Rostock University
Faculty of Computer Science and Electronic Engineering
Electronic Circuits and Appliances Division (Institute GS)

Lab 4 "Programmable Integrated Circuits“ Control of VGA Display by FPGA-Hardware

One part of a VGA-circuit is the hardware to create signals for synchronization of the display (CRT) and the DAC (Digital to Analog Converter). Dependent upon the resolution and the picture rate of the display it has to create a distinct pattern for the signals:

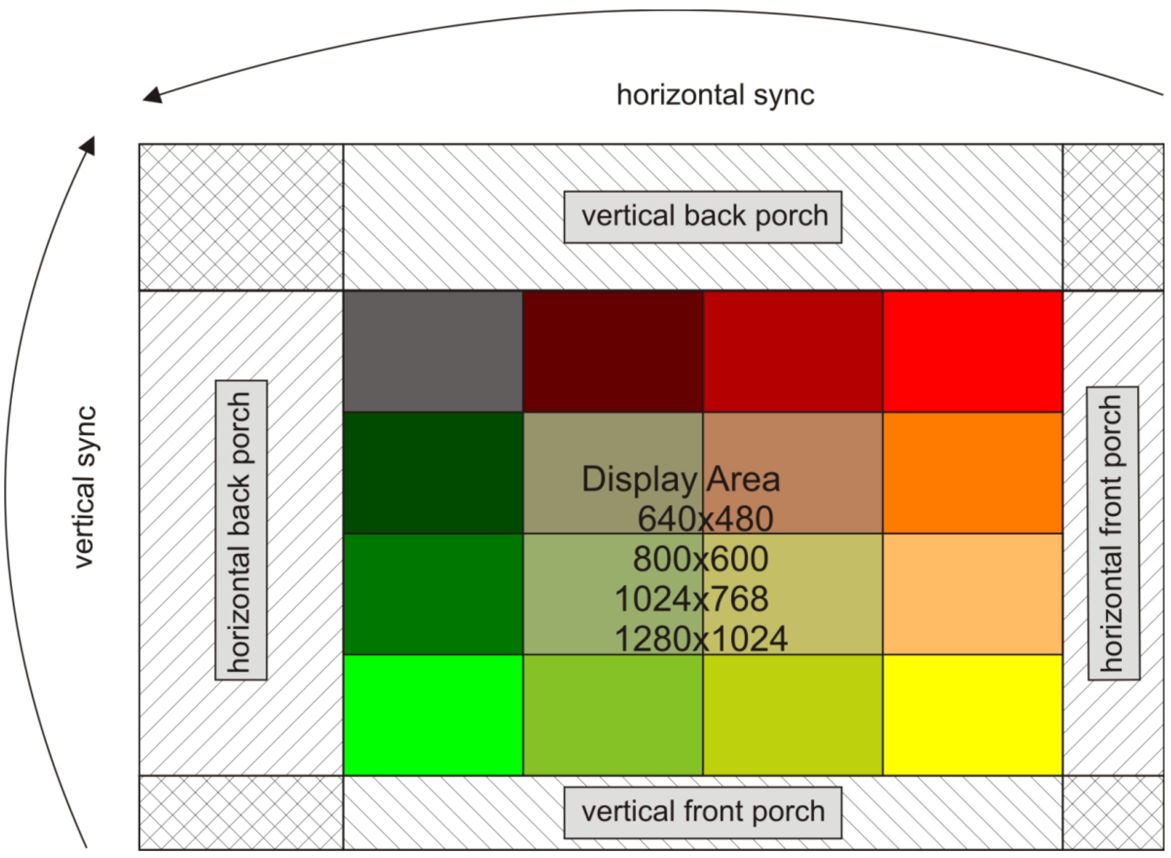
HSYNC - horizontal synchronization signal (raw)
VSYNC - vertical synchronization signal (picture) und
BLANK - blanking (setting to black-value) of the beam (horizontal and vertical), signal to the DAC

This part of a VGA controller will be designed and tested in this exercise. In addition we will generate some color patterns like a test pattern as an output to the display.

