NYCU-EE IC LAB – FALL 2023

Lab08 Exercise

Design: Low Power on Siamese Neural Network (from Lab04)

Data Preparation

1. Extract files from TA's directory:
   % tar xvf ~iclabTA01/Lab08.tar
2. The extracted LAB directory contains:
   a. **EXERCISE/**
   b. **EXERCISE_wocg/**
   c. **PRACTICE/**
   d. **JG/**

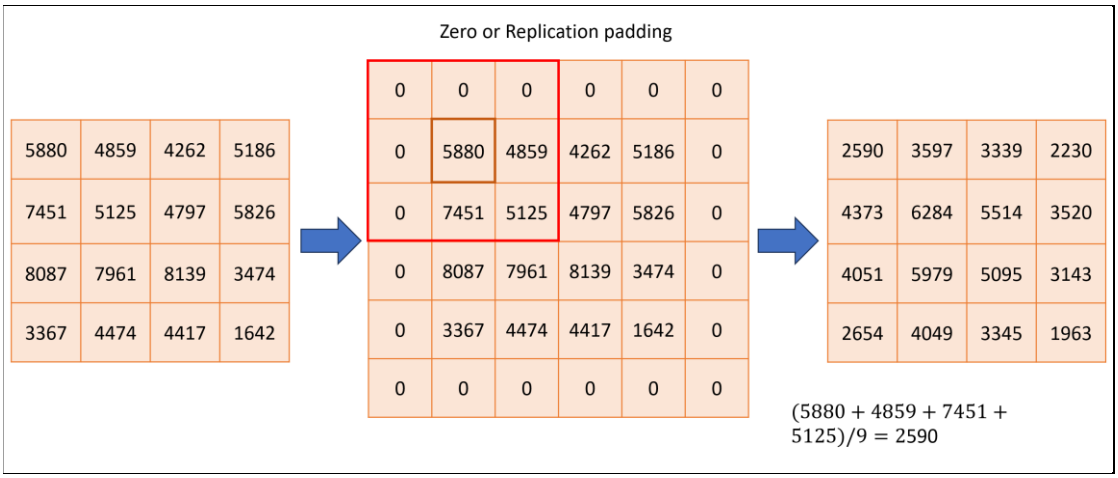
Design Description

You need to employ low-power design techniques on SNN (from Lab04). You need to complete the following things:

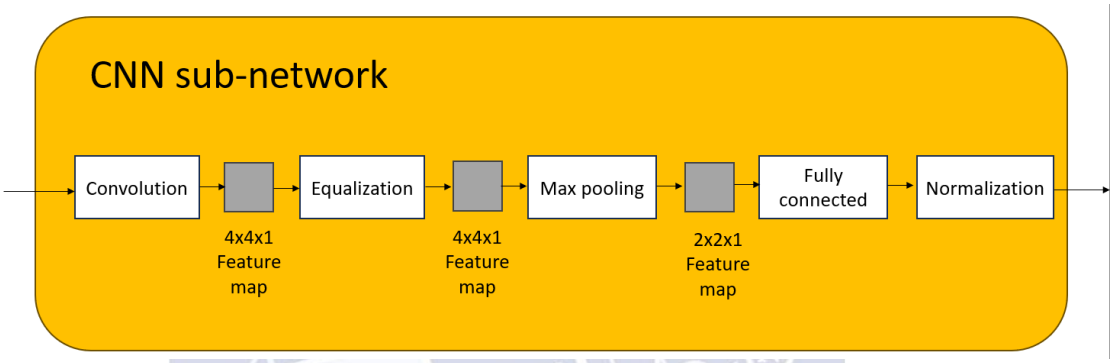
1. There will be two versions of the design module.
   - Stage 1:
     At EXERCISE/SNN.v, you should design an SNN module with a clock gating cell as below figure. The main difference part is the clock gating cell and cg_en input signal, you can decide your own clock gating states.
     Stage 2:
     At EXERCISE_wocg/SNN_wocg.v, which means SNN without clock gating.
2. The only part different from Lab04 is the CNN sub-network. You need to add an equalization state between convolution and max pooling. The padding also depends on the Opt signal. The following figure demonstrates the operation based on zero padding.
3. Run the JasperGold SEC to ensure the gated and non-gated designs have the same output logic.

