461 Final Project Report MLP

A well-balanced pipeline design for MLP

Team Members:

Ben Guan, Chenfeng Zhao, Shu Yang, Enrique Savillo

Section 1- System Architecture:

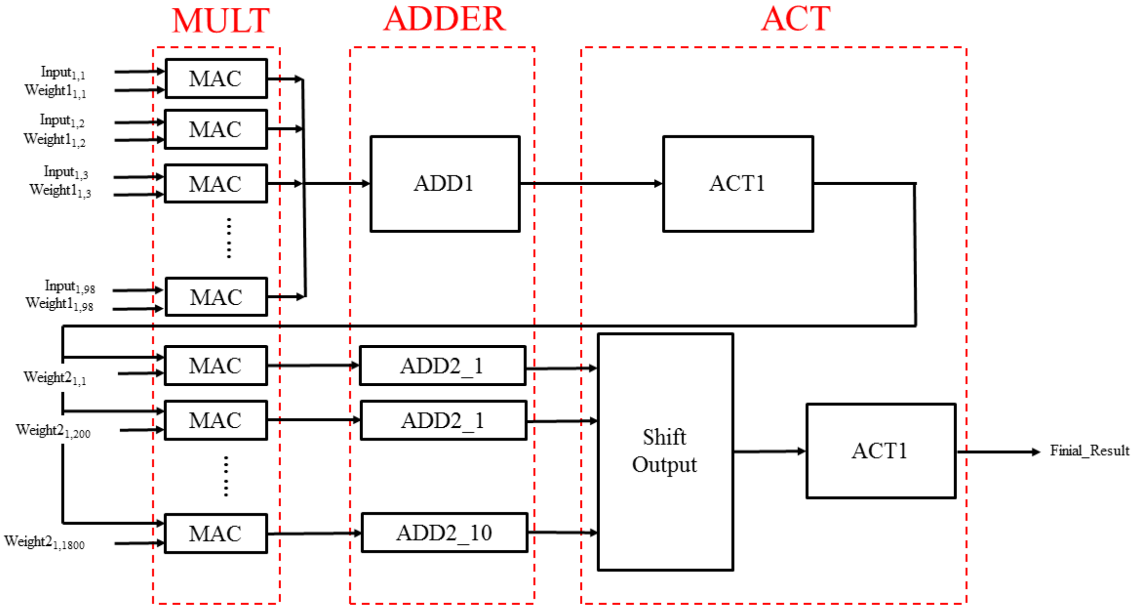


Figure 1 Overview Architecture of Our MLP Design

Figure 1 shows the overview architecture of our MLP design. The principle is that in the MULT module, there are 108 macs in total including 98 macs to compute the hidden result and 10 macs to compute the final result. Weight_matrix_1 and image inputs are passed to the first 98 macs as inputs and then fed into the adder to calculate the total sum of the 98 macs. After 8 clock cycles, we get one element of the hidden layer. The result is then passed to the ACT1, implemented by a look-up table, to find out the corresponding activation outputs. Then the activation result is passed to the 10 macs of the MULT module in parallel to compute the final result. The ten macs take the same hidden data and ten weight2 data in the same row as input.

After 200 clock cycles, we have 98 macs implemented in the first layer, it takes 8 clock cycles to compute the dot product of the first row of image data and first column of weight data. It also takes 200 clock cycles to compute dot product of the second layer. Since we have 10 images in total, the total number of clock cycles of a well-balanced pipeline will take 8*200*10 = 16000 cycles to finish. The simulation result also proves that our design takes exactly 16000 cycles to finish. Since the theory result matches simulation result very well, we can conclude that our design is a perfect-balanced pipeline as the result of the first layer is directly fed into the second layer without any additional pipeline stalls.

Section 2- Design Details:

2.1 First Layer

For the first layer, we implemented 98 Macs, each one takes in an input and a weight. The results of the 98 Macs will be passed into the adder. The adder will sum up the 98 products. After 8 clock cycles, 1 element of hidden results is computed. The result is then passed to the activation function module realized by look-up table in which the given input is looked up and the correct data is outputted. To make sure we have a well-balanced pipeline throughout the design, we designed not to store the intermediate data back to the sram. Instead, The result of the activation function will be broadcast to each mac in the second layer.

2.2 Second Layer

For the second layer, we have 10 Macs, each one takes in an input and a weight. Each mac will compute dot product of of one of the rows in the hidden layer and one of the columns in the weight 2. After 200 clock cycles, the result of the second layer is computed. Since the first layer takes 8 clock cycles and the second layer takes 200 clock cycles. Each image will be done processing after 1,600 clock cycles.

In the top.v module, we instantiated the CNT which is used to calculate rows and columns address of the memory. We designed the CTRL module to control the data flow. We also designed MULT module containing the 98 macs for layer 1 and 10 macs from layer 2. Then ADDER module is designed to sum up the products of 98 macs and perform an accumulating sum (partial sum) of the 10 macs in layer 2. Finally, LUT module is instantiated to realize sigmoid function by look-up table.

