

Sistemas Operacionais

Cristina Boeres

minha página

<http://www.ic.uff.br/~boeres>

página do curso de SO

<https://sites.google.com/ic.uff.br/sistemas-operacionais-icuff/home>

meu e-mail

boeres@ic.uff.br

Sistemas Operacionais

O que é um SO?

- ◆ um gerenciador de recursos?
- ◆ uma interface?
- ◆ máquina virtual?

Sistemas Operacionais

- ◆ Revisão: UCP, Memória, Processos e Interrupções
- ◆ Processos e *threads*
- ◆ Gerenciamento de Memória
- ◆ Memória Virtual
- ◆ Concorrência
- ◆ *Deadlocks*
- ◆ Escalonamento de Processos
- ◆ Sistemas de Arquivos

Sistemas Operacionais

Livro texto

- Operating Systems: Internals and Design Principles, W. Stallings
 - (<http://williamstallings.com/OperatingSystems/>)

Bibliografia Complementar:

- Operating Systems Concepts, A. Silberschatz e P.B. Galvin, Wiley
 - (em português, Sistemas Operacionais em Java, Ed Elsevier)
- ◆ Modern Operating Systems, A. Tanenbaum and H. Bos, Ed. Prentice Hall

Sistemas Operacionais

Atividades do curso

- P1 : 9 de maio
- P2: 27 de junho
- um trabalho prático T
- participação (bônus)

- Nota final: $P1 * 0.4 + P2 * 0.4 + T * 0.2 + \text{bônus}$

- 2ª chamada - somente para quem faltar a uma das provas

- VS

"Que é um Sistema Computacional ?"™



"Que é um Sistema Computacional ?"™

- Uma ferramenta, que tem
 - Bom desempenho ?
 - Baixo custo ?
 - Facilidade de uso ?
 - ... ?
- Pode ajudar usuários resolver problemas que envolvam
 - Processamento
 - Armazenamento
 - Comunicação



Arquitetura X Organização de computadores

- **Arquitetura** de computadores refere-se às características visíveis pelo programador
 - conjunto de instruções
 - número de bits de dados
 - mecanismos de E/S
 - modos de endereçamento

Arquitetura X Organização de computadores

- **Organização** de computadores refere-se às unidades operacionais e suas interconexões que implementam as características da arquitetura
 - detalhes de h/w como sinais de controle
 - interfaces
 - tecnologia de memória
 - etc

Arquitetura x Organização

- Especificar se um computador deve ou não ter uma instrução de multiplicação constitui uma decisão de projeto de ...

Arquitetura

- Definir se essa instrução será implementada por uma unidade específica de multiplicação ou por um mecanismo que utiliza repetidamente sua unidade de soma é uma decisão de ...

Organização

Arquitetura x Organização

- Fabricantes oferecem famílias de modelos, com mesma arquitetura e diferentes organizações
 - custo e desempenho diferentes
 - compatibilidade de código
 - mudança da organização com tecnologia
- Conclusão
 - uma organização deve ser projetada para implementar uma especificação particular de arquitetura

Estrutura e Função

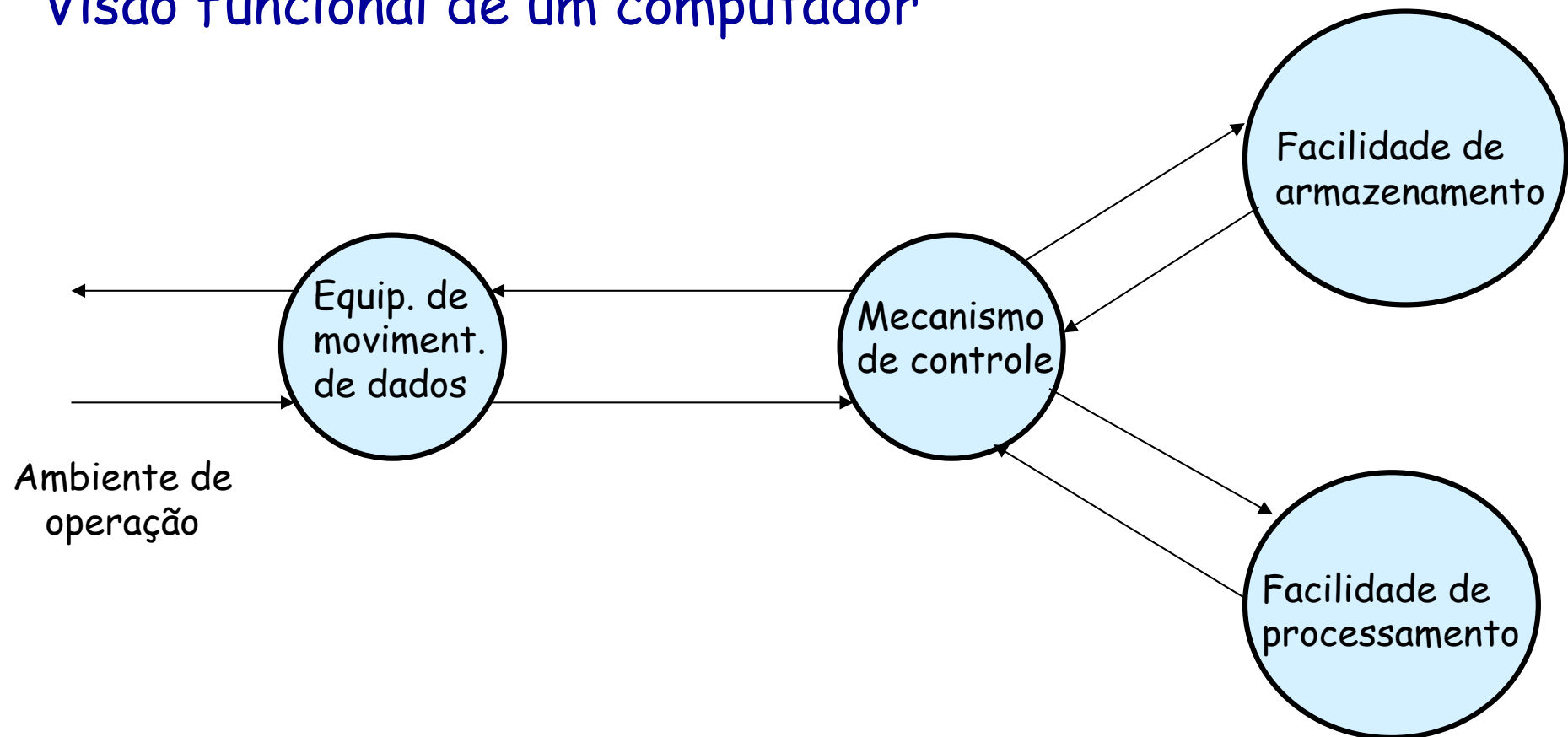
- Um computador é um sistema hierárquico
 - *máquina multiníveis*
- Em cada nível, o projetista se preocupa com sua estrutura e sua função
 - Estrutura é a forma como os componentes se relacionam
 - Função é a operação de cada componente parte dessa estrutura

