

Sistemas Operacionais

Prof. Dra. Kalinka Regina Lucas Jaquie Castelo Branco
kalinka@icmc.usp.br

Apresentação baseada nos slides do Prof. Dr. Antônio Carlos Sementille e da Profa. Dra. Luciana A. F. Martimiano e nas transparências fornecidas no site de compra do livro "Sistemas Operacionais Modernos"

[2]

Dispositivos de Entrada e Saída

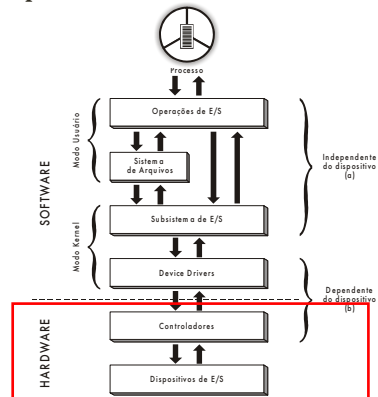
- SO pode atuar de duas maneiras diferentes:
 - Como máquina estendida (*top-down*) – tornar uma tarefa de baixo nível mais fácil de ser realizada pelo usuário;
 - Como gerenciador de recursos (*bottom-up*) – gerenciar os dispositivos que compõem o computador.

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:
 - Hardware;
 - Software.

[3]

Dispositivos de Entrada e Saída



[4]

Dispositivos de E/S Princípios de Hardware

Uma das funções principais de um Sistema Operacional é controlar todos os dispositivos de entrada/saída do computador. Ele deve:

- enviar comandos aos dispositivos;
- atender interrupções;
- fornecer uma interface entre os dispositivos e o resto do sistema que seja simples e fácil de usar.

Geralmente, o código para tratamento da entrada e saída representa uma fração significativa do sistema operacional total

[5]

Dispositivos de E/S Princípios de Hardware

Módulos de E/S: Controladores de Dispositivos

As Unidades de E/S são geralmente compostas de dois componentes principais:

- Controlador de dispositivo: parte programável (Nos PCs é normalmente uma placa de circuito impresso);
- Componente Mecânico

- Muitos controladores podem controlar vários dispositivos idênticos
- Órgãos de padronização: IEEE, ISO, ANSI, etc.

[6]

Dispositivos de E/S Princípios de Hardware

- O S.O. sempre trata com o controlador, não com os dispositivos.
- A Comunicação entre CPU e controladores é feita através de barramentos comuns (interface de alto nível)
- Interface entre controlador e dispositivo: baixo nível
- *Mainframes*: múltiplos barramentos e processadores especializados em E/S (canais de E/S).

{ 7 }

Dispositivos de E/S Princípios de Hardware

Classificação quanto ao tipo de transferência de E/S

• Podem ser divididos em 2 categorias:

- Dispositivos de Bloco – armazenam informações em blocos de tamanhos fixos, cada um com seus próprios endereços. Os tamanhos dos blocos geralmente variam de 512 à 32.768 bytes.
- A principal característica dos dispositivos desta categoria, é a possibilidade de ler e escrever cada bloco de maneira independente e permitir operações de busca.
- Exemplos: Discos rígidos.

{ 8 }

Dispositivos de E/S Princípios de Hardware

- Dispositivos de Caracter:
 - aceitam uma sequência de caracteres sem se importar com a estrutura do bloco;
 - Informação não é endereçável e não possuem qualquer operação de busca ("seek operation").
 - Exemplos: impressoras, interfaces de redes, placas de som e etc., fazem parte desta categoria.

{ 9 }

Dispositivos de E/S Princípios de Hardware

- Este esquema de classificação não é perfeito, porém é genérico o suficiente (por ex., o *timer* não se encaixa). *Clocks*: provocam interrupções em intervalos definidos.
- O sistema de arquivos, por exemplo, trata com dispositivos de bloco abstratos.
- Entretanto a classificação auxilia na obtenção de *independência de dispositivo*
 - Parte dependente está a cargo dos *drivers* – software que controla o acionamento dos dispositivos.

{ 10 }

Dispositivos de E/S Princípios de Hardware

- Os dispositivos de E/S podem apresentar uma grande variedade de velocidade

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

{ 11 }

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.

{ 12 }

Dispositivos de E/S Princípios de Hardware

- Cada controladora possui um conjunto de registradores de controle, que são utilizados na comunicação com a CPU;
- 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.

