



# ORGANIZAÇÃO E ARQUITETURA DE COMPUTADORES II

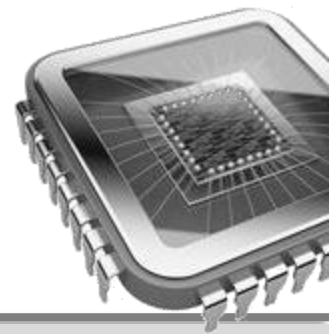
## AULA 09: PARALELISMO EM NÍVEL DE INSTRUÇÃO

Prof. Max Santana Rolemberg Farias

[max.santana@univasf.edu.br](mailto:max.santana@univasf.edu.br)

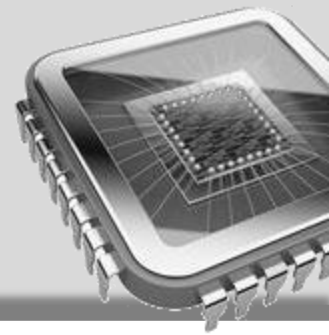
Colegiado de Engenharia de Computação



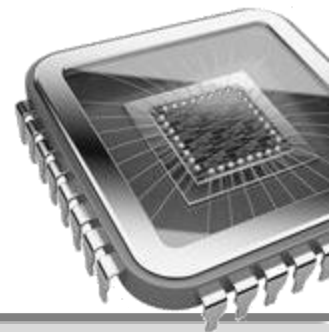


# **COMO MELHORAR A PERFORMANCE DE PROCESSADORES IMPLEMENTADOS COM PIPELINE?**

# PARALELISMO EM NÍVEL DE INSTRUÇÃO

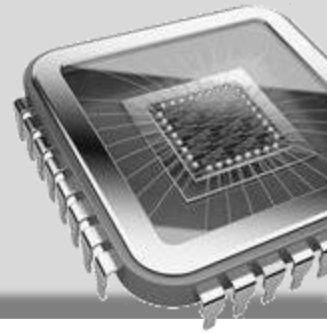


- Para fazer o paralelismo em nível de instrução foi primeiramente adicionado mais recursos de hardware.
  - Necessidades e restrições precisam ser observadas e controladas.
  - Processadores superescalares
  - Processadores com superpipeline
  - VLIW (Very Long Instruction Word)



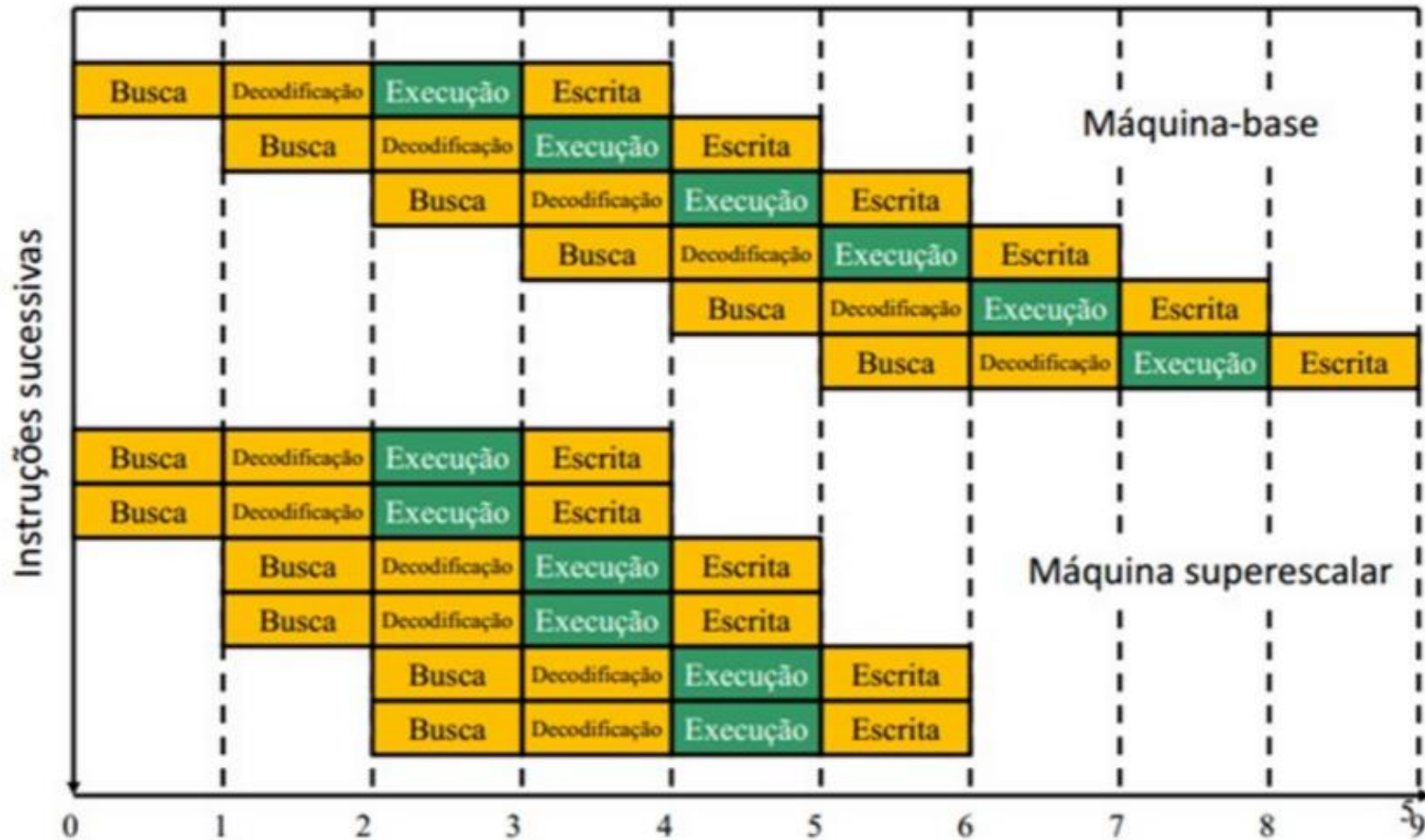
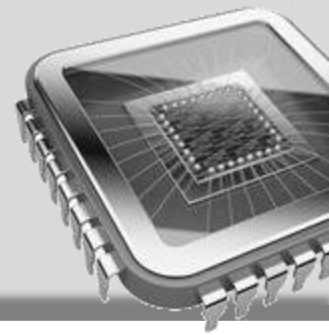
# O QUE SÃO PROCESSADORES SUPERESCALARES?

