

Introdução à Organização de Computadores

Sistemas da Computação

Prof. Rossano Pablo Pinto, Msc.

rossano at gmail com

2 semestre 2007

Tópicos

- Processadores
- Memória Principal
- Memória Secundária
- Entrada e Saída

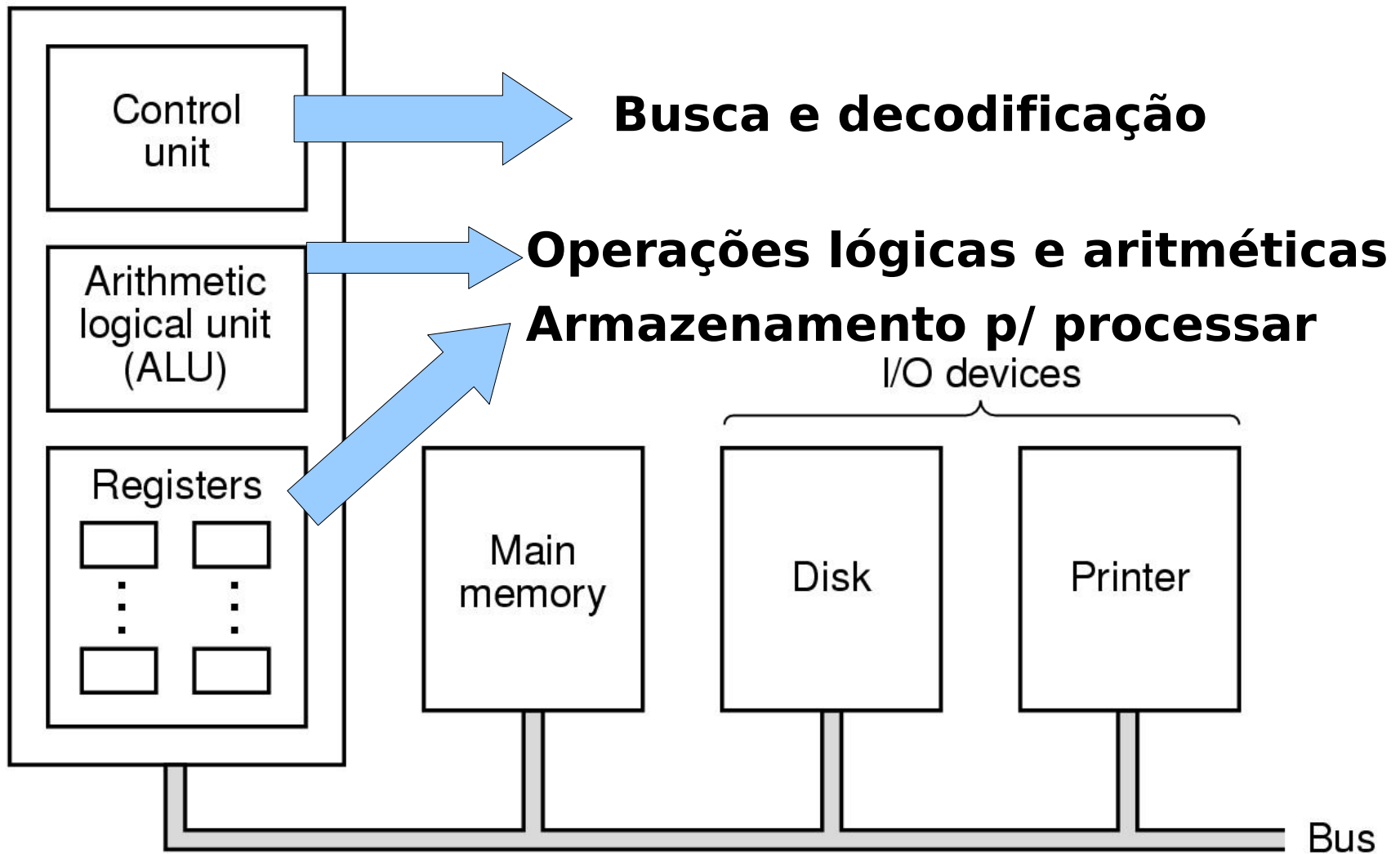
Processadores

Processadores

- CPU (Central Processing Unit) é o coordenador de todas as “atividades” de um computador (“cérebro”)
- É interligado aos outros elementos do computador a partir de um barramento:
 - Barramento: coleção de fios para transmitir **endereços, dados e sinais de controle.**
- Existem barramentos internos e externos ao processador

Processadores

Central processing unit (CPU)