{ 13 }

Dispositivos de E/S Princípios de Hardware

- Como a CPU se comunica com esses registradores de controle?
 - **Porta**: cada registrador de controle possui um número de porta (ou porto) de E/S de 8 ou 16 bits;
 - Instrução em *Assembler*;
 - *Mainframes* IBM;
 - SOs atuais fazem uso dessa estratégia para a maioria dos dispositivos.

{ 14 }

Dispositivos de E/S Princípios de Hardware

- **Memory-mapped**: mapear os registradores de controle em espaços de memória;
 - Cada registrador possui um único endereço de 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;
- **Estratégia híbrida**:
 - Registradores → Porta;
 - Buffers → Memória;
 - Exemplo: Pentium - endereços de 640k a 1M para os buffers e as portas de E/S de 0 a 64k para registradores.

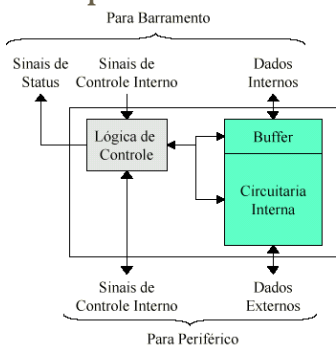
{ 15 }

Dispositivos de E/S Princípios de Hardware

- Como funciona a comunicação da CPU com os dispositivos?
 - Quando a CPU deseja ler uma palavra, ela coloca o endereço que ela está desejando no barramento de endereço e manda um comando **READ** no barramento de controle;
 - Essa comunicação pode ser controlada pela própria CPU ou pela DMA.

{ 16 }

Dispositivos de E/S Princípios de Hardware



Estrutura genérica de um controlador (módulo de E/S)

{ 17 }

Dispositivos de E/S Princípios de Hardware

- Controlador de disco: converte o fluxo serial de bits em um bloco de bytes, executando qualquer correção necessária.
- Cada controlador possui registradores para a comunicação com a CPU.
- Em alguns computadores: estes registradores podem fazer parte do espaço de endereçamento da memória principal.

{ 18 }

Dispositivos de E/S Princípios de Hardware

- O S.O.: executa E/S escrevendo comandos (e seus parâmetros, se existirem) nos registradores dos controladores.
- Quando um comando é aceito, a CPU pode deixar que o controlador trabalhe sozinho, indo executar outra tarefa.
- Quando o dispositivo termina, avisa a CPU através de uma interrupção.

[19]

Dispositivos de E/S Princípios de Hardware

- Os módulos de E/S podem operar de 3 maneiras básicas:
 - E/S programada
 - Mais usada em sistemas embarcados
 - E/S orientada à Interrupções
 - E/S com uso da DMA (Acesso Direto à Memória)
- O que distingue as três formas: a participação da CPU e a utilização das interrupções

[20]

Dispositivos de E/S Princípios de Hardware

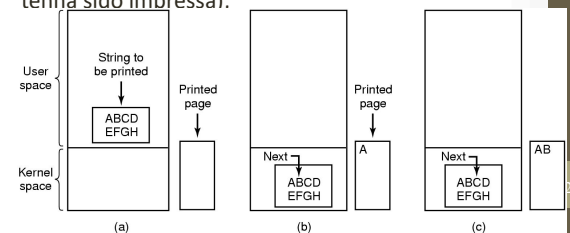
E/S Programada

- Na E/S programada: os dados são trocados entre a CPU e o Módulo de E/S
- A CPU executa um programa que:
 - verifica o estado do módulo de E/S, preparando-o para a operação;
 - se necessário, enviando o comando que deve ser executado; e
 - aguardando o resultado do comando, para então, efetuar a transferência entre o módulo de E/S e algum registrador da CPU.

[21]

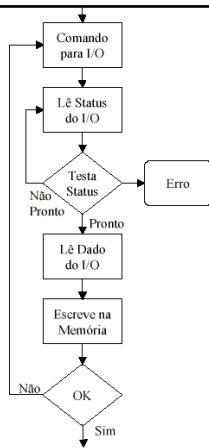
Dispositivos de E/S Princípios de Hardware

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



[22]

E/S Programada



[23]

Dispositivos de E/S Princípios de Hardware

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

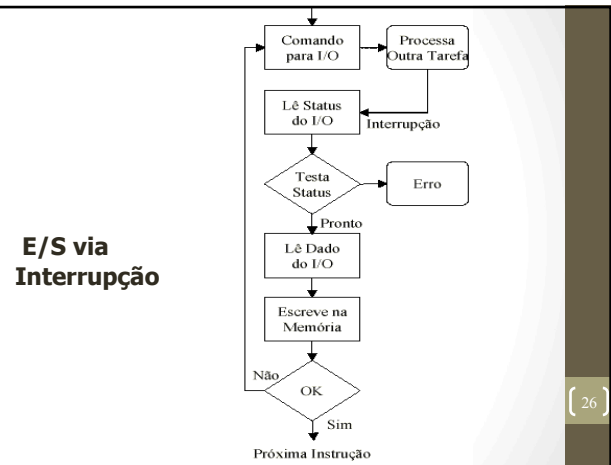
[24]

