# **UNIVERSIDADE DE SÃO PAULO**

ESCOLA DE ENGENHARIA DE SÃO CARLOS

# **SEL0384 - Laboratório de Sistemas Digitais 1**

Prof. Maximiliam Luppe

Bárbara Fernandes Madera - nº: 11915032

Johnny Caselato Guimarães - nº: 11915481

**Prática 3 - Relatório de Prática de Laboratório: Dispositivos de Lógica Programável tipo FPGA - Circuitos Combinacionais**

**SÃO CARLOS**

**2023**

1. **Objetivos**

O objetivo desta prática de laboratório é a familiarização com a ferramenta Quartus II da Intel/Altera e a linguagem de descrição de *hardware* VHDL. O domínio destas ferramentas possibilitam o projeto e a síntese de circuitos combinacionais em dispositivos reconfiguráveis, especificamente FPGA, nos quais é possível implementarmos uma série de arquiteturas e componentes que constituem sistemas de processamento mais complexos. Ademais, durante esta atividade será implementado um decodificador binário para 7 segmentos no kit Mercurio® IV (Cyclone® IV EP4CE30F23).

1. **Equipamentos Necessários para Prática:**

* Kit Mercurio® IV
* Software Quartus II Web Edition

1. **Procedimento Experimental:**

**3.1. Apresentação das Equações Booleanas e Diagrama Esquemático:**

Em seguida são apresentadas as equações booleanas de um decodificador binário para 7 segmentos, levando em consideração um display de 7 segmentos do tipo catodo comum (ativo em '1'). Cada segmento teve sua lógica obtida através da simplificação das combinações presentes na tabela verdade abaixo pelos respectivos mapas de *Karnaugh*.

**Tabela 1 - Tabela verdade para o display de 7 segmentos**

| **Tabela verdade para os segmentos** | | | **Segmento** | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Número nas chaves** | **Bin** | **Dec** | **A** | **B** | **C** | **D** | **E** | **F** | **G** |
| **0 0 0 0** | **0** | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| **0 0 0 1** | **1** | 0 | 1 | 1 | 0 | 0 | 0 | 0 |
| **0 0 1 0** | **2** | 1 | 1 | 0 | 1 | 1 | 0 | 1 |
| **0 0 1 1** | **3** | 1 | 1 | 1 | 1 | 0 | 0 | 1 |
| **0 1 0 0** | **4** | 0 | 1 | 1 | 0 | 0 | 1 | 1 |
| **0 1 0 1** | **5** | 1 | 0 | 1 | 1 | 0 | 1 | 1 |
| **0 1 1 0** | **6** | 1 | 0 | 1 | 1 | 1 | 1 | 0 |
| **0 1 1 1** | **7** | 1 | 1 | 1 | 0 | 0 | 0 | 0 |
| **1 0 0 0** | **8** | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| **1 0 0 1** | **9** | 1 | 1 | 1 | 0 | 0 | 1 | 1 |
| **Valores independentes** | **1 0 1 0** | **10** | X | X | X | X | X | X | X |
| **1 0 1 1** | **11** | X | X | X | X | X | X | X |
| **1 1 0 0** | **12** | X | X | X | X | X | X | X |
| **1 1 0 1** | **13** | X | X | X | X | X | X | X |
| **1 1 1 0** | **14** | X | X | X | X | X | X | X |
| **1 1 1 1** | **15** | X | X | X | X | X | X | X |

**Tabela 2 - Mapa de Karnaugh e expressão do segmento A**

| **LED A | Seg(0)** | | **sA sB** | | | |
| --- | --- | --- | --- | --- | --- |
| **0 0** | **0 1** | **1 1** | **1 0** |
| **sC sD** | **0 0** | 1 | 0 | X | 1 |
| **0 1** | 0 | 1 | X | 1 |
| **1 1** | 1 | 1 | X | X |
| **1 0** | 1 | 1 | X | X |
| **Expressão** | | **C + A + BD + !B!D** | | | |

**Tabela 3 - Mapa de Karnaugh e expressão do segmento B**

| **LED B | Seg(0)** | | **sA sB** | | | |
| --- | --- | --- | --- | --- | --- |
| **0 0** | **0 1** | **1 1** | **1 0** |
| **sC sD** | **0 0** | 1 | 1 | X | 1 |
| **0 1** | 1 | 0 | X | 1 |
| **1 1** | 1 | 1 | X | X |
| **1 0** | 1 | 0 | X | X |
| **Expressão** | | **A + !B + !C!D + CD** | | | |

