



# ISEL

Departamento de Engenharia  
Eletrónica e Telecomunicações  
e de Computadores

Licenciatura em Engenharia Informática e de Computadores  
e  
Licenciatura em Engenharia Informática, Redes e Telecomunicações

*O relatório deve ser entregue no formato PDF.*

## Relatório do 1º Trabalho

\*

## Circuitos Combinatórios

Trabalho realizado por:

Nome: André Brito                      N° 50487

Nome: David Seves                      N° 50495

Nome: Leonel Correia                      N° 50552

Turma: LEIC13D

Docente: João P. Patriarca

Lógica e Sistemas Digitais  
2022 / 2023 inverno

01 de novembro de 2022

## 1 Objetivo

O objetivo deste trabalho é descrever um circuito combinatório *Priority Encoder* 8:3 com VHDL estrutural, simular e implementar o circuito com a placa de desenvolvimento DE10-Lite da Intel.

## 2 Descrição do Circuito a Projetar

Pretende-se implementar um circuito *priority encoder* com 8 entradas e 3 saídas (PEnc 8:3), que tem as entradas e saídas apresentadas na Figura 1. Este circuito foi construído à base de blocos funcionais que implementam um *priority encoder* com 4 entradas e 2 saídas cada (PEnc 4:2) e um *Multiplexer*, também este com 4 entradas e 2 saídas.

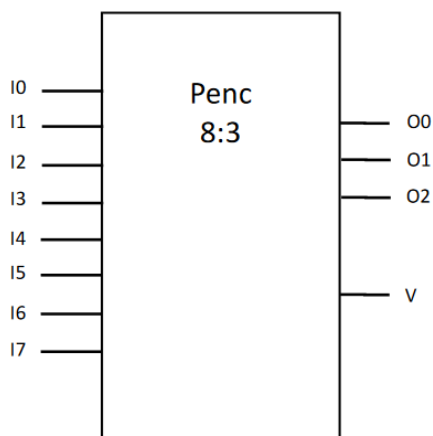


Figura 1: Representação das entradas e saídas do *Priority Encoder* 8:3

Quando uma das entradas I (I0, I1, I2, I3, I4, I5, I6 e I7) está ativa, a saída O (O2, O1 e O0) apresenta o código correspondente ao índice da entrada ativa (e.g. quando a entrada I3 estiver ativa, a saída O apresenta o valor 011). Caso mais do que uma entrada esteja ativa simultaneamente, o código na saída O corresponde ao índice da entrada que tem maior prioridade. A prioridade é definida com base no índice da entrada, sendo que ao maior índice corresponde a prioridade mais alta. Sempre que uma das entradas estiver ativa, a saída V estará ativa.

### 3 Desenvolvimento do Projeto

#### 1. Funções lógicas dos circuitos;

Cada Priority Encoder 4:2 utilizado como bloco neste projeto é definido através das funções seguintes:

$$O(0) \leq I3 \text{ or } (\text{not } I2 \text{ and } I1);$$

$$O(1) \leq I3 \text{ or } I2;$$

$$V \leq I3 \text{ or } I2 \text{ or } I1 \text{ or } I0;$$

Desta maneira, a saída O de 2 bits representa a codificação do valor inteiro inserido através da escolha da entrada ativada em binário. Já a saída V indica o estado de funcionamento deste bloco PEnc, um valor lógico 1 indica que há pelo menos uma entrada ativa e um valor lógico 0 indica que não há nenhuma entrada ativa.

O bloco *Multiplexer* 4:2 utilizado neste projeto pode ser representado pelas seguintes funções:

$$R(1) \leq (\text{Not sel and PEnc1}(1)) \text{ or } (\text{sel and PEnc2}(1));$$

$$R(0) \leq (\text{Not sel and PEnc1}(0)) \text{ or } (\text{sel and PEnc2}(0));$$

A saída de índice 0 de R indica se a entrada PEnc1(0) está ativa enquanto o seletor está desativo ou se a entrada PEnc2(0) está ativa enquanto o seletor está ativo. A saída de índice 1 de R apresenta os mesmos resultados, mas em relação às entradas de índice 1 de PEnc1 e PEnc2.

(seja sel um sinal interno de PEnc 8:3 que representa o seletor do multiplexer, PEnc1 um sinal interno de PEnc 8:3 de 2 bits que representa a saída O do bloco PEnc 4:2 unidade 1 ligado a este multiplexer e PEnc2 um sinal interno de PEnc 8:3 de 2 bits que representa a saída O do bloco PEnc 4:2 unidade 1 ligado a este multiplexer.)

