Circuit Talk

組員: 劉浩崴 B093011055
陳晉毅 B093011056

邊緣偵測-簡介[1]

- 高階影像處理的前處理 ex. 物體偵測、特徵提取、運動分析
- 偵測影像在物理、光度、幾何的變化

灰階圖片

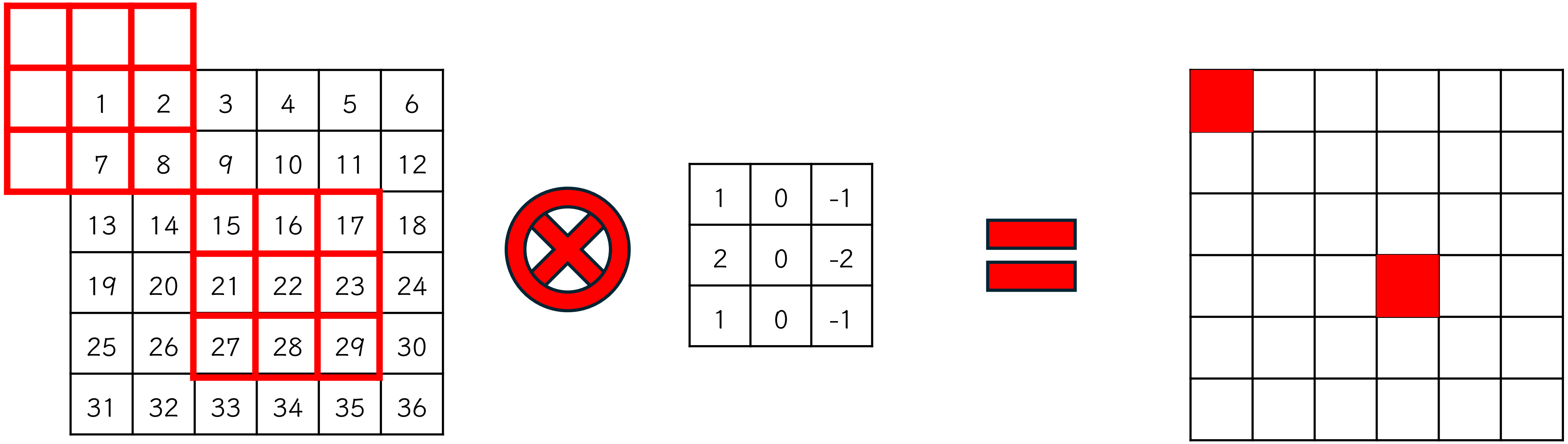
圖片邊緣

邊緣偵測-Sobel operator

- 計算單一方向的梯度
  使用差分求出變化率
- 用kernel和圖片作捲積 (周圍補0)
- 輸出：$|G_X| + |G_Y|$

Gx Kernel		
1	0	-1
2	0	-2
1	0	-1

Gy Kernel		
1	2	1
0	0	0
-1	-2	-1



原始圖片 Gx X方向的梯度大小

判斷邊緣-二值化

- 設定邊緣變化的標準：threshold
- 變化率高過threshold則判斷為邊緣

33	17	23	24	16	31
22	8	18	7	15	30
34	10	1	14	29	13
28	2	25	26	6	12
9	35	3	4	32	11
19	20	36	27	5	21

圖片梯度

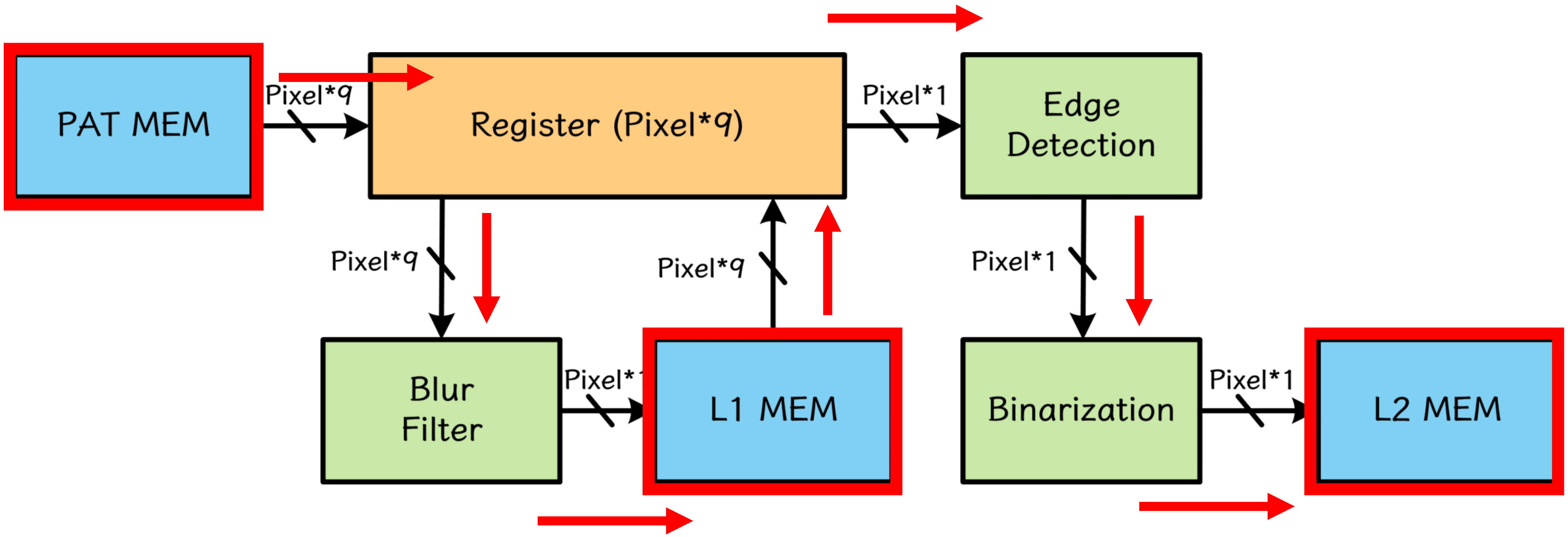
Threshold = 20

36	0	36	36	0	36
36	0	0	0	0	36
36	0	36	0	36	0
36	0	36	36	0	0
0	36	0	0	36	0
0	36	36	36	0	36

