

# Trabalho 4 da disciplina "Organização e Arquitetura de Computadores" 1 / 2018

Cristiano Silva Júnior: 13/0070629

1 de Junho de 2018

## 1 Introdução

O objetivo deste trabalho é implementar a unidade lógico-aritmética (ULA) do processador MIPS desenvolvido na disciplina. A ULA deverá implementar as seguintes operações:

- Soma e subtração aritméticas
- Soma e subtração lógicas;
- AND, OR, NOR e XOR lógicos;
- "Menor que";
- *Shift* e rotação lógicos para esquerda e direita;

A ULA deverá receber como entrada dois números  $A$  e  $B$  de 32 bits; e um código para operação definido previamente pelo professor e deverá ter como saída um número  $Z$  com o resultado da operação escolhida; um indicador se o resultado vale zero; e outro indicador para a situação de overflow na operação.

## 2 Metodologia

A implementação da ULA foi feita em VHDL utilizando a ferramenta Altera Quartus II com o auxílio do Altera ModelSim para a suíte de testes. Testes unitários para cada operação estão contidos no arquivo "testbench.vhd" enquanto a ULA foi implementada no arquivo "ALU.vhd".

## 3 Resultados

### 3.1 Circuito gerado

Por meio da implementação, pode-se gerar o circuito indicado na figura 1.

### 3.2 Operações aritméticas

Todas as operações aritméticas foram testadas para resultados incluindo números positivos, negativos e zeros, assim como para mostrar a detecção de overflow como mostra a figura 2.

### 3.3 Operações lógicas

Todas as operações lógicas foram testadas com números com os quais seriam fáceis de visualizar o resultado final, como mostra a figura 3.

Além disso, implementamos as operações de soma e subtração para os casos com números sem sinal, como mostra o resultado da simulação na figura 4.

#### 3.3.1 Operações "Menor que"

Para as operações "menor que", foram utilizadas duas situações: uma em que o número  $A$  é de fato menor que  $B$ , tornando  $Z = 1$ ; e outra em que  $A > B \rightarrow Z = 0$ . O resultado da simulação encontra-se na figura 5.

### 3.4 Operações de *Shift* e rotações

As operações de *shift* e de rotação foram testadas com casos extremos, isto é, que há interações do resultado com o limite da representação numérica de 32 bits, como mostra a figura 6.

## 4 Conclusão

A implementação de toda a ULA proposta foi possível sem maiores problemas.