Spring 2021

California State University, Northridge
Department of Electrical & Computer Engineering

Mini Project 2
HLS - Image Processing

May 15, 2021

ECE 520L

Written By: Jose Luis Martinez

Mini Project 2: Run Instructions

1. The zipped file will contain the following:
    a. Vivado HLS Project
        i. This contains the source project for Mini Project 1.
        ii. All 6 processes are run at the same time by a top level function calling them each once.
        iii. There are 5 solutions to accommodate for each of the constraints.
    b. Matlab Script
        i. This is used to generate the image into a 100x100 grayscale matrix ready for hls.
        ii. The same script will also take in matrices generated by HLS and show them as images.
    c. Verification Video
    d. Source code **image_manip.cpp**, **image_manip.h**, and **image_manip_tb.cpp**.
    e. All images used in this instruction report.
2. This project was created using Vivado HLS 2019.1 and ZYBO was used as the target hardware device. Make sure to extract the folder before starting.
    a. To open this project, go to Vivado HLS and open the folder Miniporject2hls inside the zipped folder as the project.

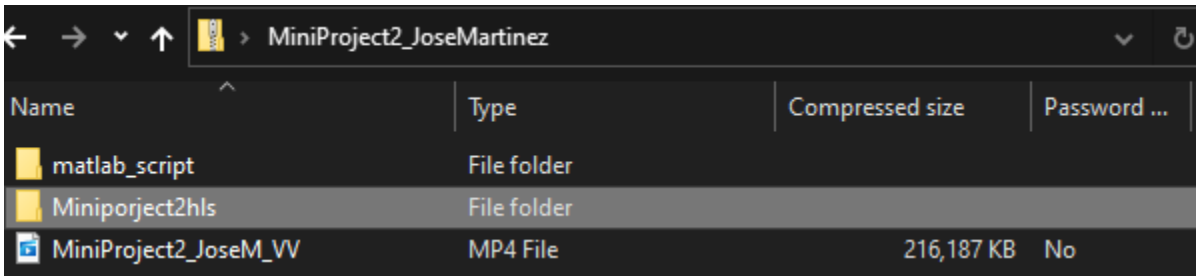


Fig. 1 Vivado HLS Project Directory

3. Once in Vivado HLS, you can verify anything that was required of this mini project.
    a. To verify Synthesis go to solution1 → syn → report and open the report found there as shown below.
    b. To verify the constraints just set the active solution to the one you want to check.

