2021 Digital IC Design

Final Exam: Image Filter Engine

1 Introduction

We often use the different types of filters to extract features from an image or remove noise from an image or signal. In this exam, please implement an image filter engine, which uses 3x3 mean filter, 5x5 mean filter, 3x3 min filter, and threshold to process the image. The specification and function of IFE circuit will be described in detail in the following section.

2 Design Specifications

2.1 Block overview

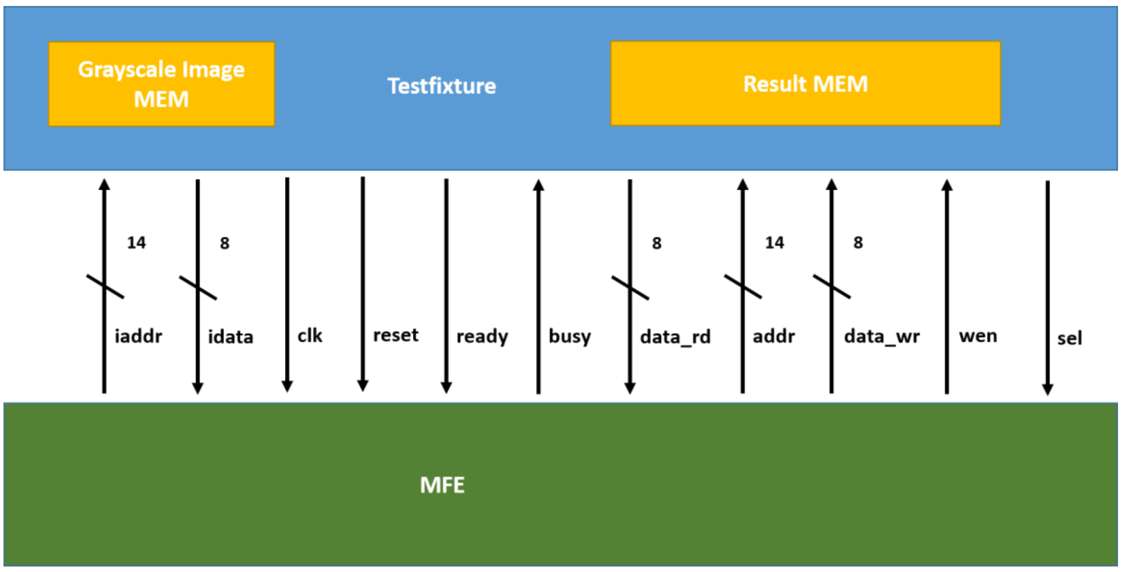

Fig. 1 – System operation flow

2.2 I/O Interface

Name	I/O	Width	Description
clk	I	1	System clock signal. This system is synchronized with the positive edge of the clock.
reset	I	1	Active-high asynchronous reset signal.
ready	I	1	Grayscale image ready indication signal. When this signal is high, the grayscale image memory is already

			prepared. IFE can start sending grayscale image address to obtain the data.
busy	O	1	System busy indication signal. When IFE receives high level ready signal and IFE is ready to start the calculation, this signal has to be set as high. After the calculation is completed and the result is completely output, set this signal to low and the testbench will start checking if the results are correct.
iaddr	O	14	The signal for grayscale image memory address, which is used to request grayscale image pixel data.
idata	I	8	Input grayscale image pixel data signal, which represents an 8 bits unsigned integer. The testbench will send the pixel data according to the address *iaddr* indicates.
data_rd	I	8	Result memory read data signal, which represents an 8 bits unsigned integer. This signal is used to send the result memory data to IFE circuit.
addr	O	14	Result memory read/write address. This signal indicates which address of the result memory will be read or written.
data_wr	O	8	Result memory write data signal, which represents an 8 bits unsigned integer. The operation result of the IFE circuit is written to result memory using this signal.
wen	O	1	Result memory write enable signal. When this signal is low (read), the data with address *addr* in result memory is sent by *data_rd*. When this signal is high (write), the data sent by data_wr is written to the address *addr* in result memory.
sel	I	2	The function selection signal. This signal will not change when an operation is being processed. If sel == 2’d0, the IFE circuit will use 3x3 mean filter to process the image. If sel == 2’d1, the IFE circuit will use 5x5 mean filter to process the image. If sel == 2’d2, the IFE circuit will use 3x3 min filter to process the image. If sel == 2’d3, the IFE circuit will use threshold to process the image.