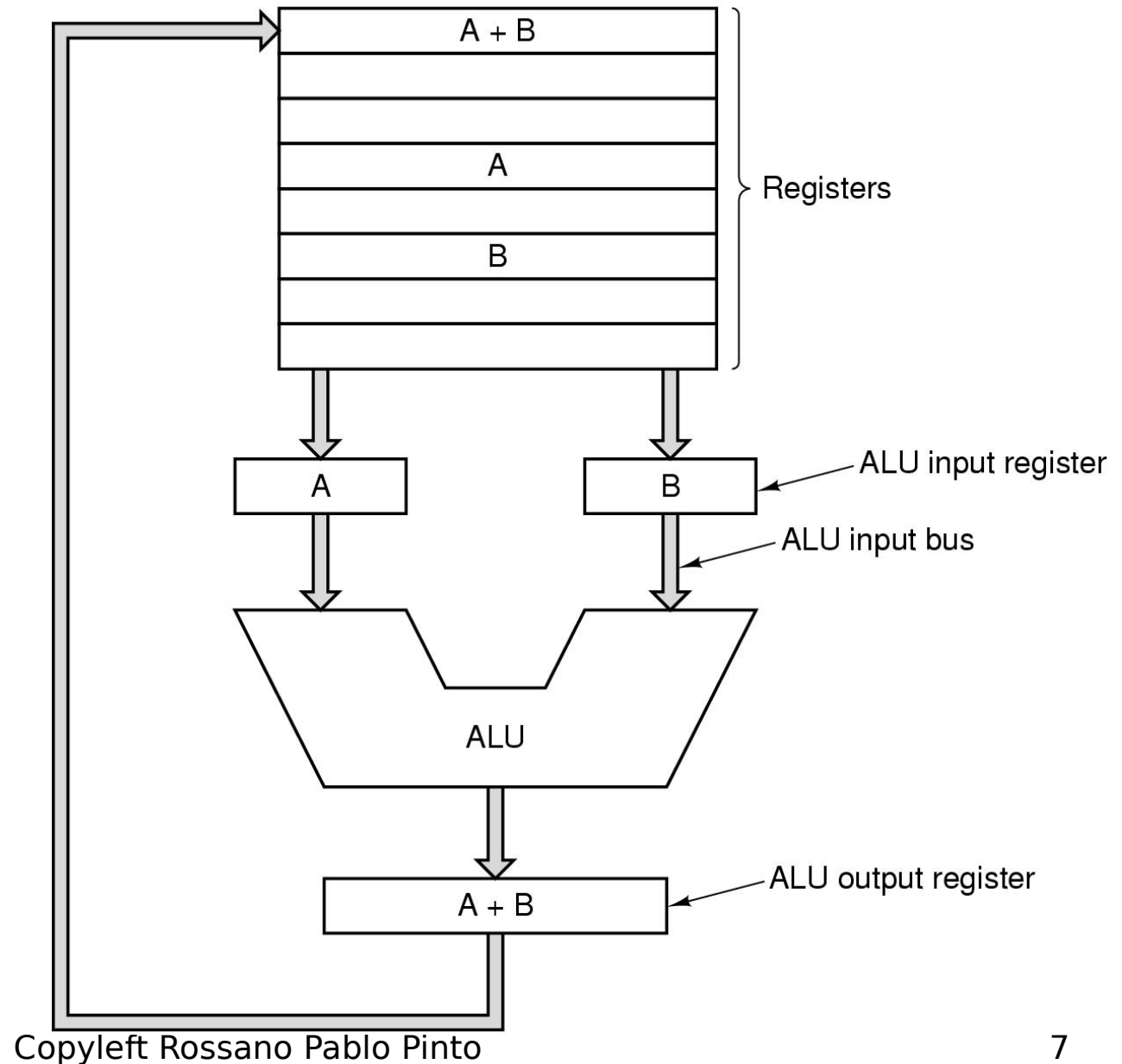


Processadores

- Função do processador: **executar instruções** obtidas a partir da memória principal
- A CPU é composta por diversas partes:
 - Unidade de Controle (UC)
 - Unidade Lógica e Aritmética (ULA)
 - Registradores (propósito geral e propósito específico)
 - PC (Program Counter), IR (Instruction Register)

Processadores

- “Data Path” da CPU de von Neumann
- Exemplo de adição de A e B



Processadores

- “Data Path” da CPU de von Neumann
 - registradores (tipicamente por volta de 32)
 - ULA (Unidade lógica e aritmética)
 - vários barramentos que conectam as partes da CPU
 - Registradores alimentam as entradas da ALU
 - Saída é armazenada no registrador de saída da ALU. O resultado pode ser escrito novamente em outros registradores

Processadores

- “Data Path” da CPU de von Neumann
 - A maioria das instruções pode ser divididas em 2 categorias:
 - registrador-memória
 - registrador-registrador
 - O processo de operar 2 operandos pela ULA e armazená-los de volta em algum registrador é conhecido por: **data path cycle** (“ciclo de caminho de dados”).
 - Quanto mais rápido o data path cycle, mais rápida a máquina.

