

# Rompendo a barreira de performance em processadores superescalares através de lógica reconfigurável



Marcelo Brandalero

Universidade Federal do Rio Grande do Sul  
Instituto de Informática  
Orientado por Prof. Dr. Antonio Carlos S. Beck Filho  
mbrandalero@inf.ufrgs.br

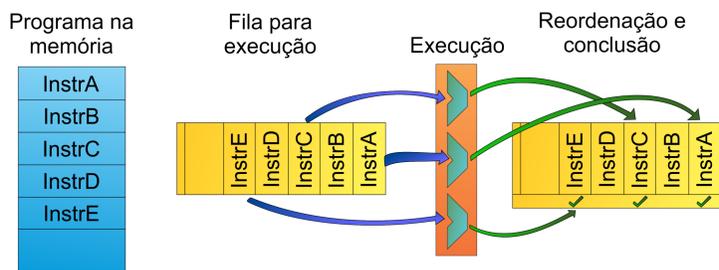


## MOTIVAÇÃO

- Demanda crescente dos consumidores por **maior performance** em seus dispositivos eletrônicos.
  - **Capacidade de integração** em chips **dobrando** a cada 18 meses - *Lei de Moore*. Aproximando-se dos limites impostos pelas tecnologias de fabricação.
  - Projeto de novos microprocessadores é *restrito* por três fatores:
    - **Área utilizada;**
    - **Potência consumida:**
      - *Sistemas embarcados:* Preocupação com o uso de bateria.
      - *Grandes datacenters:* Preocupação com o impacto ambiental.
    - **Compatibilidade binária:** Todos os programas já desenvolvidos para uma determinada família de microprocessadores devem continuar funcionando nas novas gerações.
- Como fazer uso eficiente da área disponível, dentro das limitações de potência, para aumentar a performance dos dispositivos computacionais?**

## DESAFIOS

- **Performance em computadores** diretamente relacionada à **execução paralela** de instruções independentes no programa.
- Processadores atuais empregam, na sua microarquitetura, o paradigma da **superescalaridade**.
  - *Microarquitetura:* Organização interna do microprocessador, em termos de componentes e interconexões.
- **Funcionamento da superescalaridade:** Busca *contínua* por instruções independentes para execução de forma paralela.



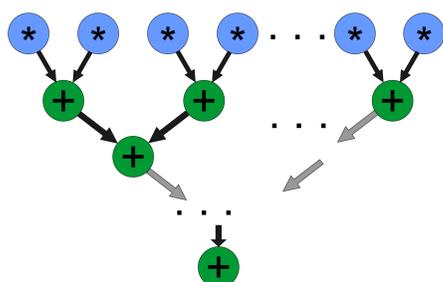
Execução de um programa em um processador superescalar. C independe de A e E independe de A e C; logo, A, C e E podem executar ao mesmo tempo.

- **Problemas da superescalaridade:**
  - **Lógica de verificação de dependências** é responsável por **25 a 50% do consumo de energia**.
  - **Verificação de dependências** é realizada múltiplas vezes sobre uma mesma sequência de instruções já executada anteriormente.
  - Limitando-se a potência, **limita-se a busca por paralelismo** e, portanto, a performance.
  - **Necessidade de buscar-se novos paradigmas microarquiteturais** que maximizem a performance provida por unidade de energia por unidade de área, mantendo-se a compatibilidade binária.

## ARQUITETURAS RECONFIGURÁVEIS

- **Adaptação em tempo de execução** à aplicação sendo executada.
- Transformação de trechos recorrentes de código em lógica combinacional.
- **Vantagens:**
  - **Elimina a verificação contínua de dependências** sobre o mesmo trecho de código, potencialmente reduzindo o consumo energético.
  - Mesmo instruções dependentes podem executar no mesmo ciclo de processador, devido ao hardware dedicado. Aumento de paralelismo que não é possível de ser obtido nos processadores superescalares.
- **Particionamento HardWare/SoftWare:** execução no processador (SW) + execução na unidade reconfigurável (HW).

```
MOV r0, 15
Laço:
MOV r1, Mem[A + i]
MOV r2, MEM[B + i]
MUL r3, r1, r2
ADD r4, r4, r3
DEC r0
LOOP_GE Laço, r0, 0
```

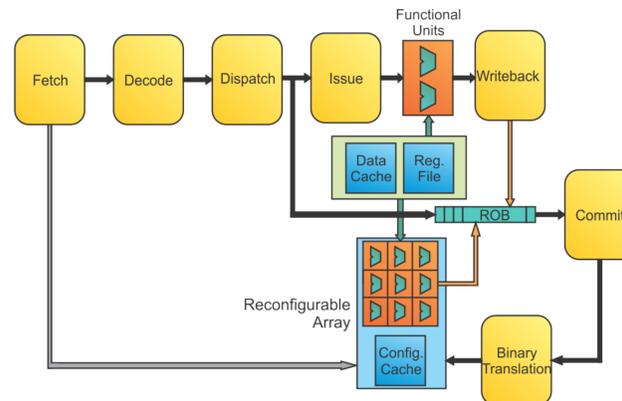


Execução em software (fluxo de instruções).

