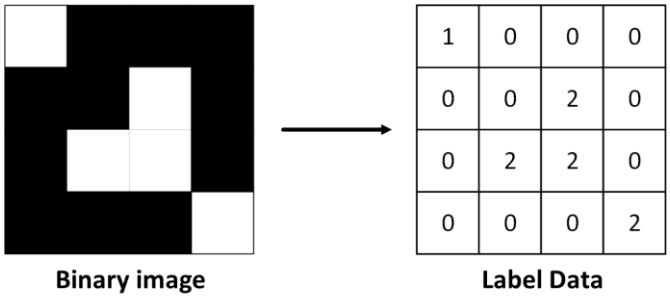
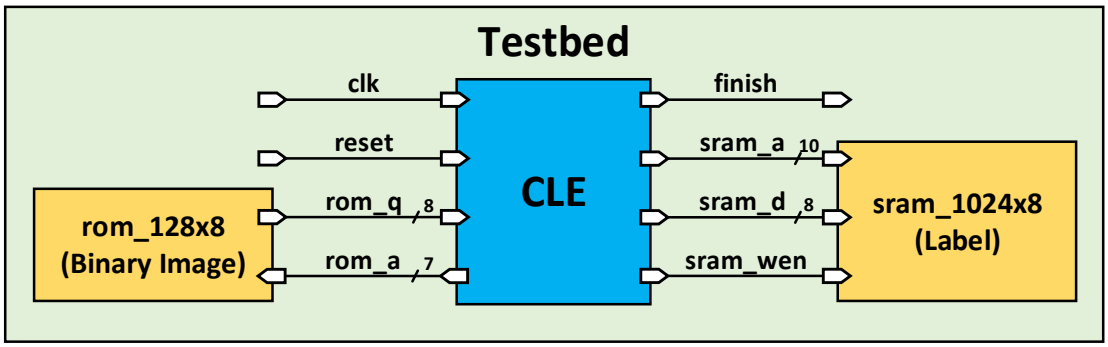
Introduction

In this final project, you are asked to design a **Component Labeling Engine (CLE)**, which can detect object segmentation from the binary image, and give the same ID number to the same object.



Block Diagram



Specifications

1. Top module name: **CLE**
2. Input/output description:

Signal Name	I/O	Width	Simple Description
clk	I	1	Clock signal in the system. All inputs are synchronized with the **positive** edge clock. All outputs should be synchronized at clock **rising** edge
reset	I	1	Active **high** asynchronous reset.
rom_a	O	7	Address for binary image memory.
rom_q	I	8	Binary data from binary image memory.
sram_a	O	10	Address for storing to label memory.
sram_d	O	8	Label data for storing to label memory.
sram_wen	O	1	Set **low** if CLE needs to write data to label memory. (reading mode is not supported in this design)
finish	O	1	Set **high** if finish Labeling, and testbed will check the correctness of label data in sram_1024x8.

Design Description

1. The input image is a 32x32 binary image as shown in **Fig.1**. For the binary signal, 0 represents the background, and 1 represents the object for each pixel. You have to check if those pixels with value 1 are connected or not. The connected pixels represent to the same object. Those pixels are given with the same label ID from the same object. The number of label ID can be created by yourself.

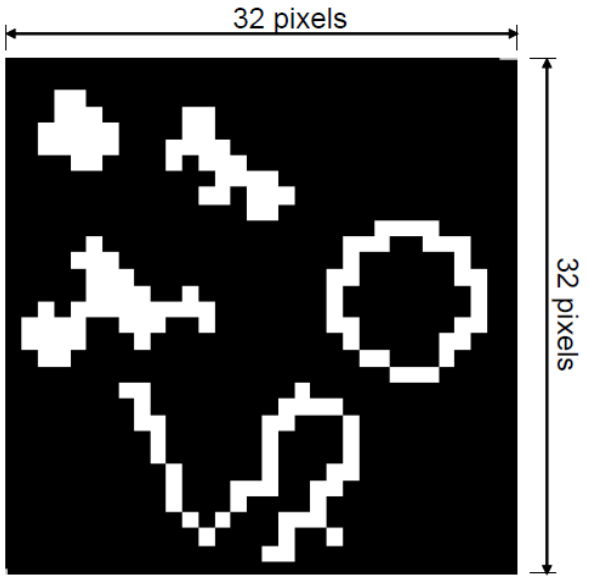


Fig.1. Binary image.

