

# Arquitetura e Organização de Computadores

## Turma B - 2013/02

### Projeto de uma ULA em VHDL

**Objetivo:** projetar, simular e sintetizar uma ULA de 32 bits no ambiente Quartus / ModelSim-Altera. Implementar a ULA de 3 maneiras diferentes:

- usando um comando concorrente
- usando um comando case dentro de um processo
- usando o comando if-then-else em um processo

#### Características:

- Duas entradas de 32 bits
- Uma Saída de 32 bits
- Sinal Zero: detecta valor zero na saída
- Sinal Overflow: ativo quando a operação de soma ou subtração gerar resultado que ultrapasse o limite de representação em 32 bits
- Operações (em Complemento de 2):

Operação	Significado	OpCode
add A, B	Saída Z recebe a soma das entradas A, B incluindo o vem-um	000
sub A, B	Saída Z recebe A - B	001
and A, B	Saída Z recebe a operação lógica A and B, bit a bit	010
or A, B	Saída Z recebe a operação lógica A or B, bit a bit	011
not A	Saída Z recebe a entrada A invertida bit a bit	100
xor A, B	Saída Z recebe a operação lógica A xor B, bit a bit	101
bypass A	Saída Z recebe A, sem alteração	110
slt A, B	Z = 1 se A < B	111

#### Interface:

```
entity ulaMIPS is
  port (
    opcode          : in std_logic(2 downto 0)
    A, B            : in std_logic_vector(31 downto 0);
    Z               : out std_logic_vector(31 downto 0)
    vai, zero, ovfl : out std_logic);
end ulaMIPS;
```

onde:

- *opcode* indica a operação a ser realizada
- *A e B*: operandos, 32 bits.
- *Z*: saída, 32 bits
- *zero*: indicação de zero na saída
- *ovfl*: overflow

**Simulação e Verificação:** simular o funcionamento da ULA de forma a verificar o funcionamento de cada uma das suas operações. Verificar igualmente a geração dos sinais *vai*, *zero* e *ovfl*. Utilizar o ModelSim Altera para a simulação, desenvolvendo um *testbench* para acionamento dos sinais.

A verificação deve incluir a execução de ao menos um teste para cada operação da ULA. As operações aritméticas devem ser testadas para resultado zero, negativo, positivo e *overflow*.

**Relatório:** configurar o Quartus para sintetizar a ULA para o FPGA Cyclone II EP270F896C6N. Informar o resultado da síntese para cada caso.

**Entrega:** 09 de novembro.

- Incluir o código da ULA e do *testbench*
- Mostrar as telas de simulação no ModelSim
- Incluir os dados da síntese que são apresentados pelo Quartus II