

114 學年度
國立中山大學
硬體描述語言

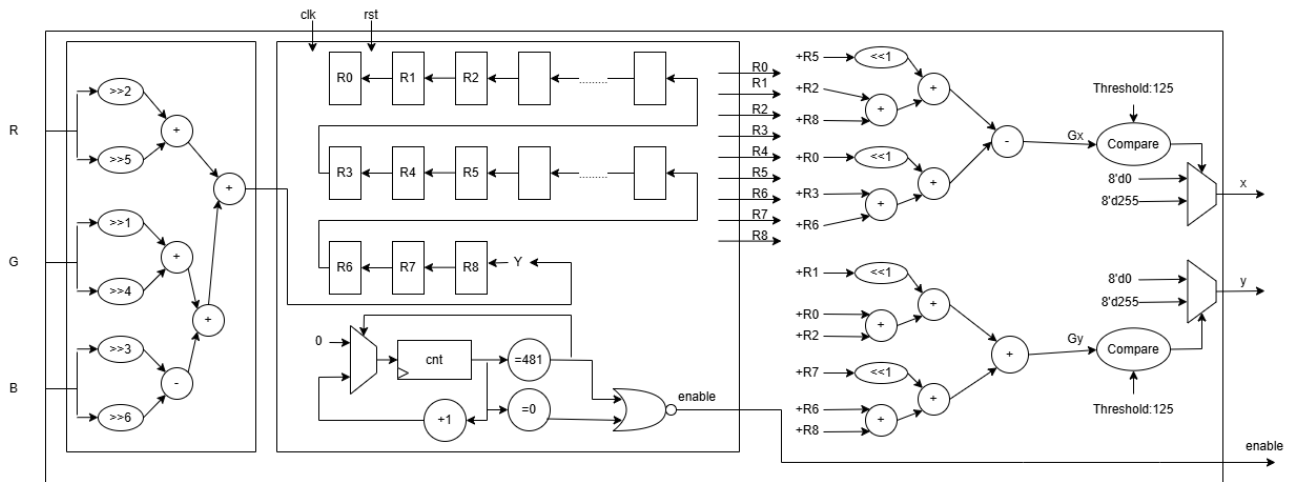
Homework 4
Sobel Edge Detector

作業

授課教師：蕭勝夫

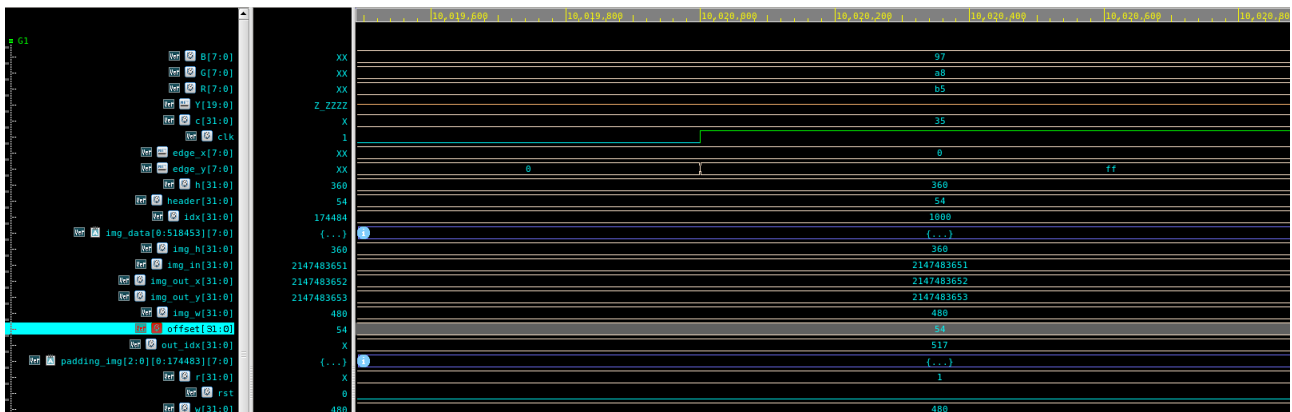
學生學號/班級/姓名：M143010068/電機工程學系碩士班/王嘉良

1.1 整體硬體架構圖 (Figure of Overall Architecture)

