

O Sistema de Computação

Profa. Débora Christina Muchaluat Saade

debora@midiacom.uff.br

<http://www.ic.uff.br/~debora/fac>

O Sistema de Computação

Fundamentos de Arquiteturas de Computadores

- ✓ **Capítulo 2 – Livro do Mário Monteiro**
- ✓ **Componentes**
- ✓ **Representação das informações**
 - *Bit, Caractere, Byte e Palavra*
 - *Conceito de Arquivos e Registros*
- ✓ **Medidas de desempenho**

Sistema de Computação

Fundamentos de Arquiteturas de Computadores

- ✓ **Conjunto de componentes integrados com o objetivo de manipular dados e gerar informações úteis.**
 - *Processador*
 - *Memória*
 - *Dispositivos de entrada e saída (I/O devices)*

Componentes Básicos

Fundamentos de Arquiteturas de Computadores

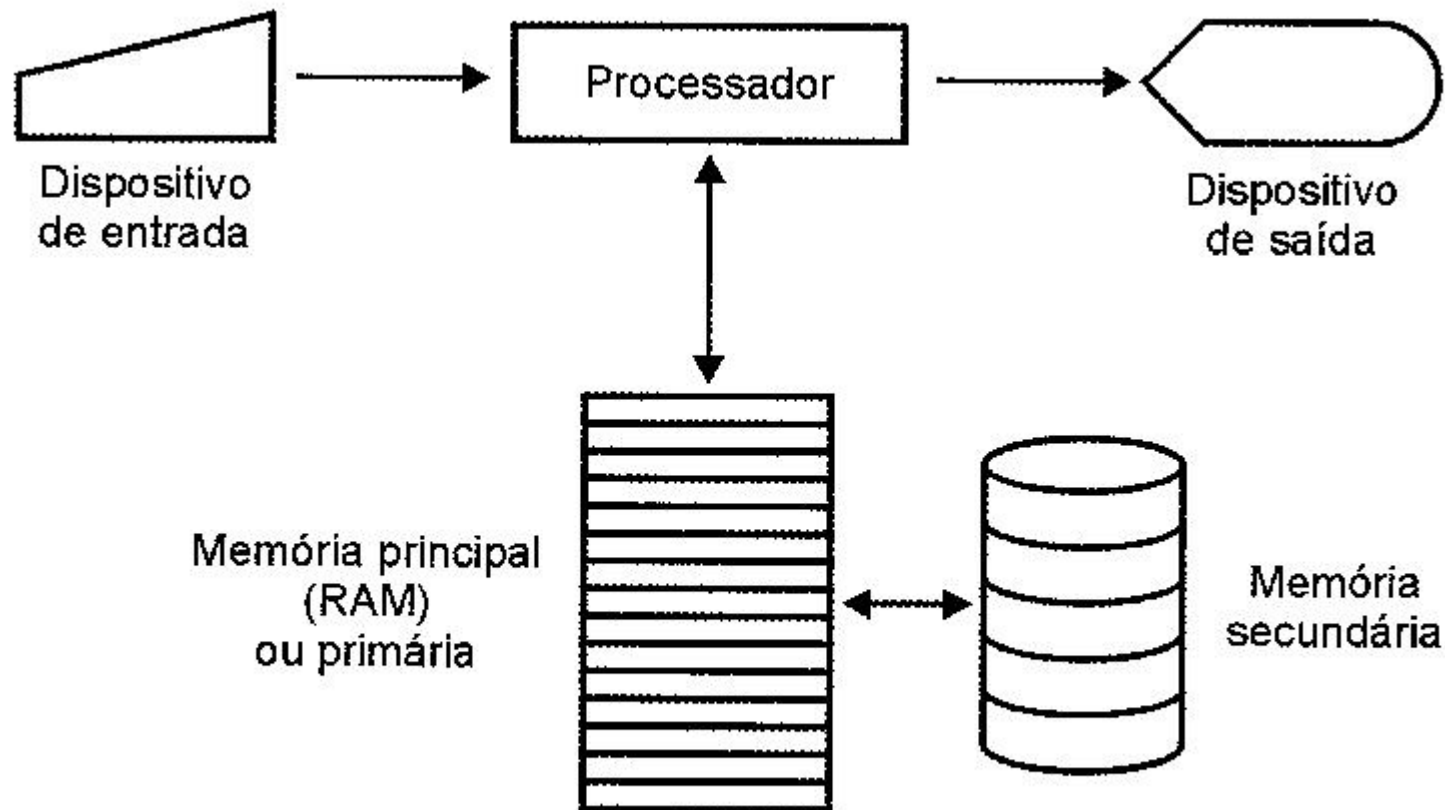


Figura 2.1 Componentes básicos de um computador.

Arquitetura de von Neumann

Exemplo

- ✓ **Atualização de saldos de contas bancárias**
 - *DOC – documento com número da conta e operação a ser realizada*
- ✓ **Especificação das operações necessárias**
 - *algoritmo*

Início do Programa
Enquanto houver DOC

Fazer

- Obter um DOC
- Ler número do DOC
- Encontrar conta com número = número do DOC
- Se tipo-DOC = depósito
Então: Novo-saldo = Saldo + Valor
- Se tipo-DOC = retirada
Então: Novo-saldo = Saldo – Valor
- Escrever Novo-saldo no lugar de Saldo

Fim do Fazer
Fim do Programa

Instruções Básicas

- ✓ **Algoritmo precisa ser detalhado com instruções que possam ser entendidas pela máquina (hardware)**
- ✓ **Exemplo – instruções básicas:**
 - *Somar dois números de cada vez*
 - *Mover um número de local para outro*
 - *Ler um caractere correspondente a uma tecla pressionada*
 - *Etc.*

Exemplo (Cont.)

