



EAD
UNISANTA

ARQUITETURA DE COMPUTADORES

Dr. Joseffe Barroso de Oliveira

**GUIA DA
DISCIPLINA**

1. O COMPUTADOR - ARQUITETURA BÁSICA

Objetivo

Compreender os conceitos iniciais de informática, diferenciar os principais elementos de Hardware e Software, apresentar os principais itens de Hardware e Software e compreender os termos utilizados no meio computacional.

Introdução

Um computador é uma máquina (conjunto de partes eletrônicas e eletromecânicas) capaz de sistematicamente coletar, manipular e fornecer os resultados da manipulação de dados para um ou mais objetivos. Por ser uma máquina composta de vários circuitos e componentes eletrônicos, também é chamado de equipamento de processamento eletrônico de dados.

O esquema básico de um processamento de dados (manual ou automático) resulta em um produto acabado: a informação. Assim, os dados precisam ser processados para que algum resultado tenha significado para alguém ou para o próprio computador.

O computador processa os dados, orientado por um conjunto de instruções, para produzir resultados completos com um mínimo de intervenção humana. Entre seus vários benefícios, podemos citar:

- a) grande velocidade no processamento e disponibilização de informações;
- b) precisão no fornecimento das informações;
- c) adequação para execução de tarefas repetitivas;
- d) redução de custos operacionais;
- e) compartilhamento de dados.

A busca de técnicas mais eficazes de processamento de dados, aliada ao natural avanço tecnológico em diversos outros ramos de atividade, como a eletrônica e a mecânica, por exemplo, conduziu ao desenvolvimento de equipamentos de processamento eletrônico de dados - os computadores - capazes de coletar, armazenar e processar dados muito mais rapidamente que os antigos meios manuais.

1.1. Hardware e Software

Um sistema baseado em computador é, na verdade, composto por hardware e software. Hardware é o nome que se dá para a parte física do computador. É tudo que você pode tocar (mouse, teclado, caixas de som, placas, fios, componentes em geral). Software é o nome que se dá a toda parte lógica do computador, ou seja, os programas que você vê funcionar na tela e que dão “vida” ao computador. Sem um software adequado às suas necessidades, o computador, por mais bem equipado e avançado que seja, é completamente inútil.

1.2. Unidades de Entrada e Saída (E/S)

Os dispositivos de E/S (Entrada e Saída) são constituídos, geralmente, de duas partes: o controlador e o dispositivo propriamente dito. O controlador é um chip ou um conjunto de chips que controla fisicamente o dispositivo; ele recebe comandos do sistema operacional (software), por exemplo, para ler dados dos dispositivos e para enviá-los (TANEMBAUM, 2013).

Unidades de entrada são, portanto, dispositivos físicos que capturam os dados a serem processados. Os dados podem ser do tipo texto, vídeo ou áudio. Para cada tipo de dado temos um dispositivo de entrada específico para sua leitura: teclado, mouse, scanner, etc.

Unidades de saída apresentam os resultados finais do processamento. Para o usuário, os dispositivos de saída são tão normais que quase não são percebidos: monitores de vídeo, impressoras, plotter, telas de terminais bancários, impressoras de extratos, painéis de senha, monitores de vídeo, quiosques de consultas de preços, etc.

Existem também alguns dispositivos que podem ser classificados com ambas as denominações, entrada e saída, por exemplo: unidades de disco (discos rígidos, disquetes, unidades leitoras e gravadoras de CD e DVD), unidades de armazenamento USB (Universal Serial Bus – Barramento Serial Universal) – conhecidos como pen drives –, telas touch screen, etc.



1.3. Memória

O computador deve ser dotado de alguma forma de armazenamento (temporário ou permanente) para que os dados coletados ou processados possam ser armazenados. A essa estrutura damos o nome genérico de memória (não está contextualizado aqui o tipo da memória). A memória armazena, essencialmente, os bits. A menor unidade de informação em um computador é o bit, que pode assumir os valores 0 ou 1.

Como um único bit é insuficiente para representar informações mais complexas, eles são agrupados e combinados. Num primeiro agrupamento, são reunidos em conjuntos de oito bits, recebendo a denominação de Byte.

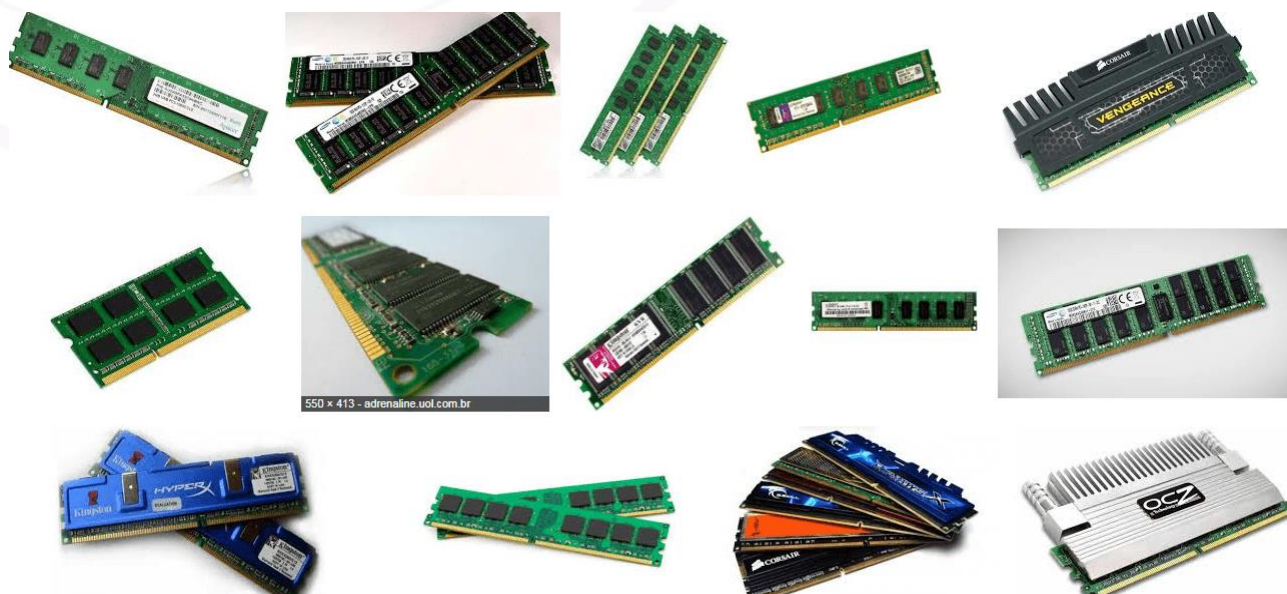
Como a unidade byte (unidade de medida de armazenamento) também é, consideravelmente, pequena quando indicamos valores mais extensos, utilizamos múltiplos do byte: quilobyte, megabyte, gigabyte, terabyte, etc.

1 Quilobyte = 1 KB = 1024 Bytes,

1 Megabyte = 1 MB = 1024 Kbytes,

1 Gigabyte = 1 GB = 1024 Mbytes, e

1 Terabyte = 1 TB = 1024 Gbytes

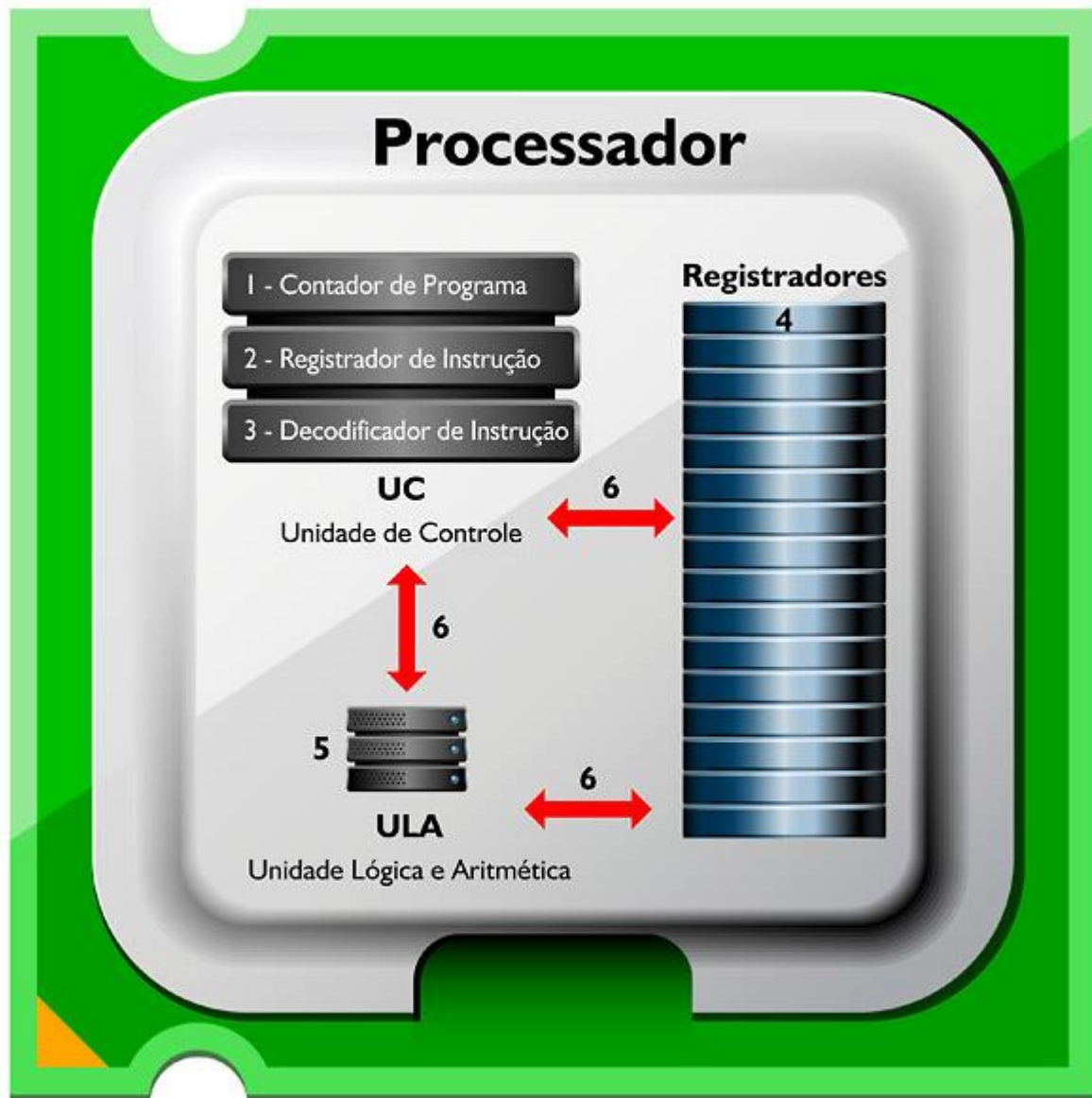


1.4. Unidade Central de Processamento (UCP ou CPU)

A CPU (Central Processing Unit ou Unidade Central de Processamento), microprocessador ou processador é a parte do computador que interpreta e executa as instruções contidas no software. Na maioria das CPUs essa tarefa é dividida entre uma unidade de controle que dirige o fluxo do programa e uma ou mais unidades que executam operações em dados.

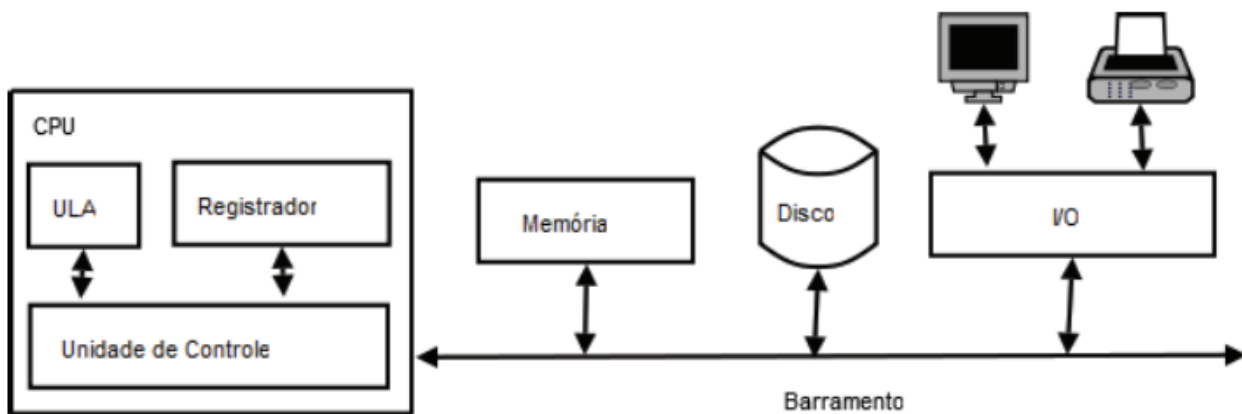
O ciclo básico de execução de qualquer CPU é buscar a primeira instrução da memória, decodificá-la para determinar seus operandos e qual operação executar com os mesmos, executá-la e então buscar, decodificar e executar a instrução subsequente (TANEMBAUM, 2013).

A CPU é constituída pelos seguintes componentes: a ULA (Unidade de Lógica e Aritmética), a UC (Unidade de Controle) e os registradores. Um tipo de registrador especial é o contador de programa, que contém o endereço de memória da próxima instrução que a CPU vai buscar. Assim, esse registrador vai ser atualizado para conter sempre o endereço da próxima instrução a ser processada.



