Python Image Conversion Script Tutorial

Converting an Image for Use in SystemVerilog

The provided python code *image_to_verilog.py* will allow you to convert an image to black-and-white data (0/1) for use in a RAM or ROM. This is handy for importing pixel art for things like start screens, game over screens, sprites/characters, or other objects.

The script takes 1–3 arguments. The script also has a help message (`-h` option) so you don't have to refer here all the time. Basic usage:

```
python3 image_to_verilog.py [-i] [-a ARRAY] image
```

- `-i` or `--invert` — Optional argument that flips the default 0 = black, 1 = white to 0 = white, 1 = black.
- `-a` or `--array` — Optional argument that changes the output from MIF format to a SystemVerilog array using the provided array name `ARRAY`.
- `image` — Required path to image to use. It is easiest to place the image in the same directory as this script so you can just use the image name directly here.

Notes:

- If you don't have python installed on your computer, you can run the script/code in a Jupyter Notebook, using something such as Anaconda or Google's Colab.
- The script requires the use of the python Pillow library: https://pillow.readthedocs.io/en/stable/installation.html
- The script outputs to the terminal. If you are using the MIF output, then use a pipe to save the output to a file (see the Example on the next page) and follow the steps in Lab 2 to initialize your ROM/RAM with this file. If you are using the array output, then copy-and-paste the output directly into a SystemVerilog code file.
- This script has been tested on a limited number of PNG and JPG files. This does not guarantee proper behavior on all files, though the heavy lifting is done by the Pillow library so it *should* work on most image files. Just be very careful if you're using a colored original image, as the output may not look the way you want it to.
- The size of the output matches the input image exactly. You will likely want to be drawing something that's low resolution or meant to take up a small number of pixels on the screen, so you may need to do some cropping or resizing on your image before running it through the script.

Example Usage

This example assumes that the following image is stored as *game_over.png* in the same directory as *image_to_verilog.py*. The image is 84 x 44 pixels:

To store the MIF version of this image into the file *game_over.mif*, run the following command:

```
python3 image_to_verilog.py game_over.png > game_over.mif
```

To get the SystemVerilog array syntax, instead run (output shown in yellow):

```
python3 image_to_verilog.py -a dead game_over.png
dead[ 0] = 84'b000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000;
dead[ 1] = 84'b000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000;
dead[ 2] = 84'b000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000;
dead[ 3] = 84'b000000001111111111111000000001111110000000001111000000001111000111111111111111111000;
dead[ 4] = 84'b000000001111111111111000000001111111100000001111100000001111000111111111111111111100;
dead[ 5] = 84'b000000001111111111111000000001111111100000001111100000001111000111111111111111111000;
dead[ 6] = 84'b000001111100000000000000011111100111110000011111111000111111100011111100000000000000;
dead[ 7] = 84'b000001111100000000000000011111100111110000011111111001111111100011111100000000000000;
dead[ 8] = 84'b000101111100000000000000011111100011111000011111111001111111100011111100000000000000;
dead[ 9] = 84'b001111110000000000000001111100000001111100111111111111111111000111110000000000000000;
dead[10] = 84'b000111110000000000000001111100000001111100111111111111111111000111110000000000000000;
dead[11] = 84'b000111110000011111110001111100000001111100111111111111111111000111111111111111100000;
dead[12] = 84'b000111110000011111110001111100000001111100111111111111111111000111111111111111100000;
dead[13] = 84'b000111110000011111110001111100000001111100111111111111111111000111111111111111100000;
dead[14] = 84'b001111110000000111110001111111111111111100111111000111001111100011111000000000000000;
dead[15] = 84'b001111110000000111110001111111111111111100111111000111001111100011111000000000000000;
dead[16] = 84'b000001111100000111110001111100000001111100111111000000001111100011111000000000000000;
dead[17] = 84'b000001111100000111110001111100000001111100111111000000001111100011111000000000000000;
dead[18] = 84'b000001111100000111110001111100000001111100111111000000001111100011111000000000000000;
dead[19] = 84'b000000001111111111110001111100000001111100111111000000001111100011111111111111111100;
dead[20] = 84'b000000001111111111110001111100000001111100111111000000001111100011111111111111111100;
dead[21] = 84'b000000000000000000000000000000000000000000000000000000000000000000000000000000000000;
dead[22] = 84'b000000000000000000000000000000000000000000000000000000000000000000000000000000000000;
dead[23] = 84'b000000000000000000000000000000000000000000000000000000000000000000000000000000000000;
dead[24] = 84'b000001111111111111100001111100000000111110011111111111111111100011111111111111100000;
dead[25] = 84'b000001111111111111100001111100000000111110011111111111111111100011111111111111100000;
dead[26] = 84'b000111111000000011101000111110000000011111001111110000000000110001111100000001111000;
dead[27] = 84'b001111110000000111110001111100000000011111001111100000000000000001111100000001111100;
dead[28] = 84'b000111110000000111110001111100000000011111001111100000000000000001111100000001111000;
dead[29] = 84'b000111110000000111110001111100000000011111001111100000000000000001111100000001111000;
dead[30] = 84'b000111110000000111110001111100000000011111001111100000000000000001111100000001111000;
dead[31] = 84'b000111110000000111110001111111000001111110011111110000000010000001111100000101111000;
dead[32] = 84'b000111110000000111110001111111100111111110011111111111111111100000111110000011111100;
dead[33] = 84'b000111110000000111110001111111100111111110011111111111111111100000111110000011111100;
dead[34] = 84'b000111110000000111110000111111111111111000001111100000000000000001111111111111000000;
dead[35] = 84'b000111110000000111110000111111111111111000001111100000000000000001111111111111000000;
dead[36] = 84'b000111110000000111110000011111111111111000001111100000000000000001111111111111000000;
dead[37] = 84'b001111110000000111110000000011111111100000001111100000000000000001111100111111100000;
dead[38] = 84'b001111110000000111110000000011111111100000001111100000000000000001111100111111100000;
dead[39] = 84'b000001111111111111100000000000011100000000001111111111111111111000111110000011111000;
dead[40] = 84'b000001111111111111100000000000011100000000001111111111111111111000111110000011111100;
dead[41] = 84'b000001111111111111100000000000011000000000001111111111111111111000111110000011111000;
dead[42] = 84'b000000000000000000000000000000000000000000000000000000000000000000000000000000000000;
dead[43] = 84'b000000000000000000000000000000000000000000000000000000000000000000000000000000000000;
```

You should be able to visibly discern the “GAME OVER” letters in the output. Note that 1 = white here, which would likely be the pixels you would want to draw on the VGA. If it’s the opposite, *i.e.*, you want to draw where the original image was black, then just add the invert flag.