

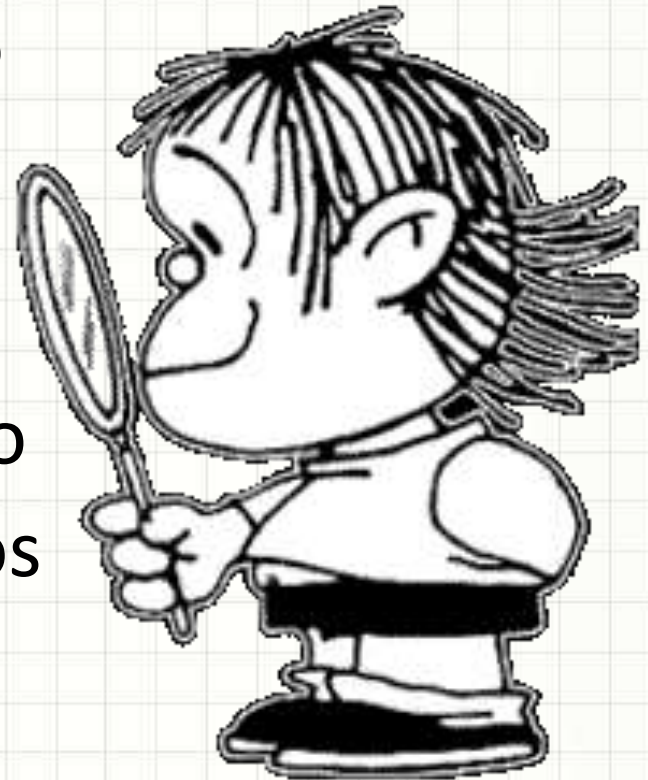
ARQUITETURA E ORGANIZAÇÃO DE COMPUTADORES DISPOSITIVOS DE ENTRADA E SAÍDA

Prof. Dr. Daniel Caetano

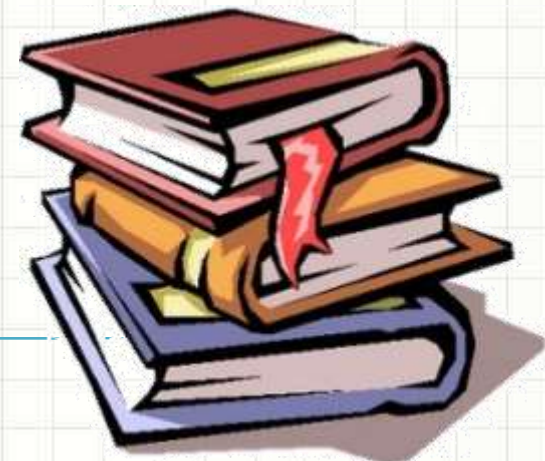
2012 - 2

Objetivos

- Compreender o que é um dispositivo
- Compreender o mecanismo de acionamento de um dispositivo
- Formas de controlar e interagir com um dispositivo
- Conhecer alguns dispositivos comuns
- **AV1!**



Material de Estudo



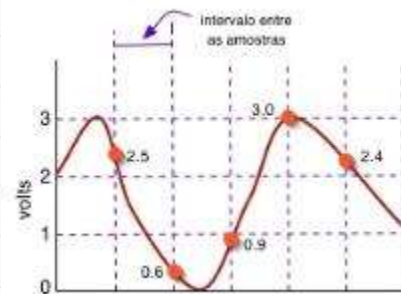
Material	Acesso ao Material
Notas de Aula	http://www.caetano.eng.br/ (Aula 8)
Apresentação	http://www.caetano.eng.br/ (Aula 8)
Material Didático	Arquitetura e Organização de Computadores, páginas 191 a 238
Biblioteca Virtual	-



O QUE É UM DISPOSITIVO?

O que é um dispositivo?

- Simples: converte informações \leftrightarrow sinais
- Dispositivos de Entrada
 - Converte informações do mundo em sinais elétricos (números)

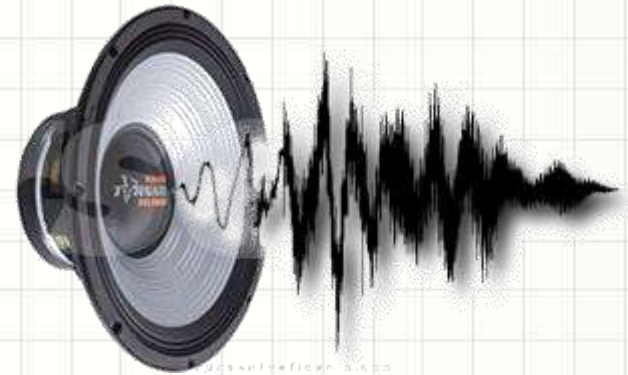
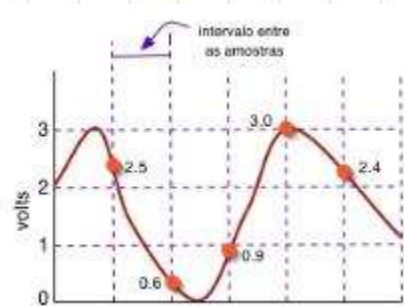


10011001
10011110
10101100
10111001
11001010
11001111
11010011
10111101

O que é um dispositivo?

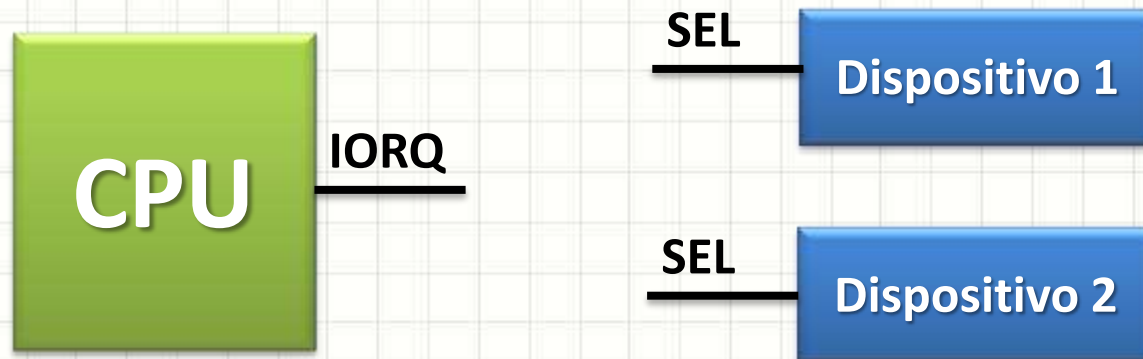
- Dispositivos de Saída
 - Converte sinais (números) em informações do mundo

10011001
10011110
10101100
10111001
11001010
11001111
11010011
10111101



O que é um dispositivo?