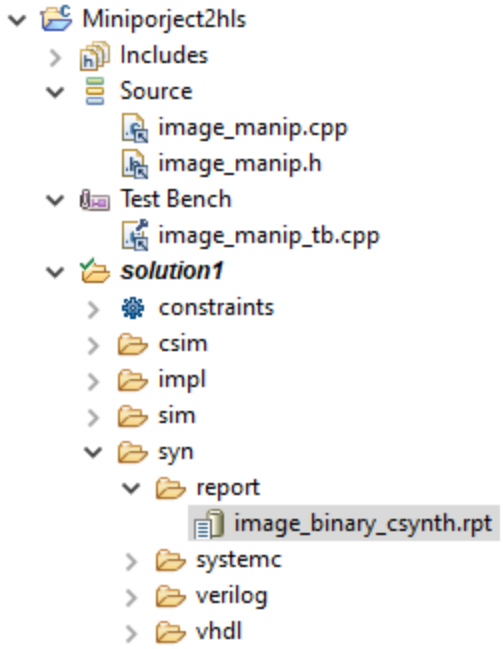


Fig. 2 Synthesis Report Location

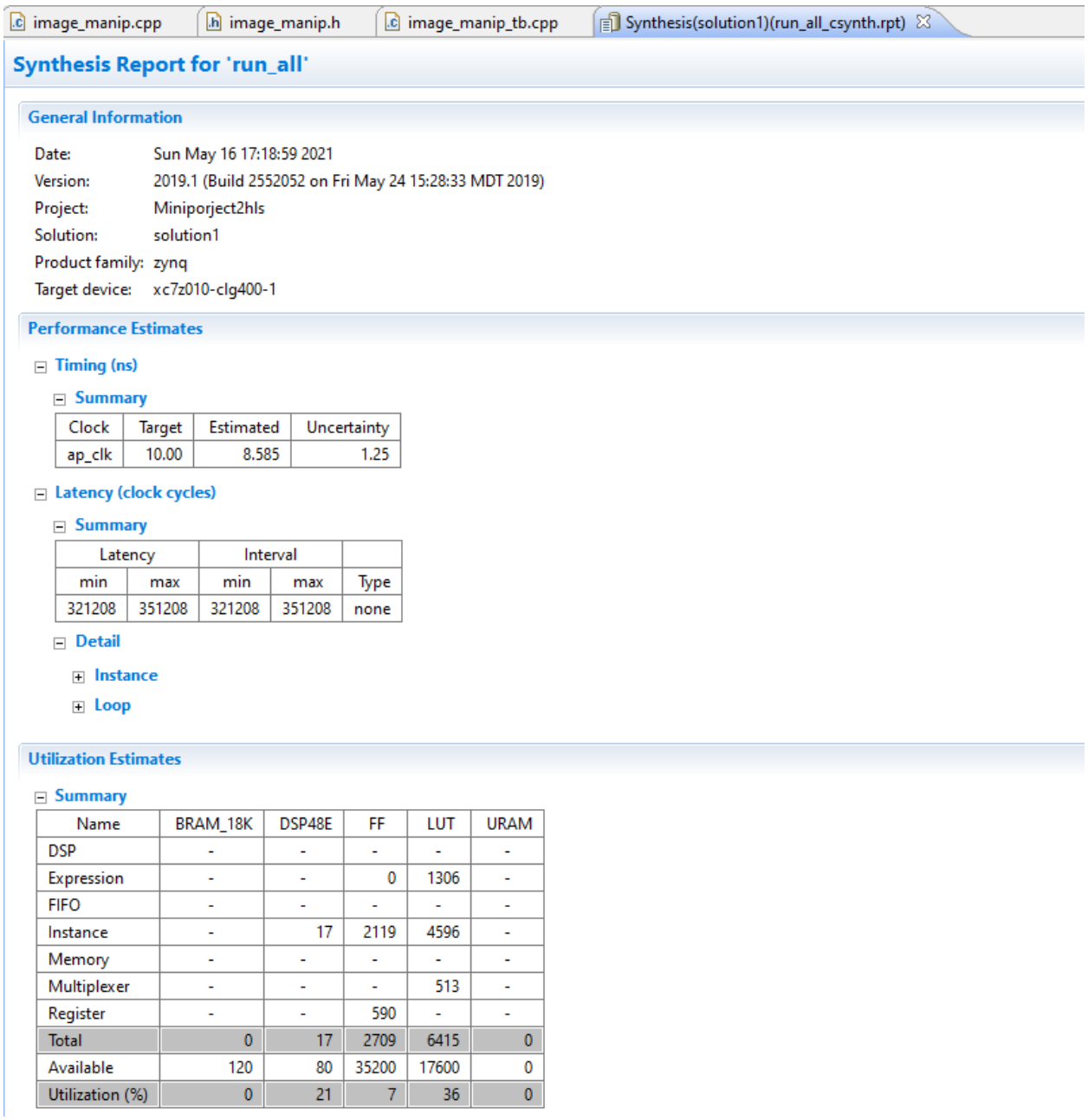


Fig. 3 Synthesis Report

4. I used Matlab to generate the original input image for me.
    a. Go to the matlab_scripts folder and open r_g.m.
    b. Run it and you should see a matrix printed out in the console.
    c. Copy and paste that into the Testbench file in hls.
    d. Run the C Simulation and you should see a similar output to **Fig. 4.**
    e. Each row of numbers corresponds to an image except the last one.
    f. Paste each in the following order: bin_image, ave_image, invert_image, scale_image, and decomp_image.
    g. When you run the mat lab simulation again you should see something like **Fig. 6.**