1.5. Componentes de um computador

A figura abaixo mostra a arquitetura simplificada de um computador. Os computadores modernos podem ter uma arquitetura bem mais complexa, mas os blocos principais estão aqui representados.



O principal módulo é a Unidade Central de Processamento, usualmente conhecida como CPU (Central Processing Unit). A CPU é o “cérebro” do computador, onde todo o processamento é realizado. Ela é constituída por três submódulos: a Unidade Lógica e Aritmética (ULA), os Registradores e a Unidade de Controle. Nos computadores modernos, a CPU é construída em um único Circuito Integrado (chip).

A Unidade Lógica e Aritmética (ULA), também conhecida como Arithmetic and Logic Unit (ALU), é responsável por realizar as operações do computador, como soma e subtração, assim como as funções lógicas, OU, E, NÃO e OU Exclusivo.

Algumas ULAs mais sofisticadas realizam também multiplicações e divisões. Os registradores são memórias utilizadas para realizar as operações na CPU. Essas memórias são muito rápidas e de tamanho pequeno, geralmente suficiente para guardar menos de uma dezena de valores. Além dos registradores, é comum haver uma memória de rascunho dentro da CPU, chamada de Cache. Ela é utilizada para aumentar a velocidade de processamento, reduzindo o tempo de acesso à memória externa.

A Unidade de Controle é responsável por controlar todo o funcionamento da CPU e, também, de todo o computador. Ela controla o processamento entre ULA e Registrador e interage com o barramento externo onde ficam os periféricos. A Unidade de Controle tem uma função chave, que é a interpretação do código do programa que irá nortear os comandos do processamento.

Já o barramento serve para interligar todos os componentes internos do computador. Para se obter altas velocidades, esse barramento é, geralmente, paralelo (os dados são

enviados paralelamente). Nessa figura representamos apenas um barramento, mas, geralmente, ele é dividido em barramento de dados (onde os dados trafegam), barramento de endereço (onde indicamos a localização de cada dado) e o barramento de controle (onde indicamos o comando, por exemplo, ler, escrever, copiar, etc). Em um computador real, existem vários níveis de barramento com velocidades diferentes. Quanto mais próximo da CPU mais rápido ele é, e quanto mais próximo dos periféricos mais lento ele é.

A Memória é constituída por um conjunto de memória semicondutora utilizada para armazenar os dados temporariamente. Podemos dizer que essa é a memória de trabalho do computador e geralmente fica localizada na placa mãe do computador. Essa memória é mais lenta que os registradores, mas é mais rápida que as unidades de armazenamento em disco. A memória semicondutora geralmente perde os dados quando o computador é desligado, por isso deve-se usar uma unidade de armazenamento permanente, o disco.

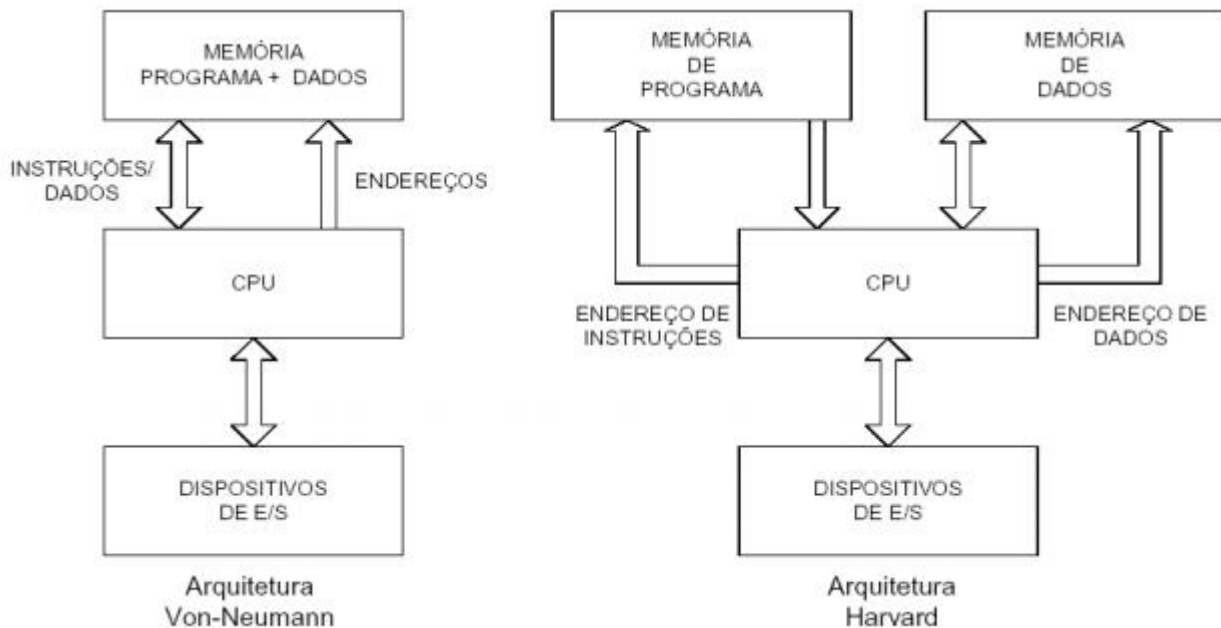
O disco consiste em uma unidade que armazena os dados em um disco magnético ou não, que mantém as informações mesmo quando o computador é desligado. Essa unidade tem grande capacidade de armazenamento, mas tem velocidade de acesso significativamente menor em relação às memórias semicondutoras.

O módulo de Entrada e Saída, ou Input/Output (I/O), estabelece a ligação do computador com o mundo externo, usando equipamentos periféricos. Essa interface permite a ligação de teclados e mouses (para entrada de dados), monitores ou impressoras (para exibição dos dados) e placas de comunicação (para trocar dados a longa distância). Essa interface é significativamente mais lenta que os demais componentes do computador devido à natureza dos periféricos e à incapacidade humana de processar informações na velocidade dos computadores.

1.6. Arquiteturas: Von-Neumann X Harvard

Existem duas arquiteturas clássicas para os microprocessadores em geral: a arquitetura Von-Neumann, onde existe apenas um barramento interno por onde circulam instruções e dados e a arquitetura Harvard, que é caracterizada por dois barramentos internos, sendo um de instruções e outro de dados. Pode-se dizer que a primeira é uma arquitetura serial e a segunda paralela; da mesma forma, pode-se dizer que a arquitetura Von-Neumann permite produzir um conjunto complexo de código de instruções para o

processador (CISC – Complex Instructions Set Computer), com um tempo de execução por instrução de vários ciclos de clock. Já a arquitetura Havard produz um conjunto simples de códigos de instruções e, dado ao paralelismo de sua estrutura, é capaz de executar apenas uma instrução por ciclo de clock. A arquitetura Von-Neumann é mais simples, com menor número de portas lógicas, entretanto, sua velocidade é menor que a Havard. A arquitetura Havard necessita de mais linhas de código para executar a mesma tarefa. Enquanto uma arquitetura Von-Neumann possui muito mais tipos de instruções.



2. O COMPUTADOR - ARQUITETURA BÁSICA

Objetivo

Compreender a linguagem numérica utilizada pelos computadores, as formas numéricas que se apresentam num sistema computacional e as formas de conversão de bases dos sistemas reconhecidos pelo homem e pela máquina. Compreender também os principais tipos de dados e sua representação interna e o formato de apresentação dos dados no âmbito computacional.

Introdução

Na natureza, todo tipo de informação pode assumir qualquer valor compreendido em um intervalo de $-\infty$ a $+\infty$. Você consegue distinguir, por exemplo, uma cor vermelha que esteja um pouco mais clara de outro tom de vermelho. Esse tipo de informação é informação analógica. Computadores lidam apenas com informações digitais, não abrindo espaço para dados estruturados de forma analógica, que podem assumir valores indefinidos.

2.1. Sistemas de numeração

O sistema de numeração decimal está tão presente no nosso cotidiano que nem notamos sua presença. O kg (quilograma), o m (metro), o Km/h (quilômetro por hora), tudo isso é medido numa forma que compreendemos no modelo decimal. Entretanto, este não é o modelo utilizado pelos computadores, pois as posições de memória basicamente conseguem representar apenas 0 ou 1 (como se estivesse ligado ou desligado). Por isso, existem outras notações, como a binária e a hexadecimal.

Essas notações são obstáculos para os iniciantes. Tornam-se, entretanto, simples depois de recordarmos o sistema de base decimal.

2.2. Decimal

Quando falamos do número 123, imaginamos certo número de itens que esse número representa e esquecemos o seu significado matemático. Na realidade 123 representa:

$$(1 \times 10^2) + (2 \times 10^1) + (3 \times 10^0), \text{ ou seja: } 100 + 20 + 3 = 123$$

Observe que cada algarismo é multiplicado por uma potência de 10. Os expoentes de 10 são numerados da direita para a esquerda começando com 0 (zero).

2.3. Binário

O sistema binário funciona exatamente da mesma forma (nossa cabeça é que insiste em dar nó). O sistema decimal possui dez dígitos (de 0 a 9); o binário possui apenas dois (0 e 1). Fazemos potência de 10 para calcular o número no sistema decimal, então faremos potência de 2 para o sistema binário.

Por exemplo: o valor binário 11001010 representa o valor decimal 202, assim:

$$(1 \times 2^7) + (1 \times 2^6) + (0 \times 2^5) + (0 \times 2^4) + (1 \times 2^3) + (0 \times 2^2) + (1 \times 2^1) + (0 \times 2^0) + (0 \times 2^0), \text{ ou seja: } 128 + 64 + 0 + 0 + 8 + 0 + 2 + 0 = 202$$

2.4. Hexadecimal

O sistema hexadecimal representa os números em base 16. É usado na informática, pois os computadores costumam utilizar o byte como unidade básica da memória e com um byte podemos representar 256 valores possíveis, o que abrange todo alfabeto (maiúsculas e minúsculas), os números e vários caracteres especiais.

Como no sistema decimal dispomos de apenas dez dígitos, devemos incluir seis letras para representar o hexadecimal. O conjunto de algarismos hexadecimais fica, portanto, assim:

0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F. Nesse sistema, o **A** vale **10**, o **B** vale **11**, etc, até o **F**, que vale **15**.

Cada algarismo é multiplicado por uma potência de 16. Os expoentes de 16 são numerados da direita para a esquerda começando com 0 (zero).

Exemplo:

$$3E0 = 3 \times 16^2 + 14 \times 16^1 + 0 \times 16 = 992$$

Na relação entre sistema binário e decimal, cada algarismo hexadecimal corresponde a um número binário de 4 bits. Por exemplo, no hexadecimal acima temos:

$$3 = 0011 = 3 \text{ decimal}$$

$$E = 1110 = 14 \text{ decimal}$$

$$0 = 0000 = 0 \text{ decimal}$$

2.5. Conversão de decimal para binário

Dividimos o número decimal seguidamente por 2 e utilizamos o resto de cada divisão para compor o número binário. Observe o exemplo a seguir.

Converter 45 (decimal) para binário:

$45 / 2 = 22$	sobra 1
$22 / 2 = 11$	sobra 0
$11 / 2 = 5$	sobra 1
$5 / 2 = 2$	sobra 1
$2 / 2 = 1$	sobra 0
$1 / 2 = 0$	sobra 1

Como não dá mais para dividir, paramos. Pegamos todos os restos olhando de baixo para cima. Concluimos que o número 45 em decimal equivale 101101 em binário.

2.6. Conversão de decimal para hexadecimal

Por um processo parecido com o que já vimos, dividimos o número decimal seguidamente por 16 e utilizamos o resto de cada divisão para compor o hexadecimal. A única diferença é que o divisor é o 16. Veja o exemplo a seguir:

Converter 970 (decimal) para hexadecimal:

$970 / 16 = 60,$	resto 10 = A
$60 / 16 = 3,$	resto 12 = C
$3 / 16 = 0,$	resto 3 = 3

Temos então que o número 970 (decimal) representa 3CA em hexadecimal (novamente pegamos os restos das divisões, de baixo para cima).

A simbologia hexadecimal não nos permite escrever o número como sendo 31210 ou 3.12.10. Então, substituímos 12 e 10 por C e A, respectivamente. Substituindo os valores de acordo com a figura acima, temos o resultado 3CA.

2.7. Conversão de binário para decimal

Você pode montar uma tabela com as seguintes linhas:

- Na primeira: escreva o número binário;
- Na segunda: escreva os expoentes de 2, da direita para a esquerda, começando com 0;
- Na terceira: calcule as potências de 2;
- Na quarta: multiplique a primeira linha pela terceira.

Some a quarta linha e terá o número binário convertido para decimal.

Veja o exemplo de conversão do número binário 101101 observando a tabela abaixo:

número em binário	1	0	1	1	0	1
expoentes	25	24	23	22	21	20
potências	32	16	8	4	2	1
primeira x terceira	32	0	8	4	0	1

A soma dos resultados da quarta linha é $32 + 8 + 4 + 1 = 45$.

