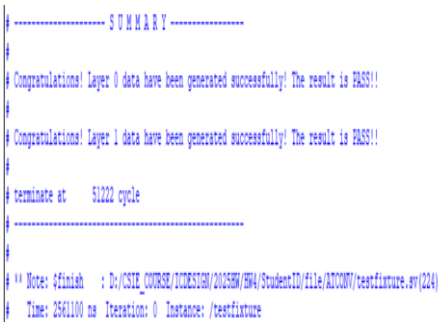
# 2025 Digital IC Design Homework 4

| NAME | 傅信豪 |
|---|---|
| Student ID | NE6121084 |

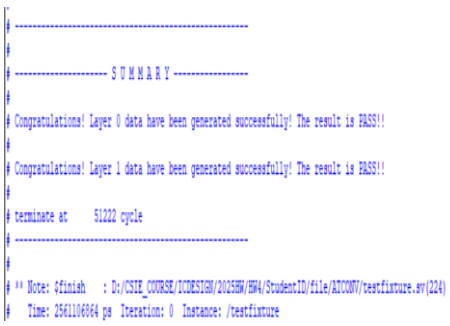
**ATCONV Simulation Result**

| Functional simulation | Pass | Pre-Layout simulation | Pass |
|---|---|---|---|

(your functional sim result)



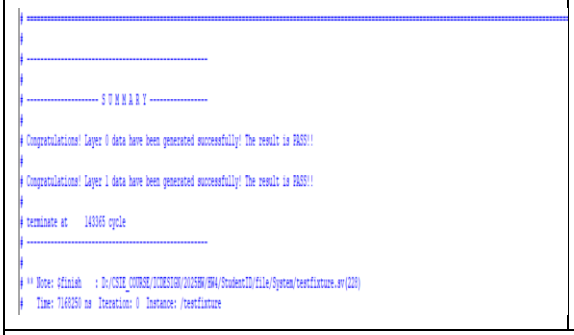

(your pre-sim result)



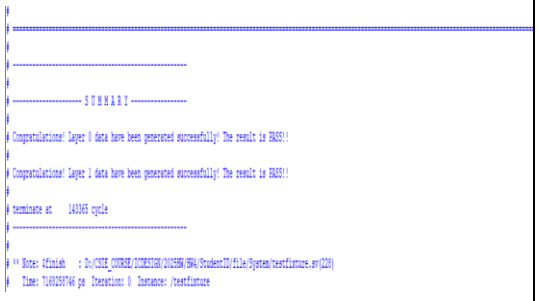

**System Simulation Result**

| Functional simulation | Pass | Pre-Layout simulation | Pass |
|---|---|---|---|

(your functional sim result)




(your pre-sim result)




**ATCONV Synthesis Result**

| | |
|---|---|
| Total logic elements | 607 |
| Total memory bits | 0 |
| Total registers | 246 |
| Embedded multiplier 9-bit elements | 0 |
| Total Cycle used | 51222 |

(your flow summary of ATCONV)

**Flow Summary**

<<Filter>>

| | |
|---|---|
| Flow Status | Successful - Thu May 8 14:47:11 2025 |
| Quartus Prime Version | 22.1std.0 Build 915 10/25/2022 SC Lite Edition |
| Revision Name | ATCONV |
| Top-level Entity Name | ATCONV |
| Family | Cyclone IV E |
| Device | EP4CE55F23A7 |
| Timing Models | Final |
| Total logic elements | 607 / 55,856 ( 1 % ) |
| Total registers | 246 |
| Total pins | 124 / 325 ( 38 % ) |
| Total virtual pins | 0 |
| Total memory bits | 0 / 2,396,160 ( 0 % ) |
| Embedded Multiplier 9-bit elements | 0 / 308 ( 0 % ) |
| Total PLLs | 0 / 4 ( 0 % ) |

**System Synthesis Result**

| | |
|---|---|
| Total logic elements | 815 |
| Total memory bits | 0 |
| Total registers | 375 |
| Embedded multiplier 9-bit elements | 0 |
| Total Cycle used | 143365 |

(your flow summary of System)

**Flow Summary**

<<Filter>>

| | |
|---|---|
| Flow Status | Successful - Wed May 14 01:02:05 2025 |
| Quartus Prime Version | 22.1std.0 Build 915 10/25/2022 SC Lite Edition |
| Revision Name | top |
| Top-level Entity Name | top |
| Family | Cyclone IV E |
| Device | EP4CE55F23A7 |
| Timing Models | Final |
| Total logic elements | 815 / 55,856 ( 1 % ) |
| Total registers | 375 |
| Total pins | 124 / 325 ( 38 % ) |
| Total virtual pins | 0 |
| Total memory bits | 0 / 2,396,160 ( 0 % ) |
| Embedded Multiplier 9-bit elements | 0 / 308 ( 0 % ) |
| Total PLLs | 0 / 4 ( 0 % ) |

