**UNIVERSIDADE FEDERAL DE RORAIMA**

**CENTRO DE CIÊNCIA E TECNOLOGIA**

**DEPARTAMENTO DE CIÊNCIA DA COMPUTAÇÃO**

**DCC301– ARQUITETURA E ORGANIZAÇÃO DE COMPUTADORES– 2024**

**PROF. DR. HEBERT OLIVEIRA ROCHA**

**LEONARDO VINÍCIUS LIMA CASTRO**

**ÁLEFE ALVES DA COSTA**

**RELATÓRIO DO PROCESSOR DE RISC DE 8 BITS**

**BOA VISTA, RR**

**2025**

**LEONARDO VINÍCIUS LIMA CASTRO**

**ÁLEFE ALVES DA COSTA**

**RELATÓRIO DO PROCESSOR DE RISC DE 8 BITS**

Trabalho da disciplina de Arquitetura e Organização de Computadores do ano de 2024.2 apresentado à Universidade Federal de Roraima do curso de Bacharelado em ciência da computação.

Docente: Prof. Dr. Hebert O. Rocha

**BOA VISTA, RR**

**2025**

**LISTA DE FIGURAS**

Figura 1 – PC viewer.................................................................................................11

Figura X – Datapath......................................................................................................

**LISTA DE TABELAS**

Tabela 1 - ..................................................................................................................11

1. INTRODUÇÃO

Este relatório técnico apresenta o projeto e a implementação de um processador RISC de 8 bits, desenvolvido utilizando a linguagem de descrição de hardware VHDL no software Quartus Prime 18.0. O processador projetado segue uma arquitetura semelhante ao MIPS e tem como objetivo a aplicação de conceitos fundamentais de arquiteturas computacionais, circuitos digitais e linguagens de descrição de hardware.

A construção do processador envolve a modelagem e implementação de seus componentes principais, incluindo a unidade de controle, o caminho de dados (datapath) e os barramentos de comunicação. Além disso, a descrição da estrutura das instruções suportadas é detalhada por classes e distribuições de bits, incluindo sua representação em linguagem assembly e binário.

O conjunto de instruções obrigatórias implementadas inclui operações essenciais, como load, store, soma, subtração, beq (branch if equal) e salto incondicional, garantindo funcionalidade mínima para a execução de programas simples. Para validar o correto funcionamento do processador, foram realizadas simulações detalhadas por meio de waveforms e testbenches, cobrindo cada instrução individualmente e um programa que integra todas as operações suportadas.

Este trabalho documenta todo o processo de desenvolvimento, desde a descrição dos componentes e suas conexões até a validação funcional do processador, contribuindo para o entendimento e a aplicação prática de arquiteturas RISC em projetos acadêmicos e profissionais.

2. ESPECIFICAÇÕES

Esta seção descreve as especificações do ambiente de desenvolvimento e o conjunto de instruções utilizadas no projeto.

2.1. Ambiente de desenvolvimento

O desenvolvimento do projeto foi realizado utilizando o Quartus Prime 18.0 Lite Edition, um ambiente integrado para o desenvolvimento de circuitos digitais em FPGA.

O Quartus Prime Lite Edition é uma ferramenta da Intel FPGA, utilizada para design, síntese, simulação e programação de FPGAs. Neste projeto, foi empregado VHDL para a implementação de um processador RISC de 8 bits, permitindo a descrição e a simulação da arquitetura antes da gravação no FPGA. Além disso, o ambiente oferece recursos gráficos para facilitar a configuração e depuração do circuito.

2.2. Conjuto de instruções

O processador RISC de 8 bits desenvolvido neste projeto possui um conjunto de instruções organizadas em três formatos principais: Tipo R, Tipo I e Tipo J. Cada instrução segue um padrão de codificação baseado em opcodes e operandos, distribuídos em um formato de 8 bits.

2.2.1. Instruções do tipo R

As instruções do tipo R são utilizadas para operações aritméticas e lógicas entre registradores. Elas seguem o seguinte formato:

|  |  |  |
| --- | --- | --- |
| Opcode | rs | rt |
| 7 - 5 | 4 - 3 | 2 - 1 |
| 3 bits | 2 bits | 2 bits |

2.2.2. Instruções do tipo I

As instruções do tipo I realizam operações que envolvem constantes imediatas ou acesso à memória. Elas são divididas em dois formatos:

|  |  |  |
| --- | --- | --- |
| Opcode | rs | Immediate |
| 7 - 5 | 4 - 3 | 2 - 0 |
| 3 bits | 2 bits | 3 bits |

2.2.3. Instruções do tipo J

As instruções do tipo J (Jump) são usadas para saltos incondicionais no programa, modificando diretamente o contador de programa (PC).

|  |  |
| --- | --- |
| Opcode | Adress |
| 7 - 5 | 4 - 0 |
| 3 bits | 5 bits |

3. Construção do processador

Esta seção apresenta os componentes desenvolvidos para o processador, e suas especificações.

3.1. PC

3.2. MÉMORIA DE INSTRUÇÕES

3.3. somador

3.4. EXTENSOR DE BITS 3X8

3.5. BANCO DE REGISTRADORES

3.x. datapath

O datapath é a estrutura que permite a execução de instruções em um processador. Ele inclui componentes como o Bit Extender, que ajusta o tamanho dos dados, e a ALU, que realiza operações matemáticas. Multiplexadores (MUX) direcionam o fluxo de dados, enquanto sinais de controle, como Jump e Branch, gerenciam desvios no programa. Operações de memória são controladas por MemRead e MemWrite, e o resultado é armazenado em registradores com RegWrite. Juntos, esses elementos garantem a execução eficiente das instruções.

Figura X - Datapath

Diagrama, Esquemático

O conteúdo gerado por IA pode estar incorreto.

4. simulações

5. CONCLUSÃO

Este projeto, desenvolvido para a matéria de Arquitetura e Organização de Computadores (AOC), foi uma experiência muito importante para o nosso aprendizado. O objetivo era criar um processador RISC de 8 bits, e, ao longo do processo, enfrentamos alguns desafios, mas também tivemos muitas conquistas.

Uma das maiores dificuldades foi aprender a usar a linguagem VHDL, que é bem diferente das linguagens de programação que já conhecíamos. No começo, foi complicado entender como descrever o hardware de forma clara e funcional, mas, com prática e dedicação, conseguimos superar essa barreira. Outro desafio foi integrar todos os componentes do processador, como a unidade de controle, a ULA e os registradores. Foi preciso bastante atenção para garantir que tudo funcionasse corretamente junto.

Apesar das dificuldades, o projeto foi muito interessante porque nos ajudou a fixar melhor o conteúdo visto em sala de aula. Conseguimos ver na prática como os conceitos de arquitetura RISC, pipeline e funcionamento de um processador são aplicados. Além disso, as dúvidas que surgiram durante o desenvolvimento foram resolvidas em grupo, o que mostrou como a colaboração é essencial para o sucesso de um projeto.

No final, conseguimos concluir o processador e entender cada parte do seu funcionamento. Esse projeto não só reforçou nosso conhecimento em AOC, mas também nos preparou para enfrentar desafios maiores no futuro. Foi uma experiência que valeu muito a pena.

6. REFERÊNCIAS

STALLINGS, William; BOSNIC, Ivan; VIEIRA, Daniel. **Arquitetura e organização de computadores**. 8. ed. São Paulo: Prentice Hall, 2006.

PATTERSON, David A. **Organização e projeto de computadores**. 3. ed. Rio de Janeiro: Elsevier, 2005.