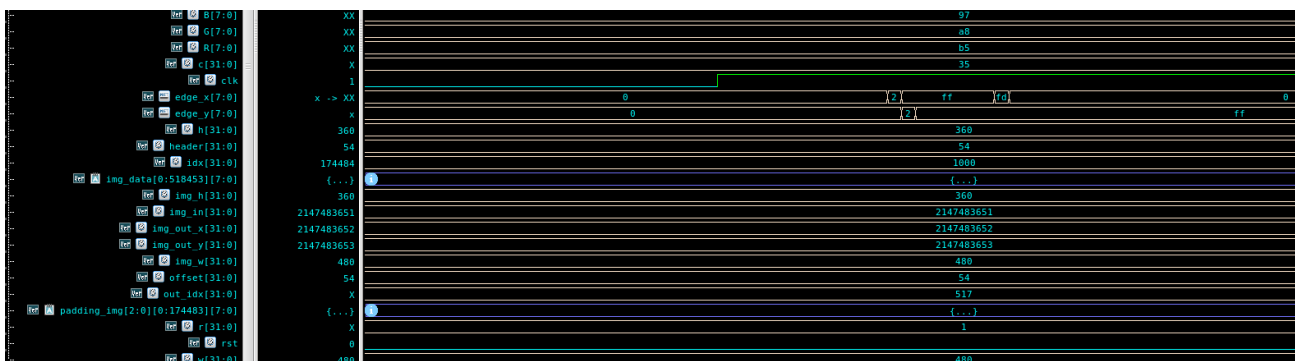


1.2 模擬波形圖與解釋 (Simulation Waveforms & Explanations)

- RTL Simulation Waveform :



- Gate-level Simulation Waveform :



- 解釋 (Explanations) :

在比較中可以發現最關鍵的差異，在 edge_y 的訊號上面，在 presim 的波形圖中，當 clk 上升的瞬間，edge_y 的值從 0 瞬間變為 ff，然而在第二張圖中的 edge_y 訊號，我觀察到當 clk 上升後，edge_y 並沒有馬上變，而是過了一小段時間才跳變成 ff，我覺得是因為 gate delay 的關係，所以當 clk 觸發後，需要經過一段時間才能把訊號傳輸到輸出端。

