

Prática de Eletrônica Digital 1 - (119466)

Turma E (Unb - Gama)

Pré-Relatório Experimento 6
Introdução ao projeto em FPGA

Outubro 04, 2016

Nome	Matrícula	Assinatura
Arthur Temporim	140016759	
Eduardo Nunes	140056149	

1 Pesquisa bibliográfica

Na seção a seguir contém as atividades pedidas para a elaboração do pré-relatório.

A) As formas de alimentação suportadas pela placa, os níveis de tensão e corrente envolvidos e como esta configuração é feita:

Formas de alimentação: A placa Basys3 pode receber energia da porta Digilent USB-JTAG(J4) ou a partir de uma fonte de alimentação externa 5V.

Níveis de tensão: É necessário limitar a tensão máxima da bateria externa a 5.5V DC. A tensão mínima da bateria depende da aplicação; se a função de host USB (J2) estiver sendo utilizada, pelo menos, 4.6V deve ser fornecida. Em outros casos, a voltagem mínima é de 3.6V.

Corrente envolvidas: É necessário 3.3, 1.8V e suprimentos de 1.0V a partir da entrada 5V de alimentação principal.

Para configurar o FPGA, há três maneiras:

1. O PC pode usar o circuito Digilent USB-JTAG (portJ4, identificado como "PROG") para programar o FPGA para que a qualquer momento o aparelho esteja ligado.

2. Um ficheiro armazenado no de série (SPI) dispositivo flash não volátil pode ser transferido para o FPGA usando a porta SPI.

3. Um arquivo de programação podem ser transferidos a partir de um cartão de memória USB ligado ao HID USB port.

B) As formas de programação suportadas pela placa e como elas podem ser configuradas: Você pode programar o FPGA a partir de um pen drive conectado à porta USB-HID (J2). Ou através da programação JTAG pode ser feito usando o servidor de hardware em Vivado. Ou com um circuito que pode programar dispositivos flash.

C) As formas de armazenamento do bitstream na placa (tipos de memória): A placa Basys3 contém um dispositivo serial flash 32Mbit não-volátil, que está ligado ao FPGA Artix 7 usando um quad-modo dedicado (x4) barramento SPI.

D) A quantidade e frequência dos clocks disponíveis: A placa Basys3 inclui um único oscilador de 100MHz conectada ao pino W5 (W5 é uma entrada no banco de MRCC 34).

E) A identificação (nome) dos pinos usados para as entradas da placa (chaves e pushbuttons) e para as saídas (LEDs e displays de 7-segmentos):

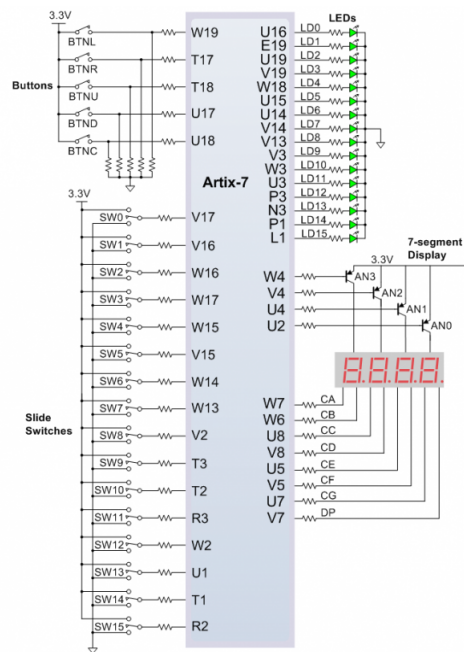


Figure 1: Identificação dos pinos - Ise Design Suit 14.7

2 Projetos e simulações

2.1 Projeto 1

Este circuito consiste na simples exibição em LEDs de acordo com os valores de entrada nos *switchs* de uma placa FPGA.

