



Problema #1 – 2025.1

Sistemas Digitais

1. Tema

Desenvolver um coprocessador aritmético especializado em multiplicação matricial.

2. Objetivos de Aprendizagem

Ao final da realização deste problema, o/a discente deverá ser capaz de:

- Aplicar conhecimentos de circuitos digitais e arquiteturas de computadores para desenvolver um coprocessador aritmético;
- Implementar a solução em Verilog numa FPGA;
- Entender dificuldades tecnológicas no desenvolvimento de processadores;
- Compreender os princípios básicos da arquitetura da plataforma DE1-SoC;
- Utilizar as interfaces disponíveis na placa DE1-SoC.

3. Problema

Este projeto atende a diversas aplicações que exigem cálculos intensivos utilizando matrizes, como processamento de imagens e visão computacional, aprendizado de máquina, computação gráfica, simulações científicas, criptografia e telecomunicações. Implementar um coprocessador dedicado permite acelerar essas operações, reduzindo o tempo de execução e otimizando o uso de recursos computacionais, tornando-se uma solução eficiente para sistemas embarcados de alto desempenho.

4. Requisitos

O problema a ser desenvolvido no Kit de desenvolvimento DE1-SoC deve atender às seguintes restrições:

- 4.1. O código deve ser escrito em linguagem Verilog;
- 4.2. O sistema só poderá utilizar os componentes disponíveis na placa;
- 4.3. Deverá fazer operações de matrizes quadradas de $N \times N$ elementos ($N \leq 5$).
- 4.4. Deverá fazer as seguintes operações:
 - 4.4.1. Adição de matrizes;
 - 4.4.2. Subtração de matrizes;
 - 4.4.3. Multiplicação de matrizes;
 - 4.4.4. Multiplicação de matriz por número real;

- 4.4.5. Determinante;
- 4.4.6. Transposição de matriz;
- 4.4.7. Matriz oposta.
- 4.5. Cada elemento da matriz é representado por um número inteiro de 8 bits.
- 4.6. O coprocessador deve implementar paralelismo para otimizar a execução.
- 4.7. Arquitetura baseada em pipeline para processamento eficiente.
- 4.8. Entrada e saída de dados via barramento simples de controle.
- 4.9. O coprocessador deve ser compatível com o processador ARM (*Hard Processor System* - HPS), ele deve receber instruções para multiplicar matrizes.

5. Produto

Todo o projeto deverá ser disponibilizado na plataforma GitHub. No prazo indicado no cronograma a seguir, cada equipe deverá apresentar:

- 5.1. Levantamento de requisitos;
- 5.2. Código
 - 5.2.1. Código em linguagem Verilog e C;
 - 5.2.2. Todos os códigos deverão estar detalhadamente comentados;
- 5.3. Script de compilação tipo Makefile para geração do código executável;
- 5.4. Documentação técnica escrita no arquivo README do projeto no GitHub, contendo, no mínimo:
 - 5.4.1. Detalhamento dos software usados no trabalho, incluindo softwares básicos;
 - 5.4.2. Arquitetura do computador usado nos testes;
 - 5.4.3. Descrição de instalação, configuração de ambiente e execução;
- 5.5. Descrição dos testes de funcionamento do sistema, bem como, análise dos resultados alcançados.

6. Avaliação

Para avaliar o envolvimento do grupo nas discussões e na apresentação, o tutor poderá fazer perguntas variadas a qualquer membro, tanto nas sessões tutoriais quanto na apresentação. O estudante que não comparecer, ou se atrasar, no dia da sessão de apresentação, terá automaticamente nota 0,0 (zero) no problema, excetuando-se as condições que permitem 2ª chamada de avaliações, conforme regulamento do curso.

A nota final será a composição de 3 (três) notas parciais:

| Critério | Critérios para a nota | Peso |
|-----------------------|---|------|
| Desempenho Individual | Participação individual nas sessões tutoriais, de acordo com o interesse e entendimento demonstrados pelo aluno, assim como sua assiduidade, pontualidade e contribuição nas discussões. Essa nota inclui o desempenho do estudante na apresentação do problema no laboratório. | 4 |

| | | |
|--------------|--|---|
| Documentação | Documentação técnica de cada grupo, considerando qualidade da redação (ortografia e gramática), organização dos tópicos, definição do problema, descrição da solução, explicação dos experimentos, análise dos resultados, detalhando os itens não atendidos, se for o caso. | 3 |
| Códigos | Qualidade do código fonte (organização e comentários), e execução correta dos códigos binários de acordo com testes de validação que explorem as situações de uso. | 3 |

7. Cronograma

| Semana | Data | Descrição |
|--------|----------------|---------------------------------------|
| 2 | qua. - 26/fev. | Problema 1 – Apresentação |
| | sex. - 28/fev. | Ponto faltativo – Pré-Carnaval |
| 3 | qua. - 05/mar. | Feriado – Quarta-Feira de Cinzas |
| | sex. - 07/mar. | Problema 1 – Lab 0 |
| 4 | qua. - 12/mar. | Problema 1 – Seção Tutorial #2 |
| | sex. - 14/mar. | Problema 1 – Seção Desenvolvimento #1 |
| 5 | qua. - 19/mar. | Problema 1 – Seção Tutorial #3 |
| | sex. - 21/mar. | Problema 1 – Lab 1 |
| 6 | qua. - 26/mar. | Problema 1 – Seção Tutorial #4 |
| | sex. - 28/mar. | Problema 1 – Seção Desenvolvimento #2 |
| 7 | qua. - 02/abr. | Problema 1 – Seção Desenvolvimento #3 |
| | sex. - 04/abr. | Problema 1 – Entrega/Avaliação |

8. Links úteis

- Site do Laboratório de Eletrônica Digital e Sistemas (LEDS): <https://sites.google.com/uefs.br/ltec3-leds>
- FPGA Academy: <https://fpgacademy.org/boards.html>