



Sistemas Operacionais

Entrada / Saída

Carlos Ferraz (cagf@cin.ufpe.br)

Jorge Cavalcanti Fonsêca (jcbf@cin.ufpe.br)

Tópicos

- Princípios do hardware de E/S
- Princípios do software de E/S
- Camadas do software de E/S

Princípios do Hardware de E/S

- Diversidade de dispositivos
- Características de dispositivos
- Arquitetura de E/S

Diversidade de dispositivos

Device	Data rate
Keyboard	10 bytes/sec
Mouse	100 bytes/sec
56K modem	7 KB/sec
Scanner	400 KB/sec
Digital camcorder	3.5 MB/sec
802.11g Wireless	6.75 MB/sec
52x CD-ROM	7.8 MB/sec
Fast Ethernet	12.5 MB/sec
Compact flash card	40 MB/sec
FireWire (IEEE 1394)	50 MB/sec
USB 2.0	60 MB/sec
SONET OC-12 network	78 MB/sec
SCSI Ultra 2 disk	80 MB/sec
Gigabit Ethernet	125 MB/sec
SATA disk drive	300 MB/sec
Ultrium tape	320 MB/sec
PCI bus	528 MB/sec

Taxas de dados típicas de dispositivos, redes e barramentos

Características de dispositivos

- Caracter: transferem bytes um a um. Ex.terminal
- Bloco: transferem bytes em bloco. Ex. disco
- Sequencial: acesso em ordem fixa. Ex. modem
- Acesso randômico: ordem pode ser alterada. Ex CD-ROM
- Síncrono: tempo de resposta previsível. Ex. fita
- Assíncrono: tempo de resposta imprevisível. Ex. teclado
- Compartilhável: pode ser usado por vários processos ao mesmo tempo. Ex. teclado
- Dedicado: só pode ser usado por um processo por vez. Ex. Impressora
- Read-write, read only e write-only: disco, cdrom, vídeo

Características de dispositivos

- Caracter: transferem bytes um a um. Ex.terminal
- Bloco: transferem bytes em bloco. Ex. disco
- Sequencial: acesso em ordem fixa. Ex. modem
- Acesso randômico: ordem pode ser alterada. Ex CD-ROM
- Síncrono: tempo de resposta previsível. Ex. fita
- Assíncrono: tempo de resposta imprevisível. Ex. teclado
- Compartilhável: pode ser usado por vários processos ao mesmo tempo. Ex. teclado
- Dedicado: só pode ser usado por um processo por vez. Ex. Impressora
- Read-write, read only e write-only: disco, cdrom, vídeo

Características de dispositivos

- Caracter: transferem bytes um a um. Ex.terminal
- Bloco: transferem bytes em bloco. Ex. disco
- Sequencial: acesso em ordem fixa. Ex. modem
- Acesso randômico: ordem pode ser alterada. Ex CD-ROM
- Síncrono: tempo de resposta previsível. Ex. fita
- Assíncrono: tempo de resposta imprevisível. Ex. teclado
- Compartilhável: pode ser usado por vários processos ao mesmo tempo. Ex. teclado
- Dedicado: só pode ser usado por um processo por vez. Ex. Impressora
- Read-write, read only e write-only: disco, cdrom, vídeo

Características de dispositivos

- Caracter: transferem bytes um a um. Ex.terminal
- Bloco: transferem bytes em bloco. Ex. disco
- Sequencial: acesso em ordem fixa. Ex. modem
- Acesso randômico: ordem pode ser alterada. Ex CD-ROM
- Síncrono: tempo de resposta previsível. Ex. fita
- Assíncrono: tempo de resposta imprevisível. Ex. teclado
- Compartilhável: pode ser usado por vários processos ao mesmo tempo. Ex. teclado
- Dedicado: só pode ser usado por um processo por vez. Ex. Impressora
- Read-write, read only e write-only: disco, cdrom, vídeo

Características de dispositivos

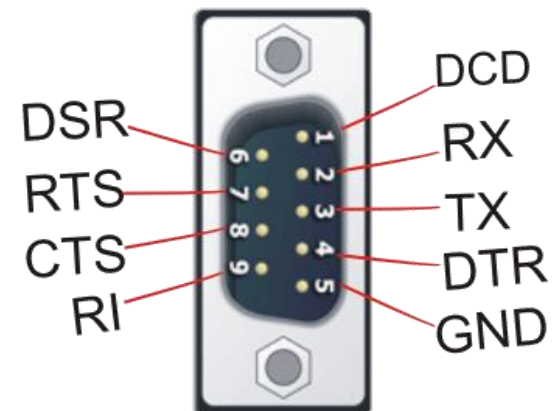
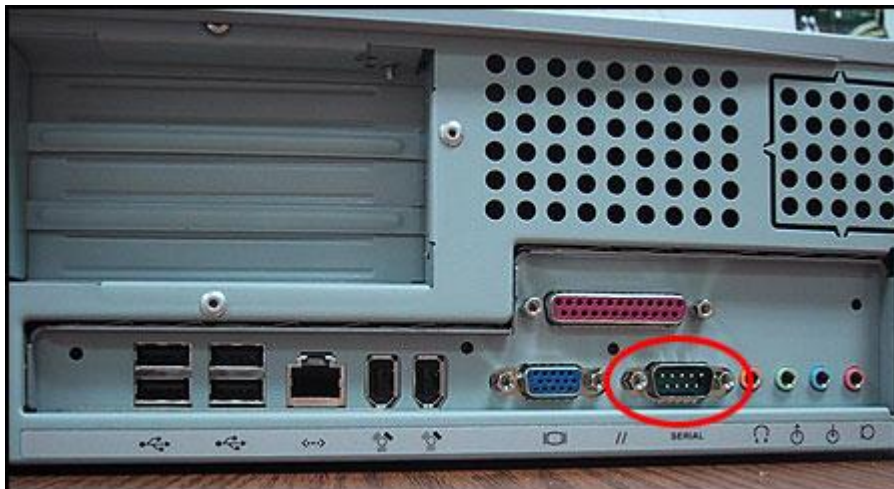
- Caracter: transferem bytes um a um. Ex.terminal
- Bloco: transferem bytes em bloco. Ex. disco
- Sequencial: acesso em ordem fixa. Ex. modem
- Acesso randômico: ordem pode ser alterada. Ex CD-ROM
- Síncrono: tempo de resposta previsível. Ex. fita
- Assíncrono: tempo de resposta imprevisível. Ex. teclado
- Compartilhável: pode ser usado por vários processos ao mesmo tempo. Ex. teclado
- Dedicado: só pode ser usado por um processo por vez. Ex. Impressora
- Read-write, read only e write-only: disco, cdrom, vídeo

Princípios do Hardware de E/S

- ✓ Diversidade de dispositivos
- ✓ Características de dispositivos
- Arquitetura de E/S

Hardware: Arquitetura de E/S

- Portas (ports)
 - Comunicação ponto a ponto
 - Ex: Porta serial e paralela

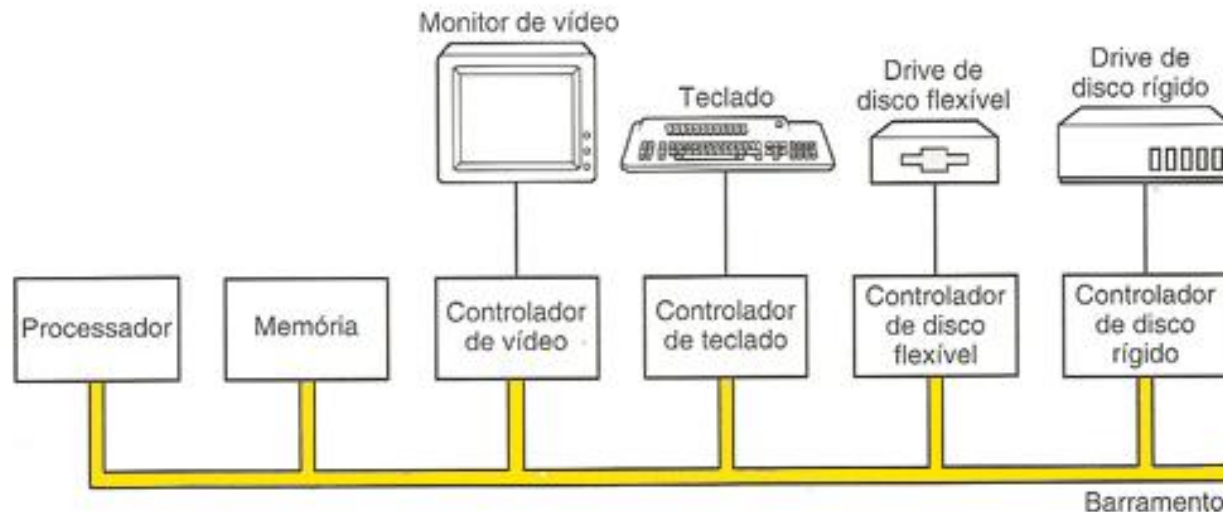


<http://www.c2o.pro.br/automacao/x834.html>

Hardware: Arquitetura de E/S

- Barramentos (bus)

