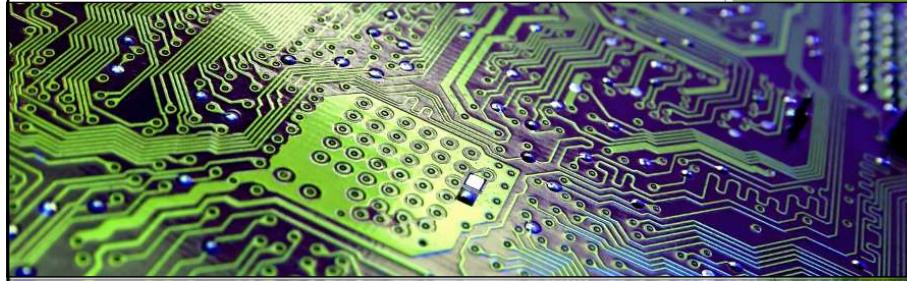


Organização e Arquitetura de Computadores

Aguardem!

 UNIFG
LAUREATE INTERNATIONAL UNIVERSITIES

1



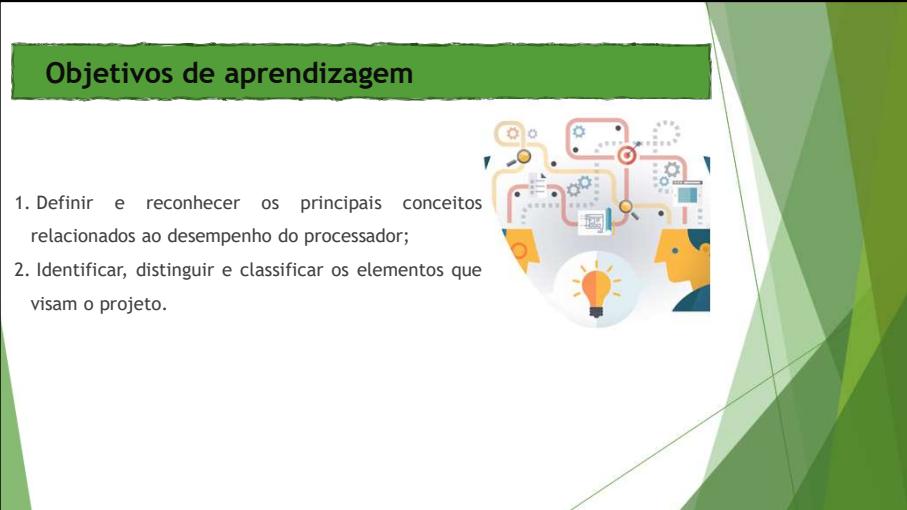
Organização e Arquitetura de Computadores

Componentes de um Computador - CPU

Prof. Marcelo Rabello
marcelo.rabello@unifg.edu.br

 UNIFG
LAUREATE INTERNATIONAL UNIVERSITIES

2



Objetivos de aprendizagem

- 1. Definir e reconhecer os principais conceitos relacionados ao desempenho do processador;
- 2. Identificar, distinguir e classificar os elementos que visam o projeto.



3



Componentes de um Computador



4

Componentes de um Computador

Computador Eletrônico Digital

➤ É um sistema composto por:

- **Memória** – Principal e Secundária
- **Unidade Central de Processamento (CPU–Central Processing Unit)** ou Processador
- **Unidades de Entrada/Saída (I/O – Input/Output Unit)**
- **Barramentos** (vias de interligação).

5

Componentes de um Computador

Memória

- Armazenar dados e programas (conjunto de instruções), permitindo o processamento automático dos dados.
- Para serem executados pela CPU, os programas devem estar armazenados na Memória (como veremos, Memória principal).
- Unidade básica é o Bit
- Diferentes maneiras de se organizar a memória
 - Hoje em dia o padrão é o byte
 - Cada byte representa um endereço de memória

6

Componentes de um Computador

Tipos de Memória

➤ Memória Principal (MP)

➤ Limitada (armazena somente alguns programas e dados).

➤ É a memória de trabalho e nela são carregados todos os programas, e, dados usados pela CPU no momento.

➤ Geralmente o seu conteúdo será perdido uma vez que o computador seja desligado.

➤ Ex.: RAM (*Random Access Memory*)

7

Componentes de um Computador

Tipos de Memória

➤ Memória Secundária (MS)

➤ Grande capacidade de armazenamento.

➤ Mais lentas do que a MP.

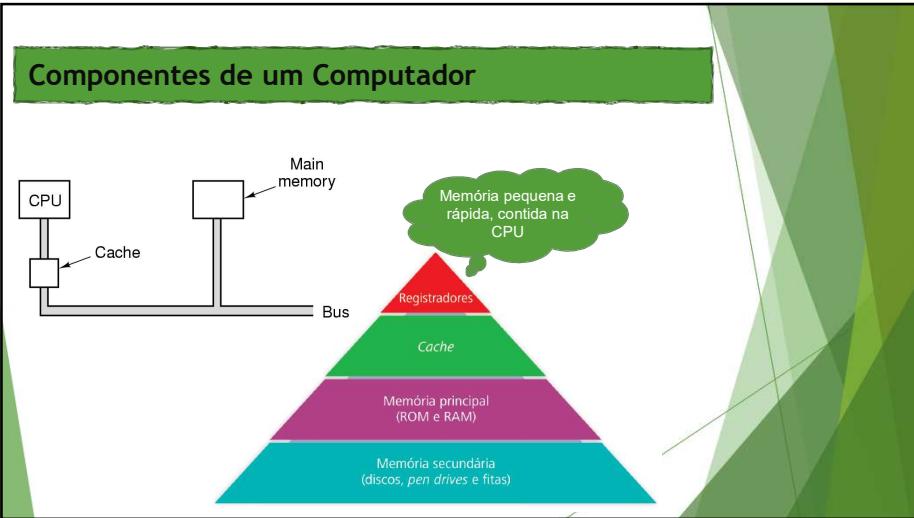
➤ Geralmente não-voláteis, permitindo guardar os dados permanentemente.