**Description of your design**

## ATCONV 模組設計說明

### 狀態機

1. IDLE
2. LOAD_FROM_ROM
3. PROCESS
4. SAVE_TO_LAYER0
5. LOAD_FROM_LAYER0
6. MAXPOOLING
7. SAVE_TO_LAYER1
8. DONE

### 狀態轉移邏輯

- 從 IDLE → LOAD_FROM_ROM : 一啟動即進入
- LOAD_FROM_ROM → PROCESS : kernel_cnt 達到 8，即已讀滿 9 筆）
- SAVE_TO_LAYER0 → LOAD_FROM_LAYER0 : kernel_center 達到 4096，即掃完 64×64）
- LOAD_FROM_LAYER0 → MAXPOOLING : pool_read_cnt 達到 3，即已讀滿 4 筆，2*2 大小
- MAXPOOLING → SAVE_TO_LAYER1 : 計算完畢即寫回
- SAVE_TO_LAYER1 → DONE : layer1_wrt_addr 達到 1024，即寫完所有池化位置
- DONE → DONE : 保持結束

## SYSTEM 模組設計說明

**1. ATCONV Wrapper**

**A.** **功能**：串接至 BUS，依序向 ROM 讀取卷積核、執行卷積+ReLU、向 Layer0／Layer1 SRAM 寫入與讀出。

**B.** **狀態機**

**i.** IDLE → WAIT_ROM_READ_READY：啟動後進行 ROM 讀通道建立

**ii.** WAIT_ROM_READ_READY → LOAD_FROM_ROM：當 BUS RVALID_M & RREADY_M 同步後進入讀取

**iii.** LOAD_FROM_ROM → PROCESS：收到 RLAST_M（單筆

burst）且 kernel_cnt 達 8

iv. PROCESS → WAIT_LAYER0_WRITE_READY：卷積+ReLU 完成，準備寫 Layer0

v. WAIT_LAYER0_WRITE_READY → SAVE_TO_LAYER0：手握 WVALID_M & WREADY_M 建立寫通道

vi. SAVE_TO_LAYER0 → WAIT_LAYER0_READ_READY 或 WAIT_ROM_READ_READY：kernel_center 滿格切換讀／寫

vii. WAIT_LAYER0_READ_READY → LOAD_FROM_LAYER0：建立 Layer0 讀通道

viii. LOAD_FROM_LAYER0 → MAXPOOLING：pool_read_cnt 滿 3

ix. MAXPOOLING → WAIT_LAYER1_WRITE_READY：計算完畢

x. WAIT_LAYER1_WRITE_READY → SAVE_TO_LAYER1：建立 Layer1 寫通道

xi. SAVE_TO_LAYER1 → DONE 或 LOAD_FROM_LAYER0：層寫完切換或迴圈

xii. DONE：結束，輸出 done=1

C. **握手訊號**（AXI-like）

i. master→slave：RVALID_M/WVALID_M + ID_M 拉起後等待 BUS 回傳 RREAD/WREADY

ii. 當成功建立傳輸時進入 LOAD(READ)或是 SAVE(WRITE)狀態進行傳輸，READ 狀態當收到 RLAST 就結束讀取狀態，而 WRITE 狀態就等傳出最後一筆時傳送 WLAST 通知 RAM 結束寫入狀態

**2. Bus**

A. **功能**：橋接 ATCONV (master) 與三個 slave (ROM、SRAM0、SRAM1)，根據通道 ID 選擇合法的 R/W 通道。

B. **無狀態機：** 單純根據 ID 建立傳送路徑，跟傳遞訊息，實際 R/W 由 master/slave 去各自完成

**3. Rom Wrapper**

A. **功能**：接受 BUS 發出的讀取請求，產生 ROM 讀取控制與回應資料。

B. **狀態機**

i. IDLE：等待 RVALID_S=1

ii. READ：連續發出 ROM_rd=1，地址 ROM_A 從 ADDR_S 起逐拍遞增，回傳 RDATA_S=ROM_Q，RLAST_S=1 於最後一拍。

**C. 握手協定**

i. BUS→Wrapper：收到 RVALID_S 下一刻進入傳送階段

ii. Wrapper→BUS：回傳 RREADY 給 mater，當 master 收到 RREADY 就會進入讀取狀態

**4. RAM Wrapper**

**A. 功能**：雙向支援對 SRAM 的讀／寫，配合 BUS 完成 burst 讀寫。。

**B. 狀態機**

i. IDLE：無效或等待 RVALID_S／WVALID_S

ii. READ：當 RVALID_S=1，回應 RDATA_S=SRAM_Q 並在最後一拍輸出 RLAST_S

iii. WRITE：當 WVALID_S=1，接收 WDATA_S 至 SRAM_D，在最後一拍偵測 WLAST_S 下一階段回到 IDLE 狀態

**C. 握手協定**

i. BUS→Wrapper：收到 RVALID/WVALID 下一刻進入資料傳送階段

ii. Wrapper→BUS：回傳 RREADY/WREADY 給 mater，當 master 收到 RREADY/WREADY 就會進入讀取或是開始傳送數據