2.4 DC Optimization

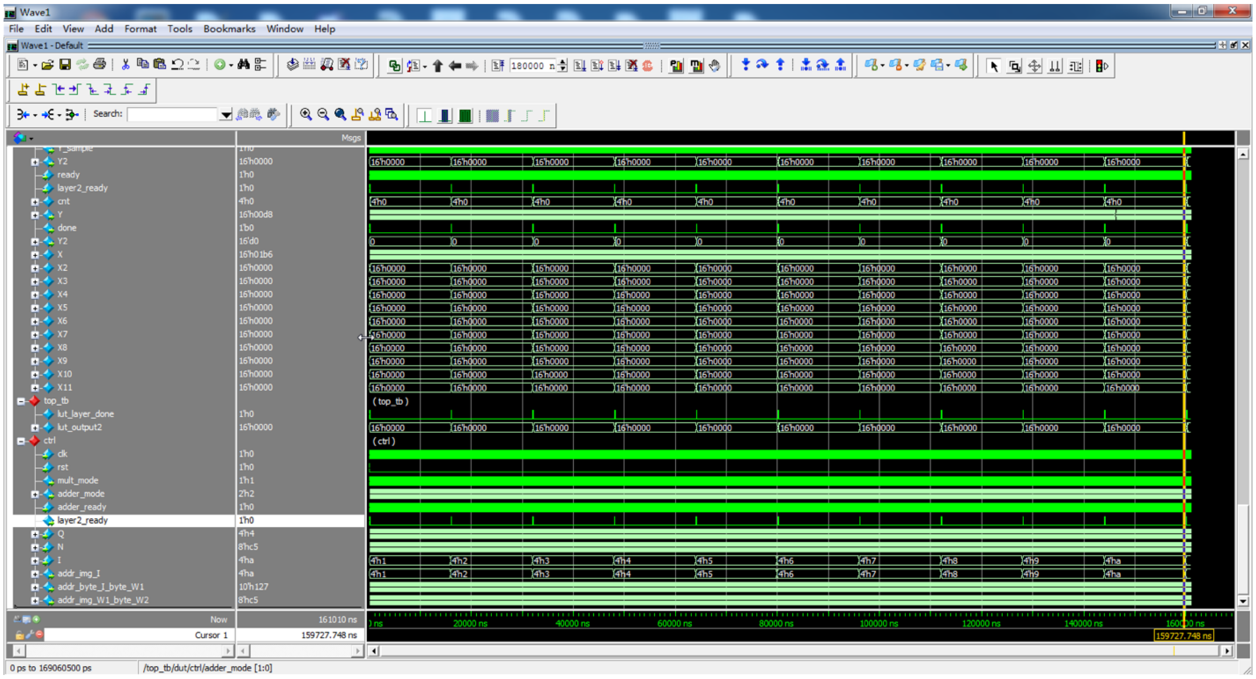
After we have confirmed our output to be consistent with the MatLab output, we performed area and power optimization in DC.

In .tcl file, we used *compile_ultra -incremental* and *set_max_area 0* to achieve optimization on area by incremental way. Besides, compared with designing our own multiplier in each mac, using “*” to perform multiplication operation in each mac would reduce 80% of total area. The reason is that once we directly use *, we could directly use optimized multiplication cell in DC library, while for our own multiplier, DC would use several gates to realize it, consuming more areas.

As for Power, we tested our design with 10 ns and found that the total power is about 27.7 mW. In order to reduce the power consumption, we decided to change to clock period to 50 ns, this reduced the power consumption linearly.

Section 3- Results Analysis:

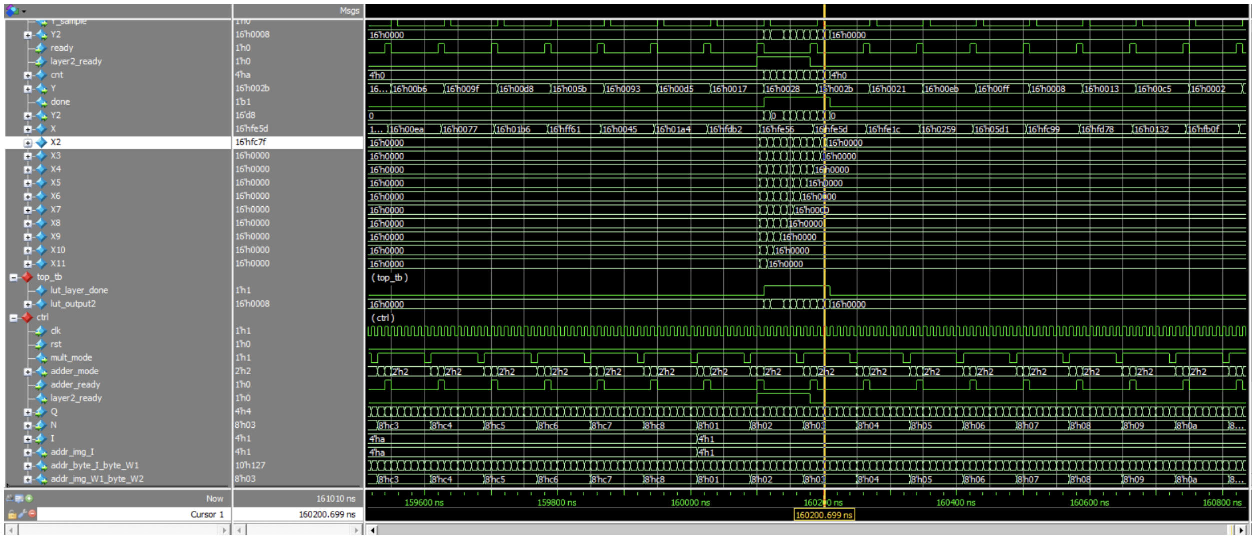
3.1 Simulation Waveform



	Msgs
Y_sample	1'h0
Y2	16'h0000
ready	1'h0
layer2_ready	1'h0
cnt	4'h0
Y	16'h00d8
done	1'b0
Y2	16'd0
X	16'h01b6
X2	16'h0000
X3	16'h0000
X4	16'h0000
X5	16'h0000
X6	16'h0000
X7	16'h0000
X8	16'h0000
X9	16'h0000
X10	16'h0000
X11	16'h0000
top_tb	
lut_layer_done	1'h0
lut_output2	16'h0000
ctrl	
clk	1'h0
rst	1'h0
mult_mode	1'h1
adder_mode	2'h2
adder_ready	1'h0
layer2_ready	1'h0
Q	4'h4
N	8'hc5
I	4'ha
addr_img_I	4'ha
addr_byte_I_byte_W1	10'h127
addr_img_W1_byte_W2	8'hc5
Now	161010 ns
Cursor 1	159727.748 ns

0 ps to 169060500 ps
/top_tb/dut/ctrl/adder_mode [1:0]

Figure 2 Overview waveform of all the ten input images



```
Y_sample                1'h0
Y2                      16'h0008
ready                   1'h0
layer2_ready            1'h0
cnt                     4'ha
Y                       16'h002b
done                    1'b1
Y2                      16'd8
X                       16'hfe5d
X2                      16'hfc7f
X3                      16'h0000
X4                      16'h0000
X5                      16'h0000
X6                      16'h0000
X7                      16'h0000
X8                      16'h0000
X9                      16'h0000
X10                     16'h0000
X11                     16'h0000
top_tb
  lut_layer_done        1'h1
  lut_output2           16'h0008
ctrl
  clk                   1'h1
  rst                   1'h0
  mult_mode             1'h1
  adder_mode            2'h2
  adder_ready           1'h0
  layer2_ready          1'h0
  Q                     4'h4
  N                     8'h03
  I                     4'h1
  addr_img_I            4'h1
  addr_byte_I_byte_W1   10'h127
  addr_img_W1_byte_W2   8'h03
Now                     161010 ns
Cursor 1                160200.699 ns
```

Figure 3 Waveform of the last final result

Figure2 shows the overview of waveform of all the ten images. From the figure, it can be observed that we have taken 16000 cycles to finish the whole simulation (10ns cycle period set in the testbench). Figure 3 shows the output process of the last element in finial result matrix. It can be seen that 10 finial results are calculated in parallel and outputted sequentially through a shifter before activation function module.

3.2 Simulation results

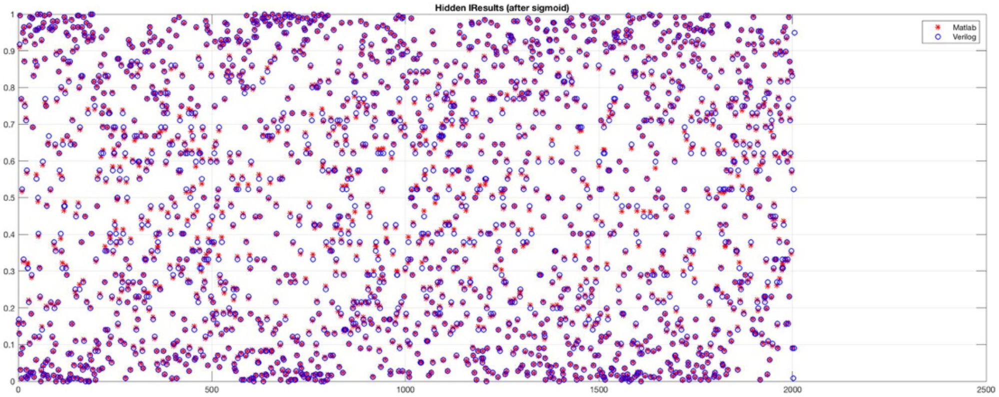


Figure 4 Hidden layer results of matlab and our MLP design (after sigmoid)