➤ Ex. HD - Hard Disk, SSD, CDs, DVDs, Pendrives, etc...

➤ E a Memória Cache?

8

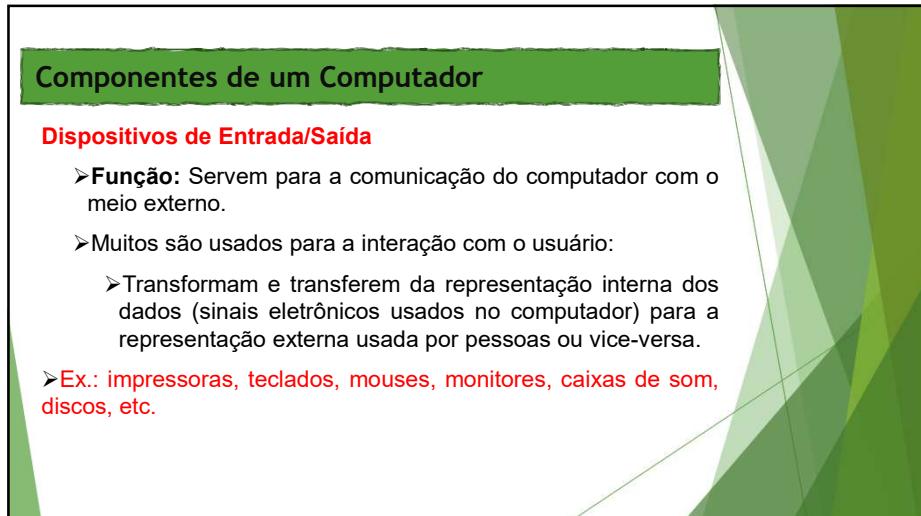
2



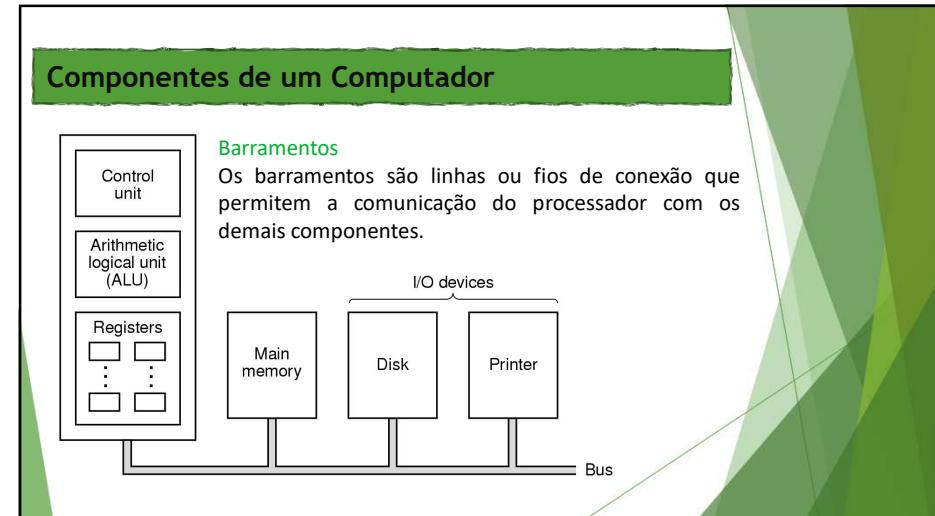
9



10



11

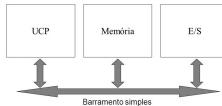


12

Componentes de um Computador

Barramentos (Bus)

- **Função:** interconectar os demais elementos básicos.



➤ Nesse modelo, é possível transferência direta de dados entre os dispositivos de E/S e a memória.

➤ Transferência feita por dispositivos especiais:

➤ Controlador de **DMA (Direct Memory Access)** ou processador E/S;

➤ Assumem momentaneamente o controle do barramento e “isolam” a comunicação CPU com a memória e dispositivos E/S.

13

Componentes de um Computador

Barramentos:

➤ **Unidirecionais:** conduzir os sinais em uma única direção;

➤ **Bidirecionais:** conduzir os sinais em ambas direções de acordo com comandos eletrônicos.

Tipos de barramentos:

➤ **Barramento de Dados:** através dele trafegam os dados que são transmitidos ou recebidos pelo processador (**bidirecional**). Os dados transmitidos podem ser enviados para a memória ou para um dispositivo de saída. Eles podem também ser recebidos da memória, ou de um dispositivo de entrada.

14

Componentes de um Computador

Tipos de barramentos:

➤ **Barramento de Endereços:** Conduz o endereço a ser selecionado na memória ou dispositivos E/S.

➤ É geralmente **unidirecional** entre a CPU (que define o endereço) e a memória e os dispositivos E/S.

➤ **Barramento de Controle:** transfere os sinais de controle(**bidirecional**) que ativam ou desativam os dispositivos, que selecionam determinado modo de operação ou sincronizam os circuitos.

➤ Formam a **Arquitetura de Barramentos**

15

Componentes de um Computador

Os principais sinais do barramento de controle do processador são:

MIO – indica se a operação se refere à memória ou a E/S.

RD – indica se a operação é uma leitura.

WR – indica se a operação é uma gravação.

INT – este sinal é uma entrada que serve para que dispositivos externos sinalizem o processador, interrompendo o processamento(suspender) para que seja realizada uma tarefa fundamental para o funcionamento do mesmo.

NMI – este é um sinal de interrupção(condições catastróficas do hardware) especial para ser usado em emergências.

