

# Módulo III

## Arquitetura de Computador

Flávio Ferry de Oliveira Moreira



**PRESIDENTE DA REPÚBLICA**

Luiz Inácio Lula da Silva

**MINISTRO DA EDUCAÇÃO**

Fernando Haddad

**GOVERNADOR DO ESTADO**

Wellington Dias

**REITOR DA UNIVERSIDADE FEDERAL DO PIAUÍ**

Luiz de Sousa Santos Júnior

**SECRETÁRIO DE EDUCAÇÃO À DISTÂNCIA DO MEC**

Carlos Eduardo Bielschowsky

**COORDENADORIA GERAL DA UNIVERSIDADE ABERTA DO BRASIL**

Celso Costa

**SECRETÁRIO DE EDUCAÇÃO DO ESTADO DO PIAUÍ**

Antonio José Medeiros

**COORDENADOR GERAL DO CENTRO DE EDUCAÇÃO ABERTA À DISTÂNCIA DA UFPI**

Gildásio Guedes Fernandes

**SUPERINTENDENTE DE EDUCAÇÃO SUPERIOR NO ESTADO**

Eliane Mendonça

**DIRETOR DO CENTRO DE CIÊNCIAS DA NATUREZA**

Helder Nunes da Cunha

**COORDENADOR DO CURSO NA MODALIDADE EAD**

Luiz Cláudio Demes da Mata Sousa

**COORDENADORA DE MATERIAL DIDÁTICO DO CEAD/UFPI**

Cleidinalva Maria Barbosa Oliveira

**DIAGRAMAÇÃO**

Aguiar Neto

# Apresentação

Este texto é destinado aos estudantes do Bacharelado em Sistemas de Informação da Universidade Aberta do Piauí que estão cursando a disciplina de Arquitetura de Computadores, do Módulo III e da grade 1. O presente trabalho é composto de 7 (sete) unidades a saber:

- Unidade 01 – Introdução: aqui será feita uma apresentação geral da evolução do hardware, organização multinível das arquiteturas modernas e o padrão de Von Newmann;
- Unidade 02 – Dispositivos de entrada, saída e armazenamento: uma visão geral dos periféricos existentes hoje com seu funcionamento e suas aplicações e também as memórias permanentes, tais como: discos magnéticos, fitas e discos ópticos;
- Unidade 03 – Memória: um estudo sobre o funcionamento das memórias dos computadores atuais, passando pela RAM, ROM e memória flash;
- Unidade 04 – Barramentos e Interfaces: mostra os padrões de comunicação entre dispositivos e também entre periféricos do computador;
- Unidade 05 – Dentro da CPU: linguagem de montagem, modos de endereçamento, repertório de instruções e interrupções;
- Unidade 06 – Suporte do Sistema Operacional: manipulação de pilha e realocação de código;
- Unidade 07 – Arquiteturas RISC: apresentação da arquitetura RISC e comparações com a CISC.

Ao final desta disciplina o aluno estará capacitado a entender melhor o funcionamento e as opções de configuração de hardware. E além deste material recomenda-se a leitura de fontes que serão citadas ao longo do texto.

Bom trabalho para todos.

# Sumário

<b>UNIDADE 1 – INTRODUÇÃO</b>	
1.1 – Máquina Multinível.....	08
1.2 – Evolução dos Equipamentos .....	09
1.3 – Padrão Von Neumann .....	11
1.4 – A Evolução dos Processadores .....	12
<b>UNIDADE 2 – DISPOSITIVOS DE ENTRADA E SAÍDA</b>	
2.1 – Dispositivos de Entrada e Saída .....	18
2.2 – Dispositivo de Armazenamento .....	30
<b>UNIDADE III – MEMÓRIA</b>	
3.1 – Introdução .....	38
3.2 – Memória ROM .....	38
3.3 – Memória Flashmidsb .....	39
3.4 – Memória RAM .....	41
3.5 – Detecção e Correção de Erros .....	44
3.6 – Memória CACHE .....	45
3.7 – Hierarquia de Memórias .....	50
<b>UNIDADE IV – BARRAMENTOS E INTERFACES</b>	
4.1 – Barramentos – Conceitos Gerais .....	55
4.2 – Barramentos Comerciais .....	62
4.3 – Interfaces – Barramentos Externos .....	68
<b>UNIDADE V – DENTRO DA CPU</b>	
5.1 – Ciclo BDE .....	77
5.2 – Modos de Endereçamento .....	88
<b>UNIDADE VI - PILHA</b>	
6.1 – Pilha .....	93
6.2 – Gerenciamento de Memória .....	94
6.3 – Realocação .....	95
6.4 – Proteção .....	98
6.5 – Considerações sobre este capítulo .....	98
<b>UNIDADE VII - MOTIVAÇÃO</b>	
7.1 – Motivação .....	102
7.2 – Filosofia RISC .....	103
7.3 – Características .....	103
7.4 – Uso de microcódigo x Execução direta em hardware ....	109
7.5 – Repertório de Instruções .....	109
7.6 – Programação e computadores RISC .....	110
7.7 – Diferenças críticas entre as arquiteturas .....	113
7.8 – Comparações de desempenho entre arquiteturas .....	114
7.9 – Entretenimento com arquitetura RISC .....	116
Apêndice .....	119

# Unidade 1

## Introdução

### **Resumo**

Nesta unidade serão mostrados os conceitos fundamentais de Arquitetura de Computadores. Fazendo uma apresentação da técnica de máquinas multiníveis, mostrando a evolução dos computadores eletrônicos desde as válvulas até os circuitos integrados de larga escala, tudo isso dentro do padrão proposto por Von Newmann.

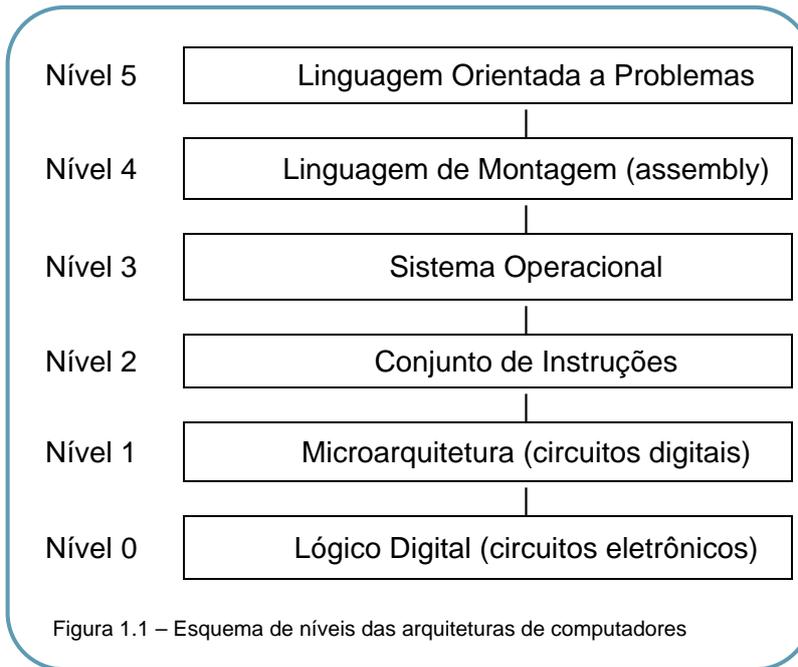
# Sumário da Unidade

## **UNIDADE 1 – INTRODUÇÃO**

1.1 – Máquina Multinível.....	08
1.2 – Evolução dos Equipamentos .....	09
1.3 – Padrão Von Newmann .....	11
1.4 – A Evolução dos Processadores .....	12

## 1.1 Máquina multinível

Para aproximar os seres humanos da máquina é usada uma divisão em camadas das arquiteturas de computadores. Assim, quanto mais camadas tiver uma arquitetura, mais próxima da linguagem humana será a linguagem de alto nível deste computador. Veja na ilustração a seguir:



Veja que tudo começa com o trabalho dos engenheiros eletrônicos e de computação, onde são determinados os componentes eletrônicos dos circuitos que vão compor as portas lógicas e demais circuitos digitais, sendo este conhecido por nível zero. Aqui se trabalha com capacitores, resistores, reguladores de potência, fusíveis, dentre outros elementos da eletrônica básica. Esses circuitos vão ser organizados na forma de “pacotes” para compor computadores, os chamados circuitos digitais. Classificados em nível um, estes circuitos são usados para compor as implementações práticas de todas as funções e mapeamentos usados na teoria dos circuitos digitais, nesse nível ainda temos o trabalho de engenheiros e projetistas de hardware. No nível dois temos, basicamente, o trabalho de projetistas de hardware, profissionais ligados à engenharia e também a área de software, pois aqui se define o conjunto de instruções que determinada CPU é capaz de reconhecer, que tipo de trabalho determinado computador é capaz de realizar, dentre outras coisas. O trabalho das equipes de soft-

ware (bacharéis em ciência da computação, sistemas de informação, dentre outros) começa a partir do nível três. Aqui é criado o Sistema Operacional do computador. Este programa é capaz de controlar todo o funcionamento do sistema, tanto em nível de software, quanto em nível de hardware, programa que todos os outros existentes no computador dependem para executar e ter acesso a recursos disponíveis no sistema. Acima desse nível tem-se a linguagem de montagem (assembly). Essa linguagem existe para fazer o programador ter acesso a funcionalidades do computador que não seriam permitidas pelas chamadas linguagens de programação de alto nível. São programações necessárias de se executar diretamente no hardware ou mais intimamente com o sistema operacional. Finalmente no nível cinco tem-se a linguagem de alto nível, patamar onde se encontram linguagens como Pascal, Delphi, Java e outras.



**ATENÇÃO:** Não confunda ASSEMBLY, que é a linguagem de programação do nível quatro, com ASSEMBLER, que é o programa utilizado para executar os códigos fontes criados em ASSEMBLY. O assembler também é conhecido por montador, exatamente por fazer a execução da linguagem de montagem.

Os computadores atuais contam com esses níveis citados, embora alguns autores já defendam que exista um sexto nível, porém isso ainda não se configura como uma realidade, visto que um sexto nível implicaria que estes equipamentos seriam tão fáceis de programar quanto um eletrodoméstico qualquer, sabendo ainda que muitos usuários não conseguem usufruir nem de metade das funcionalidades de seus eletro-eletrônicos de uso doméstico. Hoje em dia, qualquer aparelho de som funciona como despertador, poucos sabem sequer acertar o relógio do aparelho. Sem falar na subutilização dos recursos de fornos de microondas, gravadores de DVD e até dos ultrapassados videocassetes.

## 1.2 Evolução dos equipamentos

A computação eletrônica evoluiu junto com toda a eletrônica nas demais áreas de conhecimento da humanidade. Antes do surgimento dos

componentes eletrônicos os computadores eram baseados em componentes mecânicos e sujeitos às falhas de tais componentes e seus funcionamentos. Este período da computação mecânica deu-se em 1642, quando Blaise Pascal inventou a **pascalina**, até 1945 quando surgiu o primeiro computador valvulado. O detalhe interessante desse período da história da computação é que Pascal inventou uma máquina que pudesse servir de calculadora para ajudar seu pai, que era fiscal de impostos do rei da França, e Ada Lovelace, filha de Lorde Byron, soube da invenção e escreveu um procedimento para programar essa máquina, sendo assim, ela é considerada a primeira pessoa a escrever um algoritmo.



Figura 1.2 - Válvulas

A história foi evoluindo até que na Segunda Guerra Mundial já existiam alguns componentes da eletrônica dos dias de hoje, alguns pesquisadores de renome mundial empenharam-se em fazer calculadoras automáticas - nomes como Konrad Zuse, Howard Aiken, John Mauchley, dentre outros. Os relês foram usados a princípio, mas antes do final da guerra já era a válvula a responsável pelo nascimento da computação eletrônica. Essa ficou conhecida como a PRIMEIRA GERAÇÃO DE COMPUTADORES (1945-1955).

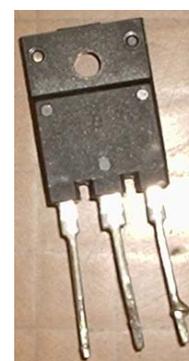


Figura 1.3 - Transistor

Na década de 50 do século XX foram inventados os transistores, que por sua vez substituíam as válvulas com mais eficiência, pois eram componentes mais ágeis, dissipavam menos calor e gastavam menos energia para fazer o mesmo trabalho que as antigas válvulas. Os computadores diminuíram de tamanho e ficaram mais eficientes nesta que ficou sendo a SEGUNDA GERAÇÃO DE COMPUTADORES (1955-1965).



Figura 1.4 – Circuito Integrado



Figura 1.5 – Pastilha VLSI

Por volta da segunda metade da década de 60 do século XX, foi inventado o circuito integrado, uma pequena pastilha (para os padrões da época) consumindo ainda menos energia e ocupando menos espaço que os equipamentos de antes, onde os transistores eram usados sem integração. Nasce a TERCEIRA GERAÇÃO DE COMPUTADORES (1965-1980). Estes circuitos

ganharam denominações diferentes em alguns lugares do planeta, aqui no Brasil é comum chamar circuito integrado de CHIP.

De 1980 até os dias de hoje, vive-se a geração mais nova da computação, chamada VLSI, ou circuitos com integração em larga escala. São Circuitos Integrados especiais que contêm milhares (ou até milhões) de transistores por centímetro quadrado de área da pastilha. Esta é a QUARTA GERAÇÃO DE COMPUTADORES. Os PCs em uso hoje pertencem todos a esta geração.

Há quem defenda a existência de uma quinta geração nos dias de hoje, fala-se no computador invisível, porém isso ainda é considerado ficção científica.

### 1.3 Padrão Von Neumann

John Von Neumann matemático húngaro, naturalizado norte-americano, propôs nos anos 40 do século XX, um padrão de arquitetura de computadores que ainda hoje é seguido, sendo hoje em dia altamente pesquisada uma alternativa a esse padrão. Ainda não se tem de fato uma idéia que seja melhor que a desse cientista.

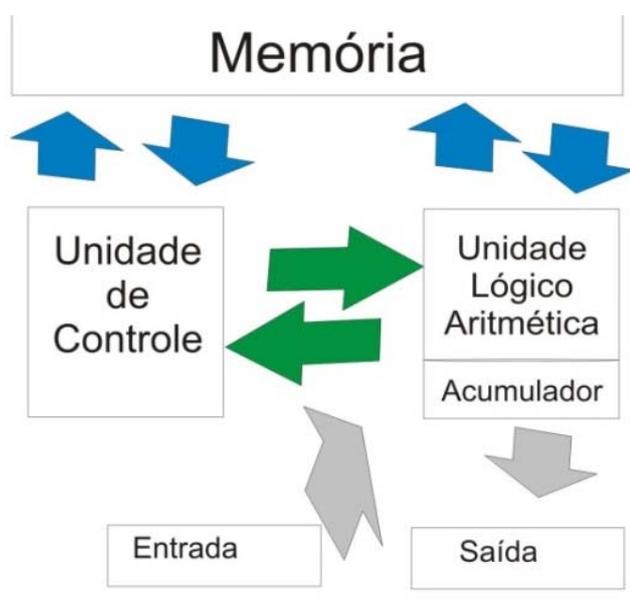


Figura 1.6 – Arquitetura Von Neumann

A máquina proposta por Von Neumann conta com os seguintes componentes: uma Memória, uma Unidade Aritmética e Lógica (ULA), uma Unidade Central de Processamento (UCP), composta por diversos Registradores, e uma Unidade de Controle (UC), cuja função é a mesma da tabela de controle da máquina de Turing universal: buscar um programa na memória, instrução por instrução, e executá-lo sobre os dados de entrada. A figura a seguir ilustra tal máquina.

Os detalhes destes componentes serão feitos nos capítulos seguintes. Mas em li-

nas gerais pode-se dizer que a **unidade de controle** “sabe” em que parte do programa está e para onde vai a execução do mesmo e o que fazer em casos de desvios. A execução das instruções juntamente com os dados de **entrada** é responsabilidade da **unidade lógico aritmética**, que por sua vez contém uma ou algumas memórias especiais denominadas de registros, caso seja apenas um registrador chama-se de **acumulador**. Os dados processados pela **ULA** podem ser armazenados diretamente na **memória principal** do sistema ou podem ser enviados para um dispositivo de **saída**.

#### 1.4 A Evolução dos Processadores

Desde a chegada da geração dos Circuitos Integrados, a Unidade Central de Processamento dos computadores passou a agregar outros componentes do sistema, tais como: o clock – dispositivo que dá ritmo aos trabalhos da UCP, a UC, a ULA e até mesmo uma parte da memória conhecida por cache. A esses novos circuitos VLSI dá-se o nome de **PROCESSADOR**. Esse tal processador nos computadores derivados do IBM-PC, os mais populares no Brasil de hoje, acaba dando o nome ao sistema como um todo, ou seja, se no seu computador tem um processador **PENTIUM™IV**, costumamos dizer que este é o nome do computador em questão. Vejamos a seguir uma tabela que mostra a evolução desses processadores.

PROCESSADOR	ANO	CLOCK (MHz)	TRANSISTORES	LARGURA DA PALAVRA (bits)	OBSERVAÇÕES
4004	1971	0,108	2.300	4	Primeiro Microprocessador em um CI
8008	1972	0,108	3.500	8	Primeiro Microprocessador de 8 bits
8080	1974	2	6.000	8	Primeira CPU de uso geral em um só CI
8086	1978	5 a 10	29.000	16	Primeiro processador de 16 bits
8088	1979	5 a 8	29.000	8/16	Adaptação do 8086 para o IBM PC
80286	1982	8 a 12	134.000	16	CPU do PC-AT
80386	1985	16 a 33	275.000	32	Primeira CPU de 32 bits
80486	1989	25-100	1,2 milhões	32	Primeira CPU com cache e coprocessador aritmético embutidos.
Pentium	1993	60-233	3,1 milhões	32/64	Tecnologia Pipeline e MMX nos clocks mais elevados.

Pentium PRO	1995	150-200	5,5 milhões	32/64	Primeira CPU com dois níveis de cache
Pentium II	1997	233-450	7,5 milhões	32/64	Tecnologia MMX somada a dois níveis de cache
Pentium III	1999	650 a 1.400	9,5 milhões	32/64	Instruções específicas para imagens em 3D
Pentium IV	2000	1.300 a 3.800	42 milhões	32/64	Hiperthreading

Tabela 1.1 – Evolução dos Microprocessadores

Além desses processadores listados na tabela, existem outros fabricados pela Intel que são bastante populares nos dias atuais, tais como:

- **Celeron** → desde o primeiro Pentium, esse é um processador alternativo para quem não precisa de todo o poder computacional do Pentium “completo”, aqui sempre tem um recurso não implementado para justificar um preço mais baixo para o consumidor, sem prejuízo de acesso a tecnologia mais recente. O celeron mais recente é o CORE2-DUO, que é alternativo ao PENTIUM IV DUAL CORE;
- **XEON** → pronuncia-se zión, é uma família especial de Pentiums voltada para os servidores de rede. São processadores que contam com toda a tecnologia disponível no Pentium mais atual e são preparados especialmente para servidores de rede;
- **CENTRINO** → processadores específicos para dispositivos portáteis e móveis, como notebooks e outros. Contam com recursos especiais de gerenciamento de energia e acesso a redes sem fio.

Atualmente fala-se muito em CORE, seja dual, duo ou quad, essa denominação refere-se na verdade ao núcleo do processador, onde fica a ULA. Nos modelos DUAL ou DUO, esse núcleo é duplicado, o que proporciona uma execução de duas instruções efetivamente ao mesmo tempo, embora isto não aconteça o tempo todo. Basta uma instrução precisar de um dado gerado por sua “concorrente” que a execução paralela torna-se inviável, tendo uma instrução que esperar pelo término da outra. Os modelos QUAD CORE possuem o núcleo quadruplicado.

Esses são os processadores fabricados pela INTEL, empresa que foi pioneira nesse tipo de produto. Temos também alguns concorrentes famosos dessa marca, tais como NEC, Cyrix e AMD; sendo que atualmente apenas essa última marca mantém-se fazendo frente aos lançamentos da INTEL no mercado. Por exemplo, um modelo muito popular de 386 foi o de 40 MHz, que nunca foi feito pela INTEL, cujo 386 mais veloz era de 33 MHz, esse processador foi obra da AMD. Desde o lançamento da linha Pentium, a AMD foi obrigada a criar também novas denominações para seus processadores, sendo lançados modelos como K5, K6-2, K7, Duron (fazendo concorrência direta à idéia do Celeron) e os mais atuais como: Athlon, Turion, Opteron e Phenom.



<http://www.guiadohardware.net/comunidade/diferenc/732461/>



## EXERCÍCIOS

---

1. Explique como funciona a idéia de fazer uma arquitetura de Computadores em Camadas.
2. Como a evolução dos componentes eletrônicos tornou possível a popularização dos computadores?
3. A redução do tamanho dos computadores ao longo do tempo colabora para torná-los cada vez mais populares? Justifique sua resposta.
4. Explique como funciona a idéia proposta por Von Neumann em sua arquitetura.
5. No contexto dos microprocessadores, trace uma linha evolutiva com os principais marcos e lançamentos ao longo dos anos.
6. Liste os principais processadores fabricados pelos concorrentes da Intel.



## WEB BIBLIOGRAFIA

---

[http://www.inforquali.com/pt/tutorials/informatives/computer\\_history.php](http://www.inforquali.com/pt/tutorials/informatives/computer_history.php)

[http://www.fundacaobradesco.org.br/vv-apostilas/mic\\_pag3.htm](http://www.fundacaobradesco.org.br/vv-apostilas/mic_pag3.htm)

[http://pt.wikipedia.org/wiki/Cronologia\\_da\\_evolu%C3%A7%C3%A3o\\_dos\\_computadores](http://pt.wikipedia.org/wiki/Cronologia_da_evolu%C3%A7%C3%A3o_dos_computadores)

[http://oficina.cienciaviva.pt/~pw020/g3/historia\\_e\\_evolucao\\_dos\\_computad.htm](http://oficina.cienciaviva.pt/~pw020/g3/historia_e_evolucao_dos_computad.htm)

<http://www.guiadohardware.net/artigos/evolucao-processadores/>

<http://www.dcc.fc.up.pt/~zp/aulas/9899/me/trabalhos/alunos/Processadores/historia/evolucao.htm>

# Unidade 2

## Dispositivos de Entrada e Saída

### **Resumo**

No restante deste material usaremos a idéia de Von Neumann para “dissecar” um computador. Sendo assim, neste capítulo serão tratados os dispositivos de entrada e saída mais conhecidos nos computadores atuais. Será descrito o funcionamento destes e suas especificações técnicas. Na segunda metade desta unidade trataremos dos dispositivos de memória permanente.

## Sumário da Unidade

<b>UNIDADE 2 – DISPOSITIVOS DE ENTRADA E SAÍDA</b>	
2.1 – Dispositivos de Entrada e Saída .....	18
2.2 – Dispositivo de Armazenamento .....	30

## 2.1 Dispositivos de Entrada e Saída

Nessa categoria enquadram-se os dispositivos por onde o usuário envia dados para a UCP.

### Teclado

Um dos primeiros dispositivos necessários para essa finalidade é o teclado, também é o dispositivo que mais tem versões de acordo com a região onde é usado, embora exista o padrão norte-americano que é conhecido como teclado internacional (ou teclado padrão). Geralmente existe um teclado para cada língua ou até mesmo dentro do mesmo idioma podem existir adaptações de teclado para países diferentes, por exemplo, para o idioma português, existem teclados diferentes para Portugal (Pt) e para o Brasil (ABNT-2). As diferenças são, basicamente, localização de algumas teclas de acentos, cedilha e outras. Por exemplo, os brasileiros encontram o cedilha próximo ao ENTER, localização herdada das máquinas de escrever, já os portugueses encontram o cedilha próximo a tecla do número 1, respeitando também a tradição local deles.

O teclado nada mais é que um conjunto de chaves elétricas cada uma acionada por uma tecla. A cada chave dessas corresponde um código binário que é enviado para a placa-mãe e esta sabe como tratar esse código de acordo com o modelo do teclado ou com o software que está em uso.



Figura 2.1 – Teclado ABNT2

### *Códigos de Caracteres*

Os códigos de caracteres são definidos basicamente por duas tabelas EBCEDIC (muito usada em computadores de grande porte da IBM) ou ASCII que é o padrão dominante nos demais computadores. Essas tabelas

basicamente têm uma parte fixa com correspondências dos caracteres mais usados, e outra parte variável que pode ser configurada de acordo com o idioma do sistema operacional instalado ou até mesmo um idioma diferente do S.O., mas configurado por este.

Por exemplo, em um computador com o Windows em Inglês, pode-se informar que o país é Brasil, o idioma local é Português e o teclado é ABNT-2 (se for o caso). Então mesmo com o Windows em Inglês, passa a ser possível usar todos os acentos gráficos da língua portuguesa, o símbolo monetário passa a ser o Real (R\$) e as teclas do teclado passam a corresponder aos símbolos que mostram.

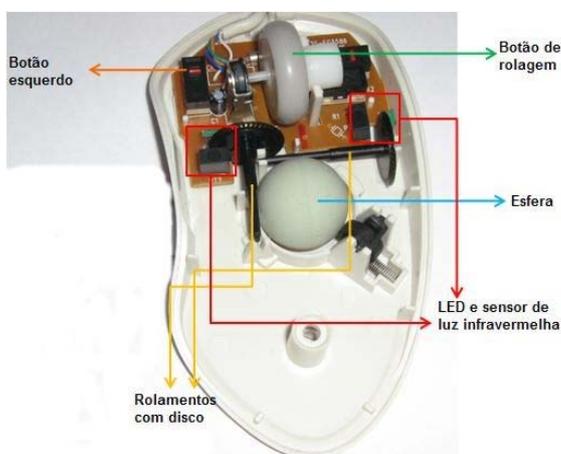


Figura 2.2 – Mouse opto-mecânico

## Mouse

Este dispositivo é bastante usado também nos dias de hoje, pois os sistemas estão cada vez mais intuitivos e organizados de forma gráfica (figuras na tela). Para usar o modo gráfico dos sistemas o mouse é fundamental, pois seu movimento na mesa é refletido na setinha da tela que faz os apontamentos necessários nas figuras.

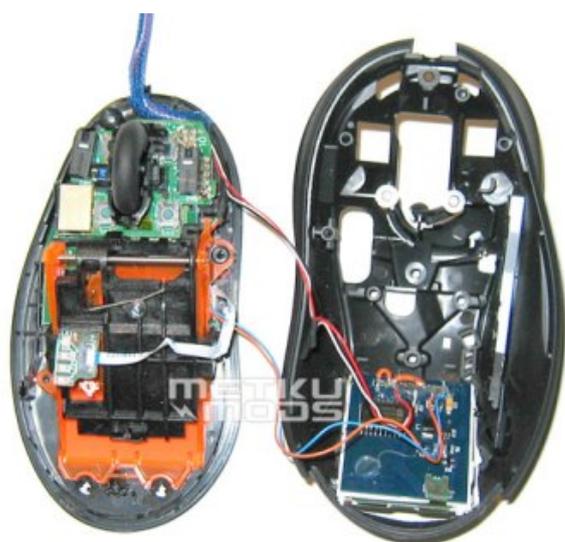


Figura 2.3 – Mouse óptico

Existem basicamente duas tecnologias para fabricação dos mice (este é o plural de mouse): a opto-mecânica e a óptica. A primeira tecnologia baseia-se em uma pequena esfera, normalmente de borracha, que gira embaixo do mouse e faz girar dois eixos, conhecidos como x e y. Dentro do aparelho cada eixo é ligado a uma roda com vários furos. De um lado da roda tem uma luz emitida por um LED (diodo emissor de luz) e do outro lado tem um sensor, as interrupções da luz no sensor significam que a roda está girando e conseqüentemente o mouse está sendo movimentado. Esses movimentos são enviados para a UCP por uma comunicação chamada de serial (mais à frente serão detalhadas essas formas de comunicação). Para melhorar a precisão desse

tipo de mouse, normalmente usa-se um tapete emborrachado sobre a mesa para que a esfera gire de forma mais precisa. O segundo tipo de mouse está ficando muito popular atualmente: é o óptico, que a princípio, já dispensa o famoso tapete (conhecido por mouse pad). Somente os primeiros mice ópticos, fabricados na década de 90 do século passado, precisavam de mouse pad especiais espelhados. Os mice ópticos atuais não precisam de qualquer tapete embaixo. A tecnologia desse dispositivo baseia-se simplesmente em enviar a luz do LED para a mesa e esperar o reflexo desta de volta para os sensores da base do mouse. De acordo com a emissão da luz e seu retorno, detectam-se os movimentos e enviam-se os dados sobre estes para a placa-mãe.

