

**MC542**

**Organização de Computadores  
Teoria e Prática**

**2006**

**Prof. Paulo Cesar Centoducatte**

**[ducatte@ic.unicamp.br](mailto:ducatte@ic.unicamp.br)**

**[www.ic.unicamp.br/~ducatte](http://www.ic.unicamp.br/~ducatte)**

**MC542**

# **Arquitetura de Computadores**

## **Sistemas de Armazenagem (IO)**

**"Computer Organization and Design:  
The Hardware/Software Interface" (Capítulo 8)**

# Sistema de Armazenagem

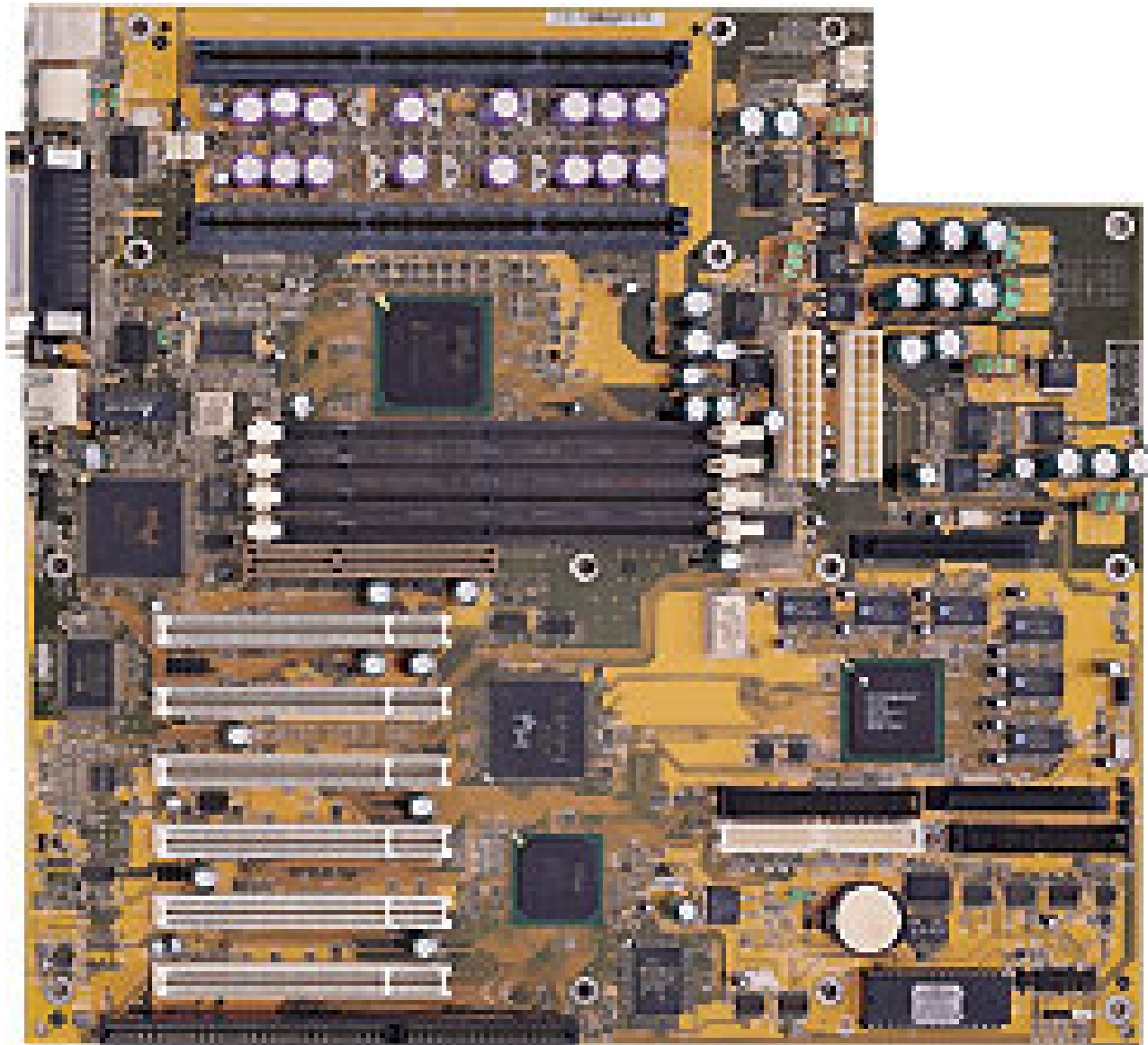
## Sumário

- **Motivação**
- **Introdução**
- **Tipos de Dispositivos de Armazenagem**
- **Discos, Desempenho, Histórico**
- **Barramentos (busses): Conectando Dispositivos de IO à CPU e Memória**
  - **Sistemas de Barramentos**
  - **Arbitragem em Barramentos**
- **Interface: Processador & I/O**
  - **Pooling e Interrupção**
- **RAID, Disponibilidade e Confiabilidade**

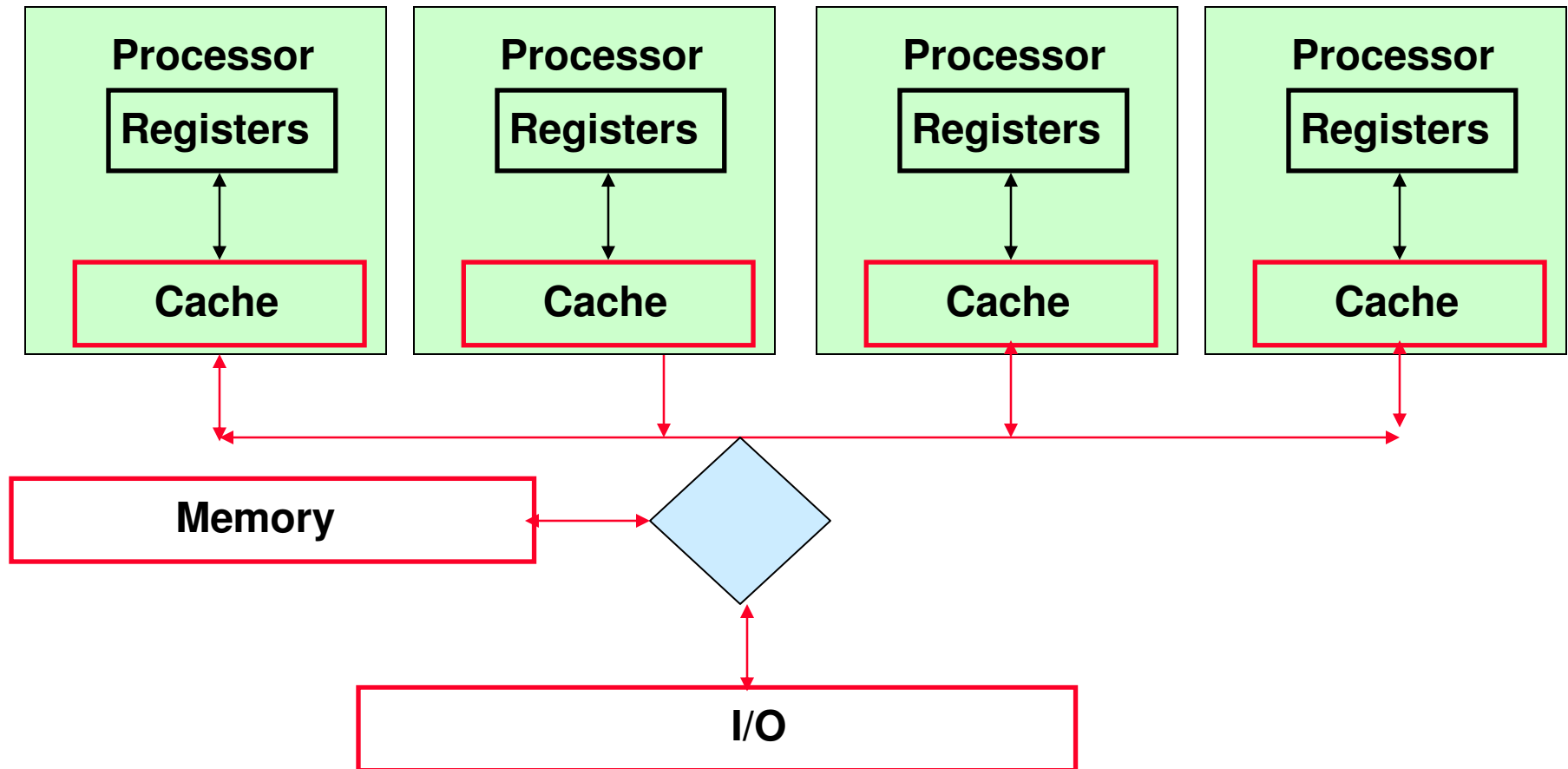
# Motivação

- Desempenho de CPU: 60% por ano
- Desempenho de sistemas de I/O: limitado por delays mecânicos (disco I/O)
  - 10% por ano (IO por seg)
- Lei de Amdahl: speed-up limitado pelo sub-sistema mais lento!
  - Se IO é 10% do tempo e melhorarmos 10x a CPU
    - » Desempenho do sistema será ~5x maior (perda de ~50%)
  - Se IO 10% do tempo e melhorarmos 100x CPU
    - » O desempenho do sistema será ~10x maior (perda de ~90%)
- I/O bottleneck:
  - Reduz a fração do tempo na CPU
  - Reduz o valor de CPUs mais rápidas

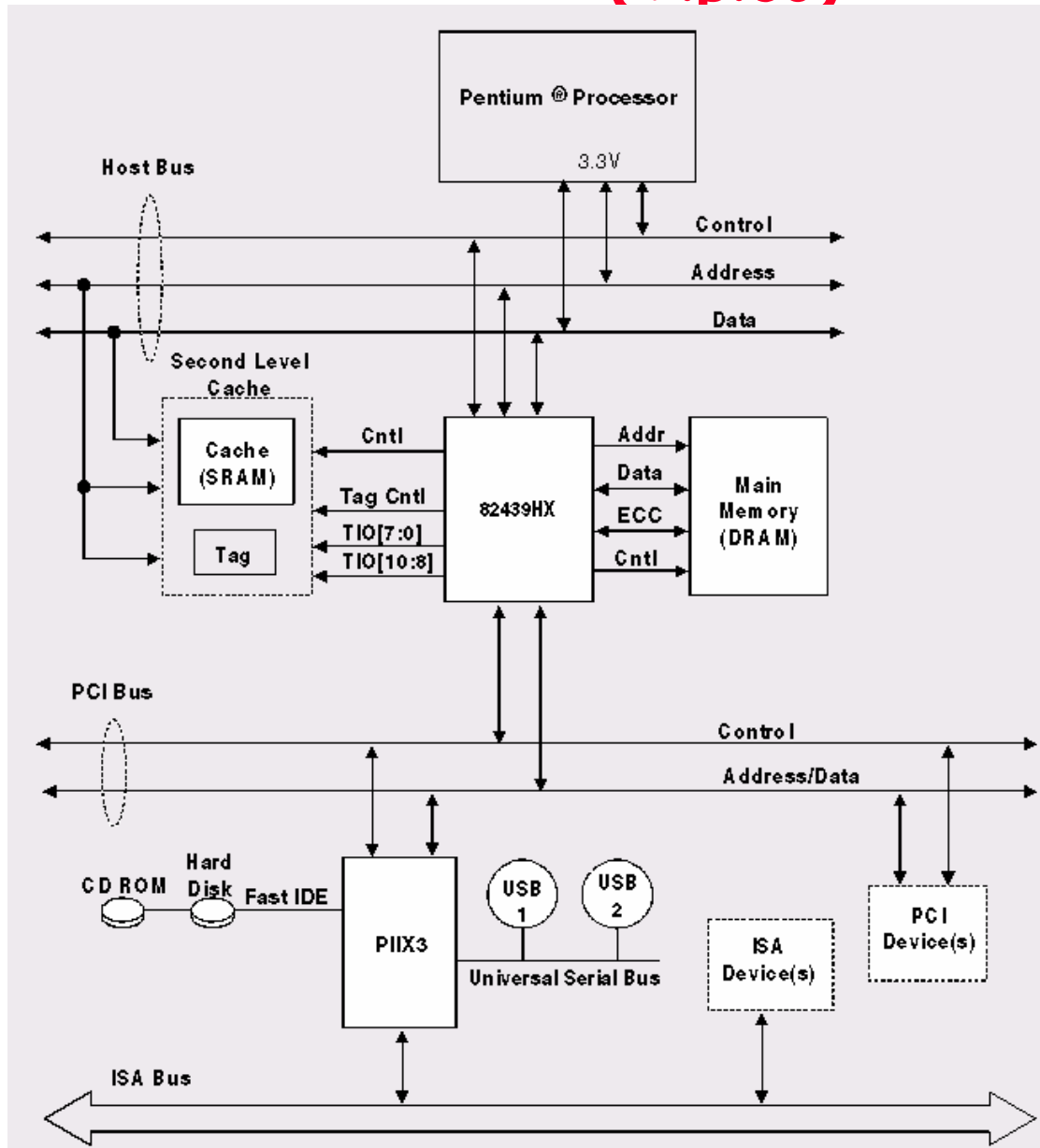
# Sistema Computacional



# Organização de Sistemas Computacionais (Típico)



# Organização de Sistemas Computacionais (Típico)

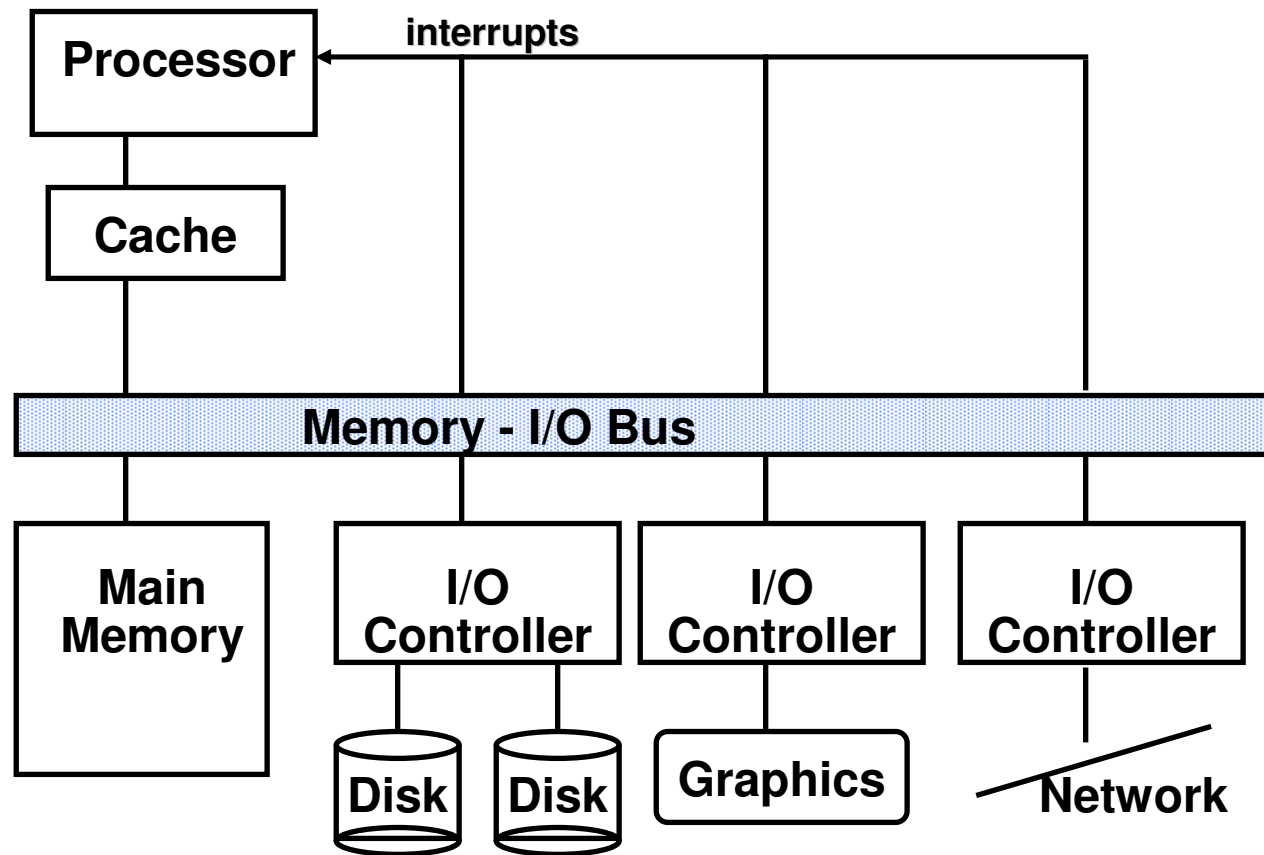


Processor/Memory Bus

PCI Bus

I/O Busses

# Sistema de IO





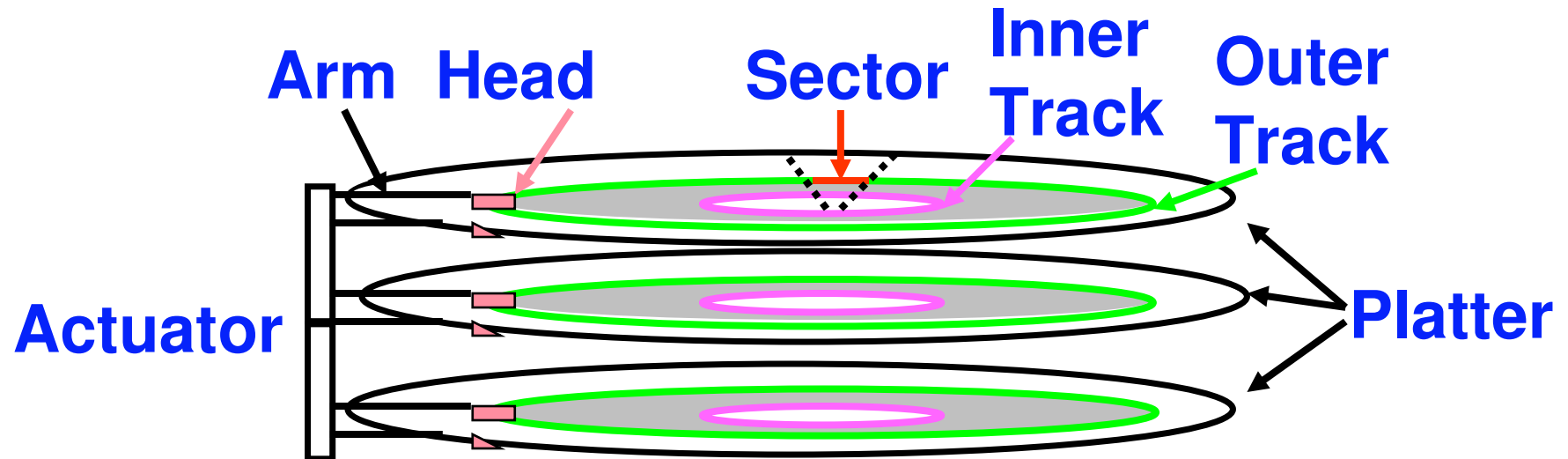
# Tecnologia dos Dispositivos

- **Dirigidos pelo paradigma de computação vigente**
  - 1950s: migração de batch para processamento on-line
  - 1990s: migração para computação ubíqua (unipresente)
    - » Computação em telefones, livros, carros, vídeo, câmeras, ...
    - » Rede de fibra opticana internacionais
    - » wireless
- **Efeitos na Indústria de Dispositivos de Armazenagem:**
  - Embedded storage
    - » pequeno, barato, mais confiável, baixo consumo
  - Dados
    - » Alta capacidade, gerenciamento hierarquico do armazenamento