二值化結果

本次使用的演算法及系統架構圖

- 模糊濾波器 → sobel filter → 二值化

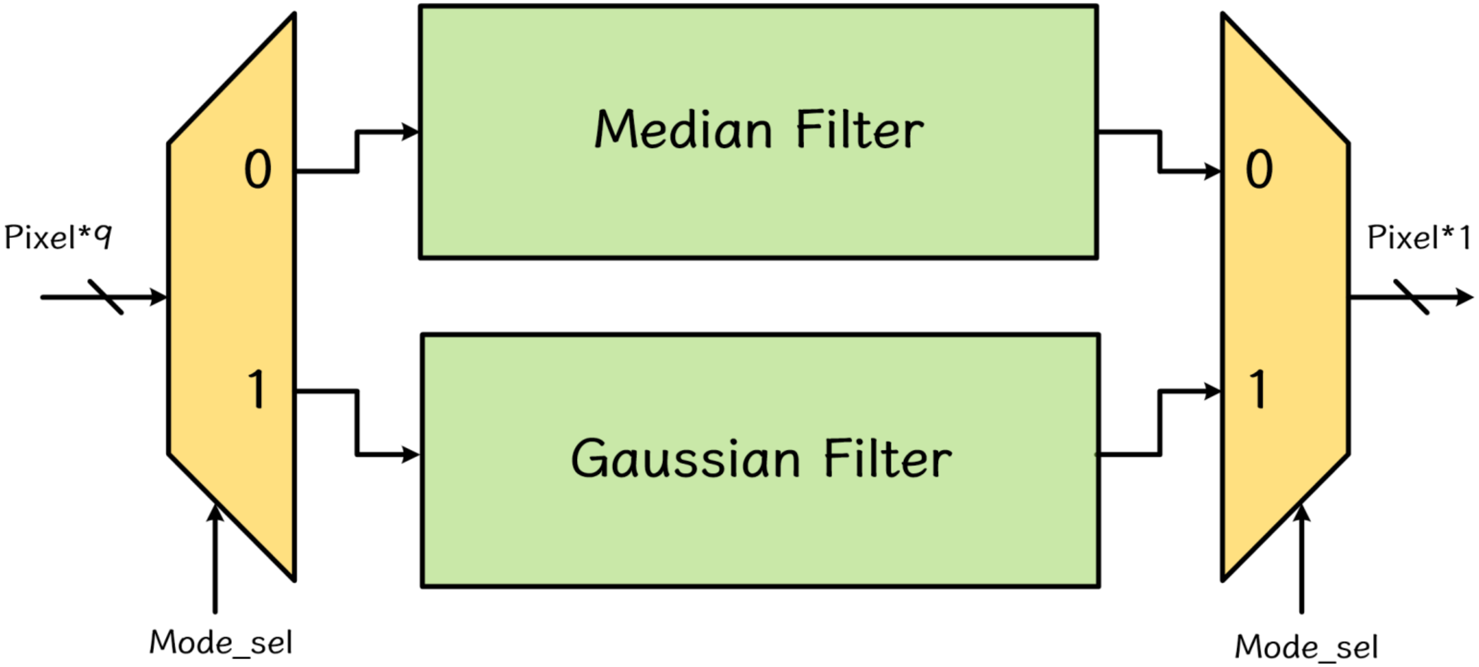


濾波器架構圖

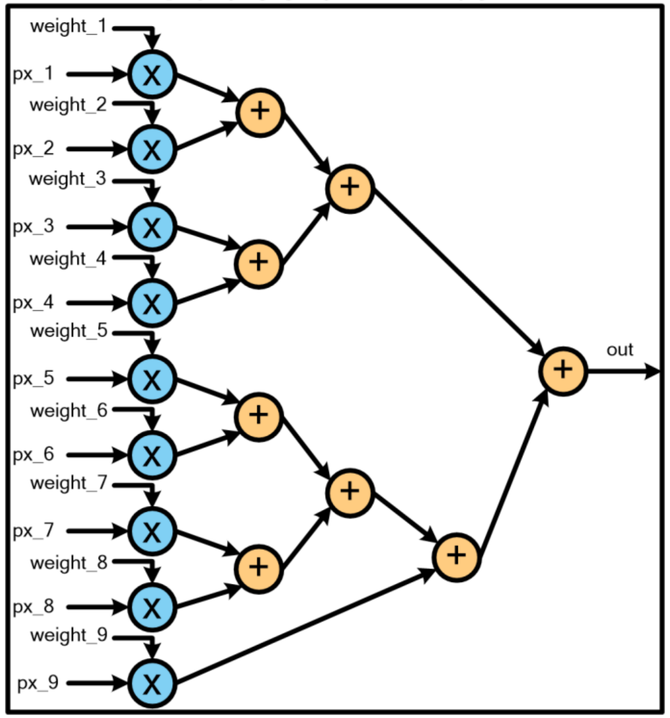
- 使用中值濾波器&高斯濾波器
- 可選擇要通過哪一種

Weight 1 0.075	Weight 2 0.124	Weight 3 0.075
Weight 4 0.124	Weight 5 0.204	Weight 6 0.124
Weight 7 0.075	Weight 8 0.124	Weight 9 0.075

