# Sistemas Operacionais

4º período

Professora: Michelle Hanne

# Nesta Aula

PUC Minas

- O papel dos controladores de E/S.
- Tipos de mapeamento de endereços dos dispositivos de E/S.
- Técnicas de acesso dos dispositivos de E/S.

# O que é Entrada e Saída

PUC Minas

- Processo pelo qual um processador se comunica com o mundo externo.
  - Um usuário humano.
  - Outro sistema computacional.
  - Sensores, atuadores...
- Comunicação pode se dar em dois sentidos:
  - Entrada: processador recebe dados do mundo externo.
  - Saída: processador envia dados para o mundo externo.
- Alguns dispositivos de E/S são **unidirecionais**, outros são **bidirecionais**.
- Também referenciado pela sigla em inglês I/O.
  - *Input/Output*.

Fonte: https://sites.google.com/a/ic.uff.br/fernanda/2017-2/fac

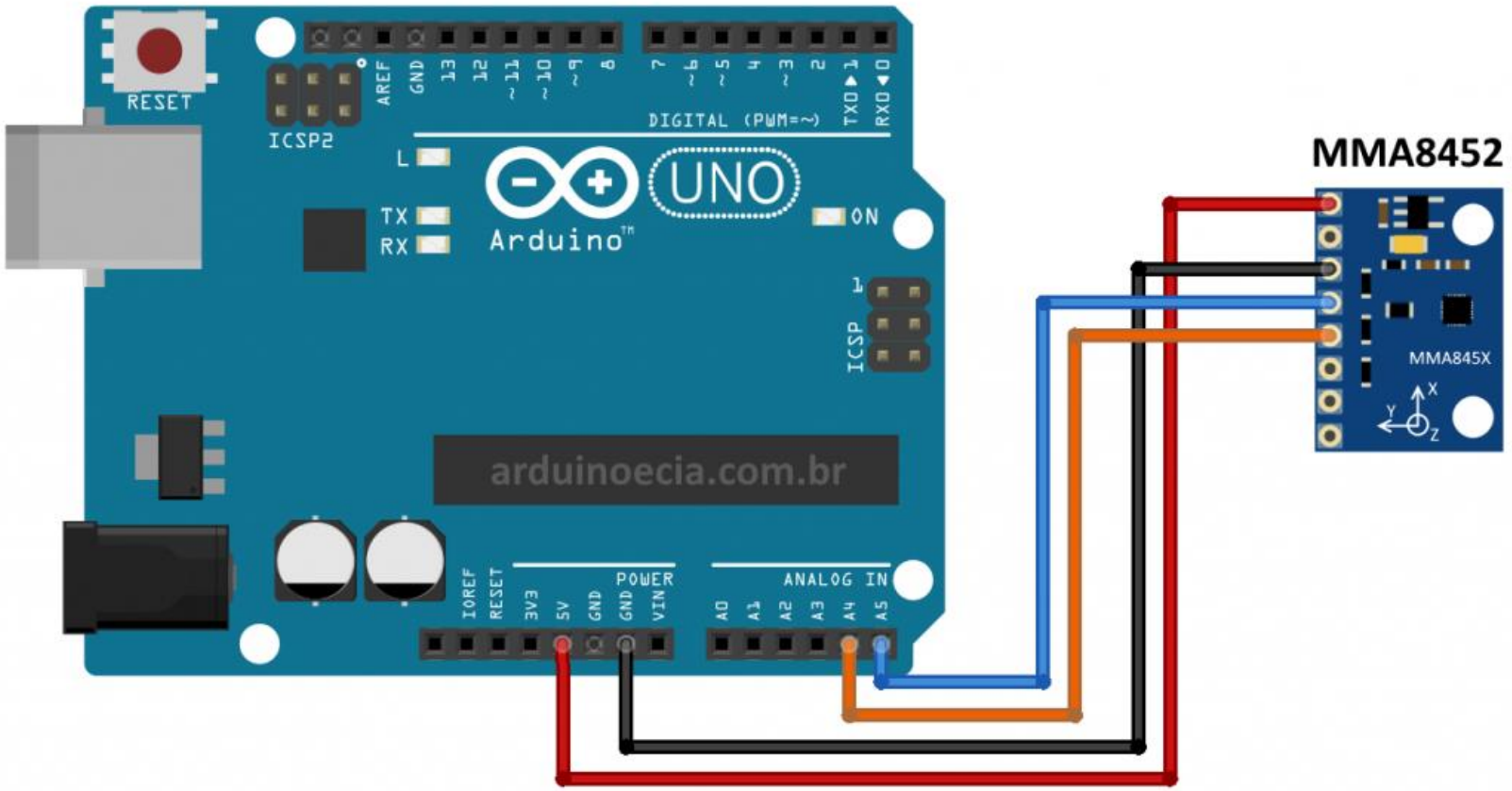
# Dispositivos de Entrada e Saída - Exemplos

PUC Minas

- Dispositivos de entrada:
  - Teclado.
  - Mouse.
  - Acelerômetro.
  - GPS.
  - Câmera.
  - Kinect.
- Dispositivos de saída:
  - Impressora.
  - Monitor.

Fonte https://sites.google.com/a/ic.uff.br/fernanda/2017-2/fac

# Exemplos - Acelerômetro

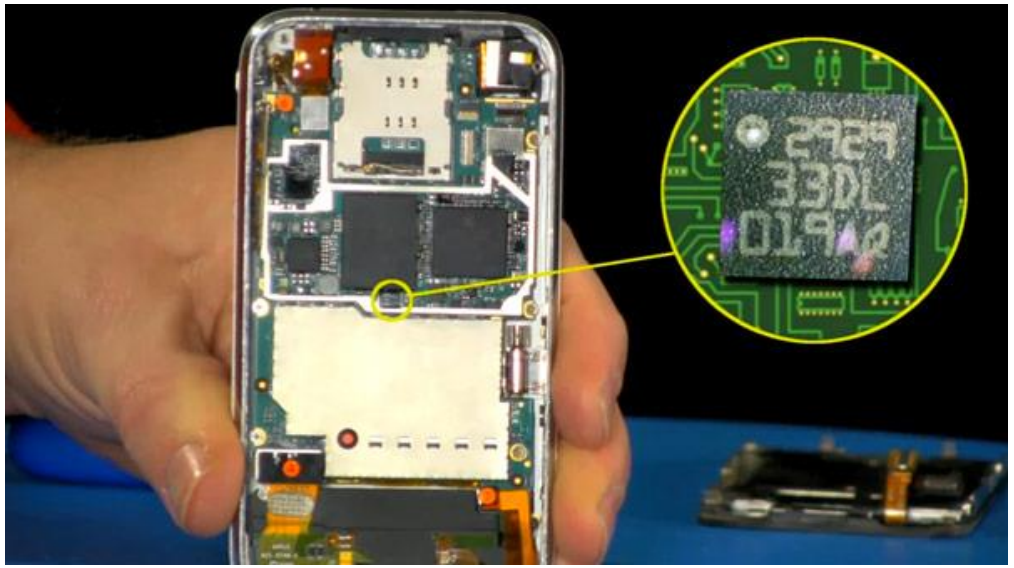



Conexão MMA8452 e Arduino Uno

Um acelerômetro é um microchip que serve para medir a aceleração de um corpo (normalmente o que está ligado nele) em relação à gravidade. Exemplo: MMA8452 com Arduino (**acelerômetro de 3 eixos)**

Fonte https://sites.google.com/a/ic.uff.br/fernanda/2017-2/fac

# Exemplos - Acelerômetro




Acelerômetro no celular

Sensores nos smartphones: Acelerômetros, GPS, Giroscópio, Magnômetro, sensores de luz e ambiente, etc.