Função

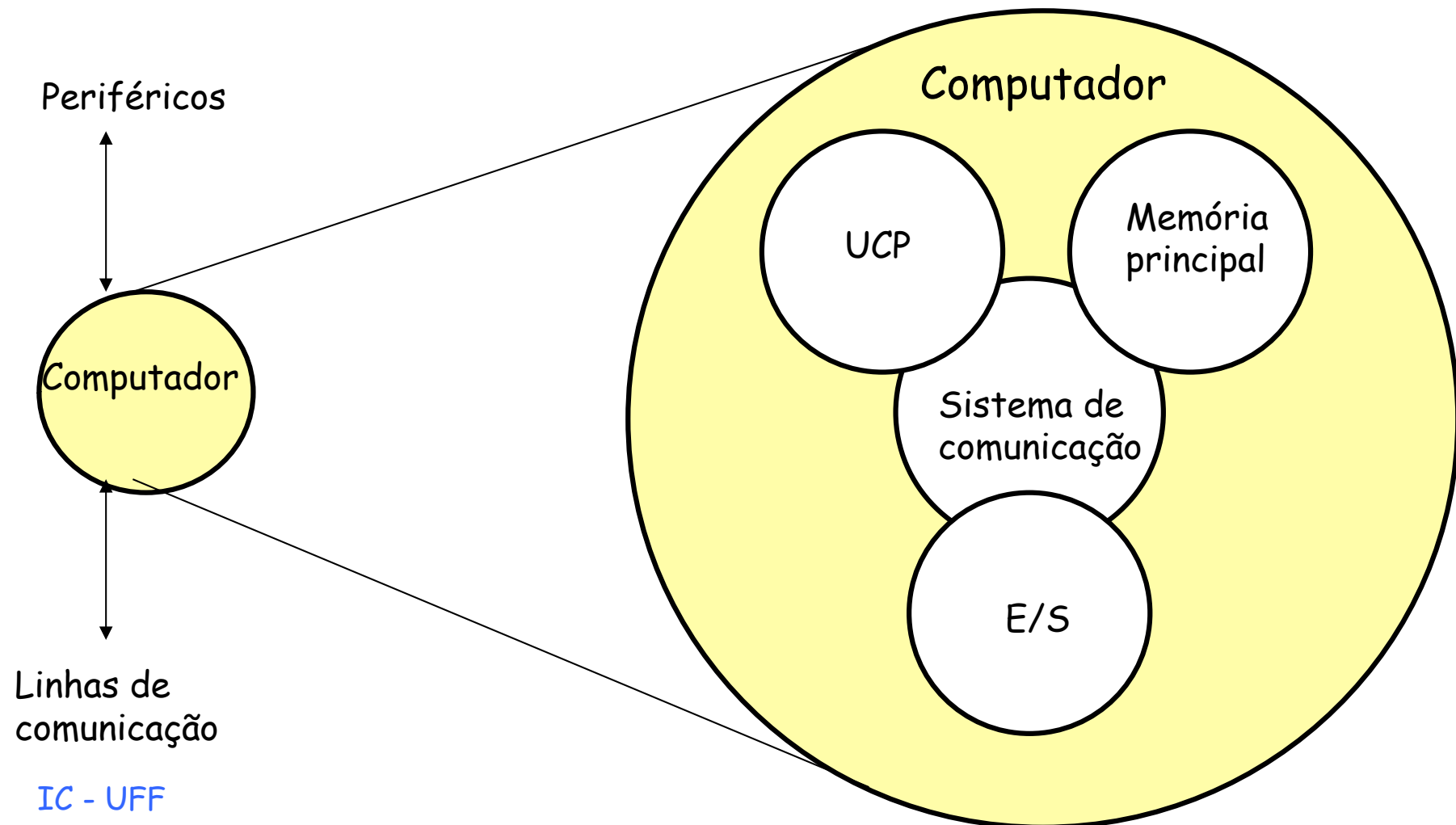
- Funções básicas de um computador são:
 - Processamento de dados
 - Armazenamento de dados
 - Movimentação de dados
 - Controle
 - Comunicação

