

## Cap. 6 – Gerenciamento de Entrada e Saída

Prof. Marcelo Moreno

[moreno@ice.ufjf.br](mailto:moreno@ice.ufjf.br)



### Controladores de Dispositivos

- Componentes de dispositivos de E/S
  - mecânico
  - eletrônico
- O componente eletrônico é o controlador do dispositivo
  - pode ser capaz de tratar múltiplos dispositivos
- Tarefas do controlador
  - converter fluxo serial de bits em bloco de bytes
  - executar toda correção de erro necessária
  - tornar o bloco disponível para ser copiado para a memória principal

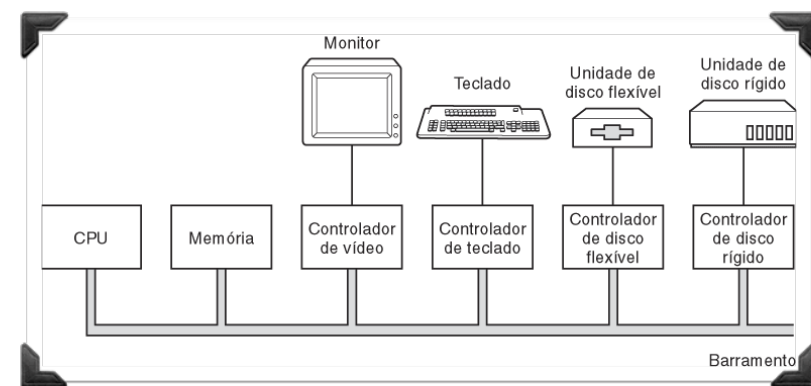


### Hardware de E/S

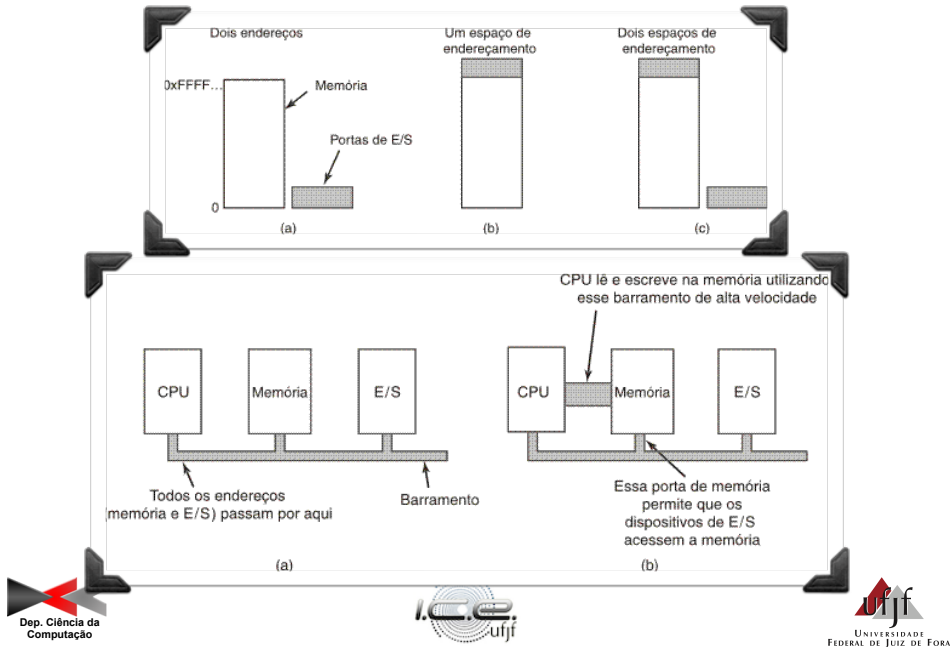
- Porta
  - Ponto de conexão disponível no host para dispositivos de E/S
- Barramento
  - Conjunto de fios e protocolo de transferência que permite que vários dispositivos sejam conectados ao host simultaneamente.
- Controlador
  - Controla a operação das portas, barramentos e dispositivos.
  - Interage com o host por meio de registradores
    - Controle, estado atual, dados