- Usualmente: pensam que são únicos
 - Não entendem endereços
 - Possuem apenas um “pino” de ativação
 - O tamanho da palavra é diferente da CPU
- CPU: apenas um “pino” de dispositivo: IORQ
 - Input/Output ReQuest

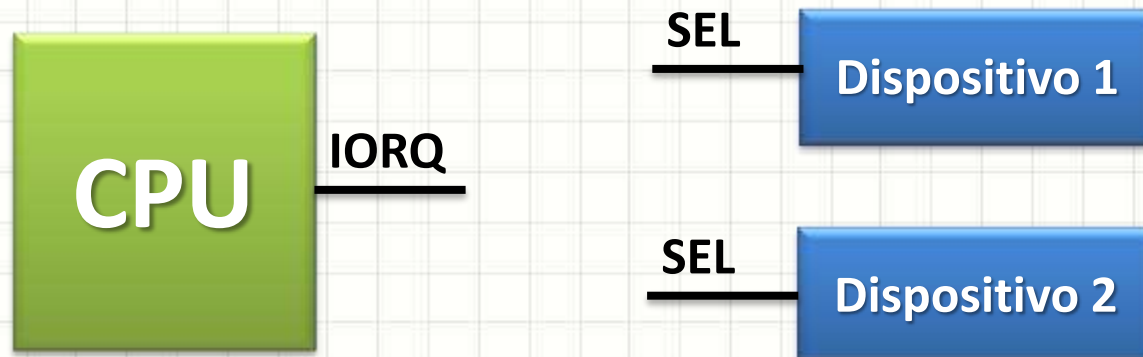


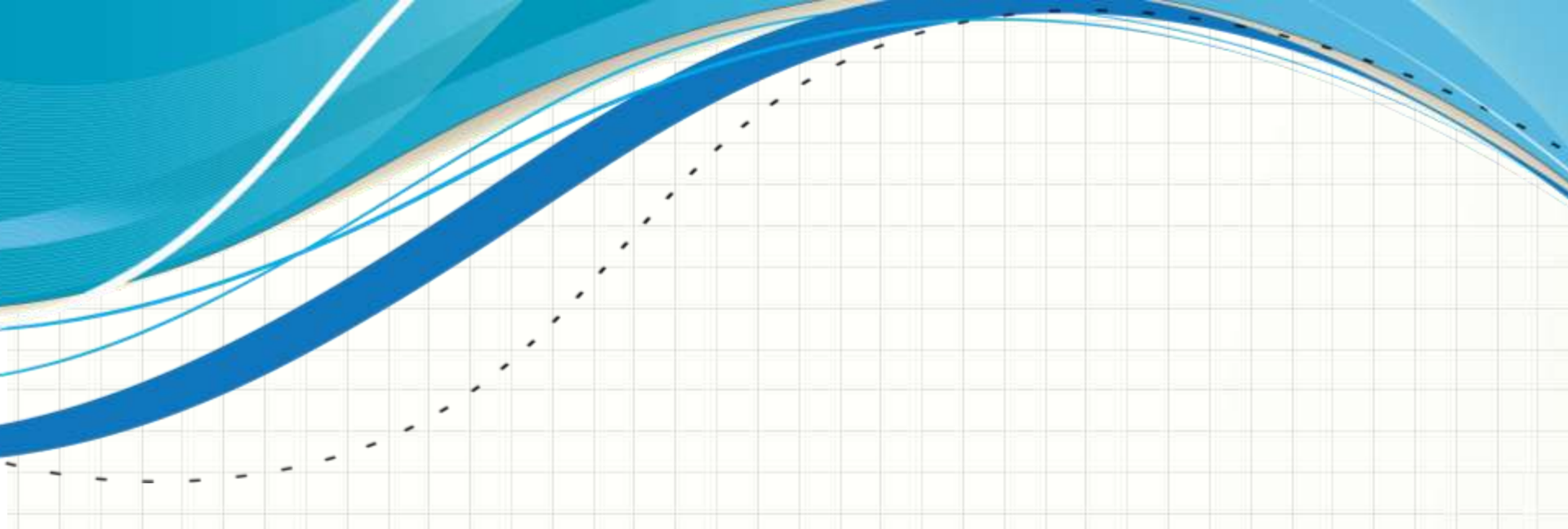
O que é um dispositivo?

- Usualmente pensam que são únicos

Como Resolver?

