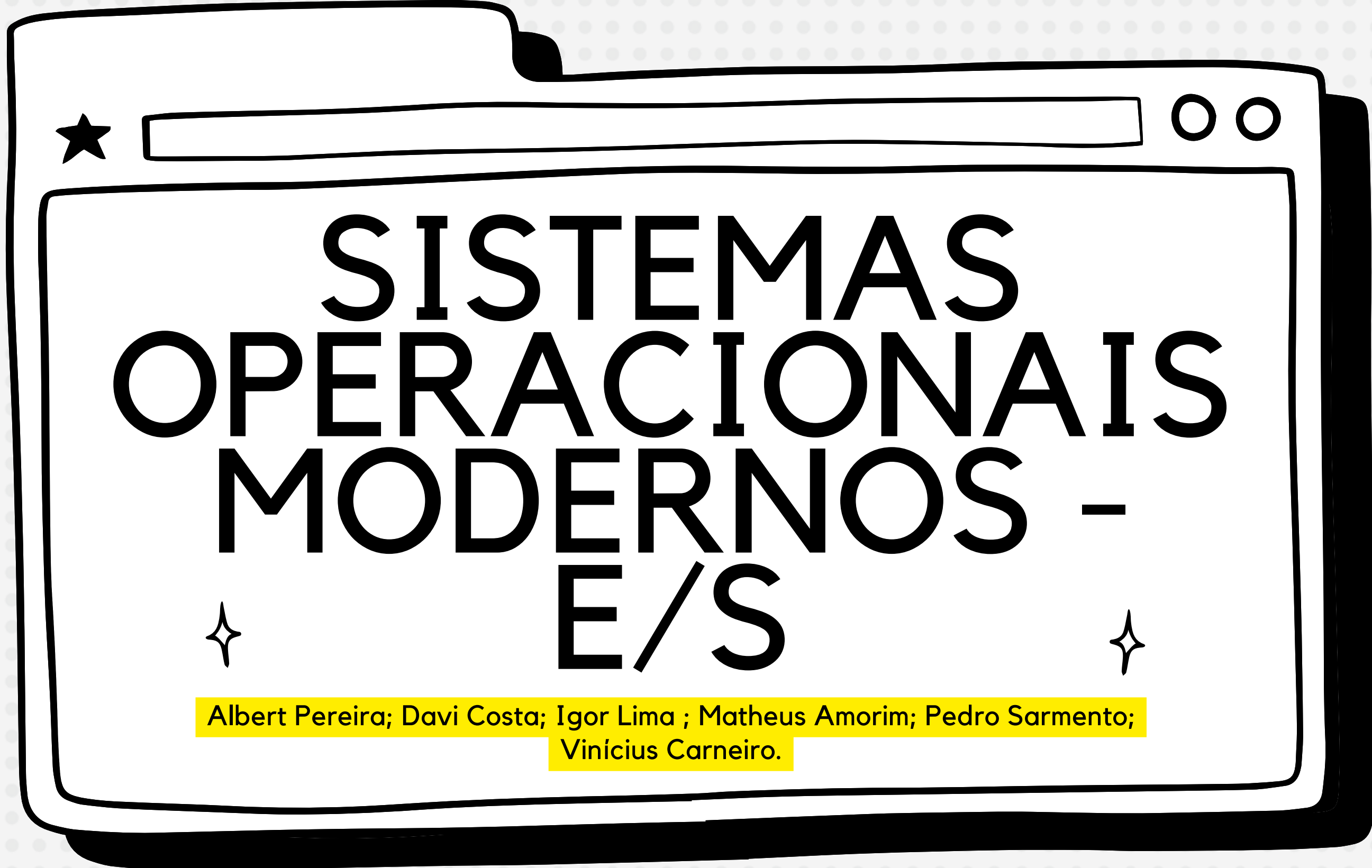
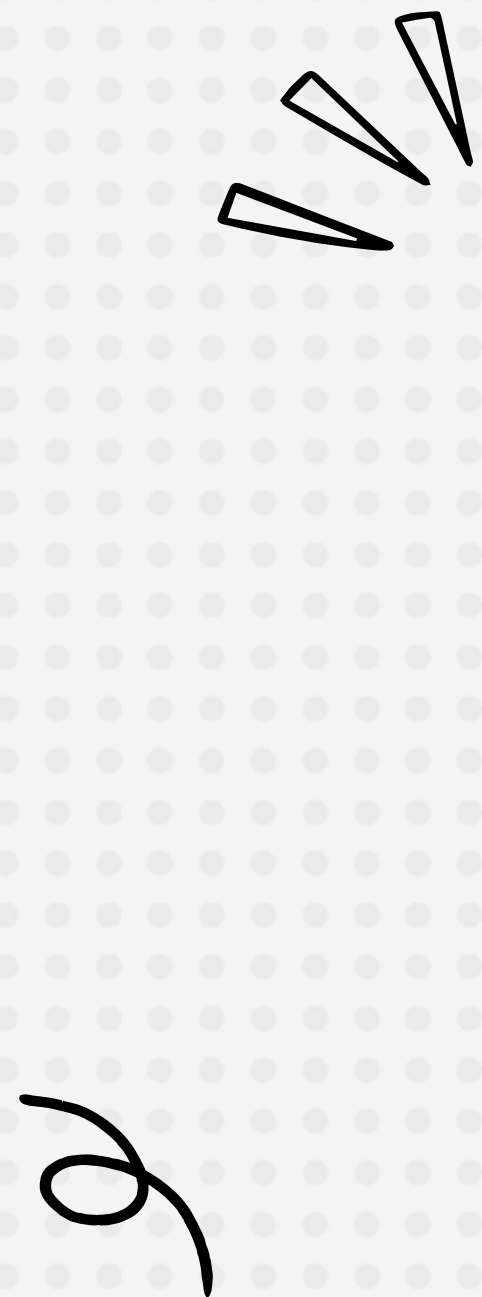
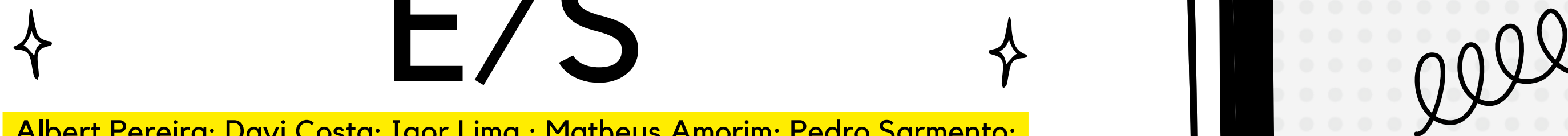
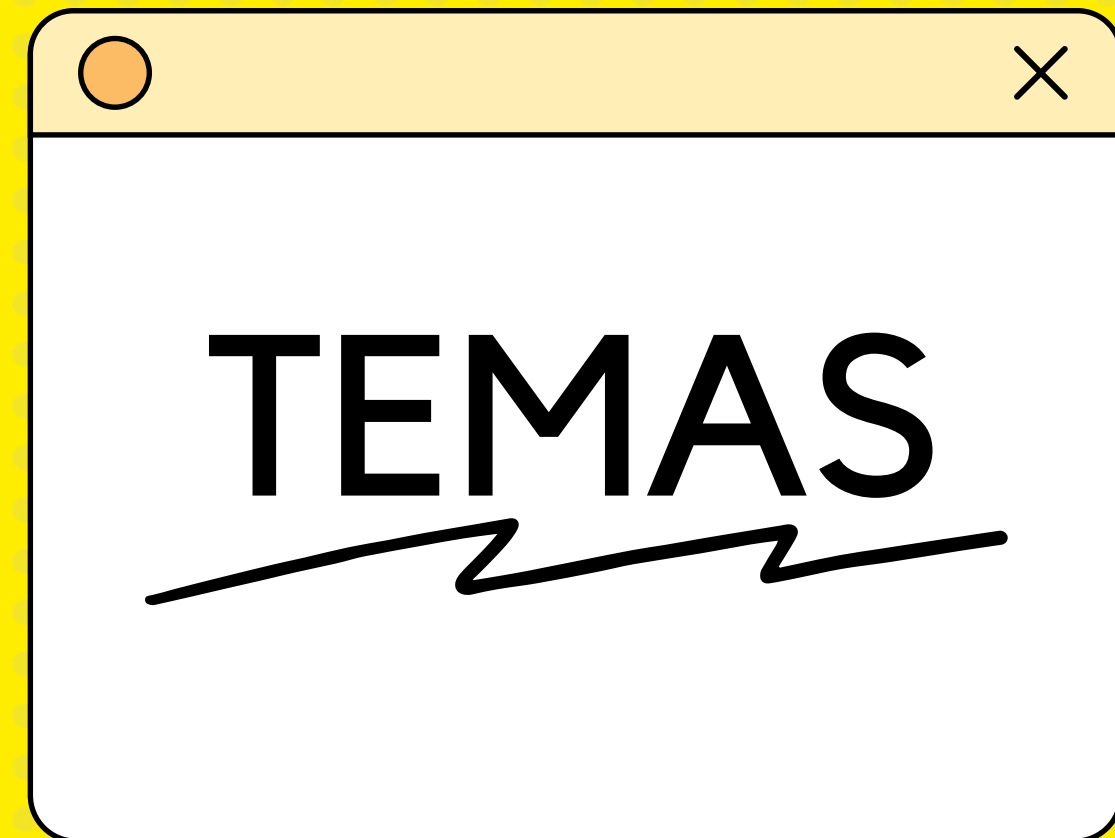




SISTEMAS OPERACIONAIS MODERNOS - E/S

Albert Pereira; Davi Costa; Igor Lima ; Matheus Amorim; Pedro Sarmento;
Vinícius Carneiro.





PRINCÍPIOS
DO
HARDWARE
DE E/S

PRINCÍPIOS
DO SOFTWARE

CAMADAS DE
SOFTWARE

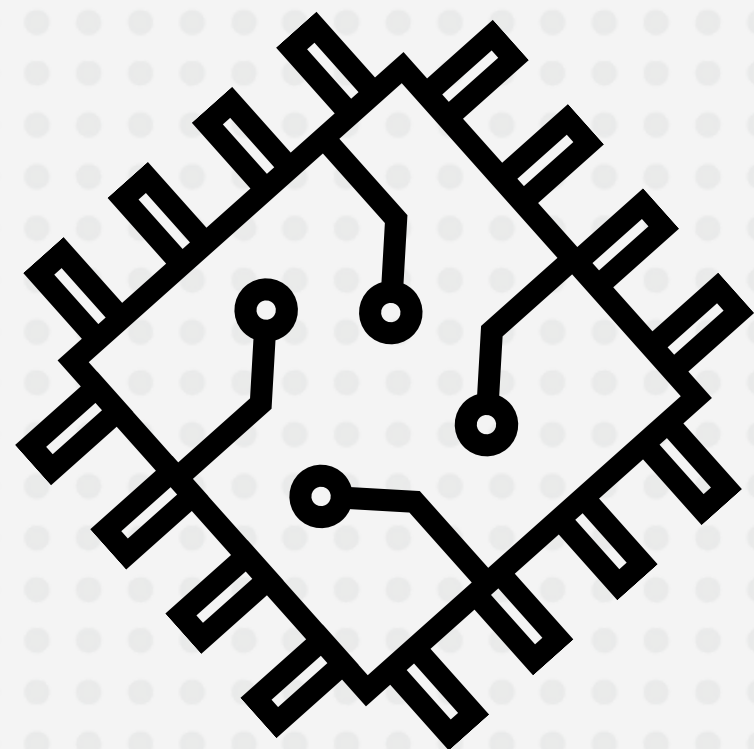
DISCOS

RELÓGIOS

INTERFACE COM
USUÁRIO: TECLADO,
MOUSE, MONITOR

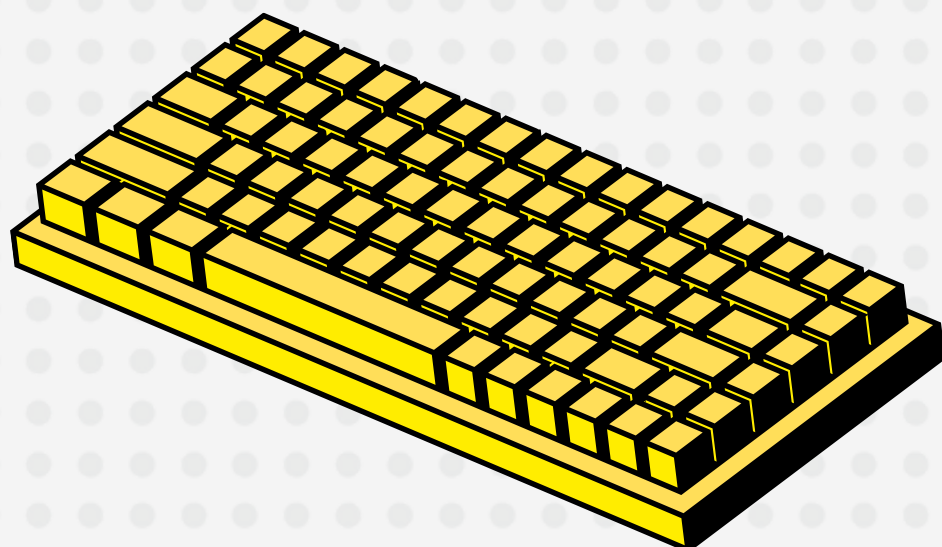
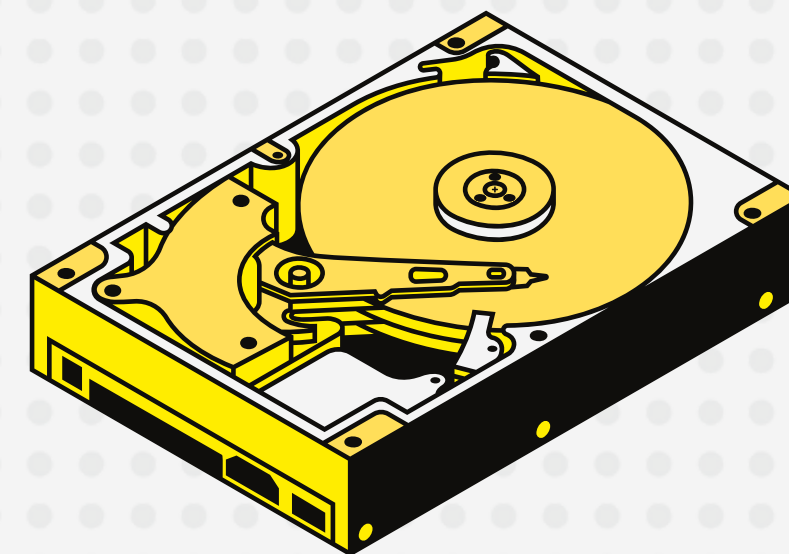
CLIENTES MAGROS
(THIN CLIENTES)

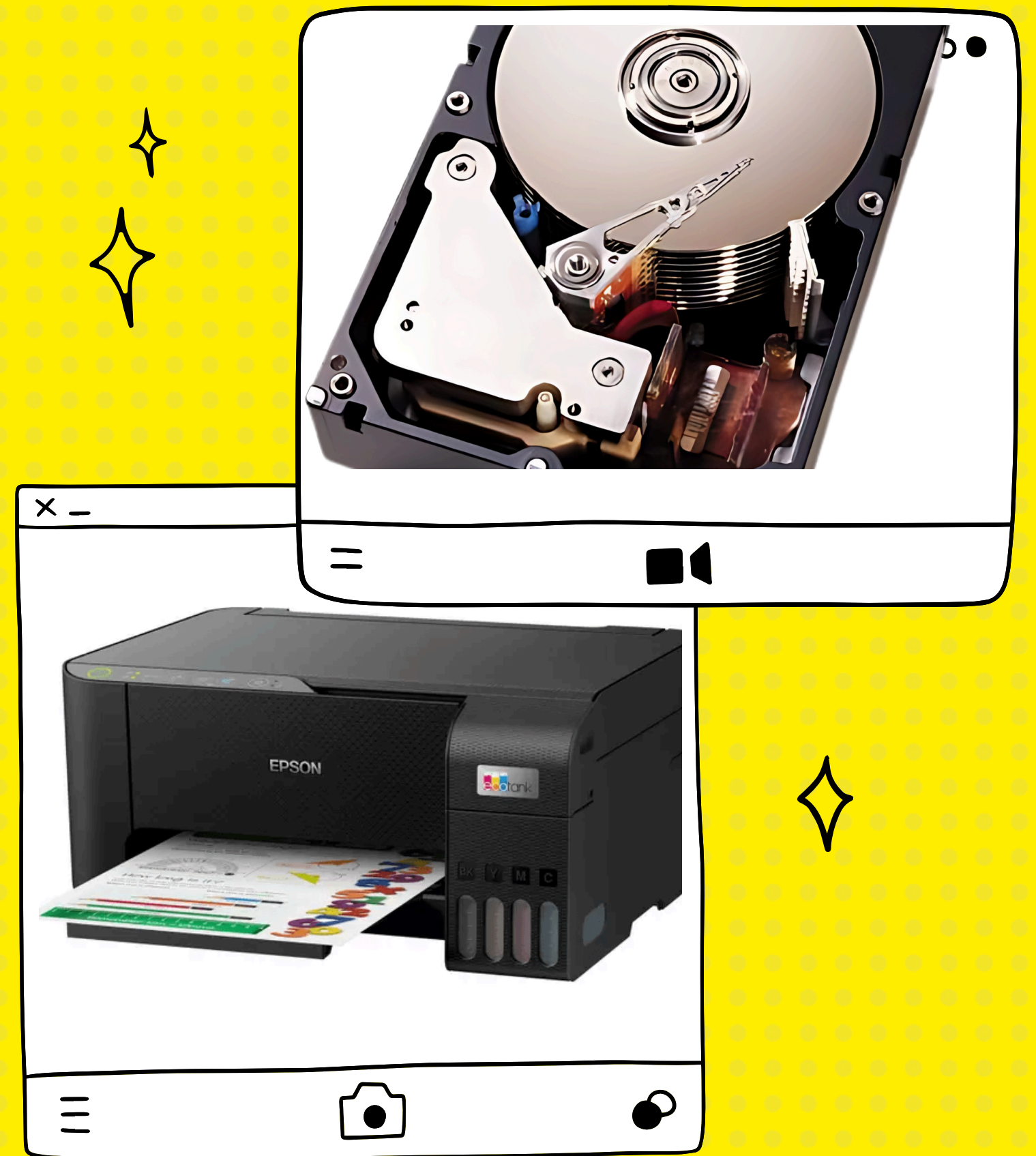
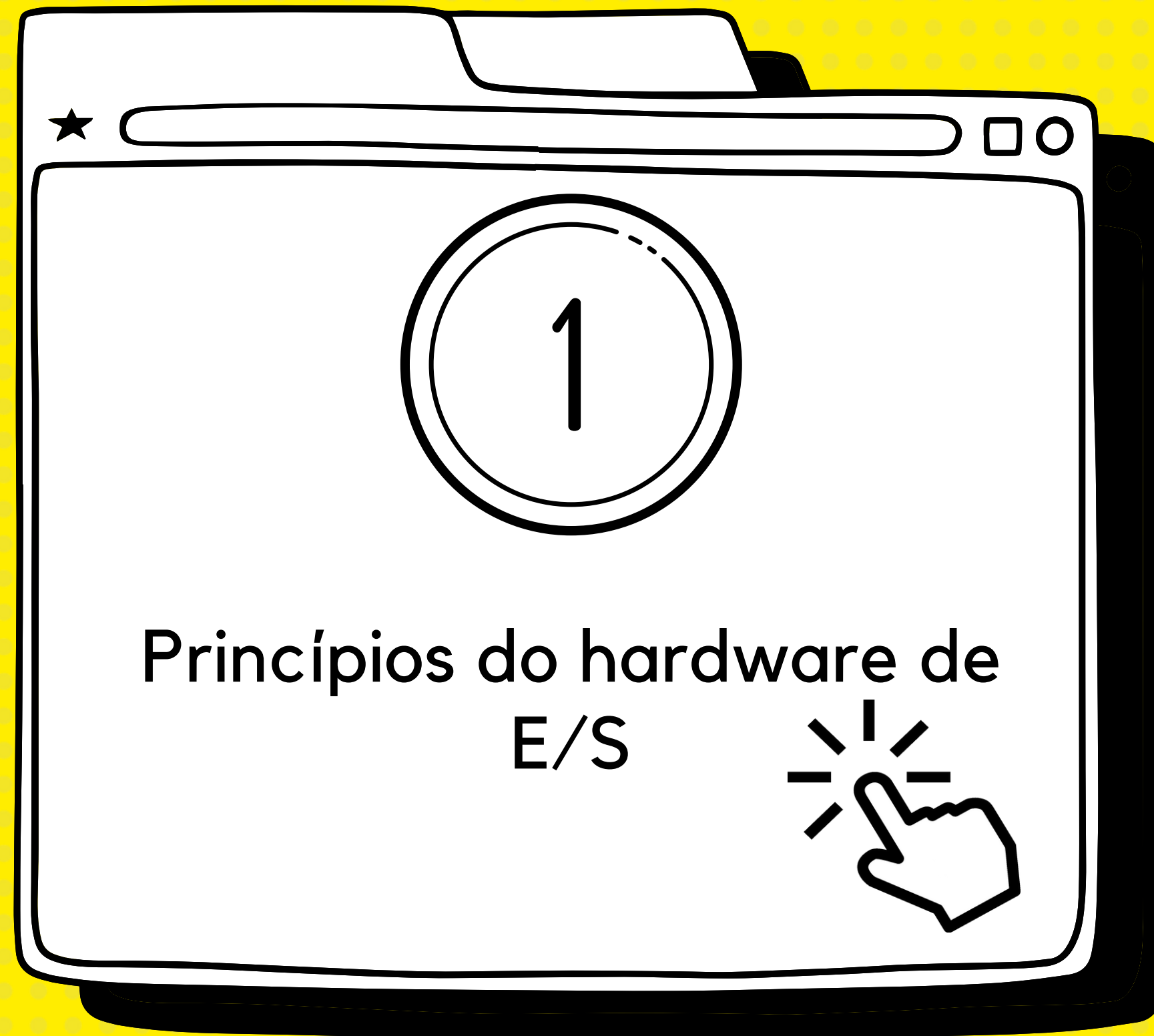
GERENCIAMENTO
DE ENERGIA

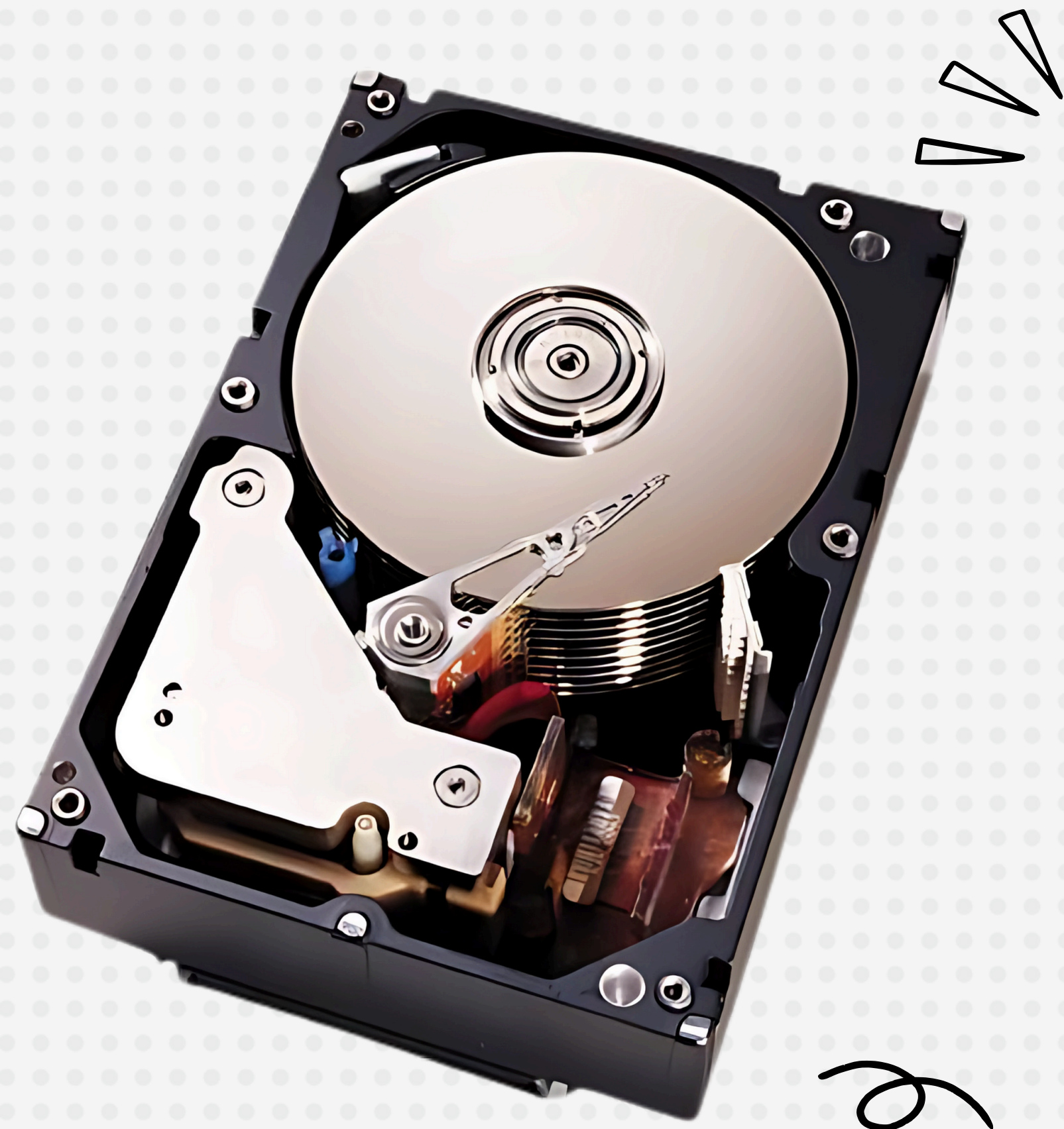


PRINCÍPIOS DO HARDWARE DE E/S

MATHEUS AMORIM







DISPOSITIVO DE BLOCO

Dispositivos de blocos armazenam as informações em blocos de tamanho fixo, cada bloco tendo seu próprio endereço.

Exemplos de dispositivos de bloco: discos rígidos(HDDs), CD-ROMs, pen drives, cartões de memória e SSDs...

DISPOSITIVO DE CARACTERE

Enviam ou recebem um fluxo de caracteres. Não é endereçável, não tem operação de busca e não dispõe de qualquer operação de posicionamento.

Exemplos de dispositivos de caractere: mouses, impressoras, interfaces de redes, projetores e a maioria outros dispositivos diferentes do disco rígido podem ser considerados também

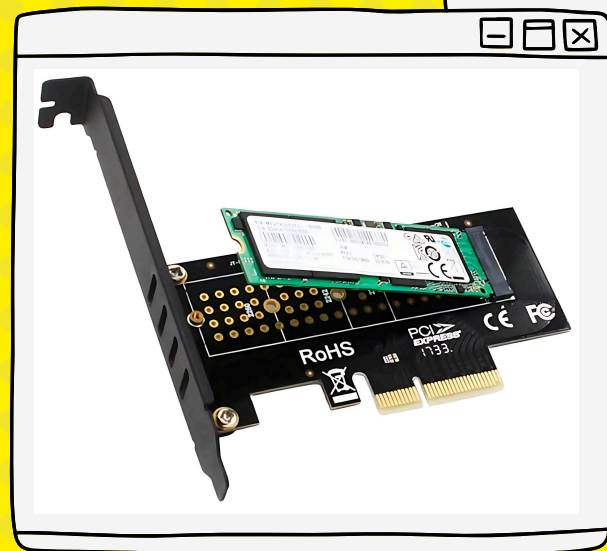
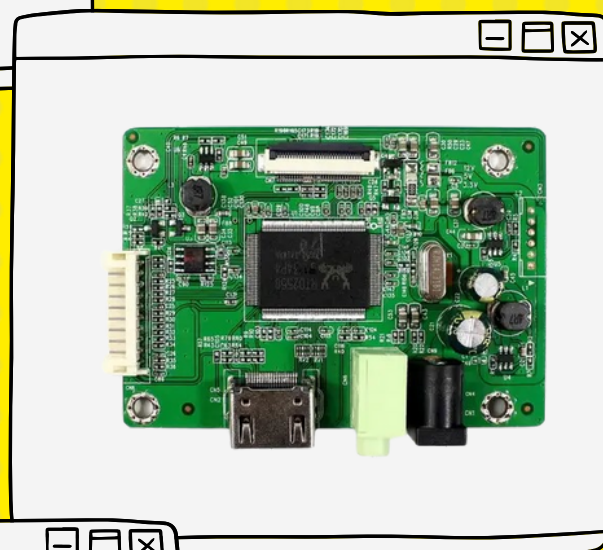
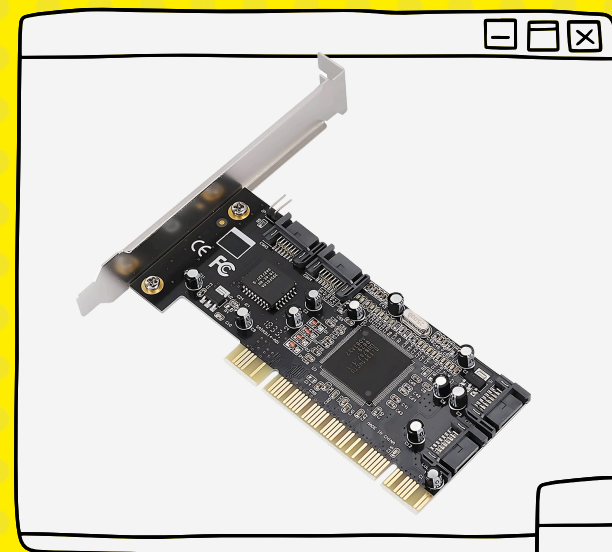




VELOCIDADE NA TRANSEFERENCIA DE DADOS



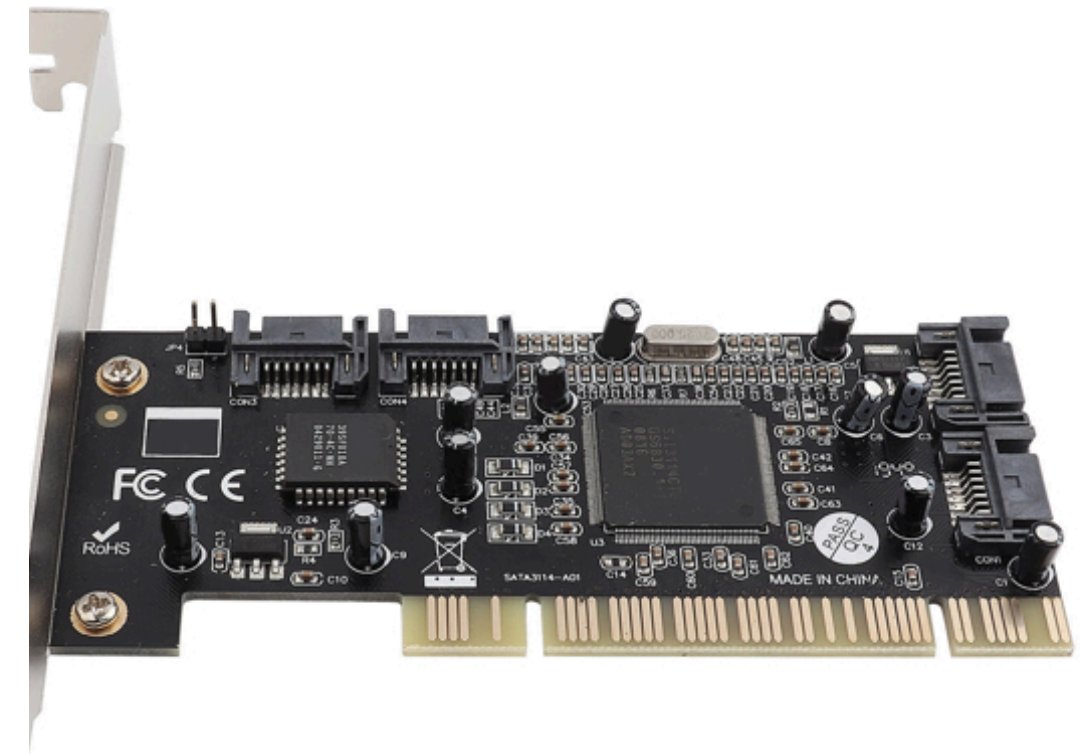
Dispositivos	Taxa de tranferência
Teclado	10 bytes/s
Mouse	100 bytes/s
Scanner	400 KB/s
Filmadora camcorder digital	3,5 MB/s
CD - ROM 52x	7,8MB/s
Cartão flash compacto	40 MB/s
USB 2.0	60 MB/s
Gigabit Ethernet	125 MB/s
Drive de disco SATA	300 MB/s
Barramento PCI	528 MB/s



O controlador ou adaptador,
geralmente se apresenta como
uma placa controladora de
circuito inserida em um
barramento PCI.

