Lab 6 Report

Labwork 6
VGA Controller with VHDL
05.12.2018
Şevki Gavrem Kulkuloğlu
21601793 EE 102 -2

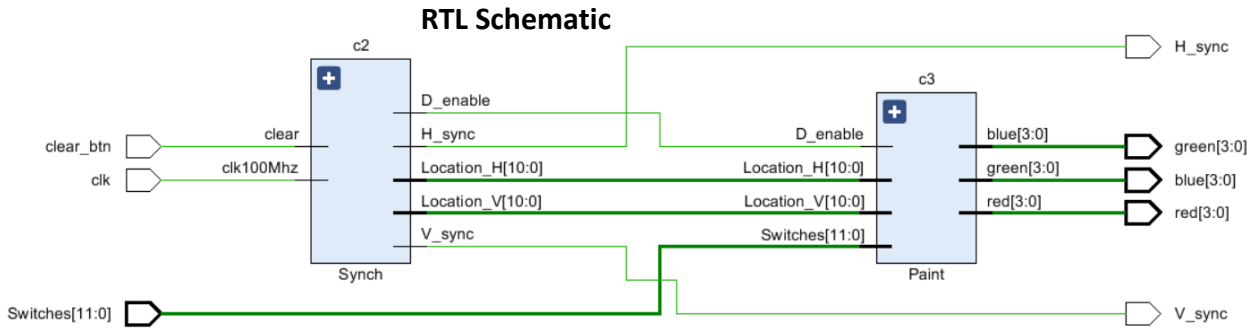
- ## Design Methodology

I started my VGA controller project with counting the pixels on screen first horizontally and when a row is done, I passed new row in downwards until every row is done. While counting I determine the pixel locations according to screens resolution which means its horizontal and vertical porches and sync pulses, so pixel locations will be in the active (displayed) screen. After I mapped out the screen, only thing remained was the color of the screen and its attachment to switches, so I assign total 12 switches which goes like 4 switches to red, 4 switches to green, 4 switches to blue. I have also assigned a button for disabling the view, simply turning off the VGA port output.

- In my preliminary lab report I stated that I want to make the initial color of the screen white. At first, I thought because at the beginning switches are '0' it made output conflicts but afterwards I got the idea that using switches as inverted way. After inverted them, it worked as I planned.

- As I worked on my project and learnt VGA more my modules and especially my design has been changed compared to my preliminary lab report.

