Relatório

Link do GitHub:
https://github.com/almeidarodrigo04/Pratica-em-Sistemas-Digitais-Atividade-1

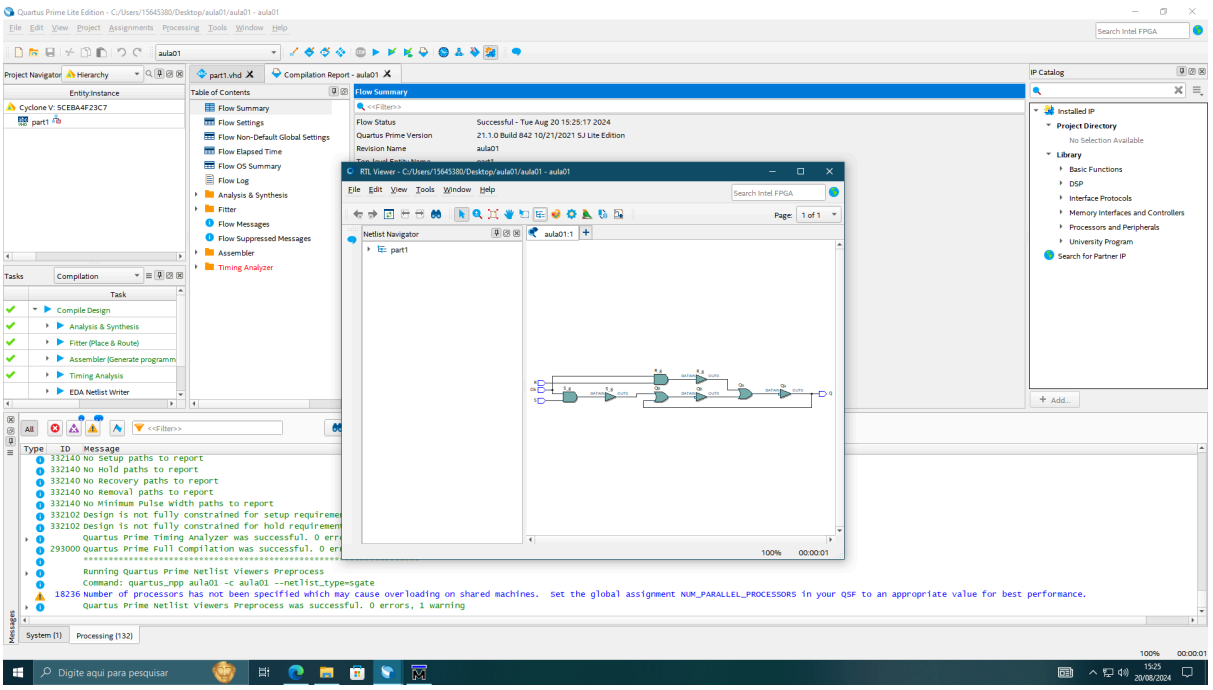
Felipe Felipe Camargo Cerri - 15451119
Rodrigo Silva de Almeida - 15645380