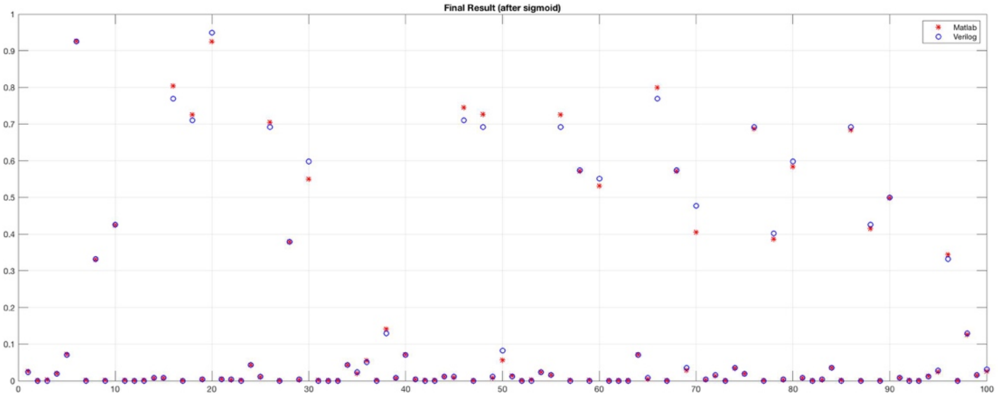


Figure 5 Final results of matlab and our MLP design (after sigmoid)

Figure 4 shows the hidden results calculated by layer and matlab results. It can be seen that most of the results of our MLP system fit well with the matlab results. Figure 5 shows the final results of matlab and our MLP design. It can be observed that 10 out of 100 results don't fit well, which means that the accuracy of our system is about 90%.

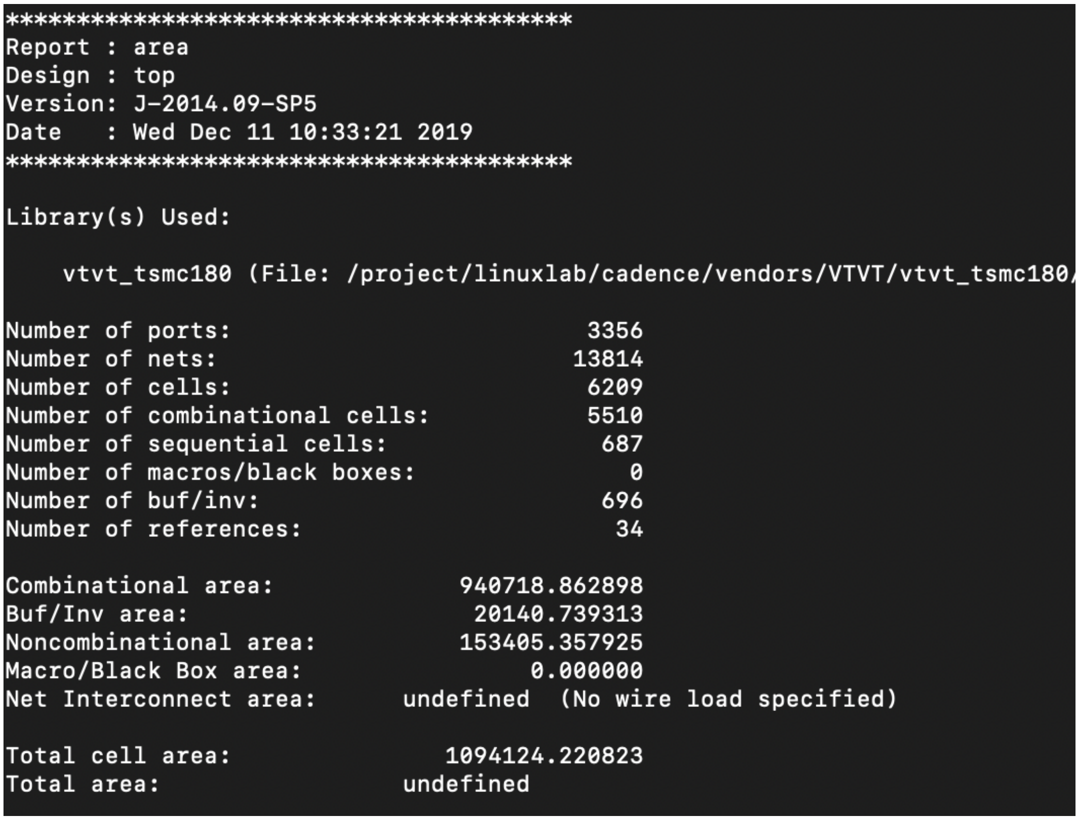
3.3 Performance Comparison:

Table 1 Performance evaluation with different clock frequencies

Clock Frequency	Area (mm^2)	Power(mW)	Number of Clock Cycles	Execution Time (us)
100 MHz (10ns)	1.069	27.768	16000	160
50 MHz (20ns)	1.094	14.79	16000	320
20 MHz (50ns)	1.094	5.894	16000	800

3.1 Area

With different clock frequencies, the total area is about 1.094 mm^2 with 0.94 mm^2 combinational area and 0.15 mm^2 non-combinational area. This is small compared to the requirement of 8 mm^2 .