Visão funcional

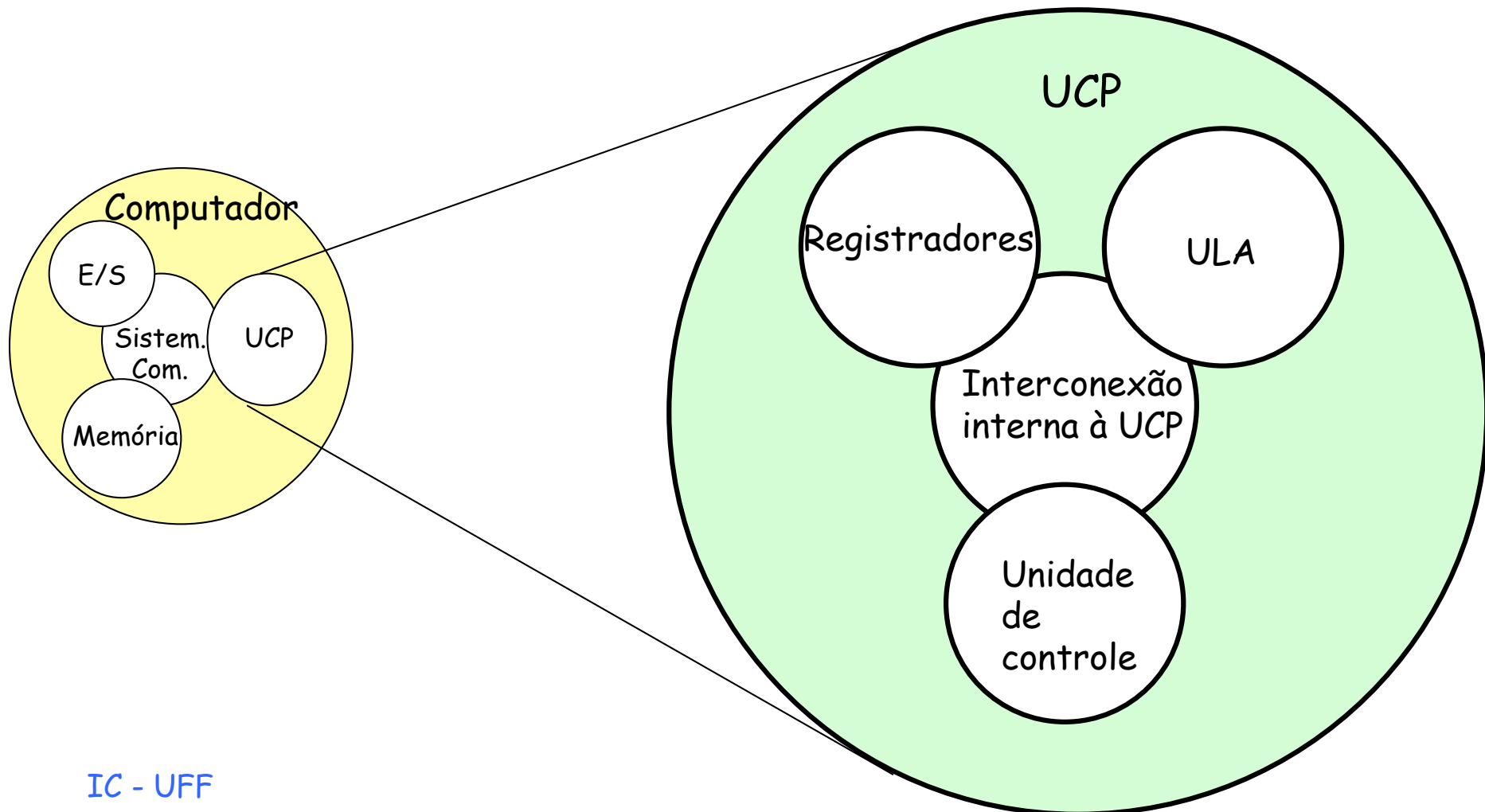
Visão funcional de um computador



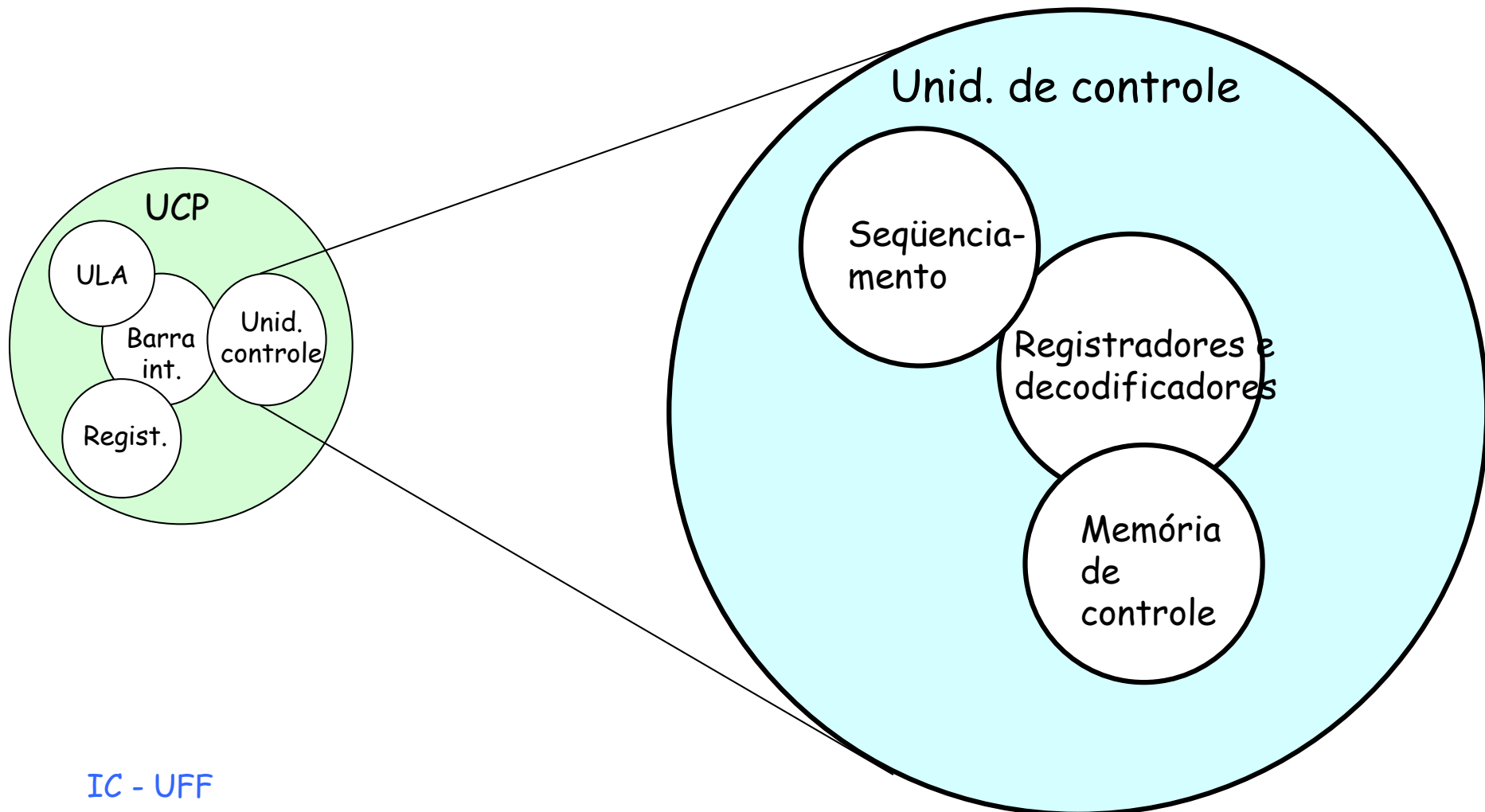
Estrutura: visão macro



Estrutura: a UCP



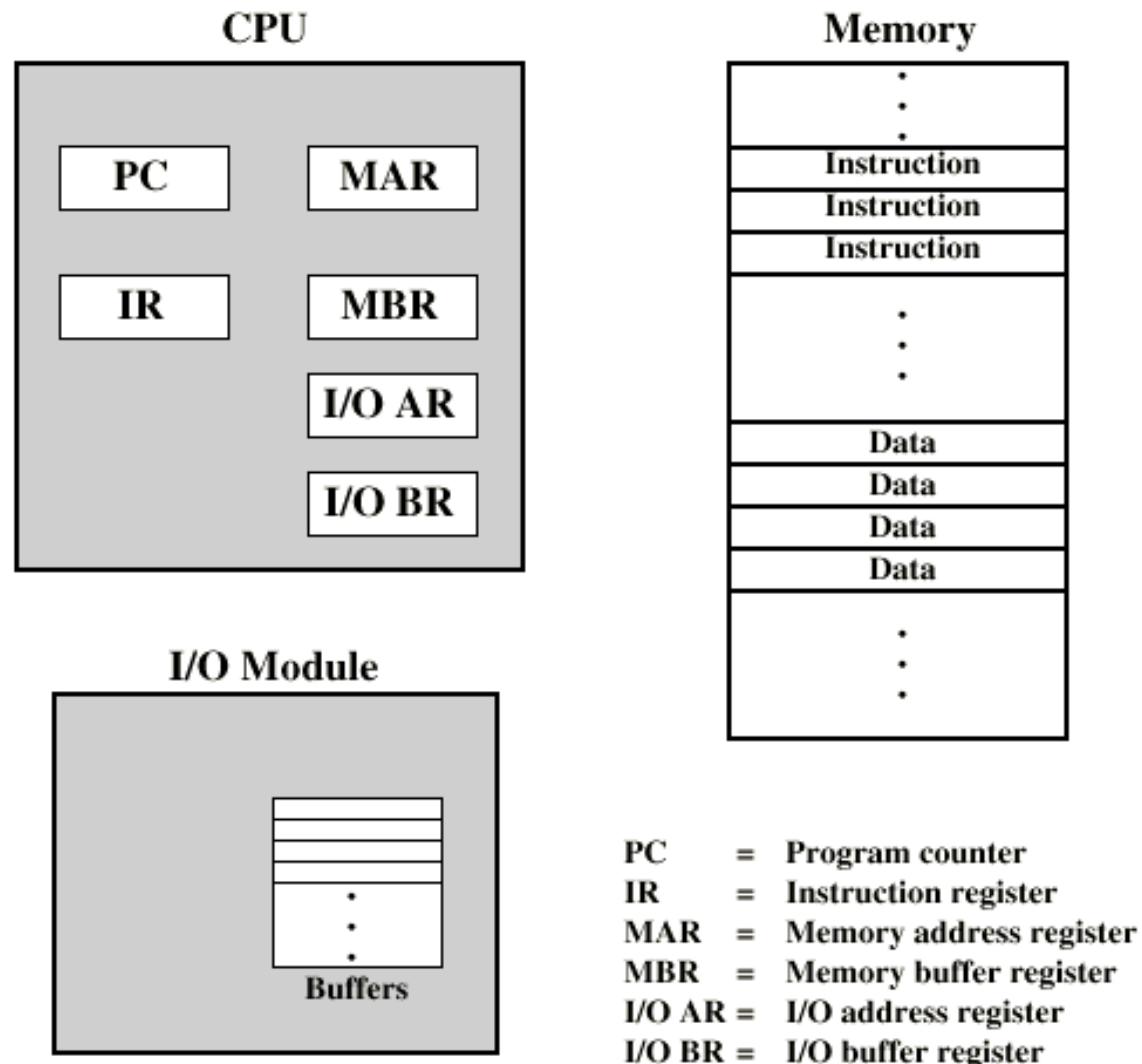
Estrutura: unidade de controle



Uma pequena revisão

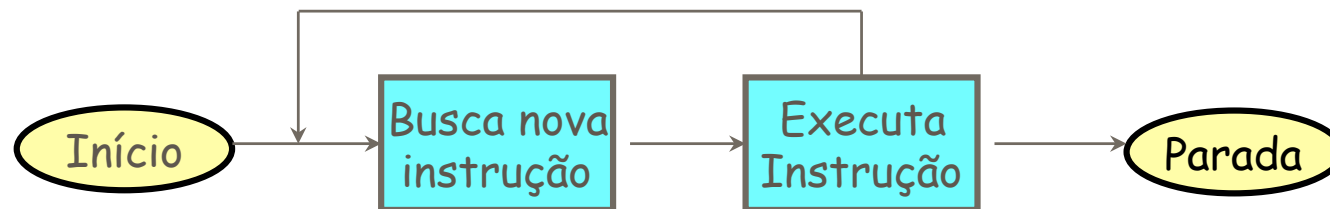
- revisando pontos importantes.....
- agora queremos entender um ambiente computacional onde vários programas podem ser submetidos e executados *ao mesmo tempo*

Computador: visão macro



Ciclo de instrução

Ciclo de busca Ciclo de execução



Ciclo de busca

- PC contém o endereço da próxima instrução
- Instrução é buscada e colocada no IR para ser interpretada
- PC é incrementado (quase sempre!)

Ciclo de execução

- UCP interpreta instrução e executa ação:
 - UCP \leftrightarrow memória: transferência de dados
 - UCP \leftrightarrow E/S: transferência de dados
 - processamento de dados: operação lógica ou aritmética
 - controle: alteração da seqüência de operação
 - combinação dessas ações

Organização da UCP

- Funções da UCP:
 - buscar instruções
 - interpretar instruções
 - buscar dados
 - processar dados
- UCP precisa fazer armazenamento temporário: registradores

Registradores

- Espaço de trabalho temporário
- Quantidade e função varia entre processadores
- Uma das principais decisões de projeto
- Nível superior da hierarquia de memória

Classificação:

- visíveis
- restritos
 - controle
 - estado

Registradores visíveis

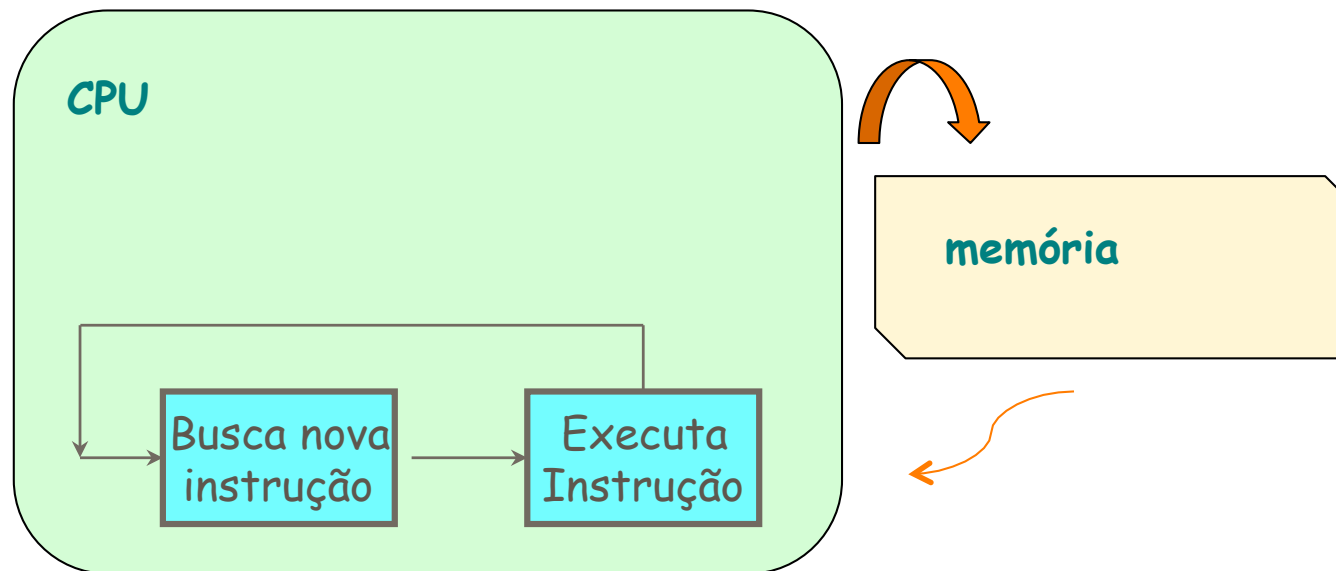
- São os que podem ser referenciados através de linguagem de máquina
 - de uso geral (dados e endereçamento)
 - de dados (e.g., acumulador)
 - de endereço: segmento, índice, pilha, ...
 - códigos de condição (só leitura)

Registradores de controle e estado

- visibilidade restrita
- essenciais: PC, IR, MAR, MBR
- PSW: *Program Status Word*
 - códigos de condição =
sinal + zero + vai-um + estouro + permissão/inibição de
interrupção + modo supervisor/não
- Outros registradores importantes: SP, apontador de PCB, interrupção vetorizada

Processador: recursos central de controle

- CPU não deve ficar ociosa
 - princípio de multiprogramação



Modos de Execução

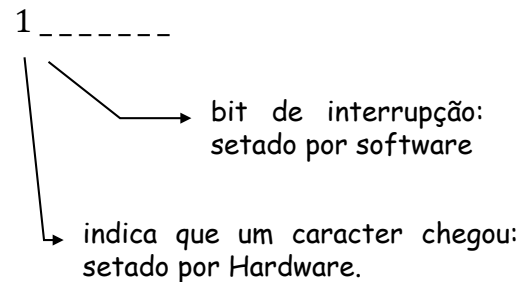
- Modos de execução do processador em relação à entrada
 - E/S programáveis
 - E/S por interrupção
 - DMA (Direct Memory Access)

E/S programáveis

- Usada em alguns microprocessadores mais simples
- O processador está sempre checando a entrada
 - 1) A instrução de entrada (ou saída) seleciona o dispositivo desejado (registradores podem estar associados a um determinado dispositivo)
 - 2) um caractere é transmitido entre um registrador específico e o dispositivo selecionado
 - 3) uma instrução de entrada (ou saída) deve ser especificada para cada caractere a ser lido (ou escrito)

E/S programáveis

- dois registradores estão associados a um dispositivo de E/S
 - registrador de *estado* (de 8 bits): dois bits mais a esq são necessários
 - Registrador de *dado*



- Um loop testa se o bit de estado está setado
- Se positivo, software (SO, por exemplo) lê o caractere para o reg de dado

E/S programáveis

Desvantagem de E/S programável:

- loop é eterno
- o processador fica em *espera ocupada*
 - não é desvantagem quando a CPU tem tarefa específica

E/S por interrupção

- Para evitar a espera ocupada: o dispositivo gera uma interrupção quando necessário.
- Sinal por hardware - avisa a CPU para iniciar a rotina de E/S

Interrupção

- emitir um sinal para a CPU
- CPU inicia o tratamento de E/S
- Avisa ao dispositivo quando acabar de operar
 - reseta o registrador de estado

E/S por interrupção

- A interrupção pode resolver o problema da espera ocupada, mas todo caractere lido/escrito é seguido de interrupção, o que ainda não é uma boa solução.
- Solução melhor
 - E/S programável, mas com um dispositivo especifica para realizar esta tarefa: DMA

DMA (Direct Memory Access)

- chip controlador que acessa diretamente o dispositivo
- tem pelo menos 4 registradores, carregados por software (série de macro instruções) executada pela CPU.

Descrição dos registradores:

- **R1:** o endereço de MP de onde serão Lidos/Escritos.
- **R2 :** Quantos bytes/palavras serão transferidos.
- **R3 :** # do dispositivo ou o espaço de end. da E/S a ser usado.
- **R4 :** dado deve ser lido ou gravado.

DMA (Direct Memory Access)

- Escrever um bloco de 32 bytes do endereço 100 da MP para o terminal (dispositivo 4)

① inicialização: A CPU escreve os dados nos registradores:

$R1 \leftarrow 100$

$R2 \leftarrow 32$

$R3 \leftarrow 4$

$R4 \leftarrow \text{WRITE}$

DMA (Direct Memory Access)

- **Início)** O controlador de DMA para ler do endereço R1
 - como faria a CPU
- DMA pede ao dispositivo selecionado (dispositivo 4) para efetuar a operação (escrita).
- DMA incrementa R1 e decrementa R2.
- Se $R2 > 0$ então volta para Início)
- Se $R2 = 0$: transferência finalizada e um sinal enviado para a linha de interrupção da CPU

DMA (Direct Memory Access)

- CPU
 - inicializa DMA
 - detecta o sinal de interrupção.
- Atenção: mesmo usando DMA, a CPU pode ficar esperando, pela lógica do próprio programa sendo executado ou por ter como executar outros processos.

Prioridade de acesso ao barramento da MP

- a DMA tem maior prioridade de acesso aos barramentos de memória. Por que?