All signals are derived from the pixel clock f_p:

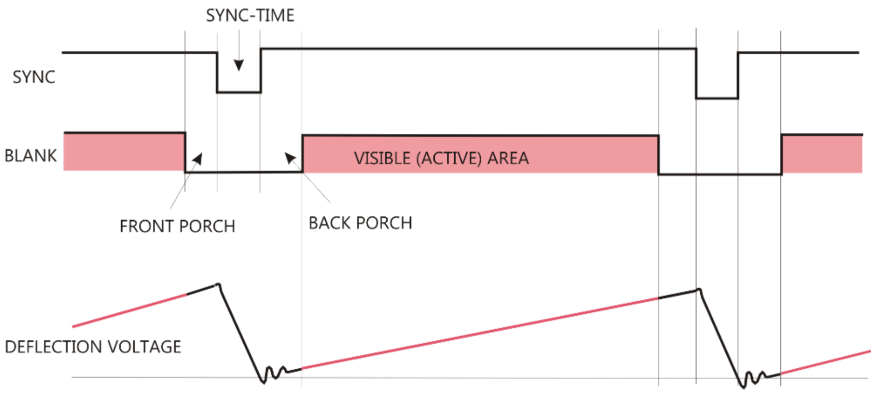
$$f_p = \text{picture rate} * \text{number of raws} * \text{number of pixels per raw}$$

The numbers of raws and pixels per row include the time (expressed in numbers of raws/pixels) for blanking outside the border of the visible picture and for the horizontal and vertical retrace of the beam.



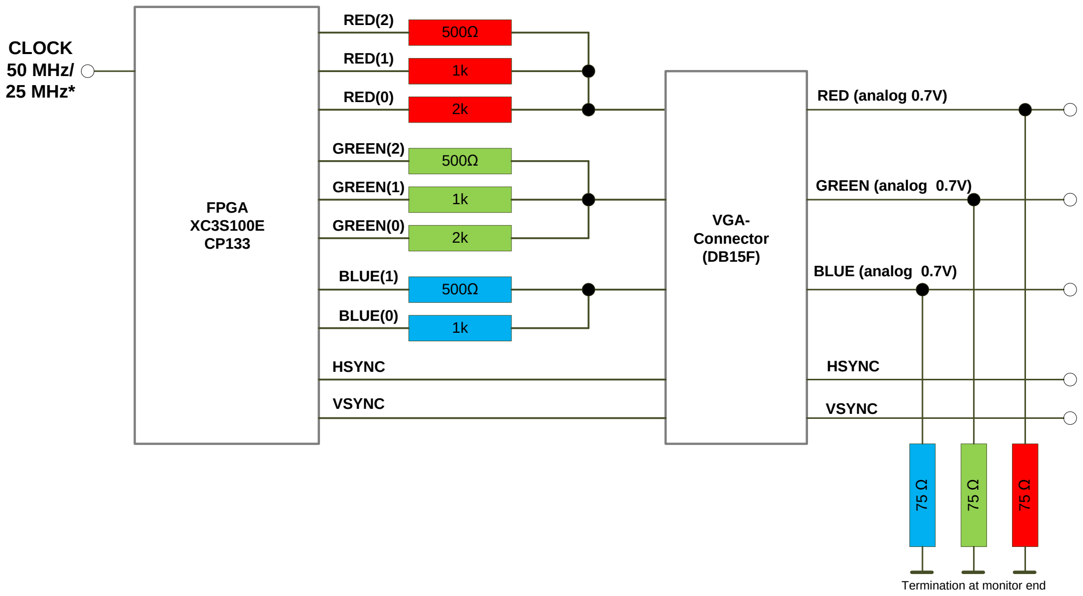
Resolution Refresh	Clock	H-Active	H-Front-Porch	H-SYNC	H-Back-Porch	V-Active	V-Front-Porch	V-SYNC	V-Back-Porch
640x480 @60Hz	25.175	640	16	96 (low)	48	480	10	2 (low)	33
640x480 @75Hz	31.5	640	16	64 (low)	120	480	1	3 (low)	16
800x600 @72Hz	50	800	56	120 (high)	64	600	37	6 (high)	23
1024x768 @60Hz	65	1024	24	136 (low)	80	768	3	6 (low)	29
1024x768 @75Hz	78,75	1024	16	96 (high)	176	768	1	3 (high)	28