Processadores

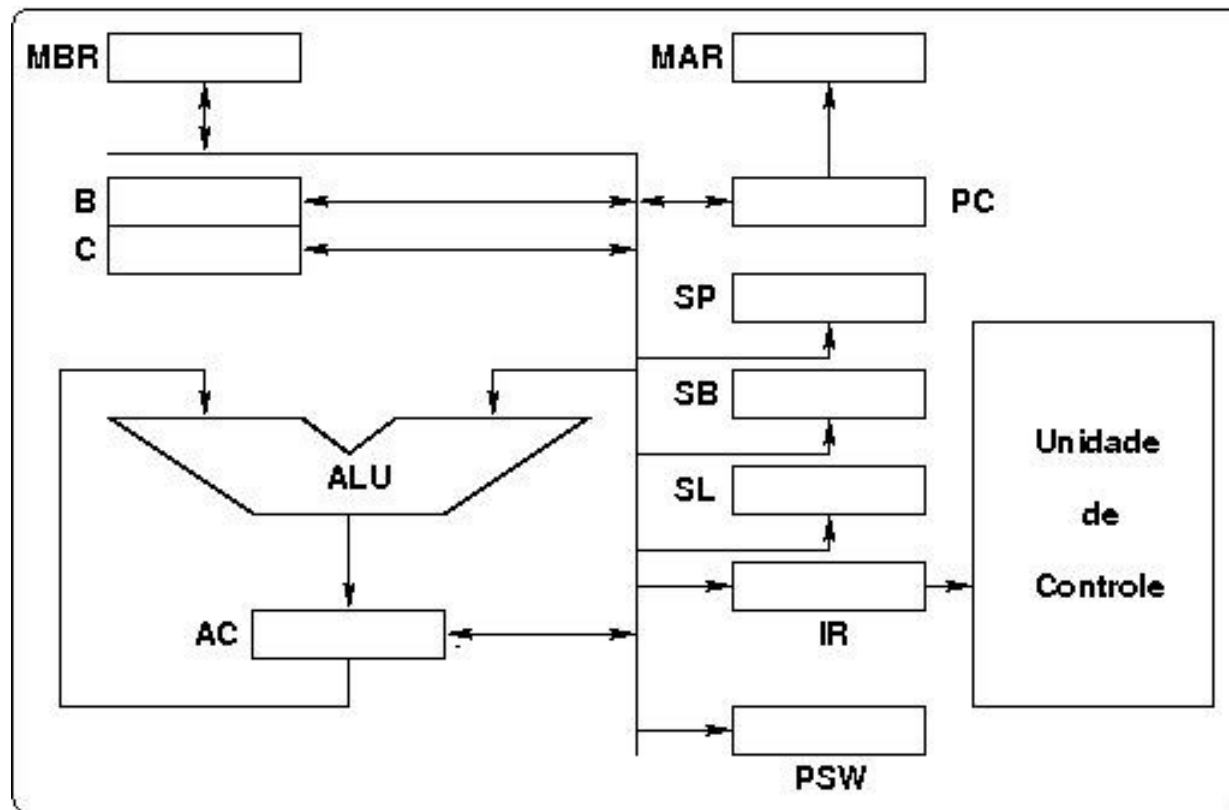
- Execução de instruções
 1. Busca próxima instrução da memória e armazena no IR
 2. Incrementa PC para “apontar” p/ a próxima instrução
 3. Determina o tipo de instrução (decode)
 4. Se instrução faz uso de operandos vindos da memória, determina seu endereço
 5. Busca operandos (palavra) da memória e armazena nos registradores

Processadores

- Execução de instruções
 6. Busca próxima instrução da memória e armazena no IR
 7. Executa a instrução
 8. Volta ao passo 1. para executar a próxima instrução

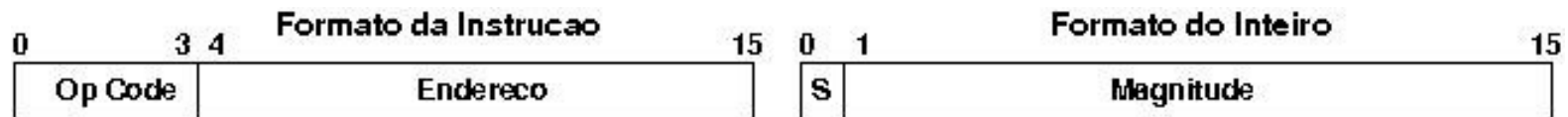
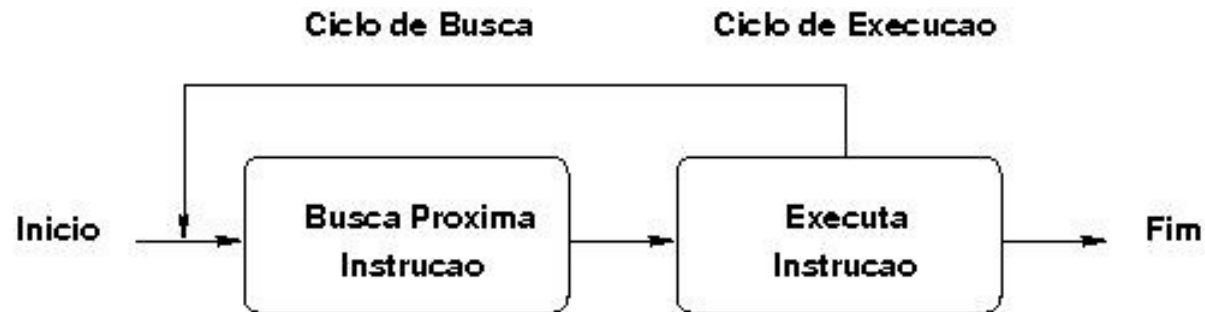
Processadores