- ✓ **Algoritmo precisa ser detalhado com instruções que possam ser entendidas pela máquina (hardware)**

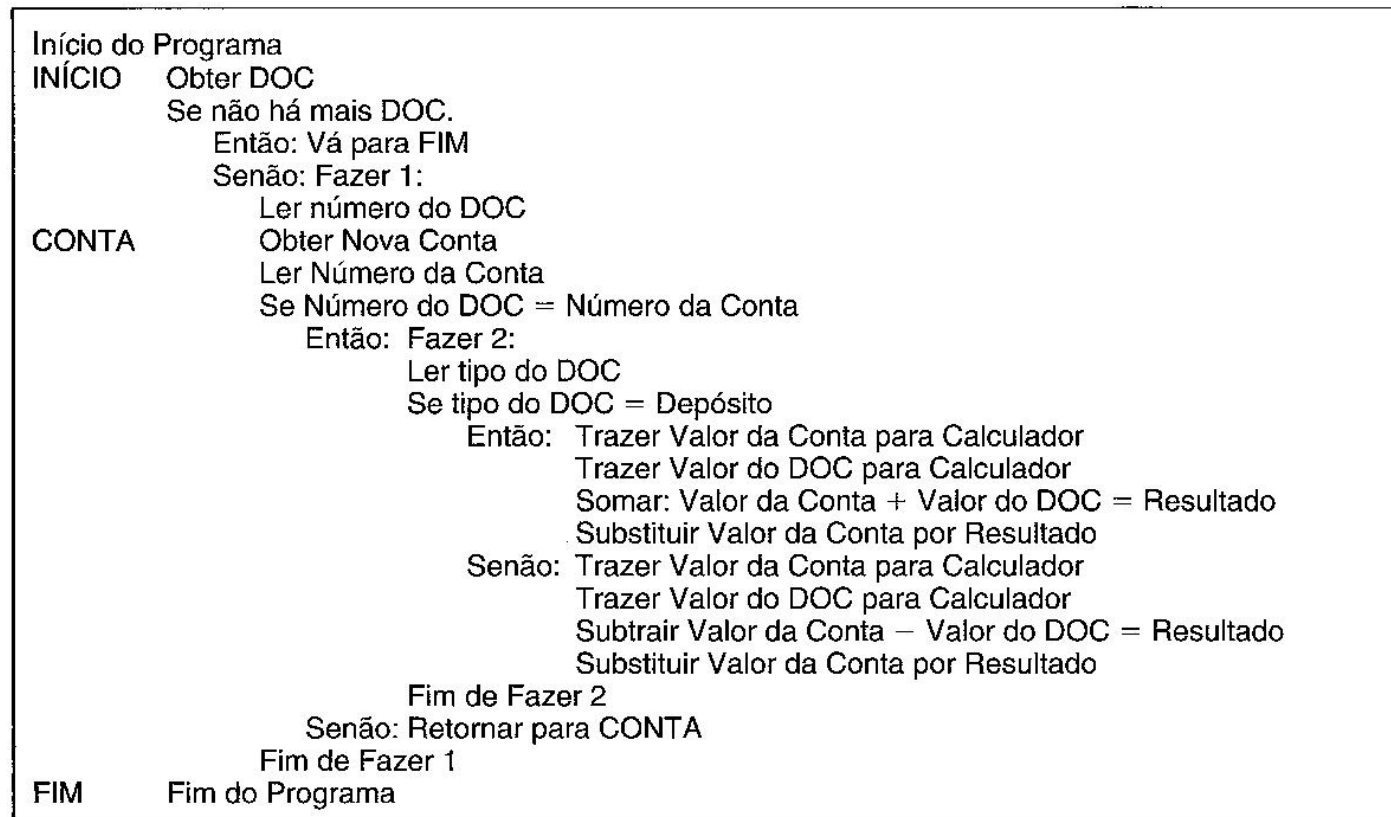


Figura 2.2(b) Descrição mais detalhada do algoritmo da Fig. 2.2(a).

Exemplo (Cont.)