- Conjunto de condutores elétricos e com um **protocolo** rígido que define como mensagens trafegam sobre esses condutores
- Permite a comunicação entre vários componentes
- **Protocolo** é um conjunto de regras que definem como as comunicações (no barramento) serão efetuadas



Arquitetura de E/S

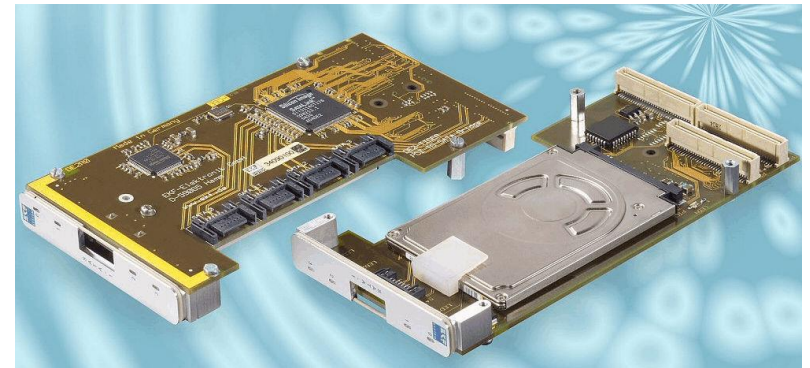
- Controladores

- Hw que controla uma porta, barramento ou dispositivo(s)

- Ex: Controlador da porta serial

- Controlador SCSI (Small Computer System Interface)

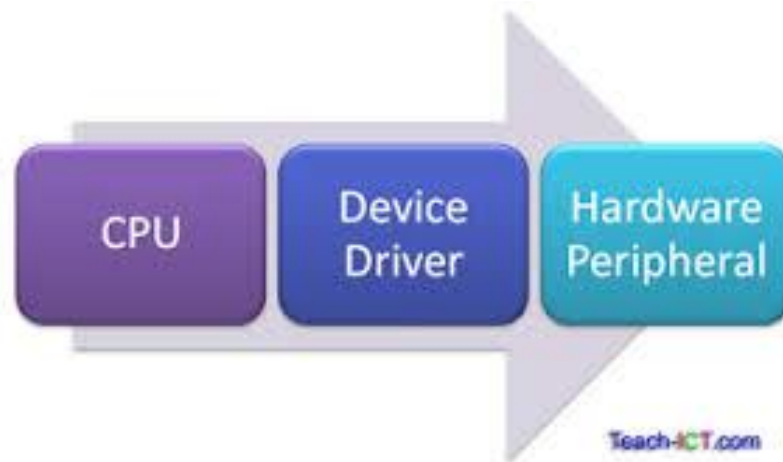
- Controlador de disco



Arquitetura de E/S

- Device Drivers

- Partes do S.O. (**software**) que fornecem uma interface de acesso uniforme para cada dispositivo
- Traduz as chamadas de alto nível (usuário) para o dispositivo específico
- Conversão de dados
- Detecção e correção de erros

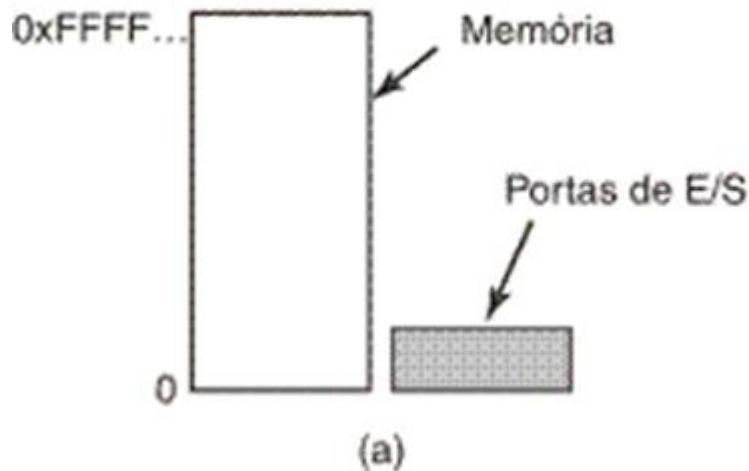


E/S: Como a CPU acessa a informação?

- **Espaço de endereçamento:** conjunto de endereços de memória que o processador consegue acessar diretamente
- **E/S isolada**
- **E/S mapeada em memória**
- **Abordagem Híbrida**
- A forma de acessar os registradores (das interfaces) dos periféricos é definida no projeto do processador:
 - Espaço único
 - Dois espaços, um deles dedicado à E/S (isolada)
- Registradores de controle
 - Nos controladores

Espaços de Memória e E/S

Dois endereços

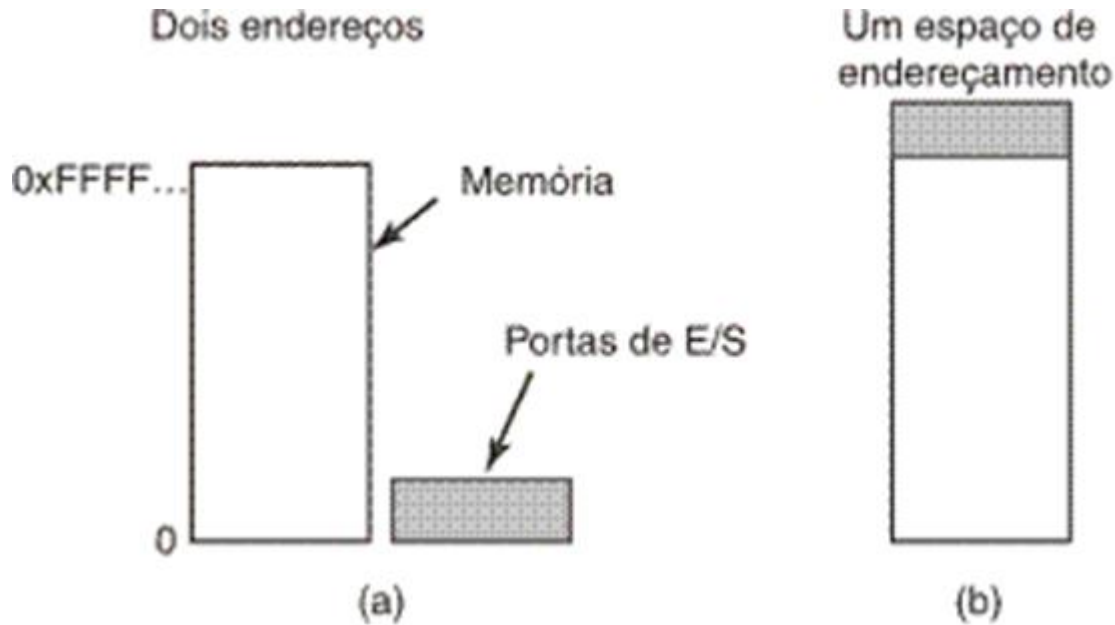


a) Espaços de memória e E/S separados - E/S isolada

– E/S isolada

- Através de instruções especiais de E/S
 - (IN REG, PORT)
- Especifica a leitura/escrita de dados numa **porta de E/S**