- ## Results

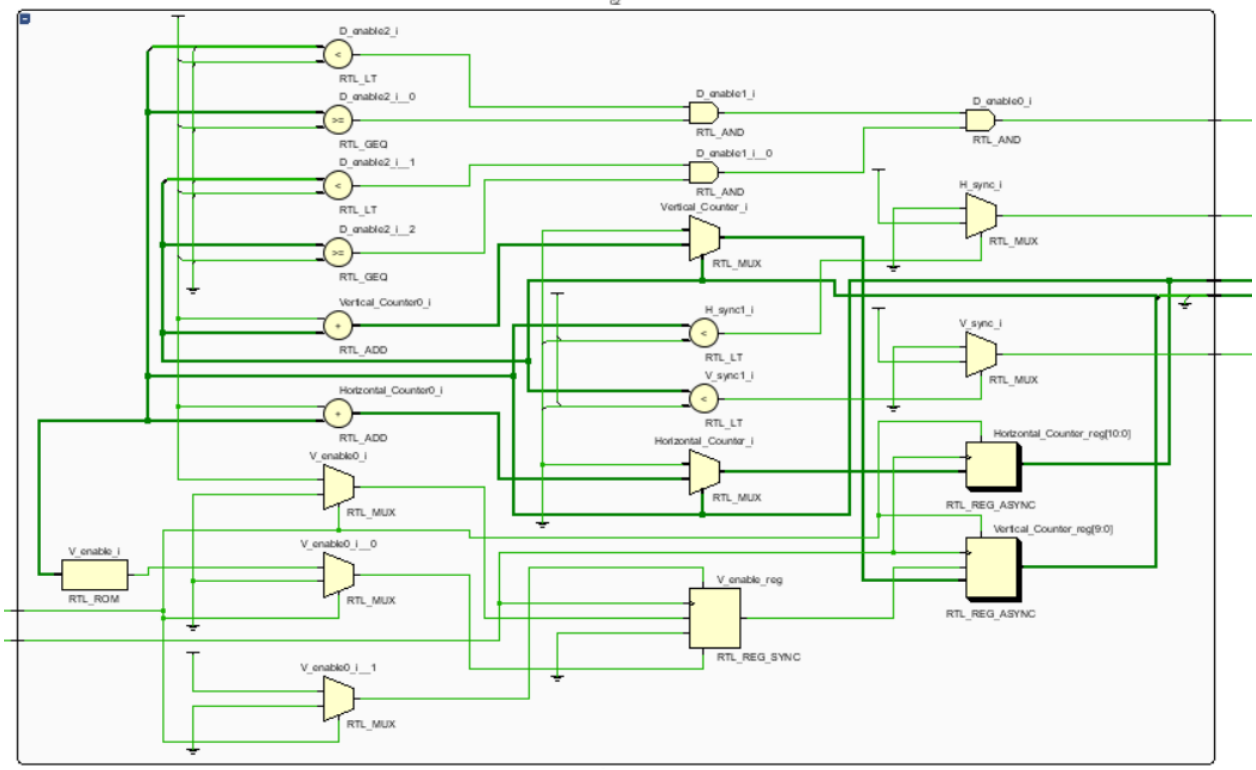


Lab 6 Report

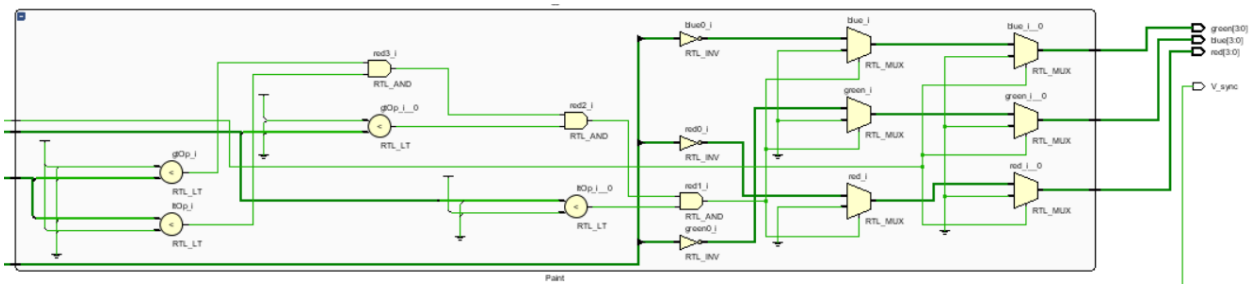
Labwork 6
VGA Controller with VHDL
05.12.2018
Şevki Gavrem Kulkuloğlu
21601793 EE 102 -2

Like it is seen in the figure above my idea at the preliminary report was substantially right but I need to make some changes in the modules and their designs.