Essa placa possui um conector
para cabo que liga ao dispositivo.

PLACA CONTROLADORA SATA PCIE 3.0 X1 DE 4 PORTAS.



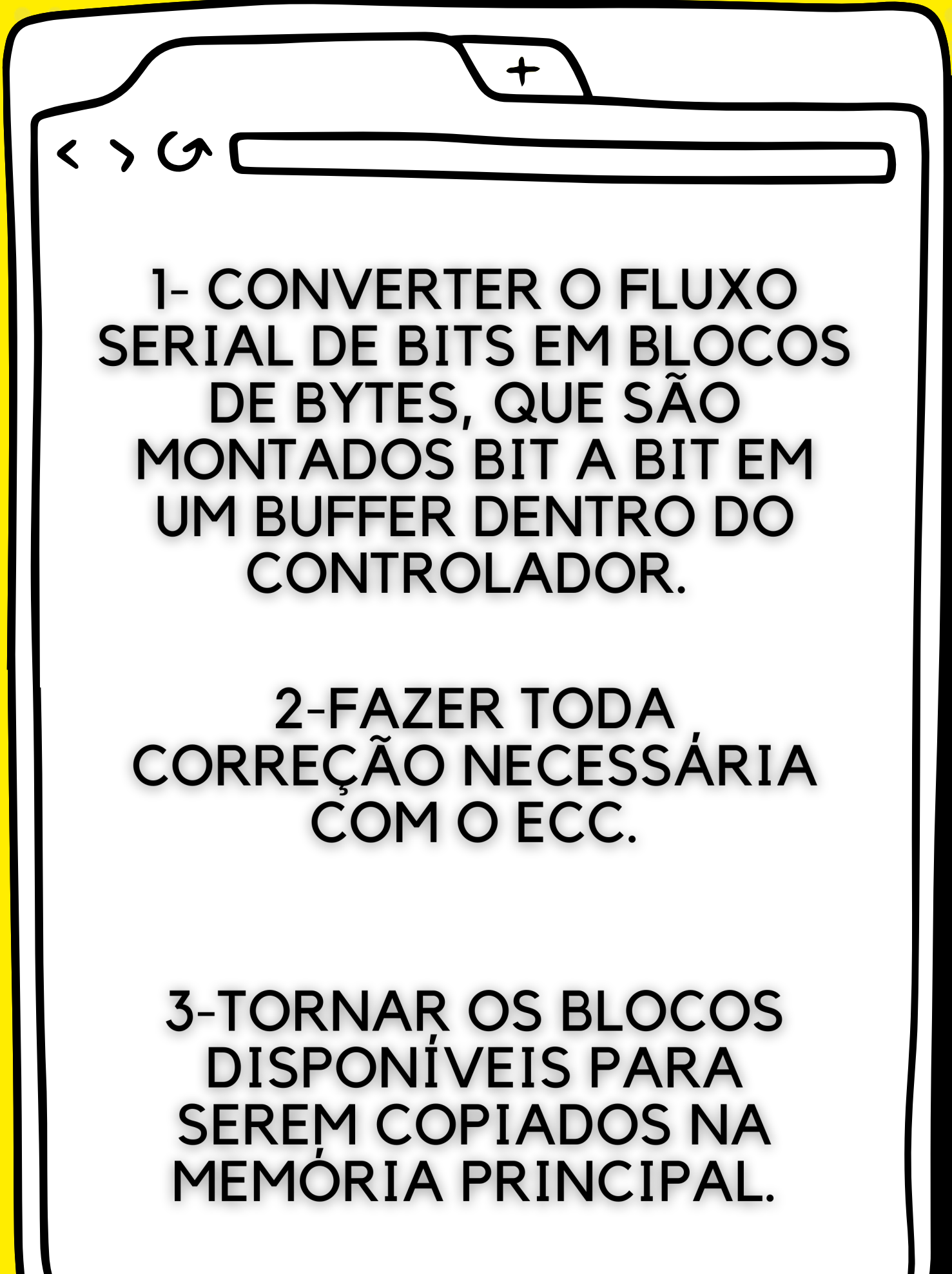


A interface entre o controlador do dispositivo e o dispositivo é com uma frequência de nível muito baixo.

a unidade de disco entrega é um fluxo serial de bits: começando com o preâmbulo e por fim faz o código de correção de erro.



AS TAREFAS DO CONTROLADOR SÃO



1- CONVERTER O FLUXO
SERIAL DE BITS EM BLOCOS
DE BYTES, QUE SÃO
MONTADOS BIT A BIT EM
UM BUFFER DENTRO DO
CONTROLADOR.

2-FAZER TODA
CORREÇÃO NECESSÁRIA
COM O ECC.

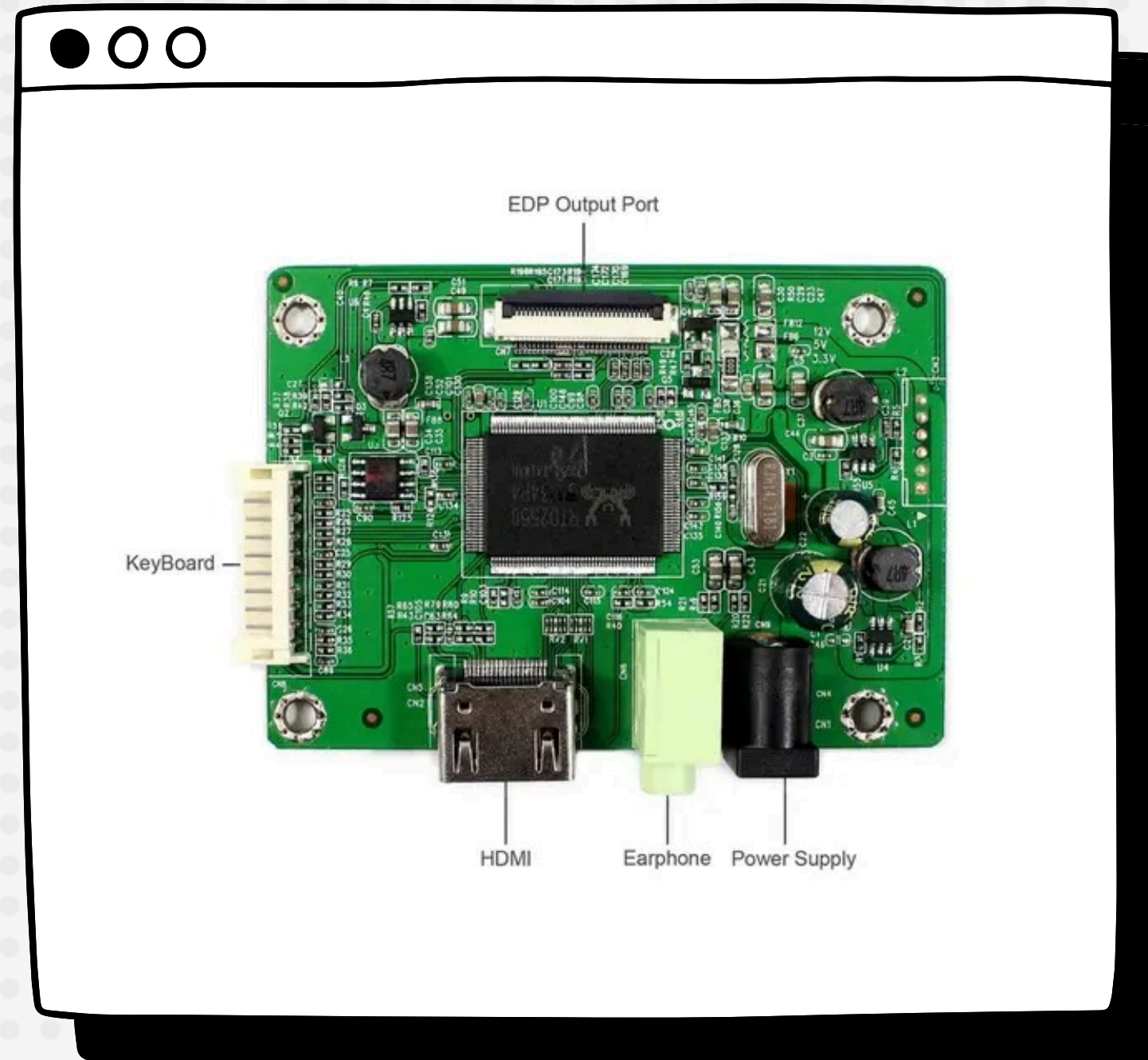


3-TORNAR OS BLOCOS
DISPONÍVEIS PARA
SEREM COPIADOS NA
MEMÓRIA PRINCIPAL.



Mais recente

Mais antigo



Controlador LCD

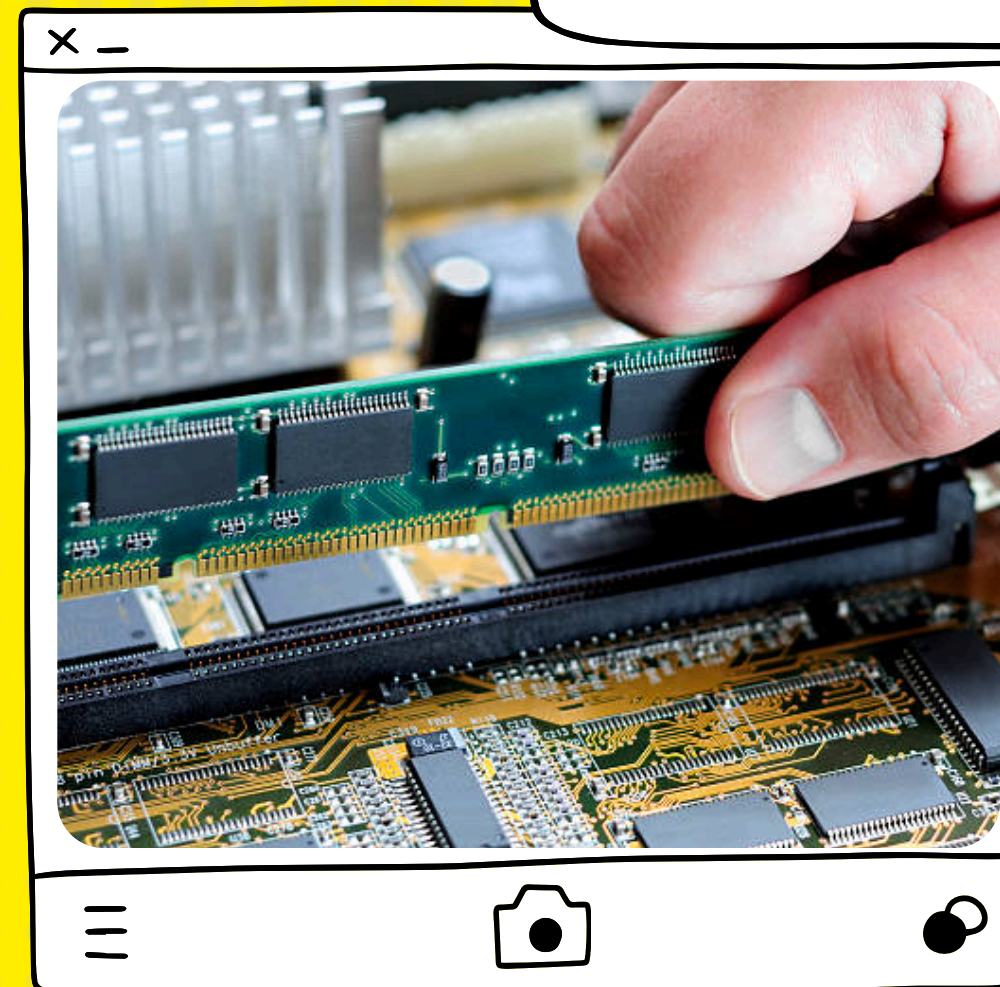
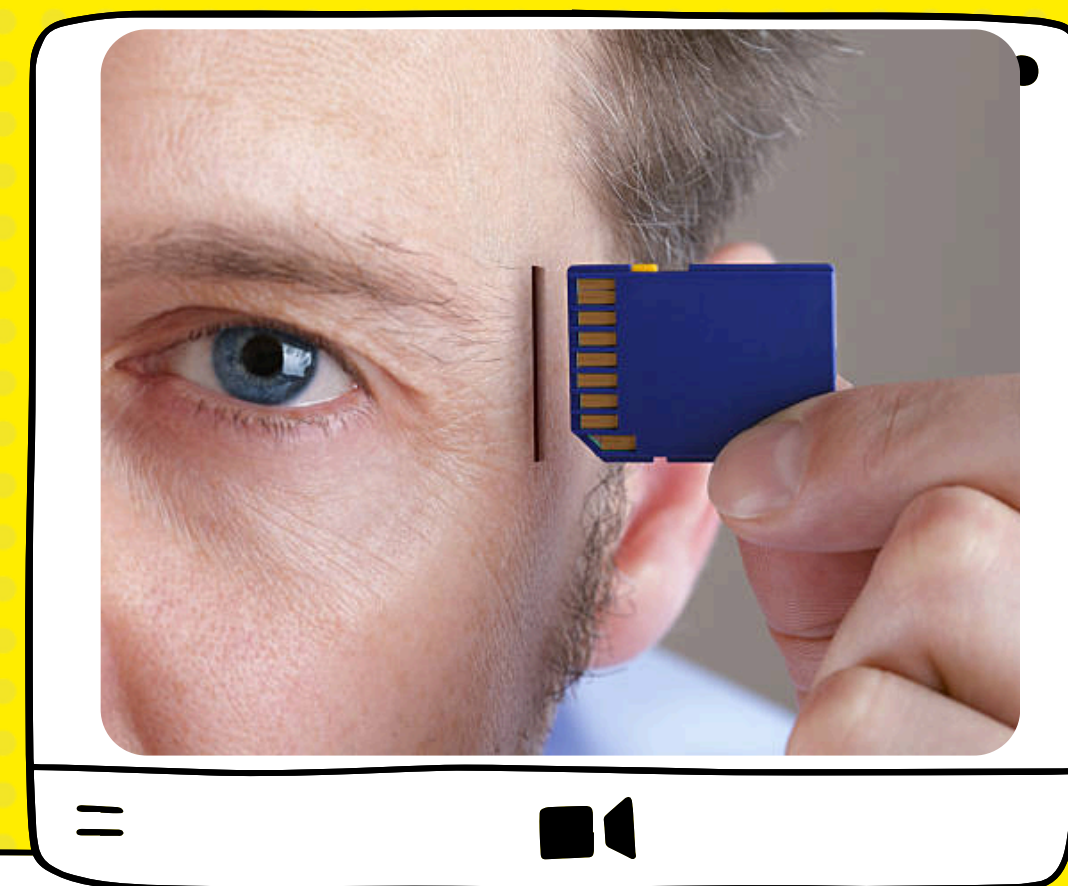
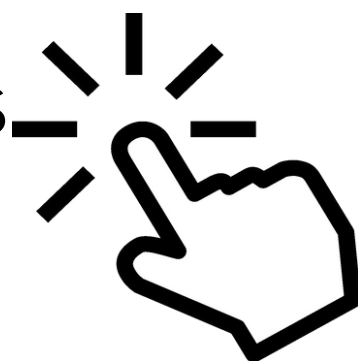


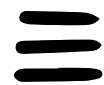
Monitor CRT

3

E/S mapeada na memória

UMA TABELA DE ENDEREÇOS

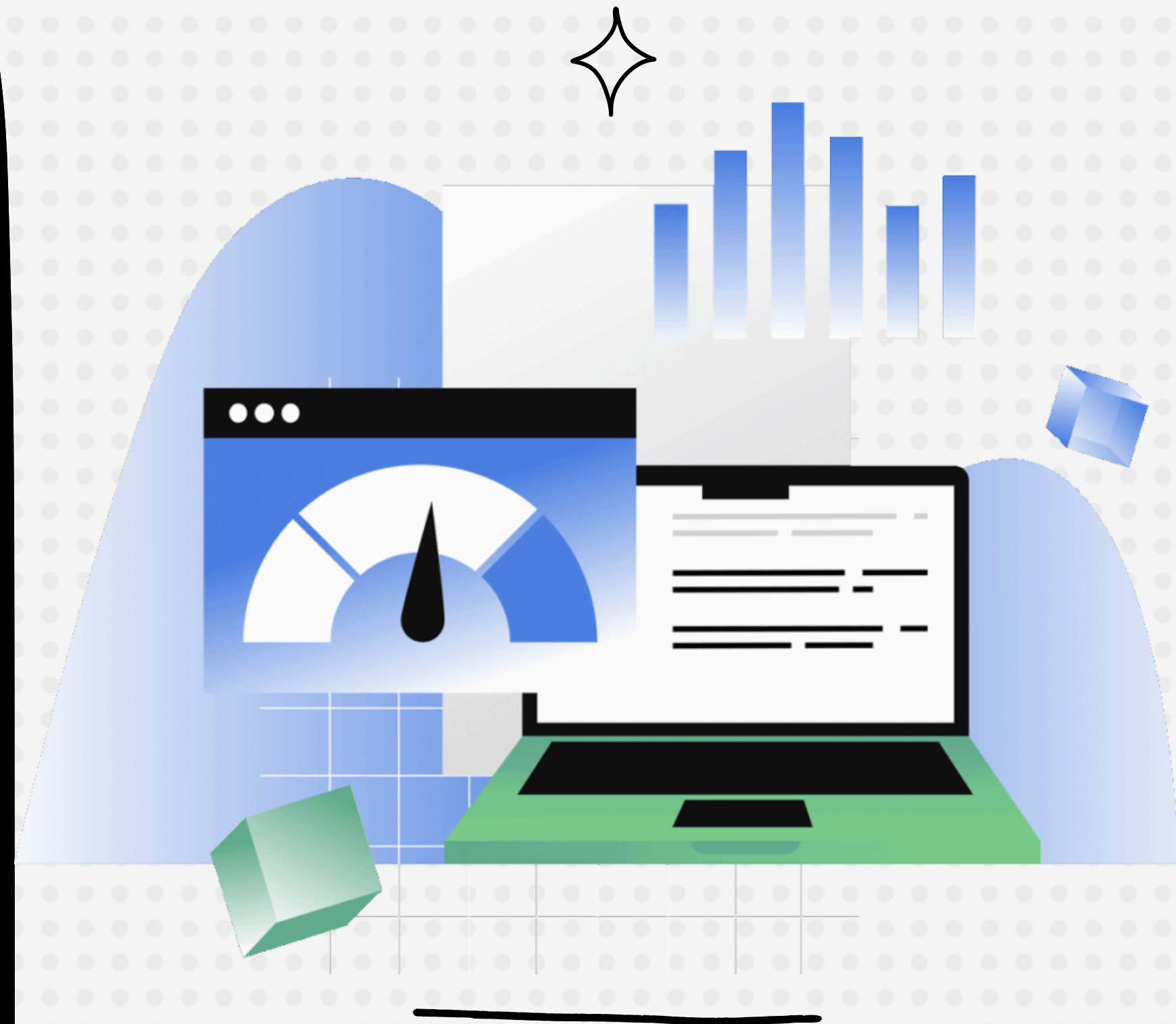




Cada controlador possui registradores usados para a comunicação com a CPU.

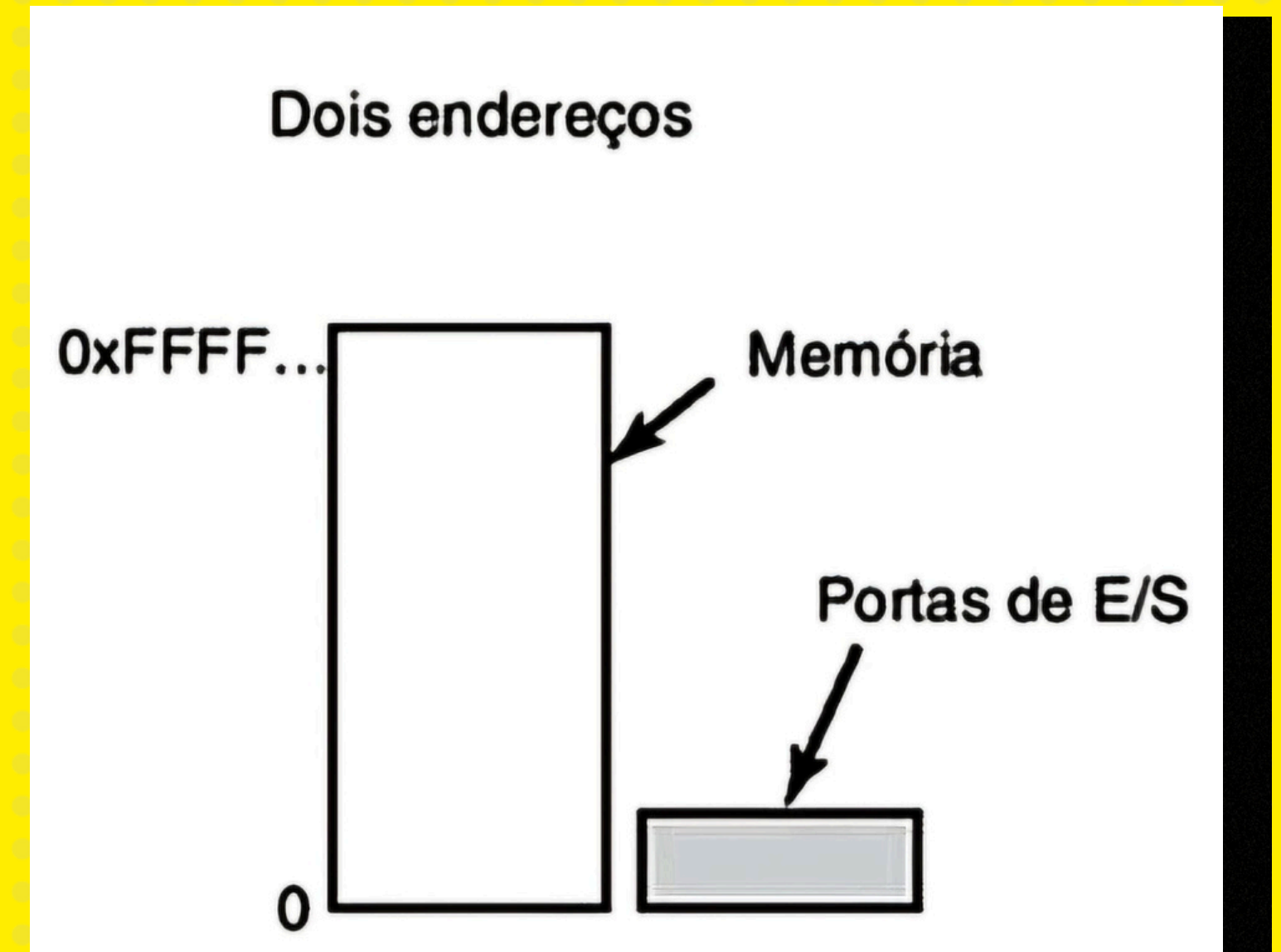
O S.O pode comandar o dispositivo para entregar ou aceitar dados, ligar ou desligar, ou executar alguma tarefa.

Muitos dispositivos também possuem um buffer de dados, como a RAM de vídeo, que o sistema operacional pode usar para leitura e escrita.



PORTAS DE E/S

- Cada registrador de controle está ligado a uma porta de E/S (8 a 64 bits).
- O conjunto de todas as portas formam o espaço de portas E/S.
- Apenas o S.O pode usar instruções especiais de E/S para acessar essas portas.
- Computadores maiores como IBM 360 utiliza esse sistema

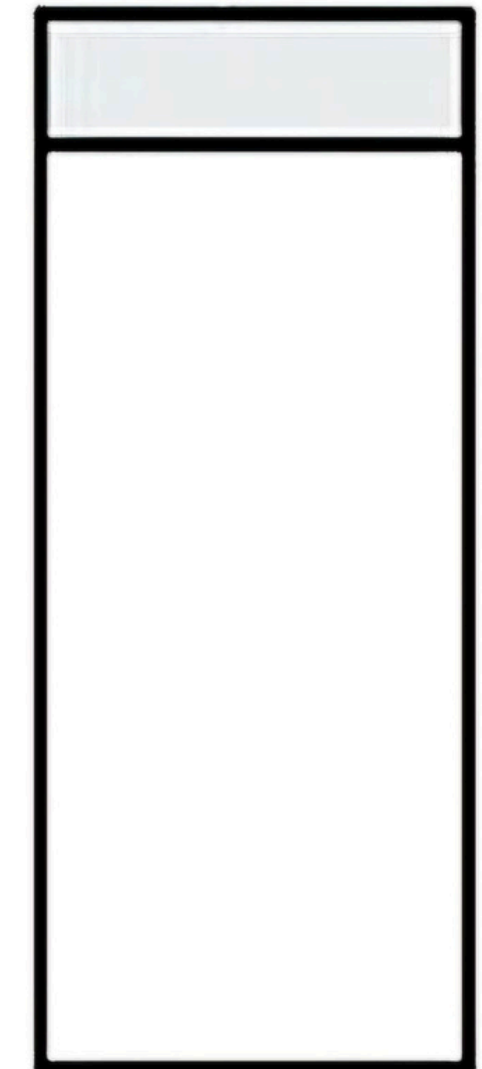


Espaços de memória e E/S separados

MAPEAMENTO DOS REGISTRADORES NA MEMÓRIA

- Trata os registradores de controle de dispositivos como posições de memória RAM
- Cada registrador recebe um endereço único, que não é usado pela RAM propriamente dita
- Simplifica a interação com dispositivos, utilizando as mesmas instruções de acesso à memória.
- esses endereços de memória reservados para E/S mapeada estão localizados nas posições mais altas do espaço de endereçamento da memória.

Um espaço de
endereçamento



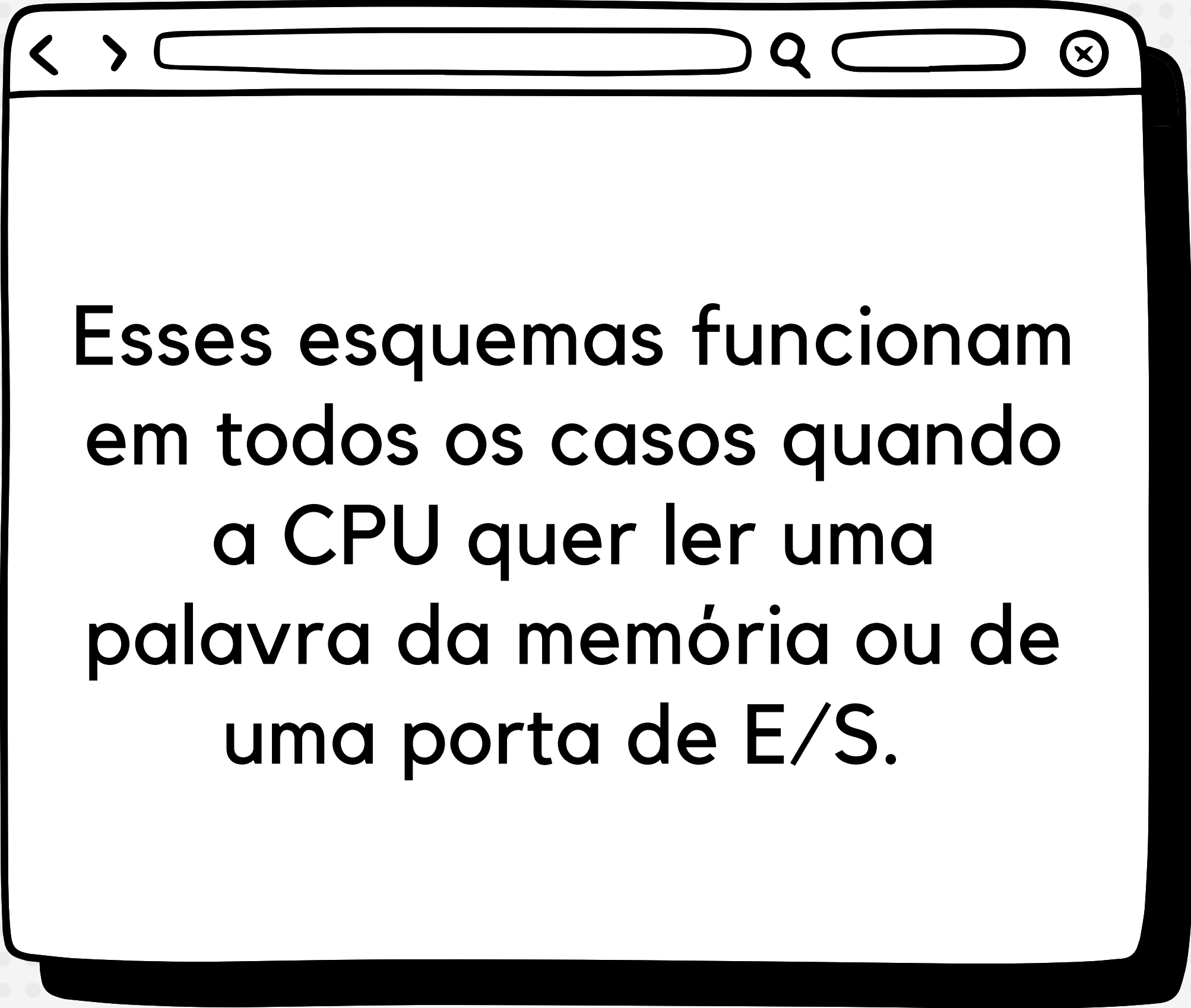
E/S mapeada na memória

ESQUEMA HÍBRIDO

- Combina a E/S mapeada na memória com as portas de E/S tradicionais.
- Buffers de dados dos dispositivos: mapeados na memória.
- Registradores de controle: acessados por portas de E/S separadas.
- Em PCs compatíveis com IBM: - Endereços de 0 a 64KB - Registradores de controle e de 64 KB a 1 MB - : buffers de dados.



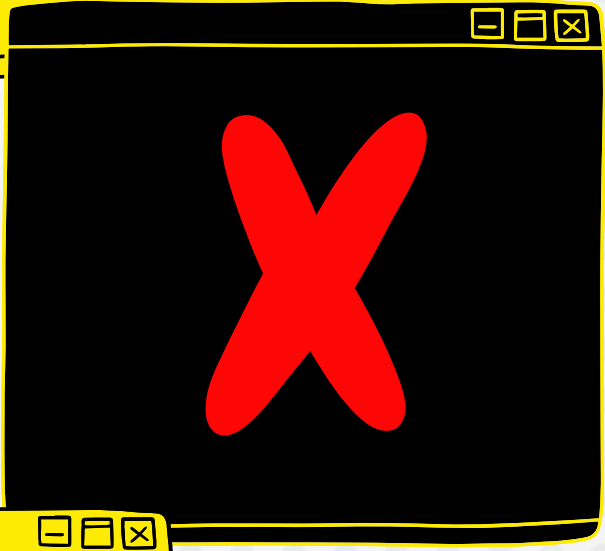
Esquema híbrido



Esse esquema funciona
em todos os casos quando
a CPU quer ler uma
palavra da memória ou de
uma porta de E/S.