Fonte https://sites.google.com/a/ic.uff.br/fernanda/2017-2/fac

# Exemplos - Acelerômetro

PUC Minas

**Mesmo com o GPS desligado é possível rastrear um Smartphone?**

Fonte https://sites.google.com/a/ic.uff.br/fernanda/2017-2/fac

# Exemplos - Acelerômetro

**Pesquisadores da Universidade de Princeton, Estados Unidos, descobriram que é possível rastrear o caminho percorrido pelo usuário mesmo que o sinal do GPS esteja desligado**. Eles afirmam que as linhas de código de um aplicativo são capazes de detectar informações a partir dos sensores do aparelho celular, como o acelerômetro, magnetômetro e o barômetro. Então, sem permissão do usuário é possível identificar, além de descobrir a localização aproximada, o endereço IP, fuso horário e tipo de conexão (dados móveis ou Wi-Fi).

Fonte https://sites.google.com/a/ic.uff.br/fernanda/2017-2/fac

# Dispositivos de Entrada e Saída

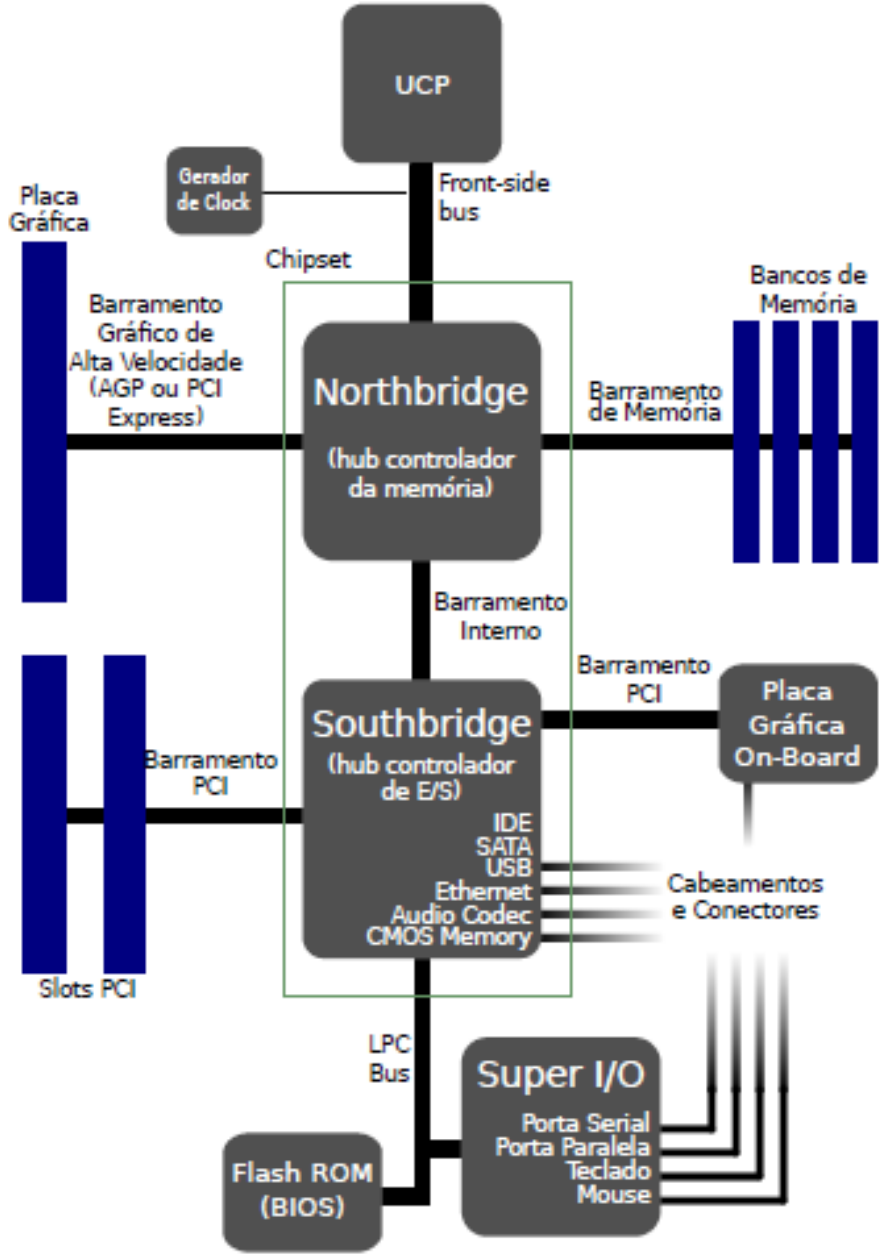
PUC Minas

- Dispositivos de entrada e saída:
  - Tela sensível ao toque.
  - Placa de som.
  - HDs, pendrives, ...
  - Interface de rede (Ethernet, Wifi, Bluetooth, ...).
  - Modem.
  - Porta serial.

Fonte https://sites.google.com/a/ic.uff.br/fernanda/2017-2/fac

# E/S e Barramentos

- Processador se comunica com outros dispositivos através de barramentos.
  - Meios de comunicação (potencialmente) compartilhados.
  - Vários dispositivos conectados ao mesmo barramento.
- Barramentos podem ser interligados a outros barramentos.
  - Através de pontes ou *bridges*.




Fonte https://sites.google.com/a/ic.uff.br/fernanda/2017-2/fac

# Disco Rígido – Dispositivo de E/S

PUC Minas

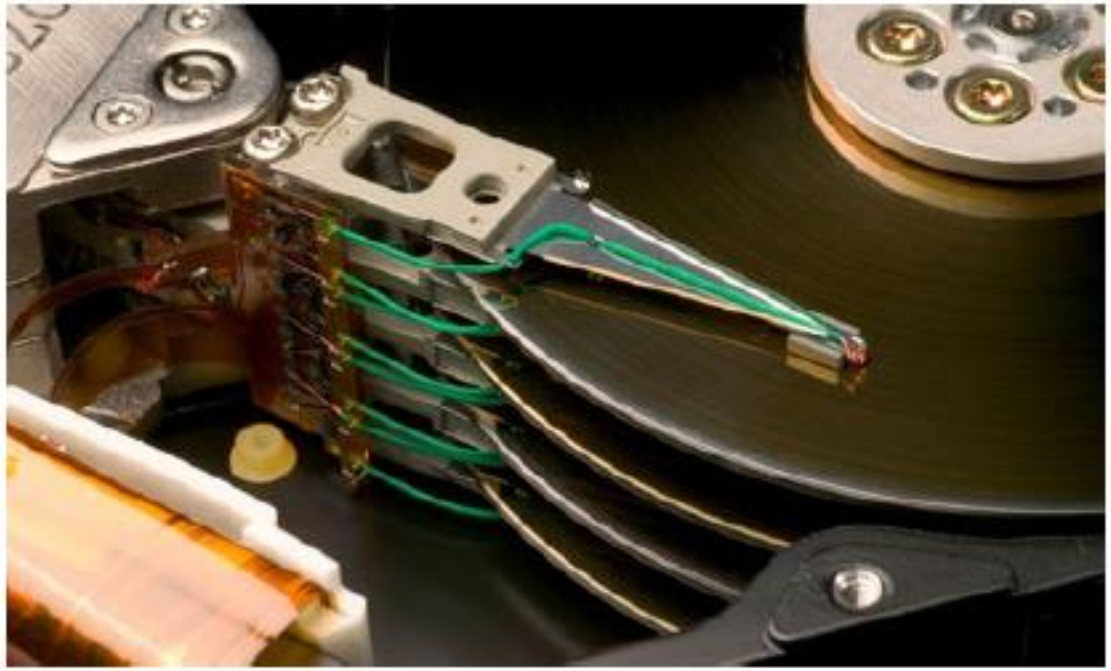
- Um dos dispositivos de E/S mais comuns em computadores modernos.
  - Embora comece a ser substituído por outras tecnologias, como SSD.
- Importante por também atuar como um tipo de memória.
  - Memória secundária, na hierarquia.
  - Nível mais baixo.
  - Grande capacidade, relativamente barato, mas lento.
- Também chamado de HD (do inglês *Hard Drive*).
  - Em oposição aos discos magnéticos flexíveis, como disquetes.