2.8. Conversão de hexadecimal para decimal

Você pode usar a mesma tabelinha, trocando 2 por 16. Vamos converter o hexadecimal 3F8 para decimal (tabela) a seguir

número em hexadecimal	3	F	8
expoentes	16^2	16^1	16^0
potências	256	16	1
primeira x terceira	3 x 256	15 x 16	8

A soma dos resultados da quarta linha é o número decimal 1016. Você pode conferir os resultados de todos os itens anteriores utilizando a calculadora científica do Windows.

2.9. Tabela de conversão Decimal x Binário x Hexadecimal

Decimal	Binário	Hexadecimal
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7
8	1000	8
9	1001	9
10	1010	A
11	1011	B
12	1100	C
13	1101	D
14	1110	E
15	1111	F

2.10. Unidades de medida na computação

Um bit (binary digit), é a menor e mais elementar unidade que os computadores utilizam para trabalhar. Ele pode assumir apenas dois valores: 0 ou 1. Fisicamente, um bit pode ser representado de várias formas: via eletricidade (dois valores de voltagem aplicados num fio), via luz (em fibras ópticas), via ondas eletromagnéticas (redes sem fio), enfim, em tudo que seja possível identificar dois estados diferentes.

Exemplos de representações “binárias” do nosso mundo real:

Uma porta: fechada ou aberta;

Uma lâmpada: acesa ou apagada;

Estado civil: solteiro ou casado;

Já o byte, é a menor unidade de armazenamento utilizada pelos computadores. Um byte é uma sequência de 8 bits. Ou seja, não é possível salvar menos do que 8 bits. Por exemplo, em uma tabela de banco de dados onde você tem uma coluna do tipo tinyint, quando salvamos o número 0 nela, estamos na prática usando 1 byte e não 1 bit. O próprio tipo tinyint já especifica que ele usa 1 byte de armazenamento, não importando se você vai salvar nele o número 1 ou o 255.

Todo dispositivo de armazenamento indica o número de bytes (8 bits) que ele pode conter. Algumas unidades de medida facilitam esse agrupamento:

Unidade	Símbolo	Número de bytes
kilobyte	kB	$2^{10} = 1024$ bytes
megabyte	MB	$2^{20} = 1,048,576$ bytes
gigabyte	GB	$2^{30} = 1,073,741,824$ bytes
terabyte	TB	$2^{40} = 1,099,511,627,776$ bytes
petabyte	PB	$2^{50} = 1,125,899,906,842,624$ bytes
exabyte	EB	$2^{60} = 1,152,921,504,606,846,976$ bytes
zettabyte	ZB	$2^{70} = 1,180,591,620,717,411,303,424$ bytes
yottabyte	YB	$2^{80} = 1,208,925,819,614,629,174,706,176$ bytes

2.11. Códigos de representação - Tabela ASCII

A representação interna de informação em um computador é realizada pela especificação de uma correspondência entre o símbolo da informação e um grupo de algarismos binários (bits). Isso porque o computador, possuindo somente dois símbolos (0 ou 1) para representação, requer mais de um bit para identificar todos os possíveis símbolos que constituem as informações usadas pelo homem e que precisam ser armazenadas e processadas na máquina, conforme afirma Monteiro (2007).

Para podermos representar todos os caracteres alfabéticos e decimais é utilizado um método denominado de codificação. Nesse modelo, cada símbolo (caractere) da nossa linguagem é atribuído a um conjunto de bits que o identifica de forma única.

Dec	Hex	Oct	Chr	Dec	Hex	Oct	HTML	Chr	Dec	Hex	Oct	HTML	Chr	Dec	Hex	Oct	HTML	Chr
0	0	000	NULL	32	20	040	 	Space	64	40	100	@	@	96	60	140	`	`
1	1	001	Start of Header	33	21	041	!	!	65	41	101	A	A	97	61	141	a	a
2	2	002	Start of Text	34	22	042	"	"	66	42	102	B	B	98	62	142	b	b
3	3	003	End of Text	35	23	043	#	#	67	43	103	C	C	99	63	143	c	c
4	4	004	End of Transmission	36	24	044	$	\$	68	44	104	D	D	100	64	144	d	d
5	5	005	Enquiry	37	25	045	%	%	69	45	105	E	E	101	65	145	e	e
6	6	006	Acknowledgment	38	26	046	&	&	70	46	106	F	F	102	66	146	f	f
7	7	007	Bell	39	27	047	'	'	71	47	107	G	G	103	67	147	g	g
8	8	010	Backspace	40	28	050	((72	48	110	H	H	104	68	150	h	h
9	9	011	Horizontal Tab	41	29	051))	73	49	111	I	I	105	69	151	i	i
10	A	012	Line feed	42	2A	052	*	*	74	4A	112	J	J	106	6A	152	j	j
11	B	013	Vertical Tab	43	2B	053	+	+	75	4B	113	K	K	107	6B	153	k	k
12	C	014	Form feed	44	2C	054	,	,	76	4C	114	L	L	108	6C	154	l	l
13	D	015	Carriage return	45	2D	055	-	-	77	4D	115	M	M	109	6D	155	m	m
14	E	016	Shift Out	46	2E	056	.	.	78	4E	116	N	N	110	6E	156	n	n
15	F	017	Shift In	47	2F	057	/	/	79	4F	117	O	O	111	6F	157	o	o
16	10	020	Data Link Escape	48	30	060	0	0	80	50	120	P	P	112	70	160	p	p
17	11	021	Device Control 1	49	31	061	1	1	81	51	121	Q	Q	113	71	161	q	q
18	12	022	Device Control 2	50	32	062	2	2	82	52	122	R	R	114	72	162	r	r
19	13	023	Device Control 3	51	33	063	3	3	83	53	123	S	S	115	73	163	s	s
20	14	024	Device Control 4	52	34	064	4	4	84	54	124	T	T	116	74	164	t	t
21	15	025	Negative Ack.	53	35	065	5	5	85	55	125	U	U	117	75	165	u	u
22	16	026	Synchronous idle	54	36	066	6	6	86	56	126	V	V	118	76	166	v	v
23	17	027	End of Trans. Block	55	37	067	7	7	87	57	127	W	W	119	77	167	w	w
24	18	030	Cancel	56	38	070	8	8	88	58	130	X	X	120	78	170	x	x
25	19	031	End of Medium	57	39	071	9	9	89	59	131	Y	Y	121	79	171	y	y
26	1A	032	Substitute	58	3A	072	:	:	90	5A	132	Z	Z	122	7A	172	z	z
27	1B	033	Escape	59	3B	073	;	;	91	5B	133	[[123	7B	173	{	{
28	1C	034	File Separator	60	3C	074	<	<	92	5C	134	\	\	124	7C	174	|	
29	1D	035	Group Separator	61	3D	075	=	=	93	5D	135]]	125	7D	175	}	}
30	1E	036	Record Separator	62	3E	076	>	>	94	5E	136	^	^	126	7E	176	~	~
31	1F	037	Unit Separator	63	3F	077	?	?	95	5F	137	_	_	127	7F	177		Del

asciicharstable.com

3. UNIDADE CENTRAL DE PROCESSAMENTO - CPU

Objetivo

Examinar a estrutura e o funcionamento interno da CPU, abordar questões relativas à arquitetura da CPU e tratar aspectos de organização da CPU.

Introdução

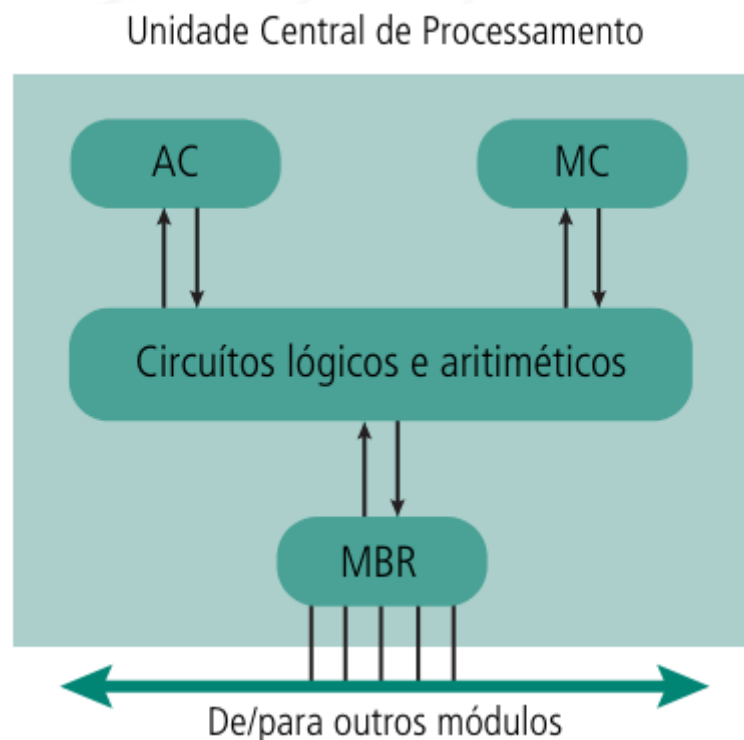
O processador ou Unidade Central de Processamento é o responsável pelas atividades de processamento do computador. Conforme cita Monteiro (2007), os processadores atuais são fabricados de modo que, num único invólucro (pastilha – chip) são inseridos todos os elementos necessários à realização de suas funções.

3.1. Unidade Lógica e Aritmética (ULA)

Assim, como todos os demais elementos de um computador, a ULA é baseada em dispositivos lógicos digitais simples, capazes de armazenar números binários e efetuar operações aritméticas e lógicas.



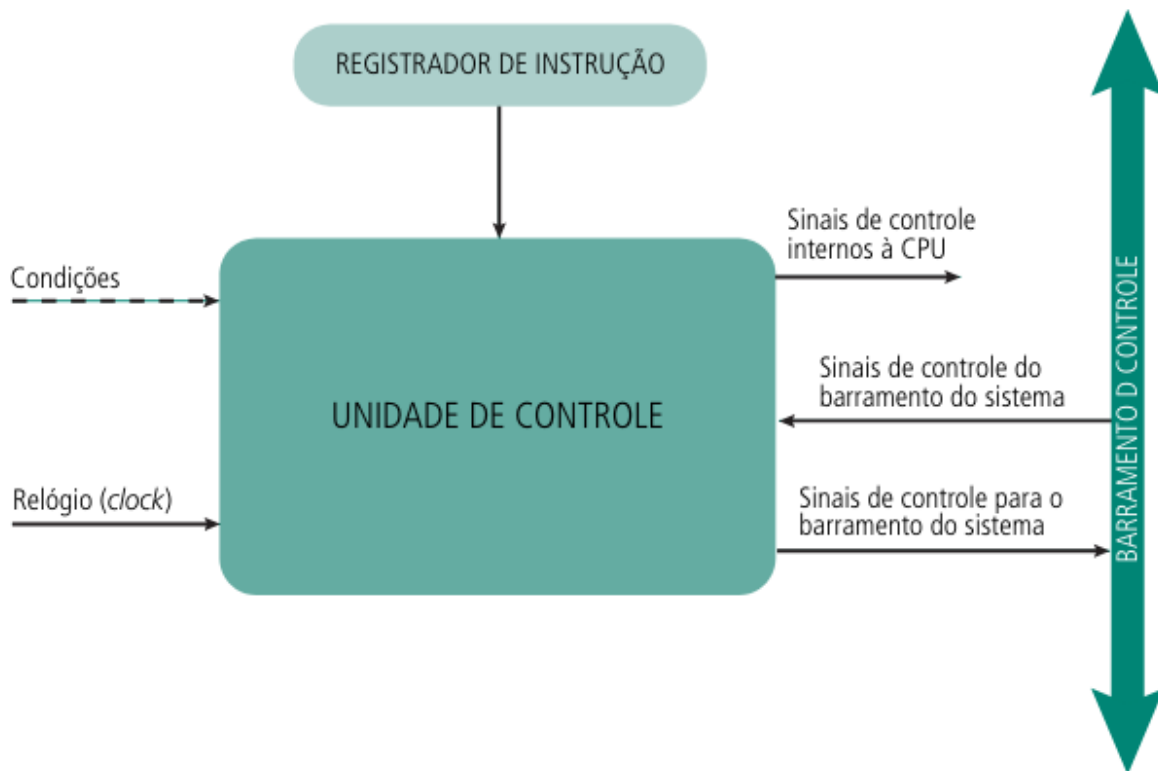
Na figura acima, podemos observar a interação e o funcionamento com o restante do sistema computacional. Os dados são fornecidos à ULA através dos registradores. Os registradores são áreas de armazenamento temporário dentro da CPU, que se conectam a ULA através de caminhos de sinal. Segundo Stallings (2002), a ULA pode também ativar bits especiais (flags) para indicar o resultado de uma operação. Por exemplo, caso o resultado de uma operação exceda a capacidade de armazenamento de um registrador, isso é indicado atribuindo o valor 1 ao bit de overflow. A unidade de controle (UC) controla a transferência de dados entre a ULA e os registradores.



Na figura acima, podemos observar que já conhecemos algumas estruturas, como o MBR (Memory Buffer Register – Registrador Temporário de Dados), que contém uma palavra com dados a ser armazenada na memória ou é utilizado para receber uma palavra da memória. Já o AC (Accumulator – Acumulador) e o MQ (Multiplier Quotient – Quociente de Multiplicação) são registradores utilizados para armazenar temporariamente os operandos e os resultados das operações efetuadas na ULA.