# Tipos de Dispositivos de Armazenamento

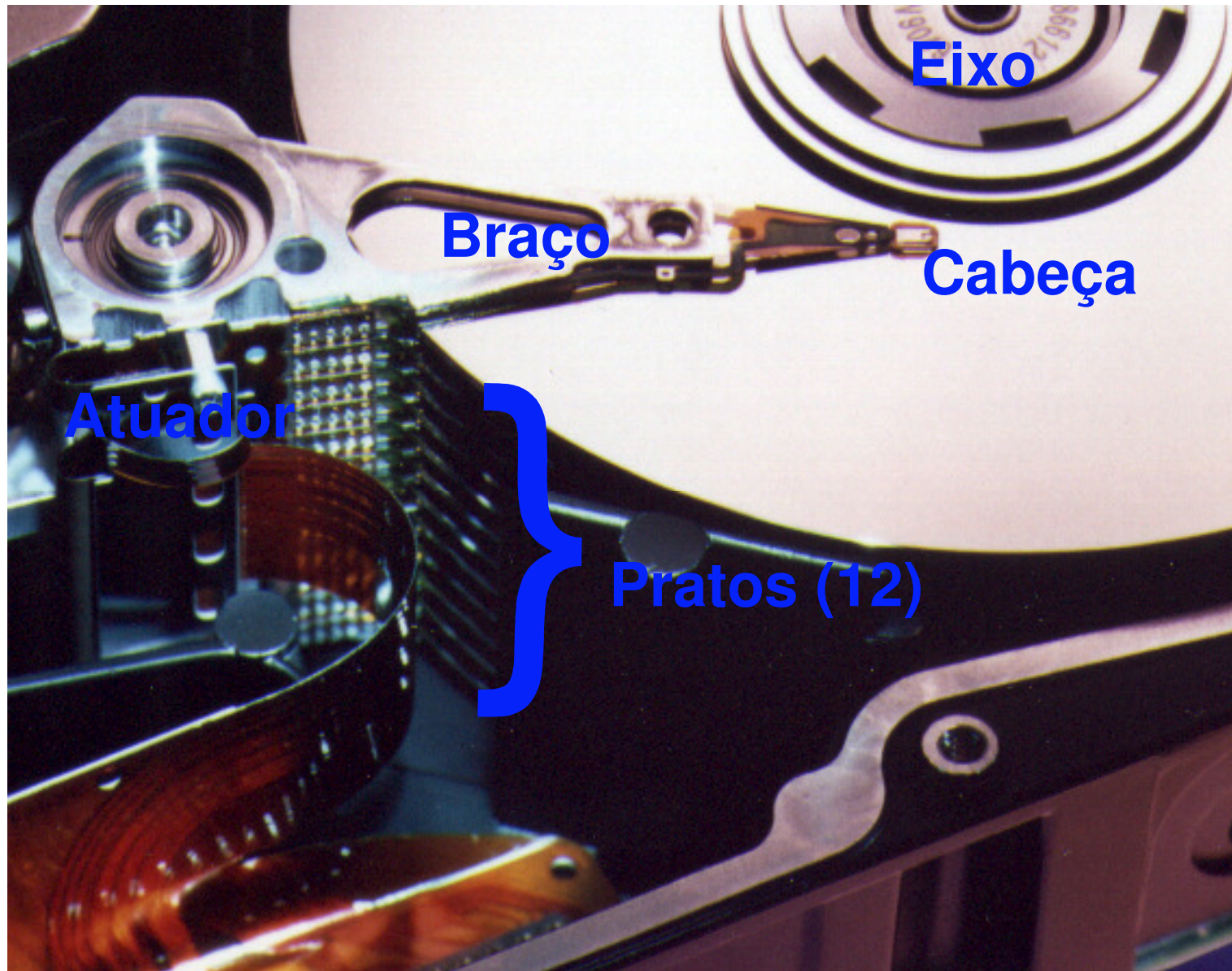
- **Finalidade:**
  - Longa duração, armazenamento não volátil
  - Grande, barato, usado nos níveis mais baixo da hierarquia
- **Bus Interface:**
  - IDE
  - SCSI - Small Computer System Interface
  - Fibre Channel
- **Taxa de Transfêrrencia**
  - Cerca de 120 Mbyte/second através da interface de barramento.
  - Cerca de 5 Mbyte/second por heads.
  - Dados são movidos em Blocos
- **Capacidade**
  - Mais de 100 Gigabytes
  - Quadruplica a cada 3 anos
  - Podem ser agrupados para armazenarem terabytes de dados.

# Disk Drivers: Terminologia



- Vários **pratos**, com a informação armazenada magneticamente em ambas superfícies (usual)
- Bits armazenados em **trilhas**, que por sua vez são divididas em **setores** (e.g., 512 Bytes)
- O **Atuador** move a **cabeça** (fim do braço, 1/superfície) sobre a trilha ("**seek**"), seleciona a **superfície**, espera pelo **setor** passar sob a **cabeça**, então lê ou escreve
  - "**Cilindro**": todas as trilhas sob as cabeças

## Foto: Braço, Cabeça, Atuador e Pratos



# Discos: Exemplos

## Seagate Cheetah ST3146807FC

**147 Gigabytes**

**10,000 RPM**

**4.7 ms avg seek time.**

**Fibre Channel**

**\$499.00**

**4 disks, 8 heads**

**290,000,000 Total Sectors**

**50,000 cylinders**

**Average of 6,000 sectors/cylinder or 800 sectors / track (but different amounts on each track.)**

**MTBF = 1,200,000 hours**

<http://www.seagate.com/cda/products/discsales/marketing/detail/0,1121,355,00.html>

# Discos: Exemplos

## Barracuda Cheetah ST320822A

**200 Gigabytes**

**7,200 RPM**

**8.5 ms avg seek time.**

**ATA**

**\$299.00**

**2 disks, 4 heads**

**390,000,000 Total Sectors**

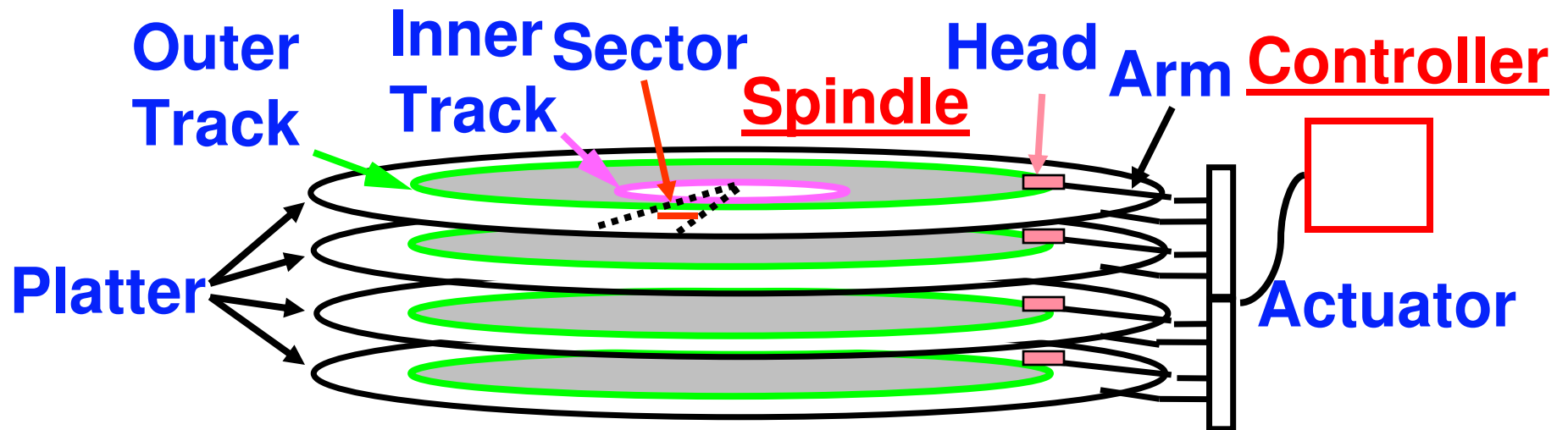
**24,000 cylinders**

**Average of 16,000 sectors/cylinder or 400 sectors / track (but different amounts on each track.)**

**MTBF = ?????????????? hours**

<http://www.seagate.com/support/disc/manuals/fc/100195490b.pdf>

# Disk Device: Desempenho



- **Disk Latency = Seek Time + Rotation Time + Transfer Time + Controller Overhead**
- **Seek Time?** Depende do no. de trilhas e velocidade de **seek** do disco
- **Rotation Time?** depende da velocidade de rotação do disco
- **Transfer Time?** depende do **data rate (bandwidth)** do disco (densidade dos bits), tamanho da requisição

# Disk Device: Desempenho

- Distância Média do setor à Cabeça?
- 1/2 tempo de uma Rotação
  - 10000 Revoluções Por Minuto  $\Rightarrow$  166.67 Rev/sec
  - 1 revolução =  $1 / 166.67$  seg  $\Rightarrow$  6.00 milissegundos
  - 1/2 rotação (revolução)  $\Rightarrow$  3.00 ms
- Nº Médio de Trilhas Saltadas pelo Braço?
  - Soma das distâncias de todos os seeks possíveis a partir de todas as trilhas possíveis / # possibilidades
    - » Assume-se distribuição randômica
  - Indústria usa benchmark padrão

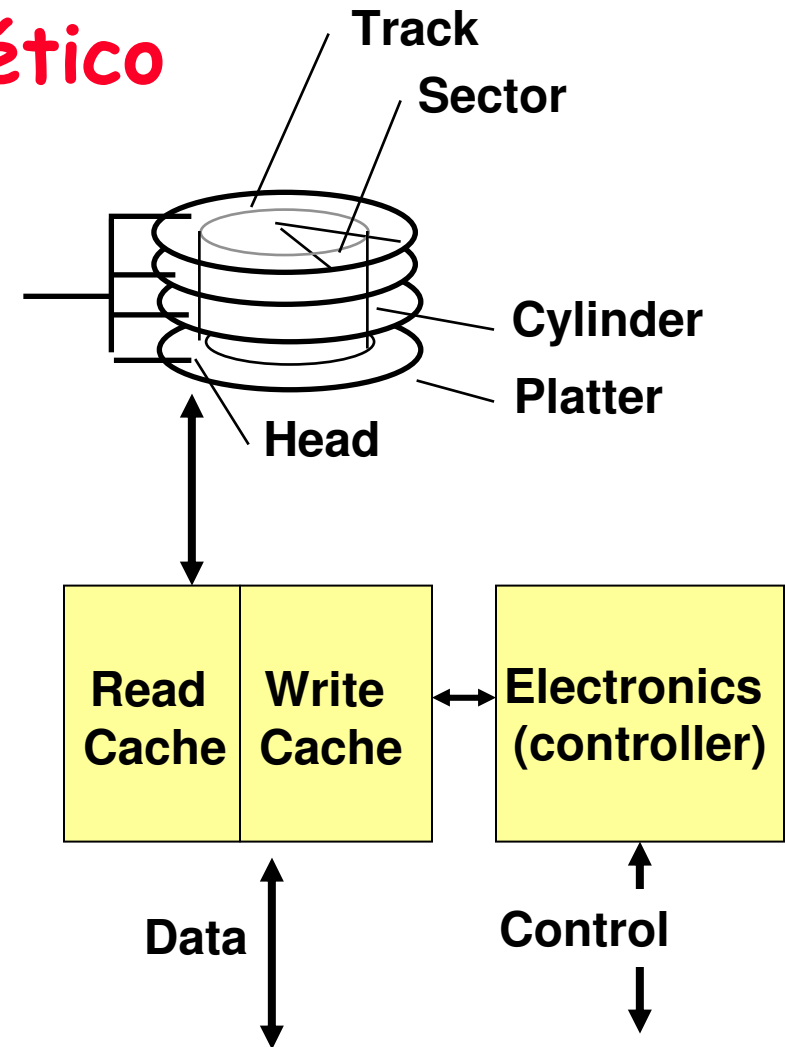


## Data Rate: Trilha Interna vs. Externa

- Por questões de simplicidade, originalmente tem-se o mesmo número de setores por trilha
  - Como as trilhas externas são maiores elas possuem menos bits por polegada
- Competição  $\Rightarrow$  decisão de se ter o mesmo BPI (bit per inch) para todas as trilhas ("densidade de bits constante")
  - $\Rightarrow$  Maior capacidade por disco
  - $\Rightarrow$  Mais setores por trilha nas bordas
  - $\Rightarrow$  Uma vez que a velocidade rotacional é constante, trilhas externas possuem data rate maior (maior velocidade linear)
- Bandwidth da trilha externa é 1.7X a da trilha interna!
  - Trilha interna possui densidade maior, trilha externa possui densidade menor, a densidade não é constante  
(2.1X length of track outer / inner; 1.7X bits outer / inner)

# Disco Magnético

- **Propósito:**
  - Longo tempo, não volátil
  - Grande, barato, baixo nível na hierarquia de memória
- **Características:**
  - Seek Time (~8 ms avg)
    - » latência posicional
    - » latência rotacional
- **Taxa de Transferência**
  - 10-40 MByte/sec
  - Blocos
- **Capacidade**
  - Gigabytes
  - 4X a cada 3 anos



Tempo de Resposta (**Response time**)  
= Queue + Controller + Seek + Rot + Xfer

Service time

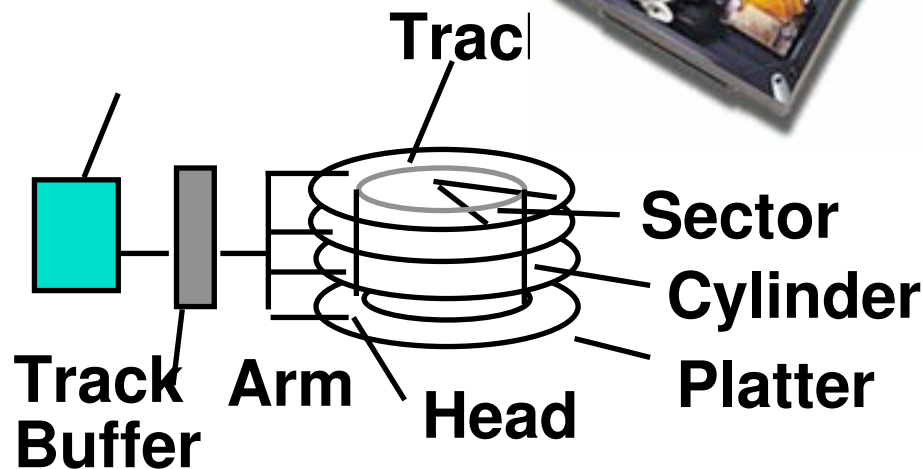
# Disco: Modelo de Desempenho

- **Capacidade**
  - + 100%/ano (2X / 1.0 ano)
- **Transfer rate (BW)**
  - + 40%/ano (2X / 2.0 anos)
- **Tempo de Rotação + Seek**
  - 8%/ ano (1/2 em 10 anos)
- **MB/\$**
  - > 100%/ano (2X / 1.0 ano)

# Barracuda 180



- 181.6 GB, 3.5 inch disk
- 12 platters, 24 surfaces
- 24,247 cylinders
- 7,200 RPM; (4.2 ms avg. latency)
- 7.4/8.2 ms avg. seek (r/w)
- 64 to 35 MB/s (internal)
- 0.1 ms controller time
- 10.3 watts (idle)



Latency =

*por acesso* { Queuing Time + Controller time + Seek Time + Rotation Time + Size / Bandwidth

+  
*por byte*

fonte: [www.seagate.com](http://www.seagate.com)

## Desempenho de Disco: Exemplo

- Tempo calculado para ler 64 KB (128 setores) no "Barracuda 180" usando os dados de desempenho informados (os setores estão na trilha externa)

latência = average seek time + average rotational delay + transfer time + controller overhead

$$= 7.4 \text{ ms} + 0.5 * 1/(7200 \text{ RPM}) \\ + 64 \text{ KB} / (64 \text{ MB/s}) + 0.1 \text{ ms}$$