2.3 Function Description

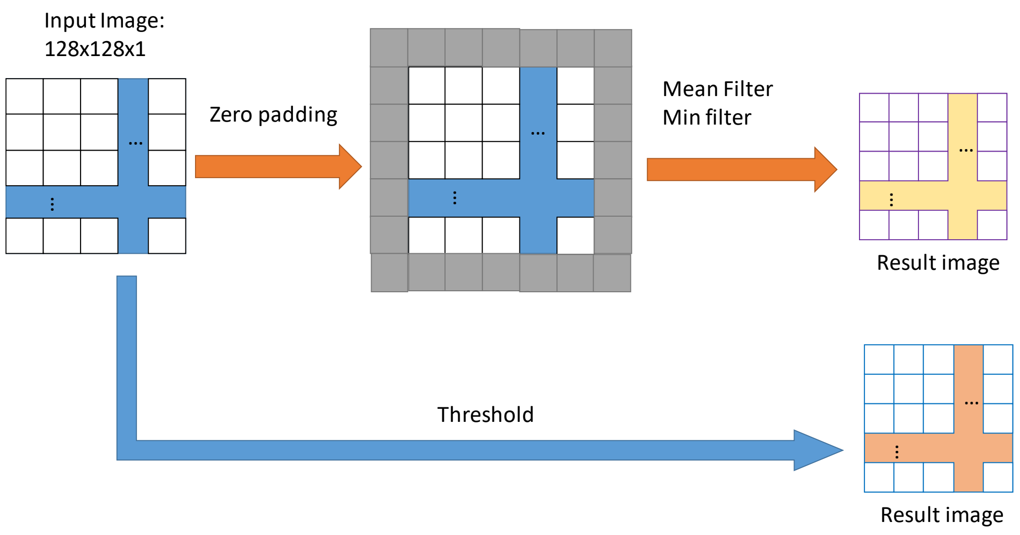

Fig. 2 – Operation flow.

The size of the input image in this system is 128x128, which is stored in the grayscale image memory. The correspondence between each pixel of the grayscale image and memory arrangement is shown in Figure 4. In the operation process, the IFE circuit uses the *iaddr* signal to send the address of requested image data (as shown in Figure 3. t1 time point). After the negative edge of each clock, the pixel data at requested address will be sent into the IFE circuit by the *idata* signal (as shown in figure 3. t2 time point).

