National Cheng Kung University

Department of Electrical Engineering

Introduction to VLSI CAD (Spring 2024)

Lab Session 7

Design of Local Binary Pattern Facial Recognition System

Name	Student ID
游宗謀	E94106151

Practical	Points	Marks
Lab 7_1	30	
Lab 7_2	65	
Report	5	
Demo	10	
Notes		

Due: 15:00 May 1, 2024@ moodle

Summary

Hardware		
TOP	RTL(∨/X)	Synthesis(∨/X)
Lab7_1	∨	∨
Lab7_2	∨	∨

Synthesis result		
Clock period(ns)	Area	Simulation time (ns)
2 ns	4188.620263 um^2	427596000 ns

Superlint(number of inline messages, just write down the final design result, i.e. if you only finish lab7_1, write your Superlint result of lab7_1, otherwise, write down lab7_2 only)			
Total lines	Warning	Error	coverage(%)
1194	0	30	97.487%

Note: You must complete and fill out this form with your design information!!!

Deliverables

1) All Verilog codes including testbenches for each problem should be uploaded.
   NOTE: Please **DO NOT** include source code in the paper report!
2) All homework requirements should be uploaded in this file hierarchy.
3) NOTE: 1. Please **DO NOT** upload waveforms (.fsdb or .vcd)!
4) If you upload a dead body which we can't even compile, you will get NO credit!
5) All Verilog file should get at least 90% SuperLint Coverage.
6) All homework requirements should be uploaded in this file hierarchy or you will not get full credit, if you want to use some sub modules in your design but you do not include them in your tar file, you will get 0 point.

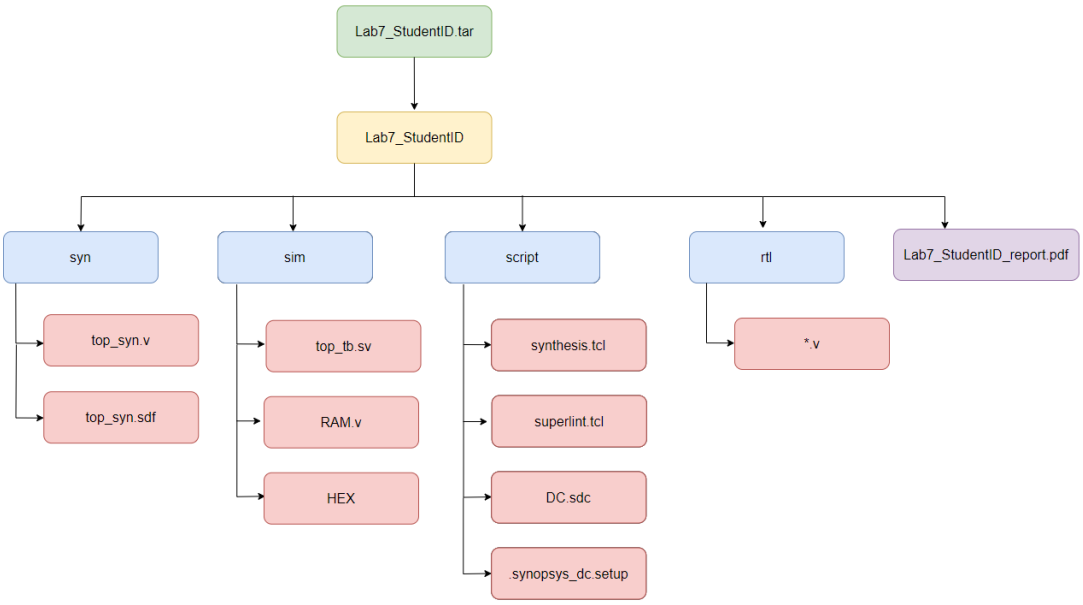
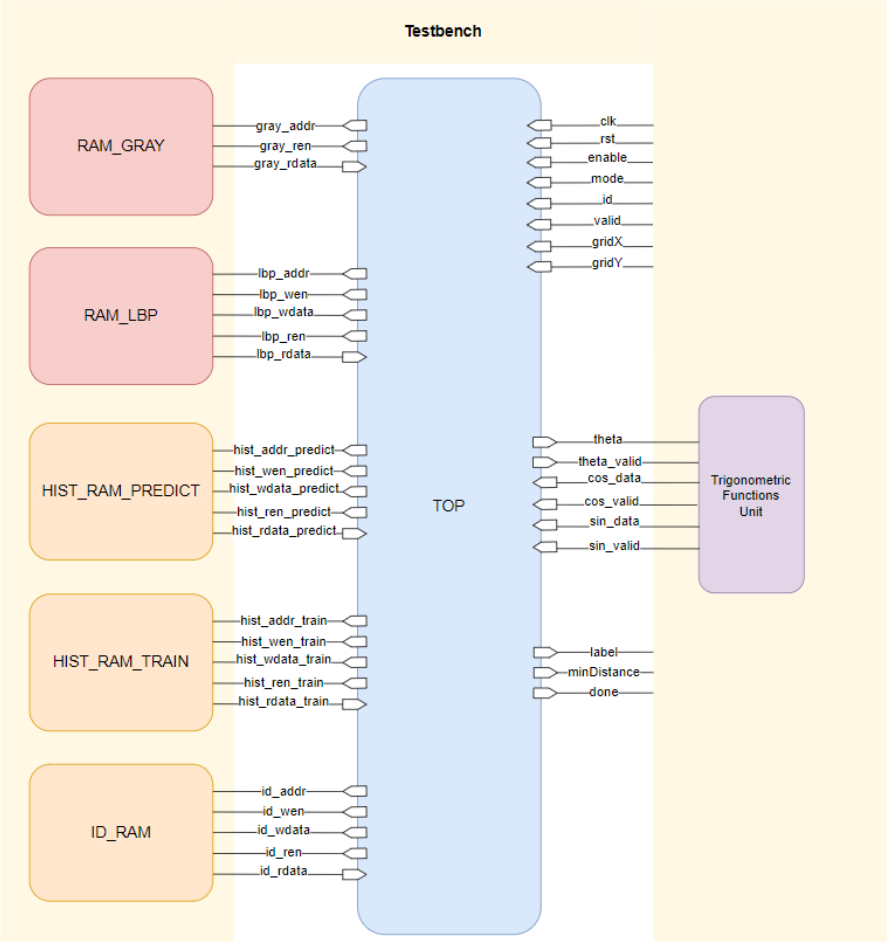
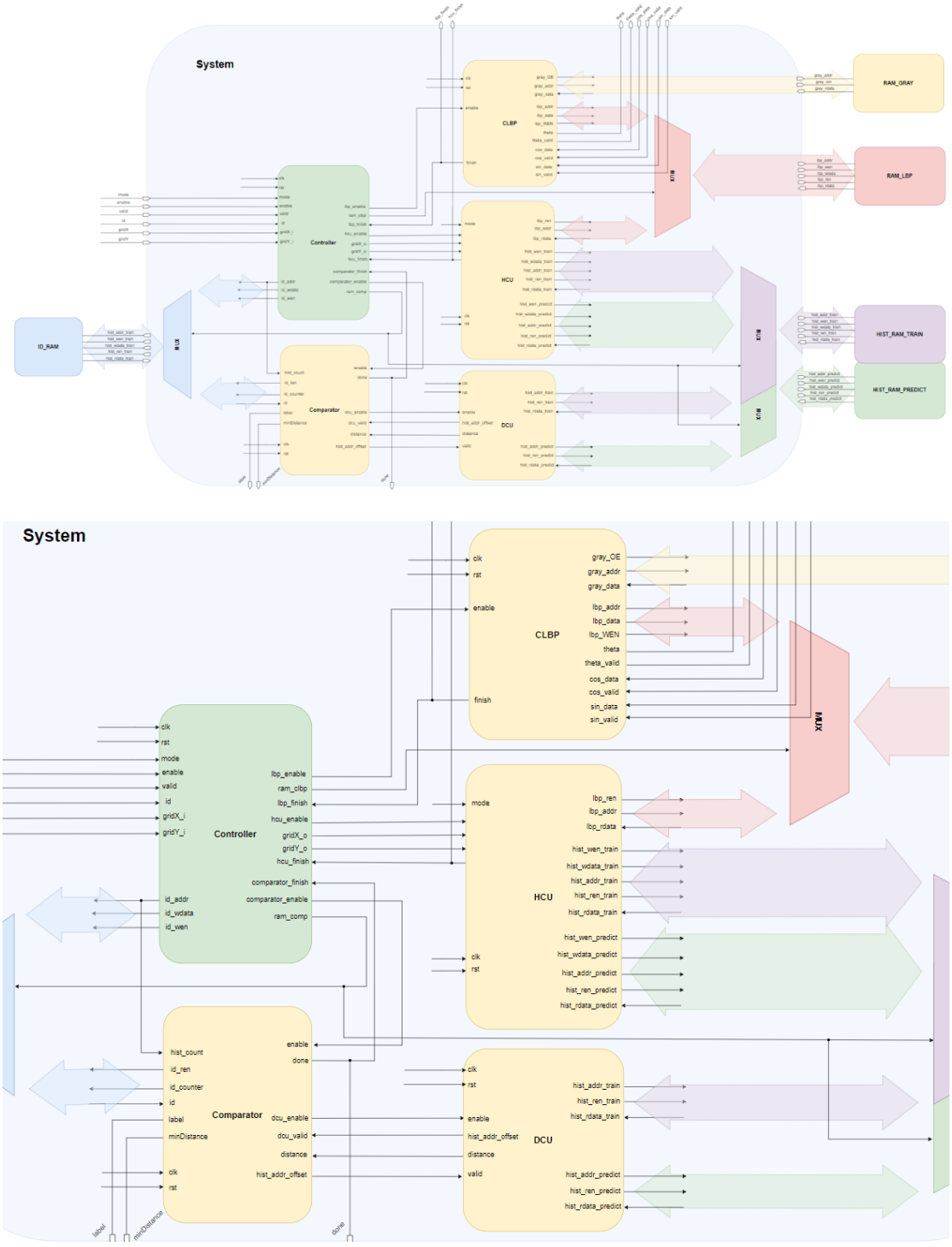


