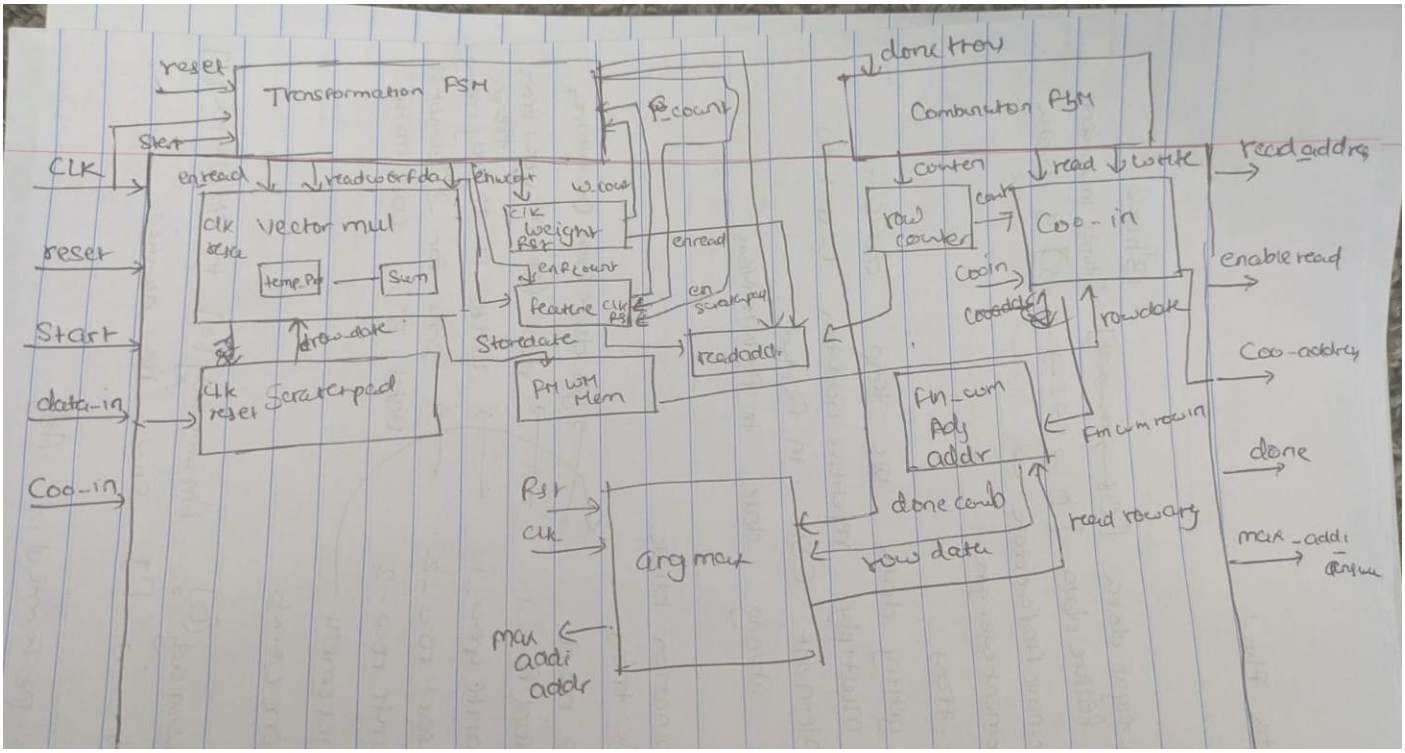
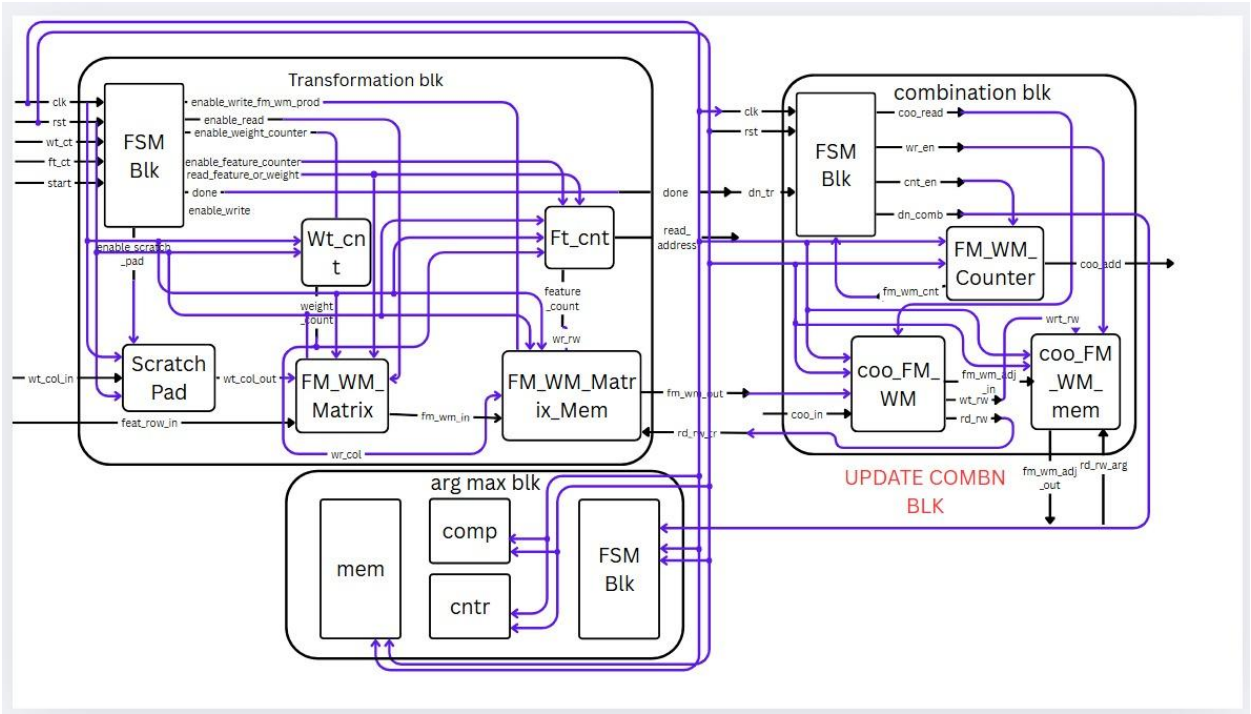