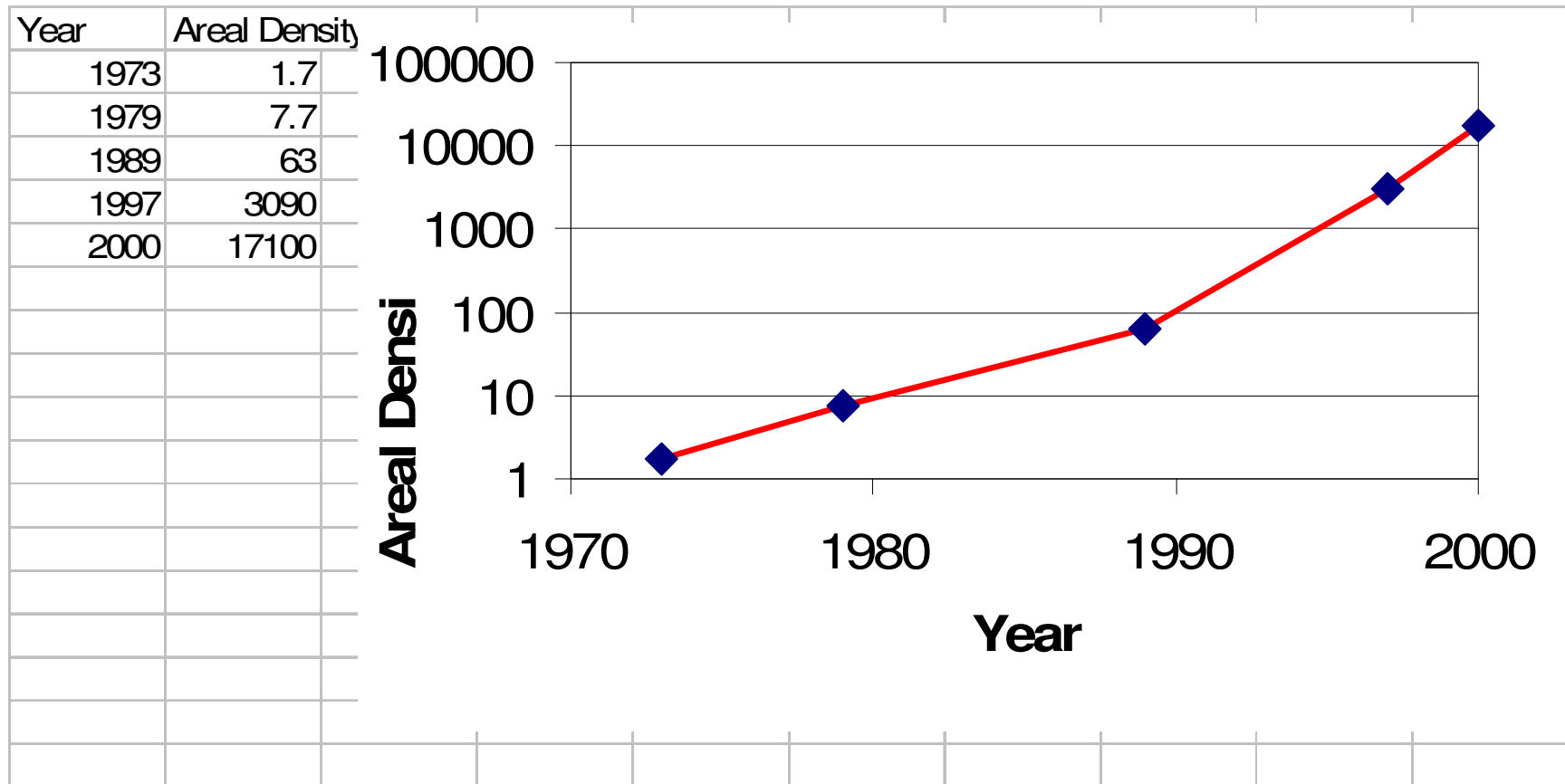
$$= 7.4 \text{ ms} + 0.5 / (7200 \text{ RPM} / (60000 \text{ ms/M})) \\ + 64 \text{ KB} / (64 \text{ KB/ms}) + 0.1 \text{ ms}$$

$$= 7.4 + 4.2 + 1.0 + 0.1 \text{ ms} = 12.7 \text{ ms}$$

# Densidade em Área

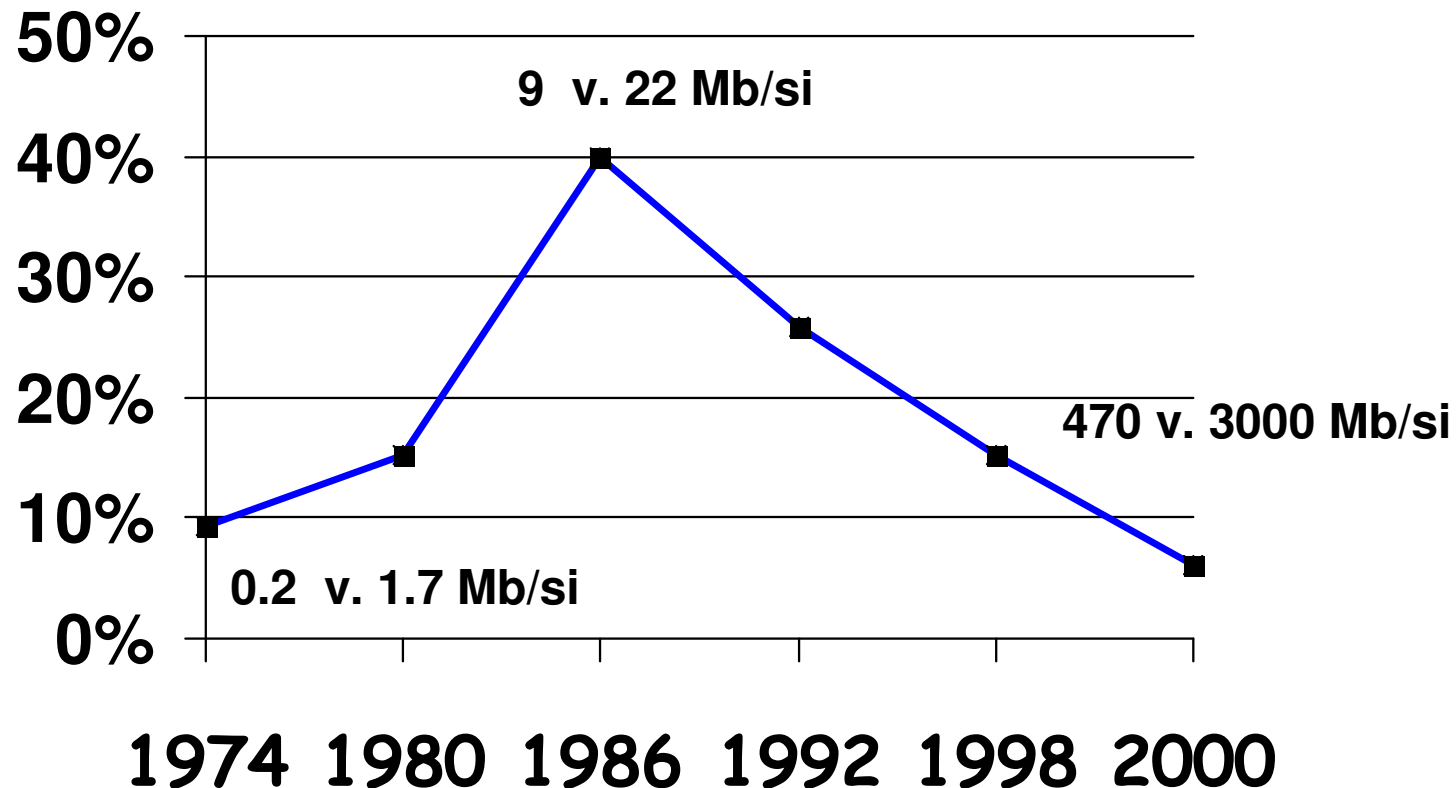
- Os Bits estão armazenados ao longo da trilha
  - Métrica: Bits Per Inch (BPI)
- Número de trilhas por superfície
  - Métrica: Tracks Per Inch (TPI)
- Projetistas de Discos falam em **densidade de bits por área**
  - Métrica: Bits Per Square Inch
  - Denominado: Areal Density
  - $\text{Areal Density} = \text{BPI} \times \text{TPI}$

# Densidade por Área



$$\text{Areal Density} = \text{BPI} \times \text{TPI}$$

# MBits per square inch: DRAM como % de Disco ao longo do tempo



fonte: New York Times, 2/23/98, page C3,

“Makers of disk drives crowd even more data into even smaller spaces”

Paulo C. Centoducatte - IC/Unicamp- 2002s1

©1998 Morgan Kaufmann Publishers



# Histórico

- 1956 IBM Ramac — início 1970s Winchester
  - Desenvolvido para computadores mainframe, interface proprietária
  - 27 inch a 14 inch
- Forma e capacidade orientaram o mercado mais que desempenho
- 1970s: Mainframes  $\Rightarrow$  discos de 14 inch de diâmetro
- 1980s: Minicomputadores, Servidores  $\Rightarrow$  8", 5 1/4" de diâmetro
- Fim 1980s/Início 1990s: PCs, workstations
  - Começou a se tornar realidade o mercado de discos de alta capacidade
    - » Padrões da indústria: SCSI, IPI, IDE
  - Pizzabox PCs  $\Rightarrow$  discos de 3.5 inch de diâmetro
  - Laptops, notebooks  $\Rightarrow$  discos de 2.5 inch
  - Palmtops não usam discos
- 2000s:
  - 1 inch para mobile devices (câmeras, telefone celular)?
  - Seagate: 12GB, 1inch hard drive disk (fev/2006)

# História

Data  
density  
Mbit/sq. in.

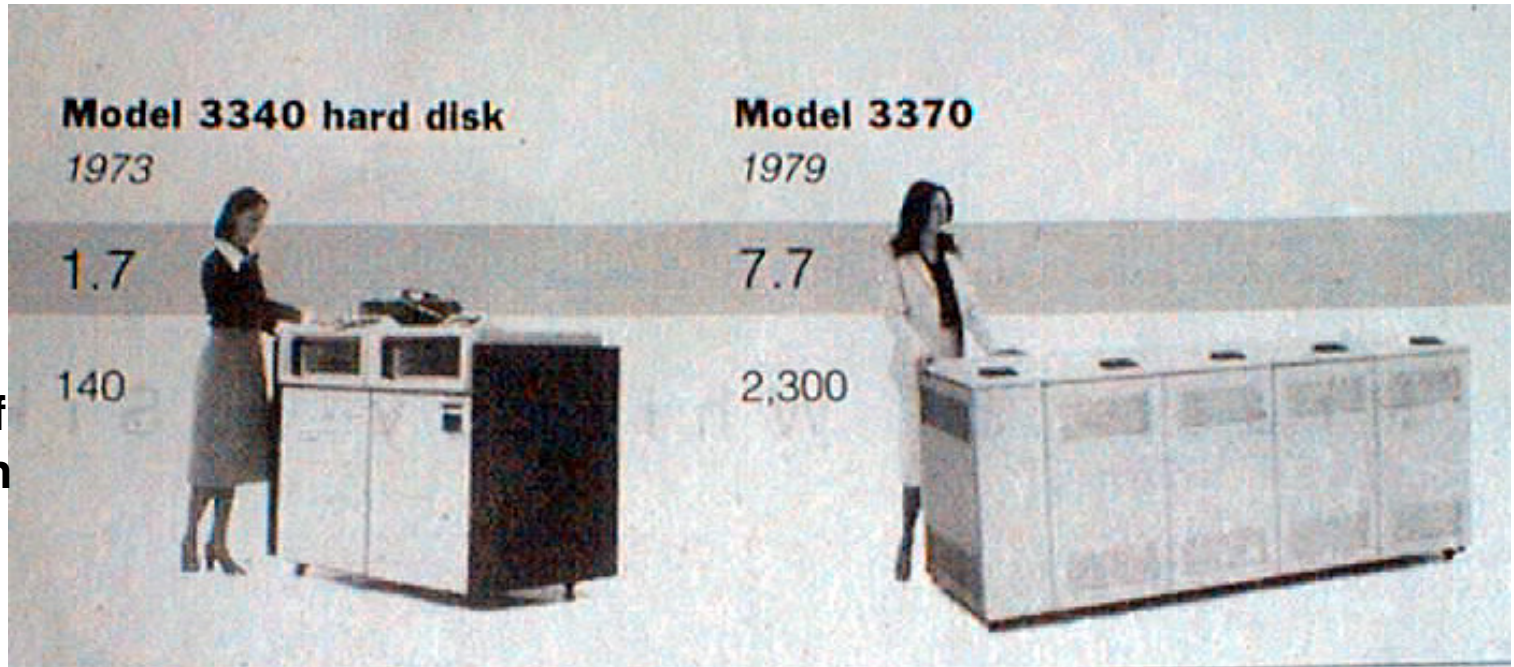
**Model 3340 hard disk**

1973

1.7

Capacity of  
Unit Shown  
Megabytes

140



**Model 3370**

1979

7.7

2,300

**1973:**

**1.7 Mbit/sq. in**

**140 MBytes**

**1979:**

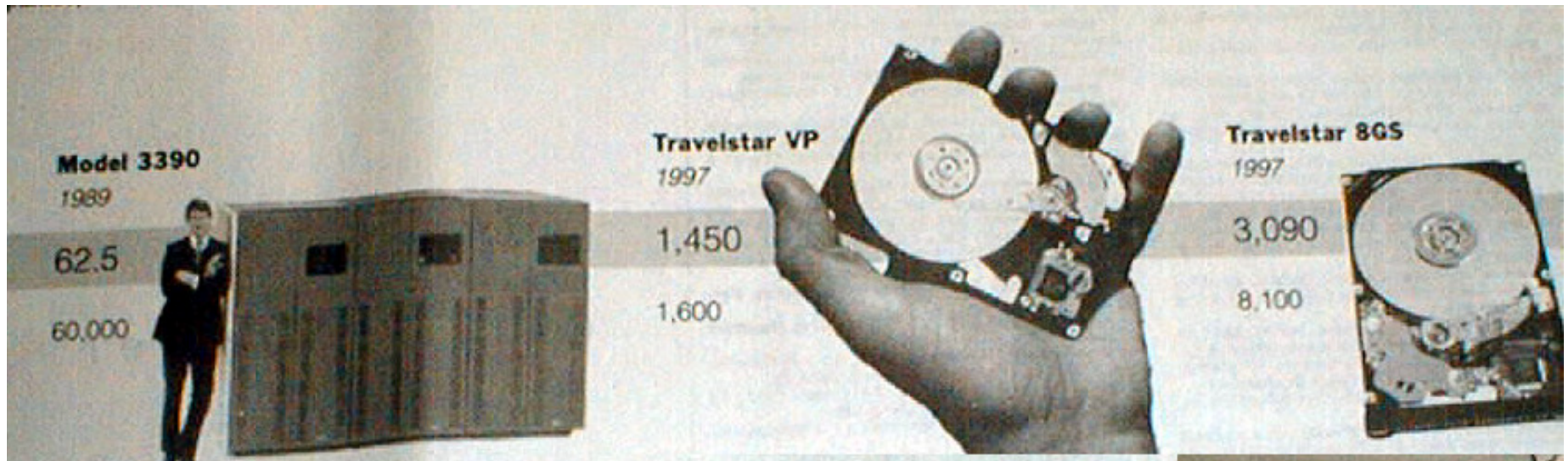
**7.7 Mbit/sq. in**

**2,300 MBytes**

*fonte: New York Times, 2/23/98, page C3,*

*“Makers of disk drives crowd even more data into even smaller spaces”*

# História



**1989:**  
**63 Mbit/sq. in**  
**60,000 MBytes**

**1997:**  
**1450 Mbit/sq. in**  
**2300 MBytes**

**1997:**  
**3090 Mbit/sq. in**  
**8100 MBytes**

*fonte: New York Times, 2/23/98, page C3,*

*“Makers of disk drives crowd even more data into even smaller spaces”*

# História

## disk drive de 1 inch

- **2000 IBM MicroDrive:**
  - 1.7" x 1.4" x 0.2"
  - 1 GB, 3600 RPM, 5 MB/s, 15 ms seek
  - Digital camera, PalmPC?
- **2006 MicroDrive?**
- **9 GB, 50 MB/s!**
  - Assumindo que tenham encontrado um nicho e o produto é um sucesso
  - Assumindo que as tendências de 2000 continuem



# Características dos discos em 2000

	<b>Seagate Cheetah ST173404LC Ultra160 SCSI</b>	<b>IBM Travelstar 32GH DJSA - 232 ATA-4</b>	<b>IBM 1GB Microdrive DSCM-11000</b>
Disk diameter (inches)	3.5	2.5	1.0
Formatted data capacity (GB)	73.4	32.0	1.0
Cylinders	14,100	21,664	7,167
Disks	12	4	1
Recording Surfaces (Heads)	24	8	2
Bytes per sector	512 to 4096	512	512
Avg Sectors per track (512 byte)	~ 424	~ 360	~ 140
Max. areal density(Gbit/sq.in.)	6.0	14.0	15.2
	<b>\$828</b>	<b>\$447</b>	<b>\$435</b>

# Características dos discos em 2000

	<b>Seagate Cheetah ST173404LC Ultra160 SCSI</b>	<b>IBM Travelstar 32GH DJSA - 232 ATA-4</b>	<b>IBM 1GB Microdrive DSCM-11000</b>
Rotation speed (RPM)	10033	5411	3600
Avg. seek ms (read/write)	5.6/6.2	12.0	12.0
Minimum seek ms (read/write)	0.6/0.9	2.5	1.0
Max. seek ms	14.0/15.0	23.0	19.0
Data transfer rate MB/second	27 to 40	11 to 21	2.6 to 4.2
Link speed to buffer MB/s	160	67	13
Power idle/operating Watts	16.4 / 23.5	2.0 / 2.6	0.5 / 0.8

# Características dos discos em 2000

	<b>Seagate Cheetah ST173404LC Ultra160 SCSI</b>	<b>IBM Travelstar 32GH DJSA - 232 ATA-4</b>	<b>IBM 1GB Microdrive DSCM-11000</b>
Buffer size in MB	4.0	2.0	0.125
Size: height x width x depth inches	1.6 x 4.0 x 5.8	0.5 x 2.7 x 3.9	0.2 x 1.4 x 1.7
Weight pounds	2.00	0.34	0.035
Rated MTTF in powered-on hours	1,200,000	(300,000?)	(20K/5 yr life?)
% of POH per month	100%	45%	20%
% of POH seeking, reading, writing	90%	20%	20%

# Características dos discos em 2000

	<b>Seagate Cheetah ST173404LC Ultra160 SCSI</b>	<b>IBM Travelstar 32GH DJSA - 232 ATA-4</b>	<b>IBM 1GB Microdri DSCM-11000</b>
Load/Unload cycles (disk powered on/off)	250 per year	300,000	300,000
Nonrecoverable read errors per bits read	<1 per $10^{15}$	< 1 per $10^{13}$	< 1 per $10^{13}$
Seek errors	<1 per $10^7$	not available	not available
Shock tolerance: Operating, Not operating	10 G, 175 G	150 G, 700 G	175 G, 1500 G
Vibration tolerance: Operating, Not operating (sine swept, 0 to peak)	5-400 Hz @ 0.5G, 22-400 Hz @ 2.0G	5-500 Hz @ 1.0G, 2.5-500 Hz @ 5.0G	5-500 Hz @ 1G, 500 Hz @ 5G



# Falácia: Use o Tempo "Average Seek" do Fabricante

- Os Fabricantes necessitam de padrões para comparações ("benchmark")
  - Calculam todos os seeks a partir de todas as trilhas, dividem pelo número de seeks => "average"
- A Média Real deve ser baseada em como os dados são armazenados no disco (definindo os seeks em aplicações reais)
  - Usualmente, a tendência é as trilhas acessadas serem próximas e não randômicas
- **Rule of Thumb:** "average seek time" observado na prática é tipicamente cerca de 1/4 a 1/3 do "average seek time" cotado pelo fabricante (i.é., 3X-4X mais rápido)
  - Barracuda 180 X avg. seek: 7.4 ms  $\Rightarrow$  2.5 ms

# Falácia: Use o "Transfer Rate" do Fabricante

- Os Fabricantes cotam a velocidade dos dados na superfície do disco ("internal media rate")
- Setores contém campos para detecção e correção de erros (pode ser até 20% do tamanho do setor); número do setor e os dados
- Existem **gaps** entre os setores em uma trilha
- **Rule of Thumb**: Os discos utilizam cerca de 3/4 da "internal media rate" (1.3X mais lento) para dados
- Por exemplo, Barracuda 180X:

64 a 35 MB/sec para a "internal media rate"

⇒ 47 a 26 MB/sec "external data rate" (74%)

## Desempenho de Discos: Exemplo

- Calcular o tempo para ler 64 KB do "Barracuda 180" outra vez, agora use 1/3 do seek time cotado e 3/4 do "internal outer track bandwidth; (Anterior: 12.7 ms)

Latência = average seek time + average rotational delay  
+ transfer time + controller overhead

$$= (\underline{0.33} * 7.4 \text{ ms}) + 0.5 * 1/(7200 \text{ RPM}) \\ + 64 \text{ KB} / (\underline{0.75} * 64 \text{ MB/s}) + 0.1 \text{ ms}$$

$$= \underline{2.5} \text{ ms} + 0.5 / (7200 \text{ RPM} / (60000 \text{ms/M})) \\ + 64 \text{ KB} / (\underline{48} \text{ KB/ms}) + 0.1 \text{ ms}$$

$$= \underline{2.5} + 4.2 + \underline{1.33} + 0.1 \text{ ms} = \underline{8.13} \text{ ms (64\% of 12.7)}$$

# Barramentos (busses): Conectando Dispositivos de IO à CPU e Memória

- De uma forma simples, um barramento (**bus**) é a conexão entre vários chips/componentes em um computador.
- O barramento é responsável por enviar dados/controlar entre esses vários componentes.

