



PROCESSOS

INF009 – SISTEMAS OPERACIONAIS

PORQUE EU PRECISO ENTENDER ESTE ASSUNTO?

Para entender como um computador consegue executar várias tarefas simultaneamente e **qual o impacto que isso pode ter em meus programas!**



OS PROGRAMAS DE ONTEM...

Antigamente, os computadores eram máquinas dedicadas:

- Possuíam apenas um usuário;
- Executavam apenas um programa por vez;
 - Programas em execução tinham total controle dos recursos do computador.

E OS PROGRAMAS DE HOJE!

Hoje trabalhamos com execução simultânea:

- Diversos programas são mantidos na memória principal;
- A execução dos programas pode ser iniciada por vários usuários.

Lembre-se também que o próprio sistema operacional é um programa!

SOBRE OS PROGRAMAS...

Quando ligamos o computador, vários programas começam a ser executados:

- Programas ativados pelo sistema operacional;
- Programas ativados pelo usuário.

Cada um destes programas possui vários **processos**.

PROCESSOS :: DEFINIÇÃO

Definição:

— Programa em execução

Silberschatz, Tanenbaum



Mas qual a diferença entre processo e programa?

PROGRAMA VS. PROCESSO

Programa:

- Sequência de instruções;
- Se apresenta de maneira estática.

Processo:

- Entidade dinâmica;
- Altera o seu estado na medida em que sua execução avança;
- Composto por um programa (código), seus dados e o seu contexto.



ATENÇÃO

- Em geral, um processo é mapeado em um programa.



- Observe que:
 - um processo pode ser executado por vários usuários;
 - um processo pode ser executado (simultaneamente) várias vezes pelo mesmo usuário.



VAMOS PENSAR SOBRE EXEMPLOS

- Um programa que possui várias **instâncias** de execução
- Um contexto onde um programa é inicializado por diversos usuários.
- Qual o fluxo de execução de um programa?

CICLO DE VIDA DE UM PROCESSO

Criação

Execução

Finalização



CRIAÇÃO DE PROCESSO

- Momento da execução
- Pode ser feita através de *chamadas de sistema* (Ex.: fork)
- Processos criados podem estar associados a sessão de trabalho (shell Linux)



CRIAÇÃO DE PROCESSO

O que motiva a criação de um processo?

- Inicialização do sistema operacional;
- Inicialização de um programa;
- Chamada de sistema;

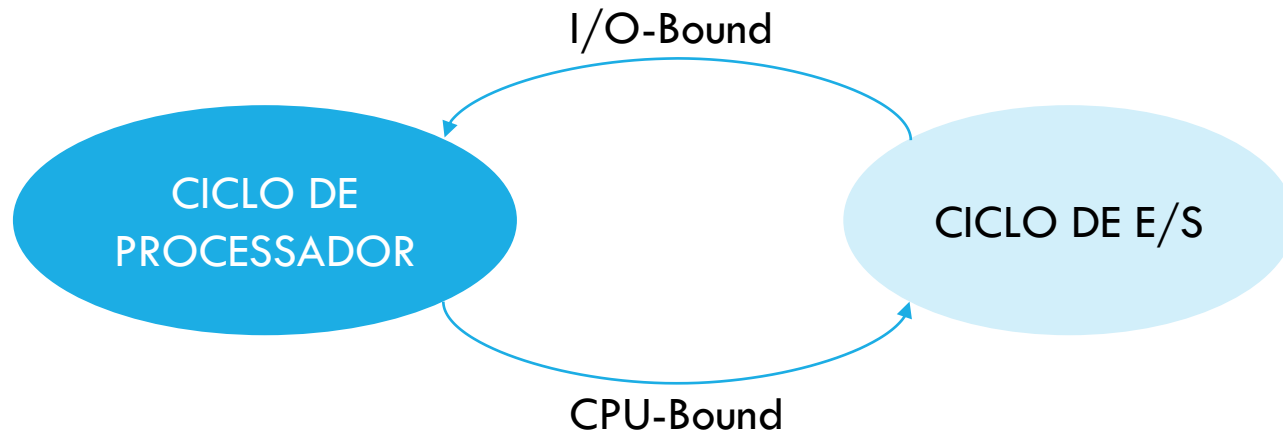


EXECUÇÃO DE PROCESSOS

- Processos podem operar de suas maneiras:
 - **CICLO DE PROCESSADOR:** baseado no tempo que ocupa na CPU
 - **CICLO DE ENTRADA E SAÍDA:** baseado no tempo em espera pela conclusão de um evento de E/S
- Primeiro ciclo é sempre de processador.