- Exemplo de processador Hipotético



Processadores

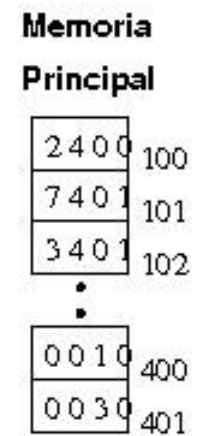
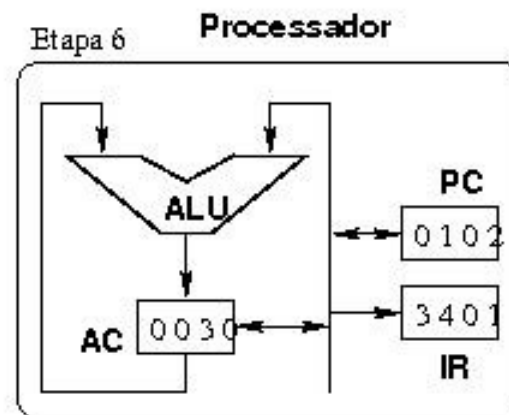
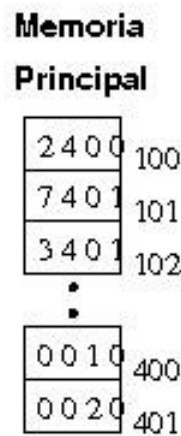
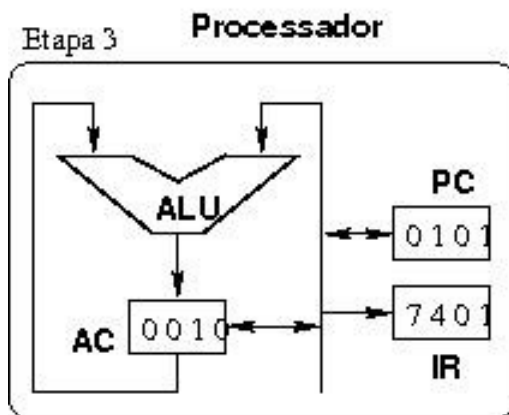
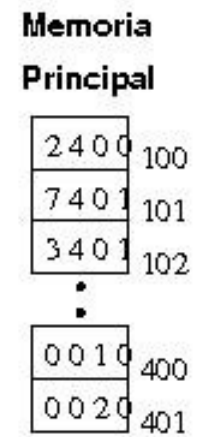
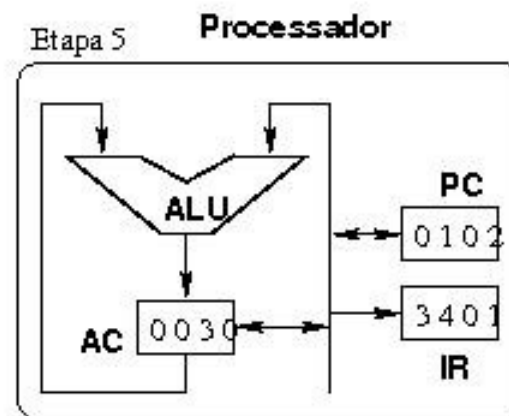
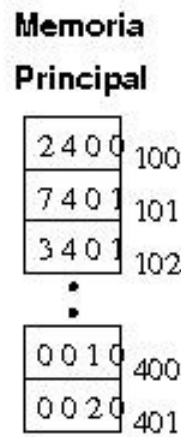
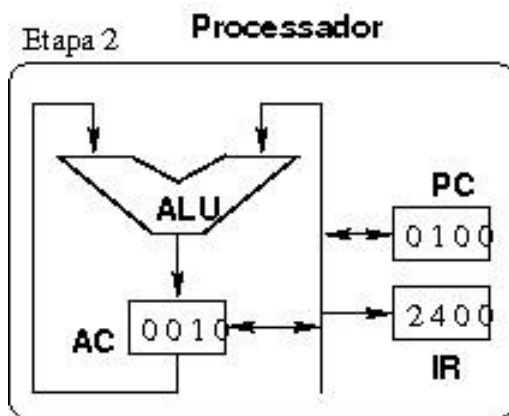
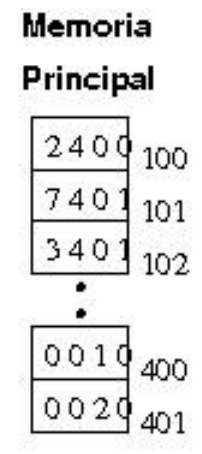
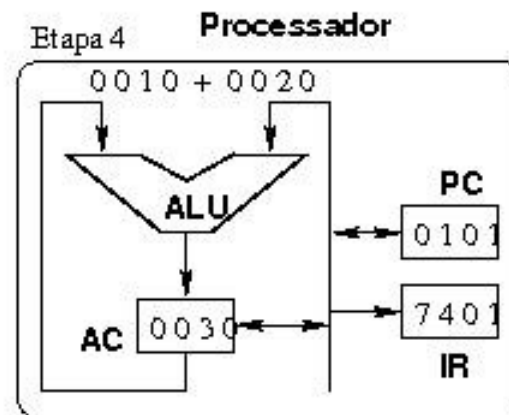
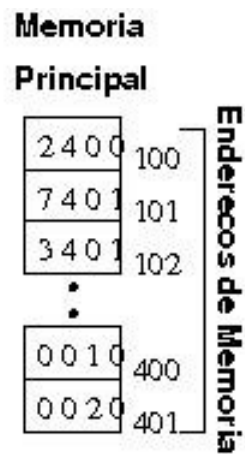
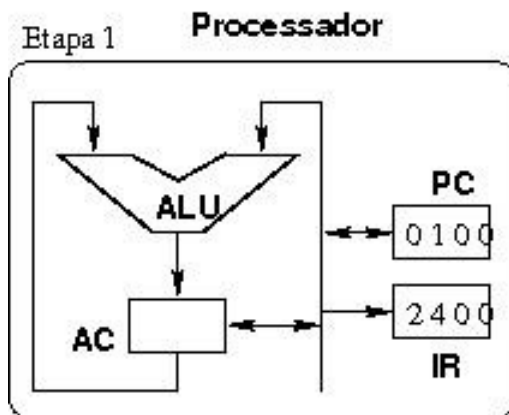
- Exemplo de processador Hipotético: ciclo de busca e execução, formato de inst.