Fonte https://sites.google.com/a/ic.uff.br/fernanda/2017-2/fac

# Disco Rígido – Dispositivo de E/S

PUC Minas

- HDs utilizam uma mídia que pode ser magnetizada.
- Informação é armazenada alterando a magnetização de parte da mídia.
- A mídia é composta por um ou mais discos concêntricos.
  - Chamados de **pratos**.
  - Por sua vez, podem ter duas superfícies (de baixo e de cima).
- HD possui uma cabeça de leitura/gravação que se posiciona sobre regiões dos discos.

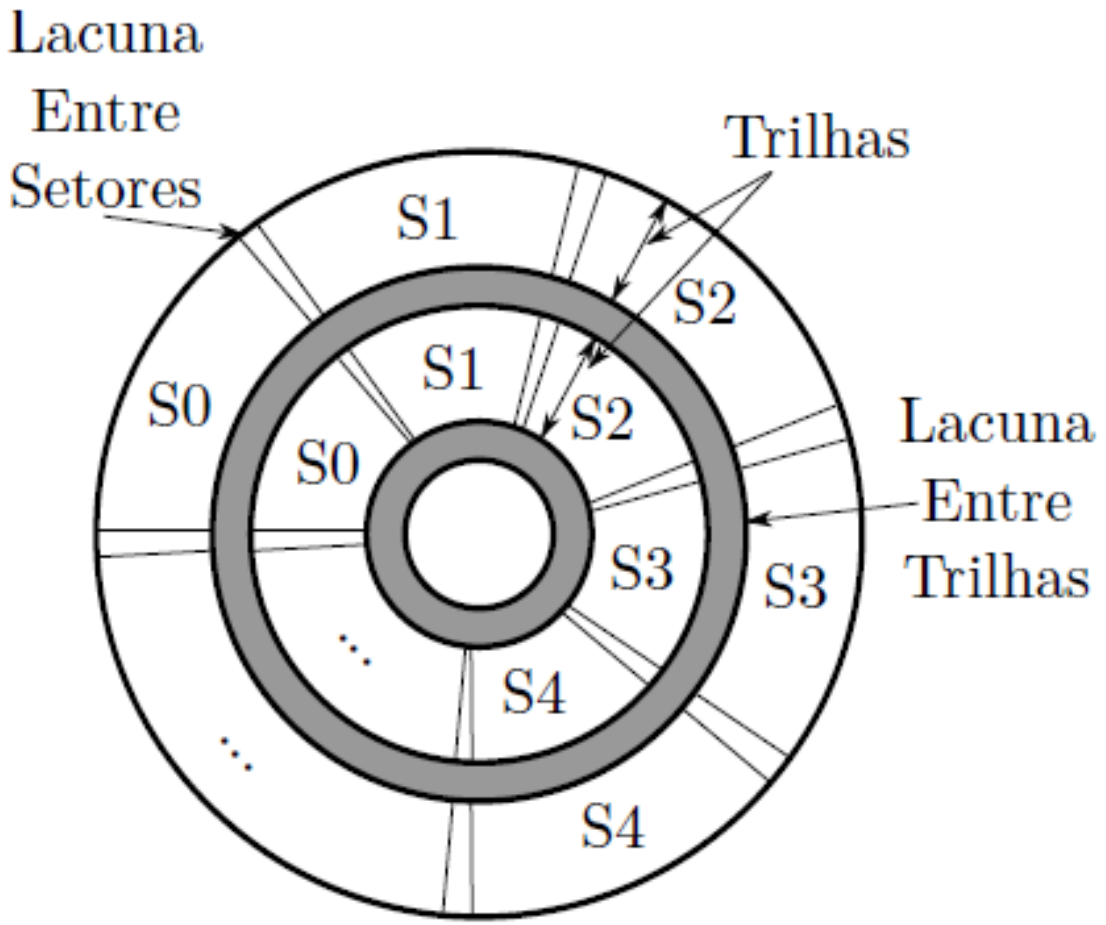


Fonte https://sites.google.com/a/ic.uff.br/fernanda/2017-2/fac

# Disco Rígido – Layout da Superfície de um prato

PUC Minas

- Cada prato de um HD é dividido em **trilhas**.
  - "Anéis" concêntricos.
- Trilhas são subdivididas em **setores**.
  - Menor unidade que pode ser lida ou escrita.
- Entre duas trilhas ou dois setores consecutivos, há uma **lacuna**.

