

Parte 5

Entrada e Saída

Bibliografia

- [1] Miles J. Murdocca e Vincent P. Heuring, “Introdução à Arquitetura de Computadores”
- [2] Andrew S. Tanenbaum, “Modern Operating Systems”
- [3] William Stallings, “Arquitetura e Organização de Computadores”

Características da E/S

- **Velocidade de transferência de dados varia muito**
 - **Muito lento: ex. teclado**
 - **Muito rápido: ex. gráficos escritos no monitor**
- **Atividades de E/S são assíncronas**
 - ***Handshaking* pode ser necessário**
 - Quando o dispositivo está pronto para recepção/envio de dados
- **Qualidade dos dados pode ser incerta**
 - **Mecanismos de detecção/correção**
- **Transferências podem ser interrompidas**
 - **Ex. impressora sem papel**
- ***Device drivers* devem tratar estes problemas**

Comunicação entre Dispositivos

- **N componentes que se comunicam uns com os outros**
 - **Pior caso: $N^2/2$ ligações entre os componentes**
- **Nem todos os dispositivos se comunicam simultaneamente...**
- **Barramento**
 - **Caminho comum que conecta um certo número de dispositivos**

Arquiteturas Simples de Barramentos

- Uma placa-mãe simplificada de um computador pessoal:

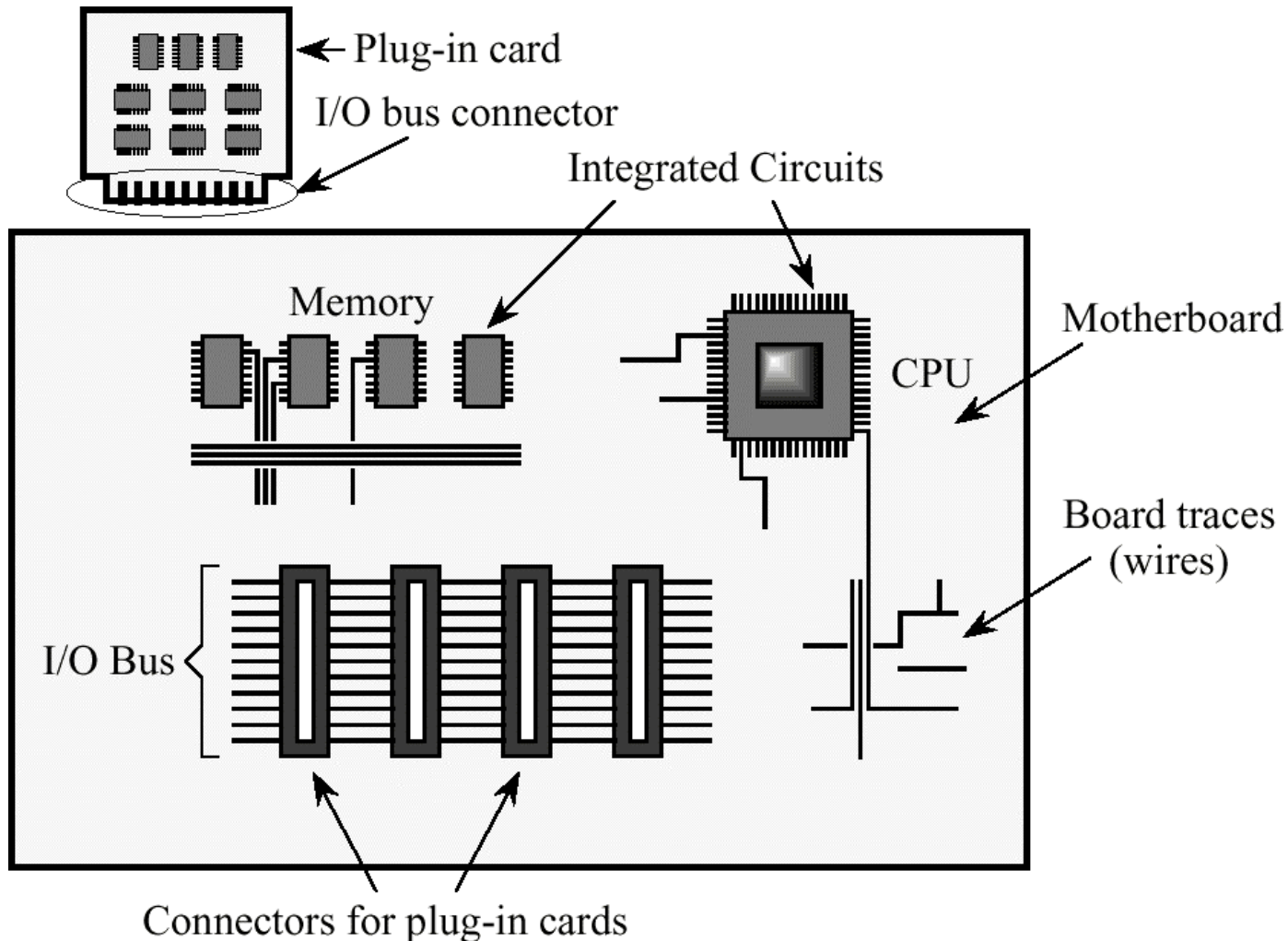
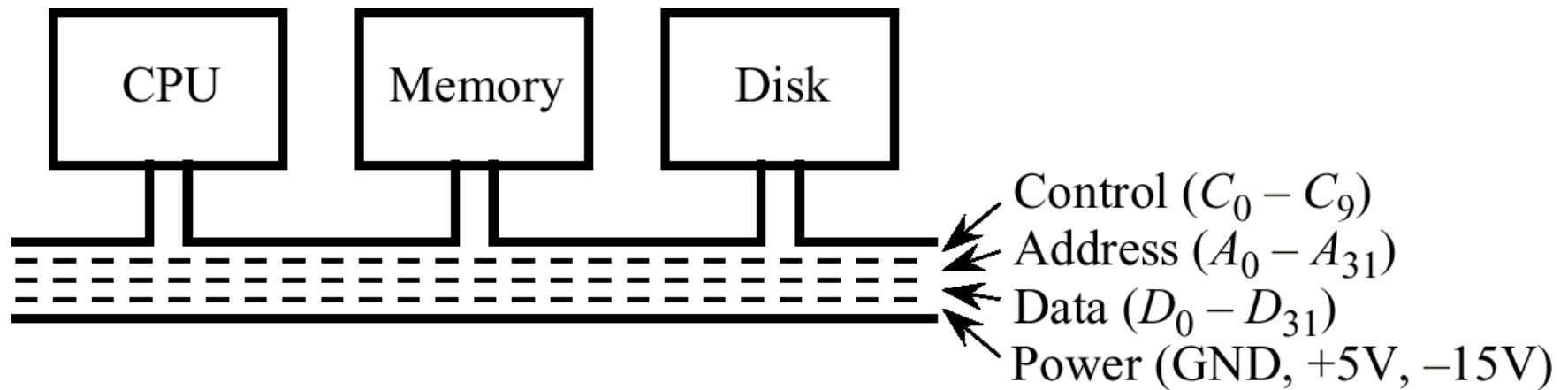


Ilustração Simplificada de um Barramento



Estrutura do Barramento

- **Parte física (fios, conectores) + Protocolo de barramento**
- **Fios são compartilhados**
 - **Somente um dispositivo pode enviar dados em um momento**
 - **Todos escutam, mas em geral só um lê os dados**
- **Barramento controlado por um mestre**
 - **Outros dispositivos são escravos**

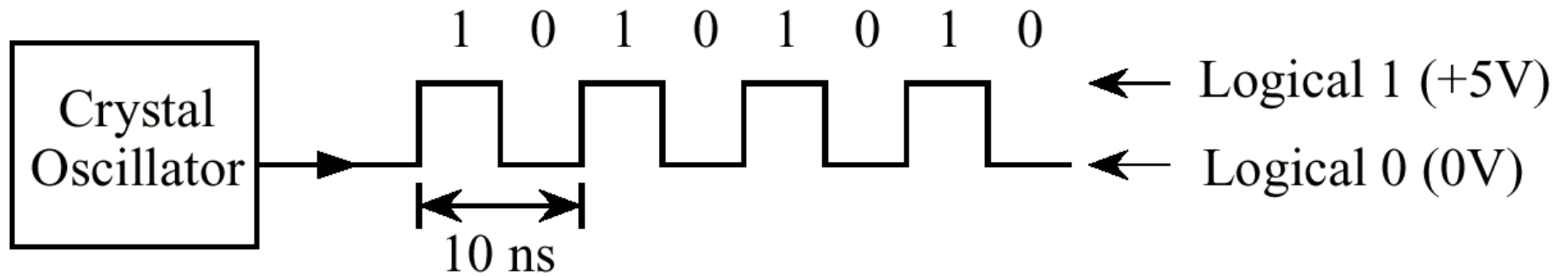
Uso de Barramentos

- **Vantagens**
 - **Elimina a necessidade de conectar cada dispositivo com todos os outros**
- **Desvantagens**
 - **Perda de velocidade (mestre/escravo)**
 - **Tempo envolvido no protocolo**
 - **Problemas de escalabilidade**
 - **Ex. fan-in / fan-out**

Tipos de Barramentos

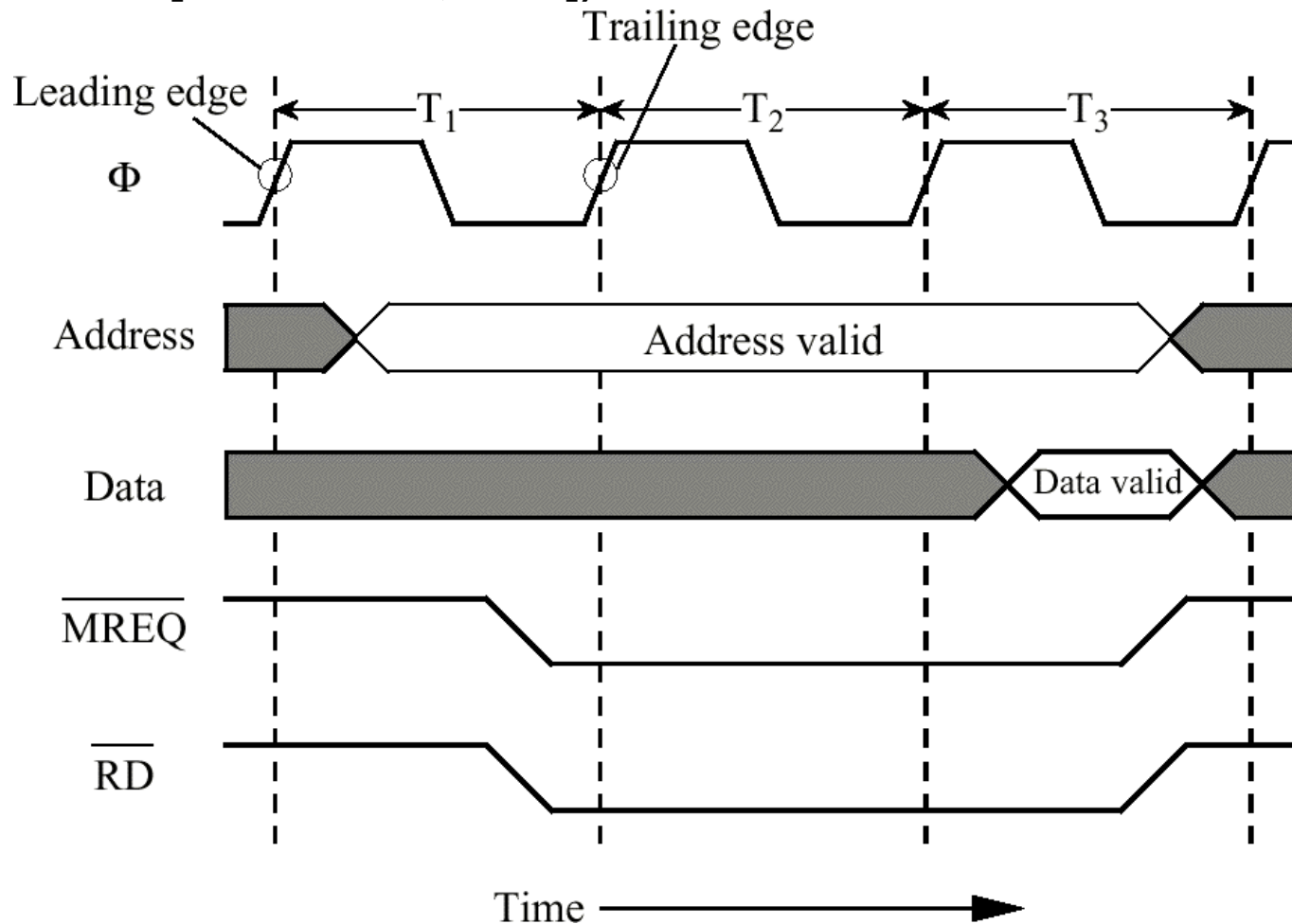
- **Síncronos**
 - Um dos dispositivos contém um oscilador
- **Assíncronos**

Um Clock de Barramento de 100 MHz



O Barramento Síncrono

- Diagrama de tempo para uma *leitura de memória síncrona* (adaptado de [Tanenbaum, 1999]).

