

Universidade Federal de Santa Catarina  
DEC7555-04655 A/B (20241) - Linguagem de Descrição de Hardware  
Semestre: 2024/1 – Projeto  
Jogo Genius

O projeto final consiste na implementação de um circuito na placa de desenvolvimento DE0 BOARD fazendo uso das estruturas e conhecimentos obtidos durante o curso e estudos extras. O circuito vai implementar um jogo interativo similar ao jogo GENIUS. O comportamento do jogo é definido a seguir:

- Para iniciar o jogo o usuário ativa a entrada de reset, SW(1), e passamos ao estado *START* onde são ativados os comandos de reset (R1 e R2). Nesse estado os displays HEX5 e HEX4 mostrarão a letra L de level e o nível de jogo, respectivamente, os displays HEX3 e HEX2 mostrarão a letra t de time e o tempo máximo de jogo, respectivamente, e por último, os displays HEX1 e HEX0 mostrarão a letra r de round e o valor da rodada do jogo, respectivamente. O nível de jogo, tempo máximo de jogo e rodada serão explicados a seguir. Do estado *START* passamos diretamente ao estado *SETUP*; (1 ponto)
- Uma vez no estado *SETUP* o usuário deve escolher uma das quatro velocidades de jogo com os Switches 9 e 8, SW(9..8), uma das quatro sequências possíveis de jogo com os Switches 7 e 6, SW(7..6), e o número máximo de iterações por sequência com os Switches 5 e 2, SW(5..2). As frequências para os quatro níveis de jogo serão 0; 5Hz, 1Hz, 2Hz e 3Hz. O nível de jogo será mostrado no HEX4. Damos início ao jogo ativando a entrada *enter* SW(0) e passando ao estado *Play\_FPGA*; (1 ponto)
- Uma vez no estado *PLAY\_FPGA* é ativada a sequência selecionada a qual será mostrada nos LEDR(3..0). Dita sequência possui 16 linhas de atribuição de 4-bits e um exemplo de sequência *SEQ1.v* está disponível no Moodle da disciplina. Corre a cargo do aluno preencher as outras três sequências à sua escolha. É importante destacar que a sequência pode ter apenas um "1" lógico por linha e que a sequência tem estar variando continuamente. Na primeira rodada será apresentada uma linha da sequência, na segunda rodada serão apresentadas duas e assim sucessivamente. O jogo passa para o próximo estado, *PLAY\_USER*, quando estiver ativo um sinal de status, chamado *end\_FPGA*; (1 ponto)
- Uma vez no estado *PLAY\_USER* o usuário deve indicar com os botões de pressão KEY (3..0) a sequência mostrada no estado anterior. Neste estado, o display HEX2 mostrará uma contagem ascendente de 0 a 9 com frequência de 1Hz. Se o tempo acabar é ativado um sinal de *status end\_time* e o jogo passa ao estado *RESULT*, se não terminou o tempo e o usuário introduz a sequência que achar correta então se ativa um sinal de status *end\_user*, o jogo passa ao estado *CHECK*; (1 ponto)
- No estado *CHECK* se avalia se o usuário errou na replicação da sequência. Se o usuário replicar a sequência corretamente, um sinal de status match está ativo e o jogo passa ao estado *NEXT\_ROUND*. Se o usuário errou, passa ao estado *RESULT*. Nesse estado, habilitamos um comando para contar a rodada, que é mostrada no HEX0; (1 ponto)
- No estado *NEXT\_ROUND* se avalia um sinal de status win que indica se o jogo chegou ao último valor da sequência. Se chegou então win está ativo e o jogo passa ao estado *RESULT* caso contrário passa ao estado *PLAY\_FPGA*. Neste estado serão resetadas as contagens das sequências das FPGA e o usuário; (1 ponto)
- No estado *RESULT* será mostrado a pontuação final em Hexadecimal nos displays HEX1 e HEX0. Para o nível de jogo  $j$ , uma sequência  $i$  (selecionados com os SW(9..6)) e o resultado das rodadas, a pontuação final será  $64 \times j + 4 \times rodadas + i$ . Corre a cargo do aluno implementar a dita operação com a menor lógica possível. Nesse estado os displays HEX5, HEX4, HEX3, HEX2 mostrarão *FPgA* ou *USER* indicando quem ganhou o jogo. Nesse estado o usuário deverá pressionar o reset para passar ao estado *START* e iniciar outro jogo; (2 pontos)
- Visando evitar problemas de temporização em função do aperto de um KEY por um ser humano durar muitos ciclos de clock, o Button Press Synchronizer (ButtonSync) será fornecido em conjunto com o projeto. O jogo pode ser reiniciado em qualquer momento com o SW(1), reset; (1 ponto)
- O projeto deve ser implementado obrigatoriamente usando a abordagem datapath-controle. (1 ponto)