## Vídeo

O vídeo do computador na verdade é o dispositivo de saída mais popular que existe. O computador mais básico tem pelo menos dois dispositivos de entrada que seriam o teclado e o mouse e um dispositivo de saída que é o vídeo.

O chamado subsistema de vídeo do computador constitui-se de uma das partes mais complexas, pois requer um processador específico para os gráficos a serem exibidos, esses gráficos processados são armazenados na chamada memória de vídeo que recentemente passou a ser uma parte da RAM nas configurações mais populares. Isso acontece devido a novas tecnologias de barramento (também a serem vistas mais adiante). Finalmente a imagem processada e armazenada em memória específica é exibida em um monitor de vídeo, que por sua vez pode ser baseado na tecnologia de raios catódicos (CRT) ou de cristal líquido (LCD).

O tubo CRT na verdade é uma grande válvula onde os elétrons que formam a imagem são bombardeados contra a parede frontal da tela dentro de um tubo de vácuo. Esse



Figura 2.4 – Placa de Vídeo



Figura 2.5 – Monitor CRT

bombardeio ocorre obedecendo determinados critérios. Se a varredura ocorre em linhas pares e ímpares da tela ao mesmo tempo, diz-se que é um monitor tipo não entrelaçado. Se a varredura ocorre alternando as linhas pares e ímpares é dito que o monitor é entrelaçado (normalmente mais barato que o outro). A velocidade de varredura é também muito importante, medida em *Hertz*, quanto mais Hz de varredura melhor será o funcionamento do monitor e, literalmente, menos dor de cabeça dará ao usuário. Então, antes de comprar um monitor CRT, é importante saber, além da resolução que ele aceita, normalmente medida em pontos, algo como 1024 x 768, é importante saber também se este é entrelaçado e de quantos Hz é sua varredura, pois esses são fatores que influenciam no preço, na qualidade da imagem e futuramente também na saúde de quem vai ficar horas diante do aparelho. Uma medida muito comum para designar qualidade de imagem de monitores CRT é o DOT PITCH. Normalmente fica em torno de 0,26mm para os modelos de 15 polegadas. Essa medida informa a distância diagonal de um ponto para outro na tela. Quanto menor o dot pitch, melhor a qualidade da imagem formada. Isso explica até mesmo diferenças de preços entre aparelhos de televisão de mesmo tamanho e de mesma tecnologia e, às vezes, até mesmo da mesma marca, é apenas o dot pitch da tela. Mas os vendedores de eletrodoméstico não sabem o que é esse conceito, não adianta insistir.

Os monitores de CRT recebem classificações de acordo com sua capacidade de exibir gráficos e a qualidade desses gráficos em exibição. Tudo começou com o monitor Hércules, que não contava com capacidade de exibição de gráficos. Passou pela fase do CGA (Computer Graphics Array), onde gráficos rudimentares passaram a ser exibidos; evoluiu para o EGA (Enhanced Graphics Array), onde essas imagens ganharam a opção de cores e melhor definição, medida pelo aumento da resolução, que é a quantidade de pontos que formam a imagem na tela, e finalmente, temos os VGA (Vídeo Graphics Array) e suas variações. Sendo atualmente usadas as variações SVGA ou XGA. A tabela a seguir sintetiza toda essa evolução.

MODELO	RESOLUÇÃO (PONTOS)	CORES	OBSERVAÇÕES
Hércules MDA	25 linhas x 80 colunas	1	Somente texto
CGA	320 x 200	4	Primeiro a suportar gráficos
EGA	640 x 450	16 ou 64	Gráficos com mais qualidade e compatibilidade com CGA
VGA	640 x 480	256	256 cores são suficiente para o olho humano ser enganado e achar que está vendo cores reais.
	800 x 600	16	
SVGA	800 x 600 e acima	1,6 milhões em 24 bits ou 4 milhões em 32 bits	Este vídeo mostra cores no limite da percepção humana 1,6 milhões de cores simultâneas, a opção dos 32 bits pode ser usada para melhorar o contraste da imagem.

Tabela 2.1 – Evolução dos monitores CRT

Os modelos baseados em painéis de cristal líquido (LCD) são menos volumosos que os de CRT, mas nem por isso são menos complexos. São na verdade compostos de várias camadas, onde a camada central é o chamado cristal líquido, as demais são camadas polarizadoras, que submetem cargas elétricas a esse cristal para formar a imagem. São pelo menos três camadas à frente do cristal e mais três atrás deste, todas flexíveis. Isso explica as deformações que a imagem sofre ao ter a tela tocada por um dedo ou outro objeto qualquer. Ao fundo de tudo tem-se uma fonte de luz para a imagem tornar-se visível. Essa tecnologia, embora tenha se tornado popular nos últimos anos, vem sendo desenvolvida desde 1960. Seu uso tornou possível a construção, por exemplo, dos relógios digitais, que já não são mais novidade desde uns trinta anos atrás.



Figura 2.6 – Monitor LCD

## Scanner



Figura 2.7 – Scanner de mesa

Dispositivo de entrada muito popular e cada vez mais usado. Pois é comum chamar de scanner o aparelho de mão ou de mesa onde as fotos em papel eram transformadas em arquivos de computador. Mas também é comum esquecer que os códigos de barras dos produtos nas lojas são lidos por scanners, os terminais de banco também os tem para ler os códigos de barras

das contas e documentos a serem processados. Isso tudo sem falar na praticidade duvidosa dos aparelhos multifuncionais que agregam scanner, impressora, copiadora e algumas vezes, fax.



Figura 2.8 – Scanner de código de barras

Esse aparelho na verdade é composto por uma fonte de luz que é emitida sobre o papel, ou objeto a ser “escaneado” e um sensor que capta os reflexos dessa imagem. Esse sensor ajuda a montar a imagem na memória do computador na forma de um arquivo gráfico ou simplesmente um padrão de barras a ser convertido em um número.

Existem também os scanners biométricos, usados na chamada identificação biométrica, nesses casos, usa-se uma parte do corpo humano que sirva para identificar o indivíduo pela imagem. Normalmente esses aparelhos pedem a presença de um dos dedos para a leitura da impressão digital e sua armazenagem ou comparação com uma já armazenada. No lugar da impressão digital, o scanner biométrico pode ler também a íris do olho, pois essa também é uma forma muito eficiente de identificar a pessoa.



Figura 2.9 – Scanner Biométrico para a íris

Para qualquer uso do scanner são necessários vários programas específicos para cada atividade, desde o processamento de imagens, tais como fotos, feitos por programas como COREL PHOTO ou PHOTOSHOP, passando pela leitura de códigos de barras do comércio, dos bancos ou de outros documentos. Mais específicos ainda são os softwares para uso em scanners biométricos. Portanto, não basta ter o aparelho, porém para uso nos sistemas de Internet Bank, normalmente basta ter o leitor de códigos de barras adaptado ao teclado do computador, o próprio

site do banco encarrega-se de fazer a leitura e decodificação das barras de suas contas a pagar.

## **Impressora**

Talvez depois do monitor de vídeo seja este o dispositivo de saída mais conhecido por todos. É o dispositivo responsável por levar ao papel tudo que se produz em termos de textos e imagens dentro de uma UCP. Existem diversas tecnologias de impressoras, dentre elas tem-se: as de impacto, as jato de tintas, as de cera e as a laser. A qualidade dos impressos vai depender da tecnologia aplicada a cada impressora e também de uma característica fundamental, a resolução desta. Medida em DPI (ponto por polegada) a resolução faz a impressora ser destinada a textos ou a gráficos ou ainda a fotos. Um texto fica legível com cerca de 150 DPI, um cartaz bem grande para ser visto a longa distância pode ser impresso até em 75 DPI, mas gráficos só ficam bem visíveis no papel se estiverem ao menos a 300 DPI, fotos ficam razoáveis com 300 DPI, porém o céu é o limite para as fotos, existem hoje impressoras que fazem milhares de DPI para o modo fotográfico. Lógico que isso tem um custo. Para quem se contenta com poucos DPI, a tinta dura mais, a impressão fica mais rápida. Tudo depende da necessidade do usuário.

### *Impacto*

Essa categoria de impressora baseia-se em usar uma fita com tinta sendo pressionada por martelos em relevo, esses modelos são conhecidos por MARGARIDA, tecnologia obsoleta onde, para trocar a letra de um texto, era necessário trocar a tal margarida da impressora. Esse tipo de impressora não faz gráficos.

Uma evolução das impressoras de impacto são as matriciais, onde uma matriz de agulhas fica por trás da fita entintada e pressionam a fita alternadamente fazendo as letras,



Figura 2.10 – Impressora Matricial

pontos e traços que compõem os gráficos. Existem impressoras matriciais de 9, 18 e 24 agulhas, quanto mais agulhas mais precisa a impressão. Porém são modelos que raramente ultrapassam os 150 DPI e ainda fazem estrago em papéis finos, pois usam impacto. A cor nesse tipo de impressora é um pesadelo. Mas o baixo custo da fita e baixo consumo de energia aliados a simplicidade de projeto, que dispensa manutenção, fazem deste tipo de impressora as preferidas do comércio.

### *Jato de Tintas*

Tecnologia inspirada nas canetas-tinteiras onde um cartucho de tinta fica ligado à chamada cabeça de impressão, em alguns modelos são substituídos juntos. Essa cabeça é composta por uma série de eletrodos que recebem sinais da placa da impressora para fazer com que microbolhas de



Figura 2.11 – Impressora a jato de tintas

tinta gotejem sobre o papel. Quanto maior a precisão desse gotejamento, mais DPI terá a impressora. A tinta por sua vez, deve ser quimicamente preparada para gotejamentos especiais, por isso que os modelos mais sofisticados de impressoras a jato de tinta costumam ter cartuchos bem caros, alguns chegam a custar mais que a própria impressora. Esse detalhe explica o porquê dos preços altos dos cartuchos de tinta, mas não os justifica, pois existe também uma grande parcela de ganância por trás desse fato. É muito comum hoje, e impressoras a jato de tintas chegam a até 5.400 DPI em modo fotográfico. O uso de tintas genéricas, como são

conhecidas no Brasil, depende do conhecimento do usuário, pois alguns genéricos são ruins demais, mas outros chegam a ser melhores que os originais do fabricante, e bem mais baratos, mas todos os fabricantes se recusam a dar assistência se no prazo de garantia for usada uma tinta genérica. Um cartucho de tinta raramente rende uma resma de papel impresso. Portanto, é importante ponderar os prós e contras dessa categoria de impressoras, bem como seu custo-benefício.

## Cera

Categoria de impressora criada para ter cor no impresso com qualidade de laser, porém o custo elevado de manutenção aliado ao surgimento da laser colorida fizeram essa tecnologia ser esquecida. A idéia aqui é usar uma sublimação de cera (aquela do lápis de cera) para fazer impressão.

## Laser

Este tipo de impressora é bastante popular também, ainda mais se a intenção é impressão monocromática. Seu preço mais elevado em relação ao da jato de tintas compensa-se na qualidade do impresso em preto e na duração de sua tinta, um pó chamado toner. Uma carga de toner chega a imprimir cerca de 3.000 páginas em preto e branco. Existe a laser colorida também, que usa um toner especial e caro. Esse tipo de impressão colorida ainda é inviável se comparada à de jato de tintas.

O funcionamento dessa impressora acontece da seguinte forma: um raio laser é emitido de acordo com ordens da placa da impressora, atinge um cilindro recoberto por toner que “carimba” o papel e este papel por sua vez é submetido a cilindros aquecidos para fixar o toner. Isso justifica o fato de uma página impressa sair quente. Como o funcionamento da fotocopadora é análogo ao dessa impressora, justifica-se também aqui a temperatura de fotocópias assim que saem da máquina. Essas impressoras também contam com boa resolução, algumas acima dos 1.200 DPI.

## Considerações sobre as impressoras

Existem algumas propagandas dúbias quanto às impressoras. A maioria dizem respeito a relação entre qualidade de impressão e velocidade. Impressoras a jato de tintas de 10 páginas por minuto (PPM) com 5.700 DPI, normalmente fazem as 10 PPM apenas no modo rascunho,

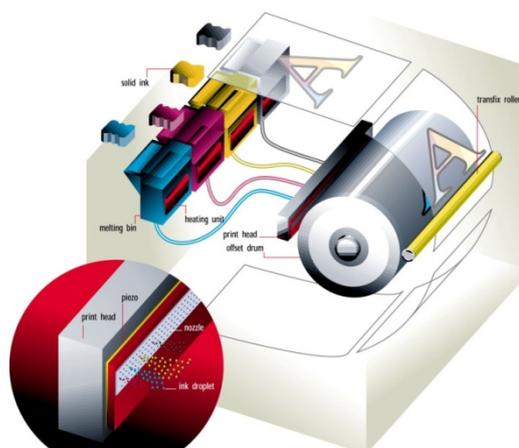


Figura 2.12 – Esquema de funcionamento da impressora a cera (tinta sólida)



Figura 2.13 – Impressora a laser

algo em torno de 100 DPI. Quando a qualidade dos 5.700 DPI está ativada normalmente, esse tipo de impressora demora uns dois minutos para terminar uma página.

Outro fato interessante é a recarga de toner das laser, e também das fotocopiadoras. O toner é um pó sensível à luz, uma vez exposto ele começa a aderir à superfície de contato na forma de tinta, portanto ao carregar um cartucho de toner é fundamental ter cuidado com os olhos, uma coçadinha de olho com dedos sujos de toner pode cegar o sujeito para sempre.

Finalmente, as impressoras matriciais têm tampas e abafadores próprios para reduzir a barulheira e ajudar a fazer a trajetória do papel. Tirar as tampas desse tipo de impressora significa expor seus ouvidos (e de quem estiver próximo) a barulhos desnecessários. A qualidade de impressão que já não é tão boa fica pior ainda com linhas falhadas, devido à falta das tampas e a impressora ainda tem sua vida útil reduzida pela poeira que entra nos circuitos e na fita de tinta.

### Câmeras

As câmeras digitais são certamente um periférico bastante popular. Seja no formato webcam para conferências pela Internet, seja no formato câmera fotográfica ou filmadora, seja profissional ou amadora.



Figura 2.14 – Webcam

O funcionamento da câmera não foge à tecnologia do tempo do filme. Existe um espaço escuro dentro de cada aparelho onde a imagem captada pelas lentes é formada. O que muda é a maneira como esta imagem ficará registrada, antes em um filme plástico recoberto por uma substância química foto-sensível, agora esta imagem fica em um dispositivo conhecido como arranjo de CCDs (Dispositivos de Carga Emparelhada). São vários CCDs, cada um detectando e capturando um sinal de luz e cor. Essa luz é filtrada e montada dentro da memória interna da câmera, que utiliza algum algoritmo de compressão de imagens para gerar um arquivo que seja compatível com computadores e outros dispositivos digitais. O algoritmo usado para fotos comumente é o JPEG que também é um dos

mais eficientes da atualidade para gravar fotos sem perder a qualidade e com máxima economia de espaço em memória. Para filmagens, o formato mais usado em gravações é o MPEG, camada 2 (gravação direta em mini-DVD) ou camada 4 (gravação em memória).

Um exemplo prático de uso de câmera digital: imagine uma câmera com 6 milhões de CCDs, juntos eles detectam cores básicas e outros detalhes de imagem. Porém essa quantidade de CCDs é suficiente para 1,5 milhões de pontos de imagem (pixels), ou uma imagem 3.000 x 2.000 pontos. Na verdade essa imagem tem 1,5 MEGAPIXEL, porém os fabricantes, pelo fato de usarem 6 milhões de CCDs afirmam que sua máquina tem 6 megapixels. Sem falar que as câmeras de baixo custo ainda fazem a chamada interpolação por software, usando ainda menos CCDs para captar a imagem e gerando arquivos com mais megapixels que o que foi realmente captado.



Figura 2.15 – Câmera Digital

Mais detalhes sobre pixels e interpolação por software o leitor terá ao chegar aos estudos de computação gráfica.

## MODEM

Esse periférico está cada vez mais popular. Seu nome vem de MOdulador-DEModulador de sinal, ou seja, é o aparelho que converte o sinal digital que circula dentro do computador em ruídos que possam ser transmitidos pela linha telefônica. Existem modems embutidos no computador, chamados internos e existem modems que ficam fora do computador, os externos. A essa categoria de modem externo somam-se ainda os novos modems DSL. O estudo mais detalhado desse importante periférico com certeza será feito nas disciplinas de redes de computadores. Po-

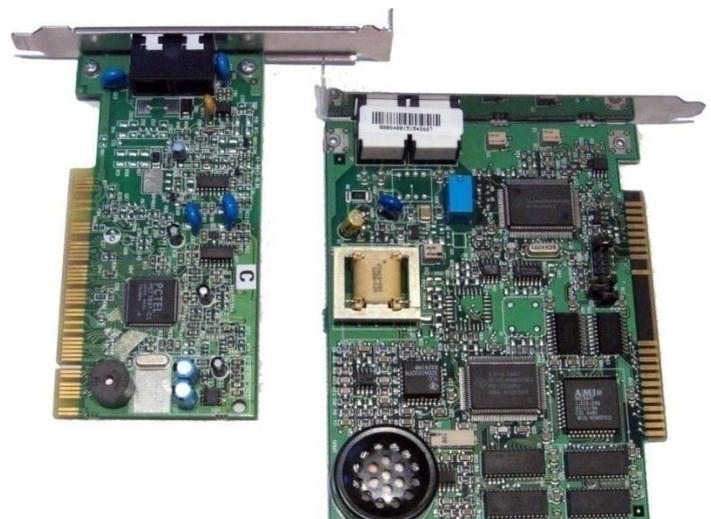


Figura 2.16 – Softmodem e Hardmodem internos

rém nesta disciplina, podemos afirmar que esse aparelho vale-se daquele sinal sonoro presente no telefone, a chamada PORTADORA. O sistema telefônico também usa essa onda para várias operações, desde discagem até informar ao usuário que do outro lado o telefone está chamando, ou ocupado e assim por diante. A Portadora, como toda onda sonora tem frequência, amplitude e fase. Comunicações entre modems ocorrem realizando modificações em uma dessas características da onda portadora.



Figura 2.17 – Modem DSL (ADSL) com saída para rede sem fio

Outro fato importante sobre modems é que existem aqueles que são baseados em hardware e deixam a UCP do computador mais livre para realizar outros processamentos. Outra opção são os mais baratos e populares softmodems, que se constituem de uma interface onde o cabo telefônico é conectado e todo o resto dos procedimentos a serem realizados pelo modem ficam a cargo da UCP do computador. A figura a seguir ilustra um softmodem e um modem implementado em hardware.

Existe uma diferença na tecnologia DSL, da chamada “banda larga”; é que o sinal modificado não é especificamente a Portadora do telefone e sim, um sinal paralelo enviado pela linha telefônica em uma frequência não ocupada pela portadora. Isso torna possível comunicações em velocidades superiores à linha discada tradicional, mas obriga o sistema a ter equipamentos específicos para esse tipo de comunicação, desde a infra-estrutura da operadora telefônica até a adição de um modem DSL no computador do usuário.

## 2.2 Dispositivos de Armazenamento

Neste tópico serão explorados os dispositivos de armazenamento das diversas arquiteturas, sendo de padrão magnético como as fitas e discos, ou sendo ópticos como CDs, DVDs e Blu-rays. Geralmente as mídias mais caras tendem a ser as mais rápidas e também as de menor capacidade, exceto se forem algum lançamento, como no caso do Blu-ray.

### FITAS MAGNÉTICAS

Dentre as mídias usadas para armazenamento, talvez essa seja a mais antiga e que mais evolui, pois sempre há uma novidade envolvendo a fita magnética, uma mídia muito popular para cópias de segurança de dados, os chamados backups. Normalmente as fitas saem de fábrica ainda virgens, ou seja, como são fitas de plástico flexível recobertas de íons de um óxido que seja composto com ferro ou com cromo, uma vez ordenados, estes armazenam informações tanto analógicas quanto digitais; essas fitas saem de fábrica com os íons desordenados na forma como foram aplicados no material, daí serem chamadas “virgens”. Para uso em backup uma fita deve ser formatada, ou seja, preparada para gravações digitais. Aliás, toda mídia magnética deve ser formatada, a diferença para discos e disquetes é que estes normalmente já vêm formatados de fábrica. Toda fita possui uma gravação especial no seu início onde fica o DIRETÓRIO e em seguida a FAT. Para se ter uma organização dos espaços ocupados pelos arquivos, durante a formatação, a fita fica dividida em áreas de mesmo tamanho onde serão gravados os arquivos de forma total ou parcial, pois um arquivo pode ocupar mais de um espaço destes. Entre os espaços de gravação ficam áreas de separação conhecidas por GAP. Assim, uma vez rebobinada a fita e consultado o diretório, a unidade de fita realiza o avanço contando os GAPs para saber onde começa a gravação do arquivo solicitado.

## DISCOS MAGNÉTICOS

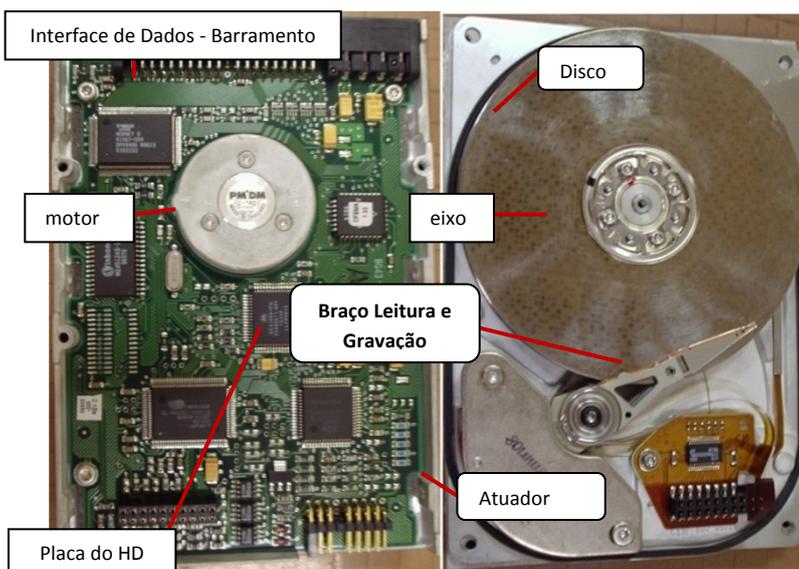
Essa categoria conta com uma subdivisão especial, têm-se os **discos fixos**, conhecidos por “winchester”, ou discos rígidos porque seu material interno normalmente é alumínio. E a outra categoria são os discos flexíveis, os populares disquetes, são tidos como flexíveis porque são feitos do mesmo material da fita magnética.

### *Disco Rígido*

São componentes internos do computador formados por uma série de discos empilhados sobre o mesmo eixo. Cada disco aceita gravações em ambas as faces, normalmente são feitos em duas camadas, onde a primeira é conhecida como substrato, normalmente alumínio, e a segunda, de material magnético para poder receber as gravações.

Como a camada magnética é extremamente fina, deve ser recoberta por uma finíssima camada protetora, que oferece alguma proteção contra pequenos impactos. O braço que movimenta as cabeças que fazem leituras e escritas no disco move-se a uma distância inferior a espessura de um fio de cabelo da superfície do disco. Este por sua vez gira muito rápido, a caixa onde tudo está montado é fechada. Isso garante uma pequena flutuação dos íons que contêm os dados gravados, portanto, as cabeças não chegam a tocar efetivamente no disco. Os discos são montados em um eixo também feito de alumínio, que deve ser sólido o suficiente para evitar

qualquer vibração dos discos, mesmo a altas rotações. Finalmente, o motor de rotação é responsável por manter uma velocidade constante. Os primeiros discos rígidos utilizavam motores de 3.600 rotações por minuto, os atuais contam com motores de 5.600 ou 7.200 RPM, alguns chegam a mais de 10.000 RPM em modelos mais caros. A velocidade de rotação é um dos principais fatores que determinam o desempenho.



Enquanto o disco rígido está desligado, as cabeças de leitura ficam numa posição de descanso, longe dos discos magnéticos. Elas só saem dessa posição quando os discos já estão girando à velocidade máxima. Para prevenir acidentes, as cabeças de leitura voltam à posição de descanso sempre que não há dados sendo acessados, apesar dos discos continuarem girando. Vibrações na mesa, faltas de energia durante acessos, transportar o computador funcionando mesmo que por uma distância muito pequena são fatores que contribuem para o surgimento de defeitos no disco rígido, pois são situações onde as cabeças estão se movimentando sobre o disco e podem tocá-lo provocando arranhões irreparáveis em sua superfície.

### *Discos Flexíveis*

Os populares disquetes funcionam de modo bem mais simples que os discos rígidos. Pois são feitos de material flexível e não suportam altas velocidades de rotação, daí a lentidão desse tipo de mídia. Esses discos possuem uma baixa capacidade, algo que nunca ultrapassou a barreira dos 2,5 MB, sendo que os mais populares ficam com apenas 1,38 MB de capacidade.

**Dica importante:** Os disquetes de 1,38 MB são conhecidos como discos de 1,44MB de capacidade, mas isso é falso. É fato que 1MB corresponde a 1.024KB e assim sucessivamente, porém, desde quando foi percebido que era melhor fazer propagandas de discos de 1,44 do que de 1,38, a indústria e os marketeiros da informática tendem a fazer conversões do tipo 1 para 1.000 e não 1 para 1.024, que é o correto. Por isso que compramos computadores com, por exemplo, 120GB de disco rígido e ao usarmos constatamos que temos pouco mais de 100GB de disco. Isso abre precedentes para reclamações de propaganda enganosa. Depende apenas de conscientização do usuário.

## DISCOS ÓPTICOS

### CD

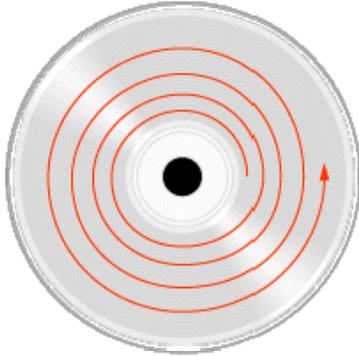


Figura 2.19 – Esquema de gravação de um CD ou DVD

A gravação de um CD dá-se em forma de espiral, começando do centro para a borda. Um laser de alta potência faz pequenos sulcos na espiral conhecidos como PITS. Os locais onde a espiral não é marcada pelos pits são conhecidos como LANDs. Existem diversos formatos de gravação de CDs, os mais populares são o CD de áudio, que segue o chamado padrão RED BOOK; tem-se ainda o CD de dados padrão YELLOW BOOK, que não aceita multisessão, ou seja, a gravação tem que acontecer de uma vez só. Além desses, outros padrões são bastante populares tais como: o GREEN BOOK, que criou o CD interativo; o ORANGE BOOK, criou o cd multisessão, aquele que pode ser gravado “aos poucos” e também passou a ser possível usar o CD regravável e finalmente o WHITE BOOK, que tornou possível gravar VCD.

