# Relatório do Projeto Final de Laboratório de Circuitos Digitais - Fase 1
# Grupo 14: Fábio Miguel e Luís Gustavo

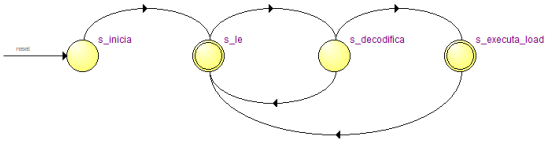
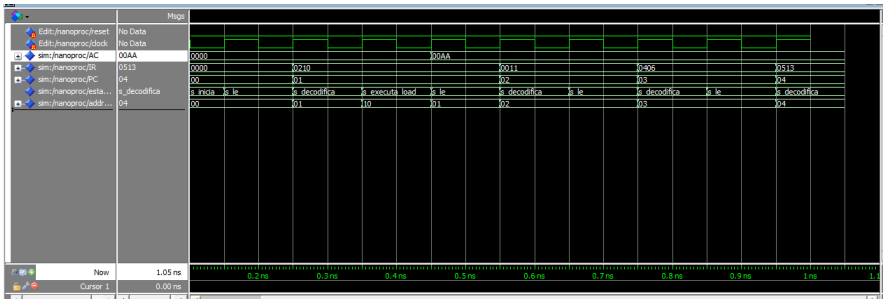



Diagrama de estados:
O diagrama  representa a nossa máquina de estados , temo o s_inicia que é o ponto de partida após o reset. Nenhum registrador muda aqui, serve para alinhar a MEF. após isso, vamos para o s_le, que coloca PC no barramento , lê a memória e prepara a carga de IR e o incremento do PC. s_decodifica examina o opcode e decide se a instrução é LOAD ou qualquer outra, se for LOAD vai para o s_executa_load, coloca IR no barramento, lê a memória e carrega o dado em AC, caso a instrução nãoseja LOAD, s_decodifica vai diretamente para s_le.




Simulação:
Linha 1 (AC): 0000 → 00AA e permanece;
Linha 2 (IR): 0000, 0210, 0011, 0406, 0513…;
Linha 3 (PC): 00, 01, 02, 03, 04…;
Linha 4 (estado_atual): s_inicia, s_le, s_decodifica, s_executa_load, s_le, s_decodifica, s_le…;
Linha 5 (addrBus): segue o PC nos fetiches e vira 10 apenas no ciclo s_executa_load.

As transições ocorrem exatamente como definidas no VHDL, s_inicia → s_le → s_decodifica → (se LOAD) s_executa_load → s_le… Para instruções cujo opcode é diferente de 02, o processador retorna diretamente a s_le, como previsto. Iniciamos com PC, IR e AC zerados, no primeito retch (s_le) o proecssador coloca o endereço 00 no barramento, lê 0210 e grava no IR, simultaneamente o PC é incrementado para 01. Em decodifica, detecta-se que o opcode é 02 (LOAD) e o barramento de endereço muda para 10, apontando o operando. No ciclo seguinte (s_executa_load), o dado de memória 00AA é lido de 10 e carregado no AC, com LOAD concluído, a FSM retorna a s_le e o PC = 01 inicia o fetch da próxima instrução (0011), assim, concluímos que o nosso processador, FSM e a função LOAD estão funcionando como o esperado para a fase 1 do projeto.