Homework 7: SOC System Integration

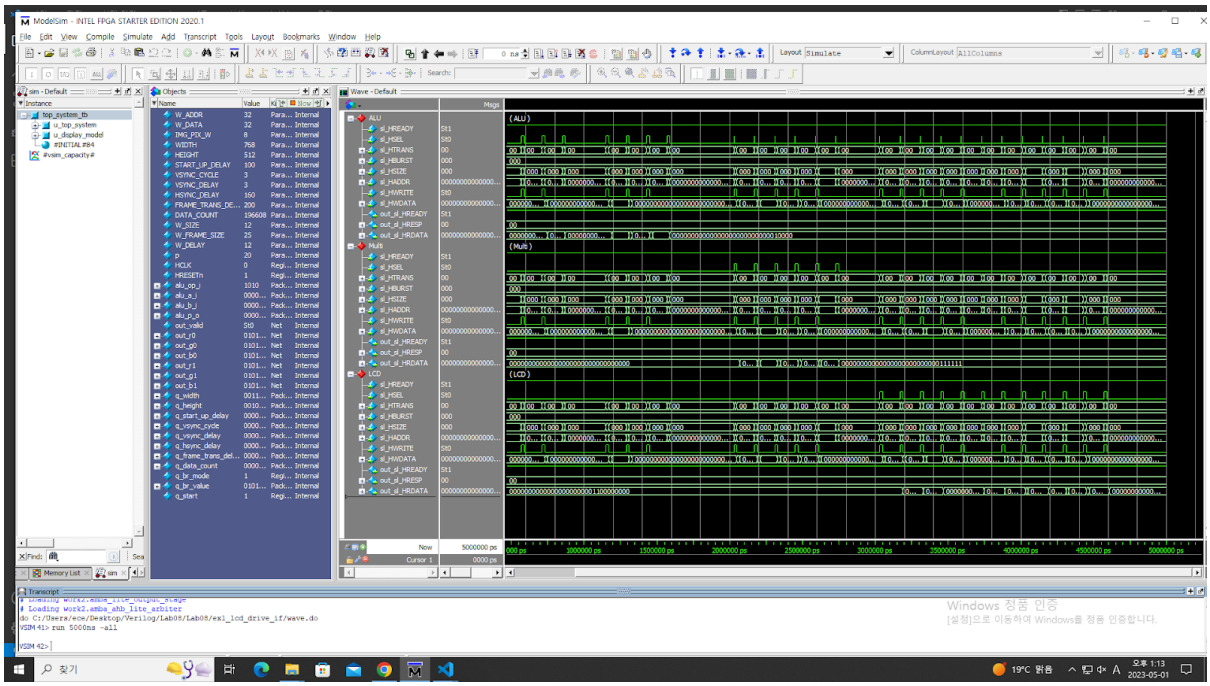
SATYAM (2023-81784)

****

Problem 1 (10p): LCD Drive Interface

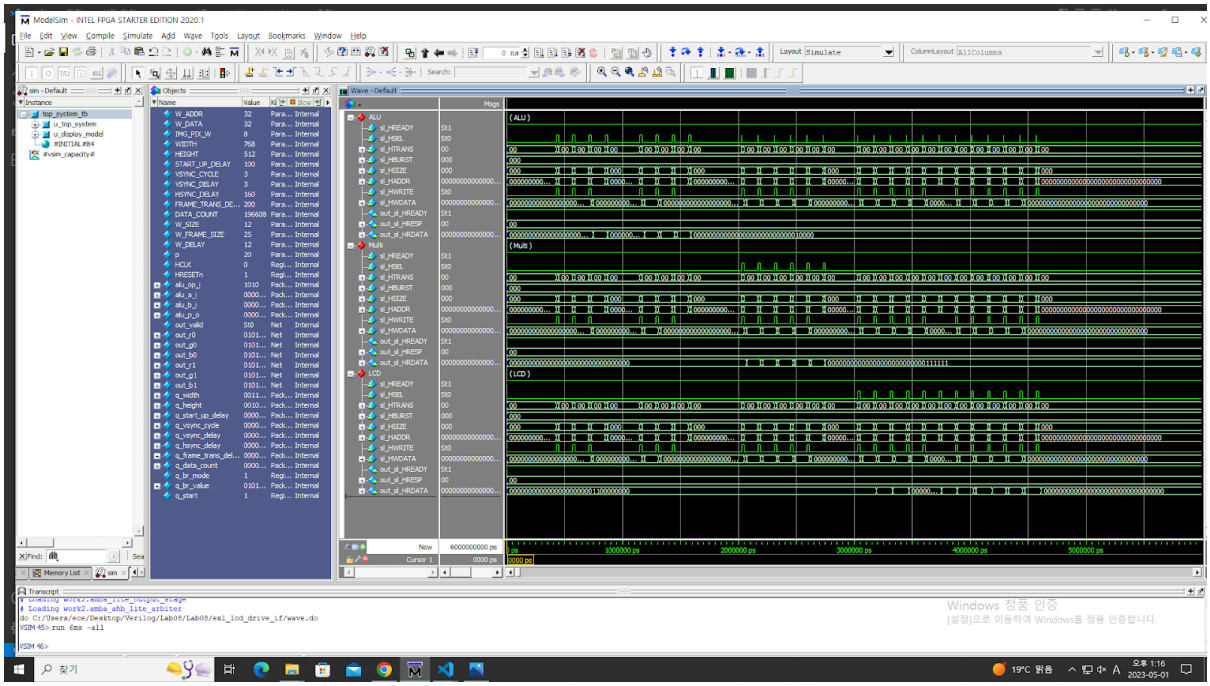
a. Create a custom IP (lcd_drive_if.v) and add it to the top system

- Simulation



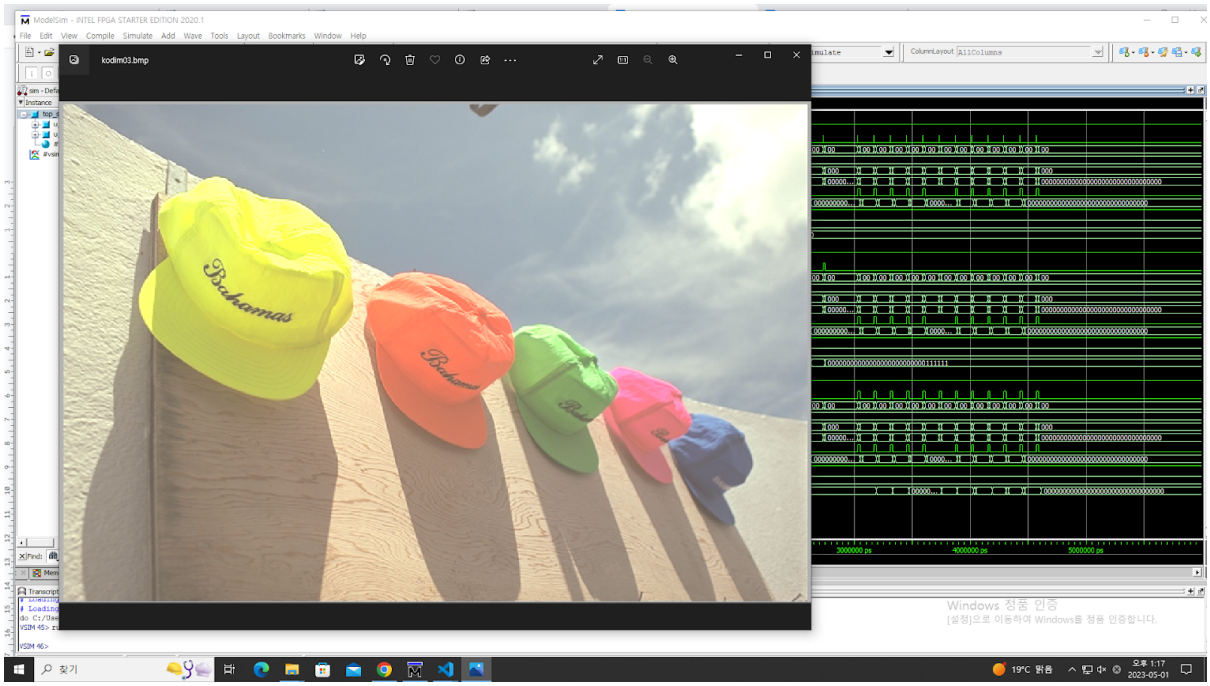
b. Generate data signals (lcd_drive_if.v)