A gravação de CD-RW se dá de forma diferente, devido a propriedade de ser possível apagar. Na verdade, o gravador compatível com CD-RW usa o laser em três níveis de potência. A mais alta grava o CD, a média retorna os PITS ao nível dos LANDs, fazendo assim o apagamento do CD e, finalmente, o laser de baixa potência faz a leitura dos CDs.

Os CDs têm capacidade variando de 650MB a 700MB. Alguns até ultrapassam essa capacidade, porém seu uso é contra-indicado pelos fabricantes de gravadores de CD, pois a gravação chega perigosamente à borda do disco e pode haver uma refração de laser de alta potência danificando algum circuito interno do gravador. Os mini-CDs também são bastante usados quando se quer gravar até cerca de 120MB a 210MB de dados, pois são bem pequenos (apenas 3 polegadas) e facilitam a distribuição de arquivos até essa capacidade. Porém o uso dessa mídia está condicionado ao fato de a gaveta do drive ter o sulco mais interno onde esse se encaixa.

## *DVD*

Enquanto nos CDs existem somente os modelos CD-R e CD-RW, nos DVDs existem mais padrões, ou seja, não existe um consenso de padrão de DVD gravável e de DVD regravável. Nos DVDs graváveis tem-se DVD-R e DVD+R, nos regraváveis tem-se DVD-RW, DVD+RW e DVD-RAM. E em todos os casos tem-se também o DVD de dupla camada, que possui o dobro da capacidade nominal dos DVDs simples equivalentes.

A mudança principal do DVD em relação ao CD é a proximidade dos pits que é maior. Com os dados gravados em densidade maior pode-se ter mais capacidade com o mesmo diâmetro de disco (em torno de 5 polegadas). Enquanto a capacidade de um CD está em torno de 700MB, os DVDs variam de 4,3GB até 17GB dependendo da tecnologia empregada na confecção da mídia. Também estão disponíveis os mini-DVDs. Com capacidade em torno de 1,2GB, são o formato preferido pelas câmeras filmadoras que usam DVD como mídia de gravação. Porém seu uso também está condicionado à existência do sulco interno na gaveta do aparelho reproduzidor ou do drive.

Uma novidade do DVD em relação ao CD é o surgimento da tecnologia de dupla camada, onde a espiral é criada do centro para a borda e, ao chegar à borda, o processo de leitura ou gravação retorna em espiral rumo ao centro. Isso faz com que a capacidade do DVD torne-se o dobro da inicial. Internamente, existem duas camadas da substância que recebe a gravação em níveis diferentes. Daí a nomenclatura da tecnologia. Na mudança de camada o drive ou o leitor de DVD faz uma pequena pausa, pois se faz necessária uma mudança no sentido de rotação do disco.

Por ter tanta capacidade a mais, essa mídia tornou-se a preferida para a distribuição de filmes para o mercado doméstico, isso aliado ao ganho de qualidade de imagem e som em relação às fitas de VHS. Logo, a nova mídia tornou-se padrão do mercado e fonte de muitas dores de cabeça para os produtores de cinema e televisão.

## *Blu-ray*

Os fabricantes conseguiram uma densidade de gravação ainda maior nessa nova mídia que funciona com um laser de cor azul (daí o nome de blu-ray). A capacidade de armazenamento subiu para algo entre 25GB e 50GB. Além de um enorme espaço para backup, essa mídia torna possível a gravação de filmes com ainda mais realismo em relação ao DVD. Portanto, essa mídia tende a ser um substituto natural do DVD para os próximos anos.



## EXERCÍCIOS

---

1. Explique o funcionamento dos teclados.
2. Como são implementadas as tabelas de caracteres e como são adaptadas para cada país?
3. Como funcionam os mice (plural de mouse)?
4. Explique o funcionamento do subsistema de vídeo do computador, incluindo o monitor tipo CRT e LCD.
5. Como funcionam os mais diversos scanners?
6. Explique o funcionamento de cada um dos tipos de impressoras.
7. Como funcionam as câmeras digitais?
8. Explique a operação dos modems.
9. Mostre as principais diferenças entre os discos ópticos e magnéticos.
10. Diferencie CD, DVD e Blu-ray.



## WEB BIBLIOGRAFIA

---

[http://pt.wikipedia.org/wiki/Disco\\_Blu-ray](http://pt.wikipedia.org/wiki/Disco_Blu-ray)

<http://informatica.hsw.uol.com.br/perifericos-canal.htm>

[http://pt.wikipedia.org/wiki/Categoria:Periféricos\\_de\\_computador](http://pt.wikipedia.org/wiki/Categoria:Periféricos_de_computador)

<http://www.laercio.com.br/>

<http://www.infowester.com/>

# Unidade 3

## Memória

### **Resumo**

Esta unidade destina-se a apresentar as vias de comunicação existentes dentro do computador, estas vias são conhecidas como barramentos, quando estes barramentos tem saídas para o meio externo a placa-mãe ganham o nome de interfaces, que devem seguir um padrão bem definido para que os periféricos conectados sejam capazes de se comunicar com a UCP.



## Sumário da Unidade

<b>UNIDADE III – MEMÓRIA</b>	
3.1 – Introdução .....	38
3.2 – Memória ROM .....	38
3.3 – Memória Flash .....	39
3.4 – Memória RAM .....	41
3.5 – Detecção e Correção de Erros .....	44
3.6 – Memória CACHE .....	45
3.7 – Hierarquia de Memórias .....	50

### 3.1 Introdução

Para um bom entendimento deste capítulo é fundamental que seus conhecimentos de Circuitos Digitais estejam bem vivos na memória.

As memórias serão abordadas, para tanto, é interessante lembrar – se de portas lógicas, seus funcionamentos, bem como os flip-flops e latches.

### 3.2 Memória ROM

Esse tipo de memória está presente em todos os computadores modernos e em grande parte de outros dispositivos eletrônicos presentes em casas e escritórios de hoje em dia.

É uma memória que não se perde quando a energia é cortada, ou seja, é uma memória não-volátil. Contém programas de ajustes ou de inicialização de algum circuito. Por exemplo, a memória ROM dos computadores possui um programa chamado BIOS (Sistema de Inicialização de Entradas e Saídas). Esse programa é ativado assim que o computador é ligado, durante um processo conhecido como BOOT (Operações Iniciais de Testes). Serve para verificar o funcionamento básico dos principais componentes do sistema tais como: CPU, memória RAM, Subsistema de Vídeo, Teclado e Discos rígidos.



Figura 3.1 – Memória ROM em Placa-mãe

A memória ROM clássica não pode ser alterada ou apagada. Mas existem algumas variações desse tipo de memória que podem ser altera-

das, isso vai depender do tipo de Circuito Integrado usado na fabricação desta ROM. Dessa forma, existem as classificações de ROM:

- PROM → ROM programável. Este chip vem de fábrica sem nenhuma gravação. Através de um periférico especial chamado “gravador de PROM”, podemos gravar um software nele e então transformá-lo em ROM, já que neste circuito o processo de apagamento não é permitido.
- EPROM → ROM apagável e programável. Esse tipo de memória ROM pode ser gravada como a anterior, porém se for necessário, existe uma pequena janela de acrílico coberta por uma etiqueta metálica, que pode ser removida e na janela ser incidida luz ultravioleta. Isso provoca o apagamento da EPROM, tornando-a novamente pronta para ser gravada.
- EEPROM → ROM apagável e programável eletronicamente. Esse tipo de ROM pode ser atualizado por software. É o tipo mais prático e também o mais perigoso, pois a praticidade de atualização pelo sistema operacional também implica risco de gravação de vírus e outros programas intrusos que possam estar no computador que acessa esse tipo de memória.



### 3.3 Memória Flash

A memória Flash permite armazenar dados por longos períodos, sem precisar de alimentação elétrica. Graças a isso, a memória Flash tornou-se rapidamente a mídia dominante em cartões de memória, pendrives, HDs de estado sólido (SSDs), memória de armazenamento em portas NOR ou NAND. O primeiro tipo está sendo pesquisado desde 1988 e é uma mídia de leitura rápida e gravação lenta. Essa primeira tecnologia não foi muito popularizada e, portanto, é mais cara que a NAND.

Os aparelhos digitais como câmeras e celulares foram ficando cada vez mais populares e isso provocou também uma grande busca por novidades na produção desse tipo de memória. A densidade de gravação tornou-se cada vez maior e isso levou a fabricação de cartões de memória como os de hoje, com vários gigabytes de capacidade.

Alguns tipos de cartões de memória:

### 1 – Compact Flash

Esse tipo de memória usa um barramento muito parecido com os discos rígidos IDE. Em algumas arquiteturas esse cartão chega a substituir o disco rígido devido à similaridade do barramento. Esse é o caso de alguns tipos de urnas eletrônicas usadas no Brasil.



Figura 3.2 – Memória Compact Flash

### 2 – Smart Media

Esse tipo de cartão representa um marco nesta indústria, pois inaugura o formato de cartão de memória usado hoje, em forma de cartão de crédito e sem precisar de interfaces especiais. O próprio leitor de cartões poderia ser plugado em uma entrada USB ou equivalente e os dados seriam acessados sem problema. Esse padrão foi criado pela Toshiba. Porém o tamanho aproximado de um cartão de crédito era muito grande para os planos dos fabricantes de eletrônicos e bens de consumo. Isso levou ao surgimento de padrões como MMC, SD, xD e outros.



Figura 3.3 – Cartão Smart Media

### 3 – xD

Esse cartão é um formato proprietário usado pela FUJI e OLYMPUS. Possui tamanho muito reduzido e grande capacidade de armazenamento. Seu principal problema é na lentidão no acesso aos dados tanto para leitura, quanto para gravação. Recentemente foram feitas melhorias no projeto desse tipo de cartão, onde os tipo M possuem maior capacidade de armazenamento e os tipo H possuem maior velocidade de acesso aos dados.



Figura 3.4 – Cartões xD Fuji e Olympus

### 4 – MMC e SD

Os cartões desses dois padrões são muito semelhantes, seus principais diferenciais são a espessura do cartão e a presença de dois contatos



Figura 3.5 – Cartão microSD com adaptadores para miniSD e SD

elétricos extras no SD. São padrões muito populares, sua especificação é vendida a uma taxa acessível o que torna possível serem fabricados por uma ampla gama de indústrias e cada vez mais pesquisados em termos de melhorias. A capacidade desse tipo de memória costuma ser bem elevada e a velocidade de acesso agrada bastante a seus usuários. É um padrão de mídia bastante adotado por fabricantes de câmeras, pen-drives de chip, e outros periféricos.

Duas variações desses cartões são bastante populares entre os dispositivos de tamanho menor: mini-SD e micro-SD. Esses cartões são bastante reduzidos e contam com as mesmas características e funcionalidades dos cartões SD em tamanho natural. A maioria dos fabricantes desse tipo de cartão fornece-o junto com estes adaptadores para que os minis e micros possam ser usados como cartões SD comuns.

### 5 Memory stick



Figura 3.6 – Cartões Memory Stick de diversos fabricantes

Os cartões de memória Memory Stick, suas versões e miniaturizações foram lançados para competir com o padrão SD e MMC, porém encontraram no mercado apenas o fabricante Sony como seu grande usuário em nível de projetos. Então os cartões dessa categoria estão, no momento, restritos aos produtos Sony e tendem a ser descontinuados, caso este fabricante decida adotar outro padrão de mídia para seus produtos.

### 3.4 Memória RAM

Essa é a memória considerada principal em qualquer sistema de computação. De tão importante, chega a tornar algumas arquiteturas incompatíveis entre si. Devido à forma de organização de memória, alguns

computadores lêem o byte a partir do bit de mais alta ordem, outros lêem a partir do bit de mais baixa ordem. Isso significa, na prática, como se um sistema lesse o byte da esquerda para a direita e o outro pelo caminho inverso, o que torna claro que esses dois tipos de sistemas não conseguem se entender.



<http://www.im.e.usp.br/~wesley/memoria.htm>

A memória RAM do computador também é conhecida como RAM Dinâmica, pois seu funcionamento é baseado em flip-flops, que como já conhecemos, precisam de realimentação constante. Essa necessidade desse tipo de circuito dá a característica dinâmica desse tipo de memória. Nos sistemas atuais também é comum haver uma via de acesso privativa entre a UCP e a RAM, conhecida como barramento de memória ou barramento local. Os barramentos serão estudados mais adiante neste material.

A RAM é volátil, ou seja, seu conteúdo é guardado enquanto o computador está alimentado, ao desligar a corrente elétrica o que está na RAM é automaticamente apagado.

#### 3.4.1 organização da memória

Para organizar melhor os bits, as memórias são estruturadas e divididas em conjuntos ordenados de bits, denominados células, cada uma podendo armazenar uma parte da informação. Se uma célula consiste em  $k$  bits ela pode conter uma em  $2^k$  diferente combinação de bits, sendo que todas as células possuem a mesma quantidade de bits.

Cada célula está associada a um número que é seu endereço. Só assim torna-se possível a busca na memória exatamente do que se estiver querendo a cada momento (acesso aleatório). Sendo assim, célula pode ser definida como a menor parte de memória endereçável.

Se uma memória tem  $n$  células o sistema de endereçamento numera as células sequencialmente a partir de zero até  $n-1$ , sendo que esses en-

dereços são fixos e representados por números binários. A quantidade de bits em um endereço está relacionado à máxima quantidade de células endereçáveis. Por exemplo, se um endereço possui  $m$  bits o número máximo de células diretamente endereçáveis é  $2^m$ .

A maioria dos fabricantes de computador padronizaram o tamanho da célula em 8 bits (1 Byte). Bytes são agrupados em palavras, ou seja, a um grupo de bytes (2,4,6,8 Bytes) é associado um endereço particular. O significado de uma palavra é que a maioria das instruções operam em palavras inteiras.

Algumas arquiteturas como os PCs organizam as células de memória em segmentos e offsets. Esse padrão ajuda a ter maiores possibilidades de instalação e uso de mais espaço de memória.

### 3.4.2 funcionamento da memória principal

Duas operações básicas são permitidas no uso da memória: escrita e leitura.

Em se tratando de Memória Principal (MP), essas operações são realizadas pela UCP operando nas células, não sendo possível trabalhar com parte dela.

A leitura não é uma operação que consiste em copiar a informação contida em uma célula da MP para a UCP, através de um comando desta.

A escrita é uma operação destrutiva, por que toda vez que se grava uma informação em uma célula da MP, o seu conteúdo anterior é eliminado.

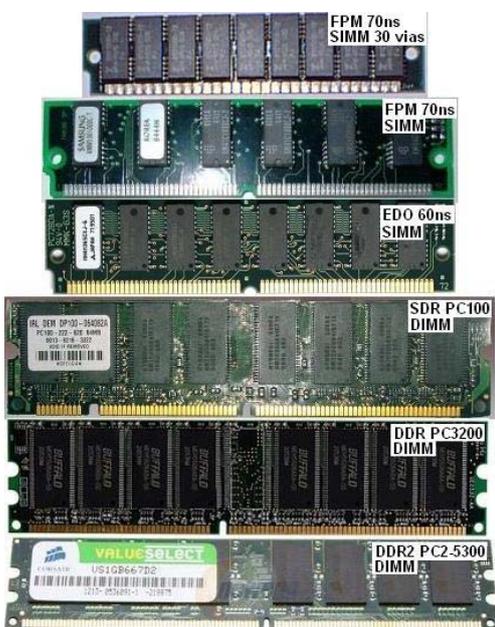


Figura 3.7 – Evolução dos pentes de memória RAM dos PCs

### 3.4.3 classificação das memórias atuais

Os PCs contam com muitas opções de padrão de memória RAM, isso se deve ao fato de sempre se buscar uma memória de maior capacidade

de, mais velocidade de acesso e menor tempo de realimentação (refresh). Esse tempo de realimentação é normalmente medido em nanossegundos ou  $10^{-9}$  do segundo. A linha evolutiva passa pelas memórias de 80ns, seguidas pelas *fast-page* de 70ns, memórias EDO de 60ns muito comuns na época do lançamento do Pentium. Hoje já se trabalha com tempos abaixo da casa dos 12ns nas atuais DDR e DDR-2 (DDR significa Double Data Rate, ou seja, memórias com o dobro da vazão de dados das suas antecessoras).

### 3.5 Detecção e Correção de Erros

Em se tratando de relevância, todo cuidado é pouco com a preservação da informação armazenada. Uma simples transferência do conteúdo de uma célula de memória da RAM para a UCP pode resultar em deturpação dos valores dos bits e uma consequente adulteração da informação armazenada. Essa preocupação é bastante presente nas cabeças dos Cientistas da Computação no mundo todo há muito tempo. Uma das descobertas mais respeitáveis nessa área aconteceu com HAMMING na década de 50 do século passado.



Visita obrigatória para entender o código de Hamming:

[http://www.di.ubi.pt/cursos/mestrados/mei/disciplinas/5052/fichs/Extra\\_Topico6.pdf](http://www.di.ubi.pt/cursos/mestrados/mei/disciplinas/5052/fichs/Extra_Topico6.pdf)

<http://foobox.org/files/uevora/TI/ti-slides-10.pdf>

A idéia é inserir bits extras no byte de informação. Estes bits extras são obtidos a partir de XOR entre alguns bits que compõem o byte original. Quanto mais bits são inseridos na palavra original, maior será a chamada distância de Hamming. Quanto maior for a distância de Hamming, mais fácil será detectar que determinado bit foi trocado, ou melhor ainda, será possível reverter essa troca e assim corrigir o bit errado sem haver necessidade de retransmissão da palavra.

Aos sistemas que implementam o código de Hamming dá-se o nome de sistemas com paridade. Isso explica por que as memórias com paridade são mais caras que as memórias sem paridade. Sendo assim, na próxima

CACHE – lê-se *quesh* ou ainda cachê como na língua francesa.

expansão de memória de computador, considere a possibilidade de adquirir memórias com paridade e trabalhar mais tranqüilo.

### 3.6 Memória CACHE

Esse tipo de memória é conhecida como estática, pois não depende de *flip-flops* e nem tem *refresh*. São memórias fabricadas com capacitores, sua composição é mais cara que a RAM, porém como não tem realimentação, os dados estão sempre disponíveis, a UCP nunca espera para acessar um dado nesse tipo de memória. Devido a sua composição ser mais cara, esta categoria de memória não substitui a RAM dinâmica, mas é possível usar um pouco dessa memória para agilizar o trabalho da UCP com os acessos a RAM.



<http://www.clubedohardwre.com.br/artigos/1410/1>

Como a CACHE é vantajosa em termos de velocidade de acesso, um pouco dela é inserido no sistema, seja na placa-mãe e, em alguns casos, até mesmo dentro da pastilha do microprocessador. Existem dois níveis de cache dentro de um computador. O nível L1 fica dentro do processador e encarrega-se de agilizar a execução do microcódigo, que é um programa que fica executando dentro da UCP, fazendo-a reconhecer as instruções dos programas que usamos. O outro nível de cache é conhecido por L2. Pode vir dentro da pastilha do microprocessador e ser complementado por mais alguns circuitos integrados da placa-mãe. A função deste tipo de cache é tornar mais rápido o resgate e a gravação de informações de programas do usuário na memória RAM.

Mas não se pode inserir memória cache em quantidade aleatória para ganhar desempenho. Existe um limite prático para cada arquitetura. Hoje tem-se 1 ou 2 gigabytes de RAM e usa-se normalmente 1 megabyte de cache. Estudos mostram que não adianta fazer a cache tão grande, pois o ganho de desempenho torna-se imperceptível com o aumento desse tipo de memória. No padrão atual, estima-se que mais que 1MB de cache seria desperdício de dinheiro e de memória.

## Memória principal

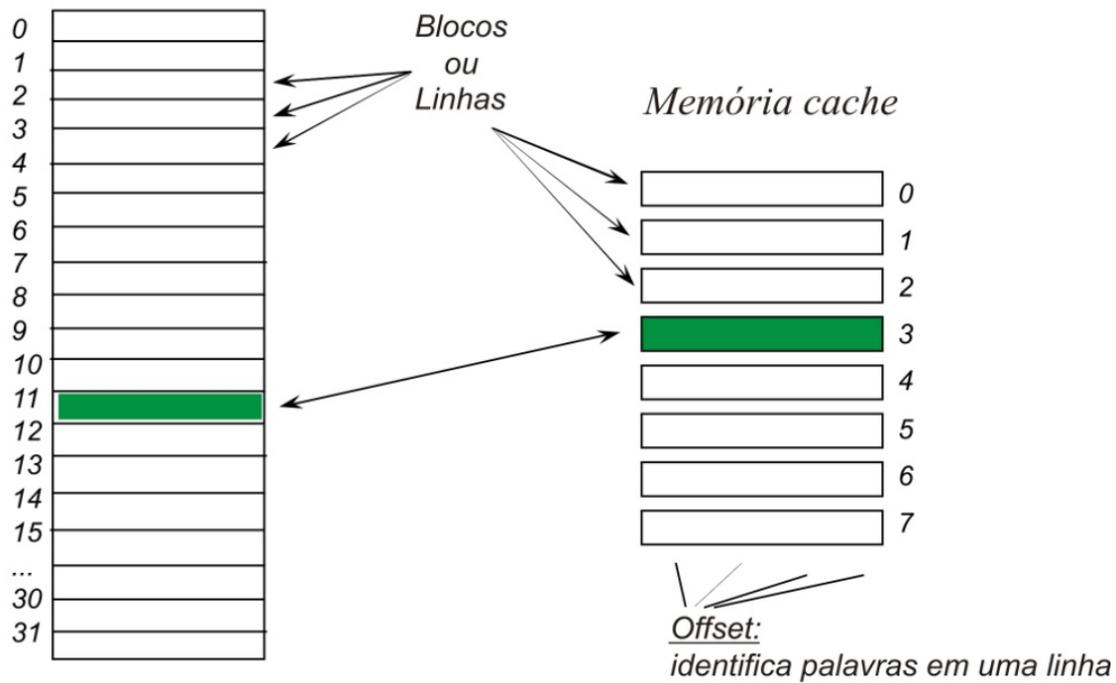


Figura 3.8 – Organização de RAM e cachê

Como fazer para que vários megabytes ou mesmo gigabytes de memória caibam em apenas 1MB de cache?

### 3.6.1 Mapeamentos de RAM em Cachê

Os algoritmos de mapeamentos de memória RAM em CACHE são basicamente três: direto, associativo e associativo por conjunto. Todos são propostas de como fazer vários MB de RAM caberem em no máximo um MB de cache.

#### MAPEAMENTO DIRETO

Uma determinada linha da memória principal é sempre mapeada em determinada linha de cache obedecendo a uma fórmula. Isso tem como grande desvantagem o fato de um programa poder usar duas variáveis

armazenadas em endereços de RAM que mapeiem coincidentemente no mesmo local de cache. Nesse caso, a memória cache vai tender a atrapalhar o desempenho do sistema em vez de ajudar, tendo em vista que a cada troca de variável exista também uma troca de dados na cache.

Fórmula de mapeamento

$$i = j \text{ modulo } m$$

onde:

*i* = linha de cache

*j* = número do bloco da memória principal

*m* = número de linhas na cache

## MAPEAMENTO ASSOCIATIVO

Nessa técnica, os blocos da memória principal podem ser carregados dentro de qualquer linha de cache. E para cada linha de memória principal, fica associado seu endereço de origem conhecido como TAG.

Teoricamente, cada dado vindo da RAM fica “estacionado” na primeira vaga que estiver livre e será removido algum dado da cache somente quando esta estiver lotada.

Essa técnica tem como principal desvantagem o fato de armazenar dados na cache sem uma lógica de ocupação, o que leva a UCP a pesquisar a localização de cada dado requerido na cache antes de mandá-lo vir da RAM.

## MAPEAMENTO ASSOCIATIVO POR CONJUNTO

Essa técnica une o que há de melhor nas duas anteriores. Existe uma função de mapeamento tal qual no caso do mapeamento direto e a cache é organizada de forma multidimensional, o que gera diversas “vagas” no mesmo endereço de mapeamento.

Isso ajuda na hora de armazenar os dados de maneira organizada e também na hora de buscar esses dados.

Equação de mapeamento

$$m = v * k$$

$$i = j \text{ modulo } v$$

onde:

v = conjuntos da cache

k = número de linha de cada conjunto

i = número do conjunto da cache

j = número do bloco da memória principal

m = número de linhas na cache

### 3.6.2 Políticas de Substituição

Quando a cache está cheia e precisa ter seus dados substituídos uma atitude deve ser tomada: escolher quem sai da cache e quem permanece. Nesse momento faz-se necessário usar um dos métodos de substituição de páginas da cache:

#### **Random**

A substituição é feita sem critério de seleção definido. Os blocos são escolhidos aleatoriamente. Isso pode levar a sérios problemas de desempenho, pois a vítima escolhida pode ser aquele dado crucial para o funcionamento do programa.

#### **LRU (Least Recently Used)**

Menor taxa de faltas - substitui aquele bloco que tem estado na cache por mais tempo sem ser usado pelo programa.

### FIFO(First in First out)

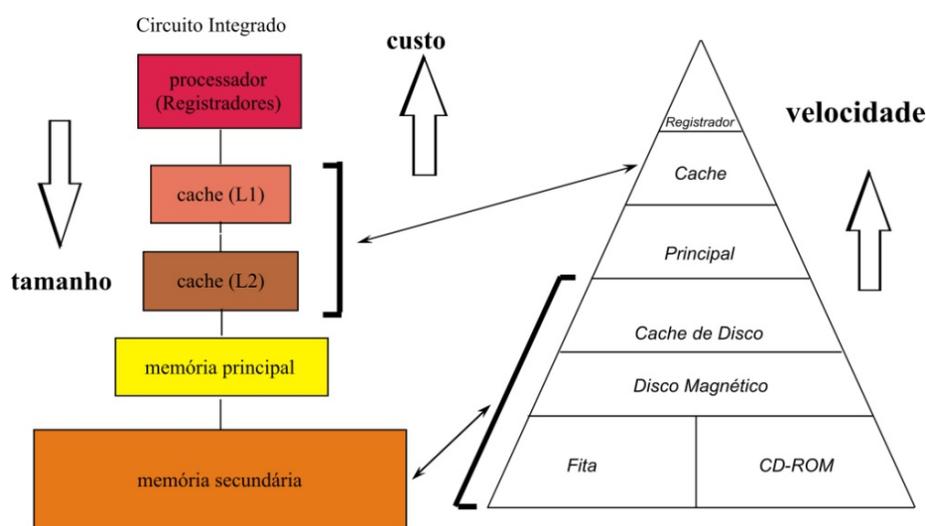
Substitui aquele bloco no conjunto que tem estado na cache por mais tempo. É o critério de fila estudado em Estruturas de Dados. Porém esse método pode não ser o mais justo, tendo em vista que o dado pode estar há muito tempo na cache e ser também aquele mais acessado pelo programa.

### LFU(Least Frequently Used)

Substitui aquele bloco dentro do conjunto que tem sido menos referenciado na cache. LFU pode ser implementado associando-se um contador a cada slot da cache. Além de ser um dos métodos mais justos de substituição, juntamente com o LRU.

#### 3.6.3 Como a cache trata as escritas

Quando a UCP busca um dado na cache e altera-o uma atitude deve ser tomada para garantir que este dado novo seja escrito em RAM, mas escrever na RAM pode degradar o desempenho do sistema. Portanto os sistemas deixam duas opções de configuração de escrita de cache. Normalmente essas opções estão disponíveis no programa da BIOS de seu PC, conhecido como SETUP. Eis as opções com os respectivos efeitos:



**Write through** - Essa técnica faz com que toda operação de escrita na memória principal seja feita também na cache, assegurando que os dados na memória principal são sempre válidos. A principal desvantagem desse método é o acréscimo no tráfego de memória que pode gerar engarrafamento, além de degradar o desempenho geral do sistema.

