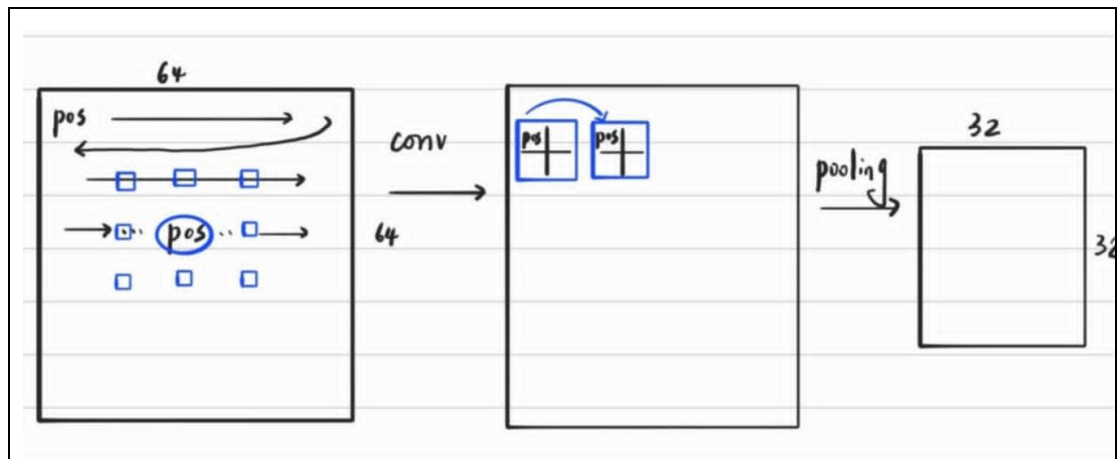


2025 Digital IC Design Homework 4

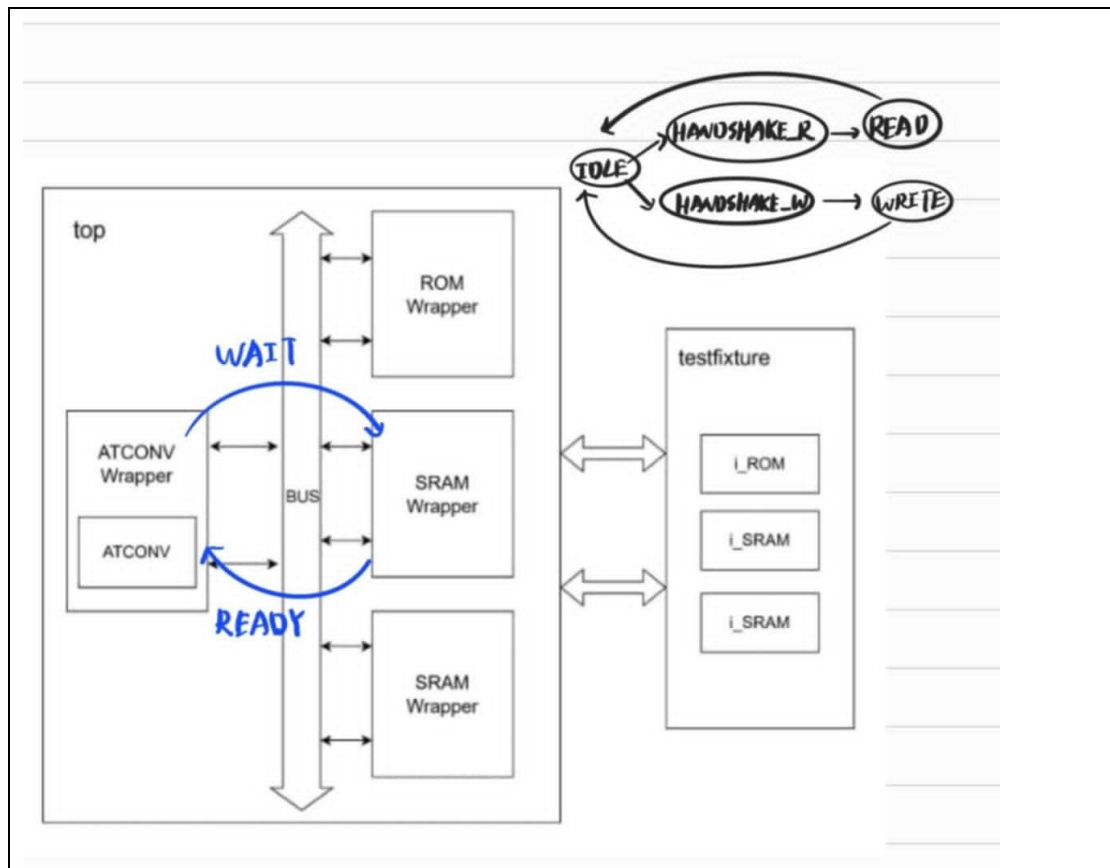
NAME	陳廷睿		
Student ID	F74101034		
ATCONV Simulation Result			
Functional simulation	Pass/Fail	Pre-Layout simulation	Pass/Fail
<pre>----- SUMMARY ----- Congratulations! Layer 0 data have been generated successfully! The result is PASS!! Congratulations! Layer 1 data have been generated successfully! The result is PASS!! terminate at 55301 cycle ----- ** Note: \$finish : C:/Users/F74101034/Downloads/2025_lab4/ATCONV/testfixture.sv(224) Time: 2765050 ns Iteration: 0 Instance: /testfixture</pre>		<pre>----- SUMMARY ----- Congratulations! Layer 0 data have been generated successfully! The result is PASS!! Congratulations! Layer 1 data have been generated successfully! The result is PASS!! terminate at 55301 cycle ----- ** Note: \$finish : C:/Users/F74101034/Downloads/2025_lab4/ATCONV/testfixture.sv(224) Time: 2765058129 ps Iteration: 0 Instance: /testfixture</pre>	
System Simulation Result			
Functional simulation	Pass/Fail	Pre-Layout simulation	Pass/Fail
<pre>----- SUMMARY ----- Congratulations! Layer 0 data have been generated successfully! The result is PASS!! Congratulations! Layer 1 data have been generated successfully! The result is PASS!! terminate at 141317 cycle ----- ** Note: \$finish : C:/Users/F74101034/Downloads/2025_lab4/System/testfixture.sv(228) Time: 7065850 ns Iteration: 0 Instance: /testfixture</pre>		<div>(y)</div> <pre>----- SUMMARY ----- Congratulations! Layer 0 data have been generated successfully! The result is PASS!! Congratulations! Layer 1 data have been generated successfully! The result is PASS!! terminate at 141317 cycle ----- ** Note: \$finish : C:/Users/F74101034/Downloads/2025_lab4/System/testfixture.sv(228) Time: 7065859281 ps Iteration: 0 Instance: /testfixture</pre>	
ATCONV Synthesis Result			
Total logic elements	291		
Total memory bits	0		
Total registers	108		
Embedded multiplier 9-bit elements	2		
Total Cycle used	55301		



第一步先透過 pos 由左至右由上至下遍歷整張圖，每個點都對其做 convolution。做 convolution 時分別分成 9 個 cycle，每個 cycle 都取一個一點和 kernel 相乘，最後存進 layer0。

第二步做 max pooling，pos 由左至右由上至下每次向右兩格遍歷圖片，每次分成四個 cycle 從 layer0 獲得資料，最後保留四個中最大者並存進 layer1。

System Synthesis Result	
Total logic elements	444
Total memory bits	0
Total registers	170
Embedded multiplier 9-bit elements	2
Total Cycle used	141317



Description of your design

Convolution 與第一部份一樣，只是多加了一個 handshake 的狀態。所以在傳送資料之前，我都加上一個 WAIT 的狀態，在第一個 CYCLE 將 VALID 拉高並等待 READY，等到 READY 後將 VALID 拉回 1，並在下一個 CYCLE 開始給資料或是讀資料。

記憶體的部分則是先在 IDLE 等待，如果收到 RVALID 進入 HANDSHAKE_READ 的狀態，此狀態回傳 READ_READY，下一個 CYCLE 後開始讀資料，WRITE 也是一樣方式。