Fig.1 File hierarchy for Homework submission

You are about to integrate all components (CLBP, HCU, Controller…) to form a LBP facial recognition system. The block diagram of system is as shown in **Fig2** and **Fig3**.



▲Fig2. The block diagram of system (external)



▲Fig3. The block diagram of system (internal)

- **Port list of top module:**
- TOP

Signal	I/O	Bit-width	Description
clk	I	1	Clock signal
rst	I	1	Reset signal
enable	I	1	Circuit enabling signal
mode	I	1	Indication signal of whether the system is in prediction or training phase, 0 is training, 1 is prediction
gridX	I	4	Image sliced portion in X direction, value is 8
gridY	I	4	Image sliced portion in Y direction, value is 8
valid	I	1	Indication that current subject ID is valid
id	I	5	Subject ID
hcu_finish	O	1	Indication that HCU circuit is done(should be asserted every time a subject picture is finished computing histogram)
label	O	5	Prediction result, output ID value
minDistance	O	18	Minimum distance between the prediction histogram and the closest histogram in HIST_TRAIN_RAM
done	O	1	Indication that prediction of one picture is finished

Signal	I/O	Bit-width	Description
gray_addr	O	12	Address signal connected to RAM_GRAY
gray_ren	O	1	Read enable signal to RAM_GRAY
gray_rdata	I	8	Read data signal from RAM_GRAY
lbp_addr	O	12	Address signal connected to RAM_LBP
lbp_wen	O	1	Write enable signal to RAM_LBP
lbp_wdata	O	8	Write data signal to RAM_LBP
lbp_ren	O	1	Read enable signal to RAM_LBP
lbp_rdata	I	8	Read data signal from RAM_LBP
theta	O	25(fixed-point)	Current neighbor's theta signal(unit is in radian)
theta_valid	O	1	Indication signal of current neighbor's theta is valid
cos_data	I	25(fixed-point)	Cosine value of the theta(from testbench)
cos_valid	I	1	Indication signal of cosine value is valid
sin_data	I	25(fixed-point)	Sine value of the theta(from testbench)
sin_valid	I	1	Indication signal of sine value is valid
lbp_finish	O	1	Indication signal of the CLBP circuit is finished

Signal	I/O	Bit-width	Description
id_addr	O	8	Address signal connected to ID_RAM
id_ren	O	1	Read enable signal to ID_RAM
id_rdata	I	5	Read data signal from ID_RAM
id_wen	O	1	Write enable signal to ID_RAM
id_wdata	O	5	Write data signal to ID_RAM
hist_addr_train	O	21	Address signal connected to HIST_RAM_TRAIN
hist_wen_train	O	1	Write enable signal to HIST_RAM_TRAIN
hist_wdata_train	O	8	Write data signal to HIST_RAM_TRAIN
hist_ren_train	O	8	Read enable signal to HIST_RAM_TRAIN
hist_rdata_train	I	8	Read data signal from HIST_RAM_TRAIN
hist_addr_predict	O	21	Address signal connected to HIST_RAM_PREDICT
hist_wen_predict	O	1	Write enable signal to HIST_RAM_PREDICT
hist_wdata_predict	O	8	Write data signal to HIST_RAM_PREDICT
hist_ren_predict	O	8	Read enable signal to HIST_RAM_PREDICT
hist_rdata_predict	I	8	Read data signal from HIST_RAM_PREDICT

