



Gerenciamento de dispositivos de entrada e saída

Prof. Marcos Ribeiro Quinet de Andrade
Instituto de Ciência e Tecnologia - ICT
Universidade Federal Fluminense - UFF

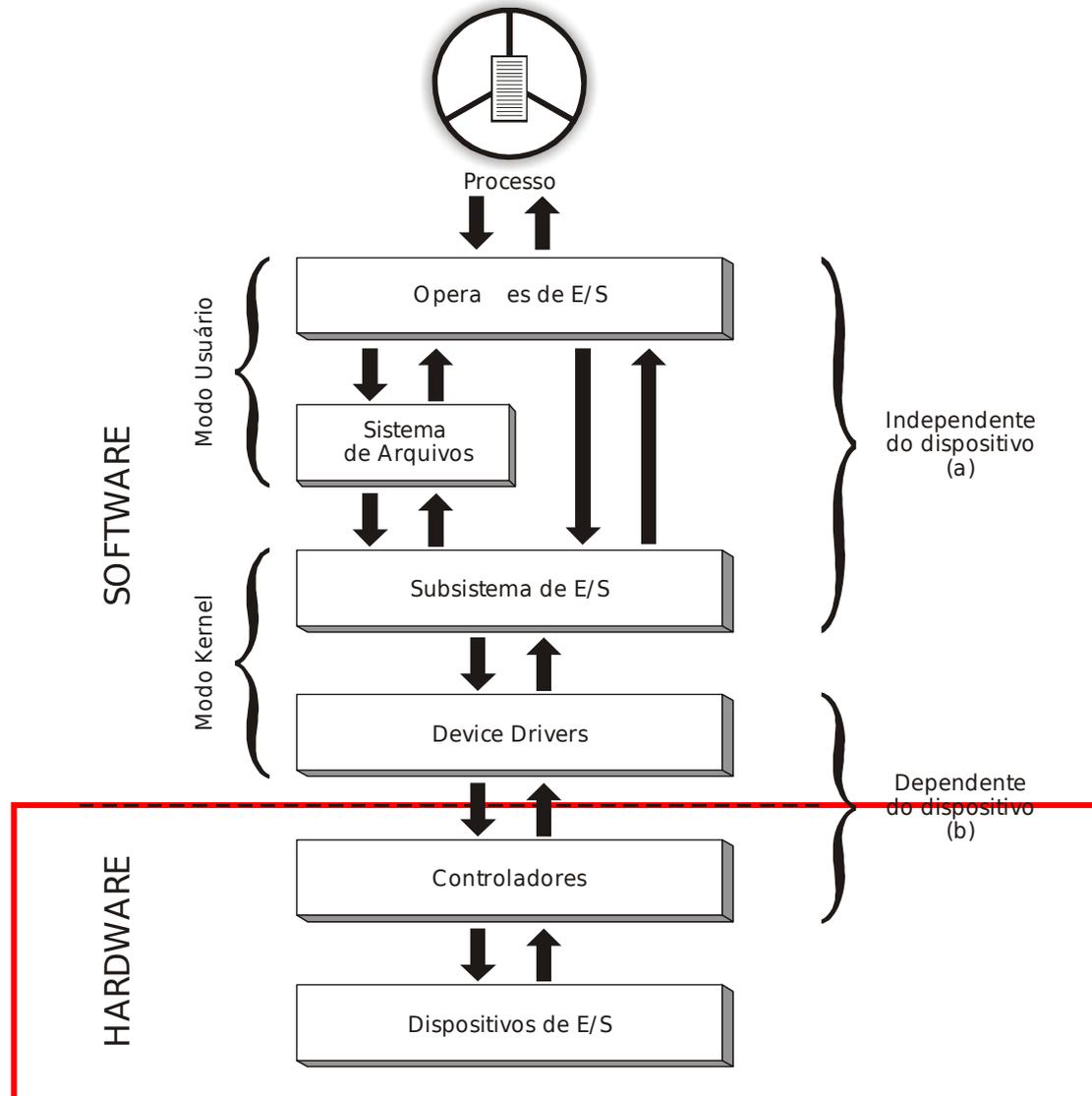
Dispositivos de Entrada e Saída

- SO pode atuar de duas maneiras diferentes (já abordadas no capítulo 1):
 - Como máquina estendida (*top-down*) - tornar uma tarefa de baixo nível mais fácil de ser realizada pelo usuário;
 - Como gerenciador de recursos (*bottom-up*) - gerenciar os dispositivos que compõem o computador;

Dispositivos de Entrada e Saída

- Funções específicas:
 - Enviar sinais para os dispositivos;
 - Atender interrupções;
 - Gerenciar comandos aceitos e funcionalidades (serviços prestados);
 - Tratar possíveis erros;
 - Prover interface entre os dispositivos e o sistema;
- Princípios divididos em camadas:
 - Hardware;
 - Software;

Dispositivos de Entrada e Saída



Dispositivos de E/S

Princípios de Hardware

- Podem ser classificados em duas categorias:
 - Dispositivos baseados em bloco: informação é armazenada em blocos de tamanho fixo, cada um com um endereço próprio; todas as unidades de transferência são em unidades de um ou mais blocos inteiros (consecutivos)
 - Tamanho varia entre 512 bytes e 32.768 bytes;
 - Permitem leitura e escrita independentemente de outros dispositivos;
 - Permitem operações de posicionamento do meio de armazenamento de dados;
 - Ex.: discos rígidos, pen drives, mídias ópticas;

Dispositivos de E/S

Princípios de Hardware

- Dispositivos baseados em caracter: aceita uma sequência de caracteres, sem se importar com a estrutura de blocos; informação não é endereçável e não possuem operações de posicionamento;
- Ex.: impressoras, interfaces de rede (placas de rede), mouses, placas de som;

Dispositivos de E/S

Princípios de Hardware

- Classificação não é perfeita, pois alguns dispositivos não se encaixam em nenhuma das duas categorias:
 - *Clocks*: provocam interrupções em intervalos definidos; não são endereçáveis por blocos nem enviam ou recebem fluxos de caracteres
- Classificação auxilia na obtenção de independência ao dispositivo;
 - Parte dependente está a cargo dos *drivers*, que são os softwares que controlam o acionamento dos dispositivos;

Dispositivos de E/S

Princípios de Hardware

- Os dispositivos de E/S podem apresentar uma grande variedade de velocidade; fica a cargo do software operar corretamente com taxas de transferências de dados de diferentes ordens de magnitude

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

Dispositivos de E/S

Princípios de Hardware

- Dispositivos de E/S possuem basicamente dois componentes:
 - Mecânico → o dispositivo propriamente dito;
 - Eletrônico → controladores ou adaptadores (placas);
- O dispositivo (periférico) e a controladora se comunicam por meio de uma **interface**:
 - Serial ou paralela;
 - Barramentos: IDE, ISA, SCSI, AGP, USB, PCI, etc.

Dispositivos de E/S

Princípios de Hardware

- Cada controladora possui um conjunto de registradores de controle, que são utilizados na comunicação com a UCP; dessa maneira, o SO pode ordenar o dispositivo para realizar tarefas, como:
 - entregar ou aceitar dados;
 - ligar e desligar;
 - etc.
- Além dos registradores, alguns dispositivos possuem um *buffer* de dados:
 - Ex.: placa de vídeo; algumas impressoras;
- SO gerencia, utilizando os *drivers*, os dispositivos de E/S escrevendo/lendo nos/dos registradores/*buffers*
 - Comunicação em baixo nível - instruções em Assembler;
 - Enviar comandos para os dispositivos;
 - Saber o estado dos dispositivos;

Mapeamento de E/S na Memória

- A UCP se comunica com esses registradores de controle e os buffers através de um dos métodos a seguir:
 - **Porta**: cada registrador de controle possui um número de porta de E/S de 8 ou 16 bits;
 - Instrução em *Assembler* para acessar os registradores;
 - Espaço de endereçamento diferente para a memória e para os dispositivos de E/S;
 - Este primeiro método foi introduzido pelos *Mainframes* IBM;
 - O conjunto de portas de E/S forma o **espaço de endereçamento de portas de E/S**; apenas o SO pode usá-las, ficando protegidas contra acessos indevidos dos programas de usuários

Mapeamento de E/S na Memória

- Exemplos:
 - IN REG, PORT → Permite a UCP ler no registrador de controle PORT e armazenar o resultado no registrador REG da UCP;
 - OUT REG, PORT → Permite a UCP escrever o conteúdo de REG para o registrador de controle PORT.

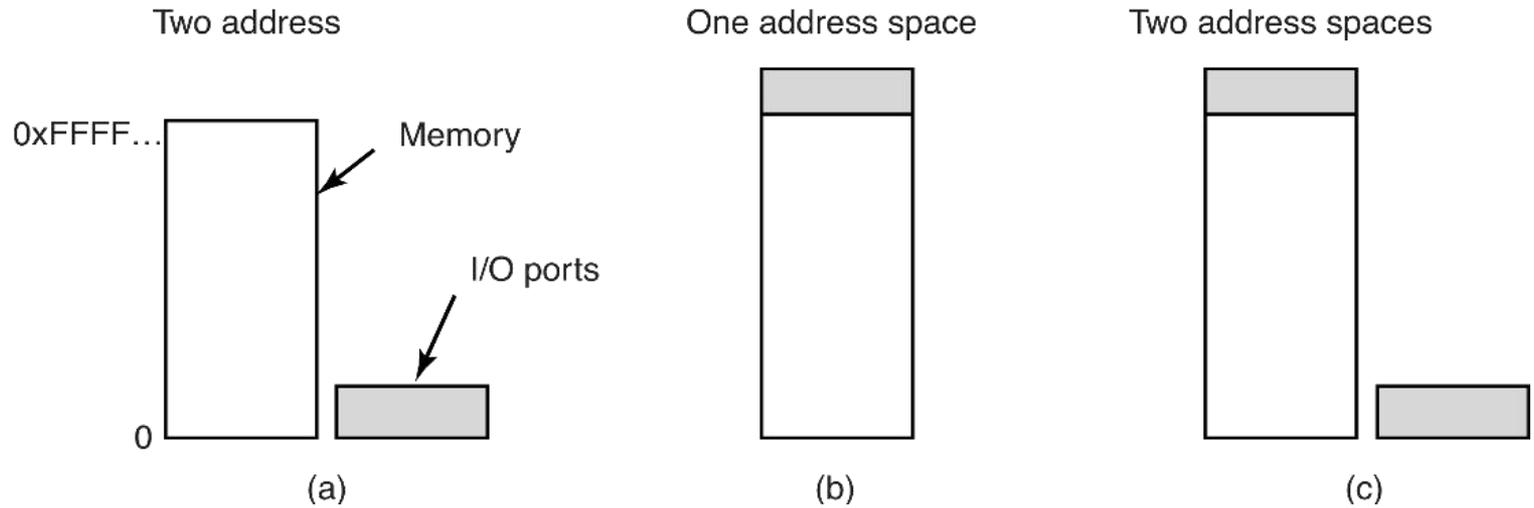
Mapeamento de E/S na Memória

- O segundo método de comunicação com os registradores de controle foi apresentado com o surgimento do PDP-11:
 - **Memory-mapped (mapeada na memória)**: mapear os registradores de controle em espaços de memória;
 - Cada registrador é mapeado em um endereço da memória;
 - Em geral, os endereços estão no topo da memória protegidos em endereços não utilizados por processos;
 - Uso de linguagem de alto nível, já que registradores são apenas variáveis na memória;
 - SOs utilizam essa estratégia para os dispositivos de vídeo;

Mapeamento de E/S na Memória

- O terceiro método é implementado através de uma estratégia híbrida:
 - Registradores de controle são mapeados para portas de E/S;
 - *Buffers* de dados de E/S são mapeados para a memória;
 - Exemplo: Pentium - endereços de 640k a 1M para os *buffers* de dados e as portas de E/S de 0 a 64k - 1;

Mapeamento de E/S na Memória



- (a) Espaços de memória e E/S separados;
- (b) E/S mapeada na memória;
- (c) Modelo híbrido