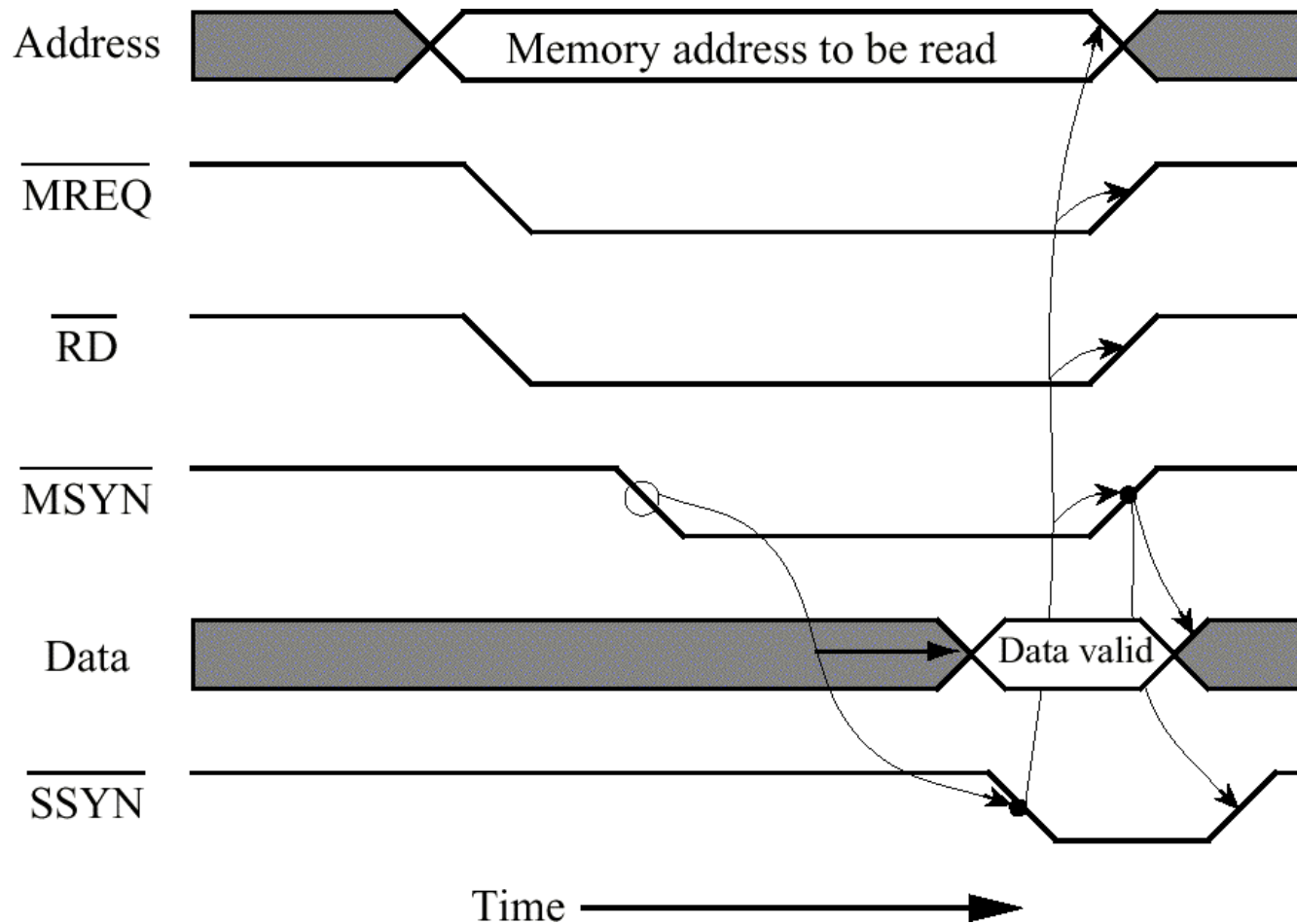


Barramento Assíncrono

- Mestre coloca tudo que precisa no barramento
 - Endereços, dados e controle
- Ativa $\overline{\text{MSYN}}$ (Master Synchronization)
- Escravo troca dados com o Mestre
- Ativa SSYN (Slave Synchronization)
 - Ao final da troca de dados
- Mestre desativa $\overline{\text{MSYN}}$
- Escravo desativa SSYN

O Barramento Assíncrono

- Diagrama de tempo para uma *leitura de memória assíncrona* (adaptado de [Tanenbaum, 1999]).

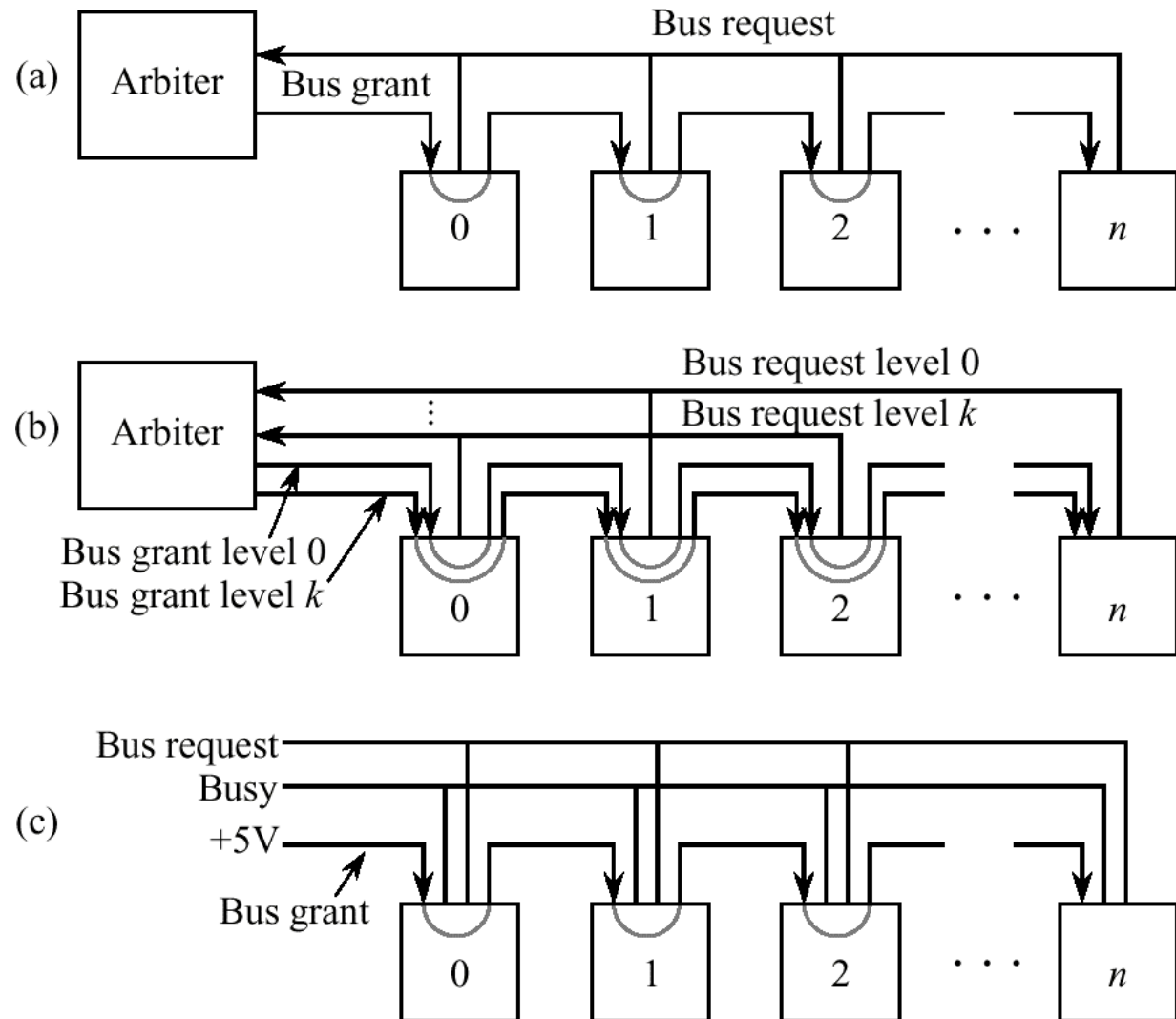


Arbitragem do Barramento

- **Mais de um dispositivo quer ser Mestre do barramento ao mesmo tempo**
- **Esquemas básicos de arbitragem**
 - **Centralizado**
 - **Descentralizado**

Arbitragem do Barramento

- (a) arbitragem centralizada simples de barramento; (b) arbitragem centralizada com níveis de prioridade; (c) arbitragem de barramento descentralizada (adaptado de [Tanenbaum, 1999]).



Arbitragem centralizada simples

- Dispositivo quer ser Mestre
 - Ativa a linha de *pedido do barramento*
- Árbitro determina se *permissão do barramento* pode ser dada
- Se sim, árbitro ativa a linha de permissão do barramento
- Linha de permissão do barramento – *daisy chained*
 - O primeiro dispositivo que quer ser mestre toma posse do barramento e não propaga a permissão

Arbitragem centralizada com níveis de prioridade

- **Várias linhas de pedido/permissão**
- **Linhas de pedido com números baixos têm maior prioridade**
- **Dentro de uma linha, prioridade definida pela proximidade elétrica do árbitro**
- **Levando-se o design ao extremo, cada dispositivo pode ter sua linha de pedido/permissão do barramento**
 - ***Arbitragem completamente centralizada***

Arbitragem de Barramento Descentralizada

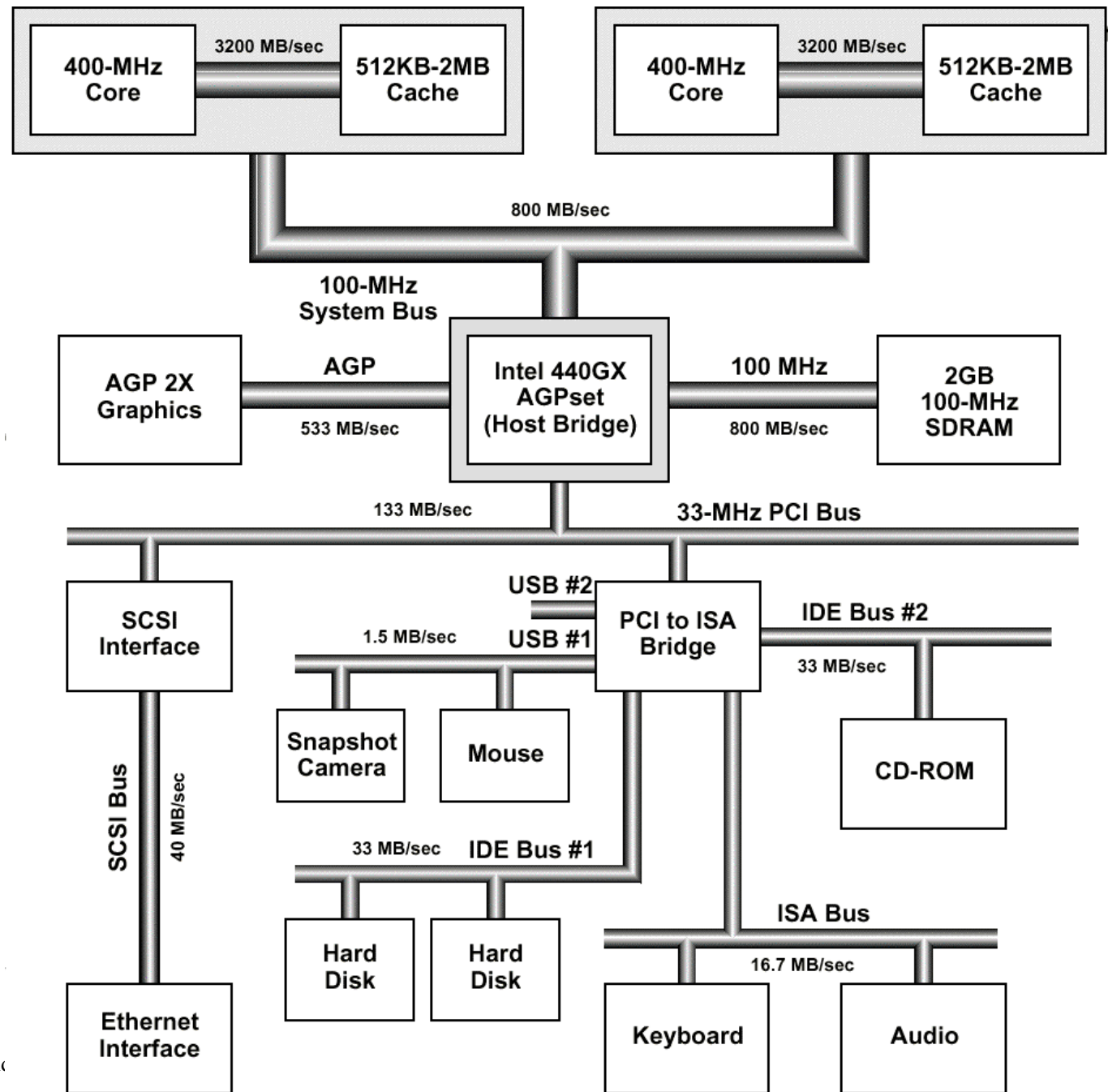
- **Dispositivo quer se tornar Mestre**
 - **Ativa linha de pedido do barramento**
 - **Verifica se está ocupado**
- **Se linha de ocupado não ativa**
 - **Envia zero ao próximo dispositivo**
 - **Ativa a linha ocupado**
 - **Desativa pedido**
- **Se linha de ocupado ativa OU**
- **Se dispositivo não quer usar o barramento**
 - **Propaga a permissão para o próximo dispositivo**