- CPU: apenas um pino de dispositivo: IORQ
 - Input/Output ReQuest





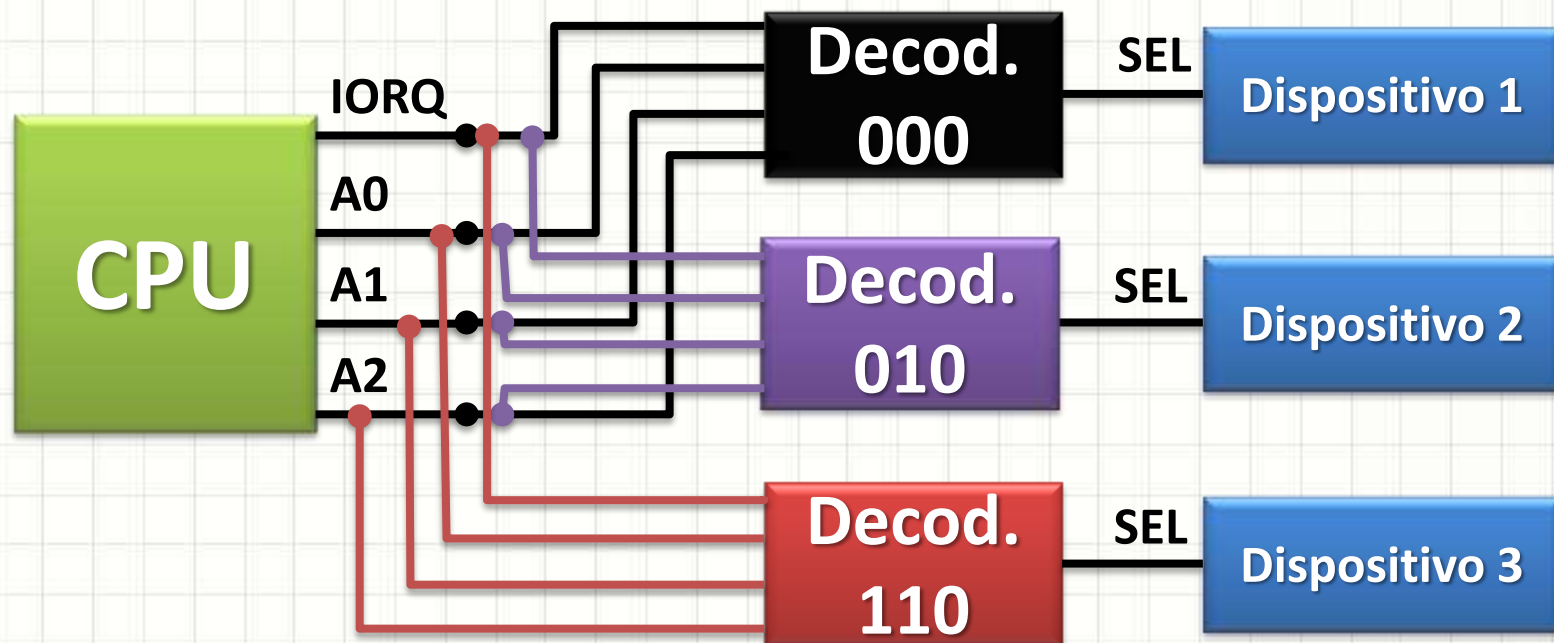
DECODIFICADOR DE ENDEREÇOS

Decodificador de Endereços

- endereços + IORQ → escolhe dispositivo
- Para isso funcionar:
 - Circuito que
 - Identifique um endereço
 - Acione o dispositivo correto
- Decodificador de Endereços

Decodificador de Endereços

- Memória: **MREQ** e A_0-A_n
- E/S: **IORQ** e A_0-A_n
- Observe como é feita a ligação:



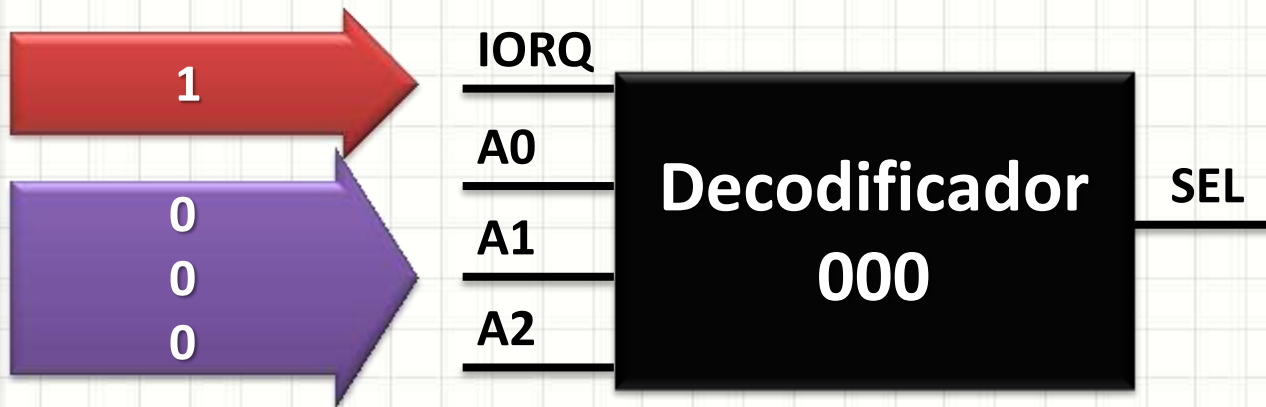
Decodificador de Endereços

- Como funciona o decodificador?



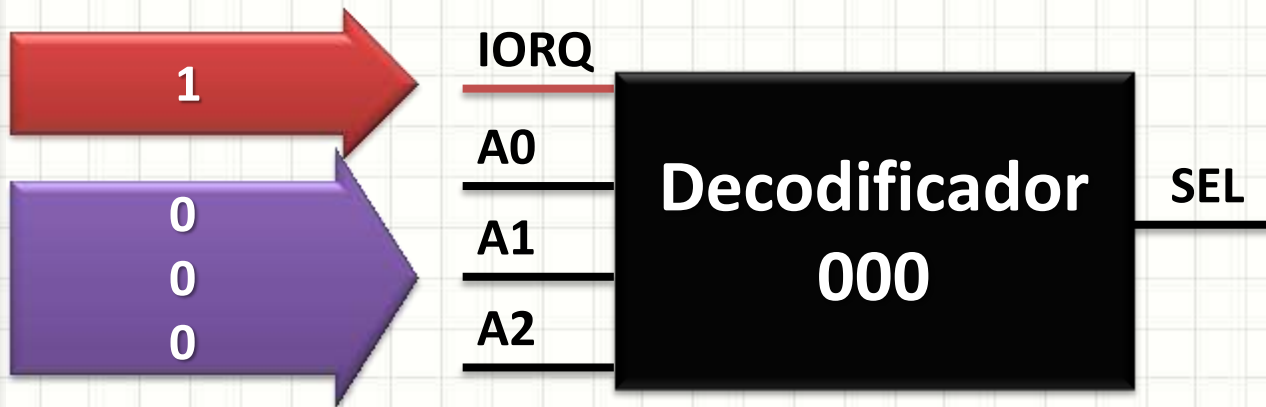
Decodificador de Endereços

- Como funciona o decodificador?
- Exemplo:



Decodificador de Endereços

- Como funciona o decodificador?
- Exemplo:



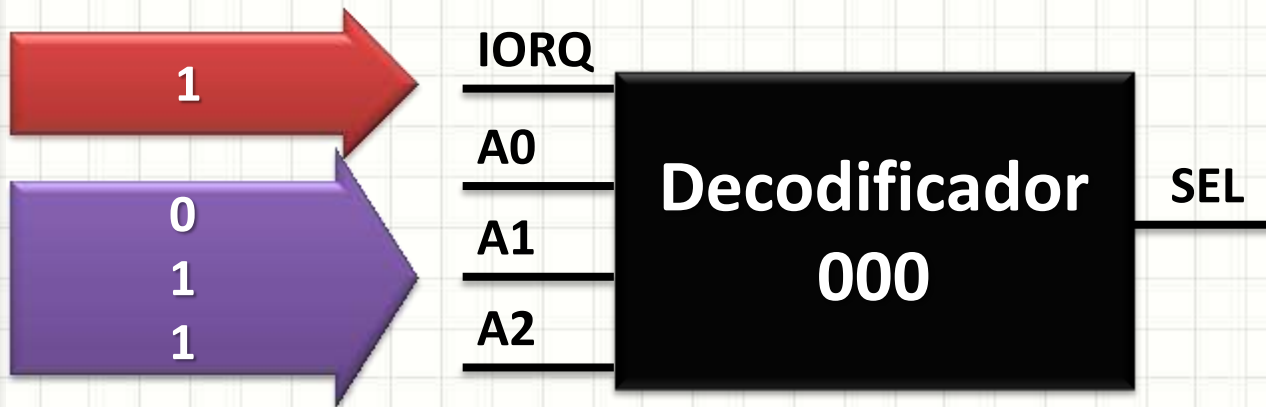
Decodificador de Endereços

- Como funciona o decodificador?
- Exemplo:



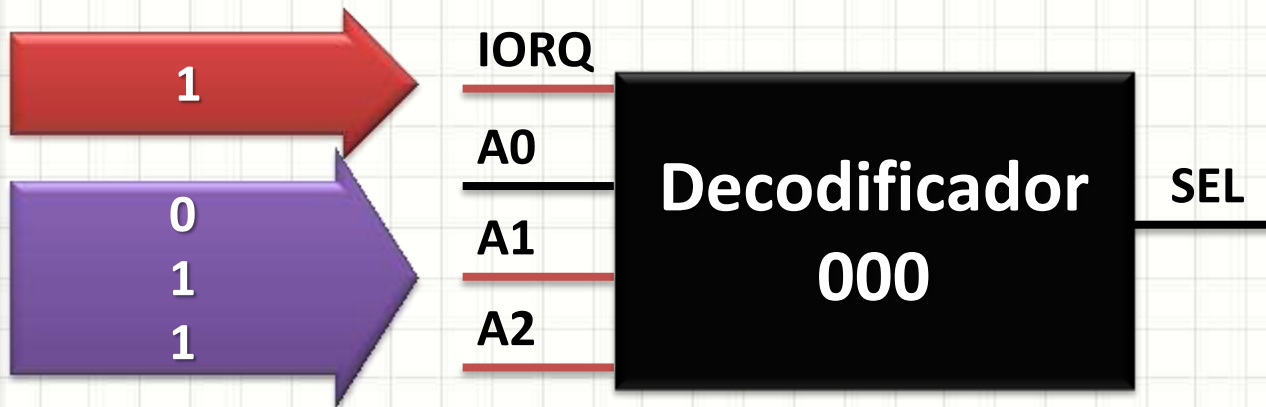
Decodificador de Endereços

- Como funciona o decodificador?
- Outro exemplo:



Decodificador de Endereços

- Como funciona o decodificador?
- Outro exemplo:



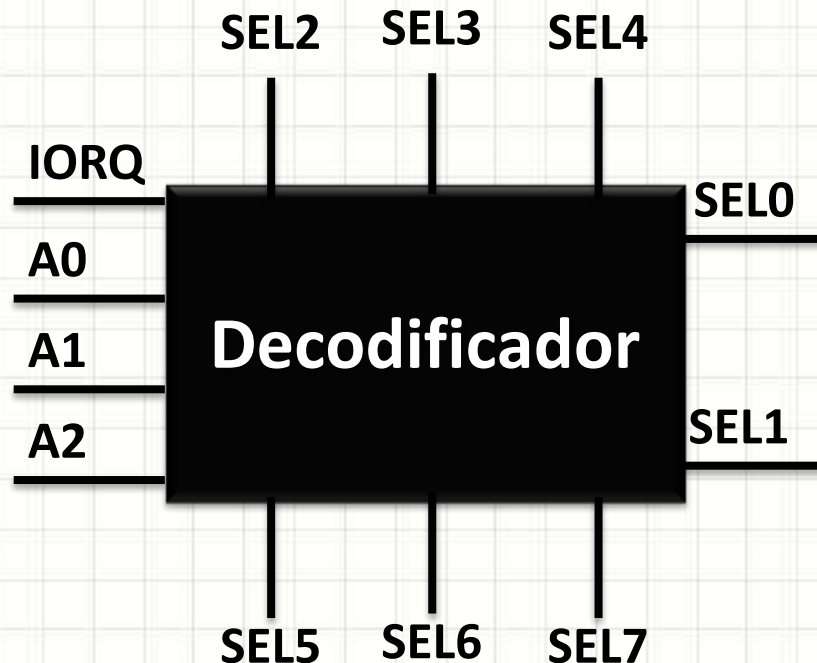
Decodificador de Endereços

- Como funciona o decodificador?
- Outro exemplo:



Decodificador de Endereços

- Decodificadores Múltiplos



Decodificador de Endereços

- Q
su

**3 bits =
8 dispositivos**

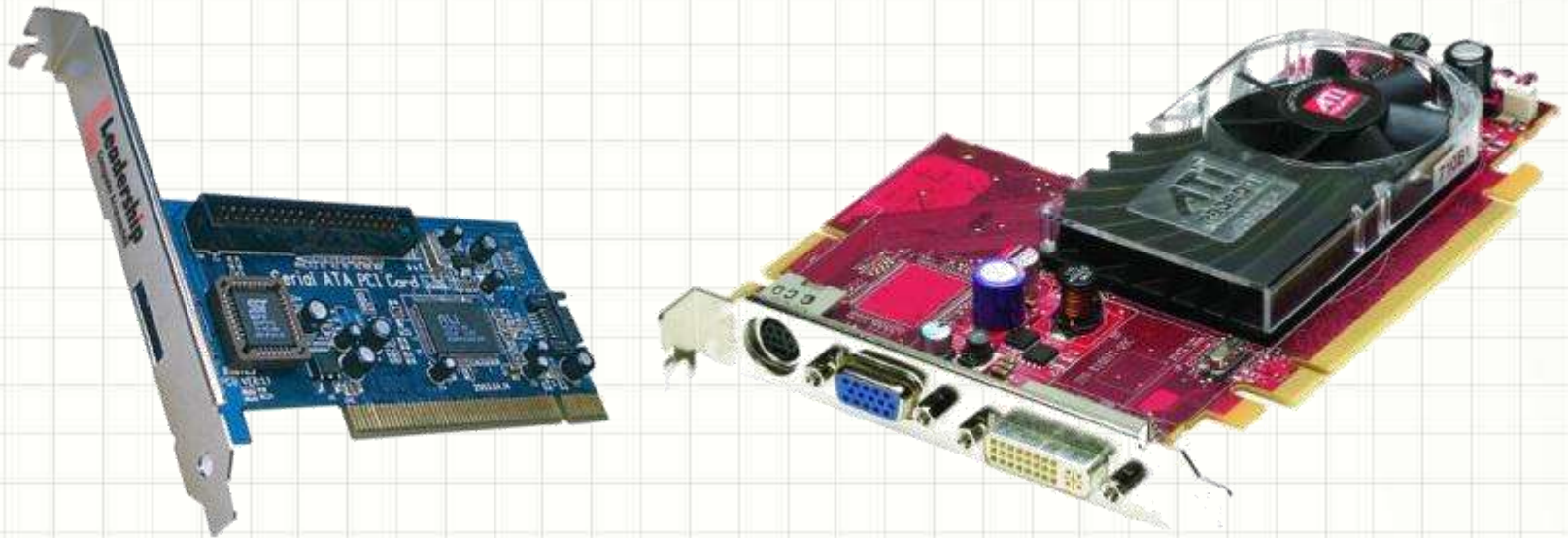


Decodificador de Endereços

- Decodificador: conjunto de circuitos lógicos
 - Cada endereço específico ativa um dispositivo
- “Endereço” de dispositivo: **porto** ou **porta**
- Dispositivos simples:
 - barramento de dados: direto
 - barramento de controle: direto
- E nos dispositivos mais complexos?