Dispositivos de E/S Princípios de Hardware

E/S via Interrupção

- Na E/S via interrupção: o mecanismo de interrupções é utilizado para superar o problema da espera da CPU por operações nos periféricos
- A interrupção permite que uma unidade ganhe a atenção imediata de outra, de forma que a primeira possa finalizar sua tarefa
- A CPU:
 - envia um comando para o módulo de E/S e passa a executar outra tarefa;
 - quando a operação for concluída, o módulo de E/S interrompe a CPU; e
 - a CPU executa a troca de dados, liberando o módulo de E/S e retomando o processamento anterior.

[25]



[26]

Dispositivos de E/S Princípios de Hardware

- 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.

[27]

Dispositivos de E/S Princípios de Hardware

E/S via Interrupção

- O maior problema no uso de interrupções: geralmente se dispõe de poucas linhas de interrupção ligadas diretamente ao processador
- Usualmente: são assinalados números para as interrupções, onde o menor número tem prioridade sobre o maior

[28]

Dispositivos de E/S Princípios de Hardware

E/S via Interrupção

- Exemplo de mapeamento das interrupções em um sistema IBM compatível

Int	Dispositivo	Int	Dispositivo
0	Cronômetro do sistema	9	Porta de comunicação COM3
1	Teclado	10	Porta de comunicação COM2
2	Controlador de interrupção	11	Ponte PCI (*)
4	Porta de comunicação COM1	12	Mouse porta PS/2 (*)
5	Placa de som (*)	13	Coprocessador numérico
6	Controlador de disco flexível	14	Controlador IDE/ESDI
7	Porta de Impressora LPT1	15	Controlador IDE/ESDI
8	CMOS/Relógio do sistema		(*) Opções não padronizadas

Mapa de Interrupções num IBM-PC compatível

[29]

Dispositivos de E/S Princípios de Hardware

E/S via Acesso Direto à Memória

- Inconvenientes das técnicas anteriores:
 - limitam a capacidade de transferência da CPU, entre o módulo de E/S e a Memória Principal
 - uso de mais de uma instrução
 - CPU fica ocupada no gerenciamento
 - se a quantidade de dados for grande, o desempenho do sistema será comprometido
- A solução deste problemas: permitir o acesso direto à memória
 - método propõe o uso de uma única interrupção, para efetuar a transferência de um bloco de dados entre o periférico e a memória principal
 - CPU tem envolvimento mínimo no gerenciamento

[30]

Dispositivos de E/S Princípios de Hardware

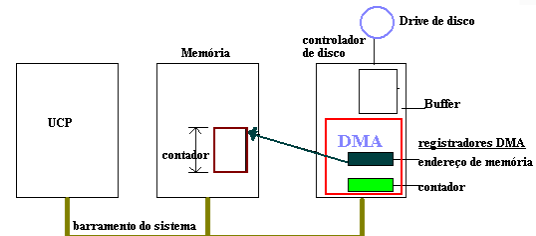
E/S via Acesso Direto à Memória

- Necessidade de um módulo adicional: o Controlador de DMA
- Operação do Controlador de DMA:
 - CPU envia comando (leitura ou escrita) para o controlador de DMA
 - CPU continua seu trabalho
 - O controlador de DMA, para acessar memória, "rouba" ciclos da CPU, atrasando-a apenas
 - Ao final da operação, o controlador de DMA aciona a interrupção para sinalizar o término da operação
 - A CPU pode executar a rotina de tratamento da interrupção, processando os dados lidos ou produzindo novos dados para serem escritos

{ 31 }

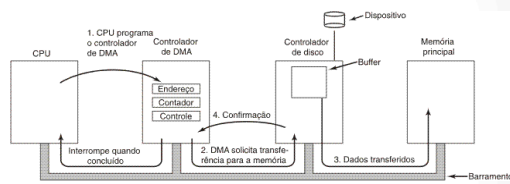
Dispositivos de E/S Princípios de Hardware

E/S via Acesso Direto à Memória (DMA)



{ 32 }

Acesso Direto à Memória (DMA)

