Univerzitet Crne Gore
Elektrotehnički fakultet

Predmet:

Automatizovano projektovanje elektronskih kola i sistema (bivši SEK)

Tema:

Semafor na FPGA DE2-70 ploči

Traffic light controller on FPGA DE2-70 board

Autori:

Dejan Brajović 15/18

Božidar Andrović 9/18

Datum i mjesto:

17.11.2018.g Podgorica

Sadrzaj:

Sažetak:

Semafor na FPGA DE2-70 ploči

Abstract:

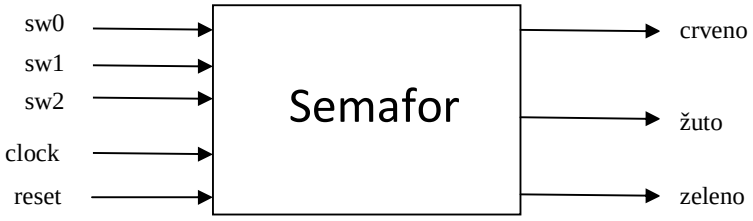
Traffic light controller: RED, YELLOW, GREEN, YELLOW, RED. With the SW1 and SW2 switches four red and green intervals are selected: 20 sec, 30 sec, 40 sec, 50 sec. With SW0 the duration of the yellow is 5 or 10 sec.

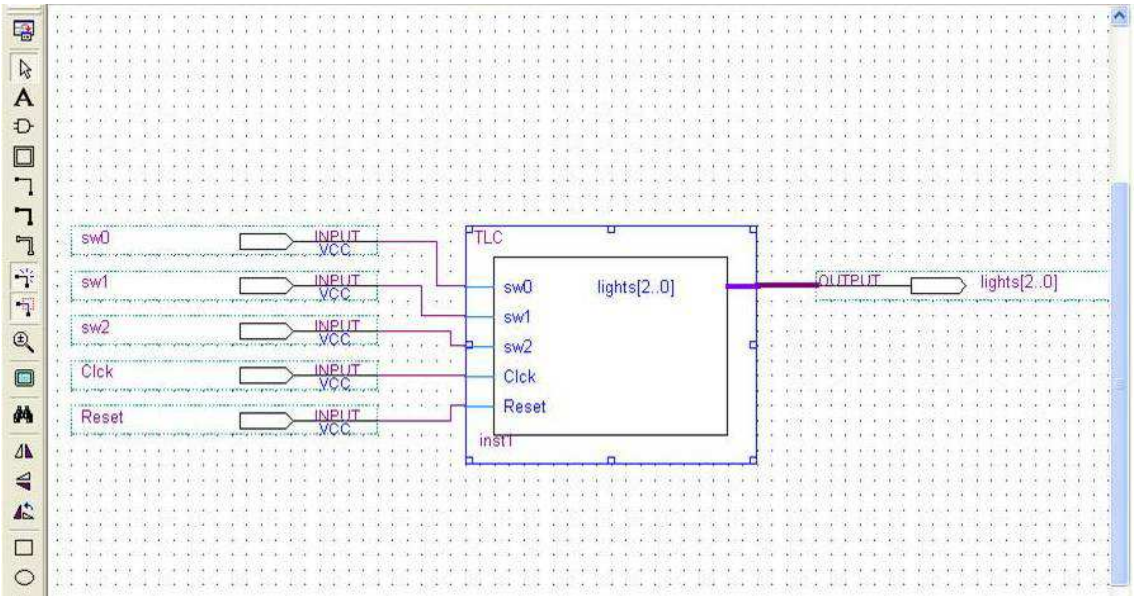
1. Opis Problema:

Projektovati semafor: CRVENO, ŽUTO, ZELENO,ŽUTO, CRVENO. Žuto traje mnogo manje. Sa prekidačima SW1 i SW2 biraju se 4 intervala crvenog i zelenog: 20S, 30s,40s,50s. Sa SW0 bira se trajanje žutog, 5 ili 10 sek.

2. Harversko-softversko resenje i simulacija:

2.1 Blok šema semafora:





2.2 Objašnjenje kola:
Imamo 5 ulaza i 3 izlaza. Prva tri su sw0, sw1 i sw2 koji služe za biranje intervala crvenog , žutog i zelenog svijetla.Sa sw0 bira se trajanje žutog 5 ili 10 sekundi , a sa sw1 i sw2 se biraju 4 intervala crvenog i zelenog 20,30,40 ili 50 sekundi. Clock predstavlja interni signal FPGA ploče od 50MHz.