- ✓ Instruções de máquina são de baixo nível
- ✓ Programadores utilizam linguagens de programação – alto nível

```
void main()
{
    printf ("Digite o número do documento: ");
    scanf ("%d", &num_doc);
    ARQ_CONTAS = fopen ("Arquivo de Contas dos Clientes", "rw");
    while (num_doc != 0)
    {
        printf ("Informe o número da conta: ");
        scanf ("%d", &num_conta);
        printf ("Informe o valor");
        scanf ("%f", &valor);
        printf ("Qual o tipo do lançamento (D) Depósito ou R (Retirada)");
        scanf ("%c", &tp_lanc);
        fseek (ARQ_CONTAS, sizeof(RegCliente)*(num_conta-1), SEEK_SET);
        fread (&RegCliente, sizeof(RegCliente), 1, ARQ_CONTAS);
        if (tp_lanc == "D")
            RegCliente.Saldo = RegCliente.Saldo + valor;
        else
            RegCliente.Saldo = RegCliente.Saldo - valor;

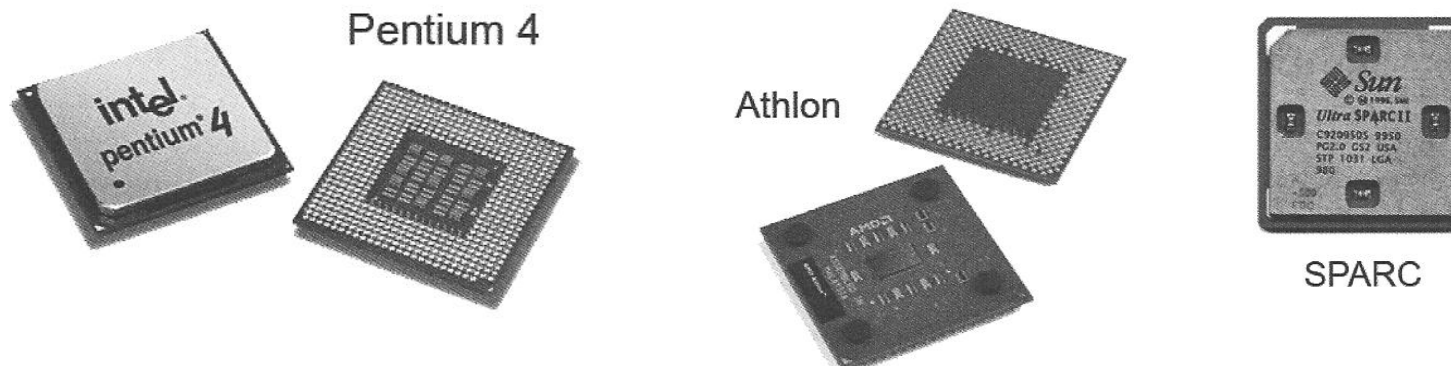
        fseek (ARQ_CONTAS, sizeof(RegCliente)*(num_conta-1), SEEK_SET);
        fwrite (RegCliente, sizeof(RegCliente), 1, ARQ_CONTAS);
        printf ("Digite o número do documento: ");
        scanf ("%d", &num_doc);
    }
    fclose (ARQ_CONTAS);
}
```

Figura 2.2(c) Programa em C para o algoritmo da Fig. 2.2(b).

Processador

Fundamentos de Arquiteturas de Computadores

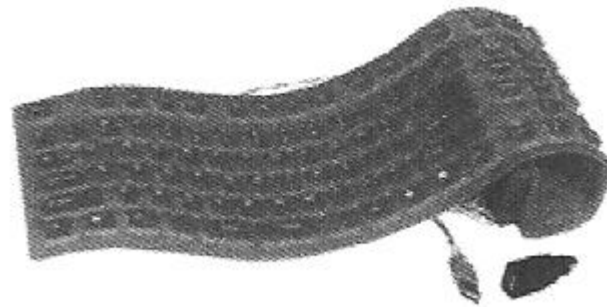
- ✓ UCP – Unidade Central de Processamento
 - *CPU – Central Processing Unit*
- ✓ Capaz de entender e executar uma operação definida por uma instrução de máquina
- ✓ Formados por milhões de minúsculos circuitos e componentes eletrônicos (transistores, resistores, etc.) encapsulados em um único invólucro (chip)



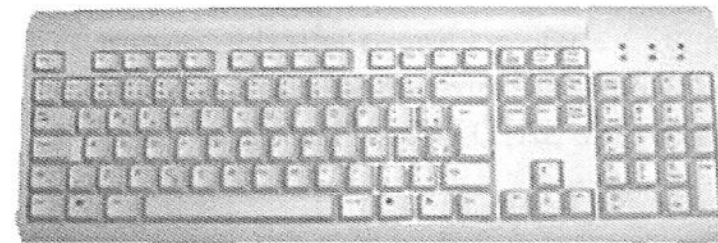
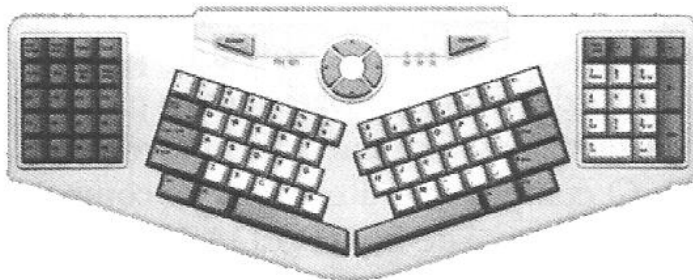
Dispositivos de Entrada

Fundamentos de Arquiteturas de Computadores

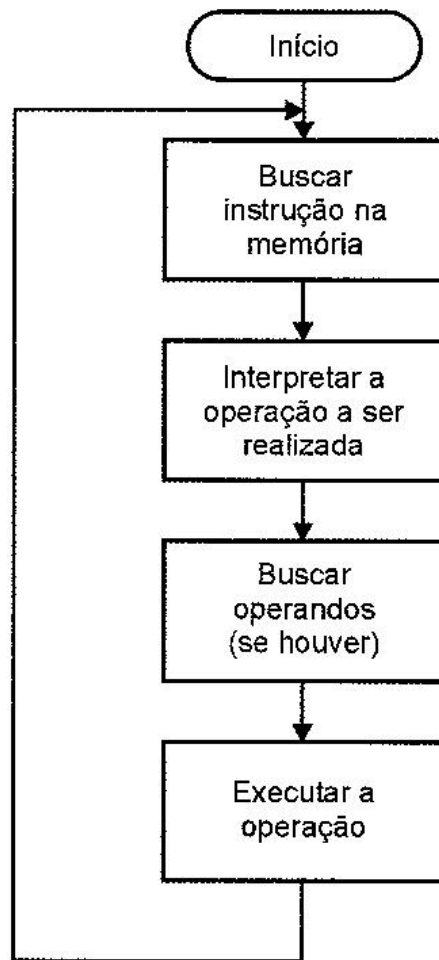
- ✓ **Necessários para introdução do programa e dos dados no sistema**
 - *Teclado, mouse, etc.*



