente
 - *polling ou interrogação*
- ✓ Módulo de E/S não informa diretamente à UCP
- ✓ Módulo de E/S não interrompe a UCP
- ✓ UCP pode esperar ou voltar mais tarde

17

E/S por programa

Fundamentos de Arquiteturas de Computadores

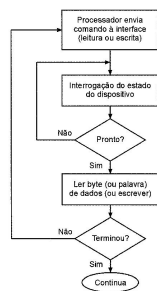
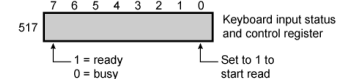


Figura 10.30 Fluxograma de operação de E/S (método E/S por programa).

18

Memória Mapeada

Fundamentos de Arquiteturas de Computadores



| Endereço | Instrução |
|----------|------------|
| 200 | add 0 1 1 |
| 201 | sw 0 1 517 |
| 202 | lw 0 2 517 |
| 203 | beq 2 0 -2 |
| 204 | lw 0 3 516 |

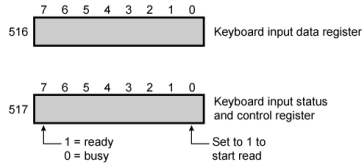
Comentário

- Carrega registrador 1 com comando 1
- Envia comando para ler teclado
- Carrega estado do teclado em registrador 2
- Fica em loop até teclado estar pronto
- Carrega dado do teclado em registrador 3

19

E/S Isolada

Fundamentos de Arquiteturas de Computadores



| Endereço | Instrução | Comentário |
|----------|------------|--|
| 200 | add 0 1 1 | Carrega registrador 1 com comando 1 |
| 201 | out 1 517 | Envia comando para ler teclado |
| 202 | in 2 517 | Carrega estado do teclado em registrador 2 |
| 203 | beq 2 0 -2 | Fica em loop até teclado estar pronto |
| 204 | in 3 516 | Carrega dado do teclado em registrador 3 |

20

E/S Dirigida por Interrupção

Fundamentos de Arquiteturas de Computadores

- ✓ Libera espera de UCP
- ✓ UCP não precisa ficar verificando estado do dispositivo repetidamente
- ✓ Módulo de E/S interrompe a UCP quando estiver pronto

21

E/S por interrupção

Fundamentos de Arquiteturas de Computadores

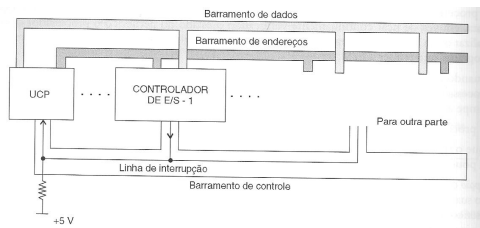


Figura 10.33 Tratamento de uma interrupção (elementos que participam do processo).

22

Operação Básica para Realização de E/S por Interrupção

Fundamentos de Arquiteturas de Computadores

- ✓ UCP envia comando de leitura
- ✓ Módulo de E/S obtém dado do periférico enquanto a UCP executa outro trabalho
- ✓ Módulo de E/S interrompe a UCP
- ✓ UCP pede dados para o módulo de E/S
- ✓ Módulo de E/S transfere dados para UCP

23

Tratamento de interrupção

Fundamentos de Arquiteturas de Computadores

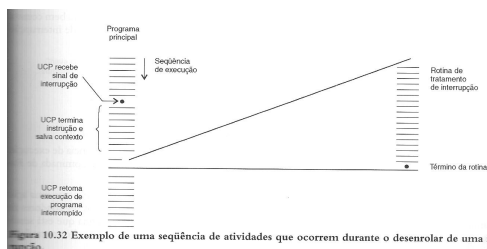
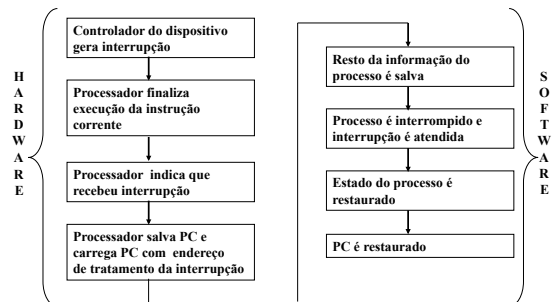


Figura 10.32 Exemplo de uma sequência de atividades que ocorrem durante o desenvolver de uma interrupção.

