

Universidade Tecnológica Federal do  
Paraná  
Engenharia da Computação  
Lógica Reconfigurável

**Relatório 2 - Introdução à placa  
DE10-Lite**

Aluno: Eduardo Yuji Yoshida Yamada  
Professor orientador: Marcelo de Oliveira

Outubro  
2024

# Conteúdo

1	Introdução	1
2	Códigos	2
3	Tabela, Mapas e Equação	3
4	Simulações	4
5	Diagramas RTL	6

# 1 Introdução

A atividade contém dois exercícios, sendo o primeiro, uma continuação da *Atividade 1: Portas lógicas*, mas dessa vez implementando na placa DE10-Lite, sendo assim foi implementado:

- Associar cada entrada (a e b) em uma chave;
- Associar um LED para cada uma das saídas.

Já para o segundo exercício foi considerado o circuito mostrado na figura 1, sendo um circuito de uma máquina de cópias. As chaves são encontradas normalmente abertas e, quando o papel passa por uma chave, a chave fecha. Considerando que é impossível que as chaves SW1 SW4 sejam fechadas ao mesmo tempo. Dessa forma o circuito produzirá um sinal alto de saída sempre que duas ou mais chaves estiverem fechadas ao mesmo tempo.

Dessa forma o exercício pede algumas implementações:

- Encontrar a tabela-verdade do circuito (utilizando *don't care* para simplificar);
- Transpor a tabela para o mapa de Karnaugh e encontrar a equação booleana;
- Implementar o circuito em VHDL e fazer a simulação;
- Por fim, implementar o circuito na placa, com as chaves atuando como as entradas e um LED sendo aceso quando a saída for alta.

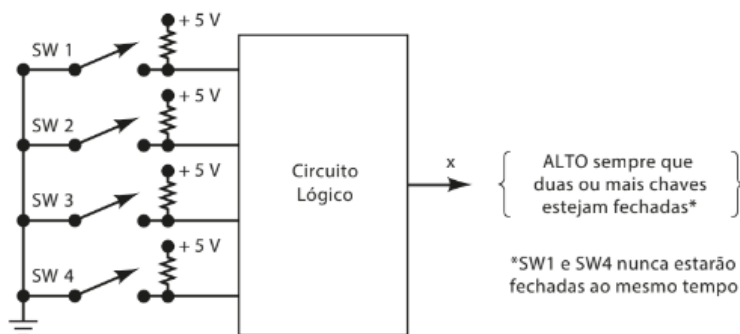


Figura 1: Circuito da máquina de cópias

## 2 Códigos

Para o segundo exercício, temos o seguinte código:

---

```
library ieee ;
use ieee . std_logic_1164 . all ;

entity projeto2 is
    port (
        a , b , c , d: in bit ;
        z: out bit
    );
end entity ;
```

---

```
architecture projeto2 of projeto2 is
    signal x: std_logic_vector (7 downto 0) ;
begin
    z <= ((not a) and (not b)) or ((not a) and (not c)) or
        ((not a) and (not d)) or ((not b) and (not c)) or
        ((not b) and (not d)) or ((not c) and (not d));
end architecture ;
```

---

Nesse código temos as portas *a*, *b*, *c* e *d* como portas de entrada e a porta *z* como saída. A porta *z* receberá apenas uma das entradas, *not a* e *not b* ou *not c* e *not d*, por exemplo.

### 3 Tabela, Mapas e Equação

A	B	C	D	Z
0	0	0	0	X
0	0	0	1	1
0	0	1	0	X
0	0	1	1	1
0	1	0	0	X
0	1	0	1	1
0	1	1	0	X
0	1	1	1	0
1	0	0	0	1
1	0	0	1	1
1	0	1	0	1
1	0	1	1	0
1	1	0	0	1
1	1	0	1	0
1	1	1	0	0
1	1	1	1	0

Tabela 1: Tabela-verdade do exercício 2

	$\overline{CD}$	$\overline{C}\overline{D}$	$CD$	$C\overline{D}$
$\overline{AB}$	X	1	1	X
$\overline{A}\overline{B}$	X	1	0	X
$AB$	1	0	0	0
$A\overline{B}$	1	1	0	1

Tabela 2: Tabela-verdade do exercício 2

Dessa forma obtemos a seguinte equação booleana o exercício 2:  $\overline{AB} + \overline{AC} + \overline{AD} + \overline{BC} + \overline{BD} + \overline{CD}$

# 4 Simulações

Foi realizado a simuação do circuito do exercício 2 para mostrar as diferentes formas de saída. Além do Pin Planner dos dois circuitos