PONTOS
FORTES



PONTOS
FRACOS





PONTOS FORTES



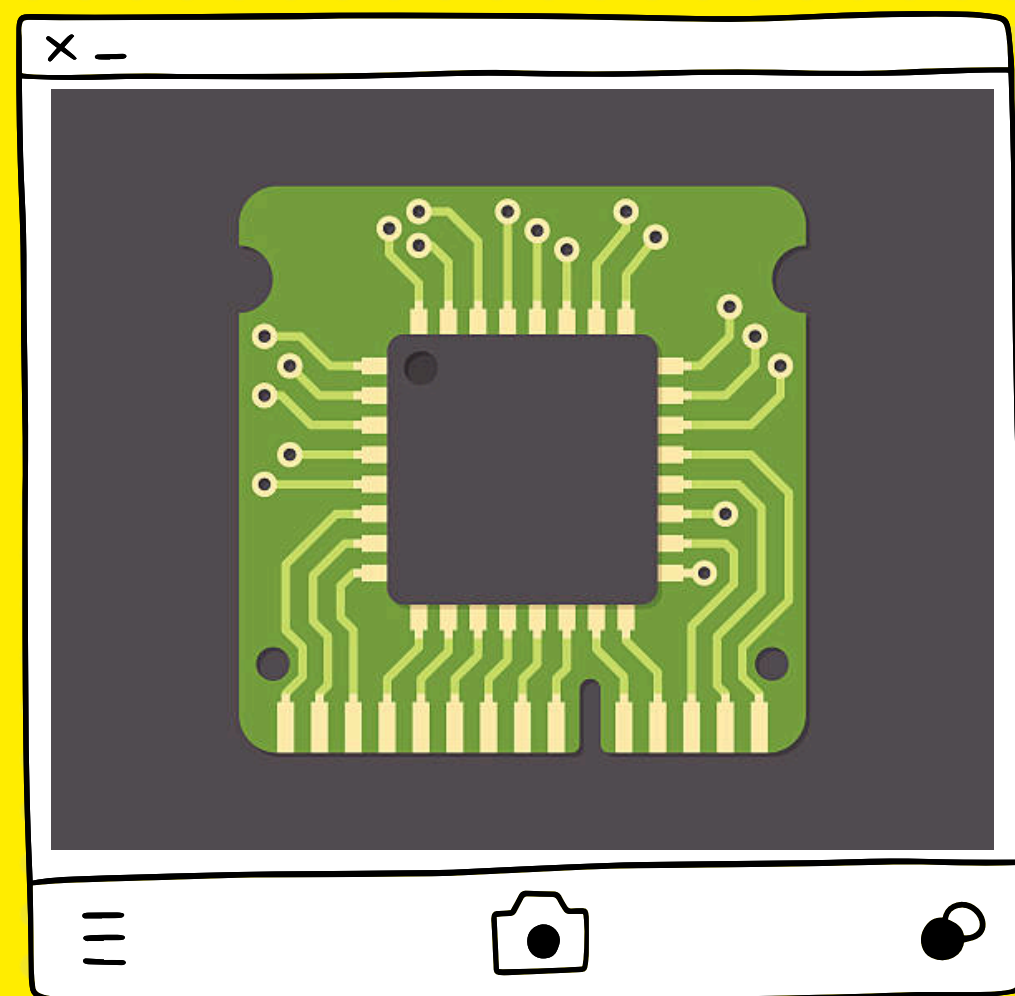
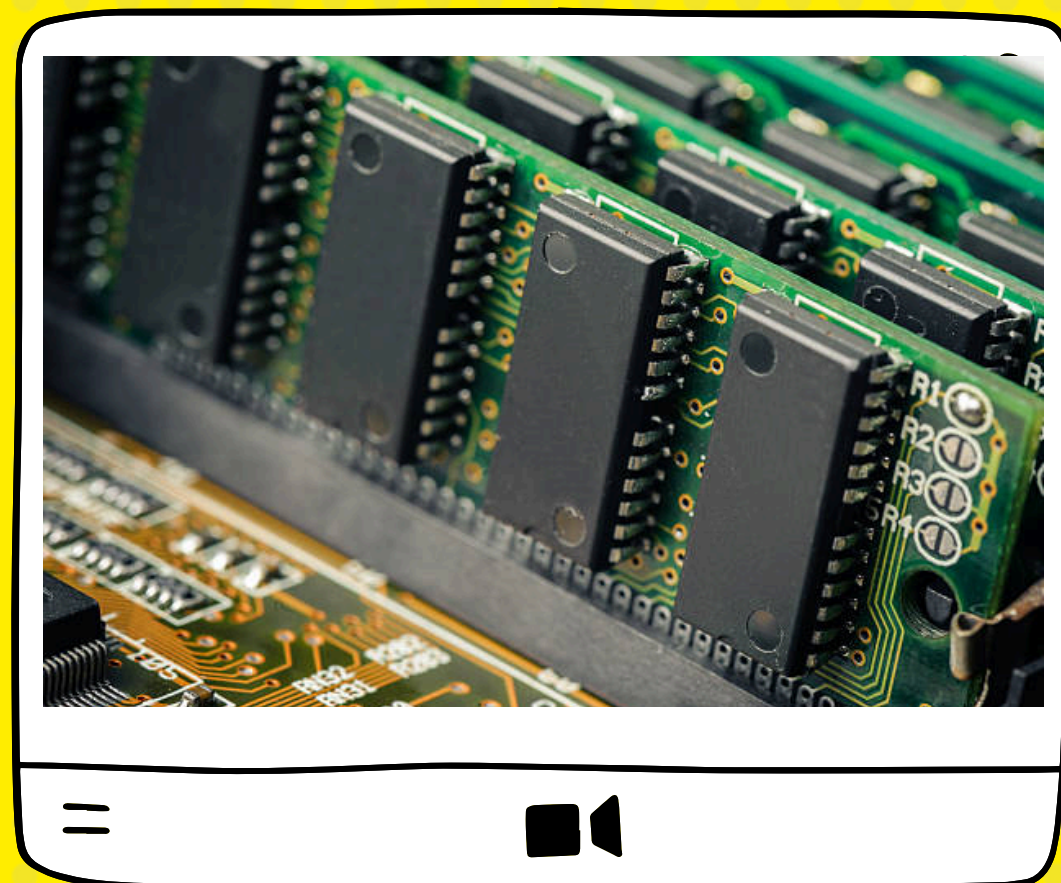
- Facilidade de programação: Permite escrever drivers de dispositivos em linguagens de alto nível (como C).
- Simplicidade de acesso: Registradores de controle são tratados como variáveis de memória.
- Eficiência: Elimina a necessidade de instruções especiais de E/S (IN/OUT).
- Velocidade: A CPU usa as mesmas instruções de acesso à memória, tornando a comunicação mais rápida.

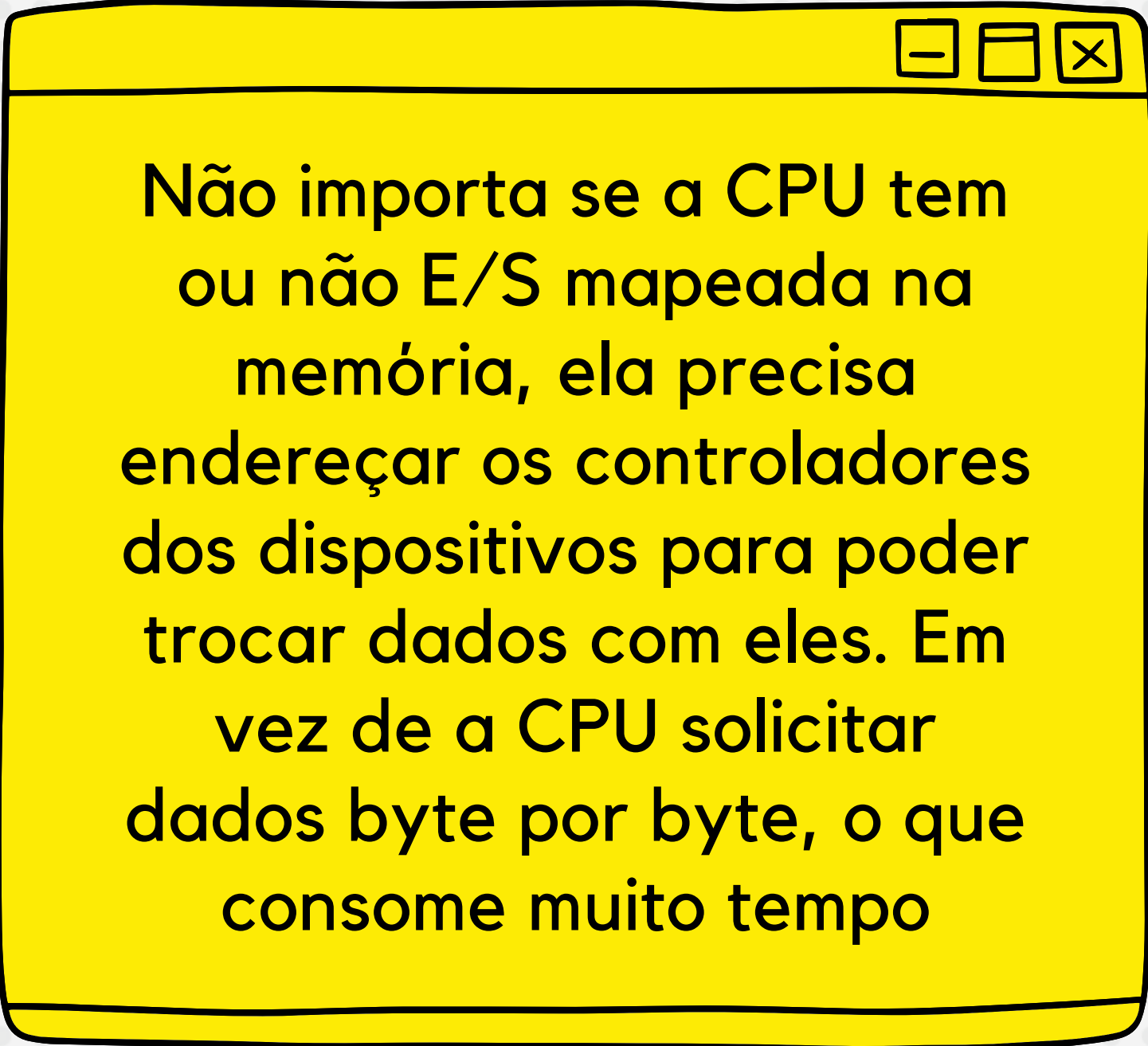


PONTOS FRACOS

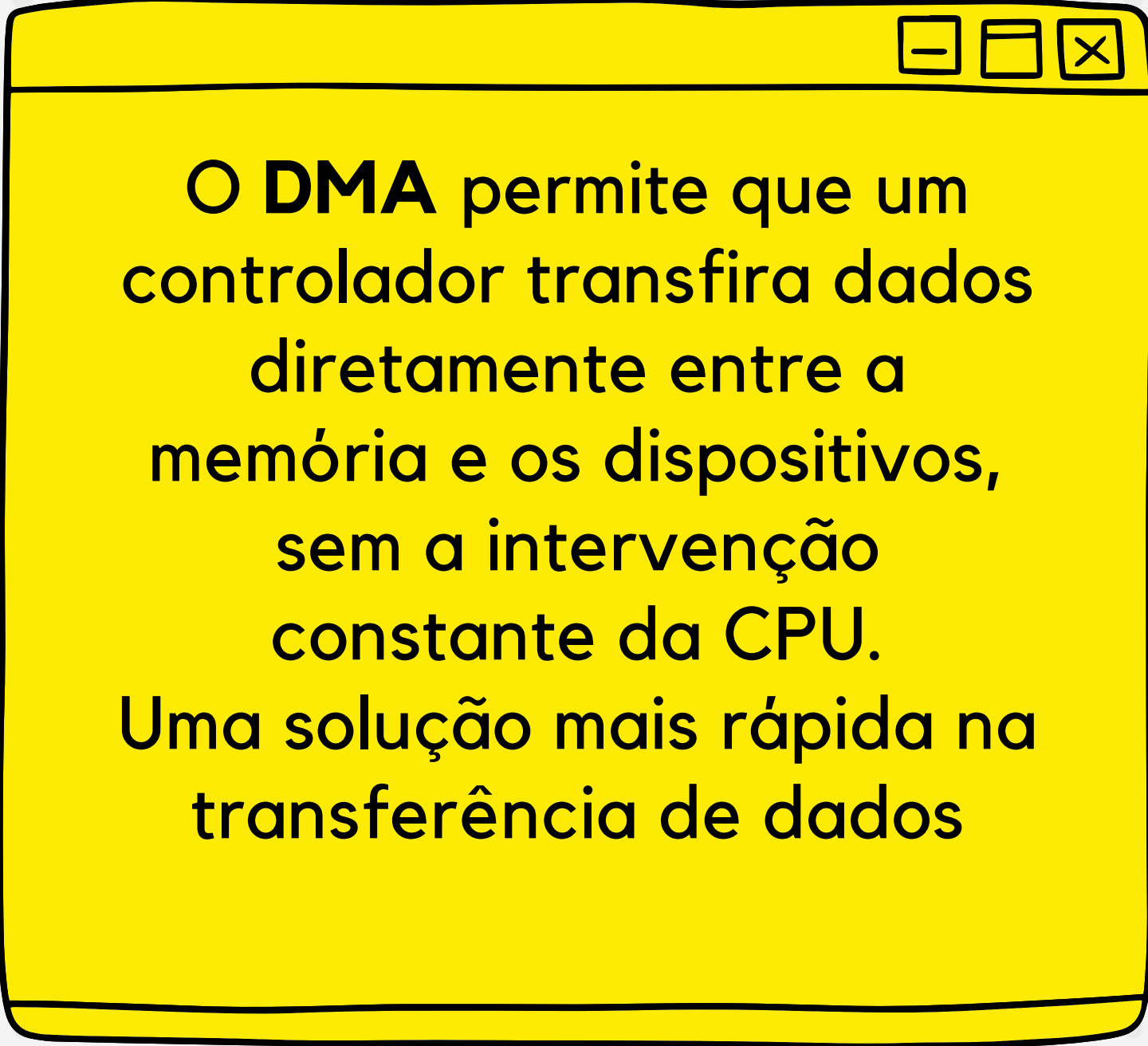


- Conflitos de endereçamento: É essencial evitar que os endereços de E/S e da memória RAM se sobreponham. Erros de configuração podem causar problemas no sistema.
- Complexidade do Hardware: A lógica de endereçamento de E/S é mais complexa devido ao maior tamanho dos endereços.





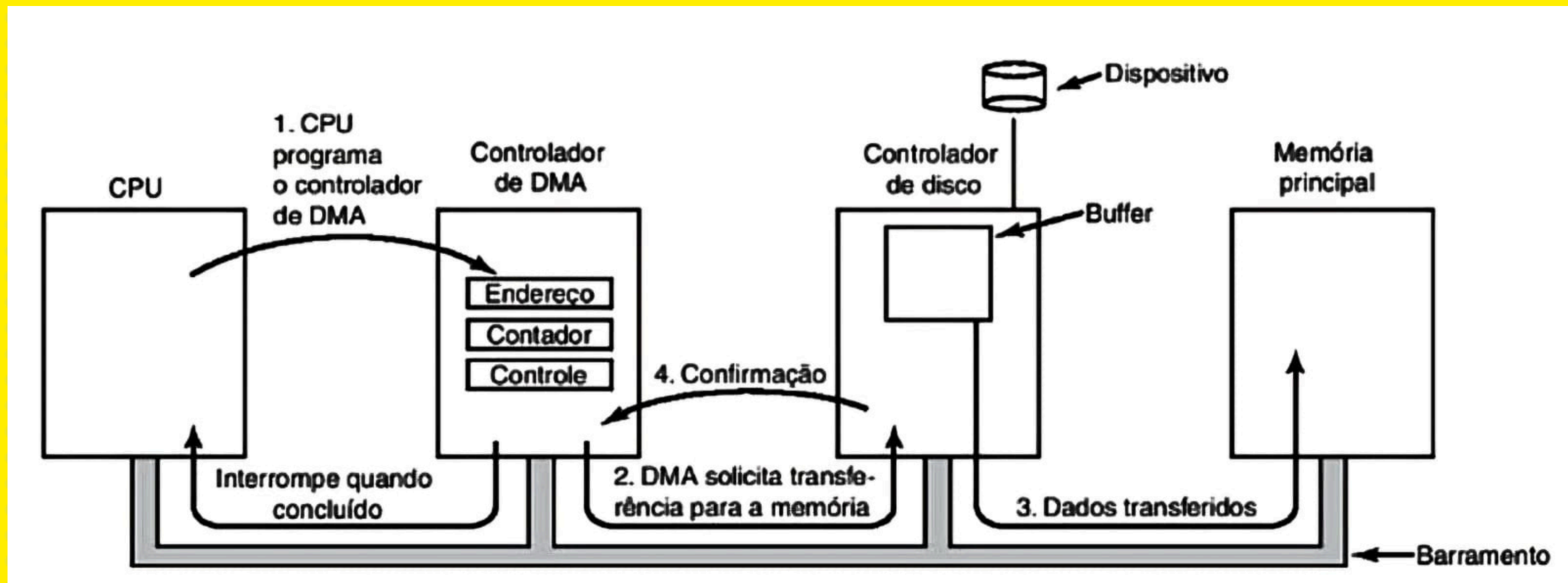
Não importa se a CPU tem ou não E/S mapeada na memória, ela precisa endereçar os controladores dos dispositivos para poder trocar dados com eles. Em vez de a CPU solicitar dados byte por byte, o que consome muito tempo



O **DMA** permite que um controlador transfira dados diretamente entre a memória e os dispositivos, sem a intervenção constante da CPU. Uma solução mais rápida na transferência de dados



O CONTROLADOR DE DMA ACESSA O BARRAMENTO DO SISTEMA INDEPENDENTEMENTE DA CPU, UTILIZANDO REGISTRADORES CONFIGURADOS POR ELA PARA DEFINIR ENDEREÇO DE MEMÓRIA, QUANTIDADE DE BYTES, DIREÇÃO DA TRANSFERÊNCIA E PORTA DE E/S. ISSO PERMITE QUE A CPU EXECUTE OUTRAS TAREFAS ENQUANTO O DMA GERENCIA TRANSFERÊNCIAS EM SEGUNDO PLANO, OTIMIZANDO O DESEMPENHO. SUA LOCALIZAÇÃO FÍSICA NÃO AFETA SEU ACESSO DIRETO AO BARRAMENTO, CONFORME ILUSTRADO.





MODO BLOCO (BURST MODE)



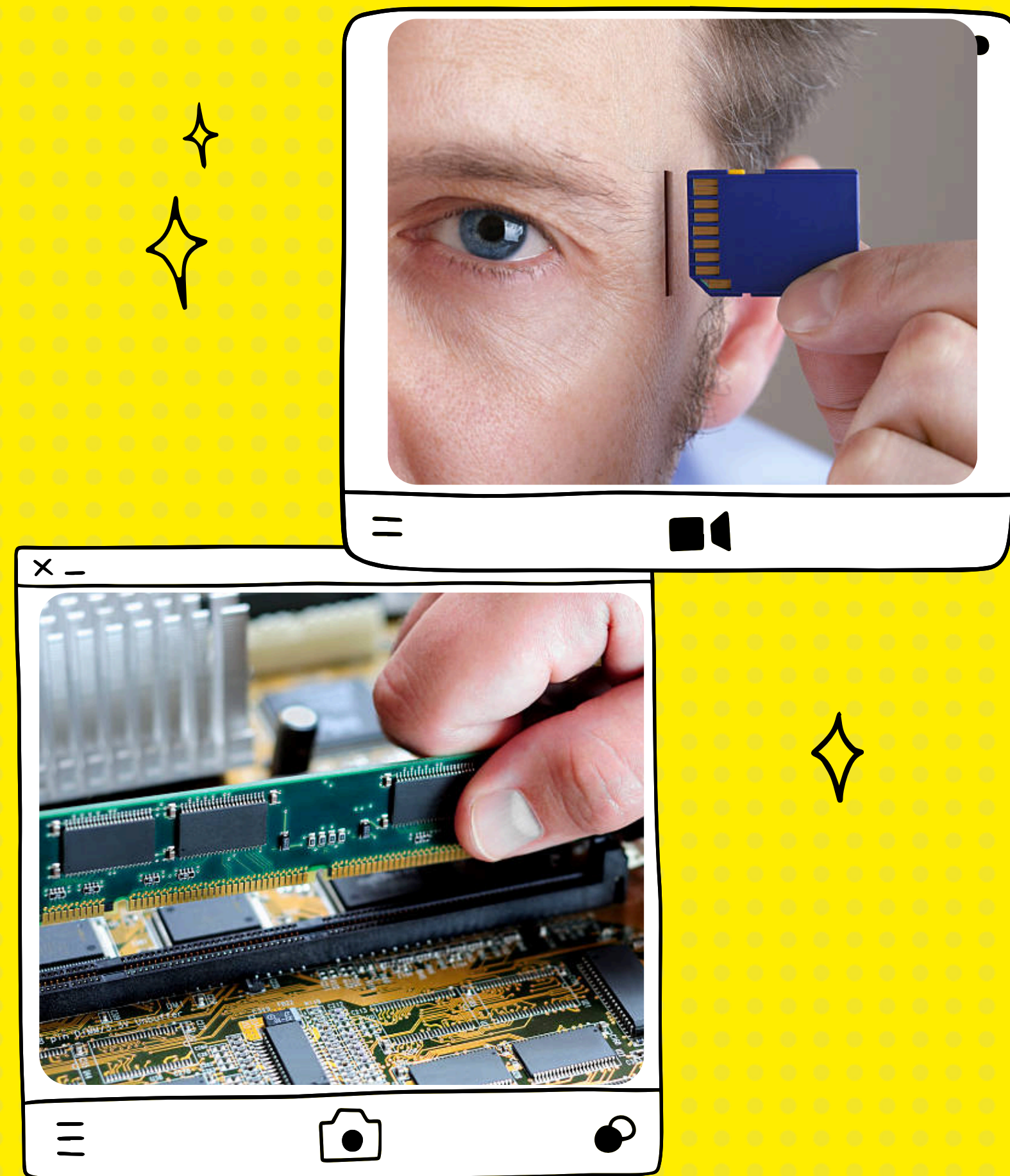
O controlador DMA instrui o dispositivo a assumir o controle do barramento, realizar uma sequência de transferências e, em seguida, liberar o barramento. Esse modo é mais eficiente porque o tempo gasto para obter o controle do barramento é aproveitado para transferir várias palavras de uma vez.



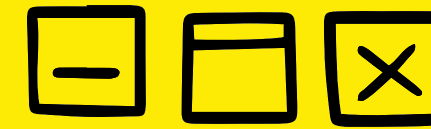
MODO PALAVRA POR VEZ



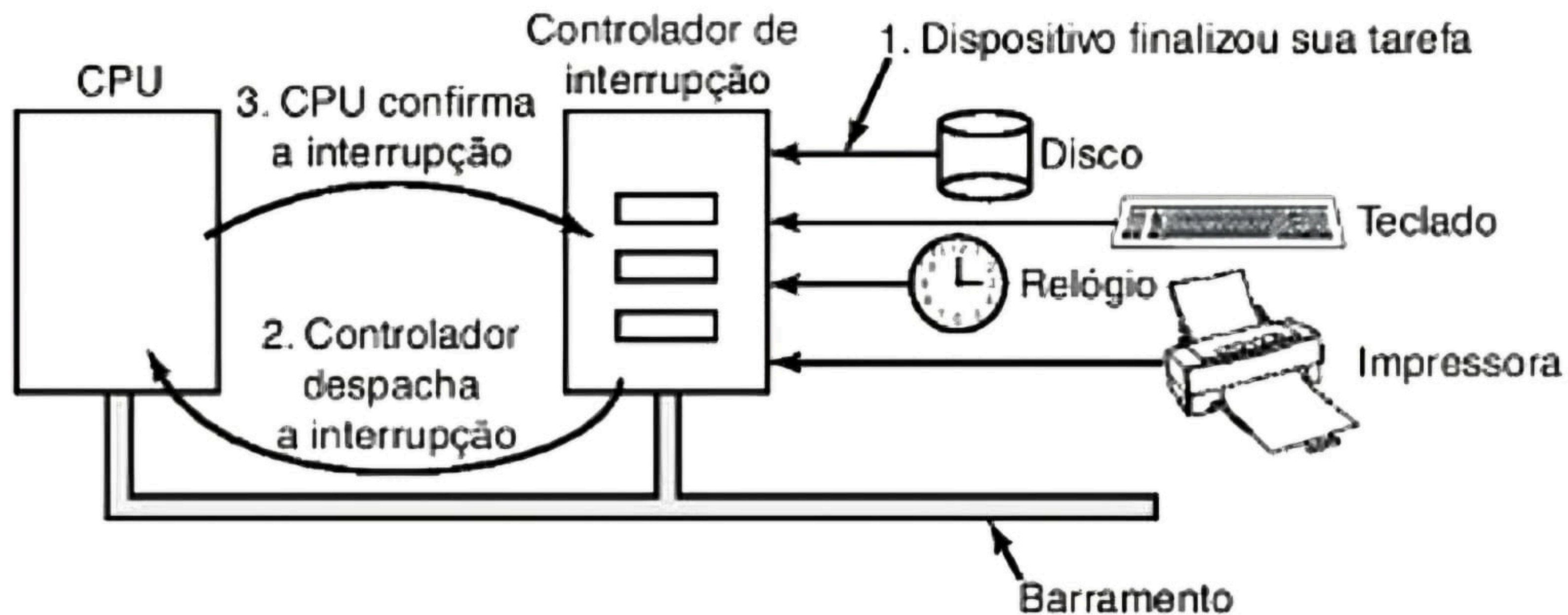
O controlador DMA solicita e obtém a transferência de uma palavra de cada vez. Se a CPU precisar usar o barramento, ela precisa esperar. Isso é chamado de "roubo de ciclo" porque o controlador DMA "rouba" pequenos períodos de uso do barramento da CPU, causando um pequeno atraso.



Interrupções de E/S



1. **Dispositivo completa tarefa → Gera interrupção via barramento.**
2. **Controlador de interrupções verifica prioridade:**
 - **Se livre, processa imediatamente.**
 - **Se ocupado, aguarda disponibilidade da CPU.**
3. **CPU é interrompida e recebe o vetor de interrupção (código do serviço).**
4. **CPU executa rotina de tratamento e envia confirmação ao controlador.**
5. **Sistema liberado para novas interrupções.**





Interrupção precisa : Uma interrupção que deixa a máquina em um estado bem definido é chamada de interrupção precisa (Walker e Cragon, 1995). Tal interrupção tem quatro propriedades:

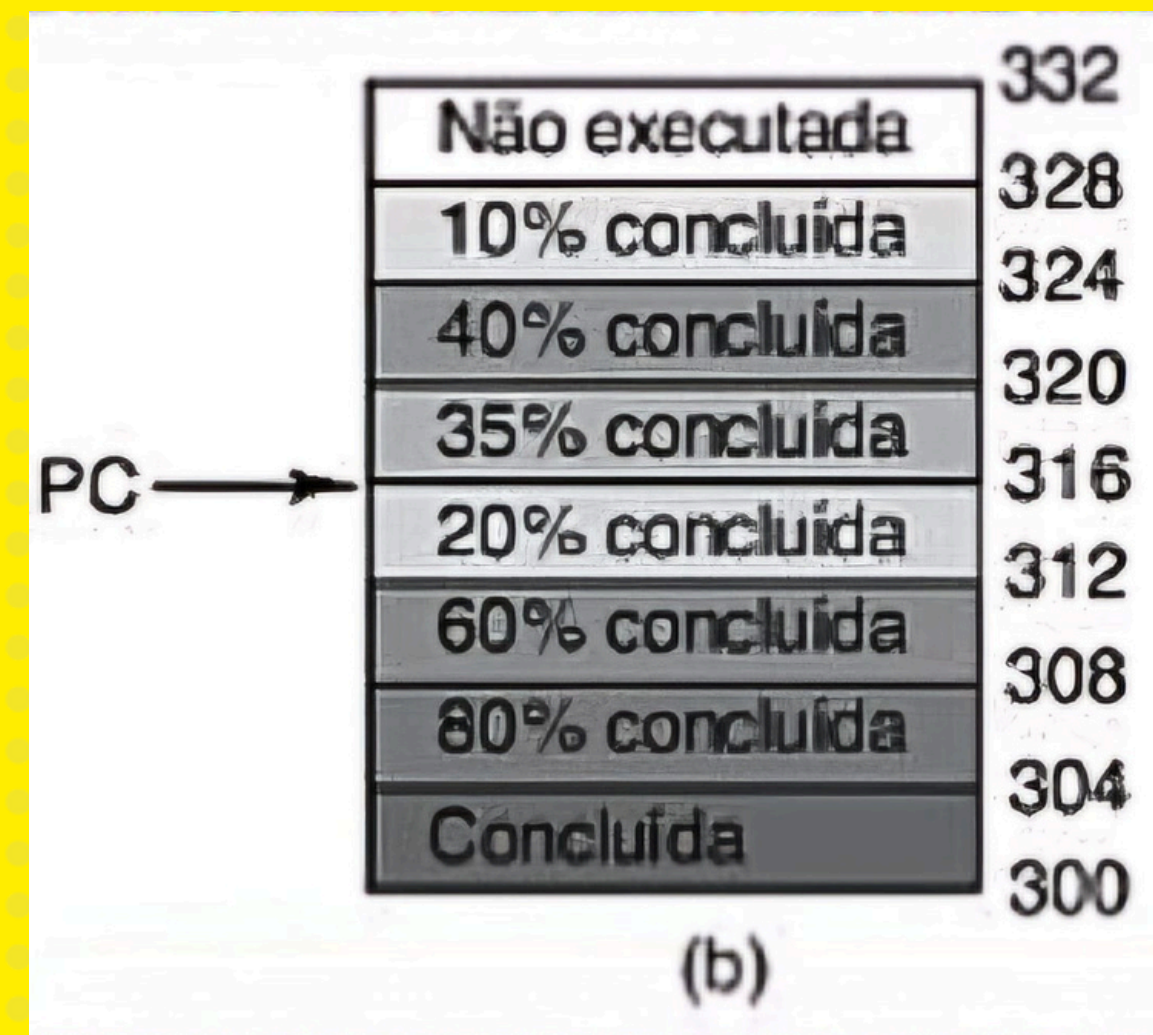
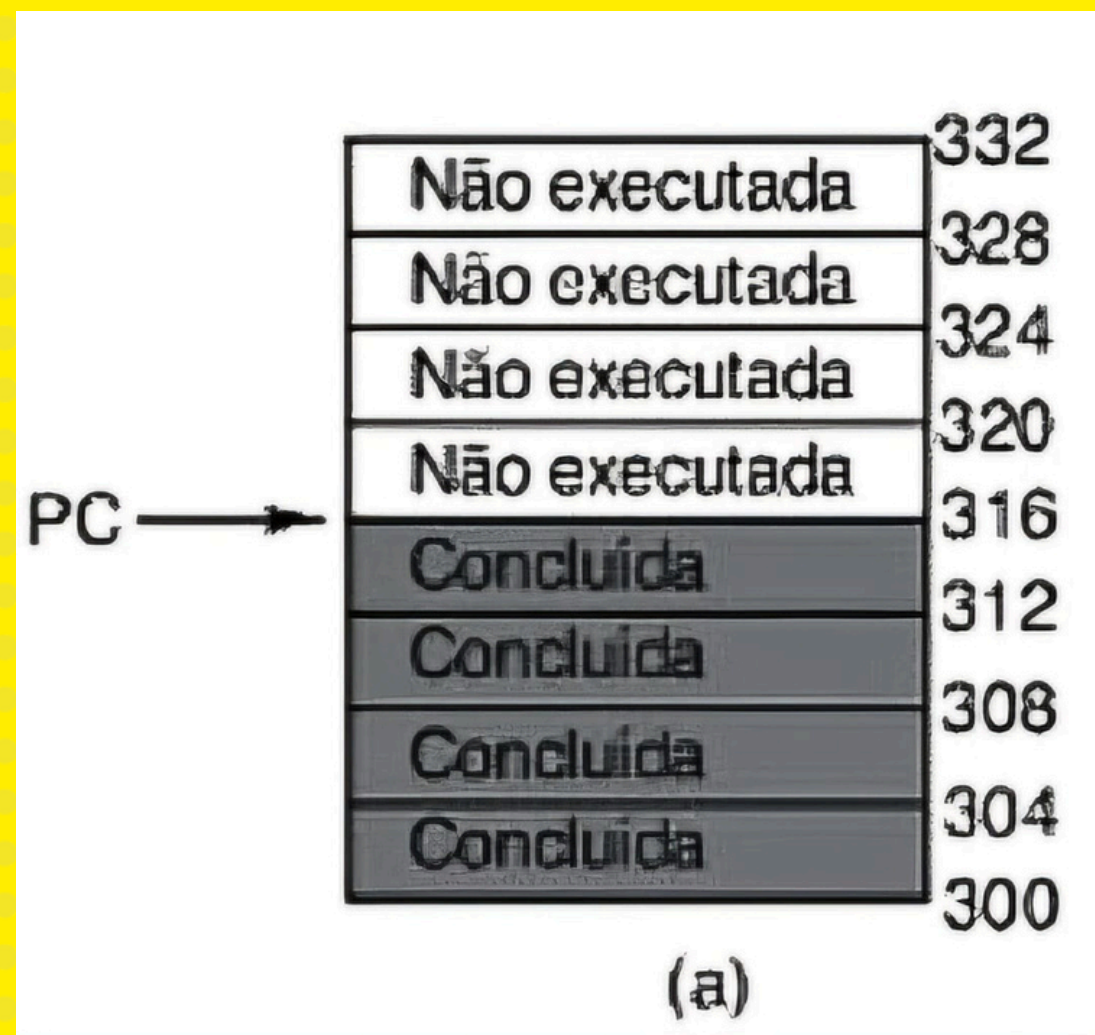
- O PC (Contador de Programa) é salvo em um local conhecido.
- Todas as instruções anteriores àquela apontada pelo PC foram concluídas.
- Nenhuma instrução além daquela apontada pelo PC foi concluída.
- O estado de execução da instrução apontada pelo PC é conhecido.