Sync module



Paint module

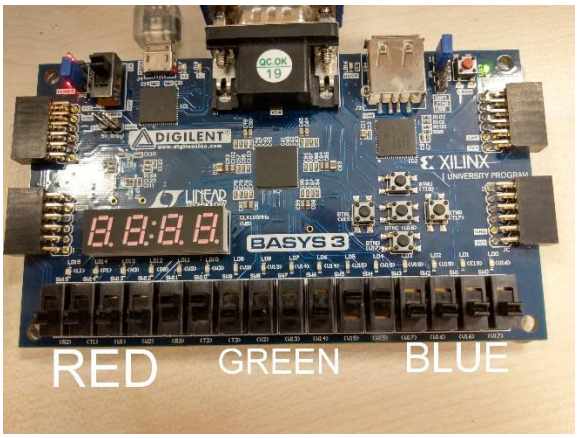


Lab 6 Report

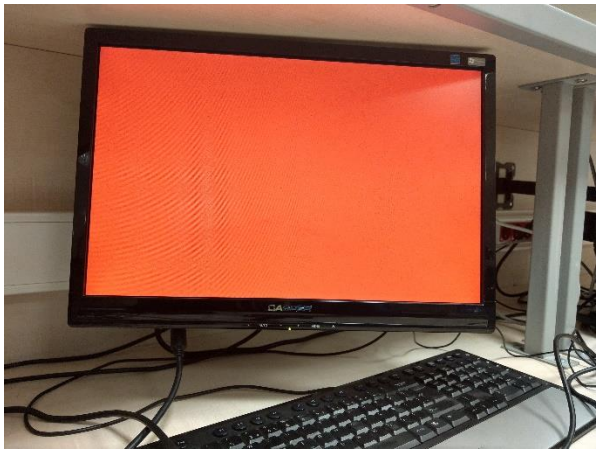
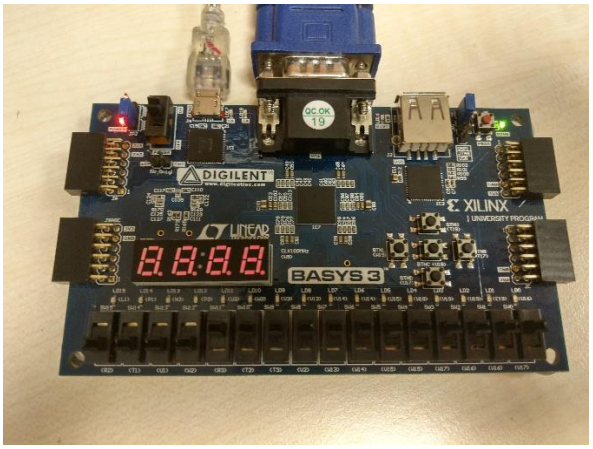
Labwork 6
VGA Controller with VHDL
05.12.2018
Şevki Gavrem Kulkuloğlu
21601793 EE 102 -2

- Screen shot of the display and Basys3

Because I want to obtain white screen at opening, I used switches as inverted. Most significant 4 switches assigned for red, 4 center switches for assigned green and least significant 4 switches for assigned blue.

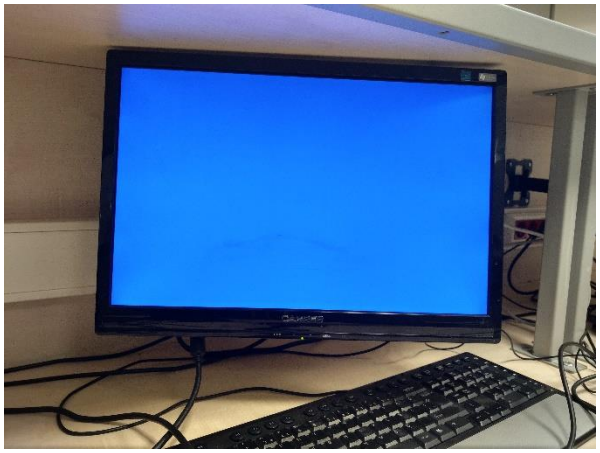
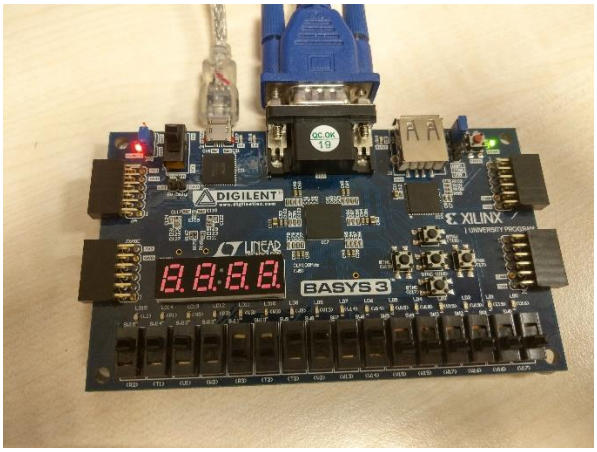
Red = “1111”, Green = “1111”, Blue = “1111”