24

Processamento da Interrupção

Fundamentos de Arquiteturas de Computadores



25

Atividades da UCP

Fundamentos de Arquiteturas de Computadores

- ✓ Envia comando de leitura
- ✓ Executa outra tarefa
- ✓ Verifica se existe interrupção ao final de cada instrução
- ✓ Caso exista interrupção:
 - *Salva contexto (registradores)*
 - *Interrompe processo*
 - Obtém dados do módulo de E/S e os armazena

26

Questões de Projeto

Fundamentos de Arquiteturas de Computadores

- ✓ Como identificar o módulo que gerou a interrupção?
- ✓ Como gerenciar muitas interrupções?
 - *Qual delas atender?*

27

Identificação do Módulo que Gera a Interrupção (1)

Fundamentos de Arquiteturas de Computadores

- ✓ Uma linha diferente para cada módulo
 - *Limita número de dispositivos porque número de linhas no barramento é limitado*
- ✓ Identificação por software
 - *Uma única linha de interrupção*
 - *UCP interroga um módulo de cada vez para verificar se ele gerou a interrupção*
 - *Lento*

28

Identificação do Módulo que Gera a Interrupção (2)

Fundamentos de Arquiteturas de Computadores

- ✓ Arbitragem do barramento
 - *Módulo precisa obter o controle do barramento e depois envia sinal de interrupção*
 - *UCP envia sinal de reconhecimento e módulo coloca o vetor de interrupção nas linhas de dados*
 - *PCI & SCSI*

29

Múltiplas Interrupções

Fundamentos de Arquiteturas de Computadores

- ✓ Com mais de uma linha de interrupção, cada linha de interrupção possui uma prioridade
- ✓ Linhas com prioridade maior podem interromper linhas com prioridade menor
- ✓ Esquema de prioridades para arbitragem de barramento

30

Exemplo – Barramento PC

Fundamentos de Arquiteturas de Computadores

- ✓ 80x86 possui uma linha de interrupção
- ✓ Controladora de interrupções 8259A
- ✓ 8259A possui 8 linhas de interrupção

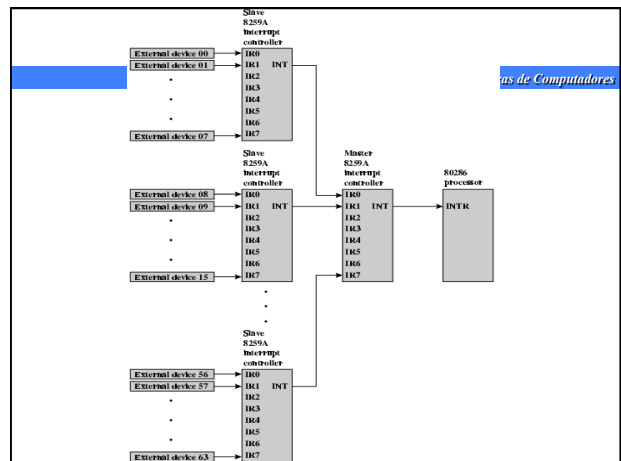
31

Sequência de Eventos

Fundamentos de Arquiteturas de Computadores

- ✓ 8259A aceita interrupções
- ✓ 8259A determina prioridade
- ✓ 8259A sinaliza interrupção para 8086 (levanta linha INTR)
- ✓ UCP reconhece interrupção
- ✓ 8259A coloca vetor correto no barramento de dados
- ✓ UCP processa interrupção

32



as de Computadores

Acesso Direto à Memória

Fundamentos de Arquiteturas de Computadores

- ✓ E/S programada e por interrupção requerem intervenção ativa da UCP
 - *Taxa de transferência é limitada pela capacidade de atendimento da UCP*
 - *UCP fica ocupada gerenciando a transferência de dados*

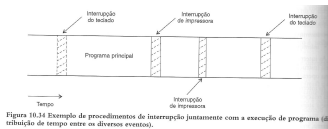


Figura 10.34 Exemplo de procedimentos de interrupção juntamente com a execução de programa (a distribuição de tempo entre os diversos eventos).