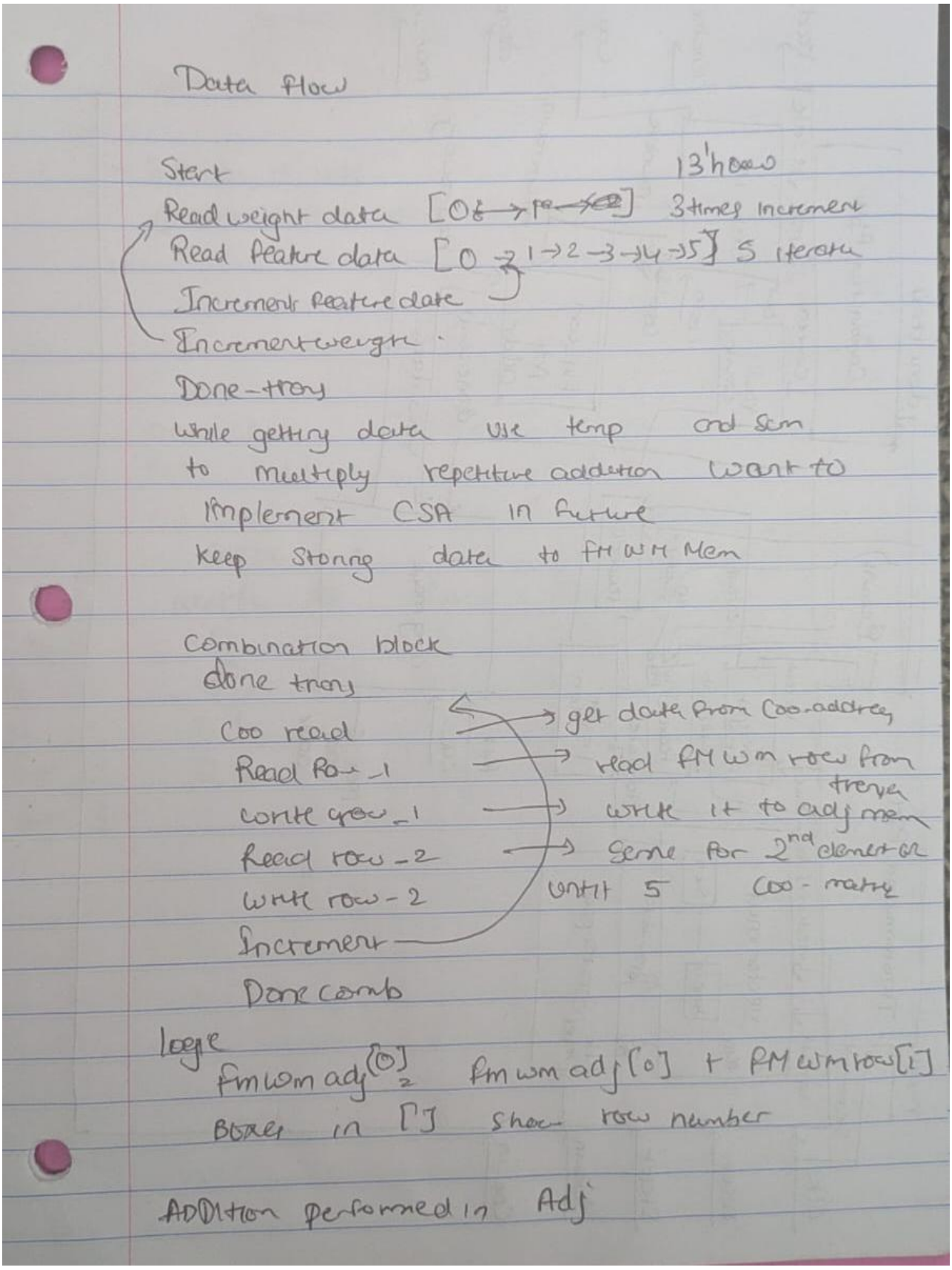