# Barramentos

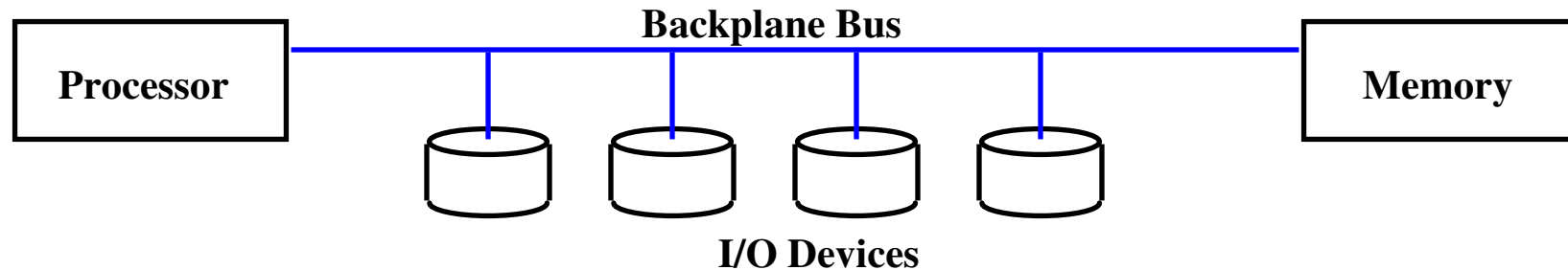
- Interconexão = liga as interfaces dos componentes do sistema
- Interfaces de hw de alta velocidade + protocolo lógico
- Networks, channels, backplanes

	Network	Channel	Backplane
<b>Conexão</b>	Máquinas	Devices	Chips
<b>Distância</b>	>1000 m	10 - 100 m	0.1 m
<b>Bandwidth</b>	10 - 1000 Mb/s	40 - 1000 Mb/s	320 - 2000+ Mb/s
<b>Latência</b>	alta ( 1ms)	média	baixa (Nanosecs.)
<b>Confiabilidade</b>	baixa Extensive CRC	média Byte Parity	alta Byte Parity

**message-based narrow pathways distributed arbitration**
←→
**memory-mapped wide pathways centralized arbitration**

# Barramentos

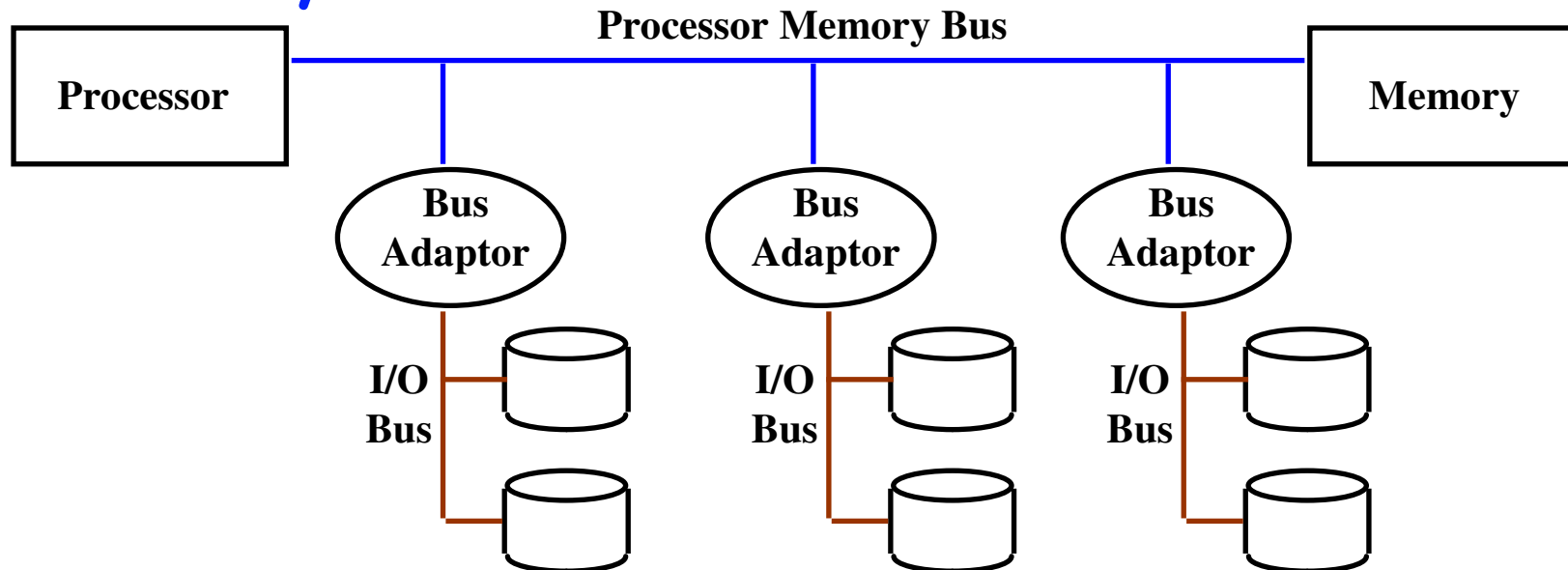
## Systemas com Um Barramento - Backplane Bus



- **Single bus (backplane bus)** é usado para:
  - Comunicação entre o Processador e a Memória
  - Comunicação entre dispositivos de I/O e memória
- **Vantagens: Simples e baixo custo**
- **Desvantagens: lento e o barramento, em geral, torna-se o maior gargalo**
- **Exemplo: IBM PC - AT**

# Barramentos

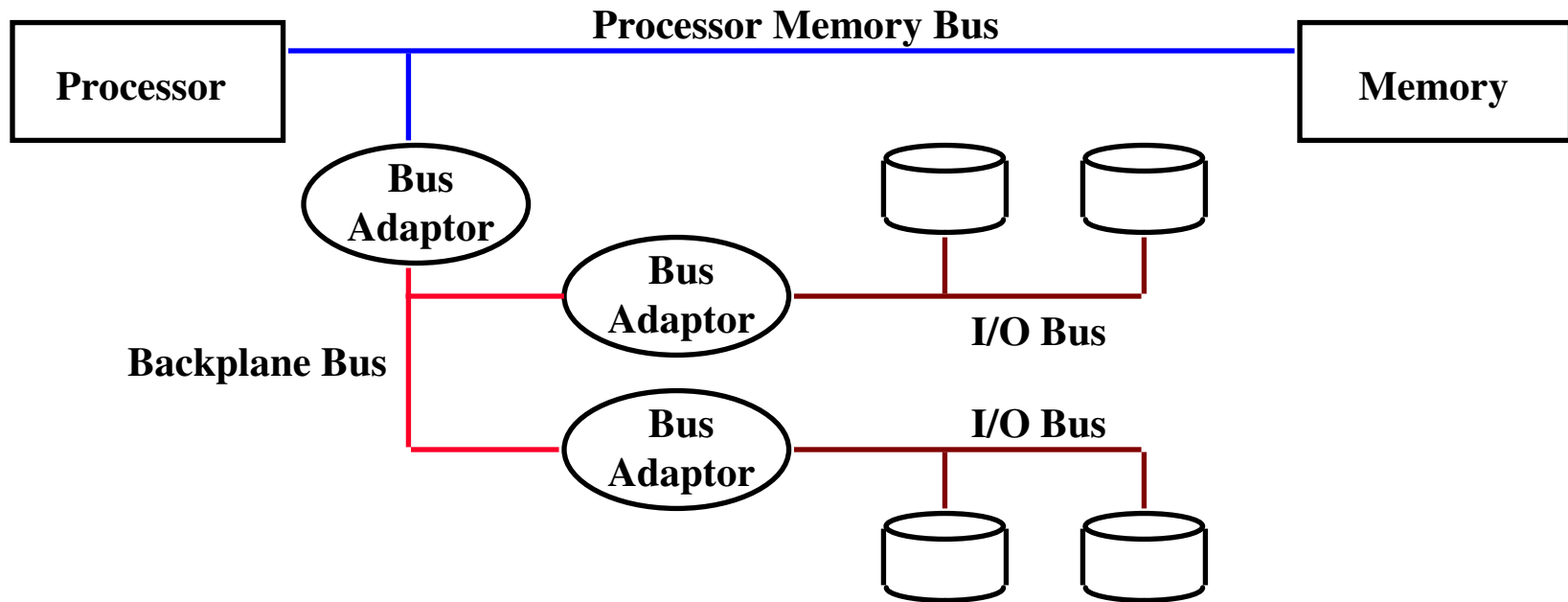
## Systemas com Dois Barramentos



- **I/O buses** ligados ao barramento processador-memória via adaptadors:
  - **Processor-memory Bus**: prioritariamente para o tráfego processador-memória
  - **I/O buses**: provê slots para expansão para **I/O devices**
- **Apple Macintosh-II**
  - **NuBus**: Processador, memória, e uns poucos (selecionados) dispositivos de I/O
  - **SCCI Bus**: para os outros dispositivos de I/O

# Barramentos

## Systemas com Três Barramentos

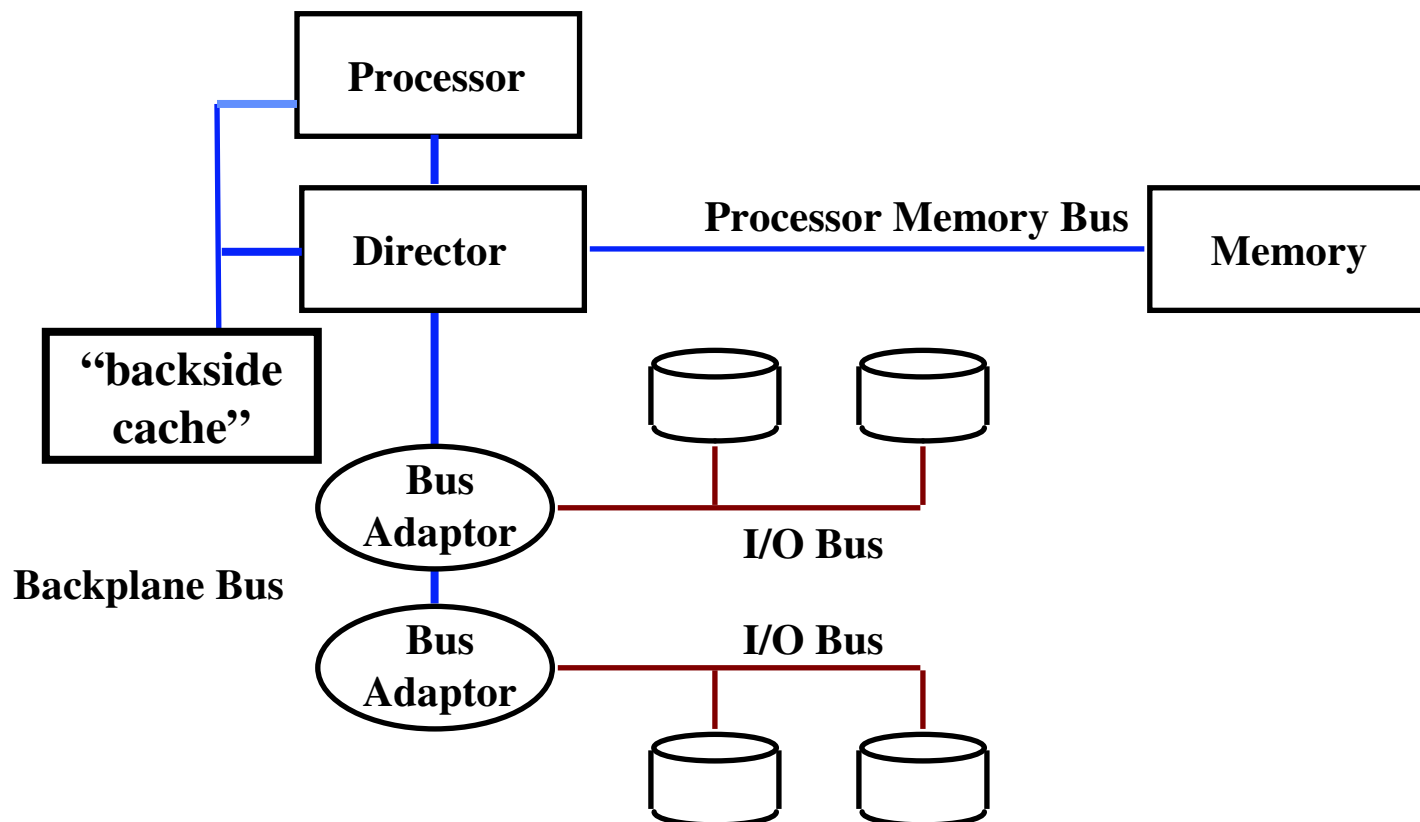


- Um pequeno **backplane bus** é ligado ao **processor-memory bus**
  - **Processor-memory bus** é dedicado ao tráfego processador-memória
  - **I/O buses** são conectados ao **backplane bus**
- **Vantagem:** A carga no **processor-memory bus** é reduzida



# Barramentos

## North/South Bridge Architectures: Busses Separados

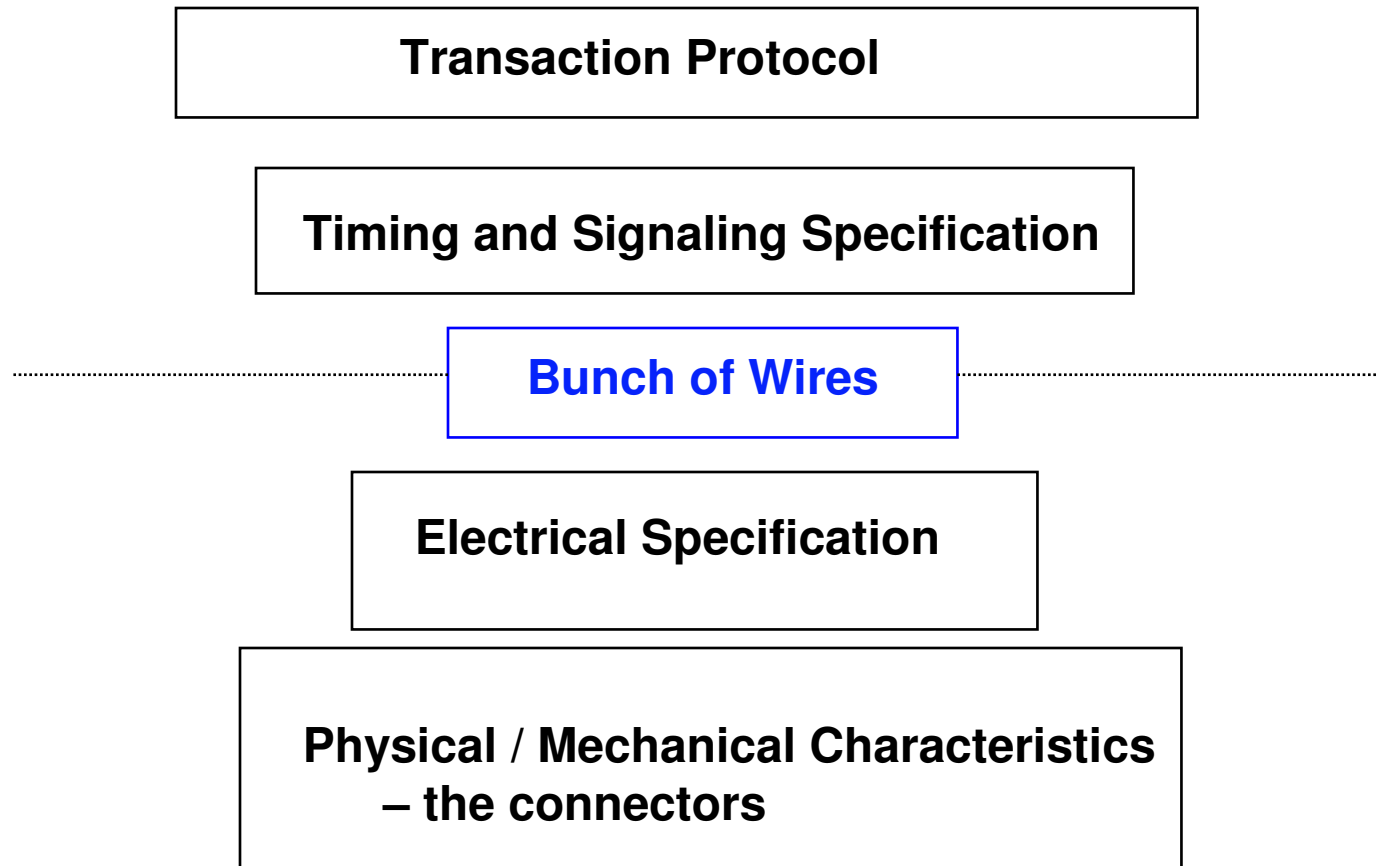


- **Conjunto Separado de pinos para diferentes funções**
  - **Memory bus**; **Caches**; **Graphics bus** (para **fast frame buffer**)
  - **I/O busses** são conectados ao **backplane bus**
- **Vantagens:**
  - Os barramentos podem operar em diferentes velocidades