Alguns Op Codes

- 0010 = Carrega AC da Memoria
- 0011 = Armazena AC na Memoria
- 0111 = Adiciona a AC valor da Memoria

Qual o resultado destas operações?



Processadores

- Implementação de Instruções via Hardware x Software
 - Hardwired (Hardware)
 - Microprogramado (software) – usa interpretação de instruções.
 - Ler seção 2.1.2 do livro Organização Estruturada de Computadores – AST 4a ou 5a edição

Processadores

- Instruções complexas
 - Uma das motivações: 2 ou mais instruções executadas sempre em conjunto podiam dar origem à uma única instrução com a mesma função
- Instruções simples
 - Uma das motivações: tempo de execução das instruções é quase sempre igual, fácil implementar tudo em hardware.

Processadores

- RISC x CISC (Ler Seção 2.1.3)
 - 1970s – muita experimentação com instruções complexas
 - minicomputador 801 foi na “contra-mão”. Possuía um projeto bem simples, voltado para alta-performance (não usava interpretação).
 - 1980 – Berkeley – Projeto de CPUs VLSI sem o uso de interpretação – Surgia o Termo RISC. **Reduced Instruction Set Computer.**

Processadores

- RISC x CISC (Ler Seção 2.1.3)
 - Processador RISC I e II tornou-se o SPARC
 - 1981 - MIPS (projeto diferente mas com idéias semelhantes) deu origem ao processador MIPS
 - Processadores Intel x86 à i386 são CISC
 - Processadores Intel i486 à atual são “híbridos” (núcleo RISC p/ instruções mais simples + interpretação p/ instruções mais complexas)

Processadores

- Princípios de projeto p/ computadores modernos (Seção 2.1.4)
 - Instruções comuns são diretamente executadas por hardware – instruções complexas são interpretadas
 - Maximizar taxa de “início” de instruções
 - Instruções devem ser fáceis de decodificar
 - Apenas LOADs e STOREs devem referenciar memória
 - Prover MUITOS registradores

Processadores

- Como aumentar a “velocidade” dos processadores?
 - Aumento do clock/freqüência do processador.
 - Problema: o limite é a velocidade da luz. O máximo que consegue-se no cobre ou fibra ótica é **20 cm/nanosegundo**.
 - Quanto maior a freqüência, maior o calor que tem que ser dissipado.
 - Duplicar unidades funcionais !!!