**Write back** - Essa técnica reduz escrita na memória. Atualizações ocorrem apenas na cache. Quando uma atualização ocorre, um bit UPDA-

TE associado com o slot de memória é ligado. Quando um bloco é substituído, e este bit está ligado, o conteúdo da cache é gravado de volta na memória principal. O problema é que com a escrita de volta (“write back”) na memória principal, parte da memória principal continuará desatualizada até que haja uma nova atualização da cache. Isso pode complicar operações de entrada e saída que sejam feitas diretamente na RAM, conhecidas como operações de DMA (Acesso Direto a Memória).

### 3.7 Hierarquia de Memórias

A figura a seguir representa um resumo de todas as memórias do computador com suas principais características que possam gerar comparações. Veja que os registradores presentes dentro do microprocessador são as menores e mais rápidas memórias, porém são as mais caras. Na outra ponta do gráfico estão as memórias de armazenamento em massa como fitas, discos e outras mídias de armazenamento secundário.



#### EXERCÍCIOS

---

1. Pesquise na Internet sobre cálculo do tamanho da memória a partir de suas células e suas características e responda o que se pede: Considere que uma memória tem um espaço de endereçamento máximo de 4K e cada célula de memória pode armazenar 8 bits. Determine:
  - a) Qual é o valor total de bits que podem ser armazenado nessa memória?
  - b) Qual é o tamanho de cada endereço?
2. Qual é a diferença construtiva entre uma Memória DRAM e SRAM? Qual é a mais rápida para acesso? Descreva por que em uma Memória Principal a quantidade de Memórias DRAM é bem maior que as SRAM?

3. Um pente de memória RAM tem a capacidade de 128Mbytes. Cada Célula de Memória armazena 2 Bytes. Pergunta-se:
  - a) Qual será o tamanho do endereço do sistema do computador?
  - b) Qual é o total de células disponíveis para uso nessa memória?
4. Como operam os sistemas que usam código de Hamming para detectar e corrigir erros de transmissão?
5. Diferencie as principais formas de memória Flash existentes no mercado.
6. Os cartões de memória SD e MMC são idênticos? Justifique sua resposta.
7. Classifique os principais tipos de memória ROM.
8. Conceitue memória CACHE e fale de sua atuação no desempenho do computador.
9. Explique os diferentes algoritmos de substituição de páginas de cache.
10. Diferencie cache *write-through* e *write-back*.

http:

#### WEB-BIBLIOGRAFIA

---

<http://www.ime.usp.br/~weslley/memoria.htm>

<http://www.guiadohardware.net/tutoriais/memoria-flash/>

[http://wnews.uol.com.br/site/noticias/materia\\_especial.php?id\\_secao=17&id\\_conteudo=227](http://wnews.uol.com.br/site/noticias/materia_especial.php?id_secao=17&id_conteudo=227)

[http://www.di.ubi.pt/cursos/mestrados/mei/disciplinas/5052/fichs/Extra\\_Topico6.pdf](http://www.di.ubi.pt/cursos/mestrados/mei/disciplinas/5052/fichs/Extra_Topico6.pdf)

<http://foobox.org/files/uevora/TI/ti-slides-10.pdf>

<http://www.clubedohardware.com.br/artigos/1410/1>

# Unidade 4

## Barramentos e Interfaces

### **Resumo**

Esta unidade destina-se a apresentar as vias de comunicação existentes dentro do computador, estas vias são conhecidas como barramentos, quando estes barramentos tem saídas para o meio externo a placa-mãe ganham o nome de interfaces, que devem seguir um padrão bem definido para que os periféricos conectados sejam capazes de se comunicar com a UCP.

## Sumário da Unidade

<b>UNIDADE IV – BARRAMENTOS E INTERFACES</b>	
4.1 – Barramentos – Conceitos Gerais .....	55
4.2 – Barramentos Comerciais .....	62
4.3 – Interfaces – Barramentos Externos .....	68

## 4.1 Barramentos – Conceitos Gerais

Os barramentos, conhecidos como BUS em inglês, são conjuntos de fios que normalmente estão presentes em todas as placas do computador. Na verdade existe barramento em todas as placas de produtos eletrônicos, porém em outros aparelhos os técnicos referem-se aos barramentos simplesmente como o “impresso da placa”.

Barramento é um conjunto de 50 a 100 fios que fazem a comunicação entre todos os dispositivos do computador: UCP, memória, dispositivos de entrada e saída e outros. Os sinais típicos encontrados no barramento são: dados, clock, endereços e controle.

Os dados trafegam por motivos claros de necessidade de serem levados às mais diversas porções do computador.

Os endereços estão presentes para indicar a localização para onde os dados vão ou vêm.

O clock trafega nos barramentos conhecidos como síncronos, pois os dispositivos são obrigados a seguir uma sincronia de tempo para se comunicarem.

O controle existe para informar aos dispositivos envolvidos na transmissão do barramento se a operação em curso é de escrita, leitura, reset ou outra qualquer. Alguns sinais de controle são bastante comuns:

- **Memory Write** - Causa a escrita de dados do barramento de dados no endereço especificado no barramento de endereços.
- **Memory Read** - Causa dados de um dado endereço especificado pelo barramento de endereço a ser posto no barramento de dados.
- **I/O Write** - Causa dados no barramento de dados serem enviados para uma porta de saída (dispositivo de I/O).
- **I/O Read** - Causa a leitura de dados de um dispositivo de I/O, os quais serão colocados no barramento de dados.
- **Bus request** - Indica que um módulo pede controle do barramento do sistema.
- **Reset** - Inicializa todos os módulos

Todo barramento é implementado seguindo um conjunto de regras de comunicação entre dispositivos conhecido como BUS STANDARD, ou simplesmente PROTOCOLO DE BARRAMENTO, que vem a ser um padrão que qualquer dispositivo que queira ser compatível com este barramento deva compreender e respeitar. Mas um ponto sempre é certeza: todo dispositivo deve ser único no acesso ao barramento, porque os dados trafegam por toda a extensão da placa-mãe ou de qualquer outra placa e uma mistura de dados seria o caos para o funcionamento do computador.

Os barramentos têm como principais vantagens o fato de ser o mesmo conjunto de fios que é usado para todos os periféricos, o que barateia o projeto do computador. Outro ponto positivo é a versatilidade, tendo em vista que toda placa sempre tem alguns slots livres para a conexão de novas placas que expandem as possibilidades do sistema.

A grande desvantagem dessa idéia é o surgimento de engarrafamentos pelo uso da mesma via por muitos periféricos, o que vem a prejudicar a vazão de dados (throughput).

#### 4.1.1 Dispositivos conectados ao barramento

- **Ativos ou Mestres** - dispositivos que comandam o acesso ao barramento para leitura ou escrita de dados
- **Passivos ou Escravos** - dispositivos que simplesmente obedecem à requisição do mestre.

Exemplo:

- CPU ordena que o controlador de disco leia ou escreva um bloco de dados.

A CPU é o mestre e o controlador de disco é o escravo.

#### 4.1.2 Classificação quanto à temporização

##### Barramentos Síncronos

Esse tipo de barramento exige que todo fluxo de dados aconteça em sincronia com uma base de tempo conhecida como clock do sistema.

Vejamos uma ilustração que

esclarece o funcionamento dessa categoria de barramentos

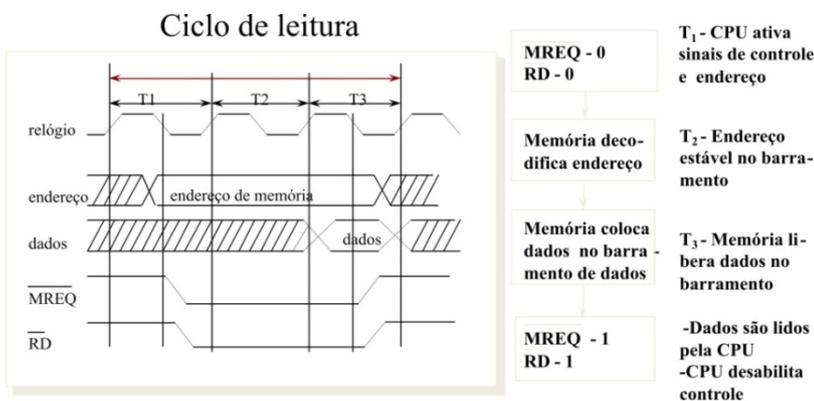


Figura 4.2 Barramento Síncrono

## Barramentos Assíncronos

Essa categoria de barramentos não segue um relógio mestre para realizar suas operações. Os ciclos de leituras e escritas podem ter durações diferenciadas de acordo com as necessidades de cada operação.

A seguir uma figura que ilustra o funcionamento dessa categoria de barramentos.

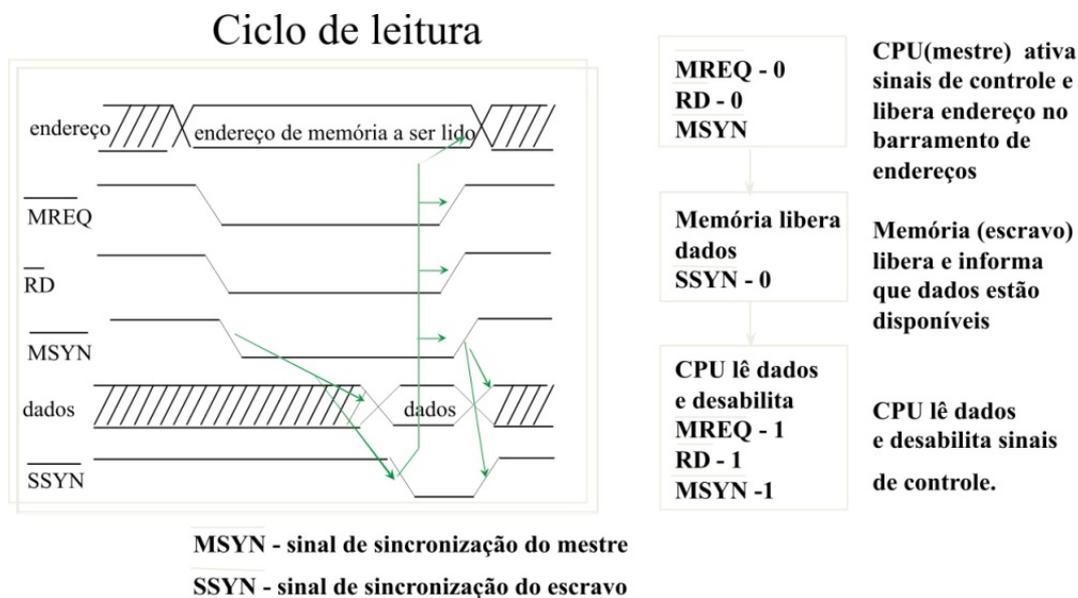


Figura 4.2 Barramento assíncrono

### 4.1.3 Arbitragem de barramento

Conforme abordado anteriormente, cada transferência de dados deve ser única no barramento, pois os dados dos diversos dispositivos não devem ser misturados. Mas o que deve acontecer caso mais de um dispositivo tente usar o barramento ao mesmo tempo?

Deve haver um mecanismo de arbitragem do uso dos barramentos, seja com o árbitro centralizado e bem definido, ou seja, com o árbitro descentralizado.

## Arbitragem centralizada

Nesse tipo de arbitragem o dispositivo conhecido como árbitro libera ou não a permissão de uso do barramento, isso cria uma ordem e uma disciplina de acesso ao meio.

Características desse tipo de arbitragem:

1. Todos os dispositivos são ligados em série, assim a permissão, dada pelo árbitro, pode ou não se propagar através da cadeia.
2. Cada dispositivo deve solicitar acesso ao barramento.
3. O dispositivo mais próximo do árbitro tem maior prioridade.
4. O dispositivo que receber a permissão bloqueia os outros dispositivos.

Veja a ilustração a seguir:

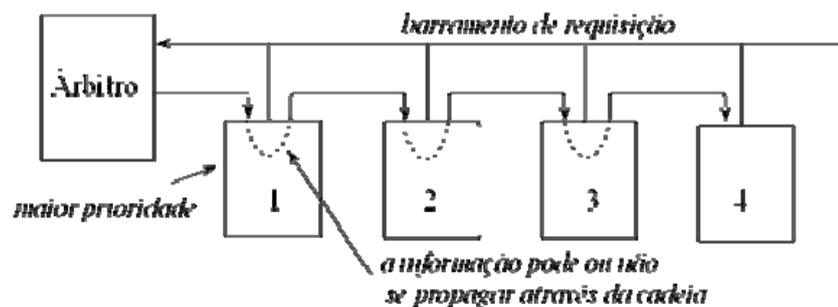


Figura 4.3 Arbitragem centralizada

## Arbitragem descentralizada

Esse tipo de arbitragem dispensa a figura do árbitro, mas todos os dispositivos devem respeitar um conjunto rígido de regras de acesso ao meio. A seguir uma figura ilustra este tipo de arbitragem de barramento e serve de base para o entendimento das regras.

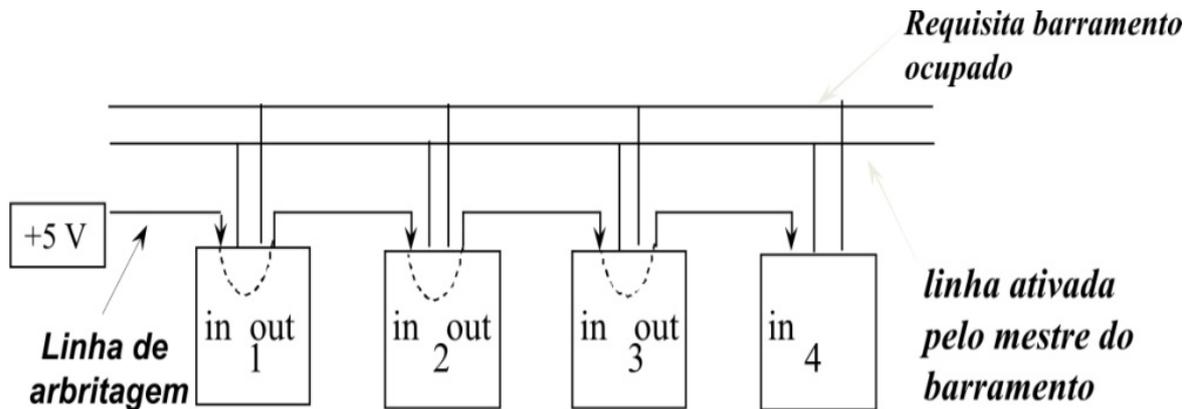


Figura 4.4 Arbitragem descentralizada

Regras da arbitragem descentralizada:

1. Quando nenhum dispositivo quer barramento, a linha de arbitragem ativada é propagada através de todos os dispositivos.
2. Para se obter o barramento, o dispositivo primeiro verifica se o barramento está disponível, e se a linha de arbitragem que está recebendo, in, está ativada.
3. Se in estiver desativada, ela não poderá tornar-se mestre do barramento.
4. Se in estiver ativada, o dispositivo requisita o barramento, desativa out, o que faz com que todos os seguintes na cadeia desativem in e out.

#### 4.1.4 Tipos de barramentos

##### **Dedicado**

Cada elemento do barramento é dedicado exclusivamente ou a uma função ou a um subconjunto de componentes do computador.

Exemplo: barramento de memória → liga a UCP à memória RAM.

## **Multiplexado**

Nesse tipo de barramento sinais podem ser multiplexados no tempo para comportar diferentes funções.

Exemplo: endereços e dados podem trafegar no mesmo barramento mediante o controle de “Address Valid Control Line”, ou seja, sob o controle de um sinal que especifica quais sinais são válidos em determinado período de tempo.

### 4.1.5 Barramentos de memória x Barramentos de E/S

#### **Barramentos de memória**

São barramentos de alta velocidade e especiais

Características:

- São pequenos
- Operam em alta velocidade
- São em geral conectados diretamente a CPU para maximizar a largura de banda entre memória e CPU (bandwidth)
- Tipos de dispositivos são conhecidos

#### **Barramentos de Entrada e Saída**

São, em geral, barramentos de ordem geral, sem que haja explicitamente definição dos dispositivos a serem conectados a ele.

Características:

- Podem ser longos.
- Podem ter diferentes tipos de dispositivos conectados a ele.

- Tem faixa de largura na banda de dados dos dispositivos conectados a eles.
- Normalmente seguem um padrão.

A figura a seguir ilustra um barramento geral de um computador:

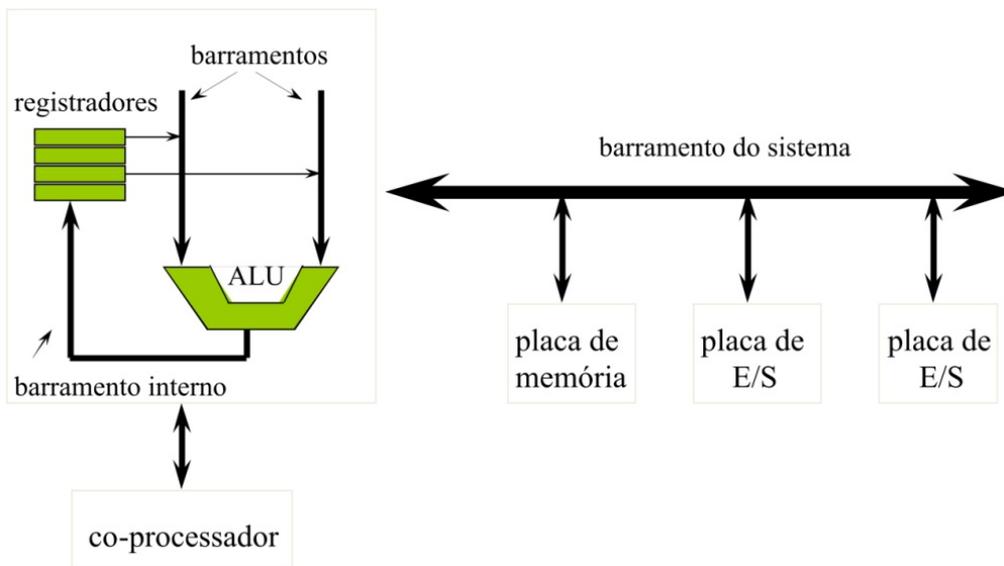


Figura 4.5 Barramento Geral

## 4.2 Barramentos Comerciais

Serão listados aqui alguns barramentos que foram e alguns que ainda são bastante usados comercialmente.

### 4.2.1 ISA – Industry Standard Architecture

Foi lançado em 1984 pela IBM para suportar o novo PC-AT. Tornou-se, de imediato, o padrão de todos os PC-compatíveis. Era um barramento único para todos os componentes do computador, operando com largura de 16 bits e com clock de 8 MHz.



<http://www.icea.gov.br/ead/anexo/24101.htm>

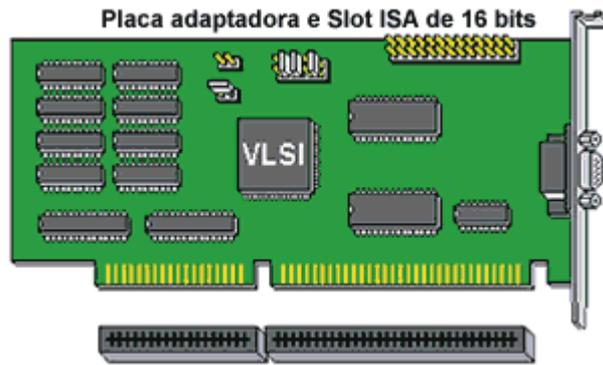


Figura 4.6 Barramento ISA 16 bits e seu slot

#### 4.2.2 MCA – Microchannel Architecture

Foi desenvolvido pela IBM, por volta de 1987, para melhorar o desempenho do ISA com os novos processadores 386 e preparando o lançamento dos 486. Projeto proprietário, ou seja, a IBM registrou esse barramento de forma que não pudesse ser usado nos clones de IBM. Foi restrito à linha PS/2. Esse barramento não é compatível com nenhum outro existente no mercado, isso rendeu a fama de que até hoje os PCs da IBM têm de aceitar somente peças originais fabricadas pela própria. Isso não é bem verdade, pois é válido somente para os PC da linha PS/2 com processadores 386 ou 486.

Operava com largura de 32 bits e com frequência de 10MHz, isso conferia um bom desempenho para os sistemas IBM. Devido ao isolamento com os demais fabricantes de placas, a IBM abandonou esse barramento ainda na primeira metade da década de 90.

#### 4.2.3 EISA – Enhanced ISA

Esse barramento foi desenvolvido em 1987 pelos fabricantes de clones de IBM, como resposta ao projeto do MCA, como é uma expansão

do projeto original do ISA para operar com 32 bits e com mesmo clock, esse barramento é 100% compatível com seu antecessor. O sucesso de mercado na época foi garantido para as placas de alto desempenho.

#### 4.2.4 VESA Local Bus

Barramento que inaugura o conceito de local bus nos PCs. Com o advento da CPU 486, a idéia de se usar dispositivos de E/S a 8 ou 10 MHz colidiu com a alta freqüência dos chips da placa-mãe, 33MHz.

VESA Local Bus foi a primeira solução para esse gargalo, sendo utilizada por máquinas desktop para suportar placas controladoras de vídeo de alta velocidade e mais um outro periférico de alta velocidade. O termo Local refere-se às linhas de barramento usadas pelo



Figura 4.7 Placa-mãe compatível com VESA Local Bus (marrom)

processador. Esse tipo de barramento tem acesso direto ao processador e trabalha na mesma velocidade do processador.

Fisicamente, as placas-mãe passaram a ter conectores extras em alguns slots para o encaixe destas placas, que também eram fisicamente mais compridas que as placas ISA, devido ao conector para o barramento local. Confira nas figuras a seguir.

#### Características

- Barramento conectado diretamente a CPU (microprocessador).
- 32 bits no barramento de dados.
- Suporta apenas 2 cartões a 33 MHz (50MHz).

- Expansões devem ser feitas via barramento ISA ou EISA.

#### 4.2.5 PCI – Peripheral Components Interconnect

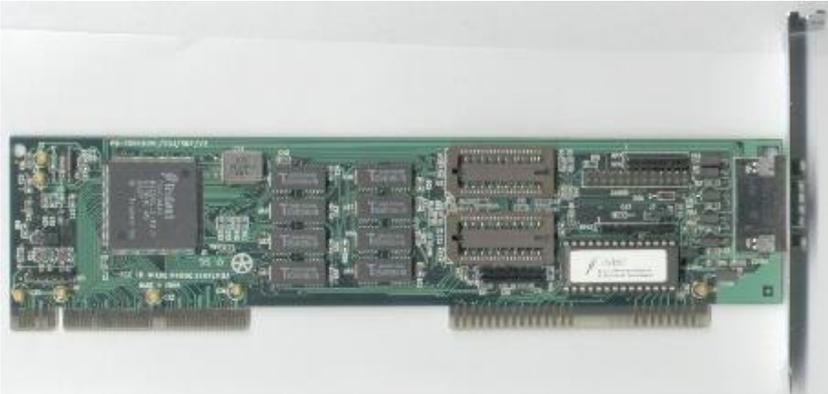


Figura 4.8 Placa de vídeo VESA Local Bus (VL BUS)

PCI é um barramento síncrono de alta performance, indicado como mecanismo entre controladores altamente integra-dos, plug-in placas, sistemas de processadores/memória. Foi o pri-meiro barramento a incorporar o conceito plug-and-play. Seu lança-mento foi em 1993, em conjunto com o processador PENTIUM™ da Intel. Assim o novo

processador realmente foi revolucionário pois chegou com uma série de inovações e um novo barramento. O PCI foi definido com o objetivo primário de estabelecer um padrão da indústria e uma arquitetura de barramento que ofereça baixo custo e permita diferenciações na implementação.

#### **Componente PCI ou PCI master**

Funciona como uma ponte entre processador e barramento PCI, no qual dispositivos add-in com interface PCI estão conectados.

#### **Add-in cards interface**

Possuem dispositivos que usam o protocolo PCI. São gerenciados pelo PCI master e são totalmente programáveis.

#### 4.2.6 AGP – Advanced Graphics Port

Esse barramento permite que uma placa controladora gráfica AGP substitua a placa gráfica no barramento PCI. O Chip controlador AGP substitui o controlador de E/S do barramento PCI. O novo conjunto AGP continua com funções herdadas do PCI. O conjunto faz a transferência de dados entre memória, o processador e o controlador ISA, tudo, simultaneamente.

Permite acesso direto mais rápido à memória. Pela porta gráfica aceleradora, a placa tem acesso direto à RAM, eliminando a necessidade de uma VRAM (vídeo RAM) na própria placa para armazenar grandes arquivos de bits como mapas e textura.

O uso desse barramento iniciou-se através de placas-mãe que usavam o chipset i440LX, da Intel, já que esse chipset foi o primeiro a ter suporte ao AGP. A principal vantagem desse barramento é o uso de uma maior quantidade de memória para armazenamento de texturas para objetos tridimensionais, além da alta velocidade no acesso a essas texturas para aplicação na tela.

O primeiro AGP (1X) trabalhava a 133 MHz, o que proporciona uma velocidade 4 vezes maior que o PCI. Além disso, sua taxa de transferência chegava a 266 MB por segundo quando operando no esquema de velocidade X1, e a 532 MB quando no esquema de velocidade 2X. Existem também as versões 4X, 8X e 16X. Geralmente, só se encontra um único slot nas placas-mãe, visto que o AGP só interessa às placas de vídeo.

#### 4.2.7 PCI Express

Na busca de uma solução para algumas limitações dos barramentos AGP e PCI, a indústria de tecnologia trabalha no barramento PCI Express, cujo nome inicial era 3GIO. Trata-se de um padrão que proporciona altas

taxas de transferência de dados entre o computador em si e um dispositivo, por exemplo, entre a placa-mãe e uma placa de vídeo 3D.

A tecnologia PCI Express conta com um recurso que permite o uso de uma ou mais conexões seriais, também chamados de *lanes* para transferência de dados. Se um determinado dispositivo usa um caminho, então diz-se que esse utiliza o barramento PCI Express 1X; se utiliza 4 *lanes*, sua denominação é PCI Express 4X e assim por diante. Cada *lane* pode ser bidirecional, ou seja, recebe e envia dados. Cada conexão usada no PCI Express trabalha com 8 bits por vez, sendo 4 em cada direção. A frequência usada é de 2,5 GHz, mas esse valor pode variar. Assim sendo, o PCI Express 1X consegue trabalhar com taxas de 250 MB por segundo, um valor bem maior que os 132 MB do padrão PCI. Esse barramento trabalha com até 16X, o equivalente a 4000 MB por segundo. A tabela abaixo mostra os valores das taxas do PCI Express comparadas às taxas do padrão AGP:

AGP 1X: 266 MBps	PCI Express 1X: 250 MBps
AGP 4X: 1064 MBps	PCI Express 2X: 500 MBps
AGP 8X: 2128 MBps	PCI Express 8X: 2000 MBps
	PCI Express 16X: 4000 MBps

É importante frisar que o padrão 1X foi pouco utilizado e, devido a isso, há empresas que chamam o PCI Express 2X de PCI Express 1X. Assim sendo, o padrão PCI Express 1X pode representar também taxas de transferência de dados de 500 MB por segundo.

A Intel é uma das grandes precursoras de inovações tecnológicas. No início de 2001, em um evento próprio, a empresa mostrou a necessidade de criação de uma tecnologia capaz de substituir o padrão PCI: tratava-se do 3GIO (Third Generation I/O – 3ª geração de Entrada e

Saída). Em agosto desse mesmo ano, um grupo de empresas chamado de PCI-SIG (composto por companhias como IBM, AMD e Microsoft) aprovou as primeiras especificações do 3GIO.

