



Laboratório Digital I - PCS3635

Relatório da Experiência 1:

Primeiro Contato com Circuitos Digitais em VHDL

Marco Aurélio C. O. Prado - NUSP 11257605

Victor Hoefling Padula - NUSP 10770051

Turma 04 - Bancada A1

São Paulo - SP

2021

1) Estudo da descrição VHDL fornecida

```
-----
-- Arquivo   : contador_163.vhd
-- Projeto   : Experiencia 01 - Primeiro Contato com VHDL
-----

-- Descricao : contador binario hexadecimal (modulo 16)
--            similar ao CI 74163
-----

-- Revisoes  :
--      Data      Versao Autor      Descricao
--      29/12/2020 1.0   Edson Midorikawa criacao
-----

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

entity contador_163 is -- entidade principal
  port (
    clock : in std_logic; -- sinais de entrada
    clr   : in std_logic;
    ld    : in std_logic;
    ent   : in std_logic;
    enp   : in std_logic;
    D     : in std_logic_vector (3 downto 0);
    Q     : out std_logic_vector (3 downto 0); -- sinais de saída
    rco   : out std_logic
  );
end contador_163;

architecture comportamental of contador_163 is -- declaração da arquitetura
  signal IQ: integer range 0 to 15;
begin

  process (clock,ent,IQ) -- inicio do process do circuito
  begin

    if clock'event and clock='1' then
      -- as mudanças no circuito ocorrem com o clock em 1
    end if;
  end process;
end;
```

```

if clr='0' then  IQ <= 0;
  -- caso o sinal clear seja 0, a contagem é reiniciada
elsif ld='0' then IQ <= to_integer(unsigned(D));
  -- caso o sinal load seja 0, a entrada D é carregada
elsif ent='1' and enp='1' then
  -- ambos os sinais de controle precisam estar em 1
  -- para que a contagem seja realizada
    if IQ=15 then  IQ <= 0;
      -- caso chegue no final da contagem, volta p/ 0
    else          IQ <= IQ + 1;
      -- caso contrário, soma-se 1 no contador
    end if;
  else          IQ <= IQ;
    -- caso um dos dois sinais de controle não esteja em nível
    -- lógico alto, o contador permanece em seu estado atual
  end if;
end if;

if IQ=15 and ent='1' then rco <= '1';
  -- caso o contador tenha chegado no final, rco assume valor 1
else          rco <= '0';
end if;

Q <= std_logic_vector(to_unsigned(IQ, Q'length));
  -- a saída Q recebe o valor do sinal utilizado para a contagem

end process; -- fim do process
end comportamental; -- fim da arquitetura

```

1.1) Perguntas sobre o componente

1.1.1) A saída Q deve variar de 0 a 15. Quais linhas de código VHDL confirmam este intervalo de valores?

As linhas onde os sinais Q, D e IQ são declaradas. Como Q e D são sinais de 4 bits, só podem representar números de 0000 (0) até 1111 (15). De modo semelhante, IQ, o sinal utilizado para a contagem, é definido com um integer de 0 a 15.

1.1.2) O sinal de CLEAR é síncrono e ativo em baixo. Quais linhas de código VHDL confirmam esta característica?

As seguintes linhas:

```
“if clock'event and clock='1' then  
    if clr='0' then  IQ <= 0;”
```

Aqui, vemos que o sinal IQ só é zerado caso o clock esteja em 1 (síncrono) e caso o sinal clr esteja em 0 (ativo em baixo).

1.1.3) Este componente é sensível à borda de subida do sinal de clock. Quais linhas de código VHDL confirmam esta característica?

A linha “if clock'event and clock='1' then”.

Todo o código que descreve o funcionamento do circuito está dentro deste laço, portanto, o componente é sensível à borda de subida do clock.