16

Componentes de um Computador

Os principais sinais do barramento de controle do processador são:

INTA – este sinal é reconhecimento de interrupção, no qual o processador indique que aceitou uma interrupção.

VCC – esta é a entrada de corrente elétrica (diversos pinos) que alimenta os circuitos internos do processador.

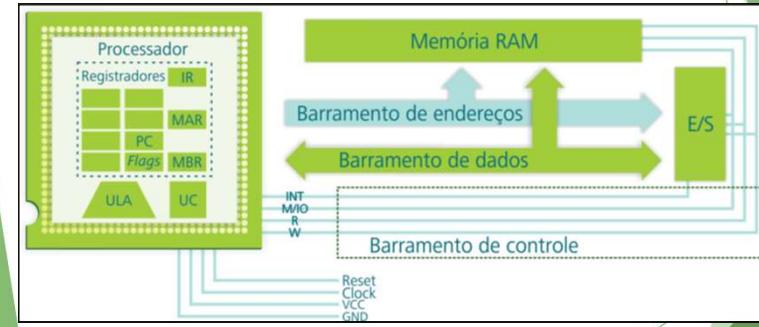
GND – significa *ground* ou terra, e é ligado ao polo negativo (diversos pinos) da fonte de alimentação.

Reset – este é um sinal que está ligado ao botão Reset do painel frontal do gabinete.

Clock – esta entrada recebe um sinal digital usado internamente para sincronizar todo o funcionamento do processador.

17

Componentes de um Computador



18

Unidade Central de Processamento - CPU



19

Unidade Central de Processamento - CPU

O processador ainda é o componente básico de qualquer computador



20

Como é feita uma CPU?



Quartzito

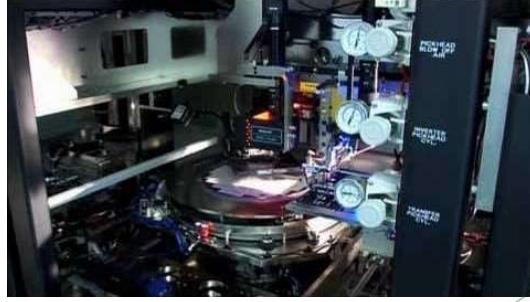
Maior dificuldade: Miniaturizar os projetos complexos e imprimi-los nos Wafers.

Si Si Si Si 109.5°

21

Como é feita uma CPU?

How microchips are made



<https://youtu.be/F2KcZGwntgg>

22

Unidade Central de Processamento - CPU

Conceitos básicos

- **Instruções:** comando que define integralmente uma operação a ser executada.
- **Programas:** instruções ordenadas logicamente "... Para que um programa possa ser executável ele precisa ser construído de uma série de instruções de máquina..."

O **processador** é o “cérebro” do computador

- **Função:** Executar os programas armazenados na memória principal, buscando cada uma das instruções do programa, examinando-as, e executando-as uma após a outra.

23

Unidade Central de Processamento - CPU

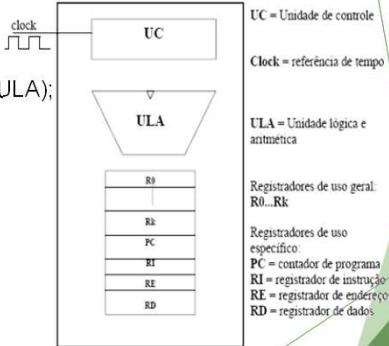
- Gera sinais de controle para os demais componentes do computador, além de reconhecer e responder sinais externos (**interrupções**).
- A CPU é responsável pela **supervisão** e **controle** de todo o **funcionamento** da máquina e pela realização das operações lógicas e aritméticas sobre os dados.
- **Clock** (ou Relógio) é a referência de tempo necessária a CPU.
- Circuito eletrônico oscilador que gera uma forma de onda quadrada, essencial para o sequenciamento das operações eletrônicas realizadas pela CPU.
- Está relacionado com a **frequência (taxa)** de operação do processador.

24

Unidade Central de Processamento - CPU

A UCP pode ser dividida em:

- Unidade de controle (UC);
- Unidade Lógica e Aritmética (ULA);
- Registradores.



25

Unidade Central de Processamento - CPU

Unidade de Controle (UC)

- Função: dirigir e coordenar as atividades das demais unidades do sistema.
- É a parte mais complexa da UCP (o centro nervoso).
- Controla todos os componentes e dispositivos de um computador, emitindo sinais elétricos (pulsos de controle):
 - Sincronizados através do clock.
- Responsável pela busca das instruções na memória principal, sua decodificação e execução:
 - Controla a ação da ULA e a movimentação de dados entre a UCP e a memória;
 - Solicita a busca da instrução que será executada, armazena-a em um **registrador**;
 - Dispara ações necessárias para execução da instrução.

26

Unidade Central de Processamento - CPU

Unidade de Controle (UC)

- Contador de Instrução (Program Counter – PC):
 - Armazena o endereço da (aponta para a) próxima instrução a ser executada.
 - Determina a sequência (fluxo) de execução de um programa, endereçando instrução por instrução.
- Registrador de Instruções (*Instruction Register* – RI):
 - Armazena (contém) a instrução que está sendo executada.

