

DCC006 - Organização de Computadores I
Trabalho Prático 02 - Versão Final
Computador Paralelo Inspirado no MIPS
2013-02

¹Departamento de Ciência da Computação – Universidade Federal de Minas Gerais (UFMG)

1. Descrição

A segunda parte do trabalho prático de OC-I consiste na implementação funcional do simulador do Computador Paralelo Inspirado no MIPS. A organização desenvolvida na primeira parte do trabalho deve ser seguida à risca.

Um dos requisitos fundamentais para se entender como implementar o simulador funcional do processador é a compreensão do termo “funcional”. Uma descrição funcional divide o processador nos blocos que existirão em uma implementação real. Portanto, o processador deverá conter algumas rotinas principais, uma para cada etapa do ciclo a ser executado em cada instrução. Note que a única forma de comunicação entre essas rotinas em um processador real é via registradores, que no caso do simulador serão implementados por estruturas de dados no programa do simulador.

Além dessas rotinas, os elementos principais do processador real deverão estar encapsulados por módulos, tais como os elementos, banco de registradores, ALUs, incrementadores e memória.

A implementação deve ser feita na linguagem C ou C++.

2. Documentação

A documentação será de grande importância na avaliação de todo o trabalho. Não é necessário que seja extensa, porém deve ser completa.

Deve conter basicamente:

- Um resumo da máquina simulada;
- Decisões de implementação;
- Tutorial de como usar o simulador, com instruções completas e precisas de como executá-lo;
- Desenho esquemático da implementação (datapath). Corrigido da versão 1 caso algum problema tenha sido detectado;
- Estruturas utilizadas para representação do hardware;
- Descrição dos módulos utilizados;
- Descrição dos testes utilizados na verificação do simulador e a explicação de por que esses testes são completos e garantem o funcionamento correto do simulador;
- Listagem do código fonte do simulador e dos testes realizados. Essa listagem pode ser impressa em duas páginas por folha.

Esses são tópicos básicos que são considerados importantes na documentação, porém não são suficientes para constituí-la por completo.

Outras informações relevantes do projeto também devem constar da documentação.

2.1. Testes

Os testes pelo qual o processador passa ajuda no desenvolvimento da sua correção e robustez. Como o objetivo primeiro do projeto é prezar pela sua correção, é esperado que se faça uso de uma bateria de testes para auxílio na depuração.

Assim, tendo em vista toda a importância dos testes no projeto, eles devem ser apresentados juntos com a documentação. Deve-se, também, explicar os motivos pelos quais aqueles testes foram usados e sua utilidade. Como sugestão de testes úteis, temos:

- Testes Massivos: procurando testar o maior número de combinações de instruções possíveis;
- Programas Reais: implementações de programas comuns para simular situações reais.

Os códigos dos testes deverão ser apresentados no assembly desenvolvido para a linguagem, além do código binário.

3. Entrega

No dia da entrega do trabalho devem ser apresentados:

- a documentação impressa, o código fonte do simulador e dos testes realizados. É importante ressaltar que o código e os testes devem ser obrigatoriamente entregues. Trabalhos iguais ou copiados terão nota zero;
- o código fonte do simulador, a listagem dos testes e o executável do simulador.

Lembre-se, uma boa documentação facilita a correção e ajuda na avaliação, valendo nota.

O trabalho deverá ser entregue pelo moodle até dia <06/12/2013>, através do minha.ufmg.br.

4. Avisos

- O grupo pode ter no máximo 4 membros, obrigatoriamente os mesmo da etapa 1;
- A documentação DEVE conter as decisões de implementação do trabalho. Atenção para este detalhe;
- Dúvidas devem ser tiradas em sala de aula.