Arquiteturas de Barramentos Baseados em Pontes

- **Objetivo**
 - Permitir algumas transferências de dados simultâneas
- **Exemplo**
 - Leitura de bloco de memória para o cache
 - Recepção de dados na rede
- **Solução**
 - Utilização de pontes

Arquitetura de Barramentos Baseados em Pontes

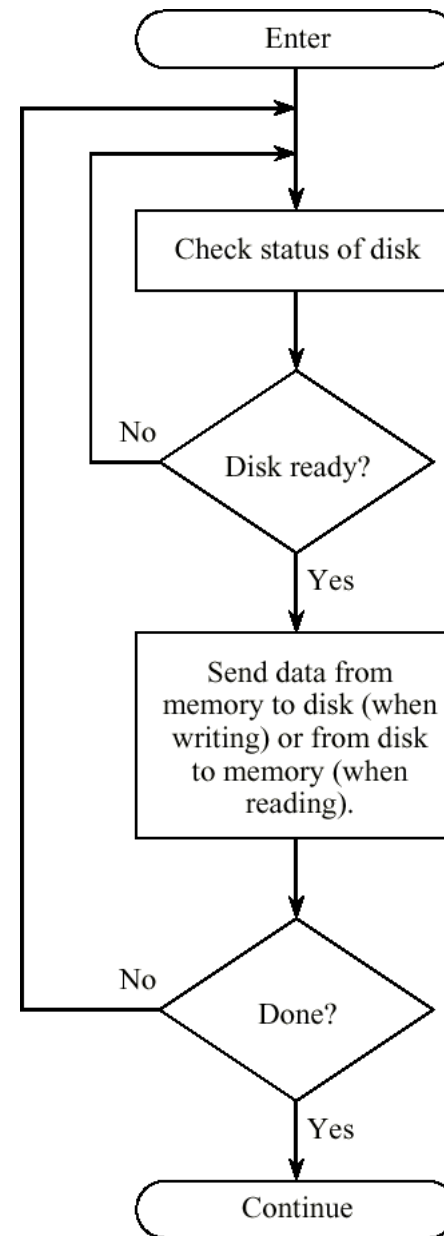
- Uso de ponte em sistema com dois processadores Pentium II Xeon (fonte: www.intel.com.)



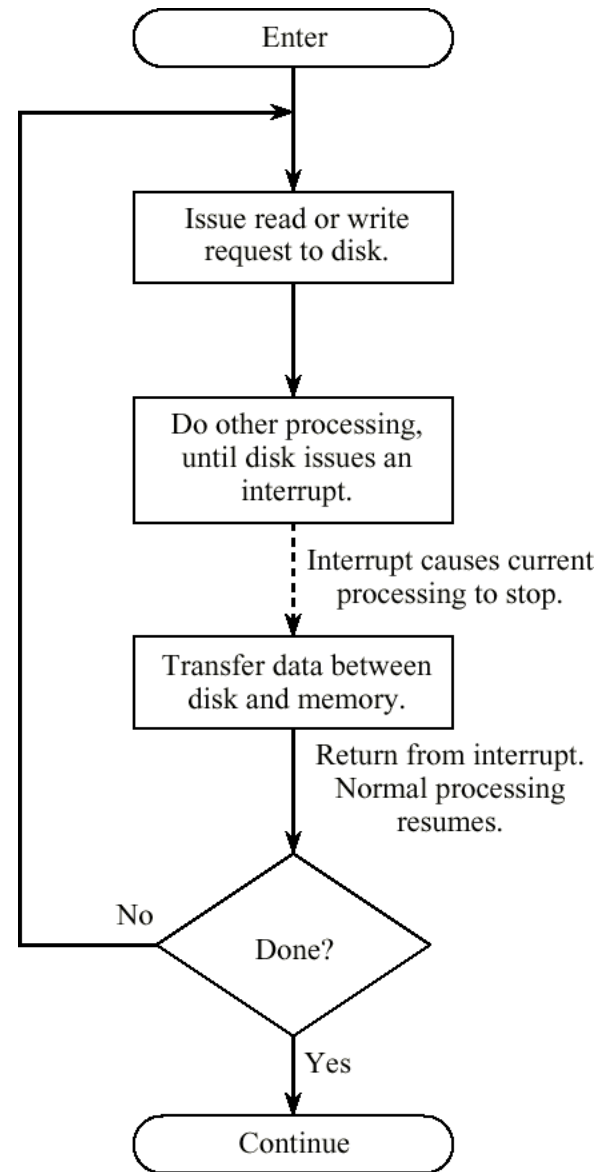
Metodologias de Comunicação

- **Entrada e saída programada (ou *polling*)**
- **Entrada e saída por interrupção**
- **Acesso direto à memória (DMA)**

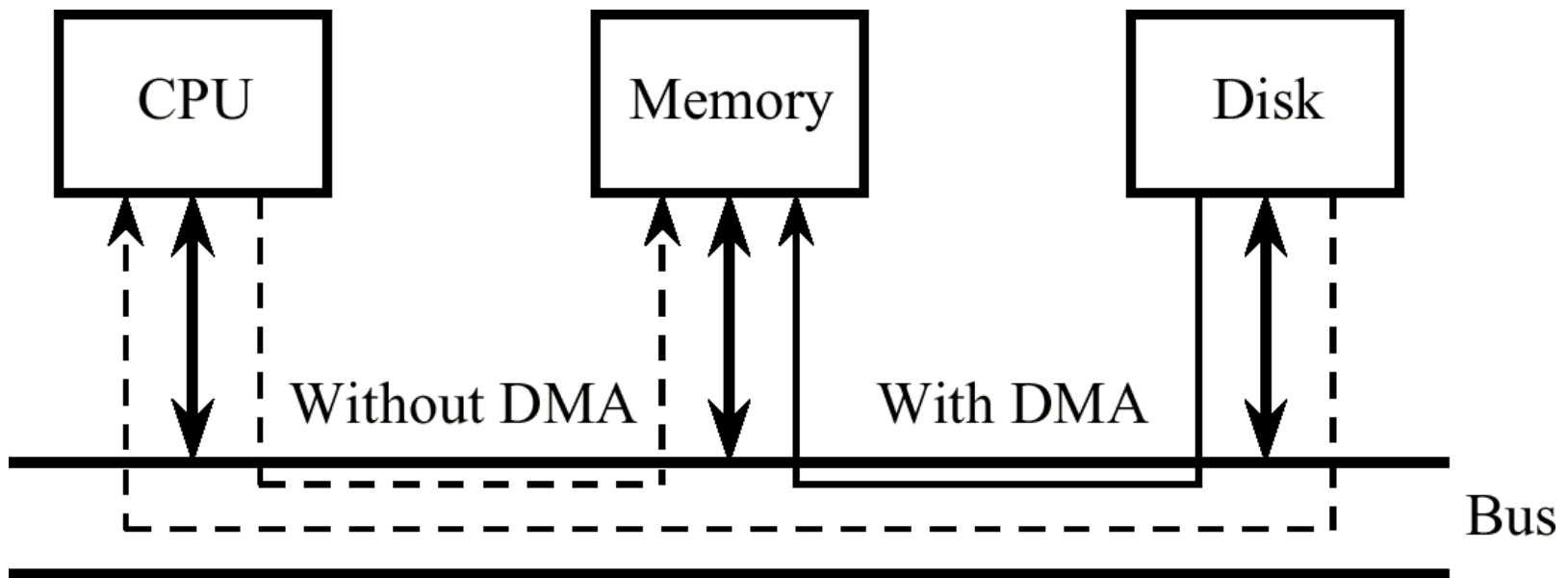
Fluxograma de E/S Programada (*polling*) para uma transferência de disco



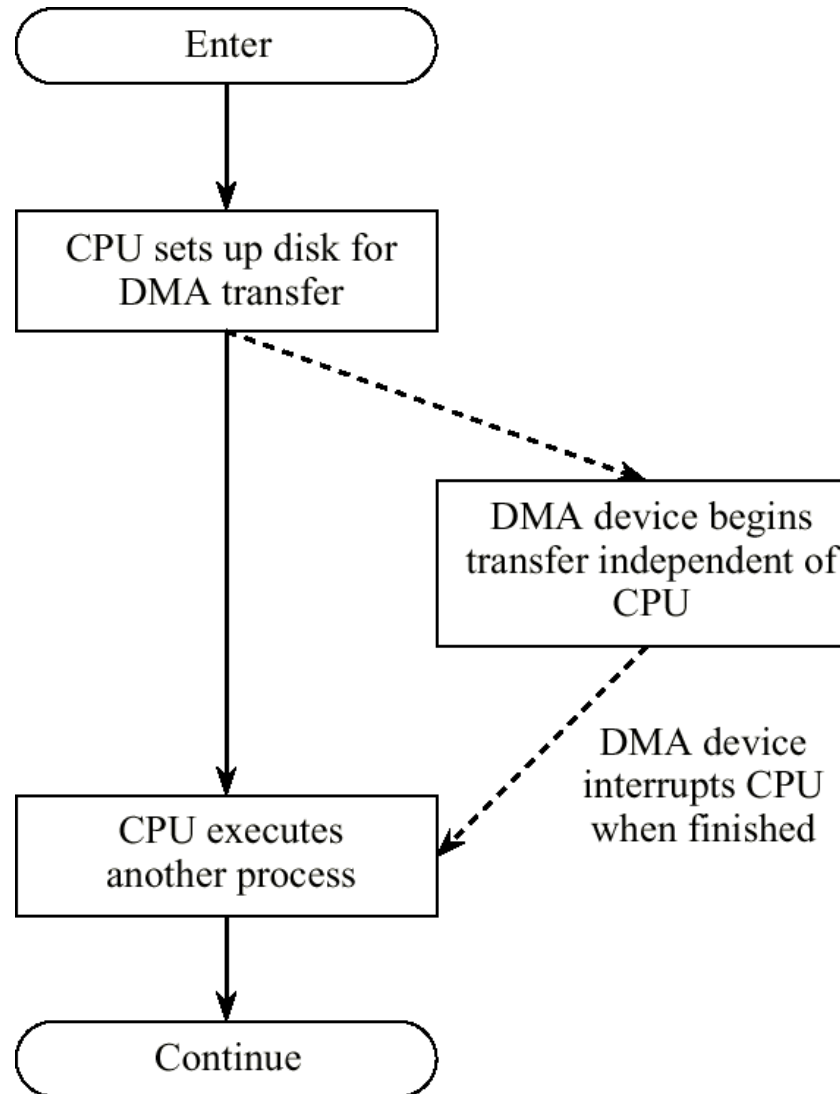
Fluxograma de E/S controlada por interrupção para uma transferência de disco



Transferência de DMA do Disco para a Memória sem usar CPU



Fluxograma de DMA para uma Transferência do Disco



Comunicação na Arquitetura Pentium

- Barra de endereços – 32 bits
 - 4 GB de memória principal
- Barra de dados – 64 bits
 - Pode transferir até uma palavra quádrupla (8 bytes) em um ciclo de barramento
- Espaços de endereçamento separados para memória e E/S
 - Linha do barramento M/IO#
 - Opcodes separados IN e OUT
 - Endereços limitados a 16 bits (64K posições de E/S)