WL:12
FL:12

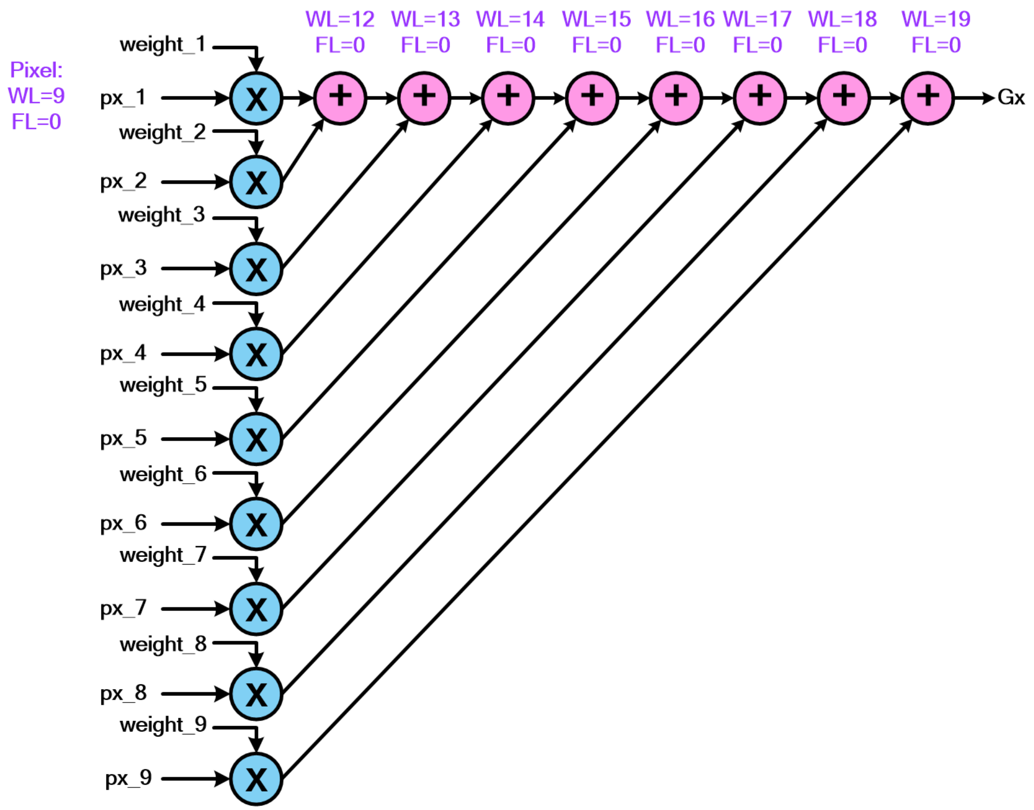


Gaussian Filter



邊緣偵測電路-Sobel

- 使用sobel Kernel的捲積電路

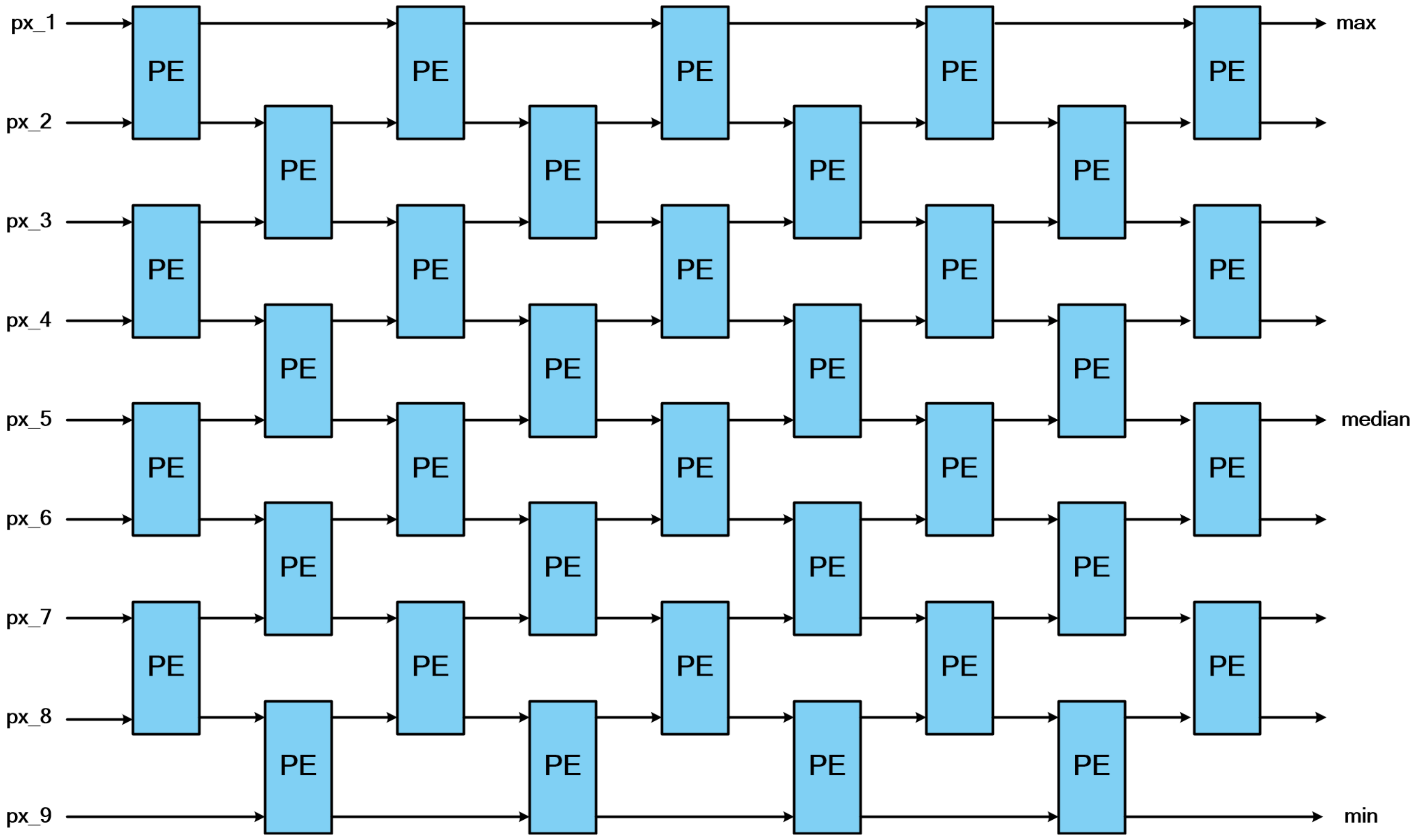


Gx Kernel		
Weight 1 -1	Weight 2 0	Weight 3 1
Weight 4 -2	Weight 5 0	Weight 6 2
Weight 7 -1	Weight 8 0	Weight 9 1

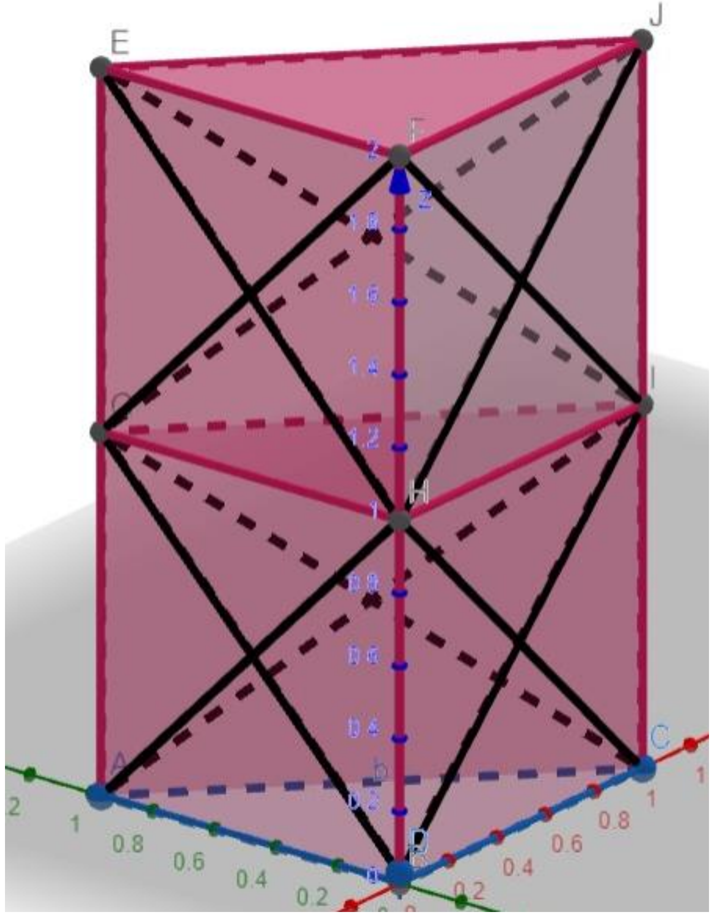
Gy Kernel		
Weight 1 1	Weight 2 2	Weight 3 1
Weight 4 0	Weight 5 0	Weight 6 0
Weight 7 -1	Weight 8 -2	Weight 9 -1

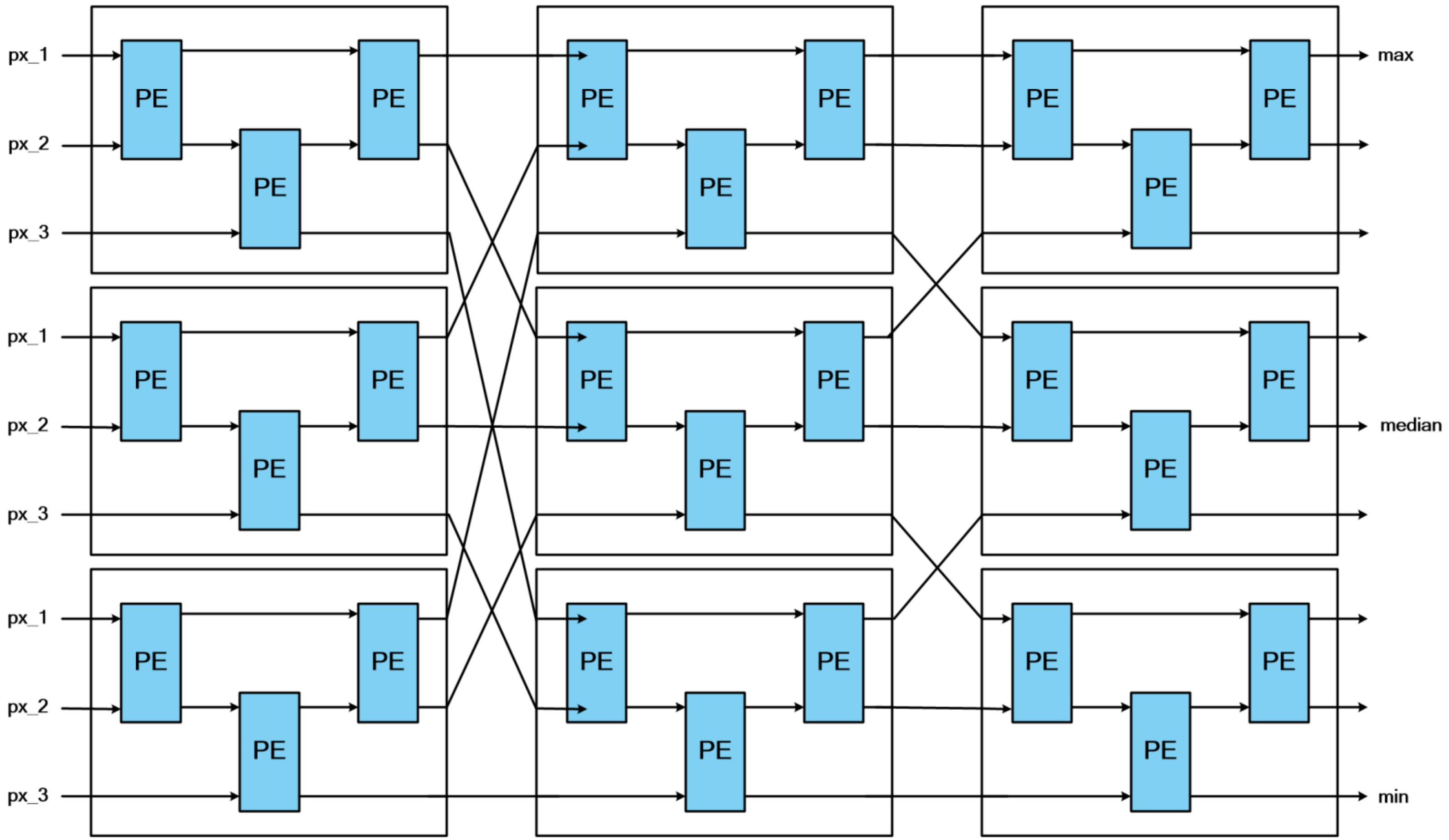
技巧1: Systolic array

使用Systolic array實現中值濾波器 [2]