Equalization

This example uses round-down output for demonstration purposes, please do not round down, the output format still follows the IEEE-754 floating point format.



Modified design for this lab



Inputs

I/O	Signal name	Bit Width	Description
Input	**clk**	1	Clock
Input	**rst_n**	1	Asynchronous active low reset
Input	**cg_en**	1	If cg_en is high, the series processing blocks should execute clock gating. Otherwise, if cg_en is low, the image processing blocks follow **clk**.
Input	**Img**	32	The kernel signals which sent in raster ordering. The arithmetic representation follows the IEEE-754 floating number format.
Input	**Kernel**	32	The kernel signals which sent in raster ordering. The arithmetic representation follows the IEEE-754 floating number format.
Input	**Weight**	32	The weight signals which sent in raster ordering. The arithmetic representation follows the IEEE-754 floating number format
Input	**Opt**	2	2'd0 : Sigmoid & {Replication} 2'd1 : Sigmoid & {Zero} 2'd2 : tanh & {Replication} 2'd3 : tanh & {Zero}

◆ Outputs

I/O	Signal name	Bit Width	Description
output	**out_valid**	1	Should be set to low after reset**.** Should be set to high when out_data is valid. Should be set to low when your **out_data** is invalid. Also, should be set to low even if you pass all of the patterns.
output	**out_data**	32	Should be set to low after reset**.** The output signals of the image. The arithmetic representation follows the IEEE-754 floating number format.

Specifications

1. Top module name：SNN (File name: SNN.v)
2. After converting to float number, You have to check an error under 0.01 for the result. If the error is higher than the value, you will fail this lab.
3. Use **asynchronous** reset active low architecture.
4. All your output registers should be set to zero after reset.
5. The instance name of the clock gating cell (GATED_OR) should be GATED_XXX (e.g., GATED_out).
6. After synthesis, check the “SNN.area” and “SNN.timing” in the folder “Report”. The area report is valid only when the slack at the end of “SP.timing” is **non-negative** and the result should be **MET**.
7. The synthesis result **cannot** contain any **LATCH except for the clock gating latch.**
8. The synthesis result **cannot** contain any errors**.**
9. Input delay and output delay are 0.5*Clock Period. And the Output loading is set to 0.05.
10. The execution latency is limited to 1000 cycles. The latency is the clock cycle between the falling edge of the in_valid and the rising edge of the first out_valid.
11. The area is limited to 7000000. Also, the synthesis time should be less than 4 hours in both designs.
12. You can adjust your clock period by yourself, but the maximum period is 70 ns. The precision of the clock period is 0.1, The clock period must be the same in SNN.v and SNN_wocg.v.
13. You can’t have timing violations in gate-level simulation.
14. Your design **can’t use memory** in this lab.
15. The **out_valid** cannot overlap with **in_valid.**
16. Your design should have **at least 20% power reduction** from **cg_en-off to cg_en-on**,

otherwise it will be treated as failed. $\frac{P_{cg_{enoff}} - P_{cg_{enon}}}{P_{cg_{enoff}}} \geq 20\%$

17. Don't use any wire/reg/submodule/parameter name called *error*, *congratulation*, *latch* or *fail* otherwise you will fail the lab. Note: * means any char in front of or behind the word. e.g: error_note is forbidden.
18. Don' t write Chinese comments or other language comments in the file you turned in.
19. **Any error messages during synthesis and simulation, regardless of the result will lead to failure in this lab.**
20. You can change the pattern number for a more accurate power estimate but be aware of the **storage capacity of the fsdb file.**
21. **You need to synthesize the Gated_or module first, then synthesize the gated design.**
22. **Power report position: 04_PTPX/Report/SNN_POWER or SNN_CG_POWER**