Mapeamento de E/S na Memória

- Como funciona a comunicação da UCP com os dispositivos nestes três métodos?
 - Quando a UCP deseja ler uma palavra (da memória ou de uma porta de E/S), ela coloca o endereço que ela está desejando no barramento de endereço e manda um comando **READ** no barramento de controle;
 - Uma segunda linha de sinal é usada para informar se o espaço requisitado é de E/S ou de memória; o dispositivo correspondente responderá a requisição;
 - Essa comunicação pode ser controlada pela própria UCP ou pelo DMA;

Dispositivos de E/S

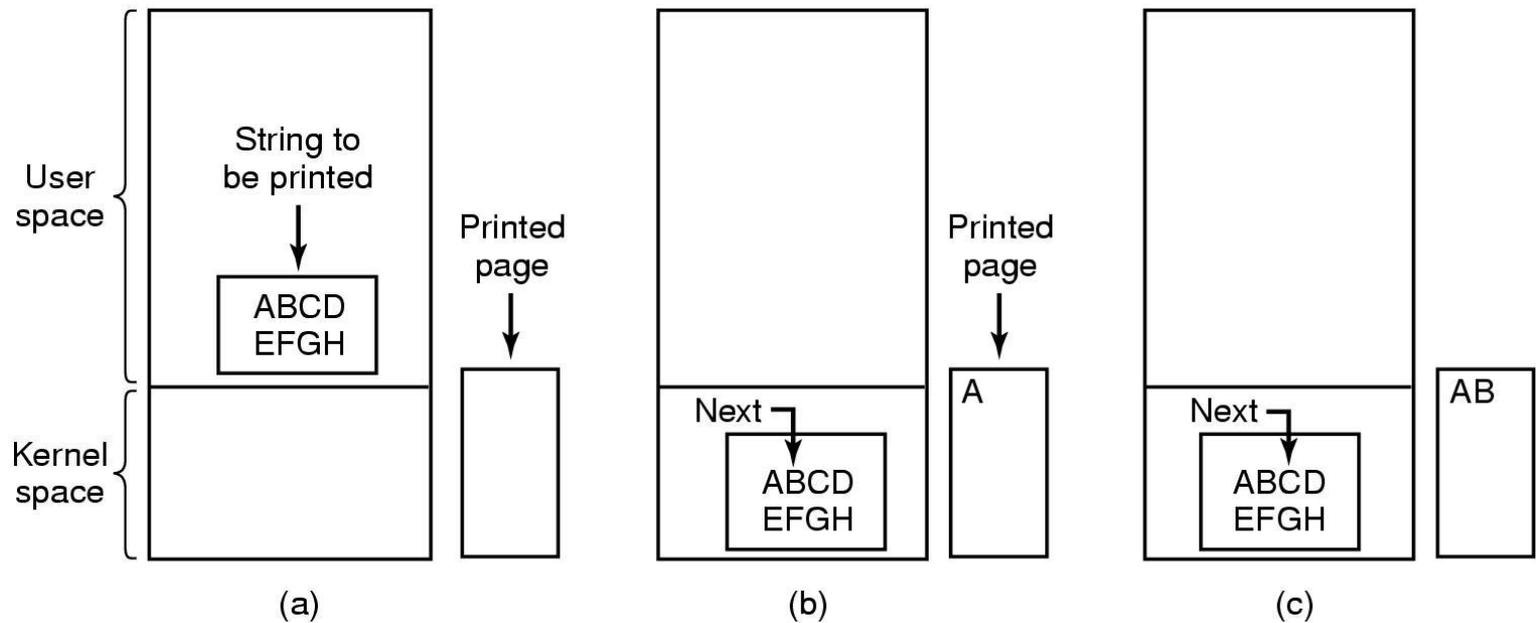
Maneiras de realizar E/S

- E/S programada;
 - Mais usada em sistemas embarcados/embutidos;
- E/S orientada à interrupção;
- E/S com uso da DMA;

Dispositivos de E/S

Maneiras de realizar E/S

- E/S programada: passos para impressão de uma cadeia de caracteres (laço até que toda a cadeia tenha sido impressa);



Dispositivos de E/S

Maneiras de realizar E/S

- E/S programada:
 - Desvantagem:
 - UCP é ocupada o tempo todo até que a E/S seja feita;
 - UCP continuamente verifica se o dispositivo está pronto para aceitar outro caracter espera ocupada;

Dispositivos de E/S

Maneiras de realizar E/S

- E/S orientada à interrupção:
 - No caso da impressão, a impressora não armazena os caracteres;
 - Quando a impressora está pronta para receber outros caracteres, gera uma interrupção;
 - Processo é bloqueado;

Dispositivos de E/S

Maneiras de realizar E/S

- E/S com uso da DMA:
 - DMA executa E/S programada controladora de DMA faz todo o trabalho ao invés da UCP;
 - Redução do número de interrupções;
 - Desvantagem:
 - DMA é mais lenta que a UCP;

Dispositivos de E/S

Maneiras de realizar E/S

- DMA (*Direct Access Memory*) acesso direto à memória:
 - Presente principalmente em dispositivos baseados em bloco discos;
 - Controladora integrada à controladora dos discos;
 - Pode estar na placa-mãe e servir vários dispositivos controladora de DMA independente do dispositivo;
 - DMA tem acesso ao barramento do sistema independentemente da UCP;

Dispositivos de E/S

Princípios de Hardware

- DMA contém vários registradores que podem ser lidos e escritos pela UCP:
 - Registrador de endereço de memória;
 - Registrador contador de bytes;
 - Registrador (es) de controle;
 - Porta de E/S em uso;
 - Tipo da transferência (leitura ou escrita);
 - Unidade de transferência (byte ou palavra);
 - Número de bytes a ser transferido;

Dispositivos de E/S

Princípios de Hardware

- Sem DMA: Leitura de um bloco de dados em um disco:
 - Controladora do dispositivo lê bloco (bit a bit) a partir do endereço fornecido pela UCP;
 - Dados são armazenados no *buffer* da controladora do dispositivo;
 - Controladora do dispositivo checa consistência dos dados;
 - Controladora do dispositivo gera interrupção;
 - SO lê (em um *loop*) os dados do *buffer* da controladora do dispositivo e armazena no endereço de memória fornecido pela UCP;

Dispositivos de E/S

Princípios de Hardware

- Com DMA: Leitura de um bloco de dados em um disco: UCP controla
 - 1. Além do endereço a ser lido, a UCP fornece à controladora de DMA duas outras informações: endereço na RAM para onde transferir os dados e o número de bytes a ser transferido;
 - 2. Controladora de DMA envia dados para a controladora do dispositivo;
 - Controladora do dispositivo lê o bloco de dados e o armazena em seu *buffer*, verificando consistência;
 - 3. Controladora do dispositivo copia os dados para RAM no endereço especificado na DMA (modo direto);

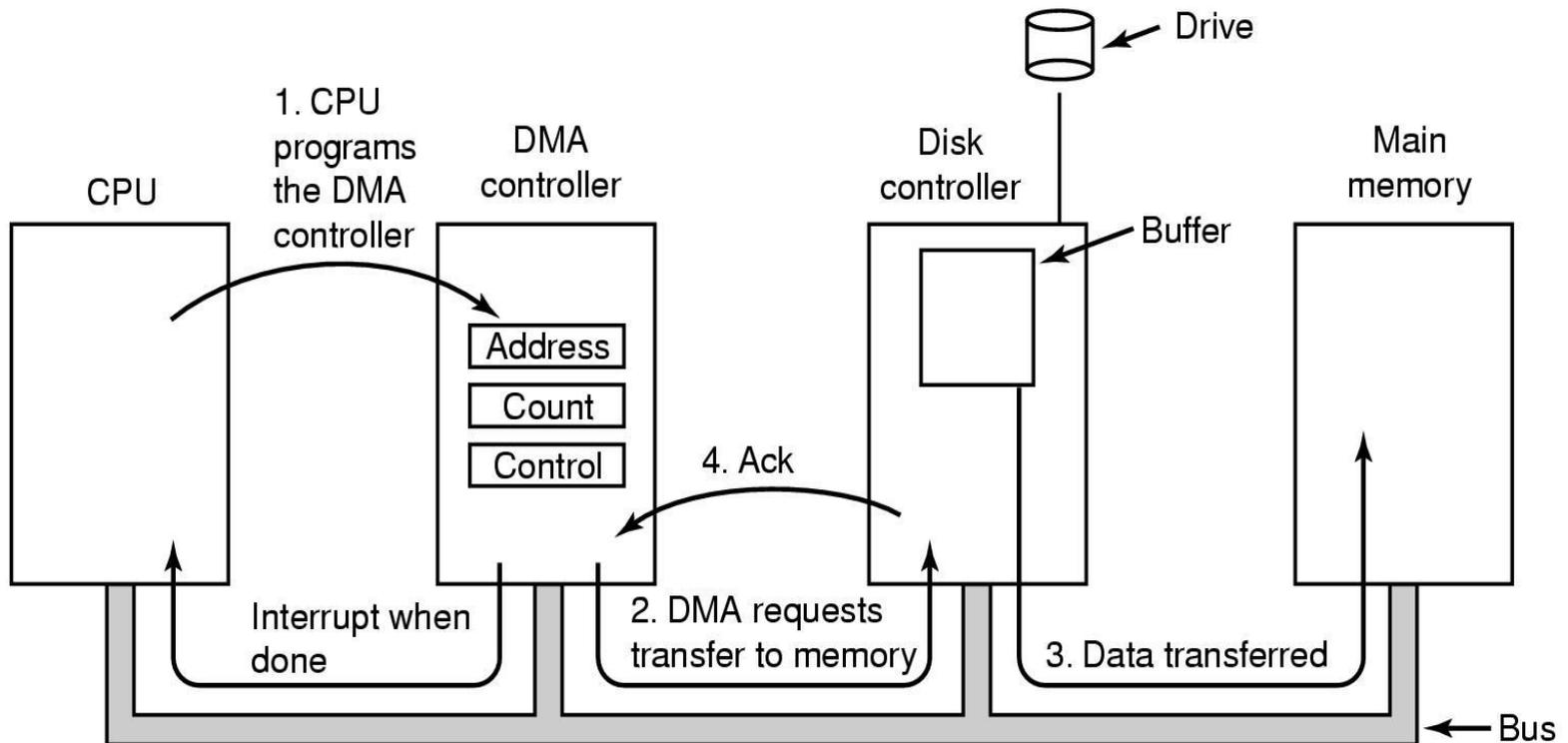
Dispositivos de E/S

Princípios de Hardware

- 4. Após confirmação de leitura, a controladora de DMA **incrementa o endereço de memória** na DMA e **decrementa o contador** da DMA com o número de bytes transferidos;
- Repete os passos de 2 a 4 até o contador da DMA chegar em 0. Assim que o contador chegar em zero (0), a controladora de DMA gera uma interrupção avisando a UCP;
- Quando o SO inicia o atendimento à interrupção, o bloco de dados já está na RAM;

Dispositivos de E/S

Princípios de Hardware



Dispositivos de E/S

Princípios de Hardware

- A DMA pode tratar múltiplas transferências simultaneamente:
 - Possuir vários conjuntos de registradores;
 - Decidir quais requisições devem ser atendidas escalonamento (*Round-Robin* ou prioridades, por exemplo);

Dispositivos de E/S

Princípios de Hardware

- Por que a DMA não escreve diretamente na RAM?
 - Permite realizar consistência dos dados antes de iniciar alguma transferência;
 - Dados (bits) são transferidos do disco a uma taxa constante, independentemente da controladora estar pronta ou não;
 - Acesso à memória depende de acesso ao barramento, que pode estar ocupado com outra tarefa;
 - Com o *buffer*, o barramento é usado apenas quando a DMA opera;

Dispositivos de E/S

Princípios de Hardware

- Interrupções de E/S (*interrupt-driven I/O*):
 - Sinais de interrupção são enviados (através dos barramentos) pelos dispositivos ao processador;
 - Após uma interrupção, o **controlador de interrupções** decide o que fazer;
 - Envia para UCP;
 - Ignora no momento dispositivos geram sinais de interrupção até serem atendidos;