EXECUÇÃO DE PROCESSO





EXECUÇÃO DE PROCESSO

Os processos podem executar de duas formas:

- Em *FOREGROUND*
 - Processos que interagem com os usuários
- EM *BACKGROUND*
 - Não associados a usuários
 - Possuem funções específicas



FINALIZAÇÃO DE PROCESSO

O que motiva a finalização de um processo?

- Saída normal;
- Saída com erro;
- *Fatal Error* (involuntário);
- Outro processo (involuntário)

ATRIBUTOS DE UM PROCESSO :: PROCESS CONTROL BLOCK (PCB)

Representação do processo para o sistema operacional

pointer	process state
process number	
program counter	
registers	
memory limits	
list of open files	
⋮	

Silberschatz, capítulo 2

ATRIBUTOS DE UM PROCESSO :: PROCESS CONTROL BLOCK (PCB)

Informações associadas a cada processo:

- Estado do processo
- Valor do Contador de Programa
 - indica a próxima instrução a ser executada
- Área para guardar valor dos registradores (dados)
- Dados para gerenciamento da CPU (escalonamento)
- Dados para gerenciamento de memória
- Número do processo
- Informações sobre E/S

ATRIBUTOS DE UM PROCESSO :: ESTADO

- Novo
- Pronto
- Executando
- Bloqueado
- Terminado

ATRIBUTOS DE UM PROCESSO :: ESTADO

Novo

- O processo está sendo criado, ou seja, seu código está sendo carregado em memória, junto com suas bibliotecas;
- As estruturas de dados do *kernel* estão sendo atualizadas para permitir sua execução.

Pronto

- Processo está em memória, pronto para ser executado, aguardando a disponibilidade do processador;
- **IMPORTANTE**: Os processos “prontos” são organizados em uma fila cuja ordem é determinada por algoritmos de escalonamento.

ATRIBUTOS DE UM PROCESSO :: ESTADO

Executando:

- Processo está executando suas instruções.

