

Exercícios de Verilog.

Instruções

- Os exercícios devem ser implementados usando a linguagem Verilog. Todos os códigos devem ser simulados no ModelSim.
- A solução deverá ser salva em um arquivo no formato V. Cada equipe vai enviar dois arquivos Equipe-X e Equipe-TB-X onde X é o número da equipe (ex: Equipe-1.v, Equipe-TB-1.v, etc). **Arquivos com nomes em formatos diferentes destes serão ignorados.**
- Entrega: **até às 23:55h do dia 11/06/2020** (não serão aceitas listas entregues após este horário).
- Os arquivos devem ser entregues pelo classroom.
- Cada equipe deve resolver a sua lista de **forma individual**.
- **CUIDADO COM CÓPIAS!** **Cópias não serão toleradas!** Será usada uma ferramenta para detecção de cópias. Será feita a comparação com soluções disponíveis na internet e com os exercícios entregues pelas demais equipes. Caso seja detectada alguma fraude as questões serão anuladas.
- A lista vale 10 pontos e tem peso 15% na nota da terceira unidade.

1. Projete uma ULA simples com as seguintes entradas/saídas.

InA (32 bits) - Operando de entrada A

InB (32 bits) - Operando de entrada B

ALUCode (6 bits) - Código da operação a ser executada pela ULA.

Out (32 bits) - Saída contendo o resultado da operação

Shamt (5 bits) - Entrada de 5 bits

A ULA deve suportar os seguintes comandos:

Código	Operação	Detalhes
0x20	ADD	Out = InA + InB
0x22	SUB	Out = InA - InB
0x24	AND	Out = InA & InB
0x25	OR	Out = InA InB
0x03	SRA	Out disponibiliza shift para direita de InA preservando o sinal. A quantidade a ser deslocada para a direita deve ser shamt.
0x02	SRL	Out disponibiliza shift para direita de inA sem preservar o sinal. A quantidade a ser deslocada para a direita deve ser shamt.
0x00	SLL	Out disponibiliza shift para esquerda de inA sem preservar o sinal. A quantidade a ser deslocada para a esquerda deve ser shamt.
Qualquer outro valor		Out = 0x00000000

2. Implemente um TestBench para testar e validar a funcionamento da ULA. O test bench deve testar todas os códigos de operação ilustrados na tabela acima. Escolha um valor entre 5 e 30 para o sinal **shamt**.