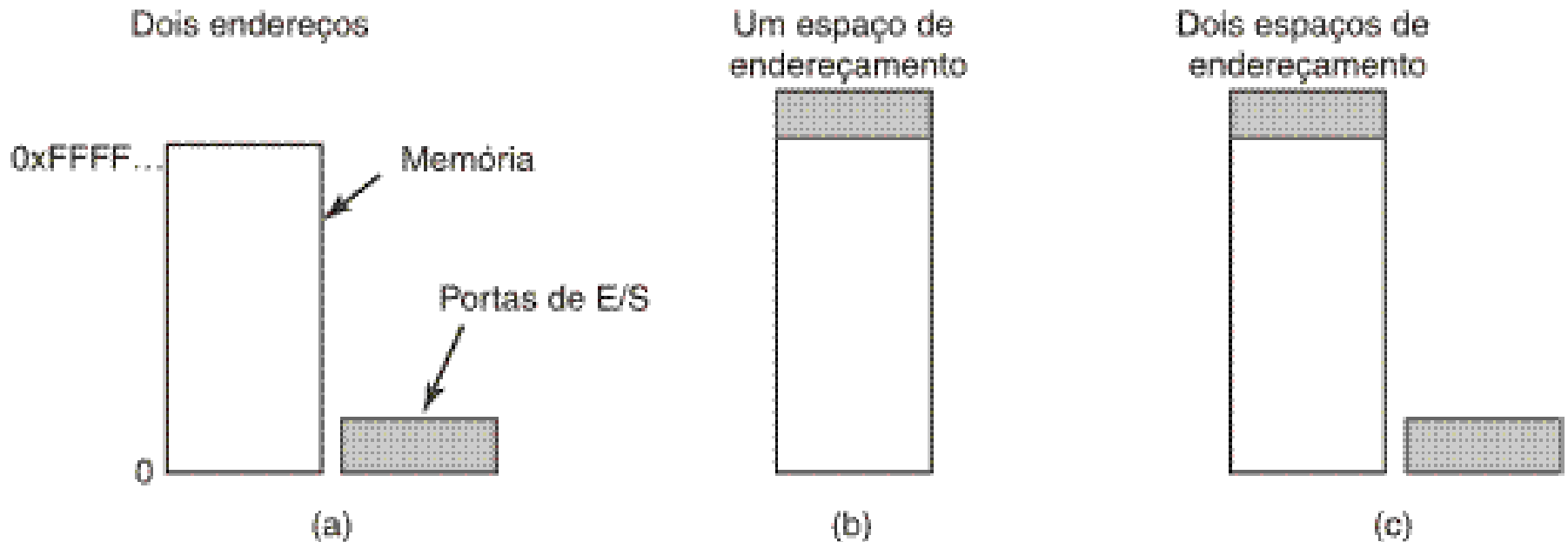
Espaços de Memória e E/S



(b) E/S mapeada na memória

- Através de instruções de leitura/escrita na memória
- Parte da memória reservada

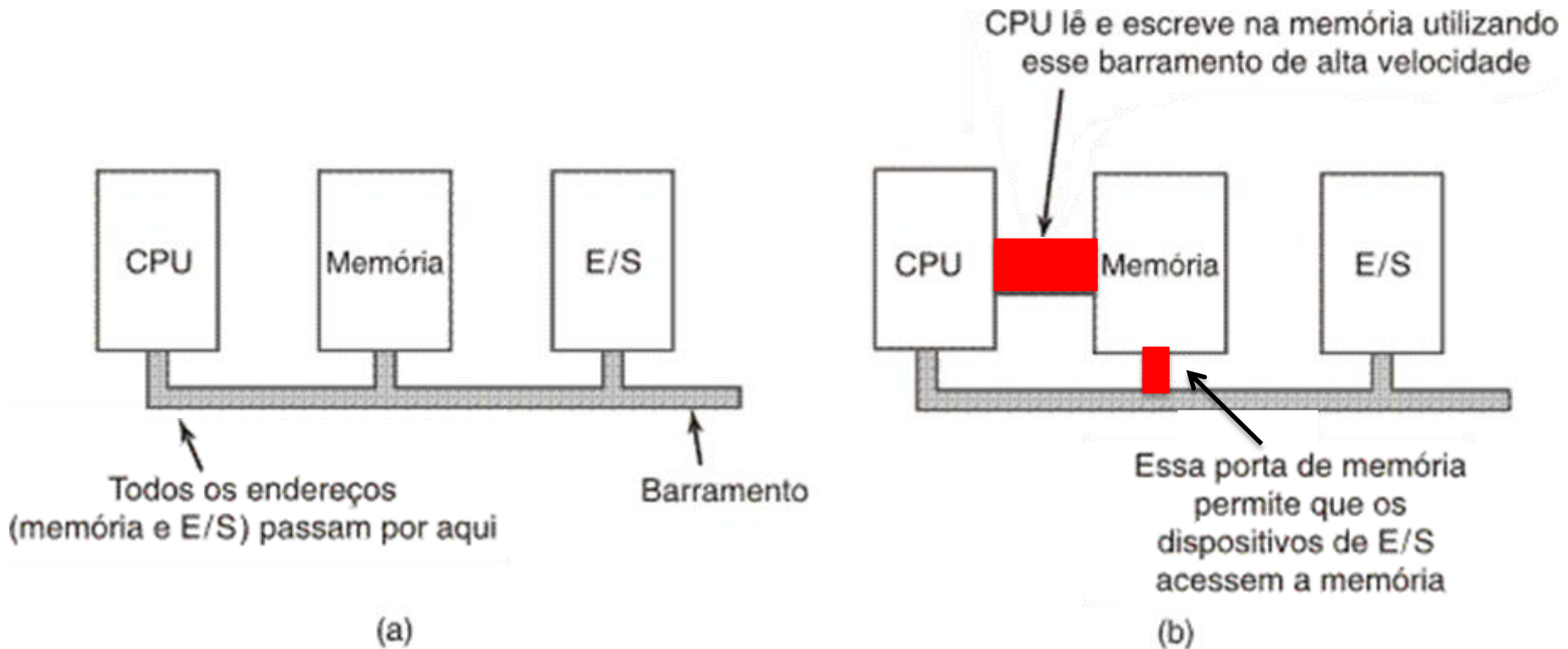
Espaços de Memória e E/S



(c) Híbrido (ex. IBM-PC):

- E/S mapeada em memória: memória de vídeo
- E/S isolada: dispositivos em geral

E/S mapeada na memória (Barramento)



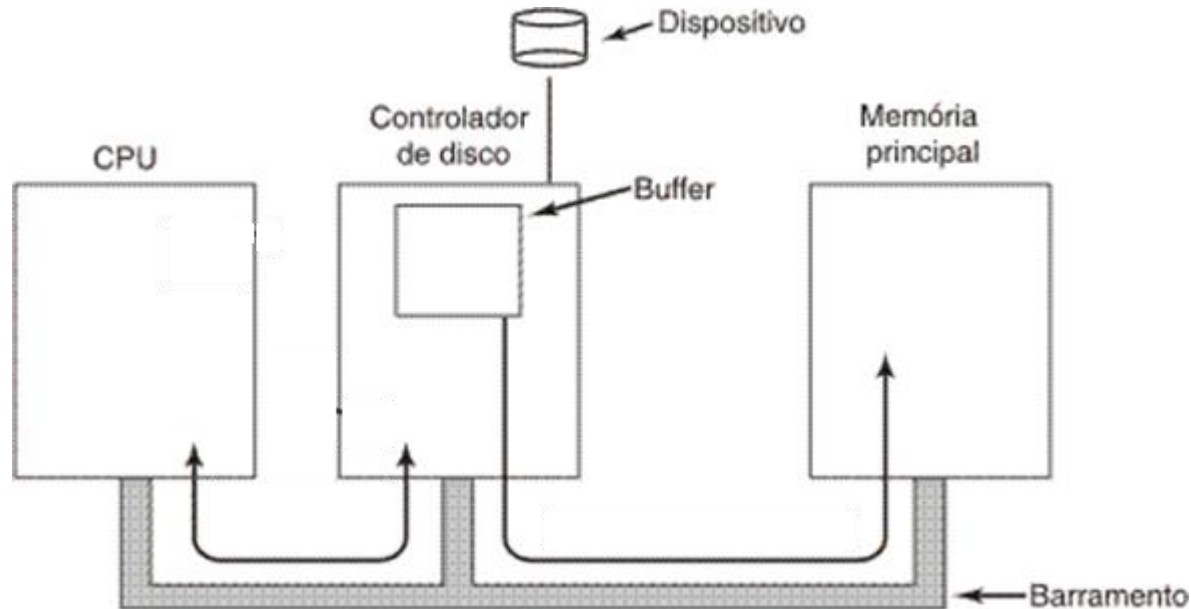
(a) Arquitetura com barramento único

(b) Arquitetura com barramento duplo (dual)

Como o processador “enxerga” a memória e os demais dispositivos ou como o processador se comunica com o seu exterior