Report Total Power Example:

SNN_POWER (w/o clock gating)

```
  Net Switching Power  =  1.135e-03   (22.33%)
  Cell Internal Power  =  3.933e-03   (77.40%)
  Cell Leakage Power   =  1.348e-05   ( 0.27%)
      Intrinsic Leakage  =  1.348e-05
      Gate Leakage       =     0.0000
                         ---------
Total Power            =  5.082e-03   (100.00%)

X Transition Power     =  1.015e-05
Glitching Power        =     0.0000

Peak Power             =     0.0428
Peak Time              =      16680
```

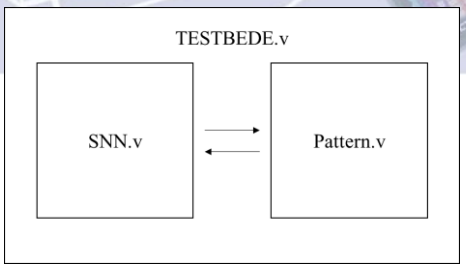
SNN_CG_POWER (with clock gating)

```
  Net Switching Power  =  1.013e-03   (28.28%)
  Cell Internal Power  =  2.556e-03   (71.35%)
  Cell Leakage Power   =  1.348e-05   ( 0.38%)
      Intrinsic Leakage  =  1.348e-05
      Gate Leakage       =     0.0000
                         ---------
Total Power            =  3.583e-03   (100.00%)

X Transition Power     =  1.015e-05
Glitching Power        =     0.0000

Peak Power             =     0.0373
Peak Time              =      16680
```

Block Diagram



Grading Policy

Functionality (80%):

- SNN and SNN_wocg : 65%
  (You can only get these points if you pass the gate-level simulation both)
- JasperGold SEC check (15%)：**Run1 (7.5%) , Run2 (7.5%)**

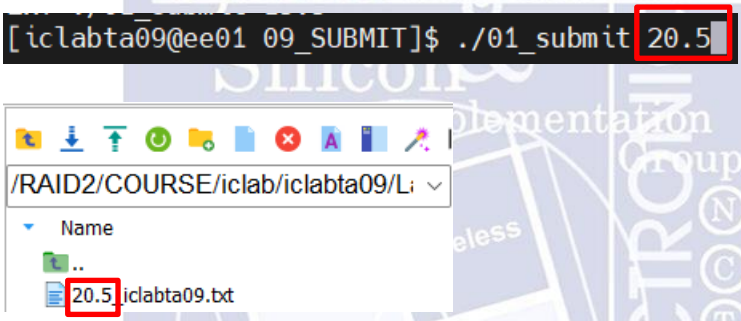
(You can only get these points if you pass the gate-level simulation. No 2nd demo chance.)

Performance (20%)：

- (**Total Latency** * **Total Power^2** *(gated with CG)*) * **Area**

Note

1. **Please submit following files under 09_SUBMIT before 12:00 at noon on Nov. 20**
   - SNN, SNN_wocg.v
     If uploaded files violate the naming rule, you will get 5 deducted points.
   - The 2nd demo deadline is **12:00 at noon on Nov. 22**.
   - Check whether there is any wire/reg/submodule name called “error”, “fail”, “pass”, or “congratulation”, if you used it, you will fail the lab.
   - In this lab, you can adjust your clock cycle time. Consequently, make sure to key in your clock cycle time after the command like the figure below. It means that the TA will demo your design under this clock cycle time.