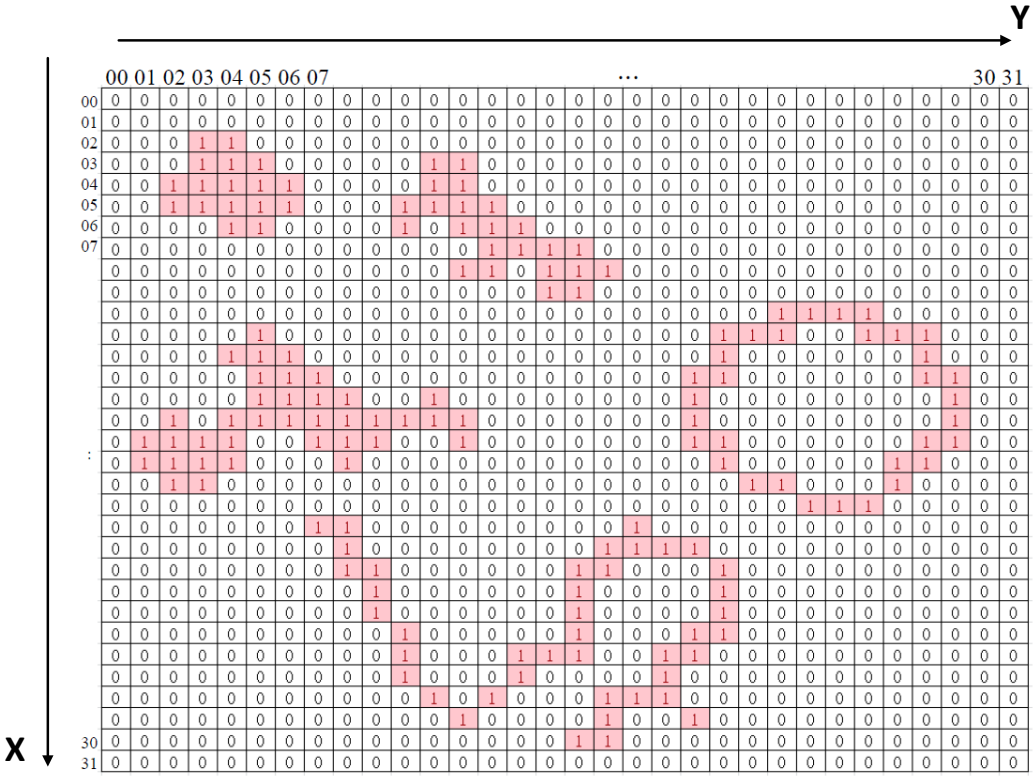


Fig.2. Actual value for each pixel in binary image.

2. The image is already stored in the 128x8 ROM. The storing order is shown in **Fig.3**. For example, if the address value is "0", the corresponding 8-bit binary data represents the pixels [X=00, Y=00-07] in **Fig.2**. The MSB is related to [X=00, Y=00], and the LSB is related to [X=00, Y=07]. The number of times to read data from ROM is not constrained, and the signal CEN from ROM is always set to 0.

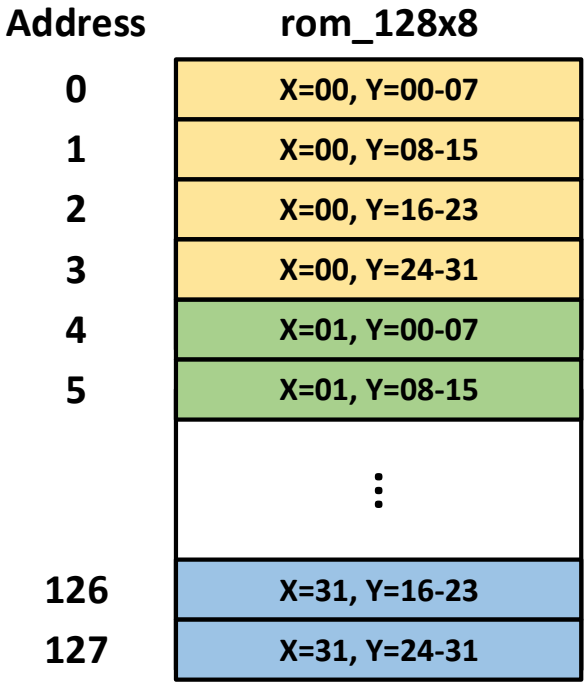


Fig.3. The storing order for the binary image in ROM.