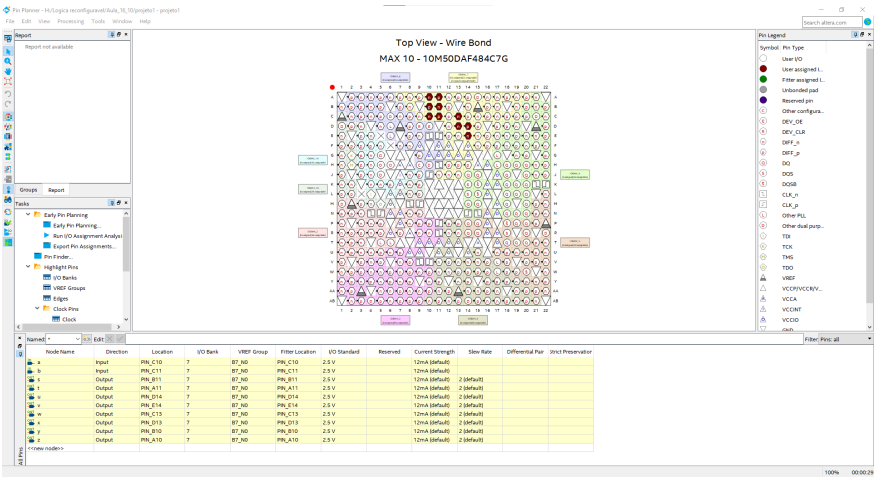


Figura 2: Pin Planner exercício 1

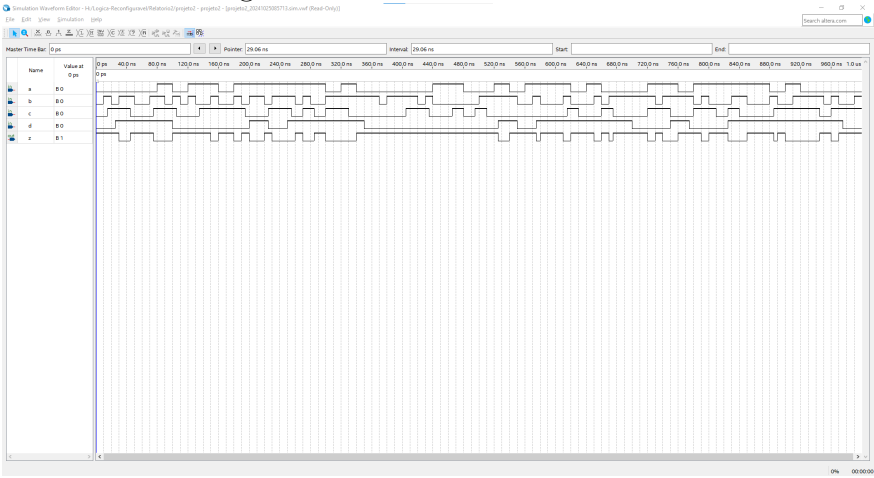


Figura 3: Simulação exercício 2

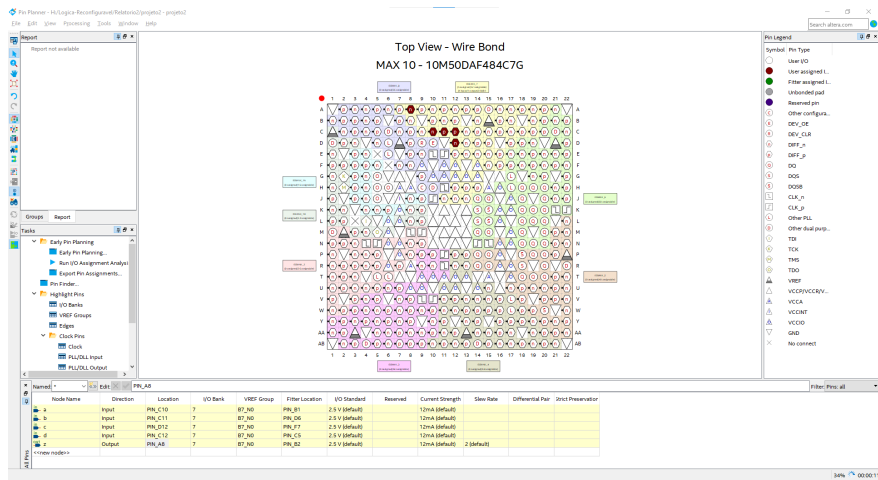
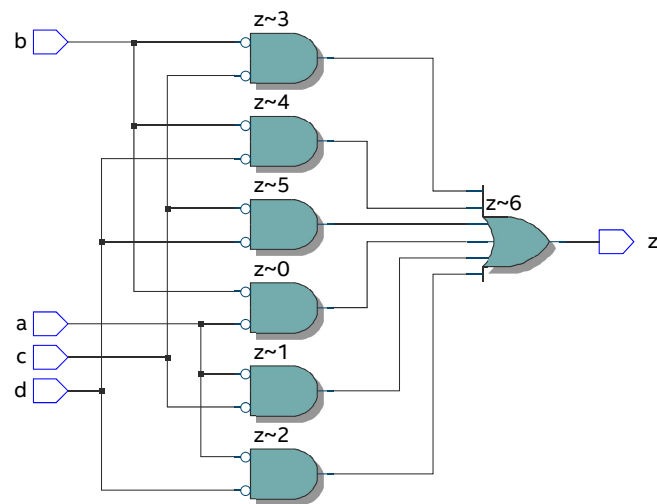


Figura 4: Pin Planner exercício 2

## 5 Diagramas RTL

Date: October 25, 2024

Project: projeto2



Page 1 of 1

Revision: projeto2

Figura 5: RTL Viewer exercício 2