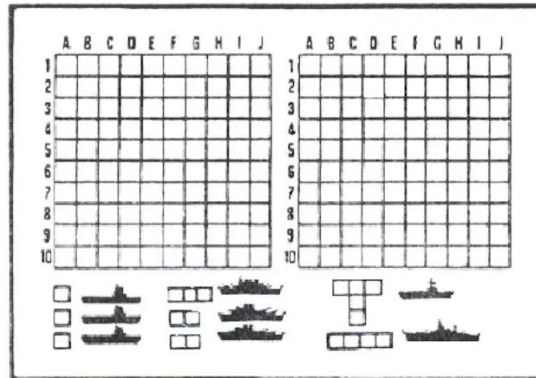


Trabalho de Sistemas Digitais: jogo de batalha naval

Disciplina: Sistemas Digitais

Data limite de entrega: 12/12/2022

Grupo: 3 pessoas



O que entregar: arquivo compactado cujo nome é “nome_1-nome_2” contendo:

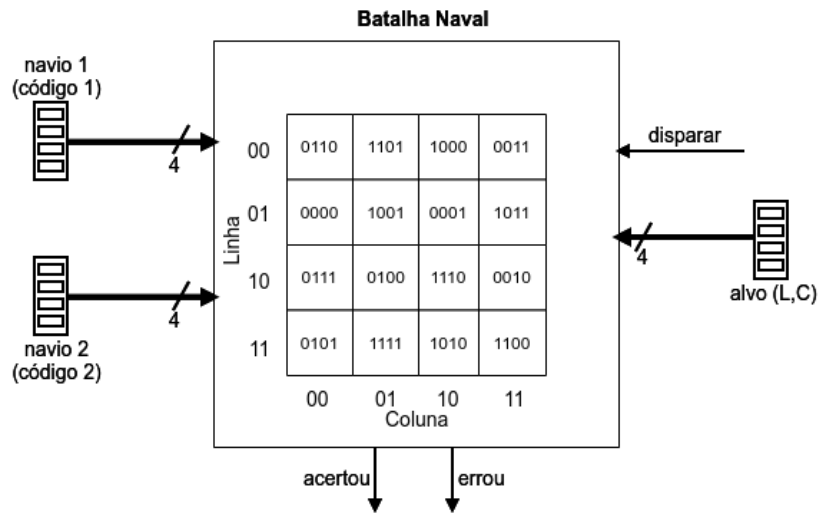
- Relatório incluindo:
 - Apresentação
 - Descrição da solução:
 - Relato da estratégia aplicada
 - Codificação do campo
 - Diagrama de estados de uma FSM
 - Conclusão
- Arquivo do projeto do Quartus

O trabalho proposto simula o jogo de batalha naval. O “mar” é formado por uma matriz de 4x4 posições e o dono deste mar territorial deve colocar 2 navios no mesmo. Destes navios um ocupa uma posição, e o outro duas posições verticais ou horizontais (dois disparos para ser afundado, caso vertical a segunda posição é abaixo da selecionada. Caso horizontal a segunda posição será a direita da selecionada). Para posicionar os navios o dono utiliza 4 chaves (SW) da FPGA onde, de forma codificada, é informada a posição dos navios no mar (cada posição da matriz possui um código associado). Note que não há chaves suficientes para adicionar todos navios ao mesmo tempo (só existem 10 chaves e precisaríamos de 11), sendo assim, cada navio pode ser posicionado utilizando um MUX, ou mesmo estados em uma máquina de estados finita.

Após posicionados os barcos, o adversário terá um conjunto de 4 chaves à sua disposição para informar a posição do alvo e uma entrada para solicitar disparo em direção aos navios. As chaves do adversário não possuem codificação associada e informam diretamente a posição (linha, coluna) onde o disparo é realizado.

Por exemplo, as chaves de posição do alvo informando “1011” indicam o disparo na linha “10”, coluna “11”. Na figura abaixo a posição da linha “1011” está codificada com o valor “0010”. Desta forma, se algum dos navios (entradas “navio_1” ou “navio_2”) possuir o código “0010” a saída acertou deve ir para nível lógico 1 quando a entrada “disparar” for pressionada pelo adversário. Quando o disparo errar o alvo a saída “errou” deve ir para nível lógico 1.

O adversário terá 6 chances de disparo. Se no último disparo o adversário acertar um barco ganha uma chance extra (e assim sucessivamente). Se ao fim dos disparos o adversário não tiver afundado todos os navios, o jogador vence. O número de disparos deve ser apresentado no display presente na placa (em ordem crescente ou decrescente).



De forma resumida, uma solução consiste em;

- 1) Armazenar as posições onde os 3 navios estão ancorados;
- 1) Transformar um código em outro (do código da posição dos navios em posição da linha e coluna onde o código se encontra);
- 2) Comparar a posição decodificada dos navios (linha, coluna), com o valor de linha e coluna do alvo informado pelo adversário.

O código deve ser implementado em VHDL e executado sobre a FPGA. No dia 14/12 a partir das 10:30 os trabalhos serão apresentados através da execução do jogo. O jogo deve apresentar um botão de Reset.

Cada grupo deve indicar a codificação para a matriz que utilizará conforme o arquivo do link a seguir.

https://docs.google.com/document/d/1VYPqGwJSu5y-qvT7QgbjKaPv1Qtke15Y1e_7M1EbA6E/edit?usp=sharing