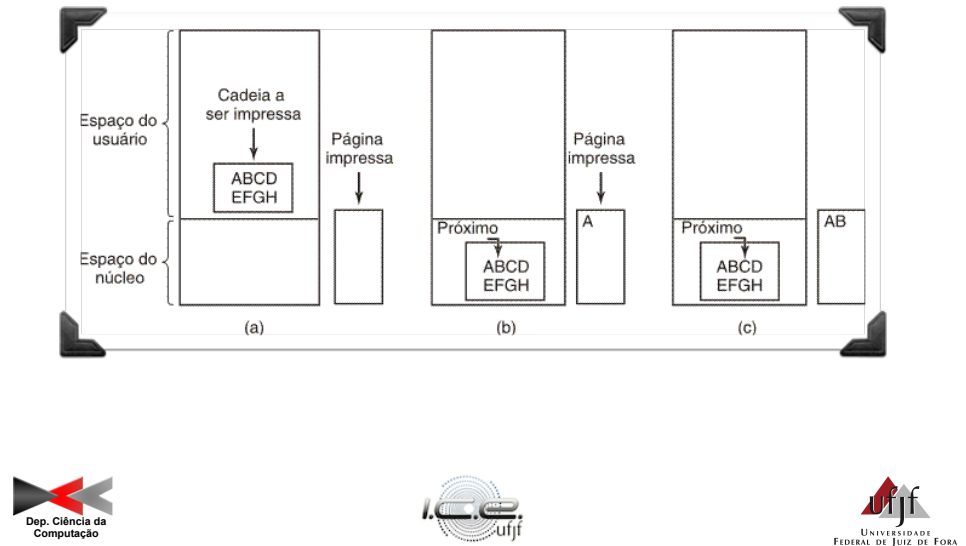
### Controladores de Dispositivos



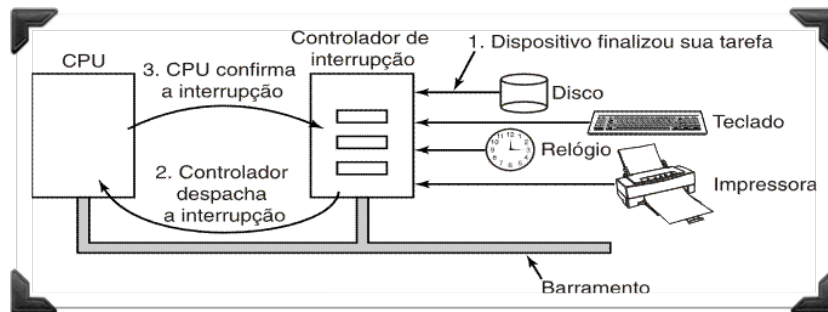
## Mapeamento de E/S em Memória



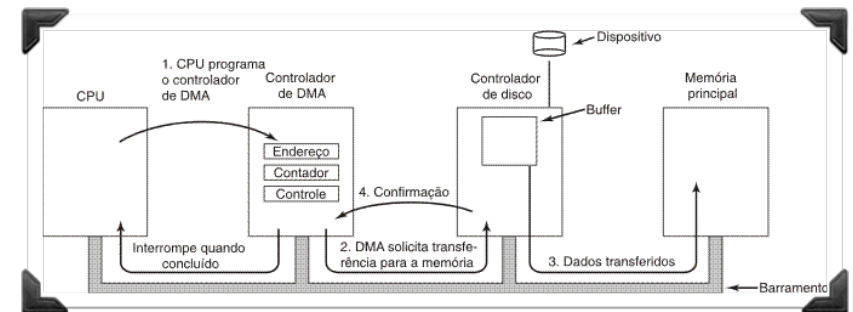
## E/S Programada (PIO)