27 de Agosto de 2024

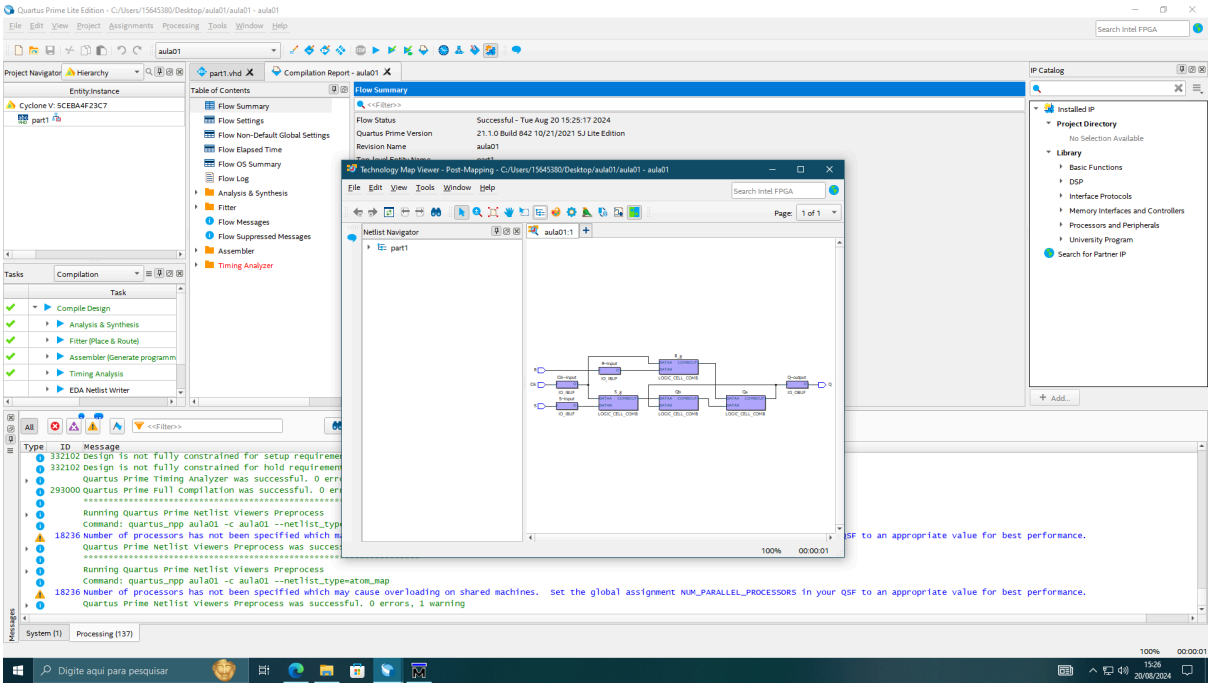
Parte I

Nessa parte, simulamos o código em VHDL fornecido pelo PDF da atividade.