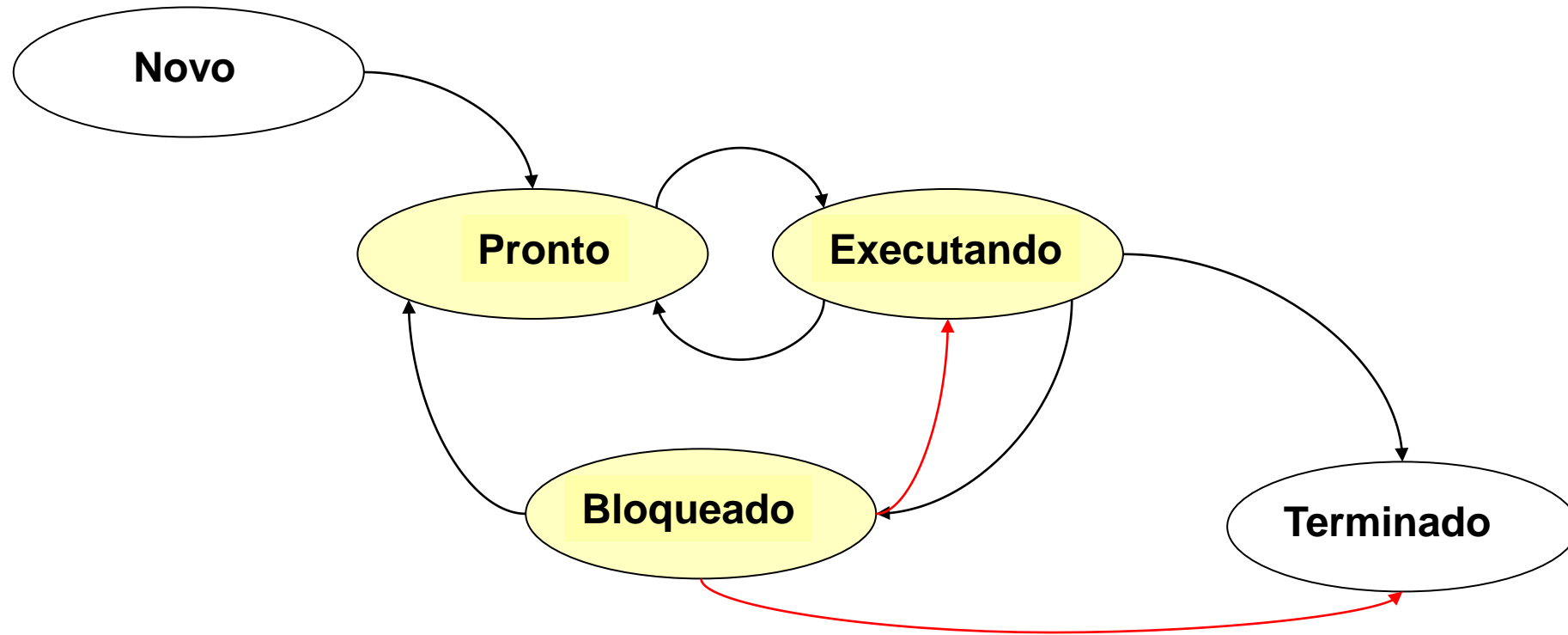
Bloqueado

- Processo não pode executar porque depende de recursos ainda não disponíveis (dados, algum tipo de sincronização, a liberação de algum recurso compartilhado);
- Processo simplesmente espera o tempo passar (em estado de “*sleeping*”).

Terminado

- A execução do processo foi encerrada e ele pode ser removido da memória do sistema.

DIAGRAMA DE ESTADOS DE UM PROCESSO



PROCESSO :: TRANSIÇÕES

... → **Novo**

- um novo processo é criado e começa a ser preparado para executar.

Novo → **Pronto**

- o novo processo termina de ser carregado em memória, estando pronto para executar.

Pronto → **Executando**

- o processo é escolhido pelo escalonador para ser executado, entre os demais processos prontos.

PROCESSO :: TRANSIÇÕES

Executando → Pronto

- esta transição ocorre quando se esgota a fatia de tempo destinada ao processo (*quantum*);
- Nesse momento o processo não precisa de outros recursos além do processador e por isso volta à fila de “pronto” para esperar novamente a disponibilidade do processador.

Executando → Terminada

- O processo encerra sua execução ou é abortado em consequência de algum erro (acesso inválido à memória, instrução ilegal, divisão por zero).
- Em geral, o processo que deseja terminar avisa ao sistema operacional através de uma chamada de sistema.

PROCESSO :: TRANSIÇÕES

Terminado → ...

- Quando terminado, um processo é removido da memória e seus registros e estruturas de controle no *kernel* são apagados.

Executando → **Bloqueado**

- caso o processo em execução solicite acesso a um recurso não disponível, ele abandona o processador e fica bloqueado até o recurso ficar disponível.

Bloqueado → **Pronto**

- quando o recurso solicitado pelo processo se torna disponível, ele pode então voltar ao estado de “pronto”.