- Simulation

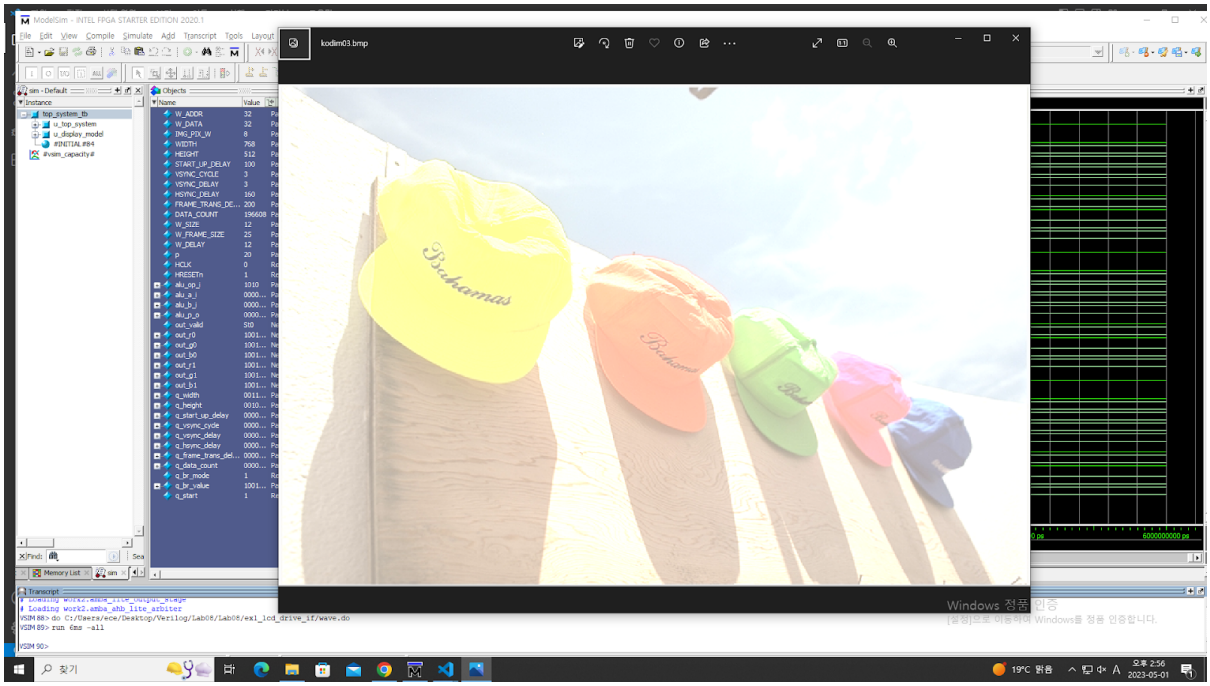


- Images

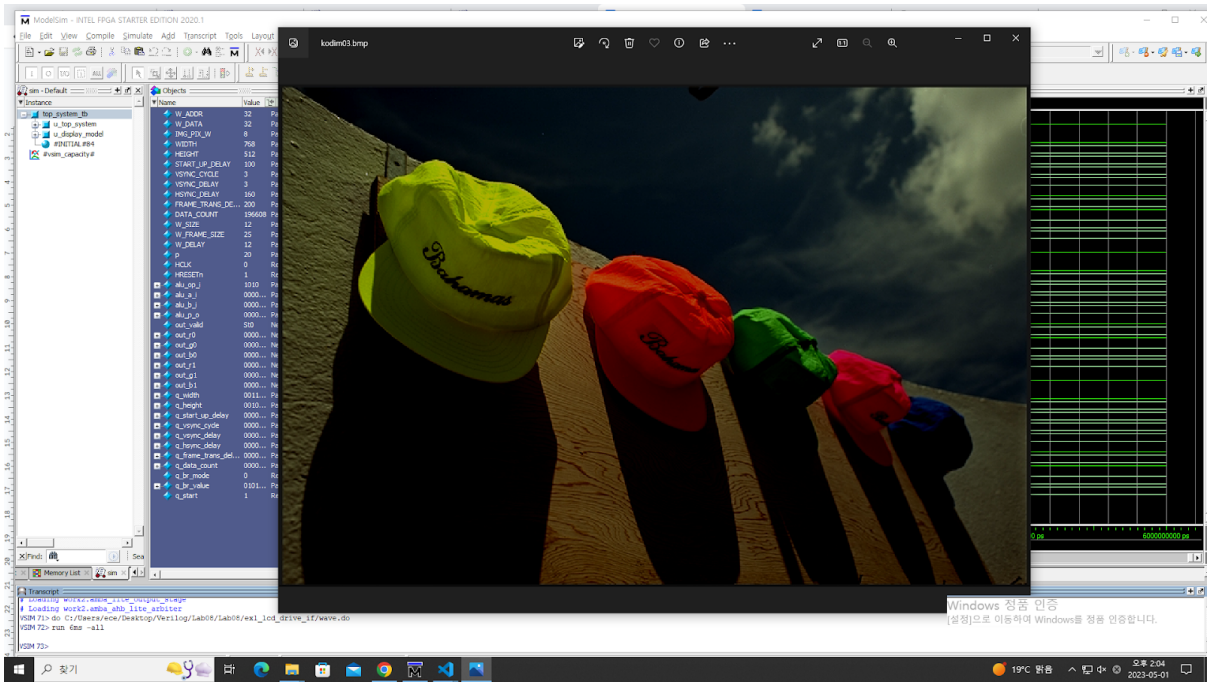
1. Brightness, 50



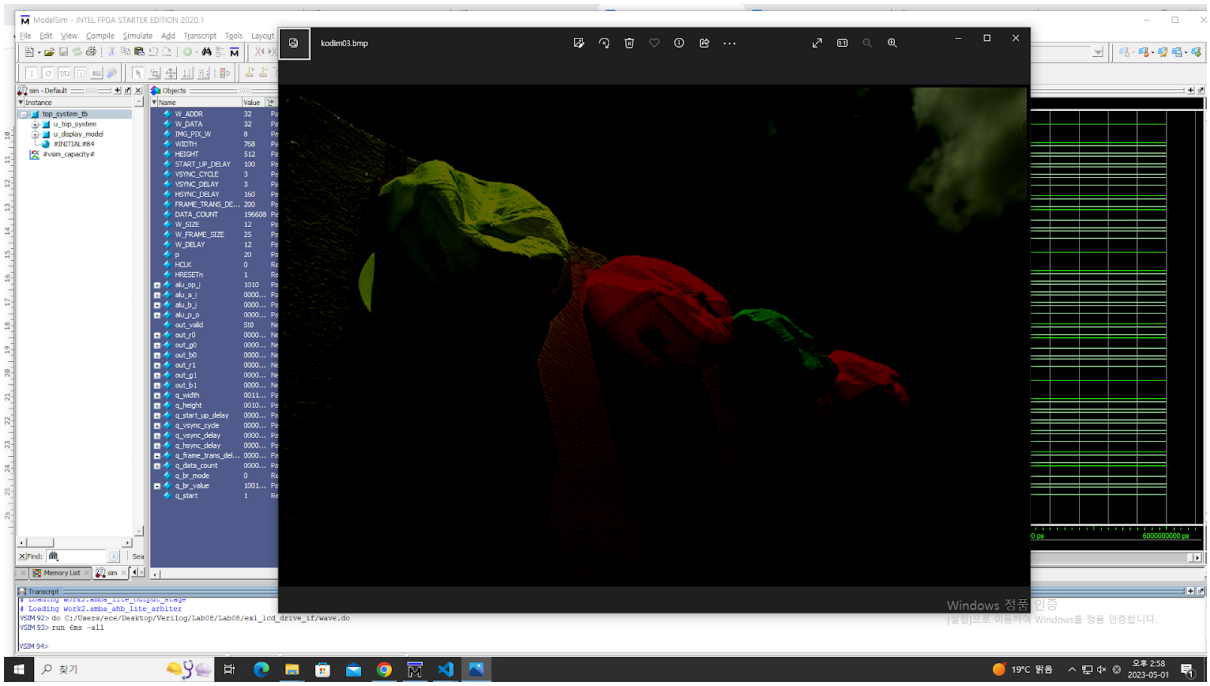
2. Brightness, 100



3. Darker, 50