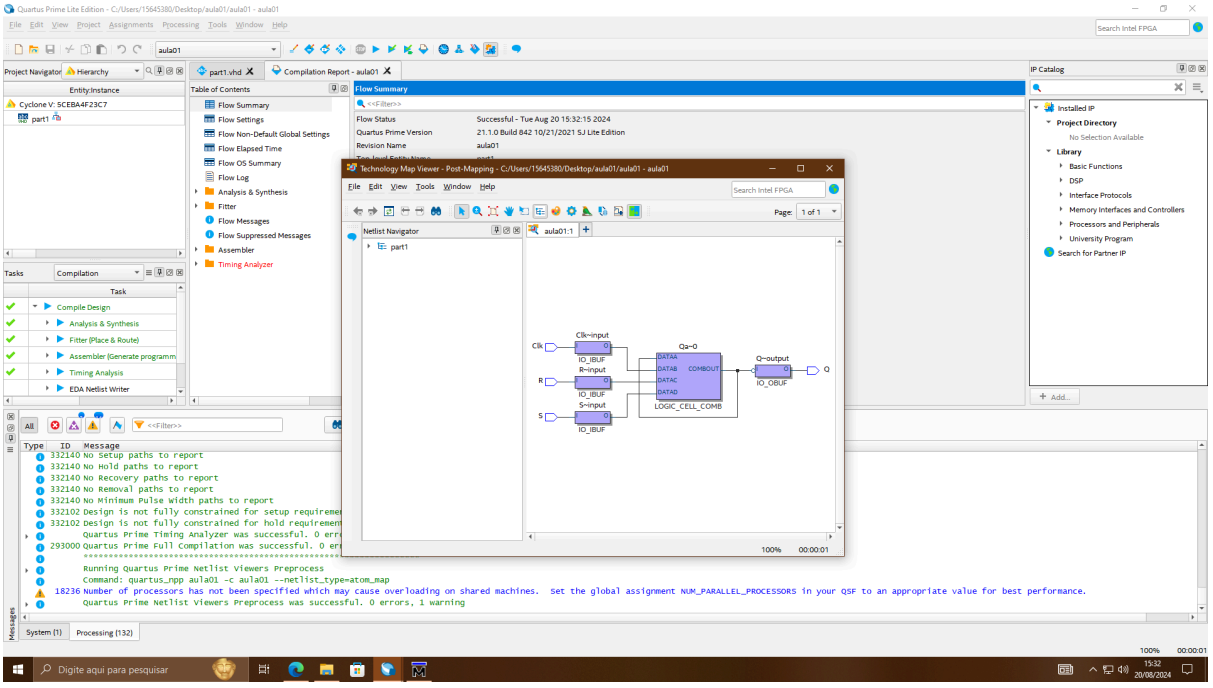
1- RTL viewer



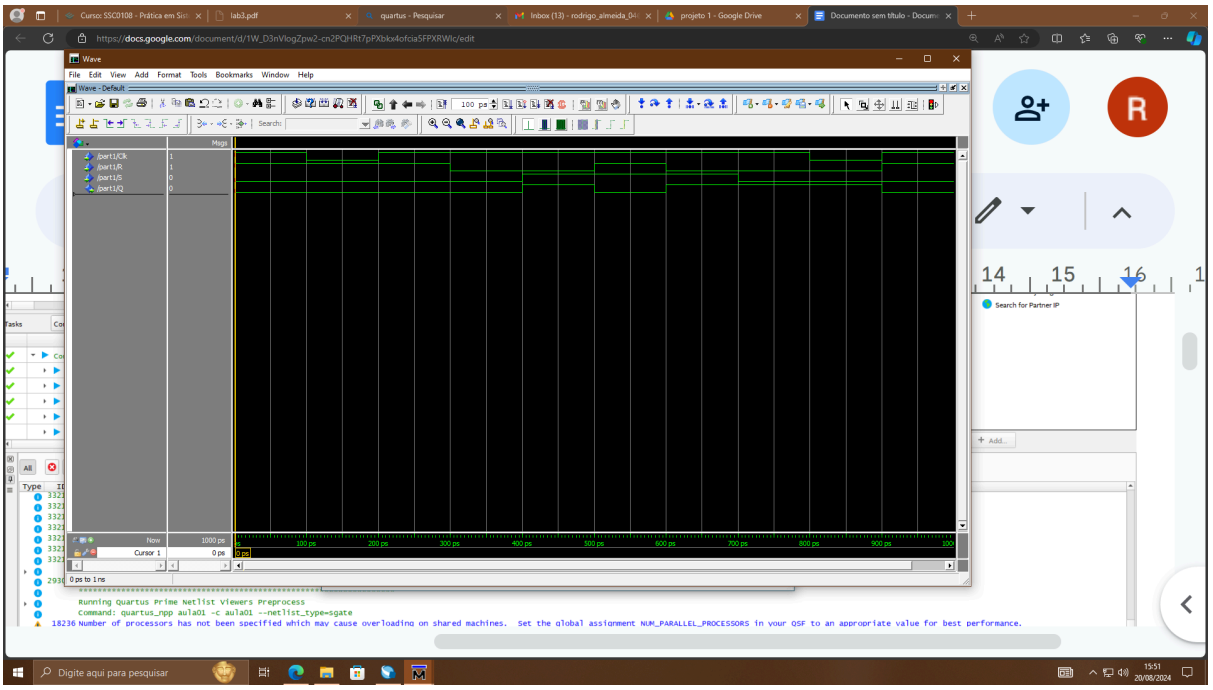
2 - Technology Map Viewer - várias LUT's



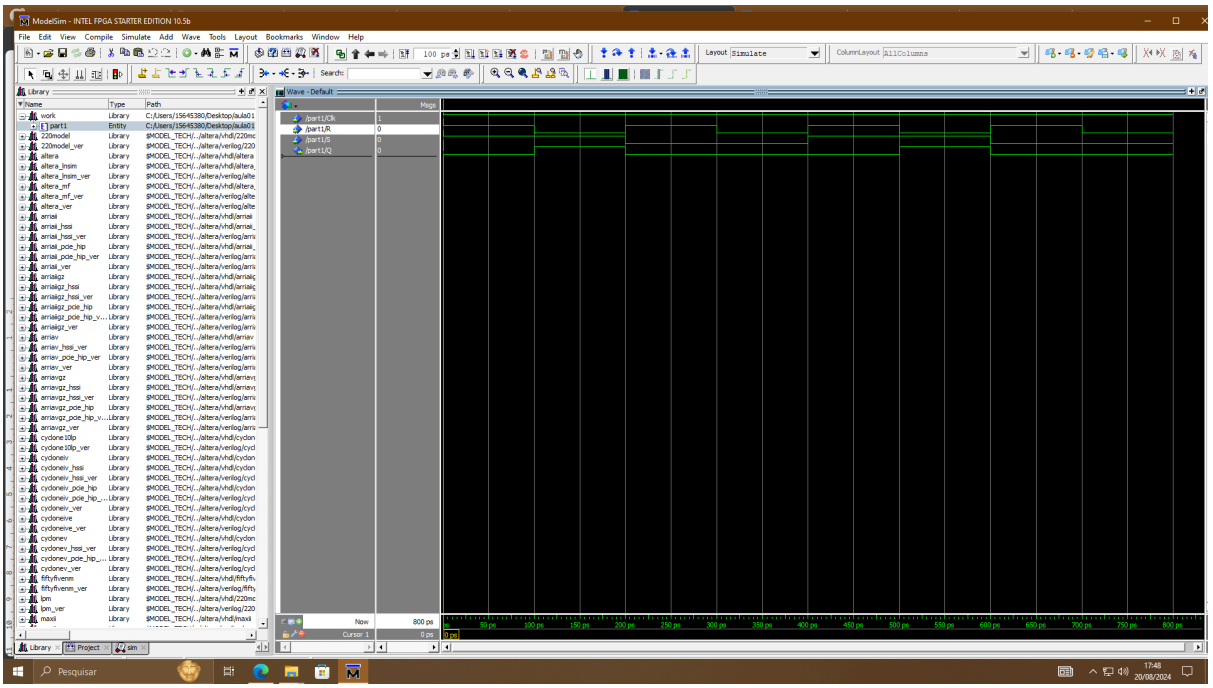
3 - Technology Map Viewer - sem os "keep's" - só uma LUT



4 - Simulação de onda, mostrando a natureza de armazenamento de memória do circuito