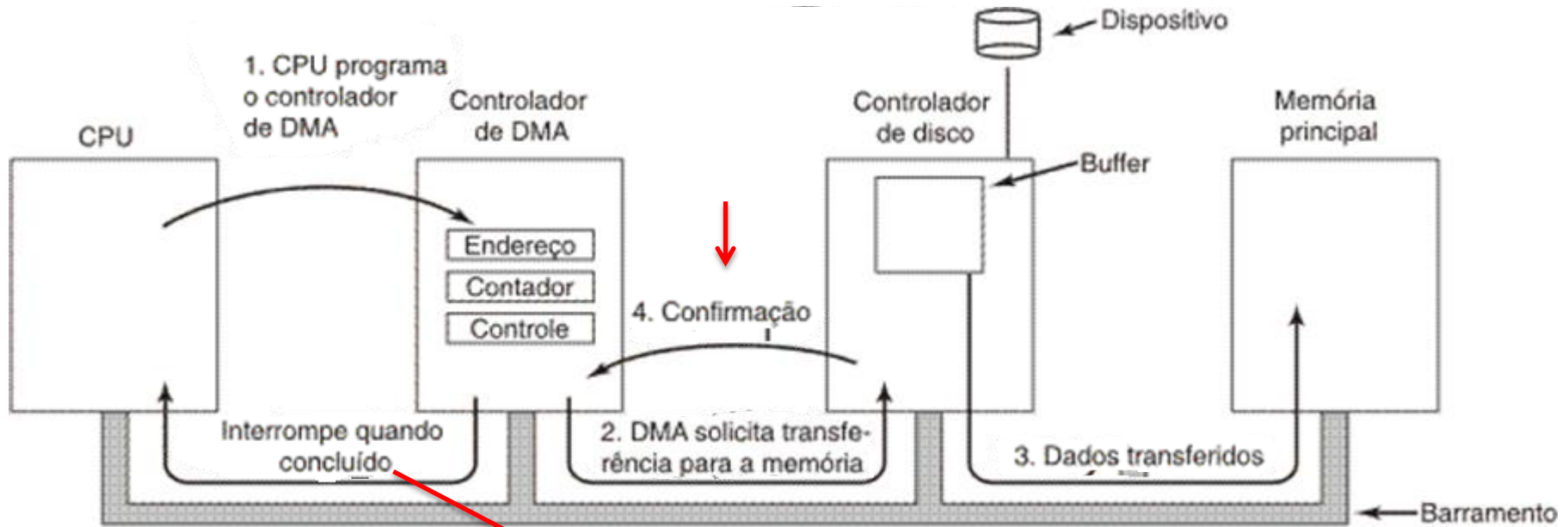
- O processador realiza operações como:
 - Ler um dado da memória
 - Escrever um dado na memória
 - Receber (ler) um dado de dispositivos de E/S
 - Enviar (escrever) dados para dispositivos de E/S
- Nas operações de **acesso à memória**, o processador escreve e lê dados, **praticamente sem intermediários**
- Nos **acessos a dispositivos de E/S**, existem circuitos intermediários, que são as **interfaces**

Operação de uma transferência



1. Controlador lê um bloco (bit a bit)
2. Controlador: checksum
3. Controlador causa interrupção
4. SO – copia bloco do buffer do controlador para a memória principal

Operação Ditona à Memória (DMA)



Quando a interrupção ocorre, **os dados já estão na memória**

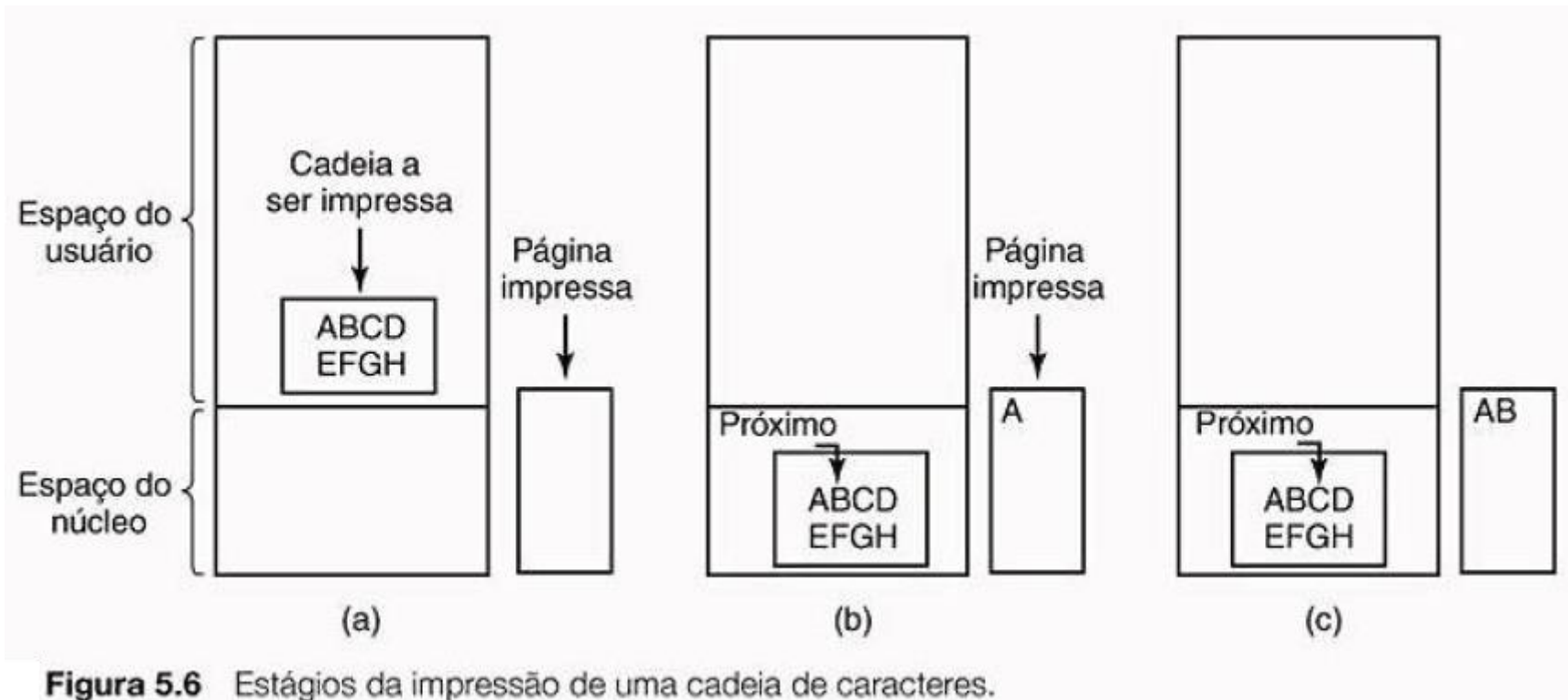
Interfaces de E/S: Diversidade

Dispositivo	Interface
Monitor	Placa de vídeo
Teclado	Interface de teclado
Alto falante	Interface de alto falante
Impressora	Interface paralela ou USB
Mouse	Interface serial, PS/2 ou USB
Disco rígido IDE	Interface IDE
Disco rígido SCSI	Interface SCSI
Joystick	Interface para jogos ou USB
Scanner	Placa de interface de scanner, paralela ou USB
Câmera digital	Interface serial, paralela ou USB

Para controlar um dispositivo de E/S, o processador precisa realizar acessos de leitura e escrita na sua interface

Como a CPU sabe que o dispositivo já executou o comando?

- E/S Programada (CPU fazendo todo o trabalho)



Como a CPU sabe que o dispositivo já executou o comando?

- E/S Programada (CPU fazendo todo o trabalho)

```
copy_from_user(buffer, p, cont);          /* 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 a impressora estar pronta*/
    *printer_data_register = p[i];        /* envia um caractere para a saída */
}
return_to_user();
```

- E/S Programada
 - CPU lê constantemente o status do controlador (*Polling* ou *Busy-waiting*)
 - Desvantagem: Espera até o fim da operação

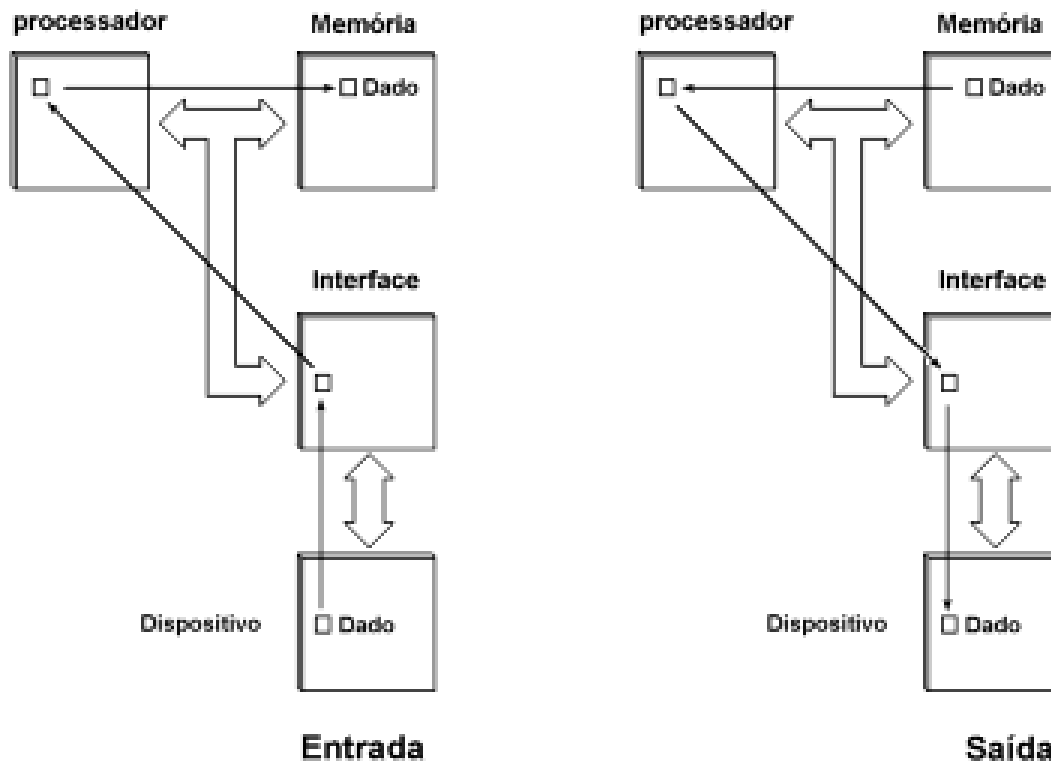
Como a CPU sabe que o dispositivo já executou o comando?

