

Universidade Tecnológica Federal do Paraná
Campus Apucarana
Engenharia de Computação

Luis Henrique Ferracciu Pagotto Mendes, RA: 2272016

Relatório 3

Display de 7 segmentos

LRCO7A - Lógica Reconfigurável
Professor: Marcelo de Oliveira

Apucarana - PR
2024

1 Introdução:

Este projeto utiliza FPGA para controlar dois displays de 7 segmentos, exibindo números de 0 a 9 com base na configuração de 8 switches (organizados em dois pares de 4 bits). Cada par de switches controla um display de forma independente, de modo que ambos funcionam de maneira idêntica para exibir números conforme a entrada binária.

Displays de 7 segmentos são comumente usados para representar dígitos decimais em formato digital. Eles consistem em sete LEDs organizados em forma de "8", permitindo formar qualquer número de 0 a 9 ao ativar diferentes combinações de segmentos (0, 1, 2, 3, 4, 5, 6 + .). No projeto, as combinações de 4 bits dos switches determinam quais segmentos acendem para representar o número desejado.

2 Código VHDL comentado:

Para o desenvolvimento do código, foram definidas portas de entrada, para a utilização dos switches, e portas de saída para ligar os segmentos do display e gerar os resultados esperados. Além disso, para este projeto, foram utilizados duas instruções essenciais: `select` e `when`.

No código, foram definidos dois vetores (`switches1` e `switches2`) com tamanho de 4 bits, o que permite representar os números de 0 a 9. Os dois conjuntos de switches trabalham de forma independente, desta forma, é possível atribuir os valores corretos de entrada e saída, se interferências entre eles.

Agora, para a representação dos números nos displays de 7 segmentos, também foram definidos dois vetores(`SSD1` e `SSD2`), mas este com tamanho de 8 bits, representando todos os segmentos do display em conjunto com o ".", também presente no mesmo.

Para o funcionamento do projeto, foram utilizadas as instruções `select` e `when`, que atuam de maneira simples. Ao definir o conjuntos de switches que exercerão as atividades, utilizando o comando `select`, define-se o que será feito. No caso do vetor `switches1`, o `SSD1` exibirá as saídas respectivas para cada entrada, a partir do comando `when`. O mesmo é feito para o vetor `switches2`.

Abaixo está uma imagem do código implementado, com comentários organizando cada funcionalidade do código, assim como as entradas e saídas esperadas para o projeto.

```

1  -----PACOTES-----
2  library ieee;
3  use ieee.std_logic_1164.all
4  -----ENTIDADES-----
5
6  entity projeto3 is
7
8  port (
9      switches1, switches2 : in std_logic_vector (3 downto 0); --vetor que armazena as combinações dos switches em binário permitindo
10     --gerar os valores de 0 a 9
11     SSD1, SSD2 : out std_logic_vector (7 downto 0)); --armazena o byte que representa o display permitindo acender apenas os segmentos
12     --desejados
13 end entity;
14 -----PROJETO-----
15 architecture projeto3 of projeto3 is --utiliza a arquitetura do próprio projeto para definir a função e operação entre as portas usadas
16
17 begin --inicia o bloco de comando do projeto
18     --bloco de código que atua na combinação dos 4 primeiros switches e ativa o primeiro display (da direita para a esquerda)
19     with switches1 select
20
21     --display da direita
22     --cada bit do primeiro conjunto representa um segmento do display --cada bit do segundo conjunto representa um switch da placa,
23     --da direita para a esquerda
24     SSD1 <= "11000000" when "0000", --0 --ativa os segmentos 0,1,2,3,4,5
25     "11111001" when "0001", --1 --ativa os segmentos 1,2
26     "10100100" when "0010", --2 --ativa os segmentos 0,1,3,4,6
27     "10110000" when "0011", --3 --ativa os segmentos 0,1,2,3,6
28     "10011001" when "0100", --4 --ativa os segmentos 1,2,5,6
29     "10010010" when "0101", --5 --ativa os segmentos 0,2,3,5,6
30     "10000010" when "0110", --6 --ativa os segmentos 0,2,3,4,5,6
31     "11111000" when "0111", --7 --ativa os segmentos 0,1,2
32     "10000000" when "1000", --8 --ativa todos os segmentos
33     "10010000" when "1001", --9 --ativa os segmentos 0,1,2,3,5,6
34     "00000000" when others;--tudo
35     --bloco de código que atua na combinação dos 4 próximos switches e ativa o segundo display (da direita para a esquerda)
36     with switches2 select
37
38     --display da esquerda
39     --cada bit do primeiro conjunto representa um segmento do display --cada bit do segundo conjunto representa um switch da placa,
40     --da direita para a esquerda, os switches 5, 6,
41     SSD2 <= "11000000" when "0000", --0 --ativa os segmentos 0,1,2,3,4,5
42     "11111001" when "0001", --1 --ativa os segmentos 1,2
43     "10100100" when "0010", --2 --ativa os segmentos 0,1,3,4,6
44     "10110000" when "0011", --3 --ativa os segmentos 0,1,2,3,6
45     "10011001" when "0100", --4 --ativa os segmentos 1,2,5,6
46     "10010010" when "0101", --5 --ativa os segmentos 0,2,3,5,6
47     "10000010" when "0110", --6 --ativa os segmentos 0,2,3,4,5,6
48     "11111000" when "0111", --7 --ativa os segmentos 0,1,2
49     "10000000" when "1000", --8 --ativa todos os segmentos
50     "10010000" when "1001", --9 --ativa os segmentos 0,1,2,3,5,6
51     "00000000" when others;--tudo
52
53 end architecture;

```

Figura 1: Código VHDL comentado.