RELAÇÃO ENTRE PROCESSOS

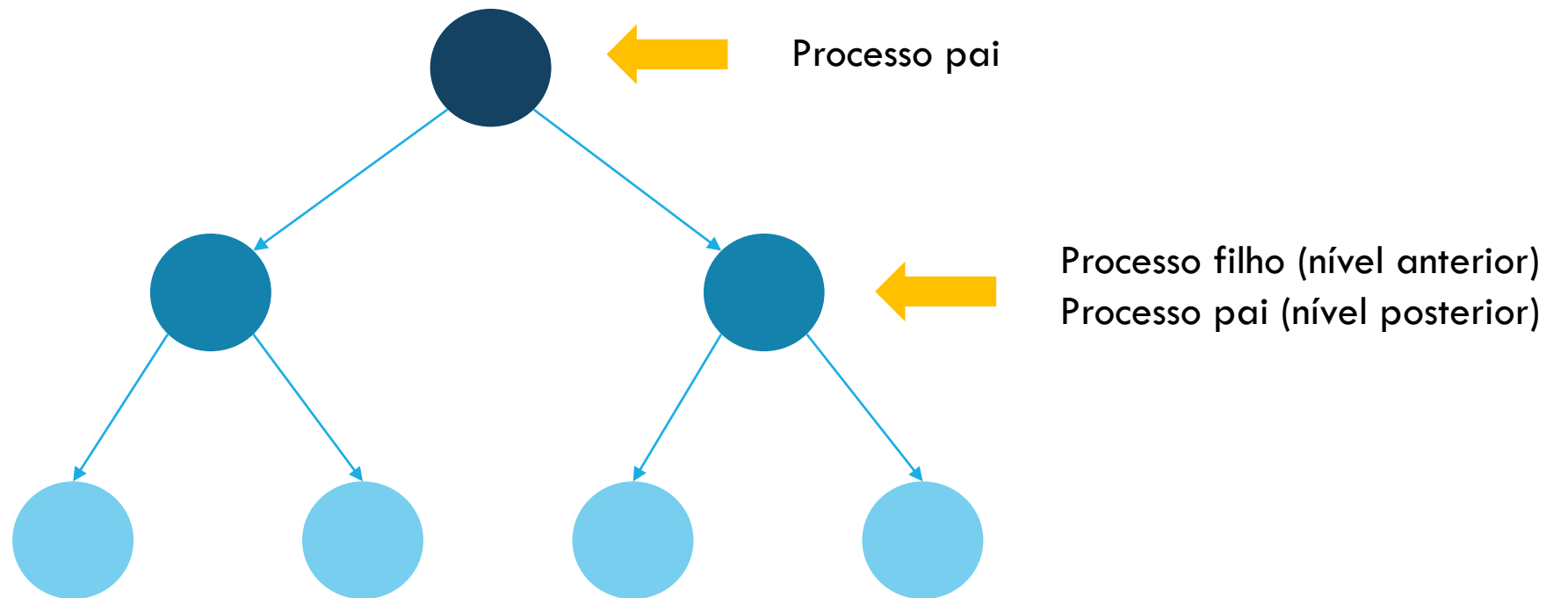
Independentes:

- Processos que não apresentam qualquer relação com outros processos;
- Este modelo pode ser usado também como uma simplificação de um modelo real, por ser menos restritivo.

Grupo de processos:

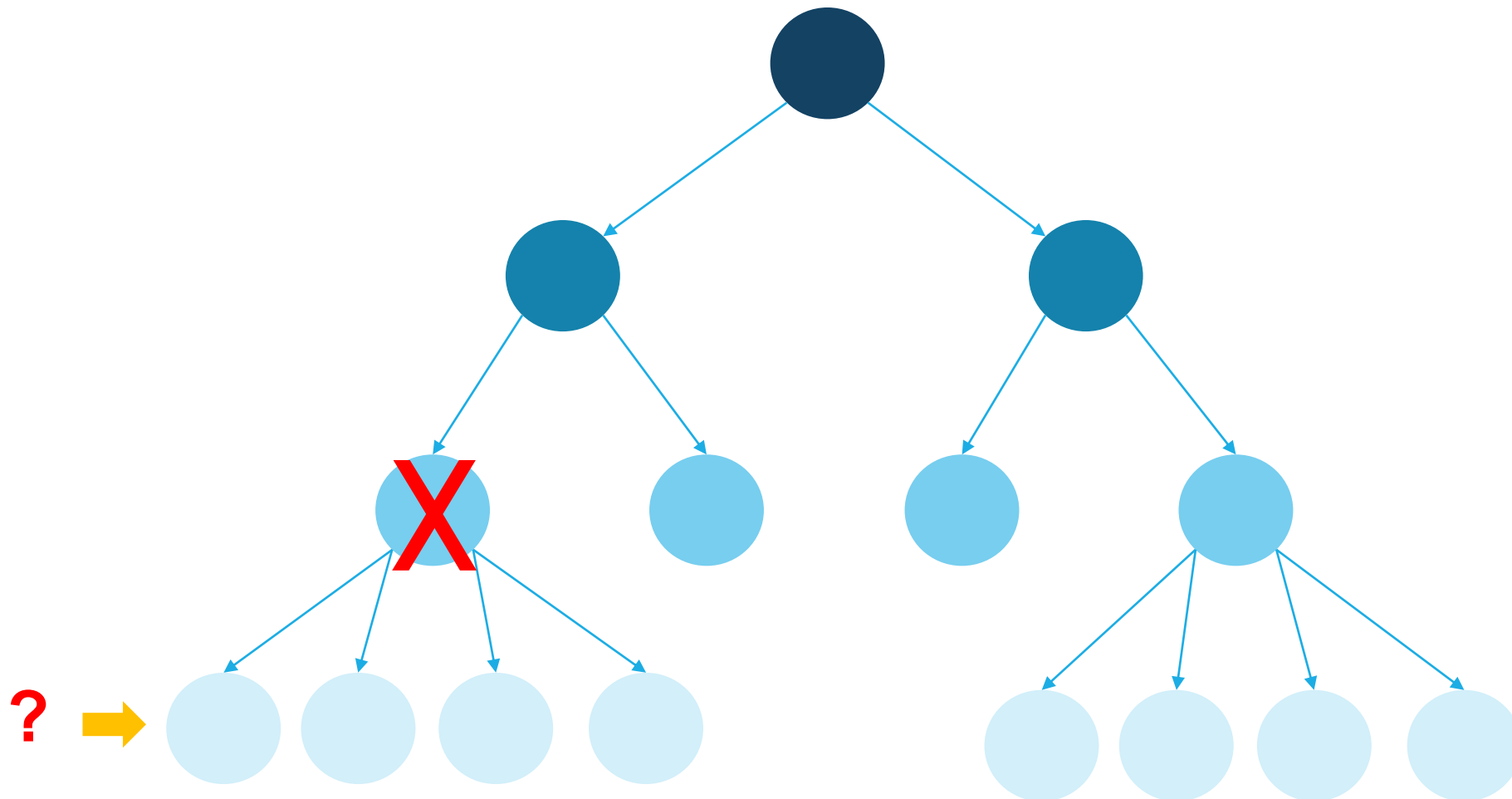
- Processos em grupo apresentam algum tipo de relação;
- Em geral, relação hierárquica (pai → filho)
 - Processos criados via system call
- Processos num grupo podem compartilhar recursos
- Representados em geral por um modelo em árvore

RELAÇÃO ENTRE PROCESSOS



IMPORTANTE: A árvore de processos é dinâmica.
A criação/finalização de processos adiciona ou remove nodos da árvore.

RELAÇÃO ENTRE PROCESSOS



RELAÇÃO ENTRE PROCESSOS

Três possíveis cenários:

- Finalizar todos os processos filhos;
- Descendência é herdada pelo processo “avô”;
- Adiar a finalização do processo pai até que os filhos sejam finalizados.

PROCESSOS E A MULTIPROGRAMAÇÃO

Para processos em grupo, recursos precisam ser compartilhados;

Compartilhamento de recursos deve ser feito de forma transparente:

- Processo não deve perceber a presença de outro processo;
- Ou seja, um processo não deve ser bloqueado, suspenso ou finalizado por um erro em outro processo.

Uma possibilidade é usar o mecanismo da interrupção.

COMO IMPLEMENTAR MULTIPROGRAMAÇÃO?

Ideia: Explorar as características físicas dos processadores

Tipicamente, é implementada através de:

- Modos de operação;
- Mecanismos de interrupção;
- Mecanismos de proteção.