Mouse



Memória

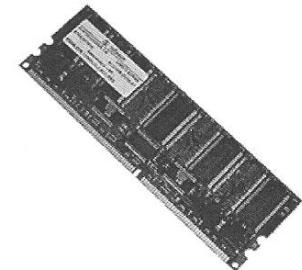


- ✓ Responsável pelo armazenamento das informações introduzidas por dispositivos de entrada

- *Registadores, memória cache, memória principal (RAM), memória secundária*

- ✓ Processador lê instruções da memória

Figura 2.6 Ciclo básico de instrução.



Dispositivos de Saída

- ✓ Utilizados para apresentar os resultados do programa ao usuário

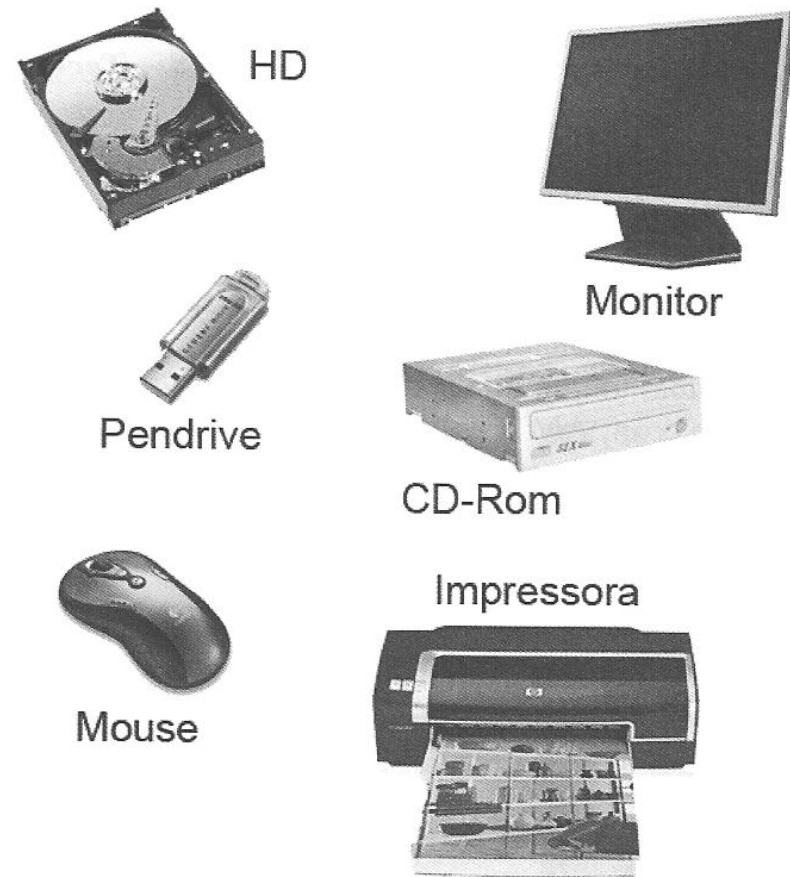


Figura 2.5 Exemplos de dispositivos de E/S.

Interconexão entre os componentes

- ✓ Sistema trabalha com sinais elétricos indicando os valores de um bit (0 ou 1)
- ✓ Barramento (*bus*)
 - *Conjunto de fios que conduzem os sinais entre os componentes principais do sistema*