- **Port list of each module:**
- CLBP

Signal	I/O	Bit-width	Description
clk	I	1	Clock signal
rst	I	1	Reset signal
enable	I	1	CLBP circuit enabling signal
gray_addr	O	12	Address signal connected to RAM_GRAY
gray_OE	O	1	Read enable signal to RAM_GRAY
gray_data	I	8	Read data signal from RAM_GRAY
lbp_addr	O	12	Address signal connected to RAM_LBP MUX to memory
lbp_WEN	O	1	Write enable signal to RAM_LBP
lbp_data	O	8	Write data signal to RAM_LBP
theta	O	25(fixed-point)	Current neighbor's theta signal(unit is in radian)
theta_valid	O	1	Indication signal of current neighbor's thetas is valid
cos_data	I	25(fixed-point)	Cosine value of the theta (from testbench)
cos_valid	I	1	Indication signal of cosine value is valid
sin_data	I	25(fixed-point)	Sine value of the theta(from testbench)
sin_valid	I	1	Indication signal of sine value is valid
finish	O	1	Indication signal of the LBP circuit is finished

➢ HCU

Signal	I/O	Bit-width	Description
clk	I	1	Clock signal
rst	I	1	Reset signal
mode	I	1	Indication signal of whether the system is in prediction or training phase, 0 is training, 1 is prediction
enable	I	1	HCU circuit enabling signal
gridX	I	4	Image sliced portion in X direction, value is 8
gridY	I	4	Image sliced portion in Y direction, value is 8
lbp_addr	O	12	Address signal connected to RAM_LBP MUX to memory
lbp_ren	O	1	Read enable signal to RAM_LBP
lbp_rdata	I	8	Read data signal from RAM_LBP

Signal	I/O	Bit-width	Description
hist_addr_train	O	21	Address signal connected to HIST_RAM_TRAIN MUX to memory
hist_wen_train	O	1	Write enable signal to HIST_RAM_TRAIN
hist_wdata_train	O	8	Write data signal to HIST_RAM_TRAIN
hist_ren_train	O	8	Read enable signal to HIST_RAM_TRAIN MUX to memory
hist_rdata_train	I	8	Read data signal from HIST_RAM_TRAIN
hist_addr_predict	O	21	Address signal connected to HIST_RAM_PREDICT MUX to memory
hist_wen_predict	O	1	Write enable signal to HIST_RAM_PREDICT
hist_wdata_predict	O	8	Write data signal to HIST_RAM_PREDICT
hist_ren_predict	O	8	Read enable signal to HIST_RAM_PREDICT MUX to memory
hist_rdata_predict	I	8	Read data signal from HIST_RAM_PREDICT
done	O	1	Indication that HCU circuit is done(should be asserted every time a subject picture is finished computing histogram)

➢ Comparator

Signal	I/O	Bit-width	Description
clk	I	1	Clock signal
rst	I	1	Reset signal
enable	I	1	Comparator circuit enabling signal
histcount	I	8	# IDs encountered during training mode
distance	I	1	DCU computed distance value
dcu_valid	I	1	Indication that the current distance value is valid
id	I	5	Id read data from ID_RAM
id_ren	O	1	Read enable signal to ID_RAM
id_counter	O	8	The current ID address it is processing MUX to memory
dcu_enable	O	1	DCU circuit enabling signal
label	O	5	Prediction result, output ID value
minDistance	O	18	Minimum distance between the prediction histogram and the closest histogram in HIST_TRAIN_RAM
hist_addr_offset	O	21	The address offset in HIST_RAM_TRAIN of the id it is processing currently
done	O	1	Indication signal of the Comparator circuit is finished

➢ Controller

Signal	I/O	Bit-width	Description
clk	I	1	Clock signal
rst	I	1	Reset signal
mode	I	1	Indication signal of whether the system is in prediction or training phase, 0 is training, 1 is prediction
enable	I	1	Comparator circuit enabling signal
valid	I	1	Indication that current subject ID is valid
id	I	5	Subject ID
id_addr	O	8	Address signal connected to ID_RAM MUX to memory
id_wen	O	1	Write enable signal to ID_RAM
id_wdata	O	5	Write data signal to ID_RAM
lbp_enable	O	1	CLBP circuit enabling signal
lbp_finish	I	1	Indication of the CLBP circuit is finished
ram_clbp	O	1	Indication that the CLBP circuit has the access to RAM_LBP

Signal	I/O	Bit-width	Description
gridX_i	I	4	Image sliced portion in X direction, value is 8, from testbench
gridY_i	I	4	Image sliced portion in Y direction, value is 8, from testbench
hcu_enable	O	1	HCU circuit enabling signal
gridX_o	O	4	Image sliced portion in X direction, value is 8, to HCU
gridY_o	O	4	Image sliced portion in Y direction, value is 8, to HCU
hcu_finish	I	1	Indication of the HCU circuit is finished
comparator_finish	I	1	Indication of the Comparator circuit is finished
comparator_enable	O	1	Comparator circuit enabling signal
ram_comp	O	1	Indication that the Comparator circuit & DCU circuit has the access to ID_RAM, HIST_RAM_TRAIN, HIST_RAM_PREDICT

- Understanding the function:

  Once system is initialized, it

  a) Receives *gridX* and *gridY* signal.
  b) Receive *valid* and *id* signal, then compute local binary pattern value and store the result into RAM_LBP.
  c) In training mode, computes histogram information from RAM_LBP and store it to HIST_RAM_TRAIN.
  d) Repeat step(b)~(c) until encounter prediction mode.
  e) Prediction mode is detected, goes to step(b).
  f) In prediction mode, computes histogram information from RAM_LBP and store it to HIST_RAM_PREDICT.
  g) Comparator starts to work, control DCU to compute D, where D is defined as:

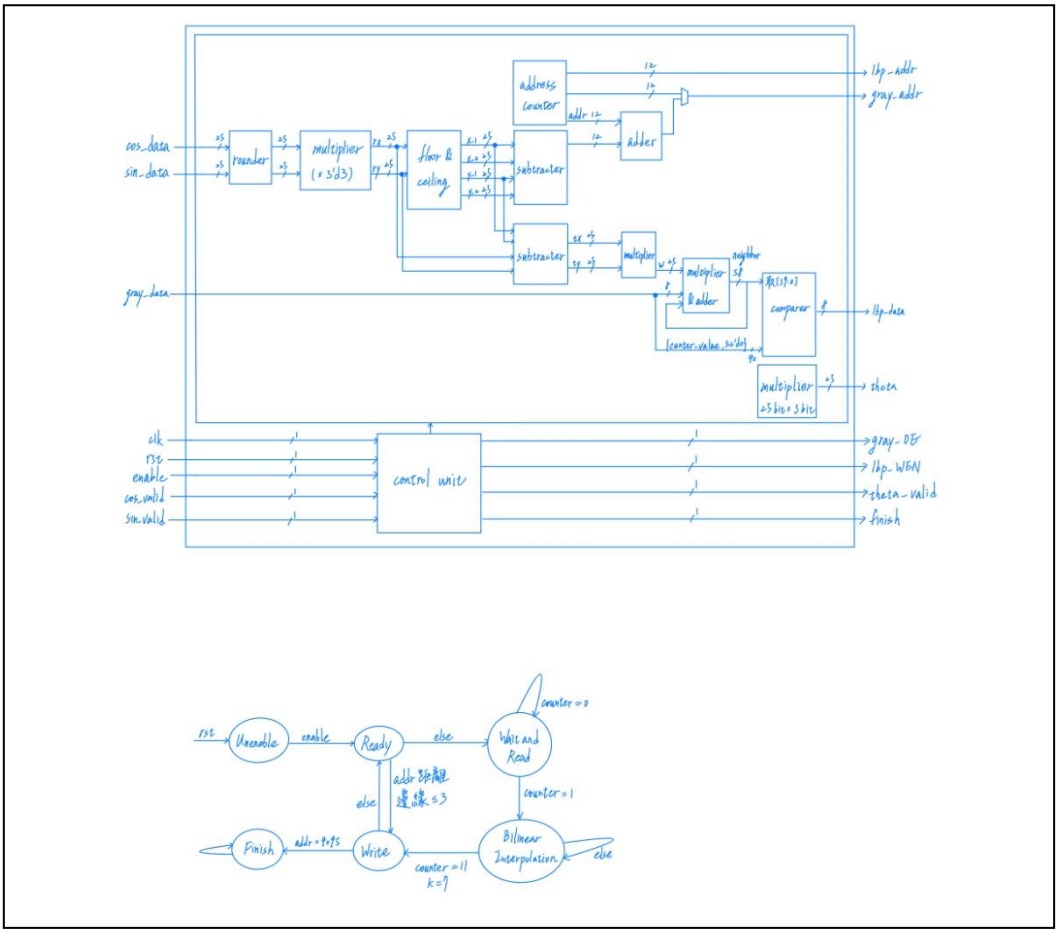
  $$D = \sum_{p=1}^{n} (hist\_predict_p - hist\_train_p)^2$$

  , where n is 16384(8x8x256).
  h) Loop step(g) for 7xN times, where N is the different subject count, and find the closest histogram in HIST_RAM_PREDICT w.r.t the prediction histogram computed in step(f), see p.18 in handout.
  i) Output *label* & *minDistance* & *done* signal.
  j) Repeat step(e)~(i) until testbench stops the simulation.

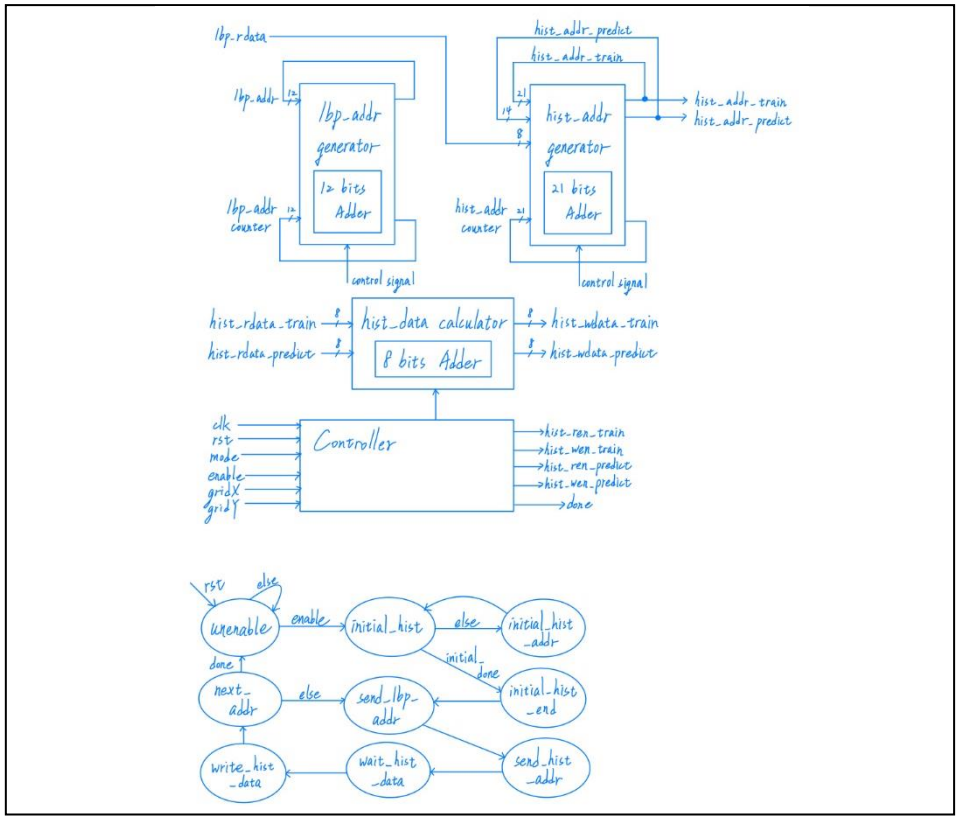
- Describe your design in detail. You can draw internal architecture or block diagram to describe your dsign. If your submodule contains any FSM, you should also depict it and elaborate as well.

  Note: **if you design your own internal architecture** other than using the provided one, please feel free to **alter the block** below, and add your own design as well as decribe them in detail.

