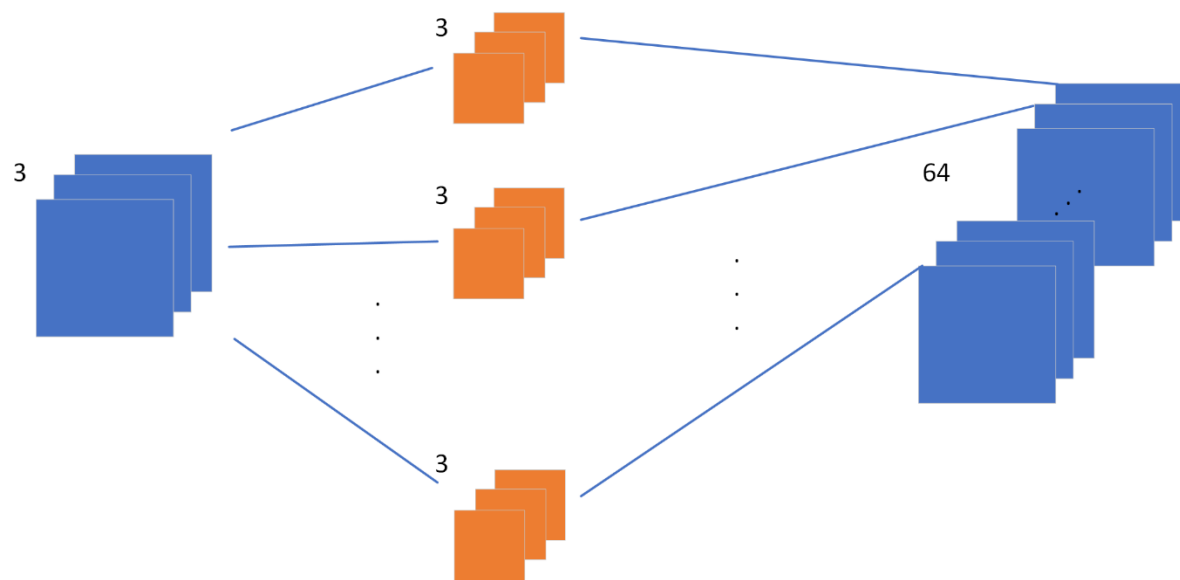


# HDL 作業六說明

王鼎睿 [rueimail4work@g-mail.nsysu.edu.tw](mailto:rueimail4work@g-mail.nsysu.edu.tw)