Bhaumik Thakker bthakker@asu.edu 1233701211



Data flow

Start — 13'h0000

Read weight data [0 → 1 → 2] 3 times increment

Read feature data [0 → 1 → 2 → 3 → 4 → 5] 5 iterations

Increment feature data

Increment weight.

Done-proc

While getting data use temp and sum to multiply repetitive addition want to implement CSA in future

Keep storing data to FM WM Mem

Combination block

done proc

Coo read → get data from Coo address

Read Ro-1 → read FM WM row from proc

write row-1 → write it to adj mem

Read row-2 → same for 2nd element of Coo-matrix

write row-2 until 5

Increment

Done comb

Logic

fm wm adj[0] fm wm adj[0] + FM WM row[i]

Boxes in [] show row number

Addition performed in Adj

1. Transformation Block

The Transformation_Block performs matrix-vector multiplication between features and weights. It includes submodules: Vector_Multiplier, Weight_Counter, Feature_Counter, Scratch_Pad, Matrix_FM_WM_Memory, and Transformation_FSM.

- Feature and weight vectors are stored in local scratchpads.
- Counters select feature rows and weight columns.
- The Vector_Multiplier performs parallel dot-products (WEIGHT_COLS = 3 parallel multipliers).
- Results are stored in Matrix_FM_WM_Memory.
- Transformation_FSM controls read, multiply, and write sequences.

2. Combination Block

The Combination_Block aggregates transformed features based on sparse graph connections (COO format).
It includes coo_in, ROW_Counter, Matrix_FM_WM_ADJ_Memory, and Combination_FSM.

- coo_in provides source-target node pairs.
- ROW_Counter iterates over edges.
- Features are accumulated selectively according to graph edges.
- Sparse aggregation is handled efficiently to reduce unnecessary operations.
- Combination_FSM manages control flow and signals completion.