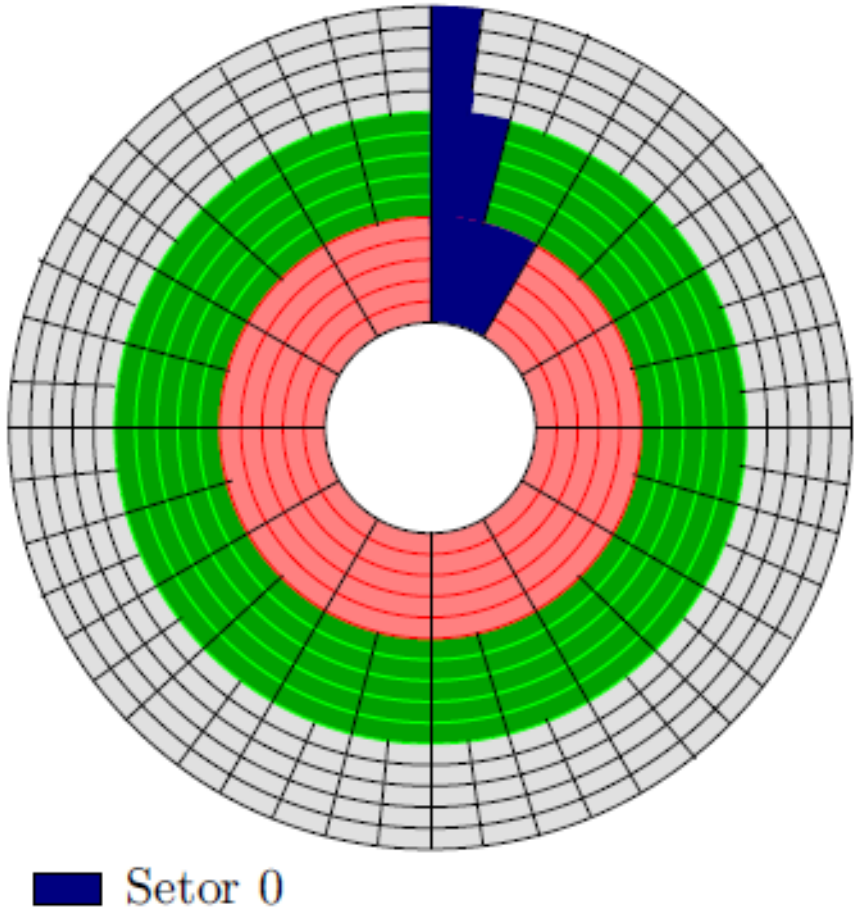



Fonte https://sites.google.com/a/ic.uff.br/fernanda/2017-2/fac

# Disco Rígido – Trilhas e Setores

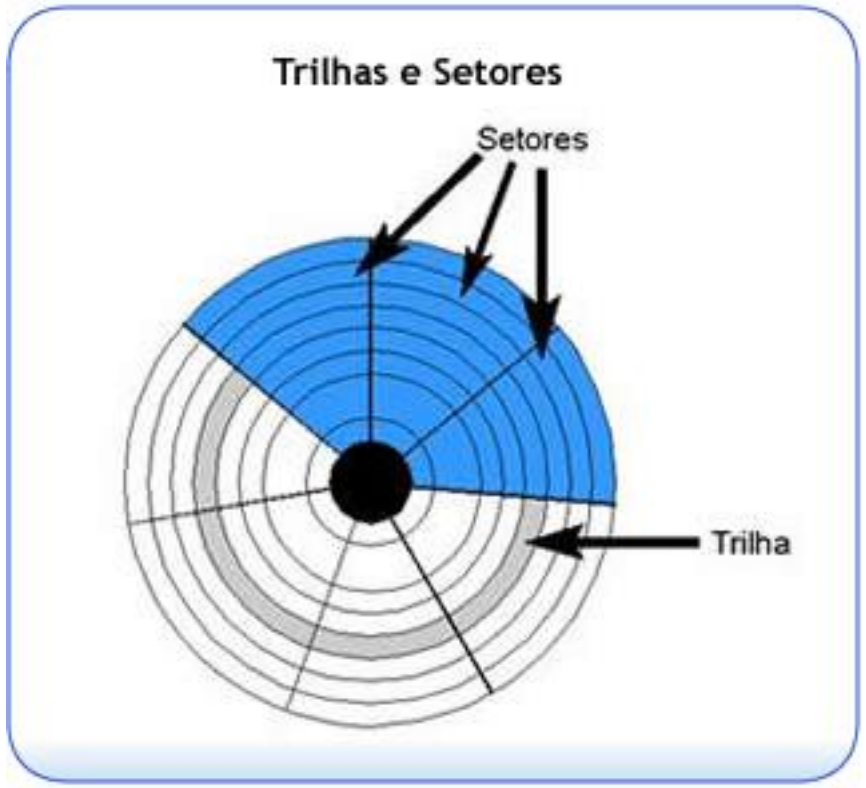
PUC Minas

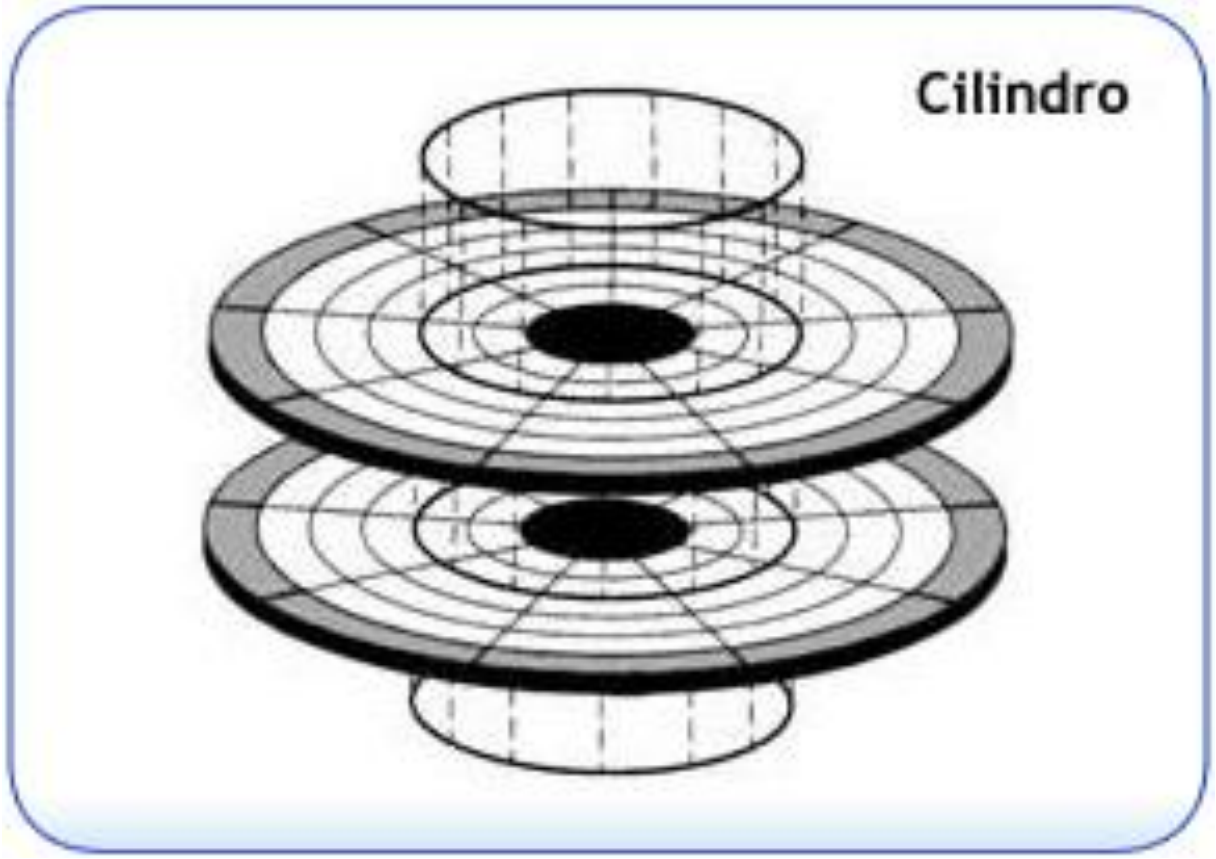
- HDs modernos (*i.e.*, desde a década de 1990), utilizam o método de **gravação em múltiplas zonas**.
  - Gravação nas trilhas mais externas é mais rápida.
    - ★ Mais bits por segundo.
  - Faz com que o número de bits por trilha seja maior nas trilhas mais externas.
  - Mantém a densidade de bits (quase) constante.
- Resultado:
  - Trilhas externas têm mais setores.




Fonte https://sites.google.com/a/ic.uff.br/fernanda/2017-2/fac

# Disco Rígido – Trilhas e Setores




Fonte https://sites.google.com/a/ic.uff.br/fernanda/2017-2/fac

# Disco Rígido – Modos de Endereçamento

PUC Minas

- O setor é a unidade básica de acesso de um HD.
  - Tanto para leitura, quanto para escrita.
  - Não podemos ler/escrever menos que um setor completo.
- Logo, o endereçamento é feito através da identificação de setores.
- Duas formas principais:
  - *Cylinder-Head-Sector*, ou CHS.
    - ★ Setor é identificado como uma tupla com três componentes.
  - *Logical Block Addressing*, ou LBA.
    - ★ Cada setor é mapeado para um identificador numérico único.

Fonte https://sites.google.com/a/ic.uff.br/fernanda/2017-2/fac

# Disco Rígido – Tempo de Acesso

PUC Minas

- Quanto tempo demora a leitura de um dado de um HD?
- Há várias etapas neste processo:
  - Posicionamento da cabeça de leitura sobre a trilha correta.
  - Espera pelo setor correto (rotação do prato).
  - Leitura em si.
- Cada uma destas etapas corresponde a uma parcela do tempo total de leitura:
  - Tempo de busca (*seek*): posicionar o braço sobre a trilha desejada.
  - Latência rotacional (*latency*): espera para que o setor desejado esteja sob a cabeça.
  - Tempo de transferência (*transfer*): transferir um bloco de bits (setor desejado).

Fonte https://sites.google.com/a/ic.uff.br/fernanda/2017-2/fac

# Disco Rígido – Tempo de Busca

PUC Minas

- Depende do quão rapidamente o HD pode mover a cabeça de leitura.
- Também depende da posição inicial da cabeça.
  - Se ele já está próxima da trilha desejada, tempo é curto.
- HDs típicos têm tempo médio de busca na casa de 10 ms.
- Mas note que o princípio de localidade espacial pode se aplicar ao HD.
  - Leituras/escritas sucessivas podem ser feitas à trilhas próximas.
  - Reduz o tempo efetivo de busca.
- Sistemas operacionais usam vários artifícios para tentar melhorar esta localidade de acessos.
  - Algoritmos de escalonamento de acessos.
  - Desfragmentação do sistema de arquivos.