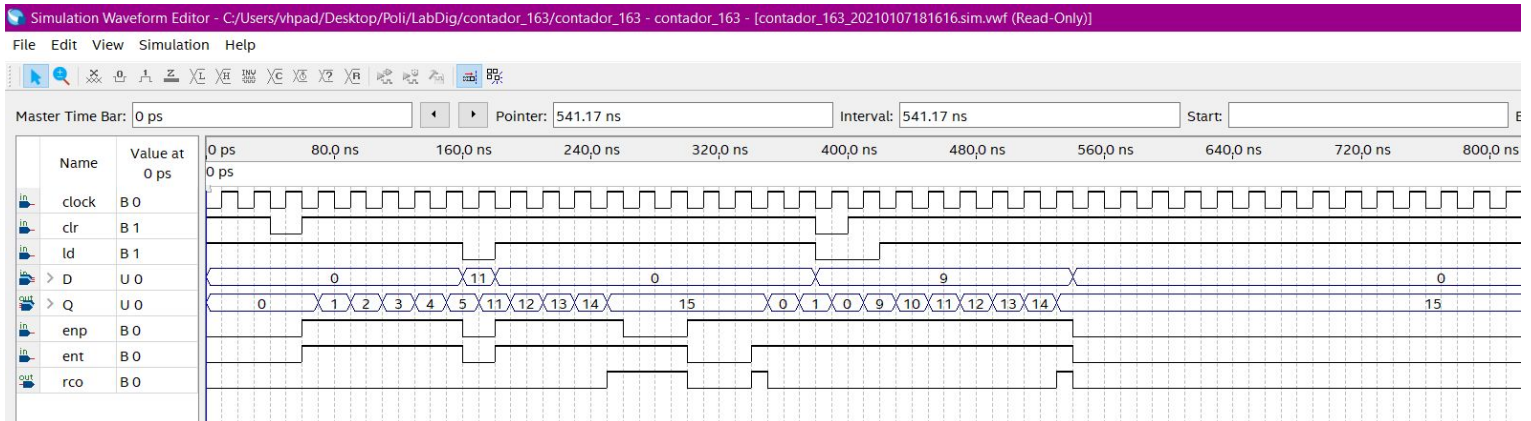
1.1.4) Se os sinais de CLEAR e de CARGA forem acionados simultaneamente, qual função é executada pelo componente?

A função executada é de zerar todos os valores, pois na arquitetura a descrição do funcionamento do load é posta como um “else if” do clear, tendo assim uma prioridade menor.

1.1.5) Os sinais ENT e RCO devem ser usados para cascadeamento de contadores. Quais linhas de código VHDL confirmam esta característica?

As linhas “elsif ent='1' and enp='1' then” e “if IQ=15 and ent='1' then rco <= '1';”. Com esses sinais operando dessa forma é possível ligar a saída rco de um contador na entrada ent de outro, expandindo o valor a ser contado. (Ex: quando o primeiro contador contar até 15 ele aciona a saída rco, que ligada a entrada ent de um segundo contador permite que este comece a contar até 15, aumentando a contagem total até 30.)

2) Simulação de Circuito em VHDL



2.1) Análise dos resultados obtidos

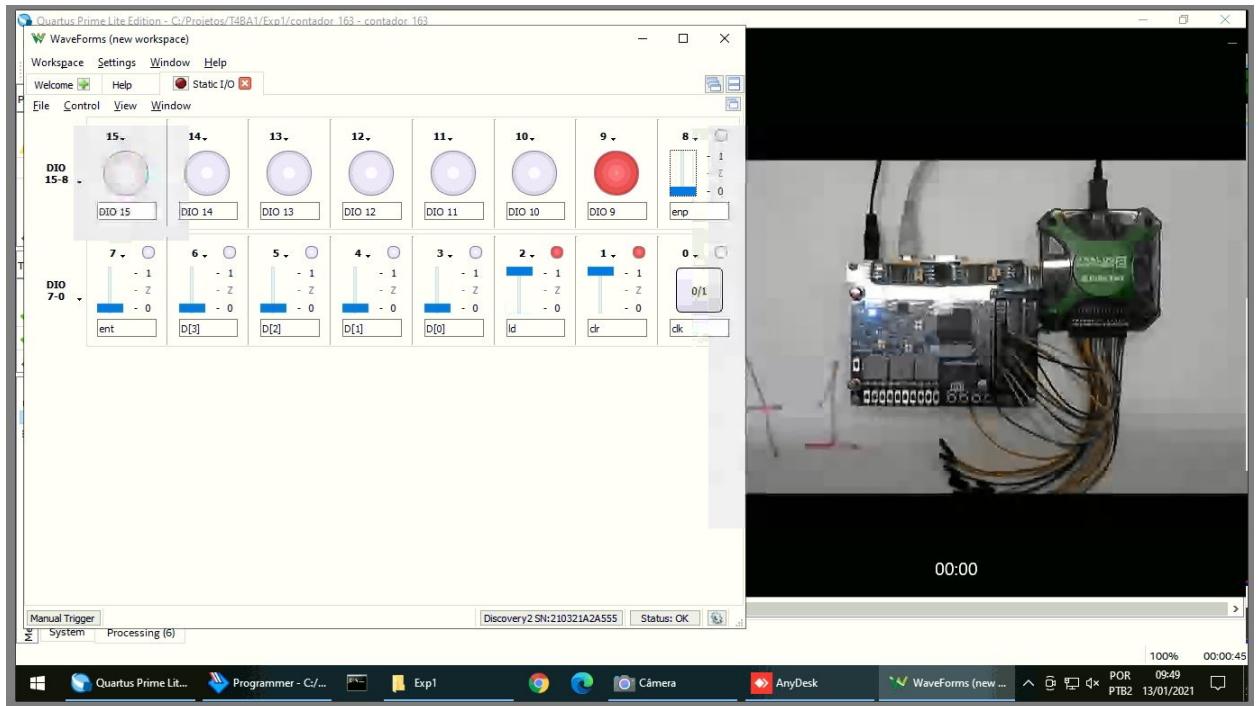
A simulação foi um sucesso, as saídas correspondem com as entradas dentro do esperado.

