



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 DE1SoC da Intel e o software QUARTUS Prime;
- Desenvolver a capacidade de análise e síntese de sistemas digitais usando uma Linguagem de Descrição de Hardware;
- Apresentar ao aluno a implementação de uma CPU Pipeline;

PARTE A: Apresentação do processador MIPS Pipeline

- 1) (4.0) Abra o projeto do processador MIPS-v1.3. Neste projeto os 3 processadores já estão sintetizados nos arquivos .sof.
 - a. (1.0) Defina o uso do processador UNICICLO, não sintetize a FPU, defina SIMULACAO, faça apenas a etapa de Analysis & Synthesis do projeto. Compare a execução passo a passo do programa testeSIMPLESPIPE.s no Mars, a simulação por forma de onda através do arquivo Waveform2.vwf e a implementação MIPS_UNICICLO.sof na DE1-SoC. Manualmente, encontre a máxima frequência de clock no qual o programa ainda é corretamente executado na simulação e seu tempo total de execução. Analise e comente os resultados.
 - b. (1.0) Defina o uso do processador MULTICICLO, não sintetize a FPU, defina SIMULACAO, faça apenas a etapa de Analysis & Synthesis do projeto. Compare a execução passo a passo do programa testeSIMPLESPIPE.s no Mars, a simulação por forma de onda através do arquivo Waveform2.vwf e a implementação MIPS_MULTICICLO.sof na DE1-SoC. Manualmente, encontre a máxima frequência de clock no qual o programa ainda é corretamente executado na simulação e seu tempo total de execução. Analise e comente os resultados.
 - c. (1.0) Defina o uso do processador PIPELINE, não sintetize a FPU, defina SIMULACAO, faça apenas a etapa de Analysis & Synthesis do projeto. Compare a execução passo a passo do programa testeSIMPLESPIPE.s no Mars, a simulação por forma de onda através do arquivo Waveform2.vwf e a implementação MIPS_PIPELINE.sof na DE1-SoC. Manualmente, encontre a máxima frequência de clock no qual o programa ainda é corretamente executado na simulação e seu tempo total de execução. Analise e comente os resultados.
 - d. (1.0) Compare os requerimentos físicos i) Número de Elementos Lógicos (ALMs), ii) Número de Registradores e iii) Quantidade de bits de memória e iv) Número de blocos DSP; e temporais: i) caminho de maior atraso tpd, ii) caminhos com piores tempos th, tco, tsu e slacks, iii) máxima frequência de clock obtida pelo TimeQuest, dos processadores MIPS UNICICLO, MULTICICLO e PIPELINE. Analise e comente os resultados.

PARTE B: Análise do processador MIPS Pipeline

- 2) (3.0) Abra e compile o projeto do processador MIPSv.1.3 com o Processador Pipeline e sem FPU, carregue o programa testeSIMPLESPIPE.s :
 - a. (0.5) Simule o programa passo a passo no Mars indicando os resultados intermediários esperados;
 - b. (1.0) Simule por forma de onda e analise os resultados comparando com a execução passo a passo no Mars;
 - c. (0.5) Carregue na DE1-SoC e filme a execução passo a passo;
 - d. (1.0) Identifique e corrija, com bolhas, a(s) instrução(ões) que está(ão) causando o resultado errôneo.
- 3) (2.0) Analise o processador MIPS Pipeline fornecido. Desenhe o Caminho de Dados completo incluindo os registradores de pipeline e especifique a tabela verdade dos sinais de controle por estágio do pipeline;
- 4) (1.0) Analise e descreva as unidades de Hazard e Forward. Com base na ISA especifique, através de exemplos, quais riscos de dados e de controle são detectados e tratados.
- 5) (0.5) Crie programas com possíveis hazards que não estão sendo tratados e teste se realmente esses hazards estão causando erros na execução em forma de onda e no passo a passo na DE1-SoC (filme).