# PROCESSADORES SUPERESCALARES

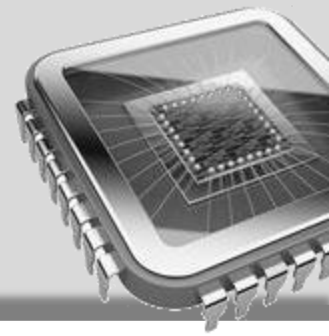


- Foram criados para otimizar a execução de tarefas, **trabalhando com mais instruções em um único ciclo de clock.**
- Utiliza múltiplos e independentes pipeline de instruções.
- Permite executar um número variado de instruções por ciclo.
- Cada pipeline consiste de múltiplos estágios e pode lidar com múltiplas instruções ao mesmo tempo.
- Quando ocorrem dependência entre as instruções, o processador pode executá-las fora de ordem.

# PROCESSADORES SUPERESCALARES

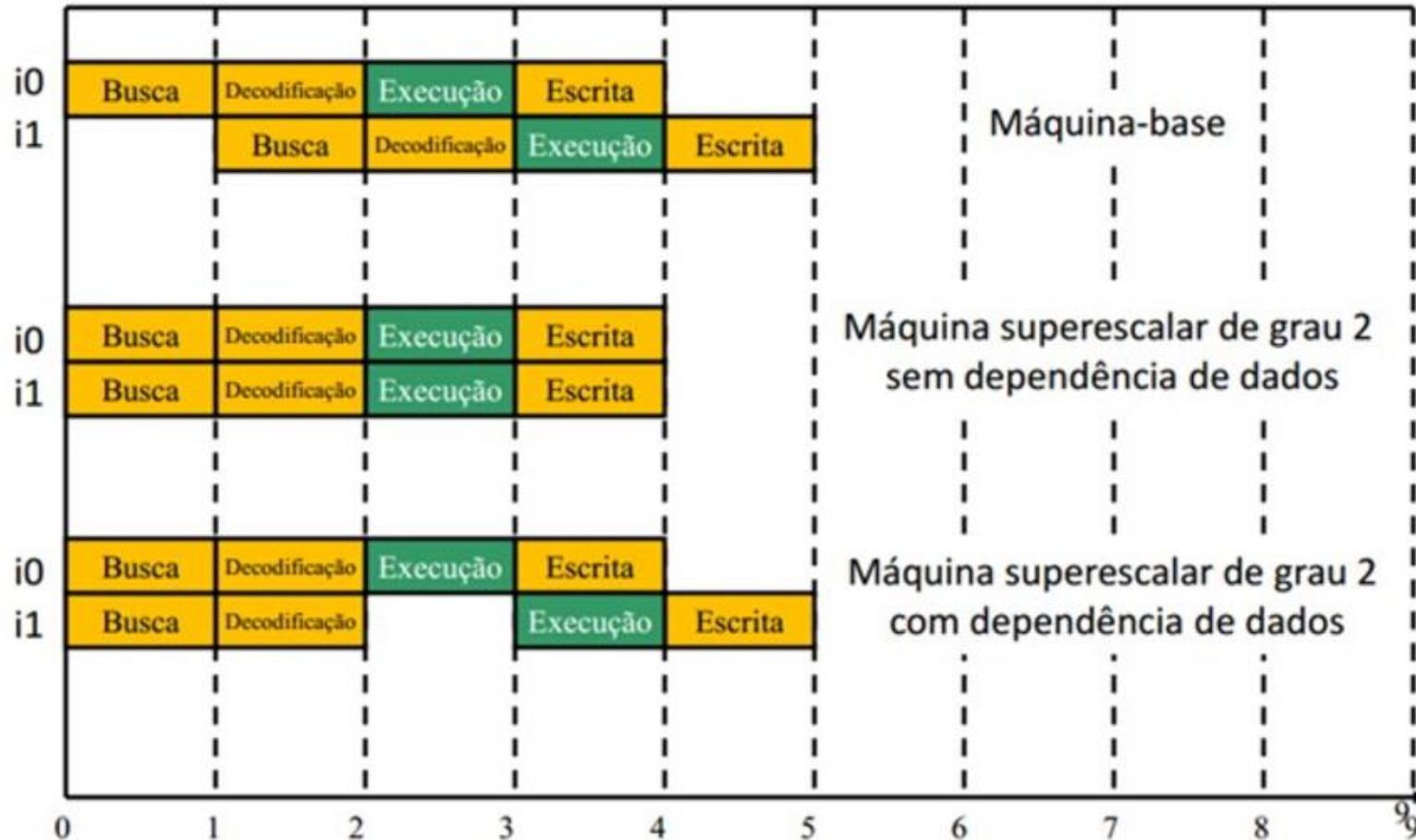
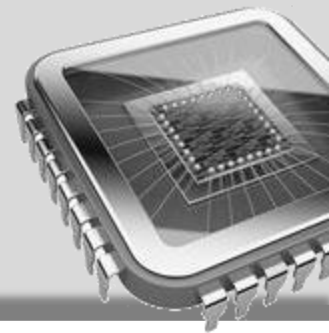


