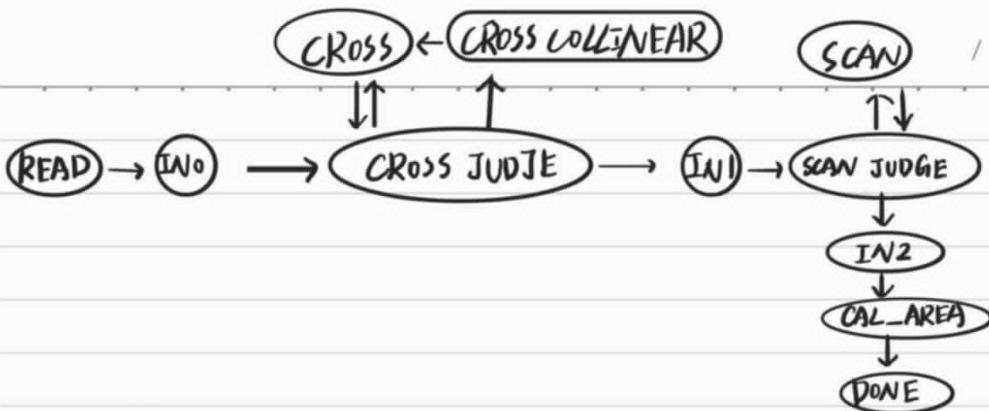


## 2025 Digital IC Design Homework 5

NAME	陳廷睿		
Student ID	F74101034		
<b>Simulation Result</b>			
Functional simulation	Pass/Fail	Pre-Layout simulation	Pass/Fail
All 10 patterns passed! Cycle: 2985  ** Note: \$finish : C:/Users/admin/Downloads/2025lab52/testfixture.sv(82) Time: 65681 ns Iteration: 0 Instance: /testfixture 1			<pre>             All 10 patterns passed!                          Cycle: 2987                         ** Note: \$finish : C:/Users/admin/Downloads/2025lab52/testfixture.sv(82)               Time: 65711529 ps Iteration: 0 Instance: /testfixture         </pre>
<b>Synthesis Result</b>			
Total logic elements	2843		
Total memory bits	0		
Total registers	778		
Embedded multiplier 9-bit elements	2		
Clock period (ns)	22		
Total Cycle used	2987		
先將 20 個 input 在 READ 狀態內讀入，同時將最左下方點設定為錨點，在之後透過外積判斷與錨點夾角並由小到大排序(CROSS, CROSS_JUDGE,CROSS_COLLINEAR 三個狀態)。 排序好後判斷下一個點是否為 left turn，若果是的話加入 stack，否則就持續 pop 直到三點為 left turn(SCAN,SCAN_JUDGE 兩個狀態)。做完 20 點後開始計算面積，透過公式分批計算後輸出。			



### Description of your design

將上述的步驟說明更清楚:

READ 會在讀取時判斷新點是否為最坐下點，如果是紀錄於陣列 0 位置，否則一陣列順序繼續填入。

IN0 會將第一組要判斷順逆時針的頂點放入，透過 combinational 電路計算出結果

依據 insertion sort 排序

CROSS JUDGE 透過正負判斷是否需要交換，如果為 0 則代表共線，會需要比較何者距離錨點較遠，並在 CROSS COLLINEAR 狀態較遠點排序放後面。

CROSS 會接續放入頂點，透過 combinational 電路計算出結果，提供 CROSS\_JUDGE 判斷。

IN1 放入第一組要計算是否為 left turn 的頂點。

SCAN\_JUDGE 判斷是否為 left turn，並移動 Stack top 指標

SCAN 會接續放入頂點，透過 combinational 電路計算出結果，提供 SCAN\_JUDGE 判斷。

IN2 放入第一組要計算的頂點面積

CAL 會在放入下點要計算面積的同時，累加前項面積結果。

DONE 輸出面積並將狀態回到 READ