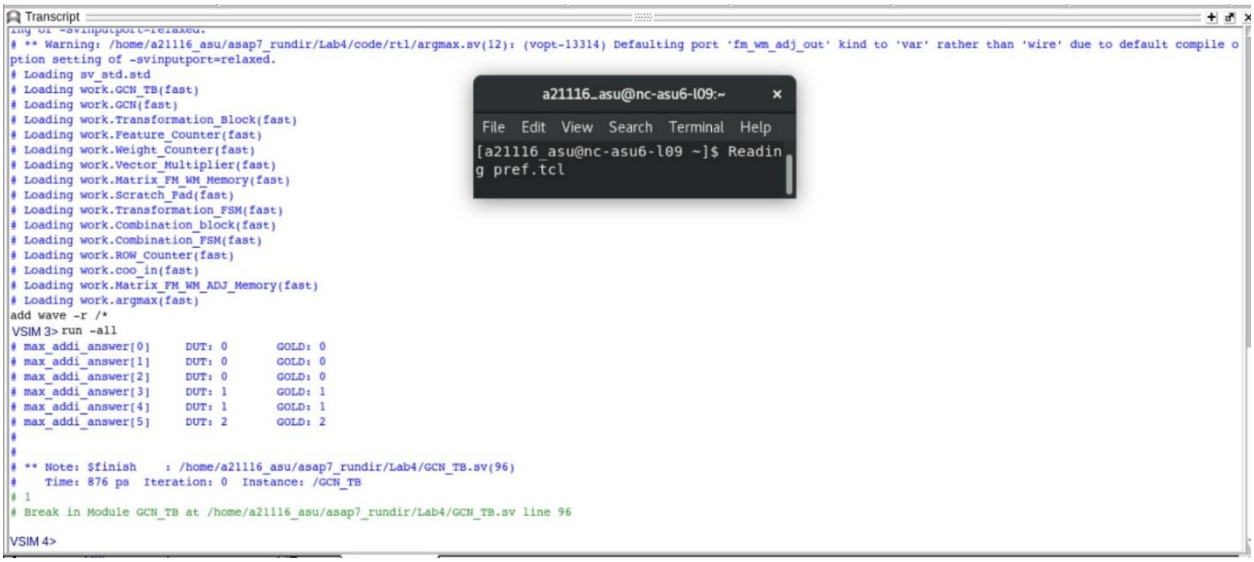
3. Argmax Block

The argmax module selects the maximum feature per node after aggregation.

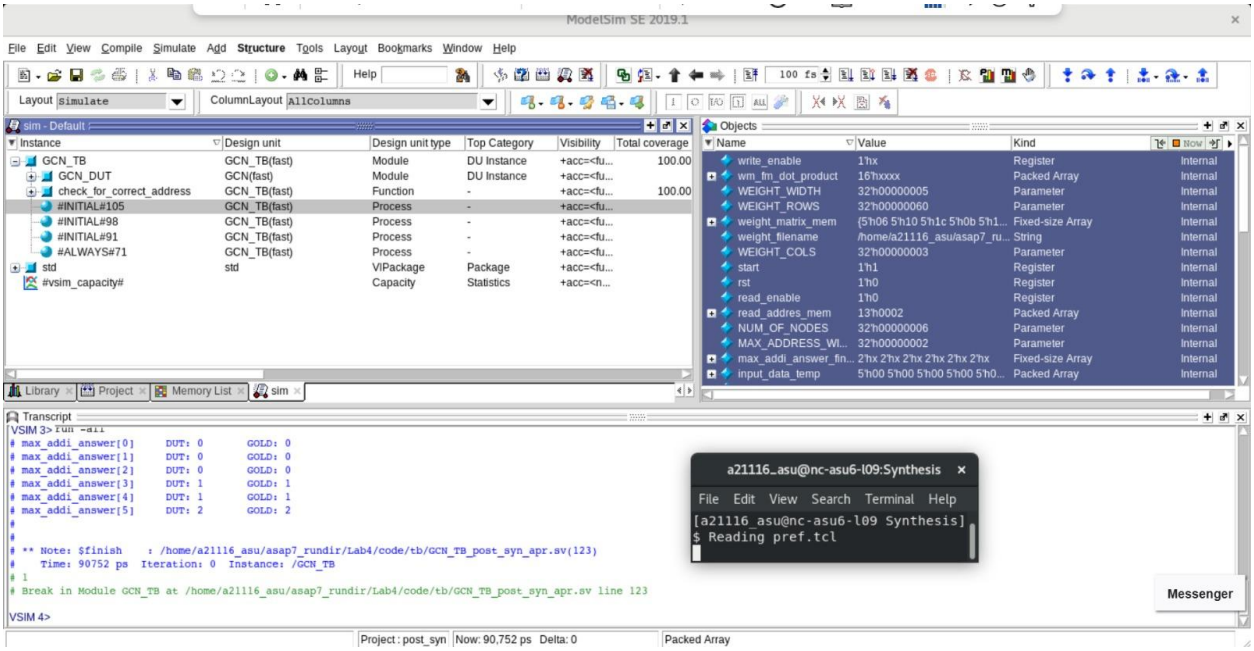
- Accepts aggregated feature vectors.
- Uses compare-and-select logic to find the maximum value per node.
- Stores maximum indices in max_addi_answer.
- Internal FSM controls idle, compare, and output states.
- Triggered after done_comb from Combination Block.