# PROCESSADORES SUPERESCALARES



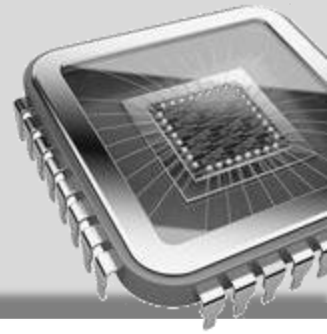
- Um processador superescalar precisa se preocupar com:
  - Dependência de dados
  - Dependência procedural
  - Conflito de recurso
  - Dependência de saída
  - Antidependência

# PROCESSADORES SUPERESCALARES



# PROCESSADORES SUPERESCALARES

## DEPENDÊNCIA DE SAÍDA



$$I_1: \mathbf{R3} = R3 + R5$$

$$I_2: R4 = R3 + 1$$

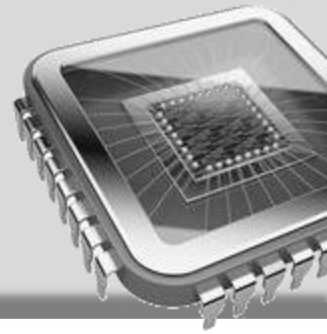
$$I_3: \mathbf{R3} = R5 + 1$$

$$I_4: R7 = \mathbf{R3} + R4$$

- A instrução 1 e 3 escrevem em R3
  - A atribuição da instrução 1 não pode ser feita após a atribuição da instrução 3.
  - A instrução 3 precisa ser congelada.

