

# Manual Do Simulador MIPS (Pipeline) NewPipe

## NewPipe

Nome: Charles Rodrigo Ribas Almeida

Eduardo Kluwe Veiga

Rafaelo Pinheiro da Rosa

Vinicius Berne da Costa

Professores: Bruno Silveira Neves

Leonardo Bidese de Pinho

Disciplina: Arquitetura e Organização de Computadores II

Assunto: Simulador MIPS (Pipeline)

Universidade Federal do Pampa (Bagé)

25/05/2009

Sumário:

Introdução.....Pag 01

Implementação do simulador.....Pag 02

Recursos do Simulador.....Pag 03

Descrições da interface com usuário

Utilizando o Simulador.....Pag 04

Introdução:

MIPS, acrônimo para Microprocessor without interlocked pipeline stages

(Microprocessador sem estágios interligados de pipeline ), é uma arquitetura de microprocessadores RISC desenvolvido pela MIPS Computer Systems.

Os processadores MIPS são usados em aplicações tais como:

Computadores da Silicon Graphics;  
Muitos sistemas embarcados;  
Dispositivos com Windows CE;  
Roteadores da Cisco;  
Videogames como Nintendo 64 e PlayStation.

### Implementação do simulador

O Simulador MIPS (Pipeline) foi implementado para possibilitar a visualização da execução de um programa em linguagem de máquina sobre a Arquitetura MIPS. O mesmo foi desenvolvido na linguagem de programação C e executado sobre o compilador gcc. O programa descreve cada componente da Arquitetura MIPS utilizando estruturas e funções.

Foram utilizadas Estruturas para implementar:

- Os Campos das Instruções;
- Os Sinais de Controle;
- Unidade de Detecção de Hazards;
- Todos os Registradores de Pipeline;
- Memórias Cache;

Foram utilizadas Funções para implementar:

- Recursos do simulador;
- Multiplexadores.

O Simulador fornece recursos para exibição de resultados:

- Ao final de uma execução completa;
- Ciclo a ciclo.

O Simulador possui suporte para monitoração (visualização) de(o):

- Valores para sinais de controle;
- Todos os registradores da arquitetura;
- Conteúdo da MP e conteúdo da Cache;

- Sinais de entrada e saída para cada componente da arquitetura.

O Simulador possui uma unidade de detecção de Hazards que detecta os três tipos de Hazards: Estrutural, de Dados e Desvio.

Foi implementada Hierarquia de Memória, representando os três níveis: disco, memória principal e cache.

#### Descrições da interface com usuário

O programa roda sobre linha de comando e não oferece nenhuma dificuldade de entendimento uma vez que não é preciso muita intervenção do usuário na execução do programa.

#### Utilizando o Simulador

O programa ao ser chamado pelo terminal executa um arquivo \*.txt que deve estar no mesmo diretório que o programa.

Os arquivos de texto lidos precisam seguir uma padronização:

- Não podem conter espaços após as vírgulas e cada linha deve ser finalizada com um ";". Ex.:

```
add $s1,$s2,$s3;
```

ao contrario de :

```
add $s1, $s2, $s3
```

- Adicionar o caractere '#' torna a linha inválida e ela é ignorada na leitura. Ex.:

```
# esta instrucao realiza a soma entre dois operandos.
```

```
add $s1,$s2,$s3;
```

esta linha seria ignorada no momento da execução como se não existisse.

- Os Registradores podem ser representados tanto pelo seu nome quanto pelo seu número:

Ex.:

```
Add 1,2,3;
```

É o mesmo que:

```
Add $at,$v0,$v1;
```

- Pode-se usar Labels para facilitar os desvios:

```
j 4;
addi $t0,$zero,20;
addi $t1,$t0,-56;
add $s0,$t0,$t1;
bne $s0,$s1,7;
addi $s0,$zero,1;
j 0;
exit: addi $s1,$zero,33;
```

Pode-se escrever trocando os endereços dos jumps e desvios condicionais por palavras que endereçam labels, como mostra abaixo:

```
j teste;
addi $t0,$zero,20;
addi $t1,$t0,-56;
add $s0,$t0,$t1;
teste: bne $s0,$s1,exit;
addi $s0,$zero,1;
j 0;
exit: addi $s1,$zero,33;
```

Executando o Simulador:

Primeiramente, o usuário deve escolher a opção de uso do Forwarding entre

Ativado(2) ou Desativado(1). Lembrando que se o Forwarding for desativado, o programa estará sujeito aos erros de dependência entre as instruções.

Após o primeiro menu, o usuário deverá escolher a associatividade da Cache entre duas ou quatro vias, opções 1 e 2 respectivamente.

Logo após, o simulador mostra a impressão dos resultados:

- conteúdo da Memória de Dados;
- Conteúdo da Cache;
- Conteúdo dos 32 Registradores.