Alunos: João Vitor Bernardon Schweikart e Henrique Ribeiro Rodrigues

Trabalho Final de Sistemas Digitais

Jogo da Batalha Naval

# Apresentação

Este relatório descreve o processo de construção de um jogo de batalha naval utilizando codificação de máquina de estados. Basicamente, o jogo consiste na presença de dois navios no “mar”, sendo esse uma matriz codificada de 4x4 posições. Para posicionar os navios (sendo que um deles ocupa apenas uma posição e outro duas posições – verticais ou horizontais), o “dono do mar” utiliza 4 chaves (SW) referente a linha e coluna desejada (de forma codificada), informando assim a posição do seu navio no mar (cada posição da matriz possui um código associado). Da mesma forma, o outro jogador informa onde efetuará o disparo para “derrubar” um dos navios do inimigo através da utilização das 4 chaves (SW) referentes a linha e coluna (desta vez não-codificado) onde se imagina em que o navio adversário está localizado.

A partir do código VHDL executado sobre a FPGA, o sistema criado irá retornar um led vermelho em caso de erro no tiro do jogador ou um led verde em caso de acerto. O número de disparos que faltam também aparecerá no display de 7 segmentos da placa (no máximo 6 tentativas de tiro – caso o jogador acerte um dos navios na última tentativa, ganha uma chance bônus).

# Relato da Estratégia Aplicada:

Para a resolução do jogo, foi aplicado primeiramente no circuito a decodificação da posição em que fica os navios. Para isso, foi construída a tabela-verdade (tópico 4) em relação ao campo codificado (tópico 3). A partir da tabela-verdade, foi aplicado Karnaugh para obter a simplificação dos 4 bits de saída, totalizando 4 expressões principais simplificadas.

No código VHDL que será executado na placa FPGA, foi montada a arquitetura de portas lógicas e led’s utilizados para entrada e saída do sistema, além de definir os estados e as variáveis que compõe a máquina. Além disso, após a simplificação da lógica do sistema através do Karnaugh, foram construídas as condições e possibilidades no código em que as coordenadas setadas pelo usuário se encontrariam com as coordenadas dos navios no mar.

# Codificação do Campo e Tabela Verdade

Foi utilizada a matriz codificada número 22 para a construção do circuito:

|  |  |  |  |
| --- | --- | --- | --- |
| 1001 | 0110 | 1111 | 1101 |
| 0000 | 1011 | 0001 | 0011 |
| 1110 | 0100 | 0010 | 1010 |
| 0111 | 1000 | 1100 | 0101 |

A partir da codificação do campo acima, foi construída a seguinte tabela verdade, transformando o código da posição dos navios em posição da linha e coluna onde o mesmo código se encontra:

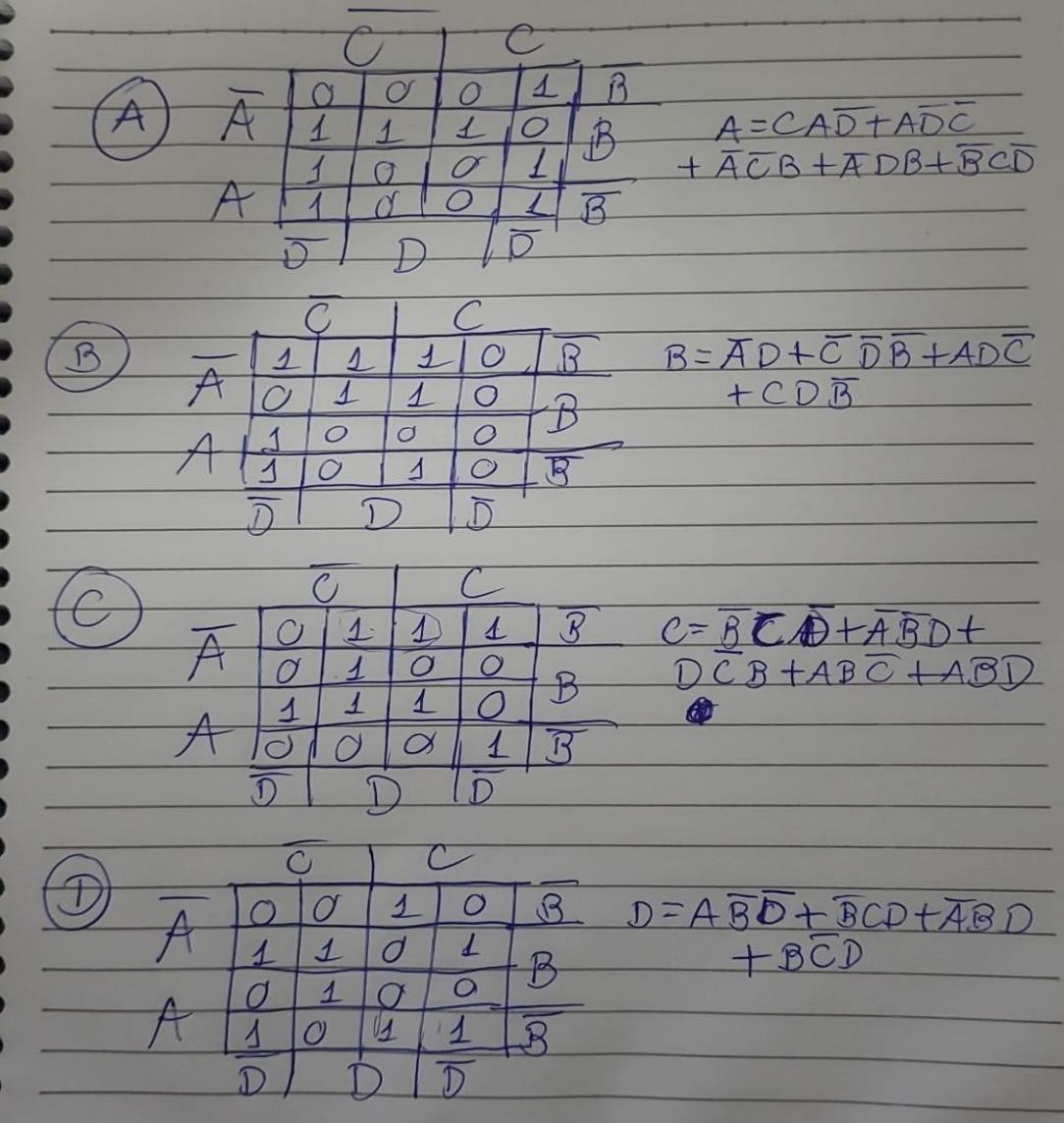
|  |  |
| --- | --- |
| **ENTRADA (POSIÇÃO DO**  **NAVIO CODIFICADO)** | **A B C D (POSIÇÃO DO NAVIO EM**  **LINHA E COLUNA DESCODIFICADO)** |
| 0 0 0 0 | 0 1 0 0 |
| 0 0 0 1 | 0 1 1 0 |
| 0 0 1 0 | 1 0 1 0 |
| 0 0 1 1 | 0 1 1 1 |
| 0 1 0 0 | 1 0 0 1 |
| 0 1 0 1 | 1 1 1 1 |
| 0 1 1 0 | 0 0 0 1 |
| 0 1 1 1 | 1 1 0 0 |
| 1 0 0 0 | 1 1 0 1 |
| 1 0 0 1 | 0 0 0 0 |
| 1 0 1 0 | 1 0 1 1 |
| 1 0 1 1 | 0 1 0 1 |
| 1 1 0 0 | 1 1 1 0 |
| 1 1 0 1 | 0 0 1 1 |
| 1 1 1 0 | 1 0 0 0 |
| 1 1 1 1 | 0 0 1 0 |

# Diagrama de Estados da FSM:

A partir da codificação do campo acima, foi construída a seguinte tabela verdade, transformando o código da posição dos navios em posição da linha e coluna onde o mesmo código se encontra:

# Simplificações:

Após a construção tabela verdade, foi aplicado Karnaugh em cada uma das 4 saídas obtidas para simplificar o circuito:



# Conclusão:

Descrever conclusão do trabalho