2.3 Tabela stanja:

SW0	SW1	SW2	CRVENO	ŽUTO	ZELENO
0	0	0	20 s	5s	20s
0	0	1	30s	5s	30s
0	1	0	40s	5s	40s
0	1	1	50s	5s	50s
1	0	0	20s	10s	10s
1	0	1	30s	10s	30s
1	1	0	40s	10s	40s
1	1	1	50s	10s	50s

2.4 Kod :

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use ieee.std_logic_unsigned.all;
entity TLC is
        Port (sw0,sw1,sw2: in std_logic;
        lights : out STD_LOGIC_Vector (2 downto 0);
       Clck : in STD_LOGIC;
        Reset : in STD_LOGIC);
end TLC;

architecture Behavioral of TLC is
        type state_type is (st0_R1, st1_R1_A1, st2_G1, st3_A1);
       signal state: state_type;
        signal count : std_logic_vector (31 downto 0);
       constant sec5 : std_logic_vector (31 downto 0)  := "00001110111001101011001010000000";
       constant sec10 : std_logic_vector (31 downto 0) := "00011101110011010110010100000000";
        constant sec20: std_logic_vector (31 downto 0 ) := "00111011100110101100101000000000";
        constant sec30: std_logic_vector (31 downto 0 ) := "01011001011010000010111100000000";
        constant sec40: std_logic_vector (31 downto 0 ) := "01110111001101011001010000000000";
        constant sec50: std_logic_vector (31 downto 0 ) := "10010101000000101111100100000000";
begin
 process (Clck,Reset,sw0,sw1,sw2)
        begin
        if Reset='1' then
                state <= st0_R1; --reset to initial state
               count <= "00000000000000000000000000000000"; -- reset counter
        elsif ( Clck' event and Clck = '1') then --
               if(sw0='0' and sw1='0' and sw2='0') then
               case (state) is
                       when st0_R1 =>
                       if count < sec20 then
                               state <= st0_R1;
                               count <= count + 1;
                       else
                                state <= st1_R1_A1;
                                count <= "00000000000000000000000000000000";
                        end if;
                        when st1_R1_A1 =>
                        if count < sec5 then
                                state <= st1_R1_A1;
                               count <= count + 1;
                        else
                                state <= st2_G1;
```

```
                count <= "00000000000000000000000000000000";
        end if;
        when st2_G1 =>
        if count < sec20 then
                state <= st2_G1;
                count <= count + 1;
        else
                state <= st3_A1;
                count <= "00000000000000000000000000000000";
        end if;
        when st3_A1=>
        if count < sec5 then
                state <= st3_A1;
                count <= count + 1;
        else
                state <=st0_R1;
                count <= "00000000000000000000000000000000";
        end if;
                when others =>
                state <= st0_R1;
 end case;
end if;
if(sw0='0' and sw1='0' and sw2='1') then
case (state) is
        when st0_R1 =>
        if count < sec30 then
                state <= st0_R1;
                count <= count + 1;
        else
                state <= st1_R1_A1;
                count <= "00000000000000000000000000000000";
        end if;
        when st1_R1_A1 =>
        if count < sec5 then
                state <= st1_R1_A1;
                count <= count + 1;
        else
                state <= st2_G1;
                count <= "00000000000000000000000000000000";
        end if;
        when st2_G1 =>
        if count < sec30 then
                state <= st2_G1;
                count <= count + 1;
        else
                state <= st3_A1;
                count <= "00000000000000000000000000000000";
        end if;
        when st3_A1=>
        if count < sec5 then
                state <= st3_A1;
                count <= count + 1;
        else
                state <=st0_R1;
                count <= "00000000000000000000000000000000";
        end if;
        when others =>
                state <= st0_R1;
end case;
 end if;
if(sw0='0' and sw1='1' and sw2='0') then
 case (state) is
        when st0_R1 =>
        if count < sec40 then
                state <= st0_R1;
                count <= count + 1;
        else
```