27

Unidade Central de Processamento - CPU

Unidade de Lógica e Aritmética (ULA)

- Possui os circuitos necessários para executar operações lógicas e aritméticas.
- Exemplos: Somas, subtrações, determinação de sinal, comparações algébricas de números, verificar se um número é maior ou menor que zero, etc...
- Os operandos utilizados pela ULA são armazenados em registradores internos da UCP
- O resultado das operações da ULA também é armazenado em um registrador
- **Acumulador (ACC)**

28

Unidade Central de Processamento - CPU	
EXECUÇÃO DAS INSTRUÇÕES	
INSTRUÇÃO	LOCAL DE EXECUÇÃO
Comece a aceitar entrada a partir da unidade de fitas nº 2	Unidade de Controle
Se X é maior que Y, vá para passo 46 do programa	Unidade lógica e Aritmética (Instrução Lógica)
Multiplique A por B e armazene o resultado em C	Unidade lógica e Aritmética (Instrução Aritmética)

29

Unidade Central de Processamento - CPU	
Conjunto (Banco) de Registradores	
➤ Memória interna de pequena capacidade de armazenamento, mas de alta velocidade, usada para armazenar resultados temporários e certas informações de controle.	
➤ Local onde os dados ficam armazenados enquanto são processados ou enquanto sujeitos a operações lógicas.	
➤ Formada por:	
➤ Registradores de uso geral : Acessíveis ao usuário no nível convencional de máquina;	
➤ Registradores de uso específico : Acessíveis diretamente apenas pela UC durante a execução de um programa.	
➤ O número de registradores e seus tamanhos variam entre arquiteturas de computadores	

30

Unidade Central de Processamento - CPU	
Conjunto (Banco) de Registradores	
➤ Contador de programa (PC) :	
➤ Aponta para as instruções a serem processadas;	
➤ Incrementado ao fim do processamento da instrução.	
➤ Registrador de instrução (RI) :	
➤ Armazena o código binário da instrução propriamente dita e modificado toda vez que uma nova instrução deve ser executada.	
➤ Registrador de endereço de memória (REM)	
➤ Mantém o endereço da posição de memória usada como operando e modificado quando é necessário apontar para um operando em outra posição de memória.	

31

Unidade Central de Processamento - CPU	
Instruções	
➤ São comandos atômicos fornecidos ao computador para que ele execute uma tarefa específica (soma, teste, desvio, etc...);	
➤ Uma série dessas instruções constituem um programa;	
➤ Cada computador (leia-se processador) tem um repertório particular de instruções, cuja rapidez de execução é um fator determinante na “potência de computação” (performance) da máquina;	
➤ Conforme proposto por Von Neumann, as instruções, tal qual os dados, devem residir na MP , e dispostas sequencialmente ;	
➤ A sequência de execução pode ser alterada por Instruções de Desvio .	

32

Unidade Central de Processamento - CPU

Formato das Instruções

Código de Instrução (OpCode)	Operando – Op (Informação Complementar)
---------------------------------	--

- **OpCode:** identifica a instrução a ser efetuada;
- **Op:** identifica sobre o que (quem) a instrução manipulará;
- Existem instruções de 1, 2 ou 3 operandos;
- O tamanho do **OpCode** indica o número máximo de instruções da máquina. n bits => até 2^n instruções;
- **Ciclo de Instrução:** conjunto de operações realizadas pelo computador para completar a execução de uma instrução.

33

Unidade Central de Processamento - CPU

Palavras são as unidades de dados movidas entre a memória e os registradores.

Instruções Registrador-Memória

- Permitem que o conteúdo de palavras de memória seja armazenado nos registradores (**load**) ou conteúdos de registradores armazenados na memória (**store**)
- Referências a uma palavra deve ser feita através de **Endereços**.

Instruções Registrador-Registrador:

- busca dois operandos em registradores;
- coloca-os nas entradas da ULA;
- realiza alguma operação sobre eles;
- armazena o resultado em um dos registradores;
- Determina um caminho de dados.

34

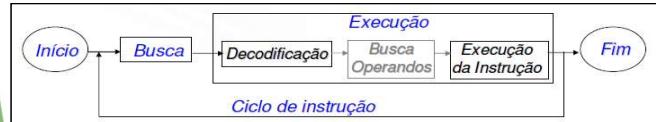
Unidade Central de Processamento - CPU

Ciclo de Instrução

- Um programa em execução é uma série de instruções de máquina que estão armazenadas na memória principal e serão interpretadas e executadas pela UCP.
- Conjunto de ações (operações) realizadas pelo computador (leia-se CPU) para completar a execução de uma instrução.

Diagrama do Ciclo de Instrução

- Ciclo de Busca (*Fetch Cycle*) e Ciclo de Execução (*Execution Cycle*).



35

Unidade Central de Processamento - CPU

Ciclo de Busca:

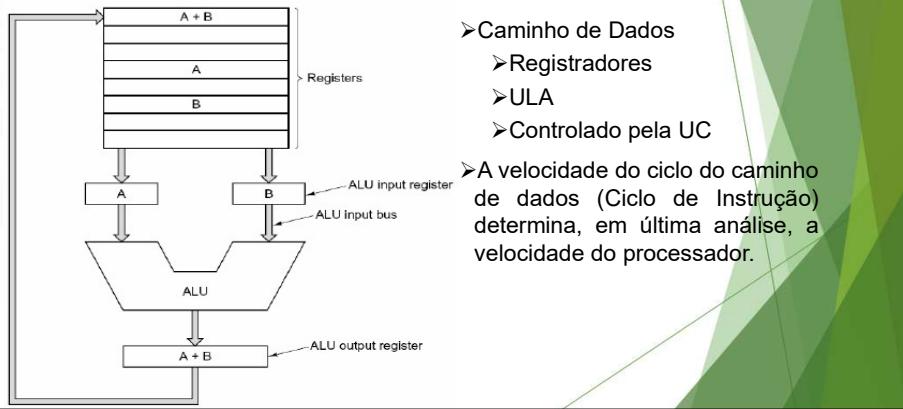
1. A UC lê (busca) a instrução na memória e carrega (armazena) a instrução em RI para ser decodificada e executada;
2. Atualização do valor de PC (incremento), fazendo-o apontar para a instrução seguinte;

Ciclo de Execução:

3. Determinação do **tipo de instrução** que está armazenada em RI (decodificação);
4. Determinação de **onde (endereço)** uma palavra de dado ESTÁ armazenada, caso necessário;
5. **Busca da palavra**, se necessário, e armazenamento em um dos registradores do processador;
6. **Execução da instrução**;
7. **Retorno ao passo 1** para iniciar a execução da instrução seguinte.