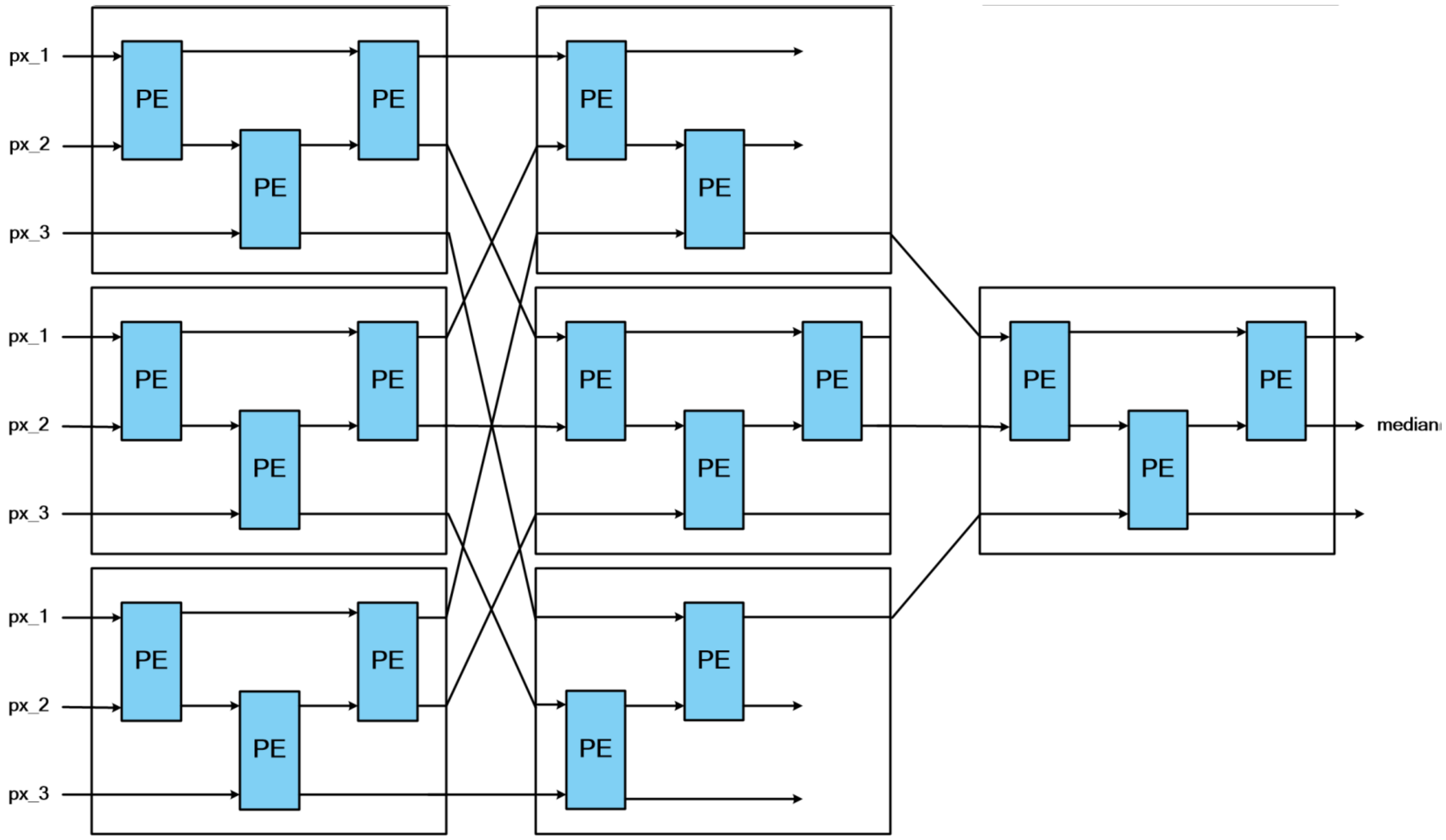


中值濾波器-變化[2]



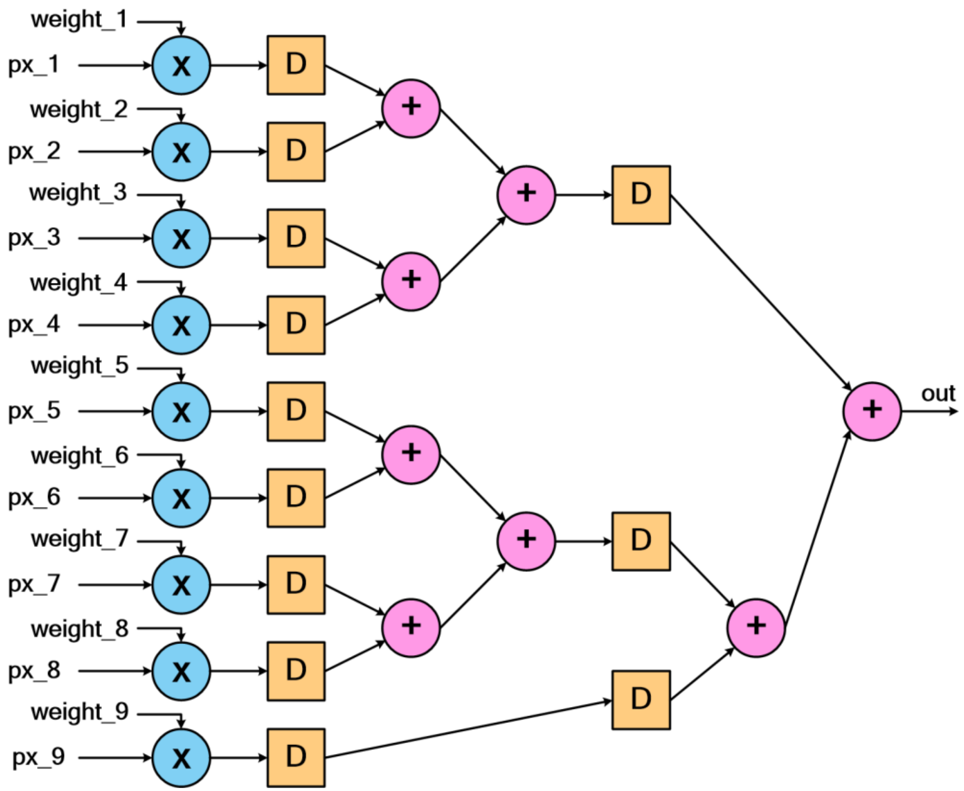


中值濾波器-化簡過程[2]