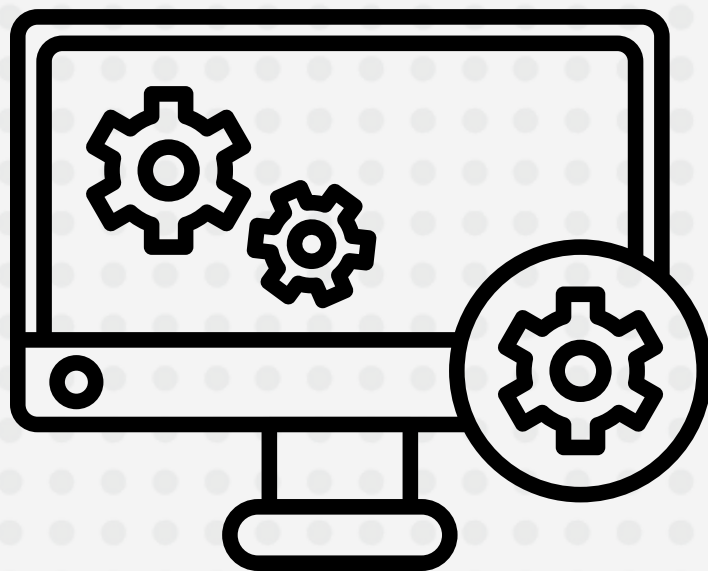


Interrupções Imprecisas

- Dificultam o trabalho do criador do sistema operacional, que precisa investigar o que ocorreu e o que ainda precisa ser feito.
- Sistemas com interrupções imprecisas geralmente armazenam uma grande quantidade de dados internos na pilha para auxiliar o sistema operacional na investigação.
- Interrupções imprecisas permitem maior uso do espaço da CPU para cache, etc., mas aumentam a complexidade do sistema operacional.

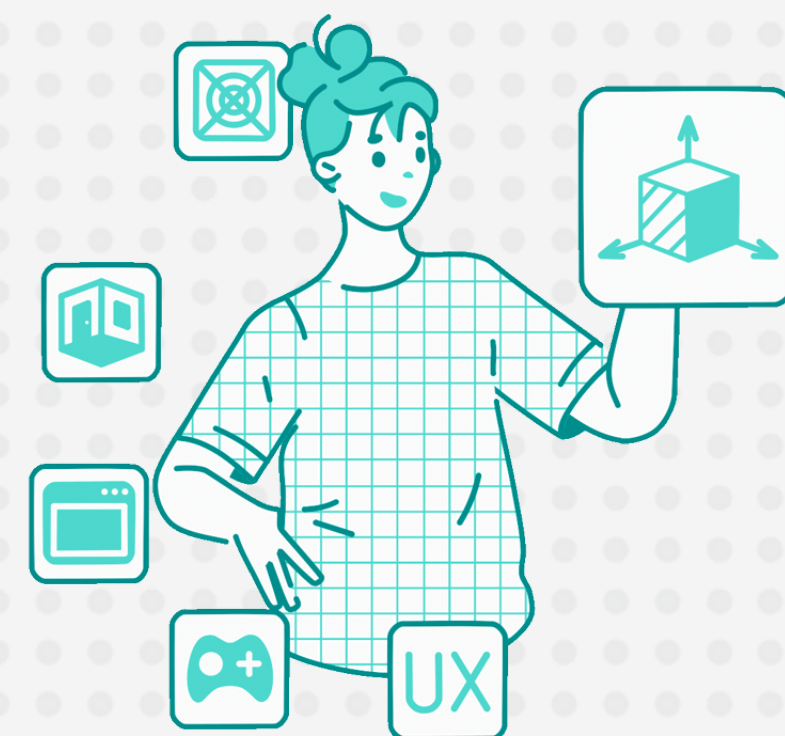
A SEGUIR É POSSÍVEL COMPARAR OS DOIS TIPOS DE INTERRUPÇÕES

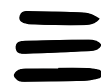




PRINCÍPIOS DO SOFTWARE DE E/S

VINÍCIUS CARNEIRO





INDEPENDENCIA DE DISPOSITIVO

Acesso a arquivos sem precisar
saber do dispositivo de origem



NOMEAÇÃO UNIFORME

Nomes simples e padronizados
facilitando o acesso e entendimento



TRATAMENTO DE ERROS

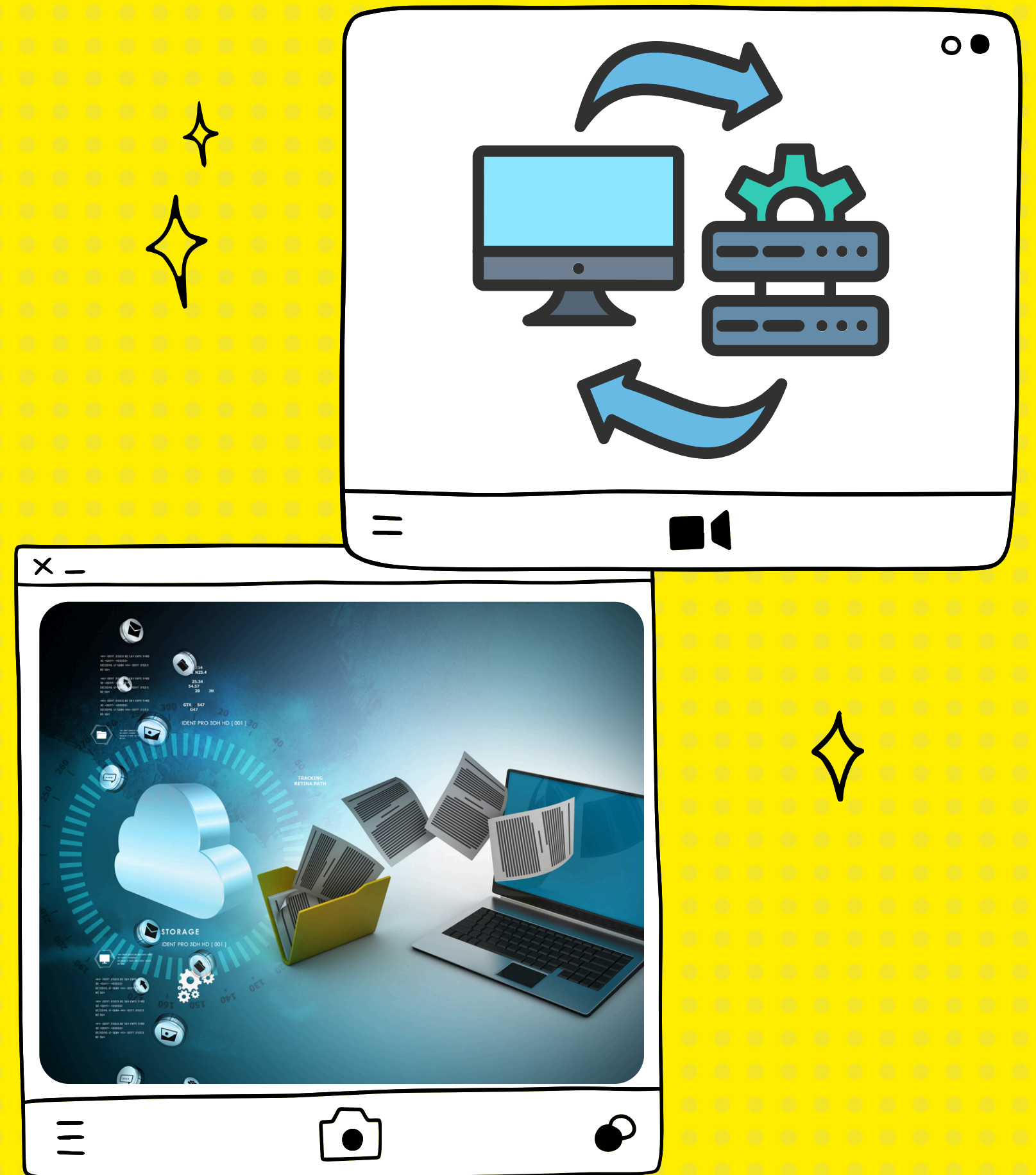
Corrige automaticamente os erros



OBJETIVOS



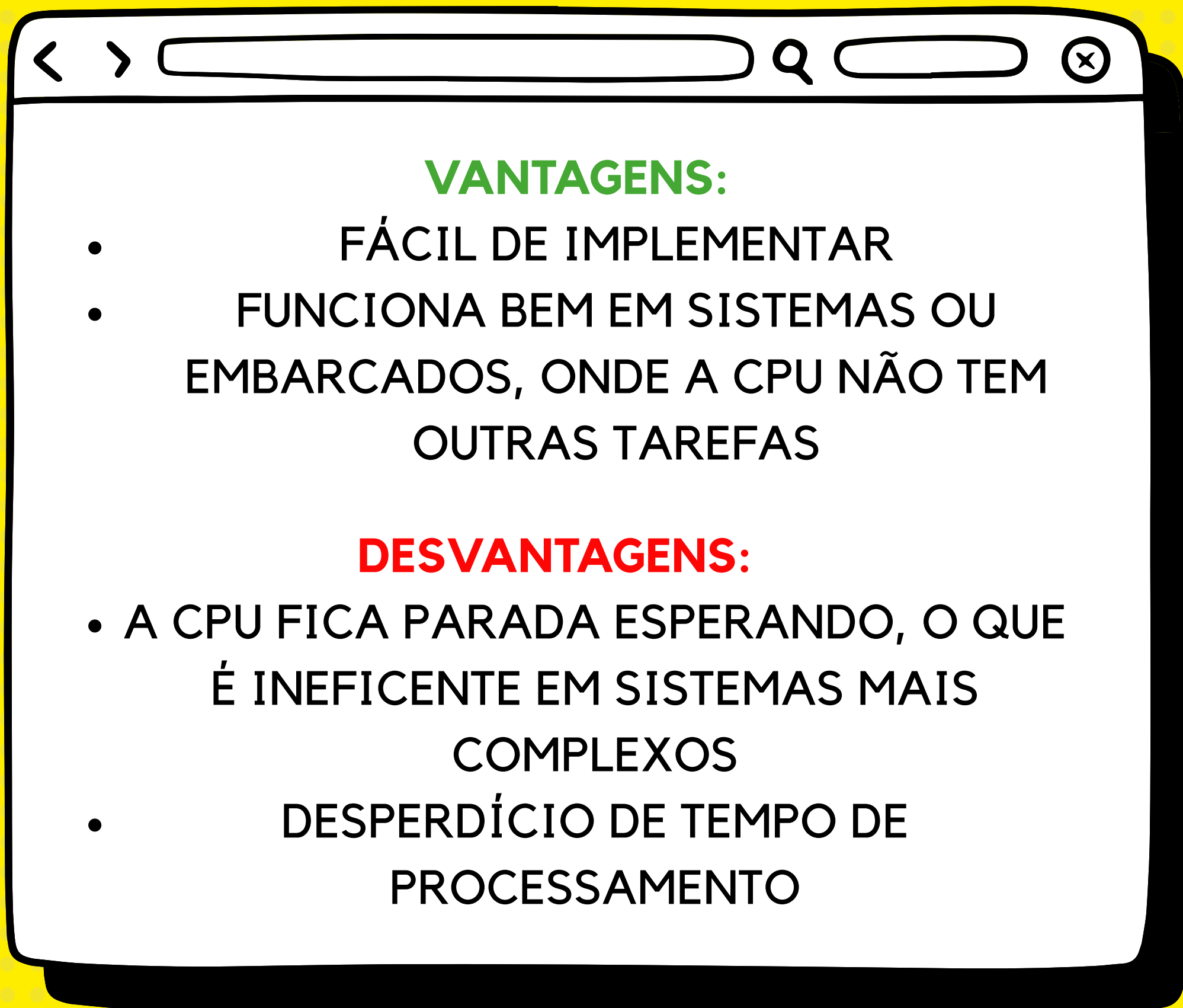
Principais formas de transferência de dados



E/S Programada

Ela é simples porém ineficiente, a CPU faz tudo sozinha, ela espera o dispositivo terminar a tarefa para realizar outra. A CPU envia um comando, dps ela checa.





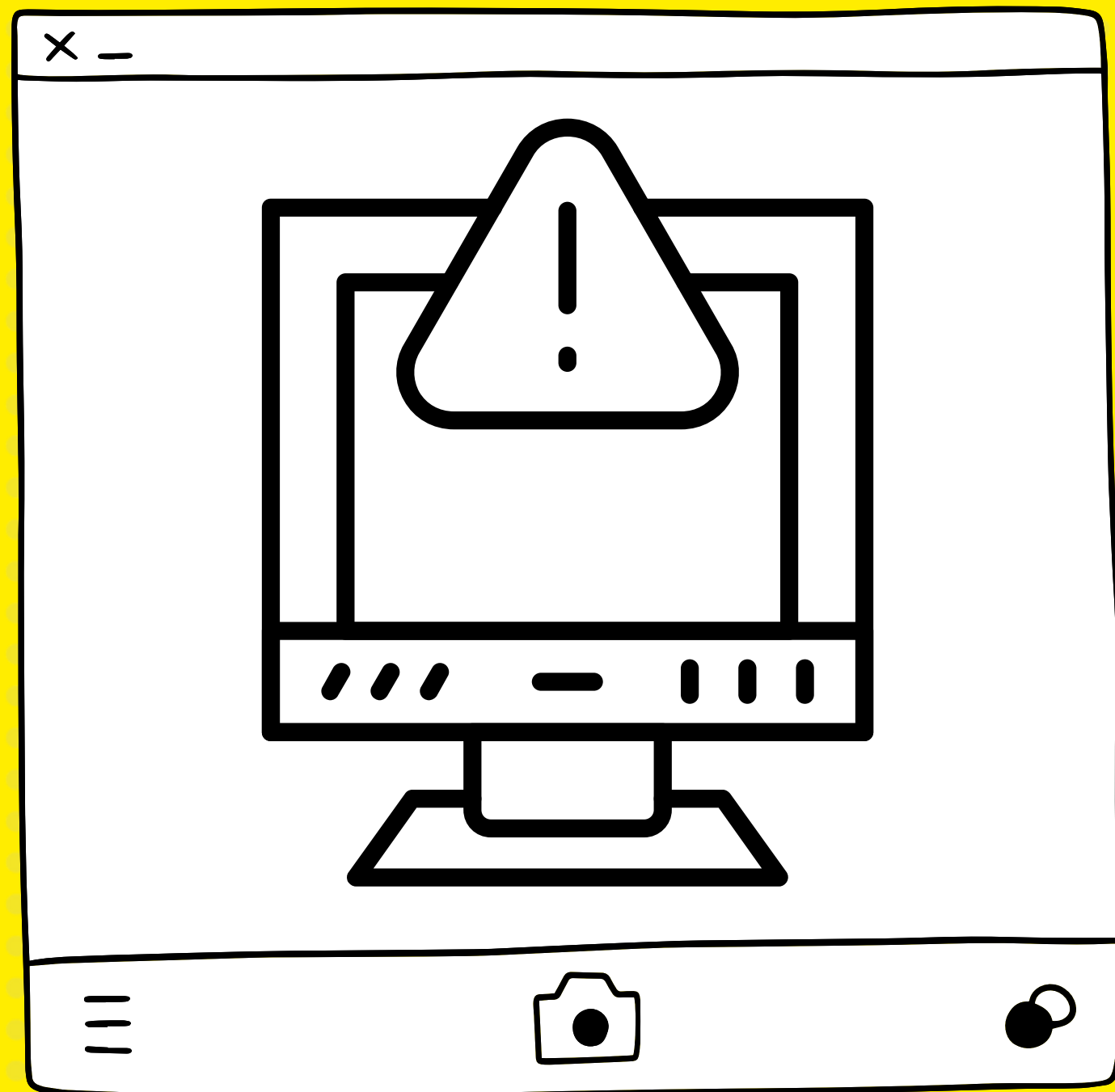
VANTAGENS:

- FÁCIL DE IMPLEMENTAR
- FUNCIONA BEM EM SISTEMAS OU EMBARCADOS, ONDE A CPU NÃO TEM OUTRAS TAREFAS

DESVANTAGENS:

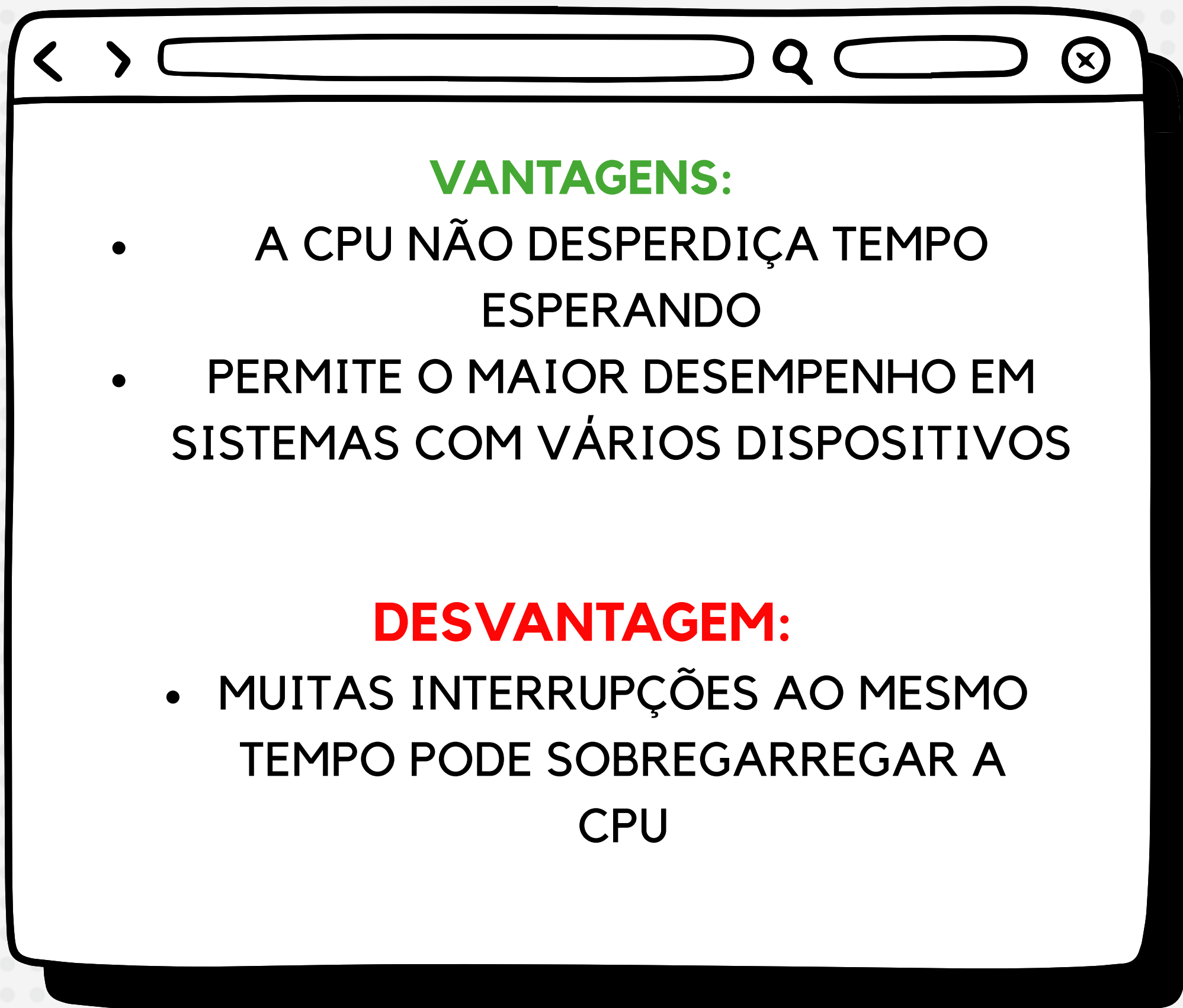
- A CPU FICA PARADA ESPERANDO, O QUE É INEFICIENTE EM SISTEMAS MAIS COMPLEXOS
- DESPERDÍCIO DE TEMPO DE PROCESSAMENTO





E/S COM INTERRUPÇÃO

Permite que a CPU possa realizar mais de uma tarefa por vez. Quando termina a tarefa manda um sinal para a CPU, e esse sinal é chamado de **INTERRUPÇÃO**



VANTAGENS:

- A CPU NÃO DESPERDIÇA TEMPO ESPERANDO
- PERMITE O MAIOR DESEMPENHO EM SISTEMAS COM VÁRIOS DISPOSITIVOS

DESVANTAGEM:

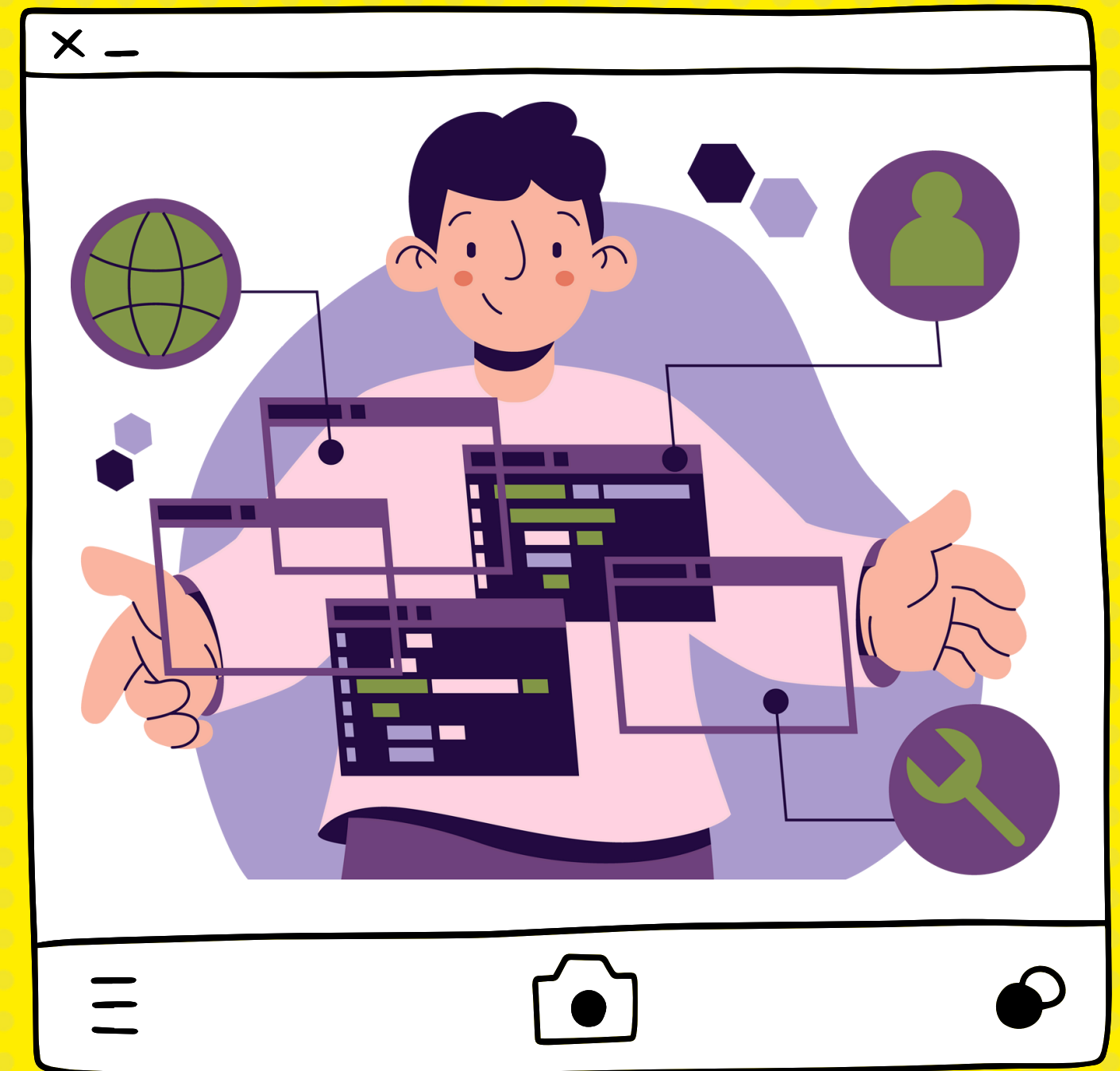
- MUITAS INTERRUPÇÕES AO MESMO TEMPO PODE SOBREGARREGAR A CPU

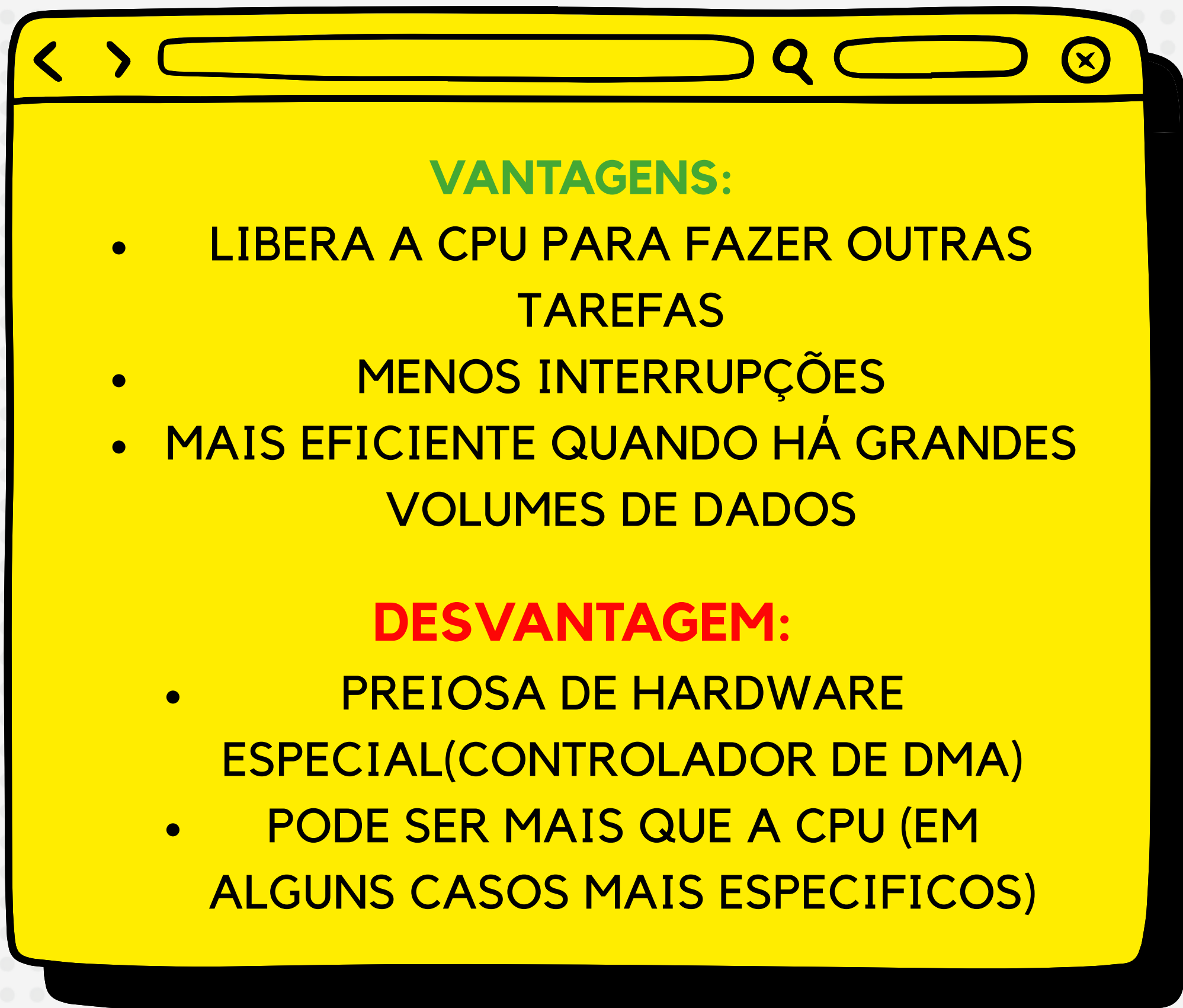


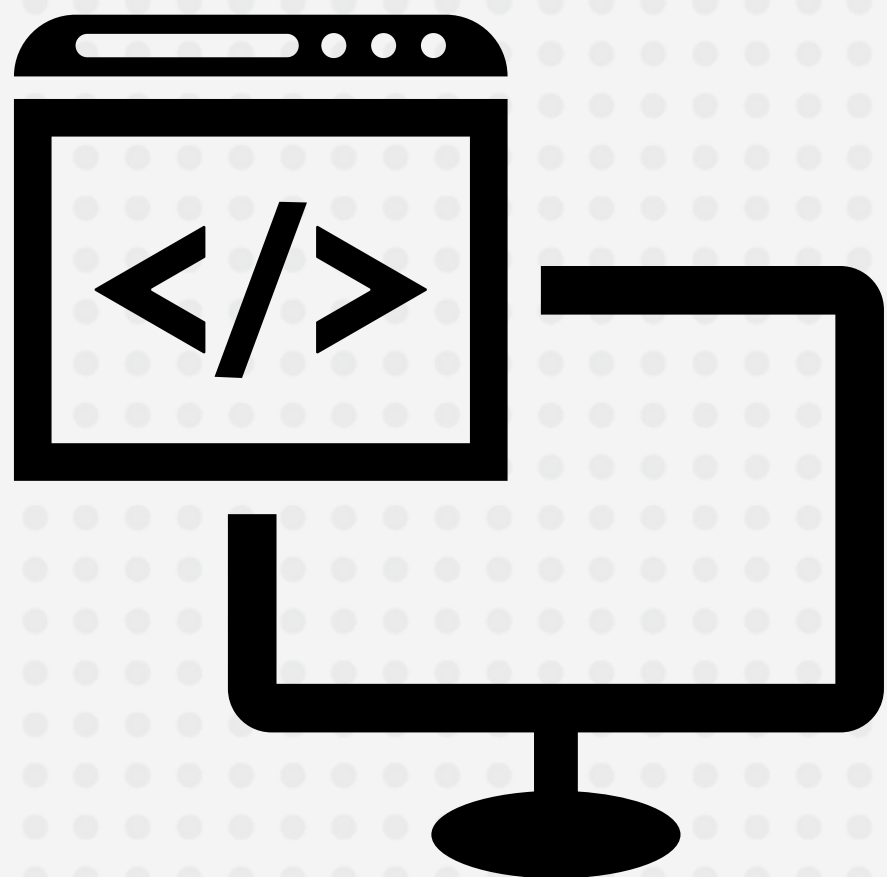
E/S com DMA

DMA (Acesso direto à memória), chip especial (controlador de DMA) faz a transferência direto para a memória e o dispositivo, sem a necessidade da CPU o tempo todo.