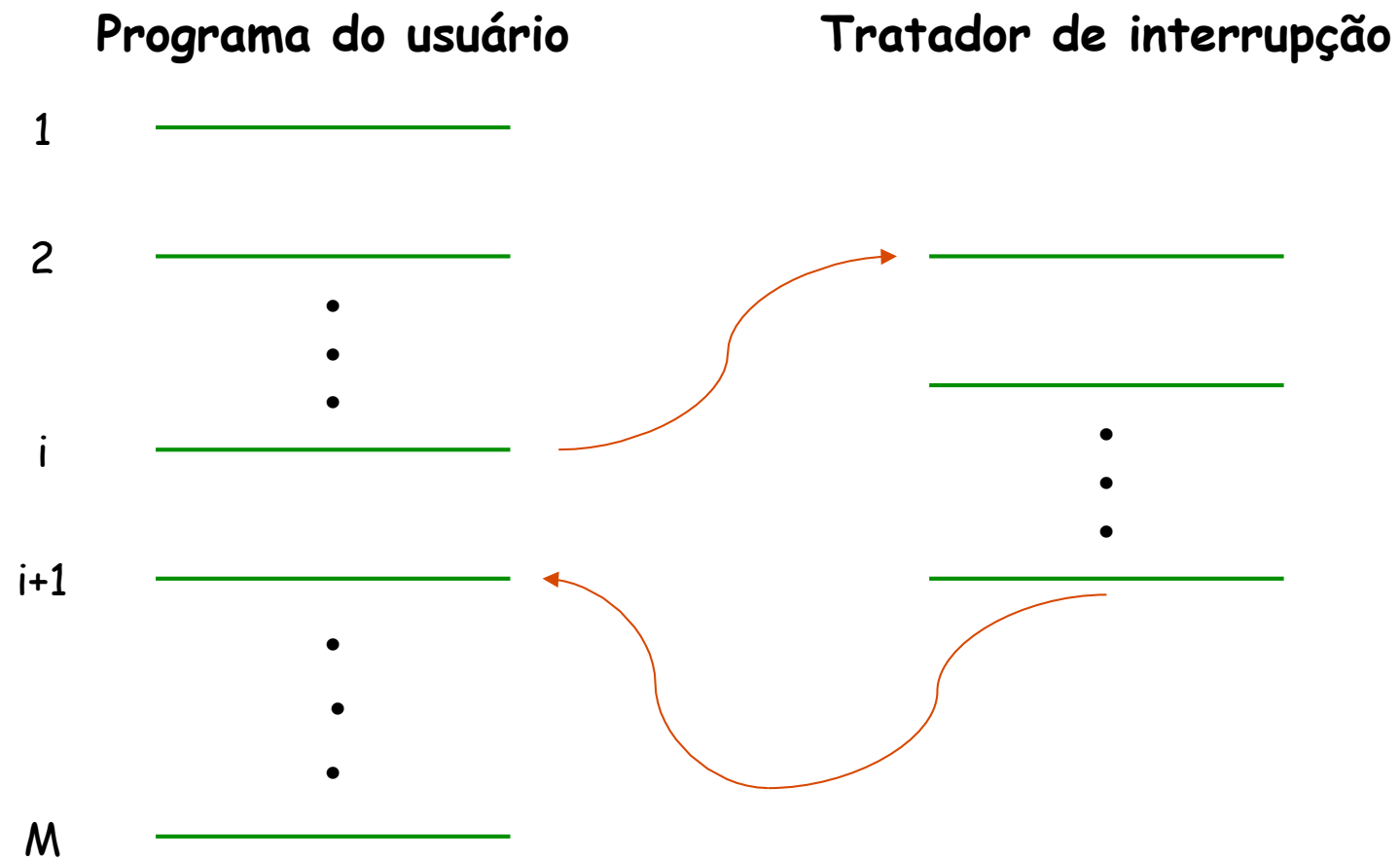
Controladores especiais

- CPU carrega para a MP um programa
- executa uma instrução de *início* que indica qual o canal e o dispositivo de E/S a ser acionado
- o canal acionado restaura o endereço do programa e começa a executá-lo
- exemplo de instruções executadas pelo canal: READ, WRITE, READ BACKWARD, CONTROL, SENSE, desvios condicionais e HALT

Interrupção

- Mecanismo pelo qual outros módulos interrompem processamento normal da UCP
 - Basicamente associado a E/S
- Tipos mais comuns de interrupção:
 - programa (e.g., $\div 0$) → *traps*
 - temporização → escalonamento de processo
 - E/S (e.g., fim de escrita em disco)
 - falha de h/w (e.g., falta de energia)