3. The operations in CLE are shown below:
   A. First, CLE has to find the pixel equal to 1 in **Fig.2**, which means the object position. Then, it has to identify whether the pixel is connected to other pixels in a 3x3 block. **Fig.4**. shows the 8 connecting situations in a 3x3 block. If the pixels are connected, they are seen as the same object. Otherwise, the pixels which are not connected are seen as different object.

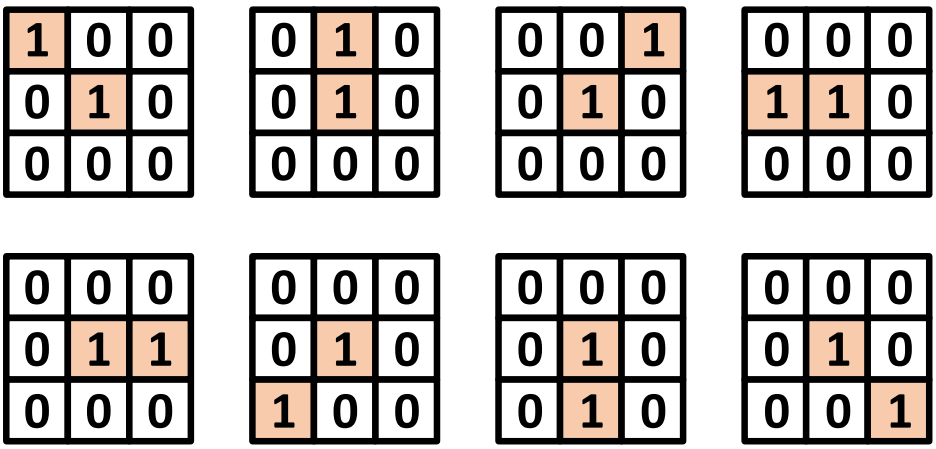


Fig.4. The 8 connecting situations in a 3x3 block.

B. Every pixel is required to be set with the same label ID for the same object. You can decide the number of label ID by yourself with the following naming rules:
   a. The number of label ID range you can use: **8'h01~8'hFB**
   b. The number of label ID range you cannot use: **8'hFC~8'hFF** (These numbers have other specific meanings)
   c. The number **8'h00** is for background. You can't give this number to object.
   d. Label ID cannot be reused for different object.

4. **Fig.5** shows the example result after processing on **Fig.2**. Five label IDs are used in this example. All labels for the 32x32 image are required to be stored in the outside SRAM (the sram_1024x8 is not included in CLE). The storing method is shown in **Fig.6**. For example, the address value "0" stores the label for pixel [X=00, Y=00], and so on. If all labels are stored in sram_1024x8, CLE sets output "finish" to high, and the testbench will check the answer.