```
                    state <= st1_R1_A1;
                    count <= "00000000000000000000000000000000";
            end if;
            when st1_R1_A1 =>
             if count < sec5 then
                    state <= st1_R1_A1;
                    count <= count + 1;
            else
                    state <= st2_G1;
                    count <= "00000000000000000000000000000000";
             end if;
             when st2_G1 =>
             if count < sec40 then
                    state <= st2_G1;
                    count <= count + 1;
             else
                    state <= st3_A1;
                    count <= "00000000000000000000000000000000";
             end if;
                   when st3_A1=>
            if count < sec5 then
                    state <= st3_A1;
                     count <= count + 1;
             else
                    state <=st0_R1;
                    count <= "00000000000000000000000000000000";
            end if;
            when others =>
                     state <= st0_R1;
end case;
end if;
if(sw0='0' and sw1='1' and sw2='1') then
 case (state) is
             when st0_R1 =>
            if count < sec50 then
                    state <= st0_R1;
                    count <= count + 1;
            else
                    state <= st1_R1_A1;
                    count <= "00000000000000000000000000000000";
            end if;
                    when st1_R1_A1 =>
                    if count < sec5 then
                    state <= st1_R1_A1;
                     count <= count + 1;
            else
                    state <= st2_G1;
                     count <= "00000000000000000000000000000000";
             end if;
             when st2_G1 =>
            if count < sec50 then
                     state <= st2_G1;
                    count <= count + 1;
             else
                     state <= st3_A1;
                     count <= "00000000000000000000000000000000";
             end if;
             when st3_A1=>
             if count < sec5 then
                     state <= st3_A1;
                     count <= count + 1;
             else
                     state <=st0_R1;
                     count <= "00000000000000000000000000000000";
             end if;
            when others =>
                     state <= st0_R1;
```

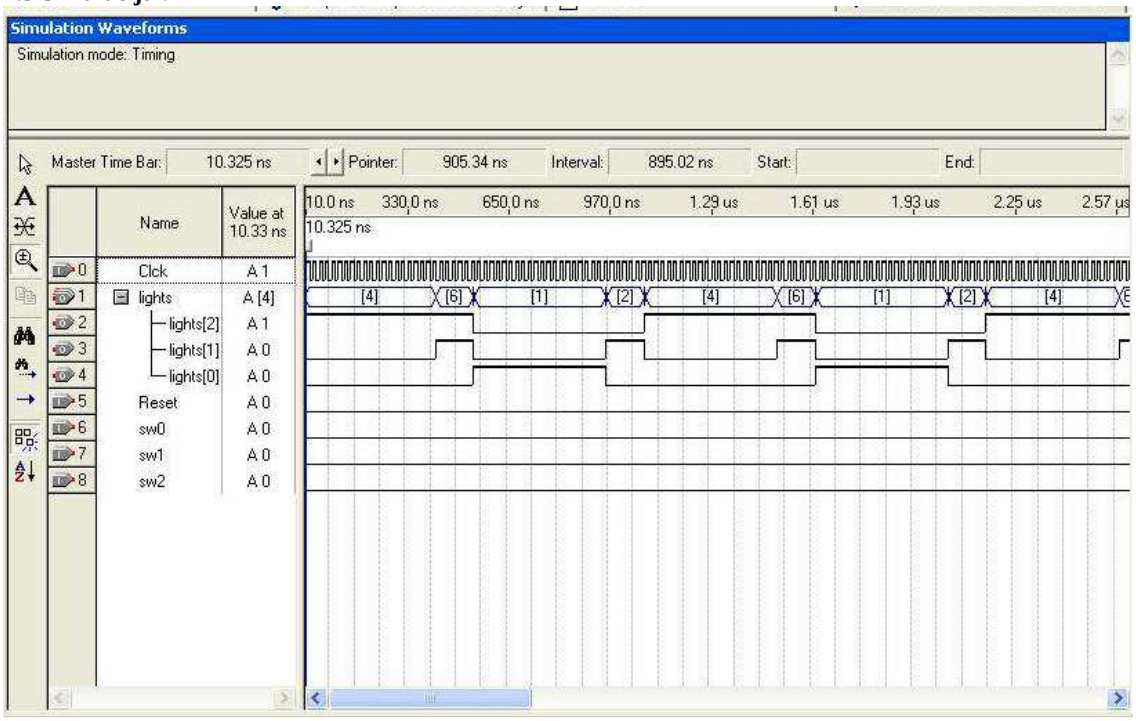
```
end case;
end if;
 if(sw0='1' and sw1='0' and sw2='0') then
case (state) is
            when st0_R1 =>
            if count < sec20 then
                        state <= st0_R1;
                        count <= count + 1;
            else
                        state <= st1_R1_A1;
                       count <= "00000000000000000000000000000000";
            end if;
            when st1_R1_A1 =>
            if count < sec10 then
                      state <= st1_R1_A1;
                        count <= count + 1;
            else
                        state <= st2_G1;
                        count <= "00000000000000000000000000000000";
            end if;
            when st2_G1 =>
            if count < sec20 then
                        state <= st2_G1;
                       count <= count + 1;
           else
                        state <= st3_A1;
                        count <= "00000000000000000000000000000000";
            end if;
            when st3_A1=>
            if count < sec10 then
                        state <= st3_A1;
                        count <= count + 1;
            else
                        state <=st0_R1;
                       count <= "00000000000000000000000000000000";
           end if;
            when others =>
                        state <= st0_R1;
end case;
end if;
if(sw0='1' and sw1='0' and sw2='1') then
case (state) is
            when st0_R1 =>
            if count < sec30 then
                        state <= st0_R1;
                        count <= count + 1;
            else
                        state <= st1_R1_A1;
                        count <= "00000000000000000000000000000000";
            end if;
            when st1_R1_A1 =>
            if count < sec10 then
                        state <= st1_R1_A1;
                       count <= count + 1;
           else
                       state <= st2_G1;
                       count <= "00000000000000000000000000000000";
           end if;
            when st2_G1 =>
            if count < sec30 then
                        state <= st2_G1;
                        count <= count + 1;
            else
                       state <= st3_A1;
                        count <= "00000000000000000000000000000000";
            end if;
           when st3_A1=>
```