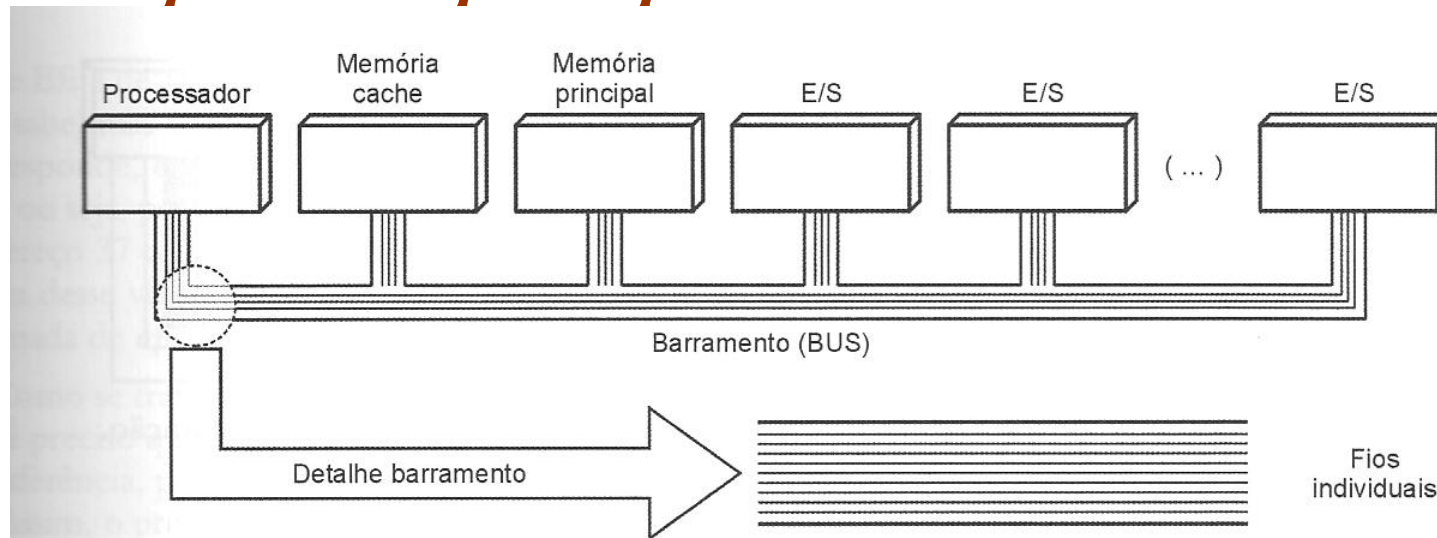


Figura 2.8 Modelo de interconexão entre componentes de um sistema de computação.

Barramento

- ✓ **Barramento é único e dividido em conjuntos de fios para cada funcionalidade**
 - *Barramento de dados (BD)*
 - *Barramento de endereços (BE)*
 - *Barramento de controle (BC)*

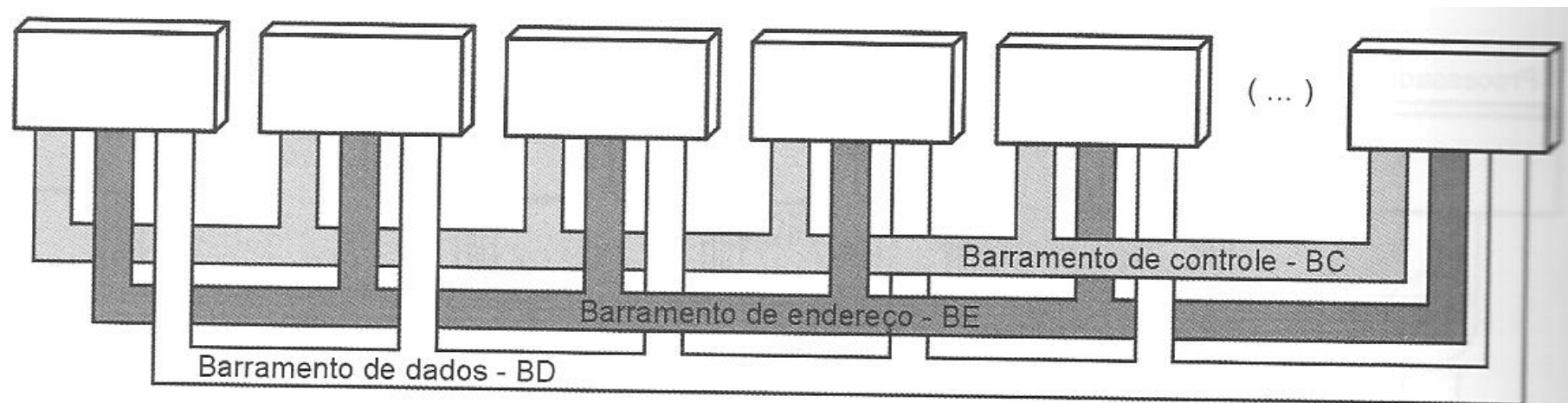
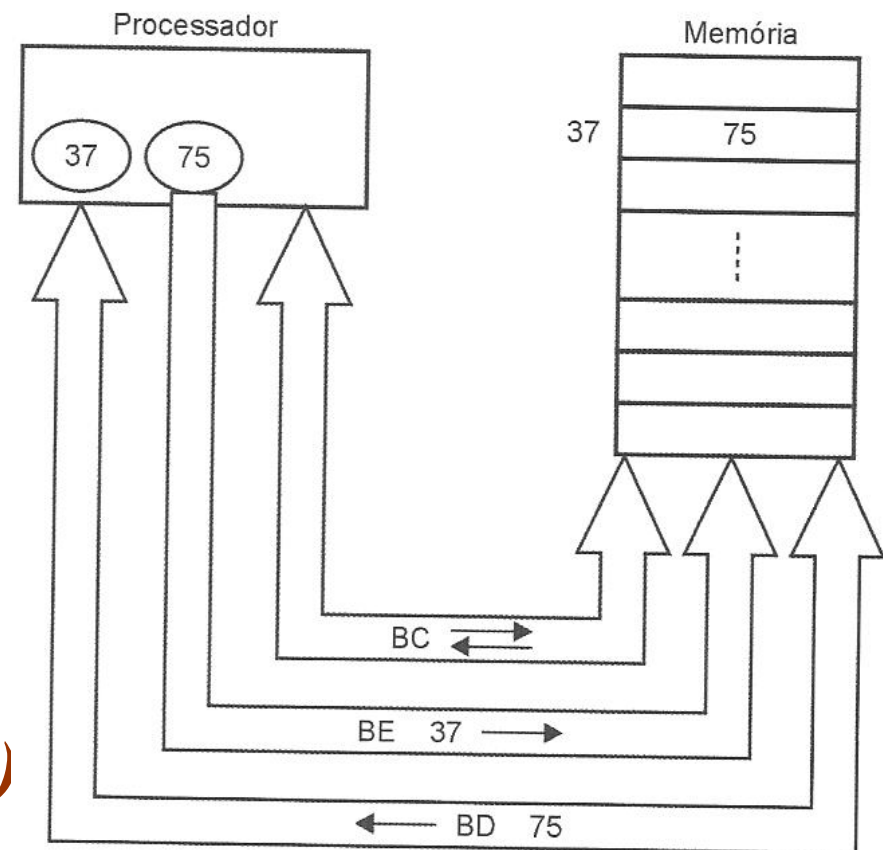


Figura 2.9 Modelo de interconexão de componentes de um sistema de computação.