The prototype board is a BASYS-2-Board from DIGILENT (http://www.digilentinc.com/basys2).



Timing of synchronization signals

The card has two clock sources. One simple RC-Generator delivering a 100 MHz master clock. The clock can be divided by 2 or 4. The clock selection is done by a jumper. 50 MHz is default without jumper applied. The clock is connected to Pad H8.
Another clock source is a 50 MHz crystal oscillator connected to Pad M6.
For our design we use the crystal clock of 50 MHz for the 800x600 @ 75 Hz resolution. The Digital to Analog Converter (DAC) is realized by a simple resistor network with 3 bits for red and green and 2 bits for blue colors. So we can generate 256 different colors including black and white.



Schematic of digital to analog converter
Pad assignment for demonstration board with XC3S100E-CP132:

CLK	M6 (B8 for simple RC-generator, Jitter!)

HSYNC	J14
VSYNC	K13

BLUE1	J13
BLUE0	H13

GREEN2	G14
GREEN1	G13
GREEN0	F14

RED2	F13
RED1	D13
RED0	C14

Todo:

1. Create a design for the 800x600 resolution from the given table with the pixel clock as input and the signals HSYNC and VSYNC as outputs. In addition it shall produce a picture at the three 2 bit color outputs. The Picture can be a simple background of unique color or a color pattern. Be aware that we have no blanking function for our "mini-DAC", so we have to hold the outputs at black (=blank) level ("00000000") for the blanking time.

2. Divide the design into the parts:
   - Horizontal counter
   - Vertical counter
   - Creation of RGB signals

3. Generate appropriate counters and comparators by the XILINX Core generator and use this self created parts in your **schematic**.

4. Simulate using a given VHDL-Testbench!

5. Synthesize the design with XST (Xilinx Synthesis Tool)!

6. Simulate the synthesized design! Can your design work with the given pixel clock?

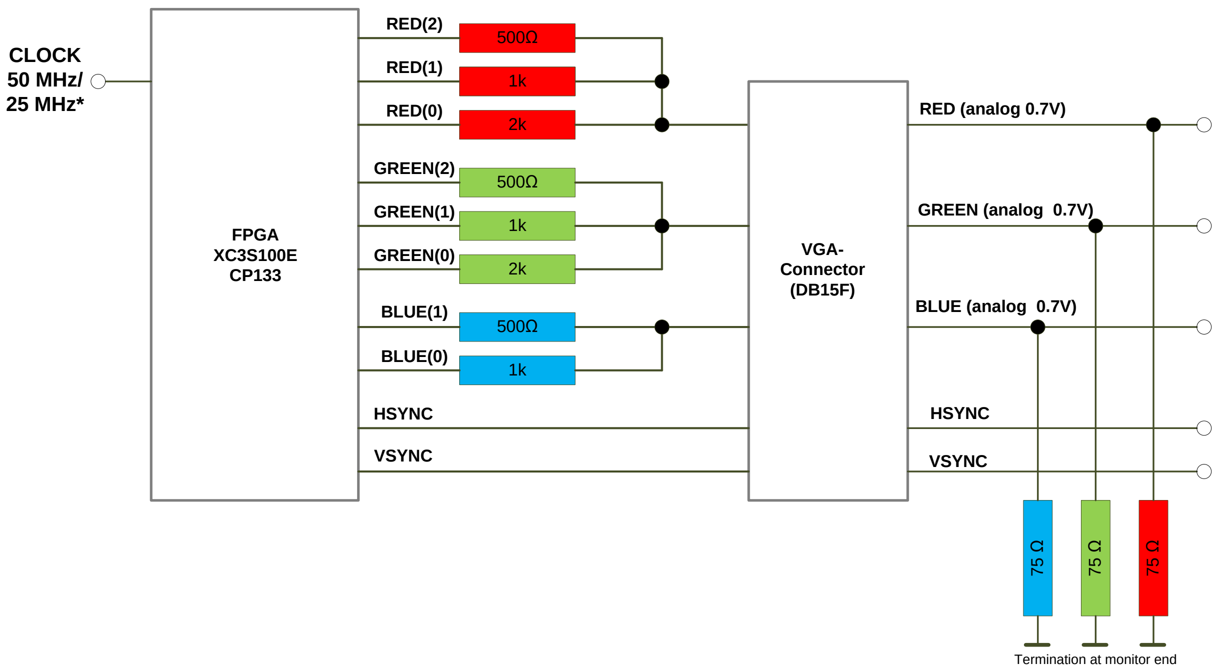
7. Implement the design!