Paulo G. Oliveira, 2000. Menos sobre cargas nos barramentos, acessos paralelos

# Barramentos

## O que define um Barramento?

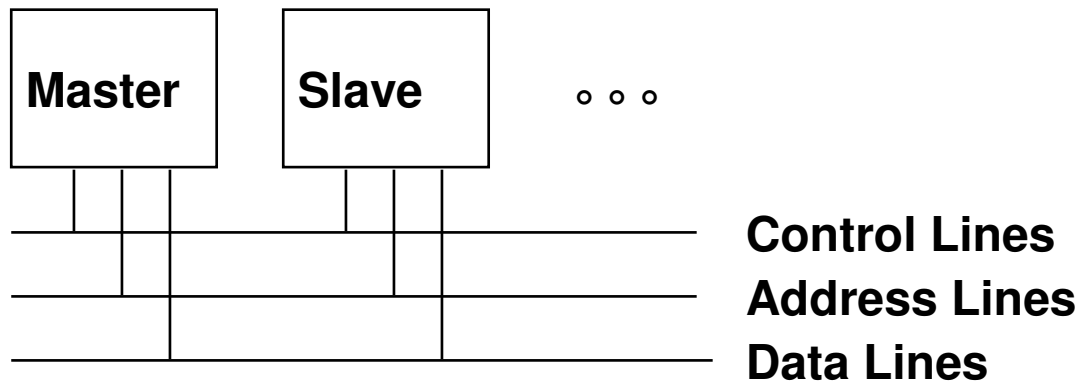


# Barramentos

## Síncronos e Assíncronos

- **Synchronous Bus:**
  - Inclui um **clock** nas linhas de controle
  - Protocolo de comunicação fixo baseado no clock
  - Vantagens: envolve muito menos lógica e pode operar em altas velocidades
  - Desvantagens:
    - » Todo dispositivo no barramento deve operar no mesmo **clock rate**
    - » Para evitar **clock skew**, os barramentos não podem ser longos se são rápidos
- **Asynchronous Bus:**
  - Não usam sinal de **clock**
  - Podem acomodar uma grande variedade de dispositivos
  - Podem ser longos sem se preocupar com **clock skew**
  - Requer um protocolo de **handshaking**

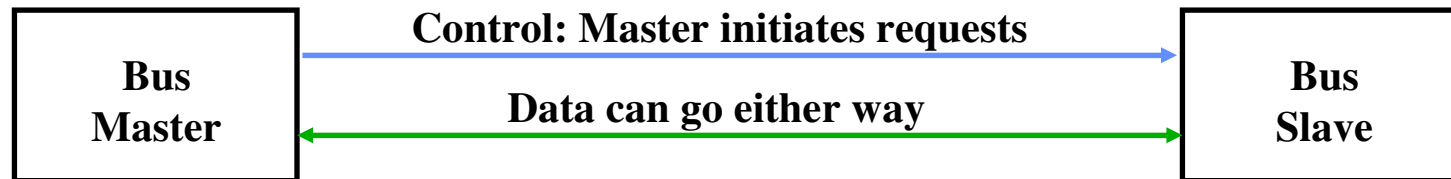
# Barramentos



- **Bus Master:** tem habilidade de controlar o barramento, inicia as transações
- **Bus Slave:** módulo ativado por uma transação
- **Bus Communication Protocol:** especificação de uma seqüência de eventos e timing requeridos em uma transferência de informação.
- **Asynchronous Bus Transfers:** linhas de controle (req, ack) servem para realizar o seqüenciamento.
- **Synchronous Bus Transfers:** a seqüência é relativa a um clock comum.

# Barramentos

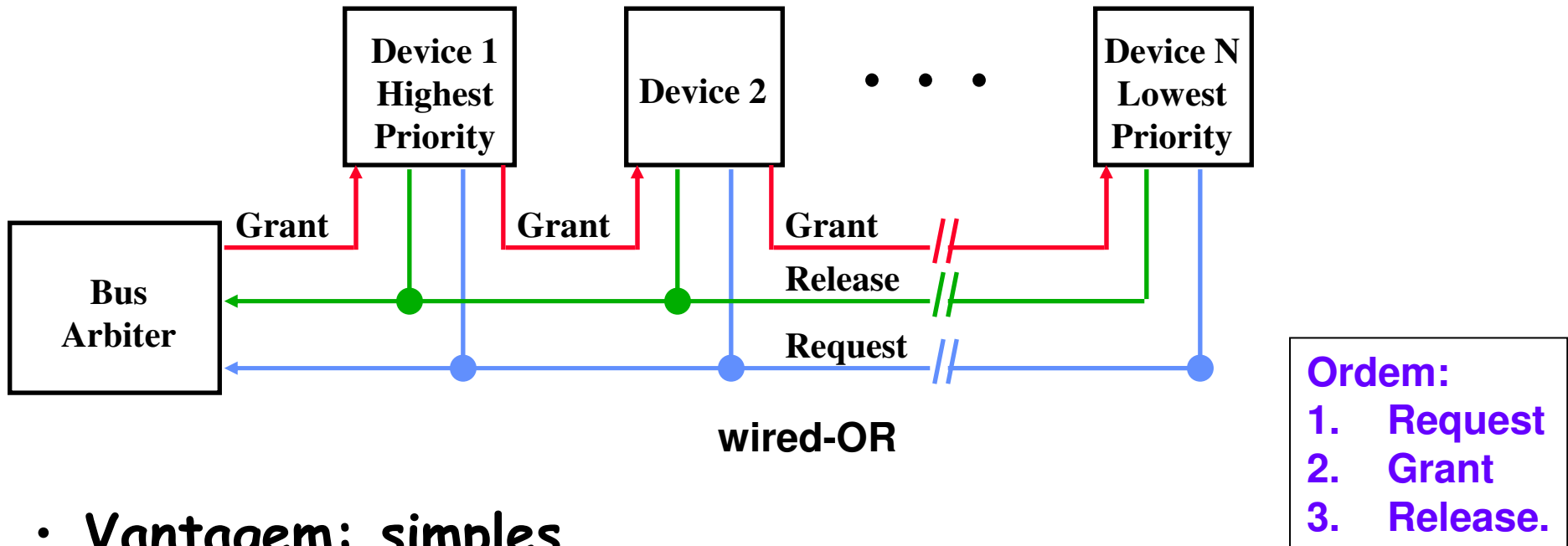
## Arbitragem: Obtenção de Acesso



- Uma das questões mais importantes em **bus design**:
  - Como o barramento é reservado por um dispositivo que o quer usar?
- O Caos é evitado pelo arranjo master-slave :
  - Somente o **bus master** pode controlar o acesso ao barramento:
    - » Ele inicia e controla todass as requisições do barramento
  - Um **bus slave** responde a requisições de leitura e/ou escrita
- Sistema mais simples:
  - O Processador é o único bus master
  - Toda **bus requests** deve ser controlada pelo processador
  - Maior desvantagem: o processador participa em todass ass transações

# Barramentos

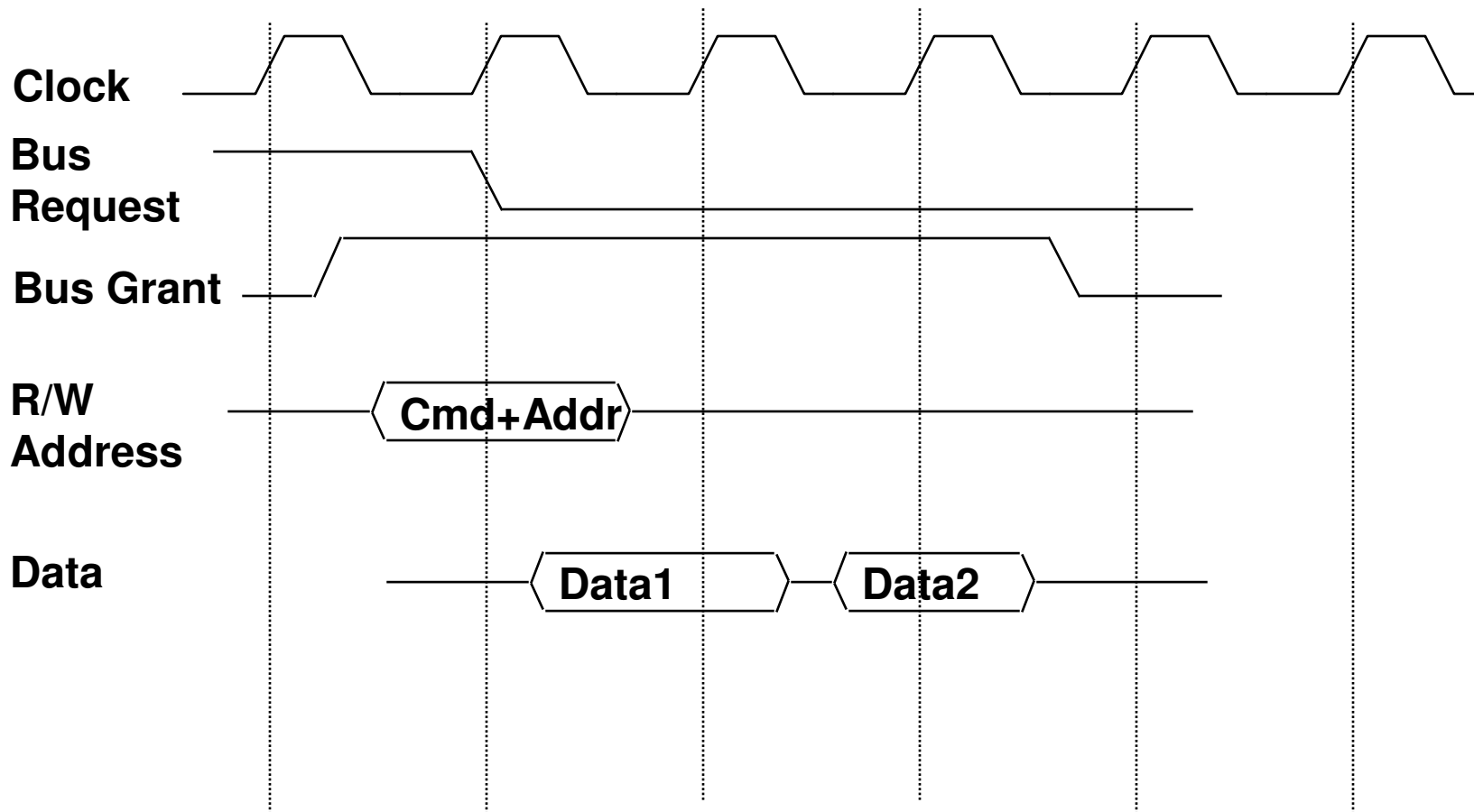
## Arbitragem: Daisy Chain



- Vantagem: simples
- Desvantagens:
  - Não pode garantir justiça:  
Um dispositivo de baixa prioridade pode ficar bloqueado indefinidamente
  - O uso do sinal **daisy chain grant** também limita a velocidade do barramento

# Barramentos

## Um Protocolo Síncrono Simples

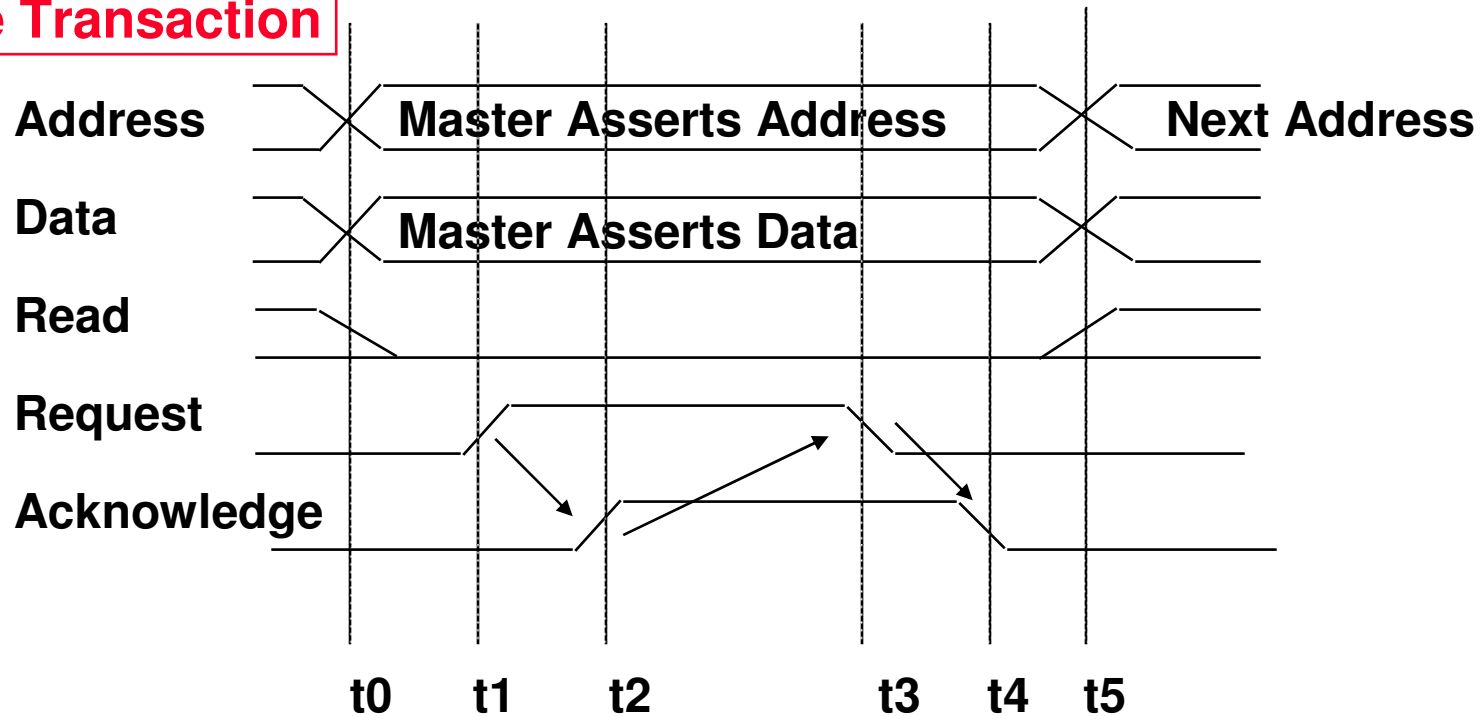


- Os **memory busses** são mais complexos que isso
  - memória (**slave**) pode levar um certo tempo para responder
  - Pode necessitar controlar o **data rate**

# Barramentos

## Protocolo Handshake Assíncrono (4-fases)

### Write Transaction



t0: Master tem o controle e **asserts** o endereço, direção (not read), dado. Espera uma quantidade pré-definida de tempo para o Slave decodificar o alvo

t1: Master **asserts** a **request line**

t2: Slave **asserts** o **ack**, indicando que recebeu o dado

t3: Master libera o **request**

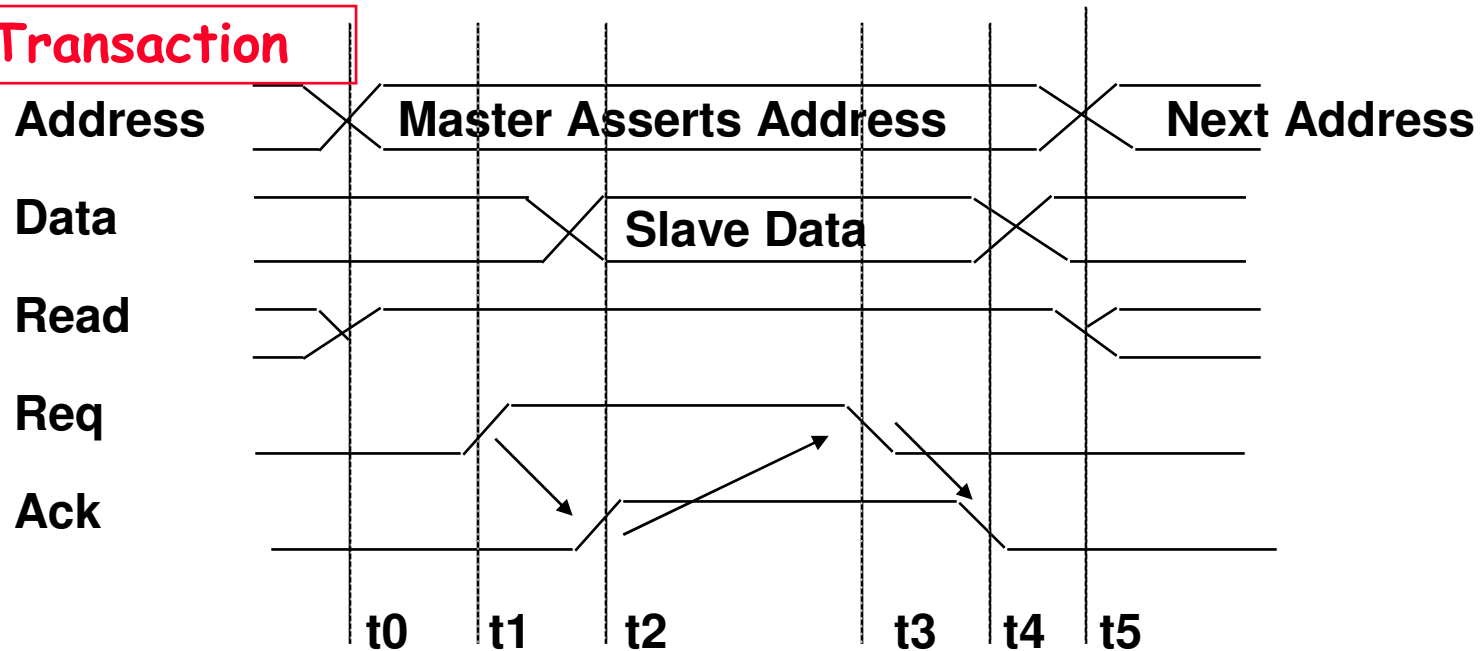
t4: Slave libera o **acknowledge**



# Barramentos

## Protocolo Handshake Assíncrono (4-fases)

### Read Transaction



t0: Master obtém o controle e **asserts** o endereço e a direção. Espera uma quantidade pré-definida de tempo para o Slave decodificar o alvo

t1: Master **asserts** a **request line**

t2: Slave **asserts** o **ack**, indicando que está pronto para transmitir o dado

t3: Master libera o **req**, o dado é recebido

t4: Slave libera o **ack**

# Barramentos

## Exemplo: PCI Read/Write Transaction

- Todos os Sinais são amostrados na borda de subida (**rising edge**)
- Arbitragem Centralizada e Paralela
  - Sobreposição com transações prévias
- Todas as transferências são **Burst** (ilimitada)
- **Address phase** tem início pelo **asserting FRAME#**
- Próximo ciclo **asserts cmd** e **address**
- A transferência de Dados ocorre quando
  - **IRDY# asserted** pelo master quando pronto para transferir dados
  - **TRDY# asserted** pelo **target** quando pronto para transferir dados
  - Transfere quando ambos **asserted** e **rising edge**
- **FRAME# de-asserted** quando master tem a intenção de **completar somente mais uma transferência de dados**