Fonte https://sites.google.com/a/ic.uff.br/fernanda/2017-2/fac

# Disco Rígido – Tempo de Latência

PUC Minas

- Uma vez que a cabeça seja posicionada na trilha, é preciso "esperar" que o setor certo seja alcançado.
  - Através da rotação do prato.
- Se dermos sorte, setor já estará sob a cabeça.
- Se dermos azar, o setor acabou de passar.
  - Precisaremos esperar uma revolução completa do prato.
- Na média, precisaremos de metade de uma revolução.
- Velocidades típicas de rotação de pratos vão de 3600 a 7200 RPM.
  - Revolução completa demora de 8 ms a 16 ms.
  - Metade corresponde de 4 ms a 8 ms.

Fonte https://sites.google.com/a/ic.uff.br/fernanda/2017-2/fac

# Disco Rígido – Tempo de Transferência

PUC Minas

- Depende de uma série de fatores:
  - Velocidade de rotação.
  - Densidade de bits.
  - Tamanho do dado.
- HDs modernos conseguem taxas de transferência na casa de 100 MB/s.

**Como Funciona o HD**: https://www.youtube.com/watch?v=lpYfep68xnA

**Como Funciona o SSD**: https://www.youtube.com/watch?v=GcwNqnUaFpo

PUC Minas

# Barramentos de E/S

# Barramentos de E/S

PUC Minas

Computadores modernos possuem múltiplos barramentos.

Para interconectar dispositivos de características diferentes ao processador.

Dispositivos de E/S tendem a ser muito mais lentos que a MP.

Usar um único barramento seria prejudicial à MP.

Fonte https://sites.google.com/a/ic.uff.br/fernanda/2017-2/fac

# Barramentos de E/S são mais lentos

PUC Minas

## Uma série de razões:

- **Velocidade dos dispositivos de E/S.**
  - ✓Normalmente lentos em relação ao processador e à MP.
- **Distância dos barramentos de E/S.**
  - ✓Normalmente longa em relação ao barramento de conexão da CPU com a MP.
- **Número de dispositivos.**
  - ✓Barramentos são compartilhados.
  - ✓Capacidade é dividida entre os dispositivos.
  - ✓Número de dispositivos de E/S é relativamente grande.

Fonte https://sites.google.com/a/ic.uff.br/fernanda/2017-2/fac

# Barramentos E/S

PUC Minas

## Barramento ATA

ATA – Advanced Technology Attachment – mais comumente chamado de IDE (Integrated Drive Electronics), O ATA é um padrão utilizado para interligar dispositivos de armazenamento como discos rígidos, drives, CD-ROMS e outros periféricos semelhantes.

Fonte https://sites.google.com/a/ic.uff.br/fernanda/2017-2/fac

# Barramentos E/S

PUC Minas

## Barramento SATA

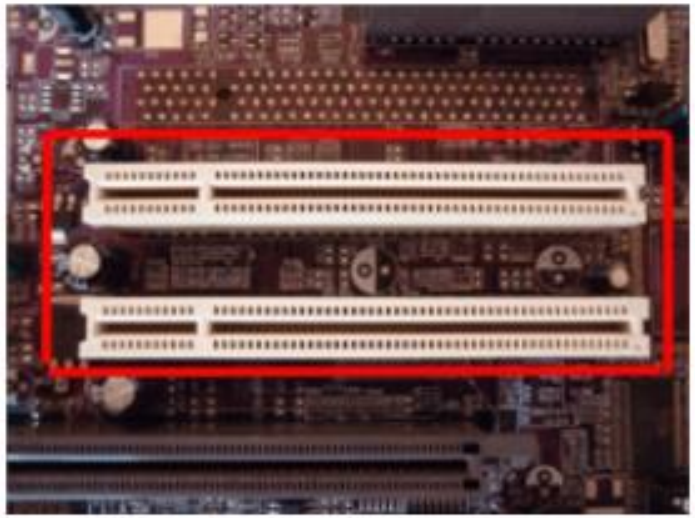

Serial ATA – SATA (Serial Advanced Technology Attachment) – padrão substituto da tecnologia ATA. Ao contrário da tecnologia ATA, onde as informações eram transmitidas de forma paralela (vários bits por vez), na SATA as informações passaram a ser transmitidas de forma serial, com um bit atrás do outro, o que resolveu problemas de perda de dados ocasionadas por interferências.

Fonte https://sites.google.com/a/ic.uff.br/fernanda/2017-2/fac

# Barramentos E/S

PUC Minas

## Barramento PCI



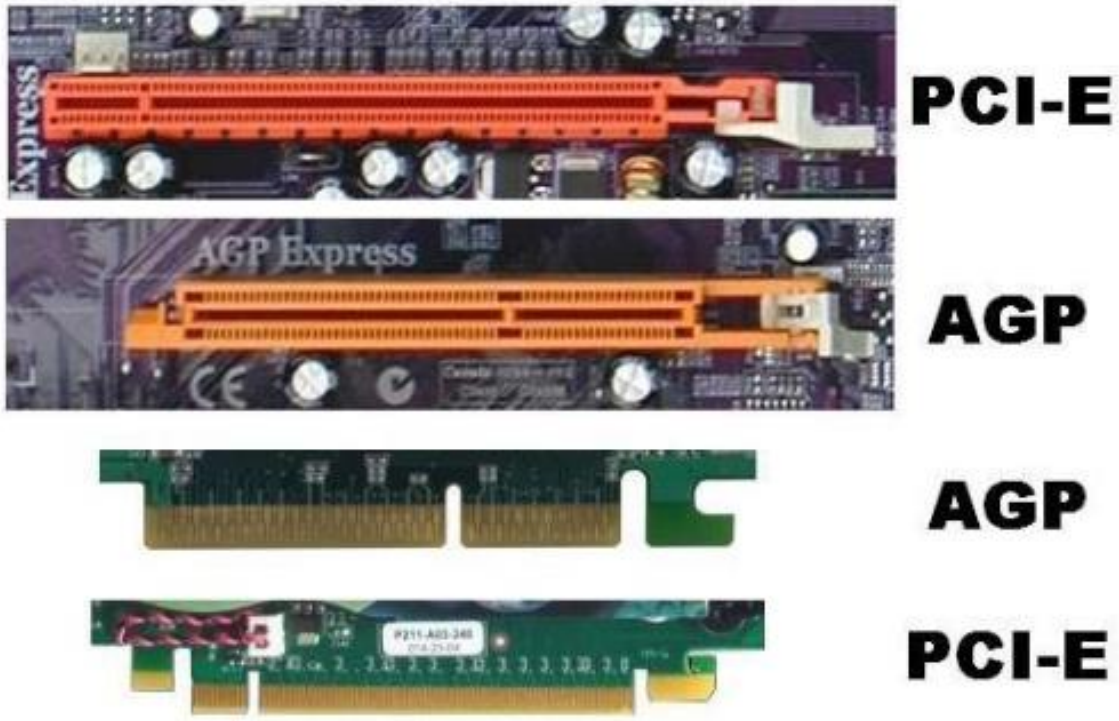
PCI é um padrão de barramentos, destinado a conectar periféricos à placa mãe do computador. Mas você também pode encontrar outras referências a ele, como “interface”, “slot” ou “soquete”. Ele foi criado em 1993 para substituir alguns padrões ancestrais, como AGP, VESA e USB (sim, lá atrás, a interface era usada na placa mãe).