Controlador de Dispositivos

- Dispositivos mais complexos
 - Exigem mais que um decodificador
 - Exemplo: harddisk, vídeo...
- Existe uma **placa controladora** ou **interface controladora de dispositivo**



Controlador de Dispositivos

- O controle ainda é da CPU?
 - Claro!
- Existem três formas de controlar dispositivos
 - *Polling* (E/S Programada)
 - Interrupções
 - DMA (*Direct Memory Access*)



E/S POR *POLLING*

E/S por *Polling*

- *Polling* ou E/S Programada
 - *Poll*: consulta
 - A CPU precisa verificar constantemente os dispositivos
- Exemplo:
 - Placa de rede recebeu dado
 - CPU precisa lê-lo antes que o próximo chegue!
 - Impressora imprimiu uma letra
 - CPU precisa fornecer a próxima

E/S por *Polling*

- Analogia:
 - CPU: Cozinheiro e Garçon
 - Dispositivos: Clientes
- Se o cozinheiro é também o garçon
 - Para de cozinhar para atender os clientes
 - Para de cozinhar para levar pratos para os clientes
 - Pode perder clientes se demorar demais para atendê-los





E/S POR INTERRUPÇÃO

E/S por Interrupção

- Interrupção: Dispositivo **interrompe** CPU
 - Sempre que dispositivo precisa de algo
 - CPU não precisa fazer consulta
- A Interrupção é um “pino” da CPU
 - Barramento de Controle
 - IRQ: Interrupt ReQuest
- Exemplo:
 - Placa de rede recebeu dado
 - Placa de rede dispara interrupção da CPU
 - Acabaram os dados da placa de som
 - Placa de som dispara interrupção da CPU

E/S por Interrupção

- Analogia:
 - CPU: Cozinheiro e Garçon
 - Dispositivos: Clientes **com sineta**
- Com a sineta
 - Para de cozinhar para atender os clientes **só quando ouve a sineta**
 - Para de cozinhar para levar pratos para os clientes
 - Os clientes **tocam a sineta** quando precisam ser atendidos (não há perda de tempo do cozinheiro!)





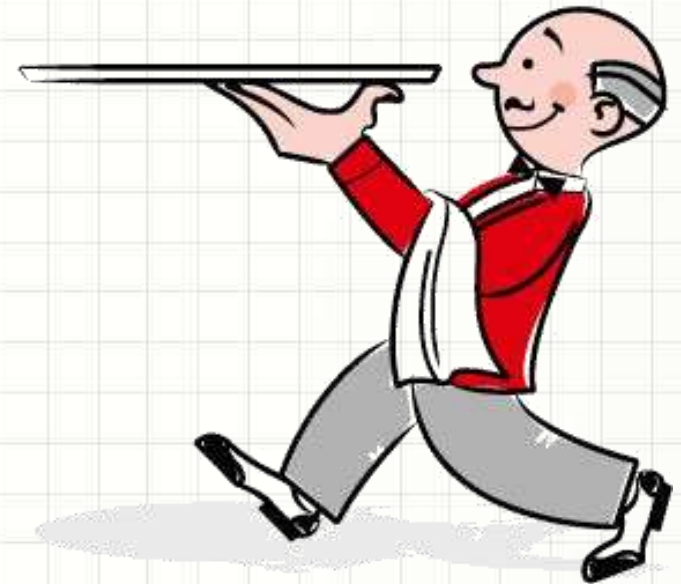
E/S POR DMA

E/S por DMA

- DMA: *Direct Memory Access*
 - Acesso Direto à Memória
 - CPU configura o DMA para transferir dados
 - CPU não precisa fazer a transferência
- DMA é um “dispositivo” adicional
 - Complemento ao *Polling* / IRQ
 - Apenas executa as transferências
- Exemplo:
 - Placa de rede recebeu um pacote de dados
 - CPU configura DMA para copiar esses dados na RAM
 - Acabaram os dados da placa de som
 - CPU configura DMA para transferir dados da RAM para placa de som

E/S por DMA

- Analogia:
 - CPU: Cozinheiro
 - DMA: Garçon
 - Dispositivos: Clientes
- Com o garçom
 - É o garçom que leva os pratos aos clientes, o cozinheiro **não para** de cozinhar



Comparativo

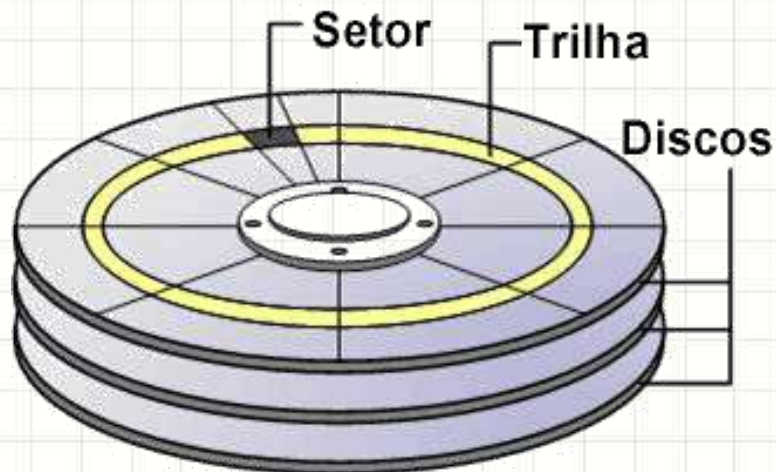
Descrição	Polling	Polling + DMA	IRQ	IRQ + DMA
Quem escolhe o momento de atender ao dispositivo?	CPU		Dispositivo	
Controle do Processamento	CPU		Compartilhado	
Pode ocorrer perda de dados?	Sim		Incomum	
Gasto de tempo de CPU com consulta aos dispositivos?	Sim		Não	
Gasto de tempo de CPU com transferência de dados?	Sim	Não	Sim	Não