Exemplo

✓ Processador requer leitura de dado da memória

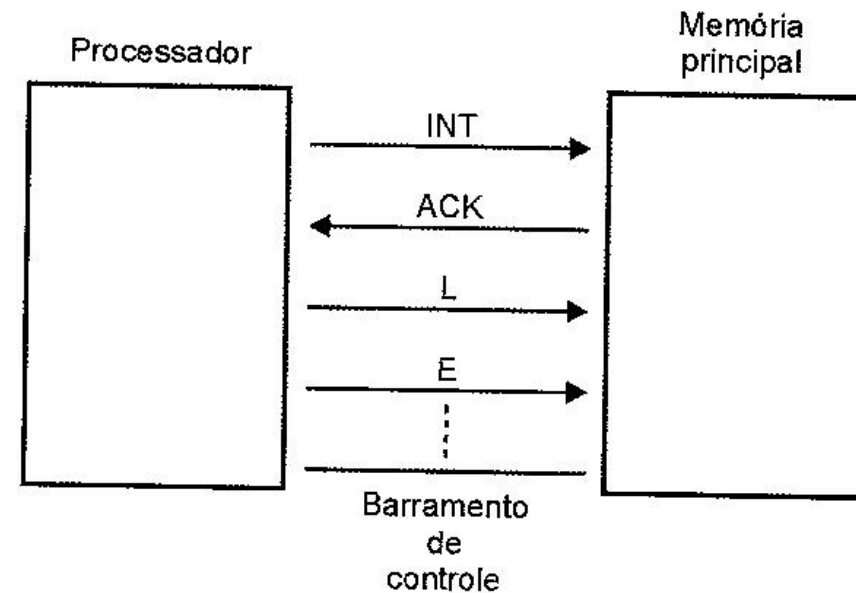
- *BC usado para comunicação entre processador e memória*
- *Valor do endereço é 37 (0000100101 – BE de 10 bits)*
- *Valor do dado é 75 (01001011 – BD de 8 bits)*



Barramento de Controle

✓ BC – fios independentes

- *Cada um com função específica*



LEGENDA:

INT - interrogação

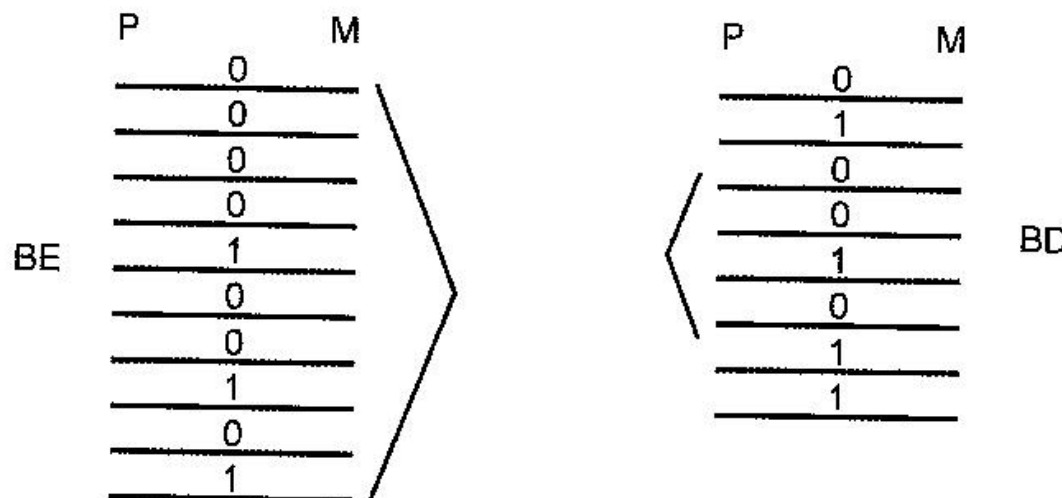
ACK - OK

L - Sinal de leitura (read)

E - Sinal de escrita (write)

Barramentos de Endereços e Dados

- ✓ **BE** – largura **L** representa a quantidade de fios
 - 2^L indica quantidade de endereços
- ✓ **BD** – largura **L** e velocidade **V**
 - *Taxa de transferência* $T = L \times V$
 - L = 10 bits e V = 100MHz (Mbps)
 - T = 1000Mbps



Computadores Atuais

- ✓ **Um único barramento é ineficaz, pois velocidades de acesso aos dispositivos são muito diferentes. Ex.:**
 - *processador e memória – velocidades elevadas*
 - *teclado e mouse – taxas de transferência muito baixas*
- ✓ **Solução => diferentes barramentos conectados por pontes**
 - *Barramento do sistema (processador, memória principal e memória cache) – system bus*
 - *Barramento de E/S de alta velocidade (HDs, placa de rede, placa de vídeo, etc.)*
 - *Barramento de E/S de baixa velocidade (teclado, mouse, impressora, scanner, etc.)*