Fonte https://sites.google.com/a/ic.uff.br/fernanda/2017-2/fac

# Barramentos E/S

PUC Minas

## Barramento AGP



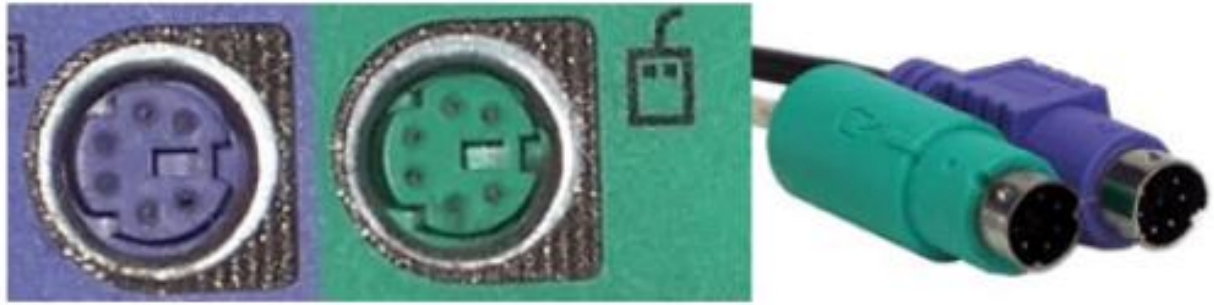

Para lidar com o volume crescente de dados gerados pelos processadores gráficos, a Intel anunciou em meados de 1996 o padrão AGP, cujo slot serve exclusivamente às placas de vídeo.

Fonte https://sites.google.com/a/ic.uff.br/fernanda/2017-2/fac

# Barramentos E/S

PUC Minas

## Barramento PS/2



Os conectores PS2 são usados até hoje em PCs modernos desafiando a praticidade do USB. Placa-mães, mouses e teclados usam esta interface por ocuparem menos espaço e liberar as conexões USB para periféricos.

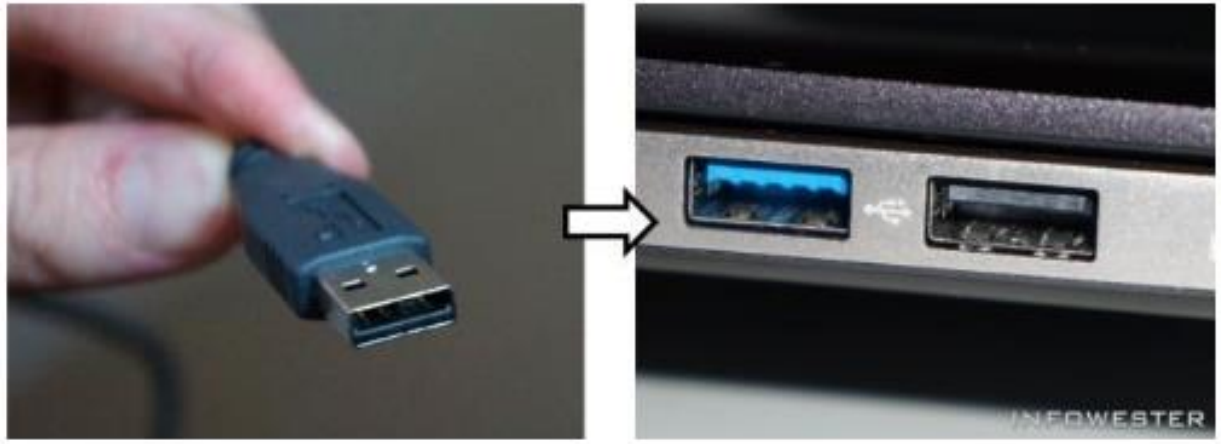
Em 1997 foram definidas cores para os padrões de entrada/saída dos PCs e os conectores PS2 ganharam as cores lilás para teclado e verde para mouse.

Fonte https://sites.google.com/a/ic.uff.br/fernanda/2017-2/fac

# Barramentos E/S

PUC Minas

## Barramento USB




USB, do inglês Universal Serial Bus, é um barramento externo que facilitou e universalizou a conexão de aparelhos e periféricos (Câmeras digitais, mouses, teclados) ao computador e a outros dispositivos.

A tecnologia "Plug and Play" (Plugar e utilizar), quase livre de instalação ou de reinicialização do sistema com o sistema operacional reconhecendo automaticamente a conexão do dispositivo, tornou muito mais fácil a instalação de periféricos, que antes necessitavam de configurações muito complexas.

- É capaz de fornecer eletricidade
- Hot Swap: permite conexão e desconexão sem a necessidade de reiniciar o computador
- Suporta até 127 dispositivos conectados na mesma saída por meio de hubs

Fonte https://sites.google.com/a/ic.uff.br/fernanda/2017-2/fac

PUC Minas

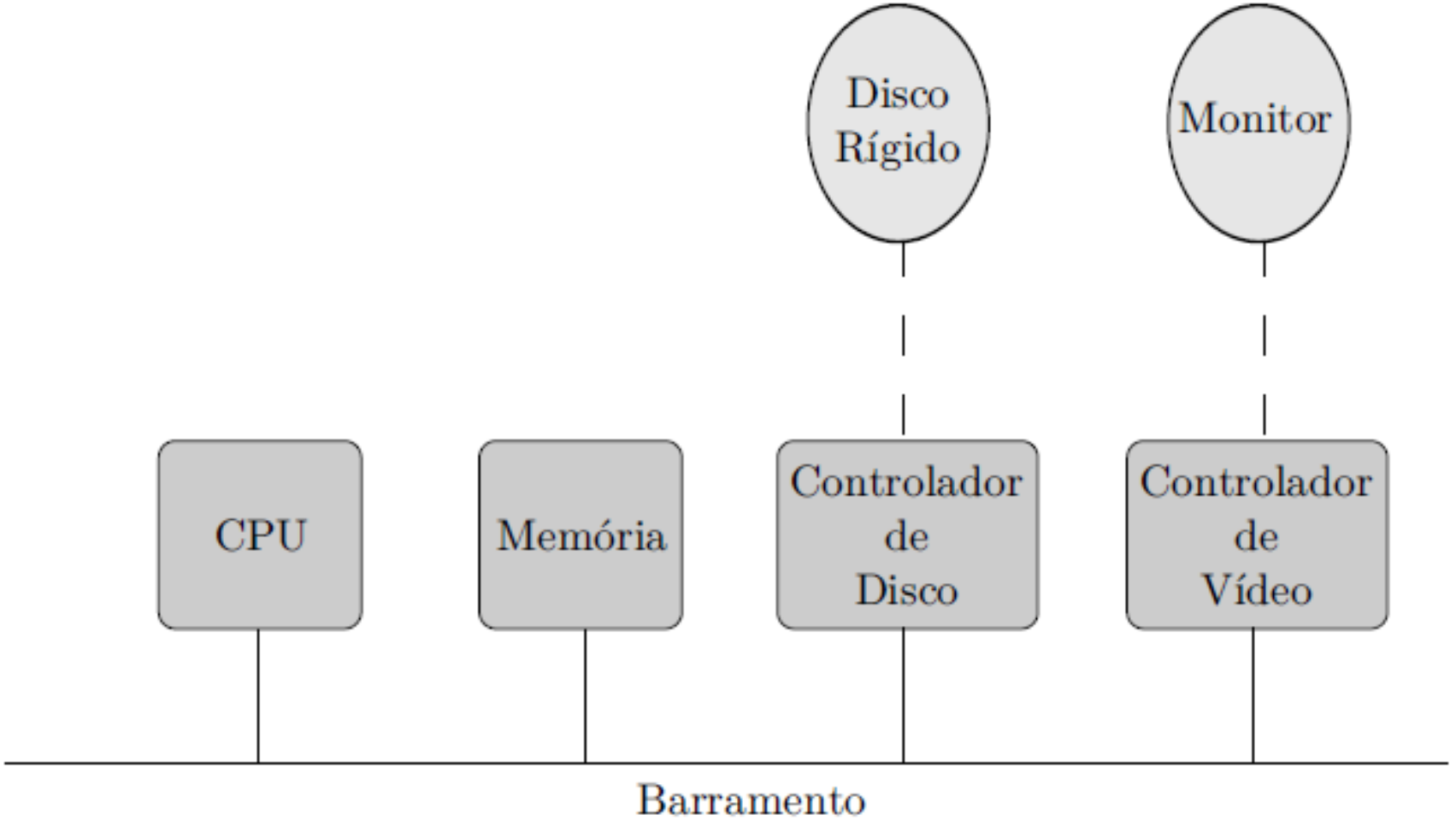
# Controladores

# Controladores

PUC Minas

- Em geral, dispositivos de E/S não estão diretamente ligados a um barramento.
- Ao contrário, há um dispositivo **intermediário** realizando esta conexão.
  - Chamado de **Controlador de E/S**.
  - Algumas vezes, também chamado de **Módulo de E/S**.
- O controlador se conecta tanto ao barramento de E/S, quanto ao dispositivo.
  - Recebe comandos e envia dados para a CPU.
  - Repassa estes comandos e interage com o dispositivo de E/S em si.