Entre os quesitos levantados nessas especificações, estão os que se seguem: suporte ao barramento PCI, possibilidade de uso de mais de uma lane, suporte a outros tipos de conexão de plataformas, melhor gerenciamento de energia, melhor proteção contra erros, entre outros. Esse barramento é fortemente voltado para uso em subsistemas de vídeo.

### 4.3 Interfaces – Barramentos Externos

Os barramentos circulam dentro do computador, cobrem toda a extensão da placa-mãe e servem para conectar as placas menores especializadas em determinadas tarefas do computador. Mas os dispositivos periféricos precisam comunicar-se com a UCP, para isso, historicamente foram desenvolvidas algumas soluções de conexão tais como: serial, paralela, USB e Firewire. Passando ainda por algumas soluções proprietárias, ou seja, que somente funcionavam com determinado periférico e de determinado fabricante.

#### 4.3.1 Interface Serial

Conhecida por seu uso em mouse e modems, esta interface no passado já conectou até

impressoras. Sua característica fundamental é que os bits

trafegam em fila, um por vez, isso torna a comunicação mais lenta, porém o cabo do dispositivo pode ser mais longo, alguns chegam até a 10 metros



Figura 4.9 Interfaces seriais DB-9 e BD-25 respectivamente

de comprimento. Isso é útil para usar uma barulhenta impressora matricial em uma sala separada daquela onde o trabalho acontece.

As velocidades de comunicação dessa interface variam de 25 bps até 57.700 bps (modems mais recentes). Na parte externa do gabinete, essas interfaces são representadas por conectores DB-9 ou DB-25 machos, conforme a figura a seguir.

#### 4.3.2 Interface Paralela



Figura 4.10 Interface paralela DB-25 fêmea

Criada para ser uma opção ágil em relação à serial, essa interface transmite um byte de cada vez. Devido aos 8 bits em paralelo existe um RISCO de interferência na corrente elétrica dos condutores que formam o cabo. Por esse motivo os cabos de comunicação desta interface são mais curtos, normalmente funcionam muito bem até a distância de 1,5 metro, embora exista no mercado cabos paralelos de até 3 metros de comprimento. A velocidade de transmissão desta porta chega até a 1,2 MB por segundo.



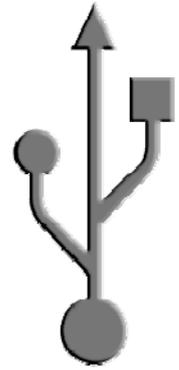
Figura 4.11 Conector Centronics no cabo de impressora

Nos gabinetes dos computadores essa porta é encontrada na forma de conectores DB-25 fêmeas. Nas impressoras, normalmente, os conectores paralelos são conhecidos como interface centronics. Veja as ilustrações.

#### 4.3.3 USB – Universal Serial Bus

O USB Implementers Forum (<http://www.usb.org>), que é o grupo de fabricantes que desenvolveu o barramento USB, já desenvolveu a segunda versão do USB, chamada USB 2.0 ou High-speed USB. Essa nova versão do USB possui uma taxa máxima de transferência de 480 Mbps

(aproximadamente 60 MB/s), ou seja, uma taxa maior que a do Firewire 1.0 e muito maior do que a versão anterior do USB, chamada 1.1, que permite a conexão de periféricos usando taxas de transferência de 12 Mbps (aproximadamente 1,5 MB/s) ou 1,5 Mbps (aproximadamente 192 KB/s), dependendo do periférico.



Saiba mais em:

<http://www.guiadohardware.net/tutoriais/usb-firewire-dvi/>

A porta USB 2.0 continua 100% compatível com periféricos USB 1.1. Ao iniciar a comunicação com um periférico, a porta tenta comunicar-se a **480 Mbps**. Caso não tenha êxito, ela abaixa a sua velocidade para **12 Mbps**. Caso a comunicação também não consiga ser efetuada, a velocidade é então abaixada para **1,5 Mbps**. Com isso, os usuários não devem se preocupar com os periféricos USB que já possuem: eles continuarão compatíveis com o novo padrão.

Os computadores com interfaces USB aceitam até 127 dispositivos conectados. Às vezes as placas têm de 2 a 6 conectores USB. Para resolver isso são vendidos os hubs USB. Um detalhe importantíssimo é que hubs USB 1.1 não conseguem estabelecer conexões a 480 Mbps para periféricos conectados a eles. Nesse caso, estes hubs atuam como gargalos de conexão. Sempre que puder escolher, dê preferência a dispositivos USB 2.0.

Outro fato interessante também é o padrão do cabo USB, mais precisamente de seus conectores. É fato que alguns fabricantes de

Tipo	Imagem da porta	Imagem do conector
Tipo A	4.5mm x 12.0mm 	
Tipo B	7.3mm x 8.5mm 	
Mini-A	3.0mm x 6.8mm 	
Mini-B	3.0mm x 6.8mm 	

câmeras e outros dispositivos podem tentar criar conectores proprietários para suas interfaces USB, sempre respeitando a ponta que se liga no computador (conector A). Mas a grande maioria dos fabricantes de dispositivos eletrônicos em geral, se usa USB, respeita o padrão de conectores apresentado na figura abaixo.

Portanto, o cabo daquela câmera que foi esquecido em uma viagem pode facilmente ser substituído agora, basta respeitar o tipo de conector usado no produto.

#### 4.3.4 Firewire



O barramento firewire, também conhecido como IEEE 1394 ou como i.Link, é um barramento de grande volume de transferência de dados entre computadores, periféricos e alguns produtos eletrônicos de consumo. Foi desenvolvido inicialmente pela Apple como um barramento serial de alta velocidade, mas eles estavam muito à frente da realidade, ainda mais com, na época, a alternativa do barramento USB que já possuía boa velocidade, era barato e rapidamente integrado no mercado. Com isso, a Apple, mesmo incluindo esse tipo de conexão/portas no Mac por algum tempo, a realidade "de fato", era a não existência de utilidade para elas devido à falta de periféricos para seu uso. Porém o desenvolvimento continuou, sendo focado principalmente pela área de vídeo, que poderia tirar grandes proveitos da maior velocidade que ele oferecia.

Figura 4.12 Tipos de conectores USB

Suas principais vantagens:

- São similares ao padrão USB;
- Conexões sem necessidade de desligamento/boot do micro (hot-plugable);

- Capacidade de conectar muitos dispositivos (até 63 por porta);
- Permite até 1023 barramentos conectados entre si;
- Transmite diferentes tipos de sinais digitais: vídeo, áudio, MIDI, comandos de controle de dispositivo, etc;
- Totalmente Digital (sem a necessidade de conversores analógico-digital, e portanto, mais seguro e rápido);

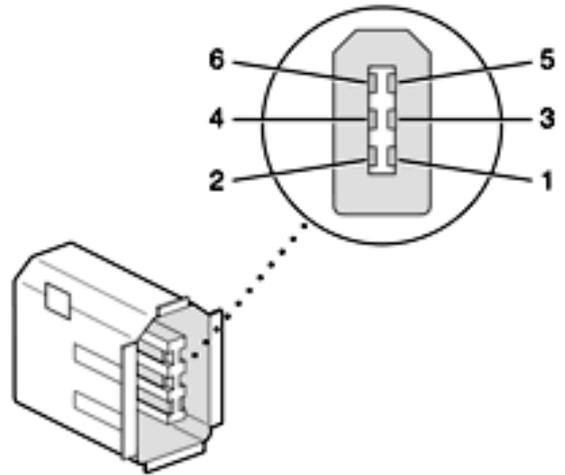


Figura 4.13 Conector FireWire

- Devido a ser digital, fisicamente é um cabo fino, flexível, barato e simples;
- Como é um barramento serial, permite conexão bem facilitada, ligando um dispositivo ao outro, sem a necessidade de conexão ao micro (somente uma ponta é conectada no micro).

A distância do cabo é limitada a 4.5 metros antes de haver distorções no sinal, porém, restringindo a velocidade do barramento podem-se alcançar maiores distâncias de cabo (até 14 metros). Lembrando que esses valores são para distâncias "ENTRE PERIFÉRICOS", e SEM A UTILIZAÇÃO DE TRANSCEIVERS (com transceivers a previsão é chegar a até 70 metros usando fibra ótica).

O barramento firewire permite a utilização de dispositivos de diferentes velocidades (100, 200, 400, 800, 1200 Mb/s) no mesmo barramento.

O suporte a esse barramento está nativamente em Macs, e em PCs através de placas de expansão específicas ou integradas com placas de captura de vídeo ou de som.

Os principais usos que estão sendo direcionados a essa interface, devido às características listadas, são na área de multimídia, especialmente na conexão de dispositivos de vídeo (placas de captura, câmeras, TVs digitais, setup boxes, home theater, etc).



## EXERCÍCIOS

---

1. Conceitue Barramentos e classifique seus fios e vias.
2. Diferencie barramento síncrono de assíncrono.
3. Classifique os barramentos quanto aos dispositivos conectados.
4. Dentre os barramentos comerciais, diferencie o ISA do MCA e do EISA.
5. Caracterize o barramento PCI.
6. Caracterize os barramentos AGP e PCI Express.
7. Caracterize as interfaces seriais e paralelas.
8. Detalhe a interface USB.
9. Cite as principais características da interface FireWire.
10. Discuta sobre a tendência de padrão de mercado entre USB e FireWire.



## WEB-BIBLIOGRAFIA

---

<http://www.icea.gov.br/ead/anexo/24101.htm>

<http://www.guiadohardware.net/tutoriais/usb-firewire-dvi/>

<http://www.boadica.com.br/layoutdica.asp?codigo=233>

<http://www.infowester.com/barramentos.php>

<http://www.clubedohardware.com.br/pagina/barramentos>

[http://www.gta.ufrj.br/grad/01\\_1/barramento/](http://www.gta.ufrj.br/grad/01_1/barramento/)

# Unidade 5

## Dentro da CPU

### **Resumo**

Esta unidade destina-se ao estudo do funcionamento interno da UCP. Como atua o ciclo busca-decodifica-executa (BDE), como são representados os dados internamente e como é usada a linguagem de montagem – Assembly.



# Sumário da Unidade

## **UNIDADE V – DENTRO DA CPU**

5.1 – Ciclo BDE .....	77
5.2 – Modos de Endereçamento .....	88



## 5.1 Ciclo BDE

Toda Unidade Central de Processamento (UCP ou CPU) faz sempre a mesma coisa desde que é energizada até ser desligada, fica sempre buscando a próxima instrução a ser executada, caso exista alguma “na vez” esta é decodificada, seja pelo microprograma no caso dos computadores CISC (Conjunto Completo de Instruções) ou pelos circuitos especializados no caso do RISC (Conjunto Reduzido de Instruções). Será mostrado inicialmente o funcionamento dos computadores CISC, tendo em vista a existência de uma unidade no final deste material somente para tratar dos equipamentos com UCP RISC.

Os microprocessadores atuais são compostos de ULA – Unidade Lógico-Aritmética, UC – Unidade de Controle, MAR – Registrador de Endereço de Memória, MBR – Registrador de Bloco de Memória, PC – Contador de Programas, Registradores de Uso Geral e outros componentes. No princípio, a maioria desses componentes ficavam separados na placa-mãe, daí a necessidade de os computadores que seguem o padrão Von Neumann fazerem cópias de dados e endereços para dentro da UCP durante o processamento. Segue a explicação de como a UCP trabalha:

### 5.1.1 Função da Unidade de Controle

Cada operação possui um código identificador único. Para cada código interpretado, uma sequência de micro-operações é realizada. O sinal de clock dá o ritmo da execução das microoperações. Dados e instruções são copiados para dentro da UCP nos registradores. A Unidade de Controle faz o gerenciamento de todas essas operações.

### 5.1.2 Ciclo de Instrução

Realizado em duas etapas: *fetch* ou **busca da instrução** e operandos, se for o caso, e **execução** propriamente dita.

A busca acontece da seguinte forma: O PC armazena o endereço de memória que contém a próxima instrução a ser executada. A UCP busca na memória esse conteúdo para decodificar e executar posteriormente. O valor do PC é incrementado para a próxima instrução a ser executada. Quando a UCP busca o conteúdo de memória, armazena-o no IR – Registrador de Instruções. A interpretação da instrução acontece e, finalmente, sua execução.

### 5.1.3 Execução da Instrução

Para interpretar a instrução é fundamental reconhecer o código que a identifica, chamado de *opcode*. Esse código é reconhecido por um programa presente nos microprocessadores CISC chamado de microprograma, que executa em laço infinito desde que o computador é ligado até o corte da energia que o alimenta. A função principal desse microprograma é verificar cada dígito que identifica a instrução e “descobrir” de qual instrução se trata para fazer a devida busca de operandos se for o caso. Por exemplo, se a instrução descoberta for ADD, que significa SOMA, será necessário buscar quais parcelas serão somadas, os chamados operandos, bem como saber onde será armazenado o resultado de tal somatório.

As instruções podem envolver operações diversas do computador tais como:

- Aritmética;
- Controle (laços, desvios condicionais, e outros);
- Entradas e saídas (de e para periféricos);
- Operações envolvendo a memória e
- Outras.

#### 5.1.4 Interrupções

O trabalho da UCP é executado em laço infinito conforme já foi afirmado, porém, os periféricos precisam de atenção da UCP de vez em quando. Para chamar a atenção da UCP para si, um periférico usa um código próprio chamado de código de interrupção, este gera na UCP uma operação de interrupção. Ao ser interrompida, a UCP precisa salvar todo o seu conteúdo em alguma área de memória para atender ao periférico. Após realizar o atendimento do periférico, a UCP retoma os valores armazenados na memória ao receber a interrupção e continua o processamento normalmente. A esse processo dá-se o nome de TROCA DE CONTEXTO, e será estudado com mais detalhes na disciplina de Sistemas Operacionais.

Uma interrupção também pode acontecer devido à execução normal de um programa. O próprio sistema operacional gera interrupções constantemente para a UCP.

No passado, muitos periféricos recém-adicionados ao PC não funcionavam a contento, pois usavam a mesma interrupção já usada por outros já instalados no sistema. Esse fenômeno era conhecido como **conflito de hardware**. Para solucionar esse problema, com o lançamento

do PENTIUM, chegou o barramento PCI, que já estudamos anteriormente, junto com os Sistemas Operacionais lançados a esta época e que implementaram uma tecnologia conhecida como *plug-and-play*, algo como *ligue-e-use*. É fato que a princípio essa tecnologia funcionou muito mal. Ocorriam mais conflitos que antes, mas com o passar dos anos aperfeiçoamentos foram feitos e a tecnologia hoje funciona a contento. Daí, quando novas placas são adicionadas aos computadores não se tem mais a preocupação de resolver problemas de conflitos de interrupções ou de endereços-base para identificação do periférico, tudo é atribuído pelo BIOS que suporta *plug-and-play* e configura automaticamente.

#### 5.1.5 Acesso Direto à Memória – DMA

DMA envolve um módulo adicional no barramento do sistema. Esse módulo é capaz de imitar a CPU e se necessário assumir o controle do barramento da CPU temporariamente.

##### **Técnica do DMA:**

Quando a CPU deseja ler ou escrever um bloco de dados, ela usa um comando para o módulo de DMA, enviando ao módulo de DMA as seguintes informações:

- Se uma operação de leitura ou escrita é solicitada;
- O endereço do I/O envolvido;
- A localização de início na memória para ser lida ou escrita;
- O número palavras a serem lidas ou escritas.

A CPU então continua seu trabalho e o módulo de DMA executa a transação.

O DMA transfere o bloco inteiro de dados palavra por palavra, uma palavra por vez diretamente da ou para a memória, sem interferência da CPU. Quando a transferência acaba, o DMA envia uma interrupção para a CPU.

A CPU é envolvida apenas no começo e no final da transação. Cada dispositivo ocupa um número de canal de DMA, que também é atribuído pela tecnologia *plug-and-play*.

### 5.1.6 Tipos de Dados

Para entender os tipos de dados em nível de arquitetura, deve-se ter em mente o seguinte esquema:

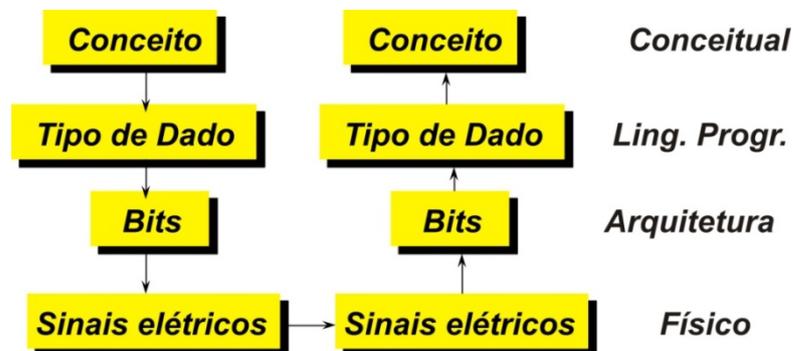


Figura 5.1 Tipos de dados em diversos níveis

Os tipos de dados podem ser classificados como:

- Escalares;
- Números (inteiros e ponto-flutuante);
- Caracteres (ASCII e EBCDIC);
- Lógicos;
- Estruturas Estáticas (Vetores, matrizes e registros);

- Estruturas Dinâmicas (todas baseadas em ponteiros).

Os dados inteiros podem ser representados de três diferentes formas:

1. Sinal-magnitude – onde o bit de mais alta ordem é zero, se o número for positivo e um, se for negativo;
2. Complemento a 1 – no número negativo todos os bits são invertidos;
3. Complemento a 2 – é o complemento a 1 com soma de mais um ao final do processo de inversão.

Atualmente os PCs usam a representação negativa na forma de complemento a 2.

Os dados lógicos podem ser representados usando uma palavra inteira da arquitetura ou apenas um bit de uma dada palavra.

Os números de ponto flutuante são representados usando notação científica, onde se tem alguns bits para o sinal, outros para o expoente e os demais para a mantissa ou fração. A limitação desses valores foi definida pelo IEEE – Instituto dos Engenheiros Eletrônicos e Eletricistas, uma entidade internacional de padronização de projetos nessa área.

A representação do IEEE acontece de acordo com a seguinte figura:

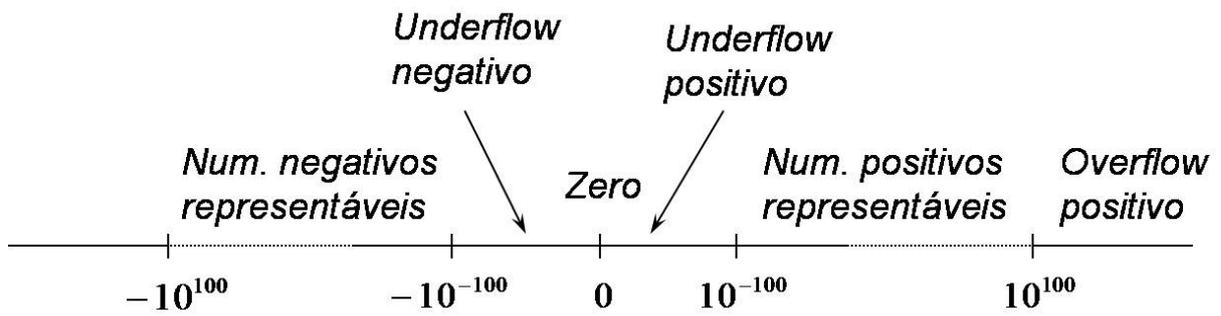


Figura 5.2 Escala de representação IEEE

As linguagens de programação implementam esta escala de acordo com a seguinte tabela:

Item	Precisão Simples	Precisão Dupla
Sinal	1	1
Expoente	8	11
Mantissa	23	52
Total	32	64

Tabela 5.1 – Implementação de ponto-flutuante nas linguagens de programação em bits

### 5.1.7 Repertório de Instruções – Programação Assembly

Esse é o nome dado ao conjunto de instruções que a máquina reconhece e executa. As instruções podem ser classificadas em uma das categorias a seguir:

- *Leitura/Escrita em memória*
- *Operações lógicas e aritméticas sobre dados*
- *Controle da sequência de execução*
- *Entrada/Saída*

As instruções seguem um formato rígido para serem reconhecidas pela UCP, com *opcode* e operandos necessários, a ilustração é feita na figura a seguir:

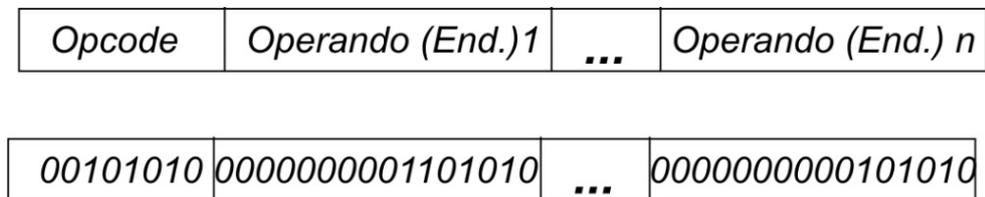


Figura 5.3 Formato das instruções

A representação interna das instruções segue o padrão de bits mostrado na figura anterior. Por outro lado, a representação em nível de programação é feita com *mnemônicos*, ou seja, palavras que representam as instruções. Como, por exemplo, **ADD R1,A** soma o conteúdo do registrador R1 com o conteúdo de uma variável chamada A.

Neste material será abordado um ASSEMBLY reduzido de uma arquitetura hipotética, portanto não será feito, por exemplo, um curso de assembly para PC, mas para aqueles leitores interessados fica a sugestão de estudo para as próximas férias. Mas lembrem-se: programar em assembly em uma máquina real significa ter acesso a algumas funcionalidades em níveis até mesmo mais baixos que o sistema operacional; portanto, deve-se saber o terreno em que se está pisando para evitar perda de dados ou outros desastres com seu sistema. Bons programadores de assembly são requisitados no mercado para operações, desde a criação de um driver para um hardware novo, ou até mesmo para a criação de um novo e mais eficiente antivírus.

As instruções de assembly são criadas de acordo com a arquitetura estudada. Podem ser baseadas em um registrador de uso geral, nesse caso conhecidas como assembly de acumulador, ou baseadas em dois registradores ou ainda baseadas em mais registradores tendo assim três parâmetros. A tabela a seguir ilustra esta classificação:

Num de Endereços	Mnemônicos	Interpretação
3	OP A,B,C	$A \leftarrow B \text{ OP } C$
2	OP A,B	$A \leftarrow A \text{ OP } B$
1	OP A	$AC \leftarrow AC \text{ OP } A$

Tabela 5.2 Ilustração dos tipos de assembly



Figura 5.4 Camadas de execução de assembly

Ao executar o assembly, a UCP segue um padrão de desempacotamento e reempacotamento dos dados conforme pode ser visto na figura a seguir:

As operações aritméticas e lógicas dispensam maiores comentários. Serão comentadas as demais operações previstas na figura anterior.

- Movimentação de dados → cópia ou remoção de dados entre a UCP e a memória;
- Transformação de formato → movimentação de bits através de deslocamento e rotação;
- Transformação de código → quando se faz necessário converter um código em outro por motivos de críticas de dados ou mesmo para compatibilizar dispositivos que a princípio usam códigos incompatíveis entre si.

Um exemplo prático de mudança de código envolve os seus programas onde se deseja implementar uma crítica de dados para saber se o usuário digitou um valor numérico em um campo que deve trabalhar com números. Usa-se a entrada em string para deixar o usuário digitar o que bem entender, após, checam-se os valores digitados se correspondem aos dígitos numéricos e finalmente se faz a transformação da string digitada em valor numérico para ser usado pelo programa. Essa transformação pertence ao grupo de instruções de transformação de código.

## Controle de Fluxo

Em linguagens de programação é comum e necessário fazer uso de controles de fluxo para resolver alguns problemas de lógica que envolvem as diversas programações no computador.

São comuns desvios baseados em estruturas do tipo **se...então...senão**, estes conhecidos como desvios condicionais. Menos comuns hoje em dia, mas não sem importância, são os desvios incondicionais, baseados em estruturas do tipo **go to** e **labels**. No assembly os desvios incondicionais são usados para sair de laços e de outras partes do programa de acordo com a lógica em implementação. Os desvios condicionais em assembly testam o valor de um registrador especial chamado **flags**, ele sempre armazena um código após uma execução de comparação ou de operação aritmética. Segue uma tabela com os valores armazenados nos flags e seus significados:

Valor (2 bits)	Significado
00	Valores iguais
01	Valor1 MAIOR QUE valor2
10	Valor1 MENOR QUE valor2
11	Overflow

Tabela 5.3 Valores dos flags

É preciso lembrar, também, que overflow nos flags ocorre sempre que o valor resultante for grande ou pequeno demais para ser armazenado na estrutura escolhida ou ainda quando for feita uma comparação de dois valores impossíveis de serem comparados, tais como uma string e um valor em ponto flutuante.

De uma forma resumida, pode-se trabalhar com o Assembly apresentado a seguir, feito para uma máquina hipotética de quatro registradores de uso geral.

Instrução	Opcod	Descrição	Tam. instrução
NOP	0000	No operation	1
LDA reg,end	0001	Carrega var de mem em reg	2
STA reg,end	0010	Armazena reg em var de mem	2
ADD reg,end	0011	Reg ← Reg + mem	2
SUB reg,end	0100	Reg ← Reg – mem	2
AND reg,end	0101	Reg ← Reg and mem	2
NOT reg	0110	Reg ← not Reg	2
CMP reg1,reg2	0111	Flags ← reg1 comp reg2	2
JMP endr	1000	Desvia para o label endr	2
JPC cond end	1001	Desvio condicional*	2

Tabela 5.4 Repertório de Instruções do Assembly

\* O parâmetro cond deve ser usado baseado na tabela 5.3

Exemplo: Passar o código-fonte de Pascal para Assembly.

PASCAL	ASSEMBLY	COMENTÁRIOS
IF A=B THEN	LDA R1,A	
A:=A+B	LDA R2,B	1
ELSE	CMP R1,R2	
B:=A-B;	JPC 00 VERDADE	
	SUB R1,B	2
	STA R1,B	3
	JMP FIM	
	VERDADE ADD R1,B	
	STA R1,A	4
	FIM NOP	

1. *Carga de valores para fazer o if.*
2. *Salta para o label VERDADE se a igualdade for confirmada.*
3. *Se a igualdade não for confirmada faz a parte do ELSE e pula pro label FIM que está ligado a uma instrução NOP que é faça nada.*
4. *Implementação da parte verdadeira do IF.*

## 5.2 Modos de Endereçamento

Um operando pode estar em diversas localizações na arquitetura tais como: na memória, no registrador, em um ponteiro, entre outras.

Para cada forma de se especificar o operando na instrução assembly, existe um modo de endereçamento.

### 5.2.1 Endereçamento Imediato

O operando é especificado diretamente na instrução, na forma de uma constante.

Ex: ADD R1,#A (a constante A será adicionada ao conteúdo de R1)

### 5.2.2 Endereçamento Direto

O endereço do operando na memória é especificado na instrução.

Ex.: ADD end1, end2

[end1] ← [end1] + [end2]

### 5.2.3 Endereçamento de Registradores

Apenas registradores são referenciados nas instruções.

Ex.: ADD R3,R5

### 5.2.4 Endereçamento Indireto

O endereço referenciado na instrução, na verdade, contém o endereço do operando real armazenado.

Ex.: ADD R1,(R3)

R3 aponta para o endereço do operando real. Na verdade, esse modo implementa os ponteiros.

### 5.2.5 Endereçamento Indexado

Esse modo de endereçamento é usado para operar vetores e matrizes. A instrução contém o endereço base do array e o deslocamento para mudar de célula.

Ex.: ADD R1, [R2]end

R1 deverá armazenar o somatório dos valores armazenados na matriz.



