A simple Digit classfier is developed using Matlab Deep Learning Toolbox. Below is the internal architecture of this DNN. The validation accuracy of this DNN is 80.25%.

ANALYSIS RESULT

	Name	Type	Activations	Learnables
1	imageinput 28x28x1 images with 'zerocenter' normalization	Image Input	28×28×1	-
2	conv 1 3x3x1 convolutions with stride [1 1] and padding 'same'	Convolution	28×28×1	Weights 3×3 Bias 1×1
3	relu_1 ReLU	ReLU	28×28×1	-
4	maxpool 2x2 max pooling with stride [2 2] and padding [0 0 0 0]	Max Pooling	14×14×1	-
5	fc_1 128 fully connected layer	Fully Connected	1×1×128	Weights 128×196 Bias 128×1
6	relu_2 ReLU	ReLU	1×1×128	-
7	fc_2 128 fully connected layer	Fully Connected	1×1×128	Weights 128×128 Bias 128×1
8	relu_3 ReLU	ReLU	1×1×128	-
9	fc_3 64 fully connected layer	Fully Connected	1×1×64	Weights 64×128 Bias 64×1
10	relu_4 ReLU	ReLU	1×1×64	-
11	fc_4 10 fully connected layer	Fully Connected	1×1×10	Weights 10×64 Bias 10×1
12	softmax softmax	Softmax	1×1×10	-
13	classoutput crossentropyex with '0' and 9 other classes	Classification Output	-	-

Figure 1- Internal architecture of Digit classifier

This is just a sample DNN that includes 1 Convolutional layer and 4 fully-connected layers. Our focus is only on fully-connected layers. The input feature maps and the weights of these 4 fully-connected layers are quantized to 8 bits and written as 1 column into .txt files. The input feature maps are positive but the weights include negative values as well.

To use the .text files, students have to consider the size of each fully-connected layer. For example, for layer 5 (fc_1), the size of the input feature map is 196. Accordingly, the number of rows written in "I_Dig_5_B_8.txt" is 196. The size of the weights for this layer is 128×196 which means that this layer has 128 filters. The "W_Dig_5_B_8.txt" has 128×196 rows, that every 196 rows belong to a filter. In more detail, the output of this layer is the summation of the element-wise multiplication of every 196 rows of "W_Dig_5_B_8.txt" and "I_Dig_5_B_8.txt". Students have to create an output file that has 128 elements (each row is the summation of the element-wise multiplication of each filter in "W_Dig_5_B_8.txt" and the input feature map in "I_Dig_5_B_8.txt"). The same thing is true for other fully-connected layers. The input and weights of different fully connected layers are written in separate files.

The size of the output of the last fully connected layer (fc_4) is 10 since this DNN classifies the given input image as one of the digits 0,…,9. Students should Plot the output of this layer in their report.

If a design has 9 multipliers, the control unit in the testbench should read 9 elements of a filter and the input feature map and feed the design in each cycle. If the number of multipliers is 3 then 3 of the elements are needed in each cycle. The control unit in the testbench should zero pad wherever it is needed. For example, if 5 elements are left and there are 9 multipliers to feed, the control unit in the testbench should feed 4 of the multipliers with zero.

Students have to simulate 2 (out of 4) of these fully-connected layers.