

CS5120 Homework Assignment 03

Student ID: 107065507

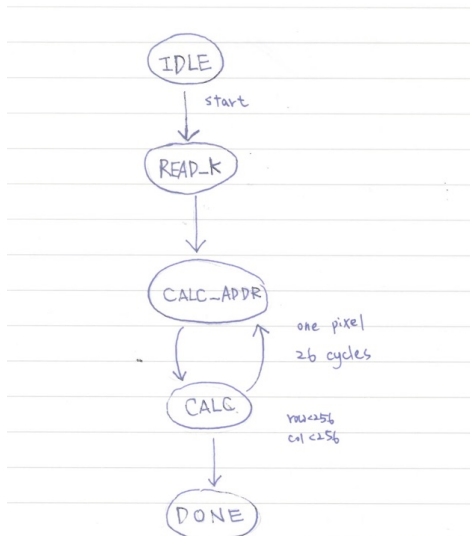
Name: 盧允凡

hw03a

1. Design Concept

Goal: 設計一個 256 x 256 Convolution Engine 做影像處理

FSM design:



IDLE: 待 start signal 進入到 READ_K。

READ_K: 為 Read Kernel 的意思，將 testbench 輸入的 filter 參數存進一個 array。

CALC_ADDR: 計算 output addr value。

CALC: 此狀態會從 SRAM 拿到 working pixel，計算 output pixel。計算 row, column < 256 且 _row, _col < 4 已達到 working pixel 周圍 5x5 的範圍。

這樣設計總共會執行 $256 \times 256 \times 25$ cycles。

Filter Padding issue:

Filter 做 convolution 時有可能會超出圖片的範圍，此時我們要做 zero padding，也就是超出的部分全部當成 0。

控制當前 working pixel 的是在 CALC 這個 state，row 和 col 從 0~255，_row 和 _col 從 -2~2，後者是因為 filter 為 5x5 的，並且從當前 working pixel，也就是 (2, 2) 這個位置做另外 24 個 pixel 的推算。所以這樣做設計。

再把這 4 個值給 CALC_ADDR 做計算。如果 CALC_ADDR 算出的 address 超出 256x256 的範圍，即 $0 \leq \text{row} + \text{_row} < 256$ & $0 \leq \text{col} + \text{_col} < 256$ ，會給一個 isPadding 的 flag，決定此回合在 CALC 需不需要做計算。

另外，如果結果 > 255 或 < 0 ，則表示為 255 或 0。

_row==2 && col==2 時，output_valid 會拉起，可輸出當前 pixel 經過 convolution 的答案。

2. Simulation and Discussion

有三張圖片(.bmp)供我們做 convolution。

分別輸出結果後，可以用以下指令跟正確答案做確認。

```
[yunfanlu@ic29 hw03_v2]$ diff out_log.txt einstein_golden.txt  
[yunfanlu@ic29 hw03_v2]$
```