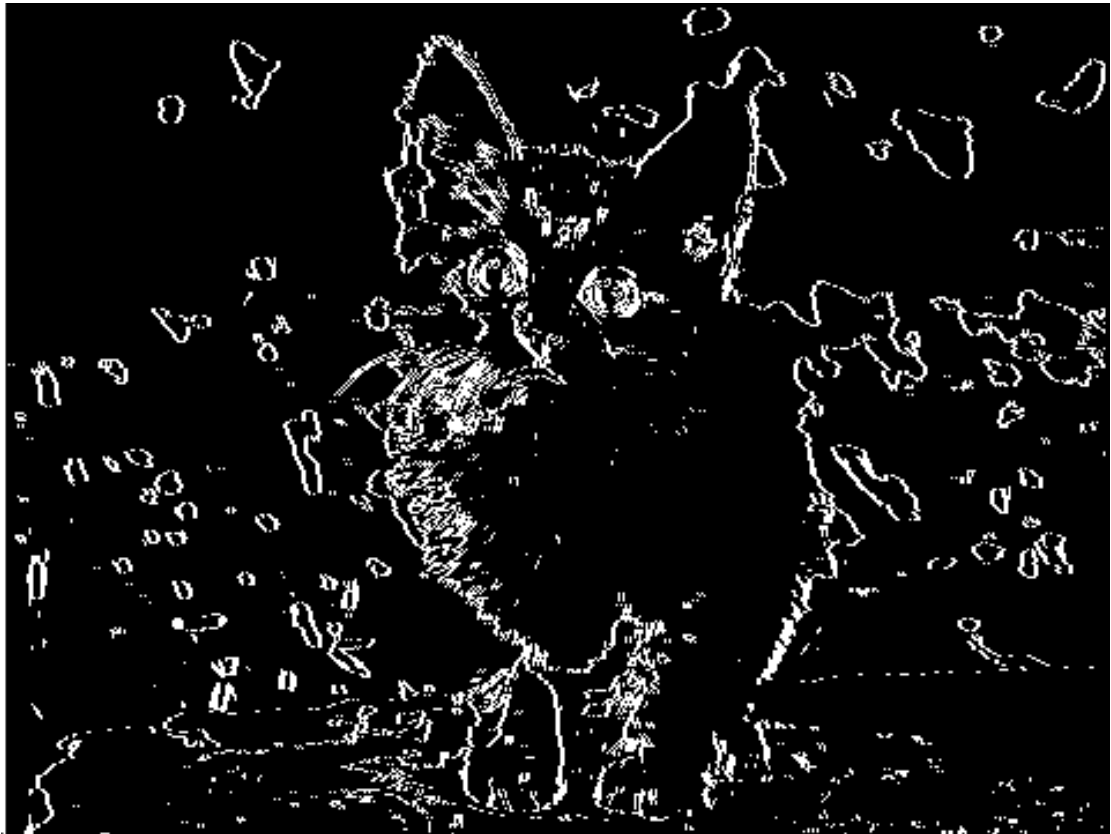
另外一點是雜訊，在 presim 的時候，edge_x 和 edge_y 的數值切換是乾淨俐落的 (例如直接 0->ff)。但是在 postsim 的波形圖中，edge_x 在 clk 上升後的變化，顯示了 2->ff->fd 等短暫的中間值 (紅色框框處的雜訊)，最後才穩定下來。之所以有這樣的情況，推測是因為電路的路徑長度不同，所以有些 bit 跑得快，有些 bit 跑得慢。

1.3 面積與關鍵路徑延遲 (Area Information & Critical Path Delay)