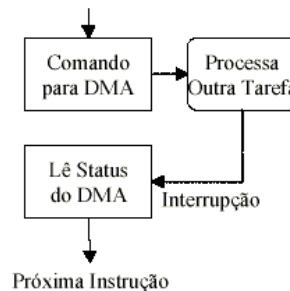


Operação de uma transferência com DMA

{ 33 }

Dispositivos de E/S Princípios de Hardware

E/S via Acesso Direto à Memória (DMA)



{ 34 }

Dispositivos de E/S Princípios de Hardware

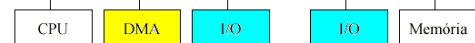
E/S via Acesso Direto à Memória

- O controlador de DMA pode suportar, tipicamente, o trabalho com vários periféricos diferentes, cada um utilizando um canal de DMA (*DMA channel*)
- Outra vantagem do DMA: pode ser implementada em hardware de diversas formas diferentes, conforme a quantidade de dispositivos e o desempenho pretendido

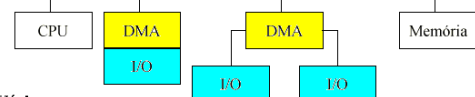
{ 35 }

Configurações de DMA

Barramento Único - DMA Desacoplado

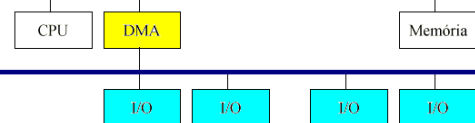


Barramento Único - DMA e I/O integrados



Vários barramentos

- DMA Desacoplado



{ 36 }

Dispositivos de E/S Princípios de Hardware

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

{ 37 }

Dispositivos de E/S Princípios de Hardware

- 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 CPU.

{ 38 }

Dispositivos de E/S Princípios de Hardware

- DMA contém vários registradores que podem ser lidos e escritos pela CPU:
 - 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.

{ 39 }

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 CPU;
 - 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 CPU.

{ 40 }

Dispositivos de E/S Princípios de Hardware

- Com DMA: Leitura de um bloco de dados em um disco: CPU controla
 1. Além do endereço a ser lido, a CPU 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;
 3. 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);

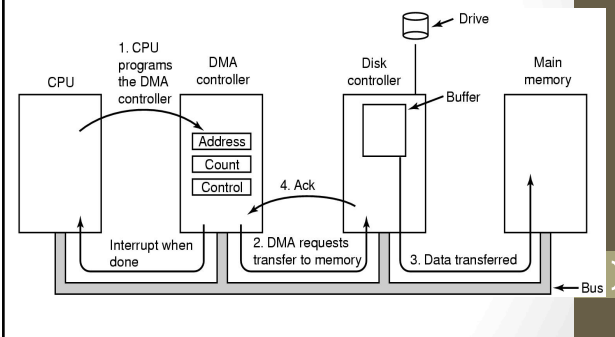
{ 41 }

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 CPU;
- Quando o SO inicia o atendimento à interrupção, o bloco de dados já está na RAM.

{ 42 }

Dispositivos de E/S Princípios de Hardware



[44]

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 precisa de um *buffer* interno?
Por que 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.

[45]

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 CPU;
 - Ignora no momento → dispositivos geram sinais de interrupção até serem atendidos.

[46]

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.

[47]

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.

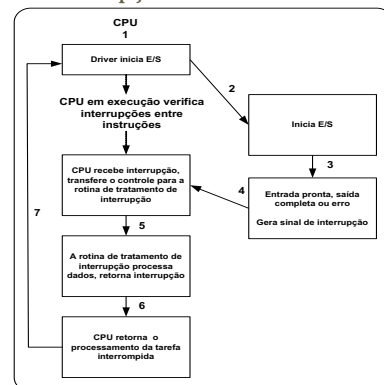
[48]

Dispositivos de E/S Tratando Interrupções

- Ciclo de instrução com interrupção: CPU
 - 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;

{ 49 }

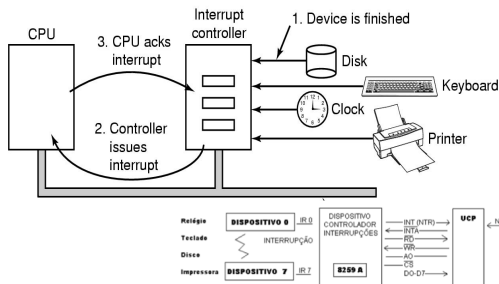
Dispositivos de E/S Tratando Interrupções



{ 50 }

Dispositivos de E/S Princípios de Hardware

- Como uma interrupção ocorre:



{ 51 }

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

{ 52 }

Dispositivos de E/S Princípios de Hardware

- Como é possível a um módulo de E/S controlar mais de um dispositivo de E/S: necessidade de associação de endereços a estes dispositivos
- Existem 2 formas de interpretação de endereços, pelo módulo de E/S, quando o barramento é compartilhado:
 - Mapeado em Memória: o módulo de E/S opera dentro do espaço de endereçamento de memória, usando um conjunto de endereços reservados (registradores são tratados como posições de memórias);
 - Mapeado em E/S ou E/S isolada: existe um espaço de endereçamento independente para os dispositivos de E/S. Uso de instruções especiais de E/S.