5 - Outra foto do simulador de onda

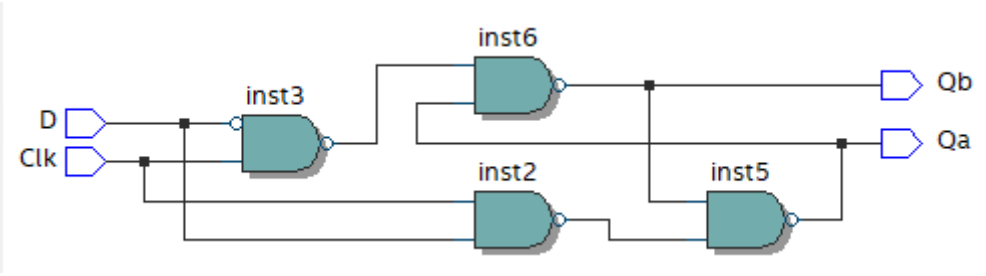


Parte II

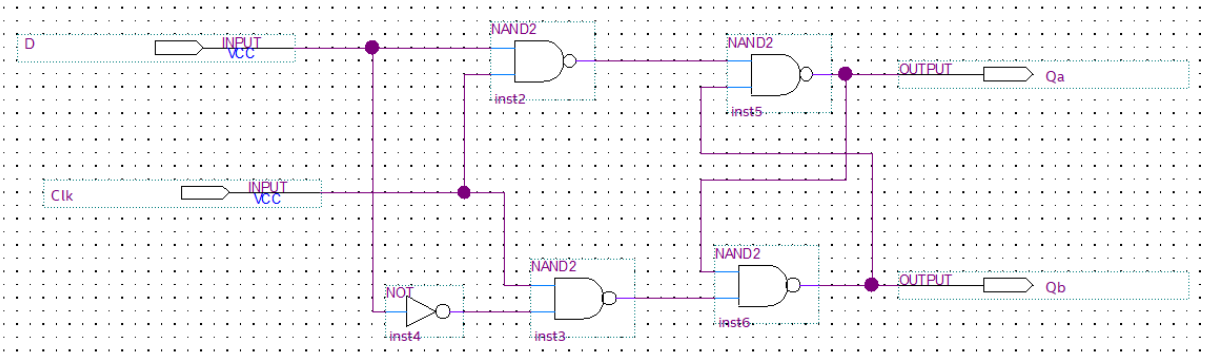
Gated D Latch

- D muda o valor guardado pela memória apenas quando o clock está em 1, ou seja, o clock é sensível ao nível