## EXERCÍCIOS

---

1. Como funciona uma transferência de dados por DMA?
2. Explique o ciclo BDE e a participação do microcódigo das UCPs CISC.
3. Como são implementados os tipos de dados em nível de arquitetura?

4. Fale da representação dos inteiros negativos.
5. Quais os números da reta real que não podem ser representados pelo padrão IEEE?
6. Como se podem representar os dados lógicos?
7. Comente as mudanças observadas no exemplo da passagem de código Pascal para Assembly.
8. Diferencie os modos de endereçamento usados nos computadores.
9. Converta o código a seguir para Assembly:

```
A:=B+C;  
IF B>C THEN  
    B:=B-A  
ELSE  
    B:=C-A;
```

## WEB-BIBLIOGRAFIA



http:

<http://www.geocities.com/SiliconValley/Campus/3064/tutoriais.html>

<http://www.cin.ufpe.br/~mel>

[http://www.ucb.br/prg/professores/gualeve/disciplinas/2005\\_2/ac2/Ramses-Modos.pdf](http://www.ucb.br/prg/professores/gualeve/disciplinas/2005_2/ac2/Ramses-Modos.pdf)

<http://www-asc.di.fct.unl.pt/~jcc/asc1/Teoricas/a8/node5.html>

<http://venus.rdc.puc-rio.br/rmano/ri3mend.html>

<http://www.guiadohardware.net/termos/dma>

[http://edisonfilho.com/Arquivos/ApostilaArqComp/A11\\_AC\\_CPU1.pdf](http://edisonfilho.com/Arquivos/ApostilaArqComp/A11_AC_CPU1.pdf)

<http://professores.faccat.br/assis/davereed/14-DentroDoComputador.html>

[http://edisonfilho.com/Arquivos/ApostilaArqComp/A11\\_AC\\_CPU1.pdf](http://edisonfilho.com/Arquivos/ApostilaArqComp/A11_AC_CPU1.pdf)

<http://www-asc.di.fct.unl.pt/~jcc/asc1/Teoricas/a3/node3.html>

<http://www.decom.iceb.ufop.br/prof/rduarte/CIC130/aulainterruptoes.pdf>

<http://www.di.ufpe.br/~pish/cadis/arg.html>

# Unidade 6

## Pilha

### **Resumo**

Nesta unidade serão abordados tópicos que tratam do suporte ao uso de recursos de programação avançada como ponteiros e sub-rotinas. O chamado suporte ao sistema operacional por parte da arquitetura.

## Sumário da Unidade

### Unidade VI - PILHA

6.1 – Pilha .....	93
6.2 – Gerenciamento de Memória .....	94
6.3 – Realocação .....	95
6.4 – Proteção .....	98
6.5 – Considerações sobre este capítulo .....	98

## 6.1 Pilha

O uso dessa estrutura de dados em nível de hardware é fundamental para que a UCP tenha controle de para onde retornar após executar uma sub-rotina.

Cada programa executado tem seus valores de registradores para cada momento de sua execução. Quando uma rotina é chamada fora da sequência do programa, deve haver um local para salvar todos os valores de UCP naquele momento. Esse local fica na memória RAM e a estrutura usada pelo hardware para saber aonde retornar quando a sub-rotina acabar é a pilha.

Todo programa em execução tem também sua pilha própria, que não deve ser confundida com a pilha geral mantida pelo hardware. Essa pilha própria é mostrada na figura a seguir.

Observe que um dado programa em execução conta com uma fatia de memória para executar. Nessa fatia tem uma área reservada para as linhas de código de um programa e suas variáveis estáticas. Em sentido contrário, cresce a pilha e demais estruturas dinâmicas criadas na área conhecida como *heap*.

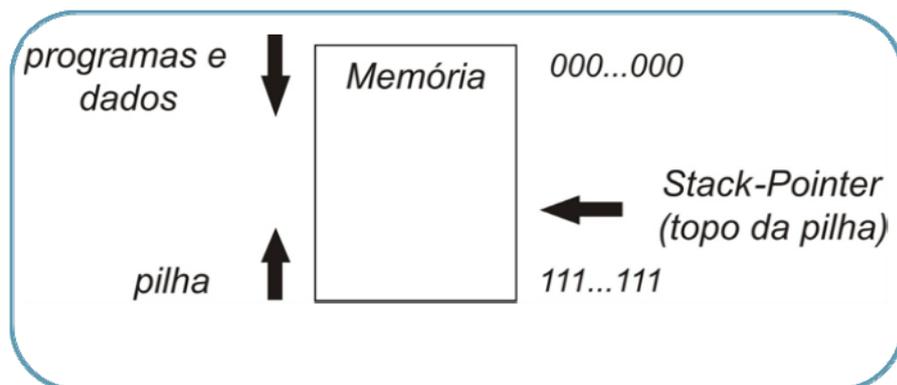


Figura 6.1 – Pilha individual de um programa

As instruções de assembly que fazem uso da pilha são *push* e *pop*, que servem respectivamente para desempilhar e empilhar um dado nesta.

Outras instruções são usadas em pilha, tais como: *call* – empilha o endereço de retorno e muda o fluxo de controle de um programa, usada para chamar uma subrotina; *Ret* – recupera o endereço de retorno, executada ao final de uma sub-rotina para fazer o retorno para o ponto onde o programa estava quando esta foi iniciada.

## 6.2 Gerenciamento de Memória

A memória pode ser classificada do ponto de vista entre a arquitetura e o sistema operacional, de três maneiras distintas:

- Leitura e escrita: variáveis;
- Leitura apenas: constante, Ex: hello world;
- Execução: código objeto.

Para evitar problemas durante a execução de diversos programas com diversos tipos de memória ativos, deve haver um mecanismo de segurança de execução conhecido como mecanismo de proteção. Um dispositivo de hardware deve fazer uma verificação do endereço de memória a cada acesso feito para evitar danos a áreas indevidas.

O sistema operacional mantém um subprograma especial encarregado de fazer o gerenciamento de memória. Este controla o uso da memória por parte dos programas em execução, inclusive quando estes estão sendo trocados por outros programas na atenção da UCP. Esse sistema de gerenciamento de memória faz, inclusive, a execução da chamada memória virtual, que vem a ser o disco rígido usado como expansão da memória RAM para não haver pane no programa por falta de memória. É claro que essa prática tem seu preço, que é exatamente a

lentidão do sistema. O acesso à memória RAM é medido em nanosegundos ( $10^{-9}$  s) e o acesso ao disco rígido é medido em milissegundos ( $10^{-3}$  s), portanto, mil vezes mais lento. É por esse motivo que, quando o computador vem com RAM pequena para os programas que vai executar, ou quando esta se torna pequena para programas novos instalados, o acesso ao disco rígido cresce muito e o sistema se torna mais lento. Algumas pessoas erroneamente apagam arquivos de trabalho, achando que esses é que estão tornando lento o sistema, porém, a verdade é que novas versões dos softwares foram instaladas e a instalação de mais alguns pentes de memória RAM se faz necessária para que se possa usar o computador com algum conforto.

O sistema operacional tenta fazer a parte dele criando no software um hardware que não existe. Porém a solução para um computador de geração recente que se tornou lento pode passar por um acréscimo de RAM. No entanto, se o sistema já tem mais de três anos de uso, será difícil achar memória para o mesmo e, considerando a legislação em vigor no Brasil atualmente, é bem mais racional partir para a aquisição de uma nova máquina, mesmo que se aproveite o monitor, estabilizador e impressora do equipamento antigo. Devido a leis federais de inclusão digital, existem hoje configurações de computadores que não ultrapassam a barreira dos R\$ 800,00 (oitocentos reais) e as lojas ainda fazem crediários para facilitar a aquisição.

### **6.3 Realocação**

Quando um programa é escrito diz-se que tem o código-fonte, que deverá ser submetido a um “processo de tradução” para linguagem de máquina. A esse processo de tradução dá-se o nome de compilação, ou interpretação ou montagem; dependendo da linguagem original onde o código foi escrito.

O processo de compilação transforma de vez o código-fonte em executável, ou binário. Isso ocorre devido às análises que o compilador faz

no fonte. São análises léxicas, sintáticas e semânticas. Isso mesmo, não é aula de Língua Portuguesa mas parece, afinal, é um trabalho com linguagens também.

O processo de interpretação é diferente e mais comum em linguagens mais antigas e em sistemas de formatação de textos como a *web* da Internet. Esse método faz todos os passos do anterior, porém não gera um binário, a cada linha interpretada a instrução é enviada de imediato para a UCP executar. Isso torna o processo mais lento e obriga o usuário a ter o fonte disponível para usar o programa.

A linguagem JAVA, por exemplo, tem seu momento de compilação, porém o produto dessa compilação não é um binário finalizado, e sim, um binário genérico chamado *bytecode*, que numa tentativa de ser universal para qualquer arquitetura, deve ser interpretado pela JVM, ou máquina virtual Java. Daí a lentidão insuportável dos programas feitos nessa linguagem.

Finalmente os códigos montados são aqueles que foram escritos originalmente em Assembly, numa tentativa de fazer um programa que seja mais íntimo da arquitetura. Geralmente os drivers de dispositivos periféricos são montados. Algumas linguagens compiladas geram o chamado código intermediário, que na verdade é um arquivo escrito em assembly que o chamado link-editor do compilador trata de executar também um montador para transformar este assembly em binário executável.

Mas um código ao ser transformado leva também todas as suas estruturas de dados para o modo binário. Aí nasce um problema: essas estruturas na máquina original vão ocupar alguns endereços de memória determinados pelos espaços livres para alocação de novos programas nessa máquina onde o código foi gerado.

Um retrato idêntico em outro computador para o mapa da memória é praticamente impossível de acontecer, daí a necessidade de haver o conceito de realocação de código, que na maioria das vezes, significa o sistema ser capaz de executar novamente esse programa em qualquer outra configuração de mapa de memória.

Por exemplo, ao compilar pela primeira vez um programa seu em linguagem C, o Microsoft Word estava aberto juntamente com o navegador de Internet. Em uma nova execução, esse programa seu foi usado sem nenhum outro estar aberto ao mesmo tempo. Isso leva a mapas de memória completamente diferentes. A execução só se torna possível graças à realocação de código.

Existem alguns modos de se criar programas com instruções especiais para a realocação, a saber:

- Código Absoluto: Esse tipo de código é carregado para a memória sem nenhum tipo de tratamento ou adaptação de endereço. Deve ser usado somente em alguns casos especiais, tais como a criação de um sistema operacional ou drivers para algum dispositivo com endereço fixo de memória;
- Código Realocável: Usado na maioria dos programas, este código é inserido no executável de seus programas e passa instruções para o hardware e para o sistema operacional, que pode ser alocado em qualquer endereço desde que sejam feitas as adaptações aos endereços de suas estruturas de dados em tempo de execução.

## 6.4 Proteção

Com tantos programas executando na memória, é necessário fazer um controle de acessos as suas áreas de execução, que incluem códigos, *heap* e dados.

Cada programa executa em uma partição de memória específica. Para não haver uma invasão, deve-se ocupar dois registradores no gerenciamento dessas áreas, um contém a base onde começa a partição e o outro contém o limite do tamanho da partição. Sendo assim, não existe possibilidade de haver invasões. Cada endereço gerado para uso desse programa é baseado no endereço da base e respeita o endereço limite. Quando o limite é atingido, acontece o erro de falta de memória.

Portanto em programas seus, que usem muita recursividade por exemplo, se ocorre um erro de falta de memória não é o caso de instalar mais um gigabyte de RAM e sim, de usar um parâmetro para o sistema operacional alocar uma partição maior para seu programa executar. Esse parâmetro varia de acordo com a linguagem de programação usada. Recomendamos verificar o manual de sua linguagem de programação se for este seu problema de programação.

## 6.5 Considerações sobre este capítulo

Esse assunto é abordado nesta disciplina de maneira introdutória, pois seu estudo pleno acontece durante as disciplinas de Sistemas Operacionais e de Compiladores.



### EXERCÍCIOS

---

1. Como a estrutura de pilha colabora para o funcionamento de uma arquitetura de computadores?

2. O gerenciamento de memória é importante sob quais aspectos para o funcionamento do sistema?
3. A realocação de código é importante? Justifique sua resposta.
4. Sem realocação de código, a computação estaria no estágio em que está hoje? Justifique sua resposta.
5. Em quê a proteção pode ser útil em um sistema de computação?

http:

WEB-BIBLIOGRAFIA

---

<http://www.fe.up.pt/~jcf/aed/2001/pf/pf2.pdf>

<http://www.portal.netium.com.br/noji/AC/05> - Conjunto de Instruções.ppt

<http://www.lam.ufrj.br/courses/graduate/felipe/cos703/pdfs/arquitetura1.2.pdf>

<http://www.inf.ufg.br/~fmc/arqcomp/Aula1.pdf>

<http://www.sapienstek.com/arqcompl/>

<http://www.dimap.ufrn.br/~ivan/PlanoDeAula20052.htm>

[http://www.dimap.ufrn.br/~ivan/orgl/Suporte ao SO.PDF](http://www.dimap.ufrn.br/~ivan/orgl/Suporte%20ao%20SO.PDF)

[http://www.dcc.ufam.edu.br/~dcc\\_ac/ac\\_08.ppt](http://www.dcc.ufam.edu.br/~dcc_ac/ac_08.ppt)

# Unidade 7

## Motivação

### **Resumo**

Esta última unidade do material trata de uma arquitetura alternativa à CISC (Computadores de Conjunto Complexo de Instruções), trata-se da arquitetura RISC (Computadores com Conjunto Reduzido de Instruções). É uma tecnologia muito interessante porque são máquinas para uso específico, e que contêm apenas aquelas instruções que interessam para o uso ao qual foram destinadas.



## Sumário da Unidade

### **UNIDADE VII - MOTIVAÇÃO**

7.1 – Motivação .....	102
7.2 – Filosofia RISC .....	103
7.3 – Características .....	103
7.4 – Uso de microcódigo x Execução direta em hardware ....	109
7.5 – Repertório de Instruções .....	109
7.6 – Programação e computadores RISC .....	110
7.7 – Diferenças críticas entre as arquiteturas .....	113
7.8 – Comparações de desempenho entre arquiteturas .....	114
7.9 – Entretenimento com arquitetura RISC .....	116

## 7.1 Motivação

As arquiteturas convencionais são muito complexas e a cada dia ganham maior complexidade, pois devem ser sistemas de uso geral onde, não importando qual finalidade vai ter o computador, sua UCP deve reconhecer o maior número de instruções possível. Isto leva a ter que usar circuitos genéricos de execução de instruções e fazer o controle destes através do microprograma. Tudo isso leva a desperdícios de alguns recursos computacionais, tais como ciclos de *clock*, estados de espera e outros.

Por esse motivo, os microprocessadores CISC ou os híbridos de CISC e RISC, conhecidos como CRISC, já romperam a barreira dos GHz (gigahertz) em seus *clocks* internos. É o caso da família Pentium e seus concorrentes atuais.

Para algumas aplicações fixas, muitos recursos desse tipo de computador são desperdiçados. Por exemplo, se o PC será servidor de rede, seus recursos de multimídia e processamento tridimensional de imagens não serão usados, mas estarão presentes no repertório de instruções do processador por toda a sua vida. Porém, se for usado como servidor de rede, um equipamento cujo processador foi concebido e fabricado apenas para esta finalidade, o serviço será mais eficiente, não existirão instruções inúteis dentro daquela UCP, o clock poderá ser mais baixo evitando até interferências em outros produtos eletrônicos que estejam por perto. Como desvantagem, um servidor desses teria o fato de provavelmente não usar um sistema operacional tão popular quanto os existentes para os dominantes PCs. Também, outro fator que costuma pesar contra os servidores ou estações de trabalho RISC é exatamente o preço do equipamento, costuma ser elevado em relação a um PC de última geração bem equipado. Porém, o uso na prática costuma mostrar as vantagens que esta arquitetura pode ter.

## 7.2 Filosofia RISC

Tornam as máquinas mais simples e mais velozes com poucas instruções, simples e com formato único tendo modos simples de endereçamento e fazendo uma implementação mais eficiente.

Para definir o conjunto de instruções de um computador RISC, os projetistas usam a seguinte regra de ouro:

- Analisar aplicações para identificar operações-chave.
- Projetar o processador que seja eficiente para essas operações.
- Projetar instruções que realizam as operações-chave.
- Acrescentar mais instruções necessárias, cuidando para não afetar a velocidade da máquina.

## 7.3 Características

Executar uma instrução por ciclo de clock. Essa é a idéia principal dessa arquitetura, que também é conhecida por arquitetura LOAD/STORE. Onde Load significa carregar instrução ou operando da memória para os registradores, e Store significa armazenar resultados de volta na memória.

Fazer uso eficiente do PIPELINE. Essa característica que já vem sendo adotada nos microprocessadores CRISC a partir do Pentium, tem a idéia de fazer a execução de instruções como se fosse uma linha de produção. Várias instruções são executadas ao mesmo tempo dentro da UCP por causa de suas unidades especializadas. Cada unidade faz uma etapa da instrução e, ao final, mais instruções são executadas na mesma base de tempo. As figuras a seguir ilustram o ciclo BDE nessas máquinas e uma execução em pipeline respectivamente.

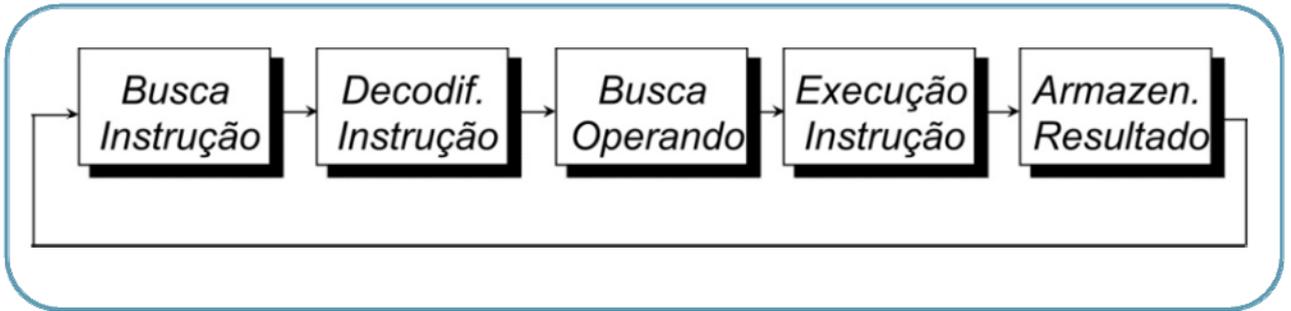


Figura 7.1 Ciclo Busca Decodifica Executa em RISC

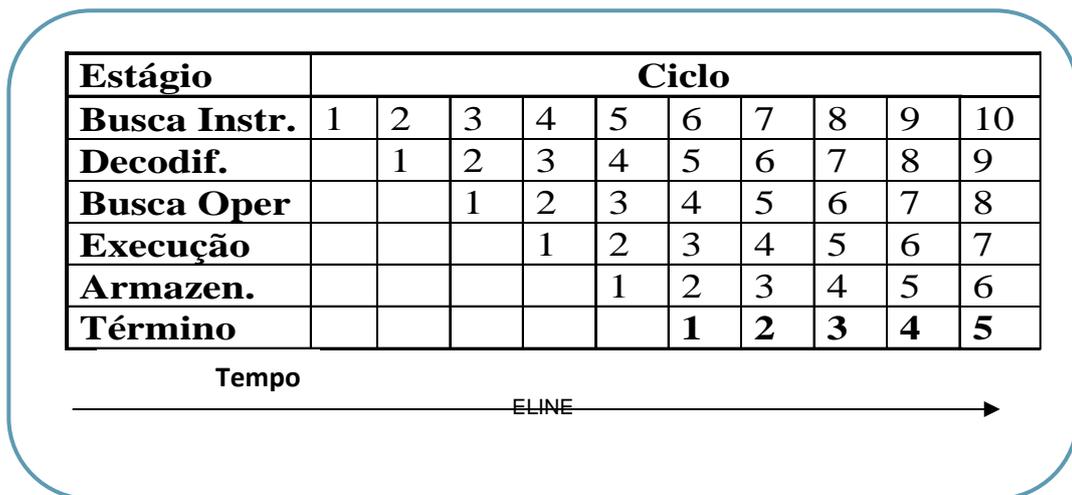


Figura 7.2 Funcionamento de um PIPELINE

A eficiência do pipeline depende de:

- estágios com a mesma duração;
- instruções independentes de resultados calculados em instrução anterior;
- execução sequencial das instruções.

Porém, na prática, acontecem fatos que prejudicam os pipelines tais como:

- dependências de dados;
- instruções de desvio.



Figura 7. : Processador AMD K6-2 450Mhz

Os pipelines podem ser melhor aproveitados quando as linguagens de programação geram códigos executáveis com os chamados HAZARDS de dados e de controle que fazem uso de técnicas de otimização que favorecem os pipelines.

O processador RISC não tem microprograma. Uma maneira de aproveitar esse espaço é colocar mais registradores. É comum ter mais de 500 registradores numa CPU do tipo RISC. Parte desses registradores adicionais pode ser usada para acelerar as chamadas de procedimentos. Outro uso dos registradores é para agilizar a mudança de contexto em interrupções.

Quando um procedimento é chamado, uma pilha em geral é usada para guardar o ponto de retorno, os parâmetros e as variáveis locais. Quando se dispõe de muitos registradores, essas informações podem ser armazenadas em registradores que atuam como uma pilha dentro do processador.

Outro uso dos registradores é guardar o contexto de um processo quando ele é suspenso em uma interrupção. Um conjunto de registradores seriam usados, cada um guarda um contexto diferente. Basta um ponteiro para indicar o contexto atual.

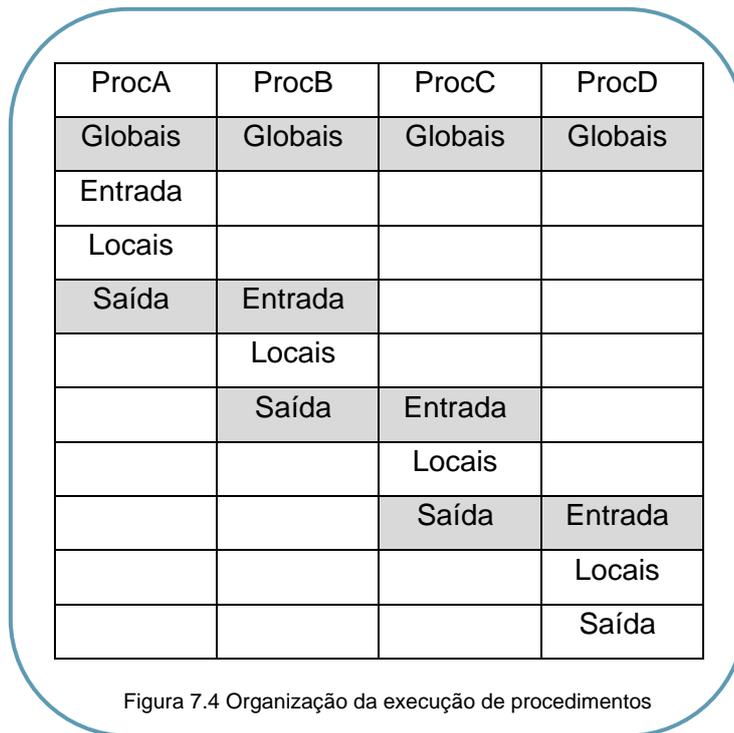
32 bits		
R0..R7	Variáveis globais e ponteiros	Utilizados por todos os procedimentos, o compilador decide o que colocar aqui.
R8..R15	Parâmetros de entrada	Evita o uso de pilha na memória, que é utilizada apenas em caso de exceder 8 parâmetros.
R16..R23	Variáveis locais	Em geral 8 variáveis são suficientes para um procedimento, excesso vai na pilha.
R24..R31	Parâmetros de saída	Parâmetros para os procedimentos chamados (entrada para eles).

Figura 7.3 Uso dos registradores em máquinas RISC

Os registradores são organizados na forma de janelas sobrepostas. Apesar de a CPU possuir muitos registradores, em média 512, nessa técnica apenas um conjunto de registradores, em geral de 32 registradores de 32 bits, é visível para o programa a cada momento. Esse conjunto é ainda dividido em subconjuntos conforme mostra a figura abaixo.

Por esse formato, facilmente conclui-se que os registradores de variáveis globais podem ser os mesmos para todos os procedimentos, assim, não há necessidade de se ter fisicamente um conjunto para cada.

Os parâmetros de saída, que são entradas para o procedimento chamado, também podem ser sobrepostos, economizando espaço.



Quando o procedimento A chama o procedimento B passando parâmetros para ele, a área é comum, este subconjunto de registradores são sobrepostos.

Essa filosofia de uso dos registradores coincide com o uso de memória cache para agilizar o processamento nas arquiteturas CISC.

### Variáveis locais

No RISC todas as variáveis locais, salvo algumas exceções, são armazenadas em registradores. Já, no caso da cache, isso é feito apenas com as variáveis mais recentemente utilizadas.

A vantagem da cache aparece quando se discute a transferência de funções ou trechos de programas para ela, o que seria impossível para os registradores. Porém, uma desvantagem aparece justamente nesse ponto, pois, a transferência é feita por blocos e, quando se trata de variáveis, algumas que são carregadas na cache são serão utilizadas.

## **Variáveis globais**

Semelhante à técnica anterior, estudos mostram que existe um bom número de variáveis globais, porém, nem todos são fortemente utilizadas. A cache, como guarda as mais recentemente utilizadas, com o tempo, terá a vantagem de guardar apenas aquelas mais necessárias.

## **Endereçamento de memória**

No caso de procedimentos ou funções, nos registradores, os dados são movimentados entre eles e a memória com baixa frequência e, como a cache, em geral é pequena e de uso geral, outras informações podem sobrepor os blocos já carregados, incorrendo no atraso de execução.

Para um registrador ser acessado, basta conhecer o seu número e o da janela. Atividade simples e rápida. Porém, para acessar a memória cache, um endereço completo deve ser calculado e a complexidade desse cálculo é diretamente proporcional à complexidade do modo de endereçamento e, sabemos que na arquitetura CISC, isso pode ser muito complexo. A cache é realmente tão rápida quanto um registrador, porém, o acesso a ela é demasiadamente lento.

## **Acesso à memória**

É impossível acessar a memória em um único ciclo. Nesse caso a arquitetura RISC tem que abrir mão da regra básica e aceitar pelo menos duas instruções maiores: leitura e escrita na memória.

Instruções comuns continuam tendo apenas registradores como operandos, o que limita os modos de endereçamento, excluindo por exemplo, o direto, o indexado e o indireto.

Assim, apenas duas instruções estabelecem a comunicação da CPU com a memória, ainda assim, devem utilizar registradores específicos.

Há que se analisar também a questão de instruções como as necessárias para trabalhar com memória virtual, multiprocessamento e assim por diante. Talvez essas tenham sido o grande problema de se manter a ideologia da arquitetura RISC proposta, então o conjunto foi incrementado com novas instruções para executar tais tarefas.