2.2) Resumo do funcionamento do circuito

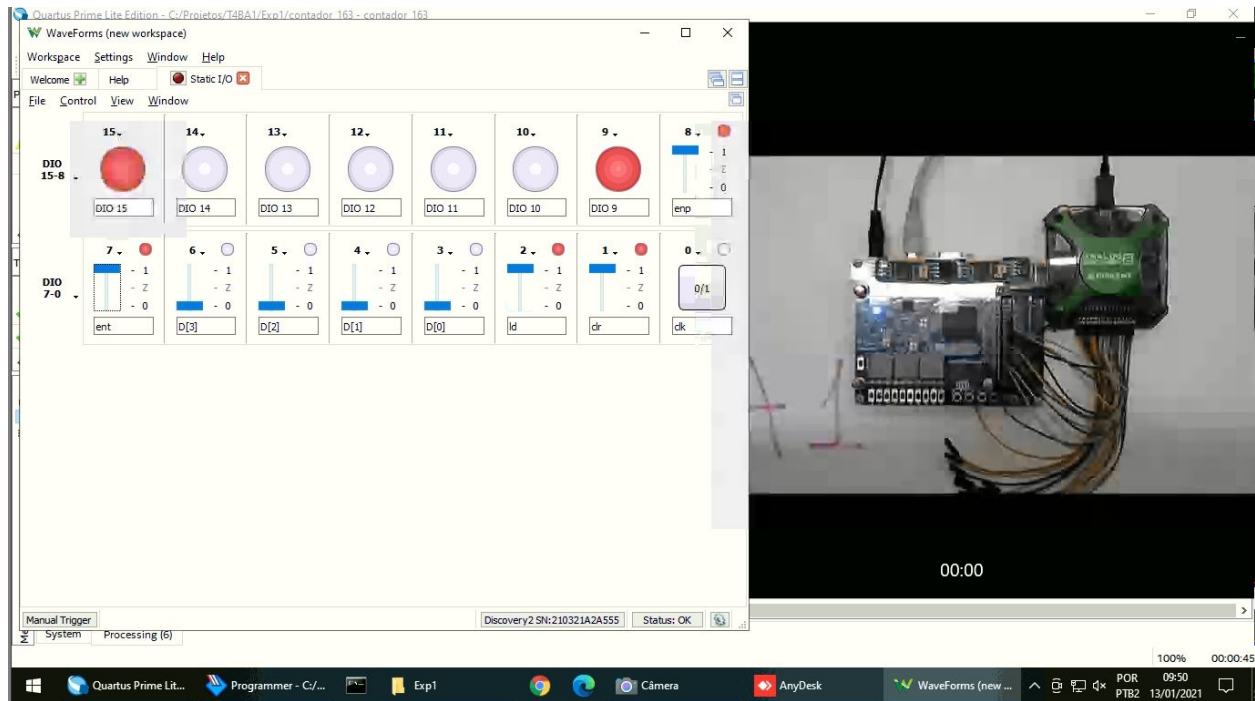
Com as entradas de controle ent e enp habilitadas, o contador conta os períodos de clock até 15. O sinal clear zera o contador, e o sinal load carrega um valor específico de 0 a 15 com que o contador pode começar. Quando ele chega em 15, a saída rco entra em nível lógico alto mostrando que a contagem terminou.