A CPU dá o comando, o controlador faz todo o trabalho sozinho da ai em diante, tudo tudo estiver pronto, o DMA avisa com a interrupção que terminou

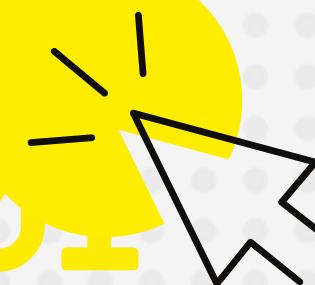






CAMADAS DE SOFTWARE

IGOR C.
LIMA

A graphic design featuring a large yellow smiley face with a thick black outline. The word "SOFTWARE" is written in large, bold, black capital letters across the top of the smiley face. Below it, the name "IGOR C. LIMA" is written in smaller, bold, black capital letters. The background is white and filled with a pattern of yellow binary code (0s and 1s) of varying sizes, some of which are partially obscured by the smiley face.

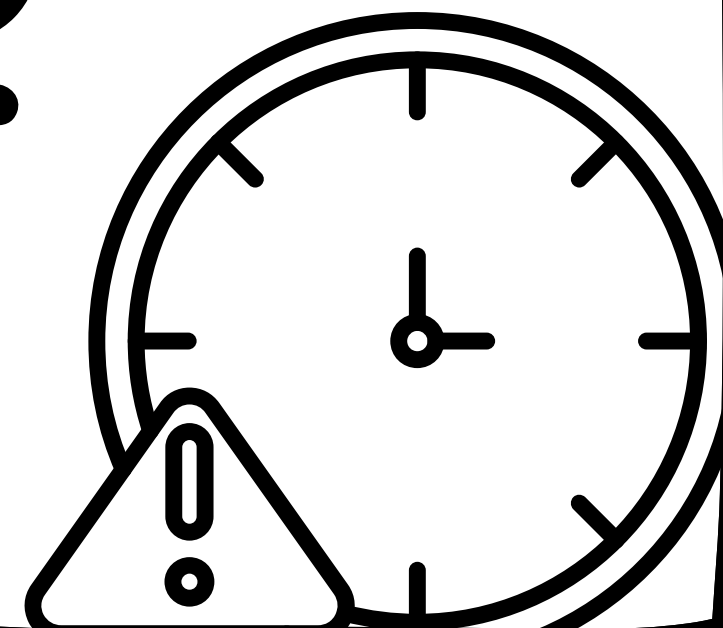
SOFTWARE

IGOR C.
LIMA



TRATADORES DE INTERRUPÇÃO

.....



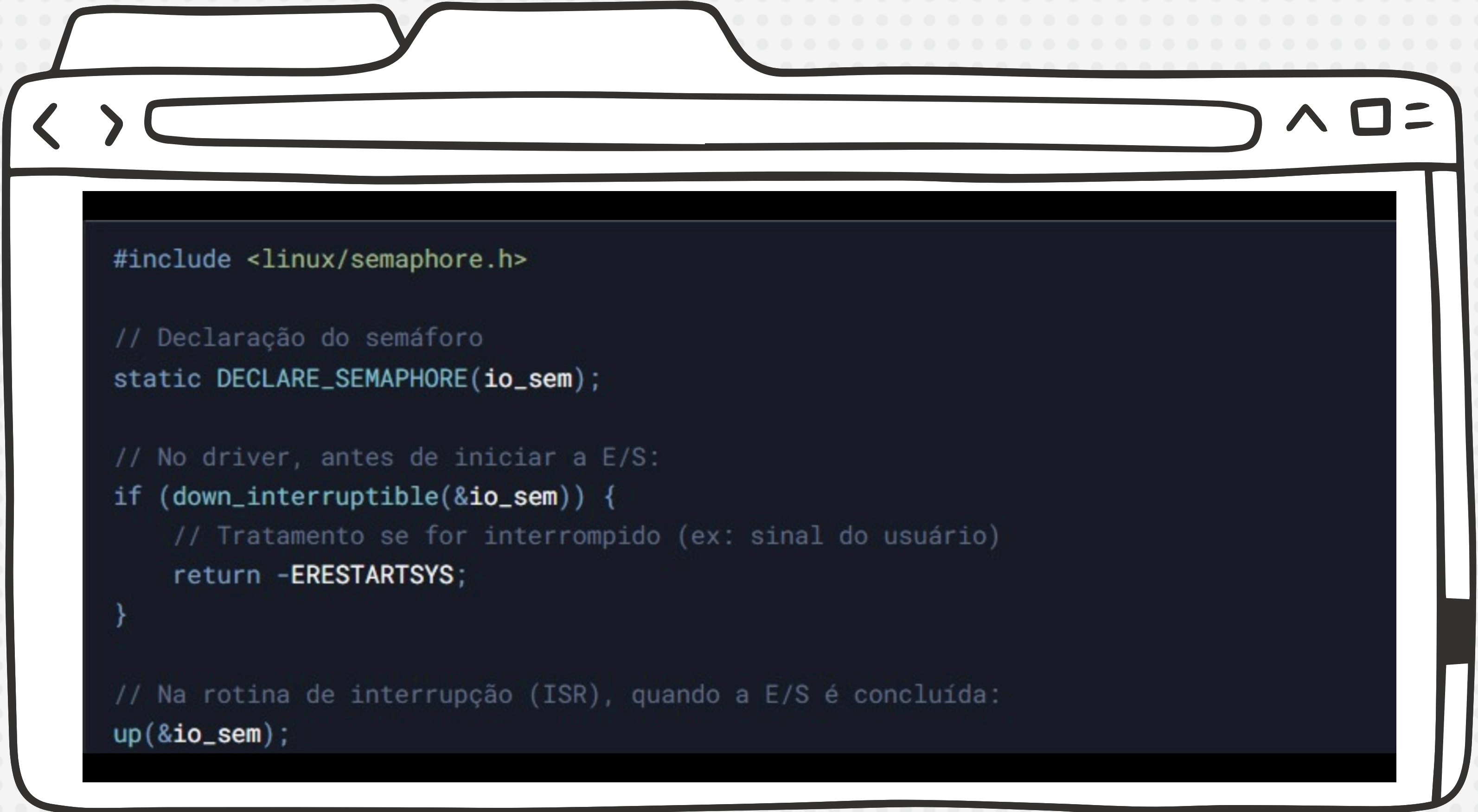
INTERRUPÇÕES

- ESCONDER A COMPLEXIDADE
-
- PROCESSO DA INTERRUPÇÃO
- SEMÁFOROS
- VARIÁVEIS DE CONDIÇÃO
- TROCA DE MENSAGENS
- EX: DRIVER DE DISCO

TRATAMENTO DE INTERRUPÇÕES

-
- SALVA ESTADO ATUAL
- PREPARAR AMBIENTE
- SINALIZAR
- REGISTRADORES DO DISPOSITIVO
- PRÓXIMO PROCESSO
- RESTAURAR
- RETOMAR

- DESVANTAGENS:
- OVERHEAD
- CONCORRÊNCIA
- HARDWARES ESPECÍFICOS
- IMPORTÂNCIA

A hand-drawn illustration of a web browser window. The window has a title bar with a close button (X), a maximize button, and a zoom-in button. The address bar is empty. The main content area displays C code for semaphore usage. The code is as follows:

```
#include <linux/semaphore.h>

// Declaração do semáforo
static DECLARE_SEMAPHORE(io_sem);

// No driver, antes de iniciar a E/S:
if (down_interruptible(&io_sem)) {
    // Tratamento se for interrompido (ex: sinal do usuário)
    return -ERESTARTSYS;
}

// Na rotina de interrupção (ISR), quando a E/S é concluída:
up(&io_sem);
```

```
// Declaração do semáforo
```

```
static DECLARE_SEMAPHORE(io_sem);
```

```
// No driver, antes de iniciar a E/S:
```

```
if (down_interruptible(&io_sem)) {
```

```
    // Tratamento se for interrompido (ex: sinal do usuário)
```

```
    return -ERESTARTSYS;
```

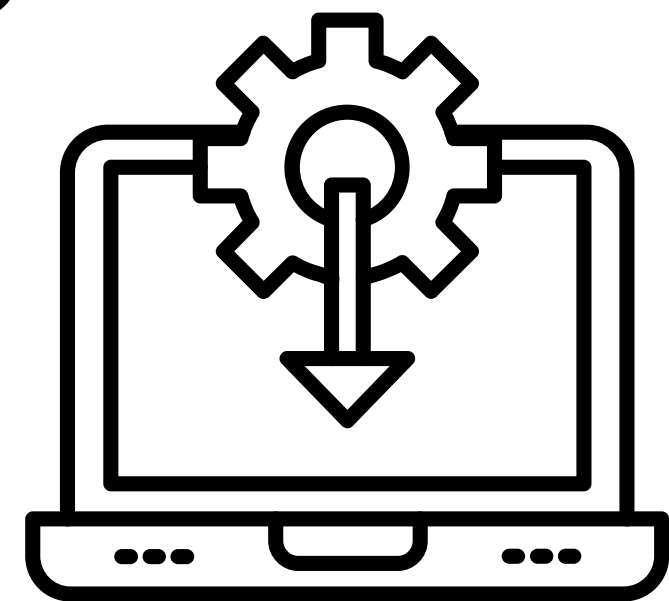
```
}
```

```
// Na rotina de interrupção (ISR), quando a E/S é concluída:
```

```
up(&io_sem);
```

DRIVERS DE DISPOSITIVOS

.....



PARA QUE SERVEM?

- Modo kernel
- Modo usuário

FUNCIONAM

DESAFIOS

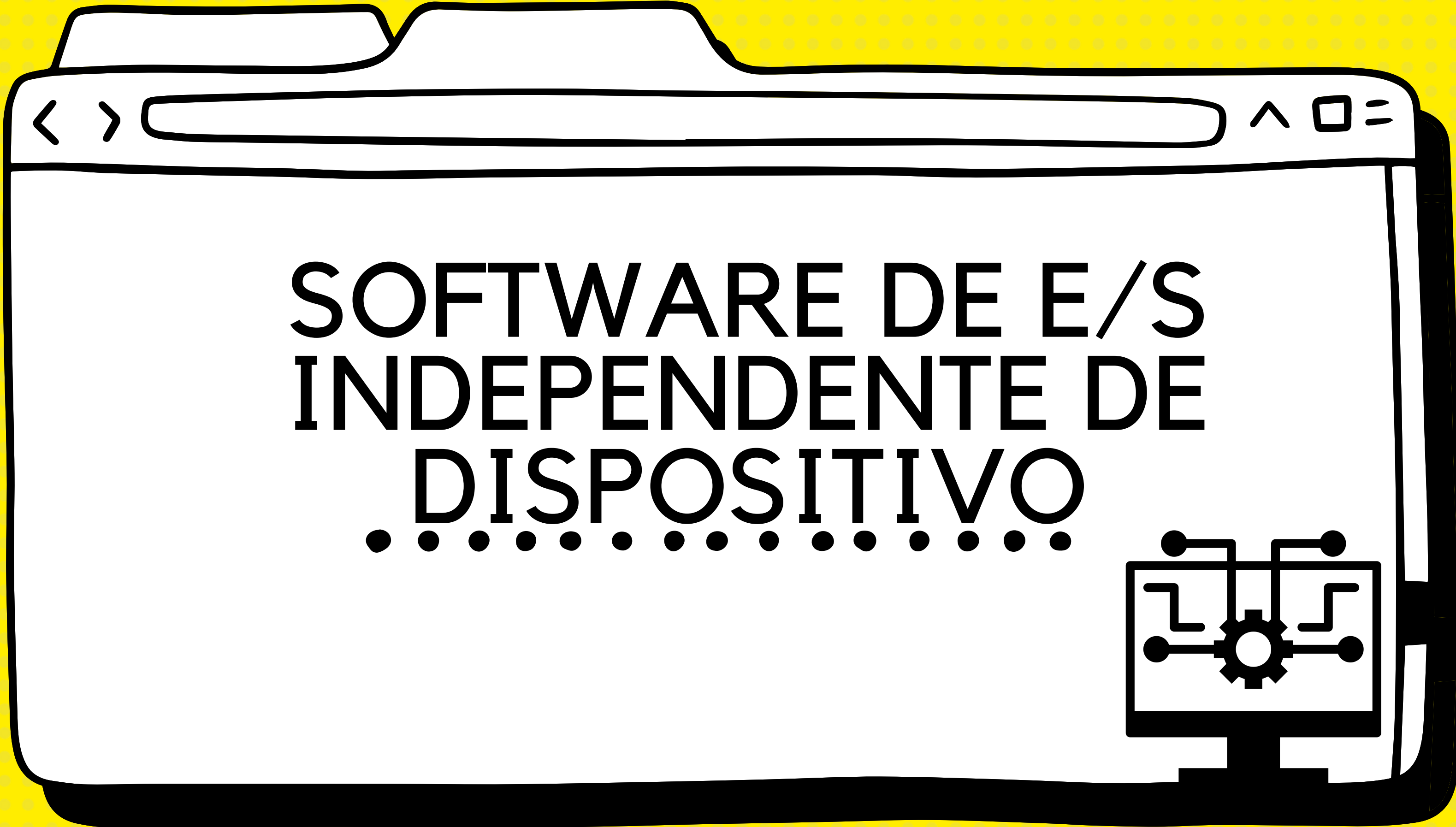
- Reentrância
- Hot plug
- Alocar recursos

COMO

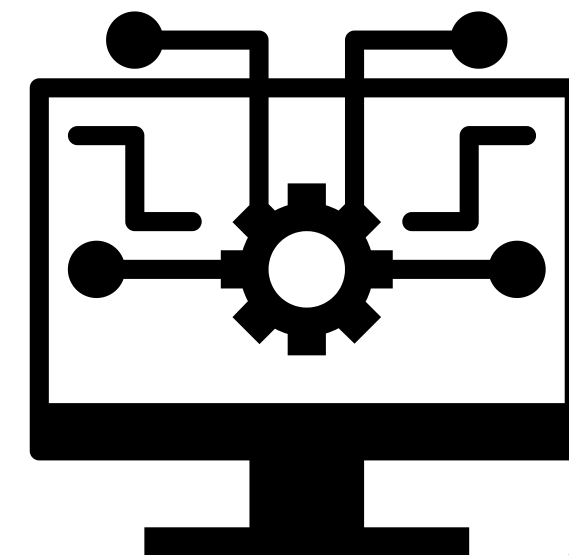
DOIS TIPOS DE DRIVER

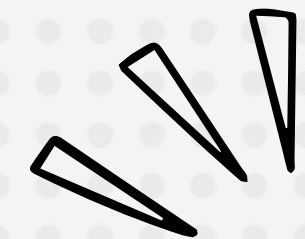
- Caracteres
- Blocos

OS DRIVERS?

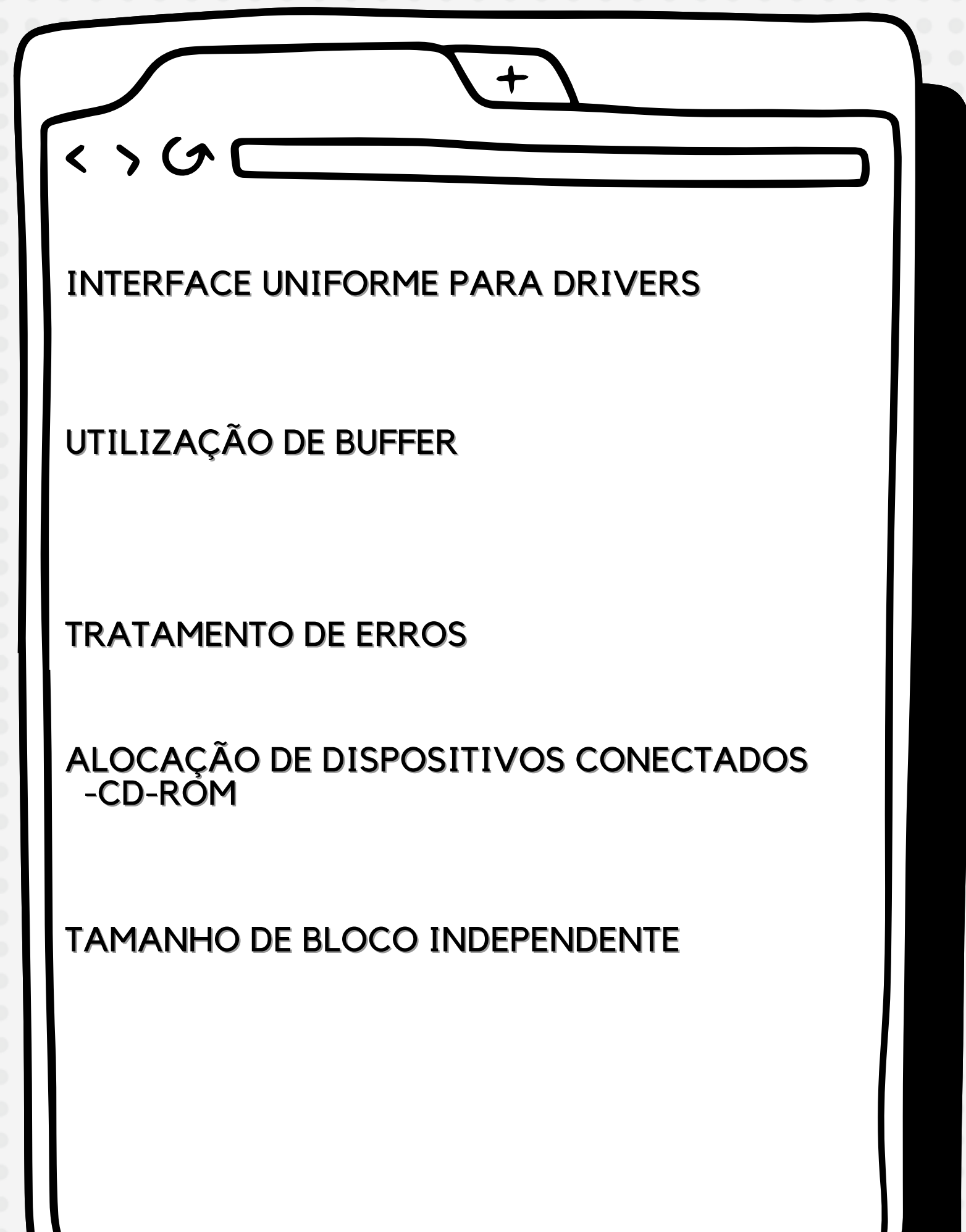


SOFTWARE DE E/S
INDEPENDENTE DE
DISPOSITIVO
.....

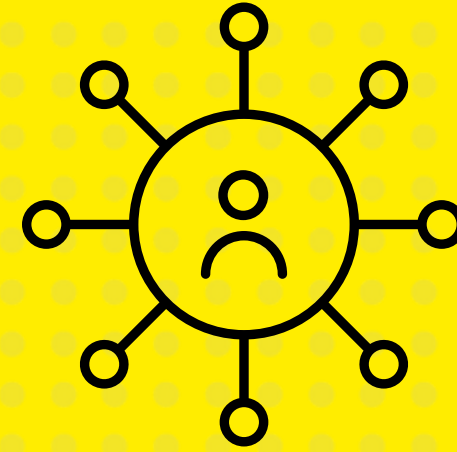




PADRONIZAR



OPERA FORA DO
NÚCLEO

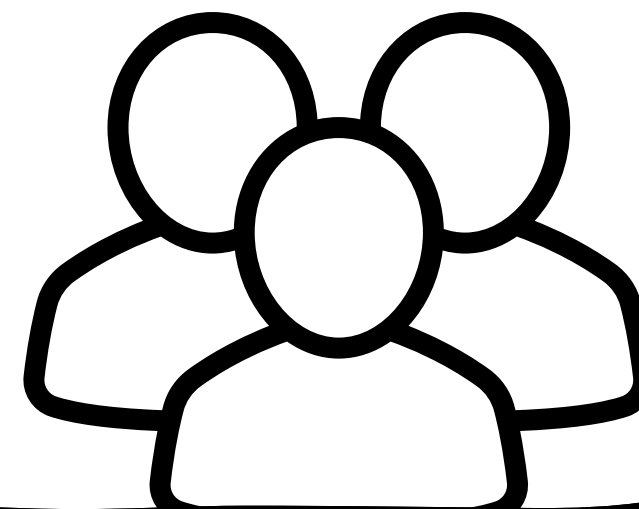


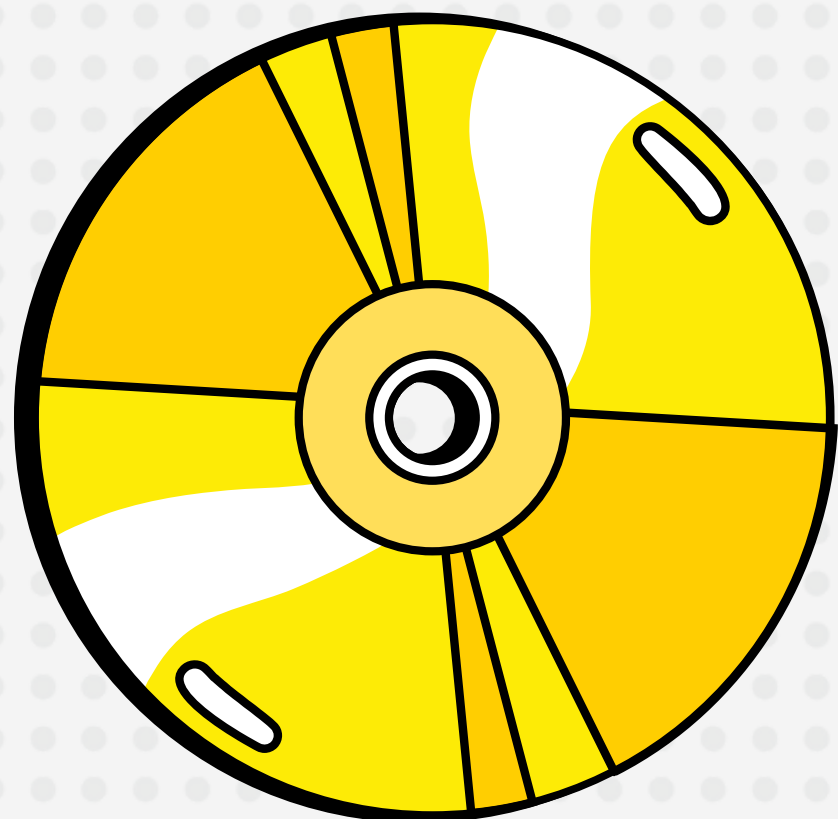
BIBLIOTECAS DE E/S

SPOOLING



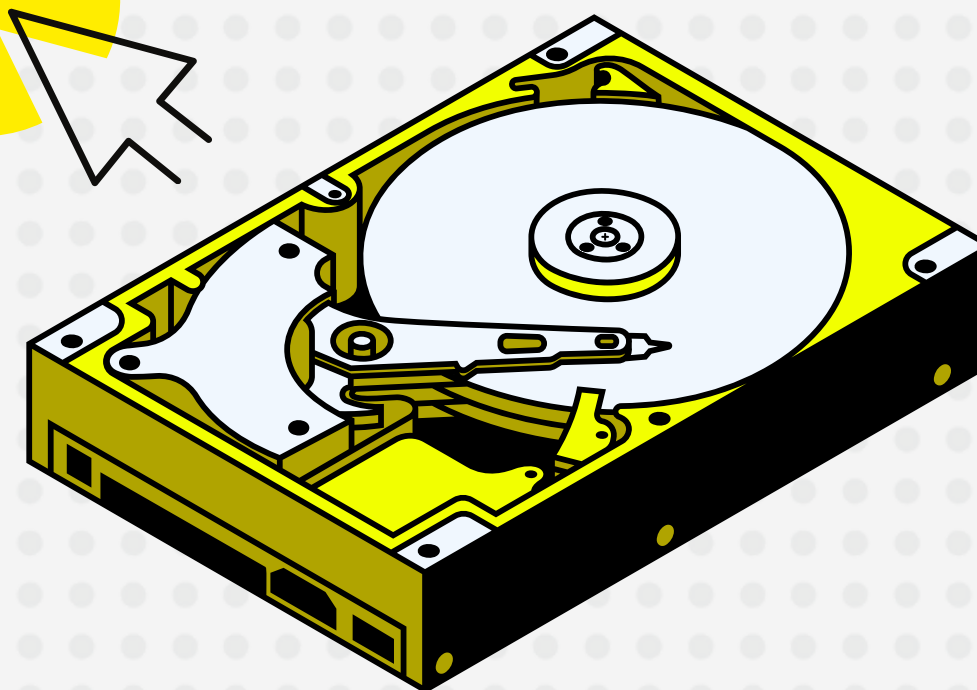
SOFTWARE DE E/S NO ESP.AÇO DO USUÁRIO





DISCOS

PEDRO
SARMENTO



Discos Magnéticos



EXEMPLOS: DISCO RÍGIDO (HDD), DISCO FLEXÍVEL

CARACTERÍSTICAS:

ALTA VELOCIDADE DE LEITURA/ESCRITA;

NÃO VOLÁTIL (IDEAL PARA PAGINAÇÃO, SISTEMAS DE ARQUIVOS);

USADOS EM RAID PARA ARMAZENAMENTO CONFIÁVEL

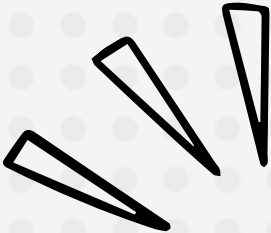
Discos Ópticos



EXEMPLOS: CD-ROM, CD gravável, DVD

Aplicações:

Distribuição de programas, dados e filmes
Armazenamento portátil e de baixo custo



COMPONENTES PRINCIPAIS



PRATOS: DISCOS METÁLICOS
GIRATÓRIOS

CABEÇOTE: "AGULHA" DE
LEITURA/GRAVAÇÃO

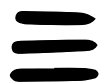
TRILHA: CÍRCULO CONCÊNTRICO
NO PRATO

CILINDRO: TRILHAS ALINHADAS
VERTICALMENTE

SETOR: SUBDIVISÃO DA TRILHA
(8—32 EM FLEXÍVEIS, MILHARES
EM RÍGIDOS)

CONTROLADOR: GERENCIA
LEITURA/GRAVAÇÃO





Gerencia
leitura/gravação

Controla cache e
remapeamento de blocos
defeituosos

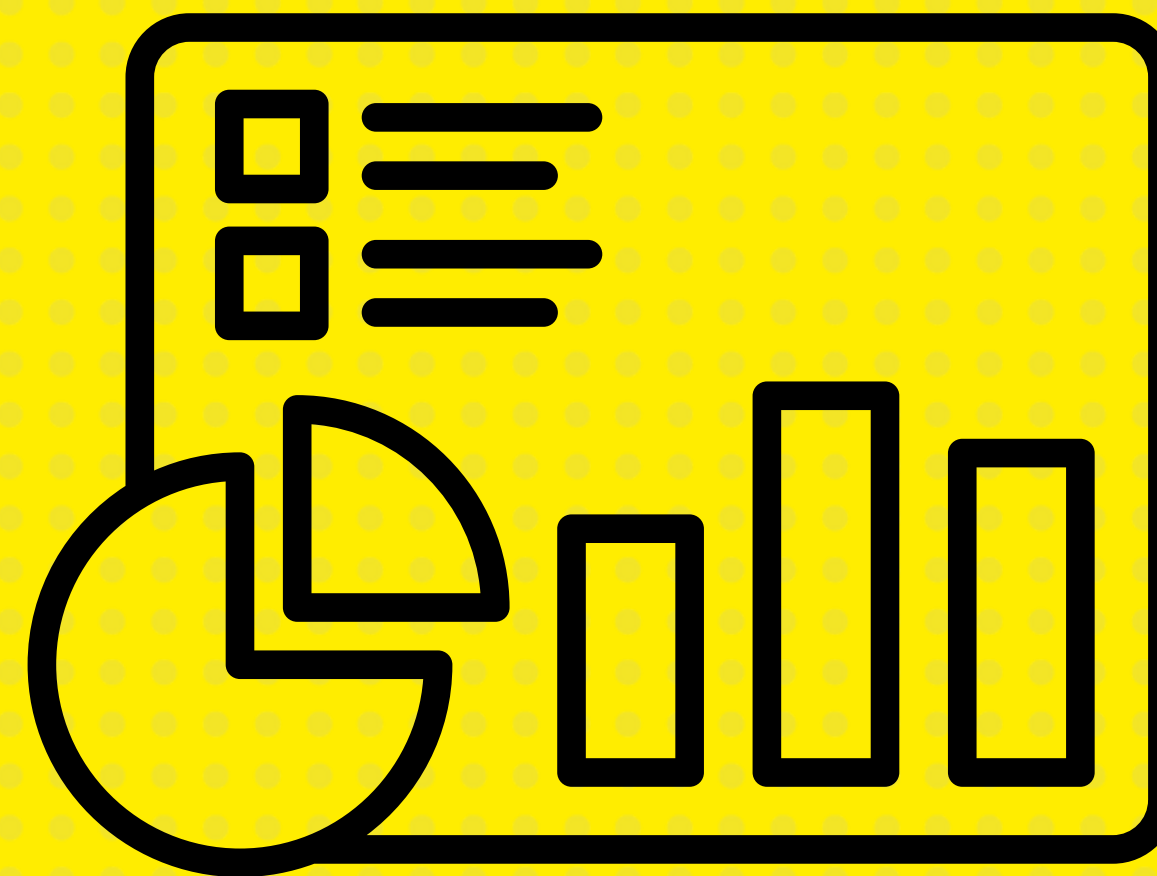
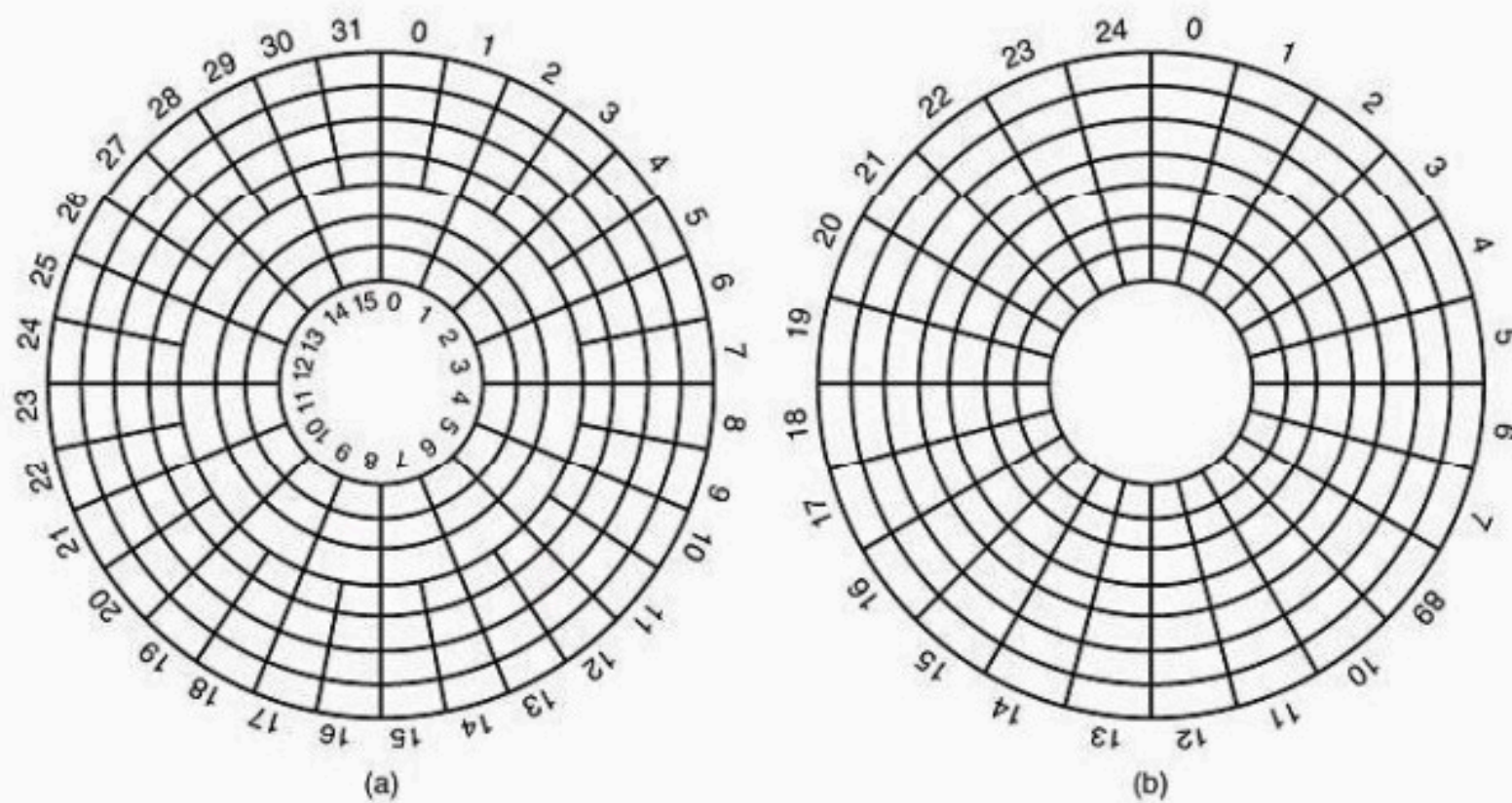
Overlapped seeks:
Posicionamento
simultâneo em múltiplos
discos