```
            if count < sec10 then
                        state <= st3_A1;
                        count <= count + 1;
             else
                         state <=st0_R1;
                         count <= "00000000000000000000000000000000";
             end if;
             when others =>
                         state <= st0_R1;
 end case;
end if;
if(sw0='1' and sw1='1' and sw2='0') then
case (state) is
             when st0_R1 =>
            if count < sec40 then
                        state <= st0_R1;
                        count <= count + 1;
             else
                        state <= st1_R1_A1;
                        count <= "00000000000000000000000000000000";
            end if;
            when st1_R1_A1 =>
            if count < sec10 then
                        state <= st1_R1_A1;
                        count <= count + 1;
            else
                        state <= st2_G1;
                        count <= "00000000000000000000000000000000";
             end if;
                        when st2_G1 =>
            if count < sec40 then
                        state <= st2_G1;
                         count <= count + 1;
             else
                         state <= st3_A1;
                         count <= "00000000000000000000000000000000";
             end if;
             when st3_A1=>
             if count < sec10 then
                        state <= st3_A1;
                         count <= count + 1;
             else
                         state <=st0_R1;
                         count <= "00000000000000000000000000000000";
             end if;
             when others =>
                        state <= st0_R1;
 end case;
 end if;
 if(sw0='1' and sw1='1' and sw2='1') then
 case (state) is
             when st0_R1 =>
             if count < sec50 then
                        state <= st0_R1;
                        count <= count + 1;
            else
                         state <= st1_R1_A1;
                         count <= "00000000000000000000000000000000";
             end if;
             when st1_R1_A1 =>
             if count < sec10 then
                         state <= st1_R1_A1;
                         count <= count + 1;
             else
                         state <= st2_G1;
                         count <= "00000000000000000000000000000000";
             end if;
```

```
                        when st2_G1 =>
                        if count < sec50 then
                                state <= st2_G1;
                                count <= count + 1;
                        else
                                 state <= st3_A1;
                                 count <="00000000000000000000000000000000";
                        end if;
                        when st3_A1=>
                       if count < sec10 then
                                state <= st3_A1;
                                 count <= count + 1;
                        else
                                state <=st0_R1;
                                count <= "00000000000000000000000000000000";
                        end if;
                       when others =>
                                state <= st0_R1;
                end case;
                 end if;

        end if;
 end process;
 OUTPUT_DECODE: process (state)
        begin
        case state is
                when st0_R1 => lights <= "100"; -- Traffic Red 1, Pedestrian Green 1
                when st1_R1_A1 => lights <= "110";
                 when st2_G1=> lights <= "001";
                when st3_A1 => lights <= "010";
                 when others => lights <= "100";
         end case;
        end process;
end Behavioral;
```

2.5 Simulacija :



3. Link na video:

https://www.youtube.com/watch?v=E9hCPkj9wGY&feature=youtu.be

4. Literatura:

http://apeg.ac.me/nastava/Getting_Started_with_DE2-70_board.pdf
http://apeg.ac.me/nastava/DE2-70-pins.pdf
https://www.researchgate.net/profile/Radovan_Stojanovic/publication/279771239_AUTOMATIZOVANO_PROJEKTOVANJE_DIGITALNIH_SISTEMA_VHDL_i_FPGA/links/559aa3e508ae21086d2765f4/AUTOMATIZOVANO-PROJEKTOVANJE-DIGITALNIH-SISTEMA-VHDL-i-FPGA.pdf