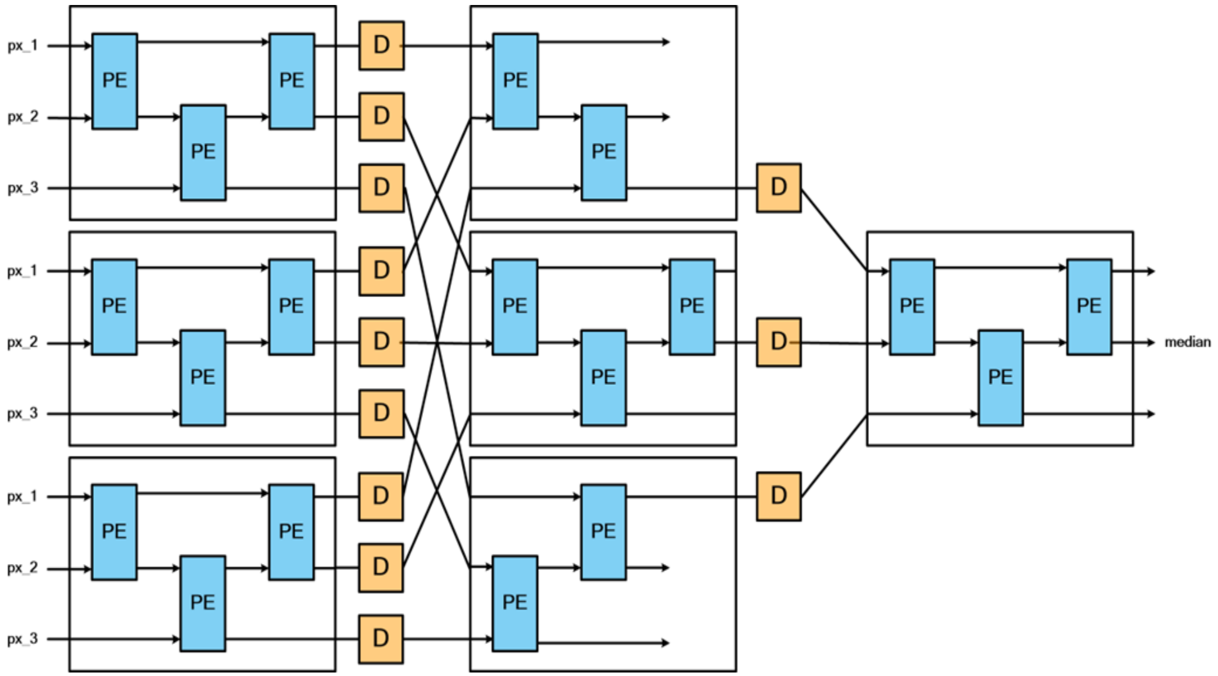
技巧2: Pipeline

在Filter電路中加入Pipeline



	Original	3-level pipline
面積 (um^2)	9734.5893	16788.9835
乘法器占比 (%)	46.2	25.5
加法器占比 (%)	44.2	26.4
Latency (cycle)	1	3
Throughput (Data/cycle)	1	1
Critical Path	1Tm + 4Ta	2Ta (3rd level)

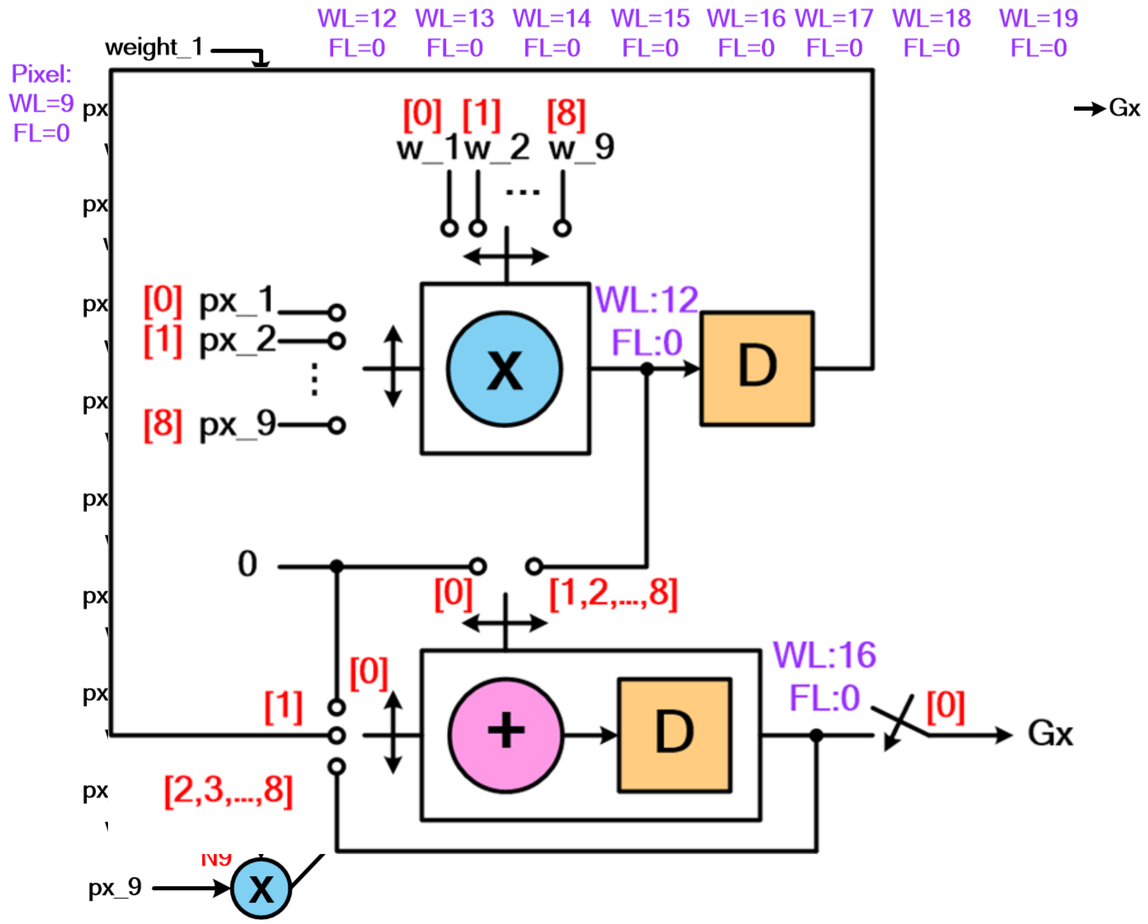
在Filter電路中加入Pipeline



	Original	3-level pipline
面積 (um^2)	11854.6413	9610.6787
PE數量 (個)	36	19
PE占比 (%)	100	77.97
Latency (cycle)	1	3
Throughput (Data/cycle)	1	1
Critical Path	$9T_{PE}$	$3T_{PE}$

技巧3: Folding

對Sobel電路使用Folding



N=9, Pm=0, Pa=1

Sm={N1, N2, N3, N4, N5, N6, N7, N8, N9}

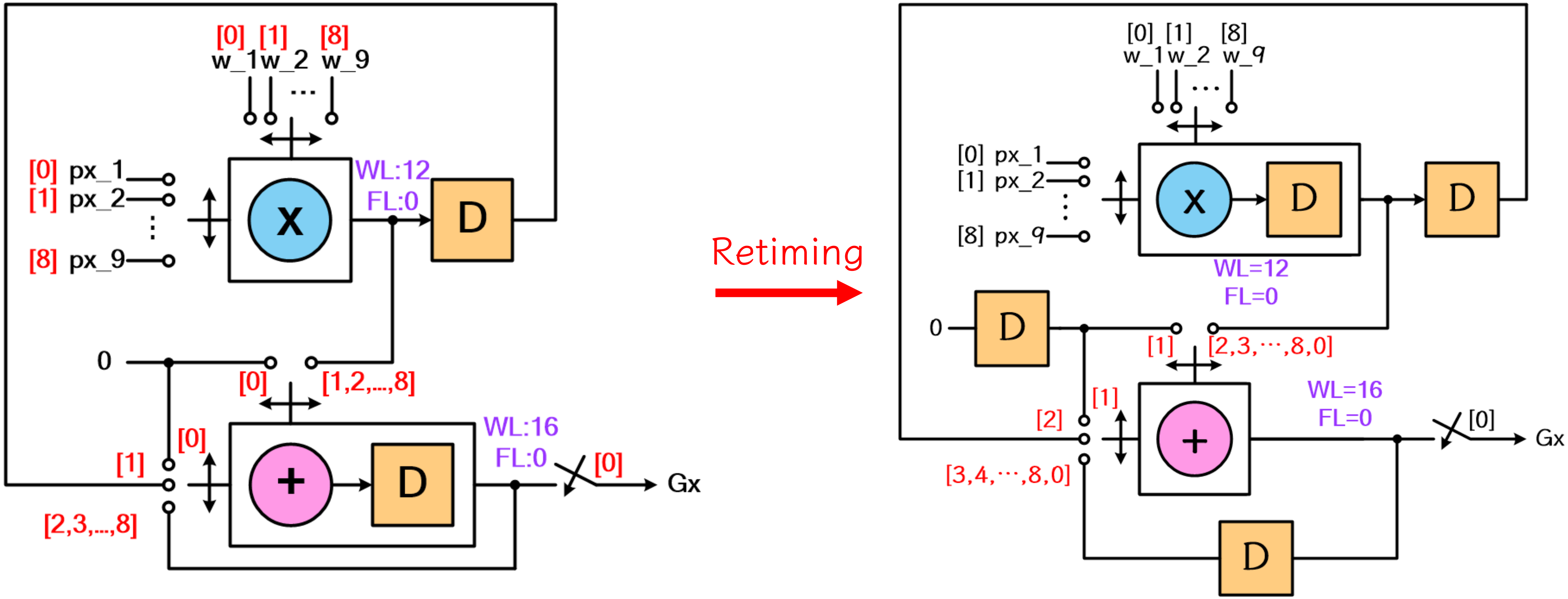
Sa={0, N10, N11, N12, N13, N14, N15, N16, N17}