3.2. Unidade de Controle (UC)

A unidade de controle é a parte do processador que controla a execução de instruções. Ela gera sinais de controle externos ao processador para comandar a transferência de dados entre o processador e a memória ou os módulos de E/S. Ela também gera sinais de controle internos ao processador para mover dados entre registradores, para comandar a ULA na execução de uma determinada função e para controlar outras operações internas. As entradas para a unidade de controle consistem do registrador de instrução, bits de condição e sinais de controle gerados por fontes externas, por exemplo, sinais de interrupções, conforme Stallings (2002).



Observando a figura acima, podemos tratar da interação da unidade de controle e os demais elementos do processador na tabela a seguir.

SINAL	TIPO	DESCRIÇÃO
Relógio (<i>clock</i>)	Entrada	É uma espécie de marcador de tempo. Ele cria a sincronia nas operações do processador (ciclos). Assim como um relógio mecânico faz todos seus movimentos baseados nos segundos, o processador faz seu trabalho baseado em ciclos.
Registrador de instrução	Entrada	Contém o código da instrução corrente usada para indicar quais as operações o processador deve executar.
Condições	Entrada	Determina o estado do processador e as saídas de operações executadas pela ULA.
Sinais de controle do barramento de controle	Entrada	Para fornecer sinais de reconhecimento e interrupção.
Sinais de controle internos à CPU	Saída	Sinais para causar movimentação de dados de um registrador ao outro e para ativar funções específicas da ULA.
Sinais de controle para o barramento de controle	Saída	Indica sinais de controle para a memória e sinais de controle para os módulos de E/S.

3.3. Registradores

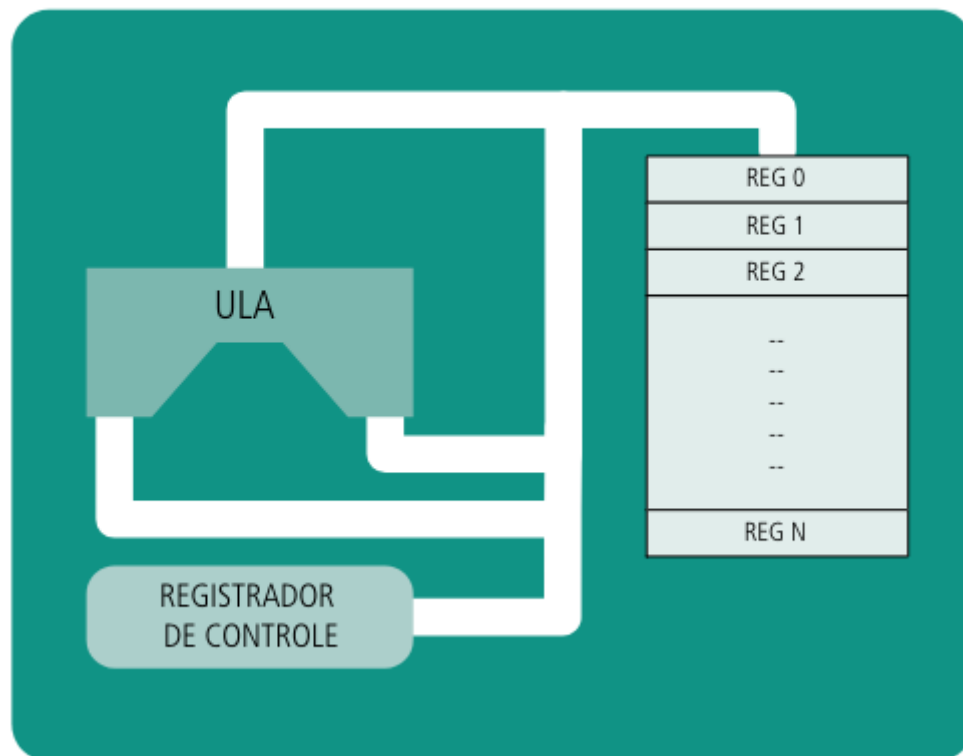
Um sistema de computação possui uma hierarquia de memória. Nos níveis mais altos dessa hierarquia, a memória é mais rápida, mais cara e menor. Os registradores estão no topo da hierarquia e como já sabemos, está localizado dentro da CPU. Os registradores podem ser:

- Registradores visíveis para o usuário: permite ao programador fazer acesso.
- Registradores de controle e de estado: usados pela UC para controlar a operação da CPU e por programas privilegiados do sistema operacional.

Um tipo importante de registradores visíveis ao usuário é o registrador de dados. Como o nome indica, ele não pode conter valores empregados no cálculo de endereços de operandos, apenas de dados.

Para que um dado possa ser transferido para ULA é necessário que ele permaneça armazenado em um registrador. Além disso, o resultado de uma operação feita na ULA precisa ser armazenado, de modo que possa ser reutilizada ou transferida para a memória

PROCESSADOR



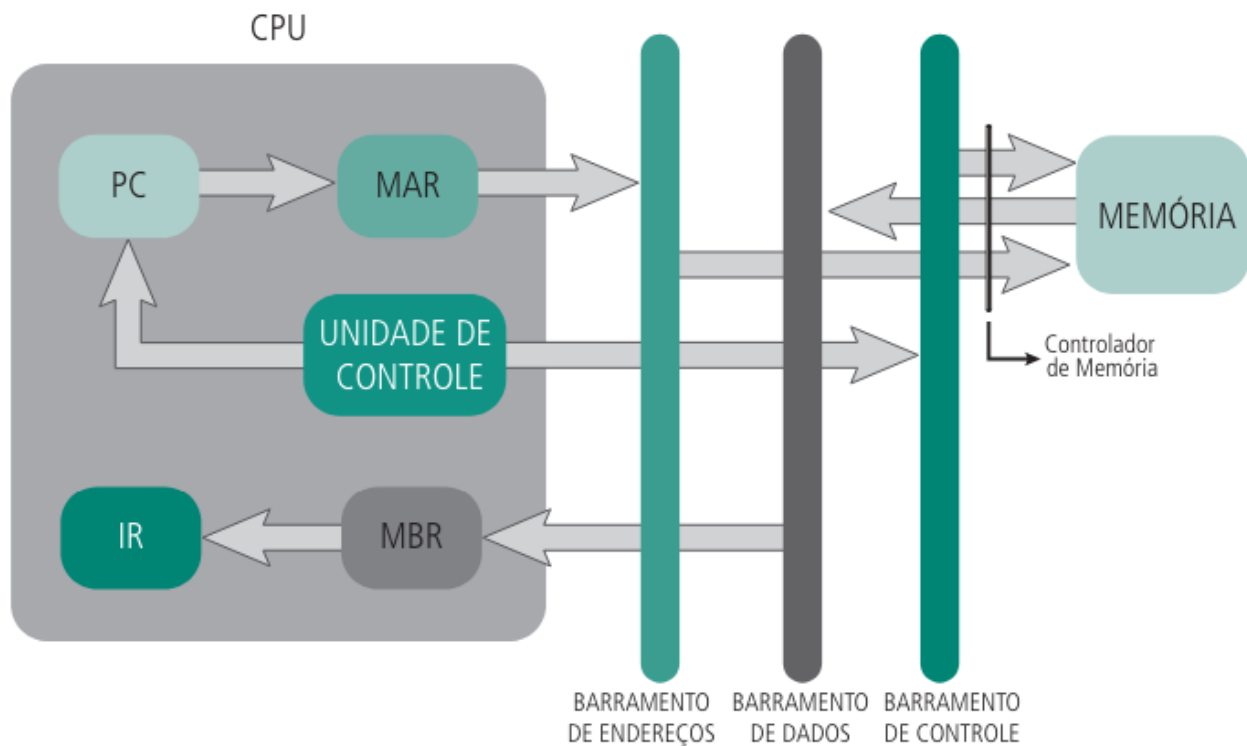
Na figura acima, os registradores de dados estão representados por REG 0 até REG N. Já os registradores de controle e estado são os vários registradores da CPU que são empregados para controlar a operação da CPU.

Quatro destes registradores são essenciais para a execução de instruções:

- a) contador de programa (PC) - contém o endereço da instrução a ser buscada;
- b) registrador de instrução (IR) - contém a última instrução buscada;
- c) registrador de endereçamento à memória (MAR) - contém o endereço de uma posição de memória;
- d) registrador de armazenamento temporário de dados (MBR) - contém uma palavra de dados a ser escrita na memória ou a palavra lida mais recentemente.

O contador de programa (PC) é atualizado pela CPU depois de cada busca de instrução, de modo que ele sempre indique a próxima instrução a ser executada e instrução buscada é carregada no registrador de instrução (IR).

A troca de dados com a memória é feita usando o MAR e o MBR. O MAR está conectado diretamente ao barramento de endereços e o MBR, ao barramento de dados.



Observe a figura acima, o contador de programa (PC) contém o endereço da próxima instrução a ser buscada. Esse endereço é movido para o MAR e colocado no barramento de endereço. A UC requisita uma leitura na memória, e o resultado é colocado no barramento de dados e copiado no MBR e então movido para o IR. Enquanto esse processo acontece, o PC é incrementado em 1 para a próxima busca.

4. MEMÓRIAS

Objetivo

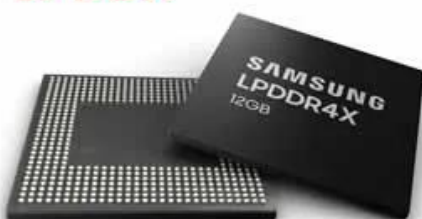
Abordar as características fundamentais dos sistemas de memórias dos computadores, examinar a hierarquia das memórias internas e seus subsistemas e apresentar as variedades de tipos, tecnologias, organização, desempenho e custos.

Introdução

As memórias em um computador estão dispostas de forma hierárquica, assim como acontece nos barramentos. No topo dessa hierarquia estão os registradores do processador, que já vimos. A seguir, estão os níveis das memórias cache, variavelmente, divididos em L1, L2, L3 ou mais, dependendo da arquitetura do processador. E por fim, estão as memórias de acesso aleatório, muitas vezes chamada de memória principal. Essas são as memórias internas de um sistema de computação. Nesta aula vamos explorar suas principais características.

4.1. Memória RAM

Uma característica importante das memórias RAM (memória de acesso aleatório) é a sua volatilidade. Isso quer dizer que ela requer o fornecimento de energia constante para que os dados fiquem armazenados na sua estrutura. Então, dizemos que essa memória é utilizada para armazenamento temporário de dados.

DDR**DDR4****DDR2****DDR5****DDR3****OPTANE****LPDDR**
arropa
 computadores
 DESDE 2001

Existem duas tecnologias de memórias RAM que merecem ser citadas. São elas:

a) Memórias RAM dinâmicas ou DRAM - feitas de células que armazenam dados com carga de capacitores. A ausência de carga no capacitor indica um bit 0 e a presença de carga, bit 1. Um capacitor descarrega-se naturalmente, sendo necessário um circuito especial que faz a regeneração periódica de seu conteúdo (chamado de refresh);

b) Memórias RAM estáticas ou SRAM - os valores binários são armazenados usando flip-flops, mantendo seus dados enquanto houver fornecimento de energia. A memória principal possui alguns parâmetros, com as seguintes características:

Parâmetros	Características
Tempo de Acesso	É o tempo gasto no ciclo de memória, ou seja, o tempo gasto nas operações de escrita ou leitura, como por exemplo, 50 ns (nanossegundos).
Capacidade	Limitado pelo projeto do processador ou pelo controlador de memória (vide Figura 6.6) embutida no <i>chipset</i> . Os processadores de 32 <i>bits</i> podem endereçar até 4GB de memória e os processadores de 64 até 16EB (<i>exabytes</i>) de memória.
Volatilidade	Os semicondutores que constituem as pastilhas de memória são construídos com capacitores, que é volátil, ou seja, necessita de alimentação elétrica para manter seus valores armazenados.
Custo	As memórias dinâmicas usadas como memória principal têm custos mais baixos do que as memórias <i>cache</i> . Dessa forma, os computadores podem ser vendidos com quantidade apreciável de memória RAM.

4.2. Memória ROM

Com relação à volatilidade apresentada no texto, há uma pequena porção da memória principal não volátil chamada de ROM, onde são armazenadas pequenas instruções executadas quando o computador é ligado.

Segundo Stallings (2002), em contraste com a memória RAM, existe a memória apenas para leitura (ROM – Read Only Memory). Esta memória contém um padrão permanente de dados que não pode ser alterado. Uma ROM é fabricada como qualquer outra pastilha, cujos dados são gravados durante o processo de fabricação. Esse processo é relativamente ruim, pois qualquer erro pode fazer com que o lote gravado seja inutilizado. As memórias ROMs evoluíram no sentido de permitir alterações pelas gravações por processos especiais. Assim, surgiram os vários tipos de ROMs, que são:



a) PROM (ROM Programável) - é uma alternativa mais barata, uma vez que após o processo de fabricação, o fornecedor ou o cliente pode gravar seu conteúdo (apenas uma única vez, pois seu processo é irreversível) por um processo elétrico;

b) EPROM (ROM Programável e Apagável) - nela, os dados podem ser lidos e gravados eletricamente, mas é utilizada uma luz ultravioleta para apagar seu conteúdo e permitir uma nova escrita, contudo, para permitir a regravação do seu conteúdo, todos os dados devem ser apagados antes de iniciar uma nova gravação;

c) EEPROM (ROM Programável e Apagável Eletricamente) - essa é uma memória mais atraente por facilitar as regravações. Quaisquer dados podem ser gravados nessa memória sem que seja necessário apagar todo o seu conteúdo, já que apenas os bytes endereçados são atualizados;

d) Flash-ROM - é um tipo específico de memória EEPROM, em que são reprogramados grandes blocos de dados. Nesse tipo de memória, todo o conteúdo é apagado antes de haver uma nova reprogramação. O nome flash foi imaginado devido à elevada velocidade de apagamento dessas memórias em comparação com as antigas EPROM e EEPROM.