Execução em hardware (fluxo de dados).

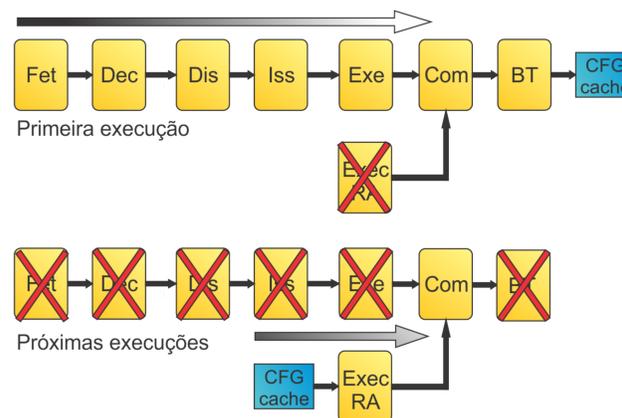
## SISTEMA PROPOSTO

- Um **processador superescalar**, ao qual um **array reconfigurável** (matriz de unidades funcionais) é incorporado. No array, as operações de cada unidade funcional, bem como as interconexões entre elas, são configuráveis *em tempo de execução*.



Microarquitetura do sistema proposto.

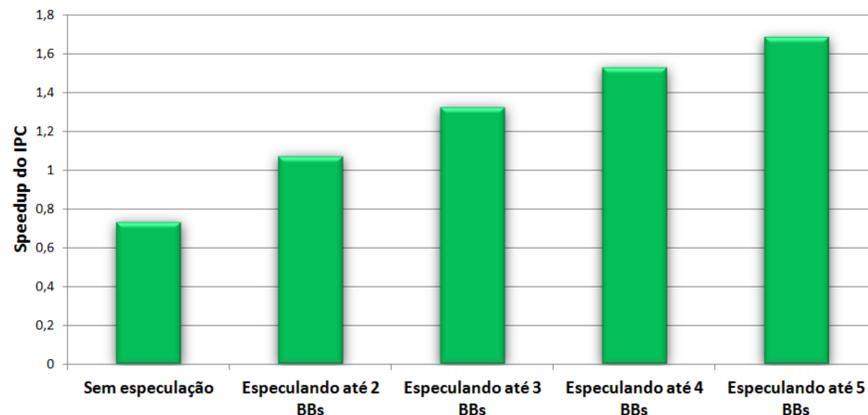
- **Funcionamento da arquitetura:**
  - **Início da execução:** Na primeira execução de um trecho de programa, o pipeline do processador superescalar é utilizado.
  - **Tradução binária:** Ao terminar a execução, cada trecho de programa é transformado em uma configuração do array e armazenado em uma cache.
  - **Próximas execuções** do trecho mapeado: configuração é lida da cache e trecho do programa executa diretamente no array.



Primeira execução (no pipeline superescalar) e execuções subsequentes (direto do array reconfigurável).

## RESULTADOS

- **Metodologia: Análise de potencial.** Blocos básicos da execução dinâmica de programas de um conjunto de benchmarks são analisados pelo paralelismo disponível.
  - **Métrica:** Instruções executadas por ciclo de processador (IPC). São analisadas múltiplas configurações da arquitetura, variando-se a quantidade de trechos de programa (*Basic Blocks*) que podem ser executados simultaneamente no array. Resultados são comparados com o paralelismo obtido na execução no processador superescalar.
  - **Speedup:** resultados normalizados com respeito ao processador superescalar. Valores maiores que um representam ganhos do sistema proposto.



Resultados sugerem que a arquitetura reconfigurável tem capacidade maior de exploração do paralelismo que o processador superescalar.

## SOBRE O TRABALHO

- **Conclusões:**
  - Alto potencial de aumento de performance.
  - Esperada redução do consumo energético devido à redução da verificação do paralelismo entre as instruções - resultado observado em trabalhos similares.
- **Trabalhos futuros:**
  - Análise detalhada da performance resultante da execução no sistema proposto.
  - Análise do uso de área pelo sistema.
  - Análise de potência.

