# 2023 Digital IC Design Homework 4

| NAME | 黃彥承 |
|---|---|
| Student ID | N26101185 |

**Simulation Result**

| Functional simulation | 100 | Gate-level simulation | 100 |
|---|---|---|---|

```
#
# ---------------------- S U M M A R Y ----------------
#
# 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      87048 cycle
# -----------------------------------------------------
```

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

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      87048 cycle
-----------------------------------------------------
```

**Synthesis Result**

| | |
|---|---|
| Total logic elements | 645 |
| Total memory bits | 0 |
| Embedded multiplier 9-bit elements | 0 |
| Total cycle used | 87048 |

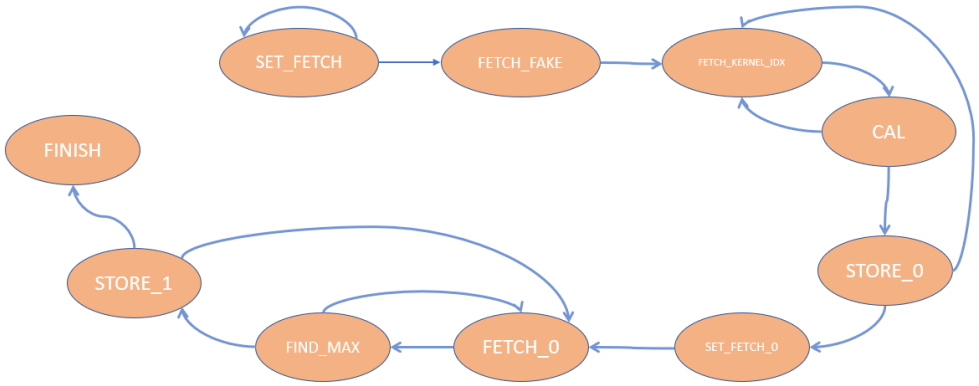
**Flow Summary**

<<Filter>>

| | |
|---|---|
| Flow Status | Successful - Wed May 17 20:11:01 2023 |
| Quartus Prime Version | 20.1.1 Build 720 11/11/2020 SJ Lite Edition |
| Revision Name | ATCONV |
| Top-level Entity Name | ATCONV |
| Family | Cyclone IV E |
| Device | EP4CE55F23A7 |
| Timing Models | Final |
| Total logic elements | 645 / 55,856 ( 1 % ) |
| Total registers | 113 |
| Total pins | 82 / 325 ( 25 % ) |
| 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**

The proposed work is implemented using a finite state machine (FSM) that is divided into ten states. The state diagram is described below:




SET_FETCH: Sets all parameters' initial value.
FETCH_FAKE: Updates layer 0 index.
FETCH_KERNEL_IDX: Gets 3*3 kernel's corresponding index.
CAL: Calculates the convolutional value.
STORE_0: Stores convolutional value to layer 0 according to layer 0 index.
SET_FETCH_0: Sets value to get layer 0 value.
FETCH_0: Gets layer 0 according to layer 0 index.
FIND_MAX: Finds the maximum value.
STORE_1: Stores the maximum value to layer 1 according to layer 1 index.
FINISH: Finish by setting the busy signal to 0.

*Scoring = (Total logic elements + Total memory bits + 9*Embedded multipliers 9-bit elements) X Total cycle used*

*** Total logic elements must not exceed 1000.**