# Barramentos

## Exemplo: PCI Read Transaction

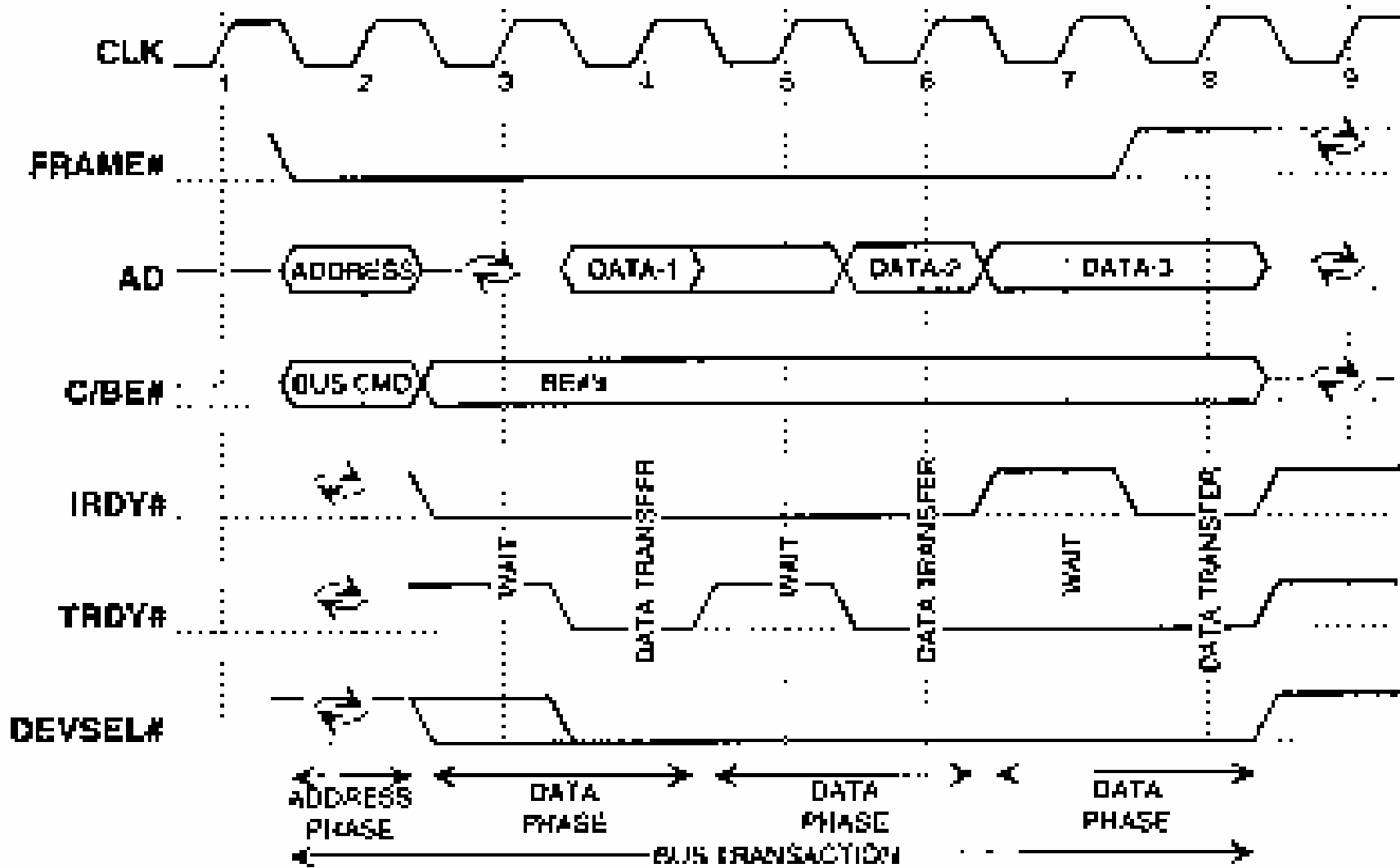
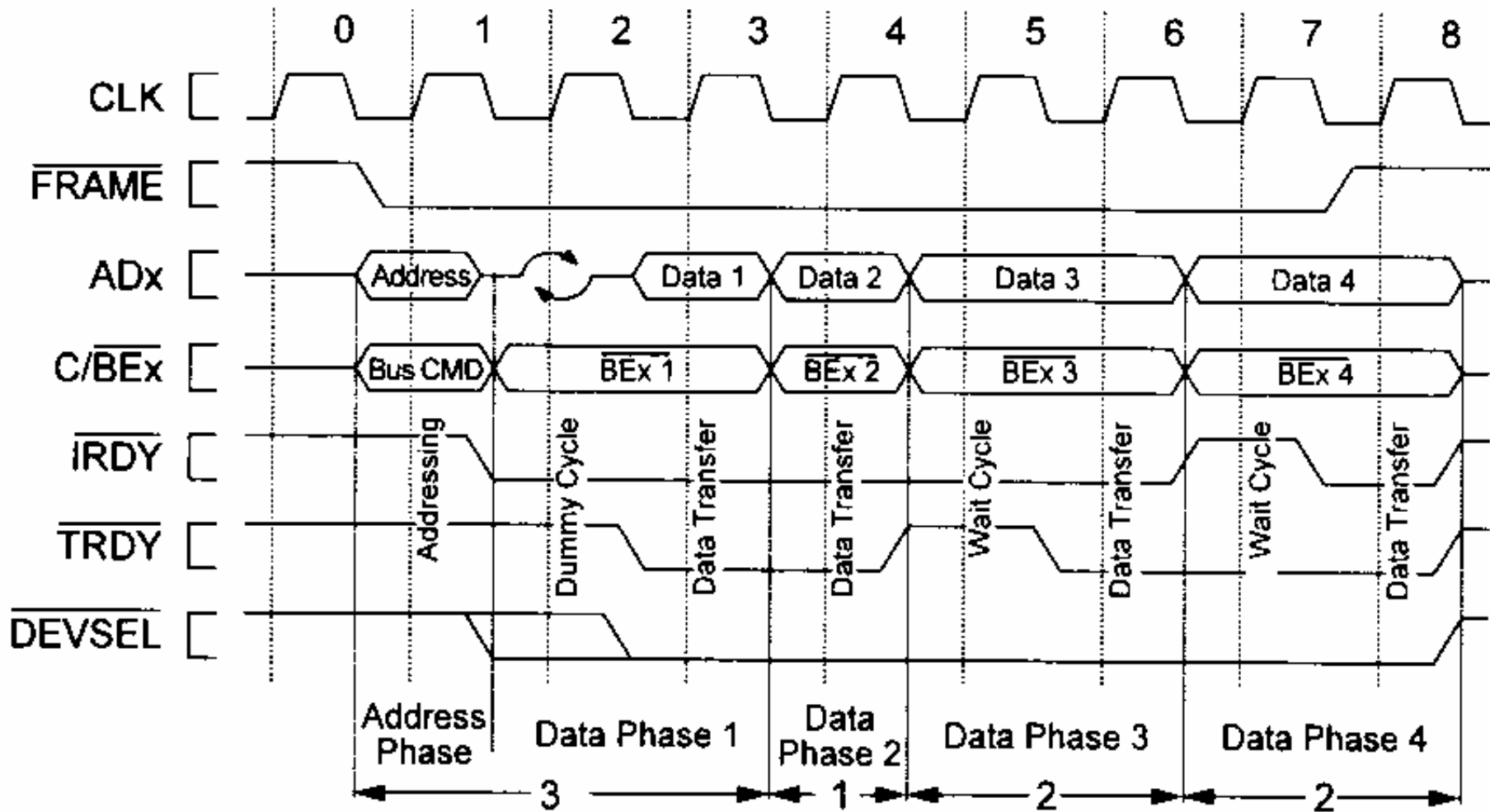


Figure 3-1: Busic Read Operation

# The PCI read transfer burst



# Interface: Processador & I/O

- A interface consiste em informar ao dispositivo como e qual operação será realizada:
  - Read ou Write
  - Tamanho da transferência
  - Localização no dispositivo
  - Localização na memória
- Acionar (triggering) o dispositivo para iniciar a operação
- Quando terminar a operação, o dispositivo interrompe o processador.

# Interface: Processador & I/O

## Dois tipos de mapeamento:

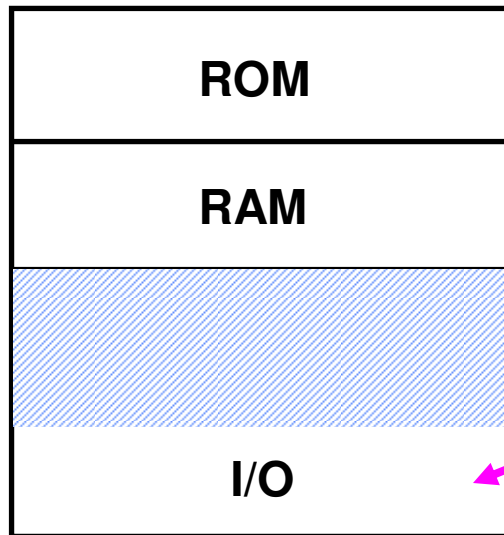
- **Mapeamento em I/O**

- Instruções de I/O específicas
- Ex.1: `LDD R0,D,P` <-- Load R0 com o conteúdo do dispositivo D, porta P
- Ex.2: `IN AX,0f1`  
`OUT AX,0f2`

- **Mapeamento em Memória**

- Não existem instruções especiais de I/O
- Ex.1: `LD R0,Mem1` <-- Load R0 com o conteúdo do dispositivo D, porta P.

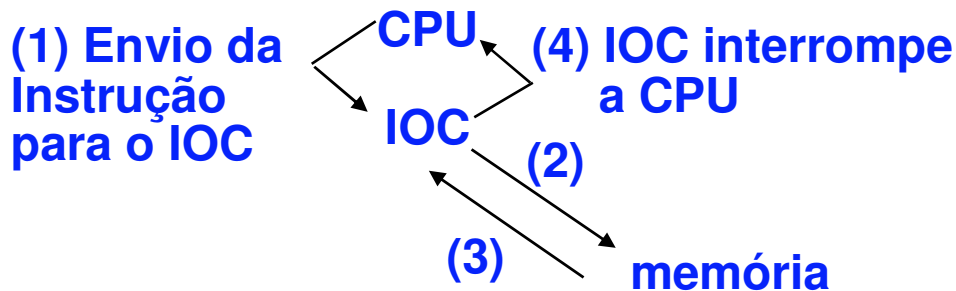
# Interface: Processador & I/O



Virtual Memory  
Aponta para o espaço de IO

Dispositivo alvo

Endereço da porta



Transferências Dispositivo  
para/de Memória são controladas  
Diretamente pelo IOC.

IOP similar à memória para  
comandos



Oper.

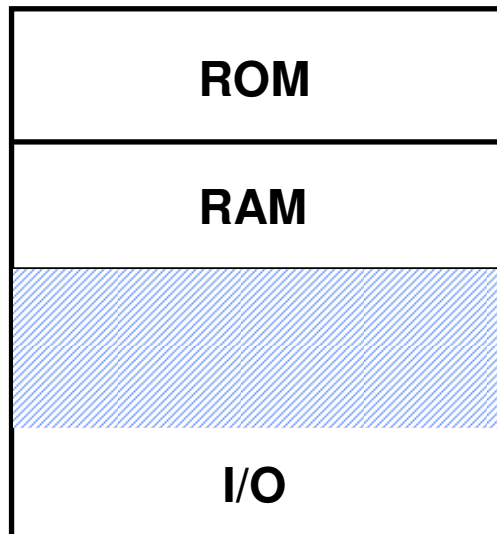
Onde  
colocar

Requisições  
especiais

quantidade

# Memory Mapped I/O

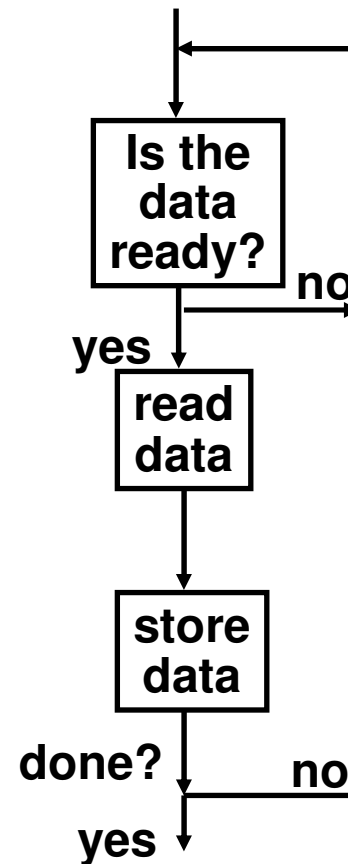
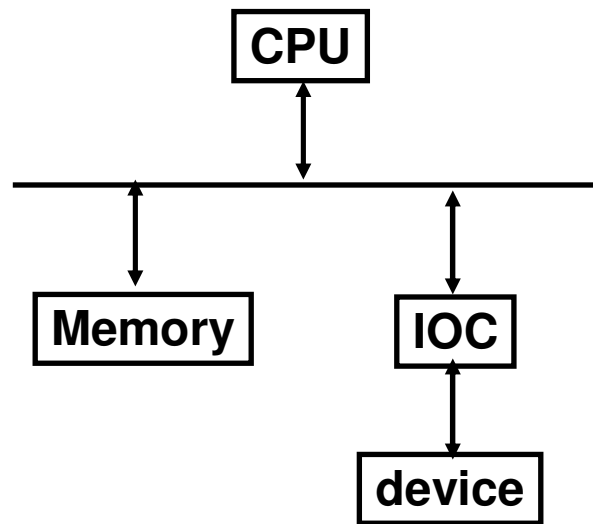
Alguns endereços físicos são usados aparte.  
Não existe memória real nesses endereços.  
No lugar da memória, quando o processador acessa  
esses endereços, ele envia comandos ao processador  
de IO.





# Métodos de Transferências: Programmed I/O (Polling)

**busy wait loop**  
(Não é eficiente)

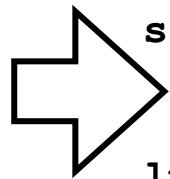


# Métodos de Transferências: Interrupção

- Uma interrupção de I/O é similar à uma exceção a menos de:
  - Uma interrupção de I/O é assíncrona
- Uma interrupção de I/O é assíncrona com respeito à execução das instruções:
  - Interrupção de I/O não é associada a alguma instrução
  - Interrupção de I/O não impede que qualquer instrução seja completada
  - » Deve haver um ponto conveniente para aceitar uma interrupção
- Interrupção de I/O é mais complexa que exceção:
  - Necessita identificar o dispositivo que gerou o pedido
  - Pedidos diferentes de interrupções podem ter diferentes prioridades :
  - » Pedidos de Interrupções precisam serem priorizados

# Métodos de Transferências: Interrupção

Interrupção Externa



```

...
add    $r1, $r2, $r3
subi   $r4, $r1, #4
slli   $r4, $r4, #2

```

**Interrupção!!!**

```

lw     $r2, 0($r4)
lw     $r3, 4($r4)
add    $r2, $r2, $r3
sw     8($r4), $r2
...

```

Salva PC  
Desabilita todas Intrs  
Modo Supervisor

Restaura o PC  
Modo Usuário

*Habilita as Intrps de maior prioridade*  
*Sava os registradores*

```

...
lw     $r1, 20($r0)
lw     $r2, 0($r1)
addi   $r3, $r0, #5
sw     $r3, 0($r1)
...

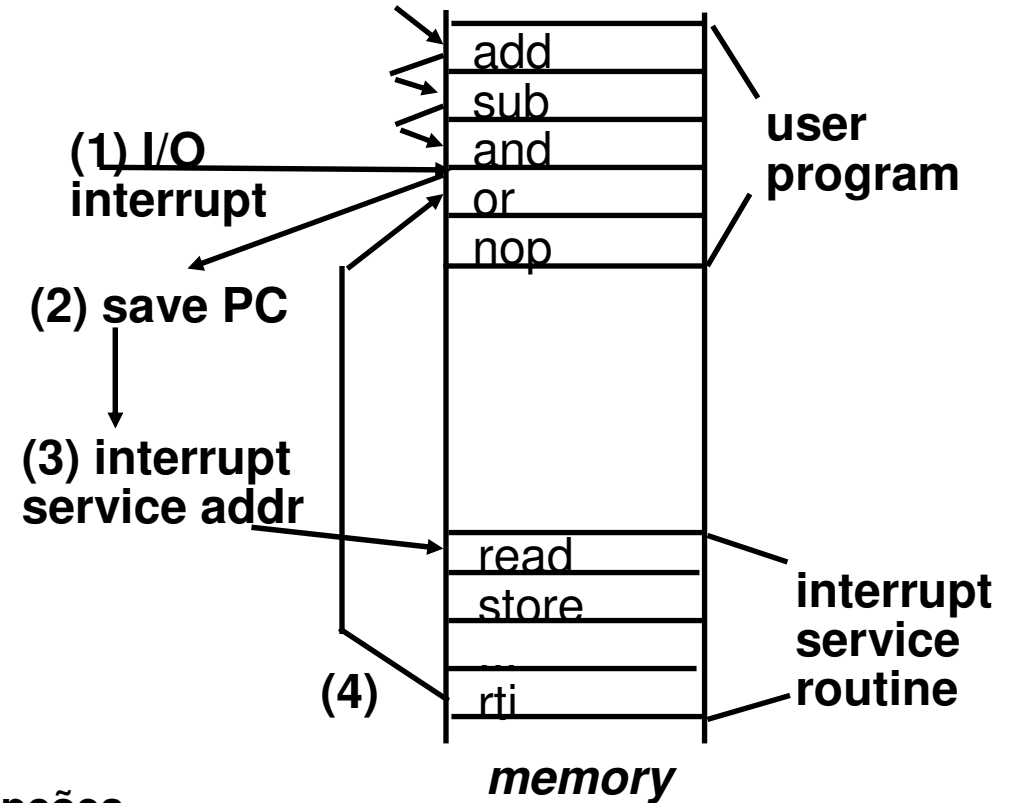
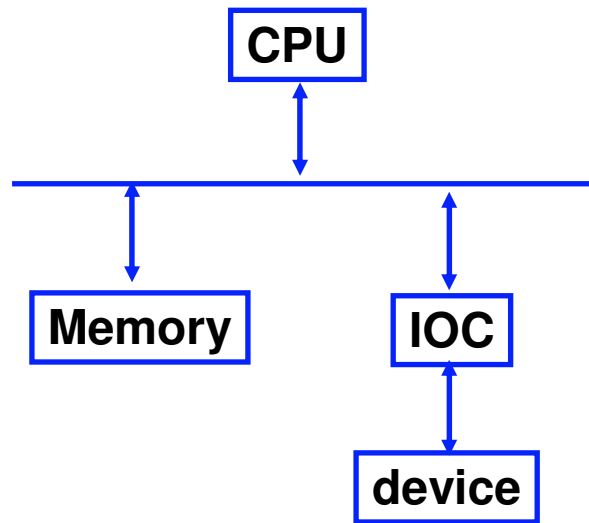
```