# Controladores




Fonte https://sites.google.com/a/ic.uff.br/fernanda/2017-2/fac

# Controladores – por quê?

PUC Minas

- Por que não conectar os dispositivos de E/S diretamente aos barramentos?
- Vários motivos:
  - Dispositivos de E/S diferentes possuem características diferentes.
    - ★ Tempos de acesso.
    - ★ Comandos.
    - ★ Formato dos dados.
    - ★ ...
  - Controladores abstraem as diferenças e tornam o acesso aos dados homogêneo.
  - Controladores podem controlar **múltiplas instâncias** de um dado dispositivo de forma transparente.
    - ★ *e.g.*, controladora de HD com múltiplos HDs conectados.
  - Controladores podem implementar **funcionalidades extras**.
    - ★ Que teriam que ser realizadas pela CPU, caso contrário.

Fonte https://sites.google.com/a/ic.uff.br/fernanda/2017-2/fac

# Controladores – Funcionalidades

PUC Minas

## Duas funcionalidades obvias de um controlador de E/S são:

- Comunicação com a CPU.
- Comunicação com o dispositivo de E/S.

## Além disso, são funcionalidades comuns:

- Controle e temporização.
- *Bufferização* dos dados.
- Detecção de erros.

Fonte https://sites.google.com/a/ic.uff.br/fernanda/2017-2/fac

# Controladores – Exemplo de Interação entre CPU e Controlador

PUC Minas

CPU solicita estado do dispositivo ao controlador.

Controlador verifica o estado do dispositivo e responde à CPU.

**Se o dispositivo está pronto:**

**CPU solicita leitura de dados.**

**Controlador recebe solicitação e a traduz para comandos do dispositivo.**

**Dispositivo executa requisição e repassa dados para o controlador.**

**Controlador traduz dados para formato adequado para comunicação com o processador.**

**Controlador envia dados para o processador.**

**Processador armazenada dados em memória.**

Fonte https://sites.google.com/a/ic.uff.br/fernanda/2017-2/fac

PUC Minas

# Mapeamentos de Dispositivos de E/S

# Mapeamentos

PUC Minas

- Do ponto de vista do programador, como acessar um dispositivo de E/S?
- Em outras palavras, como dizer ao processador que queremos realizar uma operação de E/S?
- Considere o seguinte exemplo:
  - Queremos alterar a cor de um determinado pixel exibido no monitor.
  - Como pedimos ao processador para que isso seja feito?
- Neste exemplo, temos uma operação de saída de dados.
  - Queremos "escrever" a cor de um pixel.
  - *i.e.*, sabemos o que queremos escrever.
- Mas qual é o **endereço**?

Fonte https://sites.google.com/a/ic.uff.br/fernanda/2017-2/fac

# Mapeamentos

PUC Minas

Há duas formas diferentes de lidar com o endereçamento de dispositivos de E/S:

- E/S mapeado em memória.
- E/S mapeado por porta.

É comum que arquiteturas suportem ambas as formas.

- Embora dispositivos de E/S complexos tendem a usar E/S mapeado por porta.

Fonte https://sites.google.com/a/ic.uff.br/fernanda/2017-2/fac

# Mapeamento em Memória

PUC Minas

No mapeamento em memória, dispositivos de E/S são associados a um ou mais endereços de memória.

Parte do **espaço de endereçamento** da MP é reservado para dispositivos de E/S.

- Endereços não são acessíveis na MP.

Leituras ou escritas nestes endereços são capturadas pelos dispositivos de E/S.

- Traduzidas em requisições de entrada ou saída de dados.

Instruções de leitura/escrita de memória são usadas para E/S.

- *e.g., load word, store word*.

Fonte https://sites.google.com/a/ic.uff.br/fernanda/2017-2/fac

# Mapeamento por Porta

PUC Minas

Também chamado de **E/S isolado**.

No mapeamento por porta, dispositivos são associados a outro tipo de endereço.

- Um número de porta.
- Há instruções especiais para ler/escrever em uma porta

Fonte https://sites.google.com/a/ic.uff.br/fernanda/2017-2/fac

# Mapeamento - comparação

PUC Minas

- Para que o mapeamento em memória funcione, dispositivos de E/S precisam estar conectados (de alguma forma) ao mesmo barramento da MP.
  - No mapeamento por porta, pode-se usar um barramento isolado.
  - Dedicado para I/O.
  - Pode afetar o desempenho.
- Por outro lado, mapeamento em memória simplifica lógica do processador.
  - Acesso à MP e a dispositivos de E/S é padronizado.
  - Não são necessárias instruções especializadas.
- Arquiteturas como a x86 suportam ambos os métodos.

PUC Minas

# Técnicas de Acesso

# Técnicas de Acesso

PUC Minas

- E/S é bem mais lenta que processamento.
  - E que acessos à MP.
- Uma vez que uma operação de E/S é requisitada, ela tipicamente demora vários ciclos de clock.
- Perguntas:
  - O que o processador faz neste período?
  - Como ele sabe que a operação foi concluída?
- Resposta: depende da técnica de acesso utilizada.
  - E/S programada.
  - E/S por interrupção.
  - E/S por DMA.

Fonte https://sites.google.com/a/ic.uff.br/fernanda/2017-2/fac

# Técnicas de Acesso – E/S Programada

PUC Minas

Processo de transferência de dados é completamente controlado pela CPU.

- Verifica estado do dispositivo.
- Envia pedido.
- Aguarda finalização.
- Recebe os dados.

Durante a operação, CPU fica “presa”.

- Ou ao menos gasta tempo para periodicamente verificar se a operação está pronta.
- Também gasta tempo interagindo com o dispositivo/controlador.

**Método ineficiente**.

Fonte https://sites.google.com/a/ic.uff.br/fernanda/2017-2/fac

# Técnicas de Acesso – E/S Programada

PUC Minas

**CPU solicita operação ao dispositivo/controlador.**

- Parâmetros enviados pelo barramento.

**Dispositivo realiza operação.**

**Enquanto isso, CPU periodicamente verifica se a operação terminou.**

- Chamado de polling ou interrogação.

**Eventualmente, operação termina e controlador/dispositivo armazena esta informação.**

- Um registrador interno ou um bit de controle.

Em dado momento, CPU volta a **verificar** e identifica que a operação está pronta.

Fonte https://sites.google.com/a/ic.uff.br/fernanda/2017-2/fac

# Técnicas de Acesso – E/S por Interrupção

PUC Minas

- E/S programada é muito ineficiente.
  - São gastos ciclos do processador, esperando pelo fim da operação.
  - Mesmo quando ações úteis são intercaladas.
    - ★ *i.e.*, quando não utilizamos espera ocupada.
- Ineficiência causada pela espera.
  - Não sabemos quanto tempo a operação demora.
  - Precisamos vericar de tempos em tempos.
- O ideal seria que o dispositivo/controlador pudesse **avisar** quando a operação fosse concluída.
  - Processador ficaria **completamente livre** para realizar outras tarefas.