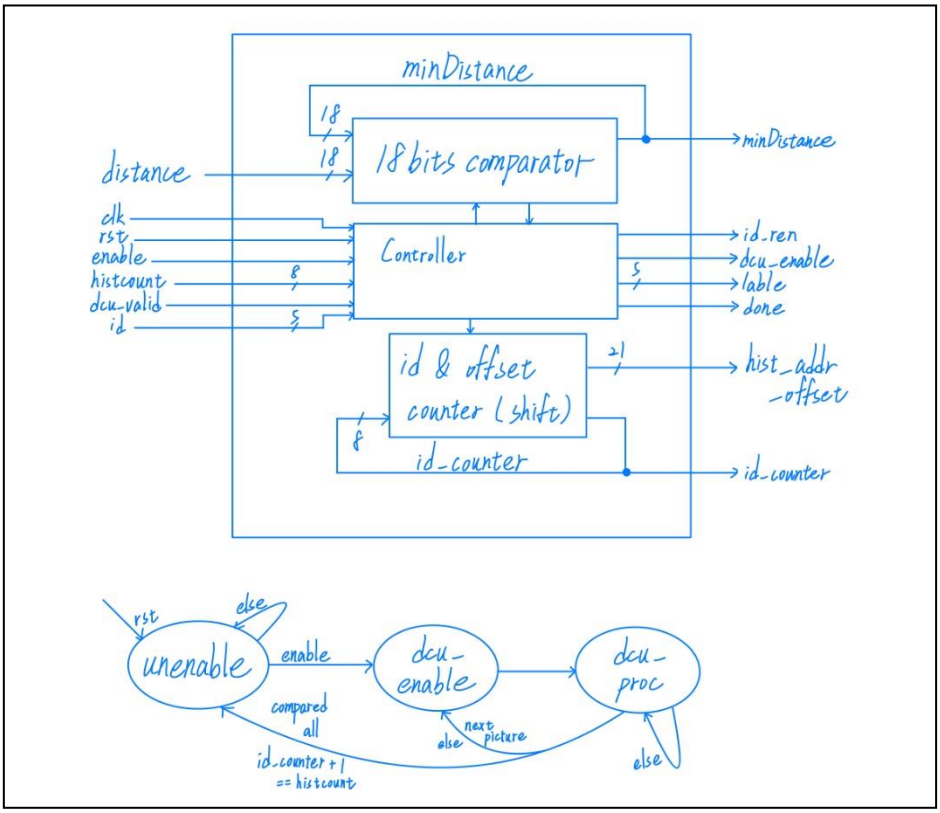
  - CLBP



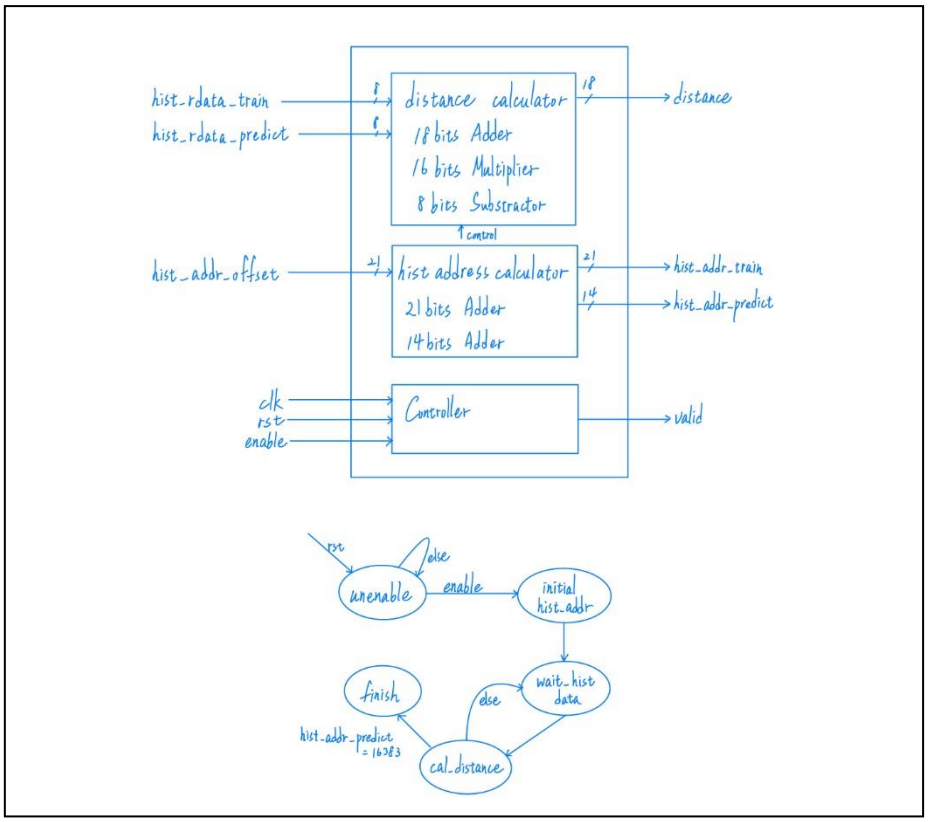
- HCU



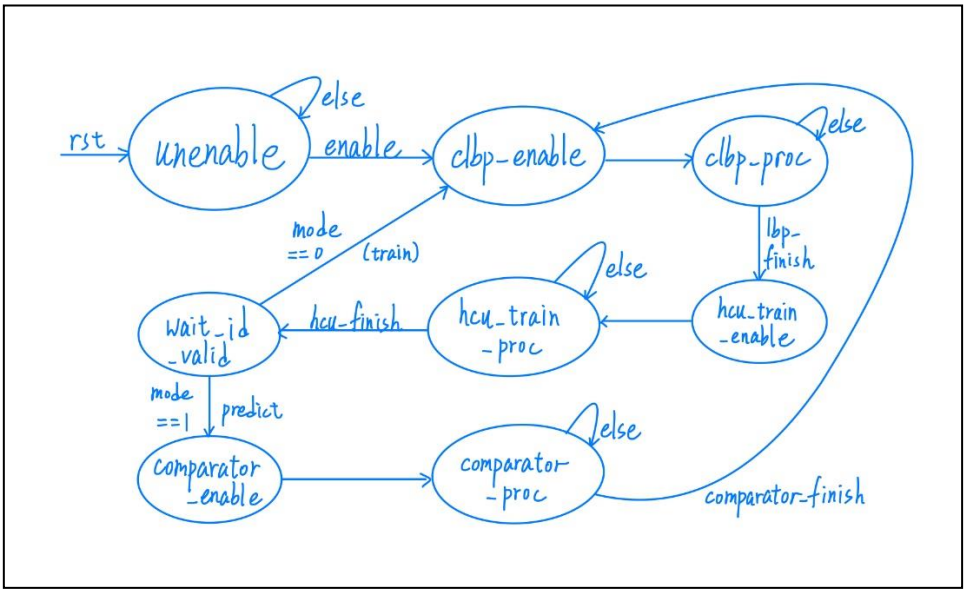
- Comparator



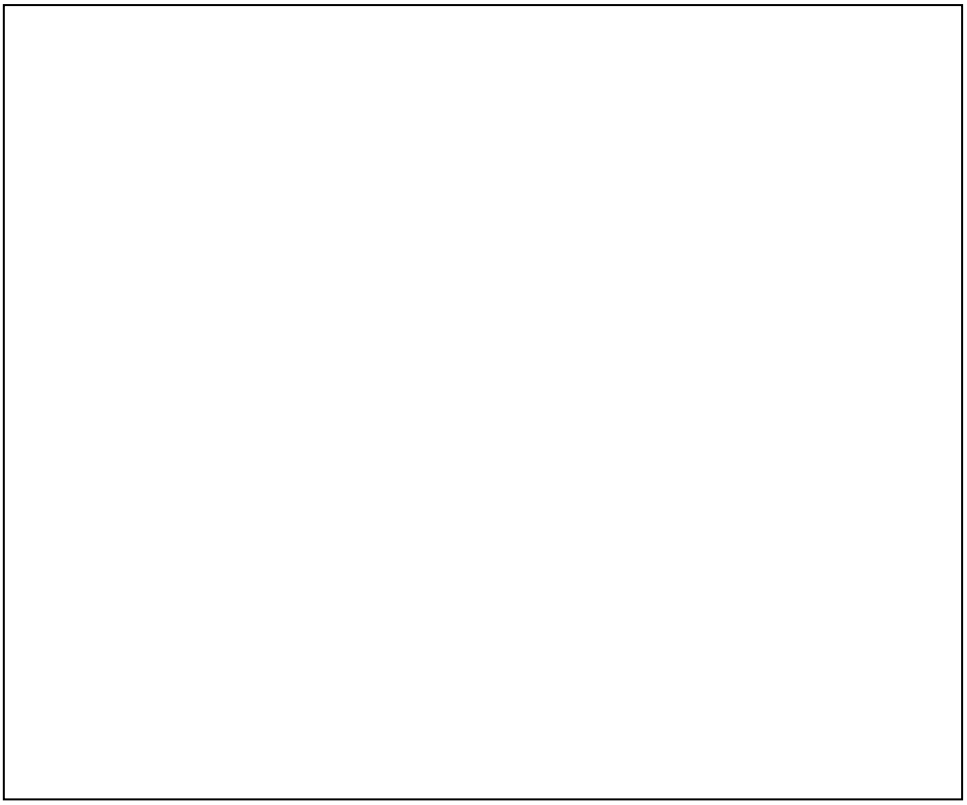
- DCU



- Controller
  - Draw your state diagram in controller and explain it



- Your own internal architecture
  - Draw and explain if you design your own architecture, if don't, you can skip this section.



1) Complete the Controller, HCU, CLBP, DCU, Comparator, and TOP module, in the system. **If you design your own architecture, please add the submodule list here!**

Submodule list:

1. …

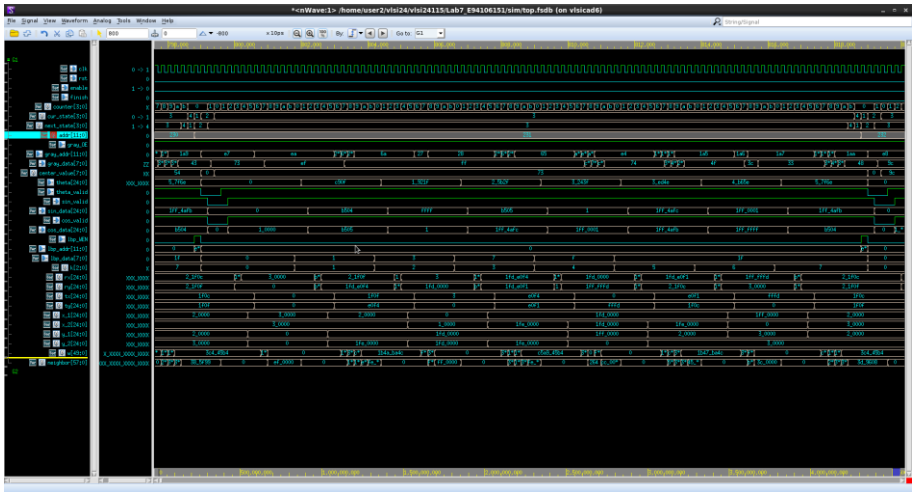
2) Compile the verilog code to verify the operations of this module works properly.

3) Synthesize your *top.v* with following the constraints:

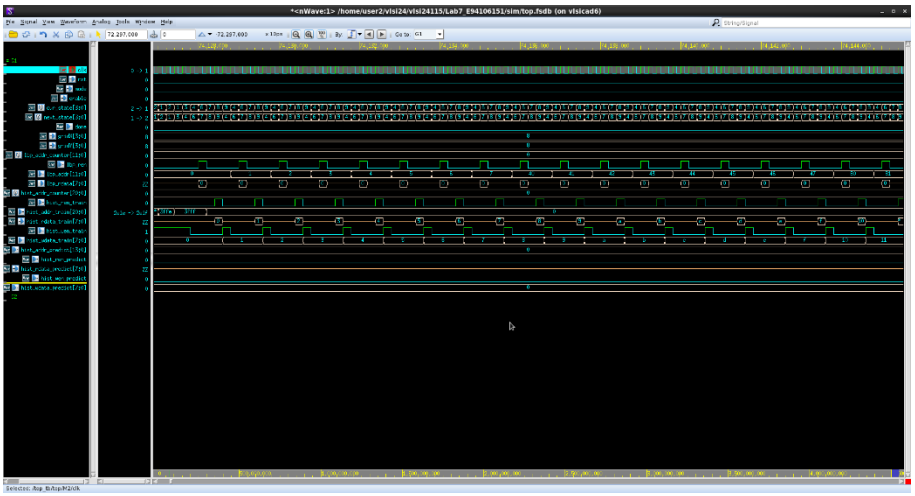
- Clock period: no more than 2.0 ns.
- Don't touch network: clk.
- Wire load model: Wire load model: N16ADFP_StdCellss0p72vm40c.
- Synthesized verilog file: *top_syn.v*.
- Timing constraint file: *top_syn.sdf*.

4) Please **attach your waveforms** and **specify your operations** on the waveforms. The more you elaborate, the higher the score is.