- 這次作業不用做VGG-16前兩層只要做第一層即可。



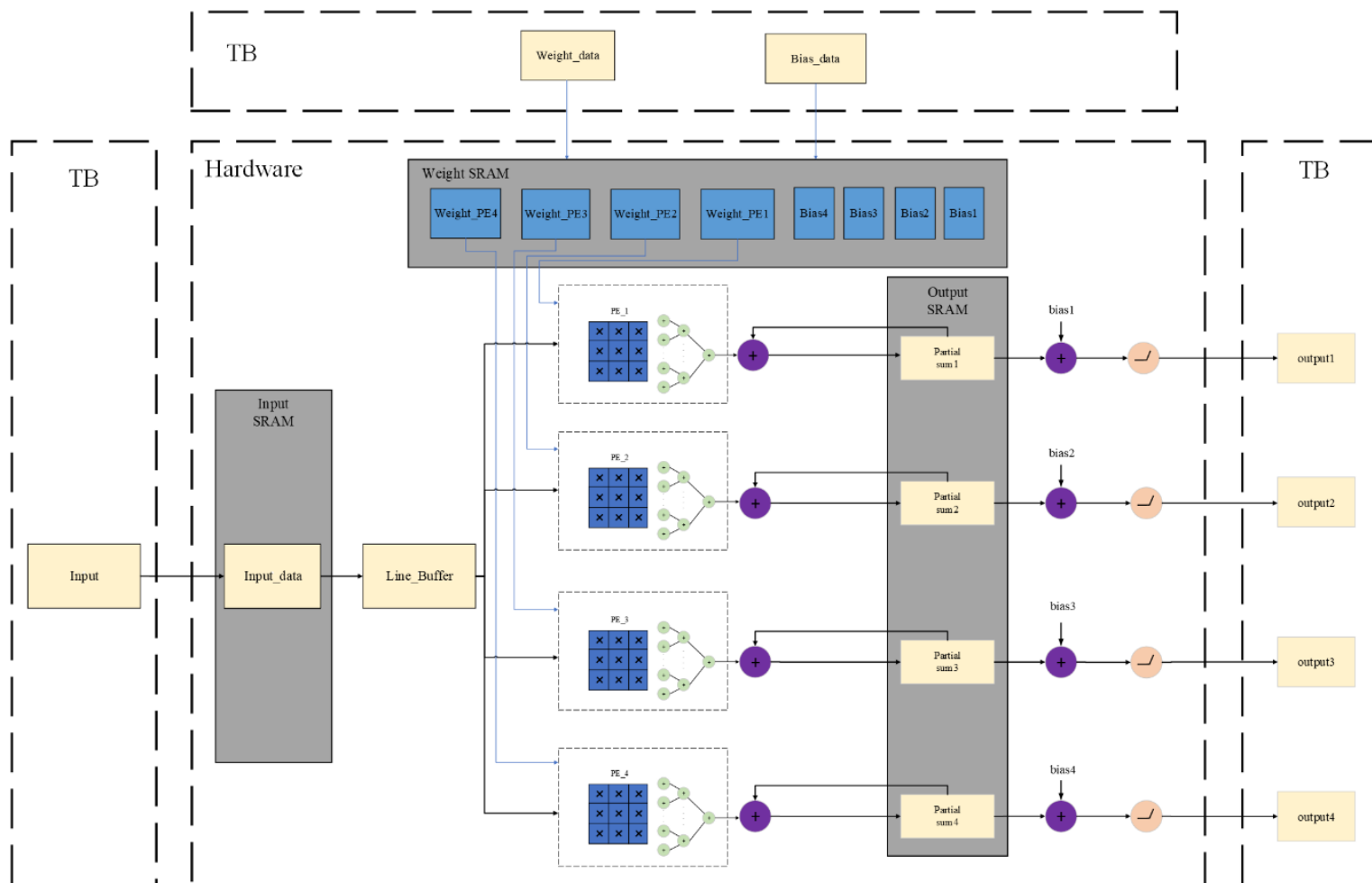
Parameter counts:  
 $(\text{kernal\_width} * \text{kernal\_height} * \text{input\_channels} + 1) * \text{filters} = (3 * 3 * 3 + 1) * 64 = 1728 + 64$