MULTIPROGRAMAÇÃO :: MODOS DE OPERAÇÃO

Arquitetura física dos processadores admite dois modos de operação:

- **Modo kernel (privilegiado ou protegido):**
 - Funciona como um modo “privilegiado”, porque protege o hardware de possíveis ações “impensadas”
 - Permite a execução de todas as instruções;
 - Modo de execução utilizado pelo **sistema operacional**.
- **Modo usuário:**
 - Instruções privilegiadas não podem ser executadas;
 - Modo de execução dos processos dos usuários

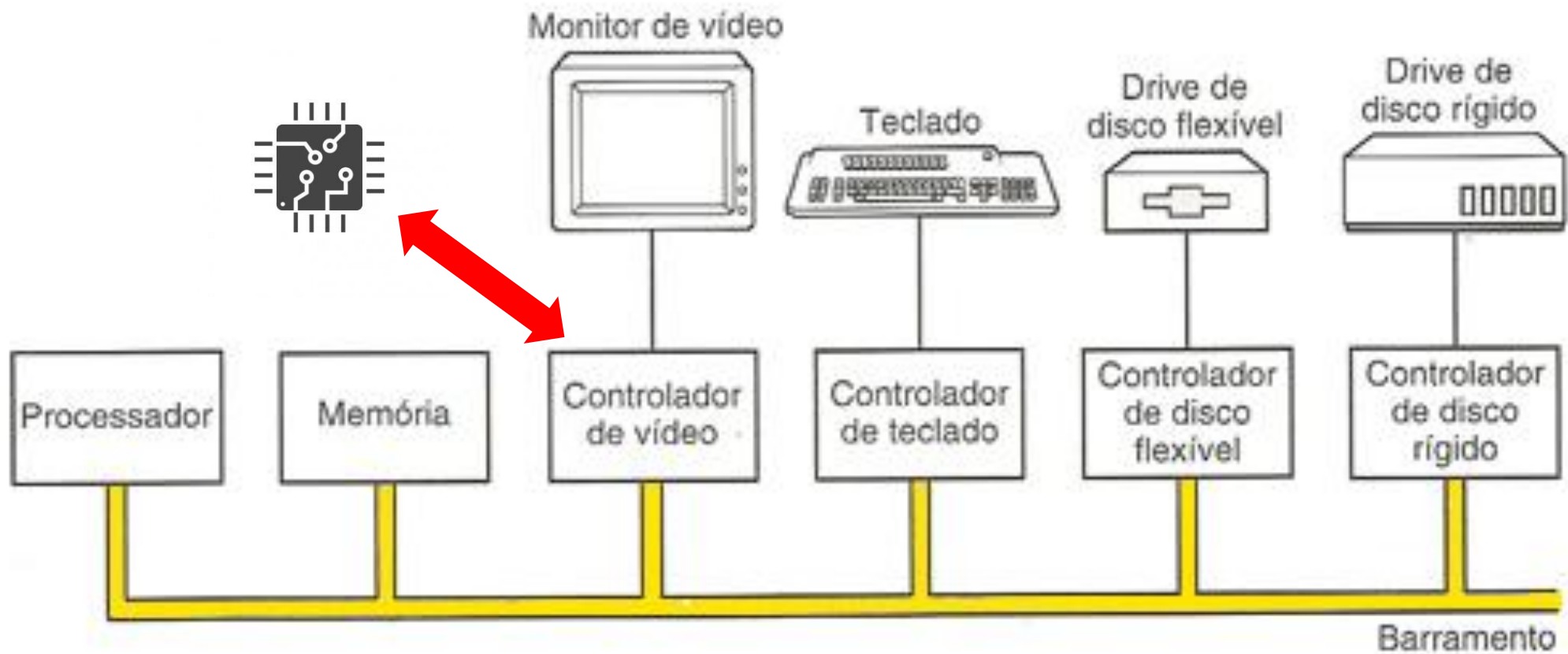
MULTIPROGRAMAÇÃO :: INTERRUPÇÃO

Definição:

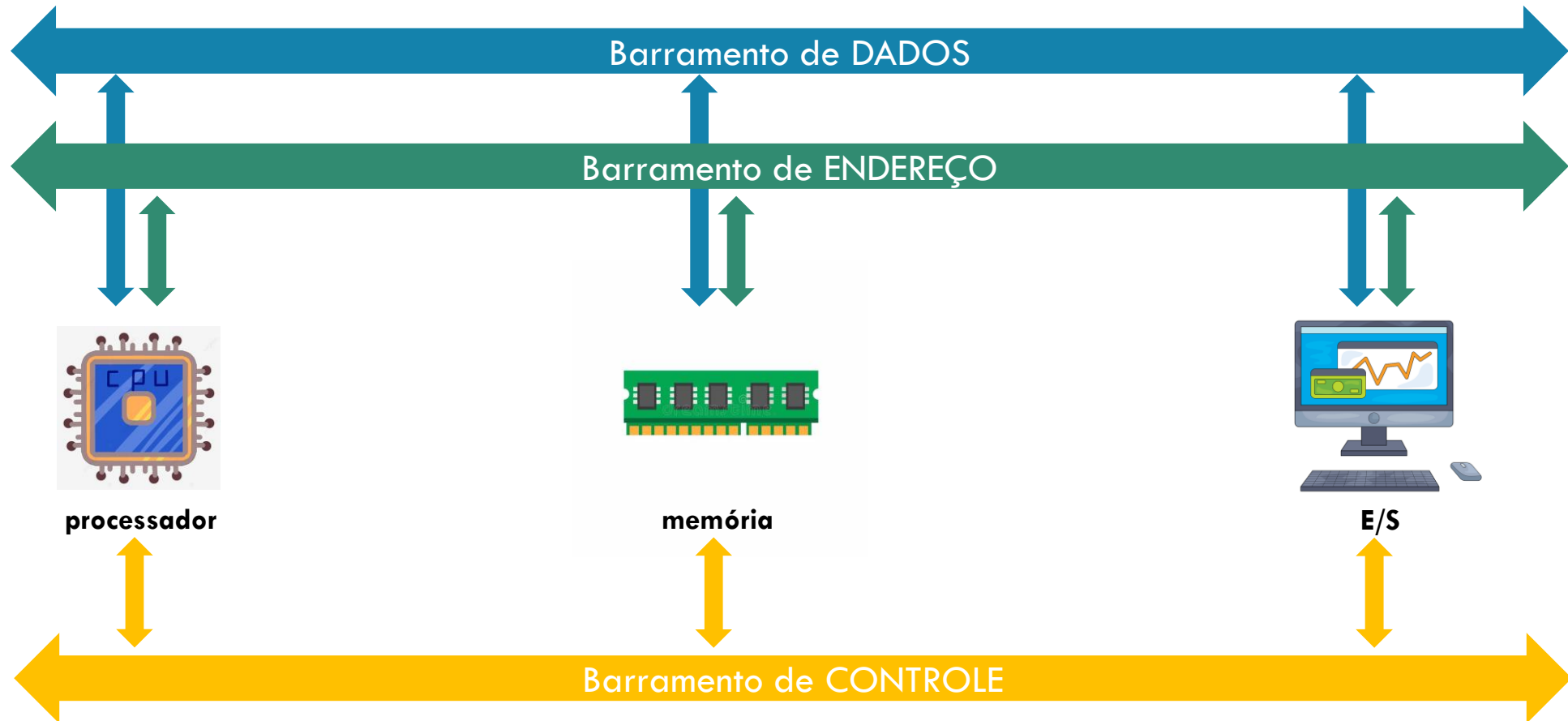
O mecanismo da **interrupção** é um recurso comum dos processadores e serve para chamar atenção do processador (principal).

