Documentation for Lab 4

Name: Nikolay Stoitsov

Group: 2

I did the lab alone with group 1.

Lab Task 1: Traffic light controller without duration

**VHDL code:**

library ieee;

use ieee.std\_logic\_1164.all;

use ieee.numeric\_std.all;

entity Traffic\_light is

port(RESET, CLK, DN : in bit; --MAIN(0) – main road cars RED

MAIN, SEC : out bit\_vector(4 downto 0)); -- MAIN(1) – m. r. cars AMBER

end entity Traffic\_light; -- MAIN(2) – m. r. cars GREEN

-- MAIN(3) – m. r. ped. RED

architecture CONTROLLER of Traffic\_light is -- MAIN(4) – m. r. ped. GREEN

signal Z : unsigned (3 downto 0);

--SEC(0) – s. r. cars RED

Begin --SEC(1) – s. r. cars AMBER

P1: process(CLK, RESET) --SEC(2) – s. r. cars GREEN

Begin --SEC(3) – s. r. ped. RED

if RESET = '0' then Z <= "0000" after 5 ns; --SEC(4) – s. r. ped. GREEN

elsif CLK = '1' and CLK' event then

if DN = '0' then

if Z = "1000" then Z <= "0001" after 5 ns;

elsif Z = "0001" then Z <= "1110" after 5 ns;

elsif Z = "1111" then Z <= "1110" after 5 ns;

else Z <= Z + 1 after 5 ns;

end if;

else

if Z = "0001" then Z <= "0010" after 5 ns;

elsif Z = "1000" then Z <= "0010" after 5 ns;

elsif Z = "1111" then Z <= "0000" after 5 ns;

else Z <= Z + 1 after 5 ns;

end if;

end if;

end if;

end process P1;

P2: process(Z)

begin

case Z is

when "0000" => MAIN <= "01010" after 5 ns; SEC <= "01010" after 5 ns;

when "0001" => MAIN <= "10010" after 5 ns; SEC <= "10010" after 5 ns;

when "0010" => MAIN <= "11001" after 5 ns; SEC <= "10010" after 5 ns;

when "0011" => MAIN <= "00101" after 5 ns; SEC <= "10010" after 5 ns;

when "0100" => MAIN <= "01010" after 5 ns; SEC <= "10010" after 5 ns;

when "0101" => MAIN <= "10010" after 5 ns; SEC <= "10010" after 5 ns;

when "0110" => MAIN <= "10010" after 5 ns; SEC <= "11001" after 5 ns;

when "0111" => MAIN <= "10010" after 5 ns; SEC <= "00101" after 5 ns;

when "1000" => MAIN <= "10010" after 5 ns; SEC <= "01010" after 5 ns;

when "1110" => MAIN <= "00000" after 5 ns; SEC <= "01000" after 5 ns;

when "1111" => MAIN <= "00000" after 5 ns; SEC <= "00000" after 5 ns;

when others => MAIN <= "00000" after 5 ns; SEC <= "00000" after 5 ns;

end case;

end process P2;

end CONTROLLER;