36

Unidade Central de Processamento - CPU



37

Unidade Central de Processamento - CPU

QUESTÃO

De acordo com o que foi passado, quais os fatores que afetam a velocidade de processamento de um computador?

- Tamanho da memória RAM e cache;
- Largura do barramento;
- Arquitetura interna do processador:
 - Tamanho e a quantidade de registradores;
 - Busca de paralelismo no nível das instruções!.

38

Paralelismo no Nível de Instruções

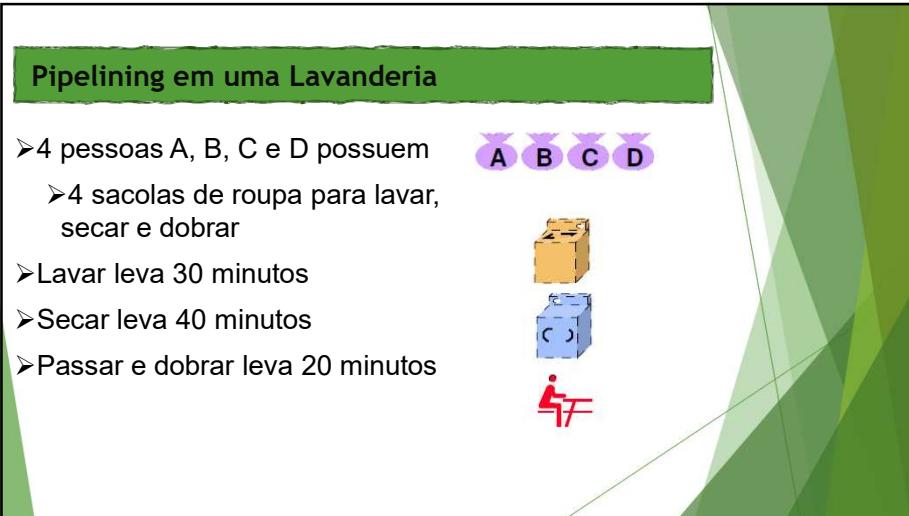
- Existem limites tecnológicos para desenvolvimento / desempenho / performance do hardware do chip de processamento.
- Solução para aumentar a velocidade do processador:
 - **Uso de Paralelismo**
 - **No Nível das Instruções:** Um único processador deve executar mais instruções por segundo;
 - **No Nível do Processador:** Vários processadores trabalhando juntos na solução do mesmo problema.

39

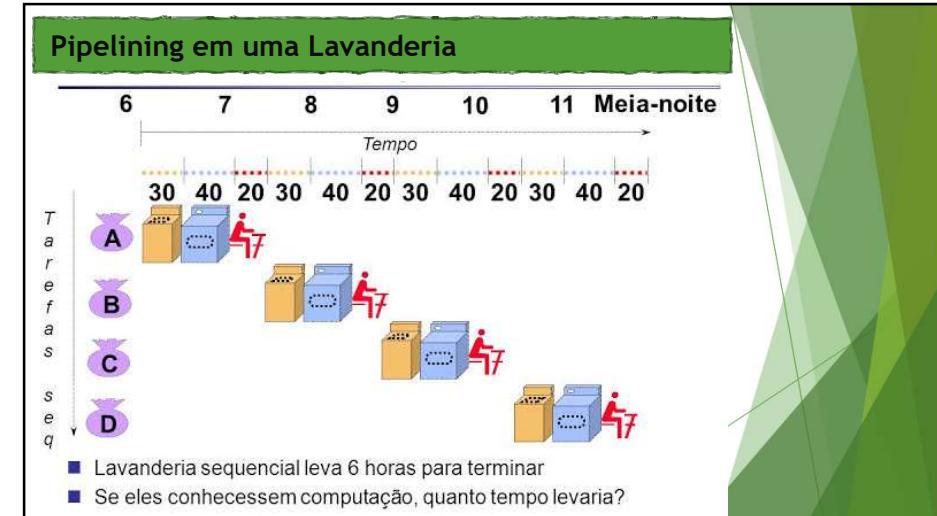
Paralelismo no Nível de Instruções

- Maior gargalo para a velocidade de execução de instruções é o acesso à memória:
- Execução em **Pipeline**:
 - O processamento em pipeline divide a execução de instruções em várias “partes”;
 - Cada “parte” é tratada por um hardware dedicado exclusivamente a ela.