Dispositivos de E/S

Princípios de Hardware

- Controlador de Interrupções:
 - Está presente na placa-mãe;
 - Possui vários manipuladores de interrupção;
 - Diferentes tipos de interrupções IRQs (*Interrupt ReQuest*);
- Manipuladores de interrupção:
 - Gerenciam interrupções realizadas pelos dispositivos de E/S;
 - Bloqueiam *driver* até dispositivo terminar a tarefa;

Dispositivos de E/S

Tratando Interrupções

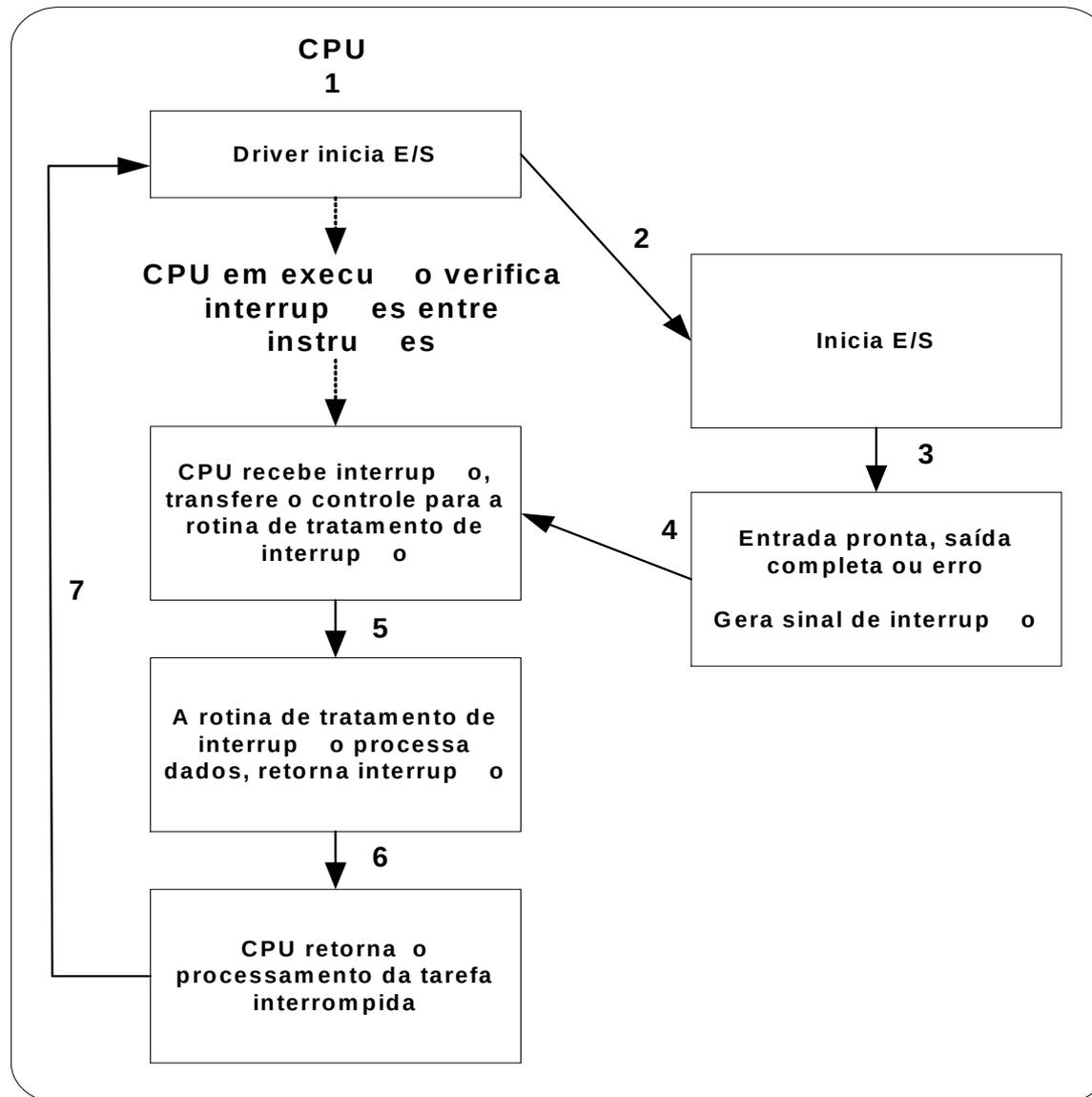
- Sinal (linha) de interrupção é amostrado dentro de cada ciclo de instrução do processador;
- Se sinal ativo salva contexto e atende a interrupção;

Dispositivos de E/S

Tratando Interrupções

- Ciclo de instrução com interrupção: UCP
 - Busca; Decodificação e Execução
 - Verifica se existe interrupção
 - Se não busca próxima instrução,...
 - Se existe interrupção pendente:
 - Suspende a execução do programa;
 - Salva contexto;
 - Atualiza PC (*Program Counter*) apontar para ISR (rotina de atendimento de interrupção);
 - Executa interrupção;
 - Recarrega contexto e continua processo interrompido;

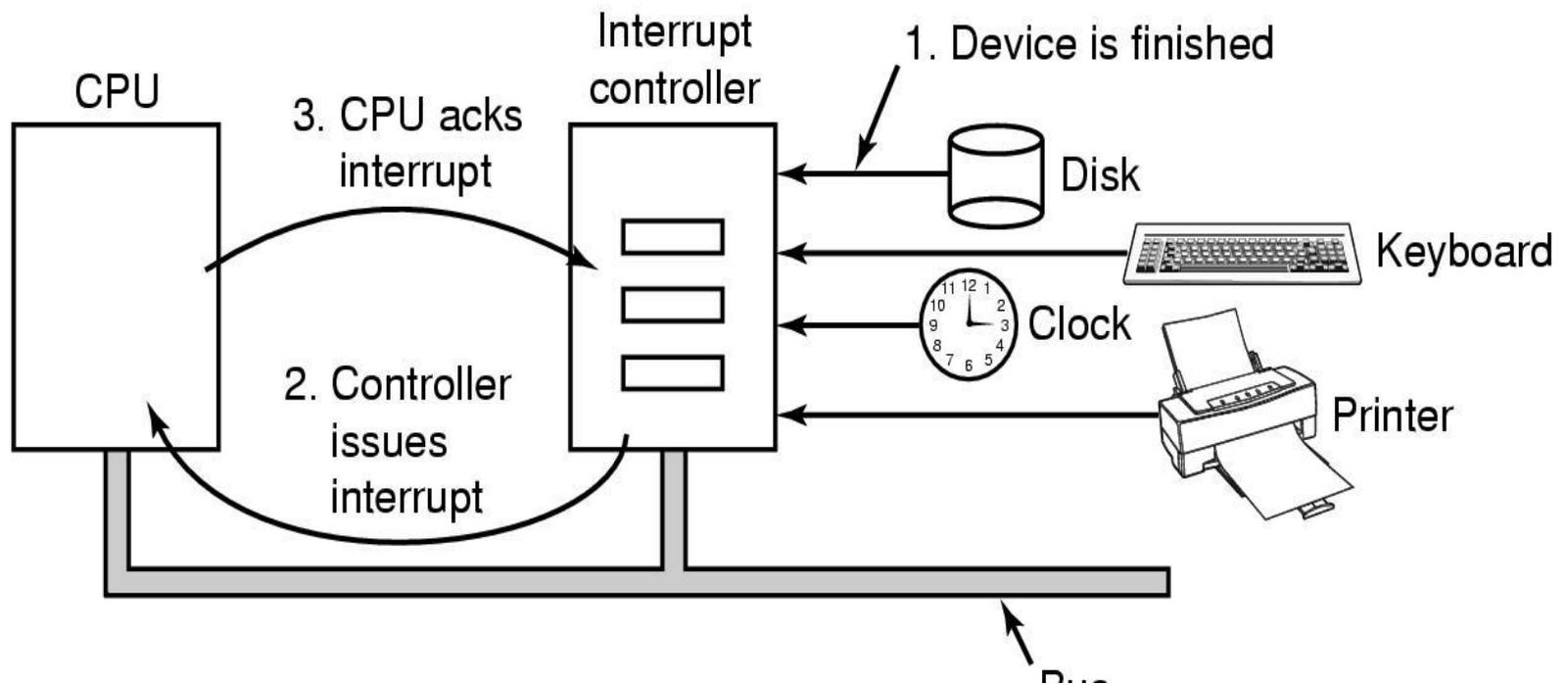
Dispositivos de E/S - Tratando Interrupções



Dispositivos de E/S

Princípios de Hardware

- Como uma interrupção ocorre:



Dispositivos de E/S

Tabela de Interrupções (exemplo)

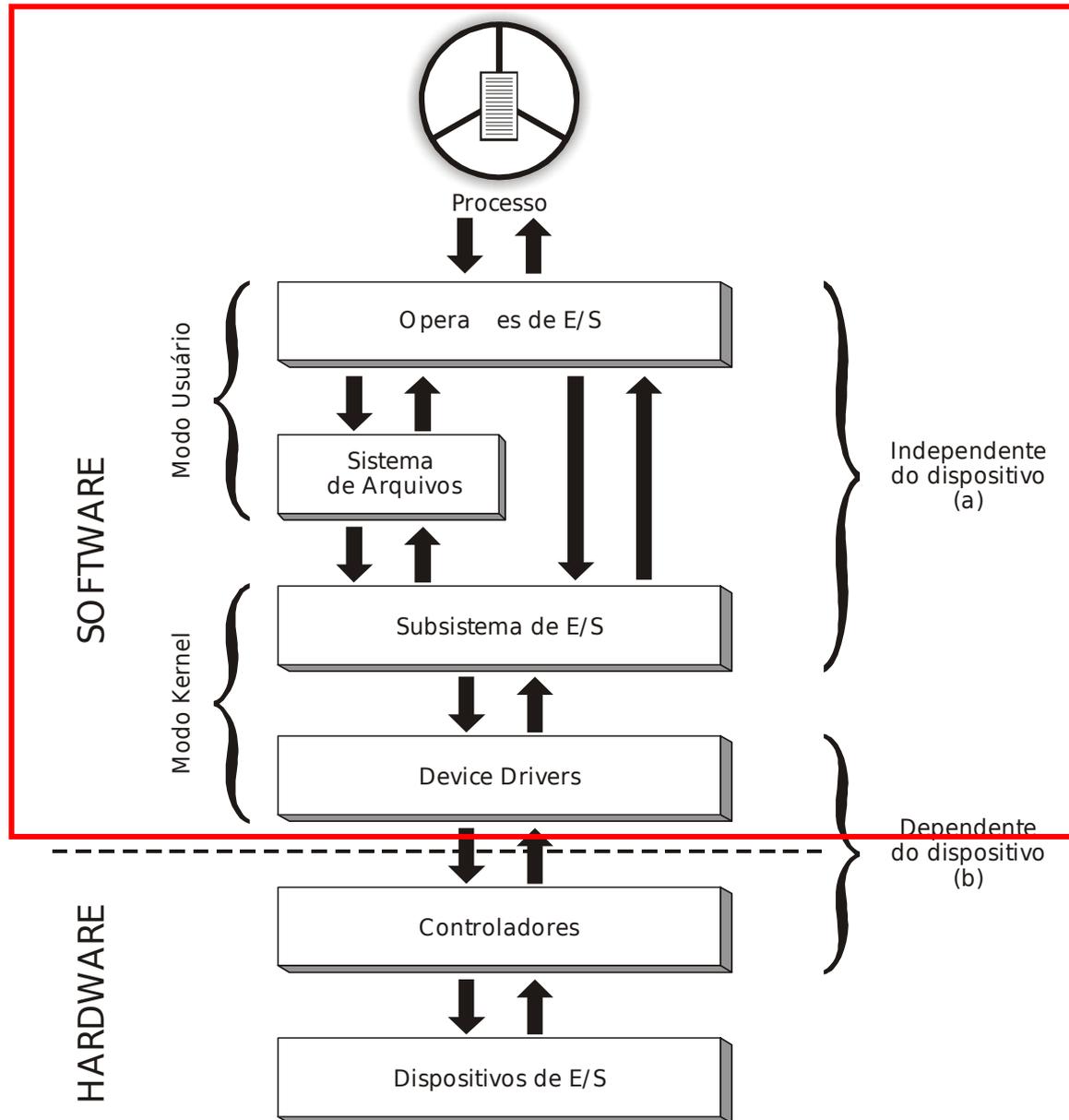
IRQ	Uso padrão	Outras utilizações
00	Timer do sistema	Nenhum
01	Teclado	Nenhum
02	IRQs 8 a 15	Modem, placa de vídeo, porta serial (3, 4), IRQ 9
03	Porta serial 2	Modem, placa de som, placa de rede
04	Porta serial 1	Modem, placa de som, placa de rede
05	Placa de som (codec)	LPT2, COM 3 e 4, Modem, placa de rede, HDC
06	FDC	Placa aceleradora de fita
07	Porta paralela 1	COM 3 e 4, Modem, placa de som, placa de rede
08	Relógio de tempo real	Nenhum
09	Placa de som (midi)	Placa de rede, SCSI, PCI
10	Nenhum	Placa de rede, placa de som, SCSI, PCI, IDE 2
11	Placa de vídeo VGA	Placa de rede, placa de som, SCSI, PCI, IDE 3
12	Mouse P/S2	Placa de rede, placa de som, SCSI, PCI, IDE 3, VGA
13	FPU (<i>Float Point Unit</i>)	Nenhum
14	IDE primária	Adaptador SCSI
15	IDE secundária	Placa de Rede, adaptador SCSI

