

<b>Disciplina:</b> PCS 3335 – Laboratório Digital A	
<b>Prof.:</b> <i>Glauber De Bona</i>	<b>Data:</b> 13/06
<b>Turma:</b> <i>Glauber - T04</i>	<b>Bancada:</b> 08
<b>Membros:</b>	
<i>11261531 - Enzo Bustos Da Silva</i>	
<i>10379694 - Davi Augusto Bandeira</i>	



## ***Experiência 08***

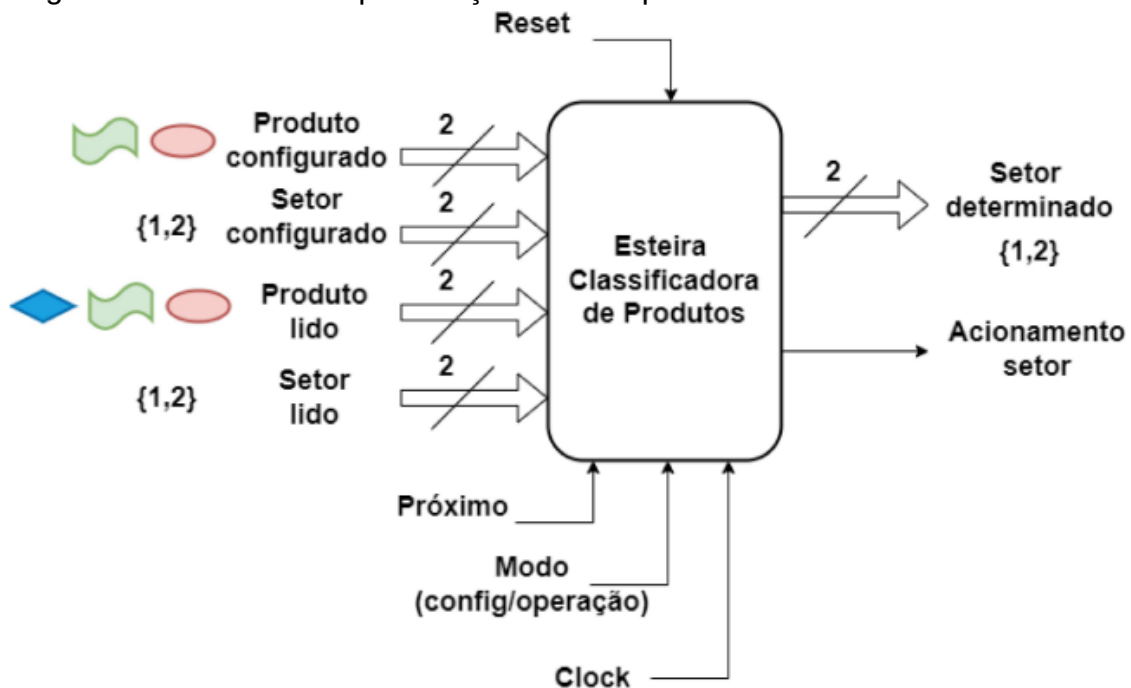
### ***Esteira Classificadora de Produtos II***

## 1. Introdução

A experiência 8 do Laboratório Digital tem como objetivo desenvolver a unidade de controle da esteira de classificação de produtos feita na experiência passada, finalizando o projeto.