DISPOSITIVOS COMUNS DE E/S

Hard Disk

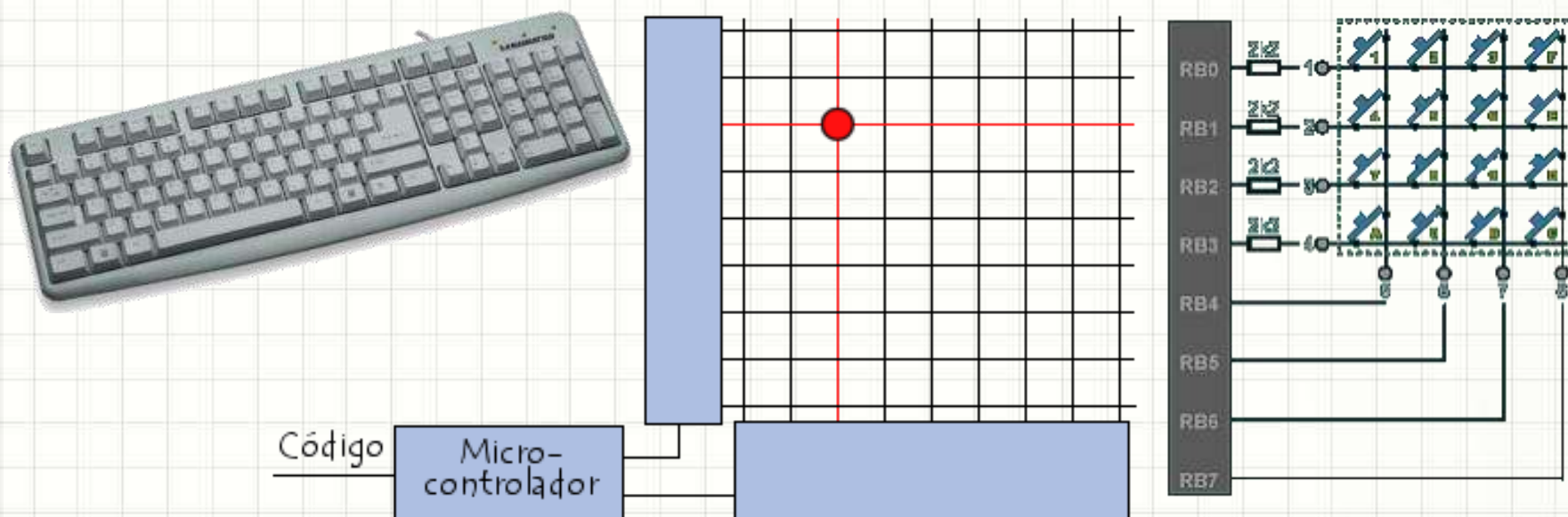
- Similar aos finados disquetes, mas...
 - Vários discos (de alumínio ou vidro)
 - Cobertos por Óxido de Ferro (magnetizado – 0, 1)
 - Fixos a um eixo, giram de 3.000 a 10.000 rpm
 - Regiões acessíveis: disco (head), trilha (cylinder), setor
 - Setores: usualmente 512 bytes



- 6 Cabeças
- 1 por vez
- Leitura: setor?
- Sistema de arquivos

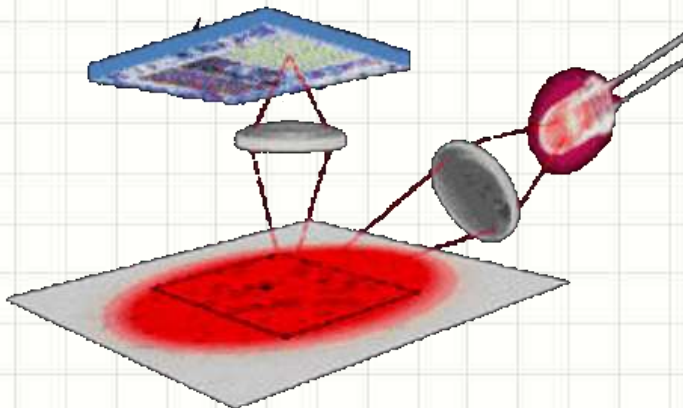
Teclados

- Matriz Eletrônica
 - Simplificado: A → Linha 04, coluna 02 → 0402
 - Teclas modificadoras enviadas separadamente
 - Mapa de Teclado: Linha/Coluna → Código
 - US-International, ABNT-2 etc.



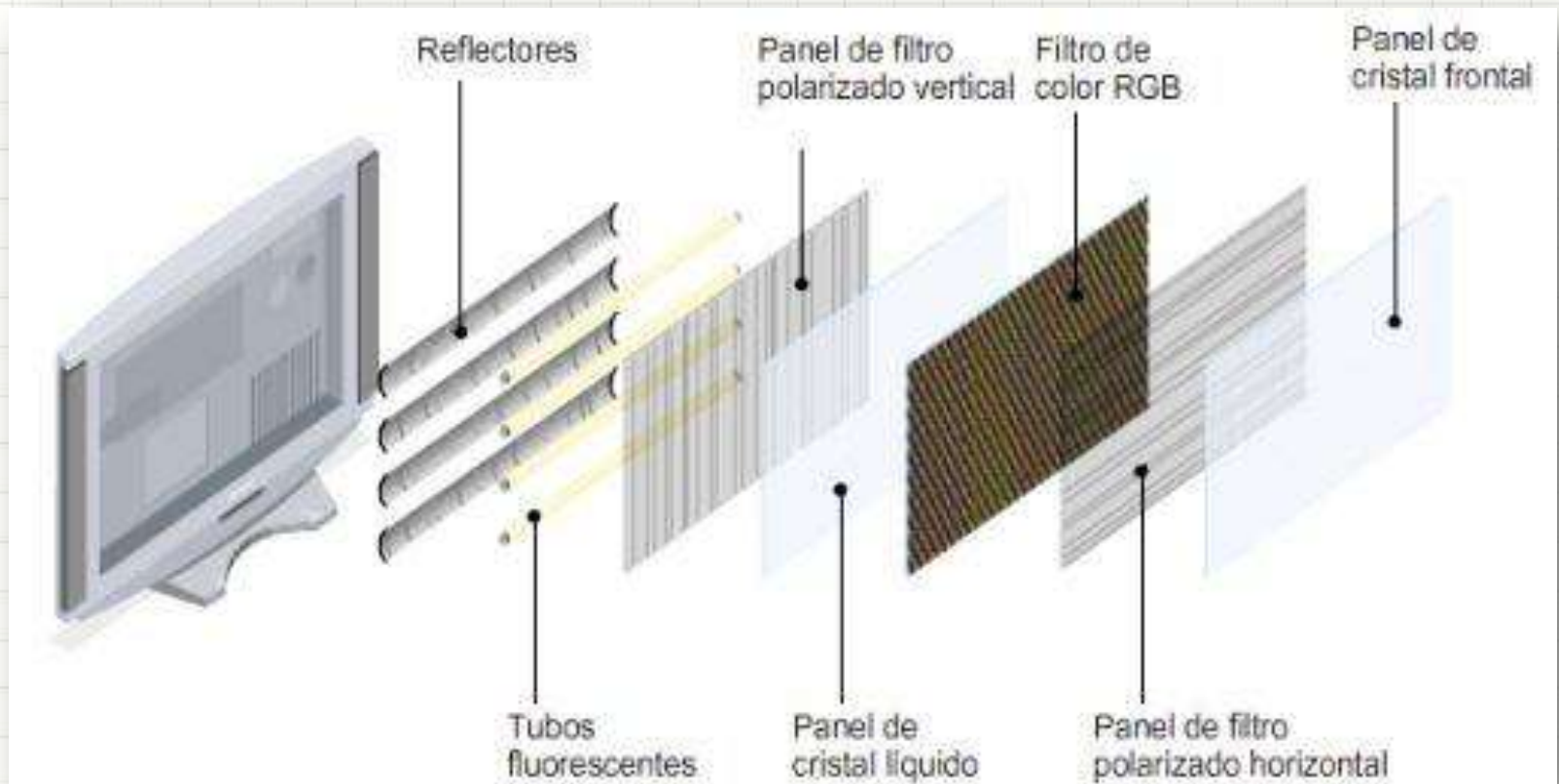
Mouse Óptico

- Fotos da Base de Apoio
 - Emite luz e digitaliza imagem: intervalos regulares
 - Detecção de deslocamento entre imagens
 - Codificação em um valor para o mouse