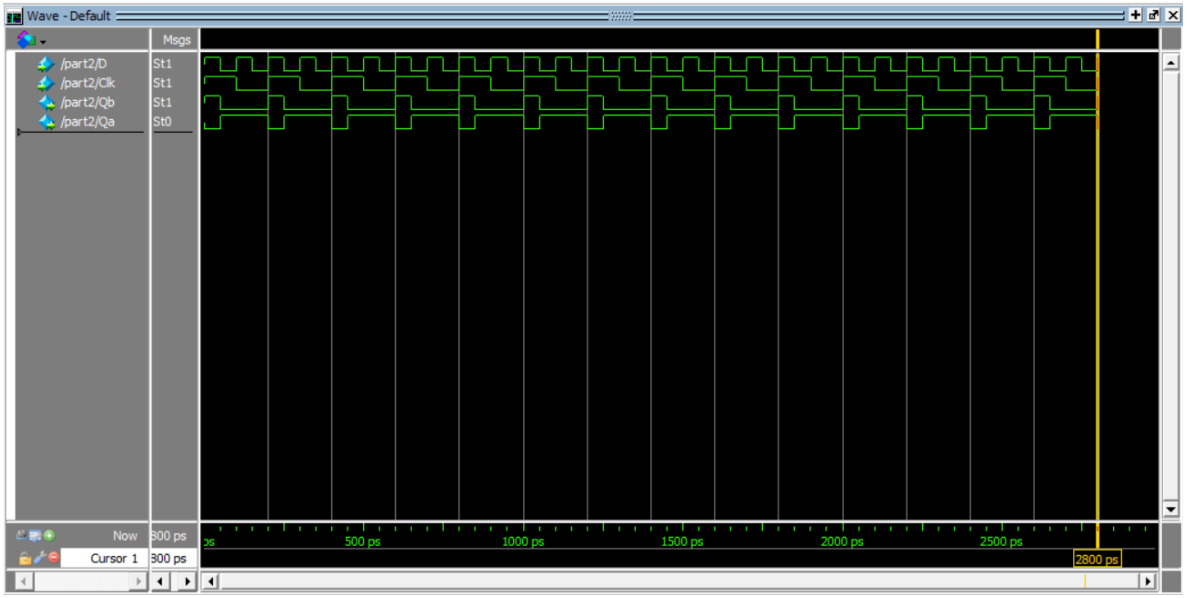
1 - RTL viewer



2 - Foto do esquemático



3 - Simulação de ondas da parte 2

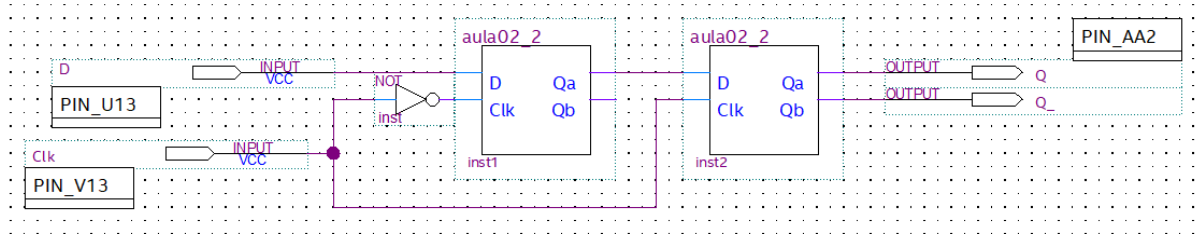


Parte III

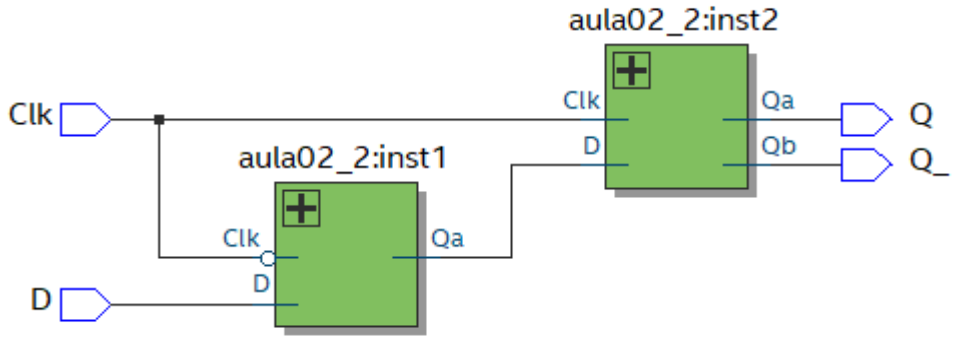
Master-Slave D flip-flop

- D altera a memória apenas quando o clk vai de 0 para 1, ou seja, o clock é sensível a borda (nesse caso positiva, positive edge-triggered)