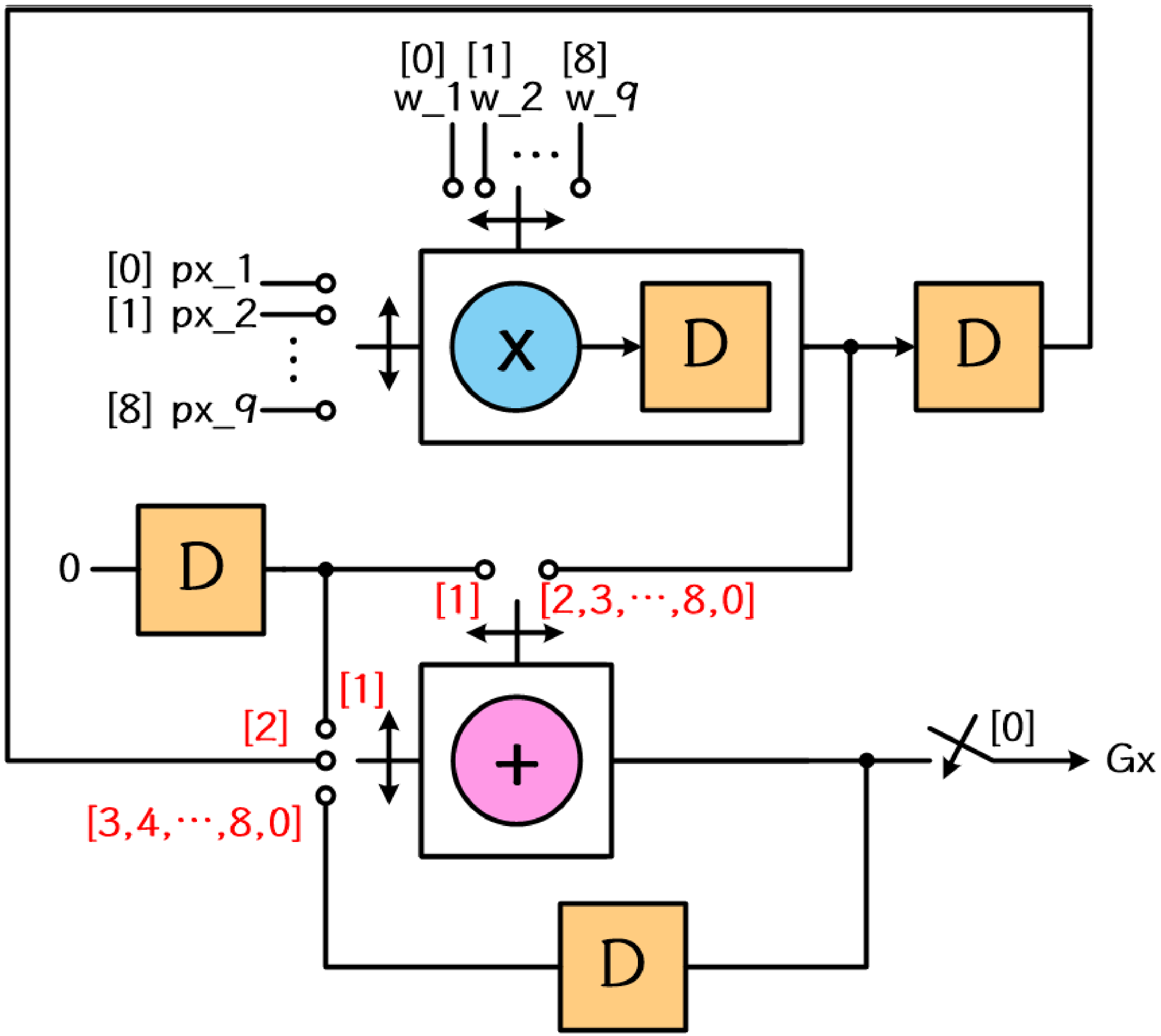
DF (U→V)	=	Nw(e)	-	Pu	+	v	-	u		
DF (N1→N10)	=	9*0	-	0	+	1	-	0	=	1
DF (N2→N10)	=	9*0	-	0	+	2	-	1	=	0
DF (N3→N11)	=	9*0	-	0	+	2	-	2	=	0
DF (N4→N12)	=	9*0	-	0	+	3	-	3	=	0
DF (N5→N13)	=	9*0	-	0	+	4	-	4	=	0
DF (N6→N14)	=	9*0	-	0	+	5	-	5	=	0
DF (N7→N15)	=	9*0	-	0	+	6	-	6	=	0
DF (N8→N16)	=	9*0	-	0	+	7	-	7	=	0
DF (N9→N17)	=	9*0	-	0	+	8	-	8	=	0
DF (N10→N11)	=	9*0	-	1	+	2	-	1	=	0
DF (N11→N12)	=	9*0	-	1	+	3	-	2	=	0
DF (N12→N13)	=	9*0	-	1	+	4	-	3	=	0
DF (N13→N14)	=	9*0	-	1	+	5	-	4	=	0
DF (N14→N15)	=	9*0	-	1	+	6	-	5	=	0
DF (N15→N16)	=	9*0	-	1	+	7	-	6	=	0
DF (N16→N17)	=	9*0	-	1	+	8	-	7	=	0