Dispositivos de E/S

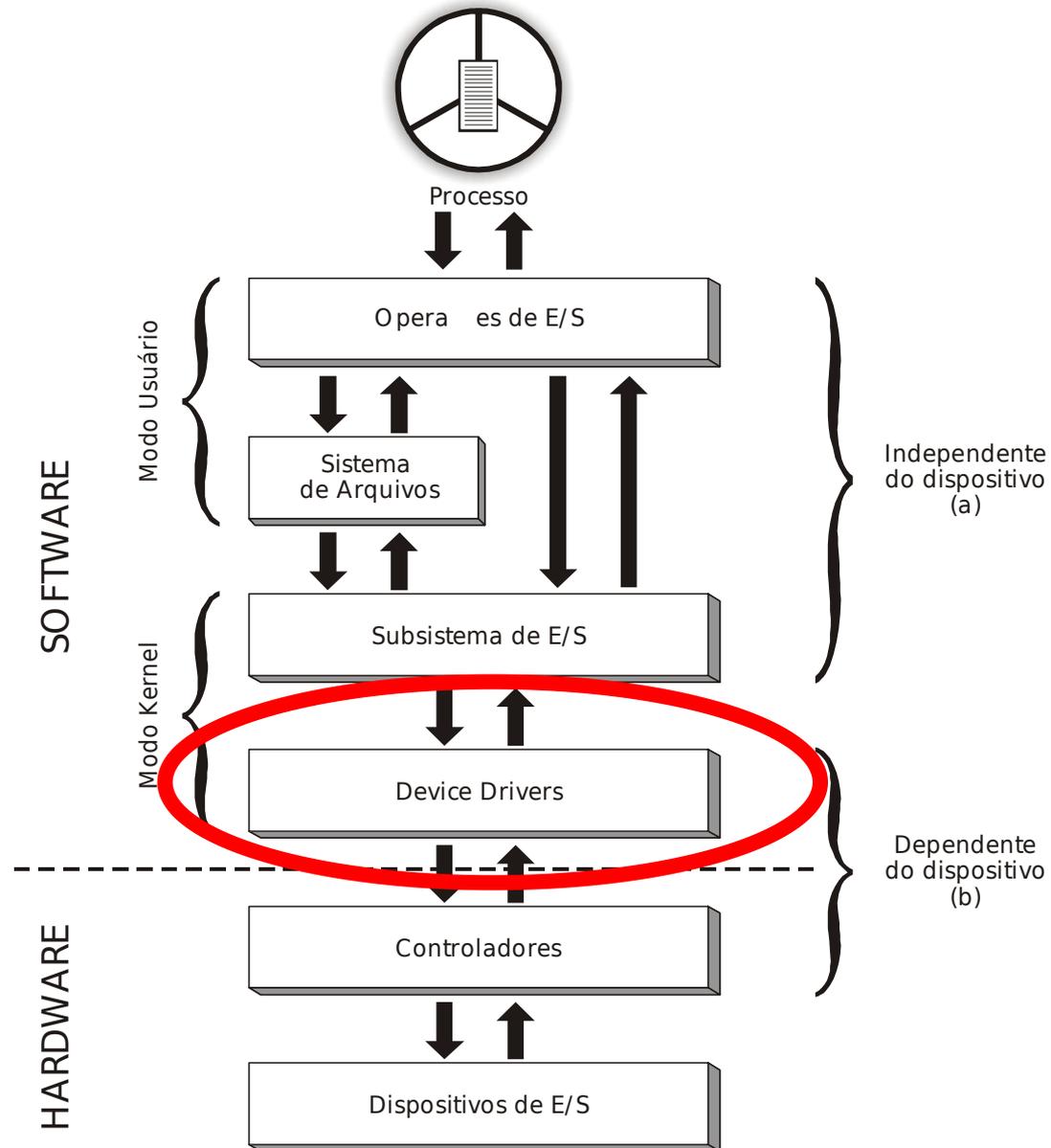
Princípios de Software

- Organizar o software como uma série de camadas facilita a independência dos dispositivos:
 - Camadas mais baixas apresentam detalhes de hardware:
 - Drivers e manipuladores de interrupção;
 - Camadas mais altas apresentam interface para o usuário:
 - Aplicações de Usuário;
 - Chamadas de Sistemas;
 - Software Independente de E/S ou Subsistema de Kernel de E/S;

Dispositivos de Entrada e Saída



Dispositivos de Entrada e Saída



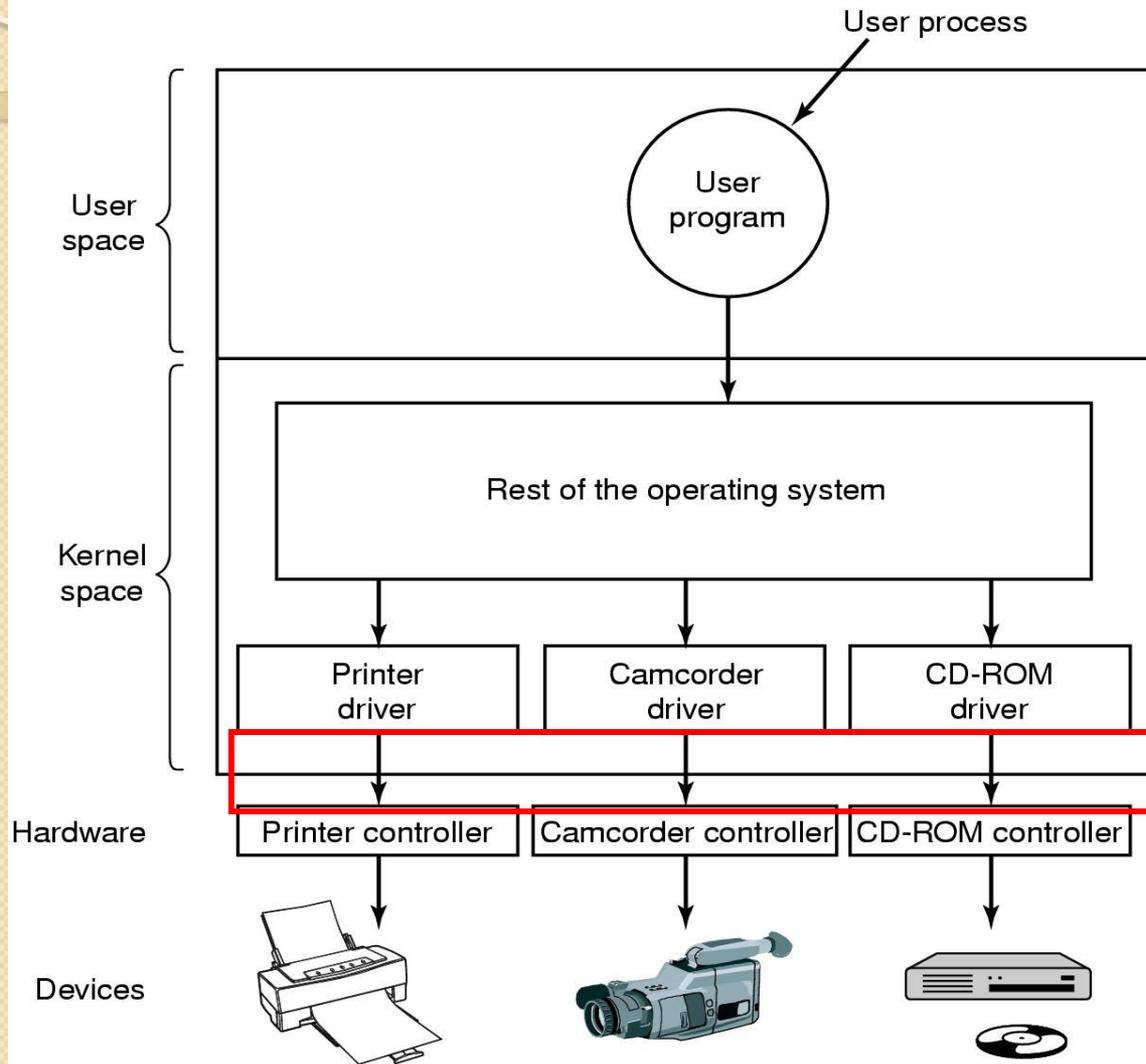
Dispositivos de E/S

Princípios de Software - Camadas

- ***Drivers:***
 - São gerenciados pelo *kernel* do SO;
 - Contêm todo o código dependente do dispositivo;
 - Controlam o funcionamento dos dispositivos por meio de sequência de comandos escritos/lidos nos/dos registradores da controladora;
 - Dispositivos diferentes possuem *drivers* diferentes;
 - Classes de dispositivos podem ter o mesmo *driver*;
 - São dinamicamente carregados;
 - *Drivers* defeituosos podem causar problemas no *kernel* do SO;