? Esta meio desengenhado

## 2. Simplificação lógica

Podemos simplificar logicamente o circuito implementado, PEnc 8:3, às expressões:

$$O(0) \leq [V2' \times O1(0)] + [V2 \times O2(0)]$$

$$O(1) \leq [V2' \times O1(1)] + [V2 \times O2(1)]$$

$$O(2) \leq V2$$

Em que:

$$V2 \leq I4 + I5 + I6 + I7$$

$$O1(0) \leq [I3 + (I2' \times I1)]$$

$$O1(1) \leq I2 + I3$$

$$O2(0) \leq [I7 + (I6' \times I5)]$$

$$O2(1) \leq I7 + I6$$

*Complicaram!*  
*A simplificação devia corresponder à*  
*tabela de verdade de PENC 4x2*  
*e respectivos mapas.*

A saída de índice 0 do PEnc 8:3 ( $O(0)$ ) indica se a saída de índice 0 do PEnc 4:2 unidade 1 ( $O1(0)$ ) está ativa enquanto nenhuma das entradas do PEnc 4:2 unidade 2 estão ativas ( $V2'$ ), ou se pelo menos uma das entradas do PEnc 4:2 unidade 2 estiver ativa e a saída de índice 0 do PEnc 4:2 unidade 2 ( $O2(0)$ ) também estiver ativa.

A saída de índice 1 do PEnc 8:3 indica a mesma computação que o de índice 0, mas em relação às saídas dos PEnc's 4:2 de índice 1.

Já a saída de índice 2 indica se pelo menos uma das entradas do PEnc 4:2 unidade 2 está ativa.

## 3. Descrição VHDL

```
LIBRARY IEEE;
```

```
use IEEE.std_logic_1164.all;
```

```
entity PEnc_83 is
```

```
    port(I0,I1,I2,I3,I4,I5,I6,I7:in STD_LOGIC;
```

```
          O:out STD_LOGIC_VECTOR(2 downto 0);
```

```
          V:out std_logic);
```

```
end PEnc_83;
```

```
architecture ARQ_PEnc_83 of PEnc_83 is
```

```
    component PEnc_42
```

```
        port(
```

```
I0 : in STD_LOGIC;  
I1 : in STD_LOGIC;  
I2 : in STD_LOGIC;  
I3 : in STD_LOGIC;  
O : out std_logic_vector(1 downto 0);  
V : out STD_LOGIC);
```

```
end component;
```

```
component Mux_4_2
```

```
port(  
  PEnc1 : in STD_LOGIC_VECTOR(1 downto 0);  
  PEnc2 : in STD_LOGIC_VECTOR(1 downto 0);  
  sel : in std_logic;  
  R : out STD_LOGIC_VECTOR(1 downto 0));
```

```
end component;
```

```
signal o1, o2 : std_logic_vector(1 downto 0);  
signal v1, v2 : std_logic ;  
signal saidaPenc3 : std_logic;  
signal nInteressa: std_logic;
```

```
begin
```

```
UPEnc_42_1: PEnc_42 port map (  
  I0 => I0,  
  I1 => I1,  
  I2 => I2,  
  I3 => I3,  
  O(0) => o1(0),  
  O(1) => o1(1),  
  V => v1  
);
```

```
UPEnc_42_2: PEnc_42 port map (  
  I0 => I4,  
  I1 => I5,  
  I2 => I6,  
  I3 => I7,
```

*Não deviam ter dado estes nomes porque podem ficar desenganchados se quiserem usar este componente em outro projeto. Preferível usarem os nomes A e B ou  $I_p$  e  $I_q$ .*