Processadores

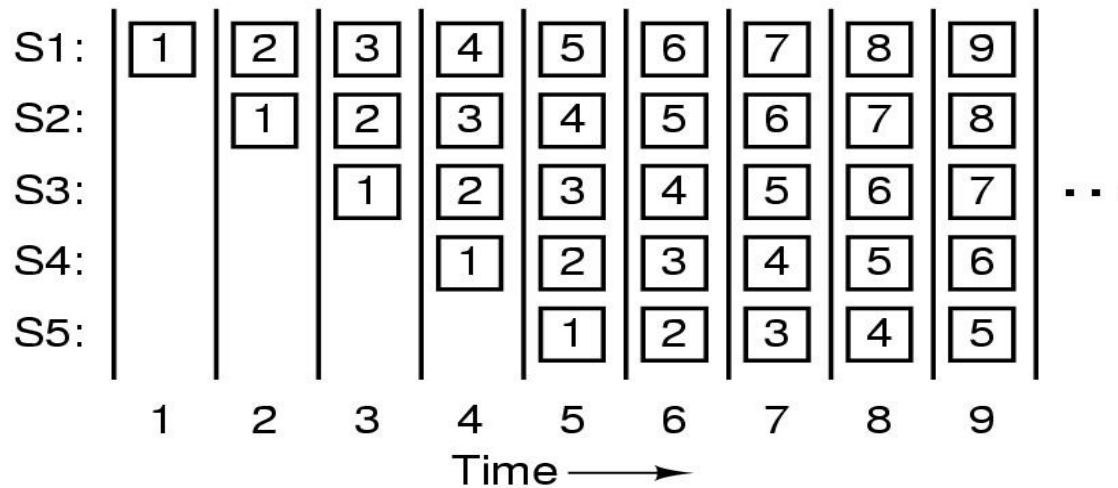
- Paralelismo por meio de Instruções (Seção 2.1.5)
 - Pipelining
 - Arquiteturas Super-escalares

Processadores

- Pipeline (ex.: 5 estágios e 9 ciclos de clock)



(a)



(b)

Processadores

- Pipeline (ex.: 5 estágios e 9 ciclos de clock)
 - Suponha um tempo de ciclo de 2 ns
 - Leva 10 ns p/ uma instrução ser executada
 - Qual o MIPS desta máquina? 100 MIPS ?
 - Não! 500 MIPS

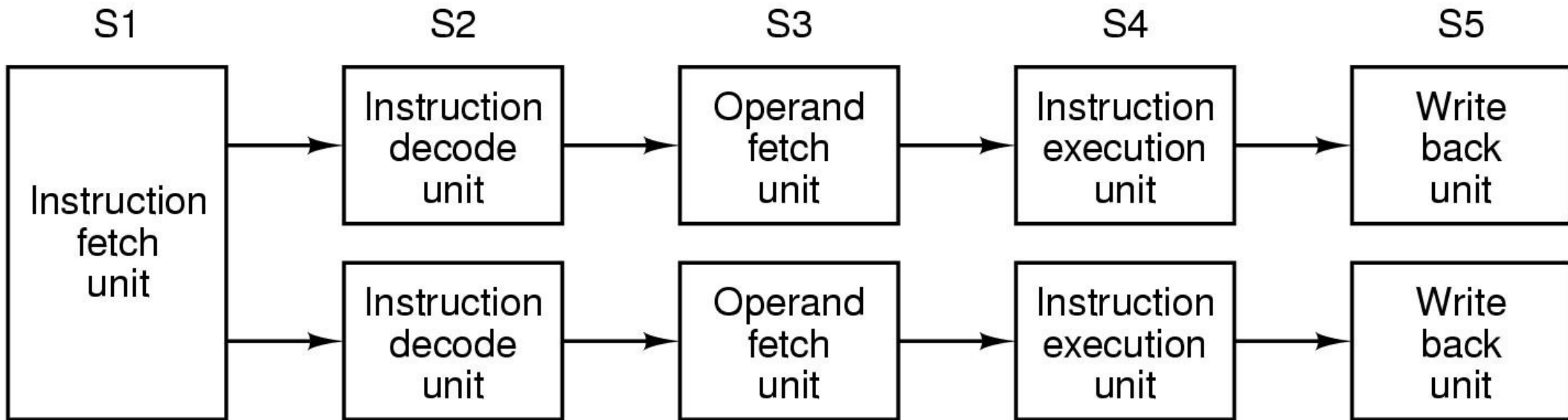
Processadores

– Pipeline

- Latência x largura de banda de processador
- **Latência** -> TEMPO que uma instrução leva para ser executada
- **Largura de banda do processador** -> quantos MIPS a CPU tem
- Exemplo de processador com pipeline: 486

Processadores

- Arquiteturas Super-Escalares



Pipeline duplo de 5 estágios com uma única unidade de busca

Processadores

- Arquiteturas Super-Escalares

- O exemplo anterior de 2 pipelines somente é possível se:
 - as 2 instruções não tiverem conflito em uso de recursos (ex.: registradores)
 - nenhuma delas pode depender do resultado da outra
- Exemplo de processador com pipeline duplo: Pentium (similar ao da figura anterior)