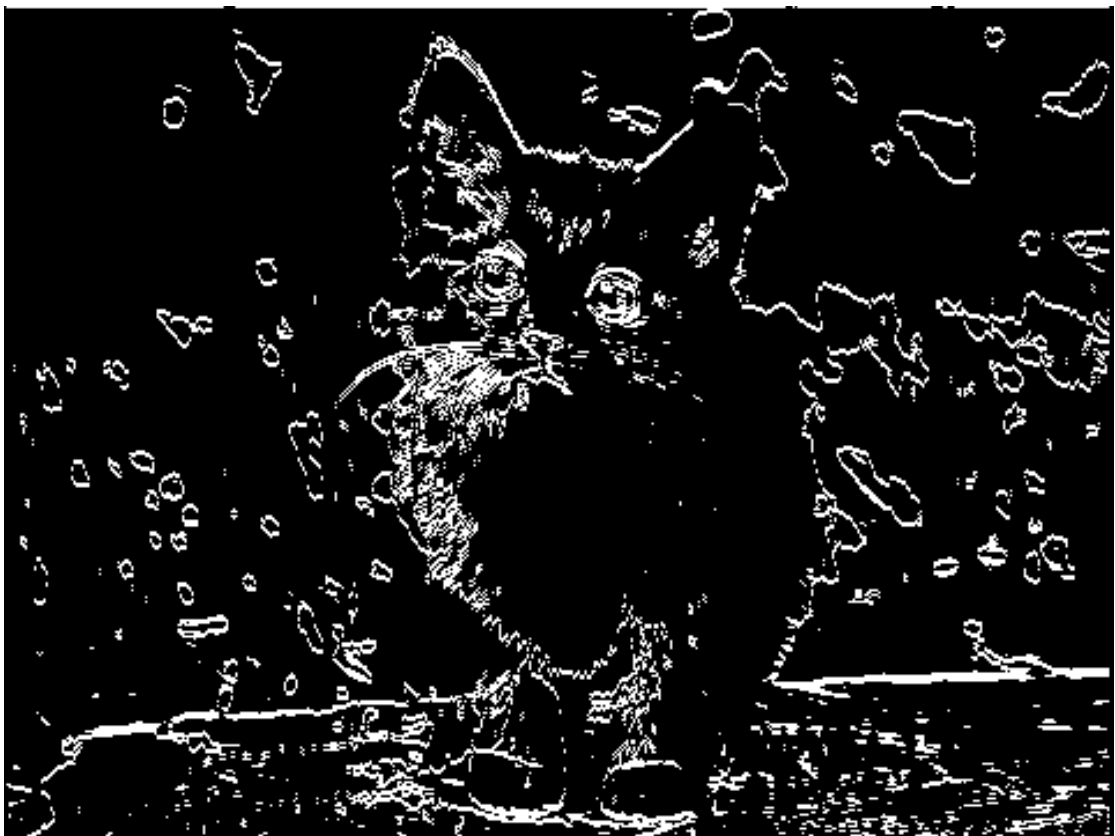
1. Area Report 截圖：

```
timing_report_Edg... area_report_Edge...
1 |
2 | *****
3 | Report : area
4 | Design : Edge
5 | Version: T-2022.03
6 | Date : Tue Nov 18 18:18:01 2025
7 | *****
8 |
9 | Library(s) Used:
10 |
11 | N16ADFP_StdCellss0p72vm40c_ccs (File: /cad/CBDK/ADFP/Executable_Package/Collaterals/IP/stdcell/N16,
12 |
13 | Number of ports: 439
14 | Number of nets: 9527
15 | Number of cells: 9011
16 | Number of combinational cells: 1251
17 | Number of sequential cells: 7745
18 | Number of macros/black boxes: 0
19 | Number of buf/inv: 1010
20 | Number of references: 38
21 |
22 | Combinational area: 370.241284
23 | Buf/Inv area: 205.182720
24 | Noncombinational area: 8833.017921
25 | Macro/Black Box area: 0.000000
26 | Net Interconnect area: undefined (Wire load has zero net area)
27 |
28 | Total cell area: 9203.259205
29 | Total area: undefined
30 |
31 | Hierarchical area distribution
32 | -----
33 |
34 | Global cell area Local cell area
35 |
36 | Hierarchical cell Absolute Total Percent Total Combi- National Noncombi- national Black- boxes Design
37 | -----
38 |
39 | Edge 9203.2592 100.0 232.8134 8833.0179 0.0000 Edge
40 | add_0_root_add_105_2 10.1088 0.1 10.1088 0.0000 0.0000 Edge_DW01_add_9
41 | add_0_root_add_105_4 10.0570 0.1 10.0570 0.0000 0.0000 Edge_DW01_add_8
42 | add_0_root_add_109_2 10.1088 0.1 10.1088 0.0000 0.0000 Edge_DW01_add_4
43 | add_0_root_add_109_4 10.0570 0.1 10.0570 0.0000 0.0000 Edge_DW01_add_5
44 | add_114 8.4499 0.1 8.4499 0.0000 0.0000 Edge_DW01_inc_1
45 | add_115 8.4499 0.1 8.4499 0.0000 0.0000 Edge_DW01_inc_0
46 | add_1_root_add_0_root_sub_18 7.2058 0.1 7.2058 0.0000 0.0000 Edge_DW01_add_1
47 | add_1_root_add_105_2 8.4499 0.1 8.4499 0.0000 0.0000 Edge_DW01_add_11
48 | add_1_root_add_105_4 8.4499 0.1 8.4499 0.0000 0.0000 Edge_DW01_add_10
49 | add_1_root_add_109_2 8.4499 0.1 8.4499 0.0000 0.0000 Edge_DW01_add_6
50 | add_1_root_add_109_4 8.4499 0.1 8.4499 0.0000 0.0000 Edge_DW01_add_7
51 | add_2_root_add_0_root_sub_18 6.7392 0.1 6.7392 0.0000 0.0000 Edge_DW01_add_2
52 | add_88 6.8429 0.1 6.8429 0.0000 0.0000 Edge_DW01_inc_2
53 | sub_105 12.8045 0.1 12.8045 0.0000 0.0000 Edge_DW01_sub_2
54 | sub_109 12.8045 0.1 12.8045 0.0000 0.0000 Edge_DW01_sub_1
55 | -----
56 | Total 370.2413 8833.0179 0.0000
57 |
58 |
```