Bhaumik Thakker bthakker@asu.edu 1233701211

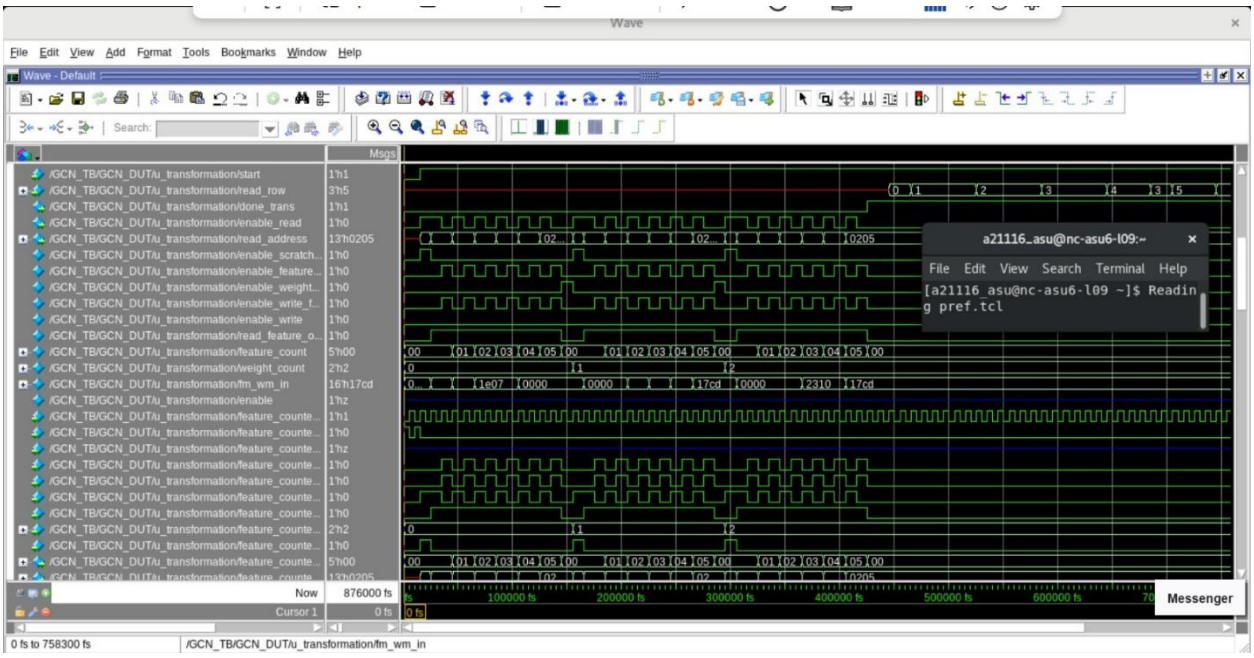
Behavioral Verilog Netlist output