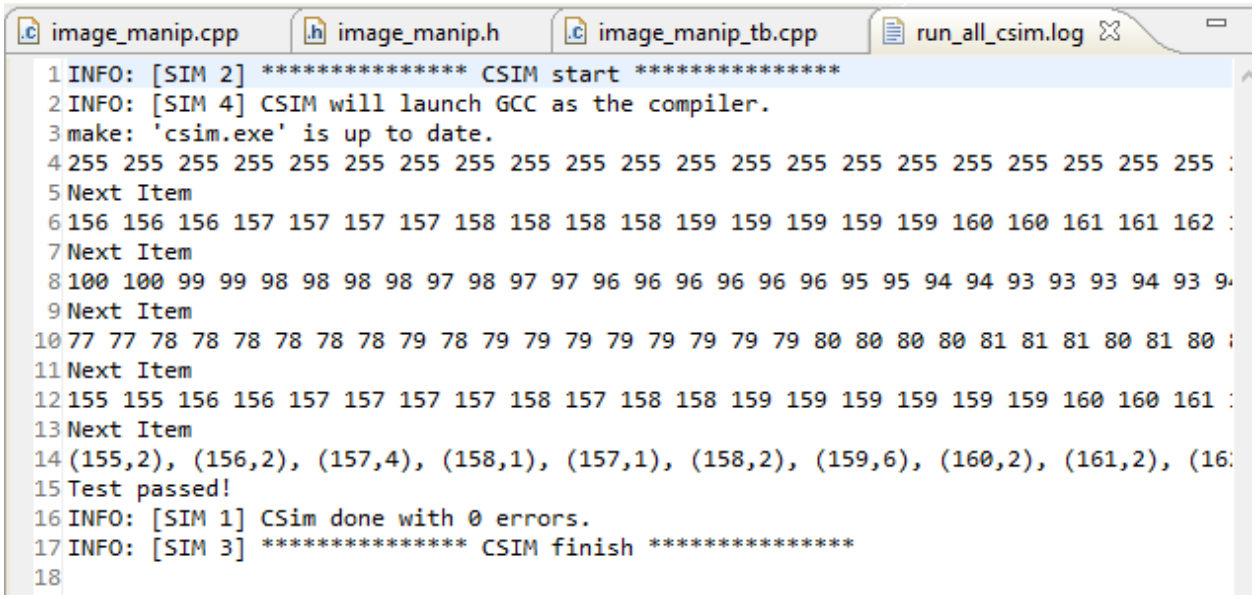
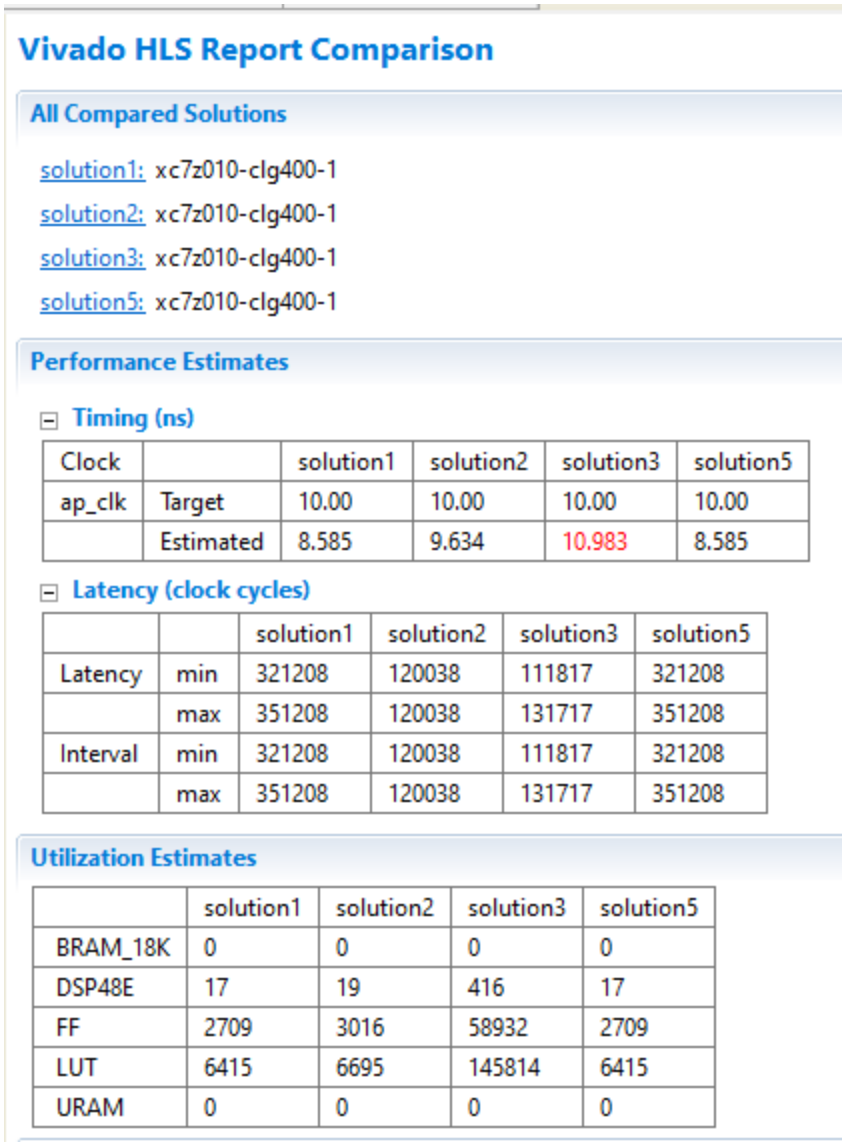


Fig. 4 C Testbench Results



Vivado HLS Report Comparison

All Compared Solutions

solution1: xc7z010-clg400-1

solution2: xc7z010-clg400-1

solution3: xc7z010-clg400-1

solution5: xc7z010-clg400-1

Performance Estimates

Timing (ns)

