

1. Definição

O caminho de dados é o componente do processador que realiza as operações especificadas e altera o estado da computação por meio dos componentes presentes nele.

2. Exercício

O exercício consiste em montar um caminho de dados e executar algum código que utilize todos os tipos de instruções implementadas.

Os componentes já desenvolvidos nas implementações anteriores devem ser utilizados, e componentes prontos do simulador (como muxes e somadores) podem ser utilizados, desde que justificados em relatório.

3. Especificação

3.1. Caminho de dados monociclo

A Figura 1 mostra um esquema para construção do *datapath* com as unidades de memória, decodificador, banco de registradores e ULA. Os sinais de controle são mostrados com seus nomes iniciando em maiúsculas nos túneis correspondentes.

A unidade de controle principal foi omitida na Figura 1 para melhor visualização do *datapath*, mas ela deve ter sua entrada conectada ao campo OPCODE do decodificador de instruções e as saídas ligadas a cada sinal de controle correspondente.

O componente nomeado como “<<2 conc” tem como entradas o campo addr da instrução e o valor de PC+4, e como saída o valor de (addr << 2) concatenado com os 4 bits mais significativos de PC+4.

O único sinal de controle de 1 bit que não faz parte da unidade de controle principal é JumpRegister, que é saída da Unidade de controle da ULA.

Para melhor visualização da execução no *datapath*, sugere-se a inclusão de pinos para verificação dos campos do decodificador, do resultado da ULA e dos dados lidos da memória.

3.2. Execução

A memória de instruções do *datapath* deve conter instruções válidas a serem executadas. Essas instruções devem possuir um código de alto nível correspondente que faça sentido sintático e semântico. A complexidade do código será levada em consideração na avaliação da execução.

Espera-se que o *datapath* seja capaz de executar todas as instruções especificadas nas entregas parciais do trabalho. Contudo, entende-se que um código de alto nível pode não conter correspondentes para todas as instruções implementadas. Dessa forma, a execução será pontuada de 0 a 70 conforme os tipos de instruções executadas e de 0 a 30 conforme as estruturas do código de alto nível correspondente.

