



## **Laboratório 5**

### **- CPU MIPS Pipeline –**

#### **Objetivos:**

- Treinar o aluno com a linguagem de descrição de hardware Verilog;
- Familiarizar o aluno com a plataforma de desenvolvimento FPGA DE2 da Altera e o software QUARTUS II;
- Desenvolver a capacidade de análise e síntese de sistemas digitais usando uma Linguagem de Descrição de Hardware;
- Estudar e implementar uma CPU MIPS com ISA mínima e Pipeline de 5 estágios;

- 1) (0.0) Abra e compile o projeto do processador MIPS PUM v.6.0 com o Processador Pipeline
  - a. Carregue o programa testePIPE.s ;
  - b. Simule no Mars e por forma de onda comparando os resultados;
  - c. Identifique e corrija o que está causando o resultado errôneo.
- 2) (1.0) Analise o processador MIPS Pipeline fornecido. Desenhe o Diagrama de Blocos do Caminho de Dados completo incluindo os registradores de pipeline e especifique a tabela verdade dos sinais de controle por estágio do pipeline;
- 3) (1.0) Analise as unidades de Hazard e Forward e com base na ISA especifique, através de exemplos, quais riscos de dados e de controle são detectados e tratados.
- 4) (1.0) Use o seu programa teste.s e verifique o correto funcionamento de TODAS as instruções da ISA implementada, teste usando simulação por forma de onda e pela implementação na DE2.
- 5) (1.0) Execute no processador em FPGA o seu programa de simulação de lançamento de bola de canhão desenvolvido no Laboratório 1 (Dica: defina os parâmetros no seu programa **usando** o syscall 6). Grave vídeos demonstrativos e disponibilize no YouTube com links no relatório.
- 6) (2.0) Verifique o correto funcionamento do Syscall 49 (leitura do cartão SD). Defina no PC e grave em um cartão SD os 12 cenários do jogo Street Fighter. Faça um programa que leia sequencialmente as telas e as apresente no monitor VGA. Faça comentários sobre as limitações e máxima taxa de quadros atingida. Filme o experimento com links no relatório.
- 7) (4.0) Implemente as instruções abaixo em conformidade com a ISA MIPS (livro See MIPS Run e Manual do MIPS):
  - mul \$t1,\$t2,\$t3 Multiplication without overflow : Set HI to high-order 32 bits, LO and \$t1 to low-order 32 bits of the product of \$t2 and \$t3
  - jalr \$t1 Jump and link register : Set \$ra to Program Counter (return address) then jump to statement whose address is in \$t1
  - jalr \$t1,\$t2 Jump and link register : Set \$t1 to Program Counter (return address) then jump to statement whose address is in \$t2
    - a. (1.0) Indique as modificações necessárias no caminho de dados
    - b. (1.0) Indique as modificações necessárias no bloco de controle
    - c. (1.0) Indique as modificações necessárias nas unidades de detecção e tratamento de Hazards
    - d. (1.0) Crie um programa teste que comprove o correto funcionamento das novas instruções. Faça a simulação em forma de onda e sintetize na DE2.