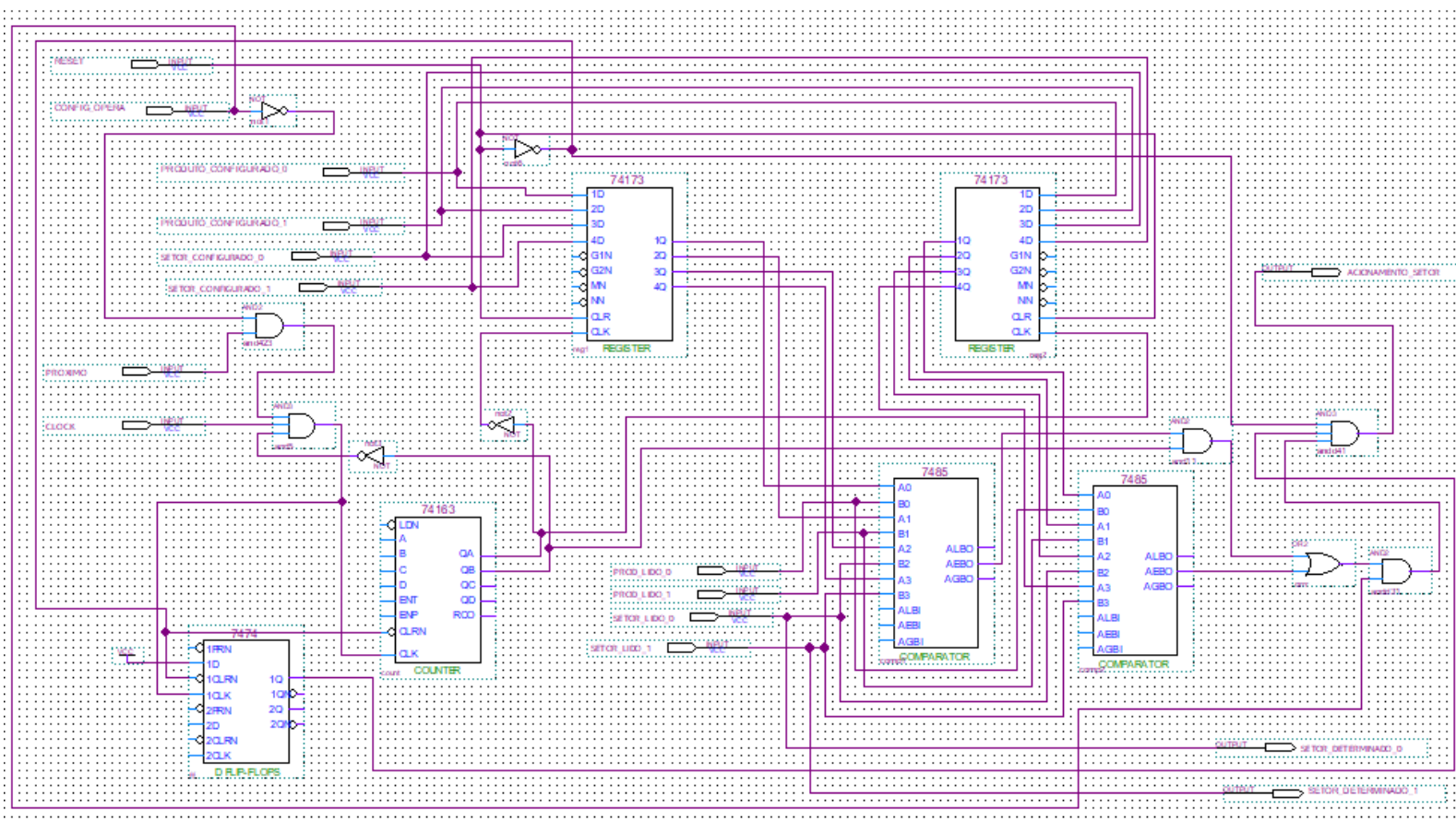
## 2. Objetivo

Nesse contexto, o objetivo desta experiência é adicionar uma nova entrada “*Modo (config/operação)*” que, caso esteja em baixo, só permite configurar um produto/setor (mas não permite a leitura) e, caso em alto, só permite ler e não configurar. As demais especificações da experiência anterior serão mantidas.



### 3. Planejamento

### a. Diagrama Lógico



### **b. Descrição funcional da Unidade de Controle**

De acordo com o diagrama de blocos acima, a unidade de controle foi acrescentada ao fluxo de dados previamente projetado. Desta forma, a UC consiste na adição da entrada “*config\_opera*” que leva o sistema a 2 estados possíveis: um de configuração e outro de operação (leitura). É importante ressaltar que ela não permite que ambos processos ocorram simultaneamente, pois estes são excludentes, em função do valor da entrada.

*config\_opera* = 0 => leitura irrelevante, acionamento não pode ser acionado, dado que o circuito está em processo de configuração;

config\_opera = 1 => configuração irrelevante, acionamento será acionado caso a leitura seja igual à configuração previamente definida.