**Screenshots:**

1) CLK; 2) RED; 3) AMBER; 4) GREEN

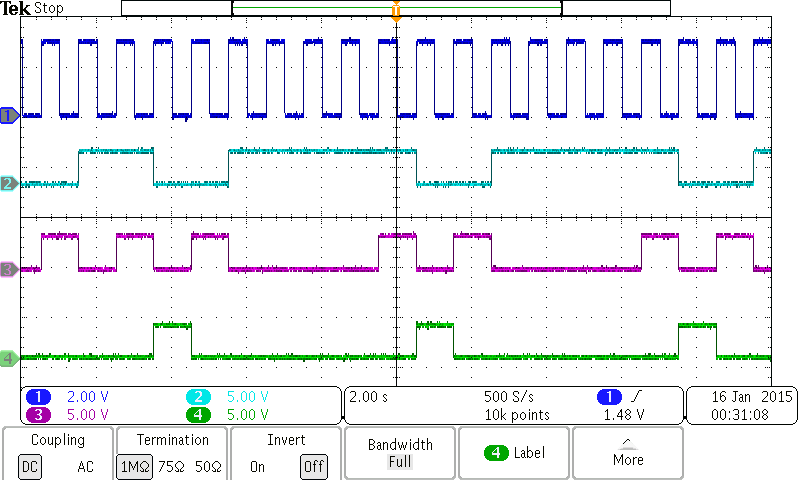


Fig1: Control signals for secondary road day cycle

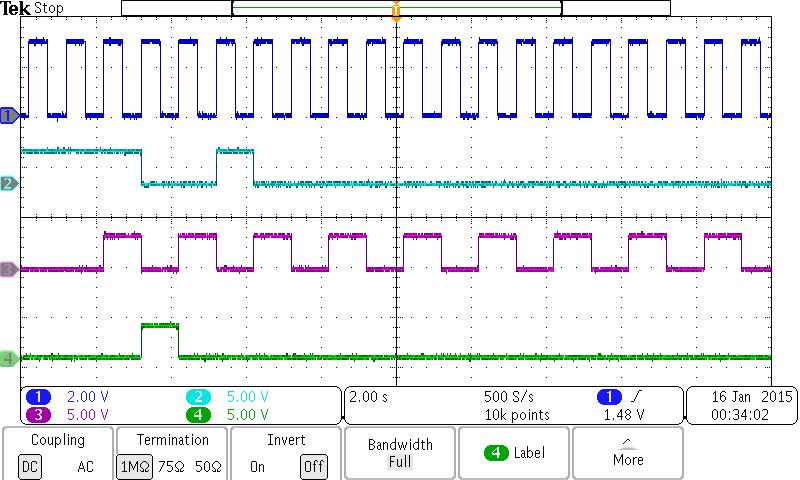


Fig 2: Day-to-night- transition

Lab Task 2: Traffic light controller with duration

**VHDL code:**

library ieee;

use ieee.std\_logic\_1164.all;

use ieee.numeric\_std.all;

entity TRAFFIC\_LIGHT\_W\_CNT is

generic(L : integer := 8);

port(RESET, CLK, DN : in bit;

MAIN, SEC : out bit\_vector(4 downto 0));

end entity TRAFFIC\_LIGHT\_W\_CNT;

architecture CONTROLLER of TRAFFIC\_LIGHT\_W\_CNT is

signal Z : unsigned (3 downto 0);

signal CLEAR : bit;

signal COUNTER : bit\_vector(3 downto 0);

component MOD\_N\_CNT is

generic(N : integer := 8);

port ( ARESETN, CLK, CLRN : in bit;

CNT : out bit\_vector (3 downto 0));

end component MOD\_N\_CNT;

begin

INST: MOD\_N\_CNT

generic map(L)

port map(RESET, CLK, CLEAR, COUNTER);

P1: process(CLK, RESET)

begin

if RESET = '0' then Z <= "0000" after 5 ns;

elsif CLK = '1' and CLK' event then

case Z is

when "0000" => MAIN <= "01010" after 5 ns; SEC <= "01010" after 5 ns;

if CLEAR = '1' then Z <= "0001" after 5 ns;

end if;

when "0001" => MAIN <= "10010" after 5 ns; SEC <= "10010" after 5 ns;

if CLEAR = '1' then

if DN = '0' then Z <= "1110" after 5 ns;

else Z <= "0010" after 5 ns;

end if;

end if;

when "0010" => MAIN <= "11001" after 5 ns; SEC <= "10010" after 5 ns;

if CLEAR = '1' then Z <= "0011" after 5 ns;

end if;

when "0011" => MAIN <= "00101" after 5 ns; SEC <= "10010" after 5 ns;

if CLEAR = '1' then Z <= "0100" after 5 ns;

end if;

when "0100" => MAIN <= "01010" after 5 ns; SEC <= "10010" after 5 ns;