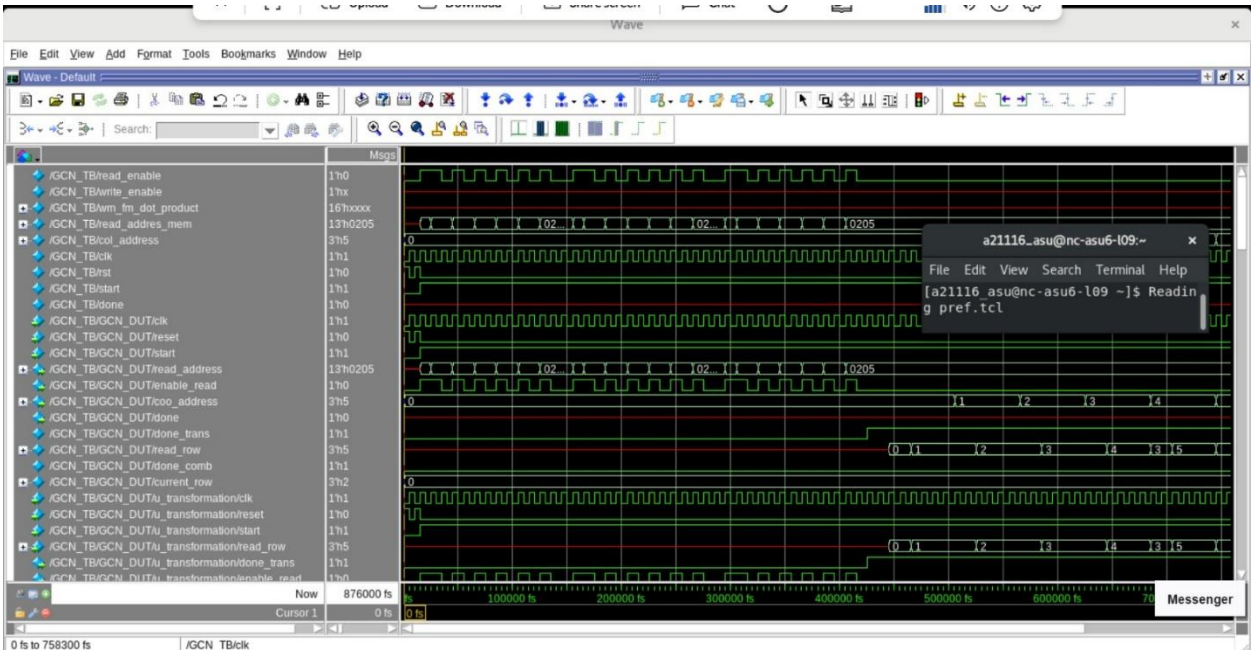


Synthesized Verilog Netlist output

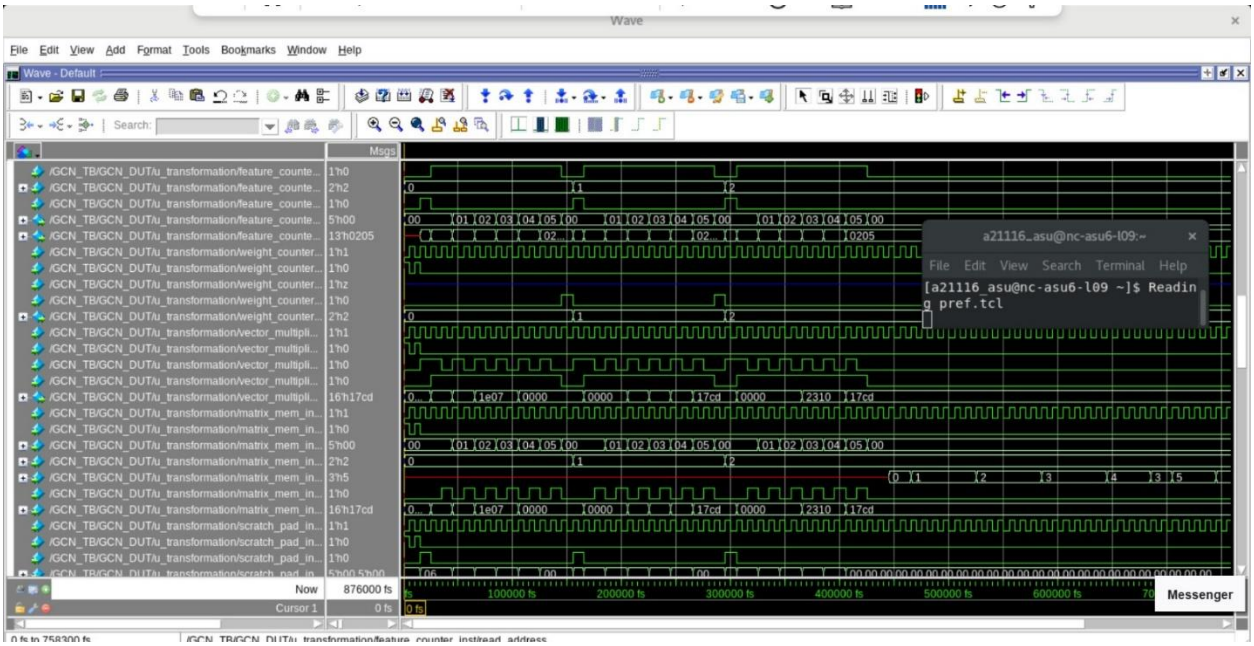