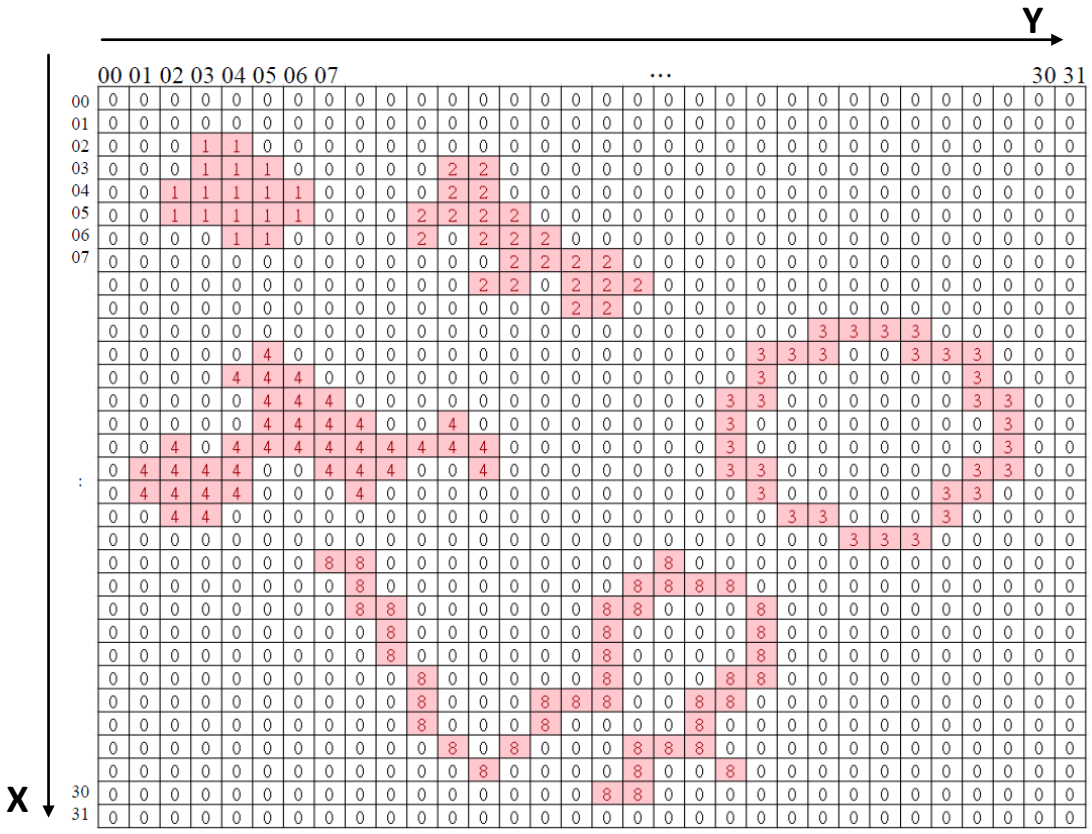


Fig.5. Labels for each pixel to store in SRAM.

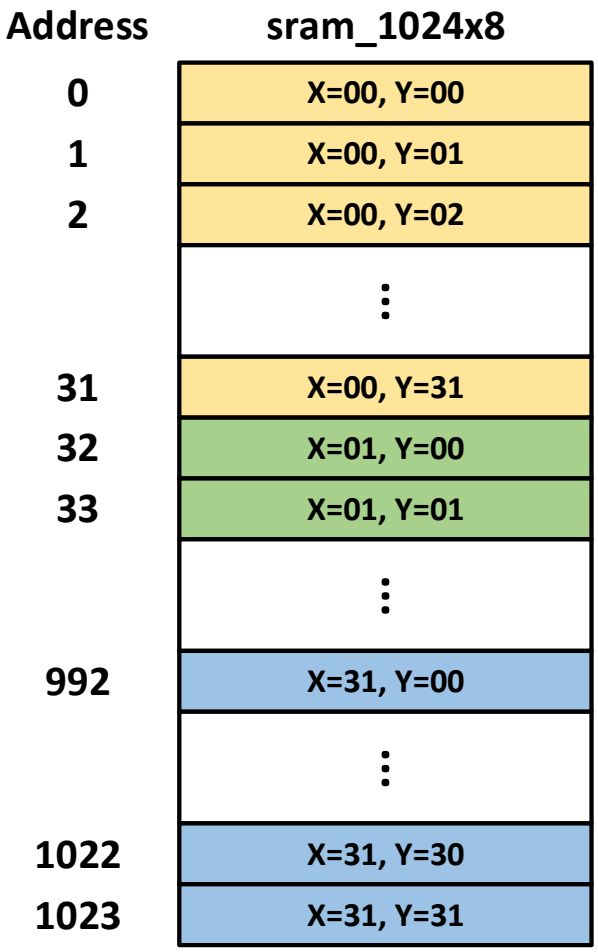


Fig.6. The storing order for the 8-bit label in SRAM.

5. The testbench will show your result as **Fig.7**, and it will also compare with the golden result. If the result of CLE is different from golden result, it will show the specific label for pixels with wrong labels as **Fig.8**. The notations for specific label are shown below:

Specific label	Description
XX	The Pixel belongs to background. The CLE identifies it to wrong ID number.
UU	The Pixel belongs to background. The CLE identifies it to Unknown.
xx	The Pixel belongs to object. The CLE identifies it to wrong ID number.
uu	The Pixel belongs to object. The CLE identifies it to Unknown.