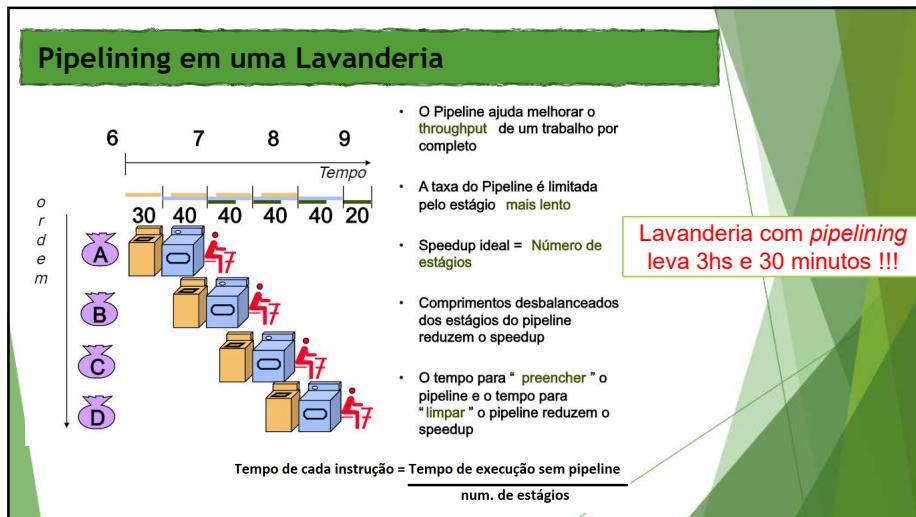
40



41



42

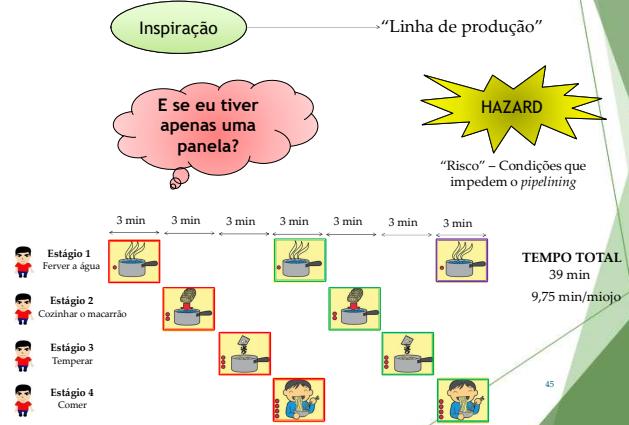


43



44

Pipelining na preparação de Miojo

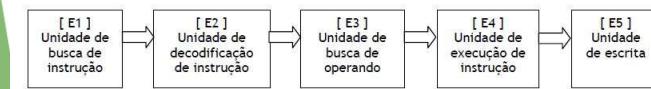


45

Paralelismo no Nível das Instruções

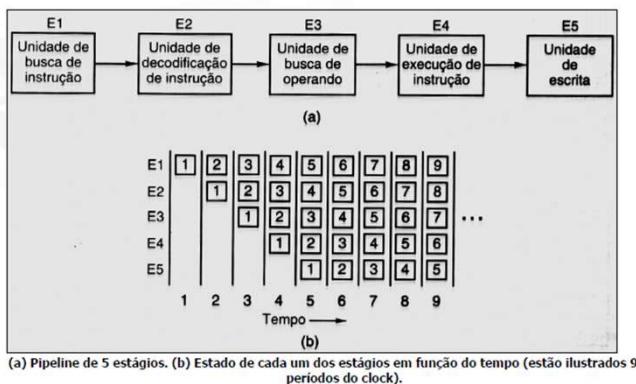
Pipeline de 5 estágios

1. Busca a instrução da memória e armazena em uma buffer até chegar a hora de executá-la;
2. Decodificação da instrução, determinando tipo e operandos;
3. Busca dos operandos na memória ou nos registradores;
4. Execução - passagem pelo caminho de dados;
5. Resultado do processamento é escrito em uma registrador.



46

Paralelismo no Nível das Instruções



47

Paralelismo no Nível das Instruções

A ideia básica do pipeline é a mesma de uma linha de produção em série.

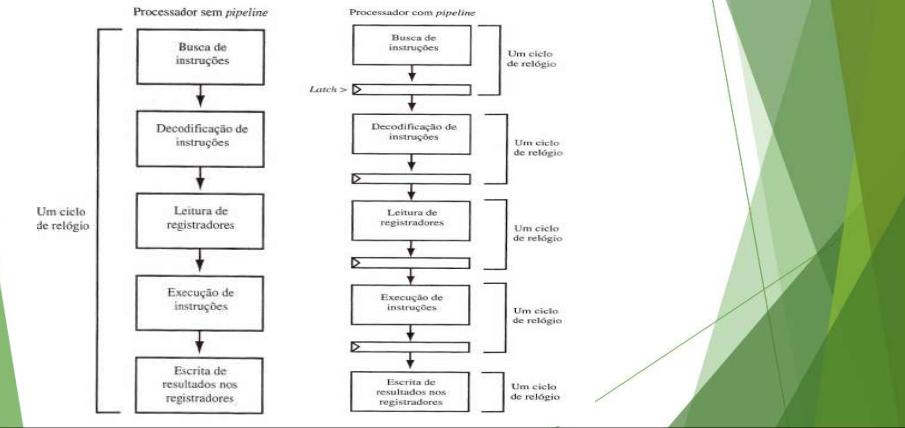
- Vários processamentos estão sendo executados ao mesmo tempo;
- Os estágios de cada processamento são aplicados a várias instruções ao mesmo tempo.

Exemplo:

- No tempo 1: a instrução 1 está sendo lida;
- No tempo 2: a instrução 1 está sendo decodificada enquanto que a instrução 2 está sendo lida;
- No tempo 3: a instrução 1 está buscando dados, a instrução 2 está sendo decodificada e a instrução 3 está sendo lida;
- ... e assim por diante.