if CLEAR = '1' then Z <= "0101" after 5 ns;

end if;

when "0101" => MAIN <= "10010" after 5 ns; SEC <= "10010" after 5 ns;

if CLEAR = '1' then Z <= "0110" after 5 ns;

end if;

when "0110" => MAIN <= "10010" after 5 ns; SEC <= "11001" after 5 ns;

if CLEAR = '1' then Z <= "0111" after 5 ns;

end if;

when "0111" => MAIN <= "10010" after 5 ns; SEC <= "00101" after 5 ns;

if CLEAR = '1' then Z <= "1000" after 5 ns;

end if;

when "1000" => MAIN <= "10010" after 5 ns; SEC <= "01010" after 5 ns;

if CLEAR = '1' then

if DN = '0' then Z <= "0001" after 5 ns;

else Z <= "0010" after 5 ns;

end if;

end if;

when "1110" => MAIN <= "00000" after 5 ns; SEC <= "01000" after 5 ns;

if CLEAR = '1' then Z <= "1111" after 5 ns;

end if;

when "1111" => MAIN <= "00000" after 5 ns; SEC <= "00000" after 5 ns;

if CLEAR = '1' then

if DN = '0' then Z <= "1110" after 5 ns;

else Z <= "0000" after 5 ns;

end if;

end if;

when others => MAIN <= "00000" after 5 ns; SEC <= "00000" after 5 ns;

end case;

end if;

end process P1;

P2: process(COUNTER, CLK)

begin

CLEAR <= '0' after 5 ns;

case Z is

when "0000" => if COUNTER = "0001" then CLEAR <= '1' after 5 ns; end if;

when "0001" => if COUNTER = "0001" then CLEAR <= '1' after 5 ns; end if;

when "0010" => if COUNTER = "0000" then CLEAR <= '1' after 5 ns; end if;

when "0011" => if COUNTER = "0111" then CLEAR <= '1' after 5 ns; end if;

when "0100" => if COUNTER = "0000" then CLEAR <= '1' after 5 ns; end if;

when "0101" => if COUNTER = "0001" then CLEAR <= '1' after 5 ns; end if;

when "0110" => if COUNTER = "0000" then CLEAR <= '1' after 5 ns; end if;

when "0111" => if COUNTER = "0011" then CLEAR <= '1' after 5 ns; end if;

when "1000" => if COUNTER = "0000" then CLEAR <= '1' after 5 ns; end if;

when "1110" => if COUNTER = "0000" then CLEAR <= '1' after 5 ns; end if;

when "1111" => if COUNTER = "0000" then CLEAR <= '1' after 5 ns; end if;

when others => CLEAR <= '0' after 5 ns;

end case;

end process P2;

end CONTROLLER;

library ieee;

use ieee.std\_logic\_1164.all;

use ieee.numeric\_std.all;

entity MOD\_N\_CNT is

generic(N : integer := 8);

port ( ARESETN, CLK, CLRN : in bit;

CNT : out bit\_vector (3 downto 0));

end entity MOD\_N\_CNT;

architecture BEHAVIORAL of MOD\_N\_CNT is

signal S : unsigned(3 downto 0);

begin

P\_SYNC : process (ARESETN, CLK)

begin

if ARESETN = '0' then S <= to\_unsigned(0,S'length) after 5 ns;

elsif CLK'event and CLK='1' then

if CLRN = '1' then S <= to\_unsigned(0,S'length) after 5 ns;

elsif S=to\_unsigned(N-1,S'length) then S<=to\_unsigned(0,S'length) after 5 ns;

else S <= S + 1 after 5 ns;

end if;

end if;

end process P\_SYNC;

P\_OUT : process (S)

begin

CNT <= to\_bitvector(std\_logic\_vector(S));

end process P\_OUT;

end architecture BEHAVIORAL;