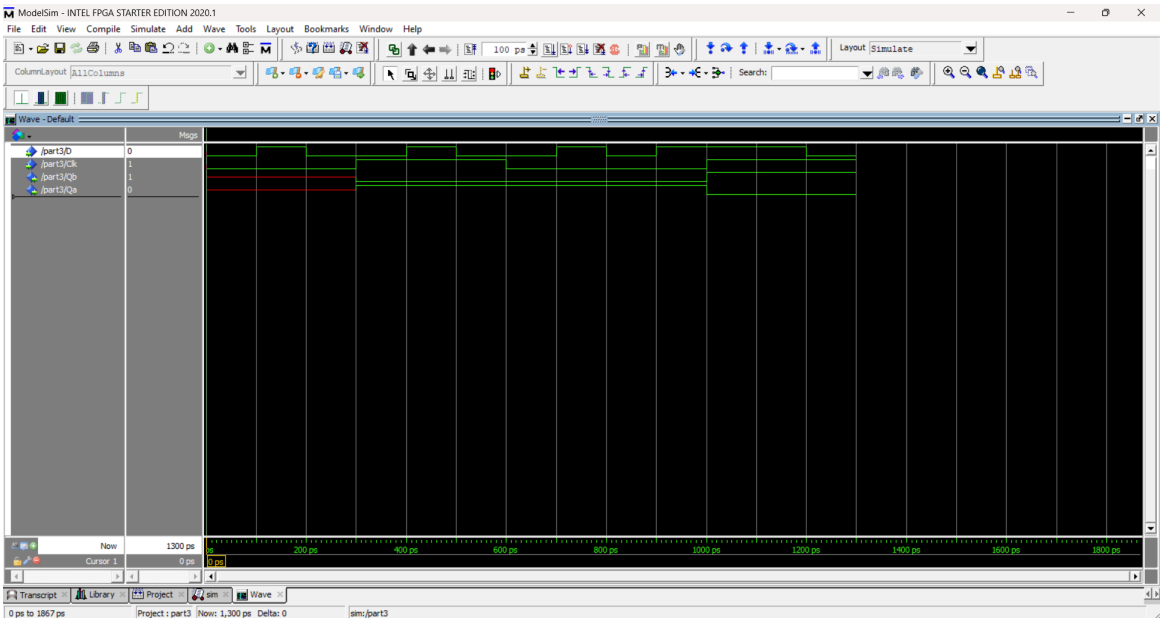
1 - Esquemático do projeto



2 - Technology Map Viewer



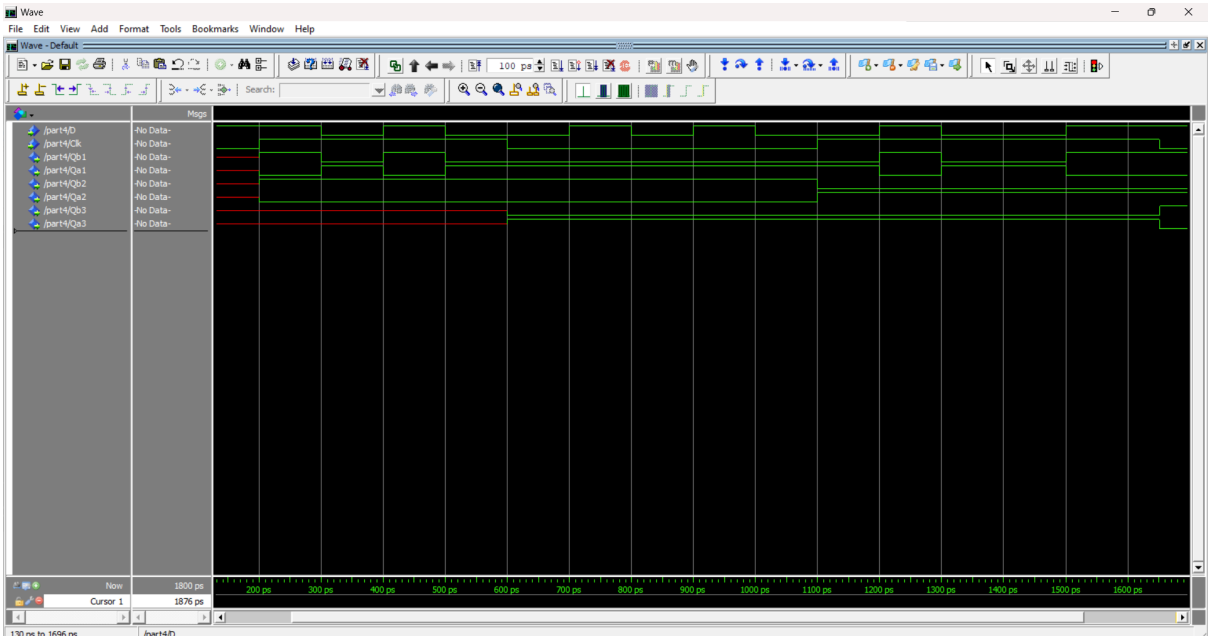
3 - Simulação de ondas da parte 3



Parte IV

Implementação de um D-Latch, um positive-edge triggered D flip-flop e um negative-edge triggered D flip-flop, todos compartilhando os mesmos 2 inputs (D e Clk)