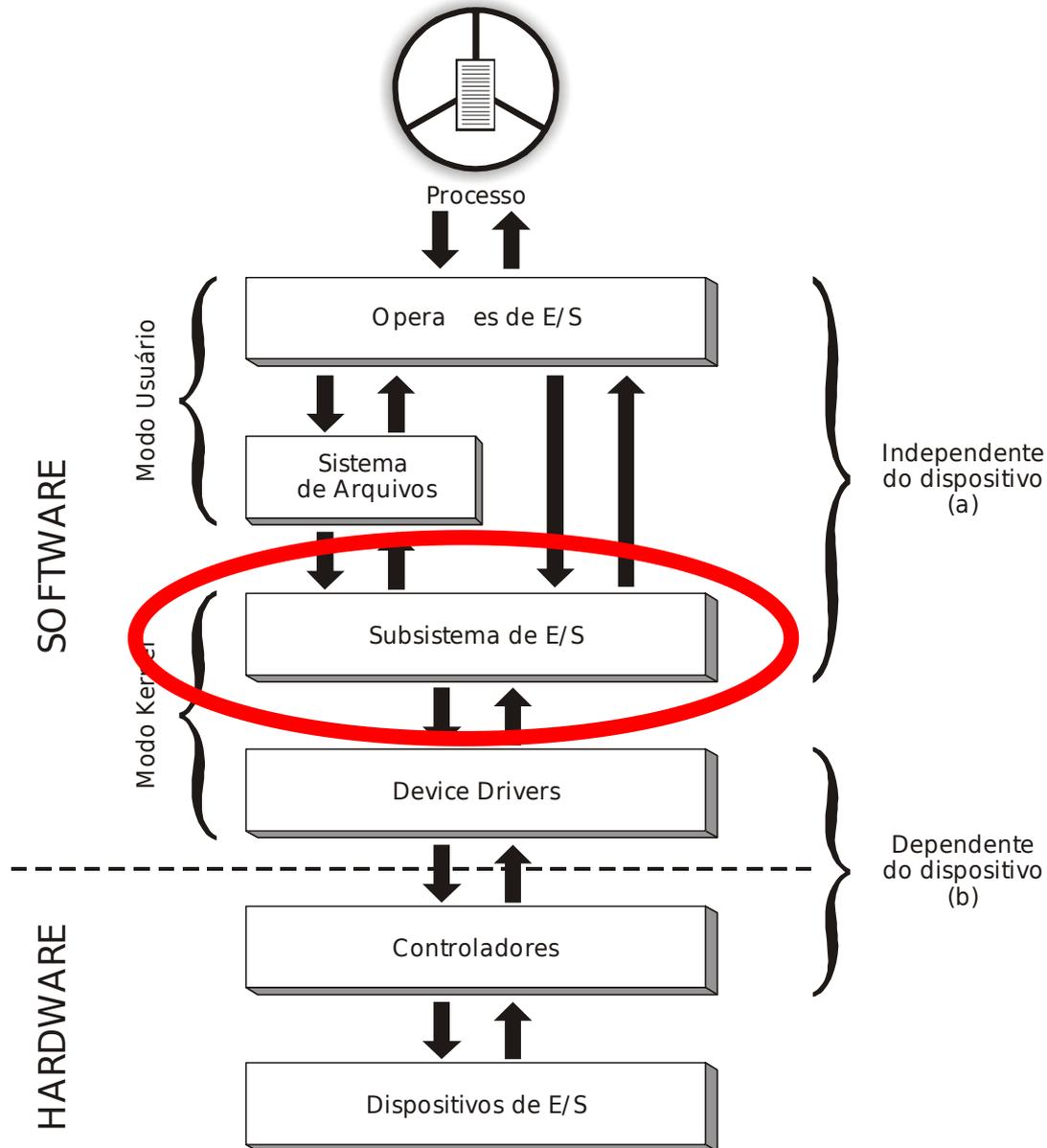
Dispositivos de E/S

Princípios de Software - Camadas



**Comunicação
feita
por barramento;**

Dispositivos de Entrada e Saída



Dispositivos de E/S

Princípios de Software - Camadas

- **Software de E/S no nível Usuário:**
 - Bibliotecas de E/S são utilizadas pelos programas dos usuários
 - Chamadas ao sistema (*system calls*);

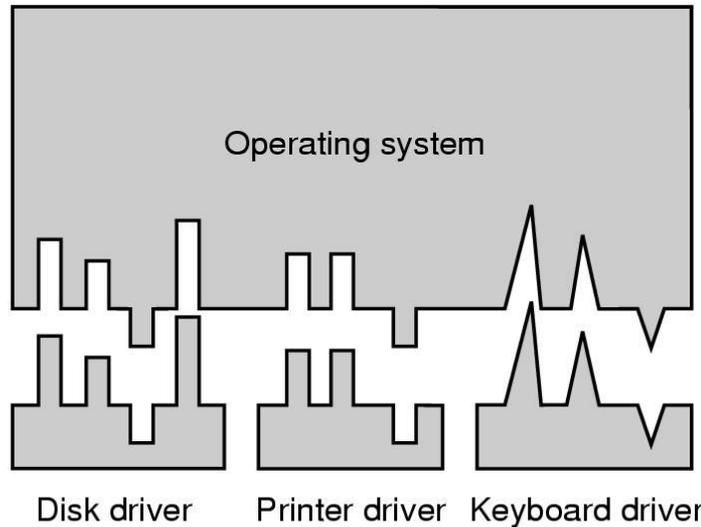
Dispositivos de E/S

Princípios de Software - Camadas

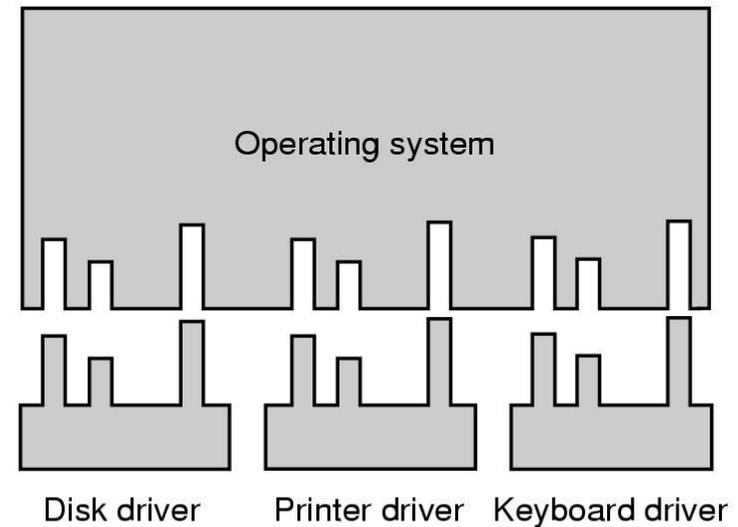
- **Software Independente de E/S:**
 - Realizar as funções comuns a qualquer dispositivo;
 - Prover uma interface uniforme para os drivers dos dispositivos
 - Número de procedimentos que o restante do SO pode utilizar para fazer o driver trabalhar para ele;
 - Fazer o escalonamento de E/S;
 - Atribuir um nome lógico a partir do qual o dispositivo é identificado;
 - Ex.: UNIX (/dev)
 - Prover *buffering*: ajuste entre a velocidade e a quantidade de dados transferidos;
 - *Cache* de dados: armazenar na memória um conjunto de dados frequentemente acessados;

Dispositivos de E/S

Princípios de Software - Camadas



(a)



(b)

(a) Sem padrão de interface

(b) Com padrão de interface (uniforme)

Dispositivos de E/S

Princípios de Software - Camadas

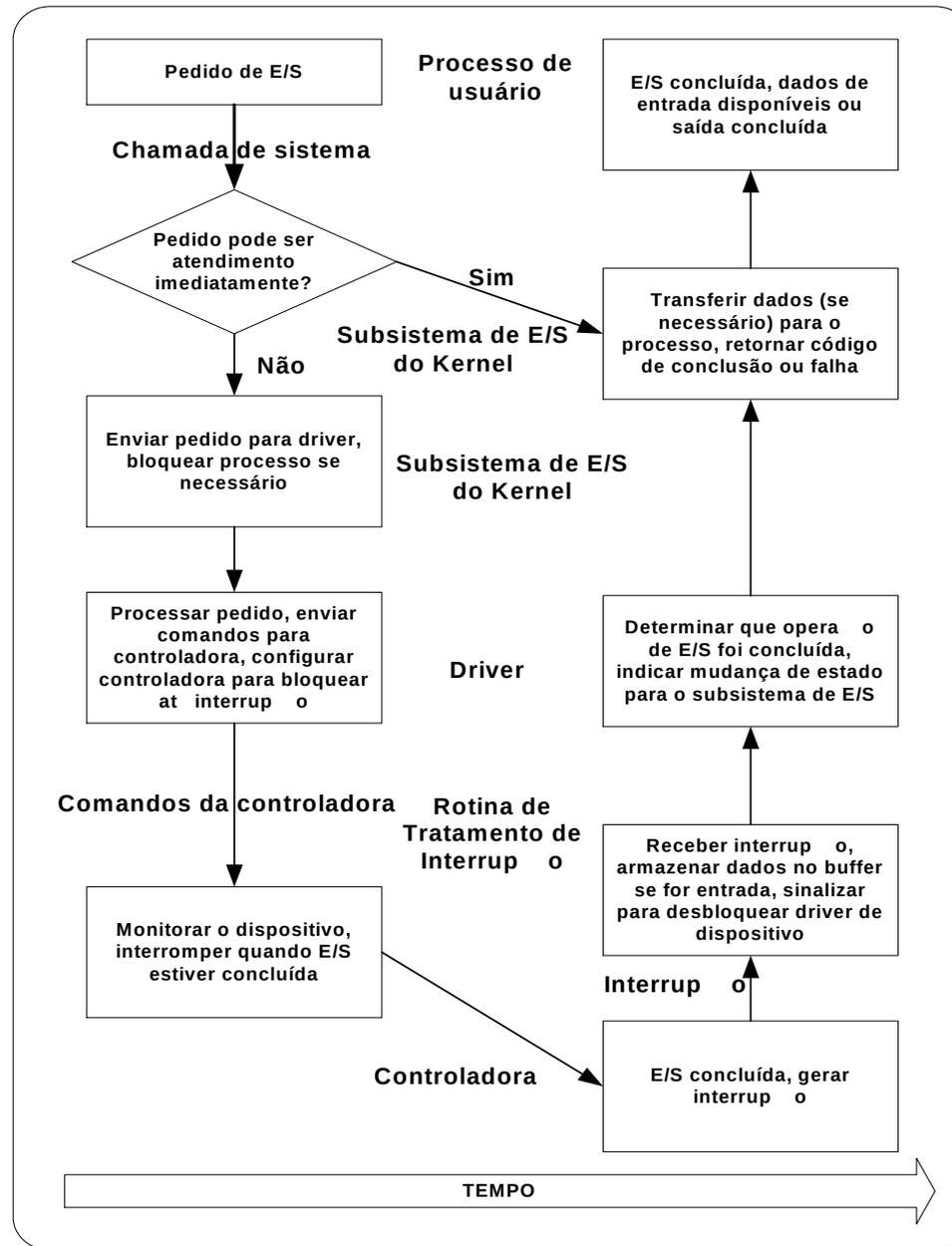
- **Software Independente de E/S:**
 - Reportar erros e proteger os dispositivos contra acessos indevidos :
 - Programação: Ex.: tentar efetuar leitura de um dispositivo de saída (impressora, vídeo);
 - E/S: Ex.: tentar imprimir em uma impressora desligada ou sem papel;
 - Memória: escrita em endereços inválidos;
 - Gerenciar alocação, uso e liberação dos dispositivos acessos concorrentes;

Dispositivos de E/S

Princípios de Software

- **Software Independente de E/S:**
 - Transferência de dados:
 - Síncrona (bloqueante): requer bloqueio até que os dados estejam prontos para transferência;
 - Assíncrona (não-bloqueante): transferências acionadas por interrupções; mais comuns;
 - Tipos de dispositivos:
 - Compartilháveis: podem ser utilizados por vários usuários ao mesmo tempo; Exemplo: disco rígido;
 - Dedicados: podem ser utilizados por apenas um usuário de cada vez; Exemplo: impressora, unidade de fita;

Dispositivos de E/S - Ciclo de E/S



Dispositivos de E/S - Ciclo de E/S

Sequência da Figura anterior

- Um processo emite uma chamada de sistema bloqueante (por exemplo: `read`) para um arquivo que já esteve aberto (`open`);
- O código da chamada de sistema verifica os parâmetros. Se os parâmetros estiverem corretos e o arquivo já estiver no *buffer* (*cache*), os dados retornam ao processo e a E/S é concluída;
- Se os parâmetros estiverem corretos, mas o arquivo não estiver no *buffer*, a E/S precisa ser realizada;
 - E/S é escalonada;
 - Subsistema envia pedido para o *driver*;