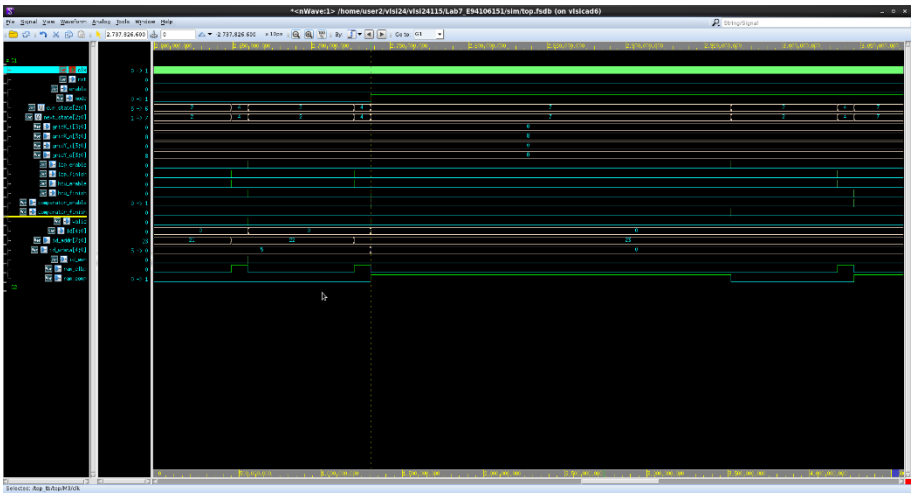
CLBP



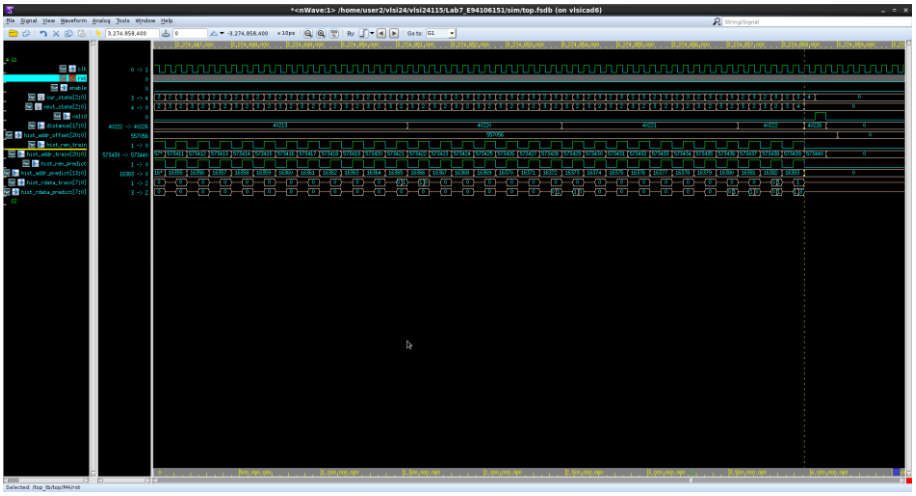
HCU



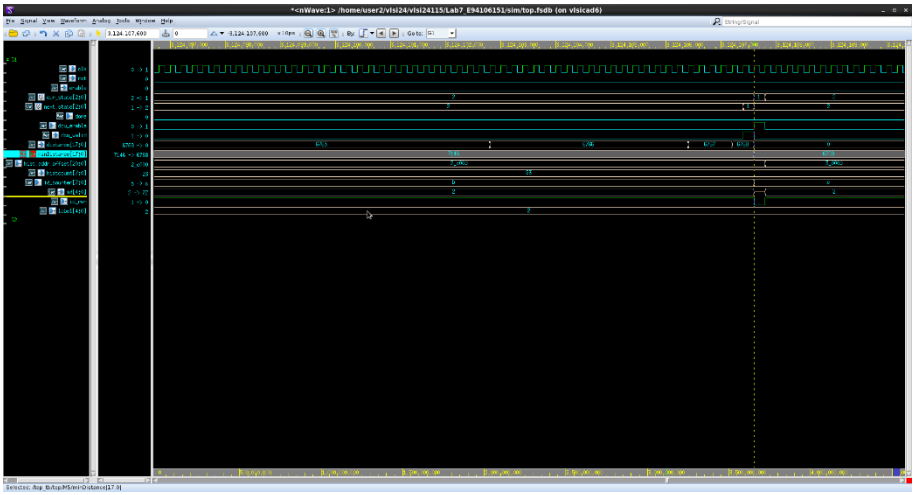
Controller



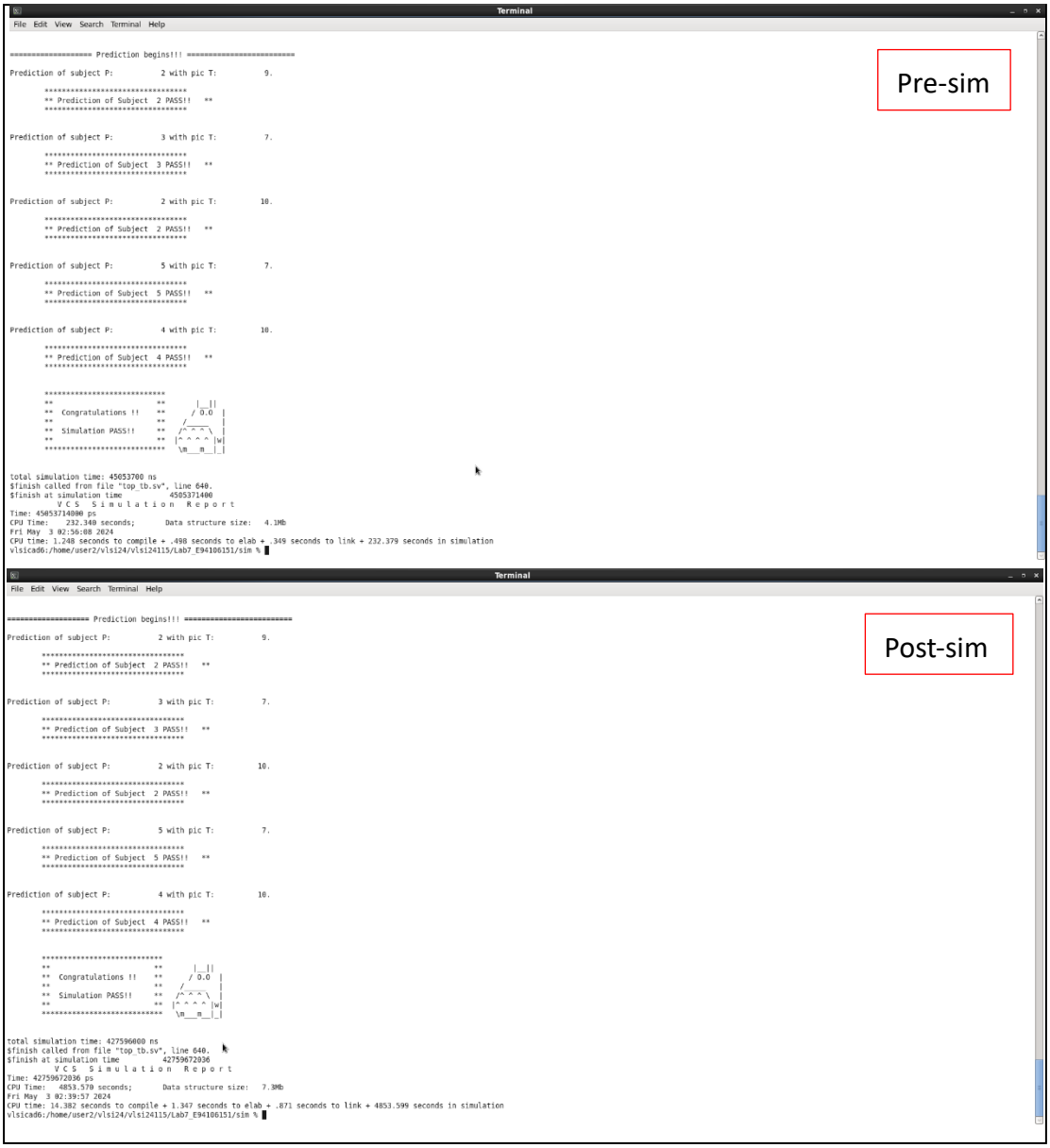
DCU



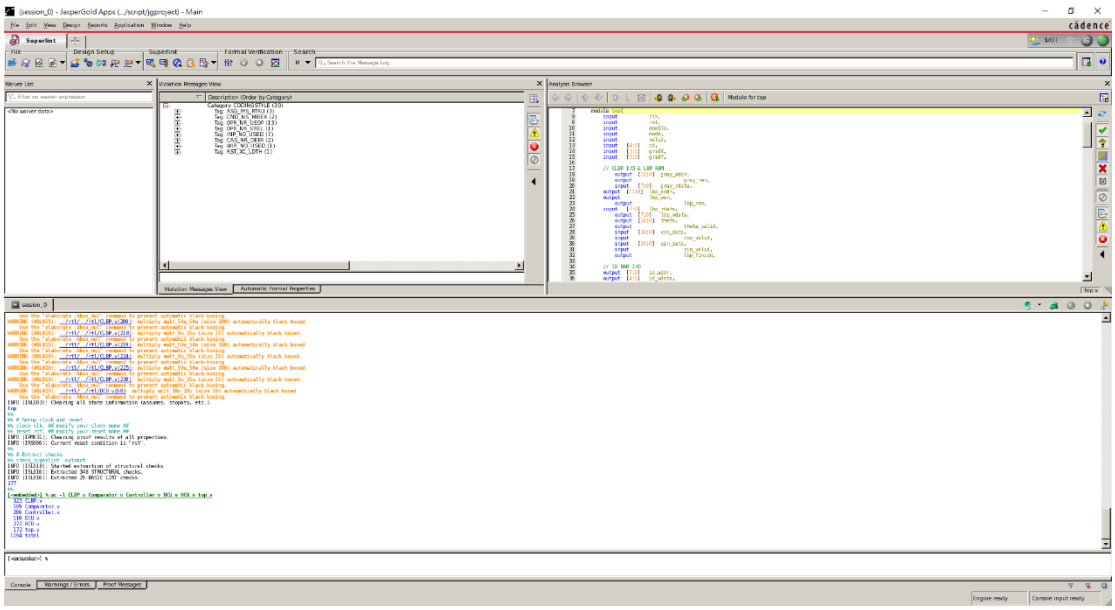
Comparator



5) Show simulation result



6) Show SuperLint coverage (top.v)



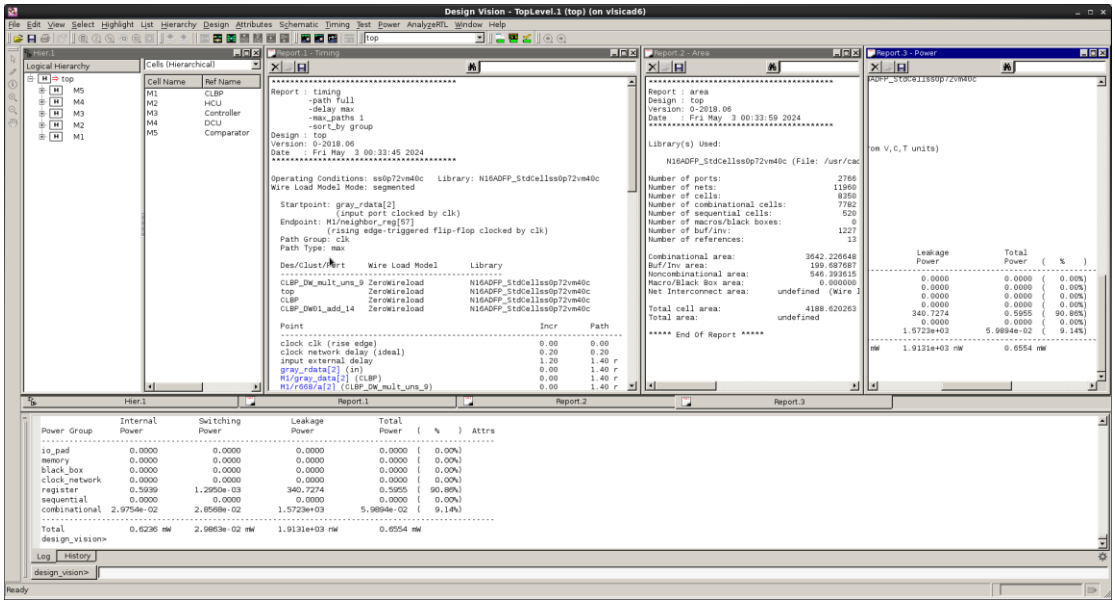
Coverage = (1-(30/1194))*100% = 97.487%

7) Your clock period, total cell area, post simulation time (top.v) in screenshot.

Clock period: 2.0 ns

Total cell area: 4188.620263 um^2

Post simulation time: 427596000 ns



8) Please describe how you optimize your design when you run into problems in synthesis .ex: plug in some registers between two instances to shorten your datapath, resource sharing for some registers to reduce your cell area.

Train and predict share the same reg, hist_addr_counter, to reduce the area in HCU.

- Lessons learned from this lab

  學到一個簡單的系統的架構和流程，還有如何用 verilog 將它描述出來。

- Suggestions for us (we appreciate your feedback)

希望 Lab 的講義能讓我們更容易懂每個 port 要做的事，如果是考慮到課堂上講解 Lab 的時間不足，還是希望即使影片長度過長，但至少在 Moodle 上講解 Lab 的影片能有更詳細的解說。

Please compress all the following files into one compressed file (".tar " format) and submit through Moodle website:

※ **NOTE**:

1. **If there are other files used in your design, please attach the files too and make sure they're properly included**.
2. Simulation commands

Lab7	Commands
superlint	% cd script % jg –superlint superlint.tcl
synthesis	% cd script % dv –f synthesis.tcl
Pre-sim	% cd sim % vcs -R -sverilog top_tb.sv -debug_access+all -full64
Post-sim	% cd sim % vcs -R -sverilog top_tb.sv -debug_access+all -full64 +define+SDF+SYN
Dump waveform	+define+FSDB

Don't use +define+FSDB when running post-sim, it'll occupy substantial amount of memory!