

Trabalho 1 (T1) - 2025-1

Objetivo: Familiarizar o aluno com a metodologia de verificação UVM, tal qual a prática com as ferramentas EDA relacionadas, através do desenvolvimento de um projeto prático. A organização do trabalho considera as combinações abaixo:

- O trabalho possui 2 entregáveis: código-fonte com scripts de execução do projeto e relatório.
 - Todos os entregáveis deverão ser entregues em formato zip pela área Moodle da disciplina até a data limite: **08/04/2025 até 23h50**.
 - Alunos que falharem em entregar seus arquivos receberão nota zero.
 - Projetos sem scripts de execução (e.g., sim.do ou Makefile) receberão nota zero.
 - Arquivos não-autorais e/ou gerados por inteligência artificial receberão nota zero e configurarão plágio acadêmico, tanto para o código-fonte, scripts, ou relatório.
- O trabalho deverá ser desenvolvido em grupos de **2 alunos** ou mais.

Enunciado: Considere o projeto de verificação para multiplicadores (ver Aula 8 - UVM), disponível no Moodle da disciplina. Seu grupo deverá adaptar o projeto para realizar a verificação de um processador de 8-bits (<https://github.com/andersondomingues/8-Bit-Processor-in-SystemVerilog>). A pontuação será atribuída de acordo com os seguintes critérios:

- [25%] Cobertura funcional (coverage) do processador de no mínimo 95%. Considere a criação de diferentes bins para organizar os pontos de cobertura e também a utilização de geradores randômicos de dados.
- [25%] Correta utilização de asserções imediatas e concorrentes (SVA). Utilize asserções com sintaxe temporal quando conveniente e separe as propriedades concorrentes dos arquivos de projeto (rtl).
- [25%] Organização do projeto para funcionamento com o framework UVM (driver, scoreboard, etc). Lembre-se de que os arquivos de verificação devem estar separados dos arquivos de projeto (rtl).
- [25%] Relatório: O relatório deverá conter um pequeno tutorial sobre como executar seu projeto (passo-a-passo), além de mostrar a organização geral do projeto (ex., figura mostrando os módulos) e uma descrição de cada componente (driver, monitor, etc.). O relatório precisa demonstrar a cobertura e 95% do projeto através de discussão dos logs de simulação (coverage), apresentando também uma breve explicação das propriedades (SVA) utilizadas e sugestões de melhorias no design.

Dica: Não modifique os arquivos RTL do processador! Você deve criar uma infraestrutura de verificação incluindo o processador no seu projeto, porém não deve "poluir" o design.