3 Foto da Placa com os Displays Apresentando os Números:

Abaixo está uma foto tirada para demonstrar o funcionamento da placa. Para definir o valor representado, o vetor switches1 está definido em '0000' (na imagem este vetor é o conjunto dos 4 primeiros switches da direita para a esquerda) e o vetor switches2 está definido em '0010' (os 4 próximos switches, também da direita para a esquerda).

Já, referente aos displays, aquele que está apresentando o número 0 é ativo a partir do vetor switches1 que ativa os segmentos 0, 1, 2, 3, 4 e 5. O display que representa o número 2 funciona com o vetor switches2 que ativa os segmentos 0, 1, 3, 4 e 6.



Figura 2: Foto da placa em funcionamento.

4 Listagem do Pin Planner:

A utilização da placa FPGA só é possível após definir a função de cada pino da mesma. O pin planner, presente no Quartus é a ferramenta que torna isso viável, que faz a listagem de todas as entradas e saídas definidas no código e, a partir disso, é possível configurar cada uma para um componente da placa.

As imagens abaixo mostram como estão configurados os componentes usados no projeto (switches e displays). A imagem da esquerda é uma representação gráfica e matricial de onde estão definidas as entradas e saídas utilizadas (em vermelho). A imagem da direita mostra a lista que define, os switches e displays usados, mostrado na primeira coluna, e a forma de definir o que deverá ser configurado, mostrado na terceira coluna.

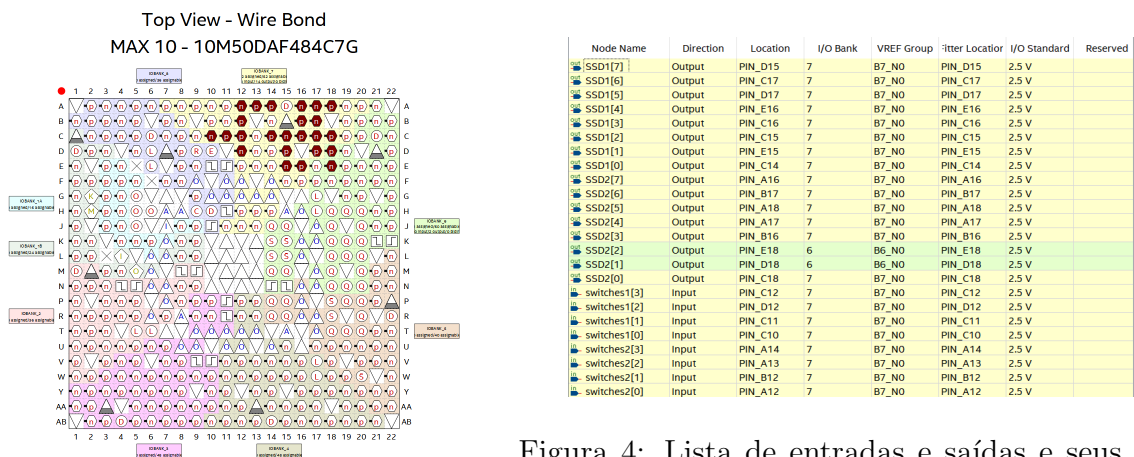


Figura 3: Pin Planner.

Figura 4: Lista de entradas e saídas e seus respectivos pinos.

5 Diagrama RTL:

O circuito gerado pelo código foi representado em duas partes, pois a construção do projeto foi realizada com dois conjuntos de entrada e saída.

As imagens abaixo mostram que, para este projeto, foram utilizados multiplexadores para gerar a saída desejada a partir dos switches ativos.

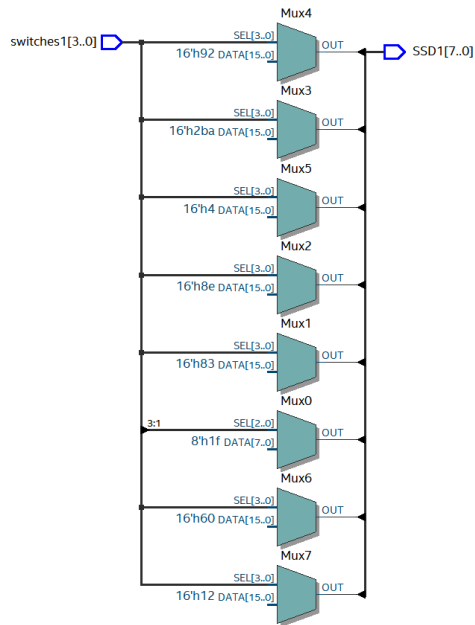


Figura 5: Estrutura 1 para o conjunto de switches1 e seu respectivo display.

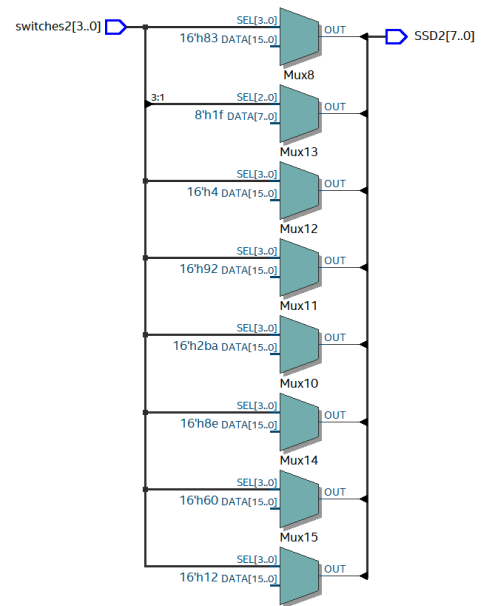


Figura 6: Estrutura 2 para o conjunto de switches2 e seu respectivo display.