4.3. Memória cache

Como já vimos, uma memória principal possui característica de possuir grande capacidade, porém seu acesso é relativamente lento. Dessa forma, essa memória é combinada com uma memória cache, menor é mais rápida. Assim, na pirâmide da memória, logo abaixo dos registradores, encontram-se as memórias cache.

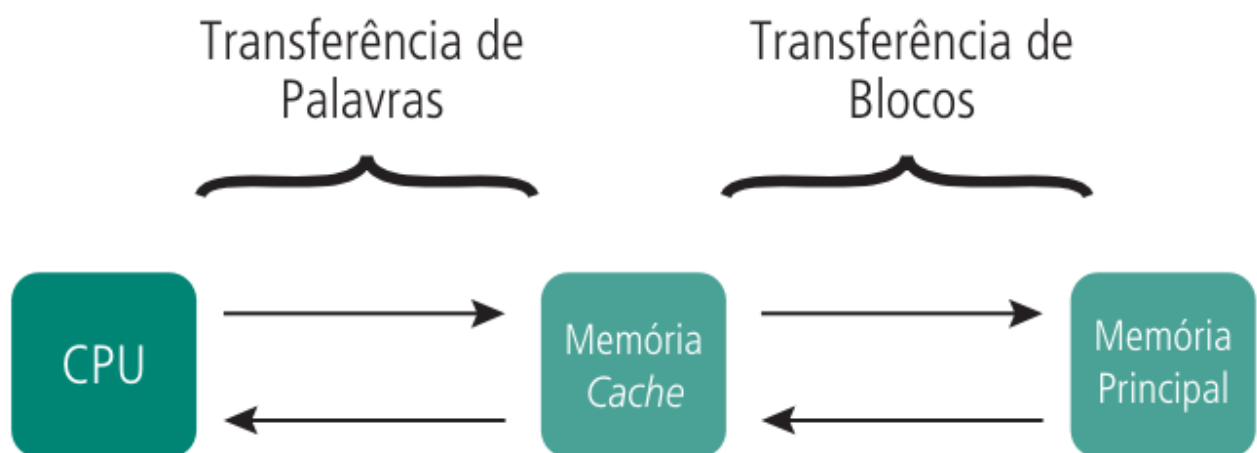
O elo entre a memória principal (RAM) e a CPU tem um ponto frágil - o ciclo de instrução é muito mais rápido do que o ciclo de memória. Durante o ciclo de refresh nenhum dado pode ser lido ou gravado na memória principal (RAM).

Na busca pela solução do problema do gargalo de congestionamento na comunicação entre o processador e a memória principal, criou-se o mecanismo de memória cache, que tem por função acelerar a velocidade de transferência entre CPU e memória principal.

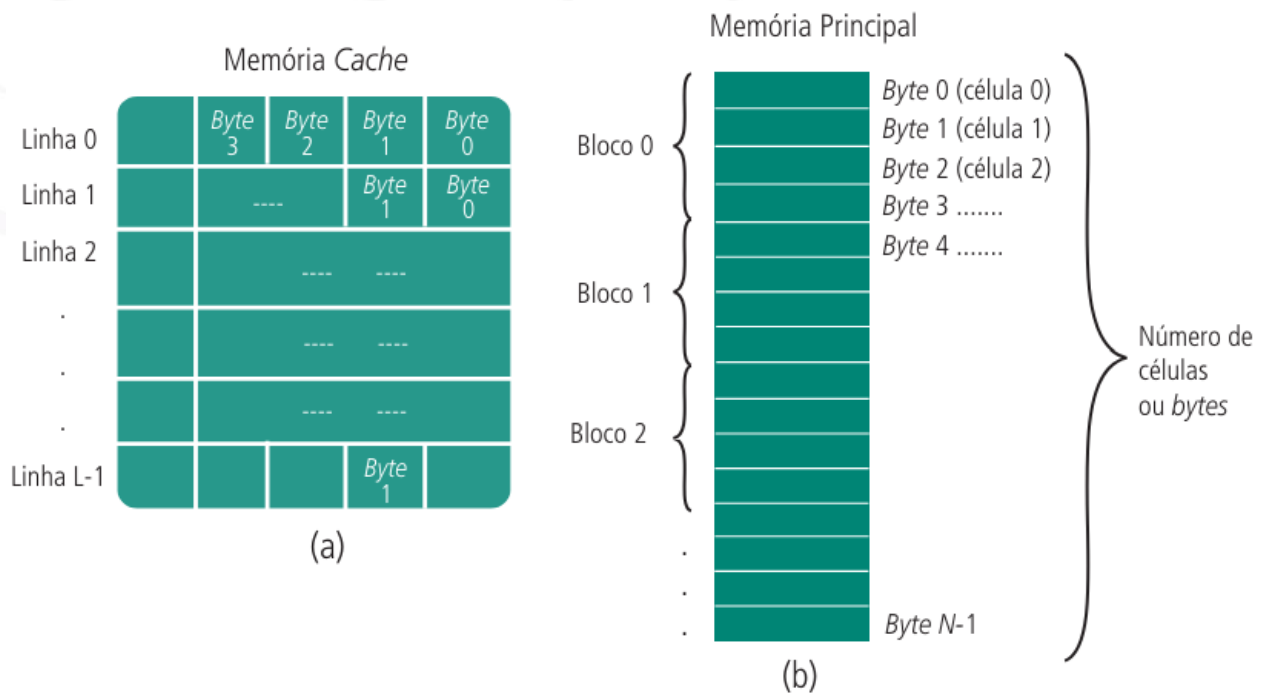
Essas memórias também possuem alguns parâmetros, conforme mostrado na tabela abaixo:

Parâmetros	Características
Tempo de Acesso	É o tempo gasto no ciclo de memória, ou seja, o tempo gasto nas operações de escrita ou leitura. No caso de <i>cache</i> , varia entre 5 e 20 ns.
Capacidade	Capacidade limitada pelo seu alto custo de fabricação e por não aumentar a eficiência do sistema na proporção em que sua capacidade.
Volatilidade	A exemplo dos registradores, as memórias <i>cache</i> são circuitos eletrônicos que precisam de energia elétrica para manter seu funcionamento e armazenamento.
Custo	Alto custo de fabricação, sendo que o valor por <i>byte</i> está situado entre o valor dos registradores e o valor da memória principal (RAM), mais barata.
Tecnologia	Em geral, são memórias estáticas, do tipo SRAM.

Quando o processador deseja ler uma palavra da memória, é realizado um teste para verificar se a palavra está na memória *cache*. Se estiver, ela é fornecida ao processador. Caso contrário, um bloco de dados da memória principal (RAM) é lida para a memória *cache*.



Para que a memória *cache* funcione no sistema CPU/CACHE/MP é necessário que o controlador de *cache* (assim como o controlador de memória) considere a MP (Memória Principal) e X células ou X bytes.

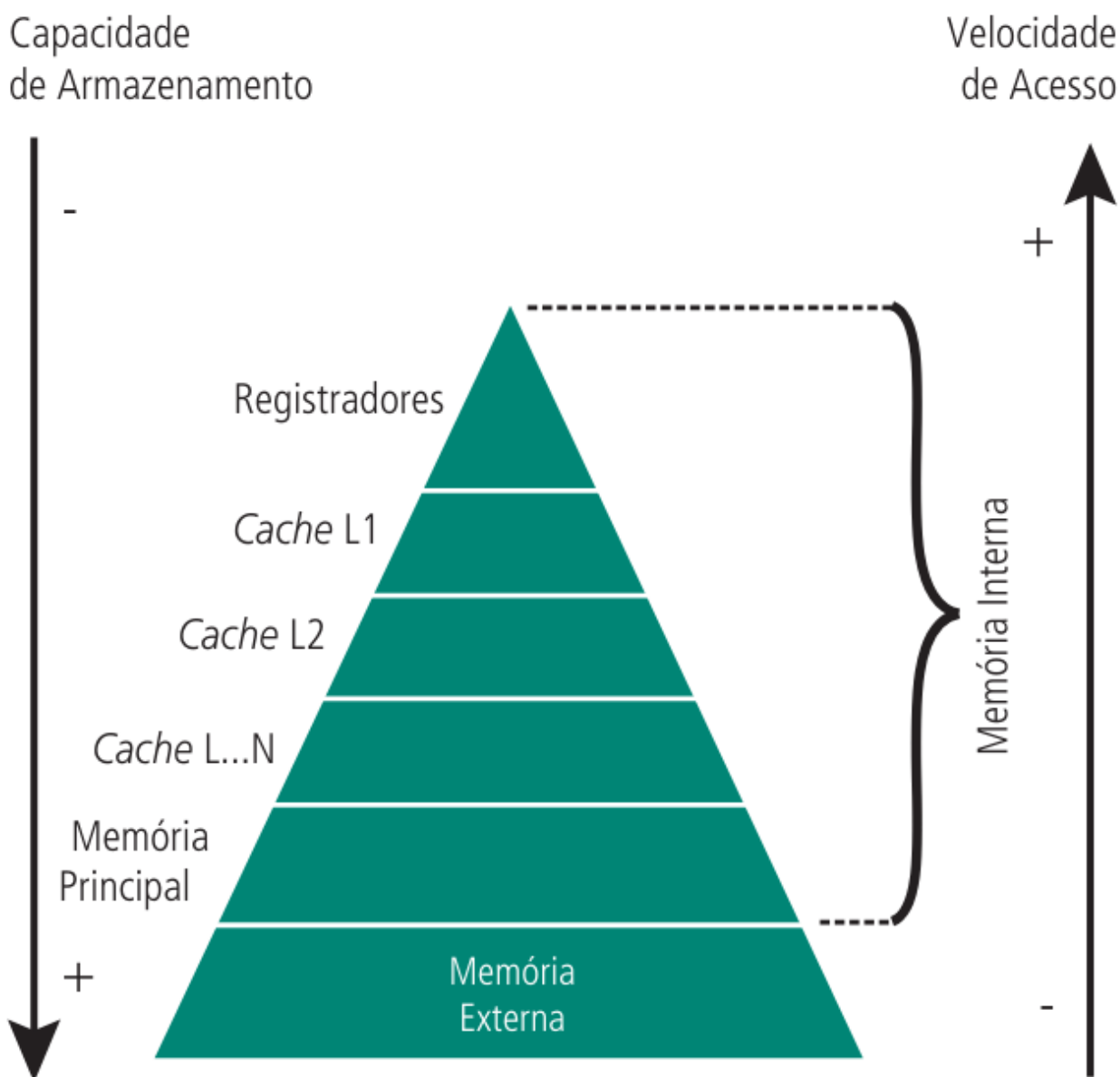


Quando há uma solicitação de transferência de dados pelo processador e o controlador de memória não o encontra na memória cache, ocorre a transferência do dado da memória principal para a cache. Entretanto, não apenas o byte requerido é transferido, mas todos os bytes subsequentes. Dessa forma, pressupõe-se que o processador precisará de outro byte do mesmo bloco, que foi, portanto, carregado para a memória cache.

Na figura acima, cada bloco possui quatro bytes de largura, que deve ser a mesma largura de uma linha de bytes da cache. A primeira coluna da cache (figura anterior) é para identificar o número do bloco da memória principal em questão, e é por vezes denominada de Tag.

4.4. Comparação dos parâmetros das memórias

Nos tópicos anteriores, foi apresentado o estudo sobre os diferentes tipos de memória interna que formam o sistema computacional. Façamos agora um breve comparativo entre suas tecnologias, no que se refere ao tamanho e desempenho.