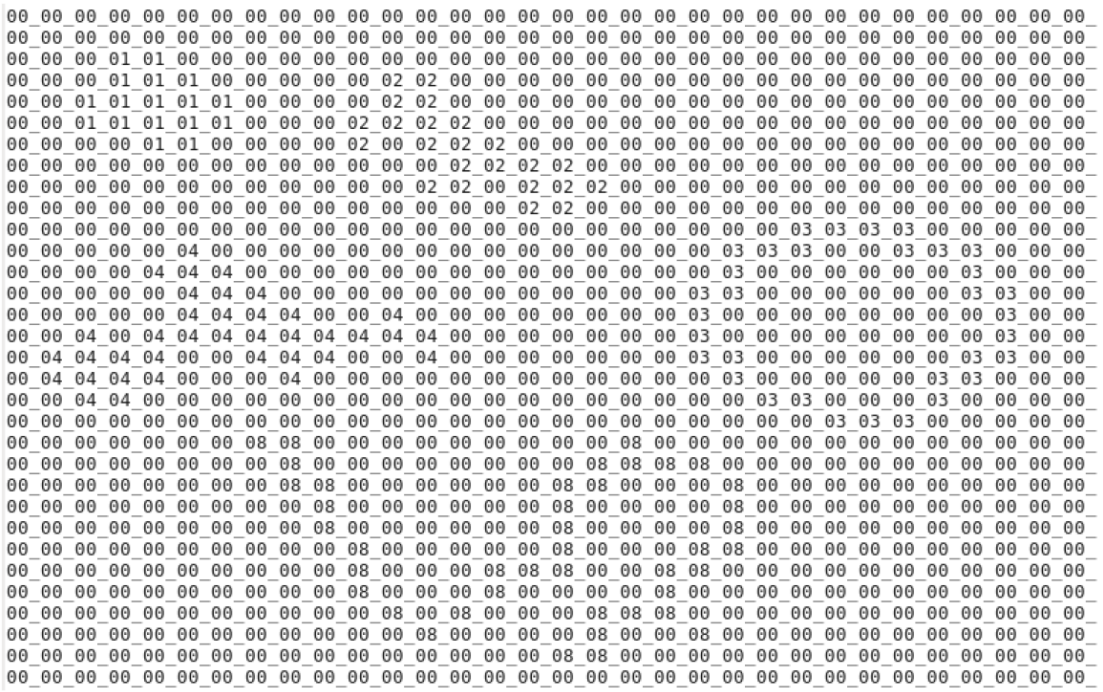


Fig.7. Log message of SRAM from function simulation.

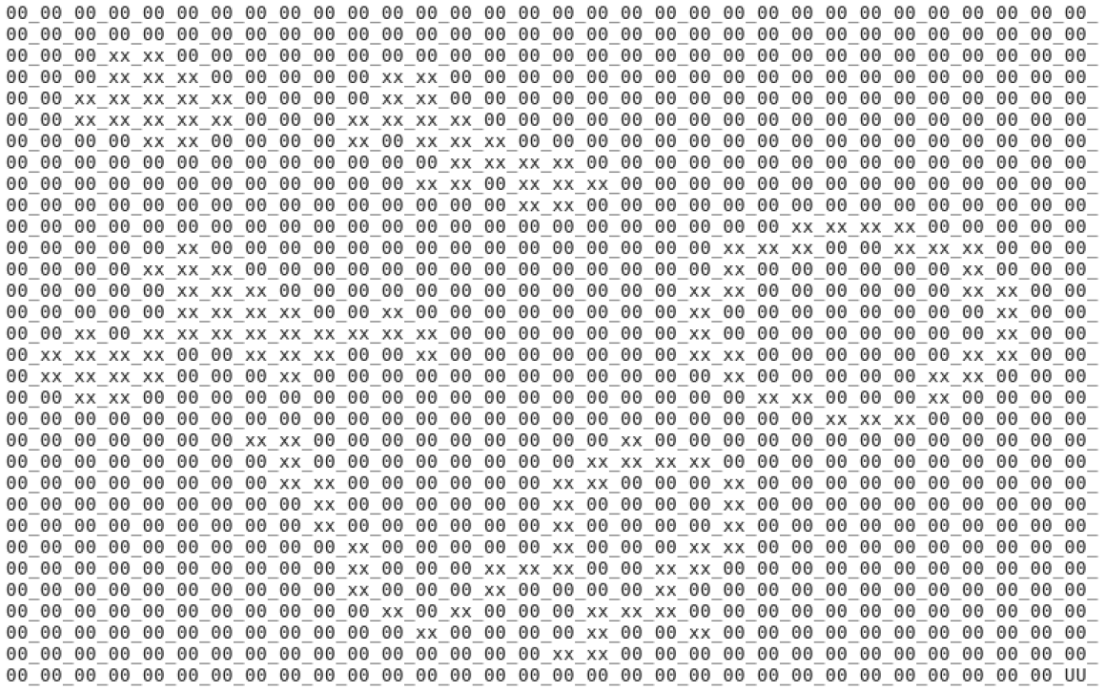


Fig.8. Log message of SRAM with specific label result from function simulation.