3) Demonstração da Experiência

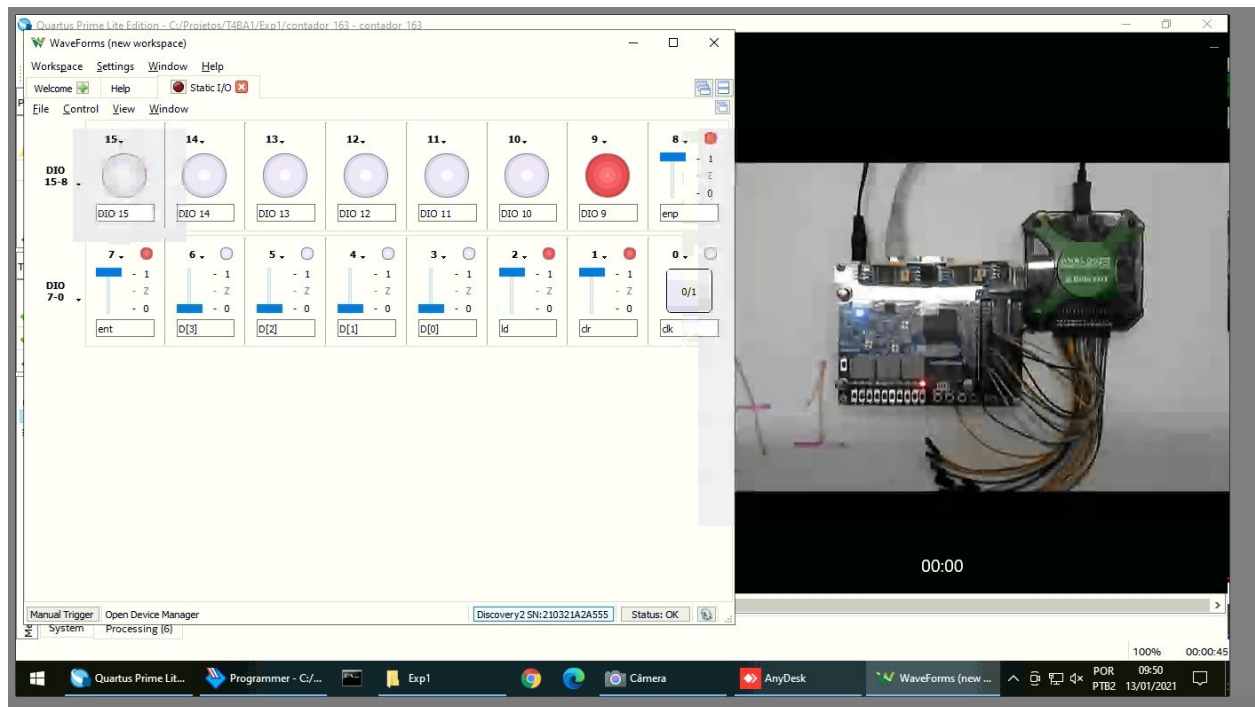
Nas fotos, a saída RCO está mapeada no LED mais à esquerda e a saída Q nos 4 LED's mais à direita