- 水平邊緣檢測結果 (Horizontal Edge Map) :



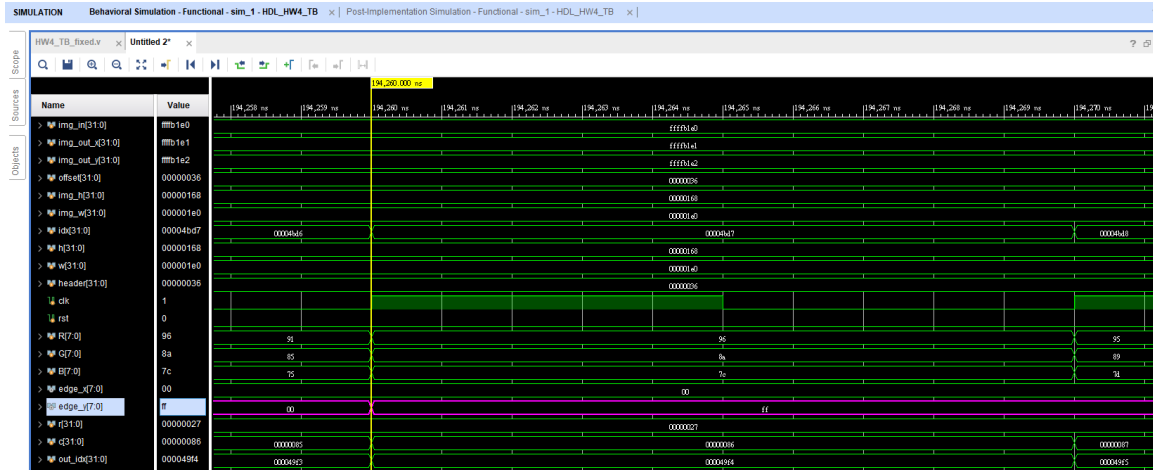
- 垂直邊緣檢測結果 (Vertical Edge Map) :



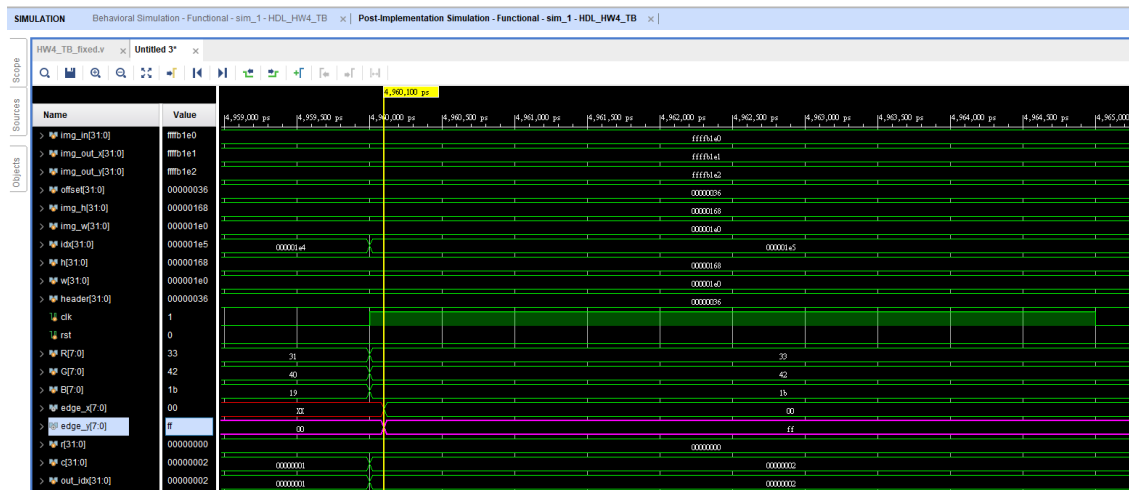
第二章：FPGA Implementation (Xilinx Vivado)