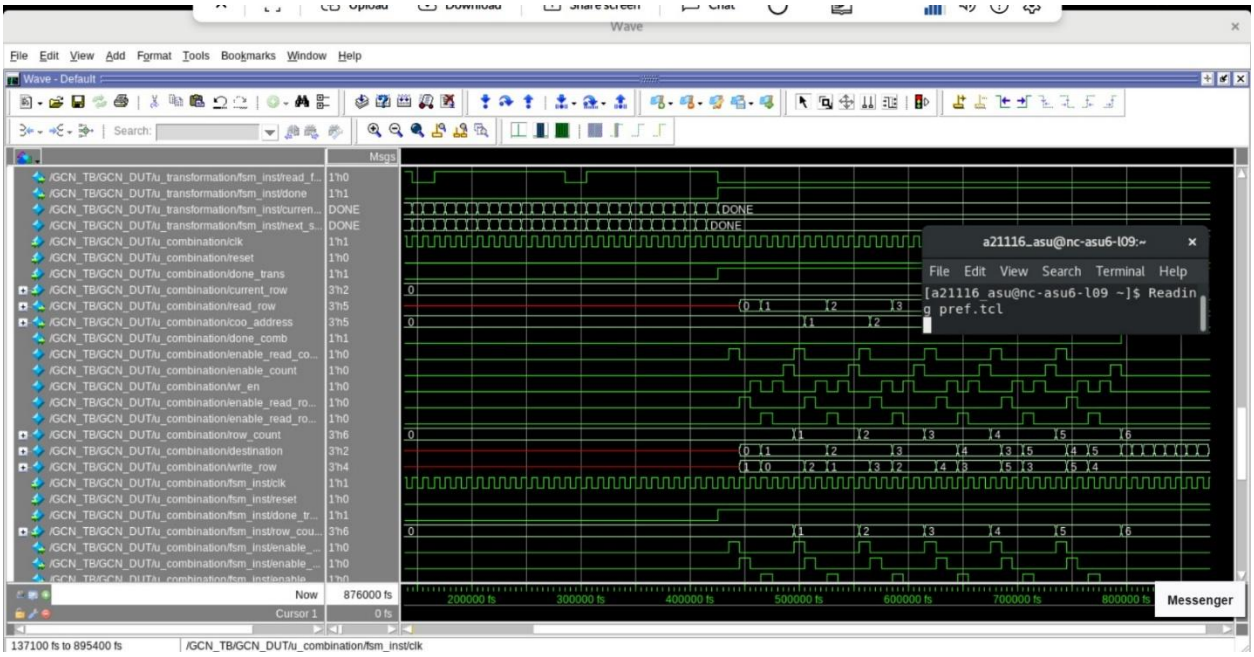
Bhaumik Thakker bthakker@asu.edu 1233701211