1 - Simulação de onda

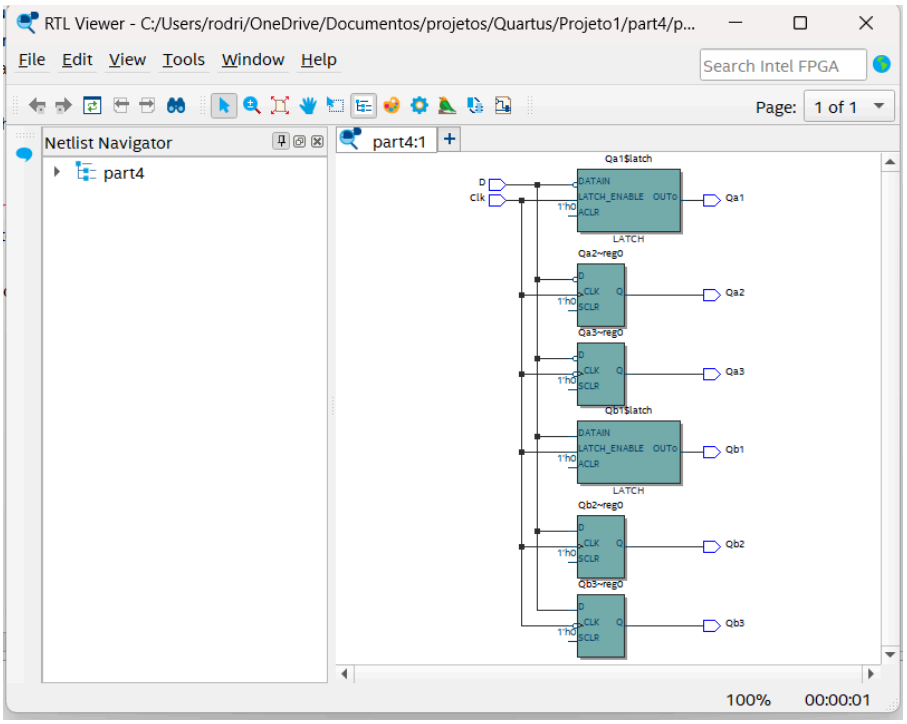


2 - Código em VHDL

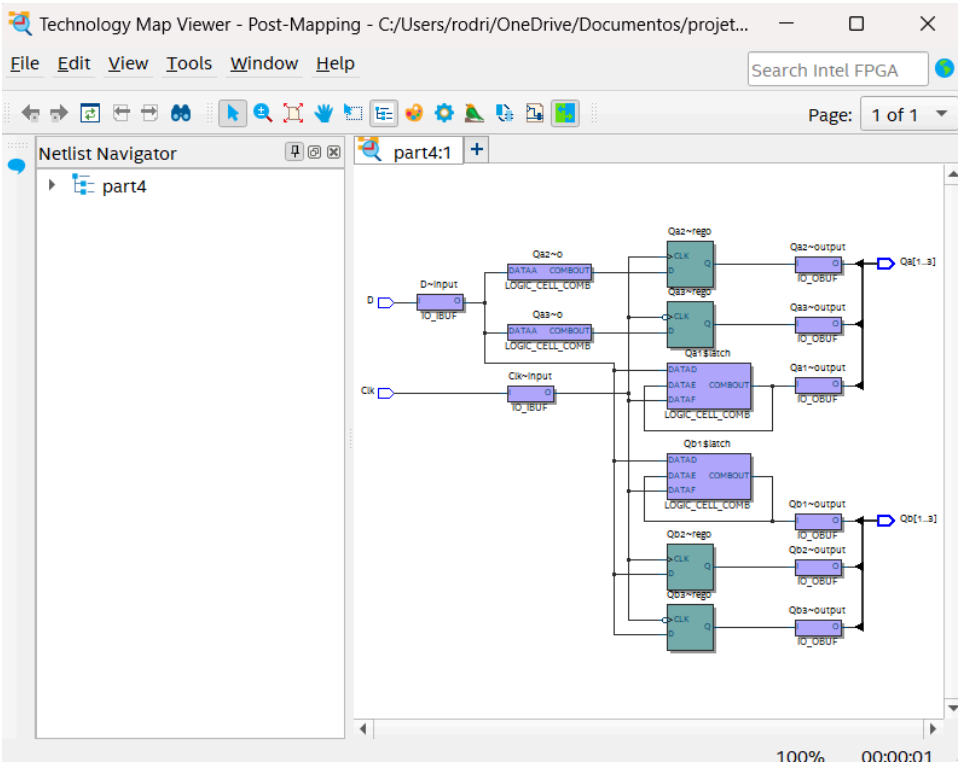
part4.vhd

```vhdl
LIBRARY IEEE;
use IEEE.std_logic_1164.all;

entity part4 is
    port (
        D: in std_logic;
        Clk: in std_logic;
        Qb1: out std_logic;
        Qa1: out std_logic;
        Qb2: out std_logic;
        Qa2: out std_logic;
        Qb3: out std_logic;
        Qa3: out std_logic
    );
end entity part4;

architecture Behaviour of part4 is
begin
    process(D, Clk)
    begin
        if Clk = '1' then
            Qb1 <= D;
            Qa1 <= not D;
        end if;
        if rising_edge(Clk) then
            Qb2 <= D;
            Qa2 <= not D;
        end if;
        if falling_edge(Clk) then
            Qb3 <= D;
            Qa3 <= not D;
        end if;
    end process;
end architecture;
```