2.1 模擬波形圖與解釋 (Simulation Waveforms & Explanations)

1. Behavior Level Simulation：



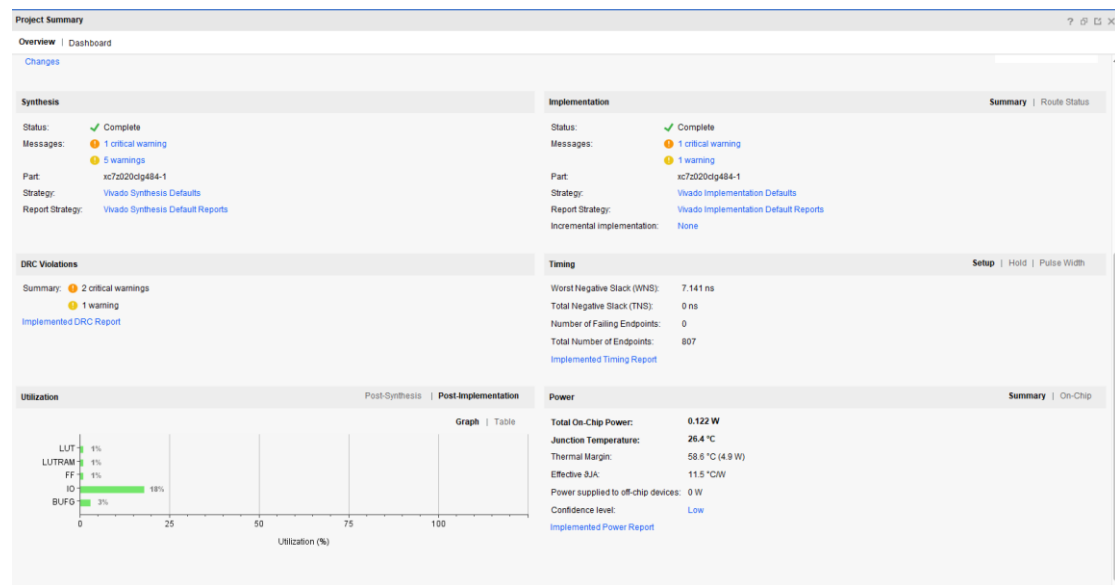
2. Post-Implementation Simulation：



3. 解釋 (Explanations)：

在觀察波形圖的時候，因為先前的 nWave 波形圖主要觀察點放在 edge_y，所以在 vivado 裡面模擬的波形圖，我也把重點觀察放在 edge_y 的電路變化，放大 post_implementation 的波形圖後可以觀察到當 clk 上升為 1 時，edge_y 的訊號並沒有馬上變化，但上圖的 behavioral 就有馬上馬上變成 ff，所以可以發現這裡有正確實現 post_implementation，可以觀察到實際上電路的 delay。

2.2 專案摘要截圖 (Project Summary-Overview Snapshots)



1.5 心得 (Comments)

這一次做的東西跟前面三個作業很不一樣，感覺到這個比較有實作東西的感覺，不知道可以用 Verilog 來做 convolution，然後在把貓咪的圖片生出來後就很有成就感，感覺之後有空可以試看看其他圖片，調整不同參數，看看生出來的圖片會長怎樣。

原本是根據講義給的 $(B \gg 3) - (B \gg 6); // B$ ，讓 B 位移六位元，但後來計算後好像帶入 7 可以讓數值更接近 0.114，但因為已經做到一個段落就沒有修改程式碼了，有機會希望試看看修改，看有什麼不同差別

我是第一次做 convolution 相關的東西，一開始有些概念要去搞懂，像是那個 padding 的觀念或是要如何偵測到哪裡 line buffer 是要被跳過的，我想這部分是遇到比較多困難的地方，在這之後做的事情就可以用到前面的作業的經驗，把一些步驟帶入進去做，基本上就沒遇到什麼問題。