- ✓ DMA pode ser uma técnica mais eficiente

34

Função do DMA

Fundamentos de Arquiteturas de Computadores

- ✓ Módulo adicional de hardware no barramento
- ✓ Controlador de DMA "imita" a UCP para realizar operações de E/S

35

E/S com uso de DMA

Fundamentos de Arquiteturas de Computadores

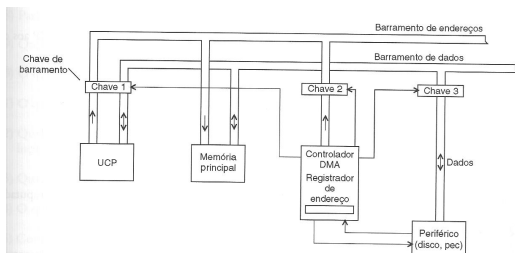
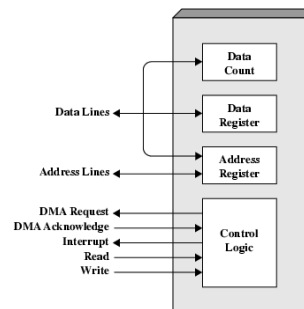


Figura 10.35 Operação de E/S com emprego de DMA.

36

Diagrama de um Módulo para DMA

Fundamentos de Arquiteturas de Computadores



37

Operação do DMA

Fundamentos de Arquiteturas de Computadores

- ✓ UCP indica ao controlador de DMA:
 - *Operação: Escrita/Leitura*
 - *Endereço do dispositivo*
 - *Endereço inicial do bloco de memória para dados*
 - *Quantidade de dados a serem transferidos*
- ✓ UCP executa outra tarefa
- ✓ Controlador de DMA processa transferência
- ✓ Controlador de DMA envia interrupção quando operação é finalizada

38

Roubo de Ciclo pelo DMA

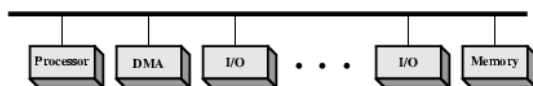
Fundamentos de Arquiteturas de Computadores

- ✓ Controlador de DMA toma conta do barramento por um ciclo
- ✓ Transfere dados
- ✓ Diferente de interrupção
 - *UCP não realiza troca de contexto*
- ✓ UCP é suspensa imediatamente antes de acessar o barramento
 - *Antes da busca da instrução e do operando, antes de armazenar dados na memória*
- ✓ Diminui velocidade de processamento da UCP mas evita que a UCP tenha que realizar a transferência

39

Configurações de DMA (1)

Fundamentos de Arquiteturas de Computadores

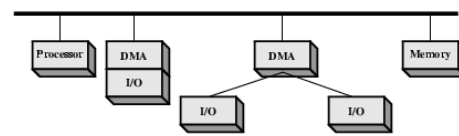


- ✓ Barramento único, Controlador de DMA separado
- ✓ Cada transferência utiliza duas vezes o barramento
 - *E/S para DMA e do DMA para memória*
- ✓ UCP é suspensa 2 vezes

41

Configurações de DMA (2)

Fundamentos de Arquiteturas de Computadores



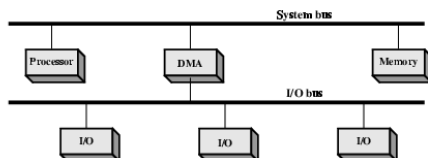
(b) Single-bus, Integrated DMA-I/O

- ✓ Barramento único, Controlador de DMA integrado
- ✓ Controlador pode suportar mais de um dispositivo
- ✓ Cada transferência utiliza o barramento uma única vez
 - *DMA para memória*
- ✓ UCP é suspensa uma única vez

42

Configurações de DMA (3)

Fundamentos de Arquiteturas de Computadores



(c) I/O bus

- ✓ Barramento de E/S separado
- ✓ Barramento suporta todos dispositivos que podem realizar DMA
- ✓ Cada transferência utiliza o barramento uma vez
 - *DMA para memória*
- ✓ UCP é suspensa uma vez

43