Espaços de Endereçamento de Memória e de E/S da Arquitetura Intel Pentium

Address
FFFFFFFF

Memory
Space

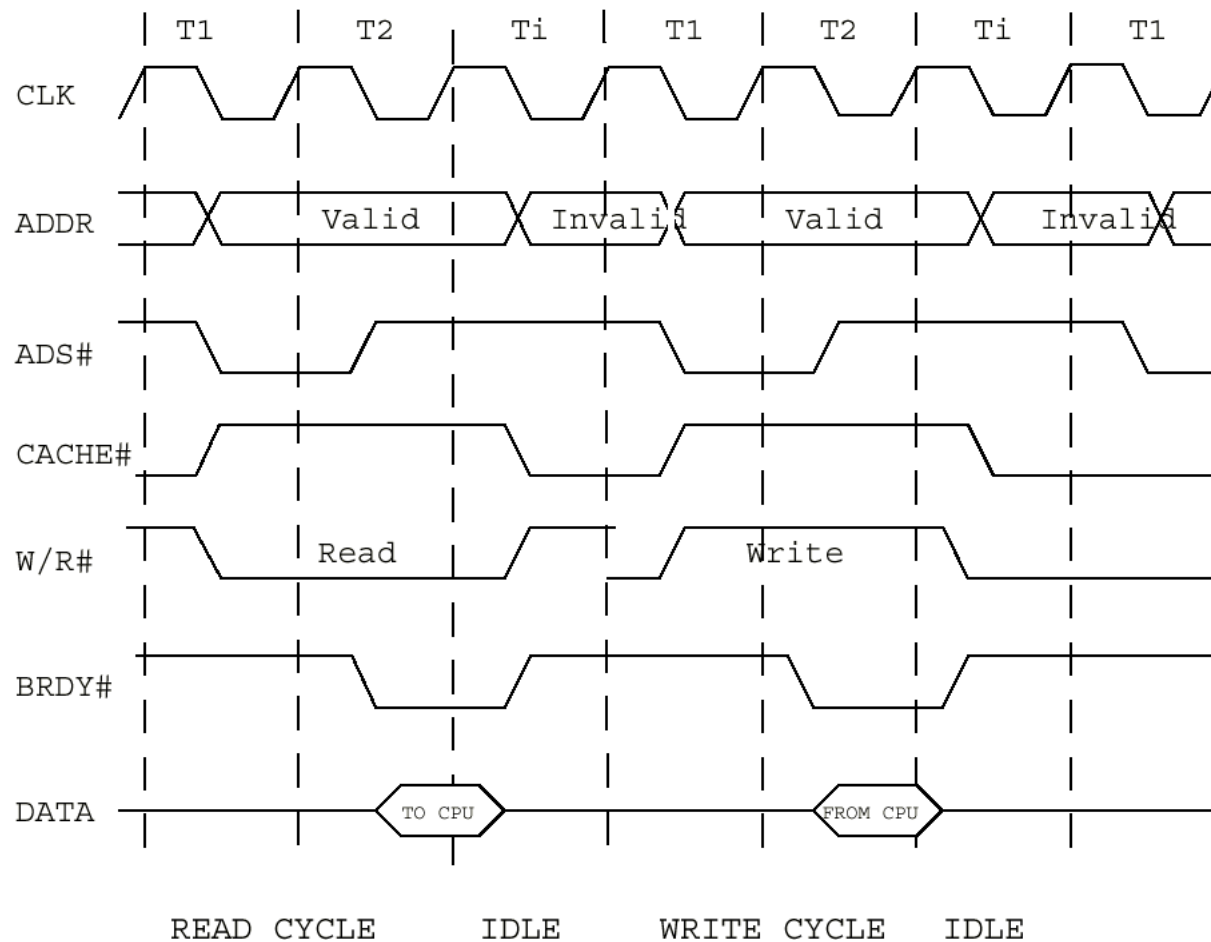
00000000

Address
FFFF

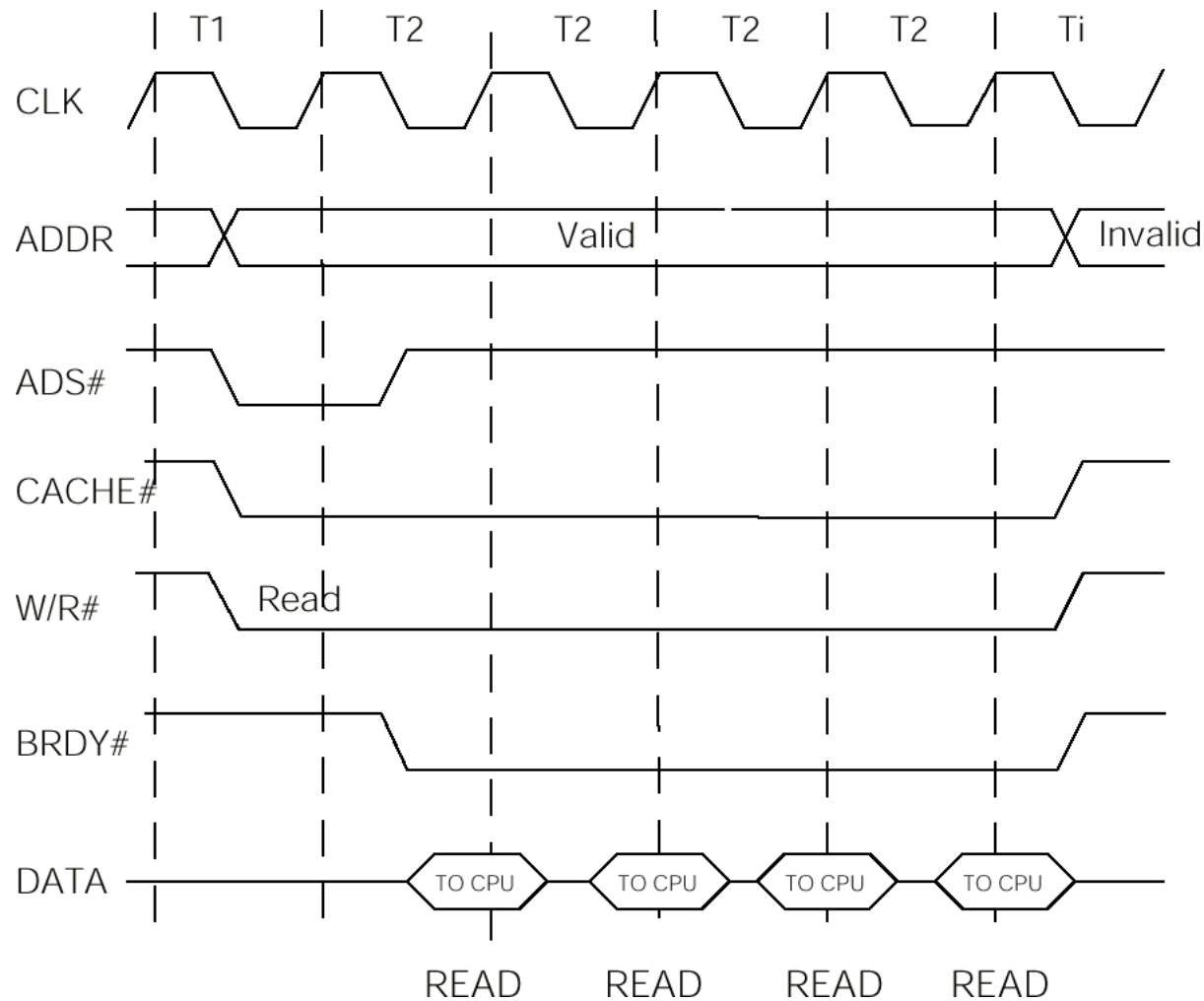
I/O
Space

0000

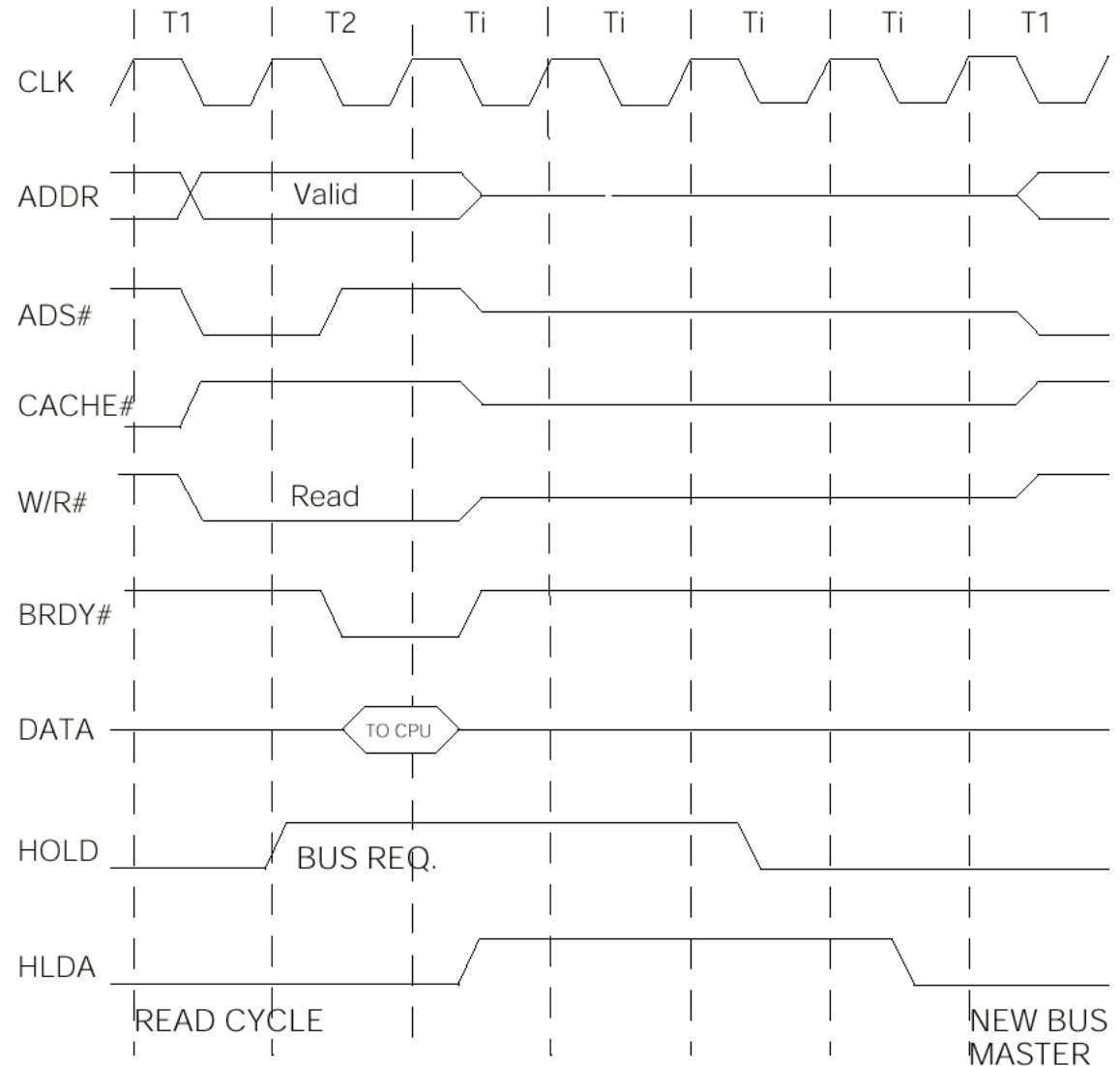
Ciclos de Barramento Intel Pentium de Leitura e Escrita Comuns



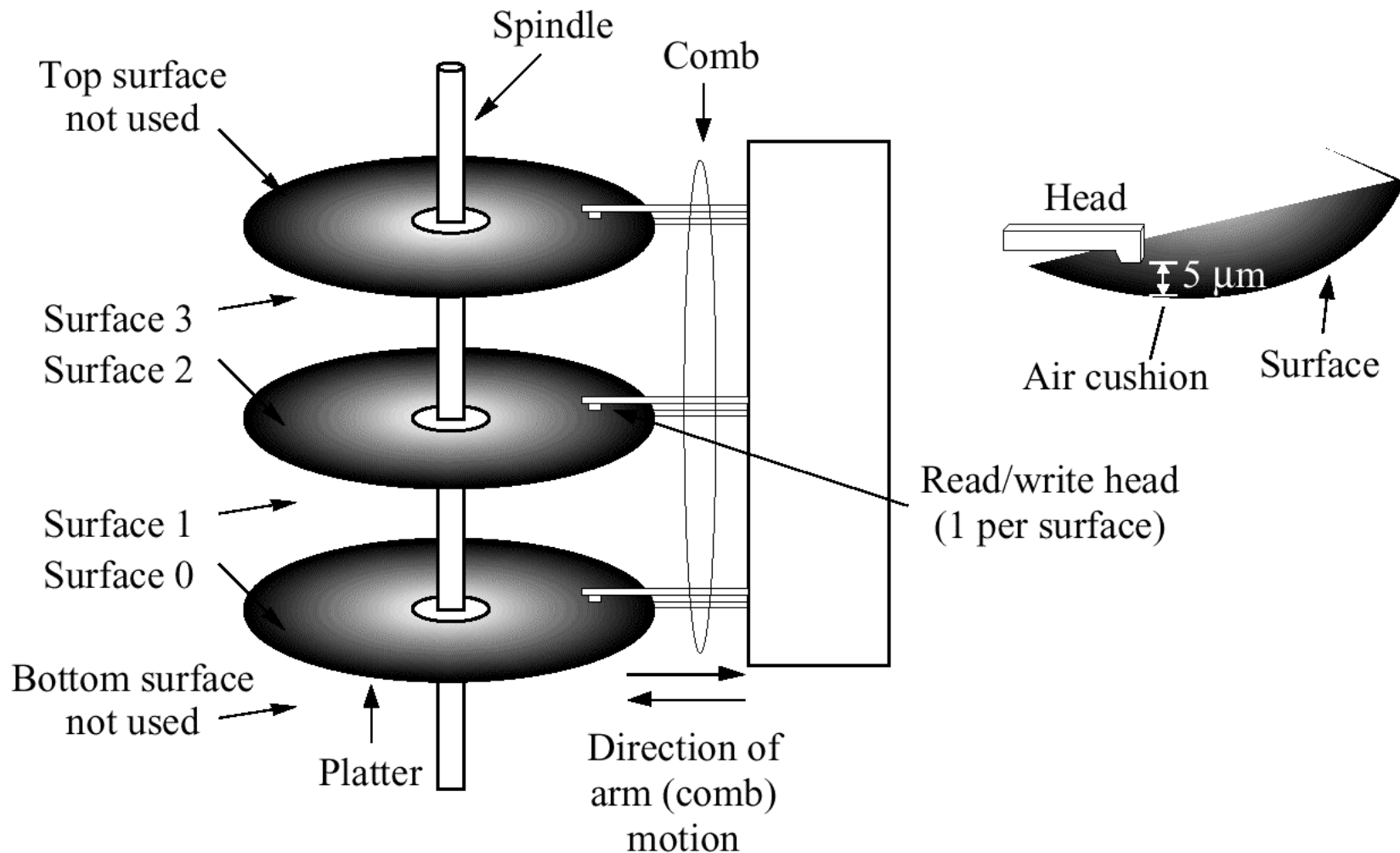
Ciclo de Barramento Intel Pentium de Leitura em Rajada