Dispositivos de E/S - Ciclo de E/S

Sequência da Figura anterior

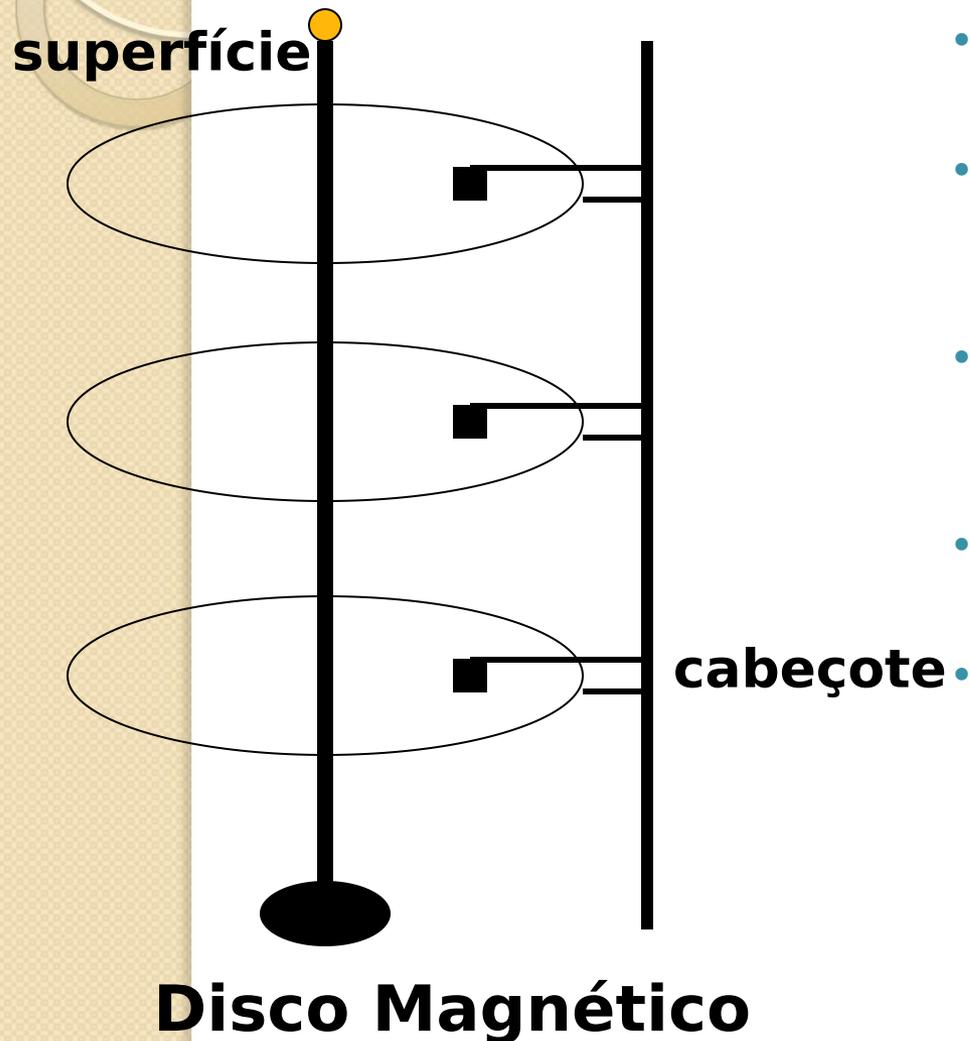
- *Driver* aloca espaço de *buffer*, escalona E/S e envia comando para a controladora do dispositivo escrevendo nos seus registradores de controle;
- *Driver* pode usar a DMA;
- A controladora do dispositivo opera o hardware, ou seja, o dispositivo propriamente dito;
- Após a conclusão da E/S, uma interrupção é gerada;
- A rotina de tratamento de interrupções apropriada recebe a interrupção via vetor de interrupção, armazena os dados, sinaliza o *driver* e retorna da interrupção;

Dispositivos de E/S - Ciclo de E/S

Sequência da Figura anterior

- *Driver* recebe o sinal, determina qual pedido de E/S foi concluído, determina o status e sinaliza que o pedido está concluído;
- Kernel transfere dados ou códigos de retorno para o espaço de endereçamento do processo que requisitou a E/S e move o processo da fila de bloqueados para a fila de prontos;
- Quando o escalonador escalona o processo para a UCP, ele retoma a execução na conclusão da chamada ao sistema.

Dispositivos de E/S - Discos



- Cada superfície é dividida em **trilhas**;
- Cada trilha é dividida em **setores** ou **blocos** (512 bytes a 32K);
- Um conjunto de trilhas (com a mesma distância do eixo central) formam um **cilindro**;
- Cabeças de leitura e gravação;
- Tamanho do disco:
nº cabeças (faces) x
nº cilindros (trilhas) x
nº setores x
tamanho_setor;

Dispositivos de E/S - Discos

- Discos Magnéticos:

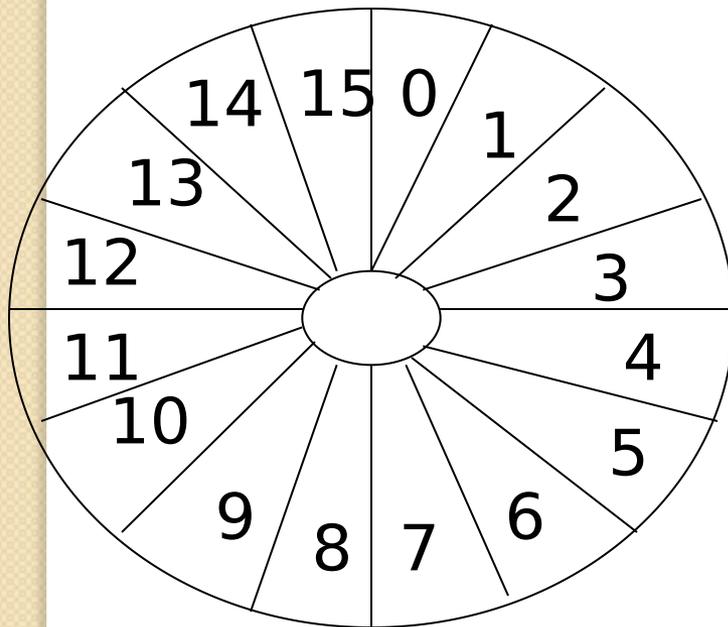
- Grande evolução em relação a:
 - Velocidade de acesso (*seek*): tempo de deslocamento do cabeçote até o cilindro correspondente à trilha a ser acessada;
 - Transferências: tempo para transferência (leitura/escrita) dos dados;
 - Capacidade;
 - Preço;

Dispositivos de E/S - Discos

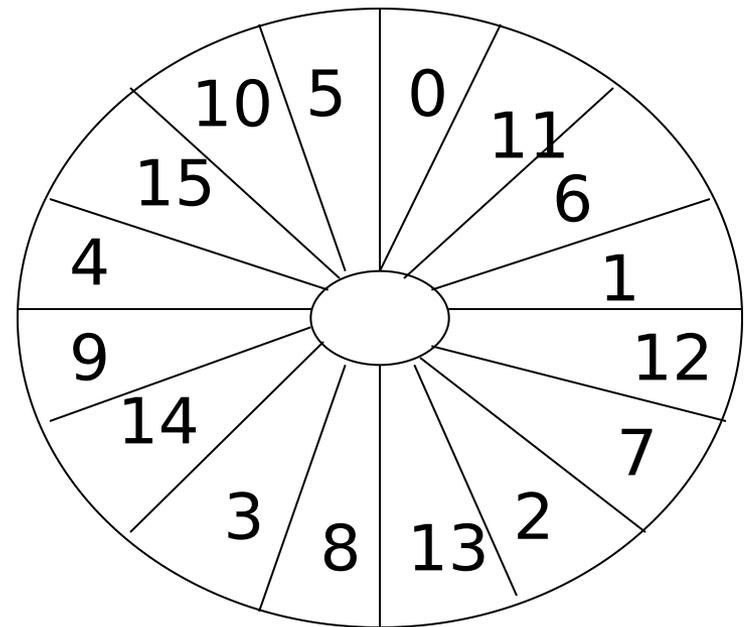
- Técnica para reduzir o tempo de acesso: entrelaçamento (*interleaving*):
 - Setores são numerados com um espaço entre eles;
 - Entre o **setor K** e o **setor K+1** existem **n** (fator de entrelaçamento) setores;
 - Número **n** depende da velocidade do processador, do barramento, da controladora e da velocidade de rotação do disco;

Dispositivos de E/S - Discos

Trilhas com 16 setores



Disco
A
 $N = 0$



Disco B
 $N = 2$

Dispositivos de E/S - Discos

- *Drivers* de Disco:
 - Fatores que influenciam tempo para leitura/escrita no disco:
 - Velocidade de acesso (*seek*) tempo para o movimento do braço até o cilindro;
 - *Delay* de rotação (latência) tempo para posicionar o setor na cabeça do disco;
 - Tempo da transferência dos dados;
 - Tempo de acesso:
 - $T_{seek} + T_{latência*} + T_{transferência}$;

* Tempo necessário para o cabeçote se posicionar no setor de escrita/leitura;

Dispositivos de E/S – Discos

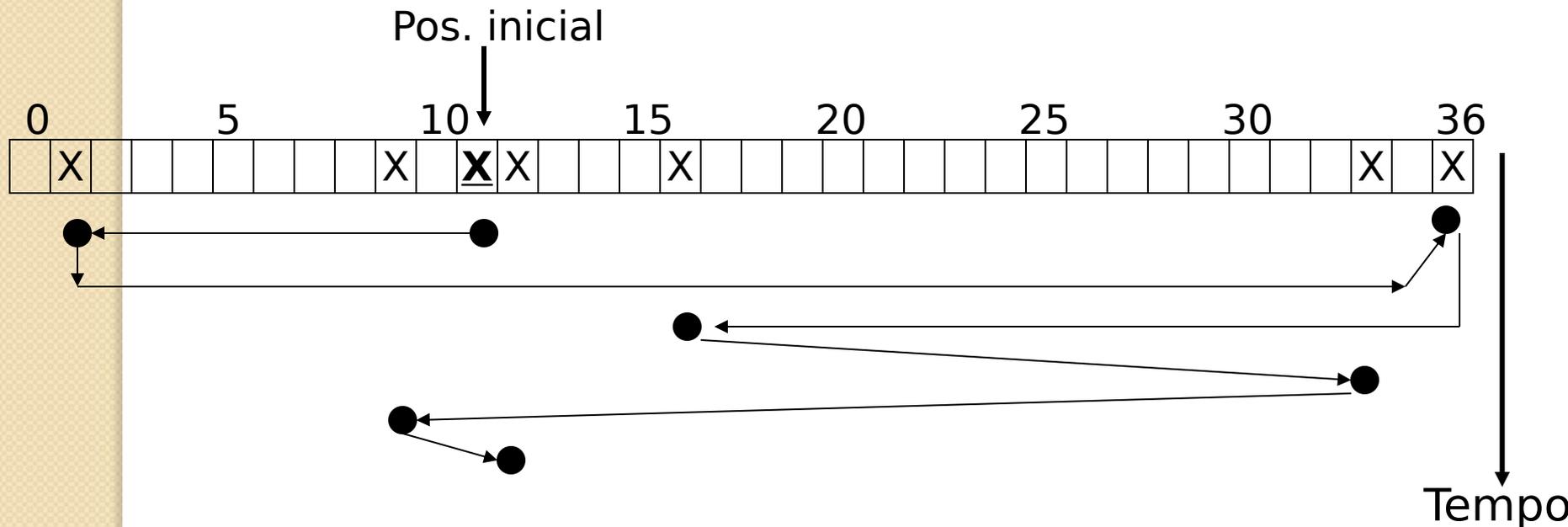
- Algoritmos de escalonamento no disco:
 - FCFS (FIFO): *First-Come First-Served*;
 - SSF: *Shortest Seek First*;
 - *Elevator* (também conhecido como SCAN);
- Escolha do algoritmo depende do número e do tipo de pedidos;
- *Driver* mantém uma lista encadeada com as requisições para cada cilindro;

Dispositivos de E/S - Discos

Disco com 37 cilindros;

Lendo bloco no cilindro 11;