Clock		solution1	solution2	solution3	solution5
ap_clk	Target	10.00	10.00	10.00	10.00
	Estimated	8.585	9.634	10.983	8.585

Latency (clock cycles)

		solution1	solution2	solution3	solution5
Latency	min	321208	120038	111817	321208
	max	351208	120038	131717	351208
Interval	min	321208	120038	111817	321208
	max	351208	120038	131717	351208

Utilization Estimates

	solution1	solution2	solution3	solution5
BRAM_18K	0	0	0	0
DSP48E	17	19	416	17
FF	2709	3016	58932	2709
LUT	6415	6695	145814	6415
URAM	0	0	0	0

Fig. 5 Solution Comparison (Solution 4 Could not Synthesize due to load and write errors)

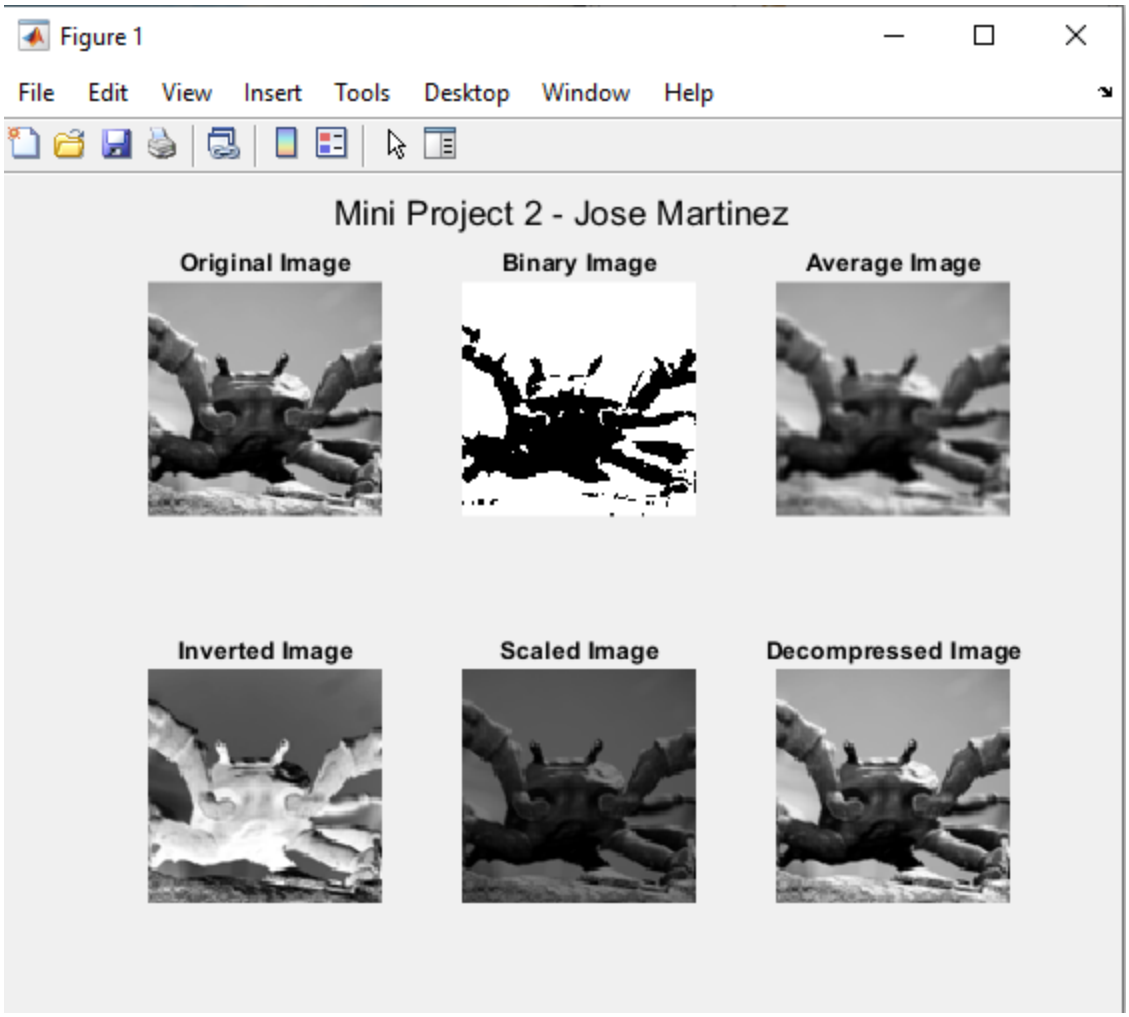


Fig. 6 Matlab Output