技巧4: Retiming

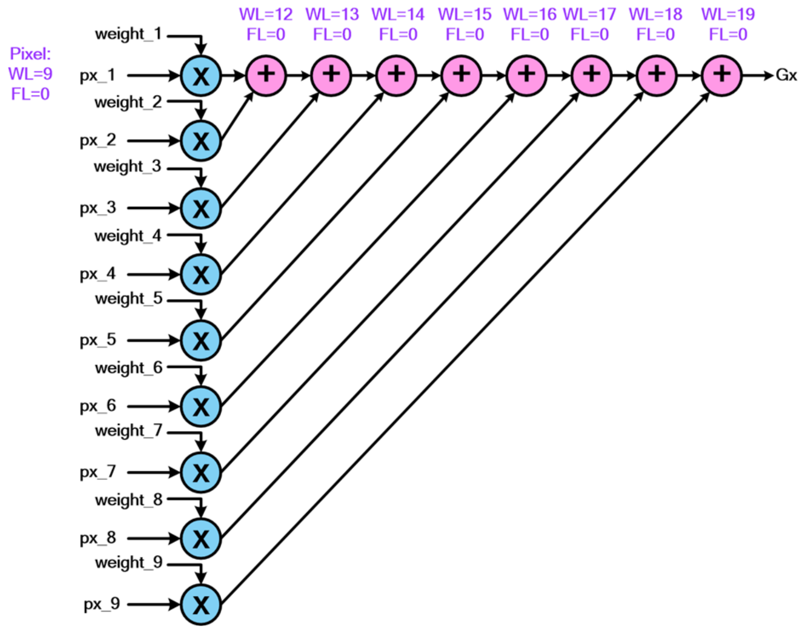
對Folding的sobel進行Retiming

- 斬斷critical path，降低T_{clk}

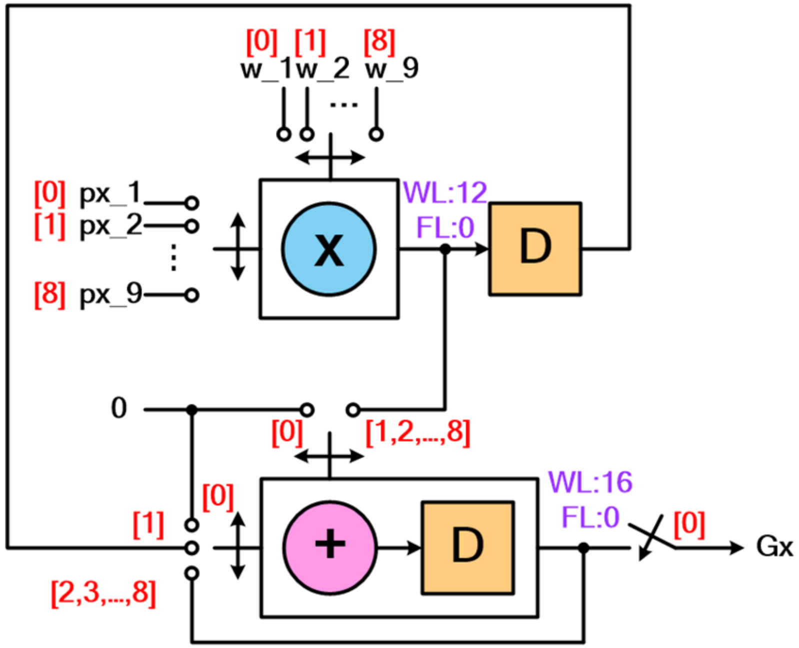




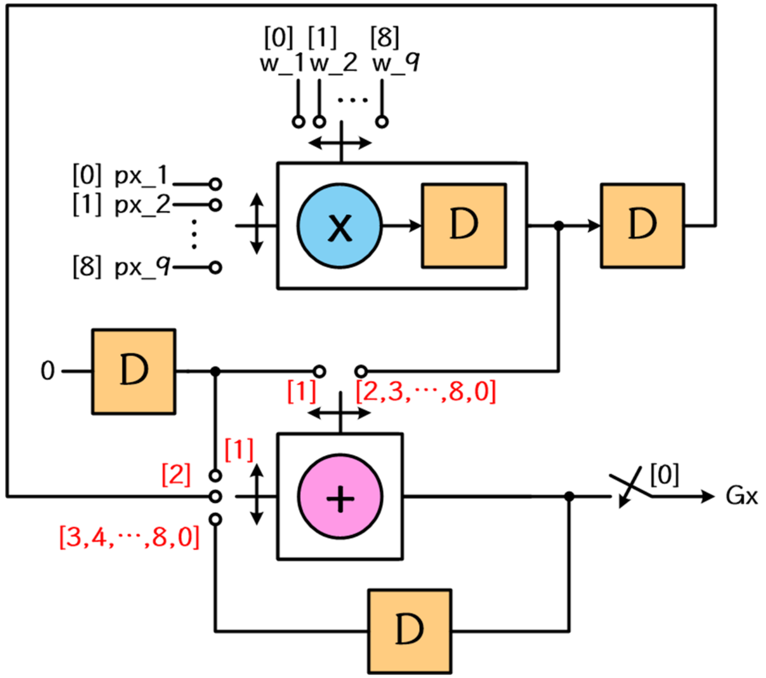
[0] [1] [8]
w_1 w_2 w_q
[0] px_1
[1] px_2
[8] px_q
x
D
D
0
D
[1]
[2,3,⋯,8,0]
[1]
[2]
[3,4,⋯,8,0]
+
[0]
Gx
D



- Original



- Folding



- Folding+Retiming

	Original	Folding	Folding+Retiming
面積 (um^2)	20835.5856	7393.8744	8200.1393
乘法器數量 (個)	18	2	2
加法器數量 (個)	16	2	2
Latency (cycle)	1	10	10
Throughput (Data/cycle)	1	0.1	0.1
Critical Path	1Tm + 8Ta	1Ta + 1Tm	1Ta

邊緣偵測電路邏輯合成結果

	Original	New
面積 (um^2)	41827.331	50215.88125
Gaussian Blur	9680.2726	16788.9835
Median Blur	11732.429	9610.6788
Sobel Filter	4642.3891	7858.962
Binarization	210.4776	210.4776
Critical Path	Control counter	Gaussian
Tclk (ns)	10	10
Power (mW)	0.8229	1.6318
總運算時間 (cycle)	16392	53259

RTL code & Testbench

- https://drive.google.com/drive/folders/1WmK_meILE0G6bZqjkJds70AE_8L7z9y_?usp=drive_link

```
----------------------------------------------------

START!!! Simulation Start .....

----------------------------------------------------

Layer 1 is correct !
Layer 2 is correct !

----------------------------------------------------

-------------------- S U M M A R Y ----------------

Congratulations! Layer 1 data have been generated successfully! The result is PASS!!

Congratulations! Layer 2 data have been generated successfully! The result is PASS!!

----------------------------------------------------
```

以中值濾波器進行影像模糊

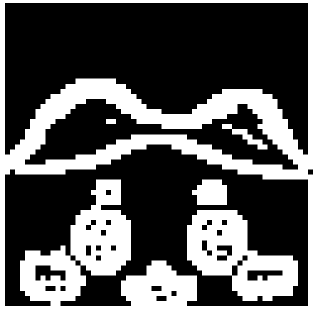
```
-----------------------------------------------------

--------------------- S U M M A R Y -----------------

FAIL!!!  There are        1810 errors! in Layer 1

FAIL!!!  There are           5 errors! in Layer 2

-----------------------------------------------------
```

以高斯濾波器進行影像模糊

模擬結果





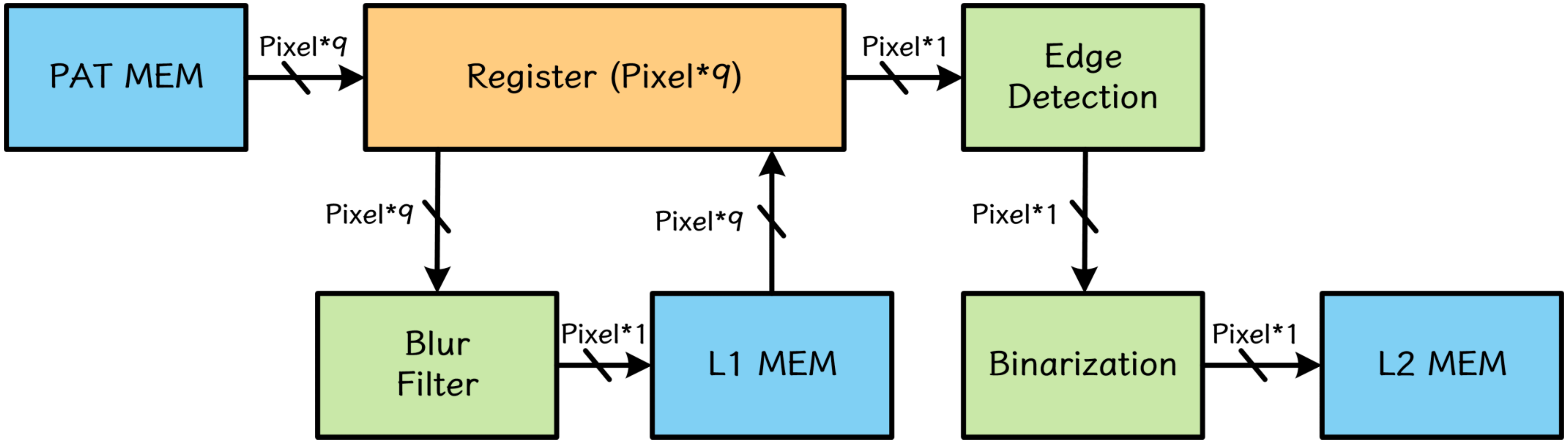
參考資料

- [1] Y. Liu, Z. Xie and H. Liu, "An Adaptive and Robust Edge Detection Method Based on Edge Proportion Statistics," *IEEE Transactions on Image Processing*, vol. 29, pp. 5206-5215, 2020.
- [2] Rasheed, A. H, "FPGA-based optimized systolic design for median filtering algorithms," *International Journal of Applied Engineering Research*, vol. 12, no. 24, pp. 16100-16113, 2017.