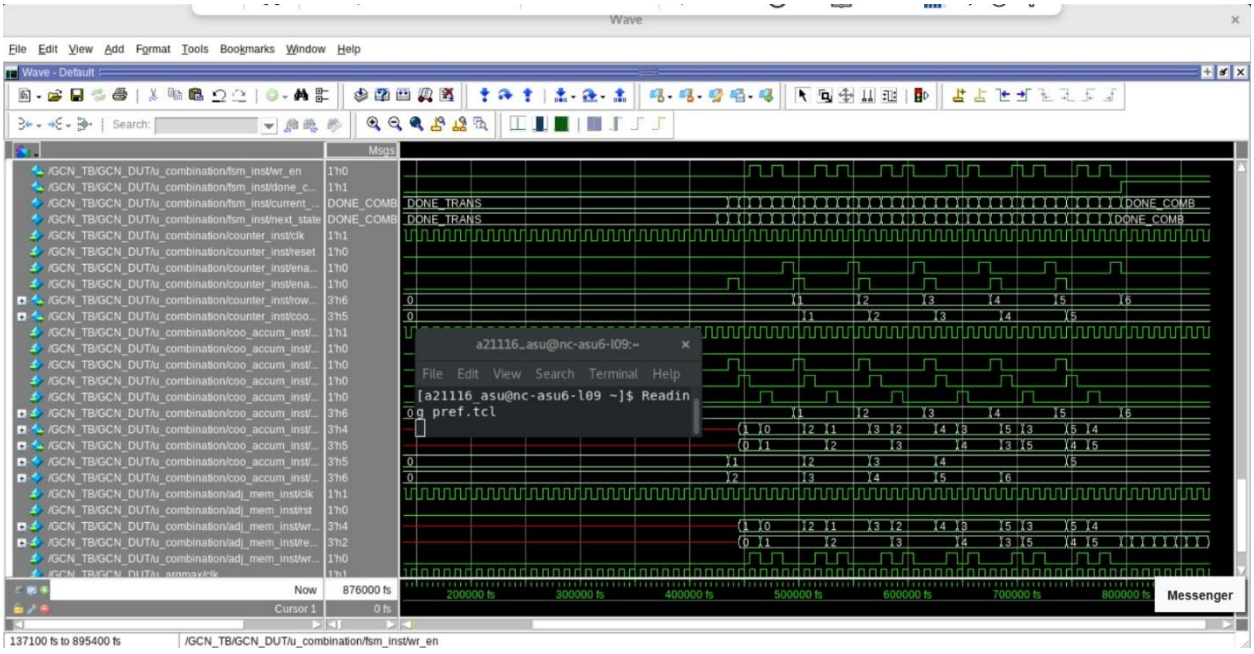


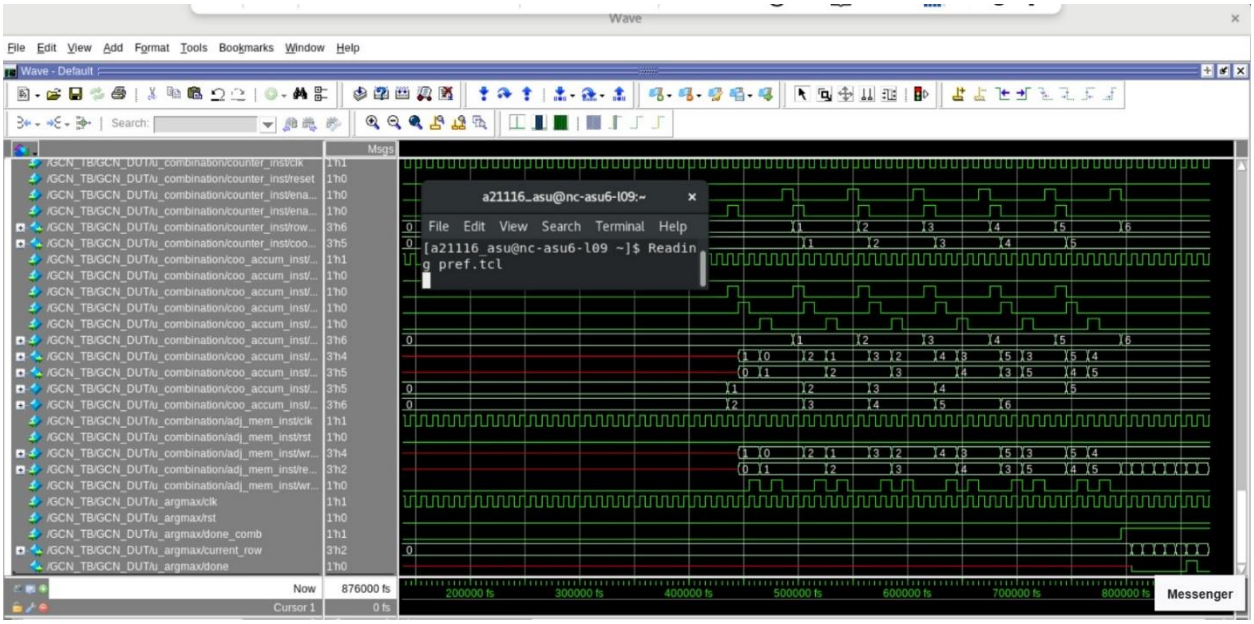
Bhaumik Thakker bthakker@asu.edu 1233701211





Bhaumik Thakker bthakker@asu.edu 1233701211





Bhaumik Thakker bthakker@asu.edu 1233701211