**SCREENSHOTS:**

Pink = RED Blue = AMBER Green = GREEN

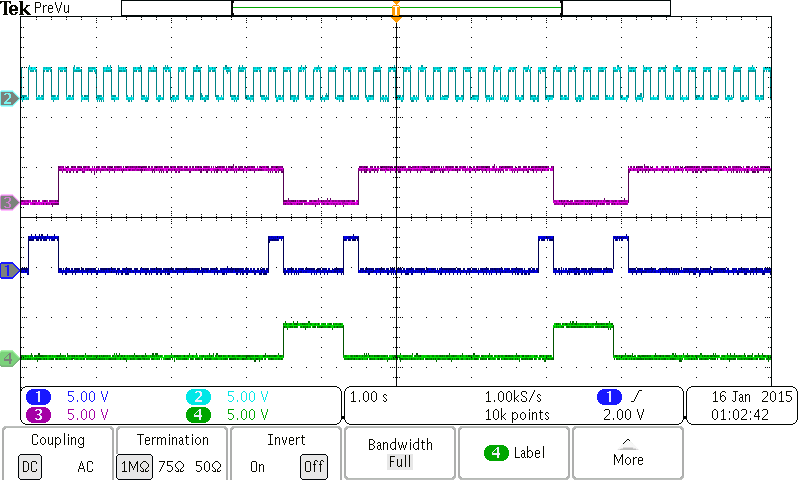


Fig 3: Day cycle for secondary road

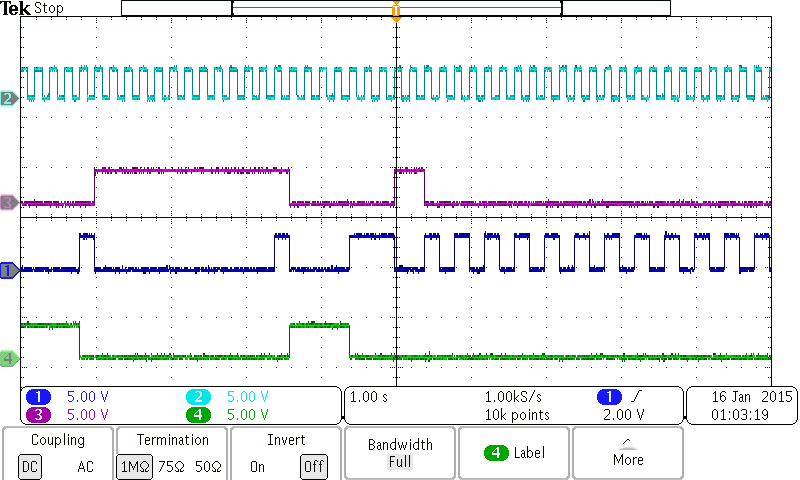


Fig 4: Day-to-night transition

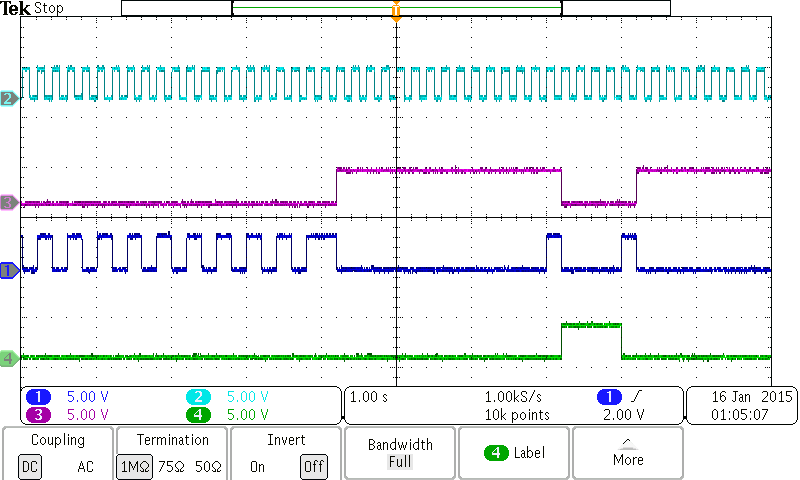


Fig 5: Night-to-day transition