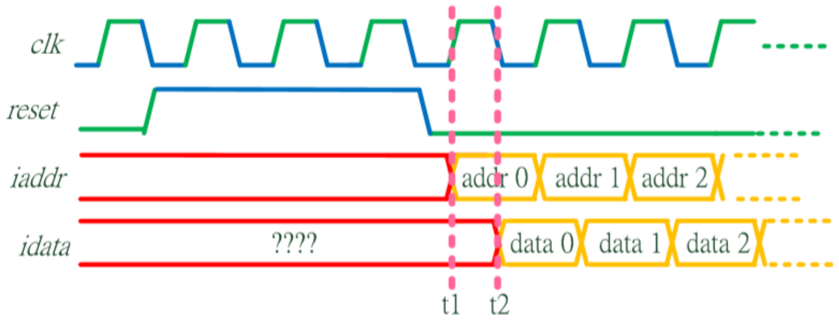

Fig. 3 – Timing diagram while reading grayscale image memory

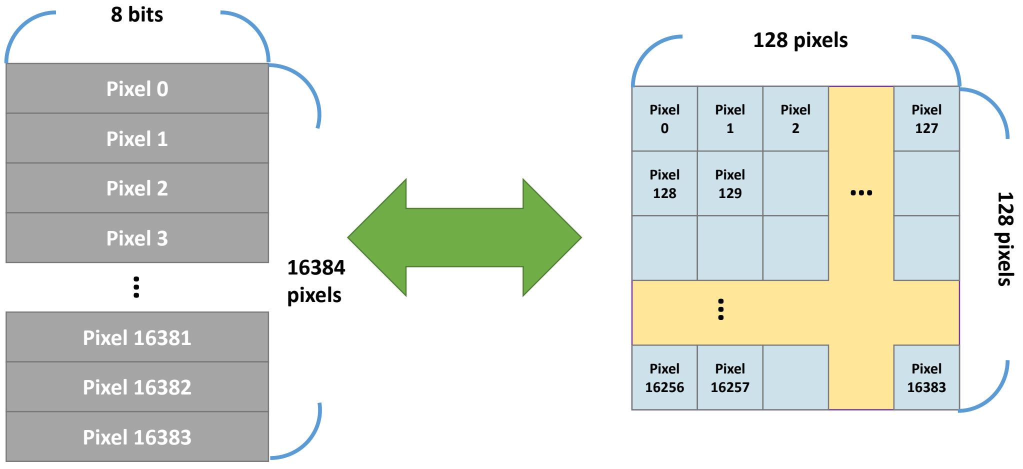

Fig. 4 – Arrangement mapping between grayscale image memory and input image.

The first step of the system is to check the select signal to decide which function will be executed.

If ***sel is 0*** , the IFE circuit will process the image with 3x3 mean filer. Zero-padding has to be applied on the image to keep the size of the image unchanged. Then the padded image is convolved with 3x3 mean filter to obtain result image. The function of 3x3 mean filter is as follows :

$$\text{Result}(x,y) = \frac{1}{mn}\sum_{(s,t)\in S_{xy}} img(s,t), where \;\; m,n = 3$$

Let S_{xy} represent the set of coordinates in a rectangular subimage window of size mxn (m,n is 3), centered at point (x, y).The arithmetic mean filtering process computes the average value of the corrupted image $img(s,t)$ in the area defined by S_{xy}. The value of the result image at any point (x, y) is simply the arithmetic mean computed using the pixels in the region defined by S_{xy}.

Hint: Round down the result value to an integer.

If ***sel is 1*** , the IFE circuit will process the image with 5x5 mean filer. Zero-padding has to be applied on the image. Since the filter size is 5x5, two rows and two columns have to be padded to each side to keep the size of the image unchanged. Then the padded image is convolved with 5x5 mean filter to obtain result image. The function of 5x5 mean filter is as follows :

$$\text{Result}(x,y) = \frac{1}{mn}\sum_{(s,t)\in S_{xy}} img(s,t), where \;\; m,n = 5$$

Hint: Round down the result value to an integer.

If ***sel is 2***, the IFE circuit will process the image with 3x3 min filter. Zero-padding has to be applied on the image to keep the size of the image unchanged. Then the padded image is convolved with 3x3 min filter to obtain result image. The function of 3x3 min filter is as follows :

$$\mathrm{Result}(\mathrm{x},\mathrm{y}) = \min_{(s,t)\in S_{xy}} \{img(s,t)\}$$

Let S_{xy} represent the set of coordinates in a rectangular subimage window of size $3x3$, centered at point (x, y). The arithmetic min filtering process computes the min value of the corrupted image $img(s,t)$ in the area defined by S_{xy}. The value of the result image at any point (x, y) is simply the min value computed using the pixels in the region defined by S_{xy}.