Rômulo Oliveira, Alexandre Carissimi, Simão Toscani

RELEMBRANDO INF028



RELEMBRANDO INF028

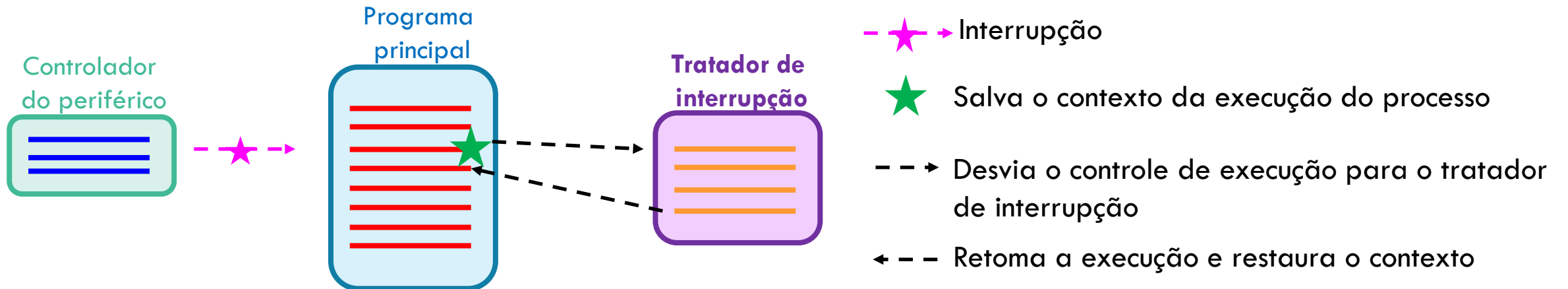


Barramento de controle é utilizado para envio de sinais elétricos associados a geração de uma interrupção.

MULTIPROGRAMAÇÃO :: INTERRUPÇÃO

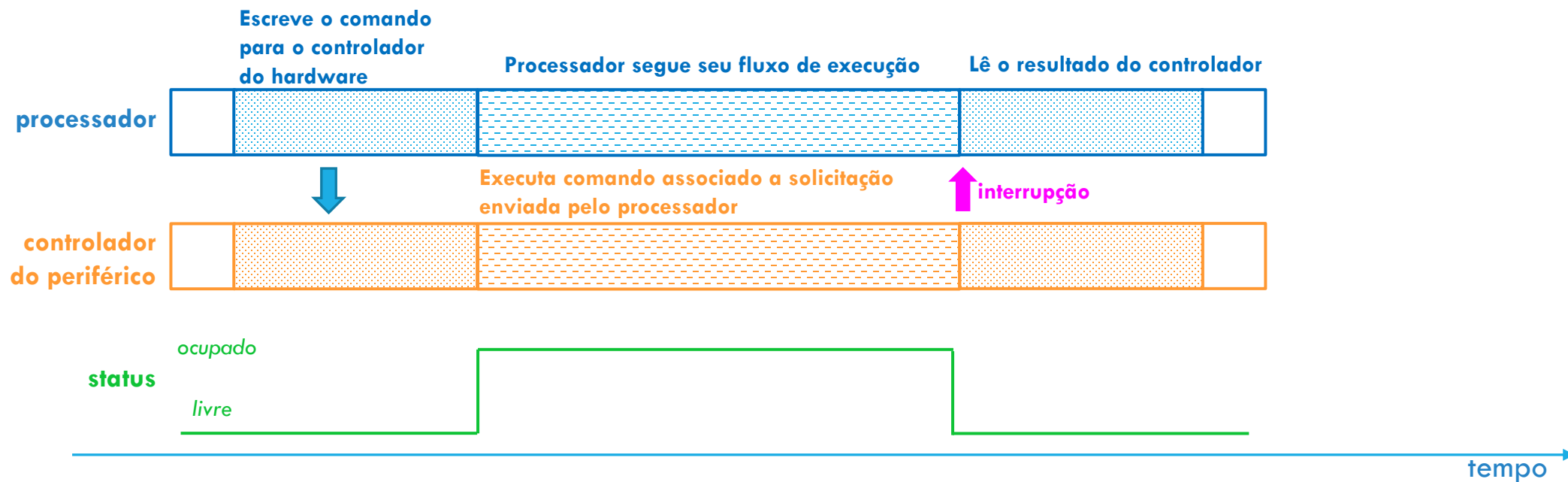
Uma interrupção sempre sinaliza a ocorrência de um evento.

Ela representa um desvio na execução da rotina principal.



MULTIPROGRAMAÇÃO :: INTERRUPÇÃO

EXEMPLO ILUSTRATIVO



Fonte: Sistemas Operacionais – R. Oliveira, A. Carissimi, S. Toscani



REFERÊNCIAS

Sistemas Operacionais. R. Oliveira, A. Carissimi, S. Toscani.

Sistemas Operacionais Modernos. A. Tanenbaum