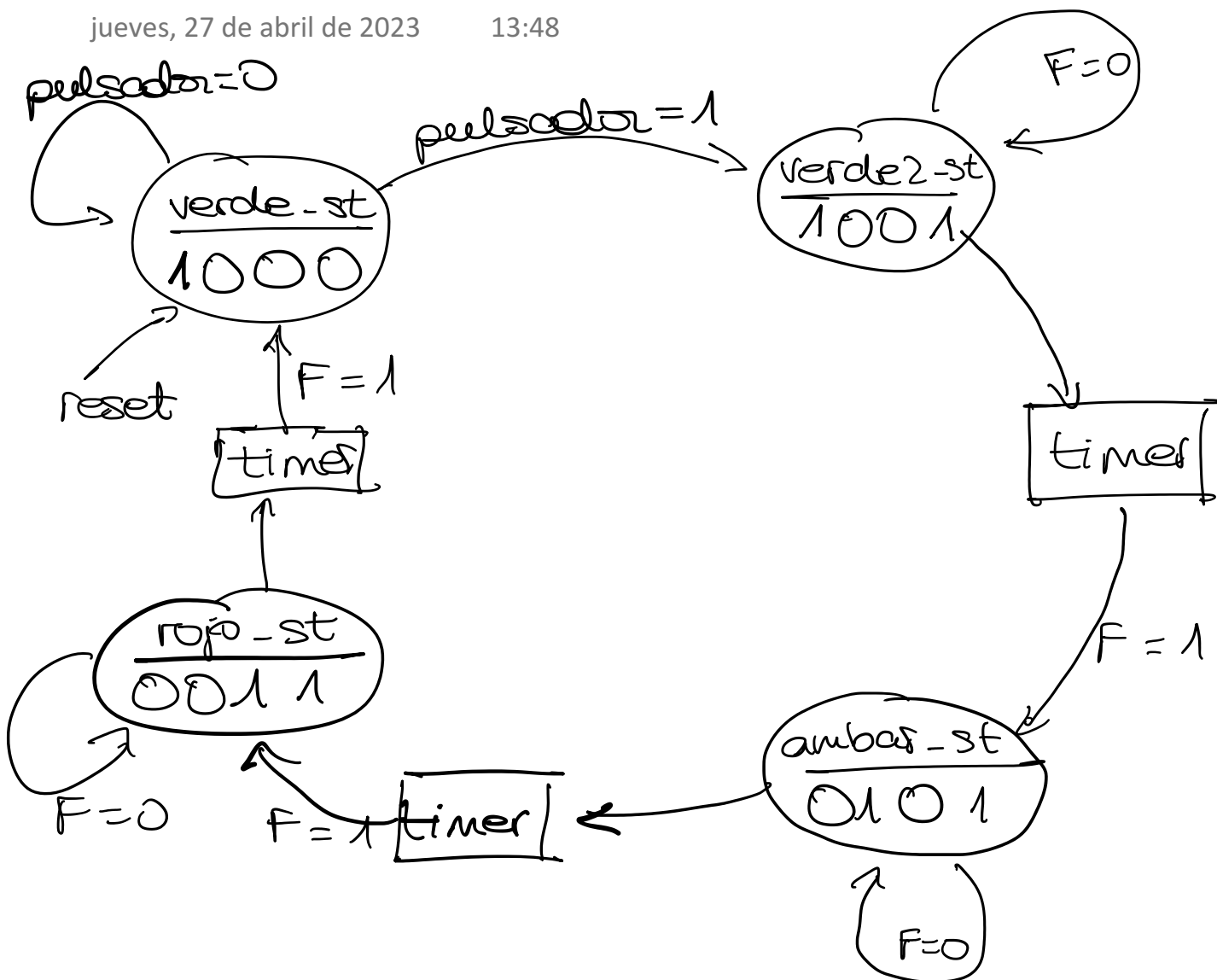
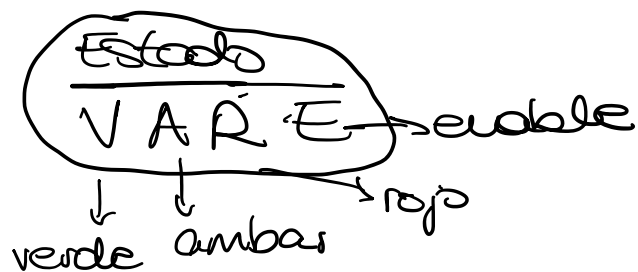


Práctica4

jueves, 27 de abril de 2023 13:48



Legenda:



Previo:

$$1) f = 1\text{Hz} \rightarrow T = \frac{1}{f} = 1\text{s/ciclo}$$

Como te pide un temporizador de 5s → realizará 5 ciclos.

Grupo 9: Miguel Fidalgo García
Gonzalo Diet Villan

C:\altera\91sp2\quartus\P4\p4.vhd

```
library ieee;
use ieee.std_logic_1164.all;
use ieee.numeric_std.all;

entity p4 is
    port(clk, reset: in std_logic;
          pulsador: in std_logic;
          rojo, verde, ambar: out std_logic
    );
end p4;

architecture a of p4 is
    type state is (verde_st, verde2_st, ambar_st, rojo_st);
    signal current_st, next_st: state;
    signal count: unsigned(2 downto 0);
    signal F: std_logic;
    signal en: std_logic; -- Enable (cuando valga 1 uso el temporizador)
    signal q0, q1, pulsador_f: std_logic;

begin
    --detector de flancos
    process(clk, reset)
    begin
        if reset = '0' then
            current_st <= verde_st; -- cuando ponga el reset vuelvo al estado inicial
        elsif clk'event and clk = '1' then
            current_st <= next_st;
        end if;
    end process;

    --proceso del pulsador en el detector de flancos
    process(clk, reset)
    begin
        if reset = '0' then
            q0 <= '0'; q1 <= '1';
        elsif clk'event and clk = '1' then
            q0 <= pulsador;
            q1 <= q0;
        end if;
    end process;

    pulsador_f <= q0 and not q1;

    --circuito combinacional
    process(current_st, pulsador, F)
    begin
        next_st <= current_st;
        case current_st is
            when verde_st =>
                if pulsador_f = '1' then
                    next_st <= verde2_st;
                end if;
                verde <= '1'; ambar <= '0'; rojo <= '0'; en <= '0';
            when verde2_st =>
                if F = '1' then
                    next_st <= ambar_st;
```

```

        end if;
        verde <= '1'; ambar <= '0'; rojo <= '0'; en <= '1';
    when ambar_st =>
        if F = '1' then
            next_st <= rojo_st;
        end if;
        verde <= '0'; ambar <= '1'; rojo <= '0'; en <= '1';
    when rojo_st =>
        if F = '1' then
            next_st <= verde_st;
        end if;
        verde <= '0'; ambar <= '0'; rojo <= '1'; en <= '1';

    end case;
end process;

--timer
process(clk, reset)
begin
    if reset = '0' then
        count <= "000";
    elsif clk'event and clk = '1' then
        if F = '1' then
            count <= "000";
            --F <= '0';
        end if;

        if en = '1' then
            count <= resize(count, 3) + "001";
            --count <= count + 1
        end if;
    end if;
end process;

-- cuando el contador llegue a 5 activo F(fin de cuenta) que indica
-- que ha alcanzado el valor máximo
--close <= '1' when count >= 9 else '0';
--when count = "101" => F <= '1';

F <= '1' when count = "101" else '0';
end a;
```