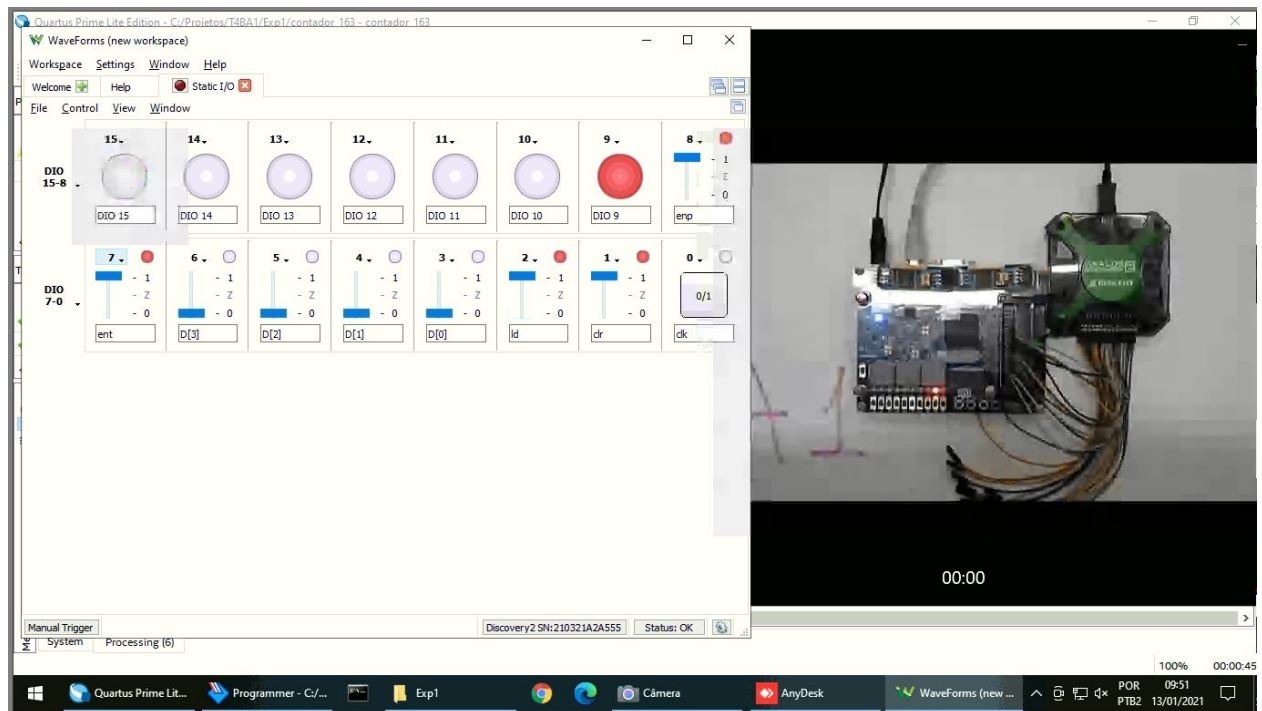
Condição inicial: clock=0, clear=1, ld=1, D=0000, ent=0, enp=0, Q=0000, rco=0



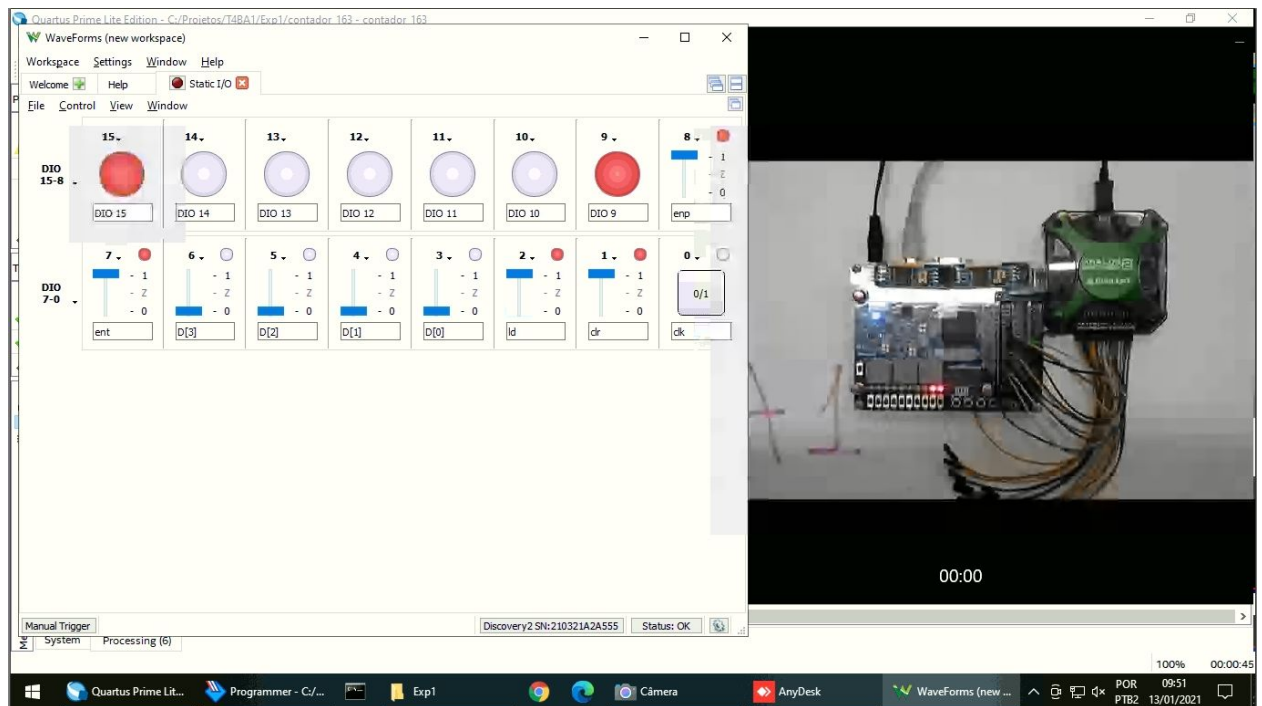
Caso 1: Uma subida de clock, clear=1, ld=1, D=0000, ent=1, enp=1, Q=0000, rco=0



