EE 310 Hardware Description Languages
Spring 2022

Laboratory Assignment #4

Due Date: 23 June 2022

Design and implement a digital Gaussian filter hardware. Then, verify your Verilog RTL implementation in the Xilinx Spartan FPGA on the DIGILENT Spartan-3E starter board.

Your hardware should first apply the Gaussian filter with filter coefficients {1/4, 2/4, 1/4} to rows of a 128x128 grayscale input image (128x128 = 16384 unsigned 8-bit pixels). It should keep the pixel values in the first column and last column as they are. Your hardware should then apply the same Gaussian filter to columns of the filtered 128x128 image. It should keep the pixel values in the first row and last row as they are. An example for a 6x6 input image is shown below. Red values are kept as they are.

(15 + 2x20 + 25) / 4 = 20

(15 + 2x20 + 20) / 4 = 19

15	20	25	25	15	10
20	15	50	30	20	15
20	50	55	60	30	20
20	15	65	30	15	30
15	20	30	20	25	30
20	25	15	20	10	15

15	20	24	23	16	10
20	25	36	33	21	15
20	44	55	51	35	20
20	29	44	35	22	30
15	21	25	24	25	30
20	21	19	16	14	15

15	20	24	23	16	10
19	28	38	35	23	15
20	35	48	43	28	21
19	31	42	36	26	28
18	23	28	25	22	21
20	21	19	16	14	15

You are given an example hardware (lab4_example) which takes 128x128 grayscale lena image as input, negates its pixels, and displays the input grayscale image and negated grayscale image as black and white on a VGA monitor connected to the FPGA board as shown below.

In DIGILENT Spartan-3E starter development board, one FPGA pin for each color channel (R, G, B) is connected to the VGA port. In this lab, input and output images will be grayscale images (each pixel is 8 bits), and they should be displayed on a VGA monitor as black and white (each pixel is 1 bit). To display a grayscale image on a VGA monitor as black and white, you should assign 1 to R, G and B channels for the pixels with a value greater than or equal to 128, and assign 0 to R, G and B channels for the pixels with a value less than 128.

You should modify the display_template module in the display_template.v file given in lab4_example to implement Gaussian filter hardware instead of negation hardware. You should not change the interface of the display_template module. You should use the pins.ucf file given in lab4_example. Your hardware should work at 50 MHz.

In your hardware, you are not allowed to use multiplier or divider hardware for implementing multiplication with a constant and division with a constant. These operations should be implemented using shifter and adder hardware.

You should reset your hardware by pressing button BTN_SOUTH. Then, your hardware should wait until button BTN_WEST is pressed. When BTN_WEST is pressed, it should display the 128x128 lena grayscale input image as black and white on the left side of the VGA monitor, perform Gaussian filter on this input image, store the filtered 128x128 grayscale output image to a BlockRAM and lit LED1 (rightmost LED). Your hardware should then wait until button BTN_EAST is pressed. When BTN_EAST is pressed, it should display the output image as black and white on the right side of the VGA monitor (while the input image is still on the left side of the VGA monitor), and lit LED2 (next to the rightmost one).

Synthesize and implement your Verilog RTL code targeting Xilinx Spartan XC3S500E FG320 FPGA with speed grade 4 using Xilinx ISE. Then, generate the FPGA configuration bitstream, download the bitstream into Xilinx Spartan FPGA, and verify your Verilog RTL implementation on the board.

Put all your Verilog files, UCF file and FPGA bitstream into a zip file named Lab4_Partner1Lastname_Partner2Lastname.zip (e.g., *Lab4_Mahdavi_Hamzaoglu.zip*) and submit this zip file to EE310 SUCourse+.