### c. Unidade de Controle em VHDL

```
library IEEE;
use IEEE.std_logic_1164.all;
use IEEE.numeric_std.all;

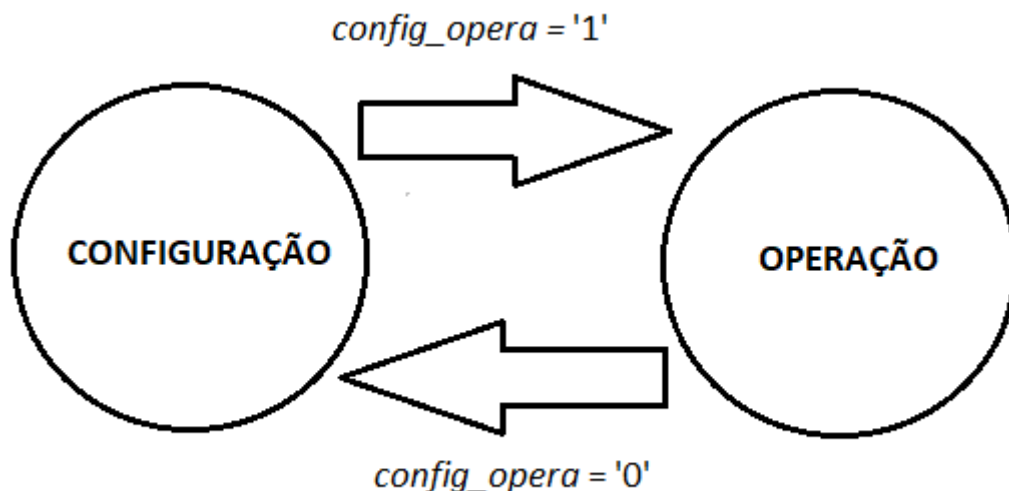
entity esteira_vhdl is
    port(config_opera: in std_logic);
end esteira_vhdl;

architecture arch_esteira of esteira_vhdl is
    type state_type is (configuracao, operacao);
    signal current_state: state_type;

begin
    define_estado: process (config_opera, current_state)
    begin
        if (config_opera = '0') then
            current_state <= configuracao;
        else
            current_state <= operacao;
        end if;
    end process define_estado;
end arch_esteira;
```

### d. Diagrama de Estados

Segue o diagrama de estados da Unidade de Controle. Por algum motivo, a funcionalidade “*State Machine Viewer*” do Quartus não estava mostrando o diagrama, mesmo colocando um “*dummy state*” (estado sem função apenas para cumprir o requisito do Quartus de se ter no mínimo 3 estados para produzir o diagrama).