Ciclo de Barramento Intel Pentium Suspensão – Confirmação da Suspensão (*Hold-Hold Acknowledge*)

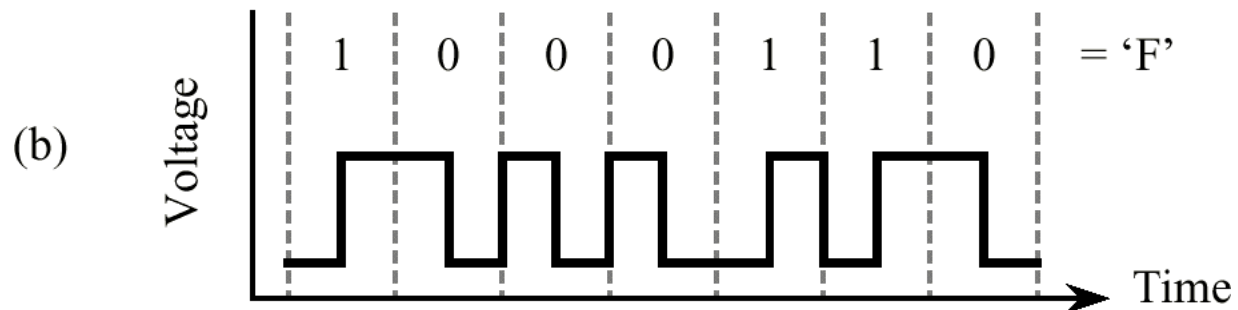
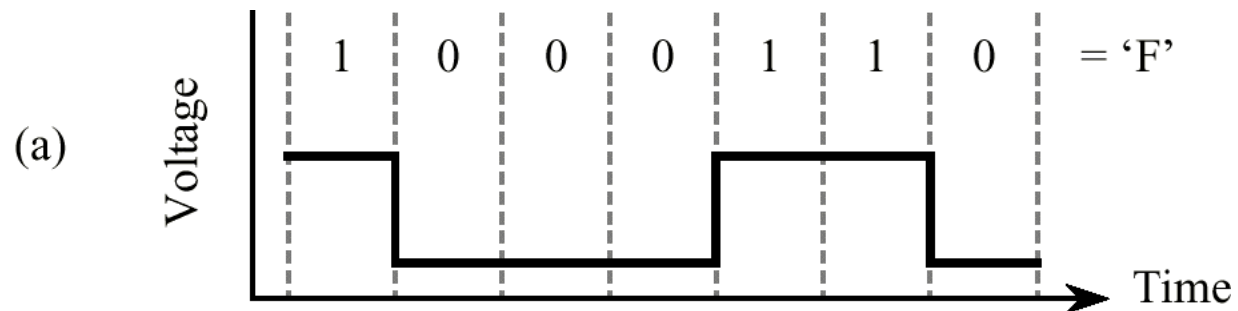


Um Disco Magnético com Três Pratos



Codificação de Manchester

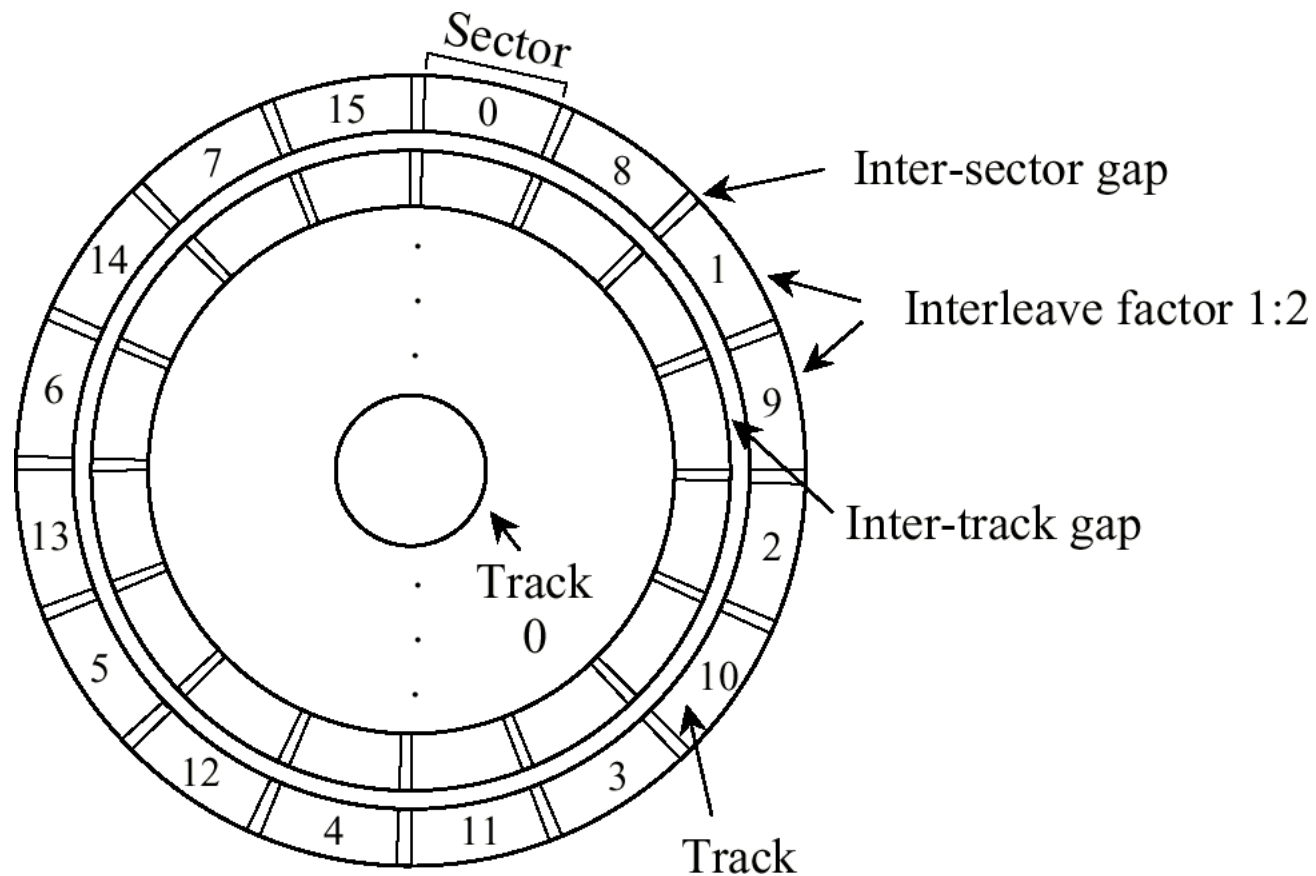
- (a) Straight amplitude (NRZ) encoding of ASCII 'F'; (b) Manchester encoding of ASCII 'F'.



Sistemas de Arquivos em Discos

- **Arquivo**
 - **Coleção de setores do disco**
 - **Setores consecutivos minimizam tempo de busca e latência rotacional**
 - **Arquivos podem se tornar fragmentados...**
 - **Otimizadores defragmentam o disco**
- **Intercalação**
 - **“Dar tempo” à CPU, caso o próximo setor seja requisitado**

Organização de um Prato de Disco com Fator de Intercalação 1:2



Organização do Disco Magnético

- Cada superfície contém várias trilhas
- Trilhas são compostas por setores
- Setores armazenam número fixo de bytes
 - Ex.: 512 bytes
- Fisicamente
 - Setores separados por inter-sector gaps
 - Trilhas separadas por inter-track gaps

Zone bit recording

- **Técnica para aumentar a capacidade do disco**
- **Trilhas separadas em zonas**
- **Zonas próximas ao centro do prato têm menos setores que as próximas à beirada do prato**

Capacidade do Disco

- Capacidade de armazenamento (C)
 - N – número de bytes por setor
 - S – número de setores por trilha
 - T – número de trilhas por superfície
 - P – número de superfícies
- $C = N \times S \times T \times P$
- Ex. N = 512 bytes por setor, S = 1000 setores por trilha, T = 5000 trilhas por superfície e P = 8 pratos
 - C = 38GB

Velocidade de Transferência Máxima

- Tempo para mover a cabeça para a trilha desejada
 - (tempo de busca ou *seek time*)
- Tempo para o setor desejado aparecer sob a cabeça
 - (latência rotacional)
- Tempo para transferir o setor do prato do disco
 - (tempo de transferência)
- Aproximações
 - Tempo de busca = $\frac{1}{2}$ da viagem completa
 - Latência rotacional = $\frac{1}{2}$ da revolução completa
 - Tempo de transferência
 - (tempo da revolução completa) / (número de setores na trilha)

Bloco-mestre de Controle

- **MCB (Master Control Block)**
 - **Seção reservada**
 - **Controla o conteúdo do restante do disco**
- **Normalmente armazenado em um lugar fixo**
 - **Ex. trilha mais interna do disco**

Master Control Block

Preamble {
 No. surfaces on disk = 4
 No. tracks/surface = 814
 No. sectors/track = 32
 No. bytes/sector = 512
 Interleave factor = 1:3

Starting sector, or sector list

	Filename	Starting sector, or sector list			Creation Date	Last Modified	Owner	Protections
		Surface	Track	Sector				
Files {	xyz.p	1	10	5	11/14/93 10:30:57	11/14/93 19:30:57	16	RWX by Owner
		1	12	7				
		2	23	4				
	ab.c	1	10	8	8/18/93 16:03:12	1/21/94 14:45:03	20	RX - All W-Owner
		3	95	2				
		2	12	0				
			⋮					
Free blocks {		1	1	0				
		1	1	1				
		1	2	5				
				⋮				
Bad blocks {		1	1	3				
		2	5	7				
				⋮				

R = Read
 W = Write
 X = Execute

Bloco-mestre de Controle (cont.)

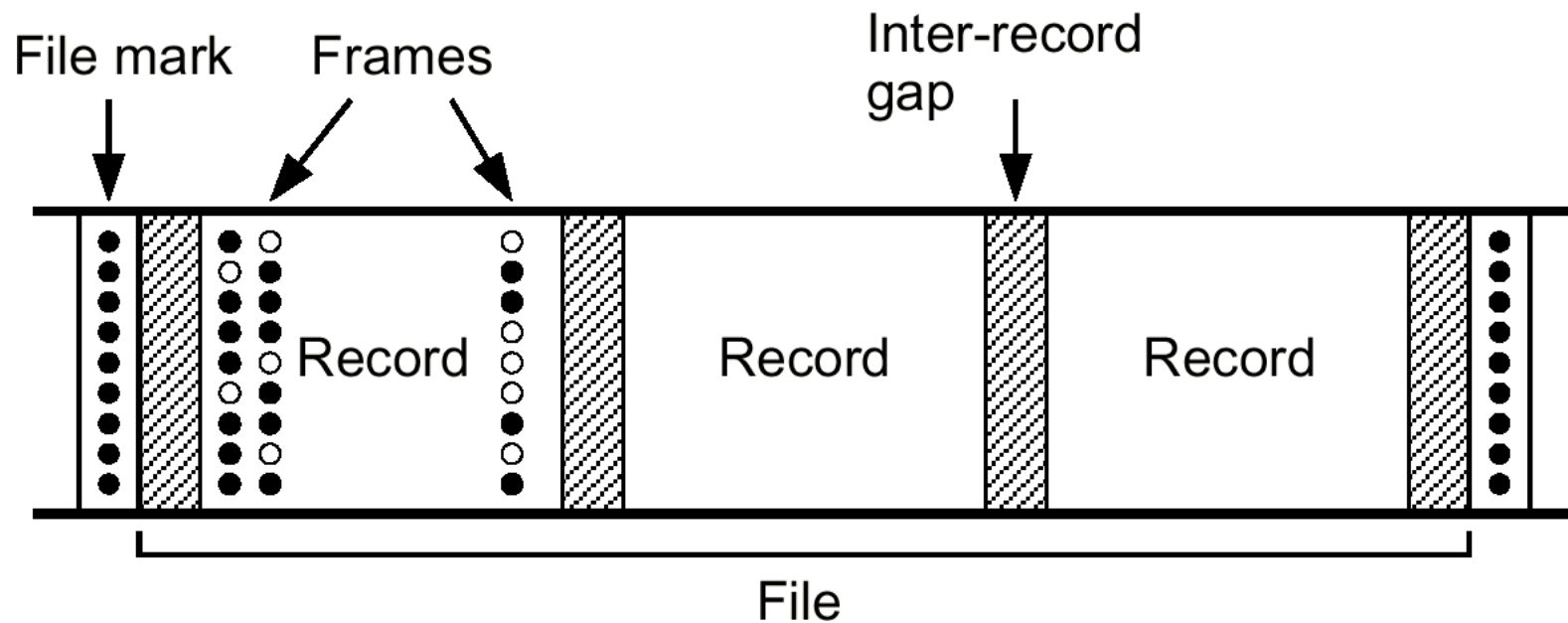
- **Arquivos crescem**
 - S.O. busca blocos livres no MCB
 - Pode haver grande movimentação da cabeça
 - (blocos livres distantes do MCB)
 - Melhora: cópia do MCB em memória
- **Sincronização do disco**
 - Atualização do MCB ao desligar a máquina