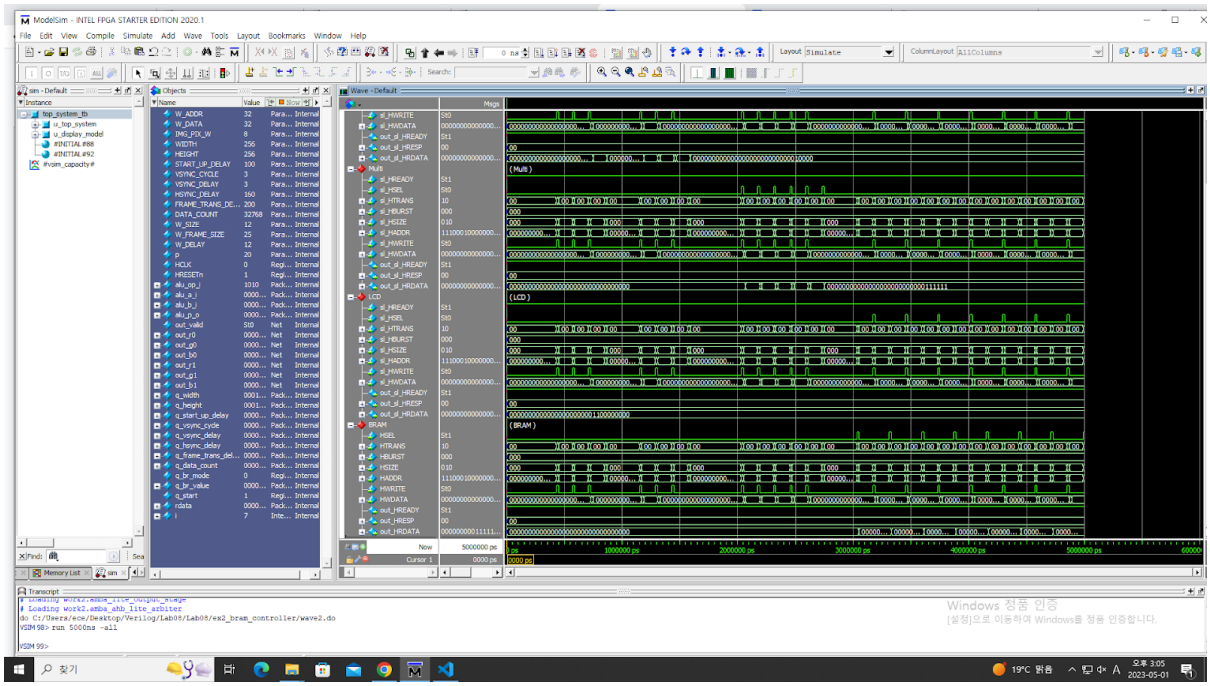
4. Darker, 100



Problem 2 (10p): BRAM Controller

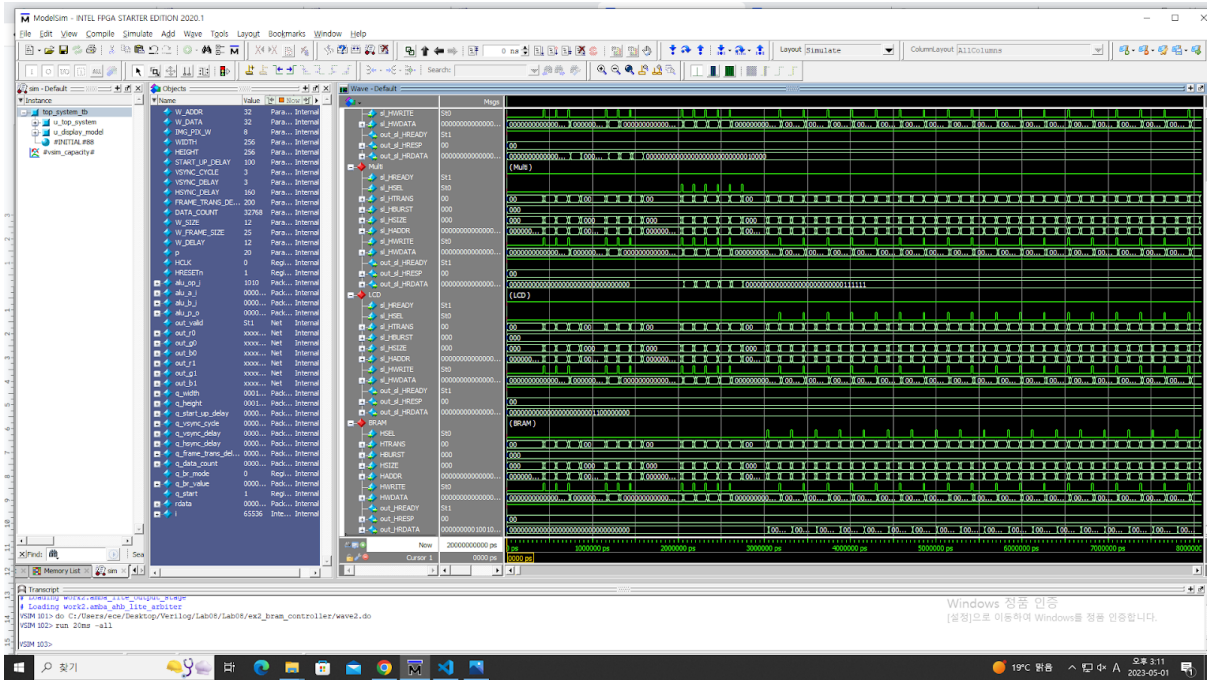
a. Bus interface and IP integration (Similar to Problem 1)