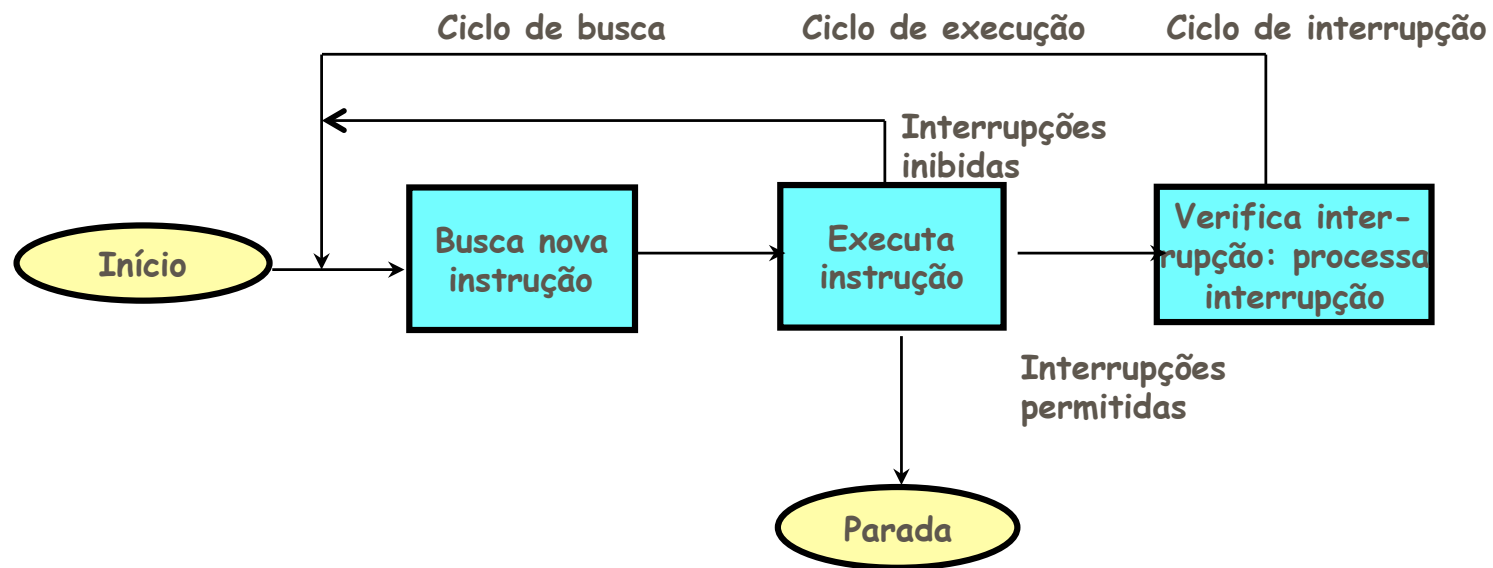
Transferência de controle



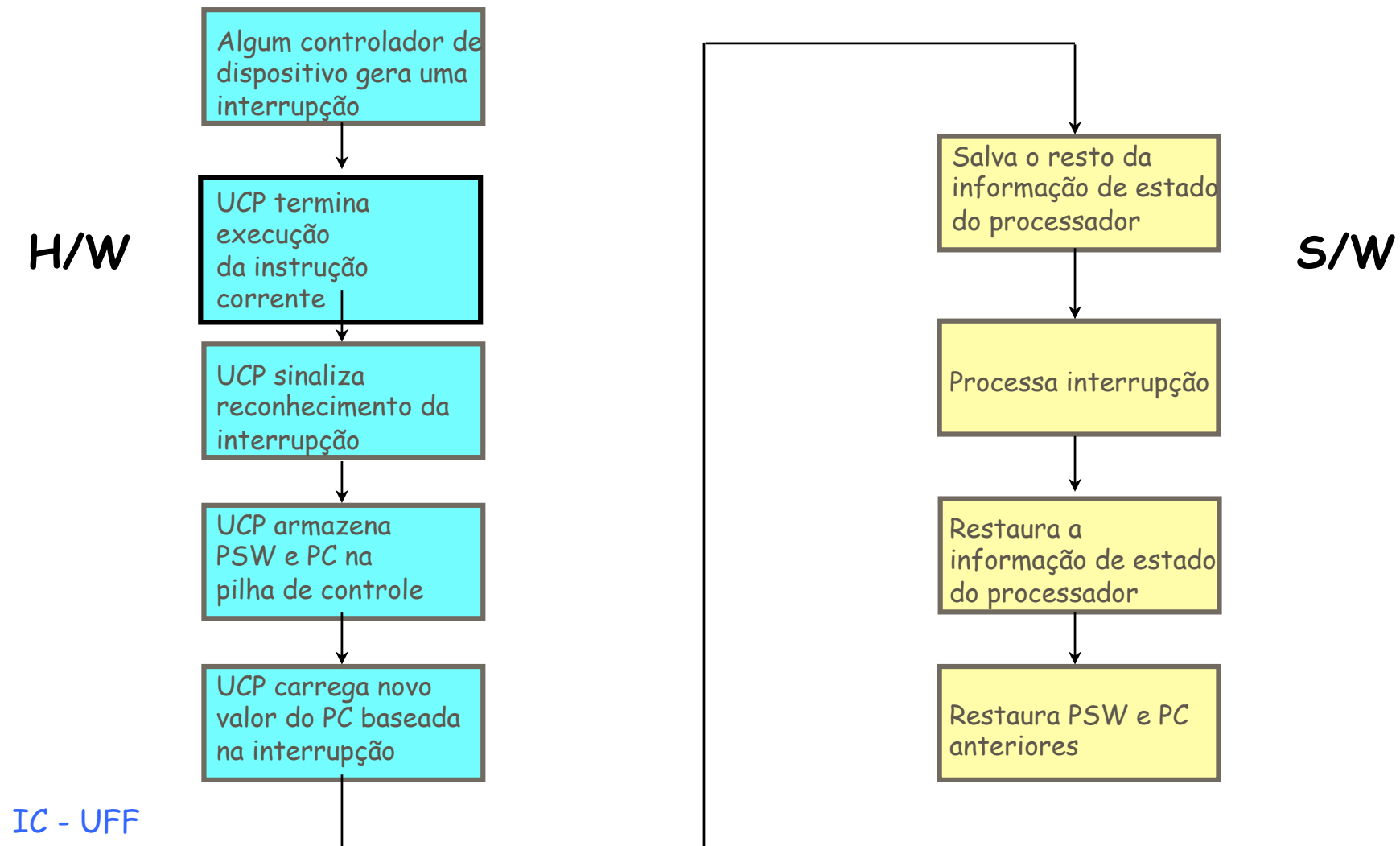
Tratador de interrupção

- Programa que determina a natureza da interrupção e que realiza o tratamento adequado
- Controle é transferido para este programa após salvamento de algumas informações
- É parte do sistema operacional