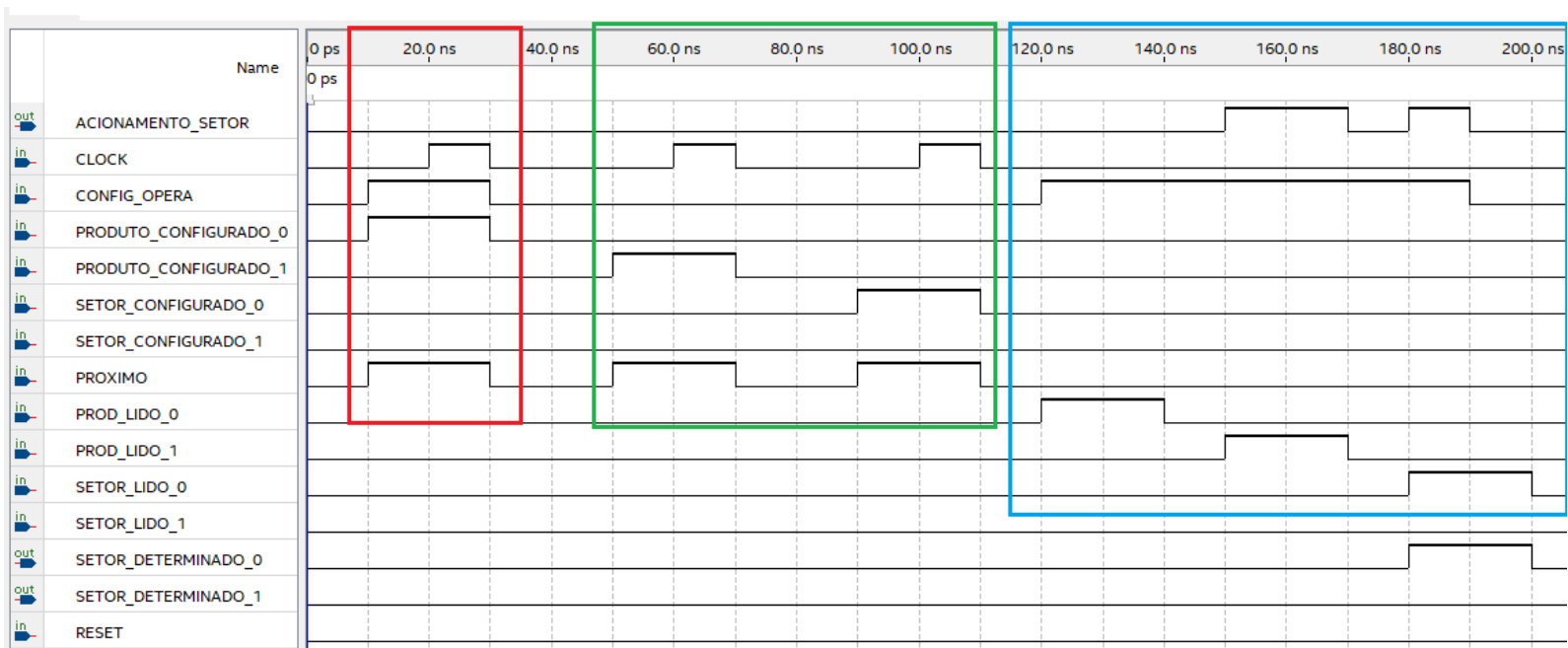
## e. Carta de Tempos

Segue abaixo a simulação do circuito completo com a Unidade de Controle. Dividiu-se esta simulação em 3 blocos para poder observar a dinâmica da entrada “*config\_opera*”.

No bloco vermelho, observa-se que o sistema está no estado de operação (leitura), pois *config\_opera* = 1. Desta forma, é dado um sinal de *Próximo* e *Clock*, para que se tente configurar o produto “01” ao setor “00”. Vale destacar que o circuito não deve realizar esta configuração, dado que está em estado de leitura, o que será testado mais adiante.

No bloco verde, o sistema se encontra em estado de configuração, pois *config\_opera* = 0. Portanto, se configura, de fato, o produto “10” ao setor “00” e, em seguida, o produto “00” ao setor “01”.

O último bloco azul serve de teste para os blocos anteriores. O circuito se encontra no estado de leitura (*config\_opera* = 1). Inicialmente, aos  $t = 120\text{ns}$ , se tenta ler o produto/setor configurados no bloco vermelho e, de fato, o setor não é acionado, como esperado, pois este estado não foi configurado a priori. Finalmente, é lido em seguida as 2 configurações definidas no bloco verde, acionando o setor conforme esperado. Vale destacar em  $t = 190\text{ns}$  que no momento que a entrada *config\_opera* se anula, o acionamento também vai a zero, mesmo que a leitura ainda continuava na configuração, o que é condizente com a condição do projeto.



## f. Tabela de Testes

Foi elaborada uma tabela de testes para descrever a carta de tempos apresentada e comentada no item anterior:

RESET	CLOCK	MODO (CONFIG/ OPERA)	PRODUTO CONFIGURADO	SETOR CONFIGURADO	PRODUTO LIDO	SETOR LIDO	PRÓXIMO	SETOR DETERMINADO	ACIONAMENTO
0	1	1	XX	XX	00	00	X	00	0
0	1	0	10	00	XX	XX	1	00	0
0	1	0	00	01	XX	XX	1	00	0
0	X	1	XX	XX	01	00	X	00	0
0	X	1	XX	XX	10	00	X	00	1
0	X	1	XX	XX	00	01	X	01	1

#### 4. Relatório

Infelizmente, mesmo tendo feito a montagem prevista e simulada no planejamento, a dupla não foi capaz de conseguir apresentar os resultados obtidos ao professor em aula, visto que a montagem não foi bem sucedida durante o período do laboratório. Foram realizados testes com sinais intermediários para tentar checar algum eventual problema, além de checagens de conexões e 2 remontagens, mas ainda assim o resultado obtido não saiu como simulado.

Ademais, cabe ressaltar que a Unidade de Controle do planejamento está demasiadamente trivial por só possuir 2 estados, pois toda a lógica do projeto foi implementada no diagrama de blocos, isto é, o Fluxo de Dados encapsulou a UC, o que não era o esperado pelo professor. Desta forma, parte da lógica do Fluxo de Dados deveria ser compensada para que fosse implementada em VHDL na UC. Cabe ressaltar que, durante aula, a montagem que não foi feita era exatamente o Fluxo de Dados que, por encapsular toda a lógica do projeto, ficou mais complexo que o necessário e foi um dos fatores determinantes ao fracasso na montagem.