2.1.1 Diagrama Esquemático

2.1.2 Código VHDL

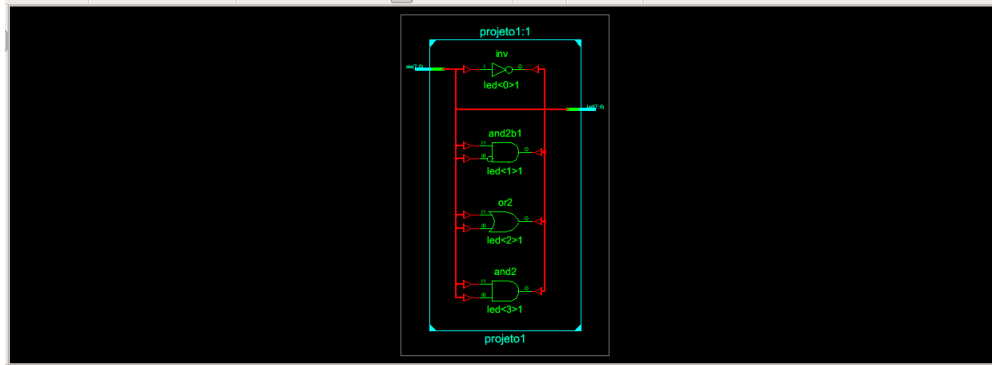


Figure 2: Diagrama 1 - Ise Design Suit 14.7

2.2 Projeto 2

Este circuito consiste em mostra os valores de 2 números de 4 bits cada em 2 displays de BCD utilizando multiplexadores e clocks para representar o resultado.

2.2.1 Código VHDL

```

1 library IEEE;
2 use IEEE.STD_LOGIC_1164.ALL;
3
4 entity projeto2 is
5     port (
6         a      : in  std_logic_vector (3 downto 0) := "0001";    — Entrada A.
7         b      : in  std_logic_vector (3 downto 0) := "0000";    — Entrada B.
8         sel    : in  std_logic              := '0';              — Seletora de displays.
9         clk    : in  std_logic              := '0';              — Clock.
10        display1 : out std_logic_vector (6 downto 0);
11        display2 : out std_logic_vector (6 downto 0)
12    );
13 end projeto2;
14
15 architecture Behavioral of projeto2 is
16
17     signal saida_mux : std_logic_vector (3 downto 0);
18     signal bcd       : std_logic_vector (6 downto 0); — BCD.
19
20 begin
21
22     — Mux 8->4.
23     process (a,b, clk)

```

```

24 begin
25     if (clk = '0') then
26         saida_mux <= a;
27     else
28         saida_mux <= b;
29     end if;
30 end process;
31
32 — BCD.
33 process (a,b,clk , saida_mux , bcd)
34 begin
35     if (saida_mux = "0000") then — 0
36         bcd <= "1111110";
37     elsif (saida_mux = "0001") then — 1
38         bcd <= "0110000";
39     elsif (saida_mux = "0010") then — 2
40         bcd <= "1101101";
41     elsif (saida_mux = "0011") then — 3
42         bcd <= "1111001";
43     elsif (saida_mux = "0100") then — 4
44         bcd <= "0110010";
45     elsif (saida_mux = "0101") then — 5
46         bcd <= "1011010";
47     elsif (saida_mux = "0110") then — 6
48         bcd <= "1011111";
49     elsif (saida_mux = "0111") then — 7
50         bcd <= "1110000";
51     elsif (saida_mux = "1000") then — 8
52         bcd <= "1111111";
53     elsif (saida_mux = "1001") then — 9
54         bcd <= "1111011";
55     elsif (saida_mux = "1010") then — A
56         bcd <= "1110111";
57     elsif (saida_mux = "1011") then — B
58         bcd <= "0011111";
59     elsif (saida_mux = "1100") then — C
60         bcd <= "1001110";
61     elsif (saida_mux = "1101") then — D
62         bcd <= "0111101";
63     elsif (saida_mux = "1110") then — E
64         bcd <= "1001111";
65     else
66         bcd <= "1000111"; — Caso defaul -> 'F'
67     end if;
68 end process;
69

```

```

70
71 -- Mux 1->2.
72 process (bcd, clk, sel)
73 begin
74   if (clk = '0' and sel = '0') then    -- Se sel = 0 ento mostra B.
75     display2 <= bcd;                    -- Mostra B no display.
76     display1 <= "00000000";            -- Desliga A.
77   elsif (clk = '1' and sel = '1') then -- Se sel = 1 ento mostra A.
78     display1 <= bcd;                    -- Mostra A no display.
79     display2 <= "00000000";            -- Desliga B.
80   else
81     -- Caso inesperado.
82     display1 <= "00000000";            -- Desliga A.
83     display2 <= "00000000";            -- Desliga B.
84   end if;
85 end process;
86
87 end Behavioral;

```