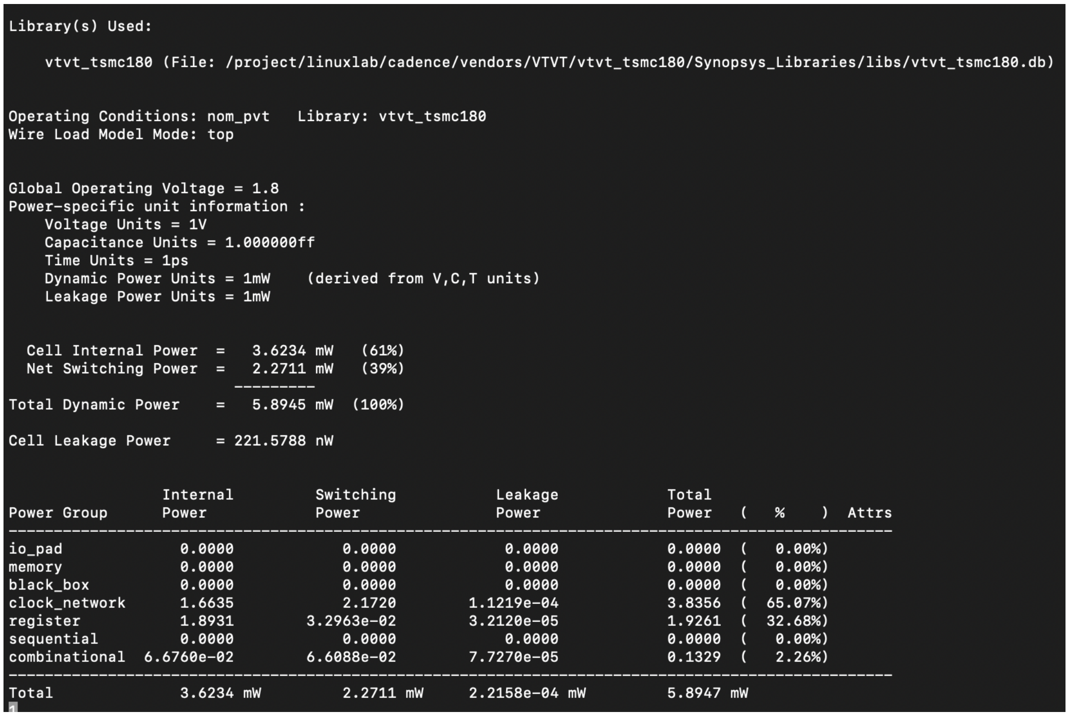


```
****************************************
Report : area
Design : top
Version: J-2014.09-SP5
Date   : Wed Dec 11 10:33:21 2019
****************************************

Library(s) Used:

    vtvt_tsmc180 (File: /project/linuxlab/cadence/vendors/VTVT/vtvt_tsmc180/

Number of ports:                         3356
Number of nets:                         13814
Number of cells:                         6209
Number of combinational cells:           5510
Number of sequential cells:               687
Number of macros/black boxes:               0
Number of buf/inv:                        696
Number of references:                      34

Combinational area:            940718.862898
Buf/Inv area:                   20140.739313
Noncombinational area:         153405.357925
Macro/Black Box area:               0.000000
Net Interconnect area:      undefined  (No wire load specified)

Total cell area:              1094124.220823
Total area:                 undefined
```

Figure 6 Area report of Design Compiler

3.2 Power

We tested the design with three different clock periods: 10ns, 20ns, and 50ns. We found out that the area for three cases are about the same, but the power consumption varies significantly. For the 50ns case, the total power of the design is 5.89mW with 3.62mW internal power and 2.27mW switching power. The total power consumption is low compared to the project requirement of 150 mW. For the 10ns case, the power consumption is 27.76mW. As shown in table 1, we can see that the power consumption increases linear with the clock frequency.