- Simulation



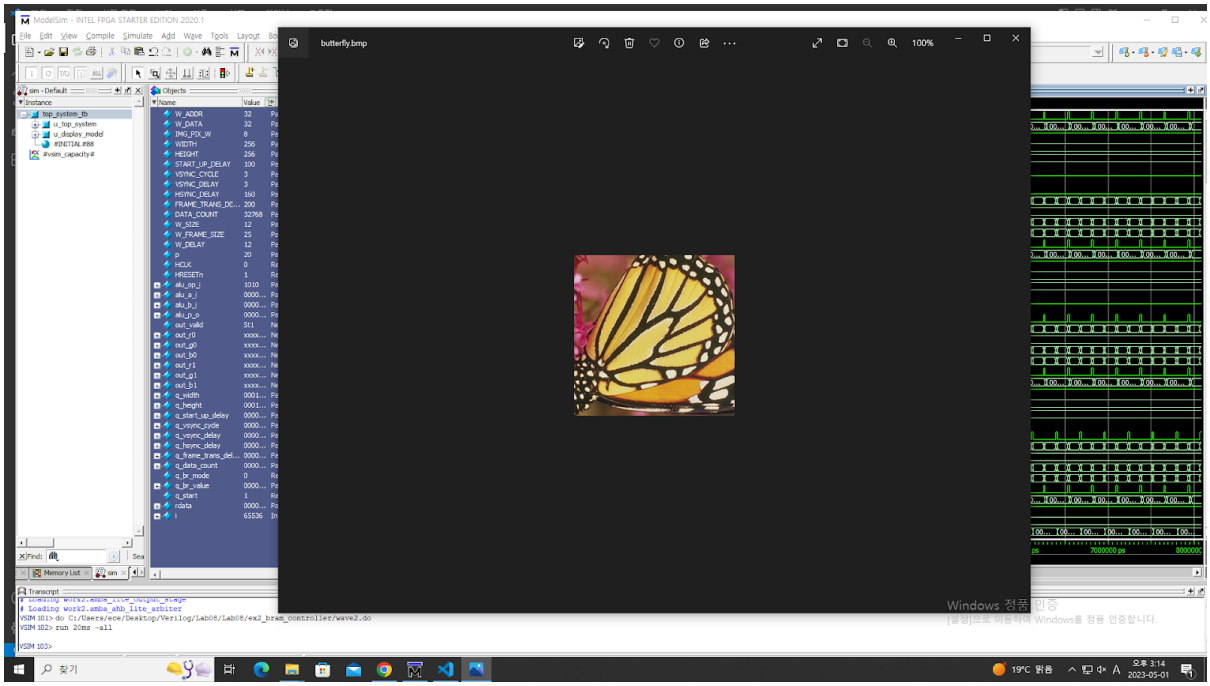
b. Generate data signals (lcd_drive_if.v)