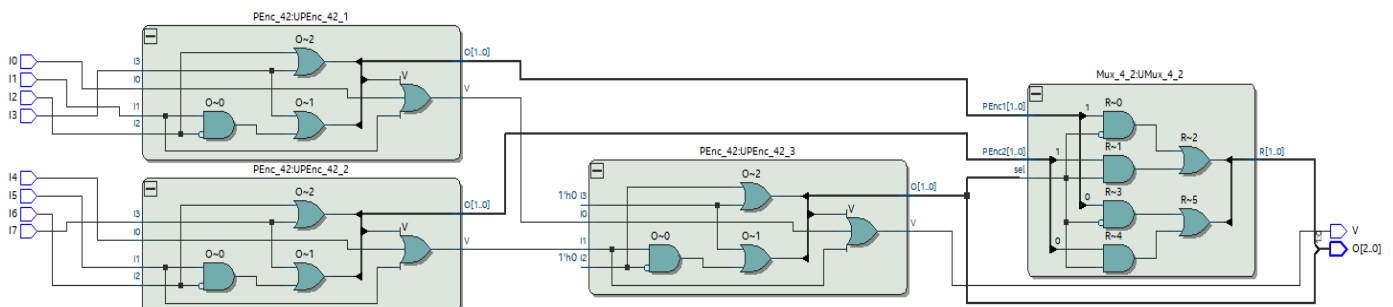
```
O(0) => o2(0),
O(1) => o2(1),
V => v2
);
```

```
UPEnc_42_3: PEnc_42 port map (
I0 => v1,
I1 => v2,
I2 => '0',
I3 => '0',
O(0) => saidaPenc3,
O(1) => nInteressa,
V => V
);
```

```
UMux_4_2 : Mux_4_2 port map(
PEnc1 => o1,
PEnc2 => o2,
sel => saidaPenc3,
R(0) => O(0),
R(1) => O(1)
);
```

```
O(2) <= saidaPenc3;
end ARQ_PEnc_83;
```

#### 4. Diagrama de blocos



## 5. Simulação

/penc_83/I0	0	1	1	1	1	1	1	1	1
/penc_83/I1	0	0	1	1	1	1	1	1	1
/penc_83/I2	0	0	0	1	1	1	1	1	1
/penc_83/I3	0	0	0	0	1	1	1	1	1
/penc_83/I4	0	0	0	0	0	1	1	1	1
/penc_83/I5	0	0	0	0	0	0	1	1	1
/penc_83/I6	0	0	0	0	0	0	0	1	1
/penc_83/I7	0	0	0	0	0	0	0	0	1
+ /penc_83/O	000	000	001	010	011	100	101	110	111
/penc_83/V	0	1	1	1	1	1	1	1	1

Éste interessante e forme como apresentava o resultado de simulação.

Através do simulador ModelSim, fomos capazes de verificar os resultados esperados na simulação: a codificação das entradas I0 a I7 em binário na saída O de forma prioritária, pois ao seleccionar duas ou mais entradas, só a de maior peso é codificada.

## 4 Teste do Circuito

### 1. Atribuição de pinos

set\_location\_assignment PIN\_C10 -to I0  
set\_location\_assignment PIN\_C11 -to I1  
set\_location\_assignment PIN\_D12 -to I2  
set\_location\_assignment PIN\_C12 -to I3  
set\_location\_assignment PIN\_A12 -to I4  
set\_location\_assignment PIN\_B12 -to I5  
set\_location\_assignment PIN\_A13 -to I6  
set\_location\_assignment PIN\_A14 -to I7  
set\_location\_assignment PIN\_A8 -to O[0]  
set\_location\_assignment PIN\_A9 -to O[1]  
set\_location\_assignment PIN\_A10 -to O[2]  
set\_location\_assignment PIN\_B10 -to V

### 2. Resultados do teste (tabela com entradas/saídas do circuito)

I7	I6	I5	I4	I3	I2	I1	I0	O0	O1	O2	V
0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	1	0	0	0	1
0	0	0	0	0	0	1	X	0	0	1	1
0	0	0	0	0	1	X	X	0	1	0	1
0	0	0	0	1	X	X	X	0	1	1	1
0	0	0	1	X	X	X	X	1	0	0	1
0	0	1	X	X	X	X	X	1	0	1	1
0	1	X	X	X	X	X	X	1	1	0	1
1	X	X	X	X	X	X	X	1	1	1	1

## 5 Conclusões

O trabalho consistiu na implementação de um circuito combinatório *Priority Encoder* com 8 entradas e 3 saídas a partir de 3 *Priority Encoders* com 4 entradas e 2 saídas cada e um *Multiplexer*, também este de 4 por 2.

Este circuito foi descrito em VHDL estrutural e testado na placa DE10-Lite da Intel.

A partir dos testes realizados experimentalmente na placa podemos constatar que os resultados experimentais estavam de acordo com os que eram previstos a partir das tabelas de verdade previamente criadas e das simulações executadas no ModelSim, para o circuito implementado.

### A. Descrição Hardware (VHDL)



Mux\_4\_2.vhd      PEnc\_4\_2.vhd      Projeto\_PEnc\_8\_3.vhd

Não constam da entrega.  
Apenas anexo a análise o PENC 8x3 porque conste no relatório.