Barramentos

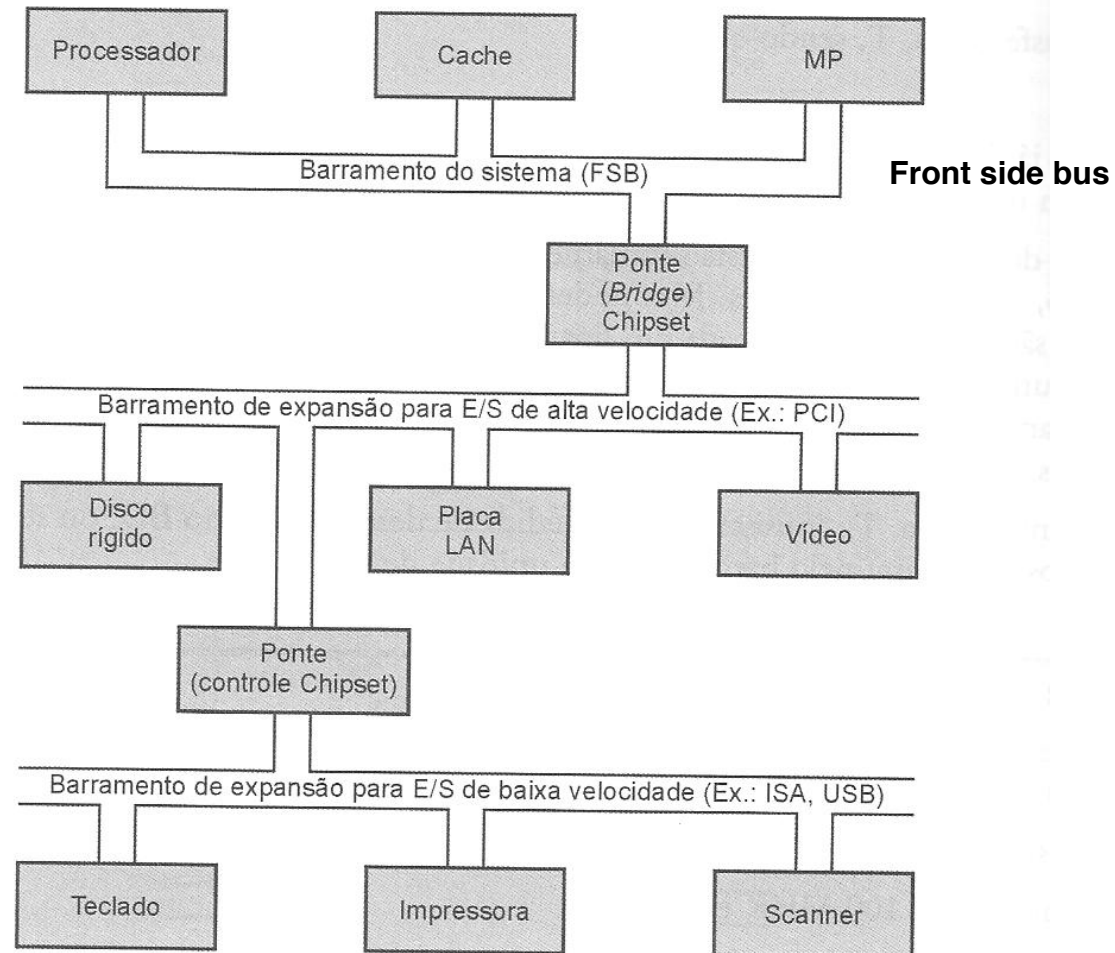


Figura 2.12 Exemplo de modelo de sistema com vários barramentos.

Representação das Informações

- ✓ bit – binary digit – 0 ou 1

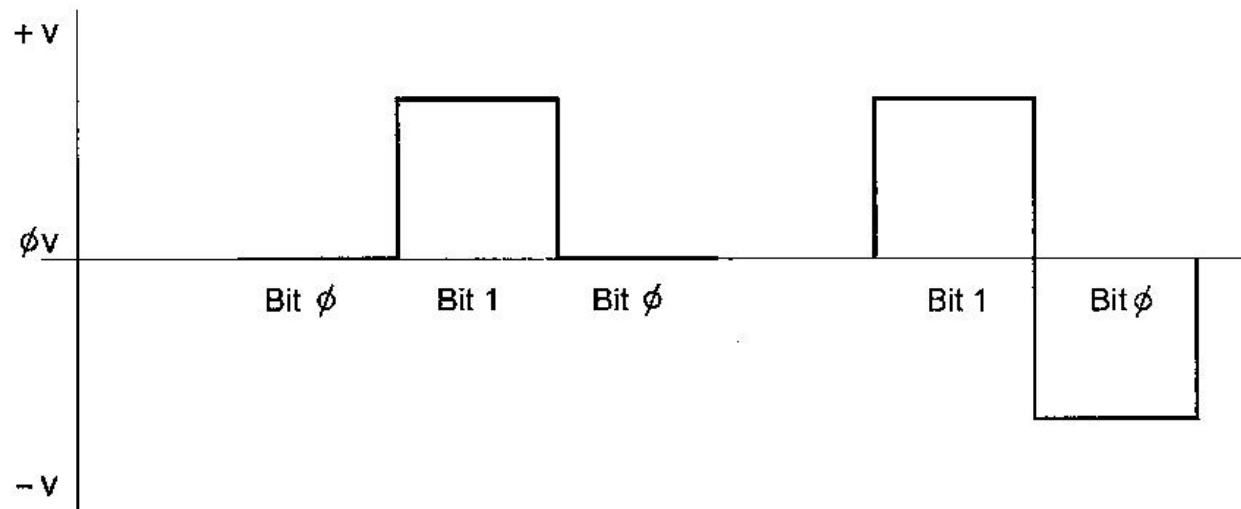


Figura 2.13 Representação de 1 bit.