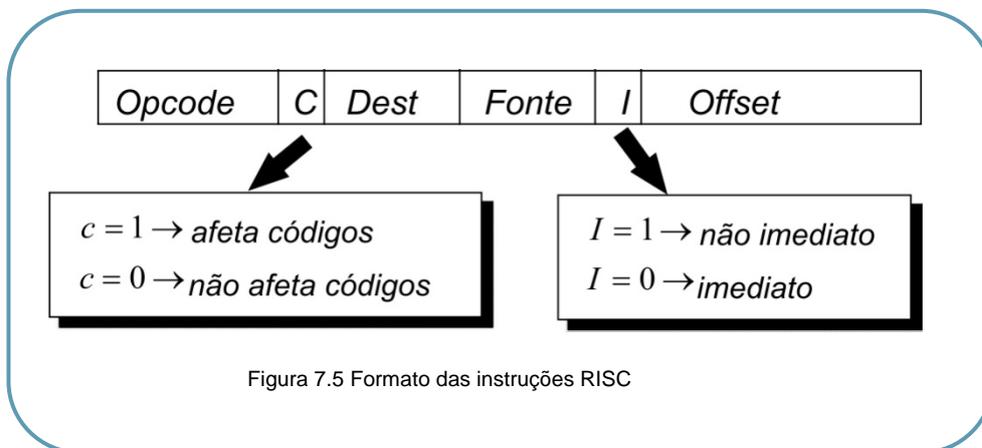
#### **7.4 Uso de microcódigo x Execução direta em hardware**

Os processadores CISC gastam muito tempo e vários ciclos de clock com seu próprio microcódigo para fazer na maior parte das vezes o ciclo BDE, especialmente a fase de decodificar a instrução, já que o opcode é analisado bit a bit para ser identificada a instrução e finalmente ser executada. Isso leva vários ciclos de clock e caracteriza um mau uso dos recursos computacionais por parte desse tipo de UCP.

Por outro lado, os processadores RISC fazem execução direta em hardware. Uma vez que a instrução é carregada, seu opcode já a direciona para o circuito que irá executá-la sem dúvidas de que esta seja a instrução para este circuito. Dessa forma, os ciclos de clock são aproveitados de maneira mais eficiente e os demais recursos computacionais também.

#### **7.5 Repertório de Instruções**

As instruções RISC são mais simples e têm formato fixo tal como ilustra a figura a seguir:



Essa arquitetura trabalha com poucos modos de endereçamento e permite executar instruções de registrador para registrador. Isso facilita o trabalho da UCP, reduz a complexidade do ciclo B-D-E, e agiliza o processamento geral do sistema, pois acessos entre registradores são praticamente instantâneos.

A complexidade maior neste tipo de arquitetura fica por conta do compilador que tem que fazer otimizações rearranjando o código de forma que favoreça o pipeline e garanta os hazards necessários para a execução tranquila dos programas.

## 7.6 Programação e computadores RISC

### Compiladores x hardware

Todo esse estudo conduz para um hardware mais simples quanto possível, não é necessário muito esforço para concluir que a complexidade do compilador cresce proporcionalmente à simplicidade do conjunto de instruções, que no RISC está diretamente ligado ao hardware.

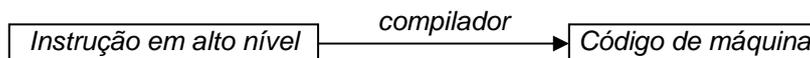


Figura 7.6 Interação entre o compilador e arquitetura

Os defensores do CISC alegam que a distância semântica conduz à ineficiência de execução, programas excessivamente grandes e compiladores complexos e argumentam que a disponibilidade de instruções de alto nível e a habilidade de especificar múltiplos operandos baseados na memória, simplifica o desenho do compilador. Naturalmente, isso leva diretamente para um conjunto de instruções grande, dezenas de modos de endereçamento e muitas declarações da linguagem de alto nível implementadas em firmware.

Existem divergências entre as estruturas estudadas e as implementadas, como exemplo pode-se citar a instrução LOOP utilizada já no PC para controle de laço de repetição. Essa instrução repete um bloco de instruções por um número fixo de vezes como o "para-faça", mas faz isto, pelo menos uma vez como o "repita-até".

```
mov cx, 32      ;estabelece o critério de parada
Aqui:          ....
               ....
               loop aqui ;decrementa cx, e desvia para aqui se cx > 0
```

Figura 7.7 Trecho de programa para exemplo

## Otimização dos compiladores

Um programa escrito em linguagem de alto nível, não explicita os registradores que serão utilizados, ao invés disso, faz referências simbólicas aos valores através das variáveis criadas. Além disso, não explicita se o valor associado à variável será guardado num registrador ou na memória. Como se sabe que os registradores são mais rápidos que a memória, é desejável que sejam largamente utilizados.

Cada variável passível de ser guardada em um registrador, recebe um tratamento especial. O compilador pode criar um número ilimitado "registradores virtuais" para armazená-las e, a partir daí, compartilhar o registrador real de acordo com alguma técnica específica. Claro, isso é feito dentro da limitação da CPU e muitas variáveis vão obrigatoriamente para a memória.

Os compiladores mantêm essa filosofia de forma transparente para o programador, mas alguns, como é o caso da linguagem C, oferecem a possibilidade do programador decidir quais variáveis devem, sempre que possível, serem armazenadas em registradores, para tanto, ao declarar uma variável, ao invés de escrever "int soma", escreve-se "register int soma", obrigando então, o compilador a priorizar os registradores para aquela variável.

As arquiteturas RISC utilizam, frequentemente, a técnica dos grafos coloridos para otimizar o uso de registradores pelas variáveis.

## 7.7 Diferenças críticas entre as arquiteturas

A tabela abaixo ilustra as grandes diferenças existentes entre as arquiteturas CISC e RISC.

RISC	CISC
Instruções simples levando um ciclo	Instruções complexas levando múltiplos ciclos
Apenas LOAD/STORE referenciam a memória	Qualquer instrução pode referenciar a memória
Altamente <i>pipelined</i>	Não tem <i>pipeline</i> ou tem pouco
Instruções executadas pelo hardware	Instruções interpretadas pelo microprograma
Instruções com formato fixo	Instruções com vários formatos
Poucas instruções e modos	Muitas instruções e modos de endereçamento
A complexidade está no compilador	A complexidade está no microprograma
Múltiplos conjuntos de registradores	Conjunto único de registradores

Tabela 7.1 Diferenças entre as arquiteturas

ARQUITETURA	RISC	CISC
<b>Implementação do controle</b>	Hardware	Microprogramação
<b>Comprimento das instruções</b>	Fixo	Variável
<b>Número de registradores</b>	Tipicamente alto (32 a 512)	Tipicamente baixo (4 a 16)
<b>Execução das instruções</b>	Alta superposição baseada em pipeline	Baixa superposição
<b>Número de instruções</b>	Médio (tipicamente 64)	Alto (mais de 100)
<b>Instruções de desvio</b>	Desvio atrasado, Hardware de apoio opcional, para predição de desvios	Normal (condicional ou absoluto)

Tabela 7.2 Comparação de características gerais das arquiteturas

## 7.8 Comparações de desempenho entre arquiteturas

Desempenho é muito dependente da aplicação. Antes de mais nada, é importante mostrar todas as unidades de medidas de desempenho entre sistemas de computação. Esses testes práticos de desempenho têm o nome de *benchmarks* e são feitos por programas específicos para realizar tais medidas.

**MIPS:** Milhões de Instruções Por Segundo

- Uma tarefa ↔ Duas máquinas: n° instruções diferentes

**MFLOPS:** Milhões de Operações em Ponto Flutuante por Segundo

**Whetstones:** Benchmark sintético

- Programa para testar desempenho em ponto flutuante

**Dhrystones:** Competidor do Whetstone

- Desenvolvido com ênfase em operações inteiras

**SPEC:** System Performance Evaluation Cooperative

- Consórcio de empresas formado em 1987

### **Padrão SPEC-benchmark**

- Um compilador;
- Um programa de minimização lógica;
- Uma planilha;
- Outros programas com ênfase no processamento aritmético;
- SPECint92 e SPECfp92.

Todos os esforços para medir desempenho buscam mostrar aos usuários o desempenho de um computador frente a sua carga de trabalho.

Agora serão mostradas tabelas que comparam alguns sistemas CISC com RISC

TIPO DE INSTRUÇÃO	NÚMERO DE CICLOS POR INSTRUÇÃO		
	486 (CISC)	SPARC (RISC)	MOTOROLA 89000 (RISC)
Load	1	2	1-3
Store	1	3	1
Register-Register	1	1	1
Jump	3/1	1/2	1
Call	3	3	1

Tabela 7.3 Comparativo de número de ciclos por instrução

Fonte: Computer World

MICROPROCESSADOR	TECNOLOGIA	NÚMERO INSTRUÇÕES	MICROPROCESSADOR
IBM POWER (RS/6000)	RISC	184	IBM POWER (RS/6000)
HP PA	RISC	140	HP PA
SUN SPARC	RISC	70	SUN SPARC
MOTOROLA 68040	CISC	120	MOTOROLA 68040
INTEL 386	CISC	96	INTEL 386

Tabela 7.4 Tamanho do repertório de alguns processadores

Fonte: Computer World

Finalmente segue um gráfico de benchmark entre alguns dos mais recentes processadores CRISC AMD e Intel, com o RISC puro SPARC 64.

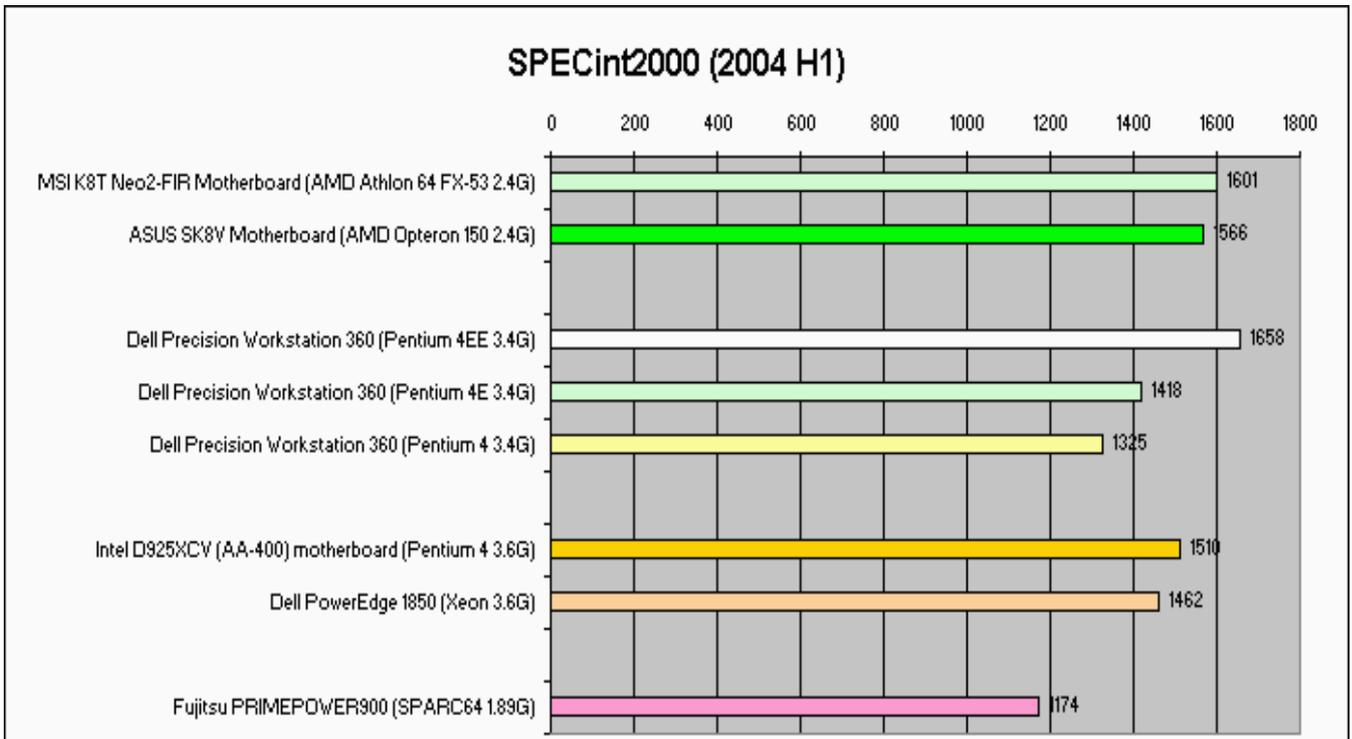


Figura 7.8 Benchmark entre UCPs recentes CRISC e RISC

### 7.9 Entretenimento com arquitetura RISC

Os consoles de vídeo-games mais atuais buscam cada vez mais a perfeição de imagens e sons na tentativa de fazer os ambientes e situações dos jogos se tornarem cada vez mais reais.

Nesse sentido, os processadores RISC foram um grande achado para os projetistas desses equipamentos de diversão. Basta projetar UCPs RISC altamente voltadas para as instruções que interessam aos games que tudo fica mais fácil.

Segue uma tabela que resume alguns dos consoles mais populares atualmete e suas características de UCP, memória e outras.

<b>Modelo</b>	<b>UCP</b>	<b>Clock</b>	<b>MIPS</b>	<b>RAM</b>
Sega Dreamcast	Hitachi SH-4	200 MHz	360	16 MB
Sony Playstation 1	R3000A 32bits	33,86 MHz	30	4 MB
Sony Playstation 2	Emotion Engine 128 bits	300 MHz	450	48 MB
Microsoft X-BOX	Pentium III ou Celeron	733 MHz	CRISC	64 MB
Sony PSP	PSP CPU	333 MHz	Não Inform	32 MB
Nintendo 64	MIPS R4300i	93,75 MHz	125	4 a 8 MB
Sony Playstation 3	Cell Broadband Engine Power PC	3,2 GHz	2 TeraFlops	256 MB dados 256 MB vídeo

Tabela 7.5 Comparação dos modelos de consoles

Observe que a garotada que gosta de jogos eletrônicos está ficando com equipamentos cada vez mais poderosos, sendo predominantemente de tecnologia RISC.



## EXERCÍCIOS

1. Caracterize as máquinas RISC.
2. Discuta por que as arquiteturas RISC devem ser específicas para uma dada aplicação.
3. O uso de vários registradores é um grande diferencial dessa arquitetura? Justifique sua resposta.
4. Por que os clocks de processadores RISC tendem a ser mais baixos que os seus contemporâneos CISC?
5. O que é Pipeline?
6. Em que aspecto difere a compilação de um programa para uma arquitetura RISC?
7. Como a tecnologia RISC se faz presente nos lares do mundo inteiro?

WEB-BIBLIOGRAFIA

http:

<http://www.ime.usp.br/~song/mac412/oc-RISC.pdf>

<http://www.din.uem.br/sica/material/4ano/RISC.doc>

<http://www.inf.ufrgs.br/procpa/disc/cmp134/trabs/T1/991/PowerPC/capi2.html>

<http://www.inf.ufsc.br/~lucia/Arquivos-INE5607/anterior/RISC-CISC.pdf>

[http://brahms.di.uminho.pt/Discip/TextoAC/cap6.html#RISC\\_CISC](http://brahms.di.uminho.pt/Discip/TextoAC/cap6.html#RISC_CISC)

[http://knol.google.com/k/jos-rodriques/arquitetura-RISC-CISC-e-hbri-da/3bxe6fr7ycrsq/4?domain=knol.google.com&locale=en#RISC\(C2\)\(A0\)x\\_CISC](http://knol.google.com/k/jos-rodriques/arquitetura-RISC-CISC-e-hbri-da/3bxe6fr7ycrsq/4?domain=knol.google.com&locale=en#RISC(C2)(A0)x_CISC)

[http://www.cs.virginia.edu/~skadron/cs654/cs654\\_01/slides/hua.ppt](http://www.cs.virginia.edu/~skadron/cs654/cs654_01/slides/hua.ppt)

<http://www.dcce.ibilce.unesp.br/~norian/cursos/aoc/1s05/CISCxRISC.pdf>

<http://www.pr.gov.br/batebyte/edicoes/1994/bb35/aspectos.htm>

[http://www.cs.virginia.edu/~skadron/cs654/cs654\\_01/slides/dhiraj.ppt](http://www.cs.virginia.edu/~skadron/cs654/cs654_01/slides/dhiraj.ppt)

<http://www.rtftechnologies.org/Design/Assets/device-images/amigobot/comparison-rpt.doc>

<http://www.us.playstation.com/>



## APÊNDICE

---

Segue um texto muito interessante encontrado no endereço <http://www.fortunecity.com/roswell/king/622/processadores.htm> sobre a evolução dos processadores.

### **Processadores**

O microprocessador é o principal componente de um computador. Um computador equipado com um processador Pentium, será chamado de "Pentium" e um outro com um processador 486 será chamado de "486". Porém, é importante entender que o desempenho de um computador não é determinado apenas pelo processador, e sim pelo trabalho conjunto de todos os componentes: placa-mãe, memória RAM, HD, Placa de Vídeo, etc. Caso apenas um desses componentes ofereça uma performance muito baixa, o desempenho do computador ficará seriamente prejudicado. Não adianta colocar um motor de Ferrari num Fusca. Um mero Pentium MMX com bastante memória RAM, um HD Rápido e uma boa placa de vídeo pode facilmente bater em performance um Pentium II com um conjunto fraco.

Vamos agora falar sobre as características dos microprocessadores utilizados nos micros PC's, tanto os produzidos pela Intel, quanto por outros fabricantes como a Cyrix e a AMD.

### **Processadores RISC x Processadores CISC**

Sempre houve uma grande polêmica em torno de qual dessas plataformas é melhor. Talvez você ache inútil estarmos falando sobre isso, mas é interessante compreender a diferença entre essas duas plataformas, para entender vários aspectos dos processadores modernos.

Um processador CISC (**C**omplex **I**nstruction **S**et **C**omputer), é capaz de executar várias centenas de instruções complexas, sendo extremamente versátil. Exemplos de processadores CISC são o 386 e o 486.

No começo da década de 80, a tendência era construir chips com conjuntos de instruções cada vez mais complexos. Mas alguns fabricantes resolveram seguir o caminho oposto, criando o padrão RISC (**R**educed **I**nstruction **S**et **C**omputer). Ao contrário dos complexos CISC, os processadores RISC são capazes de executar apenas algumas poucas instruções simples. Justamente por isso, os chips baseados nessa arquitetura são mais simples e muito mais baratos. Outra vantagem dos processadores RISC, é que por terem um menor número de circuitos internos, podem trabalhar com clocks mais altos. Um exemplo são os processadores Alpha que em 97 já operavam a 600 MHz.

Pode parecer estranho que um chip que é capaz de executar algumas poucas instruções, possa ser considerado, por muitos, mais rápido do que outro que executa centenas delas. Seria como comparar um professor de matemática com alguém que sabe apenas as quatro operações. O que acontece, é que um processador RISC é capaz de executar tais instruções muito mais rapidamente. Assim, em conjunto com um software adequado, esses processadores são capazes de desempenhar todas as funções de um processador CISC, compensando suas limitações com uma velocidade maior de operação.

É indiscutível porém, que em instruções complexas, os processadores CISC saem-se muito melhor. Por isso, ao invés da vitória de uma das duas tecnologias, atualmente vemos processadores híbridos, que são essencialmente processadores CISC, porém, que possuem internamente núcleos RISC. Assim, a parte CISC do processador pode cuidar das instruções mais complexas, enquanto que o núcleo RISC pode cuidar das mais simples, nas quais é mais rápido. Parece que o futuro nos reserva

uma fusão dessas duas tecnologias. Um bom exemplo de processador híbrido é o Pentium Pro.

### **Do 8086 ao Pentium**

Talvez você ache pouco interessante ler sobre esses processadores obsoletos, mas é interessante conhecer seu funcionamento para entender muitos dos recursos utilizados nos processadores mais modernos. Na pior das hipóteses, você irá aprender um pouco mais sobre a história da Informática.

#### **Intel 8086**

Lançado em 1978, foi o primeiro processador de 16 bits a ser criado. Acabou sendo um grande fracasso, pois na época não existiam circuitos de apoio que pudessem trabalhar a 16 bits, sendo utilizado apenas em alguns sistemas corporativos. O 8086 podia acessar até 1 MB de memória RAM e permitia o uso de um coprocessador aritmético externo, o 8087 que poderia ser adquirido separadamente.

#### **Intel 8088**

O 8088 era idêntico ao 8086, mas apesar de internamente funcionar com palavras binárias de 16 bits, externamente trabalhava com palavras de 8 bits. Isso permitia seu uso em conjunto com periféricos como placas de vídeo e discos de 8 bits, que eram muito mais baratos na época, sendo justamente esse o motivo da sua popularização. O 8088 foi usado nos micros IBM PC e IBM XT, e também em clones de outros fabricantes, e possuía velocidade de operação de 4,77 MHz.

#### **Intel 286**

O i286 trabalhava usando palavras de 16 bits tanto interna quanto externamente. Foi lançado quando já existiam circuitos de apoio 16 bits a

preços acessíveis, conseguindo uma espantosa aceitação. O 286 permitia também o uso de um coprocessador aritmético, o 80287 que deveria ser adquirido à parte. O 286 foi utilizado nos micros PC-AT da IBM e em clones de vários concorrentes.

O 286 trouxe um grande avanço sobre o 8086, com seus modos de operação: o "Modo Real" e o "Modo Protegido". No modo real, o 286 se comporta exatamente como um 8086 (apesar de mais rápido), oferecendo total compatibilidade com os programas já existentes. Já no modo protegido, ele incorpora funções mais avançadas, como a capacidade de acessar até 16 megabytes de memória RAM, multitarefa e memória virtual em disco.

Assim que ligado, o processador opera em modo real, e com uma certa instrução passa para o modo protegido. O problema é que, quando em modo protegido, o 286 deixa de ser compatível com os programas escritos para 8088; e uma vez em modo protegido, não havia uma instrução que o fizesse voltar para o modo real, (somente reiniciando o micro). Assim, apesar de oferecer os recursos do modo protegido, poucos foram os programas capazes de usá-lo. Por esse motivo, os computadores equipados com processadores 286 eram geralmente utilizados simplesmente como XT's mais rápidos.

### **Intel 386**

Lançado pela Intel em 85, o 386 trabalha interna e externamente com palavras de 32 bits, sendo capaz de acessar até 4 gigabytes de memória RAM e, ao contrário do 286, ele pode alternar entre o modo real e o modo protegido. Foram então desenvolvidos vários sistemas operacionais como o Windows 3.1, OS/2, Windows 95 e Windows NT, que funcionavam usando o modo protegido do 386.

O 386 era muito rápido para as memórias RAM existentes na época. Por isso, muitas vezes ele tinha que ficar "esperando" os dados serem liberados pela memória RAM para poder concluir suas tarefas, perdendo muito em desempenho. Para solucionar esse problema, foram inventadas as memórias cache (SRAM) que são utilizadas em pequena quantidade na grande maioria das placas-mãe para micros 386 e superiores. Essa memória cache é um tipo de memória ultra-rápida que armazena os dados da memória RAM mais usados pelo processador, de modo que mesmo uma pequena quantidade dela melhora bastante a velocidade da troca de dados entre o processador e a RAM.

O 386 exige o uso de periféricos de 32 bits, que eram muito caros na época, por isso, a Intel lançou uma versão do 386 de baixo custo, chamada de 386 SX, que internamente trabalhava à 32 bits, porém externamente funcionava à 16 bits, possibilitando a fabricação de placas-mãe mais baratas usando basicamente os mesmos componentes das placas de 286. Para não haver confusão, o 386 original passou a ser chamado de 386 DX.

O 386 permite o uso dos coprocessadores aritméticos 80387SX (para o 386 SX) e o 80387DX (para o 386 DX). Outros fabricantes como a AMD também lançaram seus modelos de 386.

#### **486DLC e 486SLC**

Lançados pela Cyrix, esses processadores nada mais são do que processadores 386 (respectivamente o DX e o SX) que possuíam um cache interno de 8 KB, usando inclusive placas-mãe de 386. Apesar disso, devido ao cache, o seu desempenho era bastante superior aos processadores 386, e como se podia trocar apenas o processador num upgrade, acabou se tornando uma boa opção para *Upgrades* de baixo custo.

## Modo real e modo protegido

Operando em modo real, o processador opera exatamente como um 8086, apenas funcionando a um clock maior. Não somente o 286 e o 386, mas todos os processadores atuais podem alternar entre o modo real e o modo protegido. No modo real, rodamos o MS-DOS e alguns aplicativos mais antigos, enquanto no modo protegido rodamos o Windows e seus programas.

Com certeza, alguma vez ao tentar rodar um programa antigo, você já se deparou com uma enigmática mensagem de falta de memória, apesar dos manuais do programa dizerem que ele precisa apenas de 500 ou 600 KB de memória e você ter instalados 16, 32, 64 ou mesmo 128 megabytes no seu computador. Essas mensagens surgem porque esses programas rodam com o processador operando em modo real, onde -como o 8086- ele é capaz apenas de reconhecer o primeiro megabyte da memória RAM. Esse primeiro megabyte é subdividido em dois blocos, chamados de memória convencional e memória estendida.

A memória convencional corresponde aos primeiros 640 KB da memória e é a área de memória usada pelos programas que operam em modo real. Os 384 KBytes restantes, são chamados de memória superior, e são reservados para uso do Bios. Nessa faixa de memória, são gravadas as ROMs de vários dispositivos, como da placa-vídeo e também do próprio Bios.

Mesmo assim, o programa não deveria rodar, já que ele precisa apenas de 600 Kbytes, e eu possuo 640 Kbytes de memória convencional? A resposta é não, pois apesar de possuímos 640 bytes de memória convencional, pronta para ser usada por qualquer programa que opere em modo real, nem toda essa memória fica disponível, já que parte dela é usada pelo MS-DOS e drivers de dispositivos de modo real. Mas de qualquer forma é possível liberar mais memória convencional, editando os arquivos

de inicialização do DOS ou do Windows 95, conseguindo assim rodar esses programas.

Quando o computador é ligado, o processador está operando em modo real. Quem dá o comando para que ele mude para o modo protegido é o sistema operacional. No caso do Windows, esse comando é dado durante o carregamento do sistema.

Em modo protegido, o processador é capaz de reconhecer toda a RAM instalada no sistema, além de incorporar recursos como a multitarefa e a memória virtual em disco, é nesse modo que usamos a interface gráfica do Windows e rodamos seus aplicativos.

### **Intel 486**

Ao contrário dos processadores anteriores, fora a maior velocidade de processamento, o 486 não trouxe nenhuma grande inovação. Como o 386, ele trabalha a 32 bits e é capaz de acessar até 4 gigabytes de memória RAM. A diferença ficou por conta do acréscimo de um cache interno (L1) de 8KB e da adoção de um coprocessador aritmético interno. Apesar disso, devido às brutais mudanças na arquitetura, o 486 é cerca de duas vezes mais rápido do que um 386 do mesmo clock.

Como anteriormente, a Intel criou um 486 de baixo custo, chamado de 486 SX, que era idêntico ao original, porém sem o coprocessador aritmético interno, podendo ser acoplado a ele o 80487SX. O 486 original passou então a ser chamado de 486 DX.

Foram lançadas versões do 486 à 25 MHz, 33 MHz e 40 MHz, porém criou-se uma barreira, pois não havia na época circuitos de apoio capazes de

trabalhar a mais de 40 MHz. Para solucionar esse problema, foi criado o recurso de Multiplicação de Clock no qual o processador trabalha internamente a uma velocidade maior do que a da placa-mãe. Foram lançados então os 486 DX-2 (que trabalhavam o dobro da freqüência da placa-mãe) e, logo depois, os 486 DX-4 (que trabalhavam o triplo da freqüência da placa-mãe)

<b>Velocidade do Processador</b>	<b>Velocidade da placa-mãe</b>	<b>Multiplicador</b>
486DX-2 50 MHz	25 MHz	2x
486DX-2 66 MHz	33 MHz	2x
486DX-2 80 MHz	40 MHz	2x
486DX-4 75 MHz	25 MHz	3x
486DX-4 100 MHz	33 MHz	3x
486DX-4 120 MHz	40 MHz	3x

Com isso, surgiram também as placas-mãe "up-gradable", que suportavam a troca direta de um DX 33 por um DX-2 66, por exemplo, simplesmente mudando-se a posição de alguns jumpers localizados na placa.

Mais uma novidade trazida pelo processadores 486 é a necessidade do uso de um ventilador (cooler) sobre o processador para evitar que ele se aqueça demais. O uso do cooler é obrigatório em todos os processadores 486 DX-2 e posteriores.