Fitas Magnéticas

- **Rolo de fita plástica com cobertura magnética**
- **Cabeça magnetiza a fita (escrevendo) ou sente o campo magnético (lendo)**
- **Acesso lento**
 - **Todas as seções passam pela cabeça antes da seção desejada**
- **Informação armazenada de forma bi-dimensional**
 - **Bytes – quadros ao longo da largura da fita**
 - **Registros ao longo do comprimento da fita**

Segmento de Fita Magnética

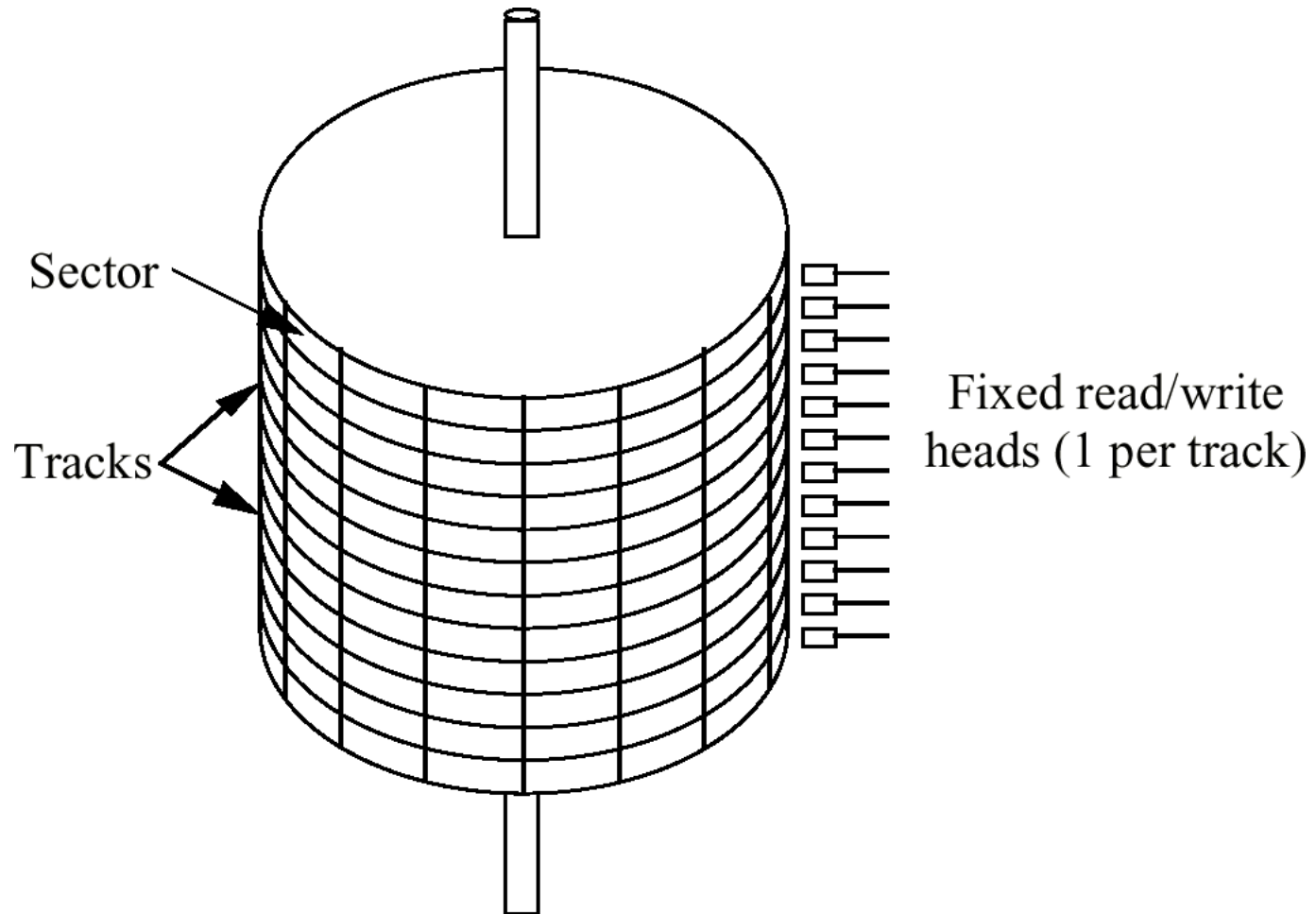
- Segmento de fita magnética



Fita Magnética (cont.)

- **Adequada para grandes quantidades de dados**
 - **Cópias de segurança**
- **Inadequada para acesso aleatório**
 - **Acesso seqüencial consome muito tempo**
 - **Em geral, não é possível escrever no meio da fita**

Tambor Magnético



Velocidade de Rotação

Taxa de dados constante, então se

- **Velocidade Angular Constante**
 - bits mais espaçados próximo à borda
 - blocos de dados acessados diretamente na trilha/setor,
 - mas há desperdício de espaço
- **Velocidade Linear Constante**
 - densidade de bits constante
 - mais informação armazenada,
 - atraso rotacional maior para trilhas próximas da borda

Discos Ópticos

Disco magnético

- velocidade angular constante

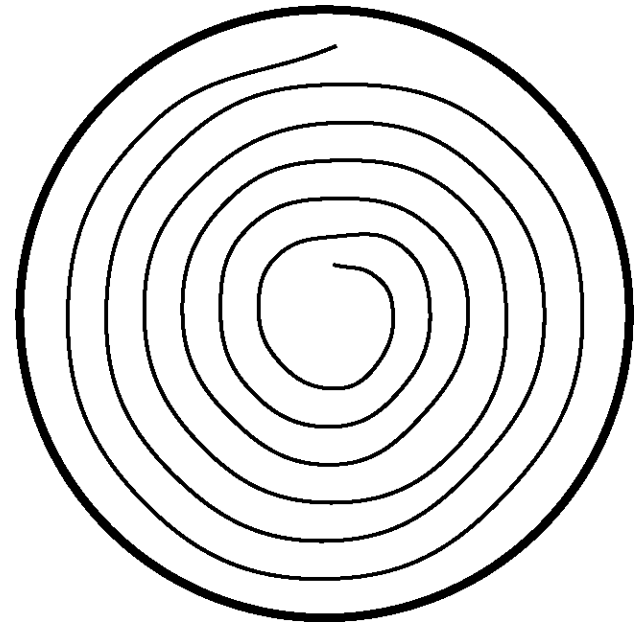
CD

- velocidade linear constante

(disco gira mais lentamente qdo
cabeça próxima da borda)

Velocidade

- dada em número de vezes a
velocidade do CD de áudio (Ex. 24x)



CD de Áudio

- 2 canais (estéreo)
- 44.000 amostras/s
- 16 bits por amostra
- Aproximadamente 74 minutos
- Capacidade ~ 745,2 MB

CD-ROM

- Diferentes densidades
- Exemplo
 - Espaçamento entre trilhas = $1,6\mu\text{m}$
 - Largura útil = $32,55\text{ mm}$
 - 20.344 trilhas
- Na verdade, uma trilha espiral de $\sim 5,27\text{km}$
- Velocidade linear = $1,2\text{m/s}$
 - 4.391s ou 73,2 min
- Dados lidos a $176,4\text{ Kbytes/s}$
- Capacidade – $774,5\text{ Mbytes}$

CD-R

- **WORM (*Write Once Read Many*)**
 - “preparado” por um laser de alta intensidade
 - usa velocidade angular constante
 - acesso mais rápido, menos capacidade
- **Método típico**
 - Na fábrica, laser de alta intensidade produz bolhas na superfície do disco
 - No drive, laser de mais baixa intensidade pode romper as bolhas

CD-RW

- **Técnica de mudança de fase**
 - **Material possui dois índices de reflexão distintos**
 - **Estado amorfo**
 - Reflete pouca luz
 - **Estado cristalino**
 - Superfície suave, com boa reflexão de luz
 - **Laser pode mudar o material de uma fase para outra**
- **Desvantagem**
 - **O material perde essa propriedade após séries de apagamentos (~500.000 a 1 milhão)**

DVD

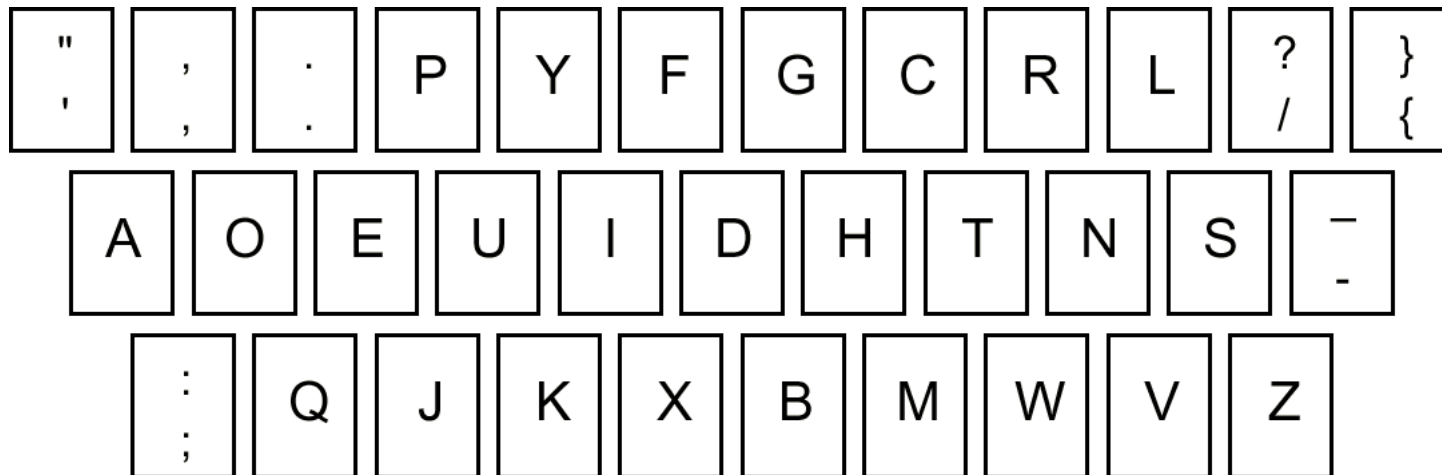
- **DVD (*Digital Versatile Disc*)**
- **4,7 Gbytes por lado, se um lado só é usado**
- **8,5 Gbytes por lado, usando-se duas camadas por lado**
 - **17Gbytes no total**

O Layout de Teclado ECMA-23

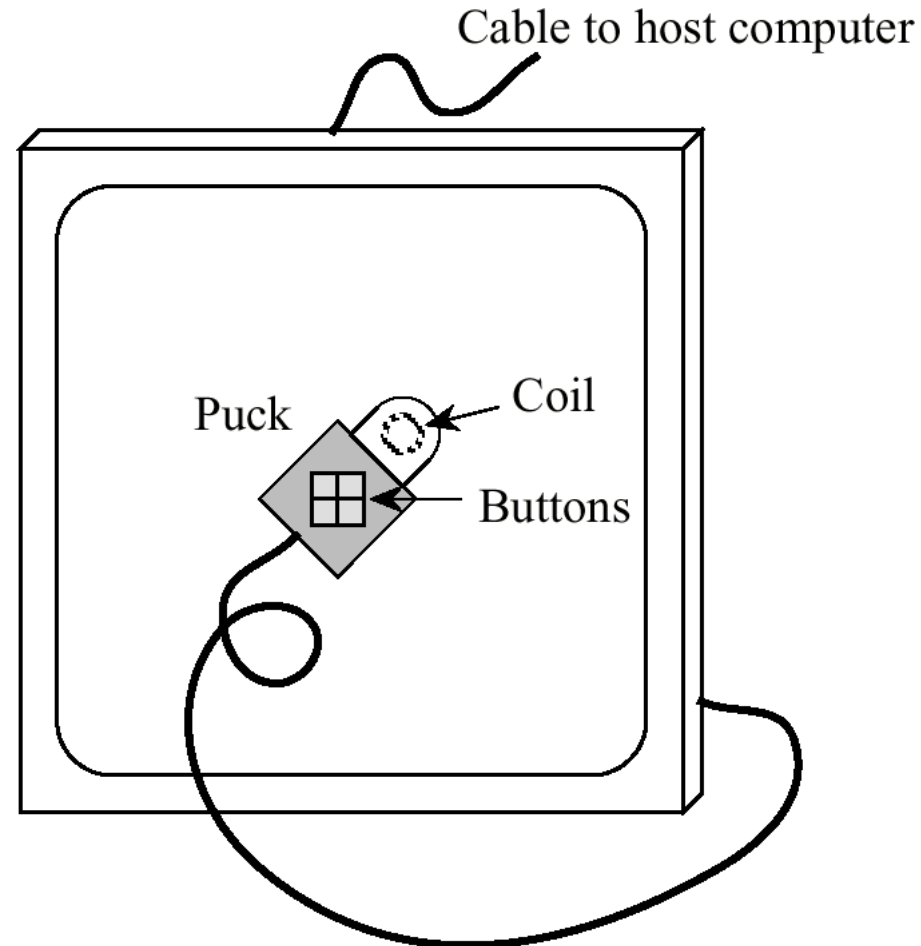
- Disposição do teclado-padrão ECMA-23 (2nd ed.). Teclas de shift são freqüentemente colocadas na fila B.