If ***sel is 3***, the IFE circuit will use a threshold to segment the image. The operation is defined as follows:

$$\text{If } \mathrm{img}(\mathrm{x},\mathrm{y}) < 127 \ \ , Result(x,y) = 0;$$
$$\text{else } Result(x,y) = \mathrm{img}(\mathrm{x},\mathrm{y});$$

If the image intensity is less than 127, set the new intensity as zero. Otherwise, set the new intensity as origin intensity.

After the operation is finished, the busy signal should be set to 0, and then testbench will start the verification process.

In the reading phase of result memory, the *addr* signal represents the memory address to read and the *data_rd* signal will provide the data. *wen* signal should be set as low. The timing diagram of result memory reading operation is shown in figure 6.

In the writing phase of result memory, the *addr* signal informs the memory address to be written and the *data_wr* signal should provide the writing data. *wen* signal should be set as high. The timing diagram of result memory writing operation is shown in figure 7.

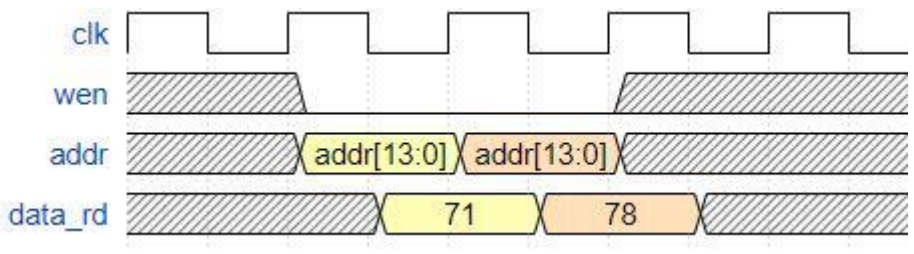

Fig. 6 – Timing diagram of reading result memory.

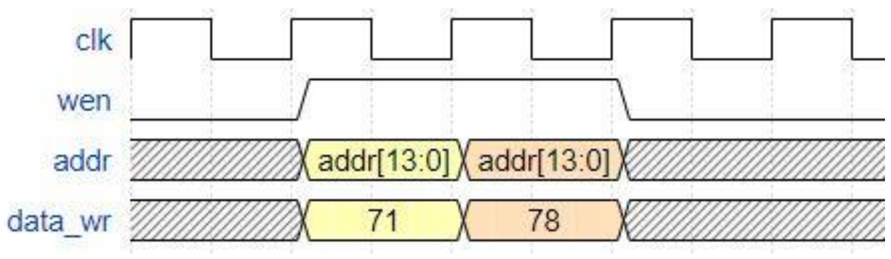

Fig. 7 – Timing diagram of writing result memory.

3 Scoring

3.1 Functional Simulation **[120%]**

3.1.1 Successfully pass the testfixture_mean_3x3.v. [20%]
3.1.2 Successfully pass the testfixture_mean_5x5.v. [20%]
3.1.3 Successfully pass the testfixture_min.v. [30%]
3.1.4 Successfully pass the testfixture_threshold.v. [30%]
3.1.5 The Program can finish successfully. [20%]

Your design must iteratively access the memory and control busy signal properly to get this part of scores.

4 Submission

4.1 Submitted files

You should classify your files into two directories and compress them to .zip format. The naming rule is final_studentID_name.zip. If your file is not named according to the naming rule, you will lose five points. Please submit the compressed file to moodle and email it to diclab62547@gmail.com. The title of email should be set as final_studentID_name.

	RTL category
*.v	All of your Verilog RTL code
	Documentary category
*.pdf	The report file of your design (in pdf).

4.2 Report file

Please follow the spec of report. If your report does not meet the spec, you may lose part of score. You are asked to describe which testfixture you passed. Any information that you want to tell TA can also be written in the report.

4.3 Note

Please do not modify any content of testfixture. Upload your code no matter your design can pass or not. Otherwise, TA can’t give you any point without code.