CONTROLADORES

GEOMETRIA FÍSICA

Física (Zonas): Mais setores nas zonas externas (devido ao formato circular)



GEOMETRIA VIRTUAL

- Virtual (Driver): Traduz para cilindros/cabeçotes/setores uniformes



RAID Nível 0:

Striping (distribuição circular em discos);
Desempenho alto (E/S paralela);
Sem redundância (falha = perda de dados);

RAID Nível 1:

Espelhamento (discos primários + backup)
Redundância total (falha = usa cópia)
Leitura rápida, escrita normal

RAID Nível 2:

Correção de erro (Hamming);
Sincronização rígida de discos;
Alta sobrecarga (pouco usado)



RAID Nível 3:

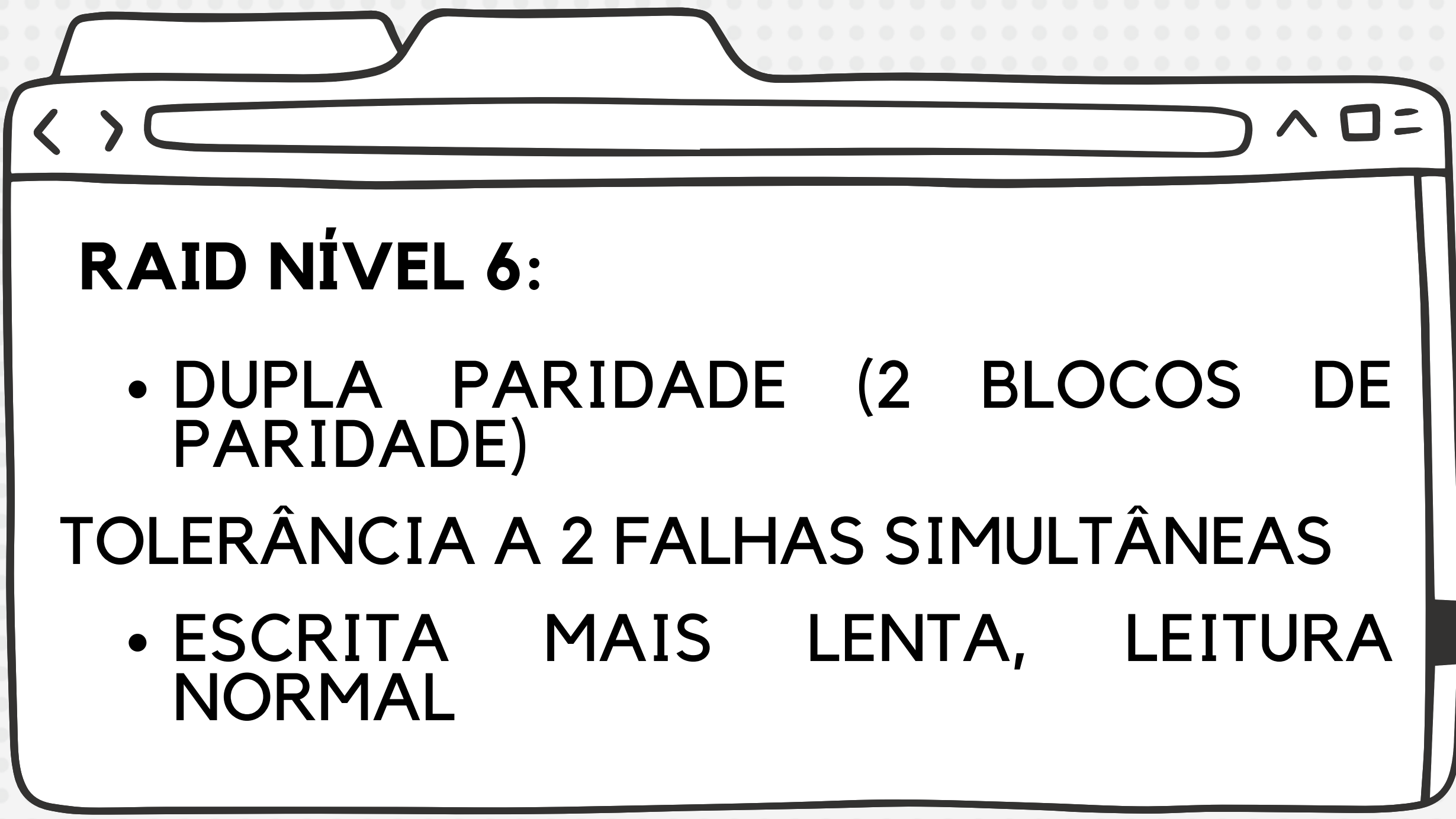
Paridade dedicada (1 disco);
Sincronização de discos
Correção de falha de 1 disco

RAID Nível 4:

Striping + paridade em disco dedicado;
Gargalo no disco de paridade;
Lento para atualizações pequenas;

RAID Nível 5:

Paridade distribuída (evita gargalo)
Balanceamento de carga
Recuperação complexa após falha

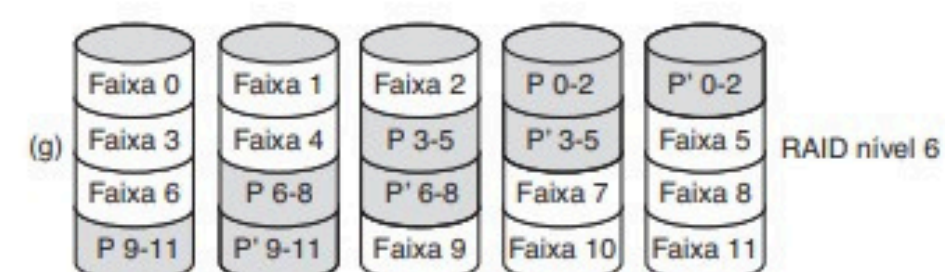
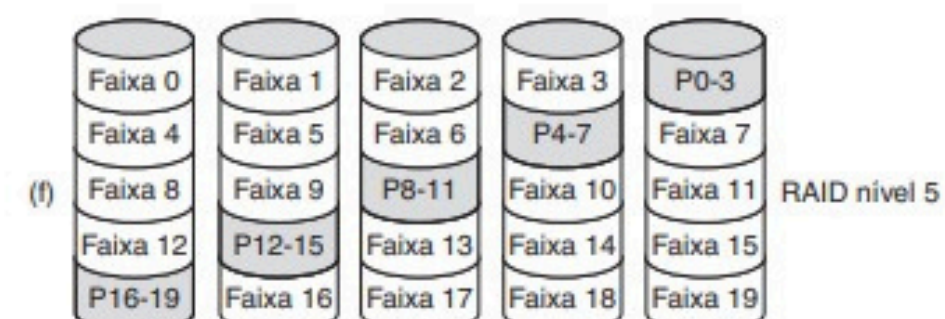
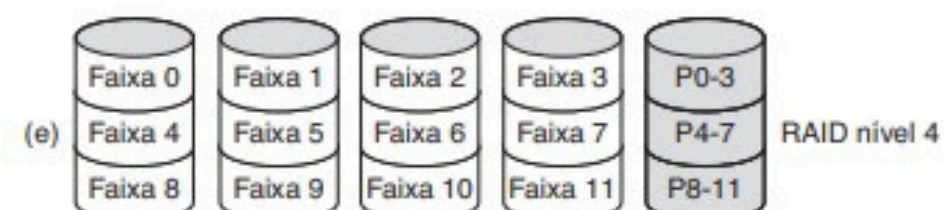
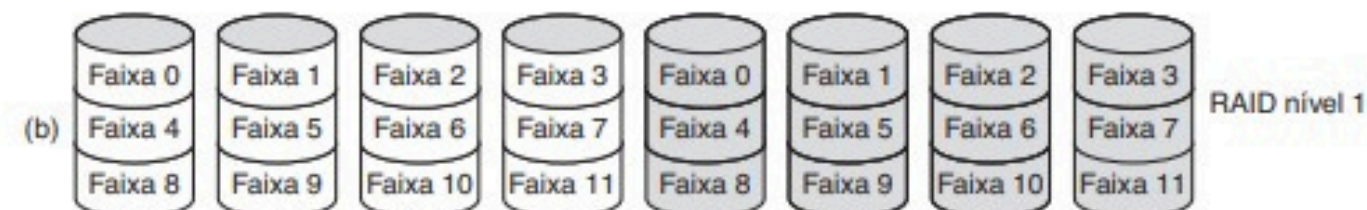
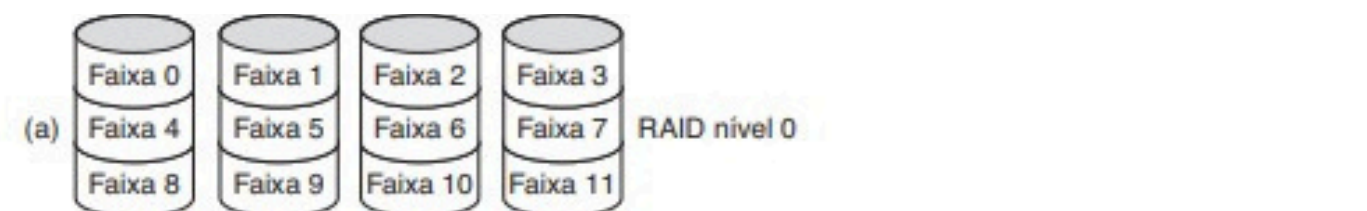


RAID NÍVEL 6:

- DUPLA PARIDADE (2 BLOCOS DE PARIDADE)

TOLERÂNCIA A 2 FALHAS SIMULTÂNEAS

- ESCRITA MAIS LENTA, LEITURA NORMAL



NÍVEIS DA RAID



1. Detecção de Defeitos

Erros menores: Corrigíveis via ECC (até alguns bits);

Erros graves: Requer substituição do setor;

2. Métodos de Substituição (Ver imagem no slide)

Mapeamento direto: Um setor reserva assume o lugar do defeituoso (ex: setor 7 → reserva);

Vantagem: Simples e rápido;

Desvantagem: Pode fragmentar acesso;

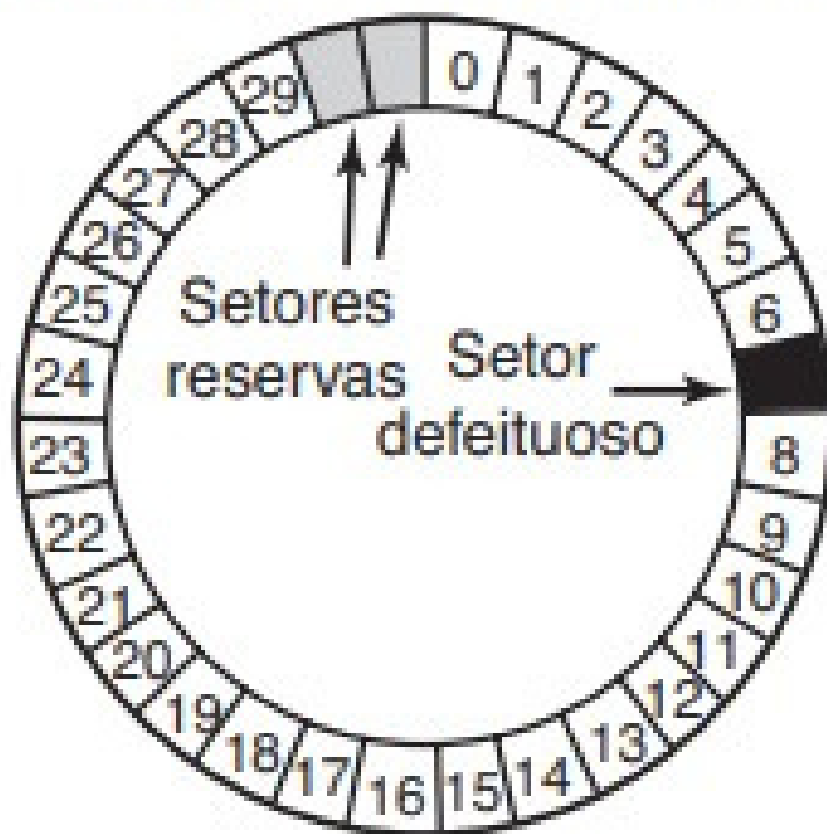
Deslocamento em cadeia: Todos os setores após o defeituoso são realocados (ex: setor 7 vira 8, 8 vira 9...);

Vantagem: Mantém desempenho sequencial;

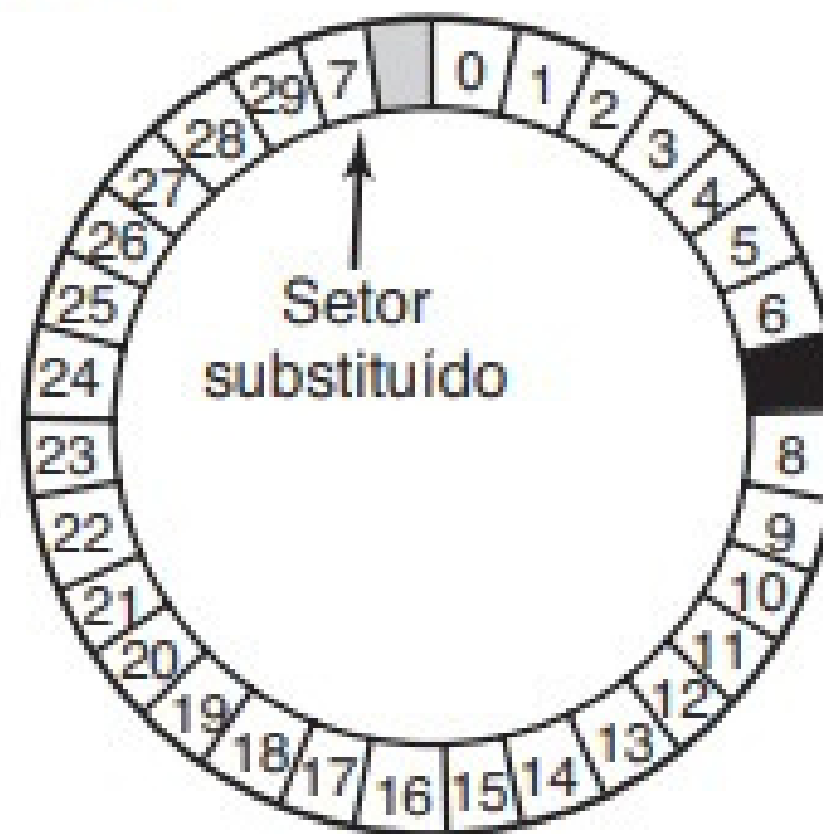
Desvantagem: Complexo (reescreve preâmbulos e dados)

Veja a seguir como funciona :

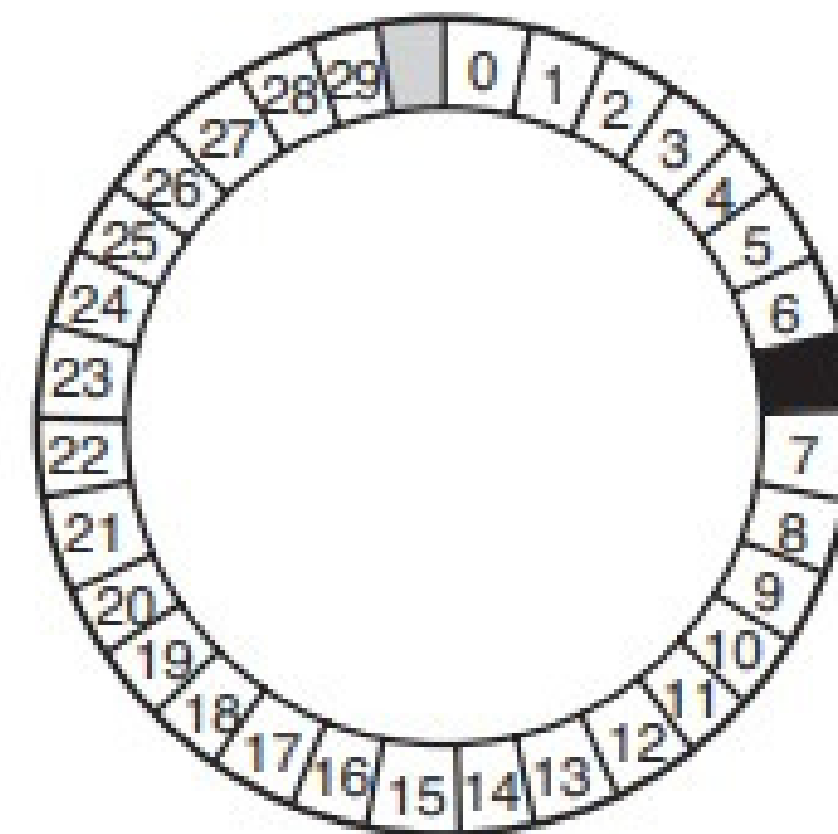
(a) Uma trilha de disco com um setor defeituoso. (b) Substituindo um setor defeituoso com um reserva. (c) Deslocando todos os setores para pular o setor defeituoso.



(a)



(b)

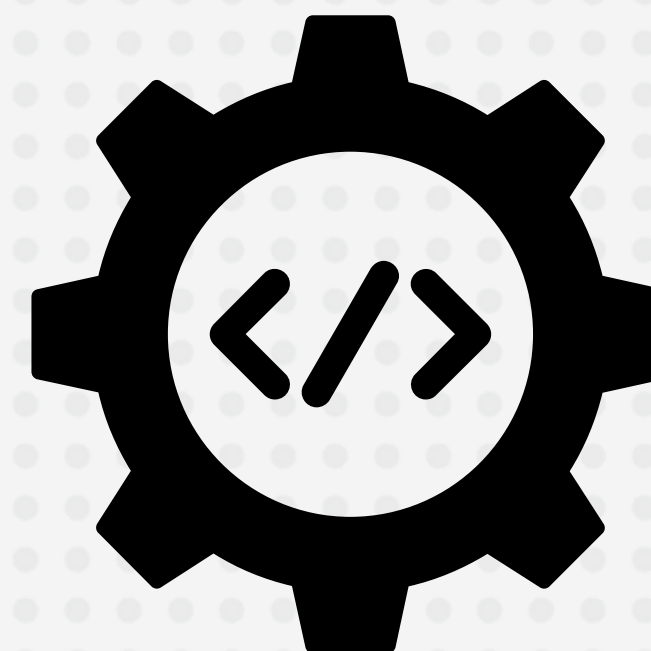


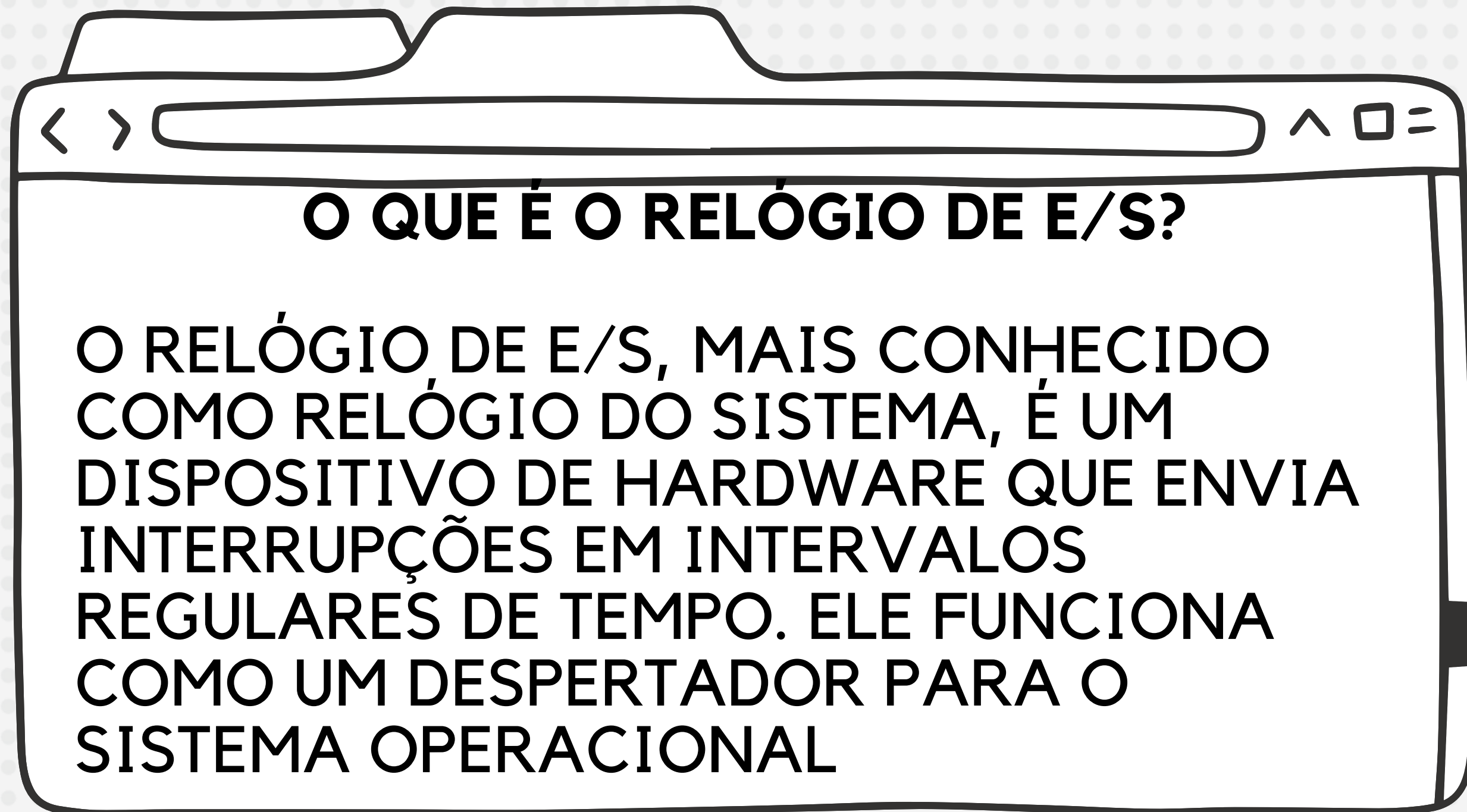
(c)



RELÓGIO

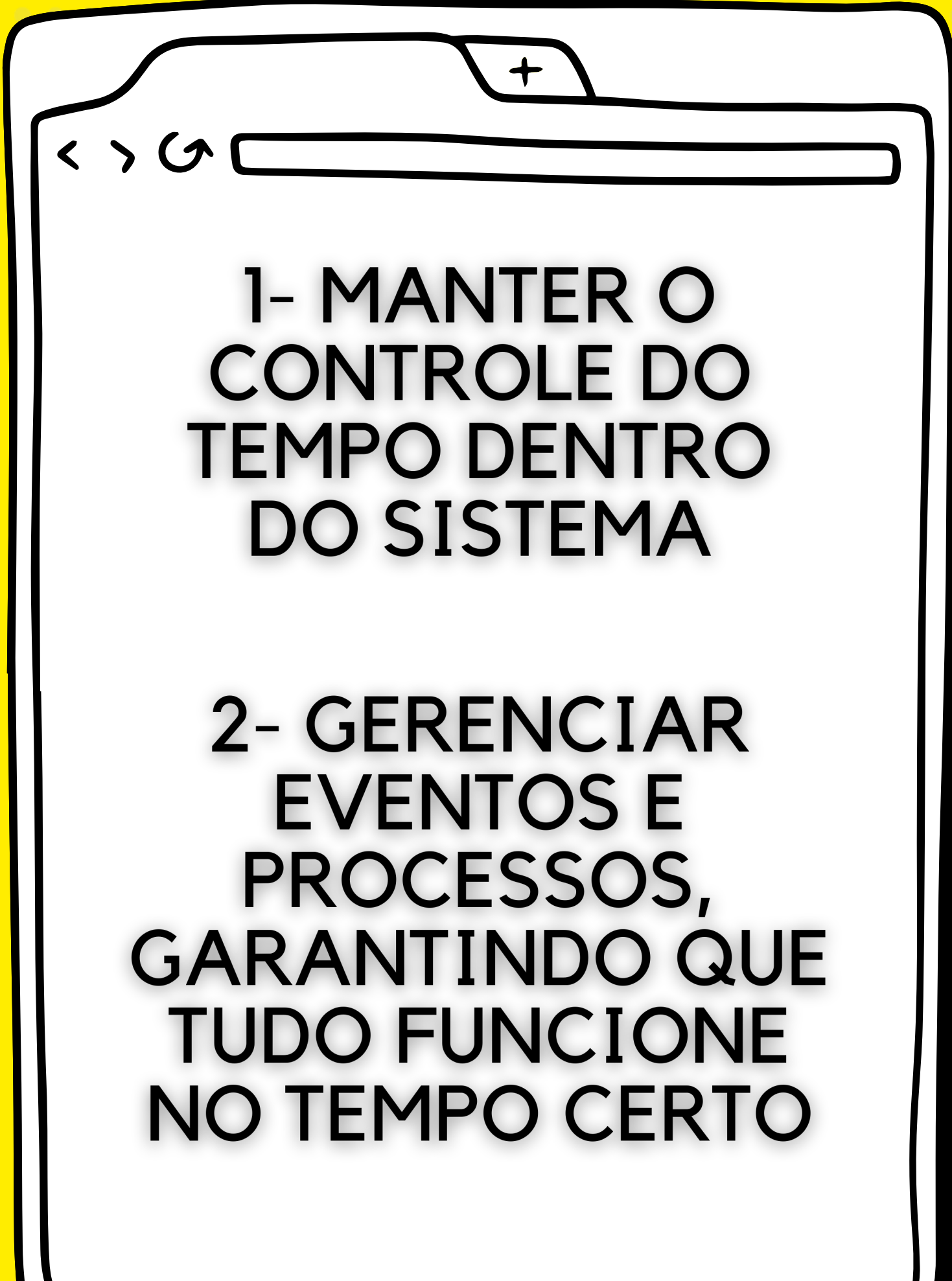
VINÍCIUS
CARNEIRO







FUNÇÕES PRINCIPAIS

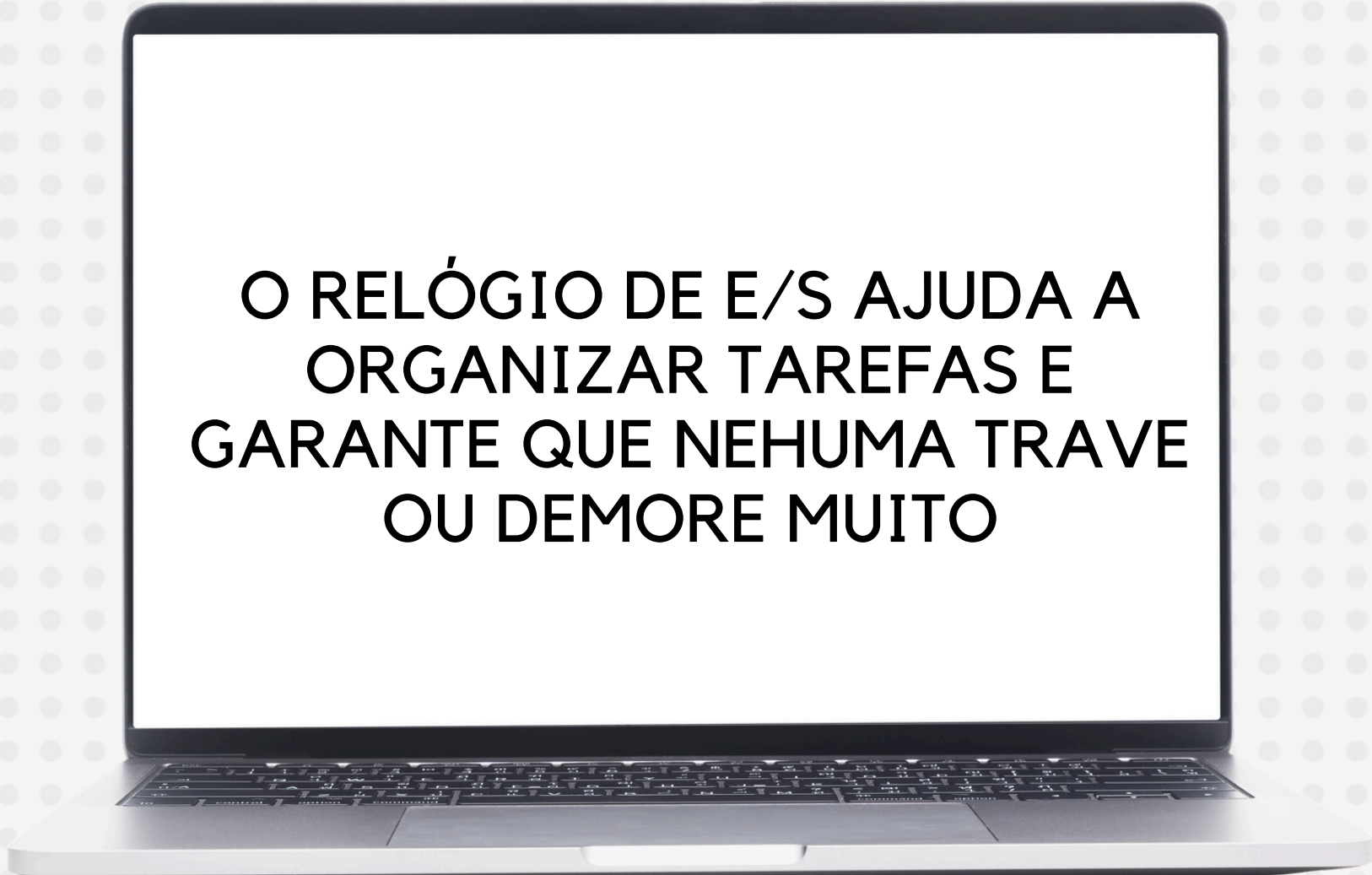


1- MANTER O
CONTROLE DO
TEMPO DENTRO
DO SISTEMA

2- GERENCIAR
EVENTOS E
PROCESSOS,
GARANTINDO QUE
TUDO FUNCIONE
NO TEMPO CERTO



PORQUE É TÃO IMPORTANTE ?