*Restaura os registradores*  
*Desabilita as Intrps*  
RTI

“Interrupt Handler”

- **Vantagem:**
  - O programa do usuário em progresso é o único a ser parado
- **Desvantagem, é necessário hw especial para:**
  - Causar uma interrupção (I/O device)
  - Detetar uma interrupção (processador)
  - **Savar o estado para reassumir a execução após a interrupção (processador)**

# Métodos de Transferências: Interrupt Driven Data Transfer



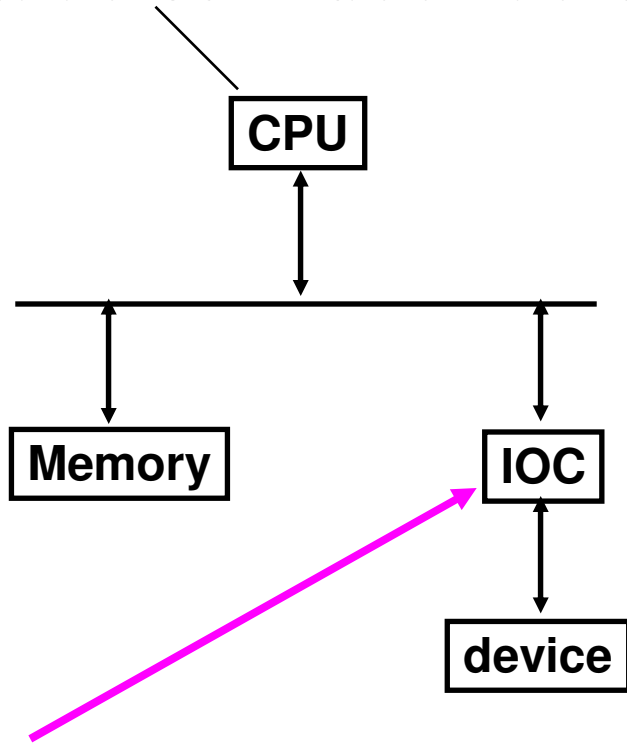
## Exemplo:

1000 transfrs de 1000 bytes cada:  
 1000 interrupções @ 2 µseg por interrupções  
 1000 transfrs por interp @ 98 µseg cada = 0.1 CPU seg

Taxa Transf. = 10 MBytes/sec =>  $0.1 \times 10^{-6}$  seg/byte => 0.1 µseg/byte  
 => 1000 bytes = 100 µseg  
 1000 transfrs x 100 µsecs = 100 ms = 0.1 CPU segs

# Métodos de Transferências: DMA - Direct Memory Access

CPU envia Endereço de início,  
Direção e Tamanho da transferência  
para o IOC. Então envia o "start".



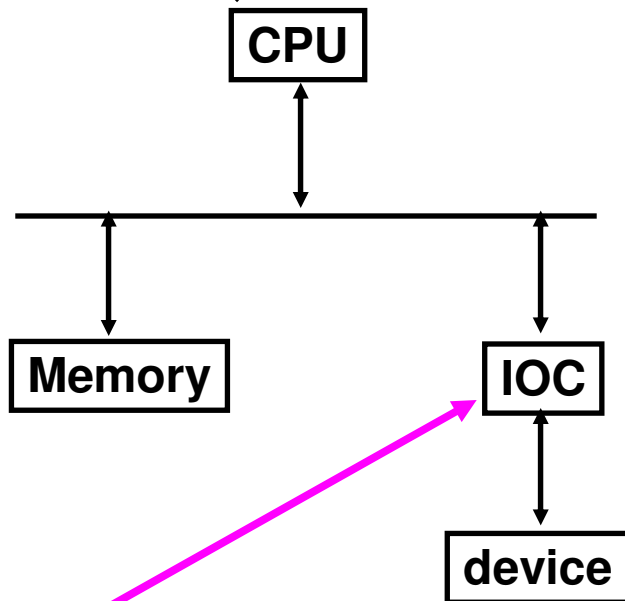
IOC provê sinais de  
Handshake para o  
Controlador do Periférico  
e Endereços e sinais  
de Handshake  
para a Memória.

- **Direct Memory Access (DMA):**
  - Externo à CPU
  - Age como um Master no barramento
  - Transfere blocos de dados para(da) a memória sem intervenção da CPU

# Métodos de Transferências:

## DMA - Direct Memory Access

CPU envia Endereço de início, Direção e Tamanho da Transferência para o IOC. Então envia o "start".



IOC provê sinais de Handshake para o Controlador do Periférico e Endereços e sinais de Handshake para a Memória.

Tempo para transfr. 1000 de 1000 bytes cada:

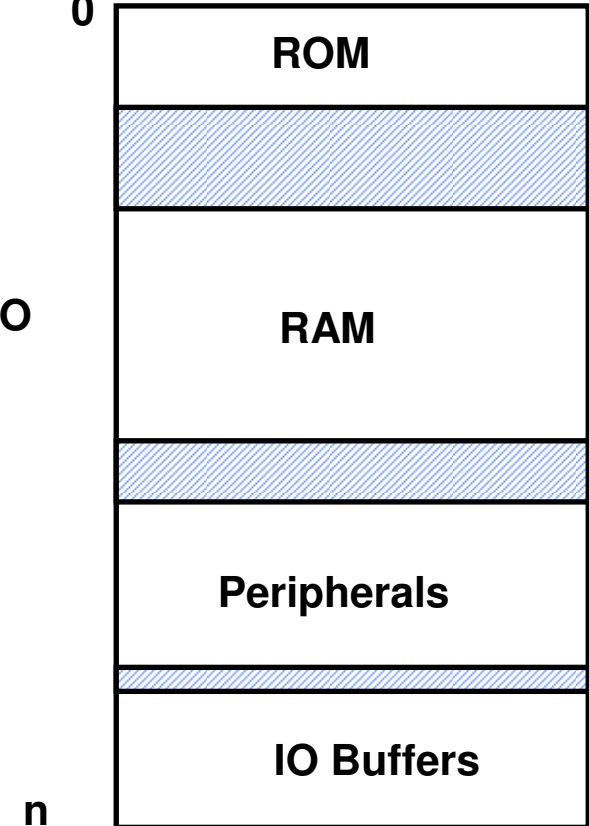
1 DMA seqüência de controle @ 50 µsec

1 interrupção @ 2 µsec

1 seq. do serviço de interrupção @ 48 µsec

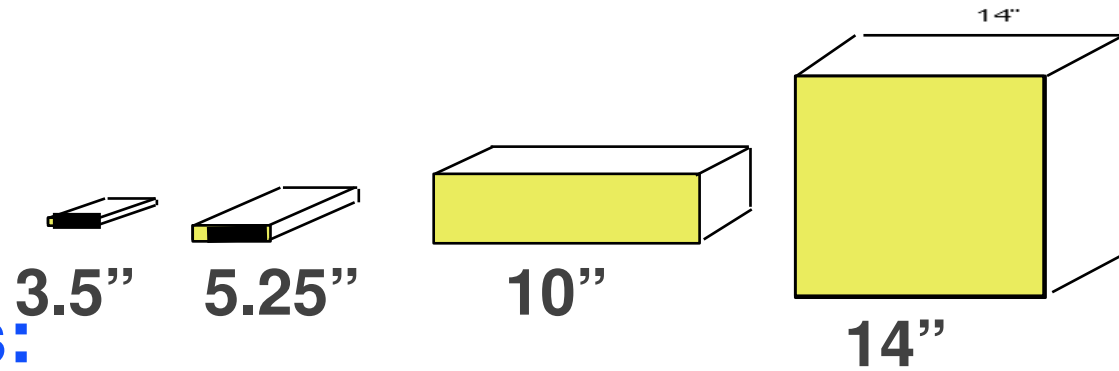
.0001 segundos => tempo de CPU

Memory Mapped I/O



# Uso de Arrays de Pequenos Discos?

Projeto de 4  
Discos  
Convencionais:



Low End → High End

Disk Array:  
1 projeto de  
disco



## Uso de um pequeno no. de discos grandes vs uso de um no. grande de pequenos discos

	IBM 3390K	IBM 3.5" 0061	x70
Capacidade	20 GBytes	320 MBytes	23 GBytes
Volume	97 cu. ft.	0.1 cu. ft.	11 cu. ft. <b>9X</b>
Power	3 KW	11 W	1 KW <b>3X</b>
Data Rate	15 MB/s	1.5 MB/s	120 MB/s <b>8X</b>
I/O Rate	600 I/Os/s	55 I/Os/s	3900 IOs/s <b>6X</b>
MTTF	250 KHrs	50 KHrs	??? Hrs
Custo	\$250K	\$2K	\$150K

Disk Arrays tem potencial para grandes quantidades de dados e I/O rates, alto MB por volume, alto MB por KW, **e confiabilidade?**



# Array: Confiabilidade

- **“Reliability” de N discos = “Reliability” de 1 Disco ÷ N**

**50,000 Horas ÷ 70 discos = 700 horas**

**Disk system MTTF: cai de 6 anos para 1 mês!**

- **Arrays (sem redundâncias) são pouco confiáveis!**

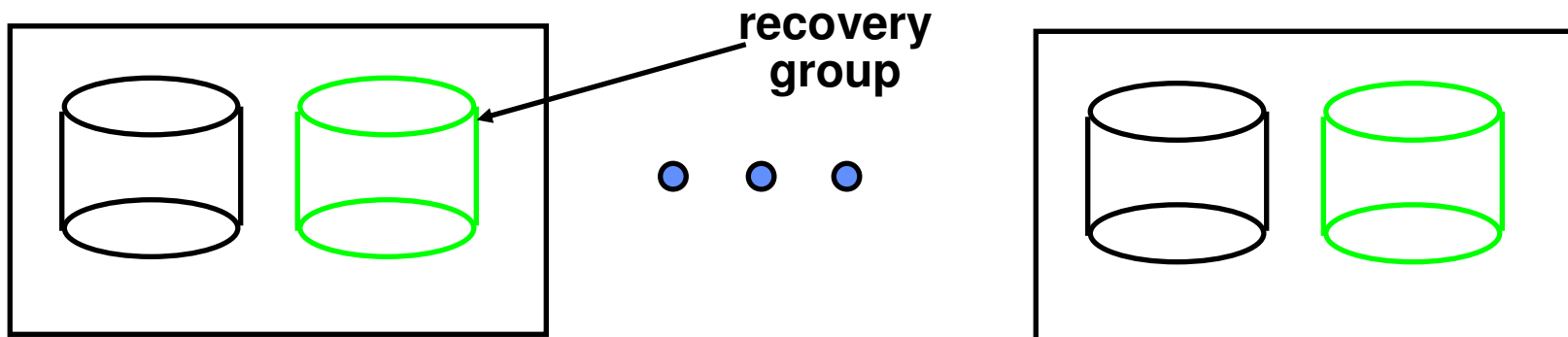
## Redundant Arrays com Discos Baratos

- Os Arquivos divididos e armazenados em múltiplos discos
- Redundância provê alta disponibilidade de dados
  - Disponibilidade: o serviço continua sendo provido mesmo que algum componente falha
- Discos ainda podem falhar
- O Conteúdo pode ser reconstruído a partir dos dados armazenados de forma redundante no array
  - ⇒ Penalidade na capacidade para armazenamento redundante
  - ⇒ Penalidade no Bandwidth para atualizar dados redundantes



# Redundant Arrays of Disks

## RAID 1: Disk Mirroring/Shadowing

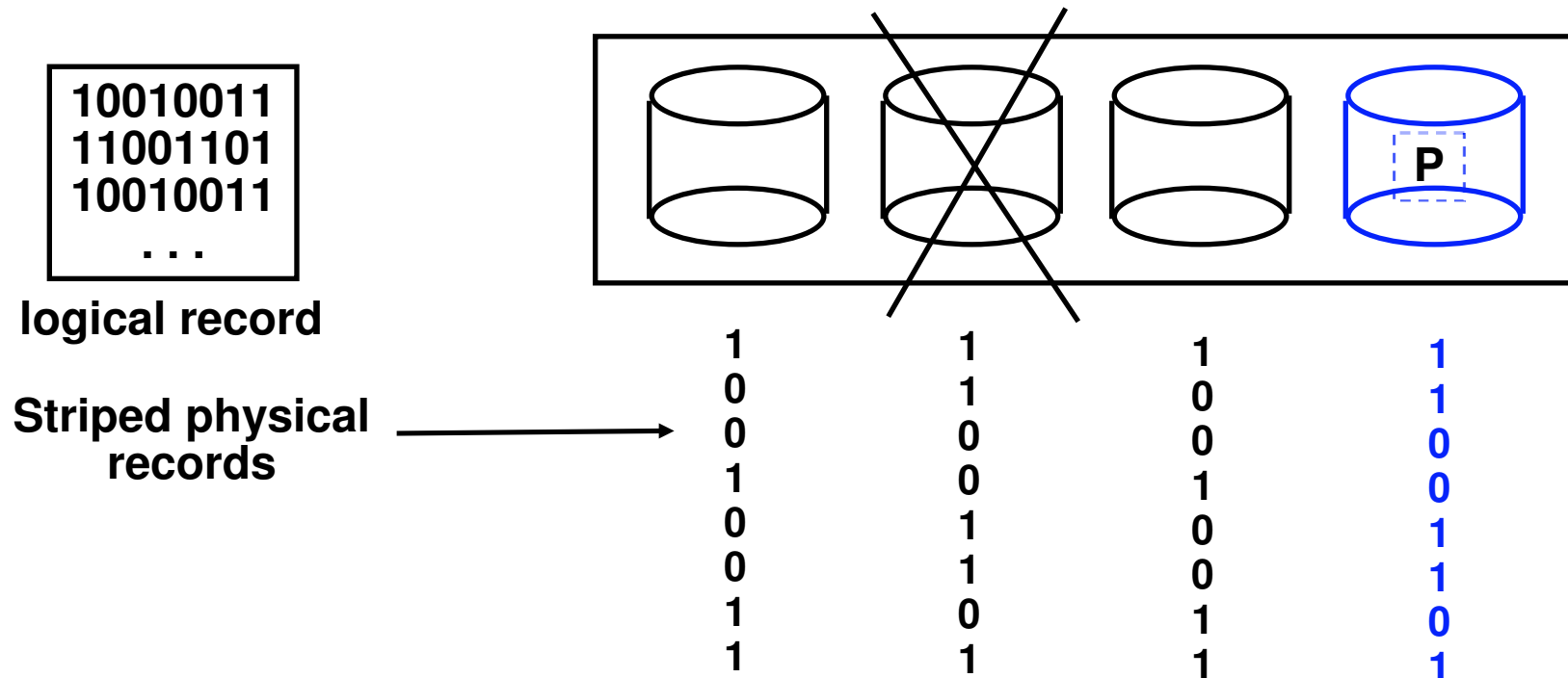


- Cada disco é totalmente duplicado em seu "shadow"  
Proporciona alta disponibilidade
- Bandwidth é sacrificado na escrita:  
Escrita lógica = duas escritas físicas
- Leituras podem ser otimizadas
- Solução mais cara : 100% de overhead na capacidade

*High I/O rate , ambientes com alta disponibilidade*

Redundant Array of Inexpensive Disks  
Independent

# Redundant Arrays of Disks RAID 3: Parity Disk



- Paridade calculada para o grupo de recuperação, protegendo contra falhas nos discos  
33% de custo de capacidade para a paridade nesta configuração  
arrays maiores reduzem o custo de capacidade, decresce a disponibilidade esperada, aumenta o tempo de reconstrução  
Eixos sincronizados

*Aplicações de alto bandwidth: Científicas, Processamento de Imagem*

# RAID 4

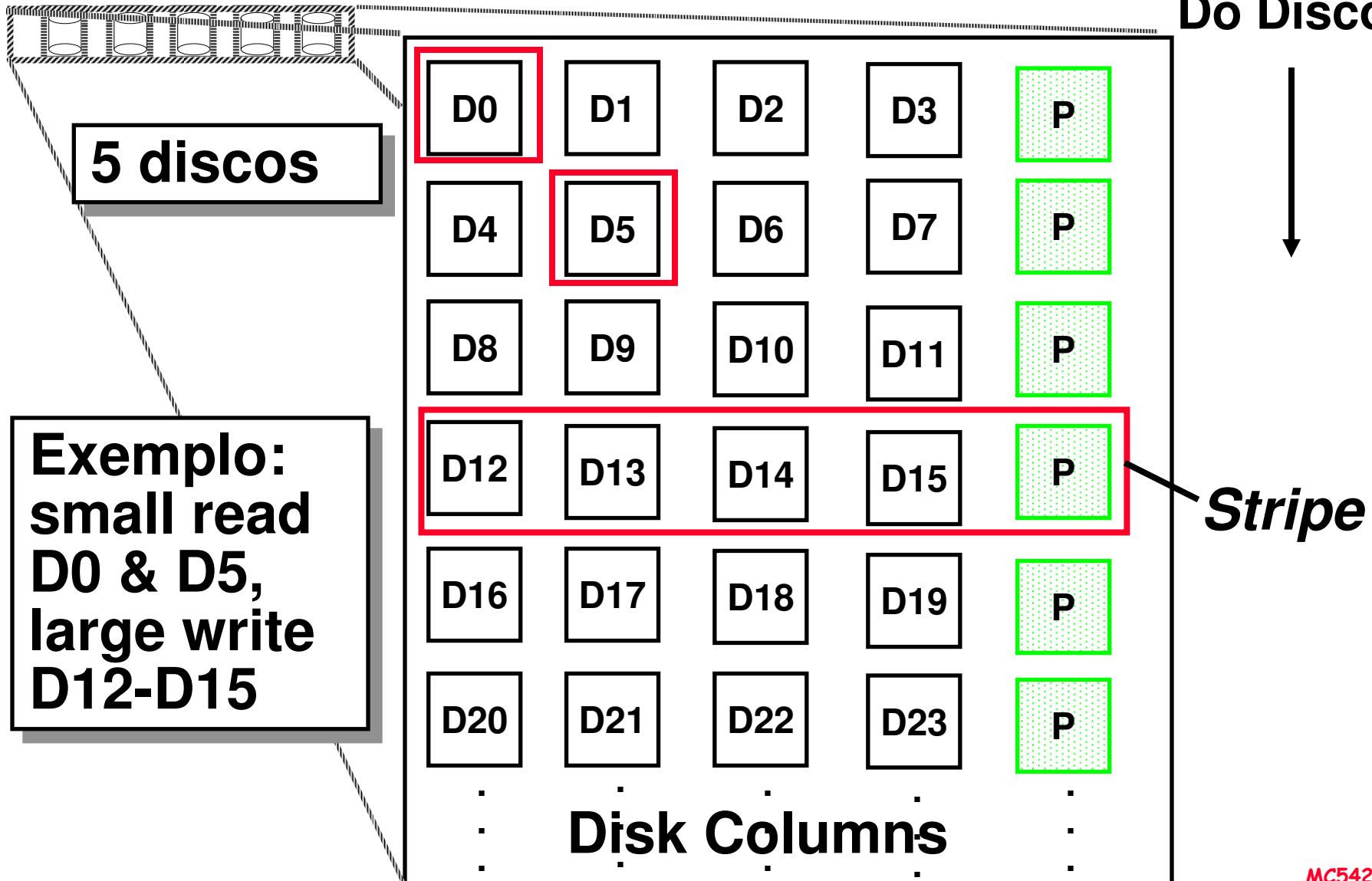
## Inspiração:

- RAID 3 utiliza (confia no) o disco de paridade para recuperar erros na leitura
- Porém, todos setores já possuem um campo para detecção de erros
- Utilizar o campo de detecção de erros para capturar erros na leitura, não o disco de paridade
- Permitir leituras independentes simultâneas em discos diferentes

# Redundant Arrays of Disks

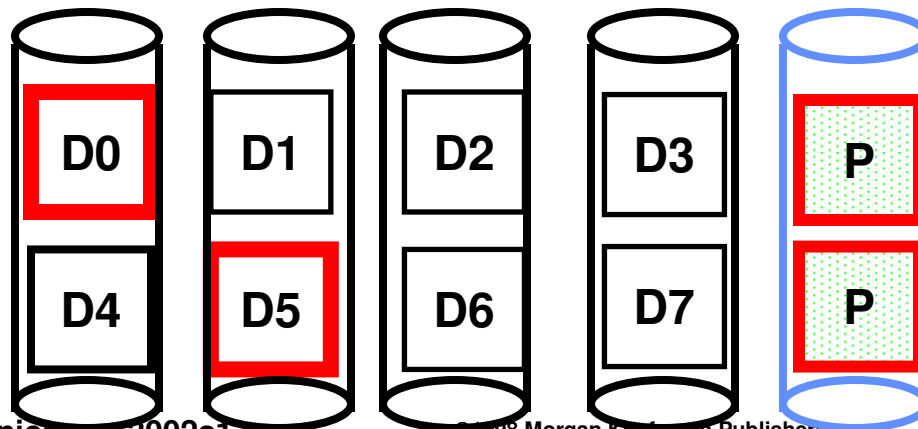
## RAID 4: High I/O Rate Parity

Aumenta o  
Endereço  
Logical  
Do Disco

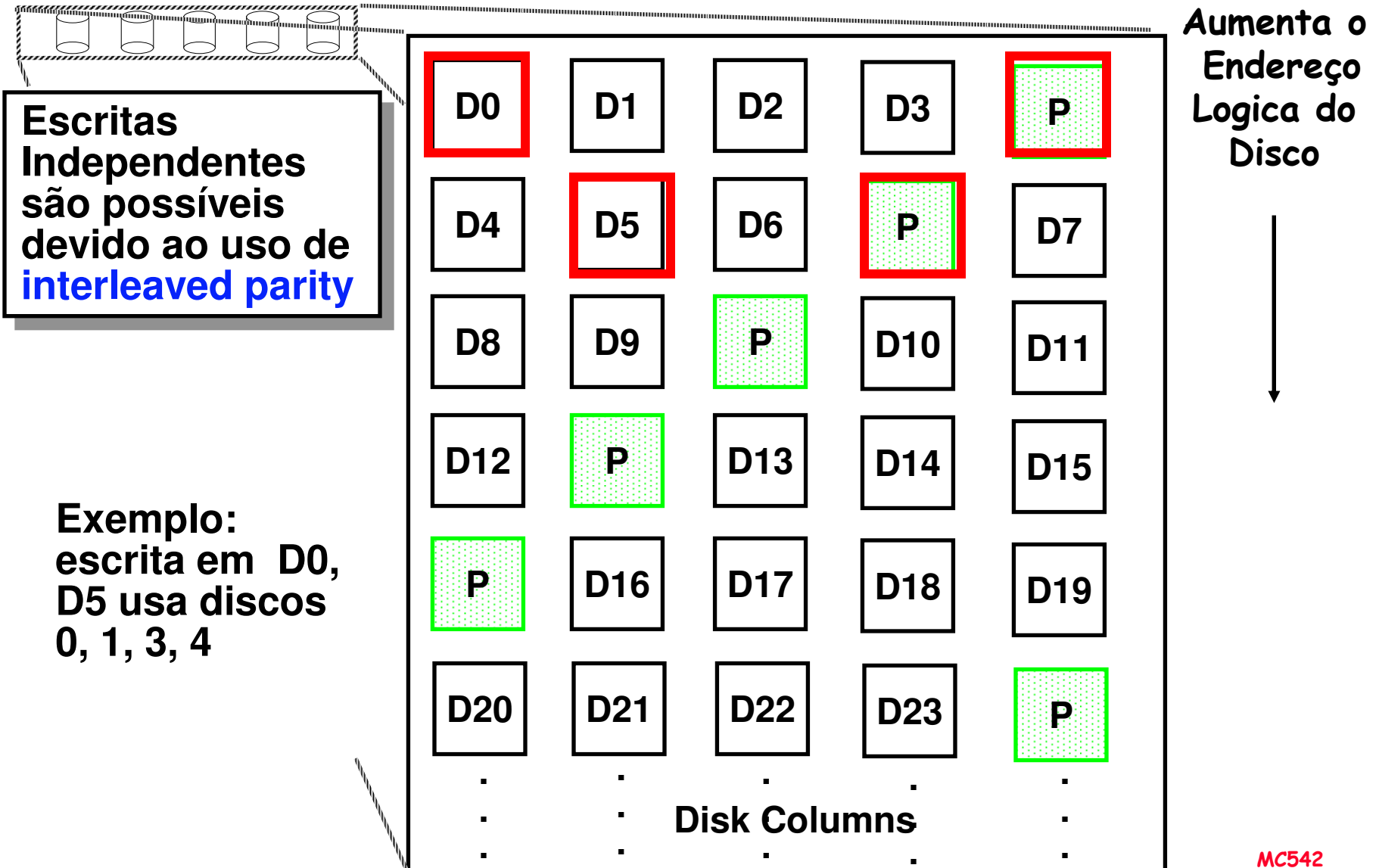


# RAID 5: Inspiração

- RAID 4 trabalha bem para leituras pequenas
- Pequenas escritas (escritas em um disco):
  - Opção 1: lêr outro disco de dados, criar nova soma e escrever no Disco de Paridade
  - Opção 2: uma vez que P tem uma soma antiga, comparar dado velho com dado novo, adicionar somente a diferença em P
- Pequenas escritas são limitadas pelo Disco de Paridade: escrever em D0, D5 em ambos os casos também se escreve no disco P



# Redundant Arrays of Inexpensive Disks RAID 5: High I/O Rate Interleaved Parity

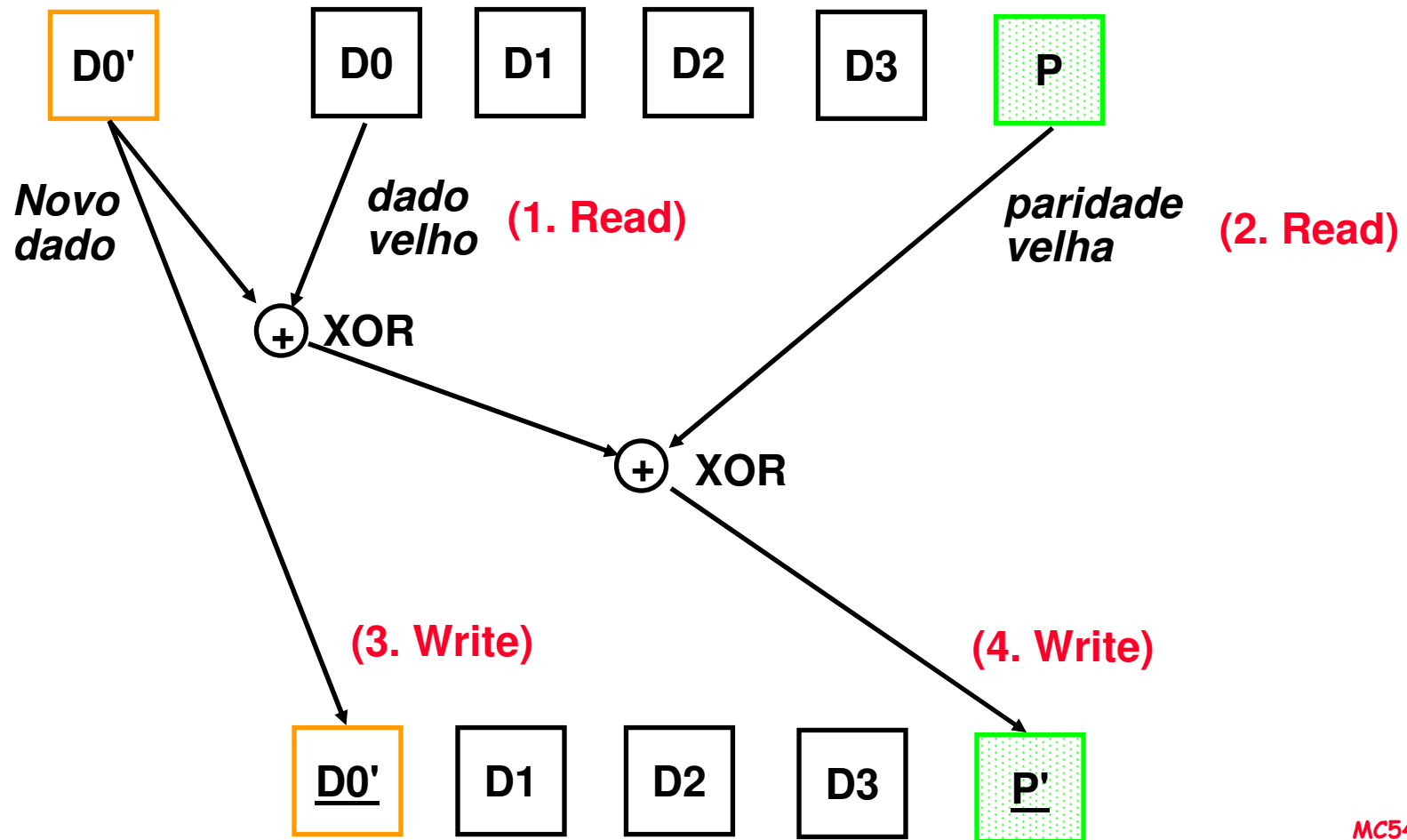




# Problemas com Disk Arrays: Escritas Pequenas

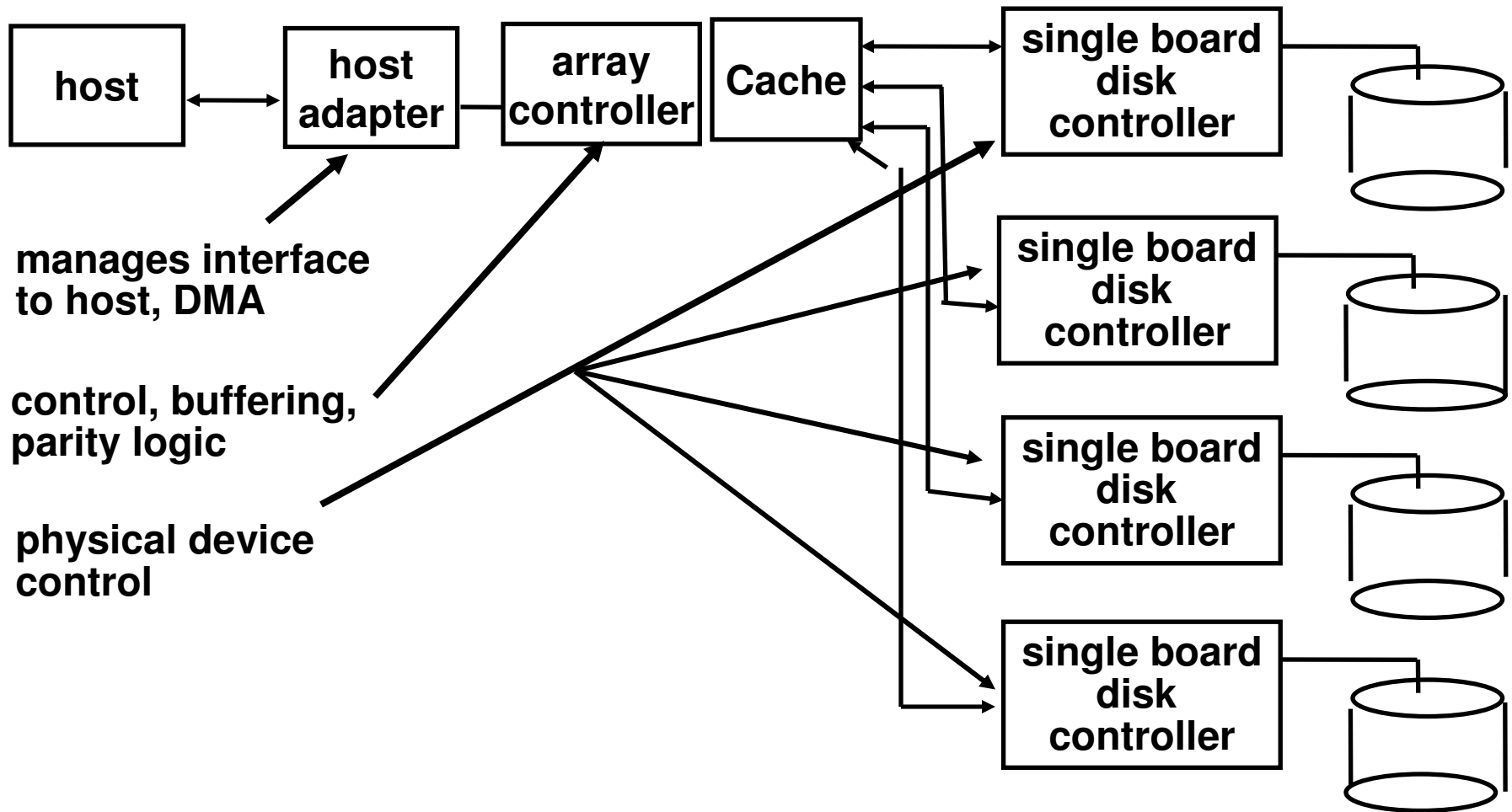
## RAID-5: Algoritmo para escritas Pequenas

1 Escrita Lógica = 2 Leituras Físicas + 2 Escritas Físicas



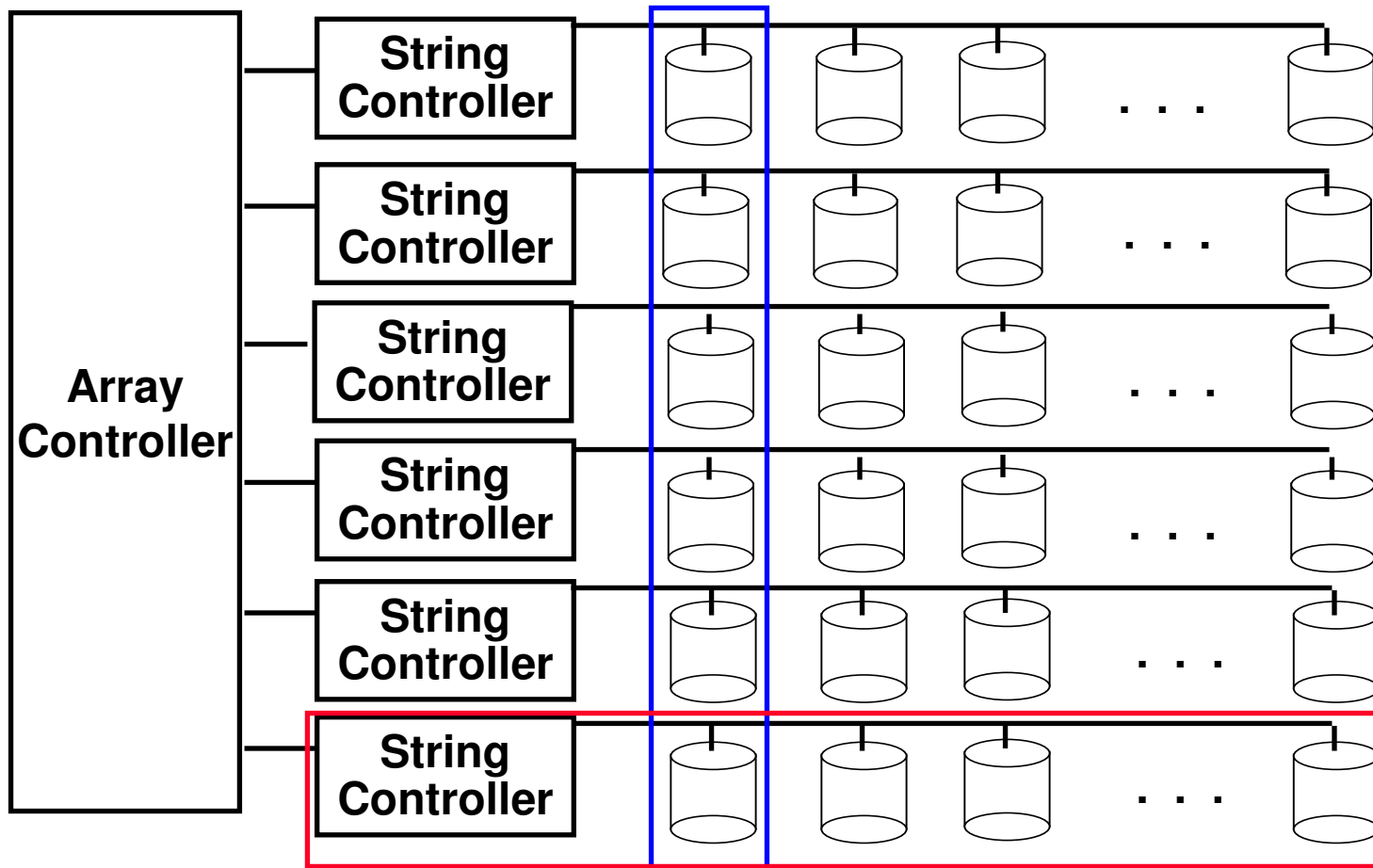
# RAID

## Organização dos Subistemas



# Disponibilidade do Sistema

## RAIDs Ortogonais



**Data Recovery Group:** unidade de redundância de dados

**Redundant Support Components:** fans, power supplies, controller, cables

**End to End Data Integrity:** paridade interna protege os data paths

# Disponibilidade do Sistema