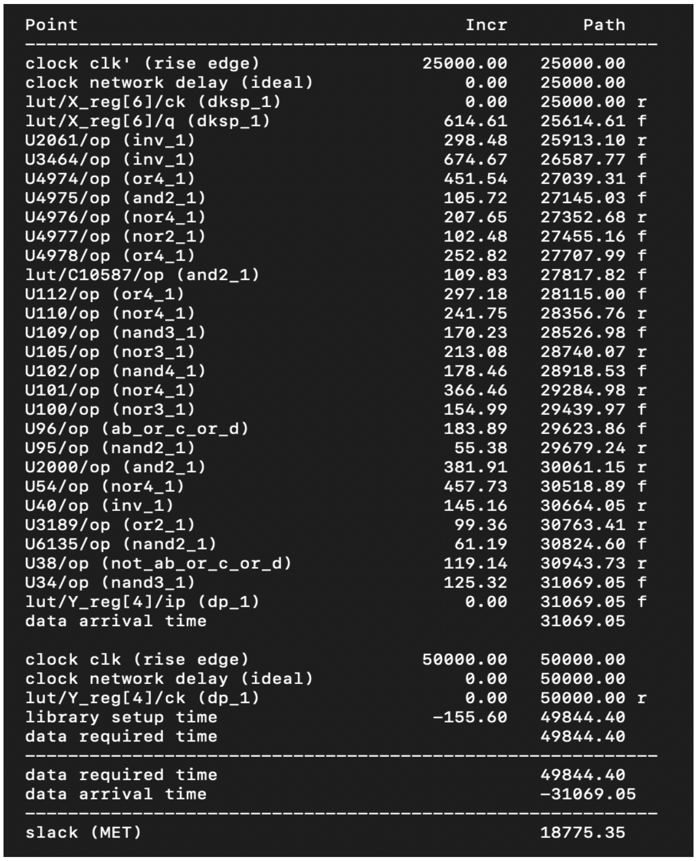


```
Library(s) Used:

    vtvt_tsmc180 (File: /project/linuxlab/cadence/vendors/VTVT/vtvt_tsmc180/Synopsys_Libraries/libs/vtvt_tsmc180.db)


Operating Conditions: nom_pvt   Library: vtvt_tsmc180
Wire Load Model Mode: top


Global Operating Voltage = 1.8
Power-specific unit information :
    Voltage Units = 1V
    Capacitance Units = 1.000000ff
    Time Units = 1ps
    Dynamic Power Units = 1mW    (derived from V,C,T units)
    Leakage Power Units = 1mW


  Cell Internal Power  =   3.6234 mW   (61%)
  Net Switching Power  =   2.2711 mW   (39%)
                         ---------
Total Dynamic Power    =   5.8945 mW  (100%)

Cell Leakage Power     = 221.5788 nW


                 Internal         Switching           Leakage            Total
Power Group      Power            Power               Power              Power   (   %    )  Attrs
--------------------------------------------------------------------------------------------------
io_pad             0.0000            0.0000            0.0000            0.0000  (   0.00%)
memory             0.0000            0.0000            0.0000            0.0000  (   0.00%)
black_box          0.0000            0.0000            0.0000            0.0000  (   0.00%)
clock_network      1.6635            2.1720        1.1219e-04            3.8356  (  65.07%)
register           1.8931        3.2963e-02        3.2120e-05            1.9261  (  32.68%)
sequential         0.0000            0.0000            0.0000            0.0000  (   0.00%)
combinational  6.6760e-02        6.6088e-02        7.7270e-05            0.1329  (   2.26%)
--------------------------------------------------------------------------------------------------
Total              3.6234 mW         2.2711 mW     2.2158e-04 mW         5.8947 mW
```

Figure 7 Power report of Design Compiler

3.3 Timing

This picture below shows we have a positive slack. This indicates that we have no time violation.



```
Point                                         Incr       Path
-----------------------------------------------------------------
clock clk' (rise edge)                    25000.00   25000.00
clock network delay (ideal)                   0.00   25000.00
lut/X_reg[6]/ck (dksp_1)                      0.00   25000.00 r
lut/X_reg[6]/q (dksp_1)                     614.61   25614.61 f
U2061/op (inv_1)                            298.48   25913.10 r
U3464/op (inv_1)                            674.67   26587.77 f
U4974/op (or4_1)                            451.54   27039.31 f
U4975/op (and2_1)                           105.72   27145.03 f
U4976/op (nor4_1)                           207.65   27352.68 r
U4977/op (nor2_1)                           102.48   27455.16 f
U4978/op (or4_1)                            252.82   27707.99 f
lut/C10587/op (and2_1)                      109.83   27817.82 f
U112/op (or4_1)                             297.18   28115.00 f
U110/op (nor4_1)                            241.75   28356.76 r
U109/op (nand3_1)                           170.23   28526.98 f
U105/op (nor3_1)                            213.08   28740.07 r
U102/op (nand4_1)                           178.46   28918.53 f
U101/op (nor4_1)                            366.46   29284.98 r
U100/op (nor3_1)                            154.99   29439.97 f
U96/op (ab_or_c_or_d)                       183.89   29623.86 f
U95/op (nand2_1)                             55.38   29679.24 r
U2000/op (and2_1)                           381.91   30061.15 r
U54/op (nor4_1)                             457.73   30518.89 f
U40/op (inv_1)                              145.16   30664.05 r
U3189/op (or2_1)                             99.36   30763.41 r
U6135/op (nand2_1)                           61.19   30824.60 f
U38/op (not_ab_or_c_or_d)                   119.14   30943.73 r
U34/op (nand3_1)                            125.32   31069.05 f
lut/Y_reg[4]/ip (dp_1)                        0.00   31069.05 f
data arrival time                                    31069.05

clock clk (rise edge)                     50000.00   50000.00
clock network delay (ideal)                   0.00   50000.00
lut/Y_reg[4]/ck (dp_1)                        0.00   50000.00 r
library setup time                         -155.60   49844.40
data required time                                   49844.40
-----------------------------------------------------------------
data required time                                   49844.40
data arrival time                                   -31069.05
-----------------------------------------------------------------
slack (MET)                                          18775.35
```