{ 53 }

Gerência de E/S

- Nos PCs: é utilizado um espaço de endereçamento especial para a E/S, com cada controlador alocado em certa posição da mesma.

Controladores de E/S	Endereços de E/S
Timer (relógio do sistema)	040-043H
Teclado	060-063H
...	...

{ 54 }

Gerência de E/S

Tipos de E/S

• Os dispositivos de E/S podem ser classificados de forma ampla, sendo que as mais utilizadas são quanto ao:

- tipo de conexão
- tipo de transferência de dados
- tipo de compartilhamento de conexões

• Quanto ao tipo de conexão:

- Leva em consideração a natureza da conexão entre o módulo de E/S e o periférico
- Do ponto de vista dos dados, as conexões são projetadas para operação:
 - Serial
 - Paralela

{ 55 }

Gerência de E/S

Conexão serial:

- Uma única linha de sinal é utilizada para o estabelecimento de toda a conexão, protocolo e transferência de dados, entre o módulo de E/S e o periférico
- Características principais:
 - mais barata que a paralela
 - mais lenta que a paralela
 - relativamente confiáveis
 - usada em dispositivos mais baratos e lentos, como impressoras e terminais

{ 56 }

Gerência de E/S

Conexão paralela:

- Várias linhas de sinais são usadas, de forma que vários bits de dados possam ser transferidos em paralelo
- É comum que existam linhas independentes para tráfego de sinais de controle
- Características principais:
 - mais complexa que a serial
 - mais cara
 - mais rápida
 - altamente confiável
 - usada em dispositivos mais velozes, como unidades de disco, fita ou impressoras rápidas

{ 57 }

Gerência de E/S

Classificação quanto ao compartilhamento de conexões

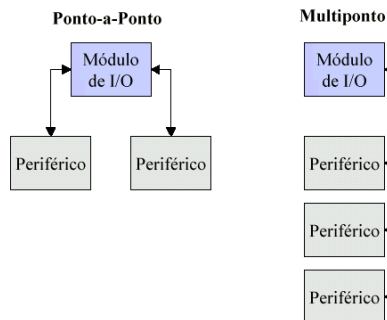
Podem ser divididos em 2 categorias:

- Ponto-a-Ponto – é a conexão mais simples, onde existe um conjunto de linhas dedicadas para a ligação entre o módulo de E/S e cada periférico.
- Multiponto – neste tipo de conexão, um módulo de E/S compartilha um conjunto de linhas de sinais entre diversos periféricos.

{ 58 }

Gerência de E/S

Classificação quanto ao compartilhamento de conexões



{ 59 }

Gerência de E/S

Conexões Ponto-a-Ponto:

- oferecem maior confiabilidade
 - permite a operação simultânea de diversos dispositivos
 - é usada em dispositivos mais simples, tais como modems, teclado e impressora
- Tem-se os seguintes exemplos de conexões ponto-a-ponto padronizadas, usados em comunicação de curta distância, usualmente na interface padrão RS - 232C:
- Protocolo RTS/CTS (*Request to Send/Clear to Send*)
 - Protocolo Xon/Xoff (*Transmission On/Transmission Off*)

{ 60 }

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.

67

Dispositivos de E/S Princípios de Software

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

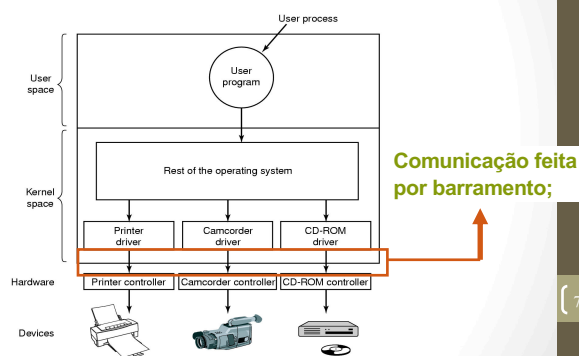
68

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;