## E/S por Interrupções



## Acesso Direto a Memória (DMA)



## E/S: PIO vs. Interrupts vs. DMA

```
copy_from_user(buffer, p, count); /* p é o buffer do núcleo */
for (i=0; i < count; i++) {      /* executa o laço para cada caractere */
    while (*printer_status_reg != READY); /* executa o laço até PRONTO */
    *printer_data_register = p[i]; /* envia um caractere para a saída */
}
return_to_user();
```

```
copy_from_user(buffer, p, count); if (count == 0) {
enable_interrupts();               unblock_user();
while (*printer_status_reg != READY); } else {
*printer_data_register = p[0];     *printer_data_register = p[i];
scheduler();                       count = count - 1;
                                   i = i + 1;
                                   }
                                   acknowledge_interrupt();
                                   return_from_interrupt();
```

```
copy_from_user(buffer, p, count); acknowledge_interrupt();
set_up_DMA_controller();           unblock_user();
scheduler();                       return_from_interrupt();
```

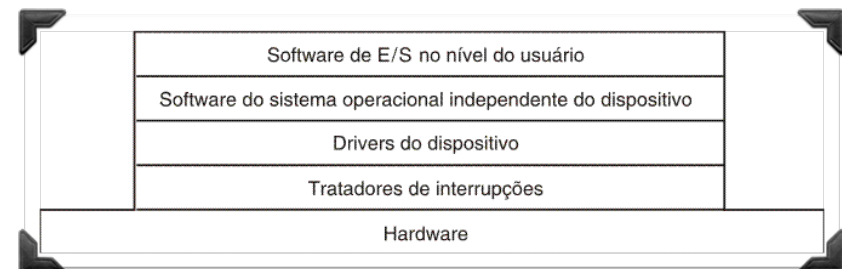
## Tratando Interrupções

- Salvar registradores que ainda não foram salvos pelo controlador de interrupção
- Estabelecer contexto para rotina de tratamento de interrupção
- Estabelecer uma pilha para a rotina de tratamento de interrupção
- Sinalizar o controlador de interrupção, reabilitar as interrupções
- Copiar os registradores de onde eles foram salvos
- Executar rotina de tratamento de interrupção
- Escolher o próximo processo a executar
- Estabelecer o contexto da MMU para o próximo processo
- Carregar os registradores do próximo processo
- Executar o próximo processo

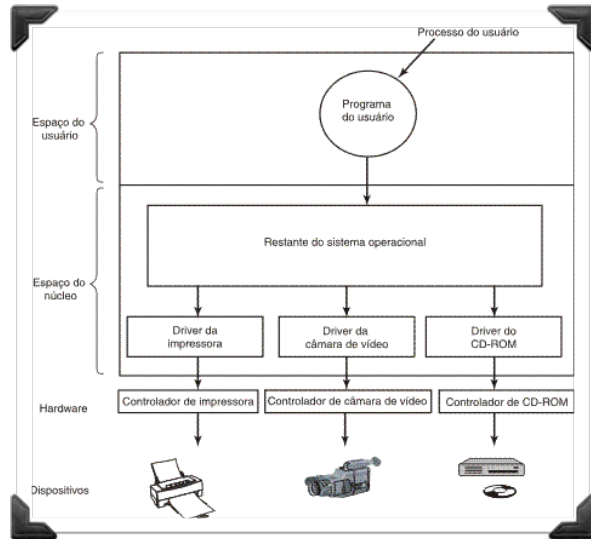
## Tratadores de Interrupções

- As interrupções devem ser escondidas o máximo possível
  - uma forma de fazer isso é bloqueando o driver que iniciou uma operação de E/S até que uma interrupção notifique que a E/S foi completada
- Rotina de tratamento de interrupção cumpre sua tarefa
  - e então desbloqueia o driver que a chamou

## Software de E/S em Camadas



# Drivers de Dispositivos



# Drivers Linux