**Tabela 4 - Mapa de Karnaugh e expressão do segmento C**

| **LED C | Seg(0)** | | **sA sB** | | | |
| --- | --- | --- | --- | --- | --- |
| **0 0** | **0 1** | **1 1** | **1 0** |
| **sC sD** | **0 0** | 1 | 1 | X | 1 |
| **0 1** | 1 | 1 | X | 1 |
| **1 1** | 1 | 1 | X | X |
| **1 0** | 0 | 1 | X | X |
| **Expressão** | | **A + B + !C + D** | | | |

**Tabela 5 - Mapa de Karnaugh e expressão do segmento D**

| **LED D | Seg(0)** | | **sA sB** | | | |
| --- | --- | --- | --- | --- | --- |
| **0 0** | **0 1** | **1 1** | **1 0** |
| **sC sD** | **0 0** | 1 | 0 | X | 1 |
| **0 1** | 0 | 1 | X | 0 |
| **1 1** | 1 | 0 | X | X |
| **1 0** | 1 | 1 | X | X |
| **Expressão** | | **C!D + AC + A!D + !BC + !B!D + B!CD** | | | |

**Tabela 6 - Mapa de Karnaugh e expressão do segmento E**

| **LED E | Seg(0)** | | **sA sB** | | | |
| --- | --- | --- | --- | --- | --- |
| **0 0** | **0 1** | **1 1** | **1 0** |
| **sC sD** | **0 0** | 1 | 0 | X | 1 |
| **0 1** | 0 | 0 | X | 0 |
| **1 1** | 0 | 0 | X | X |
| **1 0** | 1 | 1 | X | X |
| **Expressão** | | **C!D + AC + A!D + AB + !B!D** | | | |

**Tabela 7 - Mapa de Karnaugh e expressão do segmento F**

| **LED F | Seg(0)** | | **sA sB** | | | |
| --- | --- | --- | --- | --- | --- |
| **0 0** | **0 1** | **1 1** | **1 0** |
| **sC sD** | **0 0** | 1 | 1 | X | 1 |
| **0 1** | 0 | 1 | X | 1 |
| **1 1** | 0 | 0 | X | X |
| **1 0** | 0 | 1 | X | X |
| **Expressão** | | **A + !C!D + B!C + B!D** | | | |

**Tabela 8 - Mapa de Karnaugh e expressão do segmento G**

| **LED G | Seg(0)** | | **sA sB** | | | |
| --- | --- | --- | --- | --- | --- |
| **0 0** | **0 1** | **1 1** | **1 0** |
| **sC sD** | **0 0** | 0 | 1 | X | 1 |
| **0 1** | 0 | 1 | X | 1 |
| **1 1** | 1 | 0 | X | X |
| **1 0** | 1 | 0 | X | X |
| **Expressão** | | **A + B!C + !BC** | | | |

**3.2. Identificação da Necessidade de Circuitos Integrados TTL7400:**

- Foi realizada uma análise para determinar quantos circuitos integrados da família TTL7400 seriam necessários para implementar o decodificador binário para 7 segmentos. Essa análise leva em consideração a quantidade de portas lógicas necessárias para o circuito conforme a seguinte lista:

* Led A:
  + 1x NOR 3 entradas
  + 1x XOR 2 entradas
  + 2x NOT
* Led B:
  + 1x NOR 3 entradas
  + 1x XOR 2 entradas
  + 3x NOT
* Led C:
  + 1x NOR 4 entradas
  + 2x NOT
* Led D:
  + 1x OR 2 entradas
  + 2x NOR 3 entradas
  + 1x AND 3 entradas
  + 2x AND 2 entradas
  + 5x NOT
* Led E:
  + 2x AND 2 entradas
  + 1x NOR 3 entradas
  + 1x NOR 2 entradas
  + 2x NOT
* Led F:
  + 2x AND 2 entradas
  + 1x OR 2 entradas
  + 1x NOR 3 entradas
  + 3x NOT
* Led G:
  + 1x XOR 2 entradas
  + 1x OR 2 entradas