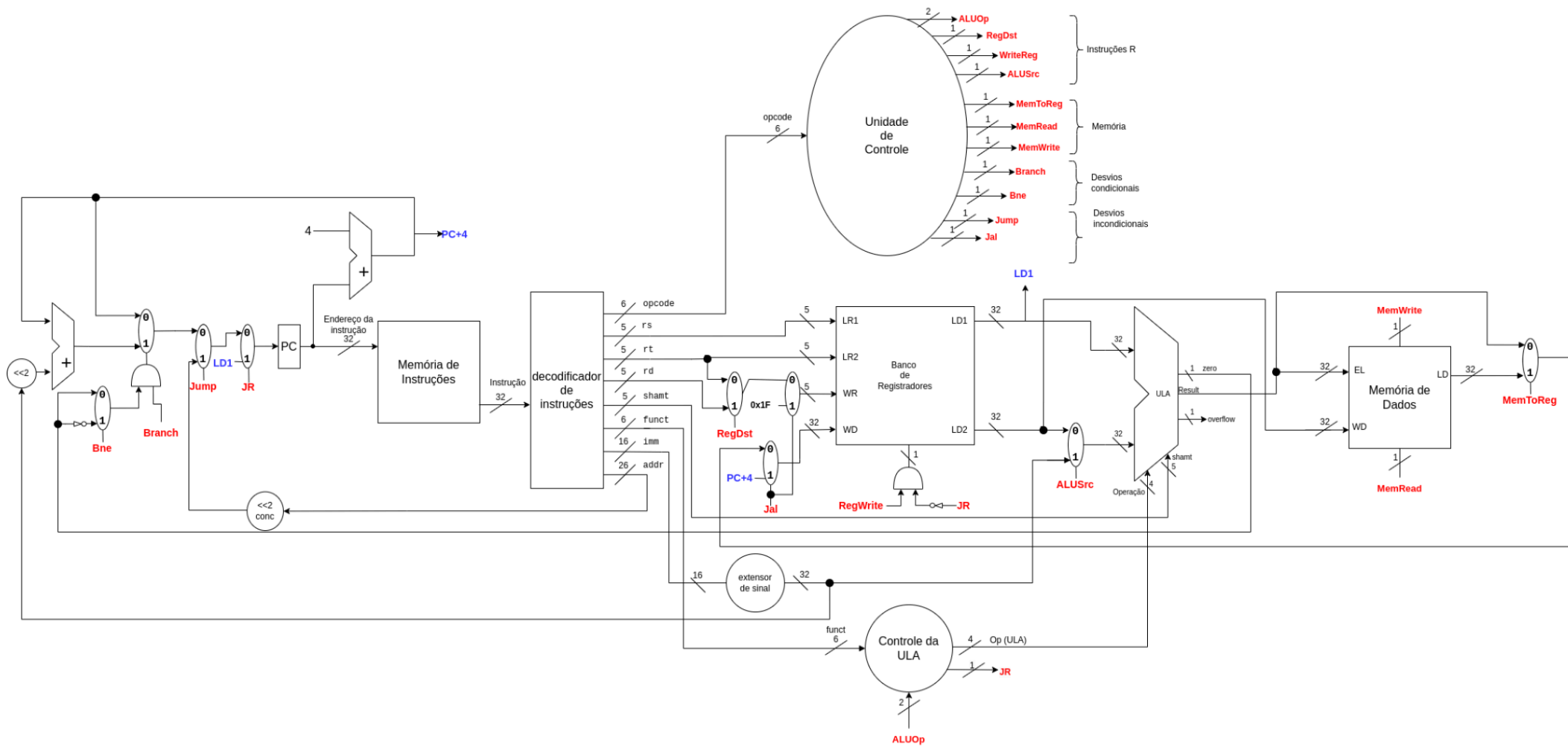


Figura 1 - Esquema do caminho de dados monociclo.

A Tabela 1 mostra a pontuação da execução no *datapath* conforme os tipos diferentes de instruções e as estruturas algorítmicas do código correspondente, na qual a coluna “Pontos” lista o ponto por ocorrência do tipo de instrução ou código, que pode ser requerida até o listado na coluna “Máximo”. Entende-se que cada tipo de instrução pontua somente uma vez, assim a execução de duas instruções do tipo “add” no código, por exemplo, pontua somente uma vez em instruções R.

Tipo		Pontos	Máximo	Máximo
Instruções	Instruções R (exceto JR)	4	20	70
	Instruções aritméticas com imediato	5	15	
	Desvios condicionais	7	14	
	Desvios incondicionais	8	24	
	Operações de memória	8	16	
Código	Estrutura condicional	2	4	30
	Laços de repetição não aninhados	4	8	
	Laços de repetição aninhados	8	8	
	Chamada de função folha	8	8	
	Chamada de função não-folha	12	12	

Tabela 1 - Pontuação conforme a complexidade do código executado corretamente no *datapath*.

4. Entrega/Apresentação

Deverão ser produzidos:

1. Um relatório que descreva o caminho de dados e todas suas unidades em formato de manual.
 - a. Neste relatório deve haver um código em linguagem de alto nível, sua tradução *assembly* e o correspondente na linguagem de máquina que foi salvo no arquivo do *datapath*.
2. O arquivo com o projeto no simulador Logisim.

O relatório da atividade deve estar no formato do modelo disponibilizado no moodle.

O relatório e o arquivo do projeto devem estar compactados em um mesmo arquivo .zip.

Referências

[1] Patterson, David A. Hennessy, John L. Organização e Projeto de Computadores. Disponível em: Minha Biblioteca, (5a. edição). Grupo GEN, 2017.