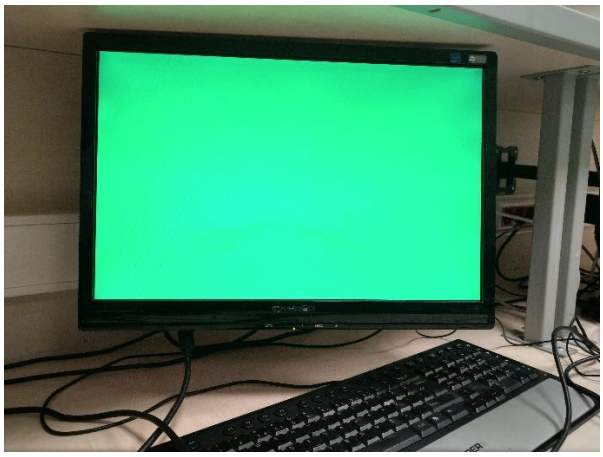
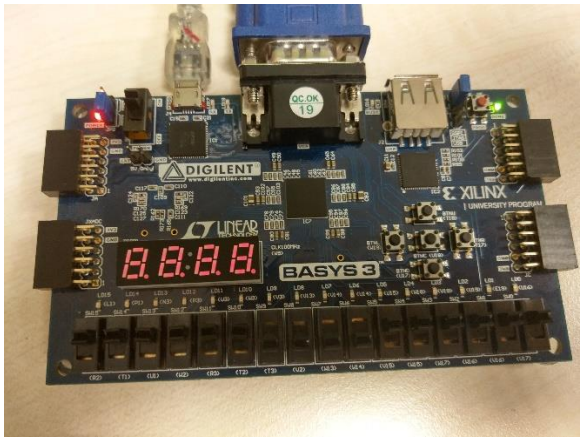
Red = “1111”, Green = “0000”, Blue = “0000”

Lab 6 Report

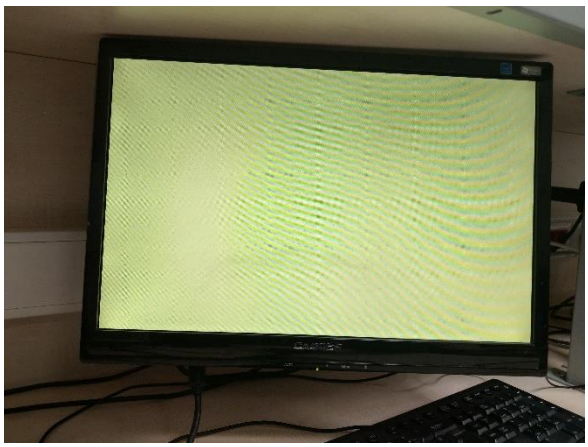
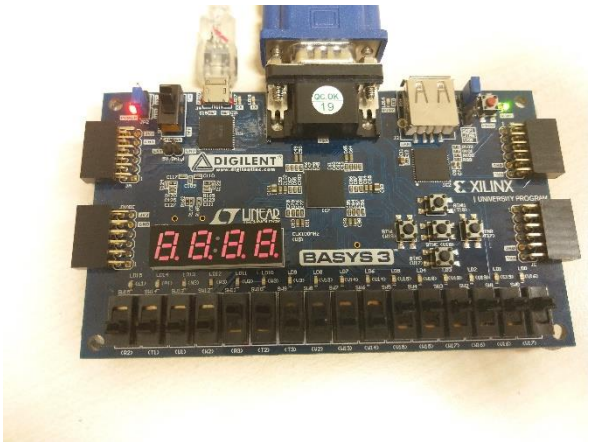
Labwork 6
VGA Controller with VHDL
05.12.2018
Şevki Gavrem Kulkuloğlu
21601793 EE 102 -2

Red = “0000”, Green = “0000”, Blue = “1111”