- Simulation

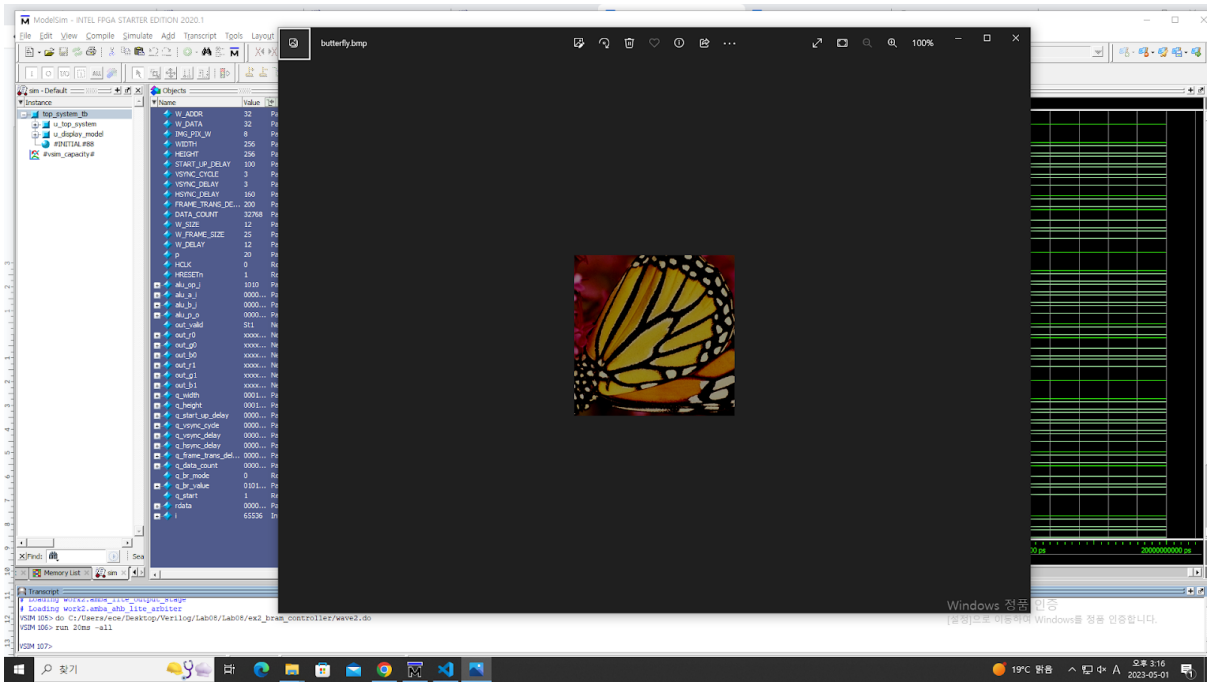


- Images

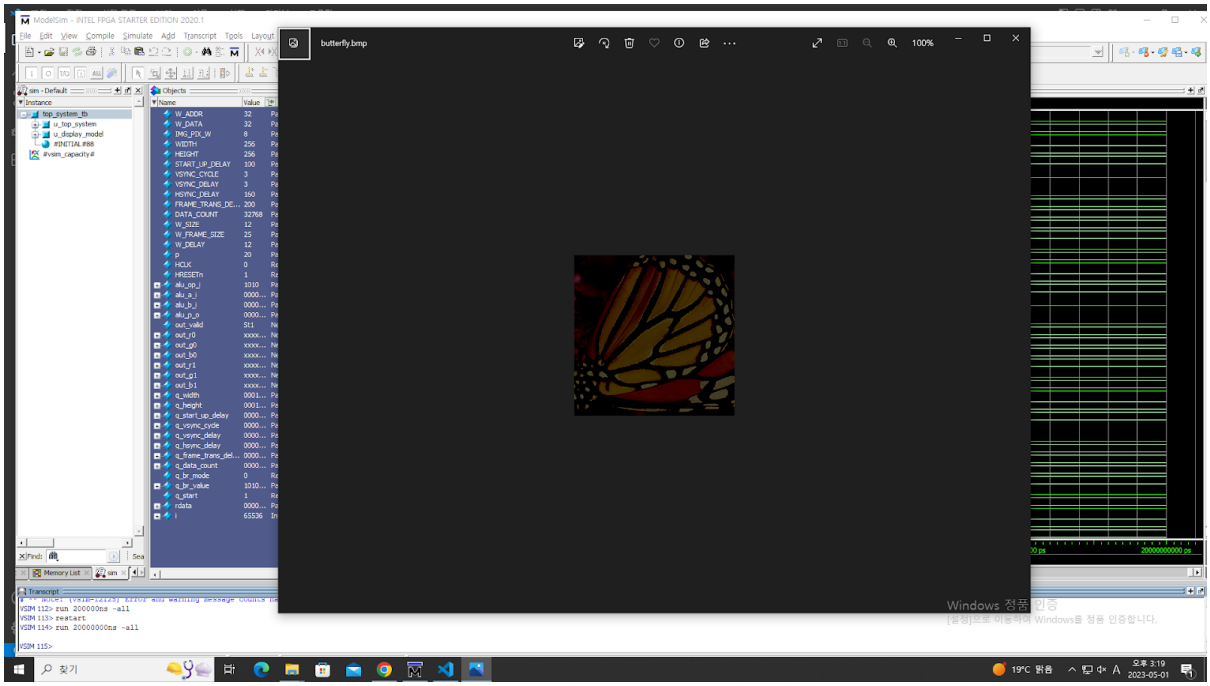
1. Regular



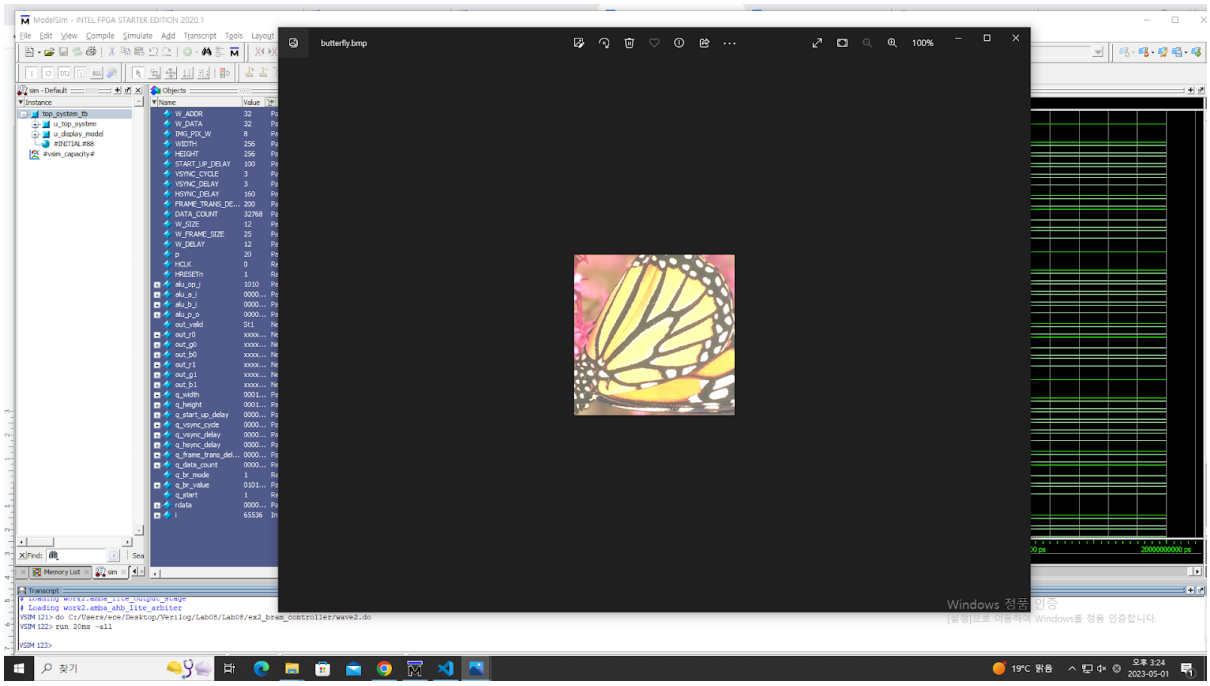
2. Darker, 50



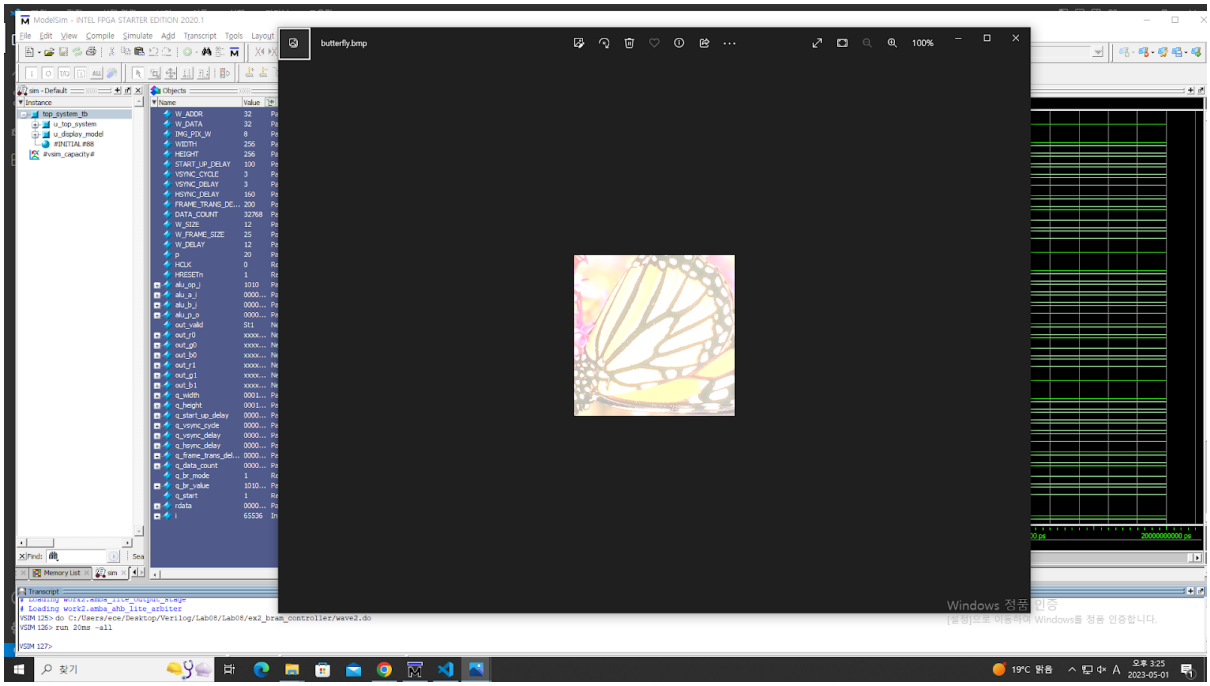
3. Darker, 100



4. Brightness, 50



5. Brightness, 100



c. Frame buffer

- **Explain the input and output signals of lcd_frame_buffer, i.e. size, meaning.**