O RELÓGIO DE E/S AJUDA A
ORGANIZAR TAREFAS E
GARANTE QUE NEHUMA TRAVE
OU DEMORE MUITO

•••

Controlar o tempo das interrupções

•••

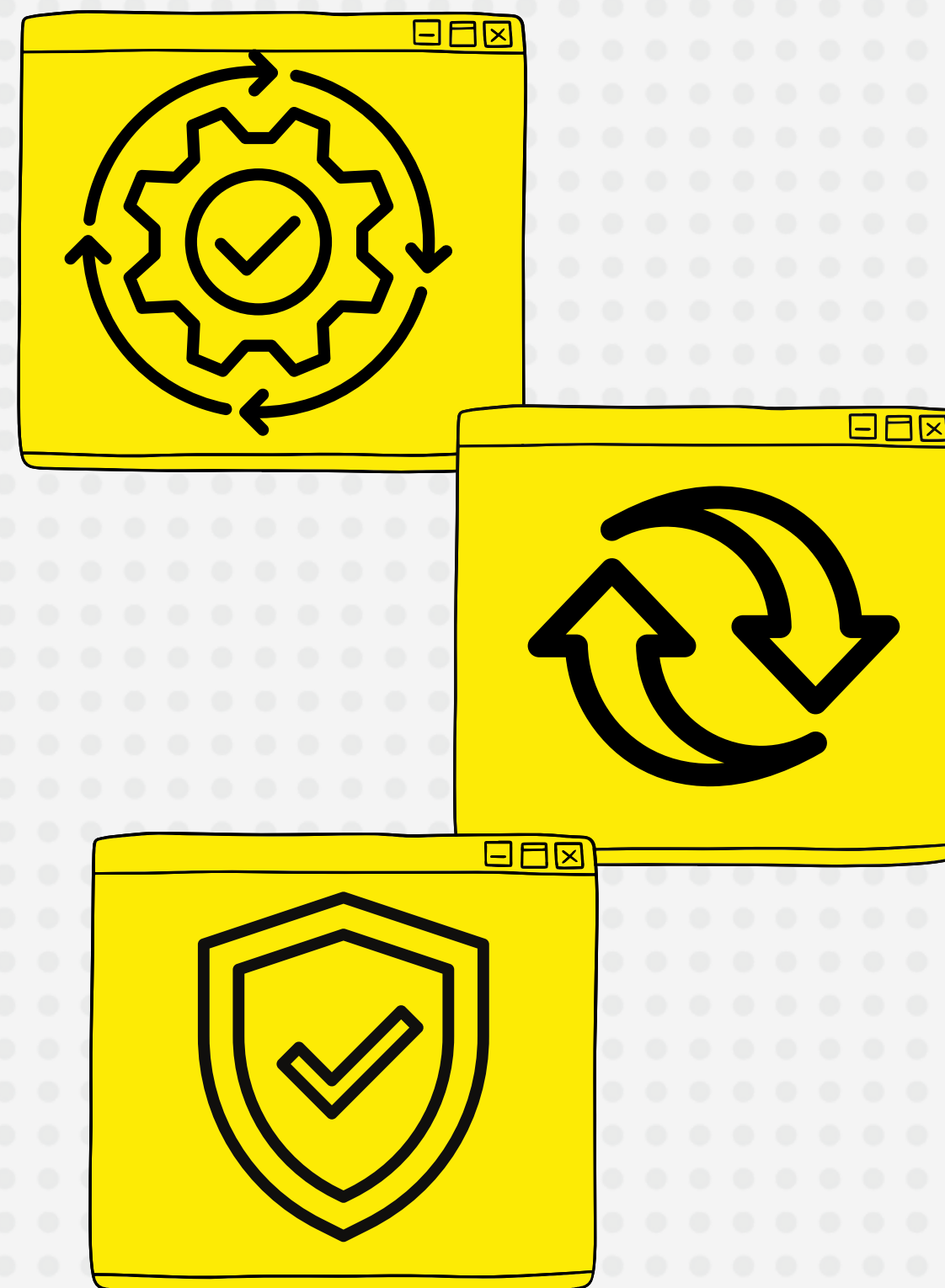
Sincronizar operações de E/S:
(Garante que os dispositivos funcionem
corretamente)

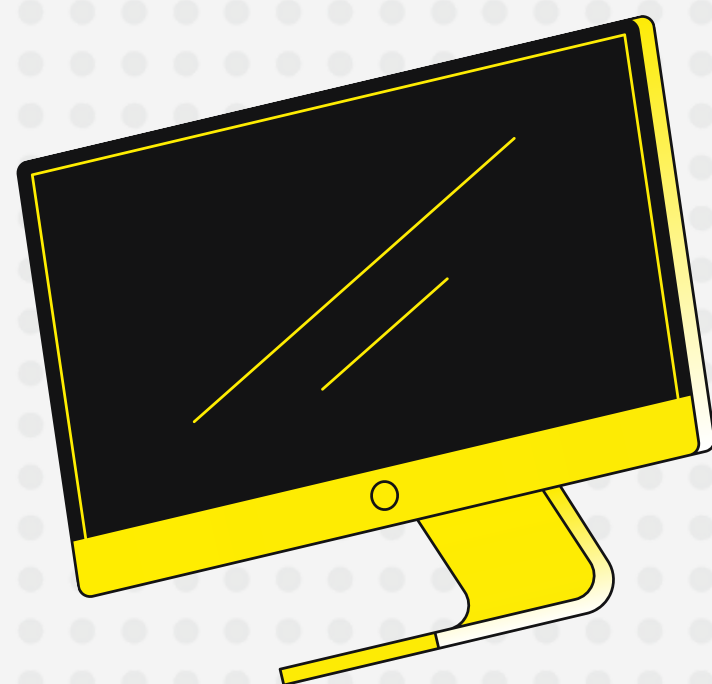
•••

**Evitar que programas fiquem
rodando infinitamente**

COMO QUE ELE FUNCIONA?

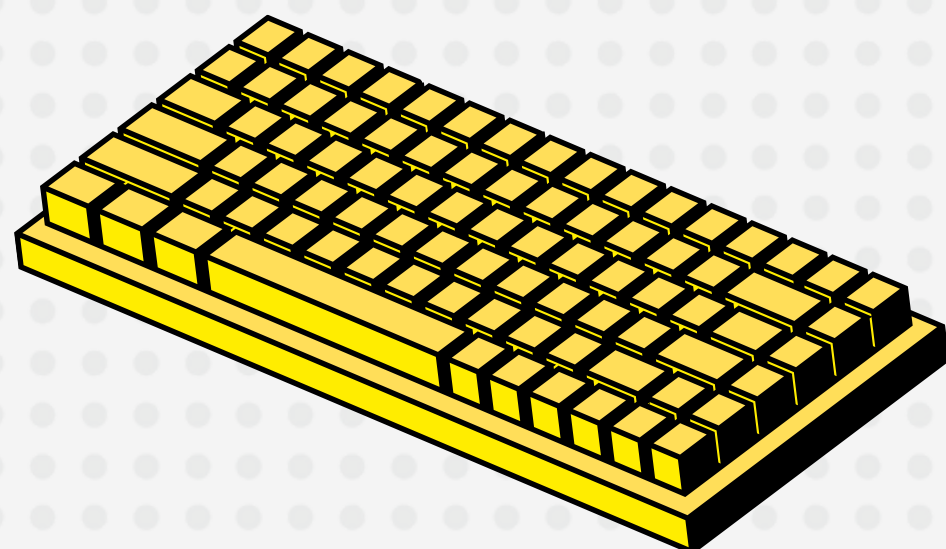
O Relógio de E/S funciona como um cronômetro interno do computador (fica contando o tempo todo). A cada intervalo específico, ele envia um sinal (interrupção) para a CPU. Quando isso acontece o relógio atualiza o tempo do sistema ou executa alguma tarefa que estava programada para aquele momento. Esse processo acontece o tempo todo





INTERFACES COM O USUÁRIO: MOUSE, TECLADO E MONITOR

DAVI COSTA

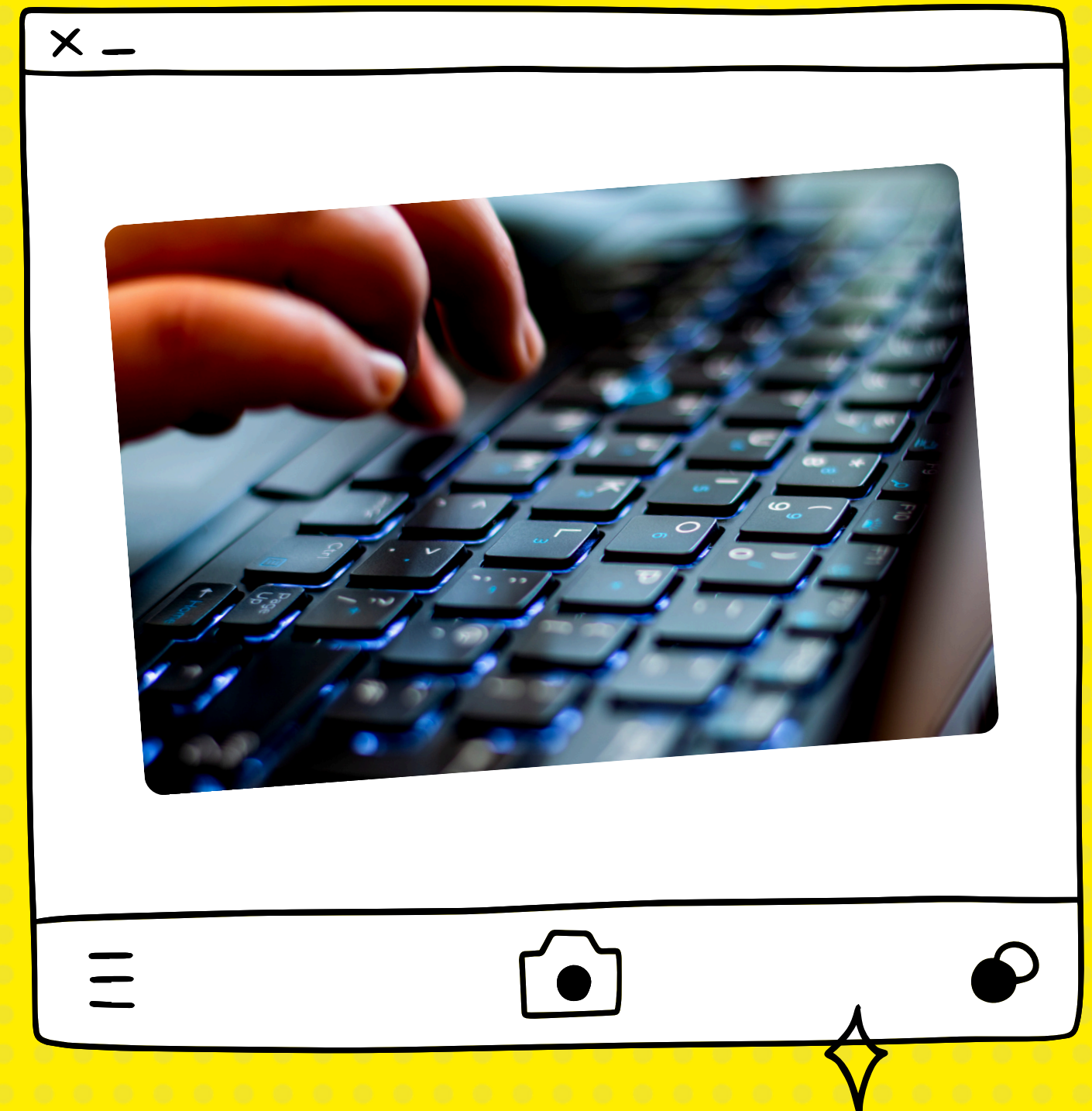




SOFTWARE DE ENTRADA

TECLADO - HARDWARE

Em um computador pessoal, o teclado contém um processador embutido que costuma se comunicar com um chip controlador na placa-mãe através da porta USB. Uma interrupção é gerada sempre que uma tecla é pressionada e outra sempre que a tecla é liberada. Em cada uma dessas interrupções, o driver do teclado extrai a informação sobre o que acontece na porta de E/S associada ao periférico.



TECLADO - SOFTWARE

. **Código de varredura:** o número na porta de E/S é o da tecla e não o código ASCII

**DEPRESS SHIFT, DEPRESS A, RELEASE A,
RELEASE SHIFT**

OU

**DEPRESS SHIFT, DEPRESS A, RELEASE
SHIFT, RELEASE A**

=

A

. O driver do teclado possui duas filosofias:

- O trabalho do driver consiste apenas em aceitar a entrada e passá-la adiante inalterada. Nem todos os programas querem esse nível de detalhe.

(d s t e a t e C R)

- O driver trata toda a edição interna da linha e somente entrega as linhas corrigidas para os programas do usuário.

A primeira filosofia é baseada em caracteres; a segunda, em linhas.

- Raw Mode
- Cooked Mode

. **Eco:** Aparição dos caracteres digitados pelos usuários na tela de vídeo.

1. Programa escreve enquanto usuário digita;
2. Limite de caracteres;
3. Tratamento da tabulação;
4. Equivalência de dispositivo.

```
lukasz@lukasz-PC: ~
Welcome to babun - the windows shell you will love!
You have got a lot of things under the hood here!

pact -> a package manager that handles installation
pact install tar -> will install tar package on y
pact remove tar -> will remove tar package from y

babun -> a script that manages babun and enables aut
babun check -> will check the configuration (prox
babun update -> will update babun to the newest v
babun --welcome -> will display this message again

There is a couple of plugins installed by default: oh-m
There is also a lot of dev tools such as git, svn, unix
If you would like to contribute, code up a plugin or re
go to github: https://github.com/babun/babun

If you like the project star it on github and follow me
This project is created and maintained by Tom Bujok (@t

Executing daily babun check:
Prompt speed      [OK]
Connection check  [OK]
Update check      [OK]
{ ~ } » |
```

CARACTERES ESPECIAIS - MODO CANÔNICO

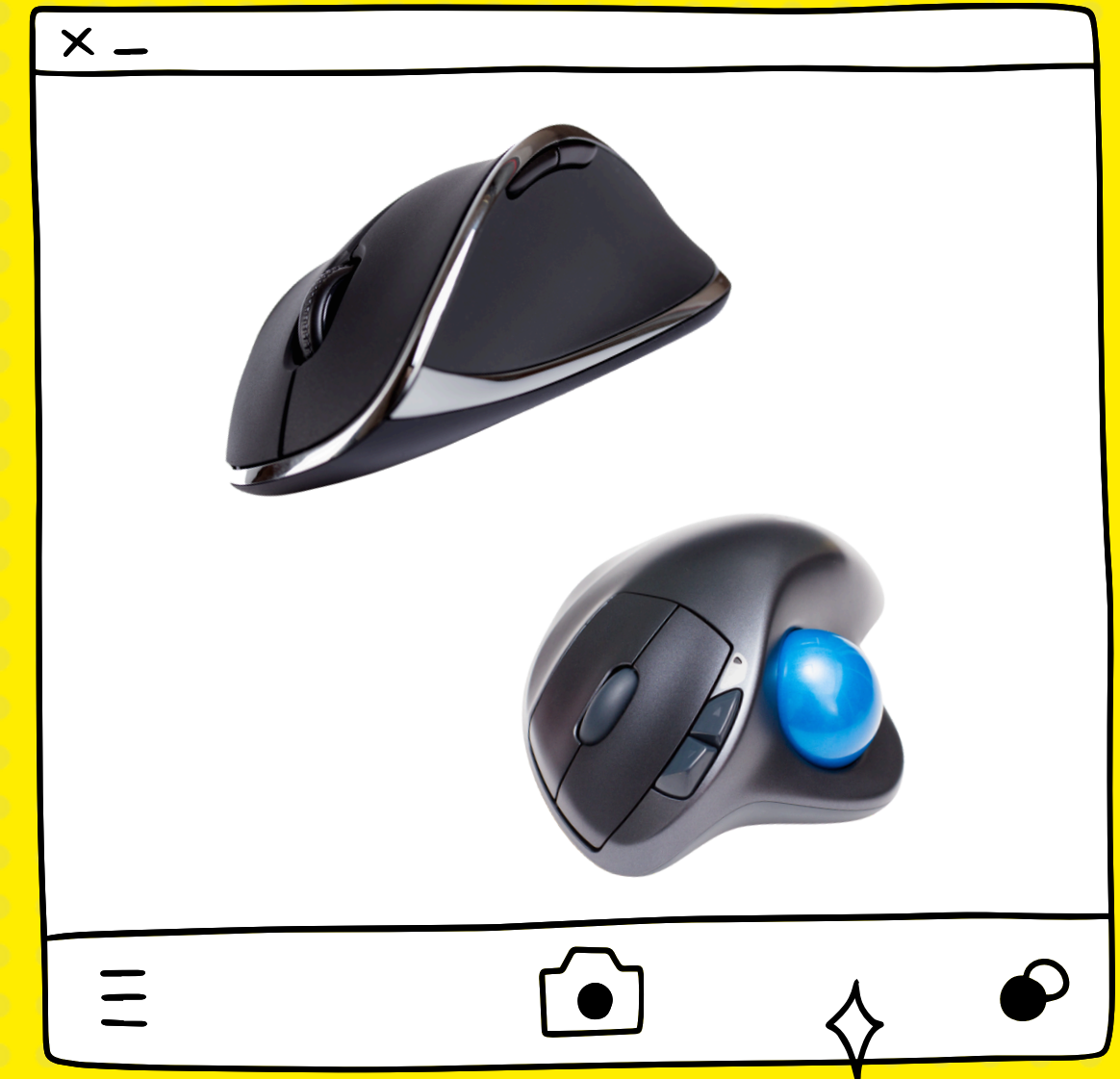
Caractere	Nome POSIX	Comentário
CTRL-H	ERASE	Apagar um caractere à esquerda
CTRL-U	KILL	Apagar toda a linha em edição
CTRL-V	LNEXT	Interpretar literalmente o próximo caractere
CTRL-S	STOP	Parar a saída
CTRL-Q	START	Iniciar a saída
DEL	INTR	Interromper processo (SIGINT)
CTRL-\	QUIT	Forçar gravação da imagem da memória (SIGQUIT)
CTRL-D	EOF	Final de arquivo
CTRL-M	CR	Retorno do carro (não modificável)
CTRL-J	NL	Alimentação de linha (não modificável)

SOFTWARE DE ENTRADA MOUSE

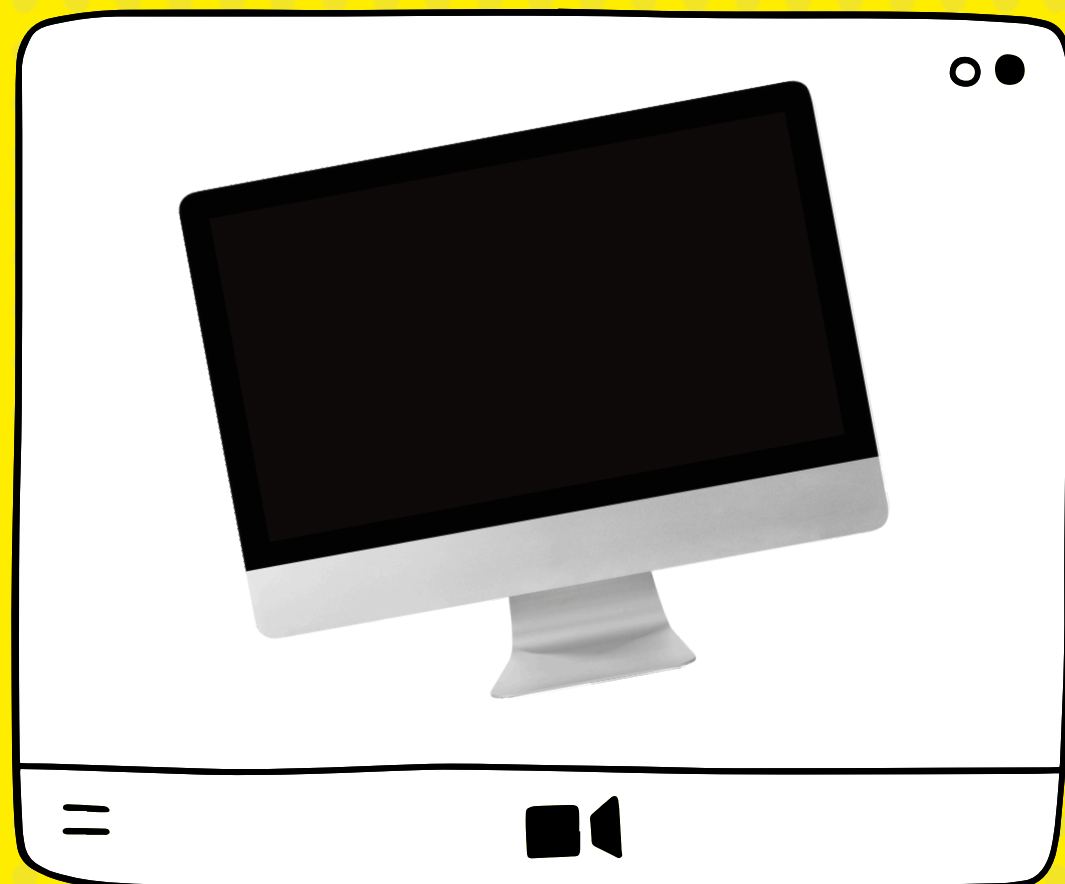


Trackball: Tem uma bola de borracha dentro que sai parcialmente através de um buraco na parte de baixo e gira à medida que o mouse é movido sobre uma superfície áspera.

Mouse óptico: É equipado com um ou mais diodos emissores de luz e fotodetectores na parte de baixo. Contêm um chip de processamento de imagens e tiram fotos de baixa resolução contínuas da superfície debaixo deles, procurando por mudanças de uma imagem para outra.
(mickey)

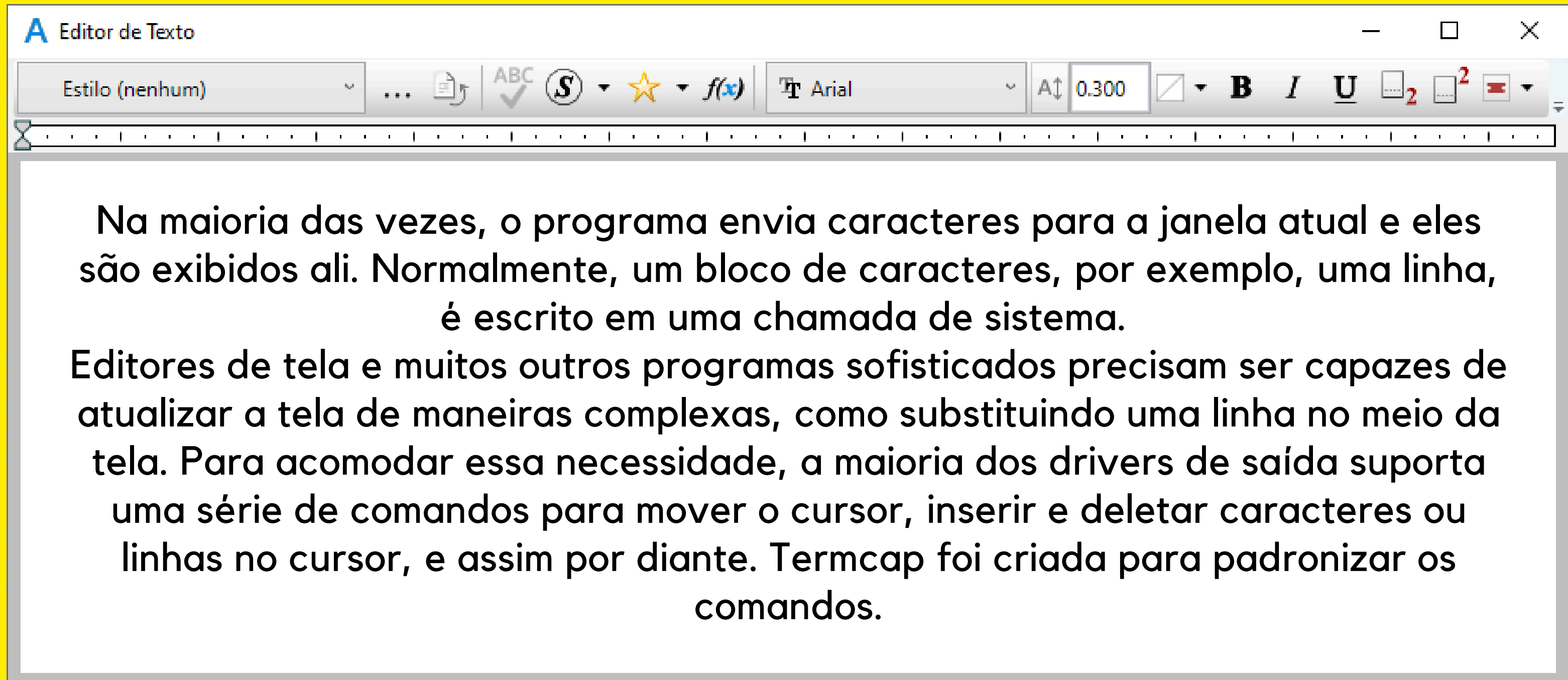


A mensagem para o computador contém três itens:
 Δx , Δy , botões



SOFTWARE DE SAÍDA

JANELA DE TEXTO





TERMCAP

Sequência de escape	Significado
ESC [<i>n</i> A	Mover <i>n</i> linhas para cima
ESC [<i>n</i> B	Mover <i>n</i> linhas para baixo
ESC [<i>n</i> C	Mover <i>n</i> espaços para a direita
ESC [<i>n</i> D	Mover <i>n</i> espaços para a esquerda
ESC [<i>m</i> ; <i>n</i> H	Mover o cursor para (<i>m</i> , <i>n</i>)
ESC [<i>s</i> J	Limpar a tela a partir do cursor (0 até o final, 1 a partir do início, 2 para ambos)
ESC [<i>s</i> K	Limpar a linha a partir do cursor (0 até o final, 1 a partir do início, 2 para ambos)
ESC [<i>n</i> L	Inserir <i>n</i> linhas a partir do cursor
ESC [<i>n</i> M	Excluir <i>n</i> linhas a partir do cursor
ESC [<i>n</i> P	Excluir <i>n</i> caracteres a partir do cursor
ESC [<i>n</i> @	Inserir <i>n</i> caracteres a partir do cursor
ESC [<i>nm</i>	Habilitar efeito <i>n</i> (0 = normal, 4 = negrito, 5 = piscante, 7 = reverso)
ESC M	Rolar a tela para cima se o cursor estiver na primeira linha

O X Window System (ou simplesmente X) é um sistema de janelas gráfico usado para fornecer uma interface gráfica de usuário (GUI) em sistemas Unix e Linux. O X Window System segue uma arquitetura cliente-servidor.

1. **Servidor X (X Server):** O software que coleta a entrada do teclado e mouse e escreve a saída para a tela é chamado de servidor X. O servidor é o responsável por controlar o hardware gráfico e gerenciar a exibição das janelas na tela. Ele lida com as interações com o monitor, o teclado, o mouse e outros dispositivos de entrada, os quais ele envia os dados aos clientes. O servidor X "entende" os comandos gráficos que são emitidos pelos aplicativos (clientes), e exibe as janelas no monitor.

2. **Cliente X (X Client):** Os clientes são os programas que solicitam ao servidor a exibição de gráficos e interações com o usuário. Cada aplicação gráfica (como um navegador, editor de texto, etc.) é um cliente X. O cliente X comunica-se com o servidor para desenhar janelas, gerenciar entradas do teclado e mouse, etc.



SISTEMA X - WINDOW



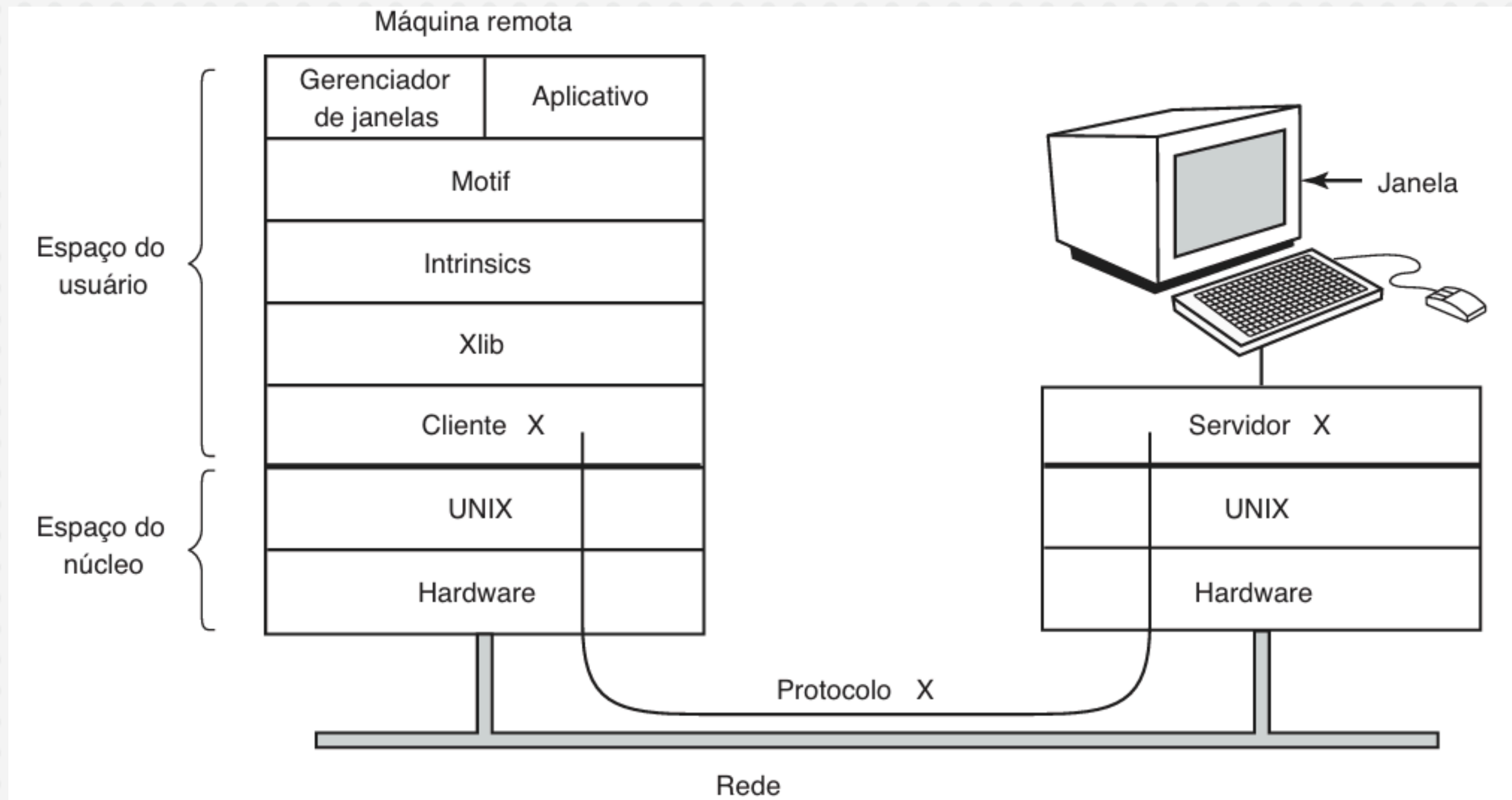
PRINCIPAIS CARACTERÍSTICAS X

- WINDOW

- **Independência do hardware:** O X foi projetado para ser independente do hardware, o que significa que ele pode ser executado em diversos tipos de dispositivos gráficos e de entrada, além de ser capaz de rodar em diferentes plataformas de hardware.
- **Arquitetura distribuída:** O X permite que o servidor de exibição e os clientes estejam em máquinas diferentes. Ou seja, um cliente pode rodar em uma máquina enquanto o servidor X (que exibe as janelas) pode estar em outra máquina na rede. Isso facilita o uso de interfaces gráficas remotas (como em um servidor e o cliente acessando a interface via rede).
- **Protocolos de comunicação:** O protocolo X permite a comunicação entre o servidor e os clientes. Ele pode transmitir comandos como "desenhar um botão" ou "mover uma janela", e também pode capturar eventos de entrada, como "clique do mouse" ou "pressionamento de tecla". A razão de ser possível executar o sistema X Window no UNIX (ou outro sistema operacional) em uma única máquina ou através de uma rede é o fato de que o que o X realmente define é o protocolo X entre o cliente X e o servidor X, como mostrado na figura.