## **Multiplicação de Clock**

Esse recurso consiste em fazer o processador trabalhar internamente a uma frequência maior do que a placa-mãe e os demais componentes do micro. Assim, apesar do processador trabalhar à sua velocidade nominal, ele comunica-se com os demais componentes na frequência da placa-mãe, que geralmente é de 66 ou 100 MHz nos processadores mais recentes, sendo geralmente de 40 MHz nos micros 486.

O uso da multiplicação de clock permite atingir velocidades elevadas, pois é muito mais fácil desenvolver processadores velozes do que placas-mãe e circuitos de apoio que funcionem a tal velocidade. Claro que existe um limite, pouco adianta criar um processador super veloz e utilizar um multiplicador muito alto para fazê-lo funcionar, se a todo momento o processador tem que ficar esperando para acessar dados na memória RAM ou HD, ou mesmo ficar esperando a placa de vídeo terminar de exibir uma imagem para poder enviar a próxima. Devido a isso, um computador equipado com um processador Pentium de 200 MHz não é duas vezes mais rápido do que um de 100 MHz com configuração semelhante, pois em ambos a placa-mãe funciona a 66 MHz. Na prática, o Pentium 200 mal chega a ser 70% mais rápido.

## **Coprocessador Aritmético**

A função desse processador é auxiliar o processador principal no cálculo de números fracionários, ou de ponto flutuante. Em aplicações que fazem uso intenso deste tipo de cálculo, como programas gráficos e jogos com gráficos poligonais, a presença desse auxiliar é indispensável. Apesar do processador principal também ser capaz de executar tais funções, isso prejudicaria muito o desempenho. Por isso, a partir dos micros 486, o coprocessador passou ser um item obrigatório. O desempenho do coprocessador aritmético é tratado como "fpu" em benchmark comparativos.

Esse recurso de criar chips auxiliares do processador principal é um recurso muito usado, pois é muito mais racional usar chips baratos para executar ações simples que antes congestionavam o processador principal, do que investir em processadores mais velozes. Por exemplo, todos os modems atuais possuem Uart, que é um conjunto de circuitos que permitem ao modem gerenciar ele mesmo o envio e recebimento de dados, deixando o processador principal livre para executar outras tarefas. Como não poderia deixar de ser, alguns fabricantes "espertos" lançaram versões de modems sem a Uart, que obrigam o processador a fazer todo o trabalho, degradando muito o desempenho geral do sistema. Tais modems são chamados de WinModems e não são uma boa opção de compra.

### **Intel Pentium**

Sucessor do 486, o Pentium ainda é um processador de 32 bits. Pode-se pensar, então, que já que ele continua trabalhando com palavras binárias de 32 bits, qual seria a vantagem dele sobre o 486. São basicamente duas:

A primeira, é que ao contrário do 486, o Pentium acessa a memória usando palavras binárias de 64 bits. São acessados dois bits por vez ao invés de apenas um, o que melhora a velocidade de acesso às memórias, ajudando a solucionar o antigo problema da lentidão da memória RAM. Outra novidade é sua arquitetura superescalar. O Pentium funciona internamente com dois processadores de 32 bits distintos, sendo capaz de executar 2 instruções por ciclo de clock, preservando também a compatibilidade com programas escritos para processadores mais antigos.

A segunda é que o Pentium possui um cache L1 de 16 KB embutido e trabalha com velocidades de barramento de 50 a 66 MHz, o que somado à maior velocidade de acesso à memória RAM, o torna cerca de 2 vezes mais rápido do que um 486 do mesmo clock.

Como no 486, os processadores Pentium possuem um coprocessador aritmético embutido e usam multiplicador de clock:

Velocidade do Processador	Velocidade da Placa-mãe	Multiplicador
P-50 MHz	50 MHz	1x
P-55 MHz	55 MHz	1x
P-60 MHz	60 MHz	1x
P-66 MHz	66 MHz	1x
P-75 MHz	50 MHz	1,5 x
P-80 MHz	55 MHz	1,5 x
P-90 MHz	60 MHz	1,5 x
P-100 MHz	66 MHz	1,5 x
P-120 MHz	60 MHz	2 x
P-133 MHz	66 MHz	2 x
P-150 MHz	60 MHz	2,5 x
P-166 MHz	66 MHz	2,5 x
P-180 MHz	60 MHz	3 x
P-200 MHz	66 MHz	3 x

Como nos 486, as placas-mãe para Pentium mais recentes suportam várias frequências de barramento e vários multiplicadores distintos, podendo ser configuradas para o uso com todos os processadores da família.

### Processadores In-a-Box

"In-a-Box" significa numa tradução livre "numa caixa". Geralmente, um mesmo processador é vendido em duas versões, a In-a-Box e a OEM. Na versão in-a-box o processador vem dentro de uma caixa, acompanhado de holografias, manuais, e de uma garantia maior. A versão OEM é vendida para integradores, e não acompanha nenhuma das quinquilharias da

versão in-a-box, tendo também uma garantia bem menor, passando segundo as más línguas, também por um controle de qualidade menos rigoroso.

A principal vantagem dos processadores in-a-box é o fato de virem acompanhados de um cooler de ótima qualidade, que é fixo sobre o processador.

Apesar de um pouco mais caros, vale investir um pouco a mais num processador in-a-box, pois além do ótimo cooler, temos a certeza de não se tratar de um processador remarcado.

### **AMD "586"**

Esse processador foi lançado pela AMD pouco depois do lançamento do Pentium pela Intel. Porém, ao contrário do que se pode pensar pelo nome, de Pentium esse processador não tem muita coisa. Ele usa placas de 486 utilizando barramento de 33 MHz e multiplicador de 4x, totalizando os seus 133 MHz. Devido à estratégia de Marketing, muitos pensavam se tratar de um "Pentium Overdrive", porém esse processador não passa de um 486 um pouco mais rápido. Comparado com um 486 DX-4 100, a diferença de performance é de apenas 33%, servindo apenas como uma alternativa barata de upgrade. A Cyrix também lançou um processador muito parecido, chamado de Cyrix 586.

### **AMD K5**

Pentium Compatível da AMD, oferece um desempenho bastante semelhante ao Pentium da Intel. Perde apenas no desempenho do coprocessador aritmético que é lento se comparado ao da concorrente. O K5 não chegou a se tornar muito popular devido ao seu lançamento atrasado. Quando finalmente saíram as versões de 100 e 133 MHz do K5, a Intel já havia lançado as versões de 166 e 200 MHz do Pentium, ficando difícil a concorrência.

### **Pentium Overdrive**

Como fez com os antigos 386 SX, a Intel lançou também um Pentium "Low Cost". Este processador apesar de internamente ter um funcionamento idêntico a um Pentium, utiliza placas-mãe para processadores 486, sendo por isso chamado de Overdrive. Foi lançado em duas versões: de 63 MHz (25 MHz x 2,5) e 83 MHz (33 MHz x 2,5).

Devido à baixa velocidade de barramento e à compatibilidade com os antigos componentes das placas de 486, esses processadores perdem feio em performance se comparados com um Pentium "de verdade": o de 63 MHz apresenta performance idêntica ao 486 DX4 100 e o de 83 MHz uma performance pouco superior. Não fizeram muito sucesso devido a serem muito caros considerando-se o ganho em performance, por isso é quase impossível encontrar um. Em termos de custo-benefício o 586 da AMD é muito melhor.

### **Nos dias de hoje**

Finalmente acabamos a nossa "aula de história" e vamos agora falar sobre os processadores mais modernos. Note que as tecnologias que já discutimos, como a multiplicação de clock, modo real e modo protegido, coprocessador aritmético, RISC, CISC e cache, entre outras, continuarão sendo utilizadas.

### **Pentium MMX**

Lançado no início de 1997, o MMX é muito parecido com o Pentium clássico na arquitetura. Foram porém adicionadas 57 novas instruções ao conjunto x86, que era o mesmo desde o 8086. As novas instruções visam a melhorar o desempenho do processador em aplicações multimídia e pro-

cessamento de imagens. Nessas aplicações, algumas rotinas podem ficar até 400% mais rápidas com o uso das instruções MMX. É necessário porém que o software adotado faça uso de tais instruções, caso contrário, não haverá nenhum ganho de performance.

Foi aumentado também o cache primário (L1) do processador, que passou a ser de 32KB o que o torna cerca de 10% mais rápido do que um Pentium clássico, mesmo em operações que não façam uso das instruções MMX.

O Pentium MMX pode ser encontrado em versões de 166, 200 e 233 MHz, todas usando barramento de 66 MHz.

A Intel lançou também modelos de MMX Overdrive, que podem substituir antigos processadores Pentium 120, 100 ou 75, simplesmente substituindo o processador. O problema é que esses processadores são mais caros e difíceis de encontrar, não sendo muito atraentes, em termos de custo-benefício. Caso a sua placa não ofereça suporte aos processadores MMX vale muito mais a pena trocá-la também.

Falando em suporte, muitas pessoas ainda tem muitas dúvidas sobre a instalação do MMX em placas mais antigas. A verdade é que na maioria delas o MMX não pode ser instalado devido ao seu duplo sistema de voltagem. No MMX, os componentes internos do processador, ou "core" funcionam utilizando voltagem de 2.8V, enquanto que os circuitos de I/O que fazem a ligação do processador com o meio externo continuam funcionando a 3.3V, como no Pentium Clássico.

Esse sistema duplo foi criado para diminuir o calor dissipado pelo processador. Acontece que placas mais antigas estão preparadas para fornecer apenas as voltagens de 3.3V e 3.5V utilizadas pelo Pentium Standard e VRE, sendo unicamente por isso incompatíveis com o MMX. Até podemos instalar um MMX nessas placas, setando a voltagem para 3.3V porém; esse procedimento é arRISCado, pois estaríamos obrigando o processador a trabalhar com uma voltagem bem superior à original. Isso fará com que o processador aqueça muito mais do que o normal, podendo danificá-lo. Você pode tentar minimizar isso melhorando a refrigeração do processador, mas de qualquer forma esse não é um procedimento recomendável.

Na verdade, qualquer placa que suporta o Pentium comum, poderia suportar também o MMX, pois o que muda são apenas os circuitos reguladores de voltagem, que além dos 3.3, e 3.5V, devem suportar a voltagem dual de 2.8 e 3.3V. As instruções MMX são apenas software, e não requerem nenhum tipo de suporte por parte da placa-mãe. Justamente por isso, todas as placas-mãe para MMX suportam também o Pentium clássico, bastando setar corretamente os jumpers que determinam a voltagem.

### **AMD K6**

O K6, concorrente da AMD para o Pentium MMX, apresenta vantagens e desvantagens sobre ele. O K6 possui um cache L1 de 64 KB, contra os 32 KB do MMX, porém, é capaz de executar apenas uma instrução MMX por ciclo de clock contra duas do concorrente, perdendo em aplicativos que façam uso dessas instruções. O coprocessador aritmético interno também é bem mais lento do que o encontrado nos processadores Pentium, por isso, o K6 perde também em aplicativos que façam muito uso de cálculos de ponto flutuante como a maioria dos jogos por exemplo.

Outro problema do K6 é o aquecimento exagerado apresentado por esse processador, que apesar de não oferecer problemas de operação, dificulta o overclock (overclock é um método para envenenar o processador, que vamos ver com detalhes no 13º capítulo deste livro).

Apesar das limitações, o K6 é mais veloz do que um MMX, de mesmo clock, em muitas aplicações. Usando o **Business Winstone 97**, famoso programa de benchmark para medir a performance do K6, obtemos os seguintes resultados:

Processador	Performance rodando o Windows 95	Performance rodando o Windows NT 4.0	Performance em aplicativos que façam uso das instruções MMX
K6 233 MHz	54.8	71.0	246.52
K6 200 MHz	51.9	67.6	214.46
K6 166 MHz	48.6	63.3	181.58
Pentium 200 MHz MMX	50.2	64.3	246.57

Podemos notar através do Benchmark que a performance do K6 em ambiente Windows é levemente superior à do MMX. Em aplicações MMX, porém, ele perde, sendo um K6 233 mais lento até mesmo do que um 200 MMX. Como dito anteriormente, ele perde também em programas e jogos que façam uso intensivo de cálculos de ponto flutuante, como o Quake 2, por exemplo.

Para aplicações de escritório como o Office, o K6 é uma boa opção, pois nessas aplicações ele é mais rápido do que o MMX, sendo bem mais barato.

A escolha entre esses dois processadores, depende da aplicação à qual o micro se destina. Para jogos ou edição de imagens, o MMX é melhor; enquanto que para aplicações mais corriqueiras, o K6 é superior (e mais barato).

Quanto mais elevada for a velocidade de operação de um processador, maior será a quantidade de calor gerado. Justamente por isso, os fabricantes procuram desenvolver novas tecnologias de fabricação, que permitam produzir chips com transistores cada vez menores, a fim de diminuir o consumo de energia e, conseqüentemente, a dissipação de calor. As primeiras versões do K6 utilizavam a técnica de produção de 0.35 micron, com transistores medindo 0.35 milésimos de milímetro, e utilizavam voltagem interna de 2.9 ou 3.2 volts com voltagem externa de 3.3V. Essas primeiras séries são chamadas de "modelo 6".

A partir da versão de 233 MHz, o K6 passou a ser produzido usando uma nova técnica de produção de 0.25 micron, o que garante uma dissipação de calor bem menor. Essas versões são chamadas de "modelo 7" e operam com voltagem menor, de apenas 2.2V.

Note que apenas as placas-mãe mais modernas oferecem a voltagem de 2.2V exigida pelos modelos mais recentes do K6. Ao comprar uma placa-mãe para esse processador, não se esqueça desse detalhe.

Todos os K6 de 166 e 200 MHz são produzidos usando-se a técnica de produção de 0.35 micron; enquanto que todos os processadores de 266 e 300 MHz o são pela técnica de 0.25 micron. O problema são os processadores de 233 MHz, pois estes foram fabricados com ambas as técnicas. Para reconhecer um ou outro, basta olhar a voltagem que está estampada

no processador, os de 0.35 micron usam voltagem interna de 2.9 ou 3.2 e voltagem externa de 3.3, enquanto que os modelos de 0.25 micron usam voltagem interna de 2.2V.

## **AMD K6-2**

A exemplo da Intel, ao incorporar as instruções MMX às instruções x86 padrão, a AMD incorporou 27 novas instruções aos seus processadores K6-2. Essas instruções são chamadas de *3D-Now!* e tem o objetivo de agilizar o processamento de imagens tridimensionais, funcionando em conjunto com uma placa aceleradora 3D. A exemplo das instruções MMX, é necessário que o software adotado faça uso do *3D-Now!*.

Além das novas instruções, os novos K6-2 trabalham com velocidade de barramento de 100 MHz e tem versões a partir de 300 MHz. Como o K6, ele é compatível com as instruções MMX, mas executa apenas uma instrução por ciclo de clock contra duas dos processadores Intel. Todos os K6-2 são fabricados usando-se a técnica de produção de 0.25 micron, o que garante uma menor dissipação de calor. Como os K6 modelo 7, o K6-2 utiliza voltagem de 2.2V.

Apesar de funcionar com bus de 100Mhz, o K6-2 também pode ser utilizado em uma placa-mãe mais antiga, que suporte apenas bus de 66 MHz. Neste caso, um K6-2 de 300 MHz, seria usado com bus de 66Mhz e multiplicador de 4,5x, perdendo um pouco em performance. Também é necessário que a placa-mãe suporte a voltagem de 2.2V usada pelo K6-2.

## **Cyrix 686MX**

O 686MX é o concorrente da Cyrix para o MMX da Intel. Como o K6, esse processador possui um cache L1 de 64 KB, e funciona usando o soquete 7. A performance em aplicações Windows é muito parecida com um K6 do mesmo clock, porém o coprocessador aritmético é ainda mais lento do que o que equipa o K6, tornando muito fraco o seu desempenho em jogos e aplicativos que façam uso intenso de cálculos de ponto flutuante.

Para aplicações de escritório como o Office, o 6x86 é uma ótima opção devido ao seu baixo preço, mas ele não é muito adequado caso o principal uso do micro seja para programas gráficos ou jogos.

O 686MX usa voltagem dual de 2.9 e 3.3V, sendo 2.9 para o núcleo do processador e 3.3 para os circuitos de I/O. Caso a placa-mãe não ofereça essa voltagem específica, podemos setar a voltagem para 2.8 e 3.3V, como no MMX, sem problemas.

O 686MX é encontrado nas versões PR150, PR166, PR200, PR233 e PR266. Note que nos processadores da família MX, o índice PR é diferente do clock do processador. O 686MX PR266, por exemplo, é vendido em versões com clock de 225 MHz (3x 75Mhz) e 233 (3x 66Mhz). O Índice PR serve apenas como um comparativo, dizendo que apesar do clock, o PR266 tem desempenho 33% superior a um Pentium MMX de 200 MHz.

## **Cyrix MII**

O MII nada mais é do que uma continuação da série 686MX, alcançando agora índices PR 300, 333 e 350, já sendo anunciado também o PR400. Como o 686MX, o MII utiliza voltagem dual de 2.9/3.3V, mas que pode ser setada para 2.8/3.3, sem problemas.

## **Cyrix Media GX**

O Media GX é um processador 6x86MX acrescido de circuitos controladores de memória e cache, assim como controladores de vídeo e som, que se destina ao mercado de PC's de baixo custo e principalmente a notebooks. Quando usado em computadores portáteis, o media GX traz a vantagem de consumir pouca eletricidade, proporcionando uma maior autonomia da bateria. Já os micros de mesa equipados com o media GX pecam por oferecerem poucas possibilidades de upgrade.

Por exigir uma placa-mãe específica, o media GX se destina somente aos computadores de arquitetura fechada. Justamente por isso, você nunca irá montar um micro usando esse processador

## **Intel Pentium Pro**

O Pentium Pro utiliza o soquete 8 e exige uma placa-mãe específica. A principal vantagem desse processador sobre o Pentium comum é que nele o cache L2 é embutido no processador e utiliza a mesma frequência que ele, o que garante um desempenho muito maior.

Justamente devido ao cache, o Pentium Pro era muito difícil de produzir, pois a complexidade do cache L2 resultava numa elevada taxa de rejeição. E, como no Pentium Pro o cache L2 está embutido no mesmo invólucro do processador, um defeito no cache L2 condenava todo o processador à lata de lixo. Esses problemas de fabricação contribuíam para tornar o Pentium Pro ainda mais caro.

Mais uma característica marcante do Pentium Pro é sua arquitetura otimizada para rodar aplicativos exclusivamente 32 bits, como o Windows NT. Rodando o Windows 95, ou sistemas 16 bits como o DOS ou o Win-

dows 3.x, ele apresenta uma performance pouco superior ou até mesmo inferior, em alguns casos, a um Pentium clássico do mesmo clock, sendo indicado apenas para servidores. Ele executa 3 instruções contra duas do Pentium comum por ciclo de clock, mas não é compatível com as instruções MMX. Usando o Business Winstone para medir a sua performance, obtemos os seguintes resultados:

<b>Processador</b>	<b>Performance rodando o Windows 95</b>	<b>Performance rodando o Windows NT 4.0</b>
Pentium MMX 200 MHz	50.2	64.3
Pentium Pro 200 MHz	52.4	71.2

Para uso doméstico não faria muito sentido o uso de um Pentium Pro, porém num servidor, o cache L2 funcionando na mesma velocidade do processador faz muita diferença, pois o processamento de dados nestas máquinas é muito repetitivo. Tanto que mesmo com o surgimento do Pentium II, onde o cache L2 apesar de ser de 512 KB funciona a apenas metade da velocidade do processador, muitos ainda preferem continuar usando o Pentium Pro, pois além do cache, ele oferece recursos interessantes para uma máquina servidora como a possibilidade de usar até quatro processadores em paralelo (o Pentium II é limitado a dois processadores), além da maior quantidade de memória suportada, recursos que só foram superados pelo Pentium II Xeon.

O Pentium Pro foi produzido em versões equipadas com 256, 512 ou 1204 KB de cache L2, e em velocidades de até 200 MHz.

## **Pentium II**

O Pentium II utiliza um novo tipo de encapsulamento e novo tipo de soquete, chamado de Slot One, o que exige uma placa-mãe específica para ele.

Isso não deixa de ser uma política predatória da Intel, pois como o Slot One foi criado e patenteado por ela, os outros fabricantes não podem usar essa tecnologia nos seus processadores.

O Pentium II apresenta 32 KB de cache L1 e 512 KB de cache L2 que é embutido no cartucho do processador. Ao contrário do L2 tradicional, que fica na placa-mãe, trabalhando na velocidade desta, o cache L2 do Pentium II trabalha à metade da velocidade do processador, o que melhora, e muito, o seu desempenho, pois no caso de um Pentium II de 266 MHz por exemplo, o L2 funciona a 133 MHz, ou seja, o dobro do barramento padrão de 66 MHz utilizado pela maioria dos outros processadores. O Pentium II incorpora também as instruções MMX, executando 3 por ciclo de clock, além de várias características encontradas nos processadores Pentium Pro.

Essas características o tornam bastante rápido em ambientes exclusivamente 32 bits, proporcionados pelo Windows NT por exemplo, sem perder em performance rodando aplicativos 16 bits ou híbridos como ocorre no Pentium Pro.

O Pentium II é produzido usando-se duas arquiteturas diferentes. As versões de até 300 MHz, utilizam a arquitetura Klamath, que consiste numa técnica de fabricação de 0.35 microm, muito parecida com a utilizada nos processadores Pentium MMX. Nas versões a partir de 333 MHz, é utilizada a arquitetura Deschutes de 0.25 microm, que garante uma menor dissipação do calor, ocasionando uma maior durabilidade e confiabilidade do processador. Vale lembrar também que no Pentium II não precisamos nos preocupar em setar corretamente a voltagem do processador, pois isso é feito automaticamente pela placa-mãe. Só por curiosidade, os processadores baseados na arquitetura Klamath utilizam 2.8volts, enquanto os baseados na arquitetura Deschutes utilizam 2.0volts.

Uma última consideração a respeito dos processadores Pentium II é sobre a velocidade de barramento utilizada pelo processador. As versões do Pentium II de até 333 MHz, funcionam usando barramento de 66 MHz; enquanto que as versões a partir de 350 MHz utilizam barramento de 100 MHz, o que acelera a troca de dados entre o processador e as memórias. Vale lembrar que apenas as placas-mãe equipadas com chipsets i440BX ou i440GX suportam essa velocidade de barramento.

### **Celeron**

A fim de abocanhar também uma fatia do mercado de PC's de baixo custo, onde estava perdendo terreno para o K6 e o 6x86MX, a Intel lançou uma versão de baixo custo do Pentium II, batizada de Celeron, do Latin "Celerus" que significa velocidade. O *Celeron* nada mais é do que um Pentium II desprovido do Cache L2 e do invólucro metálico, e foi produzido em versões de 266 e 300 MHz.

O Cache L2 é um componente extremamente importante nos processadores atuais, pois apesar da velocidade dos processadores ter aumentado mais de 1000 vezes nas últimas duas décadas, a memória RAM pouco evoluiu em velocidade. Isto aconteceu pois é muito mais difícil criar memórias mais rápidas a preços acessíveis, do que processadores rápidos. Pouco adianta um processador veloz, se a todo instante ele tem que parar o que está fazendo para esperar dados provenientes da memória RAM. É justamente aí que entra o cache secundário, reunindo os dados mais importantes da memória, para que o processador não precise ficar esperando. Retirando o cache L2, a performance do equipamento cai em mais de 30%. Justamente por isso, além de perder feio para o seu irmão mais velho, o Celeron perde até mesmo para processadores mais antigos, como o MMX, o K6 e o 6x86 MX. De fato, um Celeron de 266 MHz perde até mesmo para um 233 MMX, na maioria das aplicações.

Um ponto a favor do Celeron é seu coprocessador aritmético que, sendo idêntico ao do Pentium II, é muito mais rápido do que o do MMX ou do K6, o que lhe garante um bom desempenho em aplicações gráficas. O Celeron possui também uma outra vantagem da qual podem tirar proveito os usuários mais corajosos: ele pode funcionar sem maiores problemas a 400 ou até mesmo 450 MHz, em uma placa-mãe que suporte Bus de 100 MHz, nesse caso estaríamos fazendo um Overclock no nosso Celeron. Trataremos com detalhes desse assunto no capítulo 13 deste livro.

### **Celeron A (Mendocino)**

Devido ao baixo desempenho, o Celeron não conseguiu uma boa aceitação no mercado. Por isso, a Intel resolveu equipar as novas versões do Celeron com 128 KBytes de cache L2 que, ao contrário do cache encontrado no Pentium II, funciona na mesma velocidade do processador.

Esses 128 KB de cache fazem uma diferença incrível na performance do processador. Enquanto que um Celeron antigo é mais de 30% mais lento do que um Pentium II do mesmo clock, o Celeron Mendocino é menos de 5% mais lento, empatando em algumas aplicações. Isso acontece pois, devido ao Mendocino possuir uma quantidade 4 vezes menor de cache, nele, este funciona ao dobro da velocidade.

Como o Celeron Mendocino possui apenas uma pequena quantidade de cache, a Intel resolveu incluí-lo no próprio núcleo do processador. Aliás, a única diferença visível entre o Celeron antigo e o Celeron Mendocino é justamente o tamanho maior da capa de metal que envolve o núcleo do processador.

### **Pentium Xeon**

O Xeon usa basicamente a mesma arquitetura do Pentium II, ficando a diferença por conta do cache L2, que no Xeon funciona na mesma velocidade do processador (como acontece no Celeron Mendocino e no Pentium Pro) sendo vendido em versões com 512, 1024 e 2048 KB de cache e (até o fechamento deste livro) em velocidades de 400 e 450 MHz. O Xeon foi especialmente concebido para equipar servidores, pois nesses ambientes o processamento é muito repetitivo, e por isso o cache mais rápido e em maior quantidade faz uma grande diferença, não fazendo, porém, muito sentido sua compra para uso doméstico, justamente devido ao seu alto preço. Outro recurso importante do Xeon é a possibilidade de se usar até 8 processadores numa placa compatível, o que criaria um sistema multiprocessado de incrível desempenho a um custo relativamente baixo.

### **Intel Merced**

O Merced será a próxima geração de processadores Intel. Desde o 8086 até o Pentium II, todos os processadores até aqui são baseados na mesma arquitetura x86, sendo compatíveis entre si. O Merced, porém, será o início de uma nova arquitetura de processadores, contendo instruções de processamento totalmente diferentes dos processadores x86.

Apesar de nativamente incompatível com qualquer programa antigo, ele incluirá um mecanismo de emulação via hardware que permitirá executar programas antigos nesse processador. Segundo a própria Intel, porém, perderá muito da performance executando tais aplicativos.

Pelo menos inicialmente, a idéia da Intel é criar um processador destinado a servidores de alto desempenho, mas não estão descartadas, no entanto, futuras versões para uso doméstico. Está previsto, apenas para o final do ano 2000, o lançamento dos primeiros processadores Merced, de modo que atualmente ainda se sabe pouco sobre eles. Boatos dizem que suas primeiras versões terão velocidade entre 800 e 1000 MHz, serão fa-

bricados usando uma técnica de 0.18 mícron e funcionarão com bus de 200 MHz.



## REFERÊNCIAS BIBLIOGRÁFICAS

STALLINGS, William. Arquitetura e organização de computadores: projeto para o desempenho. 5ed. São Paulo: Prentice-Hall, 2003.

TANENBAUM, Andrew S. Organização estruturada de computadores. 3ed. Rio de Janeiro: Prentice-Hall do Brasil, 1992.

HENNESSY, John L; PATTERSON, David A. Arquitetura de computadores: uma abordagem quantitativa. Rio de Janeiro: Campus, 2003.