3 - RTL viewer



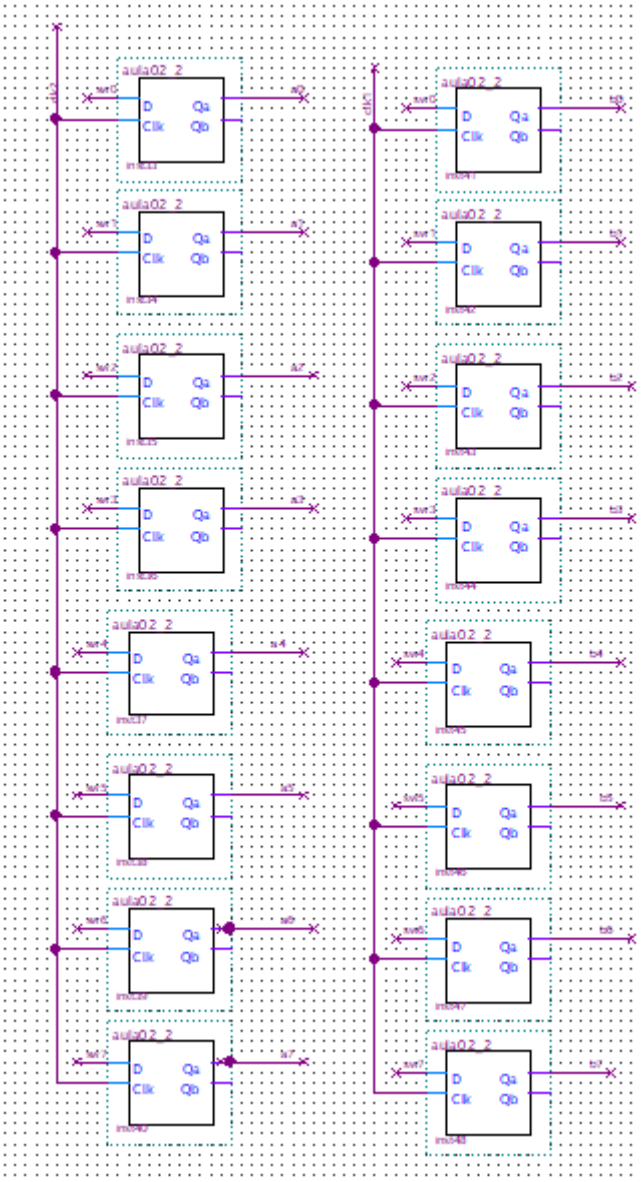
4 - Technology Map Viewer



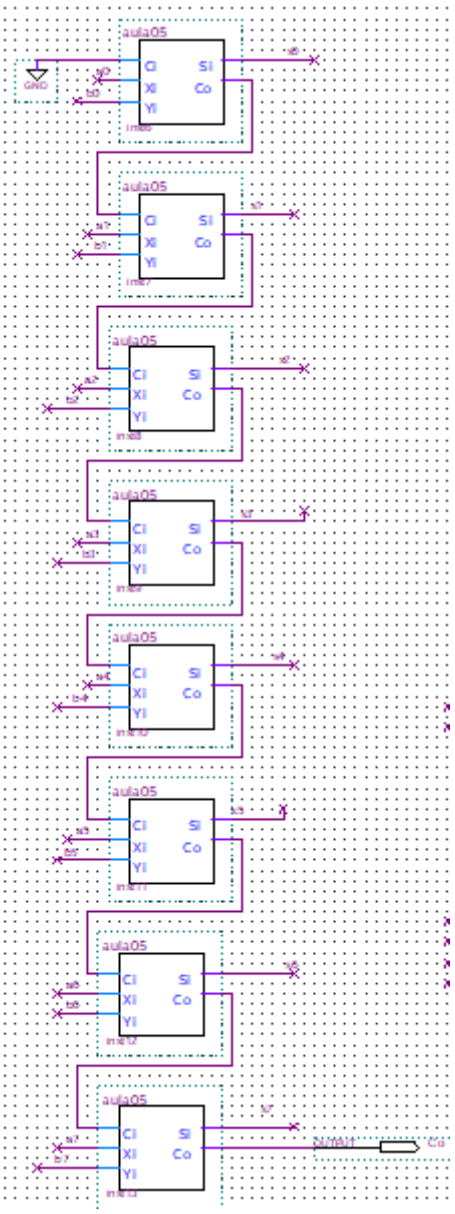
Parte V

Implementação de um somador de palavras de 8 bits com uso de flip-flop'

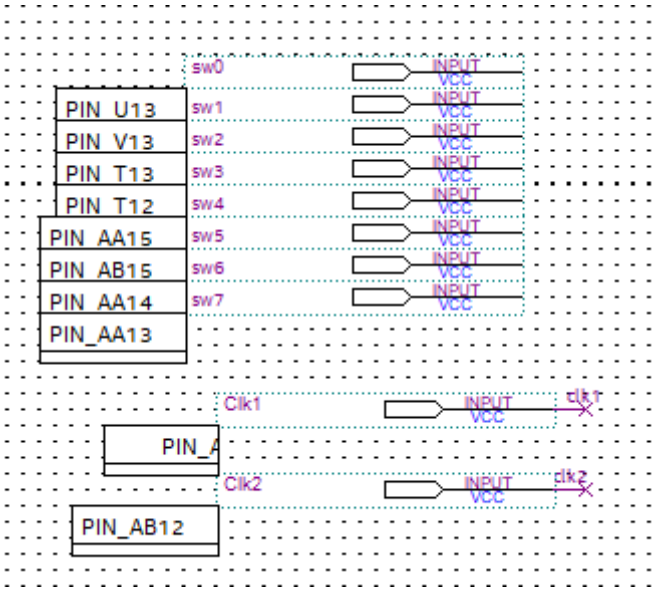
1 - Esquemático dos flip-flops usados (guarda os valores de A e B)



2 - Adder de palavras de 8 bits



3 - Pinos utilizados na FPGA



4 - Esquemático do display