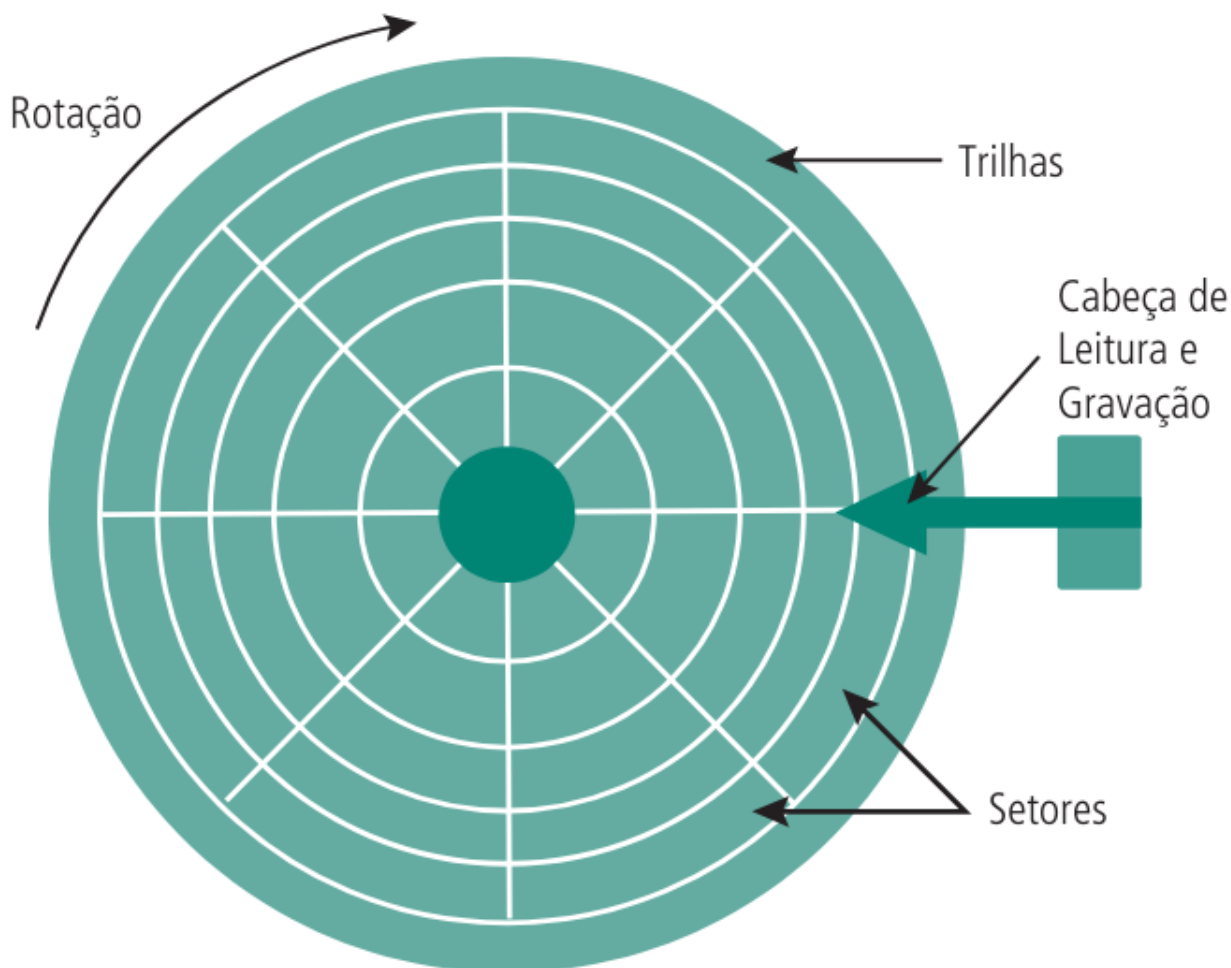
Observando a figura acima, podemos concluir o seguinte:

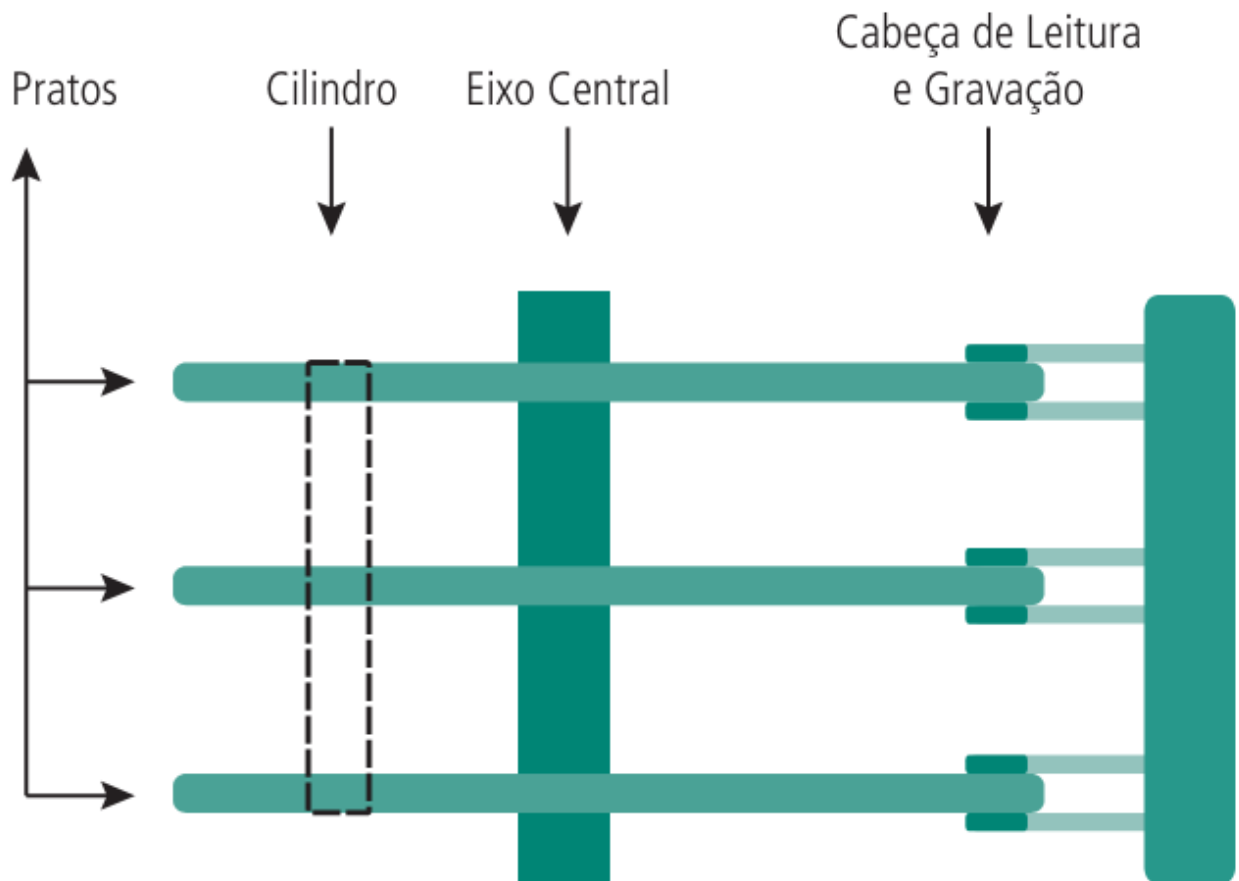
- a) os registradores são as memórias mais rápidas, porém de menor capacidade;
- b) as memórias cache são mais rápidas do que a memória principal, porém menos rápidas do que os registradores. Possuem também alta velocidade de acesso e baixa capacidade de armazenamento nos níveis mais altos da hierarquia;
- c) na organização da memória interna, a memória principal tem a maior capacidade de armazenamento e a menor velocidade de acesso;

d) a memória externa (representada pelos discos magnéticos e ópticos (HDs)) tem alta capacidade de armazenamento e baixa velocidade de acesso.

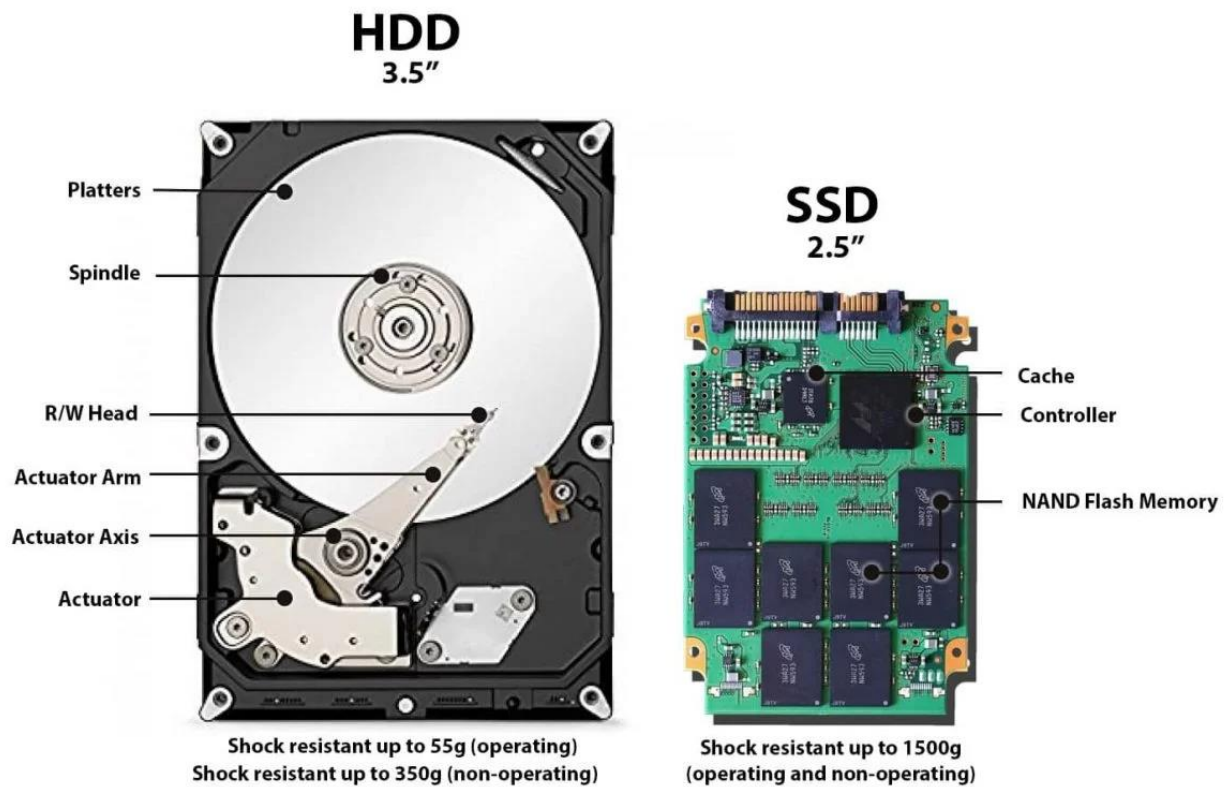
4.5. Memórias externas - HDs e SSDs

As HDDs são dispositivos de armazenamento tradicionais com pratos giratórios que leem e gravam dados. As SSDs usam tecnologia mais recente para armazenar dados em chips de memória de acesso instantâneo. As SSDs são mais rápidas, mais silenciosas, menores, mais duráveis e consomem menos energia, e as HDDs são mais baratos e oferecem mais capacidade de armazenamento e recuperação de dados mais fácil em caso de danos.





SSDs e HDDs são dispositivos de armazenamento, mas a forma como funcionam é muito diferente. A principal diferença entre uma unidade de estado sólido (SSD) e uma unidade de disco rígido (HDD) é como elas armazenam e acessam os dados. Os HDDs usam discos giratórios mecânicos e um cabeçote móvel de leitura/gravação para acessar os dados, enquanto os SSDs usam chips de memória.



4.5.1. Prós e contras de HDDs e SSDs

A inclusão de uma SSD em vez de um disco rígido em seu novo desktop ou laptop depende principalmente do seu orçamento. As SSDs geralmente são mais caras, mas podem ajudá-lo a acessar arquivos grandes várias vezes. A principal vantagem das HDDs é que elas são mais baratas, mas funcionam melhor para usuários moderados de computador.

Aqui está uma comparação prática das vantagens de uma HDD em relação a uma SSD:



faster	✓	✗	slower
more expensive	✗	✓	cheaper
non-mechanical (flash)	✓	✗	mechanical (moving parts)
shock-resistant	✓	✗	fragile
best for storing operating systems, gaming apps, and frequently used files			best for storing extra data, such as movies, photos, and documents

5. BARRAMENTOS

Objetivo

Compreender o modelo de interconexão dos principais componentes, conhecer as características básicas dos barramentos e compreender as estruturas básicas dos barramentos.

Introdução

Um computador consiste num conjunto de módulo de três tipos básicos (processador, memória e E/S), que se comunica entre si. De fato, para que isso ocorra de maneira efetiva, devem existir caminhos de conexão entre esses módulos, no caso, os barramentos. Segundo Stallings (2002), a coleção de caminhos que conectam os vários módulos é chamada de estrutura de interconexão.

5.1. Estruturas de interconexão

O modelo da estrutura de interconexão depende das informações que são trocadas entre os vários módulos de processador, memória e E/S. Observe na figura abaixo os principais tipos de troca de informações necessárias, indicando as principais formas de entrada e saída para cada tipo de módulo.



Na figura acima, a memória está tipicamente organizada em vários endereços numéricos, na qual cada endereço armazena o que chamamos de palavra, que pode ser lida ou escrita na memória. A posição de memória na qual deve ser efetuada a ação (leitura ou escrita) é especificada por um endereço. A natureza da operação (operação de leitura ou escrita) é definida por meio de sinais de controle.



Na figura acima, podemos verificar que, assim como na memória, dois tipos de operações podem ser feitas no módulo de E/S: leitura e escrita. Este módulo pode controlar vários dispositivos externos, e possuem caminhos externos para entrada (leitura) e saída (escrita).

Conforme já foi falado, o módulo de E/S também deve ser capaz de enviar sinais de interrupções ao processador. Isto é especialmente importante quando um dispositivo de E/S precisa enviar algum dado ou estabelecer comunicação com algum outro dispositivo.