Ciclo de interrupção



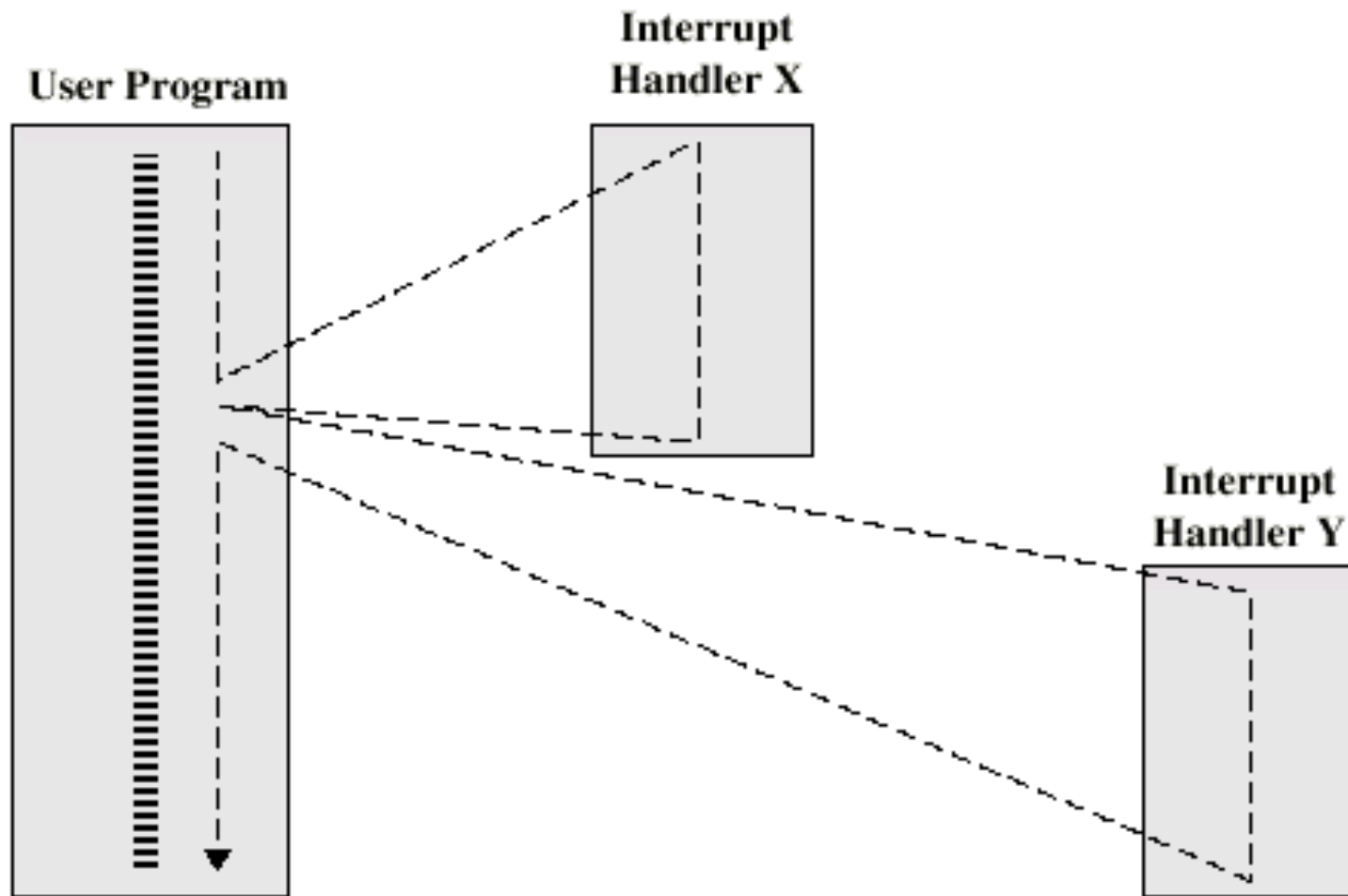
Tratamento da interrupção



Interrupções múltiplas (1)

- Desabilita outras interrupções: seqüencial
 - UCP ignora outras interrupções enquanto processa uma interrupção
 - interrupções pendentes só são verificadas ao fim do tratamento da interrupção corrente
 - interrupções tratadas na ordem seqüencial de ocorrência
 - Quando terminar o tratamento, as interrupções são habilitadas.
- **Vantagem:** simplicidade
- **Desvantagem:** falta de critério

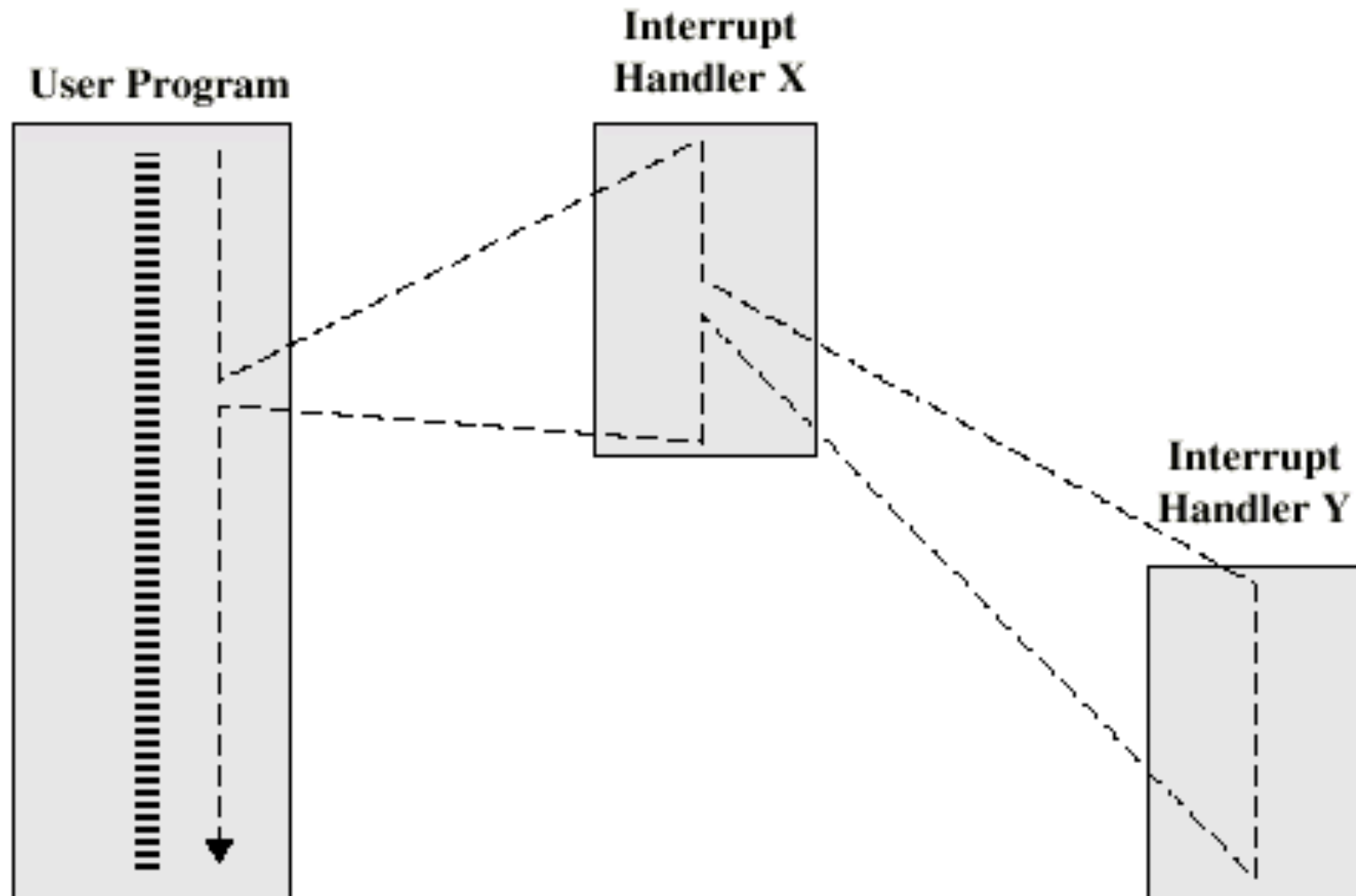
Tratamento seqüencial



Interrupções múltiplas (2)

- Definição de prioridades
 - interrupções de menor prioridade podem ser interrompidas por interrupções de maior prioridade
 - quando há o término do tratamento da(s) de maior prioridade, UCP trata a(s) de menor prioridade

Tratamento com prioridades



Tratamento com prioridades

t=0 → instruções sendo executadas

t=10 → interrupção devido impressora

- passos de hardware e software

t=15 → rotina de tratamento ainda sendo executada
linha de comunicação interrompe

rotina é interrompida e estado salvo

trata interrupção devido a linha de comunicação

t=25 → termina o tratamento devido a linha de com.
restaura o tratamento devido a impressora

.....