48

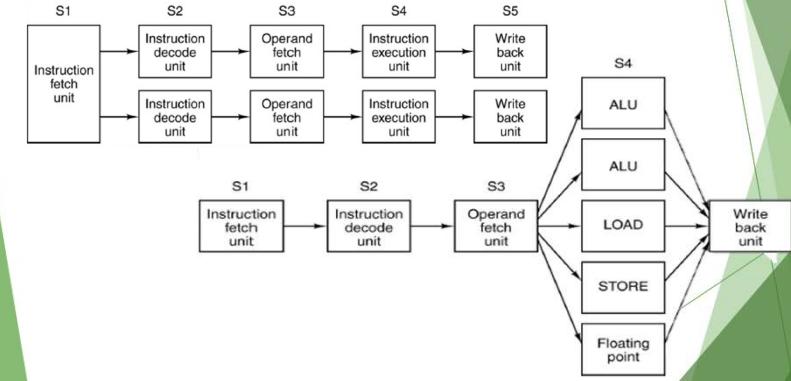
Paralelismo no Nível das Instruções



49

Paralelismo no Nível das Instruções

➤ Arquiteturas superescalares



50

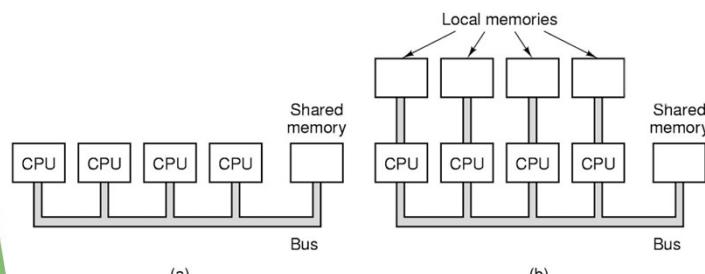
Paralelismo no Nível do Processador

Multiprocessadores

- É composto de vários processadores independentes;
- Compartilham uma mesma memória por um barramento principal:
 - OU compartilham uma memória e têm memórias locais.
- Executam processamentos locais;
- Liberam tráfego do barramento principal;
- É necessário gerenciar conflitos:
 - Ex: acesso ao mesmo end. de memória, dispositivo de E/S.

51

Paralelismo no Nível do Processador



(a) Multiprocessador organizado em torno de um único barramento.
 (b) Multiprocessador com memórias locais.

52

13

Paralelismo no Nível do Processador

Multicomputadores

- Sistemas (distribuídos) com um grande número de computadores interconectados;
- Não existe memória comum sendo compartilhada;
- Comunicação entre computadores é feita através de troca de mensagens a uma velocidade bem alta, usando os protocolos de comunicação.
- Computador não precisa estar ligado diretamente com todos os outros (uso de topologias em árvore, anéis, etc..).
- Existem também Sistemas Híbridos.

53

Paralelismo no Nível do Processador

Comparando o Desempenho de Dois Computadores

- Sejam X e Y dois computadores. X é mais rápido que Y se:

$$\frac{\text{Tempo de execução em X}}{\text{Tempo de execução em Y}} < 1$$

O desempenho de um computador X pode ser definido como:

$$\frac{1}{\text{Tempo de execução em X}}$$

A unidade de medida de desempenho computacional mais importante é o tempo.

54

Desempenho de Computadores

Medindo o Desempenho

- Em muitas situações, o tempo de execução de um programa não é fácil de se medir;
- Sistema Multiprogramado:
 - Sistema em que vários usuários ou programas compartilham a mesma CPU.
- Tempo de execução de um programa pode ser descrito por:

$$\text{User CPU time} + \text{System CPU time} + \text{I/O (Input/Output)}$$

Tempo de CPU de um programa

- = User CPU time: tempo de CPU gasto pelo usuário
- = System CPU time: tempo de CPU gasto pelo sistema
- = I/O (Input/Output) time: tempo gasto com entrada/saída

55

Desempenho de Computadores

Quais programas irão de fato testar o desempenho de um computador?
Como escolher esses programas?

Programas para Medir Desempenho

- **Programas Reais:** Aqueles usados no dia-a-dia (ex: compiladores);
- **Kernels:** Programas contendo apenas um trecho de código representativo (núcleo) de uma parte frequentemente executada por programas típicos (ex: multiplicação de matrizes);
- **Toy Benchmarks:** Programas reais bem simples;
- **Synthetic Benchmarks:** São programas artificiais. São criados apenas para gerar medidas de desempenho.

Os programas reais são os mais adequados para se medir desempenho.

56

14

Desempenho de Computadores

Benchmark Suites

- Benchmark Suite é um conjunto de programas de avaliação;
- Standard Performance Evaluation Corporation (SPEC) tem lançado vários benchmark suites: SPEC89, SPEC92, SPEC95 e SPEC2000;
- Esses benchmark suites são compostos por programas reais, escolhidos para serem representativos de programas que tipicamente demandam muita CPU e pouco I/O.

Exemplos de programas inteiros do SPEC2000:

gzip	Programa para compactar arquivos
gcc	Compilador C
vpr	FPGA Circuit Placement and Routing
craft	Game Playing: Chess
parser	Word Processing
vortex	Sistema de Banco de Dados
perlbench	PERL Programming Language

57

Desempenho de Computadores

Comparando e Sumarizando Desempenho

- Dúvidas ao comparar/sumarizar desempenho

	Computador A	Computador B	Computador C
Programa P1 (seg.)	1	10	20
Programa P2 (seg.)	1000	100	20
Tempo Total	1001	110	40

- Diferentes definições de “mais rápido”:
 - A é 10 vezes mais rápido que B para o programa P1.
 - B é 10 vezes mais rápido que A para o programa P2.
 - Se tomarmos o tempo total para executar P1 e P2 como medida de desempenho, C é o computador mais rápido; contudo, ele é 20 vezes mais lento que A na execução do programa P1.
- Melhor forma de sumarizar desempenho: **tempo total de execução**.