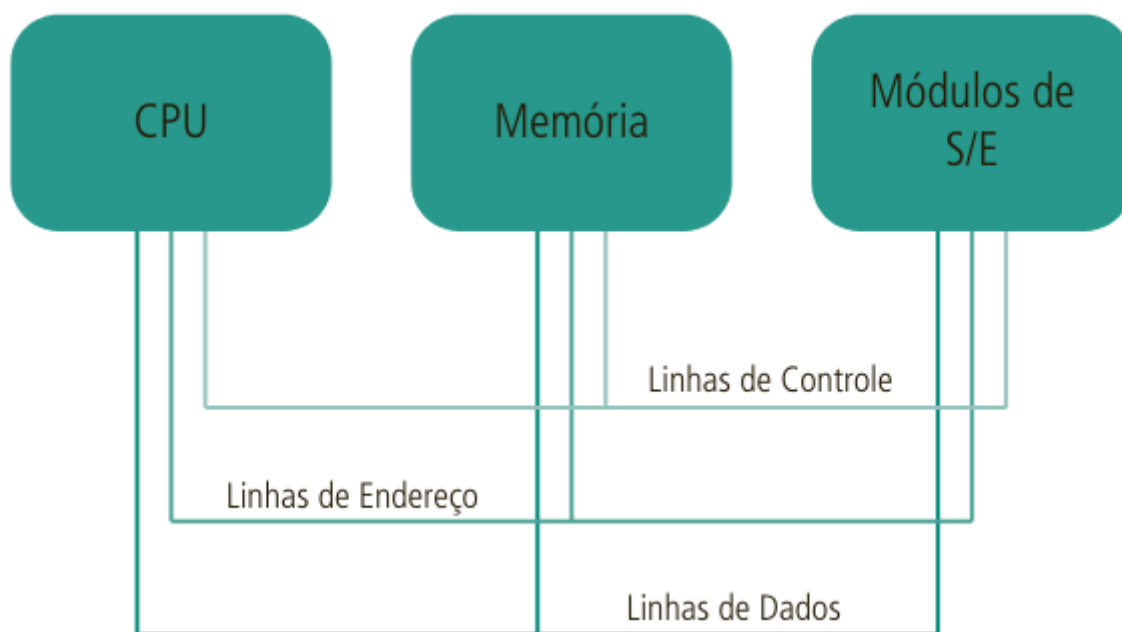


Na figura acima, podemos observar a mesma estrutura de interconexão presente na CPU. Tipicamente, os barramentos de dados, endereços e controle aparecem com partes fundamentais.

Um sistema de computação típico contém diversos barramentos, fornecendo caminhos de comunicação entre os seus componentes. O barramento usado para conectar os principais componentes do computador (processador, memória, E/S) é chamado de barramento do sistema.

5.2. Estruturas de barramentos

Um barramento do sistema contém várias linhas distintas (normalmente entre 50 e 100). De forma geral, as linhas de um barramento podem ser classificadas em três grupos: linhas de dados, linhas de endereço e linhas de controle.



Na figura acima, as linhas de dados fornecem um caminho de transferência entre os módulos do sistema (CPU, Memória e E/S). Esse conjunto de linhas é chamado de barramento de dados. Esse barramento contém, usualmente, 8, 16 ou 32 linhas (o número de linhas é conhecido como a largura do barramento). Cada linha pode conduzir apenas 1 bit por vez.

A largura do barramento de dados constitui um parâmetro fundamental para o desempenho do sistema. Por exemplo, se o barramento de dados tem largura de 16 bits e cada instrução tem tamanho de 32 bits, o processador precisa acessar duas vezes o módulo de memória em cada ciclo de instrução.

As linhas de endereços são utilizadas para designar a fonte ou o destino dos dados transferidos pelo barramento de dados. Por exemplo, quando o processador precisa ler um dado na memória (palavra de 8, 16 ou 32 bits), ele coloca o endereço da palavra desejada nas linhas de endereço. Ao conjunto dessas linhas damos o nome de barramento de endereço.

Por fim, as linhas de controle são usadas para controlar o acesso e utilização das linhas de endereço e de dados. Isto é necessário, pois as linhas de dados e endereço são compartilhadas por todos os componentes. Ao conjunto de linhas de controle damos o nome de barramento de controle.

Os sinais de controle, que viajam pelas linhas de controle, controlam, dentre outras coisas, os sinais de comando, que especificam as operações a serem executadas (como por exemplo, escrita, leitura, requisições e concessões)

Vamos exemplificar a operação de um barramento:

a) Quando um módulo do sistema deseja enviar dados para outro, ele deve:

- Obter o controle do barramento;
- Transferir os dados por meio do barramento.

b) Quando um módulo deseja requisitar dados de outro módulo, ele deve:

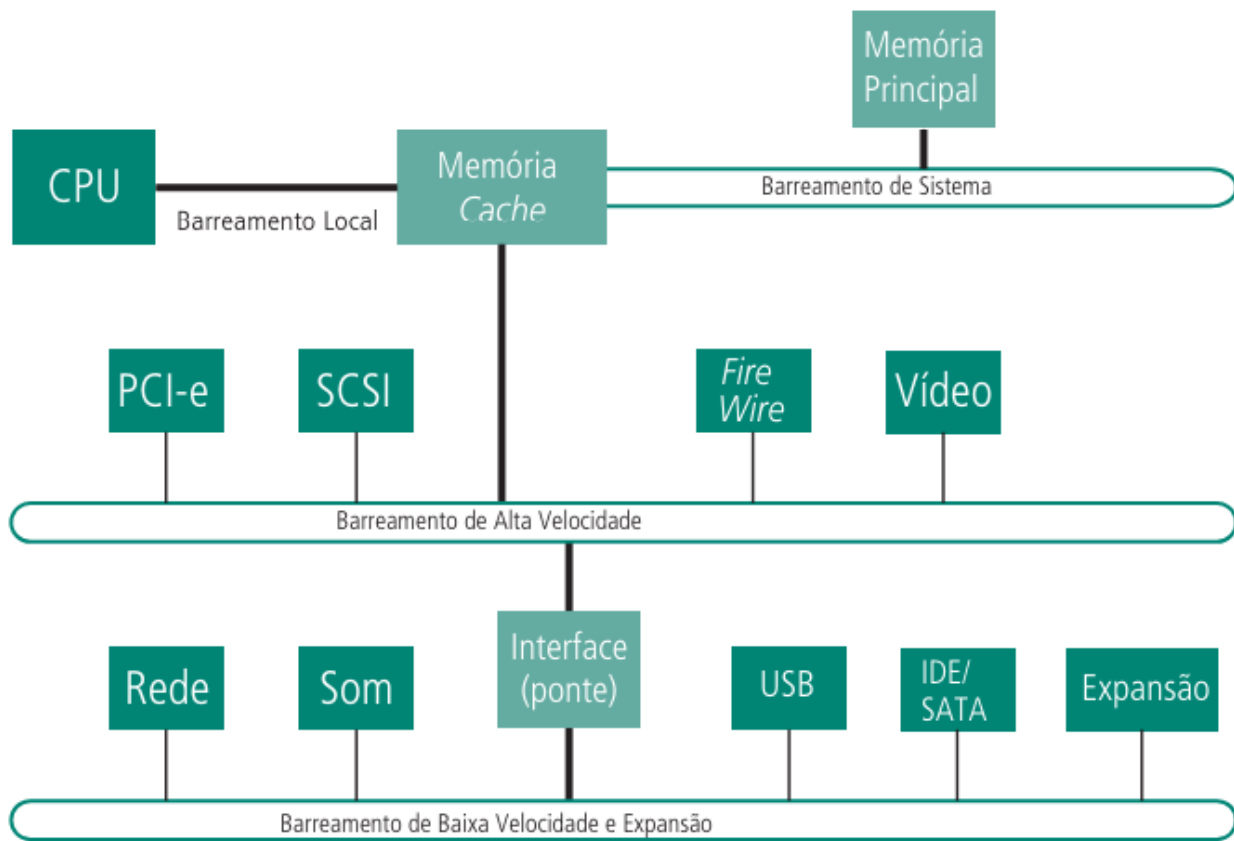
- Obter o controle do barramento;
- Transferir uma requisição para outro módulo por meio das linhas de endereços e de controle apropriadas e esperar que o módulo envie os dados requisitados.

No computador (em forma de gabinetes, placas, etc.), o barramento é formado por um conjunto de fios interconectados. Na prática, esses fios são constituídos pelas trilhas que estão presentes na placa mãe. É muito comum que a estrutura de interligação dos componentes principais do sistema (CPU, Memória, E/S) seja feita de forma modular. Assim, podemos imaginar as estruturas de barramentos presentes na placa mãe como uma haste de metal onde podemos encaixar o processador, a memória e os dispositivos de E/S.

5.3 Múltiplos barramentos

Devido à grande quantidade de dispositivos conectados ao sistema (computador), o uso de um único barramento pode prejudicar o desempenho de dispositivos. Por exemplo, dispositivos lentos “atrasam” o funcionamento dos dispositivos rápidos. Assim, não seria interessante manter, no mesmo barramento, a memória e os dispositivos USB, por exemplo.

Desse modo, grande parte dos sistemas computacionais utiliza mais de um barramento, organizados de forma hierárquica.



Na figura acima temos uma arquitetura de barramentos visando o desempenho.

Nela, podemos observar:

- barramento local - de grande velocidade e conecta o processador (CPU) ao controle da memória cache, que por sua vez se conecta ao barramento do sistema;
- barramento de alta velocidade - utilizados para interligar dispositivos de alta velocidade, como as interfaces (placas) de vídeo, os dispositivos SCSI e barramentos de expansão de alta velocidade, como o PCI-e (PCI-Express);
- barramento de baixa velocidade - utilizados para interligar dispositivos de baixa velocidade, como as unidades de leitura e gravação (disquetes, discos rígidos, unidades de CD/DVD, placas de som, placas de rede e os dispositivos USB). Nesse barramento estão as possíveis expansões, em forma do slot PCI, conforme citaremos adiante.

5.3.1 Barramento PCI

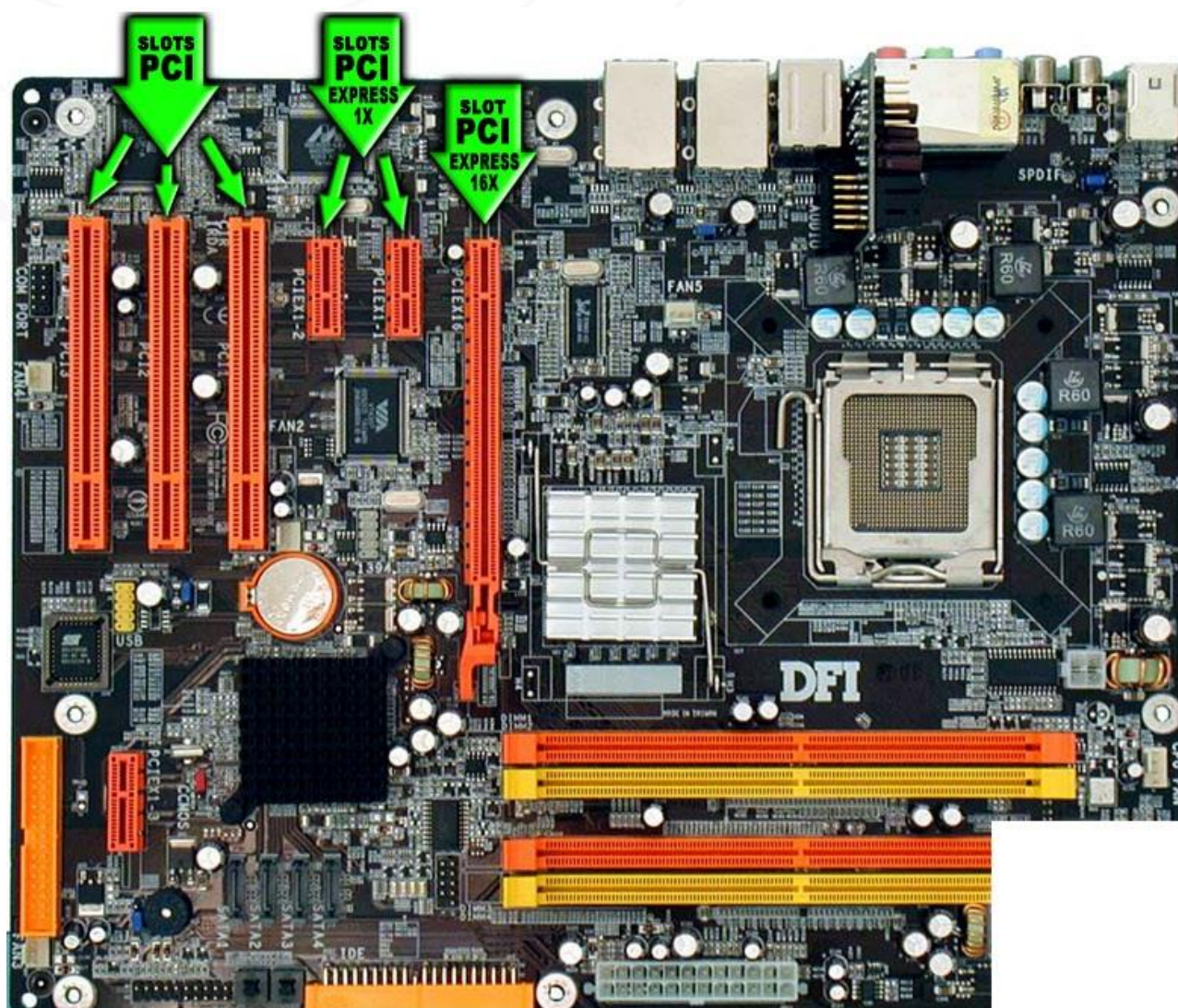
O barramento PCI (Peripheral Component Interconnect – interconexão de componentes periféricos) tem largura de banda suficiente para agregar diversos dispositivos. O PCI entrou no lugar de antigos barramentos, como o MCA, ISA, EISA, VESA (que não discutiremos aqui). No momento em que este texto está sendo escrito, o barramento PCI está lentamente cedendo lugar aos barramentos PCI-e. O PCI pode contar com 32 ou 64 linhas de dados (32 ou 64 bits) funcionando a uma taxa bruta máxima de 528 MB/s (MegaBytes por segundo). Esse barramento é tipicamente compartilhado com todos os outros dispositivos que o utiliza, causando uma queda generalizada de desempenho. Outro fator negativo é uso de transmissões paralelas, o que limita fisicamente as evoluções deste barramento, já que necessitariam de mais vias de dados para aumentar as taxas, o que nem sempre é possível.