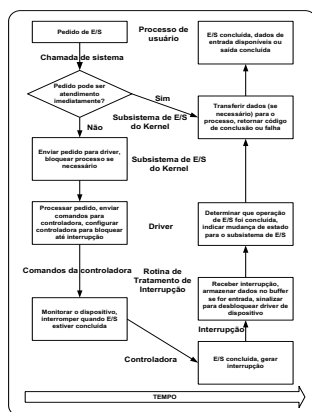
69

Dispositivos de E/S Princípios de Software - Camadas



70

Dispositivos de E/S - Ciclo de E/S



71

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*;

72

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;

{ 73 }

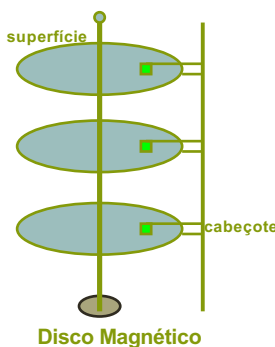
Dispositivos de E/S - Ciclo de E/S

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 CPU, ele retoma a execução na conclusão da chamada ao sistema.

{ 74 }

Dispositivos de E/S - Discos

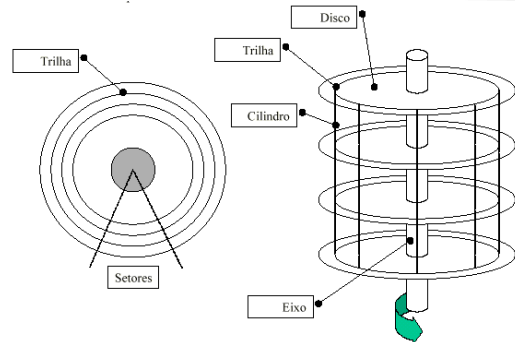


- Cada superfície é dividida em **trilhas**;
- Cada trilha é dividida em **setores** ou **blocos** (512 bytes a 32K);
- Um conjunto de trilhas (com a mesma distância do eixo central) formam um **cilindro**;
- Cabeças de leitura e gravação;
- **Tamanho do disco:**
 n° cabeças (faces) x n° cilindros (trilhas) x n° setores x tamanho_setor.

{ 75 }

Gerência de E/S

Organização dos Discos



{ 76 }

Dispositivos de E/S - Discos

- Discos Magnéticos:
 - Grande evolução com relação
 - 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.

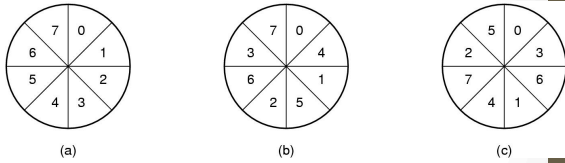
{ 77 }

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.

{ 78 }

Formatação de Disco (Torção Cilíndrica)

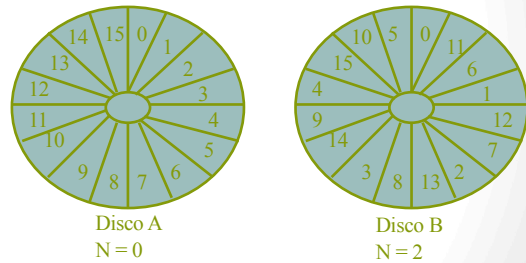


- a) Sem entrelaçamento
- b) Entrelaçamento simples
- c) Entrelaçamento duplo

[79]

Dispositivos de E/S - Discos

Trilhas com 16 setores



[80]

Dispositivos de E/S - Discos

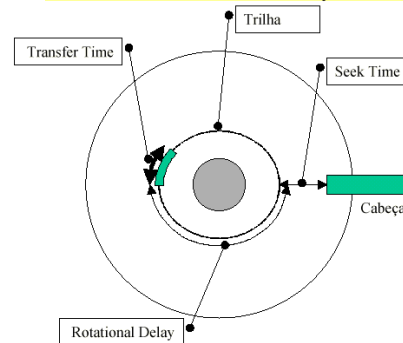
- Drivers de Disco:
 - Fatores que influenciam tempo para leitura/escrita no disco:
 - Velocidade de acesso (*seek*) → tempo para o movimento do braço até o cilindro;
 - Delay de rotação (latência) → tempo para posicionar o setor na cabeça do disco;
 - Tempo da transferência dos dados;
 - Tempo de acesso: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

[81]

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

Dispositivos de E/S

$$t_{access} = t_{seek} + t_{rotationaldelay} + t_{transfer}$$



[82]

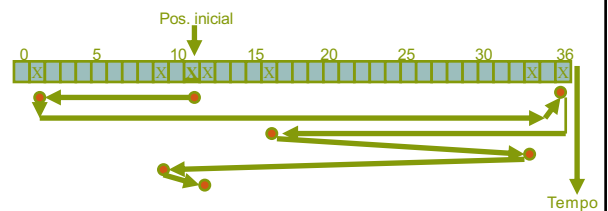
Dispositivos de E/S - Discos

- 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.