	99	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	
F																					
E		⌵	! 1	" 2	# 3	⌘ 4	% 5	& 6	' 7	(8) 9	0	=	⌵ ^							
D			Q	W	E	R	T	Y	U	I	O	P	@	}			7	8	9		
C			A	S	D	F	G	H	J	K	L	+ ;	* :	}		-	4	5	6		
B		⌵	Z	X	C	V	B	N	M	< ,	> .	? /					⌵	1	2	3	
A			[Shift Key]														0	00	.	SP	
Z																					

O Layout do Teclado Dvorak

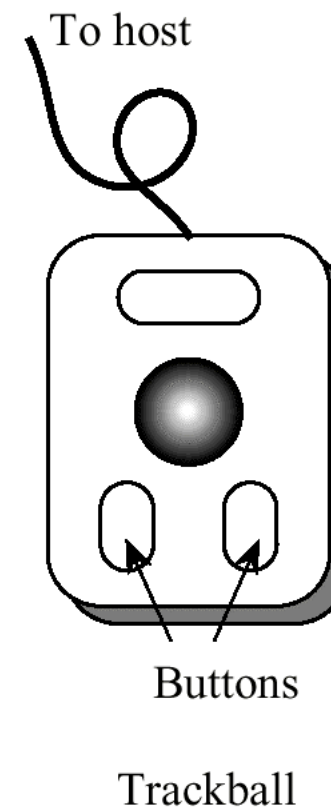
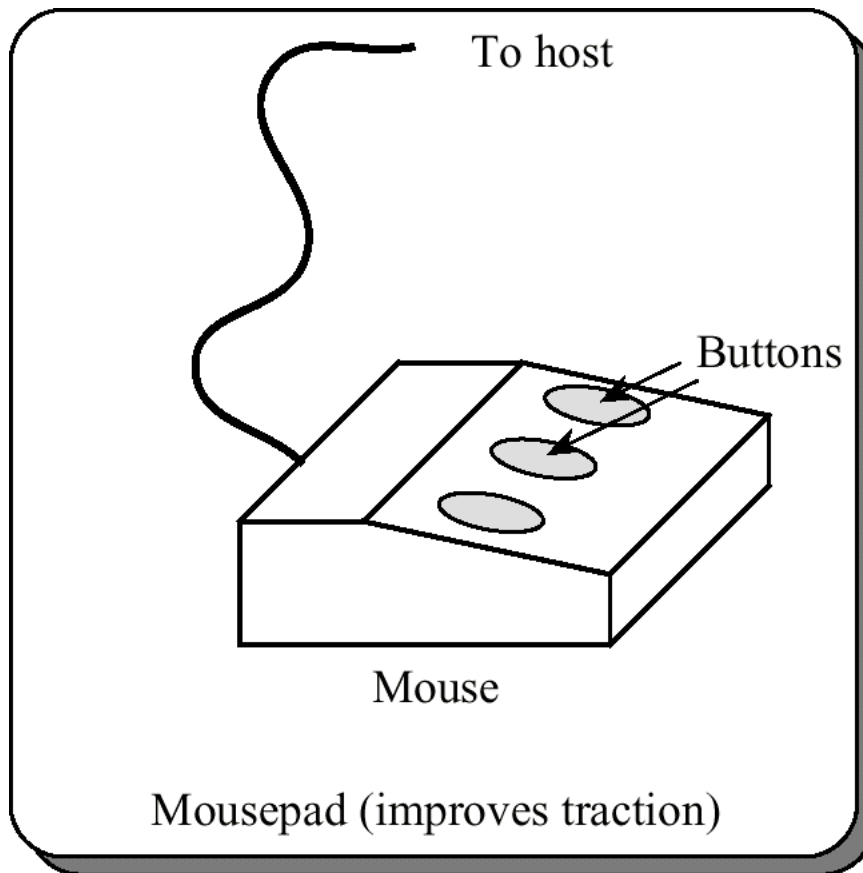


Mesa Digitalizadora (Bit Pad) com um Puck



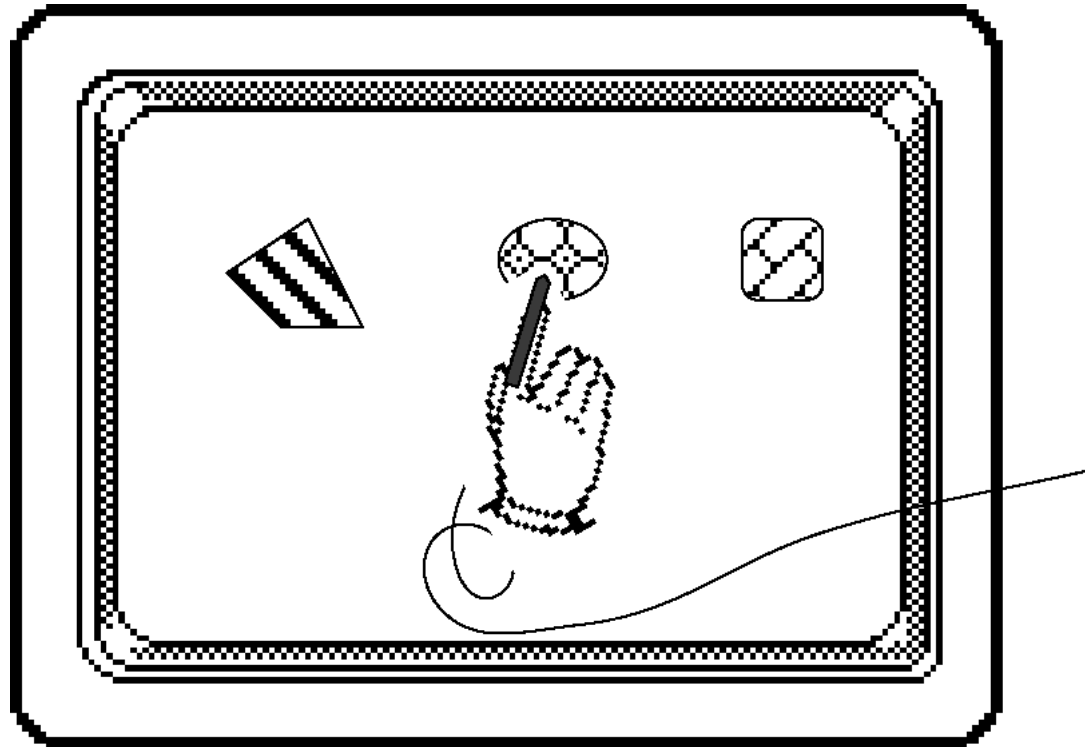
Mouse e Trackball

- Um mouse de três botões e um trackball de três botões.



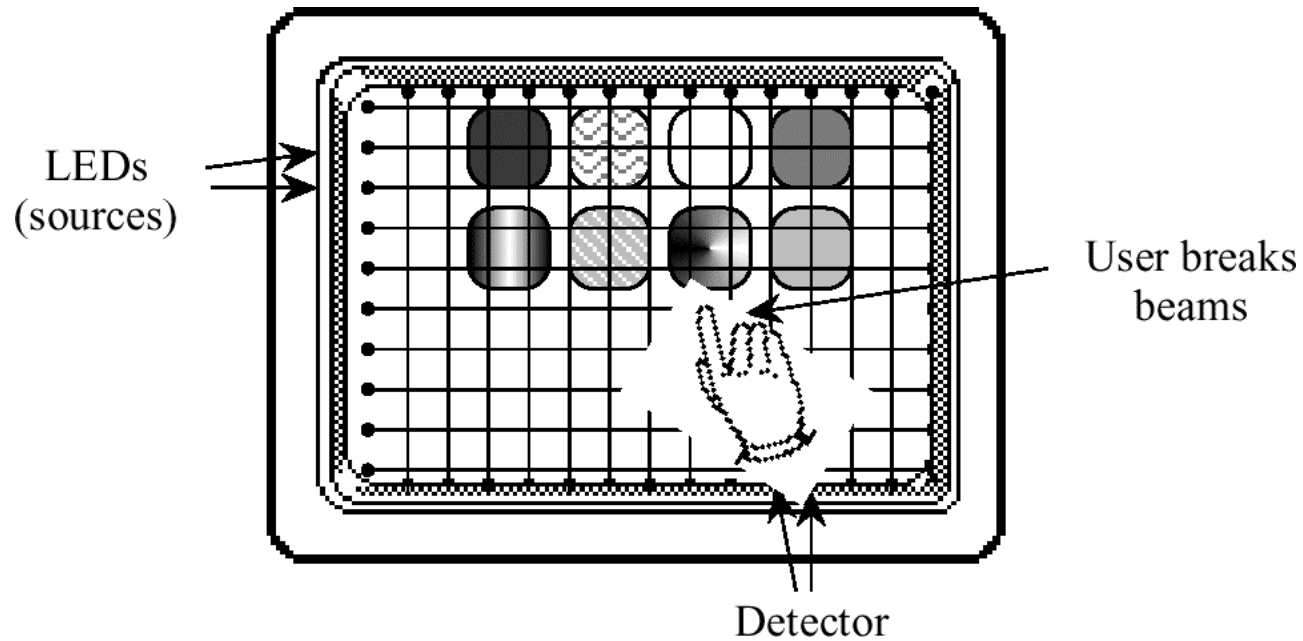
Lightpen

- Usuário selecionando um objeto com uma lightpen.



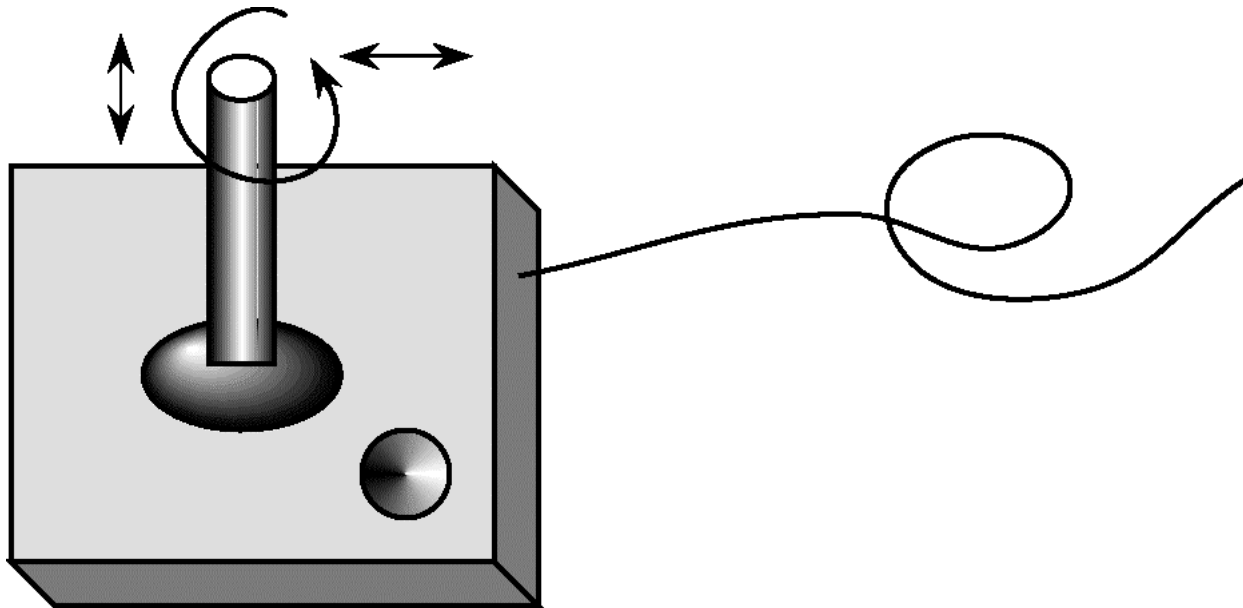
Touchscreen

- Usuário selecionando um objeto em uma tela de toque.