5.3.2 Barramento PCI - Express

Também denominado PCI-e, esse é um barramento que utiliza transmissão serial em seus circuitos, diminuindo possíveis interferências entre vias de dados. Como é um barramento de alta velocidade, está substituindo os barramentos PCI e o barramento AGP.

Uma das características fundamentais do PCI Express é que ele é um barramento ponto a ponto, em que cada periférico possui um canal exclusivo de comunicação com o chipset. No PCI tradicional, o barramento é compartilhado por todos os periféricos ligados a ele, o que pode criar gargalos, de acordo com Morimoto (2007).

Ainda segundo Morimoto (2007), existem quatro tipos de slots PCI-Express, que vão do 1x ao 16x. O número indica quantas linhas de dados são utilizadas pelo slot e, conseqüentemente, a banda disponível. Cada linha PCI-Express utiliza quatro pinos de dados (dois para enviar e dois para receber), que são capazes de transmitir a 250 MB/s em ambas as direções.



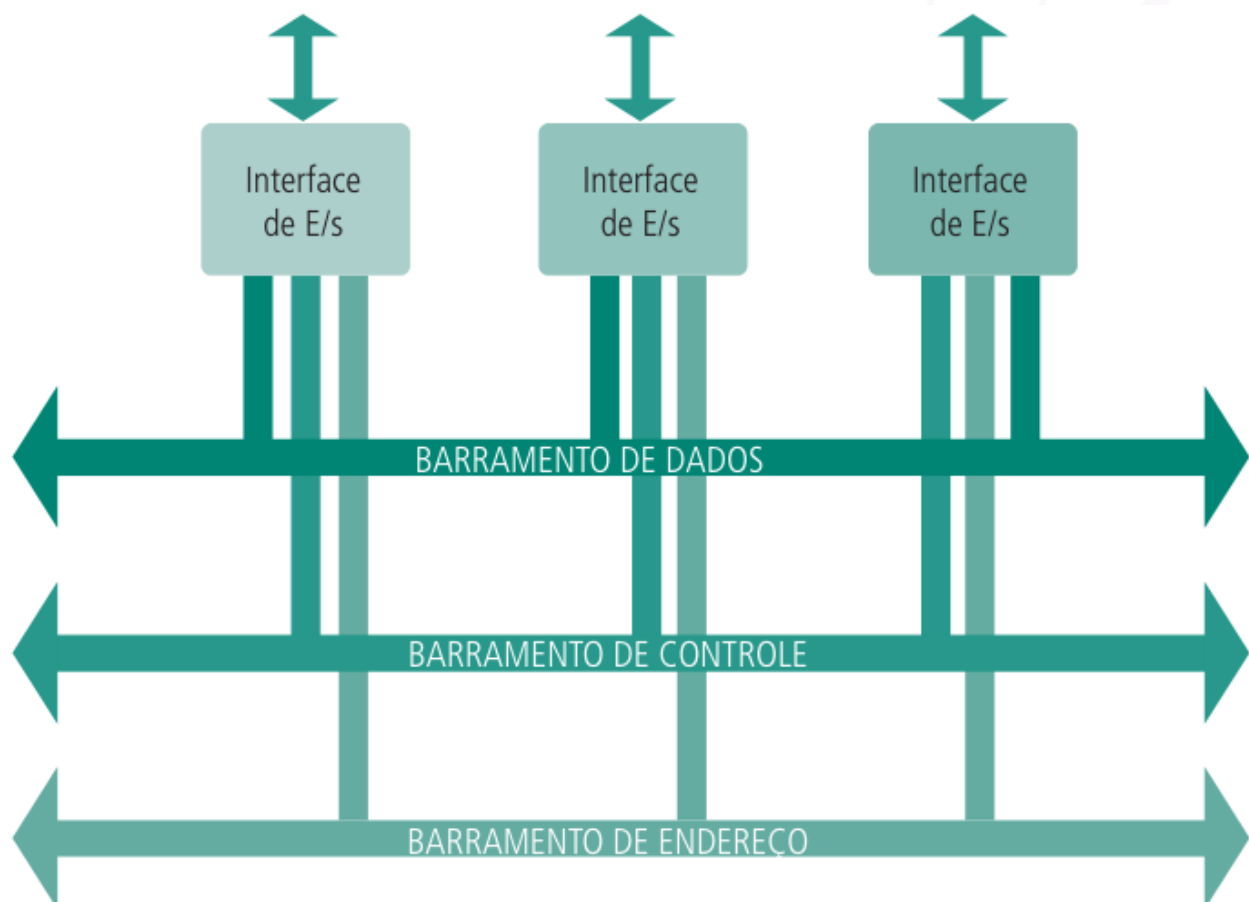
6. ENTRADA E SAÍDA (E/S)

Objetivo

Conhecer as interfaces de E/S de contato com o mundo externo, abordar as principais técnicas de E/S e apresentar as principais interfaces de E/S.

Introdução

Os módulos de entrada/saída são fundamentais na composição do sistema. Cada módulo se conecta ao barramento do sistema, podendo controlar um ou mais dispositivos periféricos. O módulo de E/S vai além de ser apenas um conjunto de conectores e cabos que ligam o dispositivo ao barramento, ele possui lógica associada para exercer o controle sobre os dispositivos periféricos.



6.1. Dispositivos externos

Os dispositivos externos oferecem um meio de troca de dados entre o ambiente externo e o computador. A conexão, como já sabemos, é feita através de um módulo de E/S.

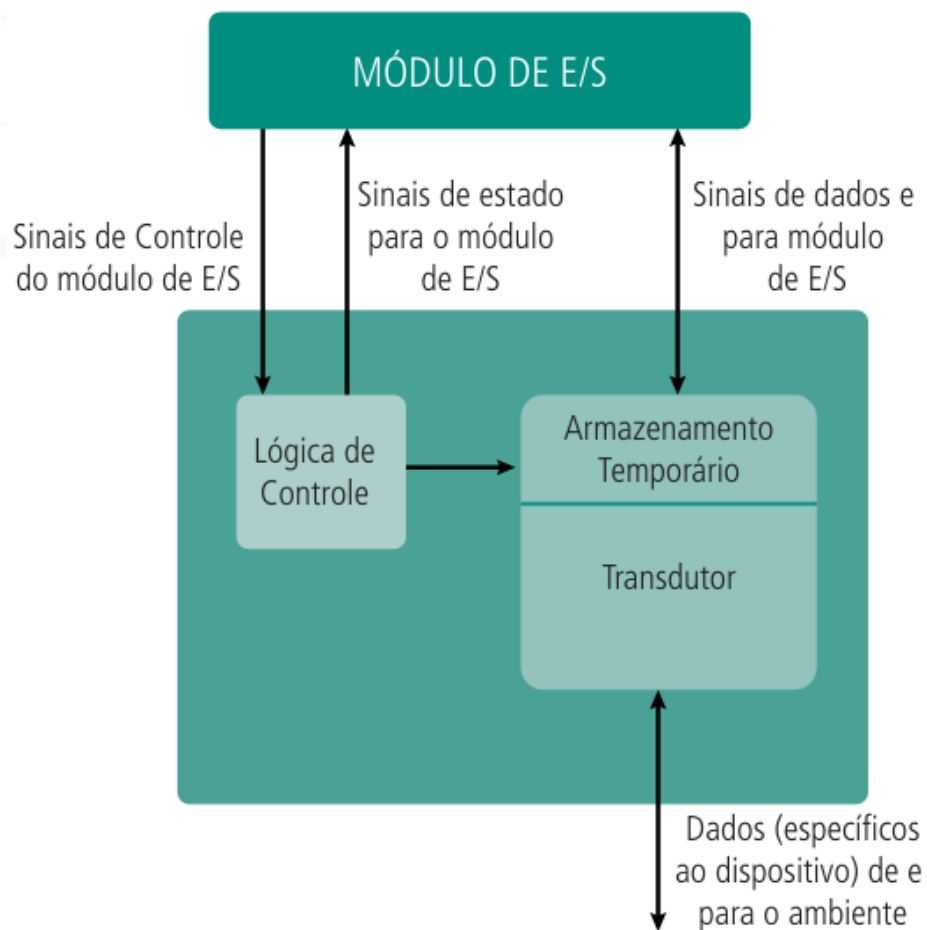
Os dispositivos externos podem ser classificados em três categorias:

a) voltados para a comunicação com o usuário - podemos citar aqui os monitores, terminais de vídeo, impressoras, teclados, mouse, dentre outros;

b) voltados para a comunicação com a máquina - discos magnéticos (discos rígidos) e discos ópticos são exemplos. É importante lembrar que, nesse momento, o ponto de vista das unidades é estrutural, ou seja, a forma como eles são controlados pelos módulos de E/S. Do ponto de vista de hierarquia, eles já foram apresentados como memória externa;

c) voltados para a comunicação com dispositivos remotos - redes de computadores usando interfaces de rede ou modem podem servir de exemplo.





Na figura acima, podemos descrever detalhes da interação do dispositivo externo com o módulo de E/S. A interface com o módulo de E/S é formada de sinais de controle, estado e dados. Observe:

- os sinais de controle determinam a função a ser executada pelo dispositivo (como enviar dados para o módulo de E/S – realizando um INPUT – ou receber dados do módulo de E/S – realizando um OUTPUT);
- os sinais de estado indicam o status do dispositivo, como por exemplo, se ele está pronto ou não (ligado, desligado, ocupado);
- os sinais de dados formam o conjunto de bits a serem enviados para o módulo de E/S ou recebido dele.

A lógica de controle controla a operação do dispositivo, como por exemplo, o recebimento de um comando para movimentar o cabeçote de um disco para determinada

posição, essa função é dada pelo sinal de controle. O transdutor converte dados decodificados para sinais elétricos e vice-versa e geralmente usa um espaço temporário para armazenar os dados trabalhados.

6.2 Módulos e funções de E/S

As funções mais importantes de um módulo de E/S podem ser assim divididos:

- a) controle e temporização;
- b) comunicação com o processador;
- c) comunicação com dispositivos;
- d) área de armazenamento temporário de dados;
- e) detecção de erros.

Levando em consideração que o processador pode se comunicar com um ou mais dispositivos para realizar E/S, um acesso a disco, por exemplo, o módulo de E/S inclui funções de controle e temporização. Observe a tabela a seguir:

Etapa	Descrição
1	O processador interroga o módulo de E/S para verificar seu <i>status</i> .
2	O módulo de E/S retorna o <i>status</i> do dispositivo.
3	Se o dispositivo estiver em operação e pronto, o processador requisitará a transferência de dados, por um comando para o módulo de E/S.
4	O módulo de E/S recebe um bloco de dados do dispositivo externo.
5	Os dados são transferidos do módulo de E/S para o processador.

O modelo apresentado na tabela acima indica que há necessidade de interação entre o módulo de E/S e o processador. Com isso, a comunicação com o processador envolve os seguintes aspectos:

- a) decodificação de comando - o módulo de E/S recebe comandos do processador, enviados como sinais, pelo barramento de controle;
- b) dados - são os dados transferidos entre o processador e o módulo de E/S pelo barramento de dados;

c) informação de estado - os periféricos, em geral, são mais lentos do que o conjunto processador-memória. Assim, é necessário saber seu estado antes de requisitar seus recursos novamente, como por exemplo, a escrita de um arquivo no disco, sendo que o módulo de E/S ainda estava processando uma solicitação anterior;

d) reconhecimento de endereço - da mesma forma que cada byte em memória tem seu endereço, cada dispositivo está associado a um endereço de E/S exclusivo.

Por último e não menos importante, o módulo de E/S precisa realizar detecção de erros. Tais erros podem variar desde um mau funcionamento mecânico ou elétrico até a alteração no padrão de bits transmitidos. Para se detectar erros de transmissão, utiliza-se algum tipo de código de transmissão. Um exemplo clássico é o bit de paridade.

6.3 Principais técnicas de E/S

Existem, basicamente, três técnicas principais de E/S, que são:

a) E/S programada (PIO – Programmed Input Output), que é efetuada pelo programa que requisitou a operação de E/S;

b) E/S dirigida por interrupção, o programa requisita a E/S ao módulo de E/S e continua sua execução até que ocorra uma interrupção gerada por hardware;

c) Acesso direto a memória (DMA – Direct Memory Access), em que a E/S é controlada por um processador especial, encarregado por transferir blocos de dados.