- E/S por Interrupção
 - CPU continua a executar outras operações
 - Desvantagem: interrupção para cada caractere/palavra (custo desse gerenciamento)
- E/S por DMA - Acesso Direto à Memória
 - Nesta fase de transferência não há envolvimento da CPU
 - Quando necessário, o controlador de E/S solicita ao controlador de DMA a transferência de dados de/para a memória
 - Somente ao final da transferência, a CPU é interrompida e informada da transação

```
copy_from_user(buffer, p, count);  
set_up_DMA_controller( );  
scheduler( );
```

E/S programada

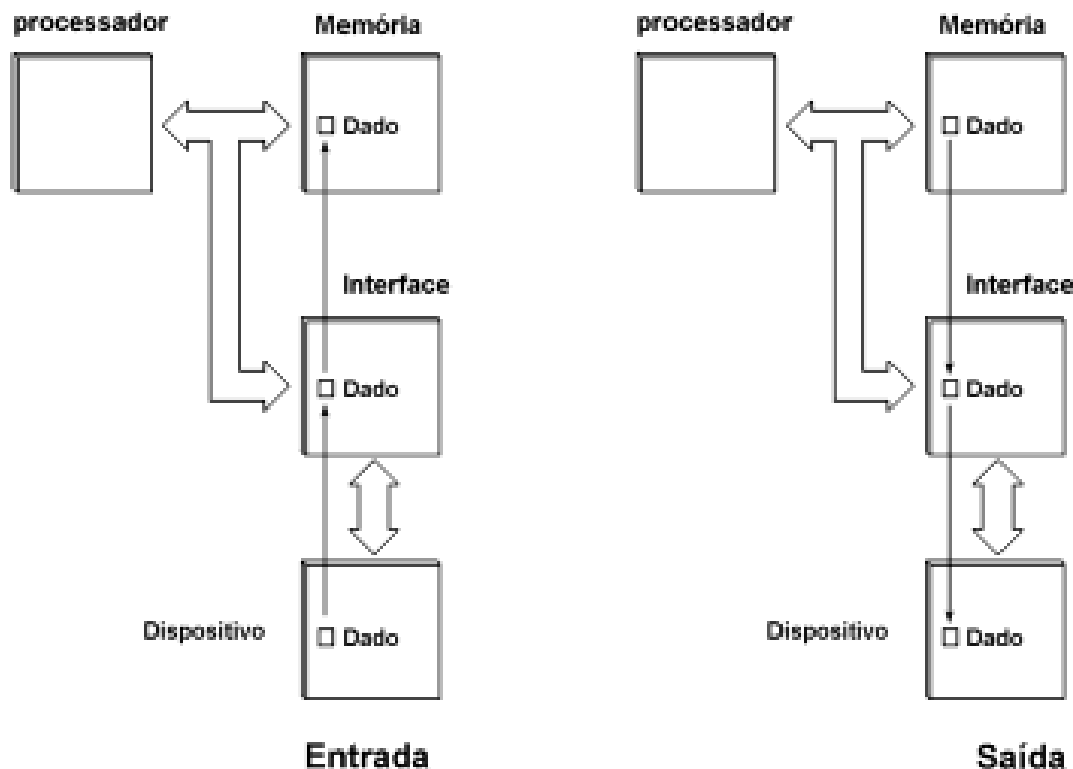
E/S Programada



- O processador controla o “transporte” dos dados entre a memória e a interface

E/S por Acesso Direto a Memória

E/S por DMA



- O processador fica momentaneamente desabilitado enquanto o controlador de DMA assume o controle dos barramentos e faz as transferências
- A grande vantagem do DMA é que o processador não precisa se ocupar diretamente da operação de recepção e transmissão de cada byte, ficando livre para executar outros processamentos
- Normalmente as interfaces que utilizam DMA, utilizam também uma **interrupção** para avisar o processador sobre o término da transferência do número de bytes pré-programado

Comunicação S.O.(CPU) – Controlador

Exemplo de comunicação com dispositivo



Entrada/Saída

- ✓ Princípios do hardware de E/S
- Princípios do software de E/S
- Camadas do software de E/S

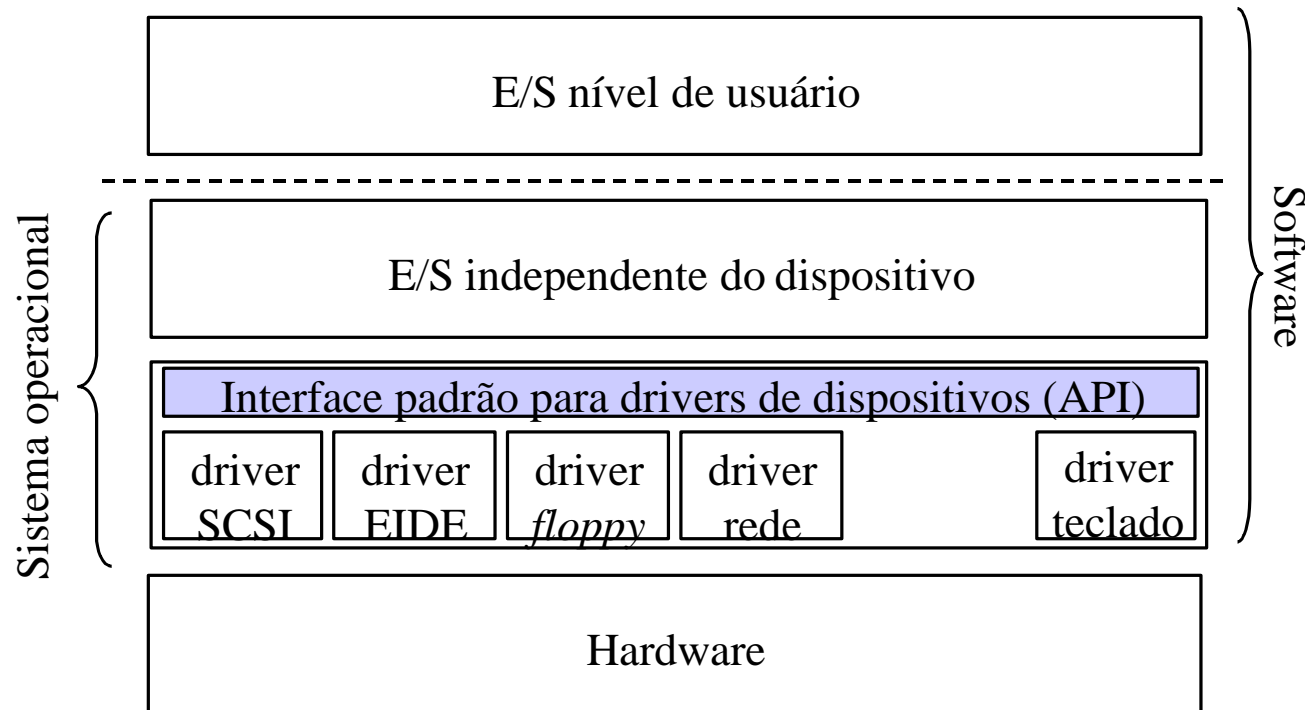
Objetivos da gerência de E/S

- Eficiência
- Uniformidade (desejável):
 - Todos dispositivos enxergados da forma mais uniforme possível
- Esconder os detalhes (estes são tratados pelas camadas de mais baixo nível)
- Fornecer abstrações genéricas: read, write, open e close

Princípios básicos do software de E/S

- Subsistema de E/S é complexo, dada a **diversidade** de periféricos
- Padronizar ao máximo para reduzir número de rotinas
 - Novos dispositivos não alteram a visão do usuário em relação ao SO
- Organizado em camadas

Estrutura do subsistema E/S



Visão Geral do software de E/S

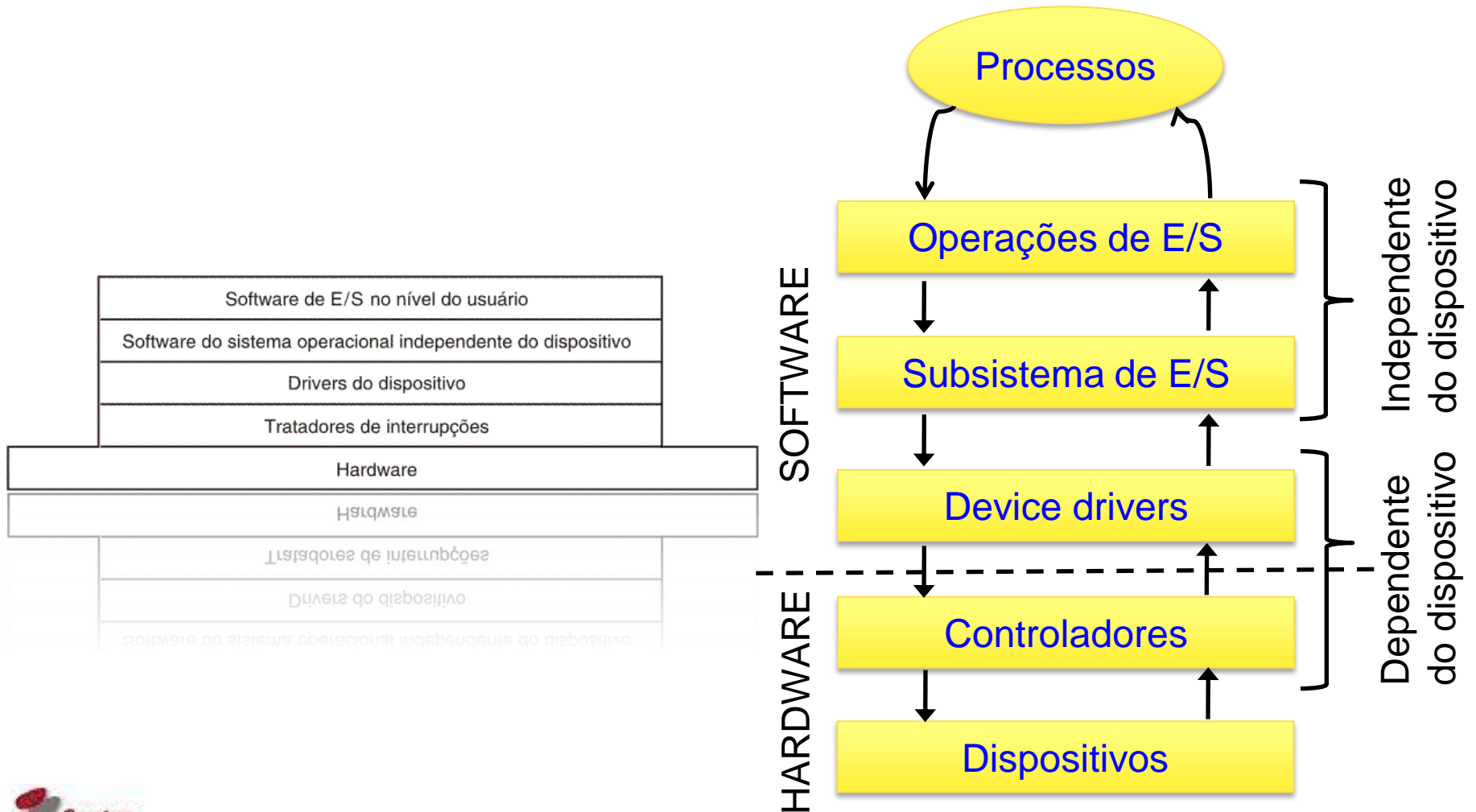
- Tratador de interrupção
 - É acionado ao final da operação de transferência
 - Aciona driver
- Driver de dispositivo
 - Recebe requisições
 - Configura (aciona) o controlador
- E/S independente de dispositivo
 - Nomes e proteção
 - bufferização
- E/S em nível de usuário
 - Chamadas de E/S

Camadas do Software de E/S

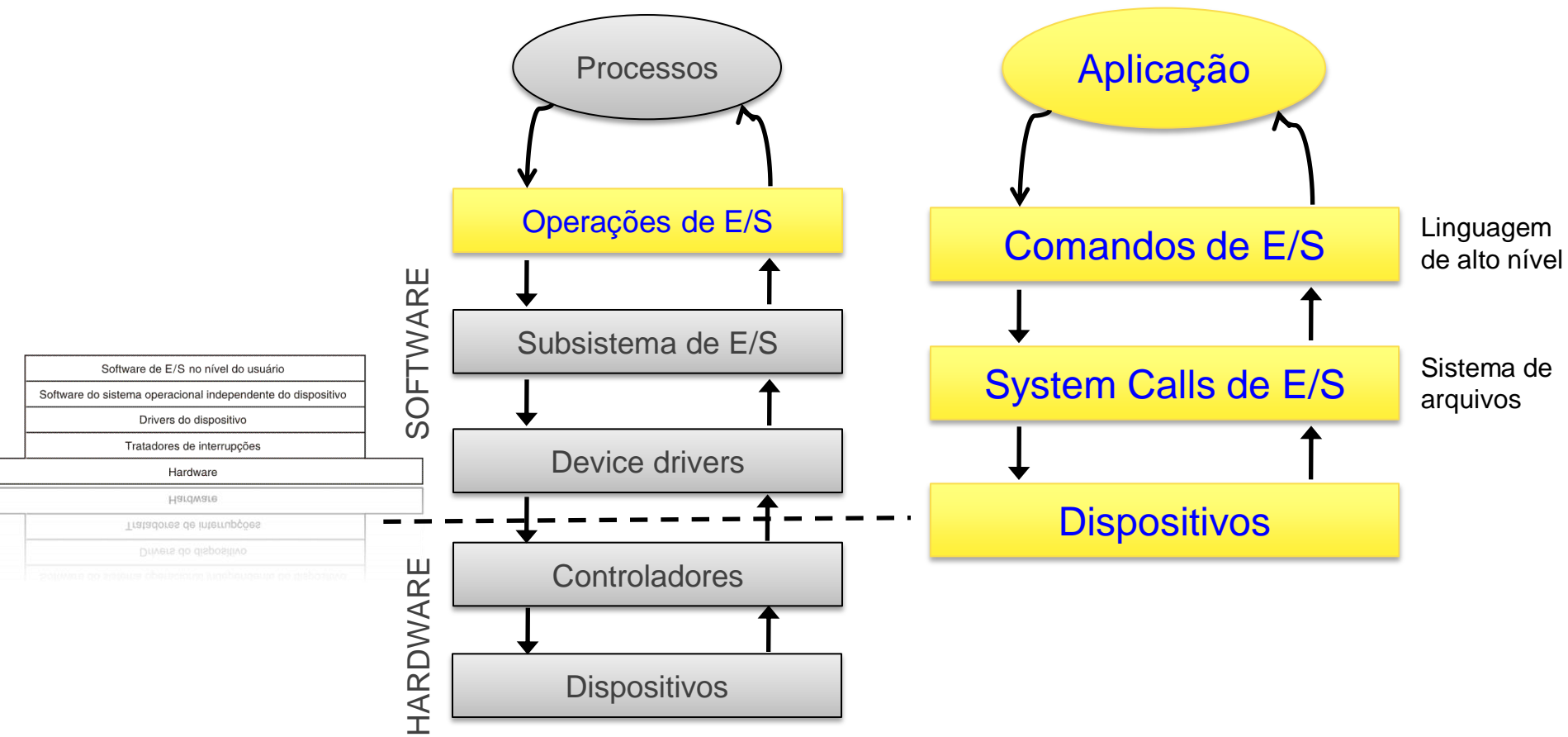


Camadas do sistema de software de E/S

Camadas do Software de E/S

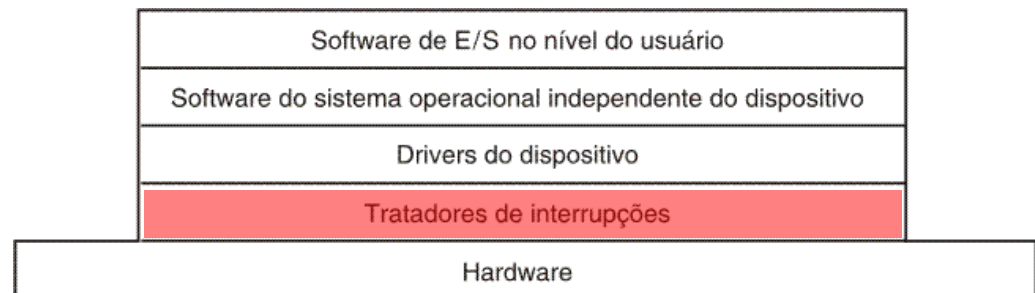


Camadas do Software de E/S

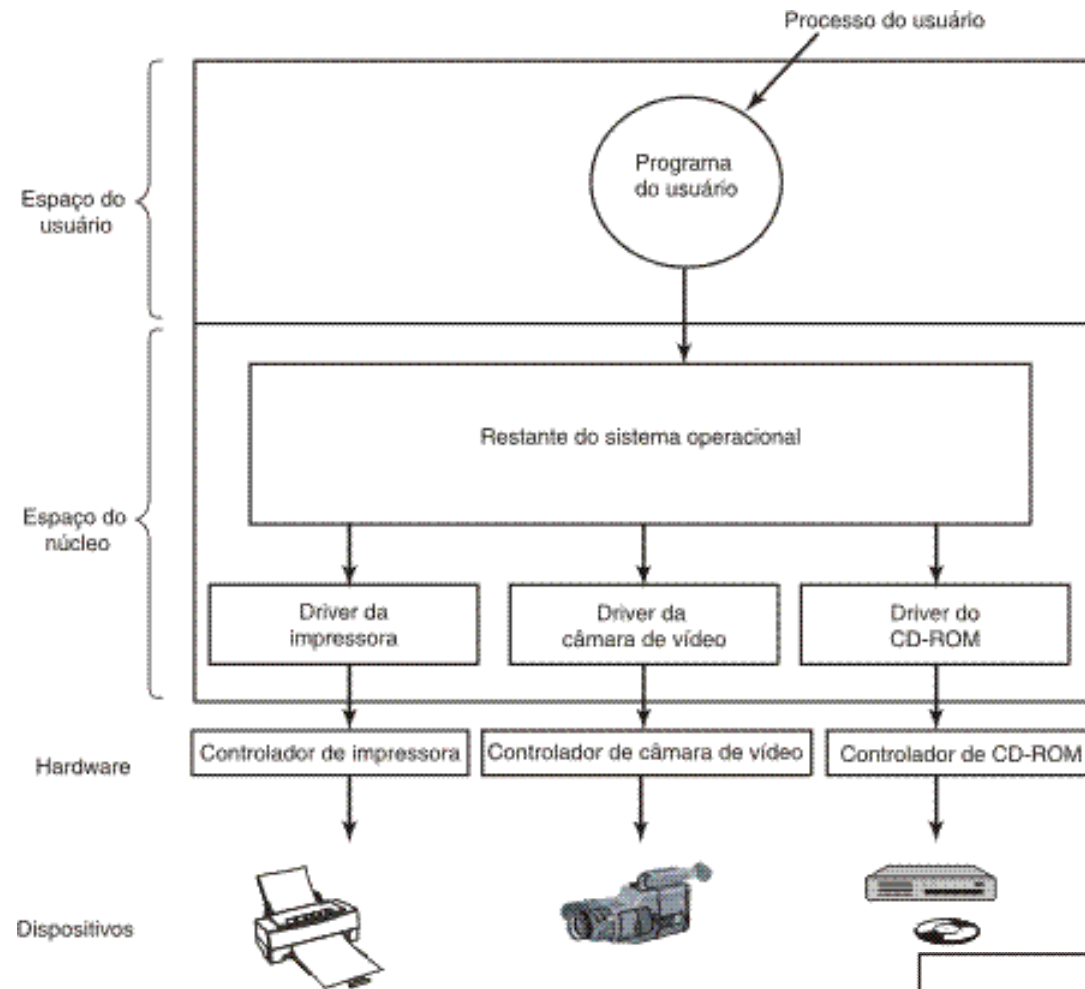


Camadas do Software de E/S Tratadores de Interrupção

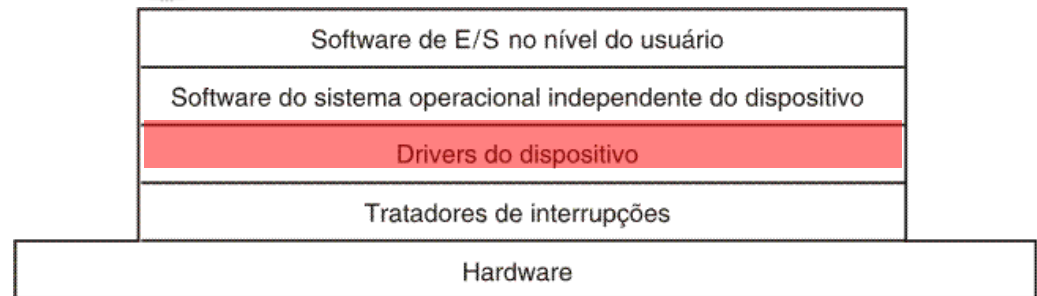
- As interrupções devem ser escondidas (transparentes) o máximo possível
 - uma forma de fazer isso é bloqueando o *driver* que iniciou uma operação de E/S até que uma interrupção notifique que a E/S foi completada
 - rotina de tratamento de interrupção cumpre sua tarefa
 - e então desbloqueia o driver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



Camadas: Drivers dos Dispositivos



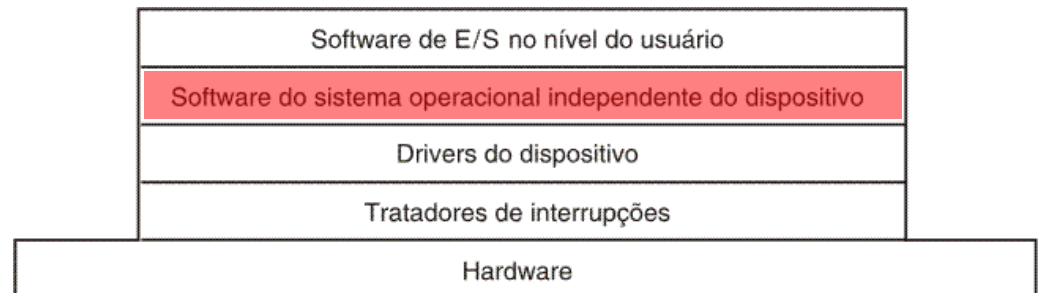
- Posição lógica dos drivers dos dispositivos
- A comunicação entre os drivers e os controladores de dispositivos é feita por meio do barramento



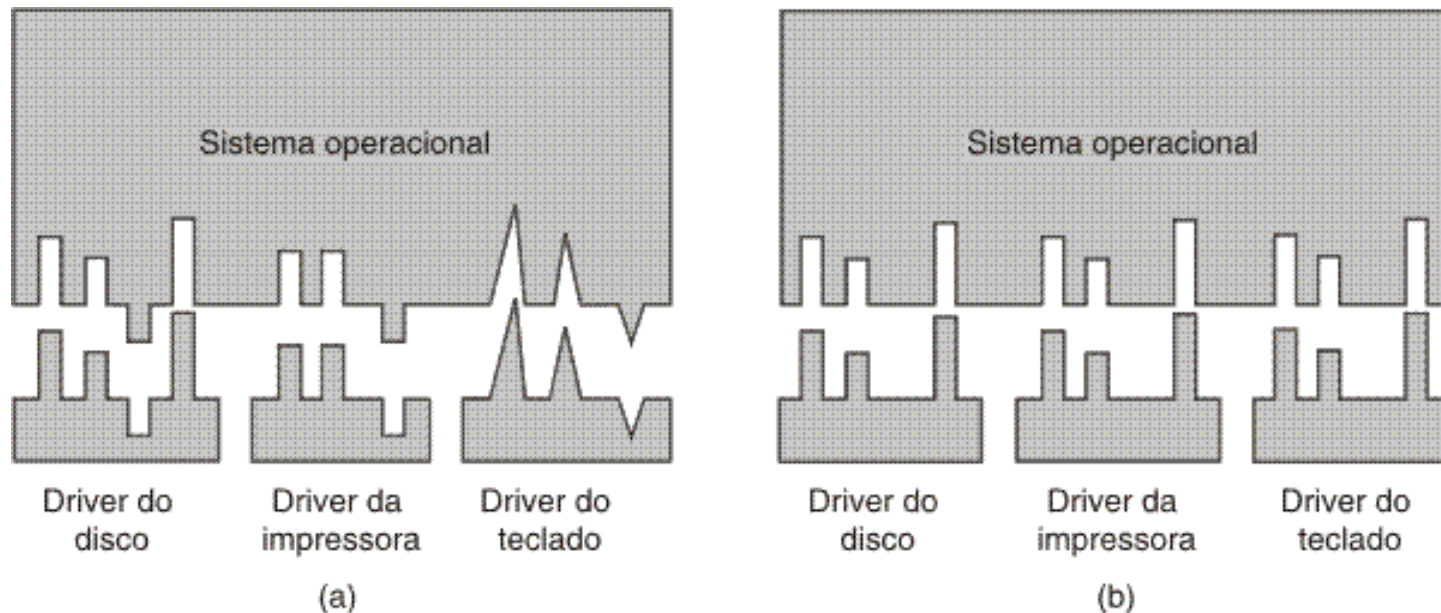
Camadas: Software de E/S Independente de Dispositivo (1)

Funções do software de E/S independente de dispositivo

Interface uniforme para os drivers dos dispositivos
Armazenamento em buffer
Relatório de erros
Alocação e liberação de dispositivos dedicados
Fornecimento de tamanho de bloco independente de dispositivo



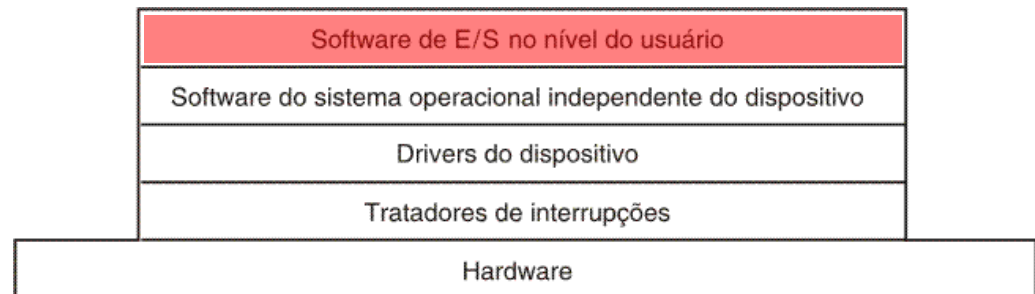
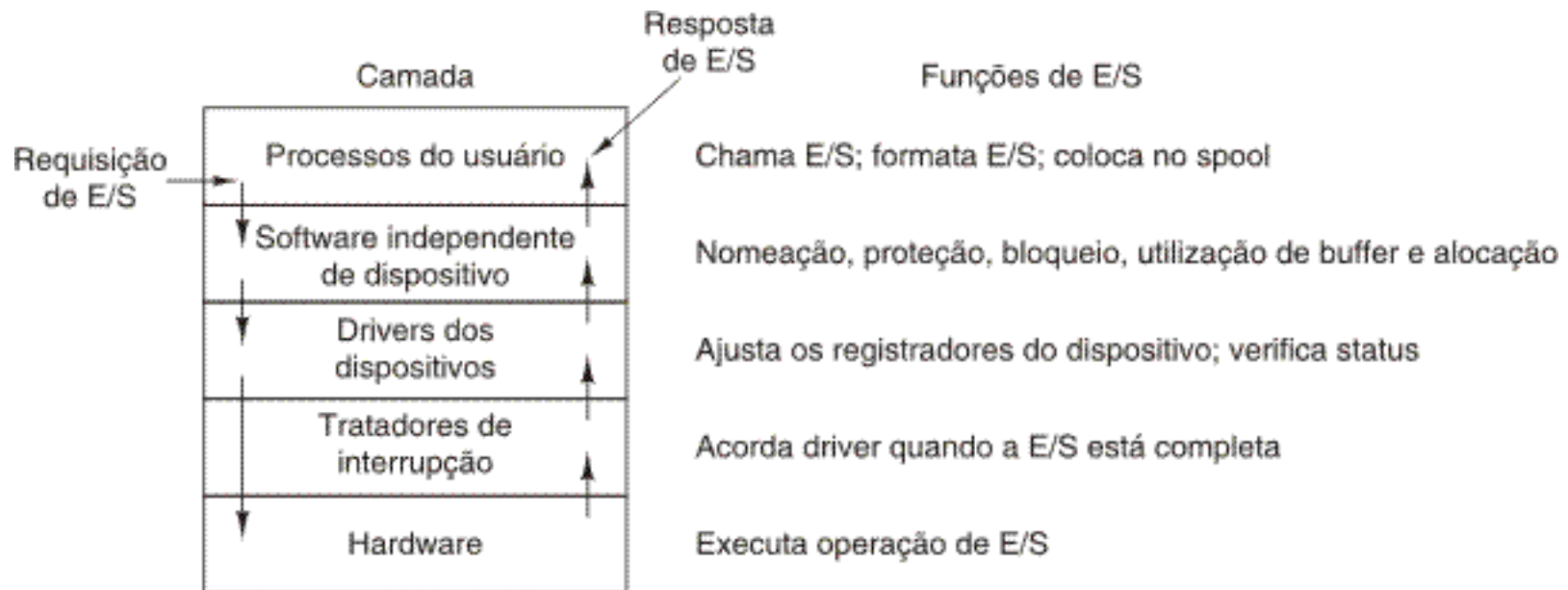
Camadas: Software de E/S Independente de Dispositivo (2)



- (a) Sem uma interface-padrão do driver
- (b) Com uma interface-padrão do driver

Camadas: Software de E/S no Espaço do Usuário

Camadas do sistema de E/S e as principais funções de cada camada



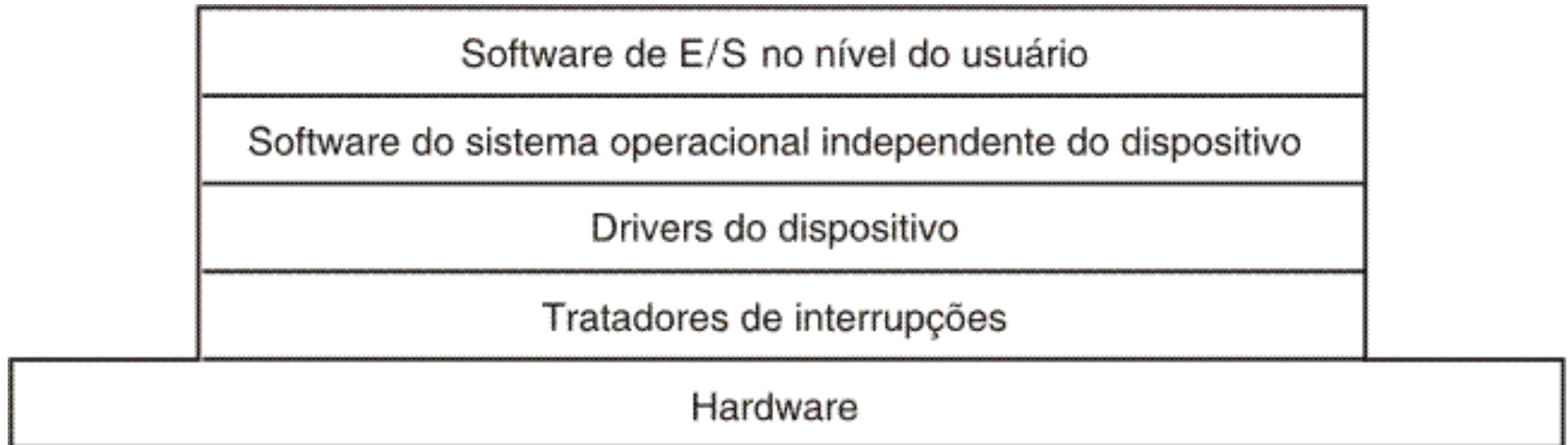
Entrada / Saída

Conclusões

Arquitetura de Entrada/Saída

- Portas (ports)
 - Comunicação ponto a ponto
- Barramentos (bus)
 - Permite a comunicação entre vários componentes
- Controladores
 - Hw que controla uma porta, barramento ou dispositivo(s)
- Device Drivers
 - Partes do S.O. que fornecem uma interface de acesso uniforme para cada dispositivo

Camadas do Software de E/S



Princípios básicos do software de E/S

- Subsistema de E/S é complexo dada a diversidade de periféricos
- Padronizar ao máximo para reduzir número de rotinas
 - Novos dispositivos não alteram a visão do usuário em relação ao SO
- Organizado em camadas



Sistemas Operacionais

Entrada / Saída

Carlos Ferraz (cagf@cin.ufpe.br)

Jorge Cavalcanti Fonsêca (jcbf@cin.ufpe.br)