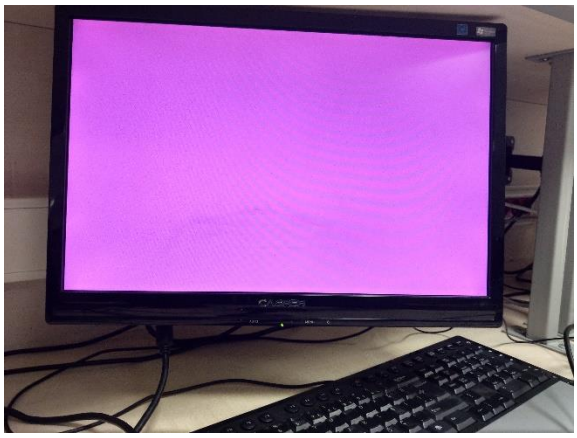
Red = “0000”, Green = “1111”, Blue = “0000”

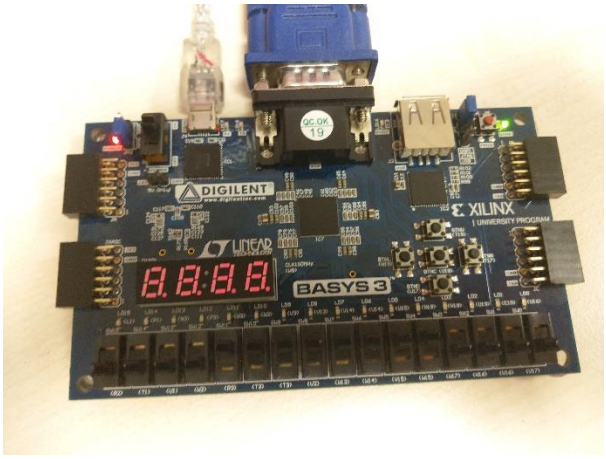



Red = “1111”, Green = “1111”, Blue = “0000”

Lab 6 Report

Labwork 6
VGA Controller with VHDL
05.12.2018
Şevki Gavrem Kulkuloğlu
21601793 EE 102 -2





Red = “1111”, Green = “0000”, Blue = “1111”

- ## Conculusion

  At first I was thinking take the resolution as 1280x1024 and I did, but I did not obtain what I expected for display, it was like 2/3 part of the screen was active and rest was black and I could not change it no matter what I did with those spects, I searched on internet about spects of the screen I used and I found that it was 1440x900. It is a rare resoulution actually. I change my code respect to those spects and afterward I finally got what I wanted from the display.

- ## Appendices

Top Module:

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;


entity Top_Module is
  Port ( clk       : in std_logic;
         clear_btn : in std_logic;
         Switches  :  in std_logic_vector(11 downto 0);
         H_sync    : out std_logic;
```

Lab 6 Report

Labwork 6
VGA Controller with VHDL
05.12.2018
Şevki Gavrem Kulkuloğlu
21601793 EE 102 -2

```
        V_sync    : out std_logic;
        red       : out std_logic_vector(3 downto 0);
        green     : out std_logic_vector(3 downto 0);
        blue      : out std_logic_vector(3 downto 0));
end Top_Module;

architecture Behavioral of Top_Module is

component Synch is
  Port ( clk100Mhz  : in std_logic;
        clear      : in std_logic;
        H_sync     : out std_logic;
        V_sync     : out std_logic;
        D_enable   : out std_logic;
        Location_H : out std_logic_vector(10 downto 0);
        Location_V : out std_logic_vector(10 downto 0));
end component;

component Paint is
  Port (D_enable   : in std_logic;
       Location_H :  in std_logic_vector(10 downto 0);
       Location_V :  in std_logic_vector(10 downto 0);
       Switches   :  in std_logic_vector(11 downto 0);
       red        :  out std_logic_vector(3 downto 0);
       blue       :  out std_logic_vector(3 downto 0);
       green      :  out std_logic_vector(3 downto 0));
end component;
```

Lab 6 Report

Labwork 6
VGA Controller with VHDL
05.12.2018
Şevki Gavrem Kulkuloğlu
21601793 EE 102 -2