Inputs

- en: Enable Signal
- we: Write enable (Read: 0, Write: 1)
- din[23:0]: R, G, B Pixels, where only data for one pixel (24 Bits) because only 32 bits can be transferred from memory in one cycle.
- addr: Request address
- addr_dual_pixel: address to access two pixels per cycle

Outputs

- dout[23:0]: for storing and transferring one pixel data (24 bits)
- dout_dual_pixel: for storing and transferring two pixel data (48 bits)

- **What is the size of the frame buffer?**

This depends on the size of the image so,

Size of the frame buffer = Width x Height x 24 bits

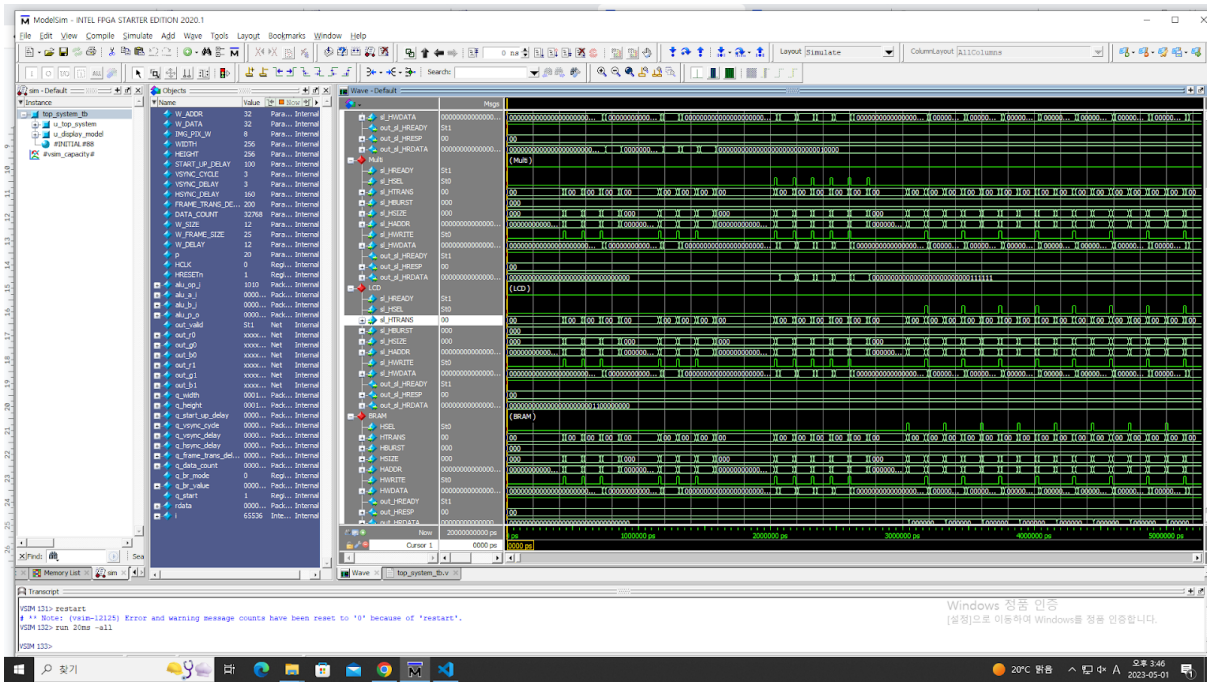
Because “Width x Height” of the image gives us the number of pixels. And one pixel have 24 bits compromising of 8 bits for R, G, B pixels individually.

Problem 3 (10p): Compressed frame buffer

a. Bus interface and IP integration (Similar to Problems 1 and 2)

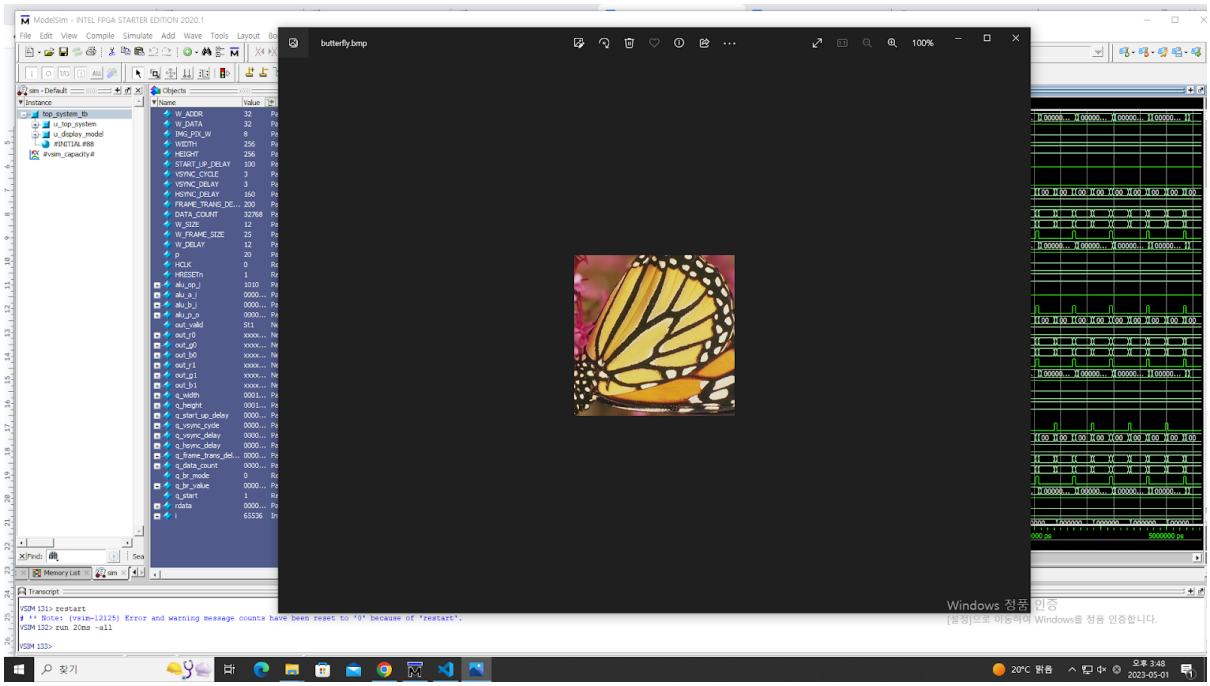
b. Generate data signals (lcd_drive_if.v)