First layer of VGG-16

- 運算流程不變

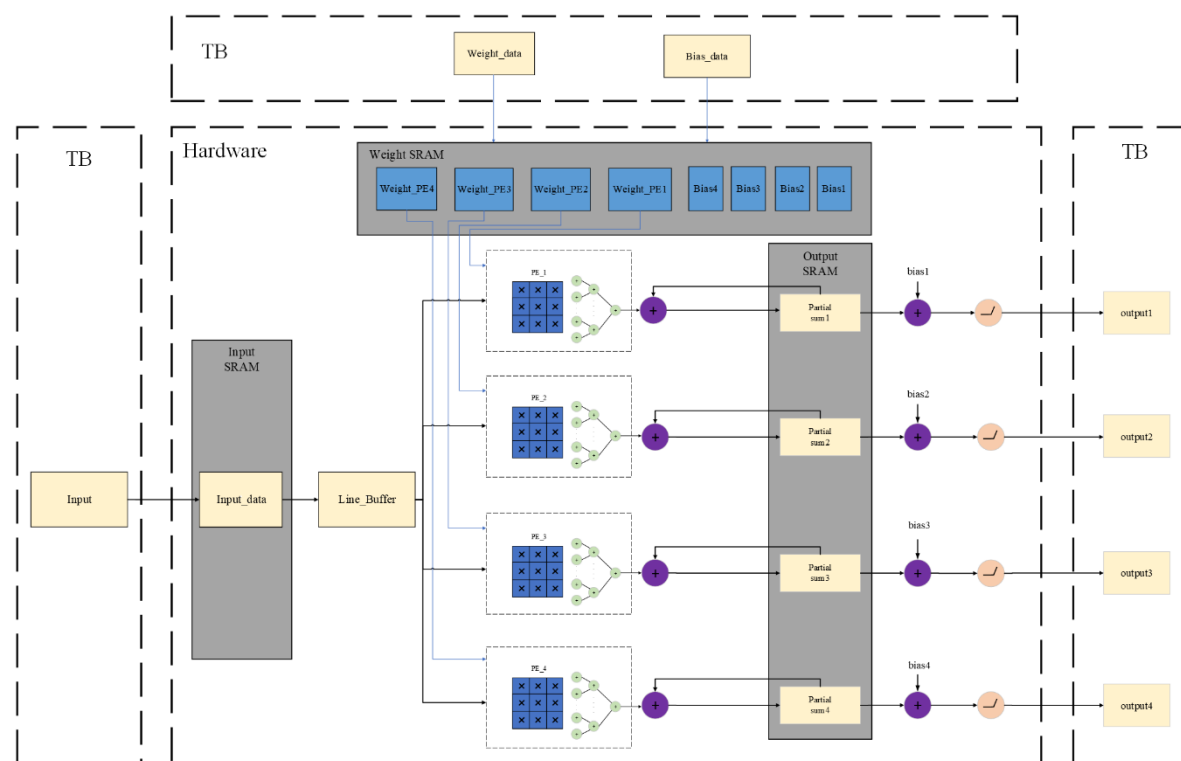


- 硬體設計只能用1種Input Channel=1、Kernel  $3 \times 3 = 9$ 、Output Channel=4(共36個乘法器、1個linebuffer)
- 模型運算方式與作業五類似只多了SRAM、input size  $56 \times 56$ 與限定架構設計。



架構圖

- Memory要有Input SRAM、Weight SRAM、Output SRAM。
- Memory用Single-port、Dual port、Two Port不指定，看自己設計。
- Memory種類用SRAM、Register File不指定，看自己設計。
- High-Speed 與High-Density不指定，但推薦用High-Density，因為memory不用跑到那麼快。
- High-Speed速度快但power 與area大。



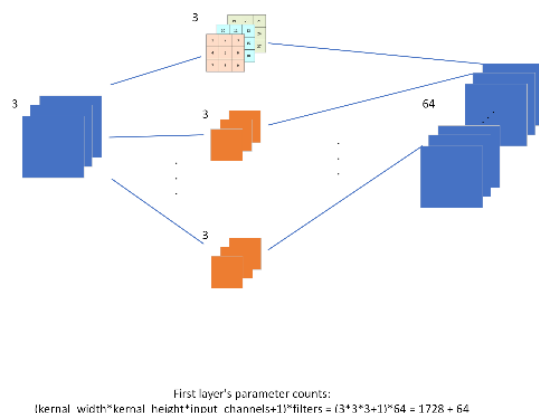
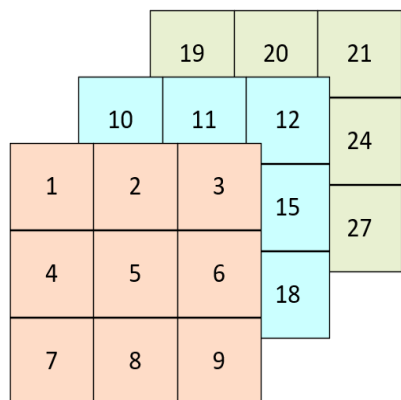
架構圖

- 作業6的第一層Bias、Kernel擺放方式以及txt內的bias、kernel數值與作業5一樣。
- HW6的貓圖也與作業5一樣。



150,582	141,852	BMP 檔案
384	?	Text 來源檔案
10,368	?	Text 來源檔案

2020/9/9 下午 ...	A0F972F1
2020/9/13 下...	F66732F1
2020/9/13 下...	5A36F88E



VGG-16第一層參數在txt中的擺放順序

HDL\_HW5\_所需檔案

EX:conv1\_kernel\_hex.txt

第1-9筆值為VGG-16第一層第一個kernel的第一個channel的參數。  
第10-18筆值為VGG-16第一層第一個kernel的第二個channel的參數。  
依此類推其他bias、kernel的檔案。