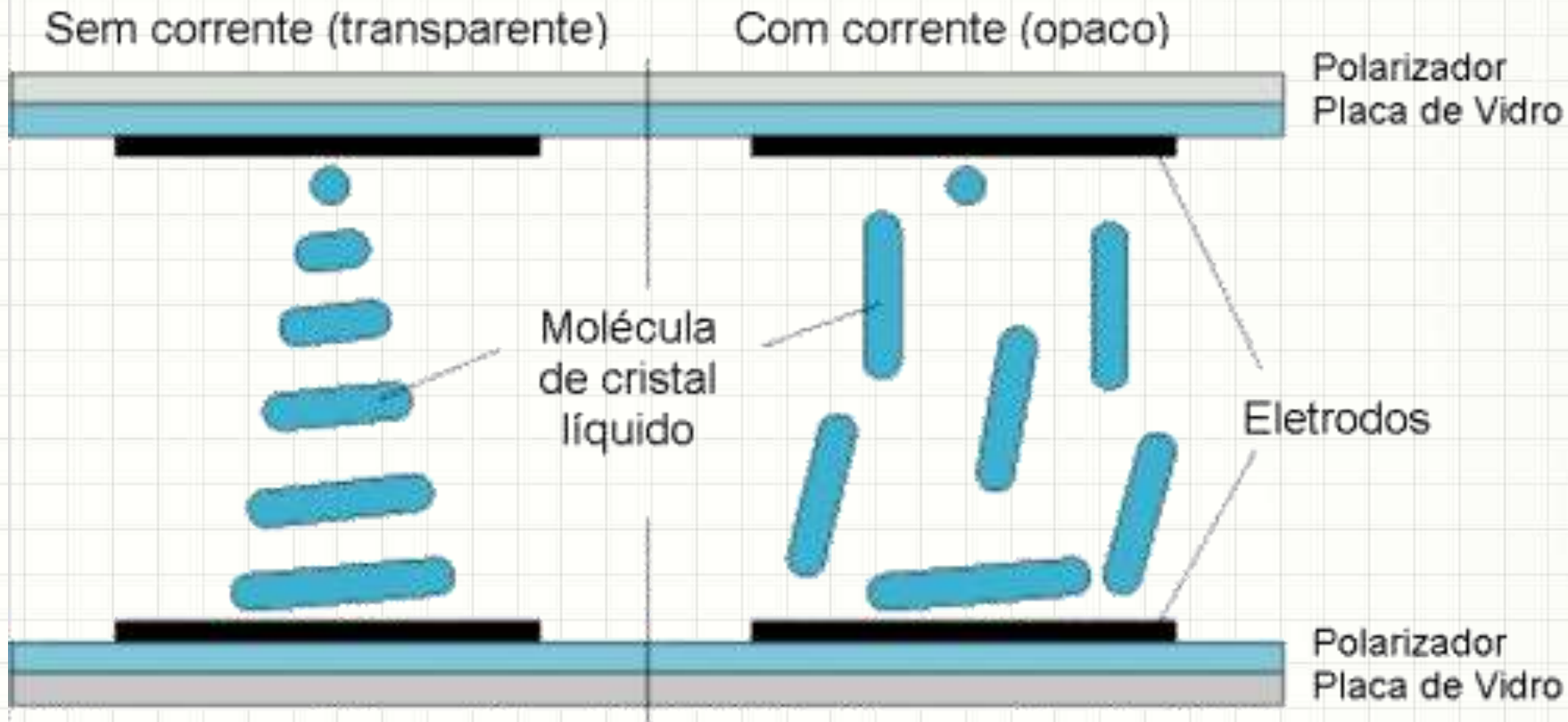
Monitores de Vídeo LCD

- Elementos Fundamentais:
 - emissor, polarizador vert., tela de cristal líquido, filtro plástico colorido, polarizador horizontal