- Simulation

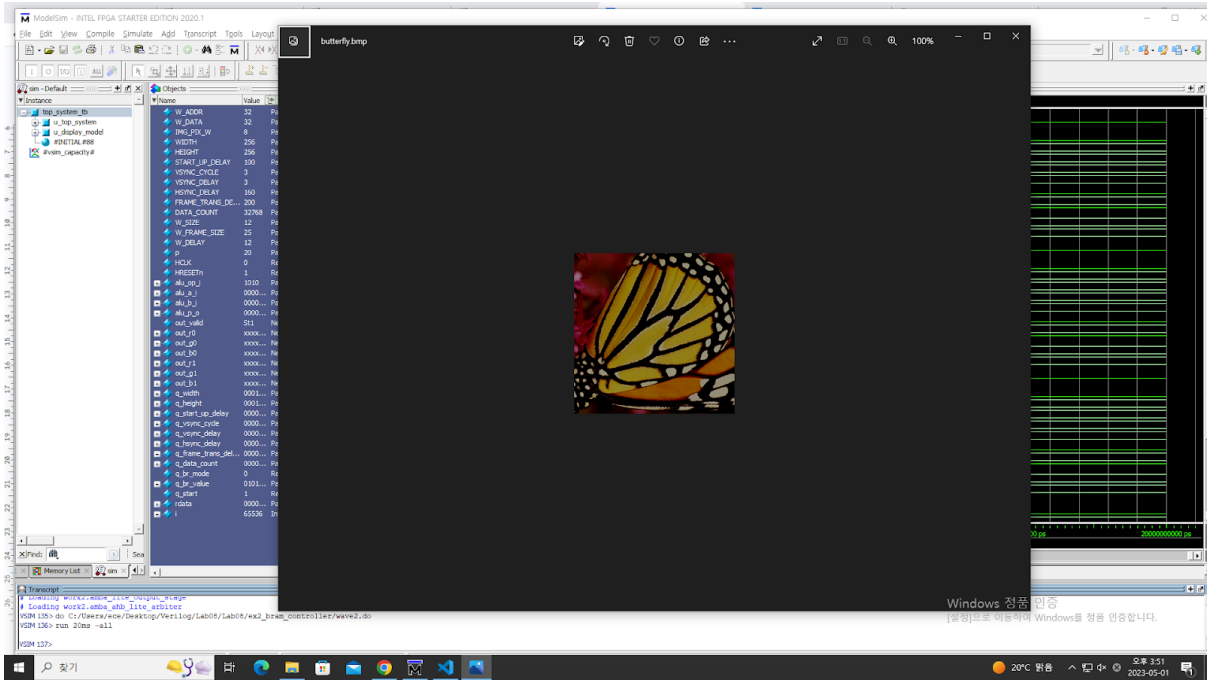


- Images

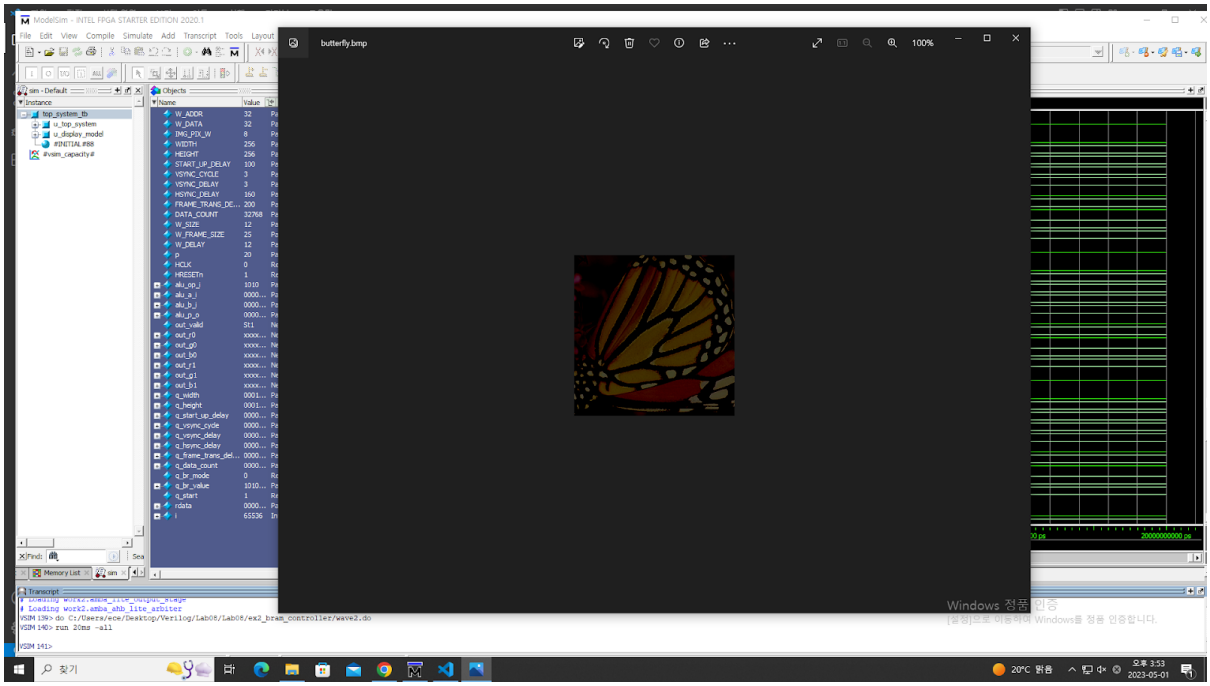
1. Regular



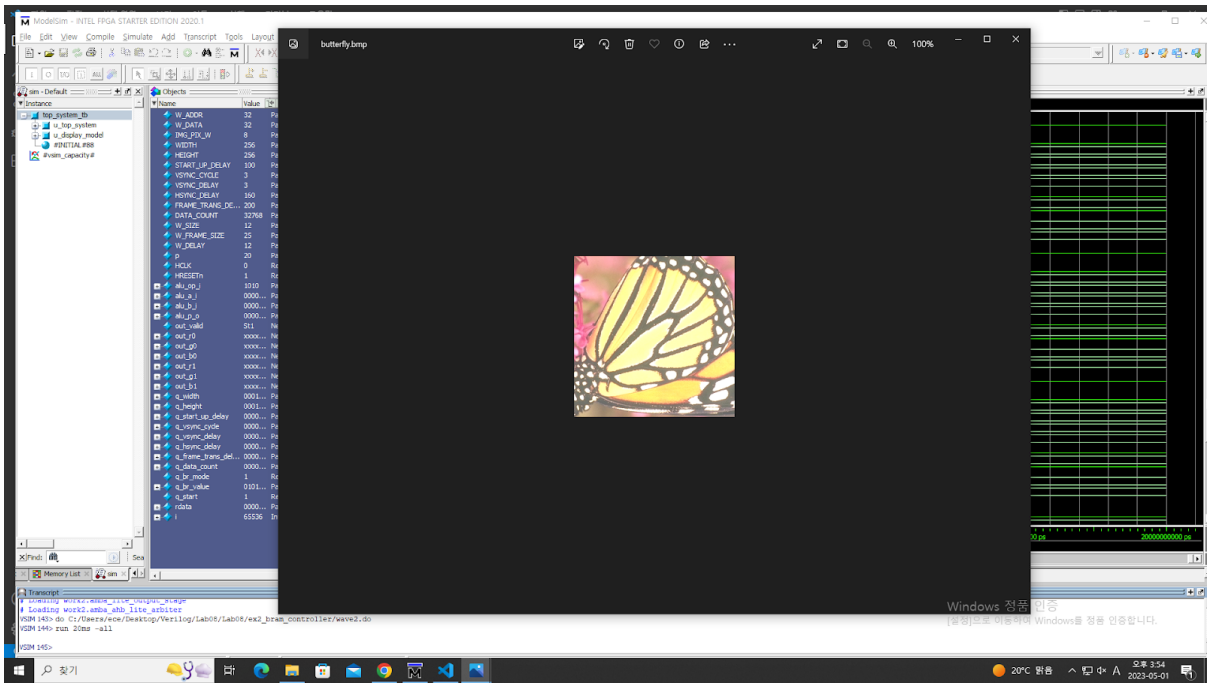
2. Darker, 50


butterfly.bmp
100%
Windows 정품 인증

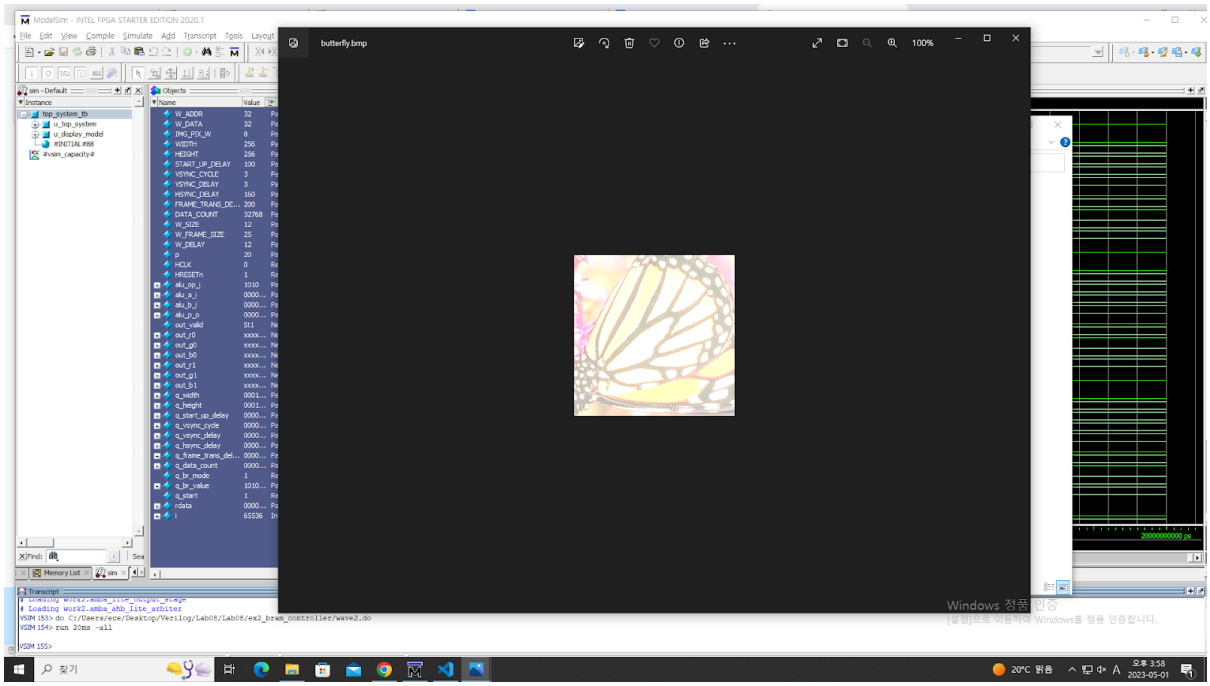
3. Darker, 100



4. Brightness, 50



5. Brightness, 100



c. Frame buffer

- **Explain the input and output signals of lcd_frame_buffer_opt, i.e. size, meaning.**

Inputs

- en: Enable Signal
- we: Write enable
- din[39:0]: to get Y0, Y1, Cb, Cr Pixels (10 bits each) after converting two pixels of RGB from memory (two pixels got from two cycles)
- addr: Request Address
- addr_dual_pixel: address to access two pixels per cycle

Outputs

- dout[39:0]: Because we use 40 Bits of data to define two pixels of the image.

- **What is the size of the frame buffer?**