1	001b
2	0017
3	fffd
4	0011
5	0002
6	ffe9
7	fffd
8	fff0
9	ffea
10	0023
11	001c
12	fffb
13	0016
14	0002
15	ffe3
16	fffd
17	ffeb
18	ffe1
19	001e
20	001a
21	fffc
22	0013
23	0003
24	ffe7
25	fffd
26	ffee
27	ffe6
28	0007

- 作業六提供的範例TB是把224x224的RGB貓圖轉成56x56，可以整張轉完存array後再送給電路，TB可以做修改但是貓圖給硬體的size要為58x58(padding)，**TB不要分兩個檔案寫(一個tb1.v檔做224轉56，一個tb2.v做56x56送給電路)**

```
//Resize the 224x224 to 56x56
for(idx = 0; idx <`cat56_size; idx = idx+1) begin
    for(jdx = (0+4*(idx/56)); jdx <(4+4*(idx/56)); jdx = jdx+1) begin
        for(kdx = (0+4*i); kdx <(4+4*i); kdx = kdx+1) begin
            R_resize <= R_resize + img_data[(kdx+(jdx*224))*3 + offset + 2];
            G_resize <= G_resize + img_data[(kdx+(jdx*224))*3 + offset + 1];
            B_resize <= B_resize + img_data[(kdx+(jdx*224))*3 + offset + 0];
            #(`period);
        end
    end
end

R_56x56 <=R_resize/16; //Take R_56x56 as input
G_56x56 <=G_resize/16; //Take G_56x56 as input
B_56x56 <=B_resize/16; //Take B_56x56 as input
#(`period);

//write cat56.bmp
$fwrite(img_out, "%c%c%c",B_56x56[7:0] , G_56x56[7:0], R_56x56[7:0]);
if(i==55) i<=0;
else i<=i+1;
#(`period/2);
R_resize <=0;
G_resize <=0;
B_resize <=0;
#(`period/2);

end
//-----
```

轉為  
56x56的  
Pixel

- 在TB內轉好的貓圖(224x224->56x56)



Cat224.bmp



Cat56.bmp



## ● 第一層輸出的圖片



conv1\_1.bmp



conv1\_2.bmp



conv1\_3.bmp



conv1\_4.bmp



conv1\_5.bmp



conv1\_6.bmp



conv1\_7.bmp



conv1\_8.bmp



conv1\_9.bmp



conv1\_10.bmp



conv1\_11.bmp



conv1\_12.bmp



conv1\_13.bmp



conv1\_14.bmp



conv1\_15.bmp



conv1\_16.bmp



conv1\_17.bmp



conv1\_18.bmp



conv1\_19.bmp



conv1\_20.bmp



conv1\_21.bmp



conv1\_22.bmp



conv1\_23.bmp



conv1\_24.bmp



conv1\_25.bmp



conv1\_26.bmp



conv1\_27.bmp



conv1\_28.bmp



conv1\_29.bmp



conv1\_30.bmp



conv1\_31.bmp



conv1\_32.bmp



conv1\_33.bmp



conv1\_34.bmp



conv1\_35.bmp



conv1\_36.bmp



conv1\_37.bmp



conv1\_38.bmp



conv1\_39.bmp



conv1\_40.bmp



conv1\_41.bmp



conv1\_42.bmp



conv1\_43.bmp



conv1\_44.bmp



conv1\_45.bmp



conv1\_46.bmp



conv1\_47.bmp



conv1\_48.bmp



conv1\_49.bmp



conv1\_50.bmp



conv1\_51.bmp



conv1\_52.bmp



conv1\_53.bmp



conv1\_54.bmp



conv1\_55.bmp



conv1\_56.bmp



conv1\_57.bmp



conv1\_58.bmp



conv1\_59.bmp



conv1\_60.bmp



conv1\_61.bmp



conv1\_62.bmp



conv1\_63.bmp



conv1\_64.bmp