[83]

Dispositivos de E/S - Discos

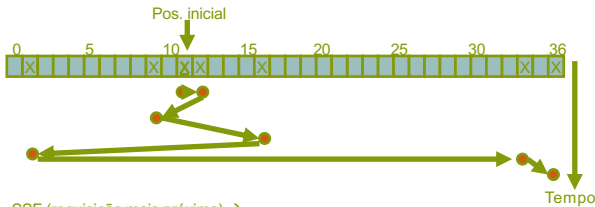
Disco com 37 cilindros;
Lendo bloco no cilindro 11;
Requisições: 1,36,16,34,9,12, nesta ordem



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

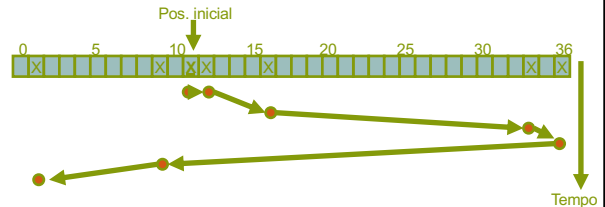


SSF (requisição mais próxima) →
atendimento: 12,9,16,1,34,36;
movimentos do braço (número de cilindros): 1,3,7,15,33,2 = 61;

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



Elevador (requisições na mesma direção) →
atendimento: 12,16,34,36,9,1
movimentos do braço (número de cilindros): 1,4,18,2,27,8 = 60;

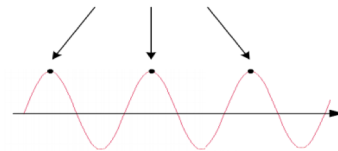
Clocks (Timers)

- Dois tipos de relógios:
 - hardware (*clock hardware*) e software (*clock driver*);
- Clock Hardware:**
 - Dispositivo que gera pulsos síncronos;
 - Localizados na CPU ou na placa-mãe;
 - Sinal utilizado para a execução de instruções;
 - Presente em qualquer sistema multiprogramado;
 - Fundamental para ambientes *TimeSharing*;
 - Frequência de *clock*
 - Número de vezes que o pulso se repete por segundo (Hz).

{ 87 }

Clocks – Tipos Hardware

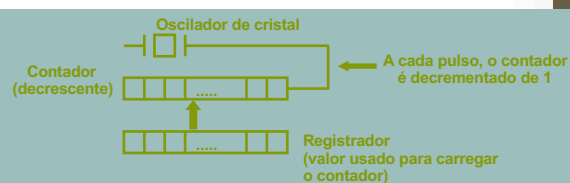
- Hardware**
 - Dois tipos:
 - Básico: usa o sinal da rede elétrica (110/220 V) para fazer contagem (50/60 Hz) → cada oscilação da rede é uma interrupção.



{ 88 }

Clocks – Tipos Hardware

- Com 03 componentes → oscilador de cristal, contador e registrador;
 - Programável;
 - Contador recebe o valor armazenado no registrador;
 - A cada pulso do oscilador, o contador é decrementado de uma unidade;
 - Quando o contador zera, é gerada uma interrupção de *clock* (interrupção da CPU);
 - Precisão.



{ 89 }

Clocks – Tipos Hardware

- Relógios programáveis podem operar de diversos modos:
 - One-shot mode*
 - Ao ser iniciado, o relógio copia o valor contido no registrador, e decrementa o contador a cada pulso do cristal;
 - Quando o contador chega a zero, uma interrupção ocorre;
 - Recomeça por intervenção de software.

{ 90 }

Clocks – Tipos

Hardware

- *Square-wave mode*
 - Repete o ciclo automaticamente, sem intervenção de software;
- As periódicas interrupções geradas pela CPU são chamadas de *clock ticks* (pulsos do relógio).

[91]

Clocks – Tipos

Software

- Hardware → gera interrupções em intervalos conhecidos (*clock ticks*);
- Tudo o mais é feito por Software: *clock driver*;
- Funções do *clock driver*:
 - Manter a hora do dia;
 - Evitar que processos executem por mais tempo que o permitido;
 - Supervisionar o uso da CPU;
 - Cuidar da chamada de sistema `alarm`;
 - Fazer monitoração e estatísticas;
 - Prover temporizadores “guardiões” para os dispositivos de E/S.