# PROCESSADORES SUPERESCALARES

## ANTIDEPENDÊNCIA



$I_1: R3 = R3 + R5$

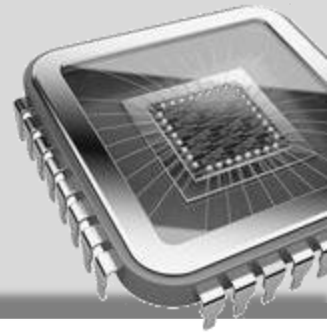
$I_2: R4 = R3 + 1$

$I_3: R3 = R5 + 1$

$I_4: R7 = R3 + R4$

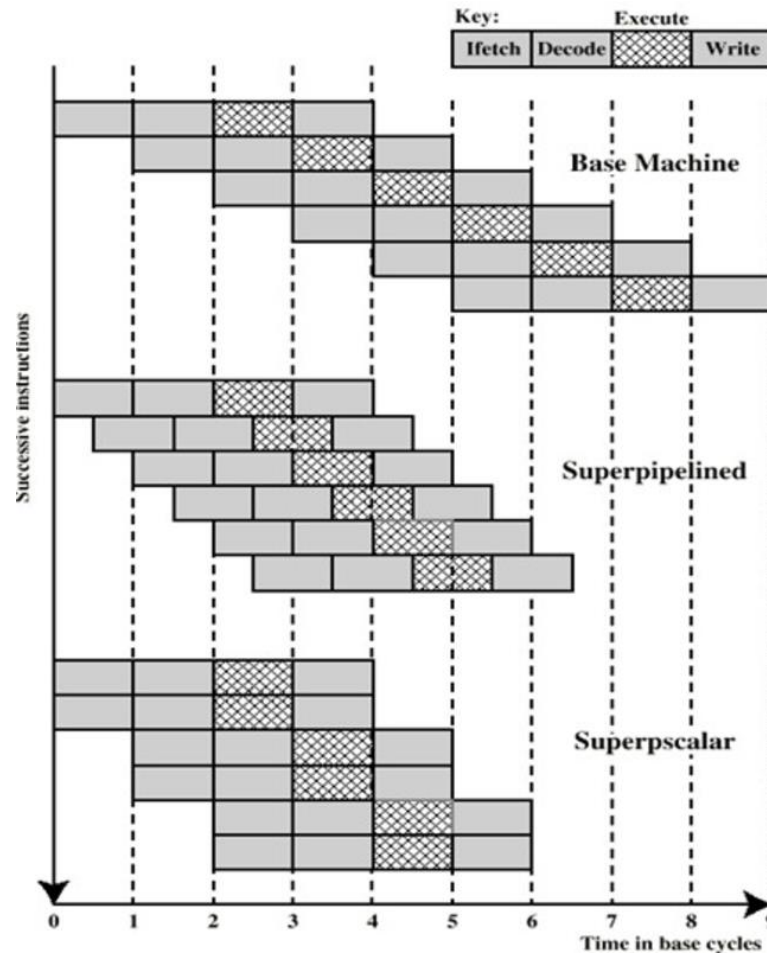
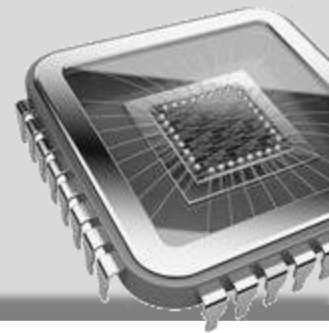
- A instrução 3 não pode concluir sua execução antes da instrução 2 ter obtido seus operandos.
  - Despacho da instrução 3 precisa ser congelado até que a instrução 2 tenha lido o valor R3.

# PROCESSADORES SUPERPIPELINE

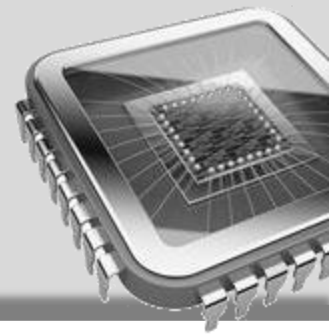


- É um pipeline com muitos estágios
  - Os estágios necessitam tempo de ciclos menores (normalmente menos que a metade).
- Velocidade interna de clocks duplicada
  - Executa duas atividades por ciclo de clock externo
- Superescalar permite executar a busca em paralelo.

# PROCESSADORES SUPERPIPELINE



# Paralelismo em Nível de instrução superescalar vs superpipeline



## Superescalar

- Emprega vários pipeline de instruções independentes
- Cada pipeline com seus estágios, executando instruções diferentes simultaneamente
- Novo nível de paralelismo: diversos fluxos de instrução de cada vez

## Superpipeline

- Instruções próximas que sejam independentes e possam ser executadas em paralelo (ao mesmo tempo)
- Problemas com a dependência de dados
- Identificadas as dependências, execução pode ser feita fora de ordem.