Processadores

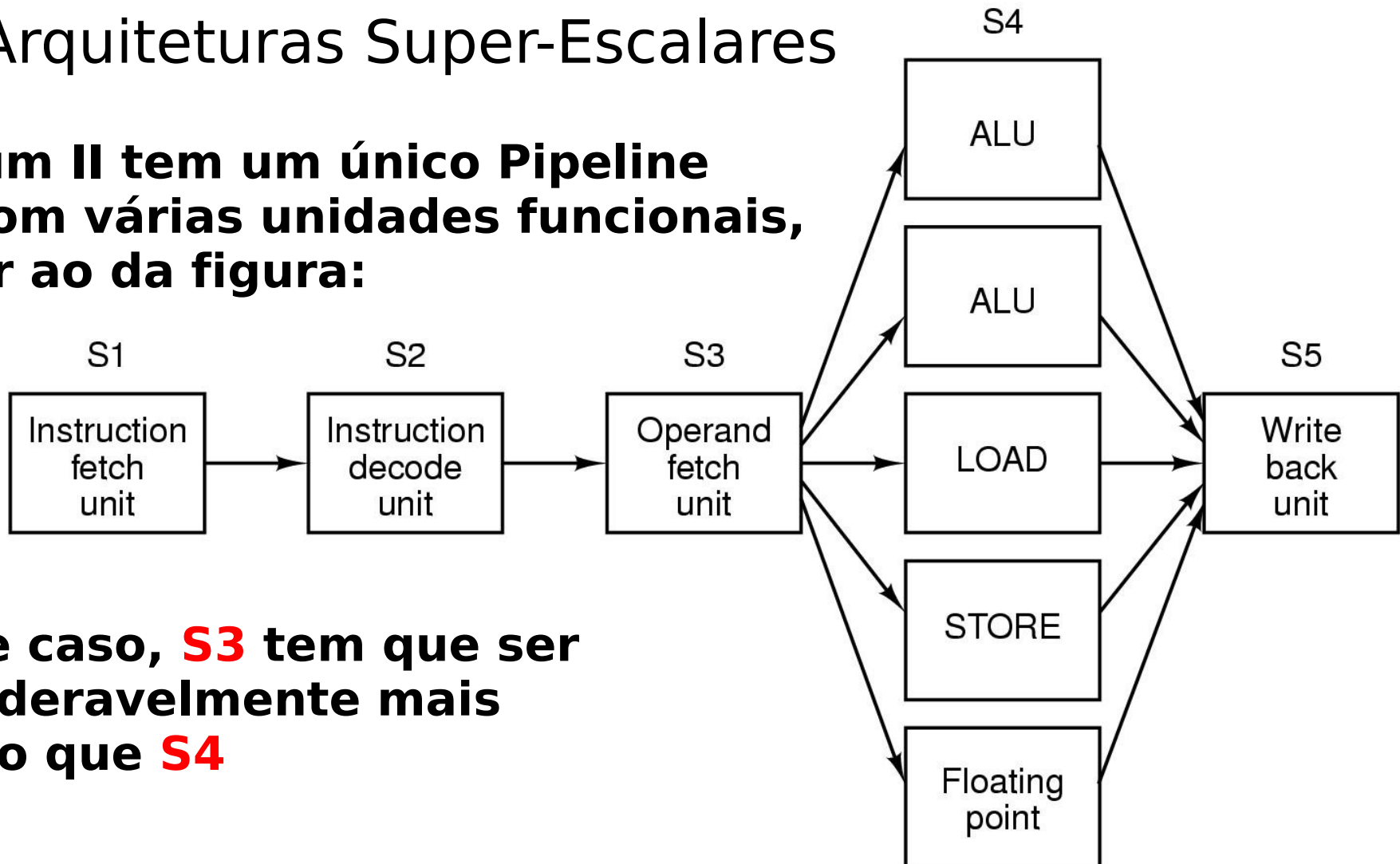
- Arquiteturas Super-Escalares

- Compiladores otimizados p/ Pentium podem gerar código 2 vezes mais rápido que os otimizados p/ 486.

Processadores

- Arquiteturas Super-Escalares

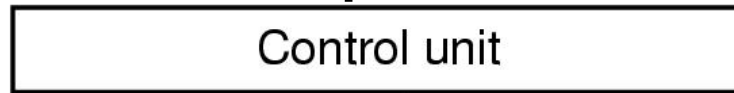
Pentium II tem um único Pipeline mas com várias unidades funcionais, similar ao da figura:



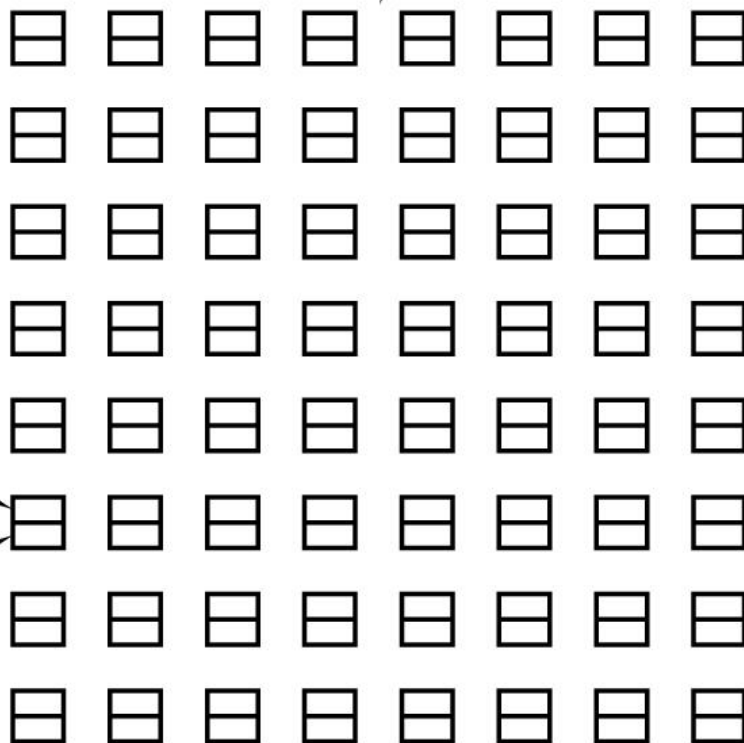
Neste caso, S3 tem que ser consideravelmente mais rápido que S4