8. Open the design inside FPGA editor and try to recognize your description in the implemented hardware!

8. Download the design to a test board and demonstrate that it works!

Addendum:

Calculation of analog voltage and driver strength:



$$\frac{(U_a - U_2)}{R_2} + \frac{(U_a - U_1)}{R_1} + \frac{(U_a - U_0)}{R_0} = 0$$

$$(U_a - U_2) \cdot R_1 R_0 + (U_a - U_1) \cdot R_2 R_0 + (U_a - U_0) \cdot R_2 R_1 = 0$$

$$U_a = \frac{R_1 R_0 \cdot U_2 + R_2 R_0 \cdot U_1 + R_2 R_1 \cdot U_0}{R_1 R_0 + R_2 R_0 + R_2 R_1} = \frac{1k\Omega \cdot 2k\Omega \cdot U_2 + 0.5k\Omega \cdot 2k\Omega \cdot U_1 + 0.5k\Omega \cdot 1k\Omega \cdot U_0}{1k\Omega \cdot 2k\Omega + 0.5k\Omega \cdot 2k\Omega + 0.5k\Omega \cdot 1k\Omega}$$

$$U_a = \frac{U_H}{3.5}(2 \cdot Bit2 + 1 \cdot Bit1 + 0.5 \cdot Bit0) = \frac{U_H}{7}(4 \cdot Bit2 + 2 \cdot Bit1 + 1 \cdot Bit0)$$

$$U_{amax} = \frac{3.3V}{7}(4 + 2 + 1) = 3.3V \quad (I/O - Voltage)$$

If the wires for the analog colour information are terminated by 75 Ω resistors we may calculate the voltage divider of the source impedance and the termination resistor:

$$R_a = R_1||R_2||R_3 = \frac{R_2 R_1 R_0}{R_2 R_1 + R_2 R_0 + R_1 R_2} = \frac{2 \cdot 1 \cdot 0{,}5}{2 + 1 + 0{,}5} k\Omega = \frac{1}{3{,}5} k\Omega = 286\Omega$$

$$U_{VGAmax} = \frac{75}{286 + 75} 3.3V = 0.68V$$

Maximum current at FPGA output:

Bit2 drives high level voltage over R2 and the parallel $R_T||R_1||R_0$ to ground.

$$I_{max} = \frac{U_{IO}}{500\Omega + (75\Omega||1k\Omega||2k\Omega)} = \frac{3{,}3V}{500\Omega + 67\Omega} = 5{,}8\ mA$$

IO should be initialized to have 8 mA drivers and 3.3V output voltage.