This depends on the size of the image so,

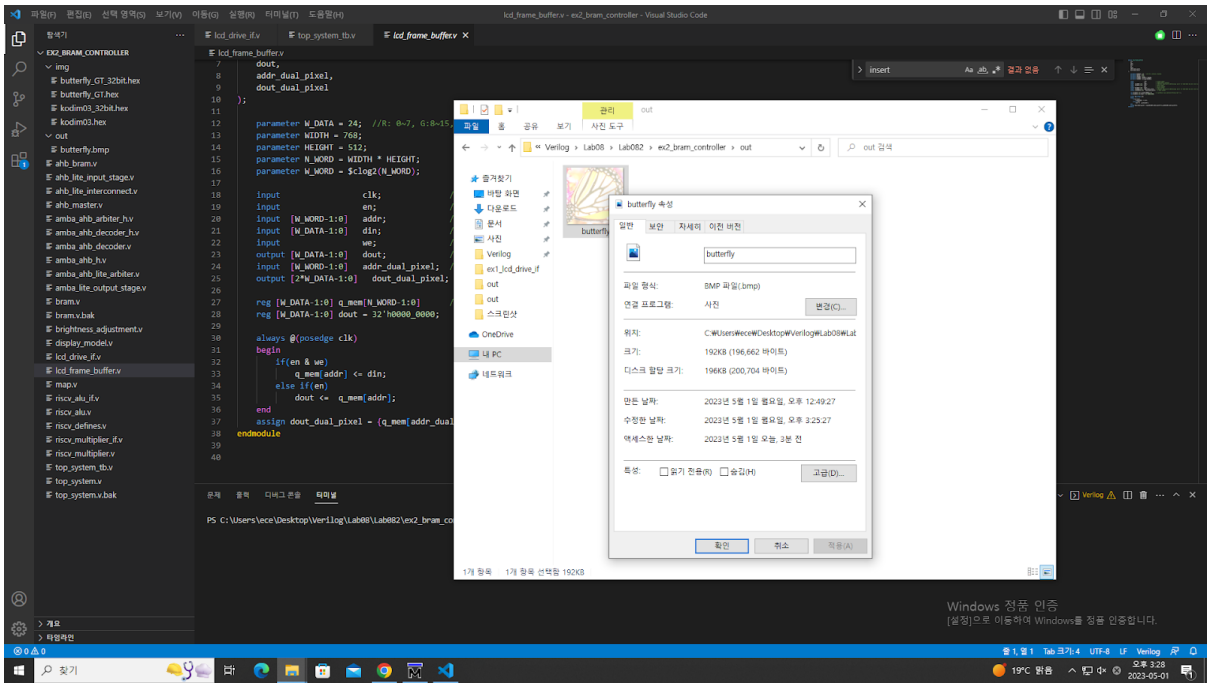
Size of the frame buffer = Width x Height x 20 bits

Because “Width x Height” of the image gives us the number of pixels. And two pixels have 40 bits compromising of 10 bits for Y0, Y1, Cb, Cr pixels individually. So, basically we have 20 bits of data for each pixel.

- **Compare the original frame buffer sizes in Problem 2 and the compressed buffer sizes in Problem 3.**

Comparing frame buffer size in Problem 2 and problem 3, frame buffer in Problem 3 have lesser size for the same image and hence useful for storing compressed image. Because for 2 pixels its storing 40 bits compared to 48 bits for two pixels in problem 2. Also, it won't affect the output image in both case as seen below from problem 2 and 3.

Problem 2's Output Image ->



Problem 3's Output Image ->