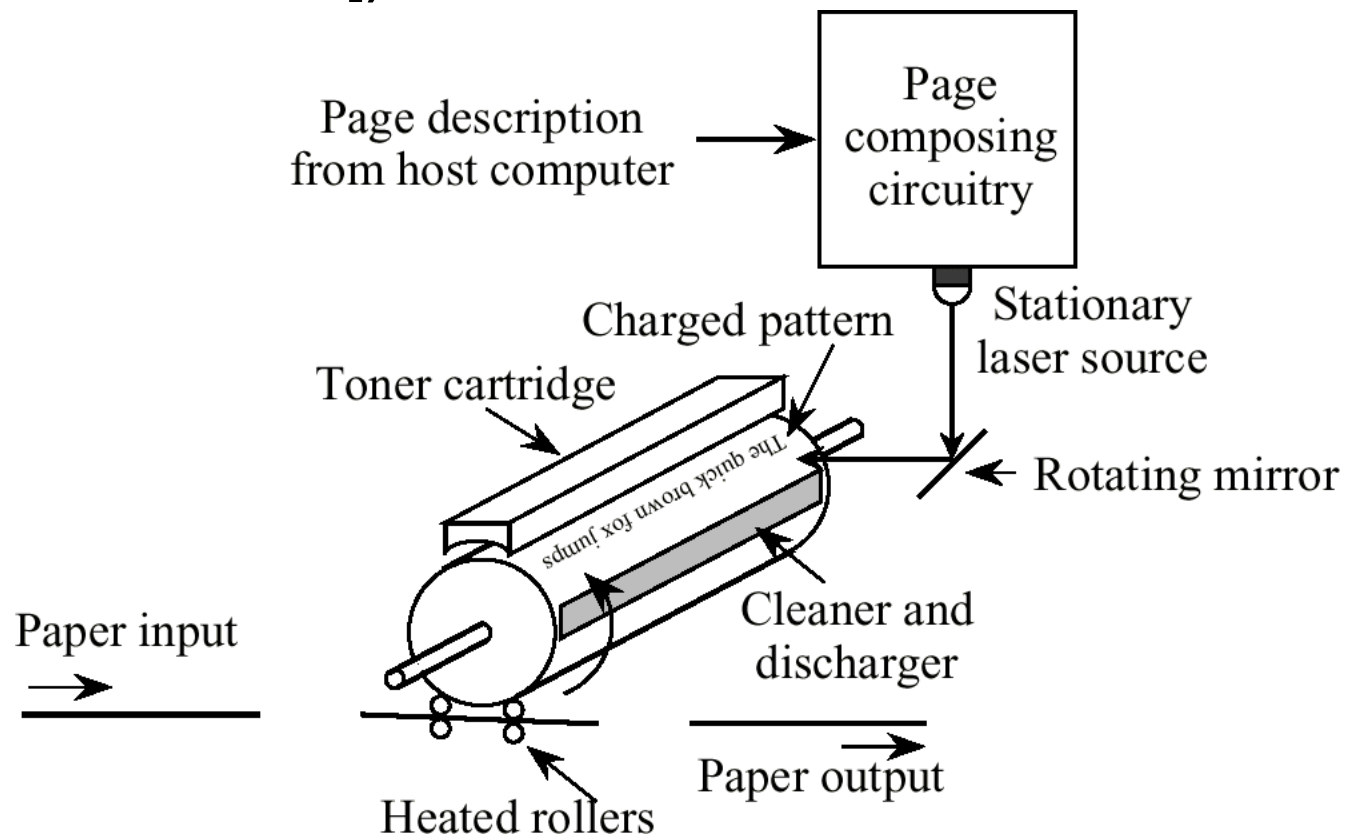
Joystick

- Um joystick com botão de seleção e eixo com rotação:



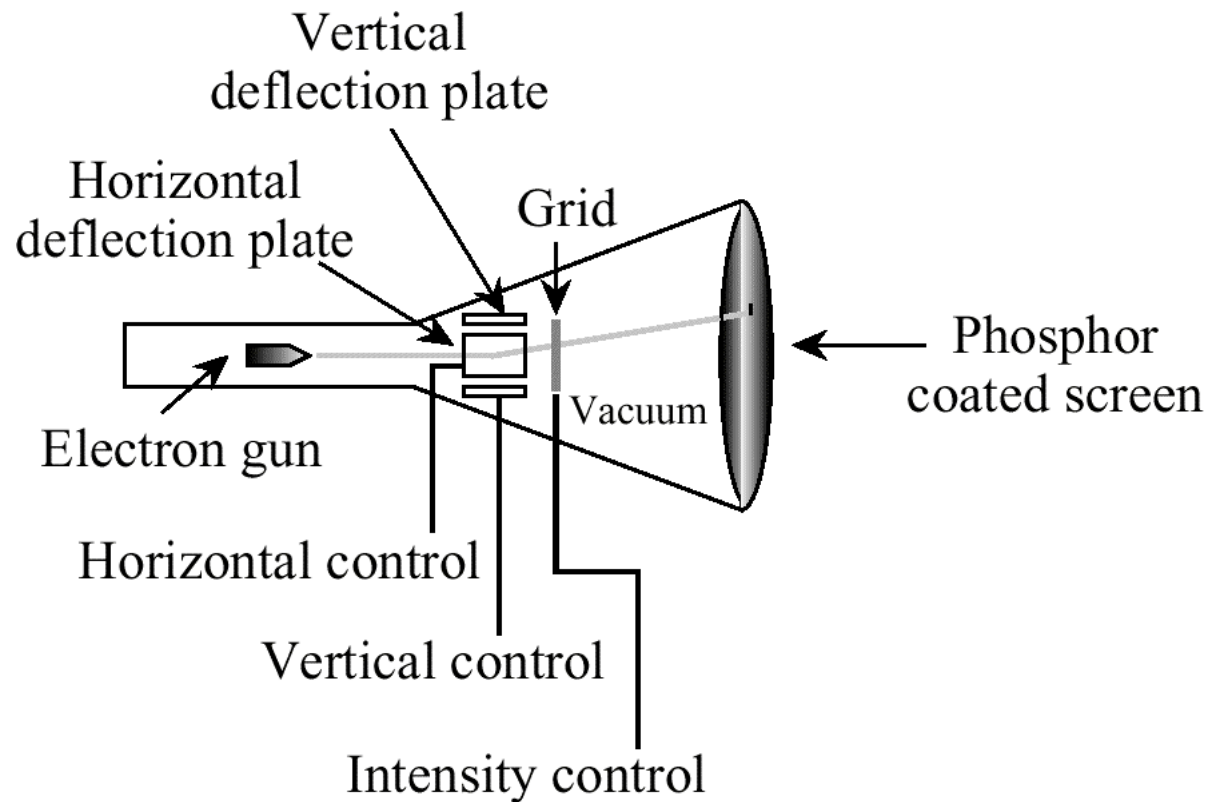
Impressora a Laser

- Esquemático de uma impressora a laser (adaptado de [Tanenbaum, 1999]).



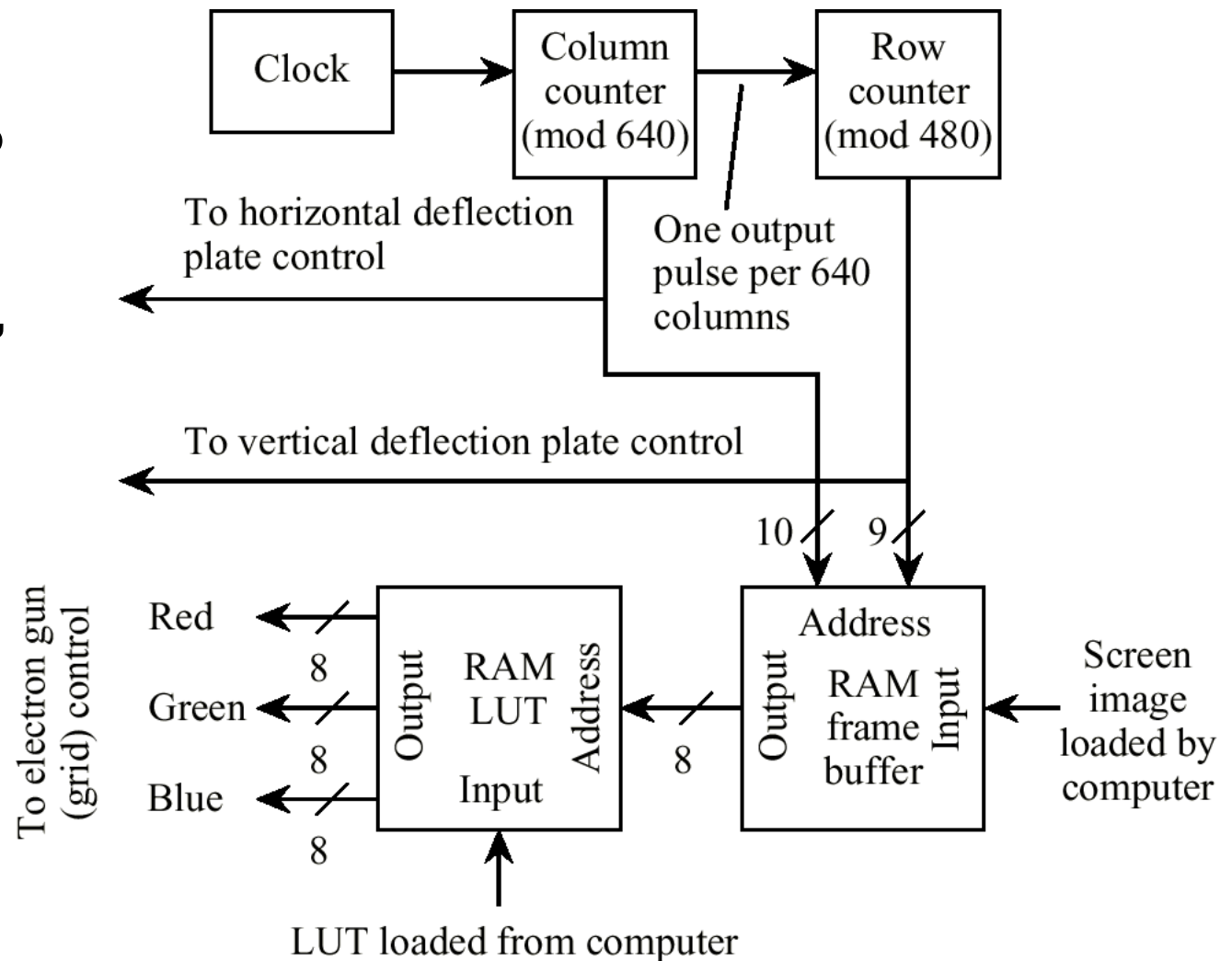
Tubo de Raios Catódicos

- Um CRT com um único canhão de elétrons:



Controlador de Vídeo

- Controlador de vídeo para monitor colorido 640x480 (adaptado de [Hamacher et al., 1990]).



Controlador de Vídeo

- **Frame buffer**
 - Armazena os padrões de bits correspondentes à imagem a ser exibida (carregada pelo computador)
 - Cada pixel
 - 1 a 32 bits
- **Tabela de Previsão de Cores**
 - Permite utilizar mais cores que o tamanho do pixel na memória de vídeo permitiria

Envio de Dados

- Taxa de dados entre computador e monitor
- Exemplo:
 - 24 bits por pixel
 - 1024x768 pixels
 - 60 Hz
 - ~140MBytes/s

VHDL Specification

Interface specification for the majority component

– **Interface**

entity **MAJORITY** is

port

(**A_IN, B_IN, C_IN**: in **BIT**

F_OUT: out **BIT**);

end **MAJORITY**;

Behavioral model for the majority component

– **Body**

architecture **LOGIC_SPEC** of **MAJORITY** is

begin

– compute the output using a Boolean expression

```
F_OUT  <=      (not A_IN and B_IN and C_IN) or
                (A_IN and not B_IN and C_IN) or
                (A_IN and B_IN and not C_IN) or
                (A_IN and B_IN and C_IN) after 4 ns;
```

end **LOGIC_SPEC**;

VHDL Specification (cont')

— Package declaration, in library WORK

```
package LOGIC_GATES is
```

```
  component AND3
```

```
    port (A, B, C : in BIT; X : out BIT);
```

```
  end component;
```

```
  component OR4
```

```
    port (A, B, C, D : in BIT; X : out BIT);
```

```
  end component;
```

```
  component NOT1
```

```
    port (A : in BIT; X : out BIT);
```

```
  end component;
```

— Interface

```
entity MAJORITY is
```

```
  port
```

```
    (A_IN, B_IN, C_IN: in BIT
```

```
    F_OUT: out BIT);
```

```
end MAJORITY;
```

VHDL Specification (cont')

```

— Body
— Uses components declared in package LOGIC_GATES
—in the WORK library
— import all the components in WORK.LOGIC_GATES
use WORK.LOGIC_GATES .all
architecture LOGIC_SPEC of MAJORITY is
— declare signals used internally in MAJORITY
signal A_BAR , B_BAR, C_BAR, I1, I2, I3, I4: BIT;
begin
— connect the logic gates
NOT_1 : NOT1 portmap (A_IN, A_BAR);
NOT_2 : NOT1 portmap (B_IN, B_BAR);
NOT_3 : NOT1 portmap (C_IN, C_BAR);
AND_1 : AND3 portmap (A_BAR, B_IN, C_IN, I1);
AND_2 : AND3 portmap (A_IN, B_BAR, C_IN, I2);
AND_3 : AND3 portmap (A_IN, B_IN, C_BAR, I3);
AND_4 : AND3 portmap (A_IN, B_IN, C_IN, I4);
OR_1 : OR3 port map (I1, I2, I3, I4, F_OUT);
end LOGIC_SPEC;

```