```
signal D_enable : std_logic;
signal clr : std_logic;
signal Location_H : std_logic_vector(10 downto 0);
signal Location_V : std_logic_vector(10 downto 0);
begin
clr <= clear_btn;
c2 : Synch port map(clk, clr, H_sync, V_sync, D_enable, Location_H, Location_V);
c3 : Paint port map( D_enable, Location_H, Location_V,Switches, red, green, blue);

end Behavioral;
```

Sync:

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;



entity Synch is
  Port ( clk100Mhz    : in std_logic;
         clear        : in std_logic;
         D_enable     : out std_logic;
         Location_H   : out std_logic_vector(10 downto 0);
         Location_V   : out std_logic_vector(10 downto 0);
         H_sync       : out std_logic;
         V_sync       : out std_logic);
end Synch;
```

Lab 6 Report

Labwork 6
VGA Controller with VHDL
05.12.2018
Şevki Gavrem Kulkuloğlu
21601793 EE 102 -2

```
architecture Behavioral of Synch is



constant Pixels_H    : integer := 1904;
constant Pixels_V    : integer := 932;
constant hf_porch    : integer := 1824;
constant vf_porch    : integer := 931;
constant hb_porch    : integer := 232;
constant vb_porch    : integer := 28;
constant hs_pulse    : integer := 152;
constant vs_pulse    : integer := 3;

signal V_enable      : std_logic;
signal Horizontal_Counter         : integer range 0 to 1904;
signal Vertical_Counter           : integer range 0 to 932;

begin
process(clk100Mhz, clear)
begin
   if clear = '1' then
      Horizontal_Counter <= 0;
   elsif(clk100Mhz'event and clk100Mhz = '1') then
      if Horizontal_Counter = Pixels_H - 1 then
         Horizontal_Counter <= 0;
         V_enable <= '1';
      else
         Horizontal_Counter <= Horizontal_Counter + 1;
         V_enable <= '0';
```

Lab 6 Report

Labwork 6
VGA Controller with VHDL
05.12.2018
Şevki Gavrem Kulkuloğlu
21601793 EE 102 -2

```
        end if;
    end if;
end process;

H_sync <= '0' when Horizontal_Counter < 112 else '1';

process(clk100Mhz, clear)
begin
   if clear = '1' then
      Vertical_Counter <= 0;
   elsif(clk100Mhz'event and clk100Mhz = '1' and V_enable = '1') then
      if Vertical_Counter = Pixels_V - 1 then
         Vertical_Counter <= 0;
      else
         Vertical_Counter <= Vertical_Counter + 1;
      end if;
    end if;
end process;

V_sync <= '0' when Vertical_Counter < 3 else '1';

D_enable <= '1' when (((Horizontal_Counter < hf_porch) and (Horizontal_Counter >= (hb_porch + hs_pulse))) and ((Vertical_Counter < vf_porch) and (Vertical_Counter >= (vb_porch + vs_pulse)))) else '0';

Location_H <= conv_std_logic_vector(Horizontal_Counter, 11);
Location_V <= conv_std_logic_vector(Vertical_Counter, 11);
end Behavioral;
```

Lab 6 Report

Labwork 6
VGA Controller with VHDL
05.12.2018
Şevki Gavrem Kulkuloğlu
21601793 EE 102 -2

Paint

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;

entity Paint is
  Port (D_enable    : in std_logic;
        Location_H  : in std_logic_vector(10 downto 0);
        Location_V  : in std_logic_vector(10 downto 0);
        Switches    : in std_logic_vector(11 downto 0);
        red         : out std_logic_vector(3 downto 0);
        blue        : out std_logic_vector(3 downto 0);
        green       : out std_logic_vector(3 downto 0));
end Paint;

architecture Behavioral of Paint is
begin
process(D_enable,Location_H,Location_V)
begin
    red   <= "0000";
    blue  <= "0000";
    green <= "0000";
     if D_enable = '1' then
        if Location_V > 30 and Location_V < 932 and Location_H > 384 and Location_H <
1904 then
          red   <= not Switches(11 downto 8);
          green <= not Switches(7 downto 4);
```

Lab 6 Report

Labwork 6
VGA Controller with VHDL
05.12.2018
Şevki Gavrem Kulkuloğlu
21601793 EE 102 -2

```
        blue  <= not Switches(3 downto 0);
      end if;
   end if;
end process;
end Behavioral;
```