[92]

Software

Manter a Hora do Dia

- Hora e data correntes:
 - Checa a *CMOS*;
 - Uso de baterias para não perder as informações
 - Pergunta ao usuário;
 - Checa pela rede em algum *host* remoto;
- Número de *clock ticks*:
 - Desde às 12 horas do dia 1º de janeiro de 1970 no UNIX;
 - Desde o dia 1º de janeiro de 1980 no Windows.

[93]

Software

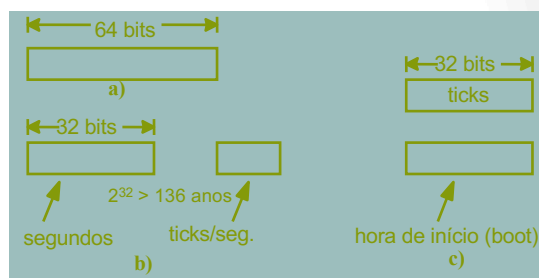
Manter a Hora do Dia

- Incrementar contador a cada *tick*;
- Três abordagens:
 - a) Contador com 64 bits → alto custo;
 - b) Contar em segundos → *ticks/seg*;
 - c) *Ticks* relativos à hora que o sistema foi iniciado.

[94]

Software

Manter a Hora do Dia



[95]

Software

Controlar duração da Execução dos Processos

- Execução inicia → escalonador inicia contador → número de *ticks* do *quantum*;
- Contador é decrementado a cada *tick*;
- Contador = 0 → hora de acionar escalonador (que pode trocar o processo).

[96]

Software

Supervisão do uso da CPU

- Quanto tempo o processo já foi executado?
 - Processo inicia → novo *clock* (segundo relógio) é iniciado;
 - Processo é parado → *clock* é lido;
 - Durante interrupções → valor do *clock* é salvo e restaurado depois;
- Possível usar a tabela de processos → variável global armazena o tempo (em *ticks*).

[97]

Software

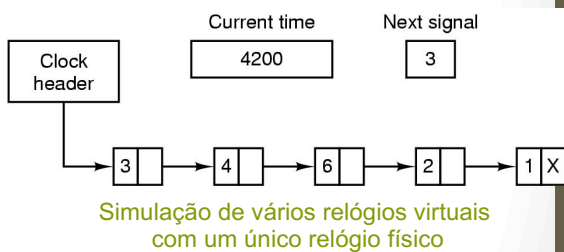
Alarmes (Avisos)

- Processos podem requerer “avisos” de tempos em tempos;
- Avisos podem ser: um sinal, uma interrupção ou uma mensagem;
 - Exemplo:
 - redes de computadores → pacotes não recebidos devem ser retransmitidos;
- Uma lista encadeada com os tempos dos alarmes pendentes é mantida:
 - Simulação de vários relógios virtuais em um único relógio físico.

[98]

Software

Alarmes (Avisos)



Tempos dos alarmes: 4203, 4207, 4213, 4215, 4216;

[99]

Software

Temporizadores Guardiões

- Esperar por um certo tempo e realizar uma tarefa:
 - Δt → registrador (contador);
 - Quando contador zera → procedimento é executado;
- Onde usar?
 - Exemplo:
 - acionador de disco flexível: somente quando o disco está em rotação na velocidade ideal é que as operações de E/S podem ser iniciadas.

[100]

Clocks – Tipos

Software

- Tarefas básicas do *driver* de relógio (*clock driver*) durante uma interrupção:
 - Incrementar o tempo real;
 - Decrementar o *quantum* e comparar com 0 (zero);
 - Contabilizar o uso da CPU;
 - Decrementar o contador de alarme;
 - Gerenciar o tempo de acionamento de dispositivos de E/S.

[101]

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.

[102]

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.

[103]

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.

[104]

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);

[105]

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;

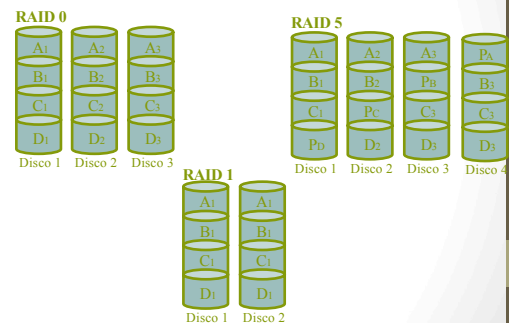
[106]

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;

[107]

Dispositivos de E/S – Discos RAID mais comuns



[108]