58

Desempenho de Computadores

Comparando o desempenho de um *benchmark suite*:

- Média Aritmética dos tempos de execução

Média aritmética de $Tempo(i,n) = \frac{1}{n} \sum_{i=1}^n Tempo_i$

- Média Harmônica

Média harmônica de $Tempo(i,n) = \frac{n}{\sum_{i=1}^n \frac{1}{Tempo_i}}$

	Computador A	Computador B	Computador C
Programa P1 (seg.)	1	10	20
Programa P2 (seg.)	1000	100	20
Tempo Total	1001	110	40
Média Aritmética	500,5	55,0	20,0
Média Harmônica	1,998	18,181	20,000

59

Desempenho de Computadores

Princípios Quantitativos para o Projeto de Computadores

- O principal princípio a ser seguido é: **Torne mais rápidas as tarefas mais frequentes**.
- **Lei de Amdahl**
 - A quantidade de melhoria de desempenho provida por um atributo da arquitetura é limitada pela quantidade de tempo que este atributo é utilizado.
 - Relaciona o **speedup** total de um sistema com o **speedup** de uma porção (atributo da arquitetura) do sistema.
- $Speedup = \frac{\text{Tempo de execução da tarefa inteira sem usar o novo atributo}}{\text{Tempo de execução da tarefa inteira usando o novo atributo quando possível}}$

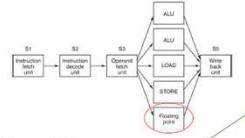
60

Lei de Amdahl - Exemplo

- Suponha que todas as instruções em ponto flutuante sejam melhoradas rodando 5 vezes mais rápido.
- O tempo de execução de alguns benchmark antes da melhoria no ponto flutuante é de 10 segundos.
- Qual será o aumento da velocidade (*speedUp*) se, antes da melhoria, metade dos 10 segundos eram gastos executando instruções em ponto flutuante?

$$T = Tf + To \quad \left\{ \begin{array}{l} Tf: \text{Tempo instruções em ponto flutuante} \\ To: \text{outras instruções} \end{array} \right.$$

$$\text{speedUp} = \frac{T_1 (\text{T sem melhoria})}{T_2 (\text{T com melhoria})} = \frac{Tf_1 + To_1}{Tf_2 + To_2} = \frac{5 + 5}{5/5 + 5} = \frac{10}{10} = 1,67$$



61

Desempenho de Computadores

Tempo de Execução de um Programa

- Tempo de CPU + I/O

➤ PS: por simplicidade chamaremos Tempo de execução o Tempo de CPU (desconsideraremos I/O):

Tempo de CPU de um Programa =

Número de Ciclos de Clock despendidos no Programa x Tempo de um Ciclo

Tempo de CPU de um Programa =

Número de Ciclos de Clock despendidos no Programa / Frequência de Clock

62

Desempenho de Computadores

Exercício:

Um programa roda em 10 segundos em um computador A, que tem um clock de 100MHz.

Um computador B gasta 1.2 vezes mais ciclos de clock para rodar o mesmo programa mas roda este programa em 6 segundos. Qual é a freqüência de clock do computador B?

Comp. A: $10s = n_{\text{ciclos}}/100\text{MHz}$

$$n_{\text{ciclos}} = 10s \times 100\text{MHz}$$

Comp. B: $6s = 1.2 \times n_{\text{ciclos}}/freq_B$

$$freq_B = 1.2 \times 10s \times 100\text{MHz} / 6s$$

$$freq_B = 200\text{MHz}$$

63

Desempenho de Computadores

Clocks per Instruction (CPI): É a média do número de ciclos por instrução.

➤ É comum calcularmos o desempenho da CPU em função do CPI:

➤ O tempo de execução de um programa, T, em uma determinada CPU:

$$T \text{ (segundos)} = N \text{ (instrução)} \times C \text{ (ciclo/instrução)} \times S \text{ (segundos/ciclo)}$$

➤ N é o número total de instruções executadas;

➤ C (ou clocks por ciclo de máquina – Clocks per Instruction (CPI)): é a média do número de ciclos por instrução;

➤ S é o número de segundos por ciclo.

➤ Quanto menor o tempo, maior o desempenho.

➤ N, C e S são afetados primariamente:

➤ pela capacidade de otimização do compilador;

➤ pela arquitetura do processador e de seu conjunto de instruções;

➤ pela tecnologia empregada na implementação da máquina.

64

Desempenho de Computadores

Exercício: Um projetista de compiladores está tentando decidir sobre qual sequência de instruções ele deve utilizar para implementar um comando de uma linguagem de alto nível. Ele dispõe das seguintes informações:

- Existem duas sequências de instruções possíveis para implementar o comando
- Sequência 1: 2 instruções A, uma B e duas C;
- Sequência 2: quatro A, uma B e uma C.
- Na máquina alvo, a instrução A tem o CPI de 1, a instrução B tem CPI de 2 e a C, de 3.
- 1 - Que sequência de código executa mais instruções?
- 2 - Qual sequência executa mais rápido?
- 3 - Qual é o CPI médio para cada sequência?

65

Desempenho de Computadores

Respostas:

- **Sequência 1:** 2 instruções A, uma B e duas C;
- **Sequência 2:** quatro A, uma B e uma C.
- Na máquina alvo, a instrução A tem o CPI de 1, a instrução B tem CPI de 2 e a C, de 3.
- 1 - Que sequência de código executa mais instruções?
Resp. seq1 = $2+1+2 = 5$ instruções; seq2 = $4+1+1 = 6$ instruções
- 2 - Qual sequência executa mais rápido?
Resp. seq1 = $2 \times 1 + 1 \times 2 + 2 \times 3 = 10$ ciclos; seq2 = $4 \times 1 + 1 \times 2 + 1 \times 3 = 9$ ciclos
- 3 - Qual é o CPI médio para cada sequência?
Resp. CPI(seq1) = 10 ciclos/ 5 instruções = 2 CPI; CPI(seq2) = $9 / 6 = 1.5$ CPI

66

Atividade Extraclasse

- * Novos processadores Intel e AMD para Servidores
 - Quantidade de Núcleos, Clock e Threads.
 - O que há de Novo na tecnologia?



67

Atividade Extraclasse

- Ler o capítulo 2 do livro Organização estruturada de computadores. 6º ed. São Paulo. Pearson Pratice Hall, 2013. (Disponível na Biblioteca Virtual - Person).

68

17



69



70