The End

技巧5: Life Time Analysis (未實現)

使用life time分析 省去記憶體面積



L1 MEM

1	2	4	1	2	4	1	2	4	1	2	4	1	2	4	1	2	4
3	5	7	3	5	7	3	5	7	3	5	7	3	5	7	3	5	7
6	8	9	6	8	9	6	8	9	6	8	9	6	8	9	6	8	9
1	2	4	1	2	4	1	2	4	1	2	4	1	2	4	1	2	4
3	5	7	3	5	7	3	5	7	3	5	7	3	5	7	3	5	7
6	8	9	6	8	9	6	8	9	6	8	9	6	8	9	6	8	9
1	2	4	1	2	4	1	2	4	1	2	4	1	2	4	1	2	4
3	5	7	3	5	7	3	5	7	3	5	7	3	5	7	3	5	7
6	8	9	6	8	9	6	8	9	6	8	9	6	8	9	6	8	9
1	2	4	1	2	4	1	2	4	1	2	4	1	2	4	1	2	4
3	5	7	3	5	7	3	5	7	3	5	7	3	5	7	3	5	7
6	8	9	6	8	9	6	8	9	6	8	9	6	8	9	6	8	9
1	2	4	1	2	4	1	2	4	1	2	4	1	2	4	1	2	4
3	5	7	3	5	7	3	5	7	3	5	7	3	5	7	3	5	7
6	8	9	6	8	9	6	8	9	6	8	9	6	8	9	6	8	9
1	2	4	1	2	4	1	2	4	1	2	4	1	2	4	1	2	4
3	5	7	3	5	7	3	5	7	3	5	7	3	5	7	3	5	7
6	8	9	6	8	9	6	8	9	6	8	9	6	8	9	6	8	9

L1 MEM

1	2	3	4	5	6
1	1	1	1	1	1
2	3	2	2	2	2
3	4	5	3	3	3
4	5	6	7	4	4
5	6	7	8	9	5
6	7	8	9	10	11

L2 MEM

5	7	3	5	7	3	5	7	3	5	7	3	5	7	3	5	7	7
8	9	6	8	9	6	8	9	6	8	9	6	8	9	6	8	9	9
2	4	1	2	4	1	2	4	1	2	4	1	2	4	1	2	4	4
5	7	3	5	7	3	5	7	3	5	7	3	5	7	3	5	7	7
8	9	6	8	9	6	8	9	6	8	9	6	8	9	6	8	9	9
2	4	1	2	4	1	2	4	1	2	4	1	2	4	1	2	4	4
5	7	3	5	7	3	5	7	3	5	7	3	5	7	3	5	7	7
8	9	6	8	9	6	8	9	6	8	9	6	8	9	6	8	9	9
2	4	1	2	4	1	2	4	1	2	4	1	2	4	1	2	4	4
5	7	3	5	7	3	5	7	3	5	7	3	5	7	3	5	7	7
8	9	6	8	9	6	8	9	6	8	9	6	8	9	6	8	9	9
2	4	1	2	4	1	2	4	1	2	4	1	2	4	1	2	4	4
5	6	3	5	7	3	5	7	3	5	7	3	5	7	3	5	7	7
7	8	6	8	9	6	8	9	6	8	9	6	8	9	6	8	9	9
2	4	1	2	4	1	2	4	1	2	4	1	2	4	1	2	4	4
5	7	3	5	7	3	5	7	3	5	7	3	5	7	3	5	7	7
8	9	6	8	9	6	8	9	6	8	9	6	8	9	6	8	9	9
8	9	6	8	9	6	8	9	6	8	9	6	8	9	6	8	9	9

L2 MEM

1	2	3	4	5	6
1	1	1	1	1	1
2	3	2	2	2	2
3	4	5	3	3	3
4	5	6	7	4	4
5	6	7	8	9	5
6	7	8	9	10	11

L1 MEM					
1	2	3	4	5	6
1	1	1	1	1	1
2	3	2	2	2	2
3	4	5	3	3	3
4	5	6	7	4	4
5	6	7	8	9	5
6	7	8	9	10	11

L2 MEM					
1	2	3	4	5	6
1	1	1	1	1	1
2	3	2	2	2	2
3	4	5	3	3	3
4	5	6	7	4	4
5	6	7	8	9	5
6	7	8	9	10	11

		1	1	2	3	1	2	3	4	5	1	2	3	4	5	6	7	1	2	3	4	5	6	7	8	9	1	2	3	4	5	6	7	8	9	10	11	
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	
1	1	1																																				1
1	2	1	1																																			2
2	3	1	1	1																																		3
3	4	1	1	1	1																																	4
1	5		1	1	1	1																																4
2	6		1	1	1	1	1																															5
3	7			1	1	1	1	1																														5
4	8			1	1	1	1	1	1																													6
5	9				1	1	1	1	1	1																												6
1	10					1	1	1	1	1	1																											6
2	11					1	1	1	1	1	1	1																										7
3	12						1	1	1	1	1	1	1																									7
4	13							1	1	1	1	1	1	1																								7
5	14							1	1	1	1	1	1	1	1																							8
6	15								1	1	1	1	1	1	1	1																						8
7	16									1	1	1	1	1	1	1	1																					8
1	17										1	1	1	1	1	1	1	1																				8
2	18										1	1	1	1	1	1	1	1	1																			9
3	19											1	1	1	1	1	1	1	1	1																		9
4	20												1	1	1	1	1	1	1	1	1																	9
5	21													1	1	1	1	1	1	1	1	1																9
6	22													1	1	1	1	1	1	1	1	1	1															10
7	23														1	1	1	1	1	1	1	1	1	1														10
8	24															1	1	1	1	1	1	1	1	1	1													10
9	25																1	1	1	1	1	1	1	1	1	1												10
1	26																	1	1	1	1	1	1	1	1	1	1											10
2	27																	1	1	1	1	1	1	1	1	1	1	1										11
3	28																		1	1	1	1	1	1	1	1		1	1									10
4	29																			1	1	1	1	1	1	1			1	1								9
5	30																				1	1	1	1	1	1				1	1							8
6	31																					1	1	1	1	1					1	1						7
7	32																					1	1	1	1	1					1	1	1					8
8	33																						1	1	1	1					1		1	1				7
9	34																							1	1	1					1			1	1			6
10	35																								1	1					1				1	1		5
11	36																									1					1					1	1	4

Cycle	Name	R1	R2	R3	R4	R5	R6	R7	R8	R9	R10	R11
0	11											
1	21	11										
2	22	21	11									
3	23	22	21	11								
4	31	23	22	21	11							
5	32	31	23	22	21							
6	33	32	31	23	22	21						
7	34	33	32	31	23	22						
8	35	34	33	32	31	23	22					
9	41	35	34	33	32	31	23					
10	42	41	35	34	33	32	31					
11	43	42	41	35	34	33	32	31				
12	44	43	42	41	35	34	33	32				
13	45	44	43	42	41	35	34	33				
14	46	45	44	43	42	41	35	34	33			
15	47	46	45	44	43	42	41	35	34			
16	51	47	46	45	44	43	42	41	35			
17	52	51	47	46	45	44	43	42	41			
18	53	52	51	47	46	45	44	43	42	41		
19	54	53	52	51	47	46	45	44	43	42		
20	55	54	53	52	51	47	46	45	44	43		
21	56	55	54	53	52	51	47	46	45	44		
22	57	56	55	54	53	52	51	47	46	45	44	
23	58	57	56	55	54	53	52	51	47	46	45	
24	59	58	57	56	55	54	53	52	51	47	46	
25	61	59	58	57	56	55	54	53	52	51	47	
26	62	61	59	58	57	56	55	54	53	52	51	
27	63	62	61	59	58	57	56	55	54	53	52	51
28	64	63	62		59	58	57	56	55	54	53	52
29	65	64	63			59	58	57	56	55	54	53
30	66	65	64				59	58	57	56	55	54
31	67	66	65					59	58	57	56	55
32	68	67	66	65				55	59	58	57	56
33	69	68	67		65				56	59	58	57
34	610	69	68			65				57	59	58
35	611	610	69				65				58	59
36		611	610					65			59	

	左上	左邊	上面
第一個	N/A	m=2(n-1)-1	N/A
縱向	2(n-1)	m=2(n-1)-1	1
縱向最後一個	2(n-1)	k	1
個數=round	N/A	N/A	2(n-1)
橫向	2(n-1)+1	1	2(n-1)
角落	2(n-1)+1	1	n

The End