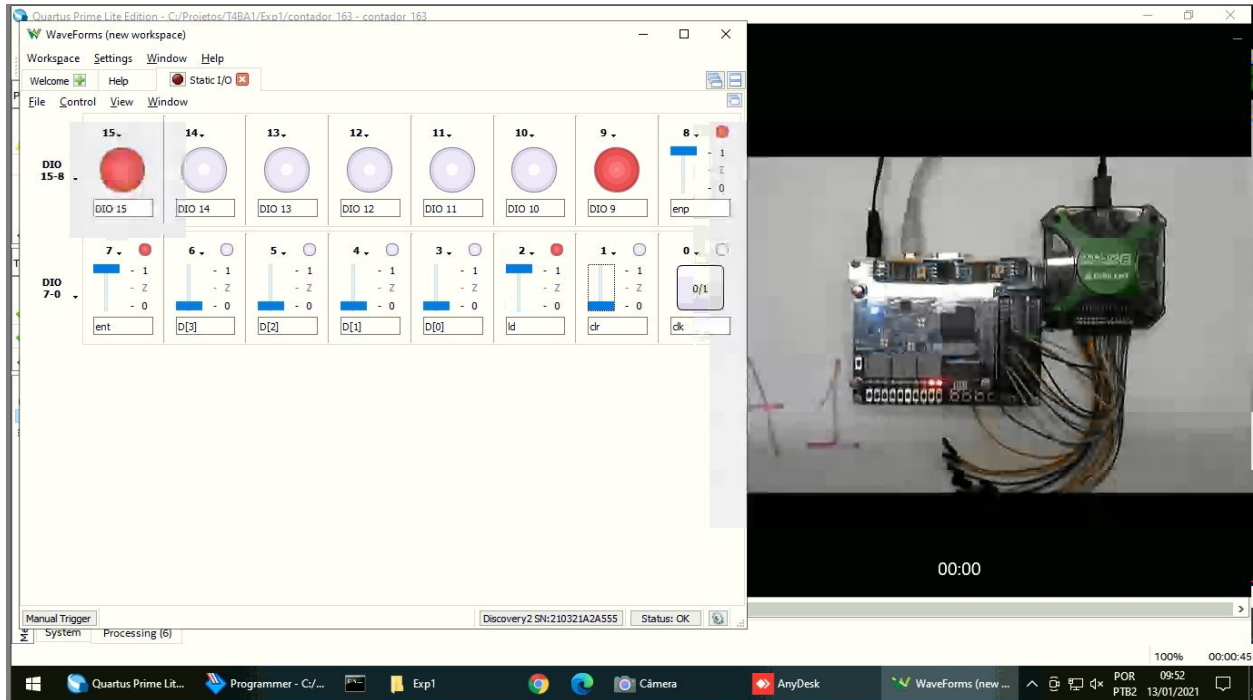
Caso 2: Uma subida de clock, clear=1, ld=1, D=0000, ent=1, enp=1, Q=0001, rco=0