PRINCIPAIS CARACTERÍSTICAS X

- WINDOW

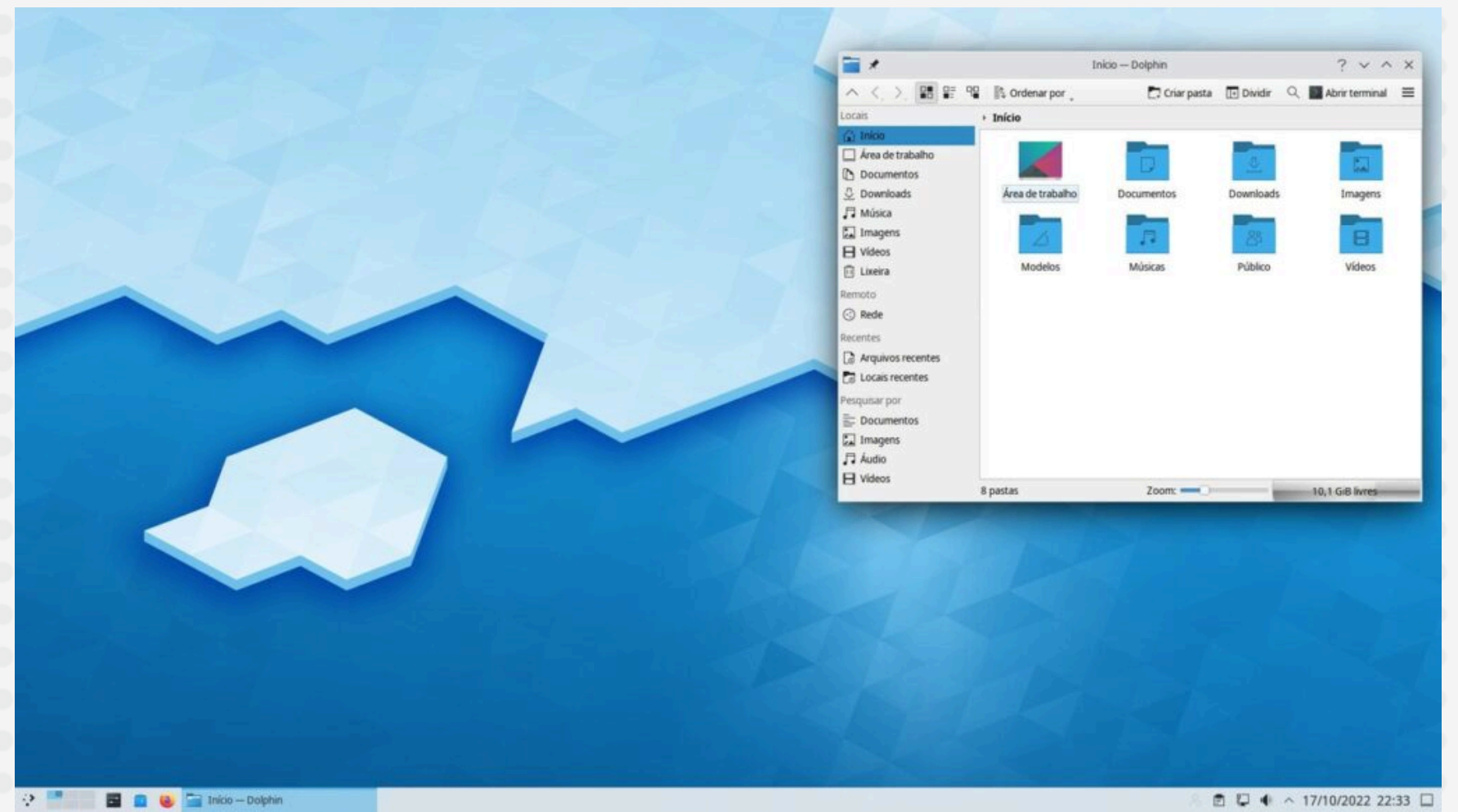
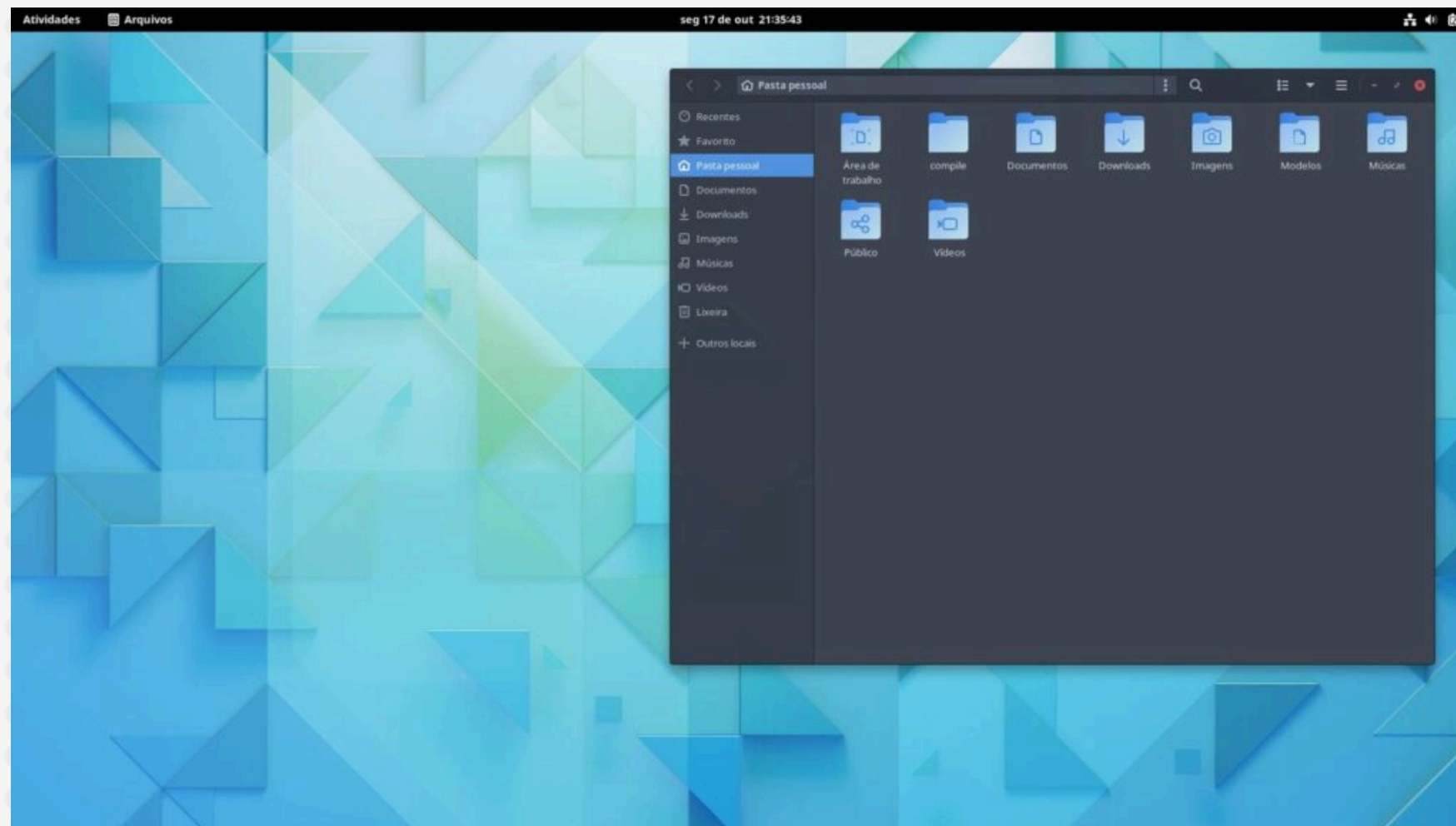


PRINCIPAIS CARACTERÍSTICAS X

- WINDOW

- **Gerenciamento de janelas:** O X não tem um "gerenciador de janelas" por padrão. Ele apenas fornece o suporte básico para desenhar e manipular janelas. O gerenciamento de janelas é feito por programas chamados de gerenciadores de janelas (como o Openbox, i3, GNOME Shell, KDE Plasma, etc.), que controlam como as janelas aparecem, se sobrepõem, são redimensionadas e organizadas na tela.
- **O X não é uma GUI:** Ele não é uma GUI completa. Para obter uma GUI completa, outras camadas de software são executadas sobre ele. Uma camada é a Xlib, um conjunto de rotinas de biblioteca para acessar a funcionalidade do X. Essas rotinas formam a base do Sistema X Window. Para tornar a programação com X mais fácil, um toolkit consistindo em Intrinsics é fornecido como parte do X. Essa camada gerencia botões, barras de rolagem e outros elementos da GUI chamados widgets. Para produzir uma verdadeira interface GUI, com aparência e sensação uniformes, outra camada é necessária (ou várias delas). Um exemplo é o Motif que é a base do Common Desktop Environment (Ambiente de Desktop Comum) usado no Solaris e outros sistemas UNIX comerciais.

PRINCIPAIS CARACTERÍSTICAS X - WINDOW



INTERFACE GRÁFICA DE USUÁRIO (GUI)

A GUI tem quatro elementos essenciais, denotados pelos caracteres **WIMP**:

W: Window

I : Icons

M: Menu

P: Pointing device

A entrada de dados para sistemas GUI ainda usa o teclado e o mouse, mas a saída quase sempre vai para um dispositivo de hardware especial chamado adaptador gráfico.

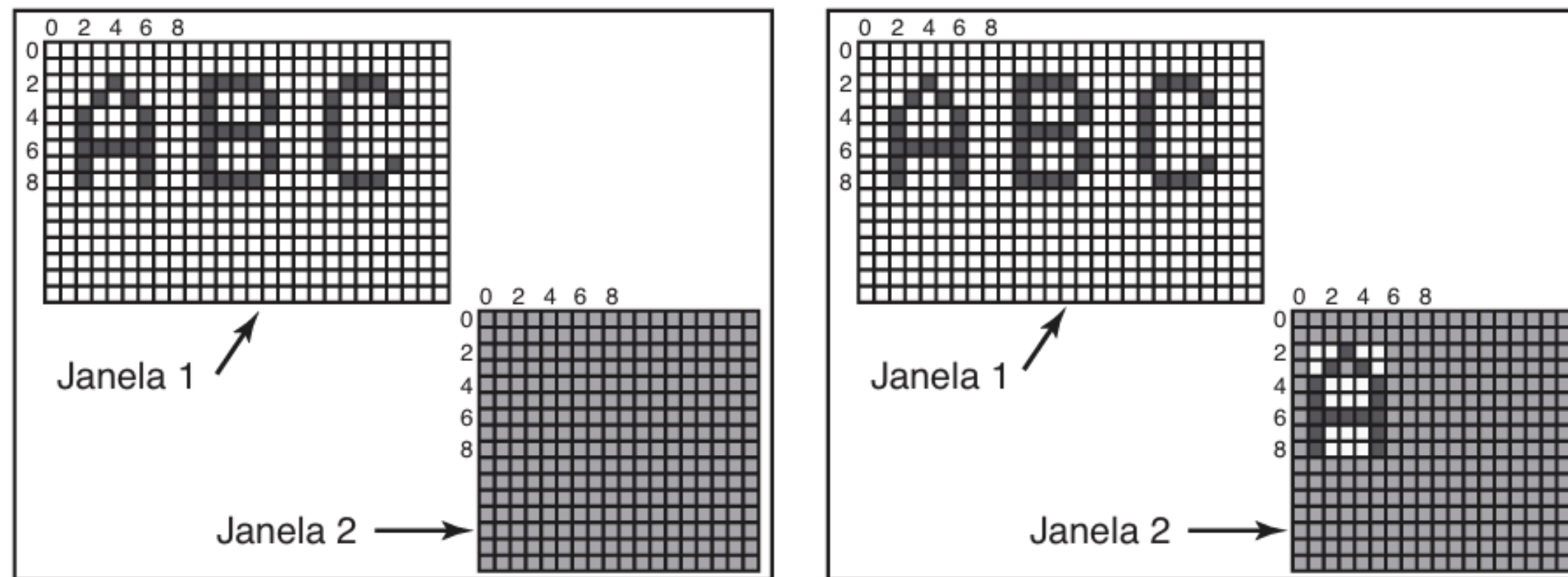


INTERFACE DO DISPOSITIVO GRÁFICO (GDI — GRAPHICS DEVICE INTERFACE)

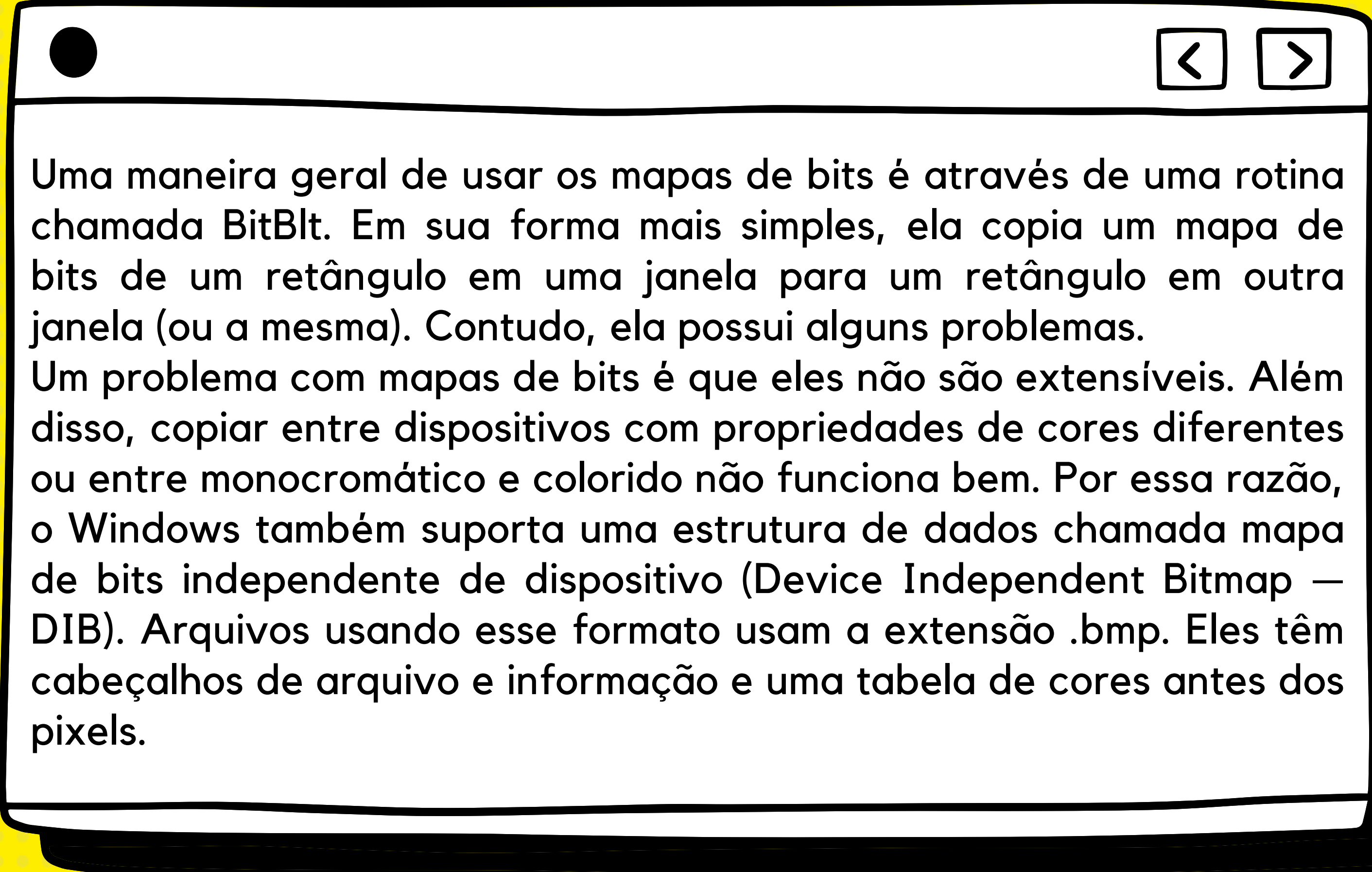
É um sistema gráfico de vídeo usado pelas aplicações para exibir ou imprimir texto em mapa de bits (fontes TrueType), imagens e outros elementos gráficos. A GDI é uma biblioteca de funções do Windows que permite que as aplicações gerem gráficos em uma variedade de dispositivos de saída, incluindo telas, impressoras e dispositivos de imagem. As rotinas GDI são exemplos de gráficos vetoriais. Eles são usados para colocar figuras geométricas e texto na tela. Podem ser escalados facilmente para telas maiores ou menores (desde que o número de pixels na tela seja o mesmo). Uma coleção de chamadas para rotinas GDI pode ser reunida em um arquivo que consiga descrever um desenho complexo. Esse arquivo é chamado de um meta-arquivo do Windows (.wmf), e é amplamente usado para transmitir desenhos de um programa do Windows para outro.

BITMAPS

No computador, há imagens que para serem manipuladas precisam ser escaneadas sobrepondo-se uma grade na imagem. Os valores médios do vermelho, verde e azul de cada quadrante da grade são então amostrados e salvos como o valor de um pixel. Esse arquivo é chamado de mapa de bits (bitmap).



BITMAPS



Uma maneira geral de usar os mapas de bits é através de uma rotina chamada BitBlt. Em sua forma mais simples, ela copia um mapa de bits de um retângulo em uma janela para um retângulo em outra janela (ou a mesma). Contudo, ela possui alguns problemas.

Um problema com mapas de bits é que eles não são extensíveis. Além disso, copiar entre dispositivos com propriedades de cores diferentes ou entre monocromático e colorido não funciona bem. Por essa razão, o Windows também suporta uma estrutura de dados chamada mapa de bits independente de dispositivo (Device Independent Bitmap — DIB). Arquivos usando esse formato usam a extensão .bmp. Eles têm cabeçalhos de arquivo e informação e uma tabela de cores antes dos pixels.



FONTES



Antes do Windows 3.1, os caracteres na tela eram armazenados como imagens (bitmaps). Isso gerava dois grandes problemas:

- . O mesmo bitmap usado na tela ficava pequeno demais para a impressora, porque esta tem maior resolução.
- . Para cada tamanho de letra (exemplo: 10 pontos, 12 pontos, 14 pontos...), era necessário um novo bitmap, o que ocupava muito espaço.

Para resolver esses problemas, foram criadas as fontes TrueType, que funcionam de forma diferente.

COMO FUNCIONA O TRUE TYPE?

- Cada letra é desenhada com uma sequência de pontos.
- O computador pode ampliar ou reduzir a letra simplesmente multiplicando os pontos pelo tamanho desejado.
- Depois de ajustada ao tamanho correto, a letra é "preenchida" e convertida em imagem para exibição na tela ou impressão.

Vantagens:

- As fontes podem ser redimensionadas para qualquer tamanho sem perda de qualidade.
- O mesmo arquivo de fonte pode ser usado para telas e impressoras, evitando desperdício de espaço.
- Pequenos ajustes ("dicas" ou hinting) melhoram a exibição, corrigindo detalhes para evitar falhas visuais.

20 pt: abcdefgh

53 pt: abcdefgh

81 pt: abcdefgh

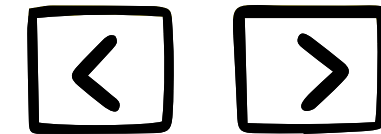


CLIENTES MAGROS

ALBERT
PEREIRA



● THIN CLIENTS



THIN CLIENTS (OU "CLIENTES MAGROS") SÃO DISPOSITIVOS DE COMPUTAÇÃO LEVE COM CONFIGURAÇÕES MÍNIMAS DE HARDWARE, PROJETADOS PARA DEPENDER QUASE INTEIRAMENTE DE UM SERVIDOR CENTRAL (COMO MAINFRAMES OU AMBIENTES EM NUVEM) PARA PROCESSAMENTO, ARMAZENAMENTO E EXECUÇÃO DE APLICATIVOS. ELES FUNCIONAM ESSENCIALMENTE COMO TERMINAIS DE ACESSO REMOTO, COM CAPACIDADE DE PROCESSAMENTO LOCAL LIMITADA OU INEXISTENTE.

Origens nos Mainframes e Terminais Burros (1960–1980)



Na década de 1960, a computação era centralizada em mainframes, computadores de grande porte que atendiam múltiplos usuários simultaneamente. Os terminais burros eram dispositivos conectados a esses mainframes, servindo apenas para exibir informações e enviar alguns comandos, sem capacidade de processamento própria.



Terminal Burro



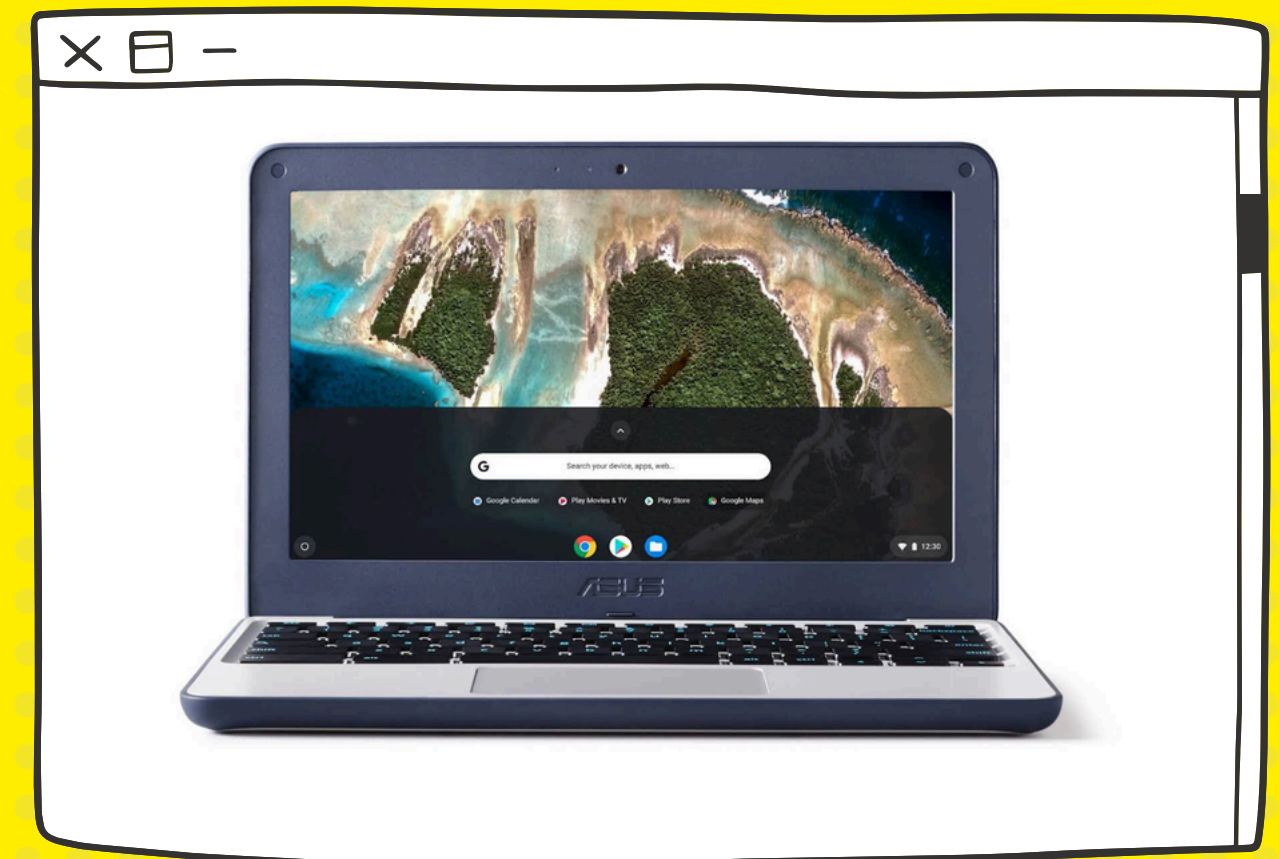
Era do PC e Descentralização (1980–1990)



A partir dos anos 1980, os Personal Computers (PCs) começaram a substituir os terminais burros (monitor e teclado). Com o lançamento do **IBM PC (1981)** e do **Apple Macintosh**, o processamento passou a ser feito localmente, eliminando a dependência de servidores centrais (mainframes). Essa descentralização melhorou a experiência do usuário, mas tornou a administração por parte das empresas mais difícil e cara.



Na década de 1990, os desafios da descentralização levaram à volta do modelo cliente-servidor, agora com uma abordagem moderna: os Thin Clients. Que era a ideia de oferecer terminais com hardware mínimo, deixando o processamento para os servidores centralizados, reduzindo custos e facilitando a manutenção.



RETORNO DA CENTRALIZAÇÃO COM THIN CLIENTS (1990–2000)

GERENCIAME NTO DE ENERGIA

ALBERT
PEREIRA





EXISTEM DUAS ESTRATÉGIAS GERAIS PARA REDUZIR O CONSUMO DE ENERGIA:

- **A PRIMEIRA** CONSISTE NO SISTEMA OPERACIONAL DESLIGAR PARTES DO COMPUTADOR, PRINCIPALMENTE OS DISPOSITIVOS DE ENTRADA E SAÍDA, QUE NÃO ESTEJAM EM USO, POIS UM DISPOSITIVO DESLIGADO USA POUCA OU NENHUMA ENERGIA.
- **A SEGUNDA** É O SISTEMA LIMITAR O USO DE ENERGIA UM DETERMINADO APLICATIVO, POSSIVELMENTE DEGRADANDO A QUALIDADE DA EXPERIÊNCIA DO USUÁRIO COM OBJETIVO DE "ESTICAR" O TEMPO DE BATERIA DE UM NOTEBOOK, POR EXEMPLO.

MODOS

SUSPENDER, HIBERNAR E DESLIGAR

●○○

No modo **Suspender**, o sistema entra em um estado de baixa energia, mantendo a memória RAM ativa enquanto o processador, disco rígido e outros componentes são colocados em modo de baixo consumo ou desligados. Isso permite um retorno quase instantâneo ao uso.

●○○○

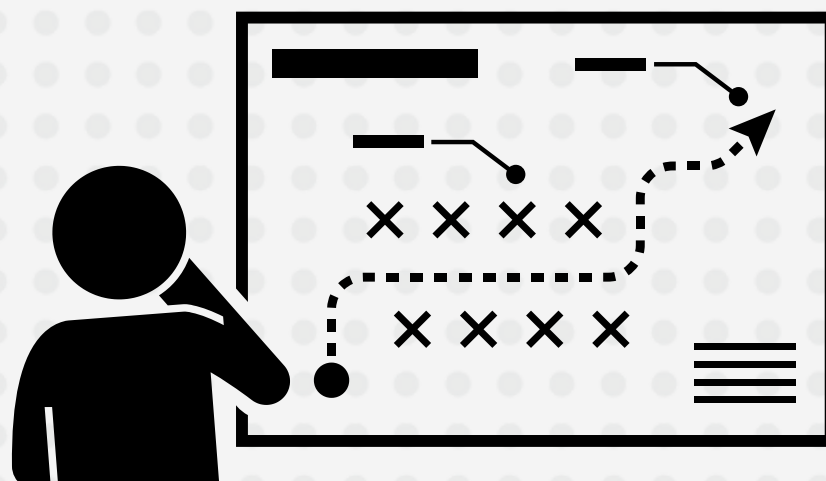
●○○○ Já no modo **Hibernar**, o estado atual do sistema, incluindo os programas e arquivos abertos, é salvo no disco rígido ou SSD, permitindo que o computador seja desligado completamente sem perder o progresso. Dessa forma, ao ligá-lo novamente, o sistema retorna exatamente ao ponto onde estava.

●○○○

Por fim, o modo **Desligar** encerra todos os processos e programas, finaliza as tarefas do sistema operacional e desliga completamente o computador. Ao religá-lo, o sistema começa do zero.

ESTRATÉGIAS

- **NOS MONITORES**, É POSSÍVEL CRIAR ZONAS QUE PODEM SER ATIVADAS OU DESATIVADAS INDIVIDUALMENTE, ECONOMIZANDO ENERGIA.
- **NOS DISCOS RÍGIDOS**, A VELOCIDADE DE ROTAÇÃO PODE SER REDUZIDA QUANDO NÃO ESTÃO EM USO, OU PARTE DA ENERGIA GERADA PELO MOVIMENTO DO DISCO PODE SER REAPROVEITADA PARA REINICIÁ-LO.
- **NA CPU**, O GERENCIAMENTO DE ENERGIA É FEITO POR MEIO DA REDUÇÃO DINÂMICA DA FREQUÊNCIA DO CLOCK E AJUSTE DA VOLTAGEM QUE DIMINUEM O CONSUMO QUANDO A CARGA DE TRABALHO É BAIXA.



- **NAS MEMÓRIAS**, COMO A RAM, É POSSÍVEL ENTRAR EM ESTADO DE BAIXO CONSUMO (DORMÊNCIA) QUANDO OCIOSAS.
- **EM NOTEBOOKS**, PARA EQUILIBRAR TEMPERATURA E O CONSUMO ENERGÉTICO, O SISTEMA OPERACIONAL DEVE DECIDIR ENTRE ATIVAR A VENTOINHA PARA RESFRIAMENTO OU REDUZIR O DESEMPENHO DA CPU E O BRILHO DA TELA.
- NOS MODELOS ATUAIS, HÁ UMA COMUNICAÇÃO ENTRE A BATERIA E SO, QUE MONITORA A VOLTAGEM E A CARGA RESTANTE. ASSIM, O SISTEMA PODE ALERTAR O USUÁRIO QUANTO AO TEMPO RESTANTE ATÉ O DESLIGAMENTO.



FIM

Obrigado pela sua atenção!!!!