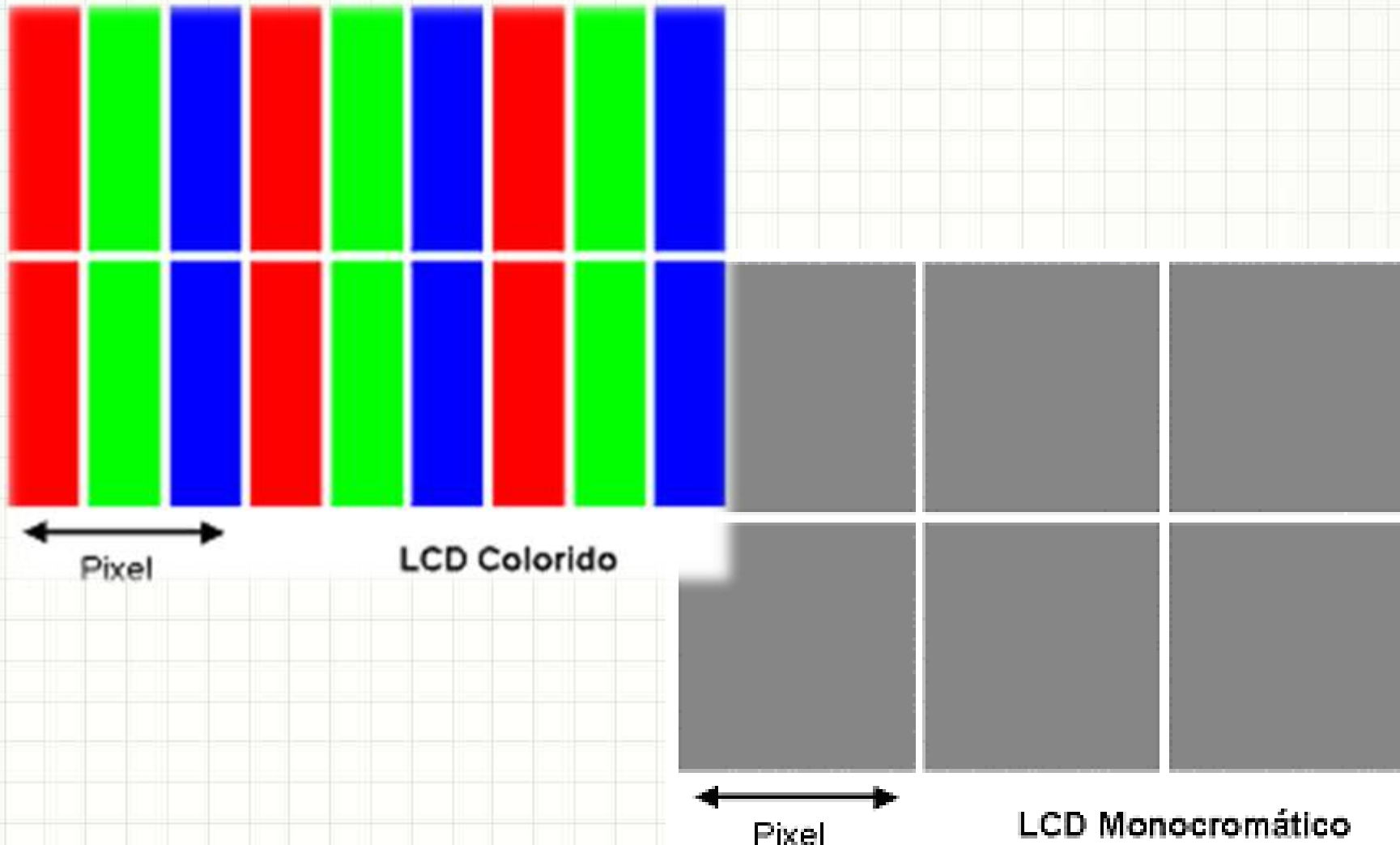
Monitores de Vídeo LCD

- Emissor, tela de cristal e polarizadores controlam quanta luz passa



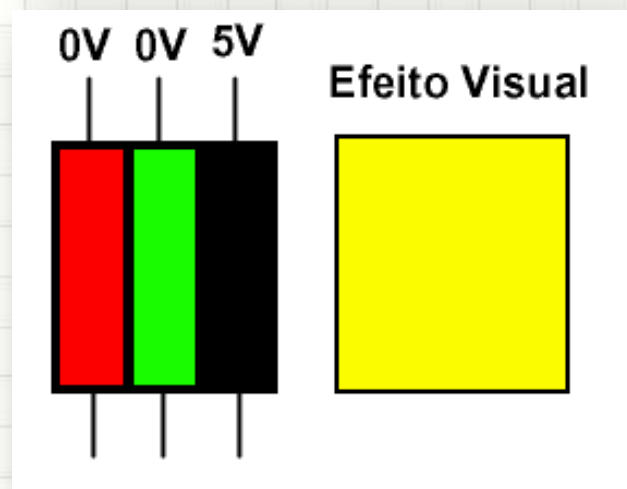
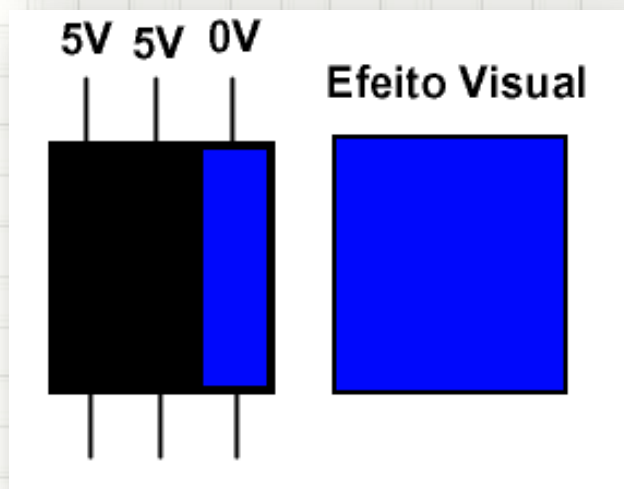
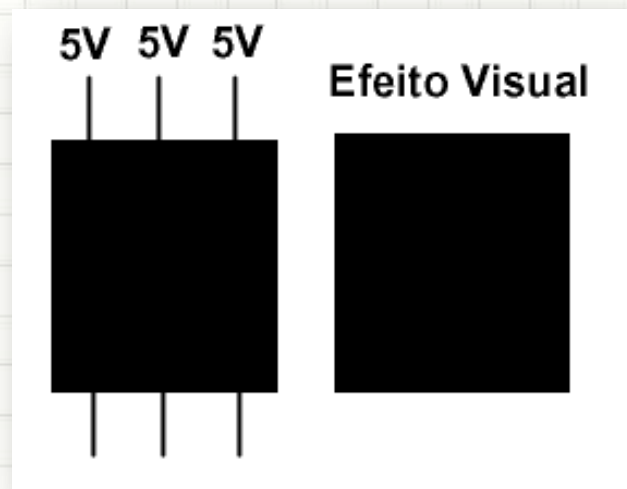
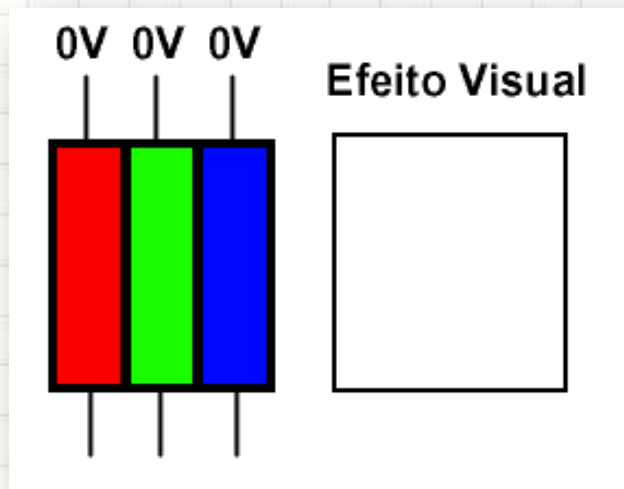
Monitores de Vídeo LCD

- O filtro RGB controla a cor de cada “subpixel”



Monitores de Vídeo LCD

- Configurações e Efeitos





CONCLUSÕES

Resumo

- Os dispositivos servem para converter sinais em informações e vice-versa
- O acesso a eles é feito pelo barramento, com auxílio do decodificador de endereços
- Dispositivos mais complexos exigem uma interface controladora
- Há diversas maneiras de gerenciar dispositivos
- Cada dispositivo lida com os sinais de uma maneira diferente
- **TAREFA**
 - AV1!

Próxima Aula



- Avaliação AV1!
- E depois?
 - Quem gerencia a memória?
 - E a CPU?
 - Sistemas Operacionais!
 - O que os SOs precisam?



PERGUNTAS?



**BOM DESCANSO
A TODOS!**