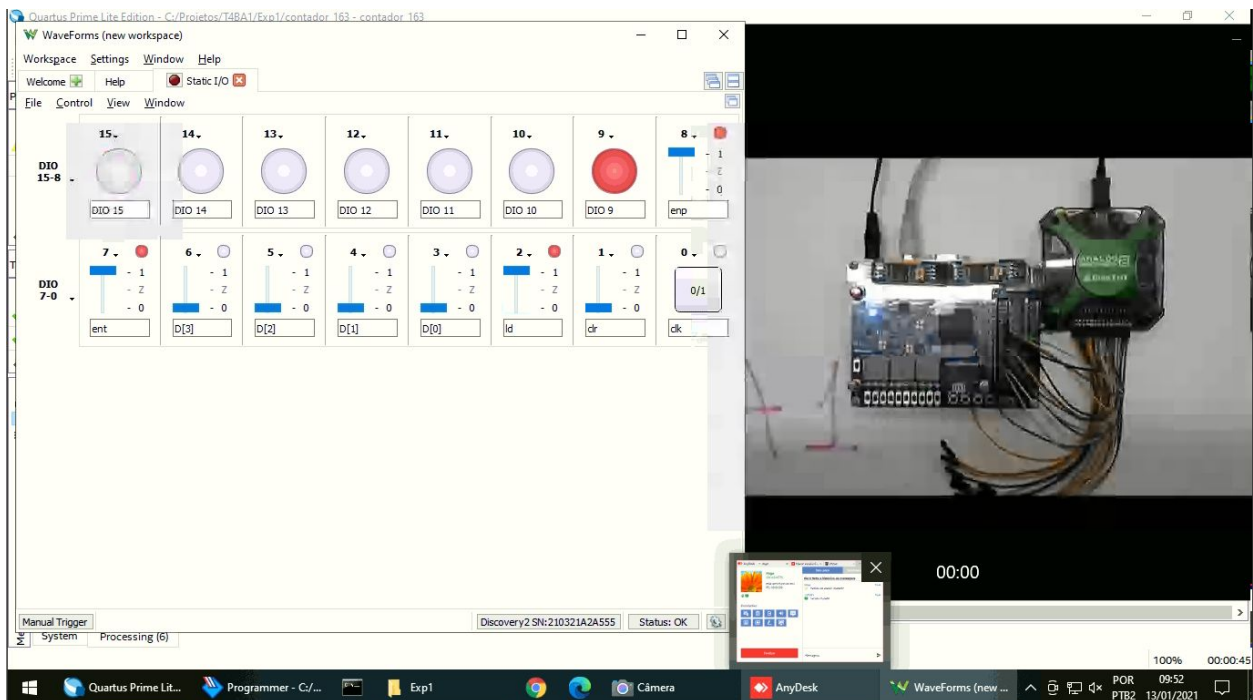
Caso 3: Uma subida de clock, clear=1, ld=1, D=0000, ent=1, enp=1, Q=0010, rco=0



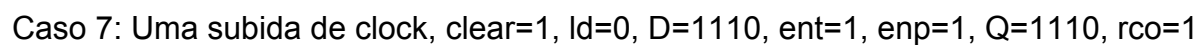
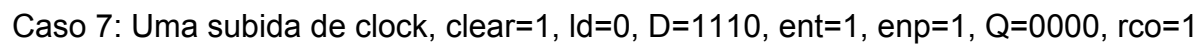
Caso 4: Uma subida de clock, clear=1, ld=1, D=0000, ent=1, enp=1, Q=0011, rco=0