Processadores

- Paralelismo por meio de Processadores



⚡ Broadcasts instructions



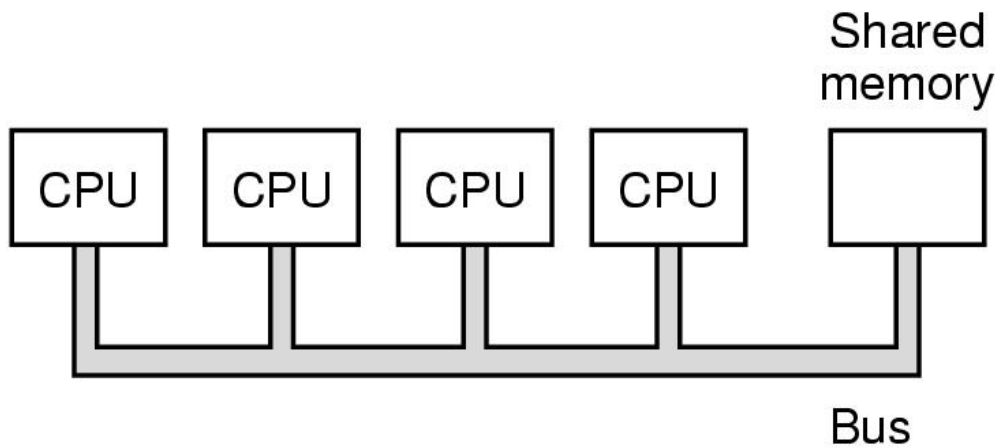
Computadores em Array: Conjunto de processadores idênticos que executam a mesma instrução em paralelo em dados distintos

8 × 8 Processor/memory grid

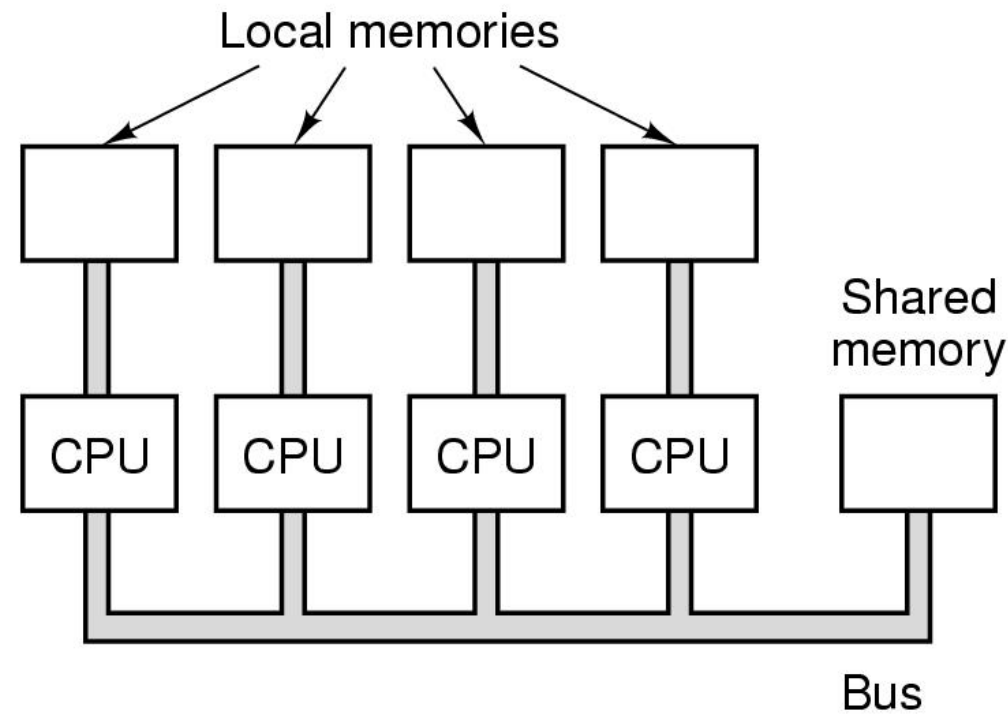
Processadores

- Paralelismo por meio de Processadores

Multi-processadores:
vários processadores
compartilhando mesma
memória. a) SMP
b) NUMA



(a)



(b)

Processadores

- Paralelismo por meio de Processadores

Multi-Computadores:
vários computadores
interligados por rede.