**3.3. Configuração e Implementação do Projeto no Quartus II:**

Conforme as orientações em sala, criou-se uma pasta chamada "MercurioIV\_decod", na qual o projeto, de mesmo nome, foi inserido.Então, fazendo o uso da linguagem VHDL, o decodificador hexadecimal para 7 segmentos foi implementado conforme a entidade descrita pelo seguinte código:

--Projeto Decodificador Hexa para 7 segmentos

--SEL0384 - Atividade 3

--Autores:

--Johnny Caselato Guimaraes - N: 11915481

--Barbara Fernandes Madera - N: 11915032

--Prof.: Maxmillian Lupe

--data: 13/09/23

—-Este código define a entidade “hex27seg” que representa um

––módulo de hardware o qual converte um valor hexadecimal de 4

--bits em uma exibição de 7 segmentos

ENTITY hex27seg is

PORT(

hexa : in bit\_vector(3 downto 0);--Entrada de 4 bits

segs : out bit\_vector(6 downto 0)--Saída de 7 bits

);

END hex27seg;

—-Definição da arquitetura logica para hex27seg

ARCHITECTURE logica of hex27seg is

BEGIN

segs(0) <= hexa(2) or hexa(0) or (hexa(1) xnor hexa(3));

segs(1) <= hexa(0) or not(hexa(1)) or (hexa(2) xnor hexa(3));

segs(2) <= hexa(0) or hexa(1) or not(hexa(2)) or hexa(3);

segs(3) <= (not(hexa(3)) and (hexa(2) or hexa(0) or not(hexa(1)))) or (hexa(2) and (hexa(0) or not(hexa(1)))) or (hexa(1) and not(hexa(2)) and hexa(3));

segs(4) <= (not(hexa(3)) and (hexa(2) or hexa(0) or not(hexa(1)))) or (hexa(0) and (hexa(1) or hexa(2)))

segs(5) <= (hexa(1) and (not(hexa(2)) or not (hexa(3)))) or (not(hexa(2)) and not(hexa(3))) or hexa(0);

segs(6) <= hexa(0) or (hexa(1) xor hexa(2));

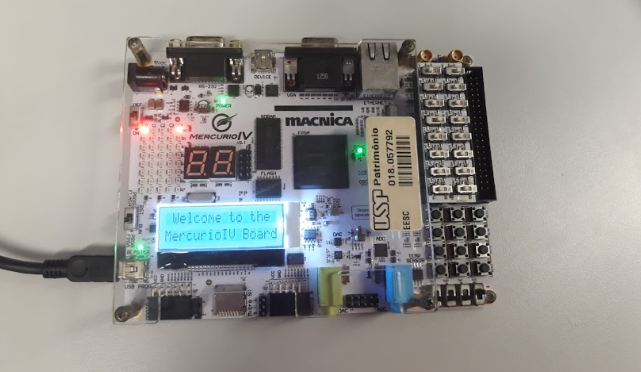
END;

A arquitetura da lógica desta implementação segue de acordo com as equações reduzidas apresentadas nas tabelas 2 a 8 anteriormente, sendo as entradas identificadas pelo vetor hexa() de 4 bits, e as saídas pelo vetor segs() de 7 bits. As expressões booleanas para cada segmento são desenvolvidas utilizando uma série de operadores lógicos, obedecendo as lógicas de hierarquia e execução do VHDL, por isso, faz-se necessário o uso de vários parênteses para delimitar o alcance da lógica.

1. **Resultados obtidos**

O código anexado ao relatório é um algoritmo feito VHDL que define uma estrutura básica para um módulo de hardware que converte um valor hexadecimal de 4 bits em uma exibição de 7 segmentos. Na figura 2 vemos portanto a placa com a devida implementação.

**Figura 1 - Foto da placa Mercurio IV.**



1. **Conclusão:**

Nesta prática de laboratório, foram introduzidos os dispositivos reconfiguráveis, em particular o FPGA baseado em linguagem de descrição de *hardware*, e o software Quartus II da Intel/Altera. Dessa forma, foi projetado e implementado um decodificador binário para 7 segmentos em um display de catodo comum. Isso proporcionou uma compreensão prática da aplicação de circuitos combinacionais em dispositivos FPGA e como usar ferramentas de design para realizar essas implementações.Por fim,verificou-se o funcionamento correto do circuito no kit Mercurio® IV. Esta prática é fundamental para o entendimento dos princípios da lógica programável e da eletrônica digital.