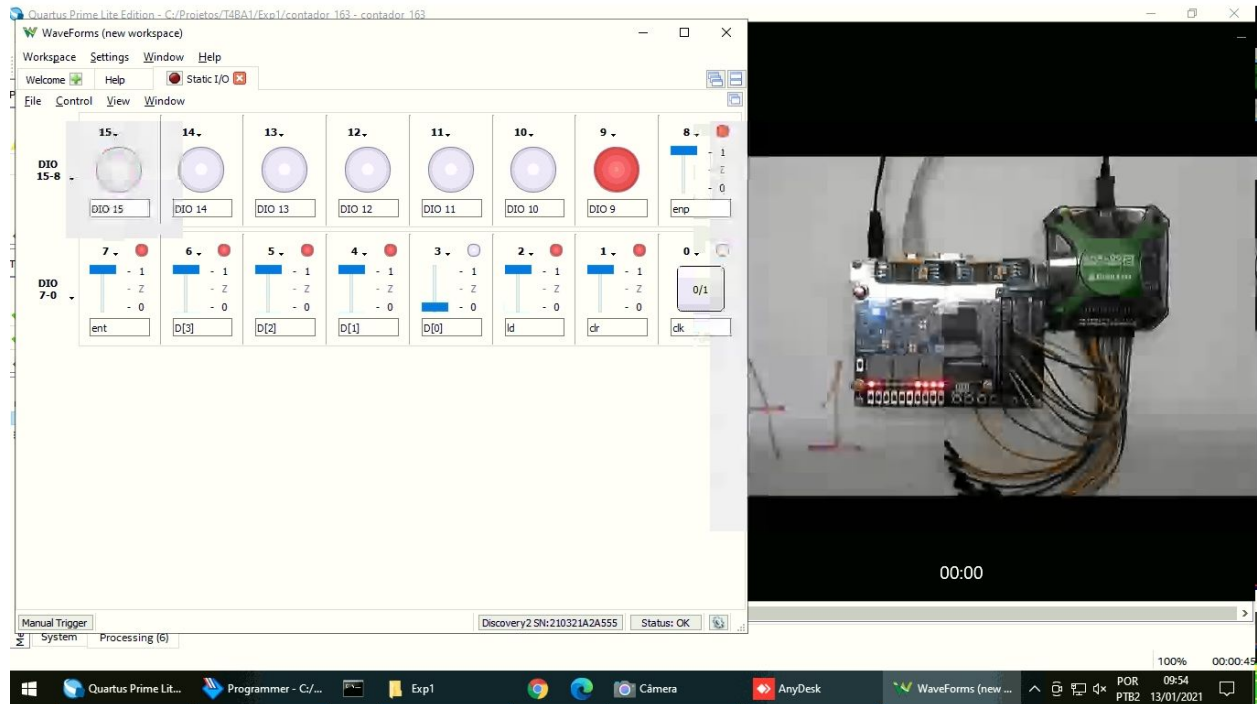


Caso 5: Uma subida de clock, clear=0, ld=1, D=0000, ent=1, enp=1, Q=0011, rco=0



Caso 6: Uma subida de clock, clear=0, ld=1, D=0000, ent=1, enp=1, Q=0000, rco=0





Caso 8: Uma subida de clock, clear=1, ld=1, D=1110, ent=1, enp=1, Q=1111, rco=1

4) Resultados Alcançados

4.1) Pontos Positivos

Os pontos positivos consistiram do bom funcionamento da ferramenta AnyDesk e Analog Discovery, que funcionaram bem para replicar a experiência do laboratório para o ensino à distância. A placa FPGA e o software do Intel Quartus também funcionaram corretamente, junto com o código VHDL fornecido para a experiência.

4.2) Pontos Negativos

Os pontos negativos consistiram de problemas com o sistema de arquivos do Intel Quartus. Quando fomos realizar o experimento, nomeamos a pasta do projeto como “Experiência 1”, o que causou problemas na hora da geração do waveform do plano de testes, pois provavelmente o Quartus não lidou bem com o acento e/ou o espaço no nome. Criamos outro projeto numa pasta com o nome “Exp1” e tudo correu bem.

4.3) Lições Aprendidas

As lições aprendidas consistiram na percepção das diferenças de um trabalho teórico para um trabalho prático. Além de podermos ver o que aprendemos em Sistemas Digitais na prática, indo do VHDL até os LED's piscando, podemos perceber que num projeto prático de engenharia os erros estão muito mais

prováveis de acontecer, o que gera a necessidade de um bom planejamento e de aprendizado com os erros.

5) Referências Técnicas

1. Apostila e apresentação da Experiência 1, disponibilizados no ambiente da disciplina PCS3635 no site E-disciplinas.
2. Datasheet do circuito integrado 7485.
3. Material sobre 7485 do livro Sistemas Digitais (Tocci).