Figure 8 Timing report of Design Compiler

Section 5- Place and Route:

The following image shows the MLP accelerator in schematic generated using Encounter. As we can see, the blue line are the grid lines.

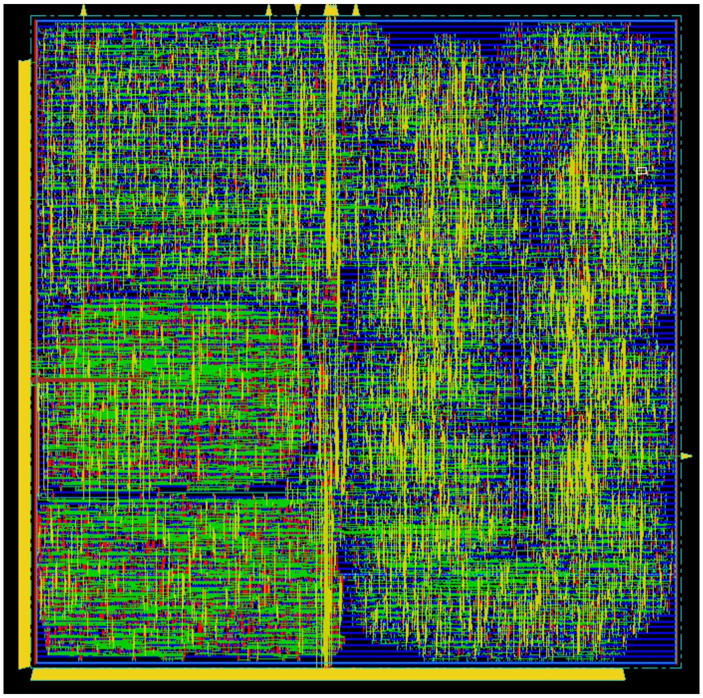
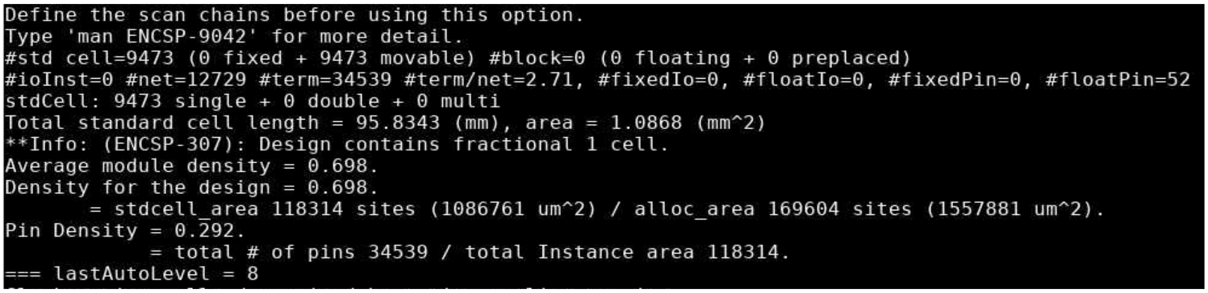