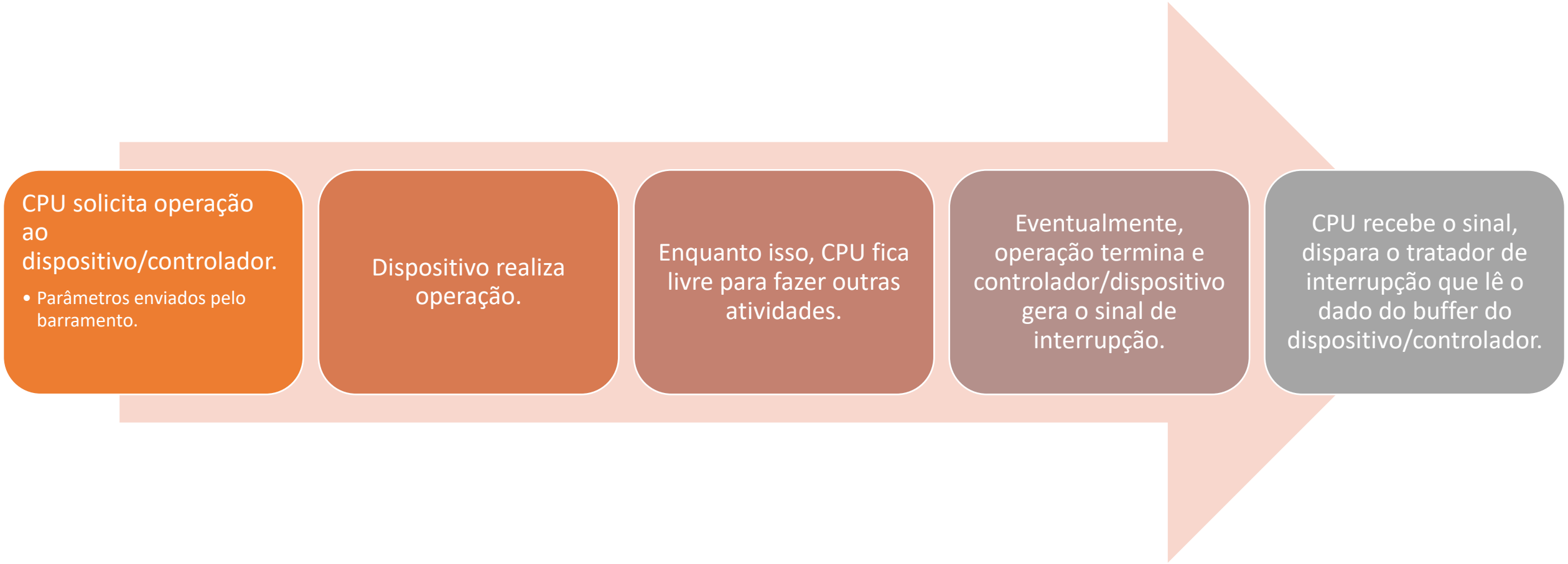
# Técnicas de Acesso – E/S por Interrupção

PUC Minas

- Esta é exatamente a estratégia usada na E/S por interrupção.
  - Processador requisita determinada operação a um dispositivo/controlador.
  - Enquanto a operação é realizada, CPU realiza outras tarefas.
    - ★ *e.g.*, executa outras instruções do programa.
  - Quando a operação é concluída, dispositivo/controlador gera uma **interrupção**.
- Uma interrupção é apenas um sinal elétrico de aviso ao processador.
- Uma vez recebido, processador para o que está fazendo e aciona um **tratador de interrupções**
  - Trecho de código que lida com o evento.

# Técnicas de Acesso – E/S por Interrupção - Exemplo

PUC Minas




Fonte https://sites.google.com/a/ic.uff.br/fernanda/2017-2/fac

# Técnicas de Acesso – E/S por Interrupção

PUC Minas

- Note que múltiplas operações de E/S podem ser executadas simultaneamente. Exemplo:
  - CPU requisita leitura do HD.
  - Enquanto a leitura é feita, CPU requisita amostra do microfone.
- Neste caso, ao receber uma interrupção, como a CPU sabe quem a gerou?
  - No exemplo, a controladora de disco ou a placa de som?
- Várias soluções:
  - Uma linha de interrupção para cada controlador/dispositivo.
    - ★ Limita número de dispositivos.
    - ★ Projeto mais caro e complexo.
  - Detecção por *software*: UCP interroga dispositivo/controlador (ineficiente).
  - Arbitração de barramento: dispositivo/controlador obtém barramento para ele momentaneamente.

Fonte https://sites.google.com/a/ic.uff.br/fernanda/2017-2/fac

# Técnicas de Acesso – DMA

PUC Minas

O uso de interrupções é bem mais eficiente que a E/S programada.

- Processador não precisa ficar verificando se a operação terminou.

Mas note que o processador ainda **controla** a execução da operação.

Considere, por exemplo, a tarefa de ler uma **grande quantidade de dados** do HD para a MP.

- Processador requisita leitura do primeiro setor.
- Processador fica livre para outras tarefas.
- Quando há uma interrupção, processador lê dado do barramento e transfere para a MP.
- Processador requisita leitura do próximo setor.

Fonte https://sites.google.com/a/ic.uff.br/fernanda/2017-2/fac

# Técnicas de Acesso – DMA

PUC Minas

- A cada interrupção, processador para o que está fazendo para dar **sequência à operação**.
- Os ciclos usados para isso poderiam ser mais bem empregados executando instruções úteis ao programa.
- Solução:
  - Usar um dispositivo auxiliar, chamado de **controlador de DMA**.
  - CPU informa os parâmetros da E/S no início.
  - Controlador de DMA interage com o controlador/dispositivo de E/S.
  - Apenas quando a transferência estiver **completamente concluída**, controlador de DMA gera uma interrupção para a CPU.

Fonte https://sites.google.com/a/ic.uff.br/fernanda/2017-2/fac

# Técnicas de Acesso – DMA

PUC Minas

Controlador de DMA transfere dados de E/S diretamente para a MP.

Para isso, controlador precisa estar conectado ao barramento da MP.

- Direta ou indiretamente.

Isso pode significar um prejuízo nos acessos da CPU à MP.

Mas, em geral, o desempenho com DMA é bastante superior **em grandes transferências de dados**.

**Podemos usar DMA, interrupções, E/S programada tanto para mapeamento em memória, quanto para mapeamento por porta.**

Fonte https://sites.google.com/a/ic.uff.br/fernanda/2017-2/fac

# Referências

PUC Minas


- NEGUS, Christopher. **Linux** -– A Bíblia -– o Mais Abrangente e Definitivo Guia Sobre Linux. Alta Books, 2014. 852 p. ISBN: 9788576087991.
- MACHADO, Francis Berenger; MAIA, Luiz Paulo. Arquitetura de sistemas operacionais. 5a ed. Rio de Janeiro: LTC, 2013. 263 p. ISBN:9788521622109.
- PAIXÃO, Renato Rodrigues. **Arquitetura de Computadores** – PCs. São Paulo: Editora Érica, 2016. ISBN digital 9788536518848.
- SILBERSCHATZ, Abraham. Fundamentos de Sistemas Operacionais. 9ª ed. LTC, 2015. 524 p. ISBN: 9788521629399. ISBN digital 9788521630012.
- STUART, Brian L.. Princípios de sistema operacionais: projetos e aplicações. São Paulo: Cenage Learning, 2011. 637 p. ISBN: 9788522107339.
- TANENBAUM, Andrew S. Sistemas Operacionais Modernos. 4a ed. São Paulo: Pearson, 2016. 864 p. ISBN: 9788543005676.
- ZACKER, Craig. **Instalação e Configuração do Windows Server 2012 R2**. Porto Alegre: Editora Bookman, 2015. ISBN digital 9788582603581.