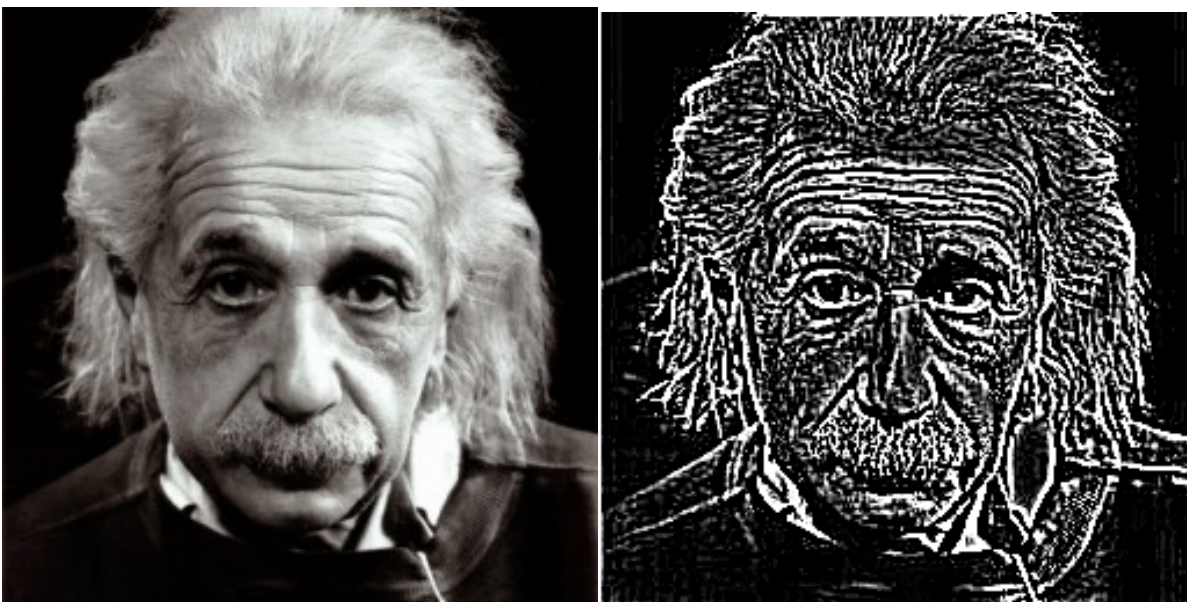
三張圖片都正確無誤。

原圖和做完 convolution 的圖片如下：

lena_output.bmp



Einstein_output.bmp



car_output.bmp



Report Area

```

*****
Report : area
Design : filter_1_DW01_add_0
Version: K-2015.06-SP1
Date   : Fri Apr 12 20:47:24 2019
*****

Information: Updating design information... (UID-85)
Library(s) Used:

    slow (File: /theda21_2/CBDK_IC_Constest/cur/SynopsysDC/db/slow.db)

Number of ports:                50
Number of nets:                 65
Number of cells:                19
Number of combinational cells:  19
Number of sequential cells:     0
Number of macros/black boxes:   0
Number of buf/inv:              2
Number of references:           5

Combinational area:             527.891426
Buf/Inv area:                   6.789600
Noncombinational area:          0.000000
Macro/Black Box area:          0.000000
Net Interconnect area:          undefined (No wire load specified)

Total cell area:                527.891426
Total area:                     undefined

**** End Of Report ****

```

Report Timing Path

Operating Conditions: slow Library: slow
Wire Load Model Mode: top

Startpoint: B[1] (input port)
Endpoint: SUM[15] (output port)
Path Group: (none)
Path Type: max

Point	Incr	Path
-----	-----	-----
input external delay	0.00	0.00 f
B[1] (in)	0.00	0.00 f
U1_1/C0 (ADDFXL)	0.59	0.59 f
U1_2/C0 (ADDFXL)	0.38	0.97 f
U1_3/C0 (ADDFXL)	0.38	1.34 f
U1_4/C0 (ADDFXL)	0.38	1.72 f
U1_5/C0 (ADDFXL)	0.38	2.10 f
U1_6/C0 (ADDFXL)	0.38	2.47 f
U1_7/C0 (ADDFXL)	0.38	2.85 f
U1_8/C0 (ADDFXL)	0.38	3.23 f
U1_9/C0 (ADDFXL)	0.38	3.60 f
U1_10/C0 (ADDFXL)	0.38	3.98 f
U1_11/C0 (ADDFXL)	0.38	4.35 f
U1_12/C0 (ADDFXL)	0.38	4.73 f
U1_13/C0 (ADDFXL)	0.38	5.11 f
U1_14/C0 (ADDFXL)	0.38	5.49 f
U1_15/Y (XOR3X1)	0.23	5.72 r
SUM[15] (out)	0.00	5.72 r
data arrival time		5.72
-----	-----	-----
(Path is unconstrained)		

***** End Of Report *****

hw03b

1. Design Concept

Goal: 使用多個 multipliers 使其 convolution 加速。

FSM 的基本架構不變，但是一次處理一個 pixel 的方式改成，處理到當前 pixel 時，把周圍 5x5 的 pixels 存在一個陣列，宣告 25 個 multipliers，再一口氣做 25 個乘加運算。

但是我這樣的設計並沒有加速 convolution 運算，所需的 cycles 數還是相同，此架構只是為 dual-port sram 做方便的設計。

2. Simulation and Discussion

模擬結果皆正確。

hw03c

1. Design Concept

Goal: 將原本的 sram 改成使用 dual-port sram。

因為是 dual-port sram，我們有 2 個 input output 可以使用。之前是依序取得一個一個 pixel，現在可以一次取得 2 個 pixels。

我的設計是：因為 filter 是 5x5 的，要達到一個 cycle 拿到 2 個 pixels，我把 5x5 filter 分成前 3 rows 以及後 2 rows，平行的取得在 row[0], row[1]以及 row[3], row[4]的 pixels。

此設計讓原本 25 個 cycles 減少為 15 個 cycles。

總共執行 $256 \times 256 \times 15$ 個 cycles。

2. Simulation and Discussion

模擬算出的結果不正確。

Summary

此次 lab 相較於前兩次複雜很多，對於第一次寫 verilog 硬體設計的我來說是相當大的挑戰。Hw03a 花了 2 個星期才把大致的架構完成，好在有成功。Hw03c 的部份是把 sram 改成 dual-port sram，我覺得只把 25 cycles 降低成 15 cycles，有點粗糙，理想上應該是可以把 25 cycles 降成 13 cycles (25/2)。而且模擬結果也不正確，很遺憾還找不到 bug。

此次 lab 也有遇到 ncverilog compile error 的情形，詢問老師後，在指令上加上+nctimescale+1ns/1ps 強制讀 timescale 順利解決。

總結本次 lab 學習到了硬體設計的知識，記憶體(sram)的運作，ncverilog 以及 Makefile 等操作。