Figure 9 Physical layout of our MLP design

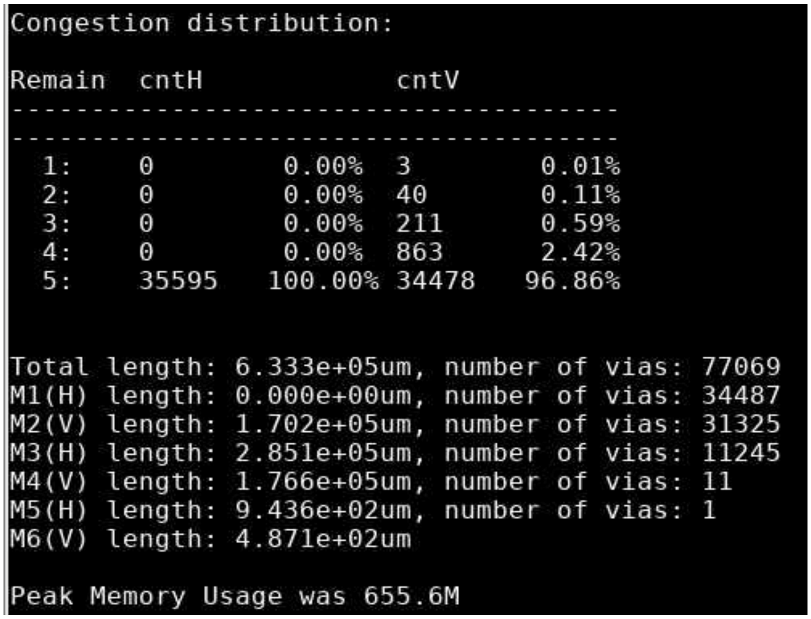
The area of the total standard cell is roughly 1.0868 mm^2. With the pins and circuit board, the total area is about 1.557 mm^2. The pin density is 0.292 and the density of the design is about 0.698.



```
Define the scan chains before using this option.
Type 'man ENCSP-9042' for more detail.
#std cell=9473 (0 fixed + 9473 movable) #block=0 (0 floating + 0 preplaced)
#ioInst=0 #net=12729 #term=34539 #term/net=2.71, #fixedIo=0, #floatIo=0, #fixedPin=0, #floatPin=52
stdCell: 9473 single + 0 double + 0 multi
Total standard cell length = 95.8343 (mm), area = 1.0868 (mm^2)
**Info: (ENCSP-307): Design contains fractional 1 cell.
Average module density = 0.698.
Density for the design = 0.698.
       = stdcell_area 118314 sites (1086761 um^2) / alloc_area 169604 sites (1557881 um^2).
Pin Density = 0.292.
            = total # of pins 34539 / total Instance area 118314.
=== lastAutoLevel = 8
```

Figure 10 Unitilies analysis of Encounter

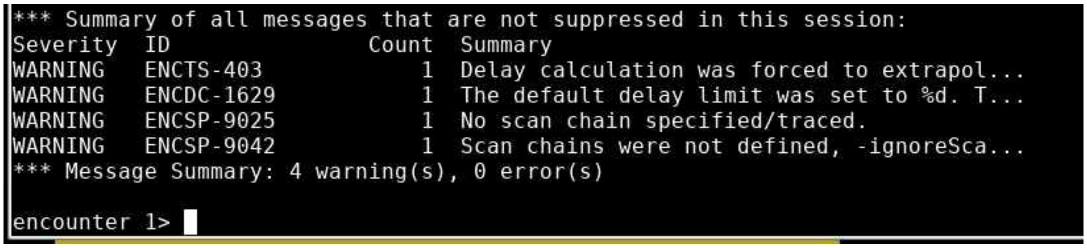
The picture below shows the number of vias and via length at different levels.



```
Congestion distribution:

Remain  cntH            cntV
--------------------------------------
--------------------------------------
  1:    0         0.00%  3        0.01%
  2:    0         0.00%  40       0.11%
  3:    0         0.00%  211      0.59%
  4:    0         0.00%  863      2.42%
  5:    35595   100.00% 34478    96.86%


Total length: 6.333e+05um, number of vias: 77069
M1(H) length: 0.000e+00um, number of vias: 34487
M2(V) length: 1.702e+05um, number of vias: 31325
M3(H) length: 2.851e+05um, number of vias: 11245
M4(V) length: 1.766e+05um, number of vias: 11
M5(H) length: 9.436e+02um, number of vias: 1
M6(V) length: 4.871e+02um

Peak Memory Usage was 655.6M
```

Figure 11 Unitilies Analysis of Encounter

As shown below, encounter shows that our design has no errors.



```
*** Summary of all messages that are not suppressed in this session:
Severity  ID              Count  Summary
WARNING   ENCTS-403           1  Delay calculation was forced to extrapol...
WARNING   ENCDC-1629          1  The default delay limit was set to %d. T...
WARNING   ENCSP-9025          1  No scan chain specified/traced.
WARNING   ENCSP-9042          1  Scan chains were not defined, -ignoreSca...
*** Message Summary: 4 warning(s), 0 error(s)

encounter 1>
```

Figure 12 error-free messages of Encounter

Section 6- Contribution:

Task	**Contributors**
Proposal Discussion	Ben Guan, Chenfeng Zhao, Shu Yang, Enrique Savillo
System Design	Chenfeng Zhao, Ben Guan
Testbench Design	Chenfeng Zhao, Enrique Savillo
Data Format Transformation	Ben Guan,Shu Yang
DC Optimization and Physical Realization	Ben Guan,Chenfeng Zhao
Final Report	Ben Guan,Chenfeng Zhao