2. **Template folders and reference commands:**

   01_RTL/ (RTL simulation)

   "**./01_run**", "**./02_run_cg**"*(only in EXERCISE folder)*

   02_SYN/ (Synthesis)

   "**./01_run_dc**"

   (Check the design which contains **latch and error** or not in **syn.log**)

   03_GATE_SIM/ (Gate Level simulation)

   "**./01_run**", "**./02_run_cg**"*(only in EXERCISE folder)*

   04_PTPX/

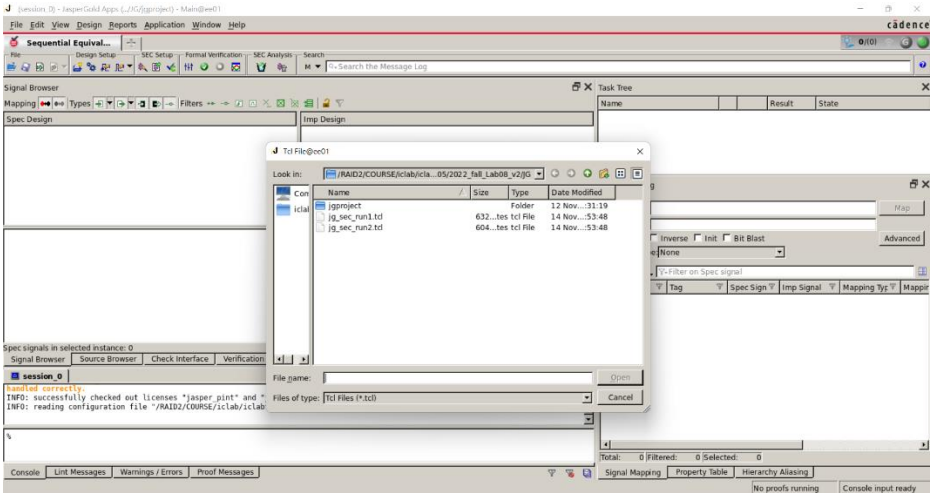
   "**./01_run_ptpx**" & "**./02_run_cg_ptpx**"to get the power of your design.

   ※ You should make sure the three clock period values are identical in 00_TESTBED/Pattern.v && /02_SYN/syn.tcl**:**

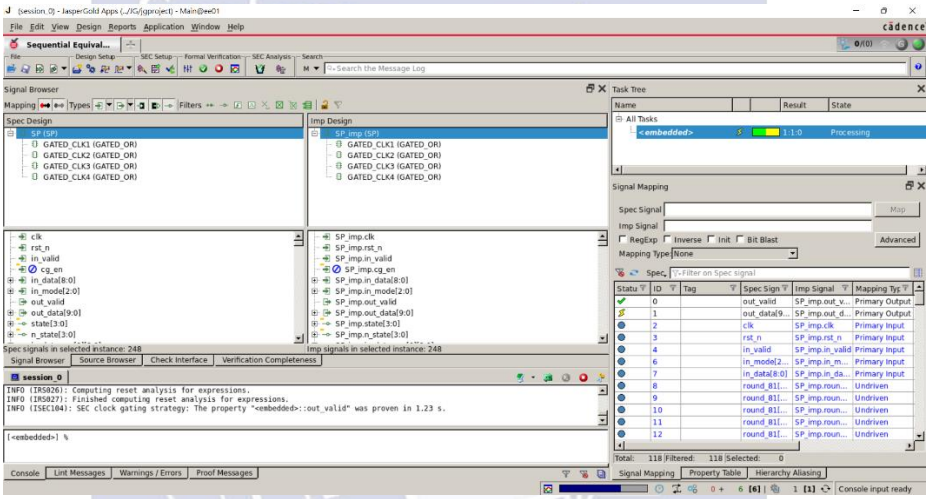
- You can key in ./09_clean_up to clear all log files and dump files in each folder

1. JasperGold SEC execution steps: In folder JG/

(1) **"./01_run"**、**"./02_run"** or "**jg –sec &**" and click the "source" button.



(2) Then, click the "Prove the SEC Properties"



(3) Finally, check all properties pass.