Representação das Informações

- ✓ Símbolos são definidos como grupos de bits
- ✓ Byte – grupo ordenado de 8 bits
 - *proposto pela IBM como unidade de armazenamento e transferência*
- ✓ Múltiplos do byte em potência de 2
 - *KB, MB, GB, etc.*

Tabela 2.1 Grandezas Usadas para Abreviar Valores em Computação

Nome da unidade	Valor em potência de 2	Valor em unidades
1K (1 quilo)	2^{10}	1024
1M (1 mega)	$1024K = 2^{20}$	1.048.576
1G (1 giga)	$1024M = 2^{30}$	1.073.741.824
1T (1 tera)	2^{40}	1.099.511.627.776
1P (1 peta)	2^{50}	1.125.899.906.843.624
1Ex (1 exa)	2^{60}	1.152.921.504.607.870.976
1Z (1 zeta)	2^{70}	1.180.591.620.718.458.879.424
1Y (1 yotta)	2^{80}	1.208.925.819.615.701.892.530.176

Representação das Informações

- ✓ **Símbolos são definidos como grupos de bits**
- ✓ **Caractere – símbolo alfanumérico usado em textos**
 - *Códigos de 6 bits – BCD (64 símbolos)*
 - *Códigos de 7 bits – ASCII 7 bits (128 símbolos)*
 - *Códigos de 8 bits – ASCII 8 bits, EBCDIC (256 símbolos)*
 - *Códigos de 16 bits – UNICODE (65536 símbolos)*

Tabela ASCII - Exemplo

Binário	Decimal	Hexa	Glifo
0010 0000	32	20	
0010 0001	33	21	!
0010 0010	34	22	"
0010 0011	35	23	#
0010 0100	36	24	\$
0010 0101	37	25	%
0010 0110	38	26	&
0010 0111	39	27	'
0010 1000	40	28	(
0010 1001	41	29)
0010 1010	42	2A	*
0010 1011	43	2B	+
0010 1100	44	2C	,
0010 1101	45	2D	-

0010 1111	47	2F	/
0011 0000	48	30	0
0011 0001	49	31	1
0011 0010	50	32	2
0011 0011	51	33	3
0011 0100	52	34	4
0011 0101	53	35	5
0011 0110	54	36	6
0011 0111	55	37	7
0011 1000	56	38	8
0011 1001	57	39	9
0011 1010	58	3A	:
0011 1011	59	3B	;

Binário	Decimal	Hexa	Glifo
0100 0000	64	40	@
0100 0001	65	41	A
0100 0010	66	42	B
0100 0011	67	43	C
0100 0100	68	44	D
0100 0101	69	45	E
0100 0110	70	46	F
0100 0111	71	47	G
0100 1000	72	48	H
0100 1001	73	49	I
0100 1010	74	4A	J
0100 1011	75	4B	K
0100 1100	76	4C	L
0100 1101	77	4D	M

Representação das Informações

- ✓ **Byte – 8 bits – unidade de armazenamento**
- ✓ **Palavra (*word*) – unidade de transferência e processamento**
 - *Valor comum é de 32 bits (pentium IV, Athlon XP)*
 - *64 bits em processadores mais novos (Athlon64)*

Tabela 2.2 Estrutura de Informações nas Linguagens dos Humanos e nos Computadores

Computadores	Linguagens dos humanos
Bit	Caractere
Byte e caractere	Palavra
Palavra	Frases
Registro	Textos
Arquivo	Livros
Banco de dados	

Arquivos e Registros

- ✓ **Arquivos – conjunto de dados (ou informações) de um mesmo tipo para uma mesma aplicação**
 - *Arquivo de alunos da turma*
 - *Arquivo com o código do programa (instruções)*
- ✓ **Cada arquivo é constituído por itens individuais chamados registros**
 - *Arquivos de 60 alunos possui 60 registros*
 - *Um programa é um arquivo de um único registro*
- ✓ **Arquivos são armazenados em memória secundária**

Unidades de Medida

Tabela 2.3 Unidades de Medida de Espaço Muito Pequeno

Unidade	Descrição
Mícron	10^{-3} mm (1 milésimo do milímetro)
Nanômetro (nm)	10^{-6} mm (1 milionésimo do milímetro) ou 1 milésimo do micron
Angström	10 nanômetros

Tabela 2.4 Unidades de Medida de Tempo Muito Curto

Unidade	Descrição
Milissegundo	10^{-3} do segundo
Microsegundo	10^{-6} do segundo
Nanosegundo	10^{-9} do segundo
Picosegundo	10^{-12} do segundo

Medidas de Desempenho

- ✓ Desempenho depende da capacidade e velocidade dos diferentes componentes e da velocidade com que eles se comunicam entre si
- ✓ Desempenho dos processadores (velocidade)
 - *Milhões de instruções por segundo – MIPS*
 - *Milhões de operações de ponto flutuante por segundo (MFLOPS) – supercomputadores*
- ✓ Desempenho da memória (tempo de acesso)
 - *Tempo para recuperação ou escrita de dados*
- ✓ Desempenho do sistema (tempo de resposta)
 - *Tempo entre requisição de um serviço e a resposta*
- ✓ Desempenho da comunicação/transferência
 - *vazão – throughput – bps (bit por segundo)*