Requisições: 1,36,16,34,9,12, nesta ordem

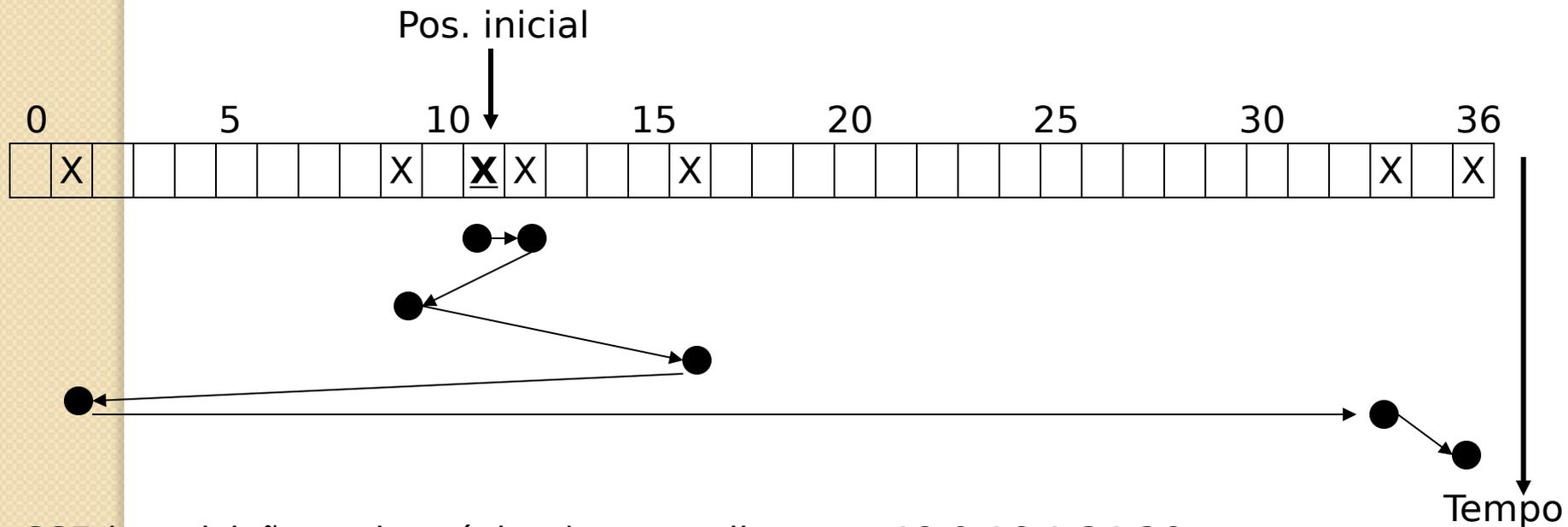


FCFS: atendimento: 1,36,16,34,9,12;

movimentos do braço (número de cilindros): $10,35,20,18,25,3 = 111$;

Dispositivos de E/S - Discos

Disco com 37 cilindros;
Lendo bloco no cilindro 11;
Requisições: 1,36,16,34,9,12, nesta ordem

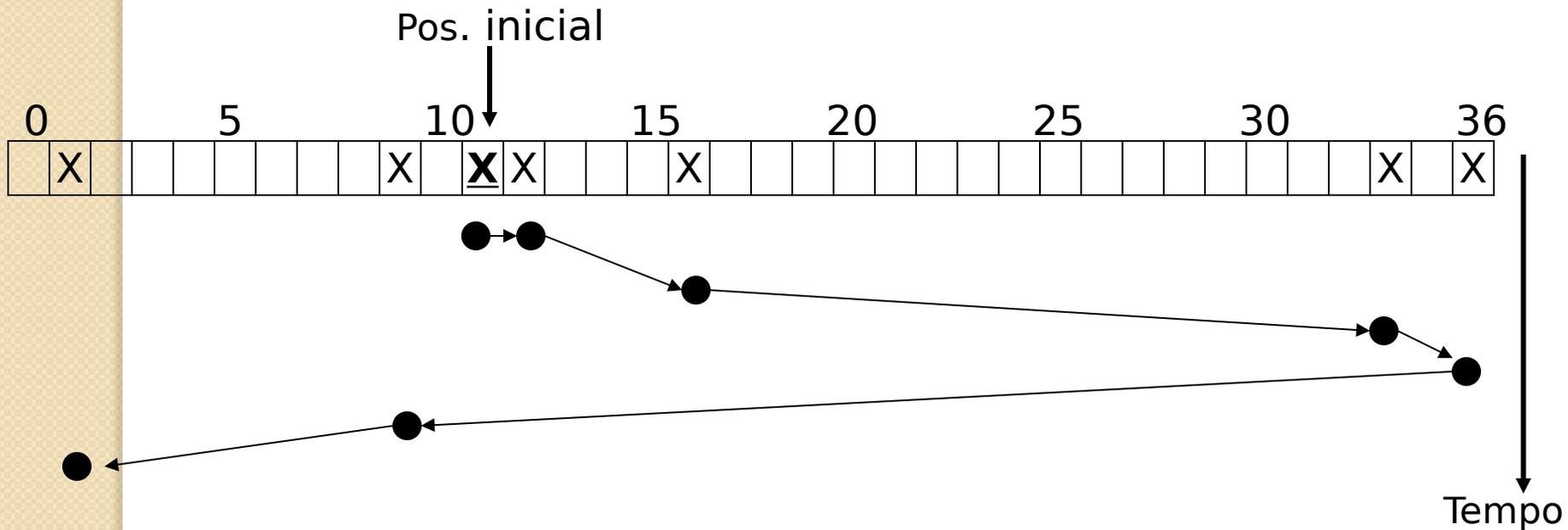


SSF (requisição mais próxima) → atendimento: 12,9,16,1,34,36;
movimentos do braço (número de cilindros): 1,3,7,15,33,2 = 61;

Dispositivos de E/S - Discos

Disco com 37 cilindros;
Lendo bloco no cilindro 11;
Requisições: 1,36,16,34,9,12, nesta ordem

Bit de direção corrente (*driver*):
Se *Up*, atende próxima requisição;
senão *Bit = Down*;
muda direção e atende requisição;



Elevator (requisições na mesma direção)

atendimento: 12,16,34,36,9,1

movimentos do braço (número de cilindros): 1,4,18,2,27,8 = 60;

Dispositivos de E/S – Discos RAID

- RAID (*Redundant Array of Independent Disks*) → armazena grandes quantidades de dados;
- RAID combina diversos discos rígidos em uma estrutura lógica:
 - Aumentar a confiabilidade, capacidade e o desempenho dos discos;
 - Recuperação de dados → redundância dos dados;
 - Armazenamento simultâneo em vários discos permite que os dados fiquem protegidos contra falha (não simultânea) dos discos;
 - Performance de acesso, já que a leitura da informação é simultânea nos vários dispositivos;

Dispositivos de E/S – Discos RAID

- Pode ser implementado por:
 - Hardware (controladora):
 - Instalação de uma placa RAID no servidor, o subsistema RAID é implementado totalmente em hardware;
 - Libera o processador para se dedicar exclusivamente a outras tarefas;
 - A segurança dos dados aumenta no caso de problemas devido à checagem da informação na placa RAID antes da gravação;

Dispositivos de E/S – Discos RAID

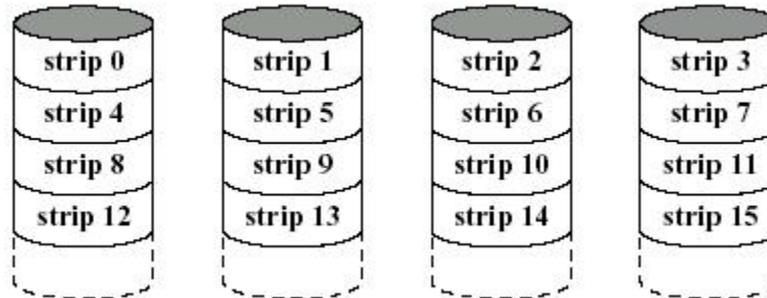
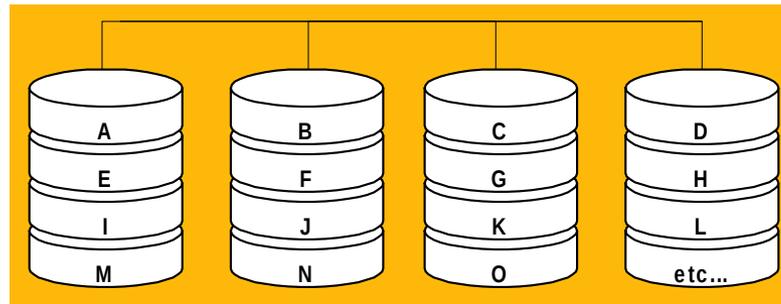
- Pode ser implementado por:
 - Software (sistema operacional)
 - Menor desempenho no acesso ao disco;
 - Oferece um menor custo e flexibilidade;
 - Sobrecarrega o processador com leitura/escrita nos discos;
- Para o SO existe um único disco;

Dispositivos de E/S – Discos RAID

- A forma pela qual os dados são escritos e acessados define os níveis de RAID (até 9 níveis):
 - RAID 0:
 - Também conhecido como *Striping*;
 - Arquivos são espalhados entre os discos em *stripes*;
 - Melhora desempenho das operações de E/S;
 - Sem controle ou correção de erros;
 - Todo o espaço do disco é utilizado para armazenamento;
 - Utilizam mesma controladora (controladora RAID);
 - Aplicações multimídia (alta taxa de transferência);

Dispositivos de E/S – Discos RAID

- Nivel 0

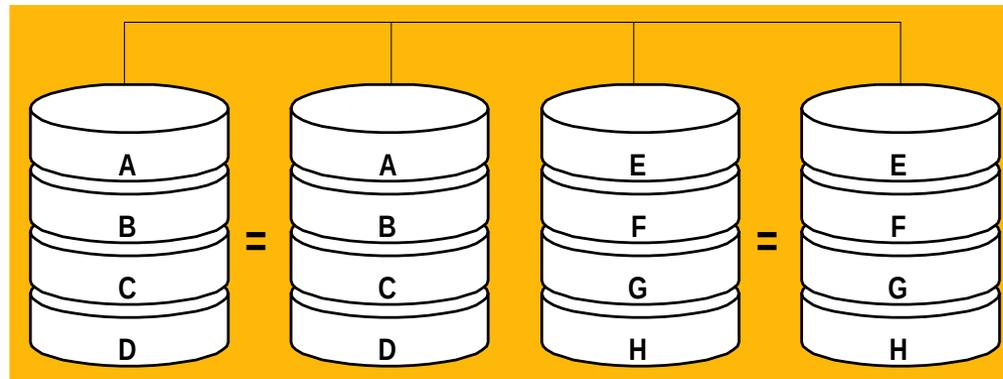
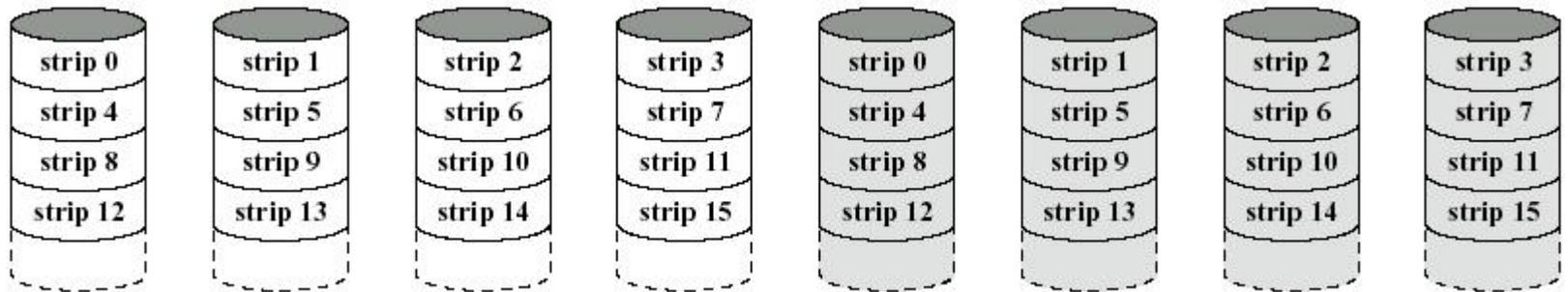


Dispositivos de E/S – Discos RAID

- RAID 1:
 - Conhecido como espelhamento (*mirroring*);
 - Operações de escrita no disco primário são replicadas em um disco secundário;
 - Pode ter controladoras diferentes;
 - Desvantagem: espaço físico em dobro (alto custo);
 - Transações *on-line* (tolerância a falhas);
- RAID 10:
 - Combinação dos RAID 1 e RAID 0;

Dispositivos de E/S – Discos RAID

- Nível 10

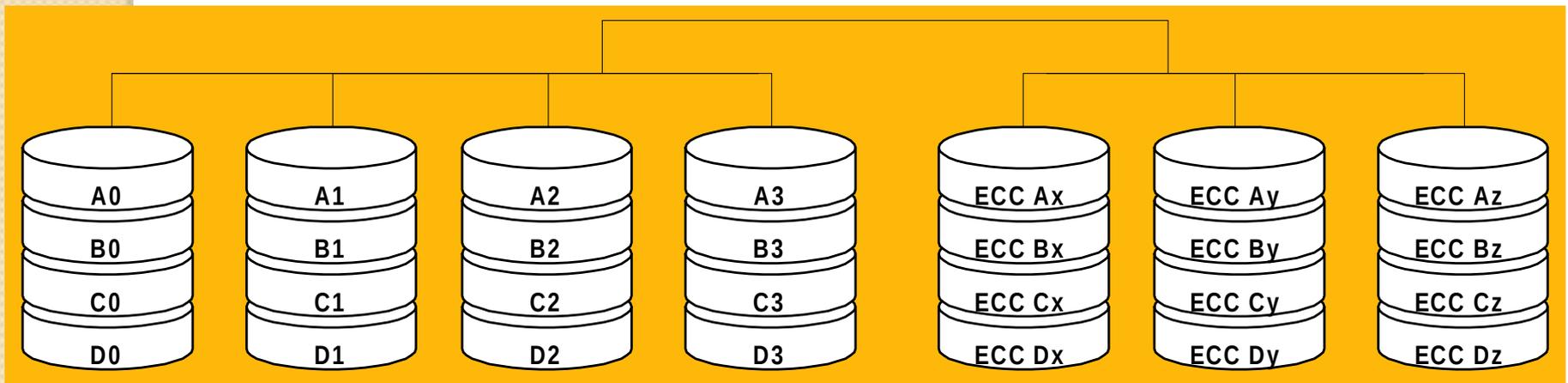
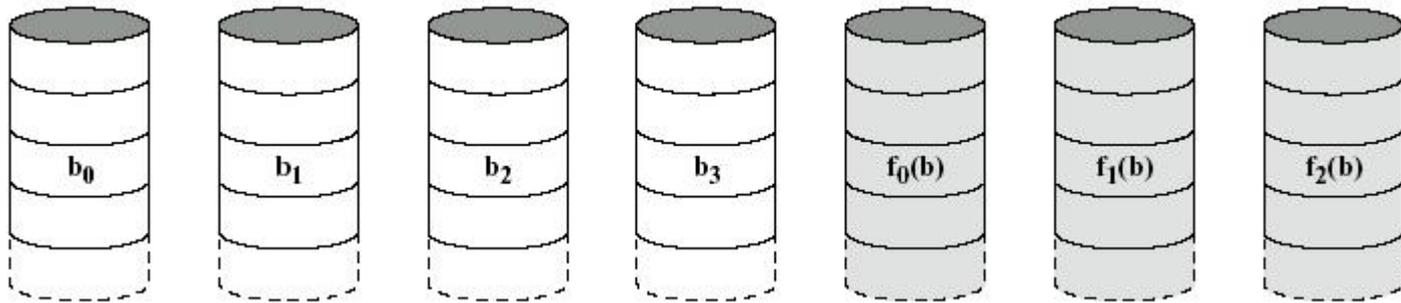


Dispositivos de E/S – Discos RAID

- RAID 2/3/4:
 - Dados são armazenados em discos diferentes com paridade (permite reconstruir dados perdidos); *Stripes*;
 - Paridade é mantida em um disco apenas;
 - Diferença básica: como a paridade é calculada (na transferência):
 - *Raid 2 - Hamming ECC (error-correcting codes)- nível de bit*;
 - *Raid 3 - XOR ECC - nível de byte ou bit*;
 - *Raid 4 - XOR ECC - nível de bloco*;
- RAID 5:
 - *Stripes*;
 - Paridade *XOR ECC* distribuída - nível de bloco;
 - Paridade está distribuída nos discos;
- RAID 6:
 - *Stripes*;
 - Raid 5 com dois discos de paridade;

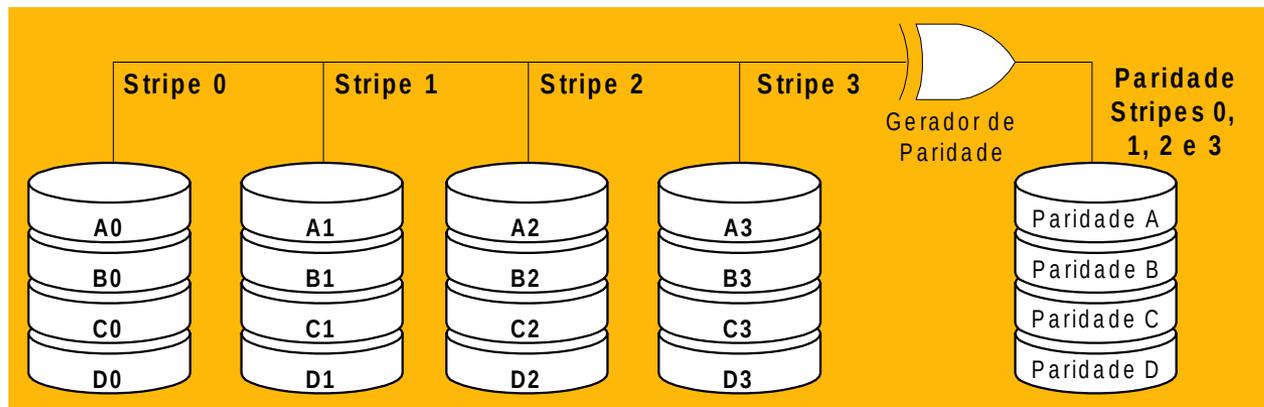
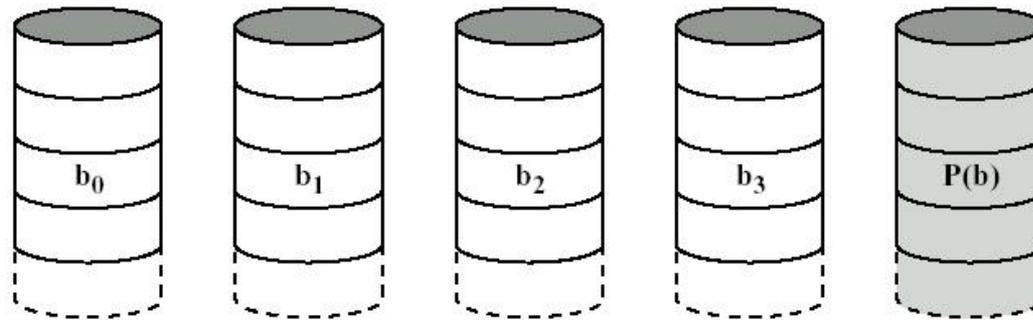
Dispositivos de E/S – Discos RAID

- Nivel 2



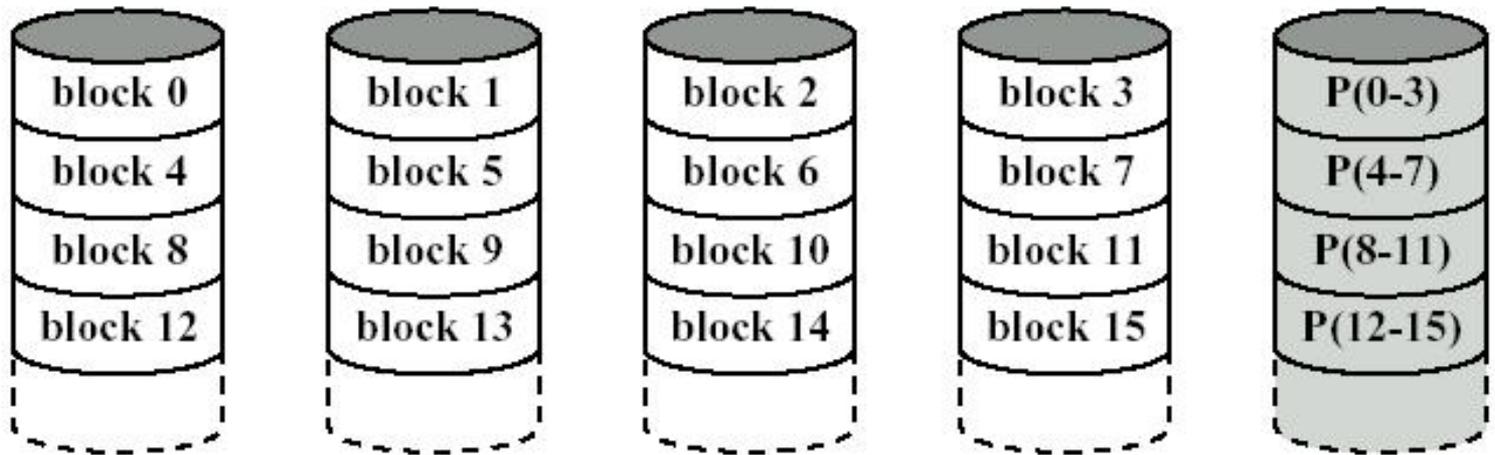
Dispositivos de E/S – Discos RAID

- Nível 3



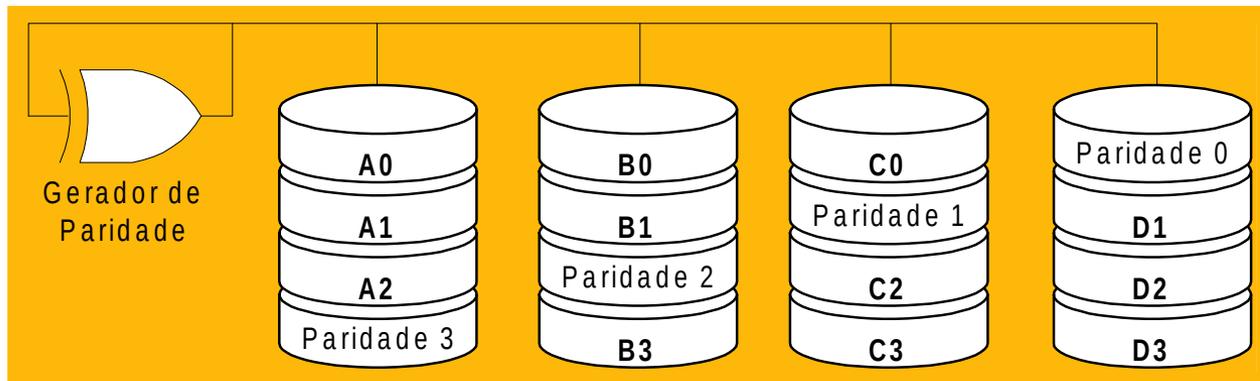
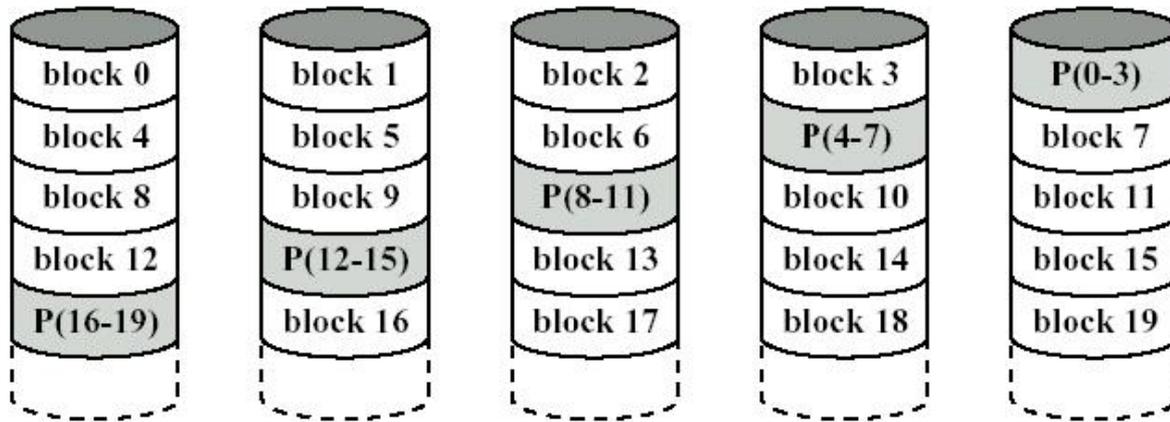
Dispositivos de E/S – Discos RAID

- Nivel 4



Dispositivos de E/S – Discos RAID